Universidad de las Ciencias Informáticas
Facultad 8

Sistema para la recuperación de información aplicando técnicas de trabajo colaborativo.

TRABAJO PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS INFORMÁTICAS

Autores: Anniel Rodríguez Izquierdo.

Dionnis Osorio Lores.

Tutores: Ing. Eduardo Manuel Macías Sotolongo

Co-Tutor: Ing. Fidel Alberto Curbelo Rosell

Ciudad de la Habana, junio de 2009

"Año del 50 aniversario del triunfo de la Revolución"

DECLARACIÓN DE AUTORÍA

Declaramos que somos los autores de este trabajo y autorizamos a la Facultad 8 de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año 2009.

AUTOR
Dionnis Osorio Lores

AUTOR
Anniel Rodríguez Izquierdo

TUTOR
Eduardo Manuel Macías Sotolongo

Dedicatoria

A mis padres: mi “**Dios**” y mi “**Franca**”, que han sido mis guías, mi mejor y primera escuela. Éste también es su trabajo.

A mis hermanos: **Randys**, **Maisilis**, **Maryanis** y **Orlando**.

A mis abuelos, tíos, primos, sobrinas, a todos; que siempre han estado pendientes de mi desempeño, dándome ánimos y apoyo.

A todas mis amistades.

Dionnis Osorio Lores

A mis dos madres y a mis padres: María Elena y Felicia, Jesús y Efraín por ser siempre mis guías y mis ejemplos, espero que estén tan orgullosos de mí como yo lo estoy de ellos, gracias por existir y estar siempre a mi lado.

A mis hermanas: Anmy, Yanisbel, Anisleidys, Cari y Ana Cecilia.

A todos les dedico este título en retribución a la confianza que han depositado en mí.

Gracias.

Anniel Rodríguez Izquierdo

Agradecimientos.

Agradecimientos especiales a mis familiares, que me han apoyado en todo momento, que siempre han estado y estarán en los momentos más importantes de mi vida.

A mi compañero de tesis Anniel Rodríguez Izquierdo, gracias.

Agradezco a todos aquellos que de una forma directa o indirecta han incidido en la voluntad de convertirme en mejor persona, en profesional entregado y comprometido con la revolución y con el deber de informatizar a nuestra sociedad.

Agradezco a nuestro tutor, que ha puesto mucho empeño en este trabajo, tanto como si fuese su propio Trabajo de Diploma.

Agradezco a Jorge Amado Soria, Tomás Orlando Junco, Yunesti Pérez la Rosa y demás compañeros que nos ayudaron.

Agradezco especialmente a Fidel Castro Ruz, por la brillante idea de crear este centro de altos estudios.
Agradezco especialmente a la Revolución y la Universidad por esta oportunidad.

Dionnis Osorio Lores

A mis padres y familia de crianza, Efraín, Felicia entre otros, quienes me han sabido encaminar en la vida, dándome consejos y ayudando a esforzarme en los estudios cada día.

A mis verdaderos padres María Elena Izquierdo y Jesús Rodríguez, dos personas que han incentivado en mi el amor por los estudios, por el amor que me dan cada día, su confianza y por haberme dado la vida.

A mi hermana por estar siempre apoyándome y siguiendo mis pasos.

A mi tutor Eduardo Manuel por su interés incondicional y por toda su ayuda.

A la Universidad y a la Revolución por darme la oportunidad de graduarme como Ingeniero en Ciencias Informáticas.

A mi compañeros de tesis Dionnis Osorio que me ayudado mucho para que este trabajo tuviese la calidad requerida, muchas gracias.

A todos los que desde los primeros días en la vida como estudiante hasta hoy, a todas las personas que de alguna forma u otra han influido en mi educación, en mis valores como persona y en la realización de este trabajo.

Anniel Rodríguez Izquierdo

Resumen

El presente trabajo reúne la información resultante de la investigación y desarrollo de una aplicación web aplicando técnicas de trabajo colaborativo para la recuperación de información en la web de la Universidad de las Ciencias Informáticas. Incluye un estudio del estado del arte, las herramientas utilizadas, las características del sistema y de las tareas realizadas para su diseño e implementación.

Como resultado de esta investigación se presenta un prototipo completamente funcional del software que debe servir de base para profundizar en el funcionamiento de los Sistemas de Recuperación de Información, para incentivar la técnica de trabajo colaborativo en los proyectos de software libre de la Universidad y la posibilidad de poseer una herramienta operacional para acceder a la información web disponible en la universidad.

ABSTRACT

This paper presents information on the research and development of a web application applying collaborative techniques for information retrieval in the web of the University of Informatics Science. It includes a study of the state of the art, the tools used, the characteristics of the system and the tasks performed to design and implement the application.

The result of this investigation is a fully functional prototype of the software that serves as a base to further the performance of information retrieval systems, to encourage collaborative work in free software projects of the University and to give the opportunity to own an operational tool to access all available information on the university.

Índice

Introducción

Con el desarrollo de internet, ha habido un crecimiento exponencial de la información en la red de redes, haciéndose extremadamente complicado recuperarla y organizarla. Razón por la cual han surgido aplicaciones encargadas de recuperar, centralizar, organizar y clasificar todos esos datos, dentro de los que podemos citar: Google, Yahoo Search, MSN.

Los sistemas de recuperación de información, son aplicaciones con una función específica: buscar la información disponible en las páginas en internet, guardarla en una base de datos, para luego mostrarla al usuario a través de enlaces en respuesta a las palabras claves introducidas por dicho usuario.

De igual forma, la Universidad de las Ciencias Informáticas (UCI) cuenta ya con seis años de vida y el volumen de información en su red, ha estado creciendo constantemente, pero también es creciente la cantidad de personas pertenecientes al centro que no conocen de la existencia de esta información y los que saben de su existencia no poseen una herramienta que les permita acceder a ésta.

Gestionar la información existente en los sitios web puede ser un proceso engorroso, por lo que sin la presencia de una herramienta eficiente para la recuperación de información, el acceso a los datos se convierte en un proceso difícil; al diseño e implementación de una herramienta para la recuperación de la información web, que brinde servicios de búsqueda de archivos, imágenes, es que está dirigido este trabajo.

La UCI, creada con una fuerte base tecnológica y un amplio perfil productivo ha hecho intentos por desarrollar buscadores web entre los que podemos citar: GPI++, el buscador de la Biblioteca de la UCI, y otros que no son tan relevantes; los que no han tenido éxito debido a sus limitaciones y escaso nivel de operatividad.

Desde hace algún tiempo se han venido aplicando técnicas de trabajo colaborativo en muchos entornos de trabajo (en los cuales se encuentran aplicaciones y sitios web), con el objetivo de fortalecer funcionalidades y servicios a través de la colaboración de los usuarios, técnica que se ha aplicado también en los sistemas de recuperación de información, un ejemplo se puede encontrar en el buscador *WIKIA SEARCH*, que intenta mejorar los resultados de las consultas valiéndose de las opiniones y sugerencias de las personas que hacen uso de dicha herramienta.

El frecuente uso de los buscadores internos de la universidad, ha demostrado la poca capacidad de éstos de brindar de forma adecuada y coherente la información que los usuarios usualmente solicitan, información que se encuentran disponible en los servidores web y de archivos del centro. La poca calidad de los resultados devueltos, la gran cantidad de enlaces duplicados o enlaces a páginas que por alguna razón ya no existen o no están disponibles, además de la ineficiente implementación de los algoritmos de recuperación de información, conlleva a plantearnos la siguiente interrogante. ¿Será posible implementar una aplicación colaborativa para la recuperación eficiente de la información web, donde se incentiven las técnicas de trabajo colaborativo en estudiantes y profesores de la UCI?

Teniendo en cuenta lo anteriormente expuesto en la universidad se viene desarrollando una estrategia para implementar un buscador web con un enfoque colaborativo y de código abierto en respuesta a los ineficientes servicios de búsqueda en la red interna.

Planteándose el siguiente **problema científico de esta investigación**:

¿Cómo facilitar de manera rápida, eficiente, organizada y segura la gestión de la información web y otros formatos en la UCI?

Objeto de estudio:

Los sistemas de Recuperación de Información en la web.

Campo de acción:

La Recuperación de Información web en la UCI.

Objetivo general:

Desarrollar una aplicación para la recuperación de información web aplicando técnicas de trabajo colaborativo que brinde servicios de búsqueda a la comunidad universitaria.

Objetivos específicos:

1. Realizar un estudio del arte y de los principales elementos teóricos y conceptos relacionados con los buscadores web existentes en la universidad y en el país.
2. Diseñar e implementar algoritmos de minería de datos.
3. Diseñar e implementar algoritmos de posicionamiento web.
4. Fomentar el trabajo colaborativo en la comunidad universitaria.

Tareas:

1. Realizar un estudio de los buscadores existentes dentro y fuera de la universidad para conocer el funcionamiento de los sistemas de recuperación de la información.
2. Realizar un estudio de los diferentes modelos de recuperación de información existentes en la actualidad y sus tendencias.
3. Identificar las principales deficiencias y potencialidades de los portales de búsqueda existentes en la UCI.
4. Identificar la plataforma y el gestor de base de datos sobre el cual estará soportado el buscador.
5. Identificar los requerimientos de hardware necesarios para el funcionamiento del buscador.

Resultados esperados:

1. Crear un sistema informático que contribuya a la búsqueda eficiente de la información web disponible en la red universitaria.
2. Incentivar el trabajo colaborativo en estudiantes y profesores.
3. Sentar las bases para los futuros servicios de búsqueda web en la universidad.
4. Fomentar la posibilidad real de materializar toda una gama de estudios sobre buscadores web en la UCI.
5. Crear una comunidad en la cual todos sus integrantes, colaboren y ayuden a mejorar el código del buscador haciendo propuestas.

Capítulo 1: Diseño Teórico

Recuperación de Información

La recuperación de información es el conjunto de tareas mediante las cuales el usuario localiza y accede a los recursos de información que son pertinentes para la resolución del problema planteado. En éstas tareas desempeñan un papel fundamental los lenguajes documentales, las técnicas de resumen, la descripción del objeto documental, etc.

En principio, la recuperación de información engloba las acciones encaminadas a identificar, seleccionar y acceder a los recursos de información útiles al usuario, sin perjuicio de otras acepciones del concepto, en las que puede profundizarse utilizando la bibliografía correspondiente. (1)

El proceso de recuperación se lleva a cabo mediante consultas a la base de datos donde se almacena la información ya estructurada, mediante un lenguaje de interrogación adecuado. Es necesario tener en cuenta los elementos o claves que permiten hacer la búsqueda, determinando un mayor grado de pertinencia y precisión, como son: los índices, las palabras claves, tesauros y los fenómenos que se pueden dar en el proceso como son el ruido y silencio documental. Uno de los problemas que surgen en la búsqueda de información es si lo que recuperamos es "mucho o poco", es decir, dependiendo del tipo de búsqueda se pueden recuperar multitud de documentos o simplemente un número muy reducido. A este fenómeno se denomina Silencio o Ruido documental.

- Silencio documental: Son aquellos documentos almacenados en la base de datos pero que no han sido recuperados, debido a que la estrategia de búsqueda ha sido demasiado específica o que las palabras claves utilizadas no son las adecuadas para definir la búsqueda.
- Ruido documental: Son aquellos documentos recuperados por el sistema pero que no son relevantes. Esto suele ocurrir cuando la estrategia de búsqueda se ha definido demasiado genérica.

Técnicas de recuperación de información.

- Sistemas de recuperación de lógica difusa.

  Esta técnica permite establecer consultas con frases normales, de forma que la máquina al realizar la búsqueda elimina signos de puntuación, artículos, conjunciones, plurales, tiempos verbales, palabras comunes (que suelen aparecer en todos los documentos), dejando sólo

aquellas palabras que el sistema considera relevantes. La recuperación se basa en proposiciones lógicas con valores de verdadero y falso, teniendo en cuenta la localización de la palabra en el documento.

- Técnicas de ponderación de términos.

  Es común que unos criterios en la búsqueda tengan más valor que otros, por tanto la ponderación pretende darle un valor adecuado a la búsqueda dependiendo de los intereses del usuario. Los documentos recuperados se encuentran en función del valor obtenido en la ponderación. El valor depende de los términos pertinentes que contenga el documento y la frecuencia con que se repita. De forma, que el documento más pertinente de búsqueda sería aquel que tenga representado todos los términos de búsqueda y además el que más valor tenga repetidos más veces, independientemente de donde se localicen en el documento.

- Técnica de *clustering.*

  Es un modelo probabilístico que permite las frecuencias de los términos de búsqueda en los documentos recuperados. Se atribuyen unos valores (pesos) que actúan como agentes para agrupar los documentos por orden de importancia, mediante algoritmos de ranking.

- Técnicas de retroalimentación por relevancia.

  Esta técnica pretende obtener el mayor número de documentos relevantes tras establecer varias estrategias de búsqueda. La idea es que, tras determinar unos criterios de búsqueda y observar los documentos recuperados se vuelva a repetir nuevamente la consulta pero esta vez con los elementos interesantes, seleccionados de los documentos primeramente recuperados.

- Técnicas de *stemming.*

  Morfológicamente las palabras están estructuradas en prefijos, sufijos y la raíz. La técnica de *Stemming* lo que pretende es eliminar las posibles confusiones semánticas que se puedan dar en la búsqueda de un concepto, para ello trunca la palabra y busca solo por la raíz.

  Pretenden acotar de una manera eficaz los documentos relevantes. Por esta razón, esta técnica lo consigue mediante una correcta indización en el proceso de tratamiento de los documentos con ayuda de índices, tesauros, etc.; evitando las ambigüedades léxicas y semánticas a la hora de establecer las consultas.

Calidad de la recuperación

A continuación se presentan unos criterios básicos para que la recuperación llevada a cabo sea de calidad.

- Consistencia: capacidad que tiene un sistema de búsqueda en coordinar su sistema de clasificación con el lenguaje de búsqueda, permitiendo de esta manera establecer ecuaciones de búsqueda sobre términos admitidos.
- Exhaustividad: cualidad de un sistema de información para recuperar la totalidad de los documentos relevantes que posee una colección, conforme a los requerimientos establecidos en la estrategia de búsqueda.
- Tasa de acierto: coeficiente que surge de dividir el número de documentos relevantes recuperados, sobre el número total de documentos relevantes de la colección.
- Relevancia: característica de un documento recuperado de cumplir con la necesidad de información.
- Tasa de relevancia: coeficiente que surge de dividir el número de documentos relevantes recuperados, sobre el número total de documentos recuperados.
- Pertinencia: cualidad que tiene el documento recuperado de adaptarse a las necesidades de información.
- Tasa de pertinencia: coeficiente que surge de dividir el número de documentos pertinentes recuperados, sobre el número total de documentos recuperados.
- Precisión: es la capacidad que tiene el sistema de búsqueda en coordinar la ecuación con los documentos más relevantes. De otra forma son aquellos documentos relevantes recuperados.
- Tasa de precisión: coeficiente que surge de dividir el número de documentos relevantes recuperados, sobre el número total de documentos de la colección.

Herramientas de recuperación de información

Herramientas informáticas que permiten establecer ecuaciones de búsqueda específicas para acceder a una información previamente almacenada, aquí se hace referencia a algunas de ellas:

- Buscadores: los buscadores son herramientas que permiten localizar y recuperar la información almacenada en internet. Almacenan las páginas con determinadas características (metadatos) y que posteriormente tras utilizar unas palabras claves emiten un listado de las páginas más relevantes.

- Las revistas electrónicas son publicaciones periódicas que se generan a través de elementos electrónicos. Sus características principales son la rápida difusión, el ahorro de coste y la fiabilidad para su uso, ya que un documento electrónico puede ser manipulado constantemente.
- Directorios: los directorios son listas organizadas que nos permite acceder a la información de forma estructurada y jerárquica. Se clasifican en categorías y el usuario enlaza de lo más general a lo más específico.
- Los agentes inteligentes son herramientas que permiten localizar información de forma automática, sólo necesitan que se le defina un perfil de búsqueda y dónde deben lanzarla (bases de datos, sitios web, etc.) y, automáticamente va presentando un informe sobre la nueva información que va surgiendo. (2)

Problemáticas en la UCI para la recuperación de Información

La Universidad de las Ciencias Informáticas cuenta con algunos *años de vida*, años durante los cuales se ha ido generando gran cantidad de información relacionada con la docencia, la productividad, el quehacer cotidiano del estudiante universitario y las demás actividades relacionadas con la universidad y su personal.

Desde los comienzos de este centro como comunidad universitaria se han venido desarrollando un sinnúmero de actividades que han contribuido a que el volumen de información acumulado en este tiempo haya sido bastante grande, comenzando con que la escuela cuenta con 10 facultades, cada una ajustada a un perfil diferente, y cada una de éstas realiza muchas actividades por separado además de las actividades comunes que comparte con el resto de las facultades. La universidad cuenta hoy en día con unos diez mil estudiantes de todos los años y más de dos mil profesores, de los cuales más de la mitad del estudiantado y todos los profesores tienen acceso pleno a internet, de donde se puede acceder a los cientos de millones de páginas que están alojadas en la web como material de consulta para el aprendizaje y como referencia.

En la universidad se realizan múltiples actividades que generan un gran volumen de información, ligadas a la producción, la investigación, la formación del personal, las tecnologías, los servicios, la docencia, el entretenimiento, la salud, el deporte, la causa revolucionaria, organizaciones políticas y de masas, la cultura, la historia, las noticias, etc. Dentro de todas éstas categorías y otras donde la UCI juega un papel importante, se pueden nombrar las Jornadas Científicas Estudiantiles, los Seminarios Juveniles Martianos, mi Web por Cuba, las Copas de Programación, los Juegos Deportivos tanto los Inter-

facultades como los Inter-años, los sitios creados para llevar mensajes de salud a todo el personal de la Universidad como la Jornada Científica del Hospital y los sitios para defender la causa de la revolución. Otros tantos que forman parte de ése gran volumen se encuentran ligados a la docencia y a la producción, entre éstos podemos citar los sitios creados para el programa docente educativo de los estudiantes, donde nombramos los pertenecientes a los siguientes departamentos:

Departamentos Docentes
- Dpto. Matemática
- Dpto. Matemática Aplicada
- Dpto. Práctica Profesional
- Dpto. Preparación para la Defensa
- Dpto. de Idiomas Extranjeros
- Dpto. de Física
- Dpto. Ciencias Empresariales
- Dpto. de Marxismo
- Dpto. Ingeniería y Gestión de Software

Direcciones Docentes
- Dirección de Teleformación.
- Dirección de Formación Postgraduada.
- Dirección de Deporte

Centro de Investigación
- CICE

Otros:
- Tesis.UCI
- Entorno Virtual de Aprendizaje
- Teleclases.
- Infodrez.

Gestión Académica:
- Akademos.

Sin contar los sitios que forman parte de cada uno de los proyectos de las distintas facultades y que son utilizados para la comunicación de los estudiantes dentro de los proyectos productivos, o los que son creados por los estudiantes cuando llevan a la práctica lo aprendido en las aulas; los sitios creados dentro de la universidad relacionados a la producción, normalmente pertenecen a:

La FEU y la Producción
- Grupos de Producción de la FEU.
- Foros de las Comunidades de Producción.
- Herramienta de Desarrollo Colaborativo GForge.
- Repositorio de ISOs de Linux.
- Repositorio de la Distribución de Linux Ubuntu.
- Repositorio de la Distribución de Linux Debian.
- Repositorio de la Distribución de Linux Gentoo.
- Repositorio de la Distribución de Linux Suse.
- Software Educativo e Hipermedia.

La comunidad universitaria hace uso además de múltiples servicios, los que poseen su sitio en la web o sus páginas en ésta.

No toda la información de la universidad se encuentra en sitios web, sino también en sitios de archivos, donde se almacenan los programas y la información usada en la docencia y la producción, llegando a cantidades inmensas de información.

Actualmente dentro de la UCI deben existir unos cuantos miles de páginas web y ficheros, con información referente a los temas abordados con anterioridad en este mismo capítulo, pero por la no interrelación que existe entre estos sitios, el rápido cambio de la información que contienen, la desaparición en sí de estos sitios, su mala implementación, las limitaciones de los buscadores con los que cuenta hoy la universidad; y principalmente por la inexistencia de una herramienta que recupere la información y la almacene en alguna Base de Datos para luego mostrarla a los usuarios de la comunidad, se hace muy difícil la gestión de información que se encuentra en alguna parte de la red de la Universidad.

Trabajo Colaborativo

El trabajo colaborativo es definido como “la nominación general y neutral de múltiples personas que trabajan juntas para producir un producto o servicio”. (3)

En estos últimos años se ha hecho creciente una forma de trabajo que permite a un grupo de personas ser apoyados por otras en la realización de una tarea, si se trata de trabajo en equipo, cada uno de los miembros del equipo puede dar su aporte al resto de los compañeros, pero no sólo aporte, sino también criterio, comentarios, ideas y experiencias; esta forma de trabajo, es el trabajo colaborativo. El trabajo colaborativo se usa en ambientes donde los miembros de un equipo tratan de lograr una meta común, alcanzar un objetivo desarrollándolo de la mejor de sus formas, aportando la mejor de las vías para la solución del problema; no se trata de una competencia entre los miembros, sino de plasmar la mejor de las soluciones para dar solventado un problema relacionado con la rama donde esta forma de trabajo sea aplicada. El trabajo colaborativo se crea en un ambiente donde es beneficioso o porque se requiere para la realización de una tarea determinada.

El trabajo colaborativo se ha visto últimamente más impulsado, por el desarrollo de las Tecnologías de la Informática y las Comunicaciones (TIC), donde es posible el intercambio de información de muchas personas simultáneamente independientemente de su localización o posición geográfica, es decir; donde los individuos trabajan juntos, por las características de sus tareas o porque se hace necesario.

Principalmente cuando se trata de desarrollo de software, específicamente cuando se implementa un sistema, el lenguaje que se utiliza es un lenguaje común a todos los miembros del grupo, el lenguaje de programación. Con el trabajo colaborativo se facilita y hace más rápido el entendimiento de lo realizado por un programador, y si se utilizan algunas de las técnicas de programación, de la Programación Orientada a Objetos (POO), el trabajo es aún más fácil, porque agiliza el entendimiento de las secciones de código entre desarrolladores del sistema, logrando así que el tiempo que tome leer, entender, modificar para mejorar sea más breve. El trabajo colaborativo no se refiere sólo al desarrollo de un sistema, el logro de una tarea en fase de construcción, sino cuando el sistema está en pleno funcionamiento, siendo usado por el usuario final, donde podrá a través de la aplicación dar sus criterios, aportar ideas para mejorar el funcionamiento del sistema, usando interfaces o funcionalidades habilitadas para estas funciones.

En internet funcionan hoy herramientas para la búsqueda y recuperación de información donde se aplican técnicas de trabajo colaborativo, tal es el caso de *Wikia Search*, que además de ser un software *Open Source* el usuario puede ante determinado parámetro de búsqueda, si la respuesta devuelta por el buscador no se ajusta a la clave introducida, introducir una sugerencia que se ajuste al resultado.

Ej. Sugerir para la clave “HTML AND XML” el resultado: www.desarrolloweb.com

Herramientas Colaborativas

Las herramientas donde se aplican trabajo colaborativo han adquirido gran auge con el desarrollo de internet y los medios de transmisión de información en los últimos años, principalmente en áreas como la educación a distancia, los blogs, etc. Dentro de este sinnúmero de herramientas no se han quedado atrás las que se encargan de organizar y clasificar la información para el usuario, facilitándole el acceso de forma rápida y eficiente a la información; ejemplo de este tipo de herramienta se encuentra *Wikia Search*.

Estudio de herramientas donde se aplican técnicas de trabajo colaborativo

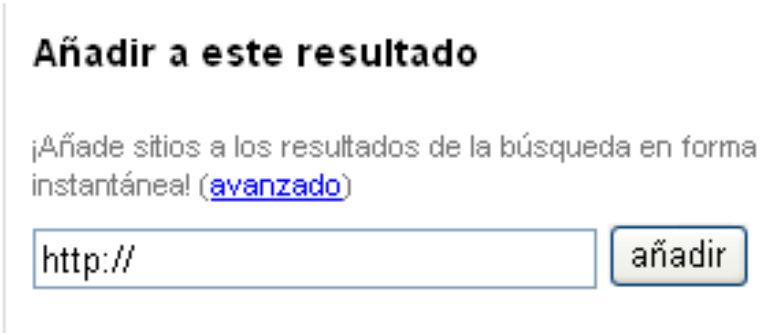
Wikia Search

Wikia Search es un buscador colaborativo que posee un motor de búsqueda de código abierto, que a diferencia de los motores de búsqueda conocidos que utilizan algoritmos para las búsquedas, éste categoriza los resultados a partir de las aportaciones de los usuarios, que contribuyen a decidir **cómo clasificar y filtrar** los resultados de las búsquedas. Los principios básicos del buscador son: **Transparencia, Comunidad, Calidad y Privacidad;** los que se describirían brevemente de la siguiente forma, principalmente la transparencia consiste en mostrar cómo operan los algoritmos y el sistema en sí, la comunidad se refiere a dar participación importante a los usuarios, nutriendo al sitio de las críticas y las sugerencias de dicha comunidad para ser cada vez mejores, la calidad es para mejorar la relevancia y ocurrencia de los resultados de las búsquedas, y la privacidad hace apego a la privacidad en las búsquedas, las que deben ser respetadas tanto a nivel tecnológico como a nivel social.

A la hora de realizar una búsqueda, *Wikia Search* ofrece tres índices de ordenación de resultados para la misma búsqueda: *Whitelist*, que es el índice por defecto y está basado en la indización de enlaces destacados. Además están *Smaller Test* y *Visvo*, ambos índices basados en la tecnología de *Nutch*, un método de búsquedas que tiene en cuenta las categorizaciones de los usuarios.

Wikia Search permite realizar acciones sobre los resultados devueltos ante las búsquedas, lo que permite caracterizar, clasificar, identificar, organizar o ignorar los resultados con las opciones editar, anotar, destacar, comentar, borrar, además de que muestra iconos que identifican el resultado en cuestión, diferenciándolo de los demás cuando se trata de un video, documento, música, o se trata de un resultado de otro buscador, etc. Entre las funcionalidades se pueden destacar, editar los elementos de cualquier resultado, añadir resultados de forma inmediata, eliminar u ocultar resultados, darles puntuación, sugerir búsquedas relacionadas, y publicar comentarios en un resultado.

De los aspectos que más sobresalen de *Wikia Search* es la posibilidad de recomendar direcciones URL (Uniform Resource Location en español Identificador Uniforme de Recurso) para los resultados devueltos de una búsqueda determinada, URL que más tarde será revisada por un grupo de edición que se encargará de aprobar o no la sugerencia, evitando que se realicen recomendaciones no ajustables al contenido de la búsqueda o se realice algún vandalismo.



Ej.: Sugerencia de URL para el resultado de una búsqueda.

El rating o ranking le permite al usuario dar un valor a un resultado con respecto a una búsqueda, que sirve como método de evaluación y puede ser de más relevancia para búsquedas futuras. Ante una clave de búsqueda muestra una serie de palabras claves similares relacionadas o no con la suministrada.

> Ej. Si se introduce el término Fidel, mostraría enlaces de búsqueda a Fidelidad, fidelity, fidelcolor, fidelitas, fidelio, fidealización, así como Castro, Alejandro Castro Ruz, entre muchos otros.

Motores de Búsqueda

Los buscadores o motores de búsqueda, son herramientas que catalogan las páginas web por medio de mecanismos llamados spiders o robots de búsqueda. Los spiders (arañas) son robots de búsqueda virtuales que recorren millones de páginas en segundos y las clasifican de acuerdo a su contenido (temáticamente). Estas arañas luego ofrecen ese mismo contenido al público por medio de palabras claves (en inglés llamadas keywords). [Los buscadores generalmente se componen de tres partes fundamentales:]

- La Base de Datos Documental.
- El Robot o Spider (Web Crawler).
- El Sistema de Recuperación de Información.

Los motores de búsqueda basados en *crawlers* consisten en bases de datos muy voluminosas generadas como resultado de la indexación de partes significativas de los documentos que han sido analizados previamente en Internet. Los motores de búsqueda suelen recoger documentos en formato *HTML* y otros tipos de recursos. La tarea es realizada por un programa denominado crawler (*robot o spider*) que recorre la red de forma automática explorando los servidores a nivel mundial, o en el ámbito de especialización del buscador (geográfico, idiomático o temático). La recuperación se realiza gracias

a un sistema de gestión de base de datos que permite distintos tipos de consulta y a la ordenación de los resultados por relevancia, en función a la estrategia de consulta. Los motores de búsqueda son más exhaustivos que los índices en cuanto al volumen de páginas referenciadas, pero son mucho menos precisos que los índices, al no ser su contenido objeto de indexación humana.

Existe una gran cantidad de motores de búsqueda en Internet, cada uno ofrece diferencias en cuanto a volumen de páginas, elementos de cada página que son indexados, interfaz, lenguaje de consulta, algoritmo de cálculo de la relevancia, etc. Estas diferencias provocan que los resultados de aplicar una misma consulta a varios buscadores en ocasiones no coincidan. A la hora de valorar la calidad de un buscador se debe tener en cuenta:

- La exhaustividad: número de documentos de Internet referenciados que almacena el motor de búsqueda en su base de datos, para las consultas.
- La calidad y flexibilidad del lenguaje de consulta: indica que tanto se pueden mejorar los resultados de una consulta en base a los operadores con los que cuenta el motor.
- La pertinencia de sus resultados (ruido y silencio): el número de resultados arrojados en una consulta no debe ser tan pequeño como para no proporcionar suficiente información, ni tan grande como para no poder definir cuáles son los resultados relevantes.
- Los servicios de valor añadido que incorporan: tales como correo electrónico, compras en Internet, noticias, disco virtual, mensajero electrónico, etc.
- La periodicidad de actualización de la base de datos: la frecuencia con la que el crawler regresa a los sitios que tiene indexados para verificar si alguno de ellos ha actualizado sus páginas, si el sitio ya no existe, o para registrar los sitios nuevos.
- La velocidad en la recuperación: la velocidad de respuesta a una consulta, es decir, el tiempo que toma el motor de búsqueda en consultar su índice y aplicar el algoritmo para regresar los resultados.
- Las dificultades de conexión: la facilidad con la cual se puede accesar al sitio del motor de búsqueda.

Los motores de búsqueda basan la recuperación en el uso de palabras claves y en la ordenación de los resultados de búsqueda por relevancia. Utilizan un programa que se comporta como un navegador pero además almacena el contenido en una forma que la hace fácil de recuperar posteriormente, este programa es conocido como crawler, *robot o spider*.

Un crawler recupera un documento y recursivamente todos los documentos con los que mantiene vínculos, indexa la información de acuerdo a un criterio predefinido. Los criterios son: el título del

documento, los meta datos, el número de veces que se repite una palabra en un documento, algoritmos para valorar la relevancia del documento, etc. y el peso de cada criterio varia de acuerdo al motor de búsqueda. La información se almacena en una base de datos, la cual puede ser consultada por los usuarios de Internet para recuperar la información deseada. Para mantener actualizada la base de datos, los *crawlers* vuelven a visitar los sitios para verificar que las páginas registradas se mantengan activas, de no ser así (cuando se mueven a otro sitio o desaparecen) las eliminan de la base de datos.

Los sitios de Internet necesitan ser registrados de tal forma que puedan aparecer en los resultados de los motores de búsqueda. Dependiendo del motor de búsqueda, alguien o alguna computadora decidirá si la *URL* se agrega a su base de datos o no. Tomará unos segundos al crawler examinar las páginas y almacenar la información relevante en la base de datos. Existen herramientas que realizan el registro del sitio en una gran cantidad de motores de búsqueda, pero también se puede registrar de forma manual. Algunas herramientas de registro cobran por el servicio, pero hay otras gratuitas, tal es el caso del servicio del sitio *broadcaster* (*www.broadcaster.co.uk*) en el reino unido (*UK*).

La mayoría de los motores de búsqueda verifican el número de veces que se repiten las palabras claves en la página, después buscan estas palabras en el nombre del dominio o en la *URL*, posteriormente en el título de la página, en el encabezado y en los meta datos. El orden en que se busca en cada uno de los elementos antes mencionados llega a variar, dependiendo del motor de búsqueda, y además cada uno utiliza sus propios algoritmos en los cuales incluyen criterios diferentes. Si el motor de búsqueda encuentra las palabras claves en todos estos criterios, entonces obtiene un estímulo para obtener una clasificación mayor.

Los motores de búsqueda proporcionan una forma para saber cuántas y cuáles páginas mantienen enlaces a un sitio. Usan comandos especiales y el nombre del dominio del sitio, en el nombre del dominio el prefijo *http://* y la *www* no son necesarios. A continuación se presenta la forma de hacerlo en algunos de ellos:

- ***AltaVista y Google:*** Para buscar páginas enlazadas a un sitio se introduce *link: dominio*. Se puede reducir la búsqueda a una *URL* particular siendo más específico: *link: dominio/paginahtml.* Para eliminar las páginas del mismo sitio que se enlazan entre si, se usa el comando *-url: link: dominio -url: dominio*.
- ***AllTheWeb.com:*** Para páginas enlazadas a un sitio se introduce *link.all: dominio*.
- ***Inktomi:*** Para búsquedas de todo el sitio se usa *linkdomain: dominio*, pero varios socios de inktomi no lo implementan. Para eliminar los resultados de las páginas del mismo sitio se usa *linkdomain: dominio -domain: dominio*. Los comandos de Inktomi funcionan también para **AOL**

(America On Line), *HotBot*, *iWon* y *MSN* (motor de búsqueda de *T1MSN*). Si se requiere encontrar los enlaces a una página en *HotBot* o *MSN* se introduce la *URL* completa incluyendo el prefijo *http://*.

CARACTERÍSTICAS DE RASTREO (CRAWLING)

Es importante conocer la forma como los *crawlers* actuarán sobre las páginas que rastrean, ya que de ello depende el éxito del registro completo del sitio y alcanzar una buena clasificación.

- ***Rastreo profundo:*** el motor de búsqueda lista muchas páginas de un sitio, aún si no están explícitamente registradas en él.
- **S*oporte de marcos:*** es una característica que permite a los motores de búsqueda seguir los enlaces a través de los marcos (*frames*).
- ***Mapas de imágenes:*** son enlaces a otras páginas a través de imágenes.
- ***Robots.txt:*** es un archivo de texto que permite indicar que páginas no deben ser indexadas en el sitio.
- ***Meta índice robot:*** tiene el mismo objetivo que el *robots.txt*, pero este es una instrucción del código *HTML* de la página.
- ***Rastreo por enlaces de popularidad:*** la popularidad de una página se detecta analizando cuantos enlaces existen hacia otra página. Los motores de búsqueda usan esa característica para determinar que páginas deben incluir en el índice de su base de datos, aunque esto no necesariamente indica que obtendrán una buena clasificación.
- ***Aprende por frecuencia:*** el motor de búsqueda aprende con que frecuencia se modifican las páginas, para estimar el tiempo en el que volverá a visitarlas el crawler.
- ***Inclusión pagada:*** muestra si el motor de búsqueda ofrece un programa donde se pueda pagar para garantizar que las páginas de un sitio se incluyan en el índice. Esto no es lo mismo que colocación pagada, la cual además de la inclusión en el índice, garantiza una posición en particular en relación a un término de búsqueda.

CARACTERÍSTICAS DE INDEXACIÓN

Las características de indexación indican lo que se indexa cuando el motor de búsqueda rastrea la página.

- ***Texto completo:*** indexan todo el texto visible en el cuerpo de la página, aunque algunos no indexan algunas palabras (*stop words*) o las excluyen por parecer spam.

- ***Stop words:*** algunos motores de búsqueda omiten palabras cuando indexan la página o al menos no las consideran durante la consulta. Estas palabras son excluidas para ahorrar espacio o aumentar la rapidez de búsqueda, ya que son palabras que aparentan ser *spam*.
- ***Meta descripción y meta palabras claves:*** son *meta índices* que describen el contenido de la página y los términos con los que se le asocia para la búsqueda.
- ***Texto alternativo y comentarios:*** el texto alternativo es aquel que se asocia con una imagen para describirla brevemente, el texto alternativo es parte del lenguaje *HTML*. Los comentarios suelen ser una anotación sobre la página y son un tipo de meta índice.

CARACTERÍSTICAS DE CLASIFICACIÓN

La mayoría de los motores de búsqueda usan la ubicación y la frecuencia de las palabras claves en las páginas como la base de clasificación en respuesta a una consulta. Además pueden ser relevantes algunos factores que estimulan la clasificación, tales como:

- ***Estímulo de clasificación por meta índices:*** algunos motores de búsqueda suelen dar un estímulo a las páginas que contienen *meta índices* si coinciden con los términos de búsqueda.
- ***Estímulo de clasificación por enlaces de popularidad:*** los motores de búsqueda pueden determinar la popularidad de una página por el número de enlaces que existen a ella desde otras páginas.
- ***Estímulo de clasificación por aciertos directos:*** es un sistema que mide las preferencias de los usuarios sobre la lista de resultados que le presentan para refinar la relevancia de la clasificación.(4)

Cuil

Cuil es una palabra galeana, que representa dos términos: conocimiento y avellana, el oficial ejecutivo en jefe (*CEO*) de Cuil es de origen irlandés, Tom Costello. Irlanda es un país con una cultura mitológica muy rica, la leyenda de la búsqueda de la sabiduría planteaba que un salmón se había comido nueve nueces que habían caído dentro de "La Fuente de la Sabiduría", cualquiera que se comiese el salmón adquiriría toda aquella sabiduría.

Cuil es una herramienta de recuperación de información propietaria, su filosofía se basa en dos aspectos fundamentales, en cómo indexar todo internet y no parte de ésta, así como analizar y organizar sus páginas para obtener resultados de relevancia.

Este buscador da importancia a muchos parámetros, como por ejemplo plantean que el tamaño interesa, no se trata de sólo tener un conjunto de lo más importante de un tema determinado, sino todo sobre ése contenido, porque puede ser necesitado en algún momento.

Los directivos de Cuil plantean que la popularidad es útil pero no es siempre importante, debido que ésta es generalmente la respuesta a las búsquedas más sencillas, pero cuando se trata de búsquedas más complejas prefieren hacer una extracción de todas las páginas que contengan los términos de búsqueda, y luego de analizar el contenido de esas páginas teniendo en cuenta el significado de los términos en el contexto específico, las clasifican para dar el resultado a la búsqueda.

Para Cuil la organización es fundamental, para ello separan la información en cuanto a ideas, separar una idea de la otra, de forma que el usuario encuentre la idea que le interesa, y cada uno de los resultados asocian una imagen, de forma que visualmente se tenga una referencia sobre lo que trata la idea en cuestión.

Cuil plantea que analiza la información y no los clic que los usuarios realizan, así como la información de éstos:- "la privacidad es un tema caliente en estos días", que no se detienen en almacenar información

sobre los hábitos de las personas ni su historial de búsquedas, a Cuil no les interesa quién realiza las consultas, de dónde lo hacen; no coleccionan información de identificación personal.

Funcionalmente Cuil es un motor de búsqueda interesante, por las facilidades que le brinda al usuario a la hora de realizar una consulta y durante la creación de ésta, mostrándole al usuario algunas opciones y sugerencias de búsquedas, después de la consulta muestra otros términos relacionados con la búsqueda realizada en cuestión, en un panel, indizados por categoría, de forma que el usuario puede diferenciar una idea de otra u acceder a la información adicional que se muestra o simplemente redefinir la búsqueda con los nuevos términos suministrados. Posee una amplia cobertura con más de un millón de millones de páginas indexadas, no posee corrector de ortografía a la hora de la búsqueda, no ofrece la posibilidad de elegir entre un formulario simple y otro más detallado, no siempre devuelve documentos relevantes en función de la búsqueda realizada o no detalla cuáles son más relevantes que otros.

Google

De los sistemas de recuperación de información en la web, Google es el más exitoso.

La innovadora tecnología de búsqueda Google y su diseño de interfaz de usuario diferencian a Google de las máquinas de búsqueda de primera generación. Se basa en los hipertextos, analiza todo el contenido de cada web y la posición de todos los términos en cada página. Se da prioridad a los resultados de acuerdo con la proximidad de los términos de la búsqueda, favoreciendo los resultados en los que los términos de búsqueda están próximos entre sí, sin perder tiempo analizando resultados irrelevantes.

Google nació de la idea de Larry Page y Sergey Brin. Ambos tenían un objetivo en común: conseguir información relevante a partir de una importante cantidad de datos. En enero de 1996 iniciaron su colaboración en un buscador llamado BackRub. Larry empezó a trabajar en la forma de conseguir un

entorno para los servidores que funcionara con PCs de gama baja y que no necesitará de potentes máquinas para funcionar. Un año después, la tecnología utilizada por BackRub para analizar los links empezaba a ser conocida, obteniendo una gran reputación. Era la base sobre la que se construiría Google.

El nombre proviene de un juego de palabras con el término "googol", acuñado por Milton Sirotta, sobrino del matemático norteamericano Edward Kasner, para referirse al número representado por un 1 seguido de 100 ceros. El uso del término refleja la misión de la compañía de organizar la inmensa cantidad de información disponible en la web y en el mundo.

Durante los primeros meses de 1998, Larry y Sergey continuaron trabajando para perfeccionar la tecnología de búsqueda que habían desarrollado. Utilizaron sus dormitorios como centro de datos y oficinas.

El 7 de septiembre de 1998, Google Inc. ya disponía de oficinas propias en Menlo Park, California. Google.com, todavía en fase beta, tenía unas 10000 búsquedas cada día.

Google se basa en la tecnología PageRank, lo que asegura que los resultados más importantes se muestren primero. PageRank mide objetivamente la importancia de las páginas web y se calcula que resuelve una ecuación de 500 millones de variables y más de 2000 millones de términos. Los complejos mecanismos automáticos de búsqueda de Google permiten prescindir de la interferencia humana. Está estructurado de manera que nadie puede comprar un lugar privilegiado en la lista ni alterar los resultados con fines comerciales. (5)

Yahoo

Yahoo! empezó como el hobby de unos estudiantes.

David Filo y Jerry Yang prácticamente pasaban más tiempo organizando sus links favoritos que con sus obligaciones universitarias. La lista de webs era cada vez más grande, así que decidieron dividirla en categorías. Como las categorías no eran suficientes para tal cantidad de información, crearon subcategorías… Era el nacimiento de Yahoo!.

En un principio la página se llamó “Jerry's Guide to the World Wide Web” (Guía de Jerry de la World Wide Web), luego cambiando de nombre. Yahoo! son las siglas de "Yet Another Hierarchical Officious Oracle” (Otro Oficioso Oráculo Jerárquico), aunque Filo y Yang insistían en que el nombre provenía de la definición general de “yahoo” (grosero, sencillo, tosco). Yahoo! empezó hospedándose en el ordenador de Yang, "Akebono," mientras que el mecanismo de búsqueda se guardaba en el de Filo, “Konishiki” (ambos eran nombres de legendarios luchadores de Sumo).

David Filo y el Dr. Jerry Yang, los fundadores de Yahoo!, eran estudiantes de doctorado de Ingeniería Eléctrica en la Universidad de Stanford cuando decidieron iniciar una recopilación de sus intereses en Internet creando la guía Yahoo! a principios de 1994.

Jerry y David pronto se dieron cuenta de que no eran los únicos que estaban interesados en un sitio donde se pudiera encontrar una base de datos con las páginas más útiles e interesantes. Cientos de personas accedían a esa información, incluso fuera de Stanford. A finales de 1994, se celebraba el millón de visitas y los casi 100000 visitantes únicos.
En abril de 1995 se fundó Yahoo!

Conscientes de que la nueva empresa tenía un enorme potencial para crecer rápidamente, Filo y Yang empezaron a constituir su equipo directivo. Contrataron a Tim Koogle, un veterano de Motorola y ex alumno del departamento de ingeniería de Stanford, y a Jeffrey Mallett, fundador de la división del consumidor de WordPerfect.

Al utilizar básicamente la base de datos de Google, su funcionamiento es idéntico. Los criterios son los mismos que utiliza Google, ya que Yahoo! toma los resultados de la misma base de datos.

Yahoo! utiliza la base de datos de su propio directorio y la de Google. Con la compra de Inktomi se pretende ganar independencia respecto a Google. (6)

GPI

El buscador GPI es un buscador desarrollado por el grupo de procesamiento de imágenes de la UCI, es un buscador pequeño, sin muchas complejidades tanto en su búsqueda como en las claves que acepta, sólo se tienen en cuenta en la consulta:

1. Todas las palabras claves introducidas.
2. Cualquier palabra.
3. La frase completa.

GPI++ devuelve resultados que contengan estos términos en su contenido pero no se tienen en cuenta la similitud término-artículo, tampoco la relación con los términos de búsqueda en sí, sólo devuelve aquellos archivos donde se encuentren las claves introducidos en la búsqueda. Este sistema presenta graves fallas, principalmente porque devuelve enlaces a contenido no disponible (enlaces rotos). Si se realizara una búsqueda con las claves “menú del día”, puede devolver información referente a términos que poseen en su contenido frases como “Las organizaciones tienen a menudo infraestructuras heterogéneas…”.

Biblioteca

El buscador de la biblioteca de la Universidad de Ciencias Informáticas o catálogo en línea es un software utilizado para desarrollar el sistema integrado de automatización de bibliotecas**,** desarrollado por la UNESCO como gestor de base de datos documentales, su característica fundamental es que es un software para la búsqueda de documentos dentro de la base de datos de la biblioteca UCI, por lo que no se puede hacer extensible para la búsqueda de información en la red en el centro. Posee opciones de búsqueda avanzada, búsqueda simple, un historial y un manual de usuario.

La búsqueda simple se puede realizar por palabras claves introducidas por el usuario, se pueden categorizar los resultados, siendo estos libros, artículos, revistas y tesis; se pueden hacer uso de operadores AND, OR, NOT en las búsquedas. La búsqueda avanzada permite además la búsqueda por palabras claves, título del documento, por autor, idioma, institución y permite la búsqueda de revistas. Esta forma de búsqueda adquiere un poco de complejidad para el personal que no sabe hacer uso de los operadores, estudiantes de primer año y trabajadores relacionados con la universidad y que generalmente no son informáticos, absteniéndolos a realizar búsquedas simples haciendo uso de

palabras claves, reduciendo en gran medida la efectividad y amplitud de la búsqueda para obtener buenos resultados.

Resultados de Motores de Búsqueda.

Sin dudas algunas *Wikia Search* posee buenas características para convertirse en un buen sistema de recuperación de información, pero para alcanzar este resultado debe tomar un buen tiempo, porque es un sistema que crece constantemente y se retroalimenta de cada búsqueda que realicen los usuarios en todas partes del mundo, a diferencia de otros sistemas de búsqueda *Wikia Search* se fortalece por poseer la ayuda del hombre para clasificar sus resultados, principalmente porque la información no se puede socializar, con esto término se refiere a que es difícil mediante un algoritmo obtener exactamente una correspondencia clave-contenido, no así con la actividad humana, aunque se debe tener en cuenta que los algoritmos son más rápidos que los humanos, y pueden ser modificados con frecuencia para ganar en exactitud, por otra parte tener en cuenta además que los errores humanos son posibles y la capacidad de procesamiento del hombre es más lenta que la que poseen los equipos de cómputo, por lo que éste puede ser un punto de debilidad para la *Wikia Search*, otro aspecto negativo con respecto a la *Wikia* es que depende fundamentalmente de la labor del hombre, y ésta labor depende mucho de su motivación y sus conocimientos, lo que provocaría que principalmente cuando se realiza una búsqueda es porque se quiere acceder de forma rápida a una información precisa y de calidad, por lo que sería un “problema ” para muchos acceder a un sistema que probablemente no devuelva excelentes, buenos o ningún resultado, lo que sin dudas llevaría al usuario a usar los buscadores que se basan en algoritmos para realizar sus búsquedas.

Cuil es un buen sistema de recuperación de información principalmente porque explota muchas de las características que usan los otros sistemas de recuperación de información, pero le falta algo por avanzar, agregar muchas otras funcionalidades que hagan más fácil la labor de búsqueda a los usuarios, así como la agregación de otras que le permitan atraer a muchos otros, como la operatividad en diferentes idiomas, permitir un mejor uso de operadores para hacer las búsquedas más complejas y más especificas, ajustadas a un tema especifico, permitir más flexibilidad a la hora de redefinir la búsqueda con nuevos parámetros de la clave.

Yahoo es un directorio excelente, por sus características, tanto de poder realizar las búsquedas en forma jerárquica, y el uso de su propio sistema de base de datos como el sistema de base de datos de Google.

Google es sin objeción el motor más exitoso, ese mérito lo ha ganado principalmente por la velocidad y calidad de sus resultados, a pesar de que no tiene en cuenta todos o casi todos los archivos existentes

en internet, pero al menos los resultados que devuelve en sus primeras páginas generalmente cumplen con las expectativas que todo el que usa este buscador espera, buena información en un corto período de tiempo.

GPI++ y Biblioteca UCI no son sistemas de recuperación propiamente dichos, principalmente por sus características, primero que no poseen spider o arañas que escudriñen la red en busca de los documentos que van a ser almacenados en sus bases de datos, además de que se encuentran limitados al funcionamiento en el entorno al que han sido creados.

Tomando en cuenta todas las características de estos buscadores para el desarrollo de un sistema de recuperación de información para la UCI, donde se aprovecharan las potencialidades de unos y de otros, principalmente se tendrían en cuenta las características de *Wikia Search* pero con un sistema de recuperación de información como el de Google, automático y no manual; principalmente porque es una herramienta donde se lleva a cabo trabajo colaborativo, donde los usuarios hacen sus aportes para el mejoramiento tanto funcional como de los resultados de la herramienta, por las facilidades a la hora de hacer recomendaciones, criticas, realizar aportes; por las características de la Universidad, donde existe toda una comunidad con muchas expectativas por un sistema que resolverá muchos de los problemas a la hora de acceder a una información que se encuentra en la red y que no se sabe dónde está exactamente. Se tomaría además de Cuil el aspecto de indexar toda la información y no sólo parte de ésta, sencillamente porque cualquier información que se tenga en la red puede ser de relevancia para alguien a la hora de desarrollar cualquier tipo de trabajo, en cualquier momento, no siendo así solo con una parte de la información disponible.

Posicionamiento Web

El arte o la ciencia de posicionar las páginas Web con un alta en los buscadores para ciertas palabras claves que son buscadas con mucha frecuencia. Un posicionamiento Web efectivo es una forma de mercadeo que le puede traer mucho tráfico virtual a los sitios que estén en las primeras posiciones de los buscadores para las palabras claves más efectivas. (7)

El posicionamiento se refiere a la forma en que las páginas de una web aparecen en las listas que ofrecen los buscadores. Una web excelentemente posicionada aparecerá en las primeras posiciones de la primera página de los resultados de una búsqueda. Los servicios de posicionamiento que se ofrecen, tienen por objetivo colocar las páginas de una web en esas posiciones de excelencia. (8)

El posicionamiento web se refiere a la ocupación que ocupan los sitios dentro de los buscadores, la cual se realiza a través de algoritmos que se encargan de encontrar la posición adecuada del sitio en respuestas a determinadas consultas de búsquedas de información realizadas por los usuarios.

El posicionamiento web posee muchas características, depende del cambio dinámico de la información en el tiempo, así como de las diferencias en los algoritmos de posicionamiento, el cambio de los propios algoritmos y las diferencias en sí en cada uno de los buscadores.

Los algoritmos de posicionamiento web son algoritmos secretos, porque si se descubriese la dinámica del algoritmo se podría buscar técnicas para agenciarse un buen posicionamiento dentro de los distintos buscadores.

El posicionamiento es de mucha importancia, pues trata de separar la mala información de la buena información, si se posee un sitio web bien posicionado se traduce en que la información que contiene debe ser buena, pero además que los usuarios la encuentran más rápido y el sitio puede ser más visitado.

Existen múltiples técnicas éticas para lograr tener un sitio web bien posicionado, en las que se destacan el desarrollo de un sitio útil, con información de calidad, información relacionada con el propósito con el que fue creado el sitio en primer lugar, con buena navegabilidad, de forma que pueda ser accesible por el usuario, bien implementado y con buenas palabras claves (META[1]) ajustadas al contenido del sitio, de forma que pueda mejorar la posición del sitio en la web. Existen otras técnicas que se pueden utilizar para lograr una buena posición dentro de los buscadores, técnicas no éticas, dentro de las que se destacan las granjas de enlaces, palabras con el mismo color que el fondo de la página que no guardan relación con la información del sitio, usar palabras claves que no se ajustan al contenido (META), entre otras. Los sitios que usan éstas técnicas no éticas generalmente son expulsados de los buscadores, una vez que son descubiertos.

Posicionamiento en Google

El posicionamiento de Google depende de muchos factores, pero de entre todos ellos destacan dos elementos muy importantes, la relevancia de una página con respecto a determinado contenido, y el otro elemento es el PageRank.

El PageRank es un algoritmo creado por Larry Page, el cual lleva su nombre, es un algoritmo que calcula el posicionamiento de una página dependiendo de su popularidad, es decir , si se posee un sitio, y muchos otros sitios hacen enlaces a éste, significa que ese sitio les da un voto, esto supone que la información del sitio referenciado debería ser relevante, esto depende también de quienes referencien a

dicho sitio, ser referenciado por un sitio de importancia supone que el sitio a que se hace enlace debe ser también de importancia, no siendo igual al ser enlazado por un sitio que no posee mucha importancia. La forma del cálculo se realiza de la siguiente forma, un sitio A referenciado por otro sitio B que posee Rank 7 y 40 enlaces salientes le dará al sitio A un Rank de 0.175 (7/40), pero un sitio B que posea Rank 4, y solo 4 enlaces salientes le proveerá al sitio A un Rank 1 (4/4). El PageRank es un algoritmo logarítmico, esto significa que es más fácil pasar de Rank 2 a 3 que de 7 a 8.

Posicionamiento en Yahoo

WebRank es número del 1 al 10 mediante el cual Yahoo asigna una relevancia a páginas webs individuales, y no a un sitio completo. WebRank es calculado usando una compleja fórmula secreta que se encuentra bajo constante cambio. Expertos apuntan a que el valor de la relevancia se calcula mediante la popularidad de los sitios, la que está dada por la antigüedad de éste como forma de medición de su credibilidad; otros apuntan que a Yahoo sigue los pasos de Google, por lo que cuanto más páginas apunten a nuestras páginas más alto será el valor de WebRank, elevando así la posición en la lista de resultados de la página, una página bien posicionada recibirá más visitas.

Modelos de Recuperación de Información

En los últimos años, la gran cantidad de información disponible en la web ha hecho de mucha importancia el uso de los buscadores y directorios para encontrar información en la red, y dentro de éstos se ha hecho aún de más importancia los métodos utilizados para la recuperación de información, los que se clasifican en dos grupos, los modelos de recuperación de información clásicos y los estructurales.

Modelos de recuperación de información clásicos:

Dentro de este modelo se encuentran el modelo probabilístico, booleano y vectorial.

Modelos de recuperación de información estructurales:

Dentro de este modelo se encuentran el modelo de listas no sobrepuestas y el método de los nodos no proximales.

Modelo de Recuperación de Información Booleano

Es un modelo de recuperación simple, basado en la teoría de conjuntos y el álgebra booleana. Dada su inherente simplicidad y su pulcro formalismo ha recibido gran atención y sido adoptado por muchos de los primeros sistemas bibliográficos comerciales. Su estrategia de recuperación está basada en un criterio de decisión binario (pertinente o no pertinente) sin ninguna noción de escala de medida, sin noción de un emparejamiento parcial en las condiciones de la pregunta.

Para el **modelo de recuperación booleano**, las variables de peso de los términos índice son todas binarias. El modelo booleano es todavía el modelo dominante en los sistemas comerciales de bases de datos de documentos y proporciona un buen punto de partida.

En éste modelo el método de representación se hace definiendo a los documentos como un conjunto de términos de indexación o palabras claves.

- **Diccionario:** Conjunto de todos los términos T = {t1, t2, t3,...}.
- **Documento:** Conjunto de términos del diccionario donde tiene valor Di = {t1, t2, t3,...} donde cada uno de los ti = Verdad si es una palabra clave del documento.

Las preguntas son expresiones booleanas cuyos componentes son términos del diccionario:

- **Operadores:** O (U), Y (∩), No (-)

El algoritmo utilizado en el **método booleano** permite calcular el valor de la función de semejanza. Como entrada se tienen listas ordenadas ascendentemente y como salida una lista ordenada con la mezcla de las dos listas de entrada.

El método de ordenación puede ser el número de identificación de los documentos que agrupan los términos a recuperar. Para todo esto se necesitará, una función que devuelva los identificadores de los documentos que contienen el término de la búsqueda, lo cual es sencillo si se observa el archivo invertido y luego se mezclan las listas.

Los beneficios de utilizar éste método es que es un modelo de recuperación sencillo. Mientras que la problemática es que básicamente se tiene que considerar la relevancia como un aspecto puramente binario. (9)

Modelo de Recuperación de Información Vectorial

Ordenando los documentos recuperados en orden decreciente a este grado de similitud, el **modelo de recuperación vectorial** toma en consideración documentos que sólo se emparejan parcialmente con la pregunta, así el conjunto de la respuesta con los documentos alineados es mucho más preciso (en el

sentido que empareja mejor la necesidad de información del usuario) que el conjunto recuperado por el modelo booleano. Los rendimientos de alineación del conjunto de la respuesta son difíciles de mejorar.
La mayoría de los motores de búsqueda lo implementan como estructura de datos y que el alineamiento suele realizarse en función del parecido (o similitud) de la pregunta con los documentos almacenados.

Funcionamiento.

La idea básica de este modelo de recuperación vectorial reside en la construcción de una matriz (*podría llamarse tabla*) de términos y documentos, donde las filas fueran estos últimos y las columnas correspondieran a los términos incluidos en ellos. Así, las filas de esta matriz (que en términos algebraicos se denominan ***vectores***) serían equivalentes a los documentos que se expresarían en función de las apariciones (***frecuencia***) de cada término. De esta manera, un documento podría expresarse de la manera:

- **d1= (1, 2, 0, 0, 0,... ... ..., 1, 3):** Siendo cada uno de estos valores el número de veces que aparece cada término en el documento.

La longitud del vector de documentos sería igual al total de términos de la matriz (el número de columnas).
De esta manera, un conjunto de m documentos se almacenaría en una matriz de m filas por n columnas, siendo n el total de términos almacenamos en ese conjunto de documentos. La segunda idea asociada a este modelo es calcular la similitud entre la pregunta (que se convertiría en el vector pregunta, expresado en función de la aparición de los n términos en la expresión de búsqueda) y los m vectores de documentos almacenados. Los más similares serían aquellos que deberían colocarse en los primeros lugares de la respuesta.

Cálculo de la similitud.

Se dispone de varias fórmulas para realizar este cálculo, la más conocida es la ***Función del Coseno***, que equivale a calcular el producto escalar de dos vectores de documentos ***(A y B)*** y dividirlo por la raíz cuadrada del sumatorio de los componentes del ***vector A*** multiplicada por la raíz cuadrada del sumatorio de los componentes del ***vector B***.
De esta manera se calcula el valor de similitud. Si no existe coincidencia alguna entre los componentes, la similitud de los vectores será cero ya que el producto escalar será cero (circunstancia muy frecuente en la realidad ya que los vectores llegan a tener miles de componentes y se da el caso de la no coincidencia con mayor frecuencia de lo que cabría pensar).

La **similitud máxima** sólo se da ***cuando todos los componentes de los vectores son iguales***, en éste caso la función del coseno obtiene su máximo valor, la unidad. Lo normal es que los términos de las columnas de la matriz hayan sido filtrados (supresión de palabras vacías) y que en lugar de corresponder a palabras, equivalgan a su raíz *'stemmed'* (agrupamiento de términos en función de su base léxica común, por ejemplo: economista, económico, economía, económicamente,
etc.). Generalmente las tildes y las mayúsculas/minúsculas son ignoradas. Esto se hace para que las dimensiones de la matriz, de por sí considerablemente grandes no alcancen valores imposibles de gestionar. (9)

Modelo de Recuperación de Información Probabilístico

Dentro de la recuperación probabilística, se analizará el **modelo de recuperación probabilístico de independencia de términos binarios** donde: "*La probabilidad de los términos es independiente (un término es independiente de los otros)" y "los pesos asignados a los términos son binarios".*
La equiparación probabilística se basa en que, dados un documento y una pregunta, es posible calcular la probabilidad de que ése documento sea relevante para esa pregunta.
Si un documento es seleccionado aleatoriamente de la base de datos hay cierta probabilidad de que sea relevante a la pregunta. Si una base de datos contiene N documentos, n de ellos son relevantes, entonces la probabilidad se estima en:

- **P(rel) = n/ N**

En concordancia con la teoría de la probabilidad, la de que un documento no sea relevante a una pregunta dada viene expresada por la siguiente fórmula:

- **P(↓rel) = 1 - P (rel) = N - n/ N**

Los documentos no son elegidos aleatoriamente, sino que se eligen sobre la base de la equiparación con la pregunta, basado en el análisis de los términos contenidos en ambos. Así, la idea de relevancia está relacionada con los términos de la pregunta que aparecen en el documento. Dividiendo la colección de documentos en dos conjuntos: los que responden a la pregunta y los que no. (9)

Métodos de Organización de Información

Métodos de Indización

Concepto de Indización.

La indexación no es más que el proceso de organizar la información, de crear un índice, generalmente en bases de datos, de un formato sencillo y procesable, de los documentos recuperados por los sistemas de recuperación de información con el objetivo de obtener resultados relevantes, de forma rápida ante determinada búsqueda de información.

Indexación en buscadores

- **Indexación automática**

La indexación automática permite dar de alta un sitio web en los buscadores de forma automatizada. Existen programas informáticos y sitios especializados que lo hacen.
Si se realiza sin control, no garantizan la inclusión del sitio en la base de datos de los buscadores, incluso algunos motores de búsqueda rechazan las inclusiones automatizadas.

- **La indexación manual**

La indexación manual permite una intervención personalizada sobre cada uno de los motores importantes. Donde los expertos asignan los términos de indexación a partir de un conjunto finito de términos.

- **Indexación libre**

Los autores de los textos asignan los términos de indexación teniendo en cuenta que términos utilizarían los usuarios a la hora de buscar información de sus textos.

- **Indexación a través del lenguaje natural**

Utilizando potentes técnicas de procesamiento del lenguaje natural el ordenador asigna los términos de indexación a partir del análisis de los textos.

Niveles de Indización

- **Indexación en el nivel submorfológico**

Sin realizar análisis sintáctico ni semántico, el análisis morfológico, ofrece un método muy flexible para la recuperación; la información se indexa como patrones de bits de forma que el texto, el sonido y las imágenes en movimiento, pueden indexarse y recuperarse de la misma manera.

- **Indexación por palabra clave**

Se crean índices inversos de raíces y palabras claves, direcciones, ubicación y frecuencia de apariciones. Constituye la forma más común de indexación de textos en la Web (esencialmente morfológico y estadístico).

- **Indexación por conceptos**.

Mediante este sistema se pueden recuperar recursos que tratan un tema dado, las palabras del documento no tienen que ser tales a las palabras de la pregunta. Existen varios procedimientos para construir bases de datos basadas en conceptos, algunas de ellas muy complejas y basadas en sofisticadas teorías lingüísticas y de Inteligencia Artificial. En otros casos, a partir de análisis estadísticos, el buscador determina qué conceptos aparecen juntos o relacionados en textos que se centran en un tema concreto.

- **Indexación por hiperenlaces**

Es una forma para indexar haciendo uso de los hipervínculos, y las relaciones que existen entre las páginas a través de ellos, puede ser interpretado como un grafo, en el que cada página constituye un nodo y los enlaces entre ellas arcos.

Modelos de Búsqueda

En la interacción buscador-usuario se pueden realizar acciones que conlleven al buscador a dar respuesta al usuario ante la búsqueda de determinada información, en estas acciones, las consultas realizadas por los usuarios se pueden realizar de diferentes maneras, en dependencia de la herramienta que se esté utilizando para la búsqueda, pero generalmente se basa en el uso de palabras claves, escritos en lenguaje natural o en una sintaxis estructurada lo suficientemente sencilla como para ser

escrita por cualquier persona y ser procesable por el buscador. La forma de realizar las búsquedas puede realzarse de dos formas, principalmente, una búsqueda simple usando palabras claves u otra búsqueda más avanzada haciendo uso de operadores.

La cadena escrita por el usuario en su búsqueda es procesada por el buscador por medio de algoritmos, éstos generan una cadena resultante o una forma interna que pueda ser utilizada en las comparaciones con los términos almacenados en la base de datos y generar resultados, direcciones URL de relevancia para el usuario, que se relacionan a los parámetros introducidos en la búsqueda.

Para realizar las búsquedas simples los buscadores realizan diferentes acciones y aplican algunos algoritmos sobre las cadenas de entrada. El procesamiento de la información puede ser dividido en cuatro etapas principales:

- El análisis léxico con el objetivo de analizar: dígitos, guiones, signos de puntuación, y letras.
- Eliminación de palabras poco relevantes (*stopword*) con el objetivo de filtrar las palabras que tiene poco valor para el contenido del documento.
    - **Ejemplo**: artículos, preposiciones, conjunciones.
- *Stemming*, proceso en el cual, a las palabras restantes del documento, se les extrae los prefijos y sufijos, obteniendo una colección de términos de las palabras analizadas.
- Seleccionar un listado de términos para determinar cuál o cuáles, serán usados como elementos de indexación.

Para realizar las búsquedas avanzadas el usuario puede además utilizar operadores para ajustar, agregar complejidad y más eficiencia en la búsqueda, con datos más acordes al contenido buscado.

Operadores utilizados para las búsquedas avanzadas:

- **Operadores booleanos.**
- **Operadores posicionales.**
- **Operadores de existencia.**
- **Operadores de truncamiento.**
- **Operadores de limite/comparación.**

Operadores booleanos

Los operadores booleanos utilizados en la recuperación de información resultan ser el operador AND (producto o intersección), el operador OR (suma o unión) y el operador NOT (resta o negación).

El operador **AND** es un operador de reducción que representa la intersección en al menos dos conjuntos de búsqueda, de forma que en los resultados sólo aparecen los elementos que estén en los dos conjuntos.

Se puede usar el operador **&** como alternativo.

EJ. Béisbol **AND** CUBA: buscará y devolverá información de béisbol que tenga relación con CUBA.

El operador **NOT** es un operador de reducción, excluye los elementos de uno de los dos conjuntos de la búsqueda, apareciendo sólo aquellos que no aparecen en el conjunto indicado con el operador **NOT**.

Se puede usar el operador ! como alternativo.

Ej. Béisbol **AND NOT** CUBA: buscará y devolverá información de béisbol pero no relacionada con CUBA.

El operador **OR** es un operador de ampliación, representa la unión de los elementos de los dos conjuntos de la búsqueda, apareciendo resultados que contengan al menos uno de los elementos de los dos conjuntos, no así donde no estén ninguno de ellos, cuando no se especifica entre dos términos de la búsqueda ningún operador, se toma el operador **OR** por defecto.

Se puede usar el operador **|** como alternativo.

Ej. Béisbol CUBA; Béisbol **|** CUBA: buscará y devolverá información relacionada con el béisbol o sobre CUBA, independientemente de que la primera se refiera o no a la segunda.

Operadores posicionales

Los operadores posicionales sirven para establecer la posición que tendrá un término de los elementos de la búsqueda con respecto a otro término de ésta, definiendo los posicionales relativos o de proximidad y los operadores que permiten buscar los términos en un lugar específico dentro del documento, que serían los posicionales absolutos; además éstos operadores ayudan a eliminar algunas

de las limitaciones que no se podían sobrepasar con los operadores booleanos, ejemplos de estos operadores posicionales son el **SAME**, **WITH**, **NEAR**, **ADJ.**

El operador **SAME** se utiliza para localizar registros en los que se encuentran todos los términos de la búsqueda, pero no necesariamente en la misma frase. Por ejemplo, si se busca "Cuba SAME Historia ", sólo se recuperarán los registros que contengan tanto información de " Cuba " como de " Historia " pero no necesariamente sólo sobre la historia de Cuba.

El operador **WITH** se utiliza para localizar registros en los que un campo contiene una frase con todos los términos especificados. Por ejemplo, si se busca " Cuba WITH Historia", sólo se recuperarán los registros que contengan tanto " Cuba " como "Historia" en la misma frase del contexto, encontrando registros de la *"Historia de Cuba"* e archivos sobre temas específicos en la historia de Cuba.

El operador **NEAR** se utiliza para localizar registros en los que un campo contiene todos los términos de búsqueda juntos; sin embargo, el orden de los términos no tiene que coincidir con el orden en el que se introdujeron. Por ejemplo, si se busca "Cuba NEAR Historia", sólo se recuperarían los registros con los términos " Cuba " e "Historia" juntos dentro del mismo contexto.

El operador **ADJ** se utiliza para localizar registros en los que un campo contiene todos los términos de búsqueda juntos y en el orden en el que se introdujeron.

Por ejemplo, si se buscara "CUBA ADJ Historia", sólo se recuperarían los registros con los términos " CUBA " e "Historia" juntos dentro del mismo contexto y con " CUBA " delante.

Operadores de existencia

Especifican palabras que deben aparecer en el resultado. Si se añade al inicio del término el símbolo más (+) se puede lograr que en la lista de resultados aparezcan documentos que contengan el término marcado en su contenido.

Ej.: +olímpico béisbol.

Para especificar palabras que no aparezcan en el resultado, se puede añadir al inicio del término el símbolo menos (-), logrando que en la lista de resultados no aparezcan documentos que contengan el término marcado con el menos en su contenido.

Ej.: béisbol **–**Cuba

Nota: Se puede obtenerlo también a través de estos dos operadores lógicos. béisbol AND NOT Cuba.

Operadores de truncamiento

Generalmente cuando se realiza una búsqueda se hace referencia a un término, pero sería adecuado en determinado caso que se tuviese en cuenta las palabras derivadas de la palabra clave, es decir con sufijos y prefijos, en otros casos se hace necesario precisamente lo contrario. Tal es el caso de AltaVista con el operador (*) o el operador ($) en Lycos, de manera que una búsqueda de la palabra *educa* encontraría también *educación*, *educador, etc.* Por el contrario el operador (.) en Lycos permitiría la búsqueda de una palara como es el caso de *ciudad,* y solo *ciudadano*, *ciudadela*…

Operadores límite/comparación

Los operadores de límite/comparación se usan generalmente cuando se realizan búsquedas con caracteres numéricos u operaciones con fechas, estos operadores permiten realizar operaciones sobre los parámetros de búsqueda, para los criterios a tener en cuenta durante las realizaciones de éstas. Estos operadores pueden ser (< **menor**, > **mayor**, = **igual a**, <>**diferente de**, <= **menor o igual que**, >= **mayor o igual que**).

Ej. Una consulta referente a “Fecha > 19850510”, buscará los registros superiores a la 19850510.

Respuesta de los Buscadores

Las respuestas de los buscadores puede diferir en uno de otros pero generalmente la mayoría de los aspectos son comunes para todos, dentro de los cuales se pueden encontrar:

- **Cantidad de resultados** y **tiempo de respuesta** para un contenido determinado.
- **Enlace:** enlace al contenido donde se encuentra la información.
- **Más resultados:** enlace a otros resultados relacionados con la información.
- **Páginas similares:** un enlace a las páginas con contenido similar a un resultado dado.

- **Descripción o resumen de la página:** muestra un pequeño párrafo de pocas líneas con la información dentro del documento resultado, párrafo relacionado con los términos introducidos en la búsqueda.
- **El tipo de formato** que posee el documento que se ha devuelto en los resultados, que puede ser PPT, PDF, DOC, HTML, etc.
- **Tamaño** del documento en kilobytes.
- **Fecha** de última actualización.
- **Idioma**.
- **Resultado con sangría aumentada** para diferentes resultados pertenecientes a un mismo sitio o dominio.
- **Búsquedas relacionadas con:** pequeños vínculos por categorías relacionados con los parámetros de búsqueda, generalmente con cierto nivel de especialización.

Otros buscadores integran imágenes a cada uno de sus resultados, iconos para obtener un nivel de socialización con la información, pueden mostrar además la última visita que hizo el usuario a un sitio determinado, entre muchos otros criterios que sirven de ayuda al usuario y que varían de un buscador a otro.

Criterio de asignación de relevancia

Los buscadores o motores de búsqueda hacen uso de algoritmos para la recuperación de información, estos son algoritmos secretos, porque si se supiese el funcionamiento de los algoritmos se podrían "*engañar"* a los buscadores para obtener un buen posicionamiento en ellos, éstos algoritmos se encuentran en constante cambio y son una parte fundamental de estas herramientas de búsqueda, principalmente porque de éstos depende la eficacia o no de los buscadores. Los buscadores tienen en cuenta muchos aspectos, cientos de ellos, a la hora de determinar la relevancia de los archivos que han sido recuperados, variando en cada uno de los buscadores, estos pueden ser:

- La popularidad y antigüedad de los documentos.
- La cantidad de enlaces que apuntan al un sitio determinado.

- La similitud de las palabras contenidas en las <META> con respecto a las palabras de la consulta.
- La aparición de palabras de la clave en los títulos y encabezados.
- La cantidad de ocurrencia de las palabras claves dentro del documento.
- La cercanía de las palabras claves dentro del documento.

Herramientas de Desarrollo

IDEs

Aptana

Aptana es un Entorno Integrado de Desarrollo (IDE) para el desarrollo web. Incorpora características completas, sincronización, y administración de proyectos. Permite incorporar funciones mediante plugins. Soporte para las plataformas Microsoft Windows, Mac y Linux. (10)

Características: desarrollo HTML (*HyperText Markup Language o Lenguaje de Marcas Hipertextuales*), CSS (*Cascading Style Sheets u Hojas de Estilo en Cascada*), Javascript, soporte para AJAX (*Asynchronous Javascript and XML en español Asincrónico Javascript y Lenguaje de Marcas Extensibles*), incluye librerías AJAX más populares (*JQuery, Prototype, YUI, Spry, entre otras*), soporte para el desarrollo Adobe AIR y iPhone mediante plugins, desarrollo Ruby on Rails, PHP mediante plugins, protocolos de comunicación FTP.

- **Características de Edición Profesional**: editor JSON (Javascript Object Notation), protocolos de Comunicación FTPS, SFTP, soporte en Fórums, etc.
- **Ventajas:** permite comprobar la compatibilidad de las funciones con los diferentes navegadores, multiplataforma, sincronización con carpetas locales y remotas, incluye plugins para Eclipse.
- **Desventajas:** consumo de recursos.

Zend Studio

Zend Studio o Zend Development Environment es un completo entorno integrado de desarrollo para el lenguaje de programación PHP. Está escrito en Java, y está disponible para las plataformas Microsoft Windows, Mac OS X y GNU/Linux. (11)

Características

- No requiere la instalación previa de PHP ni del entorno de ejecución de Java.
- Soporte para PHP 4 y PHP 5.
- Plegado de código (comentarios, cuerpo de funciones y métodos e implementación de clases).
- Inserción automática de paréntesis y corchetes de cierre.
- Sangrado automático y otras ayudas de formato de código.
- Emparejamiento (*matching*) de paréntesis y corchetes de apertura y cierre.
- Detección de errores de sintaxis en tiempo real.
- Funciones de depuración: botón de ejecución y traza, marcadores, puntos de parada.
- Seguimiento de variables y mensajes de error del intérprete de PHP. Permite también la depuración en servidores remotos.
- Soporte para gestión de grandes proyectos de desarrollo.
- Manual de PHP integrado.
- Soporte para control de versiones usando CVS o Subversión.
- Cliente FTP integrado.
- Soporte para navegación en bases de datos y ejecución de consultas SQL.

Zend Studio fue diseñado para usarse con el lenguaje PHP; sin embargo ofrece soporte básico para otros lenguajes Web, como HTML, Javascript y XML.

Netbeans

Es un IDE gratuito, de código abierto para desarrolladores de software. Puede obtener todas las herramientas que necesite para crear aplicaciones profesionales para el escritorio, la empresa, la web y equipos móviles con el lenguaje Java, C/C++, y Ruby. Se ejecuta en varias plataformas incluyendo Windows, Linux y Mac OS X y Solaris. Provee varias características nuevas y mejoras, como funciones de edición enriquecida de Javascript, soporte para el uso del framework web Spring, y mejor integración con MySQL. Esta versión también provee mejor rendimiento, especialmente un arranque más rápido

(más de 40%), menor consumo de memoria y mejor respuesta cuando se trabajan con proyectos grandes. (12)

Lenguajes de Programación

Desde los inicios de Internet, fueron surgiendo diferentes demandas por los usuarios y se dieron soluciones mediante lenguajes estáticos. A medida que paso el tiempo, las tecnologías fueron desarrollándose y surgieron nuevos problemas a solucionar. Esto dio lugar a desarrollar lenguajes de programación para la web dinámica, que permitieran interactuar con los usuarios y utilizaran sistemas de Bases de Datos.

HTML

Desde el surgimiento de internet se han publicado sitios web gracias al lenguaje HTML. Es un lenguaje estático para el desarrollo de sitios web (acrónimo en inglés de HyperText Markup Language, en español Lenguaje de Marcas Hipertextuales).

Ventajas:

- Sencillo, que permite describir hipertexto.
- Texto presentado de forma estructurada y agradable.
- No necesita de grandes conocimientos cuando se cuenta con un editor de páginas web o WYSIWYG.
- Archivos pequeños.
- Despliegue rápido.
- Lenguaje de fácil aprendizaje.
- Lo admiten todos los exploradores.

Desventajas:

- Lenguaje estático.
- La interpretación de cada navegador puede ser diferente.
- Guarda muchas etiquetas que pueden convertirse en “basura” y dificultan la corrección.
- El diseño es más lento.
- Las etiquetas son muy limitadas.

JAVASCRIPT

Es un lenguaje interpretado, no requiere compilación, utilizado principalmente en páginas web. La mayoría de los navegadores en sus últimas versiones interpretan el código Javascript.

El código Javascript puede ser integrado dentro de las páginas web. Para evitar incompatibilidades el World Wide Web Consortium (W3C) diseñó un estándar denominado DOM (Modelo de Objetos del Documento).

Ventajas:

- Lenguaje de scripting seguro y fiable.
- Los script tienen capacidades limitadas, por razones de seguridad.
- El código Javascript se ejecuta en el cliente.

Desventajas:

- Código visible por cualquier usuario.
- El código debe descargarse completamente. (13)

AJAX

Ajax, es un término que describe un nuevo acercamiento a usar un conjunto de tecnologías existentes juntas, incluyendo las siguientes: HTML o XHTML, hojas de estilo en cascada, Javascript, el DOM (Document Object Model), XML, XSLT (Transformaciones al Lenguaje Extensible de Hojas de Estilos), y el objeto XMLHttpRequest.

Cuando se combinan éstas tecnologías en el modelo Ajax, las aplicaciones funcionan mucho más rápido, ya que las interfaces de usuario se pueden actualizar por partes sin tener que actualizar toda la página. Por ejemplo, al rellenar un formulario de una página web, con Ajax se puede actualizar la parte en la que se elige el país de residencia sin tener que actualizar todo el formulario. (14)

PHP

Es un lenguaje de programación utilizado para la creación de sitio web. PHP es un acrónimo recursivo que significa “PHP Hypertext Pre-Processor”, (inicialmente se llamó Personal Home Page).

Es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse. Para su funcionamiento necesita tener instalado Apache o IIS con las librerías de PHP. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas.

Ventajas:

- Muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta la orientación a objeto. Clases y herencia.
- Es un lenguaje multiplataforma: Linux, Windows.
- Capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server.
- Capacidad de expandir su potencial utilizando módulos.
- Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Incluye gran cantidad de funciones.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

Desventajas:

- Se necesita instalar un servidor web.
- Todo el trabajo lo realiza el servidor y no delega al cliente. Por lo que puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.
- La programación orientada a objetos es aún muy deficiente para aplicaciones grandes.
- Dificulta la organización por capas de la aplicación.

Seguridad:

PHP es un poderoso lenguaje e intérprete, ya sea incluido como parte de un servidor web en forma de módulo o ejecutado como un binario CGI separado, es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. Estas propiedades hacen que cualquier cosa que sea ejecutada en un servidor web sea insegura por naturaleza.

PHP está diseñado específicamente para ser un lenguaje más seguro para escribir programas CGI que Perl o C, y con la selección correcta de opciones de configuración en tiempos de compilación y ejecución, y siguiendo algunas prácticas correctas de programación. (13)

JAVA

Java es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria.

Ventajas

- El JDK es una herramienta libre de licencias (sin costo), creada por Sun.- Está respaldado por un gran número de proveedores.
- Existe soporte dado por Sun.
- Debido a que existen diferentes productos de Java, hay más de un proveedor de servicios.
  Sun saca al mercado cada 6 meses una nueva versión del JDK.
- Es independiente de la plataforma de desarrollo.
- Existen dentro de su librería clases gráficas como awt y swing, las cuales permiten crear objetos gráficos comunes altamente configurables y con una arquitectura independiente de la plataforma.
  Java permite a los desarrolladores aprovechar la flexibilidad de la Programación Orientada a Objetos en el diseño de sus aplicaciones.
- El conocimiento sobre tecnología Java está en alto crecimiento en el mercado.
- Se puede acceder a bases de datos fácilmente con JDBC, independientemente de la plataforma utilizada. El manejo de las bases de datos es uniforme, es decir transparente y simple.

- Existen las herramientas Crystal Reports o herramientas libres como Texto que los genera en formato PDF. La API que utilizan estas herramientas en Java, es la más recomendable para generar reportes en Web.
- Simple: Elimina la complejidad de los lenguajes como "C" y da paso al contexto de los lenguajes modernos orientados a objetos. Orientado a Objetos. La filosofía de programación orientada a objetos es diferente a la programación convencional.
- Familiar: Como la mayoría de los programadores están acostumbrados a programar en C o en C++, la sintaxis de Java es muy similar al de éstos.
- Robusto: El sistema de Java maneja la memoria de la computadora. No se tiene que preocupar por apuntadores, memoria que no se esté utilizando, etc.
- Seguro: El sistema de Java tiene ciertas políticas que evitan se puedan codificar virus con este lenguaje. Existen muchas restricciones, especialmente para los applets, que limitan lo que se puede y no puede hacer con los recursos críticos de una computadora.
- Portable: Como el código compilado de Java (conocido como byte code) es interpretado, un programa compilado de Java puede ser utilizado por cualquier computadora que tenga implementado el interprete de Java.
- Independiente a la arquitectura: Al compilar un programa en Java, el código resultante un tipo de código binario conocido como byte code. Este código es interpretado por diferentes computadoras de igual manera, solamente hay que implementar un intérprete para cada plataforma. De esa manera Java logra ser un lenguaje que no depende de una arquitectura computacional definida.
- Multithreaded (Multihilos): Un lenguaje que soporta múltiples threads (hilos) es un lenguaje que puede ejecutar diferentes líneas de código al mismo tiempo.
- Interpretado: Java corre en máquina virtual, por lo tanto es interpretado.
- Dinámico: Java no requiere que se compilen todas las clases de un programa para que funcione. Si se realiza una modificación a una clase Java se encarga de realizar un Dynamic Bynding o un Dynamic Loading para encontrar las clases.

Desventajas

- Hay diferentes tipos de soporte técnico para la misma herramienta, por lo que el análisis de la mejor opción se dificulta.
- Para manejo a bajo nivel deben usarse métodos nativos, lo que limita la portabilidad. El diseño de interfaces gráficas con awt y swing no son simples.
- Existen herramientas como el JBuilder que permiten generar interfaces gráficas de manera sencilla, pero tienen un costo adicional.
- Puede ser que no haya JDBC para bases de datos poco comerciales.
- Algunas herramientas tienen un costo adicional.

JSP

Es un lenguaje para la creación de sitios web dinámicos, acrónimo de Java Server Pages. Está orientado a desarrollar páginas web en Java. JSP es un lenguaje multiplataforma. Creado para ejecutarse del lado del servidor. Fue desarrollado para la creación de aplicaciones web potentes. Posee un motor de páginas basado en los servlets de Java. Para su funcionamiento se necesita tener instalado un servidor Tomcat.

Características:

- Código separado de la lógica del programa.
- Las páginas son compiladas en la primera petición.
- Permite separar la parte dinámica de la estática en las páginas web.
- Los archivos se encuentran con la extensión (jsp).
- El código JSP puede ser incrustado en código HTML.

Ventajas:

- Crear páginas del lado del servidor.
- Multiplataforma.
- Código bien estructurado.
- Integridad con los módulos de Java.
- Permite la utilización servlets.

Desventajas:

- Complejidad de aprendizaje.

PYTHON

Es un lenguaje de programación comparado habitualmente con Perl. Los usuarios lo consideran como un lenguaje bueno para programar. Permite la creación todo tipo de programas incluyendo los sitios web.

Su código no necesita ser compilado. Es un lenguaje de programación multiplataforma, lo cual fuerza a que los programadores adopten por un estilo de programación particular:

- Programación orientada a objetos.
- Programación estructurada.
- Programación orientada a aspectos.

Ventajas:

- Libre y fuente abierta.
- Lenguaje de propósito general.
- Gran cantidad de funciones y librerías.
- Sencillo y rápido de programar.
- Multiplataforma.
- Licencia de código abierto (*Open Source*).
- Orientado a Objetos.
- Portable.

Desventajas:

- Lentitud por ser un lenguaje interpretado.

RUBY

Ruby es un lenguaje interpretado de muy alto nivel y orientado a objetos. Su sintaxis está inspirada en Phyton, Perl. Es distribuido bajo licencia de software libre (*Open Source*). Es un lenguaje dinámico para una programación orientada a objetos rápida y sencilla.

Ventajas:

- Permite desarrollar soluciones a bajo costo.
- Software libre.
- Multiplataforma. (13)

Gestores de Bases de Datos

MySQL

MySQL es un sistema de gestión de bases de datos relacional. Su diseño multihilos le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Aunque MySQL es software libre, MySQL AB distribuye una versión comercial de MySQL, que no se diferencia de la versión libre más que en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de no ser así, se vulneraría la Licencia Publica General (en inglés GPL, General Public License).

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

Características de MySQL

Las principales características de este gestor de bases de datos son las siguientes:

1. Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilos.
2. Soporta gran cantidad de tipos de datos para las columnas.

3. Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc.).
4. Gran portabilidad entre sistemas.
5. Soporta hasta 32 índices por tabla.
6. Permite el uso de índices de texto completo.
7. Gestión de usuarios y contraseñas (passwords), manteniendo un muy buen nivel de seguridad en los datos.

Desventajas

Aunque MySQL se incluye en el grupo de sistemas de bases de datos relacionales, carece de algunas de sus principales características:

1. Subconsultas: tal vez ésta sea una de las características que más se echan en falta, aunque gran parte de las veces que se necesitan, es posible reescribirlas de manera que no sean necesarias.
2. SELECT INTO TABLE: Esta característica propia de Oracle, todavía no está implementada.
3. Triggers y Procedures: Se tiene pensado incluir el uso de procedimientos de almacenados en la base de datos, pero no el de disparadores, ya que los disparadores reducen de forma significativa el rendimiento de la base de datos, incluso en aquellas consultas que no los activan.
4. Transacciones: a partir de las últimas versiones hay soporte para transacciones, aunque no por defecto (se ha de activar un modo especial).
5. Integridad referencial: aunque admite la declaración de claves ajenas en la creación de tablas, internamente no las trata de forma diferente al resto de los campos.

PostgreSQL

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales, es ampliamente considerado como una de las alternativas de sistema de bases de datos de código abierto.

Ventajas:

1. Ahorros considerables en costos de operación.
2. Estabilidad y Confiabilidad Legendarias.

3. Extensible: El código fuente está disponible para todos sin costo. Si se necesitara extender o personalizar PostgreSQL de alguna manera, se pudiera hacer con un mínimo esfuerzo, sin costos adicionales. Esto es complementado por la comunidad de profesionales y entusiastas de PostgreSQL alrededor del mundo que también extienden PostgreSQL todos los días.
4. Multiplataforma: PostgreSQL está disponible en casi cualquier Unix (34 plataformas en la última versión estable), y ahora en versión nativa para Windows.
5. Diseñado para ambientes de alto volumen PostgreSQL usa una estrategia de almacenamiento de filas llamada MVCC para conseguir una mejor respuesta en ambientes de grandes volúmenes. Los principales proveedores de sistemas de bases de datos comerciales usan también esta tecnología, por las mismas razones.
6. Alta concurrencia: PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. (15)

Después de haber hecho un estudio de los lenguajes y editores que se utilizarán para desarrollar una aplicación web, se ha decidido usar como herramientas las siguientes:

Para la interfaz de la aplicación: Aptana como editor, javascript como lenguaje, ajax como técnica de desarrollo web, para mantener una comunicación asíncrona con el servidor.

Para la capa de manejo de la información: Zend Studio como editor, php como lenguaje de programación.

Como gestor de base de datos: MySQL debido a que se ajusta más a los requerimientos de la aplicación, rapidez y recuperación de información eficiente, rapidez en la ejecución de consultas, hace uso de búsquedas sobre índices de texto completo, una técnica apropiada para la recuperación de información relevante.

Metodología

RUP

El **Proceso Unificado de Racional** (*Rational Unified Process o **RUP**)* es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (**UML**), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Principales características

- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
- Pretende implementar las mejores prácticas en Ingeniería de Software.
- Desarrollo iterativo.
- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.
- Modelado visual del software.
- Verificación de la calidad del software.

RUP se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles) y roles (papel que desempeña una persona en el proyecto).

Fases de RUP

- **Inicio**. Define el alcance del proyecto. Las iteraciones hacen mayor énfasis en actividades de modelado del negocio y de requerimientos.
- **Elaboración**. En la fase de elaboración, las iteraciones se orientan al desarrollo de la línea base de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios análisis.
- **Construcción**. En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones.
- **Transición**. En la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega.

Extreme Programming

La Programación Extrema es una metodología ligera de desarrollo de software que se basa en la simplicidad, la comunicación y la realimentación o reutilización del código desarrollado.

«Todo en el software cambia. Los requisitos cambian. El diseño cambia. El negocio cambia. La tecnología cambia. El equipo cambia. Los miembros del equipo cambian. El problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder; el problema es la incapacidad de adaptarnos a dicho cambio cuando éste tiene lugar.» Kent Beck.

XP surgió como respuesta y posible solución a los problemas derivados del cambio en los requerimientos. XP se plantea como una metodología a emplear en proyectos de riesgo. Sirve para aumentar la productividad. (16)

Las cuatro variables

- **Coste**: Máquinas, especialistas y oficinas.
- **Tiempo**: Total y de Entregas.
- **Calidad**: Externa e Interna.
- **Alcance**: Intervención del cliente.

Principios

Comunicación

El eXtreme Programming se nutre del ancho de banda más grande que se puede obtener cuando existe algún tipo de comunicación: la comunicación directa entre personas. Es muy importante entender cuales son las ventajas de este medio. Cuando dos (o más) personas se comunican directamente pueden no solo consumir las palabras formuladas por la otra persona, sino que también aprecian los gestos, miradas, etc. que hace su compañero. Sin embargo, en una conversación mediante el correo electrónico, hay muchos factores que hacen de esta una comunicación, por así decirlo, mucho menos efectiva.

Coraje

El coraje es un valor muy importante dentro de la programación extrema. Un miembro de un equipo de desarrollo extremo debe de tener el coraje de exponer sus dudas, miedos, experiencias sin "embellecer" éstas de ninguna de las maneras. Esto es muy importante ya que un equipo de desarrollo extremo se basa en la confianza para con sus miembros. Faltar a esta confianza es una falta más que grave.

Simplicidad

Dado que no se puede predecir cómo va a ser en el futuro el software que se está desarrollando; un equipo de programación extrema intenta mantener el software lo más sencillo posible. Esto quiere decir que no se va a invertir ningún esfuerzo en hacer un desarrollo que en un futuro pueda llegar a tener valor. En el XP frases como "...en un futuro vamos a necesitar..." o "Haz un sistema genérico de..." no tienen ningún sentido ya que no aportan ningún valor en el momento.

Retroalimentación (Feedback)

La agilidad se define (entre otras cosas) por la capacidad de respuesta ante los cambios que se van haciendo necesarios a lo largo del camino. Por este motivo uno de los valore que nos hace más ágiles es el continuo seguimiento o feedback que recibimos a la hora de desarrollar en un entorno ágil de desarrollo. Este feedback se toma del cliente, de los miembros del equipo, en cuestión de todo el entorno en el que se mueve un equipo de desarrollo ágil.

Ciclos de Vida

El ciclo de vida ideal de XP:

Exploración

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología.

Planificación de la Entrega (Release)

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días. Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como medida el punto. Un punto, equivale a una semana ideal de programación. Las historias generalmente valen de 1 a 3 puntos. Por otra parte, el equipo de desarrollo mantiene un registro de la "velocidad" de desarrollo, establecida en puntos por iteración, basándose principalmente en la suma de puntos correspondientes a las historias de usuario que fueron terminadas en la última iteración. La planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación.

Iteraciones

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El Plan de Entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias que fuercen la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide qué historias se implementarán en cada iteración (para maximizar el valor de negocio). Al final de la última iteración el sistema estará listo para entrar en producción. Los elementos que deben tomarse en cuenta durante la elaboración del Plan de la Iteración son: historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Todo el trabajo de la iteración es expresado en tareas de programación, cada una de ellas es asignada a un programador como responsable, pero llevadas a cabo por parejas de programadores.

Producción

La fase de producción requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente. Al mismo tiempo, se deben tomar decisiones sobre la

inclusión de nuevas características a la versión actual, debido a cambios durante esta fase. Es posible que se rebaje el tiempo que toma cada iteración, de tres a una semana. Las ideas que han sido propuestas y las sugerencias son documentadas para su posterior implementación (por ejemplo, durante la fase de mantenimiento).

Mantenimiento

Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Para realizar esto se requiere de tareas de soporte para el cliente. De esta forma, la velocidad de desarrollo puede bajar después de la puesta del sistema en producción. La fase de mantenimiento puede requerir nuevo personal dentro del equipo y cambios en su estructura.

Muerte del Proyecto

Es cuando el cliente no tiene más historias para ser incluidas en el sistema. Esto requiere que se satisfagan las necesidades del cliente en otros aspectos como rendimiento y confiabilidad del sistema. Se genera la documentación final del sistema y no se realizan más cambios en la arquitectura. La muerte del proyecto también ocurre cuando el sistema no genera los beneficios esperados por el cliente o cuando no hay presupuesto para mantenerlo. (17)

Metodología a utilizar

E**x**treme **P**rogramming (XP) fue creada en respuesta a problemas donde dominaban los cambios en los requerimientos, donde los usuarios pueden tener poca idea de de lo que hará el sistema. XP es especial para sistemas donde se espera que las funcionalidades cambien cada cierto tiempo, donde el personal con que se cuenta es pequeño, con pocos programadores se puede desarrollar software con requerimientos cambiantes y alto riesgo, porque estos programadores trabajarán juntos codo con codo, codificando, haciendo preguntas y cada uno de ellos estará envuelto en gran medida en el desarrollo del software.

Conclusiones

Con la realización de esta investigación se da cumplimiento a los objetivos propuestos, pues se lograron resultados relacionados con la selección de las herramientas y lenguajes de programación a utilizar en el desarrollo de la aplicación así como la metodología a utilizar para el desarrollo del proyecto, se logró

profundizar en los algoritmos que actualmente se utilizan para desarrollar motores de búsqueda, se realizó un resumen de las principales características, ventajas y desventajas de un grupo selecto de buscadores en internet y otros que prestan servicios en la Universidad de las Ciencias Informáticas, llegando a la conclusión de que en nuestra universidad es de vital importancia la existencia de una aplicación web que brinde servicios de búsqueda, que sirva de forma eficiente para ayudar al usuario a encontrar la información más relevante dentro de la red universitaria.

Capítulo 2. Características del Sistema

Introducción

En este apartado se pondrán al relieve las principales características del sistema, se hará una descripción de cada una de sus funcionalidades, así como las necesidades de los usuarios, teniendo siempre en cuenta como principal objetivo, la problemática de este trabajo. Se especificarán además los requisitos funcionales y no funcionales del sistema.

Descripción

Las búsquedas de información web en la Universidad de las Ciencias Informáticas no se lleva a cabo como se esperaba que se hiciera, principalmente por la inexistencia de una herramienta bien desarrollada que permita almacenar y organizar la información para luego mostrarla al usuario de forma agradable, y que esta información sea la más relevante posible, por esta razón es que se ha propuesto desarrollar un sistema para automatizar la búsqueda de información web y dar por resuelto la problemática por la que este trabajo tuvo que desarrollarse.

Situación Actual

La búsqueda de información web disponible en la red de la UCI es un proceso no automatizado. Cuando se hace necesario algún tipo de información, la búsqueda se realiza haciendo uso de los buscadores tradicionales de internet, que pertenecen a otros países, buscadores que permiten buscar una u otra vez gran cantidad de información; pero por ser precisamente internacionales, es imposible acceder a la información disponible en nuestras redes, lo que provoca que se descarguen datos que una vez pudieron haberse descargado o generado en el centro, que fueron estructurados, y pudieran ser usados en determinado momento; el poseer esta información accesible en la red, permitiría no tener que usar los buscadores de internet y volver a realizar procesos con esos datos que probablemente ya se habían hecho con anterioridad.

Propuesta del sistema

El sistema que se desarrolla para dar solución a la situación problémica de este trabajo, propone determinadas funcionalidades que permitirán a los usuarios de la UCI acceder a recursos web que estén disponibles en la red. El sistema permitirá realizar la Búsqueda de Información Web, tanto páginas web como imágenes; así como influir en el sistema de posicionamiento de la aplicación. Dada una consulta realizada por el usuario, la aplicación le permitirá: mostrar los resultados, cargar los resultados desde una caché, hacer sugerencias sobre algún resultado, proponer nuevos resultados

relacionados con la consulta realizada, acciones que permitirán recopilar información que puede ser de importancia para mejorar el funcionamiento del sistema, así como para futuros estudios de los webmasters.

El sistema será una aplicación web desarrollada con un lenguaje multiplataforma, y cada uno de sus módulos se desarrollará con un objetivo específico, para conformar entre todos lo que se conocerá como “Sistema para la Recuperación de la Información Web”.

Personas relacionadas con el sistema

Las personas relacionadas con el sistema serán aquellas que se relacionen con cualquiera de los procesos que lleva a cabo el sistema y obtienen un resultado de valor con el mismo.

Tabla 1.0 HU Personas relacionadas con el sistema.

Personas relacionadas con el sistema	Justificación
Administrador	Es la persona con facultad para acceder a la información disponible en las bases de datos de la aplicación, para llevar un control sobre la misma.
Usuario	Es la persona que interactúa con la aplicación web para hacer consultas sobre algún tema.

Requisitos funcionales del sistema

Un requisito funcional es una capacidad o condición que debe cumplir un sistema.

Los requisitos funcionales serán las acciones que realizará el usuario haciendo uso del buscador, así como las acciones que realizará el sistema y que no serán producto de la acción directa de los usuarios. El sistema debe ser capaz de:

R1: Gestión de búsqueda Web

R1.1 Mostrar el formulario de búsqueda de información web.

R1.2 Validar los datos de la consulta introducidos por el usuario.

R1.3 Mostrar los resultados de la búsqueda realizada por el usuario.

R1.4 Permitir cargar desde caché la información de los resultados devueltos en la búsqueda.

R1.5 Permitir hacer sugerencia a favor o en contra de algún resultado en específico de la consulta realizada.

R1.4 Permitir sugerir otros resultados relacionados con la consulta realizada por el usuario.

R2: Gestión de búsquedas de Imágenes.

R2.1 Mostrar el formulario de búsqueda de imágenes.

R2.2 Validar los datos de la consulta introducidos por el usuario.

R2.3 Mostrar los resultados de la búsqueda realizada por el usuario.

Requisitos no funcionales

Son las características o propiedades que debe tener el sistema, características que hacen del producto un software atractivo, usable, rápido o confiable y seguro.

Diseño e implementación:

- Aplicación Web escrita sobre el lenguaje de programación PHP 5.2.3.
- Usar el gestor de Base de Datos MySQL 5.0.45.
- Utilizar como servidor web Apache 2.4.

Apariencia o interfaz externa:

- Diseño sencillo y amigable para el usuario.
- Diseño ajustable a la resolución del monitor.
- Interfaz compatible con los diferentes navegadores web existentes.

Usabilidad:

- El sistema será lo suficientemente sencillo como para ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora y de un ambiente Web en sentido general.

Funcionalidad:

- Agrupación de resultados en grupos de 10 para búsquedas web y de 30 los resultados de las búsquedas de imágenes.

Seguridad:

- Proteger la información que consulta el sistema de accesos no autorizados.
- Validar los intentos de “SQL Inyection”, introducción de código SQL que dañe la estructura de la Base de Datos.
- Establecer tiempos de espera entre una consulta y otra de no menos de 30 segundos.

Portabilidad:

- El sistema será implementado usando lenguajes de programación que le permitan a la aplicación ejecutarse en cualquier plataforma: Windows, Linux…

Exploración y Planificación.

Fase de Exploración:

La primera de las fases de la metodología e**X**treme **P**rogramming es la de exploración, en esta fase, los clientes plantean las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. (18)

El cliente procede a escribir las historias, una vez escritas las historias, los desarrolladores proceden a estimar cuanto les tomaría implementarlas en Tiempo Ideal de Ingeniería (Ideal Engineering Time, el tiempo que tomaría implementar una historia sin interrupciones, reuniones u otras tareas). Los encargados del negocio podrían además dividir las historias, creando nuevas historias de la unión de

unas o la división de otra en partes para crear nuevas historias, en dependencia de la estimación calculada por los desarrolladores.

Historias de Usuario

Las historias de usuario son una representación de un requerimiento de software escrito en el lenguaje común del usuario, son una forma rápida de administrar los requerimientos de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos.

Las historias de usuario se pueden representar de la siguiente manera:

<table>
<tr><th colspan="3">Historia de Usuario</th></tr>
<tr><td>Número: Nro. de la Historia de Usuario</td><td colspan="2">Nombre: Nombre de la Historia de Usuario</td></tr>
<tr><td colspan="3">Usuario: Usuario del sistema que utiliza la historia.</td></tr>
<tr><td>Prioridad en Negocio: Prioridad o importancia de la HU para el cliente.</td><td colspan="2">Riesgo en Desarrollo: La complejidad que supondría para el desarrollador implementar la HU.</td></tr>
<tr><td>Puntos de Estimación: Tiempo que le tomaría al desarrollar implementar la HU.</td><td colspan="2">Iteración Asignada: A la liberación en la que se corresponde la HU.</td></tr>
<tr><td colspan="3">Descripción:
Breve narración del cliente, en lenguaje natural, de la operación que se supone puedan realizar los usuarios en el sistema.</td></tr>
<tr><td colspan="3">Observaciones: Algún que otro punto de interés, sobre los usuarios, observaciones, etc.</td></tr>
</table>

Las historias de usuario permiten responder rápidamente a los requerimientos cambiantes. Las historias de usuario documentadas para el desarrollo de este proyecto son:

Tabla 2.0 HU Gestionar Búsqueda de Recurso Web.

Historia de Usuario	
Número:1	**Nombre:** Gestionar Búsqueda de Páginas Web
Usuario: Usuario	
Prioridad en Negocio: Alta	**Riesgo en Desarrollo:** Bajo
Puntos de Estimación: 3	**Iteración Asignada:** 1
Descripción: Se realizan las acciones de búsqueda de páginas web. El usuario accede al sistema y se le brinda la posibilidad de especificar algún criterio de búsqueda para realizar la consulta. Se brinda la posibilidad de influir sobre el valor de relevancia de los resultados devueltos, haciendo uso de las funcionalidades: búsqueda de páginas similares, sugerir resultado, cargar desde la caché, e Ignorar resultado.	
Observaciones:	

Tabla 2.1 HU Gestionar Búsqueda de Imágenes.

Historia de Usuario	
Número:2	**Nombre:** Gestionar Búsqueda de Imágenes
Usuario: Usuario	
Prioridad en Negocio: Alta	**Riesgo en Desarrollo:** Bajo
Puntos de Estimación: 2	**Iteración Asignada:** 2
Descripción: Se realizan las acciones de búsqueda de imágenes. El sistema muestra dado un criterio de búsqueda especificado por el usuario, un resumen de las imágenes que más se relacionen con los términos	

especificados en la consulta.
Observaciones:

Tarea de Ingeniería	
Número de Tarea:1	**Historia de Usuario:** Gestionar Búsqueda de Páginas Web
Nombre de Tarea: Sugerir Búsqueda	
Tipo de Tarea: Desarrollo	**Puntos estimados:**1
Fecha Inicio:02/03/09	**Fecha Fin:**09/03/09
Programador Responsable: Dionnis Osorio Lores	
Descripción: Permitir al usuario sugerir una dirección que se ajuste más a los resultados asociados a los términos de la consulta realizada.	

Tarea de Ingeniería	
Número de Tarea:2	**Historia de Usuario:** Gestionar Búsqueda de Páginas Web
Nombre de Tarea: Ignorar Resultado	
Tipo de Tarea: Desarrollo	**Puntos estimados:**1
Fecha Inicio:10/03/09	**Fecha Fin:**17/03/09

Programador Responsable: Anniel Rodríguez Izquierdo
Descripción: Permitir al usuario ignorar un resultado determinado de los resultados devueltos de la búsqueda realizada.

Tarea de Ingeniería	
Número de Tarea:3	**Historia de Usuario:** Gestionar Búsqueda de Páginas Web
Nombre de Tarea: En Caché	
Tipo de Tarea: Desarrollo	**Puntos estimados:**1
Fecha Inicio:18/03/09	**Fecha Fin:**25/03/09
Programador Responsable: Anniel Rodríguez Izquierdo	
Descripción: Permitir al usuario cargar un resultado del repositorio del sistema.	

Tarea de Ingeniería	
Número de Tarea:4	**Historia de Usuario:** Gestionar Búsqueda de Páginas Web
Nombre de Tarea: Búsqueda de Páginas Similares	
Tipo de Tarea: Desarrollo	**Puntos estimados:**1

Fecha Inicio:26/03/09	**Fecha Fin:**3/04/09
Programador Responsable: Dionnis Osorio Lores	
Descripción: Permitir al usuario acceder a las direcciones similares a una dirección perteneciente a los resultados obtenidos en la búsqueda realizada.	

Tarea General de Ingeniería

Los buscadores o motores de búsqueda se componen de tres partes fundamentales, la Base de Datos documental, el Motor de búsqueda de Información y un Spider o araña (Capítulo 1), por lo que se hizo necesario para la complementación del Sistema de Recuperación que se describe en este capítulo, la implementación de un Spider (Araña) para la indexación de las páginas del dominio uci, y que serán las que se almacenarán en la Base de Datos del sistema, desde donde se mostrarán los enlaces en respuesta a las consultas de búsquedas realizadas por los usuarios de la universidad.

El Spider es un programa que inspecciona las páginas de internet de forma metódica y automatizada, se utilizan para crear una copia de todas las páginas web visitadas para su procesado posterior por un motor de búsqueda que indexa las páginas proporcionando un sistema de búsquedas rápido. Éstos comienzan visitando una lista de URLs, identifica los hiperenlaces en dichas páginas y los añade a la lista de URLs a visitar de manera recurrente de acuerdo a determinado conjunto de reglas. La operación normal es que se le da al programa un grupo de direcciones iniciales, el spider descarga las direcciones, analiza las páginas y busca enlaces a páginas nuevas, luego descarga las páginas nuevas, analiza sus enlaces, y realiza el mismo proceso una y otra vez en cada una de las nuevas páginas descargadas.

Fase de Planificación

En esta fase el cliente establece la prioridad de cada historia de usuario, y respectivamente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el

cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días. Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como medida el punto. Un punto, equivale a una semana ideal de programación. Las historias generalmente valen de 1 a 3 puntos. Por otra parte, el equipo de desarrollo mantiene un registro de la "velocidad" de desarrollo, establecida en puntos por iteración, basándose principalmente en la suma de puntos correspondientes a las historias de usuario que fueron terminadas en la última iteración. La planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuantas historias se pueden implementar antes de una fecha determinada o cuanto tiempo tomará implementar un conjunto de historias. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuantos puntos se pueden completar. Al planificar según alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación. (18)

Estimación de fuerza por historias

Historia de Usuario	Punto de Estimación
Gestionar Búsqueda de Páginas Web	3
Sugerir Búsqueda	1
Ignorar Resultado	1
En Caché	1
Páginas Similares	1
Gestionar Búsqueda de Imágenes	2

Iteraciones

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El Plan de Entrega está compuesto por iteraciones de no más de 6 semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias que fuercen la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide que historias se implementarán en cada iteración (para maximizar el valor de negocio). Al final de la última iteración el sistema estará listo para entrar en producción. Los elementos que deben tomarse en cuenta durante la elaboración del Plan de la Iteración son: historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Todo el trabajo de la iteración es expresado en tareas de programación, cada una de ellas es asignada a un programador como responsable, pero llevadas a cabo por parejas de programadores. (18)

Iteración 1:

Esta iteración tiene como objetivo la implementación del módulo de búsquedas de páginas web, módulo que se centra en la funcionalidad descrita en la historia de usuario “Gestionar Búsqueda de Páginas Web” la cual hace alusión al tipo de búsqueda más referenciada por los usuarios. Además en esta iteración se realizará la implementación de las Tareas de Ingeniería que complementan la terminación de la historia de usuario Gestión de Búsqueda de Páginas Web, que serían las funcionalidades Ignorar Resultado, Sugerir Búsqueda**,** Búsqueda de páginas similares y buscar en Caché.

Iteración 2:

Esta iteración tiene como objetivo la implementación del módulo de búsquedas de imágenes, módulo que se centra en la funcionalidad descrita en la historias de usuario “Gestionar Búsqueda de Imágenes”.

Plan de duración de las iteraciones:

Como parte del ciclo de vida de un proyecto usando la Metodología XP se crea el plan de duración de cada una de las iteraciones. Este plan tiene como objetivo mostrar la duración de cada iteración, así como el orden en que serán implementadas las historias de usuario en cada una de estas iteraciones.

Tabla: Plan de duración de las iteraciones

Iteraciones	Orden de las historias de usuario a implementar	Duración total de las iteraciones
Iteración 1	Gestionar Búsqueda de Páginas Web	3 semanas
	Sugerir Búsqueda	1 semana
	Ignorar Resultado	1 semana
	En Caché	1 semana
	Páginas Similares	1 semana
Iteración 2	Gestionar Búsqueda de Imágenes	2 semanas

Plan de entregas

El plan de entregas, elaborado para la fase de implementación, fue concebido para definir en que fechas se harán los *releases* del sistema.

Tabla: Módulos y HU abarcadas

Módulos	**Historias de usuario que abarca**
Web	Gestionar Búsqueda de Páginas Web
Imágenes	Gestionar Búsqueda de Imágenes

Tabla: Plan de duración entrega

Módulos	**Final 1era iteración** 3 semanas de Febrero	**Final 2da iteración** 2 semanas de Marzo/Abril
Web	1.0	Finalizado

Imágenes		1.0

Conclusiones

En este capítulo se ha hecho una descripción de cada uno de los artefactos generados durante las fases de exploración y planificación en el desarrollo del sistema, se han presentado y descrito cada una de las funcionalidades con que contará el sistema, con las que podrá interactuar el usuario, se tuvieron en cuenta aspectos importantes como los requisitos funcionales y no funcionales, las tareas realizadas en cada iteración del ciclo de vida del software, definiéndole a cada tarea su prioridad, su riesgo; y se definieron fechas de entrega de cada de las liberaciones del producto. Ésta fase del proceso de desarrollo permitió definir y hacer un estimado razonable de tiempo y recursos para llevar hacia una nueva fase la realización del Sistema de Recuperación de Información.

Capítulo 3: Diseño, Codificación y Pruebas

Introducción

La metodología XP propone la idea, a la hora de diseñar, “The simplest thing that could possibly work”, “Lo más simple que pueda funcionar”, aplicándose principalmente en los valores, en que: un diseño bien simple es más fácil para la *comunicación* que un diseño complejo, para lograr la *simplicidad* se debe tener una estrategia que produzca diseños simples, pero la estrategia en sí debe ser también simple. A la hora de retroalimentar (feedback), el diseño simple ayuda a solucionar cualquier problema que pueda presentarse, cuando se tiene un diseño simple se puede prestar atención a codificar. El coraje está en que con un diseño simple siempre se puede detener para agregar más de diseño cuando se haga necesario.

Basados en estos valores, XP plantea que se debería:

- Crear una estrategia de diseño que produzca diseños simples.
- Hallar una forma rápida de encontrar su calidad.
- Retroalimentar lo que se aprendió en el diseño.
- Nivelar el ciclo de vida de todo el proceso lo más corto posible.

En la fase de Diseño:

Buscar la simplicidad.

- Un diseño simple es más fácil de crear y mantener. Buscar afanosamente el diseño más simple que funcione. No implementar funcionalidad antes de tiempo.
- Advertencia: lograr un diseño simple es un gran trabajo.

Esquema de nombres (Metáforas para el Sistema)

- Nombrar clases y métodos en forma consistente y clara. Debe ser posible inferir la significación real de un nombre nunca visto antes.

Tarjetas CRC (Clase, Responsabilidad, Colaboración)

- Usar tarjetas CRC, en el diseño grupal. Ayudan a evitar el enfoque procedimental y destacan la orientación a objetos.

- Cada tarjeta CRC representa un objeto. El nombre de la clase va arriba, las responsabilidades (qué debe hacer) a la izquierda, las clases asistentes (que colaboran) a la derecha.
- No suele ser necesario escribir la tarjeta completa; los participantes se familiarizan rápidamente con el propósito de cada clase.
- En la reunión CRC alguien simula el sistema discutiendo los mensajes intercambiados entre objetos.

En nuestro sistema las clases presentes son:

En el módulo Gestionar Páginas Web:

- iPágina.
- iConexión.
- Página.
- ConexiónWeb.
- ConsultarPáginas.
- cc_GestionarPáginas.
- ReporteWeb.

En el módulo Gestionar Imágenes:

- iImagen.
- iConexión.
- Imagen.
- ConexiónImagenes.
- ConsultarImágenes.
- cc_GestionarImágenes.
- ReporteImágenes.

Clases Generales:

- Stemmer.
- Stopwords.

Clase: iPágina

Responsabilidades	Clases relacionadas
Mostrar identificador	
Mostrar relevancia	
Cambiar relevancia	
Mostrar tamaño de la página	
Mostrar fecha de última modificación	
Mostrar fecha de indexación	
Mostrar fecha de último acceso a la página	
Mostrar dirección en el repositorio	
Mostrar URL	
Mostrar dominio	
Mostrar título	
Mostrar palabras claves	
Mostrar descripción	
Cambiar descripción	
Mostrar contenido	
Mostrar texto de los hipervínculos	
Mostrar texto en negrita	
Mostrar texto en cursiva	
Mostrar texto header 1	
Mostrar texto header 2	
Mostrar texto header 3	
Mostrar texto header 4	
Mostrar textos ”Alternativos”	
Mostrar textos “Títulos”	
Mostrar textos Fuertes	
Mostrar texto de listas	
Mostrar visitantes	
Mostrar enlaces entrantes	
Mostrar enlaces salientes	
Mostrar profundidad de la página	
Mostrar categoría de la página	
Mostrar cantidad de campos ocultos	
Mostrar páginas similares	
Añadir página similar	

Clase: iConexión

Responsabilidades	Clases relacionadas
Conectarse a la base de datos	
Desconectarse de la base de datos	
Verificar estado de la conexión	

Clase: Página

Responsabilidades	Clases relacionadas
Mostrar identificador	
Mostrar relevancia	
Cambiar relevancia	
Mostrar tamaño de la página	
Mostrar fecha de última modificación	
Mostrar fecha de indexación	
Mostrar fecha de último acceso a la página	
Mostrar dirección en el repositorio	
Mostrar URL	
Mostrar dominio	
Mostrar título	
Mostrar palabras claves	
Mostrar descripción	
Cambiar descripción	
Mostrar contenido	
Mostrar texto de los hipervínculos	
Mostrar texto en negrita	
Mostrar texto en cursiva	
Mostrar texto header 1	
Mostrar texto header 2	
Mostrar texto header 3	
Mostrar texto header 4	
Mostrar textos ”Alternativos”	
Mostrar textos “Títulos”	
Mostrar textos Fuertes	
Mostrar texto de listas	
Mostrar visitantes	
Mostrar enlaces entrantes	
Mostrar enlaces salientes	
Mostrar profundidad de la página	
Mostrar categoría de la página	
Mostrar cantidad de campos ocultos	
Mostrar páginas similares	
Añadir página similar	

Clase: ConexiónWeb

Responsabilidades	Clases relacionadas
Conectarse a la base de datos	
Desconectarse de la base de datos	
Verificar estado de la conexión	

Clase: ConsultarPáginas

Responsabilidades	Clases relacionadas
Consultar a la base de datos	Conexión
Archivar visita del usuario	Conexión

Clase: cc_GestionarPáginas

Responsabilidades	Clases relacionadas
Buscar páginas en la base de datos	ConsultarPáginas
Buscar páginas en la caché del servidor	
Optimizar la consulta del usuario	
Posicionar los resultados devueltos	
Generar descripción para los resultados	
Ordenar los resultados por relevancia	
Agrupar páginas similares	
Contar visita del usuario	ConsultarPáginas

Clase: ReporteWeb

Responsabilidades	Clases relacionadas
Generar reporte de páginas web	

Clase: iImagen

Responsabilidades	Clases relacionadas
Mostrar identificador	
Mostrar URL de la imagen	
Mostrar URL de la página donde está la imagen	
Mostrar nombre de la imagen	
Mostrar descripción de la imagen	
Mostrar ancho de la imagen	
Mostrar alto de la imagen	
Mostrar peso de la imagen	
Mostrar la relevancia de la imagen	
Cambiar la relevancia de la imagen	

Clase: Imagen

Responsabilidades	Clases relacionadas

Mostrar identificador	
Mostrar URL de la imagen	
Mostrar URL de la página donde está la imagen	
Mostrar nombre de la imagen	
Mostrar descripción de la imagen	
Mostrar ancho de la imagen	
Mostrar alto de la imagen	
Mostrar peso de la imagen	
Mostrar la relevancia de la imagen	
Cambiar la relevancia de la imagen	

Clase: ConexiónImagenes

Responsabilidades	Clases relacionadas
Conectarse a la base de datos	
Desconectarse de la base de datos	
Verificar estado de la conexión	

Clase: ConsultarImágenes

Responsabilidades	Clases relacionadas
Consultar base de datos	Conexión

Clase: cc_GestionarImágenes

Responsabilidades	Clases relacionadas
Buscar imágenes en la base de datos	ConsultarImágenes
Expandir la consulta del usuario	
Calcular la relevancia de los resultados obtenidos	
Ordenar los resultados dado su relevancia	

Clase: ReporteImágenes

Responsabilidades	Clases relacionadas
Generar reporte de imágenes	

Clase: Stemmer

Responsabilidades	Clases relacionadas
Eliminar pluralidad	
Eliminar acentos	
Eliminar pronombres, gerundios, participios,	

superlativos	
Eliminar terminaciones	

Clase: StopWords

Responsabilidades	Clases relacionadas
Filtrar palabras vacías	

En la fase de Codificación:

Refactorizar.

- Refactorizar es mejorar el código existente, la estructura interna del software, no su comportamiento visible.
- Refactorizar para mantener el diseño simple: quitar redundancia, eliminar funcionalidad no usada, rehacer diseños obsoletos, mantener el código limpio y conciso para que sea fácil de entender, modificar y extender.

Normas de codificación**.**

- Debe elegirse y respetarse una norma de codificación.

Codificar primero la prueba de unidad.

- Las pruebas de unidad se escriben una vez y se corren reiteradamente a lo largo de todo el proyecto, asegurando siempre el funcionamiento correcto; evitan las ambigüedades, los requerimientos quedan afinados en la prueba.
- Para cada unidad, se codifica primero una prueba simple para una función simple; se van agregando pruebas y funcionalidades en etapas sucesivas, hasta implementar todo y probar todo en esa unidad.
- Una funcionalidad está terminada cuando pasa todas sus pruebas de unidad.

Pruebas de unidad.

- Crear u obtener un marco de prueba ("test framework") para crear una suite automática de pruebas de unidad. Probar todas las clases del sistema (métodos triviales de "set" y "get" suelen omitirse).
- Crear las pruebas antes de escribir el código. No puede integrarse código sin sus pruebas de unidad.

- Las pruebas de unidad evolucionan junto con el código. No pueden crearse al final, ni dejar de escribirse.
- El tiempo de escribir las pruebas de unidad se gana con creces en la reiteración continua de las pruebas y la confianza al encarar cambios.
- Las pruebas de unidad posibilitan la propiedad colectiva de código, la refactorización, la integración frecuente. El agregado de funcionalidad incluye el agregado de pruebas.

Pruebas de Unidad para el módulo Gestionar Búsqueda de Páginas Web

Tabla: Prueba de Unidad para "ConsultarPáginas"

Historia de usuario:	Gestionar Búsqueda de Páginas Web
Componente:	ConsultarPáginas
Funcionalidad:	Consultar
Objetivos:	Verificar que la información devuelta por el método es correcta
Descripción:	Para realizar la prueba, se aplicó el principio one-zero-many para asegurar tres pruebas fundamentales: una búsqueda que regresa exactamente un resultado, una búsqueda que regresa varios resultados y una búsqueda que no regresa ningún resultado.
Resultados esperados:	En caso de haber ocurrido algún error en el proceso de búsqueda, el método devuelve una lista vacía, en caso contrario devuelve un listado con la información de una, ninguna o varias páginas web.
Observaciones:	

Tabla: Prueba de Unidad para "GestionarDesdeCaché"

Historia de usuario:	Gestionar Búsqueda de Páginas Web
Componente:	cc_GestionarPáginas
Funcionalidad:	GestionarDesdeCaché
Objetivos:	Cargar el código HTML de una página web almacenada en el repositorio de la aplicación.
Descripción:	Para realizar la prueba, se especificó el identificador de la página que se desea cargar para recuperar su contenido.
Resultados esperados:	En caso de haber ocurrido algún error en el proceso de búsqueda el método devuelve un mensaje de error y en caso contrario devuelve el código HTML de la página especificada.
Observaciones:	

Tabla: Prueba de Unidad para "Posicionar"

Historia de usuario:	Gestionar Búsqueda de Páginas Web
Componente:	cc_GestionarPáginas
Funcionalidad:	Posicionar
Objetivos:	Asignar a cada página un nivel de relevancia haciendo

	comparaciones entre los términos de búsqueda especificados por el usuario y los términos que forman parte del contenido de la página.
Descripción:	Para realizar la prueba, se utilizaron los términos de búsqueda especificados por el usuario y se buscaron ocurrencias de éstos en diferentes partes del contenido de la página web, se le asignó una puntuación a cada término y se calculó la suma total de la puntuación de todos los términos.
Resultados esperados:	A cada página web se le asignó un valor de relevancia, dada por la puntuación calculada.
Observaciones:	

Tabla: Prueba de Unidad para "Agrupar_Páginas_Similares"

Historia de usuario:	Gestionar Búsqueda de Páginas Web
Componente:	cc_GestionarPáginas
Funcionalidad:	Agrupar_Páginas_Similares
Objetivos:	Agrupar páginas que comparten cierto grado de similitud en su contenido.
Descripción:	Para realizar la prueba, se utilizó un listado con la información de las páginas web para agruparlas dada su similitud en contenido.
Resultados esperados:	Se devolvió un listado en el cual algunas páginas contienen la información de páginas similares a ella.
Observaciones:	

Tabla: Prueba de Unidad para "GenerarReporte"

Historia de usuario:	Gestionar Búsqueda de Páginas Web
Componente:	Reporte
Funcionalidad:	GenerarReporte
Objetivos:	Representar la información de las páginas web devueltas en el proceso de búsqueda en formato XML para su presentación en la interfaz de la aplicación.
Descripción:	Para realizar la prueba, se utilizó un listado con la información de las páginas web para generar una respuesta en formato XML.
Resultados esperados:	Se devolvió un XML con la información de las páginas web.
Observaciones:	

Pruebas de Unidad para el módulo Gestionar Búsqueda de Imágenes

Tabla: Prueba de Unidad para "Consultar"

Historia de usuario:	Gestionar Búsqueda de Imágenes
Componente:	ConsultarImágenes

Funcionalidad:	Consultar
Objetivos:	Verificar que la información devuelta por el método es correcta.
Descripción:	Para realizar la prueba, se aplicó el principio one-zero-many para asegurar tres pruebas fundamentales: una búsqueda que regresa exactamente un resultado, una búsqueda que regresa varios resultados y una búsqueda que no regresa ningún resultado.
Resultados esperados:	En caso de haber ocurrido algún error en el proceso de búsqueda el método devuelve una lista vacía, en caso contrario devuelve un listado con la información de una, ninguna o varias imágenes.
Observaciones:	

Tabla: Prueba de Unidad para "Posicionar"

Historia de usuario:	Gestionar Búsqueda de Imágenes.
Componente:	cc_GestionarImágenes
Funcionalidad:	Posicionar
Objetivos:	Asignar a cada imagen un nivel de relevancia haciendo comparaciones entre los términos de búsqueda especificados por el usuario y los términos que forman parte de la descripción de la imagen.
Descripción:	Para realizar la prueba, se utilizaron los términos de búsqueda especificados por el usuario y se buscan ocurrencias de éstos en la descripción de la imagen, se le asignó una puntuación a cada término y se calculó la suma total de la puntuación de todos los términos.
Resultados esperados:	A cada imagen se le asignó un valor de relevancia, dada por la puntuación calculada.
Observaciones:	

Tabla: Prueba de Unidad para "GenerarReporte"

Historia de usuario:	Gestionar Búsqueda de Imágenes.
Componente:	Reporte
Funcionalidad:	GenerarReporte
Objetivos:	Representar la información de las imágenes devueltas en el proceso de búsqueda en formato XML para su presentación en la interfaz de la aplicación.
Descripción:	Para realizar la prueba, se utilizó un listado con la información de las páginas web para representar la información en formato XML.
Resultados esperados:	Se devolvió un XML con la información de las imágenes.
Observaciones:	

Como resultado de la realización de las pruebas de unidad se pudo comprobar el funcionamiento correcto de cada uno de los módulos del sistema, realizándose ocho pruebas sobre las funcionalidades y clases encargadas de las cargas funcionales de la aplicación, las pruebas resultaron en éxito.

	Cantidad	Fallo	Éxito	Cobertura
Pruebas	8		X	
Total	8			100 %

El prototipo de interfaz de usuario para la aplicación estará conformado de la siguiente manera:

1- Pantalla: Página principal: Donde aparece el logo de la aplicación, las opciones de búsqueda de recursos y el campo donde el usuario especifica los términos de búsqueda.

2- Pantalla: Web: donde el usuario tiene la opción de realizar búsquedas de páginas web relacionadas con los términos de búsqueda, además de poder opinar a favor o no de la aceptación de algún resultado de la búsqueda presentada, elegir desde dónde desea cargar la información, desde internet o desde el propio repositorio de la aplicación.

3- Pantalla: Imágenes: donde el usuario tiene la opción de realizar búsquedas de imágenes relacionadas con los términos de búsqueda.

En la fase de Pruebas

Prueba de aceptación**.**

- Las pruebas de aceptación se crean a partir de los relatos de usuario. El cliente define los escenarios de prueba para verificar si el relato de usuario ha sido correctamente implementado. Un relato de usuario puede tener una o varias pruebas de aceptación. El cliente es responsable de verificar el pasaje de las pruebas de aceptación y priorizar la corrección de las pruebas fallidas.
- Las pruebas de aceptación son pruebas tipo caja negra a nivel del sistema: cada prueba de aceptación corresponde a un resultado producido por el sistema.
- Las pruebas de aceptación deben ser automáticas, correrse frecuentemente, publicarse sus resultados y programarse su corrección para la próxima iteración.
- Deben crearse pruebas de aceptación en cada iteración. Si no hay pruebas de aceptación nuevas no se ha hecho nada nuevo.

- Un relato de usuario no está completo hasta no haber pasado todas sus pruebas de aceptación.

Tabla: Prueba 1 al Módulo Gestionar Información de Páginas Web

Caso de Prueba de Aceptación	
Código: HU1_P1	**Historia de Usuario: 1**
Nombre: Buscar información en páginas web.	
Descripción: Prueba para la funcionalidad de buscar información en páginas web	
Condiciones de ejecución: El usuario debe haber especificado los términos de búsqueda relacionados con la información a buscar.	
Entrada/ Pasos de ejecución: El usuario especifica los términos de búsqueda, escoge la opción de búsqueda "web" y procede a buscar la información.	
Resultado esperado: Los resultados mostrados son los esperados en relación con los términos de búsqueda especificados.	
Evaluación de la Prueba: Prueba satisfactoria.	

Tabla: Prueba 2 al Módulo Gestionar Información de Páginas Web

Caso de Prueba de Aceptación	
Código: HU1_P2	**Historia de Usuario: 1**
Nombre: Mostrar información desde el repositorio. (En caché)	
Descripción: Prueba para la funcionalidad de mostrar información desde el repositorio	
Condiciones de ejecución: El usuario debe haber obtenido los resultados relacionados con los términos de búsqueda.	

Entrada/ Pasos de ejecución: El usuario obtiene los resultados relacionados con los términos de búsqueda y escoge visualizar la información del resultado cargándola desde el repositorio.
Resultado esperado: La información mostrada fue la esperada.
Evaluación de la Prueba: Prueba satisfactoria.

Tabla: Prueba 3 al Módulo Gestionar Información de Páginas Web

Caso de Prueba de Aceptación	
Código: HU1_P3	**Historia de Usuario: 1**
Nombre: Ignorar información de un resultado devuelto.	
Descripción: Prueba para la funcionalidad de ignorar información de un resultado devuelto.	
Condiciones de ejecución: El usuario debe haber obtenido los resultados relacionados con los términos de búsqueda.	
Entrada/ Pasos de ejecución: El usuario obtiene los resultados relacionados con los términos de búsqueda y escoge ignorar un resultado que no sea de su agrado.	
Resultado esperado: Los cambios realizados fueron los esperados.	
Evaluación de la Prueba: Prueba satisfactoria.	

Tabla: Prueba 3 al Módulo Gestionar Información de Páginas Web

Caso de Prueba de Aceptación	
Código: HU1_P3	**Historia de Usuario: 1**
Nombre: Sugerir información de un resultado devuelto.	

Descripción: Prueba para la funcionalidad de sugerir información de un resultado devuelto.
Condiciones de ejecución: El usuario debe haber obtenido los resultados relacionados con los términos de búsqueda.
Entrada/ Pasos de ejecución: El usuario obtiene los resultados relacionados con los términos de búsqueda y escoge sugerir un resultado que sea de su agrado.
Resultado esperado: Los cambios realizados fueron los esperados.
Evaluación de la Prueba: Prueba satisfactoria.

Tabla: Prueba 4 al Módulo Gestionar Información de Páginas Web

Caso de Prueba de Aceptación	
Código: HU1_P4	**Historia de Usuario: 1**
Nombre: Buscar información de un resultado en páginas similares.	
Descripción: Prueba para la funcionalidad de buscar información de un resultado en páginas similares.	
Condiciones de ejecución: El usuario debe haber obtenido los resultados relacionados con los términos de búsqueda.	
Entrada/ Pasos de ejecución: El usuario obtiene los resultados relacionados con los términos de búsqueda y escoge visualizar la información de páginas similares al resultado escogido.	
Resultado esperado: La información mostrada fue la esperada.	
Evaluación de la Prueba: Prueba satisfactoria.	

Tabla: Prueba 1 al Módulo Gestionar Información de Imágenes

Caso de Prueba de Aceptación	
Código: HU2_P1	**Historia de Usuario: 2**

Nombre: Buscar información en imágenes.
Descripción: Prueba para la funcionalidad de buscar información en imágenes.
Condiciones de ejecución: El usuario realiza la búsqueda de imágenes relacionadas con los términos de búsqueda.
Entrada/ Pasos de ejecución: El usuario obtiene los resultados relacionados con los términos de búsqueda.
Resultado esperado: La información mostrada fue la esperada.
Evaluación de la Prueba: Prueba satisfactoria.

Las pruebas de aceptación en Extreme Programming son las pruebas más importantes, porque son las que dan el resultado definitivo de la conformidad del cliente por las funcionalidades implementadas por los desarrolladores, es el cliente el que tiene el conocimiento sobre que es lo que quiere que haga el sistema y es quien comprueba mediante las pruebas de aceptación las funcionalidades para darle el "*visto bueno*".

Generalmente los clientes no saben como escribir pruebas de aceptación que cubran toda la funcionalidad resumida en la Historia de Usuario, por lo que necesitan un guía para desarrollar las pruebas de aceptación, que sería papel principal del probador (Tester) acompañar al cliente durante la realización de las pruebas y alcanzar éxito en ellas. En este capítulo se ha presentado un sencillo modelo de Caso de Prueba de Aceptación con el objetivo de lograr que:

- El proceso de desarrollo de las pruebas ayuden al cliente a clarificar y concretar la funcionalidad de la historia y favorezca la comunicación entre el cliente y el equipo de desarrollo.
- Ayuden a corregir fallos u omisiones en las historias de usuario.
- Permita corregir errores en las ideas del cliente, encontrando resultados que el cliente esperó encontrar en la implementación pero para los que no existía ningún camino de ejecución que condujera a ello.
- Garantizar la cobertura de la funcionalidad de las pruebas de aceptación, garantizando que no se dejara ningún punto importante de la funcionalidad de una historia de usuario sin probar.

	Cantidad	Fallo	Éxito	Cobertura
Probadas	6		X	
Total	6			100 %

Conclusiones

Luego de realizar un estudio de los pasos a seguir para llevar a cabo las respectivas fases de Diseño, Codificación y Pruebas, y los conceptos a tener en cuenta en cada una de ellas, se ha podido comprobar que la metodología proporciona una base sólida para la creación de aplicaciones de alta calidad, fáciles de mantener y que responde en todo momento a los intereses del cliente.

El “sistema de recuperación de información web aplicando técnicas de trabajo colaborativo” hasta aquí descrito pretende dar solución a una problemática que afecta a toda una comunidad, en la cual no se habían concretado los pasos para el desarrollo de una aplicación de este tipo, y poder contar así con una aplicación funcional para tener acceso a la información web disponible en la red de la universidad. Con la realización de este trabajo se pretende promover el uso de las técnicas de trabajo colaborativo en los proyectos de la Universidad, principalmente en los proyectos de software libre, toda esta idea se apoya en las características de la universidad por poseer un gran personal dedicado a objetivos similares, que con frecuencia usan las mismas herramientas y utilizan documentación similar para lograr objetivos concretos en la docencia y la producción, se persigue además que se profundice en los estudios y se posea más conocimiento sobre los motores de búsqueda, los modelos de recuperación de información, a estudiar y mejorar el posicionamiento de los sitios de alta en los buscadores, buscadores tanto del país como los que operan libremente en internet.

Se podría afirmar que la aplicación en los próximos meses o años puede convertirse en una potente herramienta de referencia y consulta obligatoria por los estudiantes y profesores, principalmente por ser un software libre, por ser una herramienta donde se aplica el trabajo colaborativo, lo que permitirá a los usuarios día a día dar aportes para mejorar el funcionamiento del sistema, haciéndolo cada vez más amigable, eficiente, rápido y efectivo.

Recomendaciones

El Grupo de Investigación y Desarrollo de Internet (GIDI, antes Operación Verdad) es un proyecto orientado a la defensa política de Cuba en internet y a la producción, el que juega hoy un papel fundamental en el estudio de internet en la universidad. Después de un período investigando y desarrollando la aplicación sistema de recuperación de información dentro del mismo, se ha hecho necesario para un mejor funcionamiento futuro del sistema, tener en cuenta como recomendación, las siguientes ideas:

- Llevar a cabo el continuo desarrollo del sistema por parte de los integrantes del proyecto Grupo de Investigación y Desarrollo de Internet.
- Agregar nuevas funcionalidades al sistema de forma que tributen al trabajo colaborativo y cumplan con las expectativas y conformidad de los usuarios de la comunidad.
- Continuar optimizando los algoritmos de recuperación implementados.
- Continuar investigando los algoritmos de posicionamiento web.
- Incentivar el uso de la aplicación en la universidad.
- Motivar a estudiantes y profesores de todo el centro a que colaboren en el desarrollo de la aplicación.

Bibliografía

1. www.tramullas.com. *Introducción a la Documática.* [En línea] [Citado el: 27 de Enero de 2009.] http://tramullas.com/documatica/3-1.html.

2. www.mariapinto.es. *Electronic Context Management Skills.* [En línea] [Citado el: 15 de Enero de 2009.] www.mariapinto.es/e-coms/recu_infor.htm#ri1.

3. www.ecscw.org. *European Conference on Computer-Supported Cooperative Work .* [En línea] [Citado el: 20 de Enero de 2009.] http://www.ecscw.org/1991/01.pdf.

4. www.geocities.com. *Monografías: Motores de búsqueda.* [En línea] [Citado el: 22 de Enero de 2009.] www.geocities.com/motoresdebusqueda/crawlers.html.

5. www.buscadores.ws. *Información Sobre Yahoo, Historia De Yahoo, Características De Yahoo.* [En línea] [Citado el: 15 de Enero de 2009.] www.buscadores.ws/ficha_yahoo.htm.

6. www.buscadores.ws. *Ficha Técnica.* [En línea] [Citado el: 22 de Enero de 2009.] http://www.buscadores.ws/ficha_google.htm.

7. www.posicionamientosuperior.com. *Posicionamiento Superior.com.* [En línea] [Citado el: 23 de Enero de 2009.] http://www.posicionamientosuperior.com/terminologia/o.htm.

8. www.burcet.net. *Instrumentos Inmateriales.* [En línea] [Citado el: 27 de Enero de 2009.] http://www.burcet.net/posicionamiento/index.htm.

9. **Pérez Valdés, Damián.** www.radiocaribe.co.cu. *Noticias de Informática.* [En línea] [Citado el: 05 de Febrero de 2009.] http://www.radiocaribe.co.cu/secundaria/informatica/342.htm.

10. www.oness.sourceforge.net. *ONess.* [En línea] [Citado el: 08 de Febrero de 2009.] http://oness.sourceforge.net/proyecto/html/ch05s02.html.

11. www.oness.sourceforge.net. *ONess.* [En línea] [Citado el: 08 de Febrero de 2009.] http://oness.sourceforge.net/proyecto/html/ch05.html#N102B1.

12. **Pecos, Daniel.** www.netpecos.org. *PostGreSQL vs. MySQL.* [En línea] [Citado el: 08 de Febrero de 2009.] http://www.netpecos.org/docs/mysql_postgres/.

13. www.websolutionsnic.blogspot.com. *Desarrollo Web en Nicaragua.* [En línea] [Citado el: 07 de Febrero de 2009.] http://websolutionsnic.blogspot.com/2008/09/los-diferentes-lenguajes-de-programacin.html.

14. www.modelosrecuperacion.tripod.com. *Modelos de Recuperación.* [En línea] [Citado el: 05 de Febrero de 2009.] http://www.modelosrecuperacion.tripod.com/booleano.html.

15. www.es.wikipedia.org. *Wikipedia, la enciclopedia libre.* [En línea] [Citado el: 06 de Enero de 2009.] http://es.wikipedia.org/wiki/Zend_Studio.

16. www.netbeans.org. *NetBeans IDE.* [En línea] [Citado el: 07 de Febrero de 2009.] http://www.netbeans.org/features/index_es.html.

17. **Escribano, Gerardo Fernández.** www.info-ab.uclm.es. *Departamento de Sistemas Informáticos.* [En línea] [Citado el: 08 de Febrero de 2009.] http://www.info-ab.uclm.es/asignaturas/42551/trabajosAnteriores/Presentacion-XP.pdf.

Glosario de Términos

- **URL** significa ***U****niform* ***R****esource* ***L****ocator*, localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.
- **Indexar** refiere a la acción de registrar ordenadamente información para elaborar su índice, En informática, tiene como propósito la elaboración de un índice que contenga de forma ordenada la información, esto con la finalidad de obtener resultados de forma sustancialmente más rápida y relevante al momento de realizar una búsqueda.
- La **recuperación de información**, llamada en inglés *Information retrieval* (IR), es la ciencia de la búsqueda de información en documentos, búsqueda de los mismos documentos, la búsqueda de metadatos que describan documentos, o, también, la búsqueda en bases de datos, ya sea a través de internet, intranet, para textos, imágenes, sonido o datos de otras características, de manera pertinente y relevante.
- **WWW** o **World Wide Web** es un sistema de documentos de hipertexto y/o hipermedias enlazados y accesibles a través de Internet. La Web fue creada alrededor de 1989 por el inglés Tim Berners-Lee y el belga Robert Cailliau.
- **Internet** es un conjunto descentralizado de redes de comunicación interconectadas, que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial.
- **CEO: son** las siglas **CEO** del inglés ***C****hief* ***E****xecutive* ***O****fficer*. Es el encargado de máxima autoridad de la gestión y dirección administrativa en una empresa, organización o institución.
- **Jerry Yang** es un empresario Estadounidense y el Co-fundador, CEO y Director de Yahoo! Inc.
- **David Filo** es un empresario Estadounidense, Co-fundador de la empresa Yahoo!.
- **Lawrence Edward "Larry"** es un empresario estadounidense, es el presidente y director de productos de Google Inc.
- **Sergey Brin** es creador y co-fundador del popular motor de búsqueda Google.
- **Menlo Park** es una ciudad en el condado de San Mateo, California en los Estados Unidos de América.

- **Motorola** es una empresa estadounidense especializada en la electrónica y las telecomunicaciones.
- **GPI**. Grupo de Procesamiento de Imágenes, proyecto de la Universidad de las Ciencias Informáticas dedicado al procesamiento o tratamiento de imágenes, con sitio en la web con al dirección GPI.uci.cu.
- **Wikia Search** es un buscador creado por la organización Wikia.
- La **Minería de Datos** (**DM**, *Data Mining*) consiste en la extracción no trivial de información que reside de manera implícita en los datos. Dicha información era previamente desconocida y podría resultar útil para algún proceso. La minería de datos prepara, sondea y explora los datos para sacar la información oculta en ellos.
- **Hardware:** corresponde a todas las partes físicas y tangibles de una computadora: sus componentes eléctricos, electrónicos, electromecánicos y mecánicos; sus cables, gabinetes o cajas, periféricos de todo tipo y cualquier otro elemento físico involucrado.
- **Software:** la palabra «software» se refiere al equipamiento lógico o soporte lógico de un computador digital, y comprende el conjunto de los componentes lógicos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema «hardware».
- **Tesauros**: La palabra **tesauro**, derivado del neo latín que significa *tesoro*, se refiere a listado de palabras o términos empleados para representar conceptos.
- **Spider:** es un programa que inspecciona las páginas del World Wide Web de forma metódica y automatizada.
- **Hipertextos**: en informática, es el nombre que recibe el texto que en la pantalla de una computadora te conduce a otro texto relacionado. La forma más habitual de hipertexto en documentos es la de hipervínculos o referencias cruzadas automáticas que van a otros documentos.
- Una **araña web** (o araña de la web) es un programa que inspecciona las páginas del World Wide Web de forma metódica y automatizada.
- **Interfaz**: parte de un programa que permite el flujo de información entre un usuario y la aplicación, o entre la aplicación y otros programas o periféricos. Esa parte de un programa está

constituida por un conjunto de comandos y métodos que permiten estas intercomunicaciones, puede ser del tipo GUI, o línea de comandos.

- **Código abierto** (en inglés ***open source***) es el término con el que se conoce al software distribuido y desarrollado libremente.
- **Algoritmo**: un algoritmo (del latín, *dixit algorithmus* y éste a su vez del matemático persa al-Jwarizmi) es una lista bien definida, ordenada y finita de operaciones que se ejecutan en un tiempo finito, consumiendo una cantidad de recurso del dispositivo donde se ejecutan, y permiten hallar la solución a un problema.
- **META:** los **meta tags** son unos caracteres pertenecientes al HTML que se deben de escribir dentro del tag general <head> y que lo podemos definir como líneas de código que indican a los buscadores que le indexan por qué términos debe ser encontrada la página.
- **IDE** es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica de usuario (Grafic User Interface siglas en inglés GUI).
- **FTP** es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP, basado en la arquitectura cliente-servidor.
- **SFTP** es un protocolo de red que proporciona la funcionalidad necesaria para la transferencia y manipulación de archivos sobre un flujo de datos fiable.
- **HTML** (Lenguaje de Marcas de Hipertexto), es el lenguaje de marcado predominante para la construcción de páginas web.
- **CSS** es un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML, es separar la *estructura* de un documento de su *presentación*.
- **XML** es un metalenguaje extensible de etiquetas.
- **WYSIWYG** es el acrónimo de ***W****hat* ***Y****ou* ***S****ee* ***I****s* ***W****hat* ***Y****ou* ***G****et* (en inglés, "lo que ves es lo que obtienes"). Se aplica a los procesadores de texto y otros editores de texto con formato (como los editores de HTML).
- **JDK** es un software que provee herramientas de desarrollo para la creación de programas en java.

- **API** la **interfaz de programación de aplicaciones** (**API**) es el conjunto de funciones y procedimientos (o métodos, si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
- **CGI** es una importante tecnología de la World Wide Web que permite a un cliente (explorador web) solicitar datos de un programa ejecutado en un servidor web. CGI especifica un estándar para transferir datos entre el cliente y el programa.
- Un **APPLET** es un componente de una *aplicación* que se ejecuta en el contexto de otro programa, por ejemplo un navegador web.
- **Programación Orientada a Objetos** (**POO** u **OOP** según sus siglas en inglés) es un paradigma de programación que usa objetos y sus interacciones para diseñar aplicaciones y programas de computadora. Está basado en varias técnicas, incluyendo herencia, modularidad, polimorfismo y encapsulamiento.
- **Java** es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90.
- **C** es un lenguaje de programación creado en 1972 por Kenneth L. Thompson y Dennis M. Ritchie en los Laboratorios Bell.
- **C++** es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue el extender al exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos.
- Un **Trigger** (o disparador) en una Base de datos, es un procedimiento que se ejecuta cuando se cumple una condición establecida al realizar una operación de inserción (INSERT), actualización (UPDATE) o borrado (DELETE).
- **Caché** se aplica a un conjunto de datos duplicados de otros originales, con la propiedad de que los datos originales son costosos de acceder, normalmente en tiempo, con respecto a la copia en el caché. Cuando se accede por primera vez a un dato, se hace una copia en el caché; los accesos siguientes se realizan a dicha copia, haciendo que el tiempo de acceso medio al dato sea menor.
- **MSN** (abreviación de Microsoft Network) es una colección de servicios de internet proporcionados por Microsoft. Inicialmente lanzado el 24 de agosto de 1995, para coincidir con el lanzamiento de Windows 95.

- **Kent Beck:** Es el creador de la Extreme Programming, desarrollada cuando servía como líder de proyecto en el Chrysler Comprehensive Compensation Project (C3). Beck fue uno de los 17 signatarios originales del Manifiesto Ágil en el 2001. Beck ha sido pionero en el trabajo con patrones de diseño, el desarrollo orientado a pruebas y la aplicación comercial de Smalltalk. Beck popularizó las tarjetas CRC junto a Ward Cunningham, y creó junto a Erich Gamma el framework de pruebas unitarias JUnit. Tiene una maestría en Ciencias por la Universidad de Oregón.

- **Prueba Unitaria**: en programación, es una forma de probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.

- **Pruebas de integración** son aquellas que se realizan en el ámbito del desarrollo de software una vez que se han aprobado las **pruebas unitarias**. Únicamente se refieren a la prueba o pruebas de todos los elementos unitarios que componen un proceso, hecha en conjunto, de una sola vez. Consiste en realizar pruebas para verificar que un gran conjunto de partes de software funcionan juntos.

- **Mac: Macintosh**, es el nombre con el que actualmente se refiere a cualquier computadora personal diseñada, desarrollada, construida y comercializada por Apple Inc.

- **GNU/Linux** es el término empleado para referirse al sistema operativo similar a Unix que utiliza como base las herramientas de sistema de GNU y el núcleo Linux. Su desarrollo es uno de los ejemplos más prominentes de software libre; todo el código fuente puede ser utilizado, modificado y redistribuido libremente por cualquiera bajo los términos de la GPL de GNU (**L**icencia **P**ública **G**eneral de GNU).

- **Sun Microsystems** es una empresa informática de Silicon Valley, fabricante de semiconductores y software. Fue constituida en 1982 por el alemán Andreas von Bechtolsheim y los norteamericanos Vinod Khosla, Bill Joy, Scott McNealy y Marcel Newman. Las siglas SUN se derivan de «Stanford University Network», proyecto que se había creado para interconectar en red las bibliotecas de la Universidad de Stanford.

- **Solaris** es un sistema operativo de tipo Unix desarrollado por Sun Microsystems desde 1992 como sucesor de SunOS. Es un sistema certificado oficialmente como versión de Unix. Funciona en arquitecturas SPARC y x86 para servidores y estaciones de trabajo.