



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA SOFTWARE

**CONSTRUCCIÓN DE UNA HERRAMIENTA QUE PERMITA LA
COMUNICACIÓN, LA CONTRIBUCIÓN Y LA INTEGRACIÓN EN
LA TRAZABILIDAD DE REQUISITOS EN PROYECTOS DE
SOFTWARE**

Trabajo de Integración Curricular

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

INGENIERO DE SOFTWARE

AUTOR:

JOSÉ ALEXANDER REQUELME ROMERO

Riobamba – Ecuador

2022



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA SOFTWARE

**CONSTRUCCIÓN DE UNA HERRAMIENTA QUE PERMITA LA
COMUNICACIÓN, LA CONTRIBUCIÓN Y LA INTEGRACIÓN EN
LA TRAZABILIDAD DE REQUISITOS EN PROYECTOS DE
SOFTWARE**

Trabajo de Integración Curricular

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

INGENIERO DE SOFTWARE

AUTOR: JOSÉ ALEXANDER REQUELME ROMERO

DIRECTOR: Dr. RAÚL HERNÁN ROSERO MIRANDA

Riobamba – Ecuador

2022

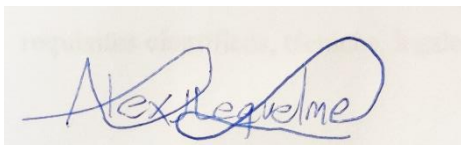
© 2022, José Alexander Requelme Romero

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, José Alexander Requelme Romero, declaro que el presente trabajo de integración curricular es de mi autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este trabajo de integración curricular; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 13 de julio de 2022



José Alexander Requelme Romero

0705677318

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA SOFTWARE

El Tribunal del Trabajo de Integración Curricular certifica que: El Trabajo de Integración Curricular; Tipo: Proyecto Técnico, **CONSTRUCCIÓN DE UNA HERRAMIENTA QUE PERMITA LA COMUNICACIÓN, LA CONTRIBUCIÓN Y LA INTEGRACIÓN EN LA TRAZABILIDAD DE REQUISITOS EN PROYECTOS DE SOFTWARE**, realizado por el señor: **JOSÉ ALEXANDER REQUELME ROMERO**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	FIRMA	FECHA
Dr. Omar Salvador Gómez Gómez PRESIDENTE DEL TRIBUNAL		2022-07-13
Dr. Raúl Hernán Rosero Miranda DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR		2022-07-13
Dr. Danilo Mauricio Pastor Ramírez MIEMBRO DEL TRIBUNAL		2022-07-13

DEDICATORIA

A mis maestros y amigos que hayan contribuido a mi proceso educativo y también a aquellos que me incentivaron a mejorar mis conocimientos y habilidades.

José

AGRADECIMIENTO

A la Escuela Superior Politécnica de Chimborazo, por permitirme avanzar en mis estudios profesionales y darme la oportunidad de haber conocido personas excelentes en el transcurso de estos años.

José

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	ix
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE GRÁFICOS.....	xi
ÍNDICE DE ANEXOS	xii
RESUMEN.....	xiii
SUMMARY	xiv
INTRODUCCIÓN	1

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA	2
1.1. Antecedentes.....	2
1.2. Formulación del Problema.....	3
1.3. Sistematización del Problema	3
1.4. Justificación.....	3
1.4.1. <i>Justificación Teórica</i>	3
1.4.2. <i>Justificación Aplicativa</i>	4
1.5. Objetivos.....	5
1.5.1. <i>Objetivo General</i>	5
1.5.2. <i>Objetivos Específicos</i>	5

CAPÍTULO II

2. MARCO TEÓRICO	6
2.1. Requisitos de Software	6
2.1.1. <i>Propiedades</i>	6
2.1.2. <i>Clasificación</i>	7
2.2. Especificación de Requerimientos de Software.....	8
2.2.1. <i>Atributos</i>	8
2.3. Gestión de Requisitos	9
2.3.1. <i>Criterios y requisitos para herramientas de gestión de requisitos</i>	9
2.4. Trazabilidad de Requisitos	10
2.4.1. <i>Proceso</i>	11

2.4.2.	<i>Escenarios de uso de la trazabilidad</i>	12
2.4.3.	<i>Propiedades de la trazabilidad</i>	12
2.4.4.	<i>Diversas fuentes de problemas de la trazabilidad</i>	13
2.4.5.	<i>Técnicas</i>	13
2.4.6.	<i>Métodos</i>	14
2.5.	Herramientas para el desarrollo	15
2.5.1.	<i>NestJS</i>	15
2.5.2.	<i>React</i>	16
2.5.3.	<i>Git</i>	16
2.5.4.	<i>PostgreSQL</i>	17
2.6.	Metodología de Desarrollo	17
2.6.1.	<i>Metodología Scrum</i>	17
2.6.1.1.	<i>Principios</i>	18
2.6.1.2.	<i>Roles</i>	19
2.6.1.3.	<i>Eventos</i>	19
2.6.1.4.	<i>Artefactos</i>	19
2.7.	Productividad	20
2.7.1.	<i>Output</i>	20
2.7.2.	<i>Input</i>	20
2.7.3.	<i>Coeficiente de productividad</i>	21
2.8.	Trabajos Relacionados	21

CAPÍTULO III

3.	MARCO METODOLÓGICO	24
3.1.	Diseño de la investigación	24
3.1.1.	<i>Tipo de investigación</i>	24
3.1.2.	<i>Métodos y Técnicas</i>	24
3.2.	Recursos que emplear para el desarrollo	26
3.2.1.	<i>Recursos software</i>	26
3.2.2.	<i>Recursos hardware</i>	26
3.3.	Desarrollo de la herramienta utilizando la metodología SCRUM	27
3.3.1.	<i>Fase de planificación</i>	27
3.3.1.1.	<i>Personas y roles involucrado en el proyecto</i>	27
3.3.1.2.	<i>Tipos de usuarios y roles involucrados en el sistema</i>	28
3.3.1.3.	<i>Product backlog</i>	28

3.3.2. Fase de diseño	30
3.3.2.1. <i>Arquitectura del sistema</i>	30
3.3.2.2. <i>Diagramas de secuencia UML</i>	31
3.3.2.3. <i>Estándar de Codificación</i>	31
3.3.2.4. <i>Diseño de la base de datos</i>	32
3.3.2.5. <i>Diccionario de datos</i>	33
3.3.2.6. <i>Diseño de interfaces de usuario</i>	33
3.3.3. Fase de desarrollo	34
3.3.3.1. <i>Sprint Backlog</i>	35
3.3.3.2. <i>Historia de usuario</i>	36

CAPÍTULO IV

4. RESULTADOS	38
4.1. Población	38
4.2. Muestra.....	38
4.3. Criterio de Evaluación	39
4.4. Productividad.....	39
4.4.1. <i>Análisis del proceso tradicional</i>	40
4.4.2. <i>Análisis del proceso con la herramienta desarrollada</i>	41
4.4.3. <i>Comparación de resultados</i>	43
4.5. Análisis de los resultados obtenidos	45

CONCLUSIONES.....	47
--------------------------	-----------

RECOMENDACIONES.....	48
-----------------------------	-----------

GLOSARIO

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

Tabla 1-2:	Criterios y requisitos para herramientas de gestión de requisitos.....	10
Tabla 2-2:	Técnicas para la trazabilidad de requisitos.	13
Tabla 3-2:	Métodos para la trazabilidad de requisitos.	15
Tabla 4-2:	Análisis de las herramientas.	22
Tabla 1-3:	Métodos, técnicas y fuentes.....	24
Tabla 2-3:	Recursos software.....	26
Tabla 3-3:	Recursos hardware.....	27
Tabla 4-3:	Personas y roles involucrados en el proyecto.	27
Tabla 5-3:	Tipos de usuarios y roles involucrados en el sistema.	28
Tabla 6-3:	Product Backlog.	28
Tabla 7-3:	Estándar de codificación.....	32
Tabla 8-3:	Diccionario de datos de la tabla Users.....	33
Tabla 9-3:	Sprint Backlog.	35
Tabla 10-3:	Historia de Usuario 01.....	37
Tabla 1-4:	Lista de historias de usuario para el caso de estudio.	38
Tabla 2-4:	Criterio de evaluación.....	39
Tabla 3-4:	Tiempo trabajado por actividad del Equipo A.....	40
Tabla 4-4:	Tiempo total trabajado por el Equipo A.	41
Tabla 5-4:	Tiempo trabajado por actividad del Equipo B.....	42
Tabla 6-4:	Tiempo total trabajado por el Equipo B.	43
Tabla 7-4:	Comparación de resultados.....	43
Tabla 8-4:	Productividad de los equipos de desarrollo.	45
Tabla 9-4:	Nivel de mejora de la productividad.....	46

ÍNDICE DE FIGURAS

Figura 1-2:	Flujo de Scrum para un sprint.....	18
Figura 1-3:	Diagrama de despliegue.....	30
Figura 2-3:	Diagrama de secuencia.	31
Figura 3-3:	Diseño físico de la base de datos.	32
Figura 4-3:	Pantalla de inicio de sesión.....	34
Figura 5-3:	Pantalla principal de la herramienta.....	34

ÍNDICE DE GRÁFICOS

Gráfico 1-4:	Comparación del tiempo trabajado por historias de usuario.	44
Gráfico 2-4:	Comparación de la productividad.....	45

ÍNDICE DE ANEXOS

ANEXO A: DIAGRAMAS DE SECUENCIA

ANEXO B: DICCIONARIO DE DATOS

ANEXO C: HISTORIAS DE USUARIO Y TÉCNICAS

ANEXO D: PLANTILLA IEEE 830 QUE SOPORTA TRAZABILIDAD DE REQUISITOS

RESUMEN

En el presente trabajo de integración curricular se desarrolló una herramienta que permita la comunicación, la contribución y la integración en la trazabilidad de requisitos en proyectos de software con el fin de medir el nivel de mejora de la productividad en el ciclo de vida del desarrollo de software al utilizar la herramienta. Para el desarrollo de la herramienta se empleó la metodología ágil SCRUM, la cual se la dividió en 3 fases: planificación, diseño y desarrollo. En la fase de planificación se definieron las personas y roles del proyecto, así como la creación del product backlog. En la fase de diseño se definió la arquitectura del sistema, el diseño de la base de datos, el estándar de codificación, así como también las interfaces de usuario. Para la fase de desarrollo se trabajó un total de 6 sprints con una duración de 2 semanas por sprint en los cuales se desarrollaron un total de 25 historias de usuario y 6 historias técnicas. Una vez finalizada la herramienta se procedió a hacer la medición del nivel de mejora de la productividad empleando el estándar IEEE Std 1045-1992, para lo cual se midió la productividad de 2 equipos de trabajo, en el cual un equipo de trabajo realizó las actividades propuestas con la herramienta desarrollada y el otro equipo la realizó sin ella, obteniendo como resultado final un nivel de mejora de la productividad del 34.52% al utilizar la herramienta desarrollada. Se concluye que la herramienta mejora la productividad de los equipos de desarrollo que lo utilizan y que además facilita la mantenibilidad de la trazabilidad de requisitos en proyectos de software. Se recomienda implementar un sistema de mensajería instantánea para incrementar la comunicación entre los miembros de un equipo de desarrollo.

Palabras clave: <INGENIERÍA DE SOFTWARE> <TRAZABILIDAD DE REQUISITOS>
<PRODUCTIVIDAD> <GESTIÓN DE REQUISITOS> <METODOLOGÍA ÁGIL SCRUM>


1613-DBRA-UTP-2022



SUMMARY

We developed a tool in this curriculum integration work to enable communication, contribution, and integration in the traceability of requirements in software projects in order to measure the level of productivity improvement in the software development life cycle by utilizing the tool. We utilized the agile SCRUM methodology for the development of the tool, which was divided into 3 phases: planning, design, and development. In the planning phase, the people and roles of the project were defined, as well as the creation of the product backlog. In the design phase, the system architecture, the database design, the coding standard, as well as the user interfaces were defined. We worked on a total of 6 sprints with a duration of 2 weeks per sprint for the development phase, in which a total of 25 user stories and 6 technical stories were developed. Once the tool was finished, the level of productivity improvement was measured utilizing the IEEE Std 1045-1992 standard, for which the productivity of 2 work teams was measured, in which one work team performed the proposed activities with the developed tool and the other team performed them without it, obtaining as a result a level of productivity improvement of 34.52% when utilizing the developed tool. It is concluded that the tool improves the productivity of the development teams that utilize it and that it also facilitates the maintainability of the traceability of requirements in software projects. It is recommended to implement an instant messaging system to increase communication between members of a development team.

Key words: <SOFTWARE ENGINEERING> <REQUIREMENTS TRACEABILITY>
<PRODUCTIVITY> <REQUIREMENTS MANAGEMENT> <AGILE SCRUM
METHODOLOGY>



Diana Carolina Campaña Díaz
180419148-2

INTRODUCCIÓN

En la Ingeniería de Software una práctica que se ha vuelto importante de mantener es la trazabilidad de requisitos, ya que ayuda a obtener el producto en el dominio de la solución lo más exacto y fiable posible a las necesidades expresadas por las partes interesadas. Es notorio que con el paso del tiempo un producto de software puede llegar a ser susceptible a cambiar su estructura y composición en espacio y tiempo durante el proceso de desarrollo de software, y es en este momento donde la trazabilidad adquiere una gran importancia. Pero gestionar esta trazabilidad puede llegar a ser muy compleja si lo realizamos de forma manual y sin ayuda de una herramienta que facilite y hasta cierto punto automatice este proceso.

Es por este motivo que se propone construir una herramienta que automatice y facilite la trazabilidad de requisitos en los proyectos de software, y que además de esto permita la contribución y comunicación incentivando así el trabajo colaborativo. Además de la trazabilidad y el trabajo colaborativo, la herramienta se enfoca en la gestión de cambios y comentarios, documentación de la historia y generación de documentación.

El objetivo del presente trabajo de integración curricular es desarrollar una herramienta que permita la comunicación, la contribución y la integración en la trazabilidad de requisitos en proyectos de software con el fin de medir el nivel de mejora de la productividad en el ciclo de vida del desarrollo de software al utilizar la herramienta desarrollada.

Este documento está estructurado en cuatro capítulos, donde el primer capítulo contiene la problemática, los antecedentes, justificaciones y objetivo; en el segundo se realiza una revisión de la literatura, en el tercero se definen la metodología del trabajo y se realizó el desarrollo de la herramienta utilizando la metodología Scrum, y finalmente en el cuarto capítulo se presentan los resultados del trabajo de integración curricular y se da respuesta a las preguntas de investigación.

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA

En la presente sección se analizan los antecedentes del trabajo de integración curricular, la formulación del problema, justificación del trabajo y los objetivos planteados.

1.1 Antecedentes

En la actualidad vivimos en un ambiente de constante cambio e innovación, en la cual la cantidad de proyectos de software que se desarrollan son innumerables, estos proyectos con el paso del tiempo tienden a crecer o ir cambiando sus funcionalidades a demanda de sus usuarios, lo cual a medio y largo plazo genera problemas relacionados a la calidad del software y su mantenibilidad.

Una pieza clave para mitigar estos problemas es la trazabilidad de requisitos, la cual hace referencia a la capacidad de describir y seguir la vida de un requisito, tanto hacia adelante como hacia atrás; es decir, desde sus orígenes, a través de su desarrollo y especificación, hasta su posterior desarrollo y uso, y a través de períodos de refinamiento e iteración continuos en cualquiera de estas fases (Gotel y Finkelstein 1994). La trazabilidad de requisitos mitiga estos problemas, debido a que con una buena trazabilidad podemos vincular las funcionalidades del software con sus requerimientos respectivos, y gracias a esto podremos reaccionar de mejor manera a los cambios o nuevas funcionalidades que se vayan a implementar en dicho software.

Desafortunadamente, a la trazabilidad de requisitos no se le da la debida importancia que merece, y en los casos donde si se lo hace, nos encontramos con el inconveniente de que mantener esta trazabilidad se vuelve más complejo a medida que el software va escalando. Dicho lo anterior, en sistemas o proyectos de software donde se gestiona una pequeña cantidad de requisitos, (Sommerville 2014) nos indica que uno puede utilizar una hoja de cálculo como Excel para crear y mantener matrices de trazabilidad, sin embargo, esto se vuelve difícil de manejar y costoso de mantener para sistemas grandes con muchos requisitos.

Por todos estos motivos mencionados anteriormente, en el presente trabajo se propone la construcción de una herramienta de software que facilite la mantenibilidad de la trazabilidad de requisitos y a su vez permita la comunicación, la contribución y la integración de esta trazabilidad en proyectos de software, la cual nombraremos “Next Requirements”. Cabe recalcar que esta

herramienta estará disponible para cualquier persona que tenga acceso a internet, puesto a que es una herramienta que utiliza la tecnología web, por lo que es agnóstica al sistema operativo.

1.2 Formulación del Problema

¿Cuál es el nivel de mejora de la productividad en el ciclo de vida del desarrollo de software al utilizar una herramienta que permita la comunicación, la contribución y la integración en la trazabilidad de requisitos?

1.3 Sistematización del Problema

¿Cuáles son las técnicas o modelos utilizados para la representación de la trazabilidad de requisitos?

¿Cómo se va a desarrollar las funcionalidades del software?

¿Cómo se va a medir el nivel de productividad del software?

1.4 Justificación

1.4.1 *Justificación Teórica*

La trazabilidad de requisitos es de suma importancia en los proyectos de software ya que se encarga de recuperar el origen de los requisitos y predecir sus efectos, por ello es fundamental para realizar un análisis de impacto cuando los requisitos cambian (IEEE Computer Society, Bourque y Fairley 2014).

Para que la trazabilidad de requisitos funcione de la manera esperada necesitamos mantener su información actualizada, ya que como nos indica (IEEE Computer Society, Bourque y Fairley 2014) si la información de trazabilidad no se actualiza a medida que se producen cambios en los requisitos, la información de trazabilidad deja de ser fiable para el análisis de impacto.

Con el transcurso del tiempo los proyectos de software tienden a la complejidad, esto debido al aumento de funcionalidades y mejora del software, debido a esto (Hoffmann et al. 2004) nos señala que para hacer frente a dicha complejidad, debe establecerse una herramienta software que nos permita a grandes rasgos la gestión de requisitos. También nos recalca que dicha herramienta debería ser accesible a través de la web para así evitar tener que instalar el software para usuarios que son ocasionales.

Debido a lo revisado anteriormente la herramienta por desarrollarse será una aplicación web, ya que gracias a esta tecnología nosotros podemos abarcar los sistemas operativos más conocidos, como por ejemplo Windows, Mac y Linux gracias al uso de un navegador web, además de esto nos permite tener una consistencia en el diseño sin importar el sistema operativo en el que se utilice.

Además, para el desarrollo de frontend se utilizará la librería de React ya que nos permite desarrollar interfaces de usuario interactivas de manera sencilla, además de esto, está basado en componentes por lo que podemos construir componentes encapsulados que manejen sus estados y luego componerlos con otros componentes para crear interfaces de usuario más complejas, y también se integra perfectamente con otras tecnologías de desarrollo, por lo que se puede desarrollar nuevas funciones en React sin reescribir el código existente (Facebook Inc 2021).

En cuanto al backend se utilizará el framework NestJS, puesto que nos ofrece ciertas características como: flexibilidad en el desarrollo, versatilidad y escalabilidad (Mysliwiec 2021). Además, comparte el mismo lenguaje de desarrollo que React el cual es TypeScript, lo que nos permite trabajar con el mismo ambiente de desarrollo tanto para el frontend como para el backend.

Para el control de versiones de software se utilizará git, ya que es fácil de aprender y ocupa poco espacio con un rendimiento ultrarrápido, también es de código abierto y gratuito diseñado para manejar todo, desde proyectos pequeños hasta muy grandes, con velocidad y eficiencia (Chacon 2021).

1.4.2 *Justificación Aplicativa*

La herramienta se centra en la gestión de requisitos y la trazabilidad de estos en proyectos de software, además permitirá la comunicación y contribución a través de todo este proceso.

Debido a que, al realizar una correcta trazabilidad de requisitos, nos será posible responder a los cambios de forma más controlada y con más información, y en consecuencia anticiparnos a lo que un cambio puede significar. Por lo que una herramienta que nos permita la trazabilidad de requisitos significaría una mejora en la productividad del proyecto, puesto que podremos calcular con mayor precisión el tiempo que se tardaría en implementar una nueva funcionalidad al programa.

En cuanto a la funcionalidad de comunicación y contribución está enfocada a mitigar los problemas que surgen en los equipos de desarrollo de software debido a esta falta de comunicación y colaboración entre los miembros del equipo, lo que ayudará en la productividad del proceso de desarrollo y repercutirá en la calidad del software desarrollado.

El presente trabajo está bajo la línea de investigación de Tecnología de la Información Y Comunicación (Tic's) en el programa de Ingeniería de Software cuyo ámbito es el Análisis y Diseño de Software, Verificación y Validación de Software. En el plan nacional de desarrollo se encuentra en el eje social, aplicando el objetivo 7 referente a potenciar las capacidades de la ciudadanía y promover una educación innovadora, inclusiva y de calidad en todos los niveles.

1.5 Objetivos

1.5.1 *Objetivo General*

Construir una herramienta que permita la comunicación, la contribución y la integración en la trazabilidad de requisitos en proyectos de software con el fin de aumentar el nivel de productividad en el ciclo de vida del desarrollo de software.

1.5.2 *Objetivos Específicos*

- Identificar las diferentes técnicas o métodos para representar la trazabilidad de requisitos.
- Implementar las funcionalidades de la herramienta para permitir la comunicación, contribución e integración en la trazabilidad de requisitos en proyectos de software.
- Medir el nivel de mejora de la productividad en el ciclo de vida del desarrollo de software al utilizar la herramienta desarrollada.

CAPÍTULO II

2. MARCO TEÓRICO

2.1 Requisitos de Software

En su forma más básica, un requisito de software es una propiedad que debe exhibir algo para resolver algún problema en el mundo real, estos requisitos suelen ser una combinación compleja de varias personas en diferentes niveles de una organización, y que de una forma u otra están involucradas o conectadas con esta característica del entorno en el que operará el software (IEEE Computer Society, Bourque y Fairley 2014).

Para (Sommerville 2016) los requisitos de un sistema son las descripciones de los servicios que debe proporcionar un sistema y las restricciones sobre su funcionamiento. Estos requisitos reflejan las necesidades de los clientes de un sistema que sirva para un determinado propósito, como controlar un dispositivo, realizar un pedido o buscar información. Nos indica que, en algunos casos, un requisito es simplemente una declaración abstracta de alto nivel de un servicio que un sistema debe proporcionar o una restricción en un sistema y que, por otro lado, es una definición formal y detallada de una función del sistema.

2.1.1 *Propiedades*

Según (IEEE SA 1998), las propiedades que deben tener una colección de requisitos son las siguientes:

- **Conjunto único:** Cada requisito debe indicarse una sola vez.
- **Normalizado:** Los requisitos no deben superponerse (es decir, no deben hacer referencia a las capacidades de otros requisitos).
- **Conjunto vinculado:** Deben definirse relaciones explícitas entre los requisitos individuales para mostrar cómo se relacionan los requisitos para formar un sistema completo.
- **Completo:** Debe incluir todos los requisitos identificados por el cliente, así como los necesarios para la definición del sistema.
- **Consistente:** El contenido debe ser consistente y no contradictorio en el nivel de detalle, el estilo de las declaraciones de requisitos y en la presentación del material.
- **Encerrado:** Deben identificarse los límites, el alcance y el contexto del conjunto de requisitos.

- **Modificable:** Debe ser modificable, los requisitos de claridad y no superposición contribuyen a esto.
- **Configurable:** Las versiones deben mantenerse a lo largo del tiempo y en todas las instancias.
- **Granular:** Este debería ser el nivel de abstracción del sistema que se está definiendo.

2.1.2 *Clasificación*

Según la (IEEE Computer Society, Bourque y Fairley 2014) los requisitos se pueden clasificar en varias dimensiones, las cuales son las siguientes:

- Si el requisito es funcional o no funcional.
- Si el requisito se deriva de uno o más requisitos de alto nivel o de una propiedad emergente, o si una parte interesada o alguna otra fuente lo impone directamente al software.
- Si el requisito es sobre el producto o el proceso, los requisitos del proceso pueden limitar la elección del contratista, el proceso de ingeniería de software que se adoptará o los estándares que se cumplirán.
- La prioridad del requisito, cuanto mayor sea la prioridad, más esencial será el requisito para cumplir los objetivos generales del software. A menudo clasificados en una escala de punto fijo como obligatorio, altamente deseable, deseable u opcional, la prioridad a menudo tiene que equilibrarse con el costo de desarrollo e implementación.
- El alcance del requisito, el cual se refiere a la medida en que un requisito afecta el software y los componentes del software. Algunos requisitos, en particular algunos no funcionales, tienen un alcance global en el sentido de que su satisfacción no puede asignarse a un componente discreto. Por lo tanto, un requerimiento con alcance global puede afectar fuertemente la arquitectura del software y el diseño de muchos componentes, mientras que uno con un alcance limitado puede ofrecer varias opciones de diseño y tener poco impacto en la satisfacción de otros requisitos.
- La volatilidad/estabilidad del requisito, durante el ciclo de vida del software e incluso durante el propio proceso de desarrollo, algunos requisitos posiblemente cambiarán, por ello, es de gran utilidad realizar alguna estimación de la probabilidad de que un requisito cambie, debido a que, si se necesita que un ingeniero de software establezca un diseño que sea más tolerante al cambio, necesitaremos previo a esto identificar los requisitos que son más potencialmente volátiles.

2.2 Especificación de Requerimientos de Software

El documento de requisitos o especificación de requisitos de software establece la base para el acuerdo entre clientes y contratistas o proveedores sobre lo que debe hacer el producto de software y lo que no se espera que haga, lo cual nos permite una evaluación rigurosa de los requisitos antes de que pueda comenzar el diseño y también reduce el rediseño posterior, además de esto la especificación de requisitos debe proporcionar una base realista para estimar los costos, riesgos y cronogramas del producto (IEEE Computer Society, Bourque y Fairley 2014).

2.2.1 Atributos

Según (Bokhari y Siddiqui 2011) los atributos de los documentos de requisitos de software, describen cómo deben especificarse los requisitos, qué deben incluir y cómo afectan a otros atributos, dichos atributos son los siguientes:

- **Inequívoco:** Un documento de requisitos debe ser inequívoco, es decir que no debe presentar ninguna ambigüedad.
- **Correcto:** Un documento de requisitos es correcto si y solo si cada oración o requisito mencionado en el documento es creado por el sistema.
- **Completo:** Se dice que un documento de requisitos está completo si el documento incluye todo lo que se supone que debe hacer el software y si existe para cada entrada del sistema alguna salida.
- **Comprensible:** Un documento de requisitos debe ser comprensible para todos los lectores, incluidos clientes, usuarios, directores de proyectos, etc.
- **Verificable:** Un documento de requisitos es verificable si existe un proceso finito rentable con el que una persona o máquina puede verificar que el producto de software creado requisitos.
- **Coherencia interna:** Se dice que un documento de requisitos es coherente internamente si ningún subconjunto de requisitos entra en conflicto, como enfocar, arrastrar y presionar con otros requisitos de un conjunto.
- **Modificable:** Se dice que un documento de requisitos es modificable si los cambios se pueden realizar de manera consistente. Las mejoras y la eliminación de errores también son importantes para la modificación de los documentos.
- **Anotado por importancia relativa:** Se dice que un documento de requisitos está anotado por importancia relativa si sus características están en orden ascendente según su importancia para los clientes. Se pueden marcar como obligatorias, recomendadas y opcionales.

- **Anotado por estabilidad relativa:** Los documentos de requisitos se pueden anotar por estabilidad relativa según la ayuda de los diseñadores. Los cambios de requisitos son la causa de la estabilidad de las características, como las que pueden ser propensas a cambios tempranos o las que pueden permanecer durante un período más largo.
- **Anotado por versión:** Se dice que un documento de requisitos está anotado por versión si el lector puede determinar qué funciones se pueden implementar en qué versión del software. Este atributo puede depender de la aplicación.
- **Preciso:** Un documento de requisitos es preciso si no debe contener detalles vagos. Por ejemplo, el sistema no debe hacer que el usuario espere el reconocimiento, pero debe tener cierto tiempo como dos minutos.
- **Rastreado:** Un documento de requisitos se rastrea si el origen de cada uno de sus requisitos tiene referencias a versiones anteriores de los documentos de respaldo.
- **Rastreable:** Se dice que un documento de requisitos es rastreable si tiene una referencia de cada requisito establecido. Esto se puede lograr numerando todos los párrafos y requisitos de manera única.

2.3 Gestión de Requisitos

La gestión de requisitos es un proceso de gestión de cambios en los requisitos, estos cambios son inevitables debido a errores del sistema y una mejor comprensión del desarrollo de las necesidades reales de los clientes. La gestión de requisitos se lleva a cabo en paralelo con otras actividades de requisitos en el desarrollo de software. Las actividades de gestión de requisitos incluyen mantener el plan del proyecto sin fecha con los requisitos, controlar las versiones de requisitos, rastrear el estado de los requisitos y rastrear los requisitos (Bokhari y Siddiqui 2011).

2.3.1 Criterios y requisitos para herramientas de gestión de requisitos

Según la (IEEE Computer Society, Bourque y Fairley 2014) las herramientas de gestión de requisitos suelen dar soporte a una serie de actividades que han tenido un impacto significativo en la práctica, tales como: la documentación, el seguimiento y la gestión del cambio.

De modo similar (Hoffmann et al. 2004) profundiza sobre los criterios y requisitos que una herramienta para la gestión de requisitos debería cumplir, los cuales se detallan a continuación en la **Tabla 1-2**.

Tabla 1-2: Criterios y requisitos para herramientas de gestión de requisitos.

Criterio/Requisito	Descripción	Consideraciones
Gestión de cambios y comentarios	La herramienta debe ofrecer la posibilidad de manejar solicitudes de cambio formales.	Esta función debe ser personalizable al proceso de cambio de los usuarios y debe estar integrada en la gestión de derechos.
Documentación de la historia	Todos los cambios en los requisitos deben ser rastreados y guardados en la base de datos.	Los objetos en la herramienta deben estar versionados y el número de versión debe incrementarse automáticamente cuando ocurren ciertos cambios
Trazabilidad	La herramienta debe permitir la trazabilidad a través de enlaces entre requisitos.	La vinculación debe implementarse de manera sencilla y fácil de usar porque solo ayuda si es relativamente completa.
Generación de documentación	La herramienta debe ser capaz de generar documentos oficiales e internos.	Se deben utilizar plantillas y se generarán con los datos actuales de la base de datos.
Trabajo colaborativo en la misma tarea de desarrollo	Debe ser posible que muchos usuarios trabajen con los mismos datos al mismo tiempo.	Se debe garantizar la consistencia e integridad de los cambios realizados sobre un objeto.

Fuente: (Hoffmann et al. 2004).

2.4 Trazabilidad de Requisitos

La trazabilidad de los requisitos se refiere a la capacidad de describir y seguir la vida de un requisito, tanto hacia adelante como hacia atrás; es decir, desde sus orígenes, a través de su desarrollo y especificación, hasta su posterior implementación y uso, y a través de todos los períodos de refinamiento y elementos en curso en cualquiera de estas fases (Gotel y Finkelstein 1994).

Según (IEEE Computer Society, Bourque y Fairley 2014) la trazabilidad de requisitos se ocupa de recuperar la fuente de los requisitos y predecir los efectos de los requisitos, dicha trazabilidad es fundamental para la realización de análisis de impacto cuando cambian los requisitos.

Otros autores como (Sommerville 2014) nos indica que la trazabilidad de requisitos es la propiedad de una especificación de requisitos, la cual refleja la facilidad de encontrar requisitos

relacionados. Así mismo para (Кулямин et al. 2006), la trazabilidad de requisitos es la capacidad de establecer vínculos entre los requisitos y sus fuentes, por un lado, así como con las secciones y elementos que surgen durante el desarrollo de textos para programas, documentos y modelos, por otro lado.

2.4.1 Proceso

Referente al proceso de la trazabilidad, la (IEEE Computer Society, Bourque y Fairley 2014) nos indica que para que un requisito pueda ser rastreable de manera eficiente debe cumplir lo siguientes puntos:

- Rastreable hacia atrás, hasta los requisitos y las partes interesadas que lo motivaron.
- Rastreable hacia adelante, desde los requisitos y las entidades de diseño que lo satisfacen hasta el caso de prueba que lo verifica.

En cuanto a la información que se debe mantener en dicha trazabilidad, (Sommerville 2014) nos propone tres tipos los cuales se indican a continuación:

- La información de trazabilidad de origen que vincula los requisitos con las partes interesadas que propusieron los requisitos y con la justificación de estos requisitos. Cuando se propone un cambio, se utiliza esta información para descubrir las partes interesadas para que puedan ser consultadas sobre el cambio.
- La información de trazabilidad de requisitos que vincula los requisitos dependientes dentro del documento de requisitos. Se utiliza esta información para evaluar cuántos requisitos es probable que se vean afectados por un cambio propuesto y el alcance de los cambios de requisitos consiguientes que pueden ser necesarios.
- La información de trazabilidad del diseño que vincula los requisitos con los módulos de diseño donde se implementan estos requisitos. Se utiliza esta información para evaluar el impacto de los cambios de requisitos propuestos en el diseño y la implementación del sistema.

Además de todo lo visto anteriormente, la (IEEE Computer Society, Bourque y Fairley 2014) nos recalca que mantener un gráfico de trazabilidad o una matriz de trazabilidad actualizado es una actividad que debe considerarse durante todo el ciclo de vida de un producto, ya que si la información de trazabilidad no se actualiza a medida que se producen cambios en los requisitos, la información de trazabilidad deja de ser fiable para el análisis de impacto.

2.4.2 Escenarios de uso de la trazabilidad

El autor (Misselwitz 2012) identifica algunos escenarios de uso para la trazabilidad de requisitos, entre los cuales tenemos los siguientes:

- Seguimiento de estado de requerimientos.
- Prueba de que se han implementado todos los requisitos descritos.
- Desarrollo de casos de prueba (directamente de los requisitos).
- Comprender cómo surgieron los requisitos individuales (por ejemplo, regulaciones, partes interesadas).
- Planificación de lanzamientos.
- Perfeccionamiento de los requisitos.
- Solución de problemas después de las pruebas.
- Estimación de esfuerzo para solicitudes de cambio.

También nos recalca que, en la práctica, el alto grado de vinculación necesario entre los requisitos solo puede garantizarse con la ayuda de un software especial.

2.4.3 Propiedades de la trazabilidad

Referente a las propiedades de la trazabilidad (Misselwitz 2012) nos detalla una lista completa de estas, las cuales son las siguientes:

- **Basado en las necesidades (propósito):** La trazabilidad respalda los requisitos de las partes interesadas.
- **Rentable:** Costos y beneficios se encuentran en una adecuada relación.
- **Configurable:** La trazabilidad se puede establecer según sea necesario y adaptarse a los requisitos cambiantes.
- **Confiable:** Todas las partes interesadas confían en la trazabilidad definida, establecida y mantenida ante inconsistencias, omisiones y cambios. Puede confiar y confiará en la trazabilidad proporcionada.
- **Escalable:** Se pueden rastrear diferentes tipos de artefactos en cantidades variables utilizando niveles variables de granularidad, mientras que la trazabilidad se extiende más allá de los límites organizacionales y comerciales.
- **Portátil:** La trazabilidad se puede intercambiar, agregar y reutilizar, dentro y entre proyectos, organizaciones, dominios, líneas de productos y herramientas de soporte.
- **Valorado:** La trazabilidad es una prioridad estratégica y es valorada por todos. Cada parte interesada tiene un papel que desempeñar y lleva a cabo activamente sus responsabilidades.

- **Omnipresente (ubicuo):** La trazabilidad está presente en todo momento sin que su presencia se perciba activamente ya que está integrada en el proceso de ingeniería del software.

2.4.4 *Diversas fuentes de problemas de la trazabilidad*

En cuanto a los problemas de la trazabilidad, (Gotel 1995) nos indica que existen diversas fuentes las cuales son las siguientes:

- La existencia de múltiples documentos incompatibles y fragmentados, de fuentes distribuidas, sin una relación clara con una Especificación de Requerimientos unificada.
- La incapacidad para manejar la creciente cantidad de documentación.
- El cambio, y la lentitud con que se toman en cuenta todas sus ramificaciones, lo que da lugar a numerosas versiones de documentos en diversas etapas de evolución.
- La falta de un proceso de trazabilidad de extremo a extremo, además de la ausencia de una descripción específica del trabajo de la trazabilidad, lo que lleva a una mala gestión de esta.
- La participación de demasiadas personas, a menudo poco cooperativas, con experiencia y agendas individuales inadecuadas.

2.4.5 *Técnicas*

De acuerdo a (Gotel 1995) existen una serie de técnicas básicas mediante las cuales se puede lograr activamente la trazabilidad de requisitos. Estos distintos tipos de técnicas difieren en la cantidad y diversidad de información que pueden rastrear entre sí, en el número de interconexiones que pueden representar entre la información y en la medida en que pueden adaptarse para reflejar los cambios y mantener así la trazabilidad a lo largo de la vida de un proyecto.

En la **Tabla 2-2** se detalla cada una de las técnicas utilizadas para la trazabilidad de requisitos, así como con sus respectivos ejemplos.

Tabla 2-2: Técnicas para la trazabilidad de requisitos.

Técnica	Descripción	Ejemplos
Centrado en la referencia cruzada	Por esquemas de referencias cruzadas simples, se refiere a aquellos esquemas que están integrados en la propia documentación principal del proyecto. Por esquemas de referencias cruzadas	Esquemas de etiquetado, numeración o indexación. Matrices de trazabilidad de requisitos. Secuencias de Matrices.

	más completos se entiende aquellos esquemas que complementan la documentación principal del proyecto, normalmente con la adición de tablas o matrices especializadas.	
Centrado en el documento	Por técnicas centradas en el documento, nos referimos a los esquemas que dictan toda o parte de la estructura y el contenido de la documentación del proyecto para garantizar la trazabilidad.	Plantillas Software Through Pictures (StP). Plantillas de documentos hipertextuales dentro del Document Integration Facility (DIF).
Centrado en la estructura	Por técnicas centradas en la estructura, se hace referencia a los esquemas que mejoran la documentación del proyecto para lograr la trazabilidad. Estos esquemas son especialmente evidentes cuando el objetivo de la trazabilidad es la actualización y propagación de los cambios en los requisitos.	Esquemas con estructura de red o de un gráfico. Esquemas basados en el uso de redes de mantenimiento de la verdad basadas en suposiciones y mecanismos de encadenamiento. Esquemas basados en el uso de redes de restricciones y propagación.

Fuente: (Gotel 1995).

Para el presente trabajo se seleccionó la técnica de trazabilidad centrada en la referencia cruzada, ya que mediante su implementación nos permite indexar distintos elementos que estén relacionados entre sí, lo cual nos facilita la visualización y navegación entre sus relaciones, este indexado se asemeja a un hipervínculo de una página web, por lo cual esta técnica es idónea para la herramienta debido a que dicha herramienta utiliza la tecnología web.

2.4.6 Métodos

Por métodos para la trazabilidad de requisitos, (Gotel 1995) hace referencia a la disposición ordenada de los procedimientos a seguir en el proceso de desarrollo, acompañados de directrices para su ejecución.

A continuación, en la **Tabla 3-2** se detallan los ejemplos de cómo el uso de estos métodos nos proporciona una trazabilidad de requisitos.

Tabla 3-2: Métodos para la trazabilidad de requisitos.

Método	Descripción	Técnica Relacionada
Metodología de Planificación y Diseño (PDM)	La trazabilidad resulta de las referencias cruzadas en la serie de pasos que se siguen para transformar los requisitos iniciales en un documento de especificación estructurado.	Centrado en la referencia cruzada (Esquema de referencia cruzada completa).
Metodología de Especificación y Trazabilidad de Requisitos (RESPECT)	La trazabilidad resulta de una simple referencia cruzada en línea y de matrices de trazabilidad más completas.	Centrado en la referencia cruzada (Esquema de referencia cruzada simple y matrices de trazabilidad).
Despliegue de la Función de Calidad (QFD)	La trazabilidad resulta de matrices que se combinan de extremo a extremo para formar desde "casas" hasta "ciudades" de calidad	Centrado en la referencia cruzada (Secuencias de Matrices).

Fuente: (Gotel 1995).

Para el presente trabajo se seleccionó el método de trazabilidad RESPECT, debido a que RESPECT examina la especificación de requisitos y el desafío de la trazabilidad en su totalidad, ya que tiene un enfoque a largo plazo y además garantiza la consistencia de los requisitos a través del ciclo de vida del desarrollo de software.

2.5 Herramientas para el desarrollo

En esta sección se describen las diferentes herramientas que se emplean para el desarrollo del presente trabajo.

2.5.1 NestJS

NestJS es un framework para crear aplicaciones del lado del servidor Node.js escalables y eficientes. Utiliza JavaScript progresivo, está construido con y es totalmente compatible con TypeScript y combina elementos de programación orientada a objetos, programación funcional y programación reactiva funcional (Mysliwiec 2021).

NestJS proporciona una arquitectura de aplicaciones lista para usar que está fuertemente inspirada en Angular, dicha arquitectura permite a los desarrolladores y equipos crear aplicaciones altamente comprobables, escalables, poco acopladas y fáciles de mantener (Mysliwiec 2021).

Según (Mysliwiec 2021) las principales características de NestJS son las siguientes:

- **Extensible:** Brinda verdadera flexibilidad al permitir el uso de cualquier otra biblioteca gracias a la arquitectura modular.
- **Versátil:** Un ecosistema adaptable que es una columna vertebral completa para todo tipo de aplicaciones del lado del servidor.
- **Progresivo:** Aprovecha las últimas funciones de JavaScript, brindando patrones de diseño y soluciones maduras al mundo de Node.js.

2.5.2 React

Según (Facebook Inc 2021) React es una biblioteca de JavaScript para crear interfaces de usuario, la cual tiene las siguientes características:

- Es declarativo, lo cual hace que sea sencillo crear interfaces de usuario interactivas. podemos diseñar vistas simples para cada estado en nuestras aplicaciones, y React actualizará y renderizará de manera eficiente los componentes correctos cuando cambien los datos de dicho componente.
- Está basado en componentes, podemos construir componentes encapsulados que manejen sus estados y luego componerlos con otros componentes para crear interfaces de usuario complejas.
- Se integra perfectamente con otras tecnologías de desarrollo, por lo que se puede desarrollar nuevas funciones en React sin reescribir el código existente.

2.5.3 Git

Git es un sistema de control de versiones distribuido de código abierto y gratuito diseñado para manejar todo, desde proyectos pequeños hasta muy grandes, con velocidad y eficiencia. Git es fácil de aprender y ocupa poco espacio con un rendimiento ultrarrápido. Supera a las herramientas SCM como Subversion, CVS, Perforce y ClearCase con características como bifurcaciones locales económicas, áreas de preparación convenientes y múltiples flujos de trabajo (Chacon 2021).

2.5.4 *PostgreSQL*

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto que utiliza y amplía el lenguaje SQL combinado con muchas características que almacenan y escalan de forma segura las cargas de trabajo de datos más complicadas. PostgreSQL se ha ganado una sólida reputación por su arquitectura comprobada, confiabilidad, integridad de datos, conjunto sólido de características, extensibilidad y la dedicación de la comunidad de código abierto detrás del software para ofrecer soluciones innovadoras y de rendimiento constante (The PostgreSQL Global Development Group 2021).

2.6 **Metodología de Desarrollo**

2.6.1 *Metodología Scrum*

Según (Schwaber y Sutherland 2020) Scrum es un marco de trabajo ligero y sencillo que ayuda tanto a las personas, los equipos como a las organizaciones a generar valor a través de soluciones adaptables para problemas complejos. Scrum es intencionalmente incompleto, ya que, sólo define las partes necesarias para aplicar la teoría de dicho marco de trabajo. Las reglas de Scrum prefieren guiar sus relaciones e interacciones, en vez de proporcionar a las personas instrucciones minuciosas, esto es debido a que Scrum se construye con la inteligencia colectiva de las personas que lo utilizan.

En cambio para (SCRUMstudy 2016), Scrum es una de las metodologías ágiles más populares, además es una metodología adaptativa, iterativa, rápida, flexible y eficaz, diseñada para ofrecer un valor significativo rápidamente y a lo largo de todo el proyecto. Además, nos indica que Scrum garantiza la transparencia en la comunicación y crea un entorno de responsabilidad colectiva y progreso continuo, por ello una fortaleza clave de Scrum radica en el uso de equipos multifuncionales, autoorganizados y empoderados que dividen su trabajo en ciclos de trabajo cortos y concentrados llamados Sprints.

En la **Figura 1-2**, se puede observar el flujo de trabajo realizado durante un sprint.

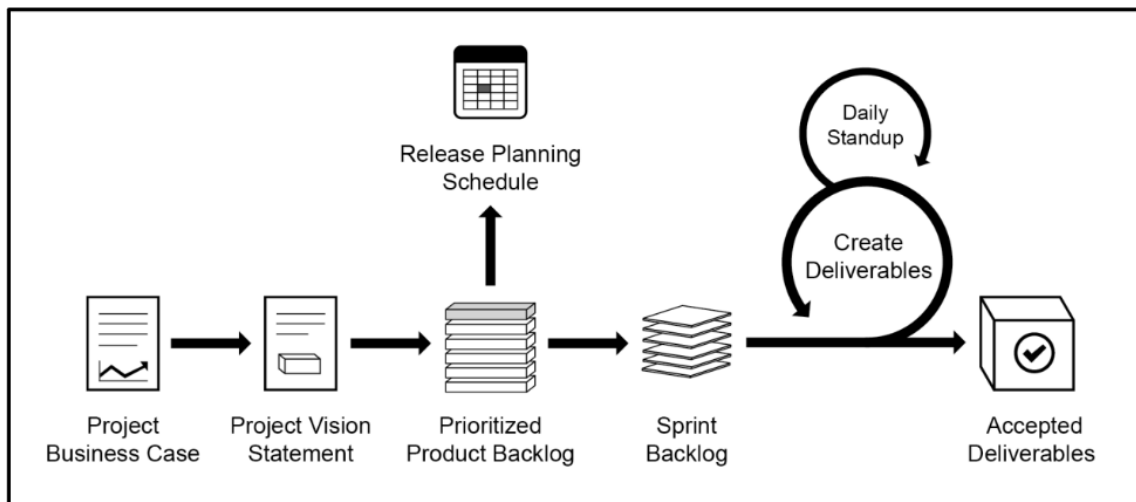


Figura 1-2. Flujo de Scrum para un sprint.

Fuente: (SCRUMstudy 2016).

2.6.1.1 Principios

Según (SCRUMstudy 2016) los principios de Scrum son seis y son las pautas centrales para aplicar el marco de trabajo, por ello deben usarse obligatoriamente en todos los proyectos de Scrum, estos principios son los siguientes:

1. **Control de procesos empíricos:** Este principio enfatiza la filosofía central de Scrum basada en las tres ideas principales de transparencia, inspección y adaptación.
2. **Autoorganización:** Este principio se enfoca en los trabajadores de hoy, quienes brindan un valor significativamente mayor cuando se autoorganizan y esto da como resultado una mejor participación del equipo y una propiedad compartida; y un entorno innovador y creativo más propicio para el crecimiento.
3. **Colaboración:** Este principio se centra en las tres dimensiones centrales relacionadas con el trabajo colaborativo: conciencia, articulación y apropiación.
4. **Priorización basada en valores:** Este principio destaca el enfoque de Scrum para ofrecer el máximo valor comercial, desde las primeras etapas del proyecto y continuando durante todo el proceso.
5. **Boxeo de tiempo:** Este principio describe cómo el tiempo se considera una restricción limitante en Scrum y se utiliza para ayudar a gestionar eficazmente la planificación y ejecución de proyectos.
6. **Desarrollo iterativo:** Este principio define el desarrollo iterativo y enfatiza cómo gestionar mejor los cambios y crear productos que satisfagan las necesidades del cliente.

2.6.1.2 Roles

Según (SCRUMstudy 2016) los principales roles de Scrum son los siguientes:

- El Product Owner es la persona responsable de lograr el máximo valor comercial para el proyecto. También es responsable de articular los requisitos del cliente y mantener la justificación comercial del proyecto. El Product Owner representa la voz del cliente.
- El Scrum Master es un facilitador que asegura que el Scrum Team cuente con un entorno propicio para completar el proyecto con éxito. El Scrum Master guía, facilita y enseña las prácticas de Scrum a todos los involucrados en el proyecto; despeja impedimentos para el equipo; y garantiza que se sigan los procesos de Scrum.
- El Scrum Team es el grupo o equipo de personas que son responsables de comprender los requisitos especificados por el Product Owner y crear los Entregables del proyecto.

2.6.1.3 Eventos

Los eventos se utilizan en Scrum para crear regularidad y minimizar la necesidad de reuniones no definidas en Scrum, cada uno de estos eventos son una oportunidad formal para inspeccionar y adaptar los artefactos Scrum, también están diseñados específicamente para permitir la transparencia requerida (Schwaber y Sutherland 2020).

Según (Schwaber y Sutherland 2020) los eventos de scrum son los siguientes:

- **Sprint:** Son eventos de duración fija de un mes o menos para crear coherencia. Un nuevo Sprint comienza inmediatamente después de la conclusión del Sprint anterior.
- **Sprint Planning:** En este evento se establece el trabajo que se realizará para el Sprint.
- **Daily Scrum:** El propósito del Daily Scrum es inspeccionar el progreso hacia el objetivo del Sprint y adaptar el Sprint Backlog según sea necesario, ajustando el próximo trabajo planificado.
- **Sprint Review:** El propósito de Sprint Review es inspeccionar el resultado del Sprint y determinar adaptaciones futuras.
- **Sprint Retrospective:** El propósito del Sprint Retrospective es planificar formas de aumentar la calidad y la eficacia.

2.6.1.4 Artefactos

Los artefactos Scrum representan trabajo o valor. Están diseñados para maximizar la transparencia de la información clave, de esta manera todos los que los inspeccionan tienen la

misma base de adaptación (Schwaber y Sutherland 2020), a continuación se detallan los siguientes artefactos de scrum:

- **Product Backlog:** El Product Backlog es una lista emergente y ordenada de lo que se necesita para mejorar el producto.
- **Sprint Backlog:** El Sprint Backlog es un plan realizado por y para los Desarrolladores. Debido a esto nos sirve como una imagen muy visible y en tiempo real del trabajo que los Desarrolladores planean realizar durante el Sprint para lograr cumplir el objetivo del Sprint.

2.7 Productividad

De acuerdo a (Wagner y Ruhe 2018), la productividad en el desarrollo de software ha sido un área de investigación importante durante varias décadas, por ello se ha vuelto fundamental que una empresa de software controle y mejore su productividad con el fin de volverse exitosa. Los autores no indican que existen varias definiciones de lo que es la productividad de las cuales la más genérica vendría hacer la relación entre las unidades de output y las unidades de input.

Muy similar a lo que nos indica la (IEEE Std 1045–1992 1992), la cual nos define la productividad como el cociente de las unidades de producción (output) divididas por las unidades de entrada (input). En donde los inputs consisten en el esfuerzo realizado para producir un producto determinado y los outputs tangibles son el código fuente y la documentación.

2.7.1 Output

Según (IEEE Std 1045–1992 1992) se definen tres categorías primitivas de output, las cuales son las siguientes:

- **Sentencias fuente:** Son una parte fundamental del software y representan la lógica codificada del producto de software.
- **Puntos de función:** Son una medida del contenido funcional del software.
- **Producción de documentos:** Es un componente importante del esfuerzo invertido en el desarrollo de software.

2.7.2 Input

Según (IEEE Std 1045–1992 1992) el input representa el esfuerzo aplicado para desarrollar el producto de software. La medida principal del esfuerzo en dicho estándar es el trabajo expresado en unidades de horas-persona. Las horas de personal recopiladas incluirán aquellas directamente

gastadas en la creación de un output de producto específico para el cual se calcula la productividad.

2.7.3 Coeficiente de productividad

Como nos indica (IEEE Std 1045–1992 1992), la productividad se calcula utilizando la relación entre la producción y el insumo para cada primitiva de producción medida para la que se recogen datos de esfuerzo de insumo relacionados. La productividad del producto a , que tiene la cantidad de primitiva de salida O_a y cuya creación ha requerido un esfuerzo E_a , se define por la siguiente relación.

$$Productivity_a = \frac{O_a}{E_a}$$

Las unidades de medida del coeficiente son las de las dimensiones de las primitivas utilizadas, por ejemplo, declaraciones fuente por hora de personal, o cantidad de documentos por hora de personal. La utilidad de los valores de productividad depende de la exactitud de los datos utilizados para calcularlos. En este caso, la exactitud se refiere al nivel de detalle y precisión del proceso de recogida de datos; las estimaciones no son aceptables. Sólo el registro del esfuerzo de entrada a medida que se trabaja y el recuento de la salida producida son métodos aceptables para medir las primitivas (IEEE Std 1045–1992 1992).

2.8 Trabajos Relacionados

A continuación, se presentan algunas herramientas con funcionalidades similares que están actualmente en el mercado.

- **Jama Connect:** Es un software que garantiza la calidad del producto y mejora la gestión de cambios con una trazabilidad completa, permitiendo mantener un riguroso proceso de gestión del cambio y revelando las interdependencias dentro del proceso para mantener los proyectos en marcha; además garantiza una colaboración segura y multifuncional entre equipos, clientes y cadenas de suministro complejas para eliminar la fricción a lo largo del proceso de desarrollo (Jama Software 2021).
- **Modern Requirements:** Es una herramienta creada para Azure DevOps, nos permite convertir un proyecto de Azure DevOps en un espacio de trabajo colaborativo, eliminando así los antiguos procesos manuales basados en copiar / pegar (Modern Requirements 2021).
- **Xebrio:** Es una de las mejores herramientas de gestión de requisitos, permite gestionar los requisitos de manera eficiente para que la ejecución del proyecto comience de manera organizada y el equipo siempre se mantenga en el camino correcto (Xebrio 2021).

- Orcanos: Es una herramienta de trazabilidad de requisitos que ayuda a las empresas a realizar un seguimiento de la cobertura y la trazabilidad entre cualquier artefacto de ALM, incluidos, entre otros, los requisitos del sistema, los requisitos de hardware y software y los casos de prueba (Orcanos 2021).
- ReqSuite® RM: Es una herramienta que ayuda a las organizaciones a analizar y gestionar los requisitos de forma más completa, sencilla y estandarizada. Como resultado, ReqSuite® RM no solo reduce los costosos reprocesos de proyectos hasta en un 40%, sino que también simplifica el análisis, la planificación y el diseño en proyectos de desarrollo o adquisiciones (OSSENO Software GmbH 2021).
- Visure Requirements ALM: Es una herramienta de gestión de requisitos completa y fácil de usar. Integra otros procesos dentro del mismo entorno, como la gestión de riesgos, la gestión de pruebas, el seguimiento de problemas y defectos y la gestión de cambios (Visure Solutions 2021).

Mediante la utilización de los criterios y requerimientos definidos en la **Tabla 1-2**, se realizó un análisis de las herramientas revisadas anteriormente, la cual se puede visualizar a continuación en la **Tabla 4-2**.

Tabla 4-2: Análisis de las herramientas.

	Jama Connect	Visure Requirements ALM	Xebrio	Orcanos	ReqSuite	Modern Requirements
Gestión de cambios y comentarios	X	X	X	X	X	X
Trazabilidad	X	X	X	X	X	X
Trabajo colaborativo	X	X	X	X	X	X
Documentación de la Historia	X	X	X	X	X	X
Generación de documentos		X		X		X

Realizado por: Requielme José, 2022.

Como podemos apreciar en la **Tabla 4-2** la mayoría de las herramientas cumplen con los criterios previamente definidos a excepción de Jama Connect, Xebrio y ReqSuite que solo carecen del criterio de generación de documentos. Con respecto a Orcanos hay que señalar que está más

enfocado a la gestión de requisitos de software para el área médica, lo cual lo hace ideal si se pertenece a esa área de trabajo, pero lo vuelve inadecuado y complejo si se pertenece a un área de trabajo diferente. En el caso de Modern Requirements nos encontramos en la situación de que depende de Azure Devops para su correcto funcionamiento, por lo que, si se posee experiencia previa con esta plataforma web, tendríamos un entorno de desarrollo mucho más completo en el ciclo de vida del desarrollo de software, pero si no se ha trabajado previamente con Azure Devops nos enfrentaríamos a una problemática referente a la alta curva de aprendizaje necesaria para poder utilizar la herramienta, debido a esto tendríamos que analizar la situación y optar por la opción que más se ajuste a nuestras necesidades. Además de todo lo explicado anteriormente, también es importante tener en cuenta que todas estas herramientas son de pago, pero disponen de una versión gratuita de prueba.

CAPÍTULO III

3. MARCO METODOLÓGICO

En el presente capítulo se describe de forma detallada el proceso y la metodología de desarrollo usada para la construcción de una herramienta que permita la comunicación, la contribución y la integración en la trazabilidad de requisitos en proyectos de software.

3.1 Diseño de la investigación

En la presente sección se detallan los métodos, técnicas y tipo de investigación que ayuda a resolver la problemática del presente trabajo.

3.1.1 Tipo de investigación

Para el presente trabajo de integración curricular se emplean dos tipos de investigación: aplicada y cuantitativa, a continuación, se describe cada tipo.

- Investigación Aplicada: Puesto que se aplicará todos los conocimientos teóricos y prácticos que nos impartieron en los años de estudio de la carrera de ingeniería en software.
- Investigación Cuantitativa: Puesto que a través de la recopilación de datos y del uso de la herramienta informáticas se medirá el nivel de mejora de la productividad.

3.1.2 Métodos y Técnicas

En la **Tabla 1-3**, se presentan los métodos y técnicas utilizados en el proyecto para lograr cumplir con los objetivos planteados.

Tabla 1-3: Métodos, técnicas y fuentes.

Objetivos	Métodos	Técnicas	Fuentes
Identificar las diferentes técnicas o métodos para representar la trazabilidad de requisitos.	Analítico Exploratorio	Revisión de literatura Observación	Internet Herramientas del mercado

Implementar las funcionalidades de la herramienta para permitir la comunicación, contribución e integración en la trazabilidad de requisitos en proyectos de software.	Scrum	Revisión de documentación Sprint Planning Sprint Retrospective	Internet Documentación de la página oficial
Medir el nivel de mejora de la productividad en el ciclo de vida del desarrollo de software al utilizar la herramienta desarrollada.	Analítico Sintético	Observación IEEE Std 1045-1992	Usuarios Herramienta: Next Requirements

Realizado por: Requelme José, 2022.

A continuación, se presentan los métodos de investigación que se utilizan en el desarrollo del presente trabajo de investigación curricular:

- Método Analítico: Es un procedimiento que descompone un todo en sus elementos básicos y, por tanto, que va de lo general a lo específico, facilitando con esto la comprensión, elaboración y obtención de resultados. En el presente trabajo este método se emplea para identificar las diferentes técnicas o métodos para representar la trazabilidad de requisitos.
- Método Exploratorio: Este método tiene el propósito de destacar los aspectos fundamentales de una problemática determinada y encontrar los procedimientos adecuados para elaborar una investigación posterior. En el presente trabajo se utiliza este método para explorar las diferentes herramientas software en el mercado que tengan un propósito similar.
- Método Sintético: Es un método de razonamiento que tiene como objetivo principal resumir los aspectos más relevantes de un proceso. Este método se lo utiliza en la medición del nivel de mejora de la productividad.

Con el objetivo de recopilar la información necesaria para llevar a cabo el desarrollo de este trabajo de investigación curricular, se determinó utilizar las siguientes técnicas:

- Observación: Es una técnica que permite visualizar el proceso de trazabilidad de requisitos de herramientas con un propósito similar que están actualmente en el mercado.
- Revisión de literatura: Esta técnica implica detectar, consultar y obtener la bibliografía y otros materiales que sean útiles para los propósitos del estudio, de donde se tiene que extraer y recopilar la información relevante y necesaria. En el presente trabajo se lo emplea

para la identificar las diferentes técnicas o métodos para representar la trazabilidad de requisitos.

- Revisión de documentación: Es una técnica que ayuda a tener una idea más clara y concisa sobre un tema en específico, en el presente trabajo se utiliza la documentación oficial de Git, React, Nestjs, PostgreSQL para el desarrollo de la herramienta.

3.2 Recursos que emplear para el desarrollo

Para el desarrollo de la herramienta “Next Requirements” desde su inicio hasta la implementación de todos sus servicios, se utilizó diversos recursos los cuales se detallarán a continuación.

3.2.1 Recursos software

En la **Tabla 2-3** están detallados las diferentes aplicaciones informáticas para realizar el proyecto, cabe recalcar que la mayoría de ellas poseen licencias OpenSource.

Tabla 2-3: Recursos software.

Recurso	Descripción	Observación
Windows 10	Sistema Operativo	Licencia Legal
DataGrip	Potente IDE que facilita el trabajo con bases de datos.	Licencia Legal (Estudiante)
Visual Studio Code	Editor de código optimizado para construir y depurar aplicaciones web modernas	Licencia Libre
Inkscape	Editor de gráficos vectoriales	Licencia Libre
Insomnia	Gestion de APIs	Licencia Libre
NestJS	Framework para el backend	Licencia Libre
React	Librería para el frontend	Licencia Libre

Realizado por: Requelme José, 2022.

3.2.2 Recursos hardware

En la **Tabla 3-3** están listados los recursos hardware para el desarrollo de la herramienta, cabe recalcar que estos recursos hardware son existentes, por lo que no se tuvieron que adquirir.

Tabla 3-3: Recursos hardware.

Recurso	Descripción	Observación
Laptop HP	Procesador Intel I5-7200U, Memoria RAM de 16 GB, SSD de 250 GB.	Disponible
Monitor	Tamaño 19.5 pulgadas, Resolución 1366x768, Entradas HDMI y VGA.	Disponible
Impresora EPSON	Modelo: C11CJ63301, Resolución Máxima de Impresión: Hasta 5.760 dpi x 1.440 dpi de resolución optimizada en varios tipos de papel.	Disponible

Realizado por: Requelme José, 2022.

3.3 Desarrollo de la herramienta utilizando la metodología SCRUM

De acuerdo con la lista de requerimientos obtenidos en la sección anterior, se analizó cada uno de ellos y se dividieron de acuerdo con la prioridad que se les asigno, de esta forma se priorizo el desarrollo de las funcionalidades más importantes en los primeros Sprints, asignando a cada sprint una duración de dos semanas.

3.3.1 Fase de planificación

En esta fase se analiza la problemática del proyecto con el propósito de definir los tipos de usuarios y roles involucrados en el sistema, así como los requerimientos por implementarse.

3.3.1.1 Personas y roles involucrado en el proyecto

En la **Tabla 4-3**, se encuentra detallada la información de las personas involucradas en el desarrollo del proyecto con sus respectivos roles.

Tabla 4-3: Personas y roles involucrados en el proyecto.

Persona	Rol	Contacto
Ing. Raúl Rosero	Product Owner	raul.rosero@epoch.edu.ec
Ing. Danilo Pastor	Scrum Master	danilo.pastor@epoch.edu.ec
José Requelme	Development Team	jose.requelme@epoch.edu.ec

Realizado por: Requelme José, 2022.

3.3.1.2 Tipos de usuarios y roles involucrados en el sistema

Para la herramienta desarrollada “Next Requirements”, se detalla tres tipos de usuario con sus respectivos roles, los cuales podremos visualizarlos a continuación en la **Tabla 5-3**.

Tabla 5-3: Tipos de usuarios y roles involucrados en el sistema.

Tipo de Usuario	Rol
Owner	Puede crear nuevos proyectos y tiene todos los permisos (Lectura, Escritura, Visualización), más permisos administrativos.
Editor	Posee permisos de lectura, escritura y visualización.
Reviewer	Posee los mismos permisos que un tipo de usuario Editor más los permisos necesarios para supervisar a un Editor.

Realizado por: Requelme José, 2022.

3.3.1.3 Product backlog

Para el product backlog, se asigna los puntos de estimación, así como la prioridad a cada uno de los requisitos de software documentados como historias de usuario.

Para la asignación de puntos de estimación se utiliza la técnica T-shirt, ya que, nos permite clasificar las historias de usuario entre las categorías que veremos a continuación, de las cuales se ha asignado puntos de estimación para cada categoría.

- XS con una estimación de 4 puntos.
- S con una estimación de 8 puntos.
- M con una estimación de 16 puntos.
- L con una estimación de 24 puntos.
- XL con una estimación de 32 puntos.

Cabe señalar que cada punto estimado hace referencia a una hora de trabajo, por lo que 4 puntos equivaldrían a 4 horas de trabajo, 8 puntos a 8 horas de trabajo, y así sucesivamente.

A continuación, en la **Tabla 6-3**, se listan las historias de usuario e historias técnicas utilizadas en el desarrollo de la herramienta.

Tabla 6-3: Product Backlog.

ID	Descripción	Puntos Estimados	Prioridad
HT-01	Definir el estándar de codificación	8	Alta

HT-02	Definir la arquitectura de la aplicación	8	Alta
HT-03	Diseño de la base de datos	24	Alta
HT-04	Diseñar las interfaces de usuario	8	Alta
HT-05	Generar diccionario de datos	8	Media
HT-06	Documentación	24	Media
HU-01	Crear una nueva cuenta de usuario en la aplicación web	8	Media
HU-02	Recuperar contraseña mediante su correo electrónico	16	Baja
HU-03	Iniciar sesión en la aplicación web	8	Media
HU-04	Actualizar la información personal del usuario	4	Baja
HU-05	Actualizar la contraseña del usuario	4	Baja
HU-06	Crear un nuevo proyecto	16	Alta
HU-07	Actualizar la información de un proyecto	4	Baja
HU-08	Eliminar un proyecto	8	Baja
HU-09	Añadir un nuevo miembro al proyecto	8	Media
HU-10	Actualizar el rol de los miembros del proyecto	4	Baja
HU-11	Eliminar un miembro del proyecto	4	Baja
HU-12	Añadir un nuevo requerimiento al proyecto	8	Alta
HU-13	Asignar un editor y un reviewer al requerimiento	4	Media
HU-14	Vincular con otro requerimiento de ser necesario	4	Alta
HU-15	Actualizar la información del requerimiento	8	Alta
HU-16	Visualizar las versiones de cambios del requerimiento	8	Alta
HU-17	Retroceder a la versión del requerimiento deseado	4	Media
HU-18	Visualizar la trazabilidad de los requerimientos	8	Alta
HU-19	Generar un PDF con los requerimientos del proyecto	4	Baja
HU-20	Generar un PDF de las versiones del requerimiento	4	Baja
HU-21	Añadir un comentario a un requerimiento	4	Baja

HU-22	Actualizar el contenido del comentario	4	Baja
HU-23	Eliminar el comentario	4	Baja
HU-24	Notificar al editor y reviewer cuando un requerimiento cambie	8	Media
HU-25	Notificar a un usuario cuando se le añada a un proyecto	4	Media

Realizado por: Requelme José, 2022.

3.3.2 Fase de diseño

En esta fase se realizaron actividades de modelado, diagramas y definición de los estándares de codificación y de base de datos con el fin de tener una idea más específica de lo que se va a desarrollar.

3.3.2.1 Arquitectura del sistema

En la **Figura 1-3**, se presenta el diagrama de despliegue que tiene la herramienta “Next Requirements”, el cual nos sirve para tener una idea general de como esta desarrollada la arquitectura del sistema.

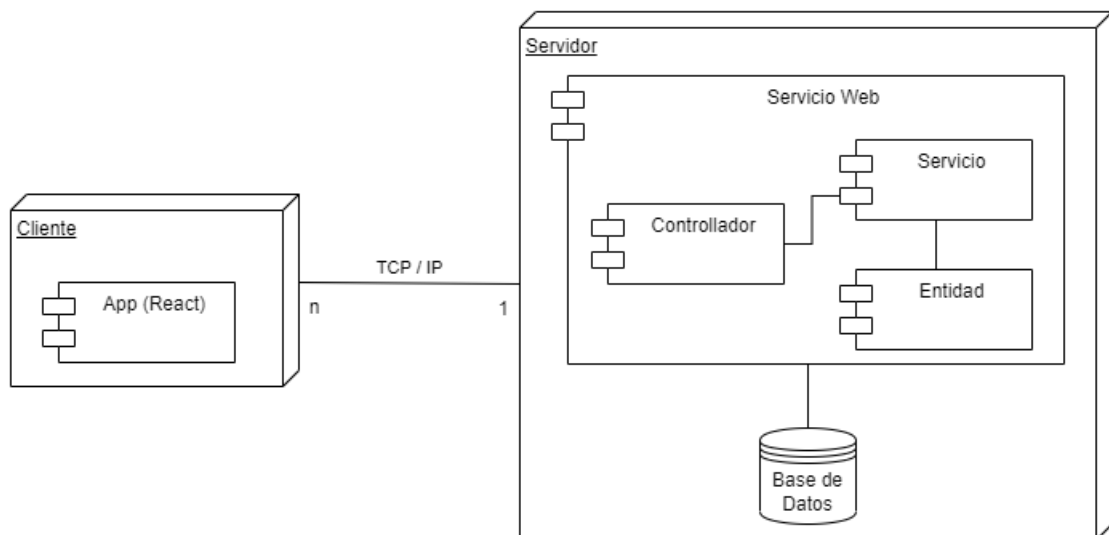


Figura 1-3. Diagrama de despliegue.

Realizado por: Requelme José, 2022.

3.3.2.2 Diagramas de secuencia UML

El diagrama de secuencia modela la interacción de los objetos de un sistema. En la **Figura 2-3** se muestra la secuencia y la interacción al momento de añadir un nuevo miembro al proyecto.

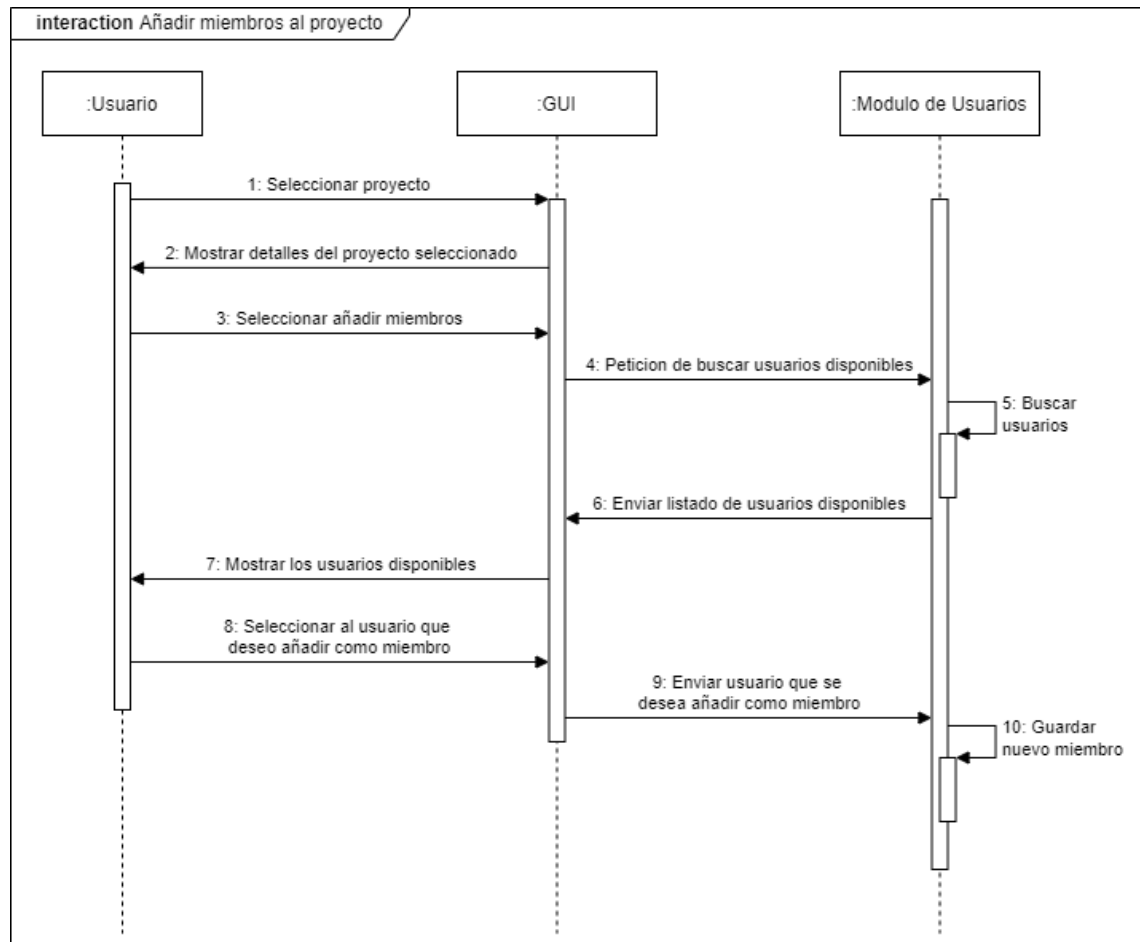


Figura 2-3. Diagrama de secuencia.

Realizado por: Requelme José, 2022.

El resto de los diagramas de secuencia UML se encuentran disponibles en el Anexo A.

3.3.2.3 Estándar de Codificación

El estándar de codificación nos permite una mejor lectura del código y gracias a esto, también nos otorga una facilidad en su mantenimiento, en la **Tabla 7-3** que se muestra a continuación se detalla el estándar de codificación utilizado para el desarrollo del proyecto.

Tabla 7-3: Estándar de codificación.

Componente	Ejemplo	Descripción
Nombre de archivos	projects.controller.ts projects.service.ts projects.module.ts projects.entity.ts	Nombre del archivo en KebabCase, seguido de punto y su tipo (controller, service, module, etc), más su extensión
Variables	titles: string[] projects: Project[]	Nombre de la variable más el tipo de variable.
Clases	ProjectController UserService UserModule	Nombre de la clase seguido de su tipo (controller, service, module, etc)
Métodos	getProjectById() getAllUsers()	Nombre de la acción que se realiza seguida del tipo de información que se procesa

Realizado por: Requelme José, 2022.

3.3.2.4 Diseño de la base de datos

En la **Figura 3-3**, se muestra el diseño físico de la base de datos, el cual contiene sus tablas y relaciones, además de estar incorporado sus atributos, claves foráneas y claves primarias.



Figura 3-3. Diseño físico de la base de datos.

Realizado por: Requelme José, 2022.

3.3.2.5 Diccionario de datos

El objetivo de un diccionario de datos es dar precisión sobre los datos que se manejan en un sistema, evitando así malas interpretaciones o ambigüedades.

En la **Tabla 8-3** está detallada la información de la tabla Users, las demás tablas se encuentran disponibles en el Anexo B.

Tabla 8-3: Diccionario de datos de la tabla Users.

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id	uuid	uuid_generate_v4()	NO	SI
username	varchar(15)		NO	NO
firstName	varchar(20)		NO	NO
lastName	varchar(20)		NO	NO
email	varchar(150)		NO	NO
password	text		NO	NO
avatar	text	'https://cdn-icons-png.flaticon.com/512/149/149071.png'	NO	NO
createdAt	timestamp with time zone	now()	NO	NO
updatedAt	timestamp with time zone	now()	NO	NO

Realizado por: Requelme José, 2022.

3.3.2.6 Diseño de interfaces de usuario

En la **Figura 4-3**, se encuentra maquetado la pantalla de inicio de sesión de la herramienta, la cual consta de un formulario con los campos de usuario y contraseña.

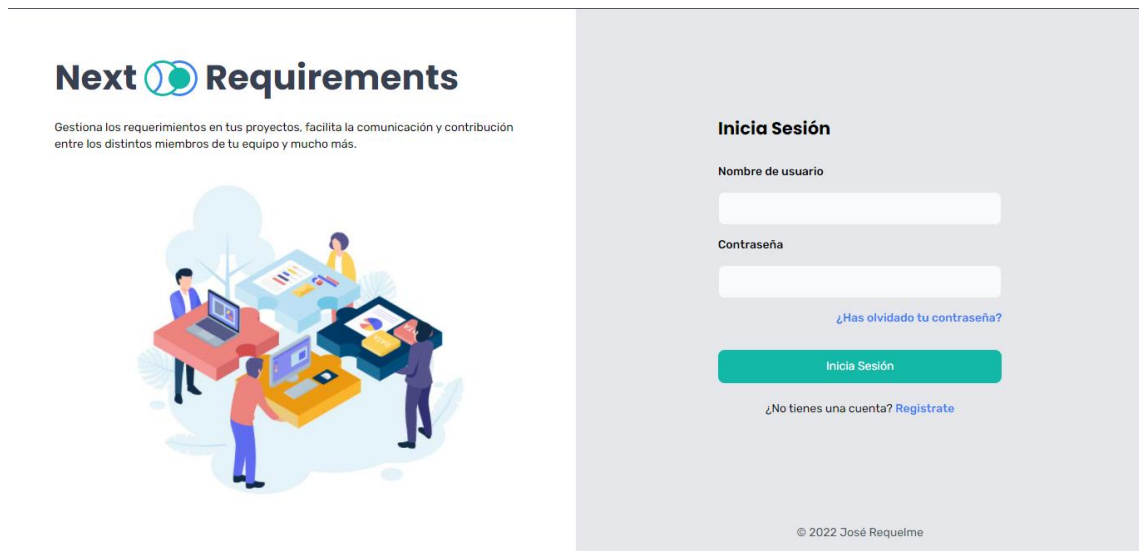


Figura 4-3. Pantalla de inicio de sesión.

Realizado por: Requelme José, 2022.

En la **Figura 5-3**, se encuentra maquetado la pantalla de inicio de la herramienta, realizado con el software de Figma.

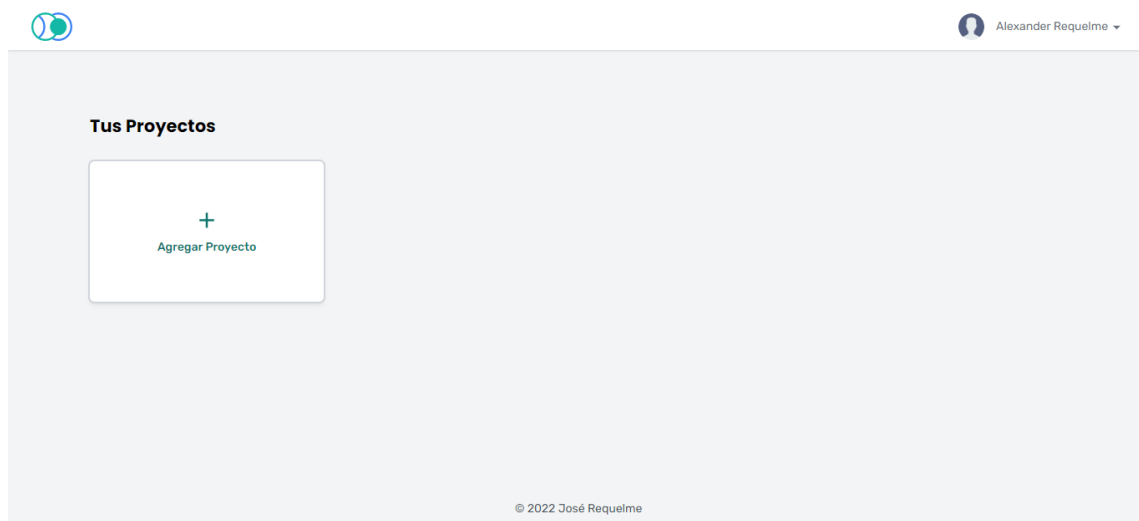


Figura 5-3. Pantalla principal de la herramienta.

Realizado por: Requelme José, 2022.

3.3.3 *Fase de desarrollo*

En esta fase se presenta la documentación de las actividades realizadas durante el desarrollo de la herramienta, tales como el sprint backlog y las historias de usuario.

3.3.3.1 *Sprint Backlog*

En la **Tabla 9-3**, se muestra detallada las historias de usuario/técnicas realizadas en cada sprint, cabe añadir que el total de sprints que se obtuvieron fueron 6, los cuales constan de 40 puntos estimados por sprint, dado que la duración de sprint es de 2 semanas, en cada semana se trabaja un total de 20 horas y cada día equivaldría a 4 horas de trabajo.

Tabla 9-3: Sprint Backlog.

ID	Historias de usuario/técnicas	Fecha de Inicio	Fecha de Fin	Puntos Estimados
Sprint 1		15/11/2021	26/11/2021	40
HT-01	Definir el estándar de codificación	15/11/2021	16/11/2021	8
HT-02	Definir la arquitectura de la aplicación	17/11/2021	18/11/2021	8
HT-03	Diseño de la base de datos	19/11/2021	26/11/2021	24
Sprint 2		29/11/2021	10/12/2021	40
HT-04	Diseñar las interfaces de usuario	29/11/2021	30/11/2021	8
HT-05	Generar diccionario de datos	01/12/2021	02/12/2021	8
HT-06	Documentación	03/12/2021	10/12/2021	24
Sprint 3		13/12/2021	24/12/2021	40
HU-01	Crear una nueva cuenta de usuario en la aplicación web	13/12/2021	14/12/2021	8
HU-02	Recuperar contraseña mediante su correo electrónico	15/12/2021	20/12/2021	16
HU-03	Iniciar sesión en la aplicación web	21/12/2021	22/12/2021	8
HU-04	Actualizar la información personal del usuario	23/12/2021	23/12/2021	4
HU-05	Actualizar la contraseña del usuario	24/12/2021	24/12/2021	4
Sprint 4		03/01/2022	14/01/2022	40
HU-06	Crear un nuevo proyecto	03/01/2022	06/01/2022	16
HU-07	Actualizar la información de un proyecto	07/01/2022	07/01/2022	4
HU-08	Eliminar un proyecto	10/01/2022	11/01/2022	8
HU-09	Añadir un nuevo miembro al proyecto	12/01/2022	13/01/2022	8
HU-10	Actualizar el rol de los miembros del proyecto	14/01/2022	14/01/2022	4
Sprint 5		17/01/2022	28/01/2022	40
HU-11	Eliminar un miembro del proyecto	17/01/2022	17/01/2022	4

HU-12	Añadir un nuevo requerimiento al proyecto	18/01/2022	19/01/2022	8
HU-13	Asignar un editor y un reviewer al requerimiento	20/01/2022	20/01/2022	4
HU-14	Vincular con otro requerimiento de ser necesario	21/01/2022	21/01/2022	4
HU-15	Actualizar la información del requerimiento	24/01/2022	25/01/2022	8
HU-16	Visualizar las versiones de cambios del requerimiento	26/01/2022	27/01/2022	8
HU-17	Retroceder a la versión del requerimiento deseado	28/01/2022	28/01/2022	4
Sprint 6		31/01/2022	11/02/2022	40
HU-18	Visualizar la trazabilidad de los requerimientos	31/01/2022	01/02/2022	8
HU-19	Generar un PDF con los requerimientos del proyecto	02/02/2022	02/02/2022	4
HU-20	Generar un PDF de las versiones del requerimiento	03/02/2022	03/02/2022	4
HU-21	Añadir un comentario a un requerimiento	04/02/2022	04/02/2022	4
HU-22	Actualizar el contenido del comentario	07/02/2022	07/02/2022	4
HU-23	Eliminar el comentario	08/02/2022	08/02/2022	4
HU-24	Notificar al editor y reviewer cuando un requerimiento cambie	09/02/2022	10/02/2022	8
HU-25	Notificar a un usuario cuando se le añada a un proyecto	11/02/2022	11/02/2022	4

Realizado por: Requelme José, 2022.

3.3.3.2 Historia de usuario

La representación de las funcionalidades se realiza por medio de las Historias de Usuario (HU), las cuales tienen una regla para documentarlas: Como (rol), quiero (evento), para (funcionalidad). También se documenta los criterios de aceptación los cuales nos sirven para determinar si una Historia de Usuario se ha completado íntegramente o no.

En la **Tabla 10-3**, se muestra documentada la Historia de Usuario para crear una nueva cuenta de usuario en la aplicación web.

Tabla 10-3: Historia de Usuario 01.

Historia de Usuario			
ID: HU-01	Nombre de la historia: Crear una nueva cuenta de usuario en la aplicación web		
Usuario: Usuario		Sprint: 3	
Prioridad: Media		Puntos Estimados: 8	
(Alta/Media/Baja)		Puntos Reales: 8	
Fecha Inicio: 13/12/2021		Fecha Fin: 14/12/2021	
Descripción: Como usuario quiero crearme una cuenta para poder utilizar los servicios de la aplicación web			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-01 PA-01	Verificar que el nombre de usuario sea único	Exitoso	José Requelme
HU-01 PA-02	Verificar que el correo electrónico sea único	Exitoso	José Requelme
HU-01 PA-03	No permitir el ingreso de campos vacíos	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-01 TI-01	Desarrollar la interfaz de usuario para el registro de una nueva cuenta y conectar con el backend	4	
HU-01 TI-02	Desarrollar el servicio web RESTful para la creación de una nueva cuenta	4	

Realizado por: Requelme José, 2022.

El resto de las historias de usuario con sus respectivas pruebas de aceptación y tareas de ingeniería se encuentran documentadas en el Anexo C.

CAPÍTULO IV

4. RESULTADOS

En este capítulo se realiza el análisis de los resultados obtenidos al haber finalizado la construcción de una herramienta que permita la comunicación, la contribución y la integración en la trazabilidad de requisitos en proyectos de software, a la cual se le ha dado el nombre de “Next Requirements”.

Para medir el nivel de mejora de la productividad en el ciclo de vida del desarrollo de software al utilizar la herramienta desarrollada, se realiza un análisis comparativo entre la productividad que se obtiene de un equipo de desarrollo en documentar la elicitación de requerimientos con procesos tradicionales y la productividad de un segundo equipo de desarrollo al documentarlo con la herramienta desarrollada.

4.1 Población

Para medir el nivel de mejora de la productividad de la herramienta se toma como población los 25 requerimientos funcionales los mismos que fueron documentados como historias de usuario, las cuales están detalladas en la **Tabla 6-3**.

4.2 Muestra

Para la muestra se tomó en cuenta 10 requerimientos funcionales documentados como historias de usuario en la **Tabla 6-3**, estos requerimientos cumplen con el criterio de tener la mayor cantidad de puntos estimados definidos con la técnica T-shirt. A continuación, en la **Tabla 1-4** se presentan estos requerimientos asociados con su respectivo identificador de historia de usuario.

Tabla 1-4: Lista de historias de usuario para el caso de estudio.

ID	Historia de Usuario
HU-01	Crear una nueva cuenta de usuario en la aplicación web
HU-02	Recuperar contraseña mediante su correo electrónico
HU-03	Iniciar sesión en la aplicación web
HU-06	Crear un nuevo proyecto
HU-08	Eliminar un proyecto
HU-09	Añadir un nuevo miembro al proyecto

HU-12	Añadir un nuevo requerimiento al proyecto
HU-15	Actualizar la información del requerimiento
HU-16	Visualizar las versiones de cambios del requerimiento
HU-18	Visualizar la trazabilidad de los requerimientos

Realizado por: Requelme José, 2022.

4.3 Criterio de Evaluación

Para medir el nivel de mejora de la productividad en el ciclo de vida del desarrollo de software al utilizar la herramienta “Next Requirements”, se establecieron los siguientes indicadores en base al estándar IEEE Std 1045–1992, los cuales se visualizan en la **Tabla 2-4**.

Tabla 2-4: Criterio de evaluación.

Variable	Indicador	Propósito	Estrategia
Productividad	Output: Requerimientos documentados	Medir el nivel de mejora de la productividad	Comparar la productividad entre dos equipos de desarrollo, en el cual el primer equipo utiliza un proceso tradicional, mientras que el segundo equipo utiliza la herramienta desarrollada.
	Input: Esfuerzo realizado		

Realizado por: Requelme José, 2022.

4.4 Productividad

Para obtener el nivel de mejora de la productividad debemos dividir la cantidad de requerimientos entregados con el esfuerzo realizado por equipo de desarrollo. Para lo cual se forman dos equipos de desarrollo, a los cuales los designaremos como Equipo A y Equipo B, siendo el Equipo A quienes realizarán la documentación de la elicitación de requerimientos con el proceso tradicional y el Equipo B quienes lo harán mediante la herramienta desarrollada, cabe recalcar que cada equipo está conformado por 3 desarrolladores de software. Para obtener el esfuerzo realizado por cada equipo, se recolectó el tiempo trabajado de cada uno mediante la técnica de observación junto con la utilización de un cronómetro.

4.4.1 *Análisis del proceso tradicional*

Para este proceso se tomó los tiempos que los desarrolladores del Equipo A tardaron en completar las actividades de documentación, revisión / retroalimentación y trazabilidad por cada historia de usuario, utilizando para ello la ayuda de un cronómetro. Para realizar estas actividades se utilizó una plantilla en Word basado en el estándar IEEE 830-1998 que soporta la trazabilidad de requisitos, la cual se adjunta en el ANEXO D. A continuación, en la **Tabla 3-4** se puede observar los tiempos obtenidos en horas por cada actividad.

Tabla 3-4: Tiempo trabajado por actividad del Equipo A.

ID	Actividad	Tiempo por Actividad	Tiempo Total
HU-01	Documentación de la funcionalidad	0.70	1.25
	Revisión y retroalimentación	0.20	
	Determinar su trazabilidad	0.35	
HU-02	Documentación de la funcionalidad	0.73	1.25
	Revisión y retroalimentación	0.15	
	Determinar su trazabilidad	0.37	
HU-03	Documentación de la funcionalidad	0.85	1.50
	Revisión y retroalimentación	0.25	
	Determinar su trazabilidad	0.40	
HU-06	Documentación de la funcionalidad	0.85	1.50
	Revisión y retroalimentación	0.25	
	Determinar su trazabilidad	0.40	
HU-08	Documentación de la funcionalidad	2.36	3.00
	Revisión y retroalimentación	0.25	
	Determinar su trazabilidad	0.39	
HU-09	Documentación de la funcionalidad	0.70	1.25
	Revisión y retroalimentación	0.15	
	Determinar su trazabilidad	0.40	
HU-12	Documentación de la funcionalidad	0.75	1.25
	Revisión y retroalimentación	0.15	
	Determinar su trazabilidad	0.35	
HU-15	Documentación de la funcionalidad	0.95	1.50
	Revisión y retroalimentación	0.20	

	Determinar su trazabilidad	0.35	
HU-16	Documentación de la funcionalidad	2.35	3.00
	Revisión y retroalimentación	0.25	
	Determinar su trazabilidad	0.40	
HU-18	Documentación de la funcionalidad	2.35	3.00
	Revisión y retroalimentación	0.25	
	Determinar su trazabilidad	0.40	

Realizado por: Requelme José, 2022.

A continuación, en la **Tabla 4-4** se detalla el tiempo total trabajado de las actividades realizadas por el Equipo A por cada historia de usuario.

Tabla 4-4: Tiempo total trabajado por el Equipo A.

ID	Historia de Usuario	Tiempo Trabajado (horas)
HU-01	Crear una nueva cuenta de usuario en la aplicación web	1.25
HU-02	Recuperar contraseña mediante su correo electrónico	1.25
HU-03	Iniciar sesión en la aplicación web	1.50
HU-06	Crear un nuevo proyecto	1.50
HU-08	Eliminar un proyecto	3.00
HU-09	Añadir un nuevo miembro al proyecto	1.25
HU-12	Añadir un nuevo requerimiento al proyecto	1.25
HU-15	Actualizar la información del requerimiento	1.50
HU-16	Visualizar las versiones de cambios del requerimiento	3.00
HU-18	Visualizar la trazabilidad de los requerimientos	3.00

Realizado por: Requelme José, 2022.

4.4.2 *Análisis del proceso con la herramienta desarrollada*

Para este proceso se tomó los tiempos que los desarrolladores del Equipo B tardaron en completar cada actividad por historia de usuario, utilizando para ello la ayuda de un cronómetro. Cabe recalcar que para este proceso se emplea la herramienta desarrollada en el presente trabajo la cual se denominó “Next Requirements”, en la **Tabla 5-4** se detallan los tiempos obtenidos en horas por cada actividad.

Tabla 5-4: Tiempo trabajado por actividad del Equipo B.

ID	Actividad	Tiempo por Actividad	Tiempo Total
HU-01	Documentación de la funcionalidad	0.44	0.75
	Revisión y retroalimentación	0.15	
	Determinar su trazabilidad	0.16	
HU-02	Documentación de la funcionalidad	0.44	0.75
	Revisión y retroalimentación	0.15	
	Determinar su trazabilidad	0.16	
HU-03	Documentación de la funcionalidad	0.57	1.00
	Revisión y retroalimentación	0.18	
	Determinar su trazabilidad	0.25	
HU-06	Documentación de la funcionalidad	0.57	1.00
	Revisión y retroalimentación	0.18	
	Determinar su trazabilidad	0.25	
HU-08	Documentación de la funcionalidad	2.07	2.50
	Revisión y retroalimentación	0.18	
	Determinar su trazabilidad	0.25	
HU-09	Documentación de la funcionalidad	0.34	0.75
	Revisión y retroalimentación	0.16	
	Determinar su trazabilidad	0.25	
HU-12	Documentación de la funcionalidad	0.41	0.75
	Revisión y retroalimentación	0.16	
	Determinar su trazabilidad	0.18	
HU-15	Documentación de la funcionalidad	0.64	1.00
	Revisión y retroalimentación	0.18	
	Determinar su trazabilidad	0.18	
HU-16	Documentación de la funcionalidad	2.07	2.50
	Revisión y retroalimentación	0.18	
	Determinar su trazabilidad	0.25	
HU-18	Documentación de la funcionalidad	2.32	2.75
	Revisión y retroalimentación	0.18	
	Determinar su trazabilidad	0.25	

Realizado por: Requelme José, 2022.

A continuación, en la **Tabla 6-4** se detalla el tiempo total trabajado de las actividades realizadas por el Equipo B por cada historia de usuario.

Tabla 6-4: Tiempo total trabajado por el Equipo B.

ID	Historia de Usuario	Tiempo Trabajado (horas)
HU-01	Crear una nueva cuenta de usuario en la aplicación web	0.75
HU-02	Recuperar contraseña mediante su correo electrónico	0.75
HU-03	Iniciar sesión en la aplicación web	1.00
HU-06	Crear un nuevo proyecto	1.00
HU-08	Eliminar un proyecto	2.50
HU-09	Añadir un nuevo miembro al proyecto	0.75
HU-12	Añadir un nuevo requerimiento al proyecto	0.75
HU-15	Actualizar la información del requerimiento	1.00
HU-16	Visualizar las versiones de cambios del requerimiento	2.50
HU-18	Visualizar la trazabilidad de los requerimientos	2.75

Realizado por: Requelme José, 2022.

4.4.3 Comparación de resultados

A continuación, en la **Tabla 7-4** se presenta una comparación del tiempo trabajado tanto del Equipo A como del Equipo B, cabe recalcar que el tiempo trabajado que se obtuvo de ambos equipos está en horas.

Tabla 7-4: Comparación de resultados.

ID	Tiempo Trabajado Equipo A	Tiempo Trabajado Equipo B
HU-01	1.25	0.75
HU-02	1.25	0.75
HU-03	1.50	1.00
HU-06	1.50	1.00
HU-08	3.00	2.50
HU-09	1.25	0.75
HU-12	1.25	0.75
HU-15	1.50	1.00
HU-16	3.00	2.50
HU-18	3.00	2.75

Total	18.50	13.75
--------------	--------------	--------------

Realizado por: Requelme José, 2022.

Obteniendo como resultado para el Equipo A un total de tiempo trabajado de 18.50 horas y para el Equipo B un total de 13.75 horas. En el **Gráfico 1-4**, se compara los valores del tiempo total trabajado entre los equipos por cada historia de usuario.

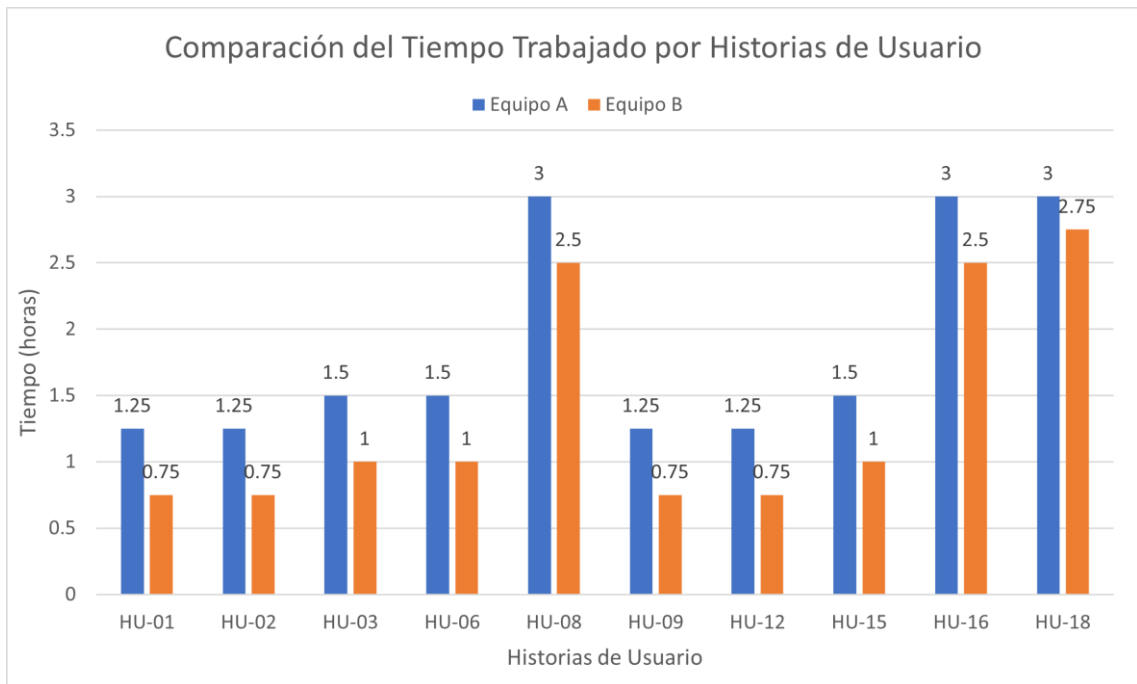


Gráfico 1-4. Comparación del tiempo trabajado por historias de usuario.

Realizado por: Requelme José, 2022.

Para obtener la productividad tanto del Equipo A como del Equipo B debemos aplicar la siguiente fórmula, según nos indica la (IEEE Std 1045–1992 1992), la productividad del producto a , que tiene la cantidad de primitiva de salida O_a y cuya creación ha requerido un esfuerzo E_a , se define por la siguiente relación:

$$Productivity_a = \frac{O_a}{E_a}$$

En la **Tabla 8-4** se puede visualizar la productividad tanto del Equipo A como la del Equipo B, recordando que el número total de requerimientos documentados son 10, el cual es el mismo para ambos equipos.

Tabla 8-4: Productividad de los equipos de desarrollo.

Equipo	Requerimientos Documentados	N° Desarrolladores	Tiempo Trabajado	Esfuerzo Realizado	Productividad
Equipo A	10	3	18.50	55.50	0.1802
Equipo B	10	3	13.75	41.25	0.2424

Realizado por: Requelme José, 2022.

Obteniendo como resultado que por cada hora-personal del Equipo A se obtiene aproximadamente 0.1802 requerimientos documentados y por cada hora-personal del Equipo B se obtiene aproximadamente 0.2424 requerimientos documentados, además cabe señalar que cuanto más alto sea el valor de la productividad se lo considera mejor, como se lo puede observar en el **Gráfico 2-4**.

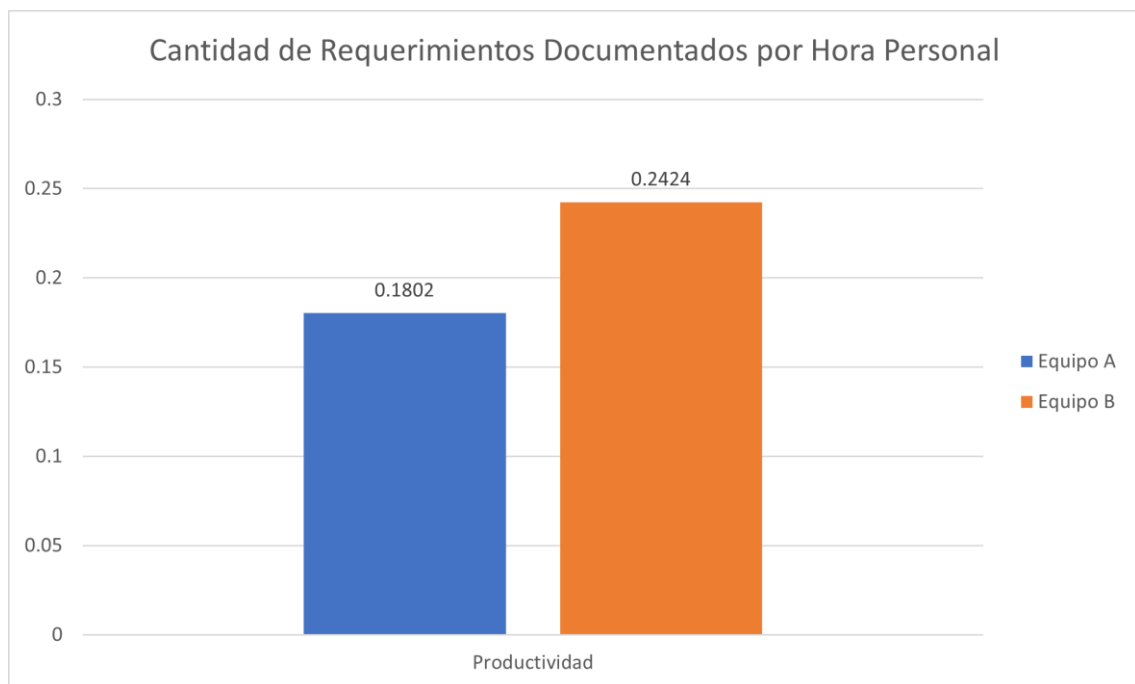


Gráfico 2-4. Comparación de la productividad.

Realizado por: Requelme José, 2022.

4.5 Análisis de los resultados obtenidos

En esta sección se analiza los resultados obtenidos de la productividad tanto del Equipo A como del Equipo B. Para medir el nivel de mejora de la productividad al utilizar la herramienta desarrollada, utilizaremos la siguiente fórmula de incremento de productividad, donde:

Productividad Inicial = Productividad del Equipo A

Productividad Final = Productividad del Equipo B

$$IP = \frac{(Productividad\ Final - Productividad\ Inicial)}{Productividad\ Inicial} \times 100$$

Al reemplazar los valores en la formula nos da el siguiente resultado

$$IP = \frac{(0.2424 - 0.1802)}{0.1802} \times 100 = 34.52 \%$$

Logrando con esto un nivel de mejora de la productividad del 34.52% al utilizar la herramienta desarrollada en comparación con el proceso tradicional en la elicitación de requerimientos, tal como se visualiza en la **Tabla 9-4**.

Tabla 9-4: Nivel de mejora de la productividad.

	Productividad Equipo A	Productividad Equipo B	% Mejora de la Productividad
Elicitación de requerimientos	0.1802	0.2424	34.52

Realizado por: Requelme José, 2022.

CONCLUSIONES

- Los requisitos de la herramienta desarrollada se definieron a través del análisis de herramientas con funcionalidades similares que están actualmente en el mercado, se determinó que dichas herramientas contaban con funcionalidades para la: gestión de requerimientos, trazabilidad, generación de reportes, trabajo colaborativo, historia y versionamiento de requisitos; y en base a esto abstraer las funcionalidades más importantes para posteriormente volverlas requisitos que la herramienta debía cumplir.
- La utilización de una herramienta que permita la trazabilidad de requisitos en proyectos de software facilita la mantenibilidad de esta en comparación con un proceso manual, debido a que utiliza las facilidades de exploración que proporcionan las bases de datos para mantener una consistencia en la vinculación de requisitos y poder generar así matrices de trazabilidad de manera automática, lo cual es de gran ayuda sobre todo en proyectos de software a grande escala donde la cantidad de requisitos es significativamente alta.
- Para la construcción de la aplicación se utilizó la metodología de desarrollo ágil Scrum, además se identificó un total de 6 historias técnicas y 25 historias de usuario las mismas que se desarrollaron en el transcurso del proyecto a lo largo de 6 sprints, los cuales contaban con una duración de 2 semanas y con un total de 40 puntos de esfuerzo por sprint.
- Para el desarrollo de la herramienta “Next Requirements” se empleó la librería React para el frontend ya que nos permite crear interfaces de usuario reactivas de manera sencilla y para el backend se empleó el framework NestJs ya que permite crear aplicaciones escalables y aplica un conjunto de buenas prácticas por defecto las cuales nos aporta un gran valor en el desarrollo.
- Para medir el nivel de mejora de la productividad al utilizar la herramienta desarrollada, se contó con 2 equipos de desarrollo conformado por 3 desarrolladores por equipo, en el cual un equipo utilizaría la herramienta para realizar la elicitación de requerimientos y el otro lo haría de manera tradicional. Una vez analizado los resultados y después de realizar los cálculos respectivos, se obtuvo un nivel de mejora de la productividad del 34.52% al utilizar la herramienta.

RECOMENDACIONES

- A pesar de que la herramienta desarrollada permite la contribución y comunicación entre los miembros del proyecto, se podría aumentar un sistema de mensajería instantánea como por ejemplo un chat en tiempo real, con el fin de tener una comunicación más directa e inmediata para la solución de inquietudes entre los miembros del proyecto.
- El diseño web de la herramienta desarrollada se enfocó en que sea intuitivo y de baja carga cognitiva. Una mejora en este diseño que podría aumentar la experiencia de usuario sería que el diseño sea responsive, es decir que se adapte a las diferentes resoluciones de las pantallas de los dispositivos, ya que proporcionaría mucho valor a los usuarios.
- En el presente trabajo de integración curricular se utilizó la metodología SCRUM para el proceso de desarrollo, logrando con ello una mejor organización del trabajo, aun así, sería interesante que para la implementación de futuras funcionalidades se lo realice con otra metodología ágil como lo es XP, ya que nos ofrece un conjunto de valores, principios y prácticas que mejoran el desarrollo de un equipo de trabajo.
- Se recomienda trabajar con la tecnología de contenedores, ya que facilita el despliegue de la aplicación, debido a que dentro un contenedor se encuentra todos los archivos y configuraciones necesarias para que la aplicación se ejecute, debido a esto permite mantener un entorno consistente a través de diferentes ambientes de trabajo lo cual es ideal para un equipo de desarrollo.

GLOSARIO

Sprint: Ciclos de ejecución muy cortos entre una y cuatro semanas, cuyo objetivo es conseguir un incremento de valor en el producto que está en proceso de construcción.

Trazabilidad: Serie de procedimientos que permiten seguir el proceso de evolución de un producto en cada una de sus etapas.

IEEE: Es la sociedad técnico-profesional más grande y prestigiosa del mundo, dedicada a promover y divulgar los avances científicos en las áreas de Ingeniería Eléctrica, Electrónica, Energética, Informática y afines.

SWEBOK: Cuerpo de conocimiento de la Ingeniería del Software.

RTM: Matriz de trazabilidad de los requisitos.

API: Es un conjunto de definiciones y protocolos que se usa para diseñar e integrar el software de las aplicaciones.

BIBLIOGRAFÍA

- BOKHARI, M. y SIDDIQUI, S.**, *Metrics for Requirements Engineering and Automated Requirements Tools*. New Delhi: Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM). 2011. ISBN 978-93-80544-00-7.
- CHACON, S.**, Git. [en línea]. 2021. [Consulta: 14 diciembre 2021]. Disponible en: <https://git-scm.com/>.
- FACEBOOK INC**, React – A JavaScript library for building user interfaces. [en línea]. 2021. [Consulta: 16 noviembre 2021]. Disponible en: <https://reactjs.org/>.
- GOTEL, O.C.Z.**, Contribution Structures for Requirements Traceability. , pp. 354. 1995.
- GOTEL, O.C.Z. y FINKELSTEIN, C.W.**, An analysis of the requirements traceability problem. *Proceedings of IEEE International Conference on Requirements Engineering* [en línea]. 1994. Colorado Springs, CO, USA: IEEE Comput. Soc. Press, pp. 94-101. [Consulta: 22 noviembre 2021]. ISBN 978-0-8186-5480-0. DOI 10.1109/ICRE.1994.292398. Disponible en: <http://ieeexplore.ieee.org/document/292398/>.
- HOFFMANN, M., KUHN, N., BITTNER, M. y WEBER, M.**, Requirements for requirements management tools. *Proceedings. 12th IEEE International Requirements Engineering Conference, 2004*. [en línea]. 2004. Kyoto, Japan: IEEE, pp. 282-289. [Consulta: 23 abril 2022]. ISBN 978-0-7695-2174-9. DOI 10.1109/ICRE.2004.1335687. Disponible en: <http://ieeexplore.ieee.org/document/1335687/>.
- IEEE COMPUTER SOCIETY, BOURQUE, P. y FAIRLEY, R.E.**, *Guide to the software engineering body of knowledge*. S.l.: s.n. 2014. ISBN 978-0-7695-5166-1.
- IEEE SA**, IEEE Guide for Developing System Requirements Specifications. [en línea]. S.l.: IEEE. 1998. [Consulta: 30 marzo 2022]. Disponible en: <http://ieeexplore.ieee.org/document/741940/>.
- IEEE STD 1045–1992**, IEEE Std 1045-1992, IEEE Standard for Software Productivity Metrics. , pp. 38. 1992.

JAMA SOFTWARE, Requirements Management Software | Jama Connect™ Features. *Jama Software* [en línea]. 2021. [Consulta: 14 diciembre 2021]. Disponible en: <https://www.jamasoftware.com/platform/jama-connect/features/>.

MISSELWITZ, M., Optimierung der Rückverfolgbarkeit von Anforderungen im Software-Entwicklungsprozess- Entwicklung eines Lösungsansatzes zum Verknüpfen von Anforderungen mit Komponenten in der Benutzerschnittstelle. , pp. 93. 2012.

MODERN REQUIREMENTS, Requirements Management Tools. *Modern Requirements* [en línea]. 2021. [Consulta: 14 diciembre 2021]. Disponible en: <https://www.modernrequirements.com/products/modern-requirements4devops/>.

MYSLIWIEC, K., Documentation | NestJS - A progressive Node.js framework. *Documentation / NestJS - A progressive Node.js framework* [en línea]. 2021. [Consulta: 14 diciembre 2021]. Disponible en: <https://docs.nestjs.com>.

ORCANOS, Requirements Management Tool for Medical Device. *ALM Software Tool – Orcanos Software – ALM And Quality Management* [en línea]. 2021. [Consulta: 14 diciembre 2021]. Disponible en: <https://www.orcanos.com/compliance/products/requirements-management/>.

OSSENO SOFTWARE GMBH, ReqSuite® RM - The Requirements Management Tool that Thinks Ahead. [en línea]. 2021. [Consulta: 14 diciembre 2021]. Disponible en: <https://www.osseno.com/en/requirements-management-tool/>.

SCHWABER, K. y SUTHERLAND, J., *The Scrum Guide*. S.l.: s.n. 2020.

SCRUMSTUDY, *A guide to the Scrum Body of knowledge (SBOK Guide)*. 2016 edition. Phoenix, Arizona: SCRUMstudy, A brand of VMEdU, Inc. 2016. ISBN 978-0-9899252-0-4. MLCM 2018/40693 (Q)

SOMMERVILLE, I., Web section. *Software Engineering 10th edition* [en línea]. 2014. [Consulta: 22 noviembre 2021]. Disponible en: <https://iansommerville.com/software-engineering-book/static/web/archpatterns/event-driven-control/>.

SOMMERVILLE, I., *Software engineering*. 10. ed., global ed. Boston Munich: Pearson. Always learning. 2016. ISBN 978-1-292-09613-1.

THE POSTGRESQL GLOBAL DEVELOPMENT GROUP, PostgreSQL: About. [en línea]. 2021. [Consulta: 10 enero 2022]. Disponible en: <https://www.postgresql.org/about/>.

VISURE SOLUTIONS, Requirements Management Tool. *Visure Solutions* [en línea]. 2021. [Consulta: 14 diciembre 2021]. Disponible en: <https://visuresolutions.com/>.

WAGNER, S. y RUHE, M., A Systematic Review of Productivity Factors in Software Development. En: arXiv: 1801.06475, *arXiv:1801.06475 [cs]* [en línea]. 2018. [Consulta: 23 abril 2022]. Disponible en: <http://arxiv.org/abs/1801.06475>.

XEBRIO, Requirements Management Software and Tool | Xebrio. [en línea]. 2021. [Consulta: 14 diciembre 2021]. Disponible en: <https://www.xebrio.com/requirements-management-software/>.

КУЛЯМИН, В.В., ПАКУЛИН, Н.В., ПЕТРЕНКО, О.Л., СОРТОВ, А.А. y ХОРОШИЛОВ, А.В., Формализация требований на практике. , pp. 35. 2006.

ANEXOS

ANEXO A: DIAGRAMAS DE SECUENCIA

El diagrama de secuencias es un esquema conceptual que permite representar el comportamiento de un sistema, para lo cual emplea la especificación de los objetos que se encuentran en un escenario y la secuencia de mensajes intercambiados entre ellos, con el fin de llevar a cabo una transacción del sistema. A continuación, se presentan los diagramas de secuencias de las funcionalidades más importantes del sistema.

Diagrama de secuencia de actualizar un requerimiento de un proyecto

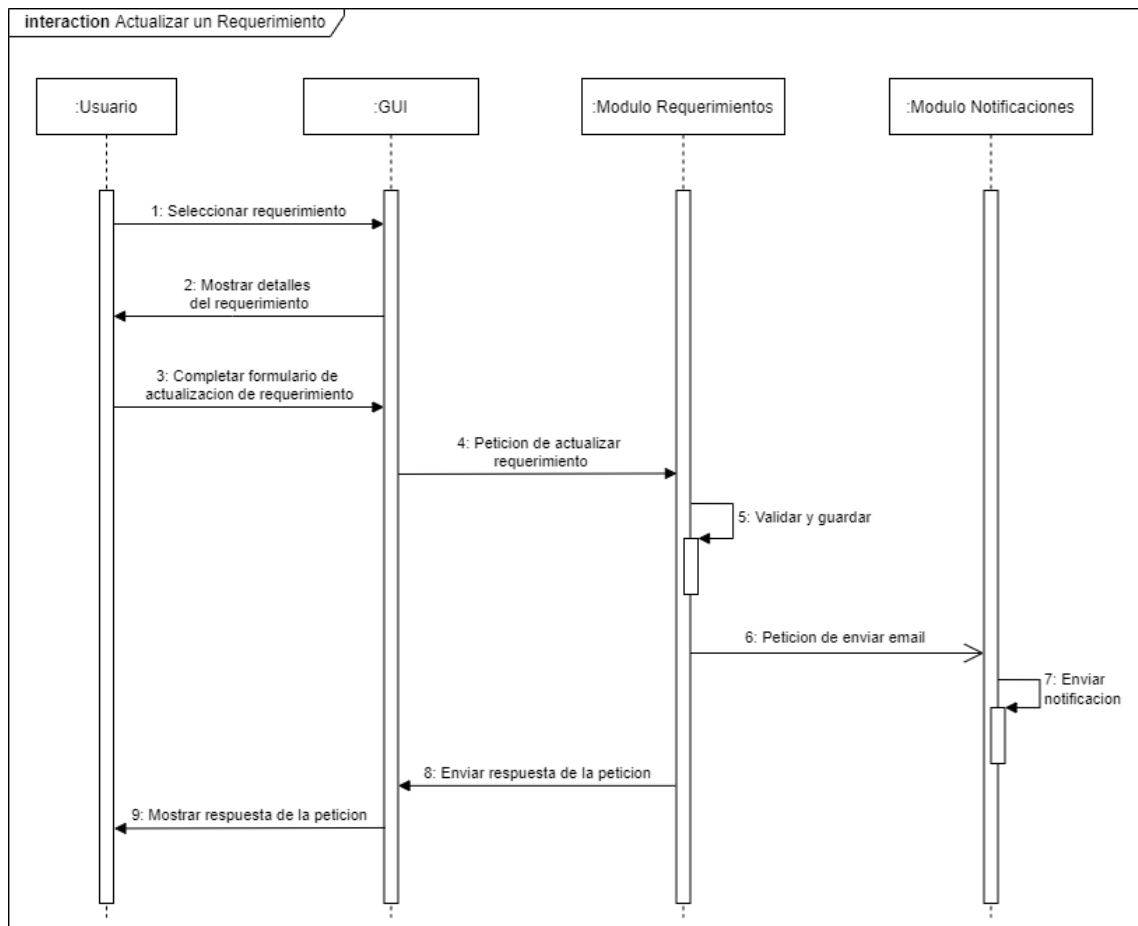


Figura 1. Diagrama de secuencia de actualizar un requerimiento de un proyecto.

Realizado por: Requelme José, 2022.

Diagrama de secuencia de eliminar un proyecto

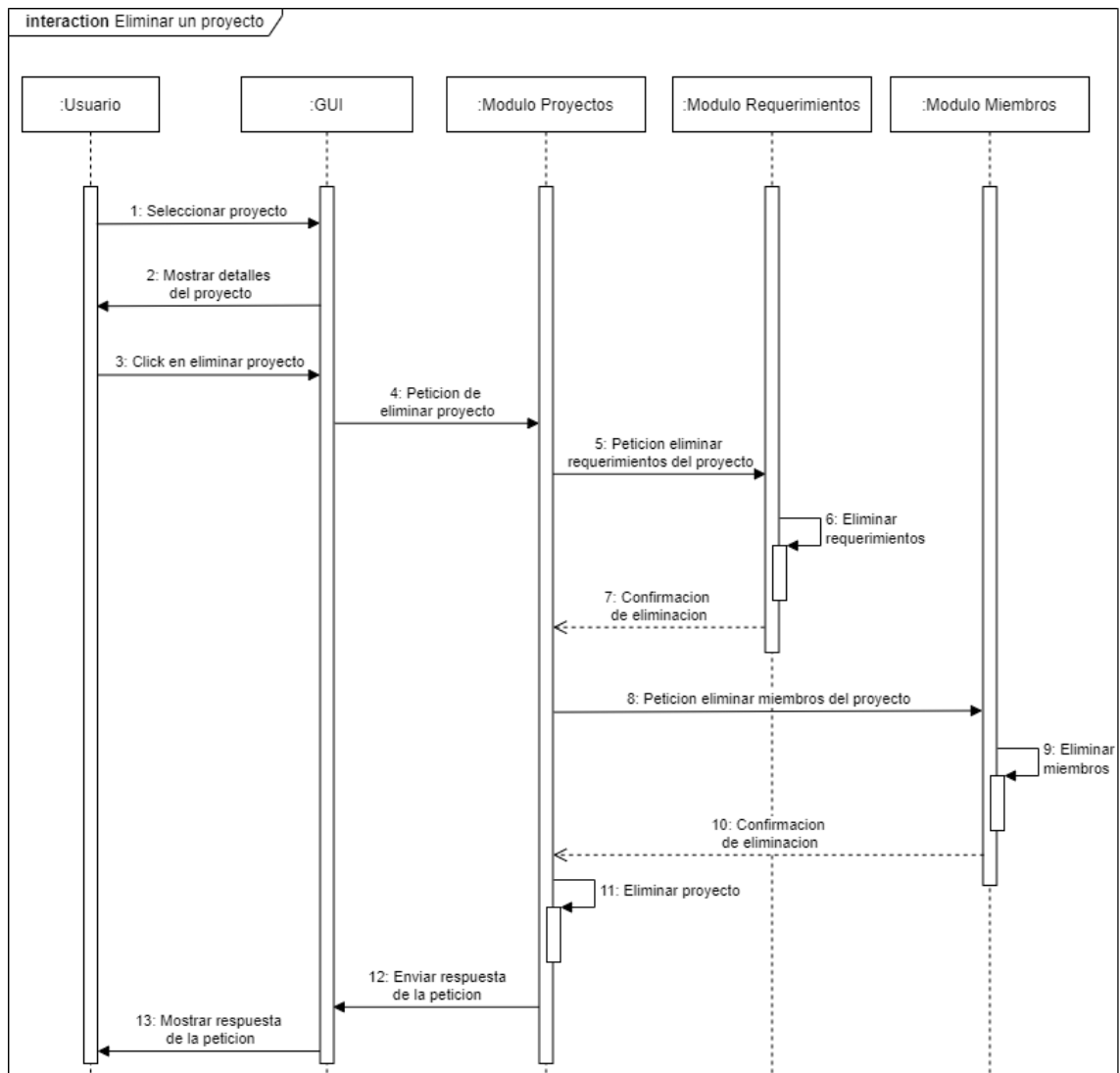


Figura 2. Diagrama de secuencia de eliminar un proyecto.

Realizado por: Requelme José, 2022.

Diagrama de secuencia de listar versiones de un requerimiento

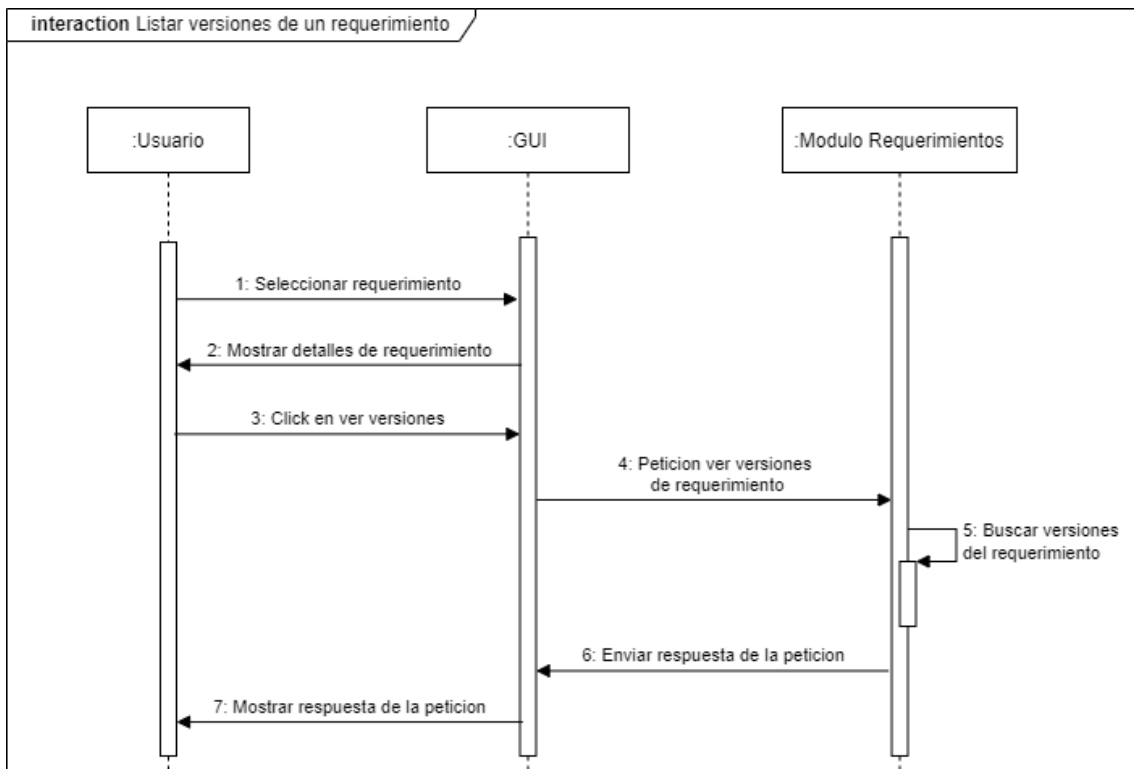


Figura 3. Diagrama de secuencia de listar versiones de un requerimiento.

Realizado por: Requelme José, 2022.

Diagrama de secuencia de generar reporte PDF de los requerimientos del proyecto

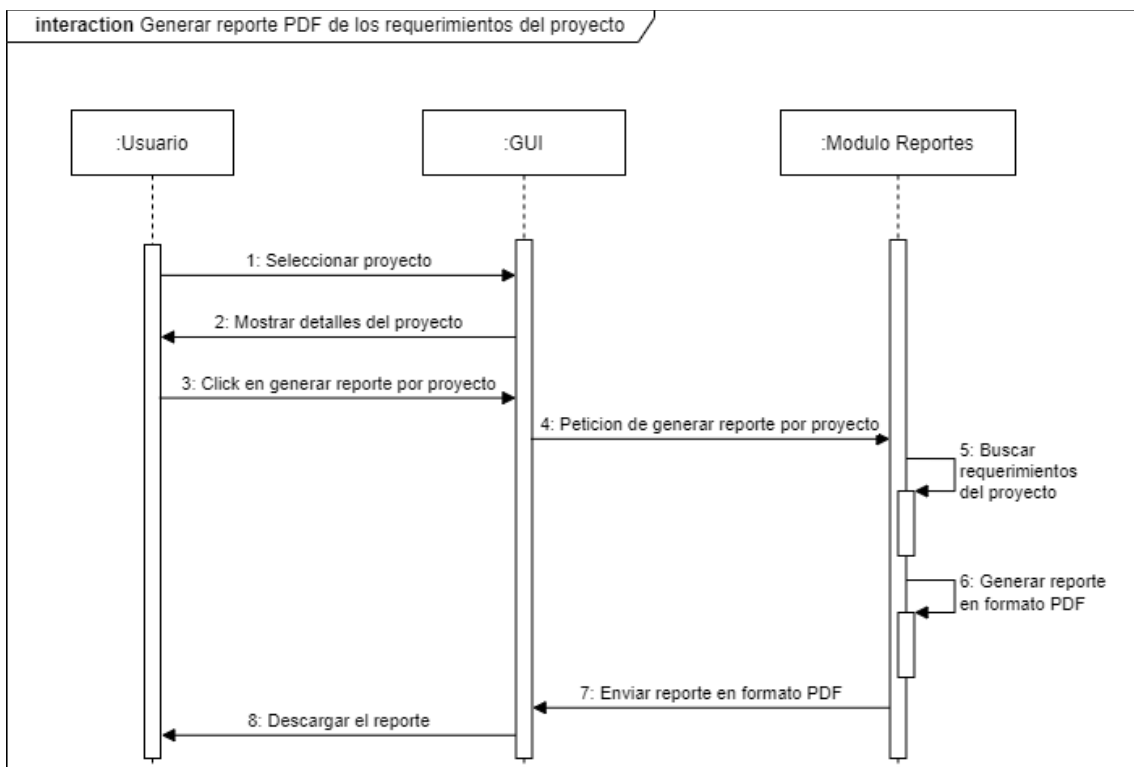


Figura 4. Diagrama de generar reporte PDF de los requerimientos del proyecto.

Realizado por: Requelme José, 2022.

ANEXO B: DICCIONARIO DE DATOS

A continuación, se presenta el diccionario de datos, en el cual identificamos: el campo, tipo de dato, valor por defecto, si es clave primaria y si permite valores nulos; para cada tabla de la base de datos.

Tabla 1: Tabla User

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id	uuid	uuid_generate_v4()	NO	SI
username	varchar(15)		NO	NO
firstName	varchar(20)		NO	NO
lastName	varchar(20)		NO	NO
email	varchar(150)		NO	NO
password	text		NO	NO
avatar	text	'https://cdn-icons-png.flaticon.com/512/149/149071.png'	NO	NO
createdAt	timestamp with time zone	now()	NO	NO
updatedAt	timestamp with time zone	now()	NO	NO

Realizado por: Requelme José, 2022.

Tabla 2: Tabla Project

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id	uuid	uuid_generate_v4()	NO	SI
title	varchar(20)		NO	NO
description	text		SI	NO
numMembers	integer	0	NO	NO
numRequirements	integer	0	NO	NO
createdAt	timestamp with time zone	now()	NO	NO
updatedAt	timestamp with time zone	now()	NO	NO

Realizado por: Requelme José, 2022.

Tabla 3: Tabla Member

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
userId	uuid		NO	SI
projectId	uuid		NO	SI
role	enum	'editor'	NO	NO

Realizado por: Requelme José, 2022.

Tabla 4: Tabla Requirement

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id	uuid	uuid_generate_v4()	NO	SI
priority	enum	'low'	NO	NO
status	enum	'created'	NO	NO
parentId	uuid		SI	NO
projectId	uuid		NO	NO
lastVersion	integer	1	NO	NO
createdAt	timestamp with time zone	now()	NO	NO
updatedAt	timestamp with time zone	now()	NO	NO

Realizado por: Requelme José, 2022.

Tabla 5: Tabla Requirement_Closure

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id_ancestor	uuid		NO	SI
id_descendant	uuid		NO	SI

Realizado por: Requelme José, 2022.

Tabla 6: Tabla Version

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id	uuid	uuid_generate_v4()	NO	SI
versionNum	integer		NO	NO

name	varchar(80)		NO	NO
description	text		NO	NO
fromId	uuid		SI	NO
createdAt	timestamp with time zone	now()	NO	NO
updatedAt	timestamp with time zone	now()	NO	NO

Realizado por: Requelme José, 2022.

Tabla 7: Tabla Requirement_Version

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
requirementId	uuid		NO	SI
versionId	uuid		NO	SI

Realizado por: Requelme José, 2022.

Tabla 8: Tabla Manager

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id	uuid	uuid_generate_v4()	NO	SI
responsibility	enum		NO	NO
memberUser	uuid		NO	NO
memberProject	uuid		NO	NO
requirementId	uuid		NO	NO
createdAt	timestamp with time zone	now()	NO	NO
updatedAt	timestamp with time zone	now()	NO	NO

Realizado por: Requelme José, 2022.

Tabla 9: Tabla Notification

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id	uuid	uuid_generate_v4()	NO	SI
data	text		NO	NO
isDeleted	boolean	false	NO	NO

readAt	timestamp without time zone		SI	NO
createdAt	timestamp with time zone	now()	NO	NO
updatedAt	timestamp with time zone	now()	NO	NO

Realizado por: Requelme José, 2022.

Tabla 10: Tabla Notification_User

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
notificationId	uuid		NO	SI
userId	uuid		NO	SI

Realizado por: Requelme José, 2022.

Tabla 11: Tabla Comment

Campo	Tipo de Dato	Valor por defecto	Permite nulos	Clave primaria
id	uuid	uuid_generate_v4()	NO	SI
text	text		NO	NO
requirementId	uuid		NO	NO
userId	uuid		NO	NO
createdAt	timestamp with time zone	now()	NO	NO
updatedAt	timestamp with time zone	now()	NO	NO

Realizado por: Requelme José, 2022.

ANEXO C: HISTORIAS DE USUARIO Y TÉCNICAS

A continuación, se presentan las historias técnicas y las historias de usuario de la herramienta desarrollada, cabe recalcar que cada historia técnica e historia de usuario tienen su respectiva prueba de aceptación y tarea de ingeniería.

Tabla 1: Historia Técnica sobre definir el estándar de codificación.

Historia Técnica			
ID: HT-01	Nombre de la historia: Definir el estándar de codificación		
Usuario: Desarrollador	Sprint: 1		
Prioridad: Alta (Alta/Media/Baja)	Puntos Estimados: 8		
	Puntos Reales: 8		
Fecha Inicio: 15/11/2021	Fecha Fin: 16/11/2021		
Descripción: Como desarrollador quiero definir el estándar de codificación para mantener la homogeneidad y consistencia del código escrito			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HT-01 PA-01	Determinar el estándar adecuado para la codificación de la aplicación	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HT-01 TI-01	Definir las reglas del estándar de codificación en el IDE de programación, mediante la ayuda de eslint	8	

Realizado por: Requelme José, 2022.

Tabla 2: Historia Técnica sobre definir la arquitectura de la aplicación.

Historia Técnica			
ID: HT-02	Nombre de la historia: Definir la arquitectura de la aplicación		
Usuario: Desarrollador	Sprint: 1		
Prioridad: Alta (Alta/Media/Baja)	Puntos Estimados: 8		
	Puntos Reales: 8		
Fecha Inicio: 17/11/2021	Fecha Fin: 18/11/2021		
Descripción: Como desarrollador quiero definir la arquitectura de la aplicación para establecer una base lógica de como estará construida la aplicación			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable

HT-02 PA-01	Examinar el diseño de la arquitectura de la aplicación web	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HT-02 TI-01	Analizar el tipo de arquitectura más adecuada para la aplicación	6	
HT-02 TI-02	Realizar el diagrama UML para documentar la arquitectura de la aplicación	2	

Realizado por: Requelme José, 2022.

Tabla 3: Historia Técnica sobre el diseño de la base de datos.

Historia Técnica			
ID: HT-03	Nombre de la historia: Diseño de la base de datos		
Usuario: Desarrollador	Sprint: 1		
Prioridad: Alta (Alta/Media/Baja)	Puntos Estimados: 24		
	Puntos Reales: 24		
Fecha Inicio: 19/11/2021	Fecha Fin: 26/11/2021		
Descripción: Como desarrollador quiero realizar el diseño de la base de datos para almacenar y mantener la información organizada			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HT-03 PA-01	Verificar la correcta relación de las entidades	Exitoso	José Requelme
HT-03 PA-02	Verificar que las tablas de la base de datos estén normalizadas	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HT-03 TI-01	Realizar el diseño lógico y conceptual de la base de datos	16	
HT-03 TI-02	Realizar la normalización de las tablas de la base de datos	8	

Realizado por: Requelme José, 2022.

Tabla 4: Historia Técnica sobre diseñar las interfaces de usuario.

Historia Técnica			
ID: HT-04	Nombre de la historia: Diseñar las interfaces de usuario		
Usuario: Desarrollador		Sprint: 2	
Prioridad: Alta		Puntos Estimados: 8	
(Alta/Media/Baja)		Puntos Reales: 8	
Fecha Inicio: 29/11/2021		Fecha Fin: 30/11/2021	
Descripción: Como desarrollador quiero diseñar las interfaces de usuario para ofrecer una interfaz con una buena accesibilidad y usabilidad			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HT-04 PA-01	Verificar que las interfaces mantengan una alta similitud entre diferentes navegadores	Exitoso	José Requelme
HT-04 PA-02	Verificar que las interfaces se adapten correctamente a diferentes tamaños de pantalla	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HT-04 TI-01	Determinar los colores primarios y secundarios de la aplicación	2	
HT-04 TI-02	Diseñar las interfaces de las aplicaciones	6	

Realizado por: Requelme José, 2022.

Tabla 5: Historia Técnica sobre generar diccionario de datos.

Historia Técnica			
ID: HT-05	Nombre de la historia: Generar diccionario de datos		
Usuario: Desarrollador		Sprint: 2	
Prioridad: Media		Puntos Estimados: 8	
(Alta/Media/Baja)		Puntos Reales: 8	
Fecha Inicio: 01/12/2021		Fecha Fin: 02/12/2021	
Descripción: Como desarrollador quiero generar el diccionario de datos para dar precisión semántica sobre los datos que se manejan en la base de datos			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HT-05 PA-01	Verificar que la semántica de los datos coincida con el diccionario de datos	Exitoso	José Requelme

Tareas de Ingeniería		
ID_TI	Descripción TI	Esfuerzo
HT-05 TI-01	Realizar el diccionario de datos	8

Realizado por: Requelme José, 2022.

Tabla 6: Historia Técnica sobre la documentación.

Historia Técnica			
ID: HT-06	Nombre de la historia: Documentación		
Usuario: Desarrollador	Sprint: 2		
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 24		
	Puntos Reales: 24		
Fecha Inicio: 03/12/2021	Fecha Fin: 10/12/2021		
Descripción: Como desarrollador quiero realizar la documentación de todo el proceso de construcción de la aplicación desde su fase inicial hasta su entrega final para tener evidencia de lo trabajado			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HT-06 PA-01	Verificar que estén descritas todas las fases de la metodología de desarrollo SCRUM	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HT-06 TI-01	Realizar la documentación de todas las fases de la metodología de desarrollo SCRUM	24	

Realizado por: Requelme José, 2022.

Tabla 7: Historia de Usuario sobre crear una nueva cuenta de usuario en la aplicación web.

Historia de Usuario	
ID: HU-01	Nombre de la historia: Crear una nueva cuenta de usuario en la aplicación web
Usuario: Usuario	Sprint: 3
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 8
	Puntos Reales: 8
Fecha Inicio: 13/12/2021	Fecha Fin: 14/12/2021
Descripción: Como usuario quiero crearme una cuenta para poder utilizar los servicios de la aplicación web	

Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-01 PA-01	Verificar que el nombre de usuario sea único	Exitoso	José Requelme
HU-01 PA-02	Verificar que el correo electrónico sea único	Exitoso	José Requelme
HU-01 PA-03	No permitir el ingreso de campos vacíos	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-01 TI-01	Desarrollar la interfaz de usuario para el registro de una nueva cuenta y conectar con el backend	4	
HU-01 TI-02	Desarrollar el servicio web RESTful para la creación de una nueva cuenta	4	

Realizado por: Requelme José, 2022.

Tabla 8: Historia de Usuario sobre recuperar contraseña mediante su correo electrónico.

Historia de Usuario			
ID: HU-02	Nombre de la historia: Recuperar contraseña mediante su correo electrónico		
Usuario: Usuario	Sprint: 3		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 16		
	Puntos Reales: 16		
Fecha Inicio: 15/12/2021	Fecha Fin: 20/12/2021		
Descripción: Como usuario quiero recuperar mi contraseña mediante correo electrónico para poder tener acceso a mi cuenta en caso de que me olvide mí contraseña.			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-02 PA-01	Verificar que el usuario exista	Exitoso	José Requelme
HU-02 PA-02	Verificar que el correo sea enviado al email correspondiente	Exitoso	José Requelme
HU-02 PA-03	Verificar que la contraseña se haya actualizado	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-02	Configurar el servidor de correo electrónico y los servicios	12	

TI-01		
HU-02 TI-02	Diseñar la plantilla del correo y enlazarlo con el servicio	4

Realizado por: Requelme José, 2022.

Tabla 9: Historia de Usuario sobre iniciar sesión en la aplicación web.

Historia de Usuario			
ID: HU-03	Nombre de la historia: Iniciar sesión en la aplicación web		
Usuario: Usuario	Sprint: 3		
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 8		
	Puntos Reales: 8		
Fecha Inicio: 21/12/2021	Fecha Fin: 22/12/2021		
Descripción: Como usuario quiero iniciar sesión para poder utilizar los servicios de la aplicación web			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-03 PA-01	Verificar que las credenciales sean correctas	Exitoso	José Requelme
HU-03 PA-02	No permitir el ingreso de campos vacíos	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-03 TI-01	Desarrollar el servicio web RESTful para el inicio de sesión	4	
HU-03 TI-02	Desarrollar la interfaz de usuario para el inicio de sesión y conectar con el backend	4	

Realizado por: Requelme José, 2022.

Tabla 10: Historia de Usuario sobre actualizar la información personal del usuario.

Historia de Usuario			
ID: HU-04	Nombre de la historia: Actualizar la información personal del usuario		
Usuario: Usuario	Sprint: 3		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 4		
	Puntos Reales: 4		
Fecha Inicio: 23/12/2021	Fecha Fin: 23/12/2021		

Descripción: Como usuario quiero actualizar mi información personal para mantener mi información al día			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-04 PA-01	Verificar si el usuario mantiene el access token de inicio de sesión	Exitoso	José Requelme
HU-04 PA-02	No permitir el ingreso de campos vacíos	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-04 TI-01	Desarrollar el servicio web RESTful para actualizar la información del usuario	2	
HU-04 TI-02	Desarrollar la interfaz de usuario para actualizar la información del usuario y conectar con el backend	2	

Realizado por: Requelme José, 2022.

Tabla 11: Historia de Usuario sobre actualizar la contraseña del usuario.

Historia de Usuario			
ID: HU-05	Nombre de la historia: Actualizar la contraseña del usuario		
Usuario: Usuario	Sprint: 3		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 4		
	Puntos Reales: 4		
Fecha Inicio: 24/12/2021	Fecha Fin: 24/12/2021		
Descripción: Como usuario quiero actualizar la contraseña para mantener mi cuenta más segura			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-05 PA-01	Verificar si el usuario mantiene el access token de inicio de sesión	Exitoso	José Requelme
HU-05 PA-02	No permitir el ingreso de campos vacíos	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-05 TI-01	Desarrollar el servicio web RESTful para actualizar la contraseña del usuario	2	

HU-05 TI-02	Desarrollar la interfaz de usuario para actualizar la contraseña del usuario y conectar con el backend	2
----------------	--	---

Realizado por: Requelme José, 2022.

Tabla 12: Historia de Usuario sobre crear un nuevo proyecto.

Historia de Usuario			
ID: HU-06	Nombre de la historia: Crear un nuevo proyecto		
Usuario: Usuario	Sprint: 4		
Prioridad: Alta (Alta/Media/Baja)	Puntos Estimados: 16		
	Puntos Reales: 16		
Fecha Inicio: 03/01/2022	Fecha Fin: 06/01/2022		
Descripción: Como usuario quiero crear un nuevo proyecto para organizar los requisitos del proyecto y gestionar al equipo de desarrollo			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-06 PA-01	Verificar que el nombre del proyecto no exista con anterioridad	Exitoso	José Requelme
HU-06 PA-02	No permitir el ingreso de campos vacíos	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-06 TI-01	Desarrollar el servicio web RESTful para crear un nuevo proyecto	12	
HU-06 TI-02	Desarrollar la interfaz de usuario para crear un nuevo proyecto y conectar con el backend	4	

Realizado por: Requelme José, 2022.

Tabla 13: Historia de Usuario sobre actualizar la información de un proyecto.

Historia de Usuario			
ID: HU-07	Nombre de la historia: Actualizar la información de un proyecto		
Usuario: Usuario	Sprint: 4		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 4		
	Puntos Reales: 4		
Fecha Inicio: 07/01/2022	Fecha Fin: 07/01/2022		

Descripción: Como usuario quiero actualizar la información de un proyecto para mantener los datos del proyecto al día			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-07 PA-01	Verificar que el usuario que actualice la información sea el dueño del proyecto	Exitoso	José Requelme
HU-07 PA-02	No permitir el ingreso de campos vacíos	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-07 TI-01	Desarrollar el servicio web RESTful para actualizar la información de un proyecto	2	
HU-07 TI-02	Desarrollar la interfaz de usuario para actualizar la información de un proyecto y conectar con el backend	2	

Realizado por: Requelme José, 2022.

Tabla 14: Historia de Usuario sobre eliminar un proyecto.

Historia de Usuario			
ID: HU-08	Nombre de la historia: Eliminar un proyecto		
Usuario: Usuario	Sprint: 4		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 8		
	Puntos Reales: 8		
Fecha Inicio: 10/01/2022	Fecha Fin: 11/01/2022		
Descripción: Como usuario quiero eliminar un proyecto para solo tener proyectos en los que se esté trabajando			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-08 PA-01	Verificar que el proyecto exista	Exitoso	José Requelme
HU-08 PA-02	Verificar que el usuario que elimine el proyecto sea el dueño del mismo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-08 TI-01	Desarrollar el servicio web RESTful para eliminar un proyecto	4	

HU-08 TI-02	Desarrollar la interfaz de usuario para eliminar un proyecto y conectar con el backend	4
----------------	--	---

Realizado por: Requelme José, 2022.

Tabla 15: Historia de Usuario sobre añadir un nuevo miembro al proyecto.

Historia de Usuario			
ID: HU-09	Nombre de la historia: Añadir un nuevo miembro al proyecto		
Usuario: Usuario		Sprint: 4	
Prioridad: Media (Alta/Media/Baja)		Puntos Estimados: 8	
		Puntos Reales: 8	
Fecha Inicio: 12/01/2022		Fecha Fin: 13/01/2022	
Descripción: Como usuario quiero añadir un nuevo miembro al proyecto para poder asignarlo a las tareas correspondientes del proyecto			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-09 PA-01	Verificar que el usuario exista	Exitoso	José Requelme
HU-09 PA-02	Verificar que el usuario no sea miembro del proyecto con anterioridad	Exitoso	José Requelme
HU-09 PA-03	Verificar que el usuario que añade al miembro sea el dueño del proyecto	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-09 TI-01	Desarrollar el servicio web RESTful para añadir un nuevo miembro al proyecto	4	
HU-09 TI-02	Desarrollar la interfaz de usuario para añadir un nuevo miembro al proyecto y conectar con el backend	4	

Realizado por: Requelme José, 2022.

Tabla 16: Historia de Usuario sobre actualizar el rol de los miembros del proyecto.

Historia de Usuario		
ID: HU-10	Nombre de la historia: Actualizar el rol de los miembros del proyecto	
Usuario: Usuario		Sprint: 4
Prioridad: Baja (Alta/Media/Baja)		Puntos Estimados: 4
		Puntos Reales: 4

Fecha Inicio: 14/01/2022		Fecha Fin: 14/01/2022	
Descripción: Como usuario quiero actualizar el rol de los miembros del proyecto para poder darle al miembro los permisos correspondientes de ser el caso			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-10 PA-01	Verificar que el miembro exista y pertenezca al proyecto	Exitoso	José Requelme
HU-10 PA-02	Verificar que el usuario que realice la actualización tenga los permisos para hacerlo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-10 TI-01	Desarrollar el servicio web RESTful para actualizar el rol de los miembros del proyecto	2	
HU-10 TI-02	Desarrollar la interfaz de usuario para actualizar el rol de los miembros del proyecto y conectar con el backend	2	

Realizado por: Requelme José, 2022.

Tabla 17: Historia de Usuario sobre eliminar un miembro del proyecto.

Historia de Usuario			
ID: HU-11	Nombre de la historia: Eliminar un miembro del proyecto		
Usuario: Usuario	Sprint: 5		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 4		
	Puntos Reales: 4		
Fecha Inicio: 17/01/2022	Fecha Fin: 17/01/2022		
Descripción: Como usuario quiero eliminar un miembro del proyecto para solo darle acceso a mi proyecto a quienes autorice			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-11 PA-01	Verificar que el miembro exista y pertenezca al proyecto	Exitoso	José Requelme
HU-11 PA-02	Verificar que el usuario que desee eliminar al miembro tenga los permisos para hacerlo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	

HU-11 TI-01	Desarrollar el servicio web RESTful para eliminar un miembro del proyecto	2
HU-11 TI-02	Desarrollar la interfaz de usuario para eliminar un miembro del proyecto y conectar con el backend	2

Realizado por: Requelme José, 2022.

Tabla 18: Historia de Usuario sobre añadir un nuevo requerimiento al proyecto.

Historia de Usuario			
ID: HU-12	Nombre de la historia: Añadir un nuevo requerimiento al proyecto		
Usuario: Usuario		Sprint: 5	
Prioridad: Alta (Alta/Media/Baja)		Puntos Estimados: 8	
		Puntos Reales: 8	
Fecha Inicio: 18/01/2022		Fecha Fin: 19/01/2022	
Descripción: Como usuario quiero añadir un nuevo requerimiento al proyecto para poder gestionar los requerimientos de una manera más organizada			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-12 PA-01	No permitir el ingreso de campos vacíos	Exitoso	José Requelme
HU-12 PA-02	Verificar que el usuario que desee añadir un nuevo requerimiento tenga los permisos para hacerlo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-12 TI-01	Desarrollar el servicio web RESTful para añadir un nuevo requerimiento al proyecto	4	
HU-12 TI-02	Desarrollar la interfaz de usuario para añadir un nuevo requerimiento al proyecto y conectar con el backend	4	

Realizado por: Requelme José, 2022.

Tabla 19: Historia de Usuario sobre asignar un editor y un reviewer al requerimiento.

Historia de Usuario		
ID: HU-13	Nombre de la historia: Asignar un editor y un reviewer al requerimiento	
Usuario: Usuario		Sprint: 5
Prioridad: Media (Alta/Media/Baja)		Puntos Estimados: 4
		Puntos Reales: 4

Fecha Inicio: 20/01/2022		Fecha Fin: 20/01/2022	
Descripción: Como usuario quiero asignar un editor y un reviewer al requerimiento para que sean los responsables del mismo			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-13 PA-01	Verificar que tanto el editor como el reviewer sean miembros del proyecto	Exitoso	José Requelme
HU-13 PA-02	Verificar que el usuario que desee asignar un editor y reviewer tenga los permisos para hacerlo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-13 TI-01	Desarrollar el servicio web RESTful para asignar un editor y un reviewer al requerimiento	2	
HU-13 TI-02	Desarrollar la interfaz de usuario para asignar un editor y un reviewer al requerimiento y conectar con el backend	2	

Realizado por: Requelme José, 2022.

Tabla 20: Historia de Usuario sobre vincular con otro requerimiento de ser necesario.

Historia de Usuario			
ID: HU-14	Nombre de la historia: Vincular con otro requerimiento de ser necesario		
Usuario: Usuario	Sprint: 5		
Prioridad: Alta	Puntos Estimados: 4		
(Alta/Media/Baja)	Puntos Reales: 4		
Fecha Inicio: 21/01/2022	Fecha Fin: 21/01/2022		
Descripción: Como usuario quiero vincular un requerimiento con otro requerimiento en caso de ser necesario para poder realizar una correcta trazabilidad			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-14 PA-01	Verificar que el requerimiento con el que se desee vincular exista y pertenezca al mismo proyecto	Exitoso	José Requelme
HU-14 PA-02	Verificar que el usuario que desee vincular un requerimiento tenga los permisos para hacerlo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	

HU-14 TI-01	Desarrollar el servicio web RESTful para vincular un requerimiento	2
HU-14 TI-02	Desarrollar la interfaz de usuario para vincular un requerimiento y conectar con el backend	2

Realizado por: Requelme José, 2022.

Tabla 21: Historia de Usuario sobre actualizar la información del requerimiento.

Historia de Usuario			
ID: HU-15	Nombre de la historia: Actualizar la información del requerimiento		
Usuario: Usuario		Sprint: 5	
Prioridad: Alta (Alta/Media/Baja)		Puntos Estimados: 8	
		Puntos Reales: 8	
Fecha Inicio: 24/01/2022		Fecha Fin: 25/01/2022	
Descripción: Como usuario quiero actualizar la información del requerimiento para mantener la información al día			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-15 PA-01	Verificar que el requerimiento exista	Exitoso	José Requelme
HU-15 PA-02	Verificar que el usuario que desee actualizar el requerimiento tenga los permisos para hacerlo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-15 TI-01	Desarrollar el servicio web RESTful para actualizar la información del requerimiento	4	
HU-15 TI-02	Desarrollar la interfaz de usuario para actualizar la información del requerimiento y conectar con el backend	4	

Realizado por: Requelme José, 2022.

Tabla 22: Historia de Usuario sobre visualizar las versiones de cambios del requerimiento.

Historia de Usuario			
ID: HU-16	Nombre de la historia: Visualizar las versiones de cambios del requerimiento		
Usuario: Usuario		Sprint: 5	
Prioridad: Alta (Alta/Media/Baja)		Puntos Estimados: 8	
		Puntos Reales: 8	

Fecha Inicio: 26/01/2022		Fecha Fin: 27/01/2022	
Descripción: Como usuario quiero visualizar las versiones de cambios del requerimiento para tener un mayor control de los cambios que se han realizado			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-16 PA-01	Verificar que el requerimiento exista	Exitoso	José Requelme
HU-16 PA-02	Verificar que la información se muestre de manera correcta	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-16 TI-01	Desarrollar el servicio web RESTful para visualizar las versiones de cambios del requerimiento	4	
HU-16 TI-02	Desarrollar la interfaz de usuario para visualizar las versiones de cambios del requerimiento y conectar con el backend	4	

Realizado por: Requelme José, 2022.

Tabla 23: Historia de Usuario sobre retroceder a la versión del requerimiento deseado.

Historia de Usuario			
ID: HU-17	Nombre de la historia: Retroceder a la versión del requerimiento deseado		
Usuario: Usuario	Sprint: 5		
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 4		
	Puntos Reales: 4		
Fecha Inicio: 28/01/2022	Fecha Fin: 28/01/2022		
Descripción: Como usuario quiero retroceder a la versión del requerimiento deseado para poder trabajar desde una versión específica en caso de que la última versión no sea correcta			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-17 PA-01	Verificar que la versión del requerimiento exista	Exitoso	José Requelme
HU-17 PA-02	Verificar que se haya creado una nueva versión basada en los datos de la versión deseada	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	

HU-17 TI-01	Desarrollar el servicio web RESTful para retroceder a la versión del requerimiento deseado	2
HU-17 TI-02	Desarrollar la interfaz de usuario para retroceder a la versión del requerimiento deseado y conectar con el backend	2

Realizado por: Requelme José, 2022.

Tabla 24: Historia de Usuario sobre visualizar la trazabilidad de los requerimientos.

Historia de Usuario			
ID: HU-18	Nombre de la historia: Visualizar la trazabilidad de los requerimientos		
Usuario: Usuario		Sprint: 6	
Prioridad: Alta (Alta/Media/Baja)		Puntos Estimados: 8	
		Puntos Reales: 8	
Fecha Inicio: 31/01/2022		Fecha Fin: 01/02/2022	
Descripción: Como usuario quiero visualizar la trazabilidad de los requerimientos para poder tener un mayor control de dichos requerimientos			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-18 PA-01	Verificar que el requerimiento exista	Exitoso	José Requelme
HU-18 PA-02	Verificar que la información se muestre de manera correcta	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-18 TI-01	Desarrollar el servicio web RESTful para visualizar la trazabilidad de los requerimientos	4	
HU-18 TI-02	Desarrollar la interfaz de usuario para visualizar la trazabilidad de los requerimientos y conectar con el backend	4	

Realizado por: Requelme José, 2022.

Tabla 25: Historia de Usuario sobre generar un PDF con los requerimientos del proyecto.

Historia de Usuario		
ID: HU-19	Nombre de la historia: Generar un PDF con los requerimientos del proyecto	
Usuario: Usuario		Sprint: 6
Prioridad: Baja (Alta/Media/Baja)		Puntos Estimados: 4
		Puntos Reales: 4

Fecha Inicio: 02/02/2022		Fecha Fin: 02/02/2022	
Descripción: Como usuario quiero generar un PDF con los requerimientos del proyecto para tener los requerimientos del proyecto documentado			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-19 PA-01	Verificar que el proyecto exista	Exitoso	José Requelme
HU-19 PA-02	Verificar que el PDF se genere de manera exitosa	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-19 TI-01	Desarrollar el servicio web RESTful para generar un PDF con los requerimientos del proyecto	2	
HU-19 TI-02	Desarrollar la interfaz de usuario para generar un PDF con los requerimientos del proyecto y conectar con el backend	2	

Realizado por: Requelme José, 2022.

Tabla 26: Historia de Usuario sobre generar un PDF de las versiones del requerimiento.

Historia de Usuario			
ID: HU-20	Nombre de la historia: Generar un PDF de las versiones del requerimiento		
Usuario: Usuario	Sprint: 6		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 4		
	Puntos Reales: 4		
Fecha Inicio: 03/02/2022	Fecha Fin: 03/02/2022		
Descripción: Como usuario quiero generar un PDF de las versiones del requerimiento para tener documentado las versiones del requerimiento deseado			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-20 PA-01	Verificar que el requerimiento exista	Exitoso	José Requelme
HU-20 PA-02	Verificar que el PDF se genere de manera exitosa	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	

HU-20 TI-01	Desarrollar el servicio web RESTful para generar un PDF de las versiones del requerimiento	2
HU-20 TI-02	Desarrollar la interfaz de usuario para generar un PDF de las versiones del requerimiento y conectar con el backend	2

Realizado por: Requelme José, 2022.

Tabla 27: Historia de Usuario sobre añadir un comentario a un requerimiento.

Historia de Usuario			
ID: HU-21	Nombre de la historia: Añadir un comentario a un requerimiento		
Usuario: Usuario		Sprint: 6	
Prioridad: Baja (Alta/Media/Baja)		Puntos Estimados: 4	
		Puntos Reales: 4	
Fecha Inicio: 04/02/2022		Fecha Fin: 04/02/2022	
Descripción: Como usuario quiero añadir un comentario a un requerimiento para permitir a los miembros del proyecto contribuir con sus sugerencias y opiniones			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-21 PA-01	Verificar que el requerimiento exista	Exitoso	José Requelme
HU-21 PA-02	Verificar que el usuario que escribe el comentario sea un miembro del proyecto	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-21 TI-01	Desarrollar el servicio web RESTful para añadir un comentario a un requerimiento	2	
HU-21 TI-02	Desarrollar la interfaz de usuario para añadir un comentario a un requerimiento y conectar con el backend	2	

Realizado por: Requelme José, 2022.

Tabla 28: Historia de Usuario sobre actualizar el contenido del comentario.

Historia de Usuario		
ID: HU-22	Nombre de la historia: Actualizar el contenido del comentario	
Usuario: Usuario		Sprint: 6
Prioridad: Baja (Alta/Media/Baja)		Puntos Estimados: 4
		Puntos Reales: 4

Fecha Inicio: 07/02/2022		Fecha Fin: 07/02/2022	
Descripción: Como usuario quiero actualizar el contenido de un comentario para permitir a los miembros del proyecto refinar sus sugerencias y opiniones			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-22 PA-01	Verificar que el comentario exista	Exitoso	José Requelme
HU-22 PA-02	Verificar que el usuario que actualice el comentario sea el dueño del mismo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-22 TI-01	Desarrollar el servicio web RESTful para actualizar el contenido de un comentario	2	
HU-22 TI-02	Desarrollar la interfaz de usuario para actualizar el contenido de un comentario y conectar con el backend	2	

Realizado por: Requelme José, 2022.

Tabla 29: Historia de Usuario sobre eliminar el comentario.

Historia de Usuario			
ID: HU-23	Nombre de la historia: Eliminar el comentario		
Usuario: Usuario	Sprint: 6		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 4 Puntos Reales: 4		
Fecha Inicio: 08/02/2022	Fecha Fin: 08/02/2022		
Descripción: Como usuario quiero eliminar un comentario para mantener visibles solo los comentarios que aporten valor al requerimiento			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-23 PA-01	Verificar que el comentario exista	Exitoso	José Requelme
HU-23 PA-02	Verificar que el usuario que elimine el comentario sea el dueño del mismo	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-23 TI-01	Desarrollar el servicio web RESTful para eliminar un comentario	2	

HU-23 TI-02	Desarrollar la interfaz de usuario para eliminar un comentario y conectar con el backend	2
----------------	--	---

Realizado por: Requelme José, 2022.

Tabla 30: Historia de Usuario sobre Notificar al editor y reviewer cuando un requerimiento cambie.

Historia de Usuario			
ID: HU-24	Nombre de la historia: Notificar al editor y reviewer cuando un requerimiento cambie		
Usuario: Usuario	Sprint: 6		
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 8		
	Puntos Reales: 8		
Fecha Inicio: 09/02/2022	Fecha Fin: 10/02/2022		
Descripción: Como usuario quiero notificar al editor y reviewer cuando un requerimiento cambie para mantener una transparencia entre la colaboración de los miembros del proyecto			
Pruebas de Aceptación			
ID_PA	Descripción PA	Estado	Responsable
HU-24 PA-01	Verificar que el correo sea enviado exitosamente al editor y reviewer	Exitoso	José Requelme
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU-24 TI-01	Desarrollar el servicio web RESTful para notificar al editor y reviewer cuando un requerimiento cambie	4	
HU-24 TI-02	Diseñar la plantilla del correo y enlazarlo con el servicio	4	

Realizado por: Requelme José, 2022.

Tabla 31: Historia de Usuario sobre Notificar a un usuario cuando se le añade a un proyecto.

Historia de Usuario		
ID: HU-25	Nombre de la historia: Notificar a un usuario cuando se le añade a un proyecto	
Usuario: Usuario	Sprint: 6	
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 4	
	Puntos Reales: 4	
Fecha Inicio: 11/02/2022	Fecha Fin: 11/02/2022	

Descripción: Como usuario quiero notificar a un usuario cuando se le añade a un proyecto para mantener informado al usuario de los proyectos a los pertenece

Pruebas de Aceptación

ID_PA	Descripción PA	Estado	Responsable
HU-25 PA-01	Verificar que el correo sea enviado exitosamente al usuario	Exitoso	José Requelme

Tareas de Ingeniería

ID_TI	Descripción TI	Esfuerzo
HU-25 TI-01	Desarrollar el servicio web RESTful para notificar a un usuario cuando se le añade a un proyecto	2
HU-25 TI-02	Diseñar la plantilla del correo y enlazarlo con el servicio	2

Realizado por: Requelme José, 2022.

ANEXO D: PLANTILLA IEEE 830 QUE SOPORTA TRAZABILIDAD DE REQUISITOS

1. Introducción

1.1. Objetivo

Especificar cuál es el objetivo que persigue el sistema, a través del requerimiento; será detallado con mayor detalle en el resto del documento. Se pretende clarificar la visión del resto de los participantes.

1.2. Alcance

En este punto se deberá establecer el marco de funcionalidades y / o características especiales que establecen el límite del requerimiento.

1.3. Definiciones, Acrónimos y Abreviaturas

Indicar aquí todas las definiciones de Acrónimos, Abreviaturas, como así también toda palabra que requiera determinar un significado común para todos los participantes, en el dominio del problema.

1.4. Referencias

2. Descripción General

2.1. Perspectiva del Producto

Describe lo que se espera lograr con el producto implementado; puede compararse en este punto contra productos anteriores existentes en la organización, como así también con productos de la competencia. Se recomienda expresar el mismo en términos verificables.

2.2. Funciones del Producto

2.3. Características de los usuarios

Indicar brevemente las características de los usuarios directamente relacionados con el requerimiento y las funciones específicas que deberá realizar con el producto.

2.4. Restricciones

Establecer las restricciones generales impuestas sobre el producto: herramientas o productos a utilizar; controles genéricos; etc.

2.5. Suposiciones y Dependencias

Indicar, si existiera, dependencia o relaciones de control con otros productos; servicios específicos relacionados; acuerdos preexistentes; toda otra información de relevancia sobre el producto.

3. Requerimientos Específicos

3.1. Requerimientos Funcionales

Describir todos los requerimientos funcionales que serán tratados por el requerimiento.

3.2. Matriz de Trazabilidad Requerimientos Funcionales

3.3. Requerimientos No Funcionales

Describir todos los requerimientos no funcionales que serán tratados por el requerimiento.

3.4. Matriz de Trazabilidad Requerimientos No Funcionales

4. Apéndices

Indicar toda información a documentación relacionada que se considere relevante: manuales, legislación; diagramas; esquemas; etc.



epoch

**Dirección de Bibliotecas y
Recursos del Aprendizaje**

**UNIDAD DE PROCESOS TÉCNICOS Y ANÁLISIS BIBLIOGRÁFICO Y
DOCUMENTAL**

REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 03 / 08 / 2022

INFORMACIÓN DEL AUTOR/A (S)
Nombres – Apellidos: JOSÉ ALEXANDER REQUELME ROMERO
INFORMACIÓN INSTITUCIONAL
Facultad: INFORMÁTICA Y ELECTRÓNICA
Carrera: SOFTWARE
Título a optar: INGENIERO DE SOFTWARE
f. Analista de Biblioteca responsable: Lcdo. Holger Ramos, MSc.

1613-DBRA-UPT-2022

