



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD 1

SUBSISTEMA DE PUBLICIDAD APLICADO A LA
PLATAFORMA DE CONTENIDOS UNIFICADOS PARA
BÚSQUEDA AVANZADA

Trabajo de diploma para optar por el título de Ingeniero en Ciencias
Informáticas

Autor: Andy Vega Fajardo

Tutores:

MSc. Leiny Amel Pons Flores

Ing. Miguel Ángel Chávez Alfonso

La Habana, junio de 2019

“Año 61 de la Revolución”



Dedicatoria

Dedico este trabajo a mi madre María Yaquelin Fajardo por ser el motor impulsor de mi vida, mi mejor amiga, mi más fiel confidente y la que ha hecho posible todo lo logrado.



Declaración de autoría

Declaro por este medio que yo **Andy Vega Fajardo**, con carné de identidad **95071127204** soy el autor principal del trabajo titulado “**Subsistema de publicidad aplicado a la plataforma de contenidos unificado de búsqueda avanzada**” y autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Para que así conste firmamos la presente a los _____ días del mes de _____ de _____

Firma del autor:

Andy Vega Fajardo

Firma del tutor

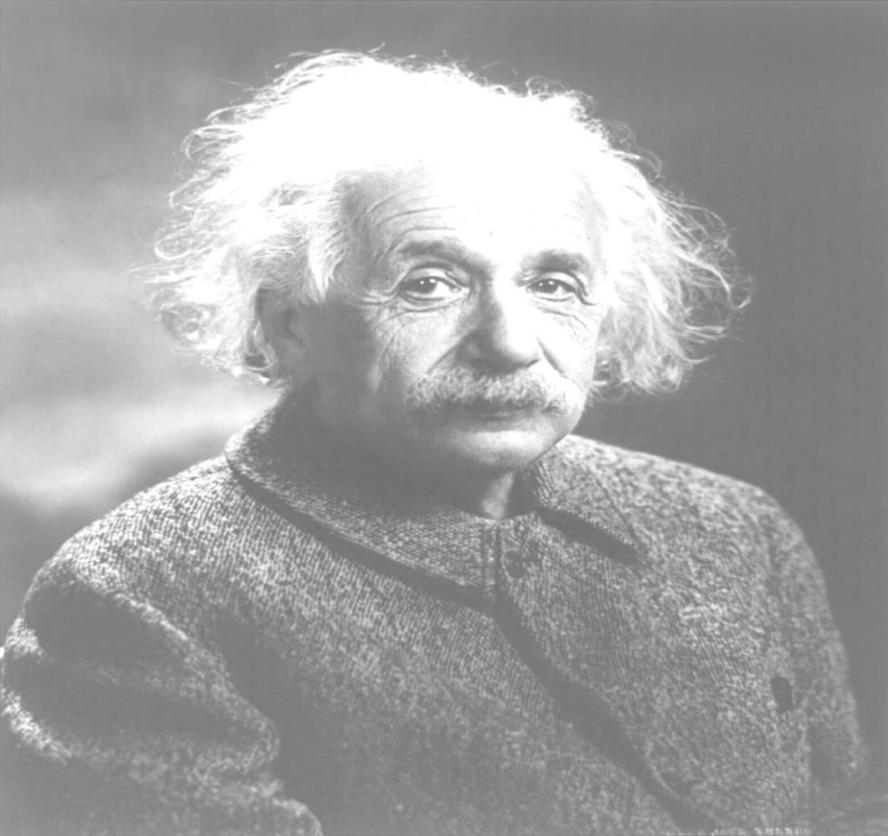
MSc. Leiny Amel Pons Flores

Firma del tutor

Ing. Miguel Ángel Chávez Alfonso



Fraser



“Hay una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad.”

-Albert Einstein-



Agradecimientos

A mis tutores, Leiny y Chávez, por todo su empeño, dedicación, exigencia y preocupación, sin los cuales nada hubiese sido posible, por apoyarme en mis decisiones y ayudarme sin importar la hora y el momento en los que se encontraran.

A mi madre por siempre confiar en mí y aportarme la mejor educación, por su preocupación, por ser mi compañía, mi ángel de la guarda y sobre todas las cosas por darme ese infinito amor que solo ella sabe dar.

A mi padre Andrés y a mi abuela Juaní, que aunque ahora están en el cielo, fueron motores indispensables para la realización de este día, por aportarme el cariño y la preocupación.

A mi segundo padre Roberto, por acogerme como su hijo, por su educación, sus consejos, sus experiencias, por enseñarme a valerme por mí mismo, a ser trabajador, dedicado, a luchar por la familia, puntos muy importantes para afrontar la vida.

A mis abuelos Arturo, Catalina y Dalina, a mis tíos Odel y Yuliet y en general a toda la familia por su apoyo, cariño y preocupación cada vez que estaba en casa.

A mi novia, por su dedicación, por aguantarme mis resabios y días de estrés, por su cariño, su entrega y por haberme acompañado estos 5 años de estudio, afrontando junto conmigo todos los peñales que en la universidad hay que escalar.

A mis suegros por haberme abierto las puertas de su casa, darme esa comida tan rica que embruja y haberme acogido como otro miembro más de su familia.



A mis amigos, Reimer, Vladimír, Luis Ernesto, Pedro Javier, Bryan, Leo, Magdiel, Anthuan, Rafael, Iván Javier, Adrián, Alfred, Susana por compartir momentos inolvidables, de los cuales me llevo muchos recuerdos. También a demás amistades que aunque no menciono, fueron importantes para mí y aportaron su granito de arena en mi estadía en la Universidad.

A todas aquellas personas que de una forma u otra aportaron a mi formación como persona y profesional.

GARCIAS A TODOS.



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. La Plataforma C.U.B.A., desarrollada en la Universidad de las Ciencias Informáticas, no cuenta con un mecanismo que permita incrementar las visitas a los sitios de las empresas, ni permite a los usuarios publicar anuncios que le den publicidad a sus servicios, para solucionar este problema se propone el desarrollo de un subsistema de publicidad y propaganda que permitirá contribuir a la publicidad de anuncios en la red 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.

Palabras claves: Plataforma C.U.B.A., publicidad, anuncios, sistemas publicitarios.



Índice

INTRODUCCIÓN.....	11
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA DEL SUBSISTEMA DE PUBLICIDAD APLICADO A LA PLATAFORMA C.U.B.A.....	16
Introducción.....	16
1.1 Definición del marco teórico conceptual de la investigación	16
1.1.1 Promoción.....	16
1.1.2 Publicidad	17
1.1.3 Otros conceptos relacionados.....	21
1.2 Estudio de los sistemas homólogos.....	21
1.2.1 Google Adwords o GoogleAds	22
1.2.2 BingAds	23
1.2.3 Yahoo Search Marketing.....	24
1.3. Selección del entorno de desarrollo para la construcción de la solución.....	26
1.3.1. Metodología de desarrollo de software.....	26
1.4 Lenguajes para la programación	27
1.4.1 Del lado del servidor	27
1.4.2 Del lado del cliente.....	27
1.4.3 <i>Framework</i> . Marco de trabajo	28
1.5 Capa de servicios.....	29
1.6 Herramientas.....	30
Conclusiones parciales del capítulo.....	31



CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL SUBSISTEMA DE PUBLICIDAD APLICADO A LA PLATAFORMA C.U.B.A.....	32
Introducción.....	32
2.1 Descripción de la propuesta de solución	32
2.2 Especificación de requisitos de software	32
2.2.1 Requisitos funcionales	32
2.2.2 Requisitos no funcionales	34
2.3 Historias de usuarios	35
2.4 Patrones de diseño.....	41
2.4.1 Patrones GRASP	41
2.4.2 Patrones GoF.....	42
2.5 Patrones arquitectónicos	43
2.5.1 Patrón arquitectónico Modelo – Vista – Controlador.....	43
2.6 Modelo de diseño	44
2.6.1 Diagrama de clases de diseño	44
2.7 Modelo de datos.....	45
2.8 Modelo de despliegue	46
Conclusiones parciales del capítulo.....	47
CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DEL SUBSISTEMA DE PUBLICIDAD APLICADO A LA PLATAFORMA C.U.B.A.....	48
Introducción.....	48
3.1 Diagramas de componentes	48
3.2 Estándares de codificación.....	49
3.3 Validación del Subsistema de Publicidad aplicado a la Plataforma C.U.B.A.....	51



3.3.1 Pruebas funcionales	51
3.3.2 Pruebas de carga y estrés	59
3.3.3 Pruebas de seguridad	60
3.3.4 Pruebas de usabilidad.....	61
Conclusiones parciales del capítulo.....	64
CONCLUSIONES	65
RECOMENDACIONES	66
BIBLIOGRAFÍA.....	67



INTRODUCCIÓN

El avance vertiginoso de las Tecnologías de la Información y las Comunicaciones (TIC) que se inició en la segunda mitad del pasado siglo, ha marcado al mundo actual y a una sociedad que se ha denominado sociedad de la información. Las TIC han facilitado que la humanidad posea fácil acceso a la información, comunicación inmediata sin importar la posición geográfica, mayores capacidades de almacenamiento de la información digital y nuevas formas de trabajos. Estas facilidades han propiciado que los volúmenes de información a nivel mundial se incrementen considerablemente. Cada minuto, los 2.700 millones de personas con acceso a Internet que se calcula hay actualmente en el mundo envían más de 200 millones de correos electrónicos, realizan 2 millones de consultas a Google, suben 48 horas de vídeo a YouTube, escriben más de 100.000 mensajes en Twitter, publican casi 30.000 nuevos artículos en sitios como Tumblr o WordPress y suben más de 6.000 fotografías a Instagram y Flickr (1).

Como consecuencia los sitios web han aumentado a gran escala existiendo actualmente más de 1500 millones de sitios (2), como se evidencia en la siguiente gráfica:

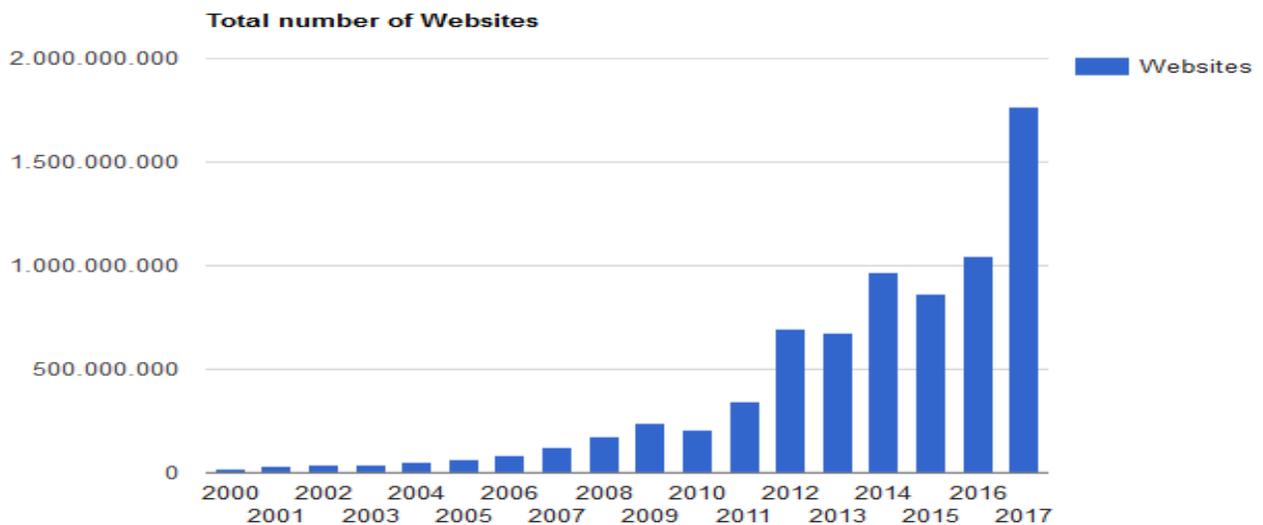


Figura 1. Cantidad de sitios web por año. Fuente: (Internet Live Stats, 2018)

En el ámbito informático existe la recuperación de información. Esta consiste en el proceso de búsqueda ejecutado por un usuario, en el cual encuentra contenidos de diferentes formatos según sus intereses y



necesidades (3). En muchos casos la información se encuentra dispersa y poco estructurada, lo que hace muy engorroso el proceso de recuperación de información útil en la red. Los Sistema de Recuperación de Información (SRI) aparecen con el propósito de manejar, la recuperación y el filtrado de la información contenida en la web. De manera general, se pueden distinguir tres tipos de herramientas para localizar la información: motores de búsqueda, directorios web y metabuscadores (4), estos son útiles para procesar grandes cantidades de documentos rápidamente.

La cantidad de datos en la red ha crecido tan rápido como la velocidad de las computadoras, y ahora utilizando los SRI somos capaces de buscar en colecciones de miles de millones de palabras contenidas en diferentes tipos de documentos. La publicidad de anuncios es uno de los principales servicios brindados por los SRI, las cuales son las principales fuentes de ingreso de estos sistemas.

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 (5) y en la actualidad alrededor del 97% de sus ingresos totales provienen de la publicidad (6).

En Cuba, como parte del continuo proceso de informatización, se han creado varios buscadores como Zoom, presentado en el contexto de la 17ma. Convención y Feria Informática 2018. Zoom fue desarrollado por especialistas de la Empresa Nacional de Software (Desoft). Su principal objetivo radicó en brindar una herramienta que indexara correctamente los sitios web cubanos. Su filosofía es permitir a los usuarios, mediante la creación de cuentas, acceso a servicios de blogs, cursos, una biblioteca virtual o consultar el estado del tiempo, etc. Dispone también de herramientas útiles para las empresas como son sus sistemas estadísticos y de posicionamiento web, vitales para la buena visibilidad de cualquier sitio (7).

Otro ejemplo de buscador es la Plataforma Contenidos Unificados de Búsqueda Avanzada (C.U.B.A.), la cual permite realizar búsquedas sobre los contenidos publicados en la web cubana. Puede ser accedido



desde la url: <https://www.redcuba.cu>. La Plataforma C.U.B.A. surge por la necesidad existente en el país de desplegar un sistema de recuperación de información que pudiera ser accedido desde cualquier parte del territorio. Posee un buscador basado en la tecnología Orión, además de otros servicios. La Plataforma C.U.B.A. tiene como objetivo, ser una fuente más de visibilidad en internet de los productos nacionales, por lo que debe promocionar los servicios ofertados en Cuba a disposición de los consumidores, elemento que aún no cumple.

Los buscadores internacionales en ocasiones no indexan los sitios web cubanos debido a que, en Cuba, un porcentaje de los sitios, portales u otros sistemas web no pueden ser accedidos desde internet, porque solo son de navegación nacional, por lo que las búsquedas realizadas que estén contenidas en alguno de ellos se ve obstaculizada. Esto no contribuye a la difusión de oportunidades económicas al estar vinculada a la publicidad digital que proponen los sitios, portales y otros sistemas web. A esta situación se suma el hecho de que en Cuba las formas más utilizadas de realizar publicidad son a través de materiales en formato duro y mediante los medios de comunicación, ejemplo de ellos son: Radio Taíno, Cubavision Internacional y Prensa Latina, que ofrecen el servicio de anuncio a organismos estatales, mientras que Ofertas y el tabloide impreso de la Agencia Cubana de Noticias es la oportunidad del sector no estatal (8).

Por otra parte, la Plataforma C.U.B.A. centra su principal función en la indexación de las páginas cubanas. Entre los servicios que ofrecen están, las portadas de publicidad, en las cuales se muestran de manera resumida los importantes eventos realizados en el país, también está el directorio temático, enfocado en dar a conocer los diferentes portales web de organizaciones, grupos empresariales, ministerios, turismo, entre otros que se encuentran operativos y que son de gran relevancia en el territorio. En lo anteriormente descrito se puede apreciar que el buscador trabaja de manera disímil la publicidad, pero no satisface las exigencias de los clientes de utilizar un mecanismo para publicar anuncios o realizar una campaña promocional, que aumente las visitas a sus sitios, incrementen las probabilidades de competir e influenciar en la cultura de las personas aunque tengan un pequeño o mediano negocio, además de utilizar el sistema para expandir su mercado, no solo en el territorio nacional, sino inmiscuirse a nivel internacional.

Tomando en consideración la problemática anteriormente descrita, se plantea como **problema de investigación**: ¿Cómo contribuir a la publicidad de anuncios nacionales a través de la Plataforma C.U.B.A. en la red cubana? Definiéndose como **objeto de estudio**: los procesos de publicidad en buscadores web; haciendo énfasis en el proceso de publicidad en la Plataforma C.U.B.A. como **campo de acción**.



Con el fin de darle una solución al problema de investigación se define como **objetivo general**: Desarrollar un subsistema de publicidad basado en tecnologías de desarrollo libre para contribuir a la publicidad de anuncios a través de la Plataforma C.U.B.A. en la red cubana.

En virtud de solucionar la problemática existente, el objetivo general ha sido desglosado en los siguientes **objetivos específicos**:

- 1) Construir el marco teórico conceptual.
- 2) Realizar el estado del arte respecto a las tecnologías y funcionalidades de los sistemas de publicidad relacionados a buscadores webs.
- 3) Desarrollar un subsistema que permita la gestión de anuncios publicitarios a través de la Plataforma C.U.B.A. en la red cubana.
- 4) Validar el correcto funcionamiento del subsistema desarrollado.

Luego de haber definido los objetivos primordiales se plantea la siguiente **idea a defender**: El desarrollo de un subsistema de publicidad aplicado a la Plataforma C.U.B.A. contribuirá a la difusión de anuncios en la red cubana.

Para lograr los objetivos descritos anteriormente se plantean las siguientes **tareas de investigación**:

- 1) Realización de un estudio sobre las tendencias de los sistemas de publicidad en Cuba.
- 2) Selección de las tecnologías, herramientas y estándares que se necesitan para implementar el subsistema de publicidad de la Plataforma C.U.B.A.
- 3) Selección de la metodología de desarrollo de software que guíe el desarrollo del subsistema de publicidad de la Plataforma C.U.B.A.
- 4) Elaboración de los artefactos requeridos por la metodología de desarrollo seleccionada.
- 5) Implementación del subsistema de publicidad de la Plataforma C.U.B.A.
- 6) Validación de la solución propuesta a partir de los métodos científicos definidos.

Métodos de Investigación utilizados:

Métodos teóricos:



Analítico-Sintético: permitió la recopilación y análisis de la información sobre los elementos esenciales referentes a las teorías, documentos y literatura en general relacionada con los sistemas de publicidad.

Histórico-Lógico: permitió conocer los antecedentes e importancia de cómo han evolucionado en el tiempo los sistemas de publicidad, así como las herramientas y tecnologías utilizadas en el desarrollo de aplicaciones web.

Métodos empíricos:

Modelación: utilizado en la representación, mediante el uso de diagramas, de las características del sistema, relaciones entre objetos; y las actividades que intervienen en los procesos implementados en el subsistema de publicidad aplicado a la Plataforma C.U.B.A.

Observación: Se hace un estudio minucioso de trabajos realizados que se encuentran estrechamente vinculados con la problemática a la que se hace referencia.

Entrevista: Se empleó para obtener información perteneciente al Centro de Ideoinformática (CIDI) mediante un modelo estructurado con preguntas predefinida.

Estructura del documento:

Capítulo 1. Fundamentación teórica del Subsistema de Publicidad aplicado a la Plataforma C.U.B.A.: se presentan un conjunto de conceptos asociados al objeto de estudio de la investigación. Se estudian los sistemas de promoción de diferentes Buscadores y se exponen las tecnologías y metodologías a utilizar para el desarrollo del software.

Capítulo 2. Análisis y diseño del Subsistema de Publicidad aplicado a la Plataforma C.U.B.A.: se exponen las características del sistema, patrones de diseño, requisitos funcionales y no funcionales, arquitectura utilizada y los artefactos que requiere la metodología de desarrollo utilizada.

Capítulo 3. Implementación y validación del Subsistema de Publicidad aplicado a la Plataforma C.U.B.A.: se presentan algunos aspectos asociados a la implementación de la solución informática, así como sus componentes. Además, se exhiben los diseños de casos de prueba a utilizar en la validación del sistema y se analizan los resultados arrojados por las pruebas realizadas.



CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA DEL SUBSISTEMA DE PUBLICIDAD APLICADO A LA PLATAFORMA C.U.B.A.

Introducción

En el presente capítulo se abordan los principales elementos que justifican y soportan teóricamente la publicidad en los buscadores. Además, se presentarán las tecnologías, herramientas y metodología que serán empleadas en la implementación de la propuesta de solución.

1.1 Definición del marco teórico conceptual de la investigación

Para lograr un mejor entendimiento del entorno de la investigación es necesario la identificación y el estudio de las principales bibliografías y así analizar algunos de los conceptos asociados al área de los sistemas de recuperación de información y su personalización.

1.1.1 Promoción

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" (9).

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" (10).

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" (11).

Teniendo en cuenta las anteriores definiciones, el autor de la presente investigación plantea la siguiente **definición de promoción**:

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



1.1.2 Publicidad

La publicidad es el conjunto de medios que se emplean para divulgar o extender la noticia de las cosas o de los hechos de acuerdo con el diccionario de la Real Academia Española (9). Sin embargo, de acuerdo con diferentes expertos en la materia, se define publicidad de la siguiente forma:

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*" (12).

Según Mariola Garcia Ulceda, 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" (13).

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 (14).

Teniendo en cuenta las anteriores definiciones, el autor propone la siguiente **definición de anuncio**:

El anuncio es dar a conocer y comunicar al público información sobre productos y servicios para persuadirlos y convencerlos de realizar su compra.

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



Fundamentación teórica

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 (15).

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* (16).

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.



Los elementos publicitarios que se pueden utilizar dentro de un sistema publicitario son los siguientes (17):

- **Banners:** Son imágenes rectangulares, ya sea horizontales o verticales, en formato JPG o GIF (para banners estáticos) y en Gif Animado o Flash (para banners animados). Por lo general, tienen tamaños *standard*.
- **Texto:** Por lo general, es una frase escrita que incluyen un *link* o enlace hacia una determinada página del sitio web que se desea dar a conocer. Se recomienda que ésta frase además de mencionar el nombre de la página web, incluya las palabras clave con las que se pretende lograr una buena posición en los buscadores y que se encuentran en mayor proporción en la página hacia la que apunta el enlace.
- **Firmas digitales:** Son textos que se incluyen al final de cada correo electrónico (que se vaya a enviar a alguien) para identificar, por ejemplo, al director del sitio web, su cargo, página web, correo electrónico y el producto o servicio que se pretende promocionar.
- **Pop Ups y Pop Unders:** Son pequeñas ventanas que se abren al momento de ingresar a una página web (*pop up*) o al salir de ella (*pop under*) y que sirven para colocar anuncios. Para publicarlos, se necesita la autorización del administrador o dueño del sitio web donde se quiere poner un *pop up* o *pop under* (por lo general, esto tiene un precio) y el código fuente para que ellos lo incluyan en las páginas web que se haya convenido.

Las principales características comunes de los sistemas de publicidad en buscadores son las siguientes (17):

- **Segmentación:** La publicidad en buscadores permite crear campañas dirigidas a determinados segmentos geográficos y de idioma. De la misma manera, la elección por parte del anunciante de las palabras clave que publicarán los anuncios facilita que éstos solamente se muestren para un público realmente interesado en unos servicios o productos concretos. Cuanto mejor segmentada esté una campaña, más posibilidades habrá de conseguir visitantes que puedan reportar beneficios a la empresa anunciante.
- **Relevancia:** Una campaña de enlaces patrocinados en buscadores tiene un alto componente de relevancia. Es decir, se muestra ante un público objetivo que, activamente, realiza una búsqueda de



Fundamentación teórica

información relacionada con los productos de dicha campaña. Las posibilidades de que el usuario se interese por las ofertas del anunciante son mucho mayores que en otras formas de publicidad.

- **Control del gasto:** La publicidad en buscadores permite supervisar y modificar el gasto de una campaña en todo momento. Los programas publicitarios de los principales buscadores incluyen paneles de control online en los que el anunciante puede consultar el rendimiento de sus anuncios y valorar si desea realizar cualquier tipo de ajuste en el presupuesto.
- **Flexibilidad:** Un anunciante puede modificar no sólo sus ajustes de gasto (como, por ejemplo, el presupuesto que destina diariamente a una campaña o el coste por clic designado para que una palabra clave específica muestre un anuncio), sino que también tiene la posibilidad de realizar cambios en los textos de sus anuncios o en sus palabras clave, pausar o suprimir una campaña y modificar los ajustes de segmentación, entre otros, en cualquier momento. La flexibilidad que ofrece la gestión de las campañas es uno de los principales atractivos de la publicidad de enlaces patrocinados en buscadores.
- **Resultados medibles:** Al contrario que otras formas de publicidad, gracias a las herramientas de medición de tráfico y resultados que existen en la actualidad es posible contar con información exacta sobre cómo llegan los usuarios hasta el sitio web y qué recorrido efectúan una vez dentro de él hasta finalizar su visita, bien habiendo realizado la compra final, o bien abandonando la página en un momento determinado. Además, estas herramientas realizan un seguimiento completo de las campañas de marketing online, gracias a las cuales el anunciante puede identificar las palabras clave y textos de anuncio que mayor número de visitas le reportan y hacer un balance de los beneficios conseguidos respecto a la inversión realizada. Los datos sobre retorno de inversión que facilitan estas herramientas de analítica web (como, por ejemplo, *Google Analytics*, integrada en el programa de publicidad de *Google AdWords*) ayudan a controlar el balance coste-efectividad de la campaña en todo momento.
- **Redes de contenido y formatos:** Los sistemas de publicidad *online* en buscadores ofrecen también la posibilidad de ampliar la visibilidad de las campañas orientándolas a un gran número de páginas web y sitios de búsqueda asociados a cada programa. Además, el anunciante podrá utilizar diversos formatos de anuncios.



1.1.3 Otros conceptos relacionados

- **Textos de anuncio:** Redactar textos de anuncio explicativos y concisos, con títulos atractivos, un lenguaje claro y que destaquen características especiales del producto anunciado (precios competitivos, ofertas, posibilidad de realizar compra online, atención postventa, etc.) aumenta las posibilidades de que alguien haga clic en el anuncio y visite la página web del anunciante.
- **Palabras clave:** Una búsqueda realizada con un término incluido en el listado de palabras clave asociado a un anuncio hará que este anuncio aparezca publicado en la página de resultados del buscador. Cuanto más orientadas estén las palabra clave al segmento de negocio del anunciante, mayor rendimiento obtendrán los anuncios. El número de palabras clave que se pueden asociar a los anuncios es ilimitado, pero no es recomendable incluir términos que no guarden relación con el negocio que se anuncia ya que éste aparecerá para búsquedas irrelevantes, lo que hará que el promedio de clics y, por tanto, la posición del anuncio, disminuyan.
- **Páginas de destino:** Cuando el usuario hace clic en un anuncio, se le dirige a la página de la empresa anunciante. Es primordial presentar una página de destino en la que el usuario encuentre lo que desea cuando inicia la búsqueda. Factores como el diseño y la facilidad de navegación por la página, la presentación atractiva de los productos, facilitar información sobre los mismos, como las opciones de pago, o incluir datos sobre la empresa y formas de contacto son muy importantes para conseguir una buena experiencia del usuario en el sitio.
- **Seguimiento y optimización según resultados:** Supervisar el rendimiento de los anuncios es muy sencillo, ya que se puede realizar visualizando los informes disponibles en el propio centro de control online de la cuenta. A través de dichos informes, es posible conocer el retorno de inversión que una campaña está obteniendo, con lo cual el anunciante tendrá elementos de juicio adecuados para optimizar tanto la campaña como la página web en función de estos resultados.

1.2 Estudio de los sistemas homólogos

En la actualidad existe en el mundo diversidad de buscadores que permiten la promoción de anuncios y la creación de campañas publicitarias, algunas de las más usadas son GoogleAds, BingAds y Yahoo Search Marketing.



1.2.1 Google Adwords o GoogleAds

Google Ads es la fuente principal de ingresos de Google y constituye un método de pago por publicidad dinámica para el cliente. Esto significa que el usuario pagará o cobrará por el tráfico generado en un sentido u otro. Los anunciantes, con el concepto de pago por clic, sólo pagarán por aquellos anuncios en los que se ha hecho clic. En la parte de los dueños de páginas web, éstos cobrarán en función del número de clics que los anuncios en su página web hayan generado (18).

Con la introducción de AdWords en 2000, Google renunció a su política inicial de mantenerse como un buscador completamente libre de publicidad. Hasta ese momento, la ausencia de publicidad quería ser un signo de la independencia de los resultados de una búsqueda, así como una garantía para la velocidad de carga de las páginas de Google. (19)

Google Ads puede activarse en las páginas de búsqueda de Google, así como también en los numerosos sitios web que pertenecen a la red de publicidad de Google. Existen programas de publicidad similares operados por otros buscadores (como por ejemplo Yahoo! Search Marketing o Bing Ads, de Microsoft) (19).

En julio de 2018, tras el cambio más importante en la interfaz de la plataforma, también deciden cambiar el nombre de estos servicios y pasa a llamarse Google Ads. Google Ads nace como una reacción del gigante tecnológico ante la inminente pérdida de mercado por parte de Facebook Ads (19).

Entre sus características se encuentran:

- Muestran sus anuncios sólo a los clientes potenciales que buscan los productos y servicios que ofrece el anunciante. De esta forma, los usuarios encuentran mensajes publicitarios relevantes para su búsqueda y los anunciantes publican sus mensajes cuando hay un público objetivo interesado.
- Segmentan fácilmente el público objetivo al que quieren dirigirse, con la posibilidad de acceder a clientes potenciales en todo el mundo y en su propio idioma.
- Pagan sólo por los clics recibidos a su sitio web. Es decir, los anunciantes no pagan por el número de veces que se muestran sus anuncios, sino por las visitas recibidas en su sitio web de usuarios atraídos por los anuncios e interesados en los productos y/o servicios promocionados.
- Invierten lo que consideran oportuno, según sus medios y objetivos, y durante el período de tiempo que deseen.



- Miden los resultados de la campaña prácticamente en tiempo real, como por ejemplo el número de visitas recibidas al sitio web o el número de ventas realizadas *online*. De esta forma, los anunciantes pueden comprobar la rentabilidad de cada euro invertido en la campaña.
- Realizan tantos cambios y correcciones como consideren necesarios, y de forma gratuita, para optimizar la inversión publicitaria.

1.2.2 BingAds

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 (20).

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 (21).



1.2.3 Yahoo Search Marketing

Yahoo! Search Marketing es la plataforma comercial de Yahoo! para las búsquedas patrocinadas y publicidad CPC. Comenzó su actividad en el año 2003 luego que Yahoo! adquiriera Overture, una compañía pionera en el mercado de los buscadores y la publicidad CPC (22).

Entre sus principales características se encuentran (22):

Segmentación geográfica. Un anuncio se puede mostrar a la totalidad del mercado o restringir su distribución a áreas específicas. La identificación del mercado ideal permite seleccionar mejor los clientes y controlar los costes.

Presupuestario de campaña. Los presupuestos mensuales se establecen para cada campaña. El programa ofrece una estimación de las impresiones y los clics que recibirán los anuncios para las ofertas máximas definidas. Se puede incrementar o disminuir el número de impresiones y clics previstos ajustando la oferta máxima o el presupuesto mensual.

Coincidencia de contenido o *Content Match*. Es posible ampliar el alcance del anuncio más allá de los resultados patrocinados presentando las listas en la red de sitios de socios.

Herramientas de optimización. El sistema de *Sponsored Search* presenta de forma automática y con más frecuencia los anuncios más eficaces.

Funciones de administración de cuentas. Se puede programar la frecuencia de publicación de la campaña definiendo fechas de inicio y de finalización específicas. El sistema también ofrece la posibilidad de obtener sugerencias relativas a las palabras clave según los anuncios existentes y de administrar los anuncios por lotes, importando lotes de campañas, anuncios y palabras clave en la cuenta mediante una hoja de cálculo.

Informes. Existen informes de rendimiento personalizables en función de varios parámetros que permiten consultar una descripción general del rendimiento de las campañas o bien un desglose detallado del rendimiento de grupos de anuncios, título/descripción y palabra clave. También pueden visualizarse informes de facturación y pago, generar gráficos y obtener las previsiones sobre el rendimiento de una palabra clave y un grupo de anuncios.

Todos estos sistemas comparten un grupo de características que podemos apreciar en la siguiente tabla.



Fundamentación teórica

Tabla 1. Sistemas de Publicidad en buscadores. Fuente: Elaboración propia.

Sistemas	Costo por clic (CPC)	Relación términos – palabras claves	Red de contenidos y formatos	Contador de visitas
AdWords	Sí	Sí	Texto expandido, estándar e imagen, video.	Sí
BingAds	Sí	Sí	Texto expandido, estándar.	Sí
Yahoo Search Marketing	Sí	Sí	Texto expandido, estándar e imagen.	Sí

Después de realizar un análisis sobre los sistemas de publicidad estudiados, se llega a la conclusión de que estos utilizan el método de pago costo por clic (CPC), esta característica es muy empleada por las estrategias de *marketing* en buscadores, la cual plantea que el buscador recibirá ganancias por la cantidad de clics que un anuncio reciba, pero no cobrará por mostrarlos a los clientes, también están los varios formatos en los que son publicados los anuncios, que permite que el usuario escoja como quiere ver el anuncio o de que tipo busca publicarlo. Los términos posibilitan que los clientes que realizan búsquedas puedan ver los anuncios publicitarios, que están relacionados con un conjunto de palabras claves definidas por la búsqueda realizada. Por último y no menos importante están las visitas, las cuales permitirán medir el impacto o calidad de un anuncio, datos que son muy útiles a los usuarios para ver las necesidades de sus productos.

Todos y cada uno de los sistemas antes mencionados, tienen su sede central en Estados Unidos, lo que conlleva que la gran mayoría de sus servicios, entre los que se encuentran los publicitarios, no sean brindados a nuestro país por las leyes injustas impuestas por el bloqueo económico, además utilizar un sistema publicitario cubano, aportaría una mayor soberanía tecnológica y otro método para evitar los controles injerencistas de nuestro vecino del norte.

El subsistema que se propone utilizará las características comunes de estos sistemas estudiados debidos a que fueron las de más relevancia.



1.3. Selección del entorno de desarrollo para la construcción de la solución

Para el desarrollo de la propuesta de solución se hace necesario el estudio de las herramientas y tecnologías que se emplearán en correspondencia con la metodología usada en el desarrollo de la Plataforma C.U.B.A.

1.3.1. Metodología de desarrollo de software

Una metodología de desarrollo de software impone un proceso de forma disciplinada sobre el desarrollo de software con el objetivo de hacerlo más predecible y eficiente (23). Está orientada a la forma adecuada de construcción del software, que puede incluir modelos de sistemas, notaciones, reglas, sugerencias de diseño y guías de procesos (24).

Un concepto más ampliado y que agrupa a los dos anteriores es el que ofrece (25); plantea que una metodología de desarrollo, en ingeniería de *software*, es un conjunto de herramientas, técnicas, procedimientos y soporte documental encaminados a estructurar, planificar y controlar el proceso de desarrollo de forma organizada y lógica, que tiene como objetivo apoyar a los desarrolladores en la creación de un nuevo *software*.

Para el desarrollo de la solución se usará AUP en variación de la metodología para la universidad (AUP-UCI), por ser la metodología empleada en el centro CIDI para el desarrollo de proyectos.

La metodología AUP-UCI contribuye de manera significativa al desarrollo del *software* que se espera obtener, ya que esta metodología además de ser ágil es también adaptable a cualquier circunstancia que pueda surgir en el proceso de desarrollo del proyecto. La UCI le ha realizado modificaciones con el fin de adaptarlo al ciclo de vida definido para la actividad productiva de dicha institución; de las 4 fases que encierra la metodología AUP se simplificaron a:

Inicio: esta etapa se mantiene de AUP pero se modifica el objetivo de la misma; realización de la planeación del proyecto en cuanto a un estudio de la organización del cliente, el alcance del proyecto, estimaciones de tiempo esfuerzo y costo que serán necesarios para decidir si se desarrolla el producto o no.

Ejecución: en esta fase se recogen las actividades que desarrolla AUP de elaboración, construcción y transición. Se ejecutan las actividades requeridas para obtener el *software* y ajustar los planes del proyecto, realizando las tareas de modelado del negocio, obtención y análisis de los requisitos, elaboración de la arquitectura, diseño, implementación y liberación de producto.



Cierre: en esta fase se analizan los resultados del proyecto, su ejecución y se llevan a cabo actividades formales de cierre del proyecto.

1.4 Lenguajes para la programación

Los lenguajes para programación son el idioma mediante el cual se comunican las máquinas entre ellas y con los seres humanos. A continuación se presentan los lenguajes que serán utilizados para la realización de la solución tanto del lado del cliente como del lado del servidor, seleccionados debido a que son los que emplea el marco de trabajo symfony.

1.4.1 Del lado del servidor

PHP: Es un lenguaje interpretado de alto nivel embebido en páginas HTML, enfocado principalmente a la programación de guiones (*scripts*) del lado del servidor, por lo que se puede recopilar datos de formularios, generar páginas con contenidos dinámicos, o enviar y recibir *cookies* (26). PHP puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno, es rápido y cuenta con una gran librería de funciones y mucha documentación (27).

Se escogió PHP como lenguaje de programación del lado del servidor por ser un lenguaje de alto nivel e interpretado, utilizado en su mayoría para el procesamiento dinámico de información en la web. Dicha selección se basa fundamentalmente en las características y múltiples ventajas que proporciona; teniendo en cuenta que es un lenguaje completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos. Además, es uno de los lenguajes de programación web que más documentación posee, dispone de un sitio oficial que brinda toda la documentación que los desarrolladores necesiten, y cuenta con una gran comunidad de desarrollo. Lo soportan la mayoría de las plataformas de alojamiento web y está en continuo desarrollo.

1.4.2 Del lado del cliente

Para el trabajo en la web se utilizará HTML (*HyperText Markup Language*), CSS y JavaScript, todos estos integrados en el Framework Bootstrap.

Bootstrap: Bootstrap, es un marco de trabajo creado por Twitter. Este *framework* permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice, es decir, el sitio web se adapta automáticamente al tamaño de una PC, una *tablet* u otro dispositivo; esta técnica de diseño y desarrollo se conoce como “*responsive design*” o diseño adaptativo. Los diseños creados con Bootstrap son simples, limpios e intuitivos, esto les da agilidad a la



hora de cargar y de adaptarse a otros dispositivos. El marco de trabajo trae varios elementos con estilos predefinidos fáciles de configurar: botones, menús desplegables, formularios incluyendo todos sus elementos e integración con *jQuery* para ofrecer ventanas e información sobre herramientas dinámicas (28). Bootstrap utiliza HTML, lenguaje de publicación especificado como un estándar por el W3C (*World Wide Web Consortium*), CCS para aplicar diferentes estilos a los documentos creados (29) y JavaScript para brindarle dinamismo a las páginas web (30).

1.4.3 Framework. Marco de trabajo

En el desarrollo de *software*, un *framework* es una estructura conceptual y tecnológica de soporte definida, normalmente, con artefactos o módulos de *software* concretos. Los objetivos principales que persigue son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones (31). A continuación, se realiza un estudio de marcos de trabajo para el desarrollo de *software*, con el objetivo de escoger el más conveniente para la implementación del subsistema de publicidad aplicado a la Plataforma C.U.B.A.

CodeIgniter: Es un marco de trabajo de código abierto para aplicaciones web en PHP. CodeIgniter tiene muchas características que lo hacen destacar entre sus homólogos, ya que a diferencia de otros *frameworks* es muy exhaustivo con la documentación. En el lado de la programación es compatible con PHP4 y PHP5, por lo que es posible que corra en la mayoría de los servidores web. También usa el patrón Modelo Vista Controlador. Lo que permite organizar la aplicación en diferentes partes como son el modelo, la vista, y el controlador. CodeIgniter también tiene una implementación del patrón Active Record, lo que hace sencillo escribir consultas y hace la aplicación más entendible (32).

Symfony: Symfony es un *framework* PHP de alto rendimiento de código abierto para el desarrollo de aplicaciones web, construido sobre una serie de componente web reutilizables. Posee la característica de que es fácil de instalar y puede ser usado tanto en Linux como en Windows. Symfony tiene una corta curva de aprendizaje y una comunidad de desarrollo comprometida para llevar PHP al próximo nivel (33).

Zend framework.: Es un marco de trabajo de código abierto de aplicaciones web, orientado a objetos e implementado en PHP 5, desarrollado con el objetivo de simplificar el desarrollo web. Permite a los desarrolladores reutilizar componentes en sus aplicaciones (Esaú, 2015). Zend Framework presenta como desventaja que es una herramienta pesada que gasta mucha memoria, además de ser muy pobre la comunidad de desarrolladores, por lo que podría ser difícil encontrar un buen manual para realizar un



proceso o aclarar una duda. La estructura de archivos de Zend no está definida por sí misma, cualquier desarrollador puede modificarla, esto indica que no sería compatible con otra aplicación desarrollada con Zend (34).

Cada uno de los anteriores *frameworks* de desarrollo PHP se ajustan a las necesidades de implementación del sistema propuesto, pero, se decide utilizar Symfony en su versión 3.4 por que este permite cambiar con facilidad de cualquier sistema gestor de base de datos (SGBD) en cualquier fase del proyecto, sencillo de utilizar, sigue la mayoría de mejores prácticas y patrones de diseño para la web, código fácil de leer que incluye comentarios de *phpDocumentor* y que permite un mantenimiento muy sencillo y fácil de extender, lo que permite su integración con las librerías de otros fabricantes.

1.5 Capa de servicios

Para la comunicación entre la Plataforma C.U.B.A. 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.

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 (35).

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 (36).



1.6 Herramientas

PHPStorm: PhpStorm es perfecto para trabajar con Symfony, Drupal, WordPress, Zend Framework, Laravel, Magento, Joomla!, CakePHP, Yii y otros *frameworks*. El editor realmente obtiene su código y comprende profundamente su estructura, y es compatible con todas las funciones del lenguaje PHP para proyectos modernos y heredados. Proporciona la mejor terminación de código, refactorizaciones, prevención de errores sobre la marcha y más. Aprovecha al máximo las tecnologías de vanguardia, como HTML5, CSS, Sass, Less, Stylus, CoffeeScript, TypeScript, Emmet y JavaScript, con refactorizaciones, depuración y pruebas de unidad disponibles. Ver los cambios al instante en el navegador gracias a Live Edit (37).

Visual Paradigm: Visual Paradigm es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de *software*: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Además es una herramienta de diseño multiplataforma. Soporta los principales estándares de la industria, tales como; UML, SysML, BPMN, XMI. Ofrece un conjunto completo de herramientas que los equipos de desarrollo necesitan para la captura de requisitos, planificación de *software*, planificación de pruebas, modelamiento de clases, modelamiento de datos y otras actividades (38).

MySQL.: Este SGBD es muy conocido y ampliamente usado por su simplicidad y notable rendimiento. Aunque carece de algunas características avanzadas disponibles en otros SGBD del mercado, es una opción atractiva tanto para aplicaciones comerciales como de entretenimiento. Esto y su libre distribución en internet bajo licencia GPL le otorgan como beneficios adicionales, (no menos importantes), contar con un alto grado de estabilidad y un rápido desarrollo (39). Está disponible en gran cantidad de plataformas y sistemas. Un gran porcentaje de las utilidades de MySQL no están documentadas y se utiliza en aplicaciones web no muy complejas que requieren muchos usuarios.

Acunetix Web Vulnerability Scanner: Es la herramienta utilizada en la universidad para la detección de vulnerabilidades en sitios y aplicaciones web. Esta herramienta posee un componente que facilita la realización de pruebas a formularios y a áreas protegidas por contraseña. Permite realizar varias peticiones simultáneamente, siendo capaz de explorar cientos de páginas sin interrupciones. Para la realización de las pruebas de seguridad al subsistema de alertas se utiliza esta herramienta en su versión 8.0 (40).

JMeter: es un proyecto de código abierto de Apache que puede ser utilizado como una herramienta de prueba de carga para analizar y medir el rendimiento. En un principio fue diseñado para probar aplicaciones



web, pero se ha expandido hacia otros tipos de servicios. Tiene la capacidad de extraer datos de formatos de respuestas como: HTML, JSON, XML o cualquier formato textual. JMeter puede ser usado como una herramienta de pruebas unitarias para conexiones de bases de datos con JDBC, FTP, LDAP, servicios web, JMS, HTTP y conexiones TCP genéricas (41).

Conclusiones parciales del capítulo

En este capítulo se han abordado los elementos correspondientes a la fundamentación teórica del subsistema de publicidad, arribando a las siguientes conclusiones:

- En la evaluación de las tendencias de aplicaciones similares a la propuesta de solución presentada en este trabajo, se identificaron sistemas con un alto nivel de aceptación pero cada uno responde a las necesidades particulares; aun así fue posible reconocer a partir de ellos, nuevos elementos que contribuyen al perfeccionamiento de la propuesta de solución.
- Las herramientas y tecnologías basadas en *software* libre, posibilitaron obtener una base tecnológica enfocada en los componentes que utilizan los sistemas de publicidad estudiados.



CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL SUBSISTEMA DE PUBLICIDAD APLICADO A LA PLATAFORMA C.U.B.A.

Introducción

En este capítulo se presentan aspectos relacionados con el diseño del subsistema a desarrollar. Se identifican los requisitos funcionales y no funcionales que deberá presentar el *software*. Se generan los artefactos relacionados con estos requisitos identificados; así como la modelación de diagramas.

2.1 Descripción de la propuesta de solución

Teniendo en cuenta la problemática existente y para darle cumplimiento al objetivo general se propone desarrollar un subsistema de publicidad de anuncios que permita a la Plataforma C.U.B.A. brindar este servicio a empresas e instituciones y así potenciar sus campañas comerciales.

Se propone implementar un subsistema que permita a los clientes promocionar sus productos y servicios a través de la Plataforma C.U.B.A.. 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 y devolverá estos anuncios a la Plataforma C.U.B.A. 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 administradores y clientes así como los anuncios que dichos clientes desean publicar.

2.2 Especificación de requisitos de software

Los requisitos del *software* permiten establecer lo que el sistema debe hacer, sus características fundamentales y las restricciones en el funcionamiento del sistema y los procesos de desarrollo del *software*. De manera general, estos requisitos expresan las necesidades objetivas que presentan los usuarios, ante un sistema que resuelve un problema en particular de un determinado dominio (24).

2.2.1 Requisitos funcionales

Los requisitos funcionales de un sistema describen lo que el sistema debe hacer (24).



Tabla 2. Requisitos funcionales. Fuente: Elaboración propia.

No.	Requisito Funcional	Prioridad
RF 1.	Crear usuario	Alta
RF 2.	Editar usuario	Medio
RF 3.	Listar usuario	Baja
RF 4.	Eliminar usuario	Media
RF 5.	Crear anuncio	Alta
RF 6.	Editar anuncio	Alta
RF 7.	Eliminar anuncio	Baja
RF 8.	Listar anuncio	Media
RF 9.	Buscar anuncio	Media
RF 10.	Crear campaña publicitaria	Alta
RF 11.	Editar campaña publicitaria	Alta
RF 12.	Eliminar campaña publicitaria	Baja
RF 13.	Listar campaña publicitaria	Media
RF 14.	Activar anuncios	Baja
RF 15.	Desactivar anuncios	Baja
RF 16.	Notificar al cliente cuando su anuncio sea publicado	Baja



RF 17.	Brindar anuncio por término	Alta
RF 18.	Crear término	Media
RF 19.	Editar término	Baja
RF 20.	Listar término	Baja
RF 21.	Eliminar término	Baja

2.2.2 Requisitos no funcionales

Los requisitos no funcionales son aquellos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento (24).

A continuación, se agrupan los requisitos no funcionales obtenidos por una serie de criterios definidos a conveniencia del autor.

RNF 1. Requisitos de usabilidad.

1.1. 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.

1.2. El subsistema podrá ser utilizado por los usuarios que se registren y accedan.

RNF 2. Requisitos de eficiencia.

2.1. El subsistema deberá tener un tiempo de respuesta máximo de 4 segundos para un total de 100 usuarios concurrentes, que realizan accesos a la base de datos.

2.2. El subsistema debe permitir la navegación de 100 usuarios como máximo simultáneamente.

3.3. El subsistema 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.

RNF 3. Requisitos de seguridad.



3.1. Confidencialidad: en el sistema solo podrá gestionar información el personal autorizado, con los permisos correspondientes, por lo que será necesaria una autenticación previa.

3.2. Integridad: el software deberá estar protegido contra acciones no autorizadas o que puedan afectar la integridad de los datos.

3.3. El sistema contará con dos niveles de acceso:

- Usuario
- Administrador

RNF 4. Requisitos de *software*.

4.1. Para el servidor de base de datos: *MariaDB* versión 5.0.12

4.2. Para el servidor web: Apache versión 2.4.34.

4.3. Dependencias php 7.2.9

4.4. Sistema operativo CENTOS (última versión estable) a 64 Bits.

4.5. PC cliente: cliente ligero y un navegador que soporte HTML5 y CCS3.

RNF 5. Requisitos de *hardware*.

5.1. Para el servidor de base de datos: RAM: 16 GB, CPU: i5 4ta generación, Disco duro: 1 TB.

5.2. Para el servidor web: RAM: 4GB, CPU: i3 4ta generación Disco duro: 50 GB.

5.3. PC cliente: RAM: 256 MB.

2.3 Historias de usuarios

Las historias de usuario son una técnica utilizada en las metodologías de desarrollo para la especificación de requisitos. Una historia de usuario es una representación de un requisito escrito en una o dos frases utilizando el lenguaje común del usuario (Suaza, 2013). En estas el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas (42).



Tabla 3. Historia de usuario del RF1. Fuente: Elaboración propia.

Número: 1	Nombre del requisito: Crear usuario
Programador: Andy Vega Fajardo	Iteración Asignada: 1
Prioridad: Alta	Tiempo Estimado: 8
Riesgo en Desarrollo: Media	Tiempo Real:
<p>Descripción: Muestra un formulario con una serie de datos que el usuario deberá llenar para completar el registro.</p> <ul style="list-style-type: none">▪ Nombre(s) (Obligatorio. Campo de texto. Permite caracteres alfabéticos y espacios).▪ Correo electrónico (Obligatorio. Campo de tipo e-mail).▪ Empresa o Institución (Obligatorio. Campo de texto. Permite caracteres alfabéticos y espacios).▪ Contraseña (Obligatorio. Campo de tipo <i>password</i>. Longitud mínima de 6 caracteres).▪ Confirmar contraseña (Obligatorio. Campo de tipo <i>password</i>. Longitud mínima de 6 caracteres).	
<p>Observaciones:</p> <ul style="list-style-type: none">▪ Al completar el registro, si no se incumplen las condiciones del mismo, el sistema notifica el registro satisfactorio, en caso contrario se muestra un mensaje de error y señalan los campos que incumplen las condiciones requeridas.	
Prototipo de interfaz:	



Registro usuario

Nombre(s)

Apellidos

Correo electrónico

Empresa o Insitución

Contraseña

Confirmar contraseña



Tabla 4. Historia de usuario del RF5. Fuente: Elaboración propia.

Número: 5	Nombre del requisito: Insertar anuncio publicitario
Programador: Andy Vega Fajardo	Iteración Asignada: 1
Prioridad: Alta	Tiempo Estimado: 12
Riesgo en Desarrollo: Alta	Tiempo Real:
<p>Descripción: Muestra un formulario con una serie de campos que el usuario deberá llenar</p> <ul style="list-style-type: none">▪ Nombre(s) (Obligatorio. Campo de texto. Permite caracteres alfabéticos y espacios).▪ Descripción (Obligatorio. Campo de texto. Permite caracteres alfanuméricos y espacios).▪ Url (Obligatorio. Campo de texto. Permite caracteres alfanuméricos).▪ Imagen (Opcional. Campo de archivo. Permite imágenes diferentes formatos).▪ Fecha Creado (Obligatorio. Campo de tipo Fecha)▪ Términos (Obligatorio). <p>Adjunto al formulario se encuentra el botón crear el cual inserta el anuncio.</p>	
<p>Observaciones:</p> <ul style="list-style-type: none">▪ Al completar el registro, si no se incumplen las condiciones del mismo, el sistema notifica el registro satisfactorio, en caso contrario se muestra un mensaje de error y señalan los campos que incumplen las condiciones requeridas.	
Prototipo de interfaz:	



Crear anuncio

Nombre

Descripción

Url del sitio Fecha creado

Imágen

Términos

Tabla 5. Historia de usuario del RF7. Fuente: Elaboración propia.

Número:	Nombre del requisito: Eliminar anuncio
Programador: Andy Vega Fajardo	Iteración Asignada: 1
Prioridad: Baja	Tiempo Estimado: 8
Riesgo en Desarrollo: Baja	Tiempo Real:
Descripción: En el listado de anuncios en la columna de la derecha se encontrara el botón mostrar usuario, el cual mostrará todos los atributos del anuncio seleccionado. Al final estará el botón eliminar, que eliminara el anuncio del listado.	



Observaciones:

- El botón eliminar solo será mostrado a los usuarios autenticados en el sistema.

Prototipo de interfaz:

Mostrar anuncio

Nombre:	Anuncio 1
Descripción:	Esto es una prueba para prototipo de interfaz
Url:	anuncio1@an.z
Fecha creado:	X / X / X
Imágen	
Términos	Término 1 Término 2

Volver Editar Eliminar

Tabla 6. Historia de usuario del RF8. Fuente: Elaboración propia.

Número:	Nombre del requisito: Listar anuncio
Programador: Andy Vega Fajardo	Iteración Asignada: 1
Prioridad: Media	Tiempo Estimado: 8
Riesgo en Desarrollo: Baja	Tiempo Real:



Descripción: En la interfaz del usuario se encontrara el botón listado de anuncios, el cual mostrará todos los anuncios creados por el usuario autenticado.

Observaciones: La interfaz listar usuario solo será mostrada a los usuarios autenticados.

Prototipo de interfaz:

Listado de anuncios

Nombre	Imágen	Descripción	Url	Fecha Creado	Términos		
Anuncio 1		Texto	Url	X/X/X	Término 1 Término 2	Mostrar	Editar
Anuncio 2		Texto	Url	X/X/X	Término 1 Término 2	Mostrar	Editar
Anuncio 3		Texto	Url	X/X/X	Término 1 Término 2	Mostrar	Editar
Anuncio 4		Texto	Url	X/X/X	Término 1 Término 2	Mostrar	Editar
Anuncio 5		Texto	Url	X/X/X	Término 1 Término 2	Mostrar	Editar

2.4 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.

2.4.1 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 (43).

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 “*UsuarioController*” el cual se encarga de mostrar un mensaje cuando se inserta un usuario.

2.4.2 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 (44).

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 “*Usuario*” 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.



2.5 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”. A continuación se muestra la arquitectura del subsistema (45).

2.5.1 Patrón arquitectónico Modelo – Vista – Controlador

Symfony3 basa su funcionamiento interno en la 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* (46).

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 “*Entity*” los que definen los modelos de datos (24).

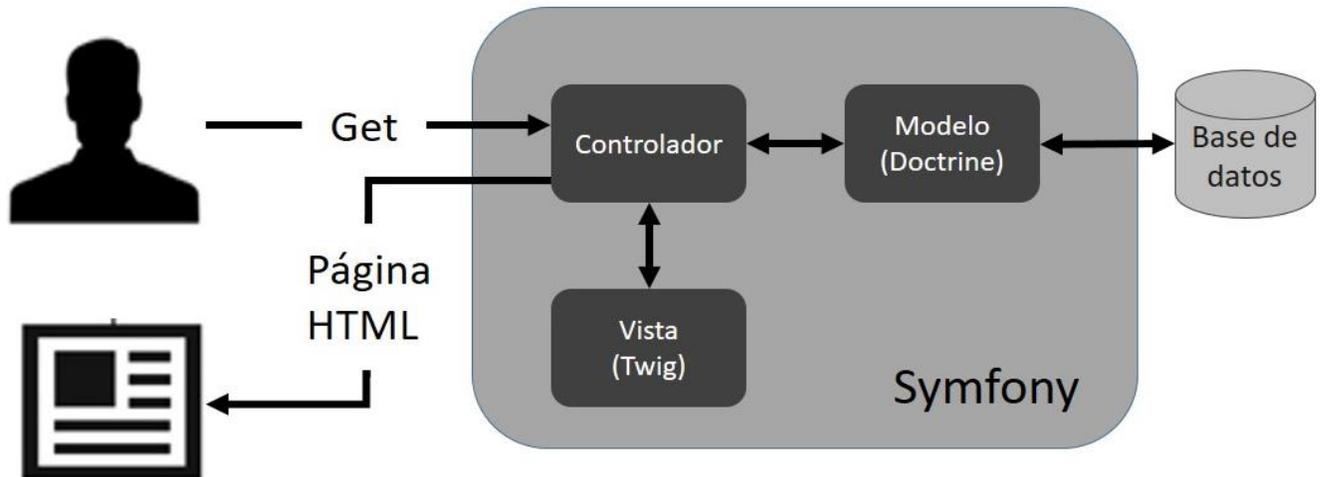


Figura 2. Modelo-Vista-Controlador. Fuente: Elaboración propia.

2.6 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.

2.6.1 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 historia de usuario Gestionar anuncio (24).

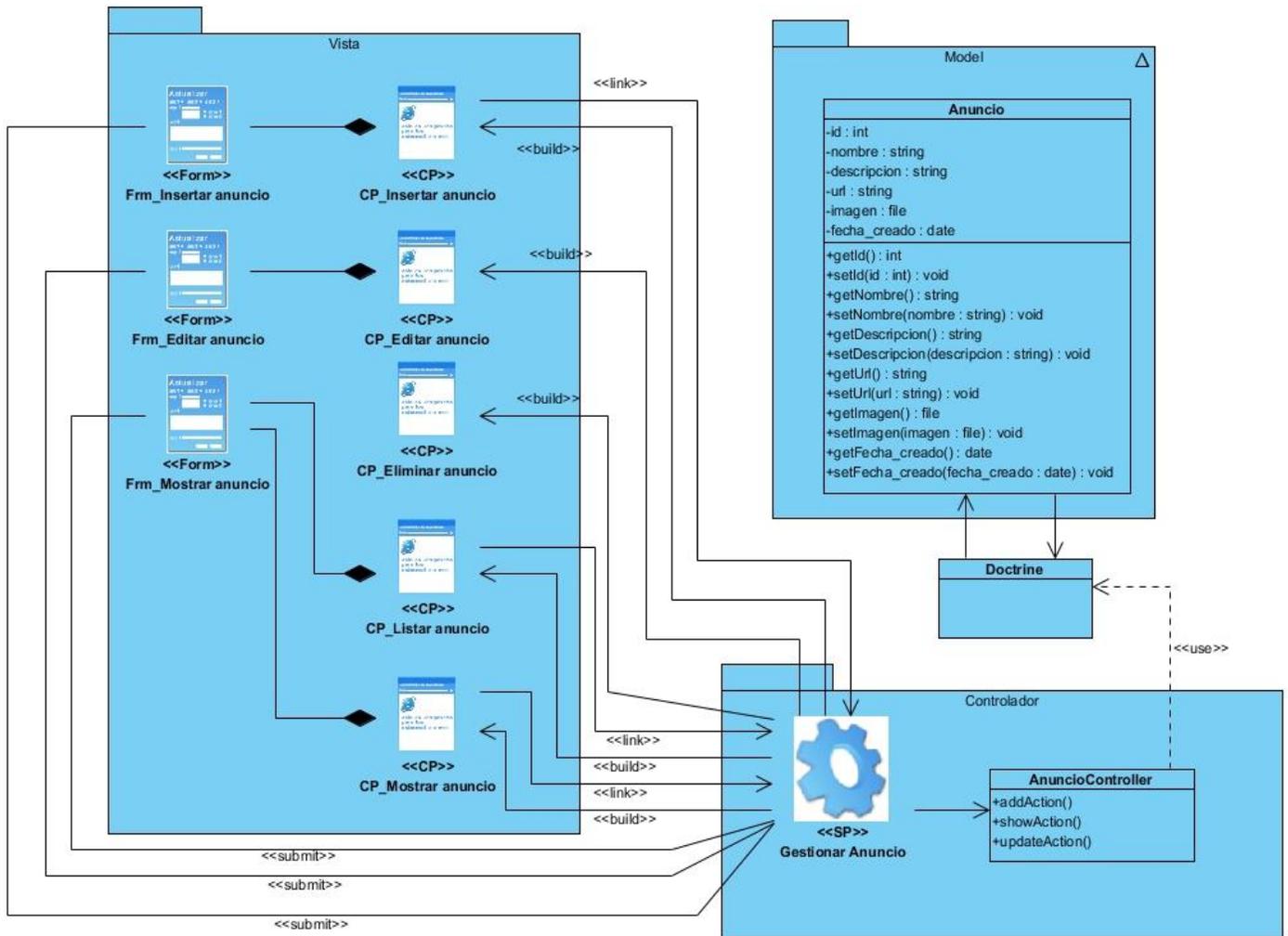


Figura 3. DCD con estereotipos web. Fuente: Elaboración propia.

2.7 Modelo de datos

Un modelo de datos está orientado a representar los elementos que va a procesar el sistema, la composición y atributos de los mismos. Además de dónde se encuentran almacenados actualmente dichos objetos, la relación entre ellos y los procesos que los transforman (47). A continuación, se muestra el modelo de datos que se propone para el desarrollo del sistema.

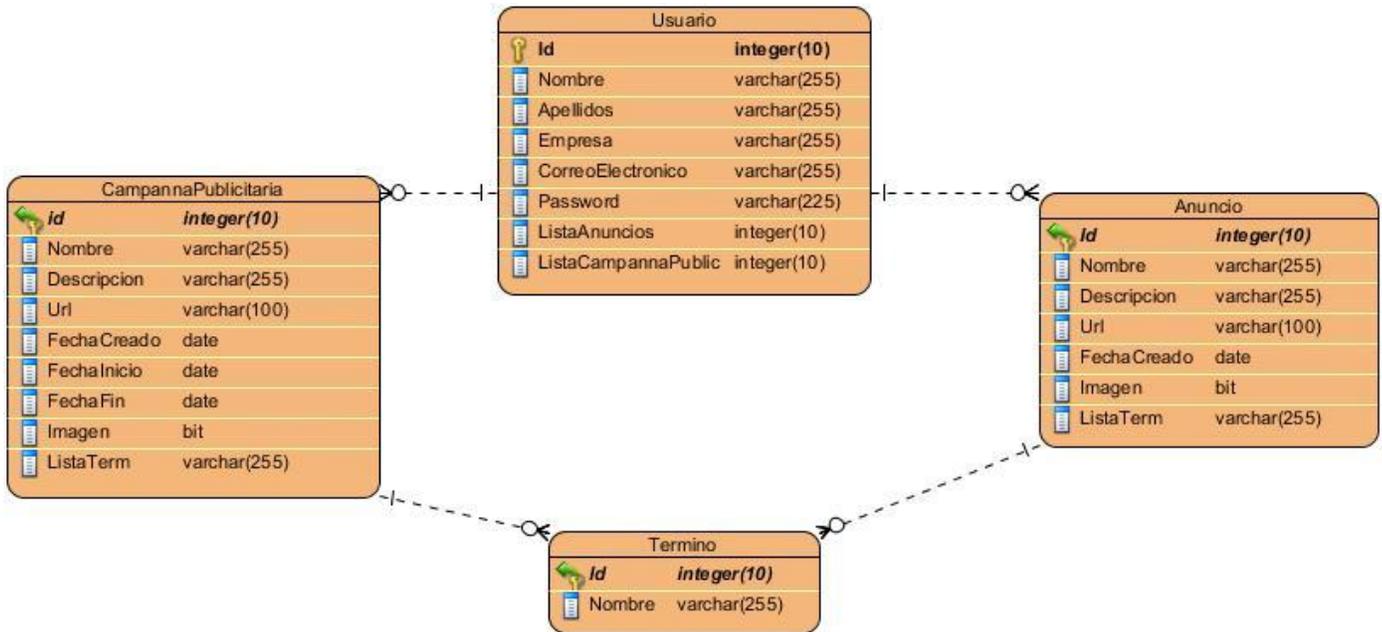


Figura 4. Modelo de datos. Fuente: Elaboración propia.

2.8 Modelo de despliegue

Un modelo de despliegue representa la relación física que se establece entre los distintos componentes o nodos que describen la topología de un sistema. Estos muestran la configuración de los elementos de *hardware* y muestra cómo los elementos y artefactos del software se relacionan en esos nodos (48).

A continuación se muestra el diagrama de despliegue del subsistema propuesto en esta investigación.

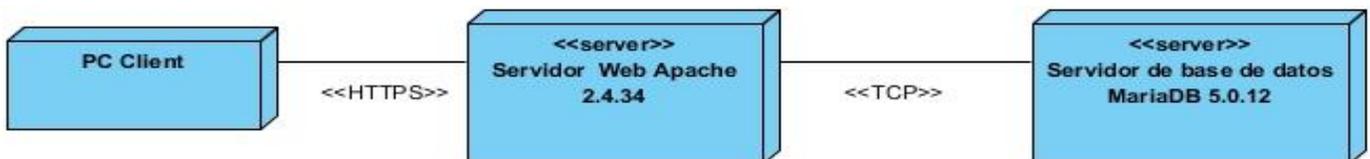


Figura 5. Modelo de despliegue. Fuente: Elaboración propia

PC Cliente: representa las computadoras 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.

Conclusiones parciales del capítulo

En el desarrollo de este capítulo se trataron aspectos correspondientes al análisis y diseño del subsistema de publicidad aplicado a la Plataforma C.U.B.A., llegándose a las siguientes conclusiones:

La especificación de los requisitos funcionales y no funcionales del sistema, así como la representación y descripción de los artefactos generados, permitieron una mayor comprensión del negocio y obtener una visión más clara de los resultados que se pretenden obtener.

La definición de la arquitectura del sistema y los patrones de diseño a utilizar, permitieron establecer buenas prácticas de programación en la fase de implementación, así como disminuir el impacto de los cambios futuros en los componentes del sistema o en el código fuente.

Se pudo identificar la disposición de *hardware* y de software de los componentes de la herramienta que se pretende desarrollar.



CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DEL SUBSISTEMA DE PUBLICIDAD APLICADO A LA PLATAFORMA C.U.B.A.

Introducción

La etapa de implementación del *software* es el proceso de convertir una especificación del sistema en un sistema ejecutable. Esta fase comprende la materialización, en forma de código, de todos los artefactos, descripciones y arquitectura propuestos en la etapa de análisis y diseño; con el objetivo de conformar el producto final requerido por el cliente (24).

Al tiempo que se desarrolla el *software*, este debe ser sometido a una serie de pruebas que muestren que el sistema se ajusta a su especificación y que cumple con las expectativas del usuario. Esta etapa es conocida como la validación e implica una serie de procesos de comprobación, como las inspecciones y revisiones (24).

3.1 Diagramas de componentes

El diagrama de componentes permite visualizar la estructura de un sistema informático atendiendo a las partes que lo conforman. Cada componente debe ser tratado como una unidad de composición independiente e indispensable dentro de un sistema, pudiéndose encontrar dependencias entre estos. Algunos ejemplos de componentes físicos lo constituyen los archivos, módulos, librerías, ejecutables y binarios (24).

A continuación, se describen los principales paquetes que componen los diagramas de componentes pertenecientes a la interfaz web y al módulo de perfil de usuario de la aplicación.

PublicBundle: agrupa en su interior todos los componentes de la interfaz web, estableciendo una estructura organizativa acorde al patrón de arquitectura MVC.

Controller: contiene las clases controladora encargada de procesar las peticiones de las páginas clientes y del usuario.

Config: contiene los archivos donde se definen las rutas de la aplicación y los servicios disponibles para las vistas.

Entity: contiene las clases que se encargan de mapear las tablas de la base de datos.



Form: contiene las clases para construir los formularios utilizados para el envío de información por parte de los usuarios.

Public: en su interior contiene diferentes componentes de apoyo a las vistas de la aplicación (CSS, JavaScripts, imágenes, entre otros recursos).

Views: agrupa las páginas referentes a las vistas de la aplicación, así como las plantillas bases.

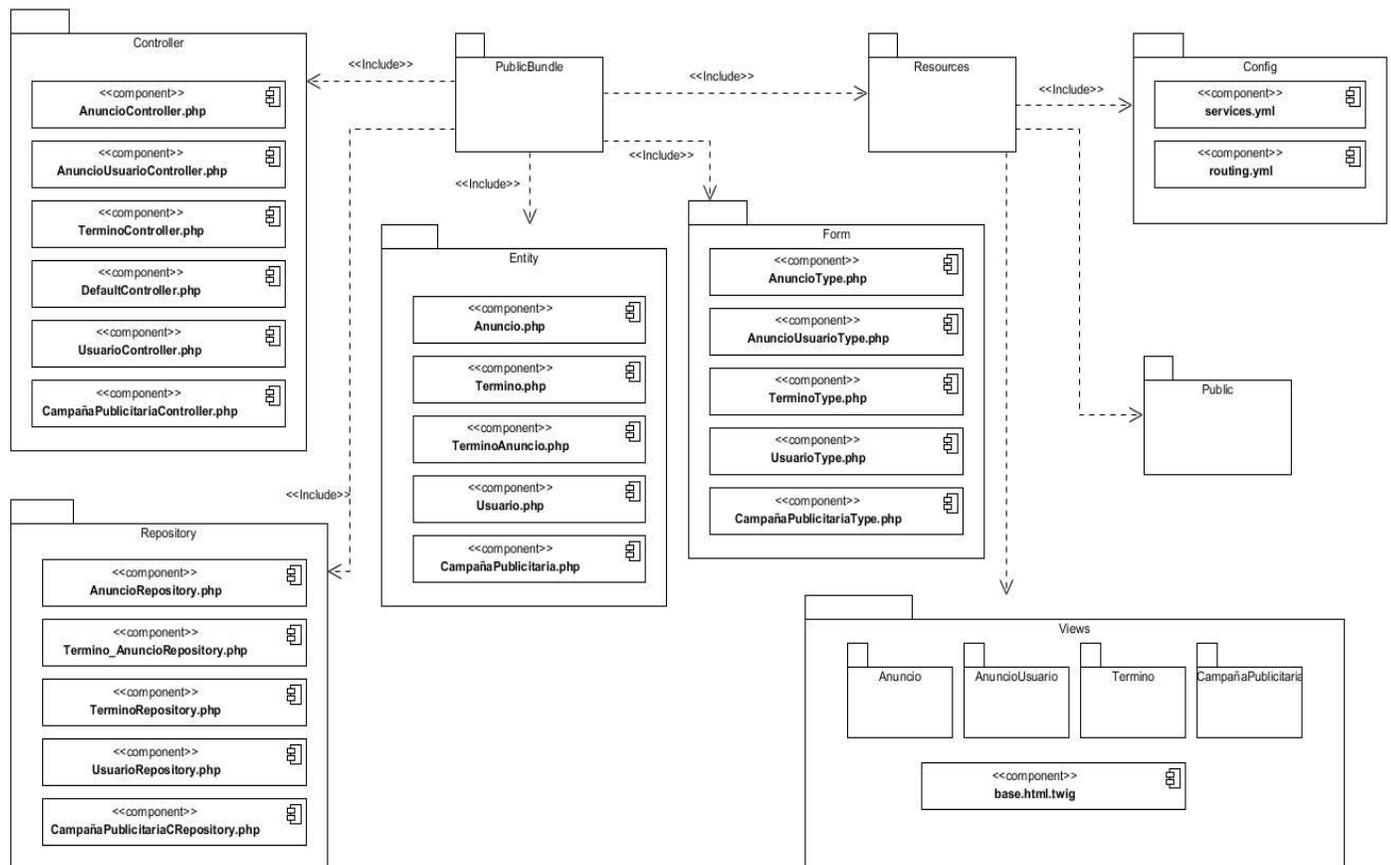


Figura 6. Modelo de componentes de PublicBundle. Fuente: Elaboración propia.

3.2 Estándares de codificación

Los estándares de codificación son un conjunto de buenas prácticas reconocidas por la industria que proporcionan una variedad de directrices para desarrollar código. El cumplimiento de las normas de codificación en el desarrollo de *software* puede mejorar la comunicación del equipo, reducir los errores y



mejorar la calidad del código (Li *et al.*, 2005). Un estilo de programación homogéneo permite que todos los programadores noveles puedan entender el código en menos tiempo y en consecuencia sea mantenible.

En el lenguaje PHP se utilizó el estándar de codificación que establece el *Framework* Symfony, el cual sigue los estándares definidos en los documentos PSR-0, PSR-1, PSR-2, PSR-4 (PHP-Fig (49), 2017; PHP-Fig (50), 2017; PHP-Fig (51), 2017; PHP-Fig (52)).

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*.

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
```



Figura 7. Ejemplo de uso de CamelCase. Fuente: Elaboración propia.

3.3 Validación del Subsistema de Publicidad aplicado a la Plataforma C.U.B.A.

A continuación, se detallan los tipos de pruebas aplicadas al subsistema desarrollado. Las mismas persiguen detectar las no conformidades respecto a las funcionalidades de la aplicación, la medición del grado de usabilidad de las funcionalidades implementadas, así como la correcta integración entre los diferentes componentes de la arquitectura del sistema.

3.3.1 Pruebas funcionales

Las pruebas funcionales se aplican a un software determinado, con el objetivo de validar que las funcionalidades implementadas se desempeñen de acuerdo a las especificaciones de los requisitos definidos con anterioridad. Al conocer las funciones específicas que se le asignaron a una aplicación para su realización, pueden llevarse a cabo casos de pruebas que demuestren que cada función es completamente operativa mientras que al mismo tiempo se buscan errores en cada función (47).

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 a las historias de usuario “Crear anuncio” y “Editar anuncio”.

Tabla 7. Descripción de las variables correspondientes al caso de prueba para las historias de usuarios “Crear anuncio” y “Editar anuncio”. Fuente: Elaboración propia.

No	Nombre del campo	Clasificación	Valor nulo	Descripción
V1	Nombre	Campo de texto	no	Campo que admite caracteres de cualquier tipo. Inicialmente contiene 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 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 búsqueda y selección	no	Campo que contiene una lista de objetos de tipo Término, puede modificarse.
V5	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 a “Crear anuncio”, “Editar anuncio”, “Eliminar anuncio” y “Listar anuncio”.

Tabla 8. Caso de prueba para la historia de usuario: Sección 1: “Crear anuncio”. Fuente: Elaboración propia.

Sección 1: “Crear anuncio”								
Escenario	Descripción	V1	V2	V3	V4	V5	Respuesta del sistema	Flujo central



EC 1.1 Insertar datos del anuncio correctamente.	Se insertan todos los datos del anuncio correctamente.	V	V	V	V	V	Inserta un nuevo anuncio. Muestra la lista de los anuncios creados y un mensaje indicando que el anuncio fue creado satisfactoriamente.	<ol style="list-style-type: none">1. Se selecciona la opción "Crear Anuncio" en la 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	NA	Señala los campos que aún estén vacíos, para que se llenen. Muestra un mensaje indicando que el campo no puede estar vacío.	<ol style="list-style-type: none">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	NA	Señala los campos que tengan datos inválidos. Muestra un mensaje indicando que el	<ol style="list-style-type: none">1. Se selecciona la opción "Crear Anuncio" en la página



								dato entrado es incorrecto.	principal. 2. Se introducen los datos. 3. Se selecciona la opción "Crear".
--	--	--	--	--	--	--	--	-----------------------------	---

Tabla 9. Caso de prueba para la historia de usuario: Sección 2: "Editar anuncio". Fuente: Elaboración propia.

Sección 2: "Editar anuncio"									
Escenario	Descripción	V1	V2	V3	V4	V5	Respuesta del sistema	del	Flujo central
EC 2.1 Insertar datos editados de un anuncio correctamente.	Se guardan correctamente los cambios realizados.	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	NA	Señala los campos que aún estén vacíos, para que se llenen. Muestra un mensaje indicando que el campo no puede estar vacío.	<ol style="list-style-type: none">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	NA	Señala los campos que	<ol style="list-style-type: none">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.	<ol style="list-style-type: none">opción "Editar" del anuncio.2. Se editan los datos.



--	--	--	--

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

V: Versión

Tabla 12. Resumen de los resultados de las pruebas aplicadas. Fuente: Elaboración propia.

Fecha	V	Caso de prueba	Cantidad de no conformidades	Cantidad de no conformidades PD	Cantidad de no conformidades RA
18/4/2019	1.0	HU: Gestionar anuncio	8	-	8
20/4/2019	1.1	HU: Gestionar anuncio	4	-	4
		HU: Gestionar usuario	10	-	10
21/4/2019	1.2	HU: Gestionar anuncio	0	-	0
		HU: Gestionar usuario	3	-	3
28/4/2019	1.3	HU: Gestionar término	7	-	7
		HU: Gestionar usuario	0	-	0



30/4/2019	1.4	HU: Gestionar término	3	-	3
31/4/2019	1.5	HU: 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 3 No conformidades de las cuales 1 eran de validación, 1 de funcionalidad y 1 de ortografía.
- Cuarta Iteración: se encontraron 7 No conformidades las cuales eran 3 de funcionalidad y 4 de ortografía.
- Quinta Iteración: se encontraron 3 No conformidades las cuales eran 3 de ortografía.
- Sexta Iteración: no se encontraron No conformidades.

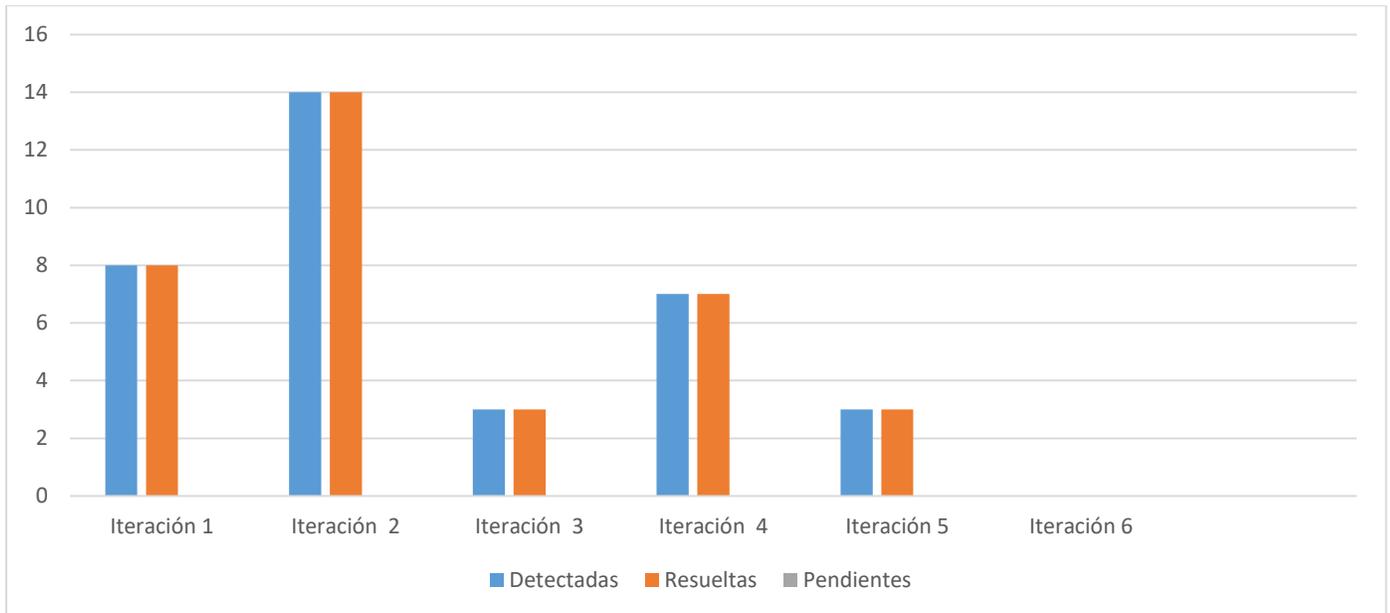


Figura 8. Resultados de las pruebas funcionales. Fuente: Elaboración propia.

3.3.2 Pruebas de carga y estrés

Las pruebas de carga y estrés se diseñan para enfrentar los programas a situaciones anormales. Evalúan la robustez y la confiabilidad del *software* sometiéndolo a condiciones de uso extremas. Ejecutan un sistema, de manera que demande recursos en cantidad, frecuencia o volúmenes extremos (47).

Para realizar las pruebas de carga y estrés al Subsistema de Publicidad aplicado a la Plataforma C.U.B.A. se utiliza la herramienta Apache JMeter en su versión 3.1, pues resulta necesario comprobar el rendimiento del sistema soportando una cantidad máxima de usuarios que soliciten recursos en la web.

La prueba define 400 hilos de concurrencia, los cuales simulan 400 usuarios accediendo concurrentemente cada "1 segundo". La prueba fue realizada en un servidor Core-i3 a 2.4Ghz de velocidad de procesador, una memoria RAM de 4.00Gb.

Tabla 13. Resultados de las pruebas de carga y estrés. Fuente: Elaboración propia.

Total	Muestras	Media	Min	Max	%Error	Rendimiento	Kb/s Recibidos
	2000	4891	507	23798	0.02%	2.0/s	22.22



Descripción de los parámetros evaluados:

Muestras: cantidad de hilos utilizados para la URL.

Media: tiempo promedio en milisegundos para un conjunto de resultados.

Min: tiempo mínimo que demora un hilo en acceder a una página.

Max: tiempo máximo que demora un hilo en acceder a una página.

%Error: porciento de error de las respuestas de las peticiones.

Rendimiento: rendimiento medido en los requerimientos por segundo / minuto / hora.

Kb/s Recibidos: rendimiento medido en Kbytes por segundo.

De manera general los resultados arrojados son satisfactorios, pues muestran que el servidor, sometido a 2000 peticiones de 400 usuarios concurrentes, atendió, con 0.0.2% de error en la respuesta de la URL, en un tiempo medio menor de 2 segundos.

3.3.3 Pruebas de seguridad

Las pruebas de seguridad se diseñan para sondear las vulnerabilidades del lado del cliente, del entorno del lado del servidor, y las comunicaciones que ocurren conforme viajan los datos entre el cliente y el servidor. Cada uno de estos dominios pueden ser atacados, siendo de vital importancia descubrir las debilidades que puedan explotar quienes tengan intención de hacerlo (47).

La realización de pruebas de seguridad contribuye a la detección temprana de vulnerabilidades y la toma de medidas para la disminución de amenazas de ataque, y con ello proveer sistemas de cómputo más seguros y confiables.

A la herramienta desarrollada se le realizaron una serie de pruebas de seguridad mediante el *software* Acunetix, las cuales se presentan a continuación:

- Ataques de inyección SQL.
- *Cross-Site Scripting* (XSS).
- Falsificación de petición (CSRF).



Tabla 14. Resultados de las pruebas aplicadas. Fuente: Elaboración propia.

Tipo	Cantidad	Descripción
Software	1	La versión de PHP presenta vulnerabilidades de seguridad.
Configuración	3	La configuración del servidor de aplicaciones permite la visualización de información sensible del negocio.

Las vulnerabilidades detectadas a partir de la ejecución de las pruebas de seguridad fueron solventadas y se recomienda que se tenga en cuenta la nueva configuración para un futuro despliegue del sistema en su entorno de producción.

3.3.4 Pruebas de usabilidad

La usabilidad es la capacidad de un producto de *software* de ser entendido, aprendido, utilizado y atractivo al usuario, cuando es utilizado bajo condiciones especificadas (Pérez, 2006). Es un atributo intangible del *software*, por lo que es difícil de visualizar, medir y reconocer como un factor determinante de su calidad. Una mayor atención a este elemento contribuiría a incrementar la calidad del producto percibida por el usuario, sin un aumento excesivo en el costo de desarrollo (53).

Las pruebas de usabilidad intentan encontrar problemas de usabilidad del sistema en su interacción con humanos. Según el enfoque tradicional, estas pruebas se aplican sobre el producto *software* para garantizar o determinar si el mismo alcanza un nivel aceptable de usabilidad. Entre los modelos que presenta para evaluar se encuentran los heurísticos y los empíricos. Los primeros implican la participación de expertos especialistas en usabilidad, mientras que los segundos constan de técnicas que implican la participación de los usuarios (53).

Para la realización de las pruebas de usabilidad a la herramienta se definieron los siguientes pasos:

Confección de una lista de especificaciones de usabilidad.

Selección de un conjunto de usuarios.



Especificación de las métricas a evaluar durante las pruebas.

Evaluación de la usabilidad del sistema mediante un *test* de usabilidad.

Para la confección de la lista de especificaciones de usabilidad se toma en consideración algunos de los requisitos funcionales que se identificaron durante el proceso de levantamiento de requisitos de la herramienta.

El proceso de selección de los usuarios que participan en las pruebas se realiza de forma aleatoria. Se seleccionaron 5 individuos de un conjunto de posibles candidatos.

Tabla 15. Candidatos seleccionados para las pruebas de usabilidad. Fuente: Elaboración propia.

No.	Experto	Edad
1	María Yaquelín Fajardo Mosegues	42
2	Roberto Rodríguez Rodríguez	43
3	Ilianys Alfonso Milián	22
4	Reimer Darián Mallesá Romero	23
5	Rachel Vázquez Barrera	18

Tabla 16. Métricas a evaluar en las pruebas de usabilidad. Fuente: Elaboración propia.

Métrica	Descripción	Unidades de medida	Valores
Facilidad de Aprendizaje	Indica qué tan fácil es aprender la funcionalidad básica del sistema.	Cualitativa	Muy Adecuado
			Adecuado
			Inadecuado
Presentación visual apropiada	Indica la correcta selección de colores,	Cualitativa	Muy Adecuado
			Adecuado



	tipos de letra, la disposición de los elementos en una ventana, entre otros aspectos de diseño gráfico.		Inadecuado
--	---	--	------------

Resultados de las pruebas de usabilidad

Luego de seleccionados los individuos de pruebas, se sometieron a un *test* de usabilidad definido para evaluar el Subsistema de Publicidad aplicado a la Plataforma C.U.B.A.

Tabla 17. Resultados de las pruebas de usabilidad. Fuente: Elaboración propia.

Métricas evaluadas	Valores	Frecuencia Absoluta	Frecuencia Relativa
Facilidad de Aprendizaje	Muy Adecuado	13	0.86
	Adecuado	2	0.14
	Inadecuado	0	0
Presentación visual apropiada	Muy Adecuado	11	0.73
	Adecuado	4	0.27
	Inadecuado	0	0

Los resultados obtenidos, tras aplicar las pruebas de usabilidad a la herramienta desarrollada, muestran que, de manera general, la aplicación presenta un grado de usabilidad muy adecuado de acuerdo a las especificaciones requeridas por el cliente. Esto lo corroboran los niveles de satisfacción de los individuos involucrados en las pruebas.



Conclusiones parciales del capítulo

En este capítulo se abordaron aspectos correspondientes a la implementación y validación del Subsistema de Publicidad aplicado a la Plataforma C.U.B.A., llegándose a las siguientes conclusiones:

Mediante la representación de los diagramas de componentes se facilita la comprensión de la estructura general de la herramienta a través de sus componentes físicos.

La implantación del código fuente de la herramienta es mucho más clara y comprensible gracias a la utilización de estándares de codificación asegurando mayor calidad y facilitando el mantenimiento.

La ejecución de pruebas a la herramienta permitió detectar y corregir deficiencias presentes en la solución y ofrecer una aplicación con mayor calidad, seguridad y usabilidad.



CONCLUSIONES

El presente trabajo de diploma da cumplimiento a las tareas de investigación propuestas inicialmente. Se logró que la Plataforma C.U.B.A., pueda brindar servicios de promoción de anuncios relacionados con las necesidades de búsqueda de los usuarios. 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.



RECOMENDACIONES

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

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



BIBLIOGRAFÍA

1. Beneyto, Raul G. Documanía 2.0. [En línea] 2015. <https://documania20.wordpress.com/2013/09/16/cuanta-informacion-se-genera-y-almacena-en-el-mundo/>.
2. Internet Live Stats. [En línea] 2018. <http://www.internetlivestats.com/total-number-of-websites/>.
3. OLIVÁN, , ANTONIO, and ARQUERO AVILÉS, . *Una aproximación al concepto de recuperación de información en el marco de la ciencia de la documentación*. s.l. : Investigación bibliotecológica, 2006. vol. 20, no 41.
4. McDonald, Sharon. *Advances in Information Retrieval*. 2008.
5. Guevara, Yurisander. Juventud Rebelde. [En línea] 22 de Marzo de 2018. <http://www.juventudrebelde.cu/cuba/2018-03-22/zoom-una-herramienta-multiproposito-1>.
6. Pérez, Javiel Fernández. Mesa Redonda. [En línea] 4 de diciembre de 2018. [Citado el: 14 de febrero de 2019.] <http://mesaredonda.cubadebate.cu/mesa-redonda/2018/12/04/todo-lo-que-debe-saber-sobre-el-internet-por-datos-moviles-en-cuba-video/>.
7. Lima, Leidys Milagros Hernández. Radio Reloj. [En línea] 8 de abril de 2016. [Citado el: 14 de febrero de 2019.] <http://www.radioreloj.cu/es/noticias-radio-reloj/sociedad/publicidad-cuba-mitos-realidades/>.
8. Diccionario de la lengua española. [En línea] [Citado el: 18 de febrero de 2019.] <http://lema.rae.es/drae/?val=promoci%C3%B3n>.
9. Diccionario de la Real Academia Española. [En línea] [Citado el: 18 de febrero de 2019.] <http://dle.rae.es/?id=UYYKIUK>.
10. Farber, Patricio Bonta Mario. *199 Preguntas sobre marketing*. s.l. : Editorial Norma,. ISBN 9789580470304..
11. Townsley, Maria. *Publicidad*. s.l. : Thomson,, 2003. ISBN 9789706863454. .



12. Uceda, Mariola García. *Las claves de la publicidad*. s.l. : ESIC Editorial, 2011. ISBN 9788473568012. .
13. The Free Dictionary. [En línea] [Citado el: 24 de febrero de 2019.] <http://es.thefreedictionary.com/anuncio> .
14. Marketing en redes sociales, social media para PYMES | Emprende Pyme. . [En línea] [Citado el: 5 de marzo de 2019.] <http://www.emprendepyme.net/marketing-en-redes-sociales-social-media-para-pymes.html> .
15. Promoción Económica Alcázar de San Juan. [En línea] [Citado el: 5 de marzo de 2019.] <http://www.promocioneconomicaalcazar.es/component/content/article.html?id=129>.
16. Thompson, I. promonegocios.net. [En línea] 6 de junio de 2016. [Citado el: 9 de marzo de 2019.] <http://www.promonegocios.net/mercadotecnia/publicidad-en-internet.html>..
17. El profesional de la información. [En línea] [Citado el: 9 de marzo de 2019.] <http://www.elprofesionaldelainformacion.com/contenidos/2009/enero/09.html> .
18. Why AdWords? - AdWords Help. [En línea] [Citado el: 9 de marzo de 2019.] <https://support.google.com/ads/answer/9712?hl=en> .
19. ¿Qué es Bing Ads? | Aprendizaje para. Bing Ads. [En línea] [Citado el: 15 de marzo de 2019.] <http://advertise.bingads.microsoft.com/es-es/cl/236/formacion/que-es-bing-ads> .
20. Bing Ads reinventan la forma de hacer publicidad online. Microsoft Advertising. [En línea] [Citado el: 15 de marzo de 2019.] <http://advertising.microsoft.com/es/cl/4307/bing-adsreinventan-la-forma-de-hacer-publicidad-online> .
21. Publicación de anuncios. yahoo. [En línea] [Citado el: 15 de marzo de 2019.] <https://policies.yahoo.com/us/es/yahoo/privacy/topics/ad-serving/index.htm> .
22. VALDÉZ, J. L. Enciclopedia y Biblioteca Virtual de las Ciencias Sociales, Económicas y Jurídicas . [En línea] Enciclopedia y Biblioteca Virtual de las Ciencias Sociales, Económicas y Jurídicas. [En línea] 1994. [Citado el: 18 de marzo de 2019.] <http://www.eumed.net/tesis-doctorales/2014/jlcv/software.htm> .



23. SOMMERVILLE, Ian y GALIPIENSO, María Isabel Alfonso. *Ingeniería del software*. s.l. : Pearson Educación, 2005. SBN 9788478290741..
24. ZAMBRANO, R. A. Metodología de la investigación. [En línea] 2013. [Citado el: 18 de marzo de 2019.] <http://es.slideshare.net/Mal3J1Ta/resumen-capitulos1234-del-libro> .
25. Munguía, Enrique. *¿Qué lenguaje de programación debo elegir?* . 7 de diciembre de 2015.
26. Arenas, Jairo. Ventajas y desventajas de los lenguajes de programación web. 11 de agosto de 2015.
27. Otto, Mark y Thornton, Jacob. Bootstrap 3, el manual oficial. [En línea] 2 de diciembre de 2016. [Citado el: 21 de marzo de 2019.] http://librosweb.es/libro/bootstrap_3/.
28. W3C. Cascading Style Sheets. [En línea] [Citado el: 21 de marzo de 2019.] <https://www.w3.org/Style/CSS/>.
29. Wrigth, Tim. *Learning JavaScript: a hands-on guide to the fundamentals of modern JavaScript*. s.l. : Addison-Wesley, 2012.
30. Alcalde, Alejandro. *Los 11 Mejores Frameworks gratuitos para Aplicaciones Web*. s.l. : Granada : s.n., 2012.
31. CodeIgniter. CodeIgniter Web Framework. . [En línea] [Citado el: 22 de marzo de 2019.] <https://www.codeigniter.com/>.
32. Symfony. Symfony PHP Framework. [En línea] [Citado el: 22 de marzo de 2019.] <https://symfony.com>.
33. Rivera, Navarrete Ricardo. *Zend Framework 2 vs Symfony2*. s.l. : Perú : s.n., 2016.
34. JetBrains. [En línea] [Citado el: 25 de marzo de 2019.] https://www.jetbrains.com/help/phpstorm/2018.2/?top&utm_source=product&utm_medium=link&utm_campaign=PS&utm_content=2018.2.
35. Paradigm, Visual. Visual Paradigm for UML - Software design tools for agile software development. [En línea] [Citado el: 25 de marzo de 2019.] <http://www.visual-paradigm.com/features>.



36. Bolton, David. 2015. Why I Choose PostgreSQL Over MySQL/MariaDB. . [En línea] [Citado el: 25 de marzo de 2019.] <http://insights.dice.com/2015/03/19/why-i-choose-postgresql-overmysqlmariadb/>. .
37. Acunetix. Audit your website security with Acunetix Web Vulnerability Scanner. [En línea]. Acunetix. [En línea] 2016. [Citado el: 25 de marzo de 2019.] <http://www.acunetix.com>.
38. foundation-JMeter, The Apache Software. Apache JMeter. [En línea] Apache JMeter, 2016. [Citado el: 25 de marzo de 2019.] <http://jmeter.apache.org>.
39. CANÓS, José H., LETELIER, Patricio y PENADÉS, M^a Carmen. *Metodologías ágiles en el desarrollo de software*. s.l. : Metodologías Ágiles en el Desarrollo de Software, 2003. vol. 1, no 10.
40. 08-Patrones.pdf., 08-Patrones1.ppt -. [En línea] [Citado el: 26 de marzo de 2019.] <http://www.inf.utfsm.cl/~visconti/ili236/Documentos/08-Patrones.pdf>.
41. 06Patrones.pdf., 06Patrones -. [En línea] [Citado el: 26 de marzo de 2019.] <http://siul02.si.ehu.es/~alfredo/iso/06Patrones.pdf> .
42. get.php., Tema 1 Patrones Arquitectónicos -. [En línea] [Citado el: 26 de marzo de 2019.] <http://www.lsi.us.es/docencia/get.php?id=1130> .
43. desarrollo-agil-symfony-2.1.pdf., Desarrollo web ágil con Symfony2 -. [En línea] [Citado el: 26 de marzo de 2019.] <http://sunshine.prod.uci.cu/gridfs/sunshine/books/desarrolloagil-symfony-2.1.pdf> .
44. Pressman, R. S. *Ingeniería de software. s.l.* s.l. : McGraw-Hill Interamericana de España, 2010. ISBN:9786071503145.
45. Systems, Sparx. . Diagrama de Despliegue UML 2. . *Sparx Systems - Tutorial UML 2 - Diagrama de Despliegue*. [En línea] [Citado el: 28 de marzo de 2019.] http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html.
46. (PSR-0), PHP FRAMEWORK INTEROP GROUP. www.phpfig.org. *PSR-0: Autoloading Standard*. . [En línea] [Citado el: 28 de marzo de 2019.] <http://www.php-fig.org/psr/psr-0/>.



47. (PSR-1)., PHP FRAMEWORK INTEROP GROUP. www.phpfig.org. *PSR-1: Basic Coding Standard*. [En línea] [Citado el: 28 de marzo de 2019.] <http://www.php-fig.org/psr/psr-1/>.
48. (PSR-2)., PHP FRAMEWORK INTEROP GROUP. www.phpfig.org. *PSR-2: Coding Style Guide*. [En línea] [Citado el: 28 de marzo de 2019.] <http://www.php-fig.org/psr/psr-2/>.
49. (PSR-4)., PHP FRAMEWORK INTEROP GROUP. www.phpfig.org. *PSR-4: Autoloader*. [En línea] [Citado el: 28 de marzo de 2019.] <http://www.php-fig.org/psr/psr-4/>.
50. MASCHERONI, Maximiliano Agustín, et al. *En: XIV Workshop de Investigadores en Ciencias de la Computación*. s.l. : Calidad de software e ingeniería de usabilidad. , 2012.
51. Olvera, María Dolores. *Sistemas de recuperación de información en la Web: Nuevos servicios de*. s.l. : Granada, 2015.
52. UPREDES. [En línea] 27. de Noviembre de 2015. <http://lupa.upr.edu.cu/acerca-de.html>.
53. *Marketing digital*. Quijije, Elizabeth del Carmen Calderón. julio-septiembre 2016, Revista Contribuciones a la Economía .
54. Ribeiro-Neto, Baeza-Yates and. [aut. libro] Addison Wesley. *Modern information retrieval*. New York : ACM, 1999, pág. 513.