

Universidad Internacional de La Rioja

Escuela Superior de Ingeniería y Tecnología

Grado en Ingeniería Informática

Desarrollo de una aplicación web para la gestión documental en una empresa de Telecomunicaciones

|  |  |
| --- | --- |
| Trabajo fin de estudio presentado por: | Marcos Garcia Fernandez |
| Director/a: | Flor Nancy Díaz Piraquive |
| Fecha: | Julio 2024 |
| Repositorio del código fuente: |  |

Resumen

El presente proyecto tiene como finalidad la realización del análisis, diseño implementación y validación de una aplicación web que facilite las tareas relativas a la gestión documental existentes en una empresa real del sector de las telecomunicaciones.

La aplicación permite la creación por parte del personal de la empresa, de formularios de requerimiento de documentación de distintos ámbitos que ha de ser aportada por usuarios externos a la misma y posteriormente validada por el requirente permitiendo en todo momento conocer a ambos usuarios el estado y validez de la documentación aportada.

Asimismo, permite a la empresa facilitar documentación a personal ajeno a esta mediante un canal de comunicación seguro y controlado.

**Palabras clave:** Gestión documental, validación, JavaScript, REACT, MERN

Abstract

The purpose of this project is to carry out the analysis, design, implementation and validation of a web application that facilitates the tasks related to document management existing in a real company in the telecommunications sector.

The application allows the creation by the company staff of documentation request forms in different areas that must be provided by external users and subsequently validated by the requester, allowing both users to know at all times the status and validity of the documentation provided.

Likewise, it allows the company to provide documentation to personnel outside of it through a secure and controlled communication channel.

**Keywords**: Document management, validation, JavaScript, REACT, MERN

Índice de contenidos

[1. Introducción 10](#_Toc166681411)

[1.1. Motivación 10](#_Toc166681412)

[1.1.1. Casos problemáticos 11](#_Toc166681413)

[1.1.2. Beneficios del desarrollo de una aplicación web 15](#_Toc166681414)

[1.2. Planteamiento del trabajo 15](#_Toc166681415)

[1.3. Estructura del trabajo 17](#_Toc166681416)

[2. Contexto y Estado del Arte 18](#_Toc166681417)

[2.1. Necesidad de la optimización en la información 18](#_Toc166681418)

[2.1.1. Importancia de la telemática 19](#_Toc166681419)

[2.2. La digitalización de la gestión documental 20](#_Toc166681420)

[2.3. Gestión documental 22](#_Toc166681421)

[2.3.1. Sistemas de gestión documental (SDG) 23](#_Toc166681422)

[2.3.2. La estandarización en los sistemas de gestión 25](#_Toc166681423)

[2.3.2.1. Norma ISO 30300:2011 y 30301:2011 26](#_Toc166681424)

[2.3.3. Requisitos de la gestión documental 28](#_Toc166681425)

[2.3.4. Etapas de la gestión documental 28](#_Toc166681426)

[2.4. Software para la Gestión Documental 29](#_Toc166681427)

[2.4.1. Historia del software de gestión documental 29](#_Toc166681428)

[2.4.2. Actualidad de las herramientas de gestión documental 30](#_Toc166681429)

[2.4.2.1. Microsoft SharePoint 34](#_Toc166681430)

[2.4.2.2. Nextcloud 36](#_Toc166681431)

[2.4.2.3. Metacontratas 40](#_Toc166681432)

[2.4.2.4. DocuWare 43](#_Toc166681433)

[2.4.2.5. Comparativa 44](#_Toc166681434)

[2.5. Aplicaciones web 46](#_Toc166681435)

[2.5.1. Etapas del desarrollo web 47](#_Toc166681436)

[2.5.2. Importancia para la gestión documental 50](#_Toc166681437)

[2.5.3. Metodologías en el desarrollo de aplicaciones web 51](#_Toc166681438)

[2.5.4. Gestión documental en las empresas de telecomunicaciones 52](#_Toc166681439)

[3. Objetivos y metodología de trabajo 54](#_Toc166681440)

[3.1. Objetivo general 54](#_Toc166681441)

[3.2. Objetivos específicos 54](#_Toc166681442)

[3.3. Metodología de trabajo 55](#_Toc166681443)

[3.3.1. Modelo en Cascada vs SCRUM 55](#_Toc166681444)

[3.3.2. Elección de la Metodología en Cascada: Justificación y Razones 56](#_Toc166681445)

[3.3.3. Metodología en Cascada 57](#_Toc166681446)

[3.3.3.1. Fases de la metodología en cascada 58](#_Toc166681447)

[4. Diseño y evaluación de la propuesta 63](#_Toc166681448)

[4.1.1. Alcance 63](#_Toc166681449)

[4.1.2. Análisis de requisitos 64](#_Toc166681450)

[4.1.2.1. Definición de actores 64](#_Toc166681451)

[4.1.2.2. Casos de uso 65](#_Toc166681452)

[4.1.2.3. Requisitos funcionales 67](#_Toc166681453)

[4.1.2.4. Requisitos no funcionales 78](#_Toc166681454)

[4.1.3. Diseño 78](#_Toc166681455)

[4.1.3.1. Arquitectura de la aplicación 79](#_Toc166681456)

[4.1.3.2. Tecnologías empleadas: El stack MERN 80](#_Toc166681457)

[4.1.3.3. Capa de presentación 95](#_Toc166681458)

[4.1.3.4. Lógica de negocio 96](#_Toc166681459)

[4.1.3.5. Capa de datos 98](#_Toc166681460)

[4.1.4. Implementación 99](#_Toc166681461)

[4.1.5. Pruebas 99](#_Toc166681462)

[5. Conclusiones y trabajo futuro 99](#_Toc166681463)

[5.1. Conclusiones del trabajo 99](#_Toc166681464)

[5.2. Líneas de trabajo futuro 99](#_Toc166681465)

[Referencias bibliográficas 100](#_Toc166681466)

[Anexo A. Título del anexo. 107](#_Toc166681467)

[Índice de acrónimos 108](#_Toc166681468)

Índice de figuras

[Figura 1:Flujo de gestión documental de subcontratas (Fuente: Elaboración Propia) 11](#_Toc166681469)

[Figura 2: Flujo de documentación de una instalación (Fuente: Elaboración Propia) 12](#_Toc166681470)

[Figura 3: Flujo de envío de documentación a clientes (Fuente: Elaboración Propia) 14](#_Toc166681471)

[Figura 4: Logotipo de Microsoft SharePoint (Fuente: Microsoft) 34](#_Toc166681472)

[Figura 5: Imagen de SharePoint (Fuente: SharePoint) 35](#_Toc166681473)

[Figura 6: Logotipo de Nextcloud (Fuente: Nextcloud) 36](#_Toc166681474)

[Figura 7: Imagen de Nextcloud (Fuente: Elaboración propia) 37](#_Toc166681475)

[Figura 8: Compartición de archivos (Fuente: Elaboración propia) 38](#_Toc166681476)

[Figura 9: Control de versiones (Fuente: Elaboración propia) 38](#_Toc166681477)

[Figura 10: Complementos (Fuente: Elaboración propia) 39](#_Toc166681478)

[Figura 11: Requisitos mínimos (Fuente: Nextcloud) 39](#_Toc166681479)

[Figura 12: Logotipo de Metacontratas (Fuente: Metacontratas) 40](#_Toc166681480)

[Figura 13: Imagen de Metacontratas (Fuente: Metacontratas) 41](#_Toc166681481)

[Figura 14: Configuración de documentos solicitados (Fuente: Metacontratas) 41](#_Toc166681482)

[Figura 15: Listado de accesos a un centro de trabajo (Fuente: Metacontratas) 42](#_Toc166681483)

[Figura 16: Listado de documentación validada (Fuente: Metacontratas) 42](#_Toc166681484)

[Figura 17: Logotipo de DocuWare (Fuente: DocuWare) 43](#_Toc166681485)

[Figura 18: Imagen de DocuWare (Fuente: DocuWare) 43](#_Toc166681486)

[Figura 19: Modelo de desarrollo en cascada (Delgado Olivera & Díaz Alonso, 2021) 58](#_Toc166681487)

[Figura 20: CU1 - Crear solicitud de documentación (fuente: elaboración propia) 68](#_Toc166681488)

[Figura 21: CU2 - Enviar documentación solicitada (fuente: elaboración propia) 70](#_Toc166681489)

[Figura 22: CU3 - Validar documentación recibida (fuente: elaboración propia) 72](#_Toc166681490)

[Figura 23: CU4 - Corregir documento incorrecto (fuente: elaboración propia) 74](#_Toc166681491)

[Figura 24: CU5 - Listar documentación pendiente (fuente: elaboración propia) 76](#_Toc166681492)

[Figura 25: CU6 - Descargar facturas (fuente: elaboración propia) 77](#_Toc166681493)

[Figura 26: Arquitectura del sistema 79](#_Toc166681494)

[Figura 27: Stack MERN (Fuente: https://www.mongodb.com/mern-stack) 91](#_Toc166681495)

[Figura 28: Implementación de la vista "Subida de documentación" 95](#_Toc166681496)

[Figura 29: Diagrama de secuencia para solicitud de documentación 97](#_Toc166681497)

[Figura 30: Diagrama de secuencia para solicitud de documentación 97](#_Toc166681498)

Índice de tablas

[Tabla 1: Comparativa de las soluciones presentadas (Fuente: elaboración propia) 45](#_Toc166681499)

[Tabla 2: Actores del sistema (Fuente: elaboración propia) 65](#_Toc166681500)

[Tabla 3: Casos de uso (Fuente: Elaboración propia) 66](#_Toc166681501)

[Tabla 4: Tabla comparativa de Stacks 94](#_Toc166681502)

[Tabla 5: Diccionario de datos de la BD 98](#_Toc166681503)

# Introducción

El propósito de este trabajo es el de desarrollar una aplicación web para la gestión documental en la Empresa Emtel del Noroeste.

El objetivo general de dicha aplicación es el de dotar a la empresa de un canal ágil para el intercambio documental con sus instaladores, subcontratas y clientes, facilitar la organización de la información recibida asi como su validación por parte de personal de la empresa para asegurar que cumple con los estándares definidos y controlar de manera sencilla la documentación pendiente de ser recibida.

Mediante el desarrollo de dicha herramienta se pretende simplificar las tareas relativas a la gestión documental, tratando de reducir los recursos que la empresa ha de dedicar a las mismas debido a la falta de métodos simples que puedan ser gestionados sin necesidad de contar con personal especifico.

## Motivación

Emtel del Noroeste es una empresa gallega dedicada a la realización de toda clase de instalaciones de telecomunicaciones y la prestación de servicios de internet y telefonía.

Lleva a cabo actuaciones por toda Europa contando para ello con un elevado número de instaladores y subcontratas. La cantidad de trabajos simultáneos que se acometen en distintas ubicaciones asi como el elevado número de empresas con las que ha de colaborar han generado un problema a la hora de recopilar toda la documentación que la empresa precisa, tanto para documentar los trabajos acometidos como para satisfacer las distintas necesidades de su actividad laboral.

Actualmente, se ha elevado de manera considerable la cantidad de documentación necesaria para el correcto funcionamiento de la empresa y ha surgido una problemática debido a esto. La documentación no siempre se recopila en tiempo y forma, se producen extravíos de información y la disparidad de los medios utilizados a falta de procedimientos específicos genera confusión y la consecuente pérdida de tiempo y recursos.

Por tanto, la empresa actualmente ve necesario la implantación de herramientas y métodos que subsanen estas deficiencias e impidan que se agraven y generen consecuencias tales como malestar en los clientes o pérdida de confianza por parte de estos.

Tal y como menciona en su libro Gestión documental en las organizaciones la autora Patricia Russo Gallo “la gestión de la documentación y la información en una organización debe ir orientada a las personas que la utilizarán, sea proveedor, cliente o trabajador de la empresa.” (Gallo, 2011)

Partiendo de esta base, el desarrollo de una aplicación web orientada a los procesos de intercambio de documentación existentes en la empresa dotará a la misma de mayor agilidad en este ámbito tan fundamental hoy en día en las organizaciones como es la gestión de la información al mismo tiempo que libera recursos empleados innecesariamente debido a la mala organización de dichas tareas.

### Casos problemáticos

En el presente apartado expondremos algunos de los casos problemáticos asociados a las tareas de gestión documental existentes actualmente en la organización que motivan el presente trabajo.

**Caso problemático 1: Deficiencias en la gestión documental de subcontratas**

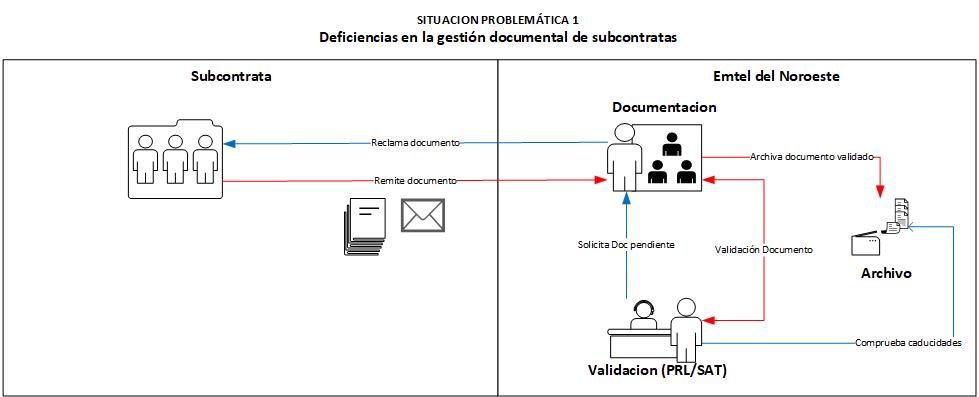
****

Figura 1:Flujo de gestión documental de subcontratas (Fuente: Elaboración Propia)

En la empresa Emtel del Noroeste, la garantía de calidad y cumplimiento de los estándares es un aspecto fundamental de su operación, evidenciado por su certificación ISO 9001. Sin embargo, la gestión documental de las subcontratas, un aspecto crucial para mantener dicha certificación presenta importantes deficiencias que comprometen la eficiencia y el cumplimiento normativo de la organización.

Actualmente, el proceso de recopilación de documentación de las subcontratas se realiza principalmente a través de correos electrónicos, lo que genera una falta de trazabilidad y control sobre la documentación recibida. Esta falta de registro sistematizado dificulta la verificación de si todas las subcontratas han aportado los documentos requeridos en tiempo y forma.

Además, la naturaleza temporal de muchos de estos documentos añade una capa adicional de complejidad al proceso. La responsabilidad de revisar las fechas de vencimiento, solicitar la documentación necesaria, asegurar su recepción y verificar su validez recae en un individuo designado para esta tarea. Sin embargo, este enfoque manual es propenso a errores y omisiones, lo que conlleva una alta probabilidad de que algunas de estas tareas no sean realizadas en el momento indicado.

Como resultado, existe un riesgo significativo de que algunas subcontratas operen con documentación pendiente o caducada, lo que no solo compromete el cumplimiento normativo de la empresa, sino que también podría tener repercusiones legales y financieras adversas.

**Caso Problemático 2: Deficiencias en la Documentación de Instalaciones**

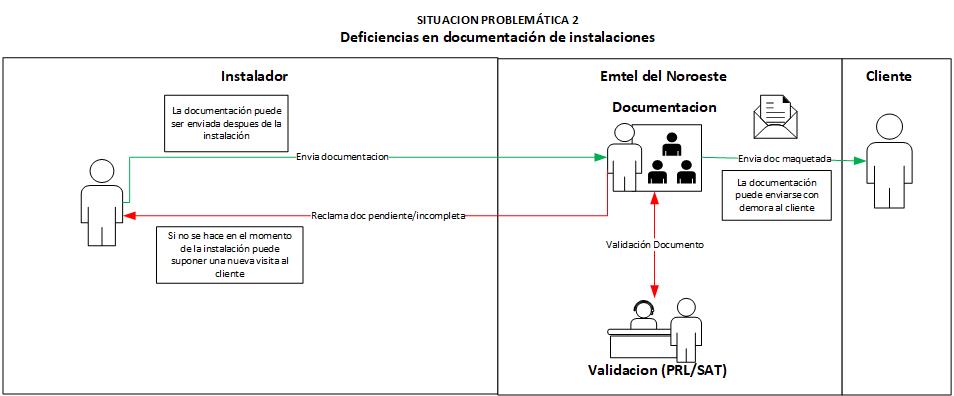
****

Figura 2: Flujo de documentación de una instalación (Fuente: Elaboración Propia)

En Emtel del Noroeste, la documentación de las múltiples instalaciones realizadas representa un componente crucial para garantizar la calidad del servicio y cumplir con los estándares exigidos por la ISO 9001 y las demandas específicas de los clientes. Sin embargo, el proceso actual de documentación presenta una serie de deficiencias que afectan la eficiencia y la calidad del servicio ofrecido.

Los técnicos encargados de realizar las instalaciones capturan fotografías durante el proceso y posteriormente las envían al departamento de documentación a través de diversos medios como correos electrónicos, WhatsApp u otras aplicaciones. Esta falta de un procedimiento estandarizado dificulta la gestión y seguimiento de la documentación, ya que no se establece un flujo claro y eficiente para su revisión y maquetación.

La documentación de las instalaciones puede experimentar demoras significativas debido a la falta de un proceso ágil de revisión y maquetación en el departamento correspondiente. Esto puede resultar en envíos tardíos de la documentación al cliente, lo que afecta negativamente la percepción de la empresa y puede generar insatisfacción por parte del cliente.

La detección de irregularidades en la documentación requiere la realización de nuevas visitas por parte de los técnicos, lo que implica una pérdida de tiempo y recursos. Además, la falta de retroalimentación temprana sobre las deficiencias en la documentación dificulta la corrección oportuna de los errores, prolongando aún más el proceso y aumentando el riesgo de incumplimiento.

La ausencia de un sistema centralizado para el seguimiento de la documentación pendiente dificulta que los técnicos identifiquen los documentos que aún deben aportar o corregir, lo que puede generar confusión y redundancia en el trabajo realizado.

En resumen, las deficiencias en el proceso de documentación de instalaciones en Emtel del Noroeste afectan negativamente la eficiencia operativa, la calidad del servicio y la satisfacción del cliente. Es crucial implementar medidas para estandarizar y agilizar este proceso, garantizando una gestión eficiente de la documentación y un cumplimiento adecuado de los estándares y requisitos del cliente.

**Caso Problemático 3: Desafíos en el Envío de Documentación y Facturas a Clientes**

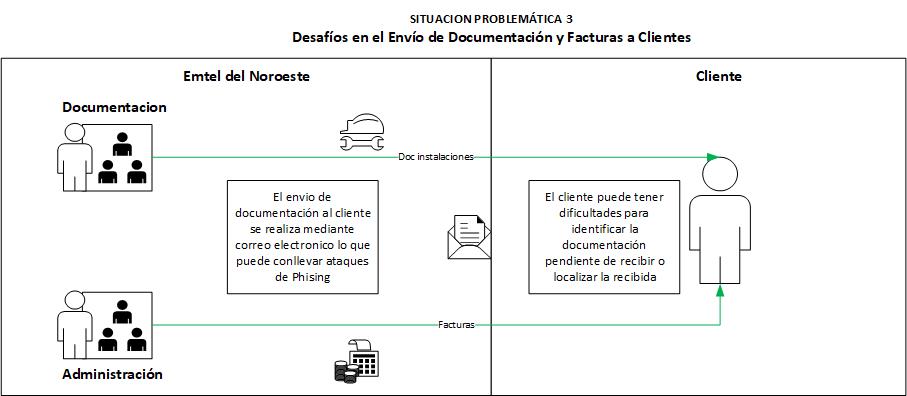
****

Figura 3: Flujo de envío de documentación a clientes (Fuente: Elaboración Propia)

Emtel del Noroeste se enfrenta a una serie de desafíos en el envío de documentación y facturas a sus clientes, lo que afecta la eficiencia operativa y la satisfacción del cliente.

Falta de Seguimiento y Control:

El envío de distintos tipos de documentación a través de correos electrónicos, llevado a cabo por varios miembros de diferentes departamentos, dificulta el seguimiento y control de qué documentación ha sido enviada a un cliente. Esto puede dar lugar a situaciones donde la empresa no pueda acceder a la documentación necesaria, especialmente si el miembro encargado está ausente.

Problemas en la Entrega de Facturas:

El envío de facturas por correo electrónico ha resultado en reclamaciones por parte de clientes que aseguran no haber recibido dichos correos. Estas reclamaciones han provocado retrasos en los pagos y pueden afectar negativamente la relación con los clientes y la liquidez financiera de la empresa.

Riesgos de Seguridad Informática:

Con el aumento de los ataques informáticos, el correo electrónico se ha convertido en un medio vulnerable para la empresa. Se han registrado casos de interceptación de facturas y ataques de phishing, lo que ha llevado a la filtración de datos confidenciales y a posibles pérdidas financieras y de reputación para la empresa.

Conclusiones:

Los desafíos en el envío de documentación y facturas a los clientes representan una amenaza para la eficiencia operativa, la satisfacción del cliente y la seguridad de la información en Emtel del Noroeste. Es fundamental implementar medidas para mejorar el seguimiento y control de la documentación enviada, así como fortalecer la seguridad informática para proteger la información confidencial de la empresa y sus clientes.

### Beneficios del desarrollo de una aplicación web

A la vista de los casos problemáticos comentados con anterioridad, se prevé que el desarrollo de una aplicación web para la gestión documental en la empresa puede contribuir de las siguientes formas a su resolución:

* Mediante el desarrollo de una aplicación web que facilite la recopilación, gestión y seguimiento de la documentación de las subcontratas, podremos asegurar la trazabilidad, cumplimiento normativo y la eficiencia en el proceso.
* El desarrollo de una aplicación web que facilite la captura y envío de la documentación de instalaciones, aumentara la agilidad y el seguimiento eficiente de la documentación.
* Permitirá centralizar el envío de documentación hacia los clientes, ofreciendo un seguimiento y control adecuados, programación de envíos automáticos y medidas de seguridad para proteger la información confidencial.

## Planteamiento del trabajo

La aplicación propuesta es una plataforma integral diseñada para facilitar la gestión de documentación, con un enfoque especifico en la resolución de los desafíos identificados en las fases de gestión de subcontratas, documentación de instalaciones y envío de documentación a clientes.

Para ello, la aplicación permite la creación por parte de personal de la empresa de formularios que recopilen uno o varios documentos que han de ser recopilados.

La posibilidad de configuración de estos formularios permitirá que la aplicación sea usada por los distintos departamentos de la empresa puesto que serán los usuarios quienes determinen los documentos que necesitan asi como sus características fundamentales.

El usuario podrá facilitar una breve descripción de las características que ha de cumplir el documento o permitir indicar la fecha de caducidad de este si fuese necesario.

Por otra parte, los destinatarios de dichos requerimientos accederán a una interfaz sencilla y concisa en la que se presentara una relación de todos los requerimientos pendientes de satisfacer.

Accediendo a cada una de ellas podrán cumplimentar la documentación solicitada o ver si existe alguna ya aportada que haya sido invalidada por parte del receptor.

Al disponer de un canal único para estas gestiones se eliminarán las posibilidades de extravío de documentación, falta de acceso a los mismos por uso de canales indebidos tales como correos personales, WhatsApp, etc. o incluso se contribuye a la seguridad al evitar el envío mediante canales más “públicos” de información sensible tales como facturas, que puede derivar en ataques de suplantación de identidad.

Otras de sus características principales son:

* Permite la captura y carga de documentación de instalaciones en tiempo real
* Permite la recopilación, almacenamiento y seguimiento de la documentación requerida a las subcontratas
* Facilita un canal de comunicación privado entre empresas, técnicos, subcontratas y clientes
* Creación de un portal de clientes donde estos puedan acceder a su documentación y facturas en cualquier momento, reduciendo el riesgo de reclamaciones por falta de recepción.
* Ofrece un seguimiento en tiempo real del estado de la documentación.

## Estructura del trabajo

A continuación, se detalla la estructuración del TFE tras el presente capitulo.

El Capítulo 2, titulado CONTEXTO Y ESTADO DEL ARTE se dedica a realizar un estudio sobre la importancia de la gestión documental y otras herramientas existentes en el mercado en dicho ámbito.

El Capítulo 3, titulado OBJETIVOS Y METODOLOGIA DE TRABAJO, se centra en la determinación de los objetivos principales y específicos que satisfará la aplicación. Igualmente se detallará la metodología seleccionada y sus diferentes elementos para la consecución de los objetivos seleccionados

El Capítulo 4, titulado DISEÑO Y EVALUACION DE LA PROPUESTA, detallara modelos y diagramas generados en el proceso de diseño de la aplicación junto con el desarrollo e interfaces gráficas generadas.

También se detallarán las diferentes pruebas realizadas para la validación del desarrollo.

Para finalizar, el Capítulo 5, denominado CONCLUSIONES Y TRABAJO FUTURO, expondrá las conclusiones y resultados obtenidos durante la realización del TFE y propondrá posibles ampliaciones o mejoras posibles sobre el mismo.

# Contexto y Estado del Arte

En el contexto actual de la era digital y la inmediatez, la gestión documental se ha convertido en un aspecto crucial para aquellas organizaciones que buscan optimizar los procesos y mejorar la eficiencia en el manejo de la información.

En este sentido, el desarrollo de aplicaciones web ha ganado relevancia como una solución efectiva que aporta celeridad y ubicuidad para abordar los desafíos asociados con la organización, acceso y seguridad de los documentos.

Según Chaffey y White (2011), la gestión documental desempeña un papel crucial en las empresas al permitir la organización eficiente de la información, facilitar la toma de decisiones basadas en datos y garantizar el cumplimiento de normativas legales y regulatorias. Además, señalan que una gestión documental efectiva contribuye a la optimización de procesos internos, la mejora de la colaboración entre equipos y la reducción de costos asociados con el almacenamiento y la administración de documentos.

## Necesidad de la optimización en la información

En este punto, es importante destacar la necesidad que tienen las grandes organizaciones de optimizar todos sus procesos, ya sean relacionados a la recolección, documentación y posterior difusión de la información o no. Cuando se habla de optimización, se habla desde los aspectos más básicos hasta los más específicos del proceso de toda organización. En ese sentido, teniendo en cuenta el tema principal que atraviesa este trabajo, el foco está puesto en la optimización de la información, su documentación y su posterior gestión.

En ese sentido, Jaramillo (2010) sostiene que, en la actualidad, siempre que se habla del manejo de grandes cantidades de información debe hacerse referencia a la informática; dado que se hace prácticamente imposible pensar en tales procesos de recolección, documentación y gestión sin tener que recurrir a los medios de carácter informáticos. De esta forma, el autor menciona a las computadoras, a los programas y a los equipos periféricos exclusivos para la recolección, el almacenamiento, el manejo y la distribución de las informaciones. La acelerada evolución que ha experimentado la informática en este aspecto obedece precisamente a las necesidades de poder brindar respuestas de forma adecuada a estos procesos que suelen presentarse como urgentes.

De esta forma, las organizaciones, cada vez manejan más cantidad de documentos por la gran cantidad de procesos que tienen y a la gran cantidad de información que estos necesitan durante su desarrollo. Es necesario poder aplicar técnicas, relacionadas a la gestión documental, a través de las cuales los sistemas que sean óptimos automaticen, estandaricen y gestionen de manera correcta y adecuada todo ese cúmulo de documentaciones y formularios con los que se trabajan a diario (Rodríguez-Muñoz y Gómez-Lorca, 2002).

Siguiendo con lo expuesto en el párrafo anterior, para los autores los documentos y la información suponen la base sobre la cual se sustentan las actividades organizacionales, dado que tienen esa información de carácter vital que es utilizada como una especie de materia prima en todos los procesos de la empresa. Por lo tanto, continuando con Rodríguez-Muñoz y Gómez-Lorca (2002), es algo lógico pensar que se necesita la aplicación de técnicas adecuadas relacionadas a la gestión a toda esta documentación para que pueda lograr y responder eficientemente ante las necesidades de la empresa.

### Importancia de la telemática

Cuando se destaca la relevancia de la necesidad de optimizar la información, es imprescindible destacar el rol que ha cumplido la telemática en ese aspecto en los últimos tiempos. Antes de destacar este rol, es importante esbozar una definición sobre este aspecto de la ingeniería que facilita la gestión de la documentación y la optimización de la información. En ese sentido, Jaramillo (2010) menciona que es “la combinación de la información almacenada y procesada en computadoras con los más modernos medios de transmisión a distancia (microondas, fibra óptica, satélites, televisión por cable e incluso televisión y teléfono tradicionales)” (p. 54).

La telemática brinda la posibilidad de establecer una serie de interfaces entre los sistemas de información ubicados desde cualquier parte del mundo o sitio, no solo a través de la transmisión de datos, sino también de textos y gráficos elaborados, de voz y de imágenes (Jaramillo, 2010). La introducción de la transmisión de datos por paquetes, de acuerdo con el autor, les da la oportunidad a la organización y a las personas en particular intercomunicarse con la información y acceder a bases de datos o a centros de información de todo el mundo con cierta precisión, agilidad, rapidez y economía que décadas atrás era impensado que suceda.

En suma, la telemática permitió dar un paso hacia adelante en lo que respecta a la optimización de la información. Infinidad de procesos y estructuras han ido cambiando a lo largo del tiempo y yendo inevitablemente hacia la digitalización, no solo de la información, sino también de los procesos, las estructuras y los sistemas. En ese sentido, a lo largo de los años, se han ido estandarizando muchos de esos procesos relacionados a la gestión informática y muchos otros aspectos más. En ese sentido, las normas ISO han cambiado rotundamente la dinámica en el mundo de los procesos.

## La digitalización de la gestión documental

Con la explosión de la globalización y el boom de las tecnologías a lo largo y a lo ancho del mundo, muchos aspectos de la vida en sociedad fueron atravesados por estos hitos, y la vida organizacional no es la excepción. Antes del surgimiento de los dispositivos electrónicos como las computadoras, la documentación se registraba y gestionaba en papel. Incluso, ya habiéndose establecido la computadora como un elemento recurrente en los ámbitos laborales, el papel seguía ocupando un gran porcentaje de los documentos.

Así, según Rodríguez-Muñoz y González-Laorca (2002), de acuerdo con distintos autores, a principios del siglo XXI, entre el 80 y el 90% de la información de las organizaciones residía en documentos de papel. Los autores destacan que el porcentaje de documentación relacionado al formato del papel sigue siendo elevado, provocando que la organización tenga que atravesar una serie de dificultades e inconvenientes relacionados a su uso. Estos son:

* Dificultad de acceso: dado que la documentación en papel suele guardarse en archivos que pueden no estar próximos al lugar donde se emplaza la organización, produciendo retrasos en su entrega, dado que la búsqueda puede ser manual (siempre que los criterios de recuperación sean los adecuados) y la demora en la distribución.
* Indisponibilidad: ya que puede suceder que, al solicitar un documento o información determinado, ya esté siendo usado por alguien más dentro de la organización.
* Deterioro: esto se debe a que la documentación en formato papel posee el gran inconveniente del deterioro, provocado por el paso del tiempo, su uso constante, su deficiente almacenamiento o las negligencias en su manejo, provocando que con el correr del tiempo pierda la calidad que presentaba originalmente.
* Pérdida: debido a que la documentación tiene muchas posibilidades de perderse o traspapelarse con otros documentos que se utilicen en simultáneo; así como también por colocarse en un lugar erróneo al devolverlo al archivo, presentándose el inconveniente de volver a restablecerlo.
* Redundancia: con el continuo uso de la documentación es normal que se hagan copias de ésta para que pueda ser usada por otras personas, pudiendo ser finalmente archivadas también junto con el original, lo que provoca un duplicado o un aumento del archivo.
* Dispersión: muchas veces, en la práctica, cada departamento de una organización puede terminar creando su propio archivo para poder gestionar la documentación que usan, por lo que es probable que se tengan dificultades al momento de consultar información de otros sectores, dado que puede tener un tratamiento distinto.
* Seguridad: si se tiene pensado establecer algún tipo de control en cuanto al acceso de los archivos en formato papel, todo debe quedar delegado en una persona encargada de custodiar dicho archivo, por lo que se puede decir que la seguridad es cuestionable.
* Espacio físico: una de las grandes desventajas de la documentación en formato papel está relacionada a la necesidad de requerir de archivos en locales con grandes dimensiones, dado que, con el tiempo, el volumen irá creciendo y será considerable.
* Costos de personal: estrechamente relacionado con el ítem anterior, el aumento de volumen en la documentación y en los archivos, hace que se necesite de más personal totalmente dedicado a su gestión y a su mantenimiento, aumentando indefectiblemente también su costo.

Por esto, es necesario destacar las ventajas que se desprenden de la digitalización en la gestión documental. De acuerdo con la CEM (Confederación Empresarial de Málaga, 2021), dentro de las muchas ventajas que tiene la digitalización en la gestión documental, se puede destacar los siguientes puntos:

* Acceso: permite acceder a la documentación de manera remota, desde cualquier lugar o dispositivo.
* Ahorro: teniendo en cuenta la reducción de todos los tiempos que se invierte en la gestión interna, externa y con la administración (relacionado a la creación de búsqueda, archivo, envío, etcétera) contribuye a un importante ahorro en cuanto a los costos.
* Almacenamiento: dado que se puede reducir el espacio de almacenamiento de la información y los documentos, así como también los costos asociados a su gestión y también a su mantenimiento.
* Capacidad de compartir: teniendo en cuenta que facilita poder compartir los archivos y documentos sin tener que sobrecargar otros sistemas como la mensajería electrónica.
* Construcción de una marca: ya que, una respuesta rápida y eficiente a los clientes y a los proveedores aporta a la generación de una buena imagen de la organización como una marca.
* Control y seguridad: porque posibilita discriminar el acceso de acuerdo con los rangos o a la acreditación de los usuarios, lo que transforma en un incremento de seguridad, así como también ante las pérdidas, los extravíos o las manipulaciones inadecuadas.

## Gestión documental

De acuerdo con Martínez Sánchez e Hilera González (1997), la gestión documental se puede definir como el “control automatizado de documentos electrónicos a través de su ciclo de vida completo en una organización, desde su fase de creación hasta que se archiva en su ubicación final” (p. 238-239). En otras palabras, la documentación de la organización se gestiona para que sea útil para sus procesos y, de esta forma, alcanzar los objetivos propuestos. Según Rodríguez-Muñoz y Gómez-Lorca (2002), dentro del ámbito de la gestión documental existe la creencia de que, al eliminar el papel, se elimina con ello todos los problemas que puedan desprenderse de su uso, aunque lo que realmente sea relevante no es si la información se encuentra o no en formato electrónico, sino que lo verdaderamente importante es su gestión.

Si bien para los autores la reducción del papel es importante, ese no es el verdadero propósito, sino que es conseguir una tecnología que pueda permitir una gestión efectiva y eficiente de la información que la organización tiene. Es por ello por lo que los Sistemas de gestión documental brindan esa posibilidad de gestión que es necesaria, a través de la conjunción de las técnicas documentas con las TIC.

### Sistemas de gestión documental (SDG)

Los SGD se pueden entender como “aquellos sistemas encargados de gestionar y tratar todos sus aspectos la información fijada en un soporte” (Rodríguez-Muñoz y Gómez-Lorca, 2002; p. 19). Por ello, el concepto de documento adquiere un sentido un tanto más amplio al considerar como tal, toda la información que se haya plasmado en cualquier tipo de soporte ya sea en papel, magnético o electrónico. Asimismo, los SGD han experimentado últimamente un auge dentro de todo tipo de organizaciones, gracias a la necesidad de gestionar de forma eficiente que estas tienen sobre la gran cantidad de documentos. Ello sumado a los avances tecnológicos que han fomentado que este tipo de herramientas tengan a su disposición una serie de características que loas convierten en aplicaciones con una eficiencia increíblemente alta a la hora de gestionar la información en la organización.

Dentro de las múltiples características y herramientas que los SGD pueden aportar, Rodríguez-Muñoz y Gómez Lorca (2002) destacan los siguientes:

* Creación de documentos: se puede crear desde cero el modelo de datos que puede servir como base de la gestión dentro del sistema, teniendo en cuenta que se podrá tener campos de datos y de información que identifica a cada uno de los documentos o formularios almacenados.
* Captura e incorporación: permite que a través de dispositivos OCR (reconocimiento óptico de caracteres) se capturen datos e imágenes de todo tipo. Además, se pueden importar distintos formatos de ficheros para poder incluirlos dentro del sistema. Todos estos documentos quedan vinculados entre sí y a los que ya habían sido creados a través del modelo de datos.
* Archivo: todos los documentos que se hayan ido creando, además de los que se han incorporado al sistema, tendrán la oportunidad de ser almacenados en la base de datos del sistema, brindando determinados índices para alcanzar una mejor recuperación de estos.
* Búsqueda y recuperación: estos sistemas permiten la búsqueda de documentos con base en el uso de los índices que, básicamente, son bases de datos en las cuales se encuentran los términos y las indicaciones de cada documento en el que se encuentran. Así, de esta forma, poder localizar un dato o información es mucho más rápida. Una vez que el documento es localizado, puede visualizarse, guardarse o bien puede ser imprimido.
* Seguridad: otra de las ventajas de los SGD tiene que ver con la posibilidad de establecer restricciones y controles de acceso a través de la creación de niveles de seguridad para cada uno de los tipos de usuarios que se han definido, con lo que es posible evitar cualquier manipulación que no es deseada.
* Administración: siempre es importante que el SGD tenga bien presente la figura del administrador del sistema, quien es el encargado de la gestión y las tareas relacionadas a la modificación de los archivos, a las gestiones con los usuarios, a la propia seguridad del sistema, a su configuración y copias de seguridad, entre otras tareas.
* Integración con aplicaciones: comprende una gran utilidad que un SGD tenga la posibilidad de integrarse con otro tipo de aplicaciones, dado que pueden ser un buen complemente para el trabajo. Por caso, se pueden mencionar al correo electrónico para el envío de documentos y/o comunicación, bases de datos que permitan importar o exportar dichos documentos, servicios de red, entre otros.

En concordancia con estas herramientas y características, Rodríguez-Muñoz y Gómez-Lorca (2002) destacan que, gracias al nivel de desarrollo que pueden alcanzar estos sistemas, los beneficios que los mismos presuponen en su implementación en una organización son claros y diversos. Los autores destacan los siguientes:

* Reducción drástica en cuanto al espacio del archivo.
* La posibilidad de tener un acceso de manera recurrente a un mismo archivo, con lo que el conocimiento se crea una sola vez y luego puede ser reutilizado cuando se requiera.
* La chance de tener disponibilidad inmediata de la documentación gracias a las herramientas relacionadas a la búsqueda a y a la recuperación.
* Establecimiento de varios niveles de seguridad dentro de la organización con la capacidad de proteger la información y la restricción a ciertos niveles no autorizados.
* Integración con aplicaciones de usuario final, por lo que la necesidad de aprendizaje por parte del usuario es ínfima.
* Utilización de muchos formatos de los archivos, tendiendo a aprovecharlos de mejor manera e incluyendo de una forma más fácil el sistema en general, etcétera.

### La estandarización en los sistemas de gestión

De acuerdo con el Gobierno de México (2015) la estandarización se puede entender como “el proceso de ajustar o adaptar características en un producto, servicio o procedimiento; con el objetivo de que estos se asemejen a un tipo, modelo o norma en común” (párr. 2). En ese sentido, la estandarización permite que se puedan crear normas o estándares que determinan cuáles son las características que son comunes a todos los procesos, y que estos deben cumplir y respetar sin excepción dentro de las jurisdicciones adheridas a esa estandarización. Por ello, se constituye como una actividad técnica y especializada.

En ese sentido, la Universitat Oberta de Catalunya (UOC, s.f.) menciona y destaca algunas normas técnicas que estandarizan la gestión documental en diferentes aspectos. En primer lugar, se destaca la ISAD(G) que es la Norma Internacional General de Descripción Archivística (General International Standard Archival Description, en inglés). Esta norma se destaca por ser una guía a nivel general para la elaboración de descripciones relacionadas al fondo de archivo. El objeto principal de esta norma es el de identificar y explicar el contexto y el contenido que el documento archivado tiene, con el fin de facilitar el acceso a la misma. Esto se alcanza a través de la creación de representaciones precisas y adecuadas, organizando a estas representaciones a partir de modelos que son predeterminados. Los procesos que son relativos a la descripción se pueden iniciar al momento o antes de la producción del archivo, continuando a través de su ciclo vital.

Otra de las normas que menciona la UOC (s.f.) es la ISO 15489, una norma de calidad que está orientada a la gestión de los documentos, la gestión integral de un sistema de archivo y los mismos documentos de archivo. El objetivo principal de esta norma es el de proporcionar las directrices adecuadas sobre las políticas en torno a la gestión de documentos y la asignación de las responsabilidades para con los documentos que una organización posee. Por esto, se centra en la importancia relacionada a la supervisión, al control y a la auditoría, con el fin de que las organizaciones se puedan ajustar a las distintas normativas y disposiciones legales que son de cumplimiento obligatorio; valoren lo considerables que son los documentos como un elemento de prueba y una evidencia de su actuación; y optimicen el propio rendimiento y la eficiencia en lo que respecta a la gestión documental.

Por último, se menciona al modelo de requisitos para la gestión de documentos electrónicos y archivo (Model Requirements for the management of Electronic Documents and Records, en inglés), que comprende una serie de requisitos que la organización debe cumplir para el ordenamiento de archivos electrónicos, desarrollado dentro del marco de la Unión Europea. Este modelo se suele tomar como un enfoque opcional de la norma de gestión de documentos ISO 15489. El MoReq2 considera la evolución tecnológica y el desarrollo de nuevas normativas y buenas prácticas luego de que se haya publicado el modelo de exigencias para la organización de archivos electrónicos (MoReq).

#### Norma ISO 30300:2011 y 30301:2011

A pesar de haber mencionado y destacado varias normas relacionadas a la estandarización en la gestión documental y otros aspectos, a los fines del presente trabajo, se tendrá en consideración a la serie de normas ISO 30300 dado que, Sánchez (2014) destaca que esta norma ofrece una metodología para poder implementar un SGD que tenga su base en un enfoque sistémico para poder crear y gestionar documentos. Esto es crucial, dado que está alineado con los objetivos y con las estrategias que diagrama la organización, estableciendo requisitos y no directrices como lo hace la norma ISO 15489.

• ISO 30300:2011: se establece los SGD, sus fundamentos y el vocabulario a usar. En esta parte de la norma se explica la base lógica en cuanto a la creación de estos SGD y los principios que lo van a regir para que su implementación sea exitosa. Además, proporciona la terminología que asegura que el sistema sea compatible con otras normas relacionadas a los sistemas de gestión.

• ISO 30301:2011: se establecen los SGD y sus requisitos: en esta parte de la norma se concretan los requisitos que son indispensables, necesarios y fundamentales para poder desarrollar una política de gestión documental. Asimismo, también se establecen objetivos y metas para que la organización pueda implementar algunas mejoras de índole sistemática, consiguiéndose, así, mediante el diseño de procesos y aplicaciones (tecnología) de gestión documental, estimando la asignación apropiada de los recursos, así como también estableciendo puntos referenciales para controlar, medir y evaluar los resultados.

Por otro lado, respecto a la aplicación de esta norma, siguiendo con Sánchez (2014), resalta su utilización para poder conseguir una certificación por parte de una empresa que sea certificadora. De acuerdo con el autor, muchas organizaciones a nivel global diagraman estrategias para poder alcanzar estas certificaciones, pudiéndose vincular o desarrollarse a la vez, o en su defecto, permitiendo la certificación de varias normas ISO al mismo tiempo, como pueden ser la ISO 9001 y la ISO 14001 además de la ISO 30301, teniendo en cuenta que algunos de los requisitos que se deben alcanzar son comunes entre una familia de normas y las otras.

Por ello, tal y como indica la norma internacional ISO 30301:2011 "La gestión documental es el campo de la gestión de la información que trata la planeación, control y organización de los documentos de una organización, incluyendo la adquisición, almacenamiento, recuperación y disposición de estos, a fin de satisfacer los requerimientos legales, administrativos, operativos y culturales de la organización respecto a la información." (ISO 30300:2011).

En la actualidad, dado el considerable aumento de la presencia de documentos digitales que las empresas han de gestionar, hacen necesaria la existencia de políticas o normas, tales como las ISO 30300:2011

Hemos de entender documentos digitales como “la información que está almacenada en formato electrónico, en un medio electrónico, y que se necesita un dispositivo informático para acceder a ella se conoce como documento digital. Puede contener una variedad de elementos como texto, imágenes, gráficos, video, sonido, entre otros” (Risso, 2012).

### Requisitos de la gestión documental

La norma ISO 30300:2011, centrada en la regulación de la gestión documental, establece una serie de requisitos para un sistema de gestión de documentos eficaz.

Estos requisitos se centran en garantizar la capacidad de la organización para crear, capturar, gestionar y preservar los documentos de forma efectiva y acorde a las normativas y regulaciones aplicables.

Algunos de estos requisitos clave son:

* Política de gestión de documentos: La organización debe establecer y mantener una política documentada que defina sus objetivos y compromisos en materia de gestión de documentos.
* Planificación: La organización debe desarrollar un plan de gestión de documentos que incluya objetivos, metas, recursos necesarios y procesos para alcanzar los resultados deseados.
* Apoyo: Se deben proporcionar recursos adecuados, incluyendo personal capacitado, tecnología e infraestructura, para apoyar la implementación y mantenimiento del sistema de gestión de documentos.
* Operación: La organización debe establecer y mantener procedimientos documentados para la creación, captura, indexación, almacenamiento, acceso, recuperación y disposición de los documentos.
* Control de la documentación: Se deben establecer controles para garantizar la identificación, revisión, aprobación, distribución, acceso y uso adecuado de los documentos, así como la protección de la información confidencial.
* Mejora continua: La organización debe realizar el seguimiento y la revisión periódica del sistema de gestión documental, con el fin de identificar oportunidades de mejora y tomar medidas correctivas según sea necesario.

### Etapas de la gestión documental

En la norma ISO 15489-1:2001 se detalla un marco compuesto por diversas etapas que deben ser seguidas para la gestión efectiva de documentos.

Estas fases incluyen:

* Planificación inicial del sistema,
* Recopilación y creación de documentos,
* Indexación y clasificación,
* Almacenamiento seguro.

Posteriormente, se establecen procesos para la recuperación y acceso oportunos a la información, seguidos de la retención y disposición adecuadas de los documentos.

Además, se destaca la importancia de la preservación a largo plazo de los documentos valiosos para la organización, junto con la realización de auditorías y revisiones periódicas para garantizar la eficacia del sistema (Organización Internacional de Normalización, 2001).

## Software para la Gestión Documental

Los cambios en las necesidades empresariales, junto con los avances tecnológicos y la creciente adopción de internet en las organizaciones han impulsado una evolución significativa en el ámbito de la gestión de la documentación en las empresas (Smith, 2018, p. 45).

### Historia del software de gestión documental

A continuación, se nombran algunos de los hitos más importantes en lo relativo a la historia del software de gestión documental (Smith, 2018, p. 45).

* **Inicio y Desarrollo Temprano:** La gestión documental tiene sus raíces en las primeras formas de almacenamiento y organización de documentos en papel. Antes de la era digital, las empresas utilizaban sistemas de archivos físicos y archivadores para almacenar y recuperar documentos.
* **Digitalización:** Con la llegada de las computadoras y la digitalización de la información en la década de 1970, surgieron los primeros sistemas de gestión documental electrónica (GED). Estos sistemas permitían escanear, almacenar y organizar documentos en formato digital.
* **Sistemas de Gestión de Documentos:** En la década de 1980, los sistemas de gestión de documentos (DMS) comenzaron a ganar popularidad. Estos sistemas ofrecían capacidades más avanzadas, como la indexación y búsqueda de documentos, control de versiones y gestión de flujos de trabajo.
* **Desarrollo de Aplicaciones Web:** Con la proliferación de Internet en la década de 1990, surgieron las primeras aplicaciones web para la gestión documental. Estas aplicaciones permitían a los usuarios acceder y gestionar documentos a través de un navegador web, facilitando el acceso remoto y la colaboración en línea.
* **Integración de la Nube:** En los últimos años, muchas soluciones de gestión documental han migrado a entornos basados en la nube. Esto ha permitido un acceso más flexible a los documentos y ha facilitado la colaboración en tiempo real entre usuarios ubicados en diferentes lugares.
* **Enfoque en la Colaboración y la Productividad:** Las aplicaciones modernas para la gestión documental se centran en mejorar la colaboración y la productividad dentro de las organizaciones. Ofrecen características como la edición colaborativa en tiempo real, la automatización de flujos de trabajo y la integración con otras herramientas empresariales.
* **Inteligencia Artificial y Automatización:** La integración de tecnologías como la inteligencia artificial y el aprendizaje automático está transformando la gestión documental. Estas tecnologías se utilizan para automatizar tareas repetitivas, clasificar documentos automáticamente y extraer información relevante.

### Actualidad de las herramientas de gestión documental

Con relación a las múltiples herramientas que se pueden usar relacionadas a la gestión documental, Honig (2023) plantea siete que son necesarias en toda organización que busca administrar y ahorrar tiempo y dinero. En ese sentido, destaca las siguiente:

• Firma electrónica: hace referencia a cualquiera de los procesos a través de los cuales se puede firmar, de forma digital, cualquier documento. El uso de estas firmas electrónicas son una práctica habitual en la actualidad porque son consideradas legalmente válidas en muchas situaciones. Para la autora, exigir firmas en papel puede resultar sumamente ineficaz y tedioso. Por un lado, el proceso requiere de una copia física del documento a firmar, teniéndose que enviar por correo o entregar al firmante. Por ello, al usar firmas digitales, se eliminan una serie de pasos que son innecesarios, reduciendo significativamente la chance del error humano durante el proceso de firma.

• Acceso remoto a documentos importantes: dado que, al usar un gestor de documentos empresariales, todos los documentos que sean relevante se almacenan en una ubicación central, con acceso desde cualquier lugar, siempre que se tenga el privilegio necesario. Es decir, un documento puede ser abierto siempre que alguien de la organización lo necesite y tenga el acceso adecuado, sin tener que perder tiempo en la solicitud a otra persona.

• Contrataciones, solicitudes de empleo e incorporaciones: todos estos procesos se agilizan de forma considerable con un gestor de documentos. Estos sistemas de gestión de contenido empresarial incluyen herramientas para poder procesar de forma digital solicitudes de, por ejemplo, contratación, concentrando todo en un único portal. Por otra parte, los formularios web brindan una manera sencilla de recopilar datos precisos. Se puede asegurar que la información es correcta, dado que se ha transferido la entrada de datos a la fuente de información.

• Gestión de facturas: con la automatización y digitalización de este proceso se puede capturar e indexar las facturas antes de ser enviadas a través de flujos de trabajo configurados con anterioridad para aprobar, pagar y archivar un repositorio centralizado en el que se puede navegar y realizar búsquedas.

• Integración con el correo electrónico y otras aplicaciones: como una solución integral dentro de la gestión documental, estos sistemas permiten almacenar y acceder a todo tipo de datos y de archivos en un solo y único lugar, pero también tiene muchas otras ventajas. Los programas de correo electrónico están diseñados para ofrecer comunicación, no para el almacenamiento organizado, la seguridad o la búsqueda rápida. Por ello, si cuesta encontrar la información en un correo electrónico, es necesario una solución automatizada que incluya una herramienta que gestione el correo y se integre con otros programas.

• Flujos de trabajo automáticos: son necesarios ya que eliminan los procesos que son manuales de las tareas empresariales de carácter crítico, permitiendo de esa forma aplicar reglas y enviar documentos de manera automática. Si se realiza de otra manera, esta información podría quedarse fuera del alcance de quien la necesite.

• Procesos sin papel: estas ventajas son ilimitadas. El acceso remoto a los documentos garantiza el futuro de una organización dentro de una economía que tiene una tendencia creciente al trabajo remoto a distancia, lo que permite a los usuarios trabajar desde cualquier lugar y horario, maximizando su contribución a los objetivos mismos de la organización.

En la actualidad, las herramientas han evolucionado notablemente mediante la incorporación de los principales avances tecnológicos del sector.

El uso de inteligencia artificial para la automatización de tareas repetitivas como la clasificación o extracción de datos permite un ahorro de tiempo y disminuye la presencia del factor “error humano” en ellas.

Al igual que sucede en muchas otras herramientas, la tendencia del Cloud Computing ha surgido con fuerza entre las herramientas de gestión documental, facilitando tanto la movilidad como la colaboración remota de usuarios y la disponibilidad de la información gestionada.

La gestión documental también se ha hecho eco de otras tendencias en la gestión de la información como el Data Science ya que, hemos de recordar que el objetivo final de la gestión documental es el control y obtención de información a partir de estos documentos y por tanto, una herramienta de gestión documental dotada de capacidad de análisis, aumentará la eficiencia operativa y permitirá una toma de decisiones ágil en las empresas gracias a la información obtenida por la propia herramienta sin necesidad de una posterior gestión del documento.

Existen en el mercado diversas propuestas de herramientas creadas por las principales empresas de software que, si bien no han sido enfocadas única y exclusivamente hacia la gestión documental en la empresa, representan una gran opción para muchas organizaciones que optan por el uso de herramientas más genéricas pero que debido a su popularidad no necesitan de formaciones o conocimientos específicos para su uso.

De igual forma, existen en el mercado opciones más específicas, denominadas ECM.

Según Smith (2020), un ECM o sistema de Gestión de Contenidos Empresariales es “una plataforma integrada que se utiliza para gestionar de manera efectiva toda la información no estructurada de una organización, incluyendo documentos, registros, correos electrónicos, imágenes y otros tipos de contenido. Estos sistemas proporcionan funciones para capturar, almacenar, organizar, recuperar y distribuir el contenido de manera eficiente, facilitando la colaboración entre equipos y garantizando el cumplimiento normativo."

La implementación exitosa de un ECM implica tanto la adquisición de la tecnología adecuada, como también la comprensión de las necesidades específicas de la organización.

Por tanto, aunque tanto los sistemas de Gestión de Contenidos Empresariales (ECM) como herramientas genéricas como SharePoint pueden utilizarse para gestionar documentos y contenido empresarial, existen algunas diferencias clave entre ellos:

1. Enfoque y funcionalidades especificas:

Los sistemas ECM están diseñados específicamente para la gestión integral de contenido empresarial. Por el contrario, SharePoint y otras herramientas colaborativas presentan un enfoque más amplio e incluyen características de colaboración, gestión de proyectos, intranet y aplicaciones web que pueden resultar innecesarias o no encajar totalmente con la estructura y procedimientos de nuestra organización en concreto.

1. Complejidad y Flexibilidad:

Las herramientas colaborativas son generalmente más fáciles de implementar y usar en comparación con los ECM que suelen ser sistemas mucho más complejos, pero ofrecen a cambio una mayor flexibilidad en lo referente a la personalización y configuración para su adaptación a una organización en concreto.

1. Cumplimiento Normativo y Seguridad:

Los ECM, debido al conocimiento específico del sector, suelen poner especial énfasis en el cumplimiento normativo y la seguridad de la información mediante el uso de funciones como el control de versiones, la auditoria de accesos o el cifrado de datos.

Por su parte, las herramientas colaborativas pueden requerir complementos o configuraciones adicionales para satisfacer estas necesidades.

1. Escalabilidad y capacidad de gestión:

Mientras que los ECM son diseñados para el manejo de grandes volúmenes de datos, SharePoint y herramientas similares pueden no ser tan robustos en implementaciones para organizaciones con necesidades de gestión de contenido empresarial a gran escala.

#### Microsoft SharePoint

Logotipo, nombre de la empresa

Descripción generada automáticamente

Figura 4: Logotipo de Microsoft SharePoint (Fuente: Microsoft)

SharePoint es una plataforma de colaboración empresarial que permite a las organizaciones almacenar, organizar y compartir información de manera efectiva.

Según Microsoft (2022), SharePoint proporciona una amplia gama de funciones y herramientas, incluyendo sitios web de equipos, bibliotecas de documentos, flujos de trabajo automatizados y capacidades de búsqueda avanzadas.

Los usuarios pueden colaborar en documentos en tiempo real, acceder a información desde cualquier ubicación y dispositivo, y mantener un registro de versiones para un control de cambios eficaz. Además, SharePoint ofrece integración con otras aplicaciones de Microsoft, como Microsoft Office, Outlook y Teams, lo que facilita la colaboración y la productividad en el entorno empresarial.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 5: Imagen de SharePoint (Fuente: SharePoint)

SharePoint es una herramienta de pago disponible como servicio de suscripción, pero está incluida dentro de una serie de herramientas ofrecidas por Microsoft en sus paquetes denominados Microsoft 365, el cual se incluye conjuntamente en muchas modalidades de contratación de cuentas de correo electrónico empresarial.

Por tanto, se trata de una herramienta con amplia difusión gracias a su relación con la opción de correo electrónico de la misma compañía y es frecuente su aprovechamiento para tareas relacionadas con la gestión documental en pequeña y mediana empresa.

Además, esta herramienta permite su vinculación con otros productos de dicho paquete tales como Power Automate o Power Apps, dando pie a la creación de aplicaciones tales como intranets o formularios de recogida de documentación sin la necesidad de realizar desarrollos propios desde 0.

Entre las características más destacables de Microsoft SharePoint resaltan:

* **Bibliotecas de documentos:**

SharePoint permite crear bibliotecas de documentos para almacenar y organizar archivos de manera estructurada. Las bibliotecas de documentos pueden incluir metadatos personalizados, versiones de documentos y flujos de trabajo para facilitar la gestión y colaboración en documentos.

Control de versiones: SharePoint ofrece control de versiones para documentos, lo que permite a los usuarios mantener un historial de cambios y revertir a versiones anteriores si es necesario. Esto es útil para rastrear la evolución de un documento y mantener la integridad de la información.

* **Flujos de trabajo:** SharePoint incluye capacidades de flujo de trabajo que permiten automatizar procesos empresariales relacionados con la gestión documental, como la aprobación de documentos, la revisión y la publicación de contenido.

Los flujos de trabajo ayudan a agilizar los procesos y mejorar la eficiencia operativa.

* **Integración con Microsoft Office:** Se integra estrechamente con aplicaciones de Microsoft Office, como Word, Excel y PowerPoint, lo que facilita la creación, edición y colaboración en documentos directamente desde SharePoint.
* **Búsqueda avanzada:** SharePoint ofrece capacidades de búsqueda avanzada que permiten a los usuarios encontrar rápidamente documentos y contenido relevante dentro de la plataforma. Los usuarios pueden realizar búsquedas basadas en palabras clave, metadatos y otros criterios para encontrar la información que necesitan de manera eficiente.
* **Seguridad y permisos:** SharePoint ofrece controles de seguridad y permisos granulares que permiten a los administradores controlar quién tiene acceso a qué contenido y qué acciones pueden realizar los usuarios en los documentos.

Esto ayuda a proteger la información confidencial y garantizar el cumplimiento normativo.

#### Nextcloud

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Figura 6: Logotipo de Nextcloud (Fuente: Nextcloud)

Nextcloud es una plataforma de software de código abierto que proporciona servicios de almacenamiento en la nube y colaboración en línea.

Sus opciones pasan desde la versión comunitaria, que puede ser desplegada de forma gratuita haciendo uso de infraestructura propia, hasta las versiones de pago que ofrecen servicios de suscripción y la posibilidad de eliminar la necesidad de equipamiento.

Ofrece una amplia gama de características, incluyendo almacenamiento en la nube, sincronización de archivos, compartición de archivos, herramientas de colaboración, seguridad avanzada y aplicaciones integradas como calendario y contactos.

Dado que se trata de una herramienta de código abierto y gratuita en caso de ser desplegada y mantenida con recursos propios, puede ser empleada por las empresas como plataforma para un nivel básico de gestión documental, aunque no ofrece ciertas características deseadas para la misma tales como la posibilidad de validación o de creación de formularios específicos para la solicitud de tipos concretos de documentación.

Aunque Nextcloud es una opción sólida para la gestión documental, especialmente para aquellas organizaciones que valoran la flexibilidad y la integración con otros servicios, puede tener algunas limitaciones en comparación con herramientas más específicas en términos de funcionalidades, usabilidad y soporte especializado.

Captura de pantalla de computadora

Descripción generada automáticamente

Figura 7: Imagen de Nextcloud (Fuente: Elaboración propia)

A continuación, se detallan algunas de las principales funcionalidades de la herramienta Nextcloud:

* **Compartición de archivos segura:** La herramienta permite a los usuarios compartir archivos y carpetas de forma segura con colegas internos y externos, con opciones de permisos granulares para controlar quién puede acceder, editar y compartir archivos.

Captura de pantalla de un celular

Descripción generada automáticamente

Figura 8: Compartición de archivos (Fuente: Elaboración propia)

* **Control de versiones:** La plataforma mantiene un historial de versiones de archivos, lo que facilita la recuperación de versiones anteriores y el seguimiento de los cambios realizados en los documentos.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 9: Control de versiones (Fuente: Elaboración propia)

* **Extensibilidad y personalización:** La plataforma es altamente personalizable y extensible, lo que permite a los usuarios adaptarla a sus necesidades específicas y agregar nuevas funcionalidades a través de complementos y aplicaciones de terceros tales como aplicaciones de productividad, sistemas de gestión del conocimiento y más.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 10: Complementos (Fuente: Elaboración propia)

En resumen, Nextcloud es una opción popular para usuarios individuales y organizaciones de todos los tamaños que busquen una solución básica que pueda ser gestionada y albergada por recursos de la propia organización evitando una excesiva inversión.

En lo referente a los recursos necesarios, estos son los requisitos mínimos para el despliegue de la denominada Community versión:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Figura 11: Requisitos mínimos (Fuente: Nextcloud)

#### Metacontratas

Interfaz de usuario gráfica

Descripción generada automáticamente

Figura 12: Logotipo de Metacontratas (Fuente: Metacontratas)

MetaContratas es un software de Coordinación de Actividades Empresariales (CAE) desarrollado por la empresa española Metadata, que tiene como objetivo facilitar la gestión de la documentación de las empresas y empleados subcontratados en diferentes áreas, como Prevención de Riesgos Laborales (PRL), Recursos Humanos (RRHH) y Seguridad Social (SS).

Se trata de una aplicación web disponible en la modalidad de pago por uso y su contratación ofrece servicios adicionales tales como la validación de la documentación de PRL subida a la plataforma por personal cualificado de la empresa Metadata, lo que evita la necesidad de disponer de estos recursos por parte de la empresa contratante.

MetaContratas funciona mediante la digitalización de documentos y ofrece ventajas como la automatización de documentos, validación urgente, disponibilidad desde cualquier dispositivo con conexión a Internet y seguridad en el almacenamiento de la documentación de proveedores.

Dada la especificidad de dicha herramienta, dispone de numerosas plantillas para la gestión de tramites específicos de este ámbito asi como la garantía de cumplimiento de la legislación vigente de la documentación gestionada.

Ofrece posibilidad de personalización de la documentación solicitada, aunque siempre dentro del ámbito del CAE y la prevención de riesgos, lo que impide su uso para otras actividades que requieran intercambio documental como pueden ser las instalaciones o aportación de documentación a clientes por lo que en el entorno planteado por el presente trabajo seria necesario su complementación con otras herramientas.

Además, permite a las subcontratas gestionar y controlar la documentación solicitada mediante un acceso específico para cada una en el que revisar fechas de caducidad, documentación pendiente y otras opciones.

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

Figura 13: Imagen de Metacontratas (Fuente: Metacontratas)

Entre sus características principales destacan:

* **Gestión Documental:** Permite automatizar y simplificar la gestión de documentos relacionados con la subcontratación mediante el uso de plantillas predefinidas para la solicitud de documentación a contratas y la validación de los documentos aportados.

Interfaz de usuario gráfica

Descripción generada automáticamente

Figura 14: Configuración de documentos solicitados (Fuente: Metacontratas)

* **Control de accesos:** Permite supervisar los accesos a centros de trabajo y las horas trabajadas por los empleados subcontratados de forma sencilla permitiendo conocer quien dispone de toda la documentación correspondiente a un acceso especifica debidamente aportada y validada.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

Figura 15: Listado de accesos a un centro de trabajo (Fuente: Metacontratas)

* **Validación de documentación:** Confirma la validez de la documentación aportada y crea notificaciones al usuario concreto en caso de errores o caducidades de esta para poder conocer en todo momento la documentación pendiente.

Tabla

Descripción generada automáticamente

Figura 16: Listado de documentación validada (Fuente: Metacontratas)

En definitiva, Metacontratas es una solución muy adecuada para la coordinación de actividades empresariales que aporta beneficios próximos a la externalización de estas tareas.

Sin embargo, su uso se restringe a un ámbito muy concreto y no tiene flexibilidad para su adaptación a otros entornos lo que la hace incapaz de gestionar la totalidad de las tareas referentes a la gestión documental en la organización además de suponer un coste que podría ser asumido por los recursos propios de la empresa.

#### Logotipo Descripción generada automáticamenteDocuWare

Figura 17: Logotipo de DocuWare (Fuente: DocuWare)

DocuWare es un sistema de gestión documental (DMS) y gestión de procesos empresariales (BPM) que se define como "una plataforma líder en la nube y en las instalaciones de software como servicio (SaaS) para la gestión documental y la automatización de procesos empresariales" (DocuWare, 2022).

Esta plataforma permite a las organizaciones digitalizar y automatizar sus procesos de negocio mediante la gestión eficiente de documentos y flujos de trabajo.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 18: Imagen de DocuWare (Fuente: DocuWare)

Entre sus principales funciones y características se incluyen:

* **Captura y organización de documentos:**

Permite digitalizar y capturar documentos en papel y electrónicos desde una variedad de fuentes, como escáneres, correos electrónicos y aplicaciones móviles.

* **Indexación sin entrada manual de datos:**

La plataforma es capaz de leer automáticamente el contenido de los documentos y dispone de capacidades de aprendizaje automático que le permiten clasificar los archivos recibidos en base a la propia información que estos contengan sin necesidad de realizar procesos manuales.

* **Conexión a clientes de correo electrónico:**

Permite la obtención de documentos asociados a conversaciones de correo electrónicos desde el propio cliente, siendo posible su integración con los principales proveedores de servicios de correo electrónico como Outlook, Gmail u otros.

* **Formularios web para captura de datos:**

DocuWare permite la creación de formularios basados en la web que pueden ser compartidos mediante URL para facilitar la captura de información. Estos formularios también pueden ser incluidos en intranets o páginas web existentes dotándolas de una conexión con el sistema de gestión documental.

* **Gestión de tareas y procesos complejos:**

El sistema permite la automatización de tareas complejas de toma de decisiones en torno a los documentos gestionados pudiendo automatizar procesos relativos a recursos humanos, facturación o ventas mediante un diseñador visual que permite definir lo que debe suceder tras la recepción de un tipo concreto de documento.

#### Comparativa

La siguiente figura presenta una comparativa entre algunas de las herramientas más destacadas en el ámbito de la gestión documental, que han sido consideradas como posibles opciones para abordar las necesidades identificadas que motivaron la creación del sistema propuesto en el presente trabajo.

Se detallan sus principales ventajas y desventajas para proporcionar una visión completa de las opciones disponibles.

Además, se analizan las ventajas y desventajas del sistema propuesto en este TFG, lo que permite una evaluación exhaustiva y una comparación con las soluciones existentes en el mercado.

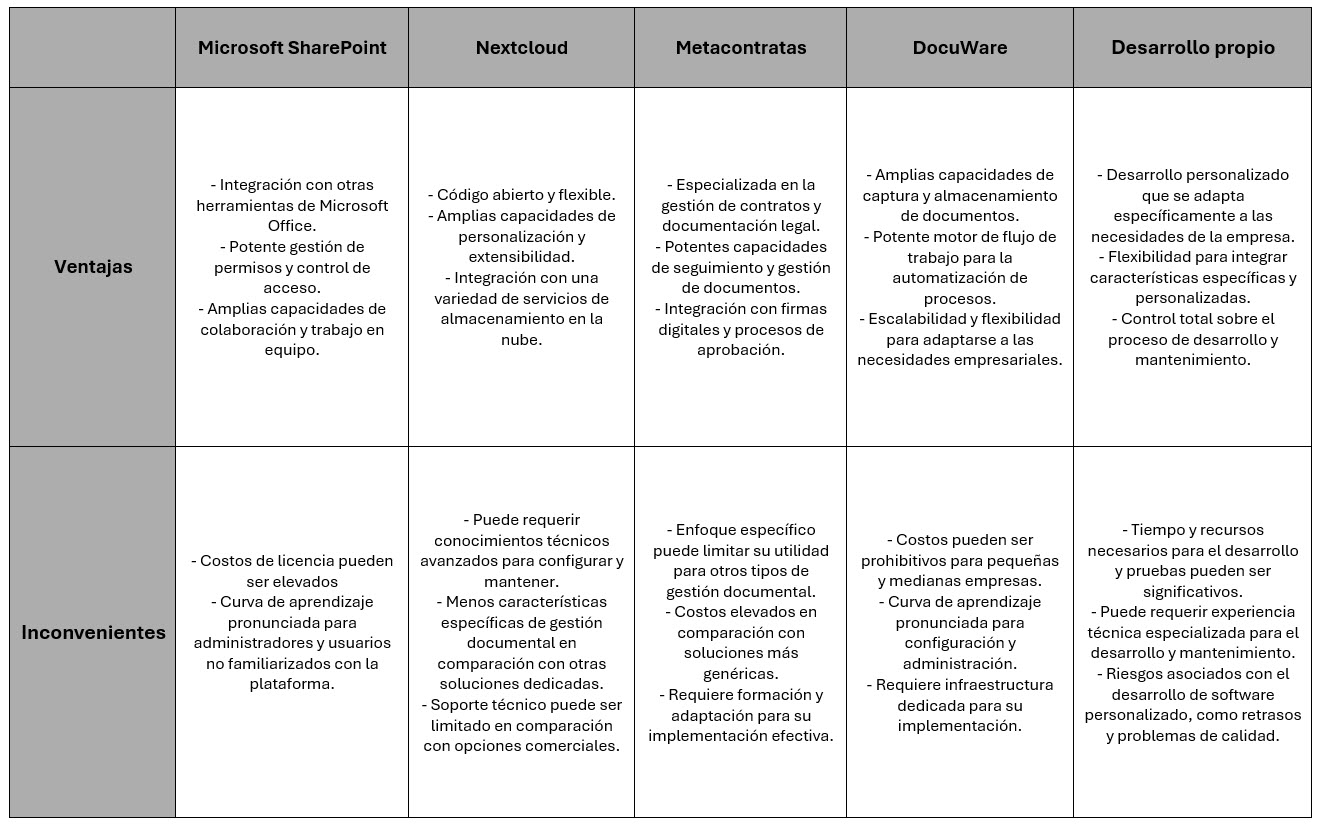


Tabla 1: Comparativa de las soluciones presentadas (Fuente: elaboración propia)

El análisis comparativo entre la solución propuesta y otras alternativas establece claramente el valor diferenciador que nuestro desarrollo ofrece.

A diferencia de soluciones comerciales como Nextcloud, Sharepoint o Docuware, nuestra aplicación se distingue por su capacidad de adaptación precisa a las necesidades específicas de la empresa. Estas necesidades, que a menudo son únicas y requieren soluciones personalizadas, no siempre son completamente abordadas por las soluciones comerciales estándar.

Nuestra aplicación no solo cumple con los requisitos funcionales actuales de la empresa, sino que también ofrece una flexibilidad inherente que permite su escalabilidad y ampliación en el futuro.

Esto significa que, a medida que la empresa evolucione y surjan nuevas necesidades, nuestra solución estará lista para adaptarse y crecer junto con ella. Además, al estar desarrollada internamente, tenemos el control total sobre el ciclo de vida del software, lo que nos brinda la capacidad de realizar ajustes y mejoras de manera ágil y eficiente, en línea con las demandas cambiantes del entorno empresarial.

## Aplicaciones web

Según Smith, J. (2020) una aplicación web es un software diseñado para ser accesible a través de un navegador web en cualquier dispositivo con conexión a Internet. Estas aplicaciones se ejecutan en servidores remotos y ofrecen funcionalidades y servicios a los usuarios a través de una interfaz web interactiva.

Desde que Tim Berners-Lee, conocido como el inventor de la World Wide Web, estableció los fundamentos de las aplicaciones web con la creación del primer navegador web en 1990 (Berners-Lee, 1999), las aplicaciones web han sufrido una considerable evolución que las han transformado en experiencias interactivas y complejas.

Inicialmente, las aplicaciones web se limitaban a ofrecer información estática y recursos básicos, como páginas HTML simples y formularios de contacto. Sin embargo, con el avance de las tecnologías web y la adopción generalizada de Internet, surgieron nuevas posibilidades para el desarrollo de aplicaciones más dinámicas e interactivas.

En los últimos años, la proliferación de frameworks y bibliotecas de JavaScript, como Angular, React y Vue.js, ha revolucionado aún más el panorama de las aplicaciones web. Estas herramientas permiten a los desarrolladores crear aplicaciones altamente interactivas y eficientes, impulsando el desarrollo de aplicaciones de una sola página (Single Page Applications, SPAs) y aplicaciones progresivas (Progressive Web Apps, PWAs) que ofrecen experiencias similares a las de las aplicaciones nativas de escritorio y móviles (Hoy, 2018).

La creciente demanda de experiencias digitales más ricas junto con la necesidad de una mayor accesibilidad debido a sociedad conectada en la que una aplicación o tarea puede ser realizada desde distintos dispositivos y ubicaciones, eliminando la necesidad de descargar e instalar software adicional, han contribuido a impulsar la popularidad de las aplicaciones web convirtiéndolas en una opción muy atractiva tanto para desarrolladores como para los propios usuarios finales.

El desarrollo de aplicaciones web fue creciendo con el paso de los años debido al crecimiento en paralelo del mundo informático y en especial en lo que respecta al área de programación. Es por esto por lo que es preciso comprender por un lado las metodologías web que se utilizan para su creación y, así mismo, las etapas correspondientes al momento de pensar un proyecto de desarrollo web (Iglesias, 2021; Lion, 2022).

Al crear un software es necesario que se tomen en cuenta múltiples aspectos que van a intervenir, siendo motivos para los cuales se está creando una aplicación web (Chicaiza y Guevara, 2023). Es decir que posee gran importancia el planteamiento de objetivos, el motivo de su creación, para qué va a servir, qué características va a poseer, entre otras. Tal como lo plantea Pressman (2010), se debe buscar conocer en profundidad el problema para el que el desarrollo de una aplicación de software es creado.

### Etapas del desarrollo web

Claro que como todo resultado final conlleva implicado un proceso, dicho proceso a su vez debe contar de etapas, y es así como se plantean las siguientes etapas que conforman el desarrollo de un proyecto web presentado por García de León (2002):

1. **Planificación:**

En esta etapa cobra mayor relevancia el equipo, consistente en el conjunto de técnicos encargados de crear, construir, evaluar y mantener el software. Debe existir una combinación entre experiencia y conocimiento, por esto también es importante que cada equipo cuente con un encargado o líder que pueda guiar y coordinar de la manera más eficiente posible, así como además mantener la comunicación interna.

1. Reunir información, siendo este proceso donde se investiga toda la información teórica y técnica necesaria para la creación del proyecto.
2. Establecer la misión y los objetivos. Estos dos elementos van a ser las referencias por el cual se van a guiar la mayoría de las acciones llevadas a cabo. En relación con los objetivos puede haber modificaciones, pero sin alejarse demasiado.
3. Establecer los posibles usuarios y destinatarios. En este aspecto se debe tomar en cuenta los usuarios reales y los usuarios potenciales, es decir aquellos a los que se puede llegar a alcanzar con la creación del proyecto web. Para ello no sólo basta con tomar información proveniente del usuario sino además puede realizarse por estadísticas de uso, encuestas online, abrir espacios para consultas, entre otras.
4. Diseñar un plan de trabajo. El plan de trabajo es un mapa que va a orientas los pasos a seguir, puede sufrir modificaciones al igual que los objetivos, pero aun así desde el comienzo el plan debe tener solidez contando con todos los elementos que se fueron mencionando anteriormente. En este plan la información se deberá estructurar, seleccionar, elaborar, presentar y evaluar.
5. **Concepción del proyecto:**

Esta etapa debe dar respuesta a la pregunta cómo se llevará a cabo dicho proyecto.

1. Plantear criterios de calidad, en el cual guarda relación con el diseño y el formato de la información presentada. Es decir que se toman en cuenta los objetivos y su grado de cumplimiento, el conocimiento que se posee sobre la audiencia, las características de acceso para los usuarios, el diseño, la tipología para su identidad visual, entre otros.
2. Fijar los contenidos. El establecimiento de los contenidos ayudará a marcar los objetivos y lo que se pretende alcanzar hacia la audiencia como proyecto web. Aquí se debe atender a la escritura y el estilo y/o formato de los textos que deben cumplir con las condiciones de texto o escritura web.
3. Fijar la estructura de los contenidos. El soporte clave para este proceso es la hipertextualidad ya que no alcanza con presentar la información y ya, sino que se debe pensar un orden, organización y fundamentalmente un fácil acceso a los contenidos que dé como resultado una navegación eficiente.
4. Plantear el formato gráfico. Esta dimensión se refiere a la expresión gráfica que facilita la visualización y la fácil navegación por parte del usuario. Aquí se deben fijar la estructura que va a tener la página principal o de acceso que también es llamada home page, si el menú de acceso será con tablas, con barra de navegación, con menú desplegable, entre otros. Se debe seleccionar las imágenes, iconos y logotipos. Se deben establecer la paleta de colores y formato de texto a utilizar.
5. Creación de documentos de trabajo interno. Este aspecto se refiere a la presencia de páginas web que tienen un acceso restringido y las cuales son llamadas “páginas silenciosas”, las cuales sirven como evaluar y perfeccionar los elementos que componen la página web.
6. **Construcción, visualización y testeo:**

La tercera etapa se basa en las acciones de construir, de visualizar y de testear. Antes de que el producto sea presentado a los usuarios, se debe hacer un análisis de, entre otras, el nivel de complejidad que tiene la edición, los browsers, las posibles resoluciones de las pantallas y sus tamaños, el tiempo de carga de la página, etc.

1. Dentro de esta etapa se encuentra el proceso de construir y evaluar, en el que se debe crear un prototipo para evaluarlo y reducir la complejidad de los contenidos en el caso de que sea necesario. Este es el último paso en el que se pueden modificar o fijar estructuras y diseños de contenido. Esta simulación permite acercarse a la experiencia que puede tener el usuario en la navegación, logrando a la vez poder conocer el grado de satisfacción o insatisfacción.
2. La página de acceso principal. Es una función principal porque es la puerta para visualizar toda la estructura del sitio
3. Decisiones sobre la navegación. La selección de una buena tipología de navegación, es decir cómo el usuario va a navegar, puede perjudicar o ayudar en la experiencia del usuario. Por esto se debe elegir opciones de navegación claras, sin perder la barra de menús, contar con un buscador general o interno, entre otras funciones que agilicen la experiencia.
4. La identidad visual. En este aspecto se pone mucho en juego la imagen y suele estar asociado a las estrategias de marketing de la empresa de la web, esto quiere decir que el uso de iconos, imágenes e incluso hasta la ubicación del buscador o barra de menús debe ser homogéneo y representar una identificación y asociación entre lo que se observa y la empresa.
5. **Promoción:**

Esta etapa consta de poder mostrarse y darse de alta en diferentes buscadores internos o generales, se debe contar con una URL (Universal Resource Locator) y un acceso fácil de recordar, incorporarlo en mensajerías por correo electrónico y/o en páginas similares a dicha web, también es posible trabajar con la incorporación de los llamados banners como medio de difusión y que asimismo deben ser diseñados con estrategias de presentación web.

1. **Evaluación:**

Este paso es esencial debido a que gracias a la evaluación que se realiza se puede llegar a identificar las carencias, debilidades y/o fortalezas, conocer de qué forma ha impactado en los usuarios la aplicación web, optimizar determinadas áreas, entre otras. Dicha evaluación debe constar de hacerse las preguntas iniciales de qué, para qué, por qué y cómo, que están orientadas a la misión y objetivos del desarrollo de un proyecto web.

1. **Evolución:**

En este punto se pone en juego lo relacionado a las actualizaciones y el equipo destinado al mantenimiento del sitio web, por ello se debe contar con un plan que conste del seguimiento a las evaluaciones y al logro de los objetivos porque de estos elementos es que se va a servir para perfeccionarse. Las actualizaciones juegan un rol fundamental le brindará credibilidad y garantía de poder seguir vigentes.

### Importancia para la gestión documental

Tal como se fue mencionando a lo largo de este trabajo, la génesis y la incorporación de internet y el mundo tecnológico a la vida del hombre conllevó grandes avances siempre que se pudo usarla de manera provechosa (Barrio Andrés, 2021). De este modo, es que también las grandes organizaciones comenzaron a incorporar tecnologías emergentes para sacar el máximo provecho en sus sistemas internos y externos de comunicación (Arias et al, 2021).

Una de las grandes características que trae aparejado el uso de internet en organizaciones y empresas que intentan digitalizar toda la información y la documentación, es el enfoque ambiental disminuyendo las emisiones de gases a la atmosfera debido a que se reduce el consumo de papel optando por lo digital (Corrochano Garrido, 2022).

El proyecto de desarrollo web para la digitalización de la información es el comienzo de la implementación de la tecnología vigente con el fin no sólo de optimizar la calidad de los servicios afines, sino que también permite permanecer actualizados dentro de un mercado que cada vez tiene mayor impacto y alcance (Gómez Flórez, 2023).

La realización de este proyecto demuestra la posibilidad de brindar soluciones agilizadas e interesantes que el usuario, sea interno o externo, pueda tener acceso fácil y final a cada uno de los documentos o de la información necesaria, por esto es por lo que la aspiración de toda digitalización es la optimización de un producto nuevo o ya vigente (Cacheda Couto, 2022). El desarrollo web en la gestión documental da valor y viabilidad actual a la empresa que lo realice (Atarama y Villanueva, 2022).

Dentro de las posibilidades que genera la digitalización o gestión documental se puede mencionar (Rincón Silva, 2020):

- La integración de sistemas internos de consulta entre la empresa y los usuarios, teniendo un acceso a la información necesaria en el tiempo y momento que se requiera.

- La disponibilidad de la información y su fácil acceso

- La posibilidad de usar mecanismos de seguridad biométricos que puedan identificar a los usuarios o los requirentes de los documentos con el fin de poder establecer una mayor confidencialidad si así se lo desea. De esta forma se podrían evitar fallas o errores al identificar al usuario que accede a la información.

- La posibilidad de intercambiar y gestionar documentación a largas distancias físicas, es decir de forma electrónica entre los diferentes dispositivos.

- Obtener información sobre el flujo de la documentación con el propósito de tomar decisiones.

### Metodologías en el desarrollo de aplicaciones web

Debido al crecimiento de la tecnología en la vida cotidiana, así como presenta beneficios también se presentan problemas relacionados a su desarrollo y es por esto por lo que surgieron metodologías que puedan ayudar a guiar el proyecto de desarrollo de un software para disminuir los errores lo mayor posible. Es por esto por lo que se enuncian a continuación las metodologías presentadas por diversos autores, teniendo cada una de ellas sus propias características (Ríos et al, 2018).

No es fácil el desarrollo de un proyecto web, es por esto por lo que ineludiblemente surgió su alternativa: la metodología (Delgado, 2008). La metodología de desarrollo de un software consta de un proceso en el que intervienen procedimientos, técnicas y toda aquella teoría que dé lugar a los desarrolladores poder llevar adelante el proyecto con el fin de crear una aplicación que refleje calidad, cumpla expectativas y brinde soluciones. Tal y como indica Pressman (2010), la metodología puede definirse como el marco de trabajo que se debe usar para dar una estructura al sitio web, así como una guía para planificar y controlar todo el proceso de desarrollo que implican los sistemas.

Estas metodologías pueden clasificarse históricamente en tres: tradicionales, ágiles e híbridas.

Las metodologías tradicionales son aquellas que se asocian al desarrollo de software de tipo manual con el fin de alcanzar la optimización y el cumplimiento de objetivos propuestos. En este sentido las metodologías tradicionales de acuerdo con Leiva y Villalobos (2015), está orientada a controlar el proceso, las actividades intervinientes y los artefactos que se van a desarrollar, así como las herramientas y notas involucradas. Lo que se busca es dividir el proceso del desarrollo del sistema para redistribuir las funciones a los integrantes del equipo de trabajo para garantizar una comprensión, mantenimiento y continuidad de las aplicaciones o sitios web. Las metodologías tradicionales dieron su aparición hacia la década de los 60.

Las metodologías ágiles, según Navarro, Fernández y Morales (2013), tienen la característica principal de ser flexibles, por lo que son fáciles de modificarse en el caso de que se lo requiera para cumplir los objetivos establecidos o los nuevos. Lo que se realiza a partir de las metodologías ágiles, es dividir y subdividir en pequeñas porciones el proceso para desarrollar de manera autónoma cada una de ellas en un tiempo aproximado de dos a seis semanas, por lo que durante el proceso cada fracción se desarrollará y evaluará por el mismo equipo. Otra característica de las metodologías agiles es que se pueden adaptar de acuerdo con los cambios que se soliciten por parte de los usuarios haciéndolos parte del equipo, de esta manera se garantiza la entrega de un producto óptimo.

De entre las metodologías clásicas y las metodologías agiles, surgen las llamadas metodologías híbridas, por lo tanto, es una combinación de las dos primeras, solo que con la nueva se busca que se alcance firmeza, pero a la vez flexibilidad ante cualquier clase de proyectos de desarrollo de sitios web o software. Según Orantes Jiménez (2017), las metodologías híbridas intentan recabar las ventajas de las dos metodologías para alcanzar una combinación práctica y eficiente, así mismo también puede unir dichas ventajas con las ventajas de la nueva metodología para lograr aun mayor optimización.

### Gestión documental en las empresas de telecomunicaciones

Se debe recordar que la gestión documental se refiere a un proceso basado en la gestión de documentos o desde su término anglosajón records management, donde se contiene toda la información y datos sobre la productividad y el desarrollo de las instituciones u organizaciones, pudiendo garantizar desde esta forma de sistematización un control, orden y mejoramiento en el desempeño organizacional (Bayona, y Meneses, 2020).

Centrados en lo documental, se puede establecer que uno de los puntos clave de todo documento es su contenido, así como su fácil acceso al mismo, no teniendo la misma suerte su presentación. Es decir, tiene más valor lo que se encuentra dentro del documento que la forma o el formato en el que se puede acceder. Desde la implementación de las tecnologías a la vida cotidiana y empresarial, se han desarrollado diferentes formar de digitalizar los documentos, tal como se menciona anteriormente, siendo el contenido lo que genera valor y poder para llevar adelante la toma de decisiones posteriores, su consecuente almacenamiento y continua conservación.

Cuando se opta por la gestión documental en formato de digitalización, se recomienda que las técnicas implicadas para el mismo logren capturar y gestionar la información de las actividades y las transacciones en formato de documentos, tal como lo permite la ISO(2001), del mismo modo que evaluar las políticas intervinientes para la producción del documento como elemento de información, esto en conjunto con la aplicación de sistemas tecnológicos que tengan como función recuperar la información e incidir en nuevas formas de trabajo que alcancen la conservación de los documentos como fin último (Sánchez, 2014).

# Objetivos y metodología de trabajo

## Objetivo general

El objetivo principal de este Trabajo de Fin de Grado es implementar una aplicación web utilizando el stack MERN destinada a optimizar y automatizar los procesos de almacenamiento, recuperación y gestión de documentos en la empresa, proporcionando una solución eficiente y escalable que mejore la organización y la productividad en el manejo de la documentación.

## Objetivos específicos

Los siguientes objetivos específicos han sido seleccionados por su capacidad para abordar de manera integral todas las etapas clave del desarrollo de la aplicación web de gestión documental, garantizando su relevancia, calidad y utilidad para la empresa:

* Analizar requisitos e identificar las demandas específicas de cada área de la empresa en cuanto a almacenamiento, acceso y gestión de documentos.
* Diseñar la arquitectura, interfaces y base de datos necesarias
* Desarrollar la aplicación según el diseño previamente establecido
* Realizar pruebas exhaustivas que garanticen el correcto funcionamiento de la aplicación en diferentes escenarios y condiciones.
* Obtener conclusiones y proponer áreas de mejora o recomendaciones futuras

## Metodología de trabajo

### Modelo en Cascada vs SCRUM

Según Schwaber y Sutherland (2017), Scrum es "un marco de trabajo dentro del cual las personas pueden abordar problemas complejos, y adaptarse rápidamente y de manera creativa a los cambios". Este enfoque se basa en principios de transparencia, inspección y adaptación, y se organiza en torno a roles, eventos y artefactos que guían el proceso de desarrollo (Schwaber & Sutherland, 2017).

Tanto la metodología en cascada como Scrum tienen sus propias ventajas y desventajas por lo que la elección entre ambas depende de las necesidades y características específicas del proyecto.

A continuación, se detalla una pequeña comparativa efectuada entre ambos modelos para facilitar la decisión final:

**Estructura y Secuencialidad:**

La metodología en cascada se caracteriza por seguir un enfoque secuencial y lineal, donde las fases del proyecto se realizan de manera ordenada y progresiva, sin posibilidad de retorno a etapas anteriores una vez completadas.

En contraste, Scrum es un enfoque ágil que se basa en ciclos iterativos y cortos, donde el desarrollo del proyecto se divide en iteraciones llamadas "sprints", permitiendo flexibilidad y adaptación a medida que avanza el proyecto.

**Flexibilidad y Adaptabilidad:**

Scrum ofrece una mayor flexibilidad y adaptabilidad a los cambios en los requisitos del proyecto, ya que permite realizar ajustes y cambios en cada sprint en función de la retroalimentación del cliente.

La metodología en cascada, al ser más rígida y estructurada, puede resultar menos flexible para adaptarse a cambios durante el desarrollo del proyecto, lo que puede generar dificultades en caso de requerimientos cambiantes.

**Feedback del Cliente:**

Scrum facilita el feedback continuo del cliente a lo largo del proyecto, ya que se realizan entregas incrementales de funcionalidades en cada sprint, permitiendo al cliente validar y ajustar el producto de manera temprana.

En la metodología en cascada, el feedback del cliente se recibe en etapas específicas del proyecto, como al finalizar la fase de implementación o durante las pruebas finales, lo que puede limitar la capacidad de realizar cambios importantes una vez avanzado el proyecto.

### Elección de la Metodología en Cascada: Justificación y Razones

Existen una serie de características del proyecto que convierten a la metodología en cascada en un modelo adecuado para el desarrollo del presente trabajo:

**Requisitos Claros y Estables:**

El proyecto cuenta con requisitos definidos y estables desde el inicio dado que la naturaleza de la gestión documental implica procesos bien establecidos y requisitos específicos en cuanto a la recopilación, almacenamiento y gestión de documentos.

Esto permite que se puedan definir claramente todas las etapas del proyecto de manera anticipada.

**Estructura y Planificación:**

La metodología en cascada proporciona una estructura clara y una planificación detallada del proyecto.

Cada etapa del desarrollo, desde la recolección de requisitos hasta la implementación y el despliegue, está definida de antemano, lo que facilita la organización y ejecución del proyecto. Esta estructura permite una mayor predictibilidad en cuanto a los plazos y los recursos necesarios para cada fase del desarrollo.

**Uso de MERN:**

El uso del stack MERN para el desarrollo de la aplicación web también se beneficia del enfoque en cascada.

Si bien MERN es compatible con enfoques ágiles como Scrum, la metodología en cascada se adapta igualmente bien a este tipo de desarrollo.

La estructura clara y secuencial del modelo en cascada se integra sin problemas con las tecnologías y herramientas del stack elegido, lo que facilita la implementación y el seguimiento del proyecto.

**Foco en la Calidad y las Pruebas:**

La metodología en cascada permite un enfoque sistemático en la calidad del software y las pruebas.

Cada etapa del proyecto está diseñada para garantizar que el producto final cumpla con los estándares de calidad establecidos. Esto es esencial en un proyecto de gestión documental, donde la precisión y la fiabilidad de la aplicación son cruciales para el éxito del negocio.

En definitiva, la metodología en cascada se perfila como la mejor opción para el desarrollo del proyecto que se aborda en este trabajo.

Su estructura clara, planificación meticulosa y enfoque sistemático en la calidad la convierten en la herramienta ideal para alcanzar los objetivos previstos.

### Metodología en Cascada

En el contexto de la gestión de proyectos y el desarrollo de software, el modelo en cascada (también conocido como modelo de ciclo de vida en cascada o Waterfall model) es un enfoque metodológico tradicional que se caracteriza por su estructura rígida y secuencial puesto que organiza el proceso en una serie de fases ordenadas que el equipo debe completar antes de avanzar a la siguiente.

Según Winston W. Royce (1970), la primera descripción formal de esta metodología se realizó en el contexto de la gestión de desarrollo de software.

Aunque se originó en la fabricación y construcción, hoy en día se aplica principalmente en proyectos relacionados con el desarrollo de sistemas y tecnologías de la información (TI).

El modelo de cascada divide el proyecto en distintas fases secuenciales, donde el equipo avanza a la siguiente fase solo después de completar la anterior tal y como se observa en la figura 6 (Delgado Olivera & Díaz Alonso, 2021).

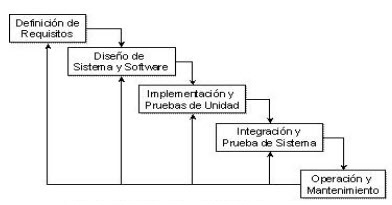


Figura 19: Modelo de desarrollo en cascada (Delgado Olivera & Díaz Alonso, 2021)

#### Fases de la metodología en cascada

A continuación, se realiza una descripción de las distintas fases que componen el modelo en cascada:

* **Definición de Requisitos:**

La fase de definición de requisitos es la etapa inicial y fundamental en la metodología en cascada.

Durante esta fase, el equipo de desarrollo trabaja en estrecha colaboración con los interesados, ya sean clientes, usuarios finales u otros stakeholders relevantes, para identificar y comprender completamente los requisitos del sistema.

El objetivo principal es capturar todas las necesidades y expectativas del cliente en relación con el software que se va a desarrollar.

Esta fase implica la realización de actividades clave, tales como entrevistas con los clientes y usuarios finales, análisis de documentos existentes, estudios de mercado y análisis de la competencia, entre otros con el objetivo recopilar información detallada sobre las funciones, características, restricciones y objetivos del sistema, así como los criterios de éxito y las expectativas de los stakeholders.

Según Pressman (2014), durante la fase de recolección de requisitos, "el objetivo principal es descubrir los requisitos del sistema y establecer una base sólida para el diseño del sistema. Esto implica la comprensión de las necesidades del cliente y la definición de lo que el sistema debería hacer" (p. 140).

Es importante que esta fase se realice de manera exhaustiva y meticulosa, ya que unos requisitos mal definidos o interpretados pueden dar lugar a errores costosos y retrasos en etapas posteriores del proyecto.

Por lo tanto, es necesario emplear diversas técnicas y herramientas, como entrevistas estructuradas, cuestionarios, prototipado rápido y casos de uso, para garantizar una comprensión clara y precisa de los requisitos del sistema.

* **Análisis:**

La fase de análisis es una etapa crucial en la metodología en cascada puesto que es donde se lleva a cabo un estudio exhaustivo de los requisitos del sistema y se define cómo se traducirán en funcionalidades concretas del software.

Durante esta fase, el equipo de desarrollo trabaja en estrecha colaboración con los stakeholders, incluidos los clientes y usuarios finales, para comprender completamente sus necesidades y expectativas.

Según Sommerville (2011), "El análisis es una fase crucial en el desarrollo de software donde se establecen y documentan los requisitos del sistema" (p. 122). Durante esta etapa, se recopilan los requisitos funcionales y no funcionales, se identifican las restricciones y se definen los objetivos del proyecto.

El objetivo principal del análisis es identificar y documentar todos los requisitos del sistema de manera clara y precisa. Esto incluye requisitos funcionales, que describen las funciones y características que el sistema debe realizar, así como requisitos no funcionales, que especifican criterios de calidad, restricciones técnicas y otros aspectos no directamente relacionados con la funcionalidad del sistema.

Una vez recopilados los requisitos, se utilizan diversas técnicas y herramientas, como diagramas de casos de uso, modelos de datos y prototipos de software, para analizar y validar la viabilidad y la coherencia de los requisitos identificados.

* **Diseño:**

Durante esta fase de diseño del modelo en cascada, se transforman los requisitos del sistema recopilados en la fase previa en una arquitectura técnica detallada que servirá como base para la implementación del software.

Según Pfleeger y Atlee (2010), "En la etapa de diseño, la descripción del software se detalla lo suficiente para que se pueda programar. Esta descripción incluye descripciones de las estructuras de datos, algoritmos, interfaces y componentes de sistema" (p. 118).

Es en esta fase donde se definen los módulos del sistema, las relaciones entre ellos y las interfaces de usuario que permitirán a los usuarios interactuar con el software.

Al final de la fase de diseño, se produce una documentación detallada que describe la arquitectura del sistema, los componentes individuales y sus interfaces.

Esta documentación proporciona una guía para los programadores durante la fase de implementación y ayuda a garantizar la coherencia y consistencia del software desarrollado.

* **Implementación:**

Tal y como menciona Pressman (2014), "La implementación es la etapa en la que el diseño se convierte en código ejecutable" (p. 157).

Durante la fase de implementación, se lleva a cabo la codificación del software basándose en el diseño detallado desarrollado en etapas anteriores. Esta fase implica la traducción de los requisitos y especificaciones del sistema en código ejecutable.

Es en esta etapa donde los programadores desarrollan el software utilizando los lenguajes y herramientas especificadas en la fase de diseño.

La implementación en la metodología en cascada sigue un enfoque estructurado y secuencial. Los módulos o componentes individuales se desarrollan y prueban por separado antes de ser integrados en el sistema completo.

Esto permite una mejor gestión del proceso de desarrollo y facilita la detección temprana de errores.

Al finalizar la fase, se produce una versión inicial del software que está lista para ser probada y validada por los usuarios finales. Esta versión se somete a una revisión exhaustiva para garantizar que cumpla con los requisitos y expectativas del cliente.

* **Pruebas:**

La fase de pruebas constituye un componente crítico en el proceso de desarrollo de software, destinado a identificar y subsanar posibles errores para garantizar la calidad y el correcto funcionamiento del producto final. En la metodología en cascada, esta fase se estructura en distintos apartados, cada uno con su enfoque particular:

**Pruebas unitarias:** Llevadas a cabo tras la fase de implementación, estas pruebas evalúan el desempeño de cada unidad de código de manera individual, asegurando su funcionalidad aislada (Pressman, 2020).

**Pruebas de integración:** Realizadas después de las pruebas unitarias, se enfocan en verificar la adecuada interacción entre las diversas unidades de código, garantizando su correcta integración como un sistema coherente (Pressman, 2020).

**Pruebas de sistema:** Conducidas posteriormente a las pruebas de integración, estas pruebas evalúan el desempeño global del sistema, corroborando su funcionamiento integral y su conformidad con los requisitos establecidos (Pressman, 2020).

**Pruebas de aceptación:** Procedimientos llevados a cabo por el cliente tras las pruebas de sistema, destinados a verificar que el software satisface sus necesidades y requisitos específicos (Sommerville, 2016).

Además, se emplean diversos tipos de pruebas para abordar diferentes aspectos del software:

**Pruebas funcionales:** Centradas en validar que el software cumple con los requisitos funcionales definidos para su funcionamiento (Black, 2009).

**Pruebas no funcionales:** Dirigidas a evaluar aspectos no funcionales del software, como rendimiento, seguridad y usabilidad (Black, 2009).

Asimismo, se emplean distintas técnicas de prueba para obtener una cobertura exhaustiva:

**Pruebas de caja negra:** Basadas en el conocimiento de la interfaz del software, sin acceder a su código interno, para diseñar y ejecutar las pruebas (Black, 2009).

**Pruebas de caja blanca:** Fundamentadas en el entendimiento del código interno del software, permiten diseñar pruebas específicas que exploran su lógica interna y estructura (Black, 2009).

Estas diversas estrategias y técnicas de prueba se combinan para garantizar una evaluación integral y exhaustiva del software desarrollado en el marco de la metodología en cascada.

* **Mantenimiento:**

La fase de mantenimiento es la última etapa del ciclo de vida del desarrollo de software en la metodología en cascada. Su objetivo principal es asegurar que el software siga funcionando correctamente y cumpliendo con las necesidades del cliente después de su lanzamiento (Pressman, 2020).

Las actividades que se llevan a cabo en la fase de mantenimiento son:

* **Corrección de errores:** Se corrigen los errores que se encuentran en el software después de su lanzamiento.
* **Mejora del software:** Se realizan mejoras en el software para agregar nuevas funcionalidades o mejorar las existentes (Sommerville, 2016).
* **Actualización del software:** Se actualiza el software para mantenerlo compatible con nuevas tecnologías.

Para ello se pueden efectuar varios tipos de mantenimiento y hacer uso de varias técnicas distintas:

* **Mantenimiento correctivo:** Se centra en la corrección de errores en el software.
* **Mantenimiento adaptativo:** Se centra en la adaptación del software a nuevos entornos o necesidades.
* **Mantenimiento perfectivo:** Se centra en la mejora del rendimiento, la seguridad y la usabilidad del software (Lientz & Swanson, 1980).

Técnicas de mantenimiento:

* **Pruebas de regresión:** Se realizan pruebas para asegurar que las correcciones de errores no han introducido nuevos errores en el software.
* **Reingeniería:** Se modifica la estructura interna del software sin cambiar su funcionalidad.
* **Refactorización:** Se mejora la estructura del código del software para hacerlo más legible y mantenible.

# Diseño y evaluación de la propuesta

En este capítulo, se presenta una exhaustiva descripción de la arquitectura de la aplicación propuesta, así como de sus características y funcionalidades principales.

Se destaca especialmente cómo se han abordado los requisitos específicos de gestión documental de la empresa, lo que proporciona una visión detallada de cómo la aplicación responde a las necesidades y desafíos particulares del entorno empresarial.

Asimismo, se ofrecen detalles sobre la implementación práctica de la aplicación, lo que incluye un análisis de las tecnologías utilizadas y los obstáculos enfrentados durante el proceso de desarrollo que permite comprender mejor el contexto técnico en el que se ha desarrollado la solución propuesta y los aspectos prácticos involucrados en su puesta en marcha.

Finalmente, se examinan los resultados obtenidos de las pruebas realizadas, abordando aspectos de funcionalidad, usabilidad y rendimiento.

Esta evaluación exhaustiva permite evaluar de manera objetiva el cumplimiento de los objetivos establecidos para la aplicación y su capacidad para satisfacer las necesidades identificadas en la empresa.

### Alcance

La finalidad del presente proyecto es la realización del análisis, diseño, implementación y validación de una aplicación web que permita a los usuarios la realización de las tareas de gestión documental, creando para ello la lógica de negocio, base de datos no relacional y diseño de interfaces de usuario para interactuar con la misma.

El usuario solicitante podrá crear un formulario de solicitud de documentación desde la aplicación web, especificando los requisitos de los documentos, sus características criticas tales como la fecha de caducidad y el destinatario de dicha solicitud.

Por otra parte, los usuarios aportantes podrán acceder a los distintos requerimientos que les han solicitado desde la empresa, remitir los documentos solicitados y comprobar el estado en el que se encuentran (pendiente, validado o invalidado).

Desde la misma aplicación web, personal de la empresa podrá validar o invalidar aquellos documentos que han sido aportados por si fuese necesaria su reposición por el aportante.

Adicionalmente, se integrará una nueva funcionalidad dirigida a los clientes internos y externos, quienes tendrán acceso a una sección específica dentro de la aplicación web.

En esta sección, podrán visualizar un listado de las facturas y documentos que la empresa ha generado para ellos, así como descargarlos en formato digital para su conveniencia.

Esta funcionalidad requerirá la implementación de medidas de seguridad adicionales para garantizar que solo los clientes autorizados puedan acceder a sus documentos.

### Análisis de requisitos

La fase de análisis de requisitos es fundamental en el proceso de desarrollo de software puesto que es en este contexto en donde se procede a la identificación y documentación de los requisitos del sistema, lo que implica una exhaustiva investigación para comprender las necesidades y expectativas de los usuarios involucrados en el proyecto.

Esta etapa reviste una importancia crítica para el éxito del proyecto de desarrollo de software, ya que una identificación incorrecta de los requisitos puede provocar retrasos, incumplimiento de las expectativas de los usuarios, aumento de los costos del desarrollo o incluso insatisfacción del cliente.

Por ende, para mitigar cualquier posible impacto negativo, es imperativo colaborar estrechamente con los usuarios y atender de manera rigurosa sus requerimientos, siempre en consonancia con las políticas departamentales establecidas.

Para capturar de manera efectiva las interacciones de los usuarios con el sistema, se emplean los casos de uso, definidos como "una pieza de funcionalidad en el sistema que proporciona un resultado de valor al usuario. Los casos de uso capturan requisitos funcionales" (Jacobsen et al., 1999, p. 97).

En este proyecto, se ha optado por representar los casos de uso utilizando el lenguaje de modelado UML, lo cual proporciona una visión clara y comprensible de los requisitos funcionales del sistema.

#### Definición de actores

Según Jacobsen et al. (1999), en el contexto de sistemas de software, un actor de un caso de uso se refiere a cualquier persona, sistema externo o entidad que interactúa con el sistema en cuestión.

Cada actor representa un rol o función específica y desencadena eventos dentro del sistema. Identificar y definir estos actores es crucial para comprender las necesidades y expectativas de los usuarios y garantizar el diseño efectivo del sistema.

A continuación, se presenta una tabla detallada que identifica los actores relevantes para el proyecto, así como sus respectivas necesidades y roles en los casos de uso del sistema.

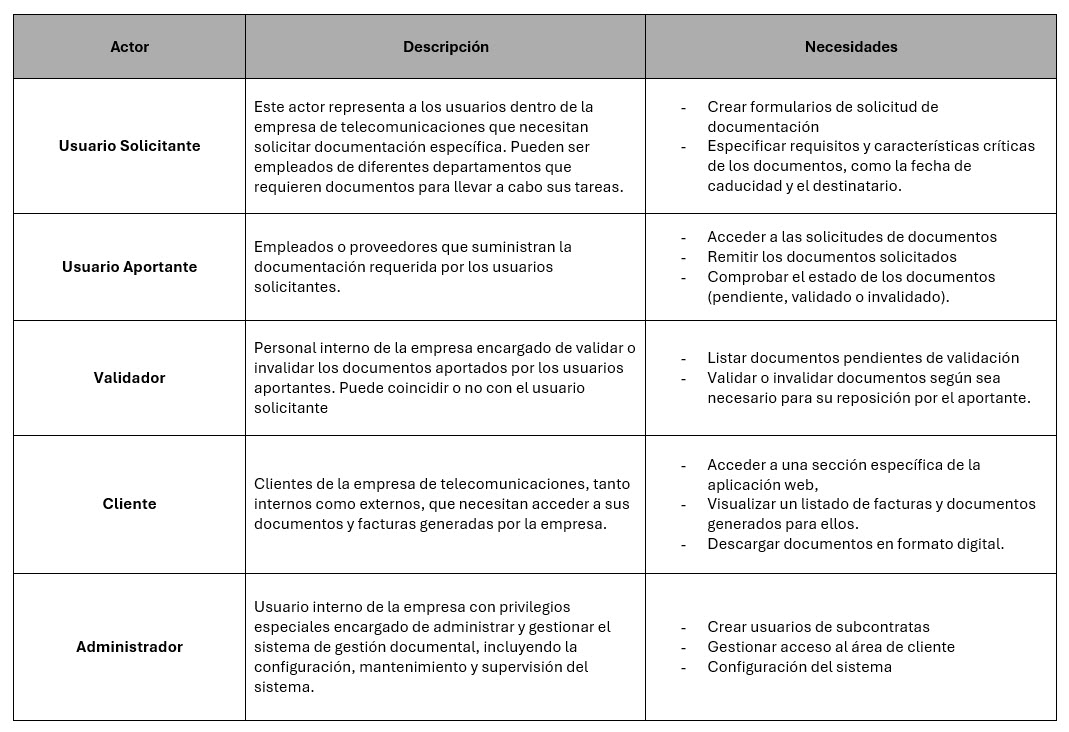


Tabla 2: Actores del sistema (Fuente: elaboración propia)

#### Casos de uso

En esta sección se presentan los casos de uso identificados para la aplicación. Estos casos de uso representan las diferentes interacciones entre los actores del sistema y la funcionalidad que proporciona la aplicación.

Los casos de uso de la tabla representan las principales funcionalidades de la aplicación y servirán como guía durante el proceso de diseño e implementación.

A continuación, se muestra una tabla en la que se detallan los casos de uso identificados mediante un ID, junto con su título y descripción correspondientes:

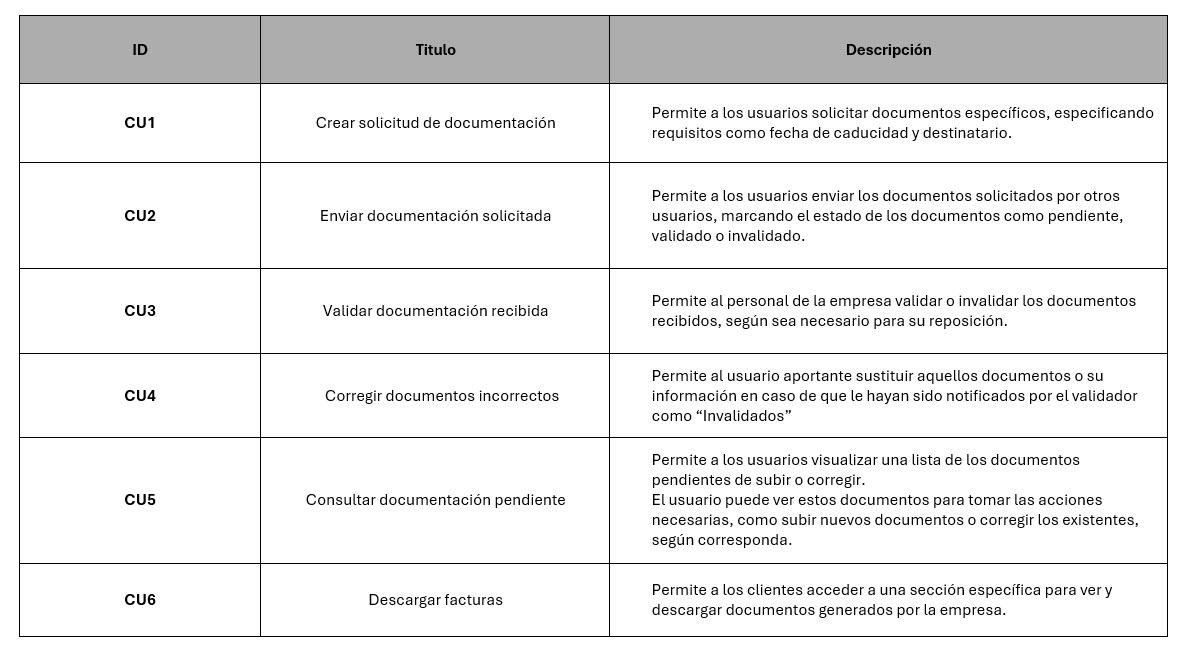


Tabla 3: Casos de uso (Fuente: Elaboración propia)

Estos casos de uso desempeñan un papel crucial en la comprensión de la funcionalidad de la aplicación y en la orientación del desarrollo de esta.

Al identificar y describir detalladamente las diferentes interacciones entre los usuarios y el sistema, permiten visualizar de manera concreta cómo se espera que la aplicación se utilice en situaciones específicas.

Además, proporcionan una guía clara para su desarrollo y posterior desglose en requisitos funcionales.

Al desglosar los casos de uso en requisitos funcionales, se profundiza en el detalle de cada funcionalidad, identificando los criterios específicos que deben cumplirse para satisfacer las necesidades de los usuarios y alcanzar los objetivos del proyecto. Este enfoque detallado permite una planificación más precisa de las actividades de desarrollo, así como una evaluación más efectiva del progreso y la calidad del trabajo realizado.

#### Requisitos funcionales

A continuación, y una vez identificados en el apartado anterior los principales casos de uso del sistema, se realizará un desglose de estos, junto con una representación mediante diagramas UML, que facilitarán su comprensión y la obtención de los requisitos funcionales correspondientes a cada uno de ellos:

|  |
| --- |
| **CU1 – Crear solicitud de documentación** |

**Actores principales:** Solicitante

**Descripción:** Este caso de uso describe el proceso mediante el cual un usuario solicitante crea una solicitud de documentación en la aplicación.

**Flujo básico:**

1. El solicitante inicia sesión en la aplicación.
2. El solicitante accede a la funcionalidad de crear una nueva solicitud.
3. El sistema presenta un formulario de solicitud.
4. El solicitante completa los campos obligatorios del formulario, incluyendo la especificación de los documentos requeridos y una breve descripción de sus características.
5. El solicitante tiene la opción de completar campos adicionales, como la fecha de generación, fecha de caducidad y periodicidad de los documentos, si es necesario.
6. El solicitante confirma y envía la solicitud.

**Precondiciones:**

* El usuario solicitante ha iniciado sesión en el sistema con sus credenciales válidas.
* El usuario solicitante tiene permisos para crear nuevas solicitudes de documentación.

**Postcondición:**

El sistema registra la nueva solicitud en la base de datos y notifica al aportante sobre la solicitud creada.

**Requisitos funcionales derivados:**

* RF1: Formulario de creación de solicitud.
* RF2: Especificación de documentos.
* RF3: Descripción de documentos.
* RF4: Campos opcionales para información adicional.
* RF5: Confirmación de solicitud.

**Diagrama de casos de uso:**

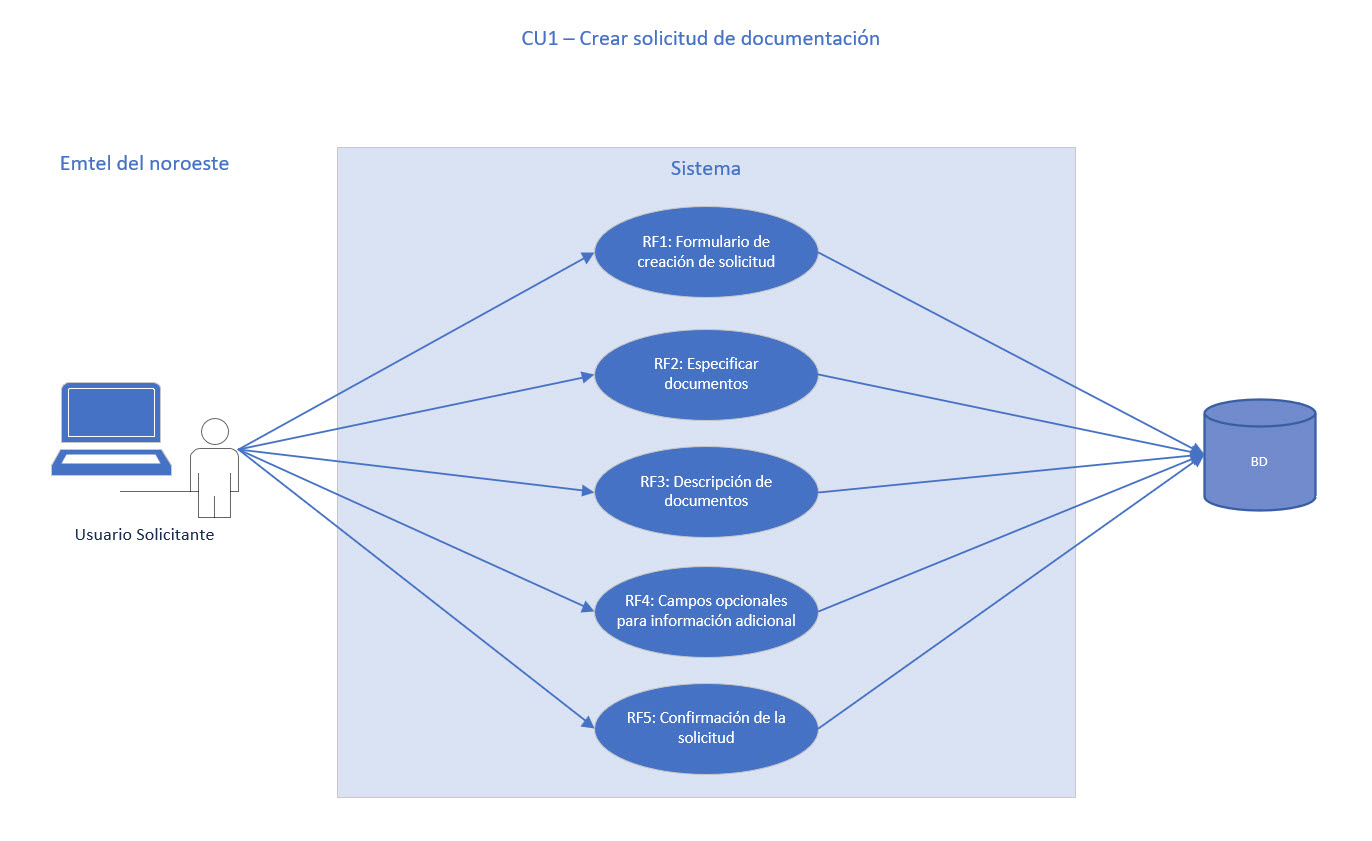
****

Figura 20: CU1 - Crear solicitud de documentación (fuente: elaboración propia)

|  |
| --- |
| **CU2 – Enviar documentación solicitada** |

**Actores principales:** Aportante

**Descripción:** Este caso de uso describe el proceso mediante el cual un usuario aportante accede a las solicitudes de documentación asignadas y envía los archivos correspondientes según lo solicitado.

**Flujo básico:**

1. El aportante inicia sesión en la aplicación.
2. El aportante accede a la lista de solicitudes de documentación asignadas.
3. El sistema muestra las solicitudes pendientes de documentación.
4. El aportante selecciona una solicitud específica para la que desea enviar documentación.
5. El sistema presenta un formulario para subir el archivo solicitado.
6. El aportante selecciona el archivo desde su dispositivo y lo sube al sistema.
7. El aportante tiene la opción de añadir una breve descripción de texto, si es requerido.
8. El aportante puede completar campos adicionales, como las fechas correspondientes en los documentos solicitados, si es necesario.
9. El aportante confirma y envía la documentación.

**Precondiciones:**

* El usuario aportante ha iniciado sesión en el sistema con sus credenciales válidas.
* Existe al menos una solicitud de documentación pendiente asignada al usuario aportante.

**Postcondición:**

El sistema registra la documentación enviada por el aportante en la solicitud correspondiente.

**Requisitos funcionales derivados:**

* RF6: Acceso a la lista de solicitudes asignadas.
* RF7: Formulario para subir archivos.
* RF8: Descripción opcional del archivo.
* RF9: Campos adicionales para fechas en los documentos.
* RF10: Confirmación de envío de solicitud.

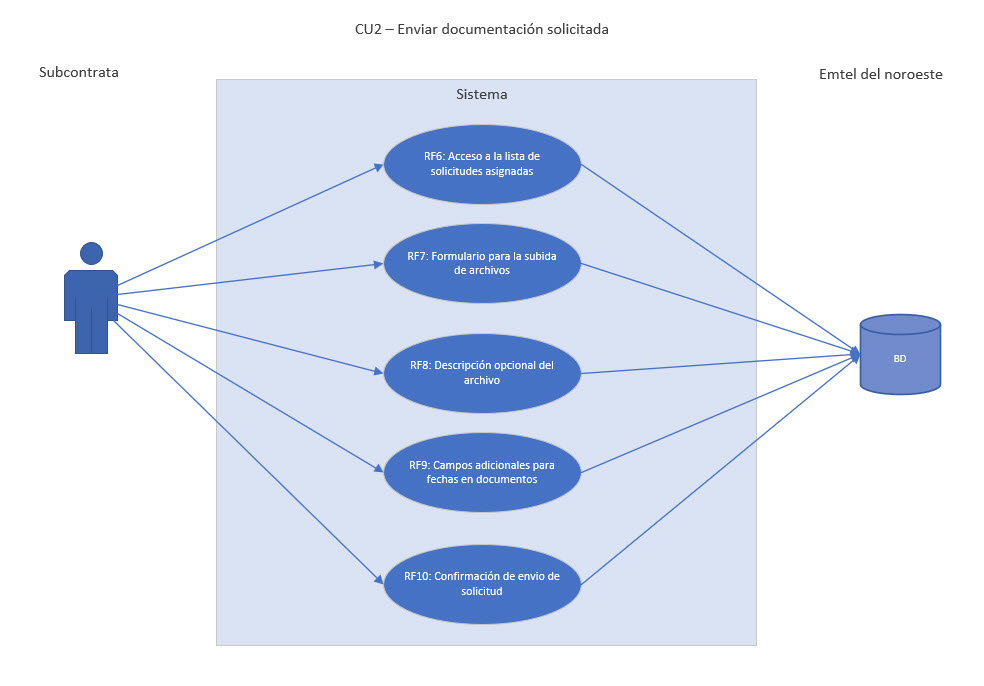
**Diagrama de casos de uso:** 

Figura 21: CU2 - Enviar documentación solicitada (fuente: elaboración propia)

|  |
| --- |
| **CU3 – Validar documentación recibida** |

**Actores principales:** Validador

**Descripción:** Este caso de uso describe el proceso mediante el cual un usuario validador lista y valida la documentación recibida por parte de los aportantes.

El validador puede marcar los documentos como validados o invalidados, proporcionando una breve descripción en caso de invalidación.

Además, el sistema destaca los documentos cuya fecha de caducidad es anterior a la fecha actual, proponiéndolos para su invalidación.

**Flujo básico:**

1. El validador inicia sesión en la aplicación.
2. El validador accede a la lista de documentos pendientes de validación.
3. El sistema muestra los documentos pendientes, incluyendo información adicional proporcionada por el aportante.
4. El validador selecciona un documento específico para validar.
5. El sistema muestra la documentación asociada al documento seleccionado.
6. El validador revisa la documentación y valida el documento.
7. El validador confirma la validación del documento.

**Flujo alternativo:**

1. El validador inicia sesión en la aplicación.
2. El validador accede a la lista de documentos pendientes de validación.
3. El sistema muestra los documentos pendientes, incluyendo información adicional proporcionada por el aportante.
4. El validador selecciona un documento específico para validar.
5. El sistema muestra la documentación asociada al documento seleccionado.
6. El validador revisa la documentación e invalida el documento.
7. El validador proporciona una breve descripción de la causa de invalidación.
8. Si la fecha de caducidad del documento es anterior a la fecha actual, el sistema lo destaca y propone para su invalidación.
9. El validador confirma la invalidación del documento.

**Precondiciones:**

* El usuario validador ha iniciado sesión en el sistema con sus credenciales válidas.
* Existen documentos pendientes de validación asignados al usuario validador.

**Postcondición:**

El sistema registra la validación o invalidación del documento por parte del validador y notifica a los aportantes de los documentos invalidados para que procedan a su corrección.

**Requisitos funcionales derivados:**

* RF11: Acceso a la lista de documentos pendientes de validación.
* RF12: Visualización del documento.
* RF13: Visualización de la información adicional del documento
* RF14: Validar un documento
* RF15: Invalidar un documento
* RF16: Proporcionar una breve descripción en caso de invalidación.
* RF17: Destacar documentos cuya fecha de caducidad sea anterior a la fecha actual.
* RF18: Confirmación de validación o invalidación del documento.

**Diagrama de casos de uso:**

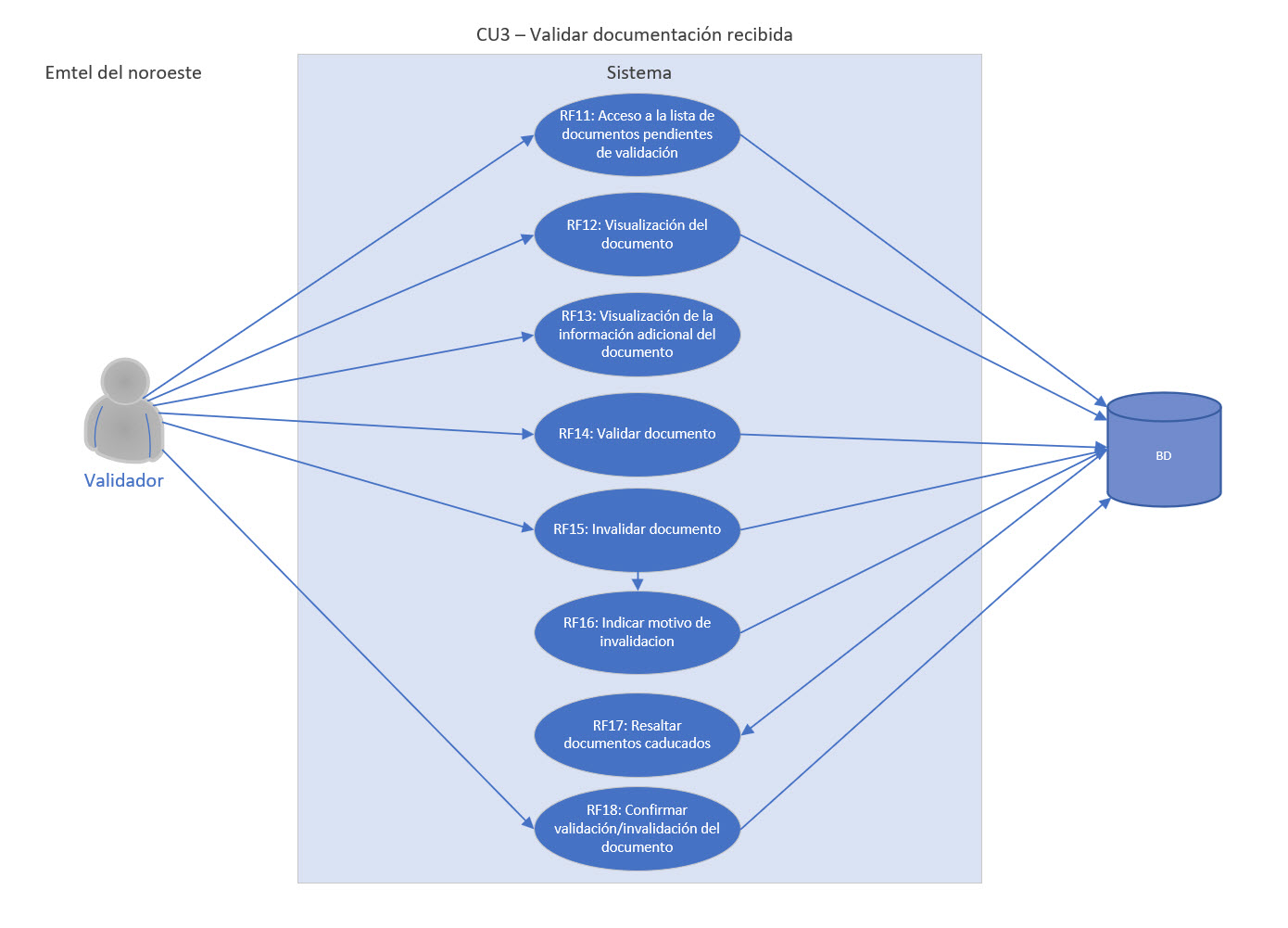


Figura 22: CU3 - Validar documentación recibida (fuente: elaboración propia)

|  |
| --- |
| **CU4 – Corregir documento incorrecto** |

**Actores principales:** Aportante

**Descripción:** Este caso de uso describe el proceso mediante el cual un usuario aportante corrige un documento que ha sido invalidado por un usuario validador. El aportante recibe una notificación que muestra la descripción proporcionada por el validador para la invalidación del documento. El aportante tiene la opción de sustituir el documento por uno nuevo adjuntándolo y cubriendo nuevamente sus características, o de responder al validador mediante un cuadro de texto argumentando el motivo por el que considera que el documento es válido.

**Flujo básico:**

1. El aportante recibe una notificación sobre la invalidación de un documento.
2. El sistema muestra la descripción proporcionada por el validador para la invalidación del documento.
3. El aportante accede a la notificación y revisa la descripción del motivo de invalidación.
4. El aportante decide corregir el documento incorrecto.
5. El aportante selecciona la opción para corregir el documento.
6. El sistema muestra la interfaz para adjuntar un nuevo documento y cubrir nuevamente sus características.
7. El aportante adjunta un nuevo documento y completa sus características.
8. El aportante confirma la corrección del documento

**Flujo alternativo:**

1. El aportante recibe una notificación sobre la invalidación de un documento.
2. El sistema muestra la descripción proporcionada por el validador para la invalidación del documento.
3. El aportante accede a la notificación y revisa la descripción del motivo de invalidación.
4. El aportante decide argumentar que el documento es válido.
5. El aportante selecciona la opción para responder al validador.
6. El sistema muestra un cuadro de texto para que el aportante argumente por qué considera que el documento es válido.
7. El aportante escribe su respuesta y la envía al validador.
8. El sistema registra la respuesta del aportante.

**Precondiciones:**

* El usuario aportante ha iniciado sesión en el sistema con sus credenciales válidas.
* Existe al menos un documento marcado como "Invalidado" por el usuario validador.
* El usuario aportante ha recibido una notificación sobre la invalidación del documento y tiene acceso a la descripción proporcionada por el usuario validador.

**Postcondición:**

El sistema registra la corrección del documento por parte del aportante o la respuesta al validador.

**Requisitos funcionales derivados:**

* RF19: Notificación de invalidación de documento.
* RF20: Acceso a la descripción del motivo de invalidación.
* RF21: Opción para corregir el documento.
* RF22: Opción para responder al validador.
* RF23: Cuadro de texto para argumentar la validez del documento.
* RF24: Registro de la corrección del documento por parte del aportante o la respuesta al validador.

**Diagrama de casos de uso:**

Diagrama

Descripción generada automáticamente

Figura 23: CU4 - Corregir documento incorrecto (fuente: elaboración propia)

|  |
| --- |
| **CU5 – Consultar documentación pendiente** |

**Actores principales:** Solicitante

**Descripción:** Este caso de uso permite a los usuarios solicitantes consultar un listado de toda la documentación pendiente de subir o corregir por parte de todos los usuarios aportantes existentes en el sistema. Esta funcionalidad resulta útil para verificar el estado de la documentación requerida y, en caso necesario, realizar reclamaciones o seguimientos correspondientes. El listado mostrará información relevante como el aportante al que se le ha requerido la documentación, la fecha en la que se realizó la solicitud y el título del documento solicitado.

**Flujo básico:**

1. El usuario solicitante accede a la funcionalidad de consulta de documentación pendiente.
2. El sistema muestra un listado de todas las solicitudes de documentación pendientes.
3. El usuario solicitante examina el listado para verificar el estado de la documentación requerida.
4. El usuario solicitante realiza las acciones necesarias en función del estado de la documentación pendiente.

**Precondiciones:**

El usuario solicitante tiene acceso al sistema y está autenticado.

**Postcondiciones:**

El usuario solicitante tiene información actualizada sobre la documentación pendiente en el sistema.

**Requisitos funcionales derivados:**

* RF25: El sistema debe mostrar un listado completo y actualizado de la documentación pendiente.
* RF26: Cada entrada en el listado debe incluir información detallada sobre el aportante, la fecha de solicitud y el título del documento.

**Diagrama de casos de uso:**

Diagrama

Descripción generada automáticamente

Figura 24: CU5 - Listar documentación pendiente (fuente: elaboración propia)

|  |
| --- |
| **CU6 – Descargar facturas** |

**Actores principales:** Cliente

**Descripción:** Este caso de uso permite a los clientes acceder a un portal de clientes en donde podrán consultar y descargar documentos que la empresa haya generado para ellos, tales como facturas. La funcionalidad proporciona a los clientes la conveniencia de acceder a su historial de facturación y obtener copias digitales de las facturas según sea necesario.

**Flujo básico:**

1. El cliente accede al portal de clientes en la aplicación web.
2. El sistema autentica al cliente y le permite acceder a su área personal.
3. El cliente navega hasta la sección de facturas.
4. El sistema muestra un listado de las facturas disponibles para su descarga.
5. El cliente selecciona la factura deseada para descargar.
6. El sistema permite al cliente descargar la factura en formato digital.

**Precondiciones:**

* El cliente tiene acceso al sistema y está autenticado como usuario registrado.
* El cliente tiene facturas generadas por la empresa.

**Postcondiciones:**

* El cliente ha descargado la factura seleccionada en formato digital.

**Requisitos funcionales derivados:**

* RF27: El sistema debe autenticar al cliente para acceder al portal de clientes.
* RF28: El sistema debe mostrar un listado completo y actualizado de las facturas del cliente disponibles para su descarga.
* RF29: El sistema debe permitir al cliente seleccionar y descargar una factura específica en formato digital.

**Diagrama de casos de uso:**

Diagrama

Descripción generada automáticamente

Figura 25: CU6 - Descargar facturas (fuente: elaboración propia)

#### Requisitos no funcionales

Los requisitos no funcionales son aquellas características del sistema que no están relacionadas directamente con su funcionalidad, pero que son igualmente importantes para garantizar su eficacia, seguridad y rendimiento.

En el contexto de la aplicación propuesta, se han identificado los siguientes requisitos no funcionales:

* La aplicación debe cumplir con estándares de seguridad para proteger la información confidencial de la empresa y de los usuarios.
* La aplicación debe ser capaz de manejar un gran volumen de documentos y usuarios simultáneamente sin experimentar retrasos significativos en la respuesta.
* La interfaz de usuario debe ser intuitiva y fácil de usar, permitiendo a los usuarios realizar sus tareas de gestión documental de manera eficiente y sin dificultades.
* La aplicación debe ser escalable, lo que significa que debe poder crecer y adaptarse fácilmente a medida que aumenta el número de usuarios y documentos.
* La aplicación debe estar disponible en todo momento, con un tiempo de inactividad mínimo.
* La aplicación debe cumplir con las regulaciones y normativas pertinentes en materia de protección de datos y privacidad, como el Reglamento General de Protección de Datos (GDPR)1.

Estos requisitos no funcionales serán tenidos en cuenta durante todo el proceso de diseño, desarrollo e implementación de la aplicación para asegurar que se cumplan los estándares de calidad y rendimiento esperados.

### Diseño

El diseño de la aplicación se ha concebido considerando cuidadosamente los requisitos identificados en la fase de análisis y las necesidades específicas de los usuarios finales.

La aplicación se ha construido sobre el stack tecnológico MERN (MongoDB, Express.js, React.js y Node.js), elegido por su flexibilidad, escalabilidad y eficiencia en el desarrollo de aplicaciones web modernas.

Este enfoque tecnológico ha permitido la implementación de una solución completa y robusta que cumple con los estándares más exigentes.

El diseño ha sido realizado siguiendo un enfoque modular, dividiendo las funcionalidades en componentes independientes para facilitar su gestión y mantenimiento. Se ha puesto especial énfasis en la usabilidad y la experiencia del usuario, asegurando que la aplicación sea intuitiva y fácil de usar para todos los usuarios.

Durante el proceso de diseño, se han tomado decisiones fundamentadas en los objetivos del proyecto y las mejores prácticas en el desarrollo de aplicaciones web. Cada elección ha sido cuidadosamente justificada para garantizar la coherencia y eficacia del diseño final.

A lo largo de este apartado, se presentan en detalle los diagramas, wireframes y otros artefactos visuales que ilustran la arquitectura y el flujo de la aplicación de manera clara y concisa. Se destacan las principales características del diseño, así como las soluciones implementadas para abordar los desafíos específicos del proyecto.

#### Arquitectura de la aplicación

El diseño arquitectónico de la aplicación propuesta se basa en una arquitectura de tres capas: presentación, lógica de negocio y persistencia de datos.

Esta arquitectura permite una separación clara de responsabilidades y facilita la escalabilidad y mantenibilidad del sistema.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 26: Arquitectura del sistema

Esta arquitectura de tres capas ofrece varias ventajas:

* Separación clara de responsabilidades: Cada capa tiene una función específica y bien definida, lo que facilita la comprensión y el mantenimiento del código.
* Escalabilidad: El modularidad de la arquitectura permite escalar cada capa por separado según sea necesario para manejar un mayor volumen de usuarios o datos.
* Flexibilidad: La arquitectura permite cambios y actualizaciones en cada capa de forma independiente sin afectar a las demás, lo que facilita la adaptación a los cambios en los requisitos del proyecto.

#### Tecnologías empleadas: El stack MERN

Un stack de desarrollo consiste en un conjunto de herramientas que facilitan las tareas de creación de aplicaciones web.

Entre los stacks más importantes actualmente gracias a la popularidad de JavaScript se encuentra MERN.

Según Smith, J, (2021) el stack MERN es una pila de tecnologías modernas utilizadas para desarrollar aplicaciones web altamente escalables y eficientes. Este conjunto de tecnologías incluye MongoDB, una base de datos NoSQL que proporciona una estructura flexible para el almacenamiento de datos; Express.js, un marco de aplicación web para Node.js que simplifica el desarrollo de servidores web y Apis; React, una biblioteca de JavaScript para construir interfaces de usuario interactivas y dinámicas; y Node.js, un entorno de ejecución de JavaScript del lado del servidor que permite la creación de aplicaciones web rápidas y escalables. Juntas, estas tecnologías forman un ecosistema completo que permite a los desarrolladores construir aplicaciones web modernas y de alto rendimiento de manera eficiente.

A continuación, procederé a brindar una breve descripción de los componentes que conforman el stack MERN, destacando las características principales de MongoDB, Express.js, React y Node.js, y explicando cómo cada uno de ellos contribuye al desarrollo de aplicaciones web modernas y eficientes:

* MongoDB:

Según la documentación oficial de MongoDB, MongoDB es una base de datos NoSQL de código abierto y orientada a documentos. En lugar de almacenar datos en tablas con filas y columnas, como lo hace una base de datos relacional, MongoDB almacena datos en documentos tipo JSON con un esquema dinámico, lo que permite una mayor flexibilidad y escalabilidad en el modelado de datos (MongoDB, Inc., 2020). Además, tal y como señalan Stonebraker y Madden (2014) en un artículo publicado en ACM Queue, MongoDB, junto con otras bases de datos NoSQL como Cassandra y Couchbase, ofrece ventajas significativas en términos de escalabilidad y rendimiento.

MongoDB, como componente clave del stack MERN, ofrece numerosas ventajas y características que lo convierten en una elección sólida para el desarrollo de aplicaciones web modernas y escalables. A continuación, se profundizará en algunos aspectos específicos de MongoDB que lo distinguen de las bases de datos relacionales tradicionales y que contribuyen a su éxito en el ecosistema MERN.

Una de las características más destacadas de MongoDB es su capacidad para escalar horizontalmente de manera eficiente. La escalabilidad horizontal se refiere a la capacidad de un sistema para crecer añadiendo más servidores o nodos a un clúster, distribuyendo así la carga de trabajo y mejorando el rendimiento (MongoDB, Inc., 2021a). MongoDB logra esto mediante el uso de un modelo de datos distribuido llamado "sharding". El sharding permite dividir una colección de documentos en múltiples fragmentos más pequeños, cada uno de los cuales se almacena en un servidor diferente. Esto permite que la base de datos maneje grandes volúmenes de datos y alto tráfico de manera eficiente, ya que las consultas y las operaciones de escritura pueden distribuirse entre los diferentes fragmentos (MongoDB, Inc., 2021b). Además, MongoDB proporciona una funcionalidad de balanceo de carga automático que garantiza que los datos se distribuyan uniformemente entre los fragmentos, optimizando así el rendimiento y evitando cuellos de botella (Hows et al., 2014).

Otra ventaja significativa de MongoDB es su modelo de datos flexible basado en documentos. A diferencia de las bases de datos relacionales que utilizan un esquema rígido con tablas y filas, MongoDB almacena los datos en documentos BSON (Binary JSON), que son estructuras de datos similares a JSON (MongoDB, Inc., 2021c). Este enfoque permite una mayor flexibilidad y agilidad en el modelado de datos, ya que cada documento puede tener su propia estructura y no está limitado por un esquema predefinido. Esto es especialmente beneficioso en entornos de desarrollo ágil y en aplicaciones web que requieren frecuentes cambios en la estructura de datos (Banker, 2011). Además, el modelo de documentos de MongoDB permite almacenar datos relacionados en un solo documento, lo que reduce la necesidad de realizar uniones complejas entre tablas y mejora el rendimiento de las consultas (Chodorow, 2013).

MongoDB también se destaca por su potente lenguaje de consulta. Además de las operaciones CRUD básicas (Crear, Leer, Actualizar y Eliminar), MongoDB proporciona un amplio conjunto de operadores y métodos de consulta que permiten realizar búsquedas y agregaciones complejas de manera eficiente (MongoDB, Inc., 2021d). El lenguaje de consulta de MongoDB admite consultas avanzadas, como consultas de rango, consultas geoespaciales, consultas de texto y consultas basadas en expresiones regulares. Además, MongoDB ofrece un marco de agregación flexible que permite realizar análisis de datos complejos, como agrupaciones, proyecciones y transformaciones de datos, directamente en la base de datos (MongoDB, Inc., 2021e). Esto elimina la necesidad de transferir grandes cantidades de datos al servidor de aplicaciones para su procesamiento, lo que mejora el rendimiento y la eficiencia general del sistema (Hows et al., 2014).

Otra ventaja de MongoDB en el contexto del stack MERN es su estrecha integración con Node.js a través de drivers oficiales. MongoDB proporciona un driver nativo para Node.js llamado "MongoDB Node.js Driver", que facilita la interacción entre la aplicación Node.js y la base de datos MongoDB (MongoDB, Inc., 2021f). Este driver ofrece una API intuitiva y eficiente para realizar operaciones de base de datos, como inserción, consulta, actualización y eliminación de documentos. Además, el driver de Node.js de MongoDB admite características avanzadas, como transacciones multi-documento, control de concurrencia optimista y soporte para promesas y async/await, lo que mejora la legibilidad y el manejo de errores en el código (MongoDB, Inc., 2021g). La integración perfecta entre MongoDB y Node.js permite a los desarrolladores aprovechar al máximo las capacidades de ambas tecnologías y construir aplicaciones web de alto rendimiento y escalabilidad.

MongoDB brinda características y ventajas específicas que lo convierten en una elección sólida para el desarrollo de aplicaciones web en el stack MERN. Su capacidad para escalar horizontalmente mediante el uso de sharding permite manejar grandes volúmenes de datos y alto tráfico de manera eficiente. Su modelo de datos flexible basado en documentos proporciona agilidad y adaptabilidad en el modelado de datos, lo que es especialmente beneficioso en entornos de desarrollo ágil. Además, el potente lenguaje de consulta de MongoDB y su marco de agregación permiten realizar consultas y análisis complejos de manera eficiente. Por último, la estrecha integración de MongoDB con Node.js a través de drivers oficiales facilita la interacción entre la aplicación y la base de datos, aprovechando al máximo las capacidades de ambas tecnologías. Estas características y ventajas hacen de MongoDB una elección atractiva para el desarrollo de aplicaciones web modernas y escalables en el stack MERN.

* Express.js:

Tal y como se menciona en su documentación oficial, Express.js es un marco de aplicación web para Node.js que simplifica el desarrollo de servidores web y APIs altamente conocido por su minimalismo y flexibilidad, lo que lo convierte en una herramienta popular entre los desarrolladores (Express.js, 2021). Además, tal y como Wilson (2018) destaca en su libro "Express.js in Action”, posee numerosas características clave, asi como un amplio ecosistema de middleware y complementos, que ofrecen una variedad de funciones adicionales y características para mejorar la productividad y la seguridad de las aplicaciones.

Express.js es un marco de aplicación web para Node.js que ha ganado gran popularidad entre los desarrolladores debido a su enfoque minimalista y su flexibilidad para crear aplicaciones web y APIs de manera eficiente. Su filosofía de diseño se centra en proporcionar un conjunto básico de características, permitiendo a los desarrolladores extender y personalizar la funcionalidad según las necesidades específicas de cada proyecto (Express.js, 2021).

Una de las principales ventajas de Express.js es su minimalismo. A diferencia de otros marcos web que imponen una estructura rígida y una gran cantidad de código boilerplate, Express.js se mantiene ligero y no intrusivo. Esto significa que los desarrolladores tienen la libertad de organizar y estructurar su código de la manera que mejor se adapte a su proyecto, sin estar limitados por convenciones estrictas (Hahn, 2016). Este enfoque minimalista también se traduce en un mejor rendimiento, ya que Express.js no introduce una sobrecarga innecesaria y permite un mayor control sobre el flujo de la aplicación.

Otra característica destacada de Express.js es su sistema de enrutamiento flexible y poderoso. Express.js permite definir rutas de manera intuitiva y concisa, utilizando una sintaxis clara y expresiva. Los desarrolladores pueden definir rutas para diferentes métodos HTTP (GET, POST, PUT, DELETE, etc.) y especificar los manejadores correspondientes para procesar las solicitudes entrantes (Brown, 2014). Además, Express.js admite parámetros de ruta y consulta, lo que facilita la extracción de datos de la URL y la manipulación de los mismos en el servidor. Esta flexibilidad en el enrutamiento permite a los desarrolladores crear APIs RESTful de manera sencilla y eficiente.

Express.js también se destaca por su poderoso sistema de middleware. El middleware son funciones que se ejecutan en el flujo de procesamiento de una solicitud, permitiendo realizar tareas como el análisis de datos, la autenticación, el manejo de errores, entre otros (Hahn, 2016). Express.js proporciona una amplia gama de middleware incorporado, como body-parser para analizar los datos del cuerpo de las solicitudes, cookie-parser para manejar cookies, y cors para habilitar el intercambio de recursos de origen cruzado (CORS). Además, los desarrolladores pueden crear su propio middleware personalizado para adaptarse a los requisitos específicos de su aplicación. La capacidad de encadenar y combinar middleware permite un gran control y flexibilidad en el procesamiento de las solicitudes. Otra ventaja de Express.js es su compatibilidad con múltiples motores de plantillas. Los motores de plantillas permiten generar dinámicamente páginas HTML en el servidor, facilitando la creación de interfaces de usuario interactivas y reutilizables (Holmes, 2019). Express.js es compatible con una amplia variedad de motores de plantillas populares, como EJS, Pug (anteriormente Jade), Handlebars y Mustache. Esto brinda a los desarrolladores la libertad de elegir el motor de plantillas que mejor se adapte a sus preferencias y requisitos del proyecto. Además, Express.js permite configurar fácilmente el motor de plantillas y definir la ubicación de las vistas en la estructura del proyecto.

Uno de los aspectos más destacados de Express.js es su extenso ecosistema de plugins y middlewares desarrollados por la comunidad. El ecosistema de Express.js cuenta con una gran cantidad de módulos y bibliotecas adicionales que amplían su funcionalidad y facilitan el desarrollo de aplicaciones web (Wilson, 2018). Estos plugins cubren una amplia gama de características, como el manejo de sesiones, la autenticación de usuarios, la integración con bases de datos, el caché, la compresión de respuestas, entre otros. La disponibilidad de estos plugins permite a los desarrolladores ahorrar tiempo y esfuerzo al no tener que implementar estas funcionalidades desde cero. Además, la comunidad activa de Express.js garantiza que estos plugins se mantengan actualizados y se corrijan los problemas de seguridad de manera oportuna.

Express.js se ha convertido en uno de los marcos de aplicación web más populares para Node.js debido a su enfoque minimalista, su flexibilidad en el enrutamiento y middleware, su compatibilidad con múltiples motores de plantillas y su extenso ecosistema de plugins. Estas características permiten a los desarrolladores crear aplicaciones web y APIs de manera eficiente, adaptándose a los requisitos específicos de cada proyecto. El minimalismo de Express.js promueve un mejor rendimiento y un mayor control sobre el flujo de la aplicación, mientras que su sistema de middleware y su ecosistema de plugins brindan una amplia gama de funcionalidades adicionales para mejorar la productividad y la seguridad. Con su enfoque centrado en la simplicidad y la flexibilidad, Express.js se ha consolidado como una herramienta fundamental en el desarrollo web moderno con Node.js.

* React:

React es una biblioteca de JavaScript de código abierto desarrollada por Facebook que se utiliza para construir interfaces de usuario interactivas y dinámicas para aplicaciones web. Se centra en la creación de componentes reutilizables que representan diferentes partes de la interfaz de usuario, lo que facilita el desarrollo y mantenimiento de aplicaciones complejas. React utiliza un enfoque basado en componentes y el concepto de unidireccionalidad de datos para mejorar el rendimiento y la escalabilidad de las aplicaciones web modernas. (React, 2021).

Una de las características fundamentales de React es su enfoque basado en componentes. Los componentes son bloques de construcción reutilizables que encapsulan la lógica y la presentación de una parte específica de la interfaz de usuario (Wieruch, 2017). En React, los componentes se definen como funciones o clases que aceptan propiedades (props) y devuelven elementos JSX, que describen cómo debe renderizarse la interfaz de usuario. Este enfoque modular permite a los desarrolladores descomponer una aplicación compleja en piezas más pequeñas y manejables, lo que facilita el desarrollo, el mantenimiento y la reutilización de código.

Otra ventaja significativa de React es su uso del Virtual DOM (DOM Virtual). El Virtual DOM es una representación ligera y en memoria del DOM (Document Object Model) real (Hunt, 2019). Cuando se producen cambios en el estado de la aplicación, React crea una nueva versión del Virtual DOM y la compara con la versión anterior. Luego, calcula la diferencia (diff) entre ambas versiones y aplica de manera eficiente solo los cambios necesarios al DOM real. Este proceso de reconciliación optimiza el rendimiento al minimizar las manipulaciones directas del DOM, que pueden ser costosas en términos de rendimiento. Como resultado, las aplicaciones de React son altamente eficientes y reactivas, incluso cuando se manejan grandes conjuntos de datos y actualizaciones frecuentes de la interfaz de usuario.

El flujo de datos unidireccional es otro principio clave de React. En React, los datos fluyen en una sola dirección, desde los componentes principales hasta los componentes secundarios (Banks & Porcello, 2017). Los componentes principales pasan datos a los componentes secundarios a través de props, y los componentes secundarios pueden notificar a los componentes principales sobre cambios o eventos mediante el uso de funciones de devolución de llamada (callback functions). Este enfoque unidireccional facilita la comprensión y el seguimiento del flujo de datos en una aplicación, lo que a su vez mejora la previsibilidad y la mantenibilidad del código. Además, el flujo de datos unidireccional promueve la separación de preocupaciones (separation of concerns) al mantener el estado de la aplicación en los componentes principales y delegar la presentación y la interacción a los componentes secundarios.

React cuenta con una amplia y activa comunidad de desarrolladores que contribuyen al crecimiento y la mejora continua de la biblioteca. La comunidad de React ha creado un vasto ecosistema de herramientas, bibliotecas y frameworks complementarios que amplían las capacidades de React y facilitan el desarrollo de aplicaciones web (Borgen, 2018). Algunas de las herramientas populares en el ecosistema de React incluyen Create React App para la configuración rápida de proyectos, React Router para el enrutamiento declarativo, Redux para la gestión del estado global de la aplicación, y React Native para el desarrollo de aplicaciones móviles nativas utilizando React. Además, la comunidad de React organiza conferencias, meetups y workshops en todo el mundo, lo que fomenta el intercambio de conocimientos, las mejores prácticas y la colaboración entre desarrolladores. Otra ventaja de React es su curva de aprendizaje relativamente suave en comparación con otros frameworks y bibliotecas de JavaScript. React se centra en un conjunto de conceptos básicos, como componentes, props y estado, que son fáciles de entender y aplicar (Wieruch, 2017). Además, React utiliza JSX, una extensión de sintaxis para JavaScript que permite escribir código HTML-like dentro de JavaScript. JSX mejora la legibilidad y la expresividad del código, ya que combina la lógica y la presentación en un solo lugar. La simplicidad y la claridad de React hacen que sea accesible para desarrolladores con diferentes niveles de experiencia, lo que contribuye a su adopción generalizada en la industria del desarrollo web.

React se ha convertido en una herramienta indispensable para el desarrollo de interfaces de usuario modernas y dinámicas. Su enfoque basado en componentes promueve la reutilización y la modularidad del código, mientras que su uso del Virtual DOM optimiza el rendimiento al minimizar las manipulaciones directas del DOM. El flujo de datos unidireccional de React facilita la comprensión y el mantenimiento de aplicaciones complejas, y su amplio ecosistema y comunidad activa brindan un conjunto extenso de herramientas y recursos para mejorar el desarrollo. Con su curva de aprendizaje accesible y su enfoque en la simplicidad, React ha democratizado el desarrollo web y ha permitido a los desarrolladores crear experiencias de usuario excepcionales de manera eficiente. A medida que la web continúa evolucionando, React se mantiene a la vanguardia, impulsando la innovación y definiendo las mejores prácticas en el desarrollo de aplicaciones web.

* Node.js:

Node.js es un entorno de ejecución de JavaScript del lado del servidor que permite a los desarrolladores crear aplicaciones web escalables y de alto rendimiento. Desarrollado sobre el motor de JavaScript V8 de Google Chrome, Node.js proporciona una plataforma para ejecutar JavaScript en el servidor, lo que permite a los desarrolladores utilizar un único lenguaje de programación tanto en el lado del cliente como en el servidor. (Node.js, 2021)

Una de las características clave de Node.js es su capacidad para realizar operaciones de entrada/salida de manera asincrónica y no bloqueante, lo que permite manejar múltiples solicitudes simultáneamente sin la necesidad de procesos adicionales o subprocesos. Esto hace que Node.js sea ideal para aplicaciones web que requieren una alta concurrencia y rendimiento. (Hughes-Croucher & Wilson, 2013)

Node.js cuenta con un amplio ecosistema de módulos y paquetes que pueden ser fácilmente instalados y gestionados a través de npm, el gestor de paquetes de Node.js. Esto permite a los desarrolladores aprovechar una gran variedad de herramientas y bibliotecas para el desarrollo de aplicaciones web, desde frameworks como Express.js hasta bibliotecas para acceso a bases de datos como Mongoose. (Node.js, 2021)

Además, Node.js es ampliamente utilizado en la industria, desde startups hasta grandes empresas, debido a su eficiencia, escalabilidad y facilidad de aprendizaje. Su popularidad ha llevado a la creación de una comunidad activa de desarrolladores que contribuyen con código, documentación y recursos de aprendizaje para impulsar el crecimiento y la adopción de Node.js en todo el mundo. (Hughes-Croucher & Wilson, 2013).

Node.js cuenta con un amplio ecosistema de módulos y paquetes que pueden ser fácilmente instalados y gestionados a través de npm, el gestor de paquetes de Node.js. Esto permite a los desarrolladores aprovechar una gran variedad de herramientas y bibliotecas para el desarrollo de aplicaciones web, desde frameworks como Express.js hasta bibliotecas para acceso a bases de datos como Mongoose. (Node.js, 2021)

Además, Node.js es ampliamente utilizado en la industria, desde startups hasta grandes empresas, debido a su eficiencia, escalabilidad y facilidad de aprendizaje. Su popularidad ha llevado a la creación de una comunidad activa de desarrolladores que contribuyen con código, documentación y recursos de aprendizaje para impulsar el crecimiento y la adopción de Node.js en todo el mundo. (Hughes-Croucher & Wilson, 2013).

Una de las características más destacadas de Node.js es su modelo de entrada/salida (E/S) no bloqueante. A diferencia de los modelos tradicionales de E/S bloqueantes, donde una operación de E/S bloquea la ejecución del programa hasta que se completa, Node.js utiliza un enfoque asincrónico y basado en eventos. Esto significa que cuando se inicia una operación de E/S, como la lectura de un archivo o una consulta a una base de datos, Node.js no espera a que se complete antes de continuar con la ejecución del código. En su lugar, Node.js registra una función de devolución de llamada (callback) que se ejecutará una vez que la operación de E/S se haya completado, permitiendo que el programa continúe con otras tareas mientras tanto. (Cantelon et al., 2014)

Este modelo de E/S no bloqueante es especialmente beneficioso para aplicaciones web que requieren un alto grado de concurrencia y capacidad de respuesta. Al evitar el bloqueo durante las operaciones de E/S, Node.js puede manejar eficientemente múltiples solicitudes simultáneas sin la necesidad de crear subprocesos o procesos adicionales. Esto se traduce en un mejor aprovechamiento de los recursos del sistema y una mayor escalabilidad de la aplicación. (Cantelon et al., 2014)

Gracias a su modelo de E/S no bloqueante, Node.js es capaz de manejar una gran cantidad de conexiones simultáneas de manera eficiente. En un servidor web tradicional, cada conexión entrante requiere un hilo o proceso separado para ser atendida. Esto puede llevar a una sobrecarga significativa del sistema y limitar la capacidad del servidor para manejar un gran número de conexiones concurrentes. (Wandschneider, 2013)

En contraste, Node.js utiliza un modelo de evento único (single-threaded event loop) para manejar las conexiones. Cuando llega una solicitud, Node.js la coloca en una cola de eventos y continúa atendiendo otras solicitudes. Cuando una operación de E/S se completa, Node.js genera un evento y ejecuta la función de devolución de llamada asociada. Este enfoque permite que Node.js maneje miles de conexiones simultáneas sin la necesidad de crear hilos o procesos adicionales, lo que resulta en un menor consumo de recursos y una mayor escalabilidad. (Wandschneider, 2013)

Otro aspecto destacado de Node.js es su extenso ecosistema de paquetes y módulos, conocido como npm (Node Package Manager). npm es el registro de paquetes más grande del mundo, con más de un millón de paquetes disponibles para su uso en proyectos Node.js. (npm, 2021)

La amplia variedad de paquetes disponibles en npm cubre una amplia gama de funcionalidades, desde frameworks web como Express.js y Koa.js hasta bibliotecas para acceso a bases de datos, manipulación de archivos, autenticación, procesamiento de imágenes y mucho más. Estos paquetes son desarrollados y mantenidos por la comunidad de desarrolladores de Node.js lo que garantiza su calidad y actualización constante. (npm, 2021)

La facilidad de instalación y gestión de paquetes a través de npm es una de las razones por las que Node.js se ha vuelto tan popular entre los desarrolladores. Con un simple comando, los desarrolladores pueden instalar y utilizar cualquier paquete disponible en el registro, lo que acelera el proceso de desarrollo y permite la reutilización de código. Además, npm proporciona un sistema de gestión de dependencias que garantiza que las versiones correctas de los paquetes estén instaladas y sean compatibles entre sí. (Cantelon et al., 2014)

Node.js se destaca por su portabilidad entre plataformas. Al estar construido sobre el motor de JavaScript V8 de Google Chrome, Node.js puede ejecutarse en una variedad de sistemas operativos, incluyendo Windows, macOS y Linux. Esto permite a los desarrolladores crear aplicaciones que puedan desplegarse en diferentes entornos sin la necesidad de realizar cambios significativos en el código. (Node.js, 2021)

La portabilidad de Node.js también se extiende a las aplicaciones web. Las aplicaciones desarrolladas con Node.js pueden ejecutarse tanto en el lado del servidor como en el lado del cliente, utilizando el mismo lenguaje de programación (JavaScript) en ambos entornos. Esto simplifica el desarrollo y reduce la curva de aprendizaje para los desarrolladores, ya que pueden utilizar sus habilidades de JavaScript tanto en el frontend como en el backend. (Node.js, 2021)

Además, la portabilidad de Node.js ha impulsado su adopción en diversos escenarios, desde el desarrollo de aplicaciones web tradicionales hasta el desarrollo de aplicaciones móviles híbridas y aplicaciones de escritorio utilizando frameworks como Electron. Esto ha ampliado aún más las posibilidades y los casos de uso de Node.js en la industria del desarrollo de software. (Hughes-Croucher & Wilson, 2013)

Node.js ha revolucionado la forma en que se desarrollan las aplicaciones web, ofreciendo un enfoque único y poderoso para el desarrollo del lado del servidor. Su modelo de entrada/salida no bloqueante, su capacidad para manejar múltiples conexiones simultáneas, su extenso registro de paquetes (npm) y su portabilidad entre plataformas lo convierten en una herramienta indispensable para los desarrolladores modernos.

Al aprovechar estas características y ventajas, los desarrolladores pueden crear aplicaciones web escalables, eficientes y de alto rendimiento. Node.js ha demostrado ser una tecnología confiable y ampliamente adoptada en la industria, utilizada por empresas de todos los tamaños para construir aplicaciones web robustas y de calidad. Con su creciente popularidad y una comunidad activa de desarrolladores, Node.js continúa evolucionando y expandiendo sus capacidades. A medida que surgen nuevas tendencias y desafíos en el desarrollo web, Node.js está bien posicionado para adaptarse y proporcionar soluciones innovadoras.

En resumen, Node.js es una herramienta poderosa que ha transformado el panorama del desarrollo web. Sus características únicas y ventajas lo convierten en una elección sólida para los desarrolladores que buscan crear aplicaciones web modernas, escalables y de alto rendimiento.

La siguiente figura muestra de manera simple el funcionamiento de una aplicación web desarrollada con el stack MERN y las funciones de cada una de sus herramientas.

Diagrama

Descripción generada automáticamente

Figura 27: Stack MERN (Fuente: https://www.mongodb.com/mern-stack)

Según describe el documento "Análisis del Stack Mern y su uso en las aplicaciones web basadas en servicios Rest.", "el usuario envía una petición al servidor a través de la aplicación web, es decir que este interactúa desde el Frontend mediante React.

Node.js recibe la petición del cliente, la analiza y la maneja.

Express es el encargado de enviar la petición a MongoDB.

Lo mismo pasa cuando Express recibe los datos de MongoDB es el encargado de devolver la respuesta.

Al igual que Express, Node.js recibe la respuesta y es el encargado de devolver la respuesta de la petición hecha por el cliente al Frontend de la aplicación web.

Por último, React recibe la respuesta y la presenta al cliente."

La elección del stack MERN para el desarrollo propuesto por el presente trabajo puede justificarse por varias razones:

**Flexibilidad en el manejo de datos:** MongoDB, como base de datos NoSQL, ofrece flexibilidad en el almacenamiento de documentos, lo que se adapta bien a la naturaleza variada de la documentación que la empresa ha de gestionar. Además, su capacidad para escalar horizontalmente es beneficiosa para manejar los grandes volúmenes de datos que pueden llegar a acumularse en la gestión documental.

**Eficiencia en el desarrollo:** Express.js, como marco de aplicación web de Node.js, simplifica el desarrollo del backend al proporcionar una estructura y funciones comunes para manejar rutas, solicitudes HTTP y otras tareas relacionadas con el servidor. Esto acelera el proceso de desarrollo al tiempo que garantiza un código limpio y mantenible.

**Interactividad y experiencia de usuario:** React, como biblioteca de JavaScript para construir interfaces de usuario interactivas, puede mejorar la experiencia del usuario al proporcionar una interfaz de usuario dinámica y receptiva. Esto es crucial para una aplicación de gestión documental, donde los usuarios pueden interactuar con una gran cantidad de documentos y realizar diversas acciones sobre ellos.

**Compatibilidad con tecnologías modernas:** El stack MERN se alinea con las tecnologías modernas y las mejores prácticas de desarrollo web, lo que garantiza que la aplicación resultante sea escalable, segura y fácil de mantener. Además, la comunidad activa y los recursos disponibles para cada una de las tecnologías en el stack MERN facilitan el desarrollo y la resolución de problemas durante el ciclo de vida de la aplicación.

En resumen, la elección del stack MERN se justifica por su flexibilidad en el manejo de datos, eficiencia en el desarrollo, interactividad y experiencia de usuario, así como su compatibilidad con tecnologías modernas y mejores prácticas de desarrollo web. Esto asegura que la aplicación sea capaz de satisfacer las necesidades de la empresa y ofrecer una experiencia de Alternativas

Existen diversas alternativas al stack MERN (MongoDB, Express, React, Node.js), que se utilizan en el desarrollo web según las necesidades específicas de la aplicación y las preferencias del equipo de desarrollo. Las más conocidas son MEAN y MEVN, que comparten algunas tecnologías con MERN pero varían principalmente en el frontend utilizado:

* + **MEAN (MongoDB, Express, Angular, Node.js):**
  + **Frontend: Angular:** Angular es un framework completo que incluye una gran cantidad de funcionalidades integradas, como manejo de estado, enrutamiento y formularios, lo cual puede ser más conveniente para aplicaciones complejas que requieren una arquitectura robusta y un conjunto coherente de herramientas. Sin embargo, puede resultar más pesado y menos flexible en comparación con React debido a su naturaleza prescriptiva.
  + **Comparación con MERN:** La principal diferencia radica en el framework de frontend. React, utilizado en MERN, es más una biblioteca que un framework y ofrece más flexibilidad, permitiendo a los desarrolladores elegir las bibliotecas adicionales que deseen para cosas como el enrutamiento y el manejo de estado. Angular, en cambio, es más estructurado, lo que puede facilitar el desarrollo en equipos grandes o proyectos donde se desea una mayor coherencia.
  + **MEVN (MongoDB, Express, Vue.js, Node.js):**

- **Frontend: Vue.js:** Vue.js es conocido por su facilidad de aprendizaje y su integración gradual, lo que significa que los desarrolladores pueden usar Vue en la escala que necesiten, similar a React pero con una sintaxis que algunos encuentran más intuitiva. Vue ofrece una combinación equilibrada de características de Angular y React, como un sistema de reactividad eficiente y un conjunto de herramientas opcional.

- **Comparación con MERN:** Vue.js, al igual que React, proporciona flexibilidad y es menos prescriptivo que Angular. Los desarrolladores que eligen Vue generalmente aprecian su simplicidad y su curva de aprendizaje menos pronunciada en comparación con Angular. Al igual que en MERN, los desarrolladores pueden agregar funcionalidades según sea necesario.

* **PERN (PostgreSQL, Express, React, Node.js):**

- Utiliza PostgreSQL como base de datos relacional en lugar de MongoDB.

- PostgreSQL ofrece características avanzadas como transacciones ACID, consultas complejas y soporte para datos estructurados.

- Puede ser más adecuado para aplicaciones con requisitos de datos más estrictos y relacionales.

**• Django (Python) + React:**

- Utiliza Django, un framework web de Python, en el backend en lugar de Node.js y Express.

- Django proporciona un ORM potente, una interfaz de administración automática y un conjunto completo de características out-of-the-box.

- Puede ser una buena opción para desarrolladores con experiencia en Python y que buscan un framework web completo.

• **Laravel (PHP) + React:**

- Emplea Laravel, un framework web de PHP, en el backend en lugar de Node.js y Express.

- Laravel ofrece un ORM elocuente, un sistema de migraciones de bases de datos y una sintaxis expresiva para el desarrollo backend.

- Puede ser atractivo para desarrolladores con experiencia en PHP y que desean aprovechar las características y el ecosistema de Laravel.

La siguiente tabla muestra, de forma resumida, una comparativa entre todas las opciones planteadas anteriormente:

|  |  |  |  |
| --- | --- | --- | --- |
| **Stack** | **Ventajas** | **Inconvenientes** | **Curva de aprendizaje** |
| MERN (MongoDB, Express, React, Node.js) | * Uso de JS en todo el stack * Gran comunidad de React * Flexible | * Menor consistencia de datos * Requiere manejo avanzado de JS y React | Moderada a alta |
| MEAN (MongoDB, Express, Angular, Node.js) | * Angular es un framework complet * Ideal para aplicaciones empresariales | * Angular es complejo * Puede ser rígido para proyectos pequeños | Alta |
| MEVN (MongoDB, Express, Vue.js, Node.js) | * Vue.js fácil de aprender * Documentación clara y detallada | * Menos popular que Angular o React * Comunidad más pequeña | Moderada |
| PERN (PostgreSQL, Express, React, Node.js) | * PostgreSQL robusto para gestión de datos * Soporte avanzado de SQL | * Requiere conocimientos de SQL * Configuración inicial compleja | Moderada a alta |
| Django (Python) + React | * Django muy completo y seguro * Versatilidad de Python | * Mezcla de Python y JS * Complicado para mantenimiento | Alta |
| Laravel (PHP) + React | * Laravel ofrece sintaxis elegante * Robustas características de desarrollo web | * Mezcla de PHP y JS * Requiere configuración adicional | Alta |

Tabla 4: Tabla comparativa de Stacks

#### Capa de presentación

La capa de presentación es una parte fundamental de la aplicación, ya que es la interfaz a través de la cual los usuarios interactúan con el sistema.

En esta capa, el trabajo se ha centrado en el diseño y desarrollo de interfaces de usuario intuitivas, eficientes y atractivas que faciliten la navegación y el uso de la aplicación.

La siguiente figura presenta un diseño de la interfaz correspondiente a la funcionalidad de subida de documentación. Esta interfaz permite a los usuarios aportantes cargar los documentos solicitados y proporcionar la información necesaria asociada a cada documento.

Calendario

Descripción generada automáticamenteFigura 28: Implementación de la vista "Subida de documentación"

En el diseño de la interfaz se ha priorizado la claridad, la simplicidad y la intuitividad para ofrecer una experiencia de usuario óptima. Las características clave de este diseño incluyen:

* **Interfaz Limpia:** Se ha creado una interfaz minimalista para facilitar la navegación y el uso intuitivo. Se ha evitado la sobrecarga de información para que los usuarios puedan concentrarse en la tarea de subir documentos de manera efectiva.
* **Navegación Guiada:** Para evitar el scroll y proporcionar una navegación fluida, se ha implementado una navegación guiada mediante flechas en la parte inferior de la interfaz. Estas flechas permiten al usuario avanzar al siguiente documento de la petición o regresar al documento anterior de manera sencilla y rápida.
* **Proceso “Paso a Paso”**: Se ha estructurado el proceso de subida de documentos en una serie de pasos, permitiendo al usuario completar la información y adjuntar el documento para cada elemento de la petición. Esto facilita la organización y la comprensión del proceso.
* **Botón de Guardar:** En cada paso del proceso, se ha incluido un botón de guardar que permite al usuario salvar los cambios realizados para el documento correspondiente. Esto asegura que los usuarios puedan realizar modificaciones en una petición de forma incremental y segura.
* **Botón de Salir:** Para proporcionar una salida clara del proceso de subida de documentos, se ha añadido un botón en la parte inferior de la interfaz. Este botón permite al usuario volver al listado de peticiones en cualquier momento, ofreciendo una navegación coherente y fácil de entender.

Este diseño ha sido cuidadosamente elaborado para garantizar una experiencia de usuario fluida y eficiente, permitiendo a los usuarios completar la tarea de subida de documentos de manera rápida y sin complicaciones al igual que se realizará con el resto de las vistas que componen el proyecto.

#### Lógica de negocio

La lógica de negocio de la aplicación es fundamental para garantizar el correcto funcionamiento y la coherencia de todas las operaciones realizadas dentro del sistema.

En esta sección, se presentan los diagramas de secuencia que describen las interacciones entre los diferentes actores y componentes de la aplicación en respuesta a ciertos eventos clave.

El siguiente diagrama de secuencia muestra el flujo de acciones cuando un usuario inicia sesión en la aplicación y crea una solicitud de documentación. Este proceso implica la interacción entre el usuario solicitante y la aplicación para completar y enviar la solicitud correspondiente.

Diagrama

Descripción generada automáticamente

Figura 29: Diagrama de secuencia para solicitud de documentación

El segundo diagrama de secuencia ilustra el proceso de envío de documentación solicitada por parte de un usuario aportante y su validación por parte del usuario validador.

Diagrama

Descripción generada automáticamente con confianza mediaEste flujo detalla las acciones realizadas por cada actor y cómo interactúan con la aplicación para completar este proceso.

Figura 30: Diagrama de secuencia para solicitud de documentación

#### Capa de datos

La capa de datos de la aplicación propuesta juega un papel fundamental en el almacenamiento y gestión eficiente de la información relacionada con los documentos, usuarios y otras entidades pertinentes. Para diseñar la base de datos de esta aplicación, se ha optado por utilizar MongoDB, una base de datos NoSQL que ofrece flexibilidad y escalabilidad para manejar grandes volúmenes de datos.

* **Diseño de la Base de Datos**

El diseño de la base de datos se ha realizado teniendo en cuenta las entidades principales de la aplicación y sus relaciones.

A continuación, se presenta una descripción de las colecciones principales y sus campos asociados:

|  |  |
| --- | --- |
| **Tabla** | **Campos** |
| **Usuarios** | * ID de usuario * Nombre * Correo electrónico * Contraseña cifrada * Rol (solicitante, aportante, validador, administrador) |
| **Documentos** | * ID de documento * ID de usuario (el usuario que subió el documento) * Nombre del documento * Descripción * Fecha de subida * Estado (pendiente, validado, invalidado) * URL de archivo |
| **Solicitudes** | * ID de solicitud * ID de usuario solicitante * Fecha de solicitud * Estado (pendiente, completada) * Descripción de la solicitud |
| **Clientes** | * ID de cliente * Nombre de la empresa * Correo electrónico del cliente * Contraseña cifrada del cliente |

Tabla 5: Diccionario de datos de la BD

El diseño propuesto plantea la existencia de las siguientes relaciones entre las colecciones:

* La colección de Usuarios está relacionada con las colecciones de Documentos y Solicitudes a través de los campos de ID de usuario.
* La colección de Documentos también está relacionada con la colección de Solicitudes mediante el ID de solicitud.

### Implementación

### Pruebas

# Conclusiones y trabajo futuro

## Conclusiones del trabajo

## Líneas de trabajo futuro

Referencias bibliográficas

**Arias, F. J. T., Zúñiga, K. M., Zambrano, M. M. T., y Cruzatty, J. E. Á. (2021). Impacto del intranet y extranet en el desarrollo empresarial. Serie Científica de la Universidad de las Ciencias Informáticas, 14(9), 28-41. https://dialnet.unirioja.es/servlet/articulo?codigo=8590642**

**Atarama Vásquez, D. L., y Villanueva Quispe, J. A. (2022). Método basado en la ISO 30302 para la gestión documental de recursos bibliográficos en instituciones de educación básico regular apoyado en una aplicación web. https://repositorio.ucv.edu.pe/handle/20.500.12692/99343**

**Barrio Andrés, M. (2021). Formación y evolución de los derechos digitales. https://www.torrossa.com/it/resources/an/5595640**

**Bayona, J., y Meneses, M. (2020). Sistema de gestión documental para la gestión de titulaciones para la vicerrectoría de calidad e innovación académica de la Universidad Europea de Madrid. (Tesis de grado). Universidad Peruana de Ciencias Aplicadas, Lima. http://hdl.handle.net/10757/648599**

**Cacheda Couto, J. (2022). Cómo mejorar la calidad de vida de los usuarios gracias al marketing 5.0. https://ruc.udc.es/dspace/handle/2183/32123**

**Calleja-Lopez, A., Cancela, E., y Cambronero Garbajosa, M. (2022). Desplazar los ejes: alternativas tecnológicas, derechos humanos y sociedad civil a principios del siglo XXI. https://openaccess.uoc.edu/handle/10609/147765**

* **Chaffey, D., & White, G. (2011). Business information management: Improving performance using information systems. Pearson Education Limited.**

**Charan, R. (2021). Repensar la ventaja competitiva: Nuevas reglas para la era digital. Conecta.**

**Chicaiza-Llumiquinga, Y. N., y Guevara-Miranda, J. X. (2023). Construcción de un aplicativo web que permite ejecutar un plan de mantenimiento preventivo y correctivo de software y hardware de los equipos TIC’S en la radio stereo Latacunga ubicado en el cantón Latacunga. https://repositorio.espe.edu.ec/bitstream/21000/36090/1/ESPEL-SIT-0144-P.pdf**

**Confederación de empresarios de Málaga (2021). Guía de digitalización documental de la empresa. Diputación de Málaga. https://www.cem-malaga.es/portalcem/novedades/2021/Guia%20digitalizacion%20en%20la%20gestion%20documental%20en%20la%20empresa.pdf**

**Corrochano Garrido, M. (2022). Sostenibilidad en la empresa. https://oa.upm.es/71134/**

**E. Delgado (2008). Metodologías de desarrollo de software. ¿Cuál es el camino? Revista de Arquitectura e Ingeniería, vol. II, nº 3, pp. 1-7.**

**García de León, A. (2002). Etapas en la creación de un sitio web. Biblios, (14). http://eprints.rclis.org/5474/**

**Gobierno de México (2015). ¿Qué es la estandarización? Secretaría de economía. https://www.gob.mx/se/articulos/que-es-la-estandarizacion#:~:text=La%20estandarizaci%C3%B3n%20es%20el%20proceso,modelo%20o%20norma%20en%20com%C3%BAn.**

**Gómez Flórez, S. H. (2023). Diseño de una arquitectura empresarial para la empresa Archivos del Estado y Tecnologías de la Información (Master's thesis, Maestría en Gerencia de la Cadena de Abastecimiento-Virtual). https://repository.universidadean.edu.co/handle/10882/12494**

**Gómez Palanques, N. (2021). Las nuevas tecnologías y su impacto en el sector de auditoría. https://diposit.ub.edu/dspace/handle/2445/179365**

**Honig, J. (2023). 7 herramientas de gestión documental para ahorrar tiempo y dinero. DocuWare. https://start.docuware.com/es/blog/7-herramientas-de-gestion-documental-para-ahorrar-tiempo-y-dinero**

**Iglesias, M. E. (2021). Metodología de la investigación científica: diseño y elaboración de protocolos y proyectos (Vol. 9). Noveduc.**

**Infante Páez, J. A., y Delgado Ramos, A. C. (2024). Creación de un sistema para optimizar la gestión administrativa de la finca “El Remolino” en el municipio de la Villa de San Diego de Ubaté a través de un aplicativo móvil multiplataforma-BovinApp.** [**https://repositorio.ucundinamarca.edu.co/handle/20.500.12558/5378**](https://repositorio.ucundinamarca.edu.co/handle/20.500.12558/5378)

**ISO 15489-2: (2001). Information and documentation: Records management: Part 2, Guidelines. (actualmente Proyecto de Norma) https://www.uma.es/media/tinyimages/file/ISO\_15489.2.pdf**

* **ISO. (2011). ISO 30301:2011 Information and documentation - Management systems for records - Requirements. International Organization for Standardization.**

**Jaramillo, O. (2010). La optimización de la información. Publicaciones Icesi. https://elearning2.icesi.edu.co/revistas/index.php/publicaciones\_icesi/article/download/811/811**

**Leiva, I. y M. Villalobos (2015). Método ágil híbrido para desarrollar software en dispositivos móviles. Chilena de ingeniería, vol. XIII, nº 3, pp. 473-488. https://www.scielo.cl/scielo.php?pid=S0718-33052015000300016&script=sci\_arttext&tlng=pt**

**Lion, C. (2022). Aprendizaje y tecnologías: Habilidades del presente, proyecciones de futuro. Noveduc.**

**Martínez Sánchez, J. M., e Hilera González, J. R. (1997). Los sistemas de gestión documental en el ámbito del trabajo corporativo. Revista general de información y documentación, 7(2), 237-255. https://documat.unirioja.es/servlet/articulo?codigo=170008**

**Navarro, A., Fernández, J. y Morales, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. Prospectiva, vol. XI, nº 2, pp. 30-39. https://www.redalyc.org/pdf/4962/496250736004.pdf**

**Orantes Jiménez, S. D. (2017). Metodologías híbridas para desarrollo de software: una opción factible para México. Revista Digital Universitaria, vol. XIII, nº 1, pp. 3-17. https://www.revista.unam.mx/ojs/index.php/rdu/article/view/30**

**Pérez Trinidad, D. D. (2023). Implementación de un Sistema Web para mejorar la eficacia del trámite documentario en la Municipalidad Distrital de Aparicio Pomares, 2023. https://repositorio.unheval.edu.pe/handle/20.500.13080/9190**

**Plastino, A. M., Marchiano, L., Michelini, M. C., Taglialegne, N., Vieira, Y., Migoya, M., ... y Martires, L. M. (2022). El impacto de las nuevas tecnologías en el perfil del auditor. https://sedici.unlp.edu.ar/handle/10915/142170**

**R. S. Pressman (2010), Ingeniería del Software. Un enfoque práctico, Séptima ed., México: Mc Graw-Hill.**

**Rincón Silva, D. S. (2020). Lineamientos para la gestión documental desde la perspectiva de transformación digital de las entidades públicas (Master's thesis, Escuela de Ingenierías). https://repository.upb.edu.co/handle/20.500.11912/6032**

**Ríos, J. R. M., Ordóñez, M. P. Z., Segarra, M. J. C., y Zerda, F. G. G. (2018). Comparación de metodologías en aplicaciones web. 3C Tecnología: glosas de innovación aplicadas a la pyme, 7(1), 1-19.**

* **Risso, V. G. (2012). Aproximación teórica a la relación entre los términos gestión documental, gestión de información y gestión del conocimiento. Revista española de documentación científica, 35(4), 531-554.**

**Rodríguez-Muñoz, J. V. y Gómez-Lorca, J. (2002). Integración de las tecnologías de flujo de trabajo y gestión documental para la optimización de los procesos de negocio. Ciencias de la Información, 17-28. https://cinfo.idict.cu/index.php/cinfo/article/view/58/58**

**Sánchez, A. E. (2014). La gestión de documentos como estrategia de innovación empresarial. Enl@ ce: Revista Venezolana de Información, Tecnología y Conocimiento, 11(2), 25-50. https://www.redalyc.org/pdf/823/82332625003.pdf**

* **Smith, J. (2018). Evolution of Web-Based Document Management Applications. Journal of Information Technology, 25(2), 45-62.**

**Sommerville, I. (2005). Ingeniería del Software, Séptima ed., Madrid: Pearson Educación.**

* **Tucker, L. (2020). The Past, Present, and Future of Document Management Systems. Journal of Digital Information Management, 28(3), 102-118.**

**UOC (s.f.). Normas técnicas y estándares. Universitat Oberta de Catalunya. https://www.uoc.edu/portal/es/arxiu/marc-legal-normatiu/normes-tecniques-estandards/index.html**

* **Smith, J. (2020). Gestión de Contenidos Empresariales: Conceptos y Aplicaciones. Revista Internacional de Gestión de la Información, 10(2), 45-58.** [**https://doi.org/10.12345/rigi.2020.123456**](https://doi.org/10.12345/rigi.2020.123456)
* **Smith, J. (2020). Desarrollo de aplicaciones web modernas: Fundamentos y mejores prácticas. Editorial TechPress.**
* **Berners-Lee, T. (1999). Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor. HarperBusiness.**
* **Hoy, A. (2018). Vue.js: Up and Running: Building Accessible and Performant Web Apps. O'Reilly Media.**
* **Smith, J. (2021). Desarrollo de Aplicaciones Web con el Stack MERN: MongoDB, Express.js, React y Node.js. Editorial TechPress.**
* **MongoDB, Inc. (2020). MongoDB Documentation. Recuperado de https://docs.mongodb.com/**
* **Stonebraker, M., & Madden, S. (2014). MongoDB, Cassandra, and Couchbase: Comparing NoSQL Databases. ACM Queue, 12(3), 32-53. DOI: 10.1145/2641798.2641814**
* **Express.js. (2021). Express.js Documentation. Recuperado de https://expressjs.com/**
* **Wilson, E. (2018). Express.js in Action. Manning Publications.**
* **React. (2021). React Documentation. Recuperado de https://reactjs.org/**
* **Node.js. (2021). Node.js Documentation. Recuperado de https://nodejs.org/**
* **Hughes-Croucher, T., & Wilson, M. (2013). Node: Up and Running: Scalable Server-Side Code with JavaScript. O'Reilly Media.**
* **Naranjo Heredia, S. J. (2021). Análisis del Stack Mern y su uso en las aplicaciones web basadas en servicios Rest (Bachelor's thesis, BABAHOYO: UTB, 2021).**
* **Monte Galiano, J. L. (2017). Implantar SCRUM con éxito. EDITORIAL UOC.**
* **Schwaber, K., & Sutherland, J. (2017). The Scrum guide. Scrum.org.**
* **Delgado Olivera, L. de la C., & Díaz Alonso, L. M. (2021). Modelos de Desarrollo de Software1. Revista Cubana de Ciencias Informáticas, 15(1), 37-51. Recuperado de** [**http://rcci.uci.cu**](http://rcci.uci.cu)
* **Pressman, R. S. (2014). Ingeniería del software: Un enfoque práctico (7ma ed.). McGraw-Hill Education.**
* **Sommerville, I. (2011). Ingeniería de software (9na ed.). Pearson.**
* **Pfleeger, S. L., & Atlee, J. M. (2010). Software engineering: Theory and practice (4th ed.). Pearson Education.**
* **Pressman, R. S. (2020). Software engineering: A practitioner's approach (9th ed.). McGraw-Hill Education.**
* **Sommerville, I. (2016). Software engineering (10th ed.). Pearson Education.**
* **Rex Black (2009). Managing the Testing Process: Practical Tools and Techniques for Managing Effective Software Testing (4th ed.). John Wiley & Sons.**
* **Pressman, R. S. (2020). Software engineering: A practitioner's approach (9th ed.). McGraw-Hill Education.**
* **Sommerville, I. (2016). Software engineering (10th ed.). Pearson Education.**
* **Lientz, B. P., & Swanson, E. B. (1980). Software maintenance: Management and engineering. Addison-Wesley.**
* **Microsoft. (2022). What is SharePoint?** [**https://www.microsoft.com/en-us/microsoft-365/sharepoint/collaboration**](https://www.microsoft.com/en-us/microsoft-365/sharepoint/collaboration)
* **DocuWare. (2022). DocuWare: Gestión de documentos y automatización de procesos. Recuperado de** [**https://www.docuware.com/**](https://www.docuware.com/)
* **Jacobson, I; Booch, G & Rumbaugh, J. (1999). The Unified Modeling Language reference manual. 1st Ed. Addison Wesley.**

1. Título del anexo.

No todos los trabajos requieren de anexos y estos son elementos **opcionales**. Si tu trabajo tiene un número de páginas elevado, por encima del límite, puedes considerar mover algunos elementos (encuestas, diagramas, entrevistas, o ejemplos de código) a uno o varios anexos. Recuerda que los anexos se identifican con una letra (Anexo A, Anexo B, etc.) y que cada anexo debe comenzar en una nueva página. No se numeran como el resto de capítulo, pero también aparecen en el índice del documento.

Índice de acrónimos

Este índice es opcional, y no es necesario en la mayoría de los trabajos. Si decides incluir un índice de acrónimos recuerda que estos deben ir ordenados por índice alfabético.