

Universidad de las Ciencias Informáticas

Facultad 1



Integración del Archivo Universitario con el módulo Estructura
y Composición del Sistema de Gestión Universitaria

**Trabajo de diploma para optar por el título de Ingeniero en
Ciencias Informáticas**

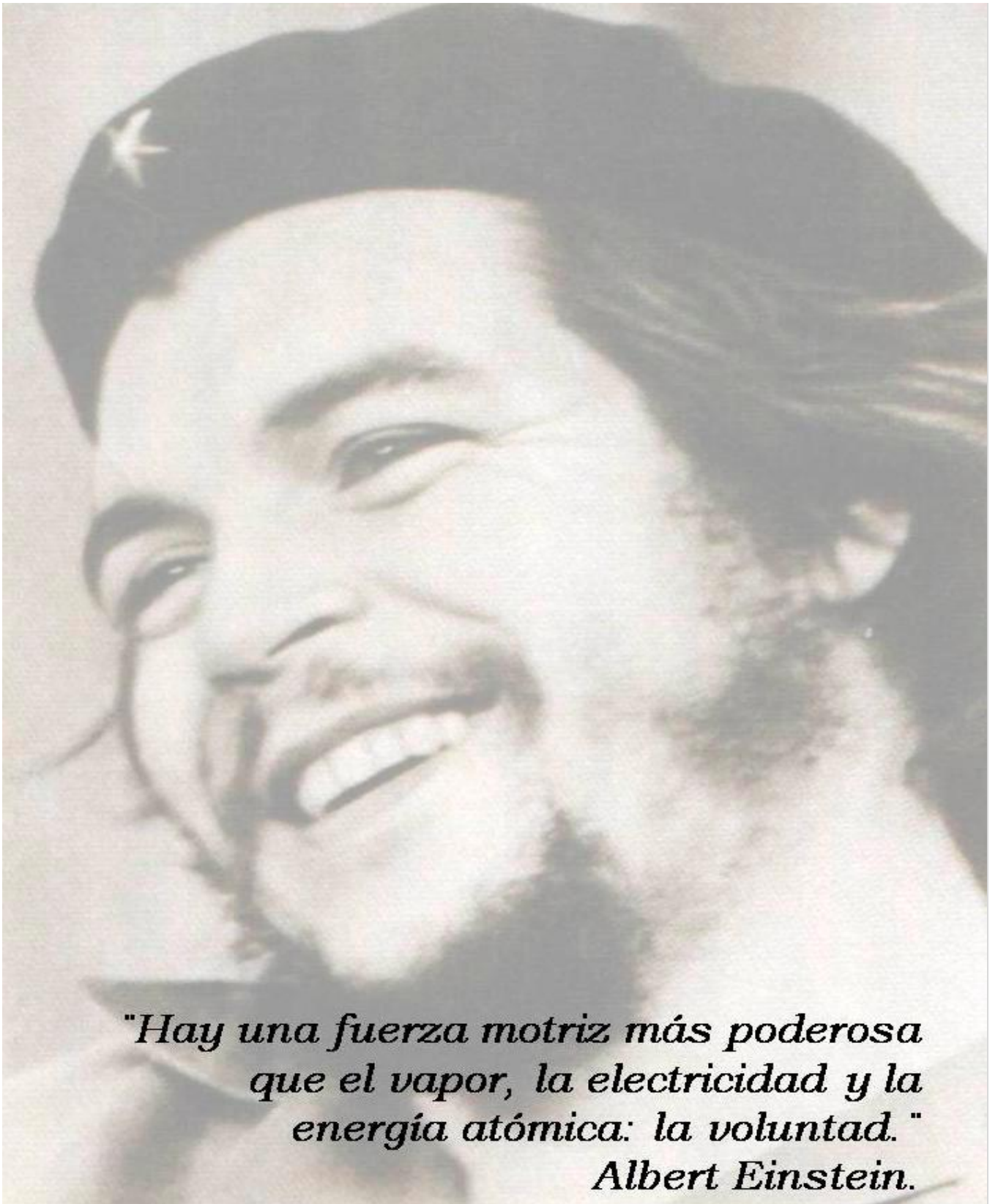
Autor: Liset Rodríguez Landín

Tutores: Ing. Reinier Elejalde Chacon

Ing. Alien Gongora Rodríguez

Ing. Ariosky Areces Gonzalez

La Habana, Junio 2012



*"Hay una fuerza motriz más poderosa
que el vapor, la electricidad y la
energía atómica: la voluntad."
Albert Einstein.*

Declaración de autoría

Declaro que soy el único autor de este trabajo y autorizo al Centro de Informatización Universitaria (CENIA) de la Universidad de las Ciencias Informáticas; así como a dicha institución para que hagan el uso que estimen pertinente con este trabajo.

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

Liset Rodríguez Landín

Firma del autor

Ing. Alien Gongora Rodríguez

Firma del tutor

Ing. Ariosky Areces Gonzalez

Firma del tutor

Ing. Reinier Elejalde Chacon

Firma del tutor

Agradecimientos

Un especial agradecimiento a mis padres y a mi hermano Yuri, por ser un ejemplo de sacrificio, confianza, comunicación, dedicación y sobre todo mucho amor.

Gracias mami por ser la madre más dedicada, sacrificada y buena de este mundo. Gracias por apoyarme siempre, por darme fuerzas cada minuto que lo he necesitado y decirme que yo sí podía. Gracias por estar a mi lado en momentos tan importante como este. Te Quiero Muchooo.

A mi papi muchas gracias por ser el mejor papá del mundo. Por estar al igual que mi mamá siempre a mi lado, guiándome por el mejor camino. Por ser mi ejemplo a seguir, mi confidente, el que siempre tapaba las cosas que hacía para que mami no me regañara. Mami siempre dice que soy igualita a ti, y para mí es un orgullo que sea así. Te Quiero Mucho

Que decir de mi hermano. Mi segundo papá. Mi todo. Gracias por sacrificarte por mí. Por estar siempre a mi lado. Por cuidarme y tenerme presente. Gracias por ayudarme a cumplir este sueño. Te Quiero Mucho.

Agradezco a mi hermano más pequeño, Bernardito por su cariño y por quererme tanto. Yo también Te Quiero.

A mis dos sobrinos pequeñitos, que aunque aún no saben mucho quien soy, me hacen reír con sus ocurrencias cada vez que los veo. Los Quiero.

A una persona muy especial en mi vida y que ya no está entre nosotros, "Mi Abuelita". Gracias, porque estuvo siempre ahí para dar su cariño, sus buenos consejos. Por ser la abuelita más linda y arregladita del mundo. "Gracias, Te quiero Mucho".

Al novio de mi mamá, que aunque me puse un poquito bravita cuando entró en nuestras vidas, le agradezco lo dedicado que es conmigo, tratando siempre de consentirme en todo. Pero sobre todo le agradezco porque sé que quiere mucho a mi mamá y la cuida.

A la pareja más bonita que he conocido y que me da mucho gusto poder contar con su amistad: Yodelkys y Rodisbel y sus dos bebés Melani y Humberto. Gracias por compartir conmigo tantos momentos buenos y malos. Sé que siempre puedo contar con ustedes.

Muchas gracias a mis tutores. A Alien, que aunque mucho me mortificó sabe que lo quiero. Gracias por decirme que si podía lograrlo. Ariosky, te agradezco porque siempre que necesité tu ayuda estuviste ahí, explicándome las cosas una y mil veces. Dios mío, qué decir del Reinier, la persona que me dijo una vez salida de la Pre – defensa y casi sin conocerme que si podía lograrlo. Que me dio fuerzas cuando más lo

Agradecimientos

necesité. Reinier, no me alcanzan las palabras para agradecerte toda la dedicación que me diste.

Siempre te llevaré en mi corazón. Muchas Gracias!!!

A Dionicio que más que oponente fue un tutor. Siempre estuvo disponible para cualquier duda. Me dio aliento y me hizo saber que todo sacrificio merece una recompensa. Y creo que esta es mi recompensa.

A Norges que lo tenía loco preguntándole cosas de Estructura y Composición. Y nunca hubo un NO como respuesta.

A las profesoras que me ayudaron a entender la ingeniería de mi propuesta: Ailec, Aneity y Damián.

A los profesores de mi laboratorio que siempre me aclararon muchas dudas y estuvieron al pendiente mío: Lizandra, Michel, Marcel, Pedro y Misael.

A Kerlyns, gracias, porque en mi vida fue muy importante. Le agradezco toda la paciencia que tuvo conmigo. Su apoyo en todo momento, su buen humor ante cualquier problema. Gracias a su familia porque me malcriaron como una hija más y por preocuparse por mí hasta el día de hoy.

A Emilio como se hizo llamar una vez. ¡ Quiero Mucho. Sabes que eres muy especial en mi vida. Te agradezco todo los momentos tan lindos que me regalaste. Nunca te olvidaré y lo sabes.

Ahora llegó el momento de agradecer a todas aquellas personas que conocí aquí en la universidad y que hicieron que este sueño se hiciera realidad.

A mis amistades desde primer año.

Adrián Rivera, gracias por todos esos repasos que me diste siempre que lo necesité, aunque tenía que sacar cita con tiempo de antelación porque siempre fuiste un chico muy ocupado, estuviste siempre disponible. Gracias por todo.

Gracias a Ivansillo como cariñosamente le digo, por aguantarnos a mí y Claudia cuando estudiábamos para la prueba de nivel de Programación. Por ayudarme a traducir aquellos textos de inglés tan largos con sus palabras para que yo pudiera entender mejor. Gracias por brindarme todos estos años de amistad. Te Quiero

A Claudia que por cosas de la vida hoy no se encuentra aquí le agradezco mucho por haber sido mi compañera de estudio hasta las tantas de la madrugada. Por hacerme reír con sus ocurrencias. Por ayudarme y darme apoyo cuando lo necesité con inglés.

Agradecimientos

A Isa, gracias por brindarme tu apoyo y estar siempre ahí para cualquier duda. Porque fuiste mi compañera en el lab hasta casi el amanecer por mucho tiempo. Por coger tu cell como medio de comunicación. Por apoyarme en todas mis locuras. Sabes que te quiero mucho.

A Jtin como cariñosamente le digo, y él un poco bravito me decía: "Jta, no Jtin". Te Quiero a pesar de que estabas loco de que pasara a ser plantilla de tu club el curso que viene, jajaja. Nunca faltó una sonrisa para mí, un chiste. Eres genial.

A las demás integrantes de "Los ángeles de Charlie", Diane y Zoe. Las quiero mucho. Ambas saben que son especiales para mí. Siempre estuvieron ahí cuando me hacía falta un buen consejo, y cuando necesitaba uno no tan bueno. Recuerdan cuando pegamos en la pared del cuarto de las 3, las fotos de "RBD" y como niñas chiquitas decíamos: esta soy yo. Las quiero mucho y le agradezco por todo lo que juntas vivimos.

Gracias a mis compañeros: Yai, Yanet, Anisley, Dayamí, Yary, Karelia, Luisón, Yasnelis, Raisa, Lisbethy, Maryanis, Yisel, Isied y Dreydis. Juntos pasamos lindos momentos.

A todos los que hicieron que estos cinco años, se convirtieran en los mejores años de mi vida. Gracias.

Liset Rodríguez Landín

Dedicatoria

A mis padres Bárbara y Bernardo porque gracias a ellos fue posible hacer realidad este gran sueño.

A mi hermano Yuri por estar siempre presente en cada etapa de mi vida, guiándome de la mejor forma posible.

A toda mi familia y en especial a mi abuelita Norma.

Liset Rodríguez Landín

Resumen

Actualmente, en la Universidad de las Ciencias Informáticas (UCI) se desarrolla el Archivo Universitario, aplicación que permite almacenar las evidencias documentales como resultado de los procesos universitarios. Sin embargo, carece de un medio que permita actualizar de forma automática los usuarios, grupos de usuarios y permisos. Con el objetivo de dar solución a este problema, la presente investigación propone la integración del Archivo Universitario con el módulo Estructura y Composición del Sistema de Gestión Universitaria. Para ello se describen los antecedentes, aspectos teóricos y conceptuales relacionados con la gestión de usuarios, grupos de usuarios y permisos en el Gestor de Contenido Empresarial Alfresco.

Se realiza el análisis de una solución existente en la universidad. Se seleccionan las herramientas, metodología y tecnologías necesarias para el desarrollo de la propuesta. Se define el diseño de la solución y se describen los artefactos. Además, se valida la propuesta de solución a través de las técnicas de construcción de requisitos y las métricas para evaluar los requisitos y el diseño.

Palabras clave: archivo universitario, estructura y composición, integración

Índice de contenido

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA	5
1.1 PRELIMINARES	5
1.2 EL ARCHIVO UNIVERSITARIO DE LA UCI. MOTIVACIONES FUNDAMENTALES	9
1.3 EL CONTEXTO TECNOLÓGICO DEL ARCHIVO UNIVERSITARIO	12
1.3.1 <i>El ECM Alfresco como base tecnológica del Archivo Universitario</i>	12
1.3.2 <i>El modelo de seguridad</i>	13
1.3.3 <i>Servicios de seguridad</i>	17
1.3.4 <i>Funcionamiento del modelo de seguridad</i>	18
1.4 MÓDULO ESTRUCTURA Y COMPOSICIÓN DEL SISTEMA DE GESTIÓN UNIVERSITARIA	19
1.5 MÓDULO ESTRUCTURA Y COMPOSICIÓN DE UCI-FAR	21
1.6 TECNOLOGÍAS DE SERVICIOS WEB.....	21
1.7 METODOLOGÍA DE DESARROLLO DE SOFTWARE	24
1.8 LENGUAJE DE MODELADO	28
1.9 HERRAMIENTA CASE	29
1.10 SOLUCIÓN PARA EL DISEÑO DE PROTOTIPOS DE INTERFACES DE USUARIO	30
CAPÍTULO 2. PROPUESTA DEL SISTEMA	32
2.1 SITUACIÓN PROBLEMÁTICA	32
2.2 OBJETO DE AUTOMATIZACIÓN	32
2.3 PROPUESTA DE SOLUCIÓN	33
2.4 EL CONTEXTO DEL SISTEMA. MODELO DE DOMINIO	36
2.4.1 <i>Diagrama de clases del modelo de dominio</i>	36
2.4.2 <i>Especificación de los requisitos de software</i>	38
2.5 TÉCNICAS PARA LA CAPTURA DE REQUISITOS	38
2.6 REQUERIMIENTOS FUNCIONALES.....	39
2.7 REQUERIMIENTOS NO FUNCIONALES.....	41
2.8 ESPECIFICACIÓN DE REQUISITOS COMO CASOS DE USO	42
2.8.1 <i>Definición de los casos de uso del sistema</i>	42
2.8.2 <i>Definición de los actores</i>	44
2.8.3 <i>Descripción de casos de uso del sistema</i>	44
2.8.4 <i>Prototipo de interfaz de usuario</i>	47
CAPÍTULO 3. ANÁLISIS Y DISEÑO	49

Índice de contenido

3.1 MODELO DE ANÁLISIS	49
3.1.1 Diagramas de clases del análisis	50
3.1.2 Diagramas de interacción	50
3.2 ARQUITECTURA DEL SISTEMA	51
3.2.1 Repositorio de contenidos (Repository)	52
3.2.2 Núcleo (Core)	53
3.3 MODELO DE DISEÑO	54
3.3.1 Patrones de diseño	54
3.3.2 Diagramas de clases del diseño	56
3.4 VALIDACIÓN DE LOS REQUERIMIENTOS	58
3.4.1 Métrica para evaluar los requisitos	58
3.4.2 Técnica de construcción de prototipos	62
3.5 Validación de diseño mediante criterio de especialistas del proyecto	62
3.5.1 Validación del modelo de diseño propuesto	62
3.5.2 Resultados del instrumento de evaluación de la métrica Tamaño Operacional de Clase (TOC)	63
CONCLUSIONES GENERALES	67
RECOMENDACIONES	68
REFERENCIAS BIBLIOGRÁFICAS	69
BIBLIOGRAFÍA	73

Introducción

El desarrollo acelerado del mundo actual ha puesto dinamismo al almacenamiento de la información, haciendo que el progreso organizativo de una institución dependa considerablemente de la manera en que se gestionen y almacenen los documentos, con vistas a satisfacer las necesidades de las personas y las organizaciones.

Los sistemas informáticos que permiten realizar la gestión de evidencias documentales se han convertido en herramientas básicas para soportar el registro, control, circulación y organización de los documentos que en formato electrónico se envían y reciben en una organización.

El sector educacional es uno de los favorecidos con la evolución de la gestión documental, siendo la Educación Superior uno de los más beneficiados, ya que generan y reciben un número considerable de documentos. Estos documentos representan la consecuencia directa de las acciones necesarias para el cumplimiento y ejercicio de sus finalidades: el estudio, la docencia y la investigación, según la legislación vigente (Ley orgánica, 1983).

Las universidades cubanas no están exentas de esta realidad y se han sumado, en algunos casos, a la creación de archivos universitarios que permitan brindar un servicio especializado en la gestión y conservación de los documentos administrativos; a partir de la aplicación de la gestión documental, con vistas a alcanzar una mejor organización, flujo, recuperación y conservación documental eficiente, en una era donde se impone el rápido acceso a la información.

En Cuba, un centro de enseñanza superior surgido en el fragor de la Batalla de Ideas y obra del pensamiento visionario del compañero Fidel es la Universidad de las Ciencias Informáticas (UCI). La UCI constituye la más abarcadora de su tipo en el país, tanto en expansión como en personal interno, se encuentra estructurada por disímiles áreas y cada una de ellas compuesta por un conjunto de personas que asumen distintas responsabilidades en la misma. La universidad no se encuentra exenta de gestionar sus evidencias documentales para así mejorar su proceso organizativo y hacer mejor uso de la información, estas evidencias generadas por las diferentes áreas y personas que en ella estudian o trabajan, constituye un elemento vital en la mejora de los procesos internos.

La estructura y composición de la universidad es muy cambiante, por lo que las evidencias documentales a almacenar, consultar o modificar se pueden encontrar expuestas a la suplantación y el fraude debido a la insuficiencia del mecanismo de actualización de usuarios y sus responsabilidades. Idealmente, las personas deben tener el permiso necesario o suficiente para crear o actualizar una evidencia documental según el rol que desempeña en un área específica de la estructura organizativa. Actualmente, en la UCI se realiza un proceso fundamental referente a las actividades de formación. En

este proceso, denominado Gestión Universitaria, participan diferentes áreas de la universidad como son: Pregrado, Posgrado, Producción, Investigación, Ingreso, Ubicación Laboral, Laboratorio, Residencia, Extensión Universitaria, Cooperación Internacional, Biblioteca y Teleformación. El mismo cuenta en su núcleo con procesos de apoyo que centralizan la información que es común para todos los subsistemas. Dentro de los módulos con los que cuenta el núcleo se encuentra Estructura y Composición, el cual gestiona la información referente a toda la estructura administrativa y la jerarquía de la misma, así como la asignación de responsabilidades a las estructuras.

Actualmente, en el archivo de la universidad la gestión de usuarios, grupos de usuarios y permisos se realiza de forma manual lo que significa que el administrador del Archivo UCI tiene que actualizar individualmente cada uno de los usuarios, grupos y permisos que hayan sufrido cambios, sin aprovechar la información que brinda el módulo Estructura y Composición del Sistema de Gestión Universitaria. Esto puede traer como consecuencia pérdida de información, o una considerable tardanza en la actualización de la misma, además de dificultarse la obtención de la documentación de forma inmediata, por lo que se afecta su disponibilidad, lo que puede provocar errores en posible toma de decisiones.

Partiendo de la situación expuesta anteriormente surge el siguiente **problema de la investigación**: ¿Cómo actualizar los datos referentes a los usuarios, grupos de usuarios y permisos en el Archivo Universitario por medio del módulo Estructura y Composición del Sistema de Gestión Universitaria?

Se define para la presente investigación como **objeto de estudio**: la gestión de usuarios y permisos en los sistemas de contenidos empresariales.

El **campo de acción**: estará enmarcado en la gestión de usuarios y permisos en el Gestor de Contenido Empresarial Alfresco. Para dar solución al problema existente se ha tomado como **objetivo general**: diseñar una propuesta de solución que permita integrar al Archivo Universitario los datos referentes a la estructura y composición de la universidad, contenidos en el sistema de Gestión Universitaria, para así lograr la gestión automática de usuarios, grupos de usuarios y permisos en el mismo.

De donde se derivan los siguientes **objetivos específicos**:

- ✓ identificar los elementos del marco teórico conceptual de la investigación
- ✓ realizar el análisis de procesos de actualización de la gestión de usuarios, grupo de usuarios y permisos
- ✓ definir el diseño de procesos de actualización de la gestión de usuarios, grupo de usuarios y permisos
- ✓ validar los requerimientos de la propuesta de solución

La investigación se sustenta en la siguiente **idea a defender**: la integración del Archivo Universitario con el módulo Estructura y Composición, le permitirá al mismo gestionar de forma automática los usuarios, grupos de usuarios y permisos, manteniéndolos actualizados y facilitando que las evidencias documentales se encuentren menos propensas a la suplantación y el fraude.

Los métodos científicos teóricos que se emplearon para darle solución a las tareas propuestas fueron:

- ✓ Histórico-lógico - permitió realizar el estudio del estado del arte e investigar acerca de aplicaciones o soluciones similares, tecnologías, herramientas y sobre el proceso de desarrollo de *software* utilizado
- ✓ Analítico-sintético - facilitó el análisis de la bibliografía consultada, con el objetivo de realizar una síntesis de los elementos relevantes que más aporten a la propuesta de solución.

Los métodos científicos empíricos que se emplearon para darle solución a las tareas propuestas fueron:

- ✓ Observación - se utilizó para identificar la situación problemática que da paso al desarrollo de la investigación, así como para realizar una evaluación de los resultados obtenidos con los resultados esperados.

Justificación de la investigación

La presente investigación se realiza para mantener actualizados de forma automática en el Archivo Universitario los usuarios, grupos de usuarios y permisos. Actualmente, el Archivo Universitario realiza la gestión de usuarios, grupos de usuarios y permisos de forma manual, sin aprovechar la información que brinda el módulo Estructura y Composición del Sistema de Gestión Universitaria. Esto puede traer como consecuencia pérdida de información, o una considerable tardanza de la misma, las evidencias documentales se encuentran expuestas a la suplantación y el fraude, además de dificultarse la obtención de las mismas de forma inmediata.

El presente trabajo, está estructurado en tres capítulos, distribuidos de la siguiente manera:

Capítulo 1. Fundamentación teórica

Se abordan los diferentes elementos que brindan un marco teórico-conceptual para el desarrollo de la solución propuesta. Se presenta el resultado del estudio realizado al módulo Estructura y Composición de UCI – FAR. Se seleccionan las herramientas, metodología y tecnologías necesarias para el desarrollo de la propuesta.

Capítulo 2. Propuesta del sistema

En este capítulo, a partir de un análisis crítico de la situación problemática, se elabora una propuesta de solución desde el punto de vista conceptual. Se definen y describen los requerimientos funcionales y no funcionales. Finalmente se especifican los casos de uso asociados a los requerimientos identificados.

Capítulo 3. Análisis y diseño

En este capítulo se realizan los diagramas de clases de análisis, los diagramas de colaboración y el diagrama de clases del diseño, así como una descripción de la arquitectura y de los patrones de diseño utilizados. Se realiza una validación de los requisitos y diseño de la propuesta aplicando algunas métricas en pos de verificar la calidad y efectividad de los mismos, comprobando que estos cumplen con las necesidades del cliente.

Capítulo 1. Fundamentación teórica

Capítulo 1. Fundamentación teórica

En el presente capítulo se abordan los diferentes elementos que brindan un marco teórico-conceptual para el desarrollo de la solución propuesta. Para ello se definen un conjunto de términos relacionados con el objeto de estudio de la presente investigación. Luego se presenta el resultado del estudio realizado al módulo Estructura y Composición de UCI – FAR y el de Gestión Universitaria. Además, se seleccionan las herramientas, metodología y tecnologías necesarias para el desarrollo de la propuesta.

1.1 Preliminares

En lo adelante, en esta misma sección, se presentan un conjunto de definiciones que fueron imprescindibles para la comprensión de los temas que durante la investigación se llevaron a cabo y que de una forma u otra se presentan como referente en el marco este documento. Partiendo de algunas definiciones esenciales desde el punto de vista teórico de la investigación, se definen otros términos más relacionados con el marco tecnológico que dan soporte en primer lugar a cada uno de los conceptos teóricos previamente citados y en segundo lugar al problema en general.

Documento

La palabra *documento* proviene de la voz griega *docere* (instruir), de la cual se deriva la palabra *documentum* que significa “con lo que alguien se instruye” y que más adelante hasta la actualidad ha pasado a ser “con lo que alguien se informa”. La definición formal de documento sin embargo, ha sido interpretada y reformulada de diferentes formas.

El Comité de Archivos Electrónicos del Consejo Internacional de Archivos por ejemplo, define documento como “*información registrada, producida o recibida en torno a la implantación, realización y ámbito de una actividad institucional o personal que engloba contenido, contexto y estructura y permite probar la existencia de la actividad que lo generó*”(STABLE, 2009).

Por otra parte, la *Public Record Office*, lo define como el “*conjunto consistente de datos registrados en un soporte, o bien, fragmento de información registrada, generada, reunida o recibida desde el comienzo, durante el seguimiento y hasta la finalización de una actividad institucional o personal, y que comprende un contenido, un contexto y una estructura suficiente para constituir una prueba de esta actividad*” (Public Record Office, 1999).

Capítulo 1. Fundamentación teórica

Documento universitario

Derivado del término documento y de la interpretación que cada cual asume al respecto, el concepto de documento universitario ha ido derivando en lo que la mayoría de los especialistas en la rama consideran como (RODRIGUEZ, 2005):

“toda expresión textual, en lenguaje escrito, natural o codificado, así como toda imagen gráfica o impresión sonora, recogida en distintos soportes que constituya un testimonio de las actividades y de las funciones de la universidad. Se exceptúan de esta definición aquellos documentos producidos por miembros de la comunidad universitaria en el ejercicio de sus actividades privadas o profesionales”.

En el contexto universitario, de acuerdo con las definiciones de documento y documento universitario antes citados, un documento de archivo es el testimonio material de un hecho o acto realizado en el ejercicio de las funciones y actividades de la universidad, por las diferentes personas y órganos que componen la comunidad universitaria, sea cual sea la fase en que estos se encuentren, la ubicación de los mismos y los procesos sustanciales o de apoyo en que se encuentren involucrados.

Expediente

Unidad documental formada por un conjunto de documentos, generado orgánica y funcionalmente por un sujeto productor en la resolución de un mismo asunto (RODRIGUEZ, 2005).

En general, pudiera decirse que un expediente electrónico es el conjunto de los documentos producidos y recibidos como resultado de diferentes actuaciones administrativas orientadas a resolver un determinado asunto o trámite de la administración. Las características básicas de un expediente son (ABDALA, 2000):

- ✓ responde a un trámite que debe ser resuelto, de acuerdo con procedimientos previamente establecidos
- ✓ en el expediente obran diferentes instancias o dependencias que aportan documentos para resolver dicho trámite
- ✓ los documentos se presentan en el mismo orden en el cual se dieron las actuaciones y trámites
- ✓ tiene un principio y fin claramente determinados
- ✓ existe un vínculo indeleble de unión entre cada uno de los documentos que lo conforman
- ✓ los documentos de un expediente pueden ser de diferentes tipos y formatos

Archivo

Conjunto orgánico de documentos producidos y / o recibidos en el ejercicio de sus funciones por las personas físicas o jurídicas, públicas o privadas. En función del organismo productor, los archivos

Capítulo 1. Fundamentación teórica

pueden ser de la Administración Central, periférica, autonómica, local y judicial. En función del ámbito de sus fondos: nacionales, generales, regionales, provinciales, distritales y municipales (RODRIGUEZ, 2005).

También se conoce como archivo al local donde se conservan y consultan los conjuntos orgánicos de documentos sea cual fuere su fecha, su forma y soporte material, acumulados en un proceso natural por una persona o institución pública o privada, en el transcurso de la gestión (RODRIGUEZ, 2005).

Archivo Universitario

El archivo universitario es una unidad funcional que tiene como misión la recepción, tratamiento, organización, accesibilidad y difusión del patrimonio documental de la universidad que resulte ser consecuencia directa de sus actividades administrativas, docentes e investigativas, abarcando todas las etapas del ciclo de la evolución documental. La recepción y conservación de los documentos en el Archivo Universitario tiene como fin: contribuir a la eficacia y eficiencia de la gestión administrativa, satisfacer las necesidades de información en la toma de decisiones y cooperar en el avance del conocimiento; al mismo tiempo, garantiza la transparencia de la gestión universitaria y el derecho de acceso a los documentos (Universidad de Sevilla, 2003).

Se considera entonces un archivo universitario como el conjunto de documentos de cualquier fecha, formato o soporte material, producidos o reunidos en el desarrollo de las funciones y actividades de los diferentes miembros y órganos universitarios, organizados y conservados para la información y la gestión administrativa así como para la investigación. Entiéndase igualmente por archivo universitario al servicio especializado en la gestión, conservación y difusión de los documentos con finalidades administrativas, docentes, investigativas y culturales de la universidad.

Gestión documental y archivística

En 1950 la *Federal Act* institucionalizó el término *records management* como forma idónea para el tratamiento documental en los Estados Unidos, definiéndola como el “*área de la gestión administrativa general que se ocupa de garantizar la economía y eficiencia en la creación, mantenimiento, uso y disposición de los documentos a lo largo de todo su ciclo de vida, con el objetivo de asegurar una documentación adecuada, evitar lo no esencial, simplificar los sistemas de creación y uso del papeleo, mejorando la forma de cómo se organizan y recuperan los documentos*” (MSc. MUGICA, Mayra Mena, 2007).

Luego, la norma UNE-ISO 15489-1 define este término como el “*área de la gestión responsable de un control eficaz y sistemático de la creación, la recepción, el mantenimiento, el uso y la disposición de documentos de archivo, incluidos los procesos para incorporar y mantener en forma de documentos la*

Capítulo 1. Fundamentación teórica

información y prueba de las actividades y operaciones de la organización” (Comité Técnico de Normalización, 1998).

Dicho de otra manera, se trata del mecanismo que dicta la forma en que se gestionan los documentos desde que nacen – *se crean o incorporan por algún mecanismo o sistema a la organización* – hasta que son finalmente conservados o eliminados, según su valor patrimonial, en un archivo.

Documento electrónico

El equipo de investigación de InterPARES definió un documento archivístico electrónico como un documento archivístico creado (por ejemplo, realizado o recibido y guardado para acción o referencia) de forma electrónica, significando que un mensaje recibido en forma electrónica pero guardado para acción en forma de papel es un documento archivístico en papel, mientras que una carta recibida en papel pero escaneada en el ordenador y utilizada sólo como fichero digital es un documento archivístico electrónico (DURANTI, 2004).

Sistema electrónico de archivo (RODRIGUEZ, 2005)

Es un conjunto coherente de componentes tecnológicos, *software* y *hardware*, necesarios para almacenar y recuperar documentos y organizarlos para controlar las funciones específicas de producción y almacenamiento de los documentos y el acceso a los mismos, con el fin de salvaguardar su autenticidad y fiabilidad. Garantizan el mantenimiento y la conservación de la autenticidad, fiabilidad y accesibilidad de los documentos a lo largo del tiempo, según las regulaciones y normativas establecidas en la organización en la que se despliegan.

Las funciones principales de un archivo electrónico son:

- ✓ la organización y gestión de documentos
- ✓ la descripción documental
- ✓ la gestión de usuarios
- ✓ las auditorías de uso y circulación de documentos
- ✓ la gestión de los accesos, el control de la autenticidad e integridad de los documentos
- ✓ los criterios de migración de los datos y renovación de los soportes para garantizar el acceso y disponibilidad de los documentos a lo largo del tiempo y
- ✓ cualquier otra tarea técnica relacionada con la creación, uso y disponibilidad de los documentos electrónicos

Capítulo 1. Fundamentación teórica

Gestión de Contenido Empresarial (ECM)

El término ECM surge en el año 2000 por la AIIM (*Association for Information and Image Management* según sus siglas en inglés) Internacional, que constituye la asociación para la gestión o administración del contenido empresarial. Su significado ha sido redefinido en determinados momentos (AIIM, 2009).

A finales de 2003 se define ECM por primera vez como el conjunto de tecnologías usadas para capturar, gestionar, almacenar, preservar y entregar el contenido y los documentos relacionados a los procesos organizacionales. A principio de 2006, AIIM amplía la definición señalando que las tecnologías y estrategias de los ECM permiten gestionar la información no estructurada de una organización, donde sea que dicha información exista. Por última vez, a principio de 2008 se define Gestión de Contenido Empresarial como el conjunto de estrategias, métodos y herramientas usadas para capturar, gestionar, almacenar, preservar y entregar el contenido y los documentos relacionados a los procesos que se llevan a cabo en las organizaciones (AIIM, 2009).

Esta última definición pretende abarcar completamente el problema que ha sido tradicionalmente dirigido a la gestión documental, incluyendo los problemas adicionales involucrados con la conversión de la información a un contexto digital.

1.2 El Archivo Universitario de la UCI. Motivaciones fundamentales

A lo largo de la historia, la universidad ha sido una de las instituciones que más ha influido en la creación de conocimientos y en el desarrollo del talento, de las ideas y de la capacidad crítica de las personas, contribuyendo de esta forma al bienestar de la sociedad. Para ello, ha asumido importantes transformaciones, adaptándose a las necesidades de la sociedad. Así, ha demostrado una extraordinaria capacidad de adaptación al cambio, como lo prueba el que siga viva y continúe siendo un instrumento clave para el desarrollo de las sociedades.

En la actualidad, el sistema universitario se enfrenta en todo el mundo a una de esas situaciones que exigen transformaciones profundas. El desarrollo acelerado de las Tecnologías de la Información y de las Comunicaciones está configurando nuevas formas de relación entre los seres humanos y también con las organizaciones creadas por ellos. La sociedad industrial y de servicios ha evolucionado hasta la Sociedad de la Información y del Conocimiento.

La universidad no es ajena a ello. Por un lado, es agente activo del cambio como generadora y transmisora de conocimientos; por otro, necesita innovar en sus propios procesos de gestión, de docencia y de investigación para hacer frente a las exigencias de esa misma evolución. Se trata de un

Capítulo 1. *Fundamentación teórica*

círculo que será virtuoso para las universidades que sean capaces de interpretar correctamente las nuevas formas de aportación de valor que la sociedad les exige, pero que llevará al fracaso a aquellas que no sean capaces de encontrar su nueva posición (BIBLIOTECA, 2003).

La Universidad de las Ciencias Informáticas, “consciente” de esta situación y de la necesidad de empujar su sistema universitario hacia nuevas formas de trabajo y de prestación de servicios, ha promovido la creación de un sistema de Archivo: el Archivo Universitario o Archivo UCI, asumiendo la urgencia de impulsar el desarrollo de un sistema de respaldo de la información patrimonial de la institución, generando un marco legislativo, innovador y ambicioso que influye de manera muy directa en el servicio universitario en general.

La meta fundamental del Archivo Universitario sería poner en manos de la comunidad universitaria una aplicación informática para la gestión de los documentos resultantes de la ejecución de los procesos universitarios y sobre todo, a disposición del personal de la administración, los medios necesarios para llevar a cabo su correcta gestión. Al mismo tiempo, contribuiría a:

- ✓ disponer de la información institucional en soporte electrónico
- ✓ facilitar la recogida, organización, almacenamiento y difusión de la información
- ✓ garantizar la integración de la información como dato único
- ✓ hacer de los medios telemáticos la principal vía de comunicación y de acceso a la información en la universidad y
- ✓ garantizar la seguridad de la información y el cumplimiento de las directivas legales relacionadas con el uso de datos personales y comunicación de la información

Teniendo en cuenta la meta a alcanzar y sus derivados beneficios, se tuvieron en cuenta un conjunto de aspectos en la concepción del modelo de gestión de los documentos de la universidad a través del Archivo UCI.

Clasificación de los documentos

Se identificó que en cualquiera de los sistemas de información es imprescindible un sistema o herramienta que permitiera la clasificación de las diferentes funciones y actividades que generan los documentos, para asumir tanto la organización física de los documentos como su posterior recuperación y la decisión para su disposición final.

Sabiendo que en un sistema de este tipo intervienen todas las personas que generan o utilizan documentos en sus funciones, se hace imprescindible una identificación y clasificación integral común en toda la organización. En este sentido la clasificación documental permitiría formar los expedientes

Capítulo 1. *Fundamentación teórica*

cuyos documentos se gestionan por diferentes unidades administrativas. Además, permitiría a los usuarios comprender qué lugar ocupan en la estructura organizativa de la universidad.

Seguimiento de los documentos en los procedimientos

El seguimiento de los documentos durante su ciclo de vida y la formación de expedientes permitirían completar la gestión global de los documentos. Para llegar a este nivel se identificó la necesidad de vincular al servicio de archivo las funciones o herramientas necesarias para el registro de entrada y salida de documentos, el análisis, seguimiento y control de los procedimientos.

Conservación y eliminación de los documentos

Se hace imprescindible poder disponer de un buen sistema que asegure tanto la eliminación controlada de aquellos documentos en los que no pervive ninguno de los valores administrativos, legales, fiscales, científicos y tecnológicos o históricos así como la conservación de los documentos que mantengan alguno de dichos valores. Por otra parte, dicho sistema debe asegurar la conservación y acceso de los documentos generados en soportes electrónicos.

Descripción y difusión de los documentos

Es también importante la descripción mecanizada de los documentos. En primer lugar para la recuperación y uso en las fases activa y semiactiva por parte de los usuarios de la comunidad universitaria, y en segundo lugar para posibilitar el acceso o consulta a distancia mediante sistemas de descripción estandarizados y en red.

Control de acceso a los documentos

Con la mecanización de la gestión se hace necesario sensibilizar el control de acceso a los documentos y por otra parte garantizar una protección especial para los documentos esenciales de la universidad. La identificación de este tipo de documentos, así como una gestión eficaz de las medidas de protección especiales necesarias, se hace imprescindible para asegurar por una parte su control y por otra su conservación y recuperación.

Recuperación de los documentos producidos por las actividades de la comunidad universitaria

En la concepción de un sistema de tal envergadura, se puede considerar o clasificar como incompleto si no se plantea la recuperación de la documentación generada por las actividades que los miembros de la comunidad universitaria desarrollan, incluyendo aquellos que a menudo no pueden vincularse a ninguna de las funciones que la universidad tiene encomendadas.

Capítulo 1. Fundamentación teórica

1.3 El contexto tecnológico del Archivo Universitario

Bien definida la meta del Archivo UCI, fue esencial tener en cuenta cómo se implementaría todo el sistema de gestión de los documentos universitarios. Para ello, se ubicó en una posición central en la estructura orgánica y funcional de la universidad, permitiendo el diseño de un sistema archivístico universitario único con una misión transversal y horizontal con incidencia directa en todos los documentos de la universidad.

Hoy, sin embargo, este sistema solo se compone de los *portafolios digitales*¹ y expedientes de cada una de las personas que pertenecen al centro. Esto significa que aún hay mucho que hacer para lograr las metas anteriormente citadas. Uno de los elementos fundamentales a tener en cuenta es la necesidad de generar confianza y ofrecer la disponibilidad, fiabilidad y seguridad de los datos que figuren en los documentos que en este sistema se almacenan. En este sentido, es necesario analizar dos elementos esenciales, en primer lugar, la estructura de la universidad y en segundo lugar, el rol que desempeña cada persona en las diferentes estructuras.

1.3.1 El ECM Alfresco como base tecnológica del Archivo Universitario

La gestión de contenido empresarial no solo se refiere a las soluciones que utilizan las tecnologías de *Internet*, sino que se concentran en las acciones sobre la información, siguiendo un enfoque de múltiples capas de componentes para propiciar la infraestructura necesaria para el desarrollo de aplicaciones que sustenten el quehacer de las empresas.

Los cinco componentes y tecnologías principales de cualquier ECM, incluyendo Alfresco pueden ser clasificados como (KAMPFFMEYER, 2006), ver figura 1:

- ✓ captura
- ✓ gestión
- ✓ almacenamiento
- ✓ entrega o presentación
- ✓ preservación

Los campos de acción tradicionales:

- ✓ Gestión documental
- ✓ Gestión del conocimiento incluyendo la colaboración
- ✓ Gestión del contenido *web*

¹ Instrumento que permite la compilación de todos los trabajos realizados por los estudiantes durante un curso o disciplina. En él pueden ser agrupados datos de visitas técnicas, resúmenes de textos, proyectos, informes, anotaciones diversas, en él se incluyen, también, las pruebas y las autoevaluaciones de los estudiantes.

Capítulo 1. Fundamentación teórica

- ✓ Gestión de archivos
- ✓ Gestión de procesos de negocio y/o flujos de trabajo

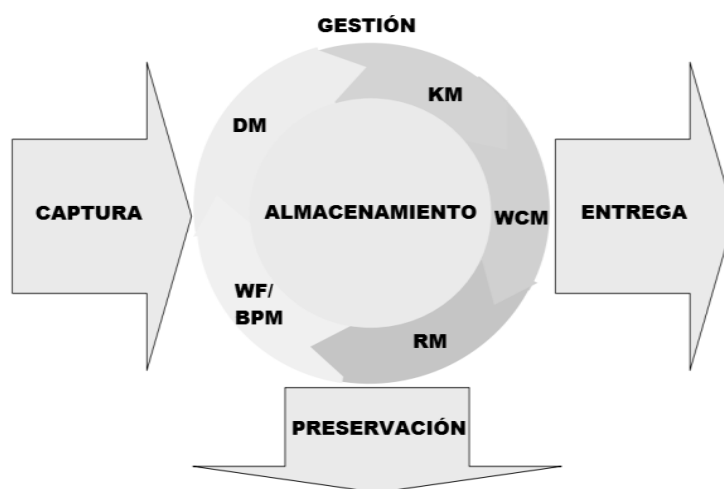


Figura 1. Componentes del ECM

1.3.2 El modelo de seguridad

Usuarios y grupos

En el contexto de Alfresco – y de otros tantos sistemas informáticos – un usuario es aquella persona o entidad que hace uso directo del sistema. Un grupo de usuarios como su nombre lo indica es un conjunto de usuarios que pertenecen a una misma estructura o poseen la misma responsabilidad o cumplen las mismas funciones sobre un área determinada. Los usuarios son miembros individuales mientras que los grupos son categorizaciones lógicas de los usuarios (POTTS, 2008).

En Alfresco, un usuario se identifica mediante un identificador único, también conocido como un log-in ID. Un usuario puede pertenecer a más de un grupo y sub-grupo como se muestra en la Figura 2, por ejemplo, el usuario Isaylín Batista Rosa pertenece a dos grupos: Ejecutivo e Ingeniería.

Capítulo 1. Fundamentación teórica

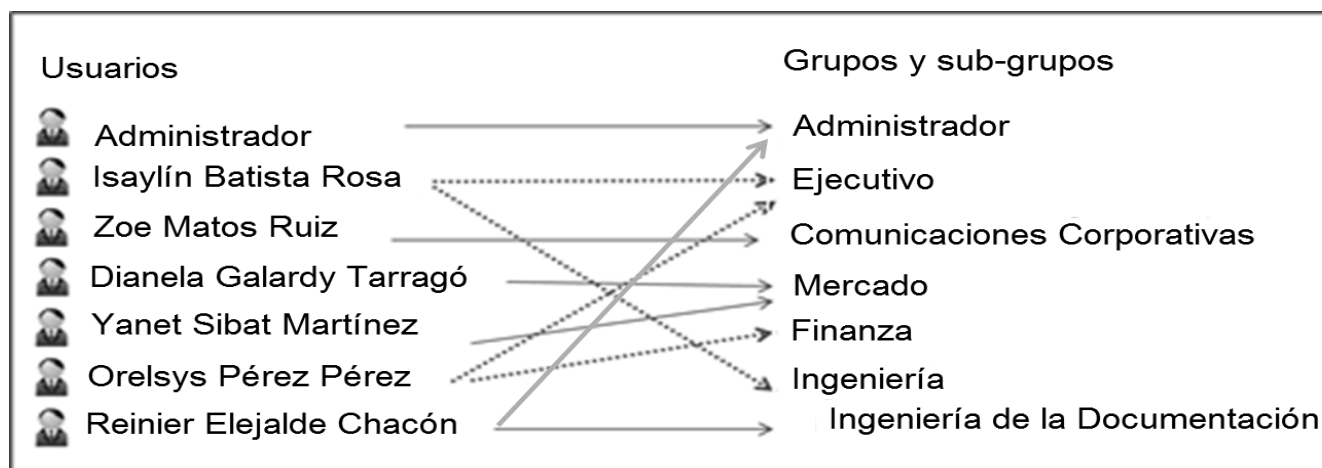


Figura 2. Usuarios, grupos y sub-grupos de usuarios

Permisos y roles

Los permisos definen los derechos de acceso sobre los espacios y contenidos. Se identifican por una cadena. Un permiso, ejemplo ReadChildren (Lectura de un nodo hijo), puede ser concedido o denegado a un usuario, grupo, administrador y propietario. Los hijos de un nodo, sub-carpetas o archivos en una carpeta, heredarán por defecto los permisos de su padre (la carpeta que lo contiene). Así, de forma predeterminada los archivos dentro de una carpeta heredarán los permisos de esa carpeta (POTTS, 2008).

A continuación en la tabla 1 se muestran los permisos por defecto sobre espacios:

Tabla 1. Permisos por defecto en el Alfresco

Nombre del permiso	Descripción	Se aplica sobre:
ReadProperties	Leer las propiedades	Espacio y contenido
ReadChildren	Ver el contenido de un espacio	Espacio
WriteProperties	Modificar el valor de las propiedades	Espacio y contenido
DeleteNode	Eliminar Nodo	Espacio y contenido
DeleteChildren	Eliminar el contenido y sub-espacios de un espacio	Espacio
CreateChildren	Crear contenido dentro de un espacio	Espacio
ReadContent	Ver el contenido	Contenido
WriteContent	Modificar contenido	Contenido
SetOwner	Cambiar el propietario	Contenido

Capítulo 1. Fundamentación teórica

Los roles o funciones son las colecciones de los permisos que pueden ser asignados a un usuario o grupo de usuarios sobre un nodo. Alfresco ofrece soporte para las siguientes funciones (POTTS, 2008):

- ✓ Lector-- ver el contenido
- ✓ Editor--puede leer y editar el contenido
- ✓ Contribuyente--puede leer y agregar contenido
- ✓ Colaborador--puede leer, editar y agregar contenido
- ✓ Coordinador--puede leer, editar, agregar y eliminar contenido (acceso pleno).

A continuación en la tabla 2 y tabla 3 se muestran una relación de los permisos y roles que se pueden establecer sobre espacios y contenidos respectivamente.

Tabla 2. Relación de roles y permisos sobre los espacios

Permisos	Lector	Editor	Contribuyente	Colaborador	Coordinador
Ver contenido de un espacio	x	x	x	x	x
Ver propiedades de un espacio	x	x	x	x	x
Ver sub- espacios	x	x	x	x	x
Copiar	x	x	x	x	x
Crear contenido de un espacio			x	x	x
Crear sub-espacios			x	x	x
Editar propiedades de un espacio		x		x	x
Adicionar o editar usuarios de un espacio		x		x	x
Eliminar usuarios de un espacio					x
Adicionar o modificar reglas de un espacio		x		x	x
Eliminar reglas de un espacio					x
Cortar contenido o sub-espacios					x
Eliminar contenido o sub-espacio					x
Crear copia de trabajo		x		x	x
Modificar contenido		x		x	x
Tomar posesión					x

Capítulo 1. Fundamentación teórica

Tabla 3. Relación de roles y permisos sobre los contenidos

Permisos	Lector	Editor	Contribuyente	Colaborador	Coordinador
Leer contenido	x	x	x	x	x
Leer propiedades de un contenido	x	x	x	x	x
Copiar	x	x	x	x	x
Editar contenido		x		x	x
Editar propiedades		x		x	x
Versionar		x		x	x
Categorizar		x		x	x
Crear copia de trabajo		x		x	x
Cambiar		x		x	x
Tomar posesión					x
Cortar					x
Eliminar					x

Espacio

Un espacio en Alfresco es una carpeta con propiedades adicionales, tales como reglas de negocio y de seguridad. Al igual que en una carpeta, un espacio puede contener sub-espacios y cualquier tipo de contenido. Sobre cada espacio los usuarios podrán ejecutar diversas acciones tales como copiar, pegar y borrar, sin embargo, estas variarán en función de los permisos que estos posean sobre dichos espacios. Por ejemplo, si un usuario no tiene permiso para eliminar en un espacio, no verá el enlace Suprimir o icono en el menú Más acciones (SHARIFF, 2006).

Contenido

El contenido en el Alfresco es cualquier tipo de documento: archivos de Microsoft Office, de Open Office, PDF, HTML², XML³, texto, imagen, audio o vídeo. Cada elemento de contenido se compone de

²HTML: Lenguaje de marcado de hipertexto, hace referencia al lenguaje de marcado predominante para la elaboración de páginas web que se utiliza para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes.

Capítulo 1. Fundamentación teórica

dos elementos principales, el contenido en sí y la información sobre el contenido, llamada metadatos⁴ o propiedades. Por defecto, cada elemento de contenido tiene propiedades tales como título, descripción, autor y la información de seguimiento de auditoría, como creador, fecha de creación, modificador y fecha de modificación (SHARIFF, 2006).

1.3.3 Servicios de seguridad

Alfresco provee tres servicios fundamentales que intervienen en el control de acceso: `PersonService`, `AuthorityService` y `PermissionService`.

PermissionService

Esta API⁵ es responsable de (CARUANA, 2010):

- ✓ proporcionar permisos
- ✓ proporcionar una API para leer, establecer y eliminarlos permisos de un nodo
- ✓ proporcionar las funcionalidades necesarias para consultar, activar y desactivar la herencia de permisos sobre un nodo
- ✓ proporcionar una API para realizar consultas, activar y desactivarla herencia de permisos para un nodo
- ✓ determinar si un usuario tiene permisos sobre un nodo

PersonService (CARUANA, 2010)

En Alfresco los usuarios y grupos de usuarios pueden ser gestionados completamente en el repositorio o en alguna otra implementación como LDAP⁶ o NTLM⁷. Mediante esta API se puede acceder a los nodos de tipo `cm:person`⁸. De manera general, esta API es responsable de:

³XML: Lenguaje de marcado extensible, es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C), el cual brinda soporte a bases de datos, siendo útil cuando varias aplicaciones se deben comunicar entre sí o integrar información.

⁴ Los metadatos son datos altamente estructurados que describen información, contenido, la calidad, la condición y otras características de los datos. Es "Información sobre información" o "datos sobre los datos".

⁵ Interfaz de programación de aplicaciones (IPA) o API (del inglés *Application Programming Interface*) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro *software* como una capa de abstracción.

⁶ LDAP: Protocolo Ligero de Acceso a Directorios que hacen referencia a un protocolo a nivel de aplicación el cual permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

⁷ NTLM: En un entorno de red, NTLM se utiliza como protocolo de autenticación para las transacciones entre dos equipos en los que al menos uno de ellos ejecuta Windows NT 4.0 o una versión anterior.

Capítulo 1. Fundamentación teórica

- ✓ obtener la referencia a un nodo de tipo “cm:person” dado un nombre de usuario
- ✓ determinar si existe una persona (un nodo de tipo “cm:person”) asociado a un nombre de usuario dado
- ✓ suministrar la lista de propiedades asociadas a cada persona
- ✓ crear, modificar y eliminar la información personal

AuthorityService

AuthorityService es la API responsable de (CARUANA, 2010):

- ✓ crear y eliminar autoridades
- ✓ realizar consultas para obtener autoridades según diferentes criterios
- ✓ estructurar las autoridades en jerarquías
- ✓ encontrar todas las autoridades que se aplican al usuario autenticado
- ✓ determinar si el usuario autenticado actual tiene derechos de administrador

1.3.4 Funcionamiento del modelo de seguridad

Una vez que haya subido su contenido al Alfresco, el principal reto siguiente consiste en asegurarlo, ya que hay documentos confidenciales que no todo el mundo debería tener acceso. Para lograr una mayor seguridad de los contenidos almacenados, Alfresco se basa en dos conceptos fundamentales: autenticación y autorización.

La autenticación es el proceso a través del cual la identidad del usuario se valida, determinando de esta manera si la persona que intenta acceder al sistema es un usuario válido o no. Normalmente, se hace esto cada vez que se pone un nombre de usuario y una contraseña. Alfresco provee diferentes mecanismos de autenticación como son: el subsistema de autenticación interna y la integración con LDAP (CARUANA, 2010).

El proceso de autorización comienza después de que el usuario está autenticado, lo que significa que el sistema tiene reconocido que la persona es un usuario válido. Ahora el sistema conoce lo que este usuario en particular puede hacer y lo que no puede hacer. Alfresco impone la autorización mediante la asignación de un rol a un usuario específico (o grupo) para un espacio específico (o contenido).

Alfresco tiene un sistema de seguridad flexible que se puede utilizar para administrar la seguridad de su contenido. Este se puede traducir como:

⁸Estructura de datos que usa Alfresco para representar o gestionar la información relacionada a los usuarios.

Capítulo 1. Fundamentación teórica

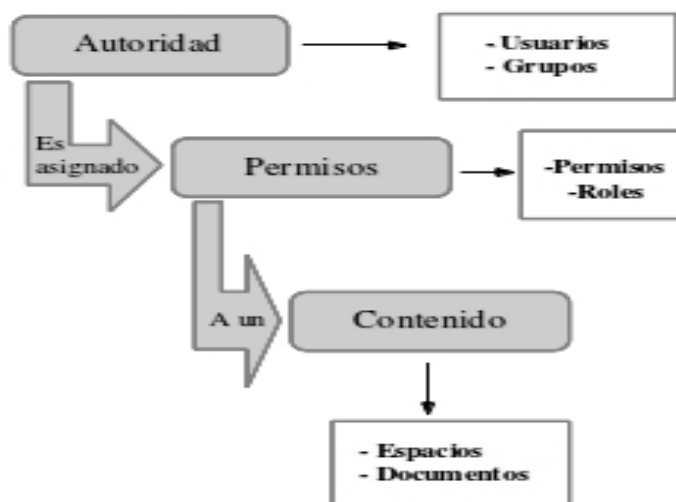


Figura 3. Funcionamiento del modelo de seguridad del Archivo UCI

1.4 Módulo Estructura y Composición del Sistema de Gestión Universitaria

Es un módulo del núcleo del Sistema Gestión Universitaria, gestiona la información referente a toda la estructura administrativa y la jerarquía de la misma, así como la asignación de responsabilidades a las estructuras establecidas por el centro de educación donde se decida implantar el Sistema de Gestión Universitaria.

Entidad

Representa una institución normalmente independiente. La UCI como sede central y las facultades regionales son ejemplos de “instituciones independientes”.

Estructura

No es más que una definición lógica donde se especifican objetos asociados a una entidad.

En la figura 4 se muestra el modelo de datos del módulo Estructura y Composición.

Capítulo 1. Fundamentación teórica

1.5 Módulo Estructura y Composición de UCI-FAR

UCI-FAR es un centro de desarrollo de soluciones informáticas para el MINFAR⁹. Entre sus aplicaciones fundamentales se encuentra Estructura y Composición, que según sus desarrolladores constituye la columna vertebral del resto de los sistemas. Su objetivo fundamental es centralizar la estructura organizativa de la entidad en la cual se encuentre instalada la solución. El módulo se compone de cinco estructuras fundamentales: entidades, organizaciones, cargos, medios materiales y agrupaciones. Sobre cada una de estas estructuras se pueden realizar las operaciones: crear, modificar, listar y eliminar. A este sistema se integran aplicaciones externas, entre ellas capital humano, la misma se retroalimenta de esta a partir de un conjunto de servicios *web*.

Luego de haber realizado un estudio al módulo Estructura y Composición de UCI – FAR, se llegó a la conclusión que la manera de comunicarse utilizada por el sistema anteriormente analizado con los sistemas externos es a través de servicios *web*. Esta vía de comunicación que emplea se convierte en una funcionalidad que puede ser utilizada para la integración entre el Archivo Universitario y el módulo Estructura y Composición del Sistema de Gestión Universitaria.

1.6 Tecnologías de Servicios Web

Definición

De las tantas definiciones que brindan especialistas en esta panorámica, pudiera decirse que un servicio *web* no es más que una colección de protocolos y estándares que se utilizan para el intercambio de datos entre distintas aplicaciones, que pueden estar desarrolladas en distintos lenguajes de programación y ejecutarse en distintas plataformas, a través de un lenguaje común para la comunicación.

La organización W3C¹⁰, por su parte, ofrece la siguiente definición:

“Un servicio web es un sistema software diseñado para soportar interacciones entre máquinas de manera interoperable en la red. Dispone de una interfaz descrita en un formato procesable de manera automática (específicamente WSDL¹¹). Los sistemas interaccionan con el servicio web de la manera

⁹ Ministerio de las Fuerzas Armadas Revolucionarias.

¹⁰ El *World Wide Web Consortium*, abreviado W3C, es un consorcio internacional que produce recomendaciones para la *World Wide Web*.

¹¹ En ocasiones leído como *wisdel* son las siglas de *Web Services Description Language*, un formato XML que se utiliza para describir servicios *web*.

Capítulo 1. Fundamentación teórica

indicada por su descripción, usando mensajes **SOAP**¹², típicamente sobre un soporte de **HTTP**¹³ con serialización en formato **XML** y siguiendo otros estándares relacionados con la web” (World Wide Web Consortium, 2002)

Ventajas

Las principales ventajas que se pudieran citar de los servicios web son, según (Funes, 2007):

- ✓ aportan una gran interoperabilidad entre distintas aplicaciones de *software*, independientemente del lenguaje de programación o la plataforma sobre la que se hallen
- ✓ los servicios *web* se fundamentan en estándares y protocolos basados en texto plano, lo que hace que sea más sencillo el acceso a su contenido y entender su funcionamiento
- ✓ dado que usan el protocolo HTTP, adquieren elementos de seguridad a partir del uso de los sistemas de seguridad firewall
- ✓ favorecen la integración de distintas aplicaciones que se encuentren en diferentes lugares geográficos para proveer servicios integrados
- ✓ el uso de protocolos estándar permite un alto grado de interoperabilidad entre plataformas de distintos fabricantes
- ✓ son accesibles desde una amplia gama de dispositivos, solo es necesario que estos puedan conectarse a Internet, lugar en el que se encuentran desplegados los servicios.

Inconvenientes

De la misma manera que se pueden citar ventajas de los servicios web, es importante tener en cuenta alguna de las desventajas que de este tipo de tecnologías se han evidenciado (Funes, 2007):

- ✓ ofrecen un bajo rendimiento en comparación con los modelos de computación distribuida. Esto es consecuencia de usar formatos basados en texto, dentro de los objetivos de XML no se encuentra la concisión ni la eficacia de procesamiento
- ✓ actualmente el uso de transacciones en entornos SOA (Arquitectura Orientada a Servicios) no está bien definido por lo que no se puede comparar con tecnologías rivales
- ✓ es necesario que el equipo desde el cual se invoca un servicio web disponga de conectividad, estos no pueden funcionar si no se dispone de conexión a la red o si en ésta se producen cortes esporádicos de conectividad

¹² SOAP: Es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.

¹³ HTTP: Protocolo usado en cada transacción de la World Wide Web.

Capítulo 1. Fundamentación teórica

✓ para algunos lenguajes de programación aún existe poca información.

Protocolos y estándares

En el desarrollo de servicios *web* se utilizan un conjunto de protocolos y estándares que pretenden dar uniformidad a los mismos, a continuación se plasman las observaciones tomadas de (Funes, 2007).

HTTP

Es el protocolo usado en las diferentes transacciones que se realizan en la *web* y fue desarrollado conjuntamente por el consorcio W3C y la IETF¹⁴.

El protocolo define la sintaxis y la semántica que utilizan los elementos *software* en arquitecturas *web* (clientes, servidores, *proxies*, entre otros) para comunicarse. Puede estar implementado encima de cualquier otro protocolo de *Internet* o en otras redes, dado que este tan solo presupone que existe una capa de transporte.

HTTP está orientado a las transacciones y se basa en el esquema petición-respuesta entre clientes y servidores. Así, el cliente, llamado *user-agent*, efectúa una petición desde un navegador, sobre un determinado recurso identificado por una URL¹⁵. El servidor que da respuesta, contiene recursos (archivos HTML e imágenes) y se denomina *origin server*. Entre el *user-agent* y el *origin server* pueden existir varios intermediarios, como por ejemplo *proxies*, *gateways* o túneles.

HTTP es un protocolo sin estado, es decir, no guarda informaciones sobre conexiones anteriores. El concepto de estado sí es importante en el desarrollo de aplicaciones *web* y para ello se usan las *cookies*, que son un conjunto de informaciones que el servidor puede almacenar en el cliente.

XML

XML es un lenguaje de marcas¹⁶ de propósito general. Puede ser clasificado además como un estándar abierto recomendado por W3C. Se clasifica como extensible dado que permite al usuario definir sus propias etiquetas. Su principal propósito es facilitar la compartición de datos estructurados sobre diferentes sistemas de información, más concretamente vía *internet*. Además se usa tanto para codificar documentos como para serializar datos.

¹⁴ El IETF o *Internet Engineering Task Force* es el grupo de la ISOC responsable del funcionamiento efectivo de *Internet* y la resolución de todos los aspectos de arquitectura y protocolos a corto y mediano plazo.

¹⁵ URL son las siglas de Localizador de Recurso Uniforme (en inglés *Uniform Resource Locator*), la dirección global de documentos y de otros recursos en la *World Wide Web*.

¹⁶ Es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación.

Capítulo 1. Fundamentación teórica

SOAP

SOAP es un protocolo estándar creado principalmente por Microsoft e IBM (de sus siglas en inglés, *International Business Machines*) y regulado actualmente por el consorcio W3C, que define cómo dos objetos en diferentes procesos de la red pueden establecer una comunicación vía HTTP/s mediante el intercambio de datos XML. Su uso es muy común en entornos de servicios web, describiendo cómo deben ser los mensajes a intercambiar entre el proveedor de servicios web y sus consumidores.

WSDL

WSDL es el lenguaje basado en XML que ofrece un modelo para describir la interfaz pública de los servicios *web*. Además, detalla la forma de acceder a los mismos, requisitos del protocolo y el formato de los mensajes a intercambiar.

Describe el modo de comunicación con los servicios, es decir, los requisitos del protocolo junto al formato de los mensajes necesarios para interactuar con los servicios listados en el registro. Las operaciones y mensajes soportados se describen de forma abstracta y posteriormente se ligan o vinculan a un protocolo y formato de mensajes establecido.

1.7 Metodología de desarrollo de *software*

En el libro *“El proceso unificado de desarrollo de software”* los autores Jacobson, Booch y Rumbaugh refieren que *“un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software”* (JACOBSON, 2004). Al mismo tiempo, lo definen como un marco de trabajo que:

- ✓ proporciona una guía para ordenar las actividades
- ✓ dirige las tareas de cada integrante del equipo
- ✓ especifica los artefactos que deben desarrollarse
- ✓ ofrece criterios para el control y la medición de los productos y actividades del proyecto

La experiencia adquirida por los analistas de *software* durante la aplicación de metodologías como guía para el desarrollo de aplicaciones, ha demostrado, que la clave del éxito de un proyecto puede estar estrechamente relacionada con su acertada selección. Tomando como punto de partida la premisa anterior, en primer lugar se analizó la metodología (RUP) por la que se rige el desarrollo del Archivo Universitario. En un segundo momento se hizo un estudio de otras metodologías ágiles para finalmente seleccionar cuál de todas estas usar durante el desarrollo del diseño de la solución propuesta.

A continuación se presenta una breve caracterización de las metodologías de *software* estudiadas:

Capítulo 1. Fundamentación teórica

SCRUM (Proyectalis, 2010)

SCRUM es una metodología ágil que define que el proceso de desarrollo de *software* debe ser iterativo e incremental. Para ello establece que en los inicios del proyecto se especifican los requisitos funcionales y no funcionales que debe cumplir el sistema, los que se detallan de acuerdo con las convenciones de la organización mediante casos de uso, diagramas de flujos de datos y tareas.

La especificación de requisitos se define durante reuniones de planeamiento y a partir de él se especifican las iteraciones a través de las cuales debe evolucionar el producto. Dentro de cada iteración se realiza diariamente una reunión para verificar el avance que se ha tenido y los obstáculos que se han presentado. Al finalizar la misma (la iteración) se realizan reuniones de iteración para evaluar los artefactos construidos y planificar la próxima iteración.

La idea principal de esta metodología es maximizar la realimentación al tiempo que se desarrolla el producto, pudiendo corregir problemas y mitigar riesgos desde temprano. Dos de sus ventajas fundamentales son:

- ✓ la posibilidad de ajustar las funcionalidades tomando como base las necesidades del cliente
- ✓ permite la visualización del avance del proyecto diariamente

Programación Extrema (XP) (Agile Modeling, 2008)

Programación Extrema o *Extreme Programming* (XP), es una metodología ágil centrada fundamentalmente en potenciar y promover el trabajo en equipo a la vez que se incrementa el nivel de aprendizaje de los desarrolladores y favorece un buen clima de trabajo, factor clave para el éxito en un proyecto de desarrollo.

XP está pensada para ser utilizada en proyectos con equipos de desarrollo pequeños y con plazos de entrega corto, siendo adecuado su uso en proyectos donde los requisitos son imprecisos, ambiguos y muy cambiantes. En su aplicación es de gran importancia la disposición del cliente para con el equipo de desarrollo como una de las medidas que propone la metodología para minimizar los riesgos de fallos durante el proceso.

Esta metodología propone el uso de historias de usuario para describir los escenarios sobre el funcionamiento del *software* constituyendo la base para el desarrollo del mismo. Con el fin de obtener una mayor calidad del código, la implementación se realiza siempre en parejas, aunque estas se rotan durante todo el proyecto permitiendo que cada integrante del equipo aprenda a trabajar con los demás y con cada uno de los componentes del *software*. La codificación puede ser modificada en cualquier momento si un desarrollador lo necesita dado que el código pertenece a todos, lo que propicia que el equipo entero posea conocimiento de la aplicación completa.

Capítulo 1. Fundamentación teórica

Proceso Unificado Rational (RUP)

Especialistas como Jacobson, Booch y Rumbaugh (2000) expresan que “*El Proceso Unificado de Rational (RUP) es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, tipos de organizaciones, niveles de aptitud y tamaños de proyecto. Está basado en componentes software, sin embargo los verdaderos aspectos definitorios son: dirigido por casos de uso, centrado en la arquitectura e iterativo e incremental*” (JACOBSON, 2004). El desarrollo de un sistema siguiendo RUP se divide en cuatro fases las cuales se ejecutan una o varias veces en dependencia del tamaño del proyecto, estas son:

- ✓ Inicio - el objetivo de esta fase es desarrollar la visión del proyecto
- ✓ Elaboración - en esta fase se especifican la mayoría de los casos de uso del producto y se diseña la arquitectura del sistema
- ✓ Construcción - tiene como fin completar el desarrollo del sistema basado en la línea base de la arquitectura
- ✓ Transición - se corrigen los problemas e incorporan mejoras sugeridas por los usuarios, en sí se garantiza que el *software* esté listo para entregar al usuario. El objetivo es conseguir la liberación del producto

Dentro de cada una de las fases se sigue un modelo de cascada para los flujos de trabajo que constituyen nueve actividades a realizar en cada fase, seis de control del proceso (modelado de negocio, requisitos, análisis y diseño, implementación, prueba y despliegue) y tres de apoyo (gestión de configuración y cambios, gestión de proyecto y ambiente).

Justificación de la selección

A continuación en la tabla 4 se realiza una comparación de las diferentes metodologías de desarrollo de *software* estudiadas.

Capítulo 1. Fundamentación teórica

Tabla 4. Comparación de aspectos entre metodologías

Criterio de comparación	SCRUM	XP	RUP
Calidad de la documentación	No se genera toda la evidencia requerida. En ocasiones es necesario complementarlo con otra metodología.	Se evita la generación de grandes cúmulos de documentación.	Artefactos bien definidos por cada una de las actividades que se proponen.
Condiciones o forma de trabajo	Exige la presencia de todo el equipo de desarrollo. No puede ser aplicado en proyectos cuyos recursos estén distribuidos geográficamente.	Exige la presencia del cliente en todo momento. Se basa en el desarrollo en dúos.	No presenta limitante en este aspecto.
Complejidad y duración del proyecto	Está pensado para ser aplicado en proyectos de corta duración.	Está pensado para ser aplicado en proyectos de corta duración.	Está pensado para ser aplicado en proyectos de gran envergadura y tiempo de duración prolongado sin embargo es perfectamente ajustable a proyectos de menor alcance tanto de tiempo como de complejidad.
Estabilidad de los requisitos	Aplicable en proyectos donde los requisitos son imprecisos y cambiantes.	Aplicable en proyectos donde los requisitos son imprecisos y cambiantes.	Está pensada para ser aplicada en proyectos donde los requisitos no cambian con frecuencia sin embargo con una

Capítulo 1. Fundamentación teórica

			buena gestión de configuración y riesgos se pueden mitigar tales cambios.
Experiencia en el uso por parte del equipo	Ninguna	Ninguna	Como promedio los especialistas del proyecto tienen 4 años de experiencia en la aplicación de la metodología.

Una vez concluida la comparación realizada, según las características de la solución a diseñar y las capacidades de cada una de las metodologías se decide usar RUP.

Actualmente la UCI como centro productivo está acometiendo un proyecto de mejora de sus procesos basado en el modelo CMMI (*Capability Maturity Model Integration*) con el objetivo de crecerse como organización productora de *software*. Por esta razón se complementará la metodología a utilizar con nivel 2 de CMMI con vistas a mejorar el ciclo de vida dentro del desarrollo de *software* y alcanzar una mayor calidad.

1.8 Lenguaje de modelado

Durante el desarrollo de un *software* es necesario representar de manera sencilla lo que se pretende realizar, de forma tal que permita a los usuarios entender su estructura, contenido y organización, además de señalar cambios si así lo desea. La clave está en organizar el proceso de diseño de tal forma que los analistas, clientes, desarrolladores y otras personas involucradas en el desarrollo del sistema lo comprendan (LARMAN, 1999).

Lenguajes de modelado quizás existen cientos, sin embargo, una vez seleccionada la metodología RUP se hace necesario reflexionar sobre la propuesta de los autores Booch, Jacobson y Rumbaugh quienes refieren que “el proceso unificado de desarrollo utiliza el Lenguaje Unificado de Modelado (*Unified Modeling Language, UML*) para preparar todos los esquemas de un sistema. UML constituye una parte esencial de dicho proceso” (JACOBSON, 2004).

Capítulo 1. Fundamentación teórica

Por su parte Craig Larman define UML como “*lenguaje o notación estándar que permite especificar, visualizar y construir los artefactos de los sistemas de software orientados a objetos*” (LARMAN 2004). Basado en lo antes expuesto y tomando como punto de referencia la experiencia adquirida en el uso de este lenguaje por el equipo de proyecto se decidió usar UML para generar cada uno de los artefactos que propone la metodología RUP a lo largo del proceso de desarrollo.

1.9 Herramienta CASE

Las herramientas CASE (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Ordenador) establecen un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Su objetivo es facilitar la automatización del ciclo de vida del desarrollo de un *software* en su totalidad o algunas de las fases (Scribd, 2010).

Seleccionada la metodología RUP como guía para el desarrollo de *software* y UML por consiguiente para el modelado de los diferentes artefactos definidos para cada actividad correspondía identificar la herramienta CASE más idónea como apoyo al proceso. Al igual que metodologías y lenguajes para el modelado, herramientas de este tipo hay varias, sin embargo, en este caso en lugar de hacer un estudio de las diferentes herramientas CASE se verificó que Visual Paradigm – *propuesta en el marco de trabajo del grupo de proyecto* – se ajusta a las necesidades específicas para el diseño de la propuesta de solución.

A continuación se presentan algunas características que demuestran tal hecho (Visual Paradigm, 2004):

- ✓ es una herramienta que da soporte única y exclusivamente a UML y cada versión del producto se libera con la última versión estable de UML
- ✓ contiene los componentes necesarios para dar soporte al ciclo de vida completo desarrollo del *software* - desde la concepción del negocio hasta el despliegue, atravesando el análisis, diseño, construcción y pruebas
- ✓ contiene los componentes necesarios para dar soporte al ciclo de vida completo. Posee características gráficas que facilitan la realización de cada uno de los diagramas, cada uno de los cuales pueden ser distribuidos automáticamente, reorganizando figuras y conectores
- ✓ posibilita la generación automática de código a partir de los diagramas, de la misma manera es posible aplicar ingeniería inversa, obteniendo los diagramas a partir de código fuente escrito en lenguajes como Java, C++ .NET

Capítulo 1. Fundamentación teórica

- ✓ permite la creación del modelo entidad-relación, transformación de estos diagramas en secuencias de sentencias – generalmente SQL – para la generación de la base de datos. En este mismo aspecto también es posible aplicar ingeniería inversa
- ✓ es posible establecer un entorno colaborativo gracias a su integración con Subversión. De esta manera, varias personas pueden trabajar al mismo tiempo sobre un mismo proyecto
- ✓ el producto está disponible para múltiples plataformas *software* - Windows (98+), Linux, Mac OS X, Solaris, entre otros
- ✓ permite la integración con diversos IDE's (Entornos de Desarrollo Integrado) como Netbeans y Eclipse, precisamente los dos más usados en el proyecto

1.10 Solución para el diseño de prototipos de interfaces de usuario

Evolus Pencil es una extensión de Firefox que se utiliza para el diseño de los prototipos de las interfaces de usuario, se caracteriza por (Sketching, 2010):

- ✓ brindar un conjunto de componentes como -- entradas de texto, íconos y botones
- ✓ ser multi-página -- creación simultánea de varios documentos
- ✓ edición en pantalla de los elementos de texto
- ✓ permitir exportar imágenes al formato PNG, HTML o PDF
- ✓ permitir operaciones estándar de dibujo -- alineado, escalado y rotación
- ✓ ser multi-plataforma
- ✓ posibilitar, a través de las propiedades de los componentes, cambiar el estilo al diseño.

Esta herramienta se utilizará en la construcción de la interfaz gráfica

Análisis del capítulo

- ✓ El planteamiento de los principales conceptos relacionados con la gestión documental y los archivos universitarios contribuyó a un acercamiento y entendimiento de la importancia que tiene el control de acceso a la documentación producida y generada como resultado de los procesos universitarios.
- ✓ El análisis del modelo de seguridad del ECM Alfresco facilitó la comprensión de las estructuras de datos y los mecanismos que posee actualmente el Archivo Universitario para controlar el acceso a los diferentes contenidos que se almacenan en el repositorio como evidencia de las actuaciones o procesos universitarios.

Capítulo 1. Fundamentación teórica

- ✓ El estudio de los módulos Estructura y Composición del Sistema de Gestión Universitaria y de UCI-FAR evidenció la necesidad de integrar el Archivo Universitario a un sistema centralizado de control de la estructura de la universidad, así como de las responsabilidades de los ciudadanos universitarios en cada una de estas estructuras.
- ✓ El estudio realizado sobre las tecnologías de servicios web demostró la posibilidad de comunicar el Archivo Universitario con el Módulo Estructura y Composición del Sistema de Gestión Universitaria.

Capítulo 2. Propuesta del sistema

Capítulo 2. Propuesta del sistema

Partiendo del análisis de la situación problemática se describe la solución propuesta, proporcionando un mejor entendimiento de lo que se desea que realice el sistema. También se especifican los requisitos funcionales y no funcionales que debe cumplir el mismo, así como la representación y descripción de los casos de uso del sistema.

2.1 Situación problemática

Al especialista encargado de gestionar el control de acceso sobre cada uno de los espacios definidos en el Archivo Universitario se le hace muy engorroso llevar a cabo dicha tarea. Actualmente, la gestión de usuarios, grupos de usuarios y permisos se realiza de forma manual por lo que resulta complejo su control, pues no existe un mecanismo que actualice esta información de forma automática. Esto puede traer como consecuencia que en un instante dado existan personas que ya no pertenezcan a un área determinada y cuenten aún con los privilegios que la misma les proporciona. Por el nivel de importancia que tiene esta tarea se hace necesaria la utilización de herramientas que faciliten el flujo de información haciendo su organización y trasmisión más rápida. La falta de medios informáticos que permitan llevar a cabo las actualizaciones de los usuarios, grupos de usuarios y permisos, puede hacer que no se garantice la confidencialidad, integridad y disponibilidad de la información en el Archivo Universitario.

2.2 Objeto de automatización

Con el presente trabajo se propone el análisis y diseño de un módulo capaz de realizar todo el proceso que involucra la correcta actualización de usuarios, grupos de usuarios y permisos en el Archivo Universitario. Los procesos que serán objeto de automatización son la gestión de usuarios, grupos de usuarios y la asignación de permisos a estos, según el rol que desempeñe en un área determinada. De este modo se tendrán registrados los permisos que realmente posee un individuo sobre un espacio determinado, evitando la posibilidad de errores que pueden incidir negativamente en la tramitación de la información que se maneja.

2.3 Propuesta de solución

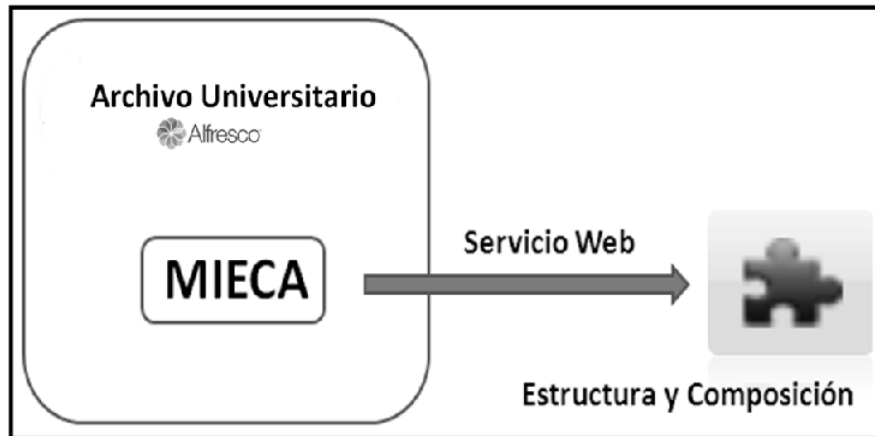


Figura 5. Módulo de Integración del Archivo Universitario con Estructura y Composición

Después de haber realizado el estudio y análisis de las dificultades existentes con la actualización de usuarios, grupos de usuarios y permisos en el Archivo Universitario, se propone el análisis y diseño de un Módulo de Integración del Archivo Universitario con Estructura y Composición (MIECA), el cual se encontrará embebido en el ECM Alfresco del Archivo Universitario, brindará funcionalidades para acceder a los servicios que proporciona el módulo Estructura y Composición, con el fin de conocer la responsabilidad que - *en un momento determinado* - tienen los usuarios en un área específica de la universidad y por ende los permisos con los que cuenta para acceder a un determinado espacio. Estos servicios brindarán información referente a las estructuras, usuarios que la componen y responsabilidades de los mismos sobre estas. Los servicios web a consumir son:

Tabla 5. Servicio web “Obtener_estructuras”

Nombre del servicio web	Obtener_estructuras
Descripción	Devuelve el conjunto de estructuras pertenecientes a una entidad dada.
Entradas	Entidad De la entidad se conoce: id, nombre.
Salidas	Listado de estructuras De cada estructura se conoce: id, código, nombre, abreviatura, estructura_padre ¹⁷ y estructuras_hijas ¹⁸ .

¹⁷ Estructura que contiene al menos una estructura.

¹⁸ Son las estructuras que se encuentran contenidas dentro de una entidad.

Capítulo 2. Propuesta del sistema

Tabla 6. Servicio web “Obtener _composición”

Nombre del servicio web	Obtener _composición
Descripción	Devuelve la composición de una estructura en concepto de personas.
Entradas	id_estructura
Salidas	Listado de usuarios Del usuario se conoce: id_expediente, usuario, nombre, apellidos, correo_electrónico.

Tabla 7. Servicio web “Obtener _permisos”

Nombre del servicio web	Obtener _permisos
Descripción	Devuelve los permisos que tiene un usuario sobre una estructura dada
Entradas	id_usuario, id_estructura
Salidas	Listado de permisos

Tabla 8. Servicio web “Obtener _estructuras”

Nombre del servicio web	Obtener _estructuras_usuario
Descripción	Devuelve las estructuras de las cuales un usuario es integrante.
Entradas	Usuario
Salidas	Listado de estructuras

Tabla 9. Servicio web “Obtener _usuarios_modificados”

Nombre del servicio web	Obtener _usuarios_modificados_responsabilidad
Descripción	Devuelve aquellos usuarios cuyas responsabilidades han cambiado en las estructuras a la cuales pertenece.

Capítulo 2. Propuesta del sistema

Entradas	Estructura
Salidas	Listado de usuarios

Tabla 10. Servicio web “Obtener_usuarios”

Nombre del servicio web	Obtener _usuarios_permiso
Descripción	Devuelve los usuarios que poseen determinado permiso, independientemente de la estructura de la cual formen parte.
Entradas	Responsabilidad
Salidas	Listado de usuarios

Tabla 11. Servicio web “Obtener_estructuras_hijas”

Nombre del servicio web	Obtener _estructuras_hijas
Descripción	Devuelve las estructuras contenidas dentro de la estructura padre.
Entradas	Estructura padre
Salidas	Listado de estructuras hijas

Debido a que los espacios en el Archivo Universitario no están en correspondencia con las áreas de Estructura y Composición, se propone desarrollar una funcionalidad que permita asociar las áreas de Estructura y Composición con los espacios en el Archivo Universitario, para tener identificada que una determinada estructura está asociada a un espacio en el Archivo. Una vez realizada esta operación se guardará en un fichero XML la asociación realizada. Luego, a través de eventos y tareas programadas

Capítulo 2. Propuesta del sistema

se actualizarán los usuarios, grupos de usuarios y permisos de determinadas áreas, según los criterios que se hayan definido para ella. Los eventos se ejecutarán cada cierto tiempo, haciendo uso de una o varias tareas programadas, sobre una o varias estructuras definidas en las tareas. Para la ejecución de las tareas programadas se hará uso del mecanismo que provee el ECM Alfresco para la gestión de tareas programadas.

El módulo permitirá además, realizar auditorías a los cambios ocurridos en los permisos de una persona determinada y para ello se propone el uso de ficheros (log), registrando en ellos los cambios ocurridos en las ejecuciones de las tareas programadas desprendidas por los eventos. Los ficheros poseerán la información referente a los usuarios que se le concedió, modificó o eliminó un permiso, cuál se le asignó o por cuál se modificó, la fecha en que se actualizaron estos permisos, la hora y la tarea programada que se ejecutó.

2.4 El contexto del sistema. Modelo de dominio

Los artefactos que propone la metodología RUP para comprender el contexto del sistema son el modelo del dominio y el modelo del negocio. Para realizar el modelo del negocio es necesario conocer cómo se realizan cada uno de los procesos y estados del negocio que se quiere automatizar, así como las competencias requeridas en cada proceso: sus trabajadores, responsabilidades y las operaciones que llevan a cabo.

El modelo del negocio actual presenta un bajo nivel de estructuración, siendo difícil identificar las personas que participan en las distintas actividades en cada uno de los procesos, por lo que se decide realizar el modelo de dominio que RUP propone para estos casos.

“Un modelo de dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las cosas que existen o los eventos que suceden en el entorno en el que trabaja el sistema” (JACOBSON, 2004).

2.4.1 Diagrama de clases del modelo de dominio

En la siguiente figura 6, se presenta el modelo del dominio mediante un diagrama UML. Su propósito fundamental es contribuir a la comprensión del contexto del sistema a través de los objetos identificados, así como de los eventos que suceden en el entorno en que se desarrolla tal problema.

Capítulo 2. Propuesta del sistema

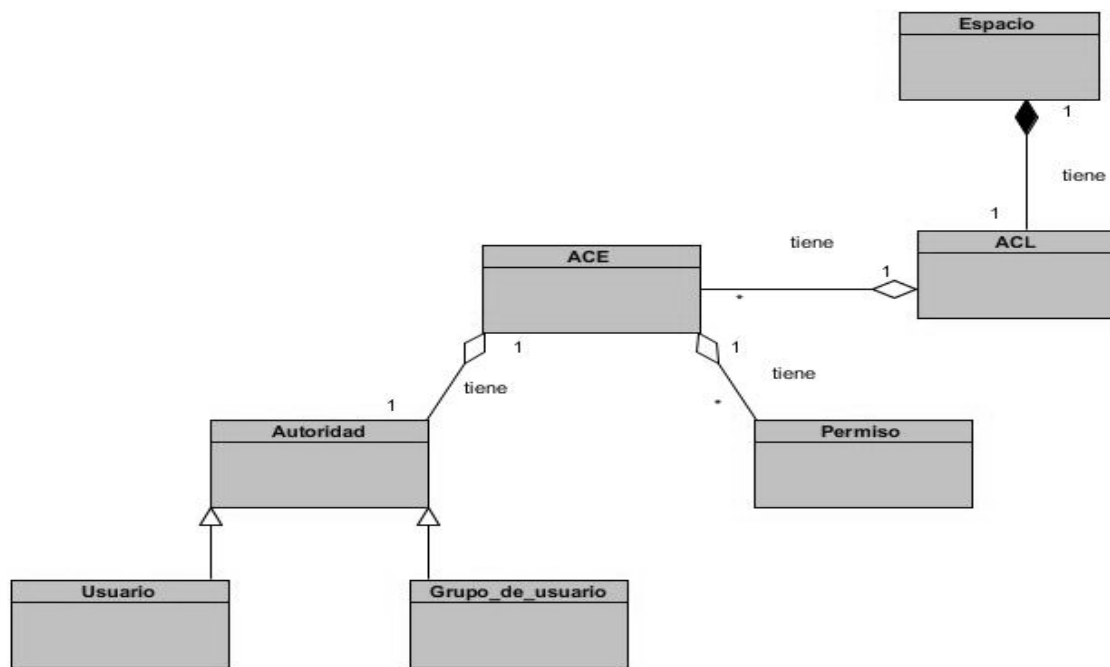


Figura 6. Modelo de dominio

Definición de las clases del modelo de dominio

- ✓ **Autoridad**: es una persona o grupo de personas.
- ✓ **Usuario**: persona que interactúa con el sistema y sus funcionalidades y puede realizar cambios en los espacios en dependencia del rol que posea.
- ✓ **Grupo_de_usuario**: grupo de personas agrupadas por la estructura a la que pertenecen o por la responsabilidad que tienen sobre una determinada área.
- ✓ **Espacio**: es una carpeta donde se almacenan documentos.
- ✓ **Permiso**: los permisos definen los derechos de acceso sobre los espacios y contenidos. Son asignados a los usuarios y grupos de usuarios según la responsabilidad que posea en una estructura dada.
- ✓ **ACE**¹⁹: una entrada de control de acceso asocia a una autoridad con uno o con varios permisos (rol).
- ✓ **ACL**²⁰: es una lista ordenada de entradas de control de acceso (ACE).

Descripción del modelo de dominio

¹⁹ Entrada de Control de Acceso.

²⁰ Lista de Control de Acceso.

Capítulo 2. Propuesta del sistema

Como se puede observar en la Figura 6, un espacio en el Archivo Universitario cuenta con una lista de control de acceso que a su vez está compuesta por un conjunto de entradas de control de acceso. Cada una de estas entradas agrupa una autoridad (usuario o grupo de usuario) y el conjunto de permisos que dicha autoridad tendrá sobre el espacio. De esta manera, se garantiza que un usuario puede tener una responsabilidad (determinada por permisos) dada sobre uno o varios espacios en el Archivo Universitario y sólo podrá ejecutar sobre dichos espacios las acciones según el rol que tenga sobre estos.

2.4.2 Especificación de los requisitos de *software*

Según el estándar 1233 de la IEEE: Guía para el desarrollo de Especificaciones de Requerimientos de Sistemas, un requisito se define como (IEEE, 1998):

- ✓ condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo
- ✓ condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.

Se puede concluir entonces que los requisitos de *software* son características y funcionalidades que debe cumplir un sistema. Están enfocados hacia todo lo que debe hacer el sistema, el usuario y los miembros del equipo de proyecto. Los requisitos pueden ser clasificados en requisitos funcionales y requisitos no funcionales.

2.5 Técnicas para la captura de requisitos

En principio, parece bastante simple preguntar al cliente, a los usuarios y a los que están involucrados en los objetivos del sistema o producto y sean expertos, investigar cómo los sistemas o productos se ajustan a las necesidades del negocio y finalmente, cómo el sistema o producto va a ser utilizado en el día a día. (PRESSMAN, 2005).

Es por ello por lo que surgen diferentes técnicas que ayudan a comprender el problema, proponer soluciones, negociar diferentes puntos de vista y finalmente especificar un conjunto básico de requisitos de la solución. Ejemplos de estas técnicas pueden ser: entrevistas, cuestionarios, tormentas de ideas, análisis de sistemas existentes, arqueología de documentos, prototipos, entre otras.

Para la realización del levantamiento de requisitos del módulo se aplicaron las técnicas de tormentas de ideas y prototipos.

Capítulo 2. Propuesta del sistema

Tormenta de ideas

Técnica basada en lluvias de ideas, implica tanto la generación como la reducción de ideas. Básicamente se busca que los involucrados en un proyecto desarrollen su creatividad, promoviendo la introducción de los principios creativos. Las ideas más creativas e innovadoras resultan con frecuencia de la combinación de ideas aparentemente sin relación (ZAPATA, 2007). El uso de esta técnica fue muy efectivo pues tanto el cliente como personal del proyecto aportaron ideas para la definición de las funcionalidades básicas que debe tener el módulo.

Prototipos

Un prototipo es un borrador de un producto potencial o de una parte del mismo. Es una simulación de los requisitos. En ocasiones los analistas no pueden continuar su trabajo porque les faltan datos. En esos casos el analista o el resto de las personas involucradas necesitan trabajar con algo más concreto que una lista de requisitos escritos y para esto utilizan un prototipo (SOMMERVILLE, 2005). El empleo de esta técnica se evidencia en el conjunto de prototipos que fueron presentados al cliente para realizar una evaluación de los mismos, también fueron utilizados para refinar los requerimientos del *software* a ser desarrollado.

2.6 Requerimientos funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir, no alteran la funcionalidad del producto, por lo que se mantienen invariables sin importarle con que propiedades o cualidades se relacionen (SOMMERVILLE, 2005).

Tabla 12. Requerimientos funcionales

No	Nombre del requisito	Descripción
RF 1	Asociar estructura con espacio	El sistema debe de garantizar que el administrador pueda asociar un área de Estructura y Composición con un espacio del Archivo Universitario.
RF 2	Modificar asociación estructura con espacio	El sistema debe de garantizar que el administrador pueda modificar una asociación existente entre un área de Estructura y Composición y un espacio en el Archivo Universitario.
RF 3	Eliminar asociación estructura con espacio.	El sistema debe de garantizar que el administrador pueda eliminar una asociación existente entre un área de

Capítulo 2. Propuesta del sistema

		Estructura y Composición y un espacio en el Archivo Universitario.
RF 4	Crear tarea programada	El sistema debe de garantizar que el administrador pueda crear tareas programadas, las cuales pueden consistir en: ✓ actualizar diariamente en el Archivo Universitario los usuarios a los cuales se les ha modificado su responsabilidad, independientemente de la estructura a la que pertenezca.
RF 5	Modificar tarea programada	El sistema debe de garantizar que el administrador pueda modificar tareas programadas.
RF 6	Eliminar tarea programada	El sistema debe de garantizar que el administrador pueda eliminar tareas programadas.
RF 7	Ejecutar evento	El sistema debe ejecutar automáticamente las tareas programadas por el administrador.
RF 8	Crear usuario	El sistema debe importar los usuarios desde Estructura y Composición.
RF 9	Modificar usuario	El sistema debe modificar de forma automática los usuarios según los cambios identificados en las estructuras y responsabilidades, así como sus datos personales.
RF 10	Eliminar usuario	El sistema debe modificar de forma automática los usuarios según los cambios identificados en las estructuras y responsabilidades, así como sus datos personales.
RF 11	Crear grupo de usuario	El sistema debe crear de forma automática los grupos de usuarios según la estructura o la responsabilidad.
RF 12	Modificar grupo de usuario	El sistema debe modificar de forma automática los grupos de usuarios según los cambios identificados en las estructuras o responsabilidades.

Capítulo 2. Propuesta del sistema

RF 13	Eliminar grupo de usuario	El sistema debe eliminar de forma automática los grupos de usuarios según los cambios identificados en las estructuras o responsabilidades.
RF 14	Crear permiso	El sistema debe importar los permisos desde Estructura y Composición.
RF 15	Modificar permiso	El sistema debe modificar de forma automática los permisos asociados a un usuario sobre un espacio según los cambios ocurridos.
RF 16	Eliminar permiso	El sistema debe eliminar de forma automática los permisos asociados a un usuario sobre un espacio según los cambios ocurridos.
RF 17	Realizar auditoría	El sistema debe registrar los cambios ocurridos a los permisos de los usuarios.

2.7 Requerimientos no funcionales

Los requisitos no funcionales son los requerimientos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de este, como la fiabilidad y el tiempo de respuesta (SOMMERVILLE, 2005).

Tabla 13. Requerimientos no funcionales

Requisitos no funcionales	
Usabilidad	
1	Utilizar el idioma español para los mensajes y textos de la interfaz.
2	Permitir auto-completado de algunos campos en la interfaz.
Seguridad	
3	Políticas de seguridad por usuarios y roles: El sistema debe contar con un grupo de políticas de accesibilidad a las diferentes funcionalidades del mismo en dependencia del nivel de autorización que presente un usuario determinado.

Capítulo 2. Propuesta del sistema

4	Registro sistemático de incidencias: El sistema debe ser capaz de registrar el accionar del usuario, así como permitir auditorías y exámenes de las trazas tanto en tiempo real como en históricos. Se precisa de un fichero log para la visualización de las mismas.
Restricciones de diseño	
5	Mantener un sistema de codificación estándar siguiendo las pautas establecidas en el documento de Línea Base de la Arquitectura.
6	Utilizar servidor de aplicaciones Tomcat.
7	Implementar el módulo en el lenguaje de programación Java.
8	Marco de trabajo Spring Framework.
Portabilidad	
9	La aplicación debe ser multiplataforma.
Soporte	
10	La estación de trabajo cliente debe tener instalado el navegador Mozilla Firefox.
Legales	
11	Las herramientas seleccionadas para el desarrollo del producto deben estar respaldadas por licencias libres, bajo las condiciones de <i>software</i> libre.
Interfaz	
12	Interfaz Web: La interfaz debe ser sencilla con colores de tonos nítidos y sin cúmulo de imágenes u objetos que distraigan al cliente del objetivo.

2.8 Especificación de requisitos como casos de uso

2.8.1 Definición de los casos de uso del sistema

“Cada forma en que los actores usan el sistema se representa con un caso de uso. Los casos de uso son fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un caso de uso especifica una secuencia de acciones que el sistema

Capítulo 2. Propuesta del sistema

“puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia” (JACOBSON, 2004).

A partir de los requisitos identificados, se propone el siguiente diagrama de casos de uso, que contiene los actores, casos de uso (que responden a 1 o más requisitos) y las relaciones que se establecen entre éstos, o sea, la forma en que los actores usan el sistema.

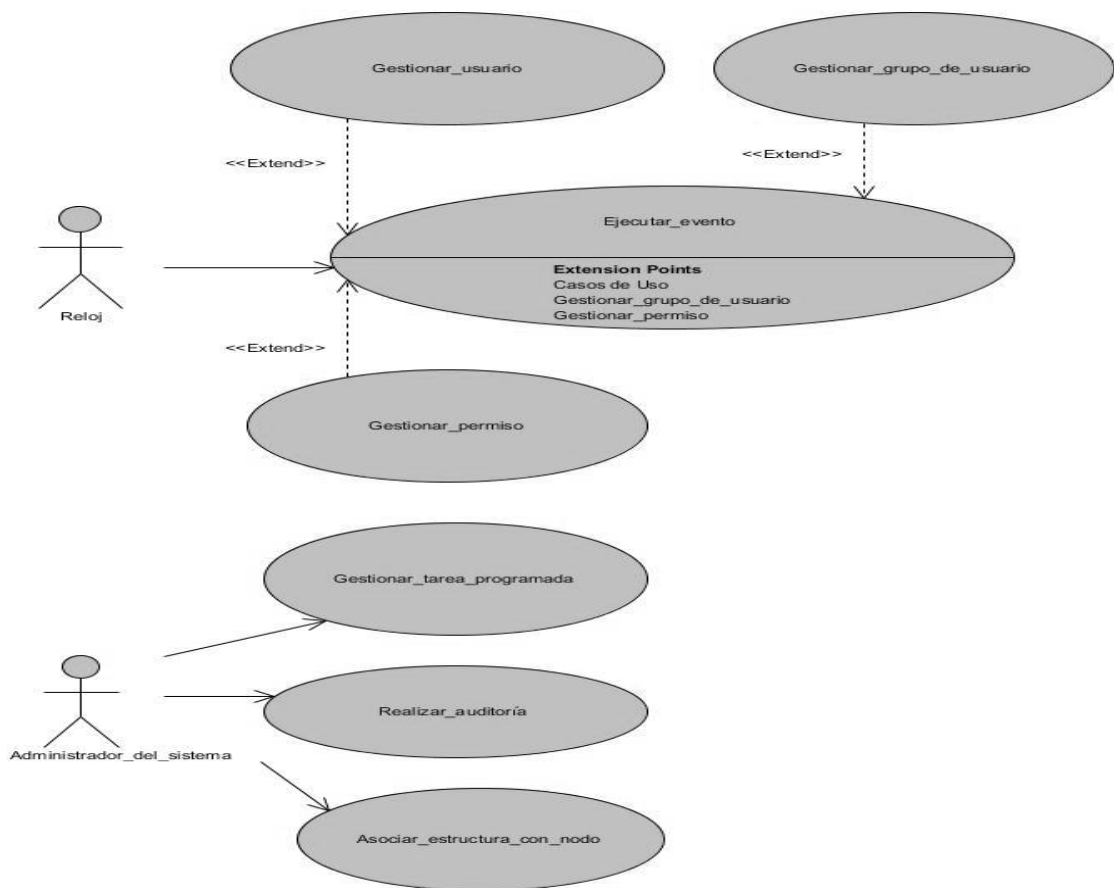


Figura 7. Diagrama de casos de uso

Capítulo 2. Propuesta del sistema

2.8.2 Definición de los actores

Un actor es un conjunto coherente de roles²¹ que los usuarios de casos de usos desempeñan cuando interactúan con estos casos de uso (JACOBSON, 2004).

Tabla 14. Actores del sistema

Actor	Justificación
Reloj	Actor que indica un temporizador en el sistema operativo del servidor, avisándole a las tareas programadas cuando tienen que ejecutarse.
Administrador del sistema	Es el encargado de gestionar (registrar, modificar, eliminar) las tareas programadas y realizar auditorías, además hace corresponder a cada área de Estructura y Composición un espacio en el Archivo Universitario.

2.8.3 Descripción de casos de uso del sistema

Tabla 15. Definición del caso de uso "Ejecutar evento"

Caso de uso	Ejecutar evento
Actor	Reloj: (Inicia)
Descripción	El caso de uso inicia cuando corresponda ejecutarse una tarea programada. En ese instante se ejecutan las operaciones definidas en esta, finalizando el caso de uso.
Referencias	RF 7

²¹ El comportamiento específico de una entidad que participa en un contexto particular.

Capítulo 2. Propuesta del sistema

Tabla 16. Definición del caso de uso “Gestionar usuario”

Caso de uso	Gestionar usuario
Actor	Reloj: (Inicia)
Descripción	El caso de uso inicia cuando se ejecuta la tarea programada para gestionar usuarios. Luego se obtiene de Estructura y Composición los datos de los usuarios que han sido añadidos, modificados o eliminados, actualizándose dichos datos en el Archivo Universitario, finalizando el caso de uso.
Referencias	RF 8, RF 9, RF 10

Tabla 17. Definición del caso de uso “Gestionar grupo de usuario”

Caso de uso	Gestionar grupo de usuario
Actor	Reloj: (Inicia)
Descripción	El caso de uso inicia cuando se ejecuta la tarea programada para gestionar grupos de usuarios. Luego se obtiene de Estructura y Composición los datos necesarios para conformar los grupos de usuarios, modificarlos o eliminarlos, según los cambios ocurridos. Se actualizan dichos datos en el Archivo Universitario y finaliza el caso de uso.
Referencias	RF 11, RF 12, RF 13

Tabla 18. Definición del caso de uso “Gestionar permiso”

Caso de uso	Gestionar permiso
Actor	Reloj: (Inicia)

Capítulo 2. Propuesta del sistema

Descripción	El caso de uso inicia cuando se ejecuta la tarea programada para gestionar permisos. Luego se obtiene de Estructura y Composición los cambios ocurridos en las áreas, en cuanto a permisos. Se actualizan los mismos en el Archivo Universitario y finaliza el caso de uso.
Referencias	RF 14, RF 15, RF 16

Tabla 19. Definición del caso de uso “Gestionar tarea programada”

Caso de uso	Gestionar tarea programada
Actor	Administrador del sistema: (Inicia)
Descripción	El caso de uso inicia cuando el administrador del sistema decide crear, modificar o eliminar una tarea programada, finalizando el caso de uso.
Referencias	RF 4, RF 5, RF 6

Tabla 20. Definición del caso de uso “Realizar auditoría”

Caso de uso	Realizar auditoría
Actor	Administrador del sistema: (Inicia)
Descripción	El caso de uso inicia cuando el administrador desea conocer los cambios ocurridos con los permisos asociados a una persona sobre un espacio, finalizando el caso de uso.
Referencias	RF 17

Capítulo 2. Propuesta del sistema

Tabla 21. Definición del caso de uso “Asociar estructura con espacio”

Caso de uso	Gestionar estructura con espacio
Actor	Administrador: (Inicia)
Descripción	El caso de uso inicia cuando el administrador del sistema desea crear, modificar, o eliminar la asociación entre un área de Estructura y Composición y un espacio del Archivo Universitario, finalizando el caso de uso.
Referencias	RF 1, RF 2, RF 3

Para consultar las descripciones textuales más detalladas de los casos de uso, ver Anexo A de la versión extendida de este documento.

2.8.4 Prototipo de interfaz de usuario

Los prototipos de interfaz de usuario permiten comprender y especificar las interacciones entre actores humanos y el sistema durante la captura de requisitos. No sólo ayudan a desarrollar una interfaz gráfica mejor, sino también a comprender mejor los casos de uso (JACOBSON, 2004).

Para ver los prototipos de interfaz de los restantes casos de uso ver Anexo A de la versión extendida de este documento.

Crear tarea programada



Modo de ejecución: *

Al iniciar la aplicación en 2 minutos

Nombre: *

Descripción:

Siguiente

Cancelar

Capítulo 2. Propuesta del sistema

Crear tarea programada

Configuración de la planificación: *

Iniciar tarea después de minuto (s)

Frecuencia: *

Responsabilidad: *

Categoría de personas: *

Rol: *

Figura 8. Prototipo "Crear tarea programada"

Análisis del capítulo

- ✓ El modelo de dominio permitió obtener una mejor comprensión y descripción de las clases dentro del contexto del sistema.
- ✓ Las técnicas de captura de requisitos permitieron identificar las principales necesidades y funcionalidades que deberían incorporarse a la solución en general.
- ✓ En modelo de casos de uso permitió que los desarrolladores y los clientes llegaran a un acuerdo sobre las condiciones y posibilidades que debe cumplir el sistema.

Capítulo 3. Análisis y diseño

En el presente capítulo se hace un análisis de los requisitos descritos en el capítulo anterior. Para ello se presenta el modelo de análisis, compuesto a su vez por los diagramas de clases de análisis y los de colaboración. Se presenta además, una vista de la arquitectura general del Archivo Universitario y se describen los principales componentes involucrados en la propuesta de solución. Finalmente se exhibe el diagrama de clases del diseño y el resultado de la validación tanto de los requisitos como del diseño.

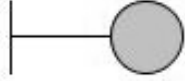


3.1 Modelo de análisis

“El modelo del análisis representa los requisitos en múltiples dimensiones, con lo que se incrementa la probabilidad de encontrar errores, de que surjan inconsistencias y de que se descubran omisiones” (PRESSMAN, 2002).

Clases del análisis: se centran en los requisitos funcionales y son evidentes en el dominio del problema porque representan sus conceptos y relaciones. Tienen atributos y entre ellas se pueden establecer relaciones de asociación, agregación, composición, generalización, especialización.

Siempre se centralizan en tres estereotipos básicos.

Tabla 22. Estereotipos en las clases del análisis

Nombre	Características	Representación
Interfaz	Modelan la interacción entre el sistema y sus actores.	 Clase de interfaz
Control	Representa coordinación secuencial, transacciones y control de otros objetos. Encapsula el control de un caso de uso en concreto. Se usa para representar derivaciones y cálculos complejos.	 Clase de control
Entidad	Modelan información que posee larga vida y que es a menudo persistente.	 Clase de entidad

3.1.1 Diagramas de clases del análisis

Un diagrama de clases del análisis es un artefacto en el que se representan los conceptos en un dominio del problema, a través de diferentes clases y sus relaciones.

A continuación se muestra el diagrama de clases del análisis para el caso de uso: Gestionar tarea programada.

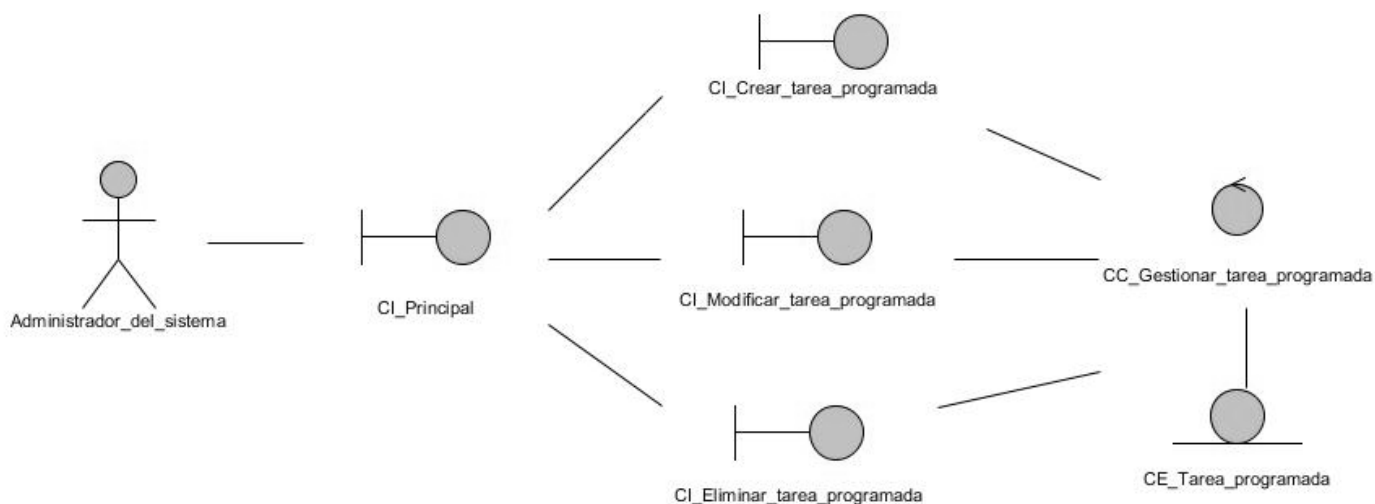


Figura 9. Diagrama de clases del análisis: caso de uso “Gestionar tarea programada”

Para ver los diagramas de clases del análisis de los restantes casos de uso, ver Anexo B de la versión extendida de este documento.

3.1.2 Diagramas de interacción

Un diagrama de interacción es un conjunto de objetos y sus relaciones, incluyendo los mensajes que pueden ser enviados entre ellos. Tratan la vista dinámica de un sistema; incluyen diagramas de colaboración y diagramas de secuencia (JACOBSON, 2004).

Los diagramas de secuencia destacan el orden temporal de los mensajes, mostrando los objetos que participan en la interacción mediante sus líneas de vida. En el caso de los diagramas de colaboración se resalta la organización estructural de los objetos que envían y reciben mensajes.

A continuación se muestra el diagrama de colaboración para el caso de uso: Crear tarea programada.

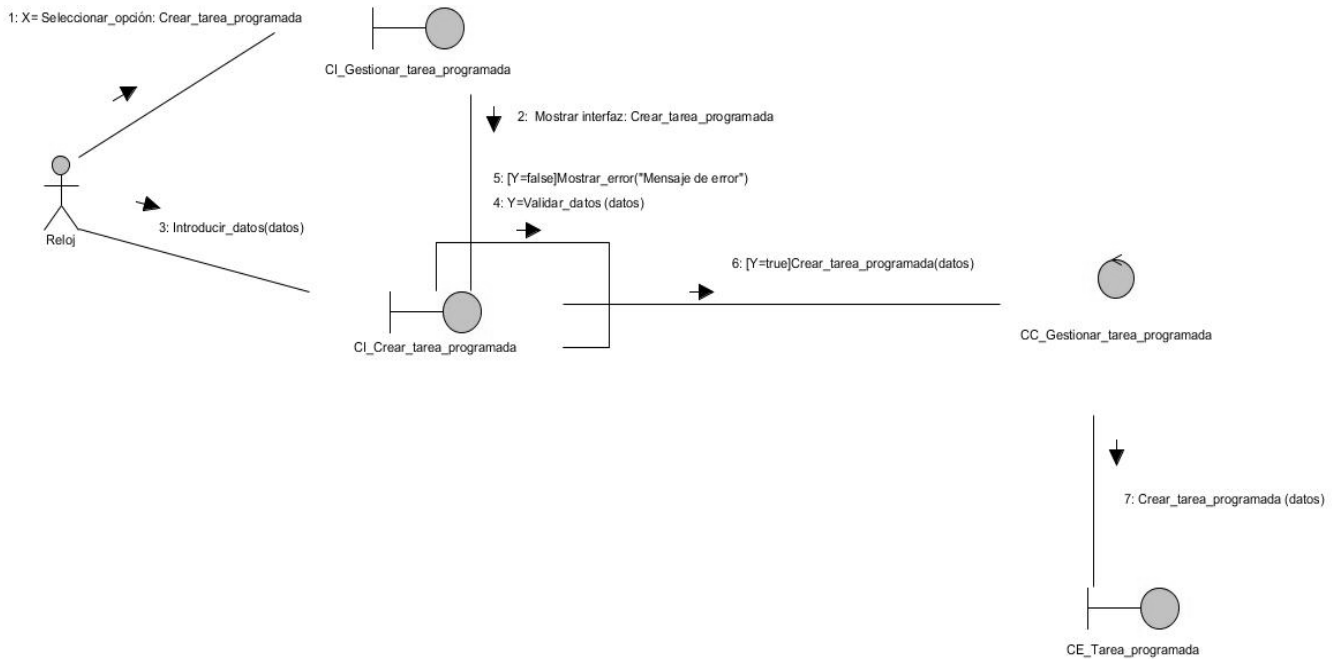


Figura 10. Diagrama de colaboración: caso de uso "Crear tarea programada"

Para ver los diagramas de colaboración de los restantes casos de uso, ver Anexo B de la versión extendida de este documento.

3.2 Arquitectura del sistema

“Una arquitectura es el sistema de decisiones significativas sobre la organización de un sistema de software, la selección de los elementos estructurales y de sus interfaces por los cuales el sistema es compuesto, junto con su comportamiento según lo especificado en las colaboraciones entre estos elementos, la composición de estos elementos estructurales y del comportamiento en subsistemas progresivamente más grandes y el estilo arquitectónico que dirigen esta organización, los elementos y sus interfaces, sus colaboraciones y su composición” (JACOBSON, 2004).

La figura 11 presenta una visión general de la plataforma de Alfresco en un enfoque por capas. Los componentes específicos de Alfresco se presentan en color blanco; los módulos, interfaces de usuario y las bibliotecas de otros fabricantes en color gris mientras que los componentes involucrados en la propuesta de solución se destacan en color negro.

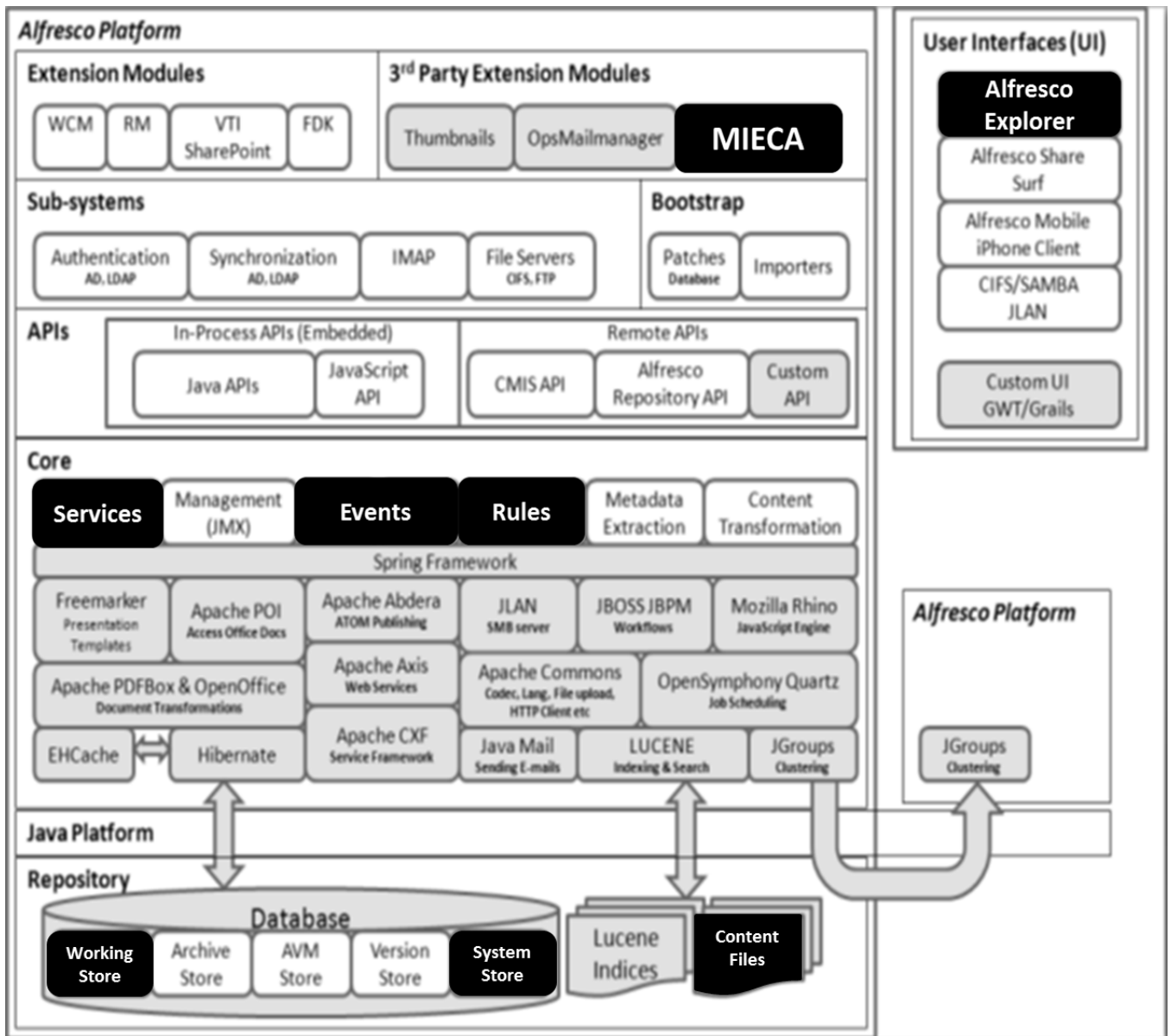


Figura 11. Arquitectura de la propuesta de solución

A continuación se brinda una breve descripción de los componentes involucrados en la propuesta de solución (BERGLJUNG, 2011):

3.2.1 Repositorio de contenidos (Repository)

La capa de datos comúnmente denominada repositorio de contenidos, hace referencia al lugar donde se almacena toda la información en Alfresco. Consiste en una jerarquía de nodos de diferentes tipos, cada nodo se encuentra asociado a un nodo padre a través de la relación padre-hijo, con la excepción

del nodo raíz que no tiene padre. Un nodo puede relacionarse con otros nodos a partir de la definición de asociaciones.

Básicamente, el repositorio de contenidos se compone de un conjunto de almacenes (stores) en los cuales residen los nodos. Cada almacén concentra la información asociada a una lógica de negocio o de aplicación en particular. Los almacenes a su vez pueden residir en una base de datos (así es como ocurre normalmente) o físicamente en el disco duro. Como se puede observar en la Figura 11, los almacenes involucrados en la solución propuesta son: *Working Store*, *System Store* y *Content Files*.

Working Store

En este almacén se acumula toda la información asociada a los documentos del Archivo Universitario, así como la estructura jerárquica definida por los espacios. También forman parte de este almacén los datos de los usuarios, los grupos, los permisos de estos sobre cada contenido y demás información relacionada con los procesos de negocio en cuestión: tareas programadas y eventos.

System Store

La función de *System Store* es almacenar la información relacionada con el sistema como tal. En el caso particular del módulo MIECA se almacenaría la información relacionada con el módulo.

Content Files

El almacén de contenidos es el único que se aloja fuera del contexto de la bases de datos. En este se almacena toda la información que se dispone en formato binario. Se divide o estructura en diferentes carpetas que funcionan de la misma manera que las raíces en el almacén Spaces Store. Para el caso de la propuesta de solución se incorporan dos estructuras nuevas: *mieca-logs* que es donde se almacenarán los logs y *mieca-xml* donde se almacenará la correspondencia entre las estructuras y los espacios en el Archivo Universitario.

3.2.2 Núcleo (Core)

Esta capa contiene un conjunto de módulos y bibliotecas que brindan las funcionalidades necesarias para implementar soluciones centradas en la gestión del contenido: soluciones de gestión documental, de gestión de contenido web, gestión de documentos archivos, entre otros. En lo que respecta al módulo de integración con estructura y composición que se propone en este trabajo se hace uso en esta capa de los subsistemas: *Services*, *Events* y *Rules*.

Services

Este subsistema brinda un conjunto de servicios de aplicación que facilitan el acceso y manipulación de los diferentes objetos del repositorio. También brinda un conjunto de servicios para la gestión de usuarios y permisos.

Events

El subsistema de gestión de eventos brinda un conjunto de API's para la ejecución de eventos. En el desarrollo del módulo propuesto se hace uso de este subsistema para la gestión de las tareas programadas.

Rules

Conjunto de API's que proveen las funcionalidades necesarias para la ejecución de reglas de negocio. Esta API es necesaria porque en ella se definen las interfaces necesarias para la ejecución de acciones, las cuales son disparadas por los eventos en el momento que sea necesario.

3.2.3 Interfaces de Usuario (UI)

Alfresco provee una serie de interfaces de usuario que se pueden utilizar para administrar los contenidos a través de Alfresco Explorer. Este último es el cliente de administración de documentos tradicional que proveen sus desarrolladores para el acceso al repositorio a través de la web. Para el desarrollo del módulo se hará uso de este cliente ya que el mismo constituye el cliente web del Archivo Universitario.

3.3 Modelo de diseño

El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso, centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Este modelo sirve de abstracción de la implementación del sistema (LARMAN, 2004).

3.3.1 Patrones de diseño

Los patrones de diseño ofrecen una solución a problemas específicos y comunes del diseño orientado a objetos en el proceso de desarrollo de *software*. Un ejemplo de patrones son los GRASP (General Responsibility Assignment *Software* Patterns) que describen los principios de asignar responsabilidades a objetos para lograr un diseño eficaz del *software* (LARMAN, 2004).

También existen los patrones GoF dentro del cual se encuentran los estructurales, de creación y comportamiento.

Bajo Acoplamiento

Uno de los principios para proteger al *software* frente al cambio es mantener bajo el acoplamiento entre clases. El acoplamiento de una clase es el conjunto de dependencias que tiene con otras clases, cuanto menor sea el acoplamiento entre clases, menor influencia tendrán los cambios (LARMAN, 2004). Para lograr un bajo acoplamiento en la solución que se propone, cada clase deberá relacionarse solo con quien lo necesita para realizar sus procedimientos (o métodos), donde se harán uso de interfaces las cuales ocultan sus detalles de implementación en aquellas clases que la implementen, logrando así que las implementaciones puedan ser cambiadas sin impactar a las clases clientes que hace uso de ellas.

Alta Cohesión

Es el encargado de asignar una responsabilidad de modo que la cohesión siga siendo alta. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme (LARMAN, 2004).

Experto

Se tiene en consideración a qué clase debe pertenecer un método, este principio sugiere que se asigne a la clase que más sepa del método (es decir al experto). Esto es una consecuencia del principio de alta cohesión, ya que si se asignan los métodos a las clases que tienen la información necesaria para ejecutarlos, se están creando clases altamente cohesionadas (LARMAN, 2004). Este patrón se evidencia en las clases responsables de centralizar la información obtenida de las personas, estructuras y responsabilidades que brindan los servicios a consumir del Sistema de Gestión Universitaria.

Inyección de dependencias (DI)

Es uno de los pilares de spring framework, el cual beneficia a las aplicaciones construidas con spring con un bajo acoplamiento. Haciendo uso de objetos que solo conoces sus dependencias a través de su interfaz, por lo que sus dependencias pueden ser cambiadas por alguna implementación que satisfaga a la interfaz sin que el objeto dependiente sepa del cambio (GENBETA, 2011). El uso de este patrón se evidencia en las clases que implementen una interfaz y sean utilizadas para realizar inversión de control haciendo uso del contenedor de inyección de dependencias de spring framework.

Fachada

Provee una interfaz unificada y sencilla que funciona de intermediaria entre un cliente y una interfaz o grupos de interfaces más complejas (LARMAN, 2004). Este patrón se evidencia en las interfaces que agrupan un conjunto de funcionalidades y son utilizadas como mediadoras en alguna clase cliente.

Inversión de control

Es un método de programación en el que el flujo de ejecución de un programa se invierte respecto a los métodos de programación tradicionales, en los que la interacción se expresa de forma imperativa haciendo llamadas a procedimientos. Este patrón es el concepto fundamental del spring framework ya que, implementa un contenedor de inyección de dependencias que se encarga de gestionar las instancias, así como sus creaciones y destrucciones de los objetos (JAVA, 2007).

3.3.2 Diagramas de clases del diseño

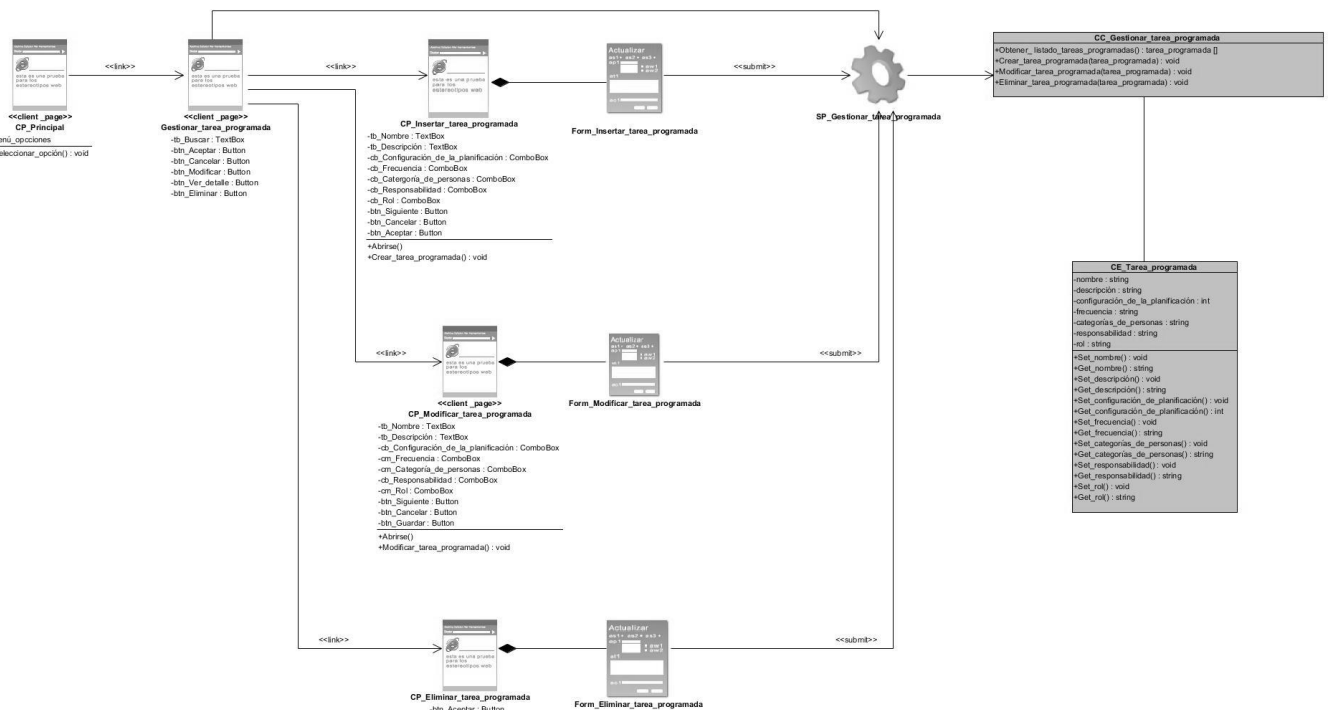


Figura 12. Diagrama de clases del diseño: caso de uso "Gestionar tarea programada"

Descripción de las clases

Tabla 23. Descripción de la clase "Gestionar_tarea_programada"

Nombre: Gestionar_tarea_programada

Tipo de clase: Controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Obtener_listado_tareas_programadas
Descripción:	Obtiene listado de tareas.
Nombre:	Crear _tarea _programada(tarea_programada)
Descripción:	Crea _una _tarea _programada
Nombre:	Modificar_tarea_programada(tarea_programada)
Descripción:	Modifica una tarea programada
Nombre:	Eliminar_tarea_programada(tarea_programada)
Descripción:	Elimina una tarea programada

Tabla 24. Descripción de la clase " Tarea_programada"

Nombre: Tarea_programada	
Tipo de clase: Entidad	
Atributo	Tipo
nombre	string
descripción	string

configuración_de_la_planificación	int
frecuencia	string
categorías_de_personas	string
responsabilidad	string
rol_de_permisos	string
Para cada responsabilidad:	
Nombre:	
Descripción:	

Para ver las descripciones de las restantes clases consultar el Anexo C de la versión extendida de este documento.

3.4 Validación de los requerimientos

Un elemento clave de cualquier proceso de ingeniería es la medición. Se emplean medidas para entender mejor los atributos de los modelos que se crean. Pero, fundamentalmente, se emplean las medidas para valorar la calidad de los productos de ingeniería o de los sistemas que son construidos (PRESSMAN, 2005). Durante la investigación se obtuvieron numerosos artefactos, producto de la metodología de *software* usada. Para validar estos artefactos, se aplican distintas métricas en pos de verificar la calidad y efectividad de los mismos.

3.4.1 Métrica para evaluar los requisitos

Para realizar la validación de los requisitos existe toda una lista de características que sugieren el uso de una o varias métricas como son: especificidad (ausencia de ambigüedad), corrección, compleción, comprensión, capacidad de verificación, consistencia externa e interna, capacidad de logro, concisión, trazabilidad, capacidad de modificación, exactitud y capacidad de reutilización (PRESSMAN, 2005).

Es importante tener en cuenta que para medir las características de la especificación, es necesario conseguir profundizar cuantitativamente en la especificidad y en la completitud.

Especificidad

Para llevar a cabo este proceso se tiene que: n_r representa el número de requisitos del sistema.

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

$$n_r = 17 + 12$$

$$n_r = 29$$

Donde n_f es el número de requisitos funcionales y n_{nf} es el número de requisitos no funcionales. Para determinar la especificidad (ausencia de ambigüedad) de los requisitos se sugiere una métrica basada en la consistencia de la interpretación de los revisores para cada requisito.

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

Donde n_{ui} es el número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas. El valor de Q a medida que se acerca a 1, se va disminuyendo la ambigüedad de la especificación.

Con el objetivo de obtener la menor ambigüedad posible, para que los requisitos tengan una mayor claridad de modo que satisfagan las necesidades de cliente, se llevaron a cabo 2 revisiones con 3 especialistas del proyecto²².

En la primera revisión se identificaron algunos requisitos funcionales y no funcionales que presentaban problemas en la redacción y de ambigüedad. Para un total de 29 requerimientos, los revisores tuvieron la misma interpretación para 22 de ellos.

$$Q = 22/29$$

$$Q = 0.75$$

En la segunda revisión, ya corregidos los errores identificados en la primera, los revisores tuvieron la misma interpretación para el total de requisitos.

$$Q = 29/29$$

$$Q = 1$$

²² Ing. Arioski Areces Gonzalez desarrollador, Ing. Michel David Suárez desarrollador, Alien Gongora Rodríguez desarrollador.



Figura 13. Gráfica de comparación entre la primera y la segunda revisión en la especificidad

Al concluir el estudio cuantitativo realizado a los resultados obtenidos en cada una de las revisiones, se evidencia que los requisitos presentan un grado bajo de ambigüedad.

Compleción

La aplicación de esta métrica siempre devuelve un resultado entre 0 y 1. Mientras más cercano a 1 se encuentre el resultado, indica un alto nivel de completitud en la definición de los requisitos. Este valor se calcula de la siguiente forma:

$$Q_1 = n_a / (n_a + n_b)$$

n_a : número de requisitos completos

n_b : número de requisitos pobremente especificados

Para la aplicación de esta métrica se procedió de igual forma que la anterior, arrojando los siguientes resultados:

En la primera revisión:

$$Q_1 = 20/29$$

$$Q_1 = 0.68$$

En la segunda revisión:

$$Q_1 = 27/29$$

$$Q_1 = 0.93$$



Figura 14. Gráfica de comparación entre la primera y la segunda revisión en la completación

Culminada las revisiones los datos arrojados muestran que la mayoría de los requisitos están completamente especificados.

Según los resultados obtenidos con la aplicación de las métricas de Especificidad y Completación para evaluar la calidad de la Especificación de Requisitos, se evidencia que los mismos están completamente especificados y son de claro entendimiento por el lector.

La siguiente gráfica muestra lo mencionado anteriormente.

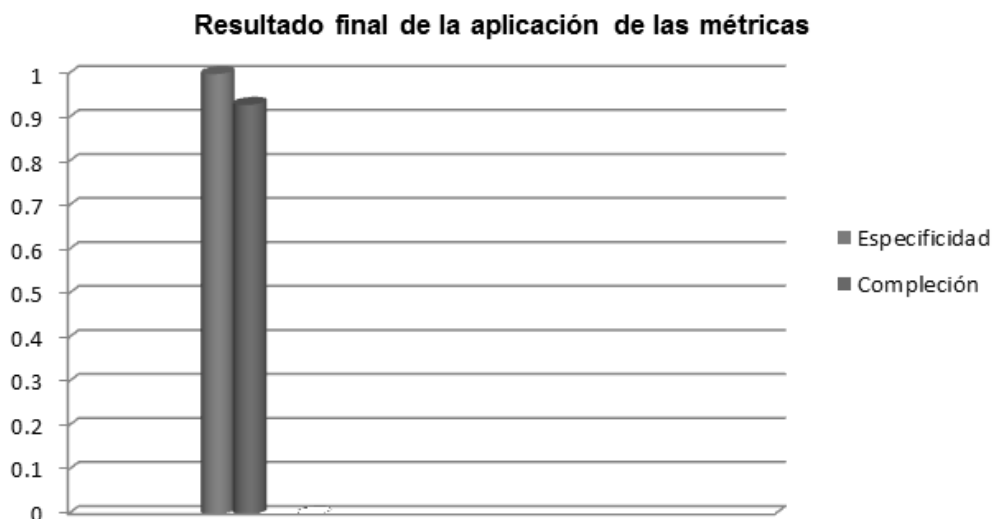


Figura 15. Gráfica con los datos finales de la aplicación de las 2 métricas

3.4.2 Técnica de construcción de prototipos

Un prototipo puede considerarse a la implementación concreta de un sistema que se crea para explotar cuestiones sobre aspectos muy diversos durante el desarrollo de un sistema. Estos prototipos permiten la comunicación y participación entre el equipo de desarrollo y el cliente, haciendo de las especificaciones una herramienta fundamental para comprobar los requisitos del *software*.

Un prototipo no funcional es el diseño de la posible interfaz del *software* a construir, esta es utilizada para tener un mejor entendimiento del problema y validar los requisitos que solicitó el usuario. La construcción de los prototipos es una alternativa para validar los requerimientos funcionales que fueron capturados durante la etapa de Requerimientos. Para la construcción de la interfaz gráfica se hizo uso de la herramienta Pencil, modelo que satisfizo la mayoría de las expectativas y necesidades de los clientes.

Las interfaces de usuario tienen como ventaja reflejar la presentación e interacción de las necesidades del usuario en un entorno fácil de entender.

3.5 Validación de diseño mediante criterio de especialistas del proyecto

Para garantizar la validación del diseño propuesto independientemente que estuvo bajo la supervisión de un desarrollador durante su realización, finalmente fue revisado por algunos especialistas con experiencia en el desarrollo de *software*. Entre ellos un analista y un desarrollador, los cuales dieron su aprobación al diseño propuesto teniendo en cuenta la experiencia en este tema.

3.5.1 Validación del modelo de diseño propuesto

Un elemento fundamental para evaluar la calidad de un producto de *software* es la aplicación de métricas al diseño. Al reflexionar en este sentido, Pressman (2005) resalta que *“las métricas del software proporcionan una manera cuantitativa de valorar la calidad de los atributos internos del producto, permitiendo por tanto al ingeniero valorar la calidad antes de construir el producto. Las métricas proporcionan la visión interna necesaria para crear modelos efectivos de análisis y diseño, un código sólido y pruebas minuciosas”* encuentran (PRESSMAN, 2005).

En su estudio sobre la calidad del diseño orientado a objeto, Pressman señala que *“las métricas de diseño a nivel de componentes se concentran en las características internas de los componentes del software, con medidas que pueden ayudar al desarrollador a juzgar la calidad de un diseño a nivel de componente”* (PRESSMAN, 2005). Teniendo en cuenta las apreciaciones de dicho autor se han creado métricas que engloban los principales atributos de calidad del *software*, entre estos se encuentran:

- ✓ Responsabilidad – es la responsabilidad asignada a una clase en un marco de modelado de un dominio o concepto, de la problemática propuesta

- ✓ Complejidad de implementación – grado de dificultad que tiene implementar un diseño de clases determinado
- ✓ Reutilización – significa cuán reutilizada es una clase o estructura de clase, dentro de un diseño de software.

El **tamaño operacional de clase (TOC)** es una de las métricas concebidas como instrumento para evaluar la calidad del diseño y su relación con los atributos de calidad definidos en este trabajo. La misma está dada por el número de métodos asignados a una clase (PRESSMAN, 2005).

Tabla 25. Atributos de calidad

Atributo de calidad	Modo en que lo afecta
Responsabilidad	Un aumento del TOC implica un aumento de la responsabilidad asignada a la clase.
Complejidad de implementación	Un aumento del TOC implica un aumento de la complejidad de implementación de la clase.
Reutilización	Un aumento del TOC implica una disminución del grado de reutilización de la clase.

3.5.2 Resultados del instrumento de evaluación de la métrica Tamaño Operacional de Clase (TOC)

La figura que se muestra a continuación representa la relación entre la cantidad de clases y la cantidad de procedimientos por intervalos definidos asignadas a cada clase.

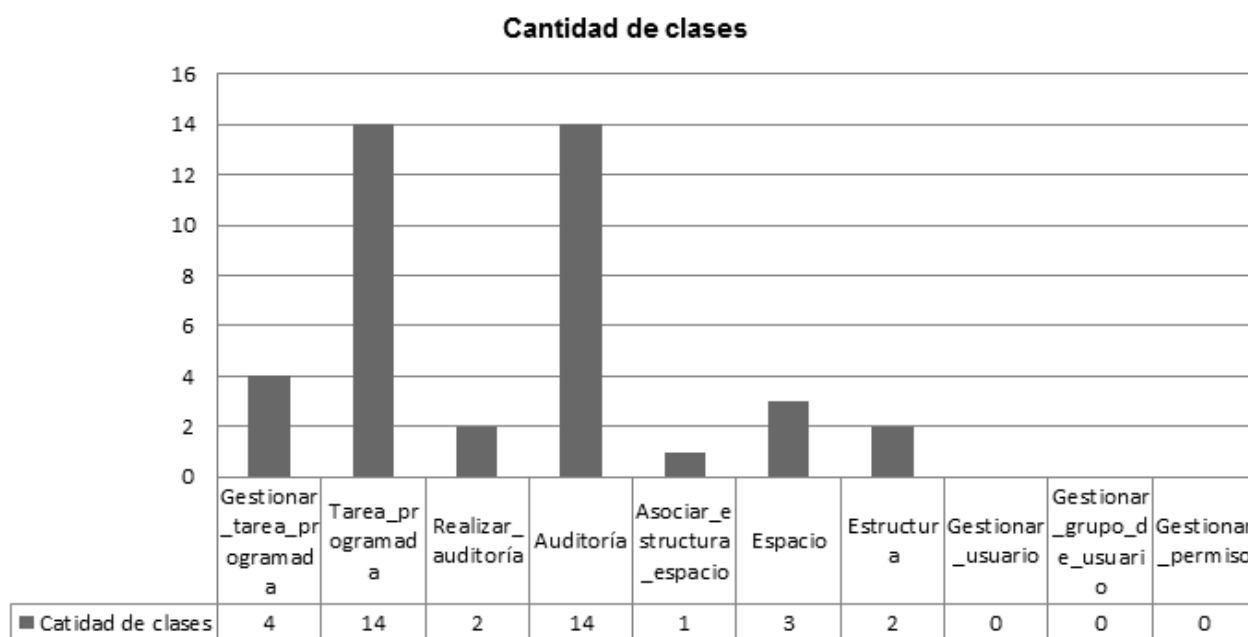


Figura 16. Representación de los resultados obtenidos en el instrumento para la cantidad de clases agrupados en los intervalos definidos

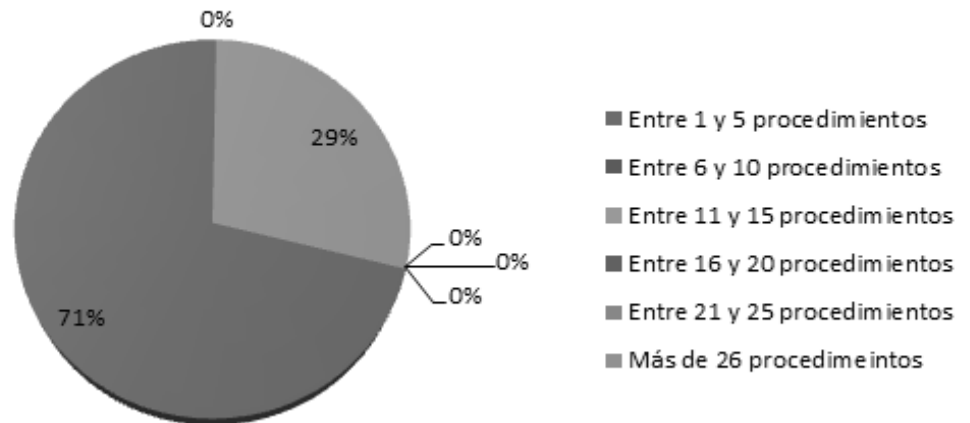


Figura 17. Representación en porcentaje de los resultados obtenidos en el instrumento, agrupados según los intervalos definidos para la cantidad de procedimientos



Figura 18. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Responsabilidad

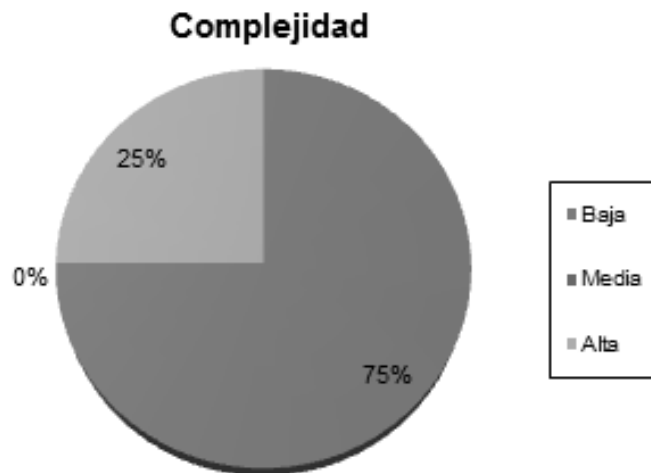


Figura 19. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Complejidad de Implementación

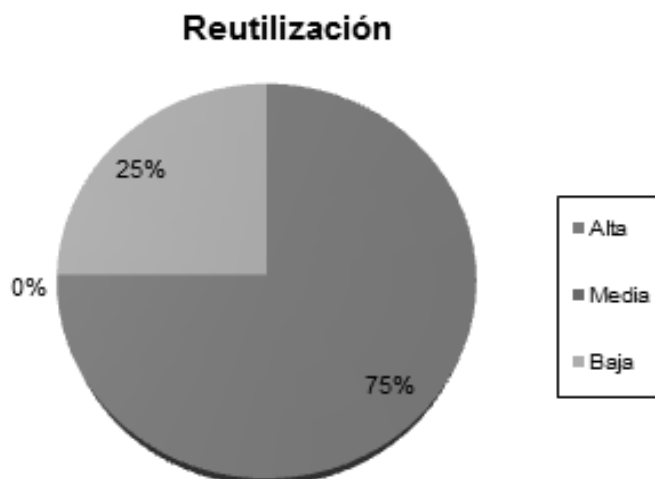


Figura 20. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Reutilización

Luego de analizados los resultados obtenidos en la evaluación del instrumento de medición de la métrica TOC, se puede plantear que el diseño propuesto para la integración del Archivo Universitario con Estructura y Composición tiene calidad. Fundamentando la anterior afirmación con los resultados obtenidos, al observarse que el 71% de las clases que componen el diseño cuentan con menos de 5 procedimientos. Esto influye de manera positiva en el hecho que predomine una baja responsabilidad y complejidad en las clases, así como una alta reutilización con un valor del 80%.

Capítulo 3. Análisis y diseño

Análisis del capítulo

- ✓ Con el desarrollo de este capítulo, los requisitos tanto funcionales como no funcionales se han ido transformando a una especificación que describe cómo implementar la solución de integrar el Archivo Universitario con el módulo Estructura y Composición.
- ✓ La aplicación de las métricas para evaluar la calidad de la Especificación de requisitos demostró que estas especificaciones se corresponden con las necesidades del cliente y que tienen correcta interpretación por parte del equipo de desarrollo de *software*.
- ✓ La aplicación de métricas para evaluar el modelo de diseño propuesto demostró una aceptable calidad de diseño. Asimismo, contribuyó a la disminución de dificultades durante la implementación del sistema, garantizando la reutilización y agilidad en el proceso de desarrollo de *software*.

Conclusiones generales

- ✓ El planteamiento de los principales conceptos relacionados con la gestión documental y los archivos universitarios contribuyó a un acercamiento y entendimiento de la importancia que tiene el control de acceso a la documentación producida y generada como resultado de los procesos universitarios.
- ✓ El uso de técnicas para realizar la captura de requisitos permitió identificar las funcionalidades y cualidades que debe tener el sistema, propiciando un mejor entendimiento del diseño de la propuesta de solución.
- ✓ El diseño de los procesos de actualización de gestión de usuarios, grupos de usuarios y permisos contribuirá a la seguridad de las evidencias documentales en el Archivo Universitario.
- ✓ Las métricas aplicadas a la propuesta de solución propiciaron una correcta validación de las funcionalidades, el análisis y diseño de los procesos de actualización de gestión de usuarios, grupos de usuarios y permisos en el Archivo Universitario.

Recomendaciones

- ✓ Hacer uso del mecanismo de extensión que provee Alfresco para incorporar nuevas soluciones sobre el repositorio.
- ✓ Implementar e integrar al Archivo Universitario las funcionalidades especificadas teniendo en cuenta todas las características de la propuesta de solución.
- ✓ Extender el módulo de salva y restaura de Alfresco, de modo que las configuraciones del módulo propuesto también se almacenen periódicamente.

Referencias bibliográficas

- ✓ ABDALA BARCENAS, Adalgisa. Objetivos de un proyecto de administración de y gestión de documentos y archivos electrónicos. Bogotá: Archivo General de la Nación; Comité de Gestión de Documentos, 2000. 15 p. ISSN: 15624730
- ✓ AIIM. 2009.< <http://www.aiim.org/FAQs>>
- ✓ ALFRESCO. 15 Septiembre 2004 <<http://www.alfresco.com/>>
- ✓ Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process [en línea]. Theresa Hudson. New York, 2008- [fecha de consulta: 9 de Noviembre 2011]. Disponible en: http://www.amazon.com/gp/product/0471202827/ref=pd_lpo_k2_dp_sr_2?pf_rd_p=486539851&pf_rd_s=lpo-top-stripe-1&pf_rd_t=201&pf_rd_i=0201710919&pf_rd_m=ATVPDKIKX0DER&pf_rd_r=1YG1C8PD53Q5PDY3H2PV#reader_0471202827. ISBN: 0471202827
- ✓ BERGLJUNG, Martin. Alfresco 3 Business Solutions [en línea]. Birmingham. Steven Wilding. 2011. [fecha de consulta: 3 Marco 2012]. Disponible en: <http://books.google.com/cu/books?id=WqZ7zdBxsc&printsec=frontcover&dq=Alfresco+3+Business+Solutions&hl=es&sa=X&ei=NxPcT9mNLM-I6AGOpYWXcw&ved=0CDMQ6AEwAA#v=onepage&q=Alfresco%20%20Business%20Solutions&f=false>. ISBN: 9781849513340
- ✓ BIBLIOTECA DEL INSTITUTO ANTONIO DE NEBRIJA DE ESTUDIOS SOBRE LA UNIVERSIDAD. ARCHIVOS UNIVERSITARIOS E HISTORIA DE LAS UNIVERSIDADES [en línea]. Madrid. JOSÉ RAMÓN CRUZ MUNDET, 2003 [fecha de consulta: 8 Mayo 2011]. Disponible en: http://e-archivo.uc3m.es/bitstream/10016/7882/1/09.archivos_universitarios.pdf. ISBN 8481556416
- ✓ CARUANA, David. Professional Alfresco [en línea]. Indiana. Dr. Paul Holmes-Higgin. 2010 [fecha de consulta: 12 de abril de 2012]. Disponible en: <http://www.wrox.com/WileyCDA/WroxTitle/Professional-Alfresco-Practical-Solutions-for-Enterprise-Content-Management.productCd-0470571047.html>. ISBN: 9780470571040
- ✓ Comité Técnico de Normalización. Norma iso 15489. 2005: Información y documentación. Gestión de documentos. Madrid, 1998. 91 p.

Referencias bibliográficas

- ✓ DURANTI, Luciana Definición de documentos archivísticos electrónicos en el sector público y su fiabilidad y autenticidad. Hanoi, 2004
- ✓ EcuRed. 13 Noviembre 2011 <
http://www.ecured.cu/index.php/M%C3%A9trica_de_dise%C3%B1o >
- ✓ Estándares IEEE. Guide for Developing System Requirements Specification, 1998. [fecha de consulta: 5 Febrero 2012]. ISBN: 155937716X
- ✓ FUNES, Ana Ballester. Prototipo de framework documental para firma electrónica: servidor. Tesis (Ingeniería Informática). Colombia. Universidad Externado de Colombia. Arquitectura de computadores, 2007. 41p.
- ✓ GENBETA: dev. 20 Abril 2011 <http://www.genbetadev.com/paradigmas-de-programacion/que-es-la-inyeccion-de-dependencias>
- ✓ JACOBSON, Ivar y BOOCH, Grady y RUMBAUGH, James. El proceso unificado de desarrollo de *software* [en línea]. Madrid, España. 2004. [fecha de consulta: 26 Noviembre 2011]. Disponible en: <http://es.scribd.com/doc/30251931/El-Proceso-Unificado-de-Desarrollo-de-Soft-Jacobson>. ISBN: 8478290362
- ✓ Java.Lang. NullPointerException. Noticias y Tutoriales de Java. 21 Mayo 2007 <
<http://javalangnullpointer.wordpress.com/2007/05/21/tecnicas-de-programacion-inversion-de-control/>>
- ✓ KAMPFMEYER, Ulrich. Procesos empresariales [en línea]. Alemania, Hamburgo. ECM Enterprise Content Management, 2006 [fecha de consulta: 18 Octubre 2011]. Disponible en: <http://es.scribd.com/doc/9709184/ECM-Enterprise-Content-Management-DMSEXPO-2006-Ulrich-Kampffmeyer>. ISBN: 3936534098
- ✓ LARMAN, Craig. UML Y PATRONES: Introducción al análisis y diseño orientado a objetos [en línea]. Prentice Hall, Hispanoamericana, S.A., 1999. 2004. [fecha de consulta: 3 Diciembre 2011]. Disponible en: <http://libropdf1.blogspot.com/2012/02/uml-y-patrones-introduccion-al-analisis.html>. ISBN: 9701702611
- ✓ Ley orgánica 11/1983, de 25 de agosto, de Reforma Universitaria (BOE de 1 Septiembre de 1983)
- ✓ MSc. MUGICA, Mayra Mena. Utilidad de las soluciones archivísticas para la gestión de la información en los sistemas electrónicos del sector de la salud. Revista ACIMED (Revista cubana de los profesionales de la información y de la comunicación en Salud), 15(3):1–15, 2007. ISSN 14094746.
- ✓ RODRIGUEZ, Mariela Álvarez. IDENTIFICACIÓN Y VALORACIÓN DE FONDOS

- DOCUMENTALES EN ARCHIVOS DE UNIVERSIDADES. Tesis (Maestría en Gestión de Documentos y Administración de Archivos). Universidad Internacional de Andalucía, 2005. 14 p
- ✓ POTTS, Jeff. Alfresco Developer Guide [en línea]. 2008. [fecha de consulta: 18 Febrero 2012]. Disponible en: <http://www.packtpub.com/alfresco-developer-guide/book>. ISBN: 1847193110
 - ✓ PRESSMAN, Roger S. Ingeniería de Sistemas. En: PRESSMAN, Roger S, Ingeniería de *Software*. Un enfoque práctico. España, 5ta edición. Félix Varela, 2002. 2005. 172 pp. ISBN: 8448132149
 - ✓ Propuesta de cuadro de clasificación funcional para el sistema de archivos de la Universidad de las Ciencias Informáticas de Cuba [en línea]: Acimed. Editorial Ciencias Médicas, 2009 - [fecha de consulta: 20 Enero 2012]. Disponible en: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352009000500004. ISSN: 15612880
 - ✓ Proyectalis. Gestión de proyectos. 16 Noviembre 2010 www.proyectalis.com/servicios/formacion/scrum/
 - ✓ PUBLIC RECORD OFFICE: Functional requirements for electronic records management systems. Londres. Public Record Office, 1999, 58 p. Disponible en : <http://www.pro.gov.uk/recordsmanagement/erecords/1999reqs/1999requirements.pdf>
 - ✓ Scribd. 14 de abril 2010 <<http://es.scribd.com/doc/3062020/Capitulo-I-HERRAMIENTAS-CASE>>
 - ✓ SHARIFF, Munwar. Enterprise Content Management Implementation [en línea]. Birmingham – Mumbai. Mike W. Walker, 2006. [fecha de consulta: 16 Febrero 2012]. Disponible en: <http://www.amazon.com/Alfresco-Enterprise-Content-Management-implementation/dp/1904811116>. ISBN: 1904811116
 - ✓ Sketching and Prototyping with Firefox Pencil Project. 15 Abril 2010 <<http://pencil.evolus.vn/en-US/UserGuides.aspx>>
 - ✓ SOMMERVILLE, Ian. Requerimientos. En: Ingeniería del *software* Séptima Edición. España, Madrid. Pearson Educación, SA. , 2005. 119 pp. ISBN: 9780321313799
 - ✓ Sketching and Prototyping with Firefox Pencil Project. 15 Abril 2010 <<http://pencil.evolus.vn/en-US/UserGuides.aspx>>

Referencias bibliográficas

- ✓ STABLE, Yudayly R, GIRALDO, Jorge E. P., MATEUS, Sandra P. S. Cambio de paradigma en la organización de la información en la sociedad del conocimiento. Revista Digital Sociedad de la Información, (15): 15 Febrero 2009. ISSN:1578326x
- ✓ Universidad de Sevilla. 2003 <<http://servicio.us.es/archivos/>>
- ✓ Visual Paradigm. 10 Octubre 2004 < <http://www.visual-paradigm.com/product/vpuml/>>
- ✓ World Wide Web Consortium [en línea]. España, 2002- [fecha de consulta: 20 de Octubre 2011]. Disponible en: <http://w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>
- ✓ ZAPATA, Carlos Mario y PALACIO, Carolina y Natalí. UNC-ANALISTA: HACIA LA CAPTURA DE UN CORPUS DE REQUISITOS A PARTIR DE LA APLICACIÓN DEL EXPERIMENTO MAGO DE OZ. Olaya Revista EIA [en línea]. n. 7. Junio 2007. [fecha de consulta: 18 Abril de 2012]. Disponible en: revista.eia.edu.co/articulos7/2-Articulo%20UNC.pdf.

Bibliografía

- ✓ ABDALA BARCENAS, Adalgisa. Objetivos de un proyecto de administración de y gestión de documentos y archivos electrónicos. Bogotá: Archivo General de la Nación; Comité de Gestión de Documentos, 2000. 15 p. ISSN: 15624730
- ✓ Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process [en línea]. Theresa Hudson. New York, 2008- [fecha de consulta: 9 de Noviembre 2011]. Disponible en: http://www.amazon.com/gp/product/0471202827/ref=pd_lpo_k2_dp_sr_2?pf_rd_p=486539851&pf_rd_s=lpo-top-stripe-1&pf_rd_t=201&pf_rd_i=0201710919&pf_rd_m=ATVPDKIKX0DER&pf_rd_r=1YG1C8PD53Q5PDY3H2PV#reader_0471202827. ISBN: 0471202827
- ✓ AIIM. 2009.< <http://www.aiim.org/FAQs>>
- ✓ ALFRESCO. 15 Septiembre 2004 <http://www.alfresco.com/>
- ✓ Alfresco 3 Business Solutions Practical implementation techniques and guidance for delivering business solutions with Alfresco [en línea]. Birmingham. Aditya Belpathak, 2011 [fecha de consulta: 5 Abril 2012]. Disponible en: <http://www.packtpub.com>. ISBN: 9781849513340
- ✓ Alfresco 3 Content Management. Enterprise Web Content Management made easy and affordable [en línea]. Birmingham. Akshara Aware, 2010 [fecha de consulta: 5 Abril 2012]. Disponible en: <http://www.packtpub.com>. ISBN: 9781847198006
- ✓ Alfresco 3 Records Management. Comply with regulations and secure your organization's records with Alfresco Records Management [en línea]. Birmingham. Aanchal Kumar, 2011 [fecha de consulta: 27 Marzo 2012]. Disponible en: <http://www.packtpub.com>. ISBN: 9781849514361
- ✓ Alfresco 3 Web Services. Build Alfresco applications using Web Services, Web Scripts and CMIS [en línea]. Birmingham. Aanchal Kumar, 2010 [fecha de consulta: 26 Marzo 2012]. Disponible en: <http://www.packtpub.com>. ISBN: 9781849511520
- ✓ Bergljung, Martin. Alfresco 3 Business Solutions [en línea]. Birmingham. Steven Wilding. 2011. [fecha de consulta: 3 Marco 2012]. Disponible en: <http://books.google.com/cu/books?id=WqZ7zddCbxsC&printsec=frontcover&dq=Alfresco+3+Business+Solutions&hl=es&sa=X&ei=NxPcT9mNLM->

- I6AGOpYWXCw&ved=0CDMQ6AEwAA#v=onepage&q=Alfresco%20%20Business%20Solutions&f=false. ISBN: 9781849513340
- ✓ BIBLIOTECA DEL INSTITUTO ANTONIO DE NEBRIJA DE ESTUDIOS SOBRE LA UNIVERSIDAD. ARCHIVOS UNIVERSITARIOS E HISTORIA DE LAS UNIVERSIDADES [en línea]. Madrid. JOSÉ RAMÓN CRUZ MUNDET, 2003 [fecha de consulta: 8 Mayo 2011]. Disponible en: http://e-archivo.uc3m.es/bitstream/10016/7882/1/09.archivos_universitarios.pdf. ISBN 8481556416
 - ✓ CAMPILLO, Irima Torres. Sistema de Gestión Integral de Documentos de archivo para empresas de la construcción del territorio de Camagüey. Tesis (Doctoral). Granada-Habana. Universidad de Granada y Universidad de Cuba, 2010.
 - ✓ CARUANA, David. Professional Alfresco [en línea]. Indiana. Dr. Paul Holmes-Higgin. 2010 [fecha de consulta: 12 de abril de 2012]. Disponible en: <http://www.wrox.com/WileyCDA/WroxTitle/Professional-Alfresco-Practical-Solutions-for-Enterprise-Content-Management.productCd-0470571047.html>. ISBN: 9780470571040
 - ✓ Comité Técnico de Normalización. Norma iso 15489. 2005: Información y documentación. Gestión de documentos. Madrid, 1998. 91 p.
 - ✓ DURANTI, Luciana Definición de documentos archivísticos electrónicos en el sector público y su fiabilidad y autenticidad. Hanoi, 2004
 - ✓ EcuRed. 13 Noviembre 2011 <
http://www.ecured.cu/index.php/M%C3%A9trica_de_dise%C3%B1o >
 - ✓ Estándares IEEE. Guide for Developing System Requirements Specification, 1998. [fecha de consulta: 5 Febrero 2012]. ISBN: ISBN 155937716X
 - ✓ FUNES, Ana Ballester. Prototipo de framework documental para firma electrónica: servidor. Tesis (Ingeniería Informática). Colombia. Universidad Externado de Colombia. Arquitectura de computadores, 2007. 41p.
 - ✓ GARNACHO, Arturo Ribagorda. Ediciones CODA, S.L. 14 Abril 2010. Disponible en: <http://www.idg.es/computerworld/Glosario-de-Terminos-de-Seguridad-de-las-TI/seccion-/articulo-8052>
 - ✓ GENBETA: dev. 20 Abril 2011 <http://www.genbetadev.com/paradigmas-de-programacion/que-es-la-inyeccion-de-dependencias>
 - ✓ GONZALEZ, Yailin Molina y PENA, Evelio Rafael Martínez. Diseño e implementación de un módulo de administración para el Sistema Unificado de Gestión de la Fuerza de Trabajo

- Calificada. Tesis (Ingeniería en Ciencias Informáticas). Universidad de las Ciencias Informáticas (UCI), 2010. 27 pp.
- ✓ JACOBSON, Ivar y BOOCH, Grady y RUMBAUGH, James. El proceso unificado de desarrollo de *software* [en línea]. Madrid, España. 2004. [fecha de consulta: 26 Noviembre 2011]. Disponible en: <http://es.scribd.com/doc/30251931/EI-Proceso-Unificado-de-Desarrollo-de-Soft-Jacobson>. ISBN: 8478290362
 - ✓ Java.Lang. NullPointerException. Noticias y Tutoriales de Java. 21 Mayo 2007 < <http://javalangnullpointer.wordpress.com/2007/05/21/tecnicas-de-programacion-inversion-de-control/>>
 - ✓ KAMPPFMEYER, Ulrich. Procesos empresariales [en línea]. Alemania, Hamburgo. ECM Enterprise Content Management, 2006 [fecha de consulta: 18 Octubre 2011]. Disponible en: <http://es.scribd.com/doc/9709184/ECM-Enterprise-Content-Management-DMSEXPO-2006-Ulrich-Kampffmeyer>. ISBN: 3936534098
 - ✓ LARMAN, Craig. UML Y PATRONES: Introducción al análisis y diseño orientado a objetos [en línea]. Prentice Hall, Hispanoamericana, S.A., 1999. 2004. [fecha de consulta: 3 Diciembre 2011]. Disponible en: <http://libropdf1.blogspot.com/2012/02/uml-y-patrones-introduccion-al-analisis.html>. ISBN: 9701702611
 - ✓ Ley orgánica 11/1983, de 25 de agosto, de Reforma Universitaria (BOE de 1 Septiembre de 1983)
 - ✓ MSc. MUGICA, Mayra Mena. Utilidad de las soluciones archivísticas para la gestión de la información en los sistemas electrónicos del sector de la salud. Revista ACIMED (Revista cubana de los profesionales de la información y de la comunicación en Salud), 15(3):1–15, 2007. ISSN 14094746.
 - ✓ RODRIGUEZ, Mariela Álvarez. IDENTIFICACIÓN Y VALORACIÓN DE FONDOS DOCUMENTALES EN ARCHIVOS DE UNIVERSIDADES. Tesis (Maestría en Gestión de Documentos y Administración de Archivos). Universidad Internacional de Andalucía, 2005. 14 p
 - ✓ POTTS, Jeff. Alfresco Developer Guide [en línea]. 2008. [fecha de consulta: 18 Febrero 2012]. Disponible en: <http://www.packtpub.com/alfresco-developer-guide/book>. ISBN: 1847193110
 - ✓ PRESSMAN, Roger S. Proceso de *Software* y Métricas de Proyectos. En: PRESSMAN, Roger S. Ingeniería de Software. Un enfoque práctico. España, 5ta edición. Félix Varela, 2002 53 pp. ISBN: 8448132149

- ✓ PRESSMAN, Roger S. Ingeniería de Sistemas. En: PRESSMAN, Roger S, Ingeniería de *Software*. Un enfoque práctico. España, 5ta edición. Félix Varela, 2002. 2005. 172 pp. ISBN: 8448132149
- ✓ Propuesta de cuadro de clasificación funcional para el sistema de archivos de la Universidad de las Ciencias Informáticas de Cuba [en línea]: Acimed. Editorial Ciencias Médicas, 2009 - [fecha de consulta: 20 Enero 2012]. Disponible en: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352009000500004. ISSN: 15612880
- ✓ Proyectalis. Gestión de proyectos. 16 Noviembre 2010 www.proyectalis.com/servicios/formacion/scrum/
- ✓ PUBLIC RECORD OFFICE: Functional requirements for electronic records management systems. Londres. Public Record Office, 1999, 58 p. Disponible en : <http://www.pro.gov.uk/recordsmanagement/erecords/1999reqs/1999requirements.pdf>
- ✓ Scribd. 14 de abril 2010 <<http://es.scribd.com/doc/3062020/Capitulo-I-HERRAMIENTAS-CASE>>
- ✓ SHARIFF, Munwar. Enterprise Content Management Implementation [en línea]. Birmingham – Mumbai. Mike W. Walker, 2006. [fecha de consulta: 16 Febrero 2012]. Disponible en: <http://www.amazon.com/Alfresco-Enterprise-Content-Management-implementation/dp/1904811116>. ISBN: 1904811116
- ✓ Sketching and Prototyping with Firefox Pencil Project. 15 Abril 2010 < <http://pencil.evolus.vn/en-US/UserGuides.aspx>>
- ✓ SOMMERVILLE, Ian. Requerimientos. En: Ingeniería del *software* Séptima Edición. España, Madrid. Pearson Educación, SA. , 2005. 119 pp. ISBN: 978-0-321-31379-9
- ✓ STABLE, Yudayly R, GIRALDO, Jorge E. P., MATEUS, Sandra P. S. Cambio de paradigma en la organización de la información en la sociedad del conocimiento. Revista Digital Sociedad de la Información, (15): 15 Febrero 2009. ISSN:1578326x
- ✓ Universidad de Sevilla. 2003 <<http://servicio.us.es/archivous/>>
- ✓ VALDERRAMA, Luis David Fernández. GESTIÓN DOCUMENTAL. Caracas-Venezuela. IESA Instituto de Estudios Superiores en Administración, 2009.
- ✓ Visual Paradigm. 10 Octubre 2004 < <http://www.visual-paradigm.com/product/vpuml/>>

Anexo A

Descripción de los casos de uso

Tabla 26. Descripción del caso de uso "Ejecutar evento"

Caso de uso:	Ejecutar evento
Actor:	Reloj: (Inicia)
Resumen:	El sistema debe ejecutar automáticamente las tareas programadas por el administrador.
Precondiciones:	Ambos sistemas tienen que estar funcionando.
Referencias:	RF 7
Prioridad:	Alta
Flujo Normal de Eventos	
Acción del actor	Acción del sistema
1. El caso de uso inicia cuando se ejecuta una tarea programada.	2. El evento se ejecuta y realiza una copia de Estructura y Composición. La copia que se realiza es en dependencia de lo que mande a ejecutar la tarea programada para ese día puede ser: Una información en específico como por ejemplo: Solo actualizar las personas que integran un departamento. Realizar la copia por fecha por ejemplo: Copiar todos los cambios ocurridos el 25/05/2012. Se consumen los servicios del ECM Alfresco y se gestiona esta información, finalizando el caso de uso.

Tabla 27. Descripción del caso de uso "Gestionar usuario"

Caso de uso:	Gestionar usuario
---------------------	-------------------

Actor:	Reloj: (Inicia)
Resumen:	El caso de uso inicia cuando se ejecuta la tarea programada para gestionar usuarios, luego se obtiene del módulo Estructura y Composición los usuarios que han sido añadidos, modificados o eliminados, obteniendo con los mismos sus atributos: nombre, apellidos, organización, usuario, correo electrónico y el id de la organización, luego se consume el servicio que proporciona el ECM Alfresco para gestionar usuarios y a partir de ahí se actualiza en el Archivo Universitario esta información, finalizando el caso de uso.
Precondiciones:	
Referencias:	RF 8, RF 9, RF 10
Prioridad:	Alta
Flujo Normal de Eventos	
Acción del actor	Acción del sistema
	1. Selecciona acción a desarrollar: <ul style="list-style-type: none"> a) Si desea Crear usuario ver Sección “Crear usuario”. b) Si desea Modificar usuario ver Sección “Modificar usuario”. c) Si desea Eliminar usuario ver Sección “Eliminar usuario”.

Escenario “Crear usuario”	
Acción del actor	Respuesta del sistema
	2. Se obtiene del módulo Estructura y Composición los usuarios que han sido creados, obteniendo con los mismos los atributos: nombre, apellidos, carnet de identidad, solapín, id de la estructura, id de la responsabilidad, correo electrónico, luego se consume el servicio que proporciona el ECM Alfresco para gestionar usuarios y a partir de ahí se actualiza en el archivo Universitario esta información, finalizando el caso de uso.
Prototipo de Interfaz	
Escenario “Modificar usuario”	
Acción del Actor	Respuesta del Sistema
	2. Se obtiene del módulo Estructura y Composición los usuarios que han sido añadidos o modificados, obteniendo con los mismos los atributos: nombre, apellidos, organización, usuario, correo electrónico y el id de la organización, luego se consume el servicio que proporciona el ECM Alfresco para gestionar usuarios y a partir de ahí se actualiza en el archivo Universitario esta información, finalizando el caso de uso.

Prototipo de Interfaz	
Escenario "Eliminar usuario"	
Acción del Actor	Respuesta del Sistema
	2. Luego de verificar a través de la información que brinda el módulo EC que el usuario dejó de existir, se consume el servicio que posibilita el ECM Alfresco para gestionar usuarios, se elimina el mismo, finalizando el caso de uso.
Prototipo de Interfaz	
Poscondiciones	

Tabla 28. Descripción del caso de uso "Gestionar grupo de usuario"

Caso de uso:	Gestionar grupo de usuario
Actores:	Reloj: (Inicia)
Resumen:	El caso de uso inicia cuando se ejecuta la tarea programada para gestionar grupos de usuario, luego se obtiene de Estructura y Composición la estructura o la responsabilidad, en dependencia de cual se obtenga, se crea el grupo de usuarios con su nombre, luego se le pide a Estructura y Composición los usuarios que pertenecen a ella y se adicionan al grupo a través de los servicios

	que proporciona el ECM Alfresco para ello y finaliza el caso de uso.
Referencias:	RF 11, RF 12, RF 13
Prioridad:	Alta
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	<p>1. Selecciona acción a desarrollar:</p> <p>d) Si desea Crear grupo de usuario ver Sección “Crear grupo de usuario”.</p> <p>e) Si desea Modificar grupo de usuario ver Sección “Modificar grupo de usuario”.</p> <p>f) Si desea Eliminar grupo de usuario ver Sección “Eliminar grupo de usuario”.</p>
Sección “Crear grupo de usuario”	
Acción del Actor	Respuesta del Sistema
	<p>2. Se coge del módulo Estructura y Composición la estructura o la responsabilidad, si se escoge la primera se crea el grupo con el nombre de la misma, luego se obtienen de Estructura y Composición los usuarios que pertenecen a esa estructura y se añaden estos usuarios al grupo correspondiente a la estructura a través de los servicios con lo que cuenta Alfresco, de escoger la responsabilidad se toma el nombre de la responsabilidad, luego se pregunta qué personas tienen esa responsabilidad y se van añadiendo, creando de esta manera los grupos de usuarios, consumiendo los permisos mencionados anteriormente y finaliza el caso de uso.</p>

Prototipo de interfaz	
Sección “Modificar grupo de usuario”	
Acción del Actor	Respuesta del Sistema
	2. Se obtiene del módulo Estructura y Composición los datos actualizados de la estructura o de la responsabilidad, es decir si un usuario dejó de pertenecer a una estructura o cambió de responsabilidad, se modifica el grupo consumiendo el servicio que proporciona el ECM Alfresco para ello y finaliza el caso de uso.
Prototipo de Interfaz	
Sección “Eliminar grupo de usuario”	
Acción del Actor	Respuesta del Sistema
	2. Se obtiene del módulo Estructura y Composición los datos actualizados de la estructura o de la responsabilidad, es decir si se elimina una de ellas, se elimina a su vez el grupo de usuarios, consumiendo el servicio que proporciona el ECM Alfresco para ello y finaliza el caso de uso.

Prototipo de Interfaz	
Poscondiciones	

Tabla 29. Descripción del caso de uso "Gestionar permiso"

Caso de uso:	Gestionar permiso
Actores:	Reloj: (Inicia)
Resumen:	El caso de uso inicia cuando se ejecuta la tarea programada para gestionar permisos, luego se obtiene de Estructura y Composición las áreas a las cuales se les gestionarán los permisos. De cada una de estas áreas se obtienen los cambios ocurridos en cuanto a permisos en un tiempo determinado. Se gestionan los permisos en el Archivo Universitario de estas estructuras y finaliza el caso de uso.
Referencias:	RF 14, RF 15 , RF 16
Prioridad:	Alta
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. Selecciona acción a desarrollar: Si desea Crear permiso ver Sección "Crear permiso".

	<p>Si desea Modificar permiso ver Sección “Modificar permiso”.</p> <p>Si desea Eliminar permiso ver Sección “Eliminar permiso”.</p>
Sección “Crear permiso”	
Acción del Actor	Respuesta del Sistema
	<p>2. Se obtiene de Estructura y Composición las áreas, se identifican los usuarios y los permisos de estos asociados a la misma. Luego haciendo uso de los servicios que proporciona el ECM Alfresco para ello, se obtiene el espacio correspondiente a la estructura, los usuarios de la misma y se le asignan los permisos correspondientes, finalizando el caso de uso.</p>
Prototipo de interfaz	
Sección “Modificar permiso”	
Acción del Actor	Respuesta del Sistema
	<p>2. Se obtiene de Estructura y Composición las áreas a las cuales se le han modificados permisos en un momento determinado, se identifican los usuarios y los permisos de estos asociados a la misma. Luego haciendo uso de los servicios que proporciona el ECM Alfresco para ello, se obtiene el espacio correspondiente a la estructura, los usuarios de la misma y se le modifican los permisos</p>

	correspondientes, finalizando el caso de uso.
Prototipo de Interfaz	
Sección "Eliminar permiso"	
Acción del Actor	Respuesta del Sistema
	2. Se obtiene de Estructura y Composición las áreas a las cuales se le han eliminados permisos en un momento determinado, un permiso solo se elimina si dejara de existir un usuario, o una responsabilidad o un espacio, luego se identifican los usuarios que sufrieron cambios y haciendo uso de los servicios que proporciona el ECM Alfresco para ello, se actualiza esta información en el Archivo Universitario y finaliza el caso de uso.
Prototipo de Interfaz	

Tabla 30. Descripción del caso de uso "Realizar auditoría"

Caso de uso:	Realizar auditoría
Actores:	Administrador del sistema: (Inicia)

Resumen:	El caso de uso inicia cuando el administrador del sistema decide realizar auditorías a los cambios ocurridos con los permisos de los usuarios y finaliza el caso de uso.
Precondiciones:	
Referencias:	RF 17
Prioridad:	Alta
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso inicia cuando el administrador del sistema decide realizar auditorías a los cambios ocurridos con los permisos de los usuarios, seleccionando la opción: "Realizar auditoría"	2. El sistema permite realizar búsqueda simple o avanzada. Luego se muestran los usuarios que sufrieron cambios con los permisos, revelando si se creó un permiso, si se modificó, por cuál fue, o si se eliminó, la fecha y hora en que ocurrieron los cambios, además de la tarea programada que se ejecutó y finaliza el caso de uso.

Historial de permisos

Búsqueda simple

Fecha
 desde hasta

Búsqueda avanzada

Filtrar búsqueda ▾

Tarea ejecutada ▾ ✖ Permisos ▾ ✖

Total: 4 Página de 4

Fecha	Hora	Usuario	Permisos que posee	Modificado por	Permisos eliminados	Tarea programada
2012-05-12	01:04:28	Liset	Lector	-	Lector	Tarea 1
2012-05-12	01:04:28	Orelsys	Lector	-	Lector	Tarea 1
2012-05-12	01:04:28	Yoya	Colaborador	Editor	Colaborador	Tarea 1
2012-05-13	09:10:35	Bernardo	Coordinador	-	Coordinador	Tarea 4
2012-05-13	09:10:35	Yuri	Contribuyente	Colaborador	-	Tarea 4

Tabla 31. Descripción del caso de uso "Gestionar asociación estructura con espacio"

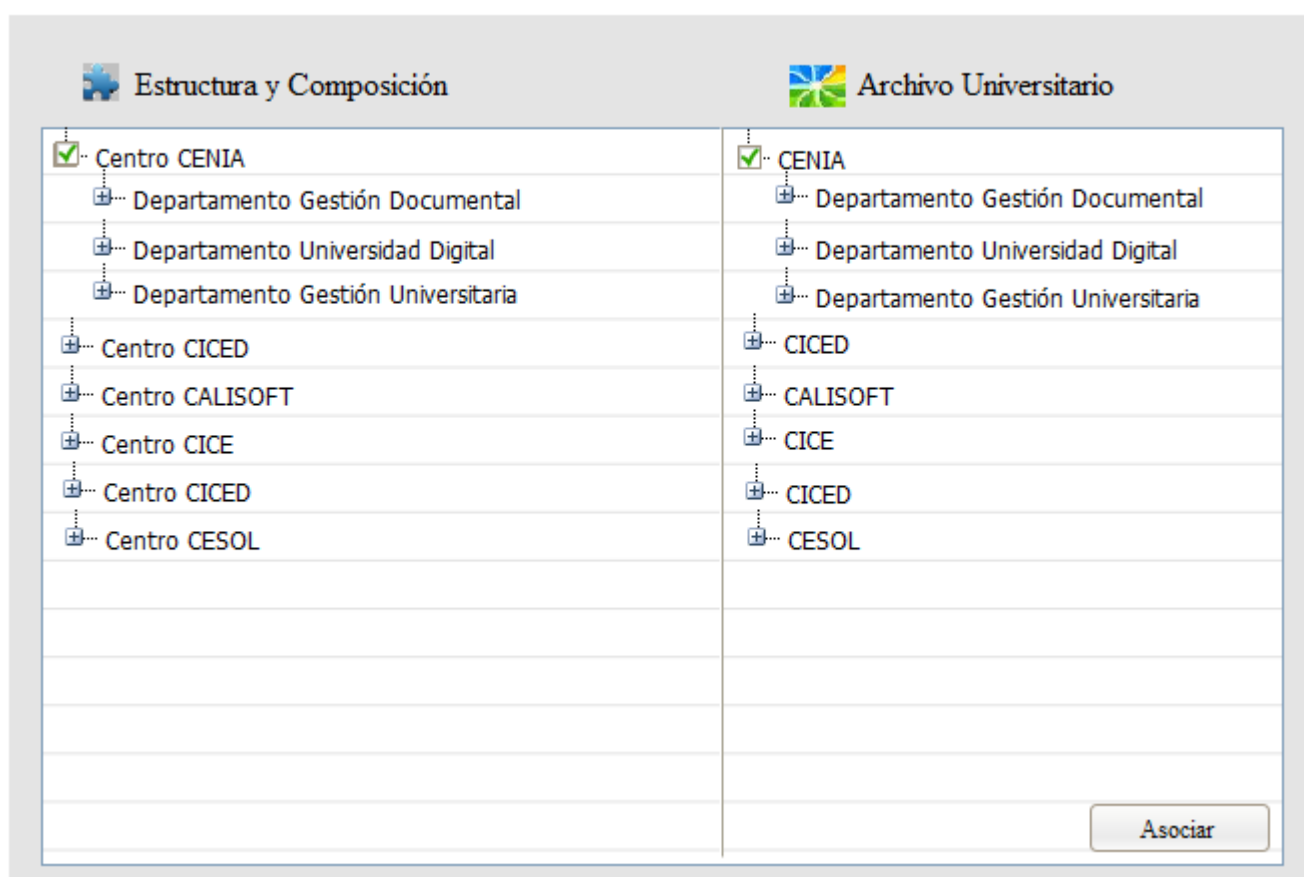
Caso de uso:	Gestionar asociación estructura con espacio
Actor:	Administrador del sistema: (Inicia)

Resumen:	El caso de uso inicia cuando el administrador del sistema decide crear, modificar o eliminar una asociación entre un área de Estructura y Composición y un espacio del Archivo Universitario, finalizando el caso de uso.
Precondiciones:	
Referencias:	RF 1, RF 2, RF 3
Prioridad:	Alta
Flujo Normal de Eventos	
Acción del actor	Acción del sistema
1. El caso de uso inicia cuando el administrador del sistema decide crear, modificar o eliminar una asociación entre un área de Estructura y Composición y un espacio del Archivo Universitario, seleccionando la opción: "Gestionar asociación estructura-espacio"	2. Selecciona acción a desarrollar: Si desea Asociar estructura con espacio ver Sección "Asociar estructura con espacio". Si desea Modificar asociación de estructura con espacio ver Sección "Modificar estructura con espacio". Si desea Eliminar asociación de estructura con espacio ver Sección "Eliminar asociación de estructura con espacio".
Escenario "Asociar estructura con espacio"	
Acción del actor	Respuesta del sistema

<p>Selecciona la opción: "Crear asociación".</p>	<p>El sistema muestra en la parte izquierda las áreas correspondientes a Estructura y Composición y en la derecha los espacios pertenecientes al Archivo Universitario que no han sido asociados.</p>
<p>Selecciona de Estructura y Composición el área y del Archivo Universitario el espacio que desea asociar. Luego selecciona la opción "Asociar".</p>	<p>El sistema realiza la asociación y la registra en un fichero XML, finalizando el caso de uso.</p>

Prototipo de Interfaz

Crear asociación estructura-nodo



Escenario "Modificar asociación de estructura con espacio"

Acción del Actor	Respuesta del Sistema
3. Selecciona de las asociaciones existentes la que desea modificar, presionando la opción: “Modificar asociación”.	4. El sistema muestra en la parte izquierda las áreas correspondientes a Estructura y Composición y en la derecha los espacios pertenecientes al Archivo Universitario que no han sido asociados.
5. Selecciona de Estructura y Composición el área y del Archivo Universitario el espacio que desea asociar. Luego selecciona la opción “Guardar”.	6. El sistema realiza la asociación y la registra en un fichero XML, finalizando el caso de uso.

Prototipo de Interfaz

Modificar asociación estructura-nodo

The interface displays two hierarchical tree structures for comparison and association. The left tree, 'Estructura y Composición', shows a tree starting with 'Centro CENIA' which has three sub-nodes: 'Departamento Gestión Documental' (with a green checkmark), 'Departamento Universidad Digital', and 'Departamento Gestión Universitaria'. Below this are 'Centro CALISOFT', 'Centro CICE', and 'Centro CICED'. The right tree, 'Archivo Universitario', shows a tree starting with 'CENIA' which has four sub-nodes: 'Departamento Gestión Documental' (with a red X), 'Dpto Gestión Documental' (with a green checkmark), 'Departamento Universidad Digital', and 'Departamento Gestión Universitaria'. Below this are 'CALISOFT', 'CICE', and 'CICED'. A 'Guardar' button is positioned at the bottom right of the interface area.

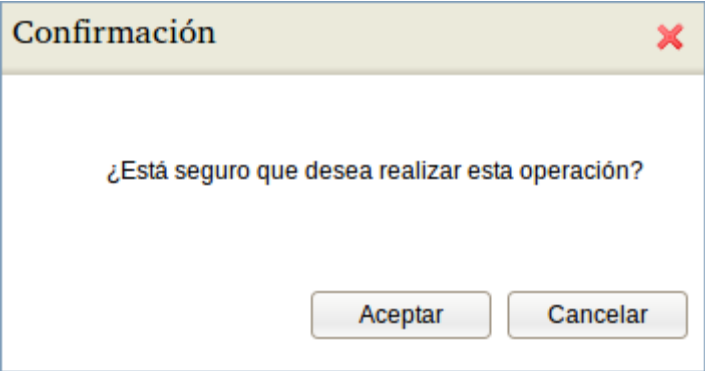
Escenario "Eliminar asociación de estructura con espacio"	
Acción del Actor	Respuesta del Sistema
3. Selecciona de las asociaciones existentes la que desea eliminar, presionando la opción: "Eliminar asociación".	4. El sistema muestra en pantalla un mensaje de alerta "¿Está seguro que desea realizar esta operación?".
5. Presiona la opción "Aceptar"	6. El sistema elimina la información y finaliza el caso de uso.
Prototipo de Interfaz	
	
Poscondiciones	

Tabla 32. Descripción del caso de uso "Gestionar tarea programada"

Caso de uso:	Gestionar tarea programada
Actores:	Administrador del sistema: (Inicia)
Resumen:	El caso de uso inicia cuando el administrador del sistema decide crear, modificar o eliminar una tarea programada

	finalizando el caso de uso.
Precondiciones:	
Referencias:	RF 4, RF 5, RF 6
Prioridad:	Alta
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<p>1. El caso de uso inicia cuando el administrador del sistema decide crear, modificar o eliminar una o varias tareas programadas, seleccionando la opción:</p> <p>"Gestionar tarea programada"</p>	<p>2. Selecciona acción a desarrollar:</p> <p>Si desea Crear tarea programada ver Sección "Crear tarea programada".</p> <p>Si desea Modificar tarea programada ver Sección "Modificar tarea programada".</p> <p>Si desea Eliminar tarea programada ver Sección "Eliminar tarea programada".</p>
Sección "Crear tarea programada"	
Acción del Actor	Respuesta del Sistema
	<p>3. El sistema muestra en pantalla un formulario con los campos a llenar para crear la tarea programada así como las opciones de:</p> <p>"Aceptar"</p> <p>"Cancelar"</p>
<p>4. Llena los campos del formulario</p> <ul style="list-style-type: none"> ✓ nombre, ✓ descripción, ✓ frecuencia (diariamente, semanalmente, mensualmente, trimestralmente, semestralmente), ✓ categorías de personas (estudiantes, profesores o 	<p>5. Verifica que los datos sean correctos.</p>

<p>trabajadores),</p> <ul style="list-style-type: none"> ✓ responsabilidad (jefe de grupo, jefe de departamento, decano), ✓ rol (lector, editor, contribuyente, colaborador, coordinador), ✓ configuración de la planificación (la misma puede iniciarse al cabo de n minutos), <p>y selecciona una de las siguientes opciones :</p> <p>“Aceptar”</p> <p>“Cancelar”</p>	
	<p>6. Verifica que todos los campos obligatorios estén completos.</p>
	<p>7. Crea la tarea programada.</p>
<p>Flujo Alternativo Sección “Crear tarea programada”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>5.1 Muestra en pantalla el mensaje de alerta.</p>
	<p>6.1 Muestra en pantalla un mensaje por cada campo obligatorio.</p>
<p>Prototipo de interfaz</p>	

Crear tarea programada



Formulario para crear una tarea programada. Incluye campos para "Modo de ejecución:" (con opción "Al iniciar la aplicación en 2 minutos"), "Nombre:" (campo de texto), "Descripción:" (área de texto), y botones "Siguiete" y "Cancelar".

Crear tarea programada



Formulario para configurar la planificación de una tarea programada. Incluye "Configuración de la planificación:" con un control deslizante para "Iniciar tarea después de" (0 minutos). Campos de selección para "Frecuencia:", "Responsabilidad:", "Categoría de personas:", y "Rol:". Botones "Aceptar" y "Cancelar".

Sección "Modificar tarea programada"

Acción del Actor

Respuesta del Sistema

3. El sistema muestra en pantalla un formulario con los campos a modificar de la tarea programada así como las opciones de "Salvar" y "Cancelar"

<p>4. Modifica los campos del formulario</p> <ul style="list-style-type: none"> ✓ nombre, ✓ descripción, ✓ frecuencia (diariamente, semanalmente, mensualmente, trimestralmente, semestralmente), ✓ categorías de personas (estudiantes, profesores o trabajadores), ✓ responsabilidad (jefe de grupo, jefe de departamento, decano), ✓ rol (lector, editor, contribuyente, colaborador, coordinador), ✓ configuración de la planificación (la misma puede iniciarse al cabo de n minutos), <p>y selecciona una de las siguientes opciones : “Salvar” “Cancelar.”</p>	<p>5. Verifica que los datos sean correctos</p>
	<p>6. Verifica que los campos obligatorios estén completos.</p>
	<p>7. Crea la tarea programada.</p>
<p>Flujo Alterno Sección “Modificar tarea programada”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>5.1. Muestra en pantalla el mensaje “Los datos están escritos incorrectamente”.</p>
	<p>6.1. Muestra en pantalla un mensaje por cada campo obligatorio.</p>
<p>Prototipo de Interfaz</p>	

Crear tarea programada



Modo de ejecución: *
 Al iniciar la aplicación en 2 minutos

Nombre: *

Descripción:

Crear tarea programada



Configuración de la planificación: *
 Iniciar tarea después de minuto (s)

Frecuencia: *

Responsabilidad: *

Categoría de personas: *

Rol: *

Sección "Eliminar tarea programada"

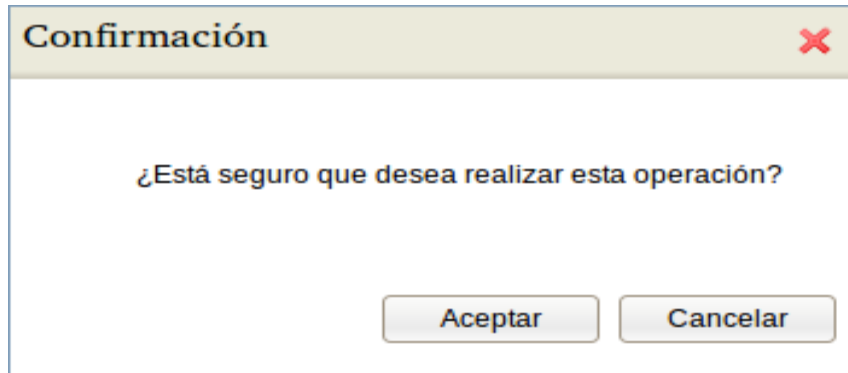
Acción del Actor

Respuesta del Sistema

5. El usuario presiona el botón "Aceptar".

4. El sistema muestra en pantalla un mensaje de alerta "¿Está seguro que desea realizar esta operación?".
6. El sistema elimina la información y finaliza el caso de uso.

Prototipo de Interfaz



Poscondiciones

Anexo B

B.1 Diagramas de clases del análisis

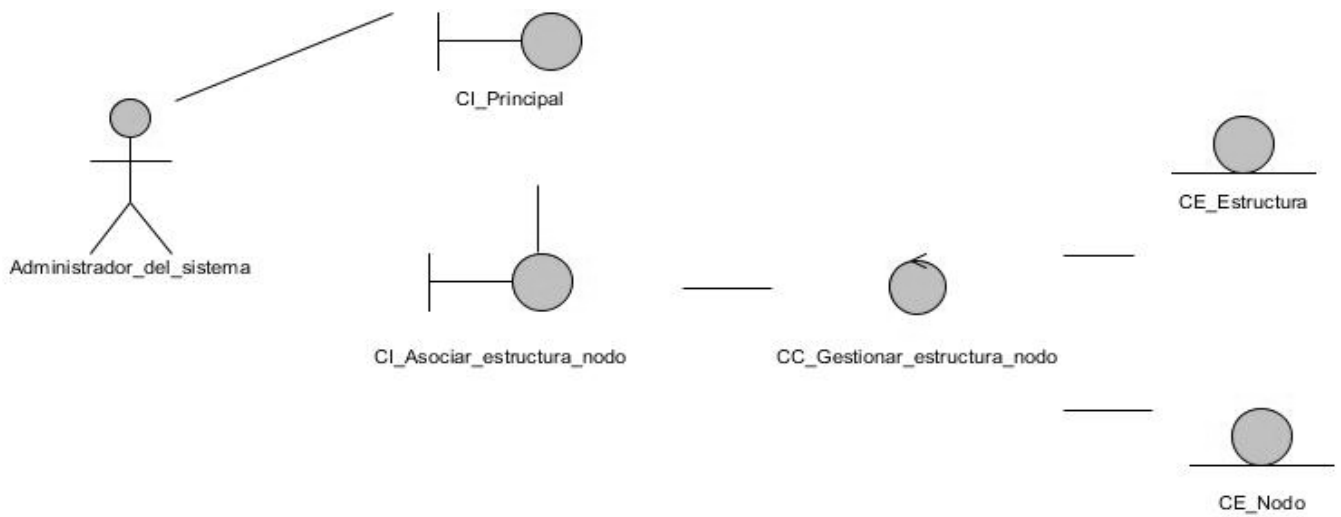


Figura 21. Diagrama de clases del análisis para el caso de uso “Asociar estructura con espacio”

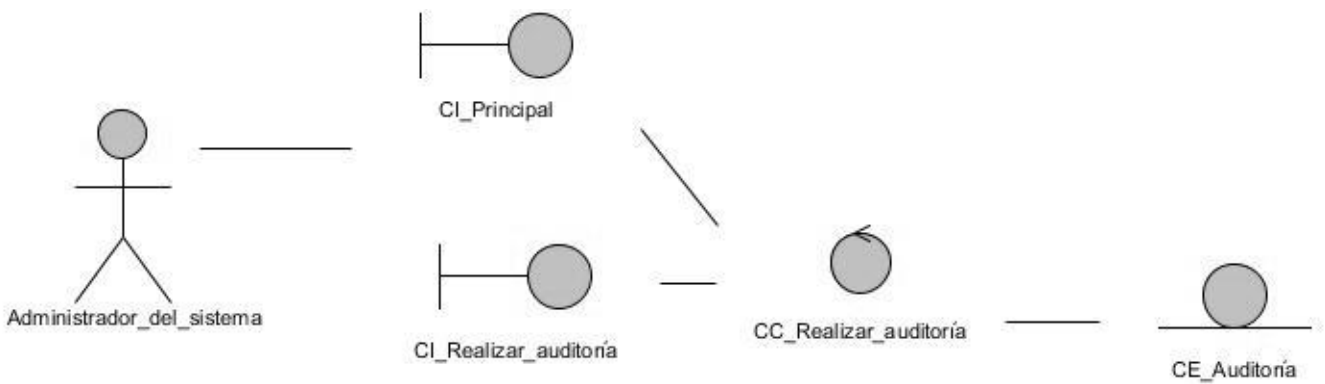


Figura 22. Diagrama de clases del análisis para el caso de uso “Realizar auditoría”

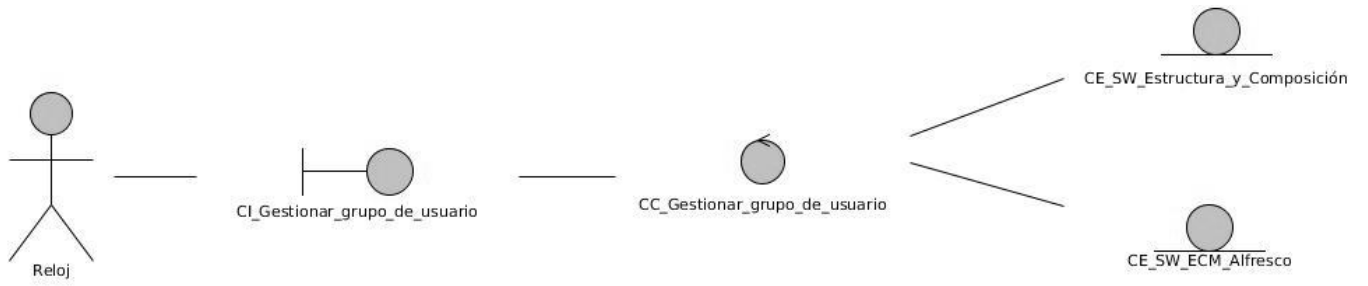


Figura 23. Diagrama de clases del análisis para el caso de uso “Gestionar grupo de usuario”

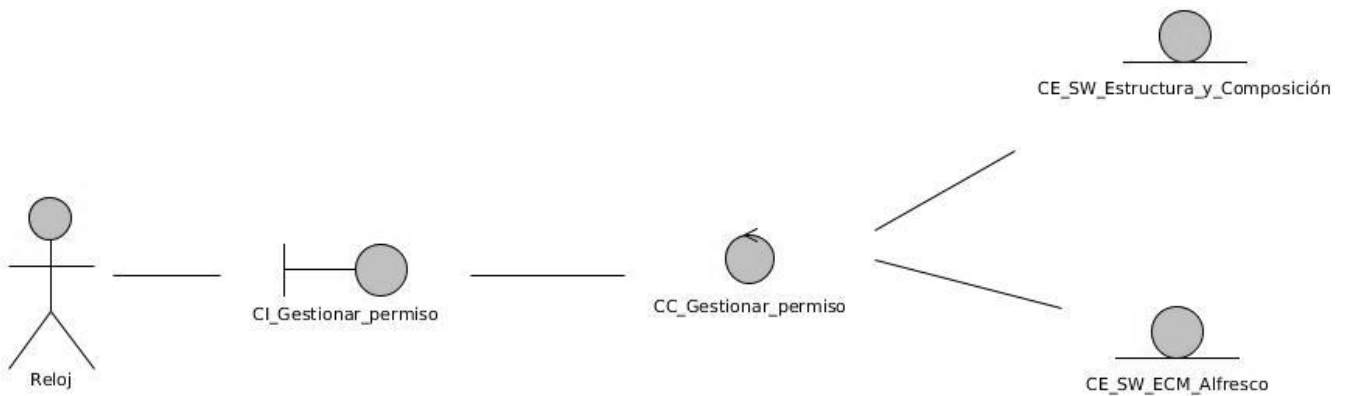


Figura 24. Diagrama de clases del análisis para el caso de uso “Gestionar permiso”

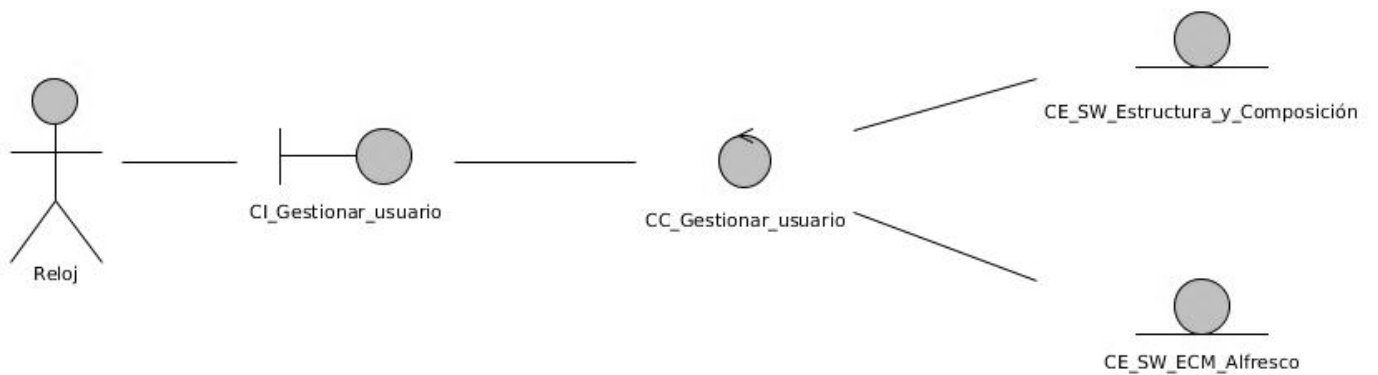


Figura 25. Diagrama de clases del análisis para el caso de uso “Gestionar usuario”

B.2 Diagramas de colaboración del análisis

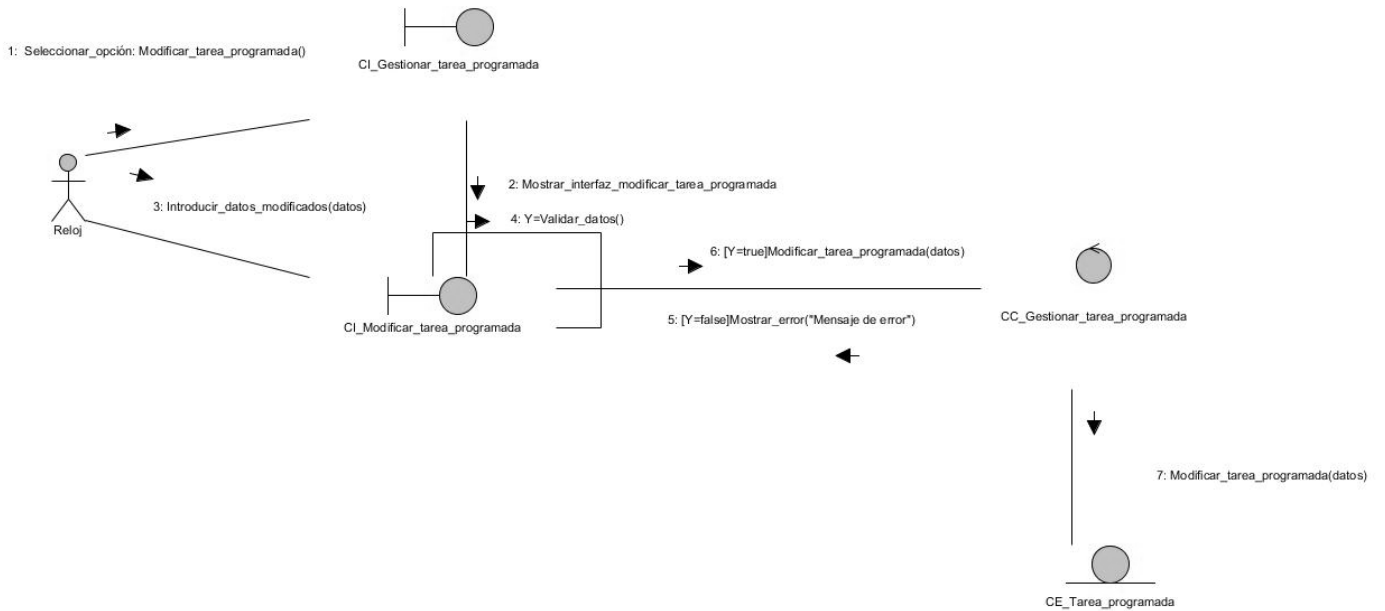


Figura 26. Diagrama de colaboración del análisis para el caso de uso “Modificar tarea programada”

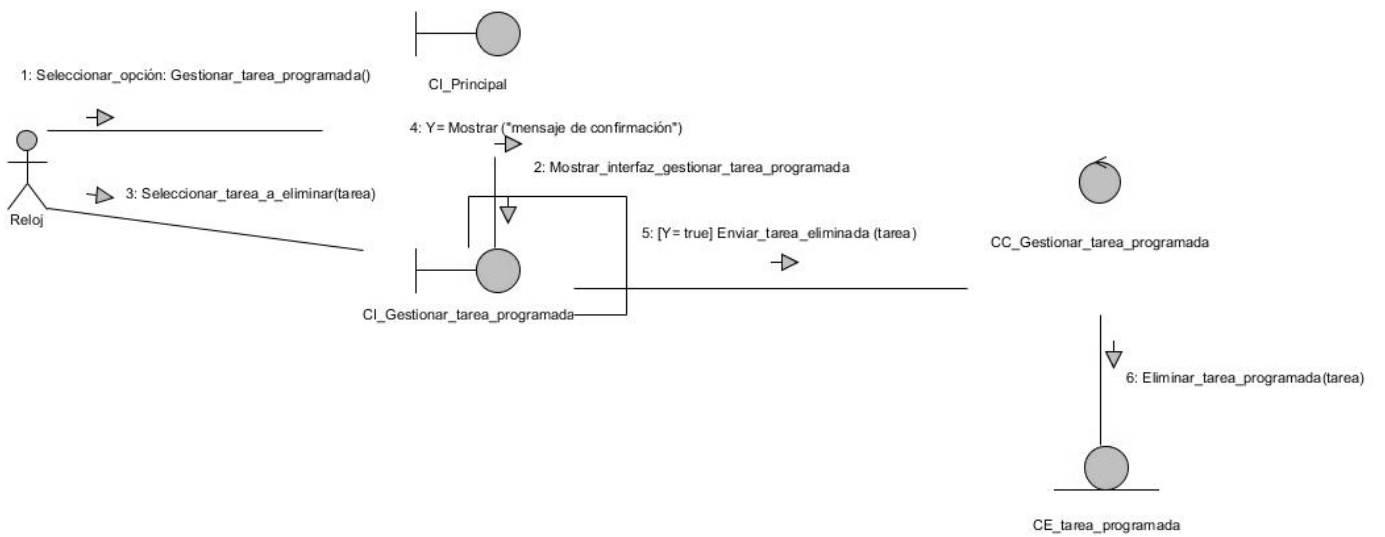


Figura 27. Diagrama de colaboración del análisis para el caso de uso “Eliminar tarea programada”

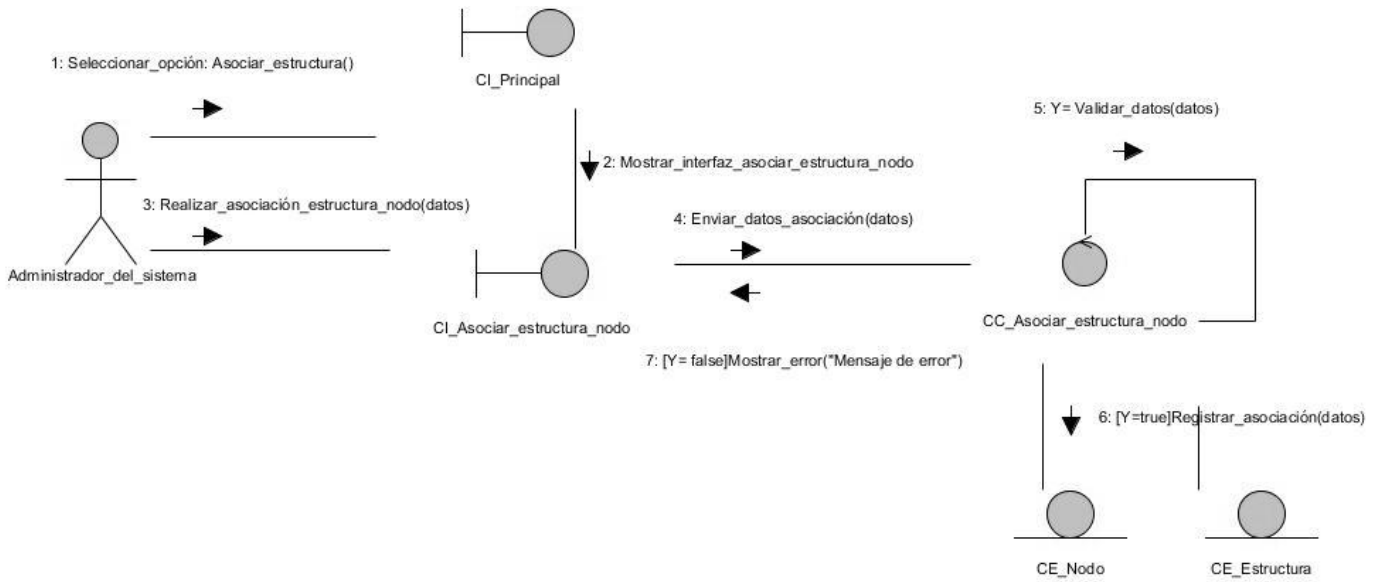


Figura 28. Diagrama de colaboración del análisis para el caso de uso “Asociar estructura con espacio”

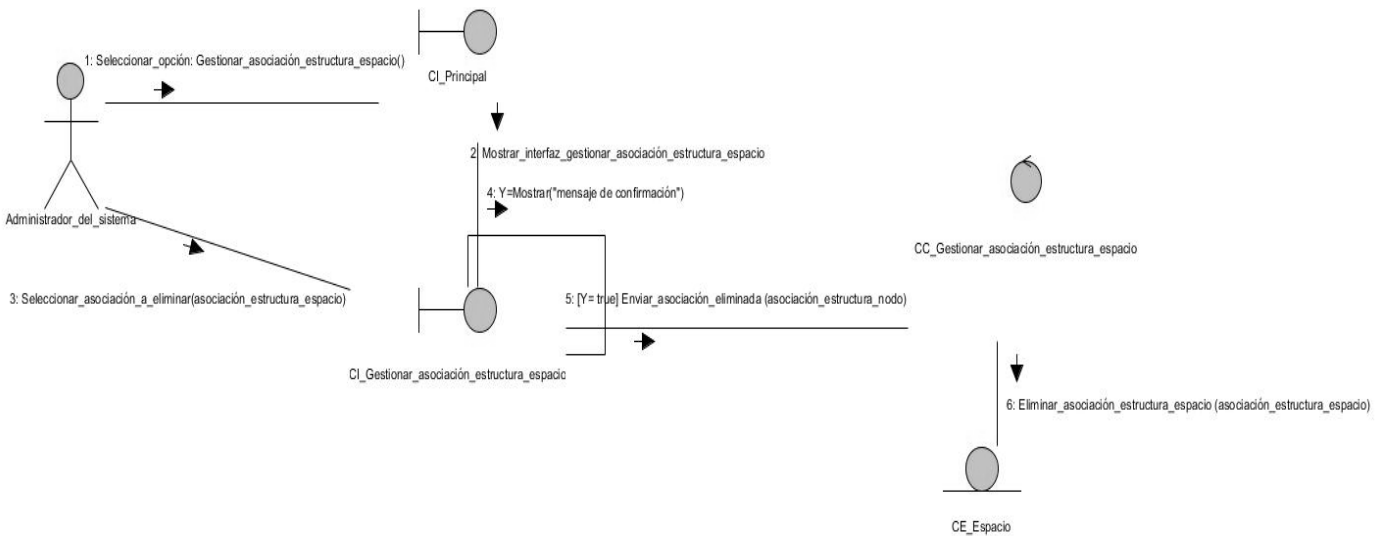


Figura 29. Diagrama de colaboración del análisis para el caso de uso “Eliminar estructura con espacio”

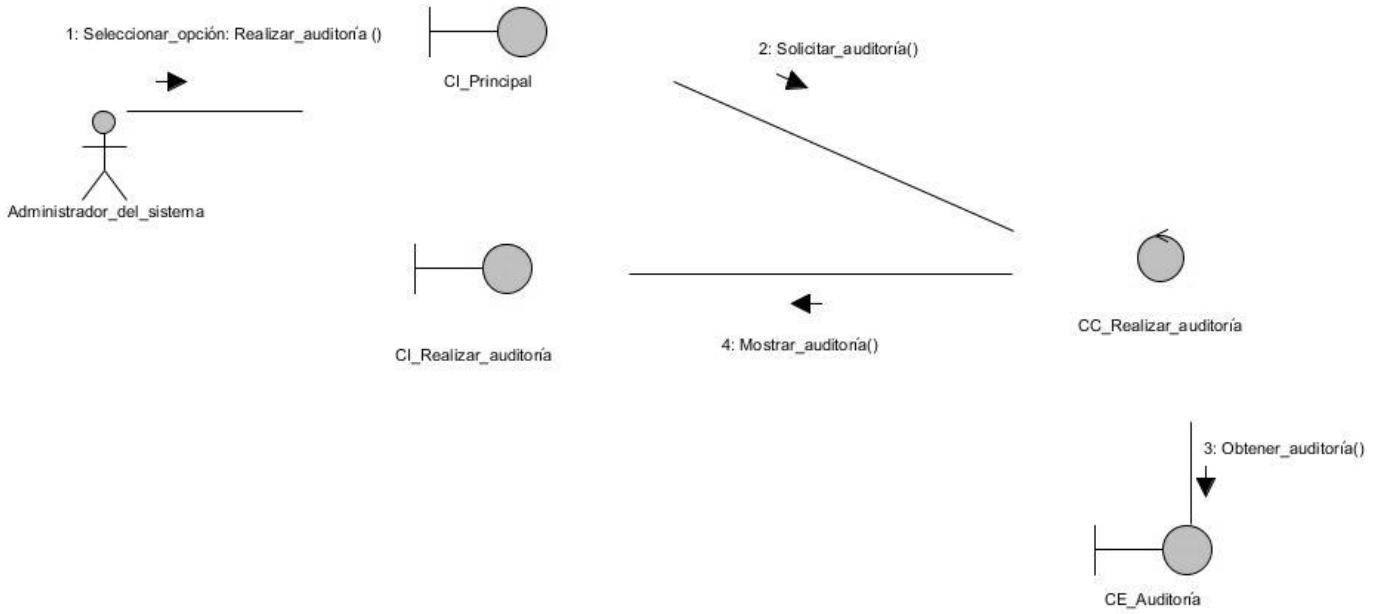


Figura 30. Diagrama de colaboración del análisis para el caso de uso “Realizar auditoría”

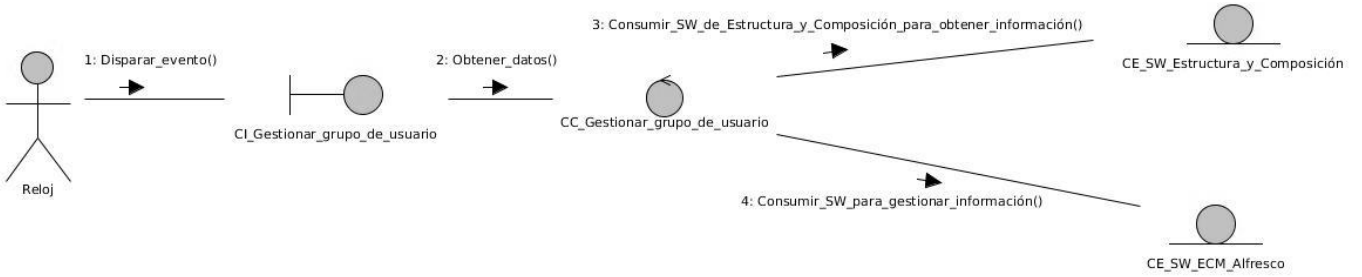


Figura 31. Diagrama de colaboración del análisis para el caso de uso “Gestionar grupo de usuario”

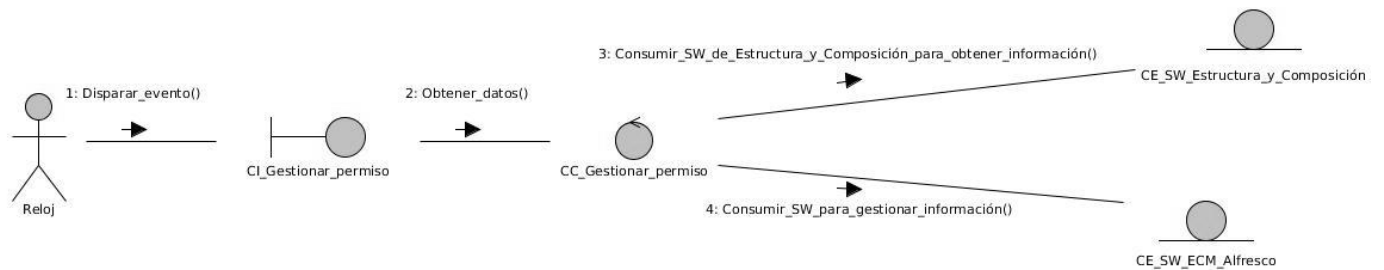


Figura 32. Diagrama de colaboración del análisis para el caso de uso “Gestionar permiso”

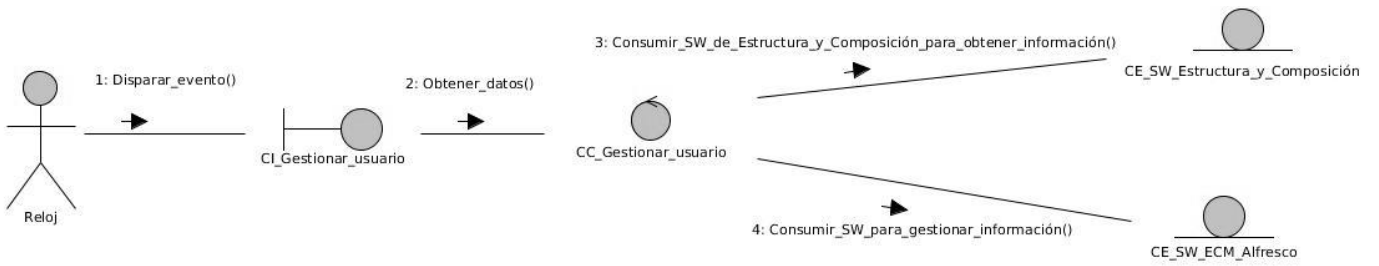


Figura 33. Diagrama de colaboración del análisis para el caso de uso “Gestionar usuario”

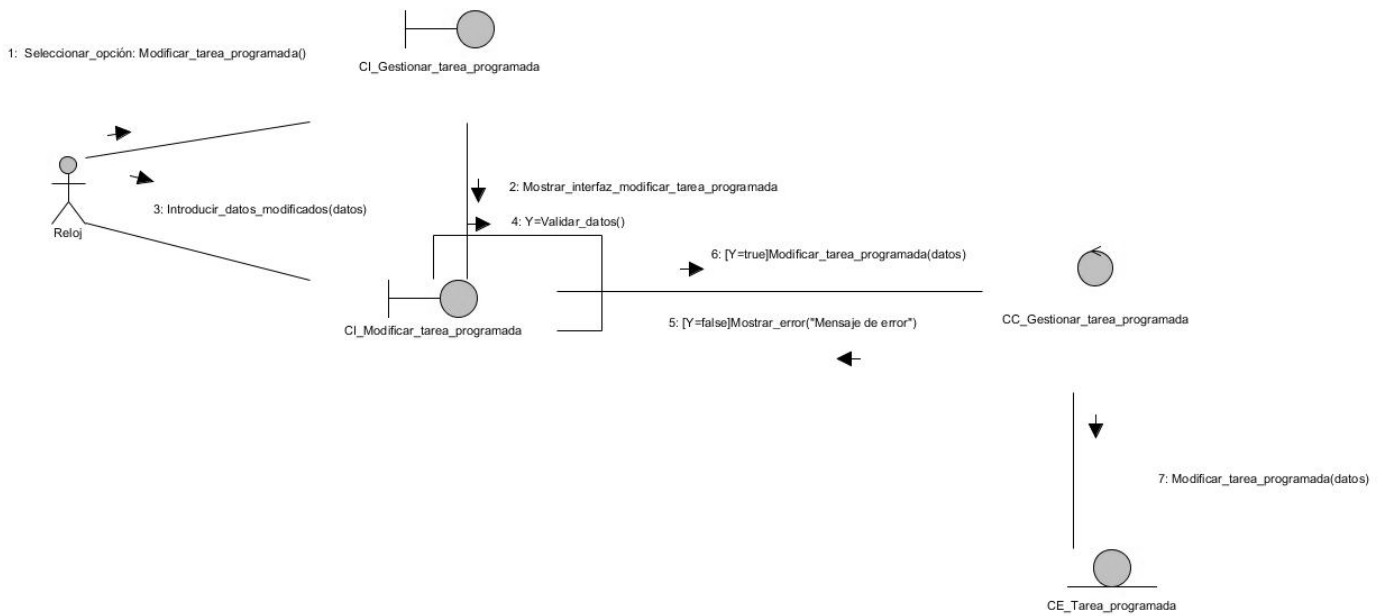


Figura 34. Diagrama de colaboración del análisis para el caso de uso “Modificar tarea programada”

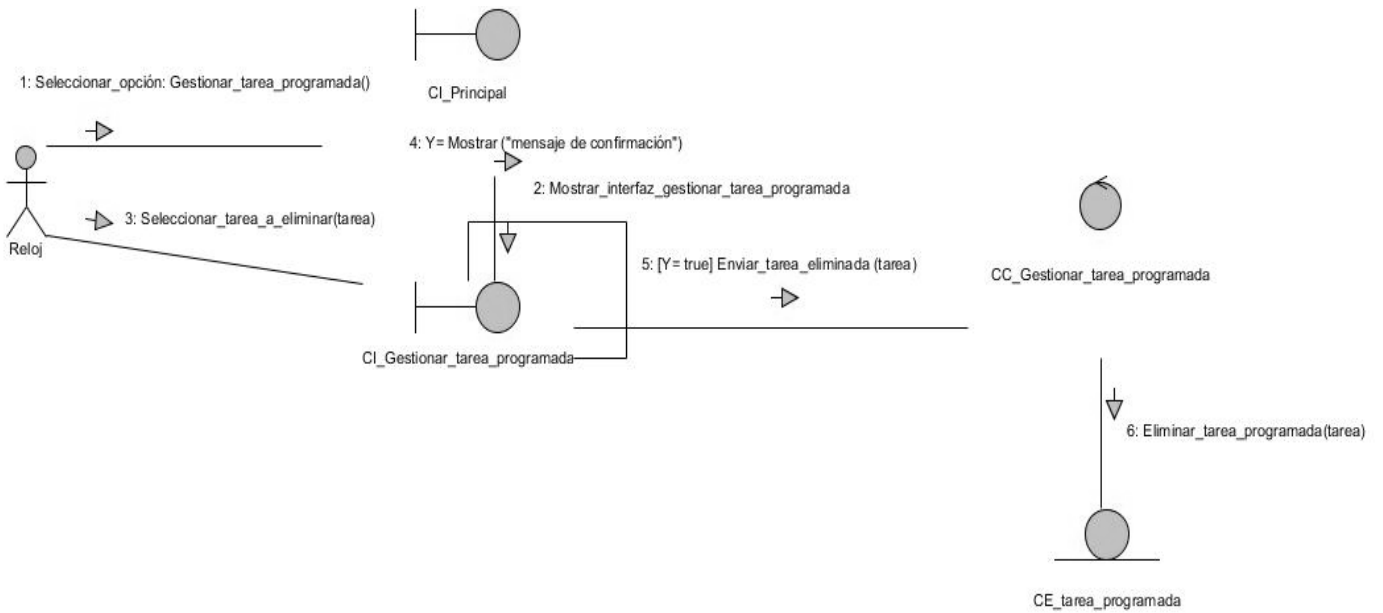


Figura 35. Diagrama de colaboración del análisis para el caso de uso “Eliminar tarea programada”

B.3 Diagramas de clases del diseño

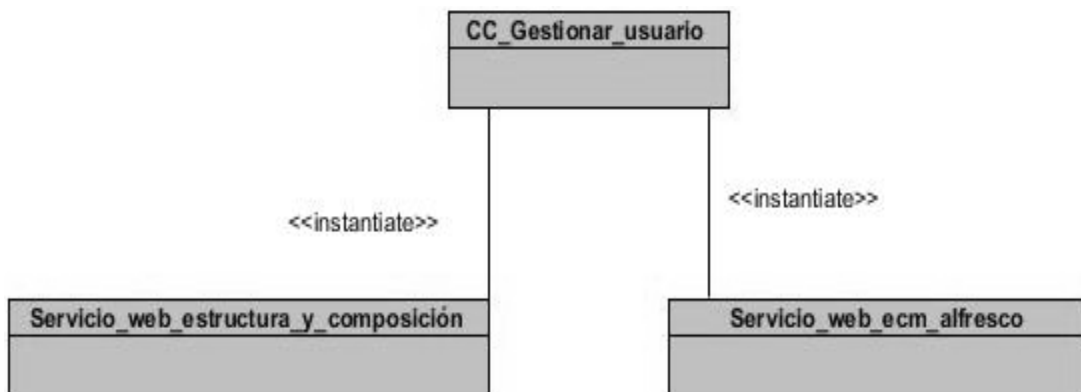


Figura 36. Diagrama de clases del diseño para el caso de uso “Gestionar usuario”

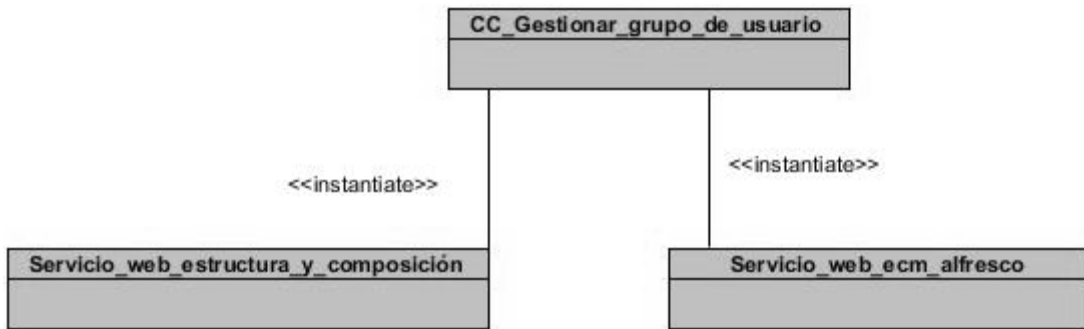


Figura 37. Diagrama de clases del diseño para el caso de uso “Gestionar grupo de usuario”

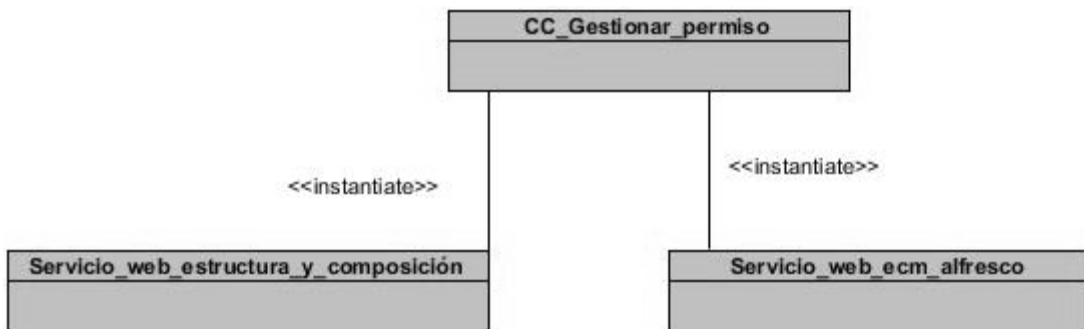


Figura 38. Diagrama de clases del diseño para el caso de uso “Gestionar permiso”

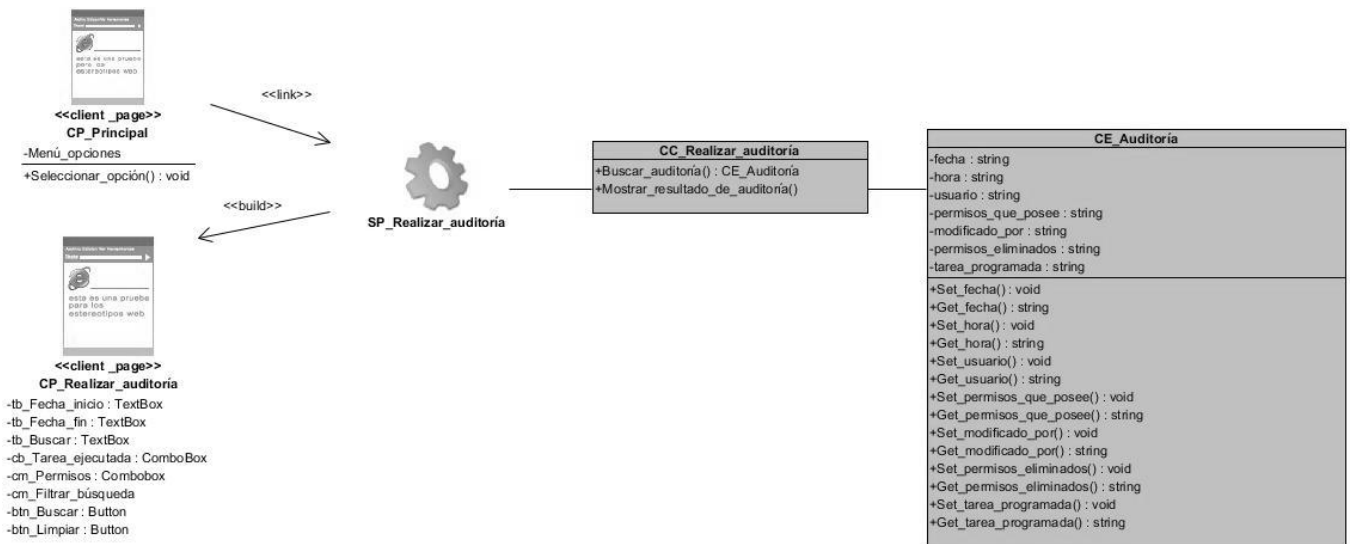


Figura 39. Diagrama de clases del diseño para el caso de uso “Realizar auditoría”

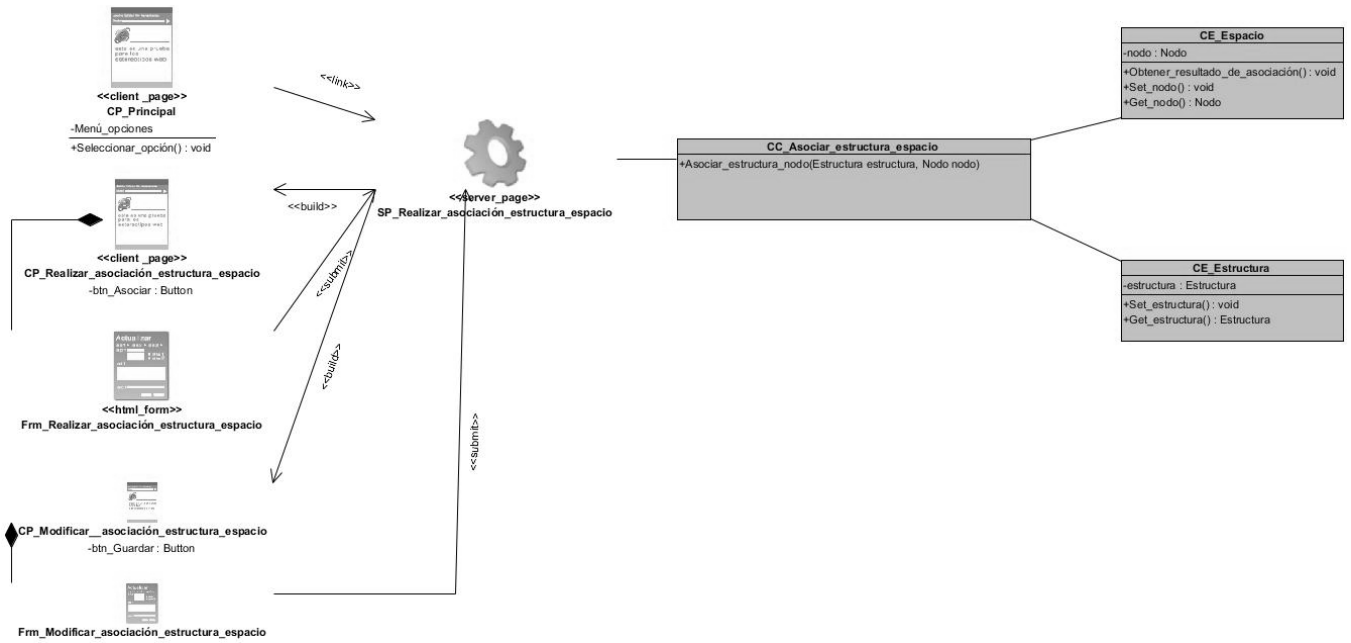


Figura 40. Diagrama de clases del diseño para el caso de uso “Asociar estructura con espacio”