Universidad de las Ciencias Informáticas
Facultad 1

Título: "Sistema de identificación de tendencias de la información web indexada por el buscador Orión."

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

Autor: Jorge Daniel Oña Rodríguez

Tutores: Ing. Yurelkys de los Ángeles Carreras Riopedre

Ing. Arleni Lázara Hernández Moya

La Habana, junio 2017

"Año 59 de la Revolución"

"Estoy convencido que la mitad de lo que separa a los emprendedores exitosos de los que han fracasado, es la perseverancia."

Steve Jobs

Declaración de autoría

Declaro por este medio que yo Jorge Daniel Oña Rodríguez, con carné de identidad 93010915840 soy el autor principal del trabajo titulado “Sistema de identificación de tendencias de la información web indexada por el buscador Orión” y autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivos.

Para que así conste firmo la presente a los ____ días del mes de junio del año 2017.

Jorge Daniel Oña Rodríguez

_____

Firma de Autor

Ing. Yurelkys de los Ángeles Carreras Riopedre

_____

Firma de Tutor

Ing. Arleni Lázara Hernández Moya

_____

Firma de Tutor

Agradecimientos:

A mi tutora Arleni A mi mama!

A mis amigos!

Dedicatoria:

A mi madre por su apoyo incondicional!

Resumen

Internet es hoy uno de los principales medios de información y cuenta con un gran volumen de contenido, a través de ello se generan las tendencias en la web. En los últimos tiempos se han desarrollado sistemas que permiten agilizar el proceso de búsqueda de tendencias en la red. Con el objetivo de dar solución a la búsqueda de tendencias en la intranet nacional, se realiza un estudio y análisis de los Sistemas de Tendencias más usados a nivel mundial para posteriormente definir los factores a tener en cuenta para la solución propuesta. Con el propósito de contribuir a la toma de decisiones, se desarrolla el Sistema de identificación de tendencias de la información web indexada por el buscador Orión que pretende conocer las tendencias de la intranet nacional. Para la implementación de la propuesta de solución, guiada por la metodología AUP UCI, se seleccionaron como principales tecnologías: Nutch como mecanismo para rastrear la Web, Solr como mecanismo de indexación, Symfony como marco de trabajo PHP y Visual Paradigm como herramienta para el modelado. La herramienta implementada posee un conjunto de características y funcionalidades que contribuyen a la búsqueda de tendencias, muestra las tendencias generadas a través de las palabras clave de los documentos indexados por el buscador, además permite identificar a través de filtros que permiten lograr resultados más exactos, estos filtros están relacionados con la fecha y una categoría determinada por el usuario, disminuyendo el tiempo en hallar la información que precisan.

Palabras Clave: búsqueda, herramienta, identificación, información, tendencias.

Índice

Índice de Figuras

Índice de Figuras

Introducción

Son numerosos los cambios que afronta en el mundo digital actual el sector de las Tecnologías de la Información y las Comunicaciones (TIC), desde los avances de las tecnologías hasta la presencia de nuevos agentes, el desplazamiento de los ingresos y el nacimiento de diferentes modelos de actividad económica. Los usuarios, tanto particulares como empresas, tienen ante sí un abanico cada vez mayor de servicios y aplicaciones que dan respuesta a sus necesidades de información, comunicación y ocio.

El fabuloso aumento previsto en términos de tráfico de datos, ocasionado por los cambios en el comportamiento de consumidores y empresas, sigue obligando a los operadores tradicionales de telecomunicaciones a revisar, adaptar y diversificar sus prácticas comerciales. Este mundo digital crea un intercambio de información a través de los usuarios, donde estos publican información de disimiles temas, creando así una ola de información. A esta información se puede acceder a través de los buscadores web. Estos buscadores, o motores de búsqueda, no son más que aplicaciones informáticas que rastrean la web catalogando, clasificando y organizando la información, para después ofrecérsela a los navegantes. Podrían definirse como grandes bases de datos indexadas de páginas web.

Para realizar la búsqueda hay que contactar con la página web de una de estas empresas. Los buscadores funcionan mediante programas que buscan en bases de datos que se mantienen automáticamente por los denominados *robots*[1]. En este tipo de búsqueda basta con introducir el término sobre el que se desea encontrar información. Además, a través de índices, que catalogan la información por temas. Estos índices suelen estar organizados desde los temas más generales a los más específicos, existe una cierta jerarquía en su organización y el usuario es guiado en todo momento en su búsqueda (Fritz, 2011).

Los buscadores más usados en Internet ayudan a configurar las campañas de marketing al ofrecer diversas tecnologías que son indispensables para el manejo de las palabras clave. En concreto, los sistemas de identificación de tendencias son una plataforma que recopilan datos y compara el nivel de búsqueda de las palabras clave. Se trata de un servicio que indica la tendencia de búsquedas para las palabras clave más publicadas en la web. Se podrá saber cuáles son los temas con mayor tendencia durante un período de tiempo determinado, lo que permitirá identificar los cambios, anticiparse al mercado y a tus competidores y apoyar a la toma de decisiones (Semymas, 2016).

[1] aplicación que se conecta a Internet periódicamente y recorre la Web en busca de información pública (Kumar, 2010).

Estos procesos de identificación de tendencias usan tecnologías web, esta tecnología web permitió la creación de contenido y la colaboración de los usuarios, como son los *wikis* y los *blogs* (los mismos le darían un impulso inicial a lo que se nombra ahora como la Web 2.0). La función de la *World Wide Web* (WWW) en el mundo se vio fuertemente potenciada, dejó de ser un repositorio de información y se transformó en un canal interactivo tanto con usuarios como proveedores de información, así como todas las entidades que lo componen (Adell, 2011).

Este cambio recurrente permitió que los usuarios pudiesen contribuir activamente en la creación de contenido, y por consiguiente, de la cantidad de información y conocimiento disponible en la web. La web fue impulsada aún más por el crecimiento de las llamadas redes sociales. Esto permitió que los usuarios compartieran tanto opiniones como información, complementando así los contenidos existentes en los sitios web con la expresión escrita de las opiniones de los usuarios, que dan a conocer sus sentimientos y percepciones respecto a variedades de temas.

El uso de los sistemas de identificación de tendencias se hace más efectivo cuando se identifica un dominio de búsqueda para realizar ese análisis, existen muchos buscadores que utilizan la identificación de tendencias para los contenidos que indexan como son Google, Yahoo, Bing, Ask, Altavista entre otros.

En el departamento de Soluciones Informáticas para Internet de la Facultad 1 de la Universidad de las Ciencias Informáticas se desarrolló el buscador Orión, que se encuentra desplegado en la red cubana e indexa todos los portales web de la red nacional, en los que se publica diariamente variada información a la que se da seguimiento a través de la lectura. La identificación y el análisis de tendencias de la información en la web es una poderosa herramienta usada hoy para apoyar la toma de decisiones. Sin embargo, Orión no posee un sistema de identificación de tendencias que permita identificar los temas sobre los que más se publica en la red nacional, para posteriormente realizar un análisis que apoye la toma de decisiones.

Teniendo en cuenta la situación problemática descrita anteriormente se enuncia el siguiente **problema de investigación:** ¿Cómo contribuir a la toma de decisiones con la información web indexada por el buscador Orión?

El problema planteado anteriormente tiene como **objeto de estudio** el proceso de identificación de tendencias y el **campo de acción** estará enmarcado en el proceso de identificación de tendencias de la información en la web.

Para darle solución al problema descrito anteriormente, se plantea el siguiente **objetivo general:** Desarrollar una herramienta para la identificación de tendencias de la información web indexada por el buscador Orión para facilitar la toma de decisiones.

Para guiar la investigación se plantean las siguientes **preguntas científicas**:

1. ¿Cuáles son los presupuestos teóricos que fundamentan el proceso de identificación de tendencias de la información web indexada por el buscador Orión?
2. ¿Cuáles son las herramientas y tecnologías más adecuadas para desarrollar el Sistema de identificación de tendencias de la información web indexada por el buscador Orión?
3. ¿Qué técnicas y pruebas aplicar durante el proceso de identificación de tendencias de la información web indexada por el buscador Orión?

Posibles resultados:

Una herramienta para la identificación de tendencias de la información web indexada por el buscador Orión para facilitar la toma de decisiones.

A continuación, se definen las **tareas de investigación** para dar cumplimento a las preguntas científicas planteadas anteriormente:

1. Definición de conceptos asociados a la identificación de tendencias en la web.
2. Realización de estudios homólogos de sistemas de identificación de tendencias de información en la web, en el ámbito nacional e internacional.
3. Investigación de los principales algoritmos y métodos usados en la identificación de tendencias en la web.
4. Selección de los métodos y algoritmos empleados en la identificación de tendencias de la información.
5. Implementación de la propuesta de solución.
6. Validación de la propuesta de solución.
7. Documentación de las pruebas realizadas.

En el desarrollo de la investigación se utilizaron los siguientes métodos de investigación:

Métodos teóricos

Histórico – Lógico: con el objetivo de constatar teóricamente cómo ha evolucionado en el tiempo el proceso de identificación de tendencias, así como los sistemas de identificación de tendencias en la web, y de igual forma las herramientas y tecnologías utilizadas en el desarrollo de aplicaciones web.

Analítico – Sintético: empleado para el análisis de los elementos esenciales referentes a las teorías, documentos y literatura en general relacionada con los sistemas de identificación de tendencias en la web.

Métodos empíricos

Modelación: utilizado en la representación, mediante el uso de diagramas, de las características del Sistema de identificación de tendencias de la información web indexada por el buscador Orión.

Entrevista: como técnica de recopilación de información, posibilitando entender el funcionamiento del buscador cubano Orión, para así integrar el Sistema de identificación de tendencias. Las entrevistas se realizaron a Ing. Eric Bárbaro Utrera Sust (Líder de proyecto del buscador cubano Orión) y MSc. Hubert Viltres Sala.

El presente trabajo se encuentra estructurado en tres capítulos.

En el **Capítulo 1: Fundamentación teórica de Sistemas de identificación de tendencias en la web** se realiza una investigación para esclarecer los conceptos más importantes. Se realiza un estudio de algunos de los sistemas que realizan identificación de tendencias de la información web. Además, se especifica sobre las herramientas, tecnologías y lenguajes de programación que serán utilizadas en el desarrollo de la propuesta de solución y se plantea la metodología de desarrollo por la cual estará transitando el Sistema de identificación de tendencias de la información web indexada por el buscador Orión.

En el **Capítulo 2: Análisis y diseño del Sistema de identificación de tendencias de la información web indexada por el buscador Orión** se describe la propuesta de solución a partir de los artefactos generados. Se expondrán los requisitos funcionales y no funcionales que garantizarán el desarrollo de la solución al problema.

En el **Capítulo 3: Implementación y pruebas del Sistema de identificación de tendencias de la información web indexada por el buscador Orión** se revisa las tareas realizadas y se mostrará el estilo de implementación utilizado, además se aplican las pruebas necesarias para verificar el correcto funcionamiento del software y para junto con el cliente probar y aceptar el software.

Capítulo 1: Fundamentación teórica del Sistema de identificación de tendencias de la información web indexada por el buscador Orión.

En este capítulo se presenta la definición de algunos conceptos asociados con el proceso de identificación de tendencias, con el objetivo de permitir una mayor comprensión de la investigación. Se estudian sistemas que realizan la identificación de tendencias a nivel nacional e internacional. Además, se exponen las características esenciales de la metodología de desarrollo de software, las herramientas y tecnologías que se utilizarán en la implementación de la solución.

Información: La información está constituida por un grupo de datos ya supervisados y ordenados, que sirven para construir un mensaje basado en un cierto fenómeno o ente. La información permite resolver problemas y tomar decisiones, ya que su aprovechamiento racional es la base del conocimiento (Pérez Porto & Gardey, 2012).

Tendencias: Propensión o inclinación en los hombres y en las cosas hacia determinados fines. Fuerza por la cual un cuerpo se inclina hacia otro o hacia alguna cosa. Idea religiosa, económica, política, artística, entre otros, que se orienta en determinada dirección.

Web: Web o la web, la red o www de *World Wide Web*, es básicamente un medio de comunicación de texto, gráficos y otros objetos multimedia a través de Internet, es decir, la web es un sistema de hipertexto que utiliza Internet como su mecanismo de transporte o desde otro punto de vista, una forma gráfica de explorar Internet.

1.1 Sistemas Homólogos

Los sistemas de identificación de tendencias son útiles en la web ya que ayudan a identificar las palabras que usan los usuarios para acceder a los contenidos y de este modo poder utilizarlas para optimizar el sitio web (Codina, 2005). Así mismo, desde el punto de vista del *marketing* en línea resulta muy valioso conocer las palabras utilizadas para realizar búsquedas, puesto que son útiles para presentar los anuncios más adecuados en cada contexto y además resulta una forma de segmentar el público a quién dirigirse.

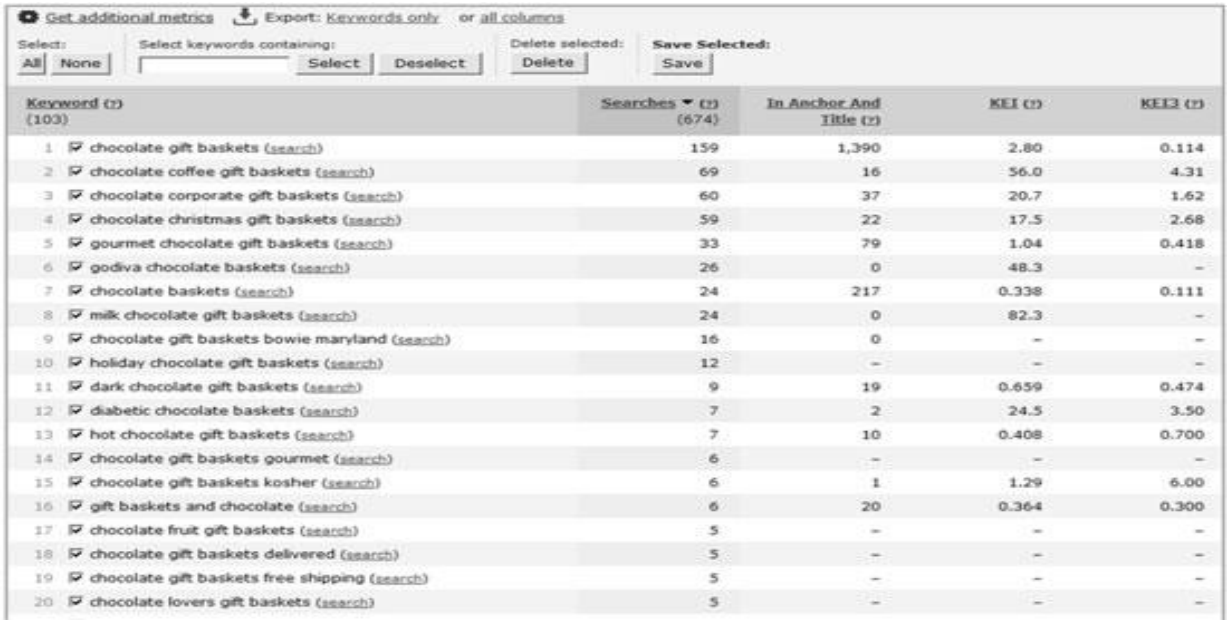
1.1.1 Análisis de herramientas que identifican tendencias

A continuación, se realiza un estudio de algunos de los sistemas de identificación de tendencias, con el objetivo de determinar elementos comunes que puedan ser utilizados en la propuesta de solución. Fueron seleccionadas aquellas herramientas que ponen especial énfasis en la selección de las palabras clave o etiquetas.

Wordtracker - Keywords

Está dirigida a los propietarios de sitios webs y a especialistas en marketing en buscadores para facilitar la identificación de las palabras clave y frases que son relevantes para el negocio y que tienen una alta probabilidad de ser utilizadas en los buscadores. Actualmente cuenta con más de 330 millones de términos de búsqueda obtenidos a partir de los metabuscadores Dogpile y Metacrawler.

- Acceso: en línea. Cuenta con versión demo, hay que hacer una suscripción de pago con la que tienes un período de prueba de siete días.
- Funcionalidades: a partir de una palabra clave localiza otras palabras clave relacionadas que considera adecuadas después de aplicar sus propios cálculos. También muestra el número de búsquedas que han generado las palabras clave en los buscadores además de otras métricas propias (figura 1). Sugiere mil palabras clave para cada término introducido (Media, 2010).



Get additional metrics | Export: Keywords only or all columns

Select: All None | Select keywords containing: Select Deselect | Delete selected: Delete | Save Selected: Save

	Keyword (?) (103)	Searches (?) (674)	In Anchor And Title (?)	KEI (?)	KEI3 (?)
1	chocolate gift baskets (search)	159	1,390	2.80	0.114
2	chocolate coffee gift baskets (search)	69	16	56.0	4.31
3	chocolate corporate gift baskets (search)	60	37	20.7	1.62
4	chocolate christmas gift baskets (search)	59	22	17.5	2.68
5	gourmet chocolate gift baskets (search)	33	79	1.04	0.418
6	godiva chocolate baskets (search)	26	0	48.3	–
7	chocolate baskets (search)	24	217	0.338	0.111
8	milk chocolate gift baskets (search)	24	0	82.3	–
9	chocolate gift baskets bowie maryland (search)	16	0	–	–
10	holiday chocolate gift baskets (search)	12	–	–	–
11	dark chocolate gift baskets (search)	9	19	0.659	0.474
12	diabetic chocolate baskets (search)	7	2	24.5	3.50
13	hot chocolate gift baskets (search)	7	10	0.408	0.700
14	chocolate gift baskets gourmet (search)	6	–	–	–
15	chocolate gift baskets kosher (search)	6	1	1.29	6.00
16	gift baskets and chocolate (search)	6	20	0.364	0.300
17	chocolate fruit gift baskets (search)	5	–	–	–
18	chocolate gift baskets delivered (search)	5	–	–	–
19	chocolate gift baskets free shipping (search)	5	–	–	–
20	chocolate lovers gift baskets (search)	5	–	–	–

Figura 1: Wordtracker-Keywords (Media, 2010).

SemRush

Es un software que ayuda a seleccionar palabras clave para un sitio web, así como términos relacionados. Su base de datos está formada por más de 80 millones de palabras clave de alrededor de 39 millones de dominios. Las actualizaciones son mensuales y la información es transparente. La herramienta se ofrece en diferentes lenguas, entre ellas el castellano.

- Acceso: en línea. Cuenta con una versión gratuita que permite utilizar todas las funcionalidades con el límite del número de consultas diarias y de ofrecer unos resultados limitados.
- Funcionalidades: a partir de una *url* identifica para qué palabras esta web es recuperada entre las primeras veinte posiciones en *Google*. Además, para cada palabra clave indica en qué posición es recuperada la página web, la media estimada del número de búsquedas que esta palabra clave genera cada mes, el número global de resultados recuperados con *Google*, y una gráfica con la tendencia estimada de las variaciones en el número de búsquedas mensuales generadas durante los últimos doce meses (figura 2) (Media, 2010).

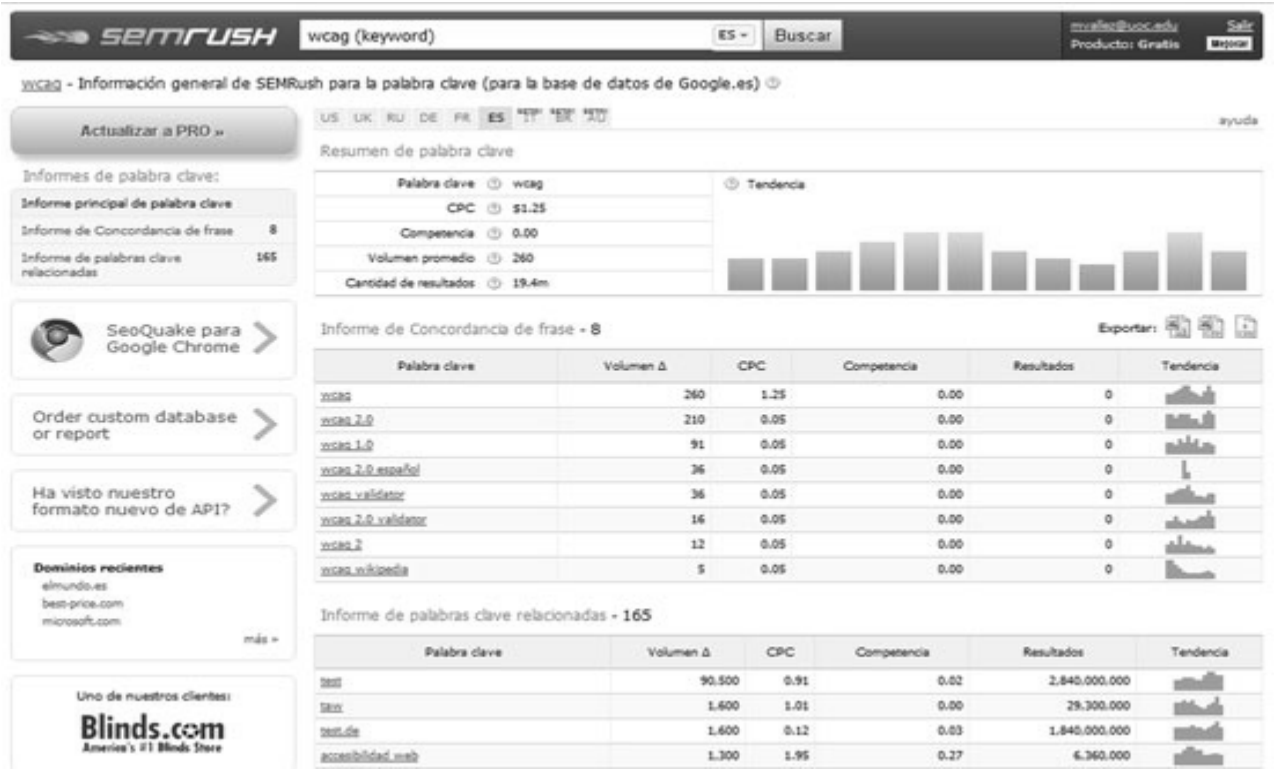


Figura 2:Semrush (Media, 2010).

Keyword Discovery

Es una de las herramientas más conocidas y con más tradición dentro de las *KRTs.* La información procede de más de 200 motores de búsqueda de todo el mundo, entre los que se encuentran *Google*, *Yahoo*, *Altavista*, etc. y sus diferentes versiones regionales. Procesa cerca de 38 mil millones de búsquedas de diferentes idiomas. Ofrece servicios complementarios que permiten gestionar toda la información procesada de forma fácil dándole un gran valor añadido a la herramienta.

- Forma de acceso: en línea. Acceso a la versión de prueba, hay dos versiones más: la estándar y la profesional.

Funcionalidades: tiene una exhaustiva base de datos que permite establecer diferentes opciones a la hora de buscar palabras clave a partir de una inicial. En la información de las palabras mostradas como resultado indica el número de veces que ha sido buscada cada palabra, así como el porcentaje de personas que visitan las páginas que aparecen en los resultados devueltos por el buscador. Además, facilita información concreta: el número estimado de resultados que están indexados en el buscador por el

término de búsqueda y el cálculo del indicador *KEI*. La última columna muestra una estimación del número de búsquedas que este término puede generar diariamente a partir de los datos que *Keyword Discovery* tiene almacenadas (figura 3) (Media, 2010).

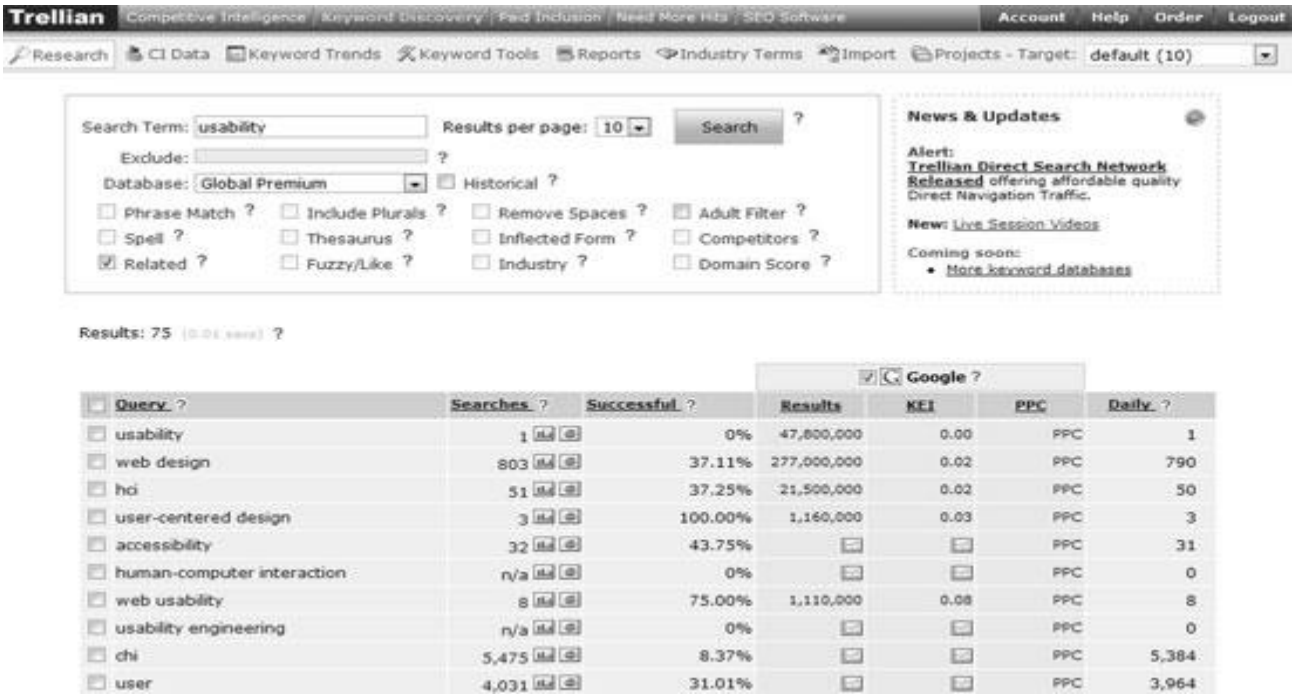


Figura 3: Keyword discovery (Media, 2010).

Google Trends

La herramienta diseñada por *Google* para analizar tendencias en Internet (ver Figura 4) una de las más útiles, pero también de las más difíciles de saber utilizar eficazmente. *Google Trends* se nutre básicamente de dos fuentes de información para devolver los resultados de tendencias: los contenidos presentes en la red y las búsquedas realizadas por los usuarios. Esta última fuente de información utilizada por la herramienta la hace indispensable a la hora de mejorar el posicionamiento en buscadores de una web en concreto (Media, 2010).

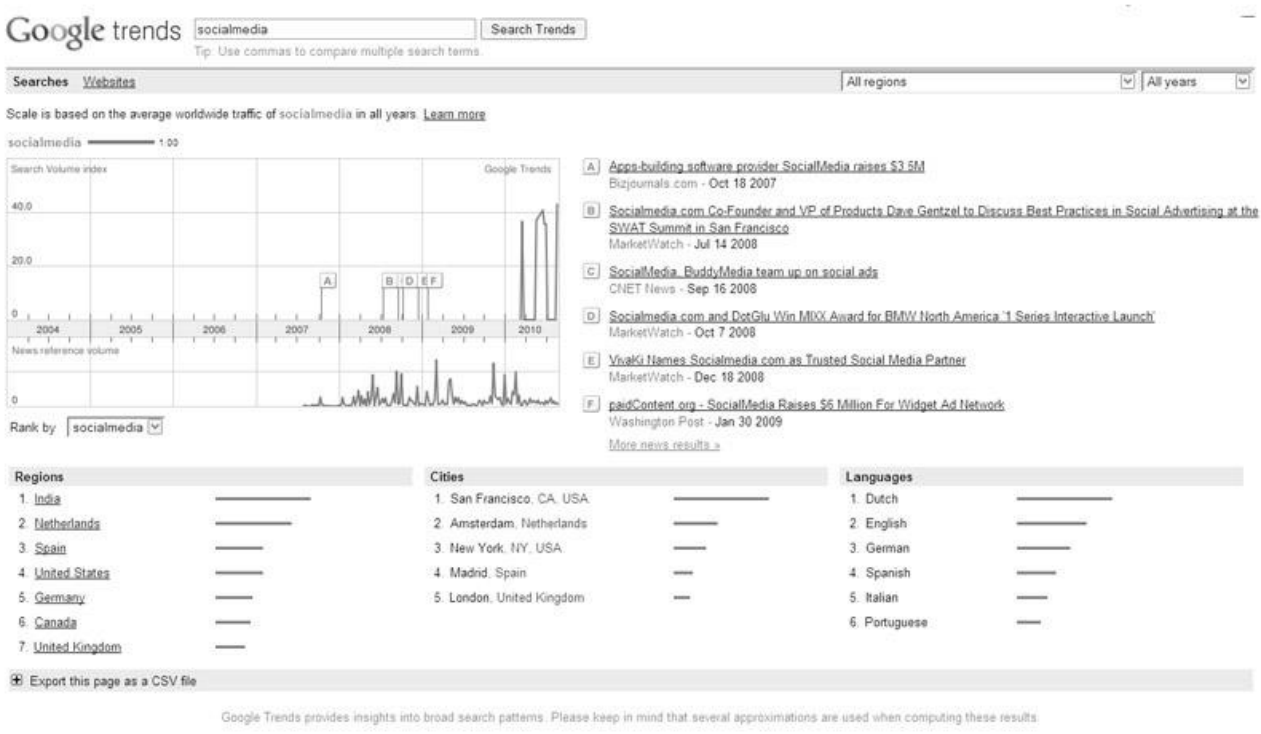


Figura 4: Google Trends (Media, 2010).

Twitter Search

Es la herramienta básica para la búsqueda de tendencias en *Twitter* (ver Figura 5). Al acceder a la página inicial se introduce la o las palabras que se quieren usar para la búsqueda, o ver los temas de los que más se está hablando actualmente en *Twitter*. Al seleccionar cualquiera de los temas podremos ver las conversaciones relacionadas con los mismos, a tiempo real (Media, 2010).

Figura 5: Tiwtter Search (Media, 2010).

Estas son solo algunas de las herramientas que existen actualmente para explorar las tendencias en la web a través de palabras claves. Las herramientas estudiadas anteriormente son privativas, por lo que se tomarán algunas de las funcionalidades de las mismas para implementar la propuesta de solución. El buscador cubano Orión, no brinda la posibilidad de identificar tendencias a partir de una consulta realizada por el usuario o de simplemente mostrarlas. Por tales motivos, se evidencia la necesidad de contar con el

Sistema de identificación de tendencias de la información web indexada por el buscador Orión, que brinde al usuario una serie de criterios de búsqueda que ayuden a obtener resultados más exactos.

1.2 Herramientas, lenguajes y tecnologías

Para desarrollar la propuesta de solución, necesidad surgida del estudio realizado de herramientas homólogas, se hace necesario estudiar varias tecnologías, lenguajes y herramientas disponibles para llevar a cabo el objetivo general de la investigación.

Mecanismos para la recolección de información

La arquitectura de un buscador está compuesta por varios subsistemas que hacen que el mismo cumpla su función. Uno de esos subsistemas es el mecanismo de rastreo o araña web, el cual se encarga de rastrear la Web en busca de documentos. Un *spider* o araña, puede definirse como una aplicación que se conecta a Internet periódicamente y recorre la Web en busca de información pública (Kumar, 2010). Los ejemplos más conocidos dentro de los recolectores web atendiendo a su uso masivo, efectividad y licencia son Wire, MnoGoSearch y Nutch (Camargo y otros, 2013). A continuación, se describen una serie de características de cada uno.

Wire

Wire[2] es un proyecto iniciado por el Centro de Investigación Web de Chile, para crear una aplicación que permita la recuperación de información; diseñada para ser utilizada en la Web (Department of Computer Sciences University of Chile, 2011).

Actualmente el software WIRE incluye:

- Un formato simple para almacenar una colección de documentos web.
- Un rastreador web.
- Herramientas para la extracción de las estadísticas de la colección.

[2] Sitio oficial: (http://www.cwr.cl/projects/WIRE/)

- Herramientas para la generación de informes acerca de la colección.

Las principales características de WIRE son las siguientes:

- Escalabilidad: diseñado para trabajar con grandes volúmenes de documentos, ha sido probado con varios millones de documentos.
- Prestaciones: programado en C/C++ para un alto rendimiento.
- Configurable: todos los parámetros para el rastreo y la indexación se pueden configurar a través de un archivo XML.
- Análisis: incluye varias herramientas para analizar, extraer estadísticas y la generación de informes sobre sub-conjuntos de la Web, por ejemplo: la web de un país o de una gran intranet.
- Licencia: el código está libremente disponible (Department of Computer Sciences University of Chile, 2011).

MnoGosearch

MnoGoSearch [3]es un motor de búsqueda de código abierto y basado en SQL. MnoGoSearch consiste en dos partes. La primera parte es un mecanismo de indización (indexer) el cual se mueve a través de vínculos de hipertexto HTML y almacena información acerca de los documentos en la base de datos. La segunda parte es una interfaz web CGI[4] la cual muestra en el navegador un formulario HTML y los resultados de búsquedas utilizan información recopilada por el indizador. Trabaja bajo licencia *GNU Public licence* GPL (Barkov, 2014).

Entre sus principales características destacan:

- Soporte para diversos protocolos: HTTP, HTTPS, FTP, NNTP.
- Soporte para autenticación de proxy.

[3] Sitio oficial: (http://www.mnogosearch.org/)
[4] CGI: Acrónimo de Interfaz de entrada común (Common Gateway Interface en inglés).

- Interfaces web CGI, Perl y PHP.
- Lenguaje de consulta *booleano*.
- Soporte para la mayoría de los conjuntos de caracteres modernos.
- Soporte para múltiples bases de datos: MySQL, PostgreSQL, SQLite, Mimer, Virtuoso, Interbase, Oracle, MS SQL, DB2, Sysbase.
- Posee una API externa para PHP.
- Manejo de clústeres de base de datos.

Nutch

Nutch[5] es un *spider* libre y de código abierto, desarrollado totalmente en Java por *The Apache Software Foundation*. Inicialmente fue implementado sobre la base de Apache Lucene, aunque ya la versión actual es independiente de Lucene, una librería de alto rendimiento para la búsqueda basada en texto y que utiliza una modificación del algoritmo "Vector Space Model" (Modelo de Espacio Vectorial en español), con un enfoque booleano que restringe las estimaciones de los resultados obtenidos (Nieto, 2009). Mantiene internamente un *ranking* a partir del cual se define el orden de rastreo de las *url*. Nutch, como mecanismo de rastreo, posee ciertos componentes llamados *parsers*, los cuales se encargan de descomponer las páginas web y analizar cada uno de los recursos que la componen o tienen relación. Uno de estos *parsers* se denomina Tika[6], el cual puede descomponer una gran cantidad de documentos, entre ellos HTML, documentos ofimáticos, pdf y muchos más. Actualmente, aunque Tika soporta una variedad de formatos sobre una gran cantidad de tipos de documentos, no es capaz de obtener toda la información que se pudiera obtener de las imágenes que encuentra, constituyendo una de sus debilidades (The Apache Software Foundation-Tika, 2014).

[5] Sitio web oficial http://nutch.apache.org/

[6] Conjunto de herramientas que detectan y extraen metadatos y textos contenido en documentos usando librerías de parseo (The Apache Software Foundation-Tika, 2014).

Tabla 1 Comparación entre servidores de rastreo.

Características	MnoGoSearch	Wire	Nutch
Multihilo	Sí	Sí	Sí
Documentos que recopila	HTML, TXT, PDF, XML, PPT, DOC, RTF, JPG, GIF, PNG, ODT	HTML, TXT, PDF, XML, PPT, DOC, RTF	HTML, TXT, PDF, XML, PPT, DOC, RTF, JPG, GIF, PNG, ODT
Configuración	Ficheros XML	Ficheros XML	Ficheros XML
Lenguajes de programación	C++	C/C++	Java
Extensible	No	Sí	Sí
Servidor de Índices	MnoGoSearch	Switch-e	Solr
Plataforma	Windows, Unix, GNU/Linux	Windows, Unix, Mac OS, GNU/Linux	Windows, Unix, Mac OS, GNU/Linux
Licencia	GNU Public licence GPL	Apache Software Foundation(ASF)	GNU Public licence GPL

Atendiendo a que MnoGoSearch utiliza su propio mecanismo de indexación y que Wire está más enfocado a realizar análisis webmétricos, el autor determina que el *spider* a utilizar será Nutch por ser de propósito general. Nutch facilita la incorporación de nuevos *plugins*[7]que permiten el proceso de identificación de tendencias y además porque es el mecanismo de rastreo utilizado por el buscador cubano Orión, lo que facilitará la integración entre ambos sistemas.

Mecanismos para la indexación de información

Otro de los subsistemas presentes en la arquitectura de un buscador es el mecanismo de indexación.

Entre los más utilizados se encuentran:

[7] Un plugin es complemento que necesita de una aplicación principal para funcionar el cual añade una funcionalidad adicional o una nueva característica al software sin requerir cambios en el software original.

Sphinx

Motor de búsqueda de texto completo, distribuido públicamente bajo licencia GPL, su nombre es un acrónimo que se descodifica oficialmente como SQL Phrase Index. Este ofrece la funcionalidad de búsqueda rápida y relevante de texto completo para aplicaciones clientes. Fue especialmente diseñado para integrarse con bases de datos SQL, y ser de fácil acceso para los lenguajes script. Sin embargo, Sphinx no depende ni requiere ninguna base de datos específica para funcionar (Nugraha, 2014).

Entre sus principales características se encuentran:

- Posee una alta velocidad de indexación (hasta 10-15 MB / seg por núcleo).
- Posee una alta velocidad de búsqueda (hasta 150-250 consultas / seg por núcleo contra 1.000.000 documentos, 1.2 GB de datos).
- Conjunto de resultados avanzados de post-procesamiento (SELECT con expresiones, WHERE, ORDER BY, GROUP BY, HAVING entre otros, sobre los resultados de búsqueda de texto).
- Comprobada escalabilidad hasta miles de millones de documentos, terabytes de datos y miles de consultas por segundo (AKSYNOFF, 2014).

Sphinx es adecuado para aquellos que ya tienen contenido digital existente dentro de los servidores de bases de datos tales como MariaDB, PostgreSQL, MS SQL y desea que el contenido sea indexado para una rápida recuperación sin tener que convertir a otro formato primero.

Elasticsearch

Elasticsearch es un servidor de búsqueda basado en Lucene, desarrollado en Java y ha sido liberado como código abierto bajo los términos de la Licencia Apache. Proporciona un motor de búsqueda de texto completo y con capacidad multiusuario, expone su funcionalidad a través de una interfaz REST recibiendo y enviando datos en formato JSON y oculta mediante esta interfaz los detalles internos de Lucene. Esta interfaz permite que pueda ser utilizado por cualquier plataforma, o sea, no solo desde Java, además puede usarse desde Python, .NET, PHP o incluso desde un navegador con JavaScript. Es persistente y

de índice incremental, es decir, que los nuevos documentos a indexar son agregados a los ya existentes en la base de datos del indexador.

Entre sus características se destacan:

- Expone un API HTTP de tipo RESTful, y usa JSON tanto para peticiones como para respuestas. También se puede operar usando la API nativa de Java;
- Está libre de esquemas de datos, en el sentido de que no necesita disponer de una definición explicita del esquema;
- Búsquedas Facetadas - Muestra contador para cada categoría en los resultados de búsqueda;
- Búsqueda Geo-espacial - Búsqueda por localización y distancia. (Buscar dentro de 5 km de la posición actual);
- Los documentos (datos) no necesariamente tienen que ser planos, permite elementos anidados;
- Replicación - El índice podría ser replicado y proporciona soporte para conmutación por error;
- Posee búsqueda distribuida, es decir, la búsqueda puede realizarse en varios fragmentos/índices y al concluir la misma los resultados serán agregados;
- Presenta indexación distribuida lo que significa que los documentos van a ser almacenados en distintos nodos.
- Arquitectura diseñada pensando siempre en la distribución para permitir escalar una solución de un nodo a cientos, ofreciendo alta disponibilidad, soportando grandes cantidades de datos y cortos tiempos de respuesta.
- Actualmente Elasticsearch presenta una comunidad pequeña de colaboradores, y consecuentemente una base de usuarios pequeña (Elasticsearch, 2014).

Solr

Solr es una plataforma de búsquedas basada en Apache Lucene, que funciona como un "servidor de búsquedas". Sus principales características incluyen búsquedas de texto completo, resaltado de resultados y manejo de documentos (como Word y PDF). Solr es escalable, permitiendo realizar búsquedas distribuidas y replicación de índices (SETA, 2010). Actualmente es frecuentemente usado por importantes sitios en Internet como son: el sitio web de la Casa Blanca, utiliza Solr vía Drupal para la búsqueda con resaltado y facetado de sitios; Instagram[8], es una compañía de Facebook que utiliza Solr para potenciar su API[9] de geo-búsqueda; Jobreez[10], es potenciado con Solr 4.0 para realizar búsquedas de ofertas de trabajo a través de más de 25000 fuentes, entre otros (Solr-WIKI, 2014). Está escrito en Java, pero puede ser usado en cualquier lenguaje, simplemente usando las peticiones GET para realizar búsquedas en el índice, y POST para agregar y actualizar documentos. Fácil de configurar y usar. Una de las características principales de Solr es su API estilo REST[11] (The Apache Software Foundation-Solr, 2014).

Otras de sus características son:

- Ofrece un API REST y un API de Java;
- No se requiere un esquema y tipo de documento;
- Búsquedas Facetadas - Muestra contador para cada categoría en los resultados de búsqueda;
- Búsqueda Geo-espacial - Búsqueda por localización y distancia. (Buscar dentro de 5 km de la posición actual);
- Los documentos (datos) tienen que ser planos, debido a que no permite elementos anidados;
- Permite importar datos desde una base de datos;
- Replicación - El índice podría ser replicado y proporciona soporte para conmutación por error;

[8] Instagram: https://instagram.com.
[9] API: Acrónimo de Interfaz de Programación de Aplicaciones (Application Programming Interface en inglés).
[10] Jobreez: http://www.jobreez.com.
[11] Es un tipo de arquitectura de desarrollo Web que se apoya totalmente en el estándar HTTP.

- Búsqueda distribuida – La búsqueda puede realizarse en varios fragmentos/índices y al concluir la misma los resultados serán agregados;
- Resaltado de los resultados de búsqueda;
- Se ha desarrollado con la capacidad de extraer el contenido del archivo desde el sistema de archivos y añadirlo como parte de índice.

Solr, al igual que Elasticsearch, facilita a los programadores desarrollar aplicaciones sofisticadas y de alto rendimiento que incluyen facetado (resultado de búsqueda ordenado en columnas con cuentas numéricas). Solr se fundamenta en otra tecnología originaria de búsqueda-Lucene, una biblioteca de Java que provee la indexación y la tecnología de búsqueda. Ambos Solr y Lucene son manejados por la Apache Software Foundation (Pugh, 2009).

Como la interfaz principal de Solr es un API REST a través del protocolo HTTP, necesita un contenedor de *servlets*[12]. Este servidor de búsqueda tiene varios servlets que exponen distintos servicios, como UPDATE o SELECT (Apache Software Foundation, 2014). Además, Solr es un mecanismo de indexación que utiliza un modelo de recuperación de información híbrido ya que se basa en el modelo booleano y en el modelo vectorial para establecer el subconjunto de documentos relevantes.

En la Tabla 2 se recogen varias características de estos tres sistemas a fin de realizar una pequeña comparación entre ellos:

[12] Servidor web capaz de ejecutar servlets de java.

Tabla 2: Comparación entre servidores de indexación.

Nombre		**Elasticsearch**	**Solr**	**Sphinx**
Documentación técnica		www.elasticsearch.org/guide	lucene.apache.org/solr/d ocumentation.html	sphinxsearch.com/docs
Sistema Operativo		Todos con la MV de Java	Todos con la MV de Java y contenedor de servlets	FreeBSD, Linux, NetBSD, OS X, Solaris, Window
Popularidad	Clasificación [13]	15	12	35
	Puntuación[14]	58.92	81.88	10.3
Lenguaje de implementación		Java	Java	C++
APIs y otros métodos de acceso		API de Java, API RESTful HTTP/JSON	API de Java, Api RESTful, HTTP	Proprietary protocol
Extensibilidad		No	*Plugins* de Java	No
Triggers		Si	Si	No
Integridad en la manipulación de datos		No	Bloqueo Optimista	No
Concurrencia en la manipulación de datos		No	Si	Si
Persistencia de datos		Si	Si	Si

Fuente: (Sphinx, 2014), (Gormley y Tong, 2014), (Apache.org, 2014), (Db-engines, 2014), (Db-engines-ranking, 2014).

Sphinx fue especialmente diseñado para integrarse con los servidores de bases de datos SQL, y ser de fácil acceso para los lenguajes de script, mientras que a diferencia de este, otras soluciones de indexación

[13] Representa el lugar que está en el ranking mundial de los modelos de base de datos (http://dbengines.com/en/ranking).
[14] Representa el nivel de popularidad que está en el ranking de los modelo de base de datos. (http://dbengines.com/en/ranking_trend/system/Elasticsearch%3BSolr%3BSphinx), (http://dbengines.com/en/system/Elasticsearch%3BSolr%3BSphinx).

de contenido digital como Solr o ElasticSearch, ofrecen sus servicios en forma de una API REST y son de propósito general sin necesidad de integrarse a una base de datos SQL, descartándose así, por el autor, el uso de Sphinx como mecanismo de indexación para la herramienta de búsqueda de imágenes en la Web. Por otra parte, aunque tanto Solr como Elasticsearch gozan de una gran popularidad, ambos son escritos en Java. Elasticsearch no presenta soporte para la manipulación de datos concurrentemente, tampoco presenta soporte para garantizar la integridad de los datos luego una transacción no atómica, mientras que Solr sí (DB-ENGINES, 2015).

Por tanto, teniendo en cuenta la anterior comparación, añadiendo que Solr hoy es el mecanismo de indexación utilizado por el buscador cubano Orión y como se muestra en la Tabla 2, es el Servidor de índice utilizado por Nutch, el autor ha determinado que como mecanismo de indexación se utilizará Solr. Es importante señalar que Solr implementa una variante del modelo TFIDF (Term Frequency Inverse Document Frequency) para determinar cuánto de relevante es un documento con respecto a la consulta del usuario, lo que se conoce como fórmula de relevancia.

Lenguajes de programación

A continuación, se presentan los lenguajes de programación seleccionados del estudio realizado.

Del lado del servidor

Java: es un lenguaje de programación multiplataforma que se introdujo a finales de 1995. Al programar en Java no se parte de cero. Cualquier aplicación que se desarrolle se apoya en un gran número de clases preexistentes. Algunas de ellas las ha podido hacer el propio usuario, otras pueden ser comerciales, pero siempre hay un número muy importante de clases que forman parte del propio lenguaje (el API o Application Programming Interface de Java). Java incorpora en el propio lenguaje muchos aspectos que en cualquier otro lenguaje son extensiones propiedad de empresas de software o fabricantes de ordenadores. Java es un lenguaje muy completo. La compañía Sun Microsystems creadora del mismo describe el lenguaje Java como “simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutra, portable, de altas prestaciones, multitarea y dinámico” (Jalón, y otros, 2000).

PHP: es un lenguaje de alto nivel e interpretado, utilizado en su mayoría para el procesamiento dinámico de información en la Web. El mismo, cuyo significado se le confiere a “PHP Hypertext Preprocessor” por sus siglas en inglés, puede ser incrustado en documentos HTML, pero solo puede ejecutarse en el lado del servidor (BAKKEN, 2013). Por ser de código abierto, posee una amplia comunidad internacional que colabora en el mejoramiento del código fuente, el desarrollo y actualización de librerías para el lenguaje y la traducción de la documentación del proyecto. Corre en (casi) cualquier plataforma utilizando el mismo código fuente, pudiendo ser compilado y ejecutado en algo así como 25 plataformas, incluyendo diferentes versiones de Unix, Windows y Macs. La sintaxis de PHP es similar a la del C, por esto cualquiera con experiencia en lenguajes del estilo C podrá entender rápidamente PHP. PHP es completamente expandible.

Está compuesto de un sistema principal (escrito por Zend), un conjunto de módulos y una variedad de extensiones de código (Mariño, 2008).

Teniendo en cuenta que PHP es un lenguaje completamente expandible, que presenta una gran variedad de módulos y es libre, incluyendo el resto de las características expuestas anteriormente, se decide seleccionarlo como lenguaje del lado del servidor a utilizar, mientras que Java se utilizará para el trabajo con Nutch, el cual es el encargado de filtrar los documentos en un buscador.

Del lado del cliente

Para el trabajo en la Web se utilizará HTML, CSS y JavaScript, todos estos integrados en el marco de trabajo Bootstrap.

1.3 Marco de trabajo Bootstrap

Bootstrap, es originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de una PC, una Tablet u otro dispositivo. Esta técnica de diseño y desarrollo se conoce como "*responsive design*" o diseño adaptativo. Los diseños creados con Bootstrap son simples, limpios e intuitivos, esto le da agilidad a la hora de cargar y al adaptarse a otros dispositivos. El Framework trae varios elementos con estilos predefinidos fáciles de configurar: Botones, Menús desplegables, Formularios incluyendo todos sus elementos e integración con jQuery para ofrecer ventanas y tooltips dinámicos (Otto y Thornton, 2013). Bootstrap utiliza HTML, acrónimo de HyperText Markup Language, es un lenguaje de publicación especificado como un estándar por el W3C (World Wide Web Consortium) que permite la creación de páginas web (World Wide Web Consortium, 2014), CCS para aplicar de forma consistente diferentes estilos a los documentos creados (Lie, y Bos, 2005) y JavaScript para proporcionarle cierto dinamismo a las páginas web (Sánchez, 2003).

1.4 Marcos de trabajo para PHP

Otro de los componentes de un buscador es la interfaz web. Para su desarrollo existen herramientas que poseen una estructura personalizable e intercambiable para desarrollar aplicaciones web. Estas herramientas son llamadas marcos de trabajo o framework. Para desarrollar la interfaz web, a partir de la selección de PHP como lenguaje del lado del servidor a utilizar para desarrollar la interfaz web, se estudiaron los framework presentes en la Tabla 4, los cuales son los más usados en el desarrollo de aplicaciones web con este lenguaje (Acosta y otros, 2014).

Tabla 3 Comparación entre marcos de trabajo (Acosta y otros, 2014).

PHP Framework	Arquitectura MVC	Validación incorporada	Soporte mapeador de objetos	Soporte múltiples base de datos	Almacenamiento en caché de objeto	Comunidad de código abierto	Gran cantidad de documentación
CodeIgniter	Sí	Sí	-	Sí	Sí	Sí	Sí
Symfony	Sí	Sí	Sí	Sí	Sí	Sí	Sí
Yii	Sí	Sí	Sí	Sí	Sí	Sí	Sí
Zend	Sí	Sí	Sí	Sí	Sí	Sí	Sí

Teniendo en cuenta las características y beneficios expuestos anteriormente se decide seleccionar Symfony en su versión 2.3 LTS. Además, es sencillo de usar, lo suficientemente flexible como para adaptarse a los casos más complejos y puede integrarse al buscador cubano Orión que contiene una interfaz web desarrollada con este popular framework (Potencier, 2011).

1.5 Metodología de Desarrollo:

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.

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

- Simplicidad: Todo se describe concisamente utilizando poca documentación
- Agilidad: El ajuste a los valores y principios de La Alianza Ágil.
- Centrarse en actividades de alto valor: La atención se centra en las actividades que en realidad lo requieren, no en todo el proyecto.

- 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.)
- 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.

Herramientas

Para agilizar la realización de determinadas actividades del proceso de desarrollo de software se seleccionó como herramienta CASE:

Visual Paradigm

Visual Paradigm es una herramienta CASE multiplataforma, que soporta el ciclo completo de desarrollo de software: análisis, diseño, implementación y pruebas. Facilita la construcción de aplicaciones informáticas con un menor coste que destacan por su alta calidad y contribuye a mejorar la experiencia de usuario mediante el diseño de un gran número de artefactos de ingeniería de software. Permite la generación de bases de datos, conversión de diagramas entidad-relación a tablas de base de datos, mapeos de objetos y relaciones, ingeniería directa e inversa, la gestión de requisitos de software y la modelación de procesos del negocio (Visual Paradigm, 2014).

Entorno de desarrollo integrado (Netbeans)

Como entorno de desarrollo integrado se utilizará Netbeans. Liberado bajo el licenciamiento dual de CDDL y GPL (versión 2), NetBeans es un potente IDE para programadores que proporciona una plataforma ideal para escribir, compilar, depurar y ejecutar programas informáticos (ORACLE, 2014). Aunque inicialmente fue ideado para Java, puede ser empleado para la codificación de aplicaciones en múltiples lenguajes de programación. Este, además de ser gratuito y sin restricciones de uso, posee versiones para los distintos sistemas operativos del mercado, convirtiéndolo en una alternativa con grandes ventajas para los desarrolladores. La estructura modular de NetBeans le proporciona estabilidad y grandes posibilidades de ser extendido gradualmente por desarrollos comunitarios, permitiendo agregar continuamente nuevas

funcionalidades. Su versatilidad lo ha convertido en el IDE por excelencia entre miles de programadores alrededor del mundo (ORACLE, 2014).

Servidor web (**Apache**)

Para atender las peticiones de los usuarios mediante una interfaz web se utilizará como servidor web HTTP Apache. Apache 2 es un servidor web de código abierto que implementa el protocolo HTTP 1.1, caracterizado fundamentalmente por su alto nivel de configuración, modularidad, robustez y estabilidad. Desarrollado bajo la licencia ASF por The Apache Software Foundation, es considerado una de las mejores y más aceptadas creaciones del mundo del software libre (Netcraft, 2015). Teniendo en cuenta las estadísticas históricas y uso diario proporcionadas por NetCraft, este servidor llegó a usarse durante el 2005 en el 70% de los sitios web en el mundo, representando su cuota máxima en el mercado hasta la actualidad. Las estadísticas presentadas demuestran que Apache 2 ha sido considerado como el servidor web HTTP por excelencia desde Julio de 1998, según las estadísticas (NetCraft, 2015), logrando que millones de servidores mundiales ratifiquen su utilización.

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

Conclusiones Parciales

En este capítulo se trataron los elementos teóricos que dan sustento a la propuesta de solución del problema planteado, arribando a las siguientes conclusiones:

- El estudio de los conceptos asociados al proceso de identificación de tendencias permitió lograr un mejor entendimiento de la investigación que se realiza.

- El análisis de las funcionalidades que brindan algunos de los sistemas de identificación de tendencias permitió identificar requerimientos necesarios que serán aplicados en la propuesta de solución.
- La selección de la metodología, herramientas y tecnologías con soporte multiplataforma y basadas en software libre, posibilitó obtener una base tecnológica enfocada en los componentes que utilizan los sistemas de identificación de tendencias estudiados.

Capítulo 2: Análisis y diseño del Sistema de identificación de tendencias de la información web indexada por el buscador Orión.

Introducción

En el presente capítulo se tratarán aspectos fundamentales relacionados con el diseño del sistema a desarrollar. Entre los elementos a destacar se generaron los artefactos relacionados a la especificación de los requerimientos funcionales y no funcionales que deberá poseer el software; así como la especificación de las historias de usuarios del sistema.

2.1 Propuesta de solución

En la presente investigación se propone el desarrollo del sistema de identificación de tendencias de la información web indexada en el buscador Orión, permitiendo mostrar tendencias a través de palabras claves, filtrando por una categoría en específico o por una fecha dada y así mostrar las tendencias que se quieran analizar.

2.2 Artefactos Generados

El proceso de desarrollo del sistema guiado por la metodología AUP-UCI genera como principal artefacto las Historias de Usuario. Teniendo en cuenta que no se modela el negocio, se ajustan las funcionalidades que se describen en un documento de Especificación de Requisitos de Software, al escenario 4 que establece esta metodología.

2.3 Modelo de Dominio

Un modelo del dominio es una representación de las clases conceptuales del mundo real, no de componentes de software. No se trata de un conjunto de diagramas que describen clases de software, u objetos de software con responsabilidades, sino que modela clases conceptuales significativas en un determinado problema (Larman, 2004).

Para lograr un mejor entendimiento de la presente investigación se hace necesario describir el proceso de identificación de tendencias en la web mediante una serie de conceptos, entidades y sus relaciones, agrupándose en un modelo del dominio con el fin de contribuir a la comprensión del contexto actual del problema.

Administrador: es el que inicia el proceso de identificación de tendencias.

Orión: constituye una herramienta de recuperación de la información en la web.

Documentos: recursos publicados en la web (páginas web, imágenes, videos, documentos ofimáticos).

Tendencias Avanzadas: permite mostrar las tendencias por filtros (fecha y categoría).

Tendencias Actuales: permite mostrar las tendencias actuales a través de un top.

Tendencias Populares: permite mostrar las tendencias populares por categorías.

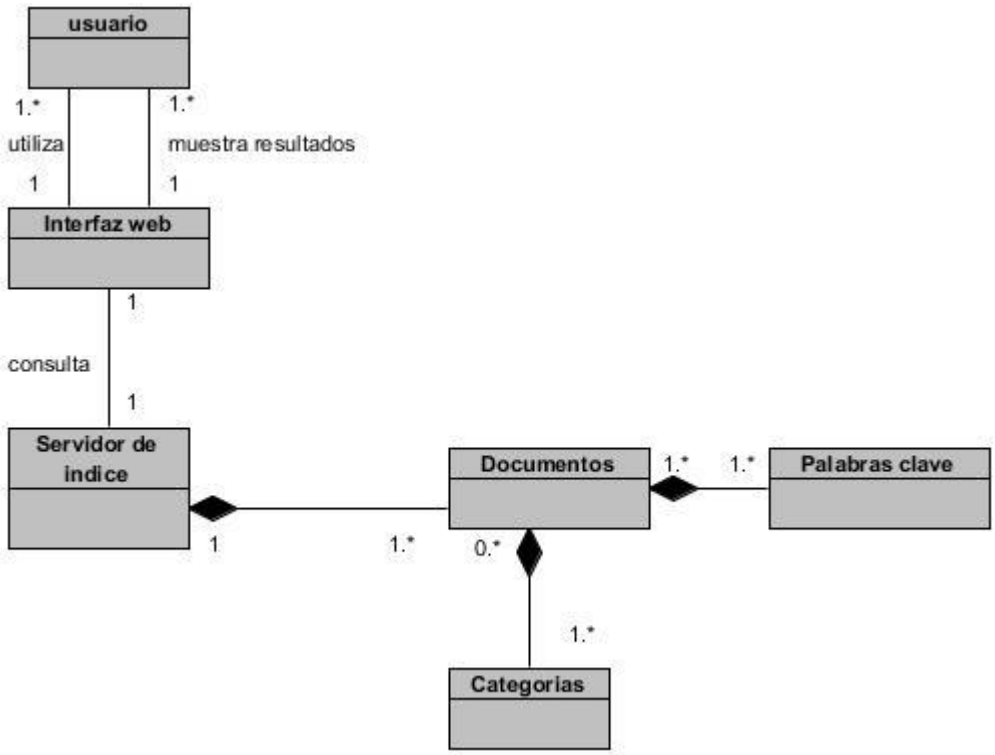


Figura 6: Diagrama de clases del modelo del dominio.

La Figura 6 muestra la relación existente entre todos los conceptos que intervienen en la investigación que se presenta.

2.2.1 Especificación de Requisitos

En Ingeniería del software, un requerimiento es una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio. Los requerimientos son declaraciones que identifican atributos, capacidades, características y/o cualidades que necesita cumplir un sistema (o un sistema de software) para que tenga valor y utilidad para el usuario. En otras palabras, los requerimientos muestran qué elementos y funciones son necesarias para un proyecto (Alegsa, 2016).

A continuación, se presenta el listado de requisitos funcionales del Sistema de identificación de tendencias de la información web indexada por el buscador Orión.

Tabla 4: Especificación de requisitos funcionales.

No.	**Nombre del Requisito Funcional**	**Descripción**
Prioridad		**Alta**
RF1	Listar tendencias actuales	Permite mostrar las tendencias actuales ordenadas según la cantidad de apariciones que tenga una determinada palabra clave en los documentos indexados por el buscador.
RF2	Listar tendencias actuales por año, mes y día.	Permite mostrar las tendencias actuales por año, mes y día.
RF3	Identificar tendencias en búsquedas de películas.	Permite mostrar tendencias en búsqueda de películas.
RF4	Identificar tendencias en búsquedas de personas.	Permite mostrar tendencias en búsqueda de personas.
RF5	Identificar tendencias en búsquedas de noticias.	Permite mostrar tendencias en búsqueda de noticias.
RF6	Identificar tendencias en búsquedas de bebidas.	Permite mostrar tendencias en búsqueda de bebidas.
RF7	Identificar tendencias en búsquedas de canciones.	Permite mostrar tendencias en búsqueda de canciones.
RF8	Identificar tendencias en búsquedas de carros.	Permite mostrar tendencias en búsqueda de carros.

RF9	Identificar tendencias en búsquedas científicas.	Permite mostrar tendencias en búsqueda científicas.
RF10	Identificar tendencias en búsquedas de hoteles.	Permite mostrar tendencias en búsqueda de hoteles.
RF11	Identificar tendencias en búsquedas de restaurantes.	Permite mostrar tendencias en búsqueda de restaurantes.
RF12	Identificar tendencias en búsquedas de páginas de ministerios.	Permite mostrar tendencias en búsqueda de páginas de ministerios.
RF13	Identificar tendencias en búsquedas de libros.	Permite mostrar tendencias en búsqueda de libros.
RF14	Identificar tendencias en búsquedas de medicamentos.	Permite mostrar tendencias en búsqueda de medicamentos.
RF15	Identificar tendencias en búsquedas de deporte.	Permite mostrar tendencias en búsqueda de deporte.
RF16	Identificar tendencias en búsquedas de política.	Permite mostrar tendencias en búsqueda de política.
RF17	Identificar tendencias en búsquedas de cultura.	Permite mostrar tendencias en búsqueda de cultura.
RF18	Identificar tendencias en búsquedas de economía.	Permite mostrar tendencias en búsqueda de economía.
RF19	Identificar tendencias en búsquedas de tecnología.	Permite mostrar tendencias en búsqueda de tecnología.

Los requisitos no funcionales describen atributos sólo del sistema o del ambiente del sistema que no están relacionados directamente con los requisitos funcionales. Incluyen restricciones cuantitativas, como el tiempo de respuesta o precisión, tipo de plataforma (lenguajes de programación y/o sistemas operativos).

A continuación, se presenta el listado de requisitos no funcionales del Sistema de identificación de tendencias de la información web indexada por el buscador Orión.

Requerimientos de software

RNF 1. Se requiere del sistema operativo Ubuntu 14.04 o superior.

RNF 2. Se requiere la instalación del servidor web Apache2 y de servlets Tomcat 7 para el correcto funcionamiento del servidor de Solr.

RNF 3. Se requiere la instalación de la Máquina Virtual de Java (JVM, por sus siglas en inglés) para el correcto funcionamiento del rastreador.

RNF 4. Se requiere la instalación de PHP 5.5 o superior para poder visualizar la interfaz web.

Requerimientos de hardware

RNF 5. Para el servidor de índice se necesita como mínimo: 2 GB RAM, CPU de 4 núcleos y 1GB de almacenamiento.

RNF 6. Para el servidor de rastreo: 2 GB RAM, CPU de 4 núcleos y al menos 80 GB de Disco Duro.

RNF 7. Para la interfaz web: 2 GB RAM, CPU de 4 núcleos y al menos 40 GB de Disco Duro.

Requerimientos de diseño e implementación

RNF 8. Como lenguaje de programación para la interfaz web se deberá utilizar PHP en su versión 5.5 o mayor.

RNF 9. Para el desarrollo de la aplicación web se deberá utilizar Symfony 2.7 LTS o superior como marco de trabajo.

Requerimientos de apariencia o interfaz de usuario

RNF 10. Todos los iconos irán acompañados de un texto descriptivo.

Requerimientos de usabilidad

RNF 11. Debe contar con la portabilidad necesaria para poder ser transferido de un ambiente a otro o reemplazado por nuevas versiones.

RNF 12. El sistema debe permitir acceder a sus distintas partes con una profundidad máxima de 3 clics.

RNF 13. Los dispositivos clientes que utilizarán la herramienta deben contar con navegadores web que soporten HTML5 y CSS3.

RNF 14. Se requiere el uso de herramientas y recursos de software libre, las cuales se podrán usar, modificar y distribuir libremente.

Requerimientos de eficiencia

RNF 15. El sistema debe ser capaz de responder como máximo en 5 segundos.

2.2.2 Historias de Usuario

Las historias de usuario son descripciones cortas y simples de una funcionalidad, escritas desde la perspectiva de la persona que necesita una nueva capacidad de un sistema, por lo general el usuario, área de negocio o cliente.

A continuación, se muestran las Historias de Usuario:

1. Generar tendencias actuales
2. Listar tendencias actuales por año, mes y día.
3. Identificar búsquedas populares por año.

Tabla 5: Historia de usuario-Generar tendencias actuales.

Número: 1	**Nombre del requisito:** Generar tendencias actuales
Programador: Jorge Daniel Oña Rodríguez	**Iteración Asignada:** 1
Prioridad: Alta	**Tiempo Estimado:**25
Riesgo en desarrollo: Alto	**Tiempo Real:** 20
Descripción: Permite mostrar las tendencias actuales en la web	
Observaciones:	
Prototipo elemental de la interfaz gráfica de usuario:	

Tabla 6: Historia de usuario-Listar tendencias actuales año, mes y día.

Número: 2	**Nombre del requisito:** Listar tendencias actuales año, mes y día.
Programador: Jorge Daniel Oña Rodríguez	**Iteración Asignada:** 1
Prioridad: Alta	**Tiempo Estimado:**15
Riesgo en desarrollo: Alto	**Tiempo Real:** 10
Descripción: Permite mostrar las tendencias actuales por año, mes y día.	
Observaciones:	

Prototipo elemental de la interfaz gráfica de usuario:

Tabla 7: Historia de usuario-Identificar búsquedas populares por año.

Número: 3	**Nombre del requisito:** Identificar búsquedas populares por año.
Programador: Jorge Daniel Oña Rodríguez	**Iteración Asignada:** 1
Prioridad: Alta	**Tiempo Estimado:**15
Riesgo en desarrollo: Alto	**Tiempo Real:** 10
Descripción: Permite mostrar las búsquedas más populares por año.	
Observaciones:	

Prototipo elemental de la interfaz gráfica de usuario:

2.3 Arquitectura del sistema

Symfony basa su funcionamiento interno en el patrón arquitectónico Modelo - Vista – Controlador (MVC), utilizado por la mayoría de frameworks web. No obstante, según su creador Fabien Potencier: "Symfony no es un framework MVC. Symfony sólo proporciona herramientas para la parte del Controlador y de la Vista. La parte del Modelo es responsabilidad del usuario (Potencier, 2011). La siguiente figura muestra la arquitectura del sistema desarrollado.

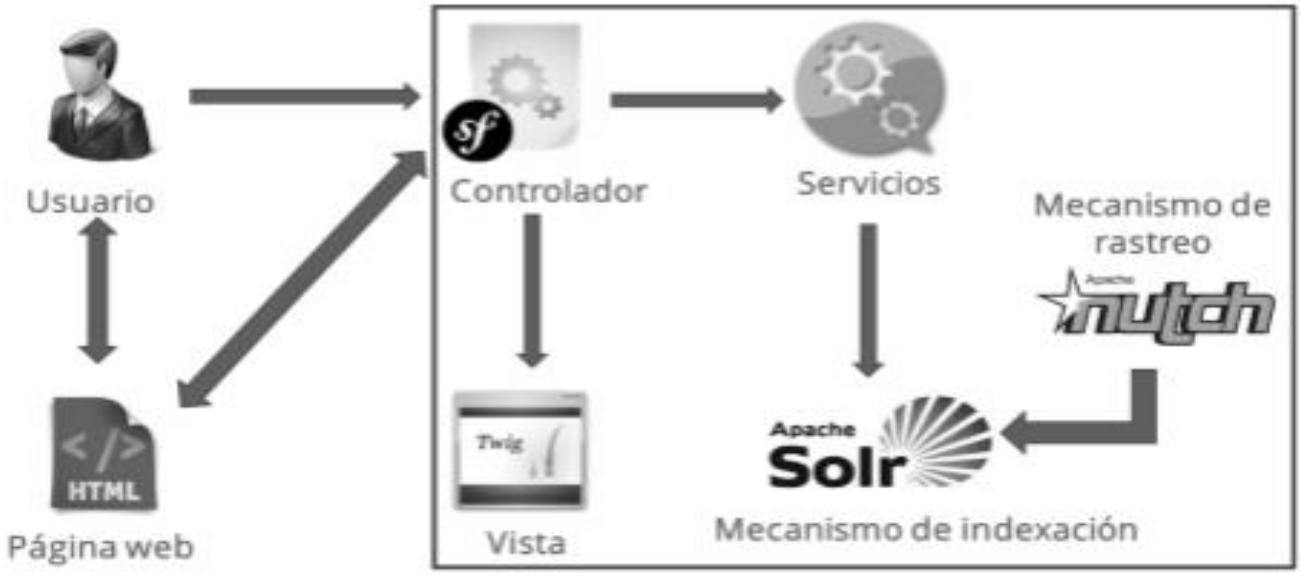


Figura 7: Arquitectura del sistema.

Como se observa en la Figura 7 a través del controlador son recibidas y atendidas todas las peticiones al sistema. Cuando el controlador recibe una petición del usuario, consulta los datos almacenados en el mecanismo de indexación a través de un servicio desarrollado en Symfony. Los datos son almacenados en el mecanismo de indexación luego del recorrido realizado por el mecanismo de rastreo. Esta arquitectura está basada en el patrón arquitectónico MVC, donde la función del modelo la tienen los servicios de Symfony los cuales son los encargados de implementar la lógica del negocio gestionando todos los accesos a la información almacenada en Solr.

2.4 Patrones de Diseño

Los patrones de diseño representan la descripción de un problema particular y recurrente, que aparece en contextos específicos, y presenta un esquema genérico demostrado con éxito para su solución; este último se especifica mediante la descripción de los componentes que la constituyen, sus responsabilidades y desarrollos, así como también la forma como estos colaboran entre sí (Larman, 2004).

En el diseño del sistema de identificación de tendencias de la información web indexada en el buscador Orión se tuvieron en cuenta los siguientes patrones GRASP (Patrones Generales de Software para Asignación de Responsabilidades), que describen los principios fundamentales de la asignación de responsabilidades a objetos:

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). En el marco de la presente investigación siguiendo el patrón Experto en Información se le asignaron responsabilidades determinadas solamente a las clases que cuentan con la información necesaria para dar cumplimiento a las mismas. De esta forma se mantiene el encapsulamiento de la información, puesto que los objetos utilizan su propia información para llevar a cabo las tareas. Normalmente, esto conlleva un bajo acoplamiento, lo que da lugar a sistemas más robustos y más fáciles de mantener, el mismo se pone de manifiesto en la clase “DefaultController” cuando se referencia la función “getDoctrine()” que es la encargada de gestionar la base de datos.

Controlador: este patrón tiene como objetivo asignar la responsabilidad a una clase de recibir o manejar un mensaje de evento del sistema generado por un actor externo, por lo general a través de una interfaz gráfica de usuario a la que accede un usuario para realizar ciertas operaciones en el sistema(Larman, 2004).La utilización de este patrón se evidencia en la clase “DefaultController”, la misma se encarga de atender y ofrecer respuesta a cada una de las peticiones realizadas por el usuario mediante la interfaz web.

Bajo Acoplamiento: Cosiste en asignar responsabilidades de modo que el acoplamiento permanezca bajo. Permite la reutilización de las clases y que no sean afectadas por cambios que se realicen en otros componentes. Este patrón se emplea en las distintas capas del sistema mediante el uso de interfaces que relacionan una capa con otra de forma que dichas relaciones no se establezcan directamente hacia las clases. Las conexiones se realizan a través del mecanismo de inyección de dependencias.

Alta Cohesión: La cohesión es una medida del grado de focalización de las responsabilidades de una clase. Permite que las clases sean fáciles de entender, mantener y reutiliza. Se manifiesta en las validaciones.

2.5 Diagrama 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).

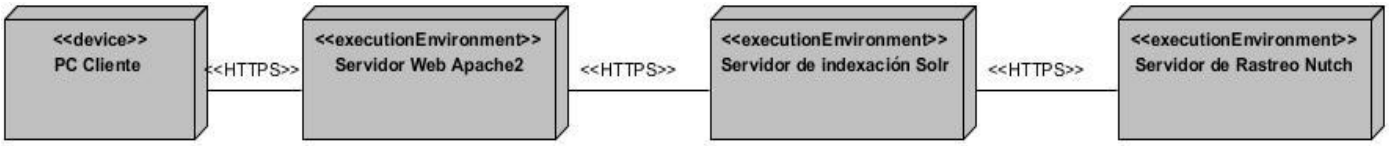


Figura 8: Diagrama de despliegue.

Como se puede apreciar en la Figura 8 el nodo “Pc Cliente” representa un dispositivo utilizado por el usuario desde el cual se podrán mostrar las tendencias generadas, a través del protocolo HTTPS, haciendo uso de un navegador web. El nodo “Servidor web Apache2” es el encargado de atender y ofrecer respuesta a cada una de las solicitudes del cliente. Además, se observan dos nodos más, uno que representa el servidor de indexación Solr con Tomcat7 como contenedor de servlets y otro donde deberá

ser instalado el servidor de rastreo Nutch. Se sugiere que se encuentren en servidores independientes con el objetivo de utilizar al máximo las características de hardware y software de estos, aunque pudieran alojarse en uno solo.

Conclusiones Parciales

En este capítulo se abordaron una serie de aspectos correspondientes al análisis y diseño del Sistema de identificación de tendencias de la información web indexada en el buscador Orión llegándose a las siguientes conclusiones:

- La representación y descripción de los artefactos generados garantizaron un mejor entendimiento de los flujos de trabajos presentes en el proceso de identificación de tendencias.
- La especificación de los requisitos funcionales y no funcionales del sistema, dieron paso a una mejor comprensión, por parte del autor, de los resultados que se pretenden obtener de una manera precisa y sirvieron de guía para la implementación del sistema.
- 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.

Capítulo 3: Implementación y pruebas del sistema de identificación de tendencias de la información web indexada por el buscador Orión.

Introducción

La implementación del sistema es una de las fases ineludibles dentro del proceso de desarrollo de software. Esta fase comprende la materialización, en forma de código, de todos los artefactos, descripciones y arquitectura propuestos en la etapa de análisis y diseño; con el objetivo de conformar el producto final requerido por el cliente.

Aparejado al proceso de implementación, el software que se construye debe ser sometido a determinadas pruebas que corroboren la correspondencia entre el producto y los requisitos definidos en las etapas anteriores. A esta etapa se le conoce como validación del sistema y en ella se realizan diferentes tipos de pruebas en función de los objetivos de las mismas.

3.1 Estándares de codificación

Los estándares de codificación son especificaciones o estilos que establecen la forma de generar el código funcional de las aplicaciones informáticas. Puesto que, en muchas ocasiones, los sistemas de cómputo son implementados por varios programadores, la adopción inicial de un único estilo de codificación constituye uno de los factores de mayor peso en la calidad, rendimiento, legibilidad y capacidad de mantenimiento del producto final. El estándar de codificación utilizado en el lenguaje PHP es el que establece el marco de trabajo Symfony2 que sigue los estándares definidos en los documentos PSR-1, PSR-2, PSR-3, PSR-4 (SensioLabsNetwork, 2014; PHP Framework Interop Group (psr-1), 2014; PHP Framework Interop Group (psr-2), 2014; PHP Framework Interop Group (psr-4), 2014). Por otra parte, el estándar de codificación utilizado para el lenguaje Java es el que establece la comunidad de Nutch para su uso. Entre sus elementos más relevantes y comunes se encuentran:

- Añadir un espacio después de cada delimitador coma ´, ´.
- Añadir un único espacio a ambos lados de un operador: =, ==, && etc.
- En los array multilínea, añade una coma al final de cada elemento, incluido el último.
- Añade un salto de línea antes de una sentencia return, a menos que el return se encuentre solo en un bloque de sentencias, y un salto después de cada llave cierre de sentencia, excepto después de la llave de cierre de clase.
- Usa notación camelCase sin guiones bajos en variables, funciones, métodos y argumentos.

- El nombre de las clases se realiza en UpperCamelCase, es decir, que comienza por mayúscula.

3.2 Diagrama de componentes

Un diagrama de componentes permite visualizar con facilidad la estructura general del sistema y el comportamiento de las funcionalidades que estos componentes proporcionan y utilizan a través de las interfaces. Además, muestra la organización y las dependencias entre un conjunto de componentes.

A continuación, se describen los principales paquetes que componen el diagrama de componentes correspondiente a la interfaz web de la herramienta implementada:

- **TrendsBundle**: Agrupa en su interior todos los componentes de la interfaz web, estableciendo una estructura organizativa acorde al patrón de arquitectura MVC.
- **Controller**: Contiene la clase controladora encargada de procesar las peticiones de las páginas clientes y del usuario según Figura 6: Arquitectura del sistema. y devolver las respuestas con la información requerida.
- **Config**: Contiene en su mayoría los archivos donde se definen las rutas de la aplicación.
- **Views**: Agrupa las páginas referentes a las vistas de la aplicación, así como las plantillas bases.

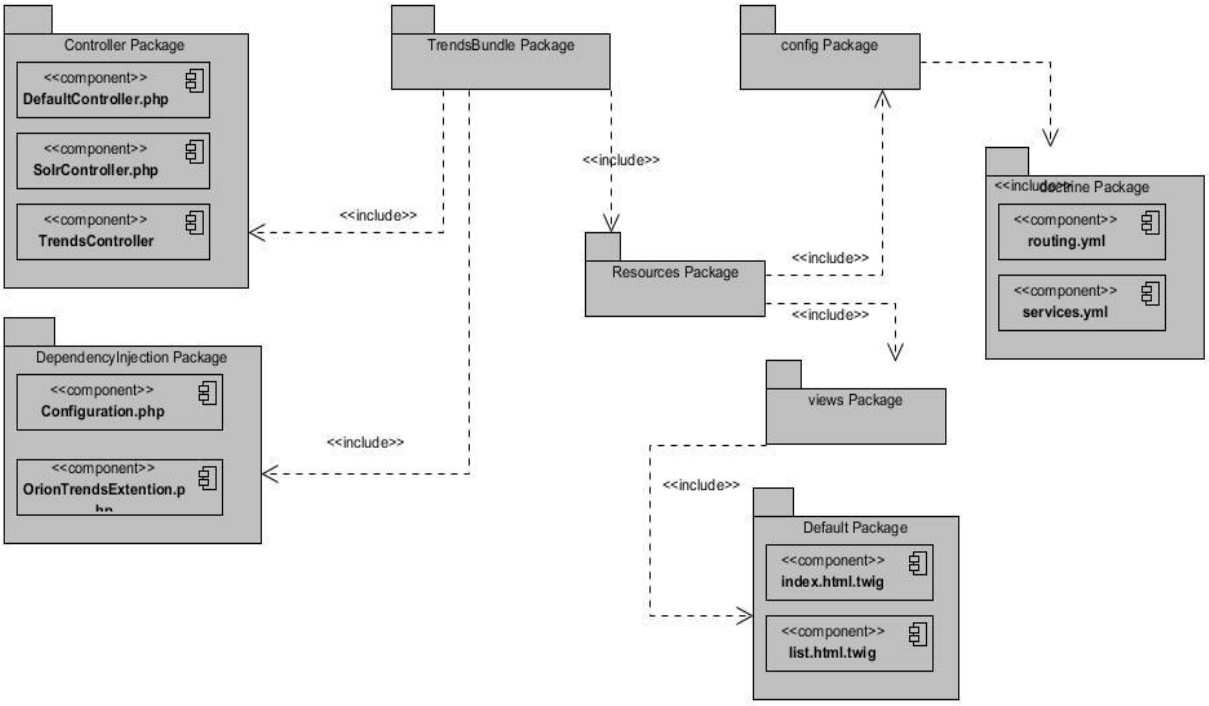


Figura 9: Diagrama de componentes.

3.3 Pruebas funcionales

Las pruebas funcionales son aquellas que se aplican al producto final, y permiten detectar en qué puntos el producto no cumple sus especificaciones funcionales. En ellas se debe verificar si el sistema lleva a cabo correctamente todas las funcionalidades requeridas y la validación de los datos; además se deben realizar pruebas de comportamiento ante distintos escenarios.

Para la puesta en marcha de este tipo de pruebas se hace necesario la presencia del diseño de casos de prueba.

Este tipo de pruebas permite encontrar:

- Funciones incorrectas o ausentes.
- Errores de interfaz.

Tabla 8: Caso de prueba-Listar tendencias actuales

Nombre del requisito	**Descripción**	**Escenarios de prueba**	**Flujo del escenario**

Listar tendencias actuales	Permite mostrar las tendencias actuales a través de un top	EP 1.1 Mostrar las tendencias actuales a través de las palabras claves de los documentos indexados por Sor, generándolas mediante un top	1. El usuario accede a los servicios del buscador Orión. 2. En el menú servicios encontrara Tendencias. 3. Se muestra el servicio de Tendencias de Orión. 4. El usuario podrá observar las tendencias actuales en la parte derecha de su pantalla.

Tabla 9: Caso de prueba-Mostrar tendencias actuales por categorías.

Nombre del requisito	**Descripción**	**Escenarios de prueba**	**Flujo del escenario**
Mostrar tendencias actuales por categorías.	Permite mostrar las tendencias actuales por categorías.	EP 1.1 Mostrar las tendencias actuales a través de una categoría definida.	5. El usuario accede a los servicios del buscador Orión. 6. En el menú servicios encontrara Tendencias. 7. Se muestra el servicio de Tendencias de Orión. 8. El usuario podrá observar las tendencias actuales por categorías en el centro e su pantalla.

Resultados de las pruebas funcionales

Con el objetivo de probar el correcto funcionamiento de las funcionalidades del sistema se realizaron 3 iteraciones de pruebas al sistema. En la Tabla 10 se muestran los resultados obtenidos en cada iteración de prueba al Sistema de identificación de tendencias de la información web indexada por el buscador Orión, así como la corrección de cada uno de los errores.

Tabla 10: Cantidad de no conformidades.

No Conformidades	1era iteración	2da iteración	3era iteración
Detectadas	2	1	0
Resueltas	1	1	0
Pendientes	1	0	0

3.4 Pruebas de integración

El proceso de integración del sistema implica construirlo a partir de sus componentes y probar el sistema resultante para encontrar problemas que pueden surgir debido a la integración. La integración del sistema implica identificar grupos de componentes que proporcionan alguna funcionalidad del sistema e integrar estos añadiendo código para hacer que funcione conjuntamente (Sommerville, 2005).

El motor de búsqueda cubano Orión, cuenta con una estructura que facilita la integración de nuevos subsistemas para incrementar sus funcionalidades. Por tal motivo, se decidió realizar pruebas de integración descendentes. Las cuales, consisten en desarrollar la infraestructura del sistema en su totalidad y luego añadirle los componentes funcionales.

Para la verificación de un correcto funcionamiento entre el sistema desarrollado y el buscador Orión, se llevaron a cabo las acciones siguientes:

- Integración entre el sistema de identificación de tendencias con el motor de búsqueda Orión.
- Verificación de la conexión entre el sistema y Solr.

  Se muestra, en la Tabla 9, los resultados de la ejecución de las dos iteraciones de las pruebas de integración realizadas al sistema:

Tabla 11: Cantidad de no conformidades por cada iteración.

No Conformidades	1era iteración
Detectadas	0
Resueltas	0

Pendientes	0

Una vez realizadas las pruebas de integración y vistos los resultados de las mismas se demuestra que el Sistema de identificación de tendencias de la información web indexada por el buscador Orión se integra correctamente con el buscador Orión y realiza una conexión satisfactoria con el servidor de indexación Solr.

3.5 Pruebas de carga y estrés

Una vez que un sistema se ha integrado correctamente, es posible probar las propiedades emergentes del sistema tales como rendimiento y fiabilidad. Las pruebas de carga y estrés (también conocidas como pruebas de rendimiento) tienen que diseñarse para asegurar que el sistema pueda procesar la carga esperada. Esto normalmente implica planificar una serie de pruebas en las que la carga se va incrementando regularmente hasta que el rendimiento del sistema se haga inaceptable (Sommerville, 2005).

Hardware:

- Tipo de procesador: Intel Celeron R a 2.30GHz
- Memoria: 2GB RAM.
- Tipo de Red: Ethernet 10/100Mbps.

Software:

- Tipo de Servidor Web: Apache2.
- Máximo de hilos concurrentes (simulación de usuarios): 150.
- Plataforma: sistema operativo Kubuntu 16.04 de 32 bits.
- Servidor de BD Solr.

Para un mejor entendimiento de las componentes “Reporte de Rendimiento”, se explica cada parámetro que la compone a continuación:

- **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 un milisegundo 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 12: Resultados de prueba de carga y estrés

Usuarios	# Muestras	Media	Mín	Máx	% Error	Rendimiento (peticiones/segundos)
150	3160	2056	2	65022	0	15.5

Como se muestra en la Tabla 10, la herramienta desarrollada, para 150 usuarios conectados de forma concurrente respondió 3160 peticiones al servidor en un promedio de 2.056 segundos, lo que equivale a 15.5 peticiones por segundo. Los resultados obtenidos demuestran que el sistema es capaz de responder a la carga esperada.

Conclusiones parciales

En este capítulo se abordaron una serie de aspectos correspondientes a la implementación y validación del Sistema de identificación de tendencias de la información web indexada por el buscador Orión llegándose a las siguientes conclusiones:

- La representación y descripción del diagrama de componentes permitió visualizar con más facilidad la estructura general de la herramienta.
- La ejecución de pruebas a la herramienta permitió detectar las deficiencias presentes, subsanarlas en el menor tiempo posible y ofrecer una aplicación con mayor calidad, seguridad y usabilidad.

Conclusiones generales

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

- A partir del estudio realizado de los fundamentos teóricos relacionados con los sistemas que identifican tendencias se determinó que existen una serie funcionalidades que implican un procesamiento previo de la información.
- El enfoque ágil propuesto por la metodología AUP UCI y el uso de las tecnologías y herramientas seleccionadas, permitieron analizar y describir los subprocesos que se debían ejecutar, concretando así, en concordancia con las especificaciones del cliente, las características que debía tener la herramienta a desarrollarse.
- Una vez estudiados los elementos que intervienen en el Sistema de identificación de tendencias de la información web indexada por el buscador Orión, fue posible la modelación de los artefactos que contribuyeron al diseño de la propuesta de solución posibilitando un mayor soporte a la implementación de los requisitos previamente expresados por el cliente; garantizando la estructura base para la organización lógica del código fuente y la disminución del impacto ante futuras modificaciones en la aplicación.
- La implementación del Sistema de identificación de tendencias de la información web indexada por el buscador Orión 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.
- La evaluación de las pruebas de software realizadas permitió erradicar las insuficiencias detectadas en el desarrollada 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:

- Implementar funcionalidades que permitan identificar tendencias por provincias.
- Añadir funcionalidades que permitan al sistema identificar tendencias a través de las consultas realizadas por los usuarios.

Referencias Bibliográficas

Alegsa, L. (12 de octubre de 2010). *DICCIONARIO DE INFORMÁTICA Y TECNOLOGÍA.* Recuperado el 5 de noviembre de 2016, de http://www.alegsa.com.ar/Dic/sgbd.php.

Adell, Jordi. Redes y educación. *Nuevas tecnologías, comunicación audiovisual y educación. Barcelona: Cedecs*, 1998, p. 177-211.

Aksynoff, A. Sphinx 2.3.2-dev reference manual. [En línea]. Sphinx | Open Source Search Server, 2014. [Citado el: 14 de Noviembre de 2014.]. Disponible en: [http://sphinxsearch.com/docs/current.html].

ElasticSearch. Apache Solr vs ElasticSearch. [En línea]. Apache Solr vs ElasticSearch - the Feature Smackdown!, 2014. Disponible en: [http://solr-vs-elasticsearch.com/].

Pugh, D., S., E. Apache Solr 3 Enterprise Search Server. s.l.: Pack Publishing, 2009. pág. 418. ISBN 978-1-84951-606-8.

Nugraha, A. Indexing Bibliographic Database Content Using MariaDB and Sphinx Search Server. [En línea]. The Code4Lib Journal – Indexing Bibliographic Database Content Using MariaDB and Sphinx Search Server, 2014. [Citado el: 25 de Octubre de 2014.] Disponible en: [http://journal.code4lib.org/articles/9793].

The Apache Software Foundation-Tika. Apache Tika. [En línea]. Apache Solr, 2014. [Citado el: 13 de Octubre de 2014.] Disponible en: [http://tika.apache.org/].

Gormley, C.; TONG, Z.. Elasticsearch: The Definitive Guide. [En línea]. Elastic, 2014. [Citado el: 11 de 66 Noviembre de 2014.] Disponible en: [http://www.elasticsearch.org/guide/].

Db-Engines. DB-Engines Ranking - Trend of Elasticsearch Popularity. [En línea]. Historical trend of Elasticsearch popularity, 2014. [Citado el: 14 de Enero de 2015.] Disponible en: [http://dbengines.com/en/ranking_trend/system/Elasticsearch].

Potencier, F. What is Symfony2?. [En línea]. SensioLabsNetwork, 2011. [Citado el: 18 de Abril de 2015.] Disponible en: [http://fabien.potencier.org/article/49/what-is-symfony2].

Db-Engines. System Properties Comparison Elasticsearch vs. Solr vs. Sphinx. [En línea]. Elasticsearch vs. Solr vs. Sphinx Comparison, 2015. [Citado el: 22 de Marzo de 2015.] Disponible en: [http://dbengines.com/en/system/Elasticsearch%3BSolr%3BSphinx].

Sphinx. System Properties Comparison Elasticsearch vs. Solr vs. Sphinx. [En línea] 2014. [Citado el: 14 de Febrero de 2015.] Disponible en: [http://db-engines.com/en/system/Elasticsearch%3BSolr%3BSphinx].

Slideshare Comparing open source search engines. [En línea]. Slideshare, 2014. [Citado el: 11 de Marzo de 2015.]. Disponible en:[http://www.slideshare.net/rboulton/comparing-open-source-searchengines].

Nieto, I., A., M. Universidad Nacional de Colombia. [En línea] 2009. [Citado el: 10 de Octubre de 2014.] . Disponible en: [http://dis.unal.edu.co/profesores/eleon/cursos/tamd/presentaciones/nutch.pdf].

Barkov, A. mnoGoSearch 3.3.15 reference manual: Full-featured search engine software.[En línea]. 2014. [Citado el: 13 de Octubre de 2014.] Disponible en: [http://www.mnogosearch.org/doc33/msearchintro.html#features].

Otto, M.; Thornton, J. Bootstrap3 Manual Oficial. [trad.] Javier Eguiluz. 2013

The Apache Software Foundation-Solr. Apache Solr. [En línea]. Apache Solr , 2014. [Citado el: 3 de Octubre de 2014.] Disponible en: [http://lucene.apache.org/solr/].

Kumar, S P. Integration of Web mining and web crawler: Relevance and State of Art. s.l. : International Journal on Computer Science and Engineering, 2010. págs. 772-776. Vol. 2. ISSN: 0975-3397.

Departament of Ccomputer Sciences University of Chile. WIRE - Web Information Retrieval Environment. [En línea]. WIRE (Web Information Retrieval Environment): Center for Web Research, 2011. [Citado el: 11 de Diciembre de 2014.]. Disponible en: [http://www.cwr.cl/projects/WIRE/]

Sánchez, B., S. La importancia de lo visual (Un ejemplo con fotografías). [En línea] 2014. [Citado el: 22 de Octubre de 2014.]. Disponible en: [http://cvc.cervantes.es/ensenanza/biblioteca_ele/asele/pdf/08/08_0757.pdf].

Camargo, F., I.; Salinas, S., O. Evolución y tendencias actuales de los Web crawlers. 2, 2013, Vol. 18.

Acunetix. Audit your website security with Acunetix Web Vulnerability Scanner. [En línea]. Acunetix, 2015. [Citado el: 26 de Mayo de 2015.]. Disponible en: [http://www.acunetix.com/].

Bakken, S. S. Manual de PHP. s.l. : The PHP Documentation Group, 2013. pág. 1063.

Lie, H., W.; BOS, B. Cascading Style Sheets – designing for the Web, ISBN: 0321193121, 2005.

García, J. D. (2012). *Desarrollo de Aplicaciones web con symfony 1.4.* Recuperado el 29 de octubre de 2016, de http://juandarodriguez.es/cursosf14/unidad7.html..

García, J. F. (2010). Innovación en modelos de negocios. La metodología de OSTERWALDER en la práctica. En J. F. García.

Acosta, J.; Greiner, C.; DAPOZO, G.; ESTAYNO, M. Medición de atributos POO en frameworks de desarrollo PHP.. Buenos Aires, Argentina : s.n., 2014.

Jalón, J., G.; RODRÍGUEZ, J., I.; MINGO, I.; IMAZ, A.; BRAZALÉZ, A.; LARZABAL, A.; CALLEJA, J. Aprenda Java como si estuviera en primero. Navarra : s.n., 2000.

Larman, C. (1999). *Uml y Patrones.Introducción al análisis y diseño orientado a objetos. .* PRENTICE HAL.

NETCRAFT. Web server survey. Sitio web de NetCraft. [En línea]. 2015. Disponible en: [http://news.netcraft.com/].

Sparxsystems. Diagrama de Despliegue UML 2. [En línea]. Sparx Systems - Tutorial UML 2 - Diagrama de Despliegue, 2014. [Citado el: 24 de Febrero de 2015.] Disponible en: [http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html].

Oracle. Sitio oficial del IDE NetBeans. [En línea]. NetBeans IDE Features, 2014. [Citado el: 24 de Octubre de 2014.]. Disponible en: [https://netbeans.org/features/index.htm].

Sánchez, A. (2014). *Prezi*. Recuperado el 5 de noviembre de 2016, de https://prezi.com/7wmx8_d6ertl/entorno-desarrollo-integrado-ide/

Sánchez, T. R. (2014). *Metodología de desarrollo para la Actividad productiva UCI.* La Habana: UCI.

Seta, L. D. (11 de junio de 2010). *http://www.dosideas.com/noticias/java/913-apache-solr-una-introduccion*. Recuperado el 6 de diciembre de 2016, de http://www.dosideas.com