



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Facultat d'Informàtica de Barcelona



Universidad Politécnica de Cataluña (UPC)

Facultad de Informática de Barcelona (FIB) - Grado en Ingeniería Informática

Especialidad en Ingeniería del Software

Trabajo de Fin de Grado

Gestor documental para una compañía farmacéutica basado en Microsoft SharePoint



Autor: Tanvir Hossain Sultana

Director: José Luis Montes Martínez (Project manager – Raona)

Ponente: Dolors Costal Costa (Departamento ESSI – FIB)

Tutor GEP: Fernando Barrabés Naval

2021 - 2022 Q1

25 de enero de 2022

Resumen

Un gestor documental basado en las tecnologías de Microsoft puede resultar útil para la colaboración y compartición de información con los compañeros de trabajo.

El objetivo de este proyecto es la creación de un espacio documental para el departamento de *compliance* de una compañía del sector farmacéutico. Para ello, se desarrollará una plataforma web que permitirá a los responsables de diferentes áreas la publicación y distribución de la documentación oficial de la organización con el fin de notificar a los usuarios implicados su lectura obligatoria. Por tanto, deberá quedar una constancia de la lectura reglamentaria de los documentos por parte de los involucrados ante una auditoría.

Resum

Un gestor documental basat en les tecnologies de Microsoft pot ser útil per a la col·laboració i la compartició d'informació amb els companys de treball.

L'objectiu d'aquest projecte és la creació d'un espai documental per al departament de *compliance* d'una companyia del sector farmacèutic. Per això, es desenvoluparà una plataforma web que permetrà als responsables de diferents àrees la publicació i distribució de la documentació oficial de la organització per tal de notificar als usuaris implicats la seva lectura obligatòria. Per tant, caldrà una constància de la lectura reglamentària dels documents per part dels involucrats davant d'una auditoria.

Abstract

A document manager based on Microsoft technologies can be useful for collaboration and information sharing with co-workers.

The goal of this project is the creation of a documentary space for the compliance department of a company in the pharmaceutical sector. For this, a web platform will be developed that will allow those responsible for different areas to publish and distribute the official documentation of the organization in order to notify the users involved of its compulsory reading. Therefore, there must be a record of the mandatory reading of the documents by those involved before an audit.

Índice de contenido

1	Introducción.....	9
1.1	Contexto del proyecto	9
1.1.1	Descripción del problema.....	9
1.2	Objetivos.....	10
1.3	Actores implicados	11
1.4	Estado del arte.....	13
1.4.1	Análisis de mercado.....	14
1.4.2	Tabla comparativa	18
1.4.3	Conclusiones.....	19
1.5	Riesgos.....	20
1.6	Metodología	21
1.7	Herramientas de seguimiento	23
1.7.1	Notion	23
1.7.2	Microsoft Teams.....	27
1.7.3	Outlook	27
1.7.4	Plataforma SharePoint Online	28
1.7.5	Git/TFS	28
1.7.6	Visual Studio Code	29
1.8	Método de validación	29
2	Glosario.....	30
3	Planificación temporal.....	32
3.1	Descripción de las tareas	32
3.1.1	Tareas de gestión de proyecto	32
3.1.2	Tareas de desarrollo	33
3.1.3	Lanzamiento del producto.....	33
3.2	Recursos.....	34
3.3	Estimación de las tareas	35
3.3.1	Diagrama de Gantt	36
4	Gestión económica	38
4.1	Identificación y estimación de los costes directos e indirectos	38
4.1.1	Costes de personal por tarea	38
4.1.2	Costes generales.....	39

4.1.3 Costes de contingencia.....	40
4.1.4 Costes de imprevistos.....	40
4.1.5 Presupuesto.....	41
4.2 Control de gestión	41
5 Especificación de requisitos.....	42
5.1 Descripción de la documentación y los procesos de negocio.....	42
5.1.1 Documentos de área de calidad.....	42
5.1.2 Procesos de negocio	48
5.2 Proceso de obtención de requisitos	53
5.3 Requisitos funcionales	54
5.3.1 Historias de usuario	55
5.3.2 Modelo conceptual de los datos	61
5.4 Requisitos no funcionales.....	63
5.4.1 Plantilla de especificación de requisitos de Volere	64
6 Arquitectura del sistema	66
6.1 Visión global.....	66
6.2 Introducción a SharePoint	66
6.3 Front-end.....	68
6.3.1 Sitio web en SharePoint.....	68
6.3.2 Interfaz de usuario personalizado	68
6.3.3 Procesos de negocio	69
6.4 Back-end	71
6.4.1 SharePoint REST API Service.....	71
6.4.2 Base de datos de SharePoint Online	71
6.5 Diagrama de clases de diseño	75
6.6 Diagramas de secuencia	77
7 Implementación.....	86
7.1 Creación y configuración de la plataforma	86
7.1.1 Elementos de un sitio	91
7.1.2 Limitaciones técnicas.....	97
7.2 SharePoint Framework con PnP	98
7.2.1 Extensión de SPFx.....	98
7.2.2 Depuración y publicación de la extensión de SPFx	102

7.3 Power Automate.....	104
7.3.1 Flujos.....	104
8 Pruebas	109
8.1 Pruebas de flujo	109
8.2 Pruebas de aceptación del usuario	110
9 Despliegue	111
10 Aspectos legales	113
10.1 Leyes aplicables al proyecto	113
10.2 Licencias.....	113
11 Seguimiento del proyecto	114
11.1 Cambio de planificación y presupuestos.....	114
11.2 Ejecución real.....	117
11.2.1 Sprint 1	117
11.2.2 Sprint 2	119
11.2.3 Sprint 3	120
11.2.4 Sprint 4	121
12 Sostenibilidad	123
12.1 Dimensión económica	123
12.2 Dimensión social.....	124
12.3 Dimensión ambiental	124
13 Conclusiones.....	126
13.1 Competencias técnicas trabajadas y relación con la especialidad de ingeniería del software.....	126
13.2 Conclusiones personales	128
13.3 Trabajo futuro.....	128
Referencias	129

Índice de figuras

Figura 1. Microsoft SharePoint.....	15
Figura 2. Google Workspace.....	16
Figura 3. Huddle.....	17
Figura 4. Box	17
Figura 5. Confluence	18
Figura 6. Raona Engineering Process.....	21
Figura 7. Ciclo de vida de la metodología REP.....	22
Figura 8. Vista por historias de usuario	24
Figura 9. Descripción de historia de usuario	24
Figura 10. Vista por tareas.....	25
Figura 11. Vista por sprints.....	25
Figura 12. Vista por estados	25
Figura 13. Descripción de una tarea.....	26
Figura 14. Vista por bugs y su formato de descripción	26
Figura 15. Team del proyecto.....	27
Figura 16. Correo al departamento de IT	27
Figura 17. Diagrama de ramas de función de Gitflow	28
Figura 18. Diagrama de Gantt con hitos.....	37
Figura 19. Proceso completo	48
Figura 20. Diagrama de proceso de edición	49
Figura 21. Diagrama de proceso de aprobación por niveles.....	50
Figura 22. Diagrama de proceso del mediador	51
Figura 23. Diagrama de proceso de publicación	51
Figura 24. Diagrama de proceso de distribución, notificación y recordatorio.....	52
Figura 25. Diagrama de caducidad	53
Figura 26. Diagrama conceptual.....	62
Figura 27. Esquema de la arquitectura del sistema en SharePoint.....	66
Figura 28. Elementos representativos de un sitio	68
Figura 29. Componentes UI	69
Figura 30. Microsoft Power Platform	70
Figura 31. Power Automate – Flujo	70
Figura 32. SharePoint API endpoints	71
Figura 33. Topología de múltiples aplicaciones de servicio	72
Figura 34. Arquitectura de SharePoint Online en 2016	73
Figura 35. Arquitectura de SharePoint Online	73
Figura 36. Lista con elementos y columnas.....	74
Figura 37. Diagrama de clases de diseño	76
Figura 38. Diagrama de secuencia – Cargar documento.....	77
Figura 39. Diagrama de secuencia – Configurar parámetros iniciales	77
Figura 40. Diagrama de secuencia – Iniciar proceso de publicación.....	78
Figura 41. Diagrama de secuencia – getDocumentCode	78
Figura 42. Diagrama de secuencia – Controlar en elaboración	79

Figura 43. Diagrama de secuencia – getDocumentAreaRelatedSubarea	80
Figura 44. Diagrama de secuencia – Respuesta Editor / Aprobador.....	80
Figura 45. Diagrama de secuencia – Gestionar tareas de editor y aprobador.....	81
Figura 46. Diagrama de secuencia – Respuesta Mediador	81
Figura 47. Diagrama de secuencia – Gestionar tareas de mediador	82
Figura 48. Diagrama de secuencia – Controlar en validación	83
Figura 49. Diagrama de secuencia – Controlar publicado.....	84
Figura 50. Diagrama de secuencia – Recordatorio tareas / justificados.....	85
Figura 51. Diagrama de secuencia – Controlar caducidad	85
Figura 52. Portal de la organización	86
Figura 53. Tipos o plantillas de sitio	87
Figura 54. Estructura plana de sitios	87
Figura 55. Información sobre el entorno de pre-producción.....	88
Figura 56. Estructura de SharePoint Online	88
Figura 57. Biblioteca de trabajo.....	89
Figura 58. Biblioteca de documentos publicados.....	89
Figura 59. Lista para tipos de documento	90
Figura 60. Lista para tareas de aprobación	90
Figura 61. Lista para tareas de justificados	90
Figura 62. Elementos de la navegación	91
Figura 63. Configuración inicial y final de la barra de inicio rápido	91
Figura 64. Vista de un usuario habitual y vista de un administrador	92
Figura 65. Navegación: migas de pan	93
Figura 66. Contenido del sitio.....	93
Figura 67. Configuración del sitio	94
Figura 68. Columnas o propiedades de sitio	94
Figura 69. Tipos de contenido de sitio	94
Figura 70. Propiedades de un tipo de contenido de sitio	95
Figura 71. Plantilla PNT con sus propiedades correspondientes	95
Figura 72. Almacén de términos de taxonomía	95
Figura 73. Configuración avanzada de permisos.....	96
Figura 74. Herramientas de desarrollo de SPFx	98
Figura 75. Estructura de la extensión	99
Figura 76. Gulp – Task runner serve	100
Figura 77. Método render para pintar componentes de la extensión.....	102
Figura 78. Método que controla estado de un documento	102
Figura 79. Comandos para generar paquete de deployment	102
Figura 80. SPFx deployment en app catalog.....	102
Figura 81. Botones para indicar acciones de un documento	103
Figura 82. Popup para iniciar proceso de publicación	103
Figura 83. Botones para indicar acciones de una tarea	103
Figura 84. Formulario para indicar el motivo del rechazo	104
Figura 85. Opciones de un flujo.....	105
Figura 86. Flujo: On Init	105

Figura 87. Flujo: Controlar en elaboración	106
Figura 88. Creación y notificación de una tarea de edición	107
Figura 89. Flujo: Controlar caducidad de documentos	108
Figura 90. Pruebas de flujo	109
Figura 91. Comprobador de flujo	109
Figura 92. Breakpoint	110
Figura 93. Información sobre el entorno de producción	111
Figura 94. ShareGate	111
Figura 95. Migración de estructura de SharePoint	112
Figura 96. Resultado de migración	112
Figura 97. Tipologías de documentos.....	117
Figura 98. Funcionamiento general.....	118
Figura 99. Mediador en el proceso de publicación	119
Figura 100. Correo de la fase de validación	121
Figura 101. Petición de cambio del correo en las notificaciones	122

Índice de tablas

Tabla 1. Comparación de alternativas	19
Tabla 2. Riesgos y planes alternativos	21
Tabla 3. Resumen de las tareas estimadas.....	35
Tabla 4. Costes de personal.....	38
Tabla 5. Costes de recursos humanos por tarea	39
Tabla 6. Costes estimados de los recursos software.....	40
Tabla 7. Costes de los recursos hardware	40
Tabla 8. Costes de contingencia con 15%.....	40
Tabla 9. Costes de imprevistos	41
Tabla 10. Estructura de propiedades del documento: acciones correctivas	44
Tabla 11. Estructura de propiedades del documento: análisis de riesgos.....	44
Tabla 12. Estructura de propiedades del documento: control de cambios	45
Tabla 13. Estructura de propiedades del documento: documentos principales	45
Tabla 14. Estructura de propiedades del documento: instrucciones de trabajo	45
Tabla 15. Estructura de propiedades del documento: instrucción técnica	46
Tabla 16. Estructura de propiedades del documento: manuales	46
Tabla 17. Estructura de propiedades del documento: no conformidades	47
Tabla 18. Estructura de propiedades del documento: plantillas	47
Tabla 19. Estructura de propiedades del documento: procedimientos generales.....	47
Tabla 20. Estructura de propiedades del documento: procedimientos normalizados de trabajo	48
Tabla 21. Historia de usuario 1 – Propiedades rellenadas de un documento borrado .	55
Tabla 22. Historia de usuario 2 – Inicio de un proceso de publicación	56
Tabla 23. Historia de usuario 3 – Gestión de un proceso de edición	56
Tabla 24. Historia de usuario 4 – Respuesta de un editor	56

Tabla 25. Historia de usuario 5 – Gestión de un proceso de aprobación	57
Tabla 26. Historia de usuario 6 – Respuesta de un aprobador	57
Tabla 27. Historia de usuario 7 – Respuesta de un mediador.....	58
Tabla 28. Historia de usuario 8 – Publicación y distribución de un documento	58
Tabla 29. Historia de usuario 9 – Volver a iniciar el proceso de publicación	58
Tabla 30. Historia de usuario 10 – Envío de notificaciones y recordatorios	59
Tabla 31. Historia de usuario 11 – Caducidad de documentos	59
Tabla 32. Historia de usuario 12 – Control de accesos.....	59
Tabla 33. Historia de usuario 13 – Mostrar menú de navegación	60
Tabla 34. Historia de usuario 14 – Enviar sugerencias	60
Tabla 35. Historia de usuario 15 – Mostrar sugerencias	60
Tabla 36. Requisito no funcional 1 – Usabilidad y humanidad: fácil de usar	64
Tabla 37. Requisito no funcional 2 – Usabilidad y humanidad: comprensión y cortesía	64
Tabla 38. Requisito no funcional 3 – Mantenimiento y soporte: mantenimiento.....	64
Tabla 39. Requisito no funcional 4 – Mantenimiento y soporte: adaptabilidad.....	65
Tabla 40. Requisito no funcional 5 – Seguridad: acceso	65
Tabla 41. Requisito no funcional 6 – Seguridad: privacidad	65
Tabla 42. Comparación entre SharePoint Online y SQL	74
Tabla 43. Resumen final de las tareas estimadas.....	115
Tabla 44. Costes finales de recursos humanos por tarea.....	115
Tabla 45. Costes finales estimados de los recursos software	116
Tabla 46. Costes finales de los recursos hardware	116
Tabla 47. Sprint backlog 1.....	118
Tabla 48. Sprint backlog 2.....	119
Tabla 49. Sprint backlog 3.....	120
Tabla 50. Sprint backlog 4.....	121

1 Introducción

La empresa Raona Engineers S.L. es una consultora internacional, con más de 20 años de experiencia, posicionada en el desarrollo de soluciones en nuevas tecnologías Microsoft, empresa de la cuál es *partner*. El equipo de Raona se caracteriza por ser joven y creativo, y está formado por más de 150 consultores y desarrolladores expertos que colaboran con clientes líderes en todos los sectores empresariales y se distribuyen entre las distintas oficinas de Barcelona, Madrid, Andorra y Argentina [1].

El trabajo final de grado de la carrera de ingeniería informática, titulado *Gestión documental para una compañía farmacéutica basado en Microsoft SharePoint*, se enfoca en la disciplina de la ingeniería del *software*, que es impartida por la Facultad de Informática de Barcelona [2]. La investigación de este proyecto se ha realizado en la empresa Raona con el objetivo de desarrollar una plataforma web que gestione el proceso de publicación y distribución de documentos de una compañía farmacéutica.

1.1 Contexto del proyecto

Raona, que es *partner* de Microsoft, proporciona soluciones a proyectos basados en herramientas de *Microsoft*. Las soluciones más destacadas que encontramos son Intranets corporativas en O365, Custom Development, Modern Apps, Cloud Integration, Power Platform, Inteligencia Artificial, CRM, etc. Por tanto, en Raona se busca la innovación, últimas soluciones y tendencias para llevar a cabo proyectos que sorprendan e inspiren a sus clientes.

El tipo de proyecto a desarrollar es un proyecto llave en mano a cliente, concretamente es para una empresa del sector sanitario. La prioridad de esta compañía reside en la creación de un espacio documental para crear, publicar y distribuir la documentación oficial de la compañía, donde el departamento de *compliance* debe publicar los documentos y notificar a los usuarios implicados su lectura obligatoria. Por tanto, tiene que haber una constancia de la lectura o no de los documentos por parte de los implicados de forma oficial y que sea auditable al contenido de los documentos.

1.1.1 Descripción del problema

La empresa del sector hospitalario de este proyecto está constituida por los directores, técnicos, administradores, diseñadores, empleados, colaboradores externos, etc. En dicha compañía el departamento o área de *compliance* se encarga de regular los documentos normativos y jurídicos de la organización. En este departamento hay una serie de profesionales, por ejemplo: editores, aprobadores, etc. que trabajan cooperativamente para diseñar y validar los documentos legales de la empresa. Antes de publicar o enviar un documento a un destinatario es necesario que este validado por los responsables del departamento ya que el objetivo final es informar a los trabajadores de manera eficaz el cumplimiento de leyes y normativas para el correcto funcionamiento de la empresa.

Sin embargo, el actual gestor documental de la compañía no dispone de una infraestructura o un sistema de almacenamiento de cualquier documento oficial, así como de un acceso único a dicha documentación. A continuación, se describen los problemas identificados en la empresa.

- Las documentaciones oficiales de la organización se encuentran almacenadas en diversos entornos, por ejemplo: Outlook, Microsoft Teams, carpetas del servidor de ficheros, etc.
 - En algunos casos, resulta inconveniente localizar un determinado documento ya que no existe una eficiencia al buscar un tipo de contenido o documento.
 - Además, no siempre se tiene la garantía de que se esté tratando la última versión del documento.
- No existe un entorno colaborativo que permita trabajar conjuntamente a diferentes profesionales sobre un documento.
- No se aplica la definición de un proceso para realizar las tareas de manera progresiva y correcta.
 - No existe un proceso de edición, aprobación y publicación de documentos. Esto hace que no haya ninguna comunicación entre los profesionales. Ralentiza el trabajo y aumenta el tiempo de finalización de un documento de *compliance*.
 - No existe un proceso automático de envío de documentos oficiales a los usuarios, sino que se hace manualmente.
 - No existe un proceso de confirmación de lectura obligatoria de documentos tal que sea demostrable ante una auditoría.

1.2 Objetivos

El objetivo principal de este proyecto es abordar el análisis, diseño y desarrollo de una plataforma para publicar y distribuir documentos oficiales de la compañía. Por tanto, el alcance del proyecto es desarrollar un *software* que permita crear un espacio documental o un sitio web para el departamento de *compliance* y el almacenamiento de cualquier documento oficial de la compañía, así como de un acceso único a dicha documentación.

Este único punto de entrada deberá permitir localizar y consultar la documentación oficial almacenada, por parte de los empleados y colaboradores externos de la empresa, pudiéndose restringir, en caso necesario, el acceso de determinados documentos a un determinado grupo de usuarios en concreto.

El funcionamiento del gestor documental será diferente según el área en que pertenezcan los documentos. Un documento puede clasificarse en dos áreas:

- Área de calidad
- Área de dirección técnica

Un documento de calidad es aquel documento que precisa una gestión especial, con el objetivo de cumplir con las normativas legales o jurídicas. Un documento de dirección técnica, también llamado documento interno, es aquel documento que debe almacenarse para que pueda ser consultado por parte de estos dos departamentos.

En este proyecto se tratarán documentos de un área, que serán los documentos de área de calidad¹.

Para este tipo de documentos, el proyecto deberá implementar un sistema que gestione el proceso de elaboración, aprobación y publicación de dichos documentos con el objetivo de mejorar y descomponer los procesos de negocio. También deberá permitir la distribución y notificación a los usuarios implicados de manera automática la lectura obligatoria. Dicha distribución puede dirigirse:

- a una persona
- a un grupo colectivo de personas

Por tanto, deberá tener constancia de la lectura o no de los documentos por parte de los implicados. El hecho de dar el consentimiento de “leído y comprendido” de forma oficial permitirá tener un registro para presentar ante una auditoría en el caso que sea necesario.

1.3 Actores implicados

En esta sección se aluden las partes interesadas o *stakeholders* del sistema. Para cada una de ellas se explica su definición y la identificación de objetivos. Es muy importante identificar los *stakeholders* para definir de forma precisa los requisitos que ha de satisfacer el sistema ya que son los que darán el valor al *software*.

Promotora: es la jefa de la directora técnica y de calidad de la empresa del sector sanitario del proyecto, quien proporciona los recursos necesarios para llevar a cabo el proyecto.

- Rol: tomar decisiones importantes de la plataforma y validar las entregas del proyecto.
- Objetivos:
 - Mejorar la gestión de documentos.
 - Hacer consciente a los trabajadores de la lectura obligatoria de los documentos legales.
 - Tener un registro de documentos leídos y comprendidos.

Director de proyecto: es el director de la empresa Raona, Jose Luis Montes Martínez, encargado de guiar y supervisar el desarrollo del proyecto.

- Rol: hacer un seguimiento, dirigir y supervisar el trabajo del desarrollador.
- Objetivo:

¹ Los detalles de un documento de calidad se explican en el apartado [5.1](#)

- Supervisar el desarrollo para que se alcancen los requisitos definidos del sistema.

Tutora ponente: es la profesora de la universidad, Dolors Costal Costa, que se encarga de dirigir la parte académica del proyecto. Por ejemplo, revisar el contenido de la memoria del TFG, verificar el estado del proyecto, etc.

- Rol: hacer un seguimiento de la memoria.
- Objetivo:
 - Verificar que la estructura de la memoria sea correcta.
 - Verificar que el contenido del informe sea entendible y que el trabajo realizado quede reflejado correctamente en el documento.
 - Corregir los errores pertinentes de la memoria.

Desarrollador: es el encargado de configurar y desarrollar la plataforma web del proyecto teniendo en cuenta los requerimientos del sistema. El proyecto consta de un solo desarrollador, que es Tanvir Hossain Sultana, para crear el sistema *software* de este proyecto.

- Rol: realizar tareas funcionales y desarrollar el sistema.
- Objetivos:
 - Cumplir con las entregas y requisitos del sistema.
 - Desarrollar una plataforma de calidad que se adecue a las expectativas del cliente.
 - Adquirir conocimientos tecnológicos.
 - Elaborar y redactar el trabajo final de grado.

Usuarios: son las personas que hacen el uso de la aplicación. Existen cuatro tipos de clientes:

Editores: son las personas encargadas de crear y editar el contenido de los documentos.

- Rol: redactar documentos.
- Objetivos:
 - Elaborar documentos legales y jurídicos explicativos.
 - Colaborar con otros editores.

Aprobadores: son aquellos que se encargan de aprobar los contenidos escritos por parte de los editores.

- Rol: validar o rechazar los documentos escritos por parte de los editores.
- Objetivos:
 - Revisar contenido.
 - Verificar que no haya errores en el documento.

Mediadores: son aquellas personas que intervienen, en el caso que sea necesario, para gestionar las peticiones o incidencias recibidas por parte de los editores y aprobadores, ya sea de rechazo o para mejorar el contenido del documento.

- Rol: gestionar las tareas de editores y aprobadores.
- Objetivo:
 - Tomar una decisión o acción acerca de la petición.

Trabajadores: son aquellas personas que trabajan en la empresa del sector hospitalario de este proyecto.

- Rol: leer y dar el consentimiento de lectura o no de los documentos.
- Objetivos:
 - Entender el contenido del documento.

1.4 Estado del arte

El estado de arte de este proyecto se basa en una investigación sobre las diferentes formas de desarrollar una plataforma web basada en las tecnologías modernas.

Actualmente, existen varios caminos para la creación de un sitio web o *website*. En primer lugar, normalmente las páginas web se implementan mediante el uso de diferentes tecnologías que aplican los conceptos básicos de un sitio web. Fundamentalmente HTML, CSS y JavaScript para un sitio web estático y ASP, Web Server, PHP o NodeJS, que son algunas de las tecnologías más conocidas, para un sitio web dinámico. Sin embargo, esta forma de desarrollar una página web requiere de un conocimiento técnico de lenguajes de programación para escribir muchas líneas de código y un mantenimiento exhaustivo para gestionar los recursos del sitio web [3].

En segundo lugar, se puede crear un sitio web desde cero usando los constructores de sitios web (*website builders*). En los últimos años estos han triunfado bastante en la industria ya que elimina el concepto de programación y se basa en la filosofía *WYSIWYG* que representa “What you see is what you get” (Lo que ves es lo que obtienes). Es decir, los *website builders* proporcionan una interfaz intuitiva y la funcionalidad de *drag-and-drop* para crear un nuevo sitio web. Por tanto, solamente proporcionan las herramientas para diseñar la página web y no están involucradas en absoluto con el alojamiento web físico. Los *website builders* más destacados son: Wix, Weebly y Shopify [4][5][6].

En tercer lugar, existe una ruta alternativa para crear una plataforma de sitios web, se puede usar los sistemas de gestión de contenidos o gestores de contenidos o denominado *CMS* (siglas que resumen el concepto *Content Management System*) *platform*. Esta plataforma es un *software online* que permite crear, administrar y gestionar un sitio web y sus contenidos. Al usar un CMS, se puede aprovechar los temas prediseñados. Normalmente, los desarrolladores más experimentados prefieren usar una plataforma CMS que *website builders* ya que proporcionan una mayor flexibilidad y escalabilidad en la personalización del diseño web. Los CMS más populares son WordPress, Drupal y Moodle [3][7].

Por último, las propuestas mencionadas anteriormente no son las únicas formas de construir un sitio web sino existe una plataforma alternativa, DMS (siglas que resumen

el concepto *Document Management System*), es una variante de CMS que engloba las funcionalidades del *website builders* y CMS, y proporciona unas nuevas. Tanto DMS como CMS son similares en que ayudan en la gestión de información, ofrecen el almacenamiento centralizado de datos, facilitan la creación, retención y distribución de información. También ofrecen una mayor seguridad de la información, admiten la colaboración basada en contenido y proporcionan flujos de trabajo automatizados. Sin embargo, estos dos sistemas se diferencian en una serie de razones [8][9][10]:

- Procesamiento de datos: DMS está diseñado para trabajar con datos contenidos en documentos estructurados (Word, Excel, PDF, PowerPoint, etc.) y CMS tiene la capacidad de administrar información electrónica tanto estructurada como no estructurada y otro tipo de activos digitales (vídeo, audio, archivos multimedia, Flash, etc.)
- Propósito: los objetivos principales de un DMS son la gestión del flujo de trabajo y el cumplimiento normativo, y de un CMS son el almacenamiento, la recuperación y la publicación de contenido.
- El DMS incorpora capacidades avanzadas con escaneo e imágenes que no podemos encontrar en la mayoría de los CMS.

En este proyecto se hará uso de la plataforma ECM (siglas que resumen el concepto *Enterprise Content Management*) para desarrollar una plataforma web que tenga la capacidad almacenar y gestionar la documentación del proceso de una organización y otros contenidos. Tanto DMS como CMS son un subconjunto de ECM ya que este es capaz de manejar las informaciones digitales y documentos vía web, por tanto, un ECM puede usarse como un CMS o un DMS o una combinación de dos [11].

En los siguientes apartados, se describe el estado en que se encuentra la plataforma ECM y qué tecnologías la desarrolla. Posteriormente, se explican los motivos por los que se ha elegido una de ellas para llevar a cabo el proyecto.

1.4.1 Análisis de mercado

En el mercado actual existen varias tecnologías o herramientas para gestionar los contenidos o documentos empresariales. A continuación, se analizan una serie de herramientas más populares y usables de la actualidad.

1.4.1.1 Microsoft SharePoint

SharePoint es una herramienta diseñada por Microsoft para la gestión documental y el trabajo en equipo. Está formada por una serie de productos y elementos de *software* que incluye funciones de colaboración, módulos de administración de procesos, módulos de búsqueda y una plataforma de administración de documentos [12][13].

SharePoint se puede usar como CMS para que forme parte de un sitio de SharePoint y/o como DMS para almacenar y manejar documentos.

SharePoint ofrece las siguientes características [14]:

- Paneles de control de usuario

- Gestión y edición de documentos
- Herramientas de publicación web
- Gestión de contenido
- Gestión de cumplimiento
- Redes sociales empresariales (a través de la herramienta Yammer)
- Flujo de trabajo automatizados (a través de la herramienta Power Automate)
- Integración de inteligencia empresarial (a través de la herramienta Power BI)

En SharePoint el almacenamiento de datos se puede alojar de tres maneras [14]:

- SharePoint Server: es la plataforma de servidor para compartir, colaborar y gestionar el contenido. Sus servicios solo pueden ser accesibles por usuarios internos con la licencia apropiada.
- SharePoint Online: es la plataforma o servicio basado en la nube a SharePoint; disponible a través de Microsoft Office 365 que ofrece tecnologías de seguridad mejoradas y se puede acceder a él a través de internet.
- SharePoint híbrido: sincroniza datos entre el servidor local y SharePoint Online con OneDrive.

SharePoint está destinado a organizaciones que necesitan almacenamiento y administración de documentos, además de una forma de colaborar en proyectos sin estar en la misma oficina. Por lo general, lo utilizan las organizaciones medianas y grandes para aprovechar los recursos técnicos y características que ofrece SharePoint.

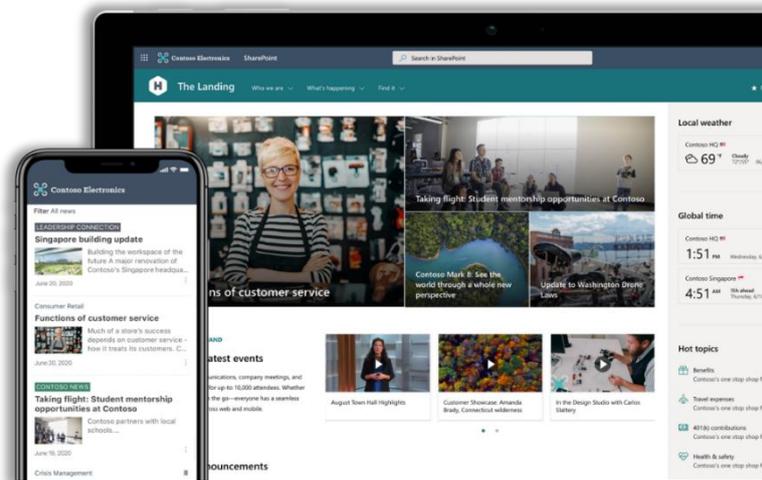


Figura 1. Microsoft SharePoint | Fuente: [12]

1.4.1.2 Google workspace or G Suite

Google Workspace es un producto de Google que aporta un enfoque similar a SharePoint, que es la creación de un entorno colaborativo integrado para las empresas. Por ejemplo, la integración de Google Meets con Docs para realizar actividades grupales. El ecosistema de Google ofrece las siguientes capacidades o características [14]:

- Colaboración y almacenamiento de documentos (a través de la herramienta Google Drive)

- Flujos de trabajo no automatizado (con Formularios y Hojas de cálculo) y automatizado (con AO Docs)
- Sitios de equipo o Intranet *sites* (con Google *sites*)
- Aplicaciones de colaboración como calendarios y contactos compartidos.
- Reuniones o videoconferencias (con Google Meet)

Tanto Google workspace como Microsoft SharePoint proporcionan la filosofía de “todo en uno”. Es decir, combinar múltiples ideas en una. Por tanto, se integran con otras herramientas.



Figura 2. Google Workspace | Fuente: [14]

1.4.1.3 Huddle

Huddle es una plataforma de gestión de proyectos y colaboración de contenido basado en la nube, aunque se centra en la gestión de documentos y el intercambio de archivos. Las funcionalidades de Huddle se concentran en ayudar a las empresas a compartir archivos, colaborar en proyectos y organizar sus flujos de trabajo. Los usuarios pueden editar archivos directamente dentro de la aplicación y disfrutar de un control de versiones completo, similar a lo que encontraría en un documento de Google. Las características más importantes son [14][15]:

- Panel de contenido inteligente
- Administración de tareas, colaboración y control automático de versiones
- Uso compartido inteligente y colaboración con seguimiento de versiones y visualización de marcas de tiempo

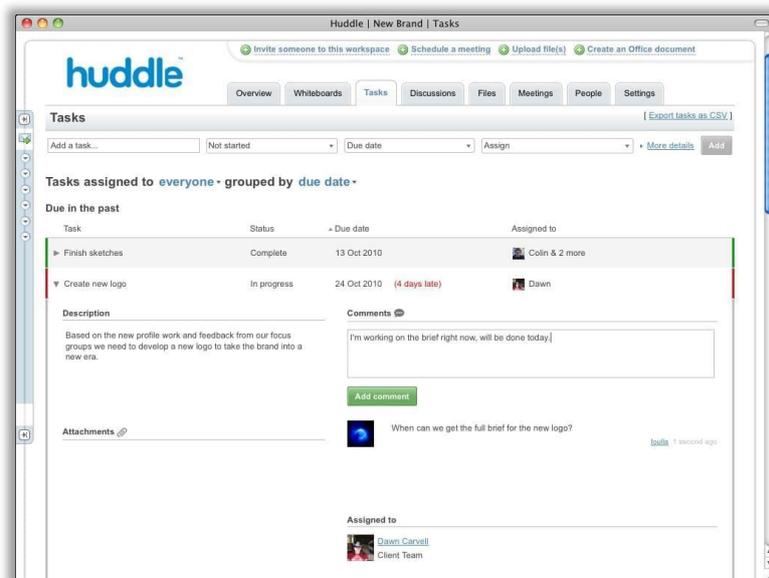


Figura 3. Huddle | Fuente: [14]

1.4.1.4 Box

Box es una plataforma de colaboración, flujos de trabajo y gestión de contenido seguros. Incorpora el servicio de almacenamiento en la nube, seguridad empresarial, sincronización móvil, control de versiones y herramientas de gestión de proyectos. Box no proporciona una edición de documentos de forma nativa, sino que se integra con Microsoft Office 365 y otras aplicaciones de terceros. Las características importantes de esta herramienta son [14][16]:

- Gestión de documento, colaboración y marca personalizada
- Flujos de trabajo (con la herramienta Box Relay)
- Controles de seguridad móvil
- Gestión de proyectos y tareas
- Vínculos para compartir protegidos con contraseña
- Acceso a estadísticas e informes
- Integración con diferentes tecnologías

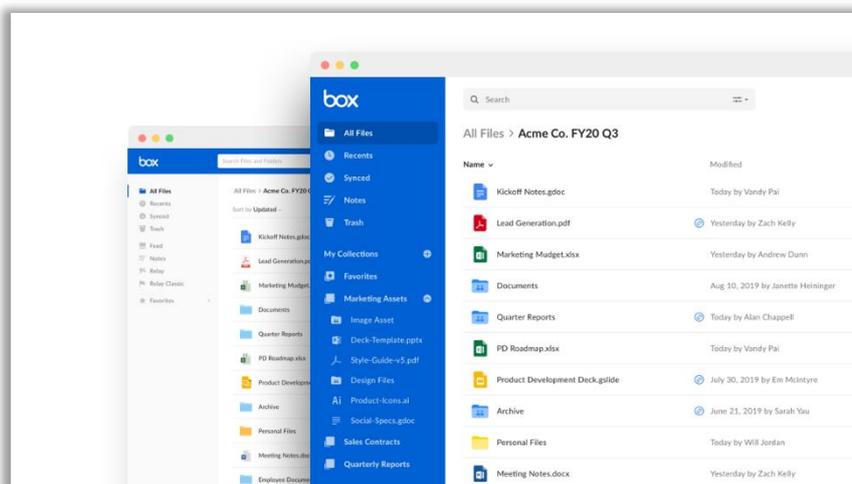


Figura 4. Box | Fuente: [16]

1.4.1.5 Confluence

Confluence es una herramienta diseñada por Atlassian para la gestión de proyectos, desarrollo de *software* y trabajo en equipo. Permite crear espacios únicos para que cada uno de los equipos pueda almacenar el trabajo y colaborar en tiempo real. Cada espacio contiene varias páginas, que utilizan plantillas intuitivas para capturar notas de reuniones, requisitos y diseños de proyectos y otros activos. Las características más interesantes de esta plataforma son [14][17]:

- Colaboración de documentos
- Administración de tareas y flujos de trabajo de desarrollo ágiles
- Sitios y espacios de equipo

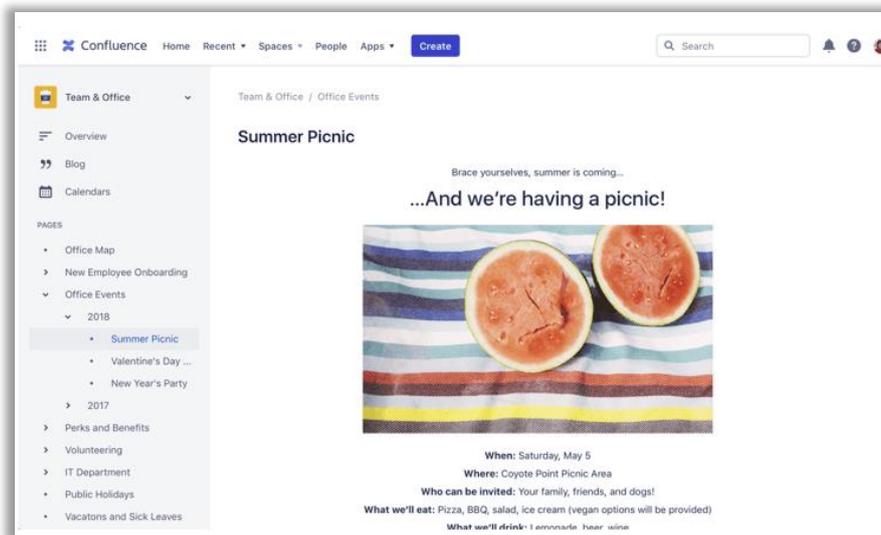


Figura 5. Confluence | Fuente: [17]

1.4.2 Tabla comparativa

En la siguiente tabla se comparan las diferentes alternativas mencionadas anteriormente según las características que proporcionan y que son relevantes para este proyecto. Se puede observar que existen varias herramientas en el mercado que proporcionan funcionalidades similares, diferenciándose solamente en pequeños detalles.

Plataformas / características	MS SharePoint	Google workspace	Huddle	Box	Confluence
Gestión documental y trabajo en equipo					
Flujo de trabajo automatizados					
Gestión de versiones					
Gestión de metadatos					
Intranet empresarial y sitios					
Customización de sitios (UI y UX)					
Redes sociales empresariales					
Servicio de <i>cloud storage</i>					
Integración con tecnologías					

Adaptación a cualquier dispositivo					
Tarifas de licencia					

Tabla 1. Comparación de alternativas | Fuente: Elaboración propia

Leyenda: Provee la característica No provee la característica

1.4.3 Conclusiones

Confluence es una herramienta que no es conveniente para desarrollar una plataforma web que gestione los procesos de documentos ya que se inclina en gran medida hacia la gestión de proyectos de desarrollo de *software*. No permite gestionar los flujos de trabajos para controlar las etapas por las que puede pasar un documento, ocurre lo mismo con Huddle, y con Box sería necesario usar herramientas de terceros para crear un sitio web.

Entre **Microsoft SharePoint** y Google Workspace, que son productos bastantes similares, se elige la primera tecnología para crear una plataforma web orientada al cliente con el propósito de almacenar y gestionar los documentos de este proyecto. Como ya se ha explicado anteriormente, Raona es *partner* de Microsoft y propone soluciones corporativas basadas en las tecnologías de este. Además, el cliente de este proyecto tiene una licencia *premium* de Microsoft Office 365. Por tanto, SharePoint es la opción idónea. Es evidente que la gestión de documentos de SharePoint es uno de los mejores sistemas de administración de documentos disponible actualmente y encaja perfectamente por la política de empresa [18].

Cabe mencionar que SharePoint es una herramienta que evoluciona constantemente y se puede categorizar en dos versiones [19][20]:

- **SharePoint On-premise:** es una solución local de SharePoint para aquellas organizaciones que necesitan operar en un servidor local de la empresa ya sea para leer o alojar contenido de una plataforma. Por tanto, se necesita un equipo de IT para mantener el servidor, para instalar actualizaciones y parches de seguridad. Además, se tiene que alojar o aplicar soluciones de granja (*Farm Solutions*) para que sean compatibles con Microsoft. Las soluciones locales más destacables son: SharePoint Server 2019, SharePoint Server 2016 y SharePoint Server 2013. Aunque el futuro de SharePoint On-premise se está moviendo hacia una solución basada en la nube.
- **SharePoint Online:** es la versión en la nube de Microsoft y se distribuye a través de Office 365. Es un servicio SaaS (Software as a Service). Eso significa que se reducen los recursos internos (personas y *hardware*) y no hay que preocuparse por la infraestructura local. Sin embargo, se ha de pagar una cuota mensualmente a Microsoft ya que este realiza las siguientes funciones:
 - Mantiene, aplica parches de seguridad e instala actualizaciones de plataforma.
 - Se encarga de alojar *farm solutions* en la infraestructura en la nube de Microsoft.
 - Se compromete a una disponibilidad del 99,9%.

En este proyecto se usará la versión más reciente de SharePoint, que es SharePoint Online. Por tanto, a partir de ahora cuando se hable de SharePoint se referirá a la versión Online (SharePoint Online). La implementación de SharePoint se hará en una red de intranet para almacenar, gestionar y trabajar simultáneamente en los documentos, y el mantenimiento necesario será realizado por Microsoft.

1.5 Riesgos

En este apartado se mencionan los posibles riesgos que pueden surgir durante el proceso de desarrollo. Dichos riesgos pueden impedir a conseguir los objetivos definidos y alterar la planificación temporal del proyecto. Para ello, es muy importante conocer de antemano estos posibles riesgos, tener un plan de prevención y afrontarlos con un plan de mitigación. A continuación, en la tabla 2 se detalla la gestión de riesgos de este proyecto.

Riesgo: Incumplimiento con las fechas de entrega		Probabilidad: baja
Descripción	El estudiante entrega el trabajo final de grado con retraso o no cumple con las fechas de entrega con el cliente.	
Impacto	La defensa del TFG se realizaría en un cuatrimestre posterior. Además, esta situación puede generar el malestar entre el director y autor de este proyecto, también con el cliente.	
Indicadores	Dedicar más horas de trabajo para cumplir la planificación de cada semana puede ser un gran indicador de este riesgo. Es decir, no poder terminar a tiempo las historias de usuario indicadas de cada <i>sprint</i> . Otro indicador puede ser la inversión de tiempo en el aprendizaje de las diferentes tecnologías utilizadas en el proyecto ya que el desarrollador no tiene experiencia sobre dichas tecnologías (SharePoint, Power automate y PnP).	
Estrategias de prevención	Es primordial planificar y gestionar las horas de trabajo. Además, desde un principio se debe priorizar las funcionalidades más importantes del sistema.	
Planes alternativos	Se puede retrasar o cambiar las fechas de entregas de cada iteración, o introducir más horas de las estimadas en la planificación de trabajo.	
Riesgo: Incorporar cambios		Probabilidad: media
Descripción	Durante el proyecto el cliente puede solicitar modificaciones, eliminaciones o agregaciones a las funcionalidades o requisitos que acordaron al inicio del proyecto.	
Impacto	Las fechas de entrega pueden cambiar. Y por parte del autor será necesario dedicar más horas de trabajo al proyecto.	
Indicadores	El cliente no está satisfecho con las funciones implementadas o con las entregas de cada iteración ya que no se adecuan con sus expectativas. Por tanto, comunica o informa al director del proyecto sobre los cambios.	
Estrategias de prevención	Aplicar la metodología ágil para solventar y adaptar los cambios al sistema. Es decir, el desarrollador debe comunicarse tanto con el director como el cliente, en el caso que sea necesario, para ir validando las funcionalidades de la aplicación.	
Planes alternativos	Adaptar el sistema con los cambios incorporados, pero siempre manteniendo las expectativas del cliente.	
Riesgo: Funcionamiento incorrecto de la aplicación		Probabilidad: alta
Descripción	La aplicación puede tener errores durante el proceso de desarrollo o despegamiento que impiden a los usuarios usar las funcionalidades correctamente.	

Impacto	Los usuarios no podrán realizar sus tareas ni podrán acceder a sus recursos.
Indicadores	Las funcionalidades implementadas no funcionan correctamente o no siguen la lógica del programa.
Estrategias de prevención	Es fundamental hacer las pruebas o <i>testing</i> de cada funcionalidad. Además, será útil hacer el uso de las herramientas o extensiones que facilitan el diagnóstico de errores de código, por ejemplo: <i>Error Lens</i> [21].
Planes alternativos	Comprobar la lógica o funcionalidad del programa. Usar la herramienta <i>debugger</i> .

Tabla 2. Riesgos y planes alternativos | Fuente: Elaboración propia

1.6 Metodología

El TFG de este proyecto requiere el desarrollo de una plataforma web que engloba el uso de diferentes tecnologías, por ejemplo: SharePoint [22], Power automate [23] y programación PnP [24]. Además, este proyecto tiene un tiempo limitado, por tanto, aplicar una metodología de trabajo de manera correcta será esencial para poder cumplir los objetivos del proyecto.

La metodología de trabajo a seguir es una metodología utilizada en la empresa Raona, que es **Raona Engineering Process (REP)**, orientada a proyectos de los clientes. REP es la especificación corporativa del proceso de desarrollo de un aplicativo o solución por fases: cada proyecto se desglosa en fases, pasos y actividades que deben ser implementadas por el equipo de proyecto para garantizar la calidad del marco común a los clientes (figura 6). Este proceso mantiene la certificación ISO 15504 y cubre las necesidades básicas de *SPICE (Software Process Improvement and Capacities)* [25].

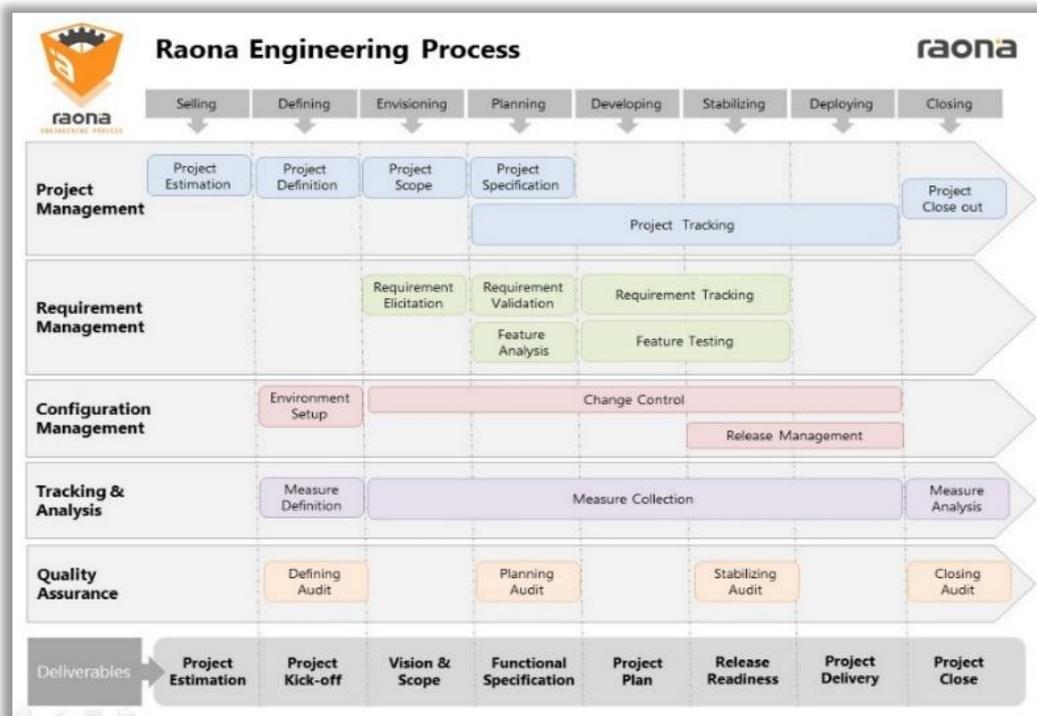


Figura 6. Raona Engineering Process | Fuente: Raona - Metodología REP

REP se basa en un modelo que forma parte de la metodología *Agile* [26], en concreto *Scrum* [27]. Usa el modelo de *sprint* y la planificación iterativa. Cada ciclo de desarrollo

llamado iteración o *sprint* tiene marcado uno o varios objetivos que conlleva a producir un incremento en las funciones del producto para conseguir el producto final. El periodo de un *sprint* no debe superar más de 60 días y en este proyecto se definen iteraciones cortas de una duración entre 2 y 4 semanas. Por tanto, el desarrollo basado en *sprints* permitirá minimizar el riesgo ya que las funcionalidades del producto se revisan en cada ciclo de iteración.

En resumen, la metodología REP o *Scrum* permite tener un desarrollo ágil donde el cliente gestiona la evolución del producto con reuniones cortas de seguimiento y el equipo de desarrollo revisa el trabajo realizado desde la reunión anterior y el previsto para la siguiente (figura 7).



Figura 7. Ciclo de vida de la metodología REP | Fuente: Raona - Metodología REP

Para empezar a aplicar la metodología REP en el proyecto se ha de generar el *product backlog* [28] inicial, que es un artefacto que contiene todos los requisitos funcionales, no funcionales e historias de usuario [28] de la aplicación. En esta lista de actividades hay que priorizar las funcionalidades que aportan más valor al *software* y que puede evolucionar y cambiar a lo largo del proyecto. Posteriormente, en cada ciclo de iteración o *sprint* se debe realizar las siguientes tareas:

- **Planificación:** se hace al inicio de cada *sprint* con el objetivo de definir el *sprint backlog* y los objetivos a cumplir. Es decir, se escogen las funcionalidades pertinentes del *product backlog* y se desglosan en pequeñas tareas para realizar o implementar en el *sprint* en cuestión.
- **Seguimiento:** se hace una breve reunión diariamente, también conocida *Daily stand-up meeting*, entre el equipo de desarrollo y el *project manager* para revisar el progreso diario del trabajo asignado. En esta reunión se suele responder las siguientes preguntas tradicionales:
 - ¿Qué hiciste ayer?

- ¿Has tenido algún problema o impedimento en realizar las tareas asignadas?
- ¿Qué harás hoy?
- **Análisis:** se hace una revisión o una *demo*, si es posible con el cliente, al final de cada *sprint* para corregir los posibles errores y desviaciones. Se verifica la ejecución de las funcionalidades y se comprueba el logro de los objetivos definidos del *sprint*.

1.7 Herramientas de seguimiento

1.7.1 Notion

Se usará la herramienta Notion [29] para gestionar el **product backlog** del proyecto. Notion es una herramienta flexible que permite tener una gran personalización para gestionar el *product backlog* con distintos *sprints*, métricas de la evolución del proyecto, etc. En Raona para el seguimiento del proyecto se usa la herramienta Task planner o Azure DevOps [30] ya que Raona trabaja con las tecnologías de *Microsoft*. Sin embargo, el autor de este proyecto ha acordado con el project manager el uso de la herramienta Notion debido a la facilidad o sencillez de uso y a la claridad visual de *task boards*.

El *product backlog* de este proyecto está dividido en varias vistas [31]:

- **por historias de usuario:** se muestra cada historia de usuario con su propia descripción, estado, duración, tareas que conlleva, etc.
- **por sprints:** se visualiza la estructura de las tareas de cada *sprint*.
- **por tareas:** se muestra la descripción y la prioridad de cada tarea.
- **por estados:** se muestra el estado en que se encuentran las historias de usuarios y tareas. Cada una puede estar en uno de los siguientes estados:
 - Not started
 - In progress
 - To verify
 - Complete
- **por bugs:** se muestra la descripción y solución del *bug*

Product backlog

▲ Historias de usuario (HU): funcionalidad cuya implementación aporta valor al cliente.
 ↓ Sprints: las iteraciones marcan un período de tiempo limitado para completar una serie de tareas.
 ◀ Tareas: son las acciones que componen las historias de usuarios.
 ● Bugs: son tareas que tienen errores y deben ser arregladas o solucionadas.

¡ Hacer click en la vista de Product backlog para ver diferentes vistas, por ejemplo, la vista de Historias de usuarios, sprints, tareas, bugs, etc. Podemos ordenar las tareas por estado, por prioridad, etc.

Historias de usuario ▼

All Projects	Sprint asig...	Depend...	Tasks
<ul style="list-style-type: none"> ▲ HU1: Propiedades rellenas de un documento borrador ▲ HU2: Inicio de un proceso de publicación ▲ HU3: Gestión de un proceso de edición ▲ HU4: Respuesta de un editor 	<ul style="list-style-type: none"> Sprint 1 Sprint 2 Sprint 2 Sprint 2 	<ul style="list-style-type: none"> HU1 HU2 HU3 	<ul style="list-style-type: none"> Crear entorno pre-producción Crear una biblioteca en un site para almacenar documentos de trabajo Crear columnas y tipos de contenido de sitio Agregar las columnas y tipos de contenido de sitio en la biblioteca SG [Flujo] Crear un flujo para controlar el estado en elaboración del documento [PnP] Agregar un botón para cambiar el estado de un documento y programar la lógica para controlar el estado del documento Crear una lista en el site para controlar tareas de publicación Hacer un sketch o un esquema de todos los procesos [Investigar] Cómo podemos desglosar personas de un grupo de AD [Flujo] Crear tareas de editor [Flujo] Gestionar respuesta de un editor [PnP] Crear un proyecto con la extensión de SP [PnP] Adaptar código base al proyecto [PnP] Añadir botones Up&Down para una tarea de editor [PnP] Desplegar el código en

Figura 8. Vista por historias de usuario | Fuente: Notion - Product backlog [31]

Open as page Share

HU4: Respuesta de un editor

- Type: HU ▲
- Status: Complete 🟢
- Sprint asignada: Sprint 2
- Prioridad: Media
- Riesgo: Baja
- Dependencia: HU3
- Horas dedicadas: 15
- Tasks:
 - [Flujo] Gestionar respuesta de un editor
 - [PnP] Crear un proyecto con la extensión de SP
 - [PnP] Adaptar código base al proyecto
 - [PnP] Añadir botones Up&Down para una tarea de editor
 - [PnP] Desplegar el código en la plataforma SP
- Bugs:
 - [PnP] El componente IconButton no renderiza
- + Add a property
- 👤 Add a comment...

Historia de usuario

Usuario: editor

Prioridad de negocio: media

Riesgo o complejidad en desarrollo: baja

Descripción

💡 Como editor autenticado en el sistema, quiero realizar mi tarea asignada para cumplir mi rol.

Criterios de aceptación (confirmación)

- Debe existir una tarea de edición para el editor en cuestión.

Figura 9. Descripción de historia de usuario | Fuente: Notion - Product backlog [31]

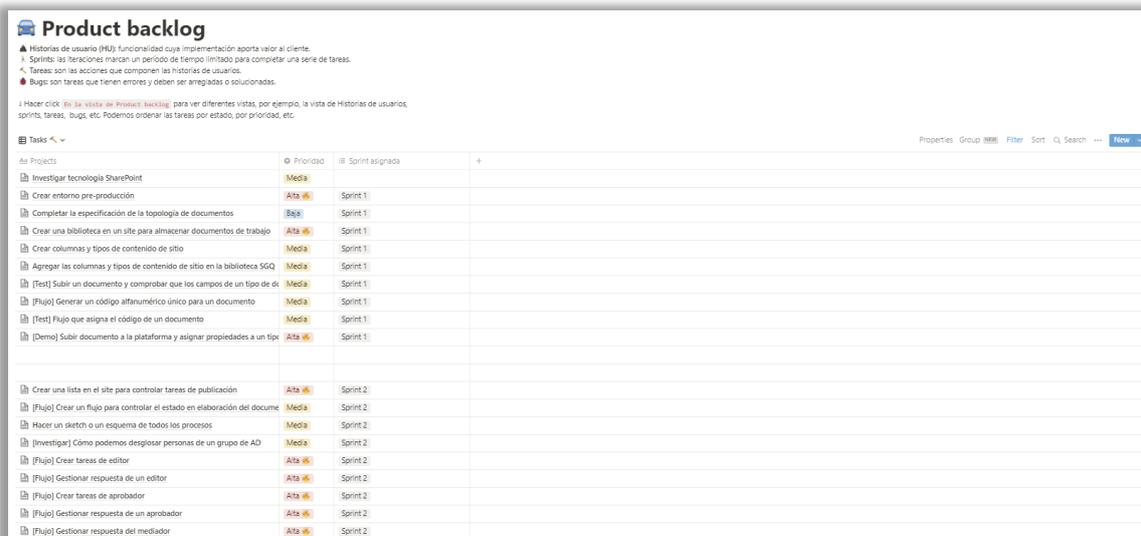


Figura 10. Vista por tareas | Fuente: Notion - Product backlog [31]

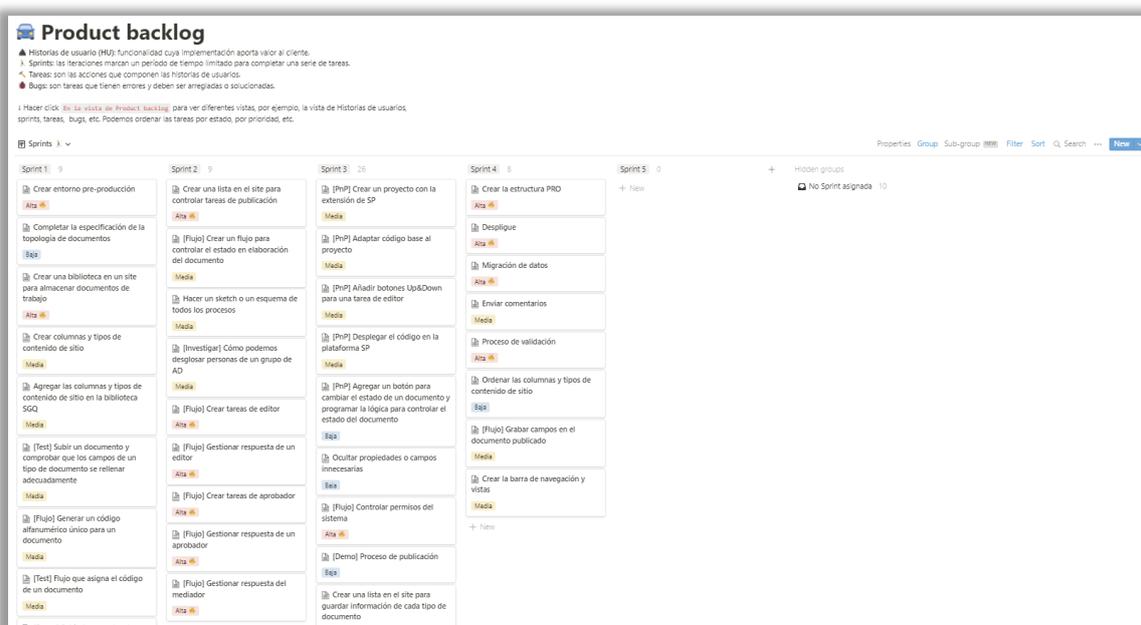


Figura 11. Vista por sprints | Fuente: Notion - Product backlog [31]

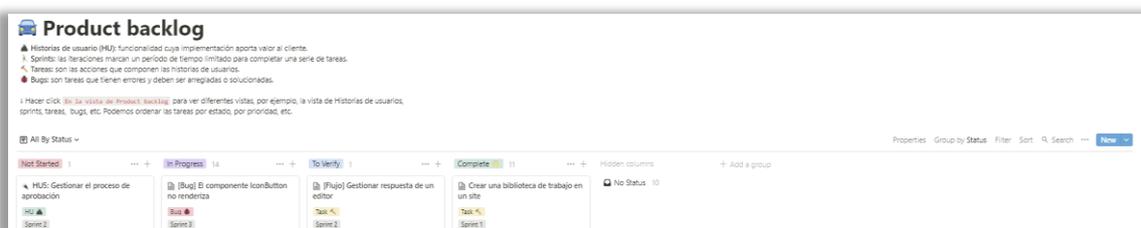


Figura 12. Vista por estados | Fuente: Notion - Product backlog [31]

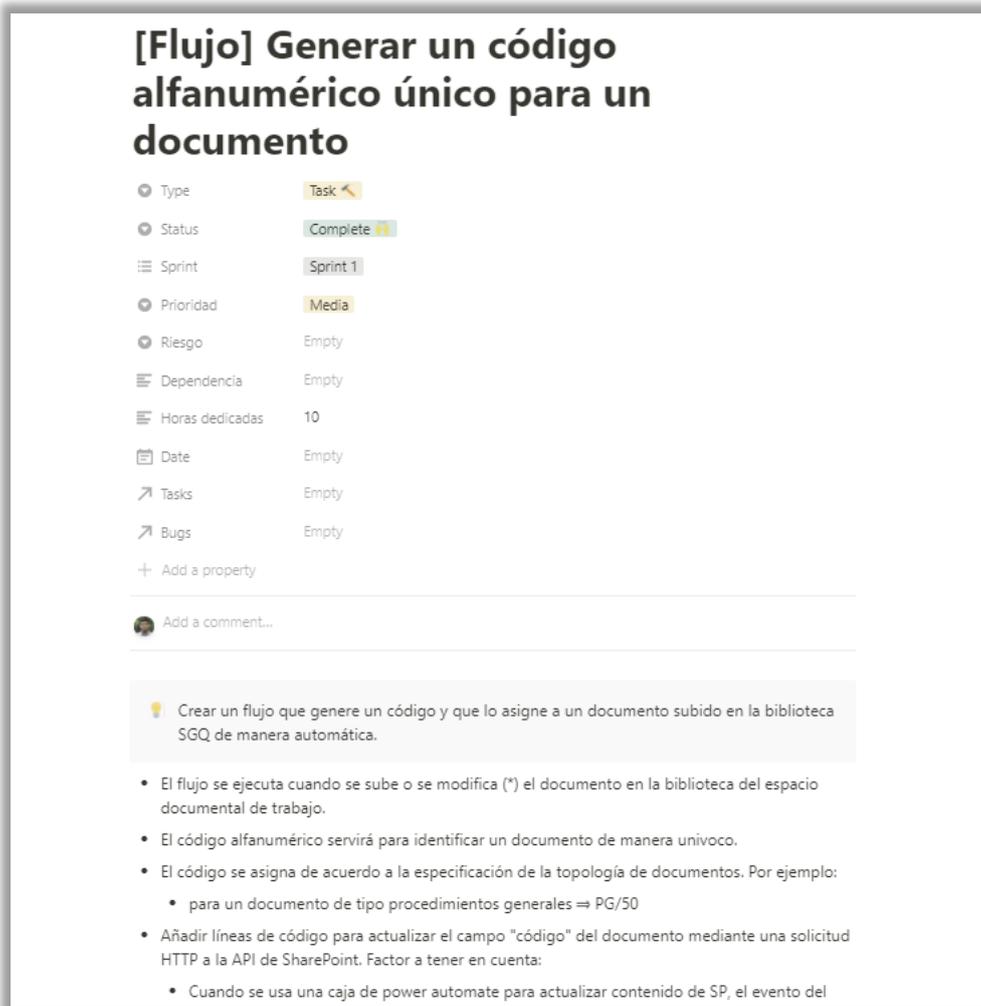


Figura 13. Descripción de una tarea | Fuente: Notion - Product backlog [31]

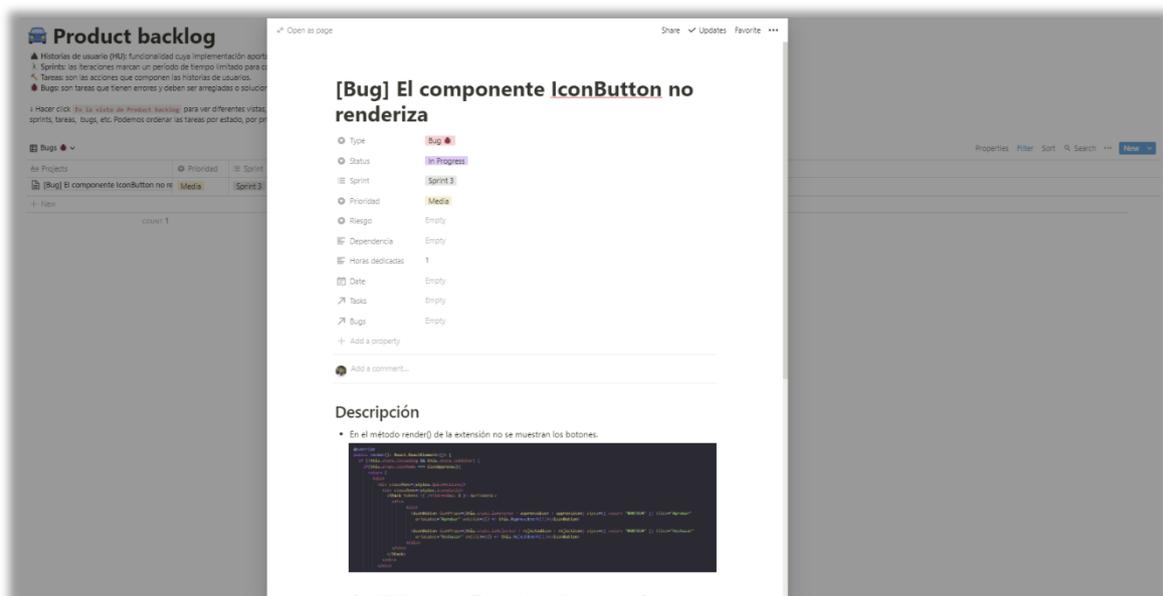


Figura 14. Vista por bugs y su formato de descripción | Fuente: Notion - Product backlog [31]

1.7.2 Microsoft Teams

Se usará la herramienta Teams para comunicarse o para gestionar reuniones con el director, cliente y compañeros del equipo en el caso que sea necesario. También para publicar documentos pertinentes del proyecto en el grupo o *team* del trabajo.

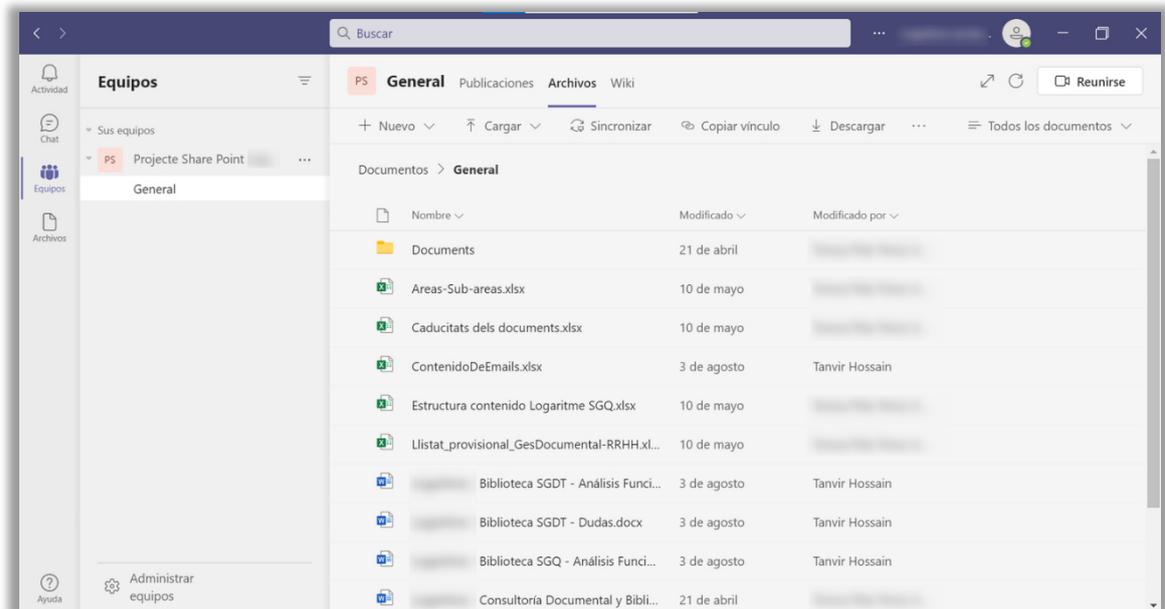


Figura 15. Team del proyecto | Fuente: Microsoft Teams

1.7.3 Outlook

Se usará la herramienta Outlook de *Microsoft* para comunicarse formalmente a través del correo electrónico con el cliente o con el departamento de IT de la empresa de este proyecto, por ejemplo: para pedir acceso a un servicio, para crear usuario o grupo en el sistema, etc. También para comunicarse e informar a los interlocutores (project manager, cliente u otros) de las reuniones convocadas.

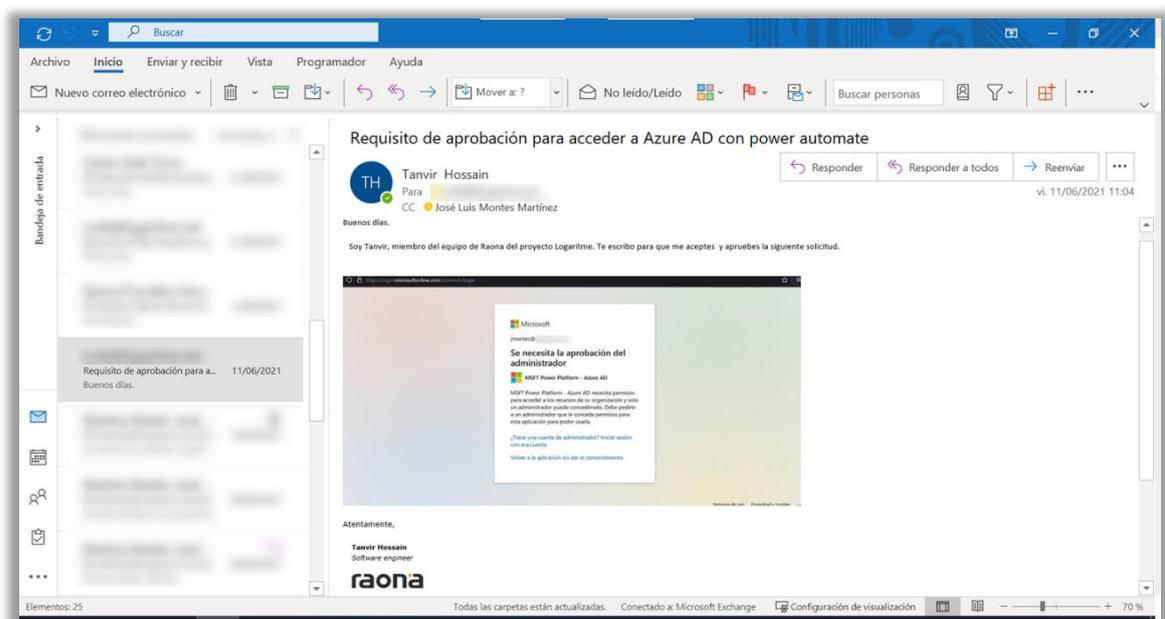


Figura 16. Correo al departamento de IT | Fuente: Outlook

1.7.4 Plataforma SharePoint Online

El proyecto está pensado para desarrollar una página web con la tecnología SharePoint. Mediante el SharePoint Online [22] se puede acceder a la plataforma web de SharePoint para crear y configurar la intranet del sistema.

Notad que la plataforma SharePoint integra varias tecnologías y entre ellas está la tecnología Power automate, que se usará para programar la lógica del sistema.

Para desarrollar la plataforma de este proyecto de una manera segura y correcta se definirá dos niveles de entorno de desarrollo [32]:

- Entorno de **pre-producción**: entorno en el que se desarrollan y se revisan las funcionalidades de la aplicación web.
- Entorno de **producción**: entorno en el que se despliega la aplicación para ser usada por los usuarios finales.

1.7.5 Git/TFS

Para gestionar el control de cambio de código, concretamente código PnP [24], se usará la herramienta **Git**. Además, se usará la herramienta **Team Foundation Server**, también conocido como Azure DevOps Server para alojar el código en un repositorio o servidor externo con el objetivo de gestionar y garantizar la disponibilidad del código fuente del proyecto. Se creará un único repositorio para gestionar el código *front-end* de la plataforma SharePoint ya que el código PnP se puede inyectar en esta plataforma.

Para trabajar con Git de manera correcta se usará la metodología **Gitflow** [33]. Se dividirá el flujo de desarrollo en varias ramas:

- **Master**: es la rama principal para alojar el código estable y funcional.
- **Development**: es la rama secundaria que contiene el código de cada *sprint*.
- **Feature branches**: cada tarea o historia de usuario se implementará en una rama distinta a partir de la rama development, con el nombre *feature/task-or-user-story-name*. De esta manera, se consigue aislar las distintas tareas y se evita que se produzcan conflictos entre ellas. Si hay conflictos entre tareas entonces deberán ser resueltos cuando se haga *merge* con la rama development.

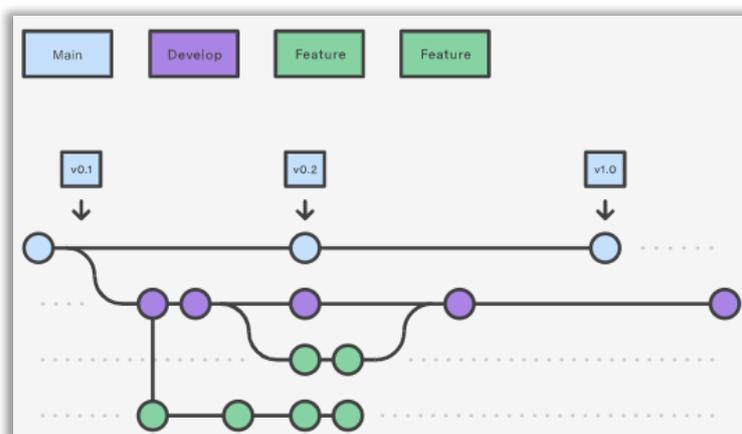


Figura 17. Diagrama de ramas de función de Gitflow | Fuente: Atlassian [33]

1.7.6 Visual Studio Code

Para el desarrollo de código PnP [24], que involucra el uso del *framework* React y el lenguaje de programación Typescript, se usará la herramienta Visual Studio Code. Esta *tool* es fácil de usar, ayuda gestionar de manera eficiente el *scaffolding* de un proyecto e incorpora varias extensiones para facilitar la programación. También es importante seguir las buenas prácticas de programación o reglas de calidad de código [34]. En este proyecto para asegurar la legibilidad y homogeneidad del código se usará una serie de directrices:

- Uso de **PascalCase** para nombres de las clases.
- Uso de **camelCase** para nombres de las variables, propiedades y métodos.
- Los nombres de las clases, así como los nombres de las propiedades, métodos, comentarios deben estar **en inglés**.

1.8 Método de validación

La validación de las funcionalidades de la aplicación se llevará a cabo mediante las reuniones de validación de cada semana o *sprint* con el director del proyecto. Así mismo, se hará una demostración *user acceptance testing* [35] con el cliente para valorar si se han alcanzado los objetivos definidos de la iteración. Para ello se realizarán las pruebas de integración de forma manual con el objetivo de comprobar la ejecución del programa y asegurar el correcto funcionamiento de la aplicación.

2 Glosario

API: Application Programming Interfaces

CSS: Cascading Style Sheets

Back-end: es la capa de acceso a datos de una aplicación, que no es accesible directamente por los usuarios, capaz de gestionar la lógica del programa que maneja dichos datos.

Breakpoint: es un punto de interrupción o pausa intencional en un programa, implementado con fines de depuración.

Debugger: es una herramienta que es capaz de detectar y corregir los errores de un programa informático.

Diseño responsive: es aquel componente o contenido que es capaz de adaptarse a pantallas de diferentes tamaños.

Drag-and-drop: es una técnica que facilita la interacción con un programa o aplicación de manera intuitiva para llevar elementos de un lugar a otro.

Front-end: es la parte visual de una aplicación a la que un usuario puede navegar e interactuar directamente.

Framework: es una herramienta o marco de trabajo que proporciona funcionalidades genéricas y repetitivas para el desarrollo y/o la implementación de una aplicación.

Grupo AD: Grupo Active Directory

Hardware: es un conjunto de componentes físicos de un sistema informático.

HTML: HyperText Markup Language

Intranet (corporativo): es un espacio privado y centralizado que permite recibir y compartir información.

IT: Information Technology

JS: JavaScript

Parche (de seguridad): es una actualización de seguridad de un sistema que intenta corregir los errores de la seguridad.

PDF: Portable Document Format

Premium: se refiere a un servicio que tiene funcionalidades exclusivas y diferenciadas. Por tanto, un usuario premium de un servicio tendrá acceso a estas características.

RNF: Requisito No Funcional

Scaffolding: es la estructura de un proyecto.

Software: es un programa o conjunto de programas que hacen posible la ejecución de diferentes tareas de un sistema informático.

UI: User Interface

UX: User eXperience

UML: Unified Modeling Language

3 Planificación temporal

El trabajo final de grado consta de 18 créditos (ECTS), 3 de los cuales son destinados al curso de gestión de proyecto y los otros 15 son del proyecto propiamente. Es decir, se requiere una dedicación de 540 horas, considerando que cada crédito equivale a 30h de trabajo [36].

La duración estimada de este proyecto es de 3 meses aproximadamente. Es decir, la fecha de inicio es el 12 de julio de 2021 y se contempla su finalización el 15 de octubre del mismo año. Los días posteriores se destinarán a las entregas de informes académicos y la preparación de la defensa ante el tribunal. Sin embargo, teniendo en cuenta que el turno de lectura del TFG es la última semana de enero de 2022, se considera que habrá un margen para posibles desviaciones del proyecto.

En resumen, teniendo en cuenta que habrá una dedicación de 8h diarias por parte del autor de este proyecto, de lunes a viernes excepto los días festivos laborales, en el caso que sea necesario se dedicará los fines de semanas para mejorar o pulir contenidos de la documentación. Se calcula que en total habrá una dedicación de $80 \text{ días} * 8 \frac{h}{\text{día}} = 640 \text{ h}$ aproximadamente.

3.1 Descripción de las tareas

Las descripciones de las tareas de este proyecto se dividen en tres grupos: gestión de proyecto, desarrollo y lanzamiento del producto. El hecho de describir las tareas como una unidad de trabajo conlleva especificar los recursos, roles y las horas de trabajo a dedicar. De esta manera, se consigue planificar, llevar un seguimiento y maximizar la productividad del proyecto.

3.1.1 Tareas de gestión de proyecto

Las tareas de este grupo hacen referencias a aquellas tareas que definen las bases del proyecto para tener una visión clara del proyecto. Es decir, se definen los marcos de trabajo y los objetivos del proyecto. Este conjunto también incluye las tareas de documentación, planificación temporal, gestión económica, etc. para tener un control óptimo del proyecto.

- **Alcance:** se analiza el contexto y los problemas actuales del proyecto. Se propone una solución alternativa para perfeccionar la situación actual, para ello, se definen los objetivos, los requerimientos básicos y el alcance del proyecto para fijar las bases de este. También, se definen las metodologías y herramientas de trabajo, riesgos, etc.
- **Planificación:** en esta tarea se estiman las horas de trabajo y se grafica la planificación temporal del proyecto en el diagrama de *Gantt* [37].
- **Presupuesto:** se estiman los cálculos de los presupuestos y costes del proyecto.
- **Sostenibilidad:** se analiza el impacto ambiental, social y económico que puede producir la idea del proyecto.

- **Estudios tecnológicos:** esta tarea consiste en familiarizarse con las tecnologías que se van a usar en este proyecto, concretamente Microsoft SharePoint, Power Automate y Programación PnP. Para ello se consultarán las documentaciones y videos tutoriales encontrados en internet.
- **Documentación:** a lo largo de este proyecto se irá elaborando diferentes tipos de documentos para explicar y reflejar el trabajo realizado por parte del autor de este proyecto. El objetivo final es agrupar los contenidos de estos en una única documentación, que será la memoria del TFG. Será una tarea repetitiva ya que se deberá incorporar las posibles mejoras y corregir los posibles errores de la documentación.
- **Defensa:** esta tarea consiste en la preparación y ensayo de la presentación del proyecto con el objetivo de explicar y demostrar ante un tribunal el funcionamiento del producto final de este proyecto.

3.1.2 Tareas de desarrollo

Las tareas de este grupo están relacionadas con la especificación, el diseño, la implementación y las pruebas de las funcionalidades del proyecto. Según la metodología definida en el apartado [1.6](#), la fase de desarrollo debe dividirse en una serie de iteraciones. Por tanto, se definen 4 sprints para el proyecto completo. Cada una de ella debe seguir la siguiente estructura:

- **Planning meeting:** se planifica o documenta las *features* o historias de usuario, se desglosan en tareas concretas y se introducen al backlog (Notion [31]).
- **Daily meeting:** son reuniones rápidas y diarias para explicar el progreso de la aplicación y concretar los impedimentos existentes.
- **Diseño, implementación y testing:** se implementan y se verifican el funcionamiento de las tareas asignadas de una iteración en cuestión.
- **Review meeting:** se hace una revisión de las tareas implementadas con el director del proyecto.
- **Retrospective meeting:** se hace una revisión con el cliente para validar las funcionalidades implementadas y obtener *feedback*.

3.1.3 Lanzamiento del producto

Las tareas de este grupo hacen referencia a la implantación del producto final en el entorno del cliente. En esta fase de cierre se realizarán las siguientes tareas:

- **Preparación del entorno del cliente (T13):** se debe construir la estructura del entorno de producción en base a la pre-producción.
- **Despliegue (T14):** se despliegan las soluciones implementadas al entorno del cliente.

3.2 Recursos

En esta sección se detallan los recursos necesarios para garantizar el desarrollo de este proyecto de manera correcta y óptima. Estos se pueden dividir en dos grupos:

- **Recursos humanos:** los roles involucrados de este proyecto de ingeniería de *software* serán los siguientes:
 - Project Manager (PM): se encarga de dirigir el proyecto, en este caso será el director del TFG.
 - Analista de software (AS): se encarga de detectar los problemas, extraer los requisitos, calcular costes, etc.
 - Desarrollador de software (DS): se encarga de desarrollar y resolver problemas o errores del sistema.
 - Tester (T): se encarga de verificar y validar las funcionalidades del sistema.
 - Documentador (D): se redactan las bases, los actos o cambios que van sucediendo durante el proceso de desarrollo de *software*.

Notad que el autor de este proyecto se encargará de ejercer los roles AS, DS, T y D.

- **Recursos materiales:** los recursos materiales se diferencian entre hardware y software. Los recursos hardware (RHW) imprescindibles que se usarán frecuentemente son: una silla, una mesa, un monitor HP 27fw, un portátil HP ZBook de la empresa, un teclado Tecknet, un ratón Logitech M185, un iPad y auriculares con micrófono. En cuanto a los recursos de *software* (RSW), se necesitará:
 - Herramientas de Microsoft Office 365 (Teams, Outlook, Word, etc.) (MO): para la comunicación y documentación del proyecto.
 - Notion (N): para apuntar notas, gestionar las tareas e historias de usuario.
 - Lucidchart (L): para crear diagramas, diseñar flujos, etc.
 - SharePoint (SP): servicio online de Microsoft para configurar la plataforma web del proyecto.
 - Power Automate (PA): para programar flujos.
 - Visual Studio Code (VSC): IDE para programar con el *framework* React.
 - Git/TFS (G): para control de código y versiones.

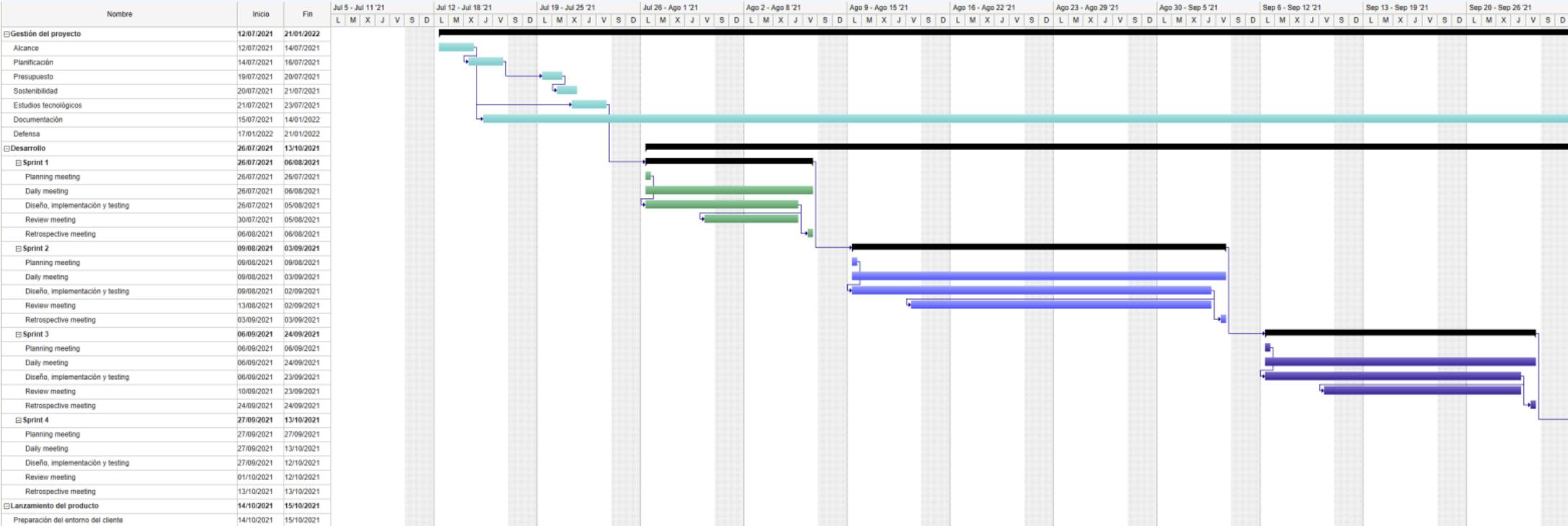
3.3 Estimación de las tareas

En la siguiente tabla, se muestra un resumen de las tareas estimadas de cada grupo o fase. Para cada una de ellas se especifica su duración en horas, dependencias con otras tareas, los recursos materiales y humanos necesarios para ejecutar la tarea.

Código	Nombre	Tiempo estimado	Dependencias	Recursos	Roles
GP	Gestión del proyecto	200h	–	–	–
GP1	Alcance	20h	–	RHW, MO	PM, AS
GP2	Planificación	20h	GP1	RHW, MO, L	PM, AS
GP3	Presupuesto	10h	GP2	RHW, MO	PM, AS
GP4	Sostenibilidad	10h	GP3	RHW, MO	PM, AS
GP5	Estudios tecnológicos	20h	GP1	RHW, N	DS
GP6	Documentación	100h	GP1	RHW, MO, L	D
GP7	Defensa	20h	GP6	RHW, MO, SP	AS, DS, T
DS	Desarrollo	424h	–	–	–
SP1	Sprint 1	70h	GP5	–	–
SP11	Planning meeting	2h	GP5	RHW, MO	PM, DS
SP12	Daily meeting	5h	GP5	RHW, SP, PA, VSC, GIT	PM, DS
SP13	Diseño, implementación y testing	58h	SP11	RHW, SP, PA, VSC, GIT	DS, T
SP14	Review meeting	2h	SP13	RHW, SP, PA, VSC, GIT	PM, DS
SP15	Retrospective meeting	3h	SP13	RHW, SP	PM, DS
SP2	Sprint 2	159h	SP1	–	–
SP21	Planning meeting	2h	SP1	RHW, MO	PM, DS
SP22	Daily meeting	10h	SP1	RHW, SP, PA, VSC, GIT	PM, DS
SP23	Diseño, implementación y testing	140h	SP21	RHW, SP, PA, VSC, GIT	DS, T
SP24	Review meeting	4h	SP23	RHW, SP, PA, VSC, GIT	PM, DS
SP25	Retrospective meeting	3h	SP23	RHW, SP	PM, DS
SP3	Sprint 3	105h	SP2	–	–
SP31	Planning meeting	2h	SP2	RHW, MO	PM, DS
SP32	Daily meeting	7h	SP2	RHW, SP, PA, VSC, GIT	PM, DS
SP33	Diseño, implementación y testing	90h	SP31	RHW, SP, PA, VSC, GIT	DS, T
SP34	Review meeting	3h	SP33	RHW, SP, PA, VSC, GIT	PM, DS
SP35	Retrospective meeting	3h	SP33	RHW, SP	PM, DS
SP4	Sprint 4	90h	SP3	–	–
SP41	Planning meeting	2h	SP3	RHW, MO	PM, DS
SP42	Daily meeting	6h	SP3	RHW, SP, PA, VSC, GIT	PM, DS
SP43	Diseño, implementación y testing	76h	SP41	RHW, SP, PA, VSC, GIT	DS, T
SP44	Review meeting	3h	SP43	RHW, SP, PA, VSC, GIT	PM, DS
SP45	Retrospective meeting	3h	SP43	RHW, SP	PM, DS
LP	Lanzamiento del producto	16h	SP4	–	–
LP1	Preparación del entorno del cliente	10h	SP4	RM, SP, PA	DS
LP2	Despliegue	6h	LP1	RM, SP	DS
	Total	640h	–	–	–

Tabla 3. Resumen de las tareas estimadas | Fuente: Elaboración propia

3.3.1 Diagrama de Gantt



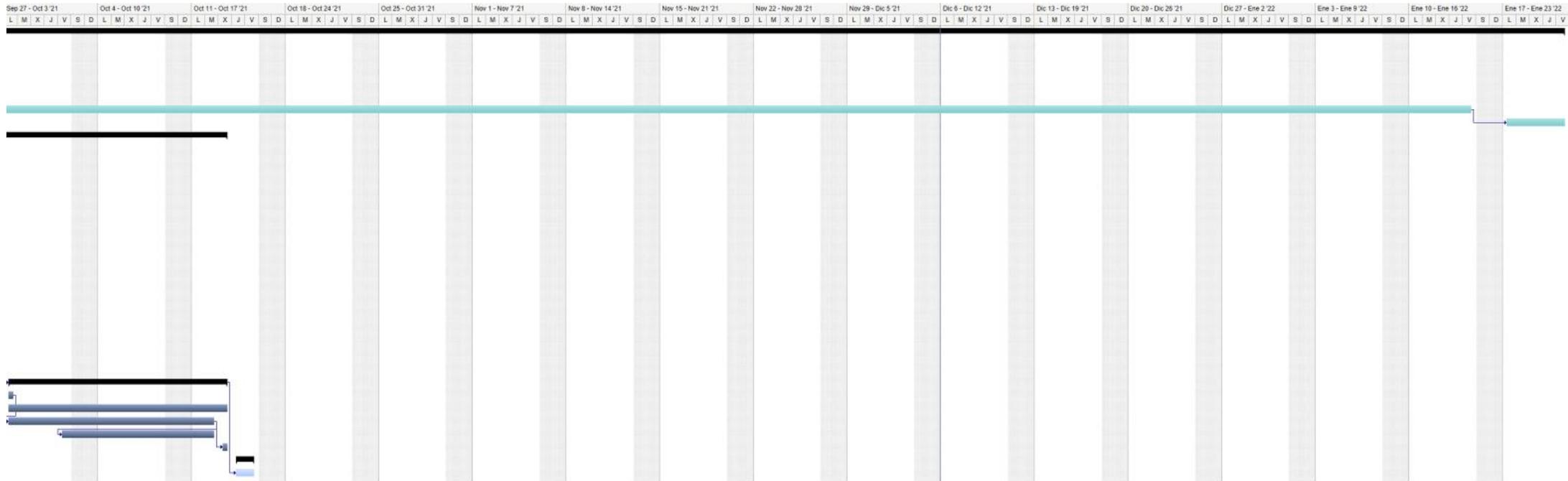


Figura 18. Diagrama de Gantt con hitos | Fuente: Elaboración propia

4 Gestión económica

4.1 Identificación y estimación de los costes directos e indirectos

En este apartado se detalla la estimación de los costes del proyecto basado en la planificación temporal del capítulo anterior. La gestión de costes está asociada a los costes del personal, costes generales (costes de recursos de *hardware*, *software* y amortizaciones), costes de contingencia para tener un margen de los costes de desarrollo y costes de posibles imprevistos.

4.1.1 Costes de personal por tarea

En esta sección se especifican los costes asociados a los recursos humanos definidos en el capítulo anterior. En la tabla 4, se puede observar el coste laboral bruto y neto por hora de cada rol, que viene determinado por la empresa en la que se ha realizado el proyecto.

Rol	Coste por hora	Coste por hora y SS ²
Project manager (PM)	55 €	71,4 €
Analista de software (AS)	45 €	58,5 €
Desarrollador de software (DS)	35 €	45,5 €
Tester (T)	30 €	39 €
Documentador (D)	15 €	19,5 €

Tabla 4. Costes de personal | Fuente: Elaboración propia con datos de Raona

En la tabla 5, se muestran los costes de personal por cada actividad. El cálculo de coste de cada tarea es la suma de costes de cada rol por la hora invertida y el factor multiplicativo de la seguridad social.

Código	PM	AS	DS	T	D	Coste
GP	60h	64h	35h	1h	100h	8 935€
GP1	20h	20h	–	–	–	2 000€
GP2	20h	20h	–	–	–	2 000€
GP3	10h	10h	–	–	–	1 000€
GP4	10h	10h	–	–	–	1 000€
GP5	–	–	20h	–	–	700€
GP6	–	–	–	–	100h	1 500€
GP7	–	4h	15h	1h	–	735€
DS	60h	0h	370h	54h	0h	17 870€
SP1	12h	0h	62h	8h	0h	3 070€
SP11	2h	–	2h	–	–	180€
SP12	5h	–	5h	–	–	450€
SP13	–	–	50h	8h	–	1 990€
SP14	2h	–	2h	–	–	180€
SP15	3h	–	3h	–	–	270€
SP2	19h	0h	134h	25h	0h	6 485€
SP21	2h	–	2h	–	–	180€

² Se aplica el incremento de coste de Seguridad Social al coste por hora, con un factor multiplicativo de 1,3.

SP22	10h	–	10h	–	–	900€
SP23	–	–	115h	25h	–	4 775€
SP24	4h	–	4h	–	–	360€
SP25	3h	–	3h	–	–	270€
SP3	15h	0h	95h	10h	0h	4 450€
SP31	2h	–	2h	–	–	180€
SP32	7h	–	7h	–	–	630€
SP33	–	–	80h	10h	–	3 100€
SP34	3h	–	3h	–	–	270€
SP35	3h	–	3h	–	–	270€
SP4	14h	0h	79h	11h	0h	3 865€
SP41	2h	–	2h	–	–	180€
SP42	6h	–	6h	–	–	540€
SP43	–	–	65h	11h	–	2 605€
SP44	3h	–	3h	–	–	270€
SP45	3h	–	3h	–	–	270€
LP	0h	0h	16h	0h	0h	560€
LP1	–	–	10	–	–	350€
LP2	–	v	6	–	–	210€
Total (h)	120h	64h	421h	55h	100h	–
Total (€)	6 600 €	2 880€	14 735€	1 650€	1 500€	27 365€
Total + SS (€)	8 580€	3 744€	19 155,5€	2 145€	1 950€	35 574,5€

Tabla 5. Costes de recursos humanos por tarea | Fuente: Elaboración propia

4.1.2 Costes generales

Los costes generales son aquellos que hacen referencias a los costes de recursos materiales, tanto software como hardware, añadiendo el concepto de amortización.

En la tabla 6, se muestran los costes amortizados de recursos *software* utilizados en este proyecto. Las herramientas de Microsoft Office 365 (Teams, Outlook, Word, etc.) son proporcionadas por la empresa Raona y está inscrita a una suscripción *Office 356 E3* con un precio de 19,70€ por usuario al mes [38]. Y el *software* SharePoint y Power Automate para desarrollar la plataforma, serán proporcionados por parte del cliente, cuenta con una suscripción *Office 356 E3* y una licencia de Power Automate con un precio de 12,60€ por usuario al mes [39]. Las herramientas restantes, Notion, Lucidchart, Visual Studio Code y Git/TFS, no requieren un coste adicional ya que son gratuitas.

En general, las amortizaciones se calculan mediante la fórmula de amortización lineal: *coste software * años de vida útil*, donde la Agencia Tributaria dicta que el periodo de años máximo debe ser 6 años para sistemas y programas informáticos [40]. Sin embargo, los cálculos de amortización de este proyecto se pueden considerar que los productos obsoletos de *software* se renuevan cada dos años ya que la Hacienda permite hacer los cambios cada dos años, y las licencias mensuales o anuales también las permiten. Por tanto, la fórmula de amortización es:

$$\frac{\text{coste software anual (€)}}{\text{vida útil(años) * días laborables/año * dedicación en horas / día}} * \text{duración proyecto (h)},$$

donde los días laborables en Raona son 230, una dedicación de 8h diaria y una duración de proyecto de 80 días * $8 \frac{h}{\text{día}} = 640 h$.

Software	Coste anual (€)	Vida útil (años)	Amortización (€)
Microsoft Office 365	236,4	2	41,11
SharePoint	236,4		41,11
Power Automate	151,2		26,30
Notion	Gratuito		–
Lucidchart	Gratuito		–
Visual Studio Code	Gratuito		–
Git/TFS	Gratuito		–
Total	624	–	108,52

Tabla 6. Costes estimados de los recursos software | Fuente: Elaboración propia

Los procedimientos de cálculos de recursos materiales son los mismos que antes, pero se ha de tener en cuenta que el coste es fijo y la vida útil es de 4 años. En la siguiente tabla se puede observar el coste hardware aproximado de este proyecto.

Hardware	Coste (€)	Vida útil (años)	Amortización (€)
Portátil HP ZBook	1 896	4	164,86
Monitor HP 27fw	250		21,74
Teclado Tecknet	30		2,61
Ratón Logitech M185	15		1,30
Ipad 6.ª generación	400		34,78
Auriculares con micrófono	10		0,87
Total	2 601		–

Tabla 7. Costes de los recursos hardware | Fuente: Elaboración propia

$$\begin{aligned} \text{costes genéricos} &= \text{coste software} + \text{coste hardware} + \text{amortizaciones} \\ &= 624 + 2 601 + (108,52 + 226,17) = 3 559,69\text{€} \end{aligned}$$

4.1.3 Costes de contingencia

Los costes de contingencia son aquellos costes que cubren los posibles imprevistos del proyecto. Para ello, se establece un porcentaje específico del 15% para calcular los costes de contingencia.

Tipos de coste	Coste (€)	Contingencia (€)
Personal	35 574,5	5 336,175
Generales	3 559,69	533,95
Total	39 134,19	5 870,125

Tabla 8. Costes de contingencia con 15% | Fuente: Elaboración propia

4.1.4 Costes de imprevistos

En la tabla 9, se puede observar los costes relacionados a los posibles imprevistos que se pueden producir a lo largo de este proyecto. Dichos valores dependen de la probabilidad asociada a los sucesos. Algunos de los cuales se definieron en el apartado gestión de riesgos. En el caso de los recursos materiales, existe la posibilidad de que alguno de los recursos *hardware* dejen de funcionar.

Concepto	Solución	Coste (€)	Riesgo (%)	Coste imprevisto (€)
Retrasos en las entregas	Dedicación 20h PM y 30h DS	2 150	10	215
Incorporar cambios	Dedicación 40h DS	1 400	25	350
Bugs	Dedicación 20h T	600	50	300
Fallos en recursos <i>hardware</i>	Sustituir por un <i>hardware</i> nuevo	2 601	5	130
Total	-	-	-	995

Tabla 9. Costes de imprevistos | Fuente: Elaboración propia

4.1.5 Presupuesto

El presupuesto aproximado para llevar a cabo este proyecto es la suma de los costes de personal, generales, contingencias e imprevistos.

$$\begin{aligned} \text{presupuesto} &= \text{personal} + \text{general} + \text{contingencia} + \text{imprevisto} \\ &= 35\,574,5 + 3\,559,69 + 5\,870,125 + 995 = 45\,999\text{€} \end{aligned}$$

4.2 Control de gestión

Para garantizar y controlar el presupuesto del proyecto se debe realizar un control semanal en las reuniones de seguimiento para valorar las horas de trabajo y detectar las posibles desviaciones respecto las horas reales estimadas. Para ello, se usarán las siguientes métricas o indicadores:

$$\text{Desviación de coste personal por tarea} = (\text{coste estimado} - \text{coste real}) * \text{horas reales}$$

$$\text{Desviación de coste en horas por tarea} = (\text{horas estimadas} - \text{horas reales}) * \text{coste estimado}$$

$$\text{Desviación de costes de contingencia} = \text{coste contingencia estimado} - \text{coste contingencia real}$$

$$\text{Desviación de costes imprevistos} = \text{coste imprevisto estimado} - \text{coste imprevisto real}$$

$$\text{Desviación total de horas} = \text{horas totales estimadas} - \text{horas totales reales}$$

$$\text{Desviación total de costes} = \text{coste total estimado} - \text{coste total real}$$

De esta forma se podrá controlar y ajustar las posibles desviaciones tal que exista la posibilidad de acotar o modificar la planificación inicial y/o el presupuesto del proyecto.

5 Especificación de requisitos

El capítulo cinco se concentra en el proceso de análisis de requisitos. En primer lugar, se hará una breve introducción de las documentaciones que se van a tratar en este proyecto y los procesos de negocios asociadas a estas ya que son el núcleo de la plataforma web del sistema. Posteriormente, se explicará el proceso de obtención de requisitos. Finalmente, se detallarán los requisitos funcionales, las historias de usuario, el modelo conceptual de datos y los requisitos no funcionales de la aplicación.

5.1 Descripción de la documentación y los procesos de negocio

En esta sección se hará una breve introducción de los documentos que se van a tratar en este proyecto y se explicarán los procesos de negocio que están implicados en dichos documentos.

5.1.1 Documentos de área de calidad

Como ya se había comentado anteriormente, el proyecto contempla aquellos documentos que pertenecen al departamento de calidad o *compliance*. Los documentos de calidad son aquellos documentos que precisan una gestión especial con el objetivo de cumplir las normativas jurídicas o legales por parte de los trabajadores.

Un documento de calidad debe seguir una topología o estructura de documentos. En los siguientes subapartados se detallan los conceptos.

5.1.1.1 Tipos de documentos

En la biblioteca de trabajo de documentos de calidad se almacenarán los siguientes tipos de documentos:

- **Acciones correctivas**
- **Análisis de riesgos**
- **Control de cambios**
- **Documentos principales**
- **Instrucciones de trabajo**
- **Instrucciones técnicas**
- **Manuales**
- **No conformidades**
- **Plantillas**
- **Procedimientos generales**
- **Procedimientos normalizados de trabajo**

5.1.1.2 Topología de documentos

La estructura de propiedades de cada tipo de documento sigue una estructura de información.

Todo el documento borrador una vez cargado en la biblioteca de trabajo, se registrará con un “**Código**”, el cual será automático y tendrá una creación en base a la topología del documento. Así mismo, en ciertos casos este código podrá ser escrito manualmente.

El campo “**Editores**” se refiere a todas aquellas personas que deben participar en el proceso de edición del documento.

El campo “**Aprobadores**” se refiere a todas aquellas personas que deben participar en el proceso de aprobación o validación del documento.

La propiedad “**Mediador**” hace referencia al responsable o mediador del documento.

El campo “**Lectores**” hace referencia a aquellos usuarios que han de leer el documento.

El campo “**Distribución externa**” se refiere a aquellos usuarios externos de la empresa que han de recibir una copia de la documentación publicada.

Las propiedades “**Justificados**” y “**justificados externos**” se refieren a aquellos usuarios que han de quedar notificados sobre la lectura obligatoria de documentos. Para cada tipo de documento se deberá poder realizar el registro de “Leído y comprendido” para una lista cerrada de usuarios (se determina a nivel del documento en cuestión). Es decir, el *software* deberá desarrollar el registro de recepción y de confirmación de lectura de documentos para poder implementar un sistema de trazabilidad.

A continuación, se muestra una tabla de propiedades que debe contener un tipo de documento.

*valores del estado ≡ Draft, En elaboración, Validación, Publicado, Cancelado

*valores de área ≡ Contrataciones, Dirección, Económico-Financiera, Infraestructuras y SSGG, IT, Operaciones, RRHH, Técnico-Calidad

*valores de subárea ≡ Almacén, Aprovisionamiento, Calidad, Centro, Contrataciones, Control de entradas, Dirección, Dirección técnica, Económico-Financiera, IT, Mantenimiento, Mensajería, Montadores, Planificación, PRL, Producción, RRHH, SAC, SSGG, Transporte

Acciones correctivas

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	AC/AA/XX (AA = Año en curso) (XX = Id SharePoint) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía ³	Sí	No	Acción Correctiva

³ Una taxonomía es una propiedad que agrupa un conjunto de términos o palabras.

				(Valor Read Only Oculto)
Estado	Taxonomía	Sí	No	*valores del estado
Área	Taxonomía	Sí	No	*valores de área
Subárea	Taxonomía	Sí	No	*valores de subárea
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 10. Estructura de propiedades del documento: acciones correctivas

Análisis de riesgos

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Obligatorio	Multivalor	AR/XX (XX = Id SharePoint) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Análisis de riesgo (Valor Read Only Oculto)
Estado	Taxonomía	Sí	No	*valores del estado
Área	Taxonomía	Sí	No	*valores de área
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 11. Estructura de propiedades del documento: análisis de riesgos

Control de cambios

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	CC/AA/XX (AA = Año en curso) (XX = Id SharePoint) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Control de Cambios (Valor Read Only Oculto)
Estado	Taxonomía	Sí	No	*valores del estado
Área	Taxonomía	Sí	No	*valores de área
Subárea	Taxonomía	Sí	No	*valores de subárea
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 12. Estructura de propiedades del documento: control de cambios

Documentos principales

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	DP/XX (XX = Id SharePoint) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Documento Principal (Valor Read Only Oculto)
Estado	Taxonomía	Sí	No	*valores del estado
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 13. Estructura de propiedades del documento: documentos principales

Instrucciones de trabajo

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	IT/ XX - Z (XX = Id SharePoint) (Z = Código del PNT o PG asociado) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Instrucción de Trabajo (Valor Read Only Oculto)
Vinculado a	Búsqueda ⁴	Sí	No	Vinculado a un PNT o PG
Estado	Taxonomía	Sí	No	*valores del estado
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 14. Estructura de propiedades del documento: instrucciones de trabajo

Instrucción técnica

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	TE/ XX - Z (XX = Id SharePoint) (Z = Código del PNT o PG asociado) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Instrucción Técnica (Valor Read Only Oculto)

⁴ Una columna de búsqueda es una propiedad que está configurada para mostrar un valor de una columna de otra lista.

Vinculado a	Búsqueda	Sí	No	Vinculado a un PNT o PG
Estado	Taxonomía	Sí	No	*valores del estado
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 15. Estructura de propiedades del documento: instrucción técnica

Manuales

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	MA/ XX - Z (XX = Id SharePoint) (Z = Código del PNT asociado) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Manuales (Valor Read Only Oculto)
Vinculado a	Búsqueda	No	No	Vinculado a un PNT o PG
Estado	Taxonomía	Sí	No	*valores del estado
Área	Taxonomía	Sí	No	*valores de área
Subárea	Taxonomía	Sí	No	*valores de subárea
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 16. Estructura de propiedades del documento: manuales

No conformidades

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	NC/AA/XX (AA = Año en curso) (XX = Id SharePoint) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	No Conformidad (Valor Read Only Oculto)
Estado	Taxonomía	Sí	No	*valores del estado
Área	Taxonomía	Sí	No	*valores de área
Subárea	Taxonomía	Sí	No	*valores de subárea
Clasificación	Taxonomía	Sí	No	Cliente Interna Proveedor
Estado asunto	Taxonomía	Sí	No	Abierto Cerrado
Fecha apertura	Fecha sin hora	Sí	No	
Fecha cierre	Fecha sin hora	No	No	
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	

Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 17. Estructura de propiedades del documento: no conformidades

Plantillas

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	PL/ XX - Z (XX = Id SharePoint) (Z = Código del PNT o PG asociado) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Plantilla (Valor Read Only Oculto)
Vinculado a	Búsqueda	Sí	No	Vinculado a un PNT o PG
Estado	Taxonomía	Sí	No	*valores del estado
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 18. Estructura de propiedades del documento: plantillas

Procedimientos Generales

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	PG/XX (XX = Id SharePoint) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Procedimiento General (Valor Read Only Oculto)
Estado	Taxonomía	Sí	No	*valores del estado
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 19. Estructura de propiedades del documento: procedimientos generales

Procedimientos normalizados de trabajo

Campo	Tipo	Obligatorio	Multivalor	Valores
Código	Texto	Sí	No	PNT/Y/XX (Y = Inicial Subtipo) (XX = Id SharePoint) Read Only
Nombre	Texto	Sí	No	
Tipo documento	Taxonomía	Sí	No	Procedimiento normalizado de trabajo (Valor Read Only Oculto)
Estado	Taxonomía	Sí	No	*valores del estado
Tipo	Taxonomía	Sí	No	Dirección Técnica Infraestructuras Logísticos RRHH Sistemas Informáticos Varios
Fecha aprobación	Fecha sin hora	Sí	No	
Fecha caducidad	Fecha sin hora	Sí	No	
Editores	Grupo	Sí	Sí	
Aprobador nivel 1	Grupo	Sí	Sí	
Aprobador nivel 2	Grupo	No	Sí	
Aprobador nivel 3	Grupo	No	Sí	
Mediador	Persona	Sí	No	
Lectores	Grupo	No	Sí	Si no se indican, permiso de lectura "para todos"
Distribución externa	Persona	No	Sí	
Justificados	Grupo	No	Sí	
Justificados externos	Persona	No	Sí	
Vigente	Booleano	Sí	No	Sí/No

Tabla 20. Estructura de propiedades del documento: procedimientos normalizados de trabajo

5.1.2 Procesos de negocio

Hay tres procesos de negocio asociados a los documentos de calidad:

- **Proceso de publicación**
- **Proceso de distribución, notificación y recordatorio**
- **Proceso de caducidad**

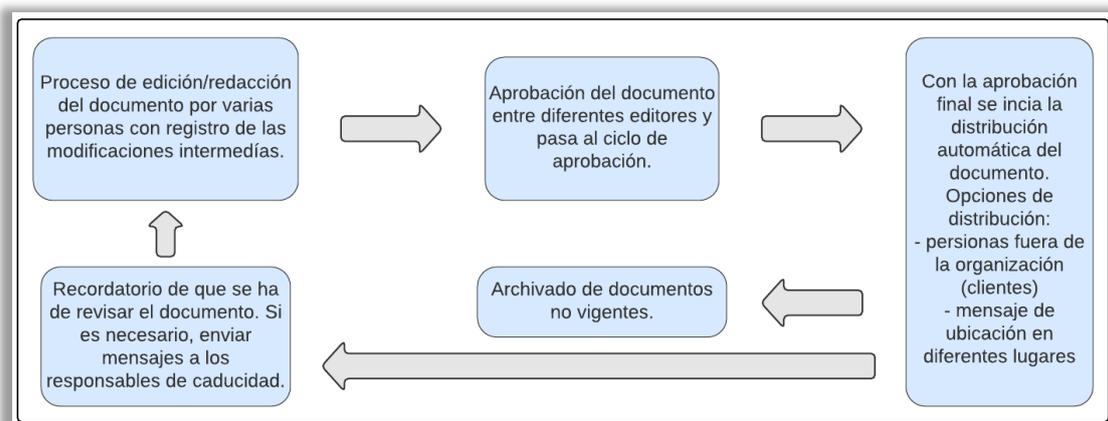


Figura 19. Proceso completo | Fuente: Elaboración propia

A continuación, se mostrarán los diagramas de estados de cada proceso de negocio para expresar el ciclo de vida que debe seguir un documento en el sistema.

5.1.2.1 Proceso de publicación

Los documentos de trabajo de la biblioteca de calidad deben seguir un proceso de publicación. Primeramente, debe existir un grupo de personas que podrán crear y editar el documento hasta su publicación. Estas se encargarán de crear el documento, editarlo y una vez que todas lo hayan redactado este documento pasará al siguiente estado.

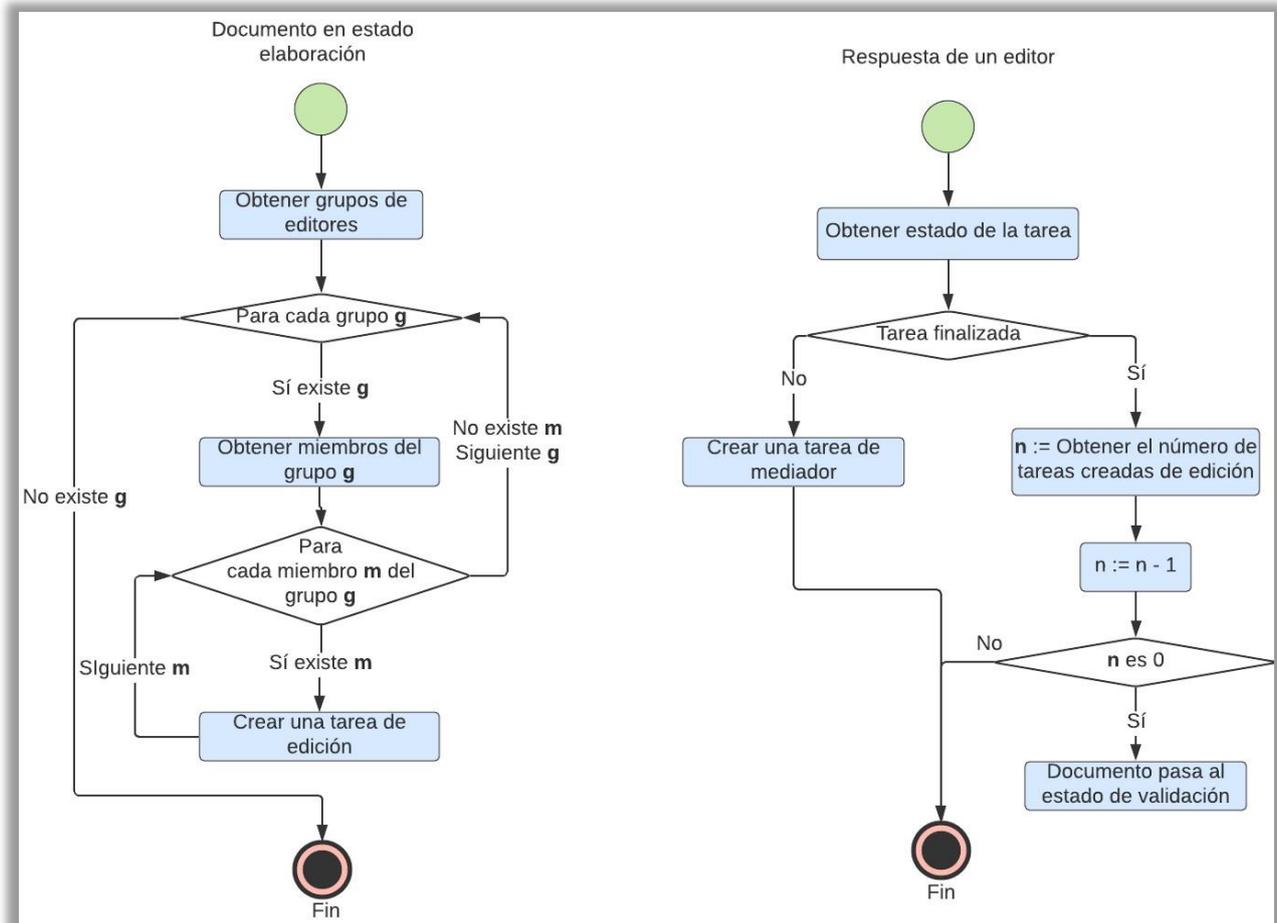


Figura 20. Diagrama de proceso de edición | Fuente: Elaboración propia

Durante el proceso de creación del documento, se debe guardar un control de versiones mayores.

Posteriormente, el documento redactado tiene que leerlo una serie de personas, que son los aprobadores, para validar el contenido de la documentación.

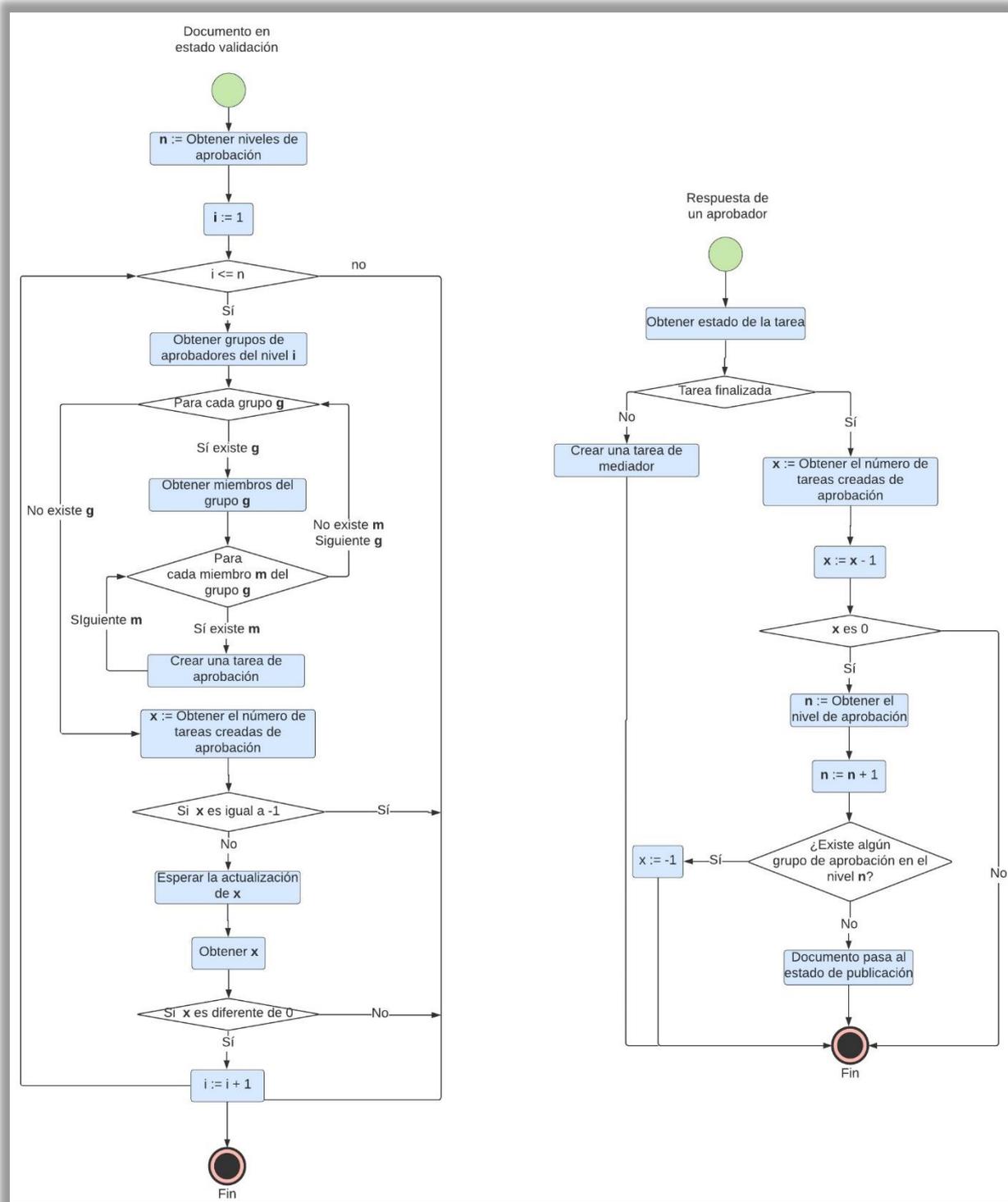


Figura 21. Diagrama de proceso de aprobación por niveles | Fuente: Elaboración propia

En cualquier caso, tanto los editores como los aprobadores tienen la potestad de rechazar sus tareas. En esta situación el mediador debe intervenir y tomar una decisión para que el proceso de publicación siga adelante.

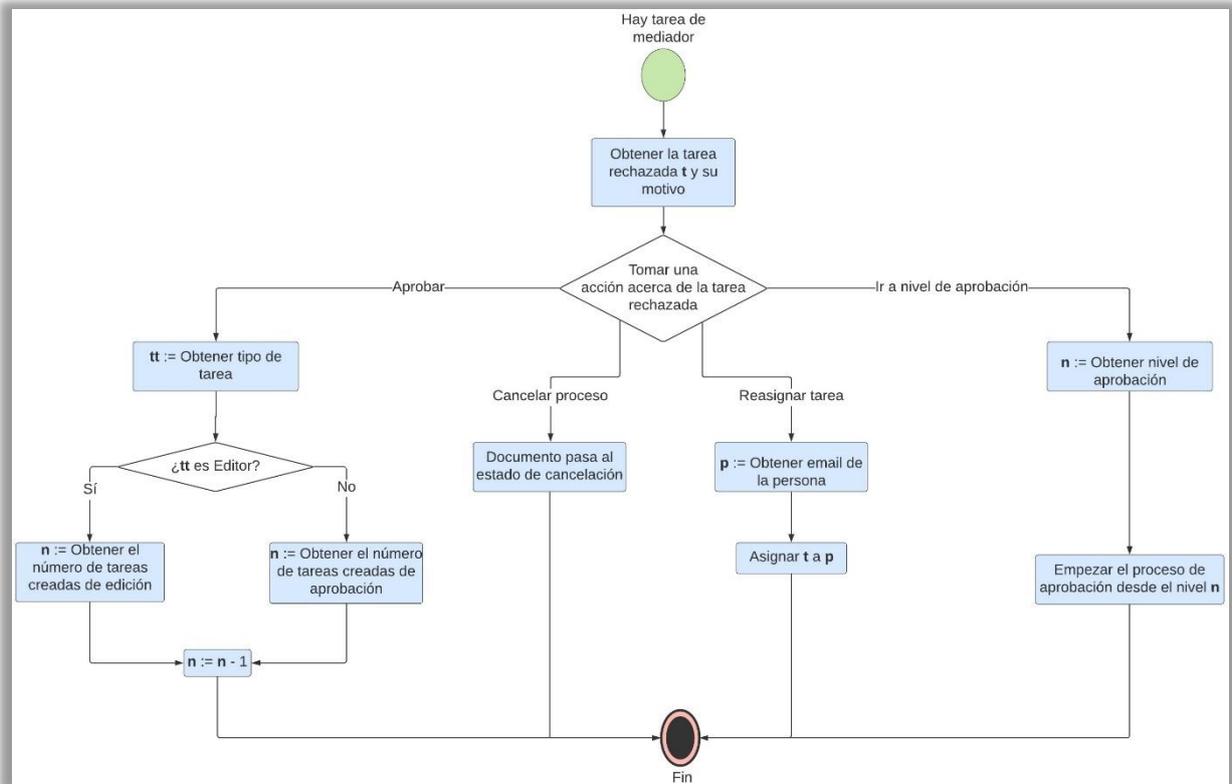


Figura 22. Diagrama de proceso del mediador | Fuente: Elaboración propia

Finalmente, la documentación debe ser publicada en una biblioteca de almacenamiento para que los usuarios puedan acceder y leer el documento oficial del departamento de *compliance*.

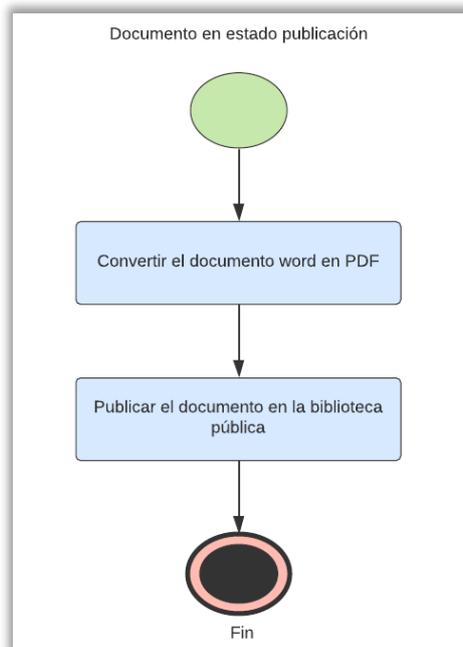


Figura 23. Diagrama de proceso de publicación | Fuente: Elaboración propia

En resumen, el proceso de publicación es un proceso complejo ya que involucra gestionar los siguientes procesos:

- Proceso de edición
- Proceso de aprobación
- Proceso de mediador
- Proceso de almacenamiento

5.1.2.2 Proceso de distribución, notificación y recordatorio

Una vez que el documento queda publicado, se inicia el proceso de distribución. El sistema ha de estar preparado para que la distribución del documento pueda ser para toda una plantilla de la empresa o cualquier otro amplio colectivo de personas. La distribución de un documento se puede hacer de dos maneras:

- Si el **documento es público**, es decir que han de leerlo todos los usuarios, entonces todas las personas destinatarias tendrán acceso de lectura a este.
- Si el **documento es de lectura obligatoria**, para una serie de personas, entonces el sistema ha de notificar a los usuarios de que tienen un documento pendiente de lectura y deben indicar la aceptación de la lectura.
 - Aquellos usuarios que no conceden el consentimiento de la lectura obligatoria, el sistema ha de notificarlos que tienen justificaciones pendientes hasta un número máximo de veces y cada un periodo de tiempo.

Se ha de tener en cuenta que, durante el proceso de publicación, el sistema deberá proporcionar notificaciones de recordatorio de tareas pendientes de realizar tanto a los editores como a los aprobadores hasta un número máximo de veces y cada un periodo de tiempo. Y deberá notificar al departamento de *compliance* si estos dos supera el número máximo de recordatorios fijado.

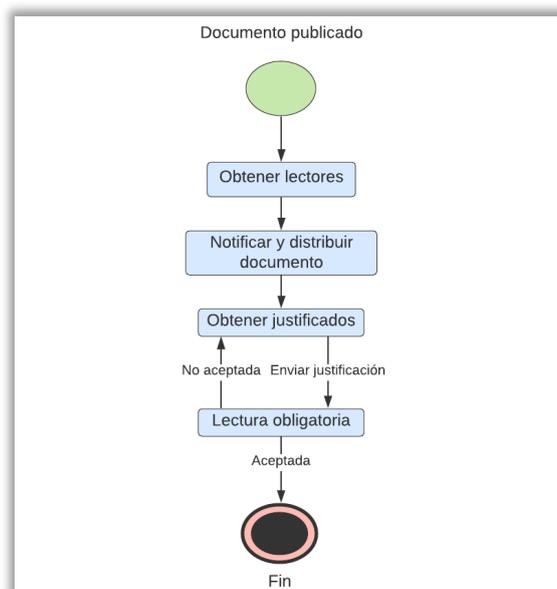


Figura 24. Diagrama de proceso de distribución, notificación y recordatorio
Fuente: Elaboración propia

5.1.2.3 Proceso de caducidad

Deberá poderse activar una caducidad de un documento, es decir, configurar un plazo, de forma que, una vez cumplido dicho plazo respecto de la fecha de publicación, se desencadene una revisión del documento.

El sistema ha de notificar a los responsables del documento un preaviso de caducidad preconfigurado a nivel de todos los documentos.

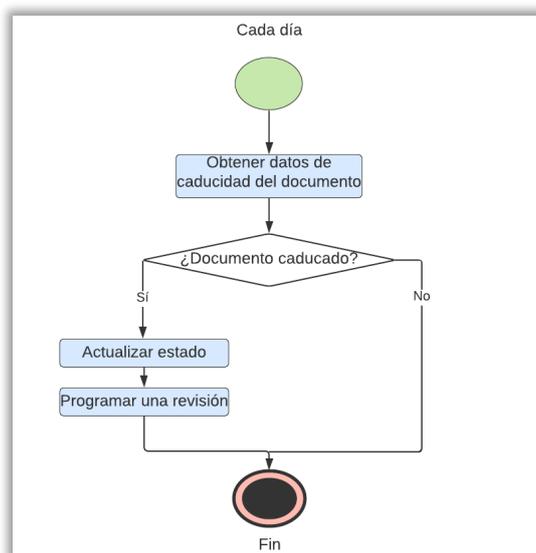


Figura 25. Diagrama de caducidad | Fuente: Elaboración propia

5.2 Proceso de obtención de requisitos

En este apartado se detalla el proceso que se ha seguido para obtener los requisitos del sistema. Un requisito [41] [42] es una condición que necesita el usuario para resolver un problema o conseguir un objetivo determinado tal que el *software* nuevo debe satisfacerlo. Por tanto, es necesario realizar un buen trabajo de análisis de requisitos [43] para definir los requerimientos de la aplicación. En este proyecto, el *project manager*, el desarrollador y el cliente tendrán un papel primordial para definir en detalle los requisitos del sistema a desarrollar.

En primer lugar, se planifica una reunión con el cliente para que explique el sistema que desea construir. Se describen los problemas y objetivos que se quiere lograr. Posteriormente, se inicia el proceso de análisis y la especificación de requisitos donde el *project manager* analiza los problemas y elecita los requisitos en una versión preliminar. A continuación, se muestra una breve descripción de los requisitos definidos:

- Crear un único punto de acceso para elaborar y publicar la documentación, y que sea accesible para todos los usuarios. Además, la gestión sobre un tipo de documento deberá cumplir:
 - Que se disponga de un control de versiones para garantizar que se está utilizando la última versión de un determinado documento.
 - Que se pueda obtener el histórico de versiones de un documento.

- Que se garantice que no exista más de una versión para un determinado documento vigente a la vez.
- Crear un proceso previo antes de que un documento se publique y sea leído o consultado por parte de los usuarios. Este proceso debe estar compuesto por:
 - Proceso de edición.
 - Proceso de validación del proceso de edición de los documentos por los implicados.
 - Proceso de publicación de los documentos con validaciones.
- El sistema debe permitir la publicación de cualquier tipo de formato de documento: PDF, PowerPoint, Word, Excel, etc. Aunque la recomendación para documentación oficial, que no sean plantillas modificables, sea en PDF, para de esta manera no permitir modificar documentación oficial.
- El sistema debe gestionar y controlar el permiso de acceso en documentos.
- El sistema debe registrar la trazabilidad de los cambios en documentos y de las respuestas de las tareas.
- El sistema debe distribuir los documentos por persona o grupo.
- El sistema debe permitir la notificación de lectura obligatoria y de aceptación de los documentos por parte de las personas de la compañía.
- El sistema debe gestionar la caducidad de un documento.
- Un documento se da de baja del sistema de dos formas:
 - Eliminándolo del sistema borrando dicho documento. Esto hace que no quede rastro de este.
 - Marcando dicho documento como no vigente, lo que provoca que no aparezca en las búsquedas, pero si en el historial de versiones.

Finalmente, este documento de requisitos se valida con el cliente y queda documentado como un contrato entre las dos partes. Dicho documento puede ser útil para calcular la estimación de costes y planificación (capítulo 3 y 4), y también para verificar y validar los artefactos resultantes acorde a la documentación.

En esta fase del ciclo de vida del *software* [43], el desarrollador describe el documento de especificación de requisitos en historias de usuarios (sección 5.2 y 5.3) que serán las entradas para el desarrollo de *software*. Notad que el proceso de requerimientos es iterativo y el hecho de usar una metodología ágil requiere prestar la atención al principio de incertidumbre de los requisitos [44] ya que pueden surgir nuevos.

5.3 Requisitos funcionales

En el apartado anterior se mencionó el principio de necesidad de los requisitos para desarrollar un sistema *software* y una breve descripción de estos. Los requisitos de *software* pueden clasificarse en dos tipos: funcionales y no funcionales. Si se usa una metodología tradicional, es conveniente documentar los requisitos funcionales mediante el uso de casos de uso. En caso contrario, es decir, para metodología ágil, se puede usar las historias de usuario para especificar tanto los requisitos funcionales como

no funcionales, que es el caso de este proyecto. En este apartado se describen los requisitos funcionales del sistema en un modelo estándar mediante el uso de historias de usuario [45] [46][47].

5.3.1 Historias de usuario

Un requisito funcional es una descripción general e informal que indica las funciones que el sistema es capaz de realizar sobre unas entradas para producir unas salidas. Una historia de usuario, que es equivalente a un requisito funcional, describe la funcionalidad valiosa de manera corta y esquemática. Consta de tres partes o tres C's [45] [46]: *card* (tarjeta), *conversation* (conversación) y *confirmation* (confirmación). A continuación, se describen las historias de usuario del sistema.

Propiedades rellenadas de un documento borrador			
Identificador	HU 1	Usuario	Responsable del documento
Sprint asignada	1	Horas reales	63
Prioridad	Alta	Complejidad	Media
Descripción			
Como responsable del departamento de <i>compliance</i> de un documento borrador autenticado en el sistema, quiero rellenar los campos de dicho documento para poder subirlo en un espacio único de trabajo.			
Criterio de aceptación			
<ul style="list-style-type: none"> • El responsable debe rellenar los campos necesarios u obligatorios de un documento (están definidos en la sección 5.1.1.2) <ul style="list-style-type: none"> ○ Los campos obligatorios son: nombre del documento, tipo de documento, área, editores, aprobadores y mediador. ○ Definir los campos de taxonomía en multi idioma (castellano y catalán). • Generar un código alfanumérico para identificar de manera unívoca al documento. El código generado debe seguir el formato definido en la sección 5.1.1.2, por ejemplo, NC/2021/10. 			

Tabla 21. Historia de usuario 1 – Propiedades rellenadas de un documento borrado
Fuente: Elaboración propia

Inicio de un proceso de publicación			
Identificador	HU 2	Usuario	Responsable del documento
Sprint asignada	2	Horas reales	5
Prioridad	Media	Complejidad	Baja
Descripción			
Como responsable de un documento borrador autenticado en el sistema, quiero iniciar el proceso de publicación para que el documento este elaborado.			
Criterio de aceptación			
<ul style="list-style-type: none"> • El documento debe tener los campos (propiedades) rellenados. • El estado del documento debe estar en <i>draft</i>. • El sistema debe cambiar el estado del documento. Es decir, el estado del documento debe pasar de <i>draft</i> a elaboración. <ul style="list-style-type: none"> ○ Añadir un botón para cambiar este estado y que sea intuitiva. 			

Tabla 22. Historia de usuario 2 – Inicio de un proceso de publicación
Fuente: Elaboración propia

Gestión de un proceso de edición			
Identificador	HU 3	Usuario	Sistema
Sprint asignada	2	Horas reales	25
Prioridad	Alta	Complejidad	Media
Descripción			
Como sistema de la aplicación, quiero gestionar el proceso de edición para que el documento pueda pasar a la fase de aprobación.			
Criterio de aceptación			
<ul style="list-style-type: none"> • El estado del documento debe estar en elaboración. • Crear una tarea para cada editor. • Si los editores finalizan la redacción de un documento, el sistema debe cambiar el estado del documento en aprobación. 			

Tabla 23. Historia de usuario 3 – Gestión de un proceso de edición | Fuente: Elaboración propia

Respuesta de un editor			
Identificador	HU 4	Usuario	Editor
Sprint asignada	2	Horas reales	15
Prioridad	Media	Complejidad	Baja
Descripción			
Como editor autenticado en el sistema, quiero realizar mi tarea asignada para cumplir mi rol.			
Criterio de aceptación			
<ul style="list-style-type: none"> • Debe existir una tarea de edición para el editor en cuestión. • El editor puede aceptar o rechazar la tarea asignada. <ul style="list-style-type: none"> ○ Añadir dos botones para realizar esta acción. • Si el editor finaliza la redacción, debe aprobar la tarea. • Si el editor rechaza su tarea, debe indicar el motivo de rechazo. <ul style="list-style-type: none"> ○ Crear una tarea para el mediador para que tome una decisión acerca de este rechazo. 			

Tabla 24. Historia de usuario 4 – Respuesta de un editor | Fuente: Elaboración propia

Gestión de un proceso de aprobación			
Identificador	HU 5	Usuario	Sistema
Sprint asignada	2	Horas reales	10
Prioridad	Alta	Complejidad	Alta
Descripción			
Como sistema de la aplicación, quiero gestionar el proceso de aprobación para que el documento pueda pasar a la fase de publicación.			
Criterio de aceptación			
<ul style="list-style-type: none"> • El estado del documento debe estar en validación. • Crear tareas de aprobadores en función del nivel de aprobador. <ul style="list-style-type: none"> ○ Crear una tarea para cada aprobador del nivel 1. 			

- Si los aprobadores del nivel 1 aprueban el documento redactado, se crea una tarea para cada aprobador del nivel 2, y así sucesivamente hasta el nivel n.
- Si el documento queda aprobado por todos los aprobadores, el sistema debe cambiar el estado del documento a publicado.

Tabla 25. Historia de usuario 5 – Gestión de un proceso de aprobación

Fuente: Elaboración propia

Respuesta de un aprobador			
Identificador	HU 6	Usuario	Aprobador
Sprint asignada	2	Horas reales	5
Prioridad	Media	Complejidad	Baja
Descripción			
Como aprobador autenticado en el sistema, quiero realizar mi tarea asignada para cumplir mi rol.			
Criterio de aceptación			
<ul style="list-style-type: none"> ● Debe existir una tarea de aprobación para el aprobador en cuestión. ● El aprobador puede aceptar o rechazar la tarea asignada. <ul style="list-style-type: none"> ○ Añadir dos botones para realizar esta acción. ● Si el aprobador aprueba la tarea, debe indicarla. ● Si el aprobador rechaza su tarea, debe indicar el motivo de rechazo. <ul style="list-style-type: none"> ○ Crear una tarea para el mediador para que tome una decisión acerca de este rechazo. 			

Tabla 26. Historia de usuario 6 – Respuesta de un aprobador | Fuente: Elaboración propia

Respuesta de un mediador			
Identificador	HU 7	Usuario	Mediador
Sprint asignada	2	Horas reales	25
Prioridad	Alta	Complejidad	Alta
Descripción			
Como mediador autenticado en el sistema, quiero tomar una decisión sobre las tareas rechazadas para que el proceso de publicación siga su curso.			
Criterio de aceptación			
<ul style="list-style-type: none"> ● Debe existir una tarea de mediador para el mediador en cuestión. ● El mediador puede tomar una de las siguientes acciones sobre su tarea: <ul style="list-style-type: none"> ○ Aprobar la tarea que ha sido rechazada por un editor o aprobador. ○ Cancelar el proceso de publicación. ○ Reasigna la tarea a otra persona. ○ Puede indicar que el proceso de aprobación continúe a partir de un nivel en concreto. ● Incorporar 4 botones para que el mediador pueda realizar su acción. <ul style="list-style-type: none"> ○ Si la acción es reasigna la tarea a otra persona entonces el sistema ha de recoger el correo de dicho usuario. Para realizar esta acción el correo debe estar validado. ○ Si la acción es ir a nivel de aprobación entonces el sistema ha de recoger el nivel a partir del cual el proceso de aprobación debe continuar. Para realizar esta acción el nivel debe estar validado. 			

Tabla 27. Historia de usuario 7 – Respuesta de un mediador | Fuente: Elaboración propia

Publicación y distribución de un documento			
Identificador	HU 8	Usuario	Sistema
Sprint asignada	2	Horas reales	35
Prioridad	Alta	Complejidad	Media
Descripción			
Como sistema de la aplicación, quiero publicar un documento en una biblioteca pública y distribuirlo para que los usuarios puedan acceder y leer los documentos.			
Criterio de aceptación			
<ul style="list-style-type: none"> • El estado del documento debe ser publicado. • El documento borrador se ha de convertir en un documento pdf. <ul style="list-style-type: none"> ○ Se ha de grabar las propiedades del documento en el pdf. • Si el documento a publicar ya existe en la biblioteca entonces se hace una actualización de este. <ul style="list-style-type: none"> ○ Si se genera una versión superior, las tareas pendientes del documento han de quedar como no activas. 			

Tabla 28. Historia de usuario 8 – Publicación y distribución de un documento
Fuente: Elaboración propia

Volver a iniciar el proceso de publicación			
Identificador	HU 9	Usuario	Responsable del documento
Sprint asignada	3	Horas reales	1
Prioridad	Baja	Complejidad	Baja
Descripción			
Como responsable de un documento publicado autenticado en el sistema, quiero volver a iniciar el proceso de publicación para actualizar el contenido del documento y generar una versión actualizada.			
Criterio de aceptación			
<ul style="list-style-type: none"> • El documento debe tener los campos (propiedades) rellenos. • El estado del documento debe ser publicado. • El responsable debe cambiar el estado del documento. Es decir, el estado del documento debe pasar de publicado a <i>draft</i>. <ul style="list-style-type: none"> ○ Añadir un botón para cambiar este estado y que sea intuitiva. 			

Tabla 29. Historia de usuario 9 – Volver a iniciar el proceso de publicación
Fuente: Elaboración propia

Envío de notificaciones y recordatorios			
Identificador	HU 10	Usuario	Sistema
Sprint asignada	3	Horas reales	34
Prioridad	Media	Complejidad	Media
Descripción			
Como sistema de la aplicación, quiero enviar notificaciones por correo electrónico a los usuarios del sistema para mantenerles informados de las situaciones.			
Criterio de aceptación			

Una notificación, a través del correo electrónico, debe contener un asunto y cuerpo obligatoriamente. Es decir, el contenido de esta será diferente según el tipo de destinatario y se puede clasificar de la siguiente manera:

- Notificación de tareas creadas.
- Notificación de publicación de documento para los lectores.
- Notificación de publicación de documento para los participantes.
- Notificación de cancelación del proceso de publicación para los participantes.
- Recordatorio y notificación de tareas pendientes.
- Recordatorio y notificación a los justificados.
- Notificación de preaviso de caducidad de un documento.
- Notificación de caducidad de un documento.
- Notificación de sugerencia de un documento publicado a los responsables.

Tabla 30. Historia de usuario 10 – Envío de notificaciones y recordatorios

Fuente: Elaboración propia

Caducidad de documentos			
Identificador	HU 11	Usuario	Sistema
Sprint asignada	3	Horas reales	7
Prioridad	Media	Complejidad	Baja
Descripción			
Como sistema de la aplicación, quiero comprobar la caducidad de un documento para desencadenar una revisión y avisar al responsable del documento.			
Criterio de aceptación			
<ul style="list-style-type: none"> • Calcular el preaviso de caducidad. • Calcular fecha de caducidad. 			

Tabla 31. Historia de usuario 11 – Caducidad de documentos | Fuente: Elaboración propia

Control de accesos			
Identificador	HU 12	Usuario	Sistema
Sprint asignada	3	Horas reales	20
Prioridad	Alta	Complejidad	Media
Descripción			
Como sistema de la aplicación, quiero asignar o eliminar permisos de un elemento de la plataforma para controlar el acceso de cada usuario.			
Criterio de aceptación			
<ul style="list-style-type: none"> • Asignar o eliminar permisos a un usuario o a un grupo sobre un archivo de SharePoint. <ul style="list-style-type: none"> ○ Romper permisos de un documento. ○ Romper permisos de las tareas. 			

Tabla 32. Historia de usuario 12 – Control de accesos | Fuente: Elaboración propia

Mostrar menú de navegación			
Identificador	HU 13	Usuario	Sistema
Sprint asignada	4	Horas reales	15
Prioridad	Media	Complejidad	Baja
Descripción			
Como sistema de la aplicación, quiero mostrar la barra de navegación según el tipo de usuario para facilitar la navegación a las listas o bibliotecas de la plataforma.			
Criterio de aceptación			
<ul style="list-style-type: none"> • Crear dos tipos de menú de navegación: uno para los usuarios habituales y otro para los administradores. • Cada menú de navegación, que apunta a una lista o biblioteca, debe contener una serie de vistas para filtrar los resultados: <ul style="list-style-type: none"> ○ Mostrar todos los elementos ○ Mostrar aquellos elementos que satisfacen unas condiciones. 			

Tabla 33. Historia de usuario 13 – Mostrar menú de navegación | Fuente: Elaboración propia

Enviar sugerencias			
Identificador	HU 14	Usuario	Usuario
Sprint asignada	4	Horas reales	5
Prioridad	Media	Complejidad	Baja
Descripción			
Como usuario de la aplicación, quiero enviar comentarios a los responsables de un documento publicado para sugerir algunos cambios o mejoras.			
Criterio de aceptación			
<ul style="list-style-type: none"> • Añadir la opción de enviar sugerencias en los documentos publicados. 			

Tabla 34. Historia de usuario 14 – Enviar sugerencias | Fuente: Elaboración propia

Mostrar sugerencias			
Identificador	HU 15	Usuario	Sistema
Sprint asignada	4	Horas reales	5
Prioridad	Media	Complejidad	Baja
Descripción			
Como sistema de la aplicación, quiero mostrar los comentarios de un documento publicado para que los usuarios del sistema puedan leerlos.			
Criterio de aceptación			
<ul style="list-style-type: none"> • Mostrar las sugerencias enviadas en una vista de comentarios. 			

Tabla 35. Historia de usuario 15 – Mostrar sugerencias | Fuente: Elaboración propia

5.3.2 Modelo conceptual de los datos

En esta sección se define un diagrama UML para representar los conceptos del dominio del sistema a partir de la especificación de requisitos definida anteriormente. También se describen las restricciones textuales para exigir una condición de entrada y garantizar el correcto funcionamiento del *software*. Este esquema conceptual de UML del sistema es una representación gráfica y estática de conceptos, y relaciones entre ellos, que ayuda a comprender los componentes o entidades con los que está formado el sistema.

En primer lugar, podemos observar que las entidades más destacadas del sistema son *Documento*, *Tarea*, *Grupo* y *Usuario*. Cada documento debe pertenecer a un tipo de documento, de los 11 existentes, que contiene información acerca del responsable y de la caducidad del documento. A su vez, un documento puede especializarse en un *documento borrador* o *publicado*. El primero necesita la intervención de cada uno de los *usuarios* de los *grupos* de *editor*, *aprobador* y *mediador* para que el documento siga el proceso de publicación con el objetivo de publicarlo. El segundo se comunica con los *grupos lector* y *justificado* para distribuirlo.

Un documento puede tener asociado dos tipos de *tareas*: de *publicación* y de *justificado*. Cada tarea creada debe relacionarse con un *usuario* del sistema para notificarlo.

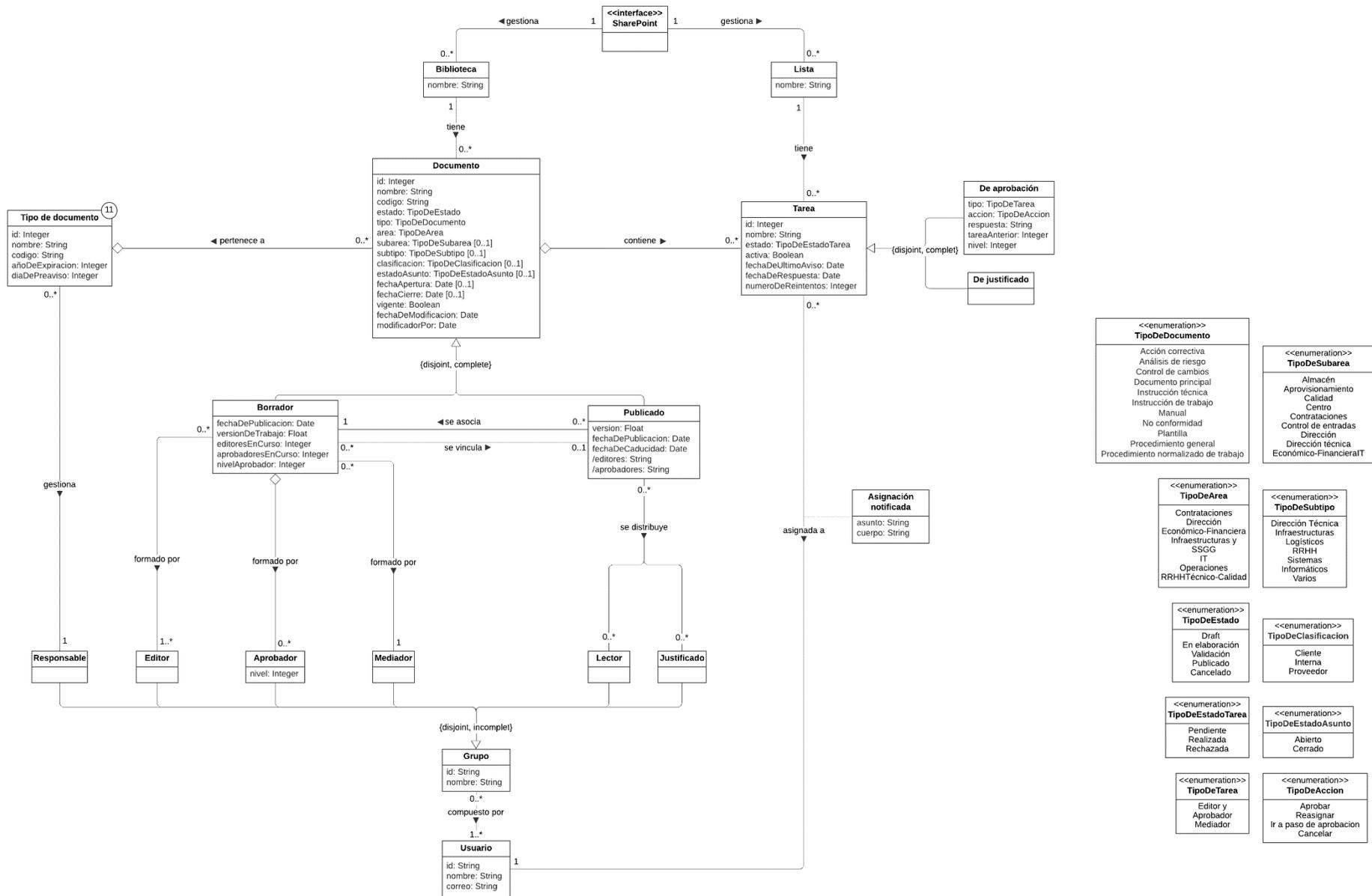


Figura 26. Diagrama conceptual | Fuente: Elaboración propia

Restricciones textuales

1. Claves externas: (Biblioteca, nombre), (Lista, nombre), (Tipo de documento, id), (Documento, id), (Tarea, id), (Grupo, id), (Usuario, id)
2. El *tipo* de un documento debe ser igual que el *nombre* del tipo de documento.
3. La *subárea* de un documento no puede ser vacía si el documento es de tipo Análisis de riesgos, Documento principal, No conformidad o Procedimiento normalizado de trabajo.
4. El *subtipo* de un documento no puede ser vacío si el documento es de tipo Procedimiento normalizado de trabajo.
5. La *clasificación*, *estado asunto*, *fecha de apertura* y *fecha de cierre* no pueden ser vacíos si el documento es de tipo No conformidad.
6. Un documento borrador de *tipo* Instrucción de trabajo, Instrucción técnica o Plantilla se puede *vincular* solamente a un documento de tipo Procedimiento general o Procedimiento normalizado de trabajo.
7. Tanto el grupo *responsable* como el *mediador* no pueden contener más de un usuario.
8. Las tareas de publicación deben ser de los grupos de *editor*, *aprobador* y *mediador*.
9. Las tareas de justificado deben ser del grupo de *justificado*.

Informaciones derivadas

1. /editores de un documento publicado es la concatenación de nombres de editores del documento borrador.
2. /aprobadores de un documento publicado es la concatenación de nombres de editores del documento borrador.

5.4 Requisitos no funcionales

En este apartado se describen los requisitos no funcionales del sistema. Un requisito no funcional es una descripción que hace referencia a los atributos o propiedades de calidad del sistema. Para clasificar los requerimientos no funcionales de este sistema se ha usado la plantilla Volere [48].

5.4.1 Plantilla de especificación de requisitos de Volere

RNF 1	
Tipo de requisito	Usabilidad y humanidad: fácil de usar
Descripción	
La aplicación debe usarse con interfaces de interacción simples e intuitivas. Se ha de mantener la coherencia entre diferentes páginas y garantizar la adecuada visualización en la plataforma web.	
Justificación	
Es importante debido a que una baja usabilidad podría hacer que el usuario deje de usar la aplicación o represente una queja al gerente de proyecto.	
Criterio de aceptación	
<ul style="list-style-type: none"> • El sistema debe seguir el diseño <i>responsive</i>. • El sistema debe mantener una vista adecuada según el tipo de usuario para facilitar la navegación en la plataforma. 	

Tabla 36. Requisito no funcional 1 – Usabilidad y humanidad: fácil de usar

Fuente: Elaboración propia

RNF 2	
Tipo de requisito	Usabilidad y humanidad: comprensión y cortesía
Descripción	
Las funcionalidades del sistema han de ser claras y precisas.	
Justificación	
Un usuario necesita, en todo momento, tener claro lo que está realizando.	
Criterio de aceptación	
<ul style="list-style-type: none"> • Si el usuario realiza una acción o funcionalidad, el sistema debe mostrar una breve explicación para facilitar el trabajo y comprensión. 	

Tabla 37. Requisito no funcional 2 – Usabilidad y humanidad: comprensión y cortesía

Fuente: Elaboración propia

RNF 3	
Tipo de requisito	Mantenimiento y soporte: mantenimiento
Descripción	
Debe ser posible la realización de modificaciones y reparaciones de uno o más procesos del sistema sin afectar la continuidad del servicio.	
Justificación	
El mantenimiento es fundamental para la continuidad, mejora y optimización del sistema.	
Criterio de aceptación	
<ul style="list-style-type: none"> • Se desarrollarán códigos y flujos reutilizables para tener una tasa alta de cambiabilidad. • Se documentará las funcionalidades del sistema de manera clara y concisa. 	

Tabla 38. Requisito no funcional 3 – Mantenimiento y soporte: mantenimiento

Fuente: Elaboración propia

RNF 4	
Tipo de requisito	Mantenimiento y soporte: adaptabilidad
Descripción	
La aplicación web debe ser compatible con los principales navegadores web de Internet.	
Justificación	
Es importante que los usuarios puedan acceder a la plataforma a través de diferentes tipos de navegadores, por ejemplo, Edge, Chrome, Firefox, Safari, Internet Explorer, etc.	
Criterio de aceptación	
<ul style="list-style-type: none"> • Se usará SharePoint Online moderno. 	

*Tabla 39. Requisito no funcional 4 – Mantenimiento y soporte: adaptabilidad
Fuente: Elaboración propia*

RNF 5	
Tipo de requisito	Seguridad: acceso
Descripción	
El sistema debe asegurar y permitir acceso sólo aquel usuario que tiene acceso a su perfil. También debe controlar acceso en ficheros y directorios según el nivel de permiso.	
Justificación	
No se puede permitir que un usuario acceda a los datos privados de otros usuarios.	
Criterio de aceptación	
<ul style="list-style-type: none"> • Cada usuario debe acceder a la plataforma con la cuenta de Microsoft de la compañía. • Cuando un documento se publique, se decidirá qué personas podrán acceder, en modo lectura, al mismo y, para los tipos de documento que aplique, que personas deberán justificar su lectura, etc. • Los grupos AD se protegerán con la seguridad de Microsoft de SharePoint. 	

Tabla 40. Requisito no funcional 5 – Seguridad: acceso | Fuente: Elaboración propia

RNF 6	
Tipo de requisito	Seguridad: privacidad
Descripción	
El sistema debe garantizar el uso responsable de los datos de los usuarios.	
Justificación	
Es importante tratar los datos o registros de las acciones de los usuarios cumpliendo las cuestiones legales y éticas.	
Criterio de aceptación	
<ul style="list-style-type: none"> • Para el uso de los datos de los usuarios se guardará los datos de manera segura siguiendo las leyes de protección de datos españolas y europeas (LOPDGDD y RGPD) 	

Tabla 41. Requisito no funcional 6 – Seguridad: privacidad | Fuente: Elaboración propia

6 Arquitectura del sistema

El capítulo seis se basa en la explicación de la arquitectura o el diseño del sistema. Primeramente, se presentará la visión global de la aplicación. Después, se hará una breve introducción a SharePoint para conocer los conceptos fundamentales de este y entender el contexto de la arquitectura. Además, se detallarán las nociones y tecnologías relacionadas a la parte *front-end* y *back-end* del sistema. Por último, se explicarán los diagramas de clases de diseño y de secuencia.

6.1 Visión global

La arquitectura del sistema de este proyecto es un diseño basado en SharePoint en el que se hace uso de los componentes de Microsoft ya que estos se integran fácilmente con SharePoint Online. Actualmente, es uno de los modelos más utilizados para construir una plataforma web basada en SharePoint. Existen diferentes alternativas y que se puede combinar otros componentes o tecnologías proporcionados por la compañía de Microsoft.

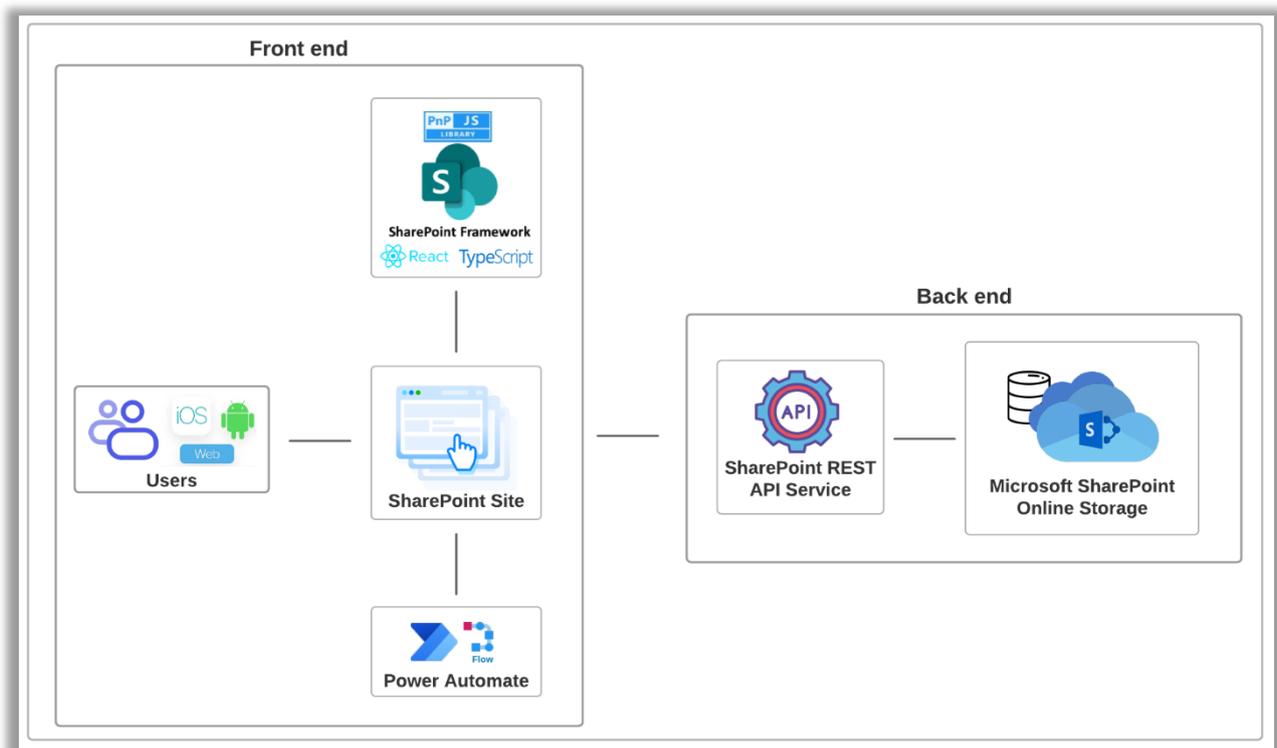


Figura 27. Esquema de la arquitectura del sistema en SharePoint | Fuente: Elaboración propia

En la figura anterior, se puede observar que la representación gráfica de la arquitectura del sistema está compuesta en dos bloques: *front-end* y *back-end*. Cada uno de ellos incorpora una serie de componentes o tecnologías y la relación entre ellas.

6.2 Introducción a SharePoint

Microsoft Office 365 (O365) es un paquete ofimático que proporciona soluciones o servicios como Word, PowerPoint, Outlook, SharePoint, Teams, Power Platform, etc.

para trabajar de manera óptima con las herramientas de Office y de colaboración, ya sea para empresas o particulares. Antiguamente, estas soluciones de colaboración y de mensajería estaban por separado. Actualmente, O365 nos permite integrar o interaccionar entre aplicaciones.

Microsoft SharePoint, también denominado SharePoint Online, es una herramienta *software* que permite compartir y administrar contenidos con las diferentes áreas de la organización para [49]:

- comunicarse con facilidad entre departamentos
- buscar información rápidamente
- potenciar el trabajo en equipo ya que es una plataforma de productividad unificada

La capa de SharePoint Online está constituida por unos pilares que describen las ideas fundamentales para construir una plataforma web. Estas bases definidas, que hace que SharePoint sea valorable, son: *sitio*, *comunidad*, *contenido* y *búsqueda efectiva de información*. Además, proporciona una serie de componentes esenciales para trabajar conjuntamente entre las dos partes, *front-end* y *back-end*, de la aplicación de manera eficaz. Los componentes centrales son [49]:

- **site**: como ya se había comentado anteriormente, *site* es uno de los pilares de SharePoint, es decir es una plataforma que es capaz de almacenar los datos existentes. Sin embargo, en términos de componentes de SharePoint es una colección de páginas web personalizables e interrelacionadas que se usa para mostrar los contenidos almacenados.
- **web page**: las páginas web se encargan de mostrar los contenidos almacenados y datos relevantes para la organización. Estas generalmente se diseñan con *layouts* o maquetación que se componen de *web parts* y *app parts*.
- **list**: las listas son un grupo de elementos o registros que se pueden compartir con los miembros del equipo. Son uno de los dos componentes que guardan los datos directamente en SharePoint.
- **library**: las bibliotecas tienen una estructura similar a una lista para gestionar los datos, pero a diferencia de las listas, almacenan archivos que contienen datos.
- **web part**: permite acceder a los contenidos que están tanto en un entorno de SharePoint como fuera de la web.
- **app part**: proporciona un acceso personalizado y un punto de vista al contenido que se almacena específicamente dentro de listas y bibliotecas.

En resumen, el elemento más importante en la parte de *front-end* es el SharePoint site, también conocido Portal o Intranet (empresarial), ya que permite la configuración y reutilización de UI predefinidas de Microsoft. Y los elementos más importantes para almacenar los datos en la parte de *back-end* son library y list.

6.3 Front-end

La parte *front-end* del sistema está orientada a proporcionar dos funcionalidades:

- Presentar informaciones mediante el uso de los componentes del sitio de SharePoint o personalizados tal que el usuario pueda interactuar con ellas.
- Ejecutar los procesos de negocio de la aplicación en el lado del cliente y comunicarse con la API de SharePoint.

6.3.1 Sitio web en SharePoint

Un sitio de SharePoint es la plataforma en la se construye una aplicación web. Es un espacio, accesible a través de un navegador, en el que se debe configurar la estructura de la página web o vistas con el objetivo de renderizar los componentes visuales para que el usuario pueda interactuar con estos. Notad que la plataforma de SharePoint debe ser configurada en el uso de los elementos estándares de *back-end* para evitar los problemas de renderizado y que SharePoint Online pueda almacenar los datos correctamente (para más detalles mirar la sección [7.1](#)).

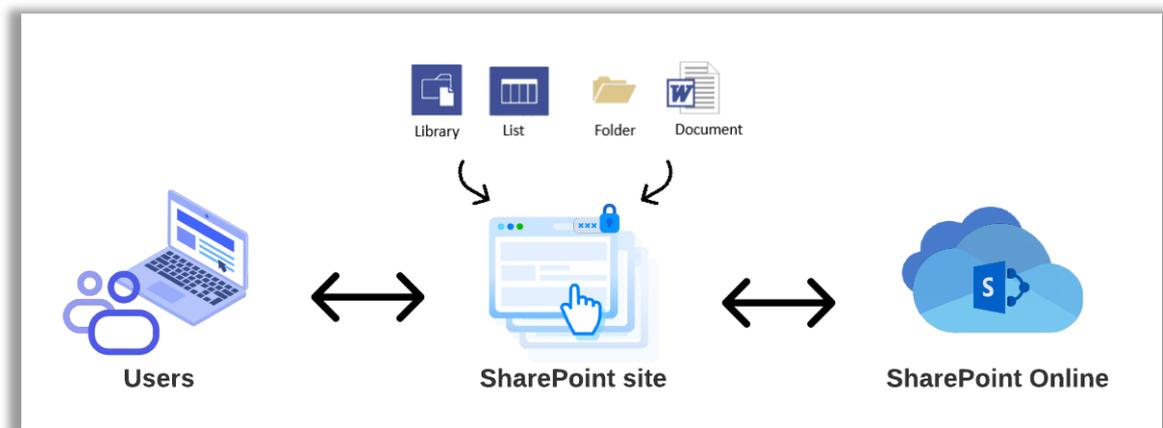


Figura 28. Elementos representativos de un sitio | Fuente: Elaboración propia

6.3.2 Interfaz de usuario personalizado

En la sección anterior se comentó que Microsoft por defecto proporciona una serie de elementos básicos a la hora de crear una plataforma SharePoint, por ejemplo, el sistema de *login*, navegación, interfaz responsive, configuración, seguridad, etc. (en la sección [7.1](#) se concretan los detalles). Sin embargo, existe la posibilidad de personalizar los componentes e incorporarlos en el sitio web.

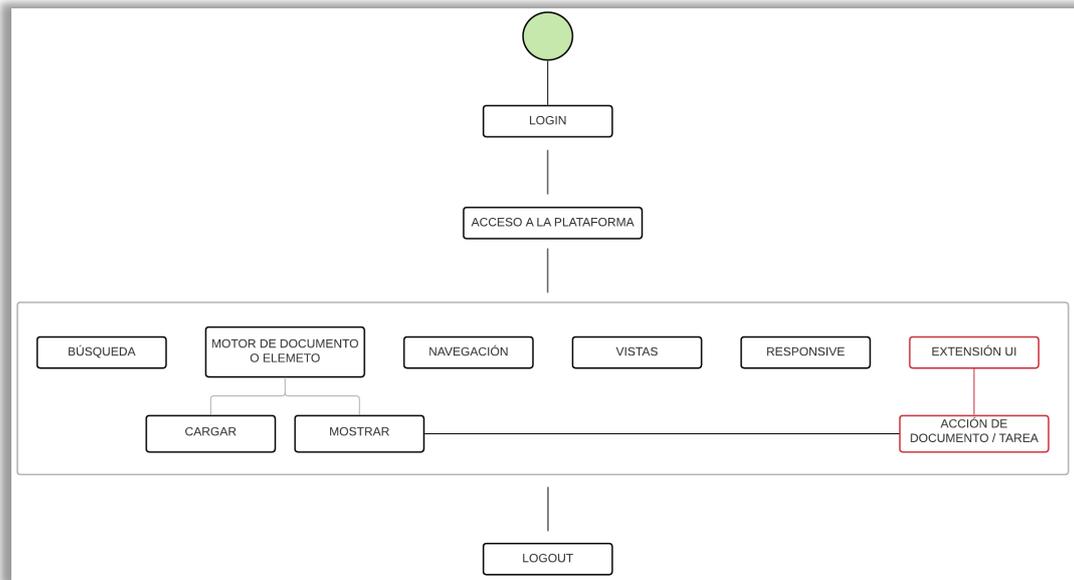


Figura 29. Componentes UI | Fuente: Elaboración propia

Leyenda: Componentes Microsoft Componentes personalizados

En la figura anterior se muestra el diseño de los componentes visuales para facilitar la interacción entre el usuario y el sistema. La mayoría de estos elementos son de Microsoft, y el objetivo principal es configurarlos tal que se adecue a la expectativa del cliente. A parte de eso, la personalización de interfaz gráfica se hará mediante el uso del componente llamado extensión y su implementación se concretará en la sección [7.2](#).

6.3.3 Procesos de negocio

Microsoft Power Platform es un conjunto de herramientas de Microsoft integrada dentro de O365, preparada para analizar, crear soluciones y automatizar los procesos de una empresa. Esta plataforma está compuesta por cuatro servicios: Power Business Intelligent, Power Apps, Power Automate y Power Virtual Agents. En este proyecto se usará la herramienta Power Automate, antes conocido como Microsoft Flow, ya que es un sistema con el que se pueden crear flujos de trabajo automatizados con el objetivo de simplificar los procesos empresariales y administrarlos de manera efectiva. Mientras que los otros tres están relacionados a los conceptos de *Big Data* e IA [50][51].

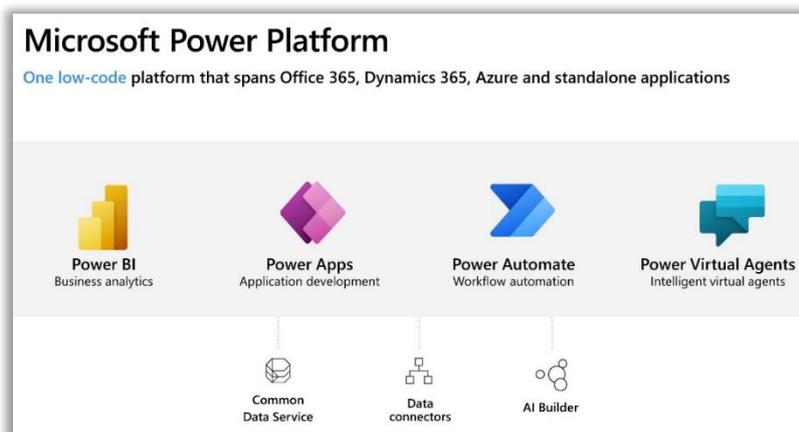


Figura 30. Microsoft Power Platform | Fuente: [50]

Este servicio online de Microsoft permitirá construir un diagrama de procesos con las funciones implementadas de Microsoft para resolver problemas o crear soluciones de negocio, es decir, permitirá controlar los procesos de negocio. A partir de aquí él será capaz de ejecutar un proceso de forma automática según el tipo de disparador que usa para iniciar el flujo, por ejemplo: cuando se modifica el registro de una entidad.

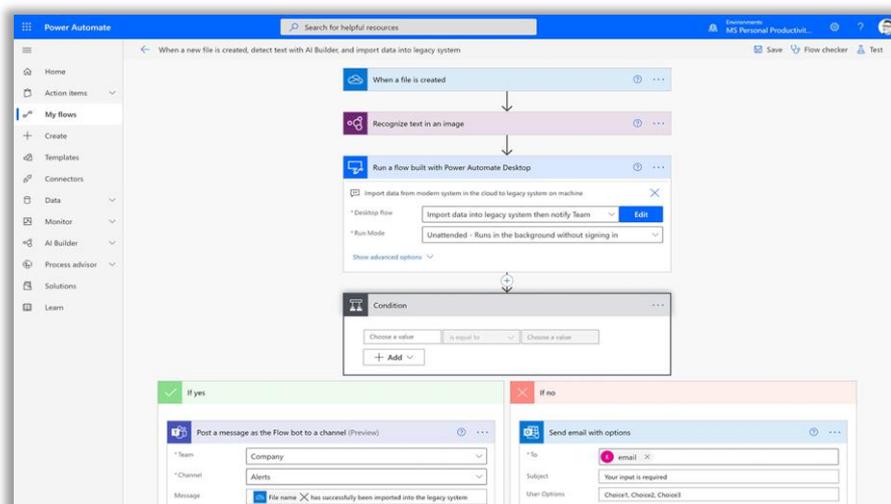


Figura 31. Power Automate – Flujo | Fuente: [23]

En la figura 31, se puede observar que Power Automate es capaz de conectarse con una gran variedad de herramientas y aplicaciones, por ejemplo: OneDrive, Power Apps, Teams, etc. Es decir, proporciona múltiples conectores predefinidos que permitirán integrarse con el resto de las aplicaciones de Microsoft o con aplicaciones de terceros. Actualmente, existen más de 300 conectores predefinidos de Microsoft (para más detalles mirar la sección [7.3](#))

También se puede observar que no es necesario conocer los conceptos avanzados de la programación para diseñar un flujo basado en Power Automate ya que este se categoriza como un servicio *low-code* para implementar soluciones de forma rápida y eficiente. La filosofía de *low-code* se basa en el uso de las herramientas de modelado visual para disminuir la codificación necesaria para crear una solución empresarial [52].

6.4 Back-end

La parte de *back-end* del sistema se encarga de realizar las siguientes funciones:

- Atender las peticiones de datos provenientes de los componentes de *front-end*.
- Comunicarse con el servidor de Microsoft para obtener los datos de la plataforma web y devolverlos.

6.4.1 SharePoint REST API Service

SharePoint REST API es un servicio de transferencia de estado representacional que incluye SharePoint para interactuar con los datos de la plataforma web. Esto significa que se puede realizar operaciones de creación, lectura, actualización y eliminación (CRUD) desde los complementos de SharePoint, soluciones y aplicaciones cliente con las tecnologías web que admiten el protocolo Rest.

Esta API proporciona el uso de los comandos o métodos HTTP para construir una petición HTTP Restful con el objetivo de indicar la acción de esta solicitud. Los comandos son: GET, POST, PUT y DELETE. Y para operar o acceder a los contenidos de la plataforma proporciona una serie de *endpoints*. Toda la URL de un *endpoint* de SharePoint API debe empezar con “https://siteurl/Sites/SiteName/_api/” y posteriormente se debe concretar la ruta final para recuperar informaciones necesarias. Por ejemplo: “https://siteurl/Sites/SiteName/_api/web/lists/getByTitle('SGQ')/items”. Los *endpoints* más destacados son:



Site	Web	Publishing	User Profiles	Search	Taxonomy
Webs	Lists	Pages	Users	Query Results	Managed Metadata
Features	Items	Variations	Profiles	Suggestions	
Event Receivers		Navigation	Activity		

Figura 32. SharePoint API endpoints | Fuente:
[<https://jandresval.files.wordpress.com/2017/04/restapi.jpg>]

6.4.2 Base de datos de SharePoint Online

Como ya se había comentado anteriormente, SharePoint Online es una versión de SharePoint basada en la nube donde la compañía Microsoft se encarga de gestionar el almacenamiento de datos o la parte de *back-end*.

Generalmente, SharePoint Online significa que la plataforma SharePoint se aloja o se instala en un servidor en la nube de Microsoft. Las informaciones se pueden subir y guardar en las bibliotecas y listas de SharePoint site, que son la parte visible de

SharePoint. Por tanto, en este proyecto no será necesario diseñar la parte lógica y física e implementar un servicio de base de datos. Sin embargo, se puede entender la filosofía o diseño que hay detrás y que Microsoft sigue para gestionar sus bases de datos.

A continuación, se detalla la arquitectura de la base de datos que utiliza Microsoft para almacenar los datos de la plataforma SharePoint.

6.4.2.1 Arquitectura de SharePoint Online

La arquitectura moderna de SharePoint Online combina o hereda los conceptos fundamentales de las arquitecturas anteriores, que están basadas en SharePoint 2013 y SharePoint 2016, ya que estas son las bases de la arquitectura en la *cloud* [53].

- **SharePoint Online 2013** proporciona soluciones basadas en aplicaciones de servicio, *Service Application*. Es decir, diferentes servidores serán capaces de ejecutar diferentes instancias de un servicio. Esto hace que el rendimiento de la aplicación mejore notablemente ya que diferentes tareas o peticiones se pueden paralelizar fácilmente.

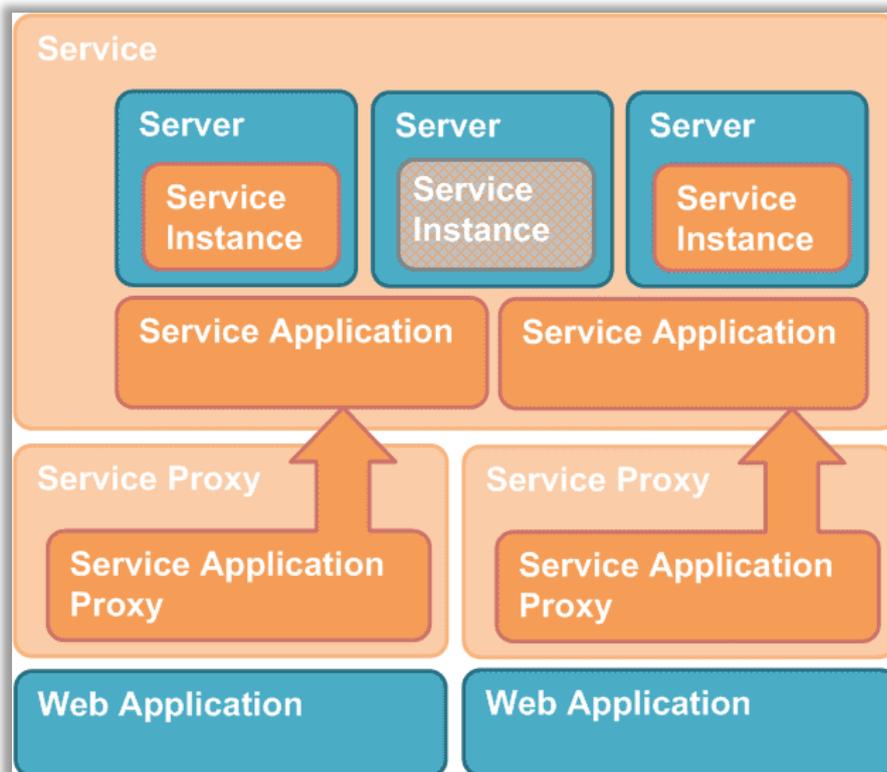


Figura 33. Topología de múltiples aplicaciones de servicio | Fuente: [53]

- En **SharePoint Online 2016** se incorporan nuevos servicios y se empiezan a desarrollar servicios fuera del núcleo de SharePoint, como puede ser el servicio de búsqueda. Además, se incorporan tres roles para balancear la carga de trabajo y para comunicarse con el servidor SQL. En la figura 34, se puede observar el funcionamiento interno de la topología de SharePoint 2016.

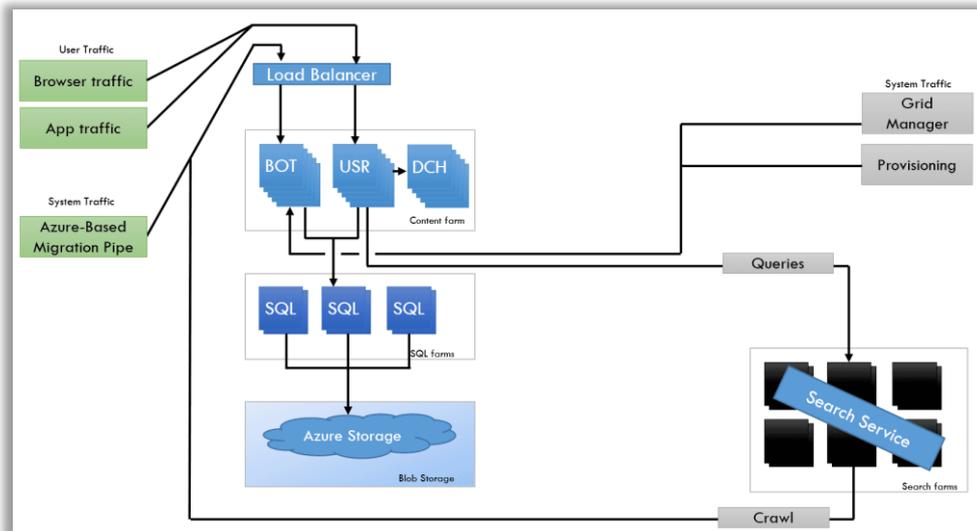


Figura 34. Arquitectura de SharePoint Online en 2016 | Fuente: [53]

A parte de estas arquitecturas, SharePoint Online ha ido evolucionando y ha ido añadiendo nuevas características de microservicios para dar soporte a los nuevos servicios de Microsoft O365. Esta nueva generación de arquitectura está preparada e implementada para ejecutarse en más de 50 *datacenters* en 20 localizaciones diferentes en todo el mundo. En la figura 35, se puede observar la arquitectura actual de SharePoint Online con una topología bastante parecida a la versión 2016.

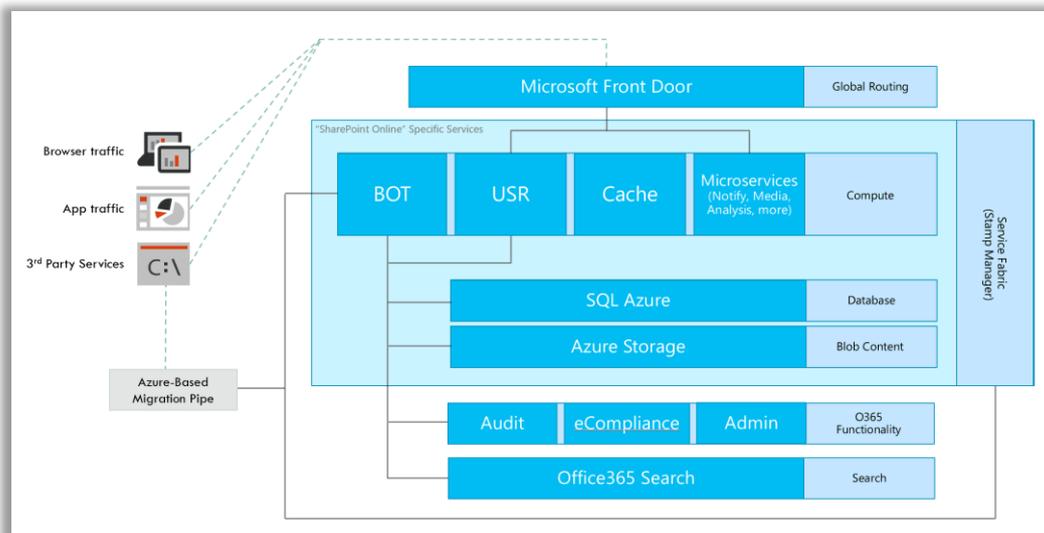


Figura 35. Arquitectura de SharePoint Online | Fuente: [53]

La novedad de esta arquitectura está en la incorporación de un rol de microservicio, y que todo se almacena en SQL Azure (servicio PaaS, *Platform as a Service*) o en *Azure Storage*. Parece ser que es un modelo bastante complejo, pero en el fondo se sigue usando una base de datos SQL⁵. La mejora está en que SharePoint Online usa varias bases de datos SQL, denominadas bases de datos de contenido, para almacenar los

⁵ Una base de datos SQL es aquella base de datos relacional que está escrita en lenguaje SQL (Structured Query Language)

datos, por ejemplo: contenido de sitios, elementos de lista, archivos, documentos, metadatos, etc. en *Azure Storage* de forma segura y cifrada.

En resumen, Microsoft gestiona la base de datos de SharePoint Online mediante el servidor SQL para almacenar información de los clientes que permanezcan dentro de SharePoint. Se parece bastante a una base de datos relacional (mirar tabla 42 y figura 36) ya que usa listas y columnas para organizar los datos, pero no se puede usar para sustituir una base de datos relacional tradicional, por ejemplo: SQL Server, ya que no es capaz de procesar grandes volúmenes de transacciones, es decir no puede manejar archivos que superen un tamaño máximo de 2GB.

SharePoint Online	SQL
Listas	Tablas
Elementos	Filas
Columnas	Columnas

Tabla 42. Comparación entre SharePoint Online y SQL | Fuente: Elaboración propia

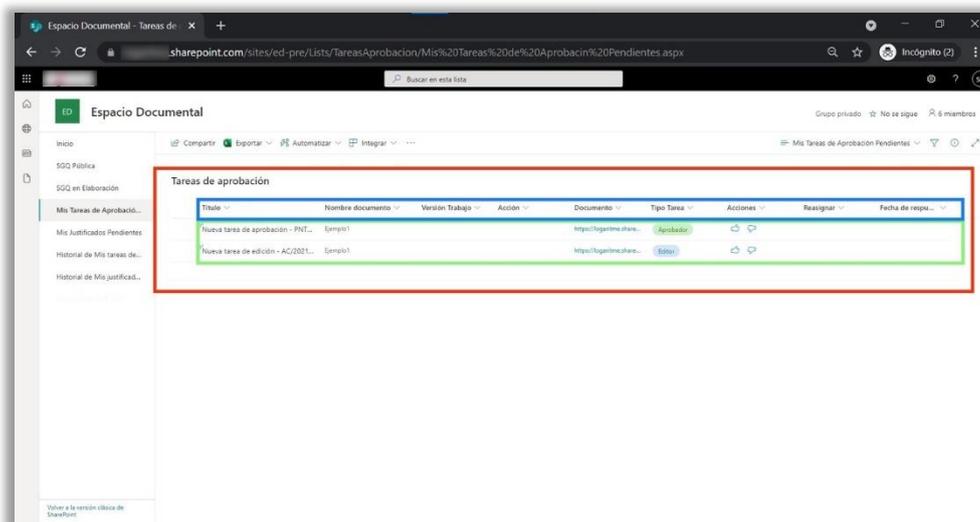


Figura 36. Lista con elementos y columnas | Fuente: SharePoint – Espacio Documental

Notad que, en este ejemplo, el color rojo hace referencia a la lista, el azul a las columnas y el verde a los elementos.

6.5 Diagrama de clases de diseño

En este apartado se muestra el diagrama de clases de diseño para describir la estructura del sistema y que se ha obtenido a partir del esquema conceptual de datos definido en la sección [5.3.2](#). Posteriormente, este diseño lógico relacional se implementará en el modelo de base de datos de SharePoint.

El diagrama de la figura 37 es muy similar al de la figura 26 ya que la mayoría de los elementos son compatibles de cara a la implementación, por ejemplo: las clases asociativas binarias, tipos enumerados, etc. Sin embargo, la clase asociativa “Asignación notificada” se ha transformado en una clase no asociativa y se relaciona con sus clases originales.

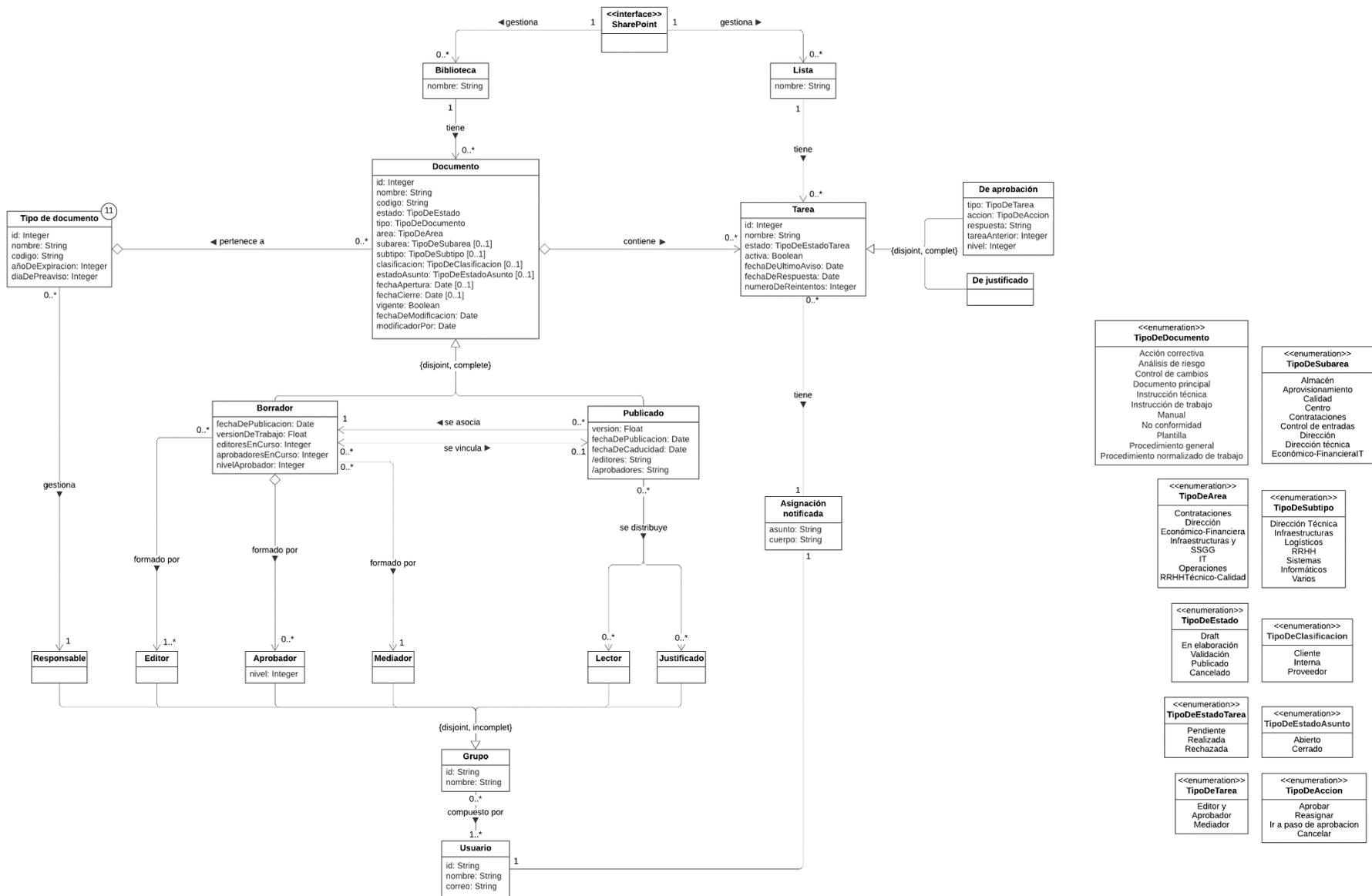


Figura 37. Diagrama de clases de diseño | Fuente: Elaboración propia

6.6 Diagramas de secuencia

En el capítulo anterior, concretamente en la sección [5.1.2](#), se explicaron los diagramas de estados de los procesos de negocio asociados a los documentos. En este apartado, se aclararán los diseños o diagramas de secuencia de las operaciones críticas del sistema para mostrar la interacción entre usuario, sistema y otros objetos en un tiempo determinado. El objetivo es simplificar los procesos o funcionalidades complejos del sistema de una manera abstracta y que posteriormente se implementarán en el entorno de SharePoint.

En el siguiente diagrama de secuencia se puede observar que hay una interacción entre el usuario y sistema al momento de subir un documento borrador en la plataforma. El sistema debe comunicarse con SharePoint Online para mantener la consistencia de base de datos.

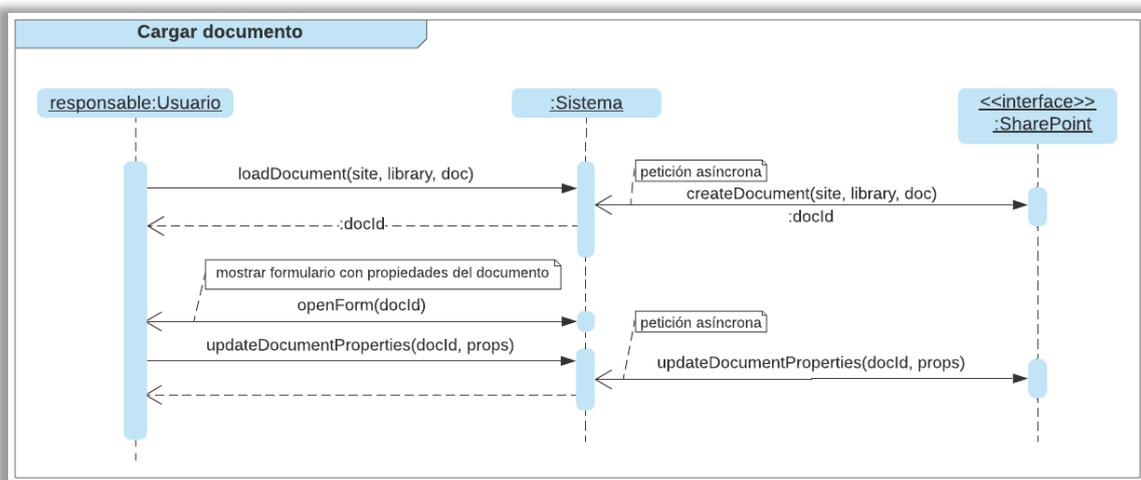


Figura 38. Diagrama de secuencia – Cargar documento | Fuente: Elaboración propia

Al comienzo, un documento se encuentra en estado draft o borrador. El sistema debe actualizar algunos campos iniciales y asignar los permisos adecuadamente.

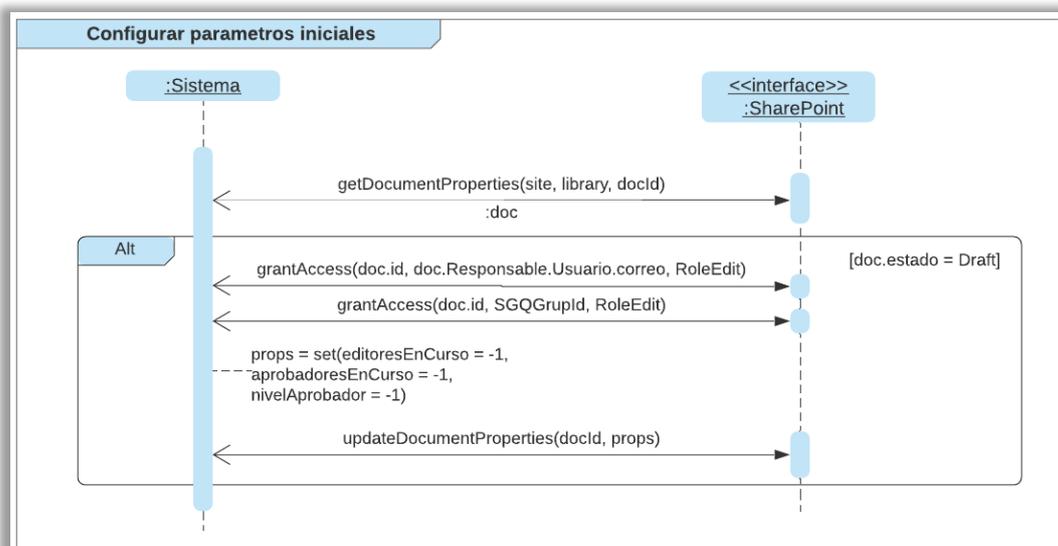


Figura 39. Diagrama de secuencia – Configurar parámetros iniciales | Fuente: Elaboración propia

Posteriormente, el responsable del documento podrá iniciar el proceso de publicación y el sistema deberá ejecutar todos los procesos de negocio descritos en el capítulo 5.

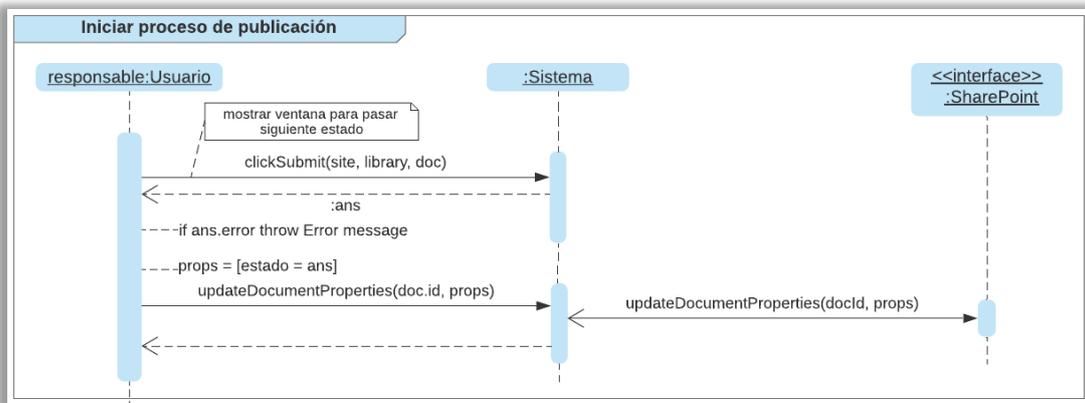


Figura 40. Diagrama de secuencia – Iniciar proceso de publicación | Fuente: Elaboración propia

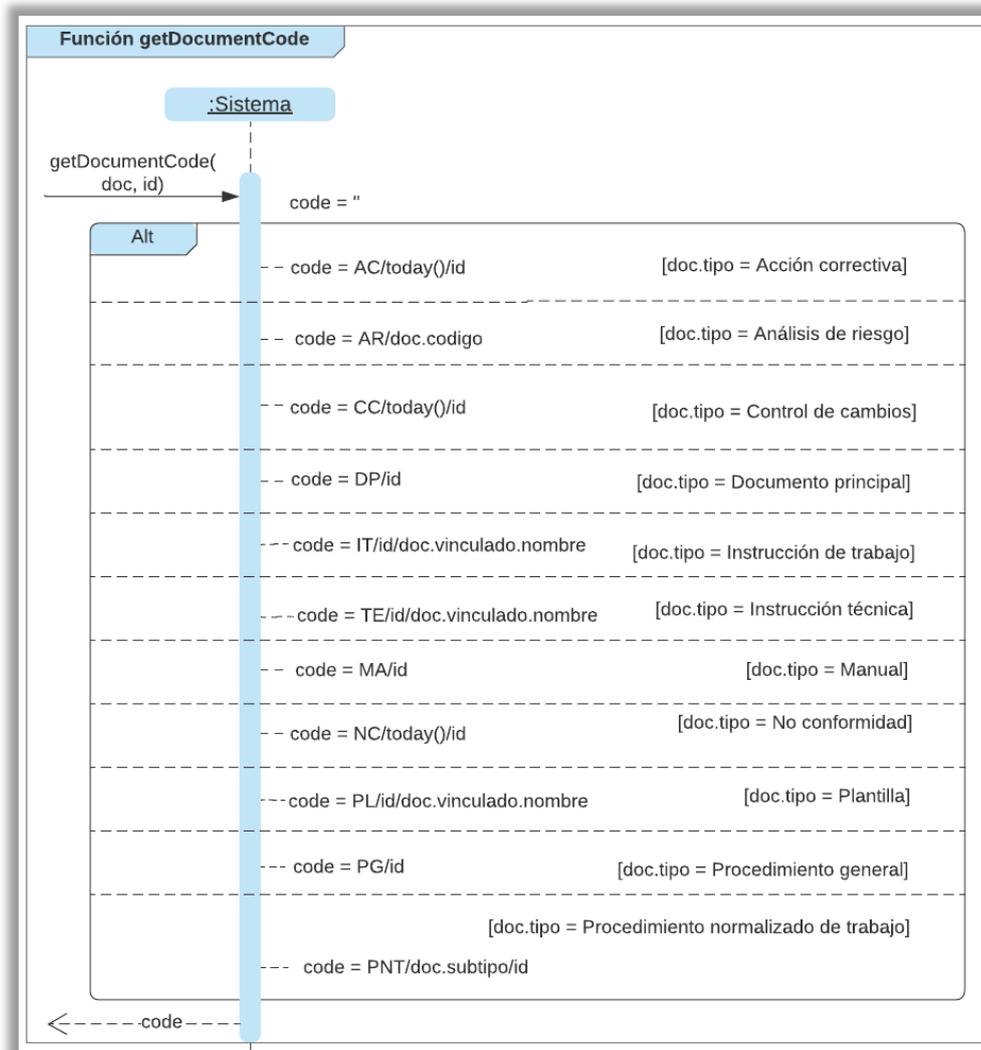


Figura 41. Diagrama de secuencia – getDocumentCode | Fuente: Elaboración propia

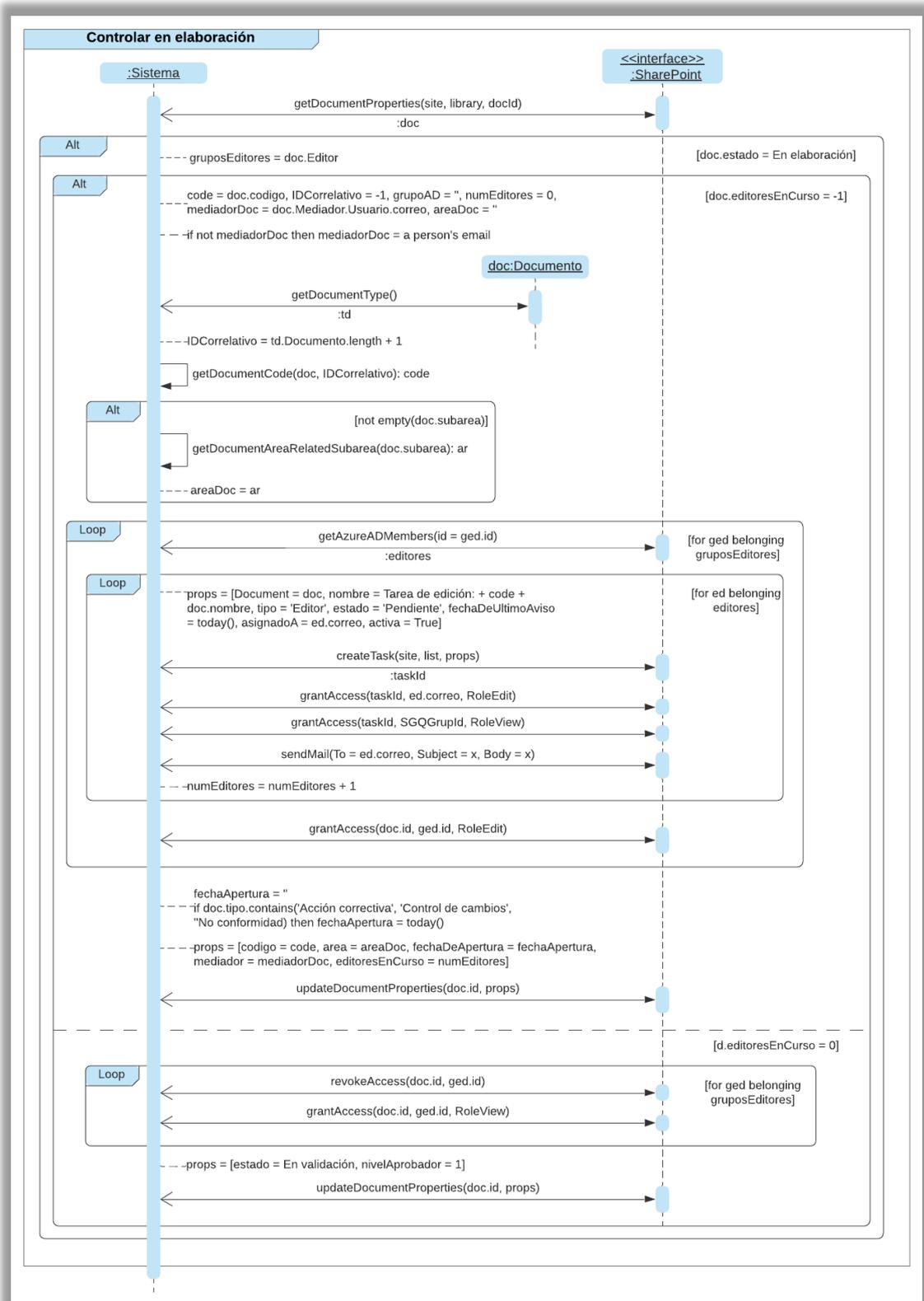


Figura 42. Diagrama de secuencia – Controlar en elaboración | Fuente: Elaboración propia

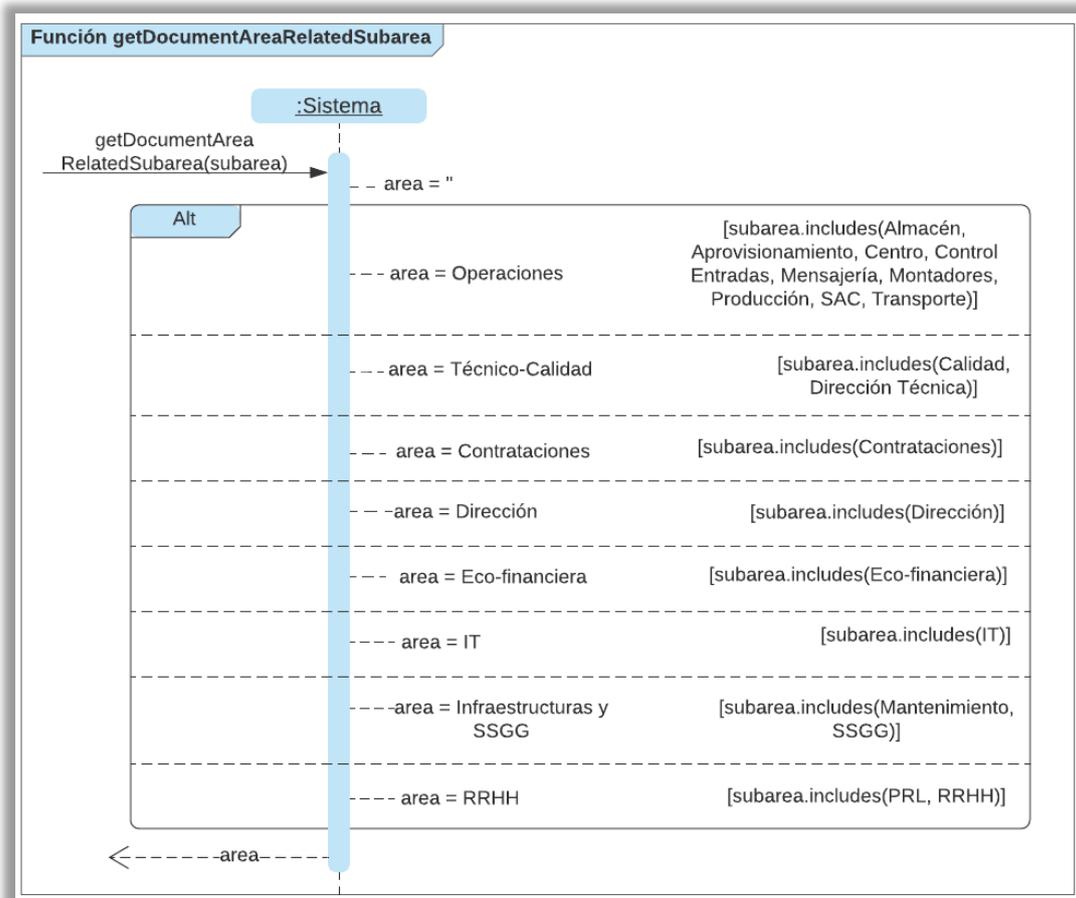


Figura 43. Diagrama de secuencia – getDocumentAreaRelatedSubarea | Fuente: Elaboración propia

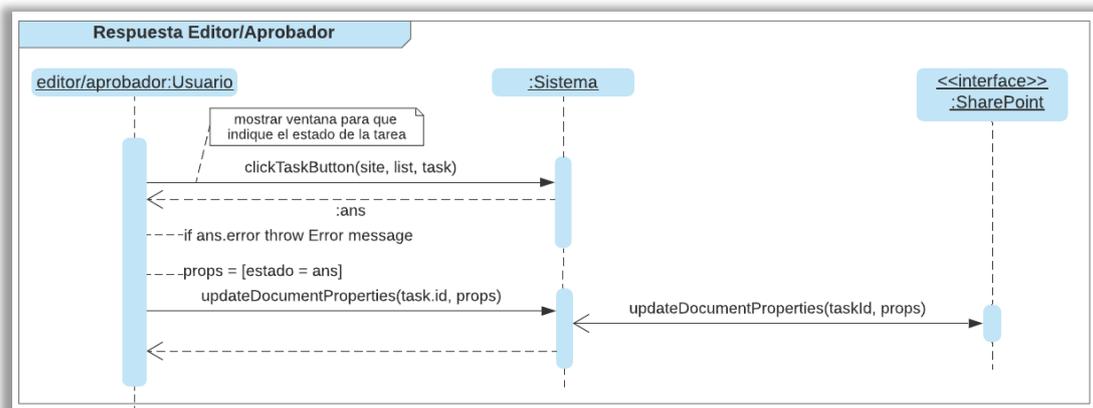


Figura 44. Diagrama de secuencia – Respuesta Editor / Aprobador | Fuente: Elaboración propia

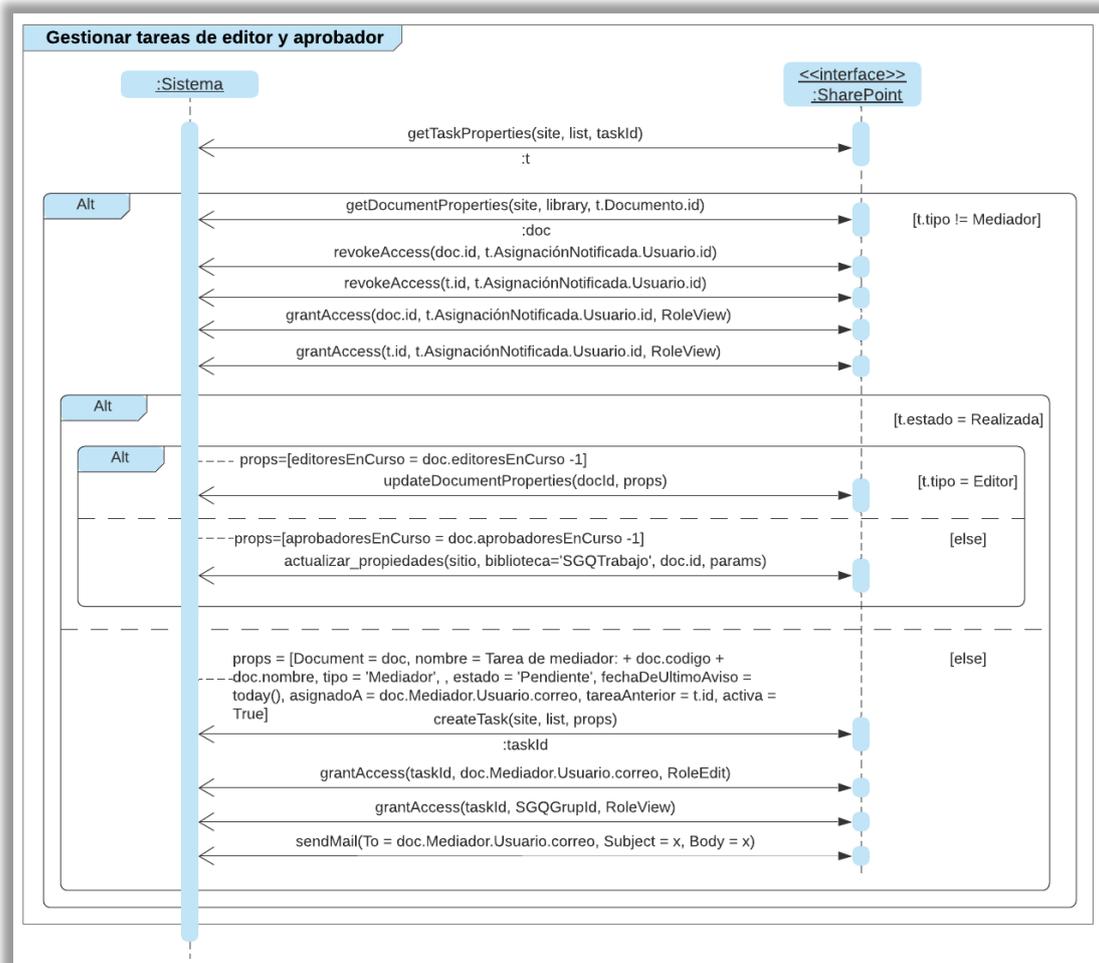


Figura 45. Diagrama de secuencia – Gestionar tareas de editor y aprobador | Fuente: Elaboración propia

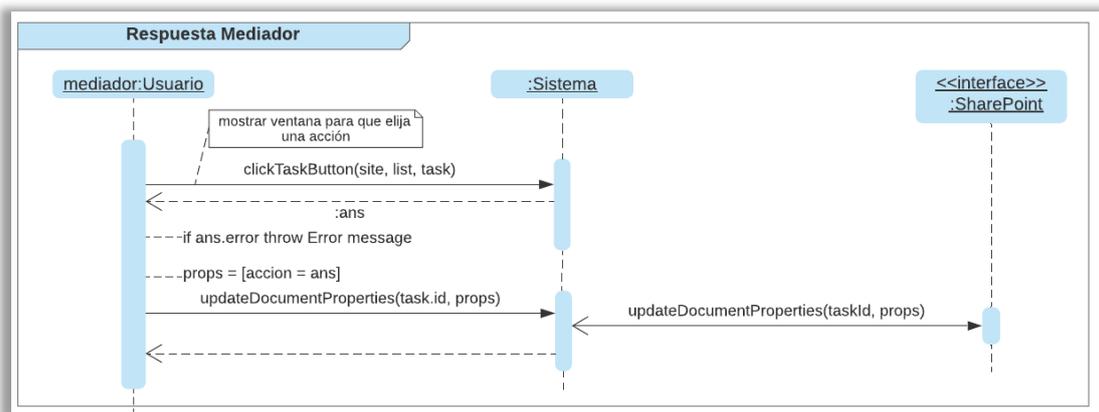


Figura 46. Diagrama de secuencia – Respuesta Mediador | Fuente: Elaboración propia

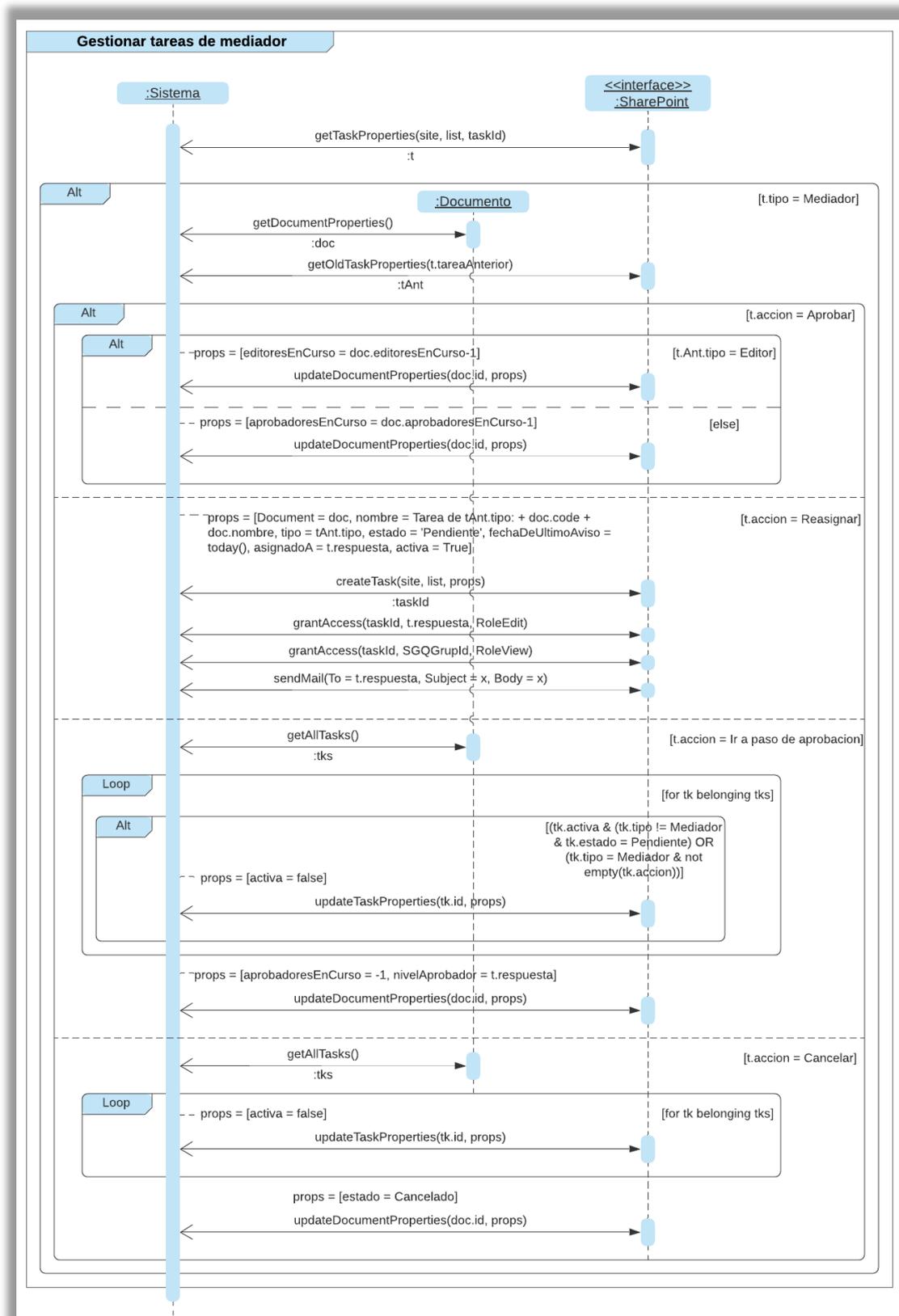


Figura 47. Diagrama de secuencia – Gestionar tareas de mediador | Fuente: Elaboración propia

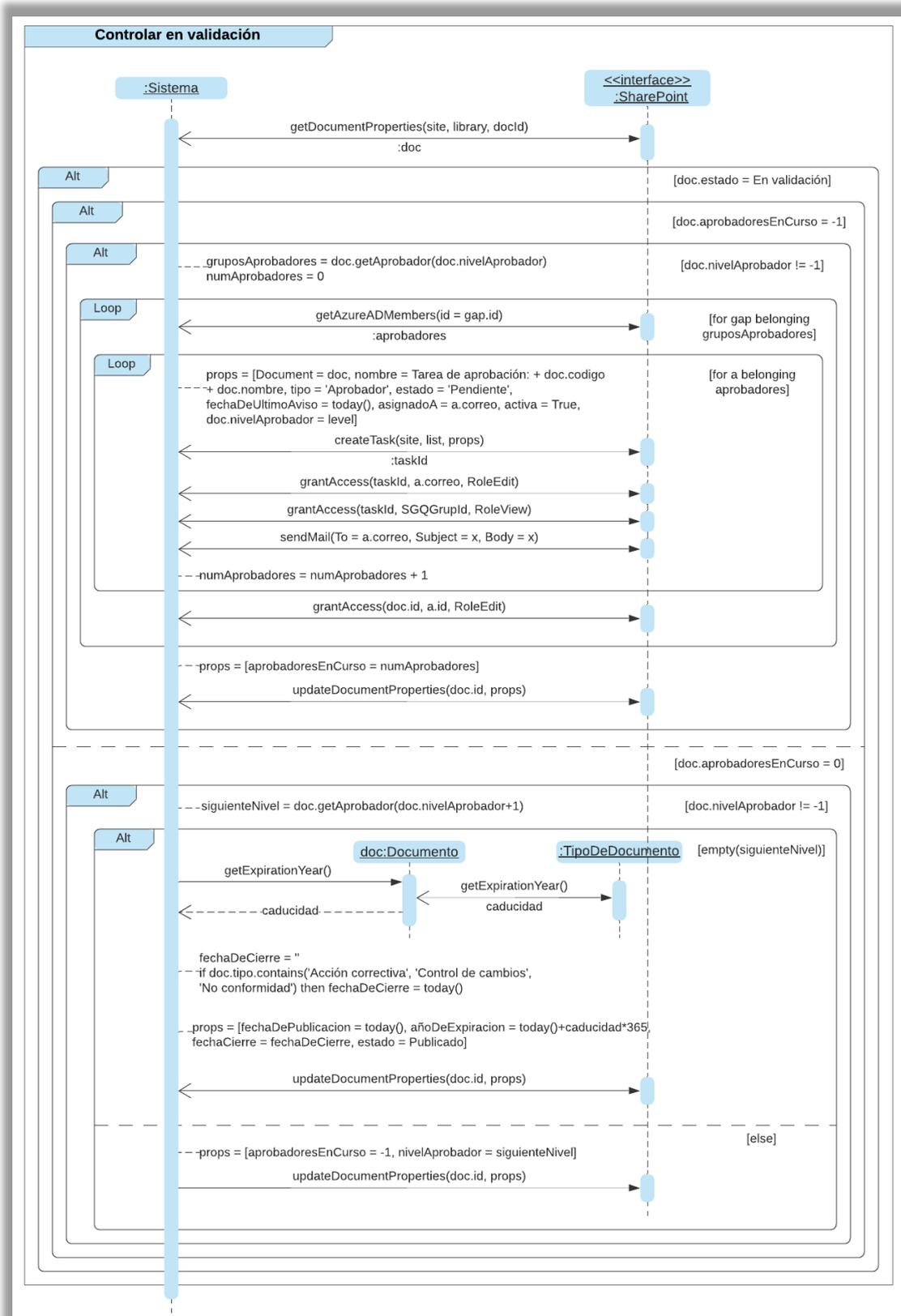


Figura 48. Diagrama de secuencia – Controlar en validación | Fuente: Elaboración propia

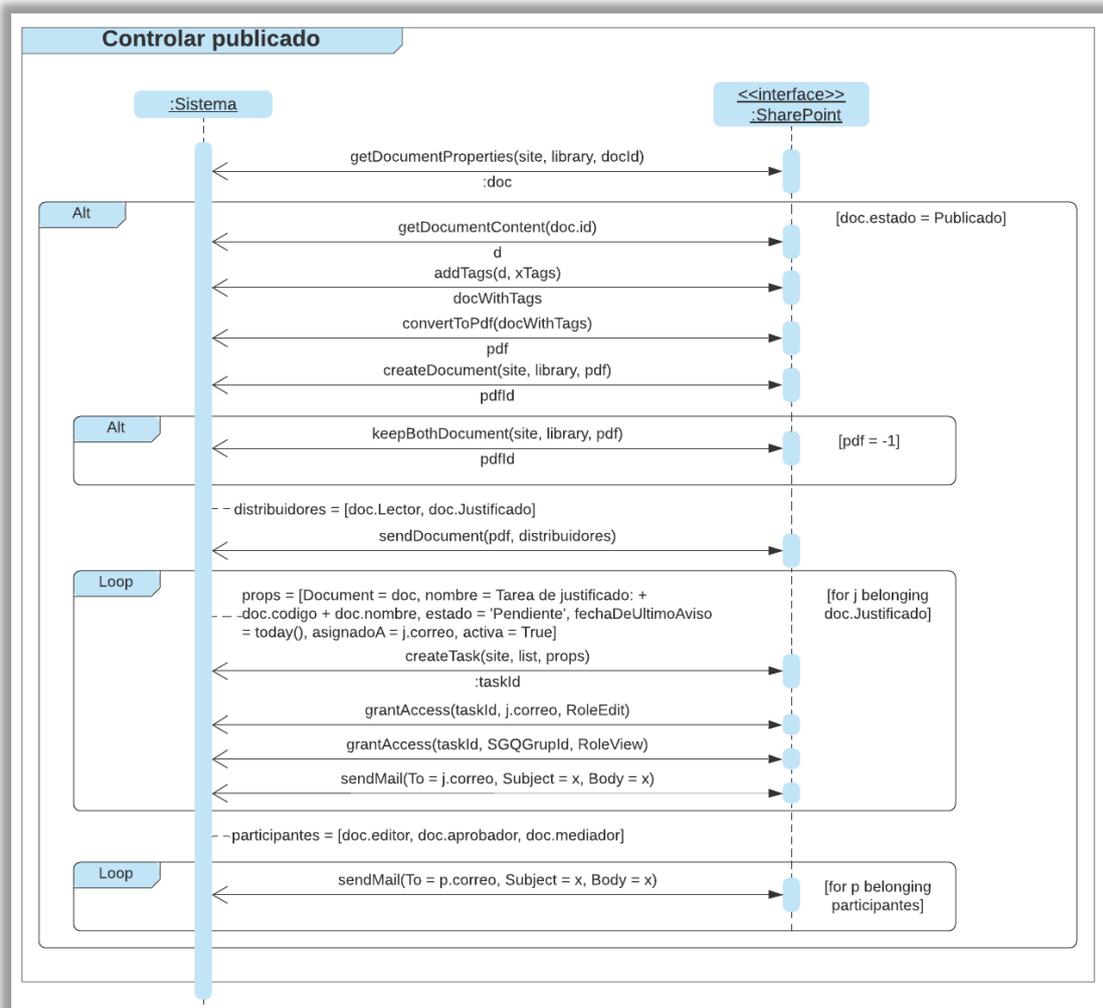


Figura 49. Diagrama de secuencia – Controlar publicado | Fuente: Elaboración propia

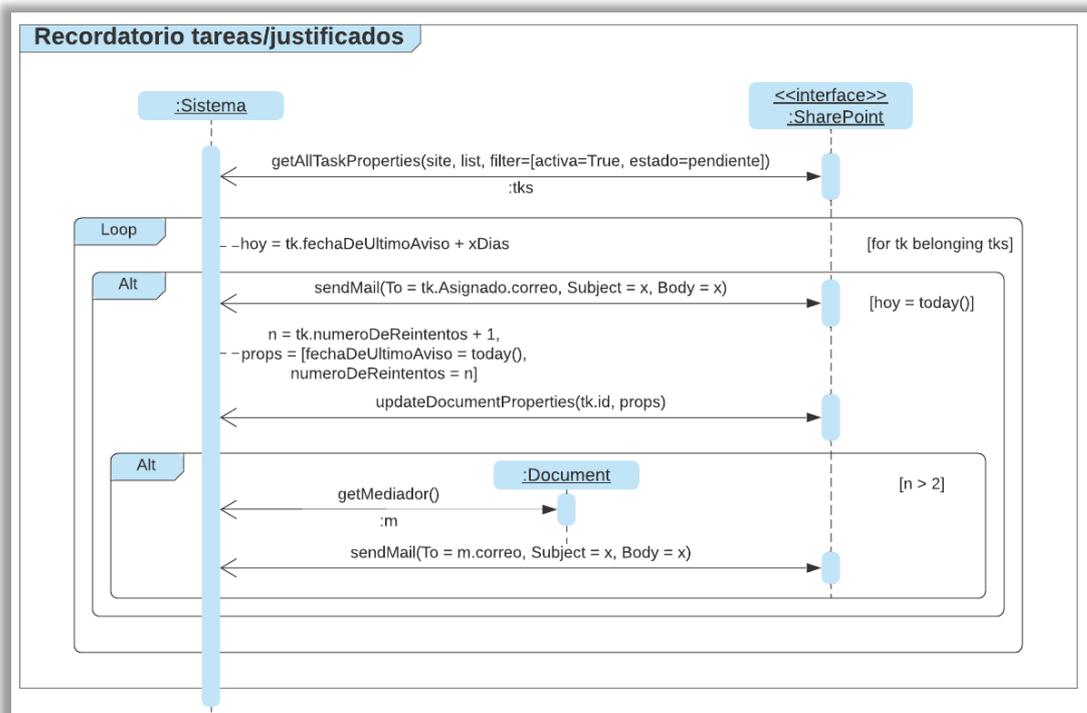


Figura 50. Diagrama de secuencia – Recordatorio tareas / justificados | Fuente: Elaboración propia

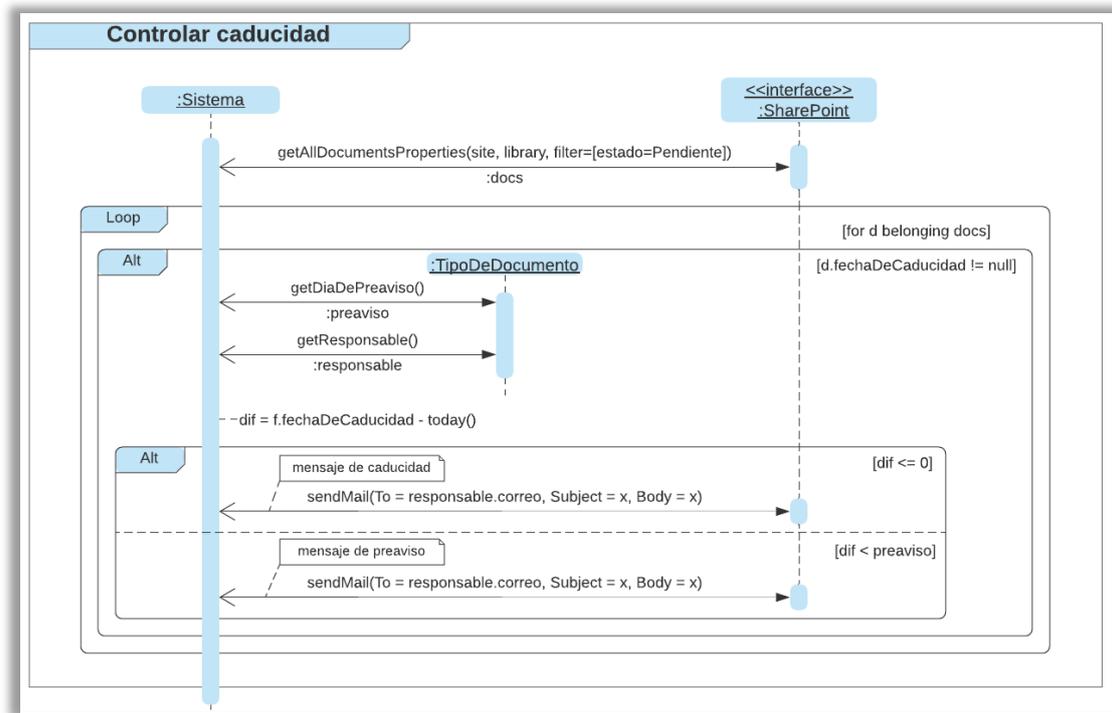


Figura 51. Diagrama de secuencia – Controlar caducidad | Fuente: Elaboración propia

7 Implementación

El capítulo siete se enfoca en la explicación de implementación del proyecto. En el primer apartado se explicará la creación y configuración de la aplicación web. Se detallarán los pasos realizados y los conceptos aplicados de la tecnología SharePoint para construir la plataforma web. En el siguiente apartado se explicarán las tecnologías, lenguajes de programación y herramientas de desarrollo que se han usado para personalizar las interfaces de usuario. Por último, se explicarán las implementaciones de los procesos de negocio para cumplir el funcionamiento completo del sistema.

7.1 Creación y configuración de la plataforma

La plataforma SharePoint Online está constituida por una serie de capas de niveles. En la capa superior tenemos la **web application** (mirar figuras 33 y 56). Esta es un servidor de Microsoft donde se encuentra desplegado el sitio web principal de SharePoint. Normalmente, la **web application** contiene el dominio o nombre principal de la empresa, también conocido como portal o página principal de SharePoint.

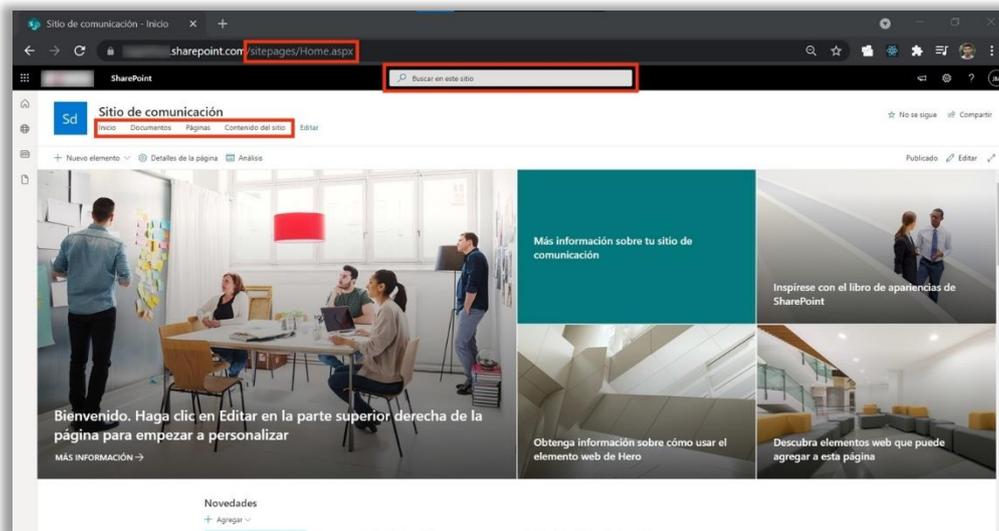


Figura 52. Portal de la organización | Fuente: SharePoint – Home page

La **web application** está formada por una **colección de sitios** (*site collection*), que es un repositorio para almacenar los sitios web de la organización en SharePoint. Por tanto, en este proyecto se creará una colección de sitios desde el centro de administración para la intranet tal que los usuarios puedan trabajar y comunicarse con diferentes miembros de la empresa. Es importante mencionar que, al momento de crear una colección de sitios, esta a su vez crea un sitio primario o de nivel superior para administrar los contenidos y presentarlos a los usuarios.

Un sitio web en SharePoint puede ser de dos tipos: **sitio de grupo** para conectar a personas de un mismo equipo o **sitio de comunicación** para llegar a un público amplio de toda la organización. Por tanto, para este proyecto es conveniente crear un sitio de grupo ya que proporcionará una ubicación única en la que un grupo de personas podrán

trabajar en un proyecto concreto y compartir la información desde cualquier lugar con cualquier dispositivo. Además, este tipo de grupo se orienta a grupos cerrados, es decir, la información se limita solo a los miembros del grupo o participantes específicos que tienen permisos de acceso, lo cual es ideal para la idea de este proyecto.

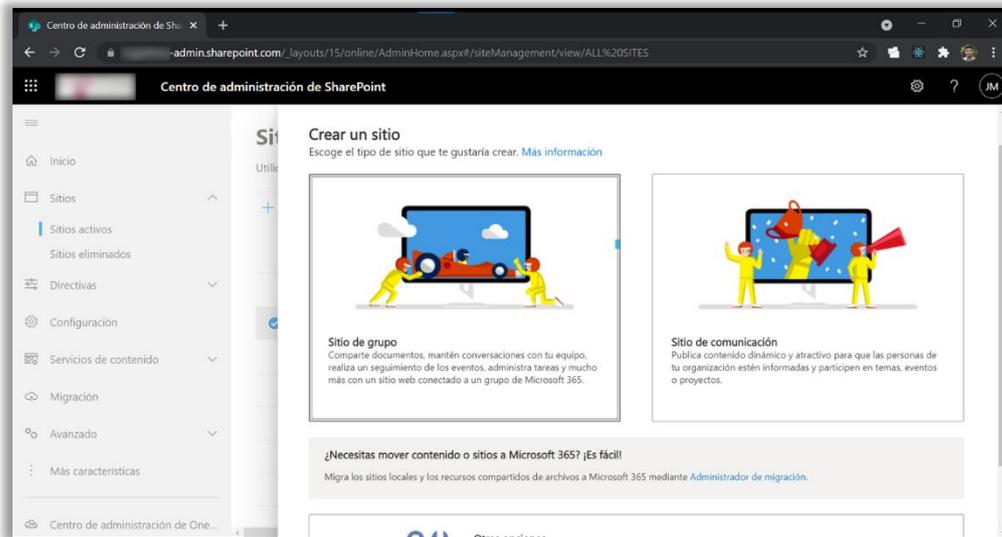


Figura 53. Tipos o plantillas de sitio | Fuente: SharePoint – Centro de administración

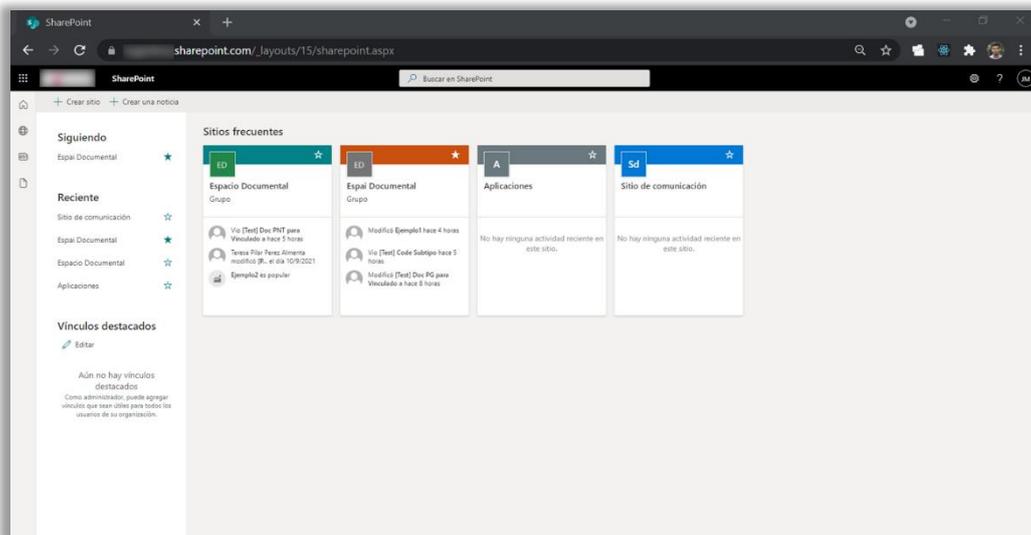


Figura 54. Estructura plana de sitios | Fuente: SharePoint – Página principal

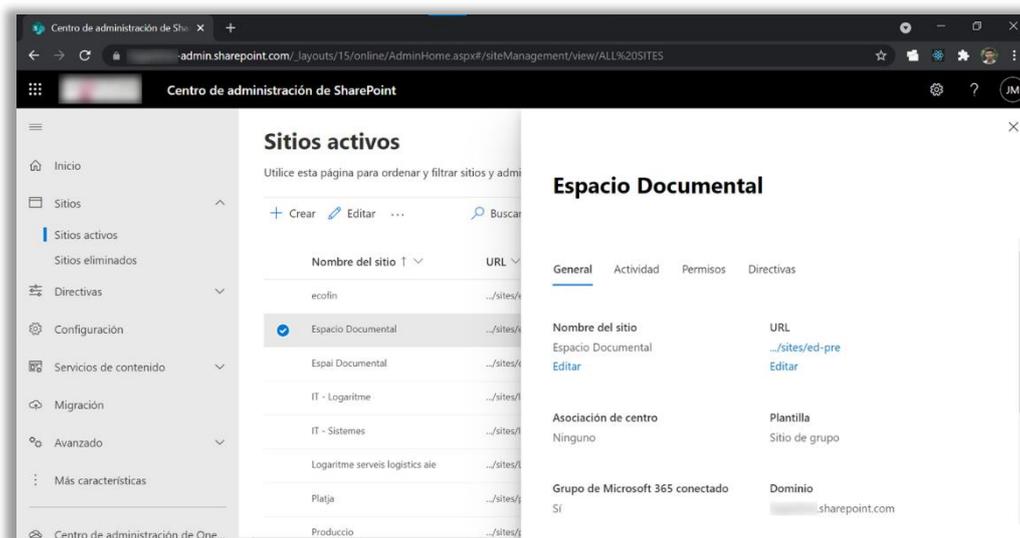


Figura 55. Información sobre el entorno de pre-producción | Fuente: SharePoint – Centro de administración

Un sitio de SharePoint puede tener varios subsitios, que son sitios creados dentro de otro sitio y estos subsitios heredan toda la configuración y diseño del sitio superior. Sin embargo, no será necesario el uso de este tipo de sitio para el proyecto ya que el objetivo es la creación de un sitio primario único para la empresa.

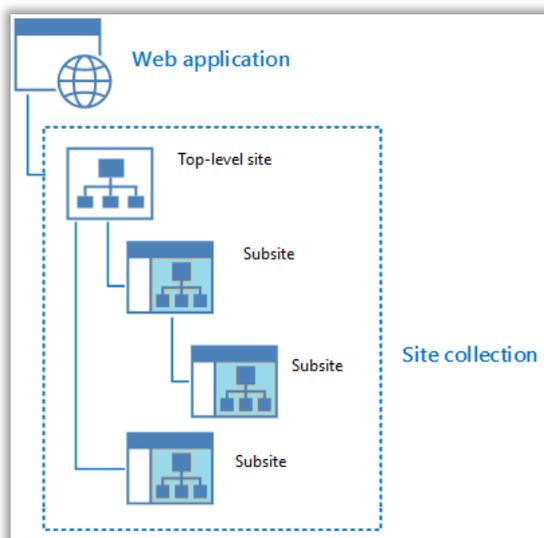


Figura 56. Estructura de SharePoint Online

Fuente: <https://www.sharepointdiary.com/2014/07/sharepoint-site-collection-vs-subsite-determine-which-one-to-create.html>

La plataforma web, que en SharePoint es un sitio, creada para el *software* de este proyecto necesitará consumir informaciones. Dichas informaciones se podrán guardar en una biblioteca de SharePoint si son datos estructurados o en una lista de SharePoint si son datos primitivos. Esta idea es equivalente a la creación de una aplicación web tradicional en la que se suele usar una base de datos de un servidor local o remoto para almacenar informaciones pertinentes del *software*.

En el sitio web principal se han creado dos bibliotecas de documentos: una para almacenar **documentos de trabajo** y otra para publicar los **documentos oficiales**.

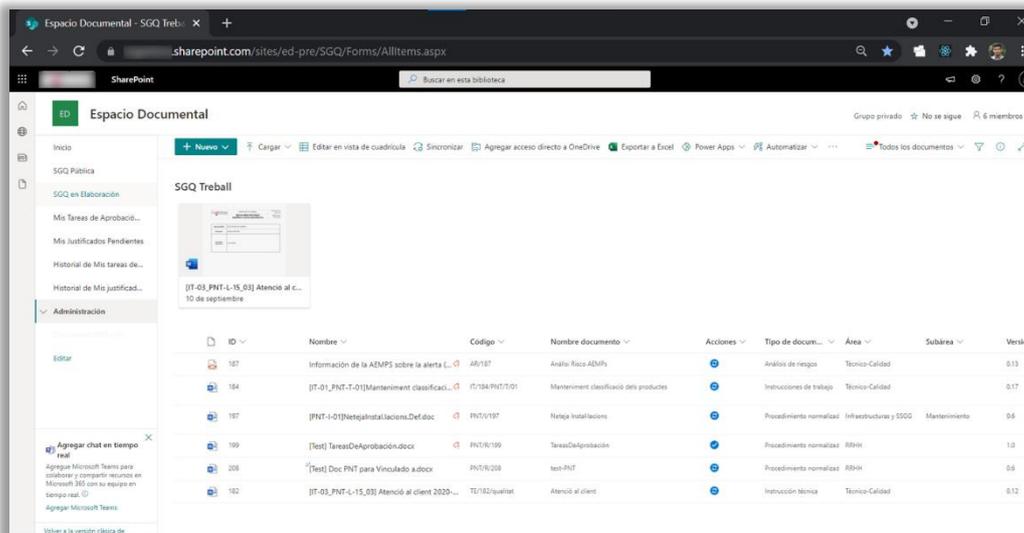


Figura 57. Biblioteca de trabajo | Fuente: SharePoint – Espacio Documental: SGQ de trabajo

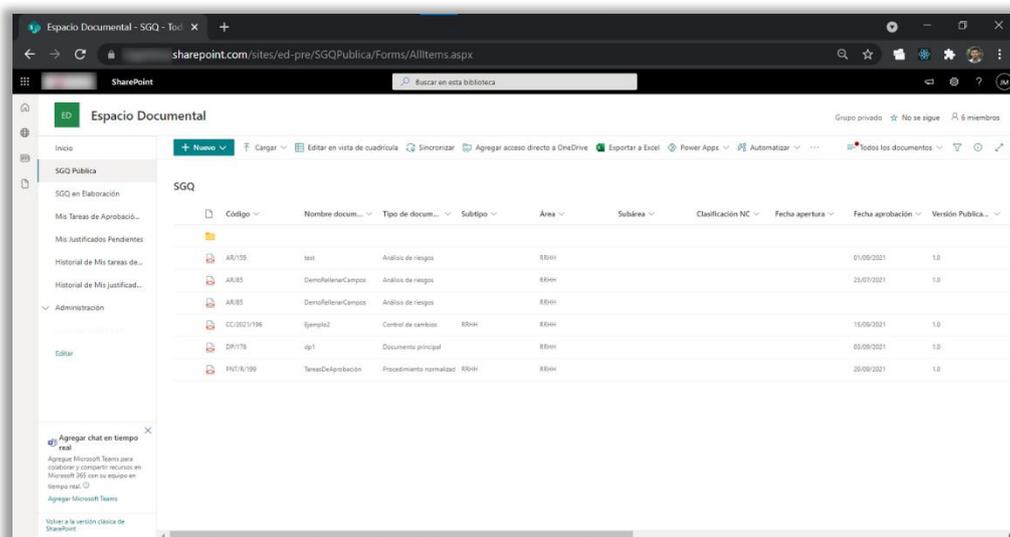


Figura 58. Biblioteca de documentos publicados | Fuente: SharePoint – Espacio Documental: SGQ pública

Además, se han creado tres listas personalizadas para la administración de datos, es decir, una primera lista para representar las informaciones de los once tipos de documentos, una segunda para guardar datos de las tareas de aprobación y una última para almacenar informaciones de las tareas de justificados.

ID	Tipo de docum...	Código	Califica (Afect)	Previo (Días)	Responsable	Agregar columna
1	Procedimiento general	PG	3	45	Quiliter Quiliter	
2	Procedimiento normaliz	PNT	3	45	Quiliter Quiliter	
3	Instrucción de trabajo	IT	3	30	Quiliter Quiliter	
4	Plantilla	PL	3	30	Quiliter Quiliter	
5	Manual	MA	3	45	Quiliter Quiliter	
6	Documento principal	DP	3	45	Quiliter Quiliter	
7	Análisis de riesgo	AR	1	30	Quiliter Quiliter	
8	No conformidad	NC	0	0	Quiliter Quiliter	
9	Acción correctiva	AC	0	0	Quiliter Quiliter	
10	Control de cambios	CC	0	0	Quiliter Quiliter	
11	Instrucción técnica	IT	4	30	Quiliter Quiliter	

Figura 59. Lista para tipos de documento
Fuente: SharePoint – Espacio Documental: tipos de documento

Título	Nombre documento	Versión Trabajo	Acción	Documento	Tipo Tarea	Acciones	Reasignar	Fecha de resp...
Nueva tarea de aprobación - PNT...	Ejemplo1			https://getmedia.share...	Aprobación			
Nueva tarea de edición - AC2021...	Ejemplo1			https://getmedia.share...	Edición			

Figura 60. Lista para tareas de aprobación
Fuente: SharePoint – Espacio Documental: tareas de aprobación

Título	Nombre documento	Documento	Acciones	Estado	Fecha de última actualización	Agregar columna
Nueva tarea de justificados - ACCI-15 ACCI...	Acci-Comercio1	https://getmedia.share...		Iniciada	26/10/2021	

Figura 61. Lista para tareas de justificados
Fuente: SharePoint – Espacio Documental: tareas de justificados

7.1.1 Elementos de un sitio

Sabemos que un *site* es uno de los pilares y componentes fundamentales de SharePoint. A continuación, se explicarán los elementos más importantes que debe tener un sitio de SharePoint. Notad que, existen tres elementos básicos que debe tener cualquier sitio de SharePoint: una página de inicio, enlaces de bienvenida y cuadro de búsqueda (mirar figura 52).

7.1.1.1 Elementos de navegación

SharePoint Online nos proporciona cuatro elementos de navegación básicos que deben estar disponibles en cualquier sitio, que son la barra de vínculos superior, la barra de inicio rápido, la barra de navegación y las migas de pan.

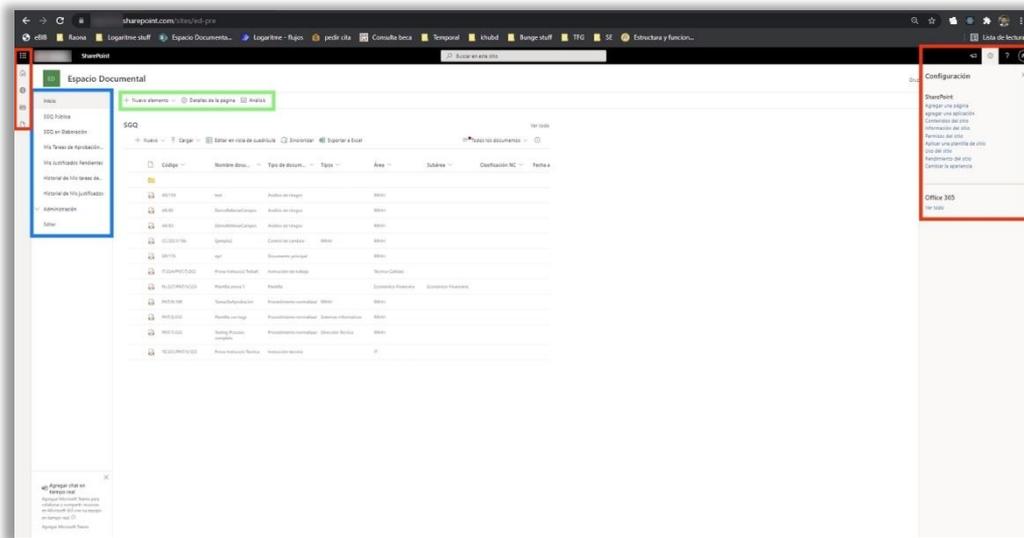


Figura 62. Elementos de la navegación | Fuente: SharePoint – Espacio Documental al: menú de navegación

En la figura 62, el color verde hace referencia a la barra de vínculos, que nos permite crear diferentes tipos de elementos en el sitio, pero generalmente su función es apuntar a otros sitios. En la parte izquierda, color azul, se encuentra la barra de inicio rápido, que tiene una serie de enlace para facilitar la navegación hacia un sitio o vista particular.

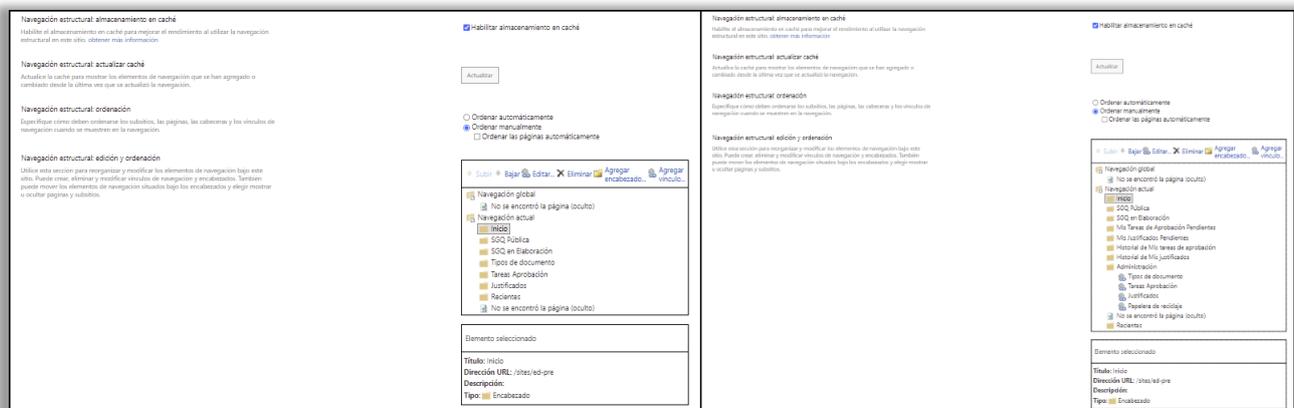


Figura 63. Configuración inicial y final de la barra de inicio rápido Fuente: SharePoint – Espacio Documental

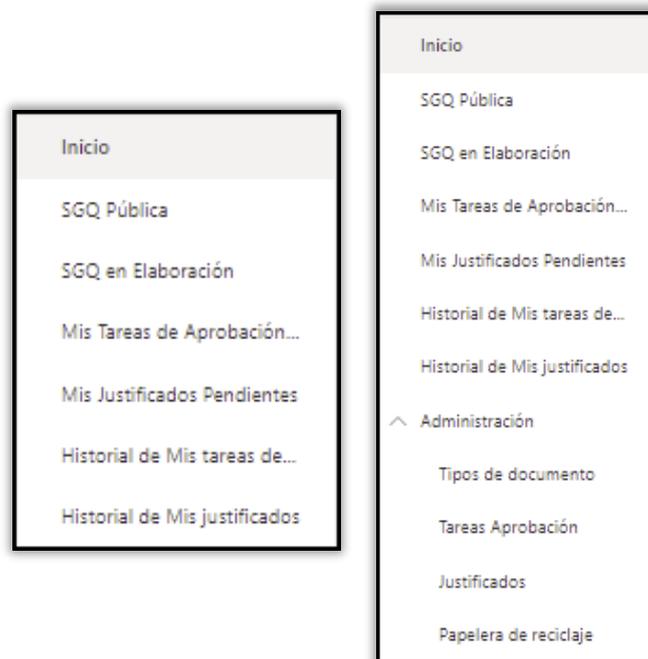


Figura 64. Vista de un usuario habitual y vista de un administrador
Fuente: SharePoint – Espacio Documental

En la figura 63, se puede observar las diferentes configuraciones, que se han llevado a cabo durante el proyecto, del menú de navegación. Y en la figura 64 se muestran los resultados obtenidos. Podemos observar que hay dos tipos de visualización:

- Una para los **usuarios habituales del sistema**, donde se puede destacar las secciones de las tareas de aprobación y justificados pendientes de cada tipo cliente, y un historial para consultar las tareas realizadas.
- Una segunda vista para los **administradores del sistema**, que engloba la vista anterior y añade un área llamado administrador para observar el registro del historial de todos los documentos.

Volviendo a la figura 62, se puede observar la barra de navegación en color rojo, que da acceso al contenido externo de Office 365, por ejemplo: correos o contactos. También, incorpora algunos botones como compartir, seguir, notificar y pedir ayuda. Sin embargo, el botón más importante es la rueda de navegación del sitio ya que a través de ella podemos acceder a los contenidos del sitio, permisos del sitio y la configuración del sitio, que son los tres enlaces más valiosos a la disposición de un usuario de SharePoint.

El cuarto y último elemento de la navegación hace referencia a las migas de pan, que indican dónde se encuentra el usuario actual y de donde proviene (mirar la figura 65). Estas migas de pan permiten navegar de un nivel hacia otro de una manera fácil y sencilla.



Figura 70. Propiedades de un tipo de contenido de sitio
Fuente: SharePoint – Espacio Documental



Figura 71. Plantilla PNT con sus propiedades correspondientes
Fuente: SharePoint – Espacio Documental

- almacén de términos para definir los valores de **taxonomía**, que representa la clasificación jerárquica de palabras, etiquetas o términos que se organizan en grupos en función de similitudes, y se pueden configurar la opción de multi idioma fácilmente.

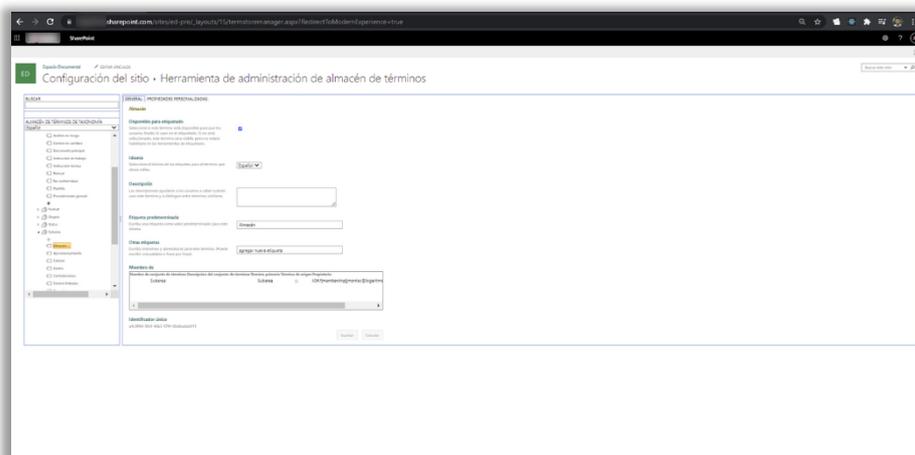


Figura 72. Almacén de términos de taxonomía
Fuente: SharePoint – Espacio Documental

7.1.1.3 Permisos del sitio

Los sitios de SharePoint Online requieren de una planificación de permisos para controlar el acceso de los usuarios a la plataforma, bibliotecas, listas, carpetas y elementos de sitio. Por defecto, SharePoint integra unos grupos de permisos estándar en cada sitio y son [49]:

- **Integrantes:** pueden contribuir o editar elementos de un sitio sin modificar su estructura.
- **Visitantes:** son aquellos usuarios que tienen solamente permisos de lectura.
- **Propietarios:** pueden modificar las configuraciones y los permisos del sitio, es decir tienen permisos de control total.

También existe la posibilidad de asignar los permisos a individuos, pero es más eficaz si se asignan los niveles de permisos a grupos en lugar de usuarios individuales. De hecho, Microsoft recomienda administrar los permisos a través de grupos. En el caso que una persona necesite permisos únicos, se deberá crear un grupo para ese individuo especial, además siempre que sea necesario habrá la posibilidad de añadir a más individuos en ese grupo. En resumen, siempre es mejor añadir los individuos en un grupo. En la figura 73, se puede observar los diferentes grupos existentes del proyecto donde cada grupo tiene asignado un nivel de permisos en concreto.

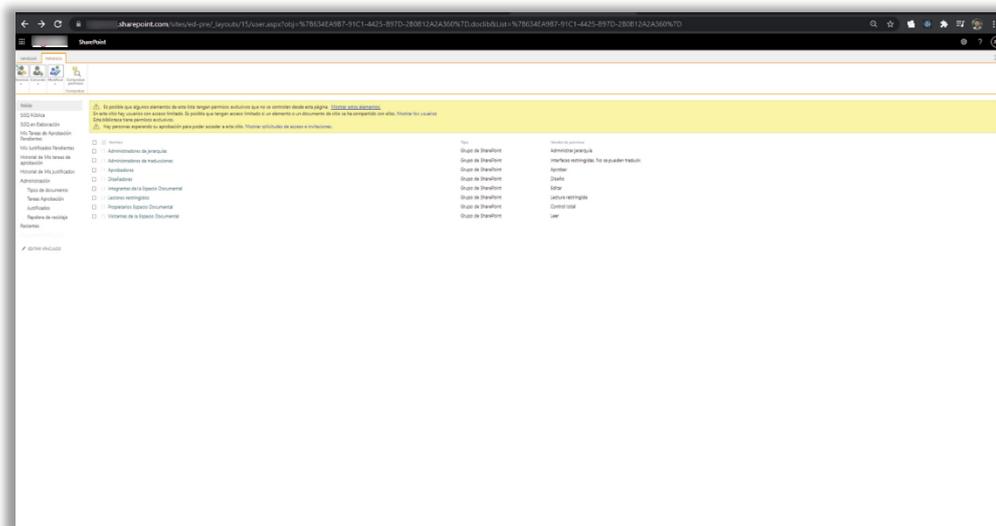


Figura 73. Configuración avanzada de permisos | Fuente: SharePoint – Espacio Documental

Es importante mencionar que existen dos tipos de grupos para administrar los permisos en SharePoint Online [49]:

- **Grupos de SharePoint:** son un contenedor para usuarios individuales o grupos a los que se les pueden asignar permisos, es la forma estándar de asignar los permisos en SharePoint. Sin embargo, estos grupos sólo se pueden usar en aquella colección de sitios en la que se han creado, es decir son grupos locales.
- **Grupos de seguridad de Active Directory:** son aquellos grupos que se pueden usar en cualquier colección de sitios de SharePoint y que tienen integrada la

seguridad de Microsoft. Estos grupos se pueden crear en Azure AD⁶ y se pueden sincronizar de forma local a través del conector Azure AD. Además, deben ser controlados y gestionados por el departamento de IT.

Por tanto, en este proyecto se hará uso de los grupos de AD para gestionar los grupos y/o para controlar el acceso de los recursos de la aplicación de una forma segura. También existe la posibilidad de usar los grupos de AD junto con los grupos de SharePoint, es decir, dentro de los grupos de SharePoint se incluirán los grupos de AD. De esta manera, se podrá aprovechar los beneficios de ambos grupos.

7.1.2 Limitaciones técnicas

Las limitaciones técnicas de SharePoint, que se han tenido en cuenta a la hora de implementar soluciones de este proyecto, son las siguientes [54]:

- **Límites de bases de datos de contenido:** se recomienda limitar el tamaño de las bases de datos de contenido a 200 GB para garantizar el rendimiento del sistema
- **Límites de listas y bibliotecas:** 5000 entradas es la cantidad máxima de elementos que debe tener una lista o biblioteca para procesar una operación de base de datos, por ejemplo: hacer una consulta a la base de datos para obtener elementos de una lista.
- **Límites de búsqueda:** una lista o biblioteca no puede contener más de 8 columnas de búsqueda para realizar una operación de consulta o actualización de elementos, ya que las columnas de búsqueda son unas columnas complejas, por ejemplo: columnas de tipo taxonomía, búsqueda o persona/grupo.

⁶ Es un servicio de gestión de identidad y de acceso que ayuda a los usuarios a identificarse y acceder a los recursos externos y/o internos del sistema.

7.2 SharePoint Framework con PnP

SharePoint Framework (SPFx) es un modelo de página y elemento web que proporciona el soporte completo para el desarrollo de SharePoint del lado del cliente. Se integra fácilmente con los datos de SharePoint y se usa para la personalización de este. Es decir, en lugar de utilizar los componentes por defecto de la plataforma SharePoint site, se puede diseñar UI o interfaces gráficas personalizables tal que se rendericen al lado del cliente.

Se hará uso de las **extensiones de SPFx** para ampliar la experiencia del usuario de SharePoint. Los componentes gráficos por personalizar son:

- botones de documentos y tareas para realizar una acción
- diálogo de mensaje para mantener informado
- diálogo de formulario para recoger datos

7.2.1 Extensión de SPFx

Actualmente, existen tres tipos de extensiones: conjuntos de comandos, personalizadores de aplicaciones y **personalizadores de campo**. Esta es la que se usará para modificar el estado de un documento o tarea a través de un botón.

Para implementar una solución personalizada se debe definir las herramientas para el desarrollo de SPFx, y Microsoft las clasifica en tres categorías:



Figura 74. Herramientas de desarrollo de SPFx | Fuente: [55]

Se usará el editor Visual Studio Code para manipular el código fuente de la extensión. En la categoría *crear procesos y herramientas* se usarán:

- **NodeJS/NPM:** para ejecutar el código de la extensión y para gestionar paquetes del proyecto.
- **Yeoman:** para generar la estructura del proyecto (*scaffolding*).

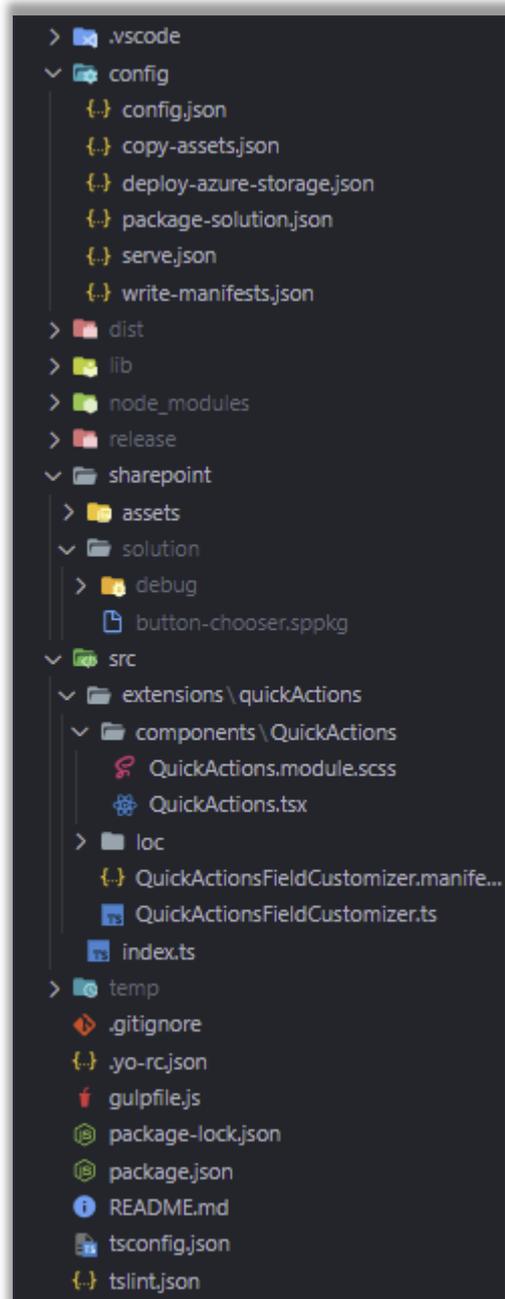


Figura 75. Estructura de la extensión | Fuente: Elaboración propia

- **TypeScript:** es un lenguaje de programación que se ejecuta en Node.js y es un superconjunto de JavaScript (JS). Se usará para dar soporte al marco de trabajo de JS, especialmente para definir tipos de variables y objetos.
- **Gulp:** es un *task runner* que sirve para compilar y ejecutar los procesos repetitivos. Por ejemplo: para comprobar el correcto funcionamiento de la extensión en una web.

```

'use strict';

const build = require('@microsoft/sp-build-web');

build.addSuppression(
  `Warning - [sass] The local CSS class 'ms-Grid' is not camelCase and will not be type-safe.`);

var getTasks = build.rig.getTasks;
build.rig.getTasks = function () {
  var result = getTasks.call(build.rig);

  result.set('serve', result.get('serve-deprecated'));

  return result;
};

build.initialize(require('gulp'));

```

Figura 76. Gulp – Task runner serve | Fuente: Elaboración propia

En la categoría *JS framework* se deberá usar un marco de trabajo del lado del cliente, y se usará **React**. Para actualizar el estado de SharePoint se usarán las utilidades predefinidas de la librería **PnP** [24], que proporciona una buena documentación y tiene apoyo de la comunidad. Y para crear algunos componentes gráficos atractivos con estilos y *responsive* predefinidos, se usará la colección de *UX framework* llamada **Office UI Fabric** [56].

```

public render(): React.ReactElement<{}> {
  let renderButtons : JSX.Element = null;
  let addRGBButtons : JSX.Element = null;
  let addRejectButton : JSX.Element = null;
  let renderDialogForm : JSX.Element = null;
  let renderDialogMessage : JSX.Element = null;
  let renderDocumentStateButton : JSX.Element = null;
  let renderViewButton : JSX.Element = null;
  let renderComment : JSX.Element = null;

  let isList : boolean = this.props.listName === listJustified || this.props.listName === listApproval;
  let isEditorORApprover = this.props.contentTypeName === contentTypeEA;
  let isMediator : boolean = this.props.contentTypeName === contentTypeM;

  addRGBButtons = <span>
    <IconButton iconProps={this.state.isReassigned ? reassignedIcon : resignIcon} style={{ color: "#0078D4" }} title='Reasignar'
      ariaLabel='Reasignar' onClick={() => this.reassingEvent()}></IconButton>

    <IconButton iconProps={this.state.isGoApprovalStep ? goneIcon : goIcon} style={{ color: "#0078D4" }} title='Ir a paso de aprobación'
      ariaLabel='Ir a paso de aprobación' onClick={() => this.goApprovalStepEvent()}></IconButton>
  </span>;

  addRejectButton = <span>
    <IconButton iconProps={this.state.isRejected ? rejectedIcon : rejectIcon} style={{ color: "#0078D4" }} title='Rechazar'
      ariaLabel='Rechazar' onClick={() => this.rejectEvent()}></IconButton>
  </span>;

  renderButtons = <div className={styles.QuickActions}>
    <div className={styles.iconsCell}>
      <Stack tokens={{ childrenGap: 8 }} horizontal>
        <div>
          <div>
            <IconButton iconProps={this.state.isApproved ? approvedIcon : approveIcon} style={{ color: "#0078D4" }} title='Aprobar'
              ariaLabel='Aprobar' onClick={() => this.approveEvent()}></IconButton>

            {(this.props.listName !== listJustified) ? addRejectButton : null}

            {(isList && isMediator) ? addRGBButtons : null}
          </div>
        </div>
      </Stack>
    </div>
  </div>;

```

```

renderDialogForm = <Dialog
  hidden={!this.state.openDialogForm}
  onDismiss={() => {this.setState({ openDialogForm: false}); if(!isEditorORApprover) this.showSpecificMessage(); }}
  dialogContentProps={{ title: this.state.titleForm, subText: this.state.subtextForm}}

  <TextField label="" multiline rows={2} onChange={(event: React.FormEvent<HTMLInputElement | HTMLTextAreaElement>,
    newValue: string) => this.updateSubmittedText(event, newValue)} />
  <DialogFooter>
    <PrimaryButton text='Enviar' onClick={() => this.checkSubmittedText()} />
    {this.props.listName !== publicLibrary && <DefaultButton text='Cancelar'
      onClick={() => {this.setState({ openDialogForm: false}); if(!isEditorORApprover) this.showSpecificMessage();}} />}
  </DialogFooter>
</Dialog>;

renderDialogMessage = <Dialog
  hidden={!this.state.openDialogMessage}
  onDismiss={() => this.setState({ openDialogMessage: false})}
  dialogContentProps={{ title: this.state.titleMessage, subText: this.state.subtextMessage }}

  <DialogFooter>
    <DefaultButton text='v' onClick={() => {
      (this.props.listName === mainLibrary) ? this.checkDialogAnswer() : this.setState({ openDialogMessage: false});}}
    />
    {(this.props.listName === mainLibrary) ? <DefaultButton text='Cancelar' onClick={() =>
      this.setState({ openDialogMessage: false })} /> : null}
  </DialogFooter>
</Dialog>;

renderDocumentStateButton = <div className={styles.QuickActions}>
  <div className={styles.iconsCell}>
    <Stack tokens={{ childrenGap: 8 }} horizontal>
      <div>
        <IconButton iconProps={{(this.state.documentState === draftedTermName || this.state.documentState === draftedError)
          ? draftedDocumentIcon : (this.state.documentState === elaboratedTermName)
          ? elaboratedDocumentIcon : (this.state.documentState === publishedTermName ||
            this.state.documentState === publishedError)
          ? publishedDocumentIcon : null }}
          style={{ color: "#0078D4" }}
          title={{(this.state.documentState === draftedError) ? draftedTermName : this.state.documentState}
            ariaLabel={{(this.state.documentState === draftedError) ? draftedTermName : this.state.documentState}
            onClick={() => this.controlDocumentState()}}></IconButton>
      </div>
    </Stack>
  </div>
</div>;

renderViewButton = <div className={styles.iconsCell}>
  <Stack tokens={{ childrenGap: 8 }} horizontal>
    <IconButton iconProps={viewIcon} style={{ color: "#0078D4" }} title='Ver' ariaLabel='Ver'></IconButton>
  </Stack>
</div>;

renderComment = <div className={styles.QuickActions}>
  <div className={styles.iconsCell}>
    <Stack tokens={{ childrenGap: 8 }} horizontal>
      <div>
        <IconButton iconProps={commentIcon}
          style={{ color: "#0078D4" }}
          title="Comentar" ariaLabel="Comentar"
          onClick={() => this.SendComment()}></IconButton>
      </div>
    </Stack>
  </div>
</div>;

if (!this.state.isLoading) {
  if(isList && this.state.isEditor){
    return (
      <div>
        {renderButtons}
        {renderDialogForm}
        {renderDialogMessage}
      </div>
    );
  }
  else if(this.props.listName === mainLibrary && this.state.isEditor){
    return (
      <div>
        {renderDocumentStateButton}
        {renderDialogMessage}
      </div>
    );
  }
  else if(this.props.listName === publicLibrary){
    return(
      <div>
        {renderComment}
        {renderDialogForm}
        {renderDialogMessage}
      </div>
    );
  }
}

return (
  <div className={styles.QuickActions}>
    {renderViewButton}
  </div>
);

```

Figura 77. Método render para pintar componentes de la extensión | Fuente: Elaboración propia

```
private async controlDocumentState(): Promise<void> {
    const {ed_Valid} = await sp.web.lists.getByTitle(this.props.listName).items.getById(+this.props.id).get();

    if(ed_Valid){
        if(this.state.documentState === draftedTermName || this.state.documentState === draftedError){
            this.checkDocumentFields();
        }
        else if(this.state.documentState === elaboratedTermName)
            this.openDialogWithMessage('Documento en elaboración', 'El documento se encuentra en el proceso de elaboración.');
```

Figura 78. Método que controla estado de un documento | Fuente: Elaboración propia

7.2.2 Depuración y publicación de la extensión de SPFx

Para depurar el código en el área de trabajo de la plataforma se usará la herramienta **SP Editor** [57]. Esta permite definir *breakpoints* con el objetivo de comprobar el correcto funcionamiento de la extensión.

Para desplegar la solución definitiva en la plataforma de SharePoint se ha de generar un paquete de extensión. El *task runner gulp* es capaz de empaquetar todos los códigos fuentes del proyecto en un único archivo (fase de empaquetamiento). Para ello debemos aplicar los siguientes comandos:

```
$ gulp clean
$ gulp bundle --ship
$ gulp package-solution --ship
```

Figura 79. Comandos para generar paquete de deployment | Fuente: Elaboración propia

Posteriormente, dicho paquete generado por *gulp* se deberá publicar en el catálogo de aplicación de la plataforma.

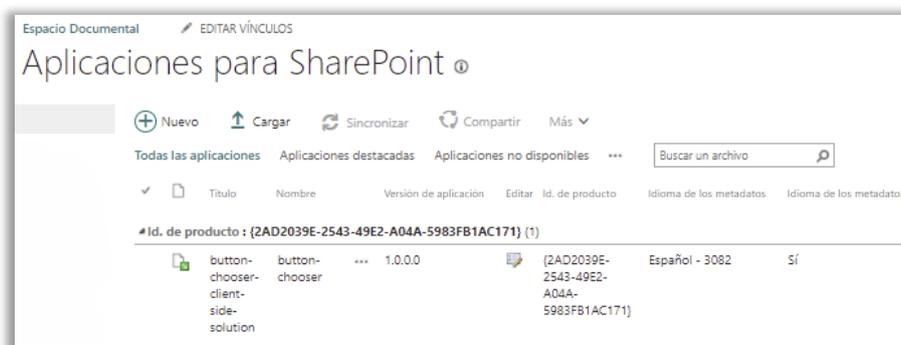


Figura 80. SPFx deployment en app catalog | Fuente: Elaboración propia

A continuación, se muestran algunos resultados de las interfaces en la plataforma web de SharePoint:

Código ▾	Nombre documento ▾	Acciones ▾	Estado Documento ▾
AR-xyz	Doc nombre		En elaboración
	PNT Expediciones		Draft
PNT/F/3	Lista de contactos		Publicado

Figura 81. Botones para indicar acciones de un documento

Código ▾	Nombre documento ▾	Acciones ▾
AR-xyz	Doc nombre	
	PNT Expediciones	
PNT/F/3	Lista de contactos	
PNT/R/2	Procediment Normalitzat de Treball	

Documento en draft

¿Desea que el estado del documento pase de draft a elaboración?

Figura 82. Popup para iniciar proceso de publicación

Editor	Realizada	
Aprobador	Realizada	
Aprobador	Rechazada	
Mediador	Pendiente	Aprobar

Figura 83. Botones para indicar acciones de una tarea

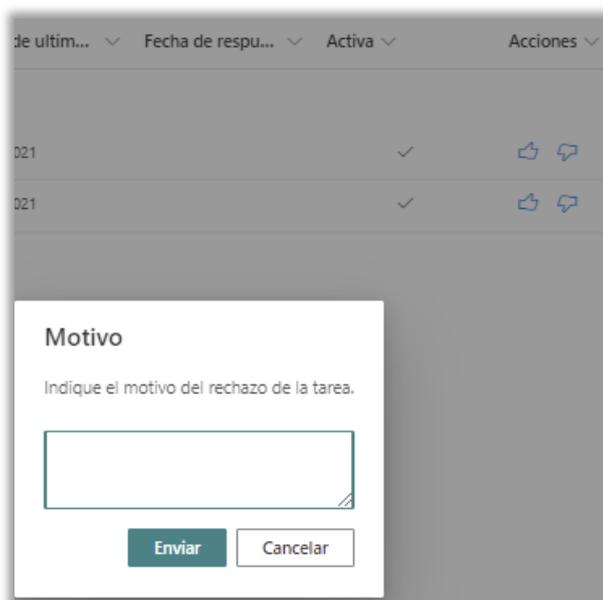


Figura 84. Formulario para indicar el motivo del rechazo

7.3 Power Automate

Como ya se había comentado en el capítulo anterior, concretamente en la sección [6.3.3](#), de que Power Automate integra más de 300 conectores predefinidos de Microsoft para facilitar las operaciones de negocio de la plataforma SharePoint. Sin embargo, en este proyecto se usarán los siguientes conectores:

- **SharePoint:** para consultar, actualizar y eliminar informaciones de los documentos y elementos de la lista.
- **Azure AD:** para obtener informaciones de los usuarios o grupos AD del sistema
- **Office 365 Outlook:** para enviar notificaciones.
- **Office 365 Users:** para obtener información de los usuarios del sistema.
- **OneDrive:** para sincronizar el contenido y la conversión de documentos.
- **Word Online:** para agregar etiquetas y actualizar contenidos de un documento Word.

7.3.1 Flujos

Un flujo de trabajo (*workflow*) es un conjunto de actividades relacionadas que siguen un orden lógico para solucionar un problema. A la hora de crear un flujo en Power Automate es necesario indicar la categoría del flujo. Existen tres tipos de flujos [58]:

- **Flujos automatizados:** se desencadenan automáticamente por un evento, por ejemplo: cuando modifica el estado de un documento.
- **Flujos instantáneos:** se ejecutan en un momento instantáneo, por ejemplo: enviar un correo electrónico a un destinatario al dar clic en el botón “enviar”.
- **Flujos programados:** se desencadenan en una fecha y hora programada, por ejemplo: enviar recordatorios de tareas pendientes cada cinco días.

Una vez creado un flujo se puede realizar las siguientes acciones:

- Implementar / probar (manual o automáticamente) / corregir errores
- Activar o desactivar flujo
- Ver y filtrar ejecuciones de flujo
- Inspeccionar códigos del flujo

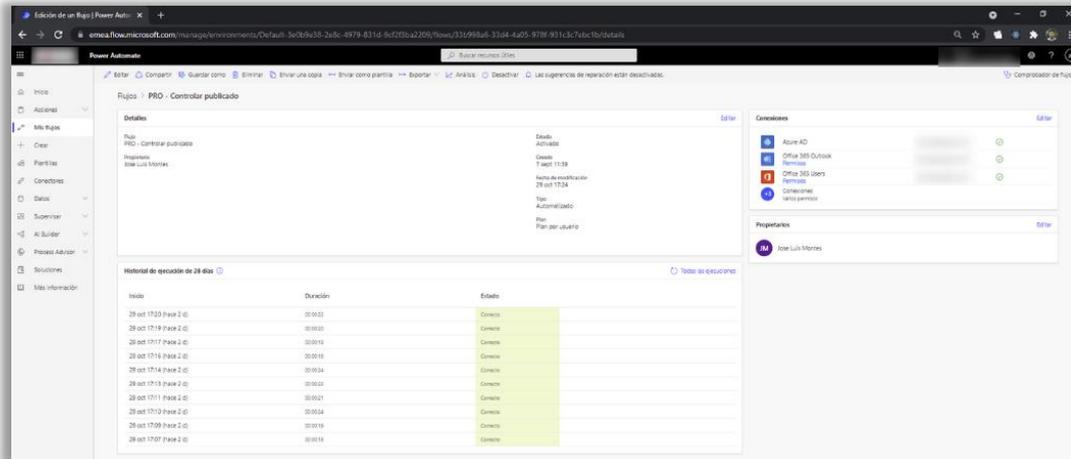


Figura 85. Opciones de un flujo | Fuente: Power Automate – Edición de un flujo

A continuación, se muestran algunos flujos implementados con Power Automate de este proyecto.

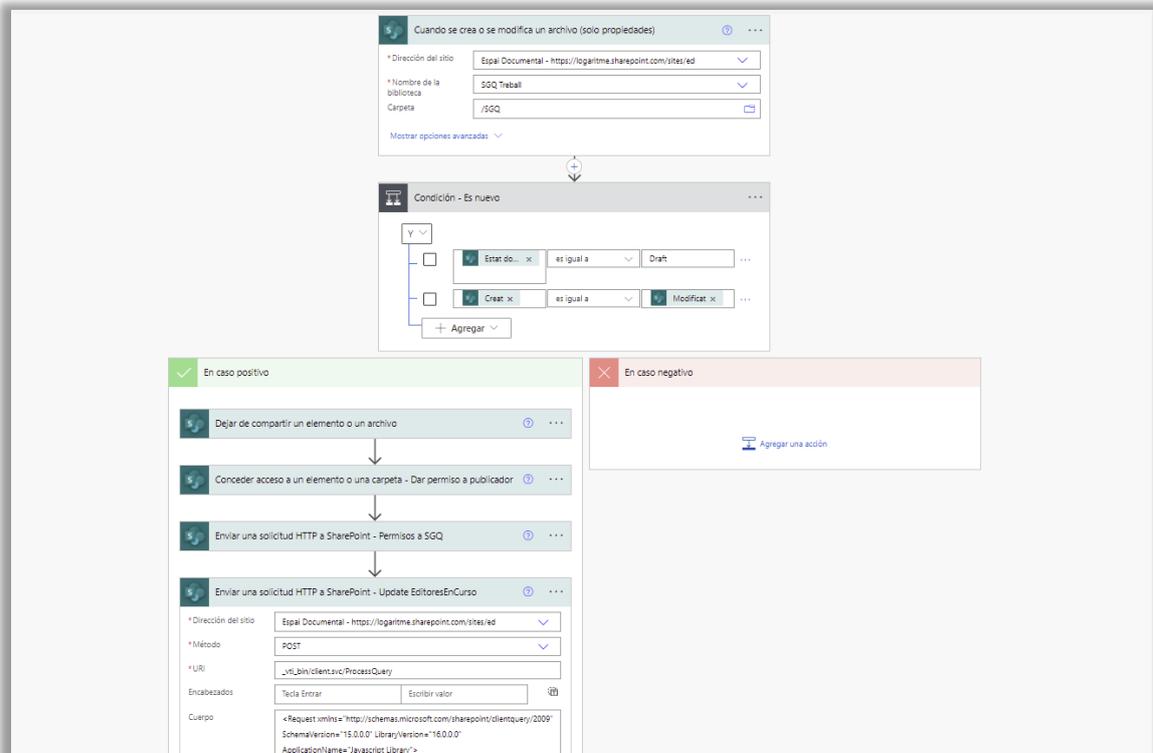


Figura 86. Flujo: On Init | Fuente: Power Automate – Flujos

La figura 86 implementa un flujo automatizado que se ejecuta cuando se carga un documento en la biblioteca de trabajo. El objetivo es asignar los valores iniciales para iniciar el proceso de publicación correctamente, es decir, aplica el diseño de la figura 62.

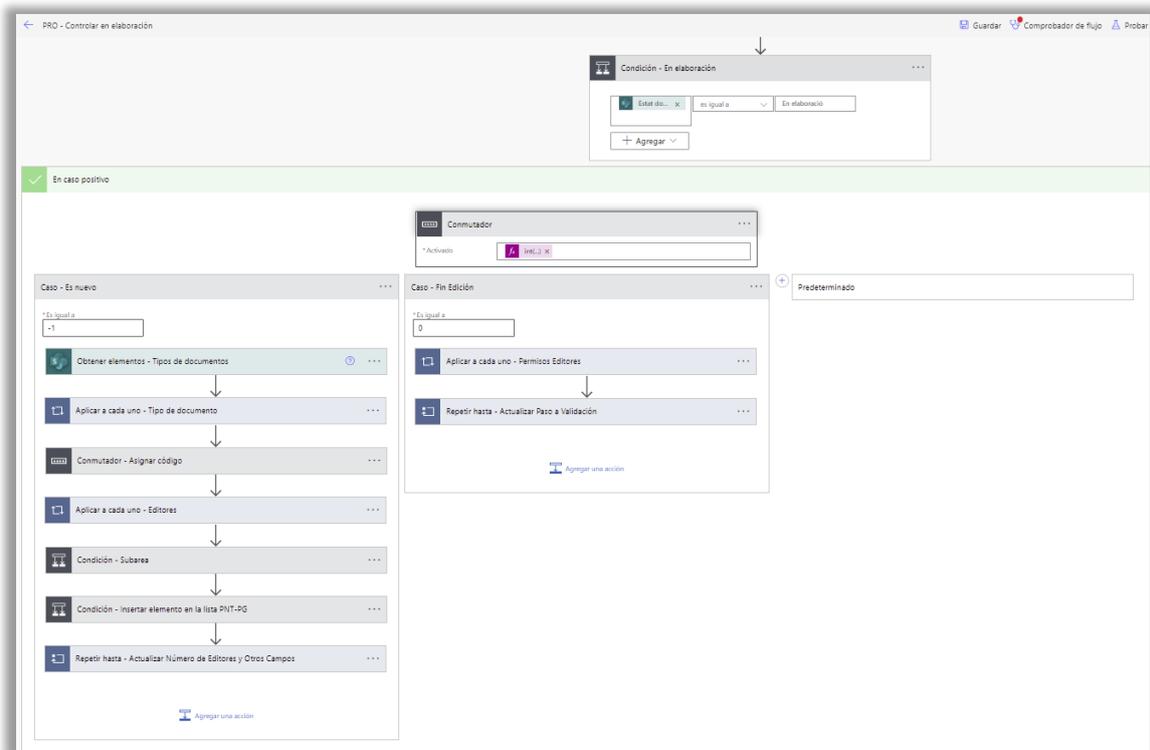


Figura 87. Flujo: Controlar en elaboración | Fuente: Power Automate – Flujos

La figura 87 es un flujo automatizado que se desencadena cuando hay un cambio en las propiedades de un documento perteneciente en la biblioteca de trabajo, concretamente cuando el responsable del documento indica enviar el documento de draft hacia el proceso de elaboración entonces el flujo realizará las acciones explicadas del diagrama de proceso de edición (mirar las figuras 20 y 65). Además, se encargará de actualizar los campos de la documentación, por ejemplo, la codificación del documento acorde a la topología de documentos (mirar la sección 5.1.1.2), área, subárea, etc.

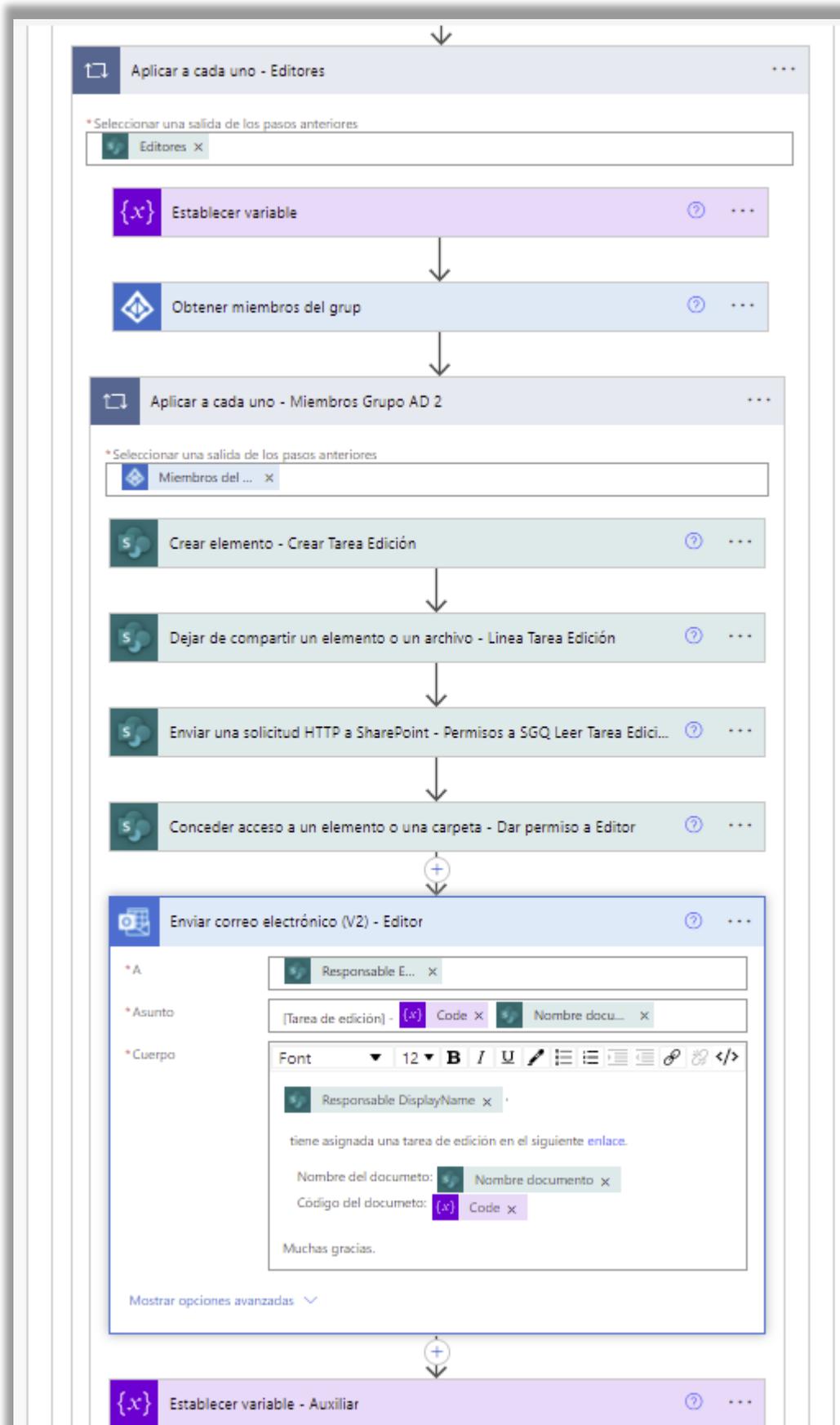


Figura 88. Creación y notificación de una tarea de edición | Fuente: Power Automate – Flujos

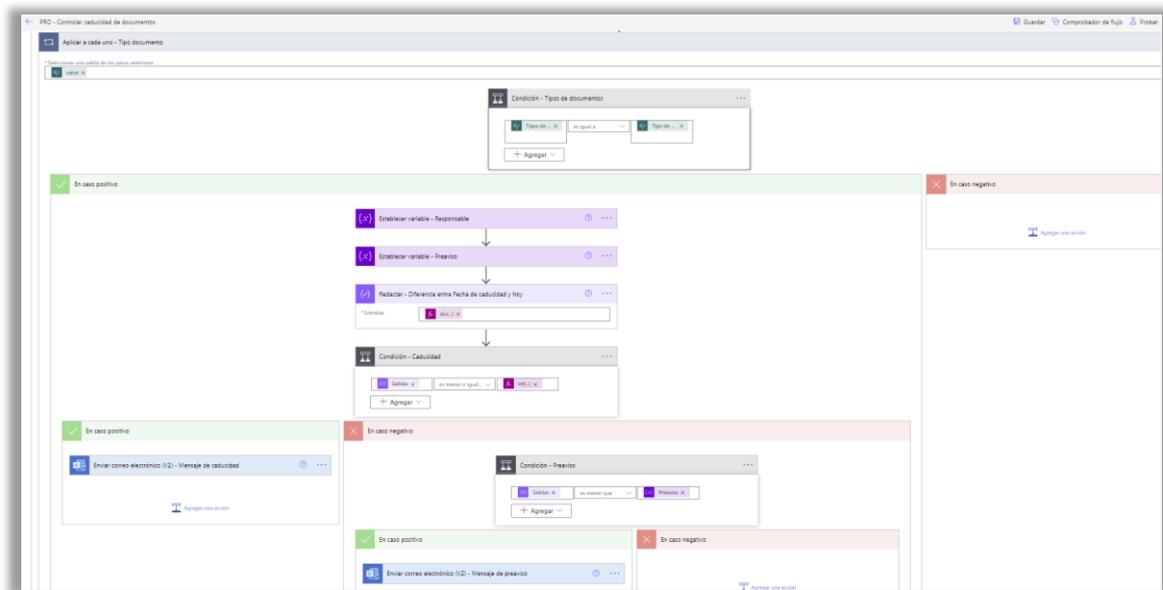


Figura 89. Flujo: Controlar caducidad de documentos | Fuente: Power Automate – Flujos

La figura 89 implementa un flujo programado que se ejecuta cada día para comprobar si existen documentos publicados que están caducados o que se acercan a la fecha de caducidad. Es decir, aplica el diagrama secuencial de la figura [73](#).

8 Pruebas

En este capítulo se explicarán las actividades de pruebas o *testing* realizadas en este proyecto para verificar y validar las funcionalidades implementadas del *software*. Estas pruebas efectuadas se pueden clasificar en dos categorías: *manual or automatic tests* y *User Acceptance Testing (UAT)*.

8.1 Pruebas de flujo

Una prueba de flujo en Power Automate es aquella herramienta que se encarga de verificar los procesos de negocio respecto a errores básicos y detectar errores que ocurren cuando se ejecuta el flujo automatizado. Se puede ejecutar de dos maneras: manual o automática.

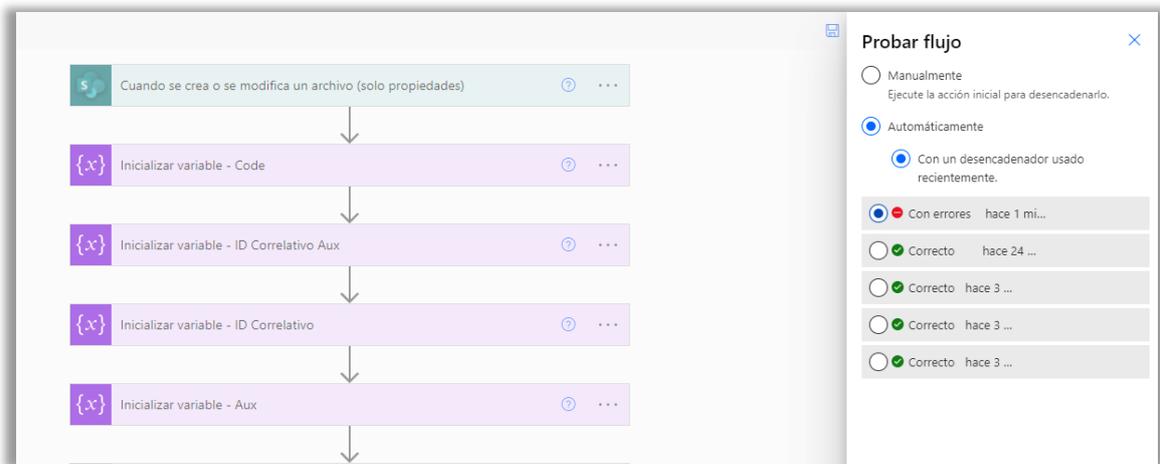


Figura 90. Pruebas de flujo | Fuente: Power Automate

Una vez completada la configuración del proceso automatizado, se deberá ejecutar el comprobador de flujos para detectar los posibles fallos y promover flujos de calidad ya que garantiza que se sigan los procedimientos recomendados al diseñar procesos. Es decir, el comprobador mostrará una guía detallada por cada problema o error que identifique, y posteriormente se deberá corregir y aplicar las mejoras (mirar figura 91).

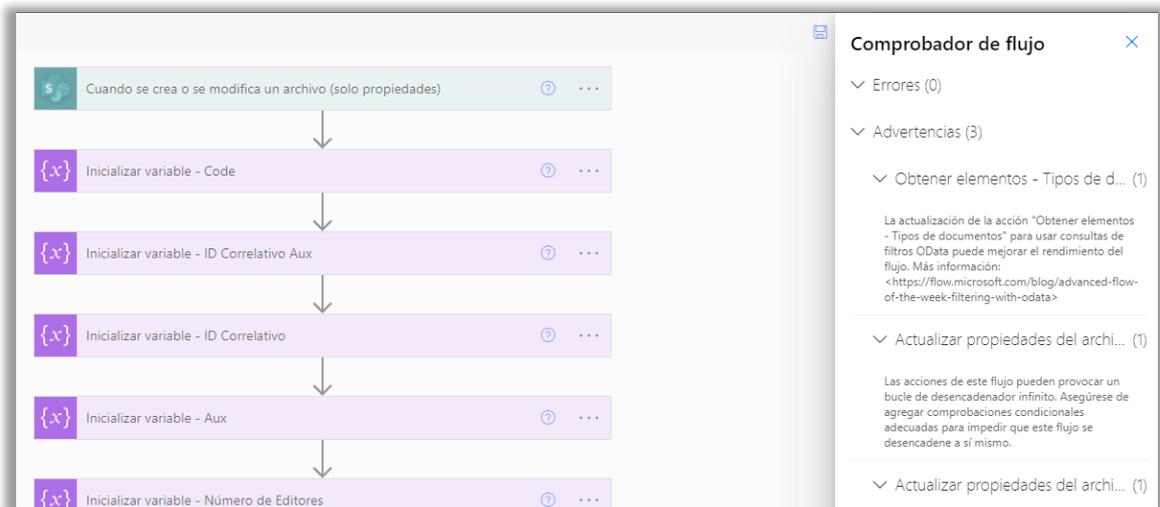


Figura 91. Comprobador de flujo | Fuente: Power Automate

También existe la posibilidad de añadir *breakpoints* y tomar decisiones alternativas en el caso que haya errores en el flujo.

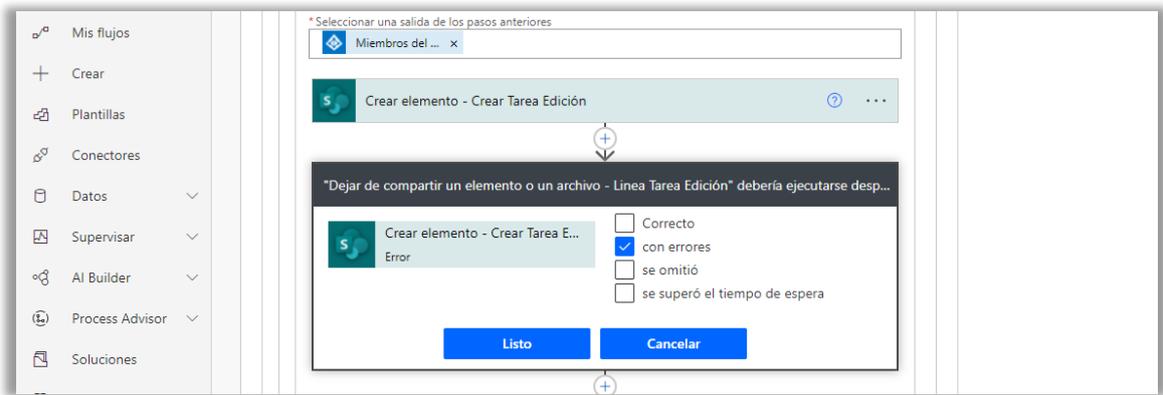


Figura 92. Breakpoint | Fuente: Power Automate

8.2 Pruebas de aceptación del usuario

La prueba de aceptación del usuario (UAT) es una actividad de verificación que se lleva a cabo para asegurar el cumplimiento de los objetivos del producto.

En el apartado [1.6](#) se describió la metodología ágil que se iba a aplicar en este proyecto, y la revisión del software con el cliente al final de cada *sprint* para demostrar el correcto funcionamiento de la aplicación, corregir los posibles errores y desviaciones.

En este proyecto se han realizado cuatro sesiones de UAT para validar las siguientes funciones:

- **UAT 1:** estructuración de documentos
- **UAT 2:** control del proceso de publicación
- **UAT 3:** procesos de negocio, seguridad y control de acceso
- **UAT 4:** fase de validación, migración de plataforma y soluciones

En cada reunión se presentan los escenarios o situaciones hipotéticas que los usuarios pueden encontrar al interactuar con la aplicación. El objetivo es guiar al cliente a través de posibles funcionalidades de uso del sistema. Para más detalles mirar la sección [11.2](#).

9 Despliegue

Una vez validado el proyecto de SharePoint con el cliente, se procede la creación del entorno de producción para desplegar las soluciones del *software* enfocadas a los usuarios finales.

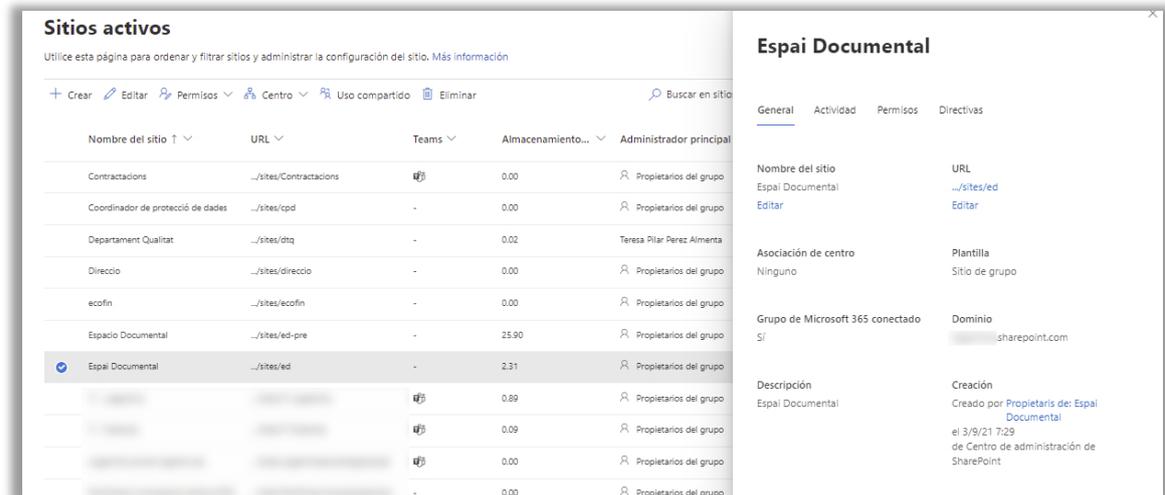


Figura 93. Información sobre el entorno de producción | Fuente: SharePoint – Centro de administración

Posteriormente, para migrar la estructura, contenidos, permisos y soluciones de negocio de la plataforma del entorno pre-producción a la producción, se ha usado la herramienta ShareGate [59].

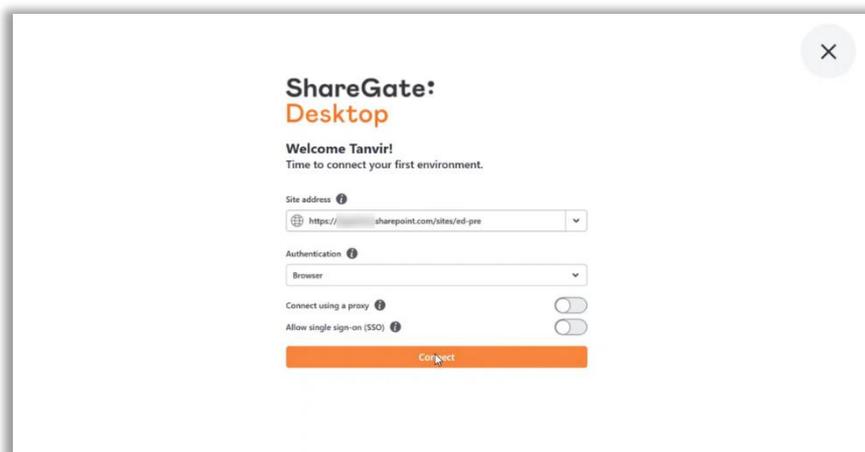


Figura 94. ShareGate | Fuente: ShareGate – Desktop

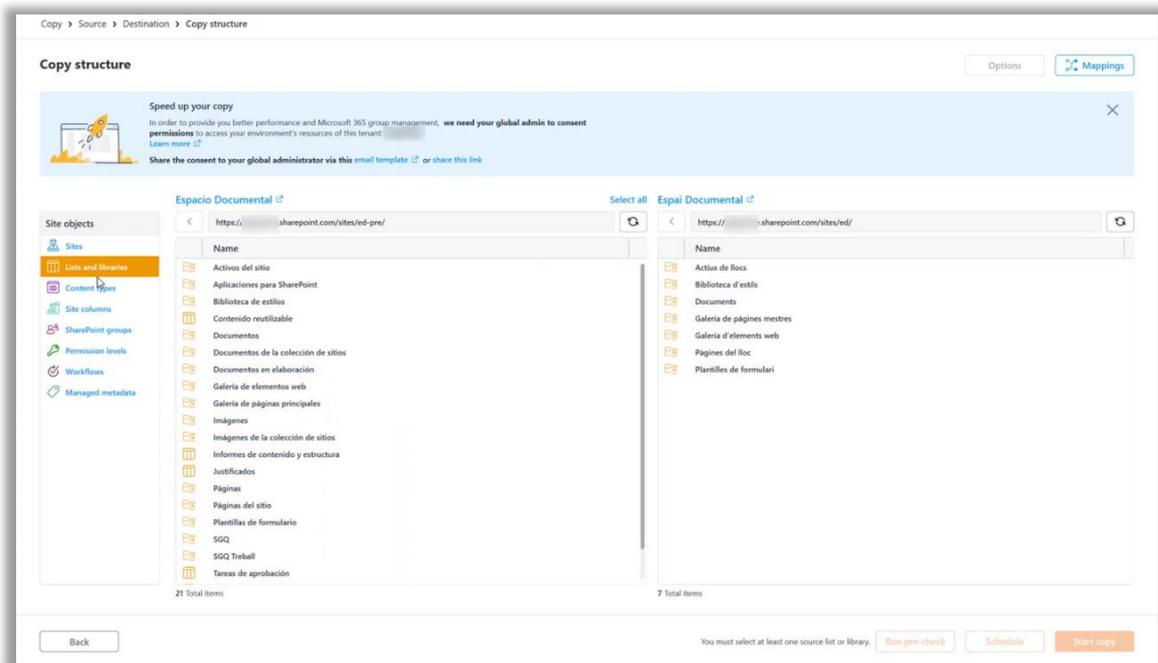


Figura 95. Migración de estructura de SharePoint | Fuente: ShareGate – Desktop

Status	Date	Type	Title	Destination site name	Destination list title	Destination path	Source version	Details
✓	9/3/2021 4:48:55 PM	SharePoint group	SGQ	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:27 PM	Role	Aprobar	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:27 PM	Role	Interfases restringidas. No...	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:27 PM	Role	Administrar jerarquia	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:26 PM	Role	Lectura restringida	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:24 PM	SharePoint group	Administradores de jerarqu...	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:24 PM	SharePoint group	Lectores restringidos	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:24 PM	SharePoint group	Aprobadores	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:24 PM	SharePoint group	Diseñadores	Espai Documental				Copied successfully.
✓	9/3/2021 4:48:24 PM	SharePoint group	Administradores de treduc...	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:46 PM	Site content type	Instrucciones de trabajo	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:45 PM	Site content type	Control de cambios	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:45 PM	Site content type	Análisis de riesgos	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:45 PM	Site content type	Procedimientos Generales	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:45 PM	Site content type	Plantilla	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:04 PM	Site content type	Instrucción técnica	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:04 PM	Site content type	No conformidad	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:03 PM	Site content type	Manuales	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:03 PM	Site content type	Documento principal	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:03 PM	Site content type	Procedimiento normalizad...	Espai Documental				Copied successfully.
✓	9/3/2021 4:44:03 PM	Site content type	Acciones correctivas	Espai Documental				Copied successfully.
✓	9/3/2021 4:43:31 PM	Site column	Estado Documento	Espai Documental				Copied successfully.
✓	9/3/2021 4:43:27 PM	Site column	Estado Abierto	Espai Documental				Copied successfully.
✓	9/3/2021 4:43:10 PM	Site column	Clasificación NC	Espai Documental				Copied successfully.
✓	9/3/2021 4:42:59 PM	Site column	Subtipo	Espai Documental				Copied successfully.

Figura 96. Resultado de migración | Fuente: ShareGate – Desktop

Finalmente, la solución empaquetada de SPFx se ha desplegado en el catálogo de la aplicación del entorno PRO.

10 Aspectos legales

El capítulo diez trata de considerar los aspectos legales aplicables al proyecto. Es decir, las leyes que pueden afectar y las licencias de software que deben tenerse en cuenta.

10.1 Leyes aplicables al proyecto

Una ley aplicable, para garantizar los derechos y el tratamiento de datos de carácter personal de los usuarios, es el **Reglamento General de Protección de Datos (RGPD)**. Esta ley fue aprobada por el parlamento Europeo en 2016 y fue aplicada por las empresas en 2018 [60]. Según el reglamento, las organizaciones deben especificar a las personas físicas la recopilación y el procesamiento de información de estas. Los usuarios tendrán el derecho de aceptar, rechazar o actualizar los datos de carácter personal. También podrán solicitar la supresión y limitación de tratamiento de estos.

En España, se aplica la **Ley Orgánica de Protección de Datos Personales y garantía de los derechos digitales (LOPD-GDD)**, acorde con el RGPD. Es decir, es una ley orgánica aprobada por las Cortes Generales de España que tiene por objeto adaptar el Derecho interno español al Reglamento General de Protección de Datos.

Los datos de carácter personal de este proyecto son los registros de trazabilidad de los documentos y tareas realizadas por los usuarios. Serán accesibles solamente por parte de los administradores del sistema para generar el informe de auditoría en el caso que sea necesario.

10.2 Licencias

Las licencias de software necesarias para desarrollar el proyecto son:

- **Windows 10:** el portátil HP ZBook de la empresa Raona tiene instalado el sistema operativo de Windows 10 con una licencia de edición profesional.
- **Microsoft O365 (Raona):** es una licencia *office* para el uso de las herramientas de Microsoft, por ejemplo: Team, Outlook, Word, OneDrive, etc.
- **Microsoft O365 (cliente):** es una licencia *premium* proporcionada por la empresa sanitaria de este proyecto para hacer uso de las tecnologías SharePoint y Power Automate.
- **Sharegate:** la licencia de este producto proporciona la migración de datos de un origen a un destino de SharePoint de manera sencilla.

11 Seguimiento del proyecto

En este capítulo se explicarán los cambios que ha habido en la planificación y presupuestos iniciales, y cómo han afectado al desarrollo del sistema. Los resultados definitivos permitirán reflexionar y tomar decisiones óptimas en los proyectos futuros. También se explicará cómo se ha llevado a cabo la ejecución real del proyecto.

11.1 Cambio de planificación y presupuestos

Respecto a la planificación inicial no ha habido un cambio en las descripciones de las actividades. A medida que se ha ido avanzando el proyecto, se ha ido perfilando las definiciones de tareas de cada historia de usuario.

Sin embargo, ha habido una pequeña modificación en las tareas de desarrollo, concretamente en la iteración 4, que comprendía una fase de validación previa al lanzamiento del producto. En este sprint se ha recibido varios *feedbacks* por parte del cliente. El hecho de incorporar algunas mejoras y corrección de errores de la aplicación ha implicado una dedicación de esfuerzo y horas al proyecto. Notad que esta pequeña desviación no ha afectado la ejecución del proyecto, pero sí al cálculo del presupuesto.

En la siguiente tabla, se muestra un resumen de las tareas estimadas de cada grupo. Se puede observar que el cambio está en la iteración 4, en la que se ha añadido unas 16h más repartidas entre DS y T para la tarea de diseño, implementación y testing. Y unas 4h en *review meeting* por parte de PM y DS.

Código	Nombre	Tiempo estimado	Dependencias	Recursos	Roles
GP	Gestión del proyecto	200h	–	–	–
GP1	Alcance	20h	–	RHW, MO	PM, AS
GP2	Planificación	20h	GP1	RHW, MO, L	PM, AS
GP3	Presupuesto	10h	GP2	RHW, MO	PM, AS
GP4	Sostenibilidad	10h	GP3	RHW, MO	PM, AS
GP5	Estudios tecnológicos	20h	GP1	RHW, N	DS
GP6	Documentación	100h	GP1	RHW, MO, L	D
GP7	Defensa	20h	GP6	RHW, MO, SP	AS, DS, T
DS	Desarrollo	424h	–	–	–
SP1	Sprint 1	70h	GP5	–	–
SP11	Planning meeting	2h	GP5	RHW, MO	PM, DS
SP12	Daily meeting	5h	GP5	RHW, SP, PA, VSC, GIT	PM, DS
SP13	Diseño, implementación y testing	58h	SP11	RHW, SP, PA, VSC, GIT	DS, T
SP14	Review meeting	2h	SP13	RHW, SP, PA, VSC, GIT	PM, DS
SP15	Retrospective meeting	3h	SP13	RHW, SP	PM, DS
SP2	Sprint 2	159h	SP1	–	–
SP21	Planning meeting	2h	SP1	RHW, MO	PM, DS
SP22	Daily meeting	10h	SP1	RHW, SP, PA, VSC, GIT	PM, DS
SP23	Diseño, implementación y testing	140h	SP21	RHW, SP, PA, VSC, GIT	DS, T
SP24	Review meeting	4h	SP23	RHW, SP, PA, VSC, GIT	PM, DS
SP25	Retrospective meeting	3h	SP23	RHW, SP	PM, DS
SP3	Sprint 3	105h	SP2	–	–
SP31	Planning meeting	2h	SP2	RHW, MO	PM, DS
SP32	Daily meeting	7h	SP2	RHW, SP, PA, VSC, GIT	PM, DS
SP33	Diseño, implementación y testing	90h	SP31	RHW, SP, PA, VSC, GIT	DS, T
SP34	Review meeting	3h	SP33	RHW, SP, PA, VSC, GIT	PM, DS
SP35	Retrospective meeting	3h	SP33	RHW, SP	PM, DS

SP4	Sprint 4	110h	SP3	–	–
SP41	Planning meeting	2h	SP3	RHW, MO	PM, DS
SP42	Daily meeting	6h	SP3	RHW, SP, PA, VSC, GIT	PM, DS
SP43	Diseño, implementación y testing	92h	SP41	RHW, SP, PA, VSC, GIT	DS, T
SP44	Review meeting	7h	SP43	RHW, SP, PA, VSC, GIT	PM, DS
SP45	Retrospective meeting	3h	SP43	RHW, SP	PM, DS
LP	Lanzamiento del producto	16h	SP4	–	–
LP1	Preparación del entorno del cliente	10h	SP4	RM, SP, PA	DS
LP2	Despliegue	6h	LP1	RM, SP	DS
	Total	640h	–	–	–

Tabla 43. Resumen final de las tareas estimadas | Fuente: Elaboración propia

En la tabla 44, se muestran los costes de personal por cada actividad.

Código	PM	AS	DS	T	D	Coste
GP	60h	64h	35h	1h	100h	8 935€
GP1	20h	20h	–	–	–	2 000€
GP2	20h	20h	–	–	–	2 000€
GP3	10h	10h	–	–	–	1 000€
GP4	10h	10h	–	–	–	1 000€
GP5	–	–	20h	–	–	700€
GP6	–	–	–	–	100h	1 500€
GP7	–	4h	15h	1h	–	735€
DS	64h	0h	386h	58h	0h	18 770€
SP1	12h	0h	62h	8h	0h	3 070€
SP11	2h	–	2h	–	–	180€
SP12	5h	–	5h	–	–	450€
SP13	–	–	50h	8h	–	1 990€
SP14	2h	–	2h	–	–	180€
SP15	3h	–	3h	–	–	270€
SP2	19h	0h	134h	25h	0h	6 485€
SP21	2h	–	2h	–	–	180€
SP22	10h	–	10h	–	–	900€
SP23	–	–	115h	25h	–	4 775€
SP24	4h	–	4h	–	–	360€
SP25	3h	–	3h	–	–	270€
SP3	15h	0h	95h	10h	0h	4 450€
SP31	2h	–	2h	–	–	180€
SP32	7h	–	7h	–	–	630€
SP33	–	–	80h	10h	–	3 100€
SP34	3h	–	3h	–	–	270€
SP35	3h	–	3h	–	–	270€
SP4	18h	0h	95h	15h	0h	4 765€
SP41	2h	–	2h	–	–	180€
SP42	6h	–	6h	–	–	540€
SP43	–	–	77h	15h	–	3 145€
SP44	7h	–	7h	–	–	630€
SP45	3h	–	3h	–	–	270€
LP	0h	0h	16h	0h	0h	560€
LP1	–	–	10	–	–	350€
LP2	–	v	6	–	–	210€
Total (h)	124h	64h	437h	59h	100h	–
Total (€)	6 820 €	2 880€	15 295€	1 770€	1 500€	28 265€
Total + SS (€)	8 866€	3 744€	19 883,5€	2 301€	1 950€	36 744,5€

Tabla 44. Costes finales de recursos humanos por tarea | Fuente: Elaboración propia

En los costes generales, concretamente en la parte de software, se ha añadido el coste anual de un producto llamado ShareGate. En los cálculos iniciales no estaba previsto el uso de este programa, pero fue necesario su uso de cara al final del proyecto para la migración de datos de un entorno pre-producción de SharePoint hacia producción. Esta herramienta es proporcionada por la empresa Raona y tiene una suscripción *SG:Desktop* con un precio de 7,08€ por cinco usuarios al año [61].

Notad que la duración del proyecto en la fórmula de amortización se ve afectada en las horas,

$$\frac{\text{coste software anual (€)}}{\text{vida útil(años)} * \text{días laborables/año} * \text{dedicación en horas / día}} * \text{duración proyecto (h)},$$

donde la duración del proyecto debe ser 660h en lugar de 640h. Por tanto, los costes de amortizaciones se modifican.

Software	Coste anual (€)	Vida útil (años)	Amortización (€)
Microsoft Office 365	236,4	2	42,39
SharePoint	236,4		42,39
Power Automate	151,2		27,12
Notion	Gratuito		–
Lucidchart	Gratuito		–
Visual Studio Code	Gratuito		–
Git/TFS	Gratuito		–
ShareGate	7,08		1,27
Total	631.08		–

Tabla 45. Costes finales estimados de los recursos software | Fuente: Elaboración propia

Los costes de recursos materiales son los mismos que los costes iniciales, que son 2601€ pero las amortizaciones cambian.

Hardware	Coste (€)	Vida útil (años)	Amortización (€)
Portátil HP ZBook	1 896	4	170,02
Monitor HP 27fw	250		22,42
Teclado Tecknet	30		2,69
Ratón Logitech M185	15		1,34
Ipad 6.ª generación	400		35,87
Auriculares con micrófono	10		0,90
Total	2 601	–	233,24

Tabla 46. Costes finales de los recursos hardware | Fuente: Elaboración propia

$$\begin{aligned} \text{costes genéricos final} &= \text{coste software} + \text{coste hardware} + \text{amortizaciones} \\ &= 631,08 + 2 601 + (113,18 + 233,24) = 3 578,50€ \end{aligned}$$

Diferencia costes personales

$$\begin{aligned} &= \text{costes personales final} - \text{costes personales inicial} \\ &= 36 744,5 - 35 574,5 = 1170€ \end{aligned}$$

Diferencia costes genéricos

$$\begin{aligned} &= \text{costes genéricos final} - \text{costes genéricos inicial} \\ &= 3 578,50 - 3 559,69 = 18,81€ \end{aligned}$$

En resumen, el aumento de las horas en la planificación inicial ha implicado una pequeña modificación en los cálculos económicos iniciales. Sin embargo, los costes de las horas dedicadas en la última iteración quedarían cubiertas por los costes de

contingencia. En la gestión económica inicial de este proyecto se decidió reservar un 15% de los costes personales y generales para los posibles imprevistos. Por tanto, en un proyecto futuro se deberá considerar y prever más tiempo en la planificación para la fase de validación ya que es un punto crítico para el lanzamiento del producto.

11.2 Ejecución real

La ejecución real de este proyecto está dividida en 4 etapas para desarrollar las actividades de la aplicación con el fin de cumplir los objetivos del sistema. A continuación, se detallan las tareas de monitoreo y los resultados alcanzados de este proyecto.

11.2.1 Sprint 1

La primera iteración de la fase de desarrollo tiene una duración de 2 semanas. La fecha de inicio es 26 de julio de 2021 y la fecha de fin es 6 de agosto de 2021. Primeramente, se convoca una reunión con el cliente para definir la estructura del documento. Es decir, en esta parte de consultoría se elabora un documento para definir los tipos de documentos del sistema y sus propiedades con una serie de documentos y propiedades.

Tipologías de Documentos

En la biblioteca de trabajo se publicaran aquellos documentos de calidad que requieren de un ciclo de aprobación:

- AR – Análisi de Riesgos. Codificación manual
- DP – Documentos Principales. Codificación DP/XX
- PG – Procedimients Generales. Codificación PG/XX
- NC – No Conformidades. Codificación NC/AA/XX AA (Año en curso)
- AC – Acciones Correctivas. Codificación AC/AA/XX AA (Año en curso)
- CC – Control de Cambios. Codificación CC/AA/XX AA (Año en curso)
- PNT – Procedimientos Normalizados de Trabajo. Codifocación PNT/L/XX, PNT/R/XX, ...

Subtipos:

- L – Logísticos
- I – Infraestructuras
- R – RRHH
- S – Sistemas Informáticos
- T – Dirección Técnica
- V – Varios

- IT – Instrucciones de trabajo. Vinculadas a PNTs o PGs. Codificación IT/XX-PNT/L/01
- TE – Instrucciones técnicas. Vinculadas a PNTs o PGs. Codificación TE/XX-PNT/L/01
- PL – Plantillas. Vincluidas a PNTs o PG. Codificación PL/XX-PNT/L/01
- MA – Manual. Hay que relacionar estos manuales a un código de PNT cuando se requiera

A parte de la tipologia de documentos, un documento de calidad ha de tener las siguientes propiedades:

- Fecha de caducidad
- Vigente?
- Formato en el que está el documento
- Área reponsable del documento

Figura 97. Tipologías de documentos | Fuente: Elaboración propia

La figura 97 es la primera versión de la estructura de documentos. Las propiedades de los documentos se han ido actualizando a lo largo de este proyecto. En la sección [5.1.1.2](#) se muestra la versión definitiva.

Posteriormente, se hace el *sprint planning* con el project manager donde se elige la historia de usuario 1 y se define un conjunto de tareas en el product backlog.

Historia de usuario	Tarea
HU 1	<ul style="list-style-type: none"> • Crear entorno pre-producción • Completar la especificación de la topología de documentos • Crear biblioteca de trabajo • Configurar base de datos • Crear columnas y tipos de contenidos • Configurar formulario de documentos • Generar un código alfanumérico único para un documento

Tabla 47. Sprint backlog 1 | Fuente: Elaboración propia

Por tanto, el objetivo de este *sprint* es plasmar la estructura de documentos en el entorno de SharePoint para que los responsables puedan subir los documentos en la plataforma y clasificarlos según la topología de documentos.

Uno de los puntos fuertes de la metodología agile es mostrar el estado del proyecto por sprint al cliente. Por tanto, se hace una *demo* para validar la estructura de documentos. Adicionalmente, se ha concretado el funcionamiento general y los procesos de negocio del sistema en detalle y se ha pactado las funcionalidades a desarrollar en el siguiente *sprint*.

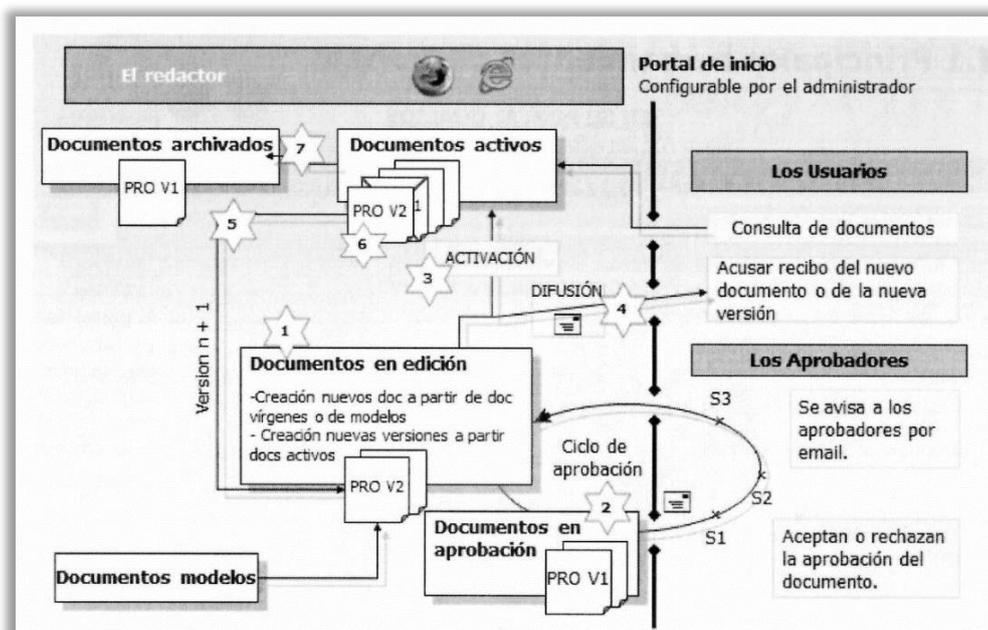


Figura 98. Funcionamiento general | Fuente: Raona

11.2.2 Sprint 2

La iteración 2 tiene una duración de 4 semanas. El objetivo principal es implementar la lógica del proceso de publicación. Es una parte compleja e importante del sistema ya que involucra diversas ideas (mirar la sección [5.1.2.1](#)).

A medida que transcurrían las semanas, iban surgiendo dudas relacionadas con el ciclo de vida de un documento. Por ejemplo, durante la redacción de un documento colaborativo puede haber un desacuerdo entre los compañeros para generar dicho documento, y lo mismo podría suceder en la fase de aprobación. Para ello, se mantuvo una comunicación constante con el cliente para aclarar o definir soluciones ante estas situaciones. Para definir correctamente el ciclo de aprobación se hizo una propuesta definitiva al cliente, en el que se incorpora un modelo de mediador para gestionar las incidencias relacionadas a los documentos (mirar la figura 99).

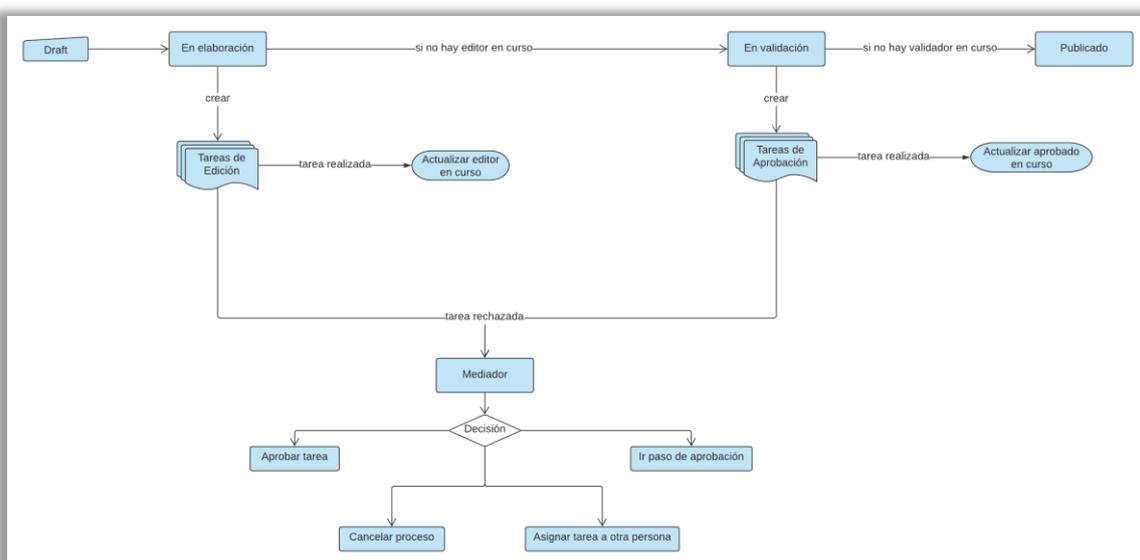


Figura 99. Mediador en el proceso de publicación | Fuente: Elaboración propia

El *sprint backlog* de la iteración 2 es el siguiente:

Historia de usuario	Tarea
HU 2	<ul style="list-style-type: none"> Controlar estado de elaboración
HU 3	<ul style="list-style-type: none"> Crear lista Crear tareas de edición Desglosar personas de un grupo AD Controlar proceso de edición
HU 4	<ul style="list-style-type: none"> Gestionar respuesta de un editor
HU 5	<ul style="list-style-type: none"> Crear tareas de aprobación Controlar niveles de aprobación
HU 6	<ul style="list-style-type: none"> Gestionar respuesta de un aprobador
HU 7	<ul style="list-style-type: none"> Gestionar respuesta de un mediador
HU 8	<ul style="list-style-type: none"> Convertir documento Word a Pdf Publicar documento en la biblioteca de prueba

Tabla 48. *Sprint backlog 2* | Fuente: Elaboración propia

Respecto a la validación de las soluciones implementadas, se hizo una *demo* para mostrar el funcionamiento del proceso de publicación y verificar diferentes escenarios posibles del sistema. Por ejemplo:

- **Caso 1:** todos los usuarios realizan sus tareas
- **Caso 2:** un usuario rechaza su tarea asignada, se introduce la figura del mediador
- **Caso 3:** mediador realiza una acción

El cliente quedó satisfecho con lo que se presentó. Aunque se manifestó de incorporar algunas mejoras y cambios en algunas historias de usuario de cara a la siguiente entrega. Y para finalizar se ha hablado de las funciones que se hará en el siguiente *sprint*.

11.2.3 Sprint 3

La iteración 3 tiene una duración de 3 semanas. El trabajo a realizar en estas semanas consistía en mejorar algunas historias de usuario del *sprint* anterior y hacer otras relacionadas con los temas de procesos de negocio y seguridad. Al inicio se pensó que se habían definido demasiadas actividades para esta iteración, pero gracias a una buena planificación y coordinación con el director y el cliente de este proyecto se pudo completar las funcionalidades acordadas de la entrega.

En la siguiente tabla, se muestra el *sprint backlog* de esta iteración.

Historia de usuario	Tarea
HU 2	<ul style="list-style-type: none"> • Añadir botón para actualizar estado del documento
HU 4, 6,7	<ul style="list-style-type: none"> • Añadir botón de opciones y <i>pop-up</i> para controlar respuesta de la tarea
HU 8	<ul style="list-style-type: none"> • Crear biblioteca pública • Controlar “leído y comprendido” de justificados • Controlar versión de un documento • Distribuir documento • Controlar tareas para usuarios nuevos y/o eliminados
HU 9	<ul style="list-style-type: none"> • Controlar estado del documento
HU 10	<ul style="list-style-type: none"> • Definir cuerpo y asunto de los correos • Implementar sistema de notificación • Recordatorio de tareas pendientes • Recordatorio a los justificados
HU 11	<ul style="list-style-type: none"> • Calcular la caducidad de un documento
HU 12	<ul style="list-style-type: none"> • Romper permisos de elementos • Controlar accesibilidad del sistema • Ocultar propiedades innecesarias de las bibliotecas y listas

Tabla 49. *Sprint backlog 3* | Fuente: *Elaboración propia*

Finalmente, se convocó una reunión de *demo* con el cliente para validar las funcionalidades de este *sprint*. Se ha mostrado el funcionamiento completo de los procesos, desde que se sube un documento en la biblioteca de trabajo, pasando por el proceso de edición y aprobación hasta que se publica en un sitio público del sistema con los permisos correspondientes, y que posteriormente se distribuye y se mantiene un control de trazabilidad de dicho documento.

Además, se habló de las actividades que se harán en el siguiente *sprint* y se concretó de abrir una fase de validación tal que los usuarios del sistema puedan verificar el funcionamiento global de la aplicación.

11.2.4 Sprint 4

La iteración 4 tiene una duración de 3 semanas. El objetivo principal de esta etapa final fue trasladar las soluciones implementadas en el entorno del cliente (entorno de producción). Sin embargo, previamente se tenía que finalizar las historias de usuarios del *backlog*.

Historia de usuario	Tarea
HU 8	<ul style="list-style-type: none"> • Generar cabeceras o <i>tags</i> del documento automáticamente • Crear plantillas de documentos con <i>tags</i>
HU 13	<ul style="list-style-type: none"> • Crear menú o barra de navegación para los usuarios y administradores • Crear vistas con filtros
HU 14	<ul style="list-style-type: none"> • Añadir botón de sugerencia en los documentos publicados
HU 15	<ul style="list-style-type: none"> • Controlar comentarios y mostrarlos en un cuadro de comentario

Tabla 50. *Sprint backlog 4* | Fuente: *Elaboración propia*

A pesar de haber terminado las historias de usuario del *backlog*, tanto los usuarios del sistema como el cliente pidieron algunas mejoras del sistema, y descubrieron algunos errores de la aplicación. A medida que se iban incorporando los cambios, y paralelamente se iba construyendo la estructura del entorno de producción para desplegar las soluciones validadas.

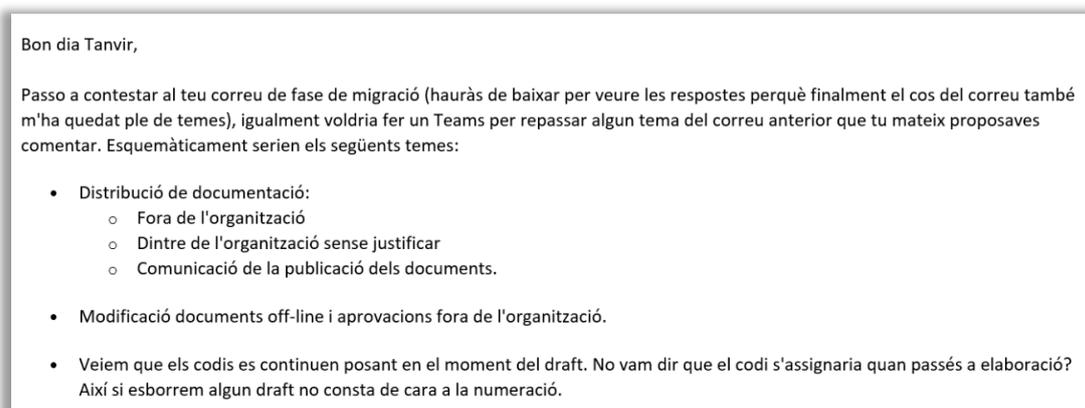


Figura 100. *Correo de la fase de validación* | Fuente: *Outlook*

Hola,

Tal com hem acordat volem que es canviï l'adreça de correu [qualitat@](#) per [notificacionsSGQ@](#) per realitzar les comunicacions en el Share Point.

Gràcies.

Figura 101. Petición de cambio del correo en las notificaciones | Fuente: Outlook

Finalmente, se convocó una reunión para validar el proyecto, se concretan unas fechas para la migración de datos y el lanzamiento del *software*. A pesar de ser una iteración intensiva para el cierre de proyecto, el cliente ha mostrado su satisfacción de haber cumplido con las expectativas propuestas. La lección aprendida de esta iteración fue que se debe considerar muy importante la fase de validación con los usuarios del sistema ya que requiere una atención y esfuerzo para entregar el proyecto correctamente.

12 Sostenibilidad

En este capítulo se explicarán los diferentes aspectos de sostenibilidad relacionados en un proyecto de *software*. El informe de sostenibilidad permitirá analizar el uso de recursos y el impacto de implantar el producto en la sociedad. Para ello, se hará un análisis en términos de sostenibilidad económica, social y ambiental.

Respecto al aspecto económico, el autor de este proyecto es consciente de la importancia de generar una rentabilidad económica de un proyecto *software*. Existen diferentes herramientas para estimar la viabilidad económica, por ejemplo: análisis DAFO, costos, presupuestos, etc. El presupuesto del proyecto ha estado dividido en costes personales, generales, contingencias e imprevistos. Estos costes han sido revisados y validados para la viabilidad económica del proyecto.

En cuanto a la dimensión social, es importante valorar la seguridad y justicia social de los diferentes tipos de usuarios de este proyecto. El objetivo es proporcionar un *software* ético, transparente y accesible a estos.

En relación con el impacto ambiental, se ha de tener en cuenta los efectos que pueden causar los productos o servicios de sistemas y programas informáticos obsoletos. En este proyecto se dará mucha importancia a la digitalización de información o documentos con el objetivo de reducir los espacios o recursos físicos y usarlos correctamente.

Para obtener una valoración correcta de la sostenibilidad del proyecto, el autor de este proyecto realizó una encuesta formada por parte del proyecto de investigación EDINSOST2-ODS⁷ para evaluar y obtener información sobre los conocimientos en relación con los Objetivos de Desarrollo Sostenible (ODS) y las competencias en sostenibilidad.

A continuación, en los siguientes apartados se presentan en detalle las explicaciones de las tres dimensiones.

12.1 Dimensión económica

El coste total del proyecto se ha calculado, detalladamente en el capítulo 4, teniendo en cuenta los costes personales, materiales, contingencias e imprevistos. El presupuesto elaborado se ha intentado ajustar al máximo posible a la previsión del desarrollo del proyecto y se ha considerado los posibles riesgos e imprevistos.

Como ya se comentó en el apartado [1.4](#), existen diferentes soluciones para crear una plataforma web basada en la gestión de documentos. La empresa sanitaria de este proyecto ejecutaba los procesos de negocio (publicación, distribución, etc.) de documentos mediante reuniones presenciales, llamadas, firma manual, escaneo y distribución por correo manualmente. Es decir, cuando se quiere generar un documento y distribuirlo, el departamento de calidad genera el documento, realiza reuniones con

⁷ <https://bit.ly/3wZjPLw>

el equipo implicado y hace que los aprobadores firmen físicamente el documento, el cual es distribuido por correo y los que lo reciben han de firmar físicamente su aceptación. Por tanto, todo este flujo de negocio de la empresa quedaría resuelto mediante el uso de la plataforma web SharePoint.

Se ha estimado que la solución propuesta tendrá un impacto positivo económico en la empresa ya que:

- Se prevé una reducción de horas del equipo interno de calidad implicado en generar y publicar documentos de unas 300 horas anuales.
- Se calcula una reducción de horas del personal del cliente para firmar físicamente los documentos de 2 000 horas.

Este impacto en horas se ha valorado en importe a una tarifa de 25€, lo que da un ahorro anual de 57 500€. Y en cuanto al material físico para generar las firmas físicas se ha estimado un ahorro aproximado de 3 000€.

12.2 Dimensión social

La realización de este trabajo ha ayudado al autor de este proyecto a tener una visión nítida sobre la gestión e implantación de un producto *software* en un cliente real. Asimismo, es un proyecto relacionado en el ámbito de ingeniería de *software*, que ha permitido adquirir conocimientos en las tecnologías de Microsoft y experiencias en el desarrollo de aplicaciones web. También, ha permitido mejorar la comunicación entre diferentes compañeros del equipo y clientes.

La solución del sistema está orientada a la simplificación de trabajo de los usuarios. El proceso de publicación es un proceso complicado que implica mucho esfuerzo personal del equipo de calidad y que en muchos casos implica “discusiones”. Al pasarse a un sistema automático hará que se despersonalice este proceso.

Adicionalmente, el principal problema de trazabilidad total de los procesos y el estado de todas las aprobaciones para poder justificar ante unas auditorías quedaría cubierta totalmente en el proyecto con las soluciones de la arquitectura de Microsoft SharePoint.

12.3 Dimensión ambiental

Se ha estimado que la realización de este proyecto tendrá un impacto positivo en el ambiente, es decir la transformación de un sistema tradicional a uno digital va a implicar la reducción de uso de papel disminuyendo el impacto en el medioambiente del proceso.

Anteriormente, el cliente realizaba el sistema de firma y validación de la documentación oficial de la compañía en formato físico, por tanto, el paso a un sistema digital quedará reflejado como un compromiso ambiental por parte de la empresa.

Se calcula que anualmente se realizan unos 500 procesos de validación, de los cuales hay un consumo medio de 30 hojas por proceso, lo que equivale a una reducción de

15 000 hojas anuales. Según algunos estudios científicos, se estiman que 500 hojas de papel equivalen a 2 500 litros de agua, 3 000 hojas o 54 kg de papel equivale la tala de un árbol y 15 000 litros de agua. Cada tonelada de papel que se elabora demanda unos 15 árboles y 225 000 litros de agua. En este proyecto se estima una reducción de 15 000 hojas anuales, lo que equivale a evitar la tala de 5 árboles y un ahorro de 75 000 litros de agua.

Por otra parte, se contempla un impacto ambiental negativo, de menor gravedad, en relación con el desarrollo y los recursos *hardware* utilizados en este proyecto. El objetivo es minimizar el consumo eléctrico tal que el proyecto se realiza en la arquitectura de Microsoft, por tanto, los recursos serán compartidos con el ecosistema de este utilizando sólo aquellos recursos que se necesitan.

13 Conclusiones

Una vez descritos todos los aspectos del proyecto, se pueden extraer varias conclusiones de este, tanto a nivel de funcionalidades implementadas como a nivel personal. Es decir, en este capítulo se detallarán las competencias técnicas trabajadas en este proyecto, los logros obtenidos, la perspectiva del autor y el trabajo futuro del proyecto.

13.1 Competencias técnicas trabajadas y relación con la especialidad de ingeniería del software

Durante la realización de este proyecto se han podido aplicar una serie de conocimientos adquiridos en las diferentes asignaturas del grado [62], concretamente las de la especialidad de *Ingeniería del Software*. Estos conocimientos han sido muy útiles para la elaboración de este proyecto.

- **ER (Ingeniería de Requisitos):** los primeros conocimientos aplicados en este proyecto, previa a la construcción de la aplicación, fueron el análisis de la situación actual de la empresa y la especificación de requisitos del sistema para garantizar el éxito del proyecto. Estos dos bloques abarcan una serie de actividades, por ejemplo: detectar los problemas actuales, proponer una solución, determinar los requisitos, analizar escenarios, etc., y que fueron eficaces el uso de estos.
- **GPS (Gestión de Proyectos de Software) y PES (Proyecto de Ingeniería del Software):** gracias a estas asignaturas el autor fue capaz de aplicar las nociones de la ingeniería de proyecto en el ámbito del software, es decir, ha podido elaborar una planificación, analizar los riesgos, aplicar los principios de la metodología ágil (Scrum), uso de herramientas adecuadas en el desarrollo, redactar la memoria técnica del proyecto, etc.
- **IES (Introducción a la Ingeniería del Software), AS (Arquitectura del Software) y ASW (Aplicaciones y Servicios Web):** en cuanto al diseño del sistema, los conocimientos adquiridos en estas asignaturas fueron indispensables en este proyecto. Aunque a nivel de implementación, relacionado con el entorno de Microsoft, se ha recurrido a la técnica de autoaprendizaje ya que no existe una asignatura que enfoque en concreto a las tecnologías de Microsoft. Sin embargo, los fundamentos aprendidos de la aplicación web en ASW fueron provechosos en la fase de implementación.

Por tanto, los conocimientos adquiridos a lo largo de 4 años de carrera han estado involucrados en este trabajo de fin de grado y han ayudado a realizarlo con seguridad.

Adicionalmente, se ha trabajado en unas competencias técnicas específicas para cumplir los logros de este proyecto. Cada una de ellas debe estar especificada con un

grado de implicación, pudiendo ser “En profundidad”, “Bastante” o “Un poco”. A continuación, se muestra una lista con la relación de competencias técnicas asociadas a este proyecto.

CES1.1: Desarrollar, mantener y evaluar sistemas y servicios software complejos y/o críticos. [Bastante]

En este proyecto se ha desarrollado un *software* complejo que se enfoca a la creación de un espacio documental para el departamento de *compliance* de una compañía del sector farmacéutico. Este programa requiere una gestión de flujos de trabajo automatizados, control de acceso y seguridad de los recursos, el almacenaje y custodia de documentos. Por tanto, la complejidad recae en el desarrollo y mantenimiento de la aplicación web mediante la tecnología SharePoint.

CES1.2: Dar solución a problemas de integración en función de las estrategias, de los estándares y de las tecnologías disponibles. [En profundidad]

El sistema *software* desarrollado en este proyecto es un gestor documental basado en Microsoft SharePoint. Se han proporcionado soluciones basadas en las tecnologías estándares de Microsoft Office 365 y que estas se integran claramente en el entorno de SharePoint.

CES1.3: Identificar, evaluar y gestionar los riesgos potenciales asociados a la construcción de software que pudiesen presentarse. [Un poco]

Al inicio del desarrollo del sistema se analizaron los posibles riesgos y obstáculos que podrían surgir a lo largo del proyecto y cómo gestionarlos. Adicionalmente, la aplicación web de este proyecto gestiona el proceso de publicación y distribución de documentos. Por tanto, se han valorado los posibles riesgos de que un documento publicado sea visible solamente para aquellos usuarios que tienen permisos y que esté firmado o validado por los aprobadores correspondientes del mismo.

CES2.1: Definir y gestionar los requisitos de un sistema software. [Bastante]

La empresa sanitaria de este proyecto hizo una solicitud de propuestas [63] (*request for proposal*) en la que explicó los problemas existentes en ésta, a partir de la cuál, se obtuvieron los requisitos del sistema y se validaron con el cliente. En el caso de los requisitos funcionales fueron descritos en historias de usuario, y en el caso de requisitos no funcionales mediante la plantilla Volere. Posteriormente, fueron distribuidos en un conjunto de tareas para detallar la construcción del *software* de manera eficaz.

CES2.2: Diseñar soluciones apropiadas en uno o más dominios de aplicación, utilizando métodos de ingeniería del software que integren aspectos éticos, sociales, legales y económicos. [Un poco]

Para llevar a cabo este proyecto se han tenido en cuenta las necesidades de los actores implicados para gestionar los documentos del sistema. El tratamiento de los aspectos legales viene heredado de la estructura del proyecto ya que se manejan informaciones o documentos confidenciales relacionados con el sector sanitario.

13.2 Conclusiones personales

La realización de este proyecto ha permitido entender de manera completa las diferentes fases de desarrollo de un producto *software*. El hecho de ser un proyecto llave en mano a cliente ha ayudado a tener una experiencia fructífera y real. Así mismo, en esta vivencia en el mundo empresarial, ha facilitado trabajar y mejorar las siguientes *soft-skills*: trabajo en equipo, aprendizaje autónomo, comunicación eficaz oral y escrita.

Por lo que respecta al desarrollo de la aplicación web mediante Microsoft SharePoint, ha supuesto un reto para el autor de este proyecto ya que fue una motivación personal para aprender y ampliar conocimientos en la rama de la programación web basada en componentes de Microsoft. Se puede decir que las tecnologías de Microsoft han sido útiles y han permitido lograr los objetivos del proyecto.

En conclusión, el trabajo de este proyecto ha sido exigente y dinámico, y a nivel de satisfacción ha sido gratificante ya que se ha podido entregar el *software* al cliente cumpliendo los objetivos y las expectativas de éste. Esto refuerza la relación existente entre el cliente y la empresa Raona para futuros proyectos.

13.3 Trabajo futuro

Al inicio de este proyecto se comentó que un documento podía pertenecer a una de los dos áreas o departamentos: área de calidad o área de dirección técnica. En este trabajo se han tratado aquellos documentos pertenecientes al departamento de calidad. Por tanto, en principio el gestor documental no termina aquí, sino que existe la posibilidad de construir otra plataforma enfocada a la gestión de documentos del departamento de dirección técnica.

Se prevé pactar con el cliente los requisitos de este sistema, aunque éste ha comunicado que tendrá una estructura de documentos muy similar al proyecto actual. Estos documentos de dirección técnica, también llamados documentos internos, no requerían tener un ciclo de aprobación ni distribución. El objetivo principal será la colaboración y compartición de dichos documentos con otros departamentos. Adicionalmente, se contempla gestionar los siguientes tipos de documentos:

- Informes
- Protocolos de calificación OQ/PQ
- Registro de tiquetes de temperatura por matrícula de vehículo
- Registro de tiquetes de limpieza
- Simulacros

En resumen, en un futuro se espera crear la estructura de documentos internos por departamentos.

Referencias

- [1] R. (2020, 12 noviembre). *Raona - Coding Corporate Cultures*. Raona. Recuperado 21 de septiembre de 2021, de <https://www.raona.com>
- [2] *Facultat d'Informàtica de Barcelona*. (s. f.). UPC. Recuperado 21 de septiembre de 2021, de <https://www.fib.upc.edu>
- [3] Alonso, A. F. (2021, 14 mayo). *¿Qué es un CMS? Conoce los mejores gestores de contenido*. Webempresa. Recuperado 23 de septiembre de 2021, de <https://www.webempresa.com/blog/que-es-cms-los-mejores-gestores-de-contenido.html>
- [4] Athow, D. (2021, 8 octubre). *Best website builder of 2021 from 50+ real expert reviews*. TechRadar. Recuperado 23 de septiembre de 2021, de <https://www.techradar.com/news/the-best-website-builder>
- [5] Steele, H. (2021, 21 julio). *CMS vs Website Builder: Which is Better?* SuperbWebsiteBuilders.Com. Recuperado 23 de septiembre de 2021, de <https://superbwebsitebuilders.com/cms-vs-website-builder>
- [6] Garcia, J. (2021, 13 octubre). *How to Set up a Website in 2021: An Updated Beginner's Guide*. Tooltester. Recuperado 23 de septiembre de 2021, de <https://www.tooltester.com/en/how-to-set-up-my-own-website>
- [7] Editorial Staff & About the Editorial Staff. (2021, 14 agosto). *15 Best and Most Popular CMS Platforms in 2021 (Compared)*. WPBeginner. Recuperado 23 de septiembre de 2021, de <https://www.wpbeginner.com/showcase/best-cms-platforms-compared>
- [8] Hanna, T. (2020, 7 agosto). *What's the Difference Between DMS and CMS?* Best Content Services Platforms, Software and Content Management ECM Vendors. Recuperado 23 de septiembre de 2021, de <https://solutionsreview.com/content-management/whats-the-difference-between-dms-and-cms>
- [9] Nicastro, D. (2021, 14 enero). *CMS vs. DMS: Understanding the Key Differences*. CMSWire.Com. Recuperado 23 de septiembre de 2021, de <https://www.cmswire.com/information-management/cms-vs-dms-understanding-the-key-differences>
- [10] Docunecta. (2019, 3 diciembre). *Gestión de documentos o de contenido: ¿son lo mismo?* Recuperado 23 de septiembre de 2021, de <https://www.docunecta.com/blog/gestion-de-documentos-y-gestion-de-contenido-cual-necesitas>
- [11] Taylor, K. (2020, 4 agosto). *ECM vs. CMS: Similarities and Differences Explained*. HitechNectar. Recuperado 25 de septiembre de 2021, de <https://www.hitechnectar.com/blogs/ecm-vs-cms>
- [12] Microsoft. (s. f.). *SharePoint, herramientas de software de colaboración en equipo*. Recuperado 25 de septiembre de 2021, de <https://www.microsoft.com/es-es/microsoft-365/sharepoint/collaboration>
- [13] *GUÍA BÁSICA SHAREPOINT*. (s. f.). UPM. Recuperado 25 de septiembre de 2021, de <https://www.upm.es/sfs/Rectorado/Vicerrectorado%20de%20Tecnologias%20de%20la%20Info>

[rmacion%20y%20Servicios%20en%20Red/Gabinete%20de%20Tele-Educacion/Perfil%20PDI/guia_basica_SharePoint.pdf](#)

[14] Peterson, A. (2021, 29 julio). *Best SharePoint Alternatives for 2021*. TechnologyAdvice. Recuperado 25 de septiembre de 2021, de <https://technologyadvice.com/blog/information-technology/sharepoint-alternatives>

[15] *Huddle | Secure Document Collaboration & Client Portal Software*. (s. f.). Huddle. Recuperado 25 de septiembre de 2021, de <https://www.huddle.com>

[16] *Secure File Sharing, Storage, and Collaboration*. (2016, 19 marzo). Box. Recuperado 25 de septiembre de 2021, de <https://www.box.com>

[17] Atlassian. (s. f.). *Confluence - Funcionalidades*. Recuperado 25 de septiembre de 2021, de <https://www.atlassian.com/es/software/confluence/features>

[18] L. (2021, 12 junio). *11 razones por las que Sharepoint es la mejor plataforma para la gestión de documentos*. Ungoti. Recuperado 25 de septiembre de 2021, de <https://ungoti.com/es/blog/por-que-sharepoint-mejor-gestion-de-documentos>

[19] Alvarez, M. (s. f.). *¿SharePoint OnPremise o SharePoint Online?* Visualmente. Recuperado 25 de septiembre de 2021, de <https://www.visualmente.cl/blog/item/7-sharepoint-onpremise-o-sharepoint-online.html>

[20] M. (s. f.). *SharePoint Online VS. On-premises & 2019 Version Review*. Computer Support. Recuperado 25 de septiembre de 2021, de <https://www.computersupport.com/itanywherelabs/sharepoint-online-on-premises-2019-version-review>

[21] P. (s. f.). *GitHub - phindle/error-lens: Visual Studio Code extension which enhances the display of errors and warnings*. GitHub. Recuperado 4 de octubre de 2021, de <https://github.com/phindle/error-lens>

[22] McDermott, M. (2020, 28 diciembre). *SharePoint Online: Everything You Should Know*. Spanning. Recuperado 26 de septiembre de 2021, de <https://spanning.com/blog/sharepoint-online>

[23] *Power Automate | Microsoft Power Platform*. (s. f.). Microsoft. Recuperado 26 de septiembre de 2021, de <https://powerautomate.microsoft.com/es-es>

[24] *PnP | Microsoft 365*. (s. f.). GitHub. Recuperado 26 de septiembre de 2021, de <https://pnp.github.io>

[25] Colaboradores de Wikipedia. (2020, 27 marzo). *ISO/IEC 15504*. Wikipedia, la enciclopedia libre. Recuperado 26 de septiembre de 2021, de https://es.wikipedia.org/wiki/ISO/IEC_15504

[26] Wikipedia contributors. (2021, 12 octubre). *Agile software development*. Wikipedia. Recuperado 26 de septiembre de 2021, de https://en.wikipedia.org/wiki/Agile_software_development

[27] Abellán, E. (2020, 5 marzo). *Scrum: qué es y cómo funciona esta metodología*. We are marketing. Recuperado 26 de septiembre de 2021, de <https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html>

- [28] Ferreira, A. (2020, 30 junio). *Product Backlog: Épico, Historia de Usuario y Tareas*. Knowledge21. Recuperado 26 de septiembre de 2021, de <https://knowledge21.es/blog/product-backlog-epico-historia-de-usuario-y-tareas>
- [29] Notion – *The all-in-one workspace for your notes, tasks, wikis, and databases*. (s. f.). Notion. Recuperado 26 de septiembre de 2021, de <https://www.notion.so>
- [30] Andreu, F. (s. f.). *Felipe's Collection - Azure DevOps*. Microsoft Docs. Recuperado 26 de septiembre de 2021, de <https://docs.microsoft.com/es-es/users/felipeandreu/collections/jq8kcp15qd8xy6>
- [31] Tanvir, H. (s. f.). *Product backlog*. TFG on Notion. Recuperado 12 de octubre de 2021, de <https://ballistic-bubbler-a71.notion.site/deb8e107e4024ba995577911f0710981?v=e6013170764f48cb87dfd4bcdfe7a481>
- [32] *¿Conoces los entornos de trabajo utilizados en el desarrollo de una aplicación web?* (s. f.). DHARMA WEB STUDIO. Recuperado 26 de septiembre de 2021, de <https://dharmawebstudio.com/blog/conoces-los-entornos-de-trabajo-utilizados-en-el-desarrollo-de-una-aplicacion-web>
- [33] Atlassian. (s. f.-b). *Flujo de trabajo de Gitflow | Atlassian Git Tutorial*. Recuperado 27 de septiembre de 2021, de <https://www.atlassian.com/es/git/tutorials/comparing-workflows/gitflow-workflow>
- [34] *Estándares de codificación – Oh my root*. (s. f.). Oh my root. Recuperado 27 de septiembre de 2021, de <https://www.ohmyroot.com/buenas-practicas-legibilidad-del-codigo>
- [35] Hamilton, T. (2021, 8 octubre). *What is User Acceptance Testing (UAT)? with Examples*. Guru99. Recuperado 2 de octubre de 2021, de <https://www.guru99.com/user-acceptance-testing.html>
- [36] *Treball de Fi de Grau*. (s. f.). Facultat d'Informàtica de Barcelona. Recuperado 4 de octubre de 2021, de <https://www.fib.upc.edu/ca/estudis/graus/grau-en-enginyeria-informatica/treball-de-fi-de-grau>
- [37] Colaboradores de Wikipedia. (2021, 24 septiembre). *Diagrama de Gantt*. Wikipedia, la enciclopedia libre. https://es.wikipedia.org/wiki/Diagrama_de_Gantt
- [38] Microsoft. (s. f.). *Office 365 E3*. Recuperado 7 de octubre de 2021, de <https://www.microsoft.com/es-es/microsoft-365/enterprise/office-365-e3?activetab=pivot%3aoverviewtab>
- [39] *Precios | Microsoft Power Automate*. (s. f.). Microsoft. Recuperado 7 de octubre de 2021, de <https://powerautomate.microsoft.com/es-es/pricing>
- [40] *Tabla de coeficientes de amortización lineal | Amortización*. (s. f.). Agencia Tributaria. Recuperado 8 de octubre de 2021, de https://www.agenciatributaria.es/AEAT.internet/Inicio/Segmentos/Empresas_y_profesionales/Empresas/Impuesto_sobre_Sociedades/Periodos_impositivos_a_partir_de_1_1_2015/Base_imponible/Amortizacion/Tabla_de_coeficientes_de_amortizacion_lineal.shtml

- [41] *Introducción al análisis de requisitos (A.R.) - Análisis y diseño detallado de aplicaciones informáticas de gestión.* (s. f.). Sites Google: adai6jfm. Recuperado 31 de octubre de 2021, de <https://sites.google.com/site/adai6jfm/home/introduccion-al-analisis-de-requisitos-ar>
- [42] Pressman, P. (2002). *Análisis de requisitos del software.* Tesuva. Recuperado 31 de octubre de 2021, de <https://tesuva.edu.co/phocadownloadpap/Analisis%20de%20requisitos%20del%20software.pdf>
- [43] @SoyIvanRamirez. (2018, 23 abril). *Ciclo de vida de desarrollo de Software.* efectodigital. Recuperado 31 de octubre de 2021, de <https://www.efectodigital.online/single-post/2018/04/23/ciclo-de-vida-de-desarrollo-de-software>
- [44] International Conference on Software Engineering. (1996, 23 agosto). *The Uncertainty Principle in Software Engineering.* University of California. Recuperado 31 de octubre de 2021, de <http://jeffsutherland.org/papers/zivchaos.pdf>
- [45] Sabadí, P. (s. f.). *Plantilla para las historias de usuario - Scrumízate.* scrumizate. Recuperado 31 de octubre de 2021, de <http://scrumizate.com/post/58/plantilla-para-las-historias-de-usuario>
- [46] Vela, J. P. (2020, 5 abril). *Los elementos de una buena historia de usuario.* Jeronimo Palacios & Associates. Recuperado 31 de octubre de 2021, de <https://jeronimopalacios.com/product-delivery/los-elementos-una-buena-historia-de-usuario>
- [47] Scrum Manager BoK. (2021, 26 abril). *Historia de usuario.* Recuperado 31 de octubre de 2021, de https://www.scrummanager.net/bok/index.php/Historia_de_usuario
- [48] J., & Robertson, S. (2006, 11 febrero). *Volere Plantilla de Especificación de Requisitos.* Volere. Recuperado 31 de octubre de 2021, de https://www.volere.org/wp-content/uploads/2018/12/template_es.pdf
- [49] K. (2021a, septiembre 13). *Introducción a SharePoint - SharePoint in Microsoft 365.* Microsoft Docs. Recuperado 31 de octubre de 2021, de <https://docs.microsoft.com/es-es/sharepoint/introduction>
- [50] Groux, B. (2020, 3 octubre). *What is the Microsoft Power Platform in Layman's Terms?* Medium. Recuperado 31 de octubre de 2021, de <https://msft.engineer/what-is-the-microsoft-power-platform-in-laymans-terms-3c5002bce676>
- [51] Bartumeu, G. (2020, 13 noviembre). *¿Qué es Power Automate? Crea flujos de trabajo automatizados.* Aglaiia. Recuperado 31 de octubre de 2021, de <https://aglaia.es/blog/power-automate/power-automate>
- [52] IFR Group. (2021, 18 mayo). *Power Automate: low-code para automatizar.* IFR. Recuperado 31 de octubre de 2021, de <https://www.ifr.es/es/blog-dynamics-365-power-automate-low-code-para-automatizar>
- [53] *La arquitectura de SharePoint Online.* (s. f.). CompartiMOSS. Recuperado 31 de octubre de 2021, de <https://www.compartimoss.com/revistas/numero-42/la-arquitectura-de-sharepoint-online>
- [54] Guendel, M. (2011, 1 octubre). *Capacidades de SharePoint 2010.* Manfred's space. Recuperado 31 de octubre de 2021, de <https://manfredquendel.wordpress.com/2011/09/30/capacidades-de-sharepoint-2010>

- [55] A. (s. f.). *Crear e implementar soluciones de SharePoint Framework - Learn*. Microsoft Docs. Recuperado 19 de diciembre de 2021, de <https://docs.microsoft.com/es-es/learn/modules/intro-sharepoint-framework/3-create-deploy>
- [56] *Office UI Fabric React Examples*. (s. f.). Fabric. Recuperado 19 de diciembre de 2021, de <https://fabricweb.z5.web.core.windows.net/oufr/6.50.2>
- [57] *SP Editor*. (s. f.). SP. Recuperado 19 de diciembre de 2021, de <https://chrome.google.com/webstore/detail/sp-editor/ecblfcmjnbqajblcpmjjoameqpbodhd?ucbcb=1>
- [58] M. (2021c, octubre 12). *Introducción a Power Automate - Power Automate*. Microsoft Docs. Recuperado 31 de octubre de 2021, de <https://docs.microsoft.com/es-es/power-automate/getting-started>
- [59] *Sharegate: la herramienta de migración más sencilla*. (s. f.). Soluciones SharePoint. Recuperado 31 de octubre de 2021, de <https://www.soluciones-sharepoint.com/2017/07/sharegate-la-herramienta-de-migracion.html>
- [60] colaboradores de Wikipedia. (2021, 6 diciembre). *Reglamento General de Protección de Datos*. Wikipedia. Recuperado 19 de diciembre de 2021, de https://es.wikipedia.org/wiki/Reglamento_General_de_Protecci%C3%B3n_de_Datos
- [61] ShareGate. (2021, 4 octubre). *Subscription Pricing*. Recuperado 19 de diciembre de 2021, de <https://sharegate.com/pricing>
- [62] *Asignaturas*. (s. f.). Facultad de Informática de Barcelona. Recuperado 27 de diciembre de 2021, de <https://www.fib.upc.edu/es/estudios/grados/grado-en-ingenieria-informatica/plan-de-estudios/asignaturas>
- [63] colaboradores de Wikipedia. (2019, 10 septiembre). *Solicitud de propuestas*. Wikipedia, la enciclopedia libre. Recuperado 27 de diciembre de 2021, de https://es.wikipedia.org/wiki/Solicitud_de_propuestas