

Universidad de las Ciencias Informáticas

Facultad 1



Generador de Reportes Webmétricos para el motor de búsqueda Orión. V2.0

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

Autor:

Duanny Sánchez Chinaea

Tutores:

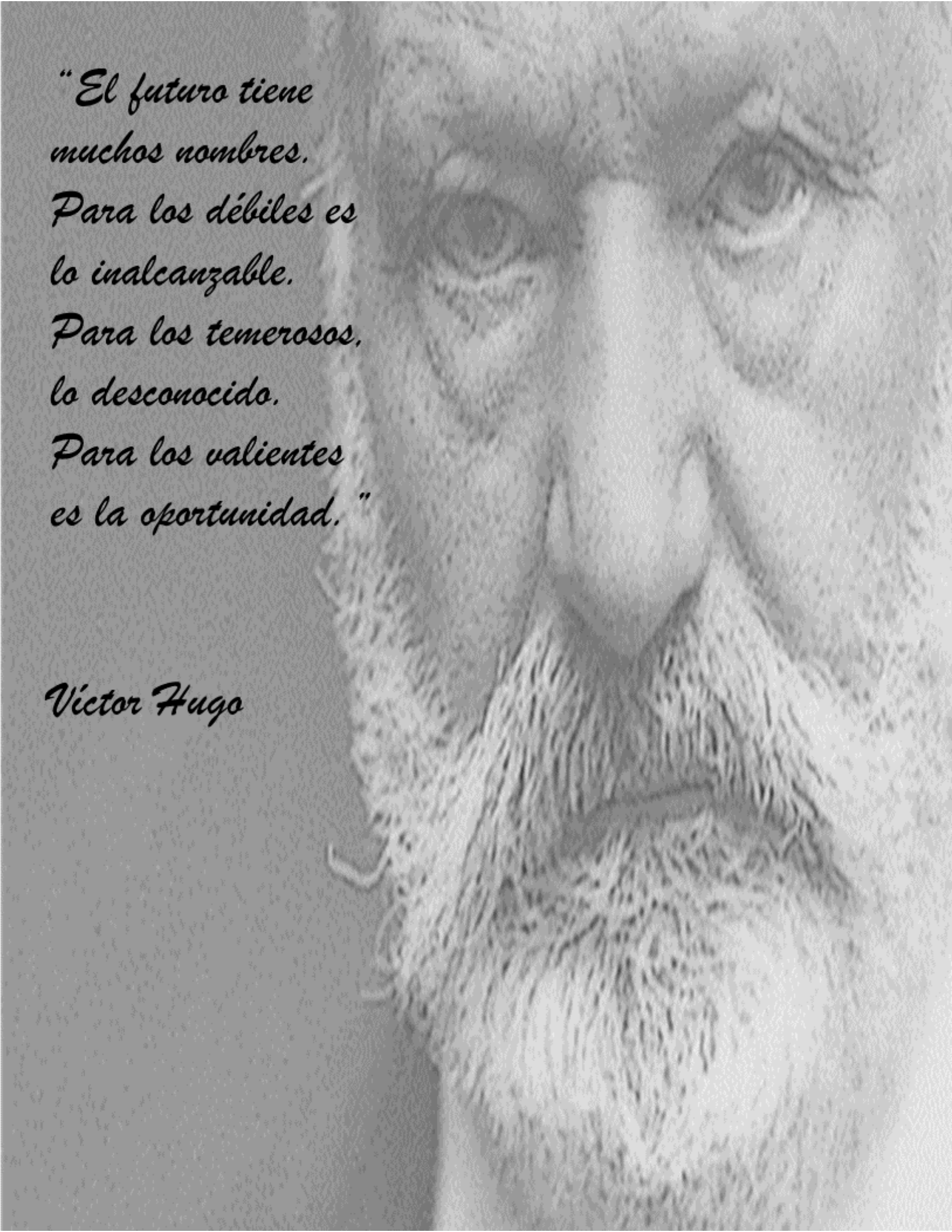
Ing. Disnayle Jorge Chacón

Ing. Paúl Rodríguez Leyva

Ing. Eric Bárbaro Utrera Sust

La Habana, junio del 2016

“Año 58 del triunfo de la Revolución”



*"El futuro tiene
muchos nombres.
Para los débiles es
lo inalcanzable.
Para los temerosos,
lo desconocido.
Para los valientes
es la oportunidad."*

Victor Hugo

Declaración de Autoría

Declaro ser el autor del presente trabajo de diploma y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales del mismo, con carácter exclusivo. Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Firma del Autor

Duanny Sánchez Chinaea

Firma del Tutor

Paúl Rodríguez Leyva

Firma del Tutor

Disnayle Jorge Chacón

Firma del Tutor

Eric Bárbaro Utrera Sust

Agradecimientos

Hace 5 años llegué a esta universidad con un montón de sueños para realizar, hoy se cumplió uno de ellos, graduarme y convertirme en un profesional. Agradezco a todas las personas que de una forma u otra han hecho posible este sueño. Especialmente a dos seres de los que me sería imposible hablar por separado, pues han dedicado toda su vida para darme una formación de la cual me siento orgulloso, dos seres que han sido mis pilares y ejemplo a seguir, a mis padres, gracias por ser siempre mis mejores maestros y por su apoyo incondicional.

A toda mi familia por su apoyo y por haber estado siempre presente para mí.

A mis tutores: Paul, Eric y en especial a Disnayle por su maravillosa dedicación en todo momento al desarrollo del trabajo de diploma.

A mis profesores por todo el tiempo dedicado.

A mi novia por soportar las infinitas y repetitivas conversaciones sobre la tesis.

A todos aquellos especialistas del centro CIDI que con su ayuda contribuyeron a la causa.

Resumen

Hoy día existen sistemas que permiten agilizar el proceso de búsqueda de información en la red. Como alternativa en la Universidad de las Ciencias Informáticas (UCI) se encuentra aún en proceso de desarrollo el motor de búsqueda Orión, el cual posibilita la recuperación de información publicada en la intranet cubana. Este buscador registra y acumula una serie de estadísticas que resultan de gran utilidad, ya que brindan informaciones que pueden ser utilizadas en aras de tributar a la optimización de cada uno de los sitios Web indexados; trayendo consigo un mejor posicionamiento Web y una mejor disponibilidad de la información o los servicios brindados. Durante esta investigación se analizan diversos sistemas que ofrecen reportes webmétricos, atendiendo las estadísticas, clasificaciones, funcionalidades y principales requisitos que poseen, para posteriormente definir los factores a tener en cuenta para la solución propuesta, en la cual se desarrolla un módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión que pretende contribuir a la toma de decisiones de los *webmasters*.

La implementación de la propuesta de solución fue guiada por la metodología AUP¹ (UCI), se seleccionaron como principales tecnologías: Solr como mecanismo de indexación, Symfony como marco de trabajo PHP y Visual Paradigm como herramienta para el modelado. El módulo implementado posee características y funcionalidades que posibilitan a los *webmasters* conocer datos estadísticos específicos del conjunto de sitios indexados en Solr, posibilitando la búsqueda agilizada a través de filtros que permiten lograr resultados más exactos disminuyendo el tiempo en hallar la información que precisan.

Palabras clave: *estadística, módulo, reporte, Web, Webmetría.*

¹ Agile Unified Process por sus siglas en inglés.

Índice

Introducción	1
Capítulo 1: Fundamentación teórica del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión.	7
1.1 Introducción.....	7
1.2 Marco conceptual	7
1.3 Sistemas homólogos	12
1.4 Herramientas, lenguajes y tecnologías de desarrollo.....	14
1.5 Conclusiones del capítulo	31
Capítulo 2: Diseño del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión	32
2.1 Introducción.....	32
2.2 Descripción de la propuesta de solución	32
2.3 Modelado del dominio	32
2.4 Especificación de requisitos de <i>software</i>	33
2.5 Análisis y Diseño	38
2.6 Conclusiones del capítulo	45
Capítulo 3: Implementación y prueba del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión	46
3.1 Introducción.....	46
3.2 Modelo de implementación.....	46
3.3 Diagrama de componentes.....	46
3.4 Estándares de codificación utilizados	50
3.5 Validación del módulo	51
3.6 Conclusiones del capítulo	61
Conclusiones	62
Recomendaciones	63
Referencias Bibliográficas	64

Introducción

La revolución tecnológica centrada en torno a las tecnologías de la información, ha modificado las bases de la sociedad a un ritmo acelerado. El impacto causado por las Tecnologías de la Informática y las Comunicaciones (TIC) ha revolucionado la manera de hacer negocios, la comunicación entre las personas y la interacción entre las empresas. Las nuevas tecnologías de la información han ido integrando al mundo en redes globales donde la información es la materia prima y los usuarios sus grandes consumidores. Todas estas tecnologías giran en torno al desarrollo del Internet o la “Red de Redes” como también se le conoce, la cual desde su surgimiento ha superado ampliamente cualquier previsión y constituye una verdadera revolución en la sociedad moderna (Camiño, 2003), (Buigas, 2002).

Esta red se ha transformado en un verdadero pilar de las comunicaciones, el entretenimiento y el comercio en todos los rincones del planeta. Debido a que en su interior se encuentran ingentes cantidades de información las opciones de búsqueda son prácticamente infinitas. Por tales motivos y en aras de facilitar la obtención de dicha información deseada, contenida en la variedad de sitios Web existentes, surgieron los buscadores Web, los cuales facilitan a los usuarios realizar las búsquedas deseadas.

“Los buscadores, motores de navegación o motores de búsqueda (MB), son aquellos programas o herramientas interactivas que facilitan la búsqueda y recuperación de información en Internet. Los motores de búsqueda ofrecen formularios para introducir los datos mediante una interfaz de fácil comprensión para el usuario, el cual teclea una palabra clave o frase y recupera una lista de recursos que se corresponden con el criterio indicado” (Camiño, 2003).

Muchas de las nuevas visitas hechas a un sitio llegan por los resultados obtenidos en una consulta realizada a un buscador, lo cual ocasiona que gran parte del tráfico de un sitio Web pueda proceder de los motores de búsqueda. El número de visitas que recibe un sitio no depende únicamente de las bases de datos a las cuales se encuentre indexadas. Uno de los principales factores que influyen en este aspecto es el posicionamiento Web que estas ocupen ya que según (Codina, 2008) solo una pequeña parte los usuarios llega a interesarse por la segunda página de resultados ofrecida por un buscador, además casi la totalidad de estos usuarios detienen su búsqueda en la tercera página. A partir de estos datos se puede inferir de alguna medida la importancia que tiene la posición de un sitio Web.

Usualmente los *webmasters* necesitan de reportes estadísticos muy específicos donde se cuenta con una colección de datos especializados de toda la información referente al sitio, esto los ayuda en gran medida

a optimizar las estrategias actuales de posicionamiento dentro de la Internet, lo cual por su parte influye directamente en las decisiones tomadas por los usuarios a la hora de consultar las respuestas ofrecidas por el buscador (Pixelar, 2015).

En Cuba la Web se encuentra aún dando sus primeros pasos, pero no por esto debe considerarse un trabajo pequeño, aunque aún existen muchos puntos que pudieran ser pulidos, el desarrollo del dominio .cu continua su ritmo creciente. Según el sitio Web oficial del Centro Cubano de Información de Red (CUBANIC) hasta el 20 de noviembre del 2015 se contaba aproximadamente con unos 6706 dominios registrados bajo .cu y con ello una amplia gama de información disponible, de ahí la importancia de realizar una búsqueda eficiente de los datos que se desean analizar.

Incluir los sitios cubanos dentro de los sistemas de búsqueda más utilizados del mundo es un paso fundamental para aumentar la cantidad de visitas que estos reciben. Según (Elizalde, 2015) algunos buscadores como Google, Bing y Yahoo, por solo mencionar tres de los más populares incluyen la indexación de páginas cubanas dentro de sus bases de datos aumentando considerablemente el número de visitas recibidas.

En la Universidad de las Ciencias Informáticas (UCI) se encuentra aún en proceso de desarrollo el motor de búsqueda Orión, el mismo posibilita la recuperación de información publicada en la intranet cubana. Este buscador registra y acumula una serie de estadísticas, dentro de las que se pueden mencionar: cifras de posicionamiento asociadas a los sitios o páginas Web (Page Rank), la cantidad de enlaces entrantes y salientes de las páginas almacenadas, así como determinados datos estadísticos sobre los sitios existentes en la colección de información indexada en la base de datos del motor.

Versiones anteriores del motor incorporaban un módulo para la creación de reportes webmétricos del cual actualmente no se conserva el código fuente. Dicho módulo no logró satisfacer en su totalidad las necesidades funcionales para las cuales fue ideado, si a esto se suma que no coincide con las pautas de diseño que posee hoy Orión, ya que la versión actual de Orión debe permitir la instalación del sistema de forma distribuida entonces se puede determinar de que ese módulo no representa una solución a las necesidades actuales, ocasionando que nuevamente los *webmasters* no cuenten con la información requerida.

Aquellos sitios Web que cuentan con problemas como enlaces rotos que no conduzcan a otra dirección válida tienden a ser rechazados por el usuario regular, este aspecto impacta negativamente al correcto desempeño del objetivo para el cual dicho sitio haya sido creado, ya sea económico, político o social. Por otra parte, la no inclusión de palabras claves se deriva en un mal posicionamiento del sitio dentro del indexado realizado por los motores de búsqueda, provocando que el usuario no encuentre fácilmente la información y por tanto que el sitio en cuestión no sea visitado con frecuencia. La excesiva inclusión de contenidos flash tiende a generar una demora excesiva e innecesaria de la respuesta del sitio, por lo cual se deben eliminar o al menos utilizar discretamente.

Sobre la base de los elementos expuestos anteriormente se formula el siguiente **problema de la investigación**. ¿Cómo contribuir a la toma de decisiones de los *webmasters* a partir de la información recolectada por el buscador Orión?

En consecuencia, con lo anterior, se determinó como **objeto de estudio** el proceso para la generación de reportes estadísticos, enmarcado en el **campo de acción** generación de reportes webmétricos.

Se define como **objetivo general** de la investigación: Desarrollar un módulo que genere reportes webmétricos haciendo uso de las pautas de diseño actuales para el motor de búsqueda Orión que contribuya a la toma de decisiones de los *webmasters*.

Con el propósito de darle cumplimiento al objetivo general antes mencionado, el mismo se define en los siguientes **objetivos específicos**.

1. Definir el marco teórico y estado del arte respecto a las tecnologías actuales para el desarrollo de aplicaciones Web para la visualización de datos estadísticos.
2. Diseñar un módulo Generador de Reportes Webmétricos para Orión que genere los reportes estadísticos específicos que los *webmasters* necesitan a partir de la información recolectada por el buscador (colección de información personalizada).
3. Implementar un módulo Generador de Reportes Webmétricos para Orión que permita obtener estadísticas de la Web y generar reportes estadísticos específicos.
4. Validar el correcto funcionamiento del módulo Generador de Reportes Webmétricos para Orión.

La presente investigación tiene como **idea a defender** que el desarrollo del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión facilitará la toma de decisiones de los *webmasters*.

Para dar cumplimiento a los objetivos anteriormente descritos se plantean las **siguientes tareas de investigación**:

1. Realización de un levantamiento bibliográfico y webgráfico² para estudiar los principales elementos teóricos, conceptos que permitan analizar el estado actual del desarrollo de tecnologías para el desarrollo de servicios Web y elaborar un marco teórico.
2. Identificación de los servicios Web que ofrecen los generadores de reportes webmétricos en Internet.
3. Identificación de los servicios Web que debe ofrecer el generador de reportes webmétricos para Orión.
4. Definición de la metodología a utilizar en el desarrollo del generador de reportes webmétricos.
5. Identificación de necesidades de información, requisitos funcionales y no funcionales
6. Documentación de los artefactos generados durante los procesos de análisis, diseño e implementación.
7. Implementación de las funcionalidades que debe ofrecer el generador de reportes webmétricos.
8. Validación de la propuesta de solución.

Resultados esperados:

Se pretende que al concluir la nueva versión del módulo Generador de Reportes Webmétricos el motor de búsqueda Orión cuente con una serie de nuevas funcionalidades que satisfaga las necesidades básicas de información de los *webmasters* para la toma de decisiones, partiendo de los datos estadísticos sobre los sitios existentes en la colección de información indexada en la base de datos del motor.

Para responder al desarrollo de la investigación se emplearán los **métodos investigativos: teóricos** que se obtienen de la interacción directa entre el sujeto y el objeto de la investigación y **empíricos** que se basan en el trabajo con conceptos, categorías, leyes, teorías, entre otros. Dentro de los métodos empíricos se

² Relativo a **webgrafía**, es un listado o referencia bibliográfica a modo de bibliografía de recursos electrónicos, es decir, de enlaces a páginas web, sitios web, blogs o portales de Internet. Se trata de un neologismo creado del término en inglés *webography* (Fundéu, 2015).

aplicará la “**Entrevista**” para obtener información que tribute a la definición de los requisitos del módulo; así como para precisar el problema de la investigación, los problemas existentes y los servicios que brinda actualmente el módulo.

La entrevista se realizó a una muestra de especialistas del centro CIDI que han colaborado con el proyecto Orión, teniendo en cuenta las preguntas definidas en el documento de entrevista. (Ver anexo # 1).

Las respuestas arrojadas por los especialistas entrevistados dieron lugar a las siguientes conclusiones por parte del autor.

Las plantillas utilizadas en las vistas para módulo que se propone como solución deben estar regidas a la actual herencia de tres niveles que posee el buscador Orión, garantizando así la uniformidad de todo el sistema. Los principales indicadores webmétricos recolectados por Solr a tener en cuenta son: las palabras claves, los contenidos MIME³, los enlaces tanto entrantes como salientes y los hosts. Algunas funcionalidades como el exportado a formato .pdf y otras, incluidas dentro de la anterior versión del Módulo Generador de Reportes Webmétricos aún resultan de utilidad, por lo que se aconseja su reutilización.

Se aplicará también la “**Revisión documental**”, que se pondrá de manifiesto en las revisiones a la bibliografía existente y a la hora de consultar la información en sitios de interés nacional e internacional, para apoyar la realización de las tareas definidas en la investigación sobre los reportes webmétricos.

De los métodos teóricos se usará el “**Analítico-Sintético**” cuando a partir de condiciones específicas se llegue a ideas generales del tema, posibilitando definir los conceptos que se manejarán durante el proceso de investigación y las principales características de las herramientas, lenguajes y metodología a utilizar en el desarrollo de la propuesta de solución y el “**Modelado**”, el cual posibilita una mejor comprensión de los elementos estudiados mediante el uso de diagramas y modelos más simples, para la implementación de la propuesta de solución.

El presente Trabajo de Diploma se encuentra estructurado por tres capítulos manteniendo la lógica siguiente:

³ Referente a *Multipurpose Internet Mail Extensions*, serie de especificaciones dirigidas al intercambio transparente de todo tipo de archivos a través de Internet (iana.org, 2016) .

Capítulo 1: Fundamentación Teórica del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión: Contiene la descripción del estado del arte para este tipo de aplicaciones generadoras de reportes, así como el análisis de las tecnologías que se utilizarán durante las fases de diseño, desarrollo e implementación del módulo

Capítulo 2: Diseño del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión: Se abordan las temáticas relacionadas con el dominio, la caracterización del sistema a desarrollar y los requerimientos planteados. Se formula la propuesta de la solución a implementar, incluyendo todos los diagramas necesarios para ello.

Capítulo 3: Implementación y prueba del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión: Aborda la implementación y los resultados obtenidos de las pruebas realizadas a la solución propuesta, incluye las pantallas principales de las interfaces de usuario de la aplicación.

Capítulo 1: Fundamentación teórica del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión.

1.1 Introducción

El presente capítulo tiene como objetivo tratar los conceptos y aspectos más significativos que se relacionan con la generación de reportes webmétricos, abordados en diferentes fuentes bibliográficas para profundizar y ampliar el estudio del arte. Se realiza un análisis a diversos sistemas informáticos referentes al tema y se detallan las tecnologías, lenguajes y herramientas utilizadas en la implementación de la propuesta de solución.

1.2 Marco conceptual

Para lograr una mejor comprensión de la investigación, se abordan un conjunto de conceptos necesarios y están estrechamente relacionados con el dominio del problema.

Información: Según el Doctor Idalberto Chiavenato, "es un conjunto de datos con un significado, o sea, que reduce la incertidumbre o que aumenta el conocimiento de algo. En verdad, la información es un mensaje con significado en un determinado contexto, disponible para uso inmediato y que proporciona orientación a las acciones por el hecho de reducir el margen de incertidumbre con respecto a las decisiones" (Chiavenato, 2006).

Según los Doctores Michael Czinkota y Masaaki Kotabe, la información "consiste en datos seleccionados y ordenados con un propósito específico (Czinkota y autores, 2001).

Para el Doctor O. C. Ferrell, jefe del Departamento de Marketing y profesor de la universidad estatal de Colorado, la información "comprende los datos y conocimientos que se usan en la toma de decisiones" (Ferrel y autores, 2010).

Todos los autores coinciden en que la información está constituida por un grupo de datos ya supervisados y ordenados, que sirven para construir un mensaje o dar respuesta a una petición, lo cual permite resolver problemas y tomar decisiones, sin embargo, la correcta clasificación y recuperación de la información acorde a su nivel de utilidad puede constituir un factor determinante en el resultado final obtenido.

Recuperación de la información (RI): Es proceso donde se accede a una información previamente almacenada. En la informática es el área de la ciencia y la tecnología que trata la adquisición, representación, almacenamiento, organización y acceso a elementos de información. Desde un punto de vista práctico, dada una necesidad de información del usuario, un proceso de RI produce como salida un conjunto de documentos cuyo contenido satisface potencialmente dicha necesidad. Su objetivo no es la de devolver la información deseada por el usuario, sino únicamente la de indicar que documentos son potencialmente relevantes para dicha necesidad de información (Oliván y autores, 2006).

En la actualidad la recuperación de información "convencional" significa la búsqueda online en bases de datos electrónicas, de forma interactiva y en tiempo real. Normalmente, esto implica que el usuario construye una estrategia de búsqueda usando términos con distintas relaciones lógicas (booleanas) y que el programa de búsqueda simplemente divide la base de datos en dos conjuntos: elementos recuperados y elementos no recuperados (Lancaster, 2001). Muchas de estas búsquedas son realizadas a través de los llamados Buscadores Web, los cuales mediante una palabra clave o frase recuperan una lista de recursos que se corresponden con el criterio indicado por el usuario.

Buscadores Web: Principalmente existen tres tipos de buscadores. Según la forma en que estos procesan la información se clasifican en: Directorios o Índices, los Metabuscaores o los denominados Motores de búsqueda (robots).

Los directorios se basan fundamentalmente sobre el trabajo realizado por humanos. Presentan la información sobre las Webs (Real Academia Española, 2010) registradas como si fuese un catálogo, ordenadas en categorías y subcategorías. Solo registran algunos de los datos de nuestra página como el título y la descripción del sitio Web, entre otros. Estos datos suministrados, son revisados por operadores humanos. Al buscar en algunos de ellos, más que en función de los contenidos de la página, los resultados son mostrados según la temática de la Web (Jaume, 2007).

Por su parte los Metabuscaores en realidad, no son buscadores, no disponen de una base de datos propia, sino que lo que hacen, es realizar consultas sobre las bases de datos de auténticos buscadores, robots y directorios, analizar los resultados, y presentar los suyos propios, reordenándolos según su criterio. Su principal deficiencia es que, por lo general, no permiten realizar búsquedas avanzadas como en un motor de búsqueda, con lo que las consultas tienen que ser generales y no se puede buscar en profundidad (González, 2002).

Robots o motores de búsqueda (MB): Se le conocen también como *crawlers* o *spiders*. La mayoría de grandes buscadores internacionales que se utilizan actualmente son de este tipo. Todos estos buscadores, tienen uno o varios robots encargados de ir visitando todos los sitios Web existentes. Fundamentalmente extraen información del texto que en ellos aparece y de otros parámetros como el título de las páginas. Al realizar una consulta sobre este tipo de buscadores, ellos consultan su base de datos, con la información que han recogido de las páginas y presentan los resultados acordes a los algoritmos que utilizan cada uno de ellos, principalmente utilizados para ordenar los resultados en función de su relevancia (González, 2002).

Cada cierto tiempo, los robots revisan las Webs, para actualizar los contenidos de su base de datos. La calidad de los resultados obtenidos por cada buscador vendrá determinada por sus parámetros de indexación, extracción de información de cada Web y algoritmos de búsqueda, análisis contextual, etc., que cada uno de ellos haya desarrollado.

Un MB no constituye un ente aislado, consiste en un conjunto de “partes” que trabajan en cooperación. Generalmente se encuentran tres fundamentales, un sistema de exploración o rastreador (en inglés: *spider* o *crawler*) y un sistema de recuperación de información (en inglés: *information retrieval system*) y una interfaz Web que es la parte que el usuario es capaz de ver y con la que puede interactuar.

El rastreador está diseñado para navegar por la WWW⁴ a partir de un listado de direcciones de manera sistemática y organizada, su función es analizar el contenido de las páginas Web que encuentra. Descarga esos contenidos, en su mayoría páginas Web, aunque también otros como presentaciones, archivos de imágenes, audio, etc. en el almacén o repositorio de documentos del MB. A partir de las páginas descargadas, el rastreador obtiene nuevas URL⁵ que añade a su lista de URL pendientes de analizar (Rueda, 2012).

El sistema de recuperación de información se encuentra a su vez dividido en dos subsistemas, el sistema de indexación y el sistema de consultas. El primero recibe las páginas recuperadas por el *spider*, extrae una representación interna de la misma y la almacena en forma de índices en una base de datos, facilitando así la recuperación de la información por el motor de búsqueda. Por su parte, el motor de búsqueda recibe la consulta proporcionada por un usuario, que consiste en la introducción de un grupo de palabras o frases

⁴ Siglas del idioma inglés correspondiente a **Word Wide Web**. Es un sistema hipermedia que integra en una interface común a todos los recursos existentes en la red para su acceso en forma organizada y normalizada, cubre todos los recursos del mundo en forma hipermedia o hipertexto. (Andalia, 2004).

⁵ Siglas del idioma inglés correspondiente a **Uniform Resource Locator (Localizador Uniforme de Recursos)**. Se trata de la secuencia de caracteres que sigue un estándar y que permite denominar recursos dentro del entorno de Internet para que puedan ser localizados. (Castro, 2015)

claves sobre la información requerida. Estas palabras claves son transformadas por el sistema de consultas en un conjunto de incógnitas entendibles por el sistema y las que serán utilizadas por el subsistema de evaluación para devolver los documentos existentes en la base de datos, otorgando un orden de relevancia a dichos documentos en correspondencia con la consulta originalmente introducida por el usuario.

Independientemente de la estructura y el funcionamiento interno de un MB, los resultados obtenidos al efectuar consultas sobre ellos dependen del estudio de los aspectos cuantitativos de la construcción y empleo de recursos, estructuras y tecnologías disponibles en la Web. A esta rama de análisis se le conoce como Webmetría.

Un aspecto importante en el ámbito de las Ciencias de la Información es el uso de indicadores métricos, los cuáles al ser analizados permitirán comprender, controlar y predecir hasta cierto grado un fenómeno en particular, para tener un mejor entendimiento de que es la Webmetría, a continuación, se muestran variados conceptos empleados por diferentes autores para referirse a este término.

La **Webmetría** “estudia la relación entre los elementos principales de la Web (comparable a una red neuronal conformada por nodos y conexiones: los nodos se corresponden con los dominios, servidores y páginas que componen la Web y las conexiones con los enlaces que se establecen entre los mismos dando lugar a una matriz de conexiones que abarca todo el entorno WWW)” (Herrera, 2010).

Es el estudio de los aspectos cuantitativos de la construcción y uso de los recursos de información, estructuras y tecnologías de una parte concreta de Internet, por regla general a una Web o portal, desde perspectivas bibliométricas e infométricas (Björneborn, 2004. u propósito es analizar y extraer ciertos elementos de la Web y básicamente sustrae la información contenida en la URL de una página o sitio Web, pueden localizar la información completa o dividirla en diversos componentes y realizar la extracción de información muy precisa como pueden ser imágenes, archivos multimedia, documentos, etc (Sánchez, 2015).

Después de haber analizado los conceptos anteriores se puede llegar a la conclusión que los reportes webmétricos son informes que organizan y muestran las estadísticas que han sido extraídas de la Web.

Indicadores webmétricos: Para poder realizar los reportes webmétricos es necesario utilizar una serie de indicadores webmétricos los cuales ayudan a una mejor selección de la información a mostrar de la Web solicitada por los usuarios. Los indicadores son elementos básicos que brindan información cualitativa o

cuantitativa del control de cómo funciona una actividad, pues hacen referencia a parámetros estables que sirven de magnitud de comprobación del funcionamiento de esta. Según (Martínez, 2006) los indicadores deben ser: accesibles, fáciles de identificar y recopilar; pertinentes, para lo que se desee medir; fieles, que informen con fidelidad de las condiciones de los datos que se recogen; objetivos, no ambiguos en su interpretación; precisos para la acción que se quiere estimar, unívocos, únicamente para parámetros exclusivos de lo que se mide y sensibles, que permitan recoger y estimar variaciones de aquello a lo que referencian.

Reporte: Según la RAE⁶ en el idioma español el concepto de reporte se utiliza para designar, ya sea a un informe o a una noticia. Es un término de uso habitual en diversas áreas como de una empresa, la ciencia, la investigación y en los medios de comunicación masiva (Revelo, 2014).

Entonces, el reporte es aquel documento que se utilizará cuando se quiera informar o dar noticia acerca de una determinada cuestión, demanda de una preparación exhaustiva, es decir, su autor deberá reunir toda la información concerniente al tema que lo ocupa y deberá aportar informaciones y explicaciones que le permitan al destinatario comprenderlo e informarse al respecto.

En el ámbito de la informática, los reportes son informes que organizan y exhiben la información contenida en una base de datos. Su función es aplicar un formato determinado a los datos para mostrarlos por medio de un diseño atractivo y que sea fácil de interpretar por los usuarios.

Contienen la descripción de las acciones interactivas con el sistema informático, la captación de la información, las operaciones realizadas por el mismo, acciones ante errores y situaciones anormales, seguridad y protección de los recursos informativos y consulta de información de resultados (Encinoza, 2012).

Estadística: Muy amplio es el espectro de autores que han dado su definición de estadística. A continuación, quedan plasmados algunos de estos autores y sus respectivos conceptos, extraídos del Manual de Estadísticas escrito por David Ruiz Muñoz, Profesor Departamento Economía y Empresa de la universidad Pablo de Olavide, en el 2004:

⁶ Real Academia Española

Para el profesor Enrique Chacón de la Universidad de Deusto esta se define como “la ciencia que tiene por objeto el estudio cuantitativo de los colectivos”.

De otra manera el profesor del departamento de Didácticas de las Matemáticas y Ciencias Experimentales del Centro de Enseñanza Superior Óscar Vázquez Mínguez concibe la Estadística como “La ciencia que tiene por objeto aplicar las leyes de la cantidad a los hechos sociales para medir su intensidad, deducir las leyes que los rigen y hacer su predicción próxima”.

Otros la definen como la expresión cuantitativa del conocimiento dispuesta en forma adecuada para el análisis y la investigación. Según (Muñoz, 2004) la estadística en general se define como: “La ciencia que trata de la recopilación, organización, presentación, análisis e interpretación de datos numéricos con el fin de realizar una toma de decisiones más efectiva”.

Acorde al objetivo seguido por este trabajo se tomará como referencia el concepto expresado por David Ruiz Muñoz por ser el que más se corresponde con la investigación.

1.3 Sistemas homólogos

Internacionales

SeoGratis: Realiza un exhaustivo análisis SEO (*Search Engine Optimization*) de páginas Web a través de varios parámetros. Es un servicio gratuito. Provee un informe detallado que permite tener una idea completa del sitio Web. Los informes contienen una explicación detallada de los problemas identificados, consejos para hacer frente a estos problemas, la optimización y el desarrollo del recurso. La auditoría Web permite averiguar que impide que una página Web se encuentre en los primeros resultados de búsqueda.

Parámetros medidos

Tabla 1: Parámetros medidos por SeoGratis para el análisis SEO.

Parámetro	Descripción
URL	El enlace debe ser corto y preciso e incluir las palabras claves que describen el contenido de la página.
Título	El título de la página debe tener entre 10-70 caracteres, incluidos los espacios. Palabra clave-1-palabra clave 2 marca comercial o Marca comercial palabra clave 1 y palabra clave 2
Meta descripción	Meta <i>keywords</i> son palabras o frases que hacen referencia al contenido del sitio. Si bien hoy en día los buscadores no tienen en cuenta esta etiqueta, son de utilidad para el cambio de enlaces relevantes. Es recomendable usar 7-12 palabras o frases claves.

Títulos	Se recomienda incluir palabras claves en los títulos. El título inicial <H1> debe incluir las mejores palabras claves. Usando sólo un título <H1> por página se fortalece la optimización SEO.
Imágenes	Se sugiere agregar un texto alternativo (ALT) a las imágenes para facilitar que aparezcan en los motores de búsqueda.
Texto/código	El porcentaje de texto/código de una página es utilizado por los motores de búsqueda para calcular la relevancia de una página Web. Cuando este ratio es muy bajo, los buscadores penalizan la página, por esta razón, es muy beneficioso incluir toda información que se considera importante.
Frames	Es difícil para los motores de búsqueda indexar páginas con <i>frames</i> (marcos). Se debe evitar el uso de marcos para optimizar el sitio Web.
Flash	Usar Flash es bueno solo en determinadas situaciones, como por ejemplo para mejorar el aspecto de una página Web. Para un mejor posicionamiento SEO no se debe utilizar flash, hoy en día las páginas con Flash no pueden ser indexadas por los motores de búsqueda. AJAX entra en la misma categoría que Flash Player.
Favicon	El favicon es un ícono, una pequeña imagen que aparece en algunos navegadores junto a la dirección de una página Web. El favicon le permite al usuario encontrar más rápido su página Web cuando navega por las pestañas. También sirve como referencia cuando se agrega la página Web a los marcadores.

WooRank

Es una aplicación Web con la que puede generar informes de *websites* en los que se incluyen datos significativos sobre el estado de un sitio Web, así como una lista de consejos, recomendaciones a implementar con tal de optimizar la presencia *online* de dichos *websites*. Su objetivo es ayudar a negocios, *marketers* y otros servicios online a conseguir mejores posicionamientos en SERP⁷, convertir más visitas en clientes, facilitar la monitorización de sus competidores y obtener un mejor ROI⁸ en su presupuesto destinado a *marketing* digital.

Nacionales

Módulo de reporte de Airesproxy

Permite el reporte de datos estadísticos a partir del análisis de los *logs* de un servidor proxy. Con el uso de técnicas de minería de datos, inteligencia artificial y estadística el módulo posibilita tanto a administradores de red como a directivos; tener datos reales sobre las actividades de los usuarios en la red. Su interfaz

⁷ Es el acrónimo de Search Engine Result Page, es decir: Página de Resultados de un Motor de Búsqueda

⁸ Siglas en inglés de Return On Investment y es un porcentaje que se calcula en función de la inversión y los beneficios obtenidos, para obtener el ratio de retorno de inversión (Alvarez, 2009).

permite visualizar de forma dinámica en distintos tipos de gráficas la información estadística generada. Permite personalizar diferentes tipos de reportes para usuarios y administradores (Alfonzo, 2012).

Los sistemas analizados anteriormente no constituyen una solución que pueda adaptarse al motor de búsqueda Orión ya que **WooRank** es una herramienta privativa, lo que impide cualquier reproducción parcial o total. En cambio, **SeoGratis** es un sistema que realizan reportes sin hacer uso de la colección de información almacenada en la base de datos de los motores de búsqueda por lo que no cumple con los objetivos del módulo que se va a desarrollar el cual tiene que consultar la base de datos del buscador Orión para generar los reportes. Además, ninguno de los sistemas mencionados anteriormente son lo totalmente modulares para que sean reutilizables. En cambio, el módulo de reportes de Airesproxy no puede ser utilizado porque está desarrollado para realizar los reportes sobre las conexiones y el funcionamiento de un servidor proxy, obteniendo los datos desde los *logs* que se generan en el servidor.

1.4 Herramientas, lenguajes y tecnologías de desarrollo

Tecnologías para generar gráficos

En esta sección se hará una revisión de varias bibliotecas para la inclusión de gráficos en el módulo. Se utilizarán como criterios de comparación los tipos de gráficos que puedan generar. Además de esto, también se tendrá en cuenta la licencia y disponibilidad del código fuente; porque esto puede afectar la decisión de usar una herramienta u otra. Finalmente, las licencias, simplemente se señalarán. Para dar una evaluación final se podrá tener en cuenta si es una licencia libre y permisiva (MIT, BSD y Apache), una licencia libre y no-permisiva (GNU GPL) o no es libre o los términos de uso la prohíben en algunas circunstancias.

La evaluación de cada uno de los criterios estará dada por los siguientes valores:

Sí, si la biblioteca soporta directamente el criterio. En el caso de los gráficos o elementos visuales, esto indica que la librería los soporta.

No, si la biblioteca no incluye soporte directo al criterio y tendría que ser modificada significativamente para añadirlo.

Indirecto, si la biblioteca no incluye el soporte directamente, pero mediante el uso de sus primitivas se puede lograr el criterio deseado.

Biblioteca Highcharts

Highcharts según (Shahid, 2014) es una librería escrita en JavaScript que permite construir gráficos interactivos para proyectos Web. Tiene una gran cantidad de demos que se pueden ver y editar en JSFiddle. Posee algunas características de gran alcance, como el no requerir ninguna configuración o requerimiento en particular en el servidor Web para ejecutar el script, como tampoco tecnologías de scripts del lado del servidor, de este modo incluso puede funcionar en páginas HTML estáticas y está basada en tecnologías de navegación nativas, no necesita *plugins* y es compatible con todos los navegadores modernos incluyendo sus versiones móviles y el viejo Internet Explorer 6. Es de uso libre para organizaciones no comerciales, para el resto de sus clientes ofrecen varios planes. Highcharts es una librería *Open Source*, puedes ver su código fuente en GitHub.

Google Charts API

Herramienta que permite la creación de gráficas en forma de imágenes .png⁹ las cuales pueden ser utilizadas en las aplicaciones Web. Su uso es muy sencillo, mediante peticiones http a una determinada URL (<http://chart.apis.google.com>) y con unos determinados parámetros será devuelta una imagen. Actualmente Google Chart tiene definidos 9 tipos de gráficos: Gráfico de líneas, Palabra gráfica, Gráfico de barras, Gráfico circular, Diagrama de Venn, Gráfico de puntos, Gráfico de radar, Mapa y Google-o-meter. Su uso de HTML5 / SVG ofrece compatibilidad entre navegadores (adoptando VML para versiones anteriores de IE) y la portabilidad entre plataformas de iOS y las nuevas versiones de Android. No se necesitan *plugins* adicionales. Ofrece diversos parámetros de personalización, lo que garantiza la integración con la apariencia del sitio Web que se desea construir (Google Developers, 2015).

Biblioteca pChart

Desarrollada en PHP, con la que se pueden generar gráficos estadísticos con una alta definición en la representación de los datos. Está liberada bajo una licencia GNU GPL y se puede acceder al código fuente. Tiene incorporado un módulo de caché que optimiza considerablemente la generación de los gráficos. Presenta una amplia gama de gráficos como son los de barras: apiladas, agrupadas y superpuestas, gráficos de línea, pastel, curvas, puntos y burbujas y soporte indirecto a los gráficos mixtos, la generación

⁹ Portable Network Graphics por sus siglas en inglés (CCM, 2014).

de múltiples ejes y de curvas y líneas con puntos, así como el remplazo de la textura en los gráficos por los colores (pChart, 2014).

Por tanto, en consideración a lo planteado en la tabla 2 se determinó seleccionar como tecnología para visualizar gráficos a HighCharts por ser la que mayor cantidad de requisitos reúne en la tabla comparativa realizada y por ende la que más se ajusta a las necesidades de la investigación.

Tabla 2: Comparación de tecnologías de visualización de información.

Características	HighCharts	Google Charts	pChart
Licencia	Gratis para uso no comercial	Google	GPL
Libre	Si	Si	Si
Gráfico de barras apiladas	Si	-	Si
Gráfico barras superpuestas	Si	-	Si
Gráfico Barras agrupadas	Si	-	Si
Gráfico de líneas	Si	Si	Si
Gráfico de curvas	Si	-	Si
Gráfico de pastel	Si	-	Si
Gráfico de puntos	Si	Si	Si
Gráfico de burbuja	Si	-	No
Gráficos mixtos	Si	-	Indirecto
Múltiples ejes	Si	-	No
Inclusión de texturas	Si	Si	No
Inclusión de etiquetas en los puntos	Si	Si	Si

Marco de Trabajo

Marco de Trabajo o framework en el desarrollo de *software*, es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de *software* concretos, en base a la cual otro proyecto de *software* puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, librerías y un lenguaje interpretado entre otros programas para ayudar a desarrollar y unir los diferentes componentes de un proyecto (GNU, 2015).

Symfony 2.7

Symfony según (Rosas, 2007) es un proyecto PHP de *software* libre que permite crear aplicaciones y sitios Web rápidos y seguros de forma profesional, es un framework PHP de tipo *full-stack* construido con varios

componentes independientes. Está diseñado para optimizar el desarrollo de las aplicaciones Web basado en el patrón Modelo Vista Controlador.

Entre sus características generales se contemplan:

1. Su código, y el de todos los componentes y librerías que incluye, se publican bajo la licencia MIT¹⁰ de *software* libre.
2. Fácil de instalar y configurar: ha sido probado con éxito en plataformas Windows y derivadas de Unix.
3. Independiente del manejador de base de datos: utiliza Propel, una capa de abstracción que le permite interactuar con varias bases de datos.
4. Simple de usar y al mismo tiempo lo suficientemente flexible para adaptarse a escenarios complejos.
5. Cumple con la mayoría de las mejores prácticas en diseño Web y patrones de diseño.
6. Utilizable en entornos empresariales: puede adaptarse a políticas y arquitecturas ya existentes en tecnologías de información y es lo suficientemente estable para proyectos de largo plazo.
7. Código legible: con comentarios en phpDocumentor para su fácil mantenimiento.
8. Fácil de extender: permitiendo la integración con otras librerías.
9. Incorpora herramientas que facilitan la prueba y depuración de aplicaciones como: unidades de generación de código, pruebas del funcionamiento del *framework*, panel de depuración, interfaz por línea de comandos y configuración en tiempo real.
10. Manejo de memoria caché: lo cual reduce el uso del ancho de banda y la carga en servidor.
11. Mecanismos de autenticación y credenciales: que facilitan la creación de secciones restringidas y la gestión de seguridad de usuarios.
12. URLs inteligentes: que permiten que las direcciones de las páginas Web sean parte de la interfaz y resulten amigables a los motores de búsqueda.
13. La documentación del proyecto también es libre e incluye varios libros y decenas de tutoriales específicos.
14. Las aplicaciones Symfony solamente necesitan permiso de escritura en dos directorios internos de la propia aplicación. Además, Symfony incluye varias herramientas gráficas y de consola para depurar fácilmente los errores que se produzcan en las aplicaciones.

¹⁰ Massachusetts Institute of Technology, por sus siglas en inglés (Opensource, 2015).

Luego del análisis realizado, teniendo en cuenta las características y beneficios expuestos se decide seleccionar Symfony, ya que además de ser sencillo de usar y lo suficientemente flexible como para adaptarse a los casos más complejos; puede integrarse al buscador cubano Orión que contiene una interfaz Web desarrollada con este popular *framework*.

Entorno Integrado de Desarrollo

Netbeans 8.0

NetBeans es un IDE¹¹ gratuito y de código abierto el cual permite al programador un rápido y fácil desarrollo, apto tanto para aplicaciones de escritorio, móviles y aplicaciones Web. Es multiplataforma, encontrándose disponible para Windows, Mac, Linux y Solaris. Posee una amplia plataforma de aplicaciones la cual posibilita desarrollar rápidamente utilizando la plataforma Java, así como Ajax, Groovy, Grails, C /C+ +, así como aplicaciones HTML5 con HTML, JavaScript y CSS. Ofrece también un gran conjunto de herramientas para PHP (Netbeans, 2015).

El proyecto de NetBeans está apoyado por una amplia comunidad de desarrolladores y ofrece una amplia documentación y recursos de capacitación, así como una gran cantidad de *plugins* de terceros.

Es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Cuenta con el autocompletado y el resaltado de código. Cuenta con un potente depurador integrado y además viene con soporte para Symfony 2 un gran *framework* escritorio en php.

Se decide utilizar a Netbeans 8.0 como entorno de desarrollo integrado por ser una herramienta que le brinda facilidades al desarrollador a la hora de programar como es el depurador, además de ser potente y de código abierto. También se tuvo en cuenta su integración con el *framework* seleccionado, convirtiéndola en la selección más indicada.

Lenguajes de desarrollo

Según la definición teórica de la RAE, se entiende como lenguaje a un sistema de comunicación definido por una determinada estructura y contenido; se entiende además en la programación como el vocabulario propio de la informática, como el procedimiento de escritura del código fuente de un *software*. Con estas

¹¹ Integrated Development Environment por sus siglas en inglés.

nociones en claro, se puede afirmar que un lenguaje de programación es aquella estructura que, con cierta base sintáctica y semántica, imparte distintas instrucciones a un programa de computadora. La selección de Symfony como *framework* de desarrollo deriva que sean empleados los lenguajes: PHP, HTML, CSS y JavaScript.

Php 5.5.9

PHP es el acrónimo de *Hipertext Preprocesor*. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, de código abierto muy popular especialmente adecuado para el desarrollo Web y que puede ser incrustado en el código HTML (Alvarez, 2001). Sus inicios se dieron como una modificación a Perl escrita por Rasmus Lerdorf a finales de 1994 con la intención de mantener un control sobre quien visitaba su curriculum en su Web (Henst, 2001).

Una de sus características más potentes es su soporte para gran cantidad de bases de datos. Entre las que se pueden mencionar InterBase, mSQL, MySQL, Oracle, Informix y PosgreSQL. PHP también ofrece la integración con las varias bibliotecas externas, que permiten que el desarrollador haga casi cualquier cosa desde generar documentos en formato .pdf hasta analizar código XML. Dentro de sus principales funcionalidades se pueden listar: el tratamiento de imágenes, la gestión de archivos, la gestión de bases de datos, funciones de correo electrónico, la gestión de archivos PDF y Flash (Alvarez, 2001) y (Pérez, 2012).

Html 5

HTML acrónimo en inglés de *Hypertext Markup Language* es el lenguaje predominante para la construcción de páginas Web. Es usado para describir la estructura y el contenido que contendrá la página Web. El HTML es un lenguaje de marcas que basa su sintaxis en un elemento de base al que se llama etiqueta, a través de estas se van describiendo los elementos del documento, como enlaces, párrafos e imágenes (Villena, 2009).

Su diseño incluye nuevas características, clasificadas en varios grupos según su función se pueden mencionar:

Sin conexión y almacenamiento: Permite a las páginas Web almacenar datos localmente en el lado del cliente y operar sin conexión de manera más eficiente.

Multimedia: Otorga un excelente soporte para utilizar contenido multimedia como lo son audio y video nativamente.

Gráficos y efectos 2D/3D: Proporciona una amplia gama de nuevas características que se ocupan de los gráficos en la Web como lo son canvas 2D, WebGL y SVG.

Rendimiento e Integración: Proporciona una mayor optimización de la velocidad y un mejor uso del *hardware*.

Acceso al dispositivo: Proporciona APIs para el uso de varios componentes internos de entrada y salida de nuestro dispositivo.

CSS3: Ofrece una nueva gran variedad de opciones para hacer diseños más sofisticados.

“CSS u hojas de estilo en cascada (en inglés *Cascading Style Sheets*) es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML. El W3C (*World Wide Web Consortium*) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores” (Ojeda, 2012).

CSS es un lenguaje para definir el estilo o la apariencia de las páginas Web, escritas con HTML o de los documentos XML. CSS se creó para separar el contenido de la forma, a la vez que permite a los diseñadores mantener un control mucho más preciso sobre la apariencia de las páginas (Alvarez, 2008).

JavaScript 1.8.5

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas Web dinámicas. Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios (Mozilla Developer Network, 2015).

Es un lenguaje orientado a eventos, se pueden desarrollar scripts que ejecuten acciones en respuesta a estos eventos. Es similar a Java, aunque no es orientado a objetos y no dispone de herencia, sigue el paradigma de programación basada en prototipos, ya que las nuevas clases se generan clonando las clases base y extendiendo su funcionalidad. El código JavaScript puede ser integrado dentro de las páginas Web.

Para evitar incompatibilidades el W3C diseñó un estándar denominado *Document Object Model* (DOM) o Modelo de Objetos de Documento que es una forma de representar documentos estructurados tales como una página Web HTML o un documento XML que es independiente de cualquier lenguaje. Es soportado por los desarrolladores de los diferentes navegadores Web mediante la implementación del estándar ECMAScript.

Servidor Web

Nginx es un servidor Web/proxy inverso ligero de alto rendimiento (originalmente escrito por Igor Sysoev) y un proxy para protocolos de correo electrónico (IMAP/POP3). Es *software* libre y de código abierto, licenciado bajo la Licencia BSD simplificada. Es multiplataforma, por lo que corre en sistemas tipo Unix (GNU/Linux, BSD, Solaris, Mac OS X, etc.) y Windows (Corona., 2015).

De acuerdo con el estudio de Netcraft, Netcraft's Jul 2014 Web Server Survey, nginx es el segundo servidor Web más usado en dominios activos (14,35%) superando a *Microsoft Information Server*. Además, pasó la marca de ser usado en más de 100 millones de sitios.

Es recomendado principalmente para sitios de Internet que necesiten velocidad, eficiencia mostrando contenido estático, tiene poco uso de memoria, y se recomienda para sitios Web funcionando en un servidor dedicado (VPS).

Apache: En sus investigaciones (Kabir, 2003) plantea que el servidor Apache es un servidor Web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), *Microsoft Windows*, *Macintosh* y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual.

Apache es el servidor Web por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. La licencia Apache es una descendiente de la licencia BSD, no es GPL. Esta licencia te permite hacer lo que quieras con el código fuente siempre y cuando se les reconozcas su trabajo (Apache, 2015).

A continuación, se mostrarán algunas de sus características:

1. Es multiplataforma por lo que puede ser utilizado en una multitud de sistemas operativos, lo que lo hace prácticamente universal.

2. Su estructura le permite trabajar con varios lenguajes de script tales como Perl, PHP y otros además funciona con Java y páginas .jsp, ofreciendo todo el soporte que se necesita para tener páginas dinámicas.
3. Permite personalizar las respuestas ante los posibles errores que puedan ocurrir en el servidor.
4. Ofrece una alta configurabilidad en la creación y gestión de *logs*¹².
5. Altamente configurable y de diseño modular. Por lo cual resulta muy sencillo ampliar sus capacidades como servidor Web.

Para la propuesta de solución se selecciona este último servidor por tratarse de uno de los mejores y más popular en la actualidad, lo cual facilita la obtención de documentación especializada, además por su robustez, su alta configuración y por ser de código abierto.

A continuación, se realiza una descripción de los sistemas de almacenamientos que se utilizarán con el objetivo de realizar consultas para conformar el reporte. Se empleará el gestor de base de datos no relacional MongoDB y el servidor de indexación Solr por ser los que utiliza actualmente el motor de búsqueda Orión para almacenar su colección de información.

Jmeter

JMeter es un proyecto de Apache que puede ser utilizado como una herramienta de prueba de carga para analizar y medir el desempeño de una variedad de servicios, con énfasis en aplicaciones Web. 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 (The Apache Software Foundation-JMeter, 2015).

Acunetix

Acunetix comprueba los sistemas en busca de múltiples vulnerabilidades que un atacante podría aprovechar para obtener acceso a los sistemas y datos. Acunetix puede utilizarse para realizar escaneos de vulnerabilidades en aplicaciones Web y para introducir pruebas de acceso frente a los problemas identificados. La herramienta provee sugerencias para mitigar las vulnerabilidades identificadas y puede

¹² Es un registro de actividad del sistema, que generalmente se guarda en un fichero de texto, al que se le van añadiendo líneas a medida que se realizan acciones sobre el propio sistema.

utilizarse para incrementar la seguridad de servidores Web o de las aplicaciones que se analizan (Acunetix, 2015).

Sistema Gestor de Base de Datos

“Un Sistema Gestor de Base de Datos (SGBD, en inglés DBMS: *DataBase Management System*) es un sistema de *software* que permite la definición de bases de datos; así como la elección de las estructuras de datos necesarios para el almacenamiento y búsqueda de los datos, ya sea de forma interactiva o a través de un lenguaje de programación” (Alvarez, 2007).

Se define también como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones:

1. Definición de los datos.
2. Mantenimiento de la integridad de los datos dentro de la base de datos.
3. Control de la seguridad y privacidad de los datos.
4. Manipulación de los datos.

Este tipo de sistema facilita el trabajo de los desarrolladores con el manejo de las bases de datos aumentando el nivel de productividad.

MongoDB

MongoDB es un sistema de base de datos NoSQL orientado a documentos, desarrollado bajo el concepto de código abierto. MongoDB forma parte de la familia de sistemas de base de datos NoSQL. En vez de guardar los datos en tablas como se hace en las bases de datos relacionales, MongoDB guarda estructuras en documentos tipo JSON con esquema dinámico (MongoDB llama ese formato BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

Es una base de datos ágil que permite a los esquemas cambiar rápidamente cuando las aplicaciones evolucionan, creado para brindar escalabilidad, rendimiento y gran disponibilidad, escalando de una implantación de servidor único a grandes arquitecturas complejas de centros multidados (MongoDB).

Estas son sus principales características según (Couto, 2014).

1. **Alta disponibilidad.**

2. **Escalabilidad:** Permite particionar, de manera transparente para el usuario, nuestra base de datos en tantas como *shards*¹³ existan. Esto aumenta el rendimiento del procesado de los datos al ser cada una de ellas más pequeña que la original.
3. **Auto balanceado de carga:** A través de los distintos *shards*, el balanceador decide cuándo migrar los datos y a qué *shard*, para que estén uniformemente distribuidos entre todos los servidores del clúster. Cada *shard* aloja los datos correspondientes a un rango de la clave escogida para particionar nuestra colección.
4. **Replicación nativa:** Sincroniza los datos entre los servidores.
5. **Seguridad:** Permite la autenticación, autorización, etc.
6. **Gestión avanzada de usuarios.**
7. **Conmutación por error automático:** Elección automática de un nuevo servidor primario cuando el original se ha caído.
8. Es capaz de actualizarse sin dejar de dar servicio.
9. No tiene los cuellos de botella que se producen en las bases de datos relacionales (RDBMS) al procesar grandes cantidades de información.
10. Utiliza objetos JSON para guardar y transmitir la información.
11. Puede almacenar y ejecutar funciones JavaScript en el servidor.
12. **Alertas configurables:** Permiten descubrir problemas antes de que afecten el funcionamiento de nuestra aplicación.
13. **Backup:** Guarda múltiples copias de cada *backup* en distintos *data centers* (distribuidos geográficamente y tolerantes al fallo).
14. **Automatización de tareas:** Permite configurar rápida y fácilmente servidores MongoDB aislados (*sharded o clusters*)¹⁴.

Servidor de indexación

Solr es un servidor de índice empresarial, que actúa como una base de datos no SQL. Una plataforma de búsqueda de código que funciona como un "servidor de búsquedas". Sus principales características incluyen

¹³ Referente a (*Sharding*), es una técnica de escalado que consiste en particionar los datos de una base de datos horizontalmente agrupándolos de algún modo que tenga sentido y que permita un direccionamiento más rápido (Pérez, 2009).

¹⁴ Conjunto o conglomerado de computadoras unidos entre sí normalmente por una red de alta velocidad y que se comportan como si fuesen una única computadora

búsquedas de texto completo, resaltado de resultados, clustering dinámico, y manejo de documentos ricos (como Word y PDF). Solr es escalable, permitiendo realizar búsquedas distribuidas y replicación de índices.

Según la página principal de Solr, las características más destacables de este servidor de búsquedas son:

1. Capacidades avanzadas de búsqueda de texto completo ("*Full-Text Search*").
2. Optimizado para un tráfico Web elevado.
3. Interfaces abiertas basadas en estándares – XML, JSON y HTTP.
4. Completas interfaces Web de administración.
5. Estadísticas del servidor expuestas mediante JMX para su monitorización.
6. Escalable linealmente, replicación automática del índice, recuperación automática.
7. Indexación cuasi-inmediata.
8. Flexible y adaptable a través de una configuración en formato XML.
9. Arquitectura extensible mediante *plugins*.

Solr está escrito en Java y se ejecuta como un servidor de búsqueda de texto completo independiente dentro de un contenedor de *servlets* como Tomcat. Solr Lucene utiliza la biblioteca Java de búsqueda en su base para la indexación de texto completo y de búsqueda, y tiene como REST HTTP / XML y JSON APIs que hacen que sea fácil de utilizar desde prácticamente cualquier lenguaje de programación. Potente configuración externa de Solr permite que sea adaptado a casi cualquier tipo de aplicación Java sin codificación, simplemente hay que utilizarlo con peticiones GET para realizar las búsquedas en el índice, y POST para agregar documentos (Apache Solr, 2015).

Metodología de desarrollo

Una metodología de desarrollo en ingeniería de *software* es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo. Es un proceso para la producción organizada del *software*, empleando para ello una colección de técnicas predefinidas y convencionales en las notaciones. Una metodología se presenta normalmente como una serie de pasos, con técnicas y notaciones asociadas a

cada paso. Los pasos de la producción del *software* se organizan normalmente en un ciclo de vida consistente en varias fases de desarrollo (Pressman, 2005).

A grandes rasgos, si se toma como criterio las notaciones utilizadas para especificar artefactos producidos en actividades de análisis y diseño, se pueden clasificar las metodologías en dos grupos, las robustas y las ágiles.

Luego del estudio de los dos grupos de metodologías para el desarrollo de *software* y teniendo en cuenta las características de cada una de ellas, en aras de optimizar tiempo y evitar las demoras innecesarias por excesiva documentación, teniendo en cuenta además la particularidad de que la propuesta de solución será desarrollada por una única persona, se decide utilizar una metodología ágil.

En consecuencia, de lo antes planteado y tomando en cuenta que la propuesta de solución deberá ser integrada a un proyecto real existente el cual reside bajo la tutela y estrategia de trabajo del Centro de Ideoinformática, CIDI a partir de ahora, se decidió elaborar la presente investigación bajo las pautas establecidas por la metodología Proceso Unificado Ágil o AUP por sus siglas, en una adaptación hecha por la UCI. Dicha metodología es la adoptada por el centro y por ende la más acorde a las necesidades del proyecto. A continuación, se describe brevemente las características propias de AUP.

Proceso Unificado Ágil (AUP)

El Proceso Unificado Ágil de Scott Ambler o (AUP) es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de *software* de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP. AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas (*test driven development* - TDD), Modelado Ágil, Gestión de Cambios Ágil, y Refactorización de Base de Datos para mejorar la productividad (Flores, 2012).

AUP define cuatro fases:

Inicio: El objetivo de esta fase es obtener una comprensión común cliente-equipo de desarrollo del alcance del nuevo sistema y definir una o varias arquitecturas candidatas para el mismo. **Elaboración:** El objetivo es que el equipo de desarrollo profundice en la comprensión de los requisitos del sistema y en validar la arquitectura.

Construcción: Durante la fase de construcción el sistema es desarrollado y probado al completo en el ambiente de desarrollo.

Transición: El sistema se lleva a los entornos de preproducción donde se somete a pruebas de validación y aceptación y finalmente se despliega en los sistemas de producción.

La estrategia de trabajo definida por AUP se rige por cinco principios fundamentales:

1. **Simplicidad:** Todo se describe concisamente utilizando poca documentación
2. **Agilidad:** El ajuste a los valores y principios de *La Alianza Ágil* ¹⁵.
3. **Centrarse en actividades de alto valor:** La atención se centra en las actividades que en realidad lo requieren, no en todo el proyecto.
4. **Herramienta de la independencia:** Las herramientas a utilizar las define el equipo de desarrollo (Se sugiere utilizar las herramientas más adecuadas, herramientas simples y herramientas de código abierto.)
5. **Adaptar el producto para satisfacer necesidades:** La metodología AUP es un producto de fácil uso utilizando cualquier herramienta. No es necesario comprar una herramienta especial, o tomar un curso, para adaptar esta metodología.

Al no existir una metodología de *software* universal, ya que toda metodología debe ser adaptada a las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigiéndose así que el proceso sea configurable. Se decide utilizar una variación de la metodología AUP, de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la UCI. La metodología que se seleccionó para que guiara el desarrollo del módulo Generador de Reportes Webmétricos fue AUP-UCI por estar diseñada para grupos de trabajo pequeños, fomentar el trabajo en equipo integrando al cliente y centrarse en la naturaleza colaborativa de desarrollo de *software*. Además, se tuvo en cuenta que la metodología seleccionada es la que se usa en el proyecto Motor de Búsqueda Cubano (Orión) al cual se le implementará el módulo que constituye la propuesta de solución.

¹⁵ Grupo de pensadores independientes de desarrollo de *software* que coincidieron en la esencia del *Manifiesto para el Desarrollo de Software Ágil*.

Lenguajes de modelado

BPMN

Business Process Modeling Notation (BPMN) es una notación gráfica que describe la lógica de los pasos de un proceso de Negocio. Esta notación ha sido especialmente diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes de las diferentes actividades. BPMN constituye un estándar internacional de modelado de procesos, el cual es independiente de cualquier metodología de modelado de procesos. Esta notación crea un puente estandarizado para disminuir la brecha entre los procesos de negocio y la implementación de estos, permitiendo modelar los procesos de una manera unificada y estandarizada logrando así un entendimiento a todas las personas de una organización. BPMN proporciona un lenguaje común para que las partes involucradas puedan comunicar los procesos de forma clara, completa y eficiente, de esta forma BPMN define su propia notación y su semántica de un Diagrama de Procesos de Negocios. A su vez está planeada para dar soporte únicamente a aquellos procesos que sean aplicables a procesos de negocios, lo cual significa que cualquier otro tipo de modelado, con fines distintos a los del negocio, no estará en el ámbito de BPMN (*Object Management Group, 2012*).

El Lenguaje Unificado de Modelado (UML)

Es el lenguaje de modelado de sistemas de *software* más conocido y utilizado en la actualidad; está respaldado por el Grupo de Gestión de Objeto (OMG, siglas del inglés *Object Management Group*). Se usa para especificar, visualizar, construir y documentar artefactos de un sistema de *software*. Captura decisiones y conocimiento sobre los sistemas que se deben construir. Se usa para entender, diseñar, hojear, configurar, mantener, y controlar la información sobre tales sistemas (Larman, 2003). Está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. El lenguaje de modelado pretende unificar la experiencia pasada sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar. UML incluye conceptos semánticos, notación, y principios generales. Tiene partes estáticas, dinámicas, de entorno y organizativas. Está pensado para ser utilizado en herramientas interactivas de modelado visual que tengan generadores de código, así como generadores de informes. La especificación de UML no define un proceso estándar, pero está pensado para ser útil en un proceso de desarrollo iterativo. Pretende dar apoyo a la mayoría de los procesos de desarrollo orientados a objetos (Cornejo, 2008).

Para la elaboración de los modelos necesarios para la propuesta de solución se selecciona como lenguaje de modelado el UML, ya que es capaz de definir, detallar, documentar y construir un sistema de *software*. Es un lenguaje fácil de aprender, adaptable a cambios y ofrece gran variedad de diagramas para visualizar el *software* desde varias vías, facilitando así la relación entre clientes y desarrolladores.

Herramientas CASE

Las herramientas CASE (**C**omputer **A**ided **S**oftware **E**ngineering, Ingeniería de *Software* Asistida por Ordenador) hacen referencia a las aplicaciones con un conjunto de herramientas y métodos para incrementar la productividad del desarrollo *software* y reducir los costos de tiempo y dinero, obteniendo un *software* de alta calidad, sin defectos y mantenible. Estas herramientas facilitan la ejecución de tareas como el proceso de diseño del proyecto, cálculo de costos, implementación de parte del código, compilación automática, documentación o detección de errores en todos los estados del ciclo de vida de desarrollo de *software* (Ambyssoft, 2014).

El empleo de herramientas CASE permite integrar durante todo el proceso de ciclo de vida del proyecto aspectos como:

1. Análisis de datos y procesos integrados mediante un repositorio.
2. Generación de interfaces entre el análisis y el diseño.
3. Generación del código a partir del diseño.
4. Control de mantenimiento.

EasyCASE

Es una herramienta que automatiza las fases de análisis y diseño del desarrollo de un proyecto, eliminando algunas de las tareas más repetitivas y mecánicas. Puede usarse para formar estructuras de análisis, diseño de estructuras y modelar información y datos.

Su amplia gama de funcionalidades permite generar esquemas de base de datos e ingeniería reversa, la capturar los detalles de diseño de un sistema y comunicar las ideas gráficamente, para que sean fáciles de ver y entender. Su diseño legítimo y modelado de datos, procesos y eventos, permite crear y mantener diagramas de flujo de datos, diagramas de entidad-relación y mapas de estructura. Posee herramientas de

corrección avanzadas que permiten revisiones generales. Permite re-usar diagramas o partes de diagramas para economizar el diseño de un proyecto. Soporta una gama amplia de metodologías estructuradas, permitiendo escoger los métodos más apropiados para realizar las tareas. Determina los tipos de esquemas según la metodología del proyecto seleccionada y notifica de errores a medida que el modelo vaya construyéndose. Posee desde el editor de diagramas flexible y un diccionario de los datos, así como una extensa cantidad de reportes y análisis. Es una herramienta multi-usuario lo cual permite compartir datos y trabajar en un proyecto con otros departamentos. El equipo completo puede acceder a proyectos localizados en el servidor de la red concurrentemente. Para asegurar la seguridad de los datos, existe el diagrama y diccionario de los datos que bloquean por niveles al registro, al archivo y al proyecto, y niveles de control de acceso (Pressman, 2005).

Visual Paradigm 8.0

Visual Paradigm para UML 8.0 es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de *software*: análisis y diseño orientados a objetos, construcción y despliegue. Es multiplataforma, utiliza UML como lenguaje de modelado y cuenta con una versión libre para la comunidad (*Community Edition*), ayudando de una manera rápida a la construcción de aplicaciones de mayor calidad y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Esta herramienta también proporciona una mejor interfaz gráfica de usuario y una mayor base de datos de esquema de apoyo, permite generar la documentación del sistema en los formatos PDF, HTML y el formato de documentos de *Microsoft Word* y permite importar proyectos de otras herramientas de modelado como *Rational Rose*, *Erwin* y *Microsoft Visio*. Soporta la revisión ortográfica, brindando sugerencias para los idiomas: inglés, español, francés, alemán y portugués (PARADIGM V, 2013).

Se selecciona como herramienta de modelado el *Visual Paradigm* (versión 8.0) proporcionando calidad y rapidez a la hora de implementar y desarrollar una aplicación ya que contiene todos los tipos de diagramas de clases y cuenta además con abundante documentación. Soporta el ciclo de vida completo del desarrollo del *software* ya sea análisis y diseño, construcción, pruebas y despliegue. Siendo así la opción que más se ajusta a las necesidades de la propuesta de solución.

1.5 Conclusiones del capítulo

Los conceptos asociados al objeto de estudio sirvieron de base y guía en el desarrollo de la investigación, adquiriendo y fomentando conocimientos necesarios, que contribuyeron en gran medida al logro de la propuesta de solución. El estudio realizado a los sistemas homólogos permitió identificar que ninguno de los sistemas le daba solución al problema, por lo que quedó demostrado la necesidad de implementar un módulo capaz de generar reportes webmétricos específicos y que a su vez dé cumplimiento a las necesidades actuales del motor de búsqueda Orión. Además, este estudio aportó funcionalidades que pueden ser útiles para la solución del problema planteado. La selección de los indicadores webmétricos definió los datos que se mostrarán en el reporte que generara el módulo, mientras que el estudio de las herramientas, tecnologías y metodologías permitió definir la base tecnológica para el desarrollo de la propuesta de solución.

Capítulo 2: Diseño del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión

2.1 Introducción

El presente capítulo abordará el diseño del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión. En el capítulo se realiza una descripción de las características del módulo, se identifican las clases del dominio y la relación que existe entre ellas es mostrada a través del diagrama de clases del modelo del dominio. También se especifican los requerimientos funcionales y no funcionales, se realiza las historias de usuario y las descripciones textuales de estas. Además, se podrán observar el diagrama de clases, el modelo de datos y el modelo de despliegue del módulo Generador de Reportes Webmétricos para Orión, así como la descripción de los patrones de diseño empleados.

2.2 Descripción de la propuesta de solución

Dadas las necesidades planteadas en la situación problemática del presente trabajo de diploma la solución propuesta constituye un módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión. Este módulo debe permitir a los *webmasters* generar reportes personalizables mediante una configuración previa, elaborados con datos estadísticos específicos sobre estado general de sus sitios con el objetivo de contribuir a la toma de decisiones de los *webmasters* y por ende a mejorar el posicionamiento Web. La arquitectura de información del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión debe ser amigable y fácil de entender para usuarios que tengan conocimientos informáticos básicos.

2.3 Modelado del dominio

Para una mejor comprensión del entorno donde se evidencia el problema en cuestión, en el presente trabajo de diploma se determinó realizar el modelo del dominio. Este permite la representación de las clases conceptuales del mundo real, no de componentes *software*. No se trata de un conjunto de diagramas que describen clases *software*, u objetos *software* con responsabilidades (Larman, 2004). El objetivo del modelado del dominio es comprender y describir las clases más importantes dentro del contexto. (Ver figura 1).

Dentro de este modelo se aprecia un *webmaster* como actor principal del proceso. Este es el encargado de seleccionar entre las distintas formas en que se pueden visualizar los reportes que el mismo puede generar.

Dichos reportes permiten mostrar al *webmaster* las estadísticas que pertenecen al sitio Web donde es administrador.

Descripción de clases del modelo del dominio

Webmaster: Persona responsable del mantenimiento o programación de un sitio Web.

Sitio Web: Un conjunto de páginas Web ordenadas jerárquicamente bajo una misma dirección Web (URL).

Reporte: Se les denomina a los datos estadísticos que se encuentran agrupados de una manera estructurada y resumida.

Estadísticas: Información relacionada con la visibilidad, la calidad, y el tamaño de un sitio Web.

Diagrama de clases del modelo del dominio

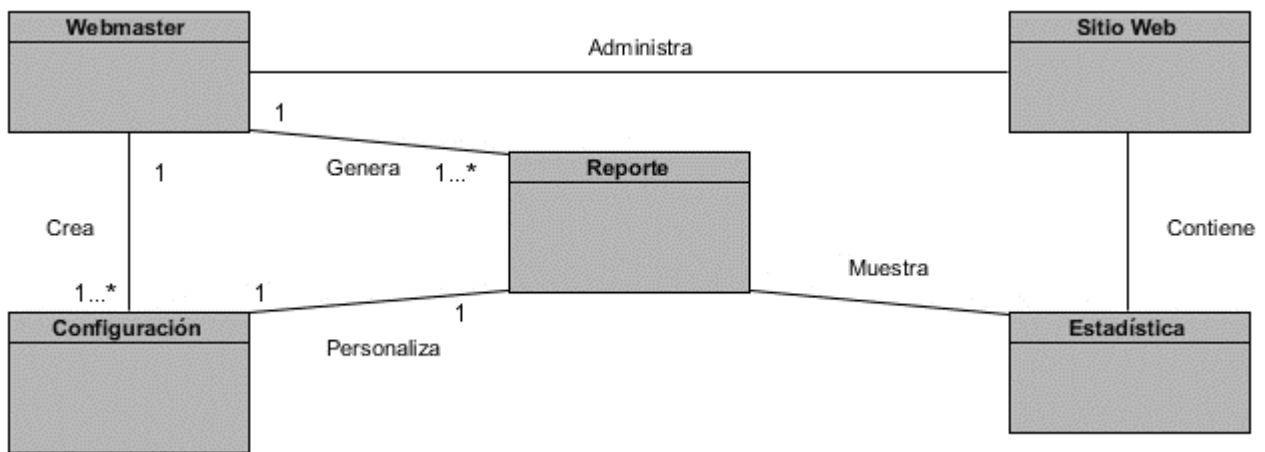


Figura 1: Modelo del dominio

2.4 Especificación de requisitos de *software*

Según (Pressman, 2005) un requerimiento de *software* es una condición o capacidad que debe ser conformada por el sistema. Según la IEEE también puede definirse como una condición o necesidad que debe exhibir o poseer un sistema para satisfacer un contrato, estándar, especificación u otra documentación formalmente establecida.

Según lo planteado por (Hadad y autores, 2008) la tarea de asignar prioridades requiere de la participación de clientes y usuarios con cierto nivel de decisión y puede realizarse de diversas maneras,

tales como reuniones, cuestionarios y otras. Se debe determinar la importancia relativa que tiene un requisito para los clientes y usuarios, y organizar aquellos requisitos que deben implementarse inicialmente frente a aquellos que pueden postergarse. Al asignar prioridades, se deben tener en cuenta la dependencia entre requisitos, la multiplicidad de intereses de los clientes y usuarios, las limitaciones de recursos, las necesidades del negocio, las imposiciones del mercado y los costos de implementación, entre otros factores.

En el caso de la presente investigación la prioridad quedó ajustada en mutuo acuerdo con el cliente durante reuniones efectuadas durante la etapa de identificación de requisitos, determinando con prioridad **alta** a aquellos requisitos sin los cuales el negocio no puede funcionar, **media** a aquellos que facilitan u enriquece el valor funcional del producto y **baja** al resto de los requisitos que no resultan imprescindibles para el correcto funcionamiento del sistema.

A continuación, se presentan los requerimientos que debe cumplir el módulo que se propone. Estos han sido clasificados en funcionales y no funcionales.

Requisitos Funcionales (RF) y Requisitos no Funcionales (RnF)

Tabla 3: Requisitos Funcionales y no Funcionales

Código	Descripción (Requisitos Funcionales)	Prioridad
RF1	Autenticar usuarios.	Alta
RF2	Registrar usuarios.	Alta
RF3	Activar la cuenta de los usuarios mediante correo electrónico.	Media
RF4	Insertar usuario.	Alta
RF5	Mostrar usuario	Alta
RF6	Editar usuario	Alta
RF7	Eliminar usuario	Alta
RF8	Modificar contraseña.	Baja
RF9	Recuperar contraseña mediante correo electrónico.	Baja
RF10	Mostrar los sitios disponibles para realizarles reportes.	Alta
RF11	Seleccionar sitio a visualizar.	Media
RF12	Mostrar estadísticas del sitio.	Media
RF13	Mostrar la cantidad enlaces salientes.	Media
RF14	Listar todos los enlaces salientes.	Media
RF15	Mostrar enlaces entrantes.	Media
RF16	Listar todos los enlaces entrantes.	Media
RF17	Mostrar los tipos MIME existentes en el sitio.	Media
RF18	Graficar estadísticas MIME del sitio.	Baja

RF19	Mostrar el total de páginas indexadas del sitio.	Media
RF20	Mostrar las palabras claves del sitio.	Media
RF21	Insertar el sitio al que se desea realizar el reporte.	Baja
RF22	Generar Reporte	Alta
RF23	Exportar el reporte en formato pdf.	Media
RF24	Personalizar el reporte.	Baja
RF25	Guardar el reporte en Base de Datos.	Alta
RF26	Visualizar los reportes almacenados en Base de datos.	Alta
RF27	Eliminar los reportes guardados en Base de Datos.	Alta
RF28	Enviar el reporte por correo electrónico.	Baja
RnF (Requisitos no Funcionales) para el módulo		
Apariencia de Interfaz o Externa		
RnF-1	La Interfaz de Usuario (IU) debe poseer un diseño adaptable que le permita ser visualizada en diferentes dispositivos móviles.	
RnF-2	La interfaz debe tener un diseño uniforme para todos los usuarios.	
Usabilidad		
RnF-3	Facilidad de uso por parte de los usuarios: el sistema debe tener una interfaz amigable y cómoda para los usuarios que posean conocimientos básicos en el tratamiento con las computadoras y de un ambiente Web.	
Rendimiento		
RnF-4	El sistema debe responder en un máximo de 2 segundos las solicitudes de un usuarios.	
RnF-5	El sistema debe atender las solicitudes de un máximo de 100 usuarios concurrentes en el módulo.	
Seguridad		
RnF-6	La sesión de un usuario debe expirar después de estar inactivo por más de 15 minutos.	
RnF-7	El <i>software</i> , bibliotecas o componentes empleados deben estar regidos en la medida de lo posible por licencias y patentes de <i>software</i> libre.	
Hardware		
RnF-8	Se recomienda que el sistema tenga un mínimo de 4 GB de memoria RAM.	
RnF-9	El sistema necesita un Procesador: Core2Duo/DualCore/ u otro superior.	

Historias de usuario (HU)

Las HU sirven para registrar los requerimientos de los clientes según el negocio y son utilizadas para poder realizar la estimación de cada una de las iteraciones durante la fase de planificación. Las HU son escritas por el equipo de trabajo en conjunto con los clientes en base a lo que se estima que es necesario para el sistema. Están escritas en un formato de oraciones en la terminología del cliente, sin necesidad de sintaxis técnicas. También son utilizadas para poder crear las pruebas de aceptación. Las HU solo proveen suficiente detalle para poder realizar la estimación de cuánto tardará en ser implementada dicha funcionalidad. Una gran diferencia entre las HU y los documentos tradicionales es que se centran en lo que el cliente necesita (Balarezo, 2013).

En las HU se considera:

La prioridad en el negocio:

- **Alta:** Cuando son consideradas por los clientes esenciales para el funcionamiento del negocio.
- **Media:** Cuando el cliente cree que son necesarias, pero estas no intervienen en gran medida en el desarrollo del negocio.
- **Baja:** Cuando constituyen procesos que se deben tener en cuenta, pero su ausencia no perjudica el flujo principal del negocio.

El riesgo en desarrollo:

- **Alto:** Cuando en la implementación de las HU pueden surgir errores que lleven a la inoperatividad del código.
- **Medio:** Cuando en la implementación de las HU pueden existir errores que retrasen la entrega del producto.
- **Bajo:** Cuando pueden aparecer errores que serán tratados con relativa facilidad sin que traigan perjuicios para el desarrollo del proyecto.

Un Punto de Estimación equivale a una semana de programación, una semana de programación corresponde a 40 horas en desarrollo, 8 horas durante 5 días de la semana del calendario normal.

A continuación, la HU del sistema perteneciente al RF-1. Insertar el sitio al que se desea realizar el reporte.

Tabla 4: Historia de Usuario # 1

Historia de usuario	
Número: HU_1	Nombre Historia de Usuario: Guardar Reporte
Modificación de Historia de Usuario Número: 0	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en el Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Programador responsable: Duanny Sánchez Chinae	
Descripción: Luego de seleccionar un host y realizarle un reporte se puede proceder a guardar este en base de datos haciendo uso del botón “Reporte” ubicado en la esquina superior derecha.	
Observaciones:	

Prototipo de Interfaz:

Tabla 5: Historia de Usuario # 2

Historia de usuario	
Número: HU_2	Nombre Historia de Usuario: Autenticar usuario
Modificación de Historia de Usuario Número: 0	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en el Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Programador responsable: Duanny Sánchez Chinaea	
Descripción: Un usuario se loguea en el sistema llenando los campos solicitados.	
Observaciones: El usuario debe estar previamente registrado y su cuenta debe permanecer activa.	
Prototipo de Interfaz:	

2.5 Análisis y Diseño

Descripción de los estilos arquitectónicos y los patrones de diseño

La utilización de patrones arquitectónicos y de diseño, contribuye a mejorar la calidad del sistema. Su éxito está dado por la forma en que estos alcanzan de manera satisfactoria los objetivos de la ingeniería de *software*. Los patrones arquitectónicos y de diseño soportan el desarrollo, mantenimiento y evolución de sistemas complejos y de gran escala (Cardeso 2004).

Los autores (Reynoso y Kicillof, 2004) en su libro “*Estilos y Patrones en la Estrategia de Arquitectura de Microsoft.*” plantean un que un estilo arquitectónico es la organización fundamental de un sistema encarnada en sus componentes, las relaciones de los componentes con cada uno de los otros y con el entorno, y los principios que orientan su diseño y evolución.

Estilos Arquitectónicos.

Para el desarrollo del módulo Generador de Reportes Webmétricos se empleará el patrón arquitectónico Modelo-Vista-Controlador (MVC). Aunque según su creador Fabien Potencier “Symfony no es un *framework* MVC”, este basa su funcionamiento interno en esta arquitectura, utilizada por la mayoría de *frameworks* Web (Figura 2). La aplicación de dicho marco de trabajo para el desarrollo del módulo propuesto conlleva a la implementación de este patrón.

La utilización de la arquitectura MVC en el desarrollo de la solución propuesta estableció una división lógica en tres partes principales. El modelo de datos, la presentación o interfaz de usuario (vistas) y la lógica de negocio (controladores).

En el **modelo** se encuentran las clases que permiten la representación de los datos del módulo, gracias a esto se evita mezclar los accesos a bases de datos con el código que implementa la lógica de negocio. Todo lo relativo al acceso, modificación y persistencia de datos se encuentra en el modelo, actuando como capa de abstracción entre el usuario y los datos. Cuando se desea acceder a los datos, no se realiza un acceso de manera directa desde un controlador, sino que se delega la acción al modelo.

El **controlador** es el centro del módulo, en clases como ReporteControler, UsuarioControler y SitioControler es donde se implementa la lógica de negocio. Por una parte, responden a las peticiones que hace el usuario, y por otra parte se encargan de realizar las solicitudes de datos al modelo. Son intermediarios entre el modelo de datos y las vistas para los usuarios.

La **vista** es la capa de presentación de los datos de la propuesta de solución, en la ruta SeoBundle\Resources\views, se encuentran todas las interfaces del módulo, las cuales constituyen abstracciones completas de la lógica de negocio y del acceso a datos, y se limitan a representar de manera gráfica los datos y todos los elementos que el usuario visualiza dentro la aplicación.

Debido a sus características internas el uso del patrón MVC trae consigo ventajas significativas dentro de las cuales se pueden mencionar:

1. La separación entre los componentes permite implementarlos por separado. Sin afectar la integridad del producto final.
2. La conexión entre el Modelo y sus Vistas es dinámica, se produce en tiempo de ejecución, no en tiempo de compilación.
3. Soporta para múltiples vistas, debido a que la Vista se separa del Modelo y no hay ninguna dependencia directa entre ambos.
4. Permite un mayor soporte a los cambios, debido a que los requisitos de interfaz tienden a cambiar más rápidamente que las reglas de negocio. Puesto que el modelo no depende de las vistas, las adiciones de nuevos tipos de vista al sistema generalmente no afectan al modelo. Por tanto, el ámbito del cambio se limita a la vista.

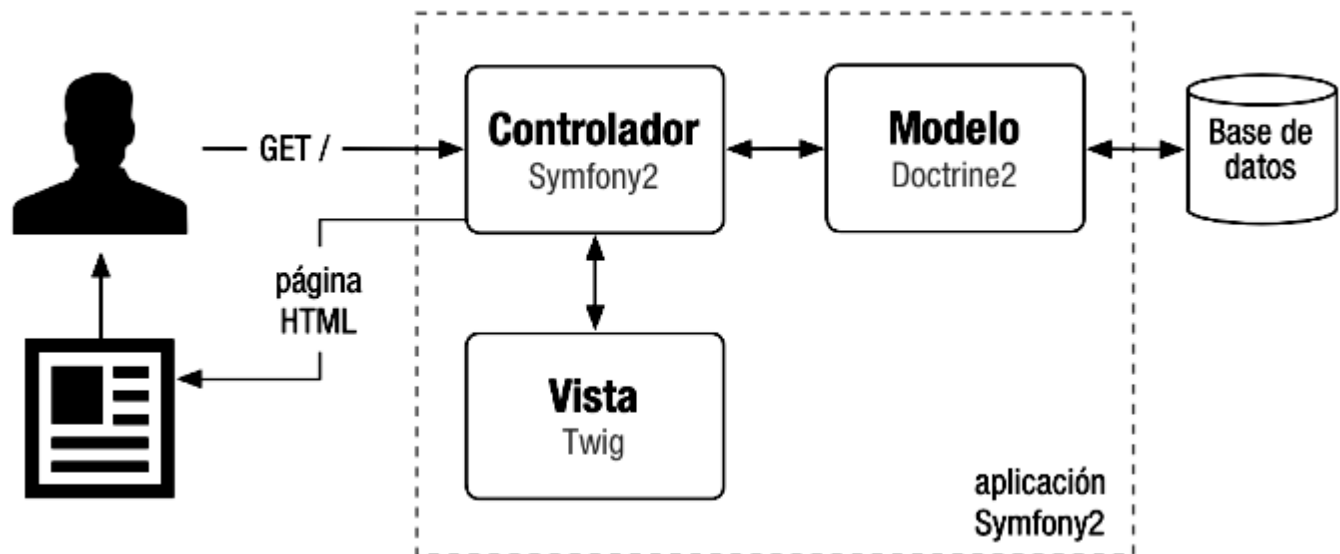


Figura 2: Esquema simplificado de la arquitectura interna de Symfony (Vega, 2016).

Patrones de diseño

Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí, adaptada para resolver un problema de diseño general en un contexto particular. El mismo identifica: clases, instancias, roles, colaboraciones y la distribución de responsabilidades. Estos modelos que se presentan como parejas de problema/solución con un nombre, codifican buenos principios y sugerencias relacionados con la asignación de responsabilidades, basados en la recopilación del conocimiento de los expertos en desarrollo de *software* (Wordpress, 2011).

Para diseñar el módulo se tienen en cuenta los Patrones de *Software* para la Asignación General de Responsabilidad o GRASP (*General Responsibility Assignment Software Patterns*) y Banda de los Cuatro o GOF (*Gang of Four*).

A continuación, se muestra una selección de estos patrones los cuales serán utilizados durante el diseño del módulo:

Experto: este patrón plantea que se debe asignar una responsabilidad al experto en información, en otras palabras, a la clase que cuenta con los datos necesarios para cumplir la responsabilidad. De esta forma, se

conserva el encapsulamiento de la información, puesto que los objetos ejecutan las tareas que le corresponden de acuerdo a la información que poseen, lo que da lugar a sistemas más robustos y fáciles de mantener (Larman, 2004). Dicho patrón se manifiesta en la propuesta de solución en cada una de las clases controladoras, ya que Symfony2 utiliza este patrón con la inclusión de Doctrine para el mapeo de bases de datos. Se utiliza específicamente para crear una capa de abstracción en el modelo, encapsular toda la lógica de los datos y generar las clases entidades con todas las funcionalidades comunes (GET, SET y el constructor de la entidad); las clases de abstracción de datos poseen un grupo de funcionalidades que están relacionadas directamente con la entidad que representan y contienen la información necesaria de la tabla asociada.

Específicamente en la propuesta de solución se observa este patrón en *doctrine*, se encuentra presente en cualquiera de los controladores, como ejemplo en la clase que se realiza la inserción a la base de datos se utiliza la funcionalidad `guardarReporteAction` como se muestra en el siguiente segmento del código fuente.

```
$em = $this->get('doctrine.odm.mongodb.document_manager');  
$reporte = new Reporte();
```

Creador: Su implementación permite identificar quién debe ser el responsable de la creación o instanciación de nuevos objetos o clases. Dicho de otra manera, este patrón plantea que se debe asignar a una clase A la responsabilidad de crear una instancia de una clase B. Como la creación de objetos es una de las actividades más frecuentes en un sistema orientado a objetos, es importante el uso de este para guiar la asignación de responsabilidades relacionadas con la creación de objetos.

Durante el desarrollo de la propuesta de solución se evidencia el uso de este patrón en las clases controladoras cuando heredan funcionalidades comunes pertenecientes a su padre, justo como se muestra en el siguiente segmento de código fuente perteneciente a la clase `ReporteController`.

```
class ReporteController extends Controller  
{  
    public function indexAction() {  
        $dm = $this->get('doctrine.odm.mongodb.document_manager');  
        $reportes = $dm->getRepository('SeoBundle:Reporte')->findAll();  
        return $this->render('SeoBundle:Reporte:index.html.twig', array('reportes' => $reportes));  
    }  
}
```

Alta cohesión: en el diseño orientado a objetos, la cohesión es una medida de la fuerza con la que se relacionan y del grado de focalización de las responsabilidades de un elemento (clase o subsistema). Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas, que colaboran entre sí y con otros objetos para simplificar su trabajo. Una clase con alta cohesión es relativamente fácil de mantener, entender y reutilizar (Larman, 2004). La alta cohesión hace referencia a cuanta responsabilidad tiene una determinada clase controladora, o sea, una clase debe tener responsabilidades moderadas. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo excesivo.

Symfony2 permite la organización del trabajo en cuanto a la estructura del proyecto y la asignación de responsabilidades con alta cohesión. Se puede observar en el sistema, ya que, cada clase controladora maneja solamente las responsabilidades correspondientes a las entidades con las que se relaciona, además para cada vista existe una página controladora encargada de manejar sus solicitudes.

En la propuesta de solución se evidencia el uso de este patrón en los controladores, en la forma que cada controlador atiende una sola entidad, por ejemplo: la clase **ReporteControler** atiende únicamente las responsabilidades referentes a los reportes.

Bajo Acoplamiento: el patrón bajo acoplamiento impulsa la asignación de responsabilidades de manera que su localización no incremente el acoplamiento hasta un nivel que lleve a los resultados negativos que puede producir un acoplamiento alto (Larman, 2004). En otras palabras, que una clase no dependa de muchas otras, lo cual potencia la reutilización y disminuye la dependencia entre estas.

En Symfony2 se evidencia en la práctica la utilización de este patrón ya que las clases controladoras heredan de la clase Controller.php alcanzando de esta manera un bajo acoplamiento. Las clases que implementan la lógica del negocio y de acceso a datos se encuentran definidas en el modelo, y no tienen asociaciones con las pertenecientes a la vista o al controlador, lo que proporciona bajas dependencias.

Decorador: El patrón Decorador responde a la necesidad de añadir dinámicamente funcionalidades a un Objeto. Esto permite no tener que crear sucesivas clases que hereden. La clave de este modelo radica en la importancia de la ejecución composicional. La herencia es una clara forma de construir sobre las características establecidas por una clase padre. En Symfony2 este patrón es fácilmente visible, ya que, la vista se separa por niveles, en hasta 3 niveles, una plantilla base y varias plantillas que heredan de esta.

Normalmente, la plantilla base es global en toda la aplicación y contiene el código HTML que es común a la mayoría de las páginas. De esta forma el patrón se manifiesta en la propuesta de solución en cada una de las vistas del sistema al heredar de una misma clase padre.

Se puede observar el uso de este patrón en la propuesta de solución en todas las plantillas, ya que todas heredan de una plantilla padre determinada, estableciendo una herencia de tres niveles. En el presente segmento de código fuente se ejemplifica lo antes descrito.

index.html.twig

```
{  
% extends 'UIBundle:Layout:layout_subsystems.html.twig' %  
}
```

Modelo de Diseño

El modelo de diseño se utiliza como medio de abstracción del modelo de implementación y el código fuente del *software*. Su objetivo fundamental es transmitir, a través de la representación mediante diagramas, una comprensión en profundidad de los aspectos relacionados con los requerimientos no funcionales y restricciones concernientes a los lenguajes de programación (Larman, 2003).

Diagramas de clases del diseño

Un Diagrama de Clases de Diseño muestra la especificación para las clases de una aplicación, se centra en mostrar las definiciones entre entidades de un *software*. En su representación incluye datos de interés como: clases, asociaciones y atributos, interfaces, con sus operaciones y constantes, métodos, navegabilidad y dependencias.

En las siguientes figuras se muestran ejemplos de Diagramas de Clases de Diseño pertenecientes al módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión.

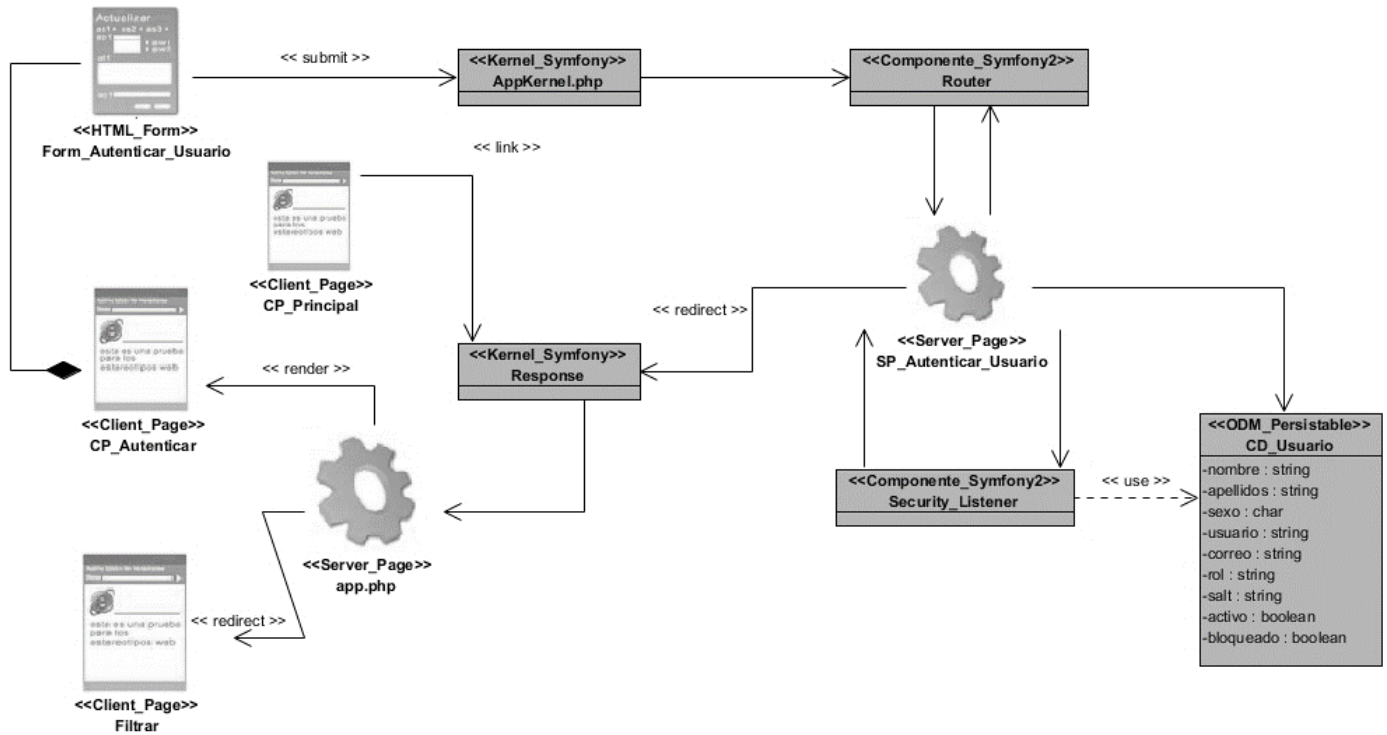


Figura 3: Diagrama de clases del diseño Autenticar usuario

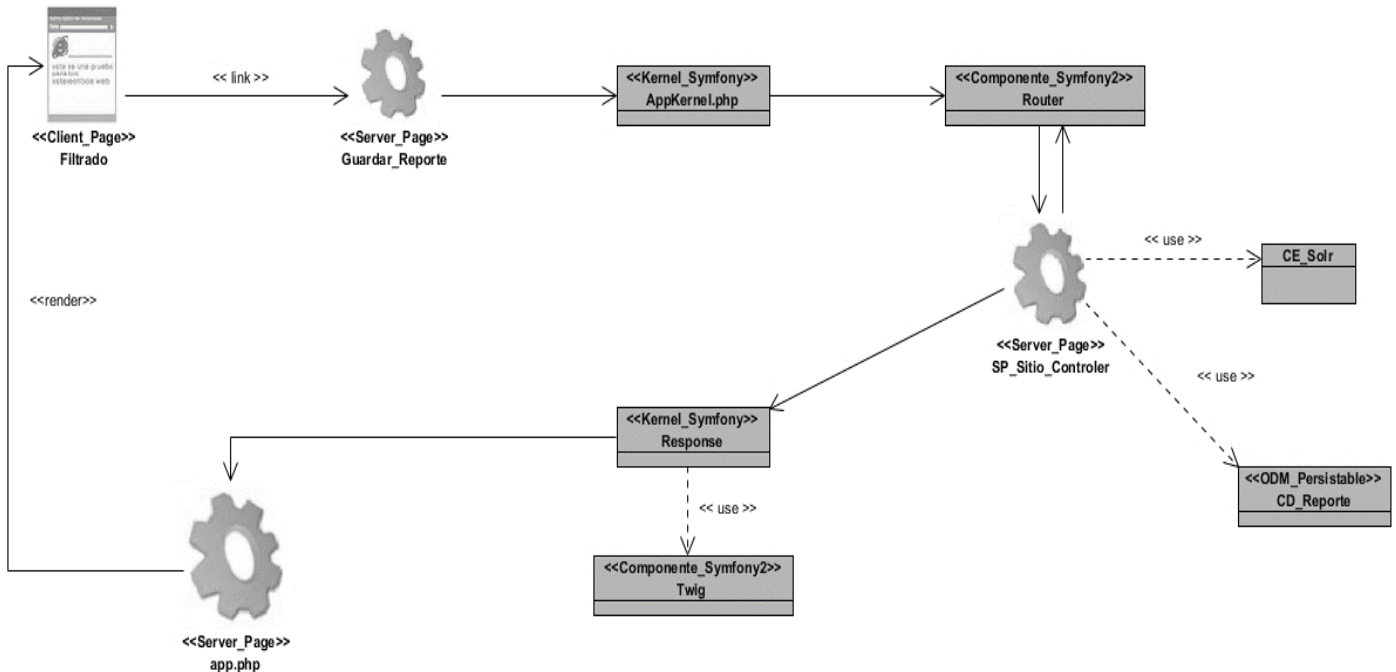


Figura 4: Diagrama de clases del diseño Guardar Reporte

Modelo de despliegue

Un Diagrama de Despliegue modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de *hardware* (nodos) y muestra cómo los elementos y artefactos del *software* se relacionan en esos nodos (SparxSystems, 2014).

El diagrama de despliegue se utiliza para mostrar la estructura física del sistema, incluyendo las relaciones entre el *hardware* y el *software* que se despliega, estas relaciones son representadas por los protocolos de comunicación que se utilizan para acceder a cada uno. En la siguiente figura puede visualizarse el diagrama de despliegue definido para la solución propuesta:

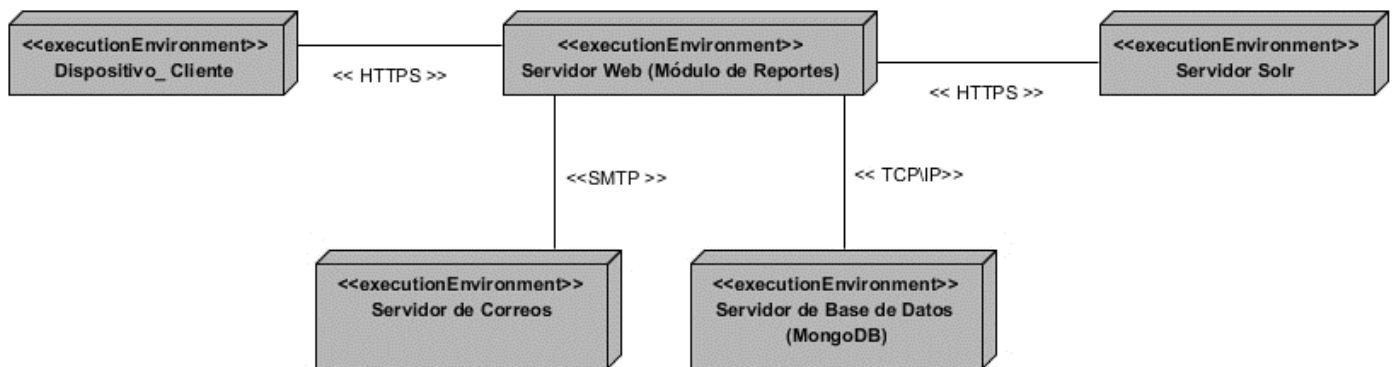


Figura 5: Diagrama de despliegue

2.6 Conclusiones del capítulo

Con la especificación de los requisitos funcionales y no funcionales del sistema, se logró una mejor comprensión, por parte del autor, de los resultados que se pretenden obtener y sirvió de guía para la implementación del módulo. La representación y descripción de los artefactos generados garantizaron un mejor entendimiento de los flujos de trabajos presentes en el proceso de generar el reporte estadístico. La definición de la arquitectura y los patrones de diseño a utilizar, permitieron establecer las bases para fomentar la reutilización y las buenas prácticas de programación entre los desarrolladores durante la fase de implementación, así como disminuir el impacto de los cambios futuros en el código fuente. La elaboración del diagrama de despliegue permitió identificar la disposición física de los artefactos del producto informático a desarrollarse.

Capítulo 3: Implementación y prueba del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión

3.1 Introducción

La etapa de implementación constituye una parte crucial durante el proceso de desarrollo de un *software*, es en este momento donde se define y organiza el código de la propuesta de solución. Es durante esta etapa que se materializan, en forma de código, de todos los artefactos de implementación, descripciones y arquitectura propuestos en la etapa de análisis y diseño; conformando así el producto final requerido por el cliente.

Todo *software* debe ser puesto a prueba, garantizando así que cumpla con todos los estándares requeridos y con todas aquellas condiciones y funcionalidades que el cliente final necesita. A esta etapa se le conoce como validación del sistema y en ella, pueden realizarse diferentes tipos de pruebas en función de los objetivos de las mismas.

3.2 Modelo de implementación

Regido bajo los estándares definidos por UML el modelo de implementación evidencia la forma en que es estructurado un sistema informático atendiendo a las diferentes partes que conforman su estructura; a la vez, muestra las dependencias existentes entre ellas. El modelo de componentes permite visualizar la estructura de alto nivel del sistema y el comportamiento del servicio que estos componentes proporcionan y usan a través de interfaces (Microsoft, 2016). Este puede considerarse también como un acercamiento basado en la reutilización para definir, implementar y agrupar componentes que se encuentran débilmente acoplados en el sistema.

3.3 Diagrama de componentes

El diagrama de componentes muestra la relación entre componentes de *software*, sus dependencias, su comunicación y su ubicación. En la figura 6 se ilustra el diagrama de componentes del módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión. En el mismo se puede apreciar que el paquete Reporte está dividido en tres paquetes como los requiere el patrón arquitectónico Modelo-Vista-Controlador el cual es utilizado para el desarrollo del módulo. Dentro de cada paquete se puede apreciar cada uno de

los componentes que lo integran. En el diagrama también se representan las dependencias a otros componentes, que el sistema necesita para realizar determinadas funcionalidades las cuales son, KnpPaginatorBundle usado en la paginación, SpraedPdfBundle utilizado para convertir una página HTML en un documento de formato PDF, DoctrineMongoDBBundle y NelmioSolariumBundle utilizados en el acceso a datos, trabajan con el gestor de base de datos MongoDB y con el servidor de indexación Solr respectivamente.

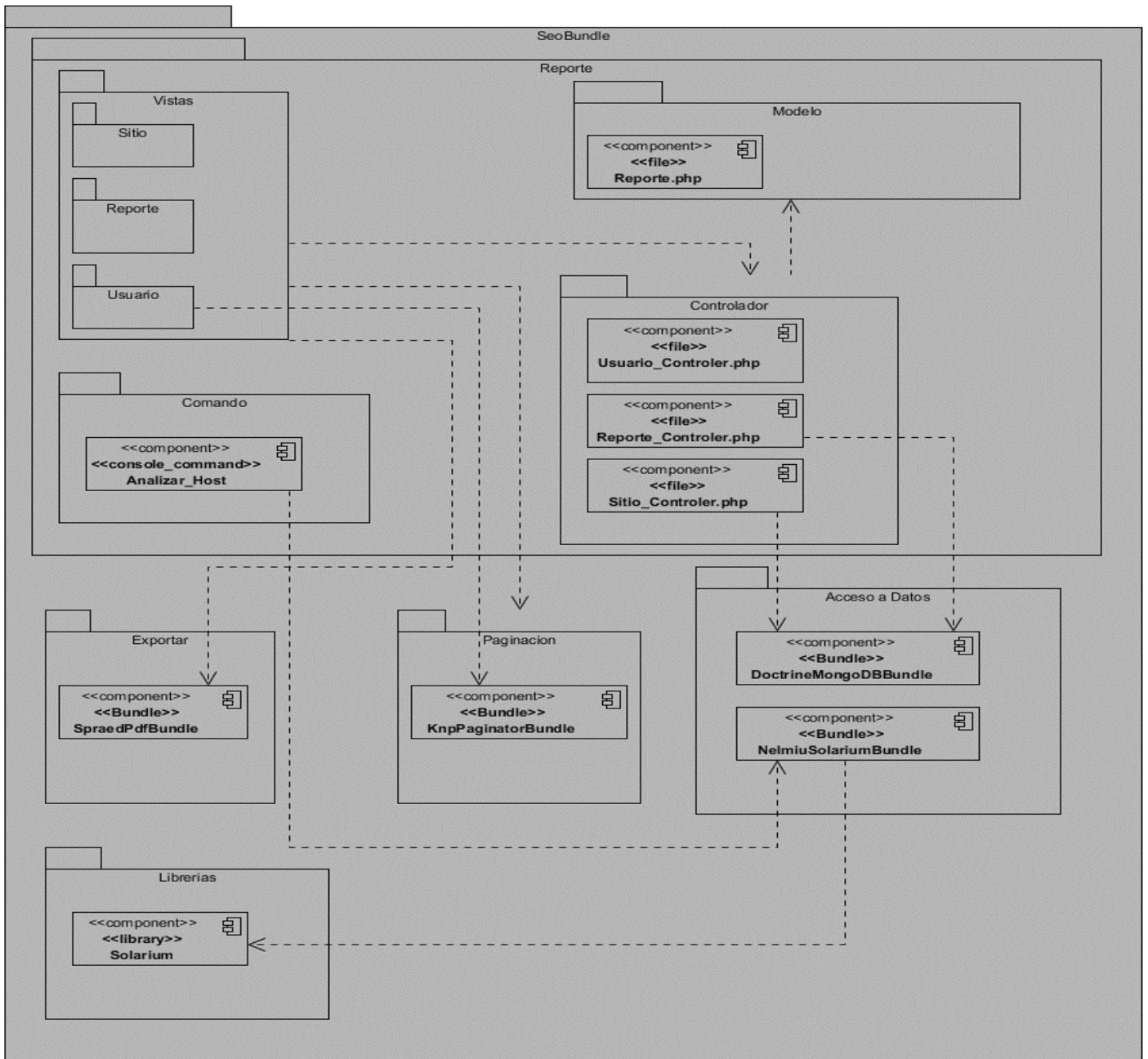


Figura 6: Diagrama de componentes del módulo Generador de Reportes Webmétricos

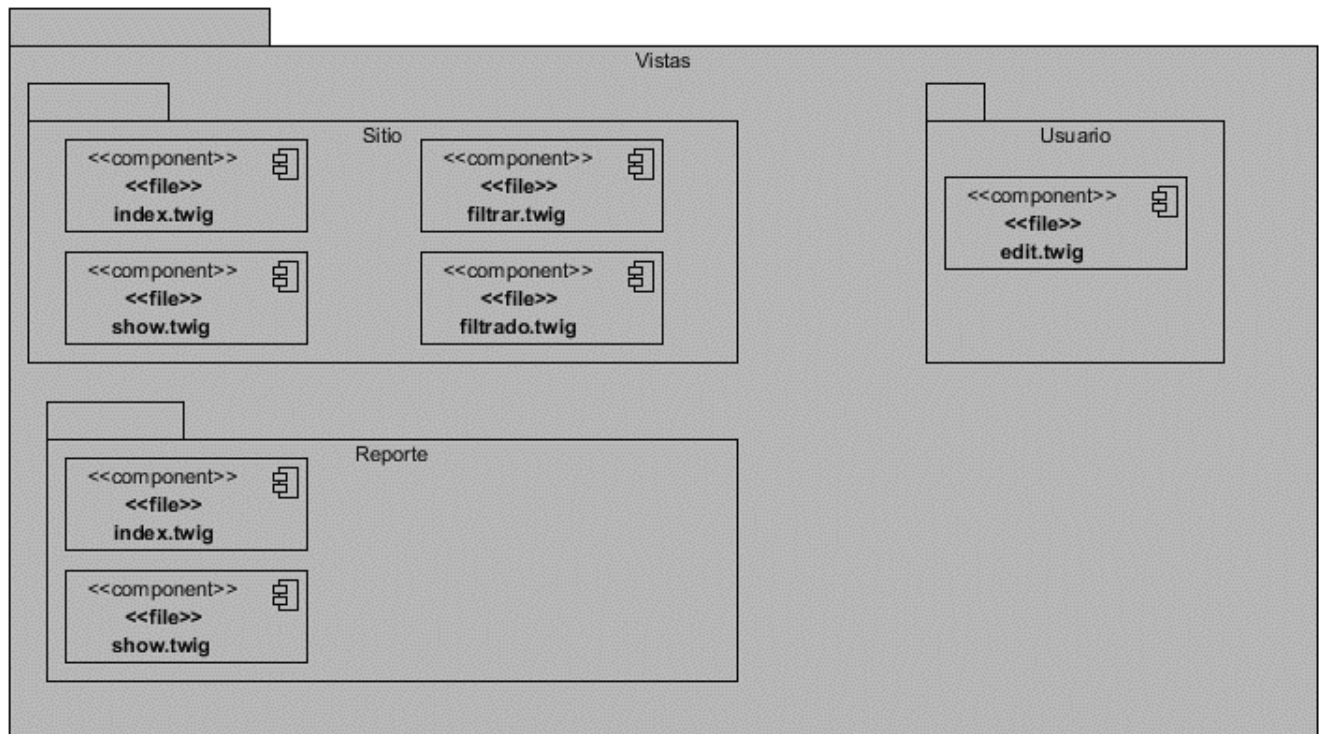


Figura 7: Especificación del componente Vistas del Diagrama de componentes del módulo Generador de Reportes Webmétricos.

Descripción de los componentes generales de la propuesta de solución:

Tabla 5: Descripción de los componentes.

Componente	Descripción
Vistas	Agrupar las páginas referentes a las vistas de la aplicación.
Modelo	Contiene la información básica de los host indexados a los cuales en algún momento se le hizo un respaldo de su información en el sistema.
Controlador	Contiene la clase controladora encargada de procesar las peticiones de las páginas clientes y del usuario según figura 2: <i>Esquema simplificado de la arquitectura interna de Symfony</i> y devolver las respuestas con la información requerida.
Comando	Comando de consola de Symfony el cual se ejecuta en la propia terminal del sistema y se encarga del filtrado de host y el análisis del estado de los enlaces entrantes y salientes del mismo.
Acceso a Datos	Contiene los bundles necesarios para la comunicación de Symfony con la base de datos y Solr, así como las consultas a los mismos.
Paginación	El KnpPaginatorBundle se encarga de la paginación de los resultados, mejorando de esta forma el rendimiento.
Exportar	El SpraedPdfBundle se encarga de la generación de PDF a partir de HTML, lo cual facilita en gran medida su personalización.
Librerías	Solarium es una librería para la comunicación entre PHP y Solr.

3.4 Estándares de codificación utilizados

La legibilidad del código fuente repercute directamente en lo bien que un programador comprende un sistema de *software*, sobre todo si se trata de un *software* que incorpore código fuente previo, o bien cuando se realice el mantenimiento de un sistema de *software* creado anteriormente. Una buena práctica que potencia la legibilidad del código es la utilización de estándares de codificación. Un estándar de codificación comprende todos los aspectos de la generación de código, la legibilidad del código, reflejando un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez.

A continuación, se mencionan las principales prácticas utilizadas para el uso del código del lenguaje PHP especialmente dirigido al trabajo con el framework Symfony2.

1. Nunca utilizar las etiquetas cortas (<?).
2. No finalizar las clases con la etiqueta de cierre (?>).
3. La indentación se realiza utilizando cuatro espacios (las tabulaciones no están permitidas).
4. Agregar un único espacio después de cada delimitador coma.
5. No poner espacios después de la apertura de un paréntesis y antes del cierre del mismo.
6. Agregar un único espacio alrededor de operadores (==, &&).
7. Agregar un único espacio antes de los paréntesis de apertura de una palabra clave de control (if, else, for, while).
8. Agregar una línea en blanco antes de la sentencia return.
9. No agregar espacios al final de las líneas.
10. Utilizar llaves para indicar el cuerpo de las estructuras de control sin importar el número de sentencias que estas contengan.
11. Colocar las llaves en sus propias líneas para clases, métodos y declaración de funciones.
12. Separar las sentencias condicionales y las llaves de apertura con un único espacio sin dejar una línea en blanco.
13. Declarar explícitamente la visibilidad de las clases, métodos y propiedades (el uso de var está prohibido).
14. Utiliza constantes de tipo PHP nativas en minúsculas: false, true y null. Lo mismo aplica para los array().
15. Utilizar letras mayúsculas para constantes, con palabras separadas por un guión bajo.

16. Definir una clase por archivo.
17. Declarar las propiedades de las clases antes de los métodos.
18. Declarar los métodos públicos primero, luego los protegidos y finalmente los privados.
19. Convención de Nombres.
20. Utilizar camelCase y no guiones bajos, para variables, funciones y nombres de métodos.
21. Utilizar guiones bajos para definir opciones, argumentos y nombres de parámetros.
22. Utilizar los namespace para todas las clases.
23. Utilizar Symfony como el namespace de primer nivel.
24. Añadir como sufijo Interface a las interfaces.
25. Utilizar caracteres alfanuméricos y guiones bajos para nombres de archivos.

3.5 Validación del módulo

En la ingeniería de *software* es el proceso de revisión que verifica que el sistema de *software* producido cumple con las especificaciones y que logra su cometido, se trata de evaluar el sistema o parte de este durante o al final del desarrollo para determinar si satisface los requisitos iniciales.

Pruebas funcionales

Constituyen pruebas de *software* aquellas que tienen por objetivo probar que los sistemas desarrollados cumplan con las funciones específicas para las cuales han sido creados (Pressman, 2005). A este tipo de pruebas se les denomina también pruebas de comportamiento y para aplicarlas a la propuesta de solución se emplea el método de caja negra, donde los probadores o analistas de pruebas no enfocan su atención en cómo se generan las respuestas del sistema, sino en el funcionamiento de la interfaz. Básicamente el enfoque de este tipo de prueba se basa en el análisis de los datos de entrada y salida, esto generalmente se define en los casos de prueba preparados antes del inicio de las pruebas.

El objetivo final es garantizar que los requerimientos hayan sido cumplidos y que el sistema goce de buena calidad. A continuación, se muestra una parte de la propuesta de diseño de casos de prueba para algunos requisitos con prioridad alta a realizar al módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión.

Tabla 6: Variables empleadas en el diseño del caso de prueba para el escenario Registrar usuario.

No	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de Texto	No	Se deben escribir palabras comenzando por mayúscula y sin caracteres extraños, que no excedan un total de 60 caracteres.
2	Apellidos	Campo de Texto	No	Se deben escribir palabras comenzando por mayúscula y sin caracteres extraños, que no excedan un total de 60 caracteres.
3	Usuario	Campo de Texto	No	Se debe escribir una palabra sin caracteres extraños, todos en minúscula y que no excedan un total de 60 caracteres.
4	Correo	Campo de Texto	No	Se debe insertar una cadena de texto, que coincida con la estructura válida de un correo.
5	Contraseña	Campo de Texto	No	Se debe insertar una cadena de texto que no exceda de 60 caracteres.
6	Verificar Contraseña	Campo de Texto	No	Se debe insertar una cadena de texto que no exceda de 60 caracteres, idéntica a la del campo "Contraseña".
7	Captcha	Campo de Texto	No	Se debe insertar una cadena de texto que coincida con la mostrada en la imagen.

Las celdas de la tabla contienen V, I, o N/A. V indica válido, I indica inválido, y N/A que no es necesario proporcionar un valor del dato en este caso, ya que es irrelevante.

Tabla 7: Caso de pruebas para el escenario Registrar usuario

Escenario	Descripción	V1	V2	V3	V4	V5	V6	V7	Respuesta del Sistema	Flujo central
EC 1.1 Datos correctos.	El Usuario introduce los datos de forma correcta en el sistema.	V	V	V	V	V	V	V	El sistema registra al usuario.	El Usuario rellena los campos de forma correcta y da clic en el botón Aceptar.
		Duanny	Sánchez China	dchina	dchina@estudiantes.uci.cu	Tesi.1234	Tesi.1234	1234		
EC 1.2 Campos vacíos.	El usuario deja uno o más campos vacíos.	V	V	I	I	V	I	V	El sistema muestra un mensaje "Por favor rellene este campo".	El usuario llenando los campos deja uno o más vacíos y da clic en el botón aceptar.
		Duanny	Sánchez China	vacío	vacío	Tesi.1234	vacío	1234		
		V	V	V	V	I	I	I		
		Duanny	Sánchez China	dchina	dchina@estudiantes.uci.cu	vacío	vacío	vacío		
		I	V	V	V	V	V	V		
		vacío	Sánchez China	dchina	dchina@estudiantes.uci.cu	Tesi.1234	Tesi.1234	1234		
		I	I	I	I	I	I	V		
		vacío	vacío	vacío	vacío	vacío	vacío	1234		
EC 1.3	El usuario	V	V	I	I	V	I	V	El sistema	El

Generador de Reportes Webmétricos para el motor de búsqueda Orión

Datos incorrectos.	introduce algún valor incorrecto en el sistema.	Duanny	Sánchez China	#@€	Dchina.cu	Tesi.1234	Te.123	1234	muestra un mensaje "Datos Incorrectos".	usuario llena los campos y da clic en el botón Aceptar.
		V	V	V	V	I	I	V		
		Duanny	Sánchez China	dchina	dchina@estudiantes.uci.cu	4521	qwer	1234		
		I	V	V	V	V	V	V		
		45	Sánchez China	dchina	dchina@estudiantes.uci.cu	Tesi.1234	Tesi.1234	1234		
		I	I	I	I	I	I	V		
		123	~€~@	54p~@#	@.cu	Ted21	Lol85	1234		

Resultados de las pruebas funcionales

Las pruebas funcionales se realizaron en tres iteraciones a medida que avanzaba el sistema, las cuales guiaron la calidad del sistema, y determinaron en cada momento si se estaba o no en condiciones de continuar avanzando con el ciclo de desarrollo. A continuación, se muestran los resultados de las pruebas realizadas en cada iteración, las cuales arrojaron un total de 18 no conformidades que fueron corregidas satisfactoriamente, siendo los principales errores detectados los ortográficos, textos en idioma inglés, validaciones incorrectas y la falta de botones para deshacer acciones.

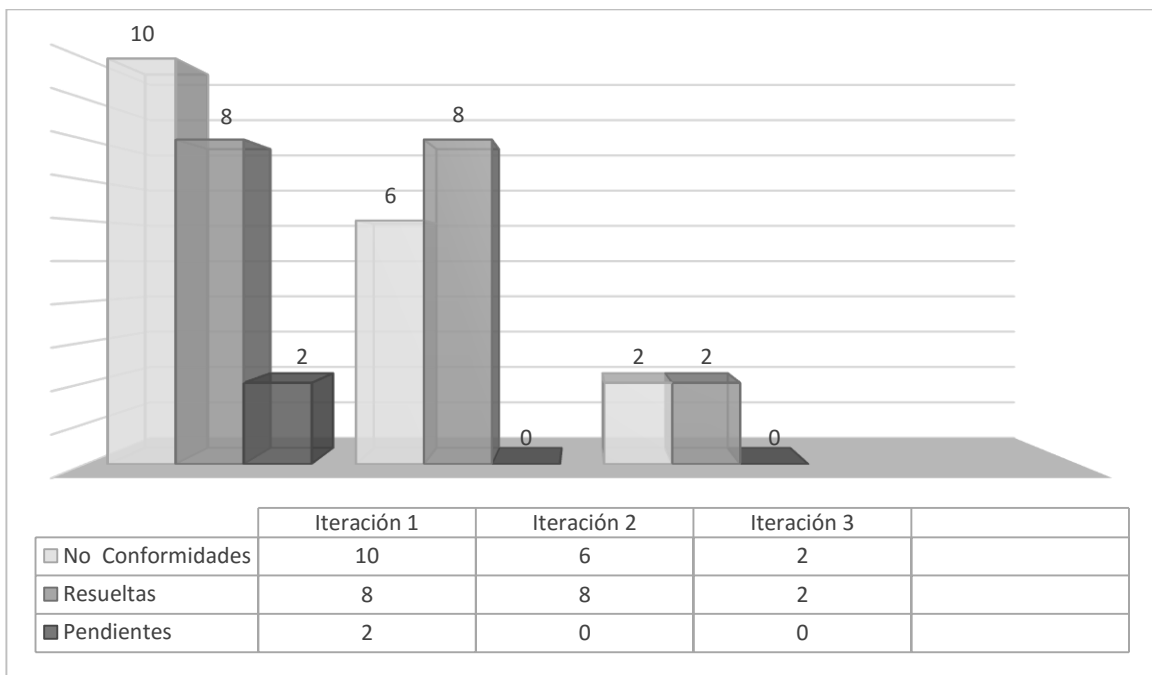


Figura 8: Resultados de las pruebas funcionales.

La anterior gráfica desglosa las no conformidades detectadas en las tres iteraciones realizadas. En una primera iteración se recoge un total de 10 no conformidades de las cuales se resolvieron 8 y las restantes 2 se quedaron pendientes para la segunda iteración, esta su vez arrojó un total de 6 no conformidades más, las cuales fueron resueltas en su totalidad agregando las dos pendientes de la iteración anterior. Por último, una tercera iteración la cual arrojó un total de 2 nuevas no conformidades, ambas terminaron siendo resueltas, finalizando así con un total de 18 no conformidades llevadas a feliz término.

Pruebas de seguridad

Realizarlas pruebas adecuadas a todo el *software* no es un asunto trivial, ya que son los procesos que permiten verificar y revelar la calidad de un producto. Las pruebas de seguridad son utilizadas para identificar posibles fallos en la integridad del sistema, entre otros aspectos garantizan que los usuarios estén restringidos a funciones específicas o que su acceso esté limitado únicamente a los datos que están autorizados a acceder. Únicamente aquellos usuarios con autorización a acceder al sistema son capaces de ejecutar las funciones disponibles. El objetivo fundamental de este tipo de pruebas es comprobar los niveles de seguridad lógica del sistema.

La UCI ha establecido tres niveles para realizar las pruebas de seguridad. Para evaluar la seguridad de las aplicaciones en un primer nivel (nivel 1), los especialistas del grupo de Seguridad del DEPSW¹⁶ definieron una lista de chequeo que cuenta con 15 indicadores agrupados en cuatro tipos de pruebas accesibles a los tres niveles.

Resultados de las pruebas de seguridad

Al aplicar las pautas definidas en dicha lista, para los cuatro tipos de pruebas, los siguientes resultados:

- 1. Pruebas de Autorización:** Ningún usuario estándar puede modificar sus privilegios ni los de otro usuario en la aplicación.
- 2. Pruebas de Gestión de Sesiones:** No se puede acceder a la aplicación copiando la URL después de estar autenticado, cerrar el navegador y volver a abrirlo. De igual manera tampoco se puede acceder al cerrar la sesión de un usuario y dar clic en el botón “Ir a la página anterior” del navegador.
- 3. Validación de Datos:** Se enmascaran datos confiables cuando se visualizan en la aplicación. Solamente se permiten contraseñas alfanuméricas, que incluyan caracteres especiales y que tengan seis caracteres, como mínimo de longitud. La funcionalidad de cambio de contraseña únicamente se permite a usuarios autenticados validando la antigua contraseña y la nueva contraseña. El sistema no muestra mensajes

¹⁶ Departamento de *software*.

indebidos al colocar en la barra de dirección o en campos de entrada los caracteres: comillas simples (‘), signos de ampersand (&), o signos: + - /.

4. Comprobación del Sistema de Autenticación: Los mensajes de error para distintas combinaciones de autenticación muestran la misma información. Los tiempos de respuestas usuario correcto – contraseña incorrecta y usuario incorrecto - contraseña incorrecta son los mismos. El sistema protege el envío de los datos mediante protocolo seguro (HTTPS). La sesión de usuario se bloquea después de 15 minutos de inactividad y se desactiva la cuenta del usuario después de tres intentos de autenticación fallidos.

Esta lista de chequeo permitió recoger los puntos eficientes e ineficientes que tienen los elementos comprobados, así como también la verificación de que el grado de seguridad de la aplicación es adecuado para la protección de la información.

Para medir la seguridad del sistema en un segundo nivel (nivel 2) se utilizó la herramienta Acunetix WVS v8. Durante el chequeo realizado por la herramienta fueron detectadas 15 no conformidades, de ellas 8 de prioridad baja, una de prioridad media y seis informativas. Entre las no conformidades de prioridad baja se encontraron: ficheros del servidor Apache vulnerables a ataques de fuerza bruta y cookies de sesión sin el indicador de seguridad, estas no conformidades no fueron corregidas ya que no se encontraban dentro del alcance del equipo de trabajo. La no conformidad de prioridad media se encuentra relacionada con el auto completamiento de los formularios del sistema, por tanto, fue corregida. El resto deja al descubierto información acerca de una página de error que contiene la versión del servidor Web y una lista de los módulos habilitados en este servidor, lo cual no constituye un problema propio del sistema desarrollado, sino del servidor donde se ejecuta.

Pruebas de rendimiento

Mediante las pruebas de rendimiento es posible hallar tendencias y comportamientos para los elementos de una aplicación. Este tipo de pruebas permite identificar cuellos de botella, capacidad de concurrencia de usuarios, tiempos de respuesta de operaciones del negocio a nivel de sistema, establecer un marco de referencia para pruebas futuras, determinar el cumplimiento de los objetivos de rendimiento y requerimientos no funcionales, entre otros (V&V Quality, 2013).

Pruebas de Carga: Mediante la ejecución de las pruebas de Carga es posible identificar la capacidad de recuperación de un sistema cuando es sometido a cargas variables, tanto de usuarios como de procesos. Este tipo de pruebas pueden determinar el tiempo de respuesta de todas las transacciones críticas del sistema y encontrar cuellos de botella de la aplicación (V&V Quality, 2013).

Prueba de Estrés: Mediante las pruebas de Estrés es posible identificar la capacidad de respuesta de un sistema bajo condiciones de carga extrema, representadas por una alta concurrencia de usuarios y/o procesos. Una vez realizadas las pruebas de estrés se podrá conocer el punto de quiebre del módulo en términos de capacidad de respuesta, con lo cual será posible establecer acciones de optimización en diferentes niveles para asegurar una mejor capacidad de concurrencia de usuarios y procesos, que se verá reflejada en una óptima operación de negocio (V&V Quality, 2013).

Resultados de las pruebas de rendimiento

Para las pruebas de Carga y Estrés realizadas al módulo Generador de Reportes Webmétricos para el motor de búsqueda Orión fue utilizada la herramienta *JMeter* v2.8.4. Durante la realización de las pruebas se simuló el entorno donde debe interactuar el módulo para obtener una correcta información de comportamiento y resultados en general. Por lo que se eligió un escenario con las siguientes características.

Hardware:

- Tipo de procesador: corei5 2450M a 2.50 GHz
- Memoria: 6GB RAM.
- Tipo de Red: Ethernet 10/100Mbps.

Software:

- Tipo de Servidor Web: Apache v2.4.
- Máximo de hilos concurrentes (simulación de usuarios): 100.
- Plataforma: sistema operativo Linuxmint 17.2.
- Servidor de BD: MongoDB 2.4.9.

Se añadirá el elemento “Informe Agregado” que brindará la siguiente información:

- **Usuarios:** total de usuarios.
- **# Muestras:** El número de peticiones.
- **Media:** El tiempo medio transcurrido para un conjunto de resultados.
- **Mín:** El mínimo tiempo transcurrido en milisegundos para las muestras de la URL dada.
- **Máx:** El máximo tiempo transcurrido en milisegundos¹ para las muestras de la URL dada.
- **% Error:** Porcentaje de las peticiones con errores.
- **Rendimiento:** Rendimiento medido en base a peticiones por segundo/minuto/hora.

Tabla 8: Resultados de las pruebas de carga y estrés del escenario Exportar Reporte.

Usuarios	# Muestras	Mín	Máx	% Error	Rendimiento (peticiones/segundos)
25	425	1	4481	0	32,1
50	850	1	7744	0	29,41
100	2550	1	12215	0	12,6

Atendiendo a la cantidad de peticiones por cada segundo que se enviaron y las prestaciones del *hardware* donde se realizaron las pruebas, se considera que los resultados alcanzados son buenos y considerablemente superables si se mejoran las características del entorno de prueba.

Pruebas de integración

Este tipo de pruebas consiste en la comprobación de que los elementos del *software* que interactúan entre sí, funcionan de manera correcta. Es una forma de chequear la correcta interrelación de los distintos componentes del sistema; en el caso de la solución desarrollada es la verificación de una correcta interoperabilidad entre el módulo desarrollado y el buscador Orión.

Durante la integración del módulo desarrollado con el buscador Orión fueron seguidos una serie de pasos tal y como se describe a continuación:

1. Añadir el *bundle* que constituye la solución, llamado SeoBundle, a la ruta donde se encuentra el resto de los bundles que componen el buscador Orión: **ORION2.1\Source Files\src\ORION\Bundle**.

2. Crear las rutas necesarias en el archivo **routing.yml** ubicado en el directorio **\app\config\routing.yml** (ver anexo 2) para que el buscador Orión reconozca los distintos enlaces con el bundle SeoBundle.
3. Importar la base de datos y crear las tablas necesarias. Desde la terminal que ofrece Symfony2 se deben ejecutar los siguientes comandos:

```
doctrine:mongodb:fixtures:load  
doctrine:mongodb:database:create  
assets:install
```

Estos permitirán cargar de forma dinámica una entrada en la tabla usuario (el administrador por defecto), crear las tablas necesarias en la base de datos e instalar los java script, css y assets necesarios para que funcione el módulo que constituye la solución dentro de la carpeta “web” respectivamente.

4. Añadir el enlace del módulo a la interfaz principal de Orión.
5. Configurar en el módulo de gestión de usuarios, los roles específicos del módulo solución.

Los roles en el sistema de autenticación están predefinidos, por tanto, solo se debe definir en el fichero **app/config/security.yml** el proveedor de usuario que se va a utilizar modificando la línea que se muestra (ver anexo 3) por la del proveedor de usuario seleccionado, luego se deben añadir los controles de acceso para cada rol con acceso al módulo (ver anexo 4).

Resultados de las pruebas de integración

Luego de realizados los pasos anteriormente descritos se pudo comprobar que el módulo Generador de Reportes Webmétricos es completamente funcional junto al motor de búsqueda Orión, no encontrándose ninguna no conformidad. Por tanto, se concluye que existe una correcta integración entre los componentes internos del sistema.

3.6 Conclusiones del capítulo

En el presente capítulo se especificó cómo está construido el sistema a partir del diagrama de componentes, lo cual permitió identificar con claridad la estructura y las relaciones que existen entre los diferentes componentes empleados en la implementación del módulo Generador de Reportes Webmétricos. La utilización de estándares de código para la implementación de la propuesta de solución permitió adoptar una estructura homogénea que facilita la comunicación y asegura la calidad, una menor cantidad de errores y el fácil mantenimiento; logrando un código más limpio y fácil de mantener, así como una mayor comunicación entre el programador y futuros terceros. Por último, la detección temprana de errores, mediante la aplicación de pruebas de funcionalidad, carga y estrés, así como las pruebas de seguridad permitieron validar que el sistema trabaja como inicialmente fue diseñado, que no se incurrió en la violación de ningún requisito y que la implementación fue correcta.

Conclusiones

Una vez completada la presente investigación, se puede concluir que:

- La implementación del módulo Generador de Reportes Webmétricos utilizando las herramientas y tecnologías estudiadas y orientada por la metodología AUP (UCI) permitió solucionar los problemas existentes planteados en la problemática de la presente investigación.
- Mediante la generación de reportes con datos estadísticos e información especializada sobre el estado general de los sitios Web se logró simplificar el proceso de toma de decisiones por parte de los webmaster, mejorando con ello el posicionamiento Web.
- A partir del estudio realizado sobre los sistemas homólogos se determinó la necesidad de implementar un Módulo Generador de Reportes Webmétricos, ya que no se encontró ningún sistema de licencia libre que cumpla con las necesidades del cliente, definiéndose una propuesta de solución de acuerdo a las necesidades existentes.
- La evaluación de las pruebas de *software* realizadas permitió erradicar las insuficiencias detectadas en el módulo desarrollado, logrando así un producto más seguro y funcional conforme a las necesidades de los usuarios finales.

Recomendaciones

Una vez concluida la investigación y el desarrollo de la propuesta de solución, el autor del presente trabajo recomienda:

- Desarrollar nuevas funcionalidades que aporten al sistema características que contribuyan a mejorar la información disponible para los *webmasters*, como por ejemplo detectar imágenes sin texto alternativo, detectar contenidos duplicados y obtener el estado de los enlaces entrantes y salientes.
- Utilizar servicios *sphinx*¹⁷ para el filtrado de usuarios en vez de las actuales consultas *like*¹⁸, lo cual aumentará el rendimiento del sistema.

¹⁷ Sphinx es un servidor de búsqueda de texto completo multiplataforma y de código abierto, escrito en C++ y diseñado desde cero con enfocado en el alto rendimiento, la relevancia y la simplicidad de integración (Sphinx, 2015).

¹⁸ Utilizadas en bases de datos, es un operador relacional que se usa para realizar comparaciones exclusivamente de cadenas (Alvarez, 2007).

Referencias Bibliográficas

Alfonzo, M. A. *Analizador Inteligente de Registros Proxy*. [En línea] Universidad de las Ciencias Informáticas. 2012 [Consultado el 18 de diciembre del 2015] Disponible en http://repositorio_institucional.uci.cu/jspui/handle/ident/JCE-2012-F318-P232-Ponencia-2896.

Alvarez, M. A. *Qué es php*. [En línea] Desarrolloweb. 2001 [Consultado el: 27 de noviembre de 2015.] Disponible en: <http://www.desarrolloweb.com/articulos/392.php>.

Alvarez, M. A. *Introducción a CSS 3*. [En línea] Desarrolloweb. 2008. [Consultado el: 27 de noviembre de 2015.] Disponible en: <http://www.desarrolloweb.com/articulos/introduccion-css3.html>.

Alvarez, M. A. *Qué es ROI*. [En línea] Desarrolloweb. 2009. [Consultado el: 20 de noviembre de 2015.] Disponible en: <http://www.desarrolloweb.com/articulos/que-es-roi.html>.

Alvarez, S. *Sistema Gestor de Base de Datos*. [En línea] Desarrolloweb, 2007. [Consultado el: 2 de diciembre de 2015.] Disponible en: <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.

Ambyssoft. *Simple Tools for Software Modeling -OR- It's "Use the Simplest Tool" not "Use Simple Tools"* [En línea] Agile Modeling. 2014 [Consultado el: 3 de diciembre de 2015.] Disponible en: <http://www.agilemodeling.com/essays/simpleTools.htm#SelectingCASE>.

Andalia, R. C. *Aproximaciones para una historia de Internet*. [En línea] ACIMED.2004 [Consultado el: 30 de octubre del 2015.] Disponible en: http://scielo.sld.cu/scielo.php?pid=S1024-94352004000100005&script=sci_artt Apache Solr.

Apache. *Apache server project*. [En línea] Apache Software Foundation. 2015 [Consultado el 2 de diciembre del 2015.] Disponible en: <http://httpd.apache.org/>.

Balarezo, J. *Metodologías Ágiles, Programación Extrema XP*. Facultad de Ciencias Físicas y Matemáticas, Universidad Nacional de Trujillo,2013.

Björneborn, L. *Small-world link structures across an academic web space: a library and information science approach*. Copenhagen: Department of Information Studies, Royal School of Library and Information Science, 2004.

Buigas, L. Z.; Avilés; A. S. *Elementos conceptuales básicos útiles para comprender las redes de telecomunicación*. [En línea] ACIMED, 2002. [Consultado el: 15 de octubre de 2015.] Disponible en: http://scielo.sld.cu/scielo.php?pid=S1024-94352002000600003&script=sci_arttext.

Camiño, R. R. *Motores de búsqueda sobre salud en Internet*. [En línea] ACIMED. 2003 [Consultado el: 15 de octubre de 2015.] Disponible en: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352003000500002.

Castro, L. *¿Qué es URL?* [En línea] About en español. 2016 [Consultado el: 30 de marzo de 2016.] Disponible en: <http://aprenderinternet.about.com/od/ConceptosBasico/a/Que-Es-Url.htm>.

CCM. *Formato PNG*. [En línea] CCM. 2014 [Consultado el: 24 de mayo de 2016.] Disponible en: <http://es.ccm.net/contents/723-formato-png>.

Chiavenato, I. *Introducción a la Teoría General de la Administración*. S.l.: McGraw-Hill Interamericana, 2006. 298 p.

Codina, L. *El componente de visibilidad en la Web*. [En línea] Document.mx. 2007 [Consultado el: 17 de noviembre de 2015.] Disponible en: <http://documents.mx/documents/posicionamiento-web-el-componente-de-visibilidad-en-la-web-lluis-codina-upf-estudios-de-periodismo-febrero-2007.html>

Codina, L. *Posicionamiento Web para comunicadores: Análisis y Métodos*. [En línea] Document.mx. 2008 [Consultado el: 17 noviembre de 2015.] Disponible en: <http://documents.mx/documents/lluis-codina-upfidec-abril-2008-posicionamiento-web-tercera-parte-analisis-y-metodos.html>

Cornejo, J. E. *¿Qué es UML?* [En línea] DocIRS. 2008 [Consultado el: 12 de diciembre de 2015.] Disponible en <http://www.docirs.cl/uml.htm>

Corona, S. *A Practical Guide To High Performance*. O'Reilly Media, 2015. 250 p.

Couto, J. R. *MongoDB: Características y futuro*. [En línea] MongoDB Spain. 2014. [Consultado el: 2 de diciembre de 2015.] Disponible en: <http://www.mongodbspain.com/es/2014/08/17/mongodb-characteristics-future/>.

Czinkota, M .R; Kotabe M. *Administración de Mercadotecnia*. S.A. Ediciones paraninfo, 2005. 600 p.

Delgado, Y. H. *Orión, un motor de búsquedas para la web de la UCI*. [En línea] Universidad de las Ciencias Informáticas, 2010. [Consultado el 13 de noviembre del 2015] Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_03440_10/1/TD_03440_10.pdf.

Elizalde, E. *Los buscadores más populares de Internet*. [En línea] About en español, 2016. [Consultado el 3 de marzo del 2016] Disponible en: <http://buscadores.about.com/od/conceptosbasicos/tp/Los-Buscadores-M-As-Populares-De-Internet.htm>.

Encinoza, L. J. *Sistema de información para el economista y el contador*. Editorial Académica Española, 2012. 512 p.

Eguiluz, J. *Introducción a JavaScript* [En línea] Librosweb, 2015 [Consultado el: 2 de diciembre de 2015.] Disponible en: <https://librosweb.es/libro/javascript/>

Ferrel, O. C; Geoffrey, A; Ferrel, L. 2004. *Introducción a los Negocios del Mundo Cambiante*. S.l.: McGraw-Hill Interamericana, 2010. 633 p.

Flores A. ; Desiree A. *Desarrollo de Sistema Transaccional Para Manejo de Reportes del Proceso de Inyección de Químicos en la Superintendencia de tratamiento y calidad de fluidos PDVSA distrito Morichal, Edo Monagas*. Universidad de Oriente. Maturín, Venezuela, 2012.

Fundéu. *Webgrafía, neologismo adecuado*. [En línea] Fundéu BBVA, 2015. [Consultado el: 24 de mayo de 2016.] Disponible en: <http://www.fundeu.es/recomendacion/webgrafia-bibliografia/>.

GNU. *¿Qué es un Framework?*. [En línea] GNUstep en español, 2015. [Consultado el: 24 de mayo de 2016.] Disponible en: <https://gnustep.wordpress.com/gnustep-a-fondo/%C2%BFque-es-un-framework-%C2%BFcomo-se-utiliza/>.

González, R. A. *Monografía sobre motores de búsqueda.* [En línea] Geocities.ws, 2002 [Consultado el: 10 de noviembre de 2015.] Disponible en: <http://www.geocities.ws/motoresdebusqueda/inicio.html>

Google Developers. *Getting Started With Charts* [En línea] Google Developers, 2015 [Consultado el: 10 de noviembre de 2015.] Disponible en: https://developers.google.com/chart/image/docs/making_charts#google-chart-usage-policy

Hadad, G; Doorn, J; Ridao, M; Kaplan, G. *Facilitando la Asignación de Prioridades a los Requisitos.*[En línea] Facultad de Ciencias Exactas, UNICEN, Argentina, 2008 [Consultado el: 24 de mayo de 2016.] Disponible en: http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER09/ridao.pdf

Henst, C. *¿Qué es php?.* [En línea] Maestros del Web, 2001. [Consultado el: 27 de noviembre de 2015.] Disponible en: [http://www.maestrosdelweb.com/phpintro/.](http://www.maestrosdelweb.com/phpintro/)

Herrera, D. F; Sánchez, F. E. *Los indicadores webmétricos y su influencia en la calidad de los sitios web.* [En línea] Universidad de las Ciencias Informáticas, 2010. [Consultado el: 15 de noviembre de 2012.] Disponible en: [http://catalogoenlinea.uci.cu/cgi-bin/koha/opac-detail.pl?biblionumber=9344.](http://catalogoenlinea.uci.cu/cgi-bin/koha/opac-detail.pl?biblionumber=9344)

iana.org. *Internet Assigned Numbers Authority. MIME Media Types - Internet Assigned Numbers Authority.* [En línea] iana.org, 2016. [Consultado el: 24 de mayo de 2016.] Disponible en: [http://www.iana.org/assignments/media-types/media-types.xhtml.](http://www.iana.org/assignments/media-types/media-types.xhtml)

Kabir, Mohammed J. *La Biblia De Servidor Apache 2.* Anaya multimedia, 2003. 864 p.

Lancaster, F. W. *"Sistemas avanzados de recuperación de información",* Arco/Libros, 2001. 224 p.

Larman, C. *UML y Patrones: una introducción al análisis y diseño orientado a objetos y al proceso unificado.* Segunda. s.l.: Prentice Hall., 2004. 520 p.

Metzner-Szigeth, A. *"El movimiento y la matriz" – Internet y transformación socio-cultural,* Revista Iberoamericana de Ciencia, Tecnología, Sociedad e Innovación (CTS+I), No. 7. 2006.

Microsoft. [En línea]. Microsoft, 2016 [Consultado el: 22 de marzo de 2016.] Disponible en: <https://msdn.microsoft.com/es-es/library/dd409390.aspx>

MongoDB. *MongoDB* [En línea] 2015 [Consultado el: 2 de diciembre de 2015.] Disponible en: <http://www.mongodb.org>.

Mozilla Developer Network. *JavaScript*. [En línea] Mozilla.org, 2015. [Consultado el: 2 de diciembre de 2015.] Disponible en: <https://developer.mozilla.org/es/docs/Web/JavaScript>.

Muñoz, D. R. *Manual de Estadística*. [En línea]. Eumed, 2004 [Consultado el: 10 de noviembre de 2015.] Disponible en: <http://www.eumed.net/cursecon/libreria/drm/index.htm>.

NetBeans. *NetBeans IDE*. [En línea]. 2015 [Consultado el: 24 de noviembre de 2015.] Disponible en: <http://netbeans.org>.

Nuñez, G. C.; Camacho, E. *Arquitecturas de software*. [En línea] 2004 [Consultado el: 24 de noviembre de 2015.] Disponible en: <http://prof.usb.ve/lmendoza/Documentos/PS 6116/Guia%20Arquitectura%20v.2.pdf>

Object Management Group. *Business Process Model and Notation*. [En línea] Object Management Group, 2012. [Consultado el: 3 de diciembre de 2015.] Disponible en: <http://www.bpmn.org/>

Oliván, J. A; Avilés, R. A. *Una aproximación al concepto de recuperación de información en el marco de la ciencia de la documentación*. [En línea] Scielo, 2006. [Consultado el: 17 de noviembre de 2015.] Disponible en: http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0187-358X2006000200002

Ojeda, A. N. *Guía Completa de CSS3*. s.l. : Autoedicion, 2012.

OpenSource. *The MIT License*. [En línea] Opensource.org, 2015. [Consultado el: 24 de mayo de 2016.] disponible en: <https://opensource.org/licenses/MIT>.

pChart. *pChart 2.0 - a PHP charting library*. [En línea] pchart.net , 2014. [Consultado el: 24 de mayo de 2016.] Disponible en: <http://www.pchart.net/>.

Pérez M.. *Sharding, ¿sí o no?* [En línea] Pensamientos ágiles, 2009. [Consultado el: 24 de mayo de 2016.] Disponible en: <http://brigomp.blogspot.com/2009/08/sharding-si-o-no.html>.

Pérez, T. A; Rodríguez, N. H. *Módulo de reportes webmétricos para el motor de búsqueda Orión*. Universidad de las Ciencias Informáticas, 2012. [Consultado el: 27 de noviembre de 2015.] Disponible en: http://repositorio_institucional.uci.cu/jspui/handle/ident/8296

Pixelar. *Ventajas del posicionamiento Web SEO*. [En línea] Pixelar, 2015. [Consultado el: 17 de noviembre de 2015.] Disponible en: <https://www.pixelar.com.mx/posicionamiento-web/ventajas-del-posicionamiento-web-seo/>.

Pressman, R. 2005. *Ingeniería de Software, un enfoque práctico*. New York : s.n., 2005

Real Academia Española. *Webs*. [En línea] FundéuBBVA, 2010. [Consultado el: 24 de mayo de 2016.] Disponible en: <http://www.fundeu.es/recomendacion/web-plural-webs-604/>.

Revelo, Y. *El reporte*. [En línea] Prezi, 2014. [Consultado el: 10 de noviembre de 2015.] Disponible en: <https://prezi.com/icfkdh92jvbf/el-reporte/>.

Rodríguez, A. M. *Indicadores cibernéticos. nuevas propuestas para medir la información en el entorno digital*. La Habana 2006, Vol. 14 [En línea] 14 de abril de 2006. [Consultado el: 10 de noviembre de 2015.] Disponible en: http://bvs.sld.cu/revistas/aci/vol14_4_06/aci03406.htm.

Rosas, J. E. *Comparativa de frameworks en PHP: CakePHP, Symfony y Zend Framework*. [En línea] Tuxpuc, 2007. [Consultado el: 10 de noviembre de 2015.] Disponible en: <http://tuxpuc.pucp.edu.pe/articulo/comparativa-de-frameworks-en-php-cakephp-symfony-y-zend-framework>

Rueda, E. R; Delgado, Y. H.. *Los spiders y su función en los motores de búsqueda*. [En línea] Universidad de las Ciencias Informáticas, 2012. [Consultado el 30 de noviembre del 2015] *Disponible en:* http://repositorio_institucional.uci.cu/jspui/handle/ident/4130..

Sánchez, C. R. *Un acercamiento a la Webmetría*. [En línea] Infotecarios, 2015. [Consultado el: 10 de noviembre de 2015.] Disponible en : <http://www.infotecarios.com/un-acercamiento-la-webmetria>.

Shahid, B. *Highcharts Essentials*. S.l.: Packt Publishing, 2014.

SPARXSYSTEMS. *Diagrama de Despliegue UML*. [En línea]. Sparx Systems, 2014. [Consultado el: 24 de febrero de 2015.] Disponible en: http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html

Sphinx. *Sphinx*. [En línea] Sphinx, 2015. [Consultado el: 10 de mayo de 2016.] Disponible en: sphinxsearch.com.

Stark, N. S. *Motores de búsqueda en internet*. Buenos Aires: Universidad Nacional de Luján, 2001.

Vega F.J. *La Informatización del departamento de RRHH de la UPR con el uso de una Aplicación Web*. [En línea] Monografías. 2016 [Consultado el 2 de enero del 2016.] Disponible en: <http://www.monografias.com/trabajos104/informatizacion-del-departamento-rrhh-upr-uso-aplicacion-web/informatizacion-del-departamento-rrhh-upr-uso-aplicacion-web.shtml#ixzz4ASizzDsu>

Vega, I. T; Domínguez, Y. A. *Estrategia de selección de Metodología de Software ágil o robusta*. Universidad de las Ciencias Informáticas, 2010. [Consultado el: 3 de diciembre de 2015.] Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_03215_10/1/TD_03215_10.pdf.

Villena, A. M. *HTML historia y características generales*. [En línea] AEEMT, 2009. [Consultado el: 30 de noviembre de 2015.] [Disponible en: http://www.aeemt.com/contenidos_socios/Informatica/Informac_Informat_Tecnolog/AMV_AGI_AEEMT_HTML_Historia.pdf.]

V&V Quality. *V&V Quality*. [En línea] 2013. [Consultado el: 24 de abril de 2016.] Disponible en: http://www.vyvquality.com/w1/index.php?option=com_content&view=article&id=91&Itemid=162.

Anexos

Anexo # 1: Preguntas realizadas en la entrevista aplicada a especialistas del centro CIDI.

Tabla 9: Preguntas realizadas a especialistas del centro CIDI

Número	Pregunta
1	¿Cuáles son las pautas de diseño actuales del buscador Orión?
2	¿Qué información recolecta Solr para el buscador Orión?
3	¿Cuáles indicadores webmétricos resultan de mayor importancia para el buscador Orión?
4	¿Qué funcionalidades de la anterior versión del módulo Generador de Reportes Webmétricos debieran ser reutilizadas?

Anexo # 2: Ruta necesaria para que el buscador Orión reconozca los distintos enlaces con el *bundle SeoBundle*.

```
seo:
  resource: "@SeoBundle/Resources/config/routing.yml"
  prefix: /seo
```

Anexo # 3: Línea del fichero security.yml donde se define el proveedor de usuario que se va a utilizar.

```
security:
  providers:
    my_mongo_provider:
      | mongodb: {class: ORION\Bundle\StoredBundle\Document\Usuario}
```

Anexo # 4: Línea del fichero security.yml se definen los controles de acceso para cada rol.

```
access_control:
  - { path: ^/ui/user_login, roles: IS_AUTHENTICATED_ANONYMOUSLY }
  - { path: ^/ui/user_register, roles: IS_AUTHENTICATED_ANONYMOUSLY }
  - { path: ^/ui/*, roles: ROLE_USUARIO }
  - { path: ^/seo/filtrar, roles: ROLE_USUARIO }
  - { path: ^/seo/reporte, roles: ROLE_ADMIN }
```