

Universidad de las Ciencias Informáticas

Facultad 7



**Universidad de las Ciencias
Informáticas**

**Título: Estructura del contenido de las plantillas del
Expediente de Proyecto**

**Trabajo de Diploma para optar por el título de
Ingeniero Informático**

Autoras: Yanisbel González Hernández
Lyna Isabel Sánchez Caballero

Tutores: Lic. Maykell Sánchez Romero
Ing. Deborat Pérez Montalván

Ciudad de La Habana, Julio 2008

“ Año 50 de la Revolución ”

DECLARACIÓN DE AUTORÍA

Declaramos ser autoras de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 11 días del mes de Julio del año 2008.

Yanisbel González Hernández

Lyna Isabel Sánchez Caballero

Firma de la Autora

Firma de la Autora

Lic. Maykell Sánchez Romero

Ing. Deborat Pérez Montalván

Firma del Tutor

Firma de la Tutora

DATOS DE CONTACTO

Lic. Maykell Sánchez Romero: maykell@uci.cu.

Ing. Deborat Pérez Montalván: dperezm@uci.cu.

AGRADECIMIENTOS

A la Revolución, a nuestro siempre Comandante Fidel y a la UCI por permitirnos formar parte del Proyecto Futuro y por formarnos como profesionales revolucionarios.

A nuestro tutor Maykell, por prestarnos todo su apoyo aún estando lejos, por fomentar nuestro espíritu de superación, por ser profesor, compañero y amigo.

A quienes nos brindaron su ayuda en la realización de esta tesis demostrando su incondicionalidad, Lourdes, Alfredo, Darién, Chony, aportándonos lo mejor de si en cada momento.

A nuestros familiares y seres queridos que han estado pendientes del logro de este sueño compartido.

A todos los profesores que durante 5 años nos han formado guiándonos en el camino de la búsqueda del conocimiento.

A nuestros compañeros de la Universidad que han sido espectadores activos de esta importante etapa de nuestras vidas.

DEDICATORIA

A mi abuela Aba, por su amor incondicional, por darme lo mejor de sí cada día de su vida y porque aún no me acostumbro a que no esté en días como estos.

A mis padres Julia y Eliseo, por su esfuerzo y dedicación, porque sin ellos no lo hubiese podido lograr, por confiar en mí y darme su apoyo en cada momento. Simplemente por amarme.

A mis segundos padres Tata y Toto, por ser un eslabón fundamental en mi vida, por su amor y apoyo, porque aprendí con ellos que lo esencial es invisible a los ojos. Porque los amo con todo mi corazón.

A Tony, el amor de mi vida, por ser mi novio, amigo, confidente y mi futuro esposo. Por estar a mi lado, por darme fuerzas cuando me creo vencida, por su amor y hasta por su paciencia. Porque lo amo.

A mis hermanos Yamy y Yoe, por ser los mejores hermanos del mundo, cada uno a su manera, por ayudarme a ser mejor cada día y por estar siempre a mi lado aún cuando están lejos.

A mis tíos Julito y Magalis quienes me han dado su apoyo sin condiciones, a quienes quiero y admiro por ser dos personas excepcionales.

A mi familia que es maravillosa, que aunque no pueda mencionar a todos los que se merecen estar en estas líneas, quiero que sepan que los amo.

A los amigos y amigas que se han mantenido a mi lado en tiempos buenos y malos.

Lyna

A mis padres por ser lo mas grande que la vida me ha dado, por su apoyo en todo momento, su cariño, su amor, por enseñarme a enfrentar cualquier barrera, por exhortarme siempre a ser mejor, por ser los mejores padres del mundo.

A mi hermana por ser un ejemplo difícil de igualar, por ser mi mejor amiga, en quien siempre puedo confiar, y sé que estará junto a mi en las buenas y malas.

A mi pequeña hermanita Debi para que siga creciendo inspirada en estudiar y pueda llegar a disfrutar un momento tan especial.

A mi gran amigo Silvano por su apoyo incondicional, por su cariño, por su grandeza, por ayudarme con su experiencia a superar los obstáculos.

A mi amiga de siempre Keyi por hacer, aún estando lejos, que este día sea tal como lo imaginé.

A mis abuelos, tíos y primos que han sido participes de los logros que poco a poco he ido alcanzando en mi vida.

Yany

RESUMEN

La industria del software es una de las más competitivas a nivel internacional. Como parte de los mecanismos que permiten mejorar la calidad de los procesos de desarrollo del software que se produce en la UCI, se ha desarrollado un Expediente de Proyecto (EP) con el objetivo de documentar el producto y su proceso de producción. Su confección resulta muy trabajosa debido a que incluye numerosas plantillas y documentos. Además, recoge gran cantidad de información, que en ocasiones es redundante, lo que dificulta su elaboración. Por esto, se hace necesario informatizar la gestión del EP, para ello debe lograrse una estructuración adecuada de la información que aparece en cada una de las plantillas que lo conforman.

El presente trabajo de diploma tiene como objetivo realizar el análisis y diseño de una estructura para el contenido de las plantillas del Expediente de Proyecto utilizado en la Universidad de las Ciencias Informáticas.

La estructura se diseñó utilizando XML Schema como lenguaje de esquema. Esta permite eliminar la repetición innecesaria en el llenado de datos, validar automáticamente la estructura del documento, realizar un análisis sintáctico de los datos y facilita la gestión del EP a partir de una aplicación. Con la utilización de la estructura propuesta se pueden eliminar irregularidades en la elaboración del EP y posibilitar su futura informatización. Además permitirá la integración con otros sistemas que puedan ser implementados en la Universidad y a largo plazo garantizar la calidad requerida en los productos.

PALABRAS CLAVES Expediente de Proyecto, Estructura, Esquema, Documento.

TABLA DE CONTENIDOS

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
1.1 Gestión de la información, del conocimiento y de la calidad	5
1.2 Gestión Documental	6
1.2.1 Antecedentes de la Gestión Documental	7
1.2.2 Gestión documental de proyectos de producción de software.....	8
1.3 Expediente de Proyecto.....	10
1.3.1 Expediente de DeSoft S.A.....	10
1.3.2 Expediente de Proyecto de la UCI	11
1.4 Análisis de soluciones existentes	14
1.4.1 Herramientas para la Gestión Documental de Proyectos	16
1.4.2 Productos comerciales más representativos.	19
1.5 Análisis crítico	21
CAPITULO 2: TENDENCIAS Y TECNOLOGÍAS.....	22
2.1 Técnicas y procesos para el manejo de la información.....	22
2.2 Lenguajes.....	27
2.3 Gestores de Base Datos.....	34
2.4 El XML y las Bases de Datos.....	36
2.5 Propuesta de tecnología a utilizar.....	38
CAPITULO 3: ANÁLISIS Y DISEÑO DE LA ESTRUCTURA.....	42
3.1 Selección de las plantillas.....	42
3.1.1 Lineamientos de calidad	43
3.1.2 Clasificación de la Documentación.....	49
3.2 Expediente de Proyecto Elemental (EPE).....	51
3.3 Análisis de la cronología de las plantillas.....	55
3.4 Clasificación de las plantillas según su composición y tiempo de elaboración	59
3.5 Naturaleza de la Información	61
3.6 Diseño de la estructura.....	62
3.7 Beneficios de la estructura.	67

CONCLUSIONES.....	68
RECOMENDACIONES	69
REFERENCIAS.....	70
BIBLIOGRAFÍA	74
ANEXOS.....	78
GLOSARIO DE TÉRMINOS.....	79

INTRODUCCIÓN

La informática es una de las ciencias que más rápido avanza. El crecimiento e impacto global de esta es indiscutible. Las tendencias modernas en gerencia, re-ingeniería y calidad total, están fuertemente centradas en la aplicación de la informática. Las organizaciones en general, que giran su actividad en torno a la transformación de información y la producción de conocimiento, deben informatizarse de manera rápida y eficiente, para no correr el riesgo de perder competitividad.

Se habla constantemente sobre la sociedad de la información, esta nueva sociedad, con organizaciones basadas en el aprendizaje, cuyo capital máspreciado es el ser humano, se sustenta en un desarrollo tecnológico sin precedentes. La información es un elemento fundamental para el desarrollo. Con el de cursar de los años, la gestión de la información ocupa un espacio mayor en la economía de los países a escala mundial. En el mundo actual, la gestión del conocimiento por parte de las organizaciones, adquiere nuevas características, determinadas por la gestión de la información y de la calidad.

Uno de los principales problemas de la información, es su exceso, por lo que se hace necesario invertir mucho tiempo en ella. Es frecuente encontrar un número significativo de información redundantes y de baja calidad mezcladas con otras importantes y sólidas, difíciles de hallar entre la información ruidosa que la "envuelve". Cuando una organización trabaja con gran cantidad de información, debe garantizar una adecuada gestión de la misma que facilite la excelencia en su desempeño global, elevando la calidad de todos los resultados. Para garantizar un manejo adecuado de la información surge el término de Gestión Documental (GD), que consiste en el uso de tecnología y procedimientos que permiten la gestión y el acceso unificado a la información generada en la organización. (1)

La GD proporciona una solución integral para la manipulación y protección de aquellos documentos que se desean preservar en los proyectos como soporte de su negocio. La aplicación de estos procesos no solo hará posible disciplinar a los involucrados, permitirá además, almacenar y disponer de los datos históricos necesarios para lograr un trabajo más predecible y eficiente. Esta disciplina es aplicable en muchas ramas de la ciencia y la técnica, pues siempre hay un producto o acción con un propósito definido, donde la documentación es un testimonio de lo que se realiza durante el proceso. Una de las ramas donde se genera gran cantidad de información es la Industria del Software por lo que se hace indispensable utilizar técnicas de Gestión Documental que garanticen el manejo adecuado de todos los documentos.

La industria de software se ha convertido en los últimos años en un elemento clave para el ascenso de la economía en varios países, se desarrolla a un ritmo acelerado, y los ingresos por exportación de software a nivel mundial han alcanzado cifras importantes. En Cuba, la informática se vislumbra hoy, como una vía importante de desarrollo social y económico, en la cual la industria de software deberá jugar un papel primordial. La situación actual y las perspectivas del país en esta industria se sustentan en el trabajo que se viene realizando en materia de capacitación y en la inserción en el mercado nacional como internacional.

La industria nacional de software impone la necesidad de formar profesionales con habilidades para el desarrollo de la informática. Su crecimiento vertiginoso requiere garantizar procesos con alta calidad y productividad. Temas como calidad de software, de productos y de procesos deben ser introducidos con urgencia en las empresas estableciendo estándares y normas nacionales, basadas en las mejores prácticas internacionales.

En año 2002, con el desarrollo del proceso de informatización de la sociedad cubana, surge la Universidad de las Ciencias Informáticas (UCI) con el objetivo de ayudar e impulsar el desarrollo de la producción de software en el país. Esta Universidad, se ha convertido en centro de referencia para la Industria Cubana del Software. A pesar de su corto tiempo de vida ha desarrollado un gran número de proyectos con excelentes resultados. Esto ha impuesto un reto en cuanto a la organización de la producción con la máxima de que la cantidad no puede afectar la calidad.

La calidad del software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. La calidad es sinónimo de eficiencia, flexibilidad, corrección, portabilidad, usabilidad, seguridad e integridad. La obtención de un software con calidad implica la utilización de metodologías o procedimientos estándares para el análisis, diseño, programación y prueba del software que permitan uniformar la filosofía de trabajo, en aras de lograr una mayor confiabilidad y facilidad de prueba (2).

Gestionar documentos para lograr una mayor eficiencia en los proyectos de producción de software, es algo necesario para alcanzar la calidad de los productos. Es indispensable documentar el proceso de desarrollo y controlar cada fase, de esto depende el éxito de cada proyecto, y la satisfacción del equipo y del cliente al conocer en cualquier momento el grado de desarrollo, y gestionar los cambios de una manera ágil y dinámica. En un principio la documentación que se recogía en muchos de los proyectos desarrollados en la UCI, se realizaba de forma independiente, incluyéndose lo que se consideraba

realmente importante. Esto no era suficiente para garantizar la calidad del proceso de desarrollo de software, así como el mantenimiento del mismo y el desarrollo de nuevas versiones.

Como parte de los mecanismos que permiten mejorar la calidad de los procesos de desarrollo del software que se produce en la UCI, se ha desarrollado un Expediente de Proyecto (EP) con el objetivo de documentar suficientemente el producto y su proceso de producción. Este se implantó en cada uno de los proyectos vigentes en la Universidad. En este expediente los proyectos pueden encontrar todas las plantillas para la documentación de la solución en la fase del proyecto que se encuentre, garantizando una estandarización en la documentación y la creación de una cultura de calidad.

El proceso de confección del EP resulta muy trabajoso debido a la cantidad de plantillas y documentación que recoge. Además presenta un gran número de información redundante, lo que dificulta su elaboración, provocado la inconformidad de los desarrolladores, retrasos en la entrega de la documentación y baja calidad de la misma.

Teniendo en cuenta la situación anterior, el **problema** radica en: ¿Cómo facilitar el proceso de llenado de las plantillas que conforman el Expediente de Proyecto?

El **Objeto de estudio** se centra en el análisis del proceso de gestión de la documentación de los proyectos productivos en la Universidad de las Ciencias Informáticas.

Para esto se especifica como **Campo de acción** el Expediente de Proyecto en la Universidad de las Ciencias Informáticas.

Para dar solución al problema antes mencionado se define como **Objetivo general** diseñar una estructura para el contenido de las plantillas que conforman el Expediente de Proyecto utilizado en la Universidad de las Ciencias Informáticas que permita la futura informatización del mismo.

Para el desarrollo del trabajo se han formulado las siguientes **preguntas de investigación**:

¿Cuáles son las características del Expediente Proyecto en la UCI?

¿Qué tecnologías y metodologías se enmarcan dentro de la gestión de documentos?

¿Cómo desarrollar de forma óptima la gestión del Expediente de Proyecto?

Para dar cumplimiento al objetivo general y considerando las preguntas de la investigación planteadas se proponen las **tareas** siguientes:

- Evaluar diferentes métodos para la gestión de documentos en los proyectos de software.
- Realizar estudio sobre el Expediente de Proyecto en la UCI.
- Definir las tecnologías y metodologías de desarrollo a utilizar.
- Diseñar una estructura para la gestión del contenido del Expediente de Proyecto.

El presente trabajo se encuentra estructurado por tres Capítulos que tratan los siguientes contenidos: En el Capítulo 1, titulado “Fundamentación teórica”, se plantean los principales conceptos y términos abordados en la investigación, se hace un estudio de las soluciones existentes en el mercado y un análisis crítico de las mismas, definiendo la manera de dar solución al problema planteado. En el Capítulo 2, “Tendencias y Tecnologías”, se estudian las tendencias actuales que permiten el tratamiento de la información. Se analizan las tecnologías y herramientas a utilizar, realizando la selección de la más adecuada. Por último en el Capítulo 3, “Análisis y Diseño de la estructura”, se explican todos los pasos seguidos para el diseño de la estructura del Expediente de Proyecto y se exponen un conjunto de resultados obtenidos con el análisis del mismo.

Como resultado de la investigación se desea obtener un análisis de las técnicas de Gestión Documental, un estudio detallado de las principales plantillas que conforman el Expediente de Proyecto. El diseño de una estructura para el contenido del mismo, que permita su informatización, dándole la posibilidad de integración con otros sistemas que puedan ser implementados en la Universidad.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

A partir del análisis de la información, el conocimiento y la calidad, en este capítulo se plantean los principales conceptos y términos abordados en la investigación, enfatizando en la Gestión Documental. Además se mencionan algunos productos que facilitan la misma y el manejo de la información. Se analiza la conformación del Expediente de Proyecto y sus principales características.

1.1 Gestión de la información, del conocimiento y de la calidad

En la era de la información se ha alcanzado un desarrollo imprevisible; cada día son mayores las diferencias sociales, políticas y económicas. Surge la llamada sociedad de la información, donde es visible el paso de las sociedades industriales a las post-industriales y del conocimiento. Muchos especialistas han llamado la atención sobre la importancia del conocimiento como un recurso gestionable y fuente de poder. Se han incrementado las teorías sobre su gestión y la relación con la organización industrial, la gestión tecnológica y estratégica.

La gestión del conocimiento es un proceso de identificación, captura, organización y diseminación de los datos claves y la información necesaria para ayudar a la organización a responder a las necesidades de los clientes (3) Con ello se busca la perpetuación y la materialización del potencial de las organizaciones, algunas de ellas son incapaces de comprender que la información es un recurso, valor o activo intangible igual que cualquier otro.

La gestión de la información se vincula con la aplicación de estrategias y el establecimiento de políticas que permiten el manejo de la información. Está fuertemente relacionada con el desarrollo de una cultura social dirigida al uso equitativo y eficiente de la información para alcanzar los objetivos y metas de la organización en materia de desempeño y de calidad.

En la creación de los nuevos sistemas de gestión de la información es imprescindible considerar las fuentes factográficas (datos), documentales y no documentales, los sistemas informáticos, la cultura de información, los modelos de comunicación, entre otros elementos. Según los requerimientos de los procesos internos de trabajo y los flujos de información propios, todos ellos deben propiciar la gestión del conocimiento organizacional y la implementación de sistemas de gestión de la calidad para la evaluación de los resultados y los proyectos de la institución. De una correcta gestión de la información y del conocimiento depende, en gran medida, la implementación de la gerencia de la calidad.

La introducción de las ideas y principios de la gerencia de calidad provocó una revolución en la filosofía de la administración. Ella incluye la planificación estratégica, la asignación de recursos, programación de las operaciones y las evaluaciones relativas a la calidad (4) Su práctica está dirigida a diseñar y generar servicios y productos siempre satisfactorios para el cliente.

Obtener la información necesaria, con la calidad requerida, es una premisa indispensable para la supervivencia de las empresas, si se considera que las organizaciones acortan cada vez más sus ciclos estratégicos y que la toma de decisiones, así como el cambio, es continuo. Es obvio que una eficiente gestión de la información, como parte de la aplicación de una política de gestión de la calidad, garantiza que la organización obtenga mayores ganancias y una mejor competitividad en el mercado.

La información es un agente importante en la modificación de las conductas existentes en la organización, su correcta gestión es una herramienta fundamental para la formación del personal, la evaluación de los productos, la determinación de los errores y el control de los procesos. La tecnología existente permite entregar herramientas que apoyan la gestión del conocimiento en las empresas. Estas facilitan la recolección, la transferencia, la seguridad y la administración sistemática de la información, junto con los sistemas diseñados para lograr el mejor uso de ese conocimiento. Existen herramientas y técnicas diseñadas para preservar la disponibilidad de la información llevada a cabo por los individuos dominantes y para facilitar la toma de decisión y la reducción de riesgo. Uno de los términos más utilizados dentro de la gestión de la información es la Gestión Documental que comprende el manejo de dicha información a través de documentos.

1.2 Gestión Documental

La paulatina incorporación de las tecnologías de la información por parte de todo tipo de empresas e instituciones en los distintos procesos de la cadena de producción y de gestión, ha hecho que se modifiquen sustancialmente los modos de trabajo dentro de las mismas, utilizándose cada vez más la Gestión Documental. Esta se basa en el conjunto de normas, técnicas y prácticas usadas para administrar el flujo de documentos de todo tipo en una organización, permitir la recuperación de información desde ellos, determinar el tiempo que los documentos deben guardarse, eliminar los que ya no sirven y asegurar la conservación indefinida de los documentos más valiosos, aplicando principios de racionalización y economía. (5)

Este hecho tiene diversas e importantes repercusiones para los actuales servicios de información y documentación, así como para los profesionales de los mismos, encargados de gestionar el conocimiento que se está generando en dichos sistemas. La información, tanto la interna como la externa, es un elemento clave y estratégico dentro de las organizaciones y de competitividad para las mismas.

Las diversas formas por las que se producen los documentos electrónicos, la variedad de tipologías de los mismos y la adecuación de las actuales herramientas informáticas para la gestión integral de la documentación circulante, serán factores esenciales que se han de analizar detenidamente por parte de los servicios de información y documentación de toda institución que produzca y maneje documentos a la hora del diseño y conceptualización de un sistema de gestión electrónica de la información. La proliferación de los documentos electrónicos ha permitido que esta conceptualización cobre una especial importancia en estos días, aunque la utilización de las mismas tiene precedentes sustanciales.

1.2.1 Antecedentes de la Gestión Documental

El factor inicial relacionado con la manera de tratar la documentación, es un problema que tiene como origen lograr un control claro y actualizado de los documentos que genera cada uno de los proyectos, procedimientos y procesos en una organización. La Gestión Documental clásica ha consistido en la impresión en papel de la información que se desea comunicar, y ha sido válida hasta principios de siglo dado que la cantidad de documentos ha sido razonable. En el mundo del papel, la función de la gestión documental englobaba dos grandes campos: la gestión de los documentos activos, y los tradicionales archivos encargados de la organización/ preservación de los documentos pasivos.

Los tradicionales archivos en papel se han mantenido en paralelo a la introducción de los sistemas informáticos hasta la actualidad, siendo realmente escasas las empresas que han conseguido una verdadera oficina sin papeles. Sin embargo el documento electrónico es una realidad que crece vertiginosamente desde el momento en que se produce la aparición de los ordenadores y la utilización masiva de las herramientas ofimáticas. La tangibilidad que proporcionaba el papel como soporte documental ha cambiado drásticamente con los documentos electrónicos, sin que los responsables de la documentación hayan sido, en muchos casos, capaces de adaptarse al rápido cambio del entorno.

La Gestión Documental actual nace porque las organizaciones necesitan soluciones que permitan gestionar la información disponible para hacerla accesible, dentro de su ciclo de vida y bajo estrictos controles de seguridad (6). Es un elemento clave en la actividad de una empresa y está dejando de ser

un simple método de archivo masivo para convertirse en una herramienta de análisis de información y gestión del conocimiento, con una creciente demanda no sólo en grandes corporaciones sino también en las pequeñas y medianas empresas. La GD contempla distribuir, compartir, proteger, cambiar, buscar y versionar la información base para la actividad de la (7).

En la industria del software la GD aún no ha sido bien comprendida, ni ampliamente aplicada. El principal objetivo de las organizaciones es resolver los problemas derivados del exceso de información a tratar, bien por la gran cantidad de papel que debe ser manejado, así como por el almacenamiento de otros objetos de información de uso corporativo como pueden ser ficheros ofimáticos, reportes de las propias aplicaciones de gestión empresarial, imágenes, sonidos, video, e-mail, faxes, entre otros.

La producción de software es una de las ramas donde se manipula gran cantidad de información de todo tipo, por lo que la Gestión Documental tiene un papel fundamental.

1.2.2 Gestión documental de proyectos de producción de software

El tratamiento de la información y el conocimiento, en particular, en el ámbito de la informática, es un tema complejo en la actualidad. La Industria del Software, que abarca toda la creación y desarrollo de los sistemas informáticos, es una de las más competitivas a nivel internacional, debido a que su mercado se encuentra cubierto por un sin número de grandes, medianas y pequeñas empresas con productos adecuadamente posicionados y clientes fijos. Por esto, es una garantía presentarse al mundo de la producción de software con efectivos métodos y formas de trabajo que demuestren eficacia y alta productividad.

La humanidad se encuentra sometida a un cambio constante, debido a la rápida evolución tecnológica de estos tiempos. Nadie escapa de este fenómeno aunque no a todos les llegue por igual. La producción de software es una práctica en la que diariamente se debe estar actualizado, pues avanza al paso de la revolución tecnológica. Actualmente se construyen numerosos sistemas, apuntando a un mayor tamaño y complejidad, para satisfacer a usuarios cada vez más exigentes.

En la actualidad es muy común el término de Ingeniería de Software, disciplina encargada de planificar los procesos de desarrollo y mantenimiento de un software. La ingeniería de software es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadoras, y la

documentación asociada requerida para desarrollar, operar y mantener dichos programas. Se conoce también como desarrollo de software o producción de software (8).

Para realizar de forma exitosa un proyecto de software es necesaria la correcta ejecución de cuatro tipos de funciones:

1. **Gestión del Proyecto:** incluye fundamentalmente la Estimación, Planificación y Seguimiento del proyecto, Organización, Dirección y Gestión de Recursos Humanos.
2. **Desarrollo Técnico:** actividades de Ingeniería del Software a lo largo de todo el ciclo de vida del producto: Análisis, Diseño, Codificación.
3. **Sistema de Calidad:** incluye las actividades de:
 - Validación: construir el producto correcto.
 - Verificación: comprobar si se está construyendo el producto correctamente.
 - Pruebas: verificar si funciona el código.
 - Actividades de Garantía de Calidad: asegurar que el producto se construye posea determinados niveles de calidad.
4. **Sistema de Gestión de Configuración:** incluye principalmente la identificación, organización y control de las modificaciones que sufre el software.

La práctica muestra que la mayor cantidad de información que se maneja en un proyecto debe quedar plasmada en documentos para su inmediata y posterior consulta. La masiva implantación de herramientas de GD da lugar a que la mayoría de los documentos se produzcan de manera electrónica. Todo esto provoca que la Gestión Documental de Proyectos pretenda lograr un adecuado tratamiento de los documentos para optimizar su aprovechamiento, permitiendo a los desarrolladores de un producto de software guardar toda la información que se genera en este proceso y poder recuperarla de forma precisa e inmediata. (9) Esto reduce tiempo de desarrollo y evita la duplicación de datos, perfeccionando la comunicación entre usuarios, clientes y entidad.

La gestión de la documentación en los proyectos de software se puede ver como la solución para el control de los documentos con una perspectiva superior, debido a que está presente en toda la etapa de su creación; desde su origen hasta que finalmente quede elaborado y se almacene, estableciendo una organización y actualización constante de los mismos. Uno de los mecanismos elaborados para la organización de la información de la documentación que se genera en los proyectos de producción de software es el Expediente de Proyecto del cual se hace un estudio más detallado.

1.3 Expediente de Proyecto

El Expediente de Proyecto (EP) se elabora para documentar todo el proceso de desarrollo del software. En él se gestiona la información necesaria para planificar y controlar adecuadamente el proyecto. Se incluye como una de las diferentes formas que permiten una correcta gestión documental en la producción de software.

Un EP se abre cuando el cliente solicita un servicio para realizar un producto. Normalmente este se compone de un único producto, pero también puede ocurrir en otros casos que sea preciso por la envergadura del mismo realizar varios expedientes. En este caso se crearían otros expedientes para los nuevos productos, de manera tal que queden proporcionados.

1.3.1 Expediente de DeSoft S.A

La Empresa DeSoft S.A. surgió a partir de la integración y asociación de capital de entidades que funcionaban de manera aislada, subordinadas al Grupo de Tecnologías de la Información (GTI) que pertenecía al Ministerio de la Informática y las Comunicaciones. Su objetivo radica en la comercialización nacional e internacional de productos y servicios, para lo cual cuenta con 15 divisiones territoriales, repartidas en todas las provincias, y una casa matriz en Ciudad de La Habana. DeSoft S.A. facilita una amplia gama de productos y servicios como la implantación de sistemas informáticos para la gestión empresarial, desarrollo de aplicaciones para Internet y comercio electrónico, consultorías y asesorías especializadas en tecnologías de la información, publicaciones de soporte magnético y multimedia, cursos y adiestramientos, así como planificación, desarrollo y evaluación de proyectos integrales especializados (10).

A continuación se muestra la estructura que presenta el expediente aplicado en el desarrollo de productos de software en esta institución, el cual se basa en RUP (*Rational Unified Process- Proceso Unificado de desarrollo del Software*) ajustándose a las necesidades de la entidad:

- Ingeniería de software
 - Artefactos resultantes del proceso de Ingeniería de software organizados por flujos de trabajo (negocio, requisitos, análisis, diseño, etc.).

- Planificación
 - Cronograma
 - Riesgos
 - Información del proyecto
 - Recursos

- Gestión de configuración
 - Control de versiones
 - Herramientas para el control de versiones
 - Bibliotecas de trabajo
 - Administración de ECS (elementos de configuración del software)

- Aseguramiento de la calidad
 - Administración de la calidad del software
 - Plantillas
 - Evaluación de la calidad del software

1.3.2 Expediente de Proyecto de la UCI

Con el objetivo de convertir a la Universidad de las Ciencias Informáticas en una Industria de Software de alto prestigio a nivel nacional e internacional se trazó una estrategia para garantizar la calidad de los productos a comercializar. La documentación de todo el proceso es una tarea fundamental para el logro de la calidad del software, por lo que se hizo inminente la creación del Expediente de Proyecto. Este se confeccionó teniendo en cuenta varios aspectos. Como son; identificar y revisar la documentación requerida por modelos, estándares y normas de calidad ya existente a nivel internacional, establecer las necesidades propias de los proyectos de UCI y adaptar las plantillas a este entorno, adicionando algunas que por las características propias de la universidad no existen

De esta manera se adaptaron y complementaron algunas plantillas propuestas por: **IEEE** (*The Institute of Electrical and Electronics Engineers- Instituto de Ingenieros Eléctricos y Electrónicos*), **NASA** (*National Aeronautics and Space Administration- Administración Nacional de Aeronáutica y del Espacio*), **ISO** (*Organización Internacional para la Estandarización*), **DoD** (*United States Department of Defense- Departamento de Defensa de Estados Unidos*) y **RUP** (*Rational Unified Process- Proceso Unificado de Desarrollo del Software*); otras fueron confeccionadas basadas en la experiencia personal.

Una característica presente en todos los estándares y normas revisados es su no adherencia a una metodología de software específica, por lo que suelen quedar poco delimitadas las secciones que abordan las áreas de ingeniería. La metodología RUP junto al Lenguaje Unificado de Modelado UML, constituye la metodología estándar más completa y utilizada para el análisis, implementación y documentación de sistemas orientados a objetos, además es fácilmente adaptable al contexto y necesidades de cada organización. Por ello se ha decidido seleccionar a RUP como la metodología base de desarrollo de las soluciones UCI. En estos momentos se están realizando estudios que permitan adaptar los artefactos de esta metodología en función del desarrollo de proyectos de Software Libre que utilizan metodologías ágiles.

La facultad 10 de la Universidad encargada del desarrollo de aplicaciones de Software libre, se encuentra realizando en estos momentos una propuesta de EP para los que se basan en metodologías ágiles, la cual deberá ser aprobada posteriormente por el Dirección de Calidad de Software (DCS).

Dentro de los modelos de calidad para la producción de software, la Universidad seleccionó para la realización del EP a CMMI (*Capability Maturity Model Integration- Modelo de Madurez y Capacidad Integrado*) como el modelo rector, haciendo el expediente compatible con dicho modelo, para esto es preciso adaptar el modelo CMMI a la metodología RUP.

El modelo CMMI establece niveles de madurez para clasificar a las organizaciones, en función de qué áreas de procesos consiguen sus objetivos y se gestionan con principios de ingeniería. Cuenta con dos tipos de representaciones: continua y escalonada. Son equivalentes, y cada organización puede seleccionar la que se adapte a sus características y prioridades de mejoras. La visión continua de una organización mostrará la representación de nivel de capacidad de cada una de las áreas de proceso del modelo y propone 6 niveles. La visión escalonada definirá a la organización dándole en su conjunto un nivel de madurez del 1 al 5, según las áreas de procesos que se tomen en cuenta. (11)

Atendiendo al nivel de desarrollo que alcanza en estos momentos la UCI, solamente se tuvo en cuenta las Áreas de Procesos (AP) pertenecientes al nivel 2 de madurez de CMMI y por su importancia para los proyectos se incluyeron algunas AP pertenecientes al nivel 3.

CMMI divide sus AP en 3 grupos:

- Ingeniería
- Soporte
- Gestión de proyecto

Estas mismas clasificaciones fueron las utilizadas para agrupar las plantillas en el EP agregando una cuarta categoría para la documentación legal que dada su naturaleza y lo delicado de la misma no se consideró conveniente mezclar con documentación técnica.

El presente trabajo de diploma se centra en el proceso: del llenado del Expediente de Proyecto. Este permite que los diferentes Grupos de Desarrollo de la Universidad de las Ciencias Informáticas puedan documentar adecuadamente el proceso de desarrollo del software para certificar la calidad de sus productos. Actualmente se utiliza la segunda versión del Expediente de Proyecto definida por la DCS que cuenta con 40 plantillas obtenidas de las normas y estándares más utilizados para el desarrollo de software.

Se elaboró una estrategia para su puesta en marcha que ya está arrojando resultados y se ha logrado estandarizar la documentación generada por los proyectos de la UCI. A pesar de los buenos resultados obtenidos a partir de la implantación del EP, aún existen algunas deficiencias que se deben erradicar. Estas se basan principalmente en la gran cantidad de información redundante que se recoge en él y la baja calidad de la documentación generada. En vísperas de la importancia del EP se hace necesario mejorar la realización del mismo.

Teniendo en cuenta las características del EP, el objetivo será analizar y diseñar una estructura, para validar cada una de las plantillas desde sus definiciones estructurales y semánticas permitiendo una adecuada gestión del mismo que garantice optimizar la información de las plantillas que conforman el expediente, facilitar su llenado y la certificación de su correcta realización. Para ello se hace un estudio sobre los diferentes productos y herramientas que existen en el mundo, que permiten el manejo de la información dentro de la producción de software.

1.4 Análisis de soluciones existentes

Existen algunas soluciones informáticas que sin ser automatizadas gestionan y documentan la información relacionada con la producción del software, entre ellas se encuentran la metodología RUP la cual genera artefactos que guían y documentan el proceso de desarrollo del software. También existen estándares como son CMMI, ISO, IEEE, NASA, DoD, que proponen plantillas con la misma finalidad.

Actualmente en Cuba no se ha desarrollado ninguna aplicación para la documentación del proceso de desarrollo del software. La Universidad de las Ciencias Informáticas no utiliza ningún sistema que automatice los procesos de calidad y ni que gestione el EP. Hasta el momento solo se utilizan algunas herramientas de GD para manejar la información como es el Subversión.

En la Facultad 10 existe un proyecto que trabaja la gestión documental de la producción de software en la UCI. Este consiste en el análisis y diseño de un Habilitador Metodológico para la integración y el desarrollo de aplicaciones en Software Libre. El proyecto pretende realizar un estudio de las metodologías ágiles existentes, modificar los artefactos de la metodología RUP para el flujo de trabajo de diseño, en función de las necesidades del desarrollo de los proyectos de Software Libre .y proponer documentación para la integración de las metodologías. Como resultado se puede obtener una guía sobre los principales elementos o componentes de los modelos de procesos genéricos y de cada una de las metodologías de desarrollo del software. Permitiendo lograr una mejor adecuación de la documentación para cada proyecto y gestionar la información generada por las diferentes herramientas que se utilizan en dicha producción.

La Gestión Documental de Proyectos adquiere cada vez mayor relevancia, debido a la necesidad de mejorar los procesos de forma continua. Para ello es necesario contar con herramientas que permitan gestionar toda la información que se maneja en los proyectos de una manera eficiente y segura. La introducción de las TIC ha permitido el desarrollo de herramientas cada vez más sofisticadas para manejar esos volúmenes de información. Éstas brindan varias ventajas tales como: ahorro en espacio físico y equipamiento, reducción de costos salariales, administrativos y de los derivados del uso del papel, acceso centralizado y consulta distribuida, rápida localización de los documentos por múltiples vías de acceso y rápido intercambio de información. (12)

En el mundo existen aplicaciones que manejan de forma general la gestión de documentos. Estas permiten relacionarlos entre si y darles una semántica común. Actualmente existen en el mercado una gran variedad de aplicaciones que satisfacen, en diferentes grados, las necesidades de gestión documental en el ámbito corporativo. Hasta hace pocos años, la gestión documental en los servicios de información, se basaba en aplicaciones verticales para la gestión de documentos estructurados en bases de datos relacionales y/o documentales, sin ninguna o poca interacción con otras infraestructuras tecnológicas corporativas.

La gestión electrónica documental, abarca hoy en día una serie de productos que cumplen con las funcionalidades de generación, captura, gestión, recuperación y distribución de cualquier tipo de documentación con independencia de su estructura y basándose en su contenido como componente inteligente dentro de los procesos productivos de cualquier organización. Las principales características que ofrecen estos productos son (13)

- Captura, indización y entrada automática

Consiste en la digitalización tanto de documentos papel a través del escáner, como de cualquier tipo de documento electrónico creado por aplicaciones de oficina (Word, Excel, etc.), documentos técnicos; imágenes, vídeos, audio, documentos html, entre otros, así como almacenamiento de ficheros spool mediante módulos COLD (*Computer Output to Laser Disk*). La entrada de nuevos documentos dentro del sistema, puede realizarse de manera sencilla desde una multitud de aplicaciones cliente.

- Gestión Integrada del documento:

Registro y control de las distintas versiones de un mismo documento a lo largo de todos los procesos productivos en los que está implicado asegurando una calidad en la información, permitiendo el manejo de múltiples versiones de forma íntegra y controlada (*chek-in chek-out*). Este control hace que los distintos usuarios implicados en una misma tarea puedan manejar, de forma transparente, la última versión de un documento asignando controles de seguridad apropiados de acceso, consulta y modificación. El sistema hace un seguimiento de todas las versiones de un mismo documento manteniendo un registro histórico de actividades desde su creación. El usuario apropiado accede a la versión correcta.

- Automatización del ciclo de vida documental

Desde su captura/creación, revisión, etc., hasta su archivado final, mediante la gestión de los procesos y tareas de negocio en los cuales está implicado con tecnología de *workflow* y *groupware*. Esta característica permite la agrupación lógica de documentos afines por diferentes características en un único documento compuesto electrónico (expedientes o *dossiers* de información) para diferentes fines: distribución por correo electrónico, publicación en portales corporativos, servicios de alerta, DSI, etc., conservando de cada elemento sus características de integridad y ciclo de vida dentro del sistema. Automatización de las diferentes etapas de archivo según parámetros previamente definidos en soportes *off-line* manteniendo una referencia para su recuperación: ubicación física del documento, tiempos de actualización para diferentes tipos de documentación (expurgo).

- Acceso y búsqueda concurrente

Por elementos descriptivos del documento creados en su captura o por el contenido del mismo en servidores geográficamente remotos. Visión y edición mediante el visor del sistema, navegadores o invocando a la aplicación nativa del documento.

- Múltiples formatos

Para la difusión de un documento o conjunto de documentos: pdf, html, xml, para ambientes de trabajo y usuarios heterogéneos.

- Integración del sistema con el resto de aplicaciones de productividad

Herramientas de escritorio (*MSOffice*) que se utilizan en las organizaciones como: bases de datos, trabajo en grupo (*Notes/Domino, MExchange*), aplicaciones ERP. Ello mediante estándares como ODMA (*Open Document Management API*) o otras herramientas API y de conectividad.

- Adaptabilidad para múltiples arquitecturas de red y plataformas

Cliente/servidor, Intranet, etc., ofreciendo funciones de escalabilidad y adecuación a las características geográficas y de crecimiento de la organización.

1.4.1 Herramientas para la Gestión Documental de Proyectos

Las herramientas que se describen a continuación, fueron seleccionadas de una amplia gama de productos del mercado. Estas incorporan en mayor o menor medida los diferentes campos que abarca la

GD: control de versiones, gestión electrónica de documentos, gestión de imágenes, etc.; utilizando los últimos avances de la tecnología.

➤ **Sistema de Gestión de Contenidos (CMS).**

La diversidad de CMS de código abierto existente, es extraordinariamente amplia, cada uno de ellos poseen las funcionalidades generales de cualquier manejador de contenidos, además de características particulares que los identifican. Habitualmente todo el software de código abierto es de acceso libre, sin ningún coste en licencias, lo cual es muy ventajoso en este caso. Muchos de estos CMS son utilizados como herramientas de gestión documental, en dependencia de las necesidades de la organización que lo vaya a utilizar. Permiten la recuperación y reutilización de documentos, y en general de cualquier objeto publicado o almacenado. Presentan control de acceso gestionando permisos aplicados a grupos o individuos. Dentro de los CMS más conocidos se encuentran:

- Plone.

Permite que se creen y adicione diferentes tipos de contenido; todos los contenidos son adicionados y editados de forma similar. Como miembro del sitio se cuenta con una carpeta donde se puede guardar el contenido creado. Permite el trabajo con documentos, imágenes, ficheros, vínculos, tópicos, carpetas y artículos noticiosos. Puede utilizarse como servidor intranet o extranet, un Sistema de Publicación de documentos y una herramienta de trabajo en grupo para colaborar entre entidades distantes. Realiza el proceso de gestión de objetos en el sitio mediante un sistema de flujo de trabajo por defecto basado en los estados de los objetos y los roles de usuario (14).

- Mambo.

Es una aplicación escrita en lenguaje PHP. Permite la creación y mantenimiento de sitios web y portales de manera fácil y dinámica, permitiendo al dueño o administrador de una página web la simplicidad para actualizarla y hacerla accesible a todo tipo de usuarios a través de una variedad de instrumentos. La simplicidad de Mambo radica en que no son necesarios conocimientos técnicos ni especializados para crear, mantener, actualizar o personalizar los contenidos de un sitio Web (15).

- Drupal.

Es una plataforma dinámica para la construcción de sitios Web que permite a un individuo o una comunidad de usuarios publicar, manejar y organizar una variedad de contenido. Integra muchas características populares de los CMS, weblogs, herramientas de colaboración y comunidad de discusión,

todo en un solo paquete fácil de utilizar. Como software de código abierto desarrollado y mantenido por una comunidad, Drupal es libre para descargarlo de Internet y usarlo (16).

- OpenCMS.

Basado en Java y en tecnología XML. Se trata de una aplicación CMS con características tales como Entorno de trabajo basado en navegador web, Gestión de activos, Sistemas de gestión de usuarios y permisos integrados, Publicación de contenidos basada en proyectos, Gestión de Workflow y tareas, Editor WYSIWYG, Soporte a la internacionalización, Versionado del contenido, Mecanismos de plantillas JSP y XML, Soporte Multi-idioma, Sistema de Ayuda Online, Publicación dinámica y estática de contenidos, Personalización, Sistemas de cacheo integrados, Mecanismo modular para las extensiones, Sistema de programación de trabajos, Mecanismo de Sincronización, Importación y Exportación de Contenidos, e Integración con el servidor de aplicaciones (17).

➤ Wiki.

Si bien es cierto que se puede utilizar un CMS para hacer gestión documental, existen alternativas especializadas que probablemente son más apropiadas. En el caso del wiki, sus aplicaciones están más orientadas a la gestión documental, que exige mucha más organización y ordenación, y son sumamente colaborativas.

Básicamente un wiki es un almacén de páginas web donde todos los usuarios pueden escribir y modificar los contenidos. Una buena cualidad es la de poder incorporar lo documentado, haciéndolo a la vez más dinámico, y ágil de captar, debido a su capacidad de poder ‘colgar’ esa misma documentación, adaptándola para una mejor comprensión. El wiki, da la posibilidad de ver la información de una forma muy rápida desde cualquier sitio, obtenerla y detectar si el procedimiento es el apropiado, y si no, poder tomar medidas. (18)

➤ GIT-DOC.

GIT-DOC es la integración de las soluciones que tradicionalmente se habían desarrollado de forma independiente para satisfacer las diferentes demandas de la gestión documental, que principalmente son: la captura e indexación de documentos, la recuperación de la información, la gestión de contenidos, y la automatización de flujos de trabajo. GIT-DOC es un sistema de gestión documental integral, completamente adaptable a las necesidades específicas de cada compañía y de cada proyecto de gestión documental, pudiéndose adaptar a cualquier requerimiento, desde la automatización de un

proceso de trabajo específico, a la evolución de una organización tradicional hacia una “empresa sin papeles”.

Es lo suficientemente flexible y escalable, de tal modo que permite la integración de nuevas funcionalidades de acuerdo a la evolución de las necesidades futuras de la empresa y a los cambios tecnológicos que se produzcan (19).

➤ **Extensión de Gestión de Documentos para Microsoft Exchange (DME)**

Es un sistema de administración empresarial que tiene la capacidad de almacenar cualquier tipo de documento, desde el momento mismo de su creación hasta que sea necesaria su destrucción, asegurando que toda la información estará disponible durante ese lapso para los integrantes de la organización que requieran tener acceso a ella. Además, DME cuenta con una interfaz totalmente integrada a Windows, se adapta a las aplicaciones de Microsoft Office y también provee una interfaz web, que permite realizar búsquedas, recuperar y guardar documentos on-line para que todos en su empresa dispongan de los datos que necesiten. (20)

➤ **Subversion (SVN)**

Es un sistema libre de control de versiones y de código fuente abierto, maneja ficheros y directorios a través del tiempo. Hay un árbol de ficheros en un repositorio central. El repositorio es como un servidor de ficheros ordinario, excepto porque recuerda todos los cambios hechos a sus ficheros y directorios. Esto le permite recuperar versiones antiguas de sus datos, o examinar el historial de cambios de los mismos. Además puede acceder al repositorio a través de redes, lo que le permite ser usado por personas que se encuentran en distintos ordenadores. A cierto nivel, la capacidad para que varias personas puedan modificar y administrar el mismo conjunto de datos desde sus respectivas ubicaciones fomenta la colaboración. Si se ha hecho un cambio incorrecto a los datos, simplemente se deshace ese cambio. (21)

1.4.2 Productos comerciales más representativos.

• **Documentum Enterprise Document Management System 98**

Potente sistema que automatiza y controla el ciclo vital de grandes cantidades de documentos dentro de las organizaciones, a través de diferentes plataformas. Documentum 4i gestiona y procesa el contenido de portales corporativos en ambientes web. (22)

• FileNET Panagon IDM Document Services

Aporta una serie de programas y herramientas para la gestión de la documentación en las organizaciones de manera distribuida a través de procesos de *workflow* tanto en arquitecturas de cliente/servidor como en ambientes web. (23)

• DOCS Open EDMS

Basado en arquitectura cliente/servidor captura, almacena y gestiona la información de múltiples formatos de documentos. Su extensión DOCS Binder provee una herramienta para la creación, gestión y distribución de conjuntos de documentos en formato XML. (24)

• Lotus Domino.DOC

Es una extensión de Notes/Domino para la gestión distribuida de documentos aportando potentes funciones de mensajería y trabajo en grupo. Asegura la integridad de la documentación desde su creación, hasta su archivado final en diferentes dispositivos, todo ello aprovechando la potencialidad del trabajo en grupo de Notes/Domino. Existen otras soluciones comerciales con diferentes características en cuanto a sus posibilidades y estructuras, como son Docuware, Novasoft ó Cindoc de Chemdata (25)

• Hyperwave Information Server

Hyperwave Information Server complementado con Hyperwave Information Portal para dotar a la empresa de una completa solución de gestión del conocimiento. Hyperwave Information Server (HIS) constituye el motor y la base que permite el manejo de la información.

Entre las características fundamentales que aporta HIS están la autenticación de usuarios, las funciones de publicación y creación de documentos, búsquedas, grupos de discusión, etc. Si sus permisos se lo permiten, los usuarios pueden contribuir a la información y el conocimiento común de la organización publicando sus propios documentos o retocando los existentes. Los documentos pueden ser simples archivos HTML, DOC o en cualquier otro formato o bien pueden ser tipos complejos como clústeres o secuencias El precio del producto es bastante alto aunque depende del número de usuarios que lo vayan a utilizar y en cualquier caso es asequible si se tiene en cuenta las empresas a las que va dirigido. (26)

Luego de haberse analizado cómo se desarrolla nivel mundial la Gestión Documental en los procesos de desarrollo de software se puede llegar a distintas conclusiones, que serán analizadas con mayor profundidad en el siguiente epígrafe, donde se realiza un análisis crítico del entorno y las soluciones existentes en el mercado.

1.5 Análisis crítico

Después del análisis de las soluciones existentes no se optó por la selección de ninguna de las herramientas y productos estudiados para la gestión del EP. Las razones más significativas se basan en que algunos de estos productos son software propietario y en Cuba se aboga por la utilización del software libre, teniendo en cuenta el ahorro de recursos económicos que esto implica. Por otra parte manipulan los documentos después de haber sido generados y no durante su proceso de elaboración, además no manejan los datos almacenados en los mismos. Esto no soluciona el problema planteado, ya que el objetivo es facilitar la confección del EP manipulando directamente su contenido; aún así es válido señalar que la solución que se desea obtener se inserta dentro del término GD, aportando una funcionalidad más en el manejo de la información.

Si lo que se desea lograr es un tratamiento global de los documentos, se hace necesaria una herramienta que integre la gestión documental y el trabajo directo con el contenido de los mismos. El EP actual está compuesto por documentos electrónicos que solo pueden ser manejados íntegramente, impidiendo manipular independiente de información que contiene. Por lo tanto se hace preciso diseñar una estructura para el contenido del EP, que permita independizar los datos para que puedan ser manipulados de una manera más flexible. Esta estructura podría facilitar la automatización del EP a través de una aplicación, dando la posibilidad de validar las características estructurales y semánticas de cada documento y permitiendo la reutilización de la información.

En este capítulo se definen los principales conceptos referentes al manejo y uso de la información. Se estudiaron las soluciones alternativas que existen en el mercado internacional, así como, algunas de sus ventajas y las principales desventajas que influyen directamente en el problema a resolver. También se realizó un análisis de la Gestión Documental y su aplicación en la Industria de Software. Además se abordaron las características principales del EP y su importancia.

CAPITULO 2: TENDENCIAS Y TECNOLOGÍAS

En este capítulo se realiza un análisis sobre las principales tecnologías y tendencias a nivel internacional para el tratamiento de la información. Se describen detalladamente aspectos importantes sobre lenguajes de programación y sistemas gestores de bases de datos definiendo finalmente cuales son las herramientas adecuadas para el desarrollo de la solución propuesta.

2.1 Técnicas y procesos para el manejo de la información

Existen numerosas técnicas para el procesamiento inteligente de la información, basadas en el desarrollo tecnológico alcanzado en la actualidad. Estas permiten manejar el conocimiento que se materializa a través de la información, la que puede ser manipulada en su manera más simple, desde datos hasta grandes volúmenes de documentación. Esto proporciona facilidades como, la confección de resúmenes inteligentes, la deducción de características y comportamientos en dependencia de los datos almacenados en las bases de datos, entre otras.

2.1.2 Minería de datos

La minería de datos (Data Mining) engloba un conjunto de técnicas encaminadas a la extracción de conocimiento procesable, implícito en las bases de datos. Las bases de la minería de datos se encuentran en la inteligencia artificial y en el análisis estadístico y mediante los modelos extraídos utilizando técnicas de minería de datos se aborda la solución a problemas de predicción, clasificación y segmentación. (27)

La extracción de información oculta y predecible de grandes bases de datos, es una poderosa tecnología con gran potencial para ayudar a las compañías a concentrarse en la información más importante de sus Bases de Información. Las herramientas de Data Mining predicen futuras tendencias y comportamientos, permitiendo en los negocios tomar decisiones proactivas y conducidas por un conocimiento acabado de la información. Además pueden responder a preguntas de negocios que tradicionalmente consumen demasiado tiempo para poder ser resueltas y a los cuales los usuarios de esta información casi no están dispuestos a aceptar. Estas herramientas exploran las bases de datos en busca de patrones ocultos, encontrando información predecible que un experto no puede hallar porque esta fuera de sus expectativas.

Un proceso típico de minería de datos consta de los siguientes pasos generales: (28)

1. Selección del conjunto de datos, tanto en lo que se refiere a las variables dependientes, como a las variables objetivo, como posiblemente al muestreo de los registros disponibles.
2. Análisis de las propiedades de los datos, en especial los histogramas, diagramas de dispersión, presencia de valores atípicos y ausencia de datos (valores nulos).
3. Transformación del conjunto de datos de entrada, se realizará de diversas formas en función del análisis previo, con el objetivo de prepararlo para aplicar la técnica de minería de datos que mejor se adapte a los datos y al problema.
4. Seleccionar y aplicar la técnica de minería de datos, se construye el modelo predictivo, de clasificación o segmentación.
5. Evaluar los resultados contrastándolos con un conjunto de datos previamente reservado para validar la generalidad del modelo.

Si el modelo final no superara esta evaluación el proceso se podría repetir desde el principio o, si el experto lo considera oportuno, a partir de cualquiera de los pasos anteriores. Esta retroalimentación se podrá repetir cuantas veces se considere necesario hasta obtener un modelo válido.

Una vez validado el modelo, si resulta ser aceptable (proporciona salidas adecuadas y/o con márgenes de error admisibles) éste ya está listo para su explotación. Los modelos obtenidos por técnicas de minería de datos se aplican incorporándolos en los sistemas de análisis de información de las organizaciones, e incluso, en los sistemas transaccionales. En este sentido cabe destacar los esfuerzos del Data Mining Group, que está estandarizando el lenguaje PMML (*Predictive Model Markup Language*) de manera que los modelos de minería de datos sean interoperables en distintas plataformas, con independencia del sistema con el que han sido construidos. Los principales fabricantes de sistemas de bases de datos y programas de análisis de la información hacen uso de este estándar.

Tradicionalmente, las técnicas de minería de datos se aplicaban sobre información contenida en almacenes de datos. De hecho, muchas grandes empresas e instituciones crean y alimentan bases de datos especialmente diseñadas para proyectos de minería de datos en las que centralizan información potencialmente útil de todas sus áreas de negocio. No obstante, actualmente está cobrando una importancia cada vez mayor la minería de datos desestructurados como información contenida en ficheros de texto y en Internet.

2.1.2 Minería de textos

El término minería de textos abarca una amplia gama de dominios, como son la visualización y organización de información y la minería de datos aplicada a textos. Es un conjunto de técnicas que permiten extraer información relevante y desconocida de forma automática dentro de grandes volúmenes de información textual, normalmente en lenguaje natural y no necesariamente estructurada. Se enmarca dentro de las muchas ramas de la lingüística computacional, apoyándose en otras dos técnicas, que son la inteligencia artificial y la lingüística, y su principal meta es la comprensión automática de textos mediante una representación formal del mismo.

La minería de textos opera sobre bases de datos textuales no estructuradas con el objetivo de detectar patrones no triviales e incluso información sobre el conocimiento almacenado en las mismas. De modo que estos sistemas pueden ayudar en la categorización de la información existente en una organización, en el filtrado y enrutado de información por ejemplo de e-mail, en la detección de información similar o relacionada con otra existente o para eliminar información duplicada. (29)

La principal diferencia de la minería de textos con la minería de datos es que en ésta última la información se obtiene normalmente de bases de datos, en la que la información está estructurada. Por este motivo es más sencilla la extracción de la información de una base de datos, que está pensada para que se pueda tratar su información de manera automática, al contrario a lo que ocurre en la minería de textos. También es necesaria diferenciarla de lo que en el entorno Web se denomina *búsqueda*, ya que en esta se busca información ya conocida, y que se ha puesto por escrito por otras personas. (30)

Dentro del ámbito del acceso, recuperación y organización de información, la minería de textos tiene un campo importante de aplicación que es la Web. Existen diferentes tipos de minería de textos web, la de contenido, la de estructura y la de uso.

Tipos de minería de textos web

- *Minería web de contenido*

Dentro de la Web se encuentran gran cantidad de documentos heterogéneos, ya sean hipertexto, en formato pdf, así como imágenes o vídeos, lo que dificulta su clasificación. La minería de contenido de la web trata de extraer información relevante de esta, de manera que pueda ayudar a clasificarlo,

aumentando la organización del mismo, para posteriormente mejorar el acceso y la recuperación de la información en él contenida.

- *Minería web de estructura*

Éste tipo de minería de textos sirve para saber cómo está organizada una web, cómo está estructurada y cómo es la navegación a través de ella.

- *Minería de uso del web*

Trata de extraer patrones de uso de la web por parte de los usuarios. Para ello se utilizan logs de los servidores Web de forma que aplicando minería de textos sobre ellos se pueda extraer información útil. Este tipo de minería tiene 2 objetivos principales: uno es sacar patrones generales de uso de un sitio web de manera que se pueda reestructurar para que sea más fácil de utilizar y mejore el acceso por parte de los usuarios. El otro uso es obtener perfiles de los distintos tipos de usuarios a través de su comportamiento y navegación, para poder atender de forma más personalizada.

De modo general la Minería de texto es el proceso encargado del descubrimiento de conocimiento que no existe en el texto, pero que surge al relacionar el contenido de varios textos; a partir de técnicas de aprendizaje automático de regularidades o patrones que se encuentran en un texto.

Dentro de las funcionalidades de la minería de texto en la Web es válido destacar la Web Semántica, por lo que se exponen sus principales características.

2.1.3 Web Semántica

La Web semántica se basa en la idea de añadir metadatos semánticos a la World Wide Web (WWW). Esas informaciones adicionales que describen el contenido, el significado y la relación de los datos se deben proporcionar de manera formal, para que así sea posible evaluarlas automáticamente por máquinas de procesamiento. El objetivo es mejorar Internet ampliando la interoperabilidad entre los sistemas informáticos y reducir la necesaria mediación de operadores humanos.

➤ **Componentes de la Web Semántica**

Los principales componentes de la Web Semántica son los metalenguajes y los estándares de representación XML, XML Schema, RDF, RDF Schema y OWL. Estos componentes están estrechamente relacionados y cada uno tiene una función determinada que hacen posible las funcionalidades de la Web

Semántica. La OWL (*Web Ontology Language Overview*) describe la función y relación de cada uno de estos componentes: (31)

- XML aporta la sintaxis superficial para los documentos estructurados, pero sin dotarles de ninguna restricción sobre el significado.
- XML Schema es un lenguaje para definir la estructura de los documentos XML.
- RDF es un modelo de datos para los recursos y las relaciones que se puedan establecer entre ellos. Aporta una semántica básica para este modelo de datos que puede representarse mediante XML.
- RDF Schema es un vocabulario para describir las propiedades y las clases de los recursos RDF, con una semántica para establecer jerarquías de generalización entre dichas propiedades y clases
- OWL añade más vocabulario para describir propiedades y clases: tales como relaciones entre clases, cardinalidad, igualdad, tipologías de propiedades más complejas, caracterización de propiedades, o clases enumeradas.

La usabilidad y aprovechamiento de la Web y sus recursos interconectados puede aumentar con la web semántica debido a:

- Los documentos etiquetados con información semántica (compárese ésta con la etiqueta <meta> de HTML, usada para facilitar el trabajo de los robots). Se pretende que esta información sea interpretada por el ordenador con una capacidad comparable a la del lector humano. El etiquetado puede incluir metadatos descriptivos de otros aspectos documentales o protocolarios.
- Vocabularios comunes de metadatos (Ontología (Informática)) y mapas entre vocabularios que permitan a quienes elaboran los documentos disponer de nociones claras sobre cómo deben etiquetarlos para que los agentes automáticos puedan usar la información contenida en los metadatos (p.ej. el metadato autor tenga el significado de "autor de la página" y no el del "autor del objeto descrito en la página").
- Agentes automáticos que realicen tareas para los usuarios de estos metadatos de la Web Semántica

- Servicios Web (a menudo con agentes propios) que provean de información a los agentes (por ejemplo un servicio de garantías a quien un agente pudiera consultar sobre si un comercio electrónico tiene un historial de mal servicio o de generar correo basura).

La Web Semántica a diferencia de la minería de textos que opera sobre bases de datos textuales no estructuradas, pretende construir toda una estructura de metadatos, información sobre la estructura y significado de los datos almacenados, e incluirlos en los documentos de forma que sean navegables, identificables y "comprensibles" por las máquinas. Por otro lado, la minería de datos pretende extraer los metadatos a partir de información textual no necesariamente estructurada. En este sentido, la minería de datos puede servir de ayuda para crear la telaraña semántica, mientras que los sistemas de minería de texto permiten el análisis léxico de los textos y especialmente la construcción automática de estructuras de clasificación y categorización que se codifican en forma de thesaurus.

El desarrollo de todas estas técnicas y procesos para el tratamiento inteligente de información, requieren la implementación de herramientas utilizando diferentes lenguajes de programación, por lo que se hace indispensable analizar algunos de ellos, enfatizando en aquellos que facilitan la estructuración de contenidos y que permitan desarrollar la solución propuesta.

2.2 Lenguajes

SGML

SGML son las siglas de *Standard Generalized Markup Language- Lenguaje de Marcado Generalizado*. Consiste en un estándar para la organización y etiquetado de documentos. El lenguaje SGML sirve para especificar las reglas de etiquetado de documentos y no impone en sí ningún conjunto de etiquetas en especial. La industria de la publicación de documentos constituye uno de los principales usuarios del lenguaje SGML. Empleando este lenguaje, se crean y mantienen documentos que luego son llevados a otros formatos finales como HTML, PostScript, RTF, etc. (32)

HTML

HTML, sigla de *Hypertext Markup Language-Lenguaje Etiquetas de Hipertexto*, es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. Este lenguaje no describe la apariencia del diseño de un documento sino que ofrece a cada plataforma que le de

formato según su capacidad y la de su navegador. Por ello no se deben diseñar los documentos basándose en como lucen en el navegador sino centrarse en proporcionar un contenido claro y bien estructurado que resulte fácil de leer y entender. (33)

XML

XML (*Extensible Markup Language- Lenguaje de Marcas Extensible*»), es un metalenguaje extensible de etiquetas, una simplificación y adaptación del SGML que permite definir la gramática de lenguajes específicos. Por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. Algunos de estos lenguajes que usan XML para su definición son XHTML, SVG, MathML. (34)

Características esenciales del XML

- Simplifica el movimiento de datos entre sistemas diferentes, sus capacidades de "metadatos" y "metalenguaje" facilita el despliegue dentro de las aplicaciones WEB complejas.
- Puede convertirse a cualquier formato para su despliegue.
- No es un lenguaje de presentación, sino para describir datos lo que facilita una búsqueda mas precisa, permite compartir datos de forma eficiente y hace la navegación más fácil.
- Permite definir nuevas etiquetas y atributos en un documento donde las estructuras pueden anidarse a cualquier nivel de complejidad, lográndose un control absoluto del mismo.
- Se pueden transportar enormes cantidades de datos bien organizados juntos con descriptores que indican los elementos y estructura de datos que contiene el documento.

La capacidad de separar la información (XML) de la (HTML) o alguna otra interfaz, es atractivo para desarrolladores, debido a que la información se puede mantener fresca en su propia Base de Datos y no en una página HTML estática. Esto permitirá que los fabricantes eviten tener que desarrollar una GUI independiente para cada plataforma de administración de Red. El XML es superior al HTML en cuanto a la extensibilidad, estructura y validación ya que los documentos XML están obligatoriamente estructurados, lográndose como resultado una jerarquía de datos.

Un documento XML puede contener una descripción opcional de su gramática para uso de las aplicaciones que necesiten realizar una validación de la estructura del mismo. Proporciona información actual que generalmente presenta vínculos hacia el contenido. Posee además la capacidad específica de ordenar la información por categorías dentro de las páginas WEB, además de otras fuentes como Microsoft Word,

BD, mensajes de Correo electrónico utilizando diccionarios de gramática especializada DTD (*Document Type Definition*) o XML Schema. Lo que lo hace más potente es que cualquier tipo de dato, incluso conceptos abstractos, pueden tener forma y estructura. Dado dos conceptos de datos, se les da forma describiendo sus componentes y la relación entre estos componentes.

La tecnología XML busca dar solución al problema de expresar información estructurada de la manera más abstracta y reutilizable posible. Qué la información sea estructurada quiere decir que se compone de partes bien definidas, y estas a su vez de otras. Entonces se tiene un árbol de pedazos de información. Juega un papel importantísimo en este mundo actual, que tiende a la globalización y la compatibilidad entre los sistemas, ya que es la tecnología que permitirá compartir la información de una manera segura, fiable y fácil. Además, XML permite al programador dedicar sus esfuerzos a las tareas importantes cuando trabaja con los datos, ya que algunas tareas tediosas como la validación de estos o el recorrido de las estructuras corre a cargo del lenguaje y está especificado por el estándar, de modo que el programador no tiene que preocuparse por ello.

El mundo XML no es un lenguaje, sino varios lenguajes, no es una sintaxis, sino varias y no es una manera totalmente nueva de trabajar, sino una manera más refinada que permitirá que todas las anteriores se puedan comunicar entre sí sin problemas, ya que los datos cobran sentido. (35)

XSL

XSL siglas de *Extensible Stylesheet Language*, expresión inglesa traducible como "lenguaje extensible de hojas de estilo" es una familia de lenguajes basados en el estándar XML que permite describir cómo la información contenida en un documento XML cualquiera debe ser transformada o formateada para su presentación en un medio.

Esta familia está formada por tres lenguajes:

- XSLT siglas de *Extensible Stylesheet Language Transformations*, lenguaje de hojas extensibles de transformación), que permite convertir documentos XML de una sintaxis a otra (por ejemplo, de un XML a otro o a un documento HTML).
- XSL-FO (*lenguaje de hojas extensibles de formateo de objetos*), que permite especificar el formato visual con el cual se quiere presentar un documento XML, es usado principalmente para generar documentos PDF.

- XPath, o XML Path Language, es una sintaxis (o lenguaje) (no basada en XML) para acceder o referirse a porciones de un documento XML.

Estas tres especificaciones son recomendaciones oficiales del World Wide Web Consortium (W3C). Desde el 2005 ya son soportadas por algunos navegadores, por ejemplo Mozilla o Internet Explorer, aunque, en su lugar, se pueden usar las CSS que son 100% compatibles aunque con una codificación diferente. (36)

RDF

RDF siglas de *Resource Description Framework- Marco de Descripción de Recursos* es un framework para metadatos en la WWW, desarrollado por el W3C. Este modelo se basa en la idea de convertir las declaraciones de los recursos en expresiones con la forma sujeto-predicado-objeto (conocidas en términos RDF como tripletes). El sujeto es el recurso, es decir aquello que se está describiendo. El predicado es la propiedad o relación que se desea establecer acerca del recurso. Por último, el objeto es el valor de la propiedad o el otro recurso con el que se establece la relación. La combinación de RDF con otras herramientas como RDF Schema y OWL permite añadir significado a las páginas, y es una de las tecnologías esenciales de la Web semántica. (37)

- **RDF Schema**

RDFS o RDF Schema es una extensión semántica de RDF. Un lenguaje primitivo de ontologías que proporciona los elementos básicos para la descripción de vocabularios. Los lenguajes de ontología permiten la formulación de un exhaustivo y riguroso esquema conceptual dentro de un dominio dado, con la finalidad de facilitar la comunicación y la compartición de la información entre diferentes sistemas. Existen actualmente otros lenguajes de ontologías más potentes, como el OWL. Un archivo RDFS se trata de un archivo con la misma sintaxis y la misma estructura que la que se usa en RDF. La sintaxis está basada en XML, es extensible, cada desarrollador puede extender el esquema RDF de manera independiente. (38)

OWL

OWL es el acrónimo del inglés *Ontology Web Language*, un lenguaje de marcado para publicar y compartir datos usando ontologías en la WWW. OWL tiene como objetivo facilitar un modelo de marcado construido sobre RDF y codificado en XML. Tiene como antecedente DAML+OIL, en los cuales se inspiraron los

creadores de OWL para crear el lenguaje. Junto al entorno RDF y otros componentes, estas herramientas hacen posible el proyecto de web semántica.

Actualmente, OWL tiene tres variantes: *OWL Lite*, *OWL DL* y *OWL Full*. Estas variantes incorporan diferentes funcionalidades, y en general, *OWL Lite* es más sencillo que *OWL DL*, y *OWL DL* es más sencillo que *OWL Full*. *OWL Lite* está construido de tal forma que toda sentencia pueda ser resuelta en tiempo finito, la versión más completa de *OWL DL* puede contener 'bucles' infinitos. (39)

DTD

DTD (*Document Type Definition*) La definición de tipo de documento (DTD) es una descripción de estructura y sintaxis de un documento XML o SGML. Su función básica es la descripción del formato de datos, para usar un formato común y mantener la consistencia entre todos los documentos que utilicen la misma DTD. De esta forma, dichos documentos, pueden ser validados, ya que conocen la estructura de los elementos y la descripción de los datos que traen consigo. A lo cual se adiciona la posibilidad de compartir la misma descripción y forma de validación dentro de un grupo de trabajo que usa el mismo tipo de información.

Las DTD se emplean generalmente para determinar la estructura de un documento mediante etiquetas (*en inglés tags*) XML o SGML. Una DTD describe:

- Elementos: indican qué etiquetas son permitidas y el contenido de dichas etiquetas;
- Estructura: indica el orden en que van las etiquetas en el documento;
- Anidamiento: indica qué etiquetas van dentro de otras.

Un esquema basado en una DTD tiene numerosas limitaciones. Una DTD no permite definir elementos locales que sólo sean válidos dentro de otros elementos. La falta de jerarquía en una DTD obliga a introducir una jerarquía a base de guiones o puntos en el espacio de nombres (*Namespace*). En una DTD es poco flexible la definición de elementos con contenido mixto, es decir, que incluyan otros elementos además de texto. Además no es posible indicar a qué tipo de dato (número, fecha, moneda) puede corresponder un atributo o el texto de un elemento.

La necesidad de superar estas limitaciones propicia la aparición de otros lenguajes de esquema como XML Schema, herramientas más completas de descripción que son una alternativa a las DTD. (40)

Esquemas en XML

- **XML Schema**

XML Schema es un lenguaje de esquema utilizado para describir la estructura y las restricciones de los contenidos de los documentos XML de una forma muy precisa, más allá de las normas sintácticas impuestas por el propio lenguaje XML. Se consigue así, una percepción del tipo de documento con un nivel alto de abstracción. Está basado en la gramática y confeccionado para proporcionar una mayor potencia expresiva que las DTD, las cuales poseen una mayor limitación en la descripción de los documentos a nivel formal.

Los XML Schema se concibieron como una alternativa a las DTD, intentando superar sus puntos débiles y buscar nuevas capacidades a la hora de definir estructuras para documentos XML. El principal aporte de XML Schema es el gran número de los tipos de datos que incorpora. De esta manera, XML Schema aumenta las posibilidades y funcionalidades de aplicaciones de procesamiento de datos. Entre sus características más notorias está el manejo de datatypes complejos y simples. Contiene mecanismos como la herencia para elementos y atributos. (41)

Los esquemas se construyen a partir de diferentes componentes, estos ofrecen la posibilidad de combinar características de alto o bajo nivel. En el alto nivel se encargan de ofrecer un significado semántico del contenido del documento, analizan el contenido y extraen de él un significado, el que puede estar predefinido en la declaración del esquema o se puede extraer de la misma estructura. En el bajo nivel se ofrecen características más concretas del documento que están incluidas en los diferentes campos del esquema y se accede a ellas de manera directa, se comparan directamente con el criterio de búsqueda definido y hallando palabras concretas en la definición de los esquemas. (42)

- **XDR**

XML-Data Reduced es la alternativa que nace principalmente por requerimientos del comercio electrónico, de ahí que su inicio fuera el XML-Data aunque también recibe la influencia de cXML, impulsado por Microsoft y Ariba. XDR actualmente es usado por el BizTalk framework de Microsoft.

- **DCD**

Document Content Description sometido a consideración a W3C por IBM y Microsoft, es un vocabulario RDF (*Resource Description Framework*) que propone un esquema para especificar reglas de estructura y contenido de documentos XML. DCD intenta definir restricciones en sintaxis XML que podrían ser usados como un tradicional DTD (43)

- **SOX**

Schema Object-Oriented XML, creado por Commerce One, fue inicialmente pensado para el desarrollo de aplicaciones de gran escala y distribuidas en el comercio electrónico. (44)

- **DSD**

Document Structure Description, desarrollado por laboratorios AT&T y BRICS. (45) Entre sus principales características se presenta como complemento al XSLT (XSL Transformations) en su poder de expresividad, permite la descripción de datos semiestructurados y su fortaleza es el manejo de clausuras o reglas para asegurar la integridad referencial de los datos "constraints".

- **RELAX NG**

Regular Language description for XML (46) diseñado por Makoto Murata, estandarizado por INSTAC XML SWG de Japón, bajo el auspicio de Japanese Standard Association (JSA). Actualmente sometido a consideración a normas ISO. Es una combinación del DTD y abundantes tipos de datos (*datatypes*) del XML Schema representado en sintaxis XML.

RELAX NG es también un lenguaje de esquema basado en la gramática, muy intuitivo y más fácil de entender que el XML Schema. Producto a esto tiene gran aceptación. Además posee un alto poder expresivo, por ejemplo permite validar elementos intercalados que pueden aparecer en cualquier orden. Las aplicaciones de definición de documentos y validación para RELAX NG son más sencillas que en XML Schema, haciéndolo más fácil de utilizar e implementar. También tiene la capacidad de usar plug-ins de definiciones de tipos de dato de XML Schema, combinando así las ventajas de ambos lenguajes. RELAX NG se ha convertido recientemente en un estándar ISO como la parte 2 de DSDL (*Document Schema Definition Language*).

- **Schematron**

Producto de la ASCC (*Centro de Control de Servicios Adaptados*) y creado por Rick Jelliffe, tiene la posibilidad de hallar árboles de patrones en el análisis gramatical del documento. Pudiendo ser aplicado sin diferencias lingüísticas como es el caso de documentos XML con información en lengua de oriente. (47)

Schematron, a diferencia de los anteriores lenguajes, se basa en afirmaciones y no en la gramática. Al basarse en una serie de reglas, utiliza expresiones de acceso en lugar de gramaticales para definir lo que se permite en un documento XML. Si el documento cumple estas reglas, entonces es válido. Este método de validación aporta una gran flexibilidad en la descripción de estructuras relacionales. En cambio, es un lenguaje muy limitado a la hora de especificar la estructura básica del documento, problema que se soluciona combinando Schematron con otros lenguajes de esquema, como RELAX NG.

Otras de las tecnologías que permite la estructuración y validación de contenidos, son las Base de Datos que cada día son más utilizadas por los implementadores y que proporciona grandes funcionalidades. (48)

2.3 Gestores de Base Datos

Una base de datos se puede definir como una colección o depósito de datos integrados con redundancia controlada y con una estructura que refleje las interrelaciones y restricciones existentes en el mundo real; los datos, que han de ser compartidos por diferentes usuarios y aplicaciones, deben mantenerse independientes de estas, y su definición y descripción, únicas para cada tipo de datos, han de estar almacenadas junto con los mismos.

Un Sistema de Gestión de Bases de Datos, SGBD, puede definirse como un paquete generalizado de software, que se ejecuta en un sistema computacional anfitrión, centralizando los accesos a los datos y actuando de interfaz entre los datos físicos y el usuario. Las principales funciones que debe cumplir un SGBD es relacionarse con la creación y mantenimiento de la base de datos, el control de accesos, la manipulación de datos de acuerdo con las necesidades del usuario, el cumplimiento de las normas de tratamiento de datos, evitar redundancias e inconsistencias y mantener la integridad. (49)

Entre los SGBD comúnmente utilizados en el mundo tenemos Oracle, MySQL, Microsoft SQL Server, PostgreSQL, Interbase, entre otros. Todos estos presentan un enfoque relacional con un buen basamento matemático centrado en el álgebra relacional. (50)

PostgreSQL

Es un sistema de gestión de base de datos objeto-relacional para múltiples plataformas. Debido a su licencia libre, puede usarse, modificarse, y distribuirse gratis para cualquier propósito, sea privado, comercial, o académico. Posee una estabilidad y confiabilidad legendaria. Tiene un buen soporte para subselects, triggers, vistas y procedimientos almacenados en el servidor. (51)

MySQL

Su principal objetivo de diseño fue la velocidad. Se sacrificaron algunas características esenciales de sistemas más serios con este fin. Otra característica importante es que consume muy pocos recursos, tanto de CPU como de memoria. Posee utilidades de administración que permiten realizar salvadas de la información y recuperarse de los errores. También soporta transacciones, rollbacks y subselects. Además considera las claves ajenas y la integridad referencial. (52)

SQL Server

Microsoft SQL Server, pertenece a la familia de los sistemas de administración de base de datos, que operan en una arquitectura cliente/servidor de gran rendimiento. Su desarrollo fue orientado para hacer posible manejar grandes volúmenes de información, y un elevado número de transacciones. SQL Server es una aplicación completa que realiza toda la gestión relacionada con los datos. Permite la creación de procedimientos almacenados que se ejecutan más rápido que instrucciones SQL independientes. Puede manejar perfectamente bases de datos de TeraBytes con millones de registros y funciona sin problemas con miles de conexiones simultáneas a los datos, sólo depende de la potencia del hardware del equipo en el que esté instalado. Corre sobre varias plataformas, Windows NT- 2000 Server, XP, Windows Server 2003 y Windows Vista. (53)

Oracle

Se considera como uno de los sistemas de bases de datos más completos, destacando su soporte de transacciones, estabilidad, escalabilidad y su condición de multiplataforma. Su mayor defecto es su enorme precio según versiones y licencias. Otro aspecto que ha sido criticado por algunos especialistas es la seguridad de la plataforma, y las políticas de suministro de parches de seguridad, modificadas a comienzos del 2005 y que incrementan el nivel de exposición de los usuarios. Aunque su dominio en el

mercado de servidores empresariales ha sido casi total hasta hace poco, recientemente sufre la competencia de la oferta de otros sistemas con licencia libre como PostgreSQL y MySQL. (54)

Es muy frecuente que los documentos XML al igual que las bases de datos sean utilizados para el almacenamiento de información, lo que trae grandes beneficios ya que resulta ligera la trasportación de los datos. No obstante, no es suficiente teniendo en cuenta que sus funcionalidades permiten principalmente la especificación del conjunto de relaciones y atributos, no permitiendo por si solos características que contienen las bases de datos como son: el trabajo con la información del dominio de valores asociados, restricciones de integridad, índices para cada relación y seguridad. Una buena práctica que ha tomado gran auge y que resuelve este problema es la integración de estas tecnologías.

2.4 El XML y las Bases de Datos

Las bases de datos relacionales ofrecen dos maneras fundamentales para el manejo de información basada en XML, dependiendo si son datos del comercio o si son el contenido del documento.

- Datos con formato (*Business Data*), necesitan ser almacenados, consultados, analizados y manipulados por diferentes aplicaciones. Esta información es almacenada de una manera óptima, internamente en una o más tablas relacionales.

- Datos sin formato (*Document Content*), tanto el texto como los datos multimedia necesitan ser ubicados de forma especializada de acuerdo al tipo de medio, y ensamblados eficientemente para producir documentos dinámicos.

Para almacenar en una BD mensajes o documentos basados en XML, es necesario que sean guardados en su óptima forma interna. Aplicaciones internas existentes y nuevas, podrían entonces trabajar eficazmente con esta información. Además cuando sea necesario extraer esta información para cooperar con otras aplicaciones, puede ser presentada una vista apropiada de los datos y el contenido específico del documento de una sola vez en un documento integro en XML. En la vida real son muchas las aplicaciones que necesitan tener junto los datos del negocio y el contenido del documento. Los datos con formato (*business data*) necesitan ser liberados como una parte íntegra del documento, mientras los datos sin formato (*document content*) tiene metadatos a los que se necesita acceder para su análisis, consulta y manipulación sin que esto signifique necesariamente acceder al contenido de características multimedia

(videos, sonidos, gráficos, etc.). Es justamente este el motivo por el que se busca mantener juntos tanto los datos con formato como los datos sin formato en la misma Base de Datos. (55)

Otra aplicación clara es cuando se pretende juntar diferentes vistas de una BD, pues se forman varios árboles estructurados de documentos XML dependiendo de lo que se desee lograr. Pero la importancia de esto es que conectando las tablas y vistas junto a los enlaces entre los datos apuntados por claves foráneas, el resultado se obtiene como diferentes documentos XML virtuales de información relativa a un mismo tema. En la búsqueda para lograr todo este manejo de datos XML en las BD, se han realizado varios estudios. A continuación se muestran algunos lenguajes utilizados para el desarrollo de herramientas de consulta sobre documentos XML:

XSLT

XSLT (*Transformaciones XSL*) es un lenguaje para la transformación de documentos XML en otros documentos XML. Es diseñado para formar parte de XSL y al igual que este, incluye un vocabulario XML para especificar formatos. Una transformación expresada en XSLT describe reglas para transformación de un árbol fuente a un árbol resultado. En la construcción de este último los elementos del árbol fuente pueden ser filtrados, reordenados, e incluso añadido estructuras arbitrarias.

XSLT realizan la transformación del documento utilizando una o varias reglas de plantilla: unidas al documento fuente a transformar, esas reglas de plantilla alimentan a un procesador de XSLT, el cual realiza las transformaciones deseadas colocando el resultado en un archivo de salida o, como en el caso de una página Web, directamente en un dispositivo de presentación, como el monitor de un usuario. Actualmente, XSLT es muy usado en la edición Web, generando páginas HTML o XHTML. La unión de XML y XSLT permite separar contenido y presentación, aumentando así la productividad. (56)

XSQL

Desarrollado por Oracle, las páginas XSQL son templates que permiten ensamblar bloques de páginas (*datapages*) XML dinámicas basadas en una o más consultas SQL parametrizadas, así como también transformar las *datapage* para producir un resultado final en HTML, XML o cualquier formato basado en texto usando el XSLT asociado. (57)

XML-GL

El XML-GL es un lenguaje gráfico de consulta o basado en representación gráfica de documentos XML. La originalidad de XML-GL con respecto a otras propuestas de lenguajes de consulta para XML tales como XML-QL, XQuery, XQL es que las consultas son formuladas visualmente, usando un formalismo basado en grafos. Sin embargo XML-GL no es una interfaz visual sobre un convencional lenguaje de consulta de texto, pues es basado en estructuras y operaciones de grafos tanto su sintaxis como su semántica. (58)

La utilización de lenguajes basados en XML posibilita un gran número de ventajas para el desarrollo de la Web cuando en materia de información se habla. Algunas de ellas están dadas por las libertades de implementación que son cada vez más acogidas entre los desarrolladores y que hacen más flexible y personalizado el desarrollo de software. Además permiten e incrementan la compatibilidad e interoperabilidad entre distintos componentes de hardware y software. Otras facilitan una mayor distribución y redifusión de la información. Algunos de estos temas serán ampliados a continuación.

2.5 Propuesta de tecnología a utilizar

Para resolver los problemas concretos que fueron identificados en la gestión actual del EP, se decidió diseñar una estructura para las plantillas que permita:

- Facilitar la confección de los documentos eliminando la repetición innecesaria de los datos
- Impedir la modificación estructural de los documentos del EP
- Manejar el contenido de los documentos de manera independiente.
- Validar la estructura del documento.
- Validar el contenido del documento.
- Permitir el fácil manejo de los documentos

Basándose en el análisis previo de las metodologías y tecnologías existentes y de las funcionalidades que se desean obtener con la estructuración de los documentos, se seleccionó a XML como lenguaje de desarrollo, específicamente el lenguaje de esquema XML Schema para implementar la estructura.

XML es un estándar internacionalmente reconocido, no pertenece a ninguna compañía y es libre, permite su uso efectivo en la Web, está diseñado para ser utilizado en cualquier lenguaje o alfabeto. Además es fácilmente procesable tanto por humanos como por software. Separa radicalmente la información y permite

poderosas técnicas de extracción de información y data mining. Las estrictas reglas para la composición de un documento XML posibilitan su fácil análisis sintáctico. En XML los tags también pueden actuar como el nombre de un campo en un programa, haciendo fácilmente identificable dicha información para aplicaciones de software, permitiendo de esta manera tener una estructuración de datos en una página web. (59)

Con la utilización de XML en aplicaciones es posible lograr que los datos sean almacenados internamente en una forma fácil y óptima. Cuando la información es requerida por otras aplicaciones esta se puede obtener en una vista apropiada directamente desde la base de datos liberada en un documento integro en XML.

La importancia de XML

Entre de los principales elementos que nos permiten ver la importancia de XML, están los siguientes:

- *Texto plano*, puesto que XML es un lenguaje de formato de texto estándar, es fácilmente editable, la depuración de sus programas también es fácil, y por otra parte utilizando XML como front end (parte del sistema que convierte la entrada del texto en una representación simbólico-fonética) de una base de datos es posible almacenar eficientemente gran cantidad de datos proporcionando escalabilidad desde pequeñas configuraciones de archivos hasta los grandes repositorios de compañías.
- *Identificación clara de datos*, ya que las diferentes partes de los datos están claramente identificadas, estos pueden ser manipulados de formas distintas por las diferentes aplicaciones de acuerdo a sus necesidades
- *Estilo*, al combinar XML con eXtensible Stylesheet Language XSL permite especificar como desplegar una etiqueta (tag), por ejemplo en HTML. Diferentes formatos XSL pueden ser usados para desplegar los mismos datos en forma diferente, para usos diferentes. (60)
- *Fácilmente procesable*, ya que la notación es más consistente y fácil, le hace especialmente sencillo para construir programas que procesen datos XML. Pues a diferencia del HTML donde el delimitador final es opcional o muy diferente, en XML existirá uno igual al elemento de inicio. Todos

los tags forman un documento XML bien formado (*well-formated*,) donde están completamente anidados.

- *Estructura Jerárquica*, como en toda estructura de este tipo se tiene la ventaja de poder acceder directamente a la parte del documento que sea necesaria, así como la posibilidad de reordenar con solo el mover su raíz, pues se estará moviendo todo lo que bajo esta se encuentre.

Por su parte XML Schema permite especificar de forma más detallada un extenso sistema de tipos de datos que tienen una analogía directa con lenguajes de programación y Bases de Datos. Puede ser analizado sintácticamente y manipulado como cualquier otro documento XML. Además define las limitaciones aplicables a los elementos y atributos que deben figurar en una clase de documentos XML, la relación entre ellos y qué tipos de datos pueden contener.

Los esquemas ayudan al desarrollo de vocabularios comunes y mejoran la interoperabilidad entre aplicaciones así como el intercambio de datos entre las distintas fuentes de información. Por tanto los esquemas son la principal cadena para proveer una estructura fiable para los datos que se encuentran en Internet, un paso fundamental para el comercio electrónico y para otros campos. Una ventaja esencial de un esquema XML es que nos permita las búsquedas con semántica, pues estas dejarían de ser por patrones textuales. Así los motores de búsqueda que utilicen XML y sus esquemas, tendrían la posibilidad de un adecuado entendimiento de los datos y su contexto. (61)

Para estructurar las plantillas del EP también podría utilizarse un gestor de base de datos, pero si lo que se desea es que las plantillas puedan cambiar con cierto dinamismo, lo indicado sería XML ya que permite mayor flexibilidad. Sería demasiado complejo que cuando ocurra un cambio en una plantilla se tengan que modificar las tablas contenidas en la Base de Datos de las instalaciones de la aplicación, es más fácil enviar o que las personas bajen un XML Schema.

2.5.1 Herramienta de desarrollo

De todas las herramientas que permiten el trabajo con XML fue seleccionado XML Spy en su versión 5.0 que como editor XML permite la elaboración de la estructura que se desea diseñar.

XML Spy

XML Spy 5.0, es una aplicación que tiene toda la potencia necesaria para trabajar en cualquier proyecto. La habilidad de desarrollar y administrar bases de datos, la combinación de soporte hacia los lenguajes de programación más avanzados de la actualidad, y las facilidades de uso, que igualmente requieren de un usuario con experiencia, son las razones por las que este producto ha sido tan comentado desde su lanzamiento.

Hay tres versiones de XML Spy 5.0; la Home Edition, la Professional Edition, finalmente se selecciono la Enterprise Edition. La Enterprise Edition posee la capacidad para realizar conversiones de archivos HTML a XML, utilidades para generar e incorporar código en Java y C++, y opciones especialmente destinadas al desarrollo Web. La Professional Edition no tiene las funciones principales de la Enterprise Edition, pero ofrece de todas formas suficientes herramientas como para trabajar a fondo con documentos XML, bases de datos, aplicaciones Web, y más. Finalmente, la versión más pequeña y la más compacta es la Home Edition, pero no hay razón para descartarla, sino que es una opción económica que posee herramientas básicas, pero bastante poderosas.

Las novedades que trae XML Spy 5.0 comienzan por el XSLT debugger (depurador), un término familiar para los desarrolladores, y que resulta de gran utilidad; otra interesante función posibilita la conversión de documentos HTML a XML, y aunque pueden surgir algunos errores, resulta una herramienta de mucha ayuda; la otra cualidad de XML Spy que resalta de la lista es la correspondiente a las bases de datos, ya que permite trabajar en varios escenarios, aunque falta el soporte hacia las extensiones DB2 y Sybase Adaptive Enterprise XML.

XML Spy 5.0 es el programa más grande que existe en relación a la programación XML. Sus avanzadas herramientas, y las novedades añadidas en esta última versión convierten a este software en una solución completa, que supera a la competencia ofreciendo mayores posibilidades de desarrollo. (62)

En este capítulo se realizó un análisis completo de las tecnologías que serán utilizadas a lo largo del desarrollo de la estructura propuesta, se estudiaron algunas tendencias utilizadas en la Web y se fundamentaron las elecciones del lenguaje seleccionado para diseñar la estructura. Además se hizo una descripción del XML Spy, herramienta de desarrollo que se va a utilizar.

CAPITULO 3: ANALISIS Y DISEÑO DE LA ESTRUCTURA

El capítulo tiene como objetivo principal realizar el análisis y diseño de la estructura que modela el contenido de las plantillas del Expediente de proyecto teniendo en cuenta las características de este y de los diferentes tipos de información contenida en los mismos.

3.1 Selección de las plantillas

El Expediente de Proyecto (EP) que se ha implantado en la Universidad de las Ciencias Informáticas (UCI) se encuentra en un período de prueba y aceptación, a pesar de estar basado en estándares y normas internacionales que son reconocidos desde hace años, constituye un mecanismo nuevo dentro de la Universidad, que por las diversas características de la misma, debe ser evaluado. Actualmente se estudia como lograr que el EP se ajuste de una manera más específica a cada proyecto productivo. Esta tarea resulta bastante difícil debido a la heterogeneidad de los proyectos que existentes en la Universidad

La UCI se divide en facultades que se especializan en un segundo perfil. Estos están relacionados con ramas de la economía que necesitan ser informatizadas según las prioridades del país y con temas sobre la producción de software que son de vital importancia para esta industria. Los proyectos productivos que se realizan en cada facultad, precisamente por estar orientados hacia un mismo perfil, presentan numerosas similitudes que permiten que sean documentados de manera semejante, aunque existen algunas excepciones que solo representa una minoría. Para lograr responder de manera general a las necesidades de todos los proyectos de la Universidad el EP actual está compuesto por un gran número de plantillas. Estas no son necesariamente utilizadas en su totalidad por los proyectos.

Teniendo cuenta lo antes planteado se decidió confeccionar un expediente de proyecto que permite satisfacer las necesidades básicas del proceso de documentación de un software, nombrado Expediente de Proyecto Elemental (EPE), que puede ser utilizado por cualquier proyecto de la Universidad. Es válido aclarar que esta propuesta de EPE se basa en una simplificación del EP propuesto por la Dirección de Calidad de Software (DCS) y que aún no está especificado para otras metodologías de desarrollo diferentes de RUP. Este aspecto está siendo estudiado para el lanzamiento de nueva versiones. El objetivo de esta propuesta es definir el expediente que va ser estructurado para garantizar una mejor confección de la documentación de los proyectos.

La selección de las plantillas que conforman el EPE se sustenta en el estudio de los lineamientos de calidad propuestos por DCS que deben cumplirse en la producción de software y en el análisis de la clasificación de la documentación establecida para los proyectos productivos.

3.1.1 Lineamientos de calidad

Los Lineamientos de Calidad establecen un grupo de normas y estrategias que deben seguir todos los proyectos productivos de la Universidad para garantizar la correcta organización de los mismos y la calidad de los productos desarrollados. Estos fueron elaborados teniendo en cuenta buenas prácticas de ingeniería de software (IS) y los requerimientos planteados por el modelo para la calidad de software CMMI en su nivel 2.

➤ Buenas prácticas de IS

Plantean la realización de una correcta gestión de proyecto, la integración de todos los factores que influyen en el desempeño de la organización como son: marketing, recursos humanos, finanzas, etc. Introducir prácticas de ingeniería en las empresas permite elevar la productividad, reducir costes, aumentar la calidad, entre otros beneficios. Aún así las prácticas de ingeniería deben ser estudiadas antes de ser ejecutadas teniendo en cuenta las características propias de cada organización (63) El impacto económico de las empresas es el elemento que debe guiar la introducción, y medida, de las prácticas de ingenierías. Esta relación que existe entre las empresas y la ingeniería de software se puede representar. (Ver figura 1)

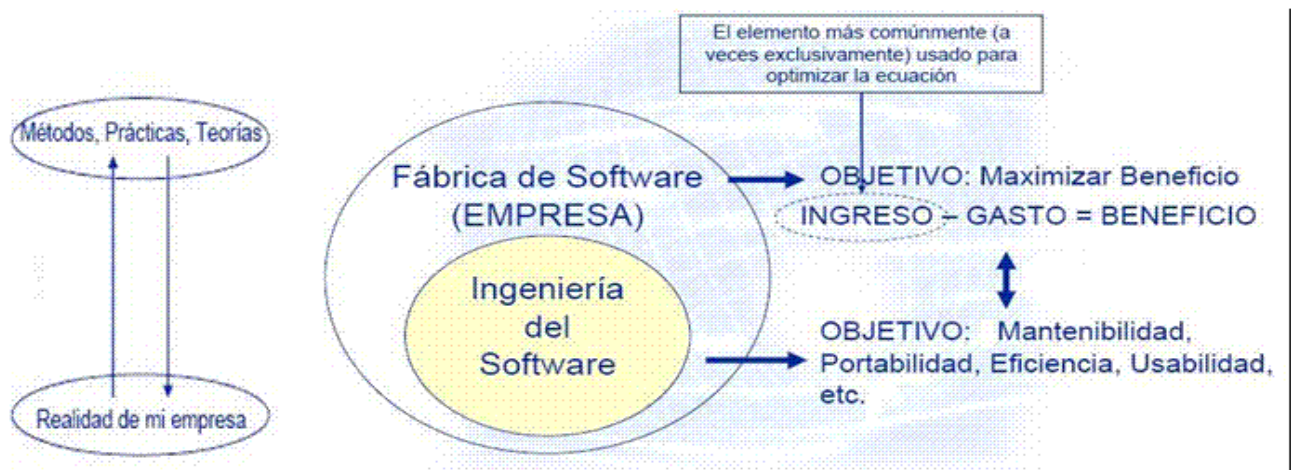


Figura1: Relación de la empresa y la IS.

➤ *Modelo de Madurez y Capacidad Integrado CMMI. Nivel 2.*

Lo que se pretende con el nivel 2 de CMMI es conseguir que en los proyectos de la organización haya una gestión de los requisitos y que los procesos estén planeados, ejecutados, medidos y controlados. (64)

Explicado un poco más:

- El uso de los procesos del nivel 2 permite conocer como se trabaja ante cualquier situación que se presente, se tienen en cuenta todo tipo de factores que pueden influir en la producción y se planifica como enfrentar cada uno de ellos. Cuando se realizan estas prácticas, los proyectos se ejecutan y gestionan de acuerdo con los planes definidos en el proyecto.

- El estado de los elementos de trabajo (análisis, diseño, código, documentación,...) están visibles a la gerencia en puntos definidos (hitos del proyecto). Se sabe cuánto trabajo está hecho y cuánto queda por hacer.

- Los compromisos adquiridos con todas las personas involucradas en el proyecto se revisan de acuerdo a las necesidades. Los elementos de trabajo se revisan con las personas involucradas y son controlados. Estos elementos de trabajo satisfacen las especificaciones, estándares y objetivos.

Estas ideas se materializan en las siguientes áreas de proceso:

- ✓ Gestión de Requisitos
- ✓ Planificación de proyectos
- ✓ Monitorización y Control de proyectos
- ✓ Medición y Análisis
- ✓ Aseguramiento de la calidad
- ✓ Gestión de la configuración

En la elaboración de los lineamientos se tuvieron en cuenta estas áreas de procesos, además estos fueron agrupados de la misma manera que las plantillas del EP logrando una correspondencia con el mismo. A continuación se muestran los lineamientos mínimos de calidad.

Generales

➤ Plan de capacitación

Definir estrategia de capacitación para el personal del proyecto que incluya cursos de postgrado, cursos optativos y cualquier otra modalidad definida por el proyecto, así como fomentar las maestrías, doctorados, tesis de pregrado, publicación de artículos, participación en eventos, etc.

➤ Equipos de proyectos definidos

Documentar la composición de los equipos de proyecto incluida la dirección del proyecto.

➤ Roles y responsabilidades definidas

Definir los roles y responsabilidades para cada miembro de los equipos incluida la dirección del proyecto.

➤ Definir equipo de Calidad de Software

Definir explícitamente el equipo que atenderá la Calidad de Software en el proyecto especificando los roles y responsabilidades para sus miembros.

Ingeniería

➤ Evaluar áreas de la organización

Realizar levantamiento de las áreas de la organización para ver el estado en el que se encuentra y ver sus necesidades.

➤ Gestión de requerimientos

Gestionar los requisitos de los productos del proyecto y componentes de producto. Identificar inconsistencias entre los requisitos y el plan de proyecto y productos del trabajo. Planificar la gestión de requisitos de software.

➤ Metodología de desarrollo definida

Definir metodología para el desarrollo del proyecto. Definir las fases y artefactos en el caso de que se modifique alguna de las metodologías existentes.

➤ **Arquitectura de software aprobada**

Definir la arquitectura de software a utilizar en la construcción de la solución. Esta arquitectura debe encontrarse aprobada por la Dirección Técnica de Producción.

➤ **Arquitectura de información definida**

Tener definida la arquitectura de información tanto para el caso de portales como de aplicaciones de escritorio.

➤ **Definir modelo de diseño**

Definir clases de diseño, diagrama de clases de diseño.

➤ **Definir estándares de código y documentación**

Contar con la definición de los estándares que se utilizarán para la escritura del código fuente para cada uno de los lenguajes a utilizar así como los estándares para la asignación de nombre a los documentos del proyecto.

➤ **Herramienta de modelado**

Utilizar herramienta CASE para el modelado aprobada por la Universidad de las Ciencias Informáticas.

➤ **Pruebas internas**

Realizar pruebas internas al proyecto por parte del grupo de calidad. Esto implica que el proyecto tenga definido el Plan de pruebas y mantenga constancias de las No Conformidades detectadas durante el proceso y las soluciones a las mismas.

Gestión de proyectos

➤ **Estimados de costo y esfuerzos**

Realizar estudio que refleje el costo del proyecto teniendo en cuenta los conceptos de licencias de software, compra de equipamiento, compra de bibliografía y otros que estime el proyecto. También debe aparecer reflejado el esfuerzo que conlleva la realización del proyecto reflejado en la cantidad de personas necesarias, el número de horas/hombres y el tiempo previsto para el desarrollo.

➤ **Desarrollar plan del Proyecto**

Definir los cronogramas para el proyecto, los planes para la ejecución de los procesos de desarrollo y cómo chequear este plan estableciendo puntos de chequeo.

➤ **Identificar riesgos del proyecto y establecer plan de mitigación de los mismos.**

➤ **Visión del proyecto.**

Definir la visión del proyecto en la fase inicial del mismo.

➤ **Gestión de recursos**

Contar con una planificación de los recursos tanto materiales como humanos

➤ **Registrar los acuerdos de trabajo y las minutas de las reuniones**

➤ **Pruebas de liberación y aceptación**

Tener en el cronograma de proyecto la planificación de las pruebas de liberación tanto las de la facultad como las de la DCS y las pruebas de aceptación por parte de Calisoft

Soporte

➤ **Herramienta para el control de versiones**

Utilizar herramienta para el control de versiones. Esta debe tener definido los permisos que tendrán cada uno de los trabajadores del proyecto. Todo lo referente al trabajo con la herramienta quedará definido en el Plan de Gestión de Configuración.

➤ **Plan de pruebas**

Definir Plan de pruebas para el proyecto donde se refleje el cronograma de pruebas y los responsables. Deben reflejarse de manera explícita los cronogramas para pruebas de liberación (Facultad y DCS) y aceptación (Calisoft).

➤ Plan de aseguramiento de la calidad

Tener definido el Plan de aseguramiento de la calidad donde queden definidos los estándares y procedimientos a utilizar en el proyecto.

➤ Definir glosario de términos

➤ Definir lista de chequeo para la revisión de los atributos de calidad

➤ Llevar control sobre las No Conformidades, las respuestas brindadas a las mismas y los pedidos de cambios

Cada lineamiento de calidad propone la confección de una o varias plantillas que conforman el EP y que satisfacen a los mismos.

Plantillas según los lineamientos

✓ Generales

Plantilla DCS- Plan de capacitación.

Plantilla DCS – Roles y responsabilidades.

✓ Ingeniería

Plantilla DCS – Evaluación de Áreas de la Organización.

Plantilla DCS - Plan de gestión de requisitos.

Plantilla DCS – Documento de arquitectura de software.

Plantilla DCS – Arquitectura de información

Plantilla DCS – Informe del levantamiento de información para la arquitectura.

Plantilla DCS – Diseño de Casos de Prueba

Plantilla DCS – Plan de Pruebas.

✓ Gestión de proyectos

Plantilla DCS – Plan de Desarrollo de Software

Plantilla DCS – Presupuesto.

Plantilla DCS – Documento Visión.

Plantilla DCS – Minuta de Reunión.

✓ Soporte

Plantilla DCS – Plan de Prueba.

Plantilla DCS -- Plan de aseguramiento de la calidad.

Plantilla DCS – Glosario de términos.

Plantilla DCS – Listas de chequeo.

Plantilla DCS – No conformidades (ampliada)

Plantilla DCS – No conformidades (reducida)

Plantilla DCS --Plan de aseguramiento de la calidad

Plantilla DCS --Plan de mediciones

Plantilla DCS – Respuestas a no conformidades

Plantilla DCS – Solicitud de cambio (reducida).

3.1.2 Clasificación de la Documentación

La documentación generada en los proyectos productivos de la Universidad fue clasificada por la DCS según las características de los documentos y teniendo en cuenta la importancia de su elaboración. Esta clasificación consiste en tres categorías:

- Rectores: Representa la documentación que es imprescindible elaborar en cada proyecto.
- Entregables: Representan la documentación que debe ser entregada por cada proyecto.
- Internos: Representa la documentación que depende de las características internas de cada proyecto.

Algunas de las plantillas del EP fueron agrupadas teniendo en cuenta estas clasificaciones:

Rectores:

- Modelo del Negocio o Modelo del Dominio
- Modelo de Casos de Uso del Sistema
- Especificación de Requisitos
- Documento de Arquitectura de Software
- Modelo de Diseño
- Modelo de despliegue
- Manual de Instalación
- Documento Visión

Internos:

- Arquitectura de Información
- Diseño de Casos de Pruebas
- Código fuente
- Plan de pruebas
- Lista de riesgos
- Plan de Mitigación de Riesgos
- Plan de Desarrollo de Software (Interno y entregable)
- Ambiente de desarrollo
- Plan de capacitación
- Registro de No Conformidades
- Respuesta a No Conformidades
- Listas de Chequeo
- Pedido de Cambio

Entregables:

- Evaluación de Áreas de la Organización
- Manual de Usuario
- Minuta de reunión (entregable e interno)
- Diagnóstico
- Glosario de Términos

Después del análisis previo y teniendo en cuenta que el propósito del EPE es su futura estructuración, se seleccionaron las plantillas que satisfacen a los lineamientos mínimos de calidad, excepto la plantilla Plantilla DCS – Presupuesto, que por las características estadísticas que presenta no se consideró gestionable dentro de la estructura. Además se tuvieron en cuenta las plantillas que dentro de EP están clasificadas como rectoras y que deben ser entregadas por todos los proyectos.

Finalmente para la estructura a desarrollar, de las 40 plantillas del EP propuesto por DCS, se proponen 21 plantillas que conforman EPE y que muestran las áreas de procesos Gestión de Proyecto, Ingeniería y Soporte, no incluyéndose ninguna de la carpeta Legal. Esta propuesta satisface las necesidades básicas de la documentación de los proyectos de la universidad.

3.2 Expediente de Proyecto Elemental (EPE)

Gestión de Proyecto

✓ Documento visión

Aparece una breve descripción de las oportunidades de negocio con el proyecto. Donde se expresan los elementos que lo hacen atractivo, las coyunturas políticas, económicas o específicas del sector del mercado a que pudiera estar dirigido. También se recogen los posibles problemas que pueden ser resueltos y las principales características del Mercado que motivan el proyecto. Se describe el entorno de trabajo del usuario principal, detalles de los involucrados y usuarios, las opciones de competencia que existen, incluyendo las principales fortalezas y debilidades de cada competidor. Define las perspectivas y funciones del producto, sus suposiciones y dependencias, así como los principales beneficios y características que el producto debe tener.

✓ Plan Desarrollo de Software

Muestra los elementos fundamentales definidos en la visión del Proyecto. Describe el propósito y los objetivos del mismo, al igual que un grupo de restricciones como son: el presupuesto, personal, equipo, cronograma que aplica al proyecto, entre otros. Se definen los artefactos que serán creados durante el mismo. Identifica la estructura organizativa del equipo, incluso la dirección y otras autoridades de la revisión. Proporciona el costo estimado y cronograma, así como la base para esas estimaciones, y los puntos y/o circunstancias en las cuales será necesaria una re-estimación. Planifica las actividades para ordenar el cierre del proyecto, reasignación del equipo y archivar los materiales del proyecto. Dentro de esta plantilla se hace referencia a otras que serán generadas posteriormente y que colaboran con el objetivo de este documento, estas son:

- Plan de capacitación
- Planes de Iteraciones
- Plan de Gestión de Requisitos
- Plan de Mediciones
- Plan de control de riesgos
- Plan de la Infraestructura
- Plan de aceptación del proyecto

- Plan de Gestión de Configuración
- Plan de aseguramiento de la calidad
- Plan de Resolución de Problemas

- ✓ Roles y responsabilidad

Se definen los roles que van a existir en el proyecto y las responsabilidades que van a tener. Además se asignan las responsabilidades a los miembros de los equipos en cada una de las fases del proyecto.

- ✓ Minuta de reunión

Se especifican los acuerdos tomados en las reuniones del proyecto.

- ✓ Plan de capacitación

Se describen las necesidades de capacitación del equipo, el alcance de la misma, el presupuesto requerido para su ejecución y el cronograma que deberá regir todas las tareas de capacitación.

Ingeniería

- ✓ Plan de gestión de requisito

Describe quien es responsable del desarrollo de las actividades descritas en el flujo de trabajo de requisito, las herramientas de software que serán utilizadas para la gestión de los requisitos mínimos y control de versiones, al igual que los beneficios que reportan el negocio. Además se hace referencia al documento de los procesos de Gestión de Configuración establecidos en la organización.

- ✓ Especificación de Requisitos

Describe los requisitos funcionales del sistema expresados en lenguaje natural. También se recogen los no funcionales agrupados por categorías: usabilidad, fiabilidad, soporte o mantenimiento, restricción de diseño en el sistema, componente comprado, interfaz de usuario, de hardware, de software y de comunicación, de licencia, legales, de derecho de autor y otros estándares aplicables.

- ✓ Modelo de dominio

Contiene el diagrama de clases del modelo del dominio y la definición y descripción de dichas clases.

✓ Modelo de CUN

Se especifican todos los actores, así como trabajadores del negocio y se le asocia una descripción simple de cada uno de ellos. Contiene el diagrama de Casos de Uso del Negocio, una descripción literal de estos, los diagramas de actividades y el modelo de objetos.

✓ Modelo de Casos de uso del sistema

Se especifican todos los actores del sistema y se asocia una descripción simple de cada uno de ellos. Contiene el diagrama de Casos de Uso del Sistema y describe los detalles de este en fusión de la acción del actor y la respuesta del sistema.

✓ Informe del Levantamiento de Información para la Arquitectura de Información

Determina cuál es el público al que estará orientado el producto organizado por categorías. Teniendo en cuenta características como capacidad física, capacidad técnica, conocimientos, necesidades de información y ubicación geográfica, especificando las necesidades y expectativas de la audiencia.

✓ Documento de Arquitectura de Software

Describe la arquitectura de software para el sistema, como se representa, los objetivos y restricciones arquitectónicas. Contiene el diagrama de casos de usos arquitectónicamente significativos, la representación de las clases más importantes, su organización en paquetes y subsistemas, y la organización de estos en capas, un diagrama que ilustra la organización del modelo de diseño en capas lógicas. Además se enumeran las tecnologías a aplicar por cada una de las capas y subsistemas. También contiene el diagrama de despliegue, de vista de procesos, de implementación y de datos. Este documento hace referencia a otras plantillas como son:

- Modelo de Casos de uso del sistema.
- Modelo de CUN (opcional).
- Modelo de Despliegue.

✓ Arquitectura de Información

Esta se utiliza para describir la navegabilidad de la aplicación. Muestra una representación simple de la estructura en cuanto a etiquetas y jerarquía de los contenidos. Describiendo textualmente cada uno de los elementos de la estructura, características y comportamiento. Contiene el Mapa de Navegación representado en forma de árbol de secciones, niveles y contenidos

✓ Plan de pruebas

Organiza el equipo de pruebas. Describe como los objetivos de la prueba serán alcanzados, los tipos de pruebas que hacen como parte del plan, la arquitectura técnica, las especificaciones de software, de hardware y los requerimientos funcionales. Además establece la estrategia de pruebas teniendo en cuenta recursos requeridos, el plan de proyecto, calendario y plazos, definición de los entregables, seguimiento y reporte de defectos.

✓ Diseño casos de prueba

Describe los aspectos generales a tener en cuenta a la hora de realizar el diseño de las pruebas, incidencias en el momento de su desarrollo y otros aspectos relevantes. Define las pruebas diseñadas para ser aplicadas a los casos de uso, especificando las funcionalidades que se desean probar.

Soporte

✓ Plan aseguramiento de la calidad,

Se establecen los estándares, guías y métricas de calidad a seguir por el proyecto. Especifican los documentos utilizados por el Plan de Calidad. Describe los aspectos del plan de requerimientos de proyecto que están alineados con los requerimientos de calidad y la estructura de la organización. Establece el plan de Revisiones y Auditorias las tareas generales, cronograma, organización y responsabilidades, resolución de problemas y actividades de corrección, además de las herramientas, técnicas y metodologías para desarrollar el plan. En esta plantilla se referencia:

- Plan de Pruebas.
- Plan de Resolución de Problema.
- Plan de Gestión de Configuración.

✓ Glosario de términos

Describe las acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de las secuencias. Se definen los conceptos de: Casos de Uso que ayudan a mitigar los riesgos más importantes, Casos de Uso arquitectónicamente significativo, Casos de Uso significativos para el negocio, No Conformidades, Solicitud de Cambios, Prototipo, Prueba de Liberación, Prueba de

Aceptación, Expediente del proyecto y Lineamientos Mínimos, además de otros que sea necesario incluir por el proyecto.

- ✓ Listas de chequeo

Se definen Niveles, Criterios de evaluación, evaluación, observaciones y métricas para chequear los procesos del desarrollo del software.

- ✓ No conformidades (reducida),

Se muestran los resultados obtenidos, describiendo los Elementos Probados y No Probados, explicando las causas que no permitieron realizar las pruebas a estos últimos. Se plantean las no conformidades detectadas y las recomendaciones que se sugieren para la corrección de las mismas.

- ✓ Solicitud de cambio (reducida)

Es la solicitud que realiza el grupo de pruebas, solicitando los cambios pertinentes para erradicar los errores detectados.

- ✓ Respuestas a No Conformidades

Contiene los mismos aspectos que la plantilla de No Conformidades diferenciándose solo en que esta incluye las respuestas de los desarrolladores antes las dificultades detectadas por el grupo de calidad. .

Todo el contenido de estas plantillas presenta información que son específicas e invariables, y que en muchas ocasiones tienden a violarse, precisamente por no existir una forma en la estructura actual que permita la validación de las mismas. Lo mismo sucede con la estructura del documento que por ser su formato editable, está expuesto a cambios que solo podrán ser detectados al ser revisados por un factor externo.

Para la correcta estructuración de las plantillas se hizo necesario determinar el orden en el cual deben ser llenados estos documentos.

3.3 Análisis de la cronología de las plantillas

La cronología se confeccionó teniendo en cuenta las fases de desarrollo del software propuestas por la metodología RUP(Procesos Unificado de Desarrollo del Software), definiendo el momento en que deben ser elaborados cada uno de los documento.

RUP divide el proceso de desarrollo del software en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace mayor o menor hincapié en dependencia de las actividades a desarrollar en cada flujo de trabajo. (65) La figura muestra cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto RUP. (Ver figura 2)

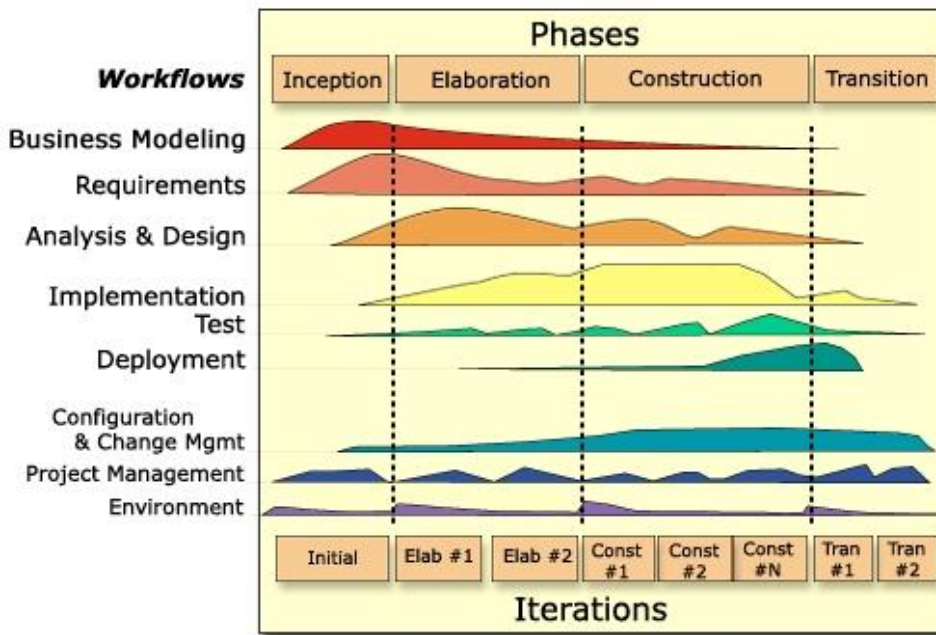


Figura 2. Ciclo de desarrollo del software.

Las primeras iteraciones (en las fases de Inicio y Elaboración) se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto y la eliminación de los riesgos críticos. Durante la fase de inicio las iteraciones hacen mayor énfasis en actividades de modelado del negocio y de requisitos. En la fase de elaboración, las iteraciones se orientan al desarrollo de la línea base de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la arquitectura.

En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones. Para cada iteración se selecciona algunos Casos de Uso, se refina su análisis y diseño y se procede a su implementación y pruebas. Se realiza una pequeña cascada para cada ciclo. Se realizan tantas iteraciones hasta que se termine la implementación de la nueva versión del producto. En la fase de

transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios. (66)

Como se puede observar en cada fase participan todas las disciplinas, pero que dependiendo de la fase el esfuerzo dedicado a una disciplina varía. RUP es un proceso de desarrollo que se caracteriza por la gran cantidad de artefactos generados en las actividades que lo componen. Sin embargo, en RUP se hace énfasis en que no es necesario utilizar todos los artefactos y estos pueden ser seleccionados de acuerdo a las características del proyecto.

Además de tenerse en cuenta el ciclo de desarrollo de RUP para la confección de la cronología, también se consideró como lograr una mayor reutilización de la información, quedando de la siguiente forma:

Cronología

Fase	Flujo de trabajo	Orden Cronológico	Plantilla
Inicio	Modelo del Negocio	1	Documento Visión
		2	Modelo de CUN
		3	Modelo de Dominio
		4	Glosario de términos
	Requerimientos	5	Modelo de Casos de uso del sistema
		6	Plan de gestión de requisito
		7	Especificación de Requisitos
		8	Documento de Arquitectura de Software.
		9	Listas de chequeo
			Glosario de términos (<i>Continuación</i>)
	Gestión de Proyecto	10	Plan Desarrollo de Software
		11	Roles y responsabilidad
		12	Plan de capacitación
		13	Minuta de reunión
14		Plan aseguramiento de la calidad	

Elaboración	Análisis-Diseño		Documento de Arquitectura de Software. (Continuación)
		15	Arquitectura de Información.
		16	Informe del Levantamiento de Información para la Arquitectura de Información.
	Gestión de Proyecto		Roles y responsabilidad (Actualización)
			Plan de capacitación (Actualización)
			Minuta de reunión (Nueva)
	Gestión de Configuración		Solicitud de cambio (reducida)
Construcción	Implementación		Documento de Arquitectura de Software. (Continuación)
	Prueba	17	Plan de pruebas
		18	Diseño casos de prueba
			Listas de chequeo (Nueva)
		19	No conformidades (reducida)
		20	Respuestas a No Conformidades
	Gestión de Proyecto		Roles y responsabilidad (Actualización)
			Plan de capacitación (Actualización)
			Minuta de reunión (Nueva)
	Gestión de Configuración	21	Solicitud de cambio (reducida)

Tabla 1: Cronología de las Plantillas

Las plantillas que presenta la etiqueta de (*Continuación*) son aquellas que después de haber iniciado su elaboración debe seguir elaborándose durante varias fases pero que la información anteriormente registrada no sufre ninguna modificación a diferencia de las etiquetadas como (*Actualización*) que si modifican su información en cada una de las fases donde aparecen. Las clasificadas como (*Nuevas*) son aquellas que a pesar de ser un mismo documento, se realiza en cada caso uno nuevo.

3.4 Clasificación de las plantillas según su composición y tiempo de elaboración

Aunque las plantillas están ordenadas cronológicamente para lograr una mejor organización de la información, esta no es invariable, debido a que existen algunos documentos que se elaboran en varias fases del ciclo de vida del software y otros que referencian a documentos que forman parte de su contenido. Teniendo en cuenta esta situación se clasificaron las plantillas en dos grupos, en dependencia del momento en que son elaborados los documentos y las plantillas a las que hacen referencia. (Ver figura 3)

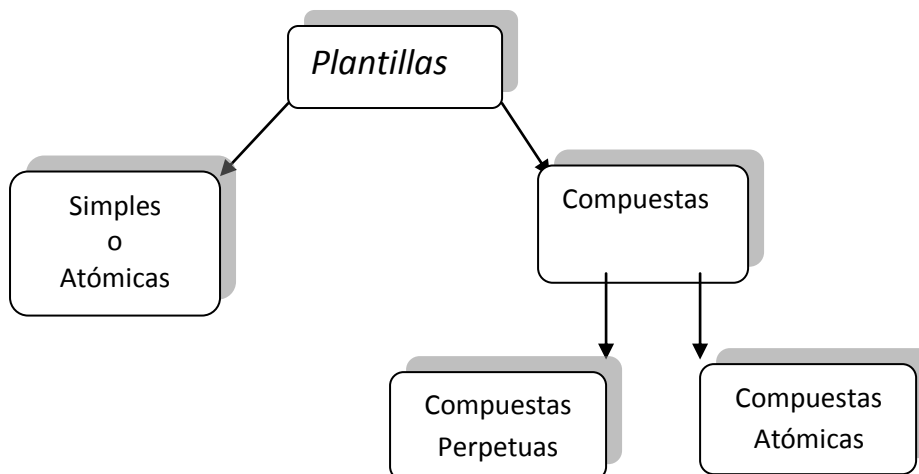


Figura 3: Clasificación según composición y tiempo de elaboración.

- Plantillas Simples o Atómicas: Son elaboradas en cualquier fase y no hacen referencia a ninguna otra plantilla.
- Plantillas compuestas: (que pueden ser de dos tipos)
- Compuestas atómicas: Son elaboradas en una sola fase y hace referencia a una o varias plantillas anteriores.
- Compuestas perpetuas: Son elaboradas en más de una fase y hace referencia a una o varias plantillas.

Plantilla	Clasificación
Documento Visión	Atómica
Modelo de CUN	Atómica
Modelo de Dominio	Atómica
Glosario de términos	Atómica
Modelo de Casos de uso del sistema	Atómica
Plan de gestión de requisito	Compuesta Atómica
Especificación de Requisitos	Atómica
Documento de Arquitectura de Software.	Compuesta Perpetua
Listas de chequeo	Atómica
Arquitectura de Información.	Compuesta Atómica
Plan de pruebas	Atómica
Diseño casos de prueba	Compuesta Atómica
Informe del Levantamiento de Información para la Arquitectura de Información.	Compuesta Atómica
Plan Desarrollo de Software	Compuesta Perpetua
Roles y responsabilidades	Atómica
Minuta de reunión	Atómica
Plan de capacitación	Atómica
Plan aseguramiento de la calidad	Compuesta Perpetua
No conformidades (reducida)	Atómica
Solicitud de cambio (reducida)	Atómica
Respuestas a No Conformidades	Atómica

Tabla 2: Clasificación según la composición y tiempo de elaboración.

Las compuestas atómicas solucionan sus dependencias en el momento que se elaboran a diferencia de las compuestas perpetuas que dependen de la futura realización de otros documentos lo que debe ser considerado en la estructuración del contenido de las plantillas. Además de la reutilización de un documento en su totalidad, también se presenta la reutilización de información específica contenida en una plantilla. Para ello la información fue separada en conceptos, teniendo en cuenta de la naturaleza de la información.

3.5 Naturaleza de la Información

Para lograr una mejor manipulación del contenido de las plantillas y la estructuración del mismo, la información fue tratada como elementos independientes que constituyen una unidad que puede o no ser reutilizable. Estos elementos fueron clasificados en conceptos según la naturaleza la información que contiene

- Concepto estructural: Elemento que representa estructuralmente una parte de la plantilla. Ejemplo: Índice, control de versiones, etc.
- Concepto semántico: Elemento que representa el tipo de dato asociado a un elemento simple.

Estos conceptos serán agrupados por plantillas formando los documentos y permitiendo la reutilización de los conceptos que sean comunes. Lo cual puede dar la posibilidad de que sean generados automáticamente con una herramienta. (Ver figura 4)

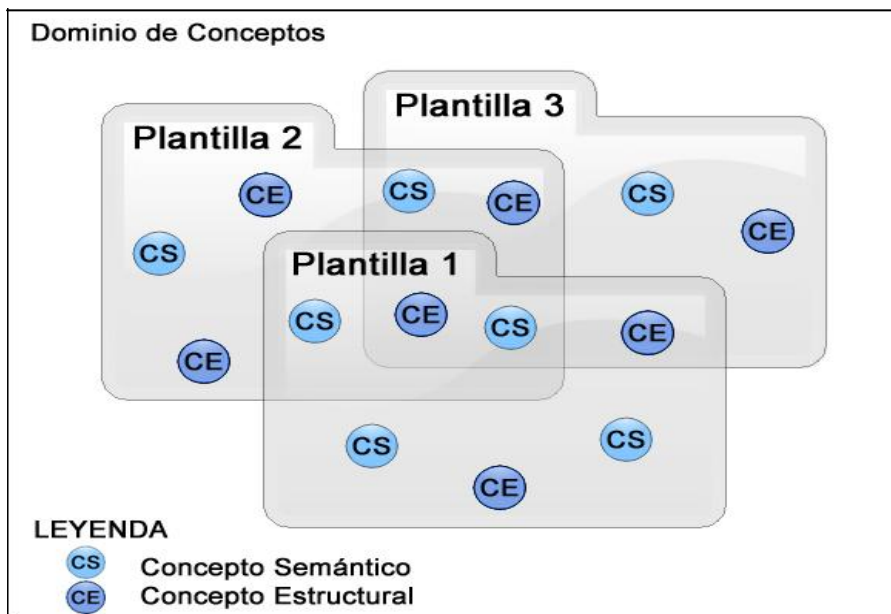


Figura 4 Dominio de conceptos

La estructura debe lograr la mayor independencia de los elementos que contiene las plantillas, agrupándolos según sus características y por su repetición a lo largo de todo el expediente. Esto

posibilita que puedan ser utilizados no sólo en el documento que aparecen, sino en cualquier otro que requiera de esta misma información. El diseño de la estructura del contenido de las plantillas del expediente, a través de los XML Schema debe garantizar estas funcionalidades.

3.6 Diseño de la estructura

El diseño de la estructura se define a través de los XML Schema, los cuales permiten representar elementos, atributos, relaciones y restricciones que se desean especificar. Los XML Schema se confeccionaron teniendo en cuenta los conceptos definidos, agrupándolos de manera que cada esquema representa uno de los documentos que conforman el expediente. Además se confeccionó un esquema donde se representan los tipos de datos definidos para identificar informaciones que contienen las plantillas y que son utilizadas en varias ocasiones.

Los tipos de datos describen la representación, interpretación y la estructura de valores u objetos y pueden ser utilizados para comprobar la validez de los datos que representan estos valores. A partir de este concepto y aprovechando las funcionalidades que ofrece XML Schema se definieron los siguientes tipos de datos:

- ✓ Texto, ImagenType, TablaType: Representan el tipo de información que puede contener un elemento determinado. Pueden aparecer independientes o combinados.
- ✓ DocumentoType: Representa la estructura general que tienen las plantillas
- ✓ dgType: Agrupa los datos generales del proyecto y del producto que se incluye en todos los documentos que conforman el EP.
- ✓ IntroduccionType: Contiene la estructura y los datos de la introducción de los documentos, puede variar en dependencia de las características de los mismos.
- ✓ Reglas_de_confedType: Representa las Reglas de Confidencialidad que aparecen en la mayoría de los documentos, redefiniendo sus valores en cada uno de ellos.
- ✓ SesionType: Representa los capítulos y epígrafes que contiene un documento.

- ✓ roltype: Representa los roles de los miembros del proyecto.
- ✓ fasesType: Representa las fases de desarrollo de software identificadas por RUP.

Estos tipos de datos fueron utilizados en los esquemas que representan a los documentos, identificando los elementos que presentan una misma estructura. Se pueden encontrar dos tipos de casos: primero, el elemento del documento presenta la misma estructura del tipo de datos y adquiere un mismo valor para todos los documentos donde es utilizado. Segundo: el elemento del documento presenta la misma estructura del tipo de dato y sus valores se redefinen según las características propias de cada documento. Estos elementos se relacionaron entre sí, estableciendo restricciones que los identifican, y que hacen válido su contenido. Finalmente se desarrollaron 21 esquemas, uno por cada documento o plantilla del EPE.

En todo momento se trató de respetar al máximo la estructura actual de las plantillas propuestas por la Dirección de Calidad de Software (DCS) ya que fue uno de los requisitos planteados. Se realizaron los cambios pertinentes para lograr la adecuada estructuración según los modelos de programación. También fue introducida información de control adicional válida para identificar unívocamente los documentos de un mismo expediente de proyecto. Estos no provocan ninguna afectación en el objetivo de los documentos, más allá de ello extiende la funcionalidad de estos brindando potencial para la autenticación y validez de los mismos. Debido al carácter flexible de XML Schema se pudieran extender estos documentos para brindar opciones de firma digital y otras, cualidades que están fuera del dominio del presente trabajo.

La relación que existe entre el documento a estructurar y su representación a través del lenguaje XML Schema es fácil de comprender, dada la legibilidad que proporcionan los lenguajes basados en XML. (Ver figura 5,6 y 7)


	<h2>Documento Visión</h2>
<h3>3.5 Perfil de los usuarios</h3>	
<h4>3.5.1 Planificadores</h4>	
Descripción	Son los planificadores en todos los niveles de dirección del MINSAP.
Tipo	Planificador
Responsabilidades	Encargado en cada nivel de dirección del MINSAP de realizar la planificación de materiales gastables de uso medico.
Criterio de éxito	Alto
Implicación	Usuario
Entregable	Requiere un Manual de Usuario y Ayuda que presente el sistema.
<h3>3.6 Principales necesidades de los Involucrados y los usuarios</h3>	
<p>Para poder elaborar la demanda de los recursos materiales de uso médico que se utiliza en un año en el país los planificadores de materiales que pertenecen al Área de Economía del Ministerio de Salud Pública tienen que recorrer todas las cabeceras de provincias, allí obtienen de los jefes de las direcciones provinciales de salud un listado con el tipo y la cantidad de materiales de uso médico que necesitará esa provincia para cubrir las necesidades de sus unidades asistenciales de salud, de la suma de los listados de las demandas de las provincias se confeccionará las necesidades de productos de uso médico para el país.</p>	

Figura 5: Fragmento del Documento Visión actual.

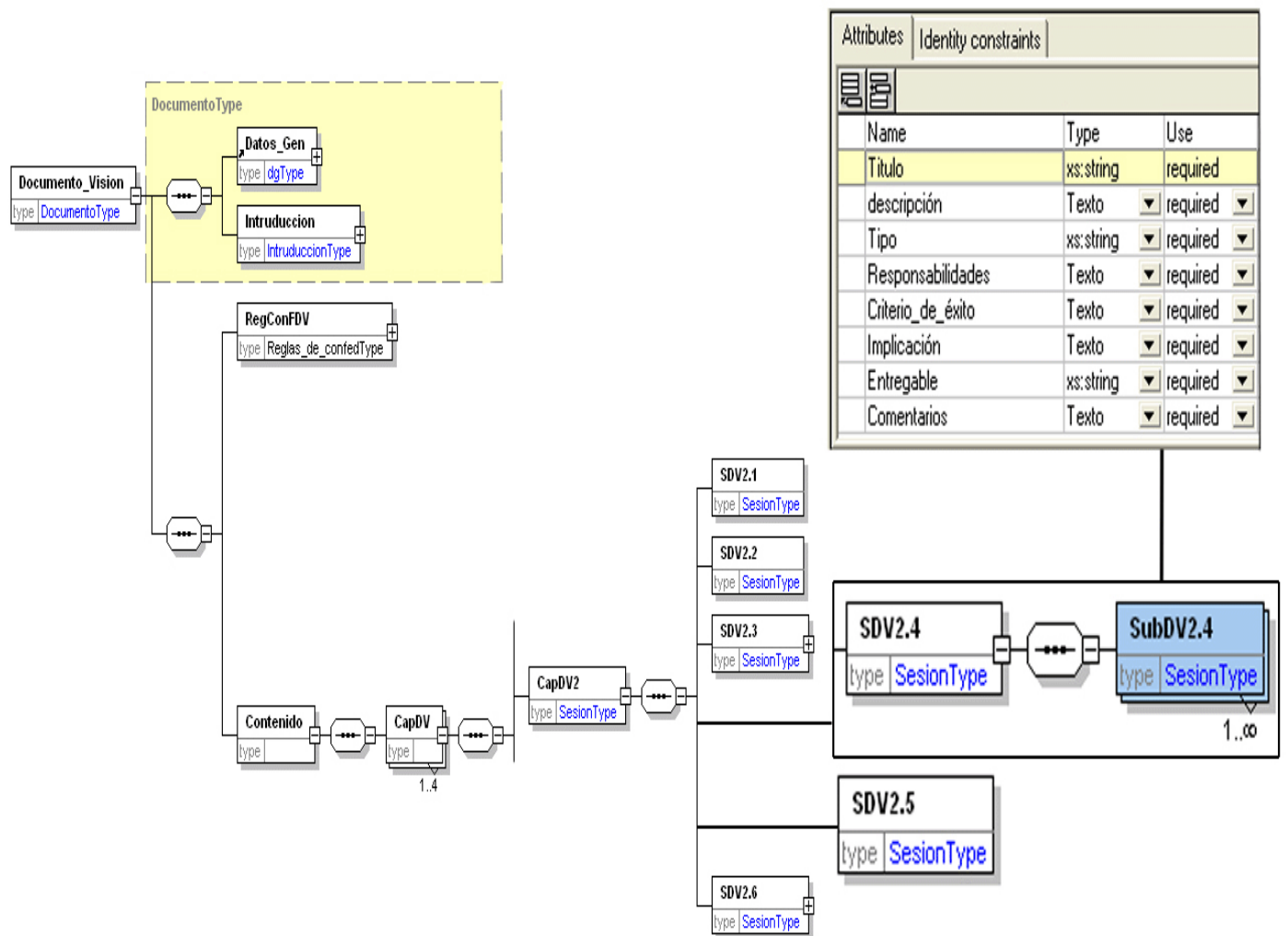


Figura 6: Estructura en XML Schema del fragmento del Documento Visión correspondiente a la figura 5.

element **Documento_Vision/Contenido/CapDV/CapDV2/SDV2.4**

diagram													
type	extension of SesionType												
children	SubDV2.4												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Titulo</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Titulo	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Titulo	xs:string	required											
source	<pre> <xs:element name="SDV2.4"> <xs:complexType> <xs:complexContent> <xs:extension base="SesionType"> <xs:sequence> <xs:element name="SubDV2.4" maxOccurs="unbounded"> <xs:complexType> <xs:complexContent> <xs:extension base="SesionType"> <xs:attribute name="descripci3n" type="Texto" use="required"/> <xs:attribute name="Tipo" type="xs:string" use="required"/> <xs:attribute name="Responsabilidades" type="Texto" use="required"/> <xs:attribute name="Criterio_de_3xito" type="Texto" use="required"/> <xs:attribute name="Implicaci3n" type="Texto" use="required"/> <xs:attribute name="Entregable" type="xs:string" use="required"/> <xs:attribute name="Comentarios" type="Texto" use="required"/> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element> </pre>												

Figura 7: fragmento de código del XML Schema Documento Visión que correspondiente a la figura 6.

3.7 Beneficios de la estructura.

Entre los principales beneficios que trae consigo la utilización de la estructura diseñada están:

- Validación automática de los documentos.
- Recuperación y reutilización automática de datos a partir de documentos previos eliminando la repetición innecesaria en el llenado de nuevos documentos del mismo expediente.
- Flexibilidad con potencial extensión y mejora sin caer en incompatibilidades hacia atrás.
- Potencial estandarización para la normalización de la creación e intercambio de documentos del expediente de proyecto.

En este capítulo se realizó un análisis de la composición actual del EP y la información contenida en el mismo. Se determinaron un grupo de clasificaciones para cada una de las plantillas. Además se conformó un expediente de proyecto básico, para la documentación de las soluciones UCI basadas en la metodología RUP. Se realizó el diseño de la estructura de las plantillas a través de XML Schema.

CONCLUSIONES

Con el desarrollo de este trabajo, se profundizó en el conocimiento de las técnicas y herramientas de Gestión Documental y se analizó el proceso de documentación del desarrollo de software, enfatizando en el Expediente de Proyecto (EP) utilizado en la Universidad de las Ciencias Informáticas.

Se estudiaron las principales tendencias para el manejo automatizado de la información, realizando un estudio de las tecnologías que permiten el mismo.

Se realizó el análisis y diseño de una estructura para el contenido de las plantillas del Expediente de proyecto, utilizando el lenguaje de estructura XML Schema que facilita la automatización de la realización del EP, a través de una herramienta.

Se puede concluir que se ha cumplido satisfactoriamente el objetivo trazado para este trabajo obteniendo los siguientes resultados:

- Análisis de las técnicas de Gestión Documental.
- Un expediente de proyecto simplificado que permite satisfacer las necesidades básicas de la documentación de los proyectos productivos de la Universidad nombrado Expediente de proyecto Elemental.
- Una cronología para la elaboración del Expediente de Proyecto Elemental.
- Diseño de la estructura del contenido del Expediente de Proyecto.
- Obtención de una estructura que sirve de base para la informatización de la gestión del Expediente de Proyecto.

RECOMENDACIONES

A continuación se mencionan algunas recomendaciones con el objetivo de realizar un seguimiento y mejora de esta investigación:

- Precisar las restricciones del contenido del EP, para mejorar la validación de los documentos perfeccionando la definición de elementos y con ello fortaleciendo el tipeo de datos en los esquemas que estructuran las plantillas.
- Incorporar, en la medida que sean necesarias, nuevas plantillas para mejorar y completar el Expediente de Proyecto.
- Implementar una aplicación sobre la base del desarrollo teórico expuesto que haga uso extensivo de las propuestas, logrando el mayor nivel de automatización en el uso del Expediente de Proyecto.
- Poner en práctica la metodología propuesta para el llenado del expediente de proyecto por fases de desarrollo.

REFERENCIAS BIBLIOGRÁFICAS

1. Gestión Documental. [En línea] 2005.
http://www.guellconsulting.com/servicios/internet/gestion_documental.
2. *Monográfico Calidad del Software / Software de calidad*. Número 137, s.l. : NOVATICA, Enero-Febrero 2004.
3. **Lloret**. *Gestión de la información vs. gestión del conocimiento*. 2004.
4. **Ishikawa, K.** *¿Qué es el control total de la calidad?* s.l. : La Habana: Ciencias Sociales, 2000.
5. **Ruesta, C.B.** Gestión Documental en las empresas: Una aproximación práctica. [En línea] 2000.
<http://www.inforarea.es/Documentos/fesabid.pdf>.
6. *Gestión Documental*. s.l. : e-doctimo: 4.
7. **Sagi, D.J.B.** *Introducción a la Gestión del Conocimiento, Gestión Documental y CRM*.
8. **Pressman, R.S.** *Ingeniería del Software: Un enfoque práctico*. 2002.
9. *Los sistemas de gestión integral de la documentación en las organizaciones. Métodos de Información*. **Allepuz Ros, Teresa y Gutierrez la Rubia, Carme**. 5, mayo 2001, Vol. 2, págs. 29-33.
10. **Gracia, Joaquin**. CMM - CMMI. [En línea] 2005. <http://www.ingenierosoftware.com/calidad/cmm-cmmi.php>.
11. —. CMM - CMMI Nivel 2. [En línea] 26 de Noviembre de 2005.
<http://www.ingenierosoftware.com/calidad/cmm-cmmi-nivel-2.php>.
12. Sistema integrado de gestion documental GIT-DOC: Nuestro enfoque. [En línea] 2007.
http://www.gitdoc.com/ctl_arch/image/infotecnica.pdf.
13. **Bárceñas, A.A.** *III Jornada de gerencia de proyectos*. [En línea] 2005.
<http://www.acis.org.co/index.php?id=357>.
14. *Project Description*. [En línea] 2006. <http://plone.org/products/plone>.
15. *New Summer Mambo templates released*. [En línea] 2004. <http://www.mamboserver.com>.
16. *Drupal 5.1 and 4.7.6 released*. [En línea] 2007. <http://drupal.org>.
17. *Welcome to the OpenCms Project*. [En línea] 2006. <http://www.opencms.org/opencms/en>.
18. *Trabajo documental y colaborativo*. [En línea] 2006.
http://www.csi.map.es/csi/tecniMAP/tecniMAP_2006/02T_PDF/trabajo%20documental.pdf.
19. *Idem referencia a 12*. [En línea]
20. *DME -Document Management Extension*. [En línea] 2007.
http://www.bs.com.ar/bsweb/MENU/SUBMENU/Archivos/bs_053.pdf.
21. **Collins- Sussman, B.** *Control de versiones con Subversion*. 2004. pág. 286.
22. **Eíto Brun, Ricardo**. Documentum lanza Edms'98. [En línea] 2000. <http://www.documentum.com>.

23. **García Caballero, Ricardo.** *Gestión de los documentos electrónicos en los nuevos servicios de información y documentación.* [ed.] Bonifacio Martín Galán. Universidad Carlos III de Madrid : Everest.
24. *Idem referencia a 23.*
25. *Idem referencia a 23.*
26. *Idem referencia a 23.*
27. **Hernández Orallo, José, Ramírez Quintana, M.José y Ferri Ramírez, Cèsar.** *Introducción a la Minería de Datos.* s.l. : Pearson, 2004. ISBN: 84 205 4091 9.
28. *Idem referencia a 27.*
29. **del Río Portilla, Jesús Antonio.** *Para qué sirve la minería de textos.* [En línea] 2006. <http://www.cie.unam.mx/~arp/>.
30. *Combine minería de texto con minería de datos para transformar la información que posea en business intelligence.* [En línea] 2007. <http://www.spss.com/es/pdf/textmining.pdf>.
31. *La web semántica.* **Castells, Pablo.** Escuela Politécnica Superior Universidad Autónoma de Madrid : s.n.
32. **Berrocal, Alonso, García-Figuerola, José Luis y Zazo Rodríguez, Carlos.** SGML/XML: Desarrollo en entornos documentales. [En línea] 2004. <http://wotan.liu.edu/doi/data/julmnbkd.html>.
33. **López Quijado, José.** *Domine HTML y DHTML.* s.l. : Ra-ma, 2006. ISBN: 8478975357.
34. **Santamaría Gallo, Abelardo.** *Codificación de instrumentos de descripción archivísticos con SGML/XML: La norma EAD v.1.0.* s.l. : Tabula, 2005. págs. 69-125.
35. **Rosa Piñero, Antonio de la.** *Entornos documentales WWW: entorno XML. 2ª Jornadas Andaluzas de Documentación.* 2da. 2004. págs. 357-372.
36. **Bray, Tim, Jean Paoli, C. M y Sperberg-McQueen.** *Extensible Markup Language (XML) 1.0 - El lenguaje extensible de marcas (XML) 1.0.* [trad.] Fabio Arciniegas A. 2007.
37. **Méndez Rodríguez, Eva Mª.** *RDF: un modelo de metadatos flexible para las bibliotecas digitales.* s.l. : Jornades Catalanes de Documentació, 2002. págs. 487-498.
38. *Idem referencia a 37.*
39. **McGuinness, Deborah L. y Van Harmelen, Frank.** *WL Web Ontology Language Overview.* [En línea] 2004. <http://www.w3.org/TR/2004/REC-owl-features-20040210/>.
40. *DTD's y Schemas, cual es su función ? .* [En línea] 2005. <http://www.osmosislatina.com/xml/schemas.htm>.

41. *Análisis de las posibilidades del XML como una herramienta para el desarrollo de un SFBD (loosely coupled) de datos estructurados y semiestructurados*. **Morocho Zurita, Villie**. Barcelona : Departamento de Lenguajes y Sistemas Informáticos, Universidad Politécnica de Cataluña, 2006.
42. "XML Schema Requirements". W3C. **Maloney, Murray**. [ed.] Ashok Malhotra. 2003.
43. "Document Content Description for XML". **Bray, Tim, Frankston, Charles y Malhotra, Ashok**. s.l. : Paerson, 2003.
44. **Davidson, Andrew**. "Schema for Object-Oriented XML". [En línea] 2006.
<http://www.w3.org/TR/NOTE-SOX/>.
45. **Klarlund, Nils, Møller, Anders y Schwartzbach, Michael I**. "Document Structure Description 1.0". [En línea] 2004. <http://www.brics.dk/DSD/dsddoc.html>.
46. **Murata, Makoto**. "RELAX REgular LAnguage description for XML". [En línea] 2002.
<http://www.xml.gr.jp/relax/>.
47. **Jelliffe, Rick**. "The Schematron Assertion Language 1.5". [En línea] octubre de 2002.
(<http://www.ascc.net/xml/resource/schematron/Schematron2000.html>).
48. Idem referencia a 42. [En línea]
49. **Jürgens, B**. Sistemas de gestión de contenidos (CMS) para la creación de un Portal Web. [En línea] 2005.
<http://72.14.203.104/search?q=cache:d6qTxa18ZRcJ:www.ugr.es/~victorhs/DyGSI/archivos/cms.pdf+CMS+comparacion&hl=es&gl=cu&ct=clnk&cd=26>.
50. *SISTEMAS DE BASE DE DATOS*. [En línea] <http://usuarios.lycos.es/cursosgbd/UD2.htm>..
51. *Sistema para el registro y control de prerreclutas en los comités militares*. **Saumell, M.G. y Duvergel, K.H**. s.l. : Instituto Superior Politécnico "José Antonio Echeverría", 2006.
52. **AB, MySQL**. Tutorial básico de MySQL. [En línea] 26 de Marzo de 2003 . <http://www.mysql-hispano.org/page.php?id=6>.
53. **Díaz, W**. Manual SQL Server. [En línea] <http://walter.freesevers.com/contsql.html>..
54. Oracle. [En línea] <http://es.wikipedia.org/wiki/Oracle>..
55. Idem referencia 42. [En línea]
56. **Clark, James**. "XSL Transformations (XSLT) Version 1.0". [En línea] 2005.
<http://www.w3.org/TR/2005/REC-xslt-19991116>.
57. "Oracle XSQL Pages and the XSQL Servlet". [En línea] noviembre de 2002.
(http://technet.oracle.com//tech/xml/xsql_servlet/htdocs/reInotes.htm).
58. **Ceri, S., y otros**. "XML-GL: a Graphical Language for Querying and Restructuring XML Documents". [En línea] 2004. <http://www8.org/w8-papers/1c-xml/xml-gl/xml-gl.html>.

59. **Bray, Tim, Frankston, Charles y Malhotra, Ashok.** "Document Content Description for XML". [En línea] 2006. <http://www.w3.org/TR/NOTE-dcd>.
60. **Adler, Sharon.** "Extensible Stylesheet Language (XSL) Version 1.0". [En línea] 2004. <http://www.w3.org/TR/xsl/>.
61. **Armstrong, Eric.** "Working with XML". [En línea] 2003. <http://java.sun.com/xml/docs/tutorial/overview/index.html>.
62. *Informe sobre de XMLSPY.* **Poggi, Fabrizio.** 2, Roma : Enea, 2007.
63. **Garzás, Javier.** Metodologías, buenas prácticas y estrategias. *kybele consulting* . [En línea] 22 de julio de 2007. <http://kybeleconsulting.blogspot.com/2007/07/metodologas-buenas-prcticas-y.html>.
64. **Team, CMMI Product.** *CMMISM for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing (CMMI-SE/SW/PPD/SS, V1.1).* s.l. : Carnegie Mellon University, 2002. 252 227 7013.
65. **Software, Rational.** *Rational Unified Process Best Practices for Software Development Teams.* 2003.
66. *Idem referencia a 65.*
67. "Document Content Description for XML". **Bray, Tim, Frankston, Charles y Malhotra, Ashok.** s.l. : Paerson, 2006.
68. —. **Bray, Tim, Frankston, Charles y Malhotra, Ashok.** 2003.
69. **Díaz, W.** Manual SQL Server. [En línea] <http://walter.freesevers.com/contsql.html>.

BIBLIOGRAFÍA

- **Adler, Sharon.** “Extensible Stylesheet Language (XSL) Version 1.0”. [En línea] 2004. <http://www.w3.org/TR/xsl/>.
- **Armstrong, Eric.** “Working with XML”. [En línea] 2003. <http://java.sun.com/xml/docs/tutorial/overview/index.html>.
- **AB, MySQL.** Tutorial básico de MySQL. [En línea] 26 de Marzo de 2003 . <http://www.mysql-hispano.org/page.php?id=6>.
- *Análisis de las posibilidades del XML como una herramienta para el desarrollo de un SFB (loosely coupled) de datos estructurados y semiestructurados.* **Morocho Zurita, Villie.** Barcelona : Departamento de Lenguajes y Sistemas Informáticos, Universidad Politécnica de Cataluña, 2006.
- **Bárceñas, A.A.** *III Jornada de gerencia de proyectos.* [En línea] 2005. <http://www.acis.org.co/index.php?id=357>.
- **Berrocal, Alonso, García-Figuerola, José Luis y Zazo Rodríguez, Carlos.** SGML/XML: Desarrollo en entornos documentales. [En línea] 2004. <http://wotan.liu.edu/doi/data/julmnbkd.html>.
- **Bray, Tim, Jean Paoli, C. M y Sperberg-McQueen.** *Extensible Markup Language (XML) 1.0 - El lenguaje extensible de marcas (XML) 1.0.* [trad.] Fabio Arciniegas A. 2007.
- **Bray, Tim, Frankston, Charles y Malhotra, Ashok.** “Document Content Description for XML”. [En línea] 2006. <http://www.w3.org/TR/NOTE-dcd>
- CMM - CMMI Nivel 2. [En línea] 26 de Noviembre de 2005. <http://www.ingenierosoftware.com/calidad/cmm-cmmi-nivel-2.php>.
- **Collins- Sussman, B.** *Control de versiones con Subversion.* 2004. pág. 286.
- *Combine minería de texto con minería de datos para transformar la información que posea en business intelligence.* [En línea] 2007. <http://www.spss.com/es/pdf/textmining.pdf>.
- **Clark, James.** “XSL Transformations (XSLT) Version 1.0”. [En línea] 2005. <http://www.w3.org/TR/2005/REC-xslt-19991116>.
- **Ceri, S., y otros.** “XML-GL: a Graphical Language for Querying and Restructuring XML Documents”. [En línea] 2004. <http://www8.org/w8-papers/1c-xml/xml-gl/xml-gl.html>.
- **Díaz, W.** Manual SQL Server. [En línea] <http://walter.freeservers.com/contsql.html>.

- [Döm98] Dömges, R., Pohl, K., Adapting Traceability Enviroments to Project-Specific Needs. Communications of ACM, Vol 41, No 21, December 1998
- “*Document Content Description for XML*”. **Bray, Tim, Frankston, Charles y Malhotra, Ashok.** s.l. : Paerson, 2006.
- **Díaz, W.** Manual SQL Server. [En línea] <http://walter.freesevers.com/contsql.html..>
- **Davidson, Andrew.** “Schema for Object-Oriented XML”. [En línea] 2006. <http://www.w3.org/TR/NOTE-SOX/>.
- *DTD's y Schemas, cual es su función ?*. [En línea] 2005. <http://www.osmosislatina.com/xml/schemas.htm>.
- “*Document Content Description for XML*”. **Bray, Tim, Frankston, Charles y Malhotra, Ashok.** s.l. : Paerson, 2003.
- *Drupal 5.1 and 4.7.6 released.* [En línea] 2007. <http://drupal.org>.
- *DME -Document Management Extension.* [En línea] 2007. http://www.bs.com.ar/bsweb/MENU/SUBMENU/Archivos/bs_053.pdf.
- **del Río Portilla, Jesús Antonio.** *Para qué sirve la minería de textos.* [En línea] 2006. <http://www.cie.unam.mx/~arp/>.
- **Eíto Brun, Ricardo.** Documentum lanza Edms'98. [En línea] 2000. <http://www.documentum.com>.
- **Garzás, Javier.** Metodologías, buenas prácticas y estrategias. *kybele consulting* . [En línea] 22 de julio de 2007. <http://kybeleconsulting.blogspot.com/2007/07/metodologas-buenas-prcticas-y.html>.
- **García Caballero, Ricardo.** *Gestión de los documentos electrónicos en los nuevos servicios de información y documentación.* [ed.] Bonifacio Martín Galán. Universidad Carlos III de Madrid : Everest.
- **Gracia, Joaquin.** CMM - CMMI. [En línea] 2005. <http://www.ingenierossoftware.com/calidad/cmm-cmmi.php..>
- Gestión Documental. [En línea] 2005. http://www.guellconsulting.com/servicios/internet/gestion_documental.
- *Gestión Documental.* s.l. : e-doctimo: 4.
- **Hernández Orallo, José, Ramírez Quintana, M.José y Ferri Ramírez, Cèsar.** *Introduccion a la Minería de Datos.* s.l. : Pearson, 2004. ISBN: 84 205 4091 9.
- **Ishikawa, K.** *¿Qué es el control total de la calidad?* s.l. : La Habana: Ciencias Sociales, 2000.
- *Informe sobre de XMLSPY.* **Poggi, Fabrizio.** 2, Roma : Enea, 2007.

- **Jelliffe, Rick.** “*The Schematron Assertion Language 1.5*”. [En línea] octubre de 2002. (<http://www.ascc.net/xml/resource/schematron/Schematron2000.html>).
- **Jürgens, B.** Sistemas de gestión de contenidos (CMS) para la creación de un Portal Web. [En línea] 2005. <http://72.14.203.104/search?q=cache:d6qTxa18ZRcJ:www.ugr.es/~victorhs/DyGSI/archivos/cms.pdf+CMS+comparacion&hl=es&gl=cu&ct=clnk&cd=26>.
- **Klarlund, Nils, Møller, Anders y Schwartzbach, Michael I.** “*Document Structure Description 1.0*”. [En línea] 2004. <http://www.brics.dk/DSD/dsddoc.html>.
- **López Quijado, José.** *Domine HTML y DHTML*. s.l. : Ra-ma, 2006. ISBN: 8478975357.
- *Los sistemas de gestión integral de la documentación en las organizaciones. Métodos de Información.* **Allepuz Ros, Teresa y Gutierrez la Rubia, Carme.** 5, mayo 2001, Vol. 2, págs. 29-33.
- *La web semántica.* **Castells, Pablo.** Escuela Politécnica Superior Universidad Autónoma de Madrid : s.n.
- **Lloret.** *Gestión de la información vs. gestión del conocimiento.* 2004.
- *Monográfico Calidad del Software / Software de calidad.* Número 137, s.l. : NOVATICA, Enero-Febrero 2004.
- **Méndez Rodríguez, Eva M^a.** *RDF: un modelo de metadatos flexible para las bibliotecas digitales.* s.l. : Jornades Catalanes de Documentació, 2002. págs. 487-498.
- **McGuinness, Deborah L. y Van Harmelen, Frank.** *WL Web Ontology Language Overview.* [En línea] 2004. <http://www.w3.org/TR/2004/REC-owl-features-20040210/>.
- **Murata, Makoto.** “*RELAX REgular LAnguage description for XML*”. [En línea] 2002. <http://www.xml.gr.jp/relax/>.
- *New Summer Mambo templates released.* [En línea] 2004. <http://www.mamboserver.com>.
- **Pressman, R.S.** *Ingeniería del Software: Un enfoque práctico.* 2002.
- *Project Description.* [En línea] 2006. <http://plone.org/products/plone>.
- **Ruesta, C.B.** *Gestión Documental en las empresas: Una aproximación práctica.* [En línea] 2000. <http://www.inforarea.es/Documentos/fesabid.pdf>.
- **Rosa Piñero, Antonio de la.** *Entornos documentales WWW: entorno XML. 2ª Jornadas Andaluzas de Documentación.* 2da. 2004. págs. 357-372.
- **Sagi, D.J.B.** *Introducción a la Gestión del Conocimiento, Gestión Documental y CRM.*

- Sistema integrado de gestión documental GIT-DOC: Nuestro enfoque. [En línea] 2007. http://www.gitdoc.com/ctl_arch/image/infotecnica.pdf.
- **Santamaría Gallo, Abelardo.** *Codificación de instrumentos de descripción archivísticos con SGML/XML: La norma EAD v.1.0.* s.l. : Tabula, 2005. págs. 69-125.
- *SISTEMAS DE BASE DE DATOS.* [En línea] <http://usuarios.lycos.es/cursosgbd/UD2.htm>.
- *Sistema para el registro y control de prerreclutas en los comités militares.* **Saumell, M.G. y Duvergel, K.H.** s.l. : Instituto Superior Politécnico "José Antonio Echeverría", 2006.
- **Software, Rational.** *Rational Unified Process Best Practices for Software Development Teams.* 2003.
- *Trabajo documental y colaborativo.* [En línea] 2006. http://www.csi.map.es/csi/tecniap/tecniap_2006/02T_PDF/trabajo%20documental.pdf.
- **Team, CMMI Product.** *CMMISM for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing (CMMI-SE/SW/IPPD/SS, V1.1).* s.l. : Carnegie Mellon University, 2002. 252 227 7013.
- *Welcome to the OpenCms Project.* [En línea] 2006. <http://www.opencms.org/opencms/en>.
- "XML Schema Requirements". W3C. **Maloney, Murray.** [ed.] Ashok Malhotra. 2003.

ANEXOS

Los anexos se adjuntan en formato digital en un dispositivo de almacenamiento, por ser demasiado extenso su contenido imposibilitando su impresión.

GLOSARIO DE TERMINOS

1. **Calidad de Software:** La Calidad del Software se define como: La concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados, y con las características implícitas que se esperan de todo software desarrollado profesionalmente.
2. **Metodología de software:** se encarga de elaborar estrategias de desarrollo de software que promuevan prácticas para adoptar en lugar de predecir, centradas en las personas o los equipos, orientadas hacia la funcionalidad y la entrega, de comunicación intensiva y que requieren implicación directa del cliente.
3. **Expediente del proyecto:** Contiene la información técnica generada durante el desarrollo del proyecto y respalda la correcta aplicación de los recursos del proyecto. Su integración, actualización y resguardo son obligatorios para los responsables técnico y administrativo del proyecto
4. **Artefactos:** Resultado parcial o final que es producido y usado durante el proyecto. Son las entradas y salidas de las actividades
5. **Groupware:** Software colaborativo, se refiere al conjunto de programas informáticos que integran el trabajo en un sólo proyecto con muchos usuarios concurrentes que se encuentran en diversas estaciones de trabajo, conectadas a través de una red.
6. **Dossieres:** Expediente
7. **Spool:** Viene de Spooling que es el proceso de transferir datos poniéndolos en un área de trabajo temporal, a la cual tiene acceso otro programa para procesarla más tarde. También se puede referir a un dispositivo de almacenamiento que incorpora un spool físico, como una unidad de cinta. El spool más común es el de impresión.

8. **Workflow: Flujo de trabajo** (en español) es el estudio de los aspectos operacionales de una actividad de trabajo: cómo se estructuran las tareas, cómo se realizan, cuál es su orden correlativo, cómo se sincronizan, cómo fluye la información que soporta las tareas y cómo se le hace seguimiento al cumplimiento de las tareas.
9. **API:** Del inglés **A**pplication **P**rogramming **I**nterface - Interfaz de Programación de Aplicaciones es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
10. **Editor WYSIWYG:** Es el acrónimo de **W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **G**et "lo que ves es lo que obtienes". Se aplica a los procesadores de texto y otros editores de texto con formato (como los editores de HTML) que permiten escribir un documento viendo directamente el resultado final, frecuentemente el resultado impreso.
11. **JSP: Java Server Pages** es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.
12. **Data Mining** Minería de datos consiste en la extracción no-trivial de información que reside de manera implícita en los datos. Dicha información era previamente desconocida y podrá resultar útil para algún proceso
13. **Data Mining Group:** Grupo de Minería de Datos.
14. **World Wide Web (WWW):** Es una Red Global Mundial es un sistema de documentos de hipertexto y/o hipermedios enlazados y accesibles a través de Internet. Con un navegador Web, un usuario visualiza páginas web que pueden contener texto, imágenes, vídeos u otros contenidos multimedia, y navega a través de ellas usando hiperenlaces.
15. **XML:** Sigla en inglés de **E**xtensible **M**arkup **L**anguage, lenguaje de marcas extensible,
16. **XML Schema:** Esquema XML

17. **RDF:** Siglas de **R**esource **D**escription **F**ramework- Marco de Descripción de Recursos
18. **RDF Schema:** Esquemas RD.
19. **OWL:** Es el acrónimo del inglés **O**ntology **W**eb **L**anguage, un lenguaje de marcado para publicar y compartir datos usando ontologías en la WWW.
20. **HTML:** Sigla de Hypertext Markup Language-Lenguaje Etiquetas de Hipertexto
21. **Ontología:** El término ontología en informática hace referencia a la formulación de un exhaustivo y riguroso esquema conceptual dentro de un dominio dado, con la finalidad de facilitar la comunicación y la compartición de la información entre diferentes sistemas.
22. **Thesaurus:** Lenguajes de indización que pueden aparecer de dos formas; los que usan términos o palabras del lenguaje natural (tesauros, sistemas de encabezamiento de materias. Los que utilizan símbolos: números, letras o la combinación de ambos (clasificación bibliográfica).
23. **GUI:** Guía de Interfaz de Usuarios
24. **W3C.** El **W**orld **W**ide **W**eb **C**onsortium, es un consorcio internacional que produce estándares para la WWW.
25. **Framework:** Es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, un *framework* puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Un *framework* representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.
26. **Namespace:** Espacios de Nombre, simplemente es un conjunto de nombres en el cual todos los nombres son únicos

27. **Constraints:** Clausuras o reglas para asegurar la integridad referencial de los datos
28. **Plug-ins:** Son programas que expanden las características de programas principales como el browser y le agregan capacidades.
29. **TeraBytes:** Es una unidad de medida informática cuyo símbolo es el **TB**, y puede equivalerse a 2^{40} bytes o a 10^{12} bytes. Viene del griego τέρας, que significa monstruo.
30. **Front-End:** En sintetizadores del habla se refiere a la parte del sistema que convierte la entrada del texto en una representación simbólico-fonética
31. **DB2:** Es una marca comercial, propiedad de IBM, bajo la cual se comercializa el sistema de gestión de base de datos

