

Universidad de las Ciencias Informáticas

Facultad 4



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

Título: Módulo para la extracción de características de las fuentes de información para el Sistema de Vigilancia Tecnológica.

Autor

Augusto Márquez Ramos

Tutores

Ing. Maikel Aparicio Reyor

MSc. Noralbis de Armas Rodríguez

La Habana, junio 2013

Datos del contacto

DECLARACIÓN DE AUTORÍA

Declaro ser el único autor de este trabajo y reconozco al Centro de Tecnologías para la Formación (FORTES) de la Universidad de las Ciencias Informáticas hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año 2013.

Augusto Márquez Ramos

Firma del Autor

Ing. Maikel Aparicio Reytor

Firma del Tutor

MSc. Noralbis de Armas Rodríguez

Firma de la Tutora



“Nosotros no podemos ser fuertes en todo, de ahí la importancia de determinar en qué campos podemos ser fuertes, más fuertes y más especializados, porque nosotros tenemos una ventaja: El recurso humano.”

Fidel Castro Ruz

A handwritten signature of Fidel Castro Ruz, written in black ink. The signature is stylized and cursive, enclosed within a horizontal line that extends slightly beyond the left and right sides of the text.

Agradecimientos

AGRADECIMIENTO

A mi mamá Ailén por ser la mejor madre del mundo y darme siempre su apoyo incondicional.

A mis hermanos Ramoncito, Raudel y a mi papá Ramón por todo el apoyo brindado.

A mi hermano Osmar porque sin tu ayuda me hubiera sido muy difícil llegar hasta aquí, por aguantar todas mis malcriadeces y estar siempre a mi lado.

A mi tía Niuris por ser después de mi mamá, mi otra guía.

A mi padrastro Lázaro por toda su inspiración y por toda la confianza depositada.

A mis abuelos por guiarme siempre.

A mi familia en general, porque todos han puesto su granito de arena en esta obra.

A mis amigos de todos estos años, a los de verdad, a los que siempre han estado presentes.

A Marlon y Yosvel por aguantar todas mis malcriadeces y estar siempre a mi lado.

A mis demás amigas (os) Liuis, Daimaris, Maritza, Yadira Albuerne, Yadira Prada, Dianne, Yaumara, Claudia y Reinier.

A los maravillosos profesores de nuestra Facultad Regional de Artemisa en especial a las profesoras Yuray, Yaskara, Lourdes y Geidis.

A mis tutores, Maikel por toda su ayuda y orientación durante la realización de esta investigación y en especial a la profe Noralbis por toda su preocupación constantemente y las horas de madrugadas dedicadas para que todo saliera bien.

A Osmar, Carlos, Prada, Yanet Roque, Alejandro, Daimaris, Maritza, Reinier, Yadira Albuerne, Marlon, Yosbel, Liuis, Anita, a los profesores Yuniór, Yenieris, Javier, por el tiempo dedicado a la revisión, los consejos y aclaración de dudas, a todos muchas gracias.

Al Comandante en Jefe Fidel Castro Ruz por brindarme la posibilidad de estudiar esta carrera.

Por último, a mis compañeros de aula y a todos aquellos que de una forma u otra me han apoyado.

Dedicatoria

DEDICATORIA

A mi mamá Ailén por ser lo más grande y hermoso que tengo, por ser mi razón de ser, mi inspiración, por estar en cada momento cuando más la necesité, por darme siempre todo su apoyo incondicional, por ser ejemplo y guía.

A mis hermanos Ramoncito y Raudel que siempre me ayudaron durante toda mi carrera, a mi otro hermano Osmar por todo el apoyo que me dio en todo momento, sin su ayuda me hubiera sido muy difícil llegar hasta aquí, esta victoria es tanto mía como tuya. También a Karla y a Mercedita.

A mi padrastro Lázaro por haber estado siempre ahí como un padre, solo no es padre aquel que crea, sino quien cría, da amor y cariño como lo has hecho tú con nosotros.

A mi papá Ramón y su esposa Ana, que siempre me han dado consejos para salir adelante.

A mis abuelos, Cuya, Giro, y en especial a mimá (Nelia) y pipa (Cristóbal) por estar siempre ahí cuidando de cada paso que doy.

A todos mis primos, mis tíos, en especial a mi tía Niuris por ser mi otra mamá.

A mis amigos (as) que me han dado un pedacito de su corazón y con él toda la confianza del mundo para llegar hasta aquí.

A mis ídolos, el Comandante en Jefe Fidel Castro Ruz y el Che.

A mi amiga Lient que no está presente hoy pero que estoy seguro le hubiese gustado mucho verme graduado.

A todos ustedes van dedicados todos mis logros, no solo estos, todos los que a partir de este momento sea capaz de alcanzar, porque gracias a ustedes hoy he llegado hasta donde estoy.

RESUMEN

El entorno actual del desarrollo empresarial ha planteado la necesidad de incorporar productos de inteligencia empresarial. Con elementos que permitan minimizar el riesgo a la hora de tomar decisiones, conocer el entorno y así poder anticiparse a los cambios tecnológicos. En este contexto, el grupo de Gestión de la Información y el Conocimiento (GIC) surge en el centro FORTES. Siendo este uno de los centros productivos que se desarrollan en la facultad 4 de la Universidad de las Ciencias Informáticas (UCI), que apoya a la vigilancia tecnológica.

Para realizar la vigilancia tecnológica una de las acciones que realiza el grupo GIC, es la extracción de características, siendo este uno de los principales procesos que desarrolla la Minería de texto. Este proceso de extracción de características de las fuentes de información en el grupo GIC se realiza de forma manual. Lo que trae consigo que la vigilancia tecnológica se convierta en un proceso lento y engorroso, haciendo que en ocasiones sus productos no se obtengan en el momento oportuno.

Sobre esta base, el objetivo del trabajo está orientado a implementar un módulo para el proceso de extracción de características de las fuentes de información para el Sistema de Vigilancia Tecnológica que le facilite al grupo GIC captar información. Por lo que se obtuvo como resultado final, un módulo que extrae las características de las fuentes de información de forma automática, permitiendo así, que exista un lugar dónde los usuarios pueden consultar las características, para un posterior análisis.

Palabras clave: minería de texto, vigilancia tecnológica, fuentes de información.

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	7
Introducción	7
1.1 Minería de texto en la extracción de información.....	7
1.1.1 Minería de texto.....	7
1.1.2 Principales funciones de la Minería de texto.....	9
1.1.3 Técnicas de la Minería de texto	11
1.1.4 Procesos de la Minería de texto	14
1.2 Análisis de soluciones similares para la extracción de las características	15
1.2.1 Herramientas internacionales de apoyo a la vigilancia tecnológica.....	15
1.2.2 Herramientas nacionales de apoyo a la vigilancia tecnológica	17
1.3 Extracción de las características de las fuentes de información para el Sistema de Vigilancia Tecnológica.....	20
1.4 Metodologías, lenguajes y herramientas a utilizar	22
1.4.1 Metodología de desarrollo de software	22
1.4.2 Lenguajes.....	24
1.4.4 Gestor de base de datos	27
1.5 Framework para el desarrollo del software.....	27
Conclusiones parciales	30
CAPÍTULO 2: EXPLORACIÓN Y PLANIFICACIÓN	32
Introducción	32
2.1 Descripción y características de la propuesta de solución.....	32
2.1.1 Propuesta del sistema	33
2.1.2 Funcionalidades del sistema	33
2.2 Fase de exploración.....	34
2.2.1 Personal relacionado con el sistema	34
2.2.2 Historias de usuario.....	35
2.3 Fase de planificación.....	38
2.3.1 Plan de entrega	38
2.3.2 Estimación de esfuerzos	39
Conclusiones parciales	39
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA.....	40

3.1 Diseño del sistema.....	40
3.1.1 Tarjetas CRC.....	40
3.1.2 Diagrama de despliegue.....	44
3.2 Fase de iteraciones.....	45
3.2.1 Primera iteración.....	45
3.2.2 Segunda iteración.....	46
3.2.3 Tercera iteración.....	46
3.2.4 Cuarta iteración.....	47
3.3 Fase de pruebas.....	47
3.3.1 Pruebas de aceptación.....	48
Conclusiones parciales.....	55
CONCLUSIONES.....	56
RECOMENDACIONES.....	57
REFERENCIAS BIBLIOGRÁFICAS.....	58
BIBLIOGRAFÍAS.....	62
ANEXOS.....	65
Anexo 1: Guía para la entrevista.....	66
Anexo 2: Prototipos de interfaz de las funcionalidades.....	67
Anexo 3: Tareas de desarrollo por iteración.....	70

Introducción

INTRODUCCIÓN

Con el desarrollo y expansión que han alcanzado las Tecnologías de la Información y las Comunicaciones (TIC), así como el crecimiento y evolución de Internet, han provocado que hoy en el mundo se genere diariamente un enorme volumen de información de todo tipo. Esto trae consigo que las instituciones dedicadas a la investigación, adopten una nueva estructuración, con el fin de analizar las técnicas y tecnologías actuales para desarrollar sistemas con calidad.

Dicha estructuración, ha configurado un entorno con un alto grado de incertidumbre, dinámico, de profundos cambios, en el que la toma de decisiones y la gestión del conocimiento, son elementos claves para consolidar organizaciones sostenibles, productivas e innovadoras. De esta forma, *“se hacen necesarias prácticas sistemáticas que garanticen la estabilidad y el crecimiento económico de las organizaciones en dicho entorno”* (Vargas, y otros, 2006), propiciando así el cambio tecnológico, la innovación y la competitividad de las organizaciones.

Una práctica que es muy utilizada por las organizaciones para monitorear los cambios descritos, *“es la vigilancia tecnológica, la cual permite a una organización estar atenta al cambio de manera sistemática por medio del estudio permanente del mercado, del ámbito científico tecnológico, del ámbito político y del ámbito social”* (Porter, y otros, 2005). Uno de los factores definitivos para el éxito de la innovación es una adecuada vigilancia tecnológica. La vigilancia se organiza como un sistema estructurado que permite coordinar las actividades de planeación, captación, procesamiento y socialización de la información.

En ese sentido, la implementación de un Sistema de Vigilancia Tecnológica tiene un papel importante en las entidades que realizan investigación. Estas brindan la posibilidad de obtener ventajas e incrementar la capacidad de organización de las entidades; de ellas dependerá con qué rapidez, organización y facilidad esté lista esa información y ese conocimiento para su uso.

Para las organizaciones que realizan vigilancia tecnológica, la información es apreciada como un recurso estratégico de gran utilidad para el buen desempeño de las mismas. Es un componente del que se puede extraer conocimiento y satisfacer las necesidades de personas e instituciones, razón por la que adquiere una importancia significativa para el desarrollo de cualquier sector del mundo.

Introducción

Gran parte de esta existe en forma de texto. La calidad de todo este conocimiento depende de la habilidad de hacer ciertas operaciones, por ejemplo la extracción de la información.

Para automatizar el procesamiento de este tipo de datos, la computación ha posibilitado el desarrollo de mecanismos como; minería de datos y la minería de texto, pues estas pertenecen al área del conocimiento de la inteligencia artificial. Este proceso, comprende varias etapas, que van desde *“la obtención de los datos hasta la aplicación del conocimiento adquirido en la toma de decisiones. Entre esas etapas, se encuentran la que puede considerarse como el núcleo del proceso de descubrimiento de conocimiento en bases de datos (del inglés Knowledge Discovery in Databases, KDD) y que se denomina minería de datos o data mining”*. (Navarro, 2009)

“La minería de datos ha sido usada como sinónimo de descubrimiento de conocimiento en bases de datos, sin embargo, corresponde a una de las fases de todo el proceso de descubrimiento, encargada de hacer uso de técnicas de aprendizaje automático para desarrollar técnicas capaces de aprender y extraer conocimiento de los datos” (Valencia, 2005). Por otra parte, la Minería de Texto (del inglés Text Mining) es una extensión de la minería de datos que su principal objetivo es el descubrimiento de patrones interesantes y nuevos conocimientos en colecciones de textos no estructurados.

En la actualidad países como Francia, España, Japón, E.E.U.U. y Colombia utilizan en sus instituciones herramientas que facilitan la vigilancia tecnológica. Estas aplican Minería de texto a la información, con el objetivo de captar información de los avances de la tecnología, analizando así las oportunidades y amenazas que estas generan.

En Cuba, a pesar del bloqueo económico impuesto por los Estados Unidos, la necesidad de establecer sistemas de vigilancia tecnológica está planteado y reconocido en el entorno empresarial cubano; *“pero aún no se concretan acciones prácticas que faciliten la implementación de programas coherentes, en este sentido”* (Betancourt, y otros, 2010). Aunque existen instituciones que no han sido conscientes en ocasiones de las ventajas de la vigilancia tecnológica. *“La vigilancia está presente en un número cada vez mayor de estas. Su aplicación puede beneficiar al conjunto del ciclo innovador”*. (Fernández, y otros, 2012)

Una de estas instituciones que pone en práctica la vigilancia tecnológica es la Universidad de las

Introducción

Ciencias Informáticas (UCI), que dentro de su estructura cuenta con el Centro de Tecnologías para la Formación (FORTES). Como parte de ese Centro pertenece el Grupo de Gestión de la Información y el Conocimiento (GIC), que brinda servicios y productos de información, como resultados de la vigilancia tecnológica.

La vigilancia tecnológica constituye una gran herramienta de apoyo y desempeño para el grupo GIC. La puesta en práctica de esta, le proporciona a la institución captar, procesar y visualizar la información precisa para apoyar a la toma de decisiones de manera oportuna. Además de poder anticiparse a los cambios con el menor riesgo posible. Es la razón por la que en el curso 2011-2012, se realiza un trabajo de diploma en el que se plantea el análisis y diseño de un Sistema de Vigilancia Tecnológica, que le permitiera al grupo GIC agilizar la captación, procesamiento y socialización de la información con un enfoque estratégico.

Para que el Sistema de Vigilancia Tecnológica funcione, primeramente se necesita captar toda la información necesaria, para que posteriormente sea procesada y socializada. Este proceso de captación el grupo GIC lo realiza de forma manual, copiando de un gran cúmulo de fuentes de información las características a procesar y guardándolas una a una en un documento de Microsoft Work, trayendo consecuencias como:

- Se obvie información.
- El proceso de extracción de características sea lento y engorroso.
- Se ve afectada la toma de decisiones.
- Se dificulta brindar a los miembros del grupo la información existente sobre las temáticas a fines al centro.
- No se puede formular a tiempo las alertas de amenazas sobre las pérdidas de repercusión en nuestro mercado.

De la situación problemática expuesta anteriormente surge el siguiente **problema científico**:

¿Cómo contribuir al proceso de captación de información que realizan los especialistas del grupo GIC del centro FORTES?

Introducción

El **objeto de estudio** de este trabajo comprende a la Minería de texto.

Para darle solución a la situación problemática planteada se define como **objetivo general** implementar un módulo para el proceso de extracción de características de las fuentes de información para el Sistema de Vigilancia Tecnológica que le facilite al grupo GIC captar información.

El **campo de acción** está enmarcado en la extracción de características de las fuentes de información para el Sistema de Vigilancia Tecnológica del grupo GIC del centro FORTES.

Se plantea como **idea a defender** con el desarrollo de un módulo para la extracción de características de las fuentes de información para el Sistema de Vigilancia Tecnológica se contribuirá al proceso de captación de información que realizan los especialistas del grupo GIC del centro FORTES.

Para dar cumplimiento al objetivo general se definen los siguientes **objetivos específicos**:

- Analizar los elementos teóricos – conceptuales referentes a la Minería de texto y sus técnicas.
- Elaborar los artefactos de las fases de exploración y planificación necesarios para guiar el proceso de desarrollo.
- Implementar las funcionalidades requeridas para el módulo de extracción de características de las fuentes de información para el Sistema de Vigilancia Tecnológica del grupo GIC perteneciente al centro FORTES.
- Probar las funcionalidades del módulo para la extracción de características de las fuentes de información para el Sistema de Vigilancia tecnológica del grupo GIC perteneciente al centro FORTES.

Las **tareas de investigación** que se deben cumplir son:

- Sistematización de los referentes teóricos que sustentan la Minería de texto y sus técnicas para la extracción de las características.
- Caracterización del estado actual de las herramientas similares para la extracción de características.

Introducción

- Ejecución de las fases de exploración y planificación del módulo de extracción de características de las fuentes de información.
- Elaboración del módulo para el proceso de extracción de características de las fuentes de información para el Sistema de Vigilancia Tecnológica.
- Validación de las funcionalidades del módulo a través de los casos de prueba.

Para dar cumplimiento a las tareas investigativas se emplearon los siguientes **métodos científicos de investigación**.

Métodos teóricos:

- **Histórico-Lógico:** se emplea para estudiar la evolución y desarrollo de los indicadores de la Minería de texto en la extracción de características y comprender lógicamente cuáles son las tendencias actuales.
- **Analítico-Sintético:** se utilizó con el objetivo de analizar la teoría recopilada sobre la Minería de textos y sus técnicas, para el proceso de extracción de características de las fuentes de información, posibilitando identificar aquellas técnicas que puedan ser aplicadas en el desarrollo del módulo.

Método empírico

- **Entrevista:** se utilizó con el objetivo de determinar los tipos de fuentes de información y las características que se le extraerán a las mismas. (Ver Anexo 1)

El presente trabajo de diploma se divide en tres capítulos:

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA. Este capítulo abarca los conceptos generales sobre la Minería de texto, las técnicas que se utilizan para la extracción de características de las fuentes de información. Se analizan las entidades y herramientas que realizan la extracción de características de las fuentes de información para apoyar a la vigilancia tecnológica. Se define la metodología, herramientas y tecnologías a utilizar en el desarrollo de la aplicación.

CAPÍTULO 2: EXPLORACIÓN Y PLANIFICACIÓN. Se describe la propuesta del sistema y las

Introducción

funcionalidades del sistema. Se identifica el personal relacionado con la aplicación, se describen las historias de usuario y se realiza un plan de entrega.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA. En este capítulo se describe la etapa de implementación que conlleva a la obtención del módulo. Se definen para cada iteración las tareas por historias de usuario correspondientes. Se establece el diseño mediante el uso de tarjetas CRC para tener un mejor entendimiento de lo que se quiere concebir. Se realiza una descripción de la solución y los resultados obtenidos por las pruebas realizadas a las funcionalidades implementadas.

Finalmente, se presentan las conclusiones, las referencias bibliográficas utilizadas durante el desarrollo del trabajo, la bibliografía y por último los anexos que complementan el cuerpo del trabajo.

Capítulo I. Fundamentación Teórica

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

La Minería de texto es un área nueva de la investigación en la computación, que trata de resolver la crisis de la sobrecarga de información a través de la combinación de técnicas. Mediante estas técnicas se puede realizar la extracción de información, acción que realiza el grupo GIC para apoyar al proceso de captación de las fuentes de información.

A continuación se introducen los conceptos generales sobre la Minería de texto, los procesos que la definen y las técnicas que se utilizan para la extracción de características de las fuentes de información. Se realiza un estudio de las tendencias actuales sobre las herramientas de apoyo a la extracción de características. Se define la metodología, tecnologías y herramientas a utilizar en el desarrollo de la aplicación.

1.1 Minería de texto en la extracción de información

1.1.1 Minería de texto

La Minería de texto es una de las extensiones de la minería de datos que aplica sus técnicas a documentos y servicios de información. La minería de datos *“es un conjunto de técnicas y tecnologías que permiten explorar grandes bases de datos, de manera automática o semiautomática, con el objetivo de encontrar patrones repetitivos, tendencias o reglas que expliquen el comportamiento de los datos en un determinado contexto”*. (Sinnexus, 2007)

La Minería de texto o minería textual es una *“aplicación de la lingüística computacional y del procesamiento de textos que pretende facilitar la identificación y extracción de nuevo conocimiento a partir de colecciones de documentos o corpus textual”*. (Eíto, y otros, 2004)

Hearst plantea que la Minería de texto *“se enfoca en el descubrimiento de patrones interesantes y nuevos conocimientos en un conjunto de textos, es decir su objetivo es descubrir tendencias, desviaciones y asociaciones en la gran cantidad de información textual disponible”*. (Hearst, 1999)

Blanco define que la Minería de texto *“es una tecnología emergente cuyo objeto es la búsqueda de conocimiento en grandes colecciones de documentos no estructurados”*. (Blanco, 2010)

Capítulo I. Fundamentación Teórica

Otra definición recogida de un documento comercial, en este caso del fabricante de Software de Análisis Estadístico (SAS), señala que la minería textual *“es el proceso de investigar una gran colección de documentos en texto libre, para descubrir y usar el conocimiento disponible en la totalidad de la colección”*. (SAS, 2010)

Algunos autores como Manuel Montes¹ definen el proceso de Minería de texto como *“una serie de pasos o procesos a saber una fase de pre-procesamiento, donde los textos son transformados a algún tipo de representación semiestructurada que permita su análisis automático”*. (Montes, 2010).

Por otro lado Martí A. Hearst², sugiere que estos procesos se componen de tres etapas: (Hearst, 1999)

1. Preparación del texto: selección, limpiado y pre-procesamiento del texto. En esta etapa tienen lugar procesos como la identificación y el etiquetado de las partes de la oración.
2. Procesamiento del texto: uso de algoritmos para procesar los datos, comprimiendo y transformándolos para identificar trozos importantes de información. En esta etapa un sistema bien entrenado de programación en lenguaje natural determinaría identidades canónicas y variables, identificando relaciones conceptuales entre las entidades por medio de árboles, ontologías, redes neuronales, reglas asociativas y algoritmos genéticos.
3. Análisis del texto: evaluación del rendimiento para valorar si la información fue proporcionada de forma correcta y relevante.

La minería de texto a diferencia de la minería de datos, se propone extraer información útil a través de colecciones de fuentes de información no estructuradas. *“Las fuentes de información no estructuradas son aquellas cuya construcción el investigador no ha tenido injerencia alguna y debe, por ende, realizar una faena de estructuración que le permita incorporarlas, como material, en su trabajo. Por el contrario, las fuentes estructuradas o semiestructuradas son aquellas en las que el investigador intervino para adaptar las fuentes a sus necesidades de manera tal que la manifestación surgida de ellas se adapte bien a las exigencias de su tarea”*. (Saltalamacchia, 2006)

¹ Manuel Monte, Laboratorio de Lenguaje Natural, Centro de Investigación en Computación, Instituto Politécnico Nacional. Av. Juan de Dios Batís, Zacatenco, 07738 México, D.F. mmontesg@susu.inaoep.mx

² Dr. Hearst se centra en el diseño, búsqueda y recuperación de información, construcción y evaluación de los sistemas de acceso a la información y lingüística computacional. Ha diseñado nuevas visualización de la información y las técnicas de análisis de texto para este propósito. hearst@ischool.berkeley.edu

Capítulo I. Fundamentación Teórica

El autor se acoge a lo planteado por Marti Hearts y la definición de María del Pilar Blanco, resumiendo que la Minería de texto es una tecnología emergente cuyo objetivo es el descubrimiento de patrones interesantes y nuevos conocimientos en grandes colecciones de documentos no estructurados. Esta mezcla de las dos definiciones planteadas ayuda a facilitar la comprensión y la interpretación de lo que es la minería de texto para poder aplicarla en el desarrollo del módulo.

Sin duda, la Minería de texto deriva gran parte de su inspiración sobre la minería de datos. Por lo tanto, no es sorprendente encontrar que la minería de texto y la minería de datos tienen un alto nivel de similitudes arquitectónicas. Por ejemplo, ambos tipos de sistemas se basan en rutinas de pre-procesamiento, en el descubrimiento de patrones. Además, la Minería de texto adopta muchos de los tipos de patrones específicos de sus principales operaciones en el análisis y la extracción de información.

Por otra parte, debido a la centralidad de textos en lenguaje natural, la Minería de texto también se basa en los avances de otras disciplinas de las ciencias de la computación. En cuestión con el manejo del lenguaje natural como la minería de datos, recuperación de la información y la lingüística computacional. De las que se apoya para desarrollar sus principales funciones.

1.1.2 Principales funciones de la Minería de texto

La Minería de texto facilita el proceso de extracción de características de las fuentes de información disponible en grandes colecciones de documentos no estructurados. Las funciones que principalmente satisfacen de la Minería de texto al grupo GIC o la salida que se puede esperar de ellas son:

Identificar hechos y datos puntuales de los documentos

La identificación de hechos presentes en los documentos es una de las aplicaciones que ofrece la Minería de texto, también se conoce como extracción de características. Es una funcionalidad que decidirá qué grupos va a generar a partir de la similitud que calcule entre los documentos de la colección.

Esta función trata de extraer del texto de los documentos el título, los nombres de las personas, las palabras clave, organizaciones, fechas y relaciones que existen entre ellas. Por lo que la hace una

Capítulo I. Fundamentación Teórica

función muy importante en el desarrollo del módulo para el proceso de extracción de características de las fuentes de información ya que su objetivo es que el grupo GIC pueda extraer el título, los autores, las palabras clave, institución, fecha y resumen de las fuentes de información a captar de forma automática.

Agrupación de documentos similares o clustering

La agrupación de documentos consiste en la concentración de documentos entre los que existe cierta similitud. La similitud se establecerá a partir de la terminología utilizada por los autores en la redacción de los textos.

La agrupación automática tiene distintas aplicaciones, entre ellas:

- Facilitar la comprensión de una colección de documentos al agrupar aquellos que son similares en una misma clase será posible obtener una visión general, de los temas tratados en todos los documentos de la clase con solo leer una parte de los textos incluidos en ella.
- Identificar relaciones entre documentos en una colección que previamente se desconocían.
- Mejorar la organización de los resultados devueltos por un motor de indexación. El clustering supondrá una mejora sustancial frente a los mecanismos de visualización de resultados actualmente generalizados, en los que cada documento recuperado se muestra de forma independiente.
- Identificar duplicados potenciales y documentos que por tener una información similar pueden no ser relevantes.

Identificar los conceptos tratados por los documentos

Esta función permitirá extraer los principales temas, ideas o conjunto de términos que son tratados en los documentos. En la Minería de texto, un concepto puede representar una idea tratada en el documento. La capacidad de identificar un concepto se basa en la ocurrencia de determinados términos y combinaciones de estos en el texto del documento.

Con la aplicación de esta función, una vez identificados los conceptos tratados por una fuente de información, será posible identificar por los especialistas del grupo GIC las fuentes de información que traten de ese mismo concepto y crear redes conceptuales a través del contenido de la colección.

Capítulo I. Fundamentación Teórica

Una de las principales ventajas de la Minería de texto es la posibilidad de crear estas redes de conceptos que pueden ser desarrolladas después por el Sistema de Vigilancia Tecnológica.

Visualización y navegación de colecciones de texto

Todas estas funciones que se han analizado ofrecen como resultado de sus análisis, listados de términos o redes de conceptos, grupos de documentos relacionados, resúmenes, palabras clave, hechos y organizaciones, que proceden de una colección de fuentes de información analizada. Esta función se aplica en la interfaz de usuario como un componente clave en el Sistema de Vigilancia Tecnológica del grupo GIC que permite la comprensión y visualización de la información. La interfaz mostrará los datos en un formato que haga posible su interpretación y permita a los especialistas de grupo GIC moverse con facilidad entre los distintos textos analizados. Para aplicar estas funciones al proceso de extracción de características de las fuentes de información, se requiere de una serie de técnicas de la Minería de texto.

1.1.3 Técnicas de la Minería de texto

Las técnicas de Minería de texto han tenido un auge a partir de principios de este siglo. Estas técnicas ahora ofrecen formas de descubrir información a partir de fuentes de información. A continuación se hará alusión a cada una de ellas.

Pre-procesamiento de los documentos

Esta técnica incluye la extracción de las palabras separadas entre sí, por espacios en blanco o signos de puntuación y la eliminación de los signos de puntuación. En el pre-procesamiento de los documentos una de las principales tareas es la eliminación de palabras vacías, carentes de significado, como son preposiciones, artículos y conjunciones. El programa informático debe convertir el documento que se va a procesar a un formato texto plano, no binario para así finalizar.

El pre-procesamiento utiliza la normalización de las palabras extraídas de los documentos. Esta normalización, también llamada lematización, consiste en dividir cada palabra en los lemas que la forman. Por ejemplo, las palabras: alumno, alumna, alumnado, alumnos, comparten una misma raíz léxica (alumn-) que les da el mismo significado semántico.

Capítulo I. Fundamentación Teórica

La normalización representaría y reduciría todas las palabras que comparten la misma raíz mediante ésta. Este proceso tiene una gran importancia para el desarrollo del módulo porque escoge así aquellos términos que son los mejores candidatos para representar el contenido del texto. Normalmente la lematización reducirá las variaciones de género y número en los sustantivos y adjetivos, así como las flexiones de los tiempos verbales.

Siguiendo la terminología utilizada por Etxeberría³ un aspecto importante en el pre-procesamiento de los documentos es la identificación de los llamados “*segmentos repetidos o frases*” (Murgiondo, 1995). Es decir, secuencias de palabras que aparecen contiguas en el texto y que usadas de esta forma tienen un significado especial. En la Minería de texto, la extracción de estos términos compuestos sí es importante, ya que busca el conjunto de conceptos que representan el contenido de un documento. Las técnicas clásicas en la Minería de texto se estructuran en tres etapas:

- **Etapa de pre-procesamiento:** es el proceso mediante el que los textos se transforman en algún tipo de representación estructurada que facilite su análisis.
- **Etapa de representación:** la representación depende de la técnica de pre-procesamiento utilizada.
- **Etapa de descubrimiento:** son algoritmos que a partir de una representación estructurada de la información, son capaces de descubrir regularidades en los textos.

Como se puede observar, todas las etapas están muy interrelacionadas, así pues, la primera etapa condiciona el descubrimiento de las características que la Minería de texto puede realizar. Por lo que esta técnica de pre-procesamiento es aplicada al módulo para el proceso de extracción de características de las fuentes de información.

Análisis de clusters

Jain⁴ definen el análisis de clusters “*como la organización de una colección de patrones, en clusters o*

³ Juan Etxeberría, profesor titular de Estadística de la Universidad del País Vasco, es doctor en Ciencias de la Educación y licenciado en Ciencias Exactas (Estadística). Especialista en Estadística Aplicada y Análisis Multivariantes con especial interés en el modelo lineal general y el análisis de datos textuales.

⁴ Anil K. Jain es Profesor Distinguido en el Departamento de la Universidad de Ciencias e Ingeniería de la Computación en la Universidad Estatal de Michigan. Recibió el título de doctorado en la Universidad Estatal de Ohio en 1970 y 1973, respectivamente. Sus intereses de investigación incluyen el reconocimiento de patrones, visión por computador y reconocimiento biométrico. Sus artículos sobre la biometría se han publicado en la revista Scientific American, Naturaleza,

Capítulo I. Fundamentación Teórica

grupos en base a su similitud” (Jain, y otros, 1999). Arenas plantea que es “una técnica que permite identificar grupos o clases de objetos similares a partir de un espacio multidimensional”. (Arenas, y otros, 1992)

El objetivo principal de la clasificación de documentos, como concepto global, es reducir la diversidad de datos y la sobrecarga de información mediante la agrupación de documentos similares. Con respecto a la gestión del conocimiento, la clasificación de documentos puede ser vista como una herramienta que permite simplificar el acceso y procesamiento del conocimiento explícito, facilitando la recuperación, organización, visualización, desarrollo e intercambio de conocimientos.

El análisis de clusters consiste en la generación automática de grupos de documentos relacionados, por ejemplo, documentos que traten un mismo tema o asunto. A diferencia de lo que ocurre en la categorización, en los procesos de clustering no existe un conjunto de categorías preestablecido, sino que el propio algoritmo a utilizar debe generar automáticamente esas categorías, contribuyendo de esta forma a generar un nuevo conocimiento. Por lo que esta técnica no es tomada para el desarrollo del módulo.

Identificación de nombres propios

Una de las principales funciones que satisface la Minería de texto es la extracción de nombres propios relativos a personas, organizaciones, eventos, funciones y fechas. La identificación de las relaciones que existen entre estos nombres propios pueden constatar los hechos descritos en los documentos. Esta técnica tiene la capacidad de identificar aquellos fragmentos que pueden corresponder a un nombre propio, identificar su tipo si se refiere a una persona, organización, lugar y las formas alternativas que se utilizan en el texto para hacer referencia.

La identificación de nombres propios es un tema complejo en la extracción de la información, por eso no se utiliza esta técnica en el desarrollo del módulo. En este sentido, es necesario recurrir a técnicas de análisis sintáctico de las sentencias, para identificar los verbos que sirven de nexo entre los nombres propios y tratar de deducir así posibles relaciones.

Capítulo I. Fundamentación Teórica

Representación de documentos mediante el modelo vectorial

Los documentos en este modelo se representan mediante una secuencia de términos o componentes que corresponden con los distintos términos utilizados para describir el contenido del documento. Una de las ventajas de este modelo es el hecho de calcular la similitud entre la ecuación de búsqueda y los documentos. Esto permite ordenar los documentos recuperados, mostrando al principio de la lista aquellos documentos que son más similares a la ecuación de búsqueda y al final los que son menos.

La necesidad de representar el contenido de los documentos mediante un modelo no es la premisa de la aplicación para el grupo de GIC, por lo que no es de gran importancia aplicar esta técnica de Minería de texto. El modelo generalizado para las aplicaciones de la minería textual es el vectorial. Por lo que el autor decide utilizar en el desarrollo del módulo, la técnica de pre-procesamiento de documentos por los motivos expuestos anteriores. Para la aplicación de esta técnica, primeramente se debe pasar por procesos de la Minería de texto.

1.1.4 Procesos de la Minería de texto

Mediante el proceso de análisis de separación, evaluación, validación y comparación, se le agrega valor a la información para convertirla en conocimientos. En el caso de la información textual, solo las computadoras pueden manipular rápidamente la inmensa masa de datos y producir conocimientos valiosos que apoyen la toma de decisiones. Sin embargo, en los últimos años se ha hecho necesario aumentar la velocidad en que se procesan esos datos así como la cantidad de los mismos. Entre las etapas más significativas del proceso de Minería de texto se encuentran: (Reyes, y otros, 2007)

- Adquisición de los datos
- Normalización de los textos
- Filtrado
- Análisis
- Visualización

En el mundo existen empresas y organizaciones que de una manera u otra realizan inteligencia empresarial. Para ello se han apoyado tanto de desarrollos propios, como de procedimientos y metodologías con herramientas de software propietarios. Al respecto, existen varias soluciones con el

Capítulo I. Fundamentación Teórica

fin de procesar y analizar cuantitativamente la información, las que fueron identificadas por la búsqueda bibliográfica llevada a cabo, para la realización del trabajo.

1.2 Análisis de soluciones similares para la extracción de las características

En la actualidad es muy común encontrarse con una gran cantidad de sistemas que utilizan métodos de la Minería de texto en:

- Sistemas de recuperación de información
- Sistemas de detección de intrusiones en redes
- Biología molecular y bioinformática
- Minería Web
- Ingeniería genética
- Sistemas de Vigilancia Tecnológica
- Análisis y organización de noticias
- Detectores de lenguajes
- Buscadores Web
- Aplicaciones de inteligencia militar

Con el desarrollo de la informática se ha hecho indispensable el uso de las herramientas para la extracción de característica en los sistemas de vigilancia tecnológica debido a la gran cantidad de información que se publica a diario, resultando difícil extraerla de forma manual. Para el desarrollo del módulo de extracción de características de las fuentes de información se decide usar la Minería de texto porque la mayor parte de la información que se gestiona en el grupo GIC es textual. Resultando así un complemento idóneo para facilitar una mayor comprensión de las necesidades de los clientes y permitir el seguimiento de la competencia. Siendo esta herramienta clave para el grupo de GIC que se dedica a la vigilancia tecnológica.

1.2.1 Herramientas internacionales de apoyo a la vigilancia tecnológica

Las herramientas internacionales usadas para los proyectos de vigilancia tecnológica constituyen un factor clave a la hora de traducir la información del entorno de la toma de decisiones. Sin embargo, no es adecuado que estas constituyan el fundamento y base estructural de los procesos desarrollados.

Capítulo I. Fundamentación Teórica

Según la bibliografía consultada se encontró en países como Colombia, Estados Unidos, Francia, España, México, Japón, un total de 21 herramientas que se dedican al apoyo de la vigilancia tecnológica. Dentro de ellas 14 utilizan Minería de texto pero una sola realiza el proceso de extracción de características de las fuentes de información que necesita implementar el grupo GIC.

Es conveniente evaluar la pertenencia de las herramientas de software que se pueden utilizar en el proceso de vigilancia tecnológica, en función de los entornos productivos y competitivos de usuarios finales. Para efectos demostrativos en la tabla 1 se tomaron cuatro tipos de instituciones que realizan vigilancia tecnológica: las PyME⁵, los CDT⁶, las universidades y las organizaciones de carácter nacional. Se realiza análisis comparativo entre las herramientas mostrando las cuatro fases que contiene el ciclo de vigilancia tecnológica.

La tabla que se presenta a continuación ilustra mejor los datos que fueron tomados como base de los cuadros comparativos propuestos por los trabajos “Valoración, selección y pertinencia de herramientas de software utilizadas en vigilancia tecnológica” (Vargas, y otros, 2006) y “Herramientas de Software especializadas para Vigilancia Tecnológica e Inteligencia Competitiva” (Palop, y otros, 2009). Además se realiza una nueva propuesta en tanto se han actualizado los costos y se han incluido nuevos criterios de comparación, principalmente el del proceso de extracción de características.

Tabla 1. Cuadro comparativo de herramientas internacionales.

Producto	Costo	Información que utiliza	Apoyo al Ciclo de Vigilancia Tecnológica				Inteligencia Minería de texto	Proceso Extracción de características	Actor de uso			
			Planeación	Búsqueda y captación	Procesamiento y análisis	Socialización			Pymes	CDTs	Universidad	Org. Nal
Copemic 6.0	199.95 Euro	Web	No	Si	Si	No	No	No	X	X	X	X
Strategic Finder	457 Euro	Web	Si	Si	Si	Si	No	No		X	X	
C-4-U Scout 1.3	Gratuito	Web	Si	Si	No	No	No	No	X		X	
CI Spider 1.2.1	No Def.	Web	Si	Si	Si	Si	No	No		X	X	
Seekip	Gratuito	Web	Si	Si	No	No	No	No	X	X	X	X
WebFerrer 5.0	29.95 USD	Web	Si	Si	Si	No	No	No	X			
WebSeeker 5.0	29.25 USD	Web	Si	Si	No	No	No	No	X			
TextAnalyst	No Def.	Texto	No	No	Si	No	Si	No				
T-LAB	618 US	Texto	No	No	Si	Si	Si	No		X	X	X
SPSS 14	1499 USD	semi-estructurada	No	No	Si	Si	Si	No			X	X
MindModel 2.1	220 USD	Web - Texto	No	Si	Si	No	Si	No		X	X	

⁵ Pequeñas y Medianas empresas, PyME.

⁶ Centros de desarrollo tecnológico, CDT.

Capítulo I. Fundamentación Teórica

Sonar Profesional	795 USD	Texto	Si	Si	Si	No	Si	No			X	
XLStat 7.5.3	395 USD	Estructurada	Si	No	Si	Si	Si	No			X	
Hamlet	No Def.	Texto	Si	No	Si	Si	Si	No		X	X	
TextPack	500 Euro	Texto	Si	No	Si	Si	Si	No			X	
Tetralogie 6.0	12000 Euro	semi-estructurada	Si	No	Si	Si	Si	No			X	X
Matheo analyzer 3.0	3450 Euro	semi-estructurada	Si	No	Si	Si	Si	No			X	X
Matheo patent 3.0	600 Euro	semi-estructurada	Si	Si	Si	Si	Si	No	X	X	X	X
Aurekal 9.2	No def.	Semi-estructurada	Si	Si	Si	Si	Si	No			X	X
Goldfire	16000 Euro	Web, Texto semi-estructurada	Si	Si	Si	Si	Si	No			X	X
Vigtech	Libre	semi-estructurada	No	Si	Si	No	Si	Si			X	

Fuente: Elaboración propia

Después de realizar un análisis de las herramientas internacionales que utilizan Minería de Texto para los Sistema de Vigilancia Tecnológica, es posible ver como la herramienta informática Vigtech cumple con el proceso de extracción de características de las fuentes de información. Vigtech permite encontrar relaciones cognitivas y sociales en un conjunto de documentos extraídos de una base referencial tal como SCOPUS. Específicamente, la herramienta soporta las actividades de obtención de información de las fuentes de información, extracción de características, cálculo de estadísticas descriptivas, análisis de redes sociales, análisis de redes de palabras clave y visualización, estas son característica que las otras herramientas no tienen.

1.2.2 Herramientas nacionales de apoyo a la vigilancia tecnológica

La existencia de Sistemas de Vigilancia Tecnológica en las instituciones de nuestro país se necesaria para mantener un seguimiento de los nuevos desarrollos. Además para evaluar sus posibles repercusiones para la economía y recomendar los cursos de acción ante ellas, tornándose como una necesidad estratégica y prioritaria para el país.

En el ámbito nacional se han estudiado varios modelos de vigilancia tecnológica, los que están enfocados en los siguientes sectores:

- Educación Superior. (Estévez, 2010)
- Empresa Constructora de Obras de Arquitectura e Industriales de Camagüey. (Betancourt, y otros, 2010)
- Estación Experimental “Indio Hatuey” de Matanzas. (Casas, 2010)
- Empresa Industrial Azucarera. (León, y otros, 2004)
- Instituto de Investigaciones de la Industria Alimentaría. (Batista, y otros, 2003)
- Otras de carácter compilador que de manera genérica enfatizan en las actividades de

Capítulo I. Fundamentación Teórica

innovación y desarrollo. (Solís, y otros, 2008)

Existen en Cuba diversas organizaciones que desde hace más de 15 años practican la vigilancia tecnológica, no solo para sí, sino como parte de su objeto social, a terceros. Esta información ha sido recopilada de las ponencias de los especialistas de estas entidades que se encuentran en las memorias del evento XII Congreso Internacional de Información 2012 como: (IDICT, 2012)

- Biomundi (perteneciente al IDICT, Ministerio de Ciencia Tecnología y Medio Ambiente).
- Oficina Cubana de Propiedad Intelectual (OCPI, CITMA).
- Centro Nacional de Investigaciones Científicas (CNIC, perteneciente al Ministerio de Educación Superior).
- DELFOS (Ministerio de Informática y Comunicaciones).

Estudios previos a esta investigación revelan el estado y alcance de los proyectos de vigilancia tecnológica emprendidos por las universidades cubanas. Ellas son:

- Universidad de la Habana (UH). (Vidal, y otros, 2008)
- Universidad de Pinar del Río (UPR). (Giráldez, y otros, 2008)
- Universidad de Ciencias Informáticas (UCI). (Pascual, 2009)
- Universidad Central de las Villas (UCLV). (Estévez, 2010)
- Universidad Agraria de la Habana. (UNICA, 2009)

A continuación, se muestra en la tabla 2 las herramientas que según la literatura consultada existen en Cuba. Se estudiaron un total de 9 herramientas que apoyan de una forma u otra al ciclo de vigilancia tecnológica, pero ninguna realiza el proceso de extracción de características de las fuentes de información que necesita el grupo GIC (Cárdenas, 2009).

Tabla 2. Cuadro comparativo de herramientas desarrolladas en Cuba.

Producto	Información que utiliza	Apoyo al Ciclo de Vigilancia Tecnológica				Inteligencia	Proceso	Institución
		Planeación	Búsqueda y captación	Procesamiento y análisis	Socialización	Minería de texto	Extracción de características	
CitlivePro 2.0	Patente, Web, Artículos científicos	Si	Si	No	No	No	No	Bioplantas (Ciego Ávila)

Capítulo I. Fundamentación Teórica

DataSOMining	Artículos científicos	Si	Si	Si	No	No	No	Finlay/UNAM
InfoCam 3.0	Patente, Web, Artículos científicos	Si	Si	Si	Si	No	No	INDER
Mimosa	Patente	Si	Si	No	No	No	No	OCPI
Patdown	Patente	Si	Si	No	No	No	No	CIGET (Villa Clara)
SiVigPat	Patente	Si	Si	Si	Si	No	No	CNIC
ToolInf	Patente, Artículos científicos	Si	Si	Si	Si	No	No	BioMundi
BioMundiPatent	Patente	Si	Si	No	No	No	No	BioMundi
ProIntec	Patente	Si	Si	Si	Si	No	No	Universidad Pinar del Rio

Fuente: Elaboración propia

Finalmente, es importante recordar que las herramientas de software son una excelente forma de fortalecer la función de vigilancia tecnológica. Se deben considerar lo que son, herramientas que facilitan enormemente el trabajo. Conocer las características de estas herramientas brinda la posibilidad de identificar más acertadamente que tipos de fuentes se pueden consultar y los resultados que son factibles de obtener para el Sistema de Vigilancia Tecnológica. Para ver las características de estas herramientas consultar el trabajo de tesis de “Análisis y diseño del Sistema de Vigilancia Tecnológica para la tecnología educativa”.

Valoración sobre las herramientas

Las herramientas internacionales analizadas que apoyan a la vigilancia tecnológica no se pueden utilizar porque la mayor parte de ellas son propietarias, costosas y de uso personal. Excluyendo la herramienta Vigtech que es gratuita, es de las más completas en la extracción de la información pero tiene como restricción el uso exclusivo de SCOPUS, una de las base de datos de resúmenes más grandes del mundo que tiene la limitante que es de pago, por lo que no constituye la solución a la situación problémica del grupo GIC. Las herramientas nacionales estudiadas tampoco pueden ser utilizadas porque no trabajan directamente con el proceso de extracción de características de las fuentes de información. Por lo que ninguna de las herramientas estudiadas a nivel nacional e internacional constituye la solución para las necesidades del grupo GIC.

Aunque si se tuvo en cuenta para la implementación del módulo, la descripción de cómo se realiza la extracción de características en la herramienta Vigtech, tomándose como base para el estudio de la implementación del módulo. Logrando de esta manera que el mismo se encuentre a la altura de los sistemas de su tipo a nivel nacional e internacional utilizando términos modernos como la Minería de texto.

Capítulo I. Fundamentación Teórica

1.3 Extracción de las características de las fuentes de información para el Sistema de Vigilancia Tecnológica

Con la implementación del módulo para el proceso de extracción de las características de las fuentes de información para el Sistema de Vigilancia Tecnológica se podrán obtener información de los documentos tales como: título, autores, palabras clave, institución, fecha, país y resumen, de forma automática, dejando atrás el procesamiento manual de las fuentes de información.

Fuentes de información

Según Gloria Carrizo⁷ define las fuentes de información como *“los materiales o productos, originales o elaborados, que aportan noticias o testimonios a través de los cuales se accede al conocimiento, cualquiera que éste sea. Estos materiales o productos que constituyen las fuentes de información son huellas, testimonios o conocimientos aportados por el hombre en el discurrir del tiempo y pueden ser restos biológicos, monumentos, documentos, libros o productos de ordenador, todo aquello que suministre una noticia, una información o un dato.”* (Carrizo, y otros, 2009)

Matizando lo anterior, el término fuentes de información se refiere a todo aquello, ya sea una institución, un documento o una persona, que proporciona la información requerida por cualquiera, se haya creado o no con tal fin. También se le denomina fuentes de información a diversos tipos de documentos que contienen datos útiles para satisfacer una demanda de información o conocimiento. Todos los documentos que de una forma u otra difunden los conocimientos propios de un área como educación, salud, artes y humanidades, ciencias exactas y computación.

En lo que atañe al nivel informativo, las fuentes de información se clasifican en primarias, secundarias y terciarias. La aplicación a implementar trabajará directamente con las fuentes de información primarias y dentro de esta las de literatura gris.

Fuentes primarias

“Las fuentes primarias contienen artículos o informes que exponen por primera vez descubrimientos

⁷ Dra. Gloria Carrizo Sainero. Profesora titular de la Universidad de Carlos III, Madrid. Áreas de especialización: Fuentes de información en Ciencias Sociales y Humanidades; Bibliografía.

Capítulo I. Fundamentación Teórica

científicos, observaciones originales o los resultados de la investigación experimental o de campo, los cuales comprenden contribuciones nuevas al conocimiento, su publicación establece el registro en forma permanente del progreso de la ciencia, la tecnología, las humanidades y las artes. La función de las fuentes primarias es de difundir el conocimiento nuevo, permitiendo su evaluación en la comunidad general.” (Informativos, 2012)

Literatura gris

Emilia Currás⁸ define que la literatura gris *“puede obtenerse públicamente, no siendo convencional en su contenido, no estando bien controlada su publicación, ni siendo accesible por los canales normales de distribución lo que la hace difícil de obtener y de localizar, además de ello son documentos de tipo muy variado, que van desde las publicaciones no revisadas hasta los documentos de contenido no muy concreto.”* (Curras, 2010)

En resumen, literatura gris es aquella producida, ya sea en ámbitos gubernamentales, académicos, comerciales, empresariales, industriales, en formato impreso y electrónico, que no está controlada por los editores comerciales y su distribución se restringe a un público científico y grupos cerrados. Según Lara Rey Vázquez en su informe sobre la vigilancia tecnológica establece la tipología de documentos que abarca la literatura gris como: (Vázquez, 2009)

- Informes públicos y privados con información científica, técnica, económica y social.
- Tesis no publicadas en el mercado editorial.
- Comunicaciones a congresos no publicadas en actas disponibles en el mercado editorial.
- Normas y recomendaciones de carácter técnico, marcas y patentes.
- Traducciones inéditas.
- Algunas publicaciones periódicas no publicadas comercialmente.
- Algunos documentos oficiales de ámbito restringido o muy limitado.
- Documentación técnica publicitaria.

⁸ Emilia Currás Puente es Doctora en Ciencias Químicas, Documentalista Científica, Profesora Titular de la Universidad Autónoma de Madrid y Presidenta de Honor de ISKO-España. Pionera en la reivindicación del estatuto profesional de los documentalistas en España. Emilia.curras@uam.es

Capítulo I. Fundamentación Teórica

- Informes de laboratorios, estudios de mercado.
- Programas de ordenador de circulación restringida entre grupos de personas.
- Otros documentos que se ajusten a las características indicadas.

Esta herramienta se extenderá hacia el procesamiento de las fuentes de información primarias.

Tomando como principal insumo las de tipo literatura gris como tesis, artículos e informes, a las que se les extraerán las características ya mencionadas.

Características

Las características no son más que la información propia de un sujeto, objeto o estado y que lo define como tal. Las características contienen una serie de aspectos o variables que configuran el estado e identidad de una entidad en particular, que puede ser tanto una persona como un animal, un vegetal, un objeto o incluso una condición o escenario.

De todas las definiciones consultadas se pueden extraer varios puntos cruciales que resultaron ser útiles para la realización de la aplicación para el Sistema de Vigilancia Tecnológica del grupo GIC, de tal forma que resulte posible concluir que, característica es toda aquella información descriptiva sobre el contexto, calidad, condición de un recurso, dato u objeto que tiene la finalidad de facilitar su recuperación, autenticación, evaluación, preservación. Para el desarrollo de esta aplicación se tendrán presentes metodologías, herramientas y tecnologías que ayudaran al desarrollo de la solución.

1.4 Metodologías, lenguajes y herramientas a utilizar

Después de analizar el trabajo de diploma “Análisis y diseño del Sistema de Vigilancia Tecnológica para la tecnología educativa”, se determinó hacer algunos cambios (los que serán descriptos a continuación y sus razones) en el uso de la metodología y tecnologías propuestas para la realización del mismo, en los lenguajes no se realizó ningún cambio.

1.4.1 Metodología de desarrollo de software

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de software. Van indicando paso a paso todas las actividades a

Capítulo I. Fundamentación Teórica

realizar para lograr el producto informático deseado, guiando qué personas deben participar en el desarrollo de las actividades y qué papel deben de tener. Además, detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

En el trabajo de diploma de “Análisis y diseño del Sistema de Vigilancia Tecnológica para la tecnología educativa” se había determinado utilizar la metodología de Proceso Unificado de Rational (del inglés Rational Unified Process, RUP) y a raíz de que el sistema se divide en los tres módulos a implementar estos pasan a ser proyectos pequeños, por lo que se decide cambiar a la metodología ágil Programación Extrema.

Programación Extrema (XP)

Kent Beck⁹ uno de los creadores de Programación Extrema (del inglés Extreme Programming, XP) define esta metodología como “una forma de desarrollar software: liviana, de bajo riesgo, flexible, predecible, científica y divertida”. (Well, 2006)

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo. Se basa en retroalimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios.

XP propone un ciclo de vida dinámico, donde se admite expresamente que, en muchos casos, los clientes no son capaces de especificar sus requerimientos al comienzo de un proyecto. La metodología XP se centra en satisfacer al cliente y potenciar al máximo el trabajo en equipo. Si bien el ciclo de vida de un proyecto XP es muy dinámico, se puede separar en las siguientes fases. (Joskowicz, 2008)

- Exploración
- Planificación
- Iteraciones

⁹ Kent Beck es ingeniero de software estadounidense, uno de los creadores de las metodologías de desarrollo de software de programación extrema y el desarrollo guiado por pruebas, también llamados metodología ágil.

Capítulo I. Fundamentación Teórica

➤ Producción

XP tiene una serie de características las que se pueden ver en (Fernández, y otros, 2012). Se decidió utilizar XP debido a que se adapta en gran medida tanto al tipo de proyecto a desarrollar como a las condiciones de trabajo. A continuación se exponen varias de las razones que llevaron al uso de esta metodología.

- El proyecto es pequeño. XP está concebida para ser utilizada dentro de proyectos pequeños. No existe un contrato previo especificando tiempo, recursos y alcance. Para el desarrollo del sistema no se dispone de un contrato con un presupuesto ni un alcance previamente definidos, puesto que es un proyecto para el uso interno de la UCI y será llevado a cabo por un programador perteneciente a la misma.
- Las funcionalidades del sistema cambian frecuentemente. Con la aceptación de nuevos tipos de proyectos, con estructura y requerimientos disímiles, el módulo debe cambiar y ampliar sus funcionalidades de forma que sea capaz de adaptarse a cada nueva situación. Uno de los principios básicos de XP es que el cambio frecuente de los requerimientos es algo normal en el proceso de desarrollo. Esta metodología se adapta perfectamente a los proyectos cuyos requerimientos cambian a menudo.
- El cliente forma parte del equipo de desarrollo. Mediante la aplicación de XP se puede lograr una retroalimentación mayor y lograr un producto que satisfaga sus necesidades. Debido al corto tiempo de entrega planteado y a los continuos cambios de requerimientos.

1.4.2 Lenguajes

Se realiza una pequeña descripción de los lenguajes, ya que en el trabajo de diploma "Análisis y diseño del Sistema de Vigilancia Tecnológica para la tecnología educativa" se describen estos.

HTML 5

Es el lenguaje utilizado para la creación de páginas web, Lenguaje para el Formato de Documentos de Hipertexto (del inglés HyperText Markup Language, HTML). Es decir, los documentos HTML no son de texto normal, sino de hipertexto ya que en el propio documento aparecen enlaces a otros. Más concretamente es el lenguaje con el que se escriben la mayoría de páginas web. HTML es "un

Capítulo I. Fundamentación Teórica

lenguaje reconocido universalmente y que permite publicar información de forma global" (Pérez, 2009). Desde su creación, este lenguaje ha pasado de ser un utilizado exclusivamente para crear documentos electrónicos. Ver más en (Fernández, y otros, 2012).

CCS 3

Hojas de estilo en cascada (del inglés Cascading Style Sheets, CCS), *"es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas Web complejas."* (Web, 2010)

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo. Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes. Ver más en (Fernández, y otros, 2012).

JavaScript

Es un lenguaje utilizado para realizar acciones dentro del ámbito de una página web, permitiendo el desarrollo de interfaces de usuario mejoradas. Es un lenguaje de programación interpretado, multiplataforma, orientado a eventos, que se utiliza principalmente para crear páginas web dinámicas.

Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario. Es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. Los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

Ofrece la posibilidad de crear *"aplicaciones en línea o modificar páginas web en tiempo real, por ejemplo, cambiar el aspecto de la página web. Actualmente, todos los navegadores incluyen JavaScript y es uno de los lenguajes más populares para la Web"* (Lidia, y otros, 2010). Ver más en (Fernández, y otros, 2012).

Capítulo I. Fundamentación Teórica

PHP 5

Es un lenguaje multiplataforma, ejecutado en el servidor. Una de sus características más potentes es la conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destacándose su conectividad con PostgreSQL. Como producto de código abierto, goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren. “El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar sus capacidades”. (Web, 2010)

1.4.3 Herramientas a utilizar

En las herramientas se mantiene la seleccionada NetBeans y se incluye Bootstrap que es un framework pero se utiliza como una herramienta de apoyo al framework CodeIgniter que sigue la línea del mismo.

NetBeans IDE 7.0.1

NetBeans está completamente escrito en Java, por lo que puede ser utilizado desde cualquier sistema operativo compatible con la máquina virtual de Java. Permite el desarrollo de aplicaciones de escritorio, web y móviles. Es un producto libre y gratuito sin restricciones de uso.

Es una herramienta para programadores pensada para escribir, compilar y ejecutar programas. Su misión consiste en evitar tareas repetitivas, facilitar la escritura correcta de código, disminuir el tiempo de depuración e incrementar la productividad del desarrollador. (NetBeans, 2011)

Bootstrap 2.1.1

Bootstrap es un framework desarrollado por Twitter que simplifica muchísimo a la hora de aplicar estilos a los desarrolladores web. Combina CSS y JavaScript asegurando una total compatibilidad con los navegadores web más extendidos. Fomenta las buenas prácticas de diseño y desarrollo web, es un potente framework con numerosos componentes web. (Maekotto, 2013)

Las principales características de Bootstrap son: (Genbeta, 2013)

Capítulo I. Fundamentación Teórica

- Ofrece una serie de plantillas CSS y ficheros JavaScript que permiten integrar el framework de forma sencilla y potente en nuestros proyectos webs.
- Permite crear interfaces que se adapten a los diferentes navegadores.
- Se integra perfectamente con las principales librerías JavaScript.
- Es un framework ligero que se integra de forma limpia en nuestro proyecto actual.

1.4.4 Gestor de base de datos

PostgreSQL 9.1.9

Es un sistema de gestión de bases de datos relacionales de código abierto (del inglés Open Source), gratuito y que al tener licencia de tipo BSD, permite manejar libremente el código fuente del gestor de bases de datos PostgreSQL, mejorando u optimizando su código. Incluso se permite redistribuirlo como producto comercial y combinarlo con herramientas de licencia propietaria. (Arévalo, 2010)

Ventajas:

- Posee manejo y control de transacciones para asegurar la consistencia de los datos.
- Soporta los tipos de datos, cláusulas, funciones y comandos de tipo estándarSQL92/SQL99 y extendidos propios de PostgreSQL. Los tipos de datos internos han sido mejorados incluyendo nuevos tipos.
- PostgreSQL puede operar sobre distintas plataformas incluyendo GNU/Linux, Unix, MacOSX, Solaris y Windows.
- Posee un buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios, permisos y contraseñas.
- Posee una gran capacidad de almacenamiento.
- Posee ciertas características de orientación a objetos, como la herencia entre tablas.
- Tiene una buena escalabilidad ya que es capaz de ajustarse al número de micro-procesadores y a la cantidad de memoria que posee el sistema de forma óptima, soportando una mayor cantidad de peticiones simultáneas a la base de datos de manera correcta.

1.5 Framework para el desarrollo del software

Con el objetivo de acelerar el proceso de creación del producto y promover buenas prácticas de

Capítulo I. Fundamentación Teórica

desarrollo como el uso de patrones de diseño, se decidió utilizar un framework. Un framework es un concepto sumamente genérico, se refiere a un ambiente de trabajo y ejecución (Frameworks, 2007). En general los framework son soluciones completas que contemplan herramientas de apoyo a la construcción de ambiente de trabajo o desarrollo.

Ventajas de utilizar un framework para el desarrollo de un software.

- El desarrollo rápido de aplicaciones. Los componentes incluidos en un framework constituyen una capa que libera al programador de la escritura de código de bajo nivel.
- La reutilización de componentes software al por mayor. Los framework son los paradigmas de la reutilización.
- El uso y la programación de componentes que siguen una política de diseño uniforme. Un framework orientado a objetos logra que los componentes sean clases que pertenezcan a una gran jerarquía de clases, lo que resulta en bibliotecas más fáciles de aprender a usar.

De los framework de desarrollo web que utilizan PHP, los que reunía las características en cuanto a documentación y una comunidad amplia de desarrolladores eran: CodeIgniter, Symfony y Zendframework, en estos se centró el autor del presente trabajo para realizar la selección. Se tuvieron varios criterios (curva de aprendizaje, tiempo para captación y el mapeo objeto-relación) de selección los que fueron aplicados a cada uno de estos framework, que se podrán ver en la siguiente tabla.

Tabla 3: Comparación entre los distintos framework de desarrollo.

Framework	Curva de aprendizaje	Tiempo para capacitación	Mapeo Objeto – Relación (ORM)
CodeIgniter	Baja	Media	No
Symfony	Alta	Alta	Si
Zendframework	Alta	Alta	Si

En un proyecto donde el tiempo de entrega es corto, los principales aspectos a tener en cuenta son: la curva de aprendizaje y el tiempo de capacitación, sacrificando la ventaja de tener un framework con ORM. Una de las características importantes que se tuvo en cuenta es la ligereza en cuanto a carga de componentes donde la de CodeIgniter es mucho más baja que la de Symfony y Zendframework.

Capítulo I. Fundamentación Teórica

Por estas razones se selecciona el framework de desarrollo PHP CodeIgniter. A continuación se describen los elementos más generales sobre el framework seleccionado.

Framework CodeIgniter 2.1.3

CodeIgniter es un framework en PHP para la creación de cualquier tipo de aplicación web. Es un producto de código libre, independiente de uso para cualquier aplicación. CodeIgniter implementa el proceso de desarrollo llamado Modelo Vista Controlador (del inglés Model View Controller, MVC), que es un estándar de programación de aplicaciones, utilizado tanto para hacer sitios web como programas tradicionales, que hacen que el proceso de desarrollo sea más rápido.

Como cualquier otro framework, CodeIgniter contiene una serie de librerías que sirven para el desarrollo de aplicaciones web y además propone una manera de desarrollarlas que se debe seguir para obtener provecho de la aplicación. Esto marca una manera específica de codificar las páginas web y clasificar sus diferentes scripts, que sirve para que el código esté organizado y sea más fácil de crear y mantener. CodeIgniter se crean utilizando el Modelo-Vista-Controlador. El modelo es el esquema de su base de datos y las funciones necesarias para acceder al servidor de la misma. La vista es lo que se ve cuando se visita un sitio web o utiliza una aplicación web. Los controladores son los procesos de servidor web y los recursos necesarios para mostrar una página web y aplicaciones de servidor de acceso.

Con CodeIgniter se desarrollan aplicaciones web más profesionales y con código reutilizable, con la diferencia a otros framework de PHP, CodeIgniter está creado para que sea fácil de instalar en cualquier servidor y de empezar a usar. Además muchas de sus utilidades y modos de funcionamiento son opcionales, lo que hace que se goce de mayor libertad a la hora de desarrollar sitios web.

Algunas características de CodeIgniter. (Álvarez, 2010)

- **Versatilidad:** es la característica principal de CodeIgniter, en comparación con otros framework PHP. CodeIgniter es capaz de trabajar la mayoría de los entornos o servidores, incluso en sistemas de alojamiento compartido, donde solo se tiene un acceso por FTP para enviar los archivos al servidor y donde no tiene acceso a su configuración.

Capítulo I. Fundamentación Teórica

- **Compatibilidad:** codelgniter es compatible con la versión PHP 5, lo que hace que se pueda utilizar en cualquier servidor, incluso en algunos antiguos. Por supuesto, funciona correctamente también en PHP 4.
- **Facilidad de instalación:** no es necesario más que una cuenta de FTP para subir Codelgniter al servidor y su configuración se realiza con apenas la edición de un archivo, donde se debe escribir cosas como el acceso a la base de datos. Durante la configuración no se necesita acceso a herramientas como la línea de comandos, que no suelen estar disponibles en todos los alojamientos.
- **Ligereza:** el núcleo de Codelgniter es bastante ligero, lo que permite que el servidor no se sobrecargue interpretando o ejecutando grandes porciones de código. La mayoría de los módulos o clases que ofrece se pueden cargar de manera opcional, solo cuando se van a utilizar realmente.
- **Documentación tutorializada:** la documentación de Codelgniter es fácil de seguir y de asimilar, porque está escrita en modo de tutorial. Esto facilita mucho la referencia rápida, cuando ya se sabe acerca del framework y se quiere consultar sobre una función o un método en concreto, pero para iniciar sin duda se agradece mucho.

Sin duda, lo más destacable de Codelgniter es su accesibilidad, ya que se puede utilizar en la mayor gama de entornos. Esta es la razón por la que se ha elegido este framework PHP y por todas las características ya descritas, principalmente que su curva de aprendizaje es baja.

Conclusiones parciales

Una vez concluido este capítulo sobre la fundamentación teórica de la Minería de texto y dentro de esta, la extracción de características de las fuentes de información del grupo GIC se concluye que:

- A partir del estudio de los conceptos y definiciones relacionados con el objeto de estudio y campo de acción de la investigación, se pudo llegar a comprender mejor lo referente a la Minería de texto y el proceso de extracción de características para las fuentes de información.
- Con el estudio de algunas herramientas que apoyan a la vigilancia tecnológica en el proceso de extracción de características, tanto en el ámbito nacional como internacional se valoraron

Capítulo I. Fundamentación Teórica

las tendencias actuales. Se obtuvo un acercamiento de como realizaban el proceso de extracción de características de las fuentes de información en dichas herramientas.

- El análisis y estudio de las metodologías, tecnologías y herramientas de desarrollo de software, permitieron sentar las bases para la selección de las apropiadas a utilizar en el desarrollo del sistema.

Capítulo II. Exploración y planificación

CAPÍTULO 2: EXPLORACIÓN Y PLANIFICACIÓN

Introducción

En este capítulo se describen las fases de exploración y planificación de la metodología XP, propuestas por José Joskowicz en su libro “Reglas y Prácticas en eXtreme Programming” (Joskowicz, 2008). Se realizará una descripción de la propuesta de solución, de las funcionalidades y características del sistema para el desarrollo de un módulo capaz de extraer características de las fuentes de información para facilitar el trabajo al grupo GIC.

Diagnóstico inicial

Para el diagnóstico inicial se le realizó una entrevista a la Lic. Yenieris Moyares Norchales responsable del grupo de GIC del centro FORTES. Sobre esta base, se decidió hacer un rediseño a lo planteado en la propuesta de solución del trabajo de diploma de “Análisis y diseño de un Sistema de Vigilancia Tecnológica para la tecnología educativa” (Fernández, y otros, 2012). Ya que surgen otras necesidades en el grupo y que son importantes resolver.

El grupo GIC necesita que la extracción de características como parte del proceso de captación se realice de forma automática a las siguientes fuentes de información: tesis, informes y artículos científicos. Además que las características a extraer sean: título, autores, institución, país, fecha, resumen y palabras clave. Siendo estas las características más importantes con las que el grupo necesita trabajar.

2.1 Descripción y características de la propuesta de solución

En el capítulo uno se realizó una investigación asociada a la Minería de texto y su proceso de extracción de información para tener una base de conocimiento a la hora de implementar el módulo. Este proceso de extracción de características de las fuentes de información le facilita al grupo GIC captar información. Actualmente en el grupo GIC para realizar el proceso de captación de información para apoyar a la vigilancia tecnológica en el centro FORTES, tiene que extraer manualmente las características de un gran número de fuentes de información. Trayendo consigo que este proceso sea lento y engorroso, obviándose información que no se descubre mediante la extracción realizada por los especialistas del Grupo.

Capítulo II. Exploración y planificación

2.1.1 Propuesta del sistema

La solución a lo antes planteado está en la implementación del módulo que realice el proceso de extracción de características de las fuentes de información de forma automática y las guarde en una base de datos permitiendo ser consultadas por los interesados. La creación del módulo para el proceso de extracción de características de las fuentes de información permitirá, mediante una interfaz visual, gestionar las fuentes de información y extraerle las características como: título, autores, país, institución, fecha, palabras clave y resumen, los que se mostrarán mediante una interfaz cuando sean extraídos de la fuente de información, para su aprobación.

Esto le permitirá al especialista, revisar las características antes de ser guardadas en la base de datos. Para que el usuario pueda buscar los agrupamientos naturales de las fuentes de información por esos indicadores.



Figura 1. Esquema del módulo para la extracción de características de las fuentes de información.

Fuente: Elaboración propia.

2.1.2 Funcionalidades del sistema

Las funcionalidades con que debe cumplir la aplicación a implementar son:

FS-1: Gestionar características de las fuentes de información.

Capítulo II. Exploración y planificación

FS-1.1: Extraer características de las fuentes de información.

FS-1.2: Modificar características de las fuentes de información.

FS-1.3: Mostrar características de las fuentes de información.

FS-1.4: Eliminar características de las fuentes de información.

FS-2: Buscar fuentes de información.

FS-3: Autenticar Usuarios.

FS-4: Gestionar Usuario.

Sobre las características del sistema se retoman las mismas planteadas en el trabajo de diploma “Análisis y diseño del Sistema de Vigilancia Tecnológica para la tecnológica educativa”. A continuación se le da inicio a la fase de exploración que desarrolla la metodología XP.

2.2 Fase de exploración

En la fase exploración, los clientes plantean todas las historias de usuario que son de interés para la primera entrega del producto. Mientras el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se hacen pruebas a las diferentes tecnologías y se construye un prototipo para explorar las posibles arquitecturas del sistema. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología. (Gómez, y otros, 2011)

2.2.1 Personal relacionado con el sistema

El personal relacionado con la aplicación son todas aquellas personas que harán uso de la misma.

Tabla 4. Descripción del personal relacionado con el sistema.

Personal relacionado	Justificación
Administrador	Persona encargada de gestionar los usuarios y darle permisos.
Especialista	Es la persona que gestiona las características de las fuentes de información.
Usuario	Persona que realiza consultas a las características extraídas sin la necesidad de entrar al sistema.

Capítulo II. Exploración y planificación

2.2.2 Historias de usuario

Las Historias de Usuario (del inglés User History, HU) según Daniel H. Steinberg y Daniel W. Palmer en su libro Extreme Software Engineering son “una forma rápida de administrar los requisitos de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos”. (Steinberg, y otros, 2011)

Es la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas en las que el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas. (Beck, 1999)

Las HU son similares a los casos de uso, son una descripción de las necesidades funcionales. Usadas para la estimación de tiempo en la planificación de las liberaciones. “Las historias de usuario son más “amigables” que los casos de uso formales” (Villegas, 2011). Unas de las principales ventajas de las historias de usuario es que permiten responder rápidamente a los requisitos cambiantes. Las historias de usuario que se describen seguidamente en las tablas, abordan más sobre las funcionalidades con las que contará el módulo para la extracción de características de las fuentes de información.

Tabla 5. Historia de usuario “Extraer características de las fuentes de información”.

Historia de usuario		Número	HU_1
Nombre:	Extraer características de las fuentes de información		
Actor: Especialista	Iteración asignada: 1		
Prioridad negocio: Alta	Puntos estimados: 3 semanas		
Nivel de complejidad: Alto	Puntos reales: 13 días		
Descripción: El sistema debe permitir extraer características de las fuentes de información según el tipo.			
Observaciones: Esta HU da cumplimiento a la funcionalidad FS-1.1.			
Prototipo de interfaz: Ver anexo 2.			

Capítulo II. Exploración y planificación

Tabla 6. Historia de usuario “Modificar características de las fuentes de información”.

Historia de usuario		Número	HU_2
Nombre:	Modificar características de las fuentes de información		
Actor: Especialista	Iteración asignada: 2		
Prioridad negocio: Alta	Puntos estimados: 1 semana		
Nivel de complejidad: Medio	Puntos reales: 5 días		
Descripción: El sistema debe permitir cambiar características de las fuentes de información según sea necesario.			
Observaciones: Esta HU da cumplimiento a la funcionalidad FS-1.2			
Prototipo de interfaz: Ver anexo 2.			

Tabla 7: Historia de usuario “Mostrar características de las fuentes de información”.

Historia de usuario		Número	HU_3
Nombre:	Mostrar características de las fuentes de información		
Actor: Especialista	Iteración asignada: 2		
Prioridad negocio: Alta	Puntos estimados: 1 semana		
Nivel de complejidad: Medio	Puntos reales: 5 días		
Descripción: El sistema debe mostrar las características de las fuentes de información procesadas.			
Observaciones: Esta HU da cumplimiento a la funcionalidad FS-1.3			
Prototipo de interfaz: Ver anexo 2.			

Tabla 8. Historia de usuario “Eliminar fuentes de información”.

Historia de usuario		Número	HU_4
Nombre:	Eliminar fuentes de información		
Actor: Especialista	Iteración asignada: 2		
Prioridad negocio: Alta	Puntos estimados: 1 semana		
Nivel de complejidad: Medio	Puntos reales: 5 días		
Descripción: El sistema debe eliminar las características de las fuentes de información procesadas.			
Observaciones: Esta HU da cumplimiento a la funcionalidad FS-1.4			

Capítulo II. Exploración y planificación

Prototipo de interfaz: Ver anexo 2.

Tabla 9. Historia de usuario “Buscar fuentes de información”.

Historia de usuario		Número	HU_5
Nombre:	Buscar fuentes de información		
Actor: Usuario	Iteración asignada: 3		
Prioridad negocio: Media	Puntos estimados: 2 semanas		
Nivel de complejidad: Alto	Puntos reales: 9 días		
Descripción: El sistema debe ser capaz de encontrar las fuentes de información.			
Observaciones: Esta HU da cumplimiento a la funcionalidad FS-2.			
Prototipo de interfaz: Ver anexo 2.			

Tabla 10. Historia de usuario “Autenticar usuario”.

Historia de usuario		Número	HU_6
Nombre:	Autenticar usuario		
Actor: Administrador y especialista	Iteración asignada: 3		
Prioridad negocio: Media	Puntos estimados: 2 semanas		
Nivel de complejidad: Alto	Puntos reales: 6 días		
Descripción: El usuario podrá acceder al sistema según sus privilegios. Los datos asociados son: usuario y contraseña.			
Observaciones: Esta HU da cumplimiento a la funcionalidad FS-3. El usuario queda autenticado con los permisos necesarios. Los diferentes usuarios serán: administrador y especialista.			
Prototipo de interfaz: Ver anexo 2.			

Tabla 11. Historia de usuario “Gestionar usuario”.

Historia de usuario		Número	HU_7
Nombre:	Gestionar usuario		
Actor: Administrador	Iteración asignada: 4		
Prioridad negocio: Media	Puntos estimados: 1 semana		
Nivel de complejidad: Medio	Puntos reales: 5 días		
Descripción: El sistema debe permitir al administrador gestionar usuarios en la aplicación.			

Capítulo II. Exploración y planificación

Observaciones: Esta HU da cumplimiento a la funcionalidad FS-4. El usuario queda eliminado o creado con los permisos necesarios.

Prototipo de interfaz: Ver anexo 2.

Una vez que se culmina la elaboración de las historias de usuario, se comienza con su planificación.

2.3 Fase de planificación

La planificación es una etapa de pocos días, una entrega debe obtenerse en no más de tres meses. La planificación se realiza para que los programadores obtengan una estimación del esfuerzo necesario de cada HU, o sea para determinar el período de tiempo que se puede tardar en la implementación de cada una, que debe de estar entre una y tres semanas. El resultado de esta fase es un Plan de entregas.

2.3.1 Plan de entrega

Tabla 12. Plan de entrega.

Historias de usuario	Iteración			
	<u>1ra</u> 8/3-28/3	<u>2da</u> 29/3-18/4	<u>3ra</u> 19/4-9/5	<u>4ta</u> 10/5-16/5
1. Extraer características de las fuentes de información	1.0	F	F	F
2. Modificar características de las fuentes de información				
3. Mostrar características de las fuentes de información	---	1.0	F	F
4. Eliminar características de las fuentes de información				
5. Buscar fuentes de información	---	---	1.0	F
6. Autenticar Usuario				
7. Gestionar Usuario	---	---	---	1.0

Leyenda:

---: No comenzada.

1.0: Entregada la primera versión.

F: Finalizada.

Capítulo II. Exploración y planificación

2.3.2 Estimación de esfuerzos

Tabla 13. Estimación del esfuerzo por historias de usuario.

Iteración	Historias de usuario	Tiempo
1	Extraer características de las fuentes de información	3 semanas
2	Modificar características de las fuentes de información Mostrar características de las fuentes de información Eliminar características de las fuentes de información	3 semanas
3	Buscar fuentes de información Autenticar Usuario	3 semanas
4	Gestionar Usuario	1 semana

Conclusiones parciales

Una vez concluido este capítulo donde se describen las funcionalidades del sistema, las fases de exploración y planificación del módulo de extracción de características se concluye que:

- La descripción de las características de la propuesta de solución permitió comprender la situación actual del proceso de captación de las fuentes de información que realizan los especialistas del grupo GIC. Esto propició la realización de la propuesta de solución con los principales conceptos asociados al grupo.
- Con la aplicación de las actividades propias de las fases de exploración y planificación de la metodología de desarrollo de software seleccionada, se generaron una serie de artefactos que permitieron definir: las funcionales del sistema y el personal relacionado. Además de las historias de usuario, lográndose una visión clara y objetiva de las funcionalidades, así como el plan de entrega para establecer el cronograma de trabajo.
- Se definió por medio del diagrama de despliegue la distribución física, mediante la cual funcionará la aplicación.

Capítulo III. Implementación y prueba

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA

Introducción

“La metodología XP plantea que la implementación debe realizarse de forma iterativa e incremental, obteniéndose en cada iteración un producto que debe ser probado y mostrado al cliente, logrando así una constante retroalimentación entre el cliente y los desarrolladores” (Jacobson, y otros, 2005). En el presente capítulo se le da inicio a las fases de iteraciones y prueba. Además de la descripción de las tarjetas CRC y los resultados arrojados por las pruebas realizadas a cada funcionalidad implementada.

3.1 Diseño del sistema

Unas de las principales ventajas del uso de la metodología XP según Joskowicz es que “hace especial énfasis en los diseños simples y claros” (Joskowicz, 2008). Por otro lado Manuel Calero Solís define el diseño como “una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar” (Solís, 2006). La metodología XP a diferencia de otras metodologías no requiere de la descripción del sistema por medio de diagramas, sino que hace uso de la técnica de las tarjetas CRC (Contenido, Responsabilidad y Colaboración).

3.1.1 Tarjetas CRC

Para la representación de las clases la metodología XP utiliza las tarjetas CRC. “Las características más sobresalientes de las tarjetas CRC son su simpleza y ductilidad. Una tarjeta CRC no es más que una ficha de papel o cartón que representa a una entidad del sistema”. (Casas, y otros, 2010)

XP estimula el uso de tarjetas CRC como un mecanismo para pensar en el software en un contexto orientado a objetos. Las tarjetas CRC identifican y organizan las clases orientadas a objetos que son relevantes para el incremento actual de software. Las tarjetas CRC son el único producto de trabajo de diseño que se generan como parte del proceso XP. Estas tarjetas se utilizan para estructurar las clases y a su vez definir las responsabilidades sobre las mismas, así como la simulación de escenarios en el sistema.

Capítulo III. Implementación y prueba

Tabla 14. Tarjeta CRC “Documents”.

Tarjeta CRC	
Clase: Documents	
Súper Clase: CI_Controller	
Sub Clase(s):	
Responsabilidades: Se encarga de subir el documento al servidor, convertir el documento pdf a formato txt, carga la información del archivo txt y hace una copia del pdf a su carpeta de destino en dependencia del tipo de documento que sea.	Colaboraciones:

Tabla 15. Tarjeta CRC “Frontpage”.

Tarjeta CRC	
Clase: Frontpage	
Súper Clase: CI_Controller	
Sub Clase(s):	
Responsabilidades: Se encarga de cargar la página donde se van a indexar las fuentes de información y después extraerle las características a las fuentes de información. Crea un temporal según el tipo de fuente de información, guarda el archivo en el servidor y carga los datos. Salva el archivo y los borra del temporal. Carga las páginas de buscar o eliminar. Envía la información extraída a una vista. Carga la página para que el usuario pueda hacer búsqueda.	Colaboraciones: Hace falta colaboración con las clases Documents, TakeDataArticle, TakeDataTesis, TakeDataReport.

Capítulo III. Implementación y prueba

Tabla 16. Tarjeta CRC “ManagerDocuments”.

Tarjeta CRC	
Clase: ManagerDocuments	
Súper Clase: CI_Controller	
Sub Clase(s):	
Responsabilidades: Es la que se encarga de salvar la información de los documentos, capturar el documento que se va a modificar y el que se va a eliminar. Salva las características modificadas.	Colaboraciones: Hace falta colaboración con la clase Documents_model.

Tabla 17. Tarjeta CRC “TakeDataArticle”.

Tarjeta CRC	
Clase: TakeDataArticle	
Súper Clase: CI_Controller	
Sub Clase(s):	
Responsabilidades: Es la encargada de obtener la parte del texto con la que se va a trabajar de los artículos. De ahí extrae las características como el título, autor, institución, fecha, resumen y palabras clave. En esta clase se ponen en práctica la técnica de pre-procesamiento de los documentos.	Colaboraciones:

Tabla 18. Tarjeta CRC “TakeDataThesis”.

Tarjeta CRC	
Clase: TakeDataThesis	
Súper Clase: CI_Controller	
Sub Clase(s):	
Responsabilidades: Es la encargada de obtener la parte del texto con la que se va a	Colaboraciones:

Capítulo III. Implementación y prueba

trabajar de las tesis. De ahí extrae las características como el título, autor, institución, fecha, resumen y palabras clave. En esta clase se ponen en práctica la técnica de pre-procesamiento de los documentos.	
---	--

Tabla 19. Tarjeta CRC "TakeDataReport".

Tarjeta CRC	
Clase: TakeDataReport	
Súper Clase: CI_Controller	
Sub Clase(s):	
Responsabilidades: Es la encargada de obtener la parte del texto con la que se va a trabajar de los informes. De ahí extrae las características como el título, autor, institución, fecha, resumen y palabras clave. En esta clase se ponen en práctica la técnica de pre-procesamiento de los documentos.	Colaboraciones:

Tabla 20. Tarjeta CRC "User".

Tarjeta CRC	
Clase: User	
Súper Clase: CI_Controller	
Sub Clase(s):	
Responsabilidades: Es la encargada de gestionar los usuarios.	Colaboraciones:

Tabla 21. Tarjeta CRC "Documents_model".

Tarjeta CRC	
Clase: Documents_model	
Súper Clase: CI_Model	

Capítulo III. Implementación y prueba

Sub Clase(s):	
Responsabilidades: Es la clase modelo encargada de conectarse con la base de datos y hacerle las peticiones o almacenar los datos.	Colaboraciones:

Tabla 22. Tarjeta CRC "User_model".

Tarjeta CRC	
Clase: User_model	
Súper Clase: CI_Model	
Sub Clase(s):	
Responsabilidades: Es la encargada de conectarse con la base de datos.	Colaboraciones:

3.1.2 Diagrama de despliegue

Un diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema, en términos de cómo se distribuye la funcionalidad entre nodos de cómputo (Jacobson, y otros, 2005). Siempre que la creación de los diagramas no implique mayor esfuerzo que la implementación del mismo la metodología XP plantea que para un mejor entendimiento de los métodos de desarrollo de las funcionalidades, se pueden crear diagramas. Siguiendo este principio, se elaboró el diagrama de despliegue que permite apreciar de forma visual cómo se encuentran relacionados físicamente los componentes en la aplicación.

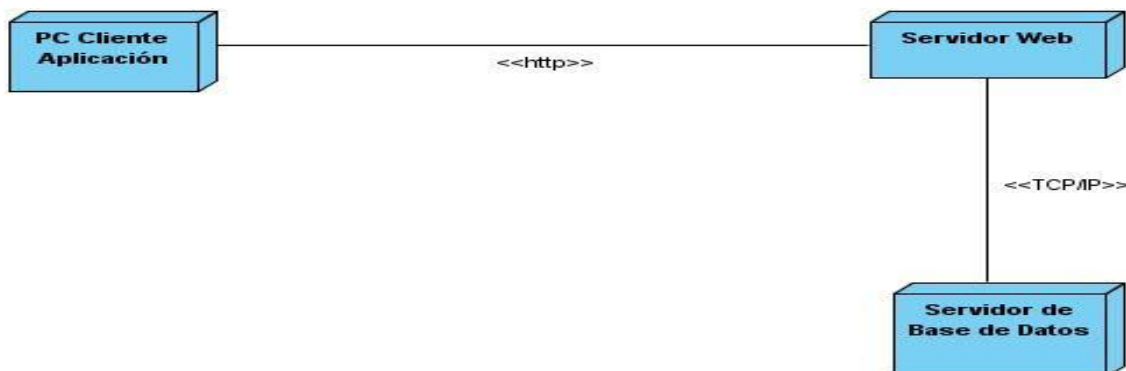


Figura 2. Diagrama de despliegue.

Capítulo III. Implementación y prueba

3.2 Fase de iteraciones

En el ciclo de desarrollo de la metodología de XP las iteraciones son la fase principal. En esta etapa son desarrolladas las funcionalidades del sistema, creando al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración. Al principio de cada iteración se realizan las tareas de análisis, ya que las historias de usuario no tienen suficiente detalle como para permitir sus análisis y desarrollo. Las iteraciones son también utilizadas para medir el progreso del proyecto.

En la primera iteración se puede intentar establecer una arquitectura del sistema, que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias que fueren la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide qué historias se implementarán en cada iteración. Al final de la última iteración el sistema estará listo para entrar en producción. (Letelier, 2008)

Implementación

Al principio de cada iteración se lleva a cabo una revisión del plan de iteraciones y se modifica en caso de ser necesario. Durante el transcurso de una iteración se va implementando las historias de usuario agrupadas a cada iteración. Estas tareas son para el uso estricto de los programadores, por lo que pueden ser escritas en lenguaje técnico y no necesariamente entendible por el cliente.

Por otro lado se vinculan las historias de usuario con tareas concretas de desarrollo, las que se le asignan al programador que asume la implementación de las mismas. Para el desarrollo completo del módulo se definieron cuatro iteraciones de forma tal que al concluir cada iteración se pudiera obtener un producto funcional que cumpliera con las características deseadas por el cliente. Al finalizar cada iteración quedan detalladas las tareas de desarrollo realizadas para cumplir el propósito de las mismas.

3.2.1 Primera iteración

La primera iteración tendrá como objetivo darle cumplimiento a la historia de usuario HU_1, que representa un mayor valor para el cliente, pues con la misma se conformarán las bases de la estructura del negocio. Esta iteración recoge la funcionalidad de extraer las características de las

Capítulo III. Implementación y prueba

fuentes de información siendo la más importante para la aplicación, pues a través de ella se definen aspectos que serán utilizados, luego por las demás funcionalidades. Las tareas por historias de usuario definidas en esta iteración se muestran en el Anexo 3.

Tabla 23. Iteración 1.

Historia de usuario	Tiempo estimado (semanas)	Tiempo real (días)
Extraer características de las fuentes de información	3	15

3.2.2 Segunda iteración

En esta iteración se implementan las historias de usuario HU_2, HU_3 y HU_4. Al concluir esta iteración se deben haber implementado todas las funcionalidades propuestas de forma tal que se le pueda mostrar el resultado al cliente mediante un producto funcional. Esta iteración recoge las funcionalidades de extraer, modificar y eliminar características de las fuentes de información. Las tareas por historias de usuario definidas en esta iteración se muestran en el Anexo 3.

Tabla 24. Iteración 2.

Historia de usuario	Tiempo estimado (semanas)	Tiempo real (días)
Modificar características de las fuentes de información	1	5
Mostrar características de las fuentes de información	1	5
Eliminar características de las fuentes de información	1	5

3.2.3 Tercera iteración

En esta iteración se implementarán las historias de usuario HU_5 y HU_6. Al concluir la misma se debe obtener una segunda versión de la aplicación con nuevas funcionalidades listas para mostrar al usuario. Esta iteración recoge las funcionalidades de buscar fuentes de información y autenticar usuario. Las tareas por historias de usuario definidas en esta iteración se muestran en el Anexo 3.

Capítulo III. Implementación y prueba

Tabla 25. Iteración 3.

Historia de usuario	Tiempo estimado (semanas)	Tiempo real (días)
Buscar fuentes de información	2	9
Autenticar Usuario	1	6

3.2.4 Cuarta iteración

La cuarta iteración y ultima esta centra en desarrollar la historia de usuario HU_7. Al concluir esta iteración se le debe ofrecer al cliente un producto final totalmente funcional que satisfaga las necesidades del mismo. Las tareas por historias de usuario definidas en esta iteración se muestran en el Anexo 3.

Tabla 26. Iteración 4.

Historia de usuario	Tiempo estimado (semanas)	Tiempo real (días)
Gestionar Usuario	1	5

Las iteraciones de desarrollo sobre el sistema, permitieron que al finalizar se obtuviera un producto con todas las restricciones y características deseadas por el cliente. Cabe destacar que el desarrollo de la solución se dividió en cuatro iteraciones para ordenar la implementación, realizándose pequeñas entregas al cliente una vez cada tres semanas, al final de cada una de ellas, para recibir la aceptación del mismo.

3.3 Fase de pruebas

XP es una metodología de desarrollo dirigido por pruebas (del inglés Test Driven Development, TDD). Según Kent Beck *“uno de los pilares de XP es el proceso de pruebas”* (Beck, 1999). En este proceso serán probados todos los componentes del módulo tanto por el equipo de desarrollo como por el cliente. *“Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección”*. (Rodríguez, y otros, 2007)

Capítulo III. Implementación y prueba

El objetivo principal de las pruebas es verificar las funcionalidades del sistema a través de sus interfaces externas comprobando que dicha funcionalidad sea la esperada en función de los requisitos del sistema (SWEBOK, 2009). La metodología XP divide las pruebas del sistema en dos grupos:

Pruebas unitarias: Estas pruebas son hechas durante el proceso de codificación, encargada de verificar el código al terminar cada funcionalidad implementada y diseñadas por los programadores. Garantizando que las funcionalidades requeridas por los clientes estén siendo implementadas correctamente. Las pruebas unitarias son características del periodo de implementación.

Pruebas de aceptación: Se le realizan a las funcionalidades descritas en las historias de usuario, por los clientes y usuarios finales de la aplicación. Una vez que la aplicación tenga resultados satisfactorios con la realización de la prueba de aceptación se podrá considerar que la misma esta apta para su uso.

Se selecciona las pruebas de aceptación porque poseen un peso superior a las unitarias, pues las mismas arrojan a vista de los desarrolladores la satisfacción del cliente. Por lo que son las que más encajan para realizar las pruebas del sistema.

3.3.1 Pruebas de aceptación

Las pruebas de aceptación o pruebas funcionales son pruebas de caja negra que se crean a partir de las historias de usuario (Crispin, y otros, 2002). A una historia de usuario se le puede realizar todas las pruebas de aceptación que sean necesarias para asegurar un resultado satisfactorio, con el objetivo de garantizar que las funcionalidades han sido cumplidas y que el sistema es aceptable. (Wells, 2009)

Los clientes son los que diseñan las pruebas de aceptación, seleccionan las historias de usuario que durante la iteración se le realizaran las pruebas, especificando de cada una los aspectos a examinar, y además evaluando el resultado de cada funcionalidad antes de recibir oficialmente el producto siendo el mayor responsable de verificar cada una de las pruebas y de priorizar la corrección de las pruebas que fallan. A continuación se presentan los casos de prueba que fueron realizados al módulo dividido en las iteraciones correspondiente a cada historia de usuario.

Capítulo III. Implementación y prueba

Iteración 1

Tabla 27. Prueba de aceptación “Extraer características de las fuentes de información”.

Caso de prueba de aceptación	
Código: HU1_PA1	Historia de usuario: 1
Nombre: Extraer características de las fuentes de información.	
Descripción: El sistema debe permitir extraer las características de las fuentes de información.	
Condiciones de ejecución: Las fuentes de información tienen que estar en formato pdf, el servidor montado en una máquina con sistema operativo Ubuntu y el usuario que va a ejecutar la acción debe estar registrado con el rol de especialista.	
Pasos de ejecución: <ol style="list-style-type: none">1. Autenticarse.2. Ir al menú de “Tipos de documentos”.3. Seleccionar el tipo de fuente a cargar.4. Dar clic en el botón “Examinar”.5. Se muestra una ventana para cargar el origen del archivo, donde se selecciona la fuente de información que se le quiere extraer las características.5. Dar clic en el botón “Cargar documento”.6. Se muestra la ventana con las características extraídas de la fuente de información y una vez confirmadas, se da clic en el botón “Guardar datos” para que las características se guarden o en el botón “Cancelar” para que no se guarden en el sistema.	
Resultados esperados: El sistema debe mostrar un mensaje “Las características se han extraído correctamente”.	
Evaluación de la prueba: Satisfactoria	

Iteración 2

Tabla 28. Prueba de aceptación “Modificar características de las fuentes de información”.

Caso de prueba de aceptación	
Código: HU2_PA2	Historia de usuario: 2

Capítulo III. Implementación y prueba

Nombre: Modificar características de las fuentes de información.
Descripción: El sistema debe permitir cambiar las características de las fuentes de información según sea necesario.
Condiciones de ejecución: Que las características estén guardadas en la base de datos y el usuario que va a ejecutar la acción debe estar registrado con el rol de especialista.
Pasos de ejecución: <ol style="list-style-type: none"> 1. Autenticarse. 2. Ir al menú “Eliminar/Modificar”. 3. Seleccionar en el menú “Tipo de fuente de información”, la fuente a modificar. 4. Dar clic en el botón “Buscar”. 5. Seleccionar el icono “Editar” en la lista de fuente de información que muestra. 6. Se muestra la ventana con todas las características de las fuentes de información y una vez modificadas se da clic en el botón “Guardar cambios” para que las características sean guardadas o en el botón “Cancelar” para que los cambios no sean guardados en el sistema.
Resultados esperados: El sistema debe mostrar un mensaje de confirmación “El documento ha sido editado correctamente”.
Evaluación de la prueba: Satisfactoria

Tabla 29. Prueba de aceptación “Mostrar características de las fuentes de información”.

Caso de prueba de aceptación	
Código: HU3_PA3	Historia de usuario: 3
Nombre: Mostrar características de las fuentes de información.	
Descripción: El sistema debe mostrar las características de las fuentes de información procesadas.	
Condiciones de ejecución: Las características deben estar guardadas en la base de datos y el usuario que va a ejecutar la acción debe estar registrado con el rol de especialista.	
Pasos de ejecución: <ol style="list-style-type: none"> 1. Autenticarse 	

Capítulo III. Implementación y prueba

<ol style="list-style-type: none"> 2. Ir al menú de “Eliminar/Modificar” 3. Seleccionar el tipo de fuente a mostrar. 4. Dar clic en el botón “Buscar”. 5. Dar clic en el botón “Mostrar”
Resultados esperados: Se realiza la operación de mostrar las características satisfactoriamente.
Evaluación de la prueba: Satisfactoria

Tabla 30. Prueba de aceptación “Eliminar características de las fuentes de información”.

Caso de prueba de aceptación	
Código: HU4_PA4	Historia de usuario: 4
Nombre: Eliminar características de las fuentes de información.	
Descripción: El sistema debe eliminar las características de las fuentes de información.	
Condiciones de ejecución: Las características deben de estar en la base de datos y el usuario que va a ejecutar la acción debe estar registrado como especialista del sistema.	
Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Autenticarse 2. Ir al menú “Eliminar/Modificar” 3. Seleccionar el tipo de fuente a eliminar. 4. Dar clic en el botón “Buscar”. 5. Ir al icono “Eliminar”. 	
Resultados esperados: El sistema debe mostrar un mensaje de confirmación “El documento ha sido eliminado correctamente”.	
Evaluación de la prueba: Satisfactoria	

Iteración 3

Tabla 31. Prueba de aceptación “Buscar las características de las fuentes de información”.

Caso de prueba de aceptación	
Código: HU5_PA5	Historia de usuario: 5
Nombre: Buscar fuentes de información.	

Capítulo III. Implementación y prueba

Descripción: El sistema debe ser capaz de encontrar las características y las fuentes de información que se buscan.
Condiciones de ejecución: Las características deben de estar guardadas en la base de datos principalmente las del título y resumen.
Pasos de ejecución: <ol style="list-style-type: none"> 1. Introducir el contenido que se quiere buscar. 2. Seleccionar en el menú “Tipo de documento”, el tipo de fuente que se quiere buscar. 3. Dar clic en el botón “Buscar” 4. Ir al icono “Descargar” de la fuente de información que se quiere obtener. 5. Dar clic en el botón “Aceptar”.
Resultados esperados: Se realiza la búsqueda de características satisfactoriamente.
Evaluación de la prueba: Satisfactoria

Tabla 32. Prueba de aceptación “Autenticar usuario”.

Caso de prueba de aceptación	
Código: HU6_PA6	Historia de usuario: 6
Nombre: Autenticar usuario.	
Descripción: El usuario podrá acceder al sistema según sus privilegios.	
Condiciones de ejecución: El usuario debe estar creado y con privilegios definidos.	
Pasos de ejecución: <ol style="list-style-type: none"> 1. Introducir el usuario y contraseña. 2. Dar clic en el botón “Entrar”. 	
Resultados esperados: El sistema le da acceso, según su permiso.	
Evaluación de la prueba: Satisfactoria	

Iteración 4

Tabla 33. Prueba de aceptación “Agregar usuario”.

Caso de prueba de aceptación	
Código: HU7_PA7.1	Historia de usuario: 7
Nombre: Agregar usuario	

Capítulo III. Implementación y prueba

Descripción: El sistema debe permitir agregar un nuevo usuario y asignarle un rol.
Condiciones de ejecución: El usuarios que va a ejecutar la acción debe estar registrado como administrador del sistema.
Pasos de ejecución: <ol style="list-style-type: none"> 1. Autenticarse 2. Ir al menú “Agregar usuario”. 3. Introducir usuario y contraseña. 4. Seleccionar el tipo de rol que se le asignara al usuario, en el menú “Rol”. 5. Dar clic en el botón “Guardar Usuario”.
Resultados esperados: El sistema debe mostrar un mensaje de confirmación que el usuario quedo registrado en el sistema (“Usuario guardado correctamente”). Permitir que el usuario que ha sido creado pueda acceder a la aplicación según los permisos establecidos.
Evaluación de la prueba: Satisfactoria

Tabla 34. Prueba de aceptación “Modificar usuario”.

Caso de prueba de aceptación	
Código: HU7_PA7.2	Historia de usuario: 7
Nombre: Modificar usuario	
Descripción: El sistema debe permitir modificar un usuario registrado.	
Condiciones de ejecución: El usuarios que va a ejecutar la acción debe estar registrado como administrador del sistema.	
Pasos de ejecución: <ol style="list-style-type: none"> 1. Autenticarse 2. Ir al menú “Eliminar/Modificar”. 3. En la lista de usuarios que se muestra, seleccionar el icono de “Editar”, del usuario que quiere modificar. 4. Se muestra la ventana donde se modifican los datos y una vez modificados se da clic en el botón “Guardar Cambios” para guardar los datos o en el botón “Cancelar” para no guardarlos. 	
Resultados esperados: El sistema debe mostrar un mensaje de confirmación de la	

Capítulo III. Implementación y prueba

modificación (“Usuario editado correctamente”).
Evaluación de la prueba: Satisfactoria

Tabla 34. Prueba de aceptación “Eliminar usuario”.

Caso de prueba de aceptación	
Código: HU7_PA7.3	Historia de usuario: 7
Nombre: Eliminar usuario	
Descripción: El sistema debe permitir eliminar un usuario registrado.	
Condiciones de ejecución: El usuarios que va a ejecutar la acción debe estar registrado como administrador del sistema.	
Pasos de ejecución: 1. Autenticarse 2. Ir al menú “Eliminar/Modificar”. 3. En la lista de usuarios que se muestra, seleccionar el icono de “Eliminar”, del usuario que quiere eliminar.	
Resultados esperados: El sistema debe mostrar un mensaje de confirmación del usuario eliminado del sistema (“Usuario eliminado correctamente”).	
Evaluación de la prueba: Satisfactoria	

Resultado de las pruebas realizadas al sistema.

Los casos de prueba realizados tienen como objetivo detectar la mayor cantidad de errores posibles en el módulo de extracción de características de las fuentes de información, se realizaron 4 iteraciones para validar la propuesta de solución. Estos errores se definen como no conformidades, clasificadas según su importancia en:

- **Significativa:** Aquellas que puedan afectar el funcionamiento del módulo.
- **No significativas:** Las enfocadas en el diseño u otro aspecto que no afecte el funcionamiento de la propuesta de solución.
- **Recomendación:** Las sugerencias emitidas por los probadores.

Capítulo III. Implementación y prueba

Durante las pruebas de aceptación se encontraron 19 no conformidades. Entre las Significativa se encontraban problemas con la extracción de características. Otros fallos considerados como No significativos estaban los errores ortográficos en los nombres de los campos. Las Recomendaciones sugerían cambios y mejoras con respecto al diseño de las interfaces. Todas las recomendaciones fueron aceptadas y su incorporación trajo consigo mejoras a la aplicación. A continuación se muestra la figura del resumen de las no conformidades detectadas.

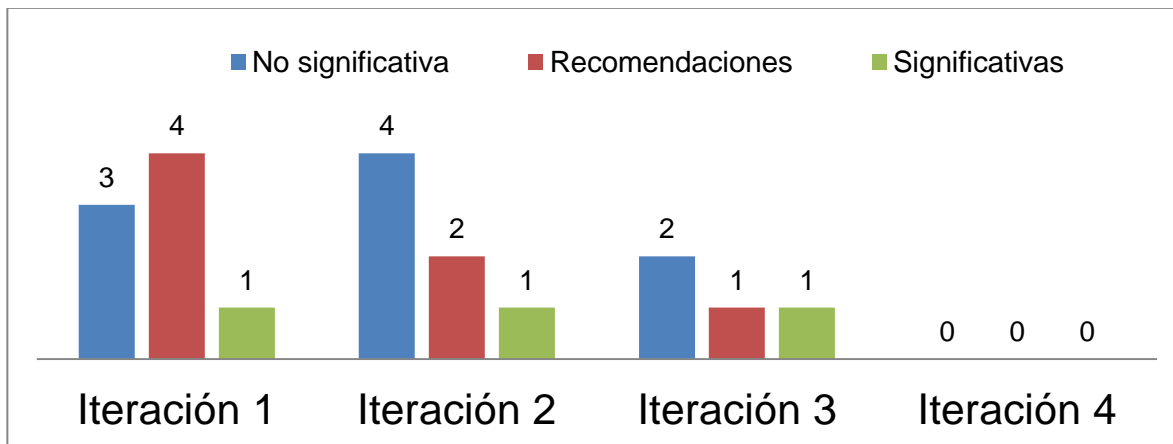


Figura 3: Resumen de las no conformidades.

Conclusiones parciales

En el presente capítulo se describieron las tarjetas CRC, se explicó la importancia de las pruebas, en especial las pruebas de aceptación, con el objetivo de evaluar la calidad del producto y determinar el nivel de conformidad del cliente. Se presentaron los casos de prueba concluyendo que:

- Los casos de prueba permitieron identificar una serie de no conformidades, las que fueron corregidas hasta lograr que el sistema cumpliera de manera correcta con las funcionalidades esperada por el usuario.
- La descripción de las tarjetas CRC permitieron estructurar las clases y a su vez definir las responsabilidades sobre las mismas.

CONCLUSIONES

Una vez finalizada la investigación se concluye que:

- El estudio teórico conceptual realizado determinó utilizar la técnica de Pre-procesamiento de documentos de la Minería de texto, permitiendo así poder identificar los patrones para poder extraer las características de las fuentes de información.
- La elaboración de los artefactos de las fases de exploración y planificación de la metodología XP, permitió definir las funcionalidades del sistema, las historias de usuario y el plan de entrega, aspectos fundamentales para el desarrollo de la propuesta de solución.
- Quedó implementado el módulo propuesto, permitiendo al grupo GIC del centro FORTES eliminar el proceso manual de extracción de características de las fuentes de información, contribuyendo así al proceso de captación.
- Los resultados de los casos de prueba de aceptación realizada al módulo implementado, demostraron que las funcionalidades desarrolladas funcionan correctamente, además de la aceptación esperada por el cliente.

RECOMENDACIONES

Como resultado del proceso de investigación y la aplicación de los casos de pruebas de aceptación, se recomienda para continuar desarrollando el módulo en un futuro, las siguientes ideas.

- Profundizar en el estudio de las técnicas existentes, para aplicar en el proceso de extracción de características de las fuentes de información de tipo "Informe", ya que este no cuenta con un formato estandarizado.
- Estudiar técnicas para extraer las características que se encuentren contenidas en imágenes.

Referencias Bibliográficas

REFERENCIAS BIBLIOGRÁFICAS

- Álvarez, Miguel A. 2010.** Manual de CodeIgniter. Desarrolladores web. [Citado el: 14 de enero de 2013.] <http://www.desarrolloweb.com/manuales/manual-codeigniter.html>.
- Arenas, Lourdes y Anselmo, Moral. 1992.** Automatic indexing of documents". Universidad de Deusto. Bilbao : Nuevas tendencias en inteligencia artificial, 1992. págs. 355-367.
- Arévalo, Mauricio. 2010.** Ventajas y Desventajas de PostgreSQL.
- Batista, D. S. y Sánchez, M. V. G. 2003.** Establecimiento de un sistema de vigilancia científico-tecnológica. s.l. : Acimed.
- Beck, Kent. 1999.** Embracing Change with Extreme Programming. s.l. : Computer, Revista de la IEEE Computer Society. págs. 70-77. 10.
- Betancourt, O Bouza y Gutiérrez, M Álvarez. 2010.** Sistematización de la Vigilancia Científica y Tecnológica en organizaciones cubanas. s.l. : Ciencias de la Información. Vol. (2).
- Blanco, María del Pilar. 2010.** Posible aplicación de la minería de textos a los trabajos de la comisión ministerial de informática. s.l.: Comunicación. pág. 6.
- Cárdenas, Gema Cossio. 2009.** Las herramientas de software para el análisis cuantitativo información. La Habana : Estudio preliminar de aplicaciones desarrolladas en Cuba.
- Carrizo, Gloria Sainero, Irueta, Pilar Goyena y López de Quintana, Eugenio. 2009.** Manual de fuentes de información. Madrid : Confederación Española de Gremios y Asociaciones de Empresarios del Comercio del Libro.
- Casas, L. C. 2010.** Propuesta de sistema de vigilancia tecnológica apropiado para la estación experimental "Indio Hatuey". La habana : Universidad de Matanzas "Camilo Cienfuegos". Vol. 84.
- Casas, Sadra y Reinaga, Héctor. 2010.** Identificación y Modelado de Aspectos Tempranos dirigido por Tarjetas de Responsabilidades y Colaboraciones. [Citado el: 15 de abril de 2013.] <http://www.oocities.org/espanol/profeprog2/INVPAPER25.pdf>.
- Crispin, Lisa y House, Tip. 2002.** Testing Extreme Programming. s.l. : Addison-Wesley.
- Curras, Emilia. 2010.** Sistema experto hipermedia para el reconocimiento, indización y recuperación de literatura gris. págs. 117-130. Vol. 4.
- Eíto, Ricardo Brun y Senso, José A. 2004.** Minería textual. págs. 11-27. Vol. 13.
- Estévez, V Chaviano. 2010.** Diseño de un sistema de vigilancia tecnológica en la educación superior. IntEmpress 2010. La habana Cuba : INFO 2010. Vol. 17.

Referencias Bibliográficas

- Fernández, Yuniel Perdomo y Gámez, Arianna Columbié. 2012.** Análisis y diseño de un Sistema de Vigilancia Tecnológica para la tecnología educativa. La Habana : Universidad de las Ciencias Informáticas.
- Frameworks. 2007.** Frameworks para Java. Desarrollo ágil de software en JAVA. [Citado el: 3 de diciembre de 2013.] <http://daiteSrc.wikispaces.com/Frameworks+para+Java>.
- Genbeta. 2013.** Características principales de Bootstrap. [Citado el: 10 de marzo de 2013.] <http://www.genbetadev.com/frameworks/bootstrap>.
- Giráldez, R Reyes y Díaz, M Pérez. 2008.** PROInTec: un software para el tratamiento inteligente de datos sobre patentes. La Habana : Acimed. 17(5).
- Gómez, Arguello y Wilson, Javier. 2011.** Metodología de desarrollo de software un enfoque práctico y global versión. otcolombia. [Citado el: 15 de febrero de 2013.] <http://www.otcolombia.com/documentos/mds360-1.0.11-beta.pdf>.
- Hearst, Marti. 1999.** Untangling text data mining. Maryland : Association for Computational Linguistics.
- IDICT. 2012.** XII Congreso Internacional de Información. [ed.] Memorias 2002 al 2012. La habana : Palacio de las Convenciones de la Habana. ISBN: 978-959-234-081-7.
- Informativos, Recursos. 2012.** Recursos Informativos. Universidad Autónoma de Yucatán. [Citado el: 3 de noviembre de 2012.] <http://www.tizimin.uady.mx/biblioteca/recursosinfo/fuentesdeinformacion.html>.
- Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2005.** El Proceso Unificado de Desarrollo de Software. [Citado el: 15 de marzo de 2013.] <http://bibliodoc.uci.cu/pdf/8478290362.pdf>.
- Jain, A K, Murty, M N y Flynn, P J. 1999.** Data clustering: a review. s.l. : ACM Computing Surveys. págs. 265-323. Vol. 31.
- Joskowicz, Jose. 2008.** Reglas y Prácticas en eXtreme Programming. España : Doctorado de Ingeniería Telemática de la Universidad de Vigo.
- León, T y González, E. 2004.** Diseño e implementación de un Sistema de Vigilancia Tecnológica en una empresa de escasos recursos. La habana : Interpret Info2004. Vol. 10.
- Letelier, Patricio. 2008.** Metodologías Ágiles en el Desarrollo de Software.
- Lidia, Ana y Mercadillo, Careaga. 2010.** Paradigmas de la Programación: JavaScript y Python. México : Instituto Tecnológico de Teléfonos de México S.C.
- Maekotto. 2013.** Bootstrap. [Citado el: 19 de febrero de 2013.] <http://twitter.github.io/bootstrap/>.

Referencias Bibliográficas

- Montes, Manuel. 2010.** Minería de Texto un nuevo reto computacional. México : Laboratorio de Lenguaje Natural. 07738.
- Murgiondo, Juan Etxeberria. 1995.** Análisis de datos y textos. Madrid. pág. 372.
- Navarro, Miguel Ángel Montero. 2009.** Extracción de conocimiento en bases de datos astronómicas. Sevilla: Memoria del periodo de investigación.
- NetBeans. 2011.** NetBeans. Oracle Corporation. [Citado el: 5 de diciembre de 2012.]
<http://netbeans.org/features/index.html>.
- Palop, Fernando y Sánchez, M. 2009.** Herramientas de Software especializadas para Vigilancia Tecnológica e Inteligencia Competitiva. España.
- Pascual, Y Riverón. 2009.** Diseño de un sistema de vigilancia tecnológica para el Centro de Gestión de la Información y Tecnologías Libres de la Universidad de las Ciencias Informáticas. La Habana : Proyecto de Investigación, Centro de desarrollo de Gestión de la Información y Tecnologías Libres (GEITEL) de la Facultad 10 de la Universidad de las Ciencias Informáticas.
- Pérez, Javier Eguíluz. 2009.** Introducción a XHTML.
- Porter, Alan L y Scott W, Cunningham. 2005.** Tech Mining, Exploiting new technologies for competitive advantage. New Jersey: John Wiley & Sons. 10.1002/0471698466.ch2.
- Reyes, Sady Carina Fuentes y Ruiz, Marina Lobaina. 2007.** Minería de Textos: Aplicación de Web Mining. La Habana: IDICT.
- Rodríguez, M y Ordoñez, M. 2007.** La metodología XP aplicable al desarrollo del software educativo en Cuba. UCI. La Habana.
- Saltamacchia, Homero R. 2006.** Posible aplicación de la minería de textos a los trabajos de la comisión ministerial de informática. Comunicación. pág. 6.
- SAS. 2010.** Getting started with SAS text miner software. Institute Inc. 58859.
- Sinnexus. 2007.** Dataminnig. [Citado el: 15 de noviembre de 2012.]
http://www.sinnexus.com/business_intelligence/datamining.aspx.
- Solís, Manuel Calero. 2006.** Una explicación de la programación extrema. Madrid.
- Solís, Pablos, G, A y Rizo, E. 2008.** Informatización para la toma de decisiones. [ed.] Ministerio de Ciencia Tecnología y Medio Ambiente (CITMA). La Habana: Proyecto CITMA. Vol. 13.
- Steinberg, Daniel H y Palmares, Daniel W. 2011.** Extreme Software Engineering.
- SWEBOK. 2009.** Guide to the Software Engineering Body of Knowledge. IEEE Computer Society.
- UNICA. 2009.** Informe Cumplimiento Estrategia Maestra de Informatización. La Habana : Universidad

Referencias Bibliográficas

Agraria de la Habana.

Valencia, Carlos Mario Cardona. 2005. Utilidad Práctica Derivada De Aplicar Minería de Datos en Algunas Empresas de Medellín.

Vargas, F. W., León, A. y Castellanos, O. 2006. Valoración, selección y pertinencia de herramientas de software utilizadas en vigilancia tecnológica. Vol. 26. 92102.

Vázquez, Lara Rey. 2009. Informe APEI sobre vigilancia tecnológica. Asociación Profesional de Especialistas en Información.

Vidal, J Larramendi y Pérez, A Perdomo. 2008. Observatorio de la Universidad de la Universidad de La Habana. La Habana.

Villegas, Adrian Anaya. 2011. A propósito de programación extrema XP.

Web, Libros. 2010. Introducción a CSS. LibrosWeb.es. [Citado el: 20 de enero de 2013.]

<http://www.librosweb.es/css/>.

Well, Don. 2006. Extreme Programming. [Citado el: 18 de enero de 2013.]

<http://www.extremeprogramming.org/>.

Wells, Don. 2009. Acceptance Tests. Extreme Programming. [Citado el: 16 de febrero de 2013.]

<http://www.extremeprogramming.org/rules/functionaltests.html>.

BIBLIOGRAFÍAS

- AENOR. 2006.** Gestión de la I+D+i: Sistema de VT. UNE 166006 EX. Asociación Española de Normalización y Certificación, Madrid.
- Álvarez, Miguel A. 2010.** Manual de CodeIgniter. Desarrolladores web. [Citado el: 14 de enero de 2013.] <http://www.desarrolloweb.com/manuales/manual-codeigniter.html>.
- Arenas, Lourdes y Anselmo, Moral. 1992.** "Automatic indexing of documents". Universidad de Deusto. Bilbao : Nuevas tendencias en inteligencia artificial, 1992. págs. 355-367.
- Arévalo, Mauricio. 2010.** Ventajas y Desventajas de PostgreSQL.
- Batista, D. S. y Sánchez, M. V. G. 2003.** Establecimiento de un sistema de vigilancia científico-tecnológica. s.l. : Acimed.
- Beck, Kent. 1999.** Embracing Change with Extreme Programming. s.l. : Computer, Revista de la IEEE Computer Society. págs. 70-77. 10.
- Betancourt, O Bouza y Gutiérrez, M Álvarez. 2010.** Sistematización de la Vigilancia Científica y Tecnológica en organizaciones cubanas. s.l. : Ciencias de la Información. Vol. (2).
- Blanco, María del Pilar. 2010.** Posible aplicación de la minería de textos a los trabajos de la comisión ministerial de informática. s.l.: Comunicación. pág. 6.
- Bressán G. 2005.** Vigilancia Tecnológica e Inteligencia Competitiva: usos y aplicaciones en la empresa navarra. [Consultado el 25 de abril 2013].
URL: <http://www.navactiva.com/web/es/documentacion?tag=210>
- Cárdenas, Gema Cossio. 2009.** Las herramientas de software para el análisis cuantitativo información. La Habana : Estudio preliminar de aplicaciones desarrolladas en Cuba.
- Carrizo, Gloria Sainero, Irueta, Pilar Goyena y López de Quintana, Eugenio. 2009.** Manual de fuentes de información. Madrid : Confederación Española de Gremios y Asociaciones de Empresarios del Comercio del Libro.
- Casas, L. C. 2010.** Propuesta de sistema de vigilancia tecnológica apropiado para la estación experimental "Indio Hatuey". La habana : Universidad deMatanzas "Camilo Cienfuegos". Vol. 84.
- Casas, Sadra y Reinaga, Héctor. 2010.** Identificación y Modelado de Aspectos Tempranos dirigido por Tarjetas de Responsabilidades y Colaboraciones. [Citado el: 15 de abril de 2013.]
<http://www.oocities.org/espanol/profeprog2/INVPAPER25.pdf>.
- Crispin, Lisa y House, Tip. 2002.** Testing Extreme Programming. s.l. : Addison-Wesley.

- Curras, Emilia. 2010.** Sistema experto hipermedia para el reconocimiento, indización y recuperación de literatura gris. págs. 117-130. Vol. 4.
- Eíto, Ricardo Brun y Senso, José A. 2004.** Minería textual. págs. 11-27. Vol. 13.
- Estévez, V Chaviano. 2010.** Diseño de un sistema de vigilancia tecnológica en la educación superior. IntEmpress 2010. La Habana Cuba : INFO 2010. Vol. 17.
- Fernández, Yuniel Perdomo y Gámez, Arianna Columbié. 2012.** Análisis y diseño de un Sistema de Vigilancia Tecnológica para la tecnología educativa. La Habana : Universidad de las Ciencias Informáticas.
- Frameworks. 2007.** Frameworks para Java. Desarrollo ágil de software en JAVA. [Citado el: 3 de diciembre de 2013.] <http://daitesrc.wikispaces.com/Frameworks+para+Java>.
- Genbeta. 2013.** Características principales de Bootstrap. [Citado el: 10 de marzo de 2013.] <http://www.genbetadev.com/frameworks/bootstrap>.
- Giráldez, R Reyes y Díaz, M Pérez. 2008.** PROInTec: un software para el tratamiento inteligente de datos sobre patentes. La Habana : Acimed. 17(5).
- Gómez, Arguello y Wilson, Javier. 2011.** Metodología de desarrollo de software un enfoque práctico y global versión. otcolombia. [Citado el: 15 de febrero de 2013.] <http://www.otcolombia.com/documentos/mds360-1.0.11-beta.pdf>.
- Hearst, Marti. 1999.** Untangling text data mining. Maryland : Association for Computational Linguistics.
- IDICT. 2012.** XII Congreso Internacional de Información. [ed.] Memorias 2002 al 2012. La Habana : Palacio de las Convenciones de la Habana. ISBN: 978-959-234-081-7.
- Informativos, Recursos. 2012.** Recursos Informativos. Universidad Autónoma de Yucatán. [Citado el: 3 de noviembre de 2012.] <http://www.tizimin.uady.mx/biblioteca/recursosinfo/fuentesdeinformacion.html>.
- Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2005.** El Proceso Unificado de Desarrollo de Software. [Citado el: 15 de marzo de 2013.] <http://bibliodoc.uci.cu/pdf/8478290362.pdf>.
- Jain, A K, Murty, M N y Flynn, P J. 1999.** Data clustering: a review. s.l. : ACM Computing Surveys. págs. 265-323. Vol. 31.
- Joskowicz, Jose. 2008.** Reglas y Prácticas en eXtreme Programming. España : Doctorado de Ingeniería Telemática de la Universidad de Vigo.
- León, T y González, E. 2004.** Diseño e implementación de un Sistema de Vigilancia Tecnológica en

- una empresa de escasos recursos. La habana : Interprest Info2004. Vol. 10.
- Letelier, Patricio. 2008.** Metodologías Ágiles en el Desarrollo de Software.
- Lidia, Ana y Mercadillo, Careaga. 2010.** Paradigmas de la Programación: JavaScript y Python. México : Instituto Tecnológico de Teléfonos de México S.C.
- Maekotto. 2013.** Bootstrap. [Citado el: 19 de febrero de 2013.] <http://twitter.github.io/bootstrap/>.
- Montes, Manuel. 2010.** Minería de Texto un nuevo reto computacional. México : Laboratorio de Lenguaje Natural. 07738.
- Murgiondo, Juan Etxeberria. 1995.** Análisis de datos y textos. Madrid. pág. 372.
- Navarro, Miguel Ángel Montero. 2009.** Extracción de conocimiento en bases de datos astronómicas. Sevilla: Memoria del periodo de investigación.
- NetBeans. 2011.** NetBeans. Oracle Corporation. [Citado el: 5 de diciembre de 2012.] <http://netbeans.org/features/index.html>.
- Palop, Fernando y Sánchez, M. 2009.** Herramientas de Software especializadas para Vigilancia Tecnológica e Inteligencia Competitiva. España.
- Pascual, Y Riverón. 2009.** Diseño de un sistema de vigilancia tecnológica para el Centro de Gestión de la Información y Tecnologías Libres de la Universidad de las Ciencias Informáticas. La habana : Proyecto de Investigación, Centro de desarrollo de Gestión de la Información y Tecnologías Libres (GEITEL) de la Facultad 10 de la Universidad de las Ciencias Informáticas.
- Pérez, Javier Eguíluz. 2009.** Introducción a XHTML.
- Porter, Alan L y Scott W, Cunningham. 2005.** Tech Mining, Exploiting new technologies for competitive advantage. New Jersey: John Wiley & Sons. 10.1002/0471698466.ch2.
- Reyes, Sady Carina Fuentes y Ruiz, Marina Lobaina. 2007.** Minería de Textos: Aplicación de Web Mining. La Habana: IDICT.
- Rodríguez, M y Ordoñez, M. 2007.** La metodología XP aplicable al desarrollo del software educativo en Cuba. UCI. La Habana.
- Sagrado Olivenza. 2007.** Minería de Texto. Recuperación y organización de la información. [Consultado: 13 de noviembre de 2012].
URL: <http://mineriainformatica.50webs.com/recuperacion-informacion.html>
- Saltalamacchia, Homero R. 2006.** Posible aplicación de la minería de textos a los trabajos de la comisión ministerial de informática. Comunicación. pág. 6.
- SAS. 2010.** Getting started with SAS text miner software. Institute Inc. 58859.

Bibliografías

- Sinnexus. 2007.** Dataminnig. [Citado el: 15 de noviembre de 2012.]
http://www.sinnexus.com/business_intelligence/datamining.aspx.
- Solís, Manuel Calero. 2006.** Una explicación de la programación extrema. Madrid.
- Solís, Pablos, G, A y Rizo, E. 2008.** Informatización para la toma de decisiones. [ed.] Ministerio de Ciencia Tecnología y Medio Ambiente (CITMA). La Habana: Proyecto CITMA. Vol. 13.
- Steinberg, Daniel H y Palmares, Daniel W. 2011.** Extreme Software Engineering.
- SWEBOK. 2009.** Guide to the Software Engineering Body of Knowledge. IEEE Computer Society.
- UNICA. 2009.** Informe Cumplimiento Estrategia Maestra de Informatización. La habana : Universidad Agraria de la Habana.
- Valencia, Carlos Mario Cardona. 2005.** Utilidad Práctica Derivada De Aplicar Minería de Datos en Algunas Empresas de Medellín.
- Vargas, F. W., León, A. y Castellanos, O. 2006.** Valoración, selección y pertinencia de herramientas de software utilizadas en vigilancia tecnológica. Vol. 26. 92102.
- Vázquez, Lara Rey. 2009.** Informe APEI sobre vigilancia tecnológica. Asociación Profesional de Especialistas en Información.
- Vidal, J Larramendi y Pérez, A Perdomo. 2008.** Observatorio de la Universidad de la Universidad de La Habana. La Habana.
- Villegas, Adrian Anaya. 2011.** A propósito de programación extrema XP.
- Web, Libros. 2010.** Introducción a CSS. LibrosWeb.es. [Citado el: 20 de enero de 2013.]
<http://www.librosweb.es/css/>.
- Well, Don. 2006.** Extreme Programming. [Citado el: 18 de enero de 2013.]
<http://www.extremeprogramming.org/>.
- Wells, Don. 2009.** Acceptance Tests. Extreme Programming. [Citado el: 16 de febrero de 2013.]
<http://www.extremeprogramming.org/rules/functionaltests.html>.

ANEXOS

Anexo 1: Guía para la entrevista

Objetivo: Determinar los tipos de fuentes de información a extraer las características en el Sistema de Vigilancia Tecnológica del grupo GIC.

Entrevista dirigida a Lic. Yenieris Moyares Norchales responsable del grupo de GIC del Centro FORTES.

Preguntas:

¿Cuáles son los tipos de fuentes de información que necesita el grupo GIC extraerle las características para el Sistema de Vigilancia Tecnológica?

¿Cuáles son las características de las fuentes de información que necesita el grupo GIC extraer?

Anexo 2: Prototipos de interfaz de las funcionalidades

Extraer características de las fuentes de información

Grupo de Gestión de la Información y el Conocimiento

[Extraer características](#)
[Eliminar/Modificar](#)
[Salir](#)

Cargar fuente de información

Seleccione el documento

/home/osmar/Escritorio/Fuentes de Informacion de

Tesis

Grupo de Gestión de la Información y el Conocimiento
 Centro FORTES
 Telf.8373765

Grupo de Gestión de la Información y el Conocimiento

[Indexar](#)
[Eliminar/Modificar](#)
[Salir](#)

Título:

Implementación de los módulos IEA del Grupo de Calidad de la Facultad 9

Autor(es):

Felix Noel Abelardo Santana

Palabras Clave:

Calidad, Procedimientos de Inicio de Proyectos, Procedimiento de Estrategia de Prueba

Institución:

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Fecha:

junio de 2010

País:

Cuba

Resumen:

Hoy en día, con el creciente auge de la informatización en todas las esferas de la sociedad y el acelerado desarrollo de las Tecnologías de la Informática y las Comunicaciones, la mayoría de las empresas e instituciones necesitan automatizar los procesos de trabajo que realizan. Debido al creciente y constante volumen de documentación que se tramita en el Grupo de Calidad perteneciente a la Facultad 9, en la Universidad de las Ciencias Informáticas (UCI), se hace necesario automatizar gran parte de los procesos allí realizados. Por tanto, el objetivo principal del presente trabajo se centra en la implementación de los módulos Procedimientos de Inicio de Proyectos y Procedimiento de Estrategia de Prueba del portal de Calidad de la Facultad 9, en la cual estas actividades se desarrollan de forma lenta y compleja. El objetivo general de esta investigación es diseñar el portal de Calidad de la Facultad 9 y ejecutar la primera iteración de la fase de construcción. Logrando una correspondencia con los requerimientos de los usuarios del mismo. Para lograr las características anteriores de la forma más

Grupo de Gestión de la Información y el Conocimiento
 Centro FORTES
 Telf.8373765

Modificar y eliminar características de las fuentes de información

Grupo de Gestión de la Información y el Conocimiento

[Indexar](#) [Eliminar/Modificar](#) [Salir](#)

El documento ha sido editado correctamente

-- buscar por --

Artículos

Buscar

Título	Autor(es)	Descargar	Editar/Eliminar
Laboratorio Virtual de Arquitectura de Computadoras.	Indira Abad López Dannays Chacón Laugart Lisset Pérez Castillo		
"Desarrollo de los módulos Administración de Casos y Actividades de Atención para el régimen Extramuros del Sistema Penitenciario Venezolano"	Zunielis C. Quintana Laguna Dennis Molina Alfonso		
Facultad 7Visor Ligero para la visualización y procesamiento básico de imágenes médicas digitales.	Leudis Hernández Sánchez Gerardo Ceruto Marrero		
ActionScript. Curso optativo	Liuber Batista Fernández Hirán Jesús Díaz Martín		
Implementación de los módulos IEA del Grupo de Calidad de la Facultad 9	Felix Noel Abelardo Santana.		

Anterior

Siguiente

Grupo de Gestión de la Información y el Conocimiento
Centro FORTES
Telf. 8373765

Mostrar características de las fuentes de información

Grupo de Gestión de la Información y el Conocimiento

Usuario

Contraseña

Entrar

ActionScript

Artículos

Buscar

Título	Autor(es)	Mostrar	Descargar
Laboratorio Virtual de Arquitectura de Computadoras.	Indira Abad López Dannays Chacón Laugart Lisset Pérez Castillo		
"Desarrollo de los módulos Administración de Casos y Actividades de Atención para el régimen Extramuros del Sistema Penitenciario Venezolano"	Zunielis C. Quintana Laguna Dennis Molina Alfonso		
Facultad 7Visor Ligero para la visualización y procesamiento básico de imágenes médicas digitales.	Leudis Hernández Sánchez Gerardo Ceruto Marrero		
ActionScript. Curso optativo	Liuber Batista Fernández Hirán Jesús Díaz Martín		
Implementación de los módulos IEA del Grupo de Calidad de la Facultad 9	Felix Noel Abelardo Santana.		

Anterior

Siguiente

Grupo de Gestión de la Información y el Conocimiento
Centro FORTES
Telf. 8373765

Buscar fuentes de información

Grupo de Gestión de la Información y el Conocimiento

Usuario

Contraseña

Entrar

Buscar

-- buscar por --

Search

Tipo de Documento: Artículo Tesis Informe

Grupo de Gestión de la Información y el Conocimiento
Centro FORTES
Telf.8373765

Gestionar usuario

Grupo de Gestión de la Información y el Conocimiento

[Agregar Usuario](#)

[Eliminar/Modificar](#)

[Salir](#)

Agregar nuevo usuario

Usuario:

augusto

Contraseña:

.....

Rol:

Especialista

Guardar Usuario

Grupo de Gestión de la Información y el Conocimiento
Centro FORTES
Telf.8373765

Anexo 3: Tareas de desarrollo por iteración

Tarea de desarrollo de la primera iteración

Extraer características de las fuentes de información

Tarea	
Número de la tarea: 1	Número de la HU: HU-1
Nombre de la tarea: Extraer características de las fuentes de información	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 08/03/2013	Fecha fin: 28/03/2013
Programador: Augusto Márquez Ramos	
Descripción: Con esta tarea se extrae las características de las fuentes de información.	

Tareas de desarrollo de la segunda iteración

Modificar características de las fuentes de información

Tarea	
Número de la tarea: 2	Número de la HU: HU-2
Nombre de la tarea: Modificar características de las fuentes de información	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 29/03/2013	Fecha fin: 04/04/2013
Programador: Augusto Márquez Ramos	
Descripción: Con esta tarea se modifican las características existentes en el sistema.	

Mostrar características de las fuentes de información

Tarea	
Número de la tarea: 3	Número de la HU: HU-3
Nombre de la tarea: Mostrar características de las fuentes de información	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 05/04/2013	Fecha fin: 11/04/2013
Programador: Augusto Márquez Ramos	
Descripción: Con esta tarea se muestran las características extraídas de las fuentes de información.	

Eliminar características de las fuentes de información

Tarea	
Número de la tarea: 4	Número de la HU: HU-4
Nombre de la tarea: Eliminar características de las fuentes de información	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 12/04/2013	Fecha fin: 18/04/2013
Programador: Augusto Márquez Ramos	
Descripción: Con esta tarea se eliminan las características extraídas de las fuentes de información.	

Tareas de desarrollo de la tercera iteración

Buscar coocurrencia de palabras

Tarea	
Número de la tarea: 5	Número de la HU: HU-5
Nombre de la tarea: Buscar fuentes de información	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 19/04/2013	Fecha fin: 01/05/2013
Programador: Augusto Márquez Ramos	
Descripción: Con esta tarea se permite buscar información de una temática.	

Autenticar Usuario

Tarea	
Número de la tarea: 6	Número de la HU: HU-6
Nombre de la tarea: Autenticar Usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 02/05/2013	Fecha fin: 09/05/2013
Programador: Augusto Márquez Ramos	
Descripción: Con esta tarea se permite el acceso a los usuarios del sistema.	

Tarea de desarrollo de la cuarta iteración

Gestionar Usuario

Tarea	
Número de la tarea: 7	Número de la HU: HU-7
Nombre de la tarea: Gestionar Usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 10/05/2013	Fecha fin: 16/05/2013
Programador: Augusto Márquez Ramos	
Descripción: Con esta tarea se permite crear, modificar y eliminar a los usuarios del sistema.	