



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 2

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

Módulo Gestión de Acuerdo en la tecnología de Alfresco Community 5.2

Autores: Pablo Ernesto Páez Lapchinskiy

Alay Marrero González

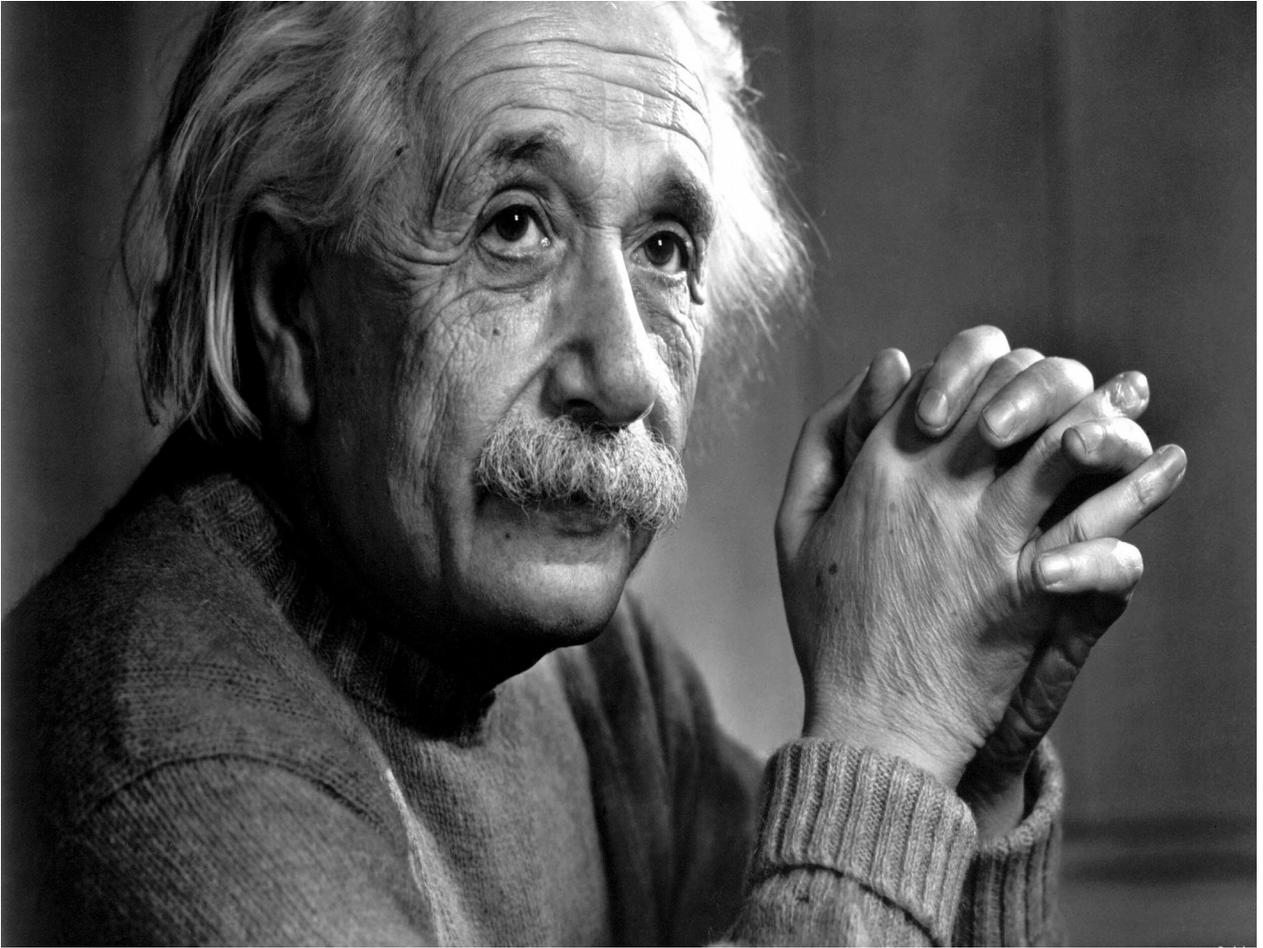
Tutores: MSc. Aurelio Antelo Collado

Ing. Liomar Rodríguez Guerra

Ing. Geider Arévalo Quintana

La Habana, 2019

“Año 61 de la Revolución”



“Hay una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad.”

Albert
Einstein

Declaración de autoría

Declaramos ser autores de la presente tesis que tiene por título: Módulo Gestión de Acuerdo en la tecnología de Alfresco Community 5.2 y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmamos la presente a los __ días del mes de _____ del año ____.

Pablo Ernesto Páez Lapchinskiy

Alay Marrero González

Msc. Aurelio Antelo Collado

Firma del Autor

Ing. Liomar Rodríguez Guerra

Firma del Autor

Agradecimientos

Agradecimiento a mis padres por el apoyo que han depositado en mí y por su confianza en todo momento. Gracias por su educación y por existir, los amo mucho.

A mi hermana por acompañarme en todo momento en momentos buenos y malos. Gracias por tu comprensión y por ser la persona que eres, te quiero mucho.

A todos mis amigos en el transcurso de estos años por su apoyo moral e incondicional.

Agradecer a la canoa por la amistad que creamos y la hermandad que tenemos, gracias a ustedes por todo y por habernos mantenido tan real hasta el final.

Gracias a todos

Dedicatoria

Dedicado a mis padres, mi hermana y a todas las personas que de una forma u otra me apoyaron.

Resumen

El Gestor de Documentos Administrativos XABAL eXcriba en su versión 3.1, es un software realizado en la Universidad de Ciencias Informáticas, diseñado para la gestión documental, que permite el trámite de los documentos que se generan o reciben en una organización. El software cuenta con un módulo Gestión de Acuerdo el cual permite la creación, modificación, prorrogación y aplazamiento de los acuerdos que se generan en las empresas y organizaciones. La investigación actual contribuye al desarrollo de un módulo de Gestión de Acuerdo en la tecnología Alfresco Community 5.2 que permita el manejo de la información y un control más efectivo de los acuerdos que se generan en las organizaciones. La investigación consta de tres capítulos que abarcan desde la Fundamentación teórica, la Propuesta de solución y la implementación y prueba. Durante el desarrollo del módulo se utilizaron como lenguajes de desarrollo JavaScript 1.6, XML 1.0 y HTML 5 y como herramientas de desarrollo el editor de texto Sublime text 3, visual Studio Code, AJAX, Webscript, FreeMaker, y Visual Paradigm para UML 8.0 para el modelado de los artefactos. La metodología usada fue AUP UCI enfocado en el escenario 2. Fue definida una estrategia de pruebas que permitió corroborar el éxito del desarrollo de la solución, en el escenario de validación.

Palabras claves: Alfresco Community 5.2, gestión, gestión de acuerdo, gestión de reuniones, acuerdo, reunión.

Summary

The Manager of Administrative Documents XABAL eXcriba in version 3.1 is a software made at the University of Computer Science, designed for document management, which allows the processing of documents that are generated or received in an organization. The software has an agreement management module which allows the creation, modification, extension and postponement of the agreements that are generated in the companies and organizations. The current research contributes to the development of an Agreement Management module in the Alfresco Community 5.2 technology that allows the management of information and a more effective control of the agreements that are generated in organizations. The research consists of three chapters that range from the theoretical foundation, the proposed solution and the implementation and testing. During the development of the module, JavaScript 1.6, XML 1.0, Alfresco Community 5.2 and HTML5 were used as development tools: Visual Paradigm for UML 8.0, Apache Tomcat 7.0 and text editor Sublime text 3 and visual code. A test strategy was defined to corroborate the success of the development of the solution, in the validation scenario.

Keywords: Alfresco 5.2, management, management of agreement, management of meetings, system, agreement, meeting, document management system.

Índice

Capítulo 1: Fundamentación Teórica.....	6
1.1. Conceptos asociados a la investigación.....	6
1.2. Sistemas similares.....	7
1.2.1. Resultados del estudio.....	11
1.3. Metodologías de desarrollo de software.....	12
1.4 Ambiente de desarrollo.....	14
1.5. Conclusiones del capítulo.....	17
CAPITULO 2: Propuesta de solución.....	18
2.1 Modelo de Dominio.....	18
2.2 Especificación de los requisitos de software.....	19
2.2.1 Requisitos funcionales.....	19
2.2.2. Requisitos no funcionales.....	27
2.3. Definición de los casos de usos del sistema.....	28
2.3.1. Definición de los actores.....	28
2.3.2. Diagrama de caso de uso del sistema.....	29
2.3.3 Descripción de casos de uso del sistema.....	29
2.4. Matriz de trazabilidad de requisitos.....	43
2.5. Arquitectura del software.....	44
2.5.1. Arquitectura en capas.....	44
2.6. Patrones de diseño.....	46
2.6.1 Patrones GRASP.....	46
2.6.2. Patrones GOF.....	47
2.7. Conclusiones parciales.....	48
Capítulo 3: Implementación y pruebas.....	49
3.1. Modelo contenido.....	49
3.2. Webscript.....	49
3.3. Interfaz gráfica del módulo.....	53
3.4. Diagrama de despliegue.....	54
3.4.1. Despliegue de la solución.....	55
3.5. Pruebas de software.....	56
3.5.1. Pruebas de caja blanca.....	56
3.5.2. Pruebas de caja negra.....	59
Conclusiones parciales.....	63

Conclusiones Generales.....	64
Referencias.....	66
Anexos.....	70
Anexo 1. Tablas descripción de los casos de uso.....	70
Anexo 2. Matrices de trazabilidad.....	119
Anexo 3. Casos de prueba.....	121

Índice de figuras

Figura 1: Escenario 2: Metodología AUP-UCI.....	13
Figura 2: Modelo de dominio. Fuente: Elaboración propia.....	18
Figura 3: Diagrama de casos de uso.....	29
Figura 4: Matriz de trazabilidad CU-RF.....	44
Figura 5: Descripción de la arquitectura.....	46
Figura 6: Listado de webscript de acceso al repositorio.....	50
Figura 7: Crear reunión.xml.....	51
Figura 8: crear reunion.js.....	51
Figura 9: crear reunion.js.....	52
Figura 10: crear reunión.ftl.....	53
Figura 11: Interfaz principal.....	53
Figura 12: Diagrama de despliegue.....	54
Figura 13: Estructura de carpeta del módulo Gestión de Acuerdo.....	55
Figura 14: Código de la función Cancelar Reunión.....	58
Figura 15: Gráfico de flujo de la función Cancelar Reunión.....	58
Figura 16: Caso de prueba: Replanificar reunión.....	60
Figura 17: Resultados de las pruebas.....	62
Figura 18: Matriz de trazabilidad de RF-RF.....	119
Figura 19: Matriz de trazabilidad CU_CP.....	120
Figura 20: Matriz de trazabilidad RF-MD.....	120

Índice de tablas

Tabla 1: Requisitos funcionales.....	19
Tabla 2 Descripción de los actores.....	28
Tabla 3: CU_Gestionar reunión.....	29
Tabla 4: CU_Gestionar acuerdos.....	35
Tabla 5: Iteraciones de las pruebas.....	62
Tabla 6: CU_Ver detalles de reunión.....	70
Tabla 7: CU_Cancelar reunión.....	72
Tabla 8: CU_Replanificar reunión.....	74
Tabla 9: CU_Generar documento Acta de reunión.....	75
Tabla 10: CU_Generar documento Orden del día.....	78
Tabla 11: CU_Calcular índice de acuerdos cumplidos por reunión.....	80
Tabla 12: CU_Visualizar reuniones asignadas a mí.....	82
Tabla 13: CU_Visualizar reuniones creadas por mí.....	83
Tabla 14: CU_Buscar reuniones.....	85
Tabla 15: CU_Filtrar reuniones por propiedades.....	87
Tabla 16: CU_Actualizar estado de acuerdo.....	88
Tabla 17: CU_Aplazar acuerdo.....	90
Tabla 18: CU_Detener acuerdo.....	92
Tabla 19: CU_Reanudar acuerdo.....	94
Tabla 20: CU_Cerrar acuerdo.....	96
Tabla 21: CU_Cancelar acuerdo.....	99
Tabla 22: CU_Visualizar acuerdos creados por mí.....	101
Tabla 23: CU_Visualizar acuerdos asignados a mí.....	102
Tabla 24: CU_Visualizar acuerdos en los que participo.....	104
Tabla 25: CU_Visualizar acuerdos aplazados.....	106
Tabla 26: CU_Visualizar acuerdos por expirar.....	108
Tabla 27: CU_Visualizar acuerdos pendientes de revisión.....	110
Tabla 28: CU_Visualizar acuerdos expirados.....	112
Tabla 29: CU_Visualizar acuerdos pendientes.....	114
Tabla 30: CU_Filtrar acuerdos por propiedades.....	116
Tabla 31: CU_Buscar acuerdos.....	118
Tabla 32: CP_Ver detalles de reunión.....	121
Tabla 33: CP_Cancelar reunión.....	121
Tabla 34: CP_Replanificar reunión.....	122
Tabla 35: CP_Visualizar reuniones asignadas a mí.....	123
Tabla 36: CP_Visualizar reuniones creadas por mí.....	124
Tabla 37: CP_Buscar reuniones.....	124
Tabla 38: CP_Buscar acuerdos.....	125
Tabla 39: CP_Filtrar reuniones por propiedades.....	126
Tabla 40: CP_Filtrar acuerdos por propiedades.....	126
Tabla 41: CP_Visualizar acuerdos creados por mí.....	127
Tabla 42: CP_Visualizar acuerdos asignados a mí.....	127
Tabla 43: CP_Visualizar acuerdos en los que participo.....	128
Tabla 44: CP_Visualizar acuerdos prorrogados.....	128

Tabla 45: CP_Visualizar acuerdos aplazados.....	129
Tabla 46: Visualizar acuerdos por expirar.....	129
Tabla 47: CP_Visualizar acuerdos pendientes de revisión.....	129

Introducción

En la actualidad las empresas se encuentran encaminadas en su perfeccionamiento y están altamente ligadas a la información, donde los hechos son recogidos en documentos, creados o recibidos para su conservación y prueba. En estas entidades se llevan a cabo un conjunto de procesos y servicios tanto internos como externos, los cuales deben ser de mayor calidad, rapidez y con el menor costo posible para poder generar ventajas competitivas sobre otras empresas y organizaciones. A raíz de esto la cantidad de volúmenes de información que se generan resulta ser abrumadora. Para ello surge como necesidad realizar una adecuada estrategia de gestión documental de dicha información.

La aplicación de la gestión documental en una organización, agiliza trámites y ayuda a un mejor y más centrado funcionamiento de la misma, facilitando y apoyando la ejecución de los procesos claves. El surgimiento de las nuevas tecnologías de la información y las comunicaciones (TIC) ha representado un impacto positivo en la gestión documental. A su vez conllevando a desarrollar trabajos en entornos digitales y a la implementación del documento electrónico. Dio paso también a la creación de sistemas informáticos que faciliten las tareas dentro de esta área, brindando un mejor tratamiento y recuperación de la información y automatizando cada uno de los procesos relacionados con la documentación. Para implementar la gestión documental dentro de cualquier organización es necesario contar con un programa de gestión documental que nos permita lograr la transición sin mayores dificultades, para los empleados, y para la organización[CITATION Fer06 \l 3082].

Relacionado a esto están los sistemas de gestión documental (SGD) los cuales aseguran una información organizada y eficiente, que favorezca la productividad empresarial. Son muchos los beneficios que trae consigo la implantación de un SGD[CITATION Pix17 \l 3082], como, por ejemplo:

- Control sobre quién puede acceder a los documentos, en qué momento, y qué actividades puede efectuar sobre las mismas.
- Acceso centralizado y sencillo a toda la documentación relevante de la empresa y posibilidad de colaboración de varias partes en un mismo documento.
- Seguridad, no sólo a la hora de controlar el acceso, sino también en la custodia y copias de seguridad que supone la adopción de las nuevas tecnologías y bases de datos.
- Ahorro de tiempo, a la hora de buscar la documentación, y de costes en su gestión y almacenamiento.
- Reducción de redundancias como duplicidad de documentos, fotocopias, grabaciones de datos.

- Agilidad y automatización de las tareas administrativas con la creación de flujos de trabajo (workflows) que permiten integrar documentos y procesos de negocio en un entorno controlado y actividades claramente definidas y monitorizadas.
- Intercambio de documentos en formato electrónico entre empresas, incluyendo el formato estándar XML y la verificación con certificados y firmas digitales.
- Para no olvidar el “efecto secundario” positivo de la mayor satisfacción de los clientes internos y externos.

Cuba no queda aislada de la necesidad de incorporar sistemas informáticos que tributen a la informatización de la sociedad. A pesar de ser un país subdesarrollado, durante los últimos años ha incorporado las TIC a los principales procesos de gestión documental de muchos sectores empresariales, con el fin de alcanzar el éxito con la calidad requerida.

La Universidad de las Ciencias Informáticas (UCI) ha sido uno de los sectores beneficiados con la aplicación de dichas tecnologías. Constituye un motor de avance y de ayuda para la informatización de la sociedad cubana. Esta institución cuenta con varios centros de desarrollo de software, entre estos el Centro de Informatización de la Gestión Documental (CIGED), dedicado al desarrollo de sistemas y servicios informáticos integrales de alta calidad y competitividad en la informatización de los procesos de gestión documental. Entre sus productos se encuentra el Gestor de Documentos Administrativos XABAL eXcriba en su versión 3.1, el cual es un software de gestión documental orientado a la automatización de los procesos documentales de las organizaciones. Contiene todos los instrumentos para cubrir desde la generación de documentos, su revisión, administración, distribución, custodia y disposición, cumpliendo con las normas internacionales para la gestión de documentos, manteniendo la seguridad de los documentos en formato electrónico.

Dicha herramienta contiene varios módulos, entre ellos el módulo denominado Gestión de acuerdos. Este módulo permite la creación, modificación, prorrogación y aplazamiento de los acuerdos que se generan en las empresas y organizaciones, así como adjuntar evidencias de cumplimiento de los mismos.

Actualmente el módulo de Gestión de acuerdo presenta algunas limitantes planteadas por los clientes que lo explotan en las diferentes instituciones que ha sido desplegado, las cuales se enuncian a continuación:

- No cuenta con un mecanismo que le permita organizar de forma estructurada y organizada todos los acuerdos creados.
- No es posible conocer las reuniones en los que esos acuerdos fueron establecidos y no cuenta con una estructura organizada para estas reuniones.

- No existe una asociación de los usuarios con los acuerdos en los que participó como Responsable o Participante.
- No es posible conocer el estado del acuerdo así como el propio contenido del acuerdo.

Teniendo en cuenta lo antes planteado, se propone como **problema a resolver**: ¿Cómo mejorar la usabilidad en la gestión documental del módulo Gestión de Acuerdos de XABAL eXcriba 3.1.?

Definiendo el **objeto de estudio** de esta investigación: El proceso de gestión de acuerdo en el sistema administrativo XABAL eXcriba 3.1.

Para dar solución al problema planteado se propone como **objetivo general**: Desarrollar un módulo de gestión de acuerdo en la tecnología de Alfresco Community 5.2, para mejorar su usabilidad y que permita la creación, seguimiento y control de acuerdos en las diferentes empresas y organizaciones. Enmarcando esta investigación en el **campo de acción**: la usabilidad en la gestión documental del módulo de Gestión de Acuerdos de XABAL eXcriba 3.1.

Las **preguntas científicas** que guían y orientan el desarrollo de esta investigación son las siguientes:

1. ¿Cuáles son los referentes teóricos y metodológicos a tener en cuenta para abordar la solución del problema planteado relacionado con el Módulo de Gestión de acuerdo en la tecnología de Alfresco Community 5.2?
2. ¿Cuál es el estado de la usabilidad en la gestión documental del módulo de Gestión de Acuerdos de XABAL eXcriba 3.1.?
3. ¿Cómo desarrollar un módulo de Gestión de acuerdo en la tecnología de Alfresco Community 5.2, que permita la creación, seguimiento y control de acuerdos en las diferentes empresas y organizaciones?
4. ¿Cómo se valida el correcto funcionamiento del Módulo de Gestión de acuerdo en la tecnología de Alfresco Community 5.2?

Para dar solución al problema planteado y dar respuesta a las preguntas científicas formuladas se proponen las siguientes **tareas de investigación**:

1. Sistematización de los referentes teóricos y metodológicos a tener en cuenta para abordar la solución del problema planteado relacionado con el Módulo Gestión de acuerdo en la tecnología de Alfresco 5.2
2. Diagnóstico del estado de la usabilidad en la gestión documental del módulo de Gestión de Acuerdos de XABAL eXcriba 3.1.

3. Desarrollo de un módulo de Gestión de acuerdo en la tecnología de Alfresco Community 5.2, que permita la creación, seguimiento y control de acuerdos en las diferentes empresas y organizaciones.
4. Identificar los diferentes tipos de pruebas de software, para su posterior aplicación sobre el módulo desarrollado

Métodos de la investigación:

Los métodos de investigación son herramientas para la recolección de datos, formular y responder preguntas para llegar a conclusiones a través de un análisis sistemático y teórico aplicado a algún campo de estudio.[CITATION Ric18 \l 3082]. Permiten descubrir en el objeto de investigación las relaciones esenciales y las cualidades fundamentales. A continuación, se describen los métodos aplicados en esta investigación:

Métodos Teóricos:

- Análítico-Sintético: para analizar y extraer de la documentación especializada los aspectos referentes a la gestión de acuerdo.
- Histórico-Lógico: para realizar un estudio cronológico y explícito de las tendencias actuales sobre los sistemas informáticos que intervienen en la gestión de acuerdo.
- Modelación: se utiliza en la modelación de los diagramas correspondientes a la etapa de análisis, diseño e implementación del módulo a desarrollar en la presente investigación.
- Sistémico-Estructural-Funcional: se utilizó para analizar cómo están relacionados el proceso de gestión de acuerdo y la gestión documental a través de Alfresco Community 5.2.

Métodos empíricos:

- Observación: para comprender la forma en que se realiza, de manera manual, el proceso de la Gestión de acuerdo. También para entender el funcionamiento del módulo de gestión de acuerdo de XABAL eXcriba 3.1.
- Entrevista: se realizaron entrevistas no estructuradas a trabajadores del Centro CIGED con el fin de obtener información acerca de la gestión de acuerdo.

El documento se encuentra estructurado como se define a continuación:

Capítulo 1. Fundamentación Teórica:

En este capítulo se realiza un estudio sobre los conceptos y aspectos más importantes para la investigación. Se realiza un análisis sobre la metodología utilizada, así como las herramientas y tecnologías necesarias para llevar a cabo el desarrollo del módulo.

Capítulo 2. Propuesta de solución:

En este capítulo se describe la arquitectura a utilizar, los requisitos que debe cumplir el módulo, los modelos y diagramas diseñados, así como sus respectivas descripciones, además se expone la arquitectura y los patrones del diseño utilizados en el desarrollo de la propuesta de solución.

Capítulo 3. Implementación y validación de la propuesta de solución:

En este capítulo se implementan el módulo de gestión de acuerdo. Se realizan pruebas internas, en pos de verificar la calidad y efectividad de acuerdo con las necesidades del cliente. Se generaron los artefactos diagrama de despliegue y casos de prueba. En la investigación se concreta como aporte práctico el perfeccionamiento de la funcionabilidad y utilización del software para la Gestión de acuerdo en la tecnología de Alfresco Community 5.2 en empresas e instituciones.

Capítulo 1: Fundamentación Teórica

En este capítulo se abordan los diferentes conceptos y definiciones que están relacionados con el dominio del problema, con el fin de lograr una mejor comprensión de la investigación. También se presenta la metodología que guía el proceso, además de los lenguajes, tecnologías y herramientas empleadas en el proceso de desarrollo del módulo.

1.1. Conceptos asociados a la investigación.

Para una mayor comprensión de la presente investigación, es importante definir algunos conceptos fundamentales vinculados con el objeto de estudio de la investigación.

Gestión

Es la capacidad de la organización para definir, alcanzar y evaluar sus propósitos, con el adecuado uso de recursos disponibles.

Según [CITATION Cla15 \l 3082], la gestión se asume como el conjunto de procesos y acciones que se ejecutan sobre uno o más recursos para el cumplimiento de la estrategia de una organización, a través de un ciclo sistémico y continuo, determinado por las funciones básicas de planificación, organización, dirección o mando y control.

Módulo de gestión

Un módulo de gestión es un conjunto de reglas y principios relacionados entre sí de forma ordenada, para contribuir a la gestión de procesos generales o específicos de una organización. Permite establecer una política, unos objetivos y alcanzar dichos objetivos[CITATION Jor19 \l 3082]. También es una estructura o modelo de administración eficaz y eficiente que busca mejorar el funcionamiento de una organización. Incluye un proceso de ideación, planeación, implementación y control.

Documento

Es todo registro de información independiente de su soporte físico. Abarca todo lo que puede transmitir el conocimiento humano. Los documentos pueden ser textuales (manuscritos, mecanografiados o impresos), iconográficos (mapas, planos, dibujos, fotografías, diapositivas, etc.), sonoros (discos, cintas magnetofónicas, discos compactos), audiovisuales (cintas de video, películas cinematográficas), electrónicos (disquetes, CD.ROM, bases de datos, aplicaciones de escritorio como procesadores de texto y planillas electrónicas, etc.), virtuales (webs, blogs, etc.), y cualquier otra propuesta que el avance tecnológico pudiera plantear en el futuro[CITATION MAR11 \l 3082].

Reunión

Proceso que desarrollan unas personas para tratar temas diversos y con finalidades diversas. Una reunión es un acto o proceso por el que un [grupo](#) de personas se unen, como un conjunto, con un propósito común. Es una técnica que permite que personas con objetivos comunes intercambien información y lleguen a compromisos colectivos para conseguir las metas[CITATION Jac17 \l 3082].

Gestión de reuniones

Es el conjunto de mecanismos de los cuales dispone el líder o conductor de una reunión con el fin de garantizar una dinámica productiva y eficaz. Las habilidades más importantes son la administración de los tiempos, el reconocimiento y reconducción del tono de la reunión, la escucha reflexiva y el uso de para generar pensamiento grupal[CITATION Kau17 \l 3082].

Acuerdo

El término acuerdo es aquel que hace referencia a la puesta en común de ideas u objetivos por parte de dos o más partes. El acuerdo siempre implica la decisión común de estas partes ya que no significa la imposición de una a la otra sino, por el contrario, el encuentro de aquello que las reúne a ambas. Un acuerdo puede ser entendido como un punto de encuentro entre las partes que toman lugar en el acto. Si bien normalmente el acuerdo se realiza entre personas, también puede llevarse a cabo entre entidades, instituciones, países o regiones que representan diversos intereses[CITATION Jac17 \l 3082].

Gestión de acuerdo

Conjunto de operaciones que le permite a una organización definir, alcanzar y evaluar los objetivos, ideas o propósitos dentro de una entidad, logrado por el consenso de las partes implicadas en el acuerdo, con el adecuado uso de los recursos disponibles. Los cuales se derivan en tareas objetivas en todos los niveles organizacionales de la institución que permiten el cumplimiento de este acuerdo[CITATION Jac17 \l 3082].

1.2. Sistemas similares

Después de una búsqueda de sistemas informáticos que gestionen información, tanto en el ámbito internacional como nacional se encontraron aplicaciones que aun cuando no se pueda generalizar su uso, sirven de referencia para comprender los procesos de gestión de acuerdo.

Características de algunos sistemas internacionales.

LogicalDOC

Es un software en de uso en navegadores de red de administración de documentos con una plataforma de fácil uso y un diseño novedoso e intuitivo, capacidad de administración de agenda, y varias opciones para subir archivos, organizar y distribuir documentos[CITATION Gas15 \l 1033].

Registro de auditoría

- LogicalDOC mantiene un registro detallado de todos los cambios hechos a un documento o folder en específico, guardando toda esta información en una base de datos en la nube. Esto quiere decir que se puede revisar todo el historial de ingreso de datos en cualquier documento, para poder observar los cambios que se han hecho y quien hizo estos cambios.
- Gracias a este registro de auditoría meticuloso, se puede hacer búsquedas por los cambios hechos a un documento o folder en específico, puede buscar por el usuario que haya hecho los cambios o por la fecha de cuando se hicieron los cambios. Este registro de documentos intuitivo le permitirá evitar perder datos y poder seguirle la pista al progreso de cada documento o proyecto.
- Los usuarios involucrados en algún proyecto en específico recibirán un aviso de actualización por correo y en sus correos personales. Esta actualización le permitirá mantenerse al corriente con los cambios hechos a varios documentos y folders en tiempo real, reduciendo el estrés y el consumo de tiempo que lleva el perder información integral debido a errores, sean humanos o tecnológicos.

Administración de flujo de trabajo

- LogicalDOC incluye un sistema de administración de flujo de trabajo que le permitirá administrar y distribuir un sistema de tareas a otros usuarios. Esto permitirá a usted visualizar tareas personales al momento de procesar los documentos. Ya que haya creado el flujo de trabajo, los usuarios a los que le haya tocado hacer una tarea serán alertados por correo cada vez que requieran completar una tarea o solo para llevar un registro de la actividad relacionada a un documento en específico.

GOC

GOC es una herramienta de software libre, diseñada para la gestión eficiente de reuniones de trabajo de entidades, empresas y órganos colegiados. Su principal objetivo es facilitar al máximo las reuniones de equipo en aquellos entornos que, por motivos profesionales, este tipo de encuentros sean frecuentes[CITATION 4ti19 \l 23562].

Entre las principales funcionalidades de GOC se encuentra:

- Permite el acceso a toda la información relativa a la reunión desde cualquier ordenador, lo que evita tener que enviar copias y notificaciones a las personas convocadas a la misma.

- Gestión de la fecha, los integrantes o del orden del día, el envío de notificaciones a los asistentes, la reproducción del acta de manera automática.

¿Cómo funciona?

1. El primer punto es dar de alta o integrar tanto los órganos a convocar como los miembros de éstos. Una vez dados de alta, se pueden asignar “Autorizados”, que serán las personas que tendrán acceso a poder realizar las convocatorias de los órganos seleccionados. El administrador de la plataforma es quién asigna estos permisos.
2. El autorizado crea la reunión, añade el orden del día y notifica a los asistentes, que automáticamente reciben un correo electrónico con los datos de la convocatoria.
3. Estos pueden confirmar la asistencia, indicar un sustituto y añadir comentarios al orden del día recibido.
4. Una vez realizada la reunión, se generan los acuerdos y deliberaciones y se incluyen en el acta, la cual puede firmarse electrónicamente y archivarse para posteriores consultas.

M-Files

M-Files es una plataforma de CMS inteligente diseñada para ayudar a las empresas a administrar, organizar y rastrear documentos e información. Puede ser usado tanto por compañías grandes como pequeñas en una variedad de industrias, desde manufactura, servicios financieros hasta ciencias de la vida. Esta amplia aplicabilidad es posible gracias a una gran cantidad de funciones destinadas a beneficiar a cualquier organización[CITATION Seb19 \l 23562]:

- Está equipado con un motor de búsqueda para encontrar documentos según el tipo de documento, los nombres y las palabras clave, entre otros.
- M-Files también ofrece una funcionalidad de gestión de información de nivel empresarial, lo que le permite compartir información de forma segura con las partes interesadas o los miembros del equipo.
- Una variedad de herramientas de gestión. Cualquiera sea su necesidad (administración de documentos, administración de contenido, integraciones y control de versiones), M-Files puede ayudar a crear cientos o incluso miles de archivos todos los días sin preocuparse por su recuperación o almacenamiento.
- Repositorio central de información. El sistema actúa como un almacenamiento central para todos los documentos, lo que le permite buscar y encontrar documentos, identificar quién los editó por

última vez y organizar todos sus documentos sin necesidad de recordar ubicaciones y versiones guardadas.

Características de algunos sistemas nacionales.

Sistema de Gestión de información del Departamento de Ingeniería Informática de Sancti Spiritus (SIGIDI).

SIGIDI contempla tres roles para los usuarios: Jefe de Departamento, Jefe de Carrera y Administrador del Sistema; el acceso se controla mediante un nombre de usuario y una contraseña, garantizando así la seguridad de la información. A continuación algunas funcionalidades del sistema[CITATION los17 \l 3082]:

- Gestionar trabajadores del departamento: permite adicionar, eliminar o modificar los siguientes datos: nombre, primer y/o segundo apellidos, categoría docente, número de carné de identidad y tipo de trabajador (docente o no docente), así como la dirección particular, título académico, número de teléfono y año en que comenzó a trabajar. Esta opción también permite listar y buscar los trabajadores por algunos de los criterios anteriores.
- Gestionar publicación permite adicionar, eliminar, modificar y listar las publicaciones realizadas por los trabajadores docentes del departamento, de las que interesa el año en que se realizó y si fue en una revista o en un libro.
- Gestionar AFT posibilita adicionar, eliminar, modificar y listar los Activos Fijos Tangibles del departamento, aspecto que ayuda a mantener un mejor control.
- Gestionar las reuniones es la última de las funcionalidades que brinda SIGIDI al Jefe de Departamento, con el objetivo de dejar constancia de ellas. En este caso se puede agregar, modificar, eliminar y listar datos relacionados con la fecha y hora de realización, la orden del día, la asistencia y los acuerdos que se tomaron.
- Gestionar reuniones de la carrera tiene el objetivo de dejar constancia de ellas. En este caso se pueden agregar, modificar, eliminar y listar datos como la fecha y hora de realización, la orden del día, la asistencia y los acuerdos que se tomaron.

GDA XABAL excriba 3.1

Software para la gestión documental, diseñado para tramitar los documentos administrativos que se generan o reciben dentro de las organizaciones a partir de sus funciones, por lo tanto, involucra todas las áreas de una organización, permitiéndoles gestionar de forma correcta la documentación como prueba, testimonio y evidencia de las actividades organizacionales.

Contiene módulos para la gestión documental, los cuales son:

- Gestión de acuerdos
- Gestión de reportes
- Gestión de registros de entrada y salida de documentos
- Gestión del cuadro de clasificación de la entidad

El módulo de gestión de acuerdos de XABAL eXcriba permite realizar las acciones básicas sobre un acuerdo, como crear y modificar las propiedades de los acuerdos. Pero no contiene las suficientes herramientas para controlar el acuerdo en todo su período de vida.

1.2.1. Resultados del estudio

Después del análisis realizado a diferentes sistemas de gestión de información, se pudo conocer el funcionamiento de dichos sistemas, así como la esencia de sus procesos de gestión documental y tenerlos en cuenta en el desarrollo de esta investigación. Además, se toman las siguientes características relevantes a tener presente en el módulo de gestión acuerdo a desarrollar:

- El control y la gestión de acuerdos y reuniones.
- Organización y estructuración de todos los acuerdos.

1.3. Metodologías de desarrollo de software.

Es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información. En un proyecto de desarrollo de software la metodología ayuda a definir: ¿Quién debe hacer Qué, ¿Cuándo y Cómo debe hacerlo? La metodología para el desarrollo de Software es un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito. Una metodología para el desarrollo de software comprende actividades a seguir para idear, implementar y mantener un producto de software desde que surge la necesidad del producto hasta que se cumple el objetivo por el cual fue creado[CITATION Enr17 \l 3082].

Variante AUP-UCI.

Para el desarrollo de esta investigación se determinó emplear la metodología ágil, Proceso Unificado Ágil para el Desarrollo de la actividad productiva en la Universidad de las Ciencias Informáticas (AUP-UCI). Dicha metodología fue estandarizada en la UCI, para guiar el proceso de desarrollo de software en sus distintos centros productivos.

(AUP-UCI) es una versión simplificada del Proceso Unificado de Software (RUP). Describe de una forma fácil de entender una aproximación al desarrollo de software empresarial mediante técnicas ágiles manteniéndose fiel a las técnicas y conceptos definidos por RUP. Las técnicas ágiles empleadas por AUP incluyen desarrollo dirigido por pruebas, modelado ágil, gestión de cambios ágil y refactorización de base de datos para mejorar la productividad. La metodología AUP-UCI responde a una variación que se realiza a la metodología ágil Proceso Unificado Ágil (AUP), de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la UCI.

Descripción de las fases que presenta AUP-UCI

AUP-UCI divide el ciclo de vida de la producción del software en tres fases:

Inicio: fase donde se realiza un estudio inicial de la organización cliente que permite obtener una información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.

Ejecución: fase donde se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto.

Cierre: fase donde se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto (Rodríguez Sánchez, 2015).

Dentro de la fase de Ejecución, dicha metodología propone 7 disciplinas para el ciclo de vida de los proyectos de la UCI, donde sus flujos de trabajos son: *Modelado de negocio, Requisitos, Análisis y diseño, Implementación, Pruebas Internas, Pruebas de Liberación y Pruebas de Aceptación.*

Escenarios para la disciplina requisitos que presenta AUP-UCI Partiendo que la disciplina Modelado de negocio propone tres variantes a utilizar en los proyectos (casos de uso del negocio, descripción de proceso de negocio y modelo conceptual) y existen tres formas de encapsular los requisitos (casos de uso del sistema, historias de usuario y descripción de requisitos por proceso), surgen cuatro escenarios para modelar el sistema en los proyectos estos son:

Escenario No 1: Proyectos que modelen el negocio con casos de uso del negocio solo pueden modelar el sistema con casos de uso del sistema.

Escenario No 2: Proyectos que modelen el negocio con modelo conceptual solo pueden modelar el sistema con casos de uso del sistema.

Escenario No 3: Proyectos que modelen el negocio con descripción de procesos de negocio solo pueden modelar el sistema con descripción de requisitos por procesos.

Escenario No 4: Proyectos que no modelen negocio solo pueden modelar el sistema con historias de usuario (Rodríguez Sánchez, 2015).

Para llevar a cabo el desarrollo del módulo se decide utilizar el escenario 2, ya que una vez que fue evaluado el negocio a informatizar se obtuvo que no fue necesario incluir las responsabilidades de las personas que ejecutan las actividades. Además, se comprobó que el negocio no está bien definido y el cliente no acompaña al equipo de desarrollo, permitiendo así, modelar exclusivamente los conceptos fundamentales del negocio. Este escenario se recomienda para proyectos donde el objetivo primario es la gestión y presentación de la información.

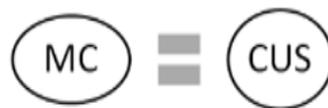


Figura 1:Escenario 2:Metodología AUP-UCI.

1.4 Ambiente de desarrollo.

Alfresco Community 5.2

Según [CITATION Elv10 \l 3082] Alfresco es un sistema gestor documental con enfoque empresarial (ECM) que proporciona, entre otras funcionalidades:

- Un repositorio de contenido con versionado que permite la edición y compartición de documentos.
- Entorno de trabajo colaborativo
- Organización de los documentos por carpetas y categorías
- Control de flujo de trabajo
- Gestión de información
- Gestión de contenido web

Es uno de los softwares open source (código abierto) que compiten con las más grandes soluciones propietarias de gestión documental y contenidos. Es un sistema de gestión de contenidos de código abierto que permite a las organizaciones capturar, almacenar, buscar y colaborar en documentos de

muchos tipos distintos. También se ocupa de la gestión de contenido web, records management, trabajo colaborativo, flujos de trabajo.

Visual Paradigm 8.0 for UML

Herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite modelar todos los tipos de diagramas de clases, generar código desde diagramas y generar documentación. La herramienta agiliza la construcción de aplicaciones con calidad y a un menor coste. Posibilita la generación de bases de datos, transformación de diagramas de Entidad-Relación en tablas de base de datos, así como obtener ingeniería inversa de bases de datos.

Axure RP Pro 7.0

Permite a diseñadores crear diagramas de flujo, prototipos, y especificaciones para aplicaciones y sitios web más rápido y fácil que creando prototipos estáticos con herramientas corrientes. Axure es una herramienta pensada especialmente para los arquitectos de la información. Posibilita la creación de prototipos web de forma profesional[CITATION Leó19 \l 3082].

PostgreSQL v9.4

es un servidor de base de datos objeto relacional libre, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional, liberado bajo la licencia BSD[CITATION Ant14 \l 3082].

Las características positivas que posee este gestor son[CITATION Ant14 \l 3082]:

1. Posee una gran escalabilidad. Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta (en algunos benchmarks se dice que ha llegado a soportar el triple de carga de lo que soporta MySQL).
2. Implementa el uso de rollback's, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz, y ofreciendo soluciones en campos en las que MySQL no podría.
3. Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle.

JQuery

jQuery es una librería de JavaScript de código abierto que sirve como base para la programación de aplicaciones para la web. Brinda un conjunto de funciones y objetos que permiten abstraerse de uno de los principales problemas a los que se enfrenta el desarrollo de una aplicación web: la incompatibilidad entre navegadores. De manera general, también soluciona o da soporte a problemas frecuentes en el desarrollo de aplicaciones web actuales, entre ellos (jQuery, 2012).

AJAX

Permite mejorar la interacción del usuario con la aplicación, evitando las recargas constantes de la página, ya que el intercambio de información con el servidor se produce en un segundo plano. Las aplicaciones construidas con AJAX eliminan esa problemática, mediante la creación de un elemento intermedio entre el usuario y el servidor. La nueva capa intermedia de AJAX mejora la respuesta de la aplicación, ya que el usuario nunca se encuentra con una ventana del navegador vacía esperando la respuesta del servidor[CITATION Egu17 \l 3082].

WebScript

Los WebScripts son elementos clave dentro de Alfresco y de uso obligado para cualquier desarrollador. Nos permiten construir servicios web para la gestión de contenidos accesibles vía HTTP, es decir, convierten el repositorio de Alfresco en un gestor de contenido controlador mediante un servidor HTTP. Se crean de forma muy rápida, usan JavaScript y [Freemarker](#) para ofrecen máxima flexibilidad y velocidad de despliegue. Pasamos a describir a continuación qué son y cómo crearlos[CITATION Yin17 \l 3082].

FreeMaker

Es un “motor de la plantilla”; una herramienta genérica para generar la salida del texto basada en plantillas. FreeMarker, es un paquete de Java, una biblioteca de la clase para los programadores de Java. Es un motor de plantillas, así como una herramienta genérica para generar la salida de texto basado en plantillas. No es una aplicación para los usuarios finales en sí mismo, sino un paquete de Java que los programadores pueden utilizar en sus productos[CITATION Tor15 \l 3082].

Sublime Text 3

Sublime 3 es un editor de código muy rápido y sofisticado que nos permitirá ejecutar un montón de tareas de forma muy rápida y sencilla. Soporte muchos lenguajes de programación y dispone de un sistema de instalación de paquetes adiciones que amplían sus características de forma ilimitada.

Según[CITATION Der18 \l 3082] Sublime Text 3 (ST3) es un editor de código liviano y multiplataforma, conocido por su velocidad, facilidad de uso y un sólido soporte comunitario. Es un editor increíble de

inmediato, pero el verdadero poder proviene de la capacidad de mejorar su funcionalidad mediante el Control de Paquetes y la creación de configuraciones personalizadas.

Visual Studio Code

Visual studio code (vs code) es un editor de texto gratuito que soporta diversos lenguajes de programación, es compatible para MAC, Windows y Linux. Cuenta con muchas características como Intellisense ,Git, snippets, y es customizable por lo que sus usuarios pueden cambiar las preferencias,shortcuts y temas a su gusto[CITATION Val17 \l 3082].

JavaScript 1.6

JavaScript es un lenguaje de los denominados lenguajes de scripting de programación en un sentido estricto, pero sin embargo sí nos permite crear páginas dinámicas, con algunos efectos realmente interesantes y que mejoren considerablemente su aspecto. Nos permite tener cierta interacción con el usuario de nuestras páginas, reconocer determinados eventos que se puedan producir y responder a éstos adecuadamente.[CITATION Alb16 \l 3082].

HTML 5 (HyperText Markup Language)

Se trata de una nueva versión de HTML, con nuevos elementos, atributos y comportamientos. Contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance. Propone estándares para cada aspecto de la web y también un propósito claro para cada una de las tecnologías involucradas. A partir de ahora, HTML provee los elementos estructurales[CITATION Gau12 \l 3082].

Apache Tomcat 7.0

Apache Tomcat es una herramienta de servidor web de código abierto desarrollada por Apache Software Foundation (ASF). Es uno de los muchos productos de código abierto relacionados con Apache que utilizan los profesionales de TI para diversas tareas y objetivos. Es un software desarrollado con Java (con lo cual puede funcionar en cualquier sistema operativo, con su máquina virtual java correspondiente) que sirve como servidor web con soporte de servlets y [JSPs](#)[CITATION Día16 \l 3082].

XML 1.0

XML no es más que un modelo de objetos en forma de API que permite acceder a las diferentes partes que pueden componer un documento XML o HTML, es un lenguaje que permite la organización y el etiquetado de **documentos**. Esto quiere decir que el XML no es un lenguaje en sí mismo, sino un sistema que permite definir lenguajes de acuerdo a las necesidades [CITATION Pet98 \l 3082].

1.5. Conclusiones del capítulo.

En este capítulo se abordaron los conceptos fundamentales asociados al dominio del problema, relacionados con el objeto de estudio y el campo de acción. El estudio realizado a la metodología, tecnologías, herramientas y lenguajes, proporcionó los conocimientos necesarios para su utilización en el desarrollo de la solución propuesta.

Capítulo 2: Propuesta de solución.

En este capítulo se abordan las principales características y se evidencia el diseño del módulo a desarrollar. Se hace un levantamiento de los requisitos funcionales y no funcionales necesarios para lograr el funcionamiento correcto del módulo. Además, se construyen los artefactos correspondientes al análisis y diseño acorde con la metodología seleccionada.

2.1 Modelo de Dominio

Un modelo de dominio es una representación de las clases conceptuales del mundo real, no de componentes software. No se trata de un conjunto de diagramas que describen clases software, u objetos software con responsabilidades (Larman, 2002). En él se describen las distintas entidades, sus atributos y relaciones, además de las restricciones que rigen el dominio del problema.

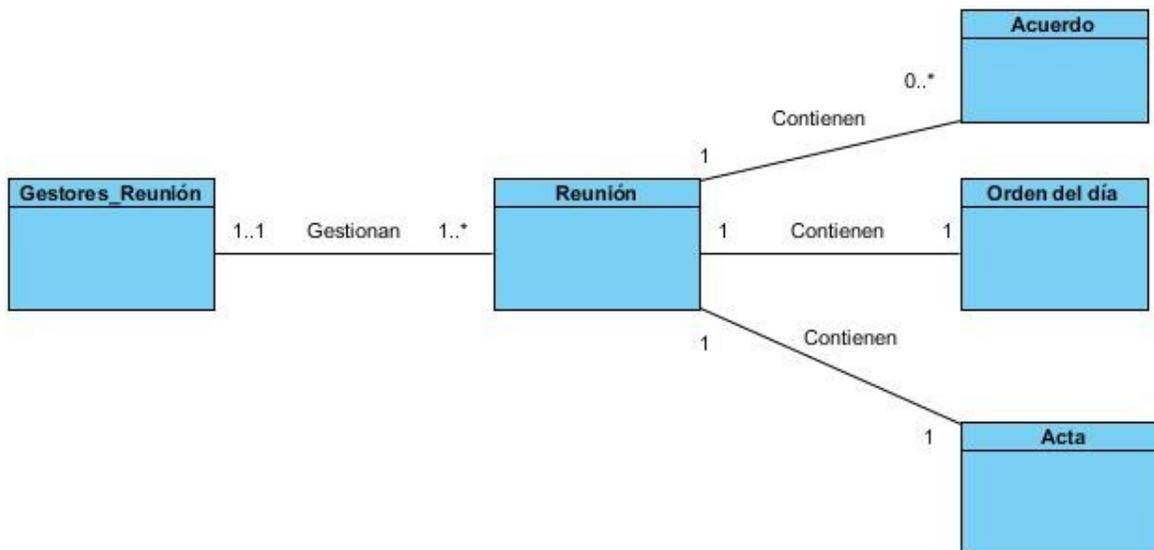


Figura 2: Modelo de dominio. Fuente: Elaboración propia

Descripción de las clases del dominio.

Gestores_Reunión: interviene en el proceso generando las reuniones.

Reunión: representa el acto o proceso por el que un grupo de personas se unen, como un conjunto, con un propósito común.

Actas: escrito que contiene lo sucedido o tratado en una reunión, este contiene acuerdos.

Acuerdo: decisión que es tomada tras una negociación o debate entre las partes involucradas.

Orden del día: es una lista de puntos a ser discutidos en una determinada [reunión](#).

2.2 Especificación de los requisitos de software

En el proceso de ingeniería de software, los requisitos se utilizan como datos de entrada en la etapa de diseño del producto que se desea lograr. Estos representan qué debe hacer el sistema, pero no cómo hacerlo y, además, incorporan las cualidades o propiedades que el mismo debe poseer[CITATION Som07 \l 3082].

2.2.1 Requisitos funcionales

Los requisitos funcionales (RF) definen el comportamiento interno de un software, son condiciones que el sistema ha de cumplir. Estos muestran las funcionalidades que deben satisfacerse para cumplir con las especificaciones de software[CITATION SOM08 \l 3082].

En el análisis para el desarrollo de la solución propuesta, fueron obtenidos 45 requisitos funcionales descritos a continuación:

Tabla 1: Requisitos funcionales

Nº	Requisitos Funcionales	Descripción del requisito.
RF1.	Crear reunión	El sistema debe permitir al grupo GESTORES_REUNION crear una reunión. Para ello debe insertar los siguientes datos: Tipo de reunión, Sesión, Motivo, Fecha, Responsable y Participantes.
RF2.	Editar propiedades de reunión	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador de la reunión editar las propiedades de la misma. Se podrán modificar los siguientes metadatos: Sesión, Motivo, Responsable y Participantes.
RF3.	Ver detalles de reunión	El sistema debe permitir a los creadores, responsable y participantes de la reunión visualizar las propiedades de la misma. Se mostrarán los siguientes datos: Creador, Tipo de reunión, Estado, Sesión, Motivo, Fecha, Responsable, Participantes, Índice acuerdos cumplidos y un vínculo al Acta y al Orden del día. Además, si el acuerdo tiene estado Cancelado se muestra el motivo.
RF4.	Cancelar reunión	El sistema debe permitir al responsable y a todos los usuarios que pertenezcan al grupo del creador de la reunión cancelar una reunión cuyo estado sea: Planificada o Replanificada. Para ello debe insertarse el motivo. Al realizar la acción se modifica el estado de la misma a Cancelada y se notifica mediante un correo electrónico la cancelación a todos los involucrados.

RF5.	Replanificar reunión	El sistema debe permitir al responsable y a todos los usuarios que pertenezcan al grupo del creador de la reunión replanificar la misma. Para ello debe insertarse la nueva fecha. Al realizar la acción se modifica el estado de la misma a Replanificada y se notifica mediante un correo electrónico la replanificación a todos los involucrados. Solo se replanifica aquellas reuniones que no tengan estado cancelado, y no están en estado ejecutada.
RF6.	Calcular índice de acuerdos cumplidos por reunión	El sistema debe calcular automáticamente el índice de acuerdos cumplidos de una reunión cuando se crea o se cierra un acuerdo de la misma. Se calculará de los acuerdos adoptados la cantidad según su estado general, así como el por ciento que representan. Así mismo se calculará de forma general el índice de acuerdos cumplidos IAC (%) = $\frac{TAC}{TA} \times 100$ ¹ .
RF7.	Visualizar reuniones asignadas a mí	El sistema debe permitirle al usuario autenticado visualizar todas las reuniones en las que es responsable. De las mismas se mostrarán los siguientes datos: Nombre, Tipo de reunión, Estado, Motivo, Responsable, Participantes y el IAC.
RF8.	Visualizar reuniones creadas por mí	El sistema debe permitirle al usuario autenticado visualizar todas las reuniones creadas por el grupo de gestores de reunión al cual pertenece. De las mismas se mostrarán los siguientes datos: Nombre, Tipo de reunión, Estado, Motivo, Responsable, Participantes y el IAC.
RF9.	Buscar reuniones	El sistema debe permitir al usuario buscar reuniones a partir de un criterio insertado. Solo se visualizarán los elementos en los que coincida parcial o totalmente el criterio insertado con el valor de los metadatos visualizados en la tabla.
RF10.	Filtrar reuniones por nombre	El sistema debe permitir al usuario filtrar las reuniones por el nombre. Se visualizarán los elementos que estén contenidos en el rango de fecha especificado.
RF11.	Filtrar reuniones por fecha	El sistema debe permitir al usuario filtrar las reuniones por la fecha. Se visualizarán los elementos que estén contenidos en el rango de fecha especificado.
RF12.	Filtrar reunión por tipo de reunión	El sistema debe permitir al usuario filtrar las reuniones por el campo tipo de reunión. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF13.	Filtrar reunión por estado	El sistema debe permitir al usuario filtrar las reuniones por el

¹TAC: Total de acuerdos cumplidos en el período; TA: Total de acuerdos a cumplir en el período.

		campo estado. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF14.	Filtrar reunión por motivo	El sistema debe permitir al usuario filtrar las reuniones por el campo motivo. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF15.	Filtrar reunión por responsable	El sistema debe permitir al usuario filtrar las reuniones por el campo responsable. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF16.	Filtrar reunión por participantes	El sistema debe permitir al usuario filtrar las reuniones por el campo participantes. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF17.	Filtrar reunión por índice de cumplimiento de acuerdos	El sistema debe permitir al usuario filtrar las reuniones por el campo índice de cumplimiento de acuerdos. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF18.	Crear acuerdo	El sistema debe permitir al grupo GESTORES_REUNION crear un acuerdo. Se registrarán los siguientes campos genéricos: Contenido, Prioridad, Responsables, Participantes, Fecha de emisión y Fecha de cumplimiento inicial.
RF19.	Ver detalles de acuerdo	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo visualizar las propiedades del mismo. Se mostrarán los siguientes datos: Estado de cumplimiento, Fecha de cumplimiento, Responsables, Participantes, Contenido.
RF20.	Editar propiedades de acuerdo	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo editar las propiedades de la misma si tiene estado No comenzado. Se podrán modificar los siguientes metadatos: Contenido, Prioridad, Responsables y Participantes.
RF21.	Eliminar acuerdo	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo eliminar un acuerdo si este tiene estado No comenzado.
RF22.	Actualizar estado de acuerdo	El sistema debe permitir a los responsables del acuerdo actualizar el estado del mismo. Para ello debe insertar los siguientes metadatos: Estado y Porcentaje.
RF23.	Aplazar acuerdo	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo aplazar un acuerdo. Para ello debe insertar la nueva fecha de cumplimiento y el motivo. Una vez ejecutada la acción se modifica la fecha de cumplimiento del acuerdo y notificado el responsable del mismo.
RF24.	Detener acuerdo	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo detener un acuerdo siempre que

		este tenga estado diferente de: Cumplido, Completado, Cancelado o Incumplido. Para ello debe insertar un Motivo. Una vez ejecutada la acción el estado pasa a Detenido.
RF25.	Reanudar acuerdo	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo reanudar un acuerdo siempre que este tenga estado: Detenido. Para ello debe insertar la nueva fecha de cumplimiento. Una vez ejecutada la acción el estado pasa a En proceso y es notificado el responsable del mismo.
RF26.	Cancelar acuerdo	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo cancelar el acuerdo. Para ello se requiere el campo Observaciones. Se le asignará el estado Cancelado al acuerdo y se cerrará.
RF27.	Cerrar acuerdo	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo cerrar un acuerdo siempre que este tenga estado: Completado. Para ello debe insertar el estado y las Observaciones. Una vez ejecutada la acción si el estado es Cumplido el Porcentaje será igual a 100, en caso contrario si el estado es Incumplido el Porcentaje no se modificará.
RF28.	Visualizar acuerdos creados por mí	El sistema debe permitirle al usuario autenticado visualizar todos los acuerdos creados por el grupo de gestores de reunión al cual pertenece y cuya fecha de emisión sea del año actual. Se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.
RF29.	Visualizar acuerdos asignados a mí	El sistema debe permitirle al usuario autenticado visualizar todos los acuerdos de los cuales es el responsable, el estado no sea: Cancelado, Cumplido o Incumplido y cuya fecha de emisión sea del año actual. Se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.
RF30.	Visualizar acuerdos en los que participo	El sistema debe permitirle al usuario autenticado visualizar todos los acuerdos de los cuales es participante, el estado no sea: Cancelado, Cumplido o Incumplido y cuya fecha de emisión sea del año actual. Se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.
RF31.	Visualizar acuerdos aplazados	El sistema debe permitirle a los creadores, responsables y participantes visualizar todos los acuerdos que tienen aspecto aplazado cuya fecha de cumplimiento sea del mes actual. Se

		visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.
RF32.	Visualizar acuerdos por expirar	El sistema debe permitirle a los creadores, responsables y participantes visualizar todos los acuerdos cuya fecha de cumplimiento estén en la semana actual. Se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.
RF33.	Visualizar acuerdos pendientes de revisión	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador del acuerdo visualizar todos los acuerdos que su estado sea Completado. Se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.
RF34.	Visualizar acuerdos expirados	El sistema debe permitirle a los creadores, responsables y participantes visualizar todos los acuerdos cuya fecha de cumplimiento haya sido expirada. Se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.
RF35.	Visualizar acuerdos pendientes	El sistema debe permitirle a los creadores, responsables y participantes visualizar todos los acuerdos pendientes. Se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.
RF36.	Filtrar acuerdo por nombre de acuerdo	El sistema debe permitir a los creadores, responsables y participantes filtrar los acuerdos por el campo Acuerdo. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF37.	Filtrar acuerdo por contenido	El sistema debe permitir a los creadores, responsables y participantes filtrar los acuerdos por el campo Contenido. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF38.	Filtrar acuerdo por fecha de emisión	El sistema debe permitir a los creadores, responsables y participantes filtrar los acuerdos por el campo fecha de emisión. Se visualizarán los elementos que estén contenidos en el rango especificado.
RF39.	Filtrar acuerdo por fecha de cumplimiento	El sistema debe permitir a los creadores, responsables y participantes filtrar los acuerdos por el campo fecha de cumplimiento. Se visualizarán los elementos que estén contenidos en el rango especificado.
RF40.	Filtrar acuerdo por estado	El sistema debe permitir a los creadores, responsables y

		participantes filtrar los acuerdos por el campo Estado. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF41.	Filtrar acuerdo por responsables	El sistema debe permitir a los creadores, responsables y participantes filtrar los acuerdos por el campo Responsables. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF42.	Filtrar acuerdo por reunión	El sistema debe permitir a los creadores, responsables y participantes filtrar los acuerdos por el campo Reunión. Se visualizarán los elementos que coincidan parcial o totalmente con el criterio de búsqueda.
RF43.	Buscar acuerdos	El sistema debe permitir al usuario buscar acuerdos a partir de un criterio insertado. Solo se visualizarán los elementos en los que coincida parcial o totalmente el criterio insertado con el valor de los metadatos visualizados en la tabla.
RF44.	Crear documento Orden del día	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador de la reunión crear el documento Orden del día. Al realizar la acción el sistema debe crear a partir de una plantilla dicho documento, asociarlo a la reunión en cuestión y establecer permiso de Consumidor para todos los involucrados en la reunión.
RF45.	Crear documento Acta de reunión	El sistema debe permitir a todos los usuarios que pertenezcan al grupo del creador de la reunión crear el documento Acta de reunión. Al realizar la acción el sistema debe crear a partir de una plantilla dicho documento, asociarlo a la reunión en cuestión y establecer permiso de Consumidor para todos los involucrados en la reunión.

2.2.2. Requisitos no funcionales.

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable[CITATION Som05 \l 3082].

A continuación, se muestran los requisitos no funcionales seleccionados para el módulo gestión de acuerdos:

Usabilidad

- **Comprensibilidad:** Por cada mensaje de error del sistema debe existir una correspondencia entre el fallo real ocurrido y el texto del mensaje mostrado.
- **Cognoscibilidad:** Los mensajes mostrados acerca de los errores cometidos por el usuario serán notificados y a su vez incluirán sugerencias de posibles soluciones.
- **Operabilidad:** El sistema pide confirmación al usuario antes de ejecutar cualquier acción.

El 100% de los escenarios del módulo contendrán criterios específicos para realizar búsquedas.

Confiabilidad

- **Madurez:** En el 90% de los formularios, el sistema impone campos obligatorios para garantizar la integridad de la información que es introducida por el usuario.

Software

- El sistema debe integrarse con el Gestor de base de datos PostgreSQL 9.4.
- Debe tener como contenedor web Apache Tomcat 7.0
- La estación de trabajo cliente debe tener instalado un navegador web. Se recomienda utilizar Mozilla Firefox 57.X o superior.

Hardware

- El hardware donde se instalará el sistema debe poseer al menos una interfaz de red cuya velocidad de transferencia iguale o supere los 100 Mbps.
- El Servidor donde se instale el sistema debe tener al menos un Microprocesador intel core i3 2.50 GHz o superior, mínimo 4 GB de memoria RAM y un espacio libre en Disco Duro de 250 GB.
- Para la instalación de la aplicación se debe disponer de una computadora de 4GB de RAM o superior, un microprocesador core i3 2.50 GHz o superior y una interfaz de red.

2.3. Definición de los casos de usos del sistema

Una vez conocidos los requisitos funcionales y no funcionales se definen los casos de uso del sistema (CUS) donde se especifican los detalles del flujo de eventos a ejecutar, así como el actor que lo inicializa. Las mismas establecen una guía para el desarrollador.

2.3.1. Definición de los actores

Un actor es un conjunto coherente de roles que los usuarios de casos de uso desempeñan cuando interactúan con estos casos de uso.[CITATION Jac042 \l 3082]

Tabla 2 Descripción de los actores

Actores	Descripción
Usuario	Puede buscar e imprimir acuerdos y reuniones, ver detalles de una reunión, filtrar acuerdos y reuniones por propiedades. Puede visualizar acuerdos prorrogados, aplazados y por expirar.
Planificador	Encargado de gestionar acuerdos. Puede cerrar, detener, aplazar y reanudar un acuerdo. Visualiza los acuerdos pendientes a prórroga, creados por él y pendientes de revisión. Puede replanificar un acuerdo y cancelar el mismo. Aprueba solicitud de prórroga.
Gestor de reuniones	Es el encargado de gestionar las reuniones, generar el documento orden del día y acta de reunión y visualizar las reuniones creadas por él.
Responsable de reunión	Visualiza las reuniones que son asignadas a él.
Responsable de acuerdo	Solicita prórroga, cancela solicitud de prórroga, puede actualizar el estado del acuerdo y visualizar los acuerdo asignados a él.
Participante de acuerdo	Visualiza los acuerdos en los que participa

2.3.2. Diagrama de caso de uso del sistema

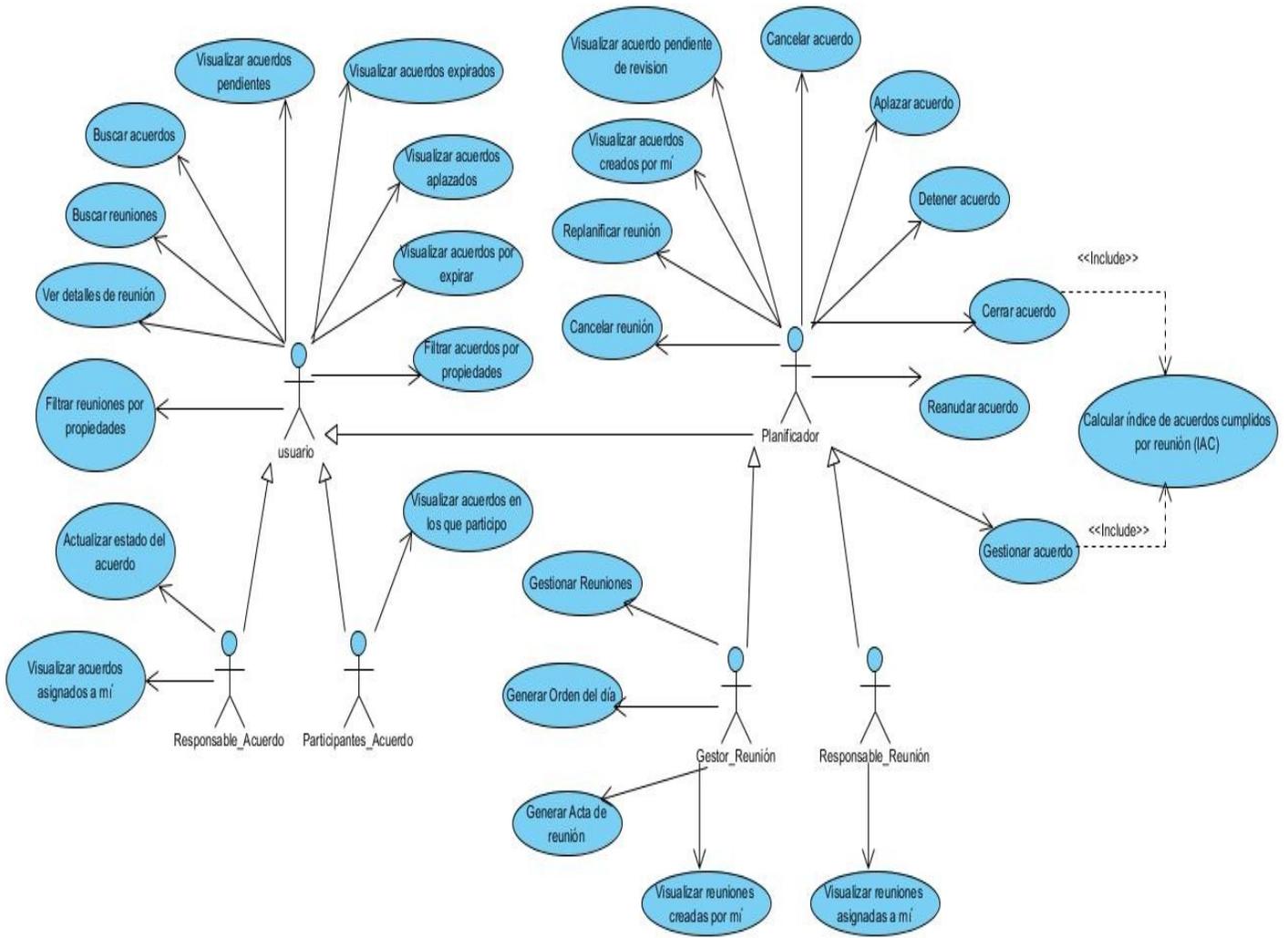


Figura 3: Diagrama de casos de uso

2.3.3 Descripción de casos de uso del sistema.

A continuación, se expone la descripción de los casos de uso Gestionar reunión y Gestionar acuerdos. El resto de los casos de uso se pueden consultar en el Anexo Casos de uso del sistema.

Tabla 3: CU_Gestionar reunión.

Caso de Uso	Gestionar reunión	
Objetivo	Permite al usuario crear y editar propiedades de una reunión.	
Actores	GESTOR_REUNION	
Resumen	El caso de uso inicia cuando el actor GESTOR_REUNION desea: Crear reunión o editar propiedades de una reunión. El caso de uso termina cuando realiza alguna de estas acciones.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.	
Post- condiciones	Se almacena la reunión creada, se actualizan las propiedades de una reunión.	
Flujo de eventos		
Flujo básico: Gestionar reunión.		
	Actor	Sistema
1.	Crea y edita propiedades de una reunión.	
2.		Permite realizar varias acciones con la reunión: <ul style="list-style-type: none"> - Crear reunión. Ver Sección 1: Crear Nueva reunión. - Editar propiedades de reunión: Ver Sección 2: Editar propiedades de reunión.
3.		Termina el caso de uso.
Sección1: “Crear Nueva reunión”.		
1.	Selecciona la opción que le permite crear una “Nueva reunión”.	
2.		Muestra un formulario con los siguientes datos a insertar: <ul style="list-style-type: none"> - Sesión - Fecha - Responsable - Participantes. Y permite: <ul style="list-style-type: none"> - Crear

		- Cancelar
3.	Inserta los valores en el formulario.	
4.	Selecciona el botón de "Crear".	
5.		Valida los datos insertados
6.		Crea la nueva reunión con los datos insertados
7.		Termina el caso de uso
Flujos alternos		
4.a Selecciona la opción que le permite "Cancelar".		
	Actor	Sistema
1.	Selecciona la opción "Cancelar"	
2.		Cancela la acción y cierra el formulario de Crear reunión.
3.		Termina el caso de uso.
Flujos alternos		
5.a Existen campos vacíos.		
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 de la Sección 1: Crear reunión.
Flujos alternos		
5.b Existen campos incorrectos		
	Actor	Sistema
1.		Muestra un mensaje: "El valor introducido en el campo es incorrecto".
2.		Regresa al paso 3 de la Sección 1: Crear reunión.
Sección 3: "Editar propiedades de reunión".		
1.	Selecciona la acción editar propiedades de reunión.	
2.		Muestra un formulario con los siguientes datos: <ul style="list-style-type: none"> - Sesión - Motivo - Responsable - Participantes. Y permite: <ul style="list-style-type: none"> - Actualizar

		- Cancelar
3.	Modifica los datos que estime conveniente.	
4.	Presiona el botón "Actualizar".	
5.		Valida los datos insertados
6.		Modifica las propiedades de la reunión con los datos insertados.
7.		Termina el caso de uso.
Flujos alternos		
4.a Presiona el botón "Cancelar"		
	Actor	Sistema
1.	Presiona el botón "Cancelar"	
2.		Cancela la acción y cierra el formulario.
3.		Termina el caso de uso.
5.a. Existen campos vacíos.		
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 de la Sección 3: Editar propiedades de reunión.
5.b.Existen campos incorrectos		
	Actor	Sistema
1.		Muestra un mensaje: "El valor introducido en el campo es incorrecto".
2.		Regresa al paso 3 de la Sección 1: Crear reunión.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Crear reunión

Tipo de reunión

Sesión

Fecha

Motivo

Responsable

Participantes

Editar reunión

Sesión

Motivo

Responsable

Participantes

Tabla 4: CU_Gestionar acuerdos

Caso de uso	Gestionar acuerdos
Objetivo	Permite al usuario crear, editar propiedades, ver detalles y eliminar un acuerdo.
Actores	Planificador

Resumen	El caso de uso inicia cuando el planificador desea: crear, editar propiedades, ver detalles o eliminar un acuerdo. El caso de uso termina cuando realiza alguna de estas acciones.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.	
Post-condiciones	Se almacena el acuerdo creado, se actualizan las propiedades de un acuerdo, permite visualizar los detalles del mismo y elimina un acuerdo el cual desaparece de manera permanente del sistema..	
Flujo de eventos		
Flujo básico: Gestionar acuerdos.		
	Actor	Sistema
1.	Crea, edita propiedades, visualiza detalles y elimina un acuerdo de una reunión.	
2.		Permite realizar varias acciones con la reunión: <ul style="list-style-type: none"> - Crear un acuerdo. Ver Sección 1: Crear acuerdo. - Editar propiedades de un acuerdo: Ver Sección 2: Editar propiedades de un acuerdo. - Ver detalles de un acuerdo: Ver sección 3: Ver detalles de acuerdo. - Eliminar acuerdo: Ver Sección 4: Eliminar acuerdo.
3.		Termina el caso de uso.
Sección1: "Crear acuerdo."		
1.	Selecciona la opción que le permite crear un acuerdo.	
2.		Muestra un formulario "Crear acuerdo" con los siguientes datos a insertar: <ul style="list-style-type: none"> - Contenido - Prioridad - Responsable - Participantes. - Fecha de emisión

		<ul style="list-style-type: none"> - Fecha de cumplimiento inicial <p>Y permite:</p> <ul style="list-style-type: none"> - Crear - Cancelar
3.	Inserta los valores en el formulario.	
4.	Selecciona el botón de “Crear”.	
5.		Valida los datos insertados
6.		Crea el nuevo acuerdo con los datos insertados
7.		Termina el caso de uso
Flujos alternos		
4.a Selecciona la opción que le permite “Cancelar”.		
	Actor	Sistema
1.	Selecciona la opción “Cancelar”	
2.		Cancela la acción y cierra el formulario de “Crear acuerdo”.
3.		Termina el caso de uso.
Flujos alternos		
5.a Existen campos vacíos.		
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: “El campo no puede estar vacío”.
2.		Regresa al paso 3 de la Sección 1:Crear acuerdo.
Flujos alternos		
5.b Existen campos incorrectos		
	Actor	Sistema
1.		Muestra un mensaje: “El valor introducido en el campo es incorrecto”.
2.		Regresa al paso 3 de la Sección 1:Crear acuerdo.
Sección 2:”Editar propiedades de acuerdo”.		
1.	Selecciona la acción editar propiedades de acuerdo.	
2.		<p>Muestra un formulario con los siguientes datos a modificar:</p> <ul style="list-style-type: none"> - Contenido - Prioridad - Responsables

		<ul style="list-style-type: none"> - Participantes. <p>Y permite:</p> <ul style="list-style-type: none"> - Actualizar - Cancelar
3.	Modifica los datos que estime conveniente.	
4.	Presiona el botón "Actualizar".	
5.		Valida los datos insertados
6.		Modifica las propiedades de acuerdo con los datos insertados.
7.		Termina el caso de uso.
Flujos alternos		
4.a Presiona el botón "Cancelar"		
	Actor	Sistema
1.	Presiona el botón "Cancelar"	
2.		Cancela la acción y cierra el formulario.
3.		Termina el caso de uso.
5.a. Existen campos vacíos.		
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 de la Sección 2: Editar propiedades de acuerdo.
5.b.Existen campos incorrectos		
	Actor	Sistema
1.		Muestra un mensaje: "El valor introducido en el campo es incorrecto".
2.		Regresa al paso 3 de la Sección 2: Editar propiedades de acuerdo..
Sección 3: "Ver detalles de un acuerdo".		
1.	Selecciona la acción ver detalles de acuerdo	
2.		<p>Muestra una interfaz con todos los detalles del acuerdo:</p> <ul style="list-style-type: none"> - Fecha de cumplimiento, - Responsables

		<ul style="list-style-type: none"> - Participantes - Contenido <p>Y permite:</p> <ul style="list-style-type: none"> - Aceptar
3.	Selecciona el botón "Aceptar".	
4.		Termina el caso de uso.
Sección 4 Eliminar acuerdo.		
1.	Presiona la acción eliminar acuerdo de un acuerdo escogido.	
2.		<p>Muestra una interfaz de confirmación de eliminación.</p> <p>Y permite:</p> <ul style="list-style-type: none"> - Eliminar - Cancelar
3.	Selecciona el botón "Eliminar"	
4.		Elimina el acuerdo escogido
5.		Actualiza la tabla
6.		Termina el caso de uso.
Flujos alternos		
3.a Selecciona el botón de "Cancelar".		
	Actor	Sistema
1.	Presiona el botón de "Cancelar".	
2.		Permite cerrar la interfaz de confirmación de eliminación.
3.		Termina el caso de uso
Relaciones	CU incluidos	Sección 1: CU.Calcular Índice de Acuerdos Cumplidos (IAC)
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Crear acuerdo

Contenido

Prioridad

Fecha cumplimiento

Responsable

Participantes

Aceptar

Cancelar

Editar acuerdo

Contenido

Tipo de reunión

Responsable

Participantes

Aceptar

Cancelar

Detalles del acuerdo	
Creador:	admin
Nombre:	A-0001-R-0001-R-0001-CD-2019
Contenido:	Asamblea nacional del poder popular
Prioridad:	Media
Estado:	Detenido
Porcentaje:	0
Fecha de emisión:	12/13/2018
Fecha de cumplimiento:	06/18/2019
Responsables:	<input type="text" value="User"/>
Participantes:	<input type="text" value="User"/>
<input type="button" value="Aceptar"/>	
Eliminar acuerdo	
¿Estás seguro que desea eliminar este elemento?	
<input type="button" value="Eliminar"/> <input type="button" value="Cancelar"/>	

2.4. Matriz de trazabilidad de requisitos.

Técnica de validación de requisitos que recoge información bi-direccional de trazas: trazabilidad entre requisitos, trazabilidad de requisitos con su origen y trazabilidad de requisitos con su implementación[CITATION Jos16 \l 3082].

Para la investigación se realizaron 4 tipos de matrices de trazabilidad: Matriz de trazabilidad RF-MD, Matriz de trazabilidad RF-RF, Matriz de trazabilidad RF-CU y Matriz de trazabilidad CU-CP.

A continuación, se muestra la matriz de trazabilidad de requisitos funcionales con los casos de uso (RF-CU), la cual establece la relación entre ellos.

2.5.1. Arquitectura en capas

Define una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior. Entre las principales características de este estilo se encuentran la descomposición de los servicios de forma que la mayoría de las interacciones ocurren solo entre capas vecinas. Los componentes de cada capa se comunican con los componentes de otras capas a través de interfaces conocidas y cada nivel agrega las responsabilidades y abstracciones del nivel inferior.[CITATION Peñ16 \l 3082]

Esta arquitectura cuenta con 3 capas, estas son: Capa de presentación, Capa de aplicación y Capa de datos.

Capa de presentación: consiste en una interfaz gráfica que reúne los aspectos de software enfocados a la interacción con los diferentes tipos de usuarios. Es decir, incluye el manejo y aspecto de las ventanas, la autenticación, el formato de los reportes, menús, gráficos y demás elementos multimedia[CITATION Tor13 \l 1033].

Capa de aplicación: reúne los aspectos de software que automatizan los procesos de negocio. Conocida también como capa de la Lógica de la Aplicación. Recibe la entrada de la capa anterior, interactúa con los servicios de datos para ejecutar las operaciones y envía el resultado procesado a la capa de presentación[CITATION Tor13 \l 1033].

Capa de datos: contiene los datos necesarios para la aplicación. Es la encargada de almacenarlos, recuperarlos y mantener su integridad. Estos datos consisten en cualquier fuente de información, incluido una base de datos de empresa como Oracle o MySQL, un conjunto de documentos XML o incluso un servicio de directorio como LDAP. Además del tradicional mecanismo de almacenamiento relacional de base de datos, existen muchas fuentes diferentes de datos de empresa a las que pueden acceder las aplicaciones[CITATION Tor13 \l 1033].

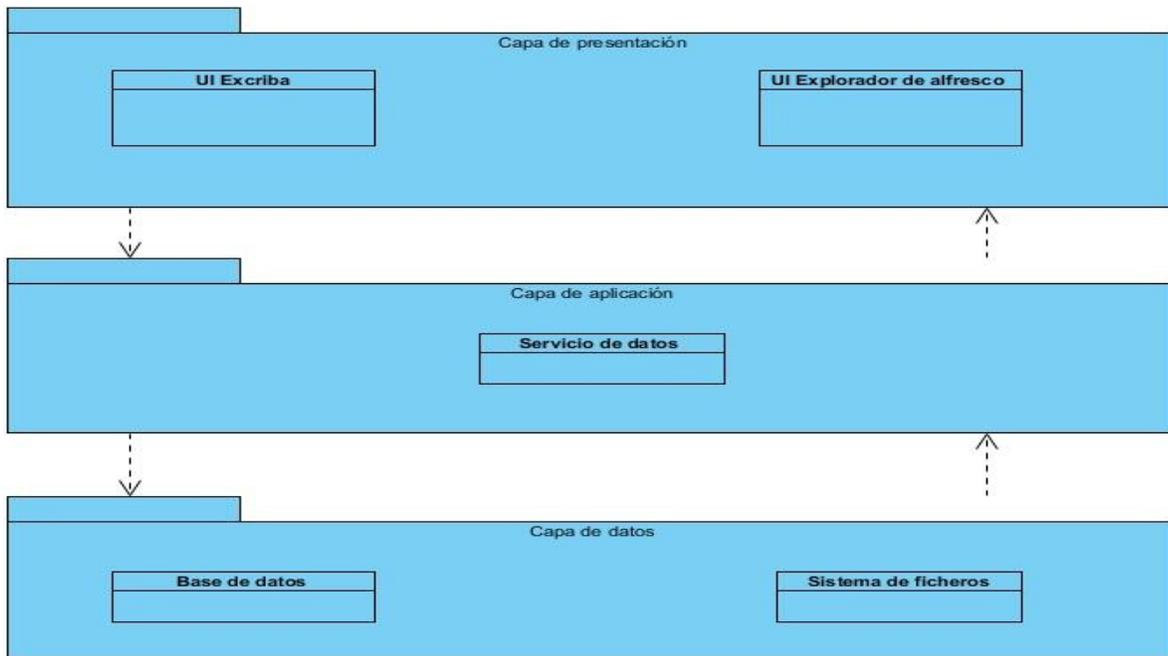


Figura 5: Descripción de la arquitectura

2.6. Patrones de diseño.

Un patrón de diseño es la solución a un problema de diseño, el cual debe haber comprobado su efectividad resolviendo problemas similares en el pasado, también tiene que ser reutilizable, por lo que se deben poder usar para resolver problemas parecidos en contextos diferentes [CITATION Bla16 \l 3082].

2.6.1 Patrones GRASP

Los patrones GRASP (General Responsibility Assignment Software Patterns) describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones [CITATION Lar03 \l 3082].

Patrón Experto: es la clase que tiene la información necesaria para realizar la responsabilidad. Este se utiliza a la hora de modelar las entidades persistentes de la solución donde las funcionalidades propias de cada información reimplementan en la clase que contiene toda la información, o sea, la clase experta en la información [CITATION Lar03 \l 3082].

Patrón Creador: guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento. Al escogerlo como creador, se da soporte al bajo acoplamiento [CITATION Lar03 \l 3082].

Patrón Controlador: propone el diseño de clases con la responsabilidad de controlar el flujo de eventos del sistema a clases específicas. Es más que asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase[CITATION Lar03 \l 3082].

Patrón Bajo Acoplamiento: es un principio que se debe recordar durante las decisiones de diseño: es la meta principal que es preciso tener presente siempre. Es un patrón evaluativo que el diseñador aplica al juzgar sus decisiones de diseño. El acoplamiento es una medida de la fuerza con que una clase está conectada a otras, con las que conoce y con que recurre a ellas. El patrón propone un diseño de clases más independiente, lo que reduce el impacto del cambio y facilita la reutilización en otros sistemas[CITATION Lar03 \l 3082].

Patrón Alta Cohesión: es un patrón evaluativo que el desarrollador aplica al valorar sus decisiones de diseño. Grady Booch señala que se da una alta cohesión funcional cuando los elementos de un componente (clase, por ejemplo) "colaboran para producir algún comportamiento bien definido"[CITATION Lar03 \l 3082].

2.6.2. Patrones GOF

Los Patrones Gang of Four (GOF) describen las formas comunes en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. Se clasifican en 3 grandes categorías basadas en su propósito: creacionales, estructurales y de comportamiento[CITATION Gam94 \l 3082].

Creacionales: describen las formas de crear instancias de objetos. El objetivo de estos patrones es abstraer el proceso de instanciación y ocultar los detalles de cómo los objetos son creados o inicializados[CITATION Gam94 \l 3082].

Estructurales: describen como las clases y objetos pueden ser combinados para formar grandes estructuras y proporcionar nuevas funcionalidades. Estos objetos adicionales pueden ser incluso objetos simples u objetos compuestos[CITATION Gam94 \l 3082].

Comportamiento: definen la comunicación e iteración entre los objetos de un sistema. El propósito de este patrón es reducir el acoplamiento entre los objetos[CITATION Gam94 \l 3082].

2.7. Conclusiones parciales.

Una vez realizadas las actividades de planificación y diseño de la propuesta de solución se obtuvo una visión más clara y detallada del sistema que se desea desarrollar. La definición del modelo conceptual con su respectiva descripción de las clases del dominio, facilitó un mejor entendimiento del negocio a

informatizar. Los requisitos funcionales y no funcionales obtenidos a partir del proceso de identificación de requisitos, constituyeron elementos claves en la construcción de la propuesta de solución. La utilización de los patrones de diseño permitió identificar aspectos importantes de la estructura del diseño del sistema web propuesto, lo que garantizó una mayor organización. Se generaron los artefactos de diseño acorde a la metodología seleccionada, sentando las bases para la implementación del software.

Capítulo 3: Implementación y pruebas

El presente capítulo tiene como objetivo la implementación y prueba del módulo, evidenciándose las disciplinas: Implementación, Pruebas Internas y Pruebas de Aceptación que propone la metodología AUP-UCI para el ciclo de vida del proyecto. Se define el modelo de contenido, los webscripts y la interfaz gráfica del módulo. Además, se diseñan los casos de pruebas para facilitar la disciplina Pruebas internas.

3.1. Modelo contenido

Un modelo de contenido describe los datos que se están guardando en el repositorio y proporciona una base para la estructuración y el trabajo con contenidos específicos. Para la creación del modelo de contenido referente al módulo gestión de acuerdo se definieron dos tipos de contenido: “ga:acuerdo” referente al acuerdo y “ga:reunión” referente a las reuniones. Ambos heredan de “cm:folder” que es el tipo de contenido que posee las propiedades básicas de todo tipo de carpetas. Se estableció la asociación del tipo Child-association la cual funciona como un cascade delete, es decir, el hijo sólo existe si existe el padre. Esta asociación se estableció de tipo “ga:acuerdo_reunion” entre el tipo de contenido “ga:acuerdo” con “ga:reuniones”, para establecer las relaciones entre los diferentes tipos de contenidos. Para insertar el modelo de contenido personalizado en Alfresco se crea el archivo bootstrap-context.xml. Luego se ubica en la carpeta extensión del servidor de Alfresco.

3.2. Webscript

Alfresco para la implementación de los módulos web propone en su arquitectura dos tipos de webscripts, webscript de presentación y Webscript de acceso al repositorio.

Webscripts de presentación: dichos webscripts modifican la capa de presentación de Alfresco. Pueden definir nuevos componentes en su interfaz de usuario.

Webscripts de acceso al repositorio: son aquellos que consultan, modifican y retornan datos de Alfresco (documentos), ofreciendo una interfaz para consumir dicho servicio por terceros.

A continuación, se muestra el listado de los webscript de acceso al repositorio, desarrollados en la realización del módulo de gestión de acuerdo.



Figura 6:Listado de webscript de acceso al repositorio.

A continuación, se explica el webscript crear reunión, el cual está compuesto por los siguientes archivos:

crearReunion.get.desc: Archivo que brinda la descripción del Webscript. El archivo se encuentra en la ruta: *WEBINF/classes/alfresco/extensión/templates/webscripts/cu/uci*.

```

<webscript>
  <shortname>crearReunion</shortname>
  <description>crearReunion</description>
  <url>/cu/uci/excriba-acuerdo-reuniones/crearReunion</url>
  <format default="json">extension</format>
  <authentication runas="admin">user</authentication>
</webscript>

```

Figura 7: Crear reunión.xml

crearReunion.get.js:

```

function crearReunion(TipoReunion, Motivo, Sesion, Fecha){
    var date=new Date();
    var properties = [];
    var todo = TipoReunion;
    var t1 = todo.split("(");
    var t2 = t1.split(")");
    var siglas = t2[0];
    var Tipo = t2[1];
    var criteria = 'TYPE:"ga:reunion"';
    criteria += ' AND @ga\\:\\:siglas:' + siglas;
    criteria+= ' AND @cm\\:\\:creted:[' + date.getFullYear() + ' "\\-01\\-01T00:00:00 TO " '+
    date.getFullYear() + ' "\\-12\\-31T59:59:00]" ';
    nodes = search.luceneSearch('workspace://SpacesStore', criteria, "@cm:created",false);

    var cont = 1;
    var Numero;
    if(nodes.length()>0){
        var cont = nodes[0].properties["ga:numero_reunion"];
        cont ++;
    }
    if(cont<10){
        Numero = "000"+cont;
    }
    if(cont >= 10 && cont < 100){
        Numero = "00"+cont;
    }
    if (cont >= 100 && cont < 1000){
        Numero = "0"+cont;
    }
    if(cont>= 1000){
        Numero = cont;
    }
}

```

Figura 8: crear reunion.js

```

var Nombre = "R-"+Numero+"-"+siglas+"-"+date.getFullYear();
var rootFolder;
var subFolder1;
var node;
//var recordDestination;
rootFolder = companyhome.childByNamePath("AGREEMENT_MODULE");
if (!rootFolder) {
    rootFolder = companyhome.createFolder("AGREEMENT_MODULE", "cm:folder");
}
subFolder1 = rootFolder.childByNamePath(siglas);
if (!subFolder1) {
    subFolder1 = rootFolder.createFolder(siglas,"cm:folder");
}
node = subFolder1.childByNamePath(Nombre);
if (!node) {
    node = subFolder1.createFolder(Nombre,"ga:reunion");

    node.properties["cm:created"]= date;
    node.properties["cm:creator"]= usuarioActual(); //usuario en el sistema
    node.properties["cm:name"]= Nombre;
    node.properties["ga:numero_reunion"]= Numero;
    node.save();
    node.properties["ga:motivo_reunion"]= Motivo;
    node.properties["ga:estado_reunion"]= "Activa";
    node.properties["ga:fecha_encuentro_reunion"]= Fecha;
    node.properties["ga:tipo_reunion"]= Tipo;
    node.properties["ga:sesion_reunion"]= Sesion;
    node.properties["ga:siglas"]= siglas;
    node.save();
}
}
return node;
}
function usuarioActual(){
    return person.properties.userName;
}
}
try {
    //model.result = crearReunion(args['Tipo'],args['Motivo'],args['Sesion'],args['Fecha']);
    //model.result = crearReunion("(CE) comision de expertos","Prueba","24");
    console.log(args['Tipo']);
    console.log(args['Motivo']);
    console.log(args['Sesion']);
    console.log(args['Fecha']);
} catch (e) {
    model.error = e.message;
}
}

```

```

}
return node;
}
function usuarioActual(){
    return person.properties.userName;
}
}
try {
    //model.result = crearReunion(args['Tipo'],args['Motivo'],args['Sesion'],args['Fecha']);
    //model.result = crearReunion("(CE) comision de expertos","Prueba","24");
    console.log(args['Tipo']);
    console.log(args['Motivo']);
    console.log(args['Sesion']);
    console.log(args['Fecha']);
} catch (e) {
    model.error = e.message;
}
}

```

Figura 9: crear reunion.js

crearReunion.get.json.ftl: Archivo donde se define la estructura, de la respuesta del servidor, este puede ser en formato html o json como el fichero actual. Se vincula junto al motor de plantillas FreeMarker.

```
<#if error??>{
    "error" : "${error?string}"
}
<#else>{
    "result" : "${result}"
}
</#if>
```

Figura 10: crear reunión.ftl

3.3. Interfaz gráfica del módulo

A continuación, se muestra una imagen con la interfaz final del módulo de gestión de acuerdos.

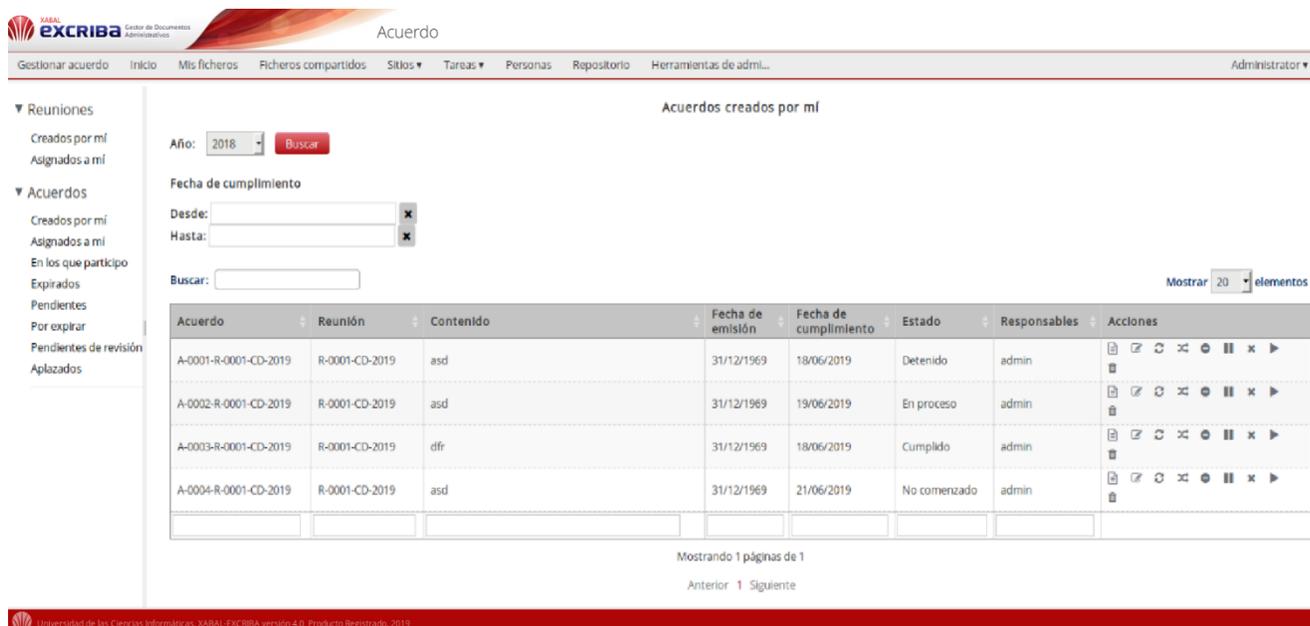


Figura 11: Interfaz principal

3.4. Diagrama de despliegue

Un diagrama de despliegue es la forma de mostrar la configuración de nodos de procesamientos en tiempo de ejecución y los componentes que en ellos residen. Estos nodos forman la topología de hardware sobre el que se ejecuta el sistema. Este diagrama se preocupa principalmente de la

distribución, entrega e instalación de las partes que constituye el sistema físico[CITATION Eli15 \l 3082].

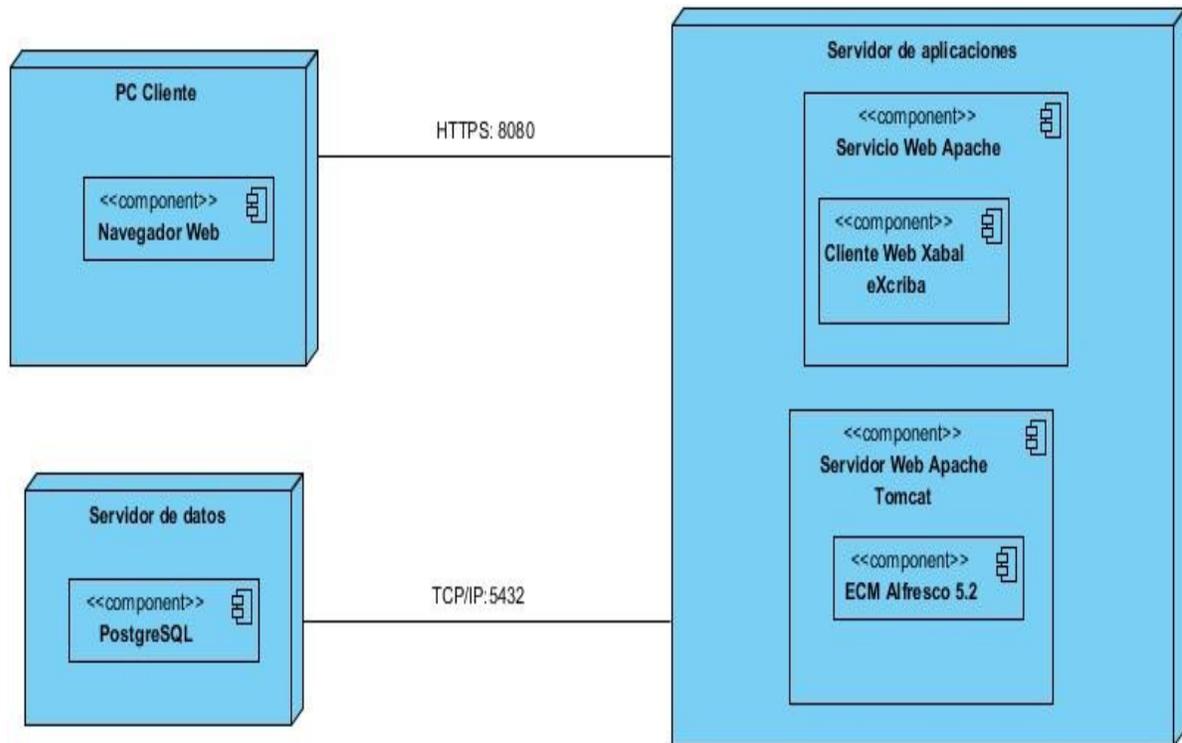


Figura 12: Diagrama de despliegue

Descripción de los nodos:

PC cliente: Nodo donde estarán alojados los navegadores web para acceder a las páginas clientes que interactúan con las páginas servidoras. Se procesan todas las interfaces de usuario las cuales son las encargadas de establecer la comunicación entre el usuario y el GDA eXcriba.

Servidor de aplicaciones: En la cual estará alojado el sistema en cuestión, o sea, la interfaz web (Front End) del sistema GDA, hosteado en el servidor web Apache, y su núcleo el ECM Alfresco 5.2 además de la capa de servicios incluida en el propio sistema y el servidor de aplicaciones Apache-Tomcat.

- **Servidor Web Apache:** El servidor web apache provee un mecanismo para ofrecer la aplicación web a los usuarios. En el caso del presente módulo, en este nodo se encuentran todos los componentes referentes a la capa lógica del negocio.
- **Servidor Web Apache Tomcat:** Tomcat es un servidor web cuyo uso sólo tiene el propósito de contener al ECM Alfresco. En el caso de este módulo, aquí se encuentran desplegados los servicios que serán usados para brindar funcionalidades al módulo.

Servidor de base de datos: En este servidor se encuentra el sistema gestor de bases de datos PostgreSQL. Aquí es donde se encuentra toda la información estructurada relacionada con el presente módulo. Se almacena toda la información referente a los acuerdos y reuniones.

3.4.1. Despliegue de la solución

A continuación, se muestra el despliegue de la solución del Módulo de Gestión de Acuerdo, donde se expone la estructura de carpetas y archivos con que cuenta el módulo.

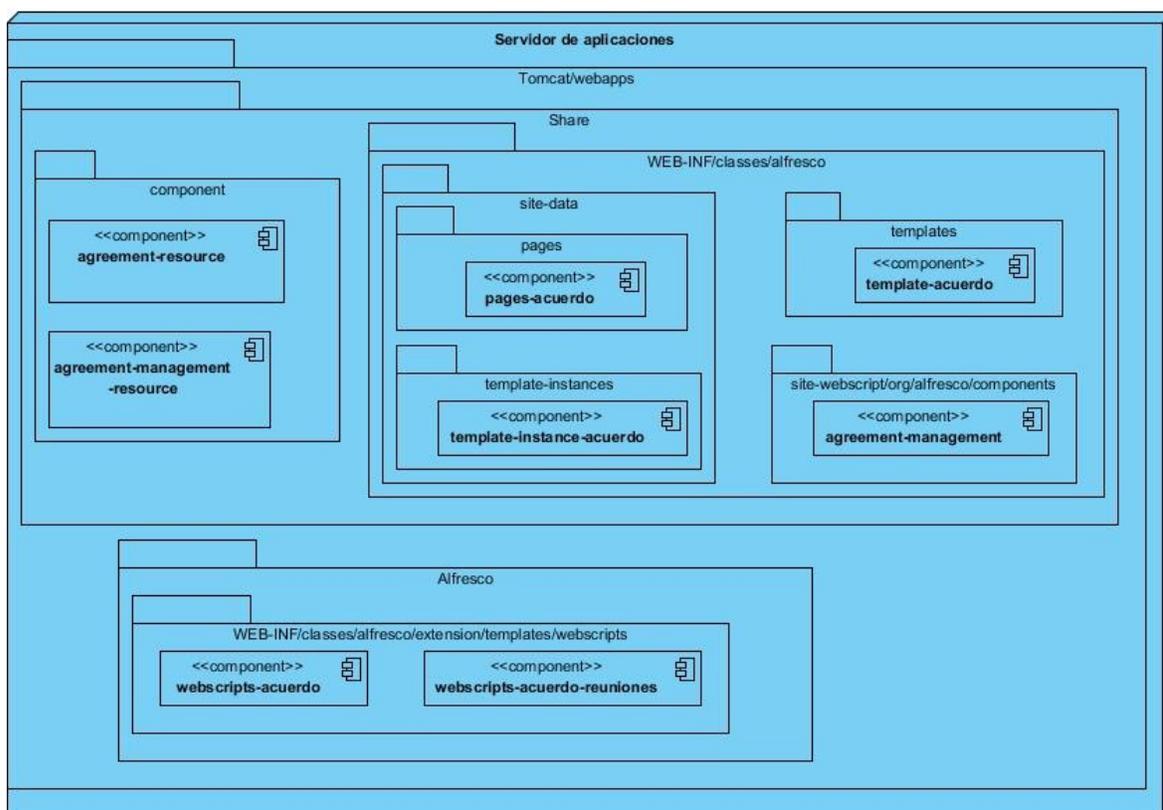


Figura 13: Estructura de carpeta del módulo Gestión de Acuerdo

3.5. Pruebas de software

Según [CITATION Pre19 \l 3082] las pruebas de software son un elemento crítico para la garantía de calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación. Se utilizan para identificar los fallos de implementación, calidad, o usabilidad de un sistema. Básicamente es el proceso de ejecución de un programa con la intención de encontrar errores [CITATION Man15 \l 3082]. Se integran dentro de las diferentes fases del ciclo de la Ingeniería de Software ejecutando un programa que mediante técnicas experimentales tratando de descubrir los errores que tiene el sistema.

Según la metodología utilizada en el presente trabajo, las pruebas a realizar son prueba de caja blanca y prueba de caja negra.

3.5.1. Pruebas de caja blanca

La prueba de caja blanca, en ocasiones llamada prueba de caja de vidrio, es una filosofía de diseño de casos de prueba que usa la estructura de control descrita como parte del diseño a nivel de componentes para derivar casos de prueba[CITATION Rog10 \l 3082]. Al usar los métodos de prueba de caja blanca se garantiza: que todas las rutas independientes dentro del módulo se revisaron al menos una vez, que se revisen todas las decisiones lógicas en sus lados verdadero y falso, que sean ejecutados todos los bucles en sus fronteras y dentro de sus fronteras operativas y revisadas las estructuras de datos internas para garantizar su validez.

Dentro de la prueba de caja blanca se incluyen las Técnicas de Pruebas que serán descritas a continuación:

- **Prueba del camino básico**

La prueba de caja blanca realizada a la propuesta de solución fue la prueba del camino básico, a partir del cálculo de la complejidad ciclomática del algoritmo a ser analizado.

Para realizarla se deben enumerar las sentencias de código y a partir de ahí elaborar el grafo de flujo de esta funcionalidad. Se definieron una serie de pasos a seguir:

Notificación del grafo de flujo: usando el código como base se realiza la representación del grafo de flujo, mediante una sencilla notación. Cada construcción estructurada tiene su correspondiente símbolo[CITATION Rog10 \l 3082].

- **Nodo:** Cada circulo denominado nodo, representa una o más sentencias procedimentales.
- **Arista:** Son las flechas del grafo de flujo, representan el flujo de control y son análogas a las flechas del diagrama de flujo.
- **Región:** Son las áreas delimitadas por aristas y nodos.

Complejidad ciclomática: es una medición de software que proporciona una evaluación cuantitativa de la complejidad lógica de un programa. Cuando se usa en el contexto del método de prueba de la ruta básica, el valor calculado por la complejidad ciclomática define el número de rutas independientes del conjunto básico de un programa y le brinda una cota superior para el número de pruebas que debe realizar a fin de asegurar que todos los enunciados se ejecutaron al menos una vez[CITATION Rog10 \l 3082]. La complejidad se calcula en una de tres formas:

- El número de regiones del gráfico de flujo corresponde a la complejidad ciclomática.
- La complejidad ciclomática $V(G)$ para un gráfico de flujo G se define como $V(G) = A - N + 2$ donde A es el número de aristas del gráfico de flujo y N el número de nodos del gráfico de flujo.
- La complejidad ciclomática $V(G)$ para un gráfico de flujo G también se define como $V(G) = P + 1$ donde P es el número de nodos predicado contenidos en el gráfico de flujo G [CITATION Rog10 \l 3082].

Determinar un conjunto básico de caminos linealmente independientes: el valor de $V(G)$ es el número de caminos linealmente independientes de la estructura de control del programa [CITATION Rog10 \l 3082].

Obtención de casos de prueba: se realizan los casos de pruebas que forzarán la ejecución de cada camino del conjunto básico [CITATION Rog10 \l 3082].

A continuación, se muestra el requisito funcional de Cancelar Reunión seleccionado como caso de prueba debido a que es una de las funcionalidades principales de la propuesta de solución:

```

function cancelarReunion(node, motivo){
    node.properties["ga:estado_reunion"] = "Cancelada";
    node.save();
    node.addAspect("ga:reunion_cancelada");
    node.properties["ga:motivo_reunion_cancelada"] = motivo;
    node.save();
1
    var criteria = 'TYPE:"ga:acuerdo"';
    nodes = search.luceneSearch('workspace://SpacesStore/'+node.properties["sys:node-uuid"], criteria, "@cm:created",true);
    var i = 0;
2
    for(i;i<nodes.length();i++){
3
        if(nodo.sourceAssocs["ga:acuerdo"]){
4
            nodes[i].properties["ga:estado_acuerdo"] = "Cancelado";
            nodes[i].save();
        }
    }
    try{
5
        model.result = cancelarReunion(BuscarNodo(args["uuid"]), args["motivo"]);
    }
    catch(e){
        model.error = e.message;
    }
}

```

Figura 14: Código de la función Cancelar Reunión

Elaboración del grafo de flujo:

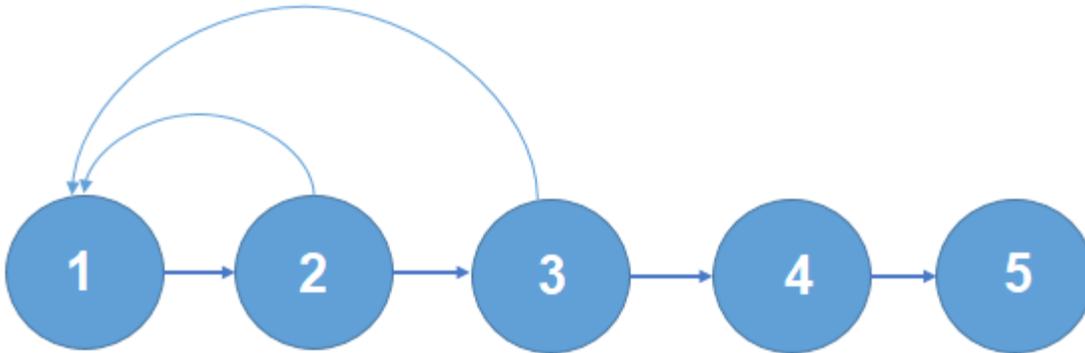


Figura 15: Gráfico de flujo de la función Cancelar Reunión

Complejidad ciclomática

$$V(G) = A - N + 2$$

$$V(G) = 6 - 5 + 2$$

$$V(G) = 3$$

$$V(G) = P + 1$$

$$V(G) = 2 + 1$$

$$V(G) = 3$$

$$V(G) = \text{Regiones}$$

$$V(G) = 3 \text{ Caminos independientes}$$

Resultados de la prueba del camino básico.

El cálculo efectuado a través de las 3 fórmulas ha dado el mismo valor, dando como resultado 3, lo que indica que existen 3 posibles caminos por donde el flujo puede circular.

Caminos independientes

Camino 1: 1,2,3,4,5

Camino 2: 1,2,3,2,5

Camino 3: 1,2,5

3.5.2. Pruebas de caja negra

Las pruebas de caja negra, también llamadas pruebas de comportamiento, se enfocan en los requerimientos funcionales del software; es decir, las técnicas de prueba de caja negra le permiten derivar conjuntos de condiciones de entrada que revisarán por completo todos los requerimientos funcionales para un programa. Las pruebas de caja negra no son una alternativa para las técnicas de caja blanca. En vez de ello, es un enfoque complementario que es probable que descubra una clase de errores diferente que los métodos de caja blanca[CITATION Rog10 \l 3082].

Las pruebas de caja negra intentan encontrar errores en las categorías siguientes[CITATION Rog10 \l 3082]:

- Funciones incorrectas o faltantes.
- Errores de interfaz.
- Errores en las estructuras de datos o en el acceso a bases de datos externas.
- Errores de comportamiento o rendimiento.
- Errores de inicialización y terminación.

Para un mejor desarrollo de las pruebas de caja negra es necesario aplicar varias técnicas, entre ellas[CITATION Jac043 \l 3082]:

- **Técnica de la partición de equivalencia:** esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- **Técnica del análisis de valores límites:** esta técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- **Técnica de grafos de causa-efecto:** es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

En la presente investigación se aplicó la técnica de partición de equivalencia para la generación de los casos de pruebas en el nivel de trabajo pruebas de integración.

A continuación, se muestra un ejemplo del caso de prueba aplicado al requisito funcional Replanificar reunión, con el objetivo de demostrar que las funciones del software son operativas, que la entrada y salida se produce de forma correcta y que además se mantiene la integridad de la información.

Figura 16: Caso de prueba: Replanificar reunión

Escenarios	Descripción	Fecha	Respuesta del sistema	Flujo central
EC 1.1 Replanificar reunión con datos correctos	Permite al usuario modificar la fecha de una reunión proporcionando la fecha correctamente.	V	Se modifica el estado de una reunión a Replanificada. Notifica mediante un correo electrónico la replanificación a todos los involucrados.	<ol style="list-style-type: none"> 1. El usuario accede al módulo Gestion de acuerdos. 2. Selecciona la acción de replanificar reunión. 3. Inserta la nueva fecha de la reunión. 4. Presiona el botón aceptar.
EC 1.2 Replanificar reunión con datos incorrectos.	El usuario introduce la fecha incorrecta en el formulario de inicio de sesión.	I	Muestra un mensaje de error: "La fecha insertada es incorrecta".	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona la acción de replanificar reunión. 3- Introduce una fecha incorrecta 4- Presiona aceptar.
EC 1.3 Replanificar reunión con campos vacíos	El usuario deja campos vacíos en el formulario de inicio de sesión	-	Muestra el mensaje: "Existen campos vacíos"	<ol style="list-style-type: none"> 1- Selecciona la acción de replanificar reunión 2- Deja el campo vacío y Presionar aceptar.

EC 1.4 Cancelar acción	Permite al usuario cancelar la acción.	-	El sistema muestra la página de origen.	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona la acción de replanificar reunión. 3- Introduce una fecha. 4- Presiona cancelar.
------------------------------	--	---	--	---

Resultados de las pruebas

Con el objetivo de que las funcionalidades correspondientes al módulo gestión de acuerdo tenga un correcto funcionamiento, se realizaron las pruebas en tres iteraciones. Durante la primera iteración se detectaron 15 no conformidades: 6 asociadas a errores de ortografía, 4 a errores de validación y 5 a errores funcionales. En la segunda iteración se detectaron 3 no conformidades: 2 errores asociados a ortografía y 1 error funcional. A los errores identificados en las dos primeras iteraciones se solucionaron rápidamente. Para finalizar en la tercera iteración no se detectaron no conformidades, es decir, que se verificó que se le ha dado solución a la problemática planteada con resultados satisfactorios. A continuación, se muestra en un gráfico de barras las iteraciones efectuadas y la cantidad de errores identificadas.

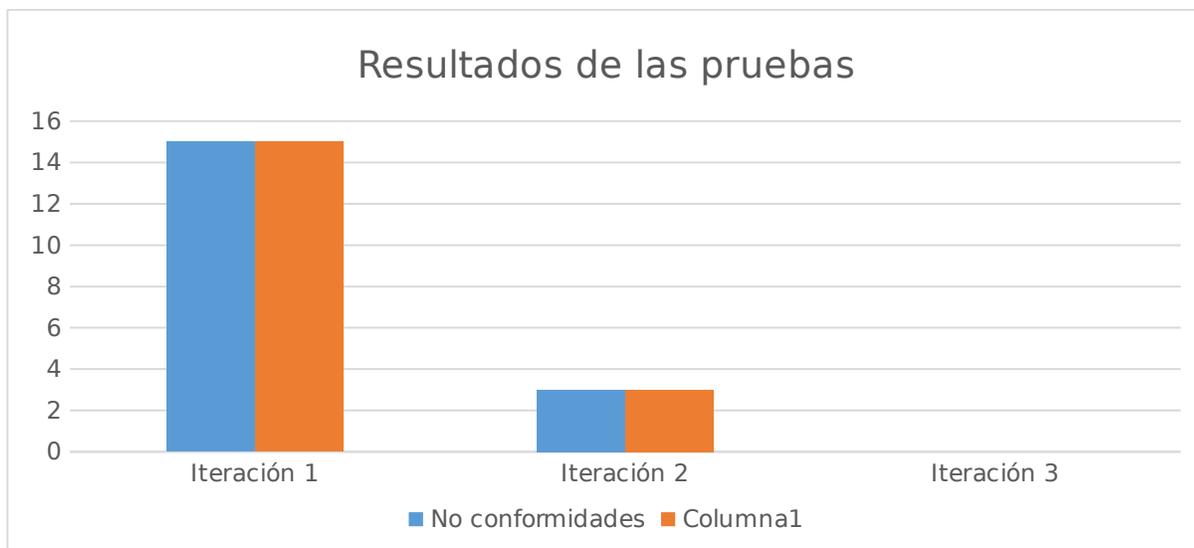


Figura 17: Resultados de las pruebas

Tabla 5: Iteraciones de las pruebas

Iteración	Ortografía	Validación	Funcional	Prioridad
1	6		5	Alta
		4		Media
2	2		1	Media
3	0	0	0	Baja
	0	0	0	Baja

Pruebas de aceptación

Para el cumplimiento de esta disciplina, en el centro CIGED se desplegó un servidor piloto para la realización de una prueba final antes de dar paso al despliegue del sistema. El objetivo que se perseguía era verificar que el software tuviera las condiciones dadas y listo para ser usuarios finales que pudieran ejecutar todas las funcionalidades y tareas para las cuales el software fue construido. Como resultado de dicha prueba final, se obtuvo un acta de aceptación por el cliente (centro CIGED) donde quedó plasmada su conformidad con el módulo.

Conclusiones parciales

La definición del modelo de contenido y el desarrollo de los webscripts contribuyeron en la construcción del módulo de gestión de acuerdos para el Alfresco 5.2. Se realizaron pruebas de caja negra mediante la técnica de partición de equivalencia a la solución informática con el equipo de trabajo del proyecto. Durante las 3 iteraciones de pruebas se detectaron un conjunto de no conformidades las cuales fueron resultas después de culminada cada iteración. Las pruebas de caja blanca realizadas a un fragmento de código de la implementación arrojaron una complejidad ciclomática de 3 unidades dando lugar a 3 caminos básicos, comprobando que eran los únicos por los que siempre transitaba el flujo.

Conclusiones Generales

Una vez finalizado el presente trabajo de diploma donde se abarca todo el proceso de desarrollo del módulo gestión de acuerdo en la tecnología Alfresco Community 5.2, se puede concluir que se les dio cumplimiento a todos los objetivos propuestos, resaltando que:

1. El estudio realizado sobre los conceptos asociados al dominio del problema permitió sentar las bases teóricas de la investigación.
2. El estudio del estado actual de las herramientas informáticas de gestión de acuerdos, evidenció la necesidad de desarrollar el componente propuesto, ya que ninguna de las aplicaciones analizadas constituye una solución al problema planteado en la presente investigación.
3. La selección de la metodología, herramientas y tecnologías posibilitaron el desarrollo del módulo de gestión de acuerdo.
4. La definición de la estrategia de prueba permitió la validación del módulo garantizando su correcto funcionamiento.

Recomendaciones

1. Se recomienda integrar al módulo la opción de imprimir los acuerdos y las reuniones.
2. Incorporarle además el uso de ciclos de revisiones para controlar el acuerdo en todo su período de vida.
3. Integrar el módulo desarrollado con el módulo Gestion de nomencladores cuando se realice la migración del mismo a la tecnología de Alfresco Community 5.2

Referencias

- Alba, R. (2016). *Manual de Javascript*. Madrid: EDITORIAL CEP S.L. Obtenido de <http://ferko.byethost31.com/Manual%20de%20Javascript.pdf>
- Alfresco Documentation. (2019). Obtenido de <https://docs.alfresco.com/5.2/concepts/welcome.html>
- Aranda, O. F. (2013). Implementación de un Sistema de Gestión Documental en. *Implementación de un Sistema de Gestión Documental en*. Granda: Editorial de la Universidad de Granada. Obtenido de <https://hera.ugr.es/tesisugr/22668548.pdf>
- Atareado . (18 de mayo de 2018). *El atareado linux para legos*. (Disqus) Recuperado el 30 de noviembre de 2018, de <https://www.atareao.es/software/programacion/visual-studio-code/>
- Blancarte, O. (2016). *Página*. Ciudad de México: Oscar. Obtenido de <https://reactiveprogramming.io/public/books/patterns/Introduccion-a-los-patrones-de-dise%C3%B1o.pdf>
- Blanchar, B. A. (2017). ANALISIS DE LAS TENDENCIAS TECNOLOGICAS DE LA GESTION DOCUMENTAL.
- Canaan, R. (2018). *lifeder.com*. Recuperado el 15 de 1 de 2019, de <https://www.lifeder.com/tipos-metodos-de-investigacion/>
- Capterra. (2018). *Document Management Software*. OpenKM.
- Cruz, Y. R. (2015). *Information Management and Knowledge Management for the organizational decision making*. La Habana, Cuba.
- Denzer, P. (2002). *PostgreSQL*. Obtenido de <http://profesores.elo.utfsm.cl/~agv/elo330/2s02/projects/denzer/informe.pdf>
- Derrick, D. (2018). *Real Phyton*. Recuperado el 15 de 1 de 2019, de <https://realpython.com/setting-up-sublime-text-3-for-full-stack-python-development/>
- Díaz, A. J. (2016). *Tomcat, Apache Tomcat, Jakarta Tomcat*. Obtenido de <http://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=769>
- Enríquez Ruiz, J. L. (2017). *Uladech*. Recuperado el 15 de 1 de 2019, de Uladech: <https://www.uladech.edu.pe/images/stories/universidad/documentos/2018/metodologia-desarrollo-software-v001.pdf>
- Felipe, J. Á. (2001). GESTION DOCUMENTAL. Obtenido de <http://www.sociedadelainformacion.com/12/Gestion%20Documental.pdf>
- Fernández, L. D. (2006). *Gestion documental*. Caracas-Venezuela.
- Flores, A. A. (2014). *PostgreSQL*.

- Gamma, E. (1994). *Design patterns: elements of reusable object-oriented software*. . Pearson Education.
- Gasparini, A. (2015). *The New Office and PDF Documents Viewer in LogicalDOC 7.6*. LogicalDOC.
- Gauchat, J. D. (2012). *El gran libro de HTML5, CSS3 y JavaScript*. Barcelona, España: MARCOMBO, S.A. 2012 . Obtenido de <https://gutl.jovenclub.cu/wp-content/uploads/2013/10/El+gran+libro+de+HTML5+CSS3+y+Javascrip.pdf>
- González, M. d. (2014). La Gestión Documental y su Impacto en el Sector Empresarial Cubano. *GECONTEC, Vol.2*.
- Jacobson. (2004). Obtenido de <https://es.scribd.com/doc/50327385/El-Proceso-Unificado-de-Desarrollo-de-Software-Jacobson-Booch-Rumbaugh>
- Jacobson. (2004). *El Proceso Unificado de Desarrollo de Software*.
- Jacobson. (2004). *El Proceso Unificado de Desarrollo de Software*. .
- Jacobson, I. B. (2004). *El Proceso Unificado de Desarrollo de Software*. La Habana: Felix Valera.
- Kausen, R. C. (2017). La gestión de reuniones. *factorhuma.org*.
- Larman, C. (2014). *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. Madrid: Pearson Educación 2da Edición.
- León, L. D. (2019). Axure esencial. *inLearning*. Obtenido de <https://es.linkedin.com/learning/axure-esencial/que-es-axure-y-para-que-sirve>
- López, J. G. (2019). *THINK&SELL* . Obtenido de THINK&SELL : <https://thinkandsell.com/servicios/consultoria/software-y-sistemas/sistemas-de-gestion-normalizados/>
- Manuel, J. (2015). *Pruebas de Software. Fundamentos y Técnicas*. UNIVERSIDAD POLITÉCNICA DE MADRID. Obtenido de http://oa.upm.es/40012/1/PFC_JOSE_MANUEL_SANCHEZ_PENO_3.pdf
- Martí. (2007). *Implicaciones del enfoque interdisciplinar en la enseñanza de la gestión de información*. Obtenido de http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352007000200005
- Martínez, I. S., Rodríguez, L. R., & Prado, C. S. (2017). SISTEMA DE GESTIÓN DE INFORMACIÓN DEL DEPARTAMENTO DE INGENIERÍA INFORMÁTICA DE SANCTI SPÍRITUS SYSTEM FOR MANAGING INFORMATION FROM THE DEPARTMENT OF COMPUTER ENGINEERING OF SANCTI SPIRITUS. *Pedagogía y Sociedad, 20*.
- MARTINIC, B. M. (2011). *Guía para el Diseño e Implementación de un Sistema de Gestión de Archivos*. Santiago-Chile. Obtenido de <http://eprints.rclis.org/17358/1/Mercado%20Beatriz%20Gesti%C3%B3n%20de%20Archivos.pdf>

- Michelli, A. (2017). *PROCESO GESTIÓN DOCUMENTAL Código GDOP03*. Colombia. Obtenido de <https://www.minsalud.gov.co/Ministerio/Institucional/Procesos%20y%20procedimientos/GDOP03.pdf>
- Mifsud, E. (Domingo 01 de Agosto de 2010). *Observatorio Tecnológico*. Obtenido de <http://recursostic.educacion.es/observatorio/web/es/software/servidores/807-monografico-alfresco?start=1>
- O'REILLY. (2014). *A Technical Introduction to XML*. Obtenido de <https://www.xml.com/pub/a/98/10/guide0.html#AEN58>.
- Odalys Jordán Enríquez, O. V. (2005). GENERACIÓN DE CASOS DE PRUEBA A PARTIR DE CASOS DE USO EN LAS PRUEBAS DE SOFTWARE. *Informatica*. Obtenido de <http://www.redalyc.org/html/3604/360433560007/>
- OLIVEIRA. (2001). *Sistemas, organização e métodos: uma abordagem gerencial*. São Paulo.
- Peña, A. D. (2016). Arquitectura de software para el sistema de visualización médica Vismedic. *Revista Cubana de Informática Médica*. Obtenido de http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1684-18592016000100006
- Pixelware. (2017). *Pixelware*. Obtenido de Pixelware: <https://pixelware.com/sistemas-gestion-documental/>
- Pressman, R. (2010). *Software Engineering. A practitioner's approach*. (Vol. Vol.6 7ma Edición). Nueva York.
- Reynoso, C. B. (2004). *Introducción a la Arquitectura de Software*. Universidad de Buenos Aires. Obtenido de <http://carlosreynoso.com.ar/archivos/arquitectura/Arquitectura-software.pdf>
- Roger S. Pressman, P. (2010). *Ingeniería del software un enfoque práctico* (Vol. Séptima Edición). University of Connecticut. Obtenido de http://artemisa.unicauca.edu.co/~cardila/Libro_Pressman_7.pdf
- Rosario Osuna, E. d. (Junio de 2010). Los sistemas de gestión de contenidos en Información y Documentación. *Revista General de Información y Documentación, Vol. 20* .
- Sánchez, A. E. (2 de Agosto de 2014). La gestión de documentos como estrategia de innovación empresarial. *Enl@ce: Revista Venezolana de Información, Tecnología y Conocimiento, vol.11*. Obtenido de <http://www.redalyc.org/pdf/823/82332625003.pdf>
- Sánchez, T. R. (2015). *Metodología de desarrollo para la Actividad productiva de la UCI*. La Habana.
- Sandra, J. (2019). *DwES*. Recuperado el 10 de diciembre de 2018, de <http://www.alba9.es/joomla/index.php/publicaciones/23-ventajas-y-desventajas-de-visual-studio-code.html>
- Sandra, J. (2019). *DwES En línea*. Obtenido de <http://www.alba9.es/joomla/index.php/publicaciones/23-ventajas-y-desventajas-de-visual-studio-code.html>.

- Santiago, J. L. (2017). *Sistema automatizado para la gestión de acuerdos como apoyo a toma de decisiones*. Obtenido de www.monografias.com:
<https://www.monografias.com/trabajos-pdf5/sistema-automatizado-gestion-acuerdos-como-apoyo-toma-decisiones/sistema-automatizado-gestion-acuerdos-como-apoyo-toma-decisiones.shtml>
- Sevilla, J. (2016). *Overti*. Madrid.
- Sifontes, C. E. (2015). *BIBLIOTECA VIRTUAL de Derecho, Economía y Ciencias Sociales*. Obtenido de Grupo EUMEDNET de la Universidad de Málaga:
<http://www.eumed.net/libros-gratis/2010c/758/La%20Gestion%20en%20las%20organizaciones.htm>
- Sistemas, I. (2003). *www.ibaiintranets.com*. Obtenido de www.ibaiintranets.com/articulos/10_ventajas_de_implantar_un_sistema_de_gestion_documental.php.
- Snopek, S. (2019). *Best Document Management Systems of 2019*. Obtenido de FinancesOnline.: <https://financesonline.com/top-15-document-management-systems/>
- Sommerville. (2005). *Ingeniería del software*. Madrid: Pearson Educación. Madrid 7ma edición.
- Sommerville. (2007). *Software Engineering*. 8. Addison-Wesley.
- SOMMERVILLE, ARAKAKI, MELNIKOFF. (2008). *Engenharia de software*. Pearson Prentice Hall.
- Torres Hernandez, O. D. (2015). *Propuesta arquitectónica para el núcleo del Gestor de Documentos Administrativos eXcriba 3.0*. La Habana.
- Valencia, L. (2017). *Sharepoint and Office 365 MVP*. Office Services and Servers MVP.
- Vera, V. D. (2015). *Sistemas de Gestión de Contenidos(SGC)*.

Anexos

Anexo 1. Tablas descripción de los casos de uso

Tabla 6: CU_Ver detalles de reunión.

Objetivo	Visualizar las propiedades de una reunión.	
Actores	Usuario	
Resumen	El caso de uso comienza cuando el usuario desea ver detalles de una reunión. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Media	
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.	
Post-condiciones	Se visualiza los datos de una reunión.	
Flujo de eventos		
Flujo básico: Ver detalles de reunión.		
	Actor	Sistema
1.	Selecciona la acción ver detalles de una reunión.	
2.		<p>Muestra una interfaz con todos los detalles de la reunión:</p> <ul style="list-style-type: none"> - Creador - Tipo de reunión - Estado - Sesión - Motivo - Fecha - Responsable - Participantes - Índice acuerdos cumplidos (IAC) - Vínculo al Acta. - Vinculo al Orden del día. <p>Y permite:</p> <ul style="list-style-type: none"> - Aceptar <p>Si el acuerdo tiene estado cancelado se muestra el motivo.</p>

3.	Selecciona el botón "Aceptar".	
4.		Termina el caso de uso.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A

Prototipo elemental de interfaz gráfica de usuario

Detalles de la reunión

Creador: admin

Nombre: R-0001-CD-2019

Fecha: 05/18/2019

Motivo: Marcha de los trabajadores

Tipo: Consejo Dirección

Estado: Activa

Responsables:

User

Participantes:

User

IAC: 0

Aceptar

Tabla 7: CU_Cancelar reunión.

Objetivo	Cancelar una reunión cuyo estado sea: Planificada o Replanificada.
-----------------	--

Actores	Planificador
Resumen	El caso de uso inicia cuando el planificador desea cancelar una reunión El caso de uso termina cuando realiza esta acción.
Complejidad	Media
Prioridad	Media
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.
Post-condiciones	Se cancela una reunión con estado: Planificada o Replanificada. Se notifica mediante un correo electrónico la cancelación a todos los involucrados.

Flujo de eventos

Flujo básico: Cancelar reunión.

	Actor	Sistema
1.	Selecciona la acción "Cancelar reunión" de una reunión.	
2.		Muestra una interfaz de cancelar reunión Y permite: <ul style="list-style-type: none"> - Insertar el motivo de cancelar de la reunión - Aceptar - Cancelar
3.	Inserta en el campo de texto el motivo de cancelar la reunión.	
4.	Selecciona la opción de "Aceptar".	
5.		Valida el dato insertado.
6.		Notifica mediante un correo electrónico la cancelación a todos los involucrados.
7.		Termina el caso de uso.

Flujos alternos

4.a Presiona el botón "Cancelar"

	Actor	Sistema
1.	Presiona el botón "Cancelar"	
2.		Cancela la acción y cierra la interfaz.
3.		Termina el caso de uso.

5.a. Existen campos vacíos.

1.		Muestra encima del campo vacío el mensaje: "El campo no puede"
----	--	--

		estar vacío”.
2.		Regresa al paso 3 del flujo básico: Cancelar reunión.
Relaciones	CU incluidos	
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		
		

Tabla 8: CU_Replanificar reunión.

Objetivo	Replanificar una reunión.
Actores	Planificador
Resumen	El caso de uso inicia cuando el planificador desea replanificar una reunión. El caso de uso termina cuando realiza esta acción.
Complejidad	Media
Prioridad	Media
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION. La reuniones no están en estado “cancelado” ni “ejecutado”.
Post-condiciones	Se modifica el estado de una reunión a Replanificada. Se notifica mediante un correo electrónico la cancelación a todos los involucrados.
Flujo de eventos	

Flujo básico: Replanificar reunión.		
	Actor	Sistema
1.	Selecciona la acción Replanificar reunión	
2.		Muestra una interfaz de Replanificar reunión. Y permite: <ul style="list-style-type: none"> - Insertar nueva fecha - Aceptar - Cancelar
3.	Inserta nueva fecha de reunión.	
4.	Selecciona la opción de "Aceptar".	
5.		Valida el dato insertado.
6.		Modifica el estado de la reunión a "Replanificada".
7.		Notifica mediante un correo electrónico la replanificación a todos los involucrados.
8.		Termina el caso de uso.
Flujos alternos		
4.a Presiona el botón "Cancelar"		
	Actor	Sistema
1.	Presiona el botón "Cancelar"	
2.		Cancela la acción y cierra la interfaz.
3.		Termina el caso de uso.
5.a. Existen campos vacíos.		
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 del flujo básico: Replanificar reunión.
Relaciones	CU incluidos	
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Replanificar reunión	
Fecha de cumplimiento	
<input type="text" value="MM/DD/YYYY"/>	
<input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/>	

Tabla 9: CU_Generar documento Acta de reunión.

Objetivo	Crear Acta de reunión, editar el acta de reunión en línea.	
Actores	GESTORES_REUNION	
Resumen	El caso de uso inicia cuando el GESTORES_REUNION desea crear acta de reunión o editar acta de reunión en línea. El caso de uso termina cuando realiza una de estas acciones.	
Complejidad	Media	
Prioridad	Media	
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.	
Post-condiciones	El sistema crea a partir de una plantilla el acta de reunión, lo asocia a la reunión en cuestión y establece permiso de Consumidor para todos los involucrados en la reunión. Edita el contenido del documento acta de reunión.	
Flujo de eventos		
Flujo básico: Generar documento Acta de reunión.		
	Actor	Sistema
1.	Crea y edita documento acta de reunión.	
2.		Permite realizar varias acciones con el acta de reunión: <ul style="list-style-type: none"> - Crear documento acta de reunión. Ver sección 1: Crear documento acta de reunión. - Editar documento acta de reunión en línea. Ver sección 2: Editar documento acta de reunión en línea.
Sección 1: Crear documento acta de reunión.		

1.	Selecciona la acción de “Crear documento acta de reunión” de las acciones de la tabla.	
2.		Muestra una interfaz y permite: <ul style="list-style-type: none"> - Aceptar - Cancelar
3.	Crea el documento acta de reunión.	
4.	Selecciona el botón “Aceptar”.	
5.		Crea el acta de reunión a partir de una plantilla.
6.		Asocia el acta de reunión a la reunión en cuestión.
7.		Establece permiso de Consumidor para todos los involucrados en la reunión.
8.		Termina el caso de uso.
Flujos alternos		
2.a Presiona el botón “Cancelar”		
	Actor	Sistema
1.	Presiona el botón “Cancelar”	
2.		Cancela la acción y cierra la interfaz.
3.		Termina el caso de uso.
Sección 2: Editar documento acta de reunión en línea.		
1.	Selecciona la opción de editar documento acta de reunión en línea.	
2.		Muestra una interfaz para editar y permite: <ul style="list-style-type: none"> - Aceptar. - Cancelar.
3.	Edita el contenido del acta.	
4.	Selecciona el botón “Aceptar”.	
5.		Guarda la información editada.
6.		Termina el caso de uso.
Flujos alternos		
4.a Selecciona el botón “Cancelar”		
1.	Presiona el botón “Cancelar”	
2.		Cancela la acción y cierra la interfaz.
3.		Termina el caso de uso.

Relaciones	CU incluidos	
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Tabla 10: CU_Generar documento Orden del día.

Objetivo	Crear documento Orden del día, editar el documento orden del día en línea.	
Actores	GESTORES_REUNION	
Resumen	El caso de uso inicia cuando el GESTORES_REUNION desea crear documento Orden del día o editar el documento Orden del día. El caso de uso termina cuando realiza una de estas acciones.	
Complejidad	Media	
Prioridad	Media	
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.	
Post-condiciones	El sistema crea a partir de una plantilla el documento Orden del día, lo asocia a la reunión en cuestión y establece permiso de Consumidor para todos los involucrados en la reunión. Edita el contenido del documento Orden del día.	
Flujo de eventos		
Flujo básico: Generar documento Orden del día.		
	Actor	Sistema
1.	Crea y edita documento Orden del día.	
2.		Permite realizar varias acciones: <ul style="list-style-type: none"> - Crear documento Orden del día. Ver sección 1: Crear documento Orden del día. - Editar documento Orden del día en línea. Ver sección 2: Editar documento Orden del día en línea.
Sección 1: Crear documento Orden del día.		

1.	Selecciona la acción de "Crear documento Orden del día" de las acciones de la tabla.	
2.		Muestra una interfaz y permite: <ul style="list-style-type: none"> - Aceptar - Cancelar
3.	Crea el documento Orden del día.	
4.	Selecciona el botón "Aceptar".	
5.		Crea el documento Orden del día a partir de una plantilla.
6.		Asocia el acta de reunión a la reunión en cuestión.
7.		Establece permiso de Consumidor para todos los involucrados en la reunión.
8.		Termina el caso de uso.
Flujos alternos		
2.a Presiona el botón "Cancelar"		
	Actor	Sistema
1.	Presiona el botón "Cancelar"	
2.		Cancela la acción y cierra la interfaz.
3.		Termina el caso de uso.
Sección 2: Editar documento acta de reunión en línea.		
1.	Selecciona la opción de editar documento Orden del día en línea.	
2.		Muestra una interfaz para editar y permite: <ul style="list-style-type: none"> - Aceptar. - Cancelar.
3.	Edita el contenido del documento.	
4.	Selecciona el botón "Aceptar".	
5.		Guarda la información editada.
6.		Termina el caso de uso.
Flujos alternos		
4.a Selecciona el botón "Cancelar"		

1.	Presiona el botón "Cancelar"	
2.		Cancela la acción y cierra la interfaz.
3.		Termina el caso de uso.
Relaciones	CU incluidos	
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Tabla 11: CU_ Calcular índice de acuerdos cumplidos por reunión.

Objetivo	Calcular automáticamente el índice de acuerdos cumplidos de una reunión.	
Actores	Planificador	
Resumen	El caso de uso inicia cuando el sistema calcula automáticamente el índice de acuerdos cumplidos de una reunión una vez creado o cerrado un acuerdo. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Media	
Precondiciones	Se crea o se cierra un acuerdo.	
Post-condiciones	Se calculará de los acuerdos adoptados la cantidad según su estado general, así como el por ciento que representan.	
Flujo de eventos		
Flujo básico: Calcular índice de acuerdos cumplidos por reunión.		
	Actor	Sistema
1.	Crea o cierra un acuerdo.	
2.		Calculará de forma general el índice de acuerdos cumplidos IAC

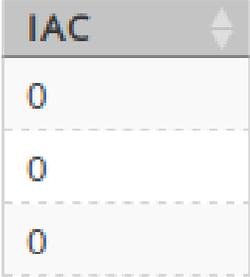
		$(\%) = \frac{TAC}{TA} \times 100$ ² .
3.		Termina el caso de uso.
Relaciones	CU incluidos	N/A
	CU extendidos	CU_Gestionar acuerdos CU_Cerrar acuerdo
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		
		

Tabla 12: CU_Visualizar reuniones asignadas a mí.

Objetivo	Visualizar todas las reuniones en las que es responsable.
Actores	Responsable_Reunión.
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todas las reuniones en las que es responsable. De las mismas se mostrarán los siguientes datos: Nombre, Tipo de reunión, Estado, Motivo, Responsable, Participantes y el IAC. El caso de uso termina cuando realiza esta acción.
Complejidad	Media
Prioridad	Alta
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.
Post-condiciones	Visualiza en la tabla todas las reuniones en las que el usuario es responsable y muestra

²TAC: Total de acuerdos cumplidos en el período; TA: Total de acuerdos a cumplir en el período.

	los siguientes datos: Nombre, Tipo de reunión, Estado, Motivo, Responsable, Participantes y el IAC.	
Flujo de eventos		
Flujo básico: Visualizar reuniones asignadas a mí.		
	Actor	Sistema
1.	Selecciona el filtro "Visualizar reuniones asignadas a mí"	
2.		<p>Visualiza en una tabla todas las reuniones en las que es responsable y muestra los siguientes datos de las mismas:</p> <ul style="list-style-type: none"> - Nombre - Tipo de reunión - Estado, Motivo - Responsable - Participantes - IAC
3.		Termina el caso de uso.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Reuniones asignadas a mí

Año: 2018

Fecha

Desde:

Hasta:

Buscar:

Mostrar 20 elementos

Tipo de reunión	Estado	Motivo	Responsable	Participante	IAC	Acciones
consejo dir	Activa	prueba	admin	admin	0	
Reunión de Coordinación	Activa	dfdfdf	admin	admin	0	
Consejo de asistencia técnica	Activa	dfdfdf	admin	admin	0	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

Tabla 13: CU_Visualizar reuniones creadas por mí.

Objetivo	Visualizar todas las reuniones creadas por el grupo de gestores de reunión al cual pertenece.	
Actores	Gestor_Reunión.	
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todas las reuniones creadas por el. De las mismas se mostrarán los siguientes datos: Nombre, Tipo de reunión, Estado, Motivo, Responsable, Participantes y el IAC. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.	
Post-condiciones	Visualiza en la tabla todas las reuniones creadas por el usuario y muestra los siguientes datos: Nombre, Tipo de reunión, Estado, Motivo, Responsable, Participantes y el IAC.	
Flujo de eventos		
Flujo básico: Visualizar reuniones creadas por mí		
	Actor	Sistema
1.	Selecciona el filtro "Visualizar reuniones creadas por mí"	

2.		Visualiza en una tabla todas las reuniones creadas por el usuario y muestra los siguientes datos de las mismas:
		<ul style="list-style-type: none"> - Nombre - Tipo de reunión - Estado, Motivo - Responsable - Participantes - IAC
3.		Termina el caso de uso.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Reuniones creadas por mí

Año: 2018

Fecha

Desde:

Hasta:

Buscar:

Mostrar 20 elementos

Tipo de reunión	Estado	Motivo	Responsable	Participante	IAC	Acciones
consejo dir	Activa	prueba	admin	admin	0	
Reunión de Coordinación	Activa	dfdfdf	admin	admin	0	
Consejo de asistencia técnica	Activa	dfdfdf	admin	admin	0	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

Mostrando 1 páginas de 1

Anterior 1 Siguiente

Tabla 14: CU_Buscar reuniones

Objetivo	Buscar reuniones a partir de un criterio insertado.	
Actores	Usuario.	
Resumen	El caso de uso inicia cuando el usuario desea buscar reuniones a partir de un criterio insertado. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado.	
Post-condiciones	Devuelve los elementos que coincidan con el criterio insertado en la búsqueda.	
Flujo de eventos		
Flujo básico: Buscar reuniones		
	Actor	Sistema
1.	Selecciona la acción "Buscar".	
2.	Inserta el criterio de búsqueda.	

		Visualiza los elementos en los que coincida parcial o totalmente el criterio insertado con el valor de los metadatos visualizados en la tabla.
3.		Termina el caso de uso.
Flujos alternos		
2.a Inserta criterio de búsqueda incorrecto		
1.		Muestra un mensaje: "No se encuentran resultados que coincidan con la búsqueda".
2.		Continuar en el paso 2 del flujo básico Buscar reuniones.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		
Buscar: <input style="width: 300px; height: 25px; border: 1px solid gray;" type="text"/>		

Tabla 15: CU_Filtrar reuniones por propiedades.

Objetivo	Buscar reuniones a partir de un criterio insertado.
Actores	Usuario.
Resumen	El caso de uso inicia cuando el usuario desea filtrar reuniones por propiedades. El caso de uso termina cuando realiza esta acción.
Complejidad	Media
Prioridad	Alta
Precondiciones	El usuario esta autenticado.
Post-condiciones	El sistema visualizará los elementos en la tabla que coincidan parcial o totalmente con el

	criterio de búsqueda.				
Flujo de eventos					
Flujo básico: Filtrar reuniones por propiedades.					
	Actor	Sistema			
1.	Filtra reuniones por propiedades.				
2.		Permite filtrar las reuniones con las siguientes propiedades: <ul style="list-style-type: none"> - Nombre - Fecha - Tipo de reunión - Estado - Motivo - Responsable - Participantes - Índice de cumplimiento de acuerdos(IAC) 			
3.		Visualiza los elementos en la tabla que coincida parcial o totalmente con el criterio de búsqueda.			
4.		Termina el caso de uso.			
Relaciones	CU incluidos	N/A			
	CU extendidos	N/A			
Requisitos funcionales	no	N/A			
Asuntos pendientes		N/A			
Prototipo elemental de interfaz gráfica de usuario					
Tipo de reunión	Estado	Motivo	Responsable	Participante	IAC
consejo dir	Activa	prueba	admin	admin	0
Reunión de Coordinación	Activa	dfdfdfdf	admin	admin	0
Consejo de asistencia técnica	Activa	dfdfdf	admin	admin	0
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Tabla 16: CU_ Actualizar estado de acuerdo

Objetivo	Actualizar el estado del acuerdo.	
Actores	Responsable_Acuerdo	
Resumen	El caso de uso inicia cuando el usuario desea actualizar estado de acuerdo. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado y es el responsable del acuerdo.	
Post-condiciones	El sistema visualizará los elementos que coincidan parcial o totalmente con el criterio de búsqueda.	
Flujo de eventos		
Flujo básico: Actualizar estado de acuerdo.		
	Actor	Sistema
1.	Selecciona la accion "Actualizar estado de acuerdo".	
2.		Muestra una interfaz para actualizar el estado con los siguientes datos: <ul style="list-style-type: none"> - Estado - Porciento Y permite: <ul style="list-style-type: none"> - Aceptar - Cancelar
3.	Inserta los datos.	
4.	Selecciona el botón "Aceptar"	
5.		Valida los datos
6.		Actualiza el estado del acuerdo en cuestion.
7.		Termina el caso de uso.
Flujos alternos		
4.a Selecciona el botón "Cancelar"		
	Actor	Sistema
1.	Selecciona el botón "Cancelar".	
2.		Cierra la interfaz.
3.		Termina el caso de uso.

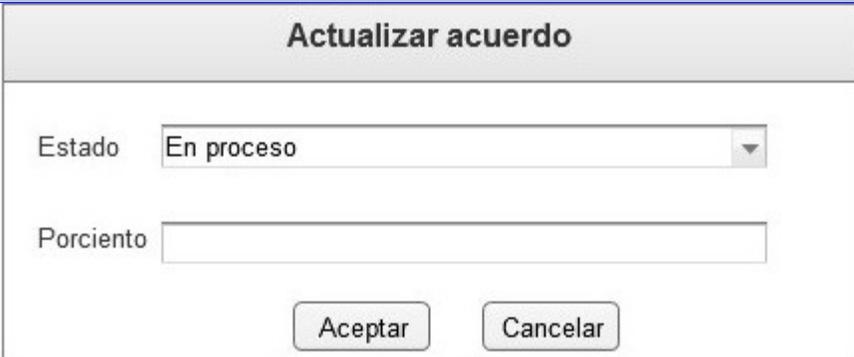
5.a Existen campos vacíos		
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 del flujo básico Actualizar estado de acuerdo.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		
		

Tabla 17: CU_Aplazar acuerdo.

Objetivo	Aplazar un acuerdo.
Actores	Planificador
Resumen	El caso de uso inicia cuando el usuario desea aplazar un acuerdo. Para ello debe insertar la nueva fecha de cumplimiento y el motivo. El caso de uso termina cuando realiza esta acción.
Complejidad	Media
Prioridad	Alta
Precondiciones	El usuario esta autenticado y pertenece al grupo creador del acuerdo.
Post-condiciones	El acuerdo es aplazado, se modifica la fecha de cumplimiento del acuerdo y notificado el responsable del mismo.
Flujo de eventos	

Flujo básico: Aplazar acuerdo.		
	Actor	Sistema
1.	Selecciona la acción "Aplazar acuerdo".	
2.		Muestra una interfaz para aplazar el acuerdo con los siguientes datos: <ul style="list-style-type: none"> - Nueva fecha de cumplimiento - Motivo. Y permite: <ul style="list-style-type: none"> - Aceptar - Cancelar
3.	Inserta los datos.	
4.	Selecciona el botón "Aceptar"	
5.		Valida los datos
6.		Modifica la fecha de cumplimiento del acuerdo.
7.		Notifica al responsable del mismo.
8.		Termina el caso de uso.
Flujos alternos		
4.a Selecciona el botón "Cancelar"		
	Actor	Sistema
1.	Selecciona el botón "Cancelar".	
2.		Cierra la interfaz.
3.		Termina el caso de uso.
5.a Existen campos vacíos		
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 del flujo básico Aplazar acuerdo.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos		N/A

pendientes	
Prototipo elemental de interfaz gráfica de usuario	
<div style="border: 1px solid gray; padding: 10px; margin: 10px auto; width: 60%;"> <div style="background-color: #e0e0e0; padding: 5px; text-align: center; border: 1px solid gray;">Aplazar acuerdo</div> <p>Fecha de cumplimiento</p> <input style="width: 100%;" type="text" value="06/26/2019"/> <p>Motivo</p> <div style="border: 1px solid gray; height: 60px; width: 100%;"></div> <div style="text-align: center; margin-top: 10px;"> <input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/> </div> </div>	

Tabla 18: CU_Detener acuerdo.

Objetivo	Detener un acuerdo.	
Actores	Planificador	
Resumen	El caso de uso inicia cuando el usuario desea detener un acuerdo. Para ello debe insertar un motivo. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autentificado, pertenece al grupo creador del acuerdo y el estado del acuerdo es diferente a: Cumplido, Completado, Cancelado o Incumplido.	
Post-condiciones	El acuerdo es detenido y su estado pasa a Detenido	
Flujo de eventos		
Flujo básico: Detener acuerdo.		
	Actor	Sistema
1.	Selecciona la accion "Detener acuerdo".	
2.		Muestra una interfaz con los siguientes datos: - Motivo

		Y permite:
		<ul style="list-style-type: none"> - Aceptar - Cancelar
3.	Inserta los datos.	
4.	Selecciona el botón "Aceptar"	
5.		Valida los datos
6.		Modifica el estado del acuerdo a "Detenido".
7.		Termina el caso de uso.
Flujos alternos		
4.a Selecciona el botón "Cancelar"		
	Actor	Sistema
1.	Selecciona el botón "Cancelar".	
2.		Cierra la interfaz.
3.		Termina el caso de uso.
5.a Existen campos vacíos		
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 del flujo básico Detener acuerdo.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Tabla 19: CU_Reanudar acuerdo.

Objetivo	Reanudar un acuerdo.	
Actores	Planificador	
Resumen	El caso de uso inicia cuando el usuario desea detener un acuerdo. Para ello debe insertar un motivo. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autentificado, pertenece al grupo creador del acuerdo y el estado del acuerdo es "Detenido".	
Post-condiciones	El acuerdo es reanudado, su estado pasa a "En proceso" el responsable del mismo es notificado.	
Flujo de eventos		
Flujo básico: Reanudar acuerdo.		
	Actor	Sistema
1.	Selecciona la acción "Reanudar acuerdo".	
2.		Muestra una interfaz con los siguientes datos: <ul style="list-style-type: none"> - Fecha de cumplimiento Y permite: <ul style="list-style-type: none"> - Aceptar - Cancelar

3.	Inserta los datos.	
4.	Selecciona el botón "Aceptar"	
5.		Valida los datos
6.		Modifica el estado del acuerdo a "En proceso".
7.		Notifica al responsable del acuerdo.
8.		Termina el caso de uso.

Flujos alternos

4.a Selecciona el botón "Cancelar"

	Actor	Sistema
1.	Selecciona el botón "Cancelar".	
2.		Cierra la interfaz.
3.		Termina el caso de uso.

5.a Existen campos vacíos

	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 del flujo básico Reanudar acuerdo.

Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A

Prototipo elemental de interfaz gráfica de usuario

Reanudar acuerdo

Fecha de cumplimiento

06/13/2019

Tabla 20: CU_Cerrar acuerdo.

Objetivo	Cerrar un acuerdo.	
Actores	Planificador	
Resumen	El caso de uso inicia cuando el usuario desea cerrar un acuerdo. Para ello debe insertar el estado y las Observaciones. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado, pertenece al grupo creador del acuerdo y el estado del acuerdo es "Completado".	
Post-condiciones	El acuerdo es cerrado, si el estado es Cumplido el Porciento será igual a 100, en caso contrario si el estado es Incumplido el Porciento no se modificará.	
Flujo de eventos		
Flujo básico: Cerrar acuerdo.		
	Actor	Sistema
1.	Selecciona la accion "Cerrar acuerdo".	
2.		Muestra una interfaz con los siguientes datos: <ul style="list-style-type: none"> - Estado - Observaciones Y permite: <ul style="list-style-type: none"> - Aceptar - Cancelar
3.	Inserta los datos.	
4.	Selecciona el botón "Aceptar"	
5.		Valida los datos
6.		Termina el caso de uso.
Flujos alternos		
4.a Selecciona el botón "Cancelar"		
	Actor	Sistema
1.	Selecciona el botón "Cancelar".	
2.		Cierra la interfaz.
3.		Termina el caso de uso.
5.a Existen campos vacíos		

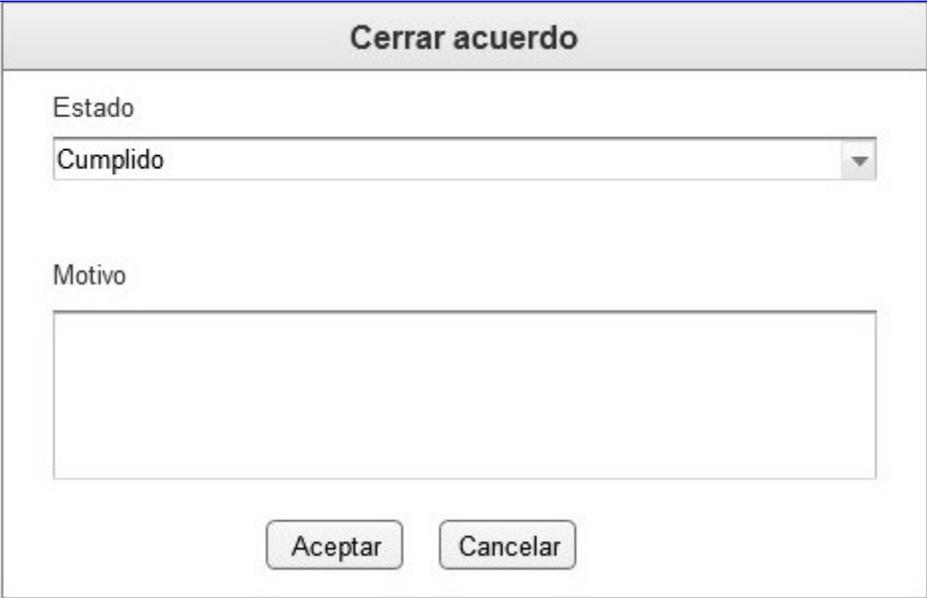
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".
2.		Regresa al paso 3 del flujo básico Cerrar acuerdo.
Relaciones	CU incluidos	CU.Calcular Índice de Acuerdos Cumplidos (IAC)
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		
		

Tabla 21: CU_Cancelar acuerdo

Objetivo	Cancelar acuerdo	
Actores	Planificador	
Resumen	El caso de uso inicia cuando el usuario desea cancelar un acuerdo. Para ello debe insertar las Observaciones. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado, pertenece al grupo creador del acuerdo.	
Post-condiciones	El acuerdo se cancela, el estado cambia a Cancelado.	
Flujo de eventos		
Flujo básico: Cerrar acuerdo.		
	Actor	Sistema
1.	Selecciona la acción "Cancelar acuerdo".	
2.		Muestra una interfaz con los siguientes datos: - Observaciones Y permite: - Aceptar - Cancelar
3.	Inserta los datos.	
4.	Selecciona el botón "Aceptar"	
5.		Valida los datos
6.		Termina el caso de uso.
Flujos alternos		
4.a Selecciona el botón "Cancelar"		
	Actor	Sistema
1.	Selecciona el botón "Cancelar".	
2.		Cierra la interfaz.
3.		Termina el caso de uso.
5.a Existen campos vacíos		
	Actor	Sistema
1.		Muestra encima del campo vacío el mensaje: "El campo no puede estar vacío".

2.	Regresa al paso 3 del flujo básico Cancelar acuerdo.	
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes	N/A	
Prototipo elemental de interfaz gráfica de usuario		
		

Tabla 22: CU_Visualizar acuerdos creados por mí.

Objetivo	Visualizar todos los acuerdos creados por el grupo de gestores de reunión al cual el usuario pertenece.
Actores	Planificador.
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todos los acuerdos creados por el grupo de gestores de reunión al cual pertenece y cuya fecha de emisión sea del año actual. El caso de uso termina cuando realiza esta acción.
Complejidad	Media
Prioridad	Alta
Precondiciones	El usuario esta autenticado y pertenece al grupo GESTOR_REUNION.
Post-condiciones	Del acuerdo se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.

Flujo de eventos		
Flujo básico: Visualizar acuerdos creados por mí.		
	Actor	Sistema
1.	Selecciona el filtro "Visualizar acuerdos creados por mí".	
2.		<p>Visualiza en una tabla todos los acuerdos cuya fecha de emisión sea del año actual con los siguientes metadatos:</p> <ul style="list-style-type: none"> - Acuerdo - Reunión - Contenido - Estado - Fecha de cumplimiento - Responsable
3.		Termina el caso de uso.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Acuerdos creados por mí

Año: 2018

Fecha de cumplimiento

Desde:

Hasta:

Buscar:

Mostrar 20 elementos

Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables	Acciones
A-0002-R-0001-RC-2019	R-0001-RC-2019	Este	31/12/1969	31/12/1969	No comenzado	admin	       
A-0003-R-0001-CD-2019	R-0001-CD-2019	xsfsfsf	31/12/1969	31/12/1969	No comenzado	admin	       
A-0004-R-0001-RC-2019	R-0001-RC-2019	dc	31/12/1969	31/12/1969	No comenzado	admin	       
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

Mostrando 1 páginas de 1

Anterior 1 Siguiente

Tabla 23: CU_Visualizar acuerdos asignados a mí.

Objetivo	Visualizar todos los acuerdos de los cuales es el responsable.
Actores	Responsable_Acuerdo.
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todos los acuerdos de los cuales es el responsable, el estado no sea: Cancelado, Cumplido o Incumplido y cuya fecha de emisión sea del año actual. El caso de uso termina cuando realiza esta acción.
Complejidad	Media
Prioridad	Alta

Precondiciones	El usuario esta autenticado y es participante del acuerdo.	
Post-condiciones	Del acuerdo se visualizarán los siguientes metadatos en la tabla: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	
Flujo de eventos		
Flujo básico: Visualizar acuerdos asignados a mí.		
	Actor	Sistema
1.	Selecciona el filtro "Visualizar acuerdos asignados a mí".	
2.		<p>Visualiza en una tabla todos los acuerdos cuya fecha de emisión sea del año actual con los siguientes metadatos:</p> <ul style="list-style-type: none"> - Acuerdo - Reunión - Contenido - Estado - Fecha de cumplimiento - Responsable
3.		Termina el caso de uso.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Acuerdos asignados a mí

Año: 2018 Buscar

Fecha de cumplimiento

Desde: x

Hasta: x

Buscar:

Mostrar 20 elementos

Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables	Acciones
A-0002-R-0001-RC-2019	R-0001-RC-2019	Este	31/12/1969	31/12/1969	No comenzado	admin	
A-0003-R-0001-CD-2019	R-0001-CD-2019	xsfsfsf	31/12/1969	31/12/1969	No comenzado	admin	
A-0004-R-0001-RC-2019	R-0001-RC-2019	dc	31/12/1969	31/12/1969	No comenzado	admin	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

Tabla 24: CU_Visualizar acuerdos en los que participo.

Objetivo	Visualizar todos los acuerdos de los cuales es el participante.	
Actores	Participantes.	
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todos los acuerdos de los cuales es participante, el estado no sea: Cancelado, Cumplido o Incumplido y cuya fecha de emisión sea del año actual. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado y es el responsable del acuerdo.	
Post-condiciones	Del acuerdo se visualizarán los siguientes metadatos en la tabla: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	
Flujo de eventos		
Flujo básico: Visualizar acuerdos en los que participo.		
	Actor	Sistema
1.	Selecciona el filtro "Visualizar acuerdos"	

	en los que participo".		
2.			<p>Visualiza en una tabla todos los acuerdos cuya fecha de emisión sea del año actual con los siguientes metadatos:</p> <ul style="list-style-type: none"> - Acuerdo - Reunión - Contenido - Estado - Fecha de cumplimiento - Responsable
3.			Termina el caso de uso.
Relaciones	CU incluidos	N/A	
	CU extendidos	N/A	
Requisitos funcionales	no	N/A	
Asuntos pendientes		N/A	
Prototipo elemental de interfaz gráfica de usuario			

Acuerdos en los que participo

Año:

Fecha de cumplimiento

Desde:

Hasta:

Buscar:

Mostrar elementos

Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables	Acciones
No se encontraron resultados.							
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
No se encontraron resultados.							

Anterior Siguiente

Tabla 25: CU_Visualizar acuerdos aplazados.

Objetivo	Visualizar todos los acuerdos que tienen aspecto aplazados.	
Actores	Usuario.	
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todos los acuerdos que tienen aspecto aplazado cuya fecha de cumplimiento sea del mes actual. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado.	
Post-condiciones	Del acuerdo se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	
Flujo de eventos		
Flujo básico: Visualizar acuerdos aplazados.		
	Actor	Sistema
1.	Selecciona el filtro "Visualizar acuerdos aplazados".	

2.		<p>Visualiza en una tabla todos los acuerdos de cumplimiento sea del mes actual con los siguientes metadatos:</p> <ul style="list-style-type: none"> - Acuerdo, - Reunión, - Contenido, - Estado, - Fecha de cumplimiento - Responsable.
----	--	--

3. Termina el caso de uso.

Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A

Prototipo elemental de interfaz gráfica de usuario

Acuerdos aplazados

Año:

Fecha de cumplimiento

Desde:

Hasta:

Buscar:

Mostrar elementos

Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables	Acciones
No se encontraron resultados.							
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
No se encontraron resultados.							

Anterior Siguiente

Tabla 26: CU_Visualizar acuerdos por expirar.

Objetivo	Visualizar todos los acuerdos cuya fecha de cumplimiento estén en la semana actual	
Actores	Participantes.	
Resumen	El caso de uso inicia cuando el usuario desea visualizar todos los acuerdos cuya fecha de cumplimiento estén en la semana actual. De los mismos se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado.	
Post-condiciones	Del acuerdo se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	
Flujo de eventos		
Flujo básico: Visualizar acuerdos por expirar.		
	Actor	Sistema
1.	Selecciona el filtro "Visualizar acuerdos por expirar".	
2.		<p>Visualiza en una tabla todos los acuerdos cuya fecha de cumplimiento estén en la semana actual con los siguientes metadatos:</p> <ul style="list-style-type: none"> - Acuerdo, - Reunión, - Contenido, - Estado, - Fecha de cumplimiento - Responsable.
3.		Termina el caso de uso.
Relaciones	CU incluidos	N/A

	CU extendidos	N/A					
Requisitos funcionales	no	N/A					
Asuntos pendientes		N/A					
Prototipo elemental de interfaz gráfica de usuario							
Acuerdos por expirar							
Año:	2018	<input type="button" value="Buscar"/>					
Fecha de cumplimiento							
Desde:	<input type="text"/>	<input type="button" value="X"/>					
Hasta:	<input type="text"/>	<input type="button" value="X"/>					
Buscar:	<input type="text"/>	Mostrar <input type="text" value="20"/> elementos					
Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables	Acciones
No se encontraron resultados.							
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
No se encontraron resultados.							
Anterior <input type="button" value=""/> <input type="button" value=""/> Siguiente							

Tabla 27: CU_ Visualizar acuerdos pendientes de revisión.

Objetivo	Visualizar todos los acuerdos que su estado sea Completado.
Actores	Planificador
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todos los acuerdos que su estado sea Completado. De los mismos se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable. El caso de uso termina cuando realiza esta acción.
Complejidad	Media

Prioridad	Alta	
Precondiciones	El usuario esta autenticado y pertenece al grupo creador del acuerdo.	
Post-condiciones	Del acuerdo se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	
Flujo de eventos		
Flujo básico: Visualizar acuerdos pendientes de revisión.		
	Actor	Sistema
1.	Selecciona el filtro “Visualizar acuerdos pendientes de revisión”.	
2.		<p>Visualiza en una tabla todos los acuerdos que su estado sea Completado con los siguientes metadatos:</p> <ul style="list-style-type: none"> - Acuerdo, - Reunión, - Contenido, - Estado, - Fecha de cumplimiento - Responsable.
3.		Termina el caso de uso.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A
Prototipo elemental de interfaz gráfica de usuario		

Acuerdos pendientes de revisión

Año:

Fecha de cumplimiento

Desde:

Hasta:

Buscar:

Mostrar elementos

Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables	Acciones
No se encontraron resultados.							
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
No se encontraron resultados.							

Anterior

Tabla 28: CU_Visualizar acuerdos expirados

Objetivo	Visualizar todos los acuerdos que su fecha de cumplimiento haya sido expirada.	
Actores	Usuario	
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todos los acuerdos que su fecha de cumplimiento haya sido expirada. De los mismos se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado.	
Post-condiciones	Del acuerdo se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	
Flujo de eventos		
Flujo básico: Visualizar acuerdos pendientes de revisión.		
	Actor	Sistema
1.	Selecciona el filtro "Visualizar acuerdos expirados".	
2.		Visualiza en una los siguientes metadatos del acuerdo:

		<ul style="list-style-type: none"> - Acuerdo, - Reunión, - Contenido, - Estado, - Fecha de cumplimiento - Responsable.
--	--	--

3. Termina el caso de uso.

Relaciones	CU incluidos	N/A
-------------------	---------------------	-----

	CU extendidos	N/A
--	----------------------	-----

Requisitos funcionales	no	N/A
-------------------------------	-----------	-----

Asuntos pendientes	N/A
---------------------------	-----

Prototipo elemental de interfaz gráfica de usuario

Acuerdos expirados

Año:

Fecha de cumplimiento

Desde:

Hasta:

Buscar:

Mostrar elementos

Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables	Acciones
No se encontraron resultados.							
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

No se encontraron resultados.

Anterior Siguiente

Tabla 29:CU_ Visualizar acuerdos pendientes

Objetivo	Visualizar todos los acuerdos pendientes	
Actores	Usuario	
Resumen	El caso de uso inicia cuando el usuario desea visualizar en la tabla todos los acuerdos pendientes. De los mismos se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado.	
Post-condiciones	Del acuerdo se visualizarán los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	
Flujo de eventos		
Flujo básico: Visualizar acuerdos pendientes de revisión.		
	Actor	Sistema
1.	Selecciona el filtro “Visualizar acuerdos pendientes”.	
2.		Visualiza en una tabla los siguientes metadatos del acuerdo: <ul style="list-style-type: none"> - Acuerdo, - Reunión, - Contenido, - Estado, - Fecha de cumplimiento - Responsable.
3.		Termina el caso de uso.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos	N/A	

pendientes

Prototipo elemental de interfaz gráfica de usuario

Acuerdos pendientes

Año: 2018

Fecha de cumplimiento

Desde:

Hasta:

Buscar:

Mostrar 20 elementos

Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables	Acciones
No se encontraron resultados.							
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
No se encontraron resultados.							

Anterior Siguiente

Tabla 30: CU_Filtrar acuerdos por propiedades.

Objetivo	Buscar acuerdos a partir de un criterio insertado.	
Actores	Usuario.	
Resumen	El caso de uso inicia cuando el usuario desea filtrar acuerdos por propiedades. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado.	
Post-condiciones	El sistema visualizará los elementos en la tabla que coincidan parcial o totalmente con el criterio de búsqueda.	
Flujo de eventos		
Flujo básico: Filtrar acuerdos por propiedades.		
	Actor	Sistema
1.	Filtra acuerdos por propiedades.	

2.		Permite filtrar los acuerdos con las siguientes propiedades: <ul style="list-style-type: none"> - Nombre de acuerdo - Contenido - Fecha de emisión - Fecha de cumplimiento - Estado - Responsable
3.		Visualiza los elementos en la tabla que coincida parcial o totalmente con el criterio de búsqueda.
4.		Termina el caso de uso.

Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes		N/A

Prototipo elemental de interfaz gráfica de usuario						
Acuerdo	Reunión	Contenido	Fecha de emisión	Fecha de cumplimiento	Estado	Responsables
No se encontraron resultados.						
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Tabla 31: CU_Buscar acuerdos

Objetivo	Buscar acuerdos a partir de un criterio insertado.	
Actores	Usuario.	
Resumen	El caso de uso inicia cuando el usuario desea buscar acuerdos a partir de un criterio insertado. El caso de uso termina cuando realiza esta acción.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario esta autenticado.	
Post-condiciones	Devuelve los elementos que coincidan con el criterio insertado en la búsqueda.	
Flujo de eventos		
Flujo básico: Buscar acuerdos		
	Actor	Sistema
1.	Selecciona la acción "Buscar".	
2.	Inserta el criterio de búsqueda.	
		Visualiza los elementos en los que coincida parcial o totalmente el criterio insertado con el valor de los metadatos visualizados en la tabla.
3.		Termina el caso de uso.
Flujos alternos		
2.a Inserta criterio de búsqueda incorrecto		
1.		Muestra un mensaje: "No se encuentran resultados que coincidan con la búsqueda".
2.		Continuar en el paso 2 del flujo básico Buscar acuerdos.
Relaciones	CU incluidos	N/A
	CU extendidos	N/A
Requisitos funcionales	no	N/A
Asuntos pendientes	N/A	
Prototipo elemental de interfaz gráfica de usuario		
Buscar: <input type="text"/>		

Anexo 2. Matrices de trazabilidad.

	Actualizar estado de acuerdo	Aplazar acuerdo	Buscar reuniones	Calcular Índice de acuerdos cu...	Cancelar acuerdo	Cancelar reunión	Cerrar acuerdo	Crear acuerdo	Crear reunión	Detener acuerdo	Editar propiedades de acuerdo	Editar propiedades de reunión	Eliminar acuerdo	Filtrar acuerdo por reunión	Reanudar acuerdo	Replanificar reunión	Ver detalles de acuerdo	Ver detalles de reunión	Visualizar acuerdos aplazados	Visualizar acuerdos asignados ...	Visualizar acuerdos creados po...	Visualizar acuerdos en los que ...	Visualizar acuerdos expirados	Visualizar acuerdos pendientes	Visualizar acuerdos pendientes...	Visualizar acuerdos por expirar	Visualizar reuniones asignadas...	Visualizar reuniones creadas P...		
Actualizar estado de acuerdo								✓																						
Aplazar acuerdo		✓						✓																						
Buscar reuniones			✓						✓																					
Calcular Índice de acuerdos cumplidos por reunión				✓			✓																							
Cancelar acuerdo					✓			✓																						
Cancelar reunión						✓			✓																					
Cerrar acuerdo				✓			✓																							
Crear acuerdo	✓	✓		✓	✓		✓			✓	✓						✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Crear reunión			✓			✓						✓				✓		✓											✓	✓
Detener acuerdo							✓																							
Editar propiedades de acuerdo							✓																							
Editar propiedades de reunión									✓																					
Eliminar acuerdo							✓																							
Filtrar acuerdo por reunión							✓																							
Reanudar acuerdo							✓																							
Replanificar reunión									✓																					
Ver detalles de acuerdo							✓																							
Ver detalles de reunión									✓																					
Visualizar acuerdos aplazados							✓																							
Visualizar acuerdos asignados a mí							✓																							
Visualizar acuerdos creados por mí							✓																							
Visualizar acuerdos en los que participo							✓																							
Visualizar acuerdos expirados							✓																							
Visualizar acuerdos pendientes							✓																							
Visualizar acuerdos pendientes de revisión							✓																							
Visualizar acuerdos por expirar							✓																							
Visualizar reuniones asignadas a mí								✓																						
Visualizar reuniones creadas por mí								✓																						

Figura 18: Matriz de trazabilidad de RF-RF

(21) Use Case

By: Transitor

(21) Test Case

Use Case	Actualizar estado del acuerdo	Aplazar acuerdo	Buscar acuerdos	Buscar reuniones	Cancelar reunión	Cerrar acuerdo	Detener acuerdo	Filtrar acuerdos por propiedades	Filtrar reuniones por propiedades	Gestionar Reuniones	Gestionar acuerdo	Reanudar acuerdo	Replanificar reunión	Ver detalles de reunión	Visualizar acuerdo pendiente de revisión	Visualizar acuerdos aplazados	Visualizar acuerdos creados por mí	Visualizar acuerdos en los que participo	Visualizar acuerdos por expirar	Visualizar reuniones asignadas a mí	Visualizar reuniones creadas por mí
Visualizar reuniones creadas por mí.																					✓
Actualizar estado de acuerdo	✓																				
Aplazar acuerdo.		✓																			
Buscar acuerdos			✓																		
Buscar reuniones.				✓																	
Cancelar reunión					✓																
Cerrar acuerdo						✓															
Detener acuerdo							✓														
Filtrar acuerdos por propiedades.								✓													
Filtrar reuniones por propiedades									✓												
Gestionar acuerdo										✓											
Gestionar reuniones											✓										
Reanudar acuerdo												✓									
Replanificar reunión													✓								
Ver de talles de reunión														✓							
Visualizar acuerdos aplazados.															✓						
Visualizar acuerdos creados por mí.																✓					
Visualizar acuerdos en los que participo.																	✓				
Visualizar acuerdos pendientes de revisión.														✓							
Visualizar acuerdos por expirar.																		✓			
Visualizar reuniones asignadas a mí.																			✓		

Figura 19: Matriz de trazabilidad CU_CP

(45) Requirement

By: Transitor

(5) Class

Requirement	Acta	Acuerdo	Gestores_Reunión	Orden del día	Reunión
Actualizar estado de acuerdo	✓				
Aplazar acuerdo	✓				
Buscar acuerdos					
Buscar reuniones			✓		
Calcular índice de acuerdos completados por reunión			✓		
Cancelar acuerdo					
Cancelar reunión					
Cerrar acuerdo					
Crear acuerdo		✓			
Crear documento Acta de reunión					
Crear documento Orden del día				✓	
Crear reunión					
Detener acuerdo					✓
Editar propiedades de acuerdo					
Eliminar reunión					✓
Eliminar acuerdo					
Filtrar acuerdo por contenido					
Filtrar acuerdo por estado					
Filtrar acuerdo por fecha de cumplimiento					
Filtrar acuerdo por fecha de emisión					
Filtrar acuerdo por responsables					
Filtrar acuerdo por reunión					
Filtrar acuerdo por nombre de acuerdo					
Filtrar reuniones por fecha					
Filtrar reuniones por nombre					
Filtrar reuniones por tipo de reunión					
Filtrar reunión por estado					
Filtrar reunión por motivo					
Filtrar reunión por participantes					
Filtrar reunión por responsable					
Filtrar reunión por índice de cumplimiento de acuerdos					
Reanudar acuerdo					
Replanificar reunión					
Ver detalles de acuerdo					
Ver detalles de reunión					
Visualizar acuerdos aplazados					
Visualizar acuerdos asignados a mí					
Visualizar acuerdos creados por mí					
Visualizar acuerdos en los que participo					
Visualizar acuerdos expirados					
Visualizar acuerdos pendientes					
Visualizar acuerdos pendientes de revisión					
Visualizar acuerdos por expirar					
Visualizar reuniones asignadas a mí					
Visualizar reuniones creadas por mí					

Figura 20: Matriz de trazabilidad RF-MD

Anexo 3. Casos de prueba

Tabla 32: CP_ Ver detalles de reunión

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Ver detalles de reunión	Permite al usuario visualizar las propiedades de una reunión.	Muestra una interfaz con todos los detalles de la reunión.	1-Selecciona en la tabla de datos la acción ver detalles de reunión 2-Presiona el botón de aceptar.

Tabla 33: CP_ Cancelar reunión

Escenarios	Descripción	Causa	Respuesta del sistema	Flujo central
EC 1.1 Cancelar reunión con datos correctos	Permite al usuario Cancelar una reunión rellenando con la causa el campo de texto. .	V Falta de persona I	Notifica mediante un correo electrónico la cancelación a todos los involucrados.	1-Selecciona en la tabla la acción de Cancelar reunión 2-introduce la causa de cancelación y presiona el botón de aceptar

EC 1.2 Cancelar reunión con campos vacíos	El usuario deja campos vacíos en el formulario de inicio de sesión	-	Muestra el mensaje: "Existen campos vacíos"	1- Selecciona la acción de Cancelar reunión. 2- Deja el campo vacío y Presionar aceptar.
---	--	---	---	---

Tabla 34:CP_ Replanificar reunión

Escenarios	Descripción	Fecha	Respuesta del sistema	Flujo central
EC 1.1 Replanificar reunión con datos correctos	Permite al usuario modificar la fecha de una reunión proporcionando la fecha correctamente.	V	Se modifica el estado de una reunión a Replanificada. Notifica mediante un correo electrónico la replanificación a todos los involucrados.	1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona la acción de replanificar reunión. 3- Inserta la nueva fecha de la reunión. 4- Presiona el botón aceptar.
EC 1.2 Replanificar reunión con datos incorrectos.	El usuario introduce la fecha incorrecta en el formulario de inicio de sesión.	I	Muestra un mensaje de error: "La fecha insertada es incorrecta".	1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona la acción de replanificar reunión. 3- Introduce una fecha incorrecta 4- Presiona aceptar.

EC 1.3 Replanificar reunión con campos vacíos	El usuario deja campos vacíos en el formulario de inicio de sesión	-	Muestra el mensaje: "Existen campos vacíos"	1- Selecciona la acción de replanificar reunión 2- Deja el campo vacío y Presionar aceptar.
EC 1.4 Cancelar acción	Permite al usuario cancelar la acción.	-	El sistema muestra la página de origen.	1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona la acción de replanificar reunión. 3- Introduce una fecha. 4- Presiona cancelar.

Tabla 35: CP_ Visualizar reuniones asignadas a mí.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Visualizar reuniones asignadas a mí.	Permite al usuario visualizar todas las reuniones asignadas a él.	El sistema visualiza en la tabla todas las reuniones en las que el usuario es	1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar reuniones asignadas a mí.

		responsable	
--	--	-------------	--

Tabla 36: CP_ Visualizar reuniones creadas por mí.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Visualizar reuniones creadas por mí.	Permite al usuario visualizar todas las reuniones creadas por él.	El sistema visualiza en la tabla todas las reuniones creadas por el usuario.	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar reuniones creadas por mí.

Tabla 37:CP_ Buscar reuniones.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Buscar reuniones.	Permite al usuario buscar reuniones a partir de un criterio insertado.	Devuelve los elementos que coincidan con el criterio insertado en la búsqueda.	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona la opción de buscar reuniones. 3- Inserta el criterio de búsqueda.

EC 1.2 Inserta criterio de búsqueda incorrecto.	El usuario introduce un criterio de búsqueda incorrecto.	Muestra un mensaje de error: "No se encuentran resultados que coincidan con la búsqueda".	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona la opción de buscar reuniones. 3- Inserta el criterio de búsqueda
---	--	---	---

Tabla 38: CP_ Buscar acuerdos

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Buscar acuerdos.	Permite al usuario buscar acuerdos a partir de un criterio insertado.	Devuelve los elementos que coincidan con el criterio insertado en la búsqueda.	<ol style="list-style-type: none"> 1. El usuario accede al módulo Gestion de acuerdos. 2. Selecciona los filtros de acuerdos 3. Selecciona la opción de buscar acuerdos. 4. Inserta el criterio de búsqueda.

EC 1.2 Inserta criterio de búsqueda incorrecto.	El usuario introduce un criterio de búsqueda incorrecto.	Muestra un mensaje de error: "No se encuentran resultados que coincidan con la búsqueda".	<ol style="list-style-type: none"> 1. El usuario accede al módulo Gestion de acuerdos. 2. Selecciona los filtros de acuerdos 3. Selecciona la opción de buscar acuerdos. 4. Inserta el criterio de búsqueda.
---	--	---	--

Tabla 39: CP_Filtrar reuniones por propiedades.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Filtrar reuniones por propiedades.	Permite al usuario filtrar reuniones por un criterio insertado	El sistema visualizará los elementos en la tabla que coincidan parcial o totalmente con el criterio de búsqueda	<ol style="list-style-type: none"> 1. El usuario accede al módulo Gestion de acuerdos. 2. Filtra reuniones por propiedades.

Tabla 40: CP_Filtrar acuerdos por propiedades.

Escenarios	Descripción	Respuesta del sistema	Flujo central
------------	-------------	-----------------------	---------------

EC 1.1 Filtrar acuerdo por propiedades.	Permite al usuario filtrar acuerdos por un criterio insertado	El sistema visualizará los elementos en la tabla que coincidan parcial o totalmente con el criterio de búsqueda	<ol style="list-style-type: none"> 1. El usuario accede al módulo Gestion de acuerdos. 2. Filtra acuerdos por propiedades.
--	---	---	--

Tabla 41: CP_Visualizar acuerdos creados por mí.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Visualizar acuerdos creados por mí.	Permite al usuario visualizar todos los acuerdos creados por mí.	El sistema visualiza en la tabla todos los acuerdos creados por el usuario.	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar acuerdos creados por mí.

Tabla 42: CP_Visualizar acuerdos asignados a mí.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Visualizar acuerdos asignados a mí.	Permite al usuario visualizar todos los acuerdos asignados a mí.	El sistema visualiza en la tabla todos los acuerdos en los cuales el usuario es el responsable.	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar acuerdos asignados a mí.

Tabla 43: CP_ Visualizar acuerdos en los que participo.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Visualizar acuerdos en los que participo.	Permite al usuario visualizar todos los acuerdos asignados a mí.	El sistema visualiza en la tabla todos los acuerdos en los cuales el usuario es participante	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar acuerdos en los que participo.

Tabla 44: CP_ Visualizar acuerdos prorrogados.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Visualizar acuerdos prorrogados.	Permite al usuario visualizar todos los acuerdos que tienen aspecto prorrogado.	El sistema visualiza de los acuerdos en la tabla los siguientes metadatos en las tabla: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	<ol style="list-style-type: none"> 1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar acuerdos prorrogados.

Tabla 45: CP_ Visualizar acuerdos aplazados.

Escenarios	Descripción	Respuesta del sistema	Flujo central
------------	-------------	-----------------------	---------------

EC 1.1 Visualizar acuerdos aplazados.	Permite al usuario visualizar todos los acuerdos que tienen aspecto aplazado.	El sistema visualiza los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar acuerdos aplazados.
--	---	--	---

Tabla 46: Visualizar acuerdos por expirar.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Visualizar acuerdos por expirar.	Permite al usuario visualizar todos los acuerdos cuya fecha de cumplimiento estén en la semana actual	El sistema visualiza los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar acuerdos por expirar

Tabla 47: CP_ Visualizar acuerdos pendientes de revisión.

Escenarios	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Visualizar acuerdos pendientes de revisión.	Permite al usuario visualizar todos los acuerdos que su estado sea Completado.	El sistema visualiza los siguientes metadatos: Acuerdo, Reunión, Contenido, Estado, Fecha de cumplimiento y Responsable.	1- El usuario accede al módulo Gestion de acuerdos. 2- Selecciona el filtro Visualizar acuerdos pendientes de revisión.

