

Universidad de las Ciencias Informáticas



TÍTULO DE LA TESIS

MÓDULO DE GESTIÓN DOCUMENTAL DEL SISTEMA DE GESTIÓN DE LA
INFORMACIÓN Y CONTROL ESTADÍSTICO DE LA DIRECCIÓN DE
TECNOLOGÍAS Y SISTEMAS.

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE
INGENIERO EN CIENCIAS INFORMÁTICAS



Autor:

Eduardo José Figueredo Silva

Tutores:

Ing. José Miguel Reyes

La Habana, 2020



Dirección de
Tecnologías y
Sistemas

Declaración de autoría

Declaro ser autor de la presente tesis y reconozco a la Dirección de Tecnologías y Sistemas (DTS), los derechos patrimoniales de la misma, con carácter exclusivo.

Y para que así conste, firmo la presente declaración de autoría en La Habana a los ____ días del mes de _____ del año ____

Eduardo José Figueredo Silva

Ing. José Miguel Reyes



Dirección de
Tecnologías y
Sistemas

Resumen

El Órgano de Cuadro y Personal gestiona toda la información referente al personal de pregrado en la Dirección de Tecnologías y Sistemas del Ministerio del Interior, área donde la información es almacenada y tramitada en documentos y expedientes físicos que constituyen activos de la entidad. Sin embargo, dicha área carece de un sistema informatizado que respalde el proceso de gestión documental. La presente investigación propone el desarrollo de un módulo de gestión documental para el Sistema de Gestión de la Información y Control Estadístico de dicha institución atendiendo a las necesidades del área y al proceso de informatización de la sociedad cubana.

Para llevar a cabo el trabajo se realiza un diseño que cumple con los parámetros establecidos en la arquitectura y se emplean las herramientas y tecnologías seleccionadas para la implementación. Además, se realiza un período de pruebas que validan el funcionamiento y la integración del módulo asegurando que el mismo cumpla con las normas vigentes para el desarrollo y lograr el objetivo propuesto.

La implementación de esta herramienta brinda la posibilidad a los trabajadores del Órgano de Cuadro y Personal de contar con un sistema útil que contribuya a mejorar la eficiencia en el apretado proceso de gestión documental que caracteriza al ambiente empresarial.

Palabras clave: gestión documental, módulo, expediente, ambiente empresarial.



Abstract

The Cadre and Staff Organ manages all the information regarding undergraduate personnel in the Directorate of Technologies and Systems of the Ministry of the Interior; area where the information is stored and processed in documents and physical records that constitute assets in the entity. However, this area lacks a computerized system to support the document management process. This research proposes the development of a document management module for the Information Management System and Statistical Control of said institution, attending to the needs of the area and the computerization process of Cuban society.

To carry out the work, a design is made that meets the parameters established in the architecture and the tools and technologies selected for implementation are used. In addition, a period of tests is carried out and integration of the module, ensuring that it complies with current standards for development and achieving the proposed objective.

The implementation of this tool offers the possibility to the workers of the Cadre and Staff Organ to have a useful system that contributes to improving the efficiency in the tight document management process that characterizes the business environment.

Keywords: document management, module, records, business environment.



Índice

Introducción	1
Capítulo 1: Fundamentación teórica.	6
1.1 Conceptos asociados al dominio del problema	6
1.2 Estudio de homólogos	11
1.3 Metodología de desarrollo de software	14
1.4 Herramientas y tecnologías utilizadas	16
1.5 Conclusiones del capítulo	20
Capítulo 2: Análisis y diseño del módulo de gestión documental para el Sistema de Gestión de la Información y Control Estadístico de la DTS.	22
2.1 Reglas del negocio	22
2.3 Flujo actual del proceso	23
2.4 Propuesta de solución	25
2.5 Fase de exploración	26
2.6 Fase de Planificación	31
2.7 Diseño	36
2.8 Conclusiones del capítulo	45
Capítulo 3: Implementación y pruebas	47
3.1 Estándar de codificación	47
3.2 Implementación	48
3.3 Pantallas principales de la aplicación	49
3.4 Pruebas	51
3.5 Resultado de las pruebas	57
3.6 Conclusiones del capítulo	58



**Dirección de
Tecnologías y
Sistemas**

Conclusiones generales:	59
Recomendaciones:	60
Referencias bibliográficas:	61
Glosario de términos:	64
Anexos:	65



Índice de Tablas

Tabla 1: Operacionalización de la variable dependiente.....	3
Tabla 2: Análisis de los antecedentes.....	13
Tabla 3: HU Autenticar usuario.....	29
Tabla 4: HU Modificar usuario.....	29
Tabla 5: HU Listar registro.....	30
Tabla 6: Personal relacionado con el sistema.....	30
Tabla 7: Prioridad de las Historias de Usuario.....	32
Tabla 8: Estimación del esfuerzo para las HU.....	33
Tabla 9: Cronograma de liberación.....	34
Tabla 10: Primera iteración.....	35
Tabla 11: Segunda iteración.....	36
Tabla 12: Tercera iteración.....	36
Tabla 13: Tarjeta CRC de la clase DocumentoController.....	44
Tabla 14: Tarjeta CRC de la clase Documento.....	44
Tabla 15: Tarjeta CRC de la clase Documento_repository.....	44
Tabla 16: Descripción del caso de prueba número 01 a la HU_6.....	52
Tabla 17: Descripción del caso de prueba número 02 a la HU_7.....	52
Tabla 18: Caminos básicos.....	55
Tabla 19: Caso de prueba del camino básico Nro.1.....	55
Tabla 20: Caso de prueba del camino básico Nro.2.....	56
Tabla 21: Caso de prueba de Integración.....	56



Introducción

En el mundo actual la ciencia y la tecnología han llegado a ser irremplazables para disímiles sectores de la sociedad. En estos sectores, específicamente empresas y organizaciones, el uso y explotación de la información es prácticamente indispensable para el manejo de los sistemas y sus respuestas.

En Cuba, los nuevos contextos imponen a las organizaciones la necesidad vital de disponer de adecuadas infraestructuras de comunicación y de sistemas de información que les permitan obtener los datos necesarios a fin de conseguir un conocimiento real y suficiente del entorno que afecte a los procesos de planificación y toma de decisiones. En las organizaciones cubanas, en ocasiones, existe una dispersión de datos e información que duplica tareas y hace imposible su recuperación por parte de quienes la necesitan en el momento oportuno; presentando un buen número de ellas estructuras obsoletas e inadecuadas para atender, con suficiencia, los requerimientos actuales.

No obstante a estos problemas, se esfuerzan por transformarse cada día según los medios que disponen. La gestión eficaz de los recursos de información y documentación en las empresas, genera la búsqueda de soluciones prácticas y exitosas. Este postulado presupone cambios que conllevan a la obtención de mejoras continuas, así como mecanismos y procesos para el desarrollo de la gestión de documentos.

Las empresas cubanas en los momentos actuales se encuentran en procesos de renovación con la introducción de políticas y directivas que parten del criterio de, sin información no se puede dirigir, pero para dirigir, no hace falta "cualquier información", o "muchísima información", sino la que verdaderamente resulte útil para tomar decisiones, por lo que cada empresa deberá hacer un estudio de la información necesaria, teniendo en cuenta los requerimientos del nivel superior (1).

La Dirección de Tecnologías y Sistemas del Ministerio del Interior de la República de Cuba es una de estas empresas; la cual ha decidido emprender, como parte del proceso de informatización de la sociedad cubana, la modernización de sus procesos mediante el uso de tecnologías de avanzada y la implementación de un Sistema de Gestión de la Información y Control Estadístico, en adelante (SGICE), con la finalidad de elevar la eficacia organizacional y el perfeccionamiento de la toma de decisiones.

La decisión de dicha empresa persigue, al igual que el proceso de informatización de la sociedad cubana, la utilización ordenada y masiva de las tecnologías de la información y las comunicaciones (TIC) para lograr



más eficacia y eficiencia, que permitan una mayor generación de riquezas y hagan sustentable el aumento sistemático de la calidad de vida de los ciudadanos.

Sin embargo, como parte de la implementación de este sistema informático, se hace necesario el desarrollo de un módulo de Gestión Documental para el Órgano de Cuadro y Personal de dicha institución, ya que en esta área de la empresa, donde el manejo de la información relativa al personal de pregrado juega un papel fundamental, no cuenta con una solución informática que tribute a la toma de decisiones a través de la elaboración y control de planes de actividades de las diferentes estructuras que atienden, así como la emisión de evaluaciones referentes a los cadetes insertados y propios y la gestión de expedientes de los mismos. Caracterizándose dichos procesos por:

- ✓ La confección y gestión semi informatizada de los planes de actividades correspondientes a las estructuras que atiende el Órgano de Cuadro y Personal, en adelante (OCP).
- ✓ La gestión manual de un número significativo de documentos, entre ellos expedientes, con la información relativa a los cadetes insertados y propios que atiende la Dirección de Tecnologías y Sistemas, en adelante (DTS).
- ✓ Almacenamiento de la información en lotes de documentos físicos y archivos informáticos -de forma semi informatizada- en formato xlsx.

Los aspectos antes mencionados llevan a determinar la **situación problemática** que se presenta en el OCP donde la gestión documental es realizada de forma manual, impidiendo el acceso rápido y efectivo a la información, tornándose el archivado de los documentos y gestión de los expedientes de los cadetes propios e insertados un proceso engorroso que conduce a la pérdida de información y dificultades en la toma de decisiones en el área. Por ello surge como **problema de la investigación**: ¿Cómo contribuir a la gestión documental en el Órgano de Cuadro y Personal de la Dirección de Tecnologías y Sistemas del Ministerio del Interior?

Se determina como **objeto de estudio**: la gestión documental; enmarcado en el **campo de acción**: la gestión documental en el Órgano de Cuadro y Personal de la Dirección de Tecnologías y Sistemas del Ministerio del Interior.



Como **hipótesis** se plantea que: el desarrollo de un módulo de gestión documental contribuirá a la gestión de documentos y expedientes en el Órgano de Cuadro y Personal de la Dirección de Tecnologías y Sistemas del Ministerio del Interior. Siendo la **variable independiente** el módulo de gestión documental y la **variable dependiente** gestión de documentos y expedientes en el Órgano de Cuadro y Personal de la Dirección de Tecnologías y Sistemas del Ministerio del Interior. En la siguiente tabla se muestra la operacionalización de la variable dependiente.

Tabla 1: Operacionalización de la variable dependiente.

Conceptualización	Dimensiones	Indicadores
Son los aspectos en que el sistema propuesto enriquece la gestión de documentos y expedientes en un área empresarial determinada con la finalidad de que dichos procesos se desarrollen de una mejor manera.	Informatización del proceso.	<ul style="list-style-type: none">• Métodos utilizados en la recopilación de datos.• Vías empleadas para guardar la información.• Recursos humanos requeridos.• Recursos materiales disponibles.
	Satisfacción del usuario.	<ul style="list-style-type: none">• Rendimiento del sistema.• Calidad del trabajo.• Integridad de la información.• Confiabilidad de la información.• Disponibilidad de la información.

Para dar solución al problema planteado se define como **objetivo general**: Implementar un módulo de gestión documental para el Órgano de Cuadro y Personal de la Dirección de Tecnologías y Sistemas del Ministerio del Interior.

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

1. Diagnosticar el estado actual y marco teórico referencial relacionado con la gestión de documentos y expedientes.
2. Desarrollar un módulo para la gestión de documentos y expedientes que se manejan en el Órgano de Cuadro y Personal de la Dirección de Tecnologías y Sistemas.



3. Validar el módulo de gestión documental para el Sistema de Gestión de la Información y Control Estadístico de la DTS.

Desde el punto de vista investigativo se utilizaron los siguientes **métodos científicos**:

Métodos Teóricos:

- **Analítico-sintético:** se utilizó para el análisis y la extracción de los elementos más significativos de la bibliografía consultada referente al objeto de estudio; ayudando a establecer una unión entre los conceptos, características y relaciones esenciales entre la gestión documental y la gestión de expedientes; elementos que ayudaron a sustentar la necesidad de la investigación, así como estrategias en su implementación.
- **Histórico-lógico:** se utilizó para realizar un estudio de los sistemas de gestión documental empleados a nivel nacional e internacional en la actualidad y en el estudio de las tendencias más recientes en el desarrollo de aplicaciones homólogas con la finalidad de, con base en dichos datos, complementar las características necesarias del módulo de gestión documental a desarrollar.
- **Modelación:** se utilizó para la confección de los diagramas necesarios en el proceso de desarrollo de software, donde serán representadas las características y relaciones fundamentales; siendo estos una representación abstracta del módulo de gestión documental a desarrollar.

Métodos empíricos:

- **Entrevista:** se utilizó en las conversaciones planificadas con el personal del OCP con el objetivo de obtener información sobre la gestión documental en dicha área, además, para definir los indicadores referentes a la investigación que el cliente necesitaba que fueran cumplidos; intercambiándose directamente con el personal involucrado en esta actividad.
- **Observación:** se utilizó durante toda la investigación con la finalidad de estudiar y visualizar, de forma consciente, las posibles formas de contribuir de manera efectiva al proceso de gestión documental en el OCP.
- **Experimentación:** se utilizó para evaluar las diferentes formas de contribuir al proceso de gestión documental en cada posible caso que arrojó la observación científica. También para



evaluar y mejorar el módulo una vez aplicadas las pruebas y detectadas las deficiencias.

El presente documento está compuesto por 3 capítulos, estructurados de la siguiente manera:

Capítulo 1: Fundamentación teórica; se reflejan los elementos teóricos referentes al objeto de estudio de la investigación científica sobre la gestión documental, constituyendo el Marco Teórico de la misma. Abarca, además, los conceptos asociados a la gestión documental, gestión de expedientes, sistemas de gestión y aplicaciones web. Se incluye el análisis de las soluciones homólogas existentes, la metodología de desarrollo de software y las herramientas y tecnologías utilizadas; todos ellos elementos utilizados en la ejecución de la solución al problema de la investigación.

Capítulo 2: Análisis y diseño del módulo de gestión documental para el Sistema de Gestión de la Información y Control Estadístico de la DTS; contendrá la propuesta de solución al problema de la investigación, así como su funcionamiento y se hace un análisis del modelo del negocio correspondiente al sistema de gestión documental en el OCP. Se identifican los requisitos funcionales y no funcionales, se muestran los artefactos de la metodología de desarrollo de software seleccionada.

Capítulo 3: Implementación y pruebas; se abordarán aspectos relacionados con la implementación del módulo en base a la arquitectura de desarrollo de software definida. Se documentan las pruebas de software realizadas al Módulo de Gestión Documental del Sistema de Gestión de la Información y Control Estadístico de la DTS.



Capítulo 1: Fundamentación teórica.

En el presente capítulo se presentan los conceptos fundamentales que constituyen la base teórica del proceso de gestión documental. Se realiza un estudio de los sistemas homólogos tanto a nivel nacional como internacional. Se realiza el análisis de la metodología de desarrollo de software seleccionada, tecnologías y herramientas que guiarán el proceso de desarrollo de software, así como se seleccionan las adecuadas a utilizar durante el desarrollo de la solución al problema de la investigación.

1.1 Conceptos asociados al dominio del problema

Para un mejor entendimiento de lo que representa la gestión documental y sus funcionalidades, se describen a continuación los siguientes conceptos directamente relacionados con el objeto de estudio.

Gestión

La gestión consiste en un conjunto de trámites que se llevan a cabo para resolver un asunto. Dirección, administración de una empresa, negocio, entre otros (2).

Además, la gestión se considera un proceso en el cual pueden ser reconocidos ciertas etapas. La primera de ellas es la planificación, en esta etapa se fijarán los objetivos a corto y largo plazo y el modo en que serán alcanzados. Es a partir de esta organización donde se determinarán el resto de las etapas. Luego puede ser mencionada la organización, en este momento los gestores determinan detalladamente el procedimiento para alcanzar los objetivos formulados anteriormente. Por último, debe ser mencionado el control, en este caso el o los gestores examinan si la planificación es respetada y los objetivos son cumplidos. Para ello deben ser capaces de realizar ciertas correcciones si las normas no son acatadas (3).

Información

Según la Real Academia Española, la información se define como: “Comunicación o adquisición de conocimientos que permiten ampliar o precisar los que se poseen sobre una materia determinada” (4). Es el conocimiento adquirido por causa del procesamiento de datos.

La información constituye un recurso que se puede comparar con un conjunto de bienes que desempeñan un papel creciente e importante en la vida económica, social, cultural y política de toda sociedad. Existen escenarios complejos en los que la información se utiliza como un recurso económico y se aprecia un mayor



uso de ella por el público. Su contribución se enmarca en la toma de decisiones; así como en su papel de mejorar la gestión del conocimiento (5).

Documento

Tanto la normativa internacional como la legislación cubana recogen diversas referencias y definiciones sobre “Documento” en sus distintas formas, entre las que figura la definición de Documento y Documento Electrónico.

Son documentos los escritos, impresos, planos, dibujos, cuadros, mensajes de datos, fotografías, cintas cinematográficas, discos, grabaciones magnetofónicas, videograbaciones, radiografías, talones, contraseñas, cupones, etiquetas, sellos y, en general, todo objeto que tenga carácter representativo o decorativo, y las inscripciones en lápidas, monumentos, edificios o similares (5).

Son documentos electrónicos aquellos que han sido originados a partir de la información generada, enviada, recibida, almacenada y comunicada por medios electrónicos, ópticos o similares (5).

Expediente

Al igual que sucede con los documentos, existen diversas referencias y definiciones sobre “Expediente” en sus distintas formas, entre las que figuran las definiciones de Expediente y Expediente Electrónico.

Según la Real Academia Española, un expediente se define como: “Documentación, historial o conjunto de datos que permiten juzgar el comportamiento de un funcionario, empleado o estudiante acorde a un asunto o negocio” (6).

Constituye un expediente electrónico el “Conjunto de documentos y actas electrónicas producidos y recibidos durante el desarrollo de un mismo trámite o procedimiento, acumulados por cualquier causa legal, interrelacionados y vinculados entre sí, manteniendo la integridad y orden dado durante el desarrollo del asunto que les dio origen y que se conservan electrónicamente durante todo su ciclo de vida, con el fin de garantizar su consulta en el tiempo” (7).

Gestión de información

Comprende las actividades relacionadas con la obtención de la información adecuada, a un precio adecuado, en el tiempo y lugar adecuado, para tomar la decisión adecuada” (7).



La gestión de la información es un proceso que incluye operaciones como extracción, manipulación, tratamiento, depuración, conservación, acceso y/o colaboración de la información adquirida por una organización a través de diferentes fuentes y que gestiona el acceso y los derechos de los usuarios sobre la misma". La gestión de la información se establece como una disciplina transversal que aparece entrelazada en todas las diferentes capas de una organización, en todos los conceptos de administración como recursos humanos, marketing, finanzas, estrategia, operaciones, entre otros (7).

Los sistemas de gestión de información permiten la comprensión de cada organización desde un enfoque analítico, evaluador y creativo que permite develar las oportunidades que merezcan ser explotadas y contrarrestar las amenazas, además de establecer los factores que resulten críticos y buscar nuevas oportunidades (8).

Gestión de documentos

La gestión documental o gestión de documentos, constituye un conjunto de principios, métodos y procedimientos tendientes a la planificación, manejo y organización de los documentos generados y recibidos por las organizaciones, desde su origen hasta su destino final, con el objetivo de facilitar su utilización y conservación (9).

Como se manifiesta en la ISO 15489-1: 2006, la gestión de documentos ofrece amplios beneficios, que regula las prácticas efectuadas tanto por los responsables de su gestión como por cualquier otra persona que cree o use documentos en el ejercicio de sus actividades (10).

Gestión de expedientes

La gestión de expedientes es la gestión sencilla de toda la documentación que se genera en torno a la relación que se establece entre empleado y empresa una vez que el candidato entra a formar parte de la organización; pudiéndose traducir en la gestión de los documentos del empleado.

Conocido como "el arte de la eliminación de documentos", la administración de expedientes es la práctica de mantener expedientes (físicos o electrónicos) desde el momento de su creación hasta su eliminación final. Comprende la clasificación de toda la documentación del empleado conforme a múltiples criterios (por órgano emisor, empresa, estado, fecha, origen, área de organización, etc.), interrelacionar la documentación y asignar permisos de acceso por persona y por grupo, siendo posible además hacer agrupaciones de los documentos conforme a las necesidades de cada organización (10).



Sistemas de gestión

Un sistema de gestión es un conjunto de etapas unidas en un proceso continuo, que permite trabajar ordenadamente una idea hasta lograr mejoras y su continuidad. Se establecen cuatro etapas en este proceso, que hacen de este sistema un proceso circular, pues en la medida que el ciclo se repita recurrente y recursivamente, se logrará en cada ciclo obtener una mejora (11).

Las cuatro etapas de un sistema de gestión son las siguientes:

Etapas de Ideación: el objetivo de esta etapa es trabajar en la idea que guiará los primeros pasos del proceso de creación que se logra con el sistema de gestión propuesto. Existen varias metodologías para lograr refinar la idea. Sin embargo, se recomienda una práctica muy conocida como la tormenta de ideas o *brainstorming*.

Etapas de Planeación: se definen las estrategias que se utilizarán, la estructura organizacional, el personal que se necesita, la tecnología a utilizar, los recursos necesarios y la clase de controles que se aplican en todo el proceso. El proceso de planificación depende de las características particulares de cada organización, sin embargo, dentro de cualquier proceso formal de planificación, existen tres perspectivas básicas comunes: la estrategia corporativa, de negocios y funcional. A nivel de negocios y funcional, los resultados de esta etapa se enmarcan en propuestas de programas estratégicos de acción y programación de presupuestos. Estas propuestas son finalmente evaluadas y consolidadas a nivel corporativo (12).

Etapas de Implementación: en su significado más general, se entiende por gestión, la acción y efecto de administrar. Pero en un contexto empresarial, esto se refiere a la dirección que toman las decisiones y las acciones para alcanzar los objetivos trazados. Es importante destacar que las decisiones y acciones que se toman para llevar adelante un propósito, se sustentan en los mecanismos o instrumentos administrativos (estrategias, tácticas, procedimientos, presupuestos, entre otros.)

Etapas de Control: función administrativa esencialmente reguladora, que permite verificar si el elemento seleccionado (actividad, proceso, unidad, sistema, etc.), está cumpliendo sus objetivos o alcanzando los resultados que se esperan (12). En un sistema de gestión esta etapa ayuda a lograr los objetivos de una organización mediante una serie de estrategias que incluyen la optimización de procesos, el enfoque centrado en la gestión de la información, entre otros (12).



Las mejores empresas funcionan como unidades completas con una visión compartida. Ello engloba la información compartida, evaluaciones comparativas, trabajo en equipo y un funcionamiento acorde con los más rigurosos principios de calidad y del medioambiente (12).

La implementación de un sistema de gestión tiene las siguientes ventajas:

- Control efectivo de las actividades de la organización.
- Integración de nuevas tecnologías y herramientas.
- Disponibilidad de mayor y mejor información para los usuarios en tiempo real.
- Elimina la barrera de la distancia trabajando con un mismo sistema en puntos distantes.
- Disminuye errores, tiempo y recursos (12).

Entre las desventajas que presentan estos sistemas se encuentran:

- El tiempo que pueda tomar su implementación.
- La resistencia al cambio de los usuarios.
- Los problemas técnicos como fallas de hardware y/o software.
- La inadecuada implementación de las funcionalidades para el apoyo de actividades de la organización (12).

Sistema de Gestión documental

“DMS” son las siglas de Document Management System. Traducido al castellano significa Sistema de Gestión Documental y en ocasiones se emplea también el acrónimo SGD (11).

Un software de gestión documental es una solución digital dedicada al almacenamiento, seguimiento y archivado de documentos.

Se tiende a confundir el DMS con otros dos términos: Sistemas de Gestión de Contenidos (CMS) y Gestión de Contenidos Empresariales (ECM). Mientras que un sistema de gestión de documental se centra en todo el ciclo de vida de un documento, desde el escaneado hasta el almacenamiento, pasando por el uso compartido y la publicación (una especie de archivador digital), un sistema de gestión de contenidos



empresarial o un sistema de gestión de contenidos gestiona todo tipo de contenidos como vídeos, imágenes, medios con tecnología avanzada, correo electrónico, páginas web, etc. (11).

Aplicación Web

Consiste en un documento electrónico que contiene información, cuyo formato se adapta para estar insertado en la *World Wide Web*, de manera que los usuarios puedan entrar a la misma por medio del uso de un navegador, visualizándola con un dispositivo móvil como un *smartphone* o el monitor de computadora (13). Es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador (14).

Una aplicación web es más que un conjunto de páginas entrelazadas entre sí, supone la existencia de código ejecutable. Las aplicaciones web están formadas por diferentes módulos, independientes entre sí, que se pueden ejecutar en el servidor o en el cliente y pueden estar implementados utilizando lenguajes o tecnologías diferentes. Para organizar la información se suelen utilizar bases de datos (15).

1.2 Estudio de homólogos

La necesidad de hacer más factible la disponibilidad, usabilidad e integridad de la información que se maneja en las diferentes empresas a nivel mundial, han dado origen al desarrollo de diversas soluciones informáticas que brindan la posibilidad de realizar una gran cantidad de procesos y acciones que gestionen el contenido que se maneja y genera en las mismas o fuera de estas. Por lo que se hace necesario hacer un análisis de las soluciones homólogas existentes no solo a nivel nacional, sino también internacional.

1.2.1 Soluciones a nivel internacional

Alfresco

Es un sistema de gestión de contenidos de código fuente libre desarrollado en Java, basado en estándares abiertos y de escala empresarial para sistemas operativos tipo Windows, Unix, Solaris y algunas versiones de Linux. Esta aplicación unifica varios aspectos de la gestión y publicación de contenidos en una única solución informática incorporando gestión documental, gestión de contenido web, gestión de registros, gestión de imágenes, interfaz gráfica basada en navegadores de Internet, integración de escritorio con Microsoft Office y OpenOffice.Org; permitiendo tener organizado por carpetas y categorías miles de documentos (16).

OpenKM



Es una aplicación web de gestión documental que utiliza estándares y tecnologías de código abierto desarrollado en Java, que sirve para gestionar documentos, registros, correos electrónicos, imágenes y ofrecer un entorno colaborativo a escala empresarial. Entre sus características incluye gestión documental, gestión de registros, gestión de contenidos, utilidades de clasificación (palabras claves, categorías, metadatos) así como compatibilidad multiplataforma (17).

OpenProdoc

Es un gestor documental que puede instalarse en múltiples plataformas, desarrollado en Java por Joaquín Hierro, con licencia de código abierto. Incluye almacenamiento y recuperación de documentos, almacenamiento y recuperación de expedientes, gestión de metadatos de documentos y expedientes, seguridad basada en listas de control de accesos, gestión de usuarios, grupos y roles, programación de tareas y eventos, gestión y creación de informes personalizados en diversos formatos de ficheros (HTML, XML, CSV, TXT, RIS) para presentar los resultados o exportar conjuntos de documentos (18).

1.2.2 Soluciones a nivel nacional

eXcriba

El GDA eXcriba o Gestor de Documentos Administrativos eXcriba es un sistema de gestión documental basado en el ECM Alfresco, que lleva su contenido a lo largo de su ciclo de vida, permitiendo la gestión de los múltiples documentos de trabajo, ya sean documentos administrativos o de archivo. Desarrollado por la Universidad de las Ciencias Informáticas (UCI), entre las funcionalidades que brinda el software para la gestión de documentos se pueden enunciar: Automatización de los flujos documentales, gestión de documentos, gestión de carpetas, control de versiones, control de acceso y permisos, y notificaciones (19).

SiGeID (Sistema de Gestión Integral de Documentos de archivo)

Es un sistema de gestión documental integrada basado en tecnologías web desarrollado para gestionar la documentación relativa a las empresas de la construcción del territorio de Camagüey. Desarrollado por la Dra. Irina Campillo Torres, de la Universidad de la Habana, en colaboración con la Dra. María Pinto Molina de la Universidad de Granada; está programado en PHP. Posee soporte para varios dominios de explotación con administración central de sus componentes, un repositorio de documentos clasificados en categorías con soporte para documentos confidenciales y/o públicos, búsqueda de documentos según sus propiedades



dentro del sistema, centralización de los documentos en un solo servidor, procesamiento digital, entre otras funcionalidades (20).

SGEL (Sistema de Gestión de Expedientes Laborales)

Es un sistema de gestión de expedientes desarrollado por la EMPI-FAR para gestionar los expedientes laborales en el Departamento de Recursos Humanos de la UBPI Holguín. Es una aplicación web que utiliza Java, HTML y PHP como lenguaje de programación. Entre sus principales características, además de la gestión integrada de expedientes, destaca la integración de la herramienta iReport como constructor/diseñador de informes visual que permite a los usuarios corregir visualmente informes complejos con cartas, imágenes, sub informes, etc. (21).

Soluciones Informáticas	Gestión de documentos	Gestión de expedientes	Gestión de plantillas y/o informes	Metadatos
Alfresco	✓	x	✓	✓
OpenKM	✓	x	✓	✓
OpenProdoc	✓	✓	x	✓
eXcriba	✓	✓	x	✓
SiGelD	✓	x	x	✓
SEGEL	x	✓	✓	✓

1.2.3 Conclusión del estudio de homólogos

Teniendo en cuenta las características y los diversos procesos que se informatizan en los sistemas nacionales e internacionales estudiados, se puede observar que su principal objetivo está encaminado a la gestión de la documentación. Por lo que no se realizan, de forma general, el resto de los procesos descritos en la situación problemática.

En el análisis de las soluciones informáticas homólogas a nivel nacional e internacional se identificaron las diferentes características de cada uno. A continuación, se muestran los principales procesos que debe realizar la solución integral que se propone y su comparación con los que ejecutan las soluciones estudiadas.

Tabla 2: Análisis de los antecedentes.



Partiendo de este análisis, se concluyó que ninguna de las soluciones estudiadas cuenta con todos los procesos que se desean desarrollar para contribuir a la gestión documental en el OCP. Es necesario destacar que, independientemente de su flexibilidad y utilidades, las mismas han sido creadas para áreas específicas de la gestión empresarial, por lo que solucionan parcialmente las necesidades requeridas por el OCP. Dando como resultado que ninguna de ellas pueda considerarse una solución integral al problema de la investigación y hace necesario implementar una solución informática que se ajuste a las características especificadas por el cliente y a las necesidades empresariales en el OCP.

1.3 Metodología de desarrollo de software

Las metodologías de desarrollo de software surgen ante la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental a la hora de desarrollar un producto de software. Dichas metodologías pretenden guiar a los desarrolladores a crear un nuevo software, pero los requisitos de uno u otro son tan variados y cambiantes que ha dado lugar a que exista una gran variedad de metodologías para la creación del software (22).

Dichas metodologías se encuentran divididas en dos grupos fundamentales: las metodologías robustas: dentro de las que se encuentran: *Rational Unified Process* (RUP, por sus siglas en inglés), *Software Capability Maturity Model* (SW –CMM, por sus siglas en inglés), *Microsoft Solution Framework* (MSF por sus siglas en inglés). Las metodologías ágiles: dentro de las que se encuentran: *Extreme Programing* (XP, por sus siglas en inglés), SCRUM, *Feature Driven Development* (FDD por sus siglas en inglés).

XP (Extreme Programing)

La metodología Extreme Programing (XP) o Programación Extrema es una metodología centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. Además, se basa en la realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y audacia para enfrentar los cambios. Se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico (23).

La metodología se basa en:



- **Pruebas Unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que, adelantándose en algo hacia el futuro, se podrá hacer pruebas de las fallas que pudieran ocurrir. Es como si se adelantara a obtener los posibles errores
- **Re fabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- **Programación en pares:** una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa (23).

El ciclo de vida ideal consta de 6 fases:

- **Exploración:** los clientes plantean a grandes rasgos las Historias de usuario que son de interés para la primera entrega del producto.
- **Planificación de entregas:** se establece la prioridad de cada Historia de usuario y los programadores realizan una estimación del esfuerzo necesario de cada una de ellas.
- **Iteraciones:** incluye varias iteraciones sobre el sistema antes de ser entregado. El plan de entregas está compuesto por iteraciones de no más de tres semanas.
- **Producción:** requiere de pruebas adicionales y del rendimiento antes de que el sistema sea trasladado al entorno del cliente.
- **Mantenimiento:** mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones.
- **Muerte del proyecto:** un proyecto entrará en esta fase cuando el cliente no tenga más Historias de usuario para ser incluidas en el sistema (23).

Por todo lo mencionado anteriormente se decidió utilizar la metodología ágil XP, habiéndose tenido en cuenta su sencillez en cuanto a su aprendizaje, como en su aplicación. Además, el equipo de desarrollo está integrado por una persona y el cliente se encuentra disponible para las consultas requeridas durante todo el proceso de desarrollo.



1.4 Herramientas y tecnologías utilizadas

Uno de los aspectos fundamentales que se tienen en cuenta en el desarrollo de una solución informática, está determinado por la selección adecuada de las tecnologías y herramientas en su implementación. Es por eso que se debe hacer un análisis de las mismas para seleccionar las que faciliten en mayor medida el trabajo a realizar.

1.4.1 Lenguaje Unificado de Modelado (UML)

El lenguaje de modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar un diseño de software orientado a objetos. Algunas organizaciones los usan extensivamente en combinación con una metodología de desarrollo de software para avanzar de una especificación inicial a un plan de implementación y para comunicar dicho plan a todo un equipo de desarrolladores (24).

UML es un lenguaje muy conocido y utilizado en el modelado de sistemas de software, permite visualizar, especificar, construir y documentar el sistema a desarrollar. Ofrece un estándar para detallar un plano del sistema (modelo), el cual incluye características conceptuales como procesos de negocio, funciones del sistema, aspectos concretos como expresiones de lenguaje de programación, esquemas de bases de datos, entre otros (24).

A través de este lenguaje los desarrolladores de fases posteriores tendrán mayor dominio y comprensión sobre qué es lo que se debe implementar, permitiendo tanto al cliente como al equipo de trabajo tener una representación real del alcance que puede llegar a tener el producto.

Se selecciona para el desarrollo de la solución propuesta UML en su versión 2.5.1.

1.4.2 Herramienta CASE (Computer Aided Software Engineering)

Las herramientas CASE brindan soporte para desarrollar y mantener software. Además, se consideran de gran ayuda al desarrollador de software o administrador de proyecto, durante una o más fases del desarrollo de software (25).

Para el desarrollo del presente trabajo se selecciona la herramienta CASE Visual Paradigm en su versión 15.1.

Visual Paradigm



Es una herramienta que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, pruebas y despliegue. La herramienta CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas y proyectos UML (25).

Entre sus principales características están el soporte de UML en su versión 2.5.1, confección de diagramas de procesos del negocio, aplicación de ingeniería inversa, modelo a código - diagrama a código, confección de diagramas de flujo de datos, soporte ORM, generación de base de datos, transformación de diagramas Entidad-Relación en tablas de base de datos y distribución automática de diagramas.

1.4.3 Lenguajes de Programación

Entre los lenguajes de programación existentes para desarrollar aplicaciones orientadas a la web se encuentran dos grupos fundamentales de acuerdo con la arquitectura Cliente/Servidor: programación del lado del cliente y la programación del lado del servidor.

La programación del lado del cliente incluye aquellos lenguajes que son únicamente interpretados por una aplicación cliente como el navegador web, entre estos lenguajes se encuentran HTML, JavaScript y CSS.

Los lenguajes de programación del lado del servidor son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él. Entre ellos se encuentran: PHP, C#, Java, entre otros.

HTML (Hyper Text Markup Language)

Lenguaje de Marcado de Hipertexto (HTML, en inglés *HyperText Markup Language*), es mantenido por la *World Wide Web Consortium (W3C)*, es utilizado para la creación de páginas web estáticas. Esta versión incorpora algunas etiquetas nuevas pensadas para hacer que la estructura de la página sea más lógica y funcional.

Para el desarrollo de la aplicación web se selecciona HTML su versión o estándar 5 (HTML5).

El enfoque general ha cambiado bastante respecto a versiones anteriores de HTML, añadiendo semántica y accesibilidad implícitas, especificando cada detalle y borrando cualquier ambigüedad. También se tiene en cuenta que muchas páginas actuales son dinámicas, pareciéndose más a aplicaciones que a documentos. Algo básico es que *HTML5* está definido en base al *DOM* (la representación interna de una web con la que trabaja un navegador), dejando de lado la representación real, definiendo a la vez un estándar HTML y XHTML (26).



CSS

Las Hojas de Estilos en Cascada (CSS, por sus siglas en inglés) es un lenguaje creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar contenidos y su presentación y es imprescindible para crear páginas web con diseño complejo. Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos *HTML/XHTML* bien formados y con significado completo (también llamados documentos semánticos). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes (27).

Para el desarrollo de la aplicación web se selecciona CSS versión 3 (CSS3).

JavaScript

Es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. A pesar de su nombre, no guarda relación directa con el lenguaje de programación Java. Legalmente, es un lenguaje desarrollado y estandarizado por la empresa Sun Microsystems (28).

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con este lenguaje se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios (28).

Una de sus principales características, además de la anteriormente mencionada, es la posibilidad de incorporar en una página web, efectos como aparición y desaparición de texto, animaciones, acciones que se activan al pulsar botones u otros elementos y ventanas con mensajes de aviso al usuario.

Se selecciona para el desarrollo de la aplicación web JavaScript versión 1.8.5.

C#

Leído en inglés "C Sharp" y en español "C Almohadilla", es un lenguaje de programación multiparadigma diseñado y estandarizado por Microsoft para su plataforma .NET. Constituye un lenguaje fuerte con estructuras de datos eficientes y de alto nivel, lo cual, junto a su dinamicidad y naturaleza interpretada, lo convierten en el lenguaje ideal para el desarrollo rápido de aplicaciones en diversas áreas y sobre la mayoría de las plataformas (29).

Entre sus principales características destacan sencillez, modernidad, orientación a objetos, orientación a componentes, seguridad de tipos y compatibilidad.



Se selecciona para el desarrollo de la aplicación web C# versión 8.0.

1.4.4 Entorno de Desarrollo Integrado (IDE)

El entorno de desarrollo integrado es imprescindible en la producción de un software. Un Entorno de Desarrollo Integrado (IDE, en inglés *Integrated Development Environment*) es un conjunto de herramientas integradas para el desarrollo de software. Las herramientas se ejecutan a través de una interfaz gráfica (GUI) y constan de un compilador, un editor y un depurador, entre otros (30).

Algunos de los IDE más utilizados son: Visual Studio, Netbeans, Eclipse, entre otros.

Visual Studio Enterprise 2019

Es un entorno de desarrollo integrado para múltiples sistemas operativos. Soporta múltiples lenguajes de programación tales como C++, C#, Visual Basic .NET, F#, Java, Python, Ruby, PHP; al igual que entornos de desarrollo web como ASP.NET MVC, Django, etc., a lo cual sumarle las nuevas capacidades online bajo Windows Azure en forma del editor Monaco. Puede crear y ejecutar dos tipos de pruebas: unitarias y por orden (30).

Para el desarrollo de la aplicación web se selecciona Visual Studio Enterprise 2019 versión 16.4.2.

1.4.5 Marco de Trabajo (Framework)

Un marco de trabajo o *framework* es un conjunto de componentes físicos y lógicos estructurados de manera que permiten ser reutilizados en el diseño y desarrollo de nuevos sistemas de información. Permiten la utilización de modelos arquitectónicos como Cuatro Capas y Modelo Vista Controlador (Model View Controller, MVC) (31).

ASP.NET MVC 5

Es un *framework* de aplicaciones web que implementa el patrón modelo-vista-controlador (MVC). Desarrollado por Microsoft, destaca por permitir a los desarrolladores de software construir una aplicación web como una composición de tres funciones: modelo, vista y controlador; lo que ayuda a crear una separación lógica entre la lógica del negocio y la lógica de presentación. Incorpora el concepto de enrutamiento, que permite a las aplicaciones aceptar peticiones URL que no se corresponden con ficheros físicos en el servidor; enrutando las peticiones al controlador y a la vista adecuada en función de la URL (31).



Se selecciona para el desarrollo de la aplicación web ASP.NET MVC 5 versión 5.2.3.

1.4.6 Servidor de aplicaciones web

IIS

Internet Information Services o IIS es un servidor web y un conjunto de servicios para el sistema operativo Microsoft Windows. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS. Se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas, entre ellas ASP y ASP.NET; aunque también pueden incluir los de otros fabricantes, como PHP y Pearl. Presenta compatibilidad con los sistemas operativos Windows Vista RTM, Windows Vista SP1, Windows XP, Windows Server 2008, y además es compatible con las arquitecturas x86 y 64-bit (32).

Se selecciona como servidor de aplicaciones web IIS versión 10.0.

1.4.7 Sistema gestor de base de datos (SGBD)

Un Sistema Gestor de Base de Datos (SGBD) o DBMS (Data Base Management System) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Está compuesto por un lenguaje de definición y manipulación de datos y un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, además de garantizar la seguridad e integridad de los mismos (33).

Microsoft SQL Server

Es un sistema de gestión de base de datos relacional desarrollado por la empresa Microsoft disponible solo para sistemas operativos Windows, GNU/Linux y Docker. Soporta transacciones, procedimientos almacenados e incluye un entorno gráfico de administración que permite el uso de comandos DDL y DML gráficamente. Permite trabajar en modo cliente-servidor; además permite administrar información de otros servidores de datos. Puede ser configurado para utilizar varias instancias en el mismo servidor físico (34).

Se selecciona como SGBD Microsoft SQL Server versión 14.0.

1.5 Conclusiones del capítulo

El análisis de las características y funcionamiento de los sistemas de gestión documental en el ámbito nacional e internacional, permitió comprobar que los sistemas estudiados no poseen las características especificadas por el cliente para contribuir al proceso de gestión de documentos en el OCP, necesitando la



creación del módulo de gestión documental para el sistema existente SGICE; lo que permitirá contribuir de manera efectiva al manejo de plantillas, informes, expedientes y documentos.

El estudio sobre las herramientas, tecnologías y metodologías de software utilizadas en la implementación de sistemas de gestión documental, permitió la elección de las herramientas, tecnologías y metodología de desarrollo de software adecuada para desarrollar el sistema. Se seleccionó como lenguaje de programación C# en su versión 8.0, siendo ASP.NET MVC en su versión 5.2.3 el *framework* de desarrollo y Visual Studio Enterprise 2019 el IDE de desarrollo en su versión 16.4.2. Además, como herramienta de modelado Visual Paradigm en su versión 16.2, la cual usa como el lenguaje de modelado UML en su versión 2.5, como sistema gestor de base de datos Microsoft SQL Server en su versión 14.0 y IIS en su versión 10.0 como servidor web; y como metodología de desarrollo de software XP.



Capítulo 2: Análisis y diseño del módulo de gestión documental para el Sistema de Gestión de la Información y Control Estadístico de la DTS.

En este capítulo se exponen las principales características del módulo de gestión documental para el Sistema de Gestión de la Información y Control Estadístico de la DTS. Se dará una propuesta de solución, así como las funcionalidades que deberá cumplir, con su previa descripción y relación entre los componentes que la integran. Incluye la confección del plan de entrega, de acuerdo a la metodología XP, así como el plan de iteraciones. Se describen los patrones a utilizados en el diseño del sistema y se hace una propuesta de arquitectura base que brindará soporte a la aplicación.

2.1 Reglas del negocio

1. El órgano de Cuadro y Personal posee estructuras no subordinadas - Departamento de Telecomunicaciones (DETEL), Departamento de Informática (DIF), Departamento de Ciberseguridad Tecnológica (DCT), Centro de Supervisión Tecnológica (CST), Unidad de Servicios Técnicos (UST), Órgano de Operaciones (OP), Departamento de Planificación y Finanzas (DEPEF) - que forman parte del proceso de atención al personal de pregrado.
2. Cada estructura posee uno o varios oficiales al frente de los cadetes por universidades – Universidad de la Habana (UH), Universidad de las Ciencias Informáticas (UCI), Centro Universitario José Antonio Echevarría (CUJAE) e Instituto Superior Pedagógico Enrique José Varona (ISPEJV) -.
3. Existe una jerarquía para el proceso de atención al personal de pregrado. La misma está encabezada por el **Primer Oficial de Cuadro y Personal**, el cual responde por el trabajo de **los oficiales al frente de las estructuras**, los cuales tienen bajo su responsabilidad a los **coordinadores generales** de las universidades y estos últimos a los **coordinadores por año docente**, dando lugar a niveles de trabajo.
4. Un responsable de algunas de las instancias jerárquicas puede comportarse simultáneamente como responsable de una instancia inferior a la cual dirija.



5. La información del personal de pregrado sólo puede ser modificada en su totalidad por el Primer Oficial de Cuadro y Personal; siendo una excepción de lo anterior la actualización, por parte de los oficiales al frente de las estructuras, de algunos datos de los cadetes -Plan de Actividades y Evaluaciones docentes-.
6. La modificación y archivado de los expedientes del personal de pregrado sólo puede ser realizada por el Primer Oficial de Cuadro y Personal.
7. El Primer Oficial de Cuadro y Personal es principal encargado de comunicar las actividades ministeriales a los oficiales al frente de las estructuras, así como de solicitar la cantidad de cadetes a asistir.
8. Las evidencias de la realización de las actividades ministeriales sólo pueden ser creadas y entregadas por los oficiales al frente de las estructuras.

2.3 Flujo actual del proceso

Con el objetivo de lograr un mayor conocimiento sobre cómo se desarrollan actualmente los procesos en el negocio, se realizó un estudio que aportó la información necesaria para la modelación del negocio en su estado actual.

En cualquier momento del año, los funcionarios del Órgano de Cuadro y Personal reciben información relativa al personal de pregrado (cadetes propios e insertados) de parte de las estructuras. Dicha información es archivada -por el personal del OCP- en plantillas que constituyen en parte:

1. Informes sobre la situación docente del personal de pregrado (PP), militancia, participación en actividades, evaluaciones, etc.
2. Modelos de base de datos temporal donde se almacena información relevante a ser almacenada en los expedientes físicos del PP.

Generalmente el flujo de la información no se detiene en el archivado en su correspondiente expediente físico. Debiendo el primer oficial del OCP (PO) no sólo controlar que la información sea debidamente archivada en las distintas computadoras en formato .xlsx, sino emitir las plantillas que constituyen informes a otras áreas de la institución, así como tramitar las distintas solicitudes que hagan los cadetes a través del oficial al frente de sus estructuras.



Destaca también, a la hora de consultar o modificar la información de los expedientes físicos -porque las plantillas o informes están sujetas a trámites por otras áreas de la institución- la existencia de una hoja de control en cada uno de los expedientes que consta los datos del personal del OCP que consulta o modifica dichos documentos.

Finalmente, el flujo actual de la información en el proceso de gestión documental dentro del OCP puede resumirse en cuatro indicadores: accesibilidad de la información, corrección de errores, recopilación de información y flujo de control de datos.

Accesibilidad de la información

La información que se necesita para la toma de decisiones muchas veces se encuentra descentralizada, ya sea en varias computadoras o archivos físicos, ubicados en una misma área de la institución (el OCP) o en manos del oficial al frente de la estructura.

El hecho de que la información no se encuentre centralizada, evita que se pueda acceder a ella de manera inmediata desde los puestos laborales del OCP, por lo que se evidencia que limitaciones en el acceso y la disponibilidad de la información, incidiendo esto -notablemente- en demoras al momento de reunir información o emplearla en la toma de decisiones.

Corrección de errores

Los informes generados a partir de las plantillas en el OCP muchas veces contienen errores y se hace complicado corregirlos en el momento en que se detectan, teniéndose que volver a crear la mayoría de las veces. Cuando estos errores no son relevantes dichos documentos se tramitan en esas propias condiciones, lo que trae consigo baja calidad en los mismos.

Recopilación de información

El proceso de recopilación de información para generar las plantillas es bastante engorroso. A pesar de que en el OCP utilizan tecnologías como el correo electrónico, para enviar y recibir información, el proceso sigue presentando dificultades, pues el correo electrónico tiene limitantes en cuanto a volumen de datos se refiere. También por esta vía los datos tienden a llegar incompletos, lo que provoca que no se ajusten a la realidad, pues la persona que los envía puede omitir muchos de ellos que son de carácter obligatorio a la hora de confeccionar dichos documentos.

Flujo de control de datos



El proceso de controlar quién modifica o consulta la información es prácticamente inexistente, existiendo solamente una plantilla anexa a los expedientes físicos donde debe constar si se hizo alguna modificación con el nombre y la firma de la persona; siendo este un método ineficiente que atenta contra la confiabilidad e integridad de la información.

2.4 Propuesta de solución

Para solucionar el problema identificado se propone la construcción de un módulo de gestión documental, a través de una aplicación web, que permita mejorar la gestión de los documentos que son generados y manejados sistemáticamente en el Órgano de Cuadro y Personal de la DTS. Además de la gestión de documentos, se debe brindar una serie de servicios como: gestión de expedientes, enlace a los otros módulos del Sistema de Gestión de la Información y Control Estadístico y descarga de documentos.

La aplicación web contará con varios submódulos a través de los cuales los usuarios podrán acceder, y si su rol lo permite, modificar las siguientes informaciones:

- 1- **Documentos:** se encarga de la gestión de los documentos generados y almacenados en el sistema omitiéndose los expedientes. Se utiliza para la creación de Actas de reuniones, Informes y Plantillas, así como su administración. Permitirá crear, guardar, modificar o eliminar un documento determinado y listar todos los adicionados anteriormente.
- 2- **Expedientes:** se encarga de la gestión de los expedientes electrónicos que se almacenan en el sistema. Permitirá crear, guardar, modificar o eliminar un expediente determinado y listar todos los adicionados anteriormente.
- 3- **Log:** se encarga del seguimiento de las operaciones efectuadas en el sistema, es decir, los reportes de las consultas, modificaciones y creación de documentos. Permitirá llevar un registro de todas las modificaciones, consultas, eliminación y descargas de documentos -actas, informes, plantillas o expedientes- realizadas en las últimas 24 horas, así como listar todas las llevadas a cabo por espacio de un mes especificándose o no el tipo de documento.
- 4- **Usuarios:** se encarga de la gestión de los usuarios del sistema y los roles que le son asignados. Permitirá definir los permisos de acceso de los usuarios al usar la aplicación.



2.5 Fase de exploración

Esta es la primera fase de la metodología XP, en la cual los clientes plantean a grandes rasgos las funcionalidades que son de interés para la elaboración del producto, transformándose las mismas en historias de usuario. Partiendo de la información obtenida el equipo de desarrollo evaluará de forma general el tiempo de codificación, se familiarizará con las herramientas, tecnologías y prácticas que serán utilizadas en el proyecto, además se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo para ello (35).

La fase de exploración toma de unas semanas a pocos meses, dependiendo de la habilidad que tengas los programadores con las tecnologías seleccionadas. Es válido aclarar que las estimaciones realizadas en esta etapa son primarias, pues estas se basan en datos de alto nivel los cuales pueden variar a medida que se analicen con mayor cuidado y detalle en las siguientes fases.

2.5.1 Historias de Usuario

Es una simple descripción de una funcionalidad del *software* por la que el cliente va a pagar. Los detalles extra son agregados cuando la historia de usuario es implementada, siendo estos procesos simples y sencillos. En cualquier momento estas historias pueden romperse, remplazarse, unirse o dividirse. Entre las principales características que debe tener una HU se encuentran: debe ser entendida por el cliente (representa un concepto y no una especificación detallada), debe devolver algún valor para el cliente, el tamaño debe ser tal que se puedan construir varias de ellas en una iteración (duración aproximada entre 1-3 semanas ideales), deben ser independientes unas de otras y cada historia debe ser probada (36).

Para la información contenida en la HU existen varias plantillas sugeridas, pero no existe un consenso al respecto. En muchos casos sólo se propone utilizar un nombre y una descripción, además de una estimación de esfuerzo en días ideales. Otras de las actividades a tener en cuenta es la granularidad de las historias, lo cual es muy variable en dependencia de la complejidad del sistema, debido a esto, cada característica importante debe establecerse como una HU (36).

Al comienzo de cada iteración estarán registrados los cambios en las HU y es a partir de ello que se planificará la siguiente. Además, pueden ser descompuestas en tareas de programación y asignadas a los programadores para ser implementadas durante una iteración.

En el análisis para el desarrollo del sistema propuesto, fueron obtenidas veintiún (21) funcionalidades. A continuación, se plantean las funcionalidades de la aplicación:



- RF1 Autenticar usuario.
- RF2 Adicionar usuario.
- RF3 Modificar usuario.
- RF4 Mostrar datos de un usuario.
- RF5 Eliminar usuario.
- RF6 Adicionar documento.
- RF7 Modificar documento.
- RF8 Buscar documento.
- RF9 Eliminar documento.
- RF10 Listar documento.
- RF11 Descargar documento.
- RF12 Exportar documento
- RF13 Adicionar expediente
- RF14 Modificar expediente.
- RF15 Buscar expediente.
- RF16 Eliminar expediente.
- RF17 Listar expediente.
- RF18 Generar registro.
- RF19 Buscar registro.
- RF20 Eliminar registro.
- RF21 Listar registro.

2.5.2 Requisitos no funcionales

Al concebir las Historias de Usuario no se detallan ciertos aspectos que se deben precisar en la elaboración de una aplicación. Estos aspectos influyen en gran medida en el funcionamiento del producto final. La



metodología XP orienta que los detalles de implementación de la historia de usuario se tienen en cuenta en el mismo momento de la concepción, por estas razones se describen a continuación de forma general, algunas de las características no funcionales que debe tener el sistema. Los requerimientos no funcionales de un sistema son "propiedades, cualidades o restricciones que el producto debe cumplir, tales como restricciones de tiempo, estándares, de desarrollo, entre otros con el objetivo de lograr un producto confiable, atractivo y seguro" (37).

RNF 1 Usabilidad

El sistema podrá ser usado por personas con conocimientos básicos en el manejo de computadoras. Deberá contar con una interfaz amigable para facilitar la interacción.

RNF 2 Confidencialidad

Para que el usuario pueda realizar funcionalidades avanzadas en el sistema debe estar autenticado y de acuerdo con el rol que le sea asignado tendrá los permisos asociados. Esto garantizará el acceso controlado a la información, donde se mostrarán las interfaces a los usuarios, en dependencia de los niveles de acceso que posea.

RNF 3 Rendimiento

El sistema deberá ser capaz de gestionar toda la información y dar respuesta a las solicitudes en el menor tiempo posible.

RNF 4 Requisitos de software

Para las computadoras clientes:

- Sistema Operativo: Windows.
- Navegador web: Mozilla Firefox 40.0 o mayor, Chrome.
- Conectividad: Tener conexión a la red de la institución.

Para los servidores web y de base de datos:

- Servidor web: aún por determinar.
- Servidor de base de datos: Microsoft SQL Server.

RNF 5 Requisitos de hardware



Para las computadoras clientes:

- Al menos 1 GB de memoria RAM.
- Procesador Pentium 4 o superior.

Para los servidores:

- Se requiere tarjeta de red.
- Al menos 4GB de RAM.
- Al menos 4GB de espacio libre en el disco duro.
- Procesador Pentium 4 o superior.

Una vez identificadas las funcionalidades del sistema se procede a realizar las Historias de Usuario, a continuación, se muestra como quedan conformadas las siguientes HU: Autenticar usuario, Modificar usuario y Listar registro. El resto de las HU se encuentran en los anexos del presente documento (Ver Anexo 1).

Tabla 3: HU Autenticar usuario.

Historia de Usuario		
ID: 1	Nombre: Autenticar usuario	
Descripción: El usuario que va a interactuar con el sistema se debe identificar insertando su usuario y contraseña del dominio.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones:		

Tabla 4: HU Modificar usuario

Historia de Usuario		
ID: 3	Nombre: Modificar usuario	
Descripción: Permite modificar el nombre, contraseña, rol y permisos de un usuario registrado en la base de datos del sistema.		
Estimación aproximada: 15	Prioridad: Alta	H.U original: 9
Observaciones:		
1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP,		



- independientemente del rol Administrador.
2. Se debe estar autenticado en el sistema.
 3. La modificación de todos los campos no es obligatoria.

Tabla 5: HU Listar registro.

Historia de Usuario		
ID: 21	Nombre: Listar registro	
Descripción: Permite listar y acceder a registros de operaciones de usuarios en el sistema atendiendo a criterios de búsqueda.		
Estimación aproximada: 30	Prioridad: Media	H.U original: 22
Observaciones: <ol style="list-style-type: none">1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP, independientemente del rol Administrador.2. Se debe estar autenticado en el sistema.3. Los criterios de búsqueda son "Fecha" y "Hora".		

2.5.2 Personal relacionado con el sistema

Uno de los principales factores que se deben tener en cuenta cuando se comienza el desarrollo de un sistema informático es la delimitación de la audiencia a la cual va dirigido el mismo, lo cual en algunas ocasiones puede referirse a un personal relacionado con el sistema o alguien completamente ajeno a él. Se debe especificar que esta audiencia a su vez puede ser dividida en grupos atendiendo a sus necesidades (38).

Se define como persona relacionada con el sistema, a toda aquella que de una manera u otra interactúe con este, obteniendo un resultado de uno o varios procesos que se ejecutan en el mismo. También son considerados como personas relacionadas con el sistema aquellas que se encuentran involucradas en dichos procesos, participan en ellos, pero no obtienen ningún resultado de valor (38).

La siguiente tabla muestra las personas que interactúan con el sistema y las acciones que los mismos pueden realizar en él.

Tabla 6: Personal relacionado con el sistema.

Personal	Justificación
----------	---------------

Comentado [A1]: referencia?



Usuario sin autenticar	<ul style="list-style-type: none">- Interactúa solamente con los eventos disponibles en el portal principal. El menú para la gestión de documental no está incluido.
Usuario autenticado	<ul style="list-style-type: none">- Consultar documentos.- Consultar expedientes.- Gestionar documentos (según su rol).- Gestionar expedientes (según su rol).
Oficiales al frente de las estructuras	<ul style="list-style-type: none">- Consultar documentos (que pertenezcan a su estructura).- Consultar expedientes (que pertenezcan a su estructura).
Primer Oficial de Cuadro y Personal	<ul style="list-style-type: none">- Gestionar documentos (de todas las estructuras).- Gestionar expedientes (de todas las estructuras).- Gestionar usuarios (de todas las estructuras).- Ver registros.- Gestionar Registros.

2.6 Fase de Planificación

La fase de planificación tiene como propósito general que los clientes y desarrolladores lleguen a un acuerdo sobre cuáles HU deben estar listas para la primera iteración. De forma particular, es en esta fase donde los programadores obtienen la estimación del esfuerzo necesario para la elaboración de las HU. Es necesario aclarar que el tiempo ideal de desarrollo es aquel en el cual se escribe el código sin distracciones, logrando dedicarse a tiempo completo a esta actividad. Dicha estimación incluye todo el esfuerzo asociado a la implementación de la HU, como son: las pruebas unitarias, integración, refactorización del código, la preparación y ejecución de las pruebas de aceptación, entre otras actividades (39).

La fase de planificación es de muy corta duración, pero de relevante importancia para el avance de la solución a desarrollar. Esta se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de HU. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según el alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación (39).

2.6.1 Prioridad de las Historias de Usuario

La asignación de la prioridad a las HU es una actividad llevada a cabo en conjunto con el cliente, en la misma se decide el orden en el cual se implementarán las HU, siempre y cuando estén de acuerdo usuario y desarrollador. Las HU con menor número de prioridad son las más importantes quedando conformadas de la siguiente manera:



Tabla 7: Prioridad de las Historias de Usuario.

Historia de Usuario	Prioridad
Autenticar usuario	1
Adicionar usuario	1
Modificar usuario	1
Adicionar documento	1
Modificar documento	1
Listar documento	1
Adicionar expediente	1
Modificar expediente	1
Listar expediente	1
Generar registro	1
Listar registro	1
Buscar documento	2
Descargar documento	2
Buscar expediente	2
Buscar registro	2
Eliminar usuario	3
Eliminar documento	3
Exportar documento	3
Eliminar expediente	3
Eliminar registro	3
Mostrar datos de un usuario	3

2.6.2 Estimación del esfuerzo de las HU



Una vez asignadas las prioridades de las HU, se prosigue a estimar el esfuerzo necesario para la elaboración de cada una de ellas por parte del desarrollador. Esta estimación se basa principalmente en la velocidad del equipo de desarrollo y en la semejanza con historias de usuario desarrolladas con anterioridad. Las HU de la presente investigación tienen un valor entre 4-20 puntos (días ideales), 1-3 puntos (medida utilizada normalmente, en semanas ideales) (39).

Los puntos estimados son expresados en días ideales, ha de tenerse en cuenta que pocas veces el cronograma se lleva a cabo exactamente como se planifica. El esfuerzo necesario para construir las HU está basado en la técnica de estimación para el desarrollo ágil, quedando conformada de la siguiente manera:

Tabla 8: Estimación del esfuerzo para las HU.

Historia de Usuario	Esfuerzo estimado (Puntos estimados)
Autenticar usuario	4
Adicionar usuario	10
Modificar usuario	10
Adicionar documento	10
Modificar documento	10
Listar documento	4
Adicionar expediente	15
Modificar expediente	20
Listar expediente	4
Generar registro	15
Listar registro	4
Buscar documento	8
Descargar documento	8
Buscar expediente	8
Buscar registro	8
Eliminar usuario	10



Eliminar documento	10
Exportar documento	10
Eliminar expediente	10
Eliminar registro	10
Mostrar datos de un usuario	4
Total:	188 días

2.6.3 Cronograma de liberación

En el momento de planificar la liberación de un producto de software se debe llevar un balance de esta, pues si se realiza muy pronto no se tendrán suficientes funcionalidades que ameriten dicha liberación, por otro lado, esperar mucho tiempo conlleva a que el software desarrollado quede atrás respecto a la competencia (39).

Entre los elementos a tener en cuenta en el cronograma de liberación es fundamental saber que rara vez un plan marcha respecto a lo acordado, por esa razón el plan de liberación está en constante cambio. Un ejemplo de ello se tiene cada vez que el usuario cambia la prioridad de una historia, cuando el desarrollador divide o une HU, cuando aparecen imprevistos en las tecnologías a utilizar, entre otras acciones. Es por esto que los cambios deben ser aceptados (39). El cronograma de liberación de la presente investigación queda conformado de la siguiente manera:

Tabla 9: Cronograma de liberación.

Iteración	Fecha de liberación
Primera iteración	1ra semana octubre 2019
Segunda iteración	4ta semana noviembre 2019
Tercera iteración	1ra semana enero 2'019

2.6.4 Fase de iteración

Una vez identificadas las HU y estimado el esfuerzo necesario para el desarrollo de cada una de ellas, se procede a realizar la planificación de las etapas de implementación del sistema. Con este fin se define un



plan que contiene las iteraciones a realizar y la relación de las HU que serán implementadas y en qué orden para cada iteración, teniendo en cuenta la duración de las mismas. La cantidad de iteraciones necesarias (IN) para realizar todas las HU identificadas, es obtenida sumando todos los puntos de esfuerzo (PE) de las HU y dividiéndolos entre la velocidad de iteración del equipo (VIE) (39).

$$IN = PE / VIE$$

$$PE = 188 \text{ días (26 semanas)}$$

$$IN = 26 / 7.5$$

$$IN = 3.4$$

Para calcular la velocidad de iteración del equipo (VIE) se divide la cantidad de desarrolladores (CD) entre el factor de dedicación (FD) al proyecto (para la presente investigación es de 2 [50%]) y se multiplica por el tiempo de duración máximo de una iteración (DMI) (en el caso de la presente investigación el tiempo es de 15 días máximo).

$$VIE = (CD / FD) * DMI$$

$$CD = 1, FD = 50\% (2), DMI = 15$$

$$VIE = (1 / 2) * 15$$

$$VIE = 7.5$$

Del cálculo anterior se obtiene que el valor de la velocidad de equipo es aproximadamente 8, y las iteraciones necesarias para desarrollar las historias de usuario 3. De manera que el tiempo de desarrollo de las iteraciones esté balanceado se tomaron todas aquellas HU de la más alta prioridad (1, 2, 3, 6, 7, 10, 13, 14, 17, 18 y 21) y se promediaron sus puntos de esfuerzo (9.6), quedando las iteraciones conformadas de la siguiente manera:

Tabla 10: Primera iteración

Iteración		
Número: 1	HU por orden: 1,2, 3, 6,7,10, 13, 14, 17, 18 y 21	Duración total: 50



Descripción:

Se desarrollan las primeras HU por las que poseen una mayor prioridad, conformando así un producto que, aún sin terminar, pueda mostrar las funcionalidades del sistema. Además, dará una primera vista al cliente, quien suministrará su criterio para la incorporación de nuevos elementos y modificación de los existentes. Al finalizar dicho proceso se contará con las funcionalidades descritas en las Historias de Usuario 1, 2, 3, 6, 7, 10, 13, 14, 17, 18 y 21.

Tabla 11: Segunda iteración

Iteración		
Número: 2	HU por orden: 8, 11, 15 y 19	Duración total: 40
Descripción: De continúan desarrollando las historias de usuario de prioridad media que por cuestión de tiempo no pudieron añadirse a la primera iteración. Se llevan a cabo los señalamientos hechos por el cliente sobre la primera iteración.		

Tabla 12: Tercera iteración

Iteración		
Número: 3	HU por orden: 4, 9, 12, 16 y 20	Duración total: 40
Descripción: Se desarrollan las historias de usuario de prioridad más baja. Al finalizar dicho proceso se contará con las funcionalidades descritas en las historias de usuario 4, 9, 12, 16 y 20 obteniéndose el producto final.		

2.7 Diseño

El papel del diseño en el ciclo de vida del software es adquirir conocimiento de su funcionamiento, constituye el punto de partida para las actividades de implementación, dando soporte a los requisitos funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables y sistemas operativos, que debe poseer la aplicación (40).

2.7.1 Descripción de la arquitectura

Un patrón de arquitectura de software describe un problema particular y recurrente del diseño, que surge en un contexto específico, y presenta un esquema genérico y probado de su solución. Son soluciones que están apoyadas en la experiencia de las compañías y en general, de los desarrolladores (40).



Para el desarrollo del módulo de gestión documental del sistema de gestión de la información y control estadístico de la DTS, se utilizará el patrón de arquitectura de software Modelo Vista Controlador (en inglés MVC) usando el *framework* ASP.NET MVC 5. Este patrón separa los datos y la lógica del negocio de una aplicación de la interfaz de usuarios y el módulo encargado de gestionar los eventos y las comunicaciones. Permite además separar cada una de las lógicas del módulo en archivos independientes permitiendo hacerlo flexible y sencillo de mantener. En la siguiente figura se representa el funcionamiento de la arquitectura Modelo Vista Controlador implementada por el *framework* ASP.NET MVC 5.

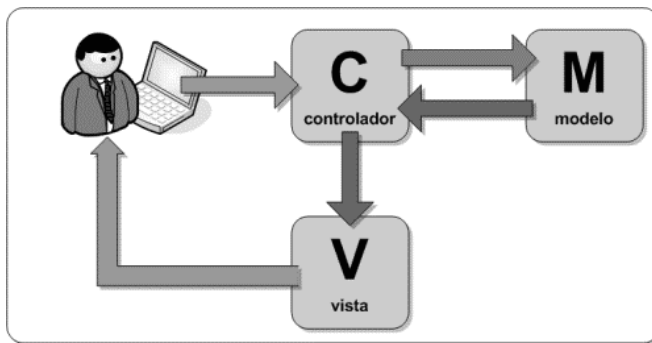


Figura 1: Arquitectura de software MVC implementada por el framework ASP.NET MVC 5.

Controlador: El controlador puede considerarse como un coordinador general del sistema, que regula la navegación y el flujo de información con el usuario, ejerciendo también como intermediario entre la capa de Vista y el Modelo. Entre sus misiones destacan capturar las acciones de este sobre la Vista, interpretarlas y actuar en función de ellas y realizar tareas de transformación de datos para hacer que los componentes de la Vista y el Modelo se entiendan. En el Controlador se encuentran los componentes capaces de procesar las interacciones del usuario, consultar o actualizar el modelo, y seleccionar las Vistas apropiadas en cada momento.

Vista: Los componentes de la Vista son los responsables de generar la interfaz de la aplicación. Contienen los elementos de interacción que permiten al usuario enviar información e invocar acciones en el sistema, como botones, cuadros de edición o cualquier otro tipo de elemento convenientemente adaptado a la tecnología del cliente. En el caso de las aplicaciones para la Web, normalmente en la Vista se encontrarán los componentes capaces de generar el lenguaje de marcado de la página que será enviada al usuario.



Modelo: En la capa Modelo encontraremos siempre una representación de los datos del dominio. Así mismo contendrá la lógica del negocio de la aplicación, por lo que será el responsable de que el sistema se encuentre siempre en un estado consciente e íntegro. También será el encargado de gestionar el almacenamiento y recuperación de datos y entidades del dominio, garantizando los mecanismos de persistencia del sistema, siendo capaz de interactuar con ellos.

En el módulo desarrollado esta arquitectura se evidencia en el siguiente esquema que representa la gestión de plantillas.

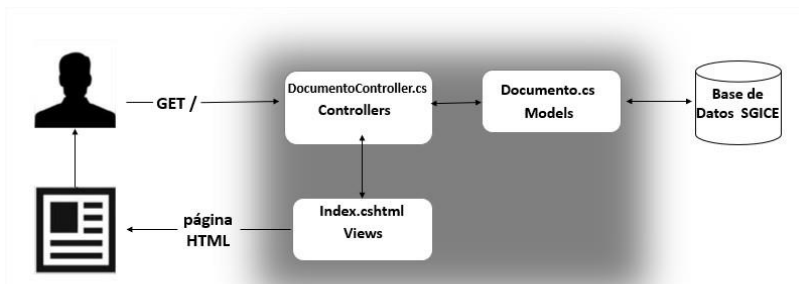


Figura 2: Esquema del funcionamiento de la gestión de documentos.

PlantillaController.cs: representa la información con la que trabaja la aplicación y al mismo tiempo trabaja directamente con la fuente de datos *Plantilla.cs*, encargándose de hacer las peticiones a la base de datos mediante el ORM de *Entity Framework* para enviárselos a *Index.cshtml* dentro de la carpeta “Plantilla” en la carpeta “Views”.

Index.cshtml: maneja la presentación visual de los datos del modelo *Plantilla.cs* generando una representación de los mismos, procesando las interacciones del usuario y comunicándoselas al controlador.

Plantilla.cs: contiene la información básica del módulo de gestión documental referente a las plantillas, sus reglas de validación y su lógica. Contiene los elementos para la representación de los datos y la información de los elementos que conformarán la plantilla.

2.7.2 Patrones de diseño



Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí, adaptada para resolver un problema de diseño general en un contexto particular, y brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares. Los patrones son una pareja problema/solución con un nombre, que codifican buenos principios y sugerencias relacionados frecuentemente con la asignación de responsabilidades. (41)

Entre los patrones que se utilizarán se encuentran los patrones generales de software para la asignación de responsabilidades (GRASP por sus siglas en inglés *General Responsibility Assignment Software Patterns*, Patrones Generales de software para asignación de responsabilidades) y los patrones GoF (por sus siglas en inglés *Gang Of Four*).

Los Patrones GRASP tienen una importante utilidad en el diseño realizado. Describen los principios fundamentales para asignar responsabilidades a los objetos. Los utilizados en el diseño del sistema son los siguientes:

- **Experto:** este patrón se evidencia, por ejemplo, en la clase DocumentoController, la cual controla el dominio de la información referente a la gestión de documentos.



```
11 public class DocumentoController : Controller
12 {
13     TesisContext db;
14
15     [HttpGet]
16     public ActionResult Index()
17     {
18         List<Documento> _archivos = new List<Documento>();
19         using (db = new TesisContext())
20         {
21             // Recuperamos la Lista de los archivos subidos.
22             _archivos = db.Documentos.OrderBy(x => x.Creado).ToList();
23         }
24         // Retornamos la Vista Index, con los archivos subidos.
25         return View(_archivos);
26     }
27
28     [HttpPost]
29     public ActionResult SubirArchivoD(HttpPostedFileBase file)
30     {
31         if (file != null && file.ContentLength > 0)
32         {
33             // Extraemos el contenido en Bytes del archivo subido.
34             var _contenido = new byte[file.ContentLength];
35             file.InputStream.Read(_contenido, 0, file.ContentLength);
36
37             // Separamos el Nombre del archivo de la Extensión.
38             int indiceDelUltimoPunto = file.FileName.LastIndexOf('.');
39             string _nombre = file.FileName.Substring(0, indiceDelUltimoPunto);
40             string _extension = file.FileName.Substring(indiceDelUltimoPunto + 1,
41                 file.FileName.Length - indiceDelUltimoPunto - 1);
42
43             // Instanciamos la clase Archivo y asignamos los valores.
44             Documento _archivo = new Documento()
45         }
```

Figura 3: Código donde se evidencia el patrón Experto.

- **Bajo Acoplamiento:** el bajo acoplamiento aporta la idea de tener una baja relación entre las clases, de forma tal que, en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de las clases, potenciando la reutilización y aumentando la facilidad para encontrar errores. (41)

El bajo acoplamiento se identifica con la especialización de cada submódulo, lo que ha permitido obtener módulos que pueden ser reutilizables para otros sistemas. Un ejemplo lo constituye el hecho de que cada expediente posee una serie de documentos y a su vez en el sistema se manejan documentos que nada tienen que ver con los cadetes, por lo que se implementa un submódulo que se encarga de generar los documentos, así como gestionarlos convirtiéndose dentro de la aplicación



en un punto único de gestión. Como dicho módulo no depende de los expedientes, puede ser llevado a otro sistema con similares características indicando las entidades a las que se les desea aplicar la gestión de documentos determinada.

Este patrón se evidencia también en la clase Registro, la cual tiene una débil dependencia con el resto de las clases y cualquier cambio en ellas no le afectaría.

- **Alta cohesión:** Este patrón plantea que la información que se almacena en una clase debe ser coherente y debe estar (en la medida de lo posible) relacionada con la clase (41).

La alta cohesión se utiliza en la construcción de cada submódulo que se propone para el sistema. Es decir, en dependencia de la arquitectura MVC, si se enfoca cada módulo como una clase en sí, cada módulo entonces representa la información correspondiente con cada funcionalidad presente en el sistema. Específicamente, se lleva a cabo con la utilización de la Cohesión Lógica, donde cada submódulo realiza múltiples tareas relacionadas, pero en tiempo de ejecución sólo una de ellas será llevada a cabo.

Un ejemplo se aprecia en el submódulo Expedientes, el cual permite la gestión de los expedientes de los cadetes y la información que estos poseen en el sistema, o sea, permite crear, editar o eliminar un expediente entre otras tareas. Asimismo, durante la ejecución de la edición o eliminación de un expediente, estas son tareas relacionadas pero que se ejecutan únicamente durante un tiempo de ejecución, en dependencia de la tarea que se esté realizando.

- **Controlador:** este patrón se evidencia en todas las clases tipo “Controller” las cuales poseen responsabilidades específicas a la hora de manejar los mensajes de un evento del sistema a una clase específica.

Patrones de diseño GOF

Los patrones *GoF* describen soluciones simples y elegantes a problemas específicos en el diseño de software orientado a objetos y se agrupan en tres grandes categorías: creacionales, estructurales y de comportamiento. (41)

Los utilizados en el diseño del sistema son los siguientes:

- **Decorador:** ASP.NET MVC presenta el denominado archivo `_layout.cshtml` o también conocido



como plantilla global, en la que convergen todos los elementos comunes a cada una de las páginas del sistema en construcción. Este fichero se complementa con las vistas restantes, decorándolas y obteniéndose la interfaz final que será mostrada al usuario.

```
Layout.cshtml
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="utf-8" />
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>@ViewBag.Title - Gestión Documental</title>
7 @Styles.Render("~/Content/css")
8 @Scripts.Render("~/bundles/modernizr")
9
10 </head>
11 <body>
12 <div class="navbar navbar-inverse navbar-fixed-top">
13 <div class="container">
14 <div class="navbar-header">
15 <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
16 <span class="icon-bar"></span>
17 <span class="icon-bar"></span>
18 <span class="icon-bar"></span>
19 <span class="icon-bar"></span>
20 <span class="icon-bar"></span>
21 </button>
22 @Html.ActionLink("Gestión Documental", "Index", "Home", new { area = "" }, new { @class = "navbar-brand" })
23 </div>
24 <div class="navbar-collapse collapse">
25 <ul class="nav navbar-nav">
26 <li>@Html.ActionLink("Inicio", "Index", "Home")</li>
27 <li>@Html.ActionLink("Documentos", "Index", "Documento")</li>
28 <li>@Html.ActionLink("Expedientes", "Contact", "Home")</li>
29 <li>@Html.ActionLink("Usuarios", "Contact", "Home")</li>
30 <li>@Html.ActionLink("Log.", "Contact", "Home")</li>
31 </ul>
32 @Html.Partial("_LoginPartial")
33 </div>
34 </div>
35 </div>
36 <div class="container body-content">
37 @RenderBody()
```

Figura 4: Código donde se evidencia el patrón Decorador.

Command (Orden): en el *framework* ASP.NET MVC este patrón, el cual está representado implícitamente por los *actions*; cada vez que se ejecuta una acción, encapsula una petición en un objeto. Un ejemplo de su uso en el sistema está dado al realizar la acción de exportar a formato .pdf un documento.

Comentado [w2]: Pon el código donde lo utilizas

```
[HttpGet]
0 references
public ActionResult ExportarArchivo(int id)
{
    Documento _archivo;
    FileContentResult _fileContent;

    using (db = new InterContext())...

    if (_archivo == null)...
    else...
}
```

Figura 5: Código donde se evidencia el patrón Command.

2.7.3 Modelo físico de la base de datos

El diseño de la base de datos representa el tratamiento de la información con carácter persistente dentro del sistema. A continuación, se muestra el modelo físico de la base de datos.

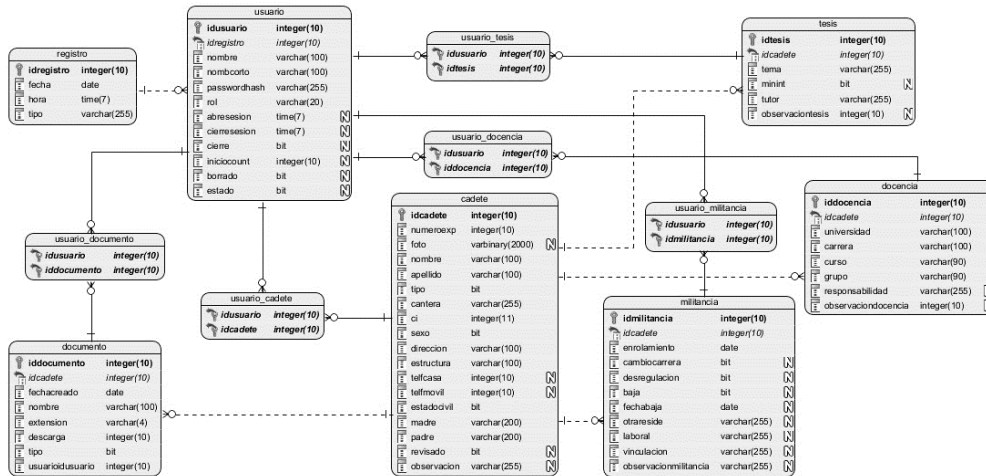


Figura 6: Diagrama del modelo entidad relación de la base de datos.

2.7.4 Tarjetas CRC

Cuando se lleva a cabo el diseño de un sistema XP propone la idea de llevar a cabo las tareas de la manera más simple posible, aplicando prácticas especializadas que la metodología propone. Dichas prácticas



inciden directamente en la realización y elaboración del diseño de un software pues, cuando se tiene un diseño simple se presta mayor atención a codificar, tarea sobre la cual hace más énfasis esta metodología. La metodología XP no recomienda la utilización de diagramas de clases, debido a la gran cantidad de documentación e inconvenientes que ello requiere, aunque tampoco excluye del todo su elaboración, en su lugar orienta la realización de Tarjetas de Contenido, Responsabilidad, Colaboración, (CRC, por sus siglas en inglés). Dichas tarjetas en cada una de las iteraciones van siendo definidas como herramientas de reflexión en el diseño de software orientado a objetos. Estas tarjetas brindan un acercamiento a la identificación de las clases y la información referente a los objetos desarrollados en el proyecto. Cada tarjeta identifica una clase, sus responsabilidades y relaciones con otras clases.

Cada tarjeta CRC contiene el nombre de la clase que representa, además existe una descripción de las responsabilidades (métodos) asociados con la clase, así como una lista de otras clases relacionadas mediante el envío de mensajes. El sistema CRC se usa principalmente como herramienta didáctica y como metodología para estudiar la conducta de los diseñadores orientados a objetos, son también un recordatorio y ayudan a los programadores experimentados y principiantes a comunicarse entre sí acerca de la modelación del entorno con objetos (42).

A continuación, se muestra como quedaron conformadas las tarjetas CRC para las clases que pertenecen al módulo Documentos, el resto de dichas tarjetas se encuentran en los anexos del presente documento (Ver Anexo 3).

Tabla 13: Tarjeta CRC de la clase DocumentoController.

Clase: DocumentoController	
Responsabilidades:	Colaboraciones:
<ol style="list-style-type: none">1. Es la clase base del submódulo.2. Maneja la información de: documento y documento_repository.	<ul style="list-style-type: none">- System (Clase nativa de Visual Studio).- TesisContext

Tabla 14: Tarjeta CRC de la clase Documento

Clase: Documento	
Responsabilidades:	Colaboraciones:
<ol style="list-style-type: none">1. Clase modelo del submódulo.2. Representa los datos del dominio.3. Implementa la lógica de negocio del submódulo.4. Gestionar el almacenamiento y recuperación de documentos.	<ul style="list-style-type: none">- System (clase nativa de Visual Studio).- Documento_repository.

Tabla 15: Tarjeta CRC de la clase Documento_repository.



Clase: documento_repository	
Responsabilidades:	Colaboraciones:
<ol style="list-style-type: none">1. Listar documento.2. Agregar documento.3. Descargar documento.4. Eliminar documento.5. Modificar documento.6. Exportar documento.	<ul style="list-style-type: none">- System (clase nativa de Visual Studio).- TesisContex.- Documento- DocumentoController

2.7.5 Diagrama de despliegue

El modelo de despliegue representa un modelo de objetos que muestra cómo se distribuye físicamente el sistema, teniendo en cuenta la funcionalidad entre cada nodo de cómputo. Este modelo es utilizado como base para la realización de las actividades de diseño e implementación. Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo hardware similar. Los nodos poseen relaciones que representan medios de comunicación entre ellos, tales como Internet, Intranet, bus y similares.

El modelo de despliegue puede describir diferentes configuraciones de red, incluidas las configuraciones para pruebas y para simulación, además de representar una correspondencia entre la arquitectura de software y la arquitectura del sistema (hardware) (43).

La figura muestra el diagrama de despliegue del sistema que se propone.

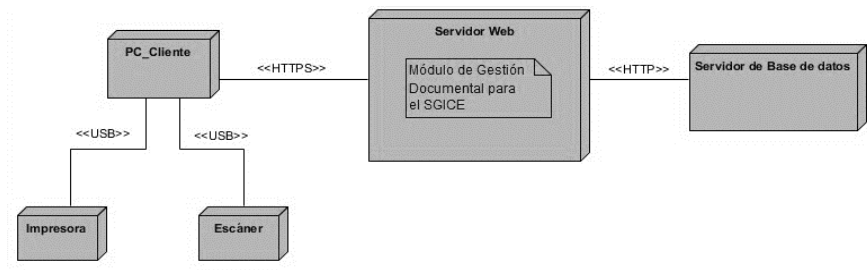


Figura 7: Diagrama de despliegue del sistema.

2.8 Conclusiones del capítulo

La solución propuesta constituye una importante herramienta cuyas funcionalidades van a dar respuesta a una necesidad existente en el Órgano de Cuadro y Personal de la DTS.



Se identificaron los requisitos a implementar en la solución, obteniéndose un total de 21.

Además, se determinaron los patrones de asignación de responsabilidades que le proveen al diseño realizado un bajo acoplamiento, baja complejidad de mantenimiento, baja cantidad de pruebas y una alta reutilización.

A partir de los artefactos obtenidos la solución podrá ser entendida por desarrolladores y clientes con vistas futuras a mejoras y adaptaciones a las condiciones de diversos proyectos.

La planificación del trabajo y el desarrollo por iteraciones aportaron limpieza y orden al proceso de implementación del sistema que continúa en el ciclo de vida, ahorrando recursos y tiempo.



Capítulo 3: Implementación y pruebas

En el capítulo se presentan los resultados de la implementación y validación del módulo para la gestión documental empleando diferentes técnicas definidas por la metodología de desarrollo de software utilizada para medir la calidad de los mismos. Se especifica el estándar de codificación empleado, se crea el diagrama de componentes representando la modelación de los componentes que ayudan a hacer las funcionalidades del sistema, y se presentan las interfaces de pruebas creadas para comprobar el funcionamiento de la solución. A partir del código resultante y su funcionamiento se ejecutan las pruebas unitarias y de aceptación del módulo.

3.1 Estándar de codificación

Los estándares de codificación son reglas de programación enfocadas a su estructura para facilitar la lectura, comprensión y mantenimiento del código (44).

El uso de estos estándares repercute en la calidad del software de manera positiva, facilitando el trabajo del desarrollador y siendo una guía para el futuro mantenimiento de la aplicación.

Estructura

Las propiedades de las clases son declaradas antes que los métodos.

Nomenclatura

Los nombres de las clases y objetos deben comenzar con mayúsculas. Los nombres de los parámetros, variables y atributos usan camellCasing, a excepción de nombres compuestos. Los guiones bajos se utilizan para nombres compuestos de parámetros, variables y atributos, empezando siempre con minúscula. Los nombres de las funciones usan el estándar camellCasing, comenzando siempre por minúscula.

Identación

La indentación se realiza utilizando cuatro espacios, la tabulación no está permitida. Agregar un único espacio después de cada delimitador coma. Agregar un único espacio alrededor de los operadores.

Agregar un único espacio en blanco antes de los paréntesis de apertura, lo mismo sucede para el caso de las instrucciones if, else, while, for. Agregar una línea en blanco antes y después de las declaraciones de



clases y estructuras. Dejar dos espacios por bloques de código. Los paréntesis requieren una línea propia, con excepción de los que están después de las instrucciones if y los ciclos.

Variables y constantes

Los nombres de atributos solo de leerlos dan el conocimiento de a que se refieren. Se utiliza *camelCase* en las declaraciones de variables.

3.2 Implementación

Luego de la fase de diseño y con los resultados obtenidos en la misma da comienzo la etapa de implementación. El modelo de implementación contiene el diagrama de componentes. Este describe la dependencia entre nodos físicos en los que funcionará el sistema, los cuales comienzan a desarrollarse en el flujo de trabajo de diseño y que se perfeccionan en la implementación (43).

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones, siendo los componentes piezas reutilizables de alto nivel a partir de las cuales se pueden construir los sistemas. Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes de software, sean estos componentes de código fuente, binarios o ejecutables. Normalmente contienen componentes, interfaces y relaciones entre ellos, y como todos los diagramas también puede contener paquetes utilizados para agrupar elementos del modelo (45).

La siguiente figura muestra el diagrama de componente del sistema:

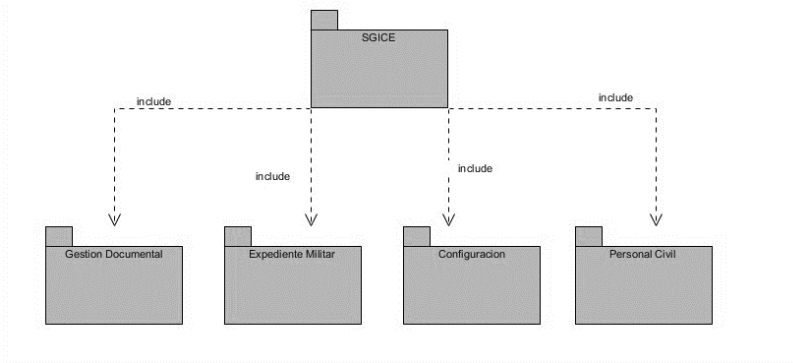


Figura 8: Diagrama de componentes del sistema.

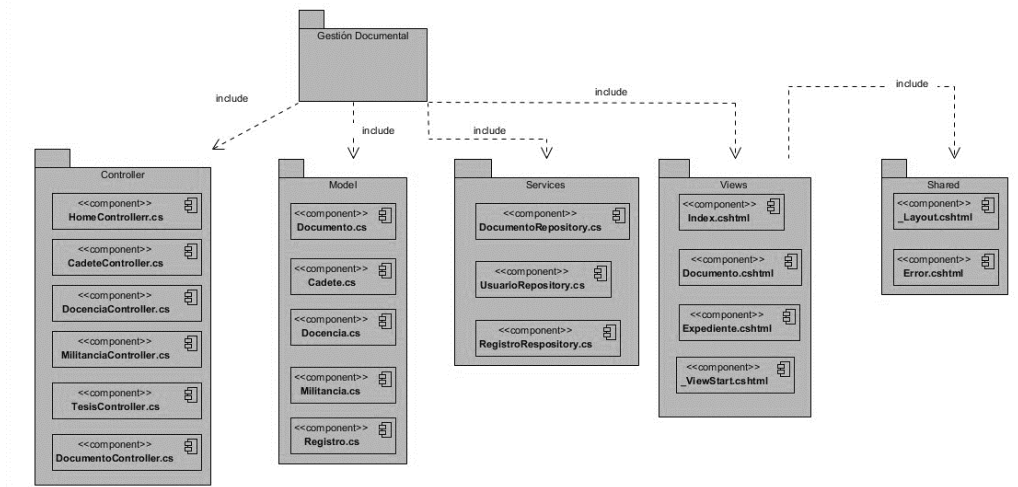


Figura 9: Diagrama de componentes del módulo.

3.3 Pantallas principales de la aplicación

A continuación, se muestra la pantalla de inicio de sesión de la aplicación donde se realiza la autenticación de usuario.

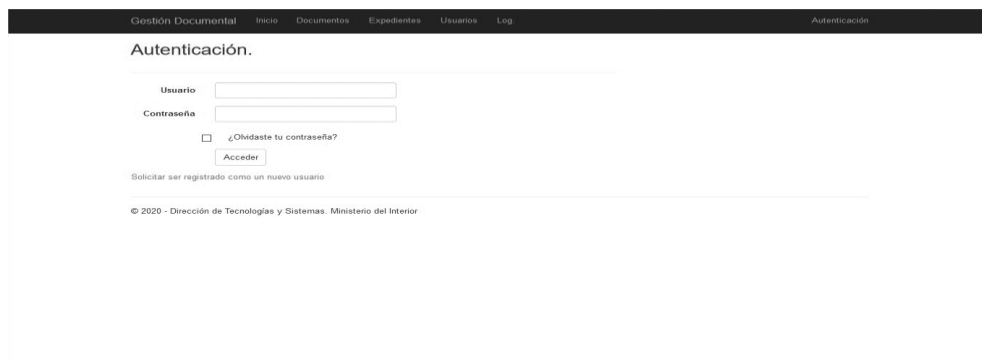


Figura 10: Pantalla de la vista autenticar usuario.



A continuación, se muestra la interfaz principal de la aplicación correspondiente al rol de administrador.

Gestión Documental Inicio Documentos Expedientes Usuarios Log Administrador

Gestión Documental

En este módulo se gestionan los documentos del Órgano de Cuadro y Personal. Incluye la gestión de expedientes del personal de pregrado.

[Leer más >](#)

Gestión de Documentos

La gestión de actas, informes, documentación del personal es ahora un proceso informatizado a disposición del personal de la Dirección de Tecnologías y Sistemas. Considere consultar la documentación relativa a su funcionamiento si va a convertirse en usuario activo del Sistema.

[Documentación >](#)

Gestión de Expedientes

Acceda al tutorial de gestión de expedientes ante cualquier duda o inconveniente con las funcionalidades de esta parte del sistema.

[Tutorial >](#)

Log de Registros del Sistema

Las consultas y modificaciones de los documentos y expedientes del sistema son recogidas en el Log ¡Atención! Acceso sólo con privilegios administrativos.

[Leer más >](#)

© 2020 - Dirección de Tecnologías y Sistemas. Ministerio del Interior

Figura 11: Interfaz principal de la aplicación.

A continuación, se muestra la interfaz correspondiente al almacenamiento y descarga de documentos del submódulo del mismo nombre.

Gestión Documental Inicio Documentos Expedientes Usuarios Log Administrador

Documentos

Examinar... No se ha seleccionado ningún archivo. [Subir archivo](#)

06/09/2020	Horario.xlsx - Descargas: 0 - Eliminar archivo
06/09/2020	Subir archivos al servidor de base de datos.docx - Descargas: 1 - Eliminar archivo
06/09/2020	nuevas aplicaciones web.bit - Descargas: 1 - Eliminar archivo

© 2020 - Dirección de Tecnologías y Sistemas. Ministerio del Interior

Figura 12: Interfaz de la vista gestionar documento.



3.4 Pruebas

Las pruebas de software son el proceso de ejercitar el software con la intención de encontrar y corregir errores. Estas verifican que el software implemente una función específica de forma correcta y validan que las acciones que realiza respondan a los requisitos del cliente. (42)

La programación extrema define dos tipos de pruebas al sistema: pruebas unitarias y pruebas de aceptación. Las pruebas unitarias son desarrolladas por los programadores, dichas pruebas son las encargadas de verificar el código de forma automática mientras que las de aceptación son destinadas a evaluar si al final de la iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente.

3.4.1 Estrategia de prueba

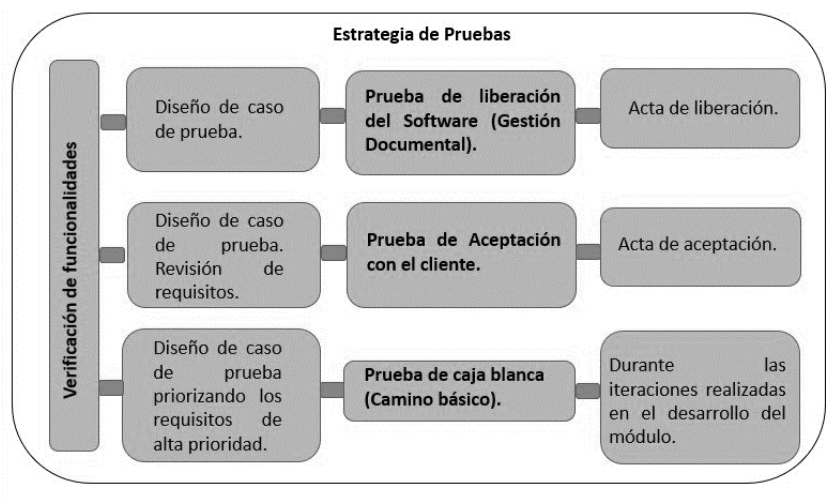


Figura 13: Estrategia de pruebas.

Primeramente, se verificará el cumplimiento de las funcionalidades del sistema a través de las pruebas de liberación -incluyendo pruebas de integración- que serán realizadas por el grupo de calidad del software del Departamento de Informatización de la DTS y la prueba de aceptación del cliente. Además, se realiza también prueba de caja blanca, usando la técnica del camino básico.



3.4.2 Prueba de aceptación

El objetivo de estas pruebas es verificar los requisitos, por lo que los propios requisitos del sistema son la principal fuente de información a la hora de construir las pruebas de aceptación. Las pruebas de aceptación son creadas a partir de las historias de usuario. Durante una iteración la historia de usuario seleccionada en la planificación de iteraciones se convertirá en una prueba de aceptación. El cliente o usuario especifica los aspectos a testear cuando una historia de usuario ha sido correctamente implementada. Una historia de usuario puede tener más de una prueba de aceptación, tantas como sean necesarias para garantizar su correcto funcionamiento y no se considera completa hasta que no supera sus pruebas de aceptación.

Para la realización de las pruebas de aceptación fueron confeccionados casos de prueba, estos no son más que un conjunto de entradas, ejecuciones y resultados esperados. A continuación, se muestran algunos casos de prueba realizados al módulo desarrollado.

Tabla 16: Descripción del caso de prueba número 01 a la HU_6.

Caso de prueba de aceptación	
Código: 06-01	Historia de usuario: Adicionar documento.
Responsable de la Prueba: Eduardo José Figueredo Silva.	
Descripción: Se realiza la prueba a la funcionalidad Adicionar documento.	
Condiciones de ejecución: El usuario debe estar autenticado y tener nivel de acceso.	
Entrada/Pasos de ejecución: <ol style="list-style-type: none">1. Ir al menú Documentos dentro del módulo Gestión Documental.2. Seleccionar la opción Examinar para buscar el documento en la computadora.3. Seleccionar la opción Subir.	
Resultado esperado: Inclusión del nuevo documento en el listado que se muestra en la interfaz.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 17: Descripción del caso de prueba número 02 a la HU_7.

Caso de prueba de aceptación	
Código: 07-02	Historia de usuario: Modificar documento
Responsable de la Prueba: Eduardo José Figueredo Silva.	
Descripción: Se realiza la prueba a la funcionalidad Modificar documento.	



Condiciones de ejecución: El usuario debe estar autenticado y tener nivel de acceso.
Entrada/Pasos de ejecución: <ol style="list-style-type: none">1. Ir al menú Documentos dentro del módulo Gestión Documental.2. Seleccionar un documento e ir a la opción Abrir.3. Modificar el contenido del documento.
Resultado esperado: Mensaje que indique que se modificó el documento.
Evaluación de la prueba: Prueba satisfactoria.

A continuación, se muestran los resultados de la prueba de aceptación, donde en una primera iteración se obtuvo un total de 10 no conformidades (NC), 4 críticas y 6 no críticas. En la segunda iteración se obtuvo un total de 4 NC, 1 crítica y 3 no críticas, y para una tercera iteración no se detectó ninguna NC.

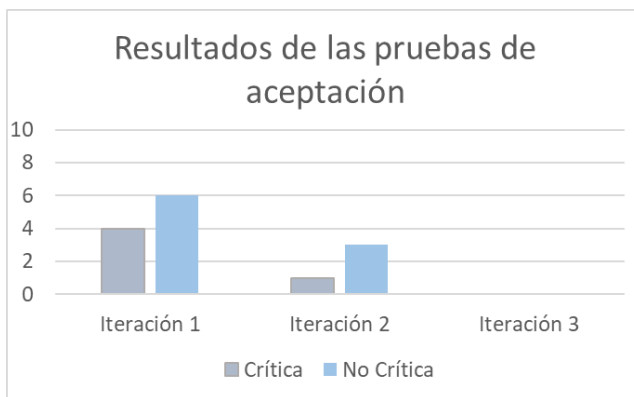


Figura 14: Resultados de las pruebas de aceptación.

3.4.3 Pruebas unitarias

Estas pruebas se realizan para comprobar que el módulo no presente errores en su codificación y que funcione adecuadamente. Para llevarlas a cabo se utilizó el método de caja blanca, utilizando la técnica del camino básico.

Pruebas de caja blanca

Estas pruebas intentan garantizar que se ejecuten al menos una vez todos los caminos independientes que presente el módulo y que todas las estructuras de datos internas sean usadas. Se realizaron pruebas a las



principales funcionalidades del módulo utilizando la técnica del camino básico, para lo cual es necesario conocer el número de caminos independientes de un determinado algoritmo mediante el cálculo de la complejidad ciclomática. El cálculo de la complejidad ciclomática se realiza de tres formas diferentes:

- El número de regiones del grafo de flujo coincide con la complejidad ciclomática.
- La complejidad ciclomática, $V(G)$, de un grafo de flujo G se define como: $V(G) = A - N + 2$, donde A es el número de aristas del grafo de flujo y N es el número de nodos del mismo.
- La complejidad ciclomática, $V(G)$, de un grafo de flujo G también se define como: $V(G) = P + 1$, donde P es el número de nodos predicado (son los nodos de los cuales parten dos o más aristas) que tiene contenido el grafo de flujo G .

Ejemplo de código de la funcionalidad EliminarArchivoD:

```
109
110 [HttpGet]
    0 references
    public ActionResult EliminarArchivoD(int id)
111 {
112     Documento _archivo;
113
114     using (db = new InterContext())
115     {
116         _archivo = db.Documentos.FirstOrDefault(x => x.Id == id);
117     }
118
119     if (_archivo != null && db.SaveChanges()>0)
120     {
121         using (db = new InterContext())
122         {
123             _archivo = db.Documentos.FirstOrDefault(x => x.Id == id);
124             db.Documentos.Remove(_archivo);
125             // Eliminamos el archivo del Servidor.
126             _archivo.EliminarArchivoD();
127         }
128         // Redirigimos a la Acción 'Index' para mostrar
129         // Los archivos subidos al Servidor.
130         return RedirectToAction("Index");
131     }
132     else
133     {
134         return HttpNotFound();
135     }
136 }
137
138
139
```

Figura 15: Código de la funcionalidad EliminarArchivoD.

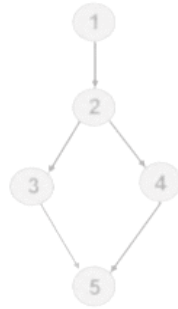


Figura 16: Grafo del flujo asociado a la funcionalidad EliminarArchivoD.

Resultado del cálculo de la complejidad ciclomática:

- El grafo de flujo tiene dos regiones.
- $V(G) = A - N + 2 = 5 - 5 + 2 = 2$
- $V(G) = P + 1 = 1 + 1 = 2$

A partir del resultado obtenido, se determina que la funcionalidad presenta una complejidad ciclomática de 2, lo que deriva que existen a lo sumo 2 caminos lógicos por donde ejecutarse dicha funcionalidad. En la siguiente tabla se presentan los caminos básicos:

Tabla 18: Caminos básicos.

Nro.	Camino básico
1	1, 2, 4, 5
2	1, 2, 3, 5

Se procede a realizar los casos de prueba para la funcionalidad en cuestión. Se debe elaborar al menos un caso de prueba por cada camino básico.

Casos de pruebas

Tabla 19: Caso de prueba del camino básico Nro.1.

Caso de prueba	
Camino	1, 2, 4, 5



Descripción:	Eliminar un documento.
Condiciones de ejecución	Tiene que estar autenticado. Tiene que poseer permisos. El documento no puede estar persistido en la BD.
Entrada	Se da click en la opción eliminar.
Resultado esperado	Se obtiene un mensaje de error.

Tabla 20: Caso de prueba del camino básico Nro.2.

Caso de prueba	
Camino	1, 2, 3, 5
Descripción:	Eliminar un documento.
Condiciones de ejecución	Tiene que estar autenticado. Tiene que poseer permisos. El documento debe estar persistido en la BD
Entrada	Se da click en la opción eliminar.
Resultado esperado	Se elimina el documento.

3.5.4 Integración

Después de validadas las pruebas unitarias se pasa a realizar las pruebas de integración. Aun cuando los módulos de un programa funcionen bien por separados es necesario probarlos conjuntamente ya que pueden traer efectos adversos sobre otros módulos. A partir del esquema del diseño, los módulos probados se vuelven a probar combinados para probar sus interfaces y verificar que todos los componentes funcionan juntos.

Tabla 21: Caso de prueba de Integración.

Caso de Prueba
Componentes que la integran: Gestión Documental, Expediente Militar, Configuración y Estadística.
Descripción: El módulo de Gestión Documental utiliza los nomencladores Tipo de Personal, Dependencia y Estructura del módulo Configuración; la Asignación Mensual del módulo Estadística y Disposición del módulo Expediente Militar.
Condiciones de ejecución: <ol style="list-style-type: none">1. Debe existir un usuario registrado en el sistema.2. Deben estar bien definidos los niveles de acceso.3. Es necesario que todos los campos estén llenos.4. Tienen que estar registrados como mínimo los nomencladores, las estructuras y dependencias, y



existir un registro de disposición para realizar la asignación mensual.
Entrada/Pasos de ejecución: <ol style="list-style-type: none">1. Autenticación.2. Crear estructura.3. Crear dependencia.4. Crear nomencladores.5. Crear registro.6. Realizar las operaciones del módulo documental.7. Realizar la asignación en el módulo estadística.
Resultado esperado: El sistema funciona en su conjunto con todos los módulos integrados.
Evaluación de la prueba: Prueba satisfactoria.

3.5 Resultado de las pruebas

Luego de las pruebas aplicadas al módulo desarrollado se encontraron y corrigieron todos los errores presentados, proporcionando un producto de calidad para el cliente. Se efectuaron 3 iteraciones: en una primera iteración fueron detectadas 6 no conformidades en las pruebas unitarias, 8 no conformidades en las pruebas de integración y 10 no conformidades en las pruebas de aceptación. Se corrigieron la mayoría de errores detectados, los demás pasaron a la próxima iteración y luego de la segunda iteración se encontraron 3 no conformidades en las pruebas unitarias, 5 no conformidades en las pruebas de integración y 4 no conformidades en las pruebas de integración. Al concluir la segunda iteración y ser resueltas todas las no conformidades encontradas en ella, se pasa a la última iteración. Luego de ser resueltas las no conformidades de la segunda iteración y concluida la tercera iteración sin encontrar no conformidades el módulo desarrollado queda listo para ser puesto en uso para la satisfacción de las necesidades del cliente.

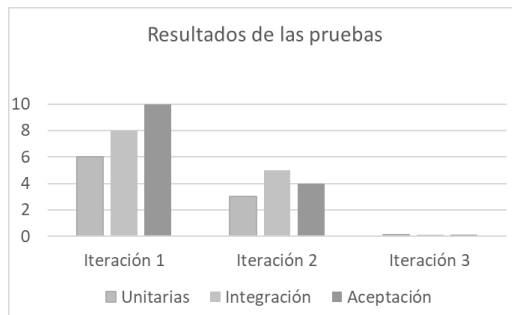


Figura 17: Resultado de las pruebas.



3.6 Conclusiones del capítulo

En el desarrollo de este capítulo se definió el estándar de codificación utilizado en la implementación del módulo permitiendo una uniformidad, claridad y fácil entendimiento del código. El diagrama de componentes presentado permitió una mayor apreciación de la lógica y la organización de los componentes existentes ayudando a su implementación. El desarrollo de las pruebas detectó no conformidades, las cuales fueron corregidas en cada iteración, asegurando una solución acertada en todo momento de la implementación.



Conclusiones generales:

Como resultado del estudio del estado del arte se concluye que la elaboración del marco teórico demuestra la necesidad de desarrollar un módulo de gestión documental para los sistemas informáticos de gestión de la información en ambientes empresariales basado en las necesidades específicas de la gestión empresarial de cada empresa.

Del desarrollo de la solución se concluye lo siguiente:

Sobre la propuesta de solución:

- Se desarrolló un módulo de gestión documental que satisface las necesidades de trabajo en el OCP a partir de los elementos aportados por las reglas del negocio y el estudio del flujo del proceso, así como consideraciones diagnosticadas a través del uso de la entrevista. El mismo puede ser utilizado dentro de grupos empresariales con características similares con el objetivo de lograr una gestión efectiva de documentos y expedientes; incorporando una herramienta informática sencilla para el ágil desempeño de los trabajadores en su vida laboral.

Sobre los resultados obtenidos:

- Se validó la solución informática mediante la aplicación de pruebas unitarias y pruebas de aceptación demostrando que el módulo Gestión Documental cumple con las funcionalidades identificadas.
- Se validó el acoplamiento del módulo Gestión Documental al SGICE mediante la aplicación de pruebas de integración, evidenciándose que el mismo funciona en su conjunto con todos los módulos integrados al sistema.



Recomendaciones:

Una vez concluido el trabajo de diploma y haber dado cumplimiento a los objetivos propuestos para facilitar la gestión documental y mejorar el flujo de trabajo que se realiza en el Órgano de Cuadro y Personal, se propone:

Permitir que la aplicación pueda emitir reportes.



Referencias bibliográficas:

- (1) Cuba. Ministerio de Justicia. *Decreto Ley No. 281 "Sobre la continuidad y el fortalecimiento del Sistema de Dirección y Gestión Empresarial Cubano"* (S.D.G.E.). (2007). La Habana: Ministerio de Justicia.
- (2) WordReference. *Gestión*. [En línea] <http://www.wordreference.com/definicion/gesti%C3%B3n>
- (3) Concepto de. *Concepto de gestión*. [En línea] <http://concepto.de/concepto-de-gestion/>.
- (4) Real Academia. *Real Academia Española*. [En línea] <http://lema.rae.es/drae/?val=informaci%C3%B3n>
- (5) Moreiro, J.A. *Introducción al estudio de la información y la documentación*. (1998). Medellín: Editorial Universidad de Antioquia.
- (6) Real Academia. *Real Academia Española*. [En línea] <http://lema.rae.es/drae/?val=expedienteC3%B3n>
- (7) Quiroga, Lourdes Aja. *Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones.*, Ciudad de La Habana: Ciencias Médicas, octubre de 2002, SciELO, Vol. 10.
- (8) Dans, Enrique. *Information Management. Reflexiones sobre las tecnologías de la información*. [En línea] <http://informationmanagement.wordpress.com/category/gestion/gestion-de-la-informacion/>.
- (9) Díaz Pérez, Maidelyn, Contreras, Yimian de Liz y Rivero Amador, Soleidys. *Características de los sistemas de información que permiten la gestión oportuna de la información y el conocimiento institucional*. Ciudad Habana: ACIMED v.20, 2009. SciELO. ISSN 1561-2880.
- (10) VERGARA, Gonzalo. *Mejora tu Gestión*. [En línea] <http://mejoratugestion.com/mejora-tu-gestion/que-es-un-sistema-de-gestion/>
- (11) The British Standards Institution. BSI. [En línea] <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion>
- (12) Integra-Visión. *La importancia de los sistemas de gestión*. [En línea] <http://veloz-informacioncontable.com/blog-3/la-importancia-de-los-sistemas-de-gestion.html>
- (13) MasterMagazine. *Definición de Web*. [En línea] <http://www.mastermagazine.info/termino/7216.php>
- (14) BuenasTareas. *Definición y tipos de aplicaciones web*. [En línea] <http://www.buenastareas.com/ensayos/Definici%C3%B3n-y-Tipos-De-Aplicaciones-Web/317130.html>
- (15) AddinTech. *Soluciones creativas y personalizadas, funcionales y de entretenimiento*. [En línea] <http://www.addintech.com/index.php?producto=99>



- (16) **ECURED** [En línea] <https://www.ecured.cu/Alfresco>
- (17) **Softwarepara.net** [En línea] <https://softwarepara.net/sistema-gestion-documental#2OpenKM>
- (18) **Softwarepara.net** [En línea] <https://softwarepara.net/sistema-gestion-documental/#4%20OpenProdoc>
- (19) **ECURED** [En línea] https://www.ecured.cu/Gestor_de_Documentos_Administrativos_eXcriba
- (20) **Campillo Torres, Irima**. "Sistema de Gestión Integral de Documentos de archivo para empresas de la construcción del territorio de Camagüey".2010
- (21) **Peña Fijoó, Damaris**. "Sistema de Gestión de Expedientes Laborales de la UBPI Holguín".2014
- (22) **Abrahamsson, P., Salo, O., Ronkainen, J., Warsta, J.** "Software development methods. Review and analysis". VTT Publications. 2002.
- (23) **Wake, W.C.** "Extreme Programming Explored". Addison-Wesley. 2002.
- (24) **HOWARD, Y.** *Use Case Modelling Lab using Visual Paradigm*. 2012
- (25) **EUI-FI.** *Introducción a Herramientas CASE y System Architect*. [En línea] http://users.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro_case_SA.pdf
- (26) **Nieto, Tito Fernando Ale.** *Introducción al HTML5 2010*. [En línea] <http://www.monografias.com/trabajos89/introduccion-al-html-5/introduccion-al-html-5.shtml>
- (27) **Pérez, Javier Eguíluz.** LibrosWeb. *Introducción a CSS 2007*. [En línea] http://www.librosweb.es/css/capitulo_1/breve_historia_de_css.html.
- (28) **Pérez, J.E.** *Introducción a JavaScript*. s.l.: s.l.: pág. 5., 2009., 2009
- (29) **González Seco, José Antonio.** *El lenguaje de programación C Sharp*. Universidad de Valencia. Valencia 2004.
- (30) **GUCKENHEIMER, Sam y PEREZ, Juan J.** *Software Engineering with Microsoft Visual Studio (Microsoft. NET Development Series)*. s.l.: s.l.: Addison-Wesley Professional, 2006., 2006
- (31) **Markiewicz, M.E. and C.J. Lucena.** *Object oriented framework development*. (2001.)
- (32) **IIS Architecture** [En línea] [https://docs.microsoft.com/en-us/previous-versions/iis/6.0-sdk/ms525977\(v=vs.90\)](https://docs.microsoft.com/en-us/previous-versions/iis/6.0-sdk/ms525977(v=vs.90))



- (33) **Cavsi.** Computer Audio Video Systems Integrator. [En línea] <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sqbd/>.
- (34) **SQL Server Development History** [En línea] <https://www.sqlservergeeks.com/sql-server>
- (35) **Kent Beck, Martin Fowler.** *Planificando la programación extrema.* Traducido de: Planning Extreme Programming.
- (36) **Penadés, Patricio Letelier y M^a Carmen.** *Métodologías ágiles para el desarrollo de software: Extreme Programming.* Valencia: Universidad Politécnica de Valencia, 2013.
- (37) **Michael Dobson, Joe Luttrell.** *LESSONS FROM HISTORY.* Functional versus NonFunctional Requirements and Testing. [En línea] <http://www.lessons-from-history.com/node/83>.
- (38) **Akus.net.** *Diseño Web.* [En línea] <https://disenowebakus.net/determinar-la-audiencia-destinada-para-el-sitio-web.php>
- (39) **Tripod.** *Fases de la Programación Extrema.* [En línea] <http://programacionextrema.tripod.com/fases.html>
- (40) **Software Web & APPS.** *Patrones de arquitectura y diseño de software.* [En línea] <https://desarrollodepaginaswb.com.mx/patrones-de-arquitectura-de-software/>
- (41) **Erick Gamma, Richard Helm, Ralph Jonhson, Jonh Vlissides.** *Patrones de Diseño: elementos de software orientados a objetos reutilizables.* [trad.] Acebal, C. F. Addison Wesley, 313 p. Traducido de: Design Patterns: Elements of Reusable Object-Oriented Software.
- (42) **Prezi.** Tarjetas CRC. [En línea] <http://prezi.com/9hc9o0eav71b/tarjetas-crc/>.
- (43) **PRESSMAN, Roger. 2005.** *“Ingeniería del Software. Un enfoque práctico”.* 2005.
- (44) **Ohmyroot!** *Estándares de codificación.* [En línea] <https://ohmyroot.com/buenas-practicas-legibilidad-del-codigo/>
- (45) **VALDÉS, J.** 2014. Diagramas de componentes. 2014.



Glosario de términos:

Artefacto: Producto que se genera durante el proceso de desarrollo de software que puede ser un documento, un diagrama, una descripción, un modelo o el propio software.

Autenticación: Proceso que consiste en probar que el usuario es quien dice ser.

HTTP: Hyper Text Transmission Protocol, protocolo para transferir archivos o documentos hipertexto a través de la red.

Metadatos: La asociación de valores a las propiedades de perfil de documentos. Los metadatos pueden estar definidos por el sistema, como el tamaño del archivo o la fecha de modificación, o definidos por el usuario, como el autor o el título.

Protocolo: Conjunto de reglas y de signos que rigen los intercambios de información entre clientes y servidores.

Web: esta palabra es utilizada para denominar uno de los servicios más importantes de la red Internet.

Hardware: conjuntos de componentes que integran la parte física de una computadora.

Software: equipo lógico o soporte lógico de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos que son llamados hardware.



Anexos:

Anexo 1: Historias de Usuario.

Historia de Usuario		
ID: 2	Nombre: Adicionar usuario	
Descripción: Permite adicionar un usuario a la base de datos.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP, independientemente del rol Administrador.2. Se debe estar autenticado en el sistema.3. Todos los campos son obligatorios.		

Historia de Usuario		
ID: 4	Nombre: Mostrar datos de un usuario	
Descripción: Muestra los datos de un usuario registrado en la base de datos del sistema.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP, independientemente del rol Administrador.2. Se debe estar autenticado en el sistema.3. Cuando un usuario es registrado en el sistema se le informan el rol y permisos asignados.4. Un usuario que interactuó con el sistema no tendrá acceso a esta funcionalidad, con excepción del Primer Oficial del OCP.		

Historia de Usuario		
ID: 5	Nombre: Eliminar usuario	
Descripción: Permite eliminar un usuario registrado en la base de datos del sistema		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP, independientemente del rol Administrador.2. Se debe estar autenticado en el sistema.		

Historia de Usuario		
ID: 6	Nombre: Adicionar documento	
Descripción: Permite seleccionar y adicionar un documento a la base de datos del sistema		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. Se debe estar autenticado en el sistema.		



2. El usuario autenticado debe poseer los permisos necesarios.
3. Todos los campos son obligatorios.

Historia de Usuario

ID: 7 **Nombre:** Modificar documento

Descripción: Permite abrir y modificar el contenido de un documento registrado en la base de datos del sistema.

Estimación aproximada: 7 **Prioridad:** Alta **H.U original:** 1

Observaciones:

1. Se debe estar autenticado en el sistema.
2. El usuario autenticado debe poseer los permisos necesarios.

Historia de Usuario

ID: 8 **Nombre:** Buscar documento

Descripción: Permite buscar y acceder a un documento registrado en la base de datos del sistema insertando su nombre.

Estimación aproximada: 7 **Prioridad:** Alta **H.U original:** 1

Observaciones:

1. Se debe estar autenticado en el sistema.
2. El usuario autenticado debe poseer los permisos necesarios.

Historia de Usuario

ID: 9 **Nombre:** Eliminar documento

Descripción: El usuario que va a interactuar con el sistema se debe identificar insertando su usuario y contraseña del dominio.

Estimación aproximada: 7 **Prioridad:** Alta **H.U original:** 1

Observaciones:

1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP, independientemente del rol Administrador.
2. Se debe estar autenticado en el sistema.

Historia de Usuario

ID: 10 **Nombre:** Listar documento

Descripción: Permite listar y acceder a documentos registrados en la base de datos del sistema según varios criterios de búsqueda.

Estimación aproximada: 7 **Prioridad:** Alta **H.U original:** 1

Observaciones:

1. Se debe estar autenticado en el sistema.
2. El usuario autenticado debe poseer los permisos necesarios.

Historia de Usuario



ID: 11	Nombre: Descargar documento	
Descripción: Permite descargar un documento registrado en la base de datos del sistema.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. Se debe estar autenticado en el sistema.2. El usuario autenticado debe poseer los permisos necesarios.		

Historia de Usuario		
ID: 12	Nombre: Exportar documento	
Descripción: Permite exportar un documento registrado en el sistema, en formato .pdf, para ser tramitado por otras áreas de la institución.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. Se debe estar autenticado en el sistema.2. El usuario autenticado debe poseer los permisos necesarios.		

Historia de Usuario		
ID: 13	Nombre: Adicionar expediente	
Descripción: Permite crear un expediente en la base de datos del sistema.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. Se debe estar autenticado en el sistema.2. Todos los campos son obligatorios.		

Historia de Usuario		
ID: 14	Nombre: Modificar expediente	
Descripción: Permite modificar los datos de un expediente registrado en la base de datos del sistema.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. Se debe estar autenticado en el sistema.2. El usuario autenticado debe poseer los permisos necesarios.		

Historia de Usuario		
ID: 15	Nombre: Buscar expediente	
Descripción: Permite buscar y acceder a un expediente de acuerdo a criterios de búsqueda.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1



Observaciones:

1. Se debe estar autenticado en el sistema.
2. El usuario autenticado debe poseer los permisos necesarios.
3. Los criterios para la búsqueda son "No. Expediente" y "Nombre".

Historia de Usuario

ID: 16 **Nombre:** Eliminar expediente

Descripción: Permite eliminar un expediente registrado en el sistema.

Estimación aproximada: 7 **Prioridad:** Alta **H.U original:** 1

Observaciones:

1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP, independientemente del rol Administrador.
2. Se debe estar autenticado en el sistema.

Historia de Usuario

ID: 17 **Nombre:** Listar expediente.

Descripción: Permite listar expedientes registrados en la base de datos del sistema de acuerdo a criterios de búsqueda.

Estimación aproximada: 7 **Prioridad:** Alta **H.U original:** 1

Observaciones:

1. Se debe estar autenticado en el sistema.
2. El usuario autenticado debe poseer los permisos necesarios.
3. Los criterios de búsqueda son "Tipo", "Estructura", "Universidad" y "Carrera".

Historia de Usuario

ID: 18 **Nombre:** Generar registro

Descripción: Permite al sistema generar un registro que recoge las operaciones de modificación que realiza un usuario en el sistema, desde que se autentica hasta que se desconecta.

Estimación aproximada: 7 **Prioridad:** Alta **H.U original:** 1

Observaciones:

1. Constituye una operación automatizada del sistema a través de la escucha de eventos.

Historia de Usuario

ID: 19 **Nombre:** Buscar registro

Descripción: Permite buscar y acceder a los registros de operaciones de un usuario en el sistema a través de criterios de búsqueda.

Estimación aproximada: 7 **Prioridad:** Alta **H.U original:** 1

Observaciones:

1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP, independientemente del rol Administrador.
2. Se debe estar autenticado en el sistema.



3. Los criterios de búsqueda son "Nombre" y "Rango de Fecha".

Historia de Usuario		
ID: 20	Nombre: Eliminar registro	
Descripción: Permite eliminar registros de operaciones de un usuario en el sistema transcurrido un tiempo dado.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones: <ol style="list-style-type: none">1. El acceso a dicha funcionalidad sólo la posee el Primer Oficial del OCP, independientemente del rol Administrador.2. Se debe estar autenticado en el sistema.3. La eliminación de registros es una operación automatizada del sistema. El usuario sólo personaliza el rango de tiempo en que debe ser llevada a cabo la eliminación.		

Anexo 2: Entrevista

Entrevista	
Objetivos: <ol style="list-style-type: none">1. Definir la situación la situación problemática de la presente investigación.2. Establecer las reglas del negocio.3. Establecer el flujo actual del proceso del negocio.4. Definir los requisitos funcionales y no funcionales del software a desarrollar.	Dirigida a: Primer Oficial de Cuadro y Personal de la DTS, Mayor Milagro Pérez Naranjo.



Guía de preguntas:

1. ¿Qué elementos caracterizan el proceso de gestión documental en el área?
2. ¿Cómo se lleva a cabo el proceso de recopilación de la información?
3. ¿Qué tipos de documento son archivados en el área?
4. ¿Existen diferencias concretas entre los expedientes de un cadete propio y de uno insertado?
5. ¿Cómo y dónde se guarda la información?
6. ¿Qué mecanismos de control se emplean en el área para garantizar la confiabilidad de la información?
7. ¿Es la documentación generada en el área manejada sólo por personal de la misma?
8. ¿Quiénes, de manera específica, intervienen en el proceso de archivado y consulta de la información?
9. ¿Existen reglas que rijan el modo de trabajo en el área? ¿Cuáles?
10. ¿Qué funcionalidades quiere que sean desarrolladas para mejorar el proceso de gestión documental?