

**Universidad de las Ciencias Informáticas
Facultad 8**



*Sistema para el apoyo a la toma de decisiones de los
webmasters, en el ámbito del Posicionamiento Web.*

Rol Analista de Sistema.

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

Autora: Aliuska Garcia Machado.

Tutor: Damir Góngora Mora.

Co-tutor: Nelson Rodríguez Proenza.

Ciudad de la Habana, Julio 2008.

Año del 50 Aniversario de la Revolución.



*“La tarea que los graduados de la UCI tienen por delante es grandiosa.
Espero que la cumplan, y la cumplirán.”
Fidel Castro Ruz
17 de julio del 2007*

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Aliuska Garcia Machado

Damir Góngora Mora

AGRADECIMIENTOS

A mis padres y hermano por su apoyo incondicional, amor, y confianza en todo momento, que me han llevado a ser lo que soy en estos momentos.

A mi "Aya" (abuela) Yuyú y a mi abuela Elida dos segundas madres para mí, son las mejores abuelas que pueda tener alguien, gracias por estar siempre preocupadas por su nieta.

A mis abuelos Giraldo y Domingo por regañarme en los momentos de mis chiquilladas y lograr que fuera la persona respetuosa que ahora soy.

A mis tíos, mis tías, mis primos y toda la familia en general, por quererme y estar al tanto en todo momento de lo que ocurre en mi vida, dándome consejos y fuerza para seguir adelante.

A mi Yoita por ser el hombre que siempre busqué, por estar siempre presente en momentos buenos y malos, durante los cinco años de esta escuela que me dio la oportunidad de conocerte.

A mis suegros y cuñadas que mejor ni mandados a hacer, también a toda la familia en general de Yoa que me acogieron desde el principio con mucho cariño.

A mi Tutor Damir, por apoyarme en lo que fuera en el desarrollo de este trabajo.

A mi Cotutor y jefe en mi proyecto Nelson, del cual he aprendido que se debe ser correcto y responsable hasta en el más mínimo trabajo que se desarrolle. Al cual también le debo el agradecimiento del desarrollo de esta tesis.

A mis amigas y amigos del grupo 8503 por haber compartido conmigo todos estos años mis alegrías y tristezas.

A todos los integrantes de mi proyecto Asdrúbal, Alfredo dirigentes que me guiaron en el trabajo, a los integrantes de mi línea y demás por apoyarme en el trabajo siempre.

Por último y el más importante agradecimiento va hacia la Revolución y en especial a Fidel Castro Ruz porque sin él no existiéramos ni como profesionales ni como personas.

DEDICATORIA

A la memoria de mi primo Alien que fue un hermano para mi y siempre lo llevaré en mi corazón.

A mis padres y hermano que son la razón por la que vivo y siento de corazón que tenerlos es poseer el mayor tesoro del mundo.

A mi chichi, que ha sido mi amor, mi compañero y amigo en todo momento.

RESUMEN

En estos años donde Internet es el centro de búsqueda de información, el Posicionamiento Web es de una importancia incalculable. No estar en los primeros lugares de los resultados de búsquedas en estos tiempos, es como pertenecer a la web invisible. Por este motivo, en el proyecto Operación Verdad se destinó una línea de investigación enfocada por completo a esta rama de la informática, y de dicha investigación surgió la idea de una herramienta que juegue un papel fundamental a la hora de que un webmaster se trace una estrategia de Posicionamiento Web para su página. En el presente trabajo se le desarrolla la ingeniería de requerimientos a la herramienta *SeoWebMaster (Sistema para el apoyo a la toma de decisiones de los webmasters, en el ámbito del Posicionamiento Web)*. Para ello se llevó a cabo un estudio de las Metodologías de Desarrollo, Lenguaje de Modelado y las Herramientas CASE a utilizar. Se desarrollan también los artefactos correspondientes al rol de analista de sistemas; esto es de gran importancia en el entendimiento entre cliente y desarrolladores. Por último se evalúa el trabajo realizado mediante métricas dirigidas a medir la calidad de la especificación de requisitos, y de la confección del diagrama de casos de uso del sistema, obteniéndose resultados positivos.

PALABRAS CLAVE

Posicionamiento Web, analista de sistemas, SeoWebMaster, ingeniería de requerimientos

TABLA DE CONTENIDOS

AGRADECIMIENTOS	I
DEDICATORIA.....	II
RESUMEN	III
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	6
1.1 Introducción	6
1.2 Elementos importantes para entender el Posicionamiento Web	6
1.2.1 Internet.....	6
1.2.2 Buscadores.....	7
1.2.3 Page Rank	10
1.2.4 Posicionamiento Web.....	11
1.3 Trabajos relacionados con el tema en el ámbito nacional e internacional.	14
1.3.1 Promotor Web (Dynamic Submission 7.0):.....	14
1.3.2 Active Link Exchange:	14
1.3.3 Active Web Traffic:.....	15
1.3.4 Google Monitor Software:	15
1.3.5 Herramientas Sencillas	15
1.4 Metodologías de Desarrollo.....	17
1.4.1 Metodología Rational Unified Process (RUP).....	17
1.4.2 Metodología Extreme Programming (XP).....	20
1.4.3 ¿Por qué RUP?.....	22
1.5 El analista de Sistema en RUP	23
1.6 Lenguaje de Modelado.	24
1.6.1 Unified Modeling Language (UML).....	24
1.7 Herramientas CASE.....	25
1.7.1 Visual Paradigm.	25
1.7.2 Rational Rose Enterprise Edition.	25
1.7.3 ¿Por qué Visual Paradigm?.....	26
1.8 Tecnología.....	26
1.8.1 Framework 2.0	26
1.8.2 C#	28
1.8.3 Visual Studio 2005 .NET.....	30
1.9 Conclusiones	32
CAPÍTULO 2: ANÁLISIS DEL SISTEMA	33
2.1 Introducción	33
2.2 Modelo de dominio.	34
2.3 Requerimientos del sistema.....	35

2.3.1 Estrategia de captura de requisitos utilizada.....	36
2.3.2 Tipos de requerimientos más comunes.....	36
2.3.3 Requerimientos funcionales del software.....	38
2.3.4 Requerimientos no funcionales del software.....	39
2.4 Ejemplificación de la selección de las características.....	41
2.5 Definición de los actores.....	43
2.6 Diagrama de Casos de Uso del Sistema	44
2.7 Descripción de los casos de uso del sistema.....	44
2.8 Patrones de casos de uso.....	54
2.8.1 ¿Qué es un patrón?	54
2.8.2 Patrones de casos de uso utilizados.....	54
2.9 Conclusiones.....	55
CAPÍTULO 3: EVALUACIÓN DE RESULTADOS	57
3.1 Introducción	57
3.2 Métricas	57
3.2.1 Métricas para el trabajo realizado.....	58
3.3 Especificación de Requisitos	58
3.4 Grado de validación de Requisitos.....	60
3.5 Algunas Métricas aplicadas en el Diagrama de Casos de Uso del Sistema	61
3.6 Buenas prácticas para confección de los prototipos no funcionales (Interfaces)	65
3.7 Análisis de los resultados	66
3.8 Conclusiones	67
CONCLUSIONES.....	68
RECOMENDACIONES	69
BIBLIOGRAFÍA REFERENCIADA.....	70
BIBLIOGRAFIA CONSULTADA.....	71
GLOSARIO.....	74

INTRODUCCIÓN

Debido al constante desarrollo de las tecnologías informáticas, Internet se ha convertido en la fuente de información más grande y popular. Cuba desde sus primeros años de revolución ha sido objeto de tergiversaciones por parte de su mayor oponente, y para que el mundo conozca la verdad del archipiélago cubano se necesita aprovechar las facilidades que brinda la gran Red de redes.

En estos tiempos el Posicionamiento Web se ha convertido en una de las tantas herramientas más cotizadas por los webmasters. Para aumentar el tráfico de visitas de una página, se tiene que aparecer en las primeras posiciones de los buscadores, ya que las características del ciudadano de estos tiempos se centran en encontrar la información lo más pronto que sea posible. Además resultados de investigaciones muestran que “más del 80% de usuarios de Internet visitan sólo los tres primeros resultados de una búsqueda” (Gosende 2007).

Es por esto que toda página web cubana además de estar en Internet tiene que estar bien posicionada, para que un mayor número de personas tenga la posibilidad de conocer sobre nuestro país. En estos momentos si una persona se dispone a buscar un tema relacionado con Cuba, a pesar de existir esa información en páginas cubanas, éstas no aparecen en las primeras posiciones; en cambio se muestran páginas que, o no dan la información exacta o la dan tergiversada.

En Cuba no existe el hábito de realizarle a una página web, las modificaciones necesarias para posicionarla. Además este trabajo lleva tiempo, no se hace de un día para otro, porque requiere atención tanto de la parte técnica de la página, como de la parte visual para el usuario, y desarrollando una aplicación que apresure este proceso se le ahorraría tiempo al webmaster a la hora de construir su página web.

Es por esto que se ha llegado a la conclusión de que nuestro país necesita de una herramienta que ayude a la toma de decisiones de los webmaster a la hora de posicionar sus sitios en los buscadores, debido a que en el mundo las aplicaciones informáticas y las compañías que brindan servicio de Posicionamiento Web son la mayoría propietarias.

Luego de analizar lo anteriormente planteado se puede señalar que uno de los problemas fundamentales que existe, es la falta de importancia que se le brinda al tema de Posicionamiento Web en Cuba, otro aspecto fundamental es la necesidad de la herramienta informática de ayude a los webmasters en éste ámbito, y para que sea posible el desarrollo de dicha herramienta primeramente se hace necesario un análisis profundo, pues sin esto no se podrían conocer las necesidades del cliente, que en este caso es el jefe de la línea de investigación de Posicionamiento Web, en el proyecto Operación Verdad, como tampoco se podría conocer con certeza si ya existían otras herramientas con funcionalidades parecidas. Se puede plantear entonces como **problema científico** la carencia del análisis de una herramienta, que permita lograr que los sitios cubanos estén bien posicionados en los diferentes buscadores.

Para que sea posible dar solución a este problema se plantea el siguiente **objetivo general**: Realizar el análisis de una herramienta informática que ayude a la toma de decisiones de los webmasters, para lograr un buen Posicionamiento Web.

El **objeto de estudio** son las estrategias y procesos de gestión de información que ayuden al Posicionamiento Web, lo cual define como **campo de acción** el análisis de los procesos automatizados de gestión de información, que ayuden a la toma de decisiones del webmaster en el ámbito del Posicionamiento Web.

Para cumplir con el objetivo general de la investigación se trazaron los siguientes **objetivos específicos** y las siguientes **tareas** para poder darle cumplimiento a cada uno de ellos:

- Realizar un estudio para analizar las estrategias del Posicionamiento Web existentes.

Tareas

- Consulta de la documentación que se utiliza en el mundo y en Cuba sobre el Posicionamiento Web.
- Realización de una búsqueda de documentación sobre el tema del funcionamiento de los buscadores.

- Realizar un estudio para analizar casos de éxitos obtenidos por otras instituciones.

Tareas

- Búsqueda e interacción con herramientas que se utilizan para el Posicionamiento Web.
- Comunicación con los webmasters de los diferentes sitios cubanos bien posicionados en la web.

- Realizar un levantamiento de requisitos que permitan conocer las necesidades del usuario y el cliente.

Tarea

- Reunirse con el cliente para levantar los requisitos esenciales que éste desee para la realización del sistema.

- Revisar bibliografía referente a Ingeniería de Software para definir artefactos referentes al desarrollo del trabajo.

Tareas

- Estudio del estado actual de la documentación que se utiliza en la UCI sobre la Ingeniería de Software.
- Análisis de documentación sobre el analista de sistema.
- Consulta de documentos sobre Metodologías de Desarrollo, Lenguajes de Modelado y Herramientas CASE.

- Elaborar los documentos que describen el trabajo del analista del sistema.

Tareas

- Confección del documento de los requerimientos funcionales y no funcionales.
- Confección del diagrama de casos de uso.
- Confección de la descripción de los casos de uso.
- Confección del documento Glosario de Términos, para la comprensión del trabajo.

- Analizar los resultados obtenidos durante el desarrollo de artefactos que obtiene el analista de sistemas.

- Estudio de las diferentes métricas de calidad que se adecuan al análisis de lo desarrollado.
- Selección de las métricas a emplear en el trabajo realizado.
- Aplicación de las métricas seleccionadas para la medición de la calidad.

- Realización de revisiones al trabajo, para asegurar no cometer errores.

Conociendo esto se puede plantear la siguiente **idea a defender**: Si se logra desarrollar el análisis de una herramienta informática que ayude a la toma de decisiones del webmaster, posibilitará el mejor entendimiento entre cliente y desarrolladores, logrando una mejor implementación de la herramienta, y por lo tanto la aplicación cumplirá con todos los requisitos planteados.

Métodos Científicos

El método científico de investigación es la forma de abordar la realidad, de estudiar la naturaleza, la sociedad y el pensamiento, con el propósito de descubrir su esencia y sus relaciones. El método científico se puede clasificar en teóricos y empíricos de los cuales se pusieron en práctica en el presente trabajo los que se muestran a continuación:

Método Teóricos

Análisis y Síntesis: Por medio de este método se realizó una investigación de todo lo relacionado con el Posicionamiento Web y sus aplicaciones, permitiendo analizar la documentación existente vinculada al desarrollo de estas estrategias, así como en la extracción de los elementos más importantes relacionados con el campo de acción.

Histórico Lógico: El presente método permitió analizar la evolución de los procesos de gestión de información que ayuden al Posicionamiento Web desde su creación hasta el día de hoy.

Método de la Modelación: Mediante este modelo se elaboraron modelaciones para obtener una visualización simplificada de los diferentes procesos, descubriendo así, nuevas cualidades del objeto de estudio.

Métodos Empíricos

Entrevista: Este método sirvió para entrevistarse con el cliente para obtener información acerca de los requerimientos que debe cumplir el software.

El presente trabajo esta compuesto por tres capítulos, conclusiones, recomendaciones y bibliografía.

El **Capítulo 1** está compuesto por: El estado del arte del Posicionamiento Web, así como elementos importantes para poder entenderlo. Se proponen Metodologías de Desarrollo, el Lenguaje de Modelado y las Herramientas CASE, también se da a conocer el Lenguaje de Programación y la Plataforma donde desarrollar el sistema.

El **Capítulo 2** está compuesto por: El desarrollo de los modelos necesarios para el trabajo del analista del sistema, ya que así se entiende con más profundidad el trabajo que se está realizando. También se reflejan los requerimientos del software que plantea el cliente, además de estar presentes los patrones que se tuvieron en cuenta para el modelado de casos de uso del sistema.

El **Capítulo 3** está compuesto por: El análisis de los resultados mediante algunas métricas que están diseñadas para evaluar este tipo de trabajo. Primeramente se definen las métricas y luego se aplican al trabajo realizado, y para concluir se analizan los resultados obtenidos, teniendo en cuenta la opinión del cliente que es el que da el resultado final.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1

1.1 Introducción

En este capítulo se abordarán conceptos fundamentales que ayuden a entender el mundo del Posicionamiento Web y a desarrollar una aplicación para la ayuda al webmaster en la realización más óptima de su página web. Además se hace necesario fundamentar la selección de la Metodología de Desarrollo, el Lenguaje de Modelado y la Herramientas CASE que se emplearon en la realización de la aplicación.

1.2 Elementos importantes para entender el Posicionamiento Web

1.2.1 Internet

Al pasar de la historia el mundo ha sufrido cambios que le han cambiado la forma de vivir a las personas, ejemplo de uno de esos cambios es Internet. En estos tiempos no es alarmante que dos personas rompan la barrera de la distancia, para intercambiar información, posibilidad que no tuvieron muchas personas antes de existir la Red de redes.

En Internet existen alrededor de 800 millones de páginas y está en constante aumento, y para acceder a toda esta información una persona se tardaría al menos veinte mil años. Otro dato interesante sería que en Inglaterra se ha llegado a traficar más datos que voz por las vías telefónicas(Millán), esto sin lugar a dudas es una forma de darse cuenta de que Internet se ha convertido en una parte importante y a veces hasta indispensable en la vida de la mayoría de esta generación. Pero cuidado, esto pudiera también traer problemas a la sociedad, por el temor a que una persona se enajene al mundo y no existan las relaciones sociales fundamentales para el desarrollo de la humanidad. Ahora vale preguntar, ¿Cómo surgió este fenómeno tecnológico?

La necesidad de la información fue la causante de que surgiera Internet, pero para fines militares y científicos como la 2da Guerra Mundial y lanzamiento al espacio del primer satélite soviético. La

primera red fue creada por Estados Unidos en el año 1969, totalmente militar, y se llamó ARPANET (Red de la Agencia de Investigación de Proyectos Avanzados). La característica principal de esta red debía ser que todos los nodos tuvieran la misma importancia, para cuando dejara de funcionar una por problemas de la guerra no afectara el sistema en general.

A partir de aquí se fueron interesando investigadores y científicos que tenían otra visión, que ya no era militar, por ejemplo Douglas Engelbart, se preocupaba por el trabajo de colaboración a través del hipertexto, y por otro lado ya Licklider discutía su concepto de “Red Galáctica”. Luego ya en 1972 se dio a conocer la red, al aparecer el primer programa de correo electrónico.

En ese mismo año es introducida la arquitectura abierta por Robert Kahn que se le llamó Internetworking, porque conectaba varias redes. Luego en 1983 es cuando se considera que nació Internet, al separarse la parte civil de la militar en la red. En 1989 Tim Berners-Lee elabora un sistema de hipertexto compartido, que era una primera versión de la WWW (World Wide Web), la cual facilitó la aparición del primer navegador Mosaic en 1993, que ya dio paso a la expresión de “surfear la Web”.

Esto trae consigo una cantidad significativa de páginas web, ya que en 1997 existían ya 17 millones de servidores, y además la información estaba dispersa y se necesitaba la presencia del buscador.

1.2.2 Buscadores

En estos tiempos la mayoría de las personas alguna vez han buscado información en la web, y para esto han utilizado un buscador, a continuación se muestran algunas definiciones:

1. *“Página en Internet que permite buscar información a través de ella, bien sea tecleando nosotros mismos una serie de palabras clave, o bien empleando el sistema de menús que la página incorpora. Cada día ofrecen más servicios, entre los que se incluyen noticias, chats (...).”* (Cabanés 2007)

2. *“Es el módulo que por medio de una interfaz captura las palabras clave en que el usuario basa su pregunta y consultado los índices genera una página con enlaces a las páginas que contienen la información correspondiente”*(Ibon Aizpurua)
3. *“Sitio web que gestiona una base de datos confeccionada por robots. Esta base de datos almacena directamente el código de las páginas visitadas por los bots o parte de él.*(Ugarte 2002)
4. *“Un buscador es un software que crea índices de bases de datos o de sitios web en función de los títulos de los archivos, de palabras clave, o del texto completo de dichos archivos. El usuario se conecta con un buscador y especifica la palabra o las palabras clave del tema que desea buscar. El buscador devuelve una lista de resultados presentados en hipertexto, es decir que se pueden pulsar y acceder directamente al archivo correspondiente.”* (MLmundoenlinea.cl)

Se puede apreciar entonces que la última definición, recoge con más exactitud el significado de un buscador, ya que hace referencia a la parte que siempre ve el usuario como también a la parte del funcionamiento interno de éste. Pero en los inicios de las búsquedas en Internet no se utilizaba este tipo de buscadores, a continuación se dará una breve historia de cómo surgió el buscador de hoy en día.

El primer buscador surgió en 1990, llamado Archie y su primera versión fue escrita por Alan Emtage, Bill Heelan y Peter J. Deutsch. Sus primeras versiones contenían listas de ficheros que se encontraban en una lista de servidores FTP (File Transfer Protocol), que no es más que un protocolo de transferencia de archivos entre sistemas conectados a la red, que facilita el intercambio de información entre dos máquinas, estas listas de ficheros para ser solicitadas, se necesitaba almacenarlas en el ordenador y se hacía la consulta con un comando en UNIX, que no es más que familias de sistemas operativos que posibilita la ejecución de distintas tareas de forma simultánea y soporta un gran número de usuarios.

Con el paso del tiempo y avance de la tecnología surgen los directorios de páginas web (sitio web donde las páginas se clasifican por temas y categorías para ayudar a recuperar la información, y están administrados por personas).

Ya en la actualidad los buscadores se guían por índices automáticos que le son asignados a determinada página web. Para entender este funcionamiento es necesario conocer las partes fundamentales del buscador.

Araña: *“Es un programa que navega la web yendo desde cada página a todas las que ésta da enlace y enviando e incorporando el código o partes de él a una base de datos de referencia”.*(Ugarte 2002)

Base de datos documental: *“Conjunto de registros (unidades de información relevante) ordenados y clasificados para su posterior consulta, actualización o cualquier tarea de mantenimiento mediante aplicaciones específicas.”* (Google 2008)

Motor de búsqueda: *“Programa que selecciona y jerarquiza resultados entre las entradas de una base de datos en función de unos términos y criterios de búsqueda.”* (Ugarte 2002)

Muchas personas piensan que cuando hacen una consulta al buscador, éste en ese momento hace la búsqueda por toda la red y después muestra la información, tal pensamiento es una equivocación pues tardaría mucho en brindar el resultado.

Lo que realmente sucede es que, las arañas recorren la red cada cierto tiempo y se crea para cada página un fichero invertido, ¿qué es un fichero invertido?

1. *“Estructura lógica conformada por varios archivos para hacer más rápida la recuperación de registros de una base de datos. Contiene el diccionario de términos recuperables cada uno de ellos por el término propiamente dicho y las referencias a los registros de la base de datos donde se encuentra incluido.(...)”* (Bonilla 1992)

Como se puede apreciar es un documento que resume las palabras relevantes de la página y otras características, es por esta razón que forma parte de la información que se almacena en la base de datos documental.

Los robots a la hora de caracterizar la página se fijan entre otras cosas en el título, en la descripción y en el contenido de la página principal, que debe estar explicado con detalle de qué se trata el sitio en

general. Otros factores que tiene en cuenta el buscador son: el texto en negrita, los primeros párrafos y en las etiquetas con mayor jerarquía.

Pero para que un usuario este satisfecho con la búsqueda, tiene que encontrar en las primeras páginas de búsqueda resultados relevantes y que se adecuen correctamente con su búsqueda. Google fue el primero en resolver este problema asignándole a cada página una importancia determinada y se denominó Page Rank.

1.2.3 Page Rank

El Page Rank fue ideado por Larry Page y Sergei Brin, lo que hizo a Google aplastar a los demás buscadores por sus resultados, que eran más efectivos que los antiguos.

1. *“Es un valor entre 1 y 10 que depende de la cantidad y calidad de las webs que tengan links hacia la web de referencia, así como de sus links internos”.*(Penela 2004)

Uno de los primeros pensamientos que existieron al principio fue que la popularidad de la página recaía completamente en los números de enlaces que enlazan la página. Por lo que existieron webmasters que se dedicaron a crear páginas fantasmas que su único objetivo era enlazar la página que se quería mejorar, cosas como éstas son penalizadas por Google. Pero por suerte no se trata solamente de eso, en esencia y a grandes rasgos se trata de la suma de páginas que enlazan la página a posicionar, ponderada por el índice de popularidad (P) de cada una de ellas, es importante resaltar que en estas ecuaciones no están reflejadas todos los aspectos que conforman la definición completa de Page Rank.

$$P(x) = 1/P(1) + 1/P(2) + \dots + 1/P(n)$$

Pero además de esto se tiene en cuenta la relación que existe entre la página y el término de búsqueda, que se determina mediante el análisis del código de la página y el texto del enlace que nos apunta. Lo que nos lleva a plantear lo siguiente: siendo G la relación.

$$F(x) = G(x) * (1/G(1)*P(1) + 1/G(2)*P(2) + \dots + 1/G(n)*P(n))$$

Otro aspecto a tener en cuenta es que el Page Rank está entre el rango de uno hasta diez, por lo que las páginas que son penalizadas tienen un Page Rank igual a cero.

Con la presencia de la clasificación de las páginas por su popularidad, y el modo de mostrarse en las búsquedas ordenadas de acuerdo a esto, surge el Posicionamiento Web.

1.2.4 Posicionamiento Web

En 1972 fueron escritos una serie de artículos que se titulaban “La era del posicionamiento” por los autores Al Ries y Jack Trout, por lo que se les atribuyó el término “Posicionamiento”. Este término se refería más bien a cómo estaba posicionado el producto en la esfera de publicidad y mercadeo. Al pasar del tiempo el término sigue vigente en el mercadeo pero también se ha extendido a otras ramas como es la de sitios que necesitan estar en las primeras páginas del buscador, y es así como nace el otro término, Posicionamiento Web.

Existen dos tipos de posicionamientos, el Posicionamiento Orgánico o Natural o SEO (Search Engine Optimization), y el otro es el Patrocinado SEM (Search Engine Marketing), los cuales se definen de la siguiente manera:

SEO: “(...) *Están basados en el algoritmo imparcial de los buscadores (...)*”. (Javier Gosende 2006)

SEM: “(...) *Cuya clasificación depende del dinero que se invierta en los anuncios.*” (Javier Gosende 2006)

El presente trabajo se basa en el posicionamiento en buscadores (SEO), del cual muchas personalidades han escrito y entre sus definiciones resaltan:

1. “*Conjunto de procedimientos y técnicas que tienen como finalidad dotar a un sitio o a una página web de la máxima visibilidad en Internet.*” (Codina and Marcos 2005)

2. *“El posicionamiento en buscadores consiste en aplicar diversas técnicas tendentes a lograr que los buscadores de Internet encuadren nuestra página web en una posición y categoría deseada dentro de su página de resultados para determinados conceptos clave de búsqueda.”* (Javier Gosende 2006)
3. *“El posicionamiento en buscadores es a la vez arte y técnica: construir u organizar páginas web de forma que aparezcan bien situadas en los buscadores. Cada buscador utiliza un algoritmo de indexación diferente que es permanentemente actualizado y mejorado. Estos algoritmos son considerados secretos industriales por las empresas que los crean. Esto hace que el posicionamiento en buscadores sea un auténtico ejercicio de ingeniería inversa.”* (Ugarte 2002)

A pesar de estas definiciones técnicas se considera que no solo consiste en la parte del código de la página, pues el diseño de ésta además de amigable para el buscador debe ser amigable para el usuario. Logrando así un aumento del tráfico de usuarios que visiten la página y de la cantidad de páginas que se puedan enlazar.

Como se puede apreciar cuando un webmaster se dedica al Posicionamiento SEO, se tiene que centrar en los algoritmos de los buscadores en los que quiere posicionarse, es evidente que nunca se conoce a ciencias ciertas el algoritmo exacto de cada buscador, y normalmente se determina realizando consultas al buscador y por los resultados se definen varios aspectos que pudo tener en cuenta para posicionar una página determinada. En cambio el webmaster que se dedica al Posicionamiento SEM solo tiene que pagar para que el buscador lo posicione en los enlaces patrocinados.

Para que una página esté bien posicionada existen varios aspectos que se tienen que tener en cuenta en el código de la página. Entre esos aspectos se encuentran las palabras claves, son palabras que definen el sitio web, y cuando se hace un buen análisis y recolección de ellas pueden llegar a ser las que teclean las personas en la búsqueda. Para que sean efectivas tienen que ser bastante usadas en el lenguaje del usuario, tener un número bajo de competidores y que estén bien relacionadas con el tema de la página.

El título es importante también; se recomienda que en él esté presente la palabra clave, sea corto y se relacione con su página. Unas de las cosas más importantes es el intercambio de links, lo cual ayuda a

la hora de asignar a la página su popularidad y además es una forma en que los buscadores la indexen a su base de datos. Esto pasa porque existen dos vías para indexarse en la base de datos de los buscadores, una mediante el alta en buscadores que es la vía larga, para ello se tiene que llenar un formulario y esperar meses para que el buscador la indexe en su base de datos, y la otra es la que se mencionó antes, porque al tener sitios que enlazan la página, el robot pasa por ellos y al encontrarse dicha página la indexan rápidamente a su base de datos.

Otro aspecto fundamental en la página son las metatags, una gran cantidad de arañas se fijan en estas etiquetas, pues indican a los buscadores la temática de la página y las palabras de búsquedas que mejor la definen, en ellas también se recomienda incluir las palabras claves.

Igualmente se tiene que tener en cuenta la presencia del mapa de navegación en la página, pues mejora la indexación de muchos sitios. Además, facilita la interacción con el sitio desde el punto de vista ergonómico para el usuario, pues muestran la arquitectura de las páginas de un sitio web y permiten orientar la navegación a través del sitio.

Para entender el Posicionamiento Web del tipo SEO, es necesario conocer los procesos de recuperación de información de los buscadores. Primeramente el sistema crea el índice, como ya se conoce esto clasifica la página, luego cuando un usuario realiza la búsqueda el sistema compara la consulta con el fichero invertido y de acuerdo a varios elementos se le da una importancia a los términos que forman parte de este fichero.

Entre los elementos anteriormente planteados se encuentran: la relevancia, frecuencia, recuento y peso.

Estos conceptos son una guía para el webmaster, porque a la hora de redactar su texto tiene que tener en cuenta la cantidad de veces que escriba una palabra, la cantidad de palabras en el texto completo y la posición en el documento.

La frecuencia se mide por la cantidad de veces que aparezca el o los términos de la búsqueda en el texto. Por otro lado está el recuento que se determina mediante el total de palabras en el documento, sin tener en cuenta las palabras del código de la página. Los conceptos anteriores ayudan a definir el peso, pues está dado por la siguiente fórmula:

$$P = n * F / R$$

Donde n es el número de palabras que conforman la consulta, F es la frecuencia y R el recuento.

La relevancia mide la distancia que hay entre el comienzo del área y la posición de la palabra. Pero no siempre el buscador lee el texto de la misma forma en que lo lee una persona, ejemplo de esto es el conocido Google, que deja las tablas para leerlas al final.

Lo explicado anteriormente no es la última palabra en cuanto al Posicionamiento Web, lo cual incluye que las fórmulas anteriores son acercamientos a lo que en realidad los buscadores tienen en cuenta, pues los algoritmos de éstos no se facilitan, porque si fuera de otra forma no serían los primeros en las competencias. Es por esto que se hacen estudios sobre la base de las búsquedas y se van sacando las conclusiones anteriores.

1.3 Trabajos relacionados con el tema en el ámbito nacional e internacional.

En la actualidad existen varias herramientas que permiten a los webmasters desarrollar sus páginas teniendo en cuenta las estrategias del Posicionamiento Web.

Las siguientes tienen funcionalidades semejantes a la herramienta a desarrollar:

1.3.1 Promotor Web (Dynamic Submission 7.0):

Este software de promoción de sitios web envía el sitio que se desea posicionar a los buscadores de forma automática, donde solo es necesario introducir la información de éste. El programa crea páginas optimizadas para buscadores con las palabras claves del sitio, las cuales están diseñadas para obtener buenas posiciones en buscadores y actuar como puertas de entrada a la página principal.

También realiza un cálculo de la popularidad de enlaces de la página y además cuenta con un generador de palabras clave, que hace una lista de sugerencias basada en las palabras clave de la página a posicionar.

1.3.2 Active Link Exchange:

Active Link Exchange es una herramienta que ayuda a construir una red de intercambio de enlaces, incrementando así la popularidad y beneficiando el tráfico, factores estos muy importantes para alcanzar una buena posición en los buscadores.

El software convierte en una tarea sencilla la consecución de enlaces entrantes hacia el sitio web. Pero no se tiene la seguridad de que no se ha caído en deficiencias que pueden ser penalizadas por el buscador, un ejemplo podría ser estar en una “granja de enlaces”.

1.3.3 Active Web Traffic:

Active Web Traffic es una solución de promoción web que incluye: un módulo de envío el cual da de alta el sitio web a más de cien buscadores, directorios y páginas de enlaces, y cuenta además con un generador de palabras claves.

También posee un analizador de la densidad de palabras clave, el cual analiza la página web comparándola con páginas de la competencia y utilizando la misma fórmula analiza cómo obtener un primer puesto en los buscadores. Además comprueba los enlaces que apuntan hacia la página web y cómo afecta al ranking.

1.3.4 Google Monitor Software:

Google Monitor Software es un software gratuito, que al introducirle la URL de un sitio y un conjunto de palabras claves, devuelve un listado ordenado por ranking con todos los sitios que responden a esas palabras claves y da la posición exacta del sitio en ese listado. Es una herramienta práctica ya que no hay que entrar necesariamente al buscador ni recorrer cientos de páginas para encontrar la página a posicionar.

También existen otros que son más simples y gratuitos como:

1.3.5 Herramientas Sencillas

Site Position: Permite saber la popularidad de un sitio.

TopDog Pro: Incrementa el tráfico de la página y permite saber la posición de la misma en diferentes buscadores.

Rank Meter: Permite saber la posición de una página determinada.

MyAdSystem: Sencillo sistema para gestionar los banners de los sitios web.

Link Popularity Check: Permite saber si una página es popular o no.

Todos los softwares antes mencionados están centrados básicamente en dar de alta automáticamente una página web en miles de buscadores de todo el mundo. Esto se puede tomar como un primer paso de avance para lograr estar indexado en las bases de datos de los buscadores, pero con solo darse de alta en aquellos buscadores que lideran en el mundo como es el caso de Google y Yahoo entre otros, se tiene el problema resuelto puesto que es a través de éstos que se realiza el mayor número de consultas actualmente.

En cambio la herramienta que se desea desarrollar no se enmarca en funcionalidades tan estrictas para el webmaster como los softwares anteriores, donde ya la herramienta toma decisiones que el usuario no sabe a ciencia cierta si en verdad va a solucionar el problema del Posicionamiento Web para su página. Además se enfoca en dar una mayor importancia al análisis lexicográfico del contenido web, al pensamiento del webmaster y a centrarse únicamente en los mejores sitios posicionados en determinados temas de búsquedas.

Como se mencionaba anteriormente en la introducción del presente trabajo, al no existir el hábito del Posicionamiento Web en Cuba, no existen muchas herramientas destinadas a este tema; en la Universidad de la Habana se desarrolló una aplicación vinculada con el Posicionamiento Web, la cual se basa en analizar el código HTML de la página a desarrollar, también muestra los términos de búsqueda más buscados en el último mes y dado un término de búsqueda muestra otros términos asociados en la web.

Por otra parte en la Universidad de las Ciencias Informática (UCI), existen dos trabajos de tesis relacionados con el tema, pero los dos con un enfoque teórico, uno referente a una propuesta de estándares para el posicionamiento de los sitios web, y el otro es una propuesta de una estrategia para posicionar los weblogs cubanos, los cuales se pueden encontrar en la biblioteca digital de la Universidad de las Ciencias Informáticas.

1.4 Metodologías de Desarrollo

Para lograr el desarrollo de la aplicación informática, se necesita su planificación y control, contar con un marco de trabajo claramente definido y estandarizado, que apoye obtener productos que garanticen los requerimientos de calidad, que cumplan con las expectativas del cliente, se desarrollen en el tiempo estimado y los costos presupuestados y para ello se necesita un proceso de desarrollo de software. Una metodología se basa en los procedimientos, técnicas, herramientas y una base documental que les permitan a los desarrolladores realizar un nuevo software y define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo para obtener los distintos productos parciales y finales.

Existen dos tipos de metodologías: las Robustas y las Ágiles, entre los ejemplos del tipo de metodologías Robustas se encuentran: Rational Unified Process (RUP), Microsoft Solutions Framework (MSF) y Métrica 3.0; dentro de las metodologías Ágiles se encuentran: Extreme Programming, Scrum y Feature Driven Development, de estos dos tipos de metodologías las más completas de su categoría son RUP y XP las cuales se analizan en el presente trabajo, para luego escoger la ideal para desarrollar el análisis de la herramienta.

1.4.1 Metodología Rational Unified Process (RUP).

El Proceso Racional Unificado es un proceso de desarrollo de software y es utilizado para realizar el análisis, diseño, implementación, documentación y pruebas a sistemas orientados a objetos.

Está centrado en la arquitectura definida con relaciones concretas entre los diferentes componentes; es guiado por casos de uso, debido a que el desarrollo se guía por satisfacer las necesidades del usuario; es iterativo incremental ya que existen mini proyectos, y estos son una iteración y cada iteración se construye sobre la base de la anterior. Además posee artefactos que son los productos tangibles del proceso, como es el modelo de casos de uso, y roles que son el determinado papel que desempeña cada persona.

También posee un grupo de principios claves, los cuales defienden que un equipo de un proyecto de software debe planificar el desarrollo, debe conocer hacia donde se dirige y debe documentar el proyecto de una manera perdurable y extensible.

Otra de sus características es que presenta cuatro fases las cuales son:

Fase de inicio: Esta fase tiene por finalidad definir la visión, los objetivos y el alcance del proyecto, tanto desde el punto de vista funcional como del técnico, obteniéndose como uno de los principales resultados una lista de los casos de uso y una lista de los factores de riesgo del proyecto.

Fase de elaboración: Esta fase tiene como principal finalidad completar el análisis de los casos de uso y definir la arquitectura del sistema. En esta etapa se busca eliminar los principales riesgos técnicos.

Fase de construcción: Esta fase está compuesta por un ciclo de varias iteraciones, en las cuales se van incorporando sucesivamente los casos de uso, de acuerdo a los factores de riesgo del proyecto. Este enfoque permite por ejemplo contar en forma temprana con versiones del sistema que satisfacen los principales casos de uso.

Fase de transición: Esta fase se inicia con una versión “beta” del sistema y culmina con el sistema en fase de producción.

Cada fase tiene un hito, que no es más que un punto de control de objetivo intermedio antes de que el proyecto finalice, los cuales son:

El hito correspondiente a la fase de Inicio es Objetivos (visión). En la fase de Elaboración se define como hito la Arquitectura, en la fase de Construcción el hito correspondiente es la Capacidad Operacional Inicial, y por último en la fase de Transición el hito es el Release del sistema que se traduce en el producto final.

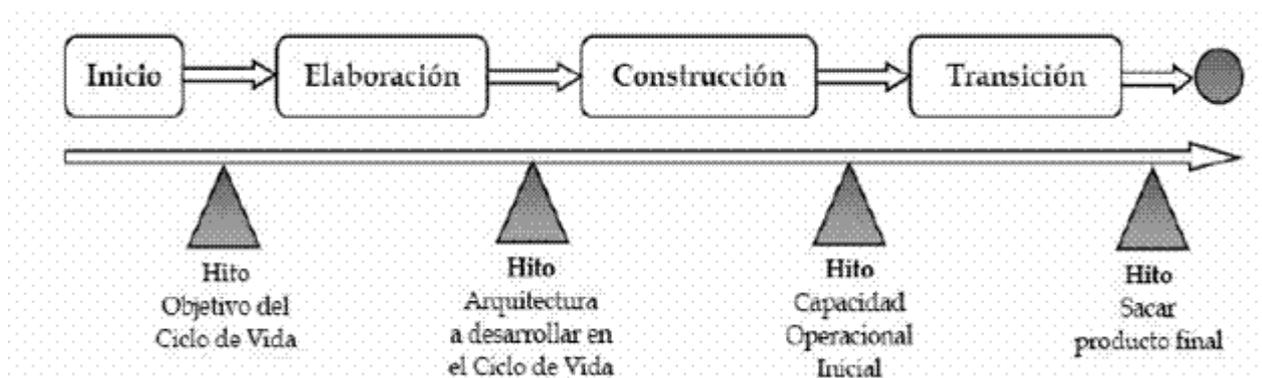


Figura 1: Hitos y Fases de RUP

También cuenta con flujos de trabajo, que no son más que las etapas por las que va pasando el software, y que en cada una de esas fases se realizan actividades y surgen artefactos necesarios para el mismo.

- **Modelamiento del negocio:** Describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización.
- **Requerimientos:** Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.
- **Análisis y diseño:** Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos), por lo que indica con precisión lo que se debe programar.
- **Implementación:** Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación.
- **Prueba (Testeo):** Busca los defectos a lo largo del ciclo de vida.
- **Instalación:** Produce Release del producto y realiza actividades (empaquete, instalación, asistencia a usuarios, etc.) para entregar el software a los usuarios finales.
- **Administración del proyecto:** Involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes.

- **Administración de configuración y cambios:** Describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a: utilización/actualización concurrente de elementos, control de versiones, etc.
- **Ambiente:** Contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto; así como el procedimiento para implementar el proceso en una organización.

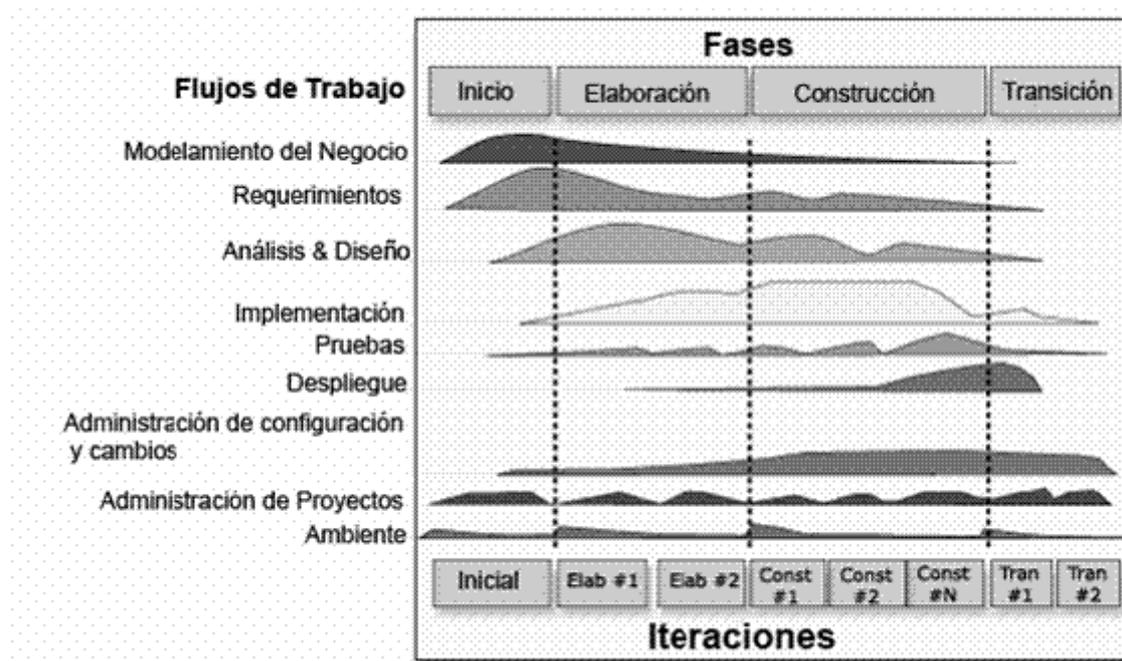


Figura 2: Flujos de trabajo de RUP

1.4.2 Metodología Extreme Programming (XP).

La metodología XP está orientada a la programación y es utilizada para proyectos cortos. Se enfoca en las relaciones interpersonales, velocidad de reacción y orientado al objetivo. Está basada en simplicidad, la comunicación y la realimentación o reutilización del código desarrollado.

Las fases de dicha metodología son:

Planificación

1. Historias de Usuario
2. Plan de Entregas
3. Velocidad de Proyecto
4. Iteraciones
5. Rotaciones
6. Reuniones

Diseño

1. Metáfora del Sistema
2. Tarjetas CRC
3. Soluciones Puntuales
4. Funcionalidad mínima
5. Reciclaje

Desarrollo

1. Disponibilidad del cliente
2. Unidad de Pruebas
3. Programación por parejas
4. Integración

Pruebas

1. Implantación
2. Pruebas de Aceptación

Además cuenta con roles y responsabilidades que tienen cada uno de éstos, los cuales son:

Programador (Programmer)

- Responsable de decisiones técnicas
- Responsable de construir el sistema
- En Xp, los programadores diseñan, programan y realizan las pruebas

Cliente (Customer)

- Es parte del equipo
- Determina qué construir y cuándo
- Escribe varios test funcionales para determinar cuándo está completo un determinado aspecto

Entrenador (Coach)

- El líder del equipo toma las decisiones importantes
- Principal responsable del proceso
- Tiende a estar en un segundo plano a medida que el equipo madura

Rastreador (Tracker)

- Metric Man
- Observa sin molestar
- Conserva datos históricos

Probador (Tester)

- Ayuda al cliente con las pruebas funcionales
- Se asegura de que los test funcionales se ejecutan

La codificación del software se realiza en parejas por lo que la calidad debe ser más elevada por la realización en dúo. Esta metodología intenta minimizar los riesgos a fallos mediante la disposición permanente de un representante del usuario a disposición del equipo de desarrollo.

1.4.3 ¿Por qué RUP?

Luego de analizar las metodologías antes mencionadas se puede apreciar que la metodología RUP es más adaptable al proyecto por ser una metodología orientada a objetos, además la metodología XP no cuenta con la documentación necesaria, solamente se limita a documentar la que le corresponde al código fuente, por ejemplo la documentación de los diagramas UML es prácticamente nula, esto representa una desventaja ya que no presenta un documento que registre la experiencia ganada a través del trabajo, y también entorpece la reutilización de las próximas versiones del software. En cambio RUP se basa en la documentación de todo el trabajo desarrollado, ejemplo de esto es que desarrolla todos los diagramas UML, logrando así que las próximas versiones del software puedan

reutilizar, sin problemas mayores, aspectos de la primera versión a entregar. Un aspecto fundamental es que en la metodología XP no existe un análisis de los requerimientos profundo, lo que trae como consecuencia que durante el desarrollo de la aplicación se tengan que realizar cambios que se podían identificar anteriormente, en cambio RUP si da una mayor importancia a la fase en que se realiza el levantamiento de requisitos, teniendo definidos roles y artefactos que describen el trabajo realizado durante la ingeniería de requisitos.

1.5 El analista de Sistema en RUP

En un tiempo se llegó a pensar que los usuarios sabían exactamente lo que querían, pero al paso del tiempo con la experiencia se concluyó que no tienen una visión general de lo que va a realizar, ni saben que parte de su trabajo es la que se puede automatizar por lo que se consideran una fuente imperfecta de información.

Es por esto que aparece el analista, para poder recoger los requisitos del software, y tratar que sean lo más global y específicos posible. Si se realiza un mal análisis o no se realiza, es muy probable que la construcción del software aunque sea elegante no resuelva correctamente el problema, lo que conlleva a un tiempo y dinero perdido, frustración personal y clientes descontentos, por esto RUP considera al analista de sistema indispensable para la calidad del software.

Las responsabilidades del analista de sistemas de acuerdo a esta metodología son:

Realizar la modelación del negocio o del dominio, en este caso se realizará un modelo de dominio porque no existe un proceso definido donde encontrar actores y trabajadores del negocio.

Confeccionar el documento de requisitos funcionales y no funcionales del sistema, para el desarrollo del presente trabajo se realizarán varias reuniones con el cliente para saber sus necesidades y detectar los requisitos funcionales y no funcionales del sistema.

Delimitar el sistema mediante los actores y casos de uso, teniendo en cuenta que el modelo de caso de uso sea completo y consistente, ya que RUP está guiado por casos de uso, en el presente trabajo se definirán los casos de uso de acuerdo con las necesidades planteadas por el cliente.

Describir cada caso de uso teniendo en cuenta las acciones del actor y del sistema, en este caso se realizarán todas las descripciones de todos los casos de uso con las descripciones de flujos alternos correspondientes a cada uno.

La metodología RUP plantea que para que exista una representación efectiva del software debe existir un buen análisis de requisitos, por lo que el analista de sistemas es un factor fundamental en el desarrollo del software.

1.6 Lenguaje de Modelado.

1.6.1 Unified Modeling Language (UML).

La comunicación entre los integrantes en un desarrollo de software es fundamental. El desarrollo de sistemas era una propuesta al azar, se generaban análisis de requerimientos en un tipo de notación que solo la persona que lo confeccionaba lo entendía, es decir que ni el cliente ni desarrolladores comprendían a fondo el producto. Hoy día es necesario contar con un plan bien definido y argumentado, se tiene que organizar los procesos de diseño, de tal forma que los desarrolladores y clientes y otras personas involucradas en el desarrollo de los sistemas lo comprendan, con el UML se puede lograr tal objetivo.

El Lenguaje Unificado de Modelado (UML), es el Lenguaje de Modelado de sistemas de software más conocido y utilizado en la actualidad, además de ser el que se imparte en la Universidad de las Ciencias Informáticas. Es un lenguaje gráfico para visualizar, construir especificar y documentar un sistema de software. Consta de diagramas de clases, de objetos, de casos de uso, de estados, de secuencias, de actividades, de colaboraciones, de componentes, de distribución y otros. Permite modelar sistemas orientados a objetos ya que modela la Abstracción, Herencia, Polimorfismo, Encapsulamiento o encapsulación, Envío de mensajes, Asociaciones y Agregación.

Uno de sus objetivos principales es posibilitar el intercambio de modelos entre distintas Herramientas CASE orientadas a objetos. UML intenta solucionar el problema de propiedad de código, al implementar un Lenguaje de Modelado común para todos los desarrollos.

1.7 Herramientas CASE.

1.7.1 Visual Paradigm.

Visual Paradigm (VP) utiliza UML como Lenguaje de Modelado, es una Herramienta CASE poderosa y fácil de usar.

Permite representar todo tipo de diagramas UML para las distintas fases como la captura de requisitos, análisis, diseño e implementación, permite además generar códigos, aplicar ingeniería inversa en los lenguajes Java, C++, CORBA IDL, PHP, XML, Ada y Python. Además la generación de código apoya C#, Visual Basic .NET, Object Definition Language (ODL), Flash, ActionScript, Delphi, Perl, Objective-C, y Ruby.

Mediante esta herramienta se puede desarrollar un producto de calidad teniendo el diseño centrado en casos de usos y enfocado al negocio. Facilita la comunicación ya que utiliza un lenguaje estándar común a todo el equipo de desarrollo. También posee disponibilidad de múltiples versiones, en múltiples plataformas y es fácil de instalar y actualizar. Presenta la posibilidad de la interoperabilidad con otras aplicaciones como es el Rational Rose.

Tiene disponible distintas versiones: Enterprise, Professional, Standard, Modeler, Personal y Community (que es gratuita). También facilita licencias especiales para fines académicos.

1.7.2 Rational Rose Enterprise Edition.

Rational Rose Enterprise Edition es una de las herramientas más poderosas para el moldeamiento visual. Se utiliza para modelar un sistema antes de desarrollarse. Permite crear y refinar, logrando así un modelo completo que representa el dominio del problema y el sistema de software. Permite especificar, analizar y diseñar el sistema antes de codificarlo.

Con dicha herramienta se puede generar código en distintos lenguajes de programación a partir de un diseño en UML y proporciona mecanismos para realizar la denominada ingeniería inversa, es decir, a partir del código de un programa podemos obtener información sobre su diseño. Una de las deficiencias que posee este tipo de herramienta es que no es soportada en el Sistema Operativo

Linux, es decir no es multiplataforma, otra deficiencia es que es propietaria ya que se tiene que pagar por su licencia para poder trabajar.

1.7.3 ¿Por qué Visual Paradigm?

Como se pudo apreciar Visual Paradigm es una herramienta multiplataforma, algunas de sus versiones son gratuitas en cambio Rational Rose es una herramienta muy usada pero es software propietario y no es multiplataforma. Entre otras razones por las que se seleccionó a Visual Paradigm como herramienta CASE para el desarrollo del modelado, están: es una herramienta que permite el modelado orientado a objetos y está orientada al Lenguaje de Modelado seleccionado anteriormente, UML.

1.8 Tecnología.

1.8.1 Framework 2.0

La herramienta a desarrollar basa su funcionamiento en utilizar información de la Red de redes para mostrar al usuario un resultado final. Por tal motivo se escoge el .Net Framework el cual cuenta con bibliotecas que facilitan el intercambio de información a través de la red, ayudando en la rapidez de elaboración y funcionamiento eficiente del sistema.

Este componente está compuesto por dos elementos fundamentales:

- *Common Language Runtime* (Lenguaje Común de Ejecución).
- *Un conjunto unificado de bibliotecas de clases*, entre las cuales se pueden utilizar en la herramienta a desarrollarse las bibliotecas *ADO.NET* para el acceso a datos sin rigidez, *Windows Forms* para aplicaciones cliente inteligentes.

A continuación se muestra la Biblioteca de clases del Framework.

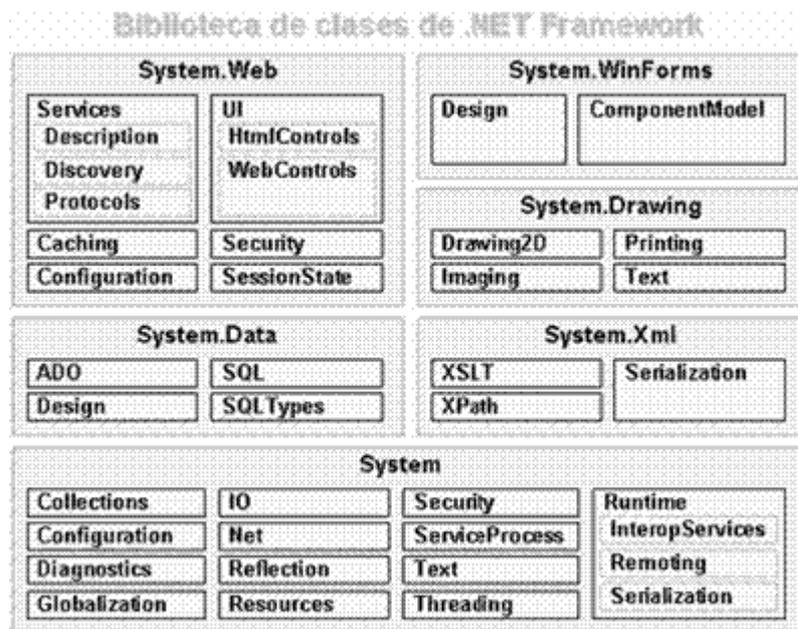


Figura 3: Biblioteca de clases de .Net Framework

En la herramienta a utilizar se necesitan entre otras las siguientes clases porque agilizan el trabajo, por ejemplo para manejar ficheros se utiliza el espacio de nombres System.IO, si lo que se quiere es obtener información de una fuente de datos se utilizará el espacio de nombres System.Data, y si se quiere consumir un servicio web se utiliza el espacio de nombres System.Net entre otras.

Dentro de los lenguajes que soporta .Net Framework se encuentran:

Lenguajes de programación compatibles		
APL	Fortran	Pascal
C++	Haskell	Perl
C#	Lenguaje Java	Python
COBOL	<i>Microsoft JScript®</i>	RPG
Component Pascal	Mercury	Scheme
Curriculum	Mondrian	SmallTalk
Eiffel	Oberon	Standard ML
Forth	Oz	<i>Microsoft Visual Basic</i>

Figura 4: Lenguajes soportados por .Net Framework

Luego de conocer que es indispensable la utilización de este componente en el trabajo a realizar se tuvo que elegir entre los lenguajes anteriormente expuestos en la Figura 4; teniendo en cuenta las características de los desarrolladores dentro del proyecto y tomando en consideración que no se disponía de tiempo para realizar una capacitación dentro del equipo de trabajo, se llegó a la conclusión que poseen mayor dominio del lenguaje C# por lo que se escoge este lenguaje para desarrollar la herramienta.

1.8.2 C#

Actualmente C# se encuentra entre los 10 lenguajes más utilizados. A pesar de su corta historia, ha recibido la aprobación del estándar de dos organizaciones: en el 2001 se aprueba el ECMA y en el 2003 el ISO.

Es un lenguaje de programación simple pero eficaz, evolucionó de los lenguajes C y C++. Utiliza muchas de las características de C++ en las áreas de instrucciones, expresiones y operadores.

Presenta considerables mejoras e innovaciones en áreas como seguridad de tipos, control de versiones, eventos y recolección de elementos no utilizados (liberación de memoria).

Proporciona acceso a los tipos de API más comunes: .NET Framework la cual es la que se utiliza en el desarrollo de la herramienta, COM, Automatización y estilo C. Asimismo, admite el modo unsafe, en el que se pueden utilizar punteros para manipular memoria que no se encuentra bajo el control del recolector de elementos no utilizados.

La sintaxis de C# simplifica muchas de las complejidades de C++ y, a la vez, ofrece funciones eficaces tales como tipos de valores que aceptan valores NULL, enumeraciones, delegados, métodos anónimos y acceso directo a memoria, que no se encuentran en Java. C# también admite métodos y tipos genéricos, que proporcionan mayor rendimiento y seguridad de tipos, e iteradores, que permiten a los implementadores de clases de colección definir comportamientos de iteración personalizados que el código de cliente puede utilizar fácilmente.

Como lenguaje orientado a objetos, C# admite los conceptos de encapsulación, herencia y polimorfismo. Además de estos principios básicos orientados a objetos, C# facilita el desarrollo de componentes de software a través de varias construcciones de lenguaje innovadoras, entre las que se incluyen:

- Firmas de métodos encapsulados denominadas delegados, que permiten notificaciones de eventos con seguridad de tipos.
- Propiedades, que actúan como descriptores de acceso para variables miembro privadas.
- Atributos, que proporcionan metadatos declarativos sobre tipos en tiempo de ejecución.
- Comentarios en línea de documentación XML.

Cuenta con un proceso denominado interoperabilidad, el cual permite que los programas de C# realicen prácticamente lo mismo que una aplicación de C++ nativa. C# admite incluso el uso de punteros y el concepto de código "no seguro" en los casos en que el acceso directo a la memoria es absolutamente crítico.

Microsoft C# es un nuevo lenguaje de programación diseñado para crear un amplio número de aplicaciones que se ejecutan en .NET Framework. Supone una evolución de Microsoft C y Microsoft C++; el código creado mediante C# se compila como código administrado, lo cual significa que se beneficia de los servicios de Common Language Runtime.

Compila a código intermedio (CIL) independiente del lenguaje en que haya sido escrita la aplicación e independiente de la máquina donde vaya a ejecutarse. Otras de las facilidades que brinda son:

- Recolección de basura automática.
- Eliminación del uso punteros, en C# no se necesitan.
- No hay que preocuparse por archivos de cabecera ".h".
- No importa el orden en que hayan sido definidas las clases ni las funciones.
- No hay necesidad de declarar funciones y clases antes de definir las.
- No existen las dependencias circulares.
- Soporta definición de clases dentro de otras.
- No existen funciones, ni variables globales, todo pertenece a una clase.
- Todos los valores son inicializados antes de ser usados (automáticamente por defecto, o manualmente desde constructores estáticos).

1.8.3 Visual Studio 2005 .NET

En las secciones pasadas se definió que se utilizará el componente .Net Framework, por tal motivo se tiene que utilizar como herramienta en que programen los desarrolladores el Visual Studio 2005 .Net, la anterior afirmación se debe a que el Framework solo es compatible con la herramienta seleccionada por ser elementos pertenecientes de *Microsoft .NET*; el cual es un programa de software que conecta información, usuarios, sistemas y dispositivos, que además incluye clientes, servidores y herramientas para programadores.

Visual Studio .NET es un conjunto de aplicaciones que posibilitan la creación tanto de aplicaciones de escritorio como de aplicaciones web. Genera aplicaciones de escritorio de alto rendimiento, y utilizando las eficaces herramientas de desarrollo basado en componentes y otras tecnologías de Visual Studio, se puede simplificar el diseño, desarrollo e implementación en equipo.

Con Visual Studio 2005, los desarrolladores profesionales pueden:

- Disfrutar un entorno de desarrollo altamente productivo con diseñadores visuales, lenguajes de programación y editores de código mejorados.
- Desarrollar y depurar aplicaciones multicapa de servidor desde un mismo entorno unificado de desarrollo (Integrated Development Environment - IDE).

- Construir soluciones para SQL Server 2005 utilizando herramientas visuales integradas de diseño de bases de datos e informes.
- Crear herramientas que extienden el IDE de Visual Studio usando el SDK de Visual Studio.
- Crear aplicaciones de línea de negocio utilizando Visual Basic, C#, C++ y J#.
- Realizar aplicaciones para Windows, la web y dispositivos móviles desde un mismo entorno unificado de desarrollo.
- Construir aplicaciones cliente/servidor usando servicios web e integrando herramientas de diseño para acceder a datos remotos.
- Usar SQL Reporting Services para resumir, tabular y mostrar los resultados.
- Desarrollar aplicaciones web departamentales con completas herramientas de depuración, pre-compilación de sitios web y comprobación de accesibilidad 508/WCAG.

Esta herramienta posibilita la definición de clases parciales, facilitando dividir el código de una clase en varios ficheros, que al compilarse se unen automáticamente para formar la clase completa. Permitiendo además que varios programadores puedan trabajar sobre diferentes partes sin conflictos de varias versiones.

Permite además crear clases que pueden recibir tipos como parámetros, admitiendo indicar los tipos que va a manejar la clase en cada instancia en particular, esto ahorra estar haciendo conversiones continuamente.

En esta herramienta aparecen componentes que permiten manejar visualmente el entorno de vinculación para el acceso a datos, algo que se debía realizar escribiendo el propio código, lo que resultaba complejo y laborioso y que ahora es mucho más sencillo.

También es posible crear una capa de acceso a datos de forma rápida y visual, con la capacidad de sincronizar con la base de datos en cualquier momento volviendo a ejecutar el asistente, sin que esto signifique perder la posibilidad de añadir el propio código cuando se desee, sin temor a que desaparezca al emplear los asistentes.

1.9 Conclusiones

En este capítulo se definieron aspectos fundamentales para entender el Posicionamiento Web. Además se mencionaron las principales características de las metodologías y de las herramientas posibles a utilizar. Luego de analizar sus ventajas y desventajas se determinó que se utilizarían la Metodología de Desarrollo RUP junto con el Lenguaje de Modelado UML y se definió trabajar en la Herramienta CASE Visual Paradigm por sus ventajas de ser multiplataforma y no ser un software propietario. Se escogió como lenguaje para programar el C# pues además de ser orientado a objetos y fácil de implementar el equipo de desarrollo del software posee mayor dominio en este lenguaje, comparándolo con otros que soporta el Framework 2.0. Como herramienta para el desarrollarlo se escogió Microsoft Visual Studio 2005 pues brinda facilidades óptimas para el trabajo con el Framework 2.0 ya que éste contiene bibliotecas que facilitan el trabajo que se debe realizar. Además se analizaron diferentes herramientas y trabajos tanto nacionales como internacionales referentes al tema del Posicionamiento Web que tuvieran alguna relación con el trabajo a desarrollar.

CAPÍTULO 2: ANÁLISIS DEL SISTEMA

2

2.1 Introducción

Antes de llevar a cabo un proyecto, se necesita entender a fondo lo que se va a desarrollar, y definitivamente ese es el objetivo de este capítulo. Para lograr este objetivo se debe investigar cual es el comportamiento del webmaster a la hora de posicionar una página y así definir junto con el usuario los requisitos del software, y junto con esto entender el funcionamiento del mismo. También se tiene que lograr una comunicación fluida entre los desarrolladores y el cliente, para ello se desarrollarán los artefactos dentro de las responsabilidades del analista de sistemas.

Lo primero que se tiene que tener en cuenta antes de desarrollar cualquier artefacto, es el objetivo de la herramienta. El cual consiste en que el webmaster analice la competencia para luego trazarse una estrategia, en base a las características que pueda analizar.



Figura 5: Objetivo de la Herramienta

2.2 Modelo de dominio.

El modelado de dominio permite hacer un correcto levantamiento de requisitos, su objetivo es comprender y describir las clases más significativas dentro del contexto del sistema, lo cual permite definir los procesos y roles más significativos. Esto ayuda a los usuarios, desarrolladores y clientes a tener un vocabulario común, contribuyendo con la aceleración de la construcción de la herramienta. Debido a la sencillez del sistema y a la falta de un proceso de negocio bien definido se decidió desarrollar un modelo de dominio, que se ejemplifica a continuación, además de un Glosario de Términos que se desarrollará mas adelante.

Consulta: Es la acción que se realiza cuando se va a realizar una búsqueda, está formada por una o por varias palabras que sirven para delimitar la información que se desea analizar, y consiste en una solicitud de información a una base de datos, en este caso a la del buscador.

Buscador: Es un sistema que dada una consulta, devuelve una lista de páginas relacionadas con dichas palabras.

Página Web: Es una fuente de información que está identificada por una única URL, en este caso representa las URLs de la competencia que se quieren analizar.

Webmaster: Es el responsable del mantenimiento de una página web, en este caso es el único usuario que interactúa con el sistema.

Proyecto: Representa la información que guarda el webmaster referente al análisis de las páginas.

Características: Representan los diferentes criterios en la página web que necesita el webmaster para poder analizar y tomar una decisión para lograr una mejor posición en los resultados de búsqueda.

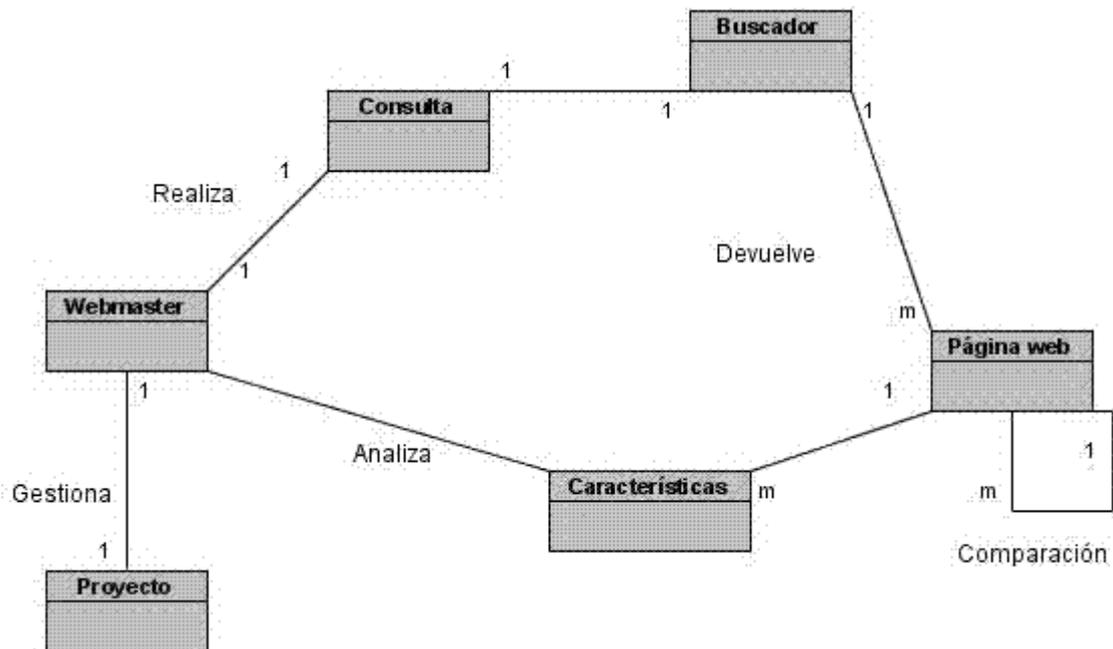


Figura 6: Modelo de Dominio

2.3 Requerimientos del sistema.

Para el desarrollo de cada sistema tiene que existir un cliente que necesite de su existencia, y para ello éste tiene que definir las funcionalidades que debe cumplir la aplicación. Para que los desarrolladores entiendan lo que tienen que producir, y que tengan una visión común junto al cliente existe el levantamiento de requisitos. Pero esta tarea es algo complicada, ya que a veces el cliente no sabe lo que quiere, o varían los requisitos con el tiempo. Es por esto que la ingeniería de requisitos se ha trazado técnicas para facilitar el trabajo.

Entre las características que debe cumplir un buen levantamiento de requisitos están: que debe especificar las funcionalidades, interfaz, rendimiento y restricciones de diseño. Además tienen que ser completos, no ambiguos y verificables.

2.3.1 Estrategia de captura de requisitos utilizada.

Para realizar y guiar el levantamiento de requisitos se utilizó la técnica de la entrevista, la que ayudó a entrevistarse con el cliente para tener bien definido lo que éste desea. Luego se utilizó la técnica de tormenta de ideas, la cual se llevó a cabo entre el equipo de trabajo, para analizar otras funcionalidades posibles y sugerencias que se le podían plantear al cliente. Durante de esta tormenta de ideas se utilizó la técnica de introspección, que consiste en ver las cosas desde el punto de vista del cliente. Otra técnica que se utilizó fue la de los casos de uso, la que consiste en llegar al requerimiento basándose en las funciones del sistema.

2.3.2 Tipos de requerimientos más comunes.

1. Requerimientos de negocio

Los requerimientos del negocio son las actividades esenciales de una empresa, estos son derivados de los objetivos del negocio. Estos requerimientos nos permiten vislumbrar el contexto general del entorno alrededor de nuestro software.

2. Requerimientos de entorno

Se considera el entorno físico y las condiciones del entorno social y cultural donde el software o sistema será usado.

3. Requerimientos de alto nivel o nivel del sistema

Para permitir la comprensión de un sistema se describen los requerimientos del sistema, comprenden los requerimientos de mayor importancia y la visión del cliente.

4. Requerimientos desconocidos

Son aquellos requerimientos que no se logran clasificar desde le inicio del proyecto, a veces se presentan requerimientos reales desconocidos que deben ser incluidos en esta categoría.

5. Requerimientos Funcionales

Los requerimientos funcionales son una categoría importante de los requerimientos reales, describen lo que el sistema debe hacer. Son llamados comúnmente requerimientos de comportamiento o de operación.

6. Requerimientos no funcionales

Referencia las especificaciones del sistema como sus propiedades, confiabilidad y seguridad.

7. Requerimientos derivados

Un requerimiento derivado se define como un refinamiento de un requerimiento de alto nivel y las necesidades del sistema.

8. Requerimientos de rendimiento

Este tipo de requerimientos define cómo los requerimientos funcionales se deben ejecutar.

9. Requerimientos de interfaz

Los requerimientos de interfaz analizan e identifican relaciones físicas y funcionales entre elementos del sistema y el entorno del sistema. Es recomendable asignar una persona en el equipo que se encargue de este tipo de requerimientos.

10. Requerimientos de verificación

Los requerimientos de verificación son requerimientos de tipo reales que deben ser satisfechos por la solución del diseño.

11. Requerimientos de validación

Este tipo de requerimientos se implementan en el sistema a entregar.

12. Requerimientos de cualificación

La cualificación hace referencia a la verificación o la validación del rendimiento de un ítem de la aplicación durante la etapa de diseño, prueba y gestión de la configuración.

13. Requerimientos especiales de ingeniería

Hace referencia a atributos de calidad como lo pueden ser:

- Eficiencia.
- Portabilidad.
- Confiabilidad.
- Capacidad.
- Memoria.
- Usabilidad.

14. Requerimientos de soporte de logística

Comprenden las herramientas, los entrenamientos, los procedimientos y facilidades que son usados en el proyecto.

2.3.3 Requerimientos funcionales del software.

R1. Analizar los mejores sitios posicionados.

R1.1. Listar las mejores páginas posicionadas en los buscadores.

R1.2. Brindar información si posee mapa de navegación para cada página.

R1.3. Informar la cantidad de tablas, imágenes y enlaces que posee cada página.

R1.4. Mostrar porcentaje de texto y código embebido que posee cada página.

R1.5. Mostrar idioma que posee cada página.

R1.6. Mostrar sitios que apuntan a cada página.

R1.7. Mostrar Popularidad de cada página.

R1.8. Mostrar correo de webmasters.

R1.9. Mostrar enlaces rotos en el sitio.

R1.10 Mostrar palabras que más predominan.

R1.10.1 Mostrar si está presente en párrafo o en título.

R1.10.2 Mostrar la frecuencia, peso, relevancia y recuento.

R2. Determinar la consulta formada con las palabras claves que mejor recupera el sitio a posicionar.

R3 .Determinar las palabras más buscadas en uno o varios buscadores dado un término de búsqueda.

R4. Mostrar comparaciones entre páginas.

R5. Crear proyecto.

R6. Modificar proyecto.

R7. Eliminar proyecto.

2.3.4 Requerimientos no funcionales del software.

- Apariencia o interfaz externa.
 - Los usuarios del producto son personas que conocen sobre el tema de Posicionamiento Web, por lo que se hace necesario que el producto sea entendible para ellos, con una interfaz externa amigable y de fácil comprensión, de forma tal que el objetivo de los clientes con el producto pueda conseguirse rápidamente.

- Usabilidad.

- El presente software podrá ser accedido por cualquier webmaster, ya que no tiene ningún tipo de restricción.
 - Tener diseño sencillo.
 - Fácil navegación.
- Implementación
- Lenguajes de programación: C#.
 - Aplicación: Desktop
 - Framework .Net v2.0
 - IDE's de Desarrollo: Visual Studio 2005.
 - Herramienta de Control de Versiones: SubVersion.
- Seguridad.
- Mantener la integridad de la información, es decir que no se pierda durante el almacenamiento.

 - Permitir que cuando se elimine cualquier documento o información pueda existir una opción de advertencia antes de realizar la acción.
- Software
- Debe poseer como Sistema Operativo el Microsoft Windows XP y sus posteriores versiones.
 - Framework .Net v2.0.
- Hardware
- Al menos 128 MB de RAM.

 - Disco duro con capacidad de 500 MB o superior.

 - Conexión Internet.

- Al menos una máquina Pentium(R) III.
- Disponibilidad.
 - El sistema debe de estar disponible las 24 horas del día, para que el webmaster trabaje cuando quiera.

2.4 Ejemplificación de la selección de las características.

Brindar información si posee mapa de navegación para cada página.

La presencia del mapa de navegación mejora la indexación de muchos sitios. Además para el usuario es bastante cómodo, ya que facilitan la interacción con el sitio del punto de vista ergonómico, pues muestran la arquitectura de las páginas y permiten orientar la navegación a través de esta estructura "física".

Informar la cantidad de tablas, imágenes y enlaces que posee cada página.

En ocasiones las personas se centran en decorar la página que se está desarrollando para agradarle al usuario, pero no tienen en cuenta el buscador, el cual al encontrarse con una "carga de imágenes" se limita a leerlas. Por este motivo se hace necesario conocer la cantidad límite de imágenes para que la página sea amigable al buscador.

Para extender el rango de análisis de competencia en el mismo tema, se tiene que tener en cuenta los enlaces que posee dicha página. Así se puede tener acceso no solo a la página que se está analizando en el momento, si no también las que han intercambiado con ésta, hasta estar satisfecho.

El buscador cuando está leyendo el contenido de la página deja para al final el análisis de las tablas y no lee la información dentro de ella como lo hace una persona. Es por esto que se debe tener en cuenta la cantidad límite a emplear en la estrategia a trazarse.

Mostrar por ciento de texto y código embebido que posee cada página.

La presencia de código embebido en la página puede causar problemas en el análisis de la página por parte del buscador. La razón por la cual sucede esto es que lenguajes como el Javascript y el Flash no son interpretados por los buscadores.

Mostrar idioma que posee cada página.

Conociendo el idioma se puede calcular el rango en el que se puede traducir la página para ser más popular, en el tema que se esté abordando, pues puede darse el caso de que un tema específico sea debatido en español en inglés y sin embargo esos términos no sean manejados en japonés.

Mostrar sitios que apuntan a cada página.

Para conocer el ranking de la página influye mucho la cantidad de enlaces que se tengan apuntando a la página, ya que mientras más enlaces apunten a ella, aumenta la posibilidad de que el robot del buscador indexe la página.

Mostrar Popularidad para cada página.

La Popularidad determina la relevancia de la página, que viene dada entre otras cosas por el número de enlaces que la enlazan. Es por esto que se tiene que tener en cuenta la popularidad, para poder escoger con cuales páginas se van a realizar los intercambios de enlaces, porque mientras más popularidad tenga la página web a enlazar aumenta la popularidad del sitio a posicionar.

Mostrar correo de webmasters.

Conocer el correo del webmaster de la página es esencial, porque por esta vía se puede intercambiar links, además de contactar con éste para intercambiar ideas y estrategias de Posicionamiento Web.

Mostrar enlaces rotos en el sitio.

Tener en cuenta los enlaces rotos que posee la página tiene una gran importancia en el Posicionamiento Web, debido a que dificultan la navegabilidad del buscador al recorrerla, también influye en el número de visitas a la página.

Mostrar palabras que más predominan.

Mostrar si está presente en párrafo o en título.

Mostrar la frecuencia, peso, relevancia y recuento.

Se tiene que tener bien claro como situar las palabras en el documento, con la frecuencia que se debe hacer, cuan separado del principio debe estar una palabra. Lo más importante que se tiene que tener en cuenta es que todo texto que se quiera publicar debe de tener calidad no solo en la ubicación de las palabras si no en el contenido del texto.

2.5 Definición de los actores.

Tabla 1: Actor del Sistema.

Actor	Descripción
Webmaster.	Es la persona encargada de darle mantenimiento a una página web, está inmerso en el mundo del Posicionamiento Web y por tal motivo es el único usuario que interactúa con el sistema.

2.6 Diagrama de Casos de Uso del Sistema

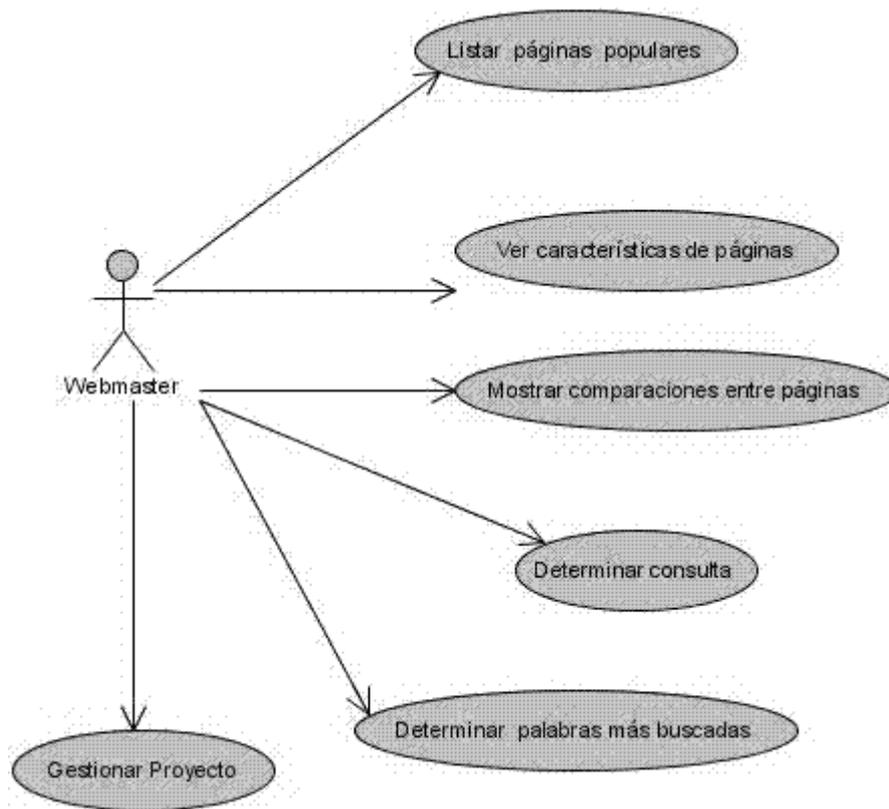


Figura 7: Diagrama de Casos de Uso del Sistema.

2.7 Descripción de los casos de uso del sistema.

Tabla 2: Descripción del Caso de Uso_ Determinar palabras más buscadas

Caso de Uso:	Determinar palabras más buscadas
--------------	----------------------------------

Actores:	Webmaster.	
Resumen:	El caso de uso se inicia cuando el webmaster introduce un término relacionado con el tema de su sitio, luego el sistema procesa la información y muestra el resultado al usuario.	
Precondiciones:		
Referencias	R3	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1 El webmaster selecciona la opción de determinar las palabras más buscadas en el último mes.	2 El sistema brinda la posibilidad de entrar el término que el webmaster desee.	
3 El webmaster entra el término.	4 El sistema muestra la información.	
Flujos Alternos		
Acción del Actor	Respuesta del Sistema	
3.1 El webmaster no entra el término.	3.1.1 El sistema muestra mensaje de que debe introducir el término.	
3.2 El webmaster entra el término mal escrito.	3.1.2 El sistema le muestra un mensaje de error de que debe entrar el término correctamente escrito.	
	4 El sistema presenta problemas de conexión y muestra un mensaje de error.	
Poscondiciones	Se muestran las palabras más buscadas.	

Tabla 3: Descripción del Caso de Uso_ Determinar consulta

Caso de Uso:	Determinar consulta.	
Actores:	Webmaster.	
Resumen:	El caso de uso se inicia cuando el webmaster introduce las palabras claves de su sitio, el sistema realiza consultas a determinado buscador con las distintas combinaciones de estas palabras, y luego muestra la combinación que mejor recupera el sitio.	
Precondiciones:	El sitio ya sea reconocido por el buscador.	
Referencias	R2	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1 El webmaster selecciona la opción de Determinar consulta.	2 El sistema brinda la posibilidad de entrar las palabras claves del sitio del webmaster y de especificar buscador.	
3 El webmaster entra las palabras claves de su página y especifica el buscador.	4 El sistema le hace las consultas al buscador y muestra la consulta.	
Flujos Alternos		
Acción del Actor	Respuesta del Sistema	
3.1 El webmaster no entra las palabras claves.	3.1.1 El sistema muestra mensaje de que debe entrar las palabras claves.	

3.2 El webmaster no especifica el buscador.	3.2.1 El sistema muestra un mensaje de que debe especificar el buscador.
3.3 El webmaster entra mal escritas las palabras claves.	3.3.1 El sistema muestra un mensaje de que debe de entrar correctamente las palabras claves.
	4 El sistema presenta error en la conexión y muestra un mensaje de error que no se ha podido conectar.
Poscondiciones	Se muestra la consulta.

Tabla 4: Descripción del Caso de Uso_ Listar páginas más populares

Caso de Uso:	Listar páginas populares.
Actores:	Webmaster.
Resumen:	El caso de uso se inicia cuando al webmaster se le da la opción de hacer la búsqueda por palabras claves o por términos de búsquedas, introduce los datos según haya seleccionado, especifica en que buscador desea realizar el trabajo y entra los datos de conexión, luego el sistema muestra el resultado al usuario.
Precondiciones:	
Referencias	R1.1
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

1 El webmaster selecciona la opción de Listar páginas más populares.	2 El sistema brinda la posibilidad de entrar las palabras claves o de entrar términos de búsquedas del usuario, especificar un buscador determinado y entrar los datos de conexión.
3 El webmaster entra las palabras claves de su página o entra los términos de búsqueda del usuario, selecciona el buscador y entra los datos de conexión.	4 El sistema muestra las páginas.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
3 El webmaster no entra las palabras claves o los términos de búsquedas, no selecciona buscador o no entra datos de conexión.	3.1 El sistema muestra mensaje de que debe entrar las palabras claves o términos de búsqueda, o que debe seleccionar el buscador, o que debe entra los datos de conexión.
	4 El sistema presenta problemas de conexión y muestra un mensaje de error que no se ha podido conectar.
Poscondiciones	Se muestran las páginas populares.

Tabla 5: Descripción del Caso de Uso_ Ver características de páginas

Caso de Uso:	Ver características de páginas.
---------------------	--

Actores:	Webmaster.
Resumen:	El caso de uso se inicia cuando el webmaster selecciona la opción de Ver Características de una página de las que se mostraron en el caso de uso (Listar páginas populares), luego el sistema procesa los datos de dicha página y al final muestra las características.
Precondiciones:	Realización del caso de uso: Listar páginas populares. Selección de una página.
Referencias	R1.2 – R1.10.2
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1 El webmaster escoge la opción de Ver Características.	2 El sistema brinda las siguientes características de cada una de las páginas : .Brindar información si posee mapa de navegación para cada página. .Informar la cantidad de tablas, imágenes y enlaces que posee cada página. .Mostrar por ciento de texto y código embebido que posee cada página. .Mostrar idioma que posee cada página. .Mostrar sitios que apuntan a cada página. .Mostrar Popularidad para cada página. .Mostrar correo de webmaster. .Mostrar palabras que más predominan. .Mostrar si esta presente en párrafo o en título. .Mostrar la frecuencia, entropía, peso,

	relevancia y recuento. .Mostrar enlaces rotos.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Poscondiciones	Se muestran las características.

Tabla 6: Descripción del Caso de Uso_ Mostrar comparaciones entre páginas

Caso de Uso:	Mostrar comparaciones entre páginas.
Actores:	Webmaster.
Resumen:	El caso de uso se inicia cuando el webmaster escoge la opción de la comparación de las páginas, luego el sistema le da la posibilidad de hacer la comparación por característica o general. Después el sistema brinda un resumen gráfico del comportamiento de las páginas a comparar.
Precondiciones:	Realización del caso de uso: Ver características de páginas.
Referencias	R4
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1 El webmaster selecciona la opción de comparación entre páginas.	2 El sistema brinda la posibilidad de escoger las páginas a comparar y por qué categoría comparar.
3 El webmaster escoge las páginas y la forma	4 El sistema analiza cada una de las páginas

en que va a comparar.	y muestra las gráficas de resumen.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
3 El webmaster no escoge las páginas o no especifica la forma en que va a comparar.	3.1 El sistema muestra mensaje de que debe escoger las páginas o de que tiene que especificar la forma de comparación.
Poscondiciones	Se muestra la gráfica de comparación.

Tabla 7: Descripción del Caso de Uso_ Gestionar Proyecto

Caso de Uso:	Gestionar Proyecto
Actores:	Webmaster.
Resumen:	El caso de uso se inicia cuando el webmaster escoge la opción de Gestionar Proyecto, luego escoge la acción a realizar e introduce los datos necesarios. Con esto el sistema realiza la acción seleccionada y termina el caso de uso.
Precondiciones:	
Referencias	R5, R6, R7.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1 El webmaster selecciona la opción de	2 El sistema brinda las opciones de: Crear

Gestionar Proyecto.	Proyecto, Modificar Proyecto y Eliminar Proyecto.
Sección "Crear Proyecto"	
Acción del Actor	Acción del Sistema
1 El webmaster selecciona la opción de Crear Proyecto. 3 El webmaster entra los datos necesarios.	2 El sistema pide los datos del nombre del proyecto y la ubicación donde se guardará. 4 El sistema verifica la información y crea el proyecto.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
3 El webmaster no entra el nombre o la ubicación.	3.1 El sistema muestra mensaje de que debe entrar los datos de nombre y ubicación.
Flujo Normal de Eventos	
Sección "Modificar Proyecto"	
Acción del actor	Respuesta del sistema
1 El webmaster escoge la opción de Modificar Proyecto. 3 El webmaster escoge el proyecto y lo carga. 5 El webmaster modifica los datos necesarios.	2 El sistema brinda la opción de escoger la ubicación donde se encuentra el proyecto, seleccionarlo y cargarlo. 4 El sistema muestra los datos referentes a dicho proyecto. 6 El sistema verifica los datos y luego modifica el respectivo proyecto.
Flujos Alternos	

Acción del Actor	Respuesta del Sistema
	6 El sistema comprueba que existe algún error en los datos entrados y emite un mensaje de error, para que se entren correctamente.
Flujo Normal de Eventos	
Sección “Eliminar Proyecto”	
Acción del actor	Respuesta del Sistema
1 El webmaster selecciona la opción de Eliminar Proyecto.	2 El sistema brinda la opción de escoger el proyecto a eliminar.
3 El webmaster escoge el proyecto a eliminar y ejecuta la acción de eliminar el proyecto seleccionado.	4 El sistema le muestra los datos del proyecto y pide verificación de eliminación.
5 El webmaster confirma que quiere eliminar los datos.	6 El sistema elimina el proyecto seleccionado.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
5 El webmaster escoge la opción de no eliminar el proyecto.	5.1 Culmina el caso de uso sin ejecutar ninguna acción.
Poscondiciones	<p>Se crea el proyecto.</p> <p>Se modifica el proyecto.</p> <p>Se elimina el proyecto.</p>

2.8 Patrones de casos de uso.

2.8.1 ¿Qué es un patrón?

El patrón ofrece orientación a la hora de proceder con el trabajo que se está realizando, teniendo de base la experiencia de desarrollo en ese problema que se plantea. Está compuesto por una descripción del problema, su solución y es identificado por un nombre, también es aplicable en diferentes contextos.

2.8.2 Patrones de casos de uso utilizados.

- El nombre revela la intención. (IntentionRevealingName)

Este patrón se utilizó en la confección de los casos de uso ya que plantea como problema que los nombres genéricos sin sentido, pueden no ubicar las expectativas del lector. Es por esto que es bueno utilizar nombres descriptivos para los casos de uso, ya que revelan la intención de cada caso de uso.

Para esto propone como solución nombrar los casos de uso utilizando un verbo o frase que represente la meta del actor primario.

- Escenario más Fragmentos.(ScenarioPlusFragments)

Este patrón fue puesto en práctica a la hora de realizar la confección de la descripción de los casos de uso, ya que plantea como problema que el lector debe ser capaz de seguir el camino a través del flujo específico o historia en que ellos están interesados, en caso contrario probablemente se frustrarán o extrañarán la información importante.

Es por esto que plantea como solución que se debe escribir los eventos del flujo principal como un escenario simple, sin considerar posibles fallos. Debajo ubicar los fragmentos del flujo que muestran la condición alternativa que puede ocurrir.

- Equipo Pequeño Escribiendo. (SmallWritingTeam)

Este patrón fue puesto en práctica a la hora de distribuir responsabilidades dentro del grupo del trabajo, ya que plantea como problema que utilizar muchas personas para escribir un caso de uso es ineficaz, y los compromisos hechos para alinear los diversos puntos de visión pueden dar lugar a un sistema menos satisfactorio.

Para dicho problema plantea la solución de que se tiene que restringir el número de un equipo de trabajo, a solo dos o tres personas. Ya que para comunicarse y llegar a un acuerdo general es más factible un equipo pequeño.

➤ Audiencia Participativa. (ParticipatingAudience)

Este patrón se utilizó en todo momento, para facilitar el trabajo con el cliente, ya que como problema plantea que no se puede satisfacer la necesidad de los Stakeholders, y como solución plantea que se debe implicar activamente a los clientes y a los Stakeholders en el proceso de desarrollo de los casos de uso, ya que se conoce mejor las verdaderas necesidades.

➤ CRUD

Este patrón consiste en un caso de uso llamado Gestión de Información o CRUD Information, que fusiona las diferentes operaciones que pueden ser realizadas como simples casos de uso tales como: crear, leer, actualizar y eliminar segmentos de información dentro de un caso de uso formando una sola unidad conceptual. Este patrón debe ser utilizado cuando todos los flujos contribuyen al mismo valor de negocio y son todos cortos y simples. CRUD posee ventajas que ayudan a la reducción del tamaño del modelo agrupando las 3 funciones básicas de crear, modificar, y eliminar proyecto en un solo caso de uso, el cual es Gestionar Proyecto, haciendo el modelo más claro para el analista, los desarrolladores y el cliente.

2.9 Conclusiones.

En el presente capítulo se elaboró un modelo de dominio, por la falta de un proceso de software bien definido. Se plantearon los requerimientos funcionales y no funcionales del software que se acordaron con el cliente. Además se modelaron los artefactos correspondientes al desarrollo del trabajo del analista de sistema, entre los cuales están el diagrama de casos de uso del sistema, actores del sistema y la descripción de los casos de uso del sistema.

CAPÍTULO 3: EVALUACIÓN DE RESULTADOS

3.1 Introducción

En el presente capítulo se analizan los resultados obtenidos en los capítulos anteriores. La medición del trabajo se realiza mediante métricas las cuales se definen para la comprensión de los que no sepan su significado. Luego de comprender las métricas que son aplicadas, se plantean los resultados obtenidos después de haberlas aplicado, ya después se analizan los resultados para medir la calidad con que fue realizado el trabajo. Por último se presenta la opinión del cliente, ya que es la palabra final de la calidad del software.

3.2 Métricas

“La medición es el proceso por el cual se asigna una serie de números o símbolos a uno o más atributos de las entidades o procesos bajo consideración; esta asignación se realiza de acuerdo a una serie de reglas definidas y claramente identificadas. Una vez que la asociación evento-valor ha sido realizada, es posible comparar las entidades de interés a través de la comparación de estos valores.”(Andriano Agosto 2006)

Las métricas ayudan a llevar un control de la calidad del trabajo que se está desarrollando. Estimula a mejorar los procesos y con ello el proyecto. Además si no se midieran los trabajos que se realizan, nunca se podría estar seguro de la calidad del trabajo realizado.

La medición puede realizarse mediante métricas de medición directa e indirectas. Las directas están presentes cuando se mide exactamente el objeto o el atributo de la entidad que se quiere medir. Por su

lado las indirectas a la hora de medir un atributo de una determinada entidad, involucran otros atributos e inclusive otras entidades, y no a partir directamente de los valores observados sobre la entidad.

3.2.1 Métricas para el trabajo realizado.

Para evaluar la calidad del trabajo realizado en cuanto a los requisitos, se tendrán en cuenta dos características, las cuales son: Especificidad (ausencia de ambigüedad), y Grado de validación de requisitos; en cuanto al modelo de casos de uso del sistema se necesitó del empleo de un modelo de métricas orientado a objetos, el cual consta de cuatro características a medir, las cuales se explicarán más adelante.

3.3 Especificación de Requisitos

Para medir el grado de especificación de los requerimientos se propone una métrica que se basa en la consistencia de la interpretación de los revisores, para esto es necesario conocer primeramente:

El total de requerimientos viene dado por n_r .

$$n_r = n_f + n_{nf}$$

Donde los requisitos funcionales son n_f , y los no funcionales n_{nf} .

Luego de tener este dato se puede determinar la especificidad (ausencia de ambigüedad) y para ello se plantea:

$$Q = n_{ui} / n_r$$

Donde n_{ui} representa el número de requisitos para los que los revisores tuvieron interpretaciones idénticas, es importante resaltar que cuando más cerca de 1 este el valor Q , menos ambigüedad presentará la especificación.

En el presente trabajo se realizaron dos revisiones para llegar a obtener un buen análisis sin ambigüedad en los requisitos, y para que se reflejaran con más claridad las necesidades del cliente, en las cuales se obtuvieron los siguientes resultados:

Revisión 1: Durante esta revisión se detectaron de los 5 requisitos funcionales uno que no estaba claramente definido, el caso específico es el caso de uso que se refiere a las acciones que se realizan con el Proyecto, ya que solo se reflejaba el Guardar Proyecto teniendo también que crearlo y modificarlo. Por lo que de los 12 requisitos existentes, los revisores tuvieron interpretaciones bastante parecidas en 11, y en uno que se explicó anteriormente no tuvieron concordancia.

$$Q = 11/12$$

$$Q = 0.92.$$

Revisión 2: En esta revisión ya se había solucionado el problema que se detectó en la revisión anterior, se adicionaron dos requisitos más para que quedaran reflejados todas las necesidades del cliente, por lo que ya se cuenta con 7 requisitos funcionales lo cual hace que los revisores hayan tenido interpretaciones iguales para los 14 requisitos.

$$Q = 14 / 14$$

$$Q = 1$$

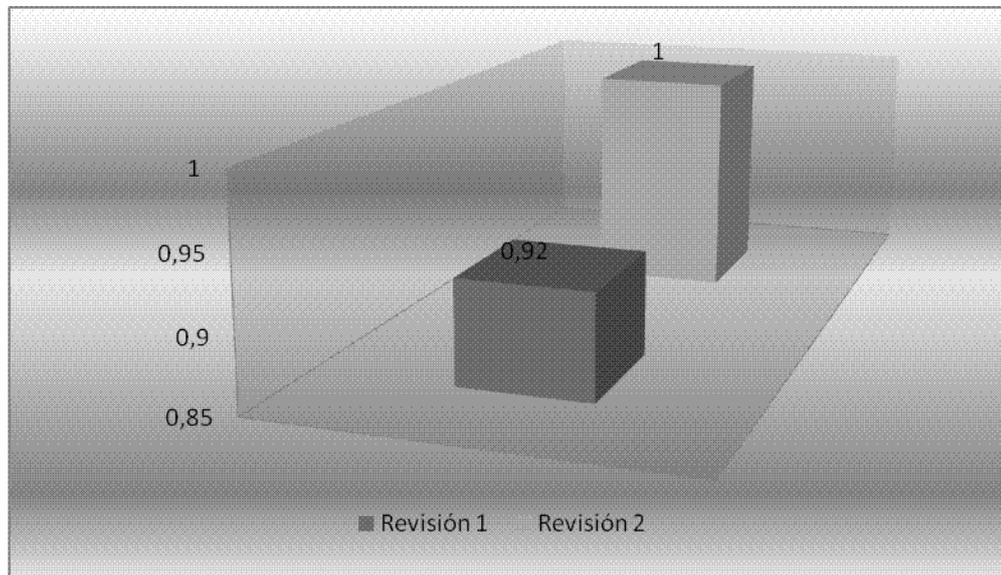


Figura 8: Gráfico de la calidad de especificación de Requisitos

Después de haber analizado las anteriores revisiones se llega a la conclusión que los requisitos presentan un bajo nivel de ambigüedad, pues el valor Q en la primera revisión se acercó bastante al valor 1 y ya en la segunda no presenta ningún nivel de ambigüedad. Lo cual es bastante satisfactorio para el presente trabajo ya que ayuda a que el desarrollo del software no tenga que atrasarse por algún desperfecto en los requisitos.

3.4 Grado de validación de Requisitos.

Esta métrica esta dada por:

$$\mathcal{V} = \mathcal{N}_c / (\mathcal{N}_c + \mathcal{N}_{nv})$$

Donde \mathcal{N}_c es el número de requisitos que se han validado correctamente, y \mathcal{N}_{nv} representa los requisitos que no fueron validados todavía.

Como se apreció anteriormente se realizaron dos revisiones al trabajo realizado, por lo que quedaron validados todos los requisitos tanto funcionales como no funcionales quedando el cliente satisfecho, igualmente los desarrolladores concluyeron que los requisitos se adaptaron también a sus aportes en el proceso, como se muestra a continuación:

$$\mathcal{V} = 12 / (12+0)$$

$$\mathcal{V} = 1.$$

3.5 Algunas Métricas aplicadas en el Diagrama de Casos de Uso del Sistema

Para medir la calidad de la funcionalidad del diagrama de casos de uso del sistema, se utilizó un modelo de métricas orientado a objetos. Éste consta de cuatro atributos que se tuvieron en cuenta, los cuales son: Consistencia, Correctitud, Completitud y Complejidad.

Donde la Completitud consiste en el grado en que se han detallado los casos de uso más relevantes. La Correctitud consiste en el grado en que las interacciones entre actor y sistema soportan el modelo de negocio. La Complejidad viene dada por el grado de presentación de los elementos que describe el contexto y la claridad del sistema. Por otra parte la Consistencia se basa en el grado en que los casos de uso del sistema describen las interacciones entre el usuario y el sistema.

Tabla 8: Atributos para medir calidad de Diagrama de Casos de Uso.

Atributo	Factores
----------	----------

Compleitud	<ol style="list-style-type: none"> 1. ¿Han sido involucradas todas las áreas funcionales a las cuales apoyará el sistema? 2. ¿Han sido definidos todos los roles relevantes de usuario encargados de generar/ modificar o consultar información? 3. ¿Están definidos todos los requisitos que justifican la funcionalidad de los casos de uso? 4. ¿Se presenta una descripción detallada (descripción extendida esencial) de todos los casos de uso del sistema? 5. ¿Están todas las acciones del flujo de eventos redactadas en función del responsable?
Complejidad	<ol style="list-style-type: none"> 6. ¿Los elementos dentro del diagrama están adecuadamente ubicados de manera que facilitan su interpretación?
Consistencia	<ol style="list-style-type: none"> 7. ¿El nombre dado a los casos de uso es una expresión verbal que describe alguna funcionalidad relevante en el contexto del usuario? 8. ¿Representan los casos de uso una interacción observable por un actor? 9. ¿Está adecuadamente redactado (en el lenguaje del usuario) el flujo de eventos? 10. ¿La descripción del flujo de eventos se inicia con la descripción de una acción externa originada por un actor o por una condición interna del sistema claramente identificable? 11. ¿Existe una adecuada separación entre el flujo básico de eventos y los flujos alternos?
Correctitud	<ol style="list-style-type: none"> 12. ¿Existe para cada caso de uso por lo menos un usuario responsable? 13. ¿Representa el caso de uso requisitos comprensibles por el usuario? 14. ¿Las interacciones definidas introducen mejoras al proceso actual?

Revisión 1:

Atributo Completitud.

No se encontraron representados todos los requisitos que justifican la funcionalidad de los casos de uso. De ellos cuatro se encontraban bien definidos, lo cual representa un 33,3% de los doce requisitos tanto funcionales como no funcionales.

No han sido definidos correctamente los roles relevantes de usuario que interactúan con la herramienta, ya que se tiene definido un actor referente al sistema buscador, pero este no se beneficia en ningún momento del sistema, mas bien le brinda información, lo que representa un 50%.

No han sido involucradas todas las áreas funcionales a las cuales apoyará el sistema ya que el sistema tiene que realizar otras dos operaciones con el Proyecto además de la que esta reflejada en el diagrama, lo cual representa un 71,4%.

Se cuenta con una descripción detallada de todos los casos de uso del sistema y están todas las acciones del flujo de eventos redactadas en función del responsable. Teniendo en cuenta la calidad de las características se concluye que el atributo Completitud tiene un 81% de calidad.

Atributo Consistencia

Para lograr la consistencia se les dio a cada caso de uso un nombre que describía su funcionalidad, cada caso de uso esta relacionado al menos con un actor, los casos de uso describen la funcionalidad del sistema, además están bien definidos los flujos básicos y alternos. Por lo que se concluye que representa un 100% de la calidad.

Atributo Complejidad

No están adecuadamente ubicados los elementos dentro del diagrama ya que se tenían dos relaciones de inclusión entre tres casos de uso, lo cual dificulta la comprensión del mismo, esto representa un 90% de calidad.

Atributo Correctitud

Todos los casos de uso poseen al menos un usuario responsable, no existe ninguno que no sea iniciado por un actor y representan requisitos comprensibles por el usuario, por otro lado aunque no existe un proceso de negocio definido, se puede afirmar que los casos de uso se ajustaron al levantamiento de requisitos además de aportarle nuevas ideas al cliente. Lo que representa un 100% de calidad.

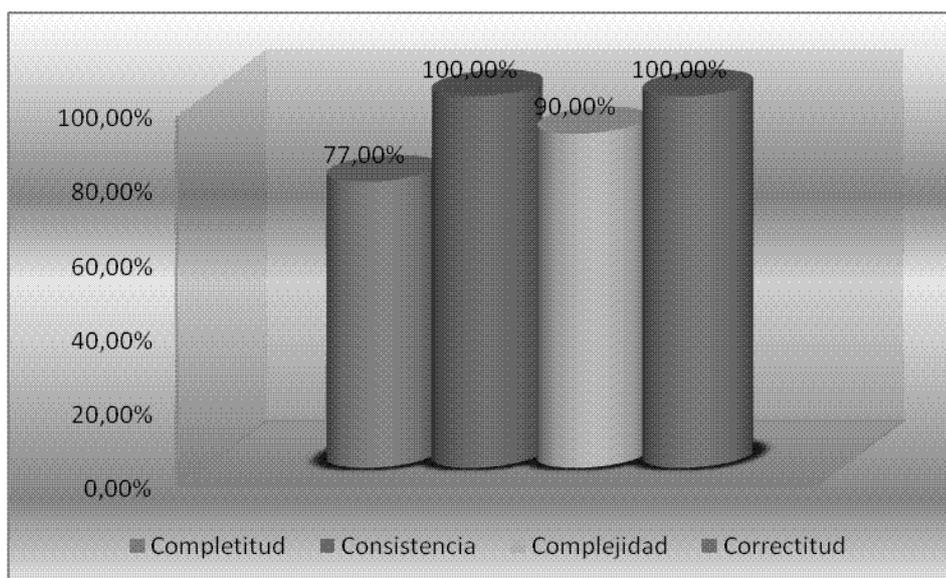


Figura 9: Gráfico de la calidad del Diagrama de Casos de Uso para la primera revisión

Revisión 2

Luego de haber realizado una primera revisión, los errores que se señalaron fueron corregidos y ya en la segunda los resultados se comportaron de acuerdo a la Figura 10, donde se puede observar que a todos los factores se les dio cumplimiento.

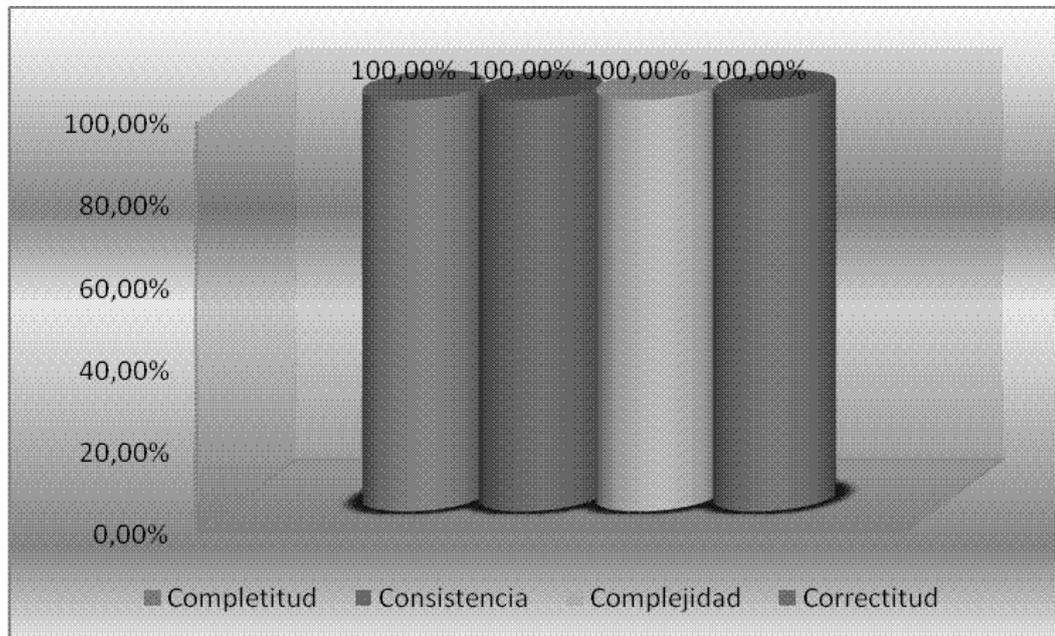


Figura 10: Gráfico de la calidad del Diagrama de Casos de Uso para la segunda revisión

3.6 Buenas prácticas para confección de los prototipos no funcionales (Interfaces)

En la medición de la calidad de la interfaz, no existen valores numéricos que den la certeza de que está bien confeccionada. Por lo que a continuación se mencionan algunas buenas técnicas, a tener cuenta a la hora de confeccionar una Interfaz.

La Interfaz debe contar con características de simplicidad, claridad y fácil comprensión. Es necesario que cuente con claridad visual, de forma que los elementos estén agrupados de forma comprensible y con significado en vez de al azar y de forma confusa.

Se tiene que tener conocimiento de qué información situar en la pantalla. Para ello, hay que poner sólo la información que es esencial y poner todos los datos relacionados con una tarea en una única pantalla. Además hay que tener en cuenta cómo situar la información en la pantalla.

En cuanto a las fuentes de letras, se recomienda utilizar minúsculas para el texto con la letra inicial de la frase en mayúsculas; para las etiquetas, encabezamientos o subtítulos utilizar mayúsculas. En cuanto a las palabras, se recomienda utilizar palabras cortas y familiares. También es necesario saber cómo alinear y/o resaltar el texto y las palabras, dónde situar las ilustraciones y los campos de datos.

Otro aspecto muy importante es el color, el cual añade una nueva dimensión a la facilidad de uso de la pantalla, ya que atrae la atención del usuario. Si se utiliza de forma adecuada, puede resaltar la organización lógica de una pantalla, facilitar la separación de componentes y acentuar las diferencias. Por el contrario, si se usa inadecuadamente, puede distraer y fatigar la visión debilitando la facilidad de uso del sistema. Por ejemplo, en las pantallas gráficas se recomienda no utilizar más de seis colores a la vez, evitar colores extremos (rojo y azul, amarillo y púrpura), evitar colores que no tengan contraste (blanco y amarillo, rojos, azules).

3.7 Análisis de los resultados

En la confección del presente trabajo de tesis se han tomado las precauciones necesarias para que el desarrollo se haga con la mayor calidad posible. Es por esto que en el análisis de la parte de levantamiento de requisitos se obtuvieron tan buenos resultados luego de haber realizado una revisión y corregido algunos errores, llegando a tener ningún indicio de ambigüedad en ellos. Otro aspecto que se midió fue el grado de validación, llegando a la conclusión que todos fueron correctamente validados. Estos resultados se deben a que en todo momento se estuvo interactuando con el cliente, para que en caso de que en cualquier momento revelara algún cambio, se realizara al momento y no al final del trabajo que implicaría pérdida de tiempo, y descontrol de cambios. En cuanto a la confección del diagrama de casos de uso se logró la Consistencia la Correctitud, la Completitud y la Complejidad luego de que se hiciera una primera revisión y se arreglaran algunos aspectos que todavía no alcanzaban la mayor calidad. En la confección de la interfaz de usuario se recomendó que se utilizaran colores que fueran refrescantes para la vista del usuario y combinaciones con un contraste que no

fuese escandaloso para el usuario. Además que la información estuviera bien organizada de acuerdo a las necesidades del usuario en cada momento.

3.8 Conclusiones

En el presente capítulo se comprobó la calidad del trabajo realizado en los anteriores capítulos. Se realizaron dos revisiones de acuerdo a los siguientes aspectos: calidad de la especificación de los requisitos, calidad del grado de validación de requisitos y la calidad de la confección del diagrama de casos de uso del sistema; los cuales después de pulir defectos detectados en el transcurso del trabajo, resultaron con buena calidad.

CONCLUSIONES

El análisis de las herramientas informáticas es de vital importancia, ayudando a entender de una manera estándar el software, en la comprensión del mismo por parte de los desarrolladores y logrando así un software que cumpla con las expectativas del cliente.

En el presente trabajo se definieron diferentes aspectos del Posicionamiento Web para que se tenga al menos una noción del tema, como también se analizaron diferentes herramientas SEO para tener en cuenta las funcionalidades y hacer comparaciones con las posibles funcionalidades de la herramienta a desarrollar.

Luego de reunirse con el cliente y realizar el levantamiento de requisitos se desarrollaron artefactos importantes para la ingeniería de requisitos, referentes al rol que se desarrolla en el presente trabajo, como la definición de requisitos funcionales y no funcionales, el diagrama de casos de uso y las descripciones de cada caso de uso.

Por último se le realizaron dos revisiones a lo antes desarrollado, obteniéndose errores en la primera revisión que fueron corregidos para la segunda, alcanzando como resultado final una buena calidad del trabajo realizado durante el desarrollo del cumplimiento del rol de analista de sistema en el presente trabajo de diploma.

RECOMENDACIONES

Se recomienda que se continúe con la realización del sistema basado en el análisis realizado, incluyendo la documentación referente a los flujos de trabajo fundamentales que faltan, como son diseño, implementación y pruebas, debido a la gran importancia que tiene para la comprensión del usuario y desarrolladores.

También se tiene que seguir investigando sobre el tema del Posicionamiento Web, lo cual puede sumar otras funcionalidades al sistema.

Impartir cursos de superación para el equipo de desarrollo dentro del proyecto, facilitando así una futura migración para una plataforma libre.

Se recomienda que en la línea de investigación de Posicionamiento Web, perteneciente al proyecto Operación Verdad, se continúe con la idea de realizar una segunda versión del software.

BIBLIOGRAFÍA REFERENCIADA

1. Andriano, I. N. V. (Agosto 2006). "Comparación del Proceso de Elicitación de Requerimientos en el desarrollo de Software a Medida y Empaquetado. Propuesta de métricas para la elicitación." La Plata, Argentina Córdoba, Argentina, UNLP, Facultad de Informática, Universidad Nacional de La Plata, UBP, Universidad Blas Pascal151.
2. Bonilla, D. A. (1992). "BASES DE DATOS PARA BIBLIOTECAS JURIDICAS:UNA APLICACION CON MICROISIS." 2005, from <http://www.metabase.net/metarecursos/documentos/TOMOIII.pdf>.
3. Codina, L. and M. C. Marcos (2005). "Posicionamiento web: conceptos y herramientas. (Spanish)." Web positioning: concepts and tools. (English) **14**(2): 84-99.
4. Cabanes, N. (2007). "Diccionario de informática." 2007, from <http://www.nachocabanes.com/diccio/ndic.php>.
5. Gosende, F. M.-J. (2007). ¿Cómo mejorar el posicionamiento de mi tienda virtual en los buscadores?
6. Google. (2008). "Definiciones de Base de datos en la web", from http://www.google.es/search?hl=es&rlz=1T4GGLJ_enCU271CU271&defl=es&q=define:Base+de+datos&sa=X&oi=glossary_definition&ct=title.
7. MLmundoenlinea.cl. "Glosario Tecnológico", 2007, from <http://www.mundoenlinea.cl/glosario.htm>.
8. Ibon Aizpurua, I. A., Nerea Ezeiza Galn: un buscador Internet/Intranet avanzado para textos en euskera.
9. Javier Gozende, F. M. (2006). Guía Practica de Posicionamiento en Buscadores.
10. Penela, C. G. (2004). "La selección de palabras clave para el posicionamiento en buscadores." Retrieved 24 de julio, 2007, from http://www.hipertext.net/web/pag225_print.htm.
11. Ugarte, D. d. (2002). El Libro del posicionamiento en buscadores.

BIBLIOGRAFIA CONSULTADA

1. Andriano, I. N. V. (Agosto 2006). "Comparación del Proceso de Elicitación de Requerimientos en el desarrollo de Software a Medida y Empaquetado. Propuesta de métricas para la elicitación." La Plata, Argentina
Córdoba, Argentina, UNLP, Facultad de Informática, Universidad Nacional de La Plata, UBP, Universidad Blas Pasca 151.
2. B. Bernárdez , A. D., M. Toro. (Agosto 2004,). "UNA PROPUESTA PARA LA VERIFICACIÓN DE REQUISITOS BASADA EN MÉTRICAS*." Retrieved 20/03, 2008, from <http://www.aemes.org/rpm/descargar.php?volumen=1&numero=2&articulo=2>.
3. Bonilla, D. A. (1992). "BASES DE DATOS PARA BIBLIOTECAS JURIDICAS:UNA APLICACION CON MICROISIS." 2005, from <http://www.metabase.net/metarecursos/documentos/TOMOIII.pdf>.
4. Cabanes, N. (2007). "Diccionario de informática." 2007, from <http://www.nachocabanes.com/diccio/ndic.php>.
5. Cano, O. S. (2007). Posicionamiento Web en Google.
6. Codina, L. and M. C. Marcos (2005). "Posicionamiento web: conceptos y herramientas. (Spanish)." Web positioning: concepts and tools. (English) **14**(2): 84-99.
7. Escribano, G. F. (9-12-2002). "Introducción a Extreme Programming
" Retrieved 02/05, 2008, from <http://www.info-ab.uclm.es/asignaturas/42551/trabajosAnteriores/Presentacion-XP.pdf>.
8. ESPAÑOLA, R, A.(2001) "DICCIONARIO DE LA LENGUA ESPAÑOLA - Vigésima segunda edición." Retrieved 20 de junio, 2007, from http://buscon.rae.es/draeI/SrvltConsulta?TIPO_BUS=3&LEMA=desarrollar.
9. GALVES, I. J. A. (10/10/06). "RESUMEN: 4 CAPITULOS DE INGENIERIA DE REQUERIMIENTOS." Retrieved 05/03, 2008, from <http://www.scribd.com/doc/270431/Ingenieria-requerimientos>
10. Google. (2008). "Definiciones de Base de datos en la web

", from

http://www.google.es/search?hl=es&rlz=1T4GGLJ_enCU271CU271&defl=es&q=define:Base+de+datos&sa=X&oi=glossary_definition&ct=title.

11. Gosende, F. M.-J. (2007). ¿Cómo mejorar el posicionamiento de mi tienda virtual en los buscadores?
12. Ibon Aizpurua, I. A., Nerea Ezeiza Galn: un buscador Internet/Intranet avanzado para textos en euskera.
13. Javier Gozende, F. M.(2006). Guía Practica de Posicionamiento en Buscadores.
14. JACOBSON, I., "El Proceso Unificado de Desarrollo de Software". 2000.
15. Kendell, K. E. K. y. J. (2002). Análisis y diseño de sistemas.
16. Luis Giraldo, Y. Z. (2005). "HERRAMIENTAS DE DESARROLLO DE INGENIERIA DE SW PARA LINUX." Retrieved 18/03, 2008, from hugolopez.phi.com.co/docs/download/file=Giraldo-Zapata-Herramientas%20de%20ISW.pdf,_id=17 .
17. Milenium.(2008)"El concepto de los Sitios Web." Retrieved 20 de junio, 2007, from <http://www.informaticamilenium.com.mx/paginas/espanol/preguntas/concepto.htm>.
18. Marcos, M.-C. (2006). "Evaluación del posicionamiento web en sistemas de información terminológicos online." Retrieved 22 de julio, 2007, from <http://www.hipertext.net/Web/pag267.htm>.
19. Millán, J. A. "Breve Historia de la Internet. El fruto caliente de la guerra fría." Retrieved 26/11, 2007, from <http://jamillan.com/histoint.htm>.
20. MLmundoenlinea.cl. "Glosario Tecnológico", 2007, from <http://www.mundoenlinea.cl/glosario.htm>.
21. Morris, M. V. (2007). "Algunas consideraciones sobre el posicionamiento Web." Retrieved 22 julio, 2007, from http://eprints.rclis.org/archive/00009821/01/Algunas_consideraciones_sobre_el_posicionamiento_Web.pdf.

22. Navarra, P. L. and J. A. M. Usero. (2004). "Agentes inteligentes en la búsqueda y recuperación de información." Retrieved 24 de julio, 2007, from <http://www.ucm.es/eprints/5840/01/2004-Lib-Agentes.pdf>.
23. Paradigm.V. "Visual Paradigm." Retrieved 18/03, 2008, from <http://www.visual-paradigm.com/product/vpuml/>.
24. Pelayo, V. M. (2005). "Interacción entre medidas de popularidad en el posicionamiento web. (Spanish)." Interaction of popularity measurements in web positioning. (English) **14**(2): 100-107.
25. Penela, C. G. (2004). "La selección de palabras clave para el posicionamiento en buscadores." Retrieved 24 de julio, 2007, from http://www.hipertext.net/web/pag225_print.htm.
26. Pressman, R., Ingeniería de Software. Un enfoque práctico. 2002: McGraw Hill.
27. Recio, M. (2006). "Mesa redonda "Una web para todos" Construir comunicación..., tender puentes
" Retrieved 24 de julio, 2007, from <http://www.proyectoweb.cubaweb.cu/info2006/index.html>.
28. Román, S. H. S.-F. B. (2005).NET Framework. **Volume**, DOI:
29. Ugarte, D. d. (2002). El Libro del posicionamiento en buscadores.
30. Vago E,R,D."Diseño de la interfaz de usuario. Pruebas del software." Retrieved 03/04, 2008, from http://html.rincondelvago.com/disenio-de-la-interfaz-de-usuario_pruebas-del-software.html.
31. Valcárcel, M. (Lima, junio 2006) GÉNESIS Y EVOLUCIÓN DEL CONCEPTO Y ENFOQUES SOBRE EL DESARROLLO. **Volume**, DOI:

GLOSARIO

Sitios:

- ❖ “Una Página de Internet o Página Web es un documento que contiene información específica de un tema en particular y que es almacenado en algún sistema de cómputo que se encuentre conectado a la red mundial de información denominada Internet, de tal forma que este documento pueda ser consultado por cualquier persona que se conecte a esta red mundial de comunicaciones. Un Sitio Web es un conjunto de páginas web relacionadas entre sí.”
- ❖ “Lugar en la red. Sitio o página Web.”
- ❖ “Punto de la red con una dirección única y al que pueden acceder los usuarios para obtener información.”

Internet:

- ❖ “Es un sistema mundial de redes de computadoras, un conjunto integrado por las diferentes redes de cada país que puede acceder información de otra computadora y poder tener inclusive comunicación directa con otros usuarios en otras computadoras.”
- ❖ “Es un amplísimo conjunto de ordenadores repartidos por todo el planeta, que se hallan interconectados entre sí -a modo de una inmensa Red-, a través de distintas líneas de intercambio de datos. Nació a finales de los años 70 en los Estados Unidos.”
- ❖ “Interconexión de redes informáticas que permite a las computadoras conectadas comunicarse directamente. El término suele referirse a una interconexión en particular, de carácter planetario y abierto al público, que conecta redes informáticas de organismos oficiales, educativos y empresariales. También existen sistemas de redes más pequeños llamados intranet, generalmente para el uso de una única organización.”
- ❖ “Red informática mundial, descentralizada, formada por la conexión directa entre computadoras u ordenadores mediante un protocolo especial de comunicación.”

Posicionamiento Web

- ❖ “Tratar de evaluar la adecuación de un recurso a una búsqueda, intentando mejorar la posición de los contenidos más indicados. “
- ❖ “Ser calculado de forma distinta por cada buscador.”
- ❖ “Literalmente, el Posicionamiento es el lugar que ocupa un producto o servicio en la mente del consumidor y es el resultado de una estrategia especialmente diseñada para proyectar la imagen específica de ese producto, servicio, idea, marca o hasta una persona (.....)”.
- ❖ El posicionamiento en buscadores consiste en aplicar diversas técnicas tendentes a lograr que los buscadores de Internet encuadren nuestra página Web en una posición y categoría deseada dentro de su página de resultados para determinados conceptos clave de búsqueda. Los motores de búsqueda generan dos tipos distintos de resultados: los resultados naturales u orgánicos (posicionamiento SEO) que son los que están basados en el algoritmo imparcial de los buscadores y los resultados patrocinados (posicionamiento SEM), cuya clasificación depende del dinero que se invierta en los anuncios.

Optimización de un Sitio Web

- ❖ “Realizar un análisis exhaustivo del código HTML, etiquetas, palabras claves, estadísticas, etcétera. Es decir, del diseño, estructura y contenidos que componen una página Web con el propósito de llevarla a los primeros puestos de los buscadores más populares.”

Buscador

- ❖ Un buscador es una página web en la que se ofrece consultar una base de datos en la cual se relacionan direcciones de páginas web con su contenido. Su uso facilita enormemente la obtención de un listado de páginas web que contienen información sobre el tema que nos interesa.
- ❖ Existen varios tipos de buscadores, en función del modo de construcción y acceso a la base de datos, pero todos ellos tienen en común que permiten una consulta en la que el buscador nos devuelve una lista de direcciones de páginas web relacionadas con el tema consultado.

SEO - Search Engine Optimizer.

- ❖ 'SEO' son las siglas de 'Search Engine Optimizer' ('Optimizador de Buscadores'). El 'SEO' no tiene por qué ser necesariamente el webmaster o el responsable del sitio Web, sino

que éste último puede contratar sus servicios. Existen cada vez más empresas que se han especializado en el posicionamiento en buscadores, y ofrecen ayuda a otros sitios Web.

Backlinks

- ❖ El número de Backlinks de una determinada página web es el número de páginas que la enlazan.

Googlebot

- ❖ Es el robot que utiliza Google para rastrear los sitios web en Internet.

Pagerank

- ❖ Es un valor numérico con el que se representa la importancia que una página web tiene respecto a las demás en la red.

Metodología

- ❖ La metodología en un proyecto de desarrollo de software, define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo. Una metodología es un proceso.

Requisitos

- ❖ Es la condición o capacidad que tiene que ser alcanzada por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.

Actor

- ❖ Entidad externa que inicia una actividad en el sistema (actor primario) e interactúa en cualquier actividad (actores secundarios).

Casos de uso

- ❖ Los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario.

Diagrama de casos de uso del sistema

- ❖ Un diagrama de casos de uso del sistema representa gráficamente a los procesos y su interacción con los actores.

Ergonomía

- ❖ Es el campo de conocimientos multidisciplinarios que estudia las características, necesidades, capacidades y habilidades de los seres humanos, analizando aquellos aspectos que afectan al entorno artificial construido por el hombre relacionado directamente con los actos y gestos involucrados en toda actividad de éste. En todas las aplicaciones su objetivo es común: se trata de adaptar los productos, las tareas, las herramientas, los espacios y el entorno en general a la capacidad y necesidades de las personas, de manera que mejore la eficiencia, seguridad y bienestar de los consumidores, usuarios o trabajadores

Metatags

- ❖ Son etiquetas de código HTML que contienen palabras clave por las que la página Web será encontrada. Los metatags son muy importantes a la hora de posicionar un sitio Web, ya que los buscadores ordenan las páginas de acuerdo a los metatags que van encontrando. Los más utilizados son el del título, descripción y keywords.

Robot

- ❖ Programa diseñado para recorrer la web siguiendo los enlaces entre páginas. Esta es la forma habitual empleada por los principales buscadores para encontrar las páginas que posteriormente forman parte de sus bases de datos.

Weblog

- ❖ Recursos informativos, en formato Web, en los que una persona o grupo de personas introducen por orden cronológico noticias, opiniones, sugerencias, artículos, reflexiones o cualquier otro tipo de contenido que consideran de interés, los cuales enlazan frecuentemente a otros recursos Web y cuya replica esta o no permitida según el propietario del Weblog.

Hipertexto

- ❖ Conjunto de documentos que contienen vínculos a otros documentos. Un ejemplo muy ilustrativo puede ser una página web. En ella hay enlaces a otros documentos que se muestran con tan sólo hacer un clic con el ratón.

Stakeholder

- ❖ Cualquier persona interesada en, afectada por y/o implicada con el funcionamiento del sistema software. Por ejemplo, el usuario, el cliente, etc.