



Universidad  
Nacional  
de Loja

# Universidad Nacional De Loja

Facultad de la Energía, las Industrias y los Recursos

Naturales no Renovables

Carrera de Computación

Desarrollo de una herramienta web para la gestión de requisitos de software en  
la Carrera Computación de la Universidad Nacional de Loja

Development of a web tool for software requirements management in the  
Computer Science program at the National University of Loja.

Trabajo de Integración Curricular  
previo a la obtención del título de  
Ingeniero en Ciencias de la  
Computación.

**AUTOR:**

Edinson Roberto Quizhpe Vizcaíno

**DIRECTOR:**

Ing. Wilman Patricio Chamba Zaragocín

Loja - Ecuador

2024

## Certificación

Loja, 04 de octubre de 2024

Ing. Wilman Patricio Chamba Zaragocín

**DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR**

### **CERTIFICO:**

Que he revisado y orientado el proceso de la elaboración del Trabajo de Titulación denominado: **Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la Universidad Nacional de Loja**, previo a la obtención del título de **Ingeniero en Ciencias de la Computación**, de la autoría del estudiante **Edinson Roberto Quizhpe Vizcaíno**, con **cédula de identidad Nro. 1150299483**, una vez que el trabajo cumple con todos los requisitos exigidos por la Universidad Nacional de Loja para el efecto, autorizo la presentación para la respectiva sustentación y defensa.

Ing. Wilman Patricio Chamba Zaragocín

**DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR**

## **Autoría**

Yo, **Edinson Roberto Quizhpe Vizcaíno**, declaro ser autor del Trabajo de Titulación y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos, de posibles reclamos y acciones legales, por el contenido del mismo. Adicionalmente acepto y autorizo a la Universidad Nacional de Loja la publicación de mi Trabajo de Titulación, en el Repositorio Digital Institucional - Biblioteca Virtual.

**Firma:**

**Cédula:** 1150299483

**Fecha:** 04 de octubre de 2024

**Correo electrónico:** edinson.quizhpe@unl.edu.ec

**Celular:** 0979360585

**Carta de autorización por parte del autor para la consulta, reproducción parcial o total y/o publicación electrónica del texto completo, del Trabajo de Titulación.**

Yo, **Edinson Roberto Quizhpe Vizcaíno**, declaro ser autor del Trabajo de Titulación denominado **Desarrollo de una herramienta para la gestión de requisitos de software, para el laboratorio de software de la Carrera de Computación de la Universidad Nacional de Loja** como requisito para optar el título de **Ingeniero en Ciencias de la Computación**, autorizo al sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Institucional.

Los usuarios pueden consultar el contenido de este trabajo en el Repositorio Institucional, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia del trabajo de integración curricular que realice un tercero.

Para constancia de esta autorización, suscribo, en la ciudad de Loja, a los cuatro días del mes de octubre de dos mil veinticuatro.

**Firma:**

**Autor:** Edinson Roberto Quizhpe Vizcaíno

**Cédula:** 1150299483

**Dirección:** Loja, Av. Eugenio Espejo e Irlanda

**Correo electrónico:** edinson.quizhpe@unl.edu.ec

**Celular:** 0979360585

**DATOS COMPLEMENTARIOS:**

**Director del Trabajo de Integración Curricular:** Ing. Wilman Patricio Chamba Zaragocín

## **Dedicatoria**

Dedico mi trabajo de integración curricular, primeramente, a Dios, por la sabiduría que me ha sabido brindar, y por permitirme cumplir con esta meta tan importante en mi vida, y ser mi guía en los buenos y malos momentos.

A mis amados padres, José y Ana, y a mis queridos hermanos, quienes han sido mi apoyo y motivación en todo momento durante cada paso que he dado. Su esfuerzo y sacrificio me han servido para cumplir con cada meta y objetivos que me he propuesto.

A mis queridos amigos, docentes y familiares, por el apoyo y respaldo incondicional que me han dado en esta etapa.

*Edinson Roberto Quizhpe Vizcaíno*

## **Agradecimiento**

Quiero expresar mi más profundo agradecimiento a mi querida familia, por el ejemplo que me han sabido brindar, por cada consejo y motivación que me han servido para no desmayar, sino al contrario ser más fuerte y valiente en cada momento.

A cada uno de mis docentes, por las enseñanzas y el conocimiento compartido, sin duda me han servido para ser un mejor profesional y crecer académicamente. Quiero resaltar, a mi director de titulación, Ing. Wilman Chamba, por su confianza, apoyo, y consejos que me han servido para mejorar y poder culminar el presente trabajo de titulación.

Finalmente agradezco a cada uno de mis amigos y familiares, así como a mi familia espiritual Jumavi.

***Edinson Roberto Quizhpe Vizcaíno***

## Índice de Contenidos:

<b>Portada</b> .....	<b>i</b>
<b>Certificación</b> .....	<b>ii</b>
<b>Autoría</b> .....	<b>iii</b>
<b>Carta de autorización</b> .....	<b>iv</b>
<b>Dedicatoria</b> .....	<b>v</b>
<b>Agradecimiento</b> .....	<b>vi</b>
<b>Índice de Contenidos:</b> .....	<b>vii</b>
<b>Índice de Tablas:</b> .....	<b>x</b>
<b>Índice de Figuras:</b> .....	<b>xi</b>
<b>Índice de Anexos:</b> .....	<b>xiii</b>
<b>1. Título</b> .....	<b>1</b>
<b>2. Resumen</b> .....	<b>2</b>
<b>3. Introducción</b> .....	<b>4</b>
<b>4. Marco teórico</b> .....	<b>6</b>
4.1 Antecedentes .....	6
4.2 Fundamentos teóricos .....	6
4.2.1 Ingeniería de Software .....	6
4.2.2 Ingeniería de Requisitos.....	8
4.2.3 Estándares y marcos de trabajo .....	10
4.3 Herramientas existentes para la gestión de requisitos.....	11
4.3.1 Case complete.....	11
4.3.2 Visure Requirements.....	11
4.3.3 Reqtest .....	12
4.4 Experiencia de usuario .....	12
4.4.1 Usabilidad.....	12
4.4.2 Metodología SUS.....	13
4.5 Tecnologías .....	16
4.5.1 JavaScript .....	16
4.5.2 NextJS .....	16
4.5.3 NodeJS.....	16

4.5.4 MongoDB .....	17
4.5.5 Arquitectura de contenedores.....	17
4.6 Metodología ICONIX .....	18
4.6.1 Fases .....	19
4.7 Trabajos relacionados .....	22
4.7.1 Software process improvement: Requirements management, verification and validation [5] .....	22
4.7.2 Systematic Literature Review on Requirement Management Tools [46] .....	23
4.7.3 Development and management requirements: Results of a literature review   Desarrollo y gestión de requisitos: Resultados de una revision de la literature [2] .....	23
4.7.4 Towards a Context-Based Framework for the Software Requirements Elicitation Process Management   Hacia un Marco basado en Contexto para la Gestión del Proceso de Educación de Requisitos Software [14] .....	23
<b>5. Metodología.....</b>	<b>24</b>
5.1. Área de estudio .....	24
5.2. Procedimiento .....	24
5.2.1. Objetivo 1. Construir la herramienta web para la gestión de requisitos de software basado en el modelado de software aplicando la metodología ICONIX. ....	25
5.2.2. Objetivo 2. Evaluar la aplicación web desarrollada para determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema). ....	27
5.3. Recursos.....	27
5.3.1. Recursos Científicos .....	27
5.3.2. Recursos Técnicos .....	28
5.3.3. Recursos de Hardware y Software .....	28
<b>6. Resultados .....</b>	<b>30</b>
6.1 Objetivo 1: Construir la herramienta web para la gestión de requisitos de software basado en el modelado de software aplicando la metodología ICONIX. ....	30
6.1.1 Modelado de requisitos de software.....	30
6.1.2 Modelo de dominio de negocio.....	36
6.1.3 Modelo de dominio – usuarios.....	37
6.1.4 Arquitectura de software .....	38
6.1.5 Codificación.....	49

6.2 Objetivo 2. Evaluar la aplicación web desarrollada para determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema). .....	53
6.2.1 Diseño de cuestionarios - encuestas: .....	53
6.2.2 Ejecución de las pruebas: .....	54
<b>7. Discusión .....</b>	<b>63</b>
7.1 Objetivo 1: Construir la herramienta web para la gestión de requisitos de software basado en el modelado de software aplicando la metodología ICONIX. ....	63
7.2 Objetivo 2. Evaluar la aplicación web desarrollada para determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema). .....	65
<b>8. Conclusiones .....</b>	<b>66</b>
<b>9. Recomendaciones .....</b>	<b>67</b>
9.1 Trabajos Futuros .....	67
<b>10. Bibliografía .....</b>	<b>68</b>
<b>11. Anexos .....</b>	<b>73</b>

## Índice de Tablas:

<b>Tabla 1.</b> Fases y entregables de la metodología ICONIX. ....	18
<b>Tabla 2.</b> Requisitos funcionales.....	30
<b>Tabla 3.</b> Requisitos no funcionales.....	35
<b>Tabla 4.</b> Caso de uso - Inicio de sesión.....	40
<b>Tabla 5.</b> Caso de uso – crear proyecto.....	41
<b>Tabla 6.</b> Resultado de ejecución de pruebas unitarias.....	51
<b>Tabla 7.</b> Preguntas de la metodología SUS.....	53
<b>Tabla 8.</b> Definición de niveles de puntuación.....	54
<b>Tabla 9.</b> Definición de media de calificación .....	55
<b>Tabla 10.</b> Resultados obtenidos por cada pregunta.....	55
<b>Tabla 11</b> Resultados obtenidos de los 31 encuestados.....	59
<b>Tabla 12.</b> Resultados de la evaluación.....	61

## Índice de Figuras:

<b>Figura 1.</b> Capas de la ingeniería de software .....	7
<b>Figura 2.</b> Case complete software .....	11
<b>Figura 3.</b> Visure software .....	11
<b>Figura 4.</b> Reqtest software.....	12
<b>Figura 5.</b> Cuestionario SUS empleado [25].....	15
<b>Figura 6.</b> Resultados SUS promedio.....	16
<b>Figura 7.</b> Docker container.....	17
<b>Figura 8.</b> Fase de análisis de requisitos. ....	19
<b>Figura 9.</b> Descripción de un caso de uso. ....	20
<b>Figura 10.</b> Diagrama de robustez - consulta de hospital. ....	21
<b>Figura 11.</b> Proceso de implementación según ICONIX.....	22
<b>Figura 12.</b> Mejoras en el proceso de gestión de requisitos [5].....	22
<b>Figura 13.</b> Croquis de la Carrera Computación de la UNL.....	24
<b>Figura 14.</b> Modelo de dominio del negocio.....	37
<b>Figura 15.</b> Modelo de dominio - usuarios.....	37
<b>Figura 16.</b> Diagrama de componentes de la aplicación.....	38
<b>Figura 17.</b> Diagrama de despliegue de la aplicación. ....	39
<b>Figura 18.</b> Diagrama de casos de uso del sistema.....	40
<b>Figura 19.</b> Prototipado de inicio de sesión.....	47
<b>Figura 20.</b> Prototipado de registro de usuario.....	47
<b>Figura 21.</b> Prototipado de creación del proyecto. ....	48
<b>Figura 22.</b> Prototipado de agregar colaborador.....	48
<b>Figura 23.</b> Prototipado de crear requisito. ....	49
<b>Figura 24.</b> Estructura del backend.....	50
<b>Figura 25.</b> Estructura del frontend.....	50
<b>Figura 26.</b> Resultados obtenidos P1 - P5.....	58

<b>Figura 27.</b> Resultados obtenidos P6-P10.....	59
<b>Figura 28.</b> Puntajes obtenidos por cada evaluador.....	61
<b>Figura 29.</b> Resultados de la pregunta 3 de usabilidad.....	62

**Índice de Anexos:**

<b>Anexo 1. Encuesta para la situación problemática.....</b>	<b>73</b>
<b>Anexo 2. Documento de especificación de requisitos (ERS) .....</b>	<b>91</b>
<b>Anexo 3. Encuesta de experiencia del usuario (SUS). .....</b>	<b>132</b>
<b>Anexo 4. Resultados de las pruebas de aceptación aplicando la metodología SUS. .....</b>	<b>135</b>
<b>Anexo 5. Cálculo de Score en base a la encuesta aplicada.....</b>	<b>140</b>
<b>Anexo 6. Informe de pruebas unitarias .....</b>	<b>142</b>
<b>Anexo 7. Revisión de antiplagio .....</b>	<b>159</b>
<b>Anexo 8. Certificado de traducción del resumen.....</b>	<b>160</b>

## **1. Título**

**Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la Universidad Nacional de Loja**

**Development of a web tool for software requirements management in the Computer Science program at the National University of Loja.**

## 2. Resumen

La gestión de requisitos es una etapa crucial en el desarrollo de proyectos de software, ya que asegura que las necesidades de los clientes se cubran adecuadamente. En este contexto, ante la falta de herramientas específicas, gratuitas y adecuadas para llevar a cabo este proceso, se planteó la siguiente pregunta de investigación: “¿Cuál sería el grado de aceptación de una herramienta con entorno web para los integrantes de un equipo de desarrollo de software, con el rol de analistas de negocios, que les permita gestionar y validar los requisitos en los proyectos de software de la Carrera Computación de la UNL?”. Para ello, el presente Trabajo de Integración Curricular (TIC), tiene como objetivo principal el desarrollo de una herramienta para la gestión de requisitos en los proyectos de software; para el cumplimiento de este objetivo, se empleó la metodología ICONIX, estableciendo la arquitectura lógica y física de la aplicación, en base a los requisitos que se identificaron por medio de las encuestas realizadas. Posteriormente, se llevó a cabo la implementación de la aplicación, empleando una arquitectura cliente servidor, para lo cual se empleó Nodejs, Express y MongoDB para la parte del servidor; y para la parte del cliente se manejaron las tecnologías NextJS, TypeScript y PrimeReact, dando como resultado una solución que cumpliera con los requisitos identificados en la fase de análisis de requisitos de la metodología ICONIX. Finalmente, se realizaron las pruebas para determinar el grado de aceptación de la aplicación web, para ello se empleó la metodología SUS, la cual permitió dar respuesta a la pregunta de investigación planteada, obteniendo como resultado un promedio de 83,1 en la escala de SUS, lo que califica a la aplicación web como un producto altamente aceptable, que cumple con las necesidades de los usuarios.

**Palabras claves:** *ICONIX, SUS, gestión de requisitos, arquitectura cliente – servidor, IEEE830, ISO/IEC 29148*

## **Abstract**

Requirements management is a crucial stage in the development of software projects, since it ensures that the clients' needs are adequately met. In this context, given the lack of specific, free and appropriate tools to carry out this process, the following research question was raised: "What would be the degree of acceptance of a web based tool for the members of a software development team, with the role of business analysts, that allows them to manage and validate requirements in software projects in the Computer Science Degree at the UNL?". To this end, the main objective of this Curricular Integration Work (TIC) is to develop a tool for requirements management in software projects. To achieve this objective, the ICONIX methodology was used, establishing the logical and physical architecture of the application, based on the requirements that were identified through the surveys carried out. Subsequently, the application was implemented, using a client-server architecture, for which Nodejs, Express and MongoDB were used for the server part; and for the client part, NextJS, TypeScript and PrimeReact technologies were used, resulting in a solution that met the requirements identified in the requirements analysis phase of the ICONIX methodology. Finally, tests were carried out to determine the degree of acceptance of the web application, for which the SUS methodology was used, which allowed to answer the research question posed, obtaining as a result an average of 83.1 on the SUS scale, which qualifies the web application as a highly acceptable product, which meets the needs of users.

**Keywords:** *ICONIX, SUS, requirements management, client-server architecture, IEEE830, ISO/IEC 29148*

### 3. Introducción

La gestión de requisitos de software es el proceso mediante el cual se identifican los elementos que se requieren para el desarrollo de un proyecto de software, siendo una de las etapas más importantes para el desarrollo de un proyecto de software [8]. Sin embargo, la falta de herramientas específicas, gratuitas y adecuadas representa una problemática en el proceso de gestión de requisitos de software. Si bien existen herramientas que permiten la gestión de un proyecto de software, como es el caso de reqTest [16], un software que, si bien realiza todo el proceso de desarrollo y gestión de un proyecto, cuenta con la limitante de ser una herramienta paga, por lo que no es accesible para todos los usuarios. Otra herramienta que se enfoca en la gestión de requisitos es Visure [7], la cual brinda una solución colaborativa, enfocada en la trazabilidad de los requisitos, pero con una limitante en la cantidad de proyectos y funcionalidades que permite hacer uso, por la misma razón de ser de pago.

Bajo este contexto, se hace necesario implementar una solución que permita mejorar el proceso de gestión de los requisitos de un proyecto de software, ayudando a los usuarios en el proceso de elicitación de los requisitos. Por lo cual, se planteó la siguiente pregunta de investigación: “¿Cuál sería el grado de aceptación de una herramienta con entorno web para los integrantes de un equipo de desarrollo de software, con el rol de analistas de negocios, que les permita gestionar y validar los requisitos en los proyectos de software de la Carrera Computación de la UNL?”.

Posteriormente, en base al análisis de los trabajos relacionados, se lograron identificar las necesidades que se podían implementar, las cuales facilitarían el proceso de gestión y validación de los requisitos en un proyecto de software. Para ello, se planteó el siguiente objetivo general: “*Determinar el grado de aceptación de la herramienta web para la gestión de requisitos de software en la Carrera de Computación de la Universidad Nacional de Loja*”. Y como objetivos específicos que permitieran alcanzar el cumplimiento del objetivo general, los siguientes: “*Construir la herramienta web para la gestión de requisitos de software basado en el modelado de software aplicando la metodología ICONIX*” y “*Evaluar la aplicación web desarrollada para determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema)*.”

En cuanto al desarrollo del trabajo de integración curricular, se encuentra organizado de la siguiente manera:

En la sección **Marco teórico**, se encuentran los antecedentes referentes al tema de investigación, así como los fundamentos teóricos y los trabajos relacionados; en la sección **Metodología**, se especifica el contexto de desarrollo del TIC, así como los procedimientos y técnicas que se emplearon en la metodología ICONIX; la sección **Resultados**, contiene el desarrollo de los objetivos específicos que se plantearon para el cumplimiento del TIC, donde se puede evidenciar: el modelado de los requisitos de software, modelo de dominio de negocio, diseño de la arquitectura de software, la codificación de la aplicación, la planificación y ejecución de las pruebas SUS, teniendo como resultado del objetivo 1 la implementación de la aplicación, y como resultado del objetivo 2, la evaluación para determinar el grado de aceptación de la aplicación web, de la que se obtuvo un promedio de 83,1 en la escala de la metodología SUS; en la sección **Discusión**, se especifica un análisis de los resultados obtenidos en cada objetivo específico, en el objetivo 1 se obtuvo la construcción de la arquitectura cliente-servidor, que resultó favorable para la construcción de la aplicación; en el objetivo 2 se pudo determinar que en base a los resultados obtenidos, la aplicación web se considera un producto altamente aceptable; la sección **Conclusiones**, resalta los aspectos más importantes y relevantes del desarrollo del TIC; finalmente en la sección **Recomendaciones**, se encuentran las sugerencias y posibles trabajos futuros que se pueden llevar a cabo a partir de esta investigación.

## **4. Marco teórico**

### **4.1 Antecedentes**

La gestión de requisitos de software constituye un eje fundamental para el desarrollo exitoso de un proyecto, ya que garantiza que las necesidades del cliente se especifiquen de manera correcta, sin ambigüedades ni requisitos mal definidos. Jayatilleke y Lei [1], señalan que una mala gestión de los requisitos desemboca en uno de los factores principales para el fracaso de un proyecto, además indican que se debe tener presente la existencia de cambios en los requisitos, esto debido a cambios presentados por parte de los clientes o en las reglas de negocio.

En la Carrera Computación de la UNL, se ha podido identificar la necesidad de mejorar este proceso de gestión, mismo que actualmente es llevado a cabo de manera manual, debido a la falta de herramientas específicas y de fácil uso; en este contexto el presente trabajo de integración curricular plantea como solución el desarrollo de una herramienta web para la gestión de requisitos de software.

### **4.2 Fundamentos teóricos**

#### ***4.2.1 Ingeniería de Software***

Según Pressman, la ingeniería de software es un enfoque adoptado de los procesos que se llevan a cabo para elaborar un software, incluyendo las tecnologías que emplea, en las que se incluye los métodos técnicos e instrumentos automatizados [2]. La ingeniería de software consta de varias capas, de las cuales se deben regir todo tipo de software, independientemente de las necesidades y objetivos del producto [3], estas son: Procesos, métodos y herramientas, todas ligadas al compromiso de una integridad en la calidad del producto. En la figura 1 se puede apreciar la organización de estas capas.



**Figura 1.** Capas de la ingeniería de software

De igual forma, las fases que conforman la ingeniería de software son 5: planificación, análisis, diseño, desarrollo, pruebas y mantenimiento [3].

- *Fase de planificación:* Se encarga de identificar qué información se va a procesar, cuál será su uso y cuál es el rendimiento que se espera [4]. Las actividades correspondientes a esta fase son:
  1. Ingeniería de Sistemas de Información.
  2. Análisis de requisitos.
  3. Planificación del proyecto de software.
- *Fase de desarrollo:* Se establece el diseño de la estructura de datos, la forma en que se implementará, y cómo se implementará. Las actividades que involucran esta fase son:
  1. Diseño del software.
  2. Codificación del software.
  3. Pruebas del software.
- *Fase de mantenimiento:* Está enfocada en los cambios, corrección de errores, y modificaciones que se realizan a partir de los requisitos que cambian por parte del cliente. Los cambios se clasifican en cuatro tipos:

1. Corrección de errores.
2. Adaptación de cambios.
3. Mejoras de requisitos.
4. Prevención de errores.

#### **4.2.2 Ingeniería de Requisitos**

La ingeniería de requisitos (IR) abarca las fases esenciales del desarrollo de soluciones de software y se enfoca en encontrar, analizar, redactar y validar los servicios y restricciones del sistema. Es un eje fundamental para el éxito de un proyecto, ya que definir los requisitos correctamente, garantizará un desarrollo exitoso del resto de etapas de desarrollo del proyecto [5]. Un estudio realizado por PMI (Project Management Institute) señala que el 47% de los proyectos llegan a fracasar como resultado de una mala gestión de los requisitos, esto se debe principalmente a la falta de procesos adecuados para la validación de requisitos; entre las habilidades más importantes para la gestión de requisitos, señala: buena comunicación, capacidad de atender a las ambigüedades y entender la complejidad de las reglas del negocio [6].

Dentro de la IR se considera dos tipos de requisitos para cualquier sistema: Funcionales y no funcionales [7].

- Funcionales: Disponen que funciones realiza el sistema, sin tener en cuenta las condiciones físicas del mismo.
- No funcionales: Establecen las características y restricciones para la implementación del sistema, considerando una serie de atributos de calidad.

##### **4.2.2.1 Características de los requisitos**

Para que un requisito se encuentre correctamente definido, debe cumplir con las siguientes características: correcto, consistente, realista y verificable [8].

- Correcto: Los requisitos deben ser apropiados, y poder ser validados tanto por el cliente como por el desarrollador, para evitar malas interpretaciones en las etapas críticas posteriores.
- Consistente: Los requisitos propuestos deben ser capaces de cumplirse de manera simultánea con otros requisitos.
- Realista: Todos los requisitos preliminares deben ser revisados para asegurarse de la factibilidad de su implementación.

- Verificable: Los requisitos deben ser susceptibles de pruebas que garanticen y evidencien su correcto cumplimiento.

#### **4.2.2.2 Proceso de gestión de los requisitos**

La gestión de requisitos comprende un conjunto de técnicas y procesos que permiten recopilar los elementos que se necesitan para la elaboración de un proyecto de software, por ende, es un proceso primordial para el desarrollo exitoso de un proyecto [8]. Dicho proceso no se lleva a cabo en una etapa en específica del desarrollo del proyecto, sino durante todo el proceso de planificación, control y cambios en el proyecto, esto debido a que existen diversos tipos de proyectos y la gestión de los requisitos cambian constantemente [7].

Las fases que comprende la gestión de requisitos son: Educción de requisitos, Análisis de requisitos, Especificación de requisitos y validación de requisitos.

- Educción: Consiste en la identificación y recopilación de los requisitos necesarios para satisfacer las necesidades del cliente por medio de un producto de software, esta etapa forma parte del proceso de elicitación de requisitos. Entre las técnicas que se emplean en esta fase, se encuentran: Brainstorming, prototipado, entrevistas, observación y análisis de tareas, casos de uso [9].
- Análisis de requisitos: La siguiente fase en el proceso de gestión de requisitos, corresponde al análisis de cada uno de los requisitos propuestos, con el objetivo de poder identificar posibles errores en los mismos, así como para establecer de qué forma se va a implementar cada requisito en el sistema, se realiza además una clasificación de los requisitos en: funcionalidad, prioridad, coste de implementación [6].
- Especificación de requisitos: Durante esta etapa se realiza la elaboración del ERS (Especificación de requisitos de Software), el cual detalla los requisitos analizados y clasificados por el desarrollador y el cliente. Existen varios estándares para la elaboración del ERS, como el IEEE 830, ISO/IEEE 29148, etc [8].
- Validación de requisitos: Dentro de esta etapa se realiza una revisión del ERS, generalmente por parte de todo el equipo de trabajo, incluyendo a los usuarios se reúne para su revisión. Entre las técnicas que se emplean para la validación de requisitos, se encuentran: listas de comprobación, prototipados, validación de modelos, entre otros [9].

### **4.2.3 Estándares y marcos de trabajo**

Para lograr que el producto de software desarrollado cumpla con las necesidades del cliente, se debe elaborar una correcta especificación de los requisitos de software. Para ello, se han establecido diversos estándares y marcos de trabajo referentes a la gestión de requisitos de software [10], entre los más conocidos se encuentran el estándar IEEE 830 y IEEE 29148, así mismo existen otros formatos que en base a la experiencia y tipo de proyectos se han creado y se pueden usar.

#### ***4.2.3.1 Estándar IEEE 830***

El IEEE 830 es un estándar de documentación de los requisitos de software. Fue desarrollado por el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) y brinda un enfoque estructurado para documentar los requisitos de software de un sistema [11]. Dicha especificación, es fundamental para el desarrollo de software, ya que sirve como base para la comunicación efectiva entre los stakeholders, como clientes, desarrolladores y equipos de pruebas.

La estructura que establece el estándar IEEE 830 es altamente recomendada y empleada para la elaboración del ERS [12]. La estructura es la siguiente:

- Introducción
- Descripción General
- Requisitos Específicos
- Apéndices
- Anexos

#### ***4.2.3.2 Estándar IEEE 29148:2018***

El estándar IEEE 29148, es un estándar para la especificación de requisitos para sistemas y software. Este estándar fue desarrollado por el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE), como una actualización al estándar IEEE 830, y brinda una guía sobre cómo gestionar los requisitos de software durante el ciclo de vida del desarrollo de software SDLC [10]. Este estándar detalla los procesos necesarios y que se implementarán en la IR, y establece los ítems de información solicitados que se elaborarán mediante el desarrollo de esos procesos.

Facilita, además, el proceso que se lleva a cabo en el SDLC, ya que cuenta con conceptos del esquema y sugerencias para realizar una especificación correcta, y aporta de manera positiva en la planificación del desarrollo de software, debido a que se tiene en consideración lo requerido y esperado por el cliente [13].

### 4.3 Herramientas existentes para la gestión de requisitos

Existen diversas herramientas que ayudan a la gestión de requisitos de software, las cuales permiten automatizar los procesos que corresponden a la IR y realizar un análisis más completo del problema al que se le va a dar solución [7]; por lo general, en la práctica, los desarrolladores realizan este proceso con herramientas tradicionales como Word, Excel o Access, esto se debe principalmente a la gran cantidad de herramientas que existen actualmente, lo que provoca que no se tenga claro y definido que herramienta usar [14].

#### 4.3.1 Case complete



**Figura 2.** Case complete software

Esta herramienta permite una alta trazabilidad entre los requerimientos, su uso es de fácil manejo, así como también cuenta con una interfaz amigable. Se encuentra disponible solo para Windows [15]. Algunas de sus características principales son:

- Cuenta con una matriz de trazabilidad entre los distintos tipos de requisitos.
- Generación de plan de pruebas en base a los casos de uso.

#### 4.3.2 Visure Requirements



**Figura 3.** Visure software

Es una de las herramientas más populares para la gestión de requisitos, ya que ofrece una plataforma colaborativa de manera integral, permitiendo una trazabilidad completa, además se integra con Microsoft Word, Microsoft Excel, entre otros [7]. Como características principales cuenta con la integración de gestión de riesgos, manejo de pruebas, seguimiento de

errores, evaluación de requisitos, análisis de requisitos, entre otros. Cuenta con varias plantillas que se basan en los distintos estándares existentes.

### 4.3.3 Reqtest



**Figura 4.** Reqtest software

Reqtest se ha convertido en la herramienta líder en los países nórdicos, para el desarrollo de proyectos, enfocada en la gestión de requisitos, pruebas y control de errores. Dentro de sus características principales, se encuentra el manejo de pruebas, ya que permite crear casos de uso, en base a los requisitos creados, y posterior a ello, realizar pruebas, buscando asegurar que los requisitos establecidos garanticen la calidad en el producto de software [16].

## 4.4 Experiencia de usuario

La experiencia de usuario (UX) se define como los criterios sobre la facilidad de uso, accesibilidad y satisfacción por parte los usuarios. Se mide en diferentes aspectos, como la experiencia de navegación, experiencia de usabilidad y experiencia sensorial [17]. Entre los elementos que conforman UX, están:

- Usabilidad: La facilidad del usuario para emplear un producto/software, para ello se incluyen herramientas que ayuden a que el software sea bastante intuitivo.
- Accesibilidad: Se incluyen aspecto como la inclusión (visual, cognitiva, motriz, auditiva) y diseño responsivo (laptop, Tablet, móvil, etc.).
- Funcionalidad: Se centra en el desempeño a nivel técnico del producto o software.
- Innovación: Se refiere a la capacidad de dar solución a las necesidades del cliente mediante un producto de software.

### 4.4.1 Usabilidad

La usabilidad es un atributo de calidad, el cual hace referencia a la facilidad de uso de un producto, el cual se enfoca en un propósito específico, ya que varía dependiendo del tipo de audiencia (usuarios finales) que harán uso del producto [18]. Se define también como el grado en que un usuario puede aprovechar el uso de un producto, en medida con el beneficio que el producto ofrezca al usuario.

Otra definición, de acuerdo a la ISO/IEC 9126 [19], es que usabilidad es un atributo de la calidad de software, dentro de un conjunto de atributos de software, los que se fundamentan en el esfuerzo requerido para el uso y la valoración de manera individual de un producto de software [20].

#### **4.4.1.1 Métricas de Evaluación**

La evaluación de usabilidad se considera la etapa más importante del diseño centrado en el usuario, el cual se puede realizar mediante distintos métodos, métricas y técnicas, algunas de estas son: evaluación heurística, test con usuarios finales, metodologías de evaluación, entre otros. [21].

- Evaluación heurística: Corresponde a un método de inspección, el cual resalta la rapidez y facilidad con la que se puede realizar.
- Test con usuarios finales: Se centra en la observación y análisis de los usuarios con el producto de software, identificando los problemas que se puedan presentar y poder darles una solución posteriormente.
- Metodologías de evaluación: Existen diversas metodologías de evaluación para la usabilidad, estas permiten una mejor percepción de la satisfacción del usuario con el producto. Algunas de estas son: cuestionario de usabilidad del software, escala de usabilidad del software, entre otras.

#### ***4.4.2 Metodología SUS***

La metodología SUS (System Usability Scale) se enfoca en evaluar la usabilidad de un sistema o producto, la cual tiene como evaluadores a los usuarios finales, está compuesta por 10 ítems que fueron definidos a partir de las definiciones de usabilidad de la ISO 9241-11 [23], fue desarrollado por Brooke en 1986 [22]. La definición de usabilidad planteada en la ISO 9241-11 señala que es el *“Grado en el que un sistema o producto puede ser empleado por usuarios específicos para alcanzar objetivos determinados con eficacia, eficiencia y satisfacción en un contexto determinado de uso”*.

Para el desarrollo de las pruebas SUS, se requiere de un producto de software terminado; estas pruebas garantizan resultados confiables y acertados, por lo que es uno de los métodos de medición de usabilidad más utilizados en la experiencia de usuario. Los involucrados en el desarrollo de las pruebas SUS son: Desarrolladores, Probadores y/o Usuarios finales [22].

Dentro de las partes que conlleva el proceso de evaluación en la metodología SUS se encuentran:

- Escala de medición
- Medición/evaluación
- Resultados

#### ***4.4.2.1 Escala de medición***

La evaluación con la metodología SUS, se realiza en un cuestionario de 10 preguntas, las cuales tienen una escala de calificación del 1 al 5, siendo 1 un extremo negativo o “Totalmente en desacuerdo”, y 5 un extremo positivo o “Totalmente de acuerdo”. Las puntuaciones finales de SUS pueden oscilar entre 0 y 100, en donde las puntuaciones más altas indican una mejor usabilidad [24]. Primeramente, se debe presentar el sistema o aplicación a los evaluadores, bajo los criterios que se hayan establecido y luego se procede a la evaluación del mismo.

#### ***4.4.2.2 Medición/Evaluación***

En esta sección se lleva a cabo la evaluación de la aplicación una vez que los usuarios evaluadores han probado la misma, para ello se responde al cuestionario de las 10 preguntas establecidas. Estas preguntas por lo general se aplican en todo sistema, aunque pueden ser modificadas según el objetivo que se quiera obtener y contrastar [25]. Por ejemplo, en un estudio realizado por Bangor y Miller [24], se realizó una modificación a la pregunta 8, reemplazando la palabra “engorroso” por “incómodo”, además se agregaron algunas instrucciones al inicio del cuestionario, para indicar la forma de responder al cuestionario. En la figura 5 se muestra el cuestionario SUS que se empleó.

Please check the box that reflects your immediate response to each statement. Don't think too long about each statement. Make sure you respond to every statement. If you don't know how to respond, simply check box "3."

	Strongly Disagree				Strongly Agree
1. I think that I would like to use this product frequently.	1	2	3	4	5
2. I found the product unnecessarily complex.	1	2	3	4	5
3. I thought the product was easy to use.	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this product.	1	2	3	4	5
5. I found the various functions in the product were well integrated.	1	2	3	4	5
6. I thought there was too much inconsistency in this product.	1	2	3	4	5
7. I imagine that most people would learn to use this product very quickly.	1	2	3	4	5
8. I found the product very awkward to use.	1	2	3	4	5
9. I felt very confident using the product.	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this product.	1	2	3	4	5

**Figura 5.** Cuestionario SUS empleado [25].

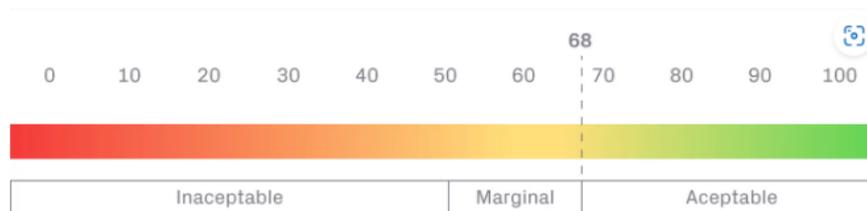
#### 4.4.2.3 Resultados

En esta última parte se contrasta los resultados obtenidos de la evaluación, para ello se sigue un proceso y formula establecida, en la cual se sigue las siguientes condiciones [26]:

- Se debe sumar las respuestas de las preguntas impares (1,3,5,7,9) y luego se le resta 5.
- Se debe sumar las respuestas de las preguntas pares (2,4,6,8,10) y luego se le resta 25.
- Se suma ambos resultados y se lo multiplica por 2,5.

Con el resultado obtenido, se tendrá un valor de 0 a 100 por cada participante, del cual se sacará un promedio por todos los resultados de los participantes, para obtener el promedio

SUS total del sistema. De acuerdo con expertos, como Jeff Sauro [26], un promedio general de SUS se encuentra en un valor de 68/100 (ver figura 6). Además, identifica distintos rangos de usabilidad de acuerdo al puntaje SUS obtenido, partiendo desde la calificación F (muy mala usabilidad) hasta la calificación A (Excelente usabilidad); obtener un puntaje de más de 78,9/100 indica una calificación A.



**Figura 6.** Resultados SUS promedio.

## 4.5 Tecnologías

### 4.5.1 JavaScript

JavaScript es un lenguaje de programación usado de manera general para la creación de aplicaciones web dinámicas. Una aplicación web dinámica es aquella que dispone de características como que texto aparece o no, animaciones, acciones luego de ejecutar una función o botón [27]. De manera técnica, JavaScript es un lenguaje de programación interpretado, ya que no es necesario compilar los programas para poder ejecutarlos [28].

### 4.5.2 NextJS

NextJS es un framework de trabajo para la construcción de aplicaciones web, que incorpora la experiencia del desarrollador con funciones listas para la producción, así como el renderizado por parte del servidor [29], capacidad de TypeScript, agrupación inteligente, precarga de rutas, entre otras. A diferencia de React.js brinda directrices de la estructura de la aplicación, haciendo la enrutación de las páginas por medio de un sistema de carpetas de archivos [30].

### 4.5.3 NodeJS

Nodejs es un entorno de ejecución que se encuentra orientado a acciones asíncronas de JavaScript, diseñado con el propósito de ejecutar aplicaciones de red que sean escalables [31]. Como características principales, Nodejs es que, al trabajar con acciones asíncronas, si no existe tareas por ejecutar, el servidor de Nodejs permanece en estado de suspensión hasta que exista

una conexión o petición hacia el mismo, esto debido a que se usa el lenguaje de programación JavaScript en el servidor [32].

#### **4.5.4 MongoDB**

MongoDB es una base de datos no relacional, diseñada para el rápido desarrollo de aplicaciones web e infraestructura de internet. Su gran ventaja es que el modelo de datos está construido para una alta lectura y escritura de datos, lo que permite una escalabilidad mayor que la que se tiene con una base de datos relacional [33]. La elección de MongoDB como base de datos para el proyecto, es por la escalabilidad que permite de los datos, ya que va a ir creciendo en medida que los datos crezcan, esto resulta muy oportuno cuando se maneja datos en una aplicación que van a ir creciendo gradualmente. El modelo de datos orientado a documentos permite que sea sencillo la distribución de datos a través de diversos servidores, MongoDB redistribuye de manera automática los documentos y operaciones de lectura y escritura [34].

#### **4.5.5 Arquitectura de contenedores**

Se puede definir un contenedor analógicamente como un container que se transporta de un barco a otro, en el cual no importa el contenido del mismo, sino solo la forma en que se almacenó y transporta de un lugar a otro [35]. En el contexto de software, al usar un contenedor se puede probar de manera segura una aplicación sin tener que esperar el comportamiento del código de manera distinta. Los contenedores, además, brindan un entorno aislado dentro del SO del servidor, llamado “Virtualización” [36].

##### **4.4.5.1 Docker**



**Figura 7.** Docker container.

Docker es una plataforma diseñada para la ejecución y distribución de aplicaciones mediante contenedores. Separa la infraestructura de la aplicación, brindando un ambiente aislado para el empaquetamiento y ejecución de la aplicación [35]. La característica principal de Docker frente a los modelos tradicionales, es que Docker emplea contenedores en lugar de

máquinas virtuales, por lo que el código se ejecuta en dicho contenedor de manera aislada de otros contenedores, siendo más ligeros en cuanto a recursos empleados, a diferencia de las máquinas virtuales [37].

#### 4.6 Metodología ICONIX

ICONIX es una metodología pesada/ligera, que tiene un enfoque en el que se lleva un control del ciclo de vida de un proyecto [38], enfocándose más en el análisis y diseño que en la fase de implementación. Esta metodología se conforma de las siguientes fases: análisis de requisitos, análisis y diseño preliminar, diseño, e implementación.

A continuación, en la tabla 1, se muestra un resumen de las fases, actividades y entregables que se emplean en la metodología ICONIX:

**Tabla 1.** Fases y entregables de la metodología ICONIX.

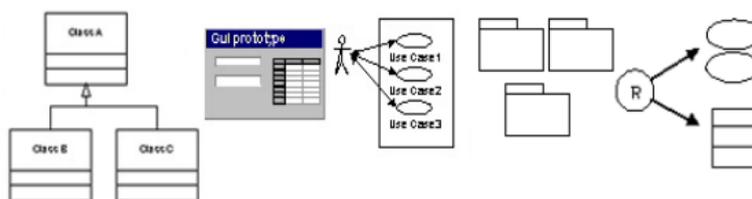
<b>FASE</b>	<b>ACTIVIDAD</b>	<b>ENTREGABLE</b>	<b>MÉTODO/TÉCNICA</b>
Análisis de requisitos	- Identificación de requerimientos funcionales y no funcionales.	- Documento de especificación de requisitos de software (ERS).	- Historias de usuario/entrevista
	- Prototipado de la aplicación	- Modelo de dominio	- Bocetos preliminares
	- Definir casos de uso	- Diagrama de casos de uso general	
Análisis y Diseño Preliminar	- Descripción de los casos de uso	- Modelo de casos de uso	- Análisis de documentación
	- Detalle del comportamiento	- Diagrama de secuencia	- Análisis de documentación
Diseño	- Identificación de relaciones	- Diagrama de clases	- Análisis de documentación
	- Generar modelo de datos	- Diagrama de datos	- Análisis de documentación.

Implementación	- Codificación de la aplicación	- Herramienta web	- Análisis de documentación - Desarrollo de la aplicación
	- Especificación de la arquitectura de la aplicación	- Modelo de la arquitectura de la aplicación (Diagrama de componentes)	- Análisis de documentación
	- Ejecución de pruebas	- Informe de pruebas	- Ejecución de pruebas de caja blanca - Ejecución de pruebas de caja negra

## 4.6.1 Fases

### 4.6.1.1 Análisis de requisitos

Se realiza un levantamiento de los requisitos que deben cumplirse en el sistema, se realiza la especificación de requisitos funcionales y no funcionales, además de la construcción del modelo de dominio, prototipado del producto, y la identificación de los casos de uso [39].



**Figura 8.** Fase de análisis de requisitos.

- Modelado del dominio: Se describe los objetos o clases del producto, los cuales representan el alcance real del proyecto, y sirve como base para identificar los posibles casos de uso; ICONIX recomienda realizar el modelo de dominio una vez que se obtengan los requisitos [40].

### 4.6.1.2 Análisis y diseño preliminar

Se lleva a cabo la descripción de los casos de uso que tendrá el sistema, considerando su flujo principal y su flujo alterno, así como un análisis de robustez para mejorar y actualizar los diagramas de dominio [41].

- Casos de uso: Describe la funcionalidad del sistema, y la forma en que un usuario (actor) interactúa con el mismo [42]. En la figura 9 se muestra la descripción de un caso de uso.

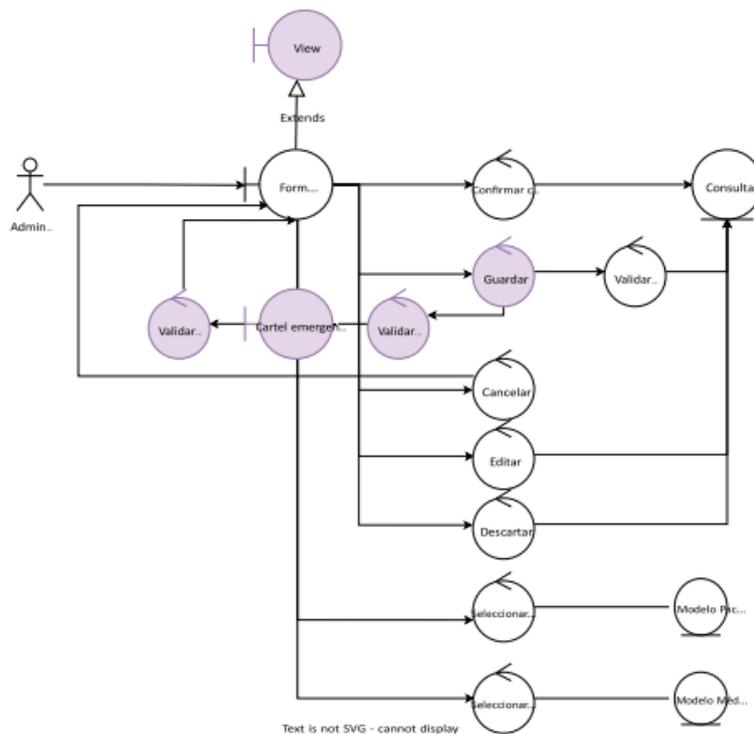
<b>Caso de Uso</b>	<< Nombre del CU >>		<< Identificador >>	
<b>Actores</b>	<< Listado de los actores participantes en el CU >> << Podemos indicar quien es el que inicia el CU usando (I) >>			
<b>Tipo</b>	<< Tipo del caso de uso >> << Primario, Secundario u Opcional   >> <<Esencial o Real >>			
<b>Referencias</b>	<< Indicamos que requisitos se pueden incluir dentro de este CU >>	<< CU que tienen relación con este >>		
<b>Precondición</b>	<< Condiciones sobre el estado del sistema que tienen que ser ciertas para que se pueda realizar el CU >>			
<b>Postcondición</b>	<< Efectos que de forma inmediata tiene la realización del CU sobre el estado del sistema >>			
<b>Autor</b>	<< Esta línea se podría repetir para mantener una historia de cambios del CU >>	<b>Fecha</b>		<b>Versión</b>
<b>Propósito</b>				
<< Descripción general del CU (Suficiente con una línea) >>				
<b>Resumen</b>				
<< Descripción de alto nivel del flujo normal (básico) del caso de uso (Suficiente con un pequeño párrafo) >>				

Figura 9. Descripción de un caso de uso.

### 4.6.1.3 Diseño

Una vez que el diseño preliminar se ha cumplido, se desarrollan los diagramas de secuencia en base a los casos de uso identificados y se define el modelo de datos del sistema, esto permite mostrar el comportamiento del modelo de dominio, así como la integración entre componentes del sistema [43], además de realizar un análisis de robustez.

- Análisis de robustez: Permite identificar los objetos que intervienen en cada caso de uso, en este se describen las funciones de software y los objetos de dominio [44]. En la figura 10 se muestra un ejemplo del diagrama de robustez para el caso de uso de “Consulta en hospital”.



**Figura 10.** Diagrama de robustez - consulta de hospital.

#### 4.6.1.4 Implementación

A partir del diseño elaborado, se lleva a cabo la codificación del sistema, así como de las pruebas establecidas anteriormente [45], en la figura 11 se muestra el proceso de implementación que plantea la metodología ICONIX. Algunas consideraciones importantes para la implementación son:

- La codificación debe partir desde el diseño creado.
- Si durante la codificación se evidencia que existen errores en el diseño, se debe cambiar y revisar el proceso.
- Hacer hincapié en las pruebas, mientras se codifica la aplicación.

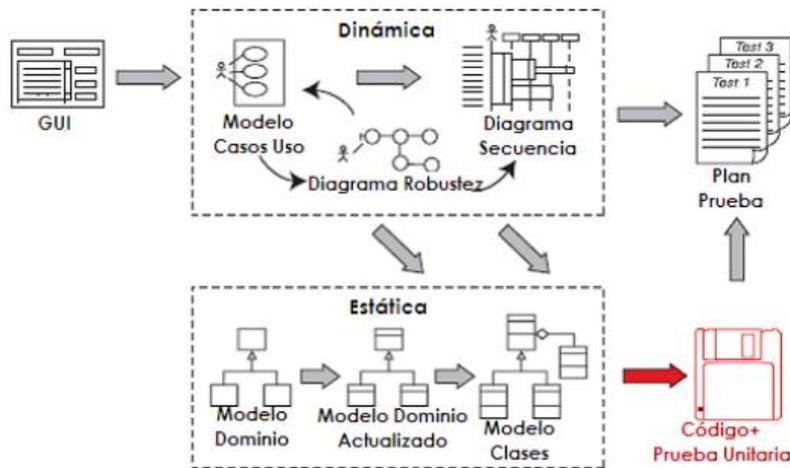


Figura 11. Proceso de implementación según ICONIX.

## 4.7 Trabajos relacionados

### 4.7.1 Software process improvement: Requirements management, verification and validation [5]

Este artículo presenta un estudio de caso de uso de las mejores prácticas CMMI para una organización que desea mejorar sus procesos de desarrollo de software. El propósito era mejorar el área de proceso Gestión de Requisitos (REQM), apoyando el nivel de cumplimiento de las áreas de proceso Verificación (VER) y Validación (VAL) [5]; como resultado se logró mejorar el indicador del nivel de cumplimiento sobre las prácticas llevadas a cabo para la gestión de requisitos, como lo muestra la figura 12.

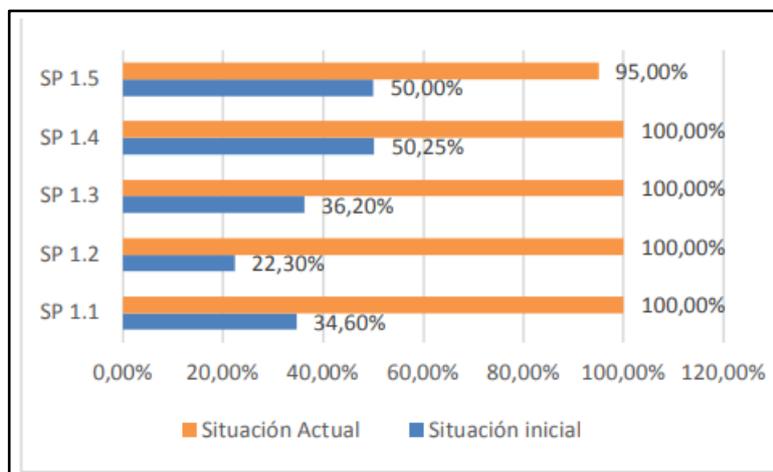


Figura 12. Mejoras en el proceso de gestión de requisitos [5].

#### **4.7.2 Systematic Literature Review on Requirement Management Tools [46]**

Esta revisión de literatura realiza un análisis de la importancia de las herramientas específicas para la gestión de requisitos, con el objetivo de determinar los beneficios de estas herramientas y sobre el cómo y cuándo se debe emplearlas. Como resultado de la investigación se concluyó que Objectiver es la herramienta más apropiada para la gestión de requisitos, ya que permite a los involucrados tener un panorama mucho más claro del sistema [46].

#### **4.7.3 Development and management requirements: Results of a literature review | Desarrollo y gestión de requisitos: Resultados de una revisión de la literatura [2]**

El artículo presenta una revisión de literatura con el objetivo de identificar las buenas prácticas de los modelos para la gestión de requisitos, Los resultados concluyen que un modelo para gestionar la IR debe presentar buenas prácticas, y potenciarlas con aportaciones de otras áreas del conocimiento [2].

#### **4.7.4 Towards a Context-Based Framework for the Software Requirements Elicitation Process Management | Hacia un Marco basado en Contexto para la Gestión del Proceso de Educción de Requisitos Software [14]**

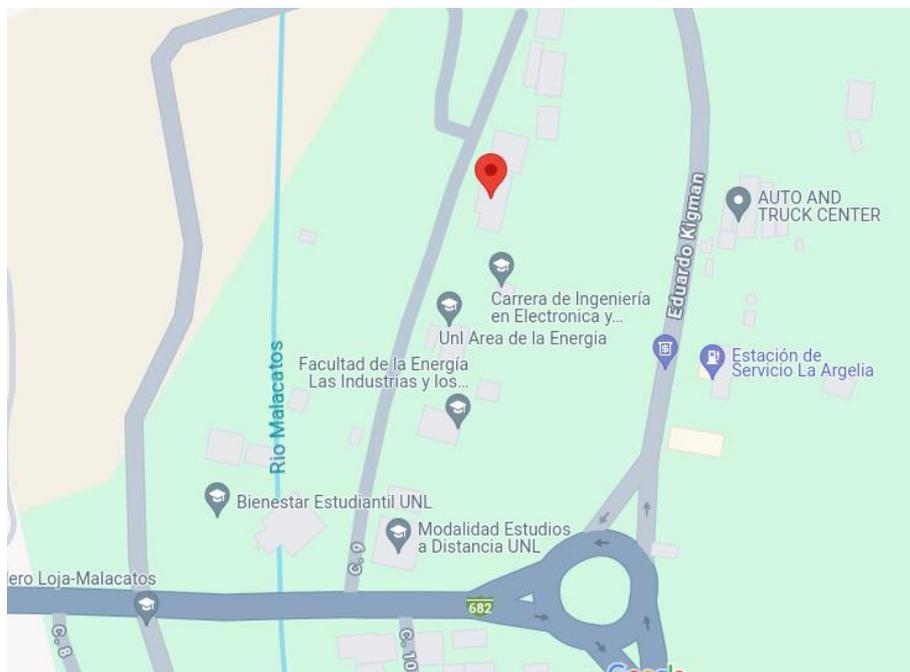
Este estudio propone un marco para el modelamiento del proceso de educación, por medio de dos enfoques, que se centran en los factores que afectan a la educación de los requisitos. Además, esta propuesta busca ser una herramienta práctica para los profesionales en ingeniería de requisitos [14].

## 5. Metodología

En la presente sección se detalla la metodología empleada para el desarrollo del presente trabajo de titulación. En la sección **5.1.** se precisa el área de estudio en que se desarrolló el trabajo de integración curricular, en la sección **5.2** se detalla el procedimiento realizado para cumplir los objetivos planteados, y en la sección **5.3** los recursos que se emplearon.

### 5.1. Área de estudio

El trabajo de integración curricular se realizó en la ciudad de Loja, en la Carrera Computación de la Universidad Nacional de Loja, -4.030879464609493, -79.1995730768181 (ver figura 13). El mismo tiene como objetivo general “Determinar el grado de aceptación de la herramienta web para la gestión de requisitos de software en la Carrera Computación de la Universidad Nacional de Loja”. Así mismo consta de dos objetivos específicos, los cuales se detallan en la sección **5.2.1.** y **5.2.2.**



**Figura 13.** Croquis de la Carrera Computación de la UNL

### 5.2. Procedimiento

El cumplimiento de los objetivos planteados se llevó a cabo con la realización de las actividades que se detallan a continuación:

**5.2.1. Objetivo 1. Construir la herramienta web para la gestión de requisitos de software basado en el modelado de software aplicando la metodología ICONIX.**

- Análisis preliminar y diseño de los requerimientos:
  - La elicitación de los requerimientos se realizó a través de una encuesta aplicada a los estudiantes de la Carrera Computación de la UNL (ver **Anexo 1**), esta encuesta permitió identificar los requisitos funcionales y no funcionales de la herramienta, los cuales se especificaron en el documento ERS (ver sección **6.1.1** y **Anexo 2**).
  - Una vez obtenido el documento de especificación de requisitos, se desarrolló el modelo de dominio de negocio y de usuarios, los que permitieron tener una comprensión de la estructura de la aplicación.
    1. Modelo de dominio de negocio: Este modelo proporcionó una representación de las entidades y su relación dentro del contexto del negocio, lo cual permitió identificar los elementos esenciales y las interacciones en el sistema (ver sección **6.1.2**).
    2. Modelo de usuarios: Este modelo describió la forma en que los usuarios interactuarían con el sistema, acorde a sus roles y permisos, garantizando la seguridad de los usuarios y del sistema (ver sección **6.1.3**).
- Diseño:
  - Se elaboró la arquitectura de la aplicación, la cual se clasificó en: arquitectura física (ver sección **6.1.4.1**) y arquitectura lógica (ver sección **6.1.4.2**). Se realizó además el prototipado de la aplicación con la herramienta Balsamiq (ver sección **6.1.4.3**).

Arquitectura física:

1. Diagrama de componentes: Este diagrama permitió visualizar la relación entre los distintos componentes de software, se incluyeron los módulos, librerías y estructura de la aplicación, tanto en el servidor backend, como en el cliente Frontend, así como la conexión con la base de datos. (ver sección **6.1.4.1.1**)
2. Diagrama de despliegue: Este diagrama mostró la distribución física de los componentes de la aplicación, en él se especificaron las

conexiones entre los nodos y la forma en que interactúan entre sí; esto permitió visualizar cómo la aplicación se ejecutaría en un entorno físico, considerando servidores y base de datos (ver sección **6.1.4.1.2**).

Arquitectura lógica:

1. Diagrama de casos de uso: Este diagrama representó las interacciones entre los actores (usuarios) con el sistema para cada funcionalidad acorde a los requisitos establecidos (ver sección **6.1.4.2.1**). Cada caso de uso describió las acciones que los usuarios pueden realizar y la respuesta por parte del sistema. Esto fue fundamental para garantizar que se cumplan todas las necesidades identificadas por los usuarios.

Prototipado de la aplicación:

Se realizaron los prototipos de la aplicación que permitieron identificar las funcionalidades y validar la interfaz de usuario y la experiencia de usuario, con base a los requisitos obtenidos; para ello se empleó la herramienta Balsamiq (ver sección **6.1.4.3**).

- Implementación:

- La codificación de la herramienta se realizó en el IDE Visual Studio Code, en el que se emplearon las siguientes tecnologías:

Backend:

- Lenguaje de programación: Javascript
- Framework: Nodejs como entorno de ejecución para el servidor backend, y express para la creación de la API.
- Base de datos: MongoDB

Frontend:

- Lenguaje de programación: Typescript
- Framework: Nextjs
- La estructura del proyecto se realizó en base al diagrama de componentes (ver sección **6.1.5.1**), lo que permitió facilitar la organización y mantenimiento del código.

- El código fuente fue alojado en el siguiente repositorio de Github (ver sección **6.1.5.1**): <https://github.com/Computacion-UNL/ers-tool-cc>

### **5.2.2. Objetivo 2. Evaluar la aplicación web desarrollada para determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema).**

La ejecución del segundo objetivo se realizó como parte de la fase de implementación. A continuación, se describen las actividades realizadas para el cumplimiento de este objetivo:

Diseño de cuestionarios y encuestas:

Se planificó la encuesta de aceptación aplicando la metodología SUS (ver **Anexo 3**), la cual constó de diez preguntas predefinidas enfocadas en la usabilidad de la aplicación web.

Ejecución de las pruebas:

Se realizaron las pruebas de aceptación acorde a la metodología SUS (ver sección 6.4), las cuales permitieron determinar el grado de aceptación de la aplicación web mediante la experiencia del usuario (UX).

## **5.3. Recursos**

Para la ejecución del trabajo de integración curricular, se emplearon los siguientes recursos:

### **5.3.1. Recursos Científicos**

#### **5.3.1.1. Casos de estudio**

Se analizaron diferentes herramientas relacionadas con la gestión de requisitos en proyectos de software, para establecer las características principales de cada una y cómo se gestionan y se lleva el control de los requisitos funcionales y no funcionales dentro de un proyecto de software.

#### **5.3.1.2. Investigación bibliográfica**

Se realizó una revisión de trabajos similares, con el fin de tener información más útil sobre las distintas herramientas y soluciones existentes para la gestión de requisitos de software (ver **sección 4.7**).

### **5.3.2. Recursos Técnicos**

#### **5.3.2.1. Entrevista**

Esta entrevista permitió realizar un análisis preliminar de los requisitos funcionales y no funcionales, de esta manera se pudo obtener la especificación de los requisitos de la aplicación siguiendo el formato IEEE 830 (ver Anexo 1).

#### **5.3.2.2. Metodología de desarrollo de Software**

La metodología de desarrollo utilizada para el trabajo de integración curricular es la metodología ICONIX, la cual consta de cuatro fases: Análisis de requisitos, Análisis y Diseño preliminar, Diseño e Implementación; permitiendo así una mayor agilidad en el desarrollo del proyecto. En la **sección 4.5** se especifica de manera más detallada la metodología empleada.

### **5.3.3. Recursos de Hardware y Software**

#### **5.3.3.1. Hardware**

- **Laptop:** Equipo informático necesario para el desarrollo del Trabajo de Integración Curricular, tanto en la parte de la memoria como el desarrollo de la herramienta.

#### **5.3.3.2. Software**

- **Visual Studio Code:** Editor de código empleado para la codificación del backend y frontend de la herramienta.
- **Postman:** Herramienta empleada para realizar pruebas de la API durante el desarrollo de la herramienta.
- **Github Desktop:** Herramienta empleada para la gestión de cambios y commits de el desarrollo de la aplicación.
- **MongoDB Compass:** Interfaz gráfica para la gestión de la base de datos empleada.
- **Microsoft Edge:** Navegador web usado para el despliegue local en la fase de desarrollo de la herramienta.
- **ArgoUML:** Herramienta empleada para la elaboración de los diagramas requeridos en cada fase de la arquitectura de la aplicación.
- **Balsamiq:** Herramienta empleada para el prototipado y bocetos de la aplicación.

#### **5.3.3.2. Tecnologías**

- **MongoDB:** Base de datos no relacional usada para la aplicación.

- **JavaScript:** Lenguaje de programación empleado para la codificación del backend y frontend.
- **Nodejs:** Entorno de ejecución empleado para el desarrollo del backend.
- **Express:** Framework empleado para el desarrollo del servidor web del backend.
- **Nextjs:** Framework empleado para el desarrollo del frontend.

## 6. Resultados

En la presente sección se muestran las actividades y tareas realizadas en cada fase, para el cumplimiento de los objetivos planteados en el Trabajo de Integración Curricular.

### 6.1 Objetivo 1: Construir la herramienta web para la gestión de requisitos de software basado en el modelado de software aplicando la metodología ICONIX.

Para el desarrollo del primer objetivo se llevaron a cabo las siguientes tareas y actividades: modelado de requisitos de software, modelo de domino, diseño de la arquitectura de la aplicación y codificación del backend y frontend en base a lo establecido en cada fase de la metodología ICONIX.

#### 6.1.1 Modelado de requisitos de software

Se aplicó el estándar IEEE 830 para la especificación de requisitos de software, con el cual se pudo definir los requisitos funcionales (RF) y requisitos no funcionales (RFN) de la herramienta de software. En la tabla 2 y tabla 3 se resume los RF y RFN obtenidos, para una revisión completa revisar el anexo 2.

Tabla 2. Requisitos funcionales

ID	Nombre	Descripción	Categoría
		El sistema debe permitir al gestor del proyecto crear un nuevo proyecto, con la siguiente información:	
RF001	Crear proyecto	<ul style="list-style-type: none"><li>• Nombre del proyecto</li><li>• Objetivo general</li><li>• Objetivos específicos</li><li>• Fecha de inicio</li><li>• Fecha de finalización</li><li>• Actores</li></ul>	Visible
		El sistema debe permitir al gestor del proyecto modificar/actualizar la siguiente información del proyecto:	Visible
RF002	Editar información del proyecto	<ul style="list-style-type: none"><li>• Nombre del proyecto</li><li>• Objetivo general</li></ul>	

ID	Nombre	Descripción	Categoría
		<ul style="list-style-type: none"> <li>• Objetivos específicos</li> <li>• Fecha de inicio</li> <li>• Fecha de finalización</li> <li>• Actores</li> </ul>	
<b>RF003</b>	Listar proyectos	El sistema debe permitir al gestor del proyecto visualizar todos los proyectos que ha creado.	Inferido
<b>RF004</b>	Dar de baja proyecto	El sistema debe permitir al gestor del proyecto dar de baja un proyecto que ha creado. Se debe mostrar un mensaje de confirmación para dar de baja el proyecto.	Visible
<b>RF005</b>	Crear actor del proyecto	<p>El sistema debe permitir al gestor del proyecto y al analista de negocios crear un nuevo actor, con la siguiente información:</p> <ul style="list-style-type: none"> <li>• Nombre</li> <li>• Descripción</li> </ul>	Visible
<b>RF006</b>	Actualizar información del actor	<p>El sistema debe permitir al gestor del proyecto y al analista de negocios modificar/actualizar la siguiente información de un actor:</p> <ul style="list-style-type: none"> <li>• Nombre</li> <li>• Descripción</li> </ul>	Visible
<b>RF007</b>	Eliminar actor del proyecto	El sistema debe permitir al gestor del proyecto y al analista de negocios eliminar un actor creado en el proyecto que es responsable.	Visible
<b>RF008</b>	Crear nuevo requisito	<p>El sistema debe permitir al gestor del proyecto y al analista de negocios crear un nuevo requisito para un proyecto en el que sea responsable, con los campos:</p> <ul style="list-style-type: none"> <li>• ID</li> <li>• Nombre del requisito</li> <li>• Descripción</li> <li>• Prioridad</li> <li>• Tipo de requisito</li> <li>• Técnica de elicitación</li> <li>• Actor</li> </ul>	Visible

<b>ID</b>	<b>Nombre</b>	<b>Descripción</b>	<b>Categoría</b>
<b>RF009</b>	Actualizar información de un requisito	<p>El sistema debe permitir al gestor del proyecto y analista de negocios actualizar/modificar la siguiente información de un requisito creado en un proyecto que sea responsable:</p> <ul style="list-style-type: none"> <li>• Nombre del requisito</li> <li>• Descripción</li> <li>• Prioridad</li> <li>• Tipo de requisito</li> <li>• Técnica de elicitación</li> <li>• Actor</li> </ul>	Visible
<b>RF010</b>	Listar requisitos	El sistema debe permitir al gestor del proyecto y analista de negocios, visualizar todos los requisitos creados de cada proyecto que sea responsable.	Inferido
<b>RF011</b>	Dar de baja requisito	El sistema debe permitir al gestor del proyecto y analista de negocios, dar de baja un requisito creado en el proyecto(s) que sea responsable.	Visible
<b>RF012</b>	Generar documento ERS	El sistema debe permitir al gestor del proyecto y analista de negocios, generar un documento de especificación de requisitos (ERS) del proyecto(s) que sea responsable.	Visible
<b>RF013</b>	Listar documentos ERS	El sistema debe permitir al gestor del proyecto y analista de negocio, visualizar un listado de todos los documentos ERS que ha generado de los proyectos que sea responsable.	Inferido
<b>RF014</b>	Agregar colaboradores al proyecto	<p>El sistema debe permitir al gestor del proyecto agregar colaboradores al proyecto que sea responsable, los cuales tendrán un rol asociado.</p> <p>Para agregar un colaborador se lo debe hacer mediante el correo electrónico del colaborador.</p>	Visible
<b>RF015</b>	Eliminar colaborador del proyecto	El sistema debe permitir al gestor del proyecto eliminar un colaborador del proyecto, se debe mostrar un mensaje de confirmación.	Visible

ID	Nombre	Descripción	Categoría
<b>RF016</b>	Revisión de requisitos	El sistema debe permitir al analista de negocios y validador, revisar los requisitos de un proyecto en el que sea colaborador o responsable, este puede agregar una observación por cada requisito.	Visible
<b>RF017</b>	Asignar prioridad a un requisito	El sistema debe permitir al analista de negocios asignar una prioridad a cada requisito creado, en un proyecto en el que sea responsable. La prioridad que puede asignar es: <ul style="list-style-type: none"> <li>● Baja</li> <li>● Media</li> <li>● Alta</li> </ul>	Visible
<b>RF018</b>	Cambiar prioridad de un requisito.	El sistema debe permitir al analista de negocios modificar la prioridad de un requisito creado en un proyecto en el que sea responsable. La prioridad puede modificarse a: <ul style="list-style-type: none"> <li>● Baja</li> <li>● Media</li> <li>● Alta</li> </ul>	Visible
<b>RF019</b>	Registrar cambios en los requisitos	El sistema debe permitir al gestor del proyecto realizar un control de cambios en los requisitos en un proyecto en el que sea responsable. Los cambios que puede hacer son: <ul style="list-style-type: none"> <li>● Nombre del requisito</li> <li>● Descripción del requisito</li> <li>● Prioridad</li> </ul>	Visible
<b>RF020</b>	Generar informe de requisitos	El sistema debe permitir al validador generar un informe de los requisitos del proyecto en el que sea responsable, en el cual se debe incluir la siguiente información: <ul style="list-style-type: none"> <li>● Estado de los requisitos.</li> <li>● Cambios relevantes en los requisitos</li> </ul> El estado de los requisitos puede ser: <ul style="list-style-type: none"> <li>● En revisión.</li> <li>● Validado.</li> <li>● Aprobado.</li> <li>● Rechazado.</li> </ul>	Visible
<b>RF021</b>	Registrar cuenta de usuario	El sistema debe permitir a un usuario no registrado, crear una cuenta mediante un formulario con los siguientes campos: <ul style="list-style-type: none"> <li>● Nombres</li> <li>● Apellidos</li> </ul>	Visible

ID	Nombre	Descripción	Categoría
		<ul style="list-style-type: none"> <li>• Correo</li> </ul>	
<b>RF022</b>	Actualizar información del usuario	<p>El sistema debe permitir al gestor del proyecto modificar/actualizar la información de su cuenta de usuario:</p> <ul style="list-style-type: none"> <li>• Celular.</li> <li>• Dirección.</li> <li>• Correo de respaldo.</li> </ul>	Visible
<b>RF023</b>	Generar contraseña	El sistema generará una contraseña para cada usuario registrado.	Visible
<b>RF024</b>	Generar nueva contraseña	El sistema debe permitir a los usuarios generar una nueva contraseña para su cuenta de usuario.	Visible
<b>RF025</b>	Listar usuarios	El sistema debe permitir al administrador visualizar todos los usuarios registrados en el sistema.	Inferido
<b>RF026</b>	Activar cuenta de usuario	El sistema debe permitir al administrador activar una cuenta de un usuario registrado en el sistema, en caso de que haya sido desactivada.	Visible
<b>RF027</b>	Asignar rol a los usuarios	<p>El sistema debe permitir al administrador asignar un único rol a cada usuario, los roles que se pueden asignar son:</p> <ul style="list-style-type: none"> <li>• Gestor del proyecto</li> <li>• Analista de negocios</li> <li>• Validador</li> </ul>	Visible
<b>RF028</b>	Modificar rol de usuario	<p>El sistema debe permitir al administrador modificar el rol asignado a un usuario, los roles que se pueden asignar son:</p> <ul style="list-style-type: none"> <li>• Gestor del proyecto</li> <li>• Analista de negocios</li> <li>• Validador</li> </ul>	
<b>RF029</b>	Mostrar gráficas del estado de los requisitos	<p>El sistema debe mostrar gráficas del estado de los requisitos, considerando:</p> <ul style="list-style-type: none"> <li>• Nombre del proyecto.</li> <li>• Número de requisitos.</li> </ul>	Inferido

<b>ID</b>	<b>Nombre</b>	<b>Descripción</b>	<b>Categoría</b>
<b>RF031</b>	Inicio de sesión	El sistema debe permitir al gestor del proyecto, analista de negocios, validador y administrador, iniciar sesión mediante sus credenciales (correo y contraseña).	Visible

A continuación, se detallan los requisitos no funcionales:

**Tabla 3.** Requisitos no funcionales

<b>ID</b>	<b>Categoría</b>	<b>Descripción</b>
<b>RNF01</b>	Eficiencia	La herramienta debe brindar un tiempo aceptable de respuesta entre 1 a 5 segundos promedio.
<b>RNF02</b>		La herramienta debe ser capaz de soportar un máximo de 1000 usuarios de manera simultánea.
<b>RNF03</b>	Seguridad	Los permisos de roles podrán ser cambiados únicamente por el administrador.
<b>RNF04</b>		Las contraseñas deben ser generadas con un algoritmo de manera aleatoria por el sistema.
<b>RNF05</b>		La herramienta debe permitir el acceso a los usuarios de acuerdo con sus credenciales y permisos asignados.
<b>RNF06</b>	Disponibilidad	La herramienta debe estar disponible las 24 horas del día, todos los días de la semana, durante un año.
<b>RNF07</b>		El tiempo para iniciar el sistema no debe ser mayor a 3 minutos.
<b>RNF08</b>	Usabilidad	La herramienta debe tener una interfaz sencilla, amigable e intuitiva para los usuarios.
<b>RNF09</b>		El tiempo de aprendizaje de la herramienta para un usuario no debe ser mayor a 3 horas.

<b>ID</b>	<b>Categoría</b>	<b>Descripción</b>
<b>RNF10</b>		La herramienta debe presentar mensajes de alerta e informativos al usuario.
<b>RNF11</b>	Mantenibilidad	El código fuente de la herramienta debe ser claro y fácil de escalar.
<b>RNF12</b>	Portabilidad	El entorno de programación de la herramienta será Nodejs, JavaScript, TypeScript, Nextjs.
<b>RNF13</b>		Se emplearán una base de datos no relacional, la cual debe ser compatible con el entorno de programación.
<b>RNF14</b>		La herramienta será implementada para navegadores web (Google Chrome, Mozilla Firefox, Microsoft Edge).

### ***6.1.2 Modelo de dominio de negocio***

En la figura 14, se muestra el diagrama que corresponde al modelo de dominio del negocio, el cual representa el flujo que existe entre cada objeto de la aplicación.

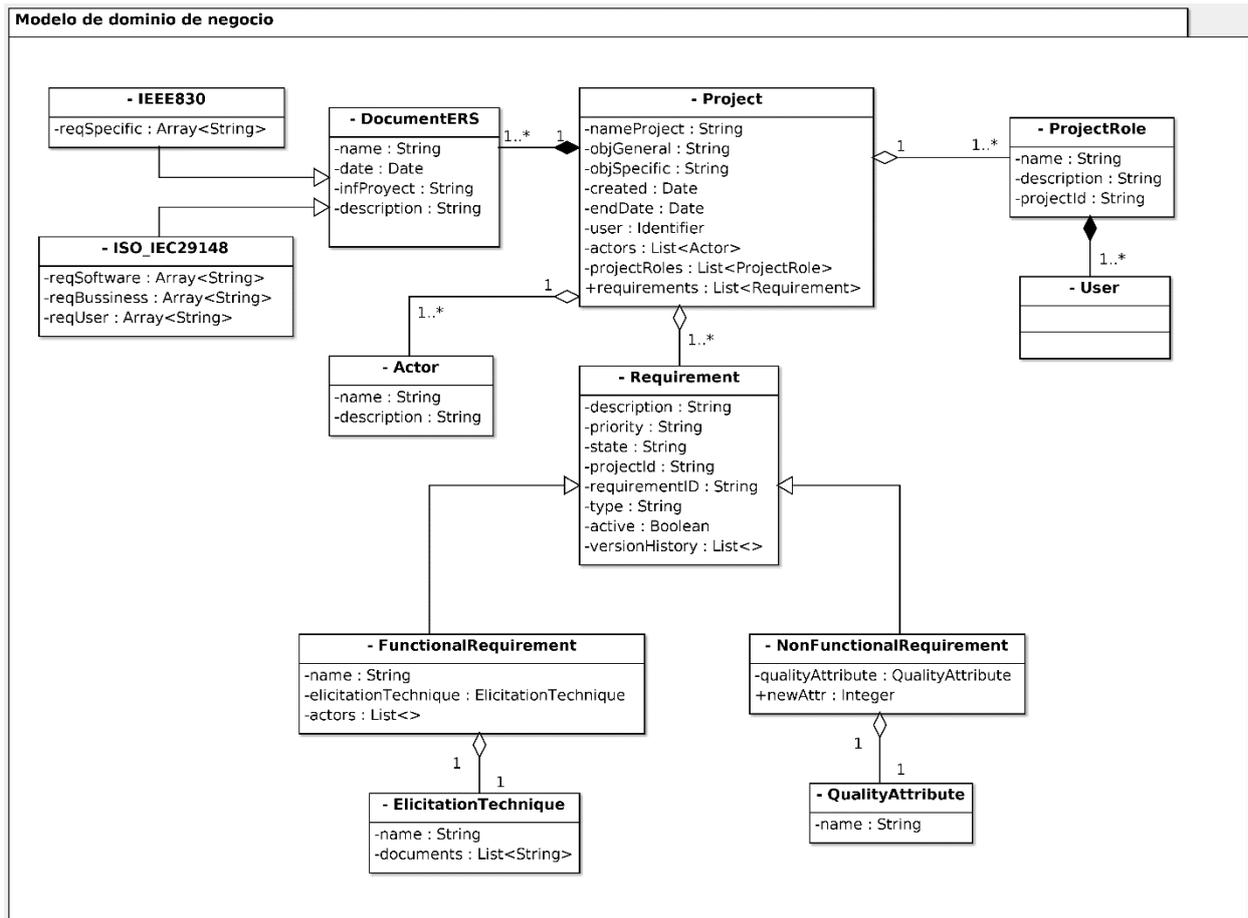


Figura 14. Modelo de dominio del negocio.

### 6.1.3 Modelo de dominio – usuarios

En la figura 15, se muestra el modelo de dominio de usuarios, que representa el modelo conceptual de seguridad de los usuarios.

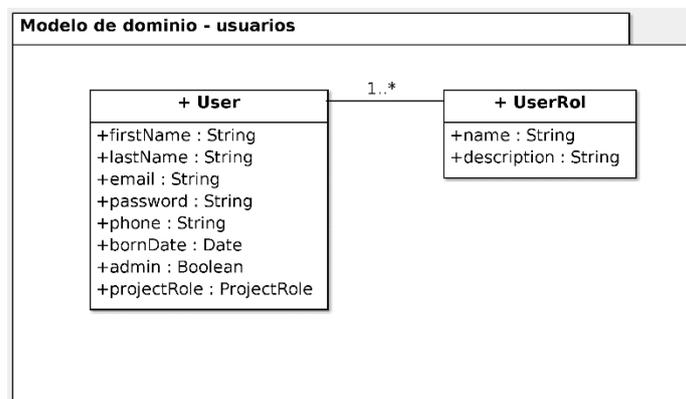


Figura 15. Modelo de dominio – usuarios

## 6.1.4 Arquitectura de software

### 6.1.4.1 Arquitectura física

Los presentes diagramas, muestran el estilo arquitectónico de la aplicación, tanto para despliegue como para construcción, especificados en los siguientes diagramas:

#### 6.1.4.1.1 Diagrama de componentes

En la figura 16, se muestra la organización de los componentes de la aplicación; el estilo arquitectónico empleado para el servidor web, es MVC, teniendo modelos, controladores, y rutas en lugar de vistas, mientras que para el cliente se empleó el patrón basado en componentes.

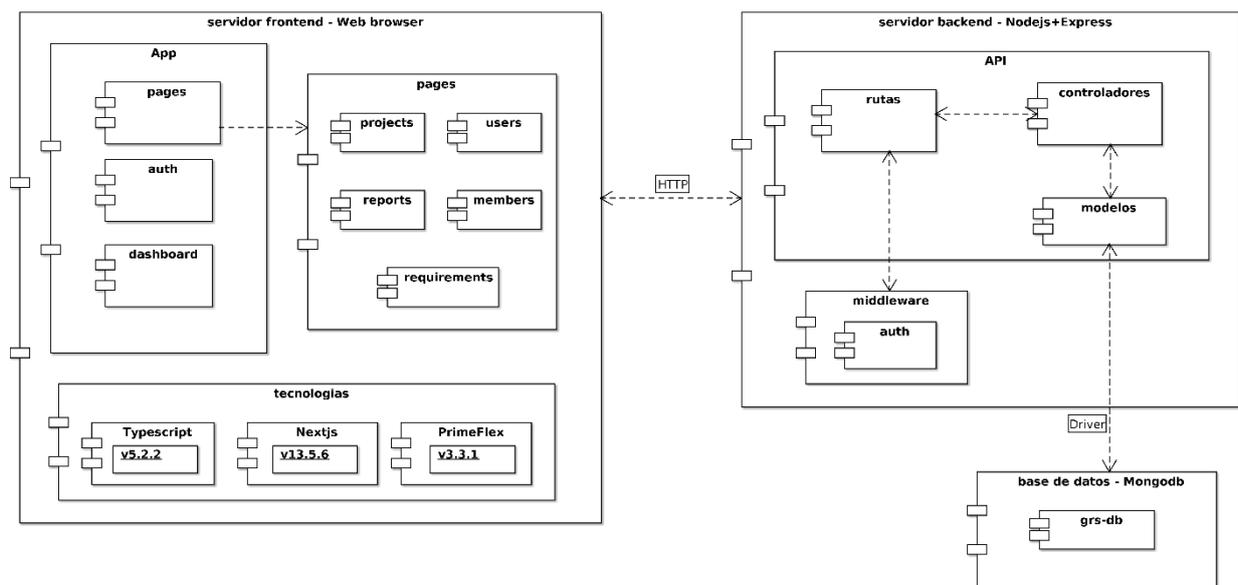
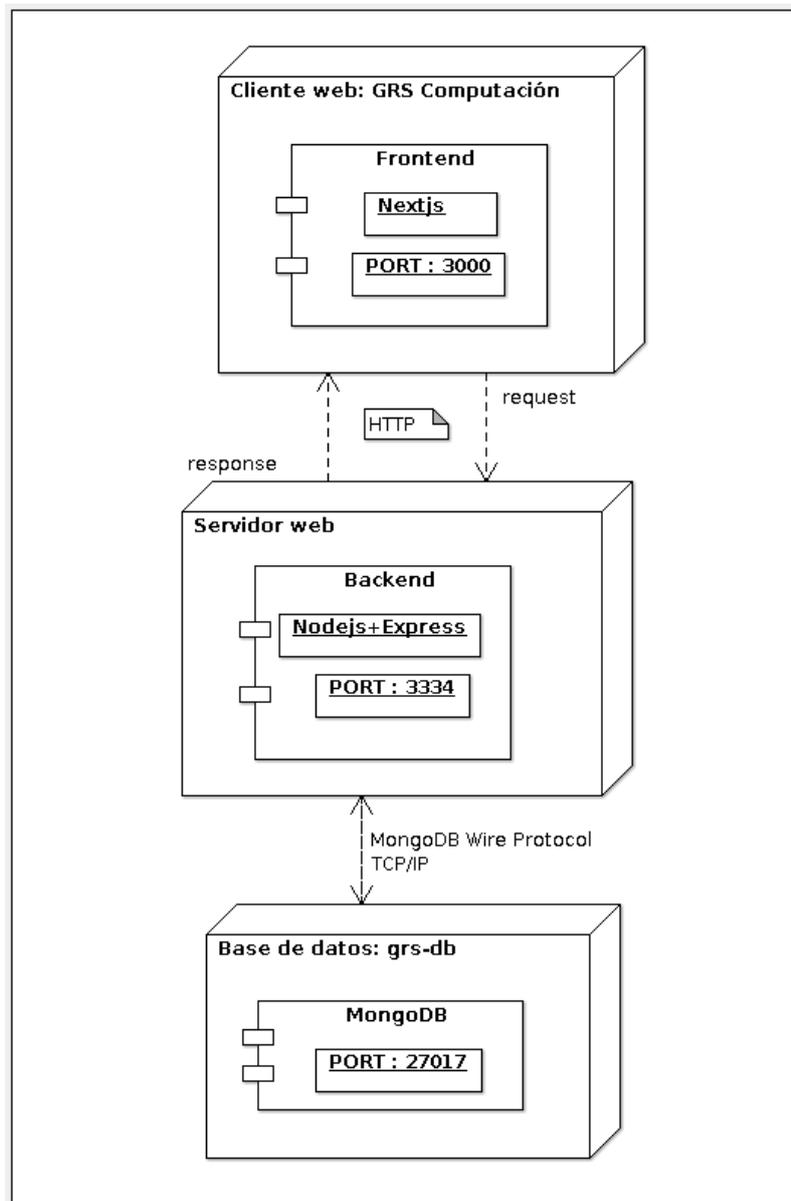


Figura 16. Diagrama de componentes de la aplicación.

#### 6.1.4.1.2 Diagrama de despliegue

Para el despliegue de la aplicación se ha empleado el estilo arquitectónico cliente – servidor, que a su vez sigue una estructura backend y frontend. El backend es el encargado de llevar a cabo las reglas del negocio, mientras que el frontend representa la parte de presentación. Además, para la comunicación del servidor con el cliente, se emplea el estilo RESTful, y para la comunicación del servidor con la base de datos, se emplea el protocolo MongoDB Wire Protocol, sobre TCP/IP.

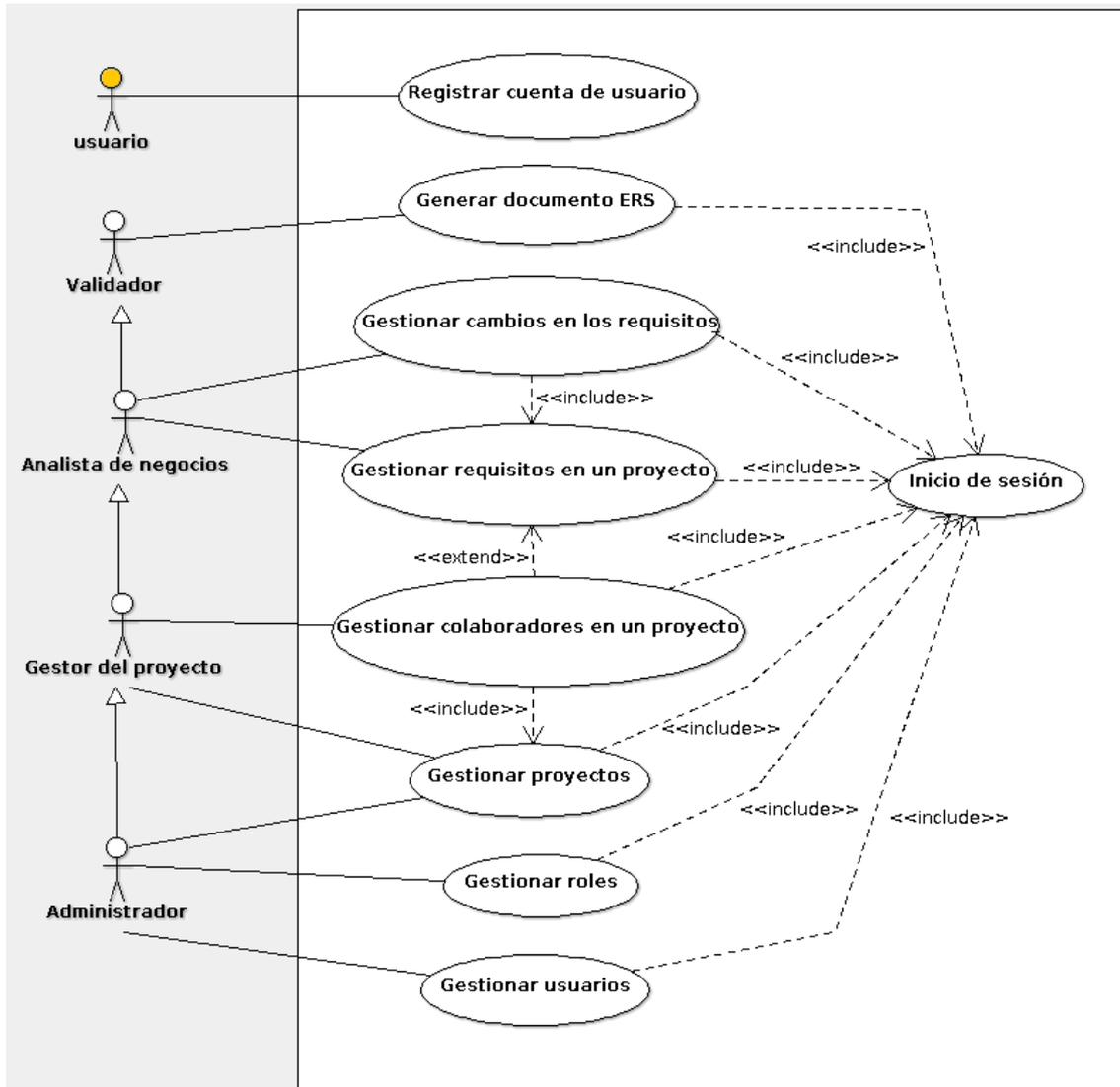


**Figura 17.** Diagrama de despliegue de la aplicación.

### **6.1.4.2 Arquitectura lógica**

#### **6.1.4.2.1 Diagrama de casos de uso general**

Con el diseño del modelo de dominio de negocio final (ver sección **6.1.2**), se procedió a realizar el diagrama de casos de uso del sistema, para una mejor comprensión de la interacción entre los componentes y los procesos del sistema (ver figura 10).



**Figura 18.** Diagrama de casos de uso del sistema.

A continuación, se muestra la descripción de los casos de uso principales.

**a. Inicio de sesión**

**Tabla 4.** Caso de uso - Inicio de sesión.

<b>CU01</b>	<b>Actores:</b> Gestor del proyecto/Analista de Negocios/ Validador/ Administrador	<b>Prioridad:</b> Alta
<b>Caso de uso:</b> Inicio de sesión		
<b>Descripción:</b> El usuario inicia sesión con sus credenciales (correo y contraseña).		
<b>Referencias:</b> RF031		

---

**Precondiciones:**

- El usuario debe estar registrado en el sistema

---

**Postcondiciones:**

- Usuario autenticado y autorizado.

---

**Flujo principal:**

- El usuario ingresa a la página de inicio de sesión.
- El usuario ingresa sus credenciales (correo y contraseña).
- El usuario da clic en el botón “Iniciar sesión”.
- El sistema verifica las credenciales de acceso del usuario.
- El sistema carga las funciones del rol del usuario.
- El sistema redirige a la página principal “dashboard”.

---

**Flujo alterno:****Campo(s) de entrada vacío(s):**

- El sistema muestra un mensaje indicando que existen campos obligatorios vacíos.

**Credenciales incorrectas:**

- El sistema muestra un mensaje indicando que las credenciales de acceso son incorrectas.
- 

**b. Gestionar proyectos – crear proyecto****Tabla 5.** Caso de uso – crear proyecto.

---

**CU02 Actores:** Gestor del proyecto/Administrador**Prioridad:** Alta

---

**Caso de uso:** Gestionar proyectos – crear proyecto

---

**Descripción:** El gestor del proyecto y administrador crean nuevos proyectos.

---

**Referencias:** RF01, RF02, RF03, RF04

---

**Precondiciones:**

- El usuario debe haber iniciado sesión.
- El usuario debe estar en la página proyectos

---

**Postcondiciones:**

- El gestor del proyecto y administrador crean un nuevo proyecto

---

**Flujo principal:**

- El usuario (Gestor del proyecto/administrador) pulsa el botón “Crear Nuevo Proyecto”.
  - El sistema muestra un formulario crear nuevo proyecto
-

- 
- El usuario ingresa la información del proyecto (nombre del proyecto, objetivo general, objetivo específico, fecha de inicio, fecha final).
  - El usuario pulsa el botón “Crear Proyecto”.
  - El sistema valida que la información del formulario este completa
  - El sistema guarda la información del proyecto y muestra un mensaje indicando que se ha creado el proyecto con éxito.
  - El sistema muestra el nuevo proyecto creado.
- 

**Flujo alternativo:**

**Campo(s) obligatorios vacío(s):**

- El sistema muestra un mensaje indicando que existen campos obligatorios vacíos.
- 

**c. Gestionar proyectos – actualizar proyecto**

**Tabla 6.** Caso de uso – actualizar proyecto.

<b>CU03</b>	<b>Actores:</b> Gestor del proyecto/Administrador	<b>Prioridad:</b> Alta
<b>Caso de uso:</b> Gestionar proyectos – crear proyecto		
<b>Descripción:</b> El gestor del proyecto y administrador actualizan la información de un proyecto		
<b>Referencias:</b> RF01, RF02, RF03, RF04		
<b>Precondiciones:</b>		
<ul style="list-style-type: none"> <li>- El usuario debe haber iniciado sesión.</li> <li>- El usuario debe estar en la página proyectos</li> </ul>		
<b>Postcondiciones:</b>		
<ul style="list-style-type: none"> <li>- Se actualiza la información de un proyecto</li> </ul>		
<b>Flujo principal:</b>		
<ul style="list-style-type: none"> <li>- El usuario selecciona un proyecto de la lista de proyectos.</li> <li>- El usuario pulsa el botón “Ver proyecto”.</li> <li>- El sistema muestra la información del proyecto seleccionado (nombre del proyecto, objetivo general, objetivos específicos, fecha de inicio, fecha de finalización, actores)</li> <li>- El usuario pulsa en el botón ‘Editar información’.</li> <li>- El usuario ingresa la información del proyecto (nombre del proyecto, objetivo general, objetivos específicos, fecha de inicio, fecha final).</li> </ul>		

---

- 
- El usuario pulsa el botón “Guardar cambios”.
  - El sistema valida que la información del formulario este completa.
  - El sistema guarda la información del proyecto y muestra un mensaje indicando que se ha actualizado la información del proyecto con éxito.
  - El sistema muestra el proyecto actualizado.
- 

**Flujo alternativo:**

**Campo(s) obligatorios vacío(s):**

- El sistema muestra un mensaje indicando que existen campos obligatorios vacíos.
- 

**d. Gestionar proyectos – dar de baja proyecto**

**Tabla 7.** Caso de uso – dar de baja proyecto.

---

<b>CU04</b>	<b>Actores:</b> Gestor del proyecto/Administrador	<b>Prioridad:</b> Alta
<b>Caso de uso:</b> Gestionar proyectos – dar de baja proyecto		
<b>Descripción:</b> El gestor del proyecto da de baja un proyecto		
<b>Referencias:</b> RF01, RF02, RF03, RF04		
<b>Precondiciones:</b>		
<ul style="list-style-type: none"> <li>- El usuario debe estar en la página proyectos</li> </ul>		
<b>Postcondiciones:</b>		
<ul style="list-style-type: none"> <li>- El proyecto seleccionado es dado de baja</li> <li>- El sistema muestra la lista de proyectos actualizada</li> </ul>		
<b>Flujo principal:</b>		
<ul style="list-style-type: none"> <li>- El usuario selecciona un proyecto de la lista de proyectos.</li> <li>- El usuario pulsa el botón “Dar de baja”.</li> <li>- El sistema muestra un mensaje de confirmación “¿Esta seguro de dar de baja este proyecto?”</li> <li>- El usuario pulsa en el botón “Si”.</li> <li>- El sistema da de baja el proyecto y muestra un mensaje de éxito.</li> <li>- El sistema actualiza la lista de proyectos.</li> <li>- El sistema muestra la lista de proyectos actualizada.</li> </ul>		

---

---

**Flujo alternativo:****Usuario pulsa en el botón “No”:**

- El sistema muestra un mensaje indicando que se ha cancelado la operación.
- 

**e. Gestionar proyecto – crear actor****Tabla 8.** Caso de uso – *crear actor*

---

<b>CU05</b>	<b>Actores:</b> Gestor del proyecto	<b>Prioridad:</b> Alta
-------------	-------------------------------------	------------------------

---

**Caso de uso:** Gestionar proyectos – crear actor

---

**Descripción:** El gestor del proyecto crea un nuevo actor

---

**Referencias:** RF01, RF02, RF03, RF04

---

**Precondiciones:**

- El usuario debe estar en la página proyectos
- El usuario debe seleccionar un proyecto
- El usuario debe pulsar el botón “Ver proyecto”

---

**Postcondiciones:**

- Se crea un nuevo actor para el proyecto.

---

**Flujo principal:**

- El usuario pulsa el botón “Crear nuevo actor”.
- El sistema muestra un formulario para crear el nuevo actor (nombre, descripción).
- El usuario ingresa la información del nuevo actor (nombre, descripción).
- El usuario pulsa el botón “Crear Actor”.
- El sistema valida que la información del formulario este completa.
- El sistema guarda la información del actor y muestra un mensaje indicando que se ha creado el actor con éxito.
- El sistema actualiza la página y muestra el actor creado.

---

**Flujo alternativo:**

**Campo(s) obligatorios vacío(s):**

- El sistema muestra un mensaje indicando que existen campos obligatorios vacíos.

---

## f. Gestionar requisitos de un proyecto – crear requisito funcional

**Tabla 9.** Caso de uso – Crear requisito funcional

---

<b>CU06</b>	<b>Actores:</b> Gestor del proyecto/analista de negocios	<b>Prioridad:</b> Alta
-------------	--	------------------------

---

**Caso de uso:** Gestionar requisitos de un proyecto – crear nuevo requisito funcional

---

**Descripción:** El gestor del proyecto, y analista de negocios crean un nuevo requisito funcional en un proyecto

---

**Referencias:** RF06, RF07, RF08, RF09

---

**Precondiciones:**

- El usuario debe seleccionar un proyecto creado.
- El usuario debe pulsar el botón “Ver proyecto”

---

**Postcondiciones:**

- Se crea un nuevo requisito en un proyecto seleccionado.

---

**Flujo principal:**

- El usuario pulsa el botón “Gestionar requisitos”
- El sistema redirige a la página requisitos del proyecto.
- El usuario pulsa el botón “Crear nuevo requisito”.
- El sistema muestra un formulario para crear el nuevo requisito
- El usuario selecciona el tipo de requisito “Funcional” de la lista de opciones.
- El sistema muestra los campos correspondientes para crear un requisito funcional (técnica de elicitación, documento de elicitación, actor, acción, contexto, prioridad).
- El usuario ingresa la información del nuevo requisito.
- El sistema muestra una previsualización del requisito y una historia de usuario del requisito.
- El usuario pulsa el botón “Crear Requisito”.
- El sistema valida la información ingresada por el usuario.
- El sistema guarda la información del requisito y muestra un mensaje de éxito.

---

**Flujo alterno:**

**Campo(s) de entrada vacío(s):**

- El sistema muestra un mensaje indicando que existen campos obligatorios vacíos.

**Documento seleccionado no admitido:**

- El sistema muestra un mensaje indicando que el documento de elicitación no es del tipo admitido.

---

## g. Gestionar requisito de un proyecto - crear requisito no funcional

**Tabla 10.** Caso de uso – Crear requisito no funcional

---

<b>CU07</b>	<b>Actores:</b> Gestor del proyecto/analista de negocios	<b>Prioridad:</b> Alta
-------------	--	------------------------

---

**Caso de uso:** Gestionar requisitos de un proyecto – crear nuevo requisito no funcional

---

**Descripción:** El gestor del proyecto, y analista de negocios crean un nuevo requisito no funcional en un proyecto.

---

**Referencias:** RF06, RF07, RF08, RF09

---

**Precondiciones:**

- El usuario debe seleccionar un proyecto creado.
- El usuario debe pulsar el botón “Ver proyecto”

---

**Postcondiciones:**

- Se crea un nuevo requisito en un proyecto seleccionado.

---

**Flujo principal:**

- El usuario pulsa el botón “Gestionar requisitos”
- El sistema redirige a la página requisitos del proyecto.
- El usuario pulsa el botón “Crear nuevo requisito”.
- El sistema muestra un formulario para crear el nuevo requisito
- El usuario selecciona el tipo de requisito “No Funcional” de la lista de opciones.
- El sistema muestra los campos correspondientes para crear un requisito no funcional (atributo de calidad, descripción, prioridad).
- El usuario ingresa la información del nuevo requisito.
- El sistema muestra una previsualización del requisito y una historia de usuario del requisito.
- El usuario pulsa el botón “Crear Requisito”.
- El sistema valida la información ingresada por el usuario.
- El sistema guarda la información del requisito y muestra un mensaje de éxito.

---

**Flujo alterno:**

**Campo(s) de entrada vacío(s):**

- El sistema muestra un mensaje indicando que existen campos obligatorios vacíos.

---

### 6.1.4.3 Prototipado de la herramienta

En las figuras 19, 20, 21, 22 y 23 se muestra el prototipado de las vistas principales de la aplicación, haciendo uso de la herramienta Balsamiq.

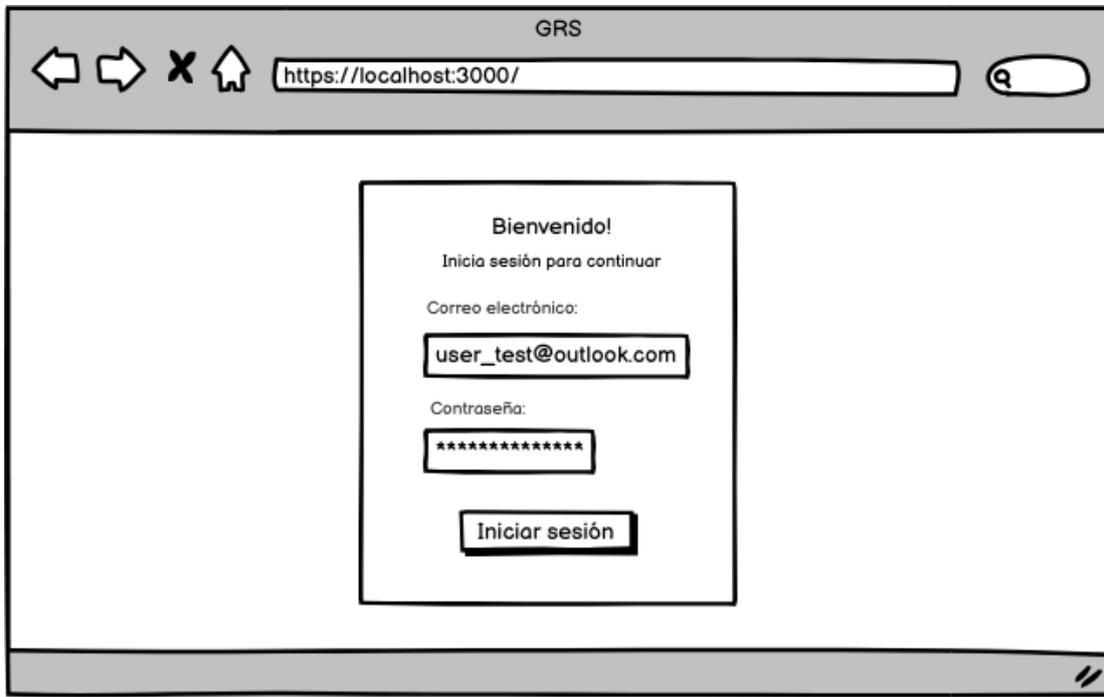


Figura 19. Prototipado de inicio de sesión

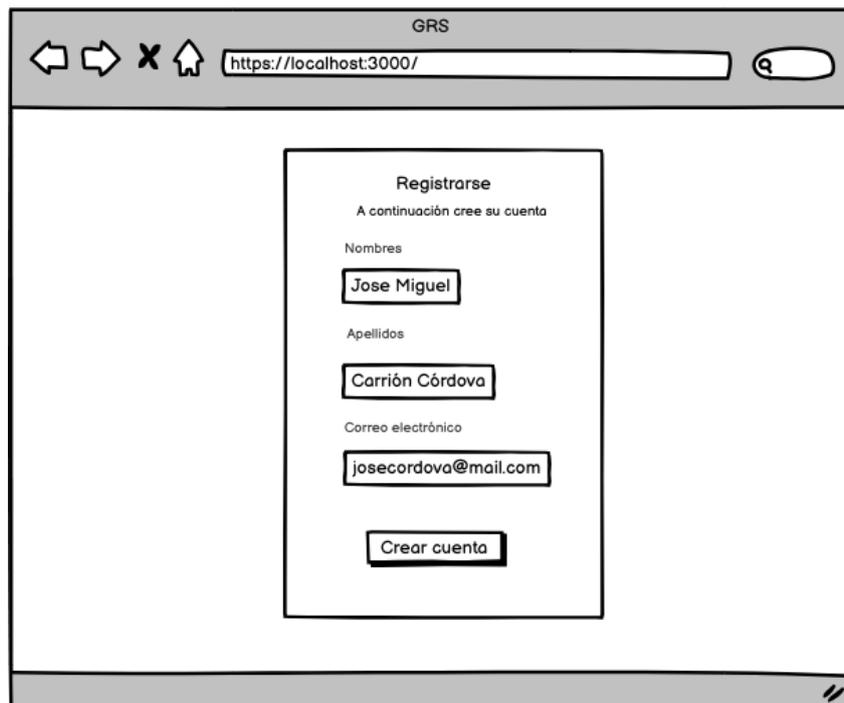


Figura 20. Prototipado de registro de usuario.

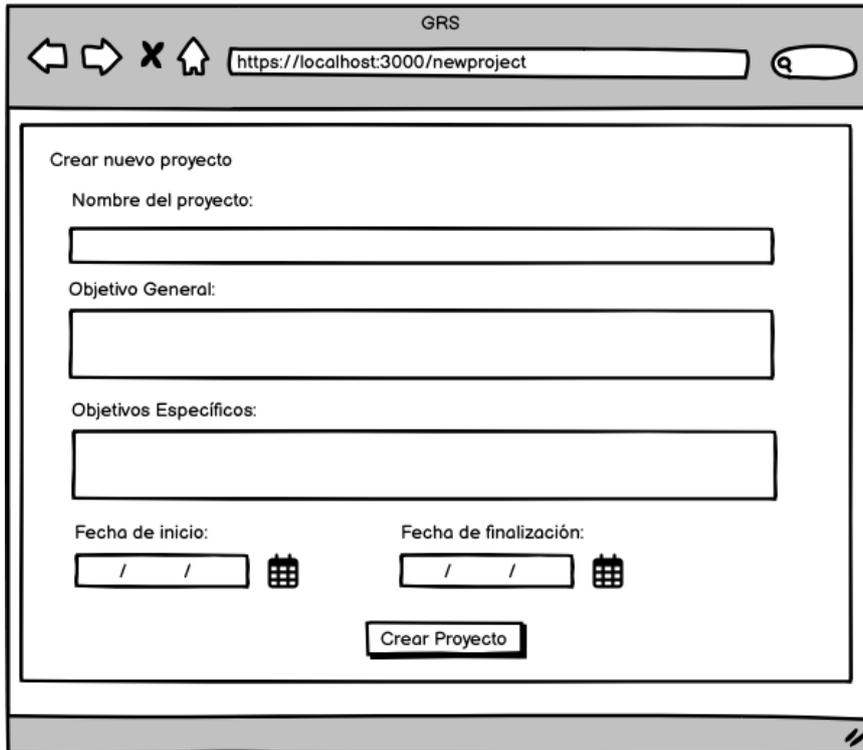


Figura 21. Prototipo de creación del proyecto.

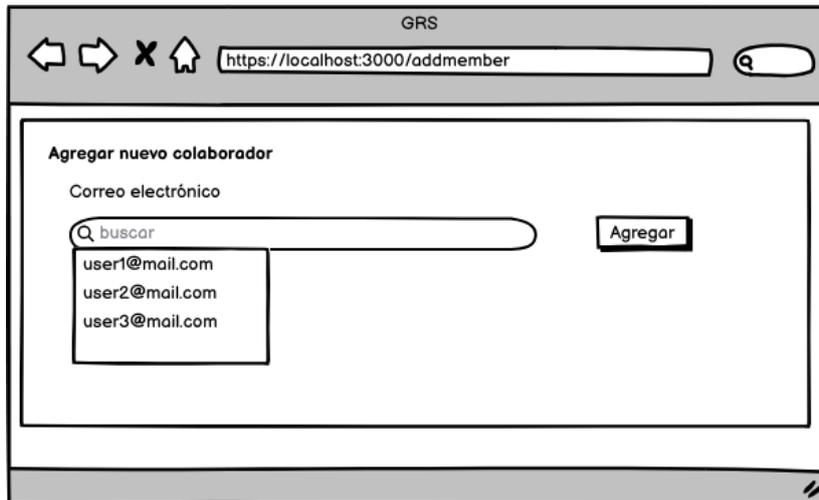


Figura 22. Prototipo de agregar colaborador.

GRS

https://localhost:3000/newrequirement

**Crear nuevo requisito**

ID:  Título:

Cómo:

Quiero:

Para poder:

Complemento:

Debe contener la siguiente información:

- Nombre del producto
- Descripción
- Categoría
- Precio

Tipo de requisito:

Técnica de elicitación:

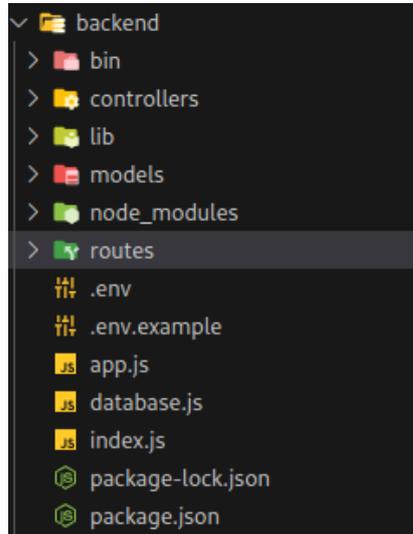
**Figura 23.** Prototipado de crear requisito.

## 6.1.5 Codificación

### 6.1.5.1 Estructura del proyecto

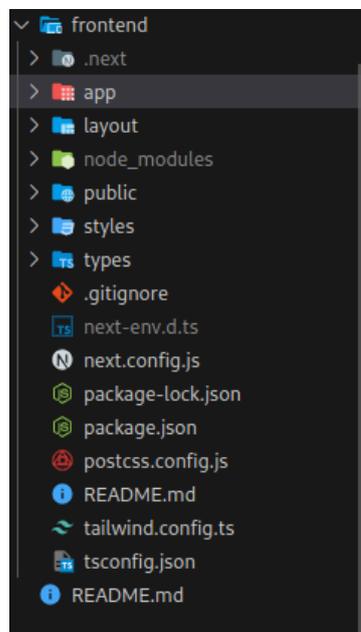
El proyecto se encuentra estructurado en dos carpetas principales: Backend y Frontend, en la figura 24 se muestra la estructura del backend, mientras que en la figura 25 la estructura del Frontend. El código fuente se encuentra alojado en el siguiente repositorio: <https://github.com/Computacion-UNL/ers-tool-cc>.

La estructura del backend contiene la siguiente estructura básica: modelos, controladores y rutas. Se incluye también la configuración de la base de datos y del servidor web.



**Figura 24.** Estructura del backend.

La estructura del frontend sigue el siguiente esquema: app, que es la carpeta principal en donde se manejan los componentes y vistas, layout en la que se encuentran los componentes y menús que se reutilizan en los distintos componentes de la aplicación; la carpeta public contiene los iconos e imágenes de la aplicación y la carpeta styles los estilos empleados para los componentes de la aplicación.



**Figura 25.** Estructura del frontend.

### 6.1.5.2 Ejecución de pruebas unitarias al servicio API-REST

Se desarrolló las pruebas unitarias al servicio web API-REST empleando la herramienta Postman, mediante las peticiones HTTP se verificó el correcto funcionamiento de los diferentes servicios/métodos creados en la API. En la tabla 6 se muestran los resultados de la ejecución de las solicitudes realizadas al servidor (ver anexo 6).

**Tabla 6.** Resultado de ejecución de pruebas unitarias

URL	Token	Método	Descripción	Tiempo de respuesta
<a href="http://localhost:3334/grs/api/v1/auth/signup">http://localhost:3334/grs/api/v1/auth/signup</a>	NO	POST	Registro de usuario	2.19s
<a href="http://localhost:3334/grs/api/v1/auth/login">http://localhost:3334/grs/api/v1/auth/login</a>	NO	POST	Inicio de sesión	182ms
<a href="http://localhost:3334/grs/api/v1/auth/logout">http://localhost:3334/grs/api/v1/auth/logout</a>	NO	POST	Cierre de sesión	218ms
<a href="http://localhost:3334/grs/api/v1/projects/user/">http://localhost:3334/grs/api/v1/projects/user/</a>	SI	GET	Obtener proyectos del usuario	200 ms
<a href="http://localhost:3334/grs/api/v1/projects/66e1238ae40a328a46742bd4">http://localhost:3334/grs/api/v1/projects/66e1238ae40a328a46742bd4</a>	SI	GET	Obtener un proyecto por id	107ms
<a href="http://localhost:3334/grs/api/v1/projects/create">http://localhost:3334/grs/api/v1/projects/create</a>	SI	POST	Crear nuevo proyecto	734ms
<a href="http://localhost:3334/grs/api/v1/projects/update/66b8f1a7ab16d9e40d54deec">http://localhost:3334/grs/api/v1/projects/update/66b8f1a7ab16d9e40d54deec</a>	SI	PUT	Actualizar un proyecto	120ms
<a href="http://localhost:3334/grs/api/v1/project-roles/all/66b8f1a7ab16d9e40d54deec">http://localhost:3334/grs/api/v1/project-roles/all/66b8f1a7ab16d9e40d54deec</a>	SI	GET	Obtener roles del proyecto	90ms
<a href="http://localhost:3334/grs/api/v1/members/create">http://localhost:3334/grs/api/v1/members/create</a>	SI	POST	Agregar nuevo colaborador a un proyecto	3.85s

URL	Token	Método	Descripción	Tiempo de respuesta
<a href="http://localhost:3334/grs/api/v1/projects/66b8f1a7ab16d9e40d54deec/all-actors">http://localhost:3334/grs/api/v1/projects/66b8f1a7ab16d9e40d54deec/all-actors</a>	SI	GET	Obtiene todos los actores de un proyecto	112ms
<a href="http://localhost:3334/grs/api/v1/projects/66e1238ae40a328a46742bd4/create-actor">http://localhost:3334/grs/api/v1/projects/66e1238ae40a328a46742bd4/create-actor</a>	SI	POST	Crear nuevo actor de un proyecto	250ms
<a href="http://localhost:3334/grs/api/v1/projects/66b8f1a7ab16d9e40d54deec/requirements">http://localhost:3334/grs/api/v1/projects/66b8f1a7ab16d9e40d54deec/requirements</a>	SI	GET	Obtener todos los requisitos de un proyecto	232ms
<a href="http://localhost:3334/grs/api/v1/requirements/66b8f2a2c4e2668f20b4c92c">http://localhost:3334/grs/api/v1/requirements/66b8f2a2c4e2668f20b4c92c</a>	SI	GET	Obtener un requisito por su ID.	108ms
<a href="http://localhost:3334/grs/api/v1/requirements/create/">http://localhost:3334/grs/api/v1/requirements/create/</a>	SI	POST	Crear un nuevo requisito en un proyecto	762ms
<a href="http://localhost:3334/grs/api/v1/requirements/update/66b8f2a2c4e2668f20b4c92c">http://localhost:3334/grs/api/v1/requirements/update/66b8f2a2c4e2668f20b4c92c</a>	SI	PUT	Actualizar un requisito	717ms
<a href="http://localhost:3334/grs/api/v1/requirements/versions/66b8f2a2c4e2668f20b4c92c">http://localhost:3334/grs/api/v1/requirements/versions/66b8f2a2c4e2668f20b4c92c</a>	SI	GET	Obtener las versiones de un requisito	412ms
<a href="http://localhost:3334/grs/api/v1/users/user-profile">http://localhost:3334/grs/api/v1/users/user-profile</a>	SI	GET	Obtener perfil de usuario	105ms
<a href="http://localhost:3334/grs/api/v1/users/search/:searchTerm">http://localhost:3334/grs/api/v1/users/search/:searchTerm</a>	SI	GET	Buscar usuario por término de búsqueda	101ms
<a href="http://localhost:3334/grs/api/v1/profile/recovery-password">http://localhost:3334/grs/api/v1/profile/recovery-password</a>	NO	POST	Reestablecer contraseña olvidada	2.61s
<a href="http://localhost:3334/grs/api/v1/profile/update-password">http://localhost:3334/grs/api/v1/profile/update-password</a>	SI	PUT	Actualizar/Modificar contraseña de usuario	3.26s

## **6.2 Objetivo 2. Evaluar la aplicación web desarrollada para determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema).**

Para el cumplimiento del objetivo 2, se llevaron a cabo las siguientes tareas: diseño de cuestionarios y encuestas, y ejecución de las pruebas, las cuales corresponden a la fase de implementación de la metodología ICONIX.

### **6.2.1 Diseño de cuestionarios - encuestas:**

Se elaboró el cuestionario planteado por la metodología SUS (ver **Anexo 3**), el cual consta de diez preguntas predefinidas, enfocadas en la usabilidad de la aplicación web. En la tabla 11, se detalla las preguntas que fueron establecidas para realizar la encuesta:

**Tabla 7.** Preguntas de la metodología SUS

<b>No.</b>	<b>Preguntas</b>
1	¿Creo que usaría esta aplicación con frecuencia?
2	Encontré la aplicación innecesariamente compleja
3	Creo que la aplicación fue fácil de usar
4	Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar esta aplicación
5	Las funciones de esta aplicación están bien integradas
6	Creo que la aplicación es muy inconsistente (comportamiento incorrecto)
7	Me imagino que la mayoría de los usuarios aprendería a usar esta aplicación de forma muy rápida
8	Encuentro que la aplicación es muy difícil de usar.
9	Me sentí muy seguro usando la aplicación
10	Necesité aprender muchas cosas antes de usar esta aplicación

El diseño del cuestionario SUS (ver Anexo 3) se realizó en el siguiente formulario de Google Forms: <https://forms.gle/BCgwY67FJT65VZHQ7>.

### **6.2.2 Ejecución de las pruebas:**

La ejecución de las pruebas de aceptación se realizó aplicando el cuestionario según la metodología SUS (ver **sección 6.2.2.3**), las cuales permitieron determinar el grado de aceptación de la aplicación web mediante la experiencia del usuario (UX).

#### **6.2.2.1 Definición de la muestra**

Para la determinación del tamaño de la muestra, se empleó un muestreo no probabilístico por conveniencia, este permitió escoger los sujetos de evaluación de acuerdo a su disponibilidad y relación con el tema de investigación. Esto favoreció a que los usuarios brindaran la información relevante para determinar el grado de aceptación de la aplicación web, mediante la metodología SUS. Se tomó un total de 31 evaluadores, entre estudiantes de 3ro, 7mo y 9no ciclo, así como profesionales afines a la Carrera Computación.

#### **6.2.2.2 Definición de escala de puntuación**

El modelo de puntuación escogido para la ejecución de las pruebas, es la escala de Likert. En la tabla 12 se especifica los niveles de puntuación que se establecieron. Así mismo, en la tabla 13 se establece la media de calificación considerada para los resultados de la evaluación.

**Tabla 8.** Definición de niveles de puntuación

<b>Valor de puntuación</b>	<b>Escala de puntuación</b>
1	Totalmente en desacuerdo
2	En desacuerdo
3	Neutro
4	De acuerdo
5	Totalmente de acuerdo

**Tabla 9.** Definición de media de calificación

<b>Rango de calificación</b>	<b>Media de calificación</b>
< 51.7	F
51.7 < 62.7	D
62.7 < 72.6	C
72.6 < 78.9	B
≥ 78.9	A

El cálculo de la puntuación total se definió acorde lo indica la metodología SUS, la cual establece que:

- Para las preguntas IMPARES: P1, P3, P5, P7, P9, se resta el valor de 1 a cada puntaje obtenido (ej: Puntaje T.= Puntaje obtenido – 1).
- Para las preguntas PARES: P2, P4, P6, P8, P10, se resta el valor de 5 respecto del puntaje obtenido (ej: Puntaje T.= 5 – Puntaje obtenido).
- El resultado Total es *SUS Score* =  $(\sum \text{Puntajes ajustados}) * 2.5$

### 6.2.2.3 Resultados de la ejecución de pruebas

En la tabla 14 se muestra los resultados obtenidos de la evaluación, estableciendo el porcentaje de ponderación obtenido en cada pregunta de acuerdo a los criterios de puntuación definidos.

**Tabla 10.** Resultados obtenidos por cada pregunta.

<b>P1</b>	<b><i>¿Creo que usaría esta aplicación con frecuencia?</i></b>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>

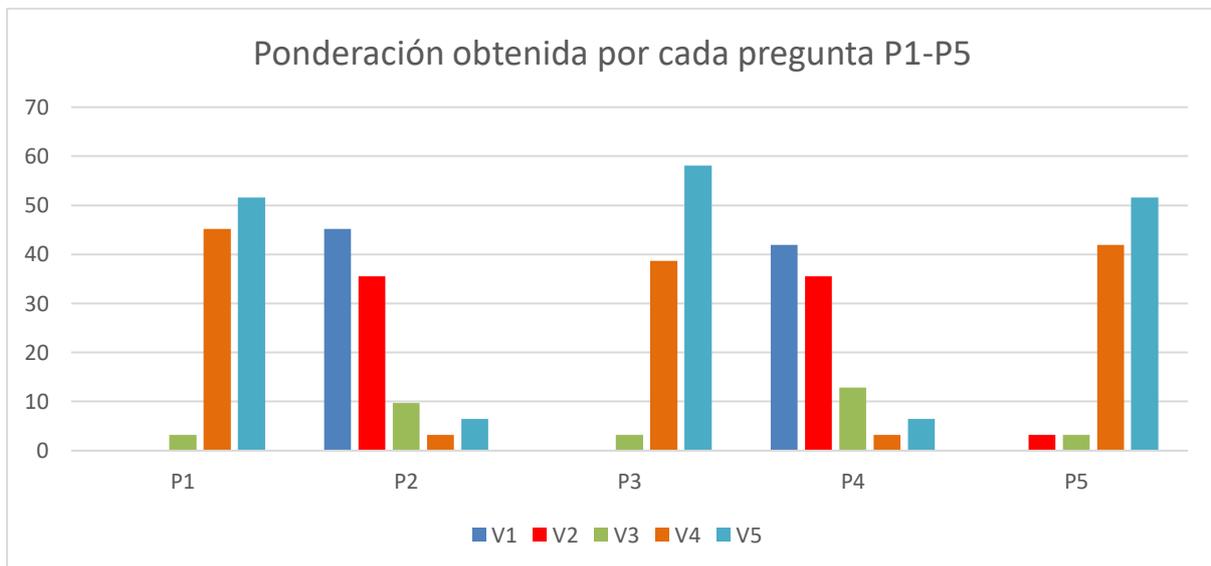
	0	0	3,2	45,2	51,6
<b>P2</b>	<i>Encontré la aplicación innecesariamente compleja</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>
	45,2	35,5	9,7	3,2	6,5
<b>P3</b>	<i>Creo que la aplicación fue fácil de usar</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>
	0	0	3,2	38,7	58,1
<b>P4</b>	<i>Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar esta aplicación</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>
	41,9	35,5	12,9	3,2	6,5
<b>P5</b>	<i>Las funciones de esta aplicación están bien integradas</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>

	0	3,2	3,2	41,9	51,6
<b>P6</b>	<i>Creo que la aplicación es muy inconsistente (comportamiento incorrecto)</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>
	58,1	25,8	9,7	3,2	3,2
<b>P7</b>	<i>Me imagino que la mayoría de los usuarios aprendería a usar esta aplicación de forma muy rápida</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>
	0	0	3,2	45,2	51,6
<b>P8</b>	<i>Encuentro que la aplicación es muy difícil de usar.</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>
	51,6	35,5	6,5	3,2	3,2
<b>P9</b>	<i>Me sentí muy seguro usando la aplicación</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>

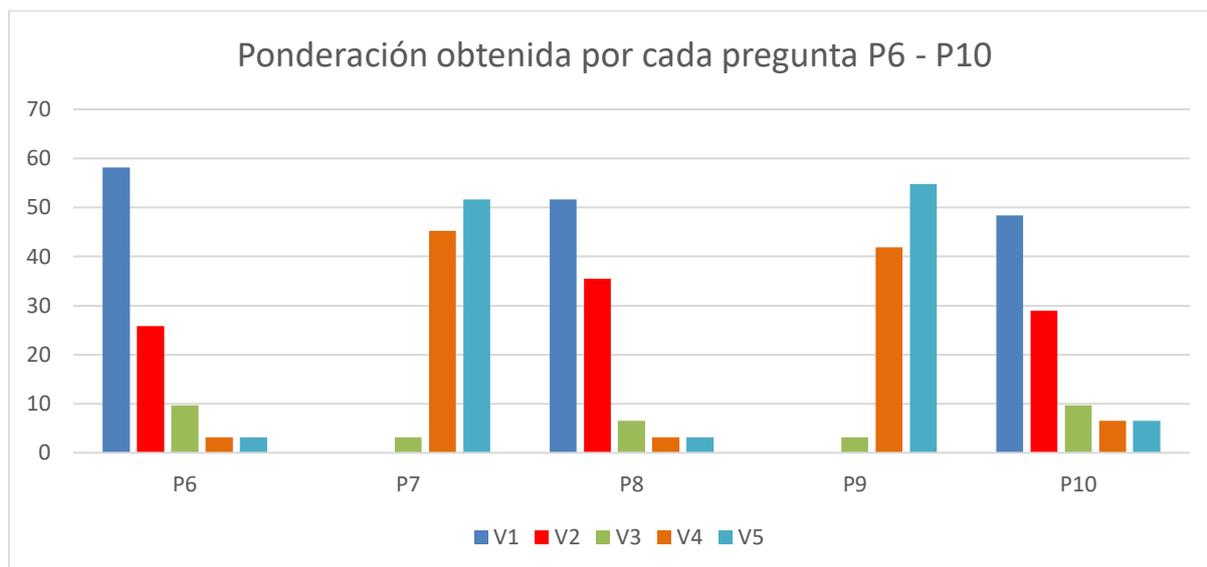
	0	0	3,2	41,9	54,8
<b>P10</b>	<i>Necesité aprender muchas cosas antes de usar esta aplicación</i>				
	<b>Criterio de Ponderación</b>				
	<b>Totalmente en desacuerdo (%)</b>	<b>En desacuerdo (%)</b>	<b>Neutro (%)</b>	<b>De acuerdo (%)</b>	<b>Totalmente de acuerdo (%)</b>
	48,4	29	9,7	6,5	6,5

Número de pregunta (N.P)

En las figuras 26 y 27 se muestran los resultados obtenidos por cada pregunta, en valor porcentual, acorde a la escala de Likert.



**Figura 26.** Resultados obtenidos P1 - P5.



**Figura 27.** Resultados obtenidos P6-P10.

En la tabla 15, se muestra los resultados obtenidos de los 31 encuestados, así como la media final obtenida de forma general. En el Anexo 4 se detallan los resultados de cada sujeto encuestado.

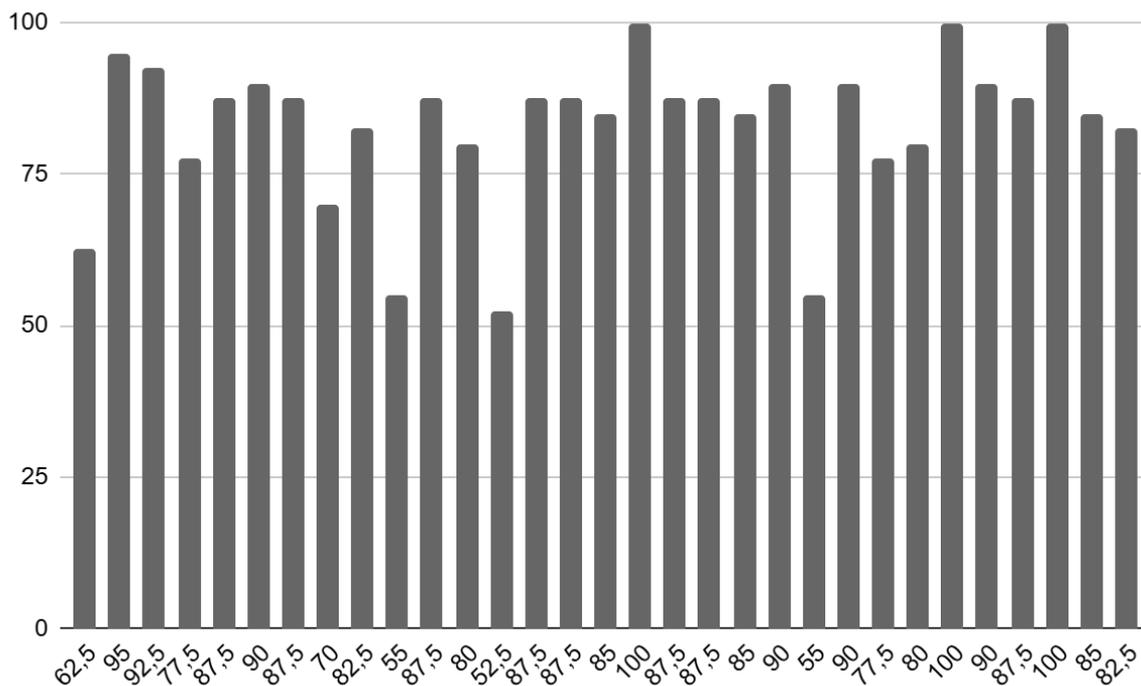
**Tabla 11** Resultados obtenidos de los 31 encuestados.

N.E.	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	N.P	M.C
<b>E1</b>	4	3	4	3	4	3	4	3	4	3	62,5	D
<b>E2</b>	5	1	5	1	4	1	5	1	4	1	95	A
<b>E3</b>	5	1	5	1	4	1	4	1	4	1	92,5	A
<b>E4</b>	4	2	4	1	4	2	4	1	4	3	77,5	B
<b>E5</b>	4	2	5	1	4	1	4	1	4	1	87,5	A
<b>E6</b>	5	1	5	2	5	1	4	2	5	2	90	A
<b>E7</b>	5	2	5	2	4	1	4	2	5	1	87,5	A
<b>E8</b>	5	3	4	3	5	3	5	3	4	3	70	C
<b>E9</b>	4	1	5	2	5	2	4	2	4	2	82,5	A

<b>N.E.</b>	<b>P1</b>	<b>P2</b>	<b>P3</b>	<b>P4</b>	<b>P5</b>	<b>P6</b>	<b>P7</b>	<b>P8</b>	<b>P9</b>	<b>P10</b>	<b>N.P</b>	<b>M.C</b>
<b>E10</b>	3	3	3	3	3	3	3	2	3	2	55	D
<b>E11</b>	4	1	4	2	5	1	5	2	4	1	87,5	A
<b>E12</b>	5	1	5	5	5	1	5	1	5	5	80	A
<b>E13</b>	4	5	4	5	4	4	4	4	5	2	52,5	D
<b>E14</b>	4	2	5	1	4	1	4	2	5	1	87,5	A
<b>E15</b>	5	2	4	1	4	1	5	2	4	1	87,5	A
<b>E16</b>	5	1	4	3	4	2	4	1	5	1	85	A
<b>E17</b>	5	1	5	1	5	1	5	1	5	1	100	A
<b>E18</b>	5	2	5	2	5	2	5	2	5	2	87,5	A
<b>E19</b>	5	2	5	2	5	2	5	2	5	2	87,5	A
<b>E20</b>	4	1	5	2	4	2	4	1	5	2	85	A
<b>E21</b>	4	1	5	2	4	2	5	1	5	1	90	A
<b>E22</b>	5	5	5	4	5	5	5	5	5	4	55	D
<b>E23</b>	4	2	4	1	5	1	5	1	4	1	90	A
<b>E24</b>	4	2	4	2	5	1	5	2	5	5	77,5	B
<b>E25</b>	5	1	4	2	2	1	4	2	4	1	80	A
<b>E26</b>	5	1	5	1	5	1	5	1	5	1	100	A
<b>E27</b>	4	2	5	1	5	2	5	1	5	2	90	A
<b>E28</b>	4	1	4	1	4	1	4	1	4	1	87,5	A
<b>E29</b>	5	1	5	1	5	1	5	1	5	1	100	A
<b>E30</b>	4	2	4	1	5	1	4	1	4	2	85	A

N.E.	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	N.P	M.C
E31	5	4	5	2	5	1	5	1	5	4	82,5	A
<b>PUNTACIÓN TOTAL PROMEDIO:</b>											<b>83,1</b>	<b>A</b>

En la figura 28, se muestra la nota total obtenida de los 31 encuestados, por medio de la cual se pudo determinar el puntaje final obtenido en la ejecución de las pruebas, el cuál arrojó un valor promedio del 83,1 en la escala de SUS.



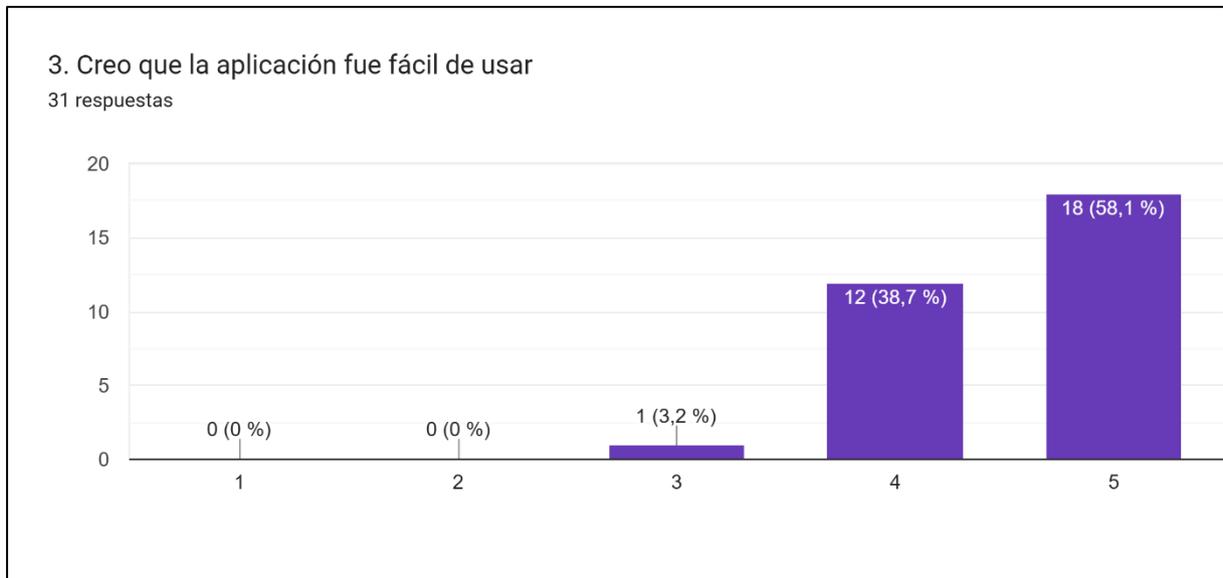
**Figura 28.** Puntajes obtenidos por cada evaluador.

En la tabla 16 se presenta la nota obtenida, así como la calificación asignada, correspondiente a los criterios de calificación; estos resultados permitieron determinar que la aplicación web es un producto altamente aceptable (A), con un puntaje promedio de 83,1.

**Tabla 12.** Resultados de la evaluación

Total encuestados	Promedio de puntaje obtenido	Media de calificación
31	83,1	A

Este puntaje representó, además, la satisfacción en usabilidad por parte los usuarios, lo cual se pudo evidenciar, en la pregunta 3 de la encuesta (ver figura 29), la cual indica que el 58,1% de los usuarios encontraron que la aplicación web es fácil de usar, y un 38,7% señaló estar de acuerdo en que la aplicación web es de fácil uso.



**Figura 29.** Resultados de la pregunta 3 de usabilidad.

## 7. Discusión

En esta sección se describe la discusión de cada uno de los objetivos del presente trabajo de integración curricular, los cuales sirvieron para el desarrollo y ejecución del mismo. En la **sección 7.1** se describe la discusión del primer objetivo específico, mientras que en la **sección 7.2** se aborda el segundo objetivo específico.

### **7.1 Objetivo 1: Construir la herramienta web para la gestión de requisitos de software basado en el modelado de software aplicando la metodología ICONIX.**

El uso de la metodología ICONIX acorde al estudio [38], resultó favorable para el desarrollo del presente trabajo de integración curricular, ya que facilitó el modelado de la arquitectura y su documentación mínima necesaria, además de poder realizar cambios durante el desarrollo de las actividades, permitiendo hacer las correcciones y mejoras necesarias.

Durante la fase de Análisis preliminar y diseño de los requerimientos, se partió con la elaboración de la entrevista (ver **Anexo 1**), similar al estudio [5], lo que permitió identificar las necesidades de los usuarios. Esta técnica resulta muy útil para realizar un contraste de los problemas que pueden identificar los usuarios, y plantear así posibles soluciones a los mismos [47]. Con los resultados de esta encuesta, se procedió a elaborar el documento de especificación de requisitos de software (ver **Anexo 2**), mediante el estándar IEE830, el cuál es una guía estructurada para la especificación y documentación de los requisitos de software [11]. En este documento se detallan los requisitos funcionales y no funcionales identificados a partir de las encuestas realizadas.

A partir de este documento de especificación de requisitos, se elaboró el modelo de dominio de negocio y modelo de usuarios; el modelo de negocio representó las entidades y su relación dentro del contexto del negocio, lo que permitió identificar los elementos esenciales y las interacciones en el sistema (ver **sección 6.1.2**); por su parte, el modelo de usuarios describió la forma en que los usuarios interactúan con la aplicación, de acuerdo a sus roles y permisos (ver **sección 6.1.3**).

Durante la fase de diseño de la metodología ICONIX, se llevó a cabo el diseño de la arquitectura de la aplicación, lo cual es una parte fundamental, tal como lo resalta el estudio [14], la cual se estructuró en: arquitectura física (ver **sección 6.1.4.1**) y arquitectura lógica (ver **sección 6.1.4.2**). En la arquitectura física se diseñaron los siguientes diagramas: diagrama de componentes y diagrama de despliegue, el diagrama de componentes permitió visualizar la

relación entre los componentes de software, así como la estructura de la aplicación, tanto del backend, como del frontend, y la comunicación con la base de datos. Por su parte, el diagrama de despliegue permitió describir la distribución física de los componentes de la aplicación, especificando las conexiones de los nodos y su interacción entre sí, de esta manera, se pudo representar de manera óptima el entorno físico y de ejecución de la aplicación.

En el diseño de la arquitectura lógica se llevaron a cabo la elaboración de los diagramas de casos de uso (ver sección **6.1.4.2.1**), el cual permitió visualizar donde los actores participan con los escenarios del sistema para cada una de las funcionalidades que se especificaron acorde al ERS (ver **Anexo 2**). Esto fue fundamental para poder garantizar que se cumplan todas las necesidades identificadas por parte de los usuarios. Además, se realizó el prototipado de la aplicación (ver **sección 6.1.4.3**), mediante la herramienta Balsamiq, lo cuál facilitó tener una representación preliminar de la interfaz de la aplicación.

Durante la fase de codificación, se detalla estructura de la aplicación (ver **sección 6.1.5.1**), para ello se empleó la arquitectura cliente-servidor, la cual es una opción que facilitó el desarrollo y organización de la aplicación, separando la parte de las reglas de negocio (backend), de la parte de presentación (frontend). Esto permitió tener una solución robusta, escalable y mantenible [48].

En el backend se aplicó el patrón modelo – ruta – controlador, el cual permitió tener una estructura organizada del código, separando la lógica del negocio y comunicación con la base de datos (modelos), los endpoints para las solicitudes de la aplicación (rutas) y la lógica de la aplicación e interacción de rutas y modelos (controlador). Las tecnologías empleadas para el desarrollo del backend fueron: Nodejs v21.6.1, Express v4.18.2, JavaScript como lenguaje de programación, y MongoDB para la base de datos. Mientras que en el frontend se aplicó el patrón basado en componentes, el cual permitió organizar la aplicación en componentes independientes y reutilizables, similar al estudio ,

## **7.2 Objetivo 2. Evaluar la aplicación web desarrollada para determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema).**

La evaluación de la aplicación web se realizó con la finalidad de determinar el grado de aceptación, enfocado en la experiencia del usuario, aplicando la metodología SUS. De esta forma, se realizó la planificación de las pruebas (ver **Anexo 3**) aplicando la escala de Likert, similar al estudio [22]. Con el diseño de la prueba, se llevó a cabo la ejecución de las pruebas (ver **sección 6.2.2** y **Anexo 4**), las cuales se realizaron en una muestra de 31 sujetos, los cuales eran afines a la Carrera Computación.

Como resultados de la ejecución de las pruebas, se determinó una nota de **83,1/100**, reflejando un cumplimiento muy alto en aceptabilidad y facilidad de uso de la aplicación web, con una nota de **A**. De esta forma, se pudo constatar el grado de aceptación de la aplicación web mediante la experiencia del usuario.

Estos resultados permitieron dar respuesta a la pregunta de investigación planteada: ¿Cuál sería el grado de aceptación de una herramienta con entorno web para los integrantes de un equipo de desarrollo de software, con el rol de analistas de negocios, que les permita gestionar y validar los requisitos en los proyectos de software de la Carrera Computación de la UNL?, creando una aplicación web que permite a los usuarios: la gestión de requisitos de un proyecto, trazabilidad de los cambios de los requisitos, generación de documentos de especificación de requisitos (ERS), validación de documentos de elicitación, validación de técnicas o atributos de calidad.

## 8. Conclusiones

- La construcción de la herramienta web para la gestión de requisitos de software, empleando la metodología ICONIX para el modelado de software, brindó un modelado estructurado y bien organizado. Gracias a esta metodología, se logró la especificación de requisitos claros y bien definidos, y se estableció una base sólida para la solución mediante el modelo de dominio de la aplicación.
- El uso de la metodología SUS, favoreció a determinar el grado de aceptación de la aplicación web, ya que se enfocó en la usabilidad por parte del usuario (UX), con un resultado promedio de 83,1 en la escala de SUS, lo cual describe al producto como altamente aceptable, de esta manera se pudo cumplir favorablemente el desarrollo del objetivo 2.
- Ante la pregunta de investigación planteada “¿Cuál sería el grado de aceptación de una herramienta con entorno web para los integrantes de un equipo de desarrollo de software, con el rol de analistas de negocios, que les permita gestionar y validar los requisitos en los proyectos de software de la Carrera Computación de la UNL?”, se pudo determinar que la aplicación web es una solución óptima para el proceso de gestión de requisitos de software y que se cumplió con las necesidades del líder y analistas de un equipo de desarrollo de software.
- La implementación del patrón cliente-servidor, permitió desarrollar una solución robusta y escalable, garantizando la separación de responsabilidades entre las reglas de negocio y la presentación de la aplicación. Esto contribuyó al cumplimiento del objetivo 1 de manera óptima.
- El uso del patrón de diseño arquitectónico modelo-ruta-controlador, resultó beneficioso para la organización y separación de responsabilidades, favoreciendo la escalabilidad y reutilización del código. Esta elección proporcionó una estructura clara y eficiente para el desarrollo de la herramienta web.
- La implementación de esta herramienta web, favorece en el proceso de gestión de requisitos de software mediante la generación de documentos de especificación de requisitos, aplicando los estándares IEEE 830, e ISO/IEC 29148; así mismo de la trazabilidad en los cambios de los requisitos y validación de documentos de elicitación. Esto permitió adoptar un enfoque más profesional y estandarizado en la gestión de requisitos, elevando la calidad y consistencia de los proyectos.

- La escala de calificación de la metodología SUS, resultó beneficiosa de aplicar, ya que se maneja por rangos de calificación y asignación de notas, permitiendo de esta manera identificar fácilmente el grado de aceptación que tuvo la aplicación web.

## **9. Recomendaciones**

Tras la culminación del presente Trabajo de Integración Curricular, se recomienda:

- Emplear la arquitectura cliente – servidor, resulta muy favorable, para poder separar responsabilidades, tanto de las reglas del negocio, como de la parte de presentación de la aplicación, esto permite a tener un código mantenible, escalable y robusto.
- El uso de la metodología SUS para medir la aceptación de un producto software, facilita la determinación del grado de aceptación que puede tener un software, en base a la experiencia de uso del usuario, permitiendo identificar el cumplimiento de las necesidades de los usuarios.
- La aplicación de patrones de arquitectura, así como patrones de desarrollo de software, resultan convenientes para optimizar recursos de tiempo y reutilización de código, ya que al aplicar patrones se puede obtener una arquitectura bien organizada, y un código mucho más mantenible y reutilizable.

### **9.1 Trabajos Futuros**

La aplicación web para la gestión de requisitos, facilita el proceso de gestión de los requisitos de un proyecto de software, sin embargo, se pueden llevar a cabo algunas oportunidades de mejoramiento en futuras investigaciones. Por ejemplo:

- A partir del proceso que se ha implementado para la elicitación de requisitos de software, se podría incluir el proceso de descripción o generación de historias de usuario o casos de uso de los requisitos que se hayan creado.
- Se puede desarrollar la generación de casos de prueba, a partir de los requisitos generados en cada proyecto.
- De igual forma, se puede trabajar en la generación de informes ejecutivos del estado de cada proyecto, lo que permitiría tener una visión general de cada proyecto en el que se está trabajando.

## 10. Bibliografía

- [1] S. Jayatilleke y R. Lai, “A systematic review of requirements change management”, *Inf Softw Technol*, vol. 93, pp. 163–185, ene. 2018, doi: 10.1016/j.infsof.2017.09.004.
- [2] M. E. Serna y A. A. Serna, “Development and management requirements: Results of a literature review”, en *2016 8th Euro American Conference on Telematics and Information Systems, EATIS 2016*, Institute of Electrical and Electronics Engineers Inc., jul. 2016. doi: 10.1109/EATIS.2016.7520128.
- [3] R. S. Pressman, J. E. M. Murrieta, V. C. Olguin, y E. P. Rojas, *Ingeniería del software: un enfoque práctico*. McGraw-Hill, 2006. [En línea]. Disponible en: <https://books.google.com.ec/books?id=t9k6vgAACAAJ>
- [4] E. Prieto, “¿Cuáles son las etapas del desarrollo de software?”, Tiffin University. [En línea]. Disponible en: <https://global.tiffin.edu/noticias/cuales-son-las-etapas-del-desarrollo-de-software#:~:text=La%20planificaci%C3%B3n%20es%20la%20fase%20inicial%20que%20permite,Software%2C%20una%20actividad%20muy%20demandada%20en%20la%20actualidad.>
- [5] S. Bayona-Oré, J. Chamilco, y D. Perez, “Software process improvement: Requirements management, verification and validation | Mejora de Procesos Software: Gestión de Requisitos, Verificación y Validación”, *Iberian Conference on Information Systems and Technologies, CISTI*, vol. 2019-June, 2019, doi: 10.23919/CISTI.2019.8760896.
- [6] A. Smith, “Gestión de requisitos — Una competencia esencial para el éxito de proyectos y programas”, 2014. Consultado: el 20 de abril de 2024. [En línea]. Disponible en: [https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/requirements-management.pdf?sc\\_lang\\_temp=es-ES](https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/requirements-management.pdf?sc_lang_temp=es-ES)
- [7] L. E. Peláez-Valencia, A. Toro-Lazo, L. D. Ledesma-Mariño, D. E. Rodríguez-Franco, y E. H. Largo-Bueno, “Revisión de herramientas para la gestión de requisitos: un aporte al aseguramiento de la calidad del software”, *Ventana Informatica*, núm. 38, ene. 2019, doi: 10.30554/ventanainform.38.2865.2018.
- [8] D. María *et al.*, *Fundamentos de Ingeniería de Software*.

- [9] G. Méndez, “Ingeniería de Requisitos”.
- [10] Manzanelli, “La Norma ISO 29148: Garantizando Calidad en requisitos de software”, NormasISO.org. [En línea]. Disponible en: <https://normasiso.org/norma-iso-29148/>
- [11] “IEEE Guide for Software Requirements Specifications”, *IEEE Std 830-1984*, pp. 1–26, 1984, doi: 10.1109/IEEESTD.1984.119205.
- [12] N. Uke,  
 “[Http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.401.8095&rep=rep1&type=pdf](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.401.8095&rep=rep1&type=pdf)”, marzo de 2014. [En línea]. Disponible en:  
[https://www.academia.edu/6390087/http\\_citeseerx\\_ist\\_psu\\_edu\\_viewdoc\\_download\\_doi\\_10\\_1\\_1\\_401\\_8095\\_and\\_rep\\_rep1\\_and\\_type\\_pdf](https://www.academia.edu/6390087/http_citeseerx_ist_psu_edu_viewdoc_download_doi_10_1_1_401_8095_and_rep_rep1_and_type_pdf)
- [13] “ISO/IEC/IEEE International Standard - Systems and software engineering – Life cycle processes – Requirements engineering”, *ISO/IEC/IEEE 29148:2018(E)*, pp. 1–104, 2018, doi: 10.1109/IEEESTD.2018.8559686.
- [14] D. Carrizo y C. Ortiz, “Towards a context-based framework for the software requirements elicitation process management”, en *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*, 2018, pp. 1–7. doi: 10.23919/CISTI.2018.8399371.
- [15] M. F. Rosique Contreras, M. Jiménez Buendía, P. Sánchez Palma, y others, “Evaluación de herramientas de gestión de requisitos”, 2010.
- [16] S. Shams Tabrez, A. Md Oqail, S. Mohammad, y R. S. Gulzar Ahmed, “Cloud Based Requirements Management Tools Should Be Selected on the Basis of Project Requirements”.
- [17] S. Gonzalez, “¿Qué es la experiencia de usuario?”, abril de 2020. [En línea]. Disponible en: <https://www.cyberclick.es/que-es/experiencia-de-usuario#:~:text=La%20experiencia%20de%20usuario%2C%20tambi%C3%A9n%20llamada%20user%20experience,a%20la%20interacci%C3%B3n%20con%20p%C3%A1ginas%20web%20y%20aplicaciones>.
- [18] Y. H. Montero, “Experiencia de Usuario: Principios y Métodos”, 2015. [En línea]. Disponible en:  
[www.yusef.es](http://www.yusef.es) Todos los derechos reservados, 2015 EstelibrohasidoescritousandoCalmly  
 Writer

- [19] H.-W. Jung, S.-G. Kim, y C.-S. Chung, “Measuring software product quality: a survey of ISO/IEC 9126”, *IEEE Softw*, vol. 21, núm. 5, pp. 88–92, 2004, doi: 10.1109/MS.2004.1331309.
- [20] L. R. Baquero Hernández, R. Valdés, y C. Ricardo, “Lógica Difusa Basada en la Experiencia del Usuario para Medir la Usabilidad”, 2016.
- [21] H.-M. Yusef y F. Martin, “Diseño Web Centrado en el Usuario: Usabilidad y Arquitectura de la Información”, 2015. [En línea]. Disponible en: <http://www.hipertext.net>
- [22] M. Aguilar y A. Villegas, “Análisis comparativo de la Escala de Usabilidad del Sistema (EUS) en dos versiones”, vol. 5, jul. 2016.
- [23] “Ergonomics of human-system interaction-Part 11: Usability: Definitions and concepts COPYRIGHT PROTECTED DOCUMENT”, 2018. [En línea]. Disponible en: <https://standards.iteh.ai/catalog/standards/sist/d38dc274-d8d4-4fb9-8206->
- [24] A. Bangor, P. T. Kortum, y J. T. Miller, “An empirical evaluation of the system usability scale”, *Int J Hum Comput Interact*, vol. 24, núm. 6, pp. 574–594, ago. 2008, doi: 10.1080/10447310802205776.
- [25] A. Bangor, P. Kortum, y J. Miller, “Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale”, 2009.
- [26] S. Serafinelli, “Qué Es la escala sus y cómo usarla para medir la usabilidad”, marzo de 2024. [En línea]. Disponible en: <https://www.teacuplab.com/es/blog/que-es-la-escala-sus-y-como-usarla-para-medir-la-usabilidad/>
- [27] T. Navarrete, “El lenguaje Javascript Pág. 1 Programación III”.
- [28] J. Eguíluz Pérez, *Introducción a JavaScript*. 2009. [En línea]. Disponible en: [www.librosweb.es](http://www.librosweb.es)
- [29] Z. Dinku, “React.js vs. Next.js”, 2022.
- [30] A. Gamela, “Next JS vs react: What are the differences?”, Blog | Imaginary Cloud. [En línea]. Disponible en: <https://www.imaginarycloud.com/blog/next-js-vs-react/>
- [31] E. Haro, T. Guarda, A. O. Zambrano Peñaherrera, y G. Ninahualpa Quiña, “Desarrollo backend para aplicaciones web, Servicios Web Restful: Node.js vs Spring Boot”.

- [32] E. Haro, T. Guarda, A. O. Z. Peñaherrera, y G. N. Quiña, “Desarrollo backend para aplicaciones web, servicios web restful: Node.js vs spring boot”, *Revista Ibérica de Sistemas e Tecnologías de Informação*, núm. E17, pp. 309–321, 2019.
- [33] K. Banker, P. Bakkum, S. Verch, D. Garrett, y T. Hawkins, *MongoDB in Action: Covers MongoDB version 3.0*, 2nd ed. USA: Manning Publications Co., 2016.
- [34] Ш. Брэдшоу, И. Брэзил, y К. Ходоров, *MongoDB - La guía completa*. 2020.
- [35] A. Saiz, “Construcción de entorno DevOps mediante arquitectura de contenedores de software”, Universidad de Cantabria, Cantabria, 2021.
- [36] D. A. Hurtado Chichande, “Tecnología de contenedores de software en entornos de pruebas”, Ecuador-PUCESE-Escuela de Sistemas y Computación, 2018.
- [37] P. Ponsico Martin, “Tecnología de Contenedores Docker”, UPC, Escola Tècnica Superior d’Enginyeria de Telecomunicació de Barcelona, 2017. [En línea]. Disponible en: <http://hdl.handle.net/2117/113040>
- [38] A. El-Abbassy y M. El-Zeweidy, “ICONIX Approach to MVC: Applying Robustness Analysis on the Model–View–Controller Architecture”, *Journal of the ACS*, vol. 7, 2013.
- [39] E. E. Porras Flores, “Metodología ágil Iconix en la calidad del producto software, Lima, 2017”, 2019.
- [40] M. L. López Martínez, “Herramienta de apoyo para el aprendizaje del modelo de dominio de la metodología Iconix”, 2009.
- [41] S. Silva Ascuntar y Y. G. Carolina García, “Documentación de la Metodología ICONIX a través del desarrollo del caso ‘Oriéntate CALI’”, 2018.
- [42] M. Vega, “Casos de uso”, *Recuperado el*, vol. 12, 2010.
- [43] C. L. Vidal, R. F. Schmal, S. Rivero, y R. H. Villarroel, “Extensión del Diagrama de Secuencias UML (Lenguaje de Modelado Unificado) para el Modelado Orientado a Aspectos”, *Información tecnológica*, vol. 23, núm. 6, pp. 51–62, 2012.
- [44] A. R. Sartorio y M. Banega, “Enseñanza del diseño de aplicaciones web mediante el proceso Iconix adaptado el framework Odo”, en *XXIX Congreso Argentino de Ciencias de la Computación (CACIC)(Luján, 9 al 12 de octubre de 2023)*, 2024.

- [45] R. S. Casa Toapanta y M. A. Pilatasig Chiluisa, “Desarrollo e implementación de un sistema para el control de inventarios en la empresa savrehsa de la ciudad de Latacunga, mediante el análisis de la metodología iconix”, LATACUNGA/UTC/2015, 2015.
- [46] Inam-Ul-Haq, W. Abbas, y W. H. Butt, “Systematic Literature Review on Requirement Management Tools”, en *2022 International Conference on Emerging Trends in Smart Technologies, ICETST 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICETST55735.2022.9922932.
- [47] H. L. Romo y others, “La metodología de la encuesta”, *Técnicas de investigación en sociedad, cultura y comunicación*, pp. 33–74, 1998.
- [48] H. S. Oluwatosin, “Client-server model”, *IOSR J Comput Eng*, vol. 16, núm. 1, pp. 67–71, 2014.

## 11. Anexos

### Anexo 1. Encuesta para la situación problemática

En el siguiente anexo se presenta los resultados de la encuesta realizada a los estudiantes de la Carrera Computación de la Universidad Nacional de Loja, la cual tuvo como objetivo identificar las necesidades y problemas presentados por los estudiantes para la gestión de requisitos de software.

## UNIVERSIDAD NACIONAL DE LOJA CARRERA DE COMPUTACIÓN Proyectos Tecnológicos

**Nombre:** Edinson Quizhpe

**Docente:** Ing. Pablo Ordoñez Mg. Sc.

### 1. Planificación

#### **Problema central:**

Los estudiantes de la Carrera de Computación enfrentan dificultades en la captura y documentación de requisitos en los proyectos académicos, lo cual causa la formulación de requisitos ambiguos, mal definidos o incompletos, lo que conduce a una falta de comprensión sobre la IR en los proyectos que realizan.

**Muestra:** La encuesta realizada fue dirigida a los estudiantes de la Carrera de Computación. Preguntas de la encuesta:

- 1. ¿Consideras que la captura de requisitos es una etapa importante en el desarrollo de proyectos?**
  - a) Sí, es fundamental.
  - b) Sí, pero no es prioritaria.
  - c) No, no es necesaria.
  - d) No estoy seguro.
- 2. ¿Cuál de los siguientes factores consideras más difícil al momento de capturar requisitos?**

- a) Comunicación efectiva con los clientes o usuarios.
- b) Identificación de requisitos funcionales y no funcionales.
- c) Gestión de cambios en los requisitos.
- d) Otras dificultades (especificar).

**3. ¿Qué técnica de captura de requisitos consideras más efectiva?**

- a) Entrevistas con los clientes o usuarios.
- b) Observación del entorno de trabajo.
- c) Cuestionarios o encuestas.
- d) Otras técnicas (especificar).

**4. ¿Cómo sueles abordar la gestión de cambios en los requisitos durante el desarrollo de proyectos académicos?**

- a) Realizo revisiones frecuentes de los requisitos y actualizo la documentación según sea necesario.
- b) No suelo gestionar los cambios en los requisitos.
- c) Discuto los cambios con el cliente o usuario antes de implementarlos.
- d) Otras estrategias (especificar).

**5. ¿Consideras que la falta de comprensión sobre la gestión de requisitos puede afectar negativamente el éxito de los proyectos académicos?**

- a) Sí, totalmente.
- b) Sí, en cierta medida.
- c) No, no creo que sea relevante.
- d) No estoy seguro.

**6. ¿Qué método utilizas con mayor frecuencia para documentar requisitos en proyectos académicos?**

- a) Diagramas de casos de uso.
- b) Diagramas de flujo.

- c) Descripción textual detallada.
- d) Otros métodos (especificar).

**7. ¿En qué medida utilizas técnicas de validación de requisitos en tus proyectos?**

- a) Siempre realizo pruebas y validaciones exhaustivas.
- b) Realizo algunas validaciones, pero no siempre de manera exhaustiva.
- c) No suelo realizar validaciones de requisitos.
- d) No estoy seguro.

**8. ¿Qué aspecto o habilidad relacionada con la captura y documentación de requisitos consideras que debería ser reforzada en la formación académica de la carrera de Computación?**

- a) Comunicación efectiva con los clientes o usuarios.
- b) Técnicas de entrevista y observación.
- c) Habilidades de documentación y especificación.
- d) Otras habilidades (especificar).

**9. ¿Estás familiarizado con el estándar IEEE 830 para la especificación de requisitos de software?**

- a) Sí, tengo un conocimiento sólido sobre el estándar IEEE 830.
- b) Sí, he escuchado sobre el estándar IEEE 830, pero no tengo un conocimiento profundo.
- c) No, no estoy familiarizado con el estándar IEEE 830.
- d) No estoy seguro.

**10. ¿Has utilizado el estándar IEEE 830 para la especificación de requisitos en tus proyectos académicos o personales?**

- a) Sí, lo he utilizado de manera frecuente y adecuada.
- b) Sí, pero solo en ocasiones puntuales.
- c) No, nunca lo he utilizado.
- d) No estoy seguro.

**11. ¿Cuál es el propósito principal del estándar IEEE 830 en relación con la especificación de requisitos?**

- a) Establecer una estructura y formato uniforme para la documentación de requisitos.
- b) Proporcionar directrices específicas para la captura de requisitos en proyectos académicos.
- b) Definir una metodología para el desarrollo de software.
- c) No estoy seguro.

**12. ¿Cuál de los siguientes elementos crees que deberían incluirse en una especificación de requisitos según el estándar IEEE 830? (Selecciona todas las opciones que apliquen).**

- a) Descripción del sistema y sus componentes.
- b) Requisitos funcionales y no funcionales.
- c) Casos de uso y escenarios de interacción.
- d) Restricciones de diseño y de implementación.
- e) Otros elementos (especificar).

**13. ¿Cómo consideras que el conocimiento y aplicación del estándar IEEE 830 puede beneficiar la captura y documentación de requisitos en proyectos académicos?**

- a) Facilita la comprensión y comunicación entre el equipo de desarrollo y los clientes o usuarios.
- b) Establece una base sólida para la gestión de cambios y el control de calidad en los requisitos.
- c) No considero que el estándar IEEE 830 sea relevante para proyectos académicos.
- d) No estoy seguro.

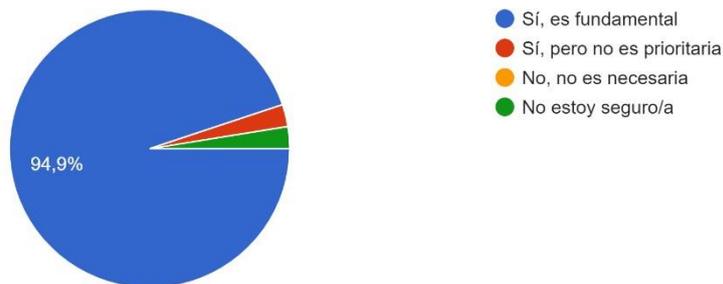
## **2. Ejecución**

Estimado/a estudiante de la carrera de Computación de la Universidad Nacional de Loja, la presente encuesta tiene como objetivo conocer su opinión y experiencia en relación con la captura y documentación de requisitos de software. Solicito comedidamente responder la presente encuesta con el fin de obtener los datos necesarios para el desarrollo del Proyecto de Integración Curricular.

### Pregunta 1

¿Consideras que la gestión de requisitos de software es una etapa importante en el desarrollo de proyectos?

39 respuestas

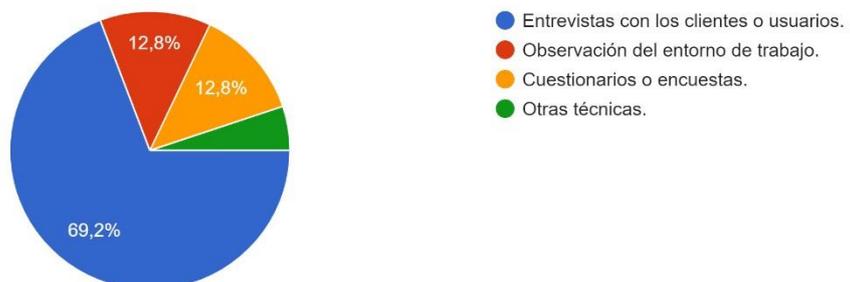


### Pregunta 2

### Pregunta 3

¿Qué técnica de captura de requisitos consideras más efectiva?

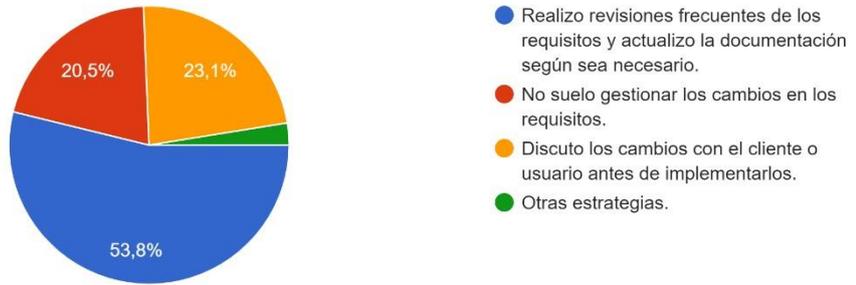
39 respuestas



#### Pregunta 4

¿Cómo sueles abordar la gestión de cambios en los requisitos durante el desarrollo de proyectos académicos?

39 respuestas



#### Pregunta 5

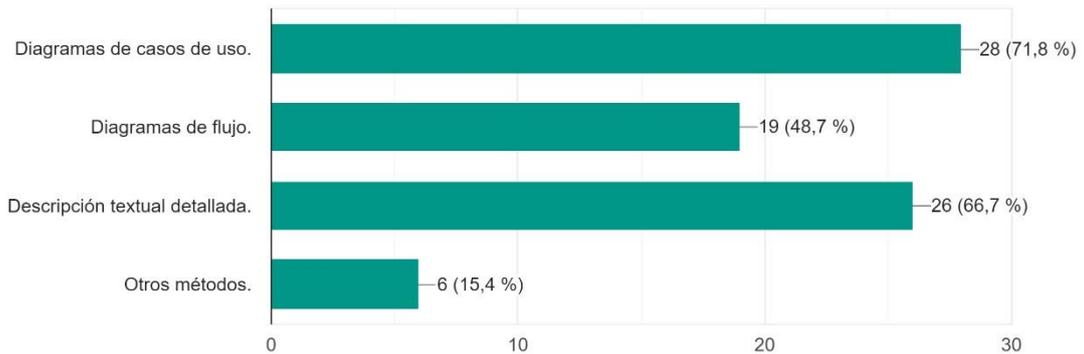
¿Consideras que la falta de comprensión sobre la gestión de requisitos puede afectar negativamente el éxito de los proyectos académicos?

39 respuestas



¿Qué método/s usas para documentar requisitos en sus proyectos?

39 respuestas



Pregunta 6:

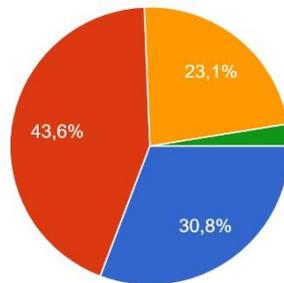
**Pregunta 7:**

**Pregunta 8:**

**Pregunta 9:**

¿Estás familiarizado con el estándar IEEE 830 para la especificación de requisitos de software?

39 respuestas



- Sí, tengo un conocimiento sólido sobre el estándar IEEE 830.
- Sí, he escuchado sobre el estándar IEEE 830, pero no tengo un conocimiento profundo.
- No, no estoy familiarizado con el estándar IEEE 830.
- No estoy seguro.

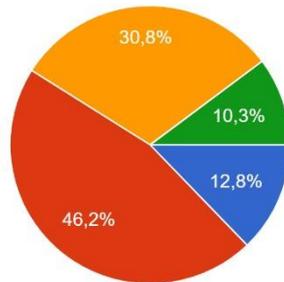
¿Que aspecto o habilidad relacionada con la captura y documentación de requisitos consideras que debería ser reforzada en la formación académica de la carrera de Computación?

39 respuestas

¿Has utilizado el estándar IEEE 830 para la especificación de requisitos en tus proyectos académicos o personales?

39 respuestas

He



- Sí, lo he utilizado de manera frecuente y adecuada.
- Sí, pero solo en ocasiones puntuales.
- No, nunca lo he utilizado.
- No estoy seguro.

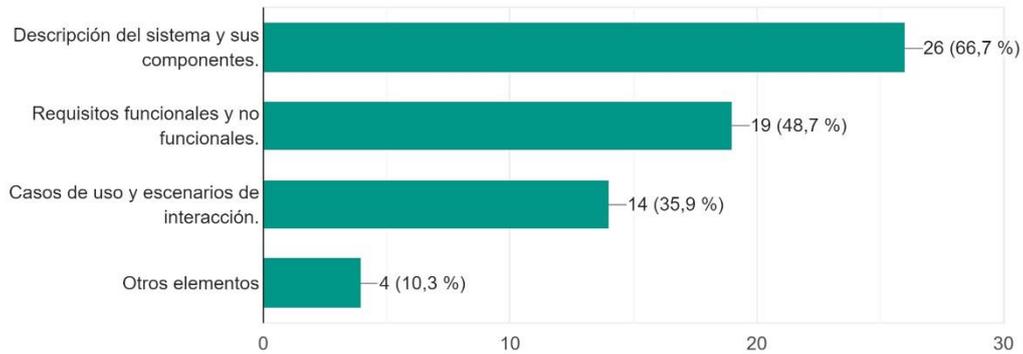
**Pregunta 10:**

### Pregunta 11:

### Pregunta 12:

¿Cuál de los siguientes elementos crees que deberían incluirse en una especificación de requisitos según el estándar IEEE 830?

39 respuestas



¿Cómo consideras que el conocimiento y aplicación del estándar IEEE 830 puede beneficiar la captura y documentación de requisitos en proyectos académicos?

39 respuestas



### Pregunta 13:

### 3. Análisis de la encuesta realizada:

PREGUNTA	ANÁLISIS
<p><b>¿Consideras que la gestión de requisitos de software es una etapa importante en el desarrollo de proyectos?</b></p>	<p>En base a la encuesta realizada, se puede determinar que los estudiantes de la carrera de Computación, consideran fundamental la gestión de requisitos como una etapa para el desarrollo de proyectos. El 94,7% considera que si, es fundamental, mientras que el 2,6% considera que, si es una etapa importante, pero no es prioritaria para el desarrollo de los proyectos; el otro 2,6% señala que no están seguros de si es una etapa importante para el desarrollo de proyectos.</p>
<p><b>¿Cuál de los siguientes factores consideras más difícil al momento de capturar requisitos?</b></p>	<p>Los resultados de esta pregunta, señalan que la mayoría (55,3%) consideran la comunicación efectiva con los clientes o usuarios, como el factor más difícil para la captura de requisitos; el 34,2% encuentran problemas en la identificación de requisitos funcionales y no funcionales, mientras que un 7,9% hallan dificultades cuando se presentan cambios en los requisitos.</p>
<p><b>¿Qué técnica de captura de requisitos consideras más efectiva?</b></p>	<p>En base a la encuesta realizada, las entrevistas con los clientes o usuarios es considera la técnica más efectiva (68,4%) para la captura de requisitos. También se debe señalar que tanto la observación del entorno de trabajo y la elaboración de encuestas o cuestionarios son técnicas importantes para la captura de requisitos, representando cada una de ellas el 13,2%.</p>

<p><b>¿Cómo sueles abordar la gestión de cambios en los requisitos durante el desarrollo de proyectos académicos?</b></p>	<p>En base a la encuesta realizada, se puede determinar que la mayoría (52,6%) de estudiantes realiza revisiones frecuentes de los requisitos, y actualiza la documentación si es necesario cuando se presentan cambios, esto es un factor positivo ya que gestionar de manera adecuada los cambios que se presenten, ayudará a tener un producto final de mejor calidad y alineado a las necesidades del cliente. Otra de las formas en que se abordan la gestión de cambios de acuerdo a la encuesta es que 23,7% es la discusión directa con el cliente o usuario antes de la implementación. Finalmente, un 21,1% indicó que no suele realizar una gestión de cambios en los requisitos, esto no es bueno para el desarrollo de proyectos y es algo que se debe abordar para mejorar.</p>
<p><b>¿Consideras que la falta de comprensión sobre la gestión de requisitos puede afectar negativamente el éxito de los proyectos académicos?</b></p>	<p>Los resultados de esta pregunta, indican que la mayoría (71,1%) de estudiantes si consideran que el no tener una comprensión sobre cómo gestionar los requisitos de software, tendrá un impacto negativo en el desarrollo de los mismos, por lo cual es fundamental abordar la gestión de requisitos en los proyectos. También el 26,3% considera que, si tiene un impacto en el desarrollo de los proyectos, sin embargo, no totalmente, si no en cierta medida, es decir que pueden haber otros factores que influyan en el éxito de un proyecto.</p>
<p><b>¿Qué método/s usas para documentar requisitos en sus proyectos?</b></p>	<p>Los métodos usados por los estudiantes son los diagramas de casos de uso 71,1%, descripción textual detallada 68,4% y diagramas de flujo 50%, siendo la más empleada los diagramas de casos de uso.</p> <p>Por otra parte, 15,8% emplea otras técnicas para la documentación de requisitos en sus proyectos.</p>

<p><b>¿En qué medida utilizas técnicas de validación de requisitos en tus proyectos?</b></p>	<p>En base a la encuesta realizada, solo el 17,9% realiza pruebas y validaciones exhaustivas de los requisitos; la gran mayoría (51,3%) realiza validaciones, pero no de manera exhaustiva.</p> <p>El 23,1% de los encuestados no realiza validaciones de los requisitos, mientras que un 7,7% no esta seguro de si realiza o no esta validación.</p>
<p><b>¿Qué aspecto o habilidad relacionada con la captura y documentación de requisitos consideras que debería ser reforzada en la formación académica de la carrera de Computación?</b></p>	<p>Dentro de las habilidades/técnicas que se podrían reforzar dentro de la carrera de Computación, el 64,1% considera la comunicación efectiva con los usuarios/clientes, el 61,5% en técnicas de entrevista y observación, mientras que el 51,3% en habilidades de documentación y especificación de requisitos.</p>
<p><b>¿Estás familiarizado con el estándar IEEE 830 para la especificación de requisitos de software?</b></p>	<p>En base a la encuesta, solo el 30,8% de los estudiantes tiene un conocimiento sólido sobre el estándar IEEE 830, la mayoría (43,6%) posee un conocimiento general de este estándar, mientras que el 23,1% no se encuentra familiarizado con este estándar, lo que refleja que existe una falta de conocimiento sobre este estándar en la Carrera de Computación.</p>
<p><b>¿Has utilizado el estándar IEEE 830 para la especificación de requisitos en tus proyectos académicos o personales?</b></p>	<p>El uso del estándar IEEE 830 por parte de los estudiantes de la Carrera de Computación se refleja en:</p> <p>Solo el 12,8% de estudiantes lo usan de manera frecuente en cada proyecto que desarrollan.</p> <p>46,2% lo ha usado solo en situaciones o proyectos específicos.</p> <p>30,8% no ha usado nunca este estándar.</p>
<p><b>¿Cuál es el propósito principal del estándar IEEE 830 en relación</b></p>	<p>Esta pregunta se desarrollo para determinar si se conoce por parte de los estudiantes, el propósito del estándar en relación con la especificación de requisitos, concluyendo que:</p>

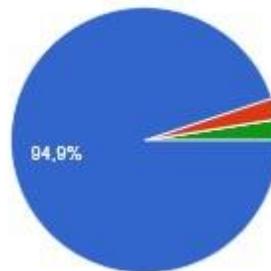
<p><b>con la especificación de requisitos?</b></p>	<p>La mayoría (53,8%) tiene claro el objetivo del mismo, que es establecer una estructura y formato uniforme para la documentación de los requisitos.</p> <p>El 28,2% no tiene definido el objetivo del mismo.</p> <p>El 12,8% considera que el objetivo es brindar directrices específicas para la captura de requisitos en los proyectos, lo cual es una respuesta general, ya que entraría dentro del objetivo general, de establecer una estructura y formato para la especificación de requisitos.</p>
<p><b>¿Cuál de los siguientes elementos crees que deberían incluirse en una especificación de requisitos según el estándar IEEE 830?</b></p>	<p>Dentro de los elementos que se deberían considerar de acuerdo al estándar IEEE 830 para la especificación de requisitos, el 66,7% considera que debe ser la descripción del sistema y de cada uno de sus componentes; el 48,7% en cambio considera a los requisitos funcionales y no funcionales, y el 35,9% los casos de uso y escenarios de interacción, esto refleja que estos tres elementos son importantes para la especificación de requisitos.</p>
<p><b>¿Cómo consideras que el conocimiento y aplicación del estándar IEEE 830 puede beneficiar la captura y documentación de requisitos en proyectos académicos?</b></p>	<p>Esta pregunta se planteó para identificar que beneficios puede brindar el uso del estándar IEEE 830 en los estudiantes de la carrera para la especificación de requisitos en sus proyectos, concluyendo que:</p> <p>El 59% considera que beneficiaría en facilitar el conocimiento y comunicación entre el equipo de desarrollo y los clientes o usuarios.</p> <p>El 23,1% encuentra el beneficio en establecer una base sólida para la gestión de cambios y control de calidad de los requisitos.</p> <p>Existe desconocimiento en un 17,9%, ya que no pueden determinar los beneficios que puede aportar este estándar, esto puede ser a causa de falta de uso del estándar, falta de comprensión de los requerimientos, entre otros.</p>

**Gráficas de los resultados de las encuestas:**

¿Consideras que la gestión de requisitos de software es una etapa importante en el desarrollo de proyectos?

[Copiar](#)

39 respuestas

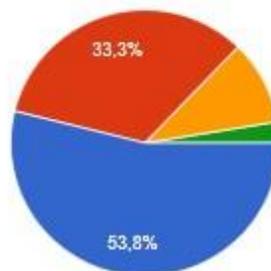


- Sí, es fundamental
- Sí, pero no es prioritaria
- No, no es necesaria
- No estoy seguro/a

¿Cuál de los siguientes factores consideras más difícil al momento de capturar requisitos?

[Copiar](#)

39 respuestas

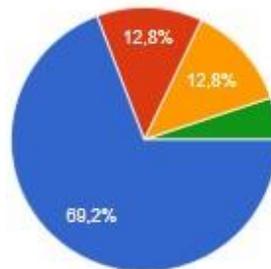


- Comunicación efectiva con los clientes o usuarios.
- Identificación de requisitos funcionales y no funcionales.
- Gestión de cambios en los requisitos.
- Otras dificultades.

### ¿Qué técnica de captura de requisitos consideras más efectiva?

 Copiar

39 respuestas

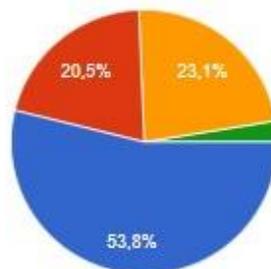


- Entrevistas con los clientes o usuarios.
- Observación del entorno de trabajo.
- Cuestionarios o encuestas.
- Otras técnicas.

### ¿Cómo sueles abordar la gestión de cambios en los requisitos durante el desarrollo de proyectos académicos?

 Copiar

39 respuestas

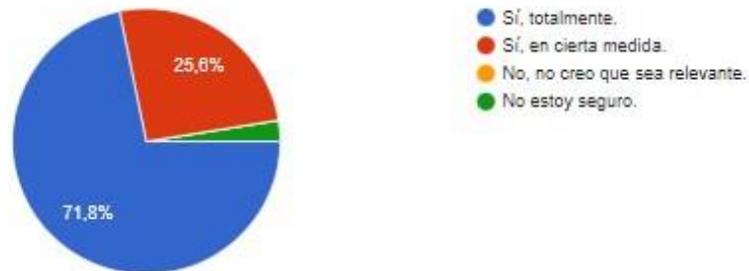


- Realizo revisiones frecuentes de los requisitos y actualizo la documentación según sea necesario.
- No suelo gestionar los cambios en los requisitos.
- Discuto los cambios con el cliente o usuario antes de implementarlos.
- Otras estrategias.

¿Consideras que la falta de comprensión sobre la gestión de requisitos puede afectar negativamente el éxito de los proyectos académicos?

 Copiar

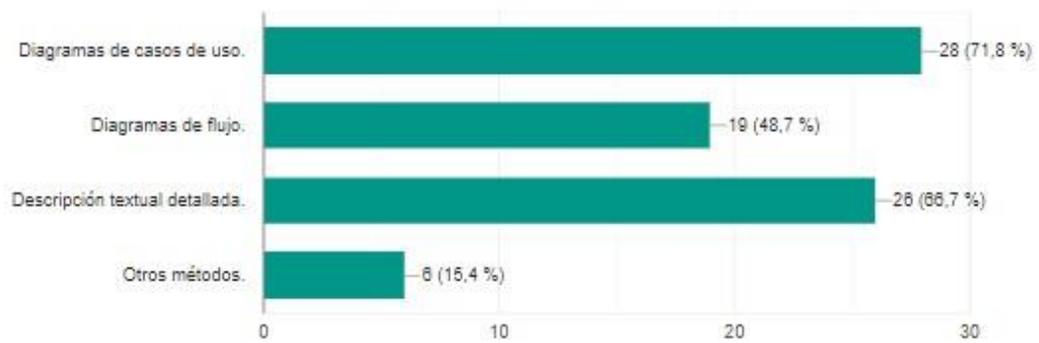
39 respuestas



¿Qué método/s usas para documentar requisitos en sus proyectos?

 Copiar

39 respuestas



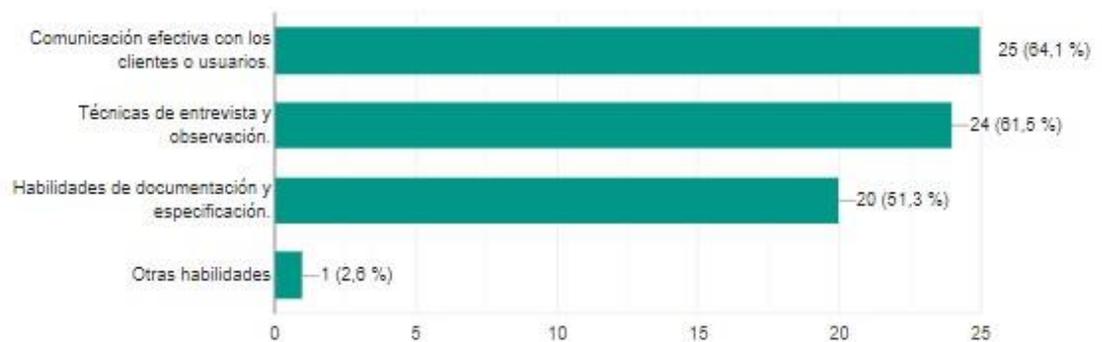
### ¿En qué medida utilizas técnicas de validación de requisitos en tus proyectos?

39 respuestas



### ¿Qué aspecto o habilidad relacionada con la captura y documentación de requisitos consideras que debería ser reforzada en la formación académica de la carrera de Computación?

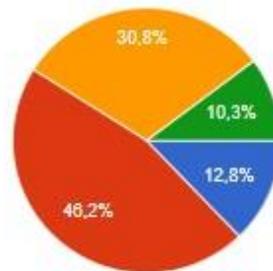
39 respuestas



### ¿Has utilizado el estándar IEEE 830 para la especificación de requisitos en tus proyectos académicos o personales?

[Copiar](#)

39 respuestas

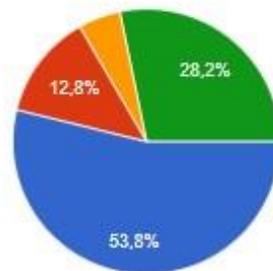


- Sí, lo he utilizado de manera frecuente y adecuada.
- Sí, pero solo en ocasiones puntuales.
- No, nunca lo he utilizado.
- No estoy seguro.

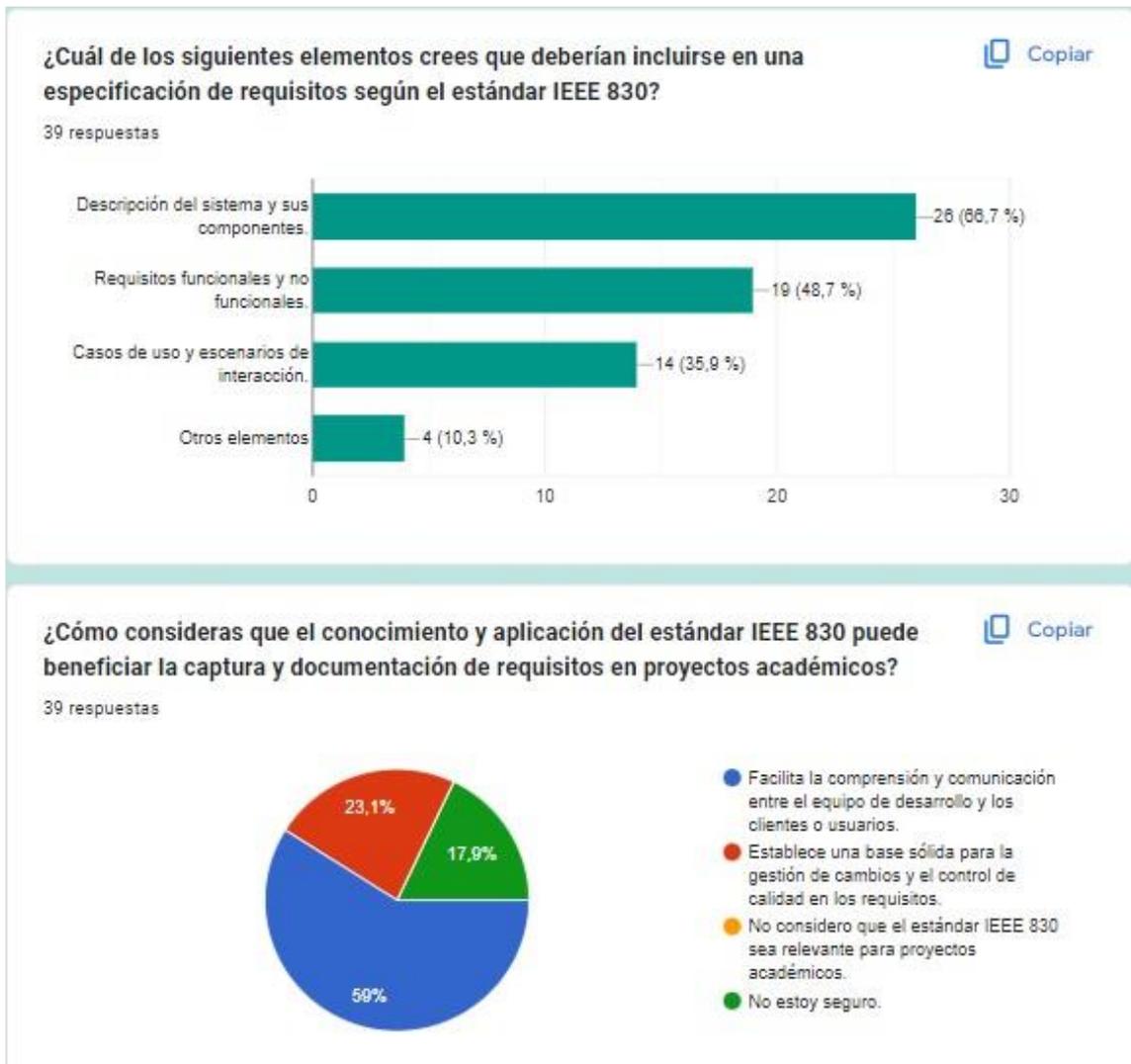
### ¿Cuál es el propósito principal del estándar IEEE 830 en relación con la especificación de requisitos?

[Copiar](#)

39 respuestas



- Establecer una estructura y formato uniforme para la documentación de requisitos.
- Proporcionar directrices específicas para la captura de requisitos en proyectos académicos.
- Definir una metodología para el desarrollo de software.
- No estoy seguro.



**Enlace del Excel de la encuesta – registro de encuestados (correo):**

<https://docs.google.com/spreadsheets/d/12kbc4UM4iN60kfWPA7rcrw2SiRWI MCC-fz8SMmsQKQM/edit?usp=sharing>

**Enlace de la encuesta:**

[https://docs.google.com/forms/d/e/1FAIpQLSfUjh\\_YwQ7u\\_6niSOpFmVQEOS2verQNP6LaLPfB6wTrdFqJrA/viewform?usp=sharing](https://docs.google.com/forms/d/e/1FAIpQLSfUjh_YwQ7u_6niSOpFmVQEOS2verQNP6LaLPfB6wTrdFqJrA/viewform?usp=sharing)

## **Anexo 2. Documento de especificación de requisitos (ERS)**

En el presente anexo se presenta el documento de especificación de requisitos de software, el cual detalla los requisitos funcionales y no funcionales identificados para el desarrollo de la aplicación.

### **[Especificación de requisitos de sistema]**

# **Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la Universidad Nacional de Loja**

Revisión 03.05

*Elaborado por:*

*Edinson Roberto Quizhpe Vizcaíno*

*Revisado por:*

*Ing. Wilman Chamba*

*Aprobado por:*

*Ing. Wilman Chamba*

Mayo 2024

## TABLA DE CONTENIDO

<b>HISTORIAL DE VERSIONES .....</b>	<b>93</b>
<b>INFORMACIÓN DEL PROYECTO .....</b>	<b>93</b>
<b>1. INTRODUCCIÓN .....</b>	<b>94</b>
1.2. DESCRIPCIÓN DEL PROYECTO .....	94
1.3. OBJETIVO .....	94
1.3. REFERENCIAS.....	94
1.4. ACTORES INVOLUCRADOS .....	95
1.4.1. ROLES DE USUARIOS .....	96
1.4.2. ENTORNO OPERATIVO .....	96
1.4.2.1. DE CONTROL DE ACCESO.....	<b>¡ERROR! MARCADOR NO DEFINIDO.</b>
<b>2. REQUERIMIENTOS DE ALTO NIVEL .....</b>	<b>96</b>
2.1. REQUERIMIENTOS FUNCIONALES.....	96
2.1.1. <i>Requerimientos del negocio</i> .....	96
2.1.2. <i>Requerimientos del usuario</i> .....	101
2.1.3. <i>Requerimientos del software</i> .....	105
<b>3. REQUERIMIENTOS NO FUNCIONALES .....</b>	<b>109</b>
<b>4. HISTORIAS DE USUARIO .....</b>	<b>114</b>
<b>5. GLOSARIO DE TÉRMINOS .....</b>	<b>131</b>
DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS .....	131
<b>6. APROBACIONES.....</b>	<b>131</b>

## Historial de versiones

<b>Versión</b>	<b>Fecha</b>	<b>Responsable</b>	<b>Descripción de la modificación</b>
03.05	03/05/2024	Edinson Roberto Quizhpe Vizcaíno	Documento inicial

## Información del Proyecto

**Tabla 1.** Herramienta para la Carrera Computación de la UNL.

<b>Unidad requirente</b>	Carrera Computación
<b>Nombre del proyecto</b>	Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la Universidad Nacional de Loja.
<b>Fecha de elaboración</b>	03-05-2024
<b>Solicitante / Patrocinador principal</b>	Carrera Computación de la Universidad Nacional de Loja

## **1. Introducción**

En la presente, se describe la Especificación de Requisitos de Software (ERS) del proyecto **“Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la Universidad Nacional de Loja”**. La presente ERS se basa en las encuestas realizadas a los estudiantes de la Carrera Computación de la Universidad Nacional de Loja.

### **1.2. Descripción del proyecto**

La Carrera Computación de la UNL, a favor de mejorar la claridad y estructura de los requisitos de software en los proyectos de los estudiantes, prevé la necesidad de desarrollar una herramienta que permita gestionar los requisitos de software, para mejorar los procesos que se realizan manualmente o sin una herramienta específica.

### **1.3. Objetivo**

El proyecto “Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la UNL” tiene como alcance desarrollar una solución que permita cubrir las siguientes necesidades:

- Registro de proyectos.
- Registro de requisitos.
- Agregar colaboradores.
- Historial de cambios.
- Generar reportes.

### **1.4. Referencias**

Como referencias del proyecto *“Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la UNL”* se tiene:

**Tabla 2.** Referencias

Referencia	Título	Ruta
Encuesta	Encuesta PIC	<a href="https://docs.google.com/forms/d/1yFHJHGY7UVixaAO0d62QYgXoLbBDHzHN9p8ER8EVh3g">https://docs.google.com/forms/d/1yFHJHGY7UVixaAO0d62QYgXoLbBDHzHN9p8ER8EVh3g</a>
Herramienta	ReqTest	<a href="https://reqtest.com/en/">https://reqtest.com/en/</a>
Herramienta	Visure Software	<a href="https://visuresolutions.com/">https://visuresolutions.com/</a>

### 1.5. Actores involucrados

**Tabla 3.** Director del Trabajo de Integración Curricular

<b>Nombre</b>	Wilman Patricio Chamba Zaragocín
<b>Rol</b>	Usuario del producto
<b>Responsabilidad</b>	Proporcionar información y directrices en el desarrollo del proyecto
<b>Información de contacto</b>	wilman.chamba@unl.edu.ec

**Tabla 4.** Autor del Trabajo de Integración Curricular

<b>Nombre</b>	Edinson Roberto Quizhpe Vizcaíno
<b>Rol</b>	Desarrollador del proyecto
<b>Responsabilidad</b>	Desarrollo del proyecto
<b>Información de contacto</b>	edinson.quizhpe@unl.edu.ec

### 1.5.1. Roles de usuarios

- **Administrador:** Cuenta con acceso a todas las funcionalidades del sistema.
- **Gestor del proyecto:** Crea, modifica, actualiza, y elimina proyectos, requisitos e informes.
- **Analista de negocios:** Análisis de los requisitos, revisión de los requisitos y seguimiento de los cambios en los requisitos.
- **Validador:** Validación y verificación de los requisitos, reportes del estado y cambios en los requisitos.

## 2. Requerimientos de alto nivel

### 2.1. Requerimientos Funcionales

#### 2.1.1. Requerimientos del negocio

**Tabla 5.** RF01

<b>Identificación del requerimiento:</b>	RF01
<b>Nombre del Requerimiento:</b>	Crear proyecto
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al gestor del proyecto crear un nuevo proyecto, con la siguiente información:</p> <ul style="list-style-type: none"><li>• Nombre del proyecto</li><li>• Objetivo general</li><li>• Objetivos específicos</li><li>• Fecha de inicio</li><li>• Fecha de finalización</li><li>• Actores</li></ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 6.** RF02

<b>Identificación del requerimiento:</b>	RF02
<b>Nombre del Requerimiento:</b>	Actualizar información del proyecto
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al gestor del proyecto modificar/actualizar la siguiente información del proyecto:</p> <ul style="list-style-type: none"><li>• Nombre del proyecto</li><li>• Objetivo general</li><li>• Objetivos específicos</li><li>• Fecha de inicio</li><li>• Fecha de finalización</li><li>• Actores</li></ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 7.** RF03

<b>Identificación del requerimiento:</b>	RF03
<b>Nombre del Requerimiento:</b>	Listar proyectos
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto visualizar todos los proyectos que ha creado.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 8.** RF04

<b>Identificación del requerimiento:</b>	RF04
<b>Nombre del Requerimiento:</b>	Dar de baja proyecto
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto dar de baja un proyecto que ha creado. Se debe mostrar un mensaje de confirmación para dar de baja el proyecto.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 9.** RF05

<b>Identificación del requerimiento:</b>	RF05
<b>Nombre del Requerimiento:</b>	Crear actor del proyecto
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto y al analista de negocios crear un nuevo actor, con la siguiente información: <ul style="list-style-type: none"><li>• Nombre</li><li>• Descripción</li></ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 10.** RF06

<b>Identificación del requerimiento:</b>	RF06
<b>Nombre del Requerimiento:</b>	Actualizar información del actor
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto y al analista de negocios modificar/actualizar la siguiente información de un actor: <ul style="list-style-type: none"><li>• Nombre</li></ul>

	<ul style="list-style-type: none"> <li>• Descripción</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 11. RF07**

<b>Identificación del requerimiento:</b>	RF07
<b>Nombre del Requerimiento:</b>	Eliminar actor del proyecto
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto y al analista de negocios eliminar un actor creado en el proyecto que es responsable.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 12. RF08**

<b>Identificación del requerimiento:</b>	RF08
<b>Nombre del Requerimiento:</b>	Crear nuevo requisito
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al gestor del proyecto y analista de negocios crear un nuevo requisito, con los campos:</p> <ul style="list-style-type: none"> <li>• ID</li> <li>• Nombre del requisito</li> <li>• Descripción</li> <li>• Prioridad</li> <li>• Tipo de requisito</li> <li>• Técnica de elicitación</li> <li>• Actor</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 13.** RF09

<b>Identificación del requerimiento:</b>	RF09
<b>Nombre del Requerimiento:</b>	Actualizar información de un requisito.
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al gestor del proyecto y analista de negocios actualizar/modificar la siguiente información de un requisito creado en un proyecto que sea responsable:</p> <ul style="list-style-type: none"><li>● Nombre del requisito</li><li>● Descripción</li><li>● Prioridad</li><li>● Tipo de requisito</li><li>● Técnica de elicitación</li><li>● Actor</li></ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 14.** RF10

<b>Identificación del requerimiento:</b>	RF10
<b>Nombre del Requerimiento:</b>	Listar requisitos
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto y analista de negocios, visualizar todos los requisitos creados de cada proyecto que sea responsable.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 15.** RF11

<b>Identificación del requerimiento:</b>	RF11
<b>Nombre del Requerimiento:</b>	Dar de baja requisito
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto y analista de negocios, dar de baja un requisito creado en el proyecto(s) que sea responsable.
<b>Prioridad del requerimiento:</b>	Alta

## 2.1.2. Requerimientos del usuario

**Tabla 16.** RF012

<b>Identificación del requerimiento:</b>	RF12
<b>Nombre del Requerimiento:</b>	Generar informe de requisitos
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto y analista de negocios, generar un documento de especificación de requisitos (ERS) del proyecto(s) que sea responsable.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 17.** RF13

<b>Identificación del requerimiento:</b>	RF13
<b>Nombre del Requerimiento:</b>	Listar informes generados
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto y analista de negocio, visualizar un listado de todos los informes que ha generado de los proyectos que sea

	responsable.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 18.** RF14

<b>Identificación del requerimiento:</b>	RF14
<b>Nombre del Requerimiento:</b>	Agregar colaboradores al proyecto
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto agregar colaboradores al proyecto que sea responsable, los cuales tendrán un rol asociado.  Para agregar un colaborador se lo debe hacer mediante el correo electrónico del colaborador.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 19.** RF15

<b>Identificación del requerimiento:</b>	RF15
<b>Nombre del Requerimiento:</b>	Eliminar colaborador del proyecto
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto eliminar un colaborador del proyecto, se debe mostrar un mensaje de confirmación.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 20.** RF16

<b>Identificación del requerimiento:</b>	RF16
<b>Nombre del Requerimiento:</b>	Revisión de requisitos

<b>Descripción del requerimiento:</b>	El sistema debe permitir al analista de negocios y validador, revisar los requisitos de un proyecto en el que sea colaborador o responsable, este puede agregar una observación por cada requisito.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 21.** RF17

<b>Identificación del requerimiento:</b>	RF17
<b>Nombre del Requerimiento:</b>	Asignar prioridad a un requisito
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al analista de negocios asignar una prioridad a cada requisito creado, en un proyecto en el que sea responsable. La prioridad que puede asignar es:</p> <ul style="list-style-type: none"> <li>• Baja</li> <li>• Media</li> <li>• Alta</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 22.** RF18

<b>Identificación del requerimiento:</b>	RF18
<b>Nombre del Requerimiento:</b>	Cambiar prioridad de un requisito
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al analista de negocios modificar la prioridad de un requisito creado en un proyecto en el que sea responsable. La prioridad puede modificarse a:</p> <ul style="list-style-type: none"> <li>• Baja</li> <li>• Media</li> <li>• Alta</li> </ul>

<b>Prioridad del requerimiento:</b>	Alta
-------------------------------------	------

**Tabla 22.** RF19

<b>Identificación del requerimiento:</b>	RF19
<b>Nombre del Requerimiento:</b>	Registrar cambios en los requisitos
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al analista de negocios realizar un control de cambios en los requisitos en un proyecto en el que sea responsable. Los cambios que puede hacer son:</p> <ul style="list-style-type: none"> <li>• Nombre del requisito</li> <li>• Descripción del requisito</li> <li>• Prioridad</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 23.** RF20

<b>Identificación del requerimiento:</b>	RF20
<b>Nombre del Requerimiento:</b>	Generar informe de requisitos
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al validador generar un informe de los requisitos del proyecto en el que sea responsable, en el cual se debe incluir la siguiente información:</p> <ul style="list-style-type: none"> <li>• Estado de los requisitos.</li> <li>• Cambios relevantes en los requisitos</li> </ul> <p>El estado de los requisitos puede ser:</p> <ul style="list-style-type: none"> <li>• En revisión.</li> <li>• Validado.</li> <li>• Aprobado.</li> <li>• Rechazado.</li> </ul>

<b>Prioridad del requerimiento:</b>	Alta
-------------------------------------	------

### 2.1.3. Requerimientos del software

**Tabla 24.** RF21

<b>Identificación del requerimiento:</b>	RF21
<b>Nombre del Requerimiento:</b>	Registrar cuenta de usuario
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir a un usuario no registrado, crear una cuenta mediante un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> <li>• Nombres</li> <li>• Apellidos</li> <li>• Correo</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 25.** RF22

<b>Identificación del requerimiento:</b>	RF22
<b>Nombre del Requerimiento:</b>	Actualizar información del usuario
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al gestor del proyecto modificar/actualizar la información de su cuenta de usuario:</p> <ul style="list-style-type: none"> <li>• Celular.</li> <li>• Dirección.</li> <li>• Correo de respaldo.</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 26.** RF23

<b>Identificación del requerimiento:</b>	RF23
<b>Nombre del Requerimiento:</b>	Generar contraseña
<b>Descripción del requerimiento:</b>	El sistema generará una contraseña para cada usuario registrado.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 27.** RF24

<b>Identificación del requerimiento:</b>	RF24
<b>Nombre del Requerimiento:</b>	Generar nueva contraseña
<b>Descripción del requerimiento:</b>	El sistema debe permitir a los usuarios generar una nueva contraseña para su cuenta de usuario.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 28.** RF25

<b>Identificación del requerimiento:</b>	RF25
<b>Nombre del Requerimiento:</b>	Dar de baja cuenta de usuario
<b>Descripción del requerimiento:</b>	El sistema debe permitir a los usuarios dar de baja (eliminar) su cuenta de usuario, se debe mostrar un mensaje de confirmación.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 29.** RF26

<b>Identificación del requerimiento:</b>	RF26
<b>Nombre del Requerimiento:</b>	Listar usuarios
<b>Descripción del requerimiento:</b>	El sistema debe permitir al administrador visualizar todos los usuarios registrados en el sistema.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 31.** RF27

<b>Identificación del requerimiento:</b>	RF27
<b>Nombre del Requerimiento:</b>	Activar cuenta de usuario.
<b>Descripción del requerimiento:</b>	El sistema debe permitir al administrador activar una cuenta de un usuario registrado en el sistema, en caso de que haya sido desactivada.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 32.** RF28

<b>Identificación del requerimiento:</b>	RF28
<b>Nombre del Requerimiento:</b>	Asignar rol a los usuarios
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al administrador asignar un único rol a cada usuario, los roles que se pueden asignar son:</p> <ul style="list-style-type: none"><li>• Gestor del proyecto</li><li>• Analista de negocios</li><li>• Validador</li></ul>

<b>Prioridad del requerimiento:</b>	Alta
-------------------------------------	------

**Tabla 33.** RF29

<b>Identificación del requerimiento:</b>	RF29
<b>Nombre del Requerimiento:</b>	Modificar rol de usuario
<b>Descripción del requerimiento:</b>	<p>El sistema debe permitir al administrador modificar el rol asignado a un usuario, los roles que se pueden asignar son:</p> <ul style="list-style-type: none"> <li>• Gestor del proyecto</li> <li>• Analista de negocios</li> <li>• Validador</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 34.** RF30

<b>Identificación del requerimiento:</b>	RF30
<b>Nombre del Requerimiento:</b>	Mostrar gráficas del estado de requisitos.
<b>Descripción del requerimiento:</b>	<p>El sistema debe mostrar gráficas del estado de los requisitos, considerando:</p> <ul style="list-style-type: none"> <li>• Fecha de creación.</li> <li>• Número de requisitos.</li> <li>• Estado (En revisión, validado, aprobado, rechazado).</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 35.** RF31

<b>Identificación del requerimiento:</b>	RF31
<b>Nombre del Requerimiento:</b>	Inicio de sesión
<b>Descripción del requerimiento:</b>	El sistema debe permitir al gestor del proyecto, analista de negocios, validador y administrador, iniciar sesión mediante sus credenciales (Correo y contraseña).
<b>Prioridad del requerimiento:</b>	Alta

### 3. Requerimientos no Funcionales

**Tabla 35.** RNF01

<b>Identificación del requerimiento:</b>	RNF01
<b>Categoría</b>	Eficiencia
<b>Descripción del requerimiento:</b>	La herramienta debe brindar un tiempo aceptable de respuesta entre 1 a 5 segundos promedio.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 36.** RNF02

<b>Identificación del requerimiento:</b>	RNF02
<b>Categoría:</b>	Eficiencia
<b>Descripción del</b>	La herramienta debe ser capaz de soportar un máximo de 1000 usuarios de manera simultánea.

<b>requerimiento:</b>	
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 37.** RNF03

<b>Identificación del requerimiento:</b>	RNF03
<b>Categoría:</b>	Seguridad
<b>Descripción del requerimiento:</b>	Los permisos de roles podrán ser cambiados únicamente por el administrador.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 38.** RNF04

<b>Identificación del requerimiento:</b>	RNF04
<b>Categoría:</b>	Seguridad
<b>Descripción del requerimiento:</b>	Las contraseñas deben ser generadas con un algoritmo de manera aleatoria por el sistema.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 39.** RNF05

<b>Identificación del requerimiento:</b>	RNF05
<b>Categoría</b>	Seguridad
<b>Descripción del requerimiento:</b>	La herramienta debe permitir el acceso a los usuarios de acuerdo con sus credenciales y permisos asignados.

<b>Prioridad del requerimiento:</b>	Alta
-------------------------------------	------

**Tabla 40.** RNF06

<b>Identificación del requerimiento:</b>	RNF06
<b>Categoría</b>	Disponibilidad
<b>Descripción del requerimiento:</b>	La herramienta debe estar disponible las 24 horas del día, todos los días de la semana, durante un año.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 41.** RNF07

<b>Identificación del requerimiento:</b>	RNF07
<b>Categoría</b>	Disponibilidad
<b>Descripción del requerimiento:</b>	El tiempo para iniciar el sistema no debe ser mayor a 3 minutos.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 42.** RNF08

<b>Identificación del requerimiento:</b>	RNF08
<b>Categoría</b>	Usabilidad
<b>Descripción del requerimiento:</b>	La herramienta debe tener una interfaz sencilla, amigable e intuitiva para los usuarios.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 43.** RNF09

<b>Identificación del requerimiento:</b>	RNF09
<b>Categoría</b>	Usabilidad
<b>Descripción del requerimiento:</b>	El tiempo de aprendizaje de la herramienta para un usuario no debe ser mayor a 3 horas.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 44.** RNF10

<b>Identificación del requerimiento:</b>	RNF10
<b>Categoría</b>	Usabilidad
<b>Descripción del requerimiento:</b>	La herramienta debe presentar mensajes de alerta e informativos al usuario.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 45.** RNF11

<b>Identificación del requerimiento:</b>	RNF11
<b>Categoría</b>	Mantenibilidad
<b>Descripción del requerimiento:</b>	El código fuente de la herramienta debe ser claro y fácil de escalar.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 46.** RNF12

<b>Identificación del requerimiento:</b>	RNF12
<b>Categoría</b>	Portabilidad
<b>Descripción del requerimiento:</b>	El entorno de programación de la herramienta será Nodejs, JavaScript, TypeScript, Nextjs.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 47.** RNF13

<b>Identificación del requerimiento:</b>	RNF13
<b>Categoría</b>	Portabilidad
<b>Descripción del requerimiento:</b>	Se emplearán una base de datos no relacional, la cual debe ser compatible con el entorno de programación.
<b>Prioridad del requerimiento:</b>	Alta

**Tabla 48.** RNF14

<b>Identificación del requerimiento:</b>	RNF14
<b>Categoría</b>	Portabilidad
<b>Descripción del requerimiento:</b>	La herramienta será implementada para navegadores web (Google Chrome, Mozilla Firefox, Microsoft Edge).
<b>Prioridad del requerimiento:</b>	Alta

**UNL**Universidad  
Nacional  
de Loja

#### 4. Historias de usuario

ID: H001	Crear proyectos		
Usuario / Rol	Gestor del proyecto	Prioridad	Alta
Descripción	Deseo poder crear nuevos proyectos.		
Nro	Criterio de aceptación		
1	Debe poder crear nuevos proyectos en el sistema, con la siguiente información: nombre del proyecto, objetivo general y objetivos específicos del proyecto, fechas de inicio y finalización del proyecto, y los actores que participan en el mismo.		

ID: H002	Cambiar información del proyecto		
Usuario / Rol	Gestor del proyecto	Prioridad	Alta
Descripción	Deseo cambiar la información de un proyecto creado		
Nro.	Criterio de aceptación		



UNL

Universidad  
Nacional  
de Loja

1	Debe poder cambiar la siguiente información de un proyecto creado por el usuario en el sistema: <ul style="list-style-type: none"><li>• Nombre del proyecto</li><li>• Objetivo general</li><li>• Objetivos específicos</li><li>• Fecha de inicio</li><li>• Fecha de finalización</li><li>• Actores</li></ul>
---	--

ID: H003	Visualizar proyectos creados		
Usuario / Rol	Gestor del proyecto	Prioridad	Alta
Descripción	Deseo mostrar todos los proyectos creados		
Nro.	<b>Criterio de aceptación</b>		
1	Debe mostrar una lista con todos los proyectos creados por el usuario en el sistema.		



1859

UNL

Universidad  
Nacional  
de Loja

ID: H004	Eliminar un proyecto creado		
Usuario / Rol	Gestor del proyecto	Prioridad	Alta
Descripción	Deseo eliminar un proyecto creado		
Nro.	Criterio de aceptación		
1	Debe poder eliminar un proyecto que ha sido creado por el usuario, y se debe mostrar un mensaje de confirmación para su eliminación.		

ID: H005	Crear actores del proyecto		
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta
Descripción	Deseo crear nuevos actores para un proyecto		
Nro.	Criterio de aceptación		
1	Debe poder crear nuevos actores para cada proyecto que un usuario haya creado, los actores pueden tener la siguiente información: nombre del actor y descripción.		

**UNL**Universidad  
Nacional  
de Loja

ID: H006	Actualizar información de un actor		
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta
Descripción	Deseo modificar la información de un actor creado		
Nro.	Criterio de aceptación		
1	Debe poder modificar la siguiente información de un actor, que haya sido creado por el usuario: nombre del actor y descripción.		

ID: H007	Eliminar actor de un proyecto		
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta
Descripción	Deseo eliminar un actor creado en un proyecto		
Nro.	Criterio de aceptación		
1	Debe poder eliminar un actor que ha sido creado por el usuario en un proyecto.		



**UNL**

Universidad  
Nacional  
de Loja

ID: H008		Crear requisitos en un proyecto	
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta
Descripción	Deseo crear nuevos requisitos en un proyecto		
Nro.	Criterio de aceptación		
1	Debe poder crear nuevos requisitos en un proyecto, con la siguiente información: <ul style="list-style-type: none"><li>• ID</li><li>• Nombre del requisito</li><li>• Descripción</li><li>• Prioridad</li><li>• Tipo de requisito</li><li>• Técnica de elicitación</li><li>• Actor</li></ul>		

ID: H009		Modificar información de un requisito	
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta



**UNL**

Universidad  
Nacional  
de Loja

<b>Descripción</b>	Deseo actualizar la información de un requisito creado en un proyecto
<b>Nro.</b>	<b>Criterio de aceptación</b>
1	Debe poder cambiar la siguiente información de un requisito creado en un proyecto del usuario: <ul style="list-style-type: none"><li>• Nombre del requisito</li><li>• Descripción</li><li>• Prioridad</li><li>• Tipo de requisito</li><li>• Técnica de elicitación</li><li>• Actores</li></ul>

<b>ID: H010</b>	<b>Visualizar los requisitos de un proyecto</b>		
<b>Usuario / Rol</b>	Gestor del proyecto/Analista de negocios	<b>Prioridad</b>	Alta
<b>Descripción</b>	Deseo mostrar todos los requisitos creados en un proyecto		



**UNL**

Universidad  
Nacional  
de Loja

Nro.	Criterio de aceptación
1	Debe mostrar una lista con todos los requisitos creados por el usuario en un proyecto.

ID: H011	Eliminar un requisito en un proyecto		
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta
Descripción	Deseo eliminar un requisito creado en un proyecto		
Nro.	Criterio de aceptación		
1	Debe poder eliminar un requisito creado en un proyecto del usuario, se mostrará un mensaje de confirmación para su eliminación.		



1859

UNL

Universidad  
Nacional  
de Loja

ID: H012	Generar documento de especificación de requisitos		
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta
Descripción	Deseo generar un documento ERS de un proyecto creado		
Nro.	<b>Criterio de aceptación</b>		
1	Debe generar un documento ERS de un proyecto creado, el mismo puede tener los siguientes formatos: <ul style="list-style-type: none"><li>• IEEE830</li><li>• ISO/IEEE 29148</li></ul>		

ID: H013	Mostrar lista de documentos ERS generados		
Usuario / Rol	Gestor del proyecto	Prioridad	Alta
Descripción	Deseo mostrar una lista de los documentos ERS que se han generado		
Nro.	<b>Criterio de aceptación</b>		
1	Debe mostrar una lista con todos los documentos ERS que se han generado de los proyectos.		



**UNL**

Universidad  
Nacional  
de Loja

ID: H014	Añadir colaboradores a un proyecto		
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta
Descripción	Deseo agregar colaboradores a un proyecto		
Nro.	Criterio de aceptación		
1	Debe poder agregar uno o varios colaboradores a un proyecto del usuario, los cuales tendrán un rol asociado. Para agregar un colaborador se lo debe realizar mediante el correo del colaborador.		

ID: H015	Eliminar colaborador de un proyecto		
Usuario / Rol	Gestor del proyecto/Analista de negocios	Prioridad	Alta
Descripción	Deseo eliminar un colaborador de un proyecto creado		
Nro.	Criterio de aceptación		
1	Debe poder eliminar un colaborador que ha sido agregado a un proyecto, y se debe mostrar un mensaje de confirmación para su eliminación.		



**UNL**

Universidad  
Nacional  
de Loja

ID: H016	Revisión de requisitos creados en un proyecto		
Usuario / Rol	Analista de negocios/Validador	Prioridad	Alta
Descripción	Deseo validar los requisitos creados en un proyecto		
Nro.	Criterio de aceptación		
1	Debe permitir agregar observaciones sobre cada requisito en un proyecto en el que sea colaborador.		

ID: H017	Asignar prioridad a un requisito en un proyecto		
Usuario / Rol	Analista de negocios	Prioridad	Alta
Descripción	Deseo asignar una prioridad a cada requisito de un proyecto		
Nro.	Criterio de aceptación		
1	Debe poder asignar una prioridad a cada requisito dentro de un proyecto en el que sea responsable. Las prioridades deben ser mostradas en una lista de ítems: <ul style="list-style-type: none"><li>• Baja</li><li>• Media</li></ul>		



UNL

Universidad  
Nacional  
de Loja

- Alta

ID: H018	Cambiar la prioridad de un requisito en un proyecto		
Usuario / Rol	Analista de negocios	Prioridad	Alta
Descripción	Deseo modificar la prioridad de cada requisito en un proyecto		
Nro.	Criterio de aceptación		
1	Debe poder cambiar la prioridad que ha sido asignada a un requisito de un proyecto en el que sea responsable. Las prioridades deben ser mostradas en una lista de ítems: <ul style="list-style-type: none"><li>• Baja</li><li>• Media</li><li>• Alta</li></ul>		

ID: H019	Registrar cambios en los requisitos de un proyecto		
Usuario / Rol	Analista de negocios	Prioridad	Alta
Descripción	Deseo realizar un control de cambios en los requisitos de un proyecto		



UNL

Universidad  
Nacional  
de Loja

1859

Nro.	Criterio de aceptación
1	<p>Debe poder hacer cambios en los requisitos de un proyecto en el que sea responsable, la información que se puede modificar es:</p> <ul style="list-style-type: none"> <li>• Nombre del requisito</li> <li>• Descripción del requisito</li> <li>• Prioridad</li> </ul>

ID: H020	Generar informe de los requisitos de un proyecto		
Usuario / Rol	Validador	Prioridad	Alta
Descripción	Deseo generar un informe de los requisitos de un proyecto		
Nro.	Criterio de aceptación		
1	<p>Debe poder generar un informe de los requisitos de un proyecto en el que sea responsable, en el cual se debe tener la siguiente información: Estado de los requisitos y cambios relevantes en los requisitos del proyecto.</p> <p>El estado de los requisitos debe mostrarse en una lista de ítems:</p> <ul style="list-style-type: none"> <li>• En revisión</li> <li>• Validado</li> <li>• Aprobado</li> <li>• Rechazado</li> </ul>		



1859

UNL

Universidad  
Nacional  
de Loja

ID: H021	Registrar una cuenta de usuario		
Usuario / Rol	Usuario	Prioridad	Alta
Descripción	Deseo registrar un usuario		
Nro.	Criterio de aceptación		
1	Debe mostrar un formulario de registro con la siguiente información: <ul style="list-style-type: none"><li>• Nombres</li><li>• Apellidos</li><li>• Correo</li></ul>		

ID: H022	Modificar perfil de usuario		
Usuario / Rol	Usuario registrado	Prioridad	Alta
Descripción	Deseo actualizar la información del perfil de mi cuenta de usuario		
Nro.	Criterio de aceptación		
1	Debe poder modificar la siguiente información del perfil de la cuenta de un usuario registrado: <ul style="list-style-type: none"><li>• Celular</li><li>• Dirección</li><li>• Correo de respaldo</li></ul>		

**UNL**Universidad  
Nacional  
de Loja

ID: H023	Generar contraseña de usuario		
Usuario / Rol	Usuario registrado	Prioridad	Alta
Descripción	Deseo generar una contraseña aleatoria para mi cuenta de usuario		
Nro.	Criterio de aceptación		
1	Debe generar una contraseña aleatoria para cada cuenta de usuario registrado en el sistema		

ID: H024	Generar nueva contraseña		
Usuario / Rol	Usuario registrado	Prioridad	Alta
Descripción	Deseo generar una nueva contraseña para mi cuenta de usuario		
Nro.	Criterio de aceptación		
1	Debe generar una nueva contraseña aleatoria para cada cuenta de usuario registrado en el sistema.		

**UNL**Universidad  
Nacional  
de Loja

ID: H025	Eliminar cuenta de usuario		
Usuario / Rol	Usuario registrado	Prioridad	Alta
Descripción	Deseo dar de baja mi cuenta de usuario		
Nro.	<b>Criterio de aceptación</b>		
1	Debe permitir dar de baja la cuenta de usuario de un usuario registrado en el sistema, y se debe mostrar un mensaje de confirmación para su eliminación.		

ID: H026	Listar usuarios		
Usuario / Rol	Administrador	Prioridad	Alta
Descripción	Deseo mostrar todos los usuarios registrados en el sistema		
Nro.	<b>Criterio de aceptación</b>		
1	Debe mostrar una lista con todos los usuarios registrados en el sistema, con el estado de su cuenta de usuario (activa, desactivada).		



1859

UNL

Universidad  
Nacional  
de Loja

ID: H027	Activar cuenta de usuario		
Usuario / Rol	Administrador	Prioridad	Alta
Descripción	Deseo activar una cuenta de un usuario registrado en el sistema		
Nro.	Criterio de aceptación		
1	Debe permitir activar una cuenta de un usuario registrado en el sistema, si la cuenta ha sido desactivada.		
ID: H028	Visualizar gráficas de los requisitos de un proyecto		
Usuario / Rol	Gestor del proyecto	Prioridad	Alta
Descripción	Deseo mostrar gráficas del estado de los requisitos en un proyecto		
Nro.	Criterio de aceptación		
1	Debe mostrar gráficas que detallen el estado de los requisitos de un proyecto en el que sea responsable, con la siguiente información: <ul style="list-style-type: none"><li>• Fecha de creación</li><li>• Número de requisitos creados</li><li>• Estado (en revisión, validado, aprobado, rechazado)</li></ul>		



**UNL**

Universidad  
Nacional  
de Loja

ID: H029		Inicio de sesión	
Usuario / Rol	Usuarios registrados	Prioridad	Alta
Descripción	Deseo iniciar sesión con mis credenciales de acceso		
Nro.	<b>Criterio de aceptación</b>		
1	Debe permitir a un usuario registrado en el sistema, iniciar sesión con sus credenciales de acceso (correo y contraseña)		

## 5. Glosario de términos

### Definiciones, Acrónimos y abreviaturas

**UNL:** Universidad Nacional de Loja

**RF:** Requisito Funcional

**RNF:** Requisito No Funcional

**RN:** Requerimientos del negocio

**RU:** Requerimiento del usuario

**RS:** Requerimiento del software

## 6. Aprobaciones

Nombre y Apellido	Cargo / Rol	Departamento	Firma
Ing. Wilman Patricio Chamba Zaragocín	Director del trabajo de integración curricular	Carrera Computación	
Edinson Roberto Quizhpe Vizcaíno	Tesista	Carrera Computación	

### Anexo 3. Encuesta de experiencia del usuario (SUS).

En el siguiente anexo se presenta los resultados de la encuesta de experiencia de usuario (SUS), aplicada a los estudiantes de la Carrera Computación de la Universidad Nacional de Loja, la cual permitió dar respuesta a la pregunta de investigación. La encuesta planteada se encuentra en el siguiente enlace: <https://forms.gle/uxYucFkuBvyJ13VG8>

## Encuesta para determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema).

La presente encuesta está dirigida a los estudiantes y profesionales afines a la Carrera Computación, con el objetivo de determinar el grado de aceptación mediante la experiencia de usuario (UX), aplicando la metodología SUS (Escala de Usabilidad del Sistema), de la herramienta "ERS TOOL CC".  
Agradezco de antemano, su participación.

edinson.quizhpe@unl.edu.ec [Cambiar de cuenta](#)



\* Indica que la pregunta es obligatoria

Correo \*

Tu dirección de correo electrónico

1. ¿Creo que usaría esta aplicación con frecuencia? \*

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo

2. Encontré la aplicación innecesariamente compleja \*

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo



**3. Creo que la aplicación fue fácil de usar \***

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo

**4. Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar esta aplicación \***

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo

**5. Las funciones de esta aplicación están bien integradas \***

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo

**6. Creo que la aplicación es muy inconsistente (comportamiento incorrecto) \***

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo

**7. Me imagino que la mayoría de los usuarios aprendería a usar esta aplicación de forma muy rápida \***

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo



**8. Encuentro que la aplicación es muy difícil de usar. \***

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo

**9. Me sentí muy seguro usando la aplicación \***

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo

**10. Necesité aprender muchas cosas antes de usar esta aplicación \***

1 2 3 4 5

Totalmente en desacuerdo      Totalmente de acuerdo

Enviar

[Borrar formulario](#)

Este formulario se creó en Universidad Nacional de Loja. [Denunciar abuso](#)

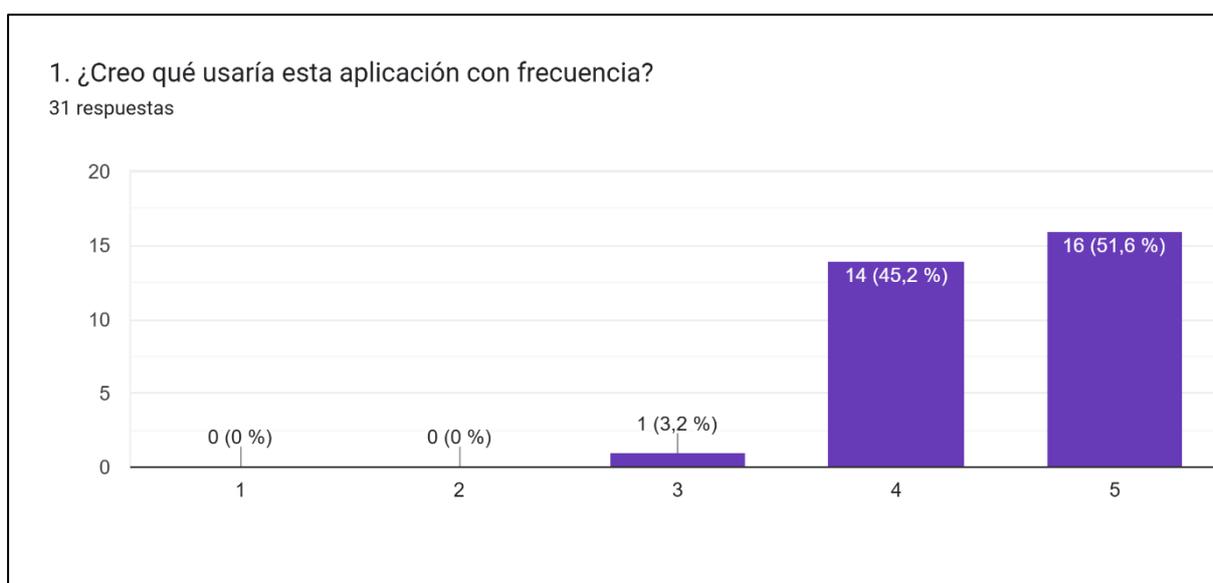
Google Formularios



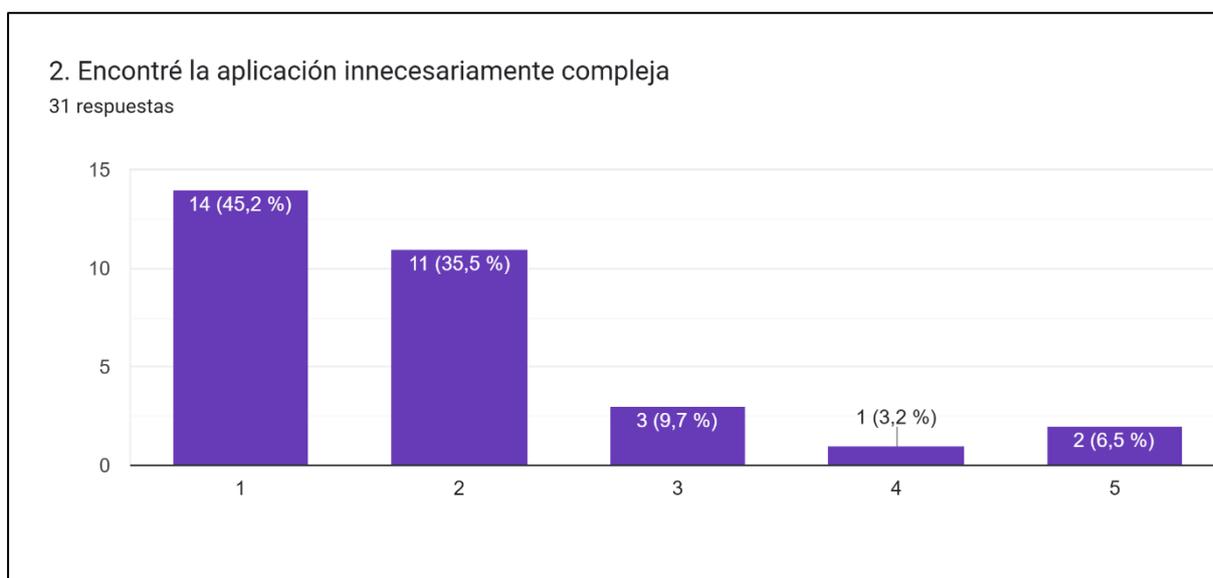
#### Anexo 4. Resultados de las pruebas de aceptación aplicando la metodología SUS.

A continuación, se presentan los resultados de las pruebas de aceptación, aplicadas mediante la metodología SUS.

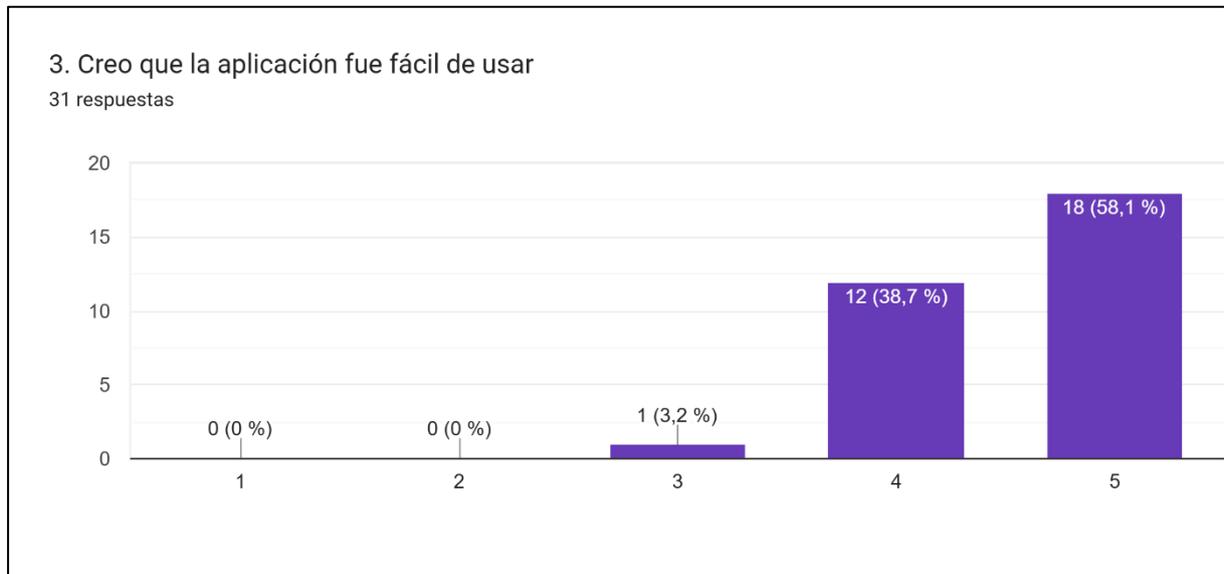
El 51,6% señala que haría uso frecuente de la aplicación web, mientras que un 45,2% esta de acuerdo en que usaría la aplicación web de manera frecuente. Solo un 3,2% se mantiene en una postura neutra sobre el uso frecuente del software.



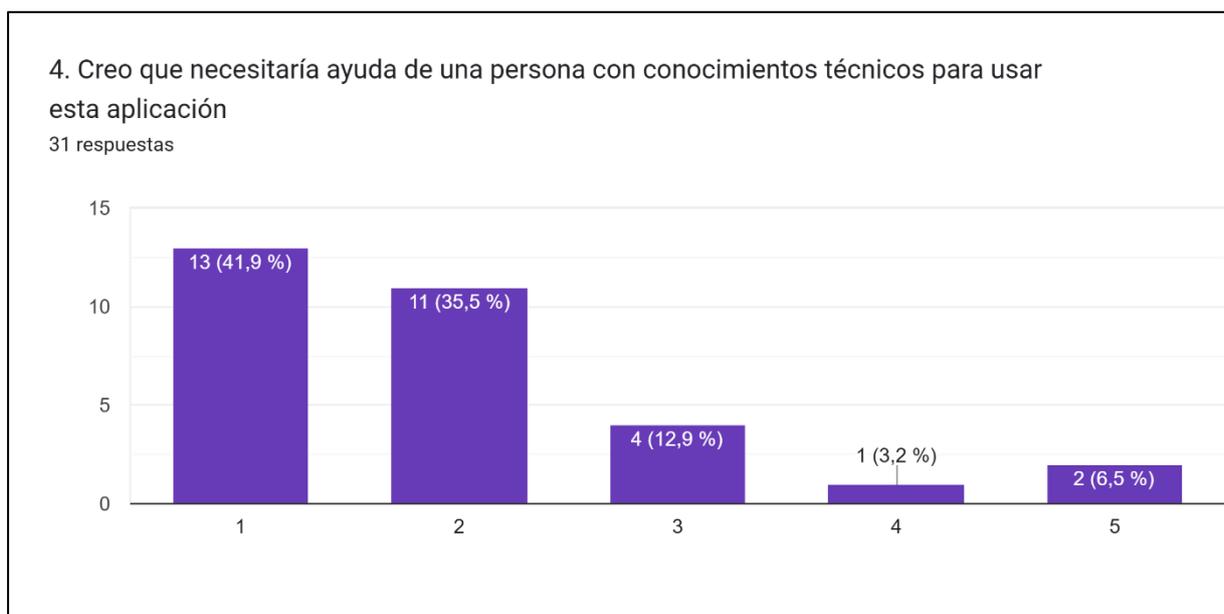
Un 45,2% de los encuestados señaló que no esta de acuerdo en que la aplicación sea innecesariamente compleja, así como un 35,5% se considera en desacuerdo sobre que la aplicación web sea compleja.



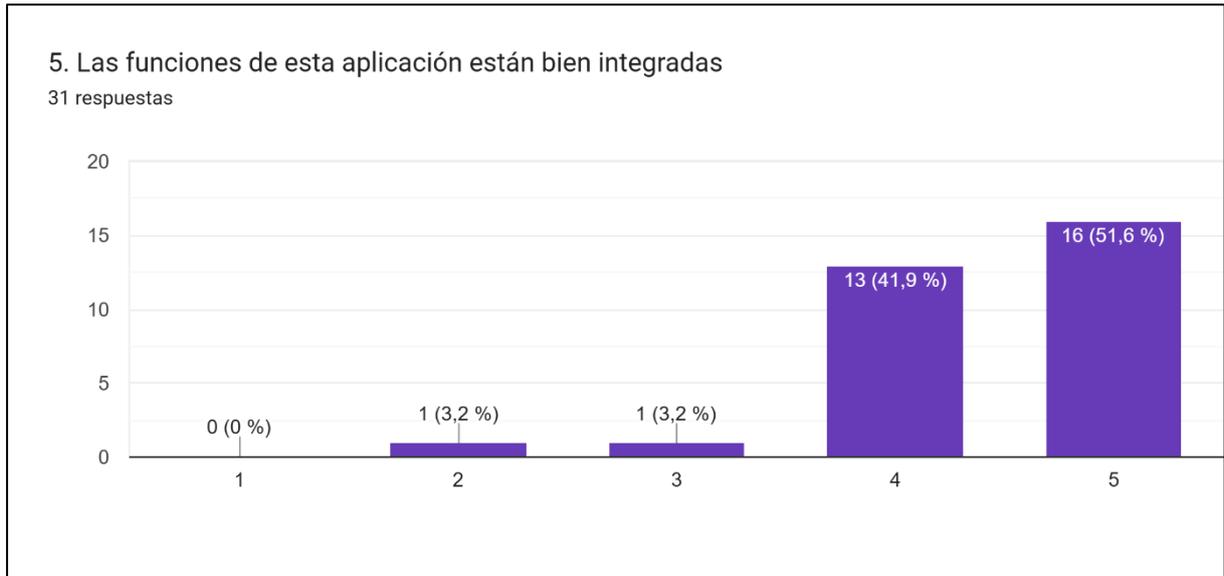
El 58,1% de los encuestados, señalo estar totalmente de acuerdo en que la aplicación resulta fácil de usar, así como un 38,7% considera estar de acuerdo en que la aplicación es de fácil uso.



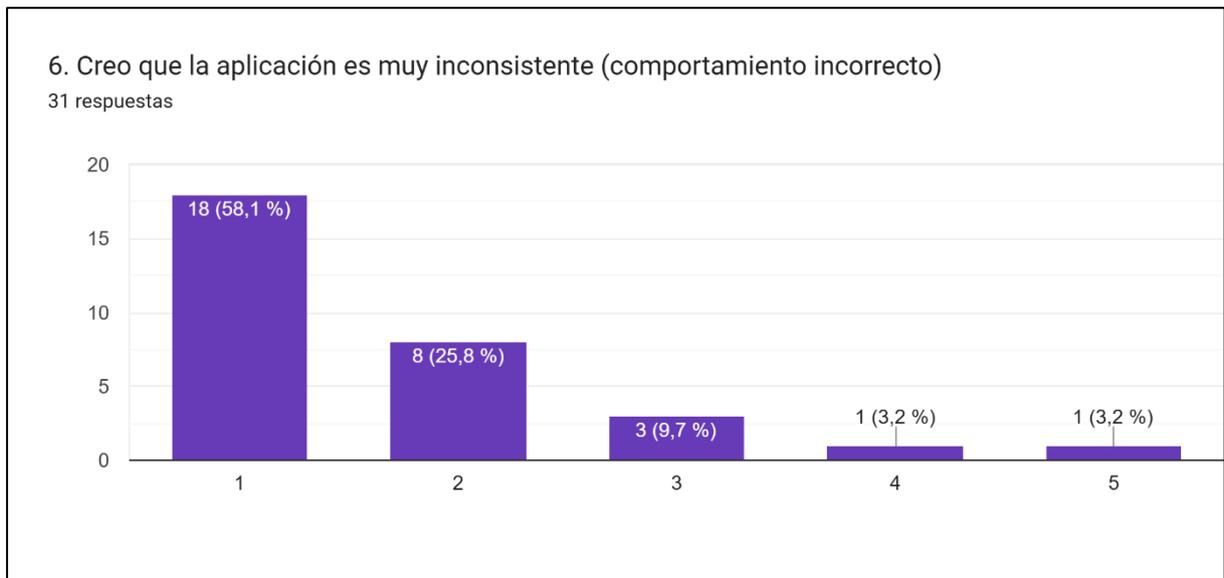
Sobre la ayuda externa para el correcto uso de la herramienta, el 41,9% indica estar totalmente en desacuerdo, así como un 35,5% señala estar en desacuerdo sobre el tener que recurrir a una persona con conocimientos especializados, para el uso de la herramienta.



La mayoría de los encuestados (51,6%), esta totalmente de acuerdo en que las funcionalidades de la aplicación se encuentran bien integra, así como un 41,9% indica estar de acuerdo en la correcta integración de las funcionalidades de la aplicación.



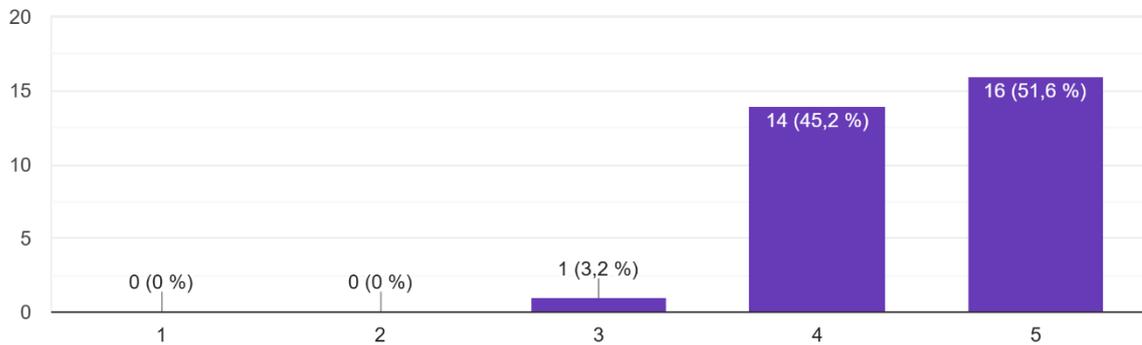
El 58,1% encontró que la aplicación no es inconsistente, así como también el 25,8% indica no estar de acuerdo sobre la existencia de inconsistencias en la aplicación web.



Sobre la facilidad de aprendizaje de la aplicación web, el 51,6% y el 45,2% encuentran en que el aprendizaje del uso de la herramienta sería un proceso muy rápido.

7. Me imagino que la mayoría de los usuarios aprendería a usar esta aplicación de forma muy rápida

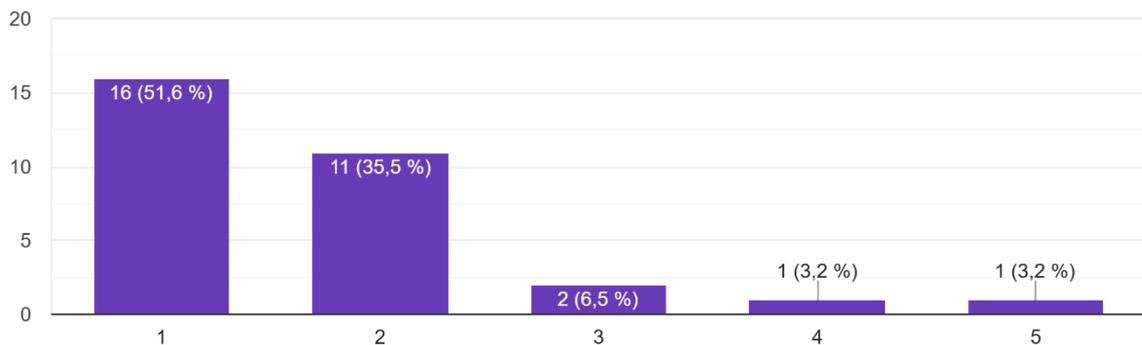
31 respuestas



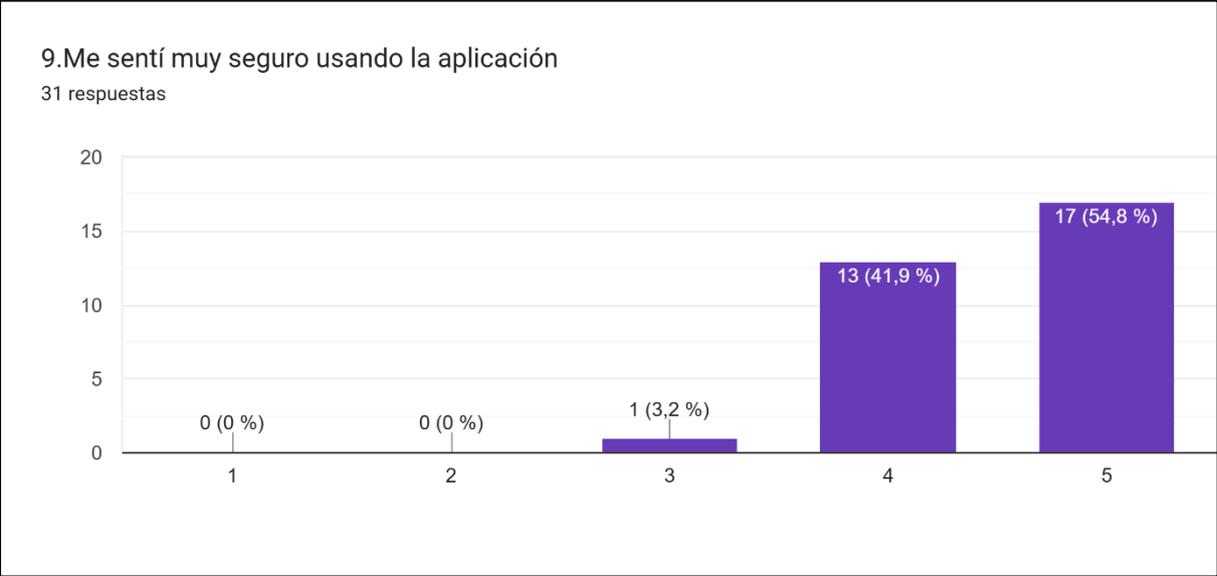
El 87,1% de los encuestados están en desacuerdo sobre que la aplicación web es difícil de usar, tanto solo un 6,4% considera que la herramienta es de uso difícil.

8. Encuentro que la aplicación es muy difícil de usar.

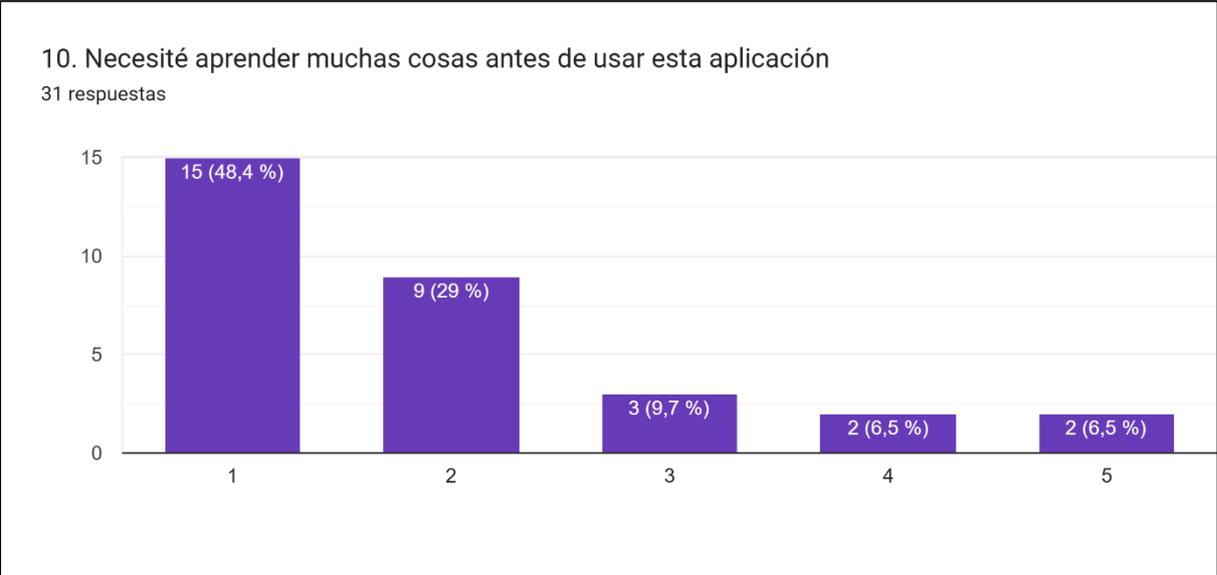
31 respuestas



La mayoría de encuestados 54,8% y 41,9%, consideran que la aplicación durante el uso de la aplicación web se sintieron seguros con la misma.



El 48,9% y el 29% de los encuestados, consideran que no se necesitó de un aprendizaje extenso para poder hacer uso de la aplicación web, solo un 6,5% considera que si tuvo que aprender muchas cosas para poder usar la aplicación.



**Anexo 5. Cálculo de Score en base a la encuesta aplicada.**

A continuación, se presentan los resultados del cálculo de las encuestas aplicadas, que sirvieron para determinar el grado de aceptación de la aplicación web mediante la metodología SUS.

Resultado Promedio	Resultado Promedio												
83,1	A												
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Resultado SUS	Calificación SUS	
	4	3	4	3	4	3	4	3	4	3	62,5	D	
	5	1	5	1	4	1	5	1	4	1	95	A	
	5	1	5	1	4	1	4	1	4	1	92,5	A	
	4	2	4	1	4	2	4	1	4	3	77,5	B	
	4	2	5	1	4	1	4	1	4	1	87,5	A	
	5	1	5	2	5	1	4	2	5	2	90	A	
	5	2	5	2	4	1	4	2	5	1	87,5	A	
	5	3	4	3	5	3	5	3	4	3	70	C	
	4	1	5	2	5	2	4	2	4	2	82,5	A	
	3	3	3	3	3	3	3	2	3	2	55	D	
	4	1	4	2	5	1	5	2	4	1	87,5	A	
	5	1	5	5	5	1	5	1	5	5	80	A	
	4	5	4	5	4	4	4	4	5	2	52,5	D	
	4	2	5	1	4	1	4	2	5	1	87,5	A	

	5	2	4	1	4	1	5	2	4	1	87,5	A
	5	1	4	3	4	2	4	1	5	1	85	A
	5	1	5	1	5	1	5	1	5	1	100	A
	5	2	5	2	5	2	5	2	5	2	87,5	A
	5	2	5	2	5	2	5	2	5	2	87,5	A
	4	1	5	2	4	2	4	1	5	2	85	A
	4	1	5	2	4	2	5	1	5	1	90	A
	5	5	5	4	5	5	5	5	5	4	55	D
	4	2	4	1	5	1	5	1	4	1	90	A
	4	2	4	2	5	1	5	2	5	5	77,5	B
	5	1	4	2	2	1	4	2	4	1	80	A
	5	1	5	1	5	1	5	1	5	1	100	A
	4	2	5	1	5	2	5	1	5	2	90	A
	4	1	4	1	4	1	4	1	4	1	87,5	A
	5	1	5	1	5	1	5	1	5	1	100	A
	4	2	4	1	5	1	4	1	4	2	85	A
	5	4	5	2	5	1	5	1	5	4	82,5	A

**[Informe de Pruebas Unitarias]**

**Desarrollo de una herramienta web para la  
Gestión de Requisitos de Software en la  
Carrera Computación de la Universidad  
Nacional de Loja**

*Elaborado por:*

*Edinson Roberto Quizhpe Vizcaíno*

*Revisado por:*

*Ing. Wilman Chamba*

*Aprobado por:*

*Ing. Wilman Chamba*

Mayo 2024



## Historial de versiones

Versión	Fecha	Responsable	Descripción de la modificación
03.05	03/05/2024	Edinson Roberto Quizhpe Vizcaíno	Documento inicial

## Información del Proyecto

Tabla 1. Herramienta para la Carrera Computación de la UNL.

<b>Unidad requirente</b>	Carrera Computación
<b>Nombre del proyecto</b>	Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la Universidad Nacional de Loja.
<b>Fecha de elaboración</b>	03-05-2024
<b>Solicitante / Patrocinador principal</b>	Carrera Computación de la Universidad Nacional de Loja



## 1. Introducción

En la presente, se describe la Especificación de Requisitos de Software (ERS) del proyecto **“Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la Universidad Nacional de Loja”**. La presente ERS se basa en las encuestas realizadas a los estudiantes de la Carrera Computación de la Universidad Nacional de Loja.

### 1.2. Descripción del proyecto

La Carrera Computación de la UNL, a favor de mejorar la claridad y estructura de los requisitos de software en los proyectos de los estudiantes, prevé la necesidad de desarrollar una herramienta que permita gestionar los requisitos de software, para mejorar los procesos que se realizan manualmente o sin una herramienta específica.

### 1.3. Objetivo

El proyecto “Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la UNL” tiene como alcance desarrollar una solución que permita cubrir las siguientes necesidades:

- Registro de proyectos.
- Registro de requisitos.
- Agregar colaboradores.
- Historial de cambios.
- Generar reportes.
- Permisos y roles de usuario.

### 1.3. Referencias

Como referencias del proyecto *“Desarrollo de una herramienta web para la Gestión de Requisitos de Software en la Carrera Computación de la UNL”* se tiene:



Tabla 2. Referencias

Referencia	Título	Ruta
Herramienta	Postman	<a href="https://www.postman.com/">https://www.postman.com/</a>

## 1.4. Ejecución de Pruebas

Para la ejecución de las pruebas unitarias se empleó la herramienta Postman, la cual permitió la solicitud de peticiones HTTP de los servicios web implementados en el servidor backend, las cuales devuelven un JSON como respuesta.

A continuación, en la tabla 3, se muestra la ejecución de las solicitudes realizadas al servidor.

Tabla 3. Ejecución de pruebas

URL	Token	Método	Descripción	Tiempo (respuesta)
http://localhost:3334/grs/api/v1/auth/signup	NO	POST	Registro de usuario	2.19s

```

1 {
2   "id": "66e10bede2536cf8940ba364",
3   "firstName": "User test 1",
4   "lastName": "Tester 1",
5   "email": "user.test@unl.edu.ec",
6   "createdAt": "2024-09-11T03:18:05.444Z",
7   "updatedAt": "2024-09-11T03:18:05.444Z"
8 }

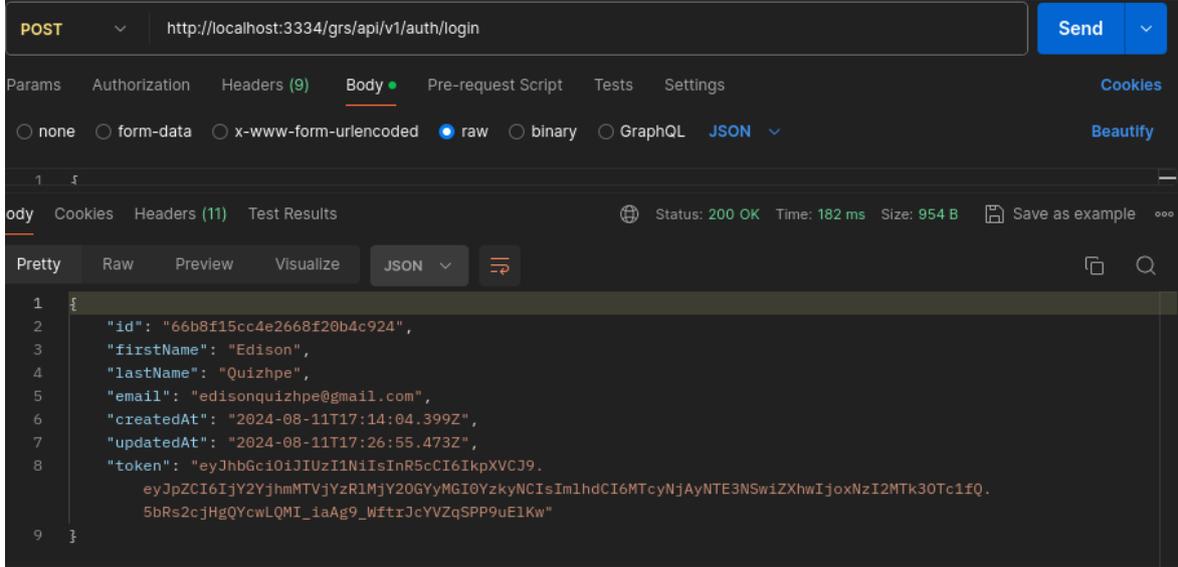
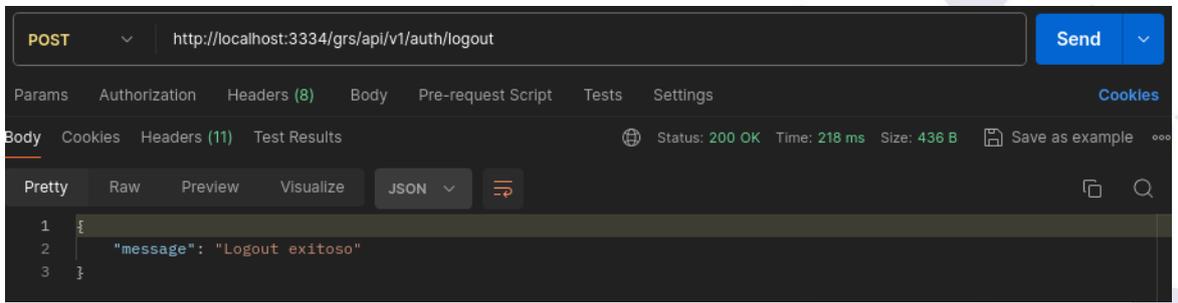
```



UNL

Universidad  
Nacional  
de Loja

1859

http://localhost:3334/grs/api/v1/auth/login	NO	POST	Inicio de sesión	182ms
				
http://localhost:3334/grs/api/v1/auth/logout	NO	POST	Cierre de sesión	218ms
				
http://localhost:3334/grs/api/v1/projects/user/	SI	GET	Obtener proyectos del usuario	200 ms



1859



Universidad Nacional de Loja

```

GET http://localhost:3334/grs/api/v1/projects/user/
Status: 200 OK Time: 200 ms Size: 2.74 KB
Body
JSON
[
  {
    "_id": "66b8f1a7ab16d9e40d54deec",
    "nameProject": "Proyecto de desarrollo de una tienda de mascotas",
    "objGeneral": "objetivo 1",
    "objSpecific": "objetivo 2",
    "created": "2024-08-01T05:00:00.000Z",
    "endDate": "2024-08-30T05:00:00.000Z",
    "user": "66b8f15cc4e2668f20b4c924",
    "actors": [
      {
        "name": "Desarrollador",
        "description": "codifica la aplicación",
        "_id": "66b8f1edbc2728bff2044879"
      }
    ],
    "projectRoles": [],
  }
]

```

http://localhost:3334/grs/api/v1/projects/66e1238ae40a328a46742bd4	SI	GET	Obtener un proyecto por id	107ms
--	----	-----	----------------------------	-------

```

GET http://localhost:3334/grs/api/v1/projects/66e1238ae40a328a46742bd4
Status: 200 OK Time: 107 ms Size: 665 B
Body
JSON
{
  "_id": "66e1238ae40a328a46742bd4",
  "nameProject": "Project Test 1",
  "objGeneral": "Objetivo 1.1",
  "objSpecific": "objetivo 2.1",
  "user": "66b8f15cc4e2668f20b4c924",
  "actors": [
    {
      "name": "Vendedor",
      "description": "Realiza ventas de productos",
      "_id": "66e70f229d8ea3a2866538d6"
    }
  ],
  "projectRoles": [],
  "requirements": [],
  "active": true,
  "__v": 1
}

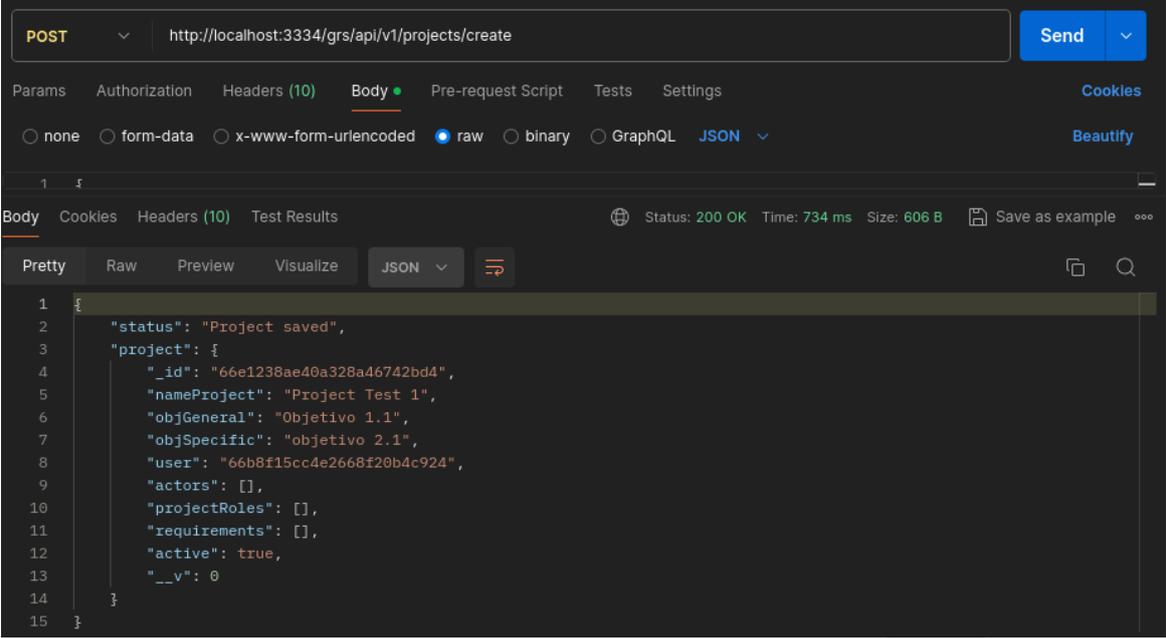
```



1859

UNL

Universidad Nacional de Loja

http://localhost:3334/grs/api/v1/projects/create	SI	POST	Crear nuevo proyecto	734ms
				
http://localhost:3334/grs/api/v1/projects/update/66b8f1a7ab16d9e40d54deec	SI	PUT	Actualizar un proyecto	120ms



1859



Universidad Nacional de Loja

```

1  {
2    "status": "Project updated",
3    "project": {
4      "_id": "66e7398b9d8ea3a2866538e5",
5      "nameProject": "Project Test 3.1.2",
6      "objGeneral": "Objetivo 1.2.1",
7      "objSpecific": "objetivo 2.1",
8      "user": "66b8f15cc4e2668f20b4c924",
9      "actors": [],
10     "projectRoles": [],
11     "requirements": [],
12     "active": true,
13     "__v": 0
14   }
15 }

```

http://localhost:3334/grs/api/v1/project-roles/all/66b8f1a7ab16d9e40d54deec	SI	GET	Obtener roles del proyecto	90ms
---	----	-----	----------------------------	------

```

1  [
2    {
3      "_id": "66b8f1a7ab16d9e40d54def3",
4      "name": "Analista de negocios",
5      "description": "",
6      "projectId": "66b8f1a7ab16d9e40d54deec",
7      "__v": 0
8    },
9    {
10     "_id": "66b8f1a7ab16d9e40d54def2",
11     "name": "Validador",
12     "description": "",
13     "projectId": "66b8f1a7ab16d9e40d54deec",
14     "__v": 0
15   },
16   {
17     "_id": "66b8f1a7ab16d9e40d54def4",
18     "name": "Gerente de proyecto",
19     "description": ""

```

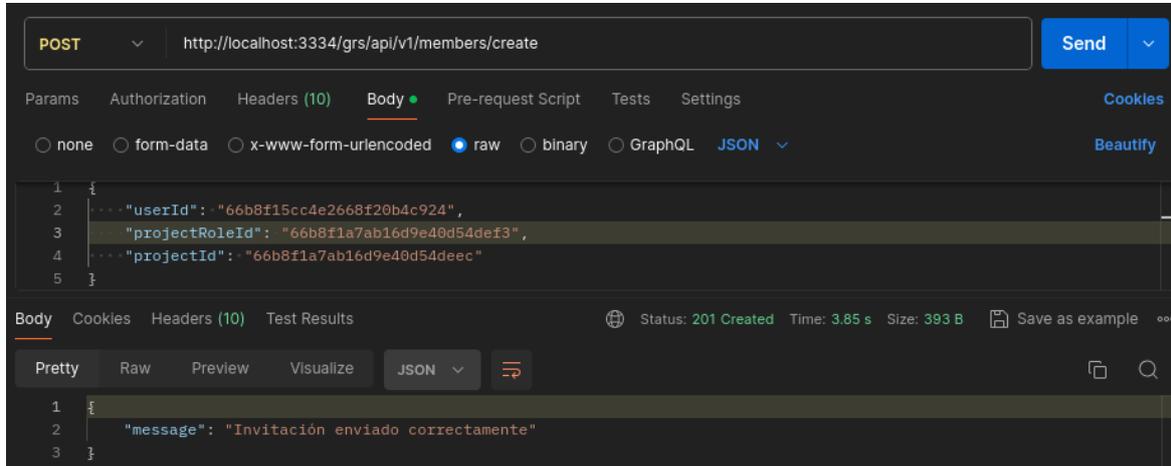


UNL

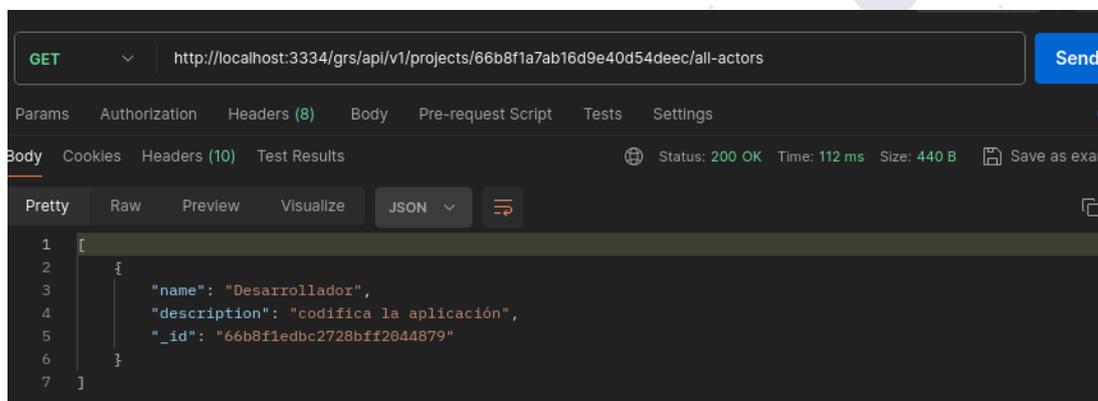
Universidad Nacional de Loja

1859

http://localhost:3334/grs/api/v1/members/create	SI	POST	Agregar nuevo colaborador a un proyecto	3.85s
---	----	------	---	-------



http://localhost:3334/grs/api/v1/projects/66b8f1a7ab16d9e40d54deec/all-actors	SI	GET	Obtiene todos los actores de un proyecto	112ms
---	----	-----	--	-------



http://localhost:3334/grs/api/v1/projects/66e1238ae40a	SI	POST	Crear nuevo actor de un proyecto	250ms
--	----	------	----------------------------------	-------



UNL

Universidad Nacional de Loja

1859

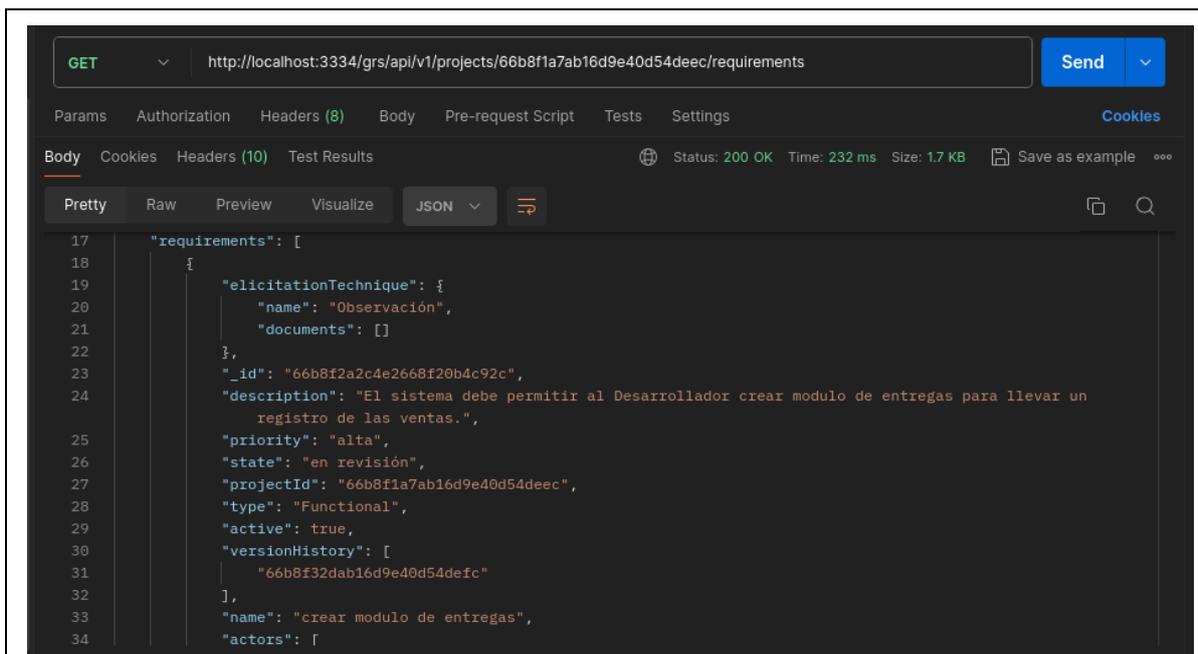
328a46742bd4/cr eate-actor				
-------------------------------	--	--	--	--

```

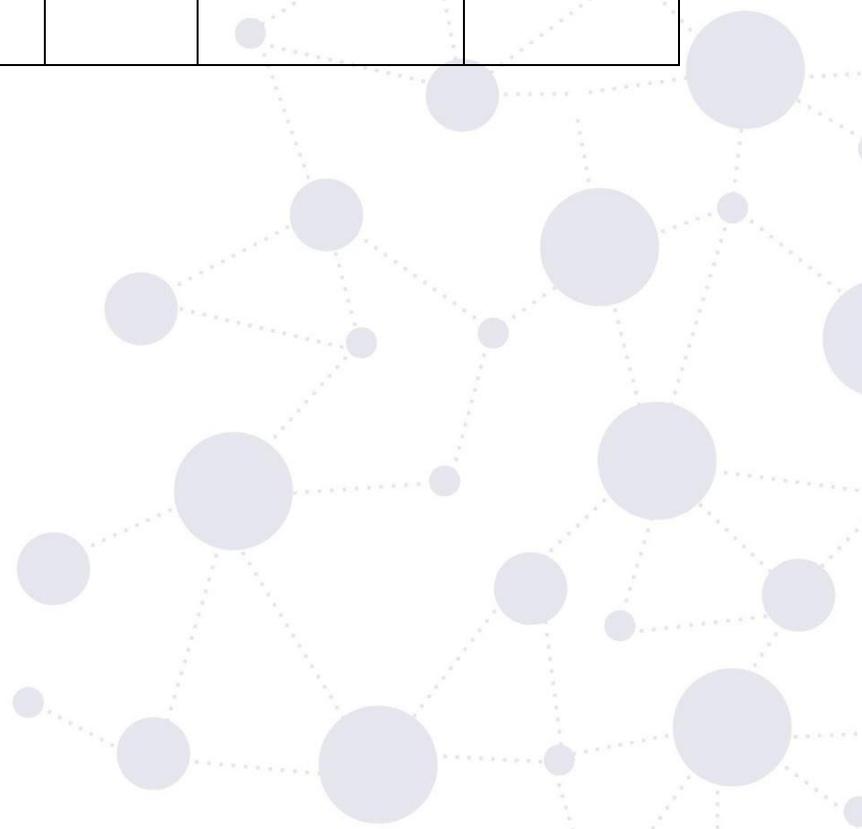
1  {
2    "status": "Actor added",
3    "project": {
4      "_id": "66e1238ae40a328a46742bd4",
5      "nameProject": "Project Test 1",
6      "objGeneral": "Objetivo 1.1",
7      "objSpecific": "Objetivo 2.1",
8      "user": "66b8f15cc4e2668f20b4c924",
9      "actors": [
10     {
11       "name": "Vendedor",
12       "description": "Realiza ventas de productos",
13       "_id": "66e70f229d8ea3a2866538d6"
14     }
15   ],
16   "projectRoles": [],
17   "requirements": [],
18   "active": true,
19   "v": 1

```

http://localhost:33 34/grs/api/v1/proje cts/66b8f1a7ab16d 9e40d54deec/req uirements	SI	GET	Obtener todos los requisitos de un proyecto	232ms
---	----	-----	---	-------



http://localhost:3334/grs/api/v1/requirements/66b8f2a2c4e2668f20b4c92c	SI	GET	Obtener un requisito por su ID.	108ms
--	----	-----	---------------------------------	-------





```

1 {
2   "elicitationTechnique": {
3     "name": "Observación",
4     "documents": []
5   },
6   "_id": "66b8f2a2c4e2668f20b4c92c",
7   "description": "El sistema debe permitir al Desarrollador crear modulo de entregas para llevar un registro de las ventas.",
8   "priority": "alta",
9   "state": "en revisión",
10  "projectId": "66b8f1a7ab16d9e40d54deec",
11  "type": "Functional",
12  "active": true,
13  "versionHistory": [
14    "66b8f32dab16d9e40d54defc"
15  ],
16  "name": "crear modulo de entregas",
17  "actors": [
18    null

```

http://localhost:3334/grs/api/v1/requirements/create/	SI	POST	Crear un nuevo requisito en un proyecto	762ms
---	----	------	---	-------

POST http://localhost:3334/grs/api/v1/requirements/create/

Body: form-data

<input checked="" type="checkbox"/>	name	Text	Subir documentos
<input checked="" type="checkbox"/>	description	Text	Gestión de documentos en proyectos
<input checked="" type="checkbox"/>	priority	Text	alta
<input checked="" type="checkbox"/>	state	Text	en revisión
<input checked="" type="checkbox"/>	projectId	Text	66b8f1a7ab16d9e40d54deec
<input checked="" type="checkbox"/>	type	Text	Functional
<input checked="" type="checkbox"/>	elicitationTechnique[name]	Text	Historia de usuario

Status: 201 Created Time: 762 ms Size: 387 B

```

1 {
2   "message": "Requisito creado con éxito"
3 }

```



UNL

Universidad Nacional de Loja

1859

http://localhost:3334/grs/api/v1/requirements/update/66b8f2a2c4e2668f20b4c92c	SI	PUT	Actualizar un requisito	717ms
---	----	-----	-------------------------	-------

The screenshot shows a REST client interface with the following details:

- Method:** PUT
- URL:** http://localhost:3334/grs/api/v1/requirements/update/66b8f2a2c4e2668f20b4c92c
- Status:** 200 OK
- Time:** 717 ms
- Size:** 913 B
- Response Body (JSON):**

```

1 {
2   "message": "Requisito actualizado con éxito",
3   "updatedRequirement": {
4     "elicitationTechnique": {
5       "documents": [],
6       "name": "Historia de usuario"
8     },
9     "_id": "66b8f2a2c4e2668f20b4c92c",
10    "description": "Gestión y administración de proyectos",
11    "priority": "alta",
12    "state": "en revisión",
13    "projectId": "66b8f1a7ab16d9e40d54deec",
14    "type": "Functional",
15    "active": true,
16    "versionHistory": [
17      "66b8f32dab16d9e40d54defc",
18      "66e756d8891d774ee2d740f0"
19    ],
20    "name": "Registro de proyecto".

```

http://localhost:3334/grs/api/v1/requirements/versions/66b8f2a2c4e2668f20b4c92c	SI	GET	Obtener las versiones de un requisito	412ms
---	----	-----	---------------------------------------	-------



```

1 {
2   "message": "Versiones del requisito obtenidas con éxito",
3   "versions": [
4     {
5       "elicitationTechnique": {
6         "name": "Historia de usuario",
7         "documents": []
8       },
9       "_id": "66b8f2a2c4e2668f20b4c92c",
10      "description": "Gestión y administración de proyectos",
11      "priority": "alta",
12      "state": "en revisión",
13      "projectId": "66b8f1a7ab16d9e40d54deec",
14      "type": "Functional",
15      "active": true,
16      "versionHistory": [
17        "66b8f32dab16d9e40d54defc",
18        "66e756d8891d774ee2d740f0"
19      ]
20    }
21  ]
22 }

```

http://localhost:3334/grs/api/v1/users/user-profile	SI	GET	Obtener perfil de usuario	105ms
---	----	-----	---------------------------	-------

```

1 {
2   "_id": "66b8f15cc4e2668f20b4c924",
3   "firstName": "Edison",
4   "lastName": "Quizhpe",
5   "email": "edisonquizhpe@gmail.com",
6   "email_2": "edisonquizhpe@gmail.com",
7   "phone": "0979360585",
8   "bornDate": "1998-09-22T05:00:00.000Z",
9   "admin": false,
10  "userRole": "66b8eede72334576ff92e963",
11  "createdAt": "2024-08-11T17:14:04.399Z",
12  "updatedAt": "2024-08-11T17:26:55.473Z",
13  "__v": 0
14 }

```



http://localhost:3334/grs/api/v1/users/search/:searchTerm	SI	GET	Buscar usuario por término de búsqueda	101ms
---	----	-----	--	-------

```

14  {
15    "_id": "66b8f15cc4e2668f20b4c924",
16    "firstName": "Edison",
17    "lastName": "Quizhpe",
18    "email": "edisonquizhpe@gmail.com",
19    "email_2": "edisonquizhpe@gmail.com",
20    "phone": "0979360585",
21    "bornDate": "1998-09-22T05:00:00.000Z",
22    "admin": false,
23    "userRole": "66b8eede72334576ff92e963",
24    "createdAt": "2024-08-11T17:14:04.399Z",
25    "updatedAt": "2024-08-11T17:26:55.473Z",
26    "password": "$2b$10$fJDaeOUPXRIVV1S7sCsVDeDI0arHw.imEArojt3CCXrYyDvL11Fhe",
27    "__v": 0
28  },
29  {
30    "_id": "66ba83b7ab16d9e40d54df12",
31    "firstName": "Lourdes Marisol",
32    "lastName": "Viñamagua Pasuy",
  
```

http://localhost:3334/grs/api/v1/profile/recovery-password	NO	POST	Reestablecer contraseña olvidada	2.61s
--	----	------	----------------------------------	-------



1859



Universidad Nacional de Loja

POST http://localhost:3334/grs/api/v1/profile/recovery-password

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

1 {
2   "email": "edisonquizhpe@gmail.com"
3 }

```

Body Cookies Headers (10) Test Results Status: 200 OK Time: 2.61 s Size: 399 B Save as example

Pretty Raw Preview Visualize JSON

```

1 {
2   "msg": "Se ha enviado un correo con la nueva contraseña"
3 }

```

http://localhost:3334/grs/api/v1/profile/update-password	SI	PUT	Actualizar/Modificar contraseña de usuario	3.26s
--	----	-----	--	-------

PUT http://localhost:3334/grs/api/v1/profile/update-password

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

1 {
2   "password": "@ed150n!2209@"
3 }

```

Body Cookies Headers (10) Test Results Status: 200 OK Time: 3.26 s Size: 399 B Save as example

Pretty Raw Preview Visualize JSON

```

1 {
2   "msg": "La contraseña ha sido actualizada correctamente"
3 }

```



## 2. Glosario de términos

### Definiciones, Acrónimos y abreviaturas

**UNL:** Universidad Nacional de Loja

**RF:** Requisito Funcional

**RNF:** Requisito No Funcional

**RN:** Requerimientos del negocio

**RU:** Requerimiento del usuario

**RS:** Requerimiento del software

## 3. Aprobaciones

Nombre y Apellido	Cargo / Rol	Departamento	Firma
Ing. Wilman Patricio Chamba Zaragocín	Director del trabajo de integración curricular	Carrera Computación	
Edinson Roberto Quizhpe Vizcaíno	Tesista	Carrera Computación	

## Anexo 7. Revisión de antiplagio



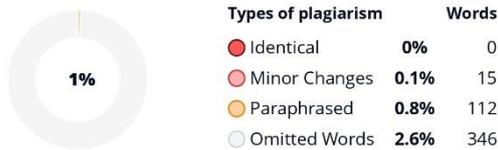
Plagiarism and AI Content Detection Report

### MEMORIA FINAL PARA REVISIÓN EDI...

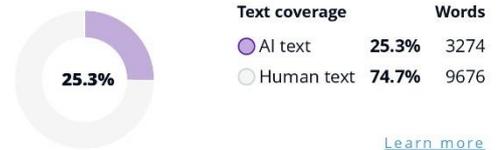
#### Scan details

Scan time: September 17th, 2024 at 12:26 UTC      Total Pages: 54      Total Words: 13296

#### Plagiarism Detection



#### AI Content Detection



#### Plagiarism Results: (2)

**Glosario de usabilidad y accesibilidad web** **0.7%**  
[https://www.usableyaccesible.com/recurso\\_glosario.php](https://www.usableyaccesible.com/recurso_glosario.php)  
Ir al contenido Menú principal: blog sobre mí ser...

**Documento\_completo.pdf?sequence=1&isAllowed=y** **0.2%**  
[http://sedici.unlp.edu.ar/bitstream/handle/10915/121741/documento\\_completo.pdf?sequence=1&isallowed=y](http://sedici.unlp.edu.ar/bitstream/handle/10915/121741/documento_completo.pdf?sequence=1&isallowed=y)  
Pablo Claus  
Klein: Proceso de desarrollo de software con trazabilidad de requerimientos, documentación mínima y aplicable con equipos pequeños y dist...

**Anexo 8. Certificado de traducción del resumen.**

**CERTIFICADO DE TRADUCCION**

**Lcdo. Luis Hernán Sánchez Villa**  
**Licenciado en Ciencias de la Educación, especialización Idioma Inglés**

**CERTIFICO:**

Que he realizado la traducción del idioma español al idioma inglés, del resumen derivado de la tesis denominada "DESARROLLO DE UNA HERRAMIENTA WEB PARA LA GESTIÓN DE REQUISITOS DE SOFTWARE EN LA CARRERA COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL DE LOJA" de autoría del señor Edinson Roberto Quizhpe Vizcaino, con cedula de identidad número 1150299483, estudiante de la Carrera de Ingeniería en Computación de la facultad de Energía, las Industrias y los Recursos Naturales no Renovables de la Universidad Nacional de Loja.

Es todo cuanto puedo certificar en honor a la verdad, facultando a la interesada, hacer uso del presente, en lo que considere pertinente.



---

**Lcdo. Luis Hernán Sánchez Villa**  
**CI: 1102404314**  
**Senescyt: 1008-02-154120**