



Universidad de las Ciencias Informáticas

Facultad 10



Título: Solución de integración Intranet2 – Alfresco

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

Autora: Irina Díaz Torres

Tutores: Ing. Roberkys Martín Cruaños

Ing. Indira Peraza Sierra

Ciudad de La Habana, 2010

“Año 52 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconozco 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 ____ días del mes de _____ del año_____.

Firma del Autor

Irina Díaz Torres

Firma del Tutor

Ing. Roberkys Martín Cruaños

Firma de la Tutora

Ing. Indira Peraza Sierra

Resumen

La UCI (Universidad de las Ciencias Informáticas), por sus características, se ha convertido en una ciudad digital, en la cual el conocimiento de las diferentes informaciones y el uso de diversos servicios son de vital importancia para el desarrollo de las actividades cotidianas. Para posibilitar todo lo anterior posee una *Intranet* como medio de información para los estudiantes y trabajadores del centro.

La *Intranet2* es una nueva versión de la *Intranet*, la cual está actualmente en proceso de explotación y pretende ofrecer una integración de servicios para el confort de sus usuarios.

Este trabajo consiste en el análisis, diseño e implementación del módulo IDAM (*Integration Drupal-Alfresco Module*) el cual será incorporado a la *Intranet2* como un nuevo servicio que permitirá que los usuarios puedan, desde la misma, tener acceso a los documentos de su espacio personal almacenados en el Gestor de Contenido Empresarial *Alfresco* y consentirá además navegar por la estructura de carpetas de su repositorio de documentos.

Para su realización se utilizó el Sistema de Administración de Contenidos *Drupal* sobre el cual se encuentra desarrollada la *Intranet2*. Se utilizaron las herramientas y tecnologías adecuadas para la creación y administración de la información en un entorno *Web*.

Palabras claves: *Alfresco*, documentos, *Drupal*, integración, *Intranet2*.

Índice

Índice

Introducción:	1
Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales	6
1.1 Desarrollo Web	6
1.2 Sistemas de Gestión de Contenidos	7
1.2.1 CMS Drupal.....	8
1.3 Gestores de Contenido Empresarial	8
1.3.1 ECM Alfresco.....	9
1.4 Módulos de Integración Drupal–Alfresco	9
1.5 Tecnologías, Metodología y Herramientas	11
1.5.1 HTML	11
1.5.1.1 XHTML 1.0	11
1.5.2 CSS.....	12
1.5.3 Javascript.....	13
1.5.3.1 jQuery.....	14
1.5.4 Lenguajes Dinámicos	14
1.5.4.1 PHP.....	15
1.5.5 Sistemas de Gestión de Base de Datos.....	16
1.5.5.1 PostgreSQL	16
1.5.5.2 MySQL	16
1.5.6 Servidor Web. Apache.....	17
1.5.7 Metodologías de Desarrollo.....	18
1.5.7.1 XP	19
1.5.7.2 FDD	19
1.5.7.3 RUP	20
1.5.8 Herramientas de desarrollo	22
1.5.8.1 Netbeans.....	22
1.5.8.2 Visual Paradigm.....	22
1.5.9 Propuesta de solución.....	23
Capítulo 2: Características de la solución de integración	25
2.1 Problema y situación problemática	25
2.2 Objeto de automatización	26
2.3 Información que se maneja	26

Índice

2.4	Propuesta del sistema	26
2.5	Modelo de dominio	27
2.6	Representación del modelo de dominio	28
2.7	Levantamiento de requerimientos	29
2.7.1	Requisitos funcionales.....	29
2.7.2	Requisitos no funcionales	30
2.8	Casos de uso del sistema	31
2.8.1	Definición de actores del sistema a automatizar	31
2.9	Diagrama de caso de uso del sistema	32
2.10	Descripción detallada de los casos de uso del sistema	33
3.1	Modelo de Análisis	43
3.1.1	Diagrama de clases de análisis.....	43
3.2	Diseño	45
3.2.1	Diagramas de Interacción (Colaboración).....	47
3.2.2	Diagramas de clases del diseño	50
3.3	Diseño de la Base de Datos	55
3.3.1	Diagrama de clases persistentes	56
3.3.2	Modelo de datos	56
3.3.3	Descripción de las tablas.....	58
Capítulo 4: Implementación y prueba		60
4.1	Diagrama de despliegue	60
4.2	Diagrama de componentes	61
4.3	Prueba	64
4.3.1	Modelo de prueba	65
Conclusiones		68
Recomendaciones		69
Bibliografía		70
Anexos		74

Introducción:

En la actualidad la tecnología está presente en prácticamente todos los requerimientos laborales. Las necesidades de una comunicación cada vez más ágil y de centralizar la información para una mejor visualización en las empresas y centros educacionales, unido a la disponibilidad de nuevas herramientas, dieron nacimiento a lo que hoy se conoce como *Intranet*, una red que facilita de una manera extraordinaria el intercambio de información entre los miembros de una institución.

Una *Intranet* puede brindar, no sólo la divulgación de la información a los usuarios del centro con efectividad, consiguiendo que estén permanentemente informados con las últimas novedades y datos de la organización, sino que también tiene gran valor como repositorio documental, convirtiéndose en un factor determinante para conseguir el objetivo de la oficina sin papeles (1). Añadiéndoles funcionalidades como un buen buscador y una organización adecuada, se puede conseguir una consulta rápida y eficaz por parte de los usuarios de un volumen considerable de documentación (2).

En los últimos tiempos el crecimiento del volumen y la diversidad de la información que se maneja en las empresas ha traído muchas consecuencias negativas entre las que se pueden mencionar: duplicidad de los archivos, desactualización de la información y pérdida de tiempo en tareas esenciales como búsqueda y entrega de la misma. Con el propósito de evitar el caos en los sistemas de información, ha sido necesario apoyarse en herramientas de gestión documental (3).

¿Qué es un documento?

La noción de documento se ha visto alterada con el avance de la informática. Un documento era antes un papel o un tipo de medio físico. En cambio para la informática, un documento es un archivo digital susceptible de ser visualizado o compartido a través de una computadora. En este sentido un documento informático puede ser un texto, una imagen, un sonido, una animación o un video (4).

La implantación de sistemas de gestión documental como elemento clave en la actividad de la empresa se ha introducido en la cultura empresarial y ha dejado 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 en grandes corporaciones (5).

Introducción

Estos sistemas tienen como objetivo resolver los problemas derivados del exceso de información a tratar, ya sea por la gran cantidad de papel a ser manejado así como por el almacenamiento de otros objetos de información de uso corporativo como pueden ser: ficheros de oficina, informes de las propias aplicaciones de gestión empresarial, imágenes, sonido, video, fax, entre otros (5).

Los ECMs (Gestores de Contenido Empresarial, del inglés *Enterprise Content Management*) son empleados en la industria de la tecnología de la información para manejar la captura, almacenamiento, seguridad, control de versiones, recuperación, distribución, conservación y destrucción de documentos y contenidos (6).

Alfresco es una instancia de los tantos ECMs (7) (8) (9) con que cuentan las empresas en la actualidad. Es un sistema de administración de contenidos de código abierto. Está diseñado para usuarios que requieran un alto grado de modularidad y rendimiento escalable. Este incluye un repositorio de contenidos, un *framework* de portal *Web* para administrar y usar contenido estándar en portales. Es un sistema de administración de contenido *Web* y posee la capacidad de visualizar aplicaciones *Web* y sitios estáticos (10).

En Cuba, la UCI (Universidad de las Ciencias Informáticas) se suma al desarrollo tecnológico basado en la gestión documental pues como toda organización académica que por su misión realiza funciones de empresa (11) pretende adoptar las facilidades que brindan estos sistemas y adaptarlos a las especificidades y necesidades del centro, entendiendo así cuestiones como la forma de poder visualizar los documentos de sus usuarios mediante la *Intranet* con que cuenta esta institución, para centralizar la información y ofrecer mayor comodidad.

Desde su surgimiento la cantidad de personas vinculadas a ella ha aumentado considerablemente. A raíz de este crecimiento se han extendido, de forma simultánea los servicios y las solicitudes de información que se facilita mediante la *Intranet*. Debido a esto, hace algún tiempo se determinó realizar una nueva versión de la misma que, dentro de sus propósitos, concibiera una integración de servicios para el confort de sus usuarios. A esta versión se le llama *Intranet2*.

Actualmente no existe, en la nueva versión de la *Intranet*, un servicio que permita su integración con el repositorio de *Alfresco* acorde a las exigencias de los usuarios de poder tener acceso a sus documentos

Introducción

previamente gestionados en dicho repositorio.

La *Intranet2* de la UCI está construida sobre el CMS *Drupal* (Sistema de Gestión de Contenido, del inglés *Content Management Systems*), el cual posee varios módulos que de cierta forma permiten integrar este CMS con el ECM *Alfresco*; pero no satisfacen las expectativas de los servicios para la gestión documental que pretende brindar esta *Intranet2*, pues la lógica de almacenamiento que establecen dichos módulos no son del todo eficientes, debido a que cargan considerablemente la base de datos.

Según la situación problemática planteada anteriormente, el **problema científico** a resolver queda formulado de la siguiente forma:

La carencia de una comunicación acorde a las necesidades actuales, entre el repositorio de documentos de *Alfresco* y la *Intranet2* está impidiendo a los usuarios acceder desde esta a sus documentos.

Teniendo en cuenta lo descrito el **objeto de estudio** lo constituye el desarrollo *Web*.

De aquí se deriva que el **campo de acción** lo comprende el desarrollo de módulos para el CMS *Drupal*.

Se plantea entonces como **objetivo general** desarrollar un módulo que establezca una comunicación entre la *Intranet2* y *Alfresco* acorde a las necesidades actuales de los usuarios.

Para darle cumplimiento a dicho objetivo se trazaron los siguientes **objetivos específicos**:

- Describir los fundamentos teóricos sobre las tendencias actuales en el empleo de las tecnologías de gestión documental y almacenamiento de la información.
- Caracterizar el desarrollo de módulos sobre el CMS *Drupal* y las posibles vías de comunicación con el ECM *Alfresco*.
- Desarrollar un módulo sobre el CMS *Drupal* que establezca una interacción entre la *Intranet2* y el ECM *Alfresco*.

Introducción

Se presenta como **idea a defender** que, con la realización del módulo para la integración de la *Intranet2* con *Alfresco*, se logrará que los usuarios puedan acceder desde la *Intranet2* a los documentos previamente gestionados en el repositorio *Alfresco*.

Con vistas a solucionar lo expuesto anteriormente se propone la realización de las siguientes **tareas**:

- Análisis de los sistemas actuales que permiten la integración del CMS *Drupal* con el ECM *Alfresco*
- Realizar un estudio del estado del arte sobre las tecnologías existentes en materia de gestión y almacenamiento de la información.
- Estudiar y fundamentar las metodologías de desarrollo de *software* para realizar el análisis, diseño e implementación del módulo.
- Realizar obtención de requisitos.
- Definir y realizar los diferentes artefactos de los flujos de Análisis, Diseño e Implementación.
- Implementación de las funcionalidades del sistema propuesto.

Los **métodos científicos** utilizados en la investigación son:

Métodos Teóricos

- **Analítico Sintético:** Para analizar todo lo referente al objeto de estudio y de ahí extraer todos los elementos más importantes para dar cumplimiento al objetivo general.
- **Análisis Histórico Lógico:** En la búsqueda de información para la solución del problema planteado, fue necesario el estudio de los principales módulos que permite la integración de *Drupal* con *Alfresco*, pues da la posibilidad de estudiar de forma analítica su evolución y desarrollo.

Métodos empíricos

- **Entrevista:** Para conocer y precisar criterios en vista al desarrollo del sistema para la integración de *Drupal* con *Alfresco*.

Introducción

Estructuración del contenido.

El documento está compuesto por 4 capítulos:

En el capítulo 1 denominado “Módulos de Integración *Drupal- Alfresco*: Tendencias y Tecnologías Actuales” se realiza un estudio del estado del arte del tema relacionado con la investigación. Se abordan las tendencias y tecnologías actuales sobre las cuales se apoya la propuesta, así como las distintas metodologías de desarrollo de *software*, herramientas que serán usadas, lenguajes de programación, etc.

En el capítulo 2 denominado “Características de la solución de integración” se describe el flujo actual de los procesos involucrados en el campo de acción, estudiando cómo se deben ejecutar actualmente dichos procesos; también se describe el objeto de automatización, la propuesta del sistema, se determinan los requerimientos funcionales y no funcionales y se describen los casos de uso y los actores del sistema.

En el capítulo 3 denominado “Análisis y diseño del sistema propuesto” se profundiza en el problema a resolver a través de su descripción, mostrándose los artefactos relevantes generados por la metodología RUP (*Rational Unified Process*). También se define el modelo de análisis, diagramas de interacción, y diagrama de clases del diseño.

En el capítulo 4 denominado “Implementación y Pruebas” se abordará todo lo relacionado con el flujo de trabajo de implementación realizando el diagrama de despliegue el cual describe cómo y dónde el sistema será puesto en funcionamiento, y el diagrama de implementación para dar una visión de cómo las clases, artefactos y otros elementos de bajo nivel, se unen para formar componentes de alto nivel así como las conexiones entre ellos. Se realizan además, las pruebas pertinentes para garantizar la calidad del producto.

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

Este capítulo tratará los temas relacionados con el desarrollo *Web*, los sistemas de administración de contenido y los gestores de contenido empresarial. Se presentan varias definiciones y se caracterizan al CMS *Drupal* y al ECM *Alfresco*.

Se muestran además las tendencias actuales en relación a la existencia de módulos que integran *Drupal* con *Alfresco*, describiendo sus ventajas y desventajas.

Se aborda lo relacionado con el uso de las tecnologías en la informática; son mencionados brevemente algunos gestores de bases de datos. Se presentan los lenguajes de programación empleados. Se trabaja el tema de las metodologías de desarrollo de *software* más conocidas y se identifican las herramientas de desarrollo que serán utilizadas en la elaboración de la solución.

1.1 Desarrollo Web

En el mundo de hoy, *Internet* ha alcanzado un desarrollo vertiginoso en el acceso a la información. De manera tal que se ha convertido en un medio importante de información y enriquecimiento de los conocimientos para el ser humano.

Uno de los servicios que más éxito ha tenido en *Internet* ha sido la *WWW* ("la *Web*", del inglés *World Wide Web*), desarrollado inicialmente en Suiza a finales de 1990 en el CERN (Laboratorio Europeo de Física de Partículas en Génova), dirigido por *Tim Berners-Lee*. Es un sistema de información que integra todos los recursos accesibles por *Internet* en páginas de información y se basa en otro protocolo, llamado HTTP (Protocolo de Transferencia de Hipertexto, del inglés *HyperText Transfer Protocol*), que permite hacer ligas de un documento, publicado en *Internet*, a otro, haciendo la búsqueda de información bastante amigable (12).

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

Desde entonces el crecimiento ha sido exponencial. Han surgido nuevas versiones de la *Web* proporcionando servicios renovados. Tal es el caso de la *Web 2.0* una *Web* social, dinámica, abierta, más humana y que promueve el diálogo. Basada en la creación de páginas *Web* donde los contenidos son compartidos y producidos por los propios usuarios del portal. Su infraestructura está relacionada con nuevas tecnologías que han hecho que sea más fácil publicar información y compartirla con otros sitios *Web*. Unido a esto se han actualizado los sistemas gestores de contenido haciendo que personas con limitados conocimientos de programación *Web* puedan gestionar su propio *blog* (13).

1.2 Sistemas de Gestión de Contenidos

La lógica evolución de la *Web*, desde hace más de 10 años, ha generado la sustitución de páginas y documentos estáticos por documentos generados dinámicamente (14).

Al reflexionar en este sentido Jesús Tramullas es de la opinión de que "...desde el año 2000 se ha producido una convergencia entre todas las plataformas, de forma que pueden encontrarse actualmente soluciones que pretenden ser globales y ofrecer soporte a todo el proceso de gestión de información en una organización. Las herramientas para este trabajo han recibido la denominación de Sistemas de Gestión de Contenidos" (15).

Un CMS es una herramienta que permite la creación y administración de contenidos principalmente en páginas *Web* (16).

Los CMSs poseen una interfaz que controla una o varias bases de datos donde se encuentra toda la información del sitio distribuida en forma de contenidos. Su configuración es fácil, puesto que separa el código del diseño y la administración permite cambiar casi todo el contenido que se encuentra publicado (17).

Algunos de los aspectos más importantes que hacen útil y necesario el empleo de un CMS son (18) (19):

- Inclusión de nuevas funcionalidades en la *Web*.

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

- Mantenimiento de gran cantidad de páginas.
- Reutilización de objetos o componentes.
- Cambios del aspecto de la *Web*.
- Navegabilidad.
- Facilidad de actualización.

1.2.1 CMS Drupal

Drupal es un CMS modular y muy configurable. Es un programa de código abierto, con licencia GNU/GPL (Licencia Pública General, del inglés *General Public License*), escrito en PHP (*Preprocessed Hypertext Pages*), desarrollado y mantenido por una activa comunidad de usuarios. Se destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la *Web* y un énfasis especial en la usabilidad y consistencia de todo el sistema (20).

Drupal se compone de una infraestructura base y un conjunto de módulos que ofrecen gran cantidad de funcionalidades, incluyendo *chat*, foros, búsquedas, entre otros. Estos módulos permiten modificar el comportamiento de *Drupal* ofreciendo nuevas opciones.

1.3 Gestores de Contenido Empresarial

Los Gestores de Contenidos Empresariales son la tecnología empleada para capturar, gestionar, almacenar, conservar y distribuir documentos y contenidos relacionados con los procesos de una institución (6).

Los ECMs pueden ser vistos de tres formas diferentes (10):

ECM como integrador de sistemas: Puede ser usado para sobreponerse a las restricciones que existen entre diferentes aplicaciones. A través de los mismos se pueden interconectar diferentes aplicaciones por separado mientras el usuario puede pasar desapercibido.

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

ECM como servicio o utilidad independiente: Puede emplearse para administrar la información sin considerar de donde proviene la misma. Esta funcionalidad se identifica como un servicio que puede ser consumido por cualquier tipo de aplicación, y contribuye de cierta manera en la interoperabilidad entre aplicaciones o la integración entre las mismas.

ECM como un repositorio para todo tipo de información: En este sentido los gestores de contenido empresarial pueden ser usados como un almacén de contenido en el cual se pueden almacenar los documentos de una entidad en un repositorio con una estructura uniforme.

1.3.1 ECM Alfresco

Alfresco es la alternativa de código abierto para la gestión de contenido empresarial, que proporciona colaboración, gestión documental, gestión de contenido *Web* e imágenes.

Este ECM es libre, desarrollado en *Java* y basado en estándares abiertos. Incluye un repositorio de contenidos, un *framework* de portal *Web* para administrar y usar contenido estándar en portales, una interfaz CIFS (*Common Internet File System*) que provee compatibilidad de sistemas de archivos en *Windows* y sistemas operativos similares a *Unix*, un sistema de administración de contenido capaz de virtualizar aplicaciones *Web* y sitios estáticos vía *Apache Tomcat* así como soporte para varios idiomas (10).

Este gestor extiende la gestión documental para dar respuesta al ciclo de vida de forma íntegra, no solo de los documentos, sino de todos los contenidos de una organización, independientemente de su naturaleza (21).

1.4 Módulos de Integración Drupal–Alfresco

En el sitio oficial de Drupal, su amplia comunidad de usuarios ha puesto a disposición tres módulos que permiten la integración de *Drupal* con *Alfresco*, los que serán descritos en detalles a continuación (22).

Módulo Alfresco

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

Alfresco es un módulo que brinda soporte para descargas directas y privadas de archivos almacenados en el repositorio de documentos de *Alfresco*, además permite almacenar en caché las propiedades de estos archivos, sincronizar los nodos y mostrar vistas de apoyo.

El módulo principal *Alfresco* está compuesto por otros tres que son listados a continuación:

- *Alfresco_browser*. Permite a los usuarios visualizar, buscar, navegar y recuperar los nodos del repositorio de *Alfresco*.
- *Alfresco_attach*. Permite crear y adjuntar elementos de *Alfresco* a otros contenidos en *Drupal*.
- *Alfresco_import*. Permite importar múltiples nodos *alfresco* desde un espacio de *Alfresco*.

Este módulo, a pesar de ser uno de los más usados, presenta el inconveniente de crear un nodo en *Drupal* por cada contenido en *Alfresco*, cargando la base de datos proporcionalmente a la cantidad de contenidos almacenados en el repositorio *Alfresco*.

CMIS API

El proyecto CMIS API pretende ofrecer una API (Interfaz de Programación de Aplicaciones, del inglés *Application Programming Interface*) genérica para la integración con CMIS (Interoperabilidad de Servicios de Gestión de Contenidos, del inglés *Content Management Interoperability Services*) compatible con los sistemas CMS empresariales. Distribuidor de módulos específicos, tales como CMIS *Alfresco*, utilizará esta API para permitir el intercambio de contenido con los repositorios específicos. La creación de este módulo es un esfuerzo conjunto entre *Optaros*, *Acquia*, y *Alfresco*, comunidades que se dedican a la ayuda y publicación de soluciones novedosas.

Este módulo no ofrece funciones de usuario final por su cuenta y debe ser usado en conjunto con otros módulos que lo requieran, tales como el módulo de CMIS *Alfresco*. Se encuentra actualmente en estado beta y sus funcionalidades son muy básicas.

CMIS Alfresco

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

El proyecto de integración CMIS *Drupal-Alfresco* tiene como objetivo proporcionar un mecanismo para lograr los beneficios de *Alfresco* y *Drupal* trabajando juntos a través de una interfaz compatible con los estándares.

Este módulo se encuentra actualmente en estado beta y sus funcionalidades son muy básicas. Es necesario aclarar que este módulo requiere el módulo de CMIS API.

1.5 Tecnologías, Metodología y Herramientas

El módulo que se implementará se encuentra en el dominio de aplicaciones *Web* debido a su alcance y disponibilidad. Para ello se hace necesario realizar una fundamentación, estableciendo comparaciones de las tecnologías, herramientas y metodologías empleadas en el desarrollo del mismo, para tomar las decisiones correctas.

1.5.1 HTML

HTML (Lenguaje de Marcación de Hipertexto, del inglés *HiperText Markup Language*) es un lenguaje que se utiliza para establecer la estructura y contenido de un sitio *Web*, tanto de texto, objetos como de las imágenes (23).

Este lenguaje de programación funciona por medio de etiquetas, que describen la apariencia o función del texto enmarcado. Puede llegar a incluir un *script* o código que tenga incidencia en el comportamiento del navegador *Web*.

1.5.1.1 XHTML 1.0

El XHTML (Lenguaje Extensible de Marcado de Hipertexto, del inglés *eXtensible Hypertext Markup Language*) es una familia de módulos y tipos de documentos que reproduce, engloba y extiende HTML 4.0. Los tipos de documentos de la familia XHTML están basados en XML (Lenguaje de Etiquetado Extensible, del inglés *eXtensible Markup Language*) y diseñados fundamentalmente para trabajar en conjunto con aplicaciones de usuario basados en XML (24).

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

Las ventajas más evidentes que ofrece el migrar a XHTML son (25):

- Los documentos XHTML se establecen sobre la base de las reglas XML. Por tanto, pueden ser visualizados, editados y validados por cualquier herramienta estándar XML.
- Los documentos XHTML pueden contener aplicaciones (por ejemplo *applets* o *scripts*) que se apoyen en DOM (Modelo de Objetos del Documento, *del inglés Document Object Model*) y que modifiquen la propia estructura del documento XHTML.
- Permite insertar en el documento XHTML nuestras propias marcas que no tienen por qué estar definidas en el estándar general. Esto es lo que se llama modularización XHTML.
- XHTML está diseñado para poder ser visualizado en cualquier tipo de plataforma.

1.5.2 CSS

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

CSS es toda una revolución en el mundo del diseño *Web*. Entre sus beneficios se pueden mencionar (17) (26):

- Control de la presentación de muchos documentos desde una única hoja de estilo.
- Logra que los documentos *Web* se vean igual en todos los navegadores.
- El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño.
- Control centralizado de la presentación de un sitio *Web* completo con lo que se agiliza de forma considerable la actualización del mismo.

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

- Optimiza los tiempos de carga y tráfico del servidor.

El consorcio W3C (*World Wide Web Consortium*), encargado de formular la especificación de las hojas de estilo ha definido hasta la fecha tres versiones de CSS dentro de las que se encuentran (27):

CSS1: fue la primera especificación que desarrolló el WWW. Las hojas de estilo CSS1 describen el formato del texto y de los componentes de una página (fuente, color, tamaño, etc.).

CSS2: permite ubicar elementos de XHTML en diferentes capas o *layers*, cuyo posicionamiento no tiene que seguir el flujo HTML (plantea el modelo de cajas donde cada bloque se define como una caja que se coloca en un lugar concreto). CSS2 incluye a CSS1 (salvo pequeños detalles).

CSS3: aún está en desarrollo por el W3C.

Para garantizar el buen funcionamiento del módulo se hace imprescindible el uso del CSS en su versión 2.

1.5.3 Javascript

Se trata de un lenguaje de tipo *script* compacto, basado en objetos y guiado por eventos diseñado específicamente para el desarrollo de aplicaciones cliente-servidor (27).

Este lenguaje fue desarrollado por *Netscape* para incrementar las funcionalidades del lenguaje HTML. Sus características más importantes son (28):

- Es un lenguaje interpretado.
- Es un lenguaje orientado a eventos.
- Es un lenguaje orientado a objetos.

En la actualidad se han elaborado diferentes *plugins* y *frameworks* para potenciar el uso del *Javascript*, de estos se pueden citar: *Prototype*, *Mootools*, *jQuery*, *Qooxdoo*, *Dojo*, *Ext Js* y muchos otros con una amplia gama de funciones listas para ser usadas en el desarrollo *Web*. De todas estas librerías se escogió el

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

jQuery ya que cuenta con gran aceptación por estar estupendamente documentado, por ser muy simple y permitir desarrollar con un código limpio y elegante (29).

1.5.3.1 jQuery

jQuery es una librería para el lenguaje *Javascript* que ofrece una infraestructura con la que se logra mucha mayor facilidad para la creación de aplicaciones complejas del lado del cliente. Brinda gran ayuda en la creación de interfaces de usuarios, efectos dinámicos y aplicaciones que hacen uso de *Ajax*. Es multiplataforma y para comenzar a utilizarlo simplemente se debe incluir un *script* que contiene el código de *jQuery*.

Entre las características más significativas que definen a *jQuery* se encuentran (30):

- Es orientada a eventos
- Manipulación de la hoja de estilos CSS.
- Efectos y animaciones
- Selección de elementos del DOM.
- Soporta extensiones.
- Ofrece utilidades como: obtener información del navegador, operar con objetos y *arrays*, función *trim()* (elimina los espacios en blanco del principio y final de una cadena de caracteres).

jQuery es serio, estable y con un gran equipo de desarrolladores a cargo de las mejoras y actualizaciones.

1.5.4 Lenguajes Dinámicos

Para la realización de este módulo, la simple utilización de *Javascript*, HTML y CSS no sería suficiente, pues con las características que posee no se puede prescindir de un lenguaje dinámico necesario para las

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

operaciones sobre bases de datos, controles de secciones y otras actividades que solo se pueden realizar con su uso.

1.5.4.1 PHP

El lenguaje de programación PHP (*Preprocessed Hypertext Pages*) fue creado por *Rasmus Lerdorf* a finales de 1994, su principal objetivo está encaminado a la construcción de páginas *Web* del lado del servidor (31).

Es un lenguaje de programación rápido y soporta la mayoría de los servidores *Web* incluyendo *Apache*, *Microsoft Internet Information Server*, *Personal Web Server*, entre otros.

Muchos *Frameworks* utilizan este lenguaje para su desarrollo entre los que se encuentran *PRADO*, *Codelgniter*, *CakePHP*, *Symphony*, entre otros. Existen además CMSs que se implementan con este lenguaje como por ejemplo *Joomla* y *Drupal* (17).

PHP es un lenguaje de alta potencia, fácil de usar e incluye la programación orientada a objetos. Las ventajas que presenta justifican la elección de este lenguaje para el desarrollo de la propuesta.

- Es un lenguaje multiplataforma (se puede usar en varios sistemas operativos como GNU *Linux* y *Windows*).
- Es libre, lo cual presenta una alternativa de fácil acceso para todos.
- Permite manipular datos desde diversas fuentes, incluso datos ingresados por los usuarios desde formularios HTML.
- Presenta gran capacidad de conexión con la mayoría de los manejadores de bases de datos utilizados en la actualidad, destacándose su conectividad con *MySQL*.
- Posee la capacidad de expandir su potencial utilizando módulos, llamados extensiones. Así como

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

una amplia documentación en su página oficial.

- El CMS utilizado para la integración, *Drupal*, está desarrollado en ese lenguaje.

1.5.5 Sistemas de Gestión de Base de Datos

Un sistema de gestión de base de datos es un paquete generalizado de *software* que se ejecuta en un sistema computacional anfitrión, centralizando el acceso a los datos y actuando de interfaz entre los datos físicos y el usuario, permitiéndole la facilidad de manejo de grandes volúmenes de información (32).

1.5.5.1 PostgreSQL

Es un magnífico gestor de bases de datos y es multiplataforma. Permite una fácil gestión de los usuarios y de las bases de datos que contenga el sistema. Sirve de soporte a los lenguajes más usados como PHP, C, C++, *Java*, *Python*, *Ruby*, entre otros, y al protocolo de comunicación encriptado por SSL (Protocolo de capa de conexión segura, del inglés *Secure Sockets Layer*).

El número de bases de datos que puede contener es ilimitado. Tiene transacciones, integridad referencial, vistas y múltiples funcionalidades (33).

PostgreSQL ofrece una garantía de integridad en los datos y a pesar de no ser rápido presenta gran escalabilidad y rendimiento bajo grandes cargas de trabajo (34).

1.5.5.2 MySQL

MySQL fue desarrollado originalmente para manejar grandes bases de datos. Su conectividad, velocidad, seguridad y poco consumo de recursos hacen de *MySQL* un servidor bastante apropiado para acceder a bases de datos en *Internet*.

La *Intranet2* de la UCI trabaja con *MySQL* como gestor de BD, por eso la elección de su uso, aunque la

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

realización del módulo será transparente a cualquier gestor utilizado por *Drupal*.

Algunas de sus principales características (32):

- Utiliza múltiples tablas para almacenar y organizar la información.
- Adaptación a diferentes entornos de desarrollo, que permite su interacción con los lenguajes de programación más utilizados como PHP, *Perl* y *Java* y su integración en distintos sistemas operativos.
- Estabilidad.
- Rapidez en ejecutar consultas.
- Multihilo.
- Multiusuario.

1.5.6 Servidor Web. Apache

Un servidor de aplicaciones proporciona servicios de aplicación a las computadoras clientes. Estos servidores gestionan la mayor parte de las funciones de lógica de negocio y de acceso a los datos de la aplicación. Los beneficios principales en la aplicación de esta tecnología son la centralización y disminución de la complejidad en el desarrollo de aplicaciones.

Apache es el servidor *Web* más usado en su tipo. Está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo.

Entre las características que posee *Apache* y que lo llevó al éxito en la inserción y utilización en ámbitos empresariales y tecnológicos se encuentran (35):

- Corre sobre una multitud de plataformas y sistemas operativos.

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

- *Apache* es un servidor *Web* altamente configurable y de diseño modular, capaz de ampliar su funcionalidad y calidad de servicios.
- Trabaja en conjunto con gran cantidad de lenguajes de programación como PHP, Perl, soporte con CGI (Interfaz de entrada común, del inglés *Common Gateway Interface*), Java, JSP (*Java Server Pages*) y otros más.
- Cuenta con archivos Log, en donde registra gran cantidad de información global del sistema y errores producidos en un determinado tiempo.
- Posee gran cantidad de documentos, ejemplos y ayuda en *Internet* en varios idiomas.
- Alto grado de adaptación.
- Ajuste dinámico del número de procesos del servidor.
- *Hosts* virtuales.

Teniendo en cuenta lo antes mencionado *Apache* se convierte en una elección prácticamente sin discusión.

1.5.7 Metodologías de Desarrollo

Desde hace algún tiempo los desarrolladores de *software* vienen buscando metodologías que mejoren la calidad y productividad del *software*, de forma tal que se trabaje eficientemente para evitar que los proyectos sean terminados sin éxito.

Dentro de la ingeniería de *software* existen algunas metodologías que se encargan de elaborar estrategias de desarrollo de *software* que promueven nuevas prácticas para un buen desarrollo del mismo, entre las más conocidas existen: XP (*eXtreme Programming*), FDD (*Feature Driven Development*) y RUP (*Rational Unified Process*) (31).

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

1.5.7.1 XP

Es una de las metodologías de desarrollo de *software* más exitosas en la actualidad, es ágil y se utiliza generalmente para proyectos de corto plazo y equipos pequeños. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto (36).

Entre sus principales características se destacan las siguientes:

- Comunicación.
- Simplicidad.
- Retroalimentación (*Feedback*).

En esta metodología la comunicación es un punto importante y debe realizarse entre los programadores, los jefes de proyecto y los clientes (37).

1.5.7.2 FDD

Esta metodología está pensada para proyectos con tiempo de desarrollo relativamente cortos (menos de un año). Basada en un proceso con iteraciones que producen un *software* funcional que el cliente y la dirección de la empresa pueden ver y monitorizar (31).

FDD se divide en 5 fases:

- Desarrollo de un modelo general.
- Construcción de la lista de funcionalidades.
- Plan de *releases* sobre la base de las funcionalidades a implementar.
- Diseñar sobre la base de las funcionalidades.

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

- Implementar sobre la base de las funcionalidades.

El trabajo (tanto de modelado como de desarrollo) se realiza en grupo, aunque siempre habrá un responsable último (arquitecto jefe o jefe de programadores en función de la fase en la que se encuentre), con mayor experiencia, que tendrá la última palabra en caso de no llegar a un acuerdo. Al hacerlo en grupo se consigue que todos formen parte del proyecto y que los inexpertos aprendan de las discusiones de los más experimentados, y al tener un responsable último, se asignan las responsabilidades que todas las empresas exigen (38).

1.5.7.3 RUP

RUP es uno de los procesos más generales de los existentes actualmente, ya que en realidad está pensado para adaptarse a cualquier proyecto.

Esta metodología divide el proceso de desarrollo en ciclos, obteniendo una versión del *software* al final de cada ciclo, que se divide en fases (Figura 1), las cuales concluyen con un hito, donde se debe tomar una decisión importante. Las fases son las siguientes (39):

- Inicio
- Elaboración
- Construcción
- Transición

A continuación se presentan los flujos de trabajo que comprende RUP:

- Modelado del negocio
- Análisis de requisitos
- Análisis y diseño

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

- Implementación
- Prueba
- Despliegue
- Gestión de configuración y cambios
- Gestión del proyecto
- Gestión del entorno

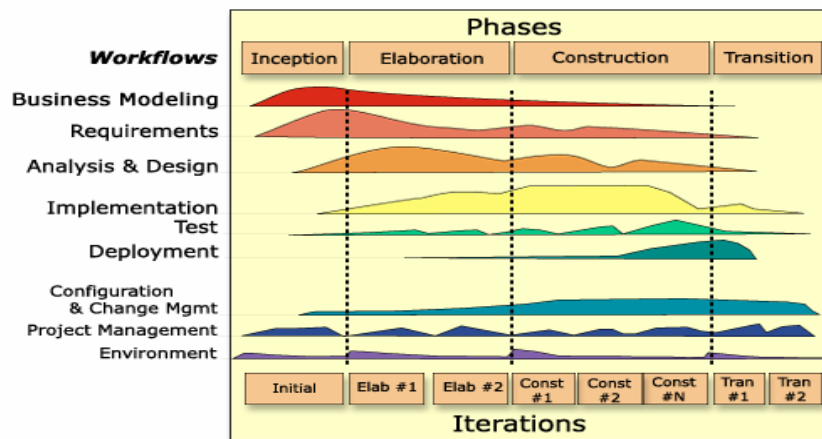


Figura 1. Fases y flujos de trabajo de RUP

RUP incluye productos tangibles del proceso como por ejemplo, el modelo de caso de uso y el código fuente, llamados artefactos de desarrollo. Define además una serie de roles que se distribuyen entre los miembros del proyecto y las tareas de cada uno a lo largo del proceso.

El ciclo de vida de RUP es iterativo incremental, centrado en la arquitectura y dirigido por caso de uso. Utiliza UML (Lenguaje Unificado de Modelado, del inglés, *Unified Modeling Language*) como herramienta

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

principal para el lenguaje de modelado (40).

Se decidió usar esta metodología, por todas sus ventajas y facilidades antes expuestas.

1.5.8 Herramientas de desarrollo

La complejidad y amplitud tecnológica que caracteriza el mundo del desarrollo *Web*, unido a todas sus técnicas de programación, algunas de ellas ya expuestas, exigen el uso de algunas potentes herramientas que facilitan el desarrollo exitoso de una aplicación consistente. Con más detalles se describen a continuación dos de estas herramientas que fueron las seleccionadas.

1.5.8.1 Netbeans

El *NetBeans IDE (Integrated Development Environment)* es un ambiente libre de desarrollo integrado con *Open Source* para desarrolladores de *software*. El mismo ofrece todas las herramientas necesarias para crear escritorios profesionales, *Enterprise*, *Web* y aplicaciones móviles con el lenguaje *Java*, *JavaFX*, *C/C++* y lenguajes dinámicos como *PHP*, *Javascript*, *Groovy* y *Ruby*. Es multiplataforma y de fácil instalación (41).

Es un producto gratuito sin restricciones de uso, presenta un excelente completamiento de código.

La programación mediante *NetBeans* se realiza a través de componentes de *software* modulares, llamados módulos, los cuales están a disposición del usuario, en su página oficial, para conseguir mejoras en las aplicaciones.

1.5.8.2 Visual Paradigm

Las potencialidades de *Visual Paradigm* no tendrían sentido sin el empleo de un robusto lenguaje de modelado, tal es el caso de UML.

UML es un lenguaje de construcción de modelos para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de *software*, con los que se construyen

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

mayormente sistemas orientados a objetos. Introduce facilidades como chequeos sintácticos de los diagramas, chequeos de consistencia, facilidades de importación y exportación, la generación de código en diferentes lenguajes, uso de ingeniería inversa para obtener diagramas a partir de código, hasta la posibilidad de definir perfiles para el diseño en un dominio específico (42).

Visual Paradigm para UML es un producto galardonado que facilita a las organizaciones el diseño visual de los distintos diagramas. Esta herramienta ayuda a los equipos de desarrollo en la confección de los distintos modelos que van desde la modelación del negocio hasta el despliegue.

Esta herramienta soporta el ciclo de vida completo del desarrollo de *software*: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Además, está diseñado para una amplia gama de roles, incluidos analistas y arquitectos de sistema y de datos. Estas características permitirán el uso de esta herramienta en su versión *Visual Paradigm for UML 6.0*.

1.5.9 Propuesta de solución

Una vez tratado el tema de las tecnologías, herramientas y metodologías a usar según las especificaciones y necesidades del trabajo, se tomó la decisión de emplear para el desarrollo del módulo:

- Como tecnología *Web* del lado del cliente: XHTML 1.0 junto al CSS2 y el *Javascript* con la librería *jQuery*.
- Como lenguaje dinámico del lado del servidor se pretende usar PHP5.
- El servidor *Web Apache 2.0* y como gestor de base de datos MySQL 5.0.
- Las herramientas usadas para complementar el trabajo serán *Visual Paradigm for UML 6.0* para generar parte de la documentación y *Netbeans 6.8* para la programación con PHP.

En este capítulo se analizaron las tendencias actuales en relación al desarrollo *Web*. Se hizo un estudio de los módulos existentes que permiten la integración de *Drupal* con *Alfresco*. Se detallaron los lenguajes

Capítulo 1: Módulos de Integración Drupal- Alfresco: Tendencias y Tecnologías Actuales

de programación y herramientas a utilizar. Se describieron además sistemas gestores de base de datos y se hace alusión a los procesos de desarrollo de *software* más conocidos actualmente.

Capítulo 2: Características de la solución de integración

En este capítulo se realiza una propuesta del sistema que se quiere implementar, luego de haber analizado otros sistemas, demostrando lo necesario que se hace implementar la propuesta de solución planteada a continuación. Se definen conceptos y se identifican entidades que se relacionan en un modelo de dominio, el cual permite determinar los requisitos necesarios que debe cumplir la aplicación para cubrir las necesidades que lo originan, dando el punto de partida para empezar a desarrollar el módulo con las características y funcionalidades requeridas. Por otra parte se realiza la descripción y representación de los casos de uso del sistema.

2.1 Problema y situación problemática.

En los inicios de la UCI, se concibió un proyecto denominado Ciudad Digital cuyo resultado fundamental fue la *Intranet 1.0*. Posteriormente se realiza una actualización (versión 1.5) la cual no cumplía con las necesidades que la comunidad universitaria exigía dado el incremento del personal y de los múltiples servicios que se ofertaban. Debido a esta situación, se decide crear una nueva versión de la *Intranet* que tuviera como premisa fundamental ofrecerle al usuario una integración de todos los servicios existentes en la universidad en un solo sistema. A esta versión se le llama *Intranet 2.0* o *Intranet2*.

Como parte del proceso de incrementar la disponibilidad de servicios de la *Intranet2*, se decide integrarle uno que proporcione un enlace a la información almacenada en el repositorio de documentos del ECM *Alfresco*.

La *Intranet2* se realizó utilizando el CMS *Drupal*, el cual posee varios módulos, desarrollados por su activa comunidad de usuarios, que brindan la integración de este CMS con *Alfresco*. Entre ellos se puede mencionar el módulo “*Alfresco*” que es uno de los más utilizados actualmente, pues los otros dos (CMIS API y CMIS *Alfresco*) se encuentran en estado beta y son muy básicos.

El módulo “*Alfresco*” brinda gran cantidad de funcionalidades, por ejemplo: permite a los usuarios visualizar, buscar, navegar y recuperar los nodos del repositorio de *Alfresco*, permite además importar y adjuntar elementos *alfresco* a los contenidos de *Drupal*. El inconveniente que presenta éste módulo es que crea un nodo en *Drupal* por cada contenido almacenado en el repositorio de documentos de *Alfresco*

Capítulo 2: Características de la solución de integración

lo que trae como consecuencia que la base de datos se cargue proporcionalmente a la cantidad de contenidos almacenados en este repositorio.

Como respuesta a la problemática planteada, se propone el *Módulo IDAM* (Módulo de Integración *Drupal-Alfresco* del inglés *Integration Drupal-Alfresco Module*), el cual permitirá que los usuarios de la *Intranet2* puedan acceder a los documentos que previamente fueron gestionados en el repositorio de documentos de *Alfresco*, de acuerdo a sus expectativas actuales.

2.2 Objeto de automatización

Será objeto de automatización el proceso vinculado a la gestión documental mediante la integración de los usuarios de la *Intranet2* con los contenidos almacenados en *Alfresco*. Se entiende como integración el hecho de que los usuarios puedan acceder a los documentos de su espacio personal almacenados en *Alfresco*, posterior a haberse autenticado en la *Intranet2*.

2.3 Información que se maneja

Para el desarrollo de este sistema se hace necesario interactuar con toda la información que nos brinda el CMS *Drupal*, dígase usuarios, roles y contenidos, como es el caso de las consultas que se crean para la recuperación de documentos almacenados en *Alfresco*, los cuales constituyen la información más valiosa que se maneja en la aplicación.

2.4 Propuesta del sistema

El sistema permite que los usuarios puedan acceder a sus documentos almacenados en *Alfresco* desde la *Intranet2* luego de haberse autenticado en la misma. Además permite que los usuarios de acuerdo con sus privilegios puedan navegar por dicho repositorio. El módulo cuenta con una interfaz para configurar todo lo relacionado a la conexión con *Alfresco*, posee además otra interfaz donde se gestionan las consultas para recuperar los diferentes tipos de información que se publican mediante bloques creados por *Drupal*; esta información se almacena en la caché para agilizar posteriores búsquedas por parte de los usuarios.

Capítulo 2: Características de la solución de integración

La diferencia que presenta IDAM con relación a los módulos existentes es que establece una nueva lógica de almacenamiento de acuerdo a las necesidades y exigencias del entorno para el cual se creó.

2.5 Modelo de dominio

Después de realizar un estudio de los procesos que se van a efectuar, se llegó a la conclusión, que el negocio estudiado presenta bajo nivel de estructuración, donde los flujos de información se encuentran difusos, y cuando se desea realizar una actividad, múltiples personas intervienen en la misma, lo que implica solapamiento de responsabilidades, además es difícil establecer las reglas de funcionamiento, por lo que se propone realizar un modelo de dominio.

El modelo de dominio o modelo conceptual es una representación visual de los conceptos u objetos del mundo real, significativos para un problema o área de interés. Se realiza si no se logra determinar el proceso de negocio con fronteras bien establecidas donde se logra ver claramente quienes son las personas que lo inician.

Representa clases conceptuales del dominio del problema que vienen a ser las ideas u objetos físicos y el/los enlaces de unos objetos con otros, ayudando de esta forma a la elaboración del glosario de términos, facilitando la comunicación entre los desarrolladores del sistema y un mayor entendimiento del contexto en que se desarrolla el sistema, producto del empleo de un lenguaje común (10).

Definición de las entidades y conceptos principales:

Drupal: Sistema de Administración de Contenidos donde está construida la *Intranet2*.

Usuarios: Representa a personas o sistemas de la entidad que interactúan con el sitio en *Drupal*. Inicia todos los procesos del módulo IDAM.

Rol: Papel que juegan los usuarios dentro del sistema.

Permisos: Permiten controlar qué es lo que pueden hacer los usuarios en el sitio.

Capítulo 2: Características de la solución de integración

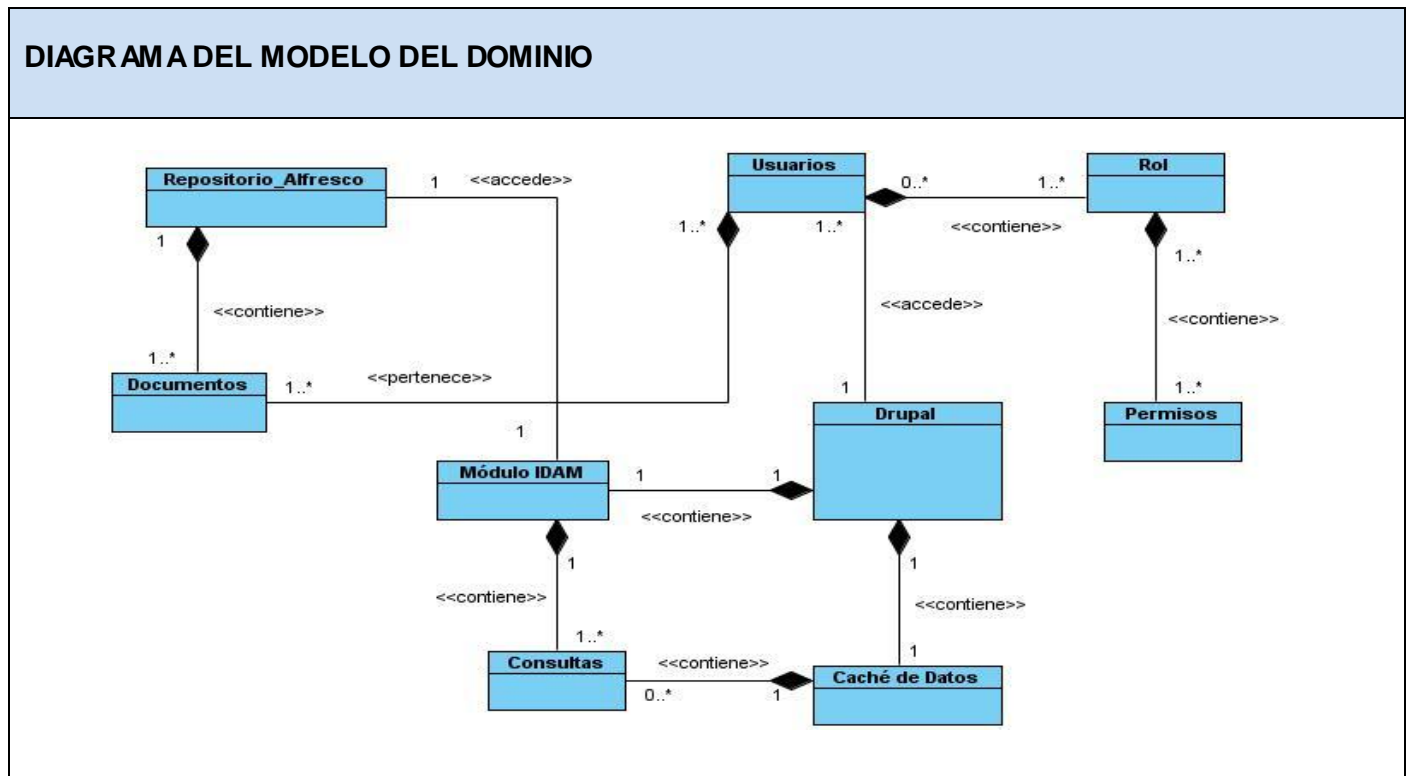
Documentos: Archivos que ofrecen información, pueden ser documentos de texto, imágenes, sonido, video, entre otros.

Repositorio Alfresco: Gestor de Contenido Empresarial donde se almacenan y se gestionan los documentos.

Consultas: Es el método para acceder a los datos en la base de datos. En este caso los datos son los documentos almacenados en el repositorio.

Caché de Datos: Sistemas que almacenan datos de forma duplicada para mejorar su capacidad de procesamiento logrando un menor tiempo de acceso a los datos almacenados en distintos tipos de memoria.

2.6 Representación del modelo de dominio



2.7 Levantamiento de requerimientos

Es uno de los flujos de trabajo más importantes, porque en él se establece qué es lo que tiene que hacer exactamente el sistema que se construye. En esta línea los requisitos son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que se especifiquen. Se dividen en dos grupos: los requisitos funcionales y los requisitos no funcionales.

2.7.1 Requisitos funcionales




Teniendo en cuenta que los requisitos funcionales son capacidades o condiciones con las cual el sistema debe cumplir e indican su comportamiento, se debe analizar cuáles son las funcionalidades del sistema que cumplan con los objetivos que se plantearon, enumerando para ello las acciones que la aplicación debe ser capaz de realizar.

RF1 Visualizar información

RF2 Navegar por el repositorio

RF3 Establecer configuración

RF4 Gestionar consulta

-  Crear consulta
-  Modificar consulta
-  Eliminar consulta

RF5 Exportar consulta como bloque

RF6 Listar consulta

RF7 Visualizar documentos del grupo

2.7.2 Requisitos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Son aquellas propiedades que hacen que el mismo sea atractivo, usable, rápido, entre otras características. Forman una parte significativa de la especificación, permiten que tanto clientes como usuarios puedan valorar las características no funcionales de la aplicación. Son fundamentales en el éxito del producto pues marcan la diferencia entre un producto bien aceptado y otro con poca aceptación.

Existen numerosas categorías para clasificar los requisitos no funcionales. A continuación se muestran los requisitos que la aplicación en cuestión debe tener.

Soporte

- El sistema debe correr sobre el gestor de base de datos MySQL o PostgreSQL.
- Se requiere utilizar la versión 6 de *Drupal*.
- Servidor *Web Apache* 2.0.x o superior.
- Versión de PHP 5.0 o superior.
- Por parte del cliente se requiere un navegador capaz de interpretar *javascript*.
- Se requiere de la librería PHP_soap.
- Se requiere del módulo jquery_update para actualizar el *jQuery* de *Drupal*.

Portabilidad

- El módulo se desarrolló sobre el CMS *Drupal* el cual contiene dentro de sus principales ventajas la de ser multiplataforma, por ende dicho módulo podrá ser usado en diferentes sistemas operativos.

Usabilidad

- Su funcionamiento será intuitivo y requerirá de conocimientos mínimos para su uso.

Capítulo 2: Características de la solución de integración

- El sistema será flexible y muy fácil de usar ya que, la creación de consultas, se implementó a nivel de filtros los cuales garantizan gran asequibilidad a la hora de realizar recuperaciones de archivos en *Alfresco*.

Legales

- La plataforma escogida para el desarrollo de la aplicación está basada en la licencia GNU/GPL.

Rendimiento

- Se hace uso de la tabla `idam_object_cache` con el objetivo cachear el HTML de los resultados obtenidos de las consultas. De esta forma se evita la realización de peticiones innecesarias al servidor de *Alfresco*, proporcionándole al sistema tiempos de respuestas muy pequeños.

Seguridad

- Está implementado a través de varios niveles de usuarios, con permisos que correspondan al rol que desempeñan en la aplicación ya que *Drupal* cuenta con el módulo *user* el cual verifica los permisos de cada usuario.
- *Alfresco* también garantiza la integridad y la disponibilidad de los documentos almacenados.

2.8 Casos de uso del sistema

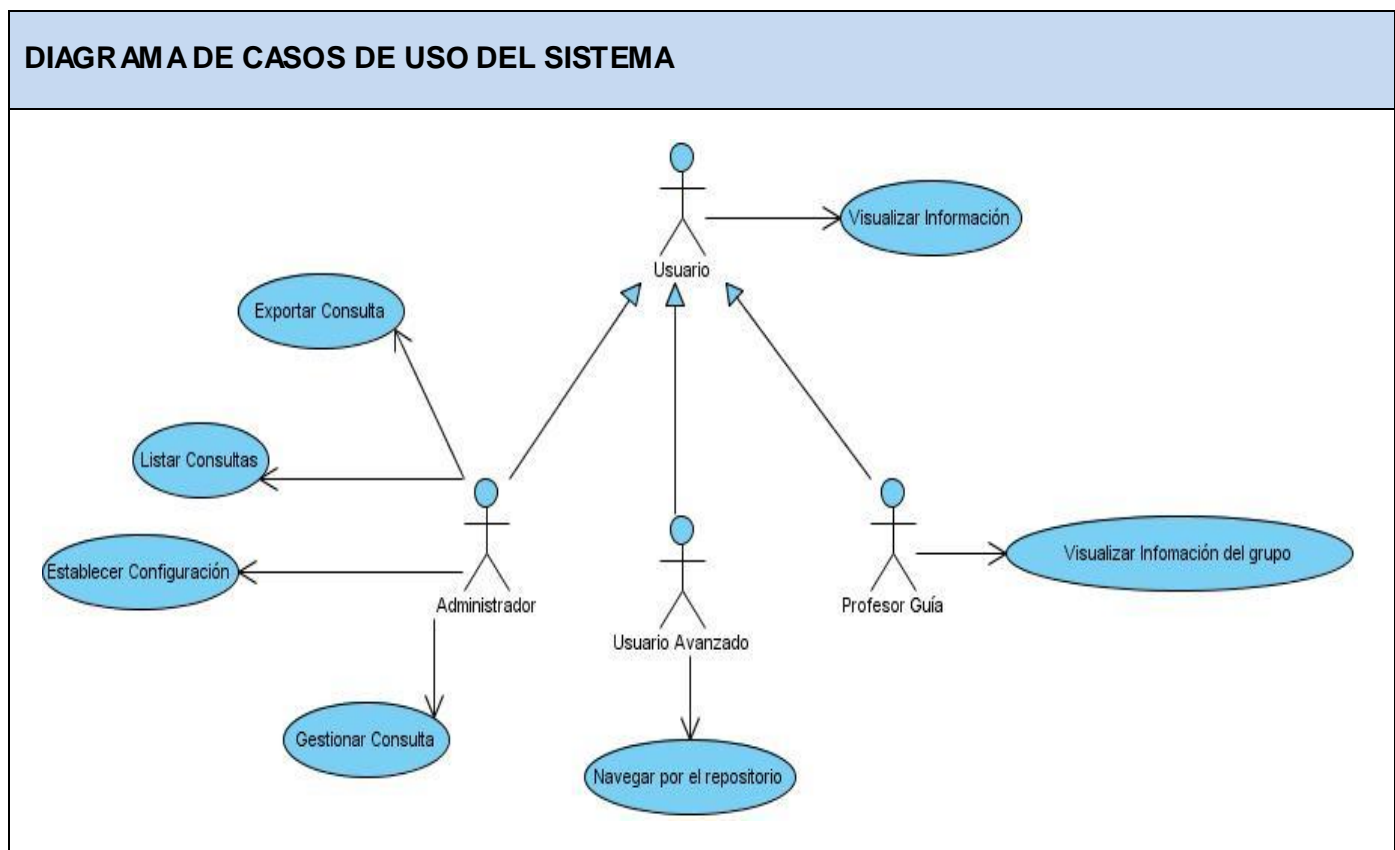
2.8.1 Definición de actores del sistema a automatizar

Actor	Justificación
Administrador	Es el encargado de administrar el módulo, dígase establecer la configuración para la conexión con <i>Alfresco</i> , gestionar las consultas y realizar las recuperaciones de la información.

Capítulo 2: Características de la solución de integración

Usuario	Es la persona que visualiza los documentos de su espacio personal, desde la <i>Intranet2</i> .
Usuario Avanzado	Es la persona que tiene algunos privilegios, dígame directivos, decanos, etc, para navegar por el repositorio.
Profesor	Es la persona que visualiza los documentos de su grupo

2.9 Diagrama de caso de uso del sistema



2.10 Descripción detallada de los casos de uso del sistema

A través de la expansión de los casos de uso se describe paso a paso la secuencia de eventos que los actores utilizan para completar un proceso a través del sistema.

Caso de uso	
CU-1	Visualizar información
Propósito	Visualizar los documentos del espacio personal del usuario.
Actores	Usuario
Resumen: El caso de uso se inicia cuando el usuario accede a la <i>Intranet2</i> y se autentica.	
Referencias	RF1
Precondiciones	El usuario se ha autenticado en la <i>Intranet2</i> .
Curso normal de los eventos	
Acción del actor	Respuesta del sistema
1. El usuario accede a la <i>Intranet2</i> y se autentica.	2. El sistema muestra un bloque con los documentos de su espacio personal en <i>Alfresco</i> .

Capítulo 2: Características de la solución de integración

3. El usuario visualiza la información correspondiente a su espacio personal.	
Flujo Alternativo de los eventos	
Viene del paso 2 del flujo normal de los eventos	
Acción del actor	Respuesta del sistema
1. El usuario no tiene documentos para visualizar	2. El sistema muestra solo la información perteneciente a la <i>Intranet2</i> .

Caso de uso	
CU-2	Navegar por el repositorio
Propósito	Navegar por el repositorio <i>Alfresco</i> .
Actores	Usuario Avanzado
Resumen: El caso de uso se inicia cuando el usuario avanzado accede a la <i>Intranet2</i> y se autentica.	
Referencias	RF2

Capítulo 2: Características de la solución de integración

Precondiciones	El usuario se ha autenticado en la <i>Intranet2</i> .	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1. El usuario accede a la <i>Intranet2</i> y se autentica.	2. El sistema muestra un vínculo de acceso "Navegar por el Repositorio".	
3. El usuario accede al vínculo para navegar por la estructura de carpetas del repositorio.	4. El sistema abre una nueva página con la estructura de carpetas del repositorio.	
5. El usuario navega por la estructura de carpetas según sus permisos.		

Caso de uso	
CU-3	Establecer configuración
Propósito	Configurar la conexión de <i>Drupal</i> con <i>Alfresco</i> .
Actores	Administrador
Resumen: El caso de uso se inicia cuando el usuario selecciona la opción configurar conexión.	

Capítulo 2: Características de la solución de integración

Referencias	RF3	
Precondiciones	El usuario se ha autenticado	
Acción del actor	Respuesta del sistema	
1. El Administrador se dirige a la barra de menú en la parte superior de la página.	2. El sistema muestra un menú llamado "Configuración del sitio".	
3. Selecciona la opción "configuración de IDAM".	4. El sistema muestra la interfaz de configuración del módulo.	
5. El usuario modifica la información deseada en un formulario y pulsa el botón "Guardar configuración".	6. El sistema valida la información y actualiza la configuración. 7. Muestra un mensaje de notificación	

Caso de uso	
CU-4	Gestionar consulta
Propósito	Gestionar las consultas hechas al repositorio de documentos de <i>Alfresco</i> para la recuperación de la información.

Capítulo 2: Características de la solución de integración

Actores	Administrador
Resumen: El caso de uso se inicia cuando el usuario selecciona la opción “Gestionar consultas”.	
Referencias	RF4
Precondiciones	El usuario se ha autenticado.
Curso normal de los eventos	
Sección Crear consulta	
Acción del actor	Respuesta del sistema
1. El Administrador se dirige a la barra de menú en la parte superior de la página.	2. El sistema muestra un menú llamado “Administración de contenido”.
3. El usuario selecciona en el menú “Administración de contenido” la opción “Alfresco”.	4. El sistema muestra una página con una opción para crear una nueva consulta.
5. El usuario selecciona la opción para crear una consulta nueva.	6. El sistema muestra un formulario para entrar los datos de la consulta.
7. El usuario introduce la información necesaria y pulsa el botón “Guardar	8. El sistema valida la información y crea la consulta.

Capítulo 2: Características de la solución de integración

consulta”.	9. Muestra un mensaje de notificación.
Sección Modificar consulta	
Acción del actor	Respuesta del sistema
1. El Administrador se dirige a la barra de menú en la parte superior de la página.	2. El sistema muestra un menú llamado “Administración de contenido”.
3. El usuario selecciona en el menú “Administración de contenido” la opción <i>Alfresco</i> .	4. El sistema muestra una página con el listado de las consultas existentes.
5. El usuario pulsa el botón “Editar” correspondiente a la consulta que desea modificar.	6. El sistema carga en un formulario los datos actuales de la consulta a modificar y permite cambiarlos.
7. El usuario modifica la información deseada y pulsa el botón “Guardar consulta”.	8. El sistema valida la información y actualiza la consulta modificada. 9. Muestra un mensaje de notificación.
Sección Eliminar consulta	
Acción del actor	Respuesta del sistema

Capítulo 2: Características de la solución de integración

1. El Administrador se dirige a la barra de menú en la parte superior de la página.	2. El sistema muestra un menú llamado "Administración de contenido".
3. El usuario selecciona en el menú "Administración de contenido" la opción "Alfresco".	4. El sistema muestra una página con el listado de las consultas existentes.
5. El usuario pulsa el botón "Eliminar" correspondiente a la consulta que desea eliminar.	6. El sistema elimina la consulta.

Caso de uso	
CU-5	Exportar consulta
Propósito	Exportar la consulta como un bloque para su publicación a determinados usuarios.
Actores	Administrador
Resumen: El caso de uso se inicia cuando el usuario selecciona la opción "Gestionar consultas".	
Referencias	RF5
Precondiciones	El usuario se ha autenticado.

Capítulo 2: Características de la solución de integración

Curso normal de los eventos	
Acción del actor	Respuesta del sistema
1. El Administrador se dirige a la barra de menú en la parte superior de la página.	2. El sistema muestra un menú llamado "Administración de contenido".
3. El usuario selecciona en el menú "Administración de contenido" la opción "Alfresco".	4. El sistema muestra un listado de las consultas existentes.
5. El usuario pulsa el botón "Exportar bloque" correspondiente a la consulta que desee exportar como un bloque.	6. El sistema crea un bloque con la información obtenida a través de la consulta y exporta el bloque para su publicación posterior.

Caso de uso	
CU-6	Listar consultas
Propósito	Listar las consultas existentes.
Actores	Administrador
Resumen: El caso de uso se inicia cuando el usuario selecciona la opción "Gestionar consultas".	

Capítulo 2: Características de la solución de integración

Referencias	RF6
Precondiciones	El usuario se ha autenticado.
Curso normal de los eventos	
Sección Crear consulta	
Acción del actor	Respuesta del sistema
1. El Administrador se dirige a la barra de menú en la parte superior de la página.	2. El sistema muestra un menú llamado "Administración de contenido".
3. El usuario selecciona en el menú "Administración de contenido" la opción "Alfresco".	4. El sistema muestra una página con el listado de las consultas existentes.

Caso de uso	
CU-7	Visualizar información del grupo
Propósito	Visualizar los documentos del grupo que le corresponde
Actores	Profesor

Capítulo 2: Características de la solución de integración

Resumen: El caso de uso se inicia cuando el usuario accede a la <i>Intranet2</i> y se autentica.	
Referencias	RF7
Precondiciones	El usuario se ha autenticado en la <i>Intranet2</i> .
Curso normal de los eventos	
Acción del actor	Respuesta del sistema
1. El usuario accede a la <i>Intranet2</i> y se autentica.	2. El sistema muestra un bloque con los documentos del grupo de quien es guía.
3. El usuario visualiza la información correspondiente a su grupo.	

En este capítulo se llevaron a cabo una serie de tareas de vital importancia para la realización del módulo IDAM que permitirá la integración de *Drupal* con *Alfresco*. A partir de su análisis se llegó a la conclusión de hacer un modelo de dominio y no de negocio debido a que los flujos de información se encontraban difusos y presentaban bajo nivel de estructuración. Posteriormente se diseñó el diagrama para dicho modelo, así como las relaciones existentes entre cada objeto. Se extrajeron los requisitos funcionales y no funcionales de la aplicación y se hizo un estudio de las capacidades y funcionalidades que el sistema debía tener.

Luego se pasó a la selección de los casos de uso a partir de los requerimientos funcionales y los actores que participaban en su ejecución para llegar así al diagrama de casos de uso del sistema donde se visualiza la relación actor- actor y la interacción actor- caso de uso.

Capítulo 3: Análisis y diseño del sistema

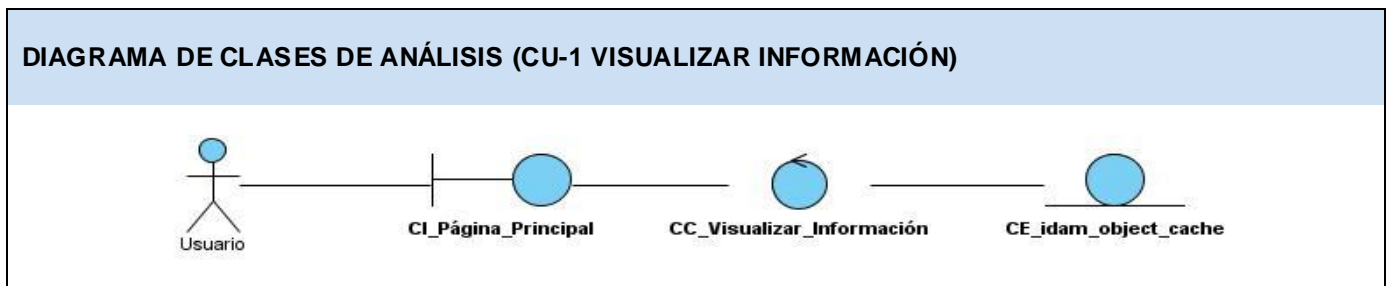
En el presente capítulo se desarrolla el flujo de trabajo Análisis y Diseño del sistema. Se presenta el modelo de clases de análisis. Se muestran los diagramas de secuencia por cada realización de casos de uso, así como el diagrama de clases del diseño. Además se realiza la descripción de las clases, se diseña la base de datos y se describen las tablas.

3.1 Modelo de Análisis

El modelo de análisis es una aproximación al modelo del diseño. En el mismo existe un refinamiento de los requisitos, sin embargo no se tiene en cuenta el lenguaje de programación que se va a utilizar en la construcción de la aplicación, debido a que el objetivo del análisis es comprender perfectamente los requisitos del *software* y no precisar cómo se implementará la solución.

A continuación se refleja el diagrama de clases de análisis correspondiente a los casos de usos descritos.

3.1.1 Diagrama de clases de análisis



Capitulo 3: Análisis y diseño del sistema propuesto

DIAGRAMA DE CLASES DE ANÁLISIS (CU-2 NAVEGAR POR EL REPOSITORIO)

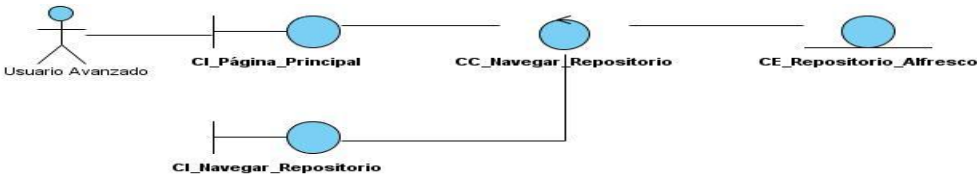


DIAGRAMA DE CLASES DE ANÁLISIS (CU-3 ESTABLECER CONFIGURACIÓN)

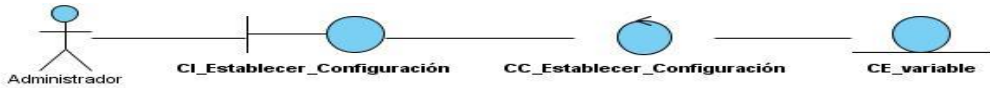
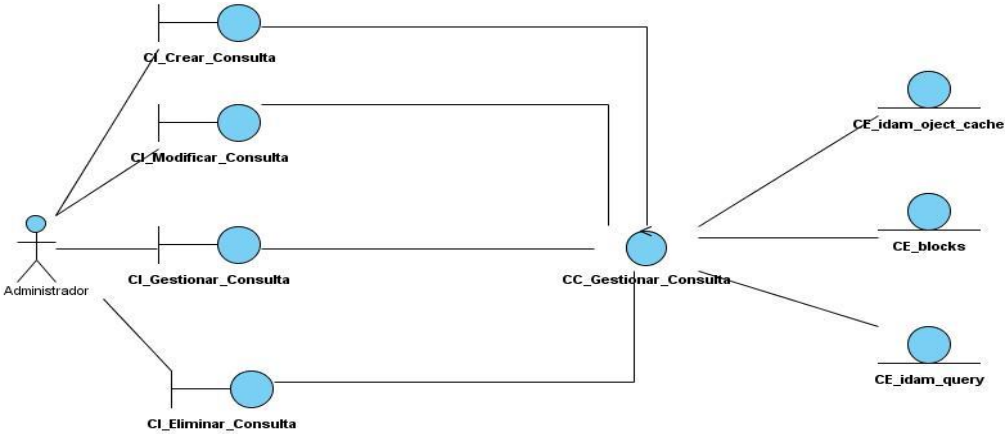
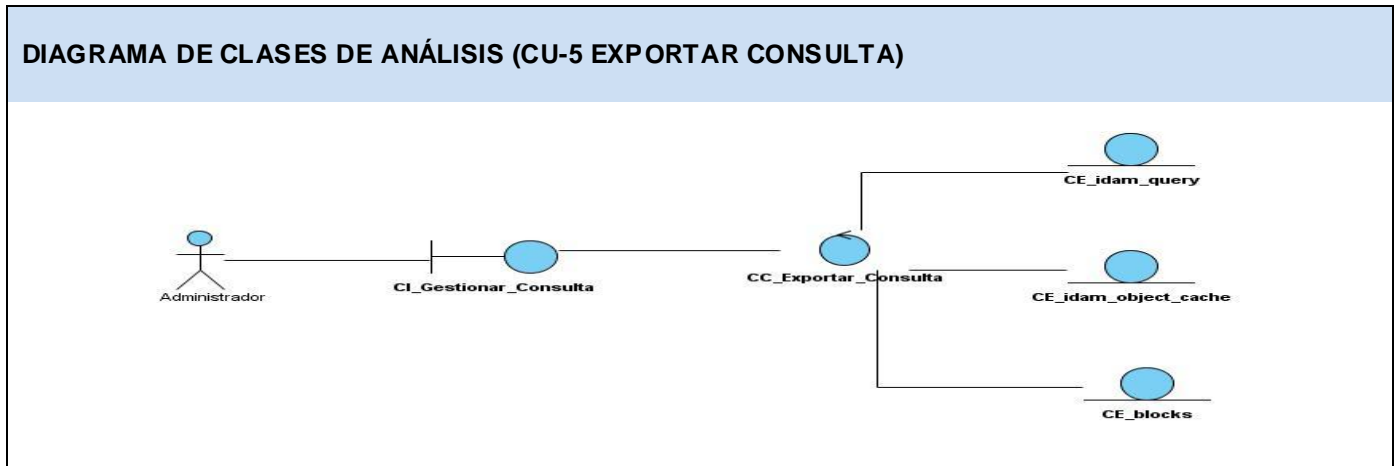


DIAGRAMA DE CLASES DE ANÁLISIS (CU-4 GESTIONAR CONSULTA)





3.2 Diseño

El diseño es la etapa del proceso de desarrollo donde se decide como se llevará a cabo el sistema. A través de esta fase se toman decisiones estratégicas y tácticas para cumplir los requerimientos funcionales y de calidad de un sistema. Se adquiere además, una comprensión en profundidad de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de interfaz de usuario, entre otros.

A pesar de que en *Drupal* no existen clases declaradas explícitamente, su diseño hace uso de varios elementos de la Programación Orientada a Objetos (POO), cabe citar: la herencia, el polimorfismo, el encapsulamiento, entre otros elementos y características que lo hacen OO. Es por ello que el funcionamiento de *Drupal* hace uso de patrones de diseños característicos de sistemas OO, particularmente los patrones GOF (*Gang Of Four*), los cuales se dividen en varias categorías entre las que se encuentran: creacionales, estructurales y de comportamiento y dentro de estas los patrones que se manifiestan específicamente en el funcionamiento de *Drupal* son (43) (44):

Singleton (instancia única): el cual pretende garantizar que una clase tenga una sola instancia y proporcionar un punto de acceso global a ella.

Capítulo 3: Análisis y diseño del sistema propuesto

Decorator (envoltorio): el cual permite añadir dinámicamente funcionalidades a un objeto, posibilitando no tener que crear sucesivas clases que hereden de la primera incorporando la nueva funcionalidad, sino otras que la implementan y se asocian a la primera. *Drupal* hace un uso extensivo de este patrón, pudiendo citar el uso de *hook_nodeapi* que permite a cualquier nodo extender el comportamiento de todos los nodos.

Observer (observador): este define una dependencia del tipo uno-a-muchos entre objetos, de manera que cuando uno de los objetos cambie su estado, el observador se encarga de notificar este cambio a todos los otros dependientes. Este también es muy utilizado en *Drupal*, por ejemplo, al implementar *hook*, estos se han definido como observadores del objeto vocabulario, cualquier cambio que se realice es notificado y actualizado en los dependientes.

También se utilizan otros patrones de diseño como son el *Bridge* (puente) que desacopla una abstracción de su implementación y el *Chain of Responsibility* (cadena de responsabilidad) el cual permite establecer la línea que deben llevar los mensajes para que los objetos realicen la actividad indicada.

El uso de todos estos patrones le confieren al diseño de *Drupal* una amplia flexibilidad y extensibilidad, contribuyendo a que funcione similar a un sistema OO.

Para el desarrollo del módulo IDAM, se hará uso del patrón de arquitectura MVC conocido como Modelo Vista Controlador. El MVC separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos, se ve frecuentemente en aplicaciones *Web*, donde la vista es la página HTML y el código que provee de datos dinámicos a la página; el modelo es el Sistema de Gestión de Base de Datos y la lógica de negocio; y el controlador es el responsable de recibir los eventos de entrada desde la vista.

En la figura 2 se muestra como está estructurado en capas el CMS *Drupal*, donde la vista son los “Temas”, “Motores de Plantilla”, “JavaScript/Ajax”; el modelo es la base de datos (BD) y el controlador es el “Núcleo de *Drupal*” y los “Módulos”: donde los “Ganchos” son las funciones para facilitar la comunicación entre el Núcleo de *Drupal* y los módulos (40).

Capítulo 3: Análisis y diseño del sistema propuesto

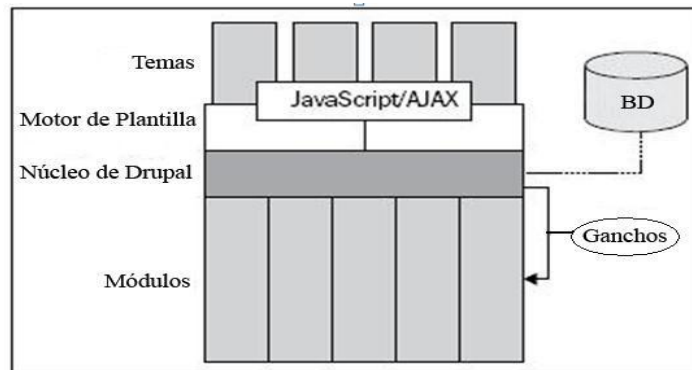
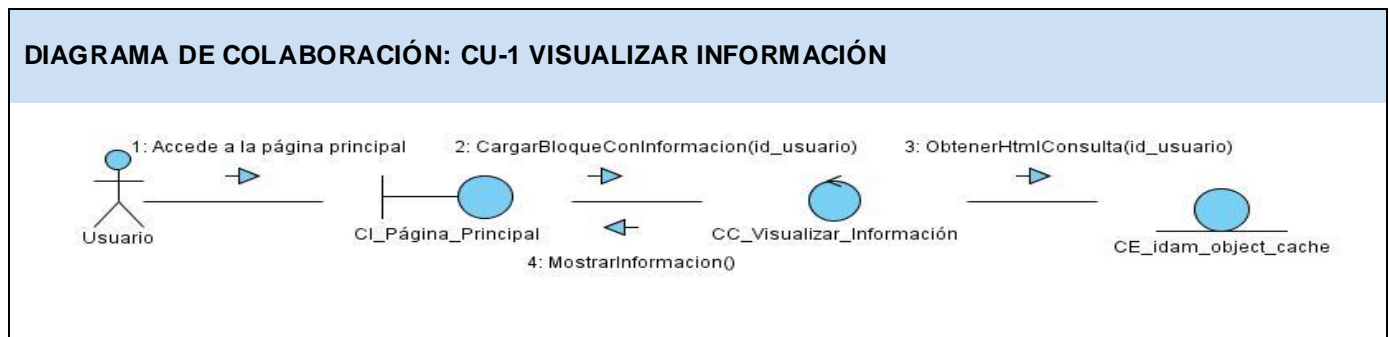


Figura 2. Estructura en capas de Drupal

3.2.1 Diagramas de Interacción (Colaboración)

Un diagrama de interacción no es más que la representación de secuencias de intercambios y mensajes entre los roles que participan y se relacionan con un sistema. Representa lo que el usuario hace/demanda/entrega y lo que el sistema arroja/ofrece/responde.



Capítulo 3: Análisis y diseño del sistema propuesto

DIAGRAMA DE COLABORACIÓN: CU-2 NAVEGAR POR EL REPOSITORIO

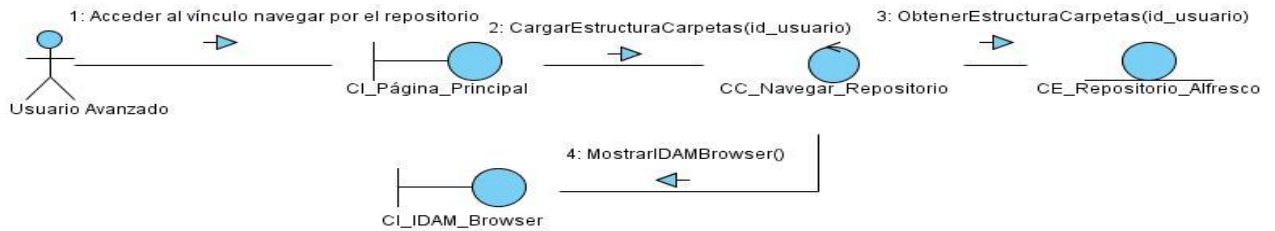
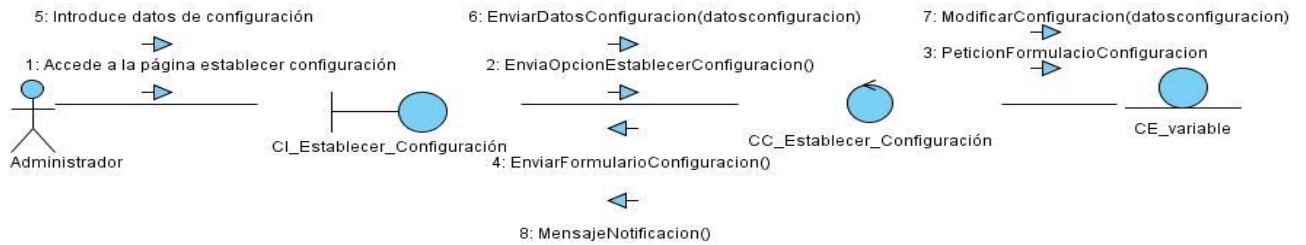


DIAGRAMA DE COLABORACIÓN: CU-3 ESTABLECER CONFIGURACIÓN



Capítulo 3: Análisis y diseño del sistema propuesto

DIAGRAMA DE COLABORACIÓN: CU- 4 GESTIONAR CONSULTA

SECCIÓN CREAR CONSULTA

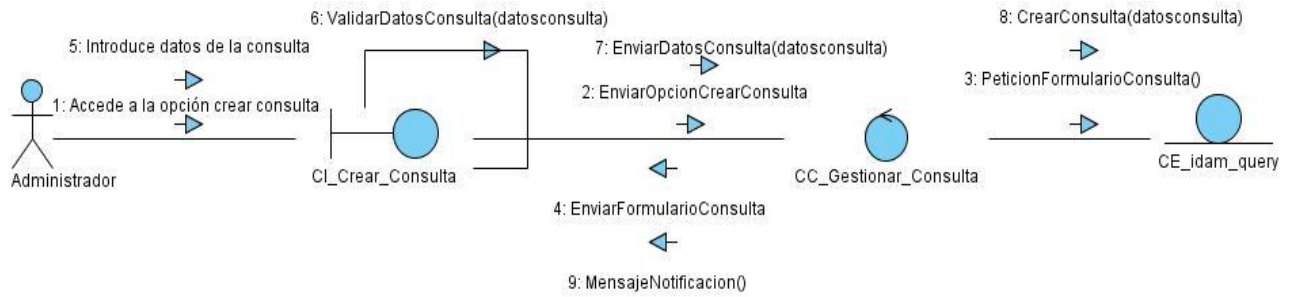
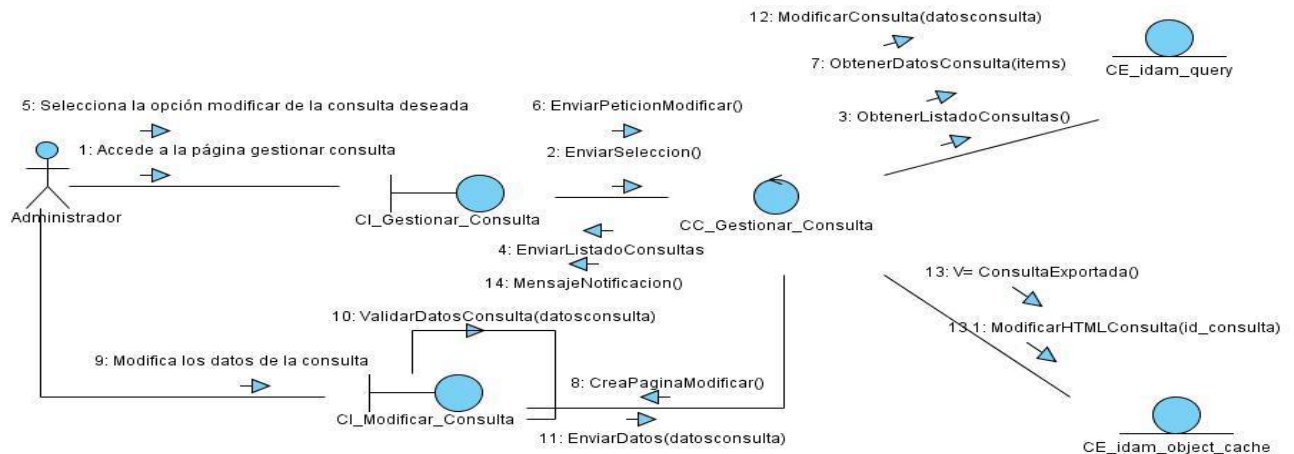


DIAGRAMA DE COLABORACIÓN: CU- 4 GESTIONAR CONSULTA

SECCIÓN MODIFICAR CONSULTA



Capítulo 3: Análisis y diseño del sistema propuesto

DIAGRAMA DE COLABORACIÓN: CU- 4 GESTIONAR CONSULTA

SECCIÓN ELIMINAR CONSULTA

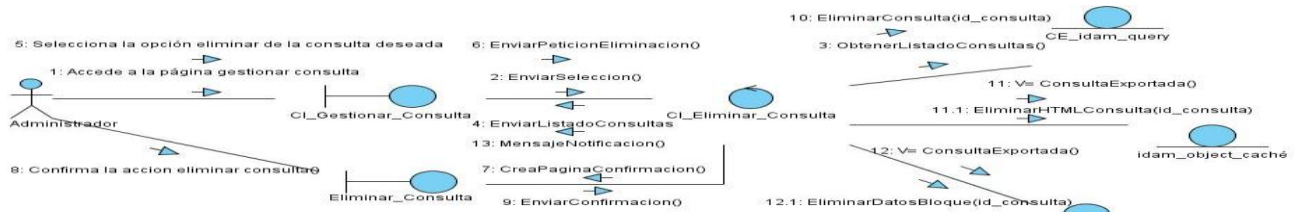
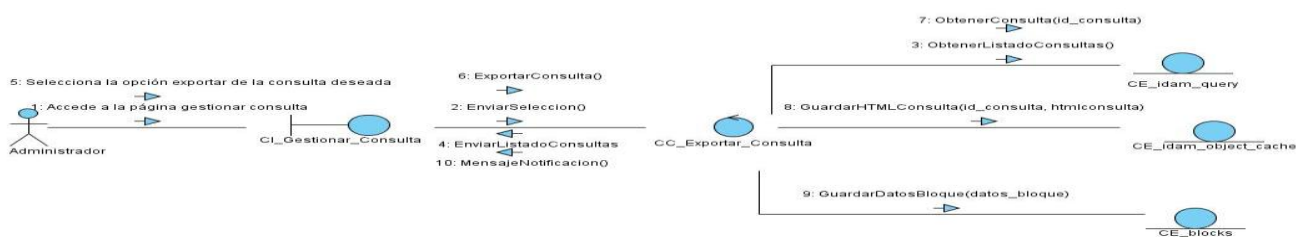


DIAGRAMA DE COLABORACIÓN: CU-5 EXPORTAR CONSULTA



3.2.2 Diagramas de clases del diseño

Para la comprensión de los diagramas de diseño *Web* se modela primeramente el diagrama general de *Drupal* y luego los diagramas correspondientes al módulo IDAM. El diagrama general muestra el funcionamiento del CMS *Drupal* con todas sus clases y módulos.

Este CMS depende de un grupo de clases o módulos que permiten a los usuarios finales interactuar con la aplicación. Dada su modularidad e independencia estructural se dividen en módulos por separados de acuerdo a sus funcionalidades los cuales se modelan en el diagrama como paquetes:

Capítulo 3: Análisis y diseño del sistema propuesto

Modules: Incluye todos los módulos, que permiten las distintas funcionalidades del CMS.

Includes: Contiene un conjunto de ficheros indispensables para el funcionamiento, como son las conexiones a base datos del sistema y APIs generales del CMS utilizadas por los módulos.

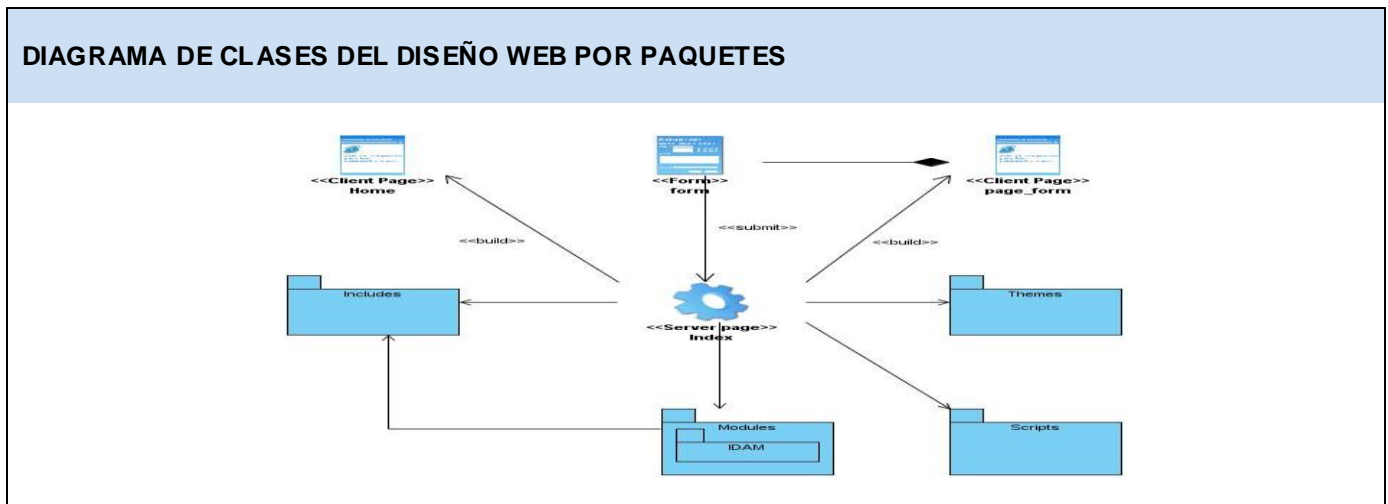
Themes: Incluye el motor que genera la presentación o interfaz del CMS e incluye las distintas plantillas de temas.

Misc: Incluye funcionalidades necesarias del CMS, como HTML y *JavaScript* utilizados en detalles básicos de la aplicación.

IDAM: Como módulo al fin se incluye dentro del paquete *modules* para habilitar su funcionamiento.

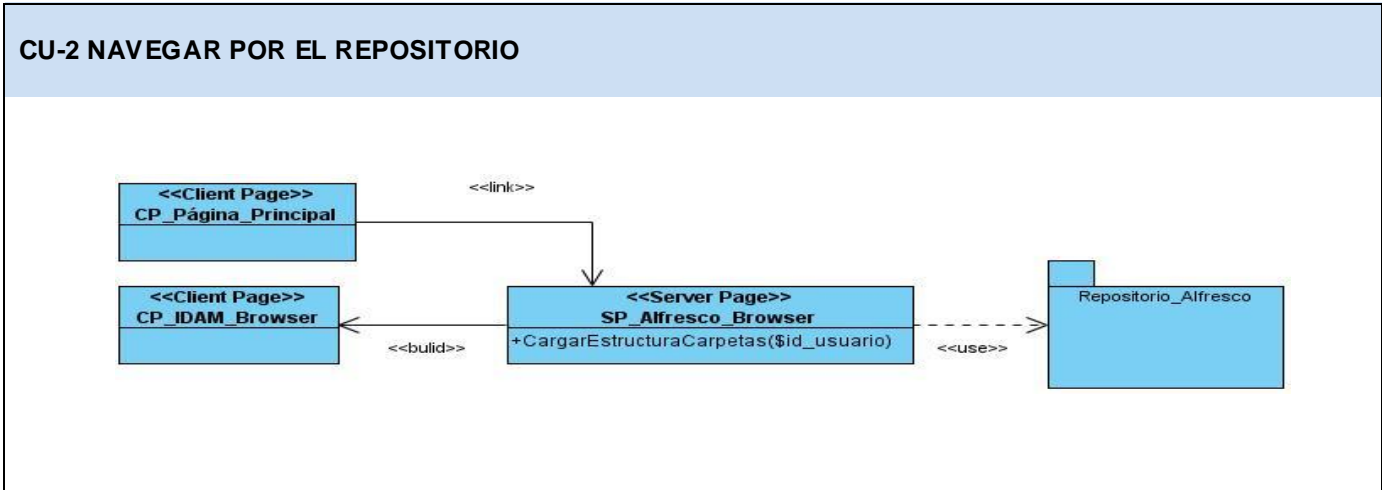
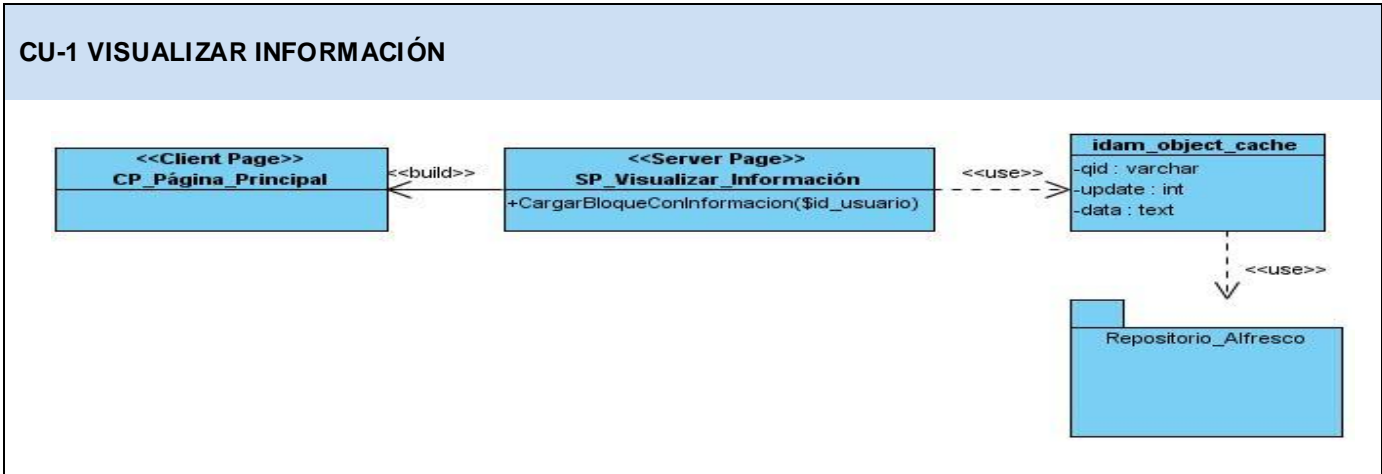
Las páginas que genera *Drupal* pueden tener o no formularios, en dependencia de lo que se desee.

Es importante señalar que este CMS contiene una única página servidora que a través de funciones o ganchos (*hook*) se conecta con los módulos, a su vez los módulos se conectan a la base datos a través de los *includes*", para finalmente mostrar la información requerida.

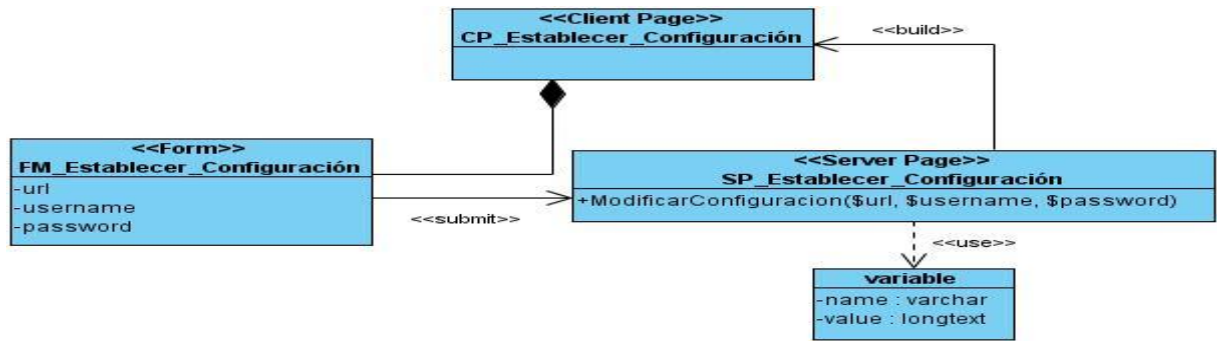


Visto el diagrama de paquetes de la figura anterior, ahora se muestran los diagramas de clases del diseño para las funcionalidades correspondientes al módulo IDAM.

Capitulo 3: Análisis y diseño del sistema propuesto

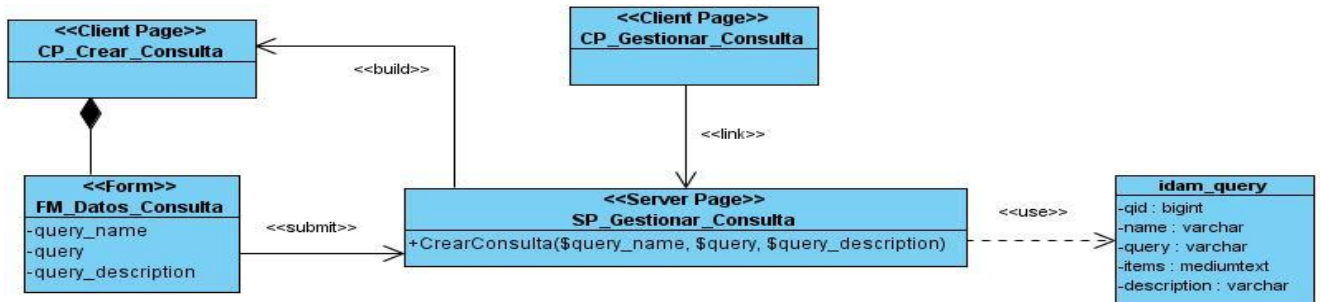


CU-3 ESTABLECER CONFIGURACIÓN



CU-4 GESTIONAR CONSULTA

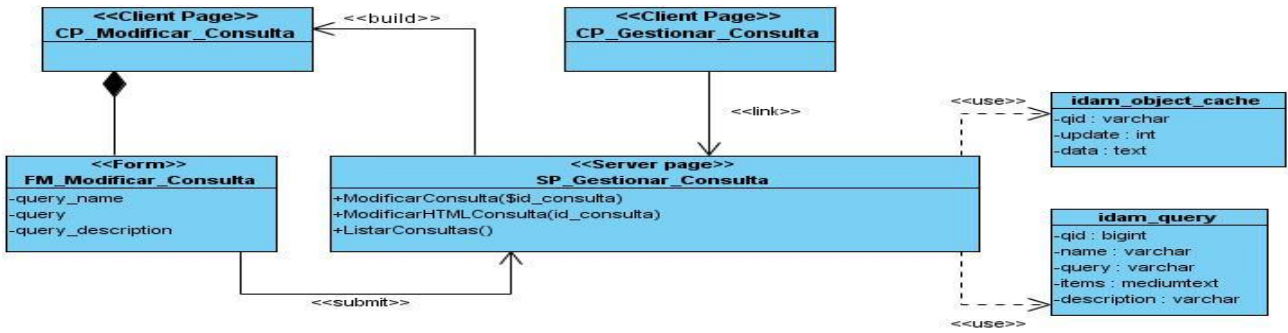
SECCIÓN CREAR CONSULTA



Capítulo 3: Análisis y diseño del sistema propuesto

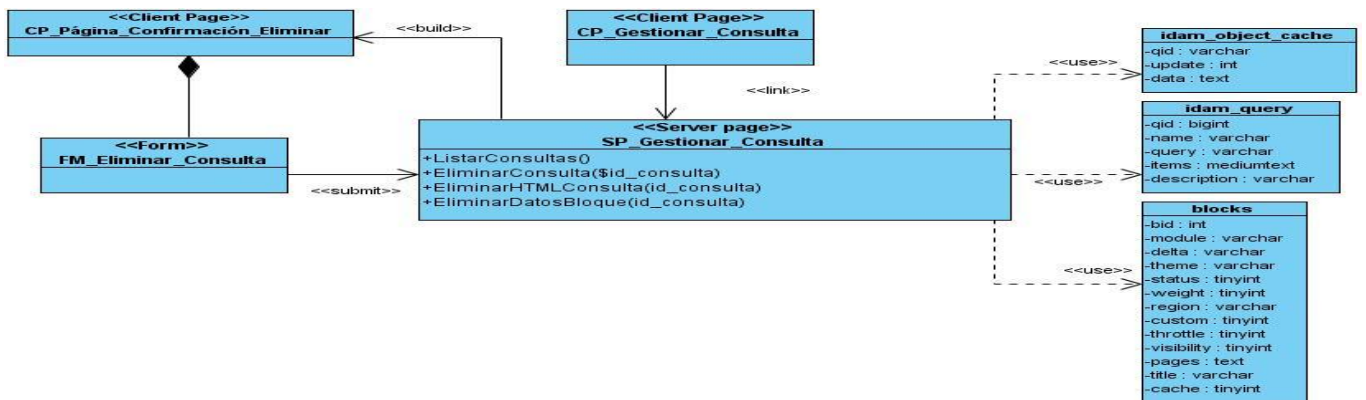
CU-4 GESTIONAR CONSULTA

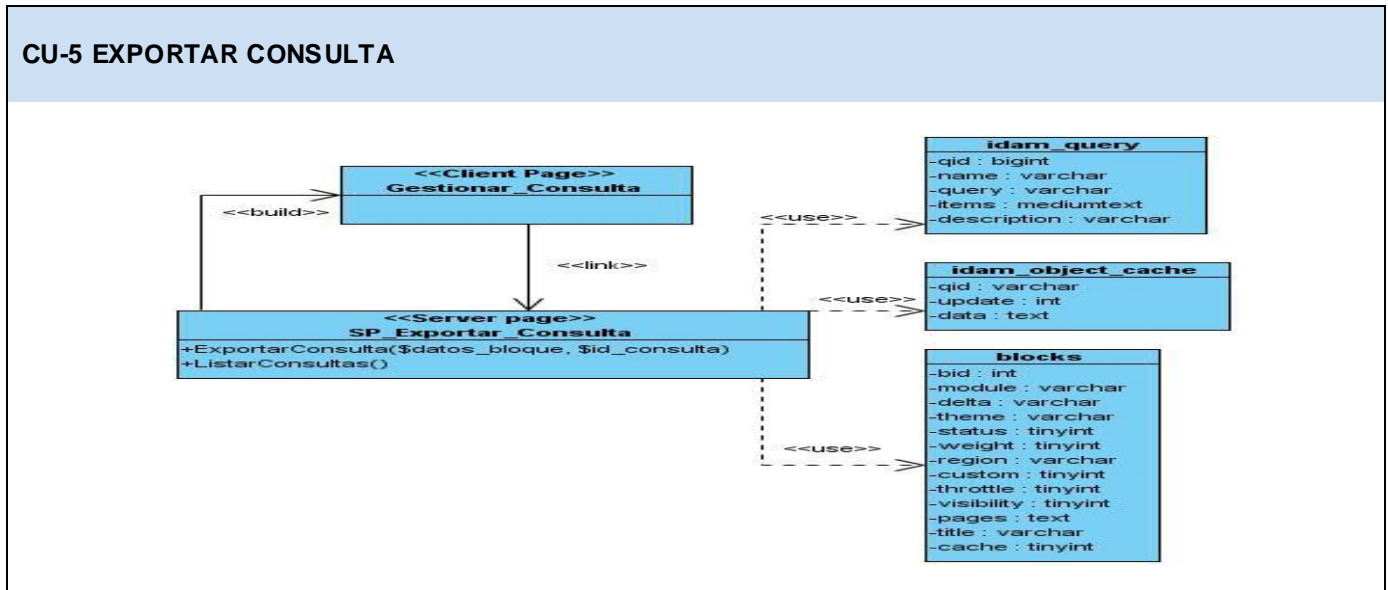
SECCIÓN MODIFICAR CONSULTA



CU-4 GESTIONAR CONSULTA

SECCIÓN ELIMINAR CONSULTA



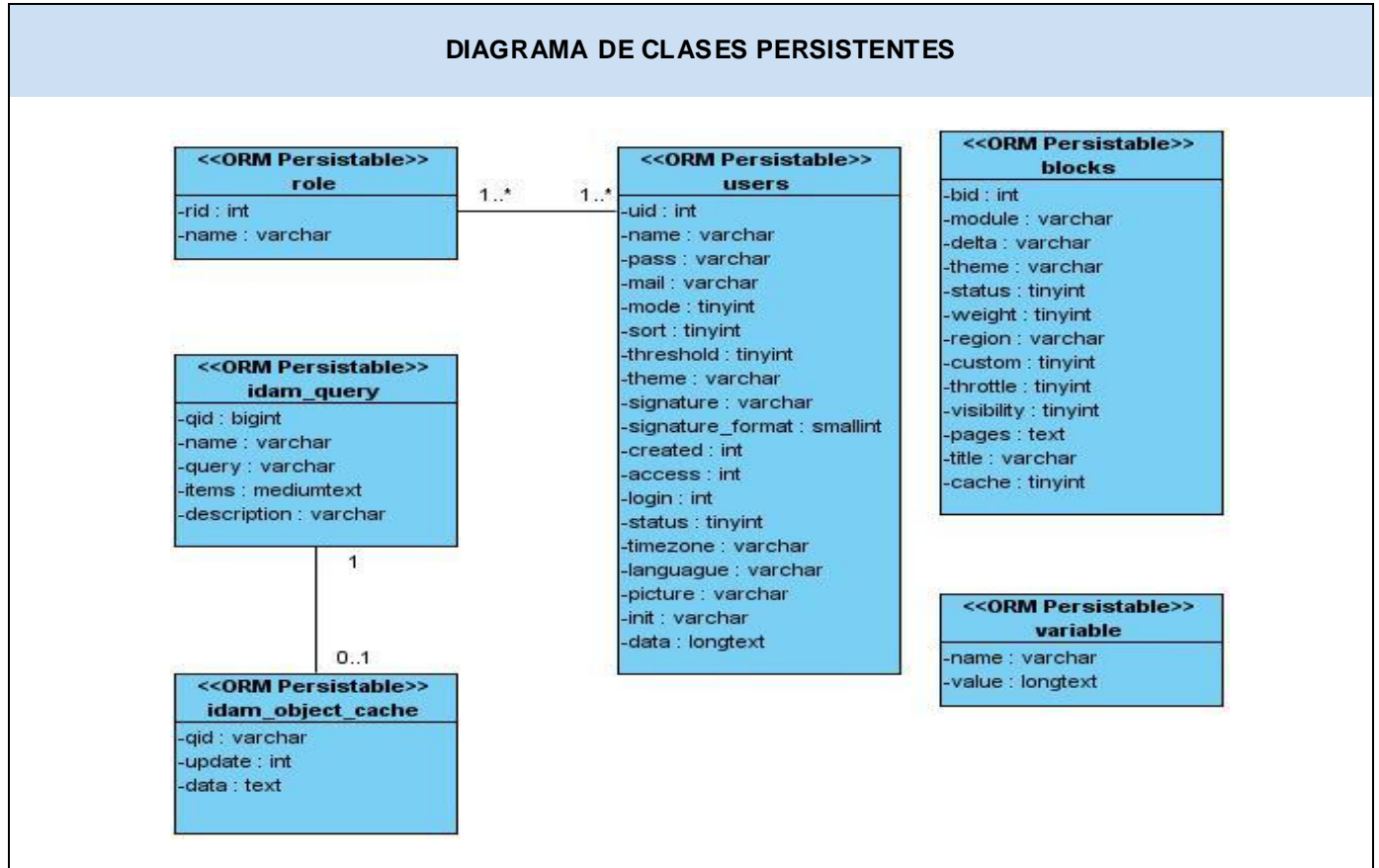


3.3 Diseño de la Base de Datos

El objetivo principal del diseño de una base de datos es generar tablas que modelan los registros en los que se guarda la información del sistema. Es importante que esta información se almacene sin redundancia, para que se pueda tener una recuperación rápida y eficiente de los datos.

A continuación se presenta el diagrama de clases persistentes y el modelo de datos que darán solución a esta parte del diseño.

3.3.1 Diagrama de clases persistentes

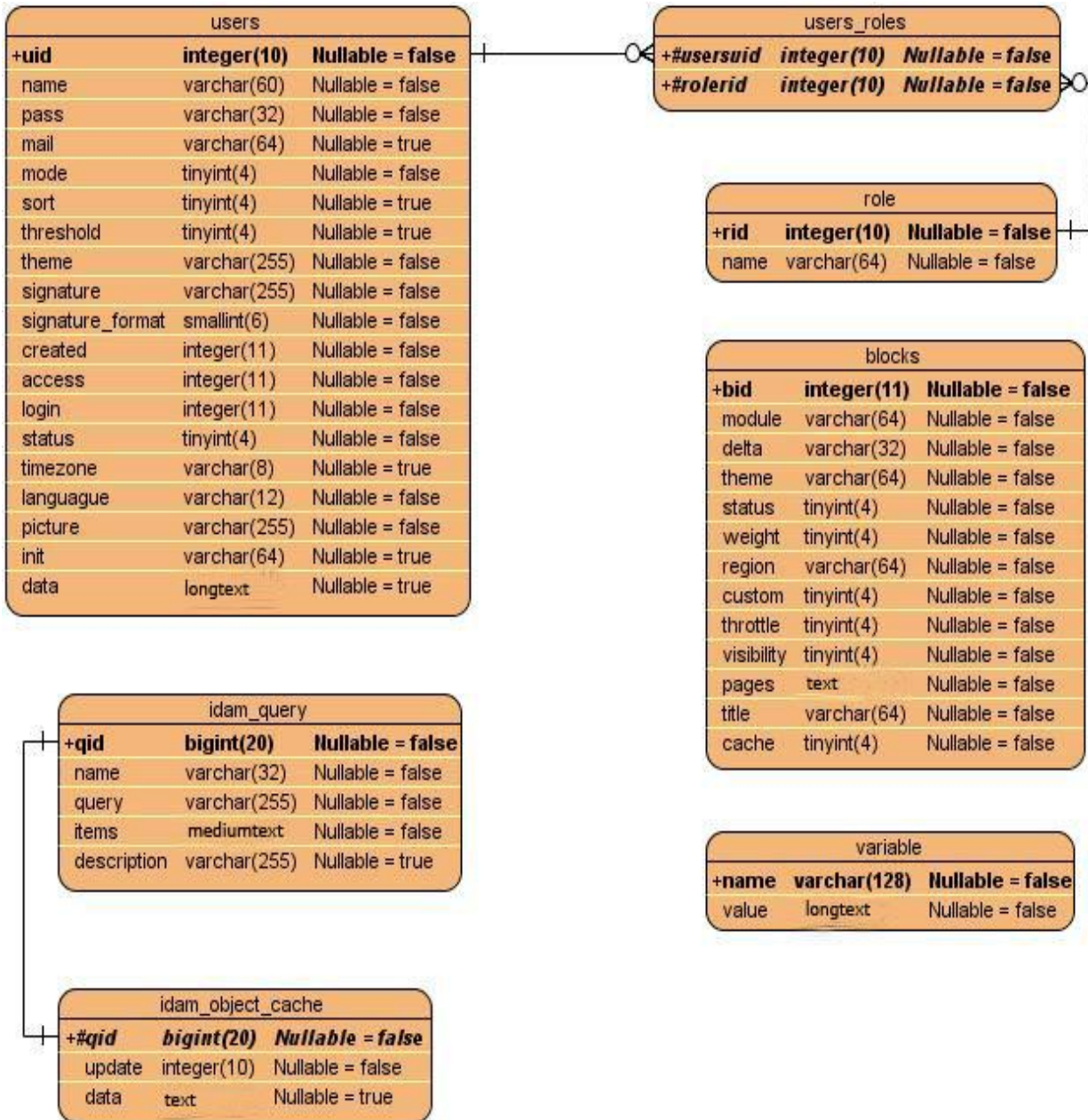


3.3.2 Modelo de datos

A partir del diagrama de clases persistentes se obtuvo el modelo de datos que se muestra a continuación.

Capítulo 3: Análisis y diseño del sistema propuesto

MODELO DE DATOS



Capítulo 3: Análisis y diseño del sistema propuesto

3.3.3 Descripción de las tablas

A continuación se describen cada una de de las tablas de la base de datos.

Nombre: idam_query		
Descripción: Almacena la información referente a las consultas del módulo IDAM.		
Atributo	Tipo	Descripción
qid	bigint	Identificador de la tabla idam_query.
Name	varchar	Nombre de la consulta.
Query	varchar	Contenido de la consulta.
Ítems	mediumtext	Campos por los que se filtró la consulta.
Description	varchar	Descripción de la consulta.

Nombre: idam_object_cache		
Descripción: Almacena la información referente al HTML de las consultas que han sido exportadas.		
Atributo	Tipo	Descripción
Qid	varchar	Identificador de la consulta.

Capítulo 3: Análisis y diseño del sistema propuesto

Update	int	Fecha de la última vez que se actualizaron los datos de la consulta.
Data	text	HTML de la consulta que se genera.

Con el análisis del sistema, se logró un refinamiento y mejor comprensión de los requisitos, así como una aproximación al modelo de diseño, realizando los diagramas de clases del análisis.

La propuesta del diseño permitió modelar el sistema que se pretende construir, de manera que soporte todos los requisitos. Además se realizó el diagrama de clases de diseño *Web*, logrando así definir las relaciones entre las interfaces, las clases, los paquetes y la base de datos.

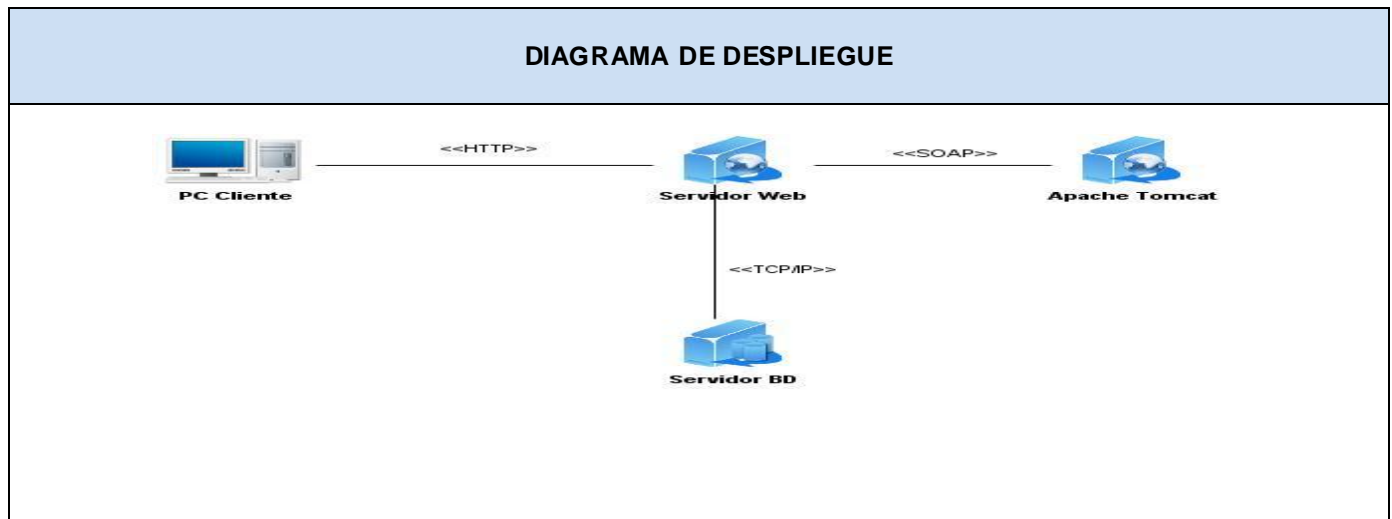
Para garantizar el almacenamiento de la información del sistema se realizó el diseño de la base de datos, a partir de una identificación de las clases persistentes expresadas en su diagrama correspondiente y la generación del modelo de datos.

Capítulo 4: Implementación y prueba

El presente capítulo aborda los flujos de trabajo de implementación y prueba. Se presenta cómo se distribuirá físicamente la aplicación desarrollada a través del modelo de despliegue. Se realiza además el diagrama de componentes para mostrar la estructura de alto nivel del modelo de implementación. Finalmente se desarrollan las pruebas correspondientes para garantizar la calidad requerida en el producto final.

4.1 Diagrama de despliegue

El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema y las funcionalidades entre los nodos de cómputo. Es una colección de nodos y arcos; donde cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar. El propósito de este modelo es capturar la configuración de los elementos de procesamiento y las conexiones entre estos elementos en el sistema.



A continuación se describen cada uno de los nodos presentes en el diagrama anterior y la comunicación entre ellos.

Capítulo 4: Implementación y prueba

PC Cliente: Representa la computadora del cliente que se conecta al **Servidor Web** mediante el protocolo HTTP.

Servidor Web: Representa el servidor sobre el cual se encuentra desplegada la aplicación *Web*. Este accede al **Servidor BD** para el manejo de información mediante el protocolo TCP/IP.

Apache Tomcat: Servidor donde se aloja el ECM *Alfresco*.

Servidor BD: Es la representación del servidor de base de datos en el cual los datos pertenecientes a la aplicación son almacenados.

4.2 Diagrama de componentes

El diagrama de componentes define cómo las clases, artefactos y otros elementos de bajo nivel, se unen para formar componentes de alto nivel y las conexiones entre ellos. Los componentes son artefactos de *software* compilados que trabajan acoplados para brindar el comportamiento requerido dentro de las restricciones definidas en el proceso de captura de requisitos.

A continuación se representan los diagramas de componentes del sistema desarrollado. El primer diagrama es el general para *Drupal* donde se describe cada componente. Posteriormente se describen los diagramas pertenecientes al módulo IDAM.

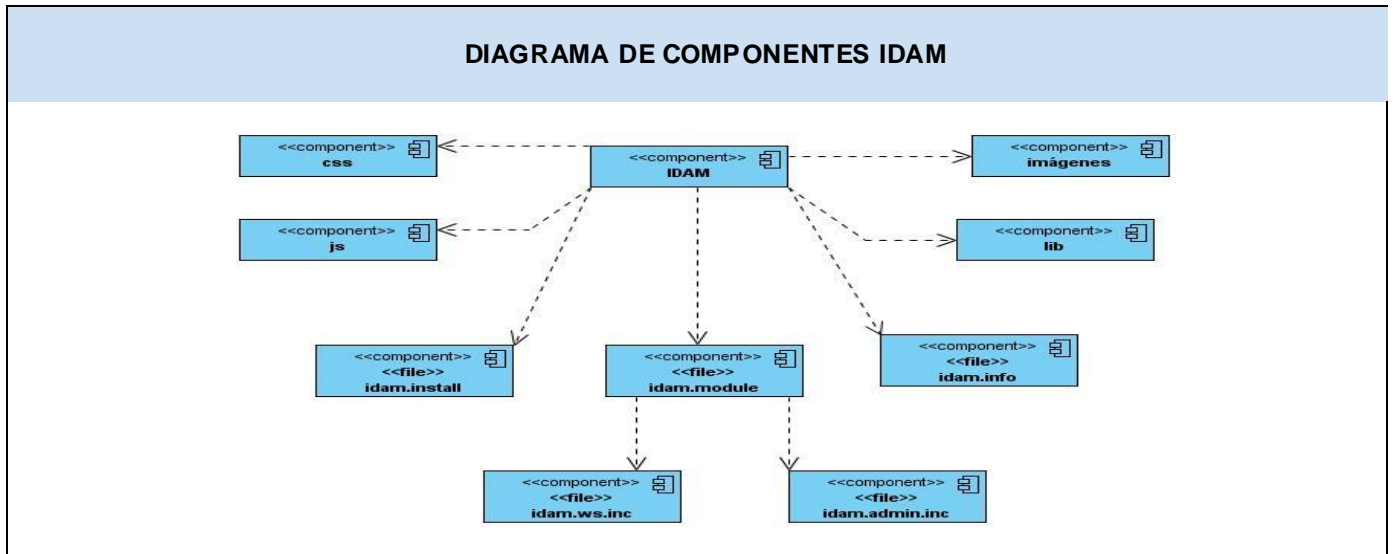
El módulo IDAM requiere de los componentes que a continuación se describen.

Componente	Propósito
Css	Contiene el fichero CSS correspondiente al módulo IDAM.
Js	Componente que contiene los ficheros js que serán utilizados en el módulo IDAM.

Capítulo 4: Implementación y prueba

Imágenes	Componente que contiene las imágenes que va a tener el módulo.
Lib	Componente de librerías para la conexión con <i>Alfresco</i> .
idam.install	Define las tablas de la base de datos que va a tener el módulo.
idam.module	Define los <i>hooks</i> de <i>Drupal</i> . Tiene las funciones de crear, eliminar y modificar las consultas. Brinda la interfaz de conexión para la configuración de <i>Alfresco</i> .
idam.info	Define la información referente al módulo como por ejemplo: nombre del módulo, descripción, núcleo, versión de php, dependencias etc.
idam.ws.inc	Archivo incluido en <i>idam.module</i> . Brinda la capa de conexión con <i>Alfresco</i> . Incluye las librerías y crea la sección.
idam.admin.inc	Archivo incluido en <i>idam.module</i> . Define los formularios del módulo y su lógica.

Capítulo 4: Implementación y prueba

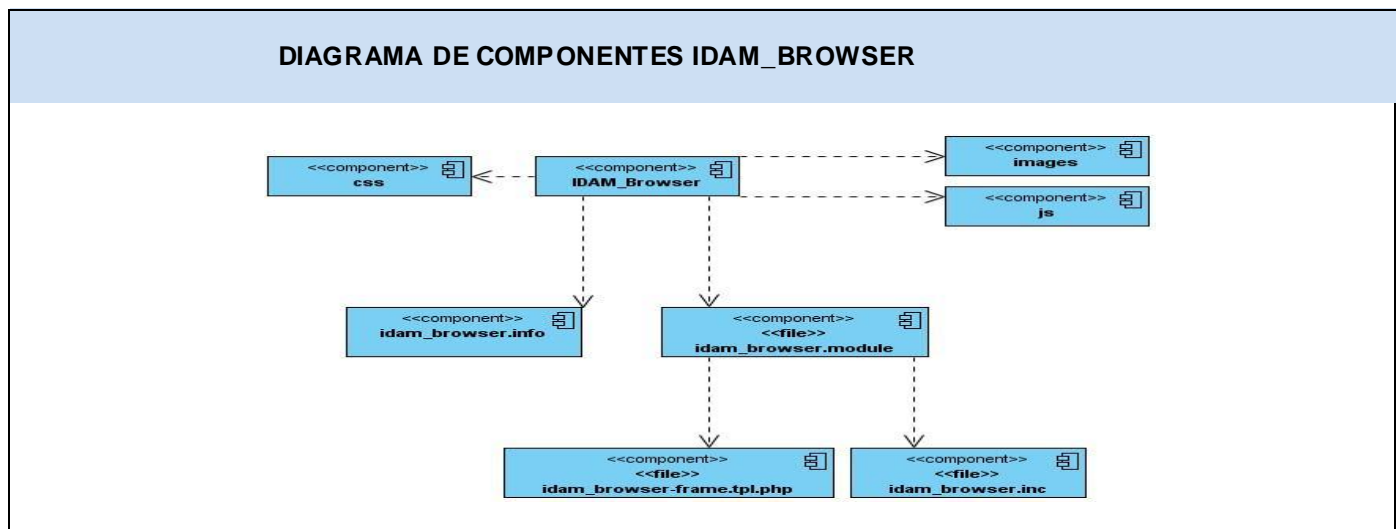


El módulo *IDAM_Browser* se encuentra dentro del módulo IDAM, el mismo proporciona el servicio correspondiente a acceder a la estructura de carpetas del repositorio de documentos de *Alfresco*. Este módulo depende, para su funcionamiento, del módulo IDAM y viceversa.

Componente	Propósito
Css	Contiene el fichero CSS correspondiente al módulo <i>IDAM_Browser</i> .
Js	Componente que contiene los ficheros js que serán utilizados en el módulo <i>IDAM_Browser</i> .
Imágenes	Componente que contiene las imágenes que va a tener el módulo <i>IDAM_Browser</i> .
idam_browser.info	Define la información referente al módulo como por ejemplo: nombre del módulo, descripción, núcleo, versión de php,

Capítulo 4: Implementación y prueba

	dependencias etc.
idam_browser.module	Define los <i>hooks</i> que va a tener el módulo <i>IDAM_Browser</i> .
idam_browser-frame.tpl.php	Define la plantilla del <i>IDAM_Browser</i> .
idam_browser.inc	Va incluido dentro del fichero <i>idam_browser.module</i> . Describe la lógica del módulo. Define la estructura del directorio que necesita el <i>plugin</i> .



4.3 Prueba

El desarrollo del *software* implica una serie de actividades de producción en las que las posibilidades de que aparezcan errores en el mismo son comunes. Los errores pueden comenzar desde el momento del proceso en que los objetivos son asimilados erróneamente. Por tales motivos el desarrollo del *software* debe ir acompañado por una actividad que garantice la calidad.

Capítulo 4: Implementación y prueba

Las pruebas son una actividad en la cual un sistema o componente es ejecutado bajo ciertas condiciones, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente.

La prueba de *software* garantiza la calidad del mismo y representa una revisión final de las especificaciones del diseño y de la codificación.

Para el caso del módulo IDA se realizaron pruebas de Caja Negra.

La prueba de Caja Negra se centra principalmente en los requisitos funcionales del software. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos.

4.3.1 Modelo de prueba

Nombre del caso de uso: Establecer Configuración

Entrada	Respuesta del Sistema	Resultados
Campos de configuración vacíos.	El sistema emite un mensaje de error para que el usuario no deje ningún campo vacío.	Satisfactorio.
El administrador introduce los datos de usuario y contraseña incorrectos.	El sistema muestra un mensaje de error haciéndole saber al usuario que los datos de autenticación están incorrectos.	Satisfactorio.

Capítulo 4: Implementación y prueba

El administrador introduce la configuración correcta.	El sistema muestra un mensaje de notificación para confirmar que la configuración se estableció con éxito.	Satisfactorio.
-------------------------------------------------------	------------------------------------------------------------------------------------------------------------	----------------

Nombre del caso de uso: Gestionar Consulta-Sección Crear Consulta

Entrada	Respuesta del Sistema	Resultados
El administrador crea la consulta sin filtros	El sistema muestra un mensaje de error indicando que la consulta debe tener al menos un filtro.	Satisfactorio.
Campos de los datos de la consulta vacíos.	El sistema emite un mensaje de error para que el usuario no deje ningún campo vacío.	Satisfactorio.
El administrador crea la consulta correctamente.	El sistema muestra un mensaje para confirmar que la consulta fue creada.	Satisfactorio.

Nombre del caso de uso: Gestionar Consulta-Sección Modificar Consulta

Entrada	Respuesta del Sistema	Resultados
El administrador selecciona la opción modificar consulta.	El sistema muestra la consulta a modificar con los datos anteriores.	Satisfactorio.
Campos de los datos de la	El sistema emite un mensaje de error para	Satisfactorio.

Capítulo 4: Implementación y prueba

consulta vacíos.	que el usuario no deje ningún campo vacío.	
El administrador modifica la consulta correctamente.	El sistema muestra un mensaje para confirmar que la consulta fue modificada.	Satisfactorio.

Nombre del caso de uso: Listar consulta

Entrada	Respuesta del Sistema	Resultados
El administrador selecciona la opción "Alfresco" en el menú superior para listar las consultas.	El sistema muestra la interfaz correspondiente con el listado de las consultas existentes.	Satisfactorio.

En este capítulo se realizaron las tareas de implementación necesarias para lograr un módulo funcional. Se realizó el modelo de despliegue, estableciendo la distribución física del sistema en tiempo de ejecución y las funcionalidades de cada nodo de cómputo que lo compone. Además se confeccionó el diagrama de componentes para definir cada uno de los elementos físicos que conforman el módulo y las relaciones entre cada uno de ellos. Finalmente se describieron los casos de pruebas correspondientes a la aplicación que juegan un papel primordial ya que proporcionan al cliente conformidad y seguridad ante el sistema para así lograr una mayor satisfacción de este con el producto final.

Conclusiones

Se diseñó e implementó un módulo para *Drupal* que es capaz de integrarse con *Alfresco* para que los usuarios puedan, desde la *Intranet2*, acceder a los documentos almacenados en el repositorio de este potente gestor documental.

La aplicación se desarrolló utilizando la metodología RUP, y apoyándose en herramientas libres las cuales tienen un valor agregado por lo que representa para la soberanía tecnológica a la que aspira Cuba en la rama de las Tecnologías de la Información y las Comunicaciones (TIC).

El estudio de los módulos realizados por la comunidad de *Drupal*, que permiten la integración de este CMS con *Alfresco*, exigió que el módulo desarrollado estableciera una nueva lógica de almacenamiento de acuerdo a las necesidades y exigencias del entorno para el cual se creó.

La realización de este módulo brinda un importante servicio para los usuarios de la universidad contribuyendo al propósito fundamental de la *Intranet2* que es el de ofrecer la integración de la mayor cantidad de servicios existentes en la UCI.

Por todo lo anteriormente expuesto se concluye que los objetivos trazados para el presente trabajo se han cumplido satisfactoriamente.

Recomendaciones

Los objetivos de este trabajo han sido logrados, teniendo en cuenta que se cumplieron todos los requerimientos planteados. No obstante se hacen las siguientes recomendaciones:

- Continuar desarrollando el módulo para implementar nuevas funcionalidades y mejorar su calidad y robustez.
- Que el producto sea incorporado en el sistema *Intranet 2.0*, a fin de explotar sus potencialidades.
- Se tengan en cuenta la implementación nuevas funcionalidades como:
 - ✓ Gestionar los contenidos de *Alfresco* desde Drupal.
 - ✓ Permitir a los usuarios subir archivos a su espacio de trabajo personal. Evitando así tener que utilizar terceras aplicaciones.

Bibliografía

1. **Vazquez, Oziel.** ¿Qué es una intranet? - ¿No es lo mismo que Internet? - ¿Cual es la diferencia? *SuEspacio.net*. [En línea] 31 de Agosto de 2007. [Citado el: 23 de Noviembre de 2009.] <http://www.suespacio.net/blog/2007/08/31/%C2%BFque-es-una-intranet-%C2%BFno-es-lo-mismo-que-internet-%C2%BFcual-es-la-diferencia/>.
2. Intranet permite la gestión documental en la empresa. *Tecnología Pyme*. [En línea] 2 de Marzo de 2010. [Citado el: 23 de Noviembre de 2009.] <http://www.tecnologiapyme.com/productividad/intranet-permite-la-gestion-documental-en-la-empresa>.
3. **Bonilla, Lucía.** El nuevo papel de la gestión de contenidos ante el incremento del volumen de información. *Sitio Web de la Universidad de Alcalá*. [En línea] 2 de Noviembre de 2009. [Citado el: 1 de Diciembre de 2009.] <http://www.computing.es/EncuentrosComputing/200911020021/-El-nuevo-papel-de-la-gestion-de-contenidos-ante-el-incremento-del-volumen-de-informacion.aspx>.
4. Definición de documento. *Definicion.de*. [En línea] 2008. [Citado el: 9 de Enero de 2010.] <http://definicion.de/documento/>.
5. **Quiroga, Guillermo.** ¿En que consiste la Gestión documental? *README2*. [En línea] 16 de Agosto de 2009. [Citado el: 1 de Diciembre de 2009.] <http://www.readme2.com.ar/index.php/escaneres/escaneres-de-documentos/147-gestdoc>.
6. **Salomón Rodríguez, Jorge Ariel; González Ríos, Osvany.** Implementación de la Interfaz Web para el Gestor de Contenido Empresarial Alfresco. *Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. Universidad de las Ciencias Informáticas. La Habana. Cuba*. [En línea] junio de 2009. [Citado el: 4 de Noviembre de 2009.] <http://biblioteca.uci.cu/sbd/biuci/index.html>.
7. Sitio Web Oficial del ECM Yerbabuena. [En línea] 2008. [Citado el: 10 de Diciembre de 2010.] <http://www.yerbabuena.es/sections/productos/gestion-documental>.
8. Sitio Web Oficial de Docshare. [En línea] 2010. [Citado el: 10 de Diciembre de 2010.] <http://www.docshare.es/>.
9. **Cestero, José M.** Documany, gestión documental básica. *Tecnología PYME*. [En línea] 25 de Junio de 2009. [Citado el: 10 de Diciembre de 2009.] <http://www.tecnologiapyme.com/software/documany-gestion-documental-basica>.

Bibliografía

10. **Elejalde Chacón, Reinier.** Módulo para la creación de modelos de contenido para Alfresco. *Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. Universidad de las Ciencias Informáticas. La Habana. Cuba.* [En línea] junio de 2009. [Citado el: 4 de Noviembre de 2009.] <http://biblioteca.uci.cu/sbd/biuci/index.html>.
11. Sitio Oficial de la Universidad de las Ciencias Informáticas. [En línea] 2010. [Citado el: 10 de Diciembre de 2010.] <http://www.uci.cu/?q=node/55>.
12. Internet: Historia, usos y aplicaciones. *Red Científica peruana.* [En línea] 2002. [Citado el: 2009 de Diciembre de 2.] http://www.yachay.com.pe/servicios/investigador/inv_internet.htm.
13. **Vinent Peña, Alexander Fernando.** Intranet 2.0. Módulo Servicios Postgrado. *Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. Universidad de las Ciencias Informáticas. La Habana. Cuba.* [En línea] Junio de 2009. [Citado el: 14 de Diciembre de 2009.] <http://biblioteca.uci.cu/sbd/biuci/index.html>.
14. **Caraballo Pérez, Yeter.** La gestión de contenidos en portales Web. [En línea] 2 de Febrero de 2007. [Citado el: 15 de Diciembre de 2009.] http://bvs.sld.cu/revistas/aci/vol15_3_07/aci07307.htm.
15. **Tramullas, Jesús;** Herramientas de software libre para la gestión de contenidos. *Hipertext.net.* [En línea] 3 de Mayo de 2005. [Citado el: 15 de Diciembre de 2009.] <http://www.hipertext.net/web/pag258.htm>.
16. **Robertson, J.** What is a C.M.S? *Sitio en Internet de Canadian Portal Software.* [En línea] 2007. [Citado el: 2 de Febrero de 2010.] <http://www.atlanticwebfitters.ca/VendorNeutralWhitePapers/WhatisaCMS/tabid/146/D>.
17. **Cordero Castellanos, Henry; García Portuondo, Israel.** Evento Virtual “Informática 2009. *Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. Universidad de las Ciencias Informáticas. La Habana. Cuba.* [En línea] Abril de 2009. [Citado el: 17 de Diciembre de 2009.] <http://biblioteca.uci.cu/sbd/biuci/index.html>.
18. CMS PHP - Sistema de gestión de contenidos. *Kovacevic. Web Bussiness.* [En línea] 2008. [Citado el: 10 de Enero de 2010.] http://www.kovaonline.com/productos_ficha.php?id=0000003.
19. **López Avellaneda, Daniel.** Introducción a los CMS. *La página web del centro con SPIP.* [En línea] 8 de Marzo de 2008. [Citado el: 10 de Enero de 2010.] <http://www.cepazahar.org/recursos/file.php/7/apuntes/1-Introduccion.pdf>.
20. **Reyero, Jose A.** Sobre Drupal. *Drupal.org.* [En línea] 16 de Marzo de 2006. [Citado el: 12 de Enero de 2010.] <http://www.drupal.org.es/drupal>.
21. Gestión de Contenidos Web Alfresco. *Sitio Web Oficial de Alfresco.* [En línea] 2010. [Citado el: 15 de Febrero de 2010.] <http://www.alfresco.com/es/products/solutions/ecm/wcm/>.

Bibliografía

22. Módulos para la integración de Drupal con Alfresco. *Drupal.org*. [En línea] 2010. [Citado el: 20 de Febrero de 2010.] <http://drupal.org/project/modules?text=alfresco>.
23. **Victoria** . Definición de HTML. *Definición ABC*. [En línea] 12 de Enero de 2009. [Citado el: 12 de Enero de 2010.] <http://www.definicionabc.com/tecnologia/html.php> .
24. XHTML™ 1.0: The Extensible HyperText Markup Language. *W3C*. [En línea] 26 de Enero de 2000. [Citado el: 14 de Enero de 2010.] <http://www.sidar.org/recur/desdi/traduc/es/xhtml/xhtml11.htm>.
25. **Miguel Sanz, José**. XHTML 1.0 - Manual de introducción. *WebExperto*. [En línea] 18 de Junio de 2002. [Citado el: 14 de Enero de 2010.] <http://www.webexperto.com/articulos/art/107/xhtml-10-manual-de-introduccion/>.
26. ¿Qué es CSS? . *Maestros del Web*. [En línea] 2003. [Citado el: 15 de Enero de 2010.] <http://www.maestrosdelweb.com/editorial/introcss/>.
27. **Cárdenas, Lola y Gracia, Joaquin**. Conceptos básicos. *WebEstilo*. [En línea] Enero de 2003. [Citado el: 15 de Enero de 2010.] <http://www.webestilo.com/javascript/js00.phtml>.
28. **Nieto Pérez,Iván;**. Curso de JavaScript. [En línea] 2008. [Citado el: 15 de Enero de 2010.] <http://www.elcodigo.net/tutoriales/javascript/javascript1.html>.
29. **Alvarez, Miguel Angel**. Los Framework Javascript y Ajax más conocidos, utilizados para webs 2.0. *Desarrollo Web*. [En línea] 2008. [Citado el: 16 de Enero de 2010.] <http://www.desarrolloweb.com/articulos/listado-distintos-framework-javascript.html>.
30. Manual de jQuery. [En línea] 2008. [Citado el: 2 de Febrero de 2010.] <http://www.desarrolloweb.com/manuales/manual-jquery.html>.
31. **Carrasco Padrón, Oleiny Omar y León Acosta, Yainel Eugenio**. Análisis y Diseño del Portal de la Revista Cubana de Ciencias Informáticas. *Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. Universidad de las Ciencias Informáticas. La Habana. Cuba*. [En línea] Julio de 2008. [Citado el: 20 de Enero de 2010.]
32. **Salazar Suzeta, Leovys Erix**. Análisis y Diseño del Sistema de Gestión de la Calidad de la Facultad 8.Módulo Pruebas. *Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. Universidad de las Ciencias Informáticas. La Habana. Cuba*. [En línea] Junio de 2009. [Citado el: 20 de Enero de 2010.] http://bibliodoc.uci.cu/TD/TD_2267_09.pdf.
33. Sitio oficial de PostgreSQL. [En línea] 2010. [Citado el: 20 de Enero de 2010.] <http://www.postgresql.org/about/>.

Bibliografía

34. MySQL vs PostgreSQL ¿cuándo emplear cada una de ellas? . [En línea] 2007. [Citado el: 23 de Enero de 2010.] http://www.javahispano.org/contenidos/es/mysql_vs_postgresql_cuando_emplear_cada_una_de_ellas_11/.
35. **Martin Maldonado, Daniel**. Apache, el servidor Web más reconocido. *aplicacionesempresariales.com*. [En línea] 20 de Abril de 2008. [Citado el: 25 de Enero de 2010.] <http://www.aplicacionesempresariales.com/apache-el-servidor-web-mas-reconocido.html>.
36. **Mendoza Sanchez, María A;**. Metodologías De Desarrollo De Software. *www.informatizate.net*. [En línea] 7 de Junio de 2004. [Citado el: 25 de Enero de 2010.] www.informatizate.net/articulos/pdfs/metodologias_de_desarrollo_de_software_07062004.pdf.
37. Wells, D. Extreme Programming: A gentle introduction. *Sitio oficial de XP Extreme Programming*. [En línea] 28 de Septiembre de 2009. [Citado el: 26 de Enero de 2010.] <http://www.extremeprogramming.org/>.
38. **Molpaceres, Alberto**. Procesos de desarrollo RUP, XP, FDD. [En línea] 2003. [Citado el: 1 de Febrero de 2010.] <http://www.willydev.net/descargas/Articulos/General/cualxpfdrup.PDF>.
39. **Jacobson, I, y otros**. El Proceso Unificado de Desarrollo de Software. *Catálogo en línea de la Biblioteca de la Universidad de las Ciencias Informáticas*. [En línea] 2000. [Citado el: 2 de Febrero de 2010.] <http://bibliodoc.uci.cu/pdf/reg00060.pdf>.
40. **Lara Pérez, Marlon y González Rodríguez, Leover Armando**. Sistema de Gestión de la Vigilancia Tecnológica en el Ministerio de la Informática y las Comunicaciones. *Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. Universidad de las Ciencias Informáticas. La Habana. Cuba*. [En línea] Julio de 2009. [Citado el: 15 de Febrero de Febrero.]
41. **Rojas, Elisabeth**. Sun lanza Netbeans IDE 6.8. *muycomputer PRO*. [En línea] 14 de Diciembre de 2009. [Citado el: 3 de Febrero de 2010.] http://muycomputerpro.com/Actualidad/Noticias/Sun-lanza-Netbeans-IDE-6-8/_wE9ERk2XxDDPVrO_6laL2INyQcHZcOuvkVG6BlcO_SGb1LBExTIOvrXUgWOT3UFI4F97ZhV3g74.
42. **Guerrero, Jesús**. Conociendo El Lenguaje De Programación Uml. *El Ram*. [En línea] 28 de Abril de 2009. [Citado el: 3 de Febrero de 2010.] <http://www.fiec.up.ac.pa/elram/index.php?option=58&cont=7>.
43. **Butcher, Matt**. *Learning Drupal 6 Module Development*. BIRMINGHAM - MUMBAI : Packt Publishing, 2008.
44. Drupal programming from an object-oriented perspective. *Sitio oficial de Drupal*. [En línea] 25 de Diciembre de 2009. [Citado el: 11 de Febrero de 2010.] <http://drupal.org/node/547518>.

Anexos

CU-1: Establecer Configuración

Actor: Administrador

Inicio > Administrar > Configuración del sitio

Idam settings **Repository** Advanced

Repository

URL repository: *

The web service end point URL for the Alfresco repository. Example: http://localhost:8080/alfresco/api

Usuario:

To change the current password, enter the new password in both fields. Default password is *guest*

CU-4: Gestionar Consulta- Sección Crear Consulta

Actor: Administrador

Edit Alfresco query

Filters

- Location folder
- Name
- Mime type

<input type="button" value="⊕"/>	Alfresco directory:	<input type="text" value="workspace://SpacesStore/f9aa5246-490c-4087-ab5c-c9a0e:"/> <input type="button" value="Browse..."/>	Match:	
			<input checked="" type="radio"/> Yes	<input type="button" value="Delete"/>
			<input type="radio"/> No	
<input type="button" value="⊕"/>	Name:	<input type="text" value="*a*"/>	Match:	
			<input type="radio"/> Yes	<input type="button" value="Delete"/>
			<input checked="" type="radio"/> No	
<input type="button" value="⊕"/>	Mime:	<input type="text" value="Microsoft Word"/>	Match:	
			<input checked="" type="radio"/> Yes	<input type="button" value="Delete"/>
			<input type="radio"/> No	