

**UNIVERSIDAD AMAZÓNICA DE PANDO  
ÁREA DE CIENCIAS Y TECNOLOGÍA  
CARRERA DE INGENIERÍA INFORMÁTICA**



**PROYECTO DE GRADO  
SISTEMA WEB DE ADMINISTRACION Y  
PUBLICACIÓN PARA LA INMOBILIARIA  
CASA LINDA**

**PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA  
MENCION: INGENIERIA DE SISTEMAS INFORMATICOS**

**Postulante:** Edwin Mamani Huanca  
**Tutor:** Msc. Lic. Humberto Fernández.  
**Asesor:** Ing. Marco A. Poma Choquehuanca.

**Cobija-Pando- Bolivia  
2012**

## **Dedicatoria**

A Dios por ayudarme e iluminarme, a mi familia en especial a mis padres por haberme brindado el apoyo incondicional en todo momento, a mi hermana Julia por apoyarme y colaborarme en todo y a todos mis hermanos y sobrinos por estar conmigo e impulsarme en el desarrollo de este proyecto.

## **Agradecimientos**

Quiero expresar mi sincero agradecimiento a todos los que formaron parte en el desarrollo y conclusión de este proyecto. Y a todos los que me han impulsado hacia adelante.

De sobremanera a dios, por darme la vida, por todos los dones y virtudes que me ha dado, y permitirme cumplir una de mis principales metas.

A mi tutor Lic. Humberto Fernández Calle, por haberme brindado su paciencia, colaboración, consejos y tiempo necesario para la conclusión de este proyecto.

A mi asesor Ing. Marco Poma, por su pre disponibilidad y tiempo, por sus observaciones, sugerencias, consejos y confianza brindada en la realización de este proyecto.

A la Dr. Cintya Cervantes Claire y Dr. Katherine Shimokawa Von Boeck, representantes de la Inmobiliaria casa linda, por su paciencia y confianza depositada en mi persona para desarrollar este proyecto en dicha empresa.

A mi familia, amigos y compañeros de trabajo por su paciencia y apoyo.

Gracias a todos.

## Resumen

Los sistemas de información bajo ambiente Web se han vuelto rápidamente comunes, Internet ha permitido la migración de aplicaciones de escritorio a aplicaciones Web (multiplataforma) a través de distintas tecnologías, aprovechando este medio; la mayoría de las empresas u organizaciones consideran importante el tener participación en la Red de redes. Estos entes se dan a conocer por medio de páginas Web, sitios Web o aplicaciones Web.

Las aplicaciones hipermedia han evolucionado en los últimos años y se han concentrado mayormente en la web. Las antiguas aplicaciones distribuidas en cd's dieron lugar a aplicaciones dinámicas, de constante actualización e incluso personalizables, capaces de adaptarse a los tipos de usuarios y en casos avanzados, a cada usuario en particular.

Ante este aspecto viendo la necesidad que requiere la Inmobiliaria Casa Linda con Sede Cobija se considera desarrollar e implementar un sistema de administración y publicación de contenidos, optimizada y posicionada en los actuales motores de búsqueda, para brindar de esta manera un mejor servicio y eficiente administración en los procesos del manejo de la información, con el fin de que cualquier persona vinculada o no a la empresa puede acceder a dichos servicios desde su hogar, su lugar de trabajo, y en general desde cualquier punto de conexión a la Internet.

Para el desarrollo del presente trabajo se aplicó la metodología OOHDMM, que propone el desarrollo de sistemas basados en la web. La implementación fue desarrollada en el lenguaje PHP, base de datos MySQL, además de tecnologías, métodos y técnicas relacionadas a conceptos como DOM, SEO, SEM entre otros.

## Índice.

|  |    |
|--|----|
| <b>CAPITULO I</b> .....                              | 8  |
| <b>MARCO REFERENCIAL</b> .....                       | 8  |
| <b>1.1 ANTECEDENTES</b> .....                        | 9  |
| <b>1.2 DESCRIPCIÓN DEL PROBLEMA</b> .....            | 11 |
| <b>1.3 OBJETIVOS</b> .....                           | 13 |
| <b>1.3.1 Objetivo General</b> .....                  | 13 |
| <b>1.3.2 Objetivo Específicos</b> .....              | 13 |
| <b>1.4 JUSTIFICACIÓN</b> .....                       | 14 |
| <b>1.5 METODOLOGÍA</b> .....                         | 15 |
| <b>1.6 ALCANCES</b> .....                            | 17 |
| <b>1.7 APORTES</b> .....                             | 18 |
| <b>CAPITULO II</b> .....                             | 19 |
| <b>MARCO TEORICO</b> .....                           | 19 |
| <b>2.1 DEFINICIONES DE TÉRMINO WEB</b> .....         | 20 |
| <b>2.1.1 Internet</b> .....                          | 20 |
| <b>2.1.2 Nombre de Dominio</b> .....                 | 20 |
| <b>2.1.3 Sistema</b> .....                           | 21 |
| <b>2.1.4 Sitio web</b> .....                         | 21 |
| <b>2.1.5 Página web</b> .....                        | 21 |
| <b>2.1.6 Sistema web</b> .....                       | 22 |
| <b>2.1.7 Software libre</b> .....                    | 22 |
| <b>2.1.8 Web 2.0</b> .....                           | 22 |
| <b>2.2 TECNOLOGÍA WEB</b> .....                      | 23 |
| <b>2.2.1 Servidor web</b> .....                      | 23 |
| <b>2.2.2 Servidor de base de datos (MySql)</b> ..... | 24 |

|   |           |
|---|-----------|
| 2.2.3 Lenguaje de programación PHP.....                   | 25        |
| 2.2.4 Navegadores.....                                    | 26        |
| 2.2.5 SEO.....  | 27        |
| 2.2.6 SMO .....   | 28        |
| 2.2.7 Redes sociales. ....                                | 29        |
| 2.2.7.1 Facebook.....                                     | 29        |
| 2.2.7.1 Twitter.....                                      | 30        |
| 2.2.7.3 Youtube.....                                      | 31        |
| 2.2.8 DOM.....  | 31        |
| 2.2.8.1 HTML. ....  | 32        |
| 2.2.8.2 JavaScript.....                                   | 32        |
| 2.2.9 API. Interfaz de programación de aplicaciones ..... | 33        |
| 2.2.10 API de YouTube.....                                | 33        |
| 2.2.10.1 API de datos.....                                | 33        |
| 2.2.11 Api de Google Maps.....                            | 34        |
| 2.3 cPanel. ....  | 35        |
| 2.4 Arquitectura cliente-servidor.....                    | 36        |
| <b>CAPITULO III.....</b>                                  | <b>38</b> |
| MARCO METODOLÓGICO.....                                   | 38        |
| 3.1 El Patrón MVC .....                                   | 39        |
| 3.2 Ingeniería Web.....                                   | 40        |
| 3.3 El Lenguaje Unificado de Modelado (UML) .....         | 41        |
| 3.4 Metodología OOADM .....                               | 43        |
| <b>CAPITULO IV.....</b>                                   | <b>47</b> |
| MARCO APLICATIVO.....                                     | 47        |
| 4.1 Especificación de los casos de uso.....               | 48        |
| 4.2 Diagramas de casos de uso. ....                       | 58        |
| 4.2.1 Diagrama principal de cada caso de uso.....         | 59        |

|  |    |
|--|----|
| <b>4.2.2 Diagrama de caso de uso general.</b> .....                            | 60 |
| <b>4.2.3 Diagrama de caso de uso usuario administrador.</b> .....              | 60 |
| <b>4.3 Especificación de los diagramas de iteración de los usuarios.</b> ..... | 61 |
| <b>4.4 Diseño del modelo conceptual.</b> .....                                 | 64 |
| <b>4.5 Diseño navegacional.</b> .....  | 65 |
| <b>4.5.1 Esquema de clase navegacional –usuario ocasional.</b> .....           | 65 |
| <b>4.5.2 Esquema de clase navegacional –usuario administrador.</b> .....       | 66 |
| <b>4.6 Diseño de la interfaz abstracta</b> .....                               | 67 |
| <b>4.7 Implementación.</b> .....   | 71 |
| <b>CAPITULO V</b> .....  | 85 |
| <b>CONCLUSIONES Y RECOMENDACIONES</b> .....                                    | 85 |
| <b>5.1 Conclusiones.</b> .....   | 86 |
| <b>5.2 Recomendaciones.</b> .....  | 87 |
| <b>5.3 Referencias Bibliográficas</b> .....                                    | 87 |
| <b>5.4 Anexos</b> .....  | 90 |

**CAPITULO I**  
**MARCO REFERENCIAL**

## 1.1 ANTECEDENTES

En el presente Proyecto de Grado, se reúnen varios intereses intelectuales, laborales, y personales en la utilización de las tecnologías de información y comunicación (TIC), [1] en la utilización de herramientas informáticas, los mismos que serán desarrollados bajo estándares de diseño y concepto de modelado de objetos de un documento así como el modelo vista y controlador para el desarrollo de la visualización del usuario navegante (**Front-end**) así como el panel de administración de sitio (**Back-end**), posterior a ello se aplicara el posicionamiento web bajo el uso de técnicas, así también se optimizara el uso de memoria en disco ya que los videos serán almacenados en la red social YouTube haciendo el uso de su API, [2] que nos ayudaran a definir de manera más confiable los recursos apropiados para el desarrollo del presente proyecto. En el ámbito empresarial ya que como estrategia es que se pretenda por estos medios no solo estar presente en internet sino hacer presencia en internet mediante el uso de las tecnologías apropiadas.

Las instituciones, empresas, modernas de la región como TAHUAMANU, ACRECORP, SIAD, etc. Están en un ámbito público o privado donde se están centrados a mecanismos y formas de poder sentar presencia en su entorno social ya sea para brindar información pública en su conjunto a la comunidad, así como para promocionar su producto ofertado o publicar información relevante a sus intereses por las mismas, por ello se utilizan medios de difusión ya sean como la radio, televisión, periódicos, etc. con el único objetivo de proporcionar información.

La fusión de las computadoras y las comunicaciones ha tenido una profunda influencia en la forma que se atienden las necesidades de las organizaciones y el

viejo modelo de funcionalidad ha sido reemplazado por uno en el cual un número grande o pequeño de computadoras separadas físicamente pero interconectadas a través de tecnologías de comunicación. Por lo cual muchas entidades día a día se adentra a este mundo de las nuevas tecnologías de la web, así mismo por el avance que tuvieron en aspectos como: confiabilidad, velocidad, seguridad, conectividad y la integración de servicios a través de la red mundial ( internet), convirtiéndose en un canalizador mundial de la información en tiempo real en muchos casos, es por ello que las empresas cuentan con sitio web para la publicación y administración de su información, ofreciendo diferentes tipos de servicios a la comunidad internauta.

En el año 2003 a raíz de la necesidad de búsqueda de nuevos mecanismos de difusión las ciudades de Bolivia se Creó la empresa Inmobiliaria Bienes y raíces Bolivia, la cual creó un servicio Web (<http://www.boliviabiensraices.com/>) para compartir información sobre sus inmuebles.

Después de 2 años apareció Bienes Raíces tu Casa (<http://www.bolivia-tucasa.com/>) y unos años después Bolivia inmobiliaria (<http://bolivia.inmobiliaria.com/>) ambos servicios Web ofrecen lo mismo que Empresa Inmobiliaria Bienes y raíces Bolivia. Además de otras dos empresas, mundo anuncio que cuenta con un sitio segmentado, a nivel internacional (<http://www.mundoanuncio.bo/>) y Cobija lo canto con la web, (<http://cobija.locanto.com.bo/>) que ofrecen servicios idénticos a las anteriormente mencionadas. Todas estas empresas ofrecen su servicio Web.

Actualmente la administración de estas webs esta diversificada de manera que es difícil encontrar un sistema eficiente.

Inmobiliaria Casa Linda fue fundada el año 2010 en la ciudad de Cobija, y la dirigen actualmente la Dra. Katherine Shimokawa Von Boek y la Dra. Cintya Cervantes Claire, con experiencia en comercialización y arrendamiento de Bienes inmuebles y especialistas en derecho inmobiliario. Su domicilio desde su fundación se ubica en la Av. 27 de Mayo N°44, frente al Parque Piñata, Casa Linda se fundó con la principal motivación de formar de una verdadera Empresa Profesional de Bienes Raíces.

A un año más de su creación Casa Linda se ha convertido en una empresa líder en la administración de condominios, ofreciendo servicios como venta, alquiler, anticrético y asesoría legal a sus clientes, además actualmente cuenta con un espacio televisivo en Pando Visión Canal 45 los días lunes, miércoles y viernes a horas 15:00.

Actualmente la empresa realiza gran parte de sus actividades de control y administración de forma manual, es decir registrando los inmuebles, clientes y tareas en cuadernos de registro, ocasionando en la mayoría dificultades para controlar estas actividades y tareas. Unas cuantas actividades y tareas son realizadas en computadora usando en especial la aplicación Microsoft Excel, el cual no es suficiente para llevar un control adecuados de la publicación, administración y centralización de la información de manera oportuna y fácil de acceder.

## **1.2 DESCRIPCIÓN DEL PROBLEMA**

En el análisis y la descripción de las necesidades de la empresa Casa linda, verificamos en la situación actual en la que se encuentra y con entrevistas concretas con los involucrados mencionados sus expectativas con respecto al nuevo milenio y

como dicha empresa necesita contar con un sistema informático web que coadyuve en la administración y publicidad de su información con relación a sus clientes y proveedores. Es entonces que encontramos la deficiente administración, organización y publicación de la información general y multimedia de los cuales se identifica lo siguiente:

La inmobiliaria no cuenta con un sistema web para la publicación y administración de contenidos por lo que causa demora en la búsqueda de inmuebles al no contar con un centro de datos disponibles, ausencia de los información de los inmuebles y su descripción, mala clasificación y distribución categórica del tipo de inmueble al que pertenece.

Todo esto genere que los clientes se sientan disconformes, molestias en no contar con información disponible y a la mano, causa mala imagen a la empresa al no contar con un buen servicio para ofrecer a sus clientes.

Ante estos aspectos se constata como problema principal lo siguiente:

**Deficiente administración y publicación de información general en la utilización de medios de difusión orientadas en la utilización de internet, ocasionando un mal servicio del manejo de información en la empresa Inmobiliaria Casa Linda.**

## **1.3 OBJETIVOS**

### **1.3.1 Objetivo General**

Implementar un Sistema web de Administración y Publicación, utilizando las herramientas de desarrollo y la metodología OOHDm, para mejorar el servicio de publicación de la información, acceso rápido y sencillo de clientes y público en general en la Inmobiliaria Casa Linda.

### **1.3.2 Objetivo Específicos**

- Analizar los requerimientos utilizando diagramas de iteración de casos de uso para identificar los actores y las tareas que se realizan en diferentes escenarios.
- Diseñar un esquema conceptual de las entidades que contienen la información y como se relacionan.
- Diseñar la navegación representada por dos esquemas de clases especificación de los nodos y contexto representa las diferentes maneras en la que un nodo puede ser accedido.
- Diseñar la interfaz abstracta que enfoca a los diferentes objetos navegacionales que aparecerán, cuales actividades y transformaciones de la interfaz de presentación.
- Implementar un sistema web basado en normas y estándares, aplicando una metodología de diseño de hipermedia orientada a objetos (OOHDm), incorporando herramientas web 2.0.

## 1.4 JUSTIFICACIÓN

La empresa cuenta con el equipo disponible y recurso necesario para la administración del sistema web, la implementación del mismo será en un servidor externo (plussoluciones.com), por las cuales la empresa acredita adquirir servicios de dominio y alojamiento del sistema web.

El desarrollo del sistema no tiene en gastos por compra de licencias, ya que se emplean herramientas con disponibilidad tipo “software libre” como es el lenguaje de programación PHP para el desarrollo de la aplicación así como el sistema gestor de base de datos MySQL. Se reducen los costes de publicidad, imprenta y envío de información a clientes entre otros. El costo de dominio empresarial (<http://www.inmobiliariacasalinda.com/>) es de 20 \$us anualmente. El costo del Hosting (espacio alquilado para alojar el sistema) es de 50 \$us anualmente con la siguiente característica: 1 GB Espacio Disco, 10 GB Transferencia Mensual, 5 Bases de Datos, Cpanel, debido a que la empresa no cuenta con un servidor propio.

Un sitio web es la imagen que ofrece la institución o empresa en internet. Las mismas cuidan sus intereses y la relación que tienen con sus clientes, por lo mismo es pertinente que el sitio web transmita su seriedad, profesionalismo, trayectoria y compromiso, que refleje la actividad y espíritu de la empresa.

El desarrollo e implementación del sistema web permitirá el aprovechamiento óptimo de los recursos computacionales y las nuevas tecnologías de información.

## 1.5 METODOLOGÍA

Para el desarrollo del proyecto se empleara la Metodología de Diseño Hipermedia Orientada A Objetos (OOHDM), porque este método permite diseñar un sistema con todo lo necesario para programar, utilizando procesos UML.

Esta metodología propone las siguientes fases:

- Análisis de requerimientos.
- Diseño conceptual.
- Diseño navegacional.
- Diseño de la interfaz abstracta.
- Implementación.

En cuanto a las herramientas para el desarrollo del sistema son:

- Lenguaje de programación PHP en su versión 5.3.8 para el desarrollo de aplicaciones.
- Gestor de base de datos MySQL.
- Servidor apache Web Server
- Css, Javascript, Herramientas web 2.0
- APIs de YouTube, Google Maps.

Metodologías para el desarrollo de sistemas de información global:  
análisis comparativo y propuesta:

| <b>Fase</b>         | <b>Productos</b>   | <b>Herramientas</b>   | <b>Mecanismos</b>  | <b>Objetivo de</b>  |
|---------------------|--|---|--|---|
| Diseño conceptual   | Diagrama de Clases, División en subsistemas y relaciones   | Técnicas de modelado O.O, patrones de diseños                                 | Clasificación, agregación, generalización y                    | Modelo semántico de la aplicación                                       |
| Diseño navegacional | Nodos, enlaces, estructuras de accesos, contextos navegacionales y transformaciones navegacionales | Técnicas de modelado O.O, patrones de diseño, diagramas de estados escenarios | Clasificación, agregación, generalización y especialización    | Establecer los recorridos que el usuario puede seguir por               |
| Diseño              | Objetos de interfaz abstracta, respuestas a eventos externos y transformaciones de interfaz        | ADVs, Diagramas de configuración, ADV-Charts y patrones de diseño             | Mapeado entre los objetos de navegación y los objetos visibles | Modelado de los objetos perceptibles por el usuario y de cómo le afecta |
| Implementación      | Aplicación ejecutable  | El entorno del lenguaje de programación                                       | Los ofrecidos por el   | Obtener   |

## 1.6 ALCANCES

El alcance del proyecto se orienta principalmente al desarrollo e implementación de un sistema web y se pudo identificar los siguientes módulos: Servicios web, control y administración de inmuebles y clientes, ofreciendo alcanzar los siguientes servicios:

- Registro, actualización, eliminación y lista de datos de los inmuebles para mejorar las operaciones de control.
- Consultas con segmentación categórica y tipo de inmuebles para facilitar la navegabilidad más distribuida para los usuarios.
- Registro de últimas publicaciones para llevar un mejor control de las publicaciones para poder ofrecer un mejor servicio a los clientes.
- Consultas sobre los reportes y listados para centralizar la información necesaria para la elaboración de reportes y listados de manera automatizada.
- Registro de videos desarrollada con la Api de youtube para en acceso a los videos, y de esta manera optimizar el uso de espacio en disco, al mismo tiempo se viriliza la información de la empresa.
- Registro, eliminación y actualización de clientes.
- Registro de inmuebles mediante la Georeferenciacion utilizando la Api de google Maps, para generar mapas dinámicos de los inmuebles.

## **1.7 APORTES.**

Con el desarrollo e implementación del Sistema Web de Administración y Publicación de Inmuebles se permite a la empresa dar un mejor servicio de publicación, administración, almacenamiento y centralización de los inmuebles para el fácil acceso y búsqueda de los inmuebles por parte de los clientes y usuarios.

Permitiendo de esta manera publicar información propia y mejorando el servicio con actualizaciones permanentes logrando la ampliación de la empresa en el mercado global.

**CAPITULO II**  
**MARCO TEORICO**

El sistema de administración y publicación de inmuebles está orientada tanto a los usuarios, como las tecnologías elegidas para realizar su implementación, así mismo estas tecnologías pueden limitar su funcionalidad, decisiones de diseño equivocadas también pueden limitar su capacidad de reusabilidad. Es por ello que el uso de la metodología que está orientada a este proyecto es de vital importancia para su ejecución, por lo que será ir desglosando conceptos que nos ayuden mejor a entender métodos, técnicas que nos ayudaran a desarrollar el presente proyecto.

## **2.1 DEFINICIONES DE TÉRMINO WEB.**

### **2.1.1 Internet.**

Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, Estados Unidos. (Castells, 2012)

### **2.1.2 Nombre de Dominio.**

Un dominio o nombre de dominio es el nombre que identifica un sitio web. Cada dominio tiene que ser único en Internet. Por ejemplo, "www.masadelante.com" es el nombre de dominio de la página web de más adelante. Un solo servidor web puede servir múltiples páginas web de múltiples dominios, pero un dominio sólo puede apuntar a un servidor. (masadelante, 2002)

### **2.1.3 Sistema.**

(System). Un sistema es un conjunto de partes o elementos organizados y relacionados que interactúan entre sí para lograr un objetivo. Los sistemas reciben (entrada) datos, energía o materia del ambiente y proveen (salida) información, energía o materia.

Un sistema puede ser físico o concreto (una computadora, un televisor, un humano) o puede ser abstracto o conceptual (un software) (Alegsa, 2005)

### **2.1.4 Sitio web.**

En inglés website o web site, un sitio web es un sitio (localización) en la World Wide Web que contiene documentos (páginas web) organizados jerárquicamente. Cada documento (página web) contiene texto y o gráficos que aparecen como información digital en la pantalla de un ordenador. Un sitio puede contener una combinación de gráficos, texto, audio, vídeo, y otros materiales dinámicos o estáticos.

### **2.1.5 Página web.**

Una página web es el nombre de un documento o información electrónica adaptada para la World Wide Web y que puede ser accedida mediante un navegador para mostrarse en un monitor de computadora o dispositivo móvil. Esta información se encuentra generalmente en formato HTML o XHTML, y puede proporcionar navegación a otras páginas web mediante enlaces de hipertexto. Las páginas web frecuentemente incluyen otros recursos como hojas de estilo en cascada, guiones (scripts) e imágenes digitales, entre otros. (Wikipedia, 2011)

### **2.1.6 Sistema web.**

Un sistema web es un sistema computacional remoto que se accesa por Internet.

Físicamente los datos se guardan en una base datos ligada a un servidor web.

Para ingresar o consultar los datos, hay que estar conectado a Internet, usar un browser (Internet Explorer, Firefox) y escribir la dirección del servidor (ej.: <http://www.misdatos.com>) donde está la información almacenada.

La entrada es restringida y es necesario identificarse como usuario para entrar al sistema. La comunicación se hace usando páginas web estilo formulario.

### **2.1.7 Software libre.**

Mantenemos esta definición de software libre para mostrar claramente qué debe cumplir un programa de software en concreto para que se le considere software libre. De vez en cuando modificamos esta definición para clarificarla. El «software libre» es una cuestión de libertad, no de precio. Para entender el concepto, debería pensar en «libre» como en «libre expresión», no como en «barra libre». (j, 2011)

### **2.1.8 Web 2.0**

El término Web 2.0 está asociado a aplicaciones web que facilitan el compartir información, la interoperabilidad, el diseño centrado en el usuario y la colaboración en la World Wide Web. Un sitio Web 2.0 permite a los usuarios interactuar y colaborar entre sí como creadores de contenido generado por usuarios en una comunidad virtual, a diferencia de sitios web donde los usuarios se limitan a la observación pasiva de los contenidos que se ha creado para ellos. Ejemplos de la

Web 2.0 son las comunidades web, los servicios web, las aplicaciones Web, los servicios de red social, los servicios de alojamiento de videos, las wikis, blogs, mashups, etc.

## **2.2 TECNOLOGÍA WEB.**

Al momento de implementar la aplicación web en internet es necesario considerar los siguientes elementos de desarrollo.

- Servidor web.
- Servidor de base de datos
- Lenguajes de desarrollo
- Navegadores
- SEO
- SMO
- DOM

### **2.2.1 Servidor web.**

Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente generando o cediendo una respuesta en cualquier lenguaje o Aplicación del lado del cliente. El código recibido por el cliente suele ser compilado y ejecutado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se utiliza el protocolo HTTP para estas comunicaciones, perteneciente a la capa de

aplicación del modelo OSI. El término también se emplea para referirse al ordenador que ejecuta el programa. (Wikipedia, Wiki, 2012)

### **2.2.2 Servidor de base de datos (MySQL)**

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.<sup>1</sup> MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

El servidor de bases de datos MySQL es muy rápido, seguro, y fácil de usar. Si eso es lo que se está buscando, se le debe dar una oportunidad a MySQL. Se pueden encontrar comparaciones de desempeño con algunos otros manejadores de bases de datos en la página de MySQL.

El servidor MySQL fue desarrollado originalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes y ha estado siendo usado exitosamente en ambientes de producción sumamente exigentes por varios años. Aunque se encuentra en desarrollo constante, el servidor MySQL ofrece hoy un conjunto rico y útil de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL un servidor bastante apropiado para acceder a bases de datos en Internet. (Salazar, 2012)

### **2.2.3 Lenguaje de programación PHP.**

PHP es un lenguaje de programación interpretado (Lenguaje de alto rendimiento), diseñado originalmente para la creación de páginas web dinámicas. Se usa principalmente para la interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

En lugar de usar muchos comandos para mostrar HTML (como en C o Perl), páginas PHP contienen HTML con código incluido en el mismo que hace "algo" (en este caso, mostrar "Hola ¡soy un script PHP!"). El código PHP está entre medio de etiquetas de comienzo y final especiales `<?php` y `?>` que nos permitirán entrar y salir del "modo PHP".

Lo que distingue a PHP de algo lado-cliente como Javascript, es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá los resultados de ejecutar el script, sin ninguna posibilidad de determinar qué código ha producido el resultado recibido. El servidor web puede ser incluso configurado para que procese todos los archivos HTML con PHP y entonces no hay manera que los usuarios puedan saber que tienes debajo de la manga.

Lo mejor de usar PHP es que es extremadamente simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales. No sienta miedo de leer la larga lista de características de PHP, en poco tiempo podrá empezar a escribir sus primeros scripts.

## **Sanitizar y Validar.**

Sanitizar y Validar los datos, son los puntos de más importancia a la hora de programar una aplicación segura, y las que más impacto tienen a la hora de analizar la seguridad, debido que estos datos pueden ser manipulados por cualquier usuario, en muchos casos malintencionados.

### **Que es Sanitizar?**

Es el proceso de aplicar una limpieza exhaustiva a un dato o grupos de datos para su uso. Por ejemplo, **FILTER\_SANITIZE\_EMAIL** quita caracteres que no son los apropiados que contiene una dirección email.

### **Que es Validar?**

Es el proceso utilizado para validar o comprobar si los datos cumplen con ciertos requisitos predefinidos. Por ejemplo, **FILTER\_VALIDATE\_EMAIL** determinar si los datos contienen una dirección válida de correo electrónico, pero sin cambiar los datos en sí. (phpya, 2009)

## **2.2.4 Navegadores**

Un navegador o navegador web (del inglés, web browser) es una aplicación que opera a través de Internet, interpretando la información de archivos y sitios web para que podamos ser capaces de leerla, (ya se encuentre ésta alojada en un servidor.

La funcionalidad básica de un navegador web es permitir la visualización de documentos de texto, posiblemente con recursos multimedia incrustados. Los documentos pueden estar ubicados en la computadora en donde está el usuario, pero

también pueden estar en cualquier otro dispositivo que esté conectado a la computadora del usuario o a través de Internet, y que tenga los recursos necesarios para la transmisión de los documentos (un software servidor web).

### **2.2.5 SEO**

El posicionamiento en buscadores, posicionamiento web u Optimizador de motores de búsqueda (SEO por sus siglas en inglés, de Search Engine Optimization que traduce, 'Optimización para motores de búsqueda') es el proceso de mejorar la visibilidad de un sitio web en los diferentes buscadores, como Google, Bing o Yahoo de manera orgánica, es decir sin pagarle dinero al buscador para tener acceso a una posición destacada en los resultados.

Este posicionamiento se logra de manera natural realizando tareas de optimización en las páginas web. Con el objetivo de aparecer en las primeras posiciones de los buscadores y aumentar el tráfico de visitas en una página web, es conveniente que en los sitios se apliquen tareas de optimización.

La tarea de optimizar la estructura de una web y el contenido de la misma, así como la utilización de diversas técnicas de linkbuilding, linkbaiting o contenidos virales con el objetivo de aparecer en las primeras posiciones de los resultados de los buscadores (cuando un usuario busca por una determinada palabra clave o keyword), es conocida como SEO.

La aplicación de técnicas SEO suele ser más intensa en sitios web con mucha competencia y lo que se pretende con su aplicación es el posicionarse por encima de los competidores por determinadas palabras clave.

Las técnicas SEO pueden ser desmedidas y afectar los resultados naturales de los grandes buscadores por lo que si incumplen las cláusulas y condiciones de uso de los mismos pueden ser consideradas, en algunos casos, como una forma de SPAM.

El trabajo es amplio, ya que el posicionamiento involucra al código de programación, al diseño y a los contenidos. También nos referimos a SEO para definir las personas que realizan este tipo de trabajo. (webmaster, 2012)

### **2.2.6 SMO**

SMO o Social Media Optimization, traducido del inglés como Optimización de los medios sociales, es un término de marketing que hace referencia a la estrategia y conjunto de acciones llevadas a cabo en redes sociales y comunidades online con una finalidad publicitaria o comercial. El término fue creado por Rohit Bhargava y debido a la proliferación de sitios sociales, el tiempo que los usuarios pasan en ellos y los beneficios que aporta en términos de tráfico y posicionamiento en buscadores ha adquirido una gran relevancia por parte de empresas y expertos en marketing digital. Este trabajo es en general realizado por un Social Media Manager, (o Community Manager), o un administrador de un medio social.

Por tanto, el Social Media Manager, tiene como objetivo apoyar a los profesionales y a las empresas, optimizando y conociendo las mejores estrategias de marketing viral y sociales, para la gestión de sus redes sociales y comunidades online (Twitter, Youtube, Facebook, LinkedIn, etc.). Así mismo conoce las tendencias de la web social, sus herramientas de optimización y productividad para aprovechar mejor los recursos de internet.

Tiene en sus manos el poder de la imagen de marca, de las actuaciones, de los productos, de las opiniones y, en definitiva, la presentación ante el cliente en un ámbito cada vez más grande, internet, que, ha superado en tiempo de consumo a la televisión. (Creative, 2009)

### **2.2.7 Redes sociales.**

Las Redes son formas de interacción social, definida como un intercambio dinámico entre personas, grupos e instituciones en contextos de complejidad. Un sistema abierto y en construcción permanente que involucra a conjuntos que se identifican en las mismas necesidades y problemáticas y que se organizan para potenciar sus recursos. (Tim, 2011)

#### **2.2.7.1 Facebook.**

Facebook es un sitio web de redes sociales creado por Mark Zuckerberg y fundado por Eduardo Saverin, Chris Hughes, Dustin Moskovitz y Mark Zuckerberg. Originalmente era un sitio para estudiantes de la Universidad de Harvard, pero actualmente está abierto a cualquier persona que tenga una cuenta de correo electrónico. Los usuarios pueden participar en una o más redes sociales, en relación con su situación académica, su lugar de trabajo o región geográfica.

Ha recibido mucha atención en la blogosfera y en los medios de comunicación al convertirse en una plataforma sobre la que terceros pueden desarrollar aplicaciones y hacer negocio a partir de la red social.

A mediados de 2007 lanzó las versiones en francés, alemán y español principalmente para impulsar su expansión fuera de Estados Unidos, ya que sus

usuarios se concentran en Estados Unidos, Canadá y Reino Unido. Facebook cuenta con más de 900 millones de miembros, y traducciones a 70 idiomas.

Su infraestructura principal está formada por una red de más de 50 000 servidores que usan distribuciones del sistema operativo GNU/Linux usando LAMP

Social media Plugins sociales te permiten ver lo que sus amigos han gustado, comentado o compartido en sitios a través de la web.

### **2.2.7.1 Twitter.**

Twitter es un servicio de microblogging, con sede en San Francisco (California), con filiales en San Antonio (Texas) y Boston (Massachusetts) en Estados Unidos. Twitter, Inc. fue creado originalmente en California, pero está bajo la jurisdicción de Delaware desde 2007.<sup>8</sup> Desde que Jack Dorsey lo creó en marzo de 2006, y lo lanzó en julio del mismo año, la red ha ganado popularidad mundialmente y se estima que tiene más de 200 millones de usuarios, generando 65 millones de tweets al día y maneja más de 800.000 peticiones de búsqueda diarias.

La red permite enviar mensajes de texto plano de corta longitud, con un máximo de 140 caracteres, llamados tweets, que se muestran en la página principal del usuario. Los usuarios pueden suscribirse a los tweets de otros usuarios – a esto se le llama "seguir" y a los suscriptores se les llaman "seguidores"<sup>10</sup> o tweeps<sup>11</sup> ('Twitter' + 'peeps'). Por defecto, los mensajes son públicos, pudiendo difundirse privadamente mostrándolos únicamente a seguidores. Los usuarios pueden twittear desde la web del servicio.

### **2.2.7.3 Youtube.**

Es un sitio web en el cual los usuarios pueden subir y compartir vídeos. Fue creado por tres antiguos empleados de PayPal en febrero de 2005.<sup>4</sup> En octubre de 2006 fue adquirido por Google Inc. a cambio de 1.650 millones de dólares, y ahora opera como una de sus filiales.

YouTube usa un reproductor en línea basado en Adobe Flash (aunque también puede ser un reproductor HTML, que incorporó YouTube poco después de que la W3C presentara y el que los Navegadores web más importantes soportaran el estándar HTML5) para servir su contenido. Es muy popular gracias a la posibilidad de alojar vídeos personales de manera sencilla. Aloja una variedad de clips de películas, programas de televisión y vídeos musicales. A pesar de las reglas de YouTube contra subir vídeos con derechos de autor, este material existe en abundancia, así como contenidos amateur como video blogs. Los enlaces a vídeos de YouTube pueden ser también insertados en blogs y sitios electrónicos personales usando API o incrustando cierto código HTML.

### **2.2.8 DOM**

El Document Object Model o DOM ('Modelo de Objetos del Documento' o 'Modelo en Objetos para la Representación de Documentos') es esencialmente una interfaz de programación de aplicaciones (API) que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos. A través del DOM, los programas pueden acceder y modificar el

contenido, estructura y estilo de los documentos HTML y XML, que es para lo que se diseñó principalmente.

El DOM es una interfaz de programación de aplicaciones para acceder, añadir y cambiar dinámicamente contenido estructurado en documentos con lenguajes como ECMAScript (JavaScript).

#### **2.2.8.1 HTML.**

Siglas de HyperText Markup Language (lenguaje de marcado de hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. El HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

#### **2.2.8.2 JavaScript.**

Es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador.

### **2.2.9 API. Interfaz de programación de aplicaciones**

Es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas (también denominadas vulgarmente "librerías").

#### **2.2.10 API de YouTube**

YouTube en cualquier pantalla, en cualquier momento. Las herramientas y API de YouTube permiten integrar las funciones y los vídeos de YouTube en tu sitio web, en tu aplicación o software, o en tu dispositivo.

##### **2.2.10.1 API de datos**

El API de datos permite a un programa realizar muchas de las operaciones disponibles en el sitio web de YouTube. Es posible realizar búsquedas de vídeos, recuperar feeds estándar y ver contenido relacionado. Un programa también se puede autenticar como un usuario para subir vídeos y modificar las listas de reproducción de un usuario, entre otras muchas cosas.

El API de datos está diseñado principalmente para desarrolladores acostumbrados a programar en lenguajes de servidor. Resulta muy útil para los sitios o para las aplicaciones que quieren una mayor integración con YouTube. Esta integración puede ser una aplicación web que permita a los usuarios subir vídeos a YouTube. También puede ser un dispositivo o una aplicación de escritorio que incluya la experiencia de YouTube en una nueva plataforma. El API de datos ofrece acceso

automático a la información sobre vídeos y sobre usuarios almacenada en YouTube. Con esta API, puedes personalizar tu sitio o tu aplicación con la información existente del usuario, así como llevar a cabo acciones en su nombre (por ejemplo, realizar comentarios y puntuar vídeos). (Google, 2012)

### **2.2.11 Api de Google Maps.**

Google Maps es el nombre de un servicio de Google. Es un servidor de aplicaciones de mapas en la Web. Ofrece imágenes de mapas desplazables, así como fotos satelitales del mundo e incluso la ruta entre diferentes ubicaciones o imágenes a pie de calle Google Street View. Desde el 6 de octubre del 2005, Google Maps es parte de Google Local.

Ofrece, asimismo, la posibilidad de que cualquier propietario de una página Web integre muchas de sus características a su sitio.

### **Georreferenciación**

Georreferenciación<sup>1</sup> es un vocablo que refiere al posicionamiento con el que se define la localización de un objeto espacial (representado mediante punto, vector, área, volumen) en un sistema de coordenadas. Este proceso es utilizado frecuentemente en los Sistemas de Información Geográfica (SIG).

La Georreferenciación, en primer lugar, posee una definición tecnocientífica, aplicada a la existencia de las cosas en un espacio físico, mediante el establecimiento de relaciones entre las imágenes de raster o vector sobre una proyección geográfica o sistema de coordenadas. Por ello la Georreferenciación se convierte en central para los modelados de datos realizados por los SIG.

## **¿Qué es el API de JavaScript de Google Maps?**

El API de Google Maps te permite insertar Google Maps en tus propias páginas web con JavaScript. El API proporciona diversas utilidades para manipular mapas (como la de la página <http://maps.google.com>) y para añadir contenido al mapa mediante diversos servicios, permitiéndote crear sólidas aplicaciones de mapas en tu sitio web.

El API de Google Maps es un servicio gratuito, disponible para cualquier sitio web que sea gratuito para el consumidor. (Google, Developers, 2012)

### **2.3 cPanel.**

cPanel (acrónimo de control Panel) es una herramienta de administración basado en tecnologías web para administrar sitios de manera fácil, con una interfaz limpia. Se trata de un software no libre disponible para un gran número de distribuciones de Linux que soporten RPM, como SuSE, Fedora, Mandriva, etc. El soporte preliminar es para Debian, se debió solicitar que saliera del estado beta por varios años, y para el cual no posee soporte. A cPanel se accede por los puertos 2082 y 2083 (para versiones de SSL). La autenticación es por HTTP o una página de ingreso.

CPanel es el panel de control para empresas de alojamiento web líder en el mercado.

Su atractiva interfaz web es la preferida por los usuarios de miles de compañías en todo el mundo.

Tareas como crear subdominios, añadir cuentas de correo, instalar scripts, proteger directorios, crear bases de datos... son sencillas de realizar por los propios usuarios sin necesidad de la intervención del personal técnico.

Acompañando a CPanel, la aplicación WHM (WebHost Manager) le permite administrar un gran número de funciones de su servidor: creación de cuentas de usuario, manejo de la cola de correo, gestión de paquetes de alojamiento, reinicio de servicios, configuración de los servidores de nombres etcétera. (S.L., 2004)

## **2.4 Arquitectura cliente-servidor**

La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, que le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa. Los tipos específicos de servidores incluyen los servidores web, los servidores de archivo, los servidores del correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma.

Una disposición muy común son los *sistemas multicapa* en los que el servidor se descompone en diferentes programas que pueden ser ejecutados por diferentes computadoras aumentando así el grado de distribución del sistema.

La red cliente-servidor es aquella red de comunicaciones en la que todos los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes cada vez que estos son solicitados. Esto significa que todas las gestiones que se realizan se concentran en el servidor, de manera que en él se disponen los requerimientos provenientes de los clientes que tienen prioridad, los archivos que son de uso público y los que son de uso restringido, los archivos que son de sólo lectura y los que, por el contrario, pueden ser modificados, etc. Este tipo de red puede utilizarse conjuntamente en caso de que se esté utilizando en una red mixta. (Hurtado, 2005)

# **CAPITULO III**

## **MARCO METODOLÓGICO**

### 3.1 El Patrón MVC

MVC: Modelo-Vista-Controlador. Es un patrón de arquitectura de las aplicaciones software, separa la lógica de negocio de la interfaz de usuario, facilita la evolución por separado de ambos aspectos, incrementa reutilización y flexibilidad.

Modelo Vista Controlador (MVC) es un patrón o modelo de abstracción de desarrollo de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos. El patrón de llamada y retorno MVC (según CMU), se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

- **Modelo:** Esta es la representación específica de la información con la cual el sistema opera. En resumen, el modelo se limita a lo relativo de la *vista* y su *controlador* facilitando las presentaciones visuales complejas. El sistema también puede operar con más datos no relativos a la presentación, haciendo uso integrado de otras lógicas de negocio y de datos afines con el sistema modelado.
- **Vista:** Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.
- **Controlador:** Este responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente, a la vista. (Mestras, 2009)

### 3.2 Ingeniería Web.

La Ingeniería Web como nueva disciplina, dan la siguiente definición: “Web Engineering is the establishment and use of sound scientific, engineering and management principles and disciplined and systematic approaches to the successful development, deployment and maintenance of high quality Web-based systems and applications.” Y que escuetamente podemos “traducir” como el proceso utilizado para crear, implantar y mantener aplicaciones y sistemas Web de alta calidad. Esta breve definición nos lleva a abordar un aspecto clave de cualquier proyecto como es determinar qué tipo de proceso es más adecuado en función de las características del mismo.

#### El Proceso de Ingeniería Web

Características como inmediatez y evolución y crecimiento continuos, nos llevan a un proceso incremental y evolutivo, que permite que el usuario se involucre activamente, facilitando el desarrollo de productos que se ajustan mucho lo que éste busca y necesita.

Según Pressman [PRE00], las actividades que formarían parte del marco de trabajo incluirían las tareas abajo enumeradas. Dichas tareas serían aplicables a cualquier aplicación Web, independientemente del tamaño y complejidad de la misma.

Las actividades que forman parte del proceso son: **formulación, planificación análisis, modelización, generación de páginas, test y evaluación del cliente**. La Formulación identifica objetivos y establece el alcance de la primera entrega. La Planificación genera la estimación del coste general del proyecto, la evaluación de

riesgos y el calendario del desarrollo y fechas de entrega. El Análisis especifica los requerimientos e identifica el contenido.

La Modelización se compone de dos secuencias paralelas de tareas. Una consiste en el diseño y producción del contenido que forma parte de la aplicación. La otra, en el diseño de la arquitectura, navegación e interfaz de usuario. Es importante destacar la importancia del diseño de la interfaz.

Independientemente del valor del contenido y servicios prestados, una buena interfaz mejora la percepción que el usuario tiene de éstos. En la

Generación de páginas se integra contenido, arquitectura, navegación e interfaz para crear estática o dinámicamente el aspecto más visible de las aplicaciones, las páginas. El Test busca errores a todos los niveles: contenido, funcional, navegacional, rendimiento, etc. El hecho de que las aplicaciones residan en la red, y que interoperen en plataformas muy distintas, hace que el proceso de test sea especialmente difícil. Finalmente, el resultado es sometido a la evaluación del cliente. (Nieto-Santisteban, 2010)

### **3.3 El Lenguaje Unificado de Modelado (UML)**

El lenguaje UML tiene una notación gráfica muy expresiva que permite representar en mayor o menor medida todas las fases de un proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos, etc., hasta la implementación y configuración con los diagramas de despliegue.

Un diagrama es la representación gráfica de un conjunto de elementos con sus relaciones. En concreto, un diagrama ofrece una vista del sistema a modelar. Para

poder representar correctamente un sistema, UML ofrece una amplia variedad de diagramas para visualizar el sistema desde varias perspectivas. UML incluye los siguientes diagramas:

- Diagrama de casos de uso.
- Diagrama de clases.
- Diagrama de objetos.
- Diagrama de secuencia.
- Diagrama de colaboración.
- Diagrama de estados.
- Diagrama de actividades.
- Diagrama de componentes.
- Diagrama de despliegue.

Los diagramas más interesantes (y los más usados) son los de casos de uso, clases y secuencia, por lo que nos centraremos en éstos.

El **diagrama de casos de usos** representa gráficamente los casos de uso que tiene un sistema. Se define un caso de uso como cada interacción supuesta con el sistema a desarrollar, donde se representan los requisitos funcionales. Es decir, se está diciendo lo que tiene que hacer un sistema y cómo. En la figura 3 se muestra un

ejemplo de casos de uso, donde se muestran tres actores (los clientes, los taquilleros y los jefes de taquilla) y las operaciones que pueden realizar (sus roles).

El **diagrama de clases** muestra un conjunto de clases, interfaces y sus relaciones. Éste es el diagrama más común a la hora de describir el diseño de los sistemas orientados a objetos. En la figura 4 se muestran las clases globales, sus atributos y las relaciones de una posible solución al problema de la venta de entradas.

En el **diagrama de secuencia** se muestra la interacción de los objetos que componen un sistema de forma temporal. Siguiendo el ejemplo de venta de entradas, la figura 5 muestra la interacción de crear una nueva sala para un espectáculo.

El resto de diagramas muestran distintos aspectos del sistema a modelar. Para modelar el comportamiento dinámico del sistema están los de interacción, colaboración, estados y actividades. Los diagramas de componentes y despliegue están enfocados a la implementación del sistema. (Orallo, 2007).

### **3.4 Metodología OOHDM**

Es un Método de Diseño de Desarrollo en Hipermedia Orientado a Objetos (Object-Oriented Hypermedia Design Method) y abarca las cuatro actividades: El modelado conceptual, diseño navegacional, diseño abstracto de interfaz y la puesta en práctica. Estas actividades se realizan en una mezcla de estilo incremental, iterativo y basado en prototipos de desarrollo.

Los modelos orientados a objetos se construyen en cada paso que mejora los modelos diseñados en iteraciones anteriores y consta de las siguientes fases:

Fase Conceptual, durante esta actividad se construye un esquema conceptual representado por los objetos del dominio, las relaciones y colaboraciones existentes establecidas entre ellos. En las aplicaciones hipermedia convencionales, cuyos componentes de hipermedia no son modificados durante la ejecución, se podría usar un modelo de datos semántico estructural (como el modelo de entidades y relaciones). De este modo, en los casos en que la información base pueda cambiar dinámicamente o se intenten ejecutar cálculos complejos, se necesitará enriquecer el comportamiento del modelo de objetos. En OOHD, el esquema conceptual está construido por clases, relaciones y subsistemas. Las clases son descritas como en los modelos orientados a objetos tradicionales. Sin embargo, los atributos pueden ser de múltiples tipos para representar perspectivas diferentes de las mismas entidades del mundo real.

Fase Navegacional, se debe tener en mente que la generación de aplicaciones Web fue pensada para realizar navegación a través del espacio de información, utilizando un simple modelo de datos de hipermedia. En OOHD, la navegación es considerada un paso crítico en el diseño de aplicaciones.

Un modelo navegacional es construido como una vista sobre un diseño conceptual, admitiendo la construcción de modelos diferentes de acuerdo con los diferentes perfiles de usuarios. Cada modelo navegacional provee una vista subjetiva del diseño conceptual. El diseño de navegación es expresado en dos esquemas: el esquema de clases navegacionales y el esquema de contextos navegacionales. En OOHD existe un conjunto de tipos predefinidos de clases navegacionales: nodos, enlaces y estructuras de acceso. La semántica de los nodos y los enlaces son las tradicionales de las aplicaciones hipermedia, y las estructuras de acceso, tales como índices o recorridos guiados, representan los posibles caminos de acceso a los nodos.

La principal estructura primitiva del espacio navegacional es la noción de contexto navegacional. Un contexto navegacional es un conjunto de nodos, enlaces, clases de contextos, y otros contextos navegacionales (contextos anidados). Pueden ser definidos por comprensión o extensión, o por enumeración de sus miembros.

Fase de Interfaz Abstracta, se debe tener las estructuras navegacionales son definidas, se deben especificar los aspectos de interfaz. Esto significa definir la forma en la cual los objetos navegacionales pueden aparecer, de cómo los objetos de interfaz activarán la navegación y el resto de la funcionalidad de la aplicación, qué transformaciones de la interfaz son pertinentes y cuándo es necesario realizarlas.

Una clara separación entre diseño navegacional y diseño de interfaz abstracta permite construir diferentes interfaces para el mismo modelo navegacional, dejando un alto grado de independencia de la tecnología de interfaz de usuario.

Fase Implementación, se tendrá en cuenta que el diseñador debe ya implementar el diseño. Hasta ahora, todos los modelos fueron construidos en forma independiente de la plataforma de implementación; en esta fase es tenido en cuenta el entorno particular en el cual se va a correr la aplicación. Al llegar a esta fase, el primer paso que debe realizar el diseñador es definir los ítems de información que son parte del dominio del problema. Debe identificar también, cómo son organizados los ítems de acuerdo con el perfil del usuario y su tarea; decidir qué interfaz debería ver y cómo debería comportarse. A fin de implementar todo en un entorno Web, el diseñador debe decidir además qué información debe ser almacenada.

OOHDM propone un conjunto de tareas que en principio pueden involucrar mayores costos de diseño, pero

que a mediano y largo plazo reducen notablemente los tiempos de desarrollo al tener como objetivo principal la reusabilidad de diseño, y así simplificar la evolución y el mantenimiento. (Silva, 2001)

**CAPITULO IV**  
**MARCO APLICATIVO.**

En el presente capítulo se describe y desarrolla el análisis y diseño del sistema web de administración y publicación de inmuebles, siguiendo las etapas de la metodología OOADM (Metodología de Diseño Hipermedia Orientada a Objetos), presentada en el capítulo anterior.

#### 4.1 Especificación de los casos de uso.

En esta etapa se presenta la especificación de los casos de uso, que es un documento narrativo que describe la secuencia de eventos que realiza el usuario para completar un proceso, a través del uso de un sistema.

La especificación de los casos de uso se refiere a la descripción de cada una de las partes definidas para lograr su descripción completa. En la organización, la especificación de los Casos de Uso se hará bajo el formato presentado a continuación.

|                                |   |  |
|--------------------------------|---|--|
| <b>CASO DE USO:</b>            | Iniciar sesión.   |  |
| <b>DESCRIPCIÓN:</b>            | El usuario que quiera ingresar a la parte privada del sistema debe contar con un nombre de usuario y contraseña para que sea autenticado. |  |
| <b>ACTORES:</b>                | Administrador   |  |
| <b>PRECONDICIÓN:</b>           | El usuario no se encuentra dado de alta en el sistema.  |  |
| <b>CURSO FORMAL DE EVENTOS</b> |   |  |
| <b>Acción de los Actores</b>   | <b>Respuesta del Sistema</b>  |  |

|  |   |
|--|---|
| <p>1. Navega a la página de logueo (acceso a usuarios).</p> <p>2. Ingresa el usuario y la contraseña y envía los datos.</p>  | <p>3. Toma y valida los datos.</p> <p>4. Autentifica los datos.</p> <p>5. Re direcciona a la página principal con un menú de opciones</p> |
| <b>CURSOS ALTERNOS</b>   |   |
| <p>3a. si una de los campos está vacío, regresa a la página de logueo y despliega un error pidiéndole que se llenen los datos faltantes.</p> <p>4a. si algunos de los datos no es correcto, ingresa a la página de logueo y despliega mensaje de error informándole que el nombre de usuario o la contraseña, es incorrecto.</p> |   |

Tabla 01-iniciar sesión  
Fuente elaboración propia

|                                |  |
|--------------------------------|--|
| <b>CASO DE USO:</b>            | Administrar usuarios   |
| <b>DESCRIPCIÓN:</b>            | El usuario puede ingresar al sistema para administrar los usuarios |
| <b>ACTORES:</b>                | Administrador  |
| <b>PRECONDICIÓN:</b>           | Se ha desarrollado el caso de uso "Iniciar sesión" exitosamente    |
| <b>CURSO FORMAL DE EVENTOS</b> |  |

| <b>Acción de los Actores</b>  | <b>Respuesta del Sistema</b>   |
|-------------------------------|--|
| Ingresa a la sección usuarios | 2. muestra en pantalla la opción: Nuevo usuario (para añadir usuarios).<br><br>3. Despliega la lista de usuarios para poder ser actualizados |
| <b>CURSOS ALTERNOS</b>        |  |
| Ninguno                       |  |

Tabla CU-02 –Administrar usuarios  
Fuente elaboración propia

| <b>CASO DE USO:</b>  | Agregar cliente  |
|--|--|
| <b>DESCRIPCIÓN:</b>  | El usuario desea agregar clientes en la base de datos  |
| <b>ACTORES:</b>  | Usuario  |
| <b>PRECONDICIÓN:</b>   | Se ha desarrollado el caso de uso “iniciar sesión” exitosamente  |
| <b>CURSO FORMAL DE EVENTOS</b>   |  |
| <b>Acción de los Actores</b>   | <b>Respuesta del Sistema</b>   |
| 1. Solicita el proceso de agregar nuevo cliente.<br><br>3. Introduce los datos del nuevo cliente | 2. solicita los datos requeridos del nuevo cliente, nombre, apellidos, ci, municipio, dirección, celular, email. |

|  |   |
|--|---|
| y solicita al sistema que los almacene   | 4. toma y valida los datos<br><br>3. Informa que el proceso ha terminado con éxito. |
| <b>CURSOS ALTERNOS</b>   |   |
| 4a. El sistema detecta que faltan datos para poder realizar la operación, y envía un mensaje de que los datos marcados con asterisco son obligatorios. |   |
| 4b. El sistema detecta que los datos son inválidos para para poder realizar la operación.  |   |

Tabla 03-agregar usuario  
Fuente elaboración propia

|                                |  |
|--------------------------------|--|
| <b>CASO DE USO:</b>            | Agregar inmueble   |
| <b>DESCRIPCIÓN:</b>            | El usuario desea agregar inmuebles en la base de datos   |
| <b>ACTORES:</b>                | Usuario  |
| <b>PRECONDICIÓN:</b>           | Se ha desarrollado el caso de uso “iniciar sesión” exitosamente, y se ha ingresado como mínimo un cliente al registro. |
| <b>CURSO FORMAL DE EVENTOS</b> |  |
| <b>Acción de los Actores</b>   | <b>Respuesta del Sistema</b>   |

|   |   |
|---|---|
| <p>1. Solicita el proceso de agregar nuevo inmueble.</p> <p>3. Introduce los datos del nuevo inmueble y solicita al sistema que los almacene</p>  | <p>2. solicita los datos requeridos del nuevo id_cliente, categoría, situación, descripción, municipio, dirección, titulo, superficie del terreno, precio, foto1, foto2, foto3, estado, fecha.</p> <p>4. toma y valida los datos</p> <p>5. Informa que el proceso ha terminado con éxito.</p> |
| <p><b>CURSOS ALTERNOS</b></p>   |   |
| <p>4a. El sistema detecta que faltan datos para poder realizar la operación, y envía un mensaje de que los datos marcados con asterisco son obligatorios.</p> <p>4b. El sistema detecta que las imágenes no son .jpg, redirección y envía un mensaje de error de q solo puede ingresar imágenes jpg.</p> <p>5a. El sistema redimensiona y asigna un nuevo tamaño a las fotografías para almacenar las en un carpeta</p> |   |

Tabla 04-agregar inmueble  
Fuente elaboración propia

|                            |   |
|----------------------------|---|
| <p><b>CASO DE USO:</b></p> | <p>Georeferenciar ubicación del inmueble</p>                |
| <p><b>DESCRIPCIÓN:</b></p> | <p>El usuario desea agregar la ubicación geográfica del</p> |

|  |   |
|--|---|
|  | inmuebles en la base de datos   |
| <b>ACTORES:</b>  | Usuario   |
| <b>PRECONDICIÓN:</b>   | Se ha desarrollado el caso de uso “iniciar sesión” exitosamente, y se ha ingresado el último registro del inmueble.   |
| <b>CURSO FORMAL DE EVENTOS</b>   |   |
| <b>Acción de los Actores</b>   | <b>Respuesta del Sistema</b>  |
| <p>1. Solicita el proceso de georeferenciar nuevo inmueble.</p> <p>3. Introduce los datos para georeferenciar el nuevo inmueble y solicita al sistema que los almacene</p>   | <p>2. solicita los datos requeridos del nuevo título, latitud, longitud, dirección, descripción, foto,</p> <p>4. toma y valida los datos</p> <p>5. Informa que el proceso ha terminado con éxito.</p> |
| <b>CURSOS ALTERNOS</b>   |   |
| <p>4a. El sistema detecta que ya fue georeferenciado el ultimo inmueble y le redirección a insertar un nuevo inmueble a georeferenciar.</p> <p>4b. El sistema detecta que las imágenes no son .jpg, redirección y envía un mensaje</p> |   |

de error de q solo puede ingresar imágenes jpg.

5a. El sistema redimensiona y asigna un nuevo tamaño a las fotografías para almacenar las en un carpeta

Tabla 05-georeferenciar inmueble  
Fuente elaboración propia

|   |  |  |
|---|--|--|
| <b>CASO DE USO:</b>   | Agregar video  |  |
| <b>DESCRIPCIÓN:</b>   | El usuario desea agregar parámetros del video en la base de datos  |  |
| <b>ACTORES:</b>   | usuario  |  |
| <b>PRECONDICIÓN:</b>  | Se ha desarrollado el caso de uso “iniciar sesión” exitosamente, y se ha ingresado como mínimo un inmueble al registro.<br><br>Contar con una cuenta el google y una canal de youtube y registrar el sitio web para adquirir la apikey de youtube. |  |
| <b>CURSO FORMAL DE EVENTOS</b>  |  |  |
| <b>Acción de los Actores</b>  | <b>Respuesta del Sistema</b>   |  |
| 1. Solicita el proceso de agregar nuevo video.<br><br>3. Introduce los datos del nuevo inmueble y solicita al sistema que los | 2. solicita los datos requeridos del nuevo video, titulo, descripción,<br><br>4. toma y valida los datos   |  |

|   |   |
|---|---|
| <p>almacene</p> <p>5. introduce el video en formato mp4, avi, mpg</p>   | <p>6. toma y valida el video</p> <p>7. Informa que el proceso ha terminado con éxito.</p> |
| <p><b>CURSOS ALTERNOS</b></p>   |   |
| <p>4a. El sistema detecta que ya fue asociado el ultimo inmueble al video y lo redirección a ingresar nuevo inmueble.</p> <p>4b. El sistema detecta que faltan datos para poder realizar la operación, y envía un mensaje de que los datos marcados con asterisco son obligatorios.</p> <p>6a. El sistema detecta que no cuenta con una api key</p> <p>6c. El sistema detecta que el video no son .del formato correcto, redirección y envía un mensaje de error.</p> |   |

Tabla 06-agregar video inmueble  
Fuente elaboración propia

|                     |   |
|---------------------|---|
| <b>CASO DE USO:</b> | Catálogo de inmuebles   |
| <b>DESCRIPCIÓN:</b> | Permitir ver el catálogo de inmuebles aprobados, para su visita en web. |
| <b>ACTORES:</b>     | Usuario, cliente,   |

|   |   |  |
|---|---|--|
| <b>PRECONDICIÓN:</b>  | Ingresar a la categoría que corresponda de los inmuebles.   |  |
| <b>CURSO FORMAL DE EVENTOS</b>  |   |  |
| <b>Acción de los Actores</b>  | <b>Respuesta del Sistema</b>  |  |
| 1. Solicita la página de catálogo de inmuebles                                | 2. El usuario puede ver el catálogo de inmuebles, desplegados en resultados de 10<br><br>Registros por página con botones e hipervínculos de navegación interna. Cada inmueble se muestra con información detallada, fotos video, precio y características generales. |  |
| <b>CURSOS ALTERNOS</b>  |   |  |
| 2a. El sistema detecta que no existes publicaciones asociadas a esa categoría |   |  |

Tabla 07-Catalogo de inmueble  
Fuente elaboración propia

|                     |  |
|---------------------|--|
| <b>CASO DE USO:</b> | Información Detallada  |
| <b>DESCRIPCIÓN:</b> | Permitir acceder la información a detalle de la Propiedad y su respectivo menú de navegación, contacto visual, ubicación |

|   |   |
|---|---|
| <b>ACTORES:</b>   | Usuario, cliente,   |
| <b>PRECONDICIÓN:</b>  | Ingresar al catálogo que corresponda a los inmuebles.   |
| <b>CURSO FORMAL DE EVENTOS</b>                              |   |
| <b>Acción de los Actores</b>                                | <b>Respuesta del Sistema</b>  |
| 1. Solicita la página de información detallada de inmuebles | 2. El usuario podrá acceder a cualquiera de nuestros inmuebles disponibles para ver información específica de su interés respecto del mismo. Así también podrá ver su mapa de ubicación generado con la api de google maps y el video enlazado de youtube |
| <b>CURSOS ALTERNOS</b>                                      |   |
| Ninguna   |   |

Tabla 08-informacion detallada  
Fuente elaboración propia

|                      |  |
|----------------------|--|
| <b>CASO DE USO:</b>  | Mapa de Ubicación  |
| <b>DESCRIPCIÓN:</b>  | Permitir al usuario conocer la ubicación exacta del inmueble |
| <b>ACTORES:</b>      | Usuario, cliente,  |
| <b>PRECONDICIÓN:</b> | Ingresar al catálogo que corresponda a los inmuebles.        |

| <b>CURSO FORMAL DE EVENTOS</b>             |   |
|--|---|
| <b>Acción de los Actores</b>               | <b>Respuesta del Sistema</b>  |
| 1. Solicita la página de mapa de inmuebles | 2. El usuario puede ver la ubicación exacta del inmueble en un mapa navegable generado por google maps. Gracias al uso de esta tecnología de elaboración de mapas es posible crear una aplicación o sitio que ofrezca toda la funcionalidad necesaria |
| <b>CURSOS ALTERNOS</b>                     |   |
| Ninguna                                    |   |

Tabla 09-mapa de ubicación  
Fuente elaboración propia

#### **4.2 Diagramas de casos de uso.**

Se presenta los diagramas de casos de uso en las siguientes figuras, que representan la funcionalidad del sistema web, mostrando su iteración y la representación se hace a través de las relaciones entre los actores y los casos de uso (acciones) dentro del sistema.

#### 4.2.1 Diagrama principal de cada caso de uso.

Definido por los actores que interactúan con el sistema se consideran dos aspectos importantes.

1. La implementación de la zona pública, que sea accesible por cualquier usuario, donde pueda encontrar información actualizada de noticias y otros servicios gratuitos que presta la empresa.
2. La implementación de la zona privada que permita, por un lado la modificación de la estructura del contenido de la zona de carácter publica y por otro lado realizar tareas de control de clientes e inmuebles.

A partir de esto, se construye un diagrama de casos de uso especificando la iteración, clasificación y estableciendo el modo de acceso al sistema de los usuarios.

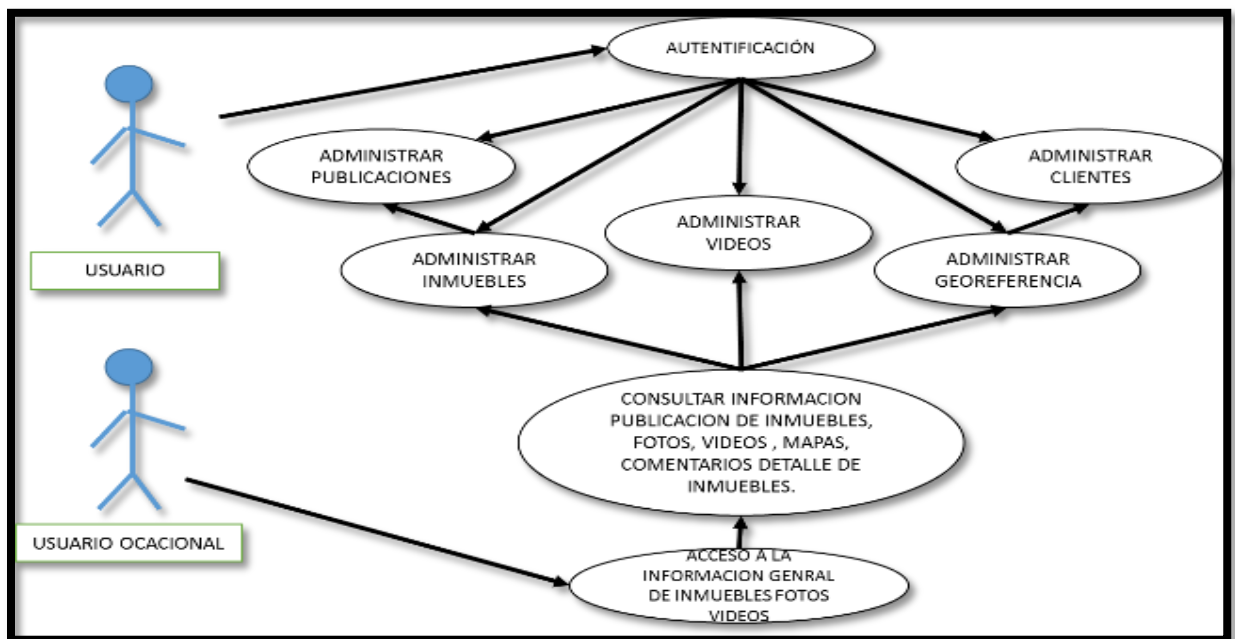


Figura 01 Diagrama -caso de uso  
Fuente elaboración propia

#### 4.2.2 Diagrama de caso de uso general.

El siguiente diagrama de casos de uso representa la funcionalidad general del sistema mostrando la iteración con el usuario visitante.

Es decir que el usuario ocasional del lado del servidor estará a disposición de los nodos publicados para su acceso y navegabilidad dentro del front end.

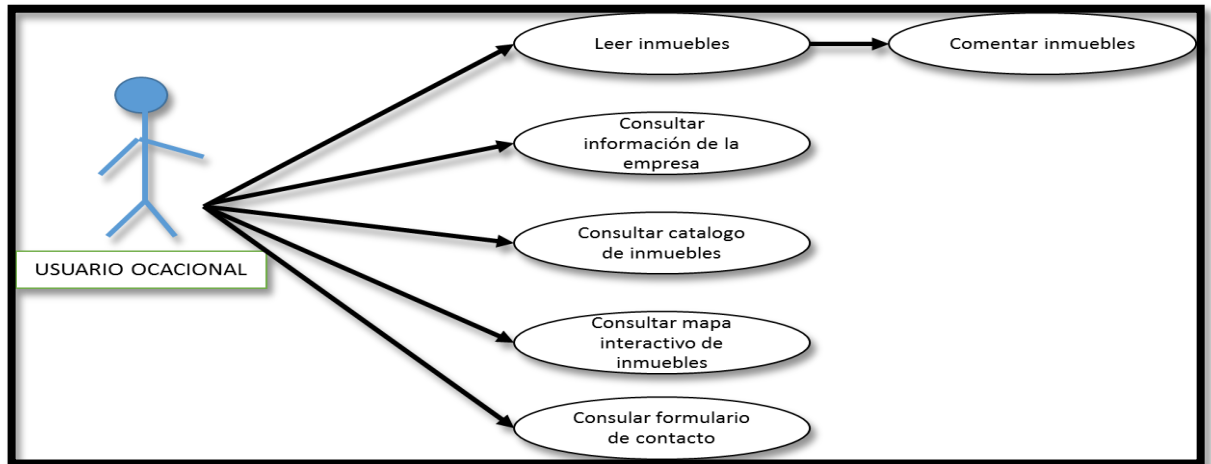


Figura 02 Diagrama -caso de uso General  
Fuente elaboración propia

#### 4.2.3 Diagrama de caso de uso usuario administrador.

En el siguiente diagrama de casos de uso representa la funcionalidad parcial del sistema mostrando la iteración con el administrador

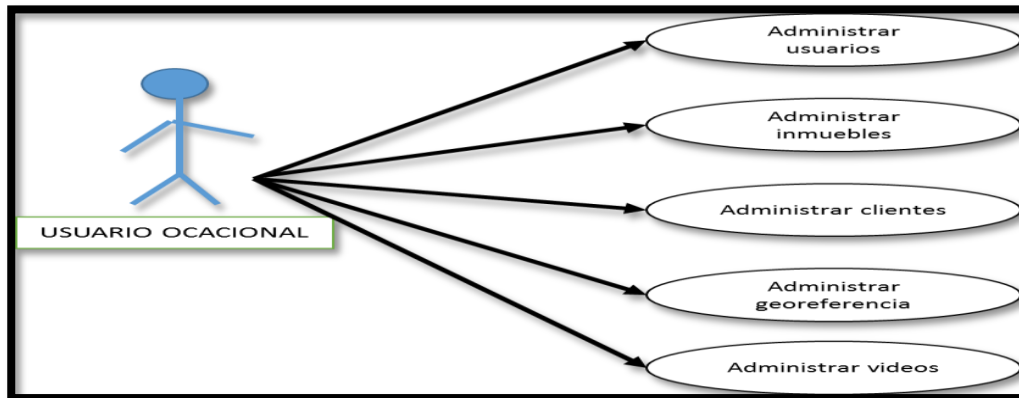


Figura 03 Diagrama -caso de uso Parcial  
Fuente elaboración propia

### 4.3 Especificación de los diagramas de iteración de los usuarios.

En el diagrama de iteración de usuario, se puede apreciar la iteración del usuario con el sistema web, pues se observa los datos que el sistema permitirá ver al usuario y la información que le permite ingresar, además del comportamiento que tendrá en caso de alguna acción o algún error.

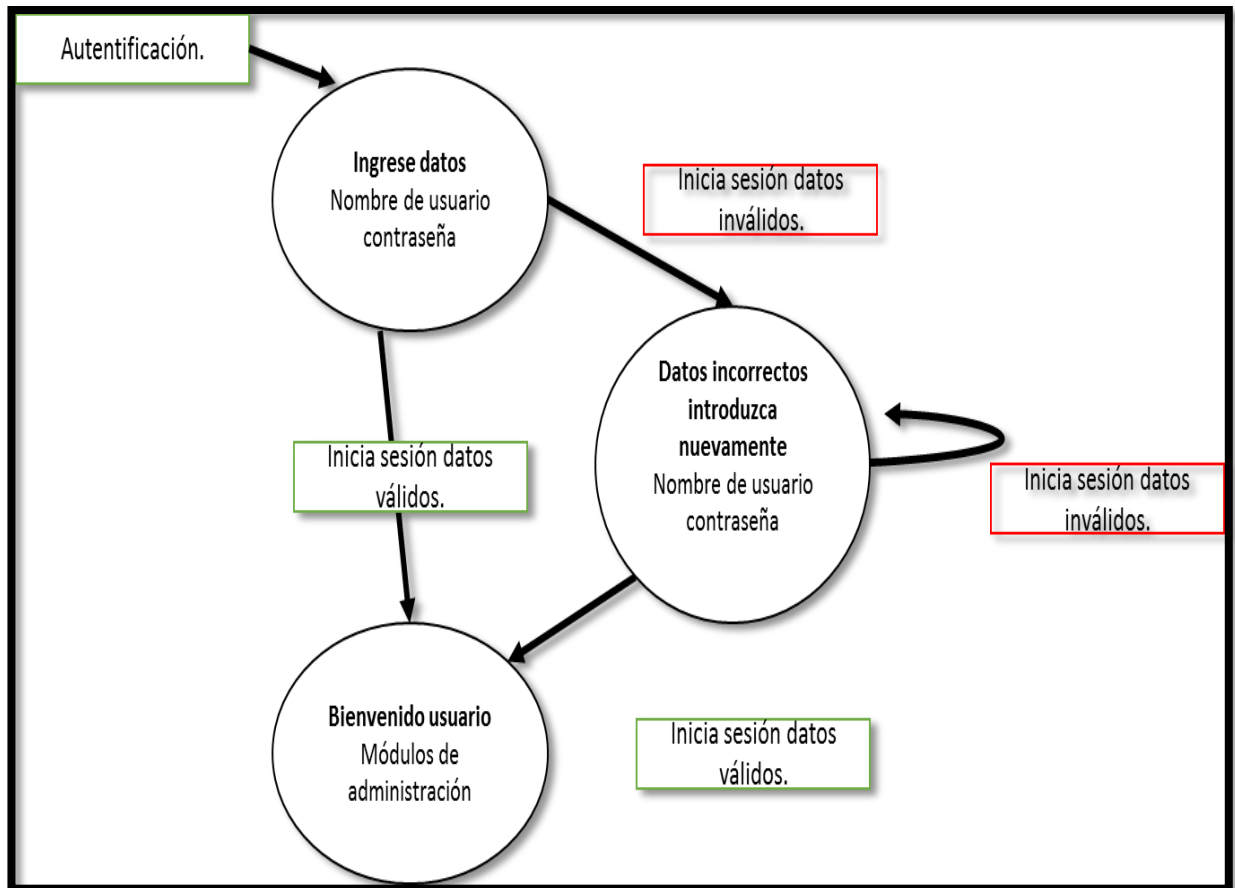


Figura 04 UID –identificación de usuario  
Fuente elaboración propia

En el siguiente diagrama de iteración de usuarios se muestra las posibles intervenciones que realiza el administrador al momento de registrar un cliente y las posibles opciones mensajes que tendría el procedimiento al realizar alguna acción por parte del administrador.

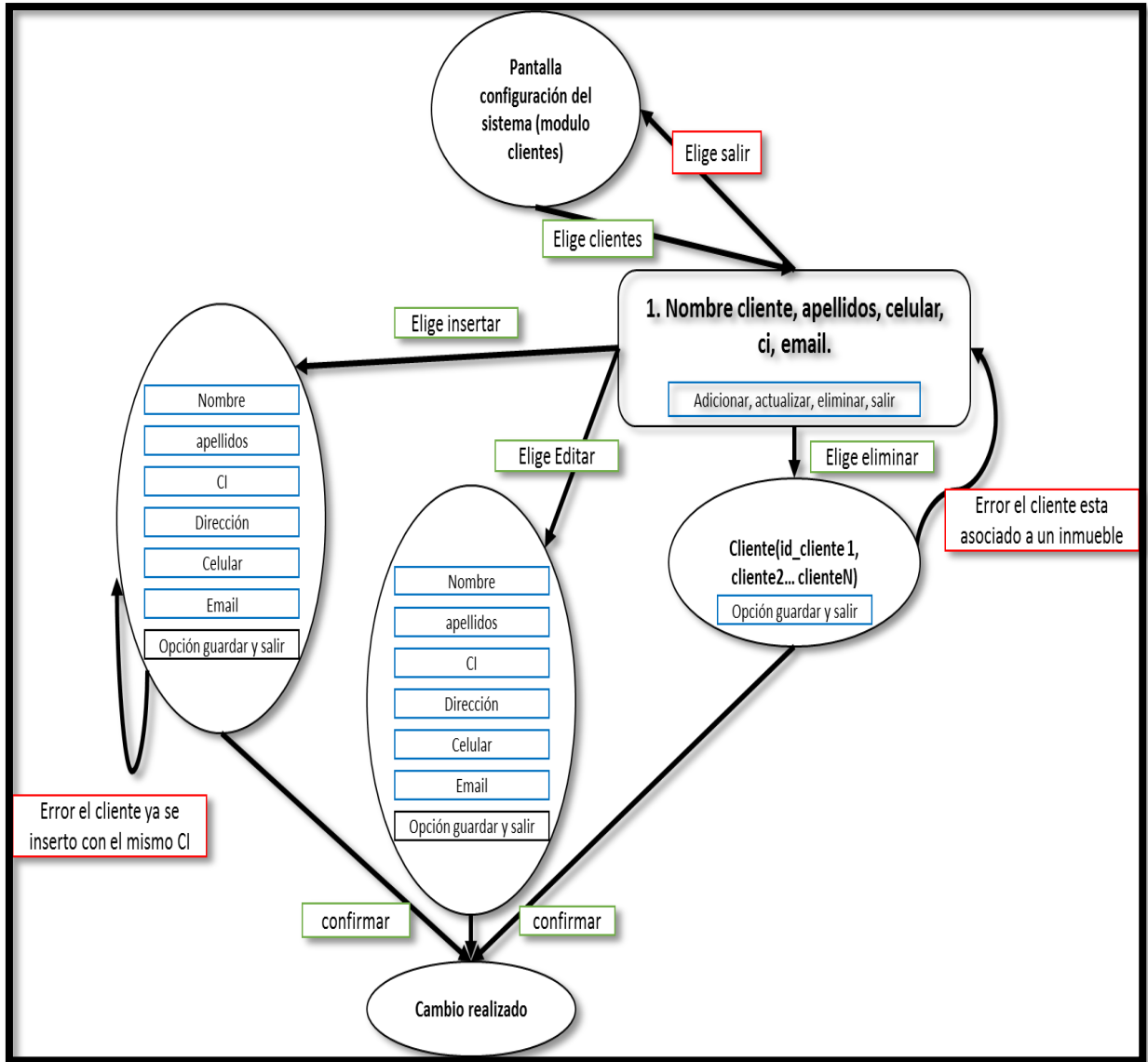


Figura 05 UID –administrar clientes  
Fuente elaboración propia

El siguiente diagrama de iteración muestra el procedimiento de la administración de inmuebles las acciones las cuales está habilitada para su administración, y su posterior publicación en nuestro Front end,

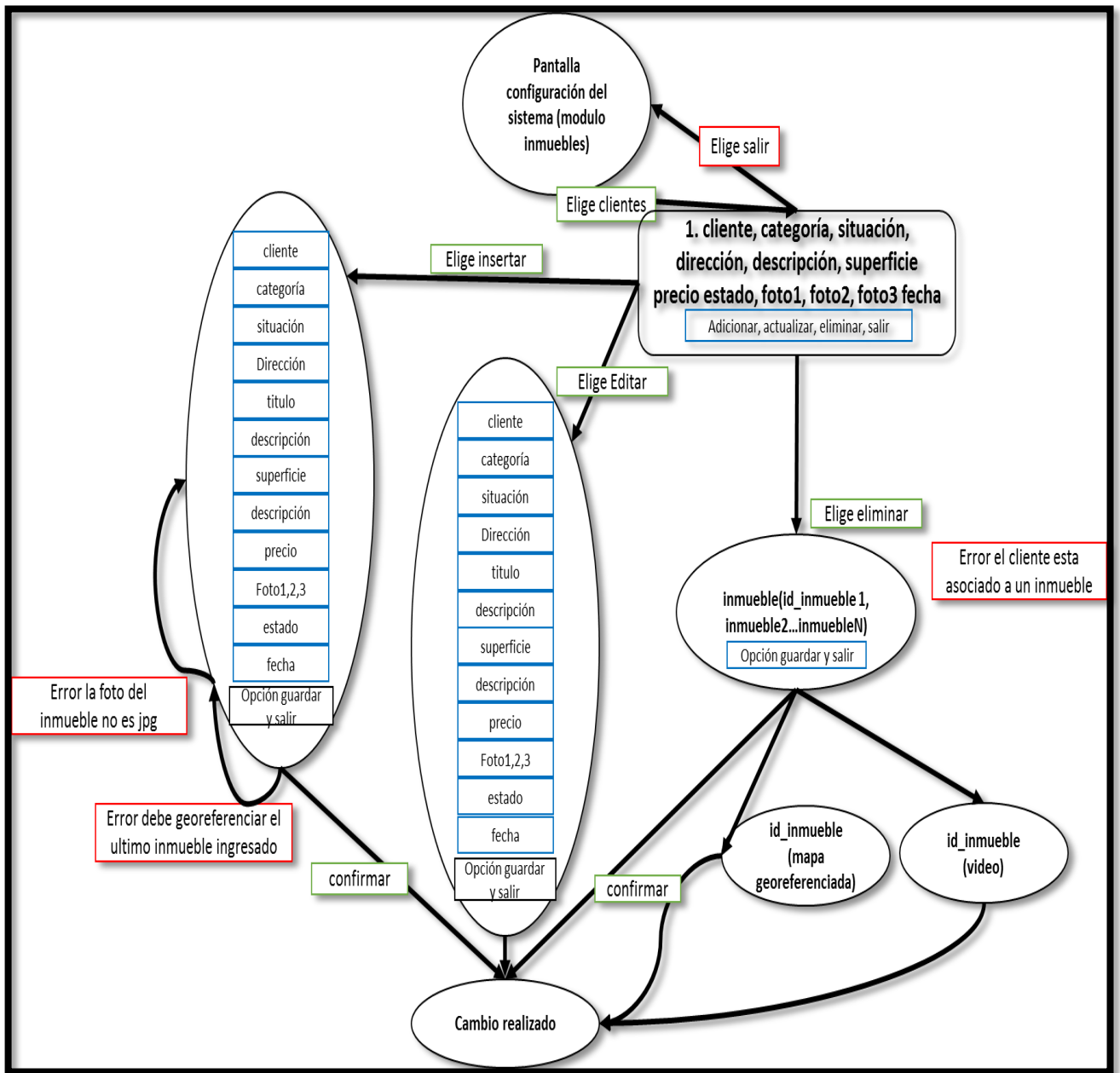


Figura 05 UID –administrar Inmuebles  
Fuente elaboración propia

#### 4.4 Diseño del modelo conceptual.

Durante esta actividad se concibe la aplicación en términos de objetos y las relaciones existentes entre ellos, utilizando UML. Esto es decir diagramas de clases para representar abstractamente la aplicación. El diseño conceptual tiene como producto el esquema conceptual de clases. En OOADM, el esquema conceptual está construido por clases, relaciones y subsistemas.

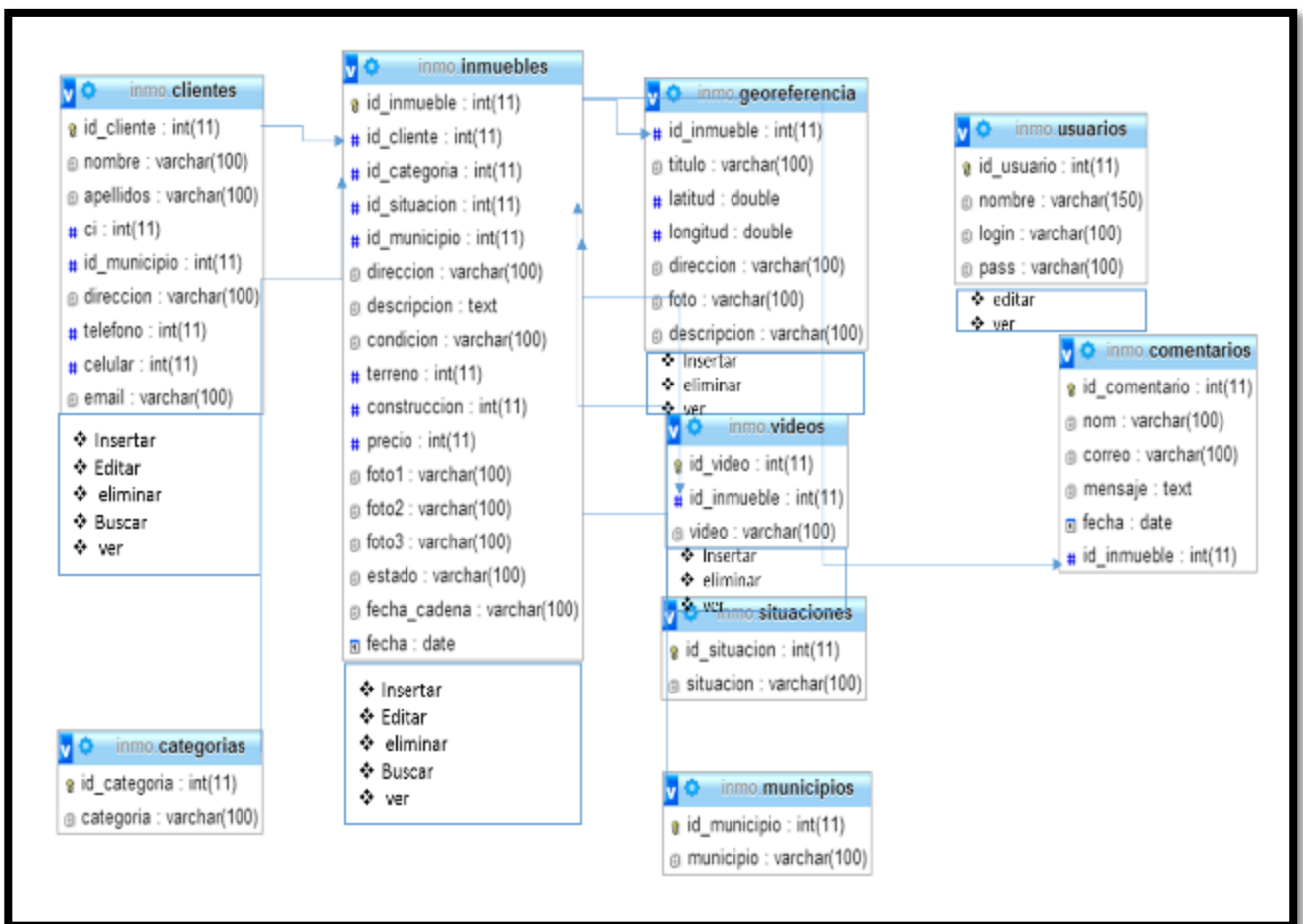


Figura 06 Modelo conceptual del sistema  
Fuente elaboración propia

## 4.5 Diseño navegacional.

En esta fase, OOHDM admite la construcción de dos modelos (el esquema de clases navegacional y el esquema de contexto navegacional).

Se definirá la información que esta presentada y la posible navegación entre ellas. A que estructura el usuario tendrá acceso: nodos, representado por rectángulos. Por donde el usuario podrá navegar: enlaces representado por líneas. Finalmente los puntos de partida para cada acceso: anclas e índices.

### 4.5.1 Esquema de clase navegacional –usuario ocasional.

Se presenta la notación de la metodología OOHDM (Object-Oriented Hypermedia Design Method ), que modela conceptualmente un sitio web

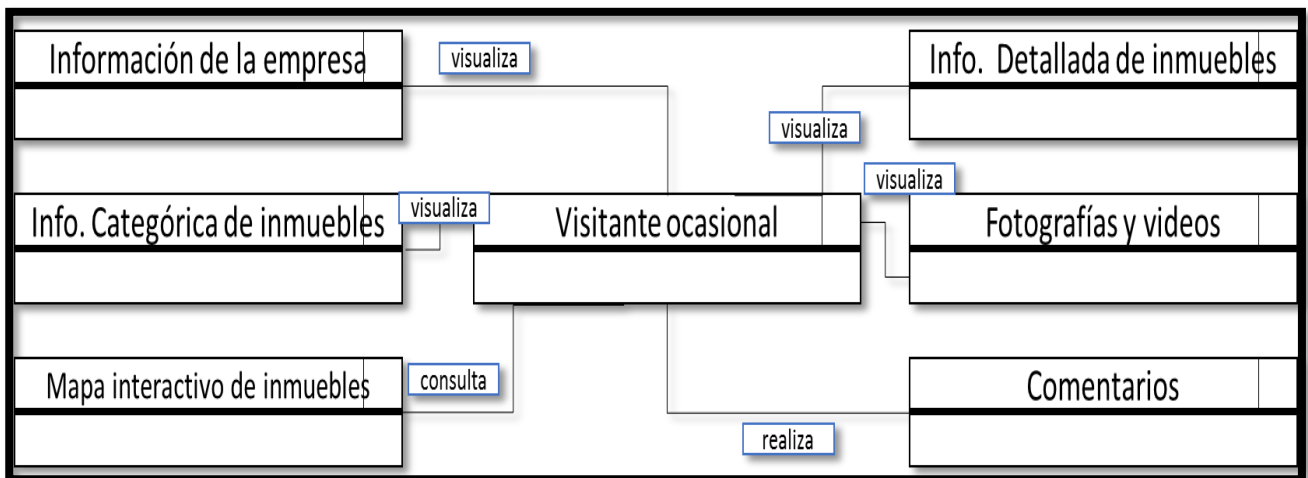


Figura 07-Esquema de clase navegacional usuario ocasional  
Fuente elaboración propia

Utilizando UML (Unified Modeling Language) y propone un modelo complementario para representar el modelamiento navegacional, que se basa principalmente en el respectivo Diagrama de Clases. Debido a que UML no trata el modelamiento navegacional, se desarrolla la notación del esquema navegacional basándose en UML

#### 4.5.2 Esquema de clase navegacional –usuario administrador.

Al igual que el esquema de clase del usuario nacional se representa los procedimientos e iteración que hace el usuario con relación al sistema web de la inmobiliaria, mediante los nodos correspondientes y el acceso que tendrá al administrar algún inmueble.

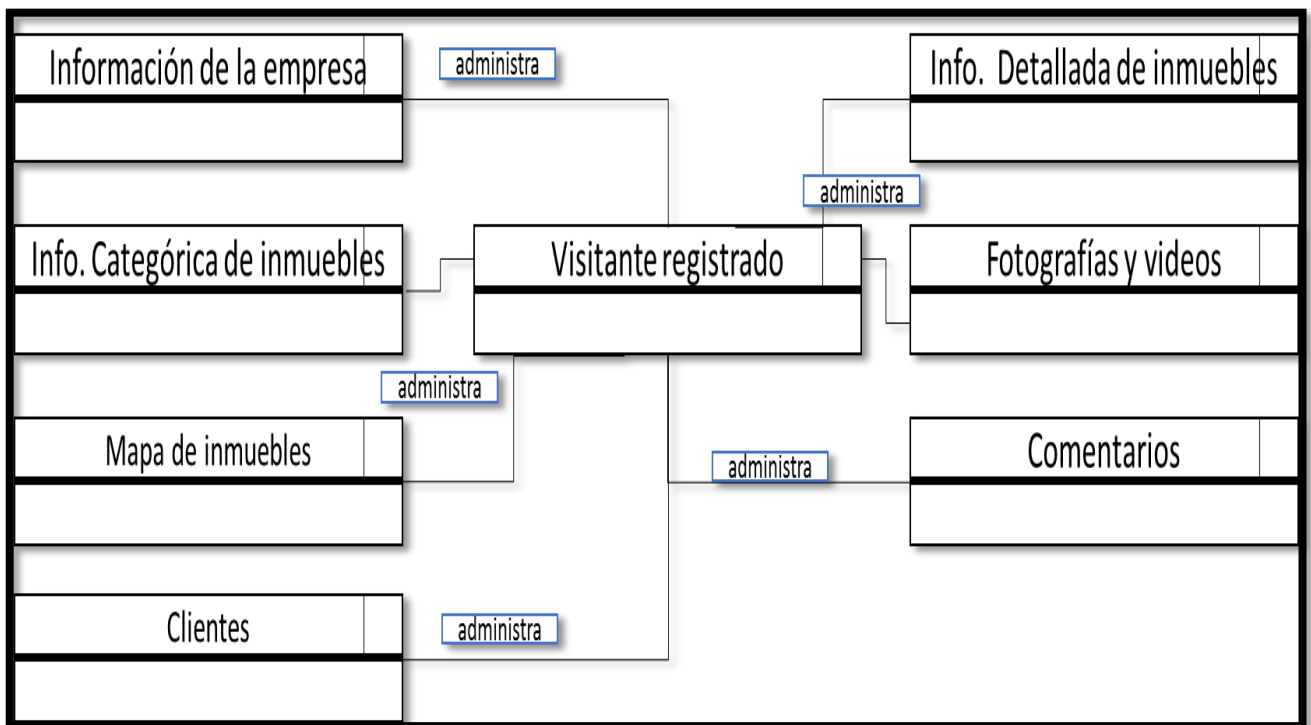


Figura 08-Esquema de clase navegacional usuario administrador  
Fuente elaboración propia

## **4.6 Diseño de la interfaz abstracta**

La fase del diseño de la interfaz abstracta se define del sistema web y el sistema de administración denominada como (back end y front end). Reconociendo los diferentes objetos de la interfaz de los cuales se activaran vista de datos u otros objetos de la interfaz, además se verificara donde y cuando se realizan las vistas o llamadas otros objetos de la interfaz.

### **Diseño de la interfaz abstracta administrador.**

Una vez que las aplicaciones de estructura navegacional han sido definidos, se debe especificar ahora aspectos de la interfaz. Esto significa definir la manera en que diferentes objetos de navegación aparecerán, qué objetos de navegación de la interfaz se activara y otra funcionalidad de aplicación, y qué transformaciones de la interfaz tendrán lugar y cuando.

Una separación ordenada entre ambas preocupaciones, de navegación y diseño de interfaz abstracta, permite construir interfaces diferentes para el mismo modelo de navegación, llevando a un grado más alto de independencia de tecnología de la interfaz de usuario. En suma, esta separación permite entender mejor la aplicación global de la estructura para indicar qué transformaciones claramente en la interfaz serán transformaciones navegacionales.

Aunque se ha discutido que el aspecto de la interfaz de usuario de aplicaciones interactivas (en particular las aplicaciones de la web) es un componente crítico, moderno, las metodologías tienden a descuidar este aspecto. Ellos relegan la especificación para herramientas de implementación-dependientes, y por consiguiente las decisiones de diseño en este nivel raramente se documentan. Es más, como llevar a cabo la interfaz de la Web normalmente se hacen aplicaciones por medio de los editores de HTML especializados, muchos críticos pueden ignorar aspectos de la interfaz.

En OOHDM, se usa un acercamiento del Diseño de Datos de Vista Abstractos (ADVs), para describir la interfaz del usuario de una aplicación de hipermedia. ADVs son objetos en los que tienen un estado y una interfaz, donde la interfaz puede ser ejercido a través de mensajes (en particular, eventos externos generados por el usuario). Las ADVs son abstractas en el sentido de que ellos sólo representan la interfaz y el estado, y no la aplicación.

El ADV administrador se muestra en la siguiente figura, este presenta la interfaz de navegación que se muestra luego de autenticarse, está bajo previa verificación del usuario registrado.

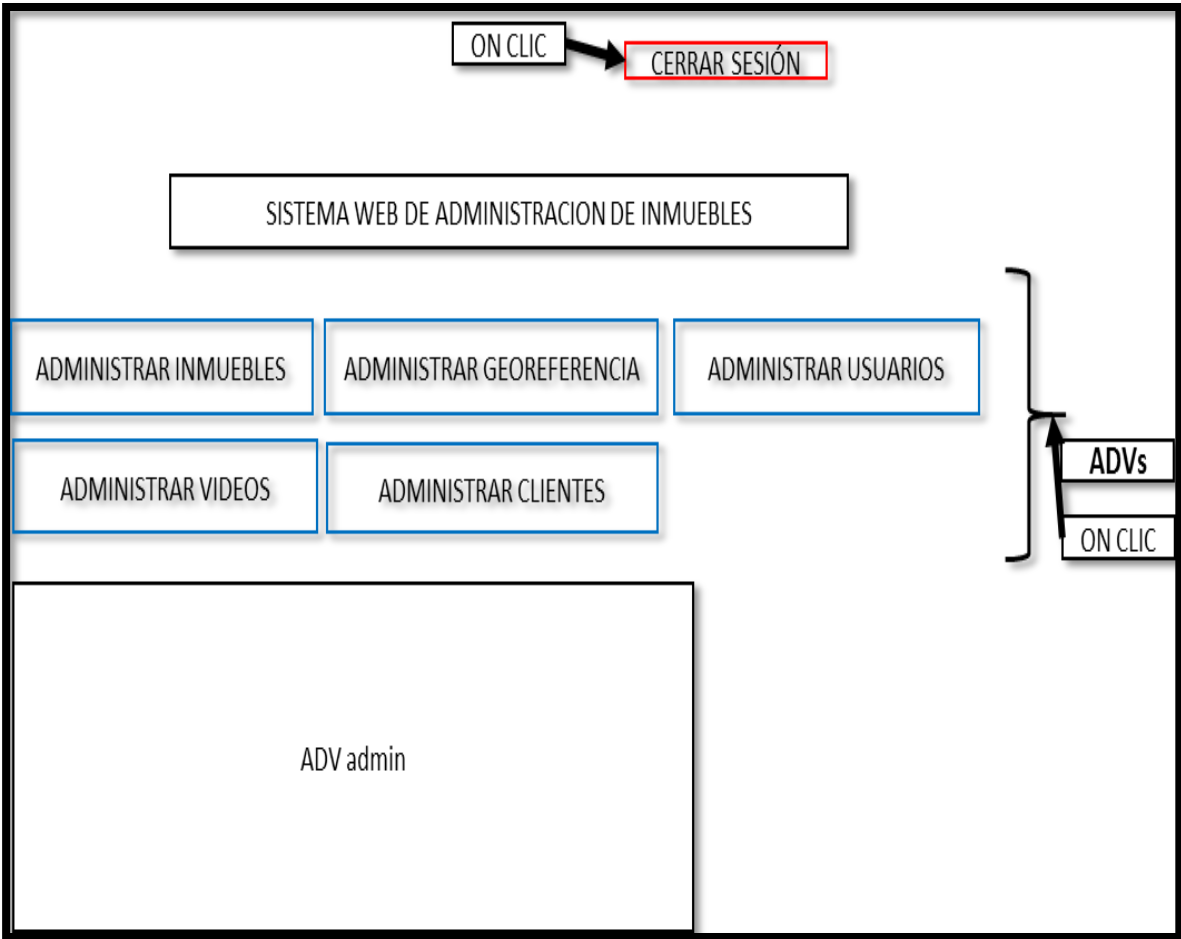


Figura 09-ADV administrador  
Fuente elaboración propia

## Diseño de la interfaz abstracta usuario ocasional.

El ADV de la página principal se muestra en la siguiente figura, este presenta la interfaz de navegación que se muestra al ingresar al sitio web de la inmobiliaria, inicialmente como usuario ocasional (visitante general).

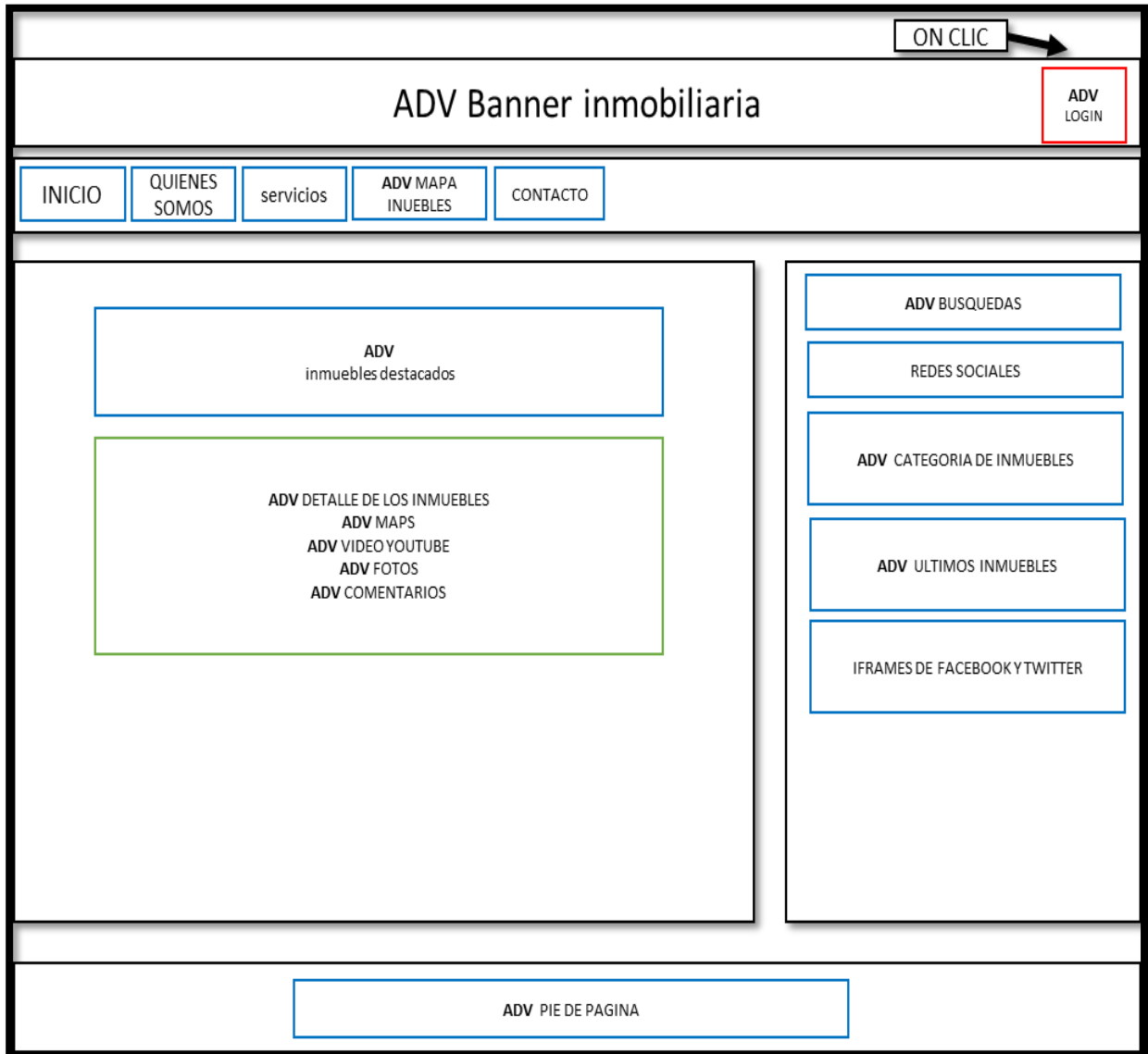


Figura 10-ADV usuario ocasional  
Fuente elaboración propia.

## 4.7 Implementación.

La implementación tendrá una arquitectura de cliente – servidor, donde el lado del cliente se estará ejecutando un navegador HTML el cual constara de una interfaz gráfica y será el encargado de mostrar todas las pantallas con las cuales el usuario tendrá el contacto con el sistema web para su manejo. Cabe mencionar que el browser se encargara de todas las peticiones necesarias al servidor.

Así mismo del lado del servidor se encontrara un servidor web el cual se encargara de atender las peticiones del usuario común y este a su vez re direccionara las peticiones propias del usuario “cliente” a otro servidor para el consumo de las Apis de YouTube y google maps y tendrá el acceso a sus recursos,

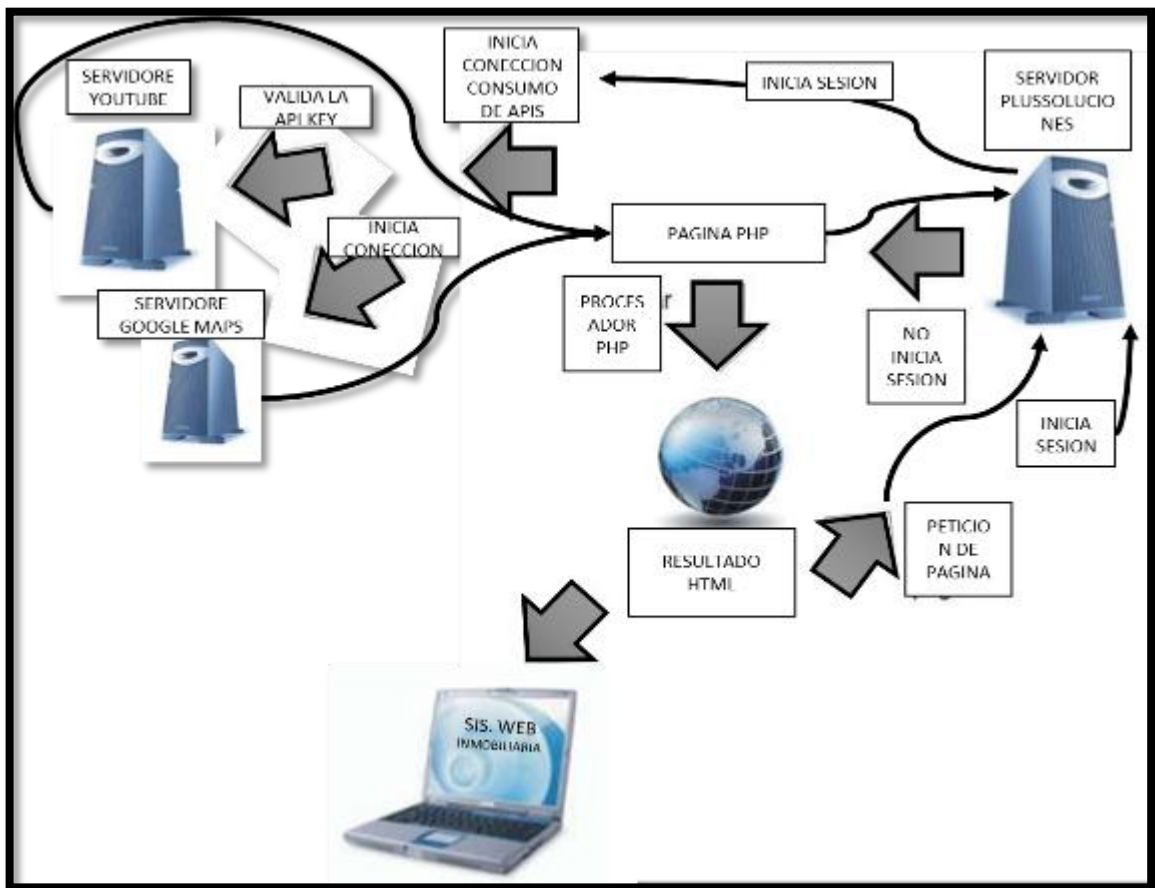


Figura 11-Arquitectura Cliente – Servidor con consumo de Apis  
Fuente elaboración propia.

La interfaz de del usuario para el sistema web de administración y publicación de inmuebles se obtiene a partir del diseño de la interfaz abstracta (ADVs) como se observa a continuación.

La siguiente figura muestra la pantalla de inicio con los menús de opciones:



Figura 12- Interfaz Principal  
Fuente elaboración propia.

En la siguiente interfaz de muestran las ventanas de quienes somos y servicios de la inmobiliaria casa linda.



Figura 13- Interfaz Quienes somos- servicios  
Fuente elaboración propia.

En la siguiente figura se muestra la interfaz del listado de los inmuebles georeferenciados en GoogleMaps e insertos mediante el api de google maps.

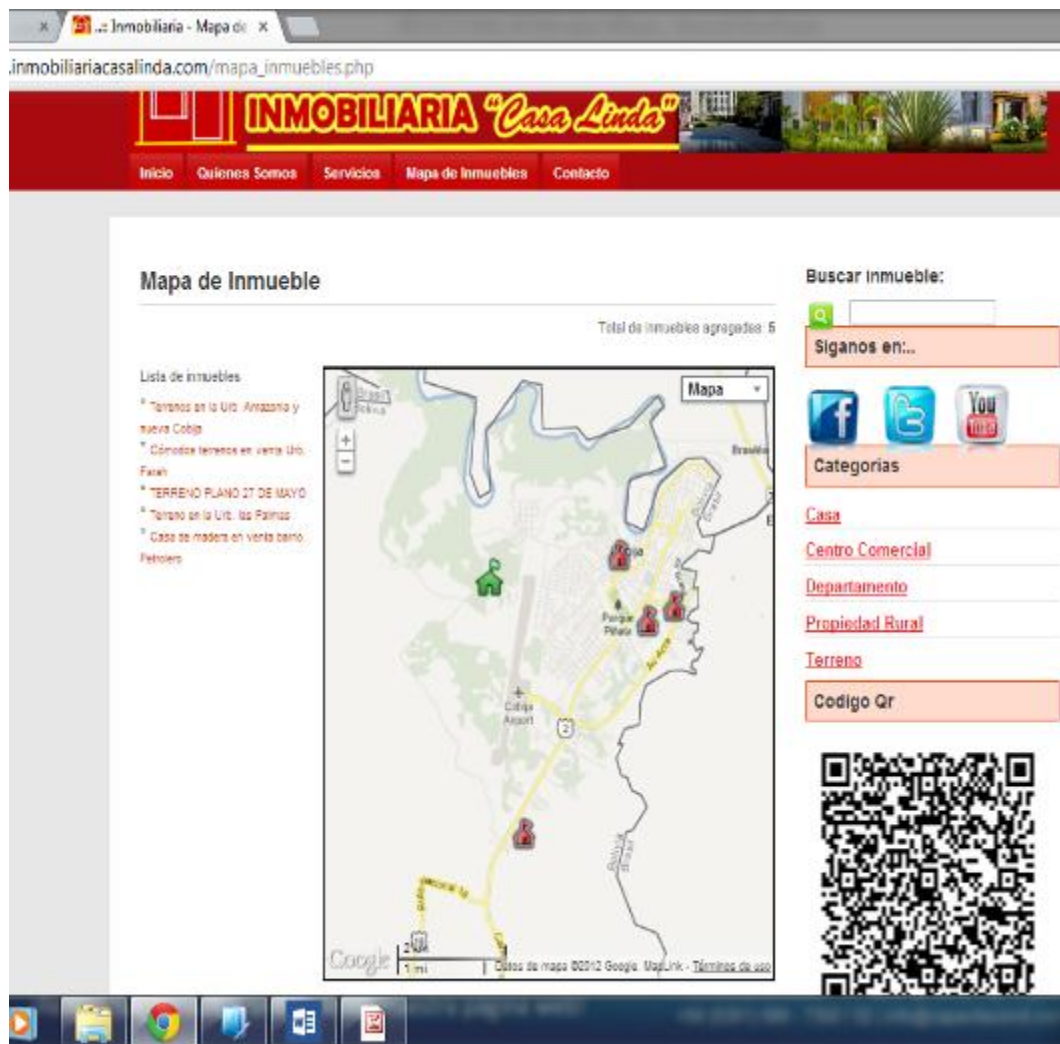


Figura 14- Interfaz Mapa de Inmuebles  
Fuente elaboración propia.

En la siguiente figura se muestra la interfaz del formulario para contactar a la empresa por correo electrónico mediante un sistema interno que redirección el formulario al correo de la empresa.

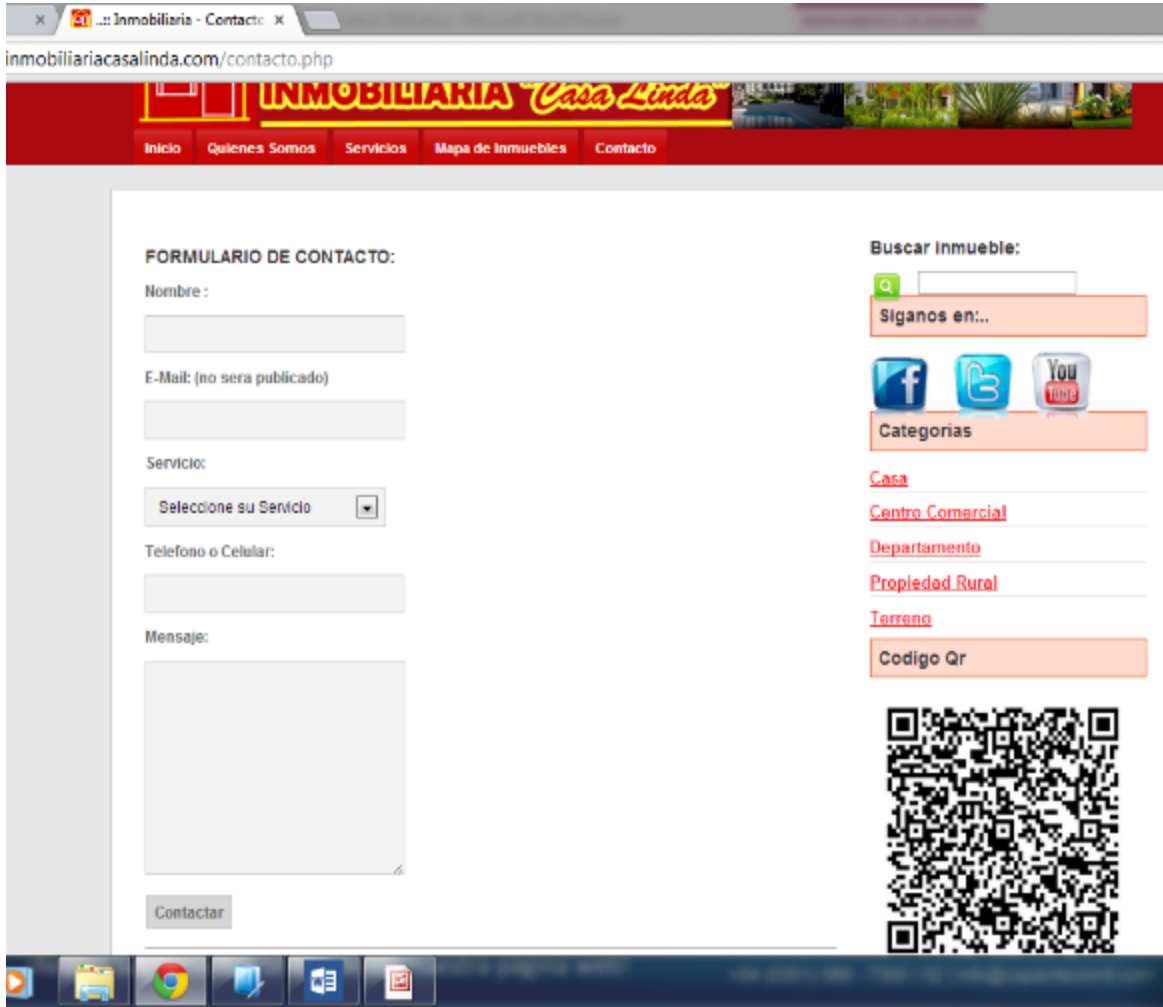


Figura 14- Interfaz del formulario de Contacto  
Fuente elaboración propia.

En la siguiente figura se muestra dinámicamente los inmuebles con una descripción detallada, la Georreferenciación, fotos, videos y un formulario para la el contacto con el inmuebles deseado.



Figura 15- Interfaz de inmuebles dinámicamente  
Fuente elaboración propia.

En la siguiente figura se muestra el sistema de logueo con la encriptación md5 del panel de administración de inmuebles de la inmobiliaria casa linda.

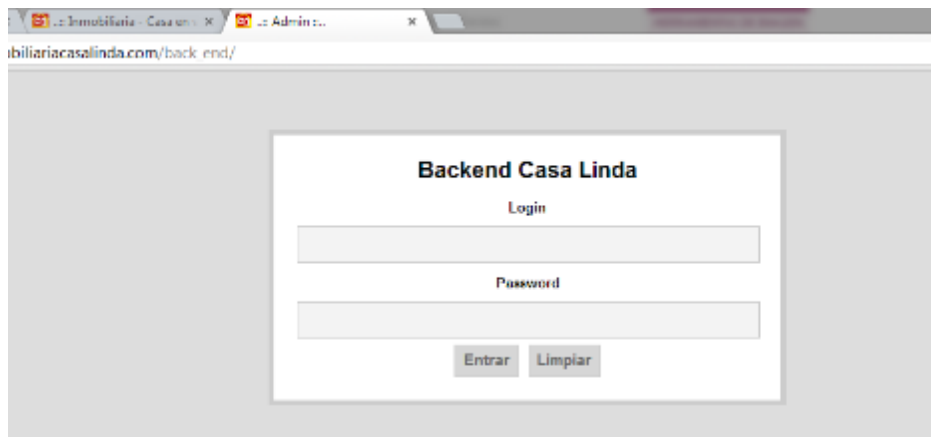


Figura 16- Interfaz del BackEnd de logueo  
Fuente elaboración propia.

En la siguiente figura se muestra el panel principal de administración, del sistema Backend de la inmobiliaria casa linda.



Figura 17- Interfaz del panel principal del Backend  
Fuente elaboración propia.

En la siguiente figura se muestra el panel de administración de inmuebles de los inmuebles se gestionan tanto el inmueble, georeferenciación, videos y lista de inmuebles.



Figura 18- Interfaz del panel de administración de inmuebles.  
Fuente elaboración propia.

En la siguiente figura se muestra el formulario de creación de inmuebles.

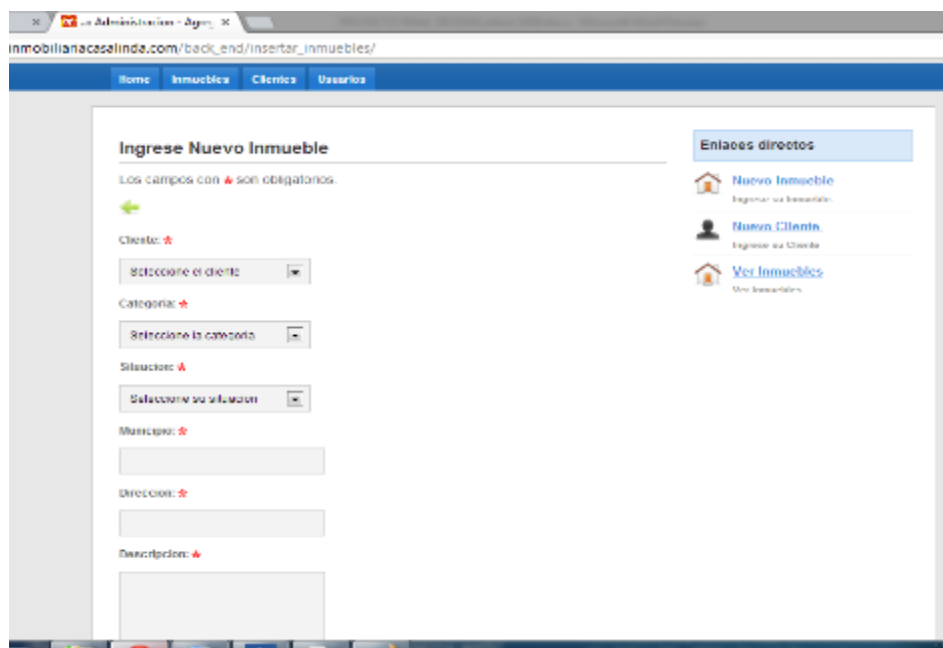


Figura 19- Interfaz de creación de inmuebles.  
Fuente elaboración propia.

En la siguiente figura se muestra la inserción de inmuebles georeferenciados mediante el api de Google Maps.

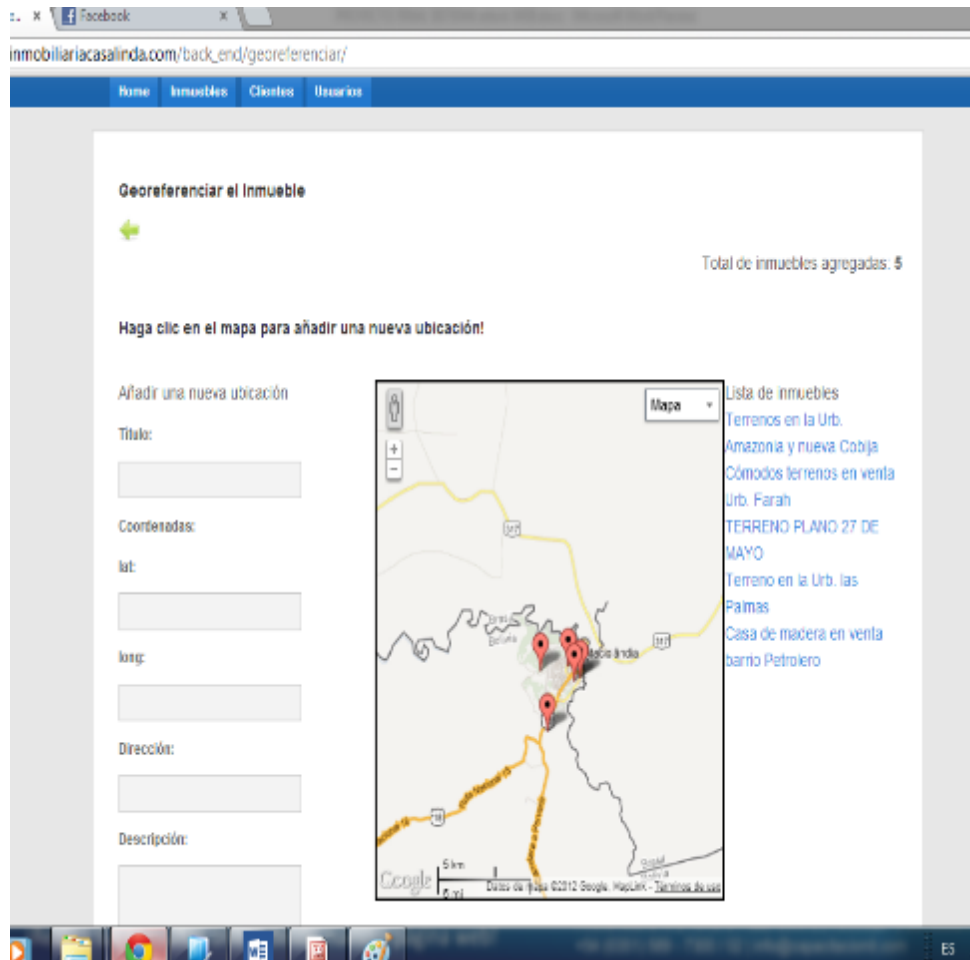


Figura 20- Interfaz del panel de administración de inmuebles georeferenciados.  
Fuente elaboración propia.

En la siguiente figura se muestra la inserción de videos mediante la api de youtube, para de esta manera optimizar el espacio en disco y viralizar la información de la



empresa online.

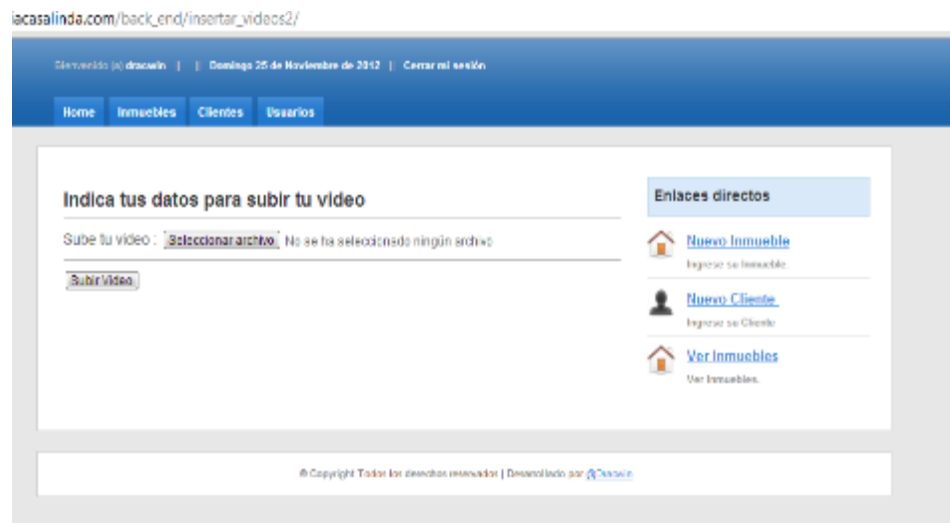


Figura 21- Interfaz del panel de administración de inmuebles intentar videos.  
Fuente elaboración propia.

En la siguiente figura se muestra el listado de inmuebles y su respectiva edición y eliminación si corresponde ya que están asociados a los clientes, que solicitaron el servicio de los inmuebles.

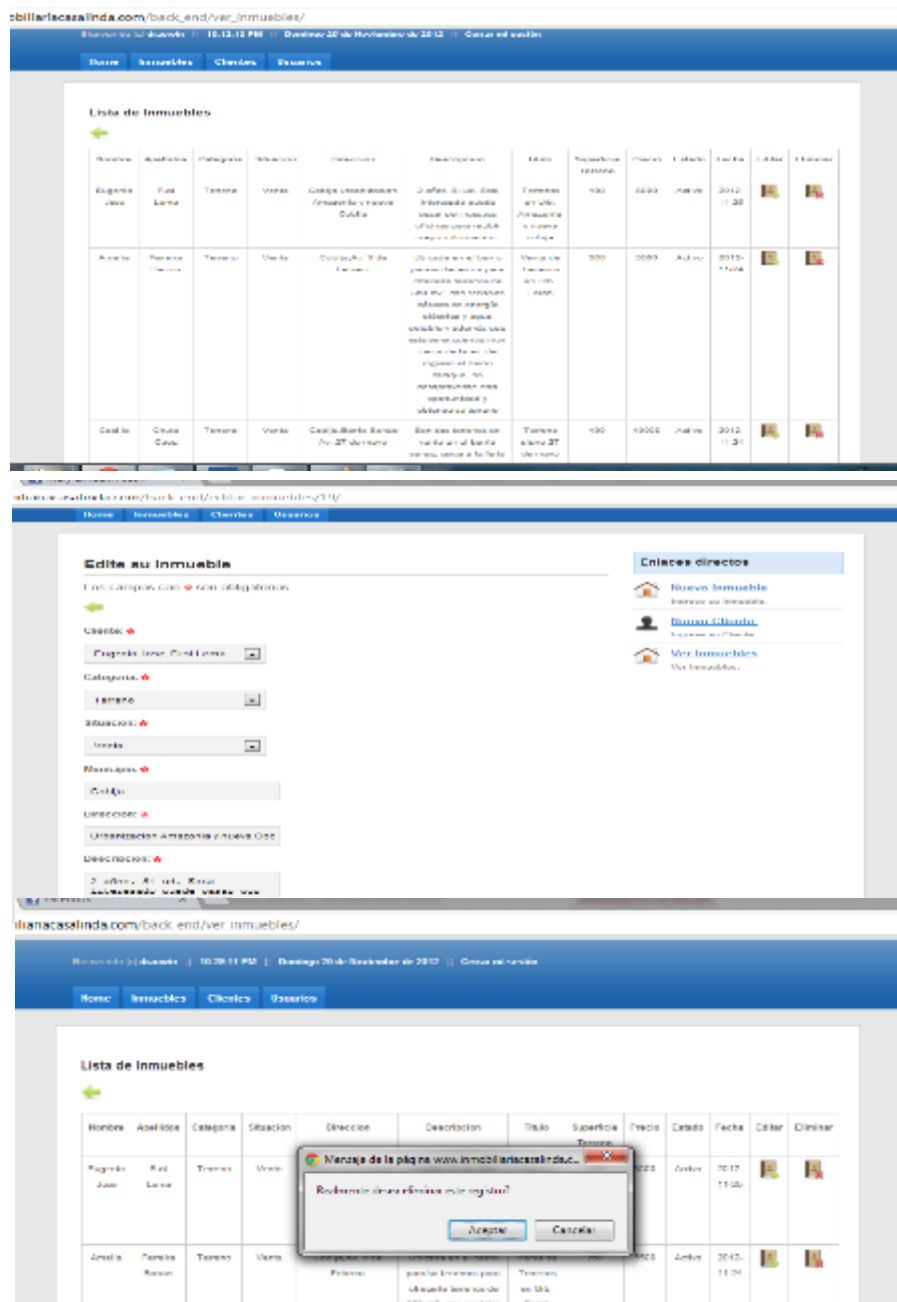


Figura 22- Interfaz del panel de administración de inmuebles edición y eliminación  
Fuente elaboración propia.

La siguiente figura muestra el formulario de administración de clientes para cada inmueble.



Figura 23- Interfaz del panel de administración de inmuebles inserción de clientes  
Fuente elaboración propia.

En la siguiente figura se muestra el panel de administración de clientes asociadas al inmueble.



Figura 24- Interfaz del panel de administración de clientes  
Fuente elaboración propia.

En la siguiente figura se muestra el listado de los clientes y su edición y eliminación de algún cliente, los clientes están asociados a uno a mas inmuebles es por lo cual solo pueden ser eliminado un cliente si no tiene asociados algún inmueble.

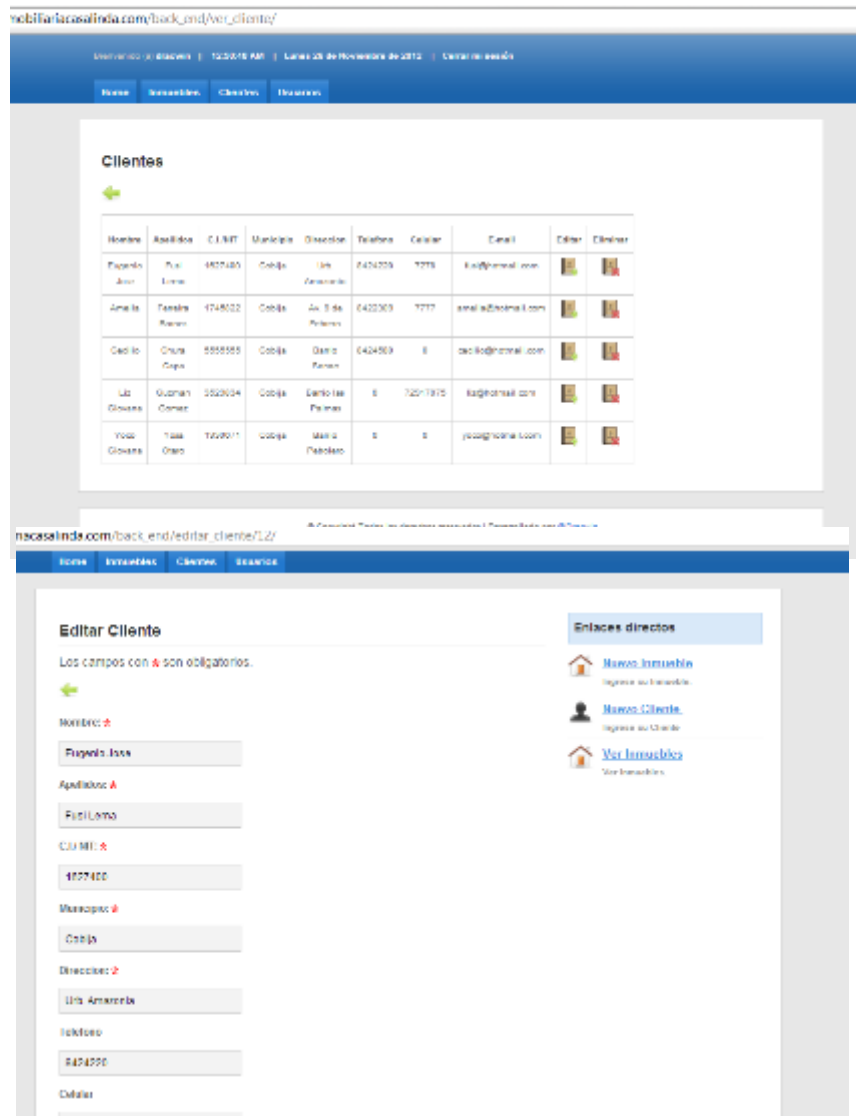


Figura 25- Interfaz del panel de administración de listado, eliminación y edición de los clientes  
Fuente elaboración propia.

En la siguiente figura se muestra el formulario de ingreso de los nuevos clientes a la base de datos.

The screenshot shows a web browser window with the URL `www.inmobiliariascasalinda.com/back_end/insertar_cliente/`. The page has a blue header with navigation links: Home, Inmuebles, Clientes, and Usuarios. The main content area is titled 'Ingreso Nuevo Cliente' and contains a form with the following fields:

- Nombre \*
- Apellido \*
- C.U.M.I. \*
- Municipio \*
- Direccion \*
- Telefono

A note above the form states: 'Los campos con \* son obligatorios.'

Figura 26- Interfaz del panel de administración de ingreso del nuevo clientes  
Fuente elaboración propia.

**CAPITULO V**

**CONCLUSIONES Y**

**RECOMENDACIONES**

## 5.1 Conclusiones.

Se ha logrado alcanzar exitosamente, los objetivos planteados en un principio, a través los diseños e implementación del sistema web de administración y publicación de inmuebles de la inmobiliaria casa linda, haciendo uso de herramientas existentes para su desarrollo. Construir el presente trabajo implicó utilizar una metodología de diseño adecuada, tal como se planteó, por eso concluimos con resultados óptimos en el uso del UML y OOADM que es la extensión lógica más cercana al desarrollo web, concluimos que podemos modelar cualquier proyecto web con UML aunque no se use todas sus herramientas, sin embargo se hace posible efectuar un buen desempeño de estas tecnologías su uso fue de gran utilidad al permitirme especificar de manera precisa los procedimientos y funciones del sistema.

Además se logró alcanzar en su totalidad las siguientes conclusiones:

- Se analizó los requerimientos utilizando diagramas de iteración de casos de uso.
- Se diseñó un esquema conceptual de las entidades que contienen la información y como se relacionan.
- Se diseñó la navegación representada por dos esquemas de clases y contexto.
- Se diseñó la interfaz abstracta que enfoca a los diferentes objetos navegacionales que aparecerán.
- Se implementó un sistema web basado en tecnologías y herramientas actuales así como el uso de SMO, SEO y el consumo de APIs.

El uso innovador que se dio y relacionado con las nuevas tecnologías que hoy en día se usan en para el desarrollo de proyectos web como el consumo de Apis en concreto la Api de YouTube y la de Google Maps, las cuales nos permiten optimizar

el sitio web, e interactuar de manera dinámica cada objeto del sistema de manera fácil y sencilla así como también el social media optimización que refiera a la integración de del like box de Facebook, el twitet de twitter y el canal de youtube personalizado para que interactúe nuestro frond end del sistema.

## **5.2 Recomendaciones.**

Se recomienda:

Se recomienda continuar el proyecto adicionando características que ofrezcan, niveles de roles de usuarios a medida que la empresa crezca, asi también convertir el sistema de administración a un sistema de gestión e ir variando la complejidad del mismo a requerimientos de la empresa.

Se recomienda incorporar una aplicación generada con la API de Facebook para la promoción de sitio y sus contenidos.

Se recomienda incorporar y reestructurar el sitio web para plataformas móviles con el uso de JQuery Mobile.

Se recomienda restringir el acceso al sistema a persona particulares para no tener posteriormente problemas y pérdida de información.

## **5.3 Referencias Bibliográficas**

Alegsa. (18 de Oct de 2005). *alegsa*. Obtenido de Alegsa:  
<http://www.alegsa.com.ar/Dic/sistema.php>

Castells, M. (13 de sep de 2012). *wikipedia*. Obtenido de wikipedia:  
<http://es.wikipedia.org/wiki/Internet>

- Creative. (12 de Mar de 2009). *talento*. Obtenido de smo:  
<http://www.talentocreativocr.com/disenio-web/smo.html>
- Google. (23 de 03 de 2012). *Developers*. Obtenido de developers:  
<https://developers.google.com/maps/documentation/javascript/v2/?hl=es>
- Google. (02 de Mar de 2012). *Google developers*. Obtenido de Google developers:  
[https://developers.google.com/youtube/getting\\_started?hl=es-#data\\_api](https://developers.google.com/youtube/getting_started?hl=es-#data_api)
- Hurtado, J. A. (2005). *Cliente / Servidor*. Popayán,: Departamento de Telemática.
- j, R. (02 de Feb de 2011). *Gnu*. Obtenido de philosophy:  
<http://www.gnu.org/philosophy/free-sw.es.html>
- masadelante. (20 de Nov de 2002). *mas adelante*. Obtenido de masadelante:  
<http://www.masadelante.com/faqs/dominio>
- Mestras, J. P. (2009). *Estructura de las Aplicaciones Orientadas a Objetos*. Madrid:  
Dep Ingenieria y software.
- Nieto-Santisteban, M. A. (2010). *Ingeniería Web. construyendo webs apps*. España:  
Universidad de Extremadura. Departamento de Informática.
- Orallo, E. H. (2007). *El Lenguaje unificado modelado UML*. Cordoba: Cronos.
- phpya. (05 de 08 de 2009). *ver php*. Obtenido de ver php:  
<http://ve2.php.net/manual/es/book.filter.php>
- S.L., G. d. (24 de 06 de 2004). *Pensilvania, EE.UU.*. Obtenido de Pensilvania,  
EE.UU.: <http://www.cpanel.com.es/>
- Salazar, C. (28 de Mar de 2012). *CARlos*. Obtenido de Calos: <http://carlos-salazar-z.blogspot.com.br/2012/03/que-es-mysql.html>
- Silva, D. A. (2001). *Metodologia OOADM*. Madrid: Garzotto.
- Tim. (14 de Nov de 2011). *web*. Obtenido de maestros de la web:  
<http://www.maestrosdelweb.com/editorial/redessociales/>

webmaster. (04 de Jun de 2012). *webmaster*. Obtenido de webmaster:  
<http://support.google.com/webmasters/bin/answer.py?hl=es&answer=35291>


Wikipedia. (11 de Sep de 2011). *Wikipedia*. Obtenido de Wiki:  
[http://es.wikipedia.org/wiki/P%C3%A1gina\\_web](http://es.wikipedia.org/wiki/P%C3%A1gina_web)

wikipedia. (10 de Sep de 2012). *wiki*. Obtenido de wiki:  
[http://es.wikipedia.org/wiki/Extensible\\_Markup\\_Language](http://es.wikipedia.org/wiki/Extensible_Markup_Language)

Wikipedia. (08 de Ago de 2012). *Wiki*. Obtenido de Wiki:  
[http://es.wikipedia.org/wiki/Servidor\\_web](http://es.wikipedia.org/wiki/Servidor_web)

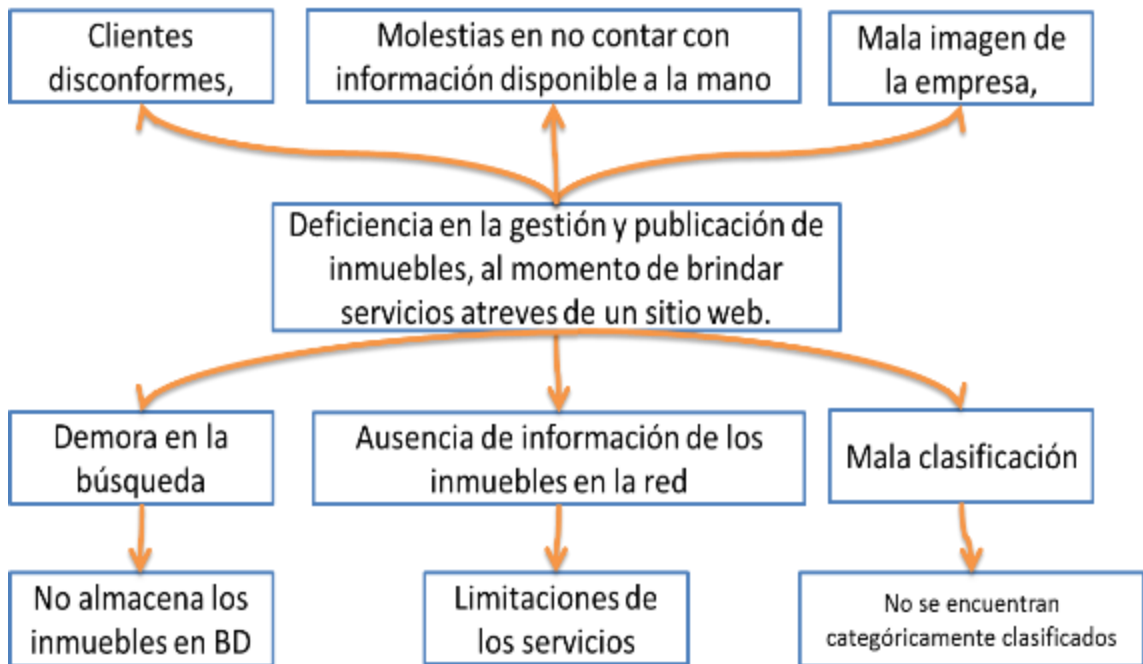
## 5.4 Anexos

### 1.1 CRONOGRAMA.

|  <b>UNIVERSIDAD AMAZÓNICA DE PANDO</b>    |                          | <b>SISTEMA WEB DE GESTIÓN Y PUBLICACIÓN PARA UNA INMOBILIARIA CASO</b> |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
|--|--------------------------|--|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|------------|---------|---------|---------|---------|---------|---------|---------|-----------|--|--|--|---------------|
| <b>Ciudad</b>  | Cobija Capital           |  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
| <b>Carrera</b>   | Ingeniería Informática   |  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
| <b>Modalidad</b>   | Taller de licenciatura I |  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
| <b>Tutor</b>   | Lic. Humberto Fernández  |  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
| <b>Día Mes Año</b><br>Inicia 0 1 0 4 1 2<br>Finaliza 1 1 1 2   |                          |  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
| ETAPAS   | PROG. CUMPL.             | ABRIL  |         |         |         | MAYO    |         |         |         | JUNIO   |         |         |         | JULIO   |         |         |         | AGOSTO  |         |         |         | SEPTIEMBRE |         |         |         | OCTUBRE |         |         |         | NOVIEMBRE |  |  |  | OBSERVACIONES |
|  |                          | semanas  | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas | semanas    | semanas | semanas | semanas | semanas | semanas | semanas | semanas |           |  |  |  |               |
| Recolección y recopilación de datos  | P                        | ■  | ■       | ■       | ■       |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  | En ejecución  |
| Análisis de requerimientos   | P                        |  |         |         |         | ■       | ■       | ■       | ■       |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
| Diseño conceptual.   | P                        |  |         |         |         |         |         |         |         | ■       | ■       | ■       | ■       |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
| Diseño navegacional de los contenidos y panel de   | P                        |  |         |         |         |         |         |         |         |         |         |         |         | ■       | ■       | ■       | ■       |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |
| Diseño de la interfaz abstracta  | P                        |  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         | ■       | ■       | ■       | ■       |            |         |         |         |         |         |         |         |           |  |  |  |               |
| Implementar un sistema web basado en normas y estándares, aplicando una Presentación del informe final y sus correcciones. | P                        |  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         | ■          | ■       | ■       | ■       |         |         |         |         |           |  |  |  |               |
|  | P                        |  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         | ■       | ■       | ■       | ■       |           |  |  |  |               |
| <b>ELABORADO:</b>  |                          | Edwin Mamani Huanca  |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |            |         |         |         |         |         |         |         |           |  |  |  |               |

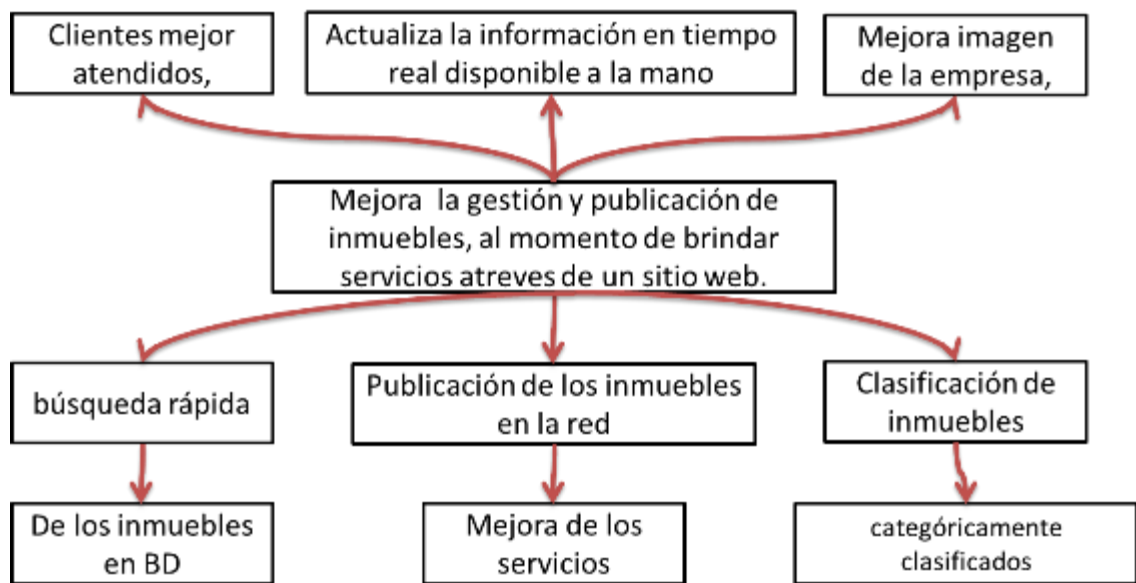
Anexo 01- Cronograma  
Fuente elaboración propia.

### Árbol de problemas.



Anexo 02- Árbol de problemas  
Fuente elaboración propia

### Árbol de objetivos



Anexo 03- Árbol de objetivos  
Fuente elaboración propia