



Trabajo de Diploma para optar por el Título de Ingeniero en Ciencias Informáticas

Título: Subsistema de promoción de anuncios del motor de búsqueda ORIÓN

Autor(es): Jorge Javier Fraga Quevedo
Jorge Luis Almarales Infante

Tutor(es): Ing. Paúl Rodríguez Leyva
Mtr. Aylin Estrada Velazco
Ing. Yasmani J. Alvarez Gómez

La Habana, Junio 2015
“Año 57 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste se firma la presente a los ____ días del mes de _____ del año _____.

Firma del autor
Jorge Javier Fraga Quevedo

Firma del autor
Jorge Luis Almarales Infante

Firma Tutor
Ing. Paúl Rodríguez Leyva

Firma Tutor
Ing. Yasmani J. Alvarez Gómez

DATOS DE CONTACTO

Autores:

Jorge Javier Fraga Quevedo
Universidad de las Ciencias Informáticas,
La Habana, Cuba

E-mail: jjfraga@estudiantes.uci.cu

Jorge Luis Almarales Infante
Universidad de las Ciencias Informáticas,
La Habana, Cuba

E-mail: jlalmarales@estudiantes.uci.cu

Tutores:

Ing. Paúl Rodríguez Leyva
Universidad de las Ciencias Informáticas,
La Habana, Cuba

E-mail: pleyva@uci.cu

Mtr. Aylin Estrada Velazco
Universidad de las Ciencias Informáticas,
La Habana, Cuba

E-mail: avelazco@uci.cu

Ing. Yasmani J. Alvarez Gómez
Universidad de las Ciencias Informáticas,
La Habana, Cuba

E-mail: yjalvarez@uci.cu

FRASE

El éxito no es ni mágico ni misterioso. El éxito es la consecuencia natural de aplicar con firmeza los principios básicos de la superación personal."

Jim Rohn.

Jorge Almarales:

Dedico este trabajo de diploma a mi mamá que siempre confió en mí y me apoyó en todo; a toda mi familia que de una forma u otra siempre me brindaron su apoyo incondicional.

Jorge Fraga:

Dedico este trabajo de diploma a mi mamá, a mi abuela y a mi papá; a toda mi familia que de una forma u otra siempre me brindaron su apoyo incondicional. A mi novia y a mis amigos.

Agradecimientos

Jorge Javier

A mi mamá Zenaida por ser la voz que me impulsa y me apoya constantemente en todas las metas que me propongo. Gracias por enseñarme que aunque el trayecto parezca difícil siempre, a partir del esfuerzo, se puede salir adelante. A ti te lo debo todo y nunca será suficiente.

A mi abuela, mi papá y a toda mi familia por apoyarme de una manera u otra y permitir que yo pudiese estudiar sin importar los problemas.

A mi novia Gretel por ayudarme y apoyarme en los momentos difíciles cuando se me presentaron dificultades en el marco estudiantil y en la vida.

A mis amigos Armando, Carlos, Ángel Luis, Alejandro, Víctor, Lachi, White, Chupi, Ronaldo, Leonardo, Yusniel, Rolando, Mauricio, Juan Carlo, Yaritza, Yeni, Lizandra, Nuri, Yanet, Vladimir, Edy, Raciél, a todos mis colegas de la facultad 4 y a todos los que de una forma u otra brindaron su apoyo en todo momento en mi carrera.

A Jorge Luis mi compañero de tesis por la preocupación, participación, colaboración y el apoyo durante todo este tiempo en que estuvimos realizando este trabajo de diploma.

A mis asesores Aylin, Paúl y Yasmani por su colaboración y dedicación durante el desarrollo de este proyecto, a Julio, Serguey, y a todos los profesores que participaron en mi formación como ingeniero.

A los jurados por su valioso aporte con el cual contribuyen a mejorar este trabajo.

Y para finalizar a la Revolución por permitirme estudiar en esta gran Universidad.

Agradecimientos

Jorge Luis

A mi mamá Yamila por ser la principal causa de mi existencia, apoyarme constantemente en todos los planes que me propongo. Por la educación brindada en el transcurso de mi vida y por enseñarme a ser quien soy.

A toda mi familia por apoyarme y confiar en mí, por preocupación durante los años de estudio y su constante trabajo de darme ánimos para continuar.

A mi primo Yusniel, por compartir su vida universitaria conmigo durante todos estos años, en los estudios, en las diversiones y momentos difíciles.

A Jorge Javier mi compañero de tesis por el esfuerzo, preocupación, colaboración y el apoyo durante la realización este trabajo de diploma.

A mis amigos Harold, Mauricio, Rolando, por compartir conmigo durante estos años, su confianza en todo momento y su apoyo incondicional.

A mis demás amigos que compartieron y se relacionaron conmigo a lo largo de estos años.

A mis tutores Aylin, Paúl y Yasmani por su dedicación y apoyo durante el desarrollo de este proyecto, a Julio, Serguey, Mengana y a todos los profesores que participaron en mi formación como ingeniero.

A los jurados y oponentes por su valioso aporte con el cual contribuyeron a mejorar este trabajo, en especial al profesor Osiris.

Y para finalizar al Comandante Fidel Castro y a la Revolución por permitirme estudiar en esta gran Universidad.

Resumen

Los servicios de promoción de anuncios han evolucionado con el paso del tiempo gracias a los avances tecnológicos del mundo actual. Internet ha dado un gran impulso a la realización de estos servicios con el surgimiento de los motores de búsqueda que han implementado herramientas para la promoción de productos que ofrecen las empresas para el consumo de la sociedad. El buscador Orión desarrollado en la Universidad de las Ciencias Informáticas posee un gran número de servidores y especialistas que los administran generando así muchos costos al país, para solucionar este problema se propone el desarrollo de un subsistema de publicidad y propaganda que permitirá crear un mecanismo de pagos por anuncios al que accederán los usuarios de la web cubana, los clientes de este servicio podrán contratar criterios de búsquedas que serán relacionados con el anuncio que desee publicar el usuario, este subsistema permitirá además la gestión de dichos anuncios así como la administración de los mismos teniendo en cuenta los distintos roles definidos para la interacción con la aplicación.

Tabla de contenido

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTOS TEÓRICOS	6
1.1. Promoción de anuncios en Internet	6
1.1.1. Medios de promoción de anuncios en Internet	8
1.1.2. Promoción de anuncios en buscadores internacionales	9
1.1.3. Promociones de anuncios en Cuba.....	12
1.2. Capa de servicios	13
1.2.1. Arquitectura Orientada a Servicios (SOA).....	13
1.2.2. Servicios web	14
1.3. Lenguajes, tecnologías, metodologías y herramientas.....	17
1.3.1. Metodologías de desarrollo	17
1.3.2. Lenguajes de programación	19
1.3.3. Otros lenguajes.....	21
1.3.4. <i>Framework</i> (marcos).....	21
1.3.5. Sistemas gestores de bases de datos.....	24
1.3.6. Herramienta CASE para el modelado UML	25
1.3.7. Entorno de desarrollo	26
1.3.8. Servidores web	27
CAPÍTULO 2. CARACTERÍSTICAS Y DISEÑO DEL SISTEMA	30
2.1. Propuesta del sistema	30
2.2. Modelo de dominio	31
2.2.1. Descripción de las clases del dominio	31
2.3. Especificación de los requisitos del sistema	32
2.3.1. Requisitos funcionales	32
2.3.2. Requisitos no funcionales.	34
2.4. Casos de uso del sistema.....	35
2.4.1. Definición de los casos de uso del sistema	36
2.4.2. Definición de los actores del sistema.....	36
2.4.3. Diagrama de casos de uso del sistema	37
2.4.4 Descripción de los casos de uso del sistema	38
2.5. Patrones de diseño.....	41
2.5.1. Patrones GRASP	41
2.5.2. Patrones GoF.....	42

Tabla de contenido

2.6. Patrones arquitectónicos	43
2.6.1. Patrón arquitectónico Modelo – Vista – Controlador	43
2.7. Modelo de diseño	44
2.7.1. Diagrama de clases de diseño	44
2.8. Modelo de datos	45
Conclusiones parciales	46
CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBAS	47
3.1. Modelo de implementación.....	47
3.1.1. Diagramas de componentes	47
3.1.2. Diagrama de despliegue	49
3.1.3. Estándares de código.....	50
3.1.4. Pantallas principales de la aplicación.....	52
3.2. Pruebas del sistema	53
3.2.1. Pruebas de caja negra.....	53
3.2.2. Pruebas de rendimiento	58
3.2.3. Pruebas de seguridad.....	59
Conclusiones parciales	60
CONCLUSIONES GENERALES	62
RECOMENDACIONES	63
REFERENCIAS BIBLIOGRÁFICAS.....	64

Índice de imágenes

Imagen 1. Uso de los estándares para servicios web.....	14
Imagen 2. Modelo de dominio.....	31
Imagen 3. Diagramas de casos de uso del sistema.....	38
Imagen 4. Modelo-Vista-Controlador.....	44
Imagen 5. Diagrama de clases de diseño del caso de uso Gestionar anuncio.....	45
Imagen 6. Diagrama de secuencia (Escenario Insertar anuncio del CU Gestionar anuncio).....	¡Error!
Marcador no definido.	
Imagen 7. Modelo de datos.....	46
Imagen 8. Diagrama de componentes.....	48
Imagen 9. Diagrama de componentes de PromocionBundle.....	49
Imagen 10. Diagrama de despliegue.....	50
Imagen 11. Ejemplo de uso de CamelCase.....	51
Imagen 12. Portada del Subsistema de promoción de anuncios de ORIÓN.....	52
Imagen 13. Lista de anuncios del Subsistema de promoción de anuncios de ORIÓN.....	52
Imagen 14. Lista de usuarios del Subsistema de promoción de anuncios de ORIÓN.....	53
Imagen 15. Reporte generado por el JMeter como resultado de las pruebas al sistema.....	59

Índice de tablas

Tabla 1. Fortalezas y debilidades de SOAP y REST.....	16
Tabla 2. Descripción de los actores del sistema.....	36
Tabla 3. Descripción del caso de uso Gestionar anuncio.....	38
Tabla 4. Descripción de las variables correspondientes al caso de prueba para el caso de uso Gestionar anuncio.....	54
Tabla 5. Caso de prueba para el caso de uso Gestionar anuncio: Sección 1: “Insertar anuncio”.....	54
Tabla 6. Caso de prueba para el caso de uso Gestionar anuncio: Sección 2: “Editar anuncio”.....	55
Tabla 7. Caso de prueba para el caso de uso Gestionar anuncio: Sección 3:“Eliminar anuncio”.....	56
Tabla 8. Caso de prueba para el caso de uso Gestionar anuncio: Sección “Buscar anuncio”.....	56
Tabla 9. Caso de prueba para el caso de uso Gestionar anuncio: Sección “Listar anuncio”.....	57
Tabla 10. Resumen de los resultados de las pruebas aplicadas.....	57

INTRODUCCIÓN

En la actualidad existe una gran cantidad de empresas e instituciones que ofrecen una variada gama de productos y servicios. Frente a esta diversidad de ofertas, los clientes toman relevancia a la hora de decidir cuál producto o servicio escoger para satisfacer sus necesidades. Para lograr atraer nuevos clientes hacia los productos y servicios, las empresas utilizan estrategias de *marketing* que le permiten una mejor gestión de sus áreas productivas y comerciales.

Las estrategias de *marketing* sustentan el desarrollo de las empresas en el mundo ya que se llevan a cabo para alcanzar determinados objetivos, tales como dar a conocer un nuevo producto, aumentar las ventas o lograr una mayor participación en el mercado. La *American Marketing Association (AMA)*, Asociación Americana de *Marketing* ha definido el *marketing* como el proceso de planificación, ejecución, fijación de precios, promoción y distribución de ideas, bienes y servicios para crear intercambios que satisfagan los objetivos individuales y organizacionales [1].

Las empresas utilizan un conjunto de herramientas para alcanzar sus objetivos de *marketing* en el mercado que han sido clasificadas como las cuatro *p's* del *marketing*: producto, precio, plaza y promoción. La que impacta directamente en los públicos y hace posible que se conozcan las ideas, bienes y servicios que permiten desarrollar tanto comercial como socialmente una institución, una empresa o un país es la promoción.

La promoción cuenta con varias herramientas: promoción de ventas, anuncios publicitarios y relaciones públicas. Es por ello que es la que más se concentra en hacer llegar a las personas los productos y servicios disponibles de las empresas e instituciones que los realizan para la comercialización.

La promoción es el conjunto de actividades, técnicas y métodos que se utilizan para lograr objetivos específicos, como informar, persuadir o recordar al público objetivo, acerca de los productos y/o servicios que se comercializan [2].

La promoción es la cuarta herramienta del *marketing*, incluye las distintas actividades que desarrollan las empresas para comunicar los méritos de sus productos y persuadir a su público objetivo para que compren. Dentro de la promoción se pueden realizar anuncios en radio, televisión, periódicos, directorios telefónicos e Internet.

Internet ha sido la gran revolución, que ha obligado a adaptar las estrategias de *marketing* a las empresas. Se presenta como un todo en *marketing* ya que ayuda a conocer e investigar el mercado, es un producto o servicio que han de saber manejar las compañías, comienza a ser un magnífico canal de distribución, ayuda a comercializar las empresas y se ha erigido como un magnífico medio de comunicación [3].

Internet es un espacio importante para hacer promociones de anuncios ya que brinda la posibilidad de llegar a muchos usuarios de manera concurrente y en cualquier parte del mundo como forma de comunicación. Los canales de comunicación que permiten que estos anuncios sean conocidos por el público objetivo, favoreciendo la aceptación de una idea o producto, pueden ser las redes sociales, páginas de opinión, *blogs*, directorios, programas de anuncios, *banner* en páginas web, anuncios clasificados, correos *spams*, motores de búsqueda, siendo estos últimos muy utilizados para satisfacer necesidades de información sobre variados temas, como por ejemplo *Google*, *Yahoo*, *Bing*, *Ask*, *MSN*.

Los motores de búsqueda son herramientas de recuperación de información alojada en la web que permiten localizar la información mediante la indexación de los sitios, mostrando a los usuarios los resultados de la indagación de una forma rápida y efectiva. La gran cantidad de información indexada por sitios web como *Google*, *Yahoo* y *Bing* permite un óptimo desarrollo de la promoción como herramienta del *marketing* dentro de los buscadores en un medio digital. Cuando alguien utiliza un motor de búsqueda está creando un contexto determinado en donde la publicidad puede entrar y segmentarse de acuerdo a lo que se esté buscando. Es decir, la publicidad, que es una herramienta de la promoción, se clasifica dependiendo de la búsqueda que se haga en el buscador.

Considerando que la sostenibilidad y desarrollo de este tipo de sistemas depende de grandes inversiones de capital, se diseñan herramientas que aprovechando los intereses de los clientes, brindan servicios de promoción de anuncios mediante los cuales los buscadores pueden obtener ingresos para su sustento y futuro desarrollo. Basándose en la enorme cantidad de usuarios que visitan estos motores de búsqueda a nivel internacional, empresas e instituciones utilizan estos servicios para anunciar sus productos, de manera que estas entidades pagan a los buscadores por la cantidad de usuarios que visitan sus sitios mediante la publicación de los anuncios en estos. Gracias a la expansión de mensajes publicitarios en la web, los ingresos del negocio de Internet de *Google* alcanzaron 15 mil 420 millones de dólares en los primeros tres meses del año 2015 [4].

En la Universidad de las Ciencias Informáticas se desarrolla el motor de búsqueda llamado ORIÓN que permitirá a los usuarios cubanos navegar por la red cubana en la búsqueda de información, lo que posibilitará que dichos usuarios conozcan sitios que pueden ser de su interés. Está propuesto para ser desplegado a nivel de país, el cual es una solución de *software* con una gran necesidad de recursos asociados a su desarrollo y despliegue.

Desde sus inicios en el 2012 este buscador ha consumido una gran cantidad de recursos, mano de obra, gastos en energía, utilización de tecnologías e infraestructura y cuenta con 16 trabajadores. Hasta el 2014 los gastos e inversiones a causa de estas razones son de \$307 022 pesos y este no posee un mecanismo de ingresos económicos que pueda contrarrestar estos gastos e inversiones. Además tiene como proyecciones futuras continuar su desarrollo de tal forma que se irán creando nuevos subsistemas para ORIÓN con el objetivo de mejorar su rendimiento y servicios que pueda brindar. Este es un proyecto presupuestado por el país, pero para un futuro se tiene pensado que pueda sustentarse cumpliendo con las estrategias económicas del país.

Los Lineamientos de la política económica y social del Partido y la Revolución constituyen el documento rector del proceso económico del país. Este define que el sistema económico que prevalecerá continuará basándose en la propiedad socialista de todo el pueblo sobre los medios fundamentales de producción. Plantea que los proyectos y empresas deben contribuir a elevar la eficiencia, deben ser sustentables y permitir desarrollar la economía del país [5].

Como principio fundamental el motor de búsqueda ORIÓN brindará gran parte de sus servicios de manera gratuita a todos los usuarios que lo utilicen. Teniendo en cuenta la necesidad de aplicar un modelo de negocio, y que la experiencia en el área de la comunicación asociada al desarrollo de un sistema de búsqueda que permita promocionar productos y servicios en el país es poco abordada, se identifica el siguiente **problema a resolver**: ¿cómo obtener ingresos con el buscador ORIÓN permitiendo su sustento y contribuyendo a la economía del país?

Lo expuesto anteriormente lleva a identificar como **objeto de estudio**: los servicios de promoción de anuncios en la web enfocándose en el **campo de acción**: los servicios de promoción de anuncios en los motores de búsqueda.

Para darle solución al problema se plantea como **objetivo general** de este trabajo de diploma: desarrollar un subsistema para el motor de búsqueda ORIÓN que le permita brindar servicios de promoción de anuncios relacionados con las necesidades de búsqueda de los usuarios para lograr la obtención de ingresos encaminados a la sostenibilidad económica de este buscador.

Para dar cumplimiento al objetivo general se define la siguiente **idea a defender**: con el desarrollo del Subsistema de promoción de anuncios del motor de búsqueda ORIÓN se contará con un mecanismo que permitirá obtener ingresos para lograr la sostenibilidad del mismo.

El objetivo general queda desglosado en los siguientes **objetivos específicos**:

1. Valorar el marco teórico conceptual y el estado del arte respecto a las tecnologías actuales para el desarrollo de herramientas de promoción de anuncios en motores de búsqueda.
2. Definir la metodología, tecnologías y herramientas para el diseño y la implementación de una aplicación que permita gestionar los servicios de promoción de anuncios en el motor de búsqueda ORIÓN.
3. Implementar una aplicación que permita gestionar los servicios de promoción de anuncios en el motor de búsqueda ORIÓN.
4. Validar el correcto funcionamiento de la aplicación.

Para el cumplimiento de estos objetivos se han trazado las siguientes **tareas de la investigación**:

1. Valoración del estado del arte de los sistemas de promoción de anuncios en motores de búsqueda.
2. Definición de la metodología a utilizar en el desarrollo de la aplicación.
3. Selección de las tecnologías de desarrollo.
4. Identificación de los requisitos funcionales y no funcionales.
5. Diseño e implementación de las funcionalidades identificadas.
6. Aplicación de las pruebas de funcionalidad a la aplicación.
7. Documentación de los artefactos generados durante los procesos de análisis, diseño e implementación de los servicios de promoción de anuncios según la metodología seleccionada.

Durante el proceso de desarrollo del subsistema de promoción de anuncios se realizarán investigaciones para profundizar en el objeto de estudio definido. Dichas investigaciones serán guiadas por métodos teóricos de la investigación.

Analítico–sintético: con el objetivo de analizar la información y la documentación relevante para el desarrollo del *software*, enfatizando en los elementos más importantes que se relacionan con el objeto de estudio.

Modelación: mediante el lenguaje de modelado UML se reflejará la estructura, relaciones internas y características de la solución a través de diagramas.

El presente trabajo consta de 3 capítulos, estructurados de la siguiente forma:

En el capítulo 1 se abordarán los conceptos fundamentales que permitirán entender el comportamiento de la promoción de anuncios en la web, centrándose en los motores de búsqueda. Así como la fundamentación de las herramientas y tecnologías que se emplearán en el desarrollo del trabajo de diploma.

En el capítulo 2 serán abordados los temas relacionados con el dominio y la caracterización del sistema a desarrollar, incluyendo los requerimientos planteados y los casos de uso del sistema, formulándose una propuesta de solución a implementar. También el diseño de la arquitectura del subsistema de promoción de anuncios.

En el capítulo 3 se abordará la implementación y prueba de la solución propuesta y se generarán los artefactos correspondientes a la fase de desarrollo según la metodología seleccionada.

CAPÍTULO 1. FUNDAMENTOS TEÓRICOS

Introducción

En el presente capítulo se brinda una visión general de los aspectos relacionados con los servicios de promoción de anuncios en buscadores web. Se describen los principales conceptos asociados al problema, necesarios para entender el negocio y la propuesta de solución.

Para una buena comprensión de estos servicios se analizan las herramientas y tecnologías en las que se desarrollan, así como realizar una comparación sobre las tecnologías estudiadas. Este estudio indicará cuáles de estas son las recomendables para la implementación de los servicios de promoción de anuncios en el buscador ORIÓN de forma que se logre su rentabilidad, objetivo del presente trabajo de diploma.

1.1. Promoción de anuncios en Internet

La **promoción** es *"la cuarta herramienta del marketing-mix, incluye las distintas actividades que desarrollan las empresas para comunicar los méritos de sus productos y persuadir a su público objetivo para que compren"* [6].

Según la Real Academia Española **promoción** es: *"acción y efecto de promover. Elevación o mejora de las condiciones de vida, de productividad, intelectuales. Conjunto de actividades cuyo objetivo es dar a conocer algo o incrementar sus ventas"*.

Según **Patricio Bonta y Mario Farber**, autores del libro "199 preguntas sobre Marketing y Publicidad", la **promoción** es *"el conjunto de técnicas integradas en el plan anual de marketing para alcanzar objetivos específicos, a través de diferentes estímulos y de acciones limitadas en el tiempo y en el espacio, orientadas a públicos determinados"* [7].

Teniendo en cuenta las anteriores definiciones, los autores resumen la siguiente **definición de promoción**:

La promoción es el conjunto de actividades, técnicas y métodos que se utilizan para informar, persuadir al público objetivo, acerca de los productos y servicios que se comercializan.

O'Guinn, Allen y Semenik, autores del libro "Publicidad", definen a la publicidad de la siguiente manera: "*La publicidad es un esfuerzo pagado, transmitido por medios masivos de información con objeto de persuadir*" [8].

Según la (AMA), la publicidad consiste en: "la colocación de avisos y mensajes persuasivos, en tiempo o espacio, comprado en cualesquiera de los medios de comunicación por empresas lucrativas, organizaciones no lucrativas, agencias del estado y los individuos que intentan informar y/o persuadir a los miembros de un mercado meta en particular o a audiencias acerca de sus productos, servicios, organizaciones o ideas" [9].

La **publicidad** es considerada como una poderosa herramienta de la mercadotecnia, específicamente de la promoción, que es utilizada por empresas, organizaciones y personas individuales, para dar a conocer un determinado mensaje relacionado con sus productos, servicios, ideas u otros, a un grupo de usuarios objetivos.

El **anuncio** según la Real Academia Española es: acción y efecto de dar noticia o aviso de algo; publicar, proclamar, hacer saber. Dar publicidad a algo con fines de propaganda comercial. Soporte visual o auditivo en que se transmite un mensaje publicitario [10].

El **anuncio** según el diccionario libre (*the free dictionary*) es: comunicación mediante la cual se da a conocer una noticia o información a alguien. Mensaje con el que se hace publicidad de un producto o servicio para convencer al público para que lo compre o lo use. Acción de comunicar lo que va a ocurrir en un futuro teniendo en cuenta determinadas señales o indicios [11].

Teniendo en cuenta las anteriores definiciones, los autores resumen la siguiente **definición de anuncio**:

El anuncio es dar a conocer, comunicar al público información sobre productos y servicios para persuadirlos y convencerlos para que compren los mismos.

Internet usa los términos de promoción, publicidad y anuncios como esquemas comunicativos que ponen en contacto de forma simultánea a emisores y receptores de todo el mundo. Dirigir esta forma de comunicación es muy efectivo y relativamente barato, además de aportar ganancias a quien brinda estos servicios. Se usa una combinación de varias estrategias, incluyendo imágenes y reseñas en páginas web relacionadas, hipervínculos, grupos de noticias y anuncios por correo electrónico. Su uso en la red se hace popular y confiable ya que son de fácil edición y acceso. Por otra parte ahorra

tiempo y dinero invertido en trabajos que realizan departamentos que se dedican a la promoción de anuncios en las empresas.

Muchas redes de publicidad exhiben anuncios de texto o gráficos que corresponden a las palabras clave de una búsqueda en Internet o al contenido de la página en la que se muestra el anuncio. Se considera que estos anuncios tienen una mayor probabilidad de atraer a un usuario, ya que tienden a compartir un contexto similar a la consulta de búsqueda del usuario [12].

Los buscadores relacionan estas palabras a los contenidos de las páginas web y devuelven los sitios con mayor relevancia en el tema de búsqueda del usuario.

Otra técnica novedosa es la incrustación de hipervínculos de palabras clave en un artículo que son patrocinados por un anunciante. Cuando un usuario sigue el enlace, son enviados a la página web de un patrocinador. Otras se basan en estrategias de *marketing* viral, social, local, y lo más importante de esta técnica es que se puede medir el retorno de ingresos.

Otra forma es la utilización de las *cookies* que son pequeños archivos de texto que se almacenan en el dispositivo del usuario de Internet al visitar una página web. Son gestionadas por terceros para enlazar a otras páginas web que proporcionan ciertos servicios de interés para el usuario, como puede ser el caso de *Facebook*, *Twitter* o *Google* [13].

Son muchos los internautas que consultan la red en búsqueda de información relevante, lo que permite tener acceso a más clientes. Un gran número de clientes puede ser alcanzado alrededor del mundo con la utilización de Internet. Los motores de búsqueda aprovechándose del gran tráfico de usuarios que enfrentan, utilizan herramientas que brindan servicios de promoción de anuncios, lo que proporciona en consideración a dicho tráfico una serie de ingresos de acuerdo con la estrategia de *marketing* trazada.

1.1.1. Medios de promoción de anuncios en Internet

En Internet existen gran cantidad de aplicaciones dedicadas a brindar servicios de promoción de anuncios.

Algunos de los medios que se pueden utilizar para realizar promoción de anuncios en la red, son los siguientes: [14]

- Programas de anuncios: son programas que permiten colocar un anuncio (imagen o texto) que será mostrado en páginas web relacionadas con el tema del anuncio. De esa manera, se logra atraer un volumen bastante apreciable de visitantes interesados en lo que un sitio web ofrece. Además, solo se paga cuando una persona hace clic en el anuncio; lo cual, reduce drásticamente el costo total e incrementa su rentabilidad. Ejemplos de programas de anuncios son: *Google Adwords* y *Google Adsense* utilizados por el buscador *Google*, *Bing Ads* utilizado por el buscador *Bing*.
- Espacios en páginas web para promoción de anuncios: muchos sitios web ofrecen un espacio en sus diferentes páginas web para que algún anunciante pueda colocar un *banner* o un texto a cambio de una suma de dinero.
- Directorios: son como las guías de teléfonos o mejor aún, como las páginas amarillas en las que se puede encontrar una página web según la categoría y tema en la que se encuentre, por lo que pueden derivar un buen volumen de visitantes interesados en lo que un sitio web ofrece. La principal ventaja de los directorios radica en que sirven de referencia a algunos buscadores que los utilizan para organizar sus propios directorios, un ejemplo de ello es *Google*, que utiliza la base de datos del directorio DMOZ. Por ello, es fundamental que un sitio web sea listado en este directorio.
- Anuncios clasificados: diversos sitios web ofrecen la posibilidad de colocar un anuncio en un sector acorde al rubro del anuncio, el país donde aplica, el tipo de producto. Algunos, brindan este servicio previo pago, otros en cambio lo hacen gratis.

En la investigación que se llevará a cabo se analizarán con mayor profundidad las herramientas y estrategias que utilizan los motores de búsqueda para brindar servicios de promoción teniendo en cuenta que la propuesta del trabajo de diploma va dirigida al buscador ORIÓN.

1.1.2. Promoción de anuncios en buscadores internacionales

Para obtener una guía y características que podría cumplir el subsistema a desarrollar, se realizó un análisis de las herramientas y estrategias que utilizan algunos buscadores existentes en Internet para brindar servicios de promoción de anuncios obteniendo ingresos a través de estos.

Estos buscadores fueron: *Google* que utiliza los programas de anuncios *Google Adwords* y *Google Adsense*, *Yahoo* que utiliza el programa de anuncios *Yahoo Search Marketing* y *Bing* que utiliza el programa de anuncios *Bing Ads*. Este análisis se realizó teniendo en cuenta que la aplicación

propuesta será utilizada por el motor de búsqueda ORIÓN como herramienta para brindar servicios de promoción de anuncios.

Google: ofrece programas publicitarios como *Google Adwords*, *Google Adsense* que ayudan a las empresas a encontrar clientes y ayudan a los editores a obtener dinero con su contenido. *Google* fue el primer buscador en desarrollar un sistema publicitario con estas características. Los factores que han contribuido a su éxito son varios: la publicidad textual funciona mejor que los *banners*; ofrece al usuario justo lo que está buscando; sólo se paga cuando se clica; y de manera especial la publicidad se puede medir con facilidad [15].

Google Adwords: es una herramienta efectiva para hacer promoción de anuncios, programa publicitario donde se insertan anuncios dentro de los resultados de un buscador. Explota al máximo el potencial de los buscadores para mostrar al usuario lo que este está buscando en el momento preciso, su funcionamiento se basa en mostrar solo aquellos anuncios relacionados con los criterios de búsqueda. Al utilizar este producto que brinda *Google* es posible que se envíen algunas *cookies* al navegador que pueden habilitarse desde una serie de dominios diferentes, incluidos *google.com*.

Utiliza *cookies* para hacer que la publicidad sea más atractiva para los usuarios y valiosa para los editores y anunciantes. Estas suelen utilizarse para seleccionar anuncios basados en contenido que es relevante para un usuario, mejorar los informes de rendimiento y evitar mostrar anuncios que el usuario haya visto. También utiliza varias *cookies* de conversión cuya finalidad principal es ayudar a los anunciantes a determinar el número de veces que las personas que hacen clic en sus anuncios terminan comprando sus productos [16].

Este como otros productos que ofrece *Google* pueden recibir o deducir información sobre la ubicación de una variedad de recursos. Puede utilizar la dirección IP para identificar la ubicación general de un usuario; recibir la ubicación precisa desde un dispositivo móvil; deducir la ubicación a partir de consultas de búsqueda del internauta y los sitios web o las aplicaciones que utiliza pueden enviar información sobre el lugar actual donde se encuentra.

Permite saber todo lo que se necesita sobre los anuncios: porcentaje de clics, precio de cada clic, número de impresiones, número de clics, anuncios más clicados, palabras que más usan los usuarios, palabras que consiguen más clics. Se trata por tanto de una publicidad barata, efectiva y medible [17].

Durante 2003 nació el programa *Google AdSense*, que ofrecía a sitios web de terceros de todos los tamaños ingresos adicionales por incluir publicidad en sus propios portales, y que era servida por *Google*. Este programa, que representa la segunda línea de ingresos de la empresa en volumen de facturación, fue un enorme avance en el Internet global. El programa *Google AdSense* difiere en que publica anuncios de *Google AdWords* en sitios web de particulares. De este modo, *Google* paga a los editores web por los anuncios que publican en su sitio en función de los clics que los usuarios efectúan en anuncios o de las impresiones de anuncios, según el tipo de anuncio [18].

Los ingresos mediante *AdSense* dependen de varios factores: [19]

El tipo de anuncios que aparece en sus páginas: al mostrar anuncios con *AdSense* para contenido, puede mostrar anuncios que se pagan cuando los usuarios hacen clic en ellos o cuando los ven. Todos los anuncios compiten en una subasta de anuncios, y el sistema elige automáticamente a los anunciantes que están dispuestos a pagar más y que aseguren una buena experiencia de usuario.

El precio de los anuncios que aparecen en sus páginas: no todos los anuncios tienen el mismo precio, por lo que sus ganancias pueden variar de un día para otro en función de los presupuestos de los anunciantes, de su contenido específico y de los anuncios en los que los usuarios hacen clic.

La cantidad de productos de *AdSense* que utiliza: mostrando anuncios en páginas web, proporcionando búsquedas a los usuarios. Cuantos más productos de *AdSense* utilice, más aumentará la posibilidad de ingresos.

Yahoo: tiene herramientas y métodos que le permiten comunicarse con los usuarios que la visitan a través de publicaciones de anuncios. Sugiere anuncios en función de diversos factores, como el contenido de la página donde aparecen, la información que el usuario proporciona para adaptar los anuncios cuando se use uno de los servicios que este brinda. Los usuarios pueden controlar el uso que hace *Yahoo* de esta información para la selección de anuncios y para mejorar la eficacia de los intercambios mediante la exclusión voluntaria.

Usa la información de la ubicación del usuario para relacionar los anuncios publicitarios con la región en la que se encuentra realizando una búsqueda. En el proceso de publicación de anuncios, utiliza herramientas llamadas *cookies*. Una *cookie* es un fragmento de información que se almacena en una computadora a fin de identificar el navegador durante la interacción en los sitios web. Se pueden utilizar para almacenar elementos como identificadores y preferencias de los usuarios [20].

Estos buscadores mencionados y otros más como: *Ask*, *Msn* y *Bing* por citar ejemplos; debido al alto nivel de rentabilidad que ofrecen los servicios de promoción de anuncios usan diferentes modelos de ingreso que permiten calcular el costo monetario en correspondencia con la cantidad de visitas que recibe un anuncio o la cantidad de clic que recibe un *link* con el fin de sustentar su desarrollo económico futuro.

Bing: es el único motor de búsqueda que realiza búsquedas en todos los dispositivos, creando una experiencia centrada en el usuario que proporciona la información que las personas necesitan, donde y cuando la necesitan. El objetivo es aprovechar la asociación con *Facebook*, *Twitter* y *Yahoo* y crear nuevas experiencias para buscadores como *Bing Smart Search* en *Windows 8.1*. La sofisticación de esta plataforma, unida a la solidez de las asociaciones y al alcance y diversidad del público de *Yahoo Bing Network*, permite ofrecer una experiencia de búsqueda innovadora tanto a clientes como a anunciantes.

Los anunciantes disfrutan de un mayor control: *Bing Ads* permite segmentar a los usuarios por dispositivo móvil y conectar con los clientes gracias a las ofertas de anuncios de llamada, como las extensiones de llamada, las extensiones de ubicación y *Bing Offers*. Cuando los clientes ven y hacen clic en anuncios, se les remite al sitio web del anunciante. Independientemente de que efectúen una compra, soliciten un servicio o realicen una suscripción para obtener más información, estas acciones de los clientes se denominan conversiones. Una conversión consiste en transformar un cliente potencial en un cliente real. Una vez creados los anuncios y segmentado el público adecuado, el objetivo es conseguir que los anuncios aparezcan en sitios web, por lo que se deberá centrar en las palabras clave y las pujas de palabras clave. Al crear un anuncio, se elige una palabra clave relevante para el producto o servicio que coincidan con las necesidades de los clientes. Para los clientes es importante crear una lista eficaz de palabras clave. Puede que tengan una gran estrategia de segmentación, pero sin una lista eficaz de palabras clave, es posible que los clientes nunca vean los anuncios [21].

Con *Bing Ads* los anunciantes ahorrarán tiempo y obtendrán resultados optimizados con recursos de segmentación por estado, ciudad, edad y horario, así como activar campañas en dispositivos móviles. También se podrá generar informes y sugerencias de optimización y expansión de palabras clave para las empresas e importar las campañas desde la competencia de manera fácil [22].

1.1.3. Promociones de anuncios en Cuba

Cuba ha aprovechado los avances tecnológicos de los últimos tiempos para desarrollar sistemas capaces de gestionar actividades de comunicación entre usuarios de la Intranet y el Internet. Existen disímiles empresas u organizaciones que promueven sus servicios o productos tanto de exportación como de importación mediante su propio sitio web, como ejemplo de esto se encuentran: etecsa.cu y cubacelonline.cu. Otras páginas como cubadebate.cu y granma.cu ofrecen noticias, opiniones, reflexiones del ámbito nacional a todo el interesado en visitarla. Pero no existe un sistema rentable capaz de brindar servicios de promoción de anuncios para que las organizaciones puedan publicar anuncios. Además, que los usuarios sean capaces de acceder a esta información en aras de satisfacer necesidades ya sea de información, productos o servicios disponibles en la red. También, que pueda aportar ganancias a la economía del país.

1.2. Capa de servicios

Para la comunicación entre el buscador ORIÓN y el subsistema a desarrollar se hace necesaria una capa de servicios que permita al primero realizar peticiones al segundo. Esta capa se encargará de verificar los datos recibidos y devolver la información relacionada a estos. Además, se desea que para un futuro otras aplicaciones web puedan utilizar el subsistema para realizar promociones de anuncios.

1.2.1. Arquitectura Orientada a Servicios (SOA)

En la actualidad, el desarrollo de aplicaciones web que brindan servicios es muy común y utilizado. Pero estos servicios deben ser orientados por una arquitectura que brinde formas de encapsulamiento, abstracción e interfaces bien definidas de manera que de la red pueda accederse a sus funcionalidades.

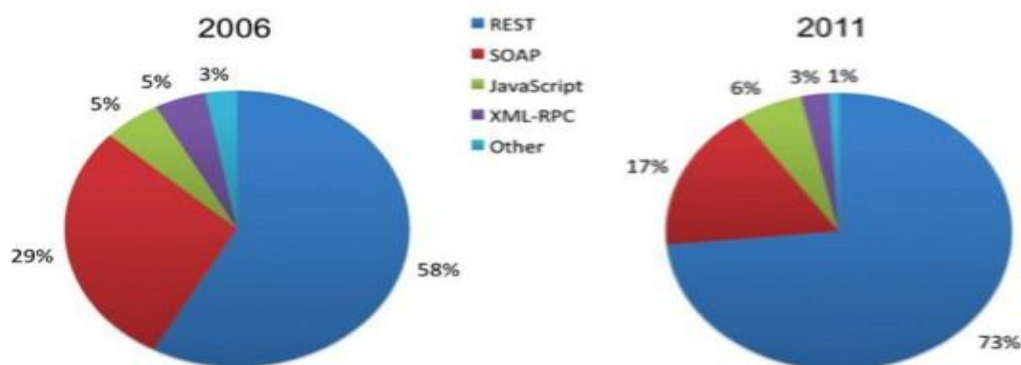
SOA es una arquitectura que trata de estructurar las aplicaciones de negocio y la tecnología para responder de forma ágil y flexible a las demandas del mercado. No se trata de algo radicalmente nuevo, sino que se debería ver como la última fase del proceso de evolución de la arquitectura tecnológica y de negocio de toda la empresa.

El objetivo primordial de SOA es: promover la reutilización de funcionalidades existentes en compañías, denominados servicios, los cuales se encuentran en diferentes lugares y además, están contruidos en diferentes lenguajes y plataformas. Esto con el único fin de agilizar la definición, construcción y divulgación de procesos de negocio que respondan al constante cambio de los requerimientos del mercado [23].

1.2.2. Servicios web

Existen múltiples definiciones sobre lo que son los servicios web. Una posible sería hablar de ellos como un conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la web [24].

Para la creación de estos servicios se utilizan varios estándares como: REST, SOAP, *JavaScript*, XML-RPC y otros. Como se puede observar en la siguiente imagen desde el año 2006 hasta el año 2011 incrementó la utilización de los servicios REST en comparación con los demás. También, se redujo mayormente el uso de SOAP.



Distribution of API protocols and styles

Based on directory of 3,200 web APIs listed at ProgrammableWeb, May 2011

Imagen 1. Uso de los estándares para servicios web.

Servicios SOAP

Simple Object Access Protocol (SOAP) es un estándar del *World Wide Web Consortium* (W3C) que define cómo objetos remotos pueden comunicarse mediante el intercambio de XML. La idea básica es que en la comunicación hay dos partes (cliente y servidor), una de las cuales (el servidor) presta una serie de servicios que son consumidos por la otra (cliente). Lo importante aquí es entender que los servicios web SOAP están orientados a funcionalidad. El servidor implementa una serie de funcionalidades y le dice al mundo cómo pueden invocarse [25].

Servicios REST

Representational State Transfer (REST) que en español significa transferencia de estado representacional. Es un conjunto de técnicas orientadas a crear servicios web en los que se renuncia a la posibilidad de especificar la interfaz de los servicios de forma abstracta, a cambio de contar con una convención que permite manejar la información mediante una serie de operaciones estándar. La convención utilizada no es otra que el protocolo HTTP. La idea detrás de REST es el desarrollo de servicios orientados a la manipulación de recursos. En un servicio REST típico, se tiene una URL por cada recurso (documento, entidad) que se gestiona, y que realiza una tarea diferente sobre dicho recurso en función del método HTTP que se utilice [26].

HTTP

El protocolo *Hypertext Transfer Protocol* (HTTP) es el protocolo base de la World Wide Web (WWW). Se trata de un protocolo simple, orientado a conexión y sin estado [27].

HTTP especifica un conjunto de métodos y cabeceras que permiten a los clientes y servidores interactuar e intercambiar información de manera uniforme y fiable. HTTP utiliza identificadores de recursos uniformes (URI) para identificar los datos en Internet. Las URIs que especifican ubicaciones de los documentos se llaman localizadores uniformes de recursos (URLs). URLs comunes se refieren a archivos, directorios u objetos que realizan tareas complejas, como las búsquedas de bases de datos y búsquedas en Internet. Si conoce la dirección URL HTTP de un documento XHTML a disposición del público en cualquier parte de la web, se puede acceder a él a través de HTTP [28].

XML

eXtensible Markup Language (XML), lenguaje extensible de marcas. Se trata de un estándar del *World Wide Web Consortium*. Es un lenguaje de marcas capaz de describir cualquier tipo de información en forma personalizada, es un sistema complejo de descripción de información libre y rigurosa [28].

JSON

JavaScript Object Notation (JSON) es un formato de intercambio de datos ligero. Es fácil para las máquinas para analizar y generar. Es un formato de texto que es completamente independiente del lenguaje, pero utiliza las convenciones que son familiares para los programadores de la familia de lenguajes C, incluyendo C, C ++, C #, *Java*, *JavaScript*, *Perl*, *Python* y otros. Estas propiedades hacen a JSON un lenguaje ideal para el intercambio de datos [29].

REST vs SOAP

Estos son los dos enfoques principales para la interfaz de la web con los servicios web. Ambos enfoques son diferentes y presentan algunas ventajas y desventajas para la conexión a los servicios web: SOAP es conceptualmente más difícil y más pesada que REST. Aunque ambas técnicas son adecuadas para el desarrollo de sistemas paralelos, pero SOAP es más pesado que REST, debido principalmente a la verbosidad de comunicaciones de SOAP: XML aumenta el tiempo necesario para analizar los mensajes.

SOAP es un protocolo para la computación distribuida, mientras que REST adhiere mucho más de cerca a un diseño basado en la web. SOAP requiere una mayor aplicación, comprensión y esfuerzo del lado del cliente a diferencia de las API basadas en REST, que centran estos esfuerzos en el lado del servidor. Es importante tener en cuenta que una de las ventajas de SOAP es el uso del transporte "genérico". Mientras REST hoy utiliza HTTP / HTTPS, SOAP puede utilizar casi cualquier transporte para enviar la solicitud [30].

En la siguiente tabla se muestra una comparación entre las fortalezas y debilidades de SOAP y REST.

Tabla 1. Fortalezas y debilidades de SOAP y REST.

SOAP	
Fortalezas	Debilidades
<ul style="list-style-type: none"> + Manejar entornos informáticos distribuidos. + Incorporado en el manejo de errores. + Extensibilidad. + Idioma, plataforma, y transporte agnóstico. + Prevalece estándar para los servicios web. + El apoyo de otras normas (WSDL, WS- *). 	<ul style="list-style-type: none"> - Más detallado. - Más difícil de desarrollar, requiere herramientas. - Más pesado y difícil que REST.
REST	
Fortalezas	Debilidades
<ul style="list-style-type: none"> + Lengua y plataforma agnóstica. + Mucho más sencillo de desarrollar que SOAP. + Curva de aprendizaje pequeña, menor dependencia de herramientas. + Conciso, sin necesidad de capa de mensajería adicional. + Más cerca en el diseño y la filosofía de la web. 	<ul style="list-style-type: none"> - Supone un modelo de comunicación punto a punto. - No es utilizable para entorno de computación distribuida. - La falta de normas de apoyo a la seguridad. - Atado al modelo de transporte HTTP.

Para la implementación de la capa de servicios se escoge utilizar el estilo arquitectónico REST dado que es uno de los más utilizados en el mundo actual y la facilidad de desarrollo que posee. Además se tuvo en cuenta que la capa de servicios de ORIÓN está implementada con este servicio. También se tuvo en cuenta que se puede utilizar HTTPS como protocolo de transporte ya que REST trabaja sobre este y ORIÓN también lo utiliza. Para la interoperabilidad se utilizará JSON por ser el lenguaje de intercambio utilizado por REST.

1.3. Lenguajes, tecnologías, metodologías y herramientas

1.3.1. Metodologías de desarrollo

Cada metodología de desarrollo de software tiene su propio enfoque para el desarrollo de aplicaciones. Además, tienen como objetivo presentar un conjunto de técnicas tradicionales y modernas de modelado, que permitan desarrollar aplicaciones de calidad, incluyendo heurísticas de construcción y criterios de comparación de modelos de sistemas.

OpenUP

OpenUP es una metodología de Proceso Unificado que aplica enfoques iterativos e incrementales dentro de un ciclo de vida estructurado, utiliza una filosofía ágil que se enfoca en la naturaleza de colaboración para el desarrollo de *software*, basada en *Rational Unified Process (RUP)*, que contiene el conjunto mínimo de prácticas que ayudan a un equipo de desarrollo de software a realizar un producto de alta calidad, de una forma eficiente [31].

Es una metodología dirigida a la gestión y desarrollo de proyectos de *software* basados en desarrollo iterativo, ágil e incremental, apropiada para proyectos pequeños y de bajos recursos. Es aplicable a un conjunto amplio de plataformas y aplicaciones de desarrollo.

Esta metodología posee varias iteraciones dentro del ciclo de vida del proyecto, que no superan las pocas semanas de duración, en dependencia de los acuerdos que se toman en el equipo de trabajo. Se debe tener en cuenta que cada iteración concluye obligatoriamente con una muestra concreta del producto, que necesariamente tiene que ser “demostrativa” o “explotable”, ya que es la forma que tiene la metodología de desarrollo de demostrarle el valor agregado al cliente [32].

El objetivo de *OpenUP* es ayudar al equipo de desarrollo, a lo largo de todo el ciclo de vida de las iteraciones, para que sea capaz de añadir valor de negocio a los clientes, de una forma predecible, con la entrega de un *software* operativo y funcional al final de cada iteración.

Ventaja: es una metodología ágil, se puede adaptar con otros procesos, permite disminuir las posibilidades de riesgo, permite descubrir errores tempranos a través de ciclos iterativos.

Desventaja: a veces omite contenido que puede ser de interés en el proyecto, se espera que cubra un amplio sistema de necesidades en los proyectos en un plazo muy corto, no es adecuada para proyectos de gran tamaño [33].

XP

Extreme Programming (XP) es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de *software*, promoviendo el trabajo en equipo, preocupándose por el trabajo de los programadores y propiciando un buen clima de trabajo. Se basa en realimentación continua entre cliente y equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Propone una serie de roles como: desarrollador, cliente, encargado de prueba, encargado de seguimiento, entrenador, consultor, gestor; los cuales dan paso a una organización muy definida del trabajo partiendo de este trabajo en equipo que propone. Estos roles cumplen tareas de forma individual, pero relacionadas entre sí, de manera que al obtener los resultados, estos conlleven como consecuencia un *software* de buena calidad.

La principal suposición que se realiza en XP es la posibilidad de disminuir la ficticia curva exponencial del costo del cambio a lo largo del cambio de proyecto, lo suficiente para que el diseño evolutivo funcione [34].

Scrum

Scrum es una metodología simple, que requiere trabajo duro, porque no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto.

Como método ágil:

- Es un modo de desarrollo adaptable, antes que predictivo.
- Orientado a las personas, más que a los procesos.
- Emplea el modelo de construcción incremental basado en iteraciones y revisiones.

Comparte los principios estructurales del desarrollo ágil: a partir del concepto o visión de la necesidad del cliente, construye el producto de forma incremental a través de iteraciones breves que comprenden fases de especulación, exploración y revisión. Estas iteraciones se repiten de forma continua hasta que el cliente da por cerrado el producto [35].

Selección de la metodología de desarrollo

Con el objetivo de lograr una guía para el proceso de desarrollo del *software* en cuestión se escoge *OpenUp*. Se tuvo en cuenta que es una metodología ágil, adecuada para proyectos pequeños. Se puede adaptar con otros procesos, lo cual garantiza su uso en cambios futuros o nuevas versiones del subsistema. Además se tuvo en cuenta que esta metodología forma parte de la base tecnológica del proyecto principal ORIÓN.

1.3.2. Lenguajes de programación

Los lenguajes de programación constituyen lenguajes formales diseñados para que los seres humanos puedan dar instrucciones a un equipo. Estos se conforman por un conjunto de reglas sintácticas y semánticas que mediante estructuras y significados lógicos de expresiones forman un programa, capaz de dar soluciones a problemas cotidianos de la vida humana.

PHP

Hypertext Preprocessor (PHP) es un lenguaje sencillo, de sintaxis cómoda y similar a la de otros lenguajes como C o C++, es rápido a pesar de ser interpretado, multiplataforma y dispone de una gran cantidad de librerías que facilitan muchísimo el desarrollo de las aplicaciones web. El código generado es estable, se lee muy bien y se puede programar utilizando objetos. Es un lenguaje ideal tanto para el que comienza a desarrollar aplicaciones web como para el desarrollador experimentado. Está basado en herramientas con licencia de *software* libre, es decir, no hay que pagar licencias, ni se está limitado en su distribución. PHP permite la compresión de cadenas largas por medio de los métodos *gzcompress()* y *gzuncompress()* [36].

Está enfocado a la programación de *scripts* del lado del servidor, por lo que se puede usar funcionalidades, como recopilar datos de formularios, generar páginas con contenidos dinámicos, o enviar y recibir *cookies*. Con este lenguaje no se está limitado a generar HTML, ya que lo trae introducido en sí. Una de las características más potentes y destacables de PHP es su soporte para un amplio abanico de bases de datos como *MySQL*, *PostgreSQL* y *MongoDB* por citar ejemplos [37].

Python

Python es un lenguaje de programación creado por Guido Van Rossum a principios de los años 90 cuyo nombre está inspirado en el grupo de cómicos ingleses “*Monty Python*”. Se trata de un lenguaje interpretado o de *script*, con tipado dinámico, multiplataforma y orientado a objetos.

La sintaxis de *Python* es tan sencilla y cercana al lenguaje natural que para todos los programas elaborados en este lenguaje parecen pseudocódigo. Por este motivo se trata además de uno de los mejores lenguajes para comenzar a programar. *Python* no es adecuado sin embargo para la programación de bajo nivel o para aplicaciones en las que el rendimiento sea crítico.

Es una herramienta para la codificación de *scripts* orientados por el sistema operativo. Este programa a menudo se lanza desde las líneas de comandos de la consola, y realizan tareas tales como el procesamiento de archivos de texto y el lanzamiento de otros programas [38].

Java

El impulso inicial de *Java* no fue Internet, sino la necesidad de un lenguaje de programación que fuera independiente de la plataforma, y que pudiera ser utilizado para crear *software* para diversos dispositivos electrónicos, como hornos, microondas, mandos a distancias, sistemas operativos para dispositivos móviles.

Si el mundo de la web no se hubiese desarrollado, al mismo tiempo que *Java* estaba siendo implementado, podría haber sido un lenguaje de utilidad para la programación de dispositivos electrónicos de consumo. Con la aparición de la *Word Wide Web*, se ha visto impulsado a lugares muy considerables en el diseño de los lenguajes de programación, porque también la web demanda programas portables.

Java se puede utilizar para crear dos tipos de programas; aplicaciones, que son programas que se ejecutan en el ordenador del usuario y *applets* (subprogramas), que son aplicaciones diseñadas para ser transmitidas por Internet y ejecutadas por un navegador compatible con este lenguaje [39].

Selección del lenguaje de programación

Se escoge PHP en su versión 5.3 por la flexibilidad que presenta este lenguaje, por la cantidad de librerías que facilitan el desarrollo de las aplicaciones web. Está basado en herramientas de

desarrollo con licencia en *software* libre aunque es multiplataforma. Compatibles con disímiles bases de datos.

1.3.3. Otros lenguajes

UML

El *Unified Modeling Language* (UML), Lenguaje Unificado de Modelado en español, es un lenguaje para especificar, visualizar, construir y documentar los artefactos de los sistemas *software*, así como para el modelado del negocio.

Se usa para entender, diseñar, hojear, configurar, mantener y controlar la información sobre tales sistemas. Está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios.

UML tiene nueve diagramas fundamentales, clasificados en dos grupos, uno para modelar la estructura estática del sistema y otro para modelar el comportamiento dinámico [40]. Estos diagramas son: 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 implementación.

Se utilizará UML para modelar a través de algunos de los diagramas que presenta lo relacionado con el diseño e implementación del subsistema propuesto.

1.3.4. *Framework* (marcos de trabajo)

En el mundo de desarrollo de aplicaciones informáticas el uso de los *frameworks* se vuelve inminente, ya que estas estructuras conceptuales y tecnológicas de soporte definido, constituyen la base con la cual otro proyecto de *software* puede ser fácilmente organizado y desarrollado. Para la creación de servicios web, estas tecnologías también ganan su espacio debido que permiten a los desarrolladores dedicar mayor cantidad de tiempo a diseñar los servicios para brindar, y no en profundizar en detalles de bajo nivel de implementación para ponerles operativos.

Symfony2

Symfony2 es un *framework* diseñado para optimizar el desarrollo de las aplicaciones web mediante algunas de sus principales características. Para empezar, separa la lógica de negocio, la lógica de

servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que desde la propia instalación del *framework*, ya se tiene una estructura definida, así como la implementación de muchas funcionalidades, por lo que es innecesario cada vez que se crea una nueva aplicación web, preocuparse por la estructura que deba tener esta. *Symfony2* está desarrollado completamente con PHP 5. Es compatible con la mayoría de gestores de bases de datos, como *MySQL*, *PostgreSQL*, *Oracle* y *Microsoft SQL Server*.

Ha sido ideado para exprimir al límite todas las nuevas características de PHP 5.3 y por eso es uno de los *frameworks* PHP con buen rendimiento. Su arquitectura interna está completamente desacoplada, lo que permite reemplazar o eliminar fácilmente aquellas partes que no encajan en un proyecto [41].

Algunas características que contiene son: [42]

- Su código, y el de todos los componentes y librerías que incluye, se publican bajo la licencia MIT de *software* libre (es una de tantas licencias de *software* que ha empleado el Instituto Tecnológico de Massachusetts (MIT, *Massachusetts Institute of Technology*)).
- La documentación del proyecto también es libre e incluye varios libros y decenas de tutoriales específicos.
- Aprender a programar con *Symfony2* te permite acceder a una gran variedad de proyectos: el *framework* *Symfony2* para crear aplicaciones complejas, el micro *framework* *Silex* para sitios web sencillos y los componentes *Symfony2* para otras aplicaciones PHP.
- Los componentes de *Symfony2* son tan útiles y están tan probados, que proyectos tan gigantescos como Drupal 8 están contruidos con ellos.

Zend Framework

Zend Framework es un *framework* de código abierto para el desarrollo de aplicaciones y servicios web usando PHP 5.3+. Utiliza el 100% de código orientado a objetos y utiliza la mayor parte de las nuevas características de PHP 5.3, es decir, espacios de nombres, funciones, vinculantes estáticas finales de lambda y cierres.

La estructura de componentes de *Zend Framework* es única; cada componente está diseñado con pocas dependencias de otros componentes. Sigue el principio orientado a objetos de diseño sólido. Esta arquitectura de acoplamiento flexible permite a los desarrolladores utilizar cualquier componente que quieren [43].

El *Zend Framework* también proporciona una implementación completa del patrón Modelo-Vista-Controlador (MVC), que permite que la lógica empresarial de la aplicación que se separa de la interfaz de usuario y modelos de datos. Se recomienda este modelo para aplicaciones de medio y gran complejidad y se utiliza comúnmente para el desarrollo de aplicaciones web, ya que fomenta la reutilización del código y produce una estructura de código más manejable [44].

Kumbia

Kumbia es un *web framework* libre escrito en PHP5. Basado en buenas prácticas de desarrollo web, usado en *software* comercial y educativo, fomenta la velocidad y eficiencia en la creación y mantenimiento de aplicaciones web. Es compatible con motores de base de datos como *MySQL*, *PostgreSQL* y *Oracle*.

Usar *Kumbia* es fácil para personas que han usado PHP y han trabajado patrones de diseño para aplicaciones de Internet cuya curva de aprendizaje está reducida a un día. El diseño limpio y la fácil lectura del código se facilitan con esta herramienta. Desarrolladores pueden aplicar principios de desarrollo como DRY, KISS o XP, enfocándose en la lógica de aplicación y dejando atrás otros detalles que quitan tiempo.

Intenta proporcionar facilidades para construir aplicaciones robustas para entornos comerciales. Esto significa que el *framework* es muy flexible y configurable. Al escoger *Kumbia* está apoyando un proyecto publicado bajo licencia de *software* libre [45].

Selección del *framework* de desarrollo

Se elige *Symfony2* en su versión 2.2.6 porque es un *framework* de desarrollo de primer nivel. Su código, componentes y librerías se publican bajo licencia de *software* libre. Reúne buenas prácticas de desarrollo y ahora tiempo a los desarrolladores ya que automatiza procesos y tareas comunes. Esto también lo hace adaptable a cualquier proyecto. Es compatible con diferentes servidores de bases de datos. Fácil de instalar y configurar. Se tuvo en cuenta también que este *framework* forma

parte de la base tecnológica del proyecto principal ORIÓN. Además fue creado para el desarrollo de aplicaciones en PHP.

1.3.5. Sistemas gestores de bases de datos

Un Sistema Gestor de Bases de Datos (SGBD) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a esos datos. El objetivo primordial de un SGBD es proporcionar un entorno que sea a la vez conveniente y eficiente para ser utilizado al extraer y almacenar información de la base de datos. Es una aplicación que permite a los usuarios definir, crear y mantener la base de datos, y proporciona acceso controlado a la misma.

En la manipulación de una base de datos, los SGBD deben incluir un control de concurrencia, o sea, deben permitir a varios usuarios tener acceso "simultáneo" a la base de datos. Controlar la concurrencia implica que si varios usuarios acceden a la base de datos, la actualización de los datos se haga de forma controlada para que no haya problemas. Un SGBD también debe encargarse de cumplir las reglas de integridad y redundancias. Otra función importante en un SGBD es su capacidad de realizar copias de seguridad y de recuperación de datos. Restricción de accesos no autorizados [46].

MySQL

MySQL es un SGBD muy conocido y ampliamente usado por su simplicidad y notable rendimiento. Su libre distribución en Internet bajo licencia GPL (Licencia Pública General GNU) le otorgan como beneficios adicionales contar con un alto grado de estabilidad y un rápido desarrollo.

MySQL está disponible para múltiples plataformas. Está desarrollado en C/C++. La API se encuentra disponible en *C*, *C++*, *Eiffel*, *Java*, *Perl*, *PHP*, *Python*, *Ruby* y *TCL*. Se puede utilizar como cliente-servidor o incrustado en aplicaciones. Soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes para poder optimizar el SGBD a cada caso concreto [47].

MongoDB

MongoDB es una base de datos ágil que permite a los esquemas cambiar rápidamente cuando las aplicaciones evolucionan, proporcionando siempre la funcionalidad que los desarrolladores esperan de las bases de datos tradicionales, tales como índices secundarios, un lenguaje completo de búsquedas y consistencia estricta.

Ha sido creada para brindar escalabilidad, rendimiento y gran disponibilidad, escalando de una implantación de servidor único a grandes arquitecturas complejas de centros multidados. Brinda un elevado rendimiento, tanto para lectura como para escritura, potenciando la computación en memoria. La replicación nativa de *MongoDB* y la tolerancia a fallos automática ofrece fiabilidad a nivel empresarial y flexibilidad operativa [48].

MongoDB es una base de datos documental. Almacena los datos en documentos. Ofrece muchas de las características tradicionales de un *Relational Data Base Management System (RDBMS)* Sistema Gestor de Base de Datos Relacionales, como índices secundarios, consultas dinámicas, clasificación. Al mantener datos relacionados entre sí en los documentos, las consultas pueden ser mucho más rápido que en una base de datos relacional donde los datos relacionados se separan en varias tablas y luego tiene que ser unido más tarde.

Hace que sea fácil de ampliar la base de datos. Es posible aumentar la capacidad sin ningún tiempo de inactividad, lo cual es muy importante en la web cuando la carga puede aumentar repentinamente. Es muy fácil de instalar, configurar, mantener y utilizar. Es un proceso de servidor que se ejecuta en *Linux*, *Windows* y *OS X*. Almacena sus datos en archivos y utiliza archivos de memoria asignada para la gestión de datos para la eficiencia [48].

Selección del sistemas gestor de bases de datos

Se elige *MongoDB* en su versión 2.0.4 ya que dadas sus características se considera una buena propuesta para el almacenamiento de datos relacionados con el subsistema de promoción de anuncios lo cual se demuestra por el tránsito de información y la gestión de datos que el *software* requiere. Además es el usado para almacenar la información del buscador ORIÓN.

1.3.6. Herramienta CASE para el modelado UML

Visual Paradigm

Visual Paradigm for UML es una herramienta CASE que soporta el ciclo de vida completo del desarrollo de *software*: análisis y diseño orientados a objetos, implementación y pruebas. Ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite construir diagramas de diversos tipos, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML demostraciones interactivas y proyectos [49].

IBM Rational Rose Enterprise

IBM Rational Rose Enterprise proporciona un conjunto de prestaciones controladas por modelo para desarrollar muchas aplicaciones de *software*, incluidas aplicaciones *Ada*, *ANSI C++*, *C++*, *CORBA*, *Java*, *Java EE*, *Visual C++* y *Visual Basic*. El *software* permite acelerar el desarrollo de estas aplicaciones con código generado a partir de modelos visuales mediante el lenguaje UML.

Rational Rose Enterprise ofrece una herramienta y un lenguaje de modelado común para simplificar el entorno de trabajo y permitir una creación más rápida de *software* de calidad. Admite patrones de análisis, *ANSI C++*, *Rose J* y *Visual C++*, *Enterprise JavaBean 2.0* e ingeniería directa e inversa para algunas de las construcciones más comunes de *Java 1.5*. Es capaz de analizar la calidad del código y de generar código gracias a las prestaciones de sincronización configurable entre el modelo y el código. Permite utilizar un lenguaje para el ciclo de vida de análisis y desarrollo. Incluye un complemento de modelado web, que proporciona visualización, modelado y herramientas para desarrollar aplicaciones web. Proporciona el modelado UML para diseños de bases de datos. Permite integrar requisitos de datos y aplicación a través de diseños lógicos y físicos [50].

Selección de la herramienta CASE para el modelado UML

Teniendo en cuenta sus características y los beneficios que brinda para la construcción de *software*, especialmente referente al modelado, se decidió utilizar *Visual Paradigm for UML* en su versión 8.0 para el modelado de sistema propuesto. Además de ello, se tuvo en cuenta que esta constituye la herramienta que utiliza la Universidad, y dentro de ella el centro CIDI para el desarrollo de *software*.

1.3.7. Entorno de desarrollo

NetBeans

NetBeans es un *Integrated Development Environment* (IDE), Entorno de Desarrollo Integrado en español, para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en *Java* pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el *NetBeans IDE*. Es un producto libre y gratuito sin restricciones de uso.

NetBeans IDE permite desarrollar rápida y fácilmente aplicaciones *desktop Java*, móviles y aplicaciones web, así como aplicaciones HTML5, *JavaScript* y CSS. También proporciona un gran conjunto de herramientas para desarrolladores de PHP y C / C ++.

La plataforma *NetBeans* proporciona una arquitectura de aplicaciones fiables y flexibles. La arquitectura que promueve prácticas de desarrollo sostenible. Debido a que es modular, es fácil crear aplicaciones que sean robustas y extensibles [51].

PHPStorm

PhpStorm es un IDE, potente para desarrollo PHP que permite simplificar procesos. Su editor de PHP entiende perfectamente su estructura y soporta las versiones 5.3, 5.4, 5.5 y 5.6 de PHP. Entre otras características, este IDE proporciona finalización inteligente de código, comprobación de errores al instante o mezcla de lenguajes.

Es creado y mantenido por *JetBrains*, una organización con muchos productos similares en otros idiomas también. La gente de *JetBrains* respeta todas las comunidades de programadores, y por lo tanto proporcionan entornos de desarrollo para muchos lenguajes de programación, como *Java*. *PhpStorm* está construido utilizando el lenguaje de programación *Java* y ha sido cuidadosamente diseñado para ser capaz de soportar casi todas las herramientas comunes y no comunes que directa o indirectamente proporcionar asistencia en la programación [52].

Selección del entorno de desarrollo

Teniendo en cuenta sus características y los beneficios que brinda para la construcción de *software* se decidió utilizar el IDE *NetBeans* en su versión 8.0 porque es un producto libre y gratuito sin restricciones de uso, permite desarrollar rápida y fácilmente aplicaciones web, es modular y es fácil crear aplicaciones que sean robustas y extensibles.

1.3.8. Servidores web

Apache 2

Apache es una plataforma de servidores web de código fuente abierto. En la versión 2, el grupo Apache se ha concentrado en la escalabilidad, en la seguridad y en el rendimiento. Hoy en día, es una plataforma web muy utilizada. Funciona en *Linux* y en otros sistemas de *Unix*, también funciona en *Windows*.

Apache es uno de los primeros servidores web en integrar el protocolo HTTP 1.1. Es totalmente compatible con el nuevo estándar HTTP 1.1 y al mismo tiempo sigue siendo compatible con HTTP 1.0. Está preparado para todas las novedades del nuevo protocolo. No posee una interfaz de usuario gráfica para su administración. Se trata de un sencillo archivo de configuración llamado `httpd.conf` que se puede utilizar para configurar *Apache*. Únicamente necesita su editor de texto favorito. Sin embargo, es lo suficientemente flexible para permitirle repartir la configuración de su *host* virtual en múltiples archivos para no sobrecargar un único archivo `httpd.conf` con toda la gestión de las múltiples configuraciones de servidores virtuales. Es compatible con *Common Gateway Interface* (CGI) y aporta características extendidas como personalización de las variables de entorno y soporte de reparación de errores o *debugging*, que son difíciles de encontrar en otros servidores web.

Apache ofrece un amplio soporte de PHP utilizando el módulo *mod-php*. Le da una gran cantidad de flexibilidad en el registro y la monitorización del estado del servidor. El estado del servidor puede monitorizarse mediante un navegador web. Además, puede adaptar sus archivos de registro a su gusto [53].

Nginx

Es un servidor web inverso ligero de alto rendimiento y un proxy para protocolos de correo electrónico (IMAP, POP3). Es *software* libre y de código abierto, licenciado bajo la Licencia BSD simplificada. Es multiplataforma, por lo que corre en sistemas tipo *Unix* (GNU/Linux, BSD, Solaris, Mac OS X) y *Windows* (*nginx*).

El sistema es usado por una larga lista de sitios web conocidos, tales como: *WordPress*, *Netflix*, *Hulu*, *GitHub*, *Ohloh*, *SourceForge*, *TorrentReactor* y partes de *Facebook* (como el servidor de descarga de archivos zip pesados) (*nginx*) [54].

Entre las principales características presentes en este servidor se pueden listar:

- Servidor de archivos estáticos, índices y autoindexado.
- Proxy inverso con opciones de caché.
- Balanceo de carga.
- Tolerancia a fallos.
- Soporte de HTTP sobre SSL.
- Soporte para *FastCGI* con opciones de caché.
- Servidores virtuales basados en nombre y/o en dirección IP.

- Soporte para autenticación.
- Compatible con IPv6.
- Soporte para protocolo SPDY.
- Compresión gzip.
- Habilitado para soportar más de 10.000 conexiones simultáneas.

Conclusiones parciales

En el presente capítulo se analizaron los principales elementos teóricos necesarios para la realización del trabajo de diploma en cuestión y para ello se realizó un estudio del arte de las principales herramientas que ofrecen servicios de promoción de anuncios, las cuales están desarrolladas en un entorno privativo pero pueden servir de guía para la implementación de las funcionalidades del subsistema propuesto.

Se puede concluir que:

- El estudio de los sistemas internacionales analizados permitió identificar algunas de las funcionalidades que son fundamentales para la solución del trabajo de diploma en cuestión como: la inyección de anuncios dentro de los resultados de búsqueda de un buscador; la gestión de palabras clave en los anuncios como forma de identificación de preferencias del público objetivo; la técnica de conteo de clic para obtener ingresos mediante un costo estipulado para un clic.
- Se seleccionó *Visual Paradigm 8.0* como herramienta CASE, la metodología de desarrollo de *software OpenUP*, el lenguaje de programación PHP 5.3, *Symfony 2.3* como *framework* de desarrollo, *Netbean 8.0* como IDE de desarrollo, *MongoDB 2.0.4* como sistema gestor de base de datos, *Apache 2.0* como servidor web en correspondencia con las facilidades que ofrecen al equipo de desarrollo y basándose también en las que utiliza ORIÓN como sistema rector, favoreciendo la integración del subsistema de promoción de anuncios.

CAPÍTULO 2. CARACTERÍSTICAS Y DISEÑO DEL SISTEMA

Introducción

En el presente capítulo se define la propuesta del sistema a desarrollar. Se identifican las clases del dominio y la relación entre ellas, los requisitos funcionales y no funcionales que debe cumplir la propuesta de solución y se realiza el modelado de casos de uso de la misma. Se realiza el diseño del sistema desarrollando los diagramas de clases del diseño, para ello se analizan los diferentes patrones arquitectónicos y de diseño y se confeccionan los diagramas de interacción.

2.1. Propuesta del sistema

Teniendo en cuenta la problemática existente y para darle cumplimiento al objetivo general se propone desarrollar un subsistema de promoción de anuncios que permita al motor de búsqueda ORIÓN brindar estos servicios a empresas e instituciones y así obtener ingresos, siendo rentable en el tiempo y aportando a la economía del país garantizando su sustento.

Se propone implementar un subsistema que permita a los clientes promocionar sus productos y servicios a través del motor de búsqueda ORIÓN. Dicha información será almacenada en una base de datos. Estos anuncios se mostrarán a los usuarios que accedan al buscador si están asociados a los términos de búsqueda. Se implementará una capa de servicios para gestionar las peticiones que realice el buscador. Esta será la encargada de verificar si existen en la base de datos del subsistema anuncios relacionados con los datos recibidos. Además, devolverá estos anuncios al buscador ORIÓN y realizará peticiones a este de los registros acerca de los usuarios que se redireccionan a las páginas patrocinadoras de los anuncios. Los términos son las palabras clave o criterios de búsqueda que introducen los usuarios que acceden al buscador y los anuncios pueden ser textos descriptivos o imágenes.

En el subsistema se gestionarán los usuarios: moderadores, administradores y clientes así como los anuncios que dichos clientes desean publicar. La propia aplicación se encargará de almacenar los clics que realizan los internautas en los anuncios mostrados en el buscador y de gestionar informes acerca del estado de los anuncios para informar al cliente de la situación que presenta este.

2.2. Modelo de dominio

Un modelo de dominio es una representación de las clases conceptuales del mundo real. Su objetivo es comprender y describir las clases más importantes dentro del contexto del sistema, por lo cual este puede ser tomado como el punto de partida para el diseño del mismo [55].

El diagrama de dominio realizado representa cómo los buscadores realizan el proceso de brindar servicios de promoción de anuncios. Al realizar el mismo se obtuvo una visión de los elementos que necesita el buscador ORIÓN para brindar estos servicios. En general, un buscador es accedido por usuarios, los cuales introducen términos de búsqueda relacionados con la información que desean obtener. El motor de búsqueda accede a la base de datos en busca de anuncios que se relacionen con los términos introducidos y los muestra al usuario. Un cliente puede solicitar estos servicios e insertar los anuncios que desea promocionar. Solo paga cuando se realice *click* en sus anuncios cuando se muestren en el buscador.

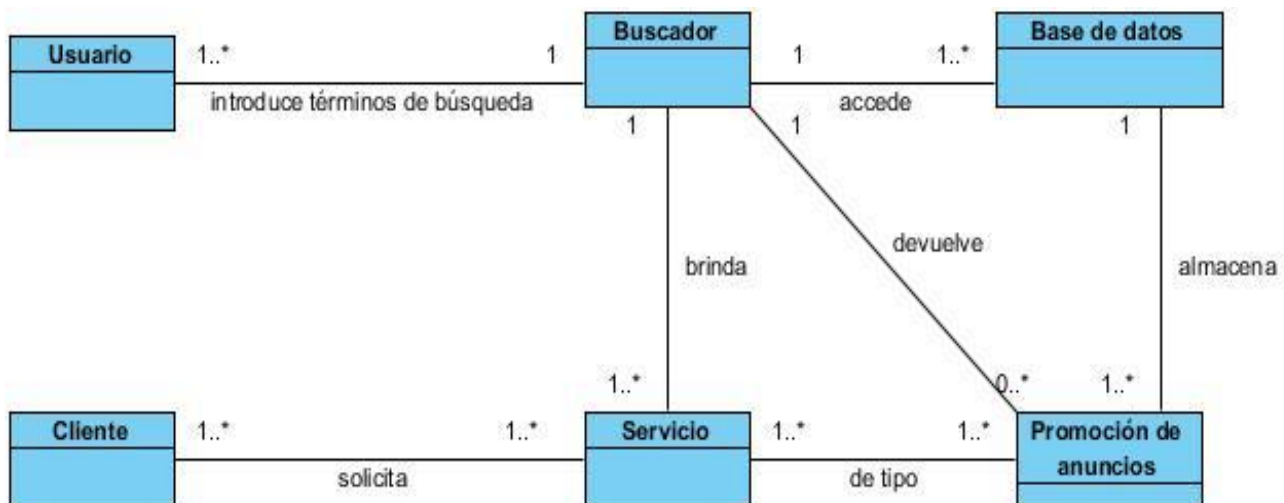


Imagen 2. Modelo de dominio.

2.2.1. Descripción de las clases del dominio

Usuario: son los internautas que acceden al buscador en busca de información para lo cual introducen los términos de búsqueda.

Buscador: aplicación informática que retorna la información asociada a los términos de búsqueda de los usuarios.

Base de datos: es donde se almacena la información que será mostrada por el buscador.

Cliente: entidad que paga cierta cantidad de dinero por promocionar sus productos y servicios a través del buscador. Puede ser una persona, empresa o institución.

Servicio: funcionalidad que es ofrecida para ser utilizada por los clientes, servicios de promoción de anuncios que brinda el buscador a estos.

Promoción de anuncios: los diferentes anuncios que muestra el buscador a los usuarios promocionando los servicios y productos de los clientes.

2.3. Especificación de los requisitos del sistema

Para la implementación del subsistema propuesto se identificaron requisitos funcionales y no funcionales que debía cumplir el mismo para lograr su funcionamiento.

2.3.1. Requisitos funcionales

Los requisitos funcionales (RF) son una representación de lo que debe cumplir el sistema para llevar a cabo su función, son los que definen que hará el mismo.

A continuación se presentan los requisitos funcionales que debe cumplir el subsistema propuesto.

Requisitos funcionales del subsistema de promoción de anuncios.

- **RF1.** Registrar usuario: este RF permitirá que los usuarios se registren en el subsistema para luego poder autenticarse y acceder a la aplicación.
- **RF2.** Autenticar usuario: este RF permitirá que los usuarios se puedan autenticar en el subsistema para acceder a este. Antes de autenticarse los usuarios deben estar registrados.
- **RF3.** Insertar usuario: este RF permitirá al administrador del subsistema insertar usuarios.
- **RF4.** Editar usuario: este RF permitirá al administrador del subsistema editar los usuarios registrados.
- **RF5.** Eliminar usuario: este RF permitirá al administrador del subsistema eliminar los usuarios registrados.
- **RF6.** Listar usuarios: este RF listará los usuarios registrados en el subsistema.
- **RF7.** Mostrar perfil de usuario: este RF permitirá mostrar el perfil de un usuario determinado.
- **RF8.** Buscar usuario: este RF permitirá buscar un usuario si existe en el subsistema.
- **RF9.** Insertar anuncio: este RF permitirá a los usuarios registrados insertar anuncios en el subsistema.

- **RF10.** Editar anuncio: este RF permitirá a los usuarios registrados editar los anuncios existentes en el subsistema.
- **RF11.** Eliminar anuncio: este RF permitirá a los usuarios registrados eliminar anuncios existentes en el subsistema si poseen los permisos para hacerlo.
- **RF12.** Listar anuncios: este RF listará los anuncios existentes en el subsistema.
- **RF13.** Buscar anuncio: este RF permitirá buscar un anuncio según su nombre si existe en el subsistema.
- **RF14.** Insertar término: este RF permitirá a los usuarios registrados insertar términos en el subsistema.
- **RF15.** Editar término: este RF permitirá a los usuarios registrados editar los términos existentes en el subsistema si poseen los permisos para hacerlo.
- **RF16.** Eliminar término: este RF permitirá a los usuarios registrados eliminar los términos existentes en el subsistema si poseen los permisos para hacerlo.
- **RF17.** Listar términos: este RF listará los términos existentes en el subsistema.
- **RF18.** Buscar término: este RF permitirá buscar un término según su nombre si existe en el subsistema.
- **RF19.** Activar anuncio: este RF permitirá a los usuarios registrados activar los anuncios existentes en el subsistema si poseen los permisos para hacerlo.
- **RF20.** Desactivar anuncio: este RF permitirá a los usuarios registrados desactivar los anuncios existentes en el subsistema si poseen los permisos para hacerlo.
- **RF21.** Insertar rol: este RF permitirá al administrador del subsistema insertar roles.
- **RF22.** Editar rol: este RF permitirá al administrador del subsistema editar los roles existentes en el subsistema.
- **RF23.** Eliminar rol: este RF permitirá al administrador del subsistema eliminar los roles existentes en el subsistema.
- **RF24.** Notificar al cliente cuando su anuncio se publique: este RF permitirá comunicarle a los clientes el momento en que su anuncio se publica.
- **RF25.** Exportar informe: este RF permitirá a los usuarios registrados exportar el informe creado sobre sus anuncios si están activos. Al seleccionar la opción "Exportar informe" el subsistema lo crea y brinda la opción para exportarlo.

Requisitos funcionales de la capa de servicios.

- **RF26.** Contar cantidad de *clics* que un usuario realiza en un anuncio: este RF permitirá contar la cantidad de *clics* que un usuario realiza en un anuncio que ha sido mostrado en el buscador ORIÓN.
- **RF27.** Devolver anuncios relacionados con término: este RF devolverá los anuncios al buscador ORIÓN si están relacionados con los criterios de búsqueda.
- **RF28.** Guardar registro: este RF permitirá guardar los datos de los usuarios cuando realicen clic en los anuncios mostrados en el buscador.
- **RF29.** Buscar registro: este RF permitirá buscar los registros de los anuncios que han sido visitados en el buscador según la fecha.
- **RF30.** Listar registros: este RF permitirá listar los registros existentes en el subsistema.
- **RF31.** Eliminar registro: este RF permitirá a los usuarios registrados eliminar los registros existentes en el subsistema si poseen los permisos para hacerlo.

2.3.2. Requisitos no funcionales.

Los requisitos no funcionales (RNF), no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de este como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento [56]. Son propiedades o cualidades que el producto debe tener, aunque no formen parte de su función. Existen múltiples categorías para clasificar los RNF, siendo las siguientes algunas de ellas:

Requisitos de usabilidad.

- **RNF1.** Facilidad de uso por parte de los usuarios: La interfaz debe ser lo más descriptiva posible, permitiendo que las operaciones a realizar por los usuarios estén bien descritas, de manera que se puedan entender claramente. Los botones tendrán nombres que sugieran la operación a realizar. Se mostrarán mensajes informando la realización de las acciones y de errores.
- **RNF2.** El subsistema podrá ser utilizado por los usuarios que se registren y accedan.

Requisitos de eficiencia.

- **RNF3.** El subsistema deberá tener un tiempo de respuesta máximo de 4 segundos para un total de 100 usuarios concurrentes, tanto para los accesos a la base de datos, como para las respuestas a las peticiones realizadas a la capa de servicios.

- **RNF4.** El subsistema debe permitir la navegación de 100 usuarios como máximo simultáneamente.

Requisitos de seguridad.

- **RNF5.** Confidencialidad: en el sistema solo podrá gestionar información el personal autorizado, con los permisos correspondientes, por lo que será necesaria una autenticación previa.
- **RNF6.** Integridad: el software deberá estar protegido contra acciones no autorizadas o que puedan afectar la integridad de los datos.
- **RNF7.** Disponibilidad: el sistema deberá estar disponible las 24 horas del día para todos los usuarios, exceptuándose los días que se encuentre en actualización, corrección de errores, o soporte.

Requisitos de *software*.

- **RNF8.** Para el servidor de base de datos: *MongoDB* versión 2.0.4.
- **RNF9.** Para el servidor web: Apache 2.2.4.
- **RNF10.** Dependencias (php5, zphp5-cli, php5-sqlite, php5-mongodb).
- **RNF11.** Sistema operativo CENTOS (última versión estable) a 64 Bits.
- **RNF12.** PC cliente: cliente ligero y un navegador.

Requisitos de *hardware*.

- **RNF13.** Para el servidor de base de datos: RAM: 4 GB, CPU: *dualcore*, Disco duro: 1 TB.
- **RNF14.** Para el servidor web: RAM: 2 GB, CPU: *dualcore*, Disco duro: 50 GB.
- **RNF15.** PC cliente: RAM: 256.

2.4. Casos de uso del sistema

Los casos de uso facilitan una descripción de cómo el sistema se usará. Describen la manera en que los actores interactúan con el sistema. Son simplemente, textos escritos que describen el papel de un actor que interactúa con el acontecer del sistema [57].

Los casos de uso son documentos de texto, no diagramas. Se utilizan para agrupar requisitos funcionales que se relacionan.

2.4.1. Definición de los casos de uso del sistema

Los casos de usos identificados fueron los siguientes:

- **CU1.** Registrar usuario: relaciona los requisitos funcionales **RF1**.
- **CU2.** Autenticar usuario: relaciona los requisitos funcionales **RF2**.
- **CU3.** Gestionar usuario: relaciona los requisitos funcionales **RF3, RF4, RF5, RF6, RF7, RF8**.
- **CU4.** Gestionar anuncio: relaciona los requisitos funcionales **RF9, RF10, RF11, RF12, RF13**.
- **CU5.** Gestionar término: relaciona los requisitos funcionales **RF14, RF15, RF16, RF17, RF18**.
- **-CU6.** Activar o desactivar anuncio: relaciona los requisitos funcionales **RF19, RF20**.
- **CU7.** Gestionar rol: relaciona los requisitos funcionales **RF21, RF22**.
- **CU7.** Notificar al cliente: relaciona los requisitos funcionales **RF23**.
- **CU8.** Crear y exportar informe: relaciona los requisitos funcionales **RF24, RF25**.
- **CU9.** Contar cantidad de clics que un usuario realiza en un anuncio: relaciona los requisitos funcionales **RF26**.
- **CU10.** Devolver anuncios relacionados con término: relaciona los requisitos funcionales **RF27**.
- **CU11.** Gestionar registro: relaciona los requisitos funcionales **RF28, RF29, RF30, RF31**.

2.4.2. Definición de los actores del sistema

Un actor del sistema representa un conjunto coherente de roles que juegan los usuarios. Además, no es parte del sistema, es el encargado de inicializar los CU, y es beneficiado con el resultado de los mismos. Es una entidad externa al sistema que se modela y que puede interactuar con él. Los actores representan a los seres humanos, a un *software*, *hardware* externo u otros sujetos que interactúen con el sistema que se esté especificando [56].

Un actor es cualquier cosa con comportamiento, incluyendo el propio sistema que se está estudiando cuando solicita los servicios de otros sistemas.

Tabla 2. Descripción de los actores del sistema.

Actor	Descripción
Usuario	Es el usuario que se registra en el subsistema y se autentica.
Cliente	Será el encargado de insertar los anuncios y los términos y podrá editar los mismos para ello

	tendrá que estar autenticado. Podrá exportar informes de sus anuncios.
Moderador	Es el encargado de revisar, activar y desactivar los anuncios que los clientes han insertado. También, puede editar e insertar anuncios y términos. Exportar informes. Para ello debe estar autenticado.
Administrador	Es el encargado de administrar todo el subsistema de promoción de anuncios y es el que tendrá el máximo de privilegios. Se encargará de gestionar los usuarios, clientes, roles y toda la información.
Buscador ORIÓN	Es el encargado de hacer la petición de los anuncios al subsistema, mediante una capa de servicios y mostrar los mismos a los usuarios.

2.4.3. Diagrama de casos de uso del sistema

El diagrama de casos de uso del sistema ayuda a comprender gráficamente los procesos del sistema y su interacción con los actores.

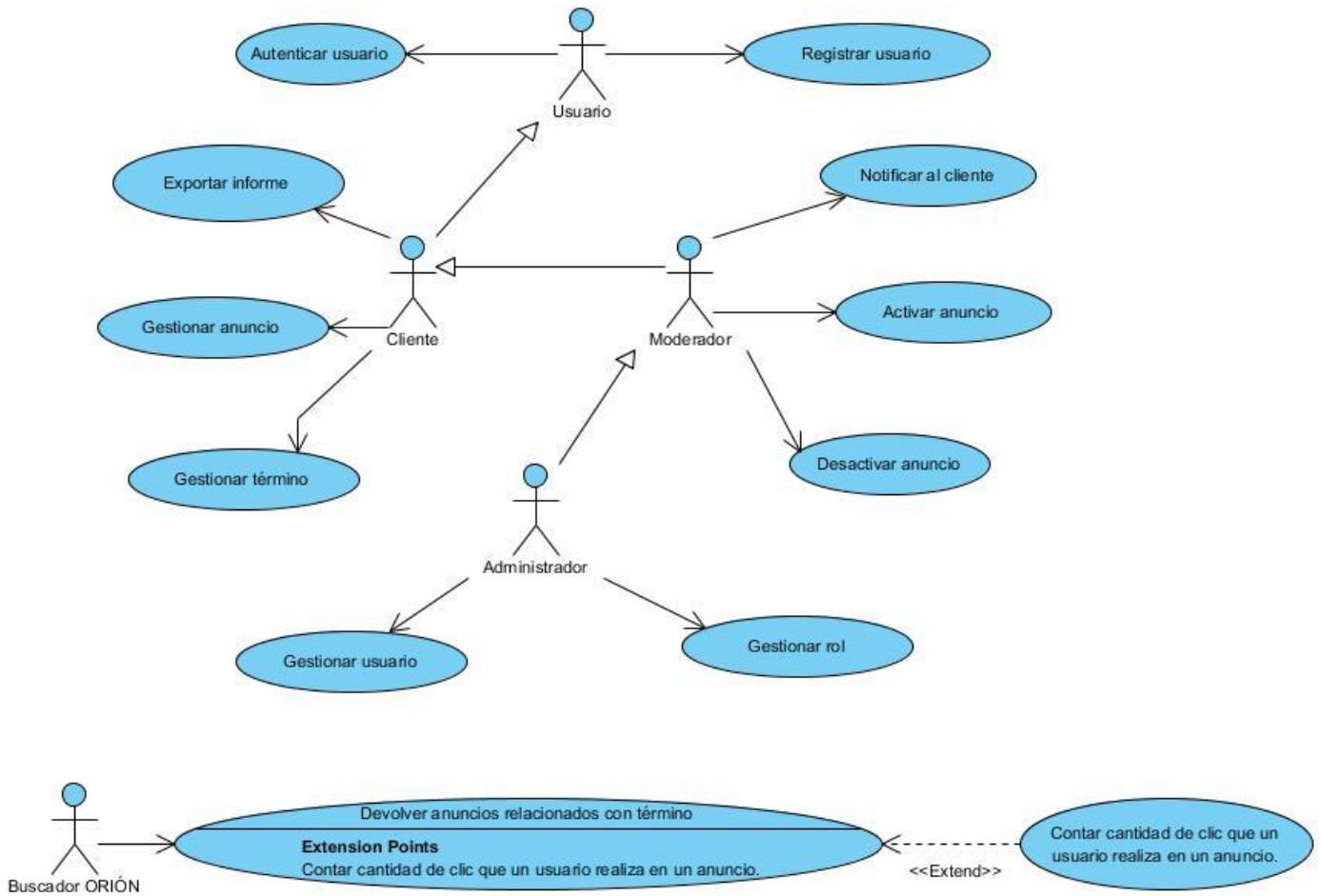


Imagen 3. Diagramas de casos de uso del sistema.

2.4.4 Descripción de los casos de uso del sistema

La descripción de los casos de uso facilita la comprensión de las diferentes funcionalidades del sistema. A continuación se describe el caso de uso gestionar anuncio, por ser uno de los más significativos en el sistema.

Tabla 3. Descripción del caso de uso Gestionar anuncio.

Caso de uso	Gestionar anuncio.
Objetivo	Insertar un nuevo anuncio en el sistema, editar, eliminar, listar o buscar los anuncios que ya estén previamente almacenados.
Actores	Moderador(inicia) Administrador(inicia) Cliente(inicia)
Resumen	El caso de uso inicia cuando el moderador, cliente o el administrador seleccionan la opción Anuncios.
Complejidad	Alta
Prioridad	Crítico

Precondiciones	El usuario debe estar autenticado en el sistema.	
Postcondiciones	Se inserta, edita, elimina, lista o busca un anuncio.	
Flujo de eventos		
Flujo básico: Gestionar anuncio		
	Actor	Sistema
1	Selecciona la opción Anuncios.	
2		Muestra una lista de las diferentes opciones del menú seleccionado que contiene: <ol style="list-style-type: none"> 1. Insertar anuncio (1). Ver Sección 1: “Insertar anuncio”. 2. Editar anuncio (2). Ver Sección 2: “Editar anuncio”. 3. Eliminar anuncio (3). Ver Sección 3: “Eliminar anuncio”. 4. Listar anuncios (4). Ver Sección 4: “Listar anuncios”. 5. Buscar anuncio (5). Ver Sección 5: “Buscar anuncio”.
3		Termina el CU.
Sección 1: “Insertar anuncio”		
Flujo básico: Insertar anuncio.		
	Actor	Sistema
1	Selecciona la opción Crear Anuncio.	
2		Muestra un formulario con los siguientes campos correspondientes para adicionar un nuevo anuncio: nombre, descripción, url, lista de términos, precio, imagen.
3	Introduce los datos correspondientes y hace clic en el botón “Crear”.	
4		Inserta un nuevo anuncio. Muestra la lista de los anuncios creados y un mensaje indicando que el anuncio fue creado satisfactoriamente.
5		Termina el CU.
Flujos alternos		
3a. El usuario deja campos vacíos.		
	Actor	Sistema
1		Señala los campos que aún estén vacíos, para que se llenen. Muestra un mensaje indicando que el campo no puede estar vacío.
3b. El usuario inserta datos inválidos.		
	Actor	Sistema
1		Señala los campos que tengan datos inválidos.

		Muestra un mensaje indicando que entre el dato correctamente.
3c. El usuario intenta subir un archivo que no sea de tipo imagen.		
	Actor	Sistema
1		Muestra un mensaje de error donde se le comunica al usuario que solo se puede subir archivos de tipo imagen.
Sección 2: "Editar anuncio"		
Flujo básico: Editar anuncio.		
	Actor	Sistema
1	El usuario selecciona la opción Editar del anuncio.	
2		Muestra un formulario con los siguientes campos correspondientes para adicionar un nuevo anuncio: nombre, descripción, url, lista de términos, precio, imagen.
3	El usuario modifica los datos que desea del anuncio seleccionado y hace clic en el botón "Editar".	
4		Se guarda el anuncio con los nuevos datos editados. Muestra un mensaje indicando que el anuncio fue guardado satisfactoriamente.
5		Termina el CU.
Flujos alternos		
3a El usuario deja campos vacíos.		
	Actor	Sistema
1		Señala los campos que aún estén vacíos, para que se llenen. Muestra un mensaje indicando que el campo no puede estar vacío.
3b El usuario inserta datos inválidos.		
	Actor	Sistema
1		Señala los campos que tengan datos inválidos. Muestra un mensaje indicando que entre el dato correctamente.
Sección 3: "Eliminar anuncio"		
Flujo básico: Eliminar anuncio.		
	Actor	Sistema
1	Selecciona la opción Eliminar del anuncio.	
2		Muestra un mensaje para que el usuario confirme la eliminación del anuncio.
3	Confirma la acción presionando en el botón Sí.	

4		Elimina un anuncio de la base de datos del sistema y muestra el listado de los anuncios restantes. Se muestra un mensaje comunicando que el anuncio ha sido eliminado satisfactoriamente.
5		Termina el CU.
Sección 4: "Listar anuncios"		
Flujo básico: Listar anuncios.		
	Actor	Sistema
1	Selecciona la opción Lista de Anuncio.	
2		Muestra la lista de los anuncios existentes.
3		Termina el CU.
Sección 5: "Buscar anuncio"		
Flujo básico: Buscar anuncio.		
	Actor	Sistema
1	Se introduce el nombre del anuncio en el formulario "Buscar".	
2		Se muestra el anuncio que concuerde con los criterios introducidos. Muestra un mensaje que comunica la cantidad de resultados encontrados.
5		Termina el CU.
Flujos alternos		
1a. La búsqueda no devuelve resultado.		
	Actor	Sistema
1.		Muestra un mensaje informado que no se obtuvo resultados para los criterios insertados.
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos funcionales	RF9, RF10, RF11, RF12, RF13.	
Asuntos pendientes		

2.5. Patrones de diseño

Un patrón de diseño provee un esquema para refinar los subsistemas o componentes de un sistema de software, o las relaciones entre ellos. Describe la estructura comúnmente recurrente de los componentes en comunicación, que resuelve un problema general de diseño en un contexto particular [58].

2.5.1. Patrones GRASP

Los patrones *General Responsibility Assignment Software Patterns* (GRASP), en español Patrones Generales de *Software* para Asignar Responsabilidades. Describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones [59].

Patrón Experto: las responsabilidades deben ser asignadas a las clases que poseen la información para realizar dicha responsabilidad. Una clase, contiene toda la información necesaria para realizar la labor que tiene encomendada. Este patrón se identificó en las clases entidades las cuáles poseen la información para realizar las responsabilidades que les fueron asignadas. Un ejemplo de la utilización de este patrón es la clase *Anuncio* que cuando se desea mostrar los anuncios almacenados en el subsistema es la única que contiene esta información. De la misma manera sucede con la clase *Usuario* que es la encargada de conocer quiénes son los usuarios registrados en el subsistema.

Patrón Creador: asignarle a una clase la responsabilidad de crear una instancia de otra. Dentro del sistema este patrón se evidencia en las acciones de los controladores, las cuales crean objetos del modelo o los formularios que representan las entidades. Este patrón se identificó en las clases controladoras ya que estas se encargan de crear instancias de objetos de otras clases. Un ejemplo de ello es la clase controladora *AnuncioController* que crea instancias de la entidad anuncio o los formularios que la representan.

Patrón Controlador: es el encargado de asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase que represente una de las siguientes opciones: Una respuesta, ejecutar un método o mostrar un mensaje. Esto se evidencia en el subsistema ya que para cada petición o evento que se genera existe un controlador con la responsabilidad de obtenerlos y devolver una de estas opciones. Un ejemplo de ello es el controlador *UsuarioControler* el cual se encarga de mostrar un mensaje cuando se inserta un usuario. El *TerminoController* que cuando se edita un término ejecuta el método *update* que actualiza los campos editados.

2.5.2. Patrones GoF

Los patrones *Gang of Four* (GoF) o “Pandilla de los Cuatro” en español, describen las formas comunes en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. Tratan la relación entre clases, la combinación de clases y la formación de estructuras de mayor complejidad. Permiten crear grupos de objetos para ayudar a realizar tareas complejas [60].

De estos patrones se utilizó:

Patrón *Observer* (Observador): define una dependencia de uno-a-muchos entre objetos, de forma que cuando un objeto cambie de estado se notifique y actualicen automáticamente todos los objetos que dependen de él. Este patrón se identificó en la clase cliente ya que si se elimina este se eliminan automáticamente los anuncios asociados a ese cliente.

Patrón *Decorator* (Decorador): añade responsabilidades adicionales a un objeto dinámicamente. Se utiliza este patrón mediante el uso de la plantilla global o base que decora el contenido de las vistas.

2.6. Patrones arquitectónicos

Los patrones arquitectónicos de *software* permiten representar de forma concreta la estructura y funcionamiento interno de un sistema, lo que posibilita su entendimiento, la organización del desarrollo del mismo. “La arquitectura de *software* de un sistema es la estructura o estructuras del sistema, lo cual abarca componentes de *software*, las propiedades visibles externamente de esos componentes, y las relaciones entre ellas” [61]. A continuación se muestra la arquitectura del subsistema.

2.6.1. Patrón arquitectónico Modelo – Vista – Controlador

Symfony2 basa su funcionamiento interno en la famosa arquitectura Modelo - Vista – Controlador. Sólo proporciona herramientas para la parte del Controlador y de la Vista. La parte del Modelo es responsabilidad del programador, aunque existen librerías para integrar fácilmente los ORM más conocidos, como *Doctrine* y *Propel* [62].

Model View Controller (MVC): (Modelo Vista Controlador). Es un patrón de arquitectura de *software* que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Este patrón plantea la separación del problema en tres capas: la capa **modelo**, que representa la realidad; la capa **controlador**, que conoce los métodos y atributos del modelo, recibe y realiza lo que el usuario quiere hacer; y la capa **vista**, que muestra un aspecto del modelo y es utilizada por la capa anterior para interactuar con el usuario.

Este patrón de arquitectura se evidencia en el subsistema en la estructura de los directorios donde se almacenan los componentes. En el directorio *Controller* se encuentran los componentes que contienen el código de los controladores. En *Views* se encuentran los de las interfaces de la aplicación y en *Document* los que definen los modelos de datos.

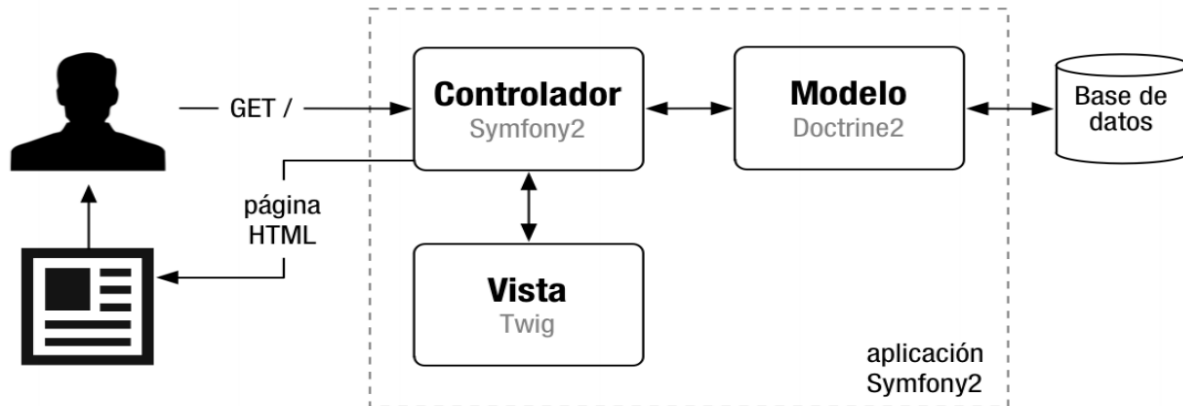


Imagen 4. Modelo-Vista-Controlador.

2.7. Modelo de diseño

El modelo de diseño muestra los objetos o clases de un sistema y los diferentes tipos de relaciones entre estas entidades. Es el puente entre los requerimientos y la implementación del sistema. Está compuesto por clases del diseño, subsistemas, paquetes, colaboraciones, y las relaciones entre ellos [56].

2.7.1. Diagrama de clases de diseño

Un diagrama de clases del diseño describe gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación. Muestra las relaciones existentes entre las clases del sistema. Son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema. A continuación se muestra el diagrama de clases de diseño del caso de uso Gestionar anuncio.

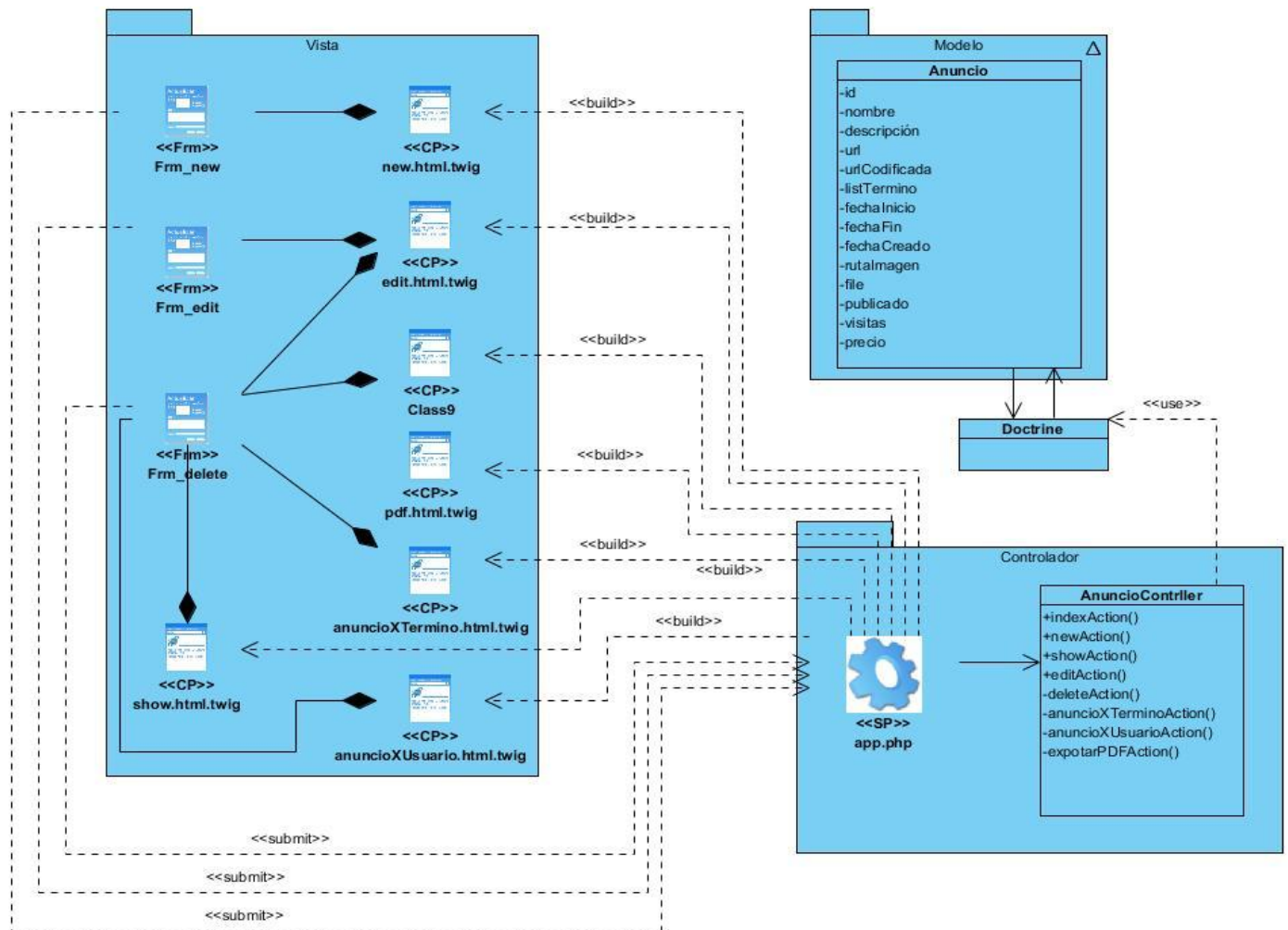


Imagen 5. Diagrama de clases de diseño del caso de uso Gestionar anuncio.

2.8. Modelo de datos

Un modelo de datos es una descripción abstracta de las clases de una organización y sus relaciones entre ellas [63]. Es la descripción de una base de datos. Típicamente un modelo de datos permite describir las estructuras de datos de la base, su tipo, descripción y la forma en que se relacionan, restricciones de integridad entre otros, es factible pensar que un modelo de datos permite describir los elementos de la realidad que intervienen en un problema dado y la forma en que se relacionan esos elementos entre sí. El Modelo de base de datos del subsistema está compuesto por cinco clases con sus atributos.

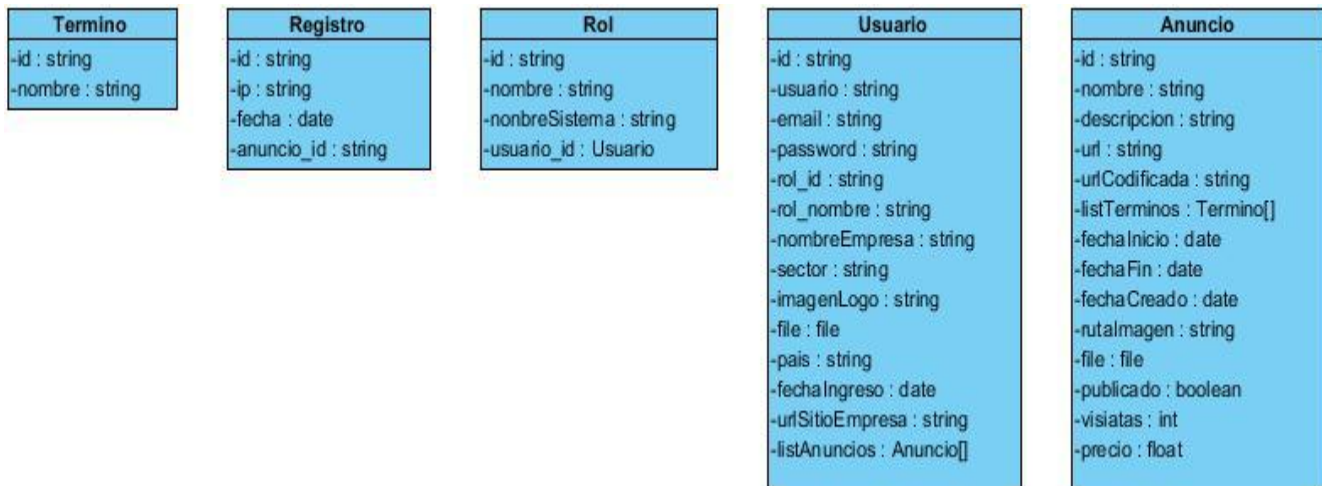


Imagen 6. Modelo de datos.

Conclusiones parciales

En este capítulo se inició el desarrollo de la propuesta de solución.

- El diseño del modelo de dominio referente a la realización de servicios de promoción de anuncios por los buscadores permitió lograr una mejor familiarización con los elementos involucrados en el desarrollo plasmado en las descripciones textuales de la propuesta de solución.
- Se obtuvo un listado de funcionalidades que debe tener el subsistema, expresados en los requerimientos funcionales y finalmente, se pudo obtener los casos de uso del sistema.
- Partiendo de este punto, base de todo el proceso de desarrollo, se pudo comenzar la construcción de la propuesta, velando por el cumplimiento de todos los requerimientos y funcionalidades consideradas en este capítulo.
- De igual manera se plasmaron los artefactos que dan respuesta a la metodología seleccionada y que brindarán un mayor entendimiento al problema y su solución.

CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBAS

Introducción

En el desarrollo del software se deben tener en cuenta las fases para la construcción de un producto. Luego de la fase de diseño se debe comenzar la implementación que no es más que llevar a cabo la codificación de las funcionalidades y materializar las fases anteriores teniendo en cuenta las descripciones, diseño y arquitecturas propuestas.

Luego de la implementación se debe comprobar que todo funcione correctamente, verificando el cumplimiento de los requisitos funcionales mediante las pruebas de *software*. En el presente capítulo se describen los temas referentes a la implementación y prueba del sistema.

3.1. Modelo de implementación

El modelo de implementación representa el empaquetado físico de las partes reutilizables del sistema en unidades sustituibles llamadas componentes. Muestra la implementación de los elementos del diseño mediante componentes.

3.1.1. Diagramas de componentes

Un diagrama de componentes muestra dependencias entre los componentes, describiendo los elementos físicos del sistema y sus relaciones.

A continuación se muestra el diagrama de componentes del *framework Symfony2*, donde se integra el sistema propuesto:

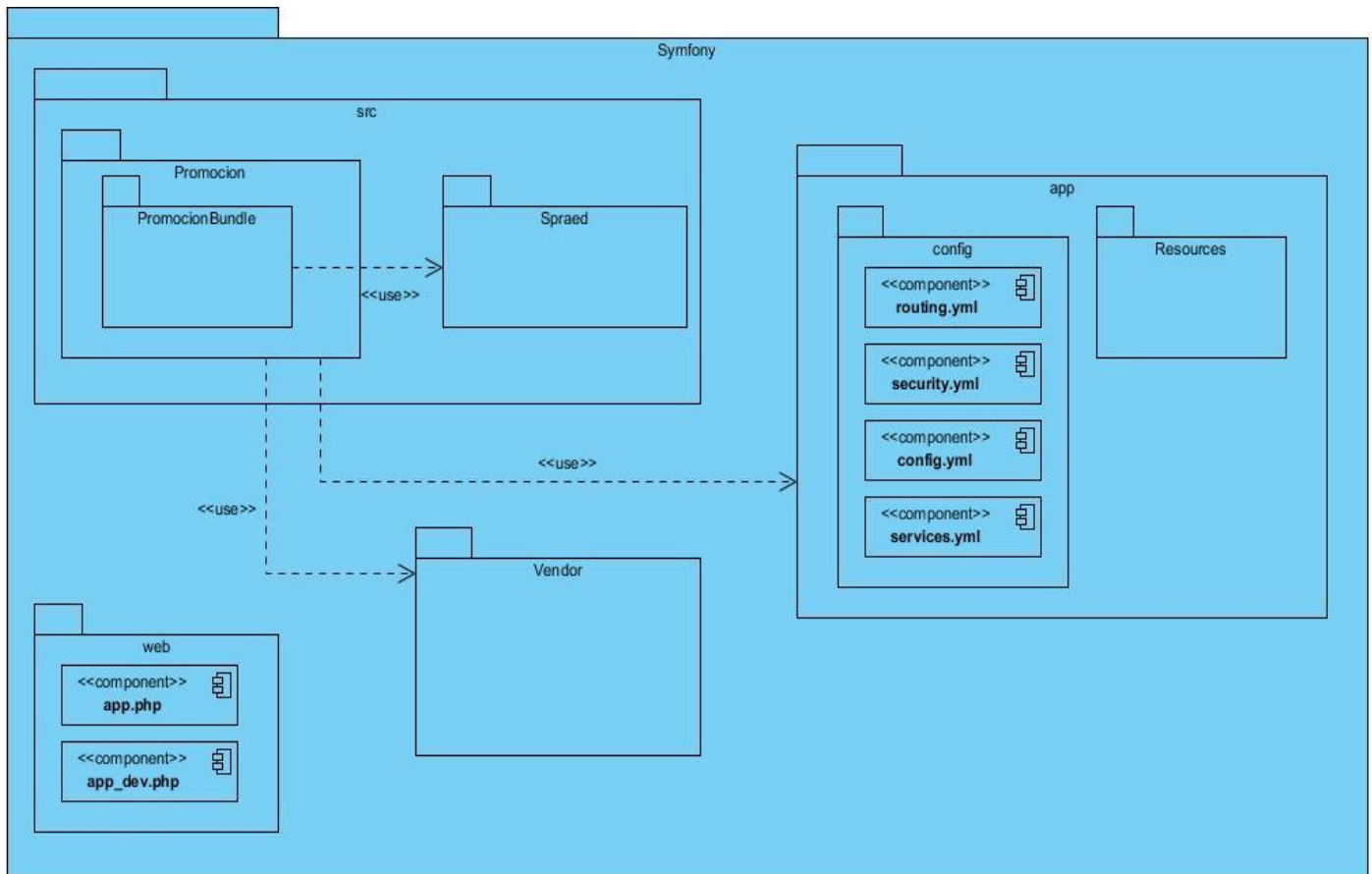


Imagen 7. Diagrama de componentes.

El sistema desarrollado se encuentra empaquetado en un *bundle*, donde se pueden encontrar distribuidos tres paquetes: modelo, vista y controlador correspondiendo con el patrón arquitectónico MVC. Las clases controladoras, encargadas de procesar las peticiones de los usuarios a través de los métodos que tienen implementados se encuentran en el paquete Controlador. Las entidades del sistema, a través de las cuales se realiza el acceso a la base de datos se encuentran en el paquete Modelo y los archivos que permiten visualizar las respuestas que devuelven los controladores al usuario se encuentran en el paquete Vista.

A continuación se muestra el diagrama de componentes de *PromocionBundle*, donde se encuentran los principales componentes del sistema desarrollado.

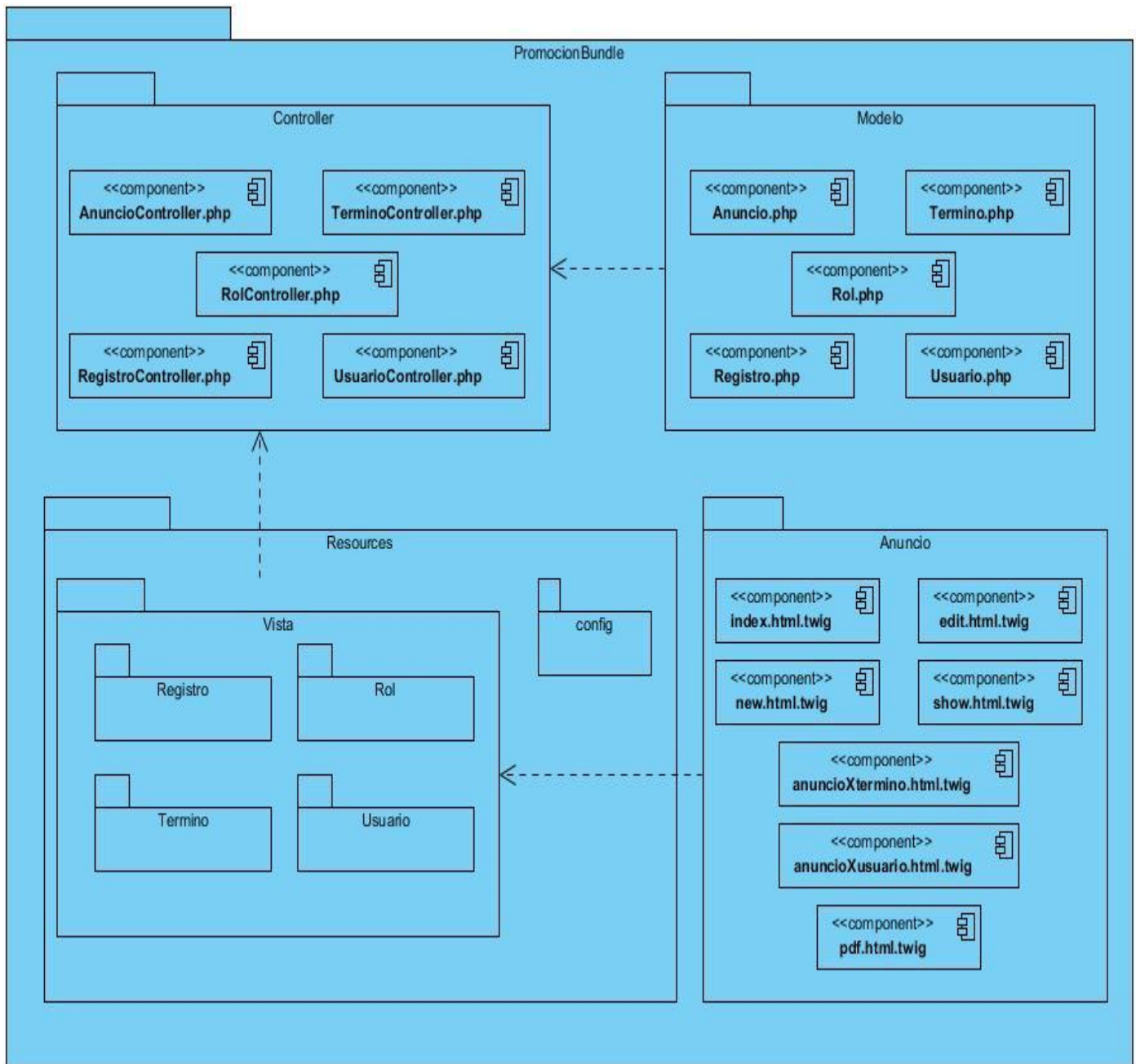


Imagen 8. Diagrama de componentes de PromocionBundle.

3.1.2. Diagrama de despliegue

El diagrama de despliegue se utiliza para modelar el hardware manejado en la implementación del sistema y sus relaciones [64]. Está compuesto por nodos, dispositivos y conectores. Describe la topología del sistema, la estructura de los elementos de *hardware* y *software*. Su propósito es capturar la configuración de los elementos de procesamiento y las conexiones entre estos elementos

en el sistema. Los nodos son adheridos por asociaciones de comunicación tales como enlaces de red, conexiones TCP.

A continuación se muestra el diagrama de despliegue del subsistema propuesto en este trabajo.

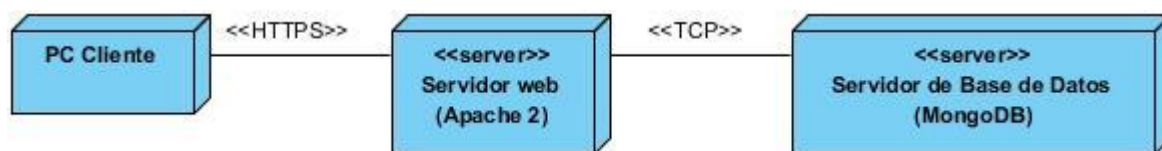


Imagen 9. Diagrama de despliegue.

PC Cliente: representa las computadoras clientes que se conectan al servidor de aplicaciones, las mismas se comunican con el servidor a través del protocolo seguro HTTPS.

Servidor web: representa el servidor donde se encuentra instalada la aplicación web. Este accede al servidor de base de datos para el manejo de la información mediante el protocolo TCP.

Servidor de Base de datos: es donde se almacena toda la información de la aplicación.

3.1.3. Estándares de código

En la fase de implementación se debe tener en cuenta la definición de un estándar de codificación de los programas. Establecer estándares para la programación de un sistema es tan significativo como cualquiera de los diagramas vistos anteriormente. Este estándar se utiliza para tener todo el código organizado y entendible para otros que no han trabajado en él. Si su implementación involucra a un grupo de desarrolladores, ayuda a lograr un código sin irregularidades y problemas de integración. Esto asegura que el código sea de alta calidad, que contenga una cantidad baja de errores y sea fácil de mantener.

A continuación se muestran algunos de estos estándares para el lenguaje PHP utilizados en *Symfony2*.

Clases y métodos

CamelCase: es una forma de notación de texto que sigue el patrón de palabras en minúscula sin espacios y con la primera letra de cada palabra en mayúsculas exceptuando la primera palabra. Este estándar fue utilizado para la definición de clases y métodos en el código de la aplicación.

Existen dos tipos de *CamelCase*:

- *UpperCamelCase*, cuando la primera letra de cada una de las palabras es mayúscula. Ejemplo: *EjemploDeUpperCamelCase*.
- *lowerCamelCase*, igual que la anterior con la excepción de que la primera letra es minúscula. Ejemplo: *ejemploDeLowerCamelCase* [65].

En la aplicación se utilizó *UpperCamelCase* para nombrar las clases existentes y *LowerCamelCase* para la nomenclatura de los métodos.

```
14  */
15  class AnuncioController extends Controller
16  {
17      /** Lists all Anuncio documents. ...*/
22      public function indexAction(Request $request){...}
43
44      /** Displays a form to create a new Anuncio document. ...*/
49      public function newAction(){...}
68
69      /** Creates a new Anuncio document. ...*/
76      public function createAction(Request $request){...}
106
```

Imagen 10. Ejemplo de uso de CamelCase.

Estructura

En la estructura se definieron los siguientes estándares:

- Las líneas podrían tener 80 caracteres o menos evitando tener más de 120 caracteres.
- Las llaves de apertura deben ir en la siguiente línea y la llave de cierre debe ir en la siguiente línea después del cuerpo.
- Las llaves de apertura en las estructuras de control debe ir en la misma línea y las llaves de cierre deben de ir después del cuerpo.
- Los paréntesis en las estructuras de control no deben usar espacios antes o después.
- Añadir un solo espacio después de cada limitador de coma.
- Añadir un solo espacio alrededor de los operadores (==, &&, ...).
- Usa llaves para indicar el control de la estructura sin tener en cuenta el número de declaraciones que el grupo pueda contener.
- Define una clase por fichero.

- Declara las propiedades de clase antes que los propios métodos de clase. (Garay, 2012)

3.1.4. Pantallas principales de la aplicación



Imagen 11. Portada del Subsistema de promoción de anuncios de ORIÓN.

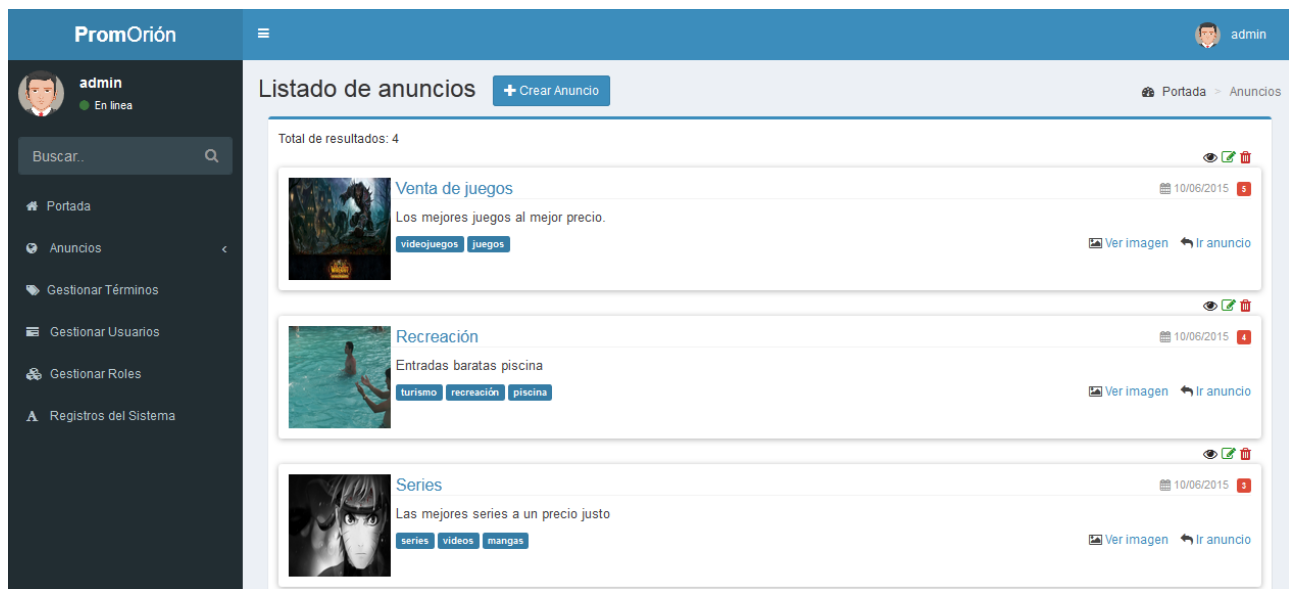


Imagen 12. Lista de anuncios del Subsistema de promoción de anuncios de ORIÓN.

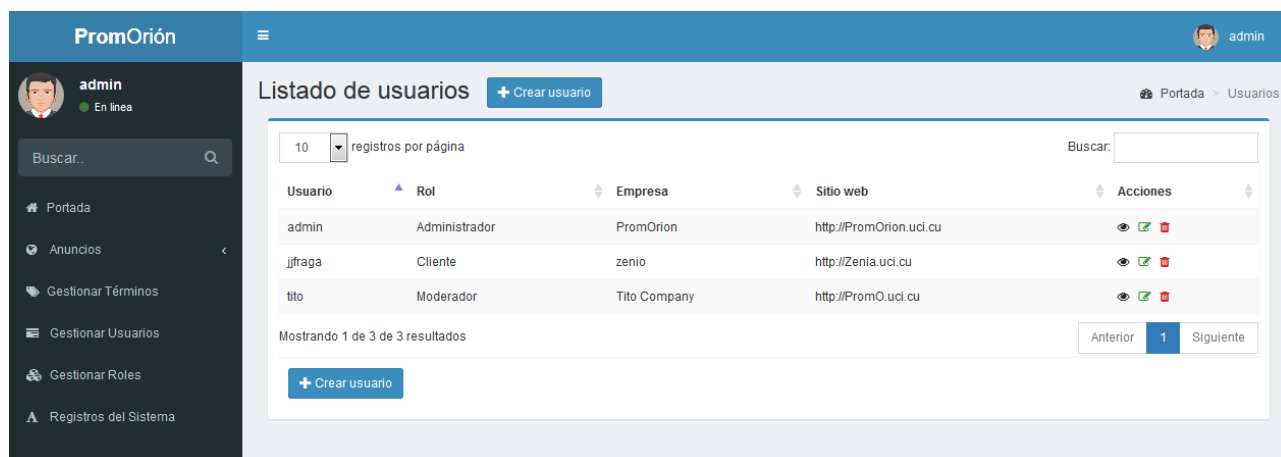


Imagen 13. Lista de usuarios del Subsistema de promoción de anuncios de ORIÓN.

3.2. Pruebas del sistema

Las pruebas que se le realizan a los sistemas ayudan a verificar el correcto funcionamiento de estos, su estado actual y los posibles errores que pueden detectarse. Estas pruebas, elevan la calidad de los mismos. Para la validación del subsistema se seleccionó el método de pruebas de caja negra.

3.2.1. Pruebas de caja negra

Las pruebas de caja negra fijan su atención en la validación de las funciones, métodos, servicios y casos de uso. Durante la aplicación de esta técnica se analiza cada funcionalidad implementada para verificar que se cumplan todos los requisitos establecidos. Los objetivos de este tipo de pruebas contemplan la navegación, entrada de datos, procesamiento y obtención de resultados, enfocándose en los requisitos funcionales y casos de uso.

Las pruebas de caja negra que se realizarán al subsistema estarán enfocadas o dirigidas a los casos de uso del sistema para verificar su correcto funcionamiento. En este tipo de pruebas se ejecutarán funcionalidades con datos correctos e incorrectos. En caso de que los datos sean incorrectos se verificará que los mensajes de error sean los deseados y en el caso opuesto que los resultados sean los esperados [66].

Dichas pruebas se realizan a partir de una matriz de datos donde:

V: indica válido.

I: indica inválido.

NA: indica que no es necesario proporcionar un valor del dato, ya que es irrelevante.

En la siguiente tabla se muestran las variables V1, V2, V3, V4, V5, V6 que representan valores de entrada de datos para los casos de prueba aplicados al caso de uso Gestionar anuncio.

Tabla 4. Descripción de las variables correspondientes al caso de prueba para el caso de uso Gestionar anuncio.

No	Nombre del campo	Clasificación	Valor nulo	Descripción
V1	Nombre	Campo de texto	no	Campo que admite caracteres de cualquier tipo. Inicialmente contiene el nombre del anuncio, aunque puede ser modificado.
V2	Descripción	Campo de texto	no	Campo que contiene la descripción de un anuncio, admite caracteres de cualquier tipo y puede ser modificado.
V3	URL del anuncio	URL	no	Campo que contiene la URL de la página donde se oferta el producto que se anuncia, admite caracteres válidos que cumplen con la estructura de una dirección electrónica y puede ser modificado.
V4	Término	Campo de selección	no	Campo que contiene una lista de objetos de tipo Término, puede modificarse.
V5	Precio del anuncio	Campo de rango de selección	no	Campo que contiene el valor del precio de un anuncio, se puede modificar.
V6	Imagen de Promoción	Campo para subir ficheros	si	Campo que contiene la imagen que representa el producto que se anuncia, se puede modificar.

A continuación se presentan los casos de prueba correspondientes al caso de uso “Gestionar anuncio”. El resto de los casos de prueba pueden ser consultados en el **Anexo**.

Caso de prueba para el caso de uso “Gestionar anuncio”

Tabla 5. Caso de prueba para el caso de uso Gestionar anuncio: Sección 1: “Insertar anuncio”.

Sección 1: “Insertar anuncio”									
Escenario	Descripción	V1	V2	V3	V4	V5	V6	Respuesta del sistema	Flujo central
EC 1.1	Se insertan todos los datos del	V	V	V	V	V	V	Inserta un nuevo anuncio. Muestra la lista de los anuncios creados	1. Se selecciona la opción “Crear Anuncio” en la

anuncio correctamente.	anuncio correctamente							y un mensaje indicando que el anuncio fue creado satisfactoriamente.	página principal. 2. Se introducen los datos. 3. Se selecciona la opción "Crear".
EC 1.2 Dejar campos vacíos.	Se intenta insertar un anuncio dejando campos vacíos.	I	I	I	I	I	I	NA Señala los campos que aún estén vacíos, para que se llenen. Muestra un mensaje indicando que el campo no puede estar vacío.	1. Se selecciona la opción "Crear Anuncio" en la página principal. 2. Se introducen los datos. 3. Se selecciona la opción "Crear".
EC 1.3 Insertar datos inválidos.	Se intenta insertar un anuncio con datos incorrectos.	I	I	I	I	I	I	Señala los campos que tengan datos inválidos. Muestra un mensaje indicando que el dato entrado es incorrecto.	1. Se selecciona la opción "Crear Anuncio" en la página principal. 2. Se introducen los datos. 3. Se selecciona la opción "Crear".

Tabla 6. Caso de prueba para el caso de uso Gestionar anuncio: Sección 2: "Editar anuncio".

Sección 2: "Editar anuncio"									
Escenario	Descripción	V1	V2	V3	V4	V5	V6	Respuesta del sistema	Flujo central
EC 2.1 Insertar datos editados de un anuncio correctamente.	Se guardan correctamente los cambios realizados.	V	V	V	V	V	V	Se guarda el anuncio con los nuevos datos editados. Muestra un mensaje indicando que el anuncio fue guardado satisfactoriamente.	1. Se selecciona la opción "Editar" del anuncio. 2. Se editan los datos. 3. Se selecciona la opción "Editar".
EC 2.2 Dejar campos vacíos.	Se intenta editar un anuncio dejando campos vacíos.	I	I	I	I	I	NA	Señala los campos que aún estén vacíos, para que se llenen. Muestra un mensaje indicando que el campo no puede estar vacío.	1. Se selecciona la opción "Editar" del anuncio. 2. Se editan los datos. 3. Se selecciona la opción "Editar". 4. Se muestran mensajes de error.
EC 2.3	Se intenta	I	I	I	I	I	I	Señala los campos que	1. Se selecciona la

Editar con datos inválidos.	editar un anuncio con datos incorrectos.						tengan datos inválidos. Muestra un mensaje indicando que el dato entrado es incorrecto.	opción "Editar" del anuncio. 2. Se editan los datos. 3. Se selecciona la opción "Editar". 4. Se muestran mensajes de error.
-----------------------------	--	--	--	--	--	--	---	--

Tabla 7. Caso de prueba para el caso de uso Gestionar anuncio: Sección 3: "Eliminar anuncio".

Sección 3: "Eliminar anuncio"			
Escenario	Descripción	Respuesta del sistema	Flujo central
EC 3.1 Eliminar correctamente un anuncio.	Se elimina un anuncio del sistema.	Elimina un anuncio de la base de datos del sistema y muestra el listado de los anuncios restantes. Se muestra un mensaje comunicando que el anuncio ha sido eliminado satisfactoriamente.	1. Se selecciona la opción "Eliminar" del anuncio. 2. Confirmar la acción presionando en el botón Sí. 3. Se muestra un mensaje comunicando que el anuncio ha sido eliminado satisfactoriamente.

Tabla 8. Caso de prueba para el caso de uso Gestionar anuncio: Sección "Buscar anuncio".

Sección 4: "Buscar anuncio"				
Escenario	Descripción	V1	Respuesta del sistema	Flujo central
EC 4.1 Realizar satisfactoriamente la búsqueda.	Se realiza la búsqueda luego de haber insertado correctamente los criterios y sus valores.	V	Se muestran los anuncios que concuerden con los criterios introducidos. Muestra un mensaje que comunica la cantidad de resultados encontrados.	1. Se introduce el nombre del anuncio en el formulario Buscar".
EC 4.2 La búsqueda no devuelve resultado.	Se realiza la búsqueda luego de haber insertado correctamente los criterios y sus valores.	V	Muestra un mensaje informado que no se obtuvo resultados para los criterios	1. Se introduce el nombre del anuncio en el formulario Buscar". 2. Se muestran mensajes

			insertados.	de informando que no se encontró información.
--	--	--	-------------	---

Tabla 9. Caso de prueba para el caso de uso Gestionar anuncio: Sección “Listar anuncio”.

Sección 5: “Listar anuncio”			
Escenario	Descripción	Respuesta del sistema	Flujo central
EC 5.1 Listar anuncios correctamente.	Se listan los anuncios del sistema.	Muestra el listado de los anuncios existentes.	1. Se selecciona la opción “Lista de anuncio”.

Para validar el correcto funcionamiento de la aplicación fueron realizadas tres iteraciones de pruebas. En la siguiente tabla se muestra un resumen general de todas las dificultades encontradas por iteraciones en las pruebas realizadas, donde:

PD: Pendiente

RA: Resuelta

Tabla 10. Resumen de los resultados de las pruebas aplicadas.

Fecha	Versión	Caso de prueba	Cantidad de no conformidades	Cantidad de no conformidades PD	Cantidad de no conformidades RA
18/5/2015	1.0	CU: Gestionar anuncio	8	-	8
20/5/2015	1.1	CU: Gestionar anuncio CU: Gestionar usuario	4 10	- -	4 10
21/5/2015	1.2	CU: Gestionar anuncio CU: Gestionar usuario	0 3	- -	0 3
27/5/2015	1.3	CU: Gestionar usuario CU: Gestionar rol	0 9	- -	0 9
28/5/2015	1.4	CU: Gestionar rol CU: Gestionar término	2 7	- -	2 7
30/5/2015	1.5	CU: Gestionar rol CU: Gestionar término	0 3	- -	0 3
31/5/2015	1.6	CU: Gestionar término	0	-	0

No conformidades

Durante la fase de prueba se realizaron seis iteraciones, las mismas arrojaron los siguientes resultados:

Primera Iteración: se encontraron 8 No conformidades de las cuales 5 eran de ortografía, 3 de validación.

Segunda Iteración: se encontraron 14 No conformidades de las cuales 7 eran de validación, 2 de funcionalidad y 5 de ortografía.

Tercera Iteración: se encontraron 9 No conformidades de las cuales 1 eran de validación, 3 de funcionalidad y 6 de ortografía.

Cuarta Iteración: se encontraron 9 No conformidades de las cuales 2 de funcionalidad y 7 de ortografía.

Quinta Iteración: se encontraron 3 No conformidades las cuales eran de funcionalidad.

Sexta Iteración: no se encontraron No conformidades.

Los otros casos de pruebas pueden observarse en los **Anexos**.

3.2.2. Pruebas de rendimiento

Las pruebas de rendimiento se realizan para determinar qué tan rápido un sistema realiza una tarea. Para llevarlas a cabo se utilizan herramientas que permiten monitorear y obtener información sobre el rendimiento del sistema según la ejecución de determinadas funcionalidades. La realización de estas pruebas está enfocada en determinar la capacidad del sistema para recibir múltiples peticiones sin que se vea afectado, así como la velocidad de respuesta del mismo.

Se decidió utilizar *Jmeter* en su versión 2.3.1, que es una herramienta de *software* libre, la cual permite medir la capacidad de carga de una aplicación. Además, con ella es posible conocer los tiempos de respuesta de un sistema dado un número de usuarios determinados y un número real de transacciones procesadas por unidad de tiempo [67].

Se realizaron pruebas de carga y estrés para comprobar el rendimiento del sistema soportando una cantidad máxima de usuarios interactuando con él y su comportamiento al aumentar esta carga con los mismos recursos disponibles.

El entorno de trabajo en el que fueron realizadas las pruebas de rendimiento cumple con las siguientes características:

Características del servidor 4GB RAM y 750GB de disco duro.

Características de red 100 Mbps.

En el diseño del plan de pruebas de rendimiento para el sistema se tuvo en cuenta las acciones que el usuario puede realizar al conectarse a la aplicación. En una muestra de 100 usuarios conectados concurrentemente, con un período de subida de 1 segundo (tiempo de espera de cada usuario para realizar una petición) la herramienta *JMeter* generó los siguientes reportes.

Label	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
/promocion/web/login	100	25047	25384	26304	20034	27224	0,00%	1,2/sec	10,1
/promocion/web/bundles/promocion/css/boot...	100	209	197	356	27	407	0,00%	1,5/sec	172,2
/promocion/web/bundles/promocion/css/skin...	100	30	17	71	2	189	0,00%	1,5/sec	5,1
/promocion/web/bundles/promocion/font-awe...	100	59	56	107	4	221	0,00%	1,5/sec	32,5
/promocion/web/bundles/promocion/css/site.c...	100	139	130	246	13	321	0,00%	1,5/sec	138,2
/promocion/web/bundles/promocion/css/style...	100	31	20	62	1	328	0,00%	1,5/sec	4,7
/promocion/web/bundles/promocion/plugins/i...	100	60	54	122	2	324	0,00%	1,5/sec	2,4
/promocion/web/bundles/promocion/js/bootstr...	100	114	109	200	8	310	0,00%	1,5/sec	52,2
/promocion/web/bundles/promocion/plugins/i...	100	48	27	132	2	247	0,00%	1,5/sec	6,7
/promocion/web/bundles/promocion/plugins/j...	100	147	138	254	17	314	0,00%	1,5/sec	123,9
/promocion/web/bundles/promocion/fonts/glyp...	100	44	32	97	3	219	0,00%	1,5/sec	34,5
/promocion/web/bundles/promocion/font-awe...	100	140	135	237	9	308	0,00%	1,5/sec	96,2
/promocion/web/bundles/promocion/plugins/i...	100	43	25	129	2	323	0,00%	1,5/sec	3,2
/promocion/web/_wdt/ca9ee1	100	24078	24222	25068	20116	26072	0,00%	1,1/sec	32,0
/promocion/web/check	100	75046	75298	77218	70799	78177	0,00%	42,4/min	6,9
/promocion/web/uploads/images/anuncios/20...	100	294	273	474	40	807	0,00%	1,5/sec	242,2
/promocion/web/uploads/images/anuncios/20...	100	241	211	368	53	1020	0,00%	1,5/sec	618,7
/promocion/web/uploads/images/anuncios/20...	100	127	128	215	14	406	0,00%	1,5/sec	163,5
/promocion/web/_wdt/a61e54	100	24525	24696	26297	20321	26911	0,00%	1,1/sec	31,0
/promocion/web/uploads/images/anuncios/	200	137543	136422	144597	125477	149695	100,00%	14,2/min	24,2
/promocion/web/anuncio/	200	48422	48422	49960	43570	52423	0,00%	44,3/min	7,1
/promocion/web/_wdt/7dd060	100	24061	24173	25212	21670	25991	0,00%	1,0/sec	28,9
/promocion/web/anuncio/5577bf5a87af03c91...	100	48358	48616	49683	44740	50567	0,00%	47,9/min	7,6
/promocion/web/_wdt/834392	100	23913	24042	25016	20580	25931	0,00%	59,4/min	28,5
/promocion/web/_wdt/60eeb9	100	23848	23958	24893	18511	25370	0,00%	58,8/min	28,3
/promocion/web/usuario/	200	48042	48164	49549	43727	50962	0,00%	44,7/min	7,1
/promocion/web/_wdt/7d0902	100	24110	24338	25240	21297	26126	0,00%	58,9/min	28,3
/promocion/web/usuario/5570684987af03030...	100	48786	49023	50402	44278	52049	0,00%	47,2/min	7,5
/promocion/web/_wdt/707383	100	23882	23854	24787	20535	25243	0,00%	59,2/min	28,4
/promocion/web/_wdt/405ab5	100	24091	24320	25092	20066	25744	0,00%	59,3/min	28,5
/promocion/web/_wdt/fe5733	100	24554	24711	25751	20923	27114	0,00%	57,5/min	27,6
/downloads?client=navclient-auto-ffox&appver...	100	2673	147	10575	0	14255	100,00%	1,5/sec	1,9
TOTAL	3800	27198	23708	50899	0	149695	7,89%	3,6/sec	164,9

Imagen 14. Reporte generado por el *JMeter* como resultado de las pruebas al sistema.

Como se puede observar en el análisis del resumen arrojado por la herramienta *JMeter* en la primera iteración, para un total de 100 usuarios con una subida por segundo se obtuvo 3800 muestras que se le realizaron al sistema se alcanzó un rendimiento de 3.6 peticiones por segundo, con un porcentaje de 7.89% de errores para el total de peticiones realizadas, lográndose un resultado positivo que cumple con los requisitos funcionales de rendimiento.

3.2.3. Pruebas de seguridad

Las pruebas de seguridad permiten validar que los datos y el sistema solamente sean accedidos por los usuarios autorizados y según sus roles. De esta forma se logra aumentar el control de la información sensible del sistema [68].

Autorización: en esta sección se asegura que cada usuario solo puede realizar las operaciones que le corresponden garantizando la integridad, confidencialidad y disponibilidad de la información. En el cumplimiento de las pruebas pertenecientes a esta sección se comprobó que los usuarios autenticados como moderadores no tenían privilegios necesarios para editar los anuncios. Este error fue solucionado.

Gestión de sesiones: las pruebas de esta sección consisten en chequear si el sistema permite acceder a las funcionalidades luego de haber cerrado la sesión garantizando la integridad. Dichas pruebas arrojaron que la aplicación no poseía este problema.

Autenticación: estas pruebas evalúan si el sistema muestra vulnerabilidades en el proceso de autenticación de usuarios a un posible atacante, mostrando mensajes que le permitan definir información sensible garantizando la integridad, confidencialidad y disponibilidad. En el cumplimiento de estas pruebas, se comprobó que el sistema no brindaba información adicional.

Validación de datos: en esta sección se encuentran las pruebas que garantizan la entrada de datos correcta, cumpliendo con los requisitos que pueda tener el sistema. En el cumplimiento de estas pruebas, se comprobó que el subsistema valida los datos de entrada para la url, el correo y la subida de imágenes.

Conclusiones parciales

En este capítulo se describe la estructura, el funcionamiento y las diferentes vistas que definen la arquitectura del subsistema propuesto, con el objetivo de lograr una mejor implementación y obtener una aplicación que cumpla con las funcionalidades establecidas, además se documentan las pruebas realizadas al sistema para validar su correcto funcionamiento, alcanzándose los siguientes resultados:

- Se estructuraron las clases del diseño en paquetes de componentes mostrando la organización y las dependencias entre los componentes que conforman el subsistema.
- Se realizó el diagrama de despliegue que propició un enfoque de la distribución física del subsistema.

- Se realizaron pruebas para validar que los requisitos fueron implementados correctamente a través de las pruebas de Caja Negra, permitiendo la identificación de no conformidades que fueron corregidas posteriormente.
- Para medir el rendimiento del subsistema se realizaron pruebas de Carga y Estrés mediante la herramienta *JMeter*, lo que propició observar el comportamiento de la aplicación bajo la cantidad de peticiones esperadas, arrojando resultados positivos.

CONCLUSIONES GENERALES

El presente trabajo de diploma da cumplimiento a las tareas de investigación propuestas inicialmente. Se logró que el motor de búsqueda ORIÓN pueda brindar servicios de promoción de anuncios relacionados con las necesidades de búsqueda de los usuarios, para lograr la implementación de un modelo de negocio que permita la sostenibilidad económica de este buscador. De esta manera se puede concluir que:

- Se valoró el estado del arte de los sistemas de promoción de anuncios en subsistemas de motores de búsqueda, garantizando un mejor entendimiento sobre el funcionamiento de los procesos y permitiendo identificar funcionalidades que sirvieron de guía para dar solución al problema planteado.
- Se obtuvo un conjunto de tecnologías y herramientas identificadas en el análisis de propuestas existentes y que posteriormente fueron utilizadas en el desarrollo del subsistema implementado, obteniéndose un buen funcionamiento del mismo.
- La metodología utilizada, permitió guiar todo el proceso de desarrollo del *software*, generando los principales artefactos que permitieron obtener una arquitectura sólida para el sistema.
- Se diseñó e implementó el conjunto de clases que dan cumplimiento a los requisitos funcionales y no funcionales asociados a los casos de uso de la solución propuesta.
- Las pruebas realizadas facilitaron la validación de la especificación de casos de uso permitiendo identificar no conformidades y su posterior solución, una mejor calidad del producto obtenido.
- El subsistema desarrollado brindará al usuario la posibilidad de gestionar anuncios de sus productos y promocionar estos a través del motor de búsqueda ORIÓN por lo cual tendrá que pagar un precio en consecuencia a la cantidad de visitas que realicen los usuarios del buscador a sus anuncios. Esto permitirá a ORIÓN obtener ingresos a cambio de sus servicios con el fin de financiarse contrarrestando los gastos de consumos de recursos debido a su desarrollo y mantenimiento con el objetivo de independizarse como empresa.

RECOMENDACIONES

Los resultados obtenidos, luego del desarrollo de la presente investigación satisfacen los requisitos definidos, no obstante se recomienda:

- Realizar las pruebas de integración y la documentación asociada a estas pruebas.
- Implementar un mecanismo de anuncios clasificados o pago directo.
- Utilizar la *cookies* para clasificar y mostrar los anuncios.

REFERENCIAS BIBLIOGRÁFICAS

1. Marketing en el Siglo XXI. 5ª Edición. *marketing-xxi.com* [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.marketing-xxi.com/Marketing-siglo-xxi.html>
2. DEFINICIÓN DE PROMOCIÓN - Promonegocios.net. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.promonegocios.net/mercadotecnia/promocion-definicion-concepto.html>
3. ÁLVAREZ HERNÁNDEZ, Nicolás. Sistematización de un modelo de gestión publicitaria para Google Adwords en las redes Google Search y Display. [En línea]. 2013. [Consulta: 16 junio 2015]. Disponible en: <http://repository.javeriana.edu.co/handle/10554/11240>
4. Los ingresos de internet de Google suben 19%. *Portafolio.com.co* [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.portafolio.co/negocios/google-ingresos-internet>
5. Lineamientos de la política económica y social del Partido y la Revolución - EcuRed. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://www.ecured.cu/index.php/Lineamientos_de_la_pol%C3%ADtica_econ%C3%B3mica_y_social_del_Partido_y_la_Revoluci%C3%B3n
6. Diccionario de la lengua española. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://lema.rae.es/drae/?val=promoci%C3%B3n>
7. FARBER, Patricio Bonta Mario. *199 Preguntas sobre marketing*. Editorial Norma, [Sin fecha]. ISBN 9789580470304.
8. TOWNSLEY, Maria. *Publicidad*. Thomson, 2003. ISBN 9789706863454.
9. UCEDA, Mariola García. *Las claves de la publicidad*. ESIC Editorial, 2011. ISBN 9788473568012.
10. Diccionario de la lengua española. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://lema.rae.es/drae/>
11. anuncio. *The Free Dictionary* [En línea]. [Sin fecha]. Disponible en: <http://es.thefreedictionary.com/anuncio>
12. Marketing en redes sociales, social media para PYMES | Emprende Pyme. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.emprendepyme.net/marketing-en-redes-sociales-social-media-para-pymes.html>
13. Promoción Económica Alcázar de San Juan. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.promocioneconomicaalcazar.es/component/content/article.html?id=129>
14. La PUBLICIDAD EN INTERNET - PromoNegocios.net. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.promonegocios.net/mercadotecnia/publicidad-en-internet.html>
15. El profesional de la información. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.elprofesionaldelainformacion.com/contenidos/2009/enero/09.html>

Referencias bibliográficas

16. Cookies de DoubleClick - Ayuda de AdSense. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <https://support.google.com/adsense/answer/2839090?hl=es>
17. Tipos de cookies que utiliza Google – Privacidad y Condiciones – Google. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.google.com/intl/es/policies/technologies/types/>
18. Why AdSense? - AdSense Help. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <https://support.google.com/adsense/answer/9712?hl=en>
19. AdSense program policies - AdSense Help. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <https://support.google.com/adsense/answer/48182?hl=en>
20. Publicación de anuncios. *yahoo* [En línea]. Disponible en: <https://policies.yahoo.com/us/es/yahoo/privacy/topics/advertising/index.htm>
21. ¿Qué es Bing Ads? | Aprendizaje para. *Bing Ads* [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://advertise.bingads.microsoft.com/es-es/cl/236/formacion/que-es-bing-ads>
22. Bing Ads reinventan la forma de hacer publicidad online. *Microsoft Advertising* [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://advertising.microsoft.com/es/cl/4307/bing-ads-reinventan-la-forma-de-hacer-publicidad-online>
23. Arquitectura Orientada a Servicios - *informesoa.pdf*. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.clubdeinvestigacion.com/docs/informesoa.pdf>
24. Guía Breve de Servicios Web. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>
25. *Manual_de_desarrollo_web_con_Grails.pdf*. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://sunshine.prod.uci.cu/gridfs/sunshine/books/Manual_de_desarrollo_web_con_Grails.pdf
26. *5.PDF - Desarrollo_web.pdf*. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://www.uoc.edu/masters/softwarelibre/esp/materials/Desarrollo_web.pdf
27. DEITEL, Harvey M. y DEITEL, Paul J. *Java How to Program, 7th Edition*. 7 edition. Upper Saddle River, NJ : Prentice Hall, 2007. ISBN 9780132222204.
28. AUMAILLE, Benjamin. *J2EE: Desarrollo de aplicaciones Web*. Ediciones ENI, 2002. ISBN 9782746019126.
29. JSON. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.json.org/>
30. CASTILLO, P. A., BERNIER, J. L., ARENAS, M. G., MERELO, J. J. y GARCIA-SANCHEZ, P. SOAP vs REST: Comparing a master-slave GA implementation. *arXiv:1105.4978 [cs]* [En línea]. 25 mayo 2011. [Consulta: 16 junio 2015]. Disponible en: <http://arxiv.org/abs/1105.4978>
31. Metodología open up ágil y tradicional. [En línea]. 09:24:47 UTC. [Consulta: 16 junio 2015]. Disponible en: <http://es.slideshare.net/carmeloh2/metodologia-open-up-39321348>
32. JUAREZ, Efrain Alberto Olguin. Open UP: Metodologías Open UP. *Open UP* [En línea]. 5 septiembre 2013. [Consulta: 16 junio 2015]. Disponible en: <http://openupeaojmp.blogspot.com/2013/09/metodologia-open-up.html>

Referencias bibliográficas

33. `cidi_configuracion_de_la_metodologia_open_up.pdf`. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://www.somosjovenes.cu/sites/default/files/cidi_configuracion_de_la_metodologia_open_up.pdf
34. LETELIER, Patricio y PENADÉS, M^a Carmen. Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). *www.cyta.com.ar/ta0502/v5n2a1.htm* [En línea]. 15 abril 2006. [Consulta: 16 junio 2015]. Disponible en: <http://www.cyta.com.ar/ta0502/v5n2a1.htm>
35. `scrum_manager_gestion_de_proyectos.pdf`. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://sunshine.prod.uci.cu/gridfs/sunshine/books/scrum_manager_gestion_de_proyectos.pdf
36. PROGRAMACION EN CASTELLANO, S. L. Webs dinámicos con PHP. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://programacion.net/articulo/webs_dinamicos_con_php_62/2
37. PHP: ¿Qué puede hacer PHP? - Manual. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://php.net/manual/es/intro-whatcando.php>
38. `Python_para_todos.pdf`. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://dSPACE.universia.net/bitstream/2024/919/1/Python_para_todos.pdf
39. `Manual_de_referencia_Java_2.pdf`. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://sunshine.prod.uci.cu/gridfs/sunshine/books/Manual_de_referencia_Java_2.pdf
40. IntroduccióN Uml. [En línea]. 10:38:41 UTC. [Consulta: 16 junio 2015]. Disponible en: <http://www.slideshare.net/zamanthag/introduccin-uml>
41. Desarrollo web ágil con Symfony2 - desarrollo-agil-symfony-2.3.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://sunshine.prod.uci.cu/gridfs/sunshine/books/desarrollo-agil-symfony-2.3.pdf>
42. ¿Qué es Symfony? *symfony.es* [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://symfony.es/pagina/que-es-symfony/>
43. Jobeet - jobeet_1_3.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://sunshine.prod.uci.cu/gridfs/sunshine/books/jobeet_1_3.pdf
44. About. *Zend Framework* [En línea]. Disponible en: <http://framework.zend.com/about/>
45. `Kumbia_PHP_Framework.pdf`. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://sunshine.prod.uci.cu/gridfs/sunshine/books/Kumbia_PHP_Framework.pdf
46. Sistema GestióN De Bases De Datos. [En línea]. 17:23:31 UTC. [Consulta: 16 junio 2015]. Disponible en: <http://es.slideshare.net/hugofern/sistema-gestin-de-bases-de-datos>
47. Databases — UCOOpenCourseWare. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/Course_listing
48. Reinventando la gestión de datos. [En línea]. Disponible en: <https://www.mongodb.com/es>
49. CABRERA GONZÁLEZ, Lianet, GONZÁLEZ, Lianet Cabrera y TORRES, Enrique Roberto Pompa. Extensión de Visual Paradigm for UML para el desarrollo dirigido por modelos de aplicaciones

Referencias bibliográficas

- de gestión de información. *Serie Científica* [En línea]. 15 octubre 2012. Vol. 5, no. 10. Disponible en: <https://publicaciones.uci.cu/index.php/SC/article/view/1032>
50. IBM Software online catalog. [En línea]. 13 noviembre 2014. [Consulta: 16 junio 2015]. Disponible en: <http://www-01.ibm.com/software/info/app/ecatalog/index.html>
51. Que es NetBeans. *NetBeans* [En línea]. Disponible en: https://netbeans.org/index_es.html
52. PhpStorm_Cookbook.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://sunshine.prod.uci.cu/gridfs/sunshine/books/PhpStorm_Cookbook.pdf
53. La_Biblia_Server_Apache.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://sunshine.prod.uci.cu/gridfs/sunshine/books/La_Biblia_Server_Apache.pdf
54. Nginx. [En línea]. Disponible en: <http://wiki.nginx.org/Main>
55. UML y patrones _Libro_.doc - Craig_Larman_-_UML_y_Patrones.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: http://sunshine.prod.uci.cu/gridfs/sunshine/books/Craig_Larman_-_UML_y_Patrones.pdf
56. SOMMERVILLE, Ian y GALIPIENSO, María Isabel Alfonso. *Ingeniería del software*. Pearson Educación, 2005. ISBN 9788478290741.
57. PRESSMAN, Roger S. *Ingeniería del software: un enfoque práctico*. Mikel Angoar, 1997.
58. 1 - Guia Arquitectura v.2.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://prof.usb.ve/lmendoza/Documentos/PS-6116/Guia%20Arquitectura%20v.2.pdf>
59. 08-Patrones1.ppt - 08-Patrones.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.inf.utfsm.cl/~visconti/ili236/Documentos/08-Patrones.pdf>
60. 06Patrones - 06Patrones.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://siul02.si.ehu.es/~alfredo/iso/06Patrones.pdf>
61. Tema 1 Patrones Arquitectónicos - get.php. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.lsi.us.es/docencia/get.php?id=1130>
62. Desarrollo web ágil con Symfony2 - desarrollo-agil-symfony-2.1.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://sunshine.prod.uci.cu/gridfs/sunshine/books/desarrollo-agil-symfony-2.1.pdf>
63. Modelo de datos. [En línea]. 13:49:37 UTC. [Consulta: 16 junio 2015]. Disponible en: <http://es.slideshare.net/csalazarc/modelo-de-datos-14506949>
64. Diagramas De Despligue Uml. [En línea]. 10:48:48 UTC. [Consulta: 16 junio 2015]. Disponible en: <http://es.slideshare.net/arcangelsombra/diagramas-de-despligue-uml-1475353?related=1>
65. Estándar de Codificación. *Scribd* [En línea]. [Consulta: 16 junio 2015]. Disponible en: <https://es.scribd.com/doc/246920425/Estandar-de-Codificacion>
66. Técnicas Evaluación de Software. *Scribd* [En línea]. [Consulta: 16 junio 2015]. Disponible en: <https://es.scribd.com/doc/70661776/Tecnicas-Evaluacion-de-Software>

Referencias bibliográficas

67. JMeter. Manual de usuario v1.2 - JMeter. Manual de usuario v1.2.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: https://www.euskadi.eus/y79-04/es/contenidos/informacion/herramientas_ejie/es_0213/adjuntos/JMeter.%20Manual%20de%20usuario%20v1.2.pdf
68. Microsoft Word - RevistaNJICS09v2.doc - PozoVol5Num2.pdf. [En línea]. [Consulta: 16 junio 2015]. Disponible en: <http://www.ati.es/IMG/pdf/PozoVol5Num2.pdf>