

Desarrollo de un Prototipo para Gestionar las
Propuestas de Extensión no Solidarias asociadas a la
Vicerrectoría de Investigación y Extensión (VIE) de la
Universidad Industrial de Santander

Brayan Sneider Daza Suarez y Andrea Juliana Urrego Paredes

Trabajo de grado para optar al título de ingeniero de sistemas

Director

Msc. Emilio Justiniano Cárcamo Troconis

Escuela de Ingeniería de Sistemas e Informática

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingeniería de Sistemas e Informática

Bucaramanga

2024

Dedicatoria

A mi abuela María, quien siempre soñó con verme graduada.

*Hoy estoy cumpliendo su sueño, aunque no esté presente en este mundo terrenal,
pero sé que está muy feliz viéndome lograrlo.*

*A mi madre y a mis tías, porque son el pilar y la motivación más
grande con la que siempre he contado para sacar
adelante mi carrera y mi vida.*

*A mi familia, quienes han demostrado ser muy influyentes en
mis deseos de ser mejor cada día.*

*A mi compañero de proyecto, Brayan Daza, por el arduo
trabajo que llevó a mi lado y por todos los obstáculos superados.*

*A mis amigos, quienes han sido mi apoyo incondicional en este
viaje. Gracias por estar siempre ahí
para animar y celebrar mis logros.*

*A los profesores de la Universidad Industrial de Santander, por
enseñarme que no hay que desfallecer
aunque el reto sea demasiado grande.*

Andrea Juliana Urrego Paredes

Dedicatoria

*A mi madre quien ha sido un apoyo enorme e inspiración en mi
proceso de desarrollo profesional.*

*A mi tío Jairo quien no solo me dio la mano y el apoyo que
necesitaba para salir adelante sino quien me ha enseñado
y mostrado cómo superar los distintos obstáculos de la vida,
una fuente de inspiración y dedicación alguien a quien
admiro mucho por lo que ha conseguido y como lo ha hecho, le
estaré profundamente agradecido toda mi vida.*

*A mi compañera Juliana quien ha estado a mi lado en todo este
proyecto apoyándome, sacando lo mejor de mí
y enseñándome que lo que es una amistad sincera,
te quiero.*

Brayan Sneider Daza Suarez

Agradecimientos

Agradecemos a nuestro director de tesis, Emilio Justiniano Cárcamo Troconis, por su invaluable orientación, sabiduría y apoyo durante todo el proceso de desarrollo de este proyecto. Su experiencia y dedicación fueron fundamentales para alcanzar los objetivos propuestos.

A nuestros compañeros de clase y amigos, por su compañía, motivación y apoyo constante a lo largo de esta travesía académica.

Agradecemos especialmente a la Universidad Industrial de Santander y a RSI por brindarnos la oportunidad de formarnos académicamente y por proporcionarnos los recursos necesarios para llevar a cabo este proyecto.

Finalmente, a nuestras familias por su amor incondicional, comprensión y apoyo en cada paso de este camino. Su sacrificio y aliento fueron la fuerza que nos impulsó a seguir adelante en los momentos más difíciles.

Contenido

Introducción	14
1. Planteamiento del problema.....	15
2. Objetivos	16
2.1 Objetivo general.....	16
2.2 Objetivos específicos	16
3. Marco de referencia	18
3.1 Estado actual del sistema	18
3.2 Fundamentos teóricos	19
3.2.1 Capa microfrontend	20
3.2.2 Capa microservicios.....	20
3.2.3 Capa de base de datos	20
3.2.4 Control de versiones y calidad de código	21
4. Metodología	23
4.1 Marco ágil de trabajo (SCRUM).....	23
4.2 Fases del proyecto.....	26
4.2.1 Fase 1: Análisis del proyecto y planeación.....	26

4.2.2 Fase 2: definición de requerimientos.....	27
4.2.3 Fase 3: Desarrollo y documentación	28
4.2.4 Fase 4: Evaluación de calidad y prototipo	29
4.2.5 Fase 5: Finalización y entrega.....	30
5. Resultados	31
5.1 Fase 1: Análisis del proyecto y planeación.....	31
5.1.1 Análisis de viabilidad	31
5.1.2 Definición de alcance	32
5.2 Fase 2: Definición de requerimientos	34
5.2.1 Análisis de documentación existente.....	34
5.2.2 Entrevistas y encuestas	36
5.2.3 Priorización de requerimientos.....	37
5.2.4 Identificación de requerimientos funcionales y no funcionales.....	38
5.2.4.1 Requerimientos generales de permisos, accesos y gestión de propuestas	38
5.2.4.2 Requerimientos sobre el manejo de la información básica de una propuesta de extensión.....	42
5.2.4.3 Requerimientos sobre el manejo de las características del proyecto	42
5.2.4.4 Requerimientos sobre la información de soporte de una propuesta de extensión.....	42

5.2.4.5 Requerimientos sobre el destinatario de una propuesta de extensión.....	43
5.2.5 Revisión y aprobación por parte del cliente.....	43
5.3 Fase 3: Desarrollo y documentación.....	43
5.3.1 Configuración de los entornos de desarrollo	43
5.3.1.1 Configuración de entorno backend	44
5.3.1.2 Configuración de entorno frontend.....	45
5.3.2 Rediseño de pantallas por parte del equipo de experiencia de usuario (UX).	46
5.3.3 Desarrollo SCRUM	48
5.4 Fase 4: Evaluación de calidad y control del prototipo	56
5.5 Fase 5: Finalización y entrega.....	60
6. Conclusiones.....	63
7. Trabajo futuro	64
Bibliografía	65

Lista de figuras

Figura 1. Arquitectura del prototipo. Fuente (Autores del proyecto).	19
Figura 2. ¿Qué es CI/CD? Fuente (Synopsys, 2022).	22
Figura 3. ¿Qué es Scrum? Fuente (Slide Egg).	25
Figura 4. Pantalla de consulta sistema actual VIE. Fuente (Manual y documentación oficial). ..	35
Figura 5. Diagrama de BPMN para la gestión de propuestas de extensión. Fuente (Análisis y documentación).	36
Figura 6. Estructura de repositorios backend. Fuente (Desarrollo y documentación).	45
Figura 7. Diseño para la consulta y gestión de propuestas de extensión. Fuente (Análisis y documentación).	47
Figura 8. Backlog del sprint 1. Fuente (Taiga). [17]	52
Figura 9. Repositorio Back Servicios. Fuente (Gitlab UIS). [18]	55
Figura 10. Repositorio Back Propuestas. Fuente (Gitlab UIS). [19]	55
Figura 11. Informe del código actual en SonarQube para el repositorio Back Propuestas. Fuente (SonarQube - SIA UIS).	58
Figura 12. Informe del código actual en SonarQube para el repositorio Back Servicios. Fuente (SonarQube - SIA UIS).	58
Figura 13. Documentación Swagger para Back Servicios. Fuente (Swagger - SIA UIS).	59
Figura 14. Documentación Swagger para Back Propuestas. Fuente (Swagger - SIA UIS).	60
Figura 15. Menús. Fuente (Sistemas de información - SIA UIS).	61
Figura 16. Menús. Fuente (Sistemas de información - SIA UIS).	62

Lista de tablas

Tabla 1. Requerimientos funcionales de permisos, accesos y gestión de propuestas.	38
Tabla 2. Requerimientos no funcionales de permisos, accesos y gestión de propuestas.	41
Tabla 3. Sprint 1.	49
Tabla 4. Sprint 2.	50
Tabla 5. Sprint 3.	51

Lista de apéndices

Apéndice A. Pantallas del sistema actual vie extensión 68

Apéndice B. Identificación de requerimientos funcionales y no funcionales..... 71

Apéndice C. Pantallas diseñadas para el nuevo sistema 79

Apéndice D. Sprints desarrollados..... 82

Glosario

Marco ágil: Se refiere a un enfoque metodológico para el desarrollo de software que se centra en la flexibilidad, la colaboración y la entrega rápida de productos de alta calidad.

BPMN (Business process model and notation): Es un estándar de modelado de procesos de negocio. Proporciona una notación gráfica para representar los procesos de negocio de manera clara y comprensible tanto para los usuarios técnicos como no técnicos.

Requerimientos funcionales: Se refieren a las características y funcionalidades específicas que el sistema debe proporcionar.

Requerimientos no funcionales: Se centran en aspectos como la seguridad, el rendimiento y la usabilidad.

Dev-test: Pruebas de desarrollo que se realizan para evaluar si se cumplen los criterios de aceptación de cada historia de usuario.

Criterios de aceptación: Conjunto de condiciones que deben ser satisfechas para considerar que una funcionalidad está completa y cumple con las expectativas del usuario final.

Resumen

Título: Desarrollo de un prototipo para la gestión de propuestas de extensión en la Universidad Industrial de Santander *

Autores: Brayan Sneider Daza Suarez y Andrea Juliana Urrego Paredes **

Palabras clave: Gestión de propuestas, extensión universitaria, desarrollo de prototipo, Universidad Industrial de Santander, metodología ágil.

Descripción: Este proyecto se centra en el desarrollo de un prototipo para la gestión de propuestas de extensión en la Universidad Industrial de Santander (UIS). Siguiendo una metodología como Scrum, se abordaron diversas fases, desde el análisis inicial hasta la implementación final del modelo. Se identificaron requerimientos funcionales y no funcionales esenciales, abarcando el manejo de la información básica, características del proyecto, soporte de la propuesta, gestión de destinatarios y aspectos de permisos y accesos. El prototipo facilita a usuarios con distintos roles el registro, consulta, edición y gestión de propuestas de extensión de manera eficiente. Entre las funcionalidades destacadas se incluyen la selección de servicios de extensión, validación de campos obligatorios, gestión de recursos físicos e interacción dinámica con el formulario. La interfaz del sistema está diseñada para ser fácil de usar, garantizando tiempos de respuesta rápidos y una experiencia de usuario coherente y amigable. Con este prototipo, se busca mejorar significativamente el proceso de registro y gestión de propuestas en la UIS, facilitando el trabajo de los docentes y administradores, y asegurando una gestión más organizada y eficiente de las iniciativas de extensión universitaria, proporcionando un sistema más efectivo y accesible.

* Trabajo de investigación

** Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería de Sistemas e Informática.
Director: Msc. Emilio Justiniano Cárcamo Troconis.

Abstract

Title: Development of a prototype for the management of extension proposals at the Industrial University of Santander.

Authors: Brayan Sneider Daza Suarez and Andrea Juliana Urrego Paredes.

Keywords: Proposal management, university extension, prototype development, Industrial University of Santander, agile methodology.

Description: This project focuses on developing a prototype for managing extension proposals at the Universidad Industrial de Santander (UIS). Following an agile methodology such as Scrum, various phases were addressed, from the initial analysis to the final implementation of the model. Essential functional and non-functional requirements were identified, encompassing the management of basic information, project characteristics, proposal support, recipient management, and aspects of permissions and access. The prototype facilitates users with different roles to register, consult, edit and manage extension proposals efficiently. Notable features include the selection of extension services, validation of mandatory fields, management of physical resources, and dynamic interaction with the form. The system interface is designed to be intuitive and easy to use, ensuring fast response times and a coherent and user-friendly experience. This prototype aims to significantly improve the registration and management process of proposals at UIS, facilitating the work of teachers and administrators, and ensuring a more organized and efficient management of university extension initiatives, providing a more effective and accessible system.

* Research Work

** Faculty of Physics-Mechanics Engineering. School of Systems Engineering and Computer Science. Advisor: Msc. Emilio Justiniano Cárcamo Troconis.

Introducción

En el contexto de la Universidad Industrial de Santander, la Vicerrectoría de Investigación y Extensión (VIE) juega un papel clave en la promoción del desarrollo social, cultural y académico de la comunidad. La extensión universitaria se presenta como un vínculo fundamental entre la institución educativa y su entorno, posibilitando la transferencia de conocimientos y el enriquecimiento mutuo.

Este proyecto de grado surge como respuesta a la necesidad de renovar y optimizar el sistema de gestión de propuestas de extensión. El objetivo del sistema no es solo agilizar el proceso y mejorar la calidad, sino también fomentar la colaboración entre profesores, administradores y jefes de unidades académicas y administrativas, quienes juegan un papel clave en la elaboración, evaluación y aprobación de estas propuestas. Se analiza en profundidad la problemática actual y se señalan las falencias existentes en el proceso administrativo y en la elaboración de propuestas de extensión. Además, se presenta el marco de referencia y la metodología que sustenta el planteamiento de esta tesis.

En general, este proyecto tiene como objetivo desarrollar un nuevo sistema de información para la gestión de propuestas de extensión siguiendo estándares de desarrollo de RSI para la renovación de los sistemas de información de la universidad, proporcionando un enfoque más flexible, colaborativo y eficiente. Se espera que este sistema optimice y mejore el proceso de elaboración y gestión de las propuestas de extensión.

1. Planteamiento del problema

En la Vicerrectoría de Investigación y Extensión de la Universidad Industrial de Santander, el sistema actual utilizado para gestionar propuestas de extensión presenta falencias en usabilidad y rendimiento que afectan significativamente su eficiencia y efectividad. Los procesos en el sistema son tediosos y ciertas tareas aún se realizan de manera manual, lo que provoca una dispersión de la información y dificulta la interacción de los usuarios debido a la falta de una interfaz intuitiva. Estas deficiencias tienen un impacto negativo en la gestión de las propuestas y en la colaboración entre los actores involucrados, generando tiempos de atención y respuesta más prolongados de lo esperado, lo que afecta la productividad y eficacia del sistema.

En el marco del macroproyecto "Renovación de los Sistemas de Información" (RSI), se ha identificado la necesidad de desarrollar un nuevo sistema de propuestas de extensión en la Vicerrectoría de Investigación y Extensión de la Universidad Industrial de Santander. El objetivo de este proyecto de grado es crear una solución funcional que cumpla con los estándares y arquitectura establecidos por RSI, con el propósito de agilizar los procesos, optimizar la gestión de la información y promover la colaboración entre los diferentes actores involucrados. De esta manera, se espera mejorar la eficiencia administrativa y la toma de decisiones en la Vicerrectoría.

En consecuencia, la pregunta de investigación que se plantea es: ¿Cómo diseñar e implementar un nuevo sistema de propuestas de extensión para la Vicerrectoría de Investigación y Extensión que supere las deficiencias del sistema actual, garantizando una gestión de propuestas y fomentando la colaboración efectiva entre los actores involucrados, para mejorar los tiempos de acción en las diferentes tareas del proceso?

2. Objetivos

2.1 Objetivo general

- Desarrollar un prototipo de aplicativo web que permita la administración de las propuestas de extensión no solidarias asociadas a la Vicerrectoría de Investigación y Extensión.

2.2 Objetivos específicos

1. Establecer los requerimientos para administrar propuestas de extensión no solidarias de la VIE, los cuales abarcan la creación de nuevas propuestas, la selección del servicio a relacionar con las mismas, la trazabilidad del proceso y la identificación de roles y unidades.
2. Implementar un prototipo de aplicación web en sus capas Frontend y Backend que se ajuste a los requerimientos previamente establecidos.
3. Validar el prototipo a través de un plan de pruebas de calidad.

Esta tabla muestra cómo cada objetivo específico se alinea con una fase específica del proyecto. Cada fase aborda un conjunto particular de tareas y actividades que contribuyen al logro de los objetivos establecidos.

Objetivo	Fase en que se cumple
Establecer los requerimientos para administrar propuestas de extensión no solidarias de la VIE, los cuales abarcan la creación de nuevas propuestas, la selección del servicio a relacionar con las mismas, la trazabilidad del proceso y la identificación de roles y unidades.	5.2.4 Fase 2: definición de requerimientos
Implementar un prototipo de aplicación web en sus capas Frontend y Backend que se ajuste a los requerimientos previamente establecidos.	5.3. Fase 3: desarrollo y documentación
Validar el prototipo a través de un plan de pruebas de calidad.	5.4 Fase 4: Evaluación de calidad y control de prototipo

3. Marco de referencia

En el contexto de este proyecto de investigación, es importante comprender tanto el estado actual del sistema como los fundamentos teóricos que sustentan el desarrollo de un prototipo para la gestión de propuestas de extensión en la Universidad Industrial de Santander.

3.1 Estado actual del sistema

El sistema actual en la Vicerrectoría de Investigación y Extensión (VIE) emplea una combinación de tecnologías, principalmente **Java** y **JSP (JavaServer Pages)**, para su funcionamiento.

Java, reconocido por su versatilidad, se utiliza en la lógica empresarial del sistema actual de la VIE. Por otro lado, JSP permite la creación de páginas web dinámicas al combinar código Java con contenido HTML, facilitando mostrar información en tiempo real. Sin embargo, aunque ha sido ampliamente utilizado, JSP puede percibirse como complejo y menos eficiente en comparación con tecnologías más modernas de desarrollo web debido a que su mantenimiento y desarrollo a menudo requieren un profundo conocimiento de Java, lo que puede resultar laborioso.

Además, el sistema actual utiliza **PostgreSQL**, un sistema de gestión de bases de datos relacional de código abierto, reconocido por su confiabilidad y capacidad para manejar grandes volúmenes de datos. Sin embargo, esta opción puede ser reconsiderada en busca de características más avanzadas o soluciones más adecuadas para las necesidades específicas de almacenamiento y recuperación de datos del desarrollo web moderno.

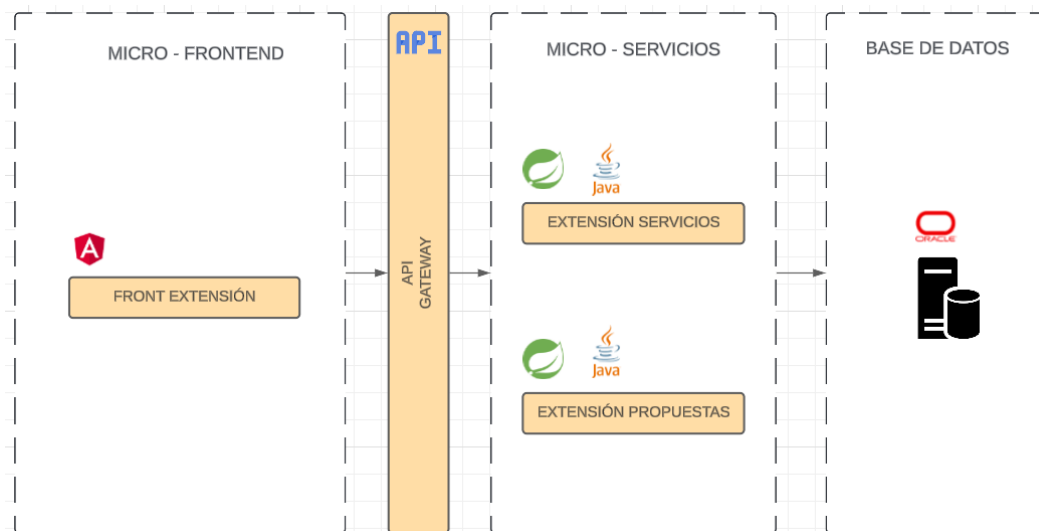
3.2 Fundamentos teóricos

Estos fundamentos teóricos proporcionan la base necesaria para comprender las tecnologías y la arquitectura a utilizar en el desarrollo del prototipo de gestión de propuestas de extensión.

La arquitectura del proyecto se fundamenta en una **arquitectura basada en microservicios** con principios de API REST, descomponiendo el sistema en unidades pequeñas y autónomas, cada una cumpliendo una función específica. Estos microservicios se comunican entre sí de manera ágil y eficiente. Esta arquitectura permite escalar cada microservicio de forma independiente, mejorando la capacidad de respuesta y disponibilidad del sistema. Además, facilita la implementación continua al tener despliegues pequeños y bien definidos, agilizando el ciclo de desarrollo y permitiendo la entrega rápida de nuevas funcionalidades. [1]

Figura 1.

Arquitectura del prototipo. Fuente (Autores del proyecto).



3.2.1 Capa microfrontend

En esta capa se utiliza **Angular**, un framework de desarrollo de aplicaciones web. Angular se basa en TypeScript y permite la creación eficiente de aplicaciones web de una sola página (SPA). Con Angular, se construyen los componentes de frontend que componen la interfaz de usuario del sistema. Este framework proporciona un conjunto de herramientas y patrones de diseño que facilitan la creación de interfaces de usuario dinámicas y altamente interactivas. Con su arquitectura basada en componentes, Angular permite una fácil reutilización y mantenimiento del código, lo que resulta en un desarrollo más rápido y escalable de la capa frontend del sistema. [\[2\]](#)

3.2.2 Capa microservicios

En esta capa se utiliza **Java Spring Boot** el cual es un framework de desarrollo de aplicaciones Java seleccionado para implementar la lógica del servidor en este proyecto. Su capacidad para crear servicios web RESTful de manera eficiente es especialmente valiosa en este contexto, ya que permite la comunicación fluida entre el frontend y el backend de la aplicación. Spring Boot ofrece una estructura sólida y conveniente para desarrollar microservicios que pueden escalar de manera independiente. [\[3\]](#)

Además, para la documentación del código y la descripción de los servicios API RESTful, se utiliza **Swagger**, esta herramienta se usa para diseñar, construir y documentar servicios API RESTful debido a que ofrece una documentación automática y una interfaz interactiva que facilita las pruebas de APIs.

3.2.3 Capa de base de datos

En este proyecto, se utiliza **Oracle DB** como la base de datos principal. Oracle es reconocido como un sistema de gestión de bases de datos confiable y escalable, que ofrece una sólida

capacidad de almacenamiento y recuperación de datos. La elección de Oracle DB garantiza la seguridad y la integridad de la información. [4]

La robustez y la escalabilidad de Oracle DB la hacen adecuada para manejar grandes volúmenes de datos, lo que permite un rendimiento óptimo incluso en entornos de alta demanda. Además, su amplia gama de características y capacidades avanzadas proporciona flexibilidad para adaptarse a las necesidades específicas del proyecto.

3.2.4 Control de versiones y calidad de código

Esta capa es fundamental para agilizar y mejorar el proceso de desarrollo de software en el proyecto. Se integran herramientas como **Git**, **DevOps**, **CI/CD (Integración Continua/Distribución Continua)** y **SonarQube** para facilitar la entrega continua y la implementación automatizada del software.

En primer lugar, Git se utiliza para el control de versiones, permitiendo una colaboración eficiente y un manejo efectivo de cambios en el proyecto. Esta herramienta está respaldada por sólidos conceptos teóricos relacionados con el control de versiones y la gestión de cambios. [5]

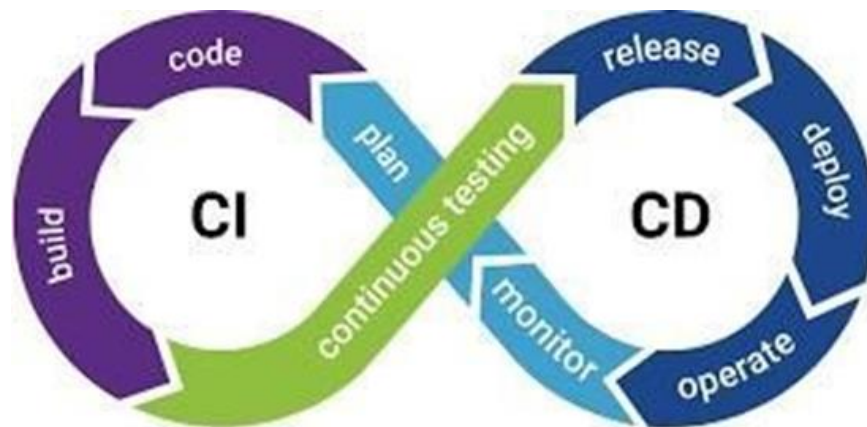
Por otro lado, DevOps es una metodología de trabajo que promueve la colaboración estrecha entre equipos de desarrollo y operaciones. [6] Se basa en la automatización, integración y entrega continuas para acelerar la entrega de software. Adicionalmente, la práctica de CI/CD implica la integración regular y automatizada de cambios de código, seguida de pruebas automáticas y entrega automatizada de software a entornos de desarrollo. [7]

Además, SonarQube, una plataforma de código abierto, se integra estrechamente con Git, DevOps y CI/CD para evaluar y mejorar la calidad del código fuente. Proporciona análisis automáticos del código, identifica problemas de calidad y ofrece retroalimentación inmediata

sobre la calidad del código en cada etapa del ciclo de desarrollo. Su capacidad para automatizar el análisis del código en cada compilación contribuye a un desarrollo más rápido y confiable del software.

Figura 2.

¿Qué es CI/CD? Fuente (Synopsys, 2022).



4. Metodología

4.1 Marco ágil de trabajo (SCRUM)

El marco de desarrollo utilizado para la metodología del proyecto fue SCRUM, el cual es un marco de desarrollo ágil de software ampliamente utilizado; se basa en ciclos de desarrollo cortos llamados "Sprints" y promueve la colaboración cercana entre los miembros del equipo. SCRUM se adapta bien a proyectos donde los requisitos son cambiantes o poco definidos y enfatiza la entrega continua de incrementos de software.[\[8\]](#)

Los actores principales dentro del marco de trabajo son:

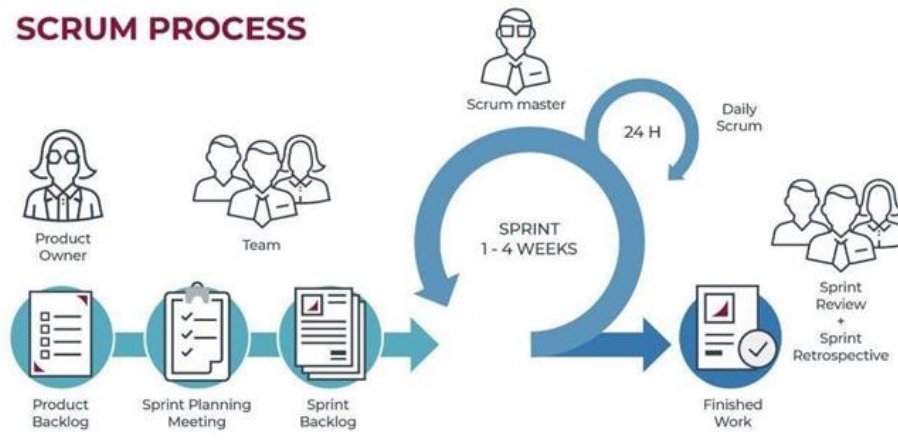
- El Product Owner representa los intereses del cliente y del negocio, define y prioriza los elementos del backlog del producto, y toma decisiones sobre las funcionalidades a incluir en cada entrega.
- El Scrum Máster facilita el equipo SCRUM, eliminando obstáculos que puedan afectar al equipo, organizando reuniones como la planificación de sprint y asegurándose de que se sigan los principios del marco ágil.
- El equipo de desarrollo convierte los elementos del backlog del producto en incrementos potencialmente entregables del producto al final de cada sprint, trabajando de manera autoorganizada para cumplir con los objetivos establecidos.
- Los Stakeholders son personas o grupos con interés en el producto, como clientes, usuarios finales y gerentes. Proporcionan retroalimentación sobre el producto y pueden participar en la revisión de sprint.

Los procesos principales dentro del marco de trabajo son:

- **Backlog del producto:** Se crea un backlog del producto que contiene todas las funcionalidades y tareas necesarias para completar el prototipo. Este backlog se prioriza en colaboración con el cliente y los autores del proyecto.
- **Reuniones de planificación de Sprint:** Al comienzo de cada sprint, se lleva a cabo una reunión de planificación y estimación donde se seleccionan las tareas del backlog del producto para proceder a desarrollarlo durante el sprint correspondiente.
- **Sprints:** El plan de trabajo se divide en sprints de duración fija, generalmente de 2 a 4 semanas, donde se desarrolla un conjunto de funcionalidades específicas del prototipo.
- **Reuniones diarias (Daily meetings):** Se llevan a cabo reuniones diarias cortas en donde se discute el progreso, los obstáculos y los próximos pasos. Estas reuniones permiten mantener el mismo camino y una mejor identificación de cualquier problema que surge.
- **Entregable del Sprint:** Al final de cada sprint, se presenta un incremento potencialmente entregable del producto, que incluye las funcionalidades completadas durante el sprint, el cual está listo para revisión por parte del equipo de Control de Calidad (QA) y el equipo de Experiencia de Usuario (UX).
- **Retrospectivas del Sprint:** Después de cada sprint, se realiza una retrospectiva en donde se analiza lo que funcionó bien, lo que podría mejorarse y se definen acciones para implementar esas mejoras en futuros Sprints.

Figura 3.

¿Qué es Scrum? Fuente (Slide Egg).



La plataforma utilizada para la gestión del proyecto fue Taiga, una herramienta ágil de gestión de proyectos que permite a los equipos planificar, desarrollar y monitorear proyectos de software de manera colaborativa y eficiente. Se enfoca en metodologías ágiles, brindando herramientas que respaldan el desarrollo de software iterativo e incremental. Taiga ofrece tableros Kanban y Scrum que permiten a los equipos visualizar y administrar tareas de forma flexible. Estos tableros facilitan el seguimiento del progreso del proyecto al organizar historias de usuarios, tareas y elementos de trabajo en columnas personalizables. Además, Taiga simplifica la planificación y estimación de historias de usuario mediante puntos de esfuerzo, lo que ayuda a priorizar el trabajo y establecer expectativas realistas sobre el tiempo y los recursos necesarios para completar las tareas.

4.2 Fases del proyecto

4.2.1 Fase 1: Análisis del proyecto y planeación

Esta fase del proyecto se concentra en la evaluación inicial y la planificación detallada del mismo. Durante esta etapa, se lleva a cabo un análisis exhaustivo de los objetivos y alcance, así como la definición de un plan estratégico para guiar todas las actividades subsiguientes.

A continuación, se presenta una lista de tareas en orden a realizar en esta fase:

1. Análisis de viabilidad.
2. Definición de alcance.

4.2.2 Fase 2: definición de requerimientos

En esta fase se centra en la definición precisa de los requerimientos. La fase de definición de requerimientos es fundamental para comprender en detalle qué funcionalidades y características debe tener el sistema para cumplir con la creación de las propuestas de extensión de la VIE.

Tareas para realizar en esta fase:

- Análisis de documentación existente.
- Entrevistas y encuestas.
- Priorización de requerimientos.
- Identificación de requerimientos funcionales y no funcionales.
- Revisión y aprobación por parte del cliente.

Esta fase cumple el primer objetivo específico:

- Establecer los requerimientos para administrar propuestas de extensión no solidarias de la VIE, los cuales abarcan la creación de nuevas propuestas, la selección del servicio a relacionar con las mismas, la trazabilidad del proceso y la identificación de roles y unidades.

4.2.3 Fase 3: Desarrollo y documentación

Esta es una fase crucial en el proyecto en la que se enfoca en la implementación del prototipo de la aplicación web. Durante esta fase, se trabaja en el desarrollo de las diferentes partes del sistema, incluyendo tanto la capa frontend, que se encarga de la interfaz de usuario y la experiencia del usuario, como la capa backend, que se encarga del procesamiento de datos y lógica empresarial.

Tareas en secuencia a realizar en esta fase:

- Configuración de los entornos de desarrollo.
- Rediseño de pantallas por parte del equipo de experiencia de usuario (UX).
- SPRINTS - Se planea un total de 3 sprints para el desarrollo del proyecto, los cuales conllevan a realizar una serie de tareas durante cada sprint, las cuales son:
 - Reestructuración de base de datos.
 - Documentación del código.
 - Desarrollo de la Capa Backend.
 - Desarrollo de la Capa Frontend.
 - Dev Test - Test de funcionalidades.
 - Corrección de Errores y Depuración.

Al finalizar esta fase se tiene por completado el siguiente de los objetivos específicos:

- Implementar un prototipo de aplicación web en sus capas Frontend y Backend que se ajuste a los requerimientos previamente establecidos.

4.2.4 Fase 4: Evaluación de calidad y prototipo

Esta fase se centra en garantizar que el prototipo de la aplicación web cumpla con los estándares de calidad y funcionalidad previamente establecidos. Durante esta fase, se llevarán a cabo pruebas y evaluaciones para identificar posibles errores o deficiencias en el sistema y corregirlos antes de su implementación final.

Tareas a realizar en esta fase:

- Realizar pruebas de QA (pruebas de funcionamiento) y UX (pruebas de estandarización y diseño).
- Realizar la integración con SonarQube para analizar tanto el repositorio backend como el frontend, garantizando la corrección de vulnerabilidades, la mejora en la calidad del código y la adhesión a las mejores prácticas de desarrollo.
- Documentar y seguir de cerca cualquier problema encontrado y trabajar en su resolución.

Al terminar esta fase se habrá completado el último objetivo específico:

- Validar el prototipo a través de un plan de pruebas de calidad.

4.2.5 Fase 5: Finalización y entrega

Esta es la fase final del proyecto se hace la entrega del proyecto en la rama de desarrollo develop, actualmente se manejan cuatro ramas en RSI:

- Predev (Pre-Development): Etapa inicial donde los desarrolladores trabajan en funcionalidades o correcciones antes de la integración principal.
- Develop (Desarrollo): Rama principal de desarrollo donde se integran cambios de todos los desarrolladores.
- PreProd (Preproducción): Etapa intermedia para pruebas finales antes de la implementación en producción.
- Prod (Producción): Rama en la que se encuentra la versión del software en el entorno de producción para uso público.

Se planea el desarrollo del prototipo hasta la rama de Develop (Desarrollo) ya que en esta misma rama es donde se hacen las pruebas de QA que se enfocan en la calidad del software desde una perspectiva técnica y de UX que se centran en la calidad de experiencia del usuario al interactuar con el software.

5. Resultados

Estos resultados describen los logros alcanzados en cada fase del proyecto. Durante este se siguió un enfoque sistemático, desde el análisis inicial hasta la implementación final del modelo.

La adopción de un enfoque colaborativo fue clave para lograr los objetivos planteados y establecer un sólido prototipo para en un futuro poder realizar mejoras.

5.1 Fase 1: Análisis del proyecto y planeación

En esta fase nos enfocamos en la planeación y análisis de las herramientas y metodologías que utilizamos para la investigación y desarrollo del prototipo para el módulo de propuestas de extensión.

Así como en realizar el análisis de la viabilidad del proyecto junto con el impacto que podría tener sobre la forma en la que se registran las propuestas de extensión en comparación con el actual sistema.

5.1.1 Análisis de viabilidad

Este análisis abarca varios aspectos que aseguran la viabilidad del proyecto, ya que es necesario que el proceso de registro de solicitudes de extensión se optimice en el sistema de información de la Vicerrectoría de Investigación y Extensión.

Un aspecto importante a tener en cuenta es el por qué realizar un prototipo del sistema para el registro y gestión de las propuestas de extensión, la razón principal se sustenta sobre las complicaciones y retrasos que en ocasiones genera el actual sistema debido a los formularios poco intuitivos que posee para los usuarios y a la falta de funcionalidades. Lo cual nos empuja al planteamiento de querer mejorar tanto el software, como de llevar a cabo el cambio de la arquitectura en la que se encuentra el actual sistema. El cambio de la arquitectura se plantea dentro de los mismos recursos dispuestos por la Universidad para ello, como lo son el proyecto de **Renovación de Sistemas de Información (RSI)**. La arquitectura de RSI nos permite la implementación de nuevas funcionalidades, la creación de interfaces web para los usuarios y el mantenimiento del proyecto a largo plazo.

Además, la viabilidad e importancia del proyecto se ven fortalecidas por las oportunidades de mejorar la gestión de las propuestas de extensión. En resumen, contamos con todos los elementos adecuados para la realización exitosa de este proyecto, lo que nos da la confianza necesaria para pasar a la siguiente etapa de desarrollo.

5.1.2 Definición de alcance

La Universidad Industrial de Santander maneja dos tipos de propuestas de extensión: las propuestas solidarias y las no solidarias.

Las propuestas no solidarias tienen como características principales que los beneficios o bonificaciones reportados deben pasar por una revisión por parte de un profesional de la VIE (Vicerrectoría de Investigación y Extensión) y de la DTC (Dirección de transferencia de

conocimiento).

Por su parte las propuestas solidarias tienen como característica que no generan beneficios y que son propuestas que tienen un flujo de aprobación adicional por parte del COVIE (Comité de la Vicerrectoría de Investigación y Extensión) lo cual añade estados y roles adicionales, así como aprobaciones necesarias.

Dentro del alcance para el desarrollo del prototipo se planteó que tuviera la estructura básica para la gestión y manejo por roles básicos, también siguiendo los objetivos del proyecto de los cuales se pueden deducir funcionalidades básicas como:

- **Trazabilidad del Proceso:** Permitir a los usuarios consultar por el estado actual de sus propuestas proporcionando transparencia en el proceso.
- **Identificación de Roles y Unidades Involucradas:** Mostrar claramente los roles y unidades involucrados en cada etapa del proceso de selección y revisión de propuestas.
- **Interfaz de Usuario Intuitiva:** Desarrollar una interfaz de usuario intuitiva y fácil de usar que permita a los usuarios navegar sin problemas por las diferentes funcionalidades y realizar acciones como registrar propuestas y permitir la búsqueda del servicio a relacionar con la misma.

Se debe tener en cuenta que al enfocarnos en las propuestas no solidarias el prototipo solo contempla estados y registros de seguimiento sin tener en cuenta el registro o aprobación con el COVIE.

Se planea el desarrollo del prototipo hasta la gestión básica de las propuestas, las cuales contemplan acciones básicas de un CRUD, pero con algunas condiciones y permisos adicionales.

El registro de la información de la propuesta se maneja hasta la parte del formulario

relacionado a **Destinatarios**, donde se ingresa la información de los destinatarios externos o no externos de la propuesta.

No se tendrán en cuenta los roles de “Auxiliar VIE”, “Profesional VIE” y “Jefe DTC” en esta etapa de desarrollo, solo se trabajará con el rol “Docente Planta” y “Jefe de Unidad”.

5.2 Fase 2: definición de requerimientos

Esta fase se centra en la definición precisa de los requerimientos que guiaron el desarrollo del prototipo. Para lograr esto, se llevaron a cabo una serie de actividades destinadas a recopilar información, comprender las necesidades del cliente y establecer los criterios clave para el proyecto.

5.2.1 *Análisis de documentación existente*

Se realiza un análisis de la documentación existente relacionada con el proceso de registro de propuestas de extensión en la Universidad Industrial de Santander (UIS). Esta revisión permite familiarizarse con los sistemas y procedimientos actuales, así como identificar áreas de mejora potencial, también con esto se inicia el rediseño de pantallas por medio de mockups junto con el equipo UX.

Para esta tarea se cuenta con un manual de usuario del sistema de extensión VIE actual [\[9\]](#) y un fomento de las capacidades de extensión [\[10\]](#) en donde se explica la extensión, fundamentada en principios básicos (comunicación, cooperación, solidaridad, equidad, transparencia y pertinencia social y académica), como eje misional de la Universidad y el cómo debe facilitar la relación existente entre la Universidad y la sociedad en general, sustentándose en el alto compromiso ético y en la capacidad y la calidad académica de la comunidad universitaria.

Mediante la extensión universitaria se establece un diálogo permanente con la comunidad, sus actores sociales, el sector productivo y el Estado, promoviendo la transferencia de conocimiento científico y tecnológico desde la Universidad hacia los sectores productivos y sociales y la generación de conocimiento a partir de las necesidades que surjan al interior de las empresas y organizaciones. (Acuerdo No. 006 del 7 de febrero del 2005 del Consejo Superior), por medio del cual se adoptan las políticas y se definen los principios orientadores y los objetivos de la función de Extensión de la Universidad. [11]

También se realiza una exploración detallada del sistema actual [12] utilizado para el registro de propuestas de extensión y durante esta se identificaron varios aspectos clave para la definición de requerimientos, a continuación, se recopilan algunas capturas de pantalla del sistema actual para ilustrar su interfaz y funcionalidades:

- Se observa el formulario de búsqueda del sistema actual de la VIE para la gestión de propuestas, donde se tienen algunos campos claves para búsqueda. (ver Apéndice A para observar en su totalidad las pantallas del sistema actual).

Figura 4.

Pantalla de consulta sistema actual VIE. Fuente (Manual y documentación oficial).

Unidad: CEDEUIS

Unidades Adscritas: [dropdown]

Cod. Proyecto: [input]

Descripción: [input]

Estado: [dropdown]

Limpiar | Buscar

Unidad	Cod. Proyecto	Nombre	Fecha Creación
CEDEUIS	2150-1424	PRODUCCIÓN PEDAGÓGICA DE CURSOS VIRTUALES PARA EL ICBF	2018/06/08
CEDEUIS	2150-3894	PRODUCCIÓN PEDAGÓGICA DE CURSOS VIRTUALES PARA EL ICBF	2023/12/04

Páginas: [1] de [1]

Volver a Menú Principal Inicio | Crear Propuesta de Extensión

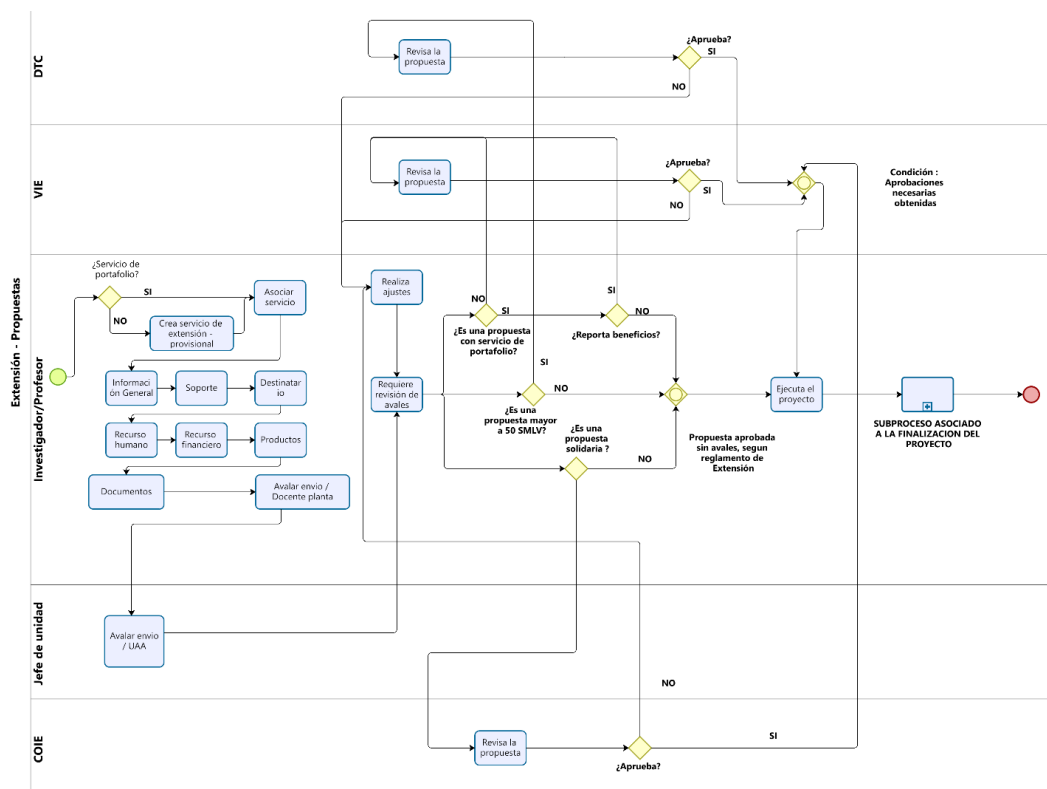
Acceso a Portafolio de Servicios UIS | Perfil | Inicio | Salir

Vicerrectoría de Investigación y Extensión - Universidad Industrial de Santander - Todos los derechos reservados © - Desarrollado por Creando.Soft

Gracias al análisis de la documentación, a la revisión del manual del sistema actual y a la exploración dentro del mismo, se creó un diagrama del modelado de proceso de negocio (por sus siglas en inglés Business Process Model and Notation BPMN) para mapear de manera correcta el flujo de proceso del registro de propuestas de extensión, el cual fue revisado y validado por la Profesional Erika Vallejo (cliente actual).

Figura 5.

Diagrama de BPMN para la gestión de propuestas de extensión. Fuente (Análisis y documentación).



5.2.2 Entrevistas y encuestas

Se realizan varias entrevistas y encuestas principalmente con Erika Tatiana Vallejo Serrano (Profesional - Programa de Extensión Vicerrectoría de Investigación y Extensión) siendo el usuario final o cliente. También algunas con Jimena Gómez Delgado profesional usuario del sistema actual de extensión VIE y Robinson León Diaz, también profesional usuario del sistema actual.

Estas interacciones brindan información valiosa sobre las expectativas, los desafíos y las oportunidades relacionadas con el proceso de registro de propuestas de extensión, permitiendo así una mejor identificación de los requerimientos funcionales y no funcionales, evitando y/o corrigiendo errores presentes en el actual sistema y generando futuras mejoras.

5.2.3 Priorización de requerimientos

Una vez se identifican todos los requerimientos, se priorizan en función de su importancia y su impacto en el proyecto. Esto permite centrar los esfuerzos en las áreas críticas y garantizar que el desarrollo del prototipo satisfaga las necesidades más importantes de los usuarios y el cliente. Para esto, se prioriza uno a uno todos los requerimientos escritos en el punto siguiente.

5.2.4 Identificación de requerimientos funcionales y no funcionales

Con base en la información recopilada, se identifican tanto los requerimientos funcionales como los no funcionales del sistema. [13] Gran parte de los requerimientos funcionales y no funcionales que se muestran a continuación fueron establecidos de la mano del usuario, en las encuestas y reuniones realizadas, se dividieron los requerimientos por secciones de información relacionada que favorecen a la construcción de las historias de usuario.

5.2.4.1 Requerimientos generales de permisos, accesos y gestión de propuestas. Estos requerimientos están enfocados en funciones generales del sistema más que en una sección en particular, buscan en gran parte cubrir aspectos de seguridad, rendimiento y escalabilidad del sistema.

Tabla 1.

Requerimientos funcionales de permisos, accesos y gestión de propuestas.

ID	Descripción	Prioridad
RFG1	Los usuarios con roles de "docente planta" o "jefe de unidad" deben poder consultar propuestas de extensión.	Media
RFG2	El panel de consulta debe tener un botón de creación donde el usuario con rol "Docente planta" pueda acceder a crear una propuesta de extensión.	Media
RFG3	El panel de consulta debe contener los siguientes campos para filtrar las propuestas de extensión: unidad gestora, título del	Alta

	proyecto, fecha de inicio, fecha de fin, código de proyecto y estado.	
RFG4	Se deben tener botones para limpiar la consulta y para realizar la consulta.	Alta
RFG5	Los campos de consulta deben permitir al usuario ingresar valores o seleccionar opciones según corresponda.	Alta
RFG6	Se debe seleccionar la unidad gestora (requerida) para realizar una consulta. En caso contrario, se debe mostrar una advertencia.	Alta
RFG7	La consulta debe generar un snackbar indicando si se realizó correctamente o si no se encontraron datos con los criterios ingresados.	Baja
RFG8	La tabla de resultados debe tener funciones de filtrado y debe mostrar información relevante como el código de proyecto, nombre de la unidad presupuestal, etc.	Media
RFG9	Se deben validar todas las funcionalidades de los campos de consulta, asegurando que no existan errores en consola y que se muestren correctamente los snackbars.	Baja
RFG10	Se debe verificar que los datos consultados sean correctos y que las traducciones estén correctamente implementadas.	Media
RFG11	Un usuario con rol de "docente planta" debe poder crear una	Alta

	propuesta y realizar las acciones de la tabla de consulta tales como editar y eliminar propuestas de extensión.	
RFG12	En el estado "borrador" los usuarios con rol de "docente planta" pueden realizar las acciones de la tabla de resultados de consulta tales como editar y eliminar.	Alta
RFG13	En el estado "en trámite" solo se permite la acción de visualización de las propuestas.	Alta
RFG14	En el estado "en corrección" los usuarios con rol de "docente planta" pueden realizar la acción de editar la propuesta, pero no eliminarla.	Alta
RFG15	Se debe seleccionar y relacionar un servicio con la propuesta de extensión que se esté creando.	Alta
RFG16	Solamente el usuario cuyo rol asignado sea "docente planta" puede iniciar la creación de una nueva propuesta de extensión.	Alta
RFG17	El sistema debe tener una portada con una imagen y frase relativa a lo que se podrá hacer dentro del mismo.	Baja

Tabla 2.*Requerimientos no funcionales de permisos, accesos y gestión de propuestas.*

ID	Descripción	Prioridad
RNFG1	La interfaz debe ser intuitiva y fácil de usar para los usuarios con roles que puedan acceder al mismo.	Media
RNFG2	El sistema debe ser eficiente en la consulta y presentación de los datos, garantizando tiempos de respuesta rápidos.	Alta
RNFG3	Todos los labels y textos que no provienen de la base de datos deben estar correctamente traducidos y adaptados al idioma correspondiente (español o inglés), asegurando una experiencia de usuario coherente y amigable.	Media
RNFG4	El sistema debe proporcionar mensajes claros y precisos a los usuarios mediante snackbar para informar sobre el éxito o el fallo de las acciones realizadas.	Media
RNFG5	El proyecto debe de tener una documentación Swagger de todos los endpoints desarrollados e implementados durante el desarrollo.	Media
RNFG6	El proyecto debe de tener una documentación SQL de la migración de tablas y datos, en donde también exista una guía para cargar toda la base de datos.	Media
RNFG7	El código debe cumplir con estándares de calidad	Media

predefinidos, incluyendo buenas prácticas de codificación, legibilidad y mantenibilidad. Se deben realizar análisis automáticos para identificar posibles problemas y mantener un alto nivel de calidad en el código fuente.

5.2.4.2 Requerimientos sobre el manejo de la información básica de una propuesta de extensión. Esta sección corresponde a la información básica necesaria para una propuesta de extensión, incluye información crucial como la unidad gestora, el servicio que se va a prestar durante la ejecución y el ordenador de gasto, entre otros. (ver Apéndice B para observar las demás tablas de requerimientos).

5.2.4.3 Requerimientos sobre el manejo de las características del proyecto. Esta sección corresponde a los aspectos generales de la propuesta de extensión, en donde se muestra e ingresa información relevante, tal como el código de la propuesta el cual es único e irrepetible para cada una. También se solicita información del lugar de ejecución de la propuesta, así como el tipo de iniciativa que tiene, entre otros aspectos generales. (ver Apéndice B para observar las demás tablas de requerimientos).

5.2.4.4 Requerimientos sobre la información de soporte de una propuesta de extensión. Esta sección corresponde a la capacidad institucional en recursos físicos (equipo, software e infraestructura), recursos académicos, talento humano y en logística, disponibilidad para comprometerla en los alcances y tiempos establecidos en la propuesta. (ver Apéndice B para

observar las demás tablas de requerimientos).

5.2.4.5 Requerimientos sobre el destinatario de una propuesta de extensión. Esta sección corresponde al destinatario de la propuesta de extensión, se señala si es externo o no, se incluye información de contacto de la entidad tal como: responsable técnico, un responsable administrativo e información de contacto. (ver Apéndice B para observar las demás tablas de requerimientos).

5.2.5 *Revisión y aprobación por parte del cliente*

Se someten los requerimientos a una revisión y aprobación por parte del cliente para asegurar que estén alineados con las expectativas, objetivos y que cumplan con algunos lineamientos o normatividad impuesta por parte de la Universidad para las propuestas de extensión. Esta validación garantiza y sienta las bases para la siguiente fase del proyecto: el desarrollo y la documentación del prototipo.

5.3 Fase 3: Desarrollo y documentación

Durante la tercera etapa del proyecto se realiza la implementación del prototipo de la aplicación web, enfocándose en el desarrollo de diferentes partes del sistema, tanto en la capa Frontend como Backend. Esta fase es crucial para transformar los requerimientos previamente establecidos en un sistema viable y bien documentado que cumpla con los objetivos del proyecto.

5.3.1 *Configuración de los entornos de desarrollo*

La configuración de los entornos es parte crucial del desarrollo, pues aquí se configuran y despliegan los repositorios necesarios para la implementación del prototipo, tanto en la capa

Frontend como Backend.

5.3.1.1 Configuración de entorno Backend. El sistema actual de la VIE para la gestión de propuestas también está relacionado a otra parte importante ya mencionada anteriormente, la cual es la gestión de servicios de portafolio y provisionales, si bien el desarrollo del prototipo se centró en las propuestas de extensión, el prototipo requiere la búsqueda y selección de servicios ya sean de portafolio o provisionales como se observa en los requerimientos planteados. Por lo cual surge la necesidad de configurar a su vez el entorno de desarrollo para la gestión de servicios, lo que resulta en un backend independiente.

En consecuencia, de lo planteado anteriormente se procede a configurar los siguientes proyectos: **Proyecto backend para gestión de propuestas de extensión** y **Proyecto backend para gestión de servicios de extensión.**

Ambos proyectos comparten características técnicas como:

- Lenguaje de programación Java con versión 11.0.1, la cual es la versión recomendada por RSI.
- Uso de swagger versión 2.9.2 para la documentación de los endpoints.
- Uso de framework como Spring Boot version 2.3.3.
- Uso de la librería oficial de la Uis que ofrece funciones como traducciones desde backend manejo de errores entre otras.

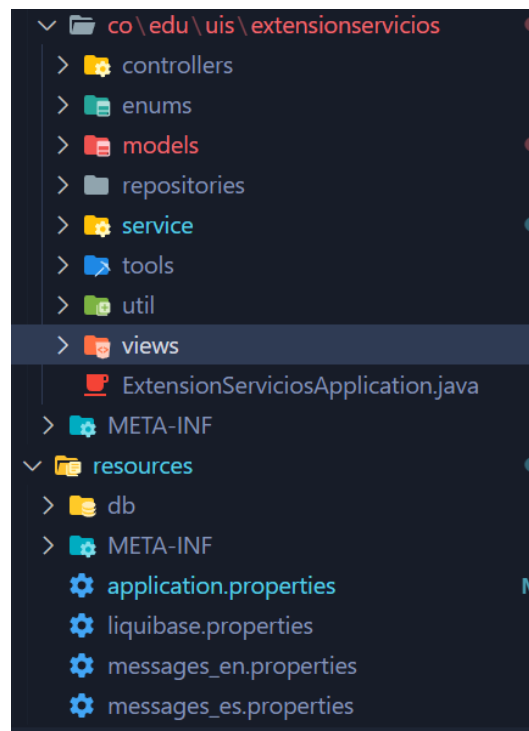
Se hace la configuración de cada proyecto en el entorno local de forma que se despliega y prueba en este entorno, luego de comprobar que el proyecto funciona de manera correcta se procede a su despliegue en el entorno de **Predev**. Este entorno es el entorno óptimo para desarrollo

pues es el usado por el frontend para la integración con el backend y pruebas de desarrollo.

Para ambos proyectos se manejó la misma estructura de carpetas y organización del proyecto como se observa a continuación:

Figura 6.

Estructura de repositorios backend. Fuente (Desarrollo y documentación).



5.3.1.2 Configuración de entorno frontend. Dentro de la configuración del entorno frontend se incluyen varias actividades y herramientas que son esenciales para el desarrollo efectivo de la parte visual y de interacción de la aplicación web.

En base a el proyecto RSI durante la fase de desarrollo del proyecto, se configura un entorno que utiliza una combinación de herramientas y tecnologías líderes en la industria, dentro

de estas se encuentran:

- **Visual Studio Code** como principal editor de código, aprovechando su interfaz intuitiva, amplia gama de extensiones y potentes capacidades de edición.
- **GitLab** como plataforma de alojamiento de repositorios para administrar el control de versiones del código fuente. Todo el desarrollo se encuentra en un repositorio denominado "Front Extensión".
- **Node.js y Yarn** como gestores de paquetes siendo fundamentales para instalar y gestionar las dependencias del proyecto frontend, asegurando un flujo de trabajo eficiente y la consistencia en el entorno de desarrollo.
- **Angular** como Framework frontend principal, aprovechando su robustez, escalabilidad y conjunto completo de características para el desarrollo de aplicaciones web de alta calidad.
- **Linting con ESLint** para mantener un código limpio y consistente.

5.3.2 Rediseño de pantallas por parte del equipo de experiencia de usuario (UX).

El proceso de rediseño de la pantalla fue una parte integral del desarrollo del proyecto. A partir de las herramientas mencionadas anteriormente y las establecidas dentro del marco del proyecto RSI (Renovación de Sistemas de Información) junto con el equipo de experiencia de usuario (UX), se han desarrollado una serie de pantallas que abordan las deficiencias del sistema actual y mejoran la experiencia del usuario durante el proceso de registro de propuestas.

Las pantallas reflejan una interfaz de usuario intuitiva, unificada y fácil de usar. Están

5.3.3 *Desarrollo SCRUM*

Se cumplió una totalidad de 3 Sprints para el desarrollo del proyecto, los cuales conllevan a una serie de tareas realizadas durante cada sprint, las cuales son:

- Reestructuración de base de datos.
- Desarrollo de la Capa Backend.
- Desarrollo de la Capa Frontend.
- Dev Test - Test de funcionalidades.
- Corrección de Errores y Depuración.
- **Documentación del código**

Durante cada sprint del desarrollo, se documentan los endpoints utilizando Swagger en donde se describe cada endpoint incluyendo su propósito, los parámetros requeridos y los métodos HTTP admitidos. Tras implementar los endpoints y documentarlos con Swagger, se despliega el proyecto, lo que genera una URL donde se aloja la documentación de la API, una para Backend Propuestas [\[15\]](#) y otra para Backend Servicios [\[16\]](#).

- **Acerca de la base de datos**

El sistema actual VIE extensión usa PostgreSQL como sistema de administración de bases de datos relacionales de objetos, se hace una exportación de la información montada en los servidores para migrar las tablas y datos al servidor de base de datos ofrecido por RSI, el cual tiene como sistema gestor Oracle Data base.

Tabla 3.*Backlog del Sprint 1.*

Sprint 1	
Historia de usuario	Puntaje
HU #675: Crear propuestas de extensión no solidarias: Step 1 Información Básica	13
HU #782: Crear propuestas de extensión no solidarias: CRUD y consulta Propuestas	8
HU #849: Documentación SWAGGER de la HU #782	2
HU #842: Rebajar deuda técnica proyectos de extensión	3
HU #818: Creación y estabilización del repositorio Back Extensión	2
HU #814: Creación y estabilización del repositorio Back Extensión	2
HU #798: Crear portada para el módulo de extensión	2
Puntaje total sprint	32

Tabla 4.*Backlog del Sprin 2.*

Sprint 2	
Historia de usuario	Puntaje
HU #807: Propuestas de extensión no solidarias: Step 2 Características	8
HU #862: Documentación SWAGGER de la HU #807	2
HU #861: Documentación de SQL de la HU #807	2
HU #869: Acceso al módulo por roles y estados	3
HU #854: Documentación de SQL de la HU #675	2
HU #848: Documentación de SQL de la HU #782	2
HU #875: Documentación SWAGGER en Back Servicios de la HU #782	2
HU #822: Documentación SWAGGER de la HU #675	2
Puntaje total sprint	23

Tabla 5.*Backlog del Sprint 3.*

Sprint 3	
Historia de usuario	Puntaje
HU #881: Propuestas de extensión no solidarias: Step 3 Soporte	8
HU #883: Propuestas de extensión no solidarias: Step 4 Destinatario	8
Puntaje total sprint	16

Sprint 1**Figura 8.**

Backlog del sprint 1. Fuente (Taiga). [\[17\]](#)



Duración del Sprint: Del 14 de noviembre al 06 de diciembre.

- **Reestructuración de base de datos.**

Se comienza por realizar el desarrollo de las tablas relacionadas a este sprint, las cuales se relacionan directamente a funcionalidades e información que se utiliza por cada pantalla.

Se desarrolla la creación de las siguientes tablas para las historias de usuario de este sprint:

1. **Vie_extension_sectores_aplicacion:** Sectores de aplicación para servicios de extensión.
2. **Vie_extension_modalidades:** Modalidades de extensión.
3. **Vie_extension_submodalidades:** Submodalidades de extensión.
4. **Vie_extension_disciplinas:** Disciplinas de extensión.
5. **Vie_extension_lineas_accion:** Líneas de acción de extensión.
6. **Vie_extension_servicios_info_basica:** Información básica de los servicios de extensión.
7. **Vie_extension_serv_sector_rels:** Relación entre servicios de extensión y sectores de aplicación.
8. **Vie_extension_marcos_legales:** Marcos legales de extensión.
9. **Vie_extension_propuestas_info_basica:** Información básica de las propuestas de extensión.

Además, se realiza la migración de la data relacionada a los campos de selección (Marcos legales, modalidades, submodalidades, disciplinas y líneas de acción) y también, la data relacionada a estados de las propuestas. Se crean las relaciones para la tabla “personas” y “unidades presupuestales”, con las cuales podemos anidar al director y jefe de unidad de la propuesta, así como a la unidad gestora.

- **Desarrollo de la capa Backend.**

Desde la capa Backend durante este sprint se desarrollaron las siguientes historias de usuario.

→ HU #842: Rebajar deuda técnica proyectos de extensión

Esta historia consiste en la depuración y corrección de errores de código relacionados con principios de buenas prácticas usando la herramienta de SonarQube, la cual nos ofrece una métrica de cuántos Code Smells (código basura) hay en el proyecto. La historia se hizo con el fin de que al iniciar el desarrollo se inicie con estas métricas en cero y que al finalizar también suceda lo mismo, como se observará más adelante en la fase 5 (Evaluación de calidad y control del prototipo).

→ HU #818: Creación y estabilización del repositorio backend

La creación del repositorio se hace en Gitlab donde se gestionan los diferentes repositorios de todos los proyectos de RSI, primero se hace despliegue en local probando su correcto funcionamiento y luego se procede a subir a Gitlab.

En el caso del backend se crean dos repositorios:

Figura 9.

Repositorio Back Servicios. Fuente (Gitlab UIS). [18]

Back Servicios
 Project ID: 253

66 Commits 4 Branches 0 Tags 146.1 MIB Project Storage

Swagger API Documentation

Merge branch 'BACKEND-SERVICIOS---FORMULARIO-DE-PROPUESTAS' into 'predev'
 Andrea Juliana Urrego Paredes authored 2 weeks ago

predev back-servicios / +

History Find file Edit ↓ ↓ Clone ↓

README CI/CD configuration Add Kubernetes cluster Configure Integrations

Name	Last commit	Last update
.mvn/wrapper	[ADD]: Deploy commit	6 months ago
src	Merge branch 'BACKEND-SERVICIOS---FORMULARIO-...	2 weeks ago
.gitignore	[ADD]: Deploy commit	6 months ago
.gitlab-ci.yml	[ADD]: Deploy commit	6 months ago
README.md	Update README.md	1 month ago
ci_settings.xml	[ADD]: Deploy commit	6 months ago
mvnw	[ADD]: Deploy commit	6 months ago
mvnw.cmd	[ADD]: Deploy commit	6 months ago
pom.xml	Eliminar dependencia de springdoc-openapi-ui en el ar...	1 month ago

Figura 10.

Repositorio Back Propuestas. Fuente (Gitlab UIS). [19]

Back Propuestas
 Project ID: 232

238 Commits 4 Branches 0 Tags 148.9 MIB Project Storage

Swagger API Documentation

Merge branch 'CRUD_PROPUESTAS_NO_SOLIDARIAS' into 'predev'
 Brayan Sneider Daza Suarez authored 3 days ago

predev back-propuestas / +

History Find file Edit ↓ ↓ Clone ↓

README CI/CD configuration Add Kubernetes cluster Configure Integrations

Name	Last commit	Last update
.mvn/wrapper	Initial Commit	7 months ago
src	[FIX] : Code smell	3 days ago
.gitignore	Initial Commit	7 months ago
.gitlab-ci.yml	Initial commit - yml	7 months ago
README.md	Update README.md	7 months ago
ci_settings.xml	Initial Commit	7 months ago
mvnw	Initial Commit	7 months ago
mvnw.cmd	Initial Commit	7 months ago
pom.xml	Merge branch 'EQ5-Crear-propuestas-de-extensión-no...	4 months ago

→ HU #782: Crear propuestas de extensión no solidarias: CRUD y consulta Propuestas

Para la historia de panel de consulta se desarrollaron los siguientes endpoints para las funcionalidades del CRUD y de la consulta:

- Endpoint para listar los estados que se pueden consultar de acuerdo con el rol.
- Endpoint para realizar la consulta con los criterios ingresados.
- Endpoint para eliminar una propuesta (Se oculta de los registros).
- Endpoint para listar la información de una propuesta de acuerdo con su identificador (ID).

5.4 Fase 4: Evaluación de calidad y control del prototipo

- Realizar pruebas de QA (pruebas de funcionamiento) y UX (pruebas de estandarización y diseño)

Las pruebas de QA, también conocidas como pruebas de aseguramiento de la calidad, se centran en garantizar que el software funcione según lo previsto y cumpla con los requisitos establecidos. El proceso de pruebas de QA implica la creación de casos de prueba, la ejecución de estos casos de prueba, el seguimiento y la documentación de los resultados, así como la comunicación de los problemas encontrados al equipo de desarrollo para su corrección.

Las pruebas de UX se centran en evaluar la experiencia general del usuario al interactuar con un producto, como un sitio web, una aplicación móvil o un software. Además, es importante destacar que existen estándares establecidos por el equipo de Experiencia de Usuario (UX) dentro del marco metodológico de RSI, los cuales sirven como base para la realización de estas pruebas.

Estos estándares proporcionan pautas y criterios específicos que se utilizan para realizar las pruebas y durante las mismas se asegura que el diseño cumpla con estos estándares para garantizar una experiencia óptima para el usuario final. [\[20\]](#)

Durante esta fase del proyecto, se llevaron a cabo pruebas de QA y UX para evaluar la calidad y el diseño del prototipo. Los resultados de estas pruebas se pueden encontrar en la carpeta compartida. [\[21\]](#)

- Realizar la integración con SonarQube para analizar tanto el repositorio backend como el frontend, garantizando la corrección de vulnerabilidades, la mejora en la calidad del código y la adhesión a las mejores prácticas de desarrollo.

Los repositorios que se crearon y desplegaron en GitLab contaban con la ventaja de tener pipelines ya configurados para la integración con SonarQube, de modo que cada vez que se suben cambios al repositorio, estos pipelines se ejecutan realizando un despliegue y revisión de calidad del código en la herramienta.

Lo anterior nos permite tener métricas sobre la cantidad de bugs o code smells dentro del código más reciente y el código general. No solo los detecta, sino que también proporciona indicaciones sobre cómo solucionarlos, para lo cual se cumple con tener las mejores métricas posibles al final del desarrollo del prototipo.

Cada uno de los proyectos, tanto los backends como el frontend, cuentan con esta integración, como se muestra a continuación:

Figura 11.

Informe del código actual en SonarQube para el repositorio Back Propuestas. Fuente (SonarQube - SIA UIS).

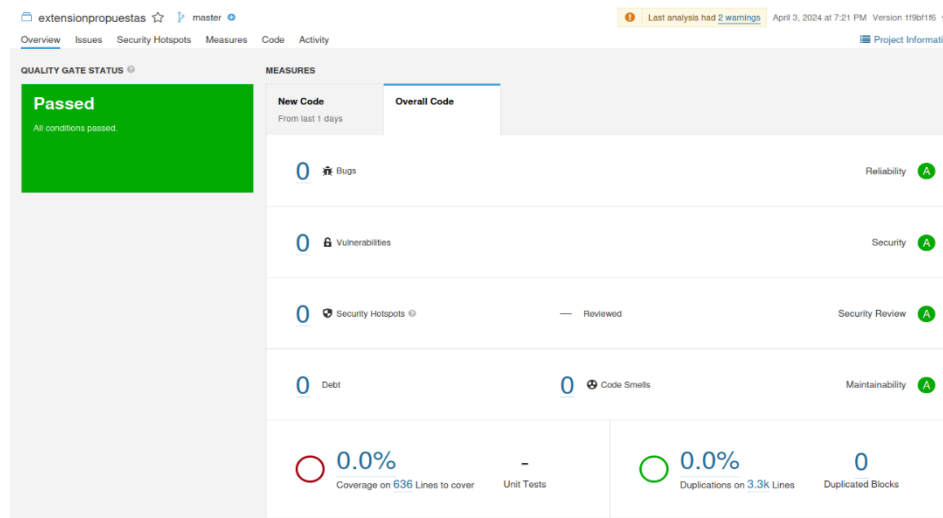
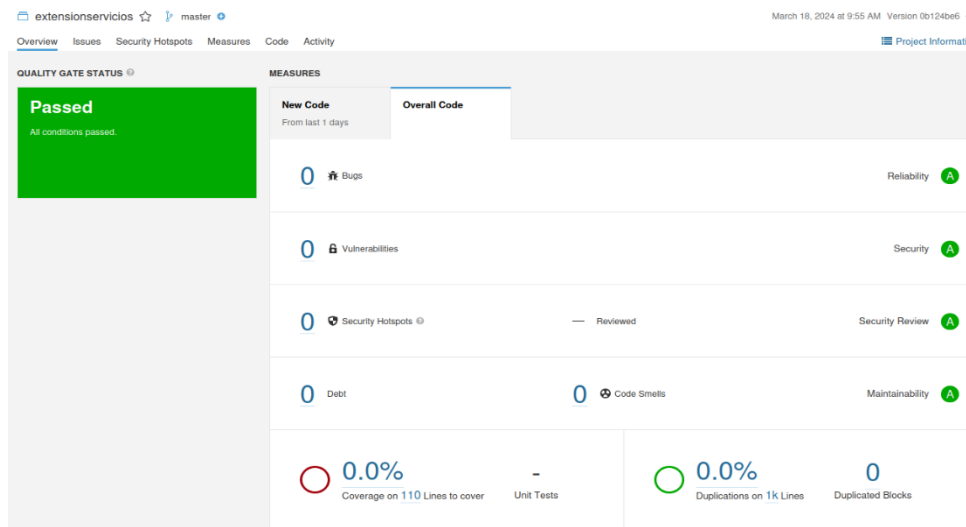


Figura 12.

Informe del código actual en SonarQube para el repositorio Back Servicios. Fuente (SonarQube - SIA UIS).

Fuente (SonarQube - SIA UIS).



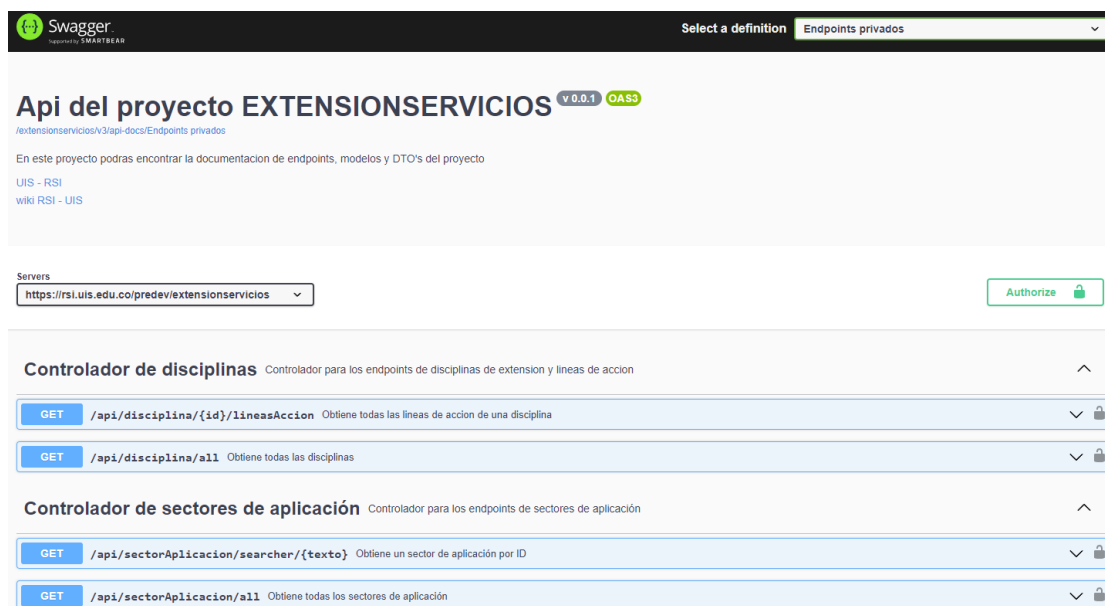
- Documentar y seguir de cerca cualquier problema encontrado y trabajar en su resolución.

Durante el desarrollo de cada sprint se realizaron correcciones y resolución de los problemas encontrados (issues), en algunos casos reportando hallazgos por parte del Dev-Test, QA-Test o UX-Test. Cada problema fue documentado y solucionado.

Parte de la documentación del código se realiza con la herramienta Swagger. Esta permite documentar los endpoints desarrollados durante el sprint y ayuda a mantener un mejor desarrollo a futuro, evitando retrocesos y haciendo más fácil la integración con los componentes del frontend. La herramienta configurada con algunos de los endpoints es la siguiente:

Figura 11.

Documentación Swagger para Back Servicios. Fuente (Swagger - SIA UIS).

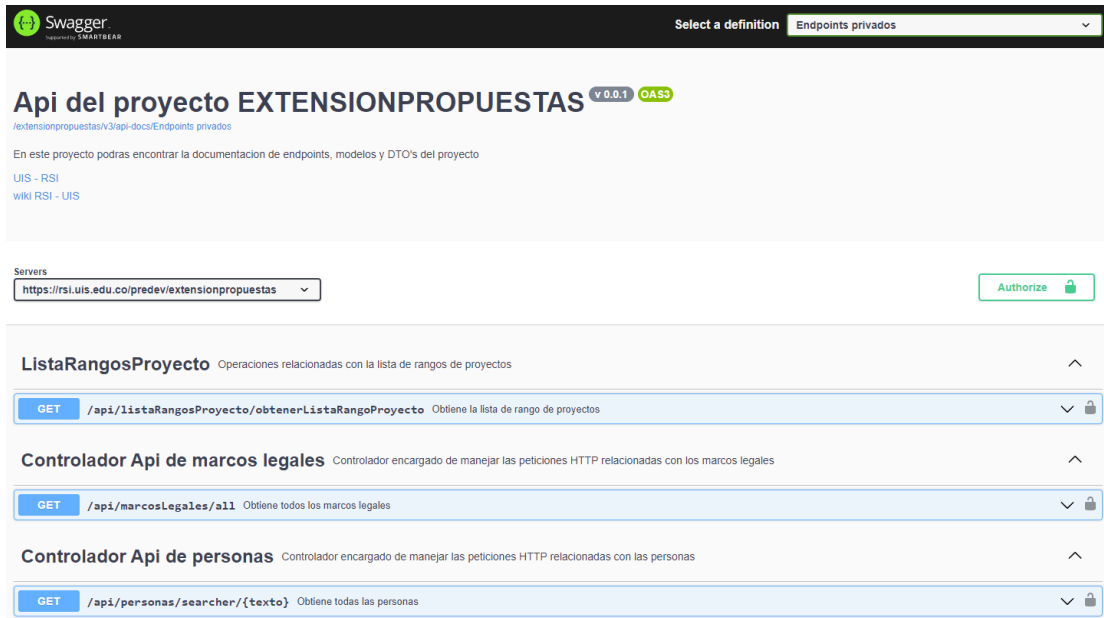


The image shows a screenshot of the Swagger UI for the 'EXTENSIONSERVICIOS' API. The interface includes a header with the Swagger logo and a dropdown menu for 'Select a definition' set to 'Endpoints privados'. The main title is 'Api del proyecto EXTENSIONSERVICIOS' with version 'v 0.0.1' and 'OAS3' tags. Below the title, there is a description: 'En este proyecto podras encontrar la documentacion de endpoints, modelos y DTO's del proyecto' and links for 'UIS - RSI' and 'wiki RSI - UIS'. A 'Servers' section shows the URL 'https://rsi.uis.edu.co/predev/extensionservicios' and an 'Authorize' button. The API endpoints are listed under two controllers: 'Controlador de disciplinas' and 'Controlador de sectores de aplicación'. Each endpoint is shown with its HTTP method (GET), path, and description.

Controller	Method	Path	Description
Controlador de disciplinas	GET	/api/disciplina/{id}/lineasAccion	Obtiene todas las lineas de accion de una disciplina
	GET	/api/disciplina/all	Obtiene todas las disciplinas
Controlador de sectores de aplicación	GET	/api/sectorAplicacion/searcher/{texto}	Obtiene un sector de aplicación por ID
	GET	/api/sectorAplicacion/all	Obtiene todas los sectores de aplicación

Figura 12.

Documentación Swagger para Back Propuestas. Fuente (Swagger - SIA UIS).



5.5 Fase 5: Finalización y entrega

La fase final del proyecto se centra en la finalización y entrega del prototipo desarrollado. El proyecto ha sido desarrollado hasta la rama de Develop. En esta rama se integran los cambios realizados y se llevan a cabo las pruebas de calidad y experiencia del usuario (QA y UX, respectivamente).

- **Estado del proyecto en la rama develop**

En la rama de Develop, el prototipo ha pasado por un proceso de desarrollo y pruebas. Todas las funcionalidades han sido implementadas según los requerimientos establecidos y se han realizado correcciones y ajustes necesarios basados en los hallazgos de las pruebas de calidad y experiencia del usuario.

- **Acceso al prototipo**

Para acceder al prototipo y realizar pruebas adicionales o revisar su funcionalidad, se puede acceder a través de la siguiente ruta dentro del SIA (Sistema de Información Académico) con los roles respectivos asignados: “Profesor planta” y “Jefe de unidad o proyecto”:

VIE → EXTENSIÓN → PROPUESTAS

Figura 13.

Menús. Fuente (Sistemas de información - SIA UIS).

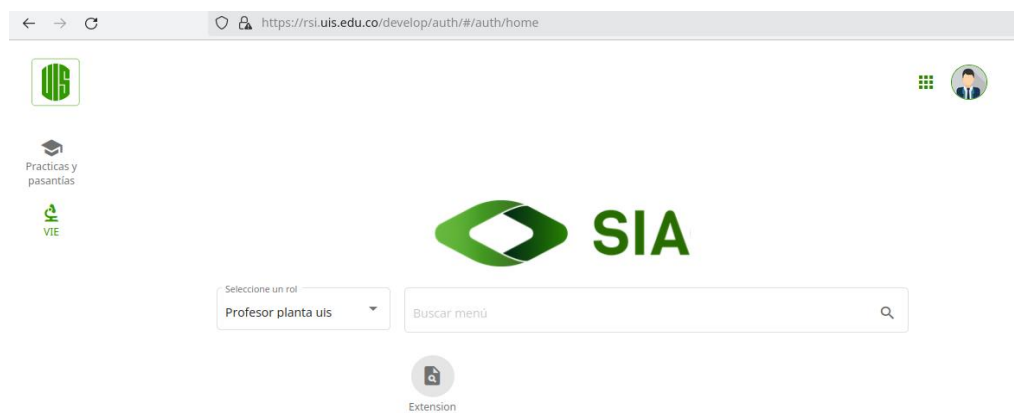
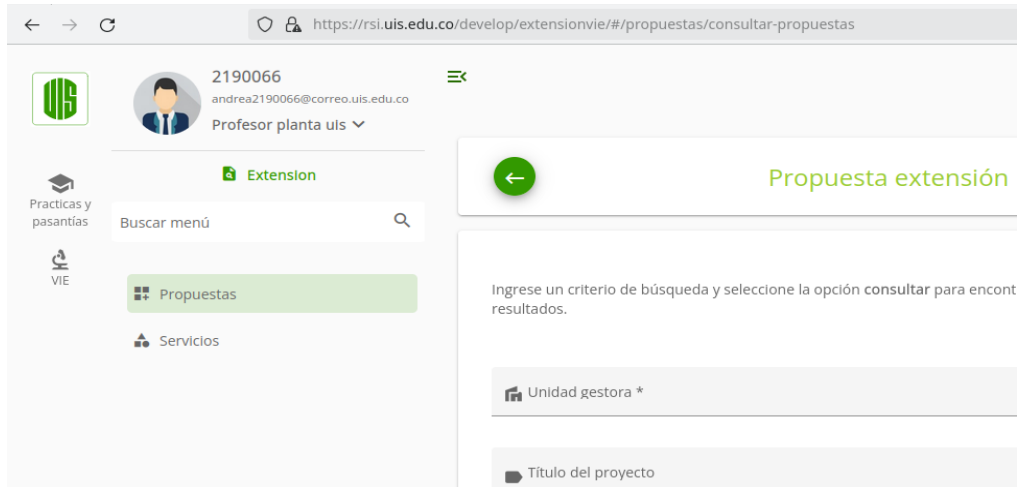


Figura 14.

Menús. Fuente (Sistemas de información - SIA UIS).



6. Conclusiones

El objetivo general ha sido cumplido de manera satisfactoria ya que se logró implementar un prototipo que abarca todas las funcionalidades requeridas para la gestión eficiente de las propuestas.

Se establecieron y cumplieron los requerimientos para administrar propuestas de extensión no solidarias de la VIE, mediante un proceso de levantamiento y análisis de documentación, lo que permitió definir claramente las funcionalidades necesarias para el sistema.

Se logró desarrollar e implementar un prototipo funcional que cubre tanto la capa frontend como backend del sistema, el cual se ajustó a los requerimientos previamente establecidos y proporcionó una interfaz intuitiva para la gestión de propuestas.

También se validó el prototipo a través de un plan de pruebas de calidad que incluyó pruebas de QA, pruebas de UX y revisión de código por medio de SonarQube. Estas pruebas permitieron identificar y corregir errores, garantizando así la calidad del prototipo.

Por otro lado, dando respuesta a la pregunta investigativa planteada, el prototipo desarrollado proporciona una solución que aborda las deficiencias del sistema actual y promueve una gestión de las propuestas de extensión. Además, la colaboración entre los diferentes actores involucrados se facilita mediante la asignación de roles y permisos específicos para cada usuario. En conjunto, estas mejoras contribuyen a la optimización de los tiempos de acción en las diferentes tareas del proceso de gestión de propuestas.

7. Trabajo futuro

A pesar de que el prototipo actual ha alcanzado un nivel satisfactorio de funcionalidad y calidad, hay módulos adicionales que podrían implementarse aprovechando el avance de RSI en la mejora de los sistemas de la universidad.

Algunas de las posibles mejoras y nuevas implementaciones para el prototipo podrían ser:

- Desarrollo del módulo de gestión de servicios (catálogo de extensión), aprovechando que la estructura base del repositorio está configurada.
- Implementación de funcionalidades adicionales como bandeja de entrada y reportes.
- Implementación de validaciones para roles adicionales que requieran usar el sistema a futuro.

Bibliografía

- [1]. KeepCoding. Descubre qué son los microservicios. <https://keepcoding.io/blog/que-son-losmicroservicios/>
- [2]. Hiberus Blog. ¿Qué es Angular y para qué sirve?
<https://www.hiberus.com/crecemoscontigo/que-es-angular-y-para-que-sirve>
- [3]. Arquitectura Java. ¿Qué es Spring Boot? <https://www.arquitecturajava.com/que-es-springboot/>
- [4]. Oracle. Bases de datos. <https://www.oracle.com/co/database/what-is-database/>
- [5]. Microsoft Learn. ¿Qué es git? <https://learn.microsoft.com/es-es/devops/develop/git/whatis-git>
- [6]. NetApp. ¿Qué es DevOps? <https://www.netapp.com/es/devops-solutions/what-is-devops/>
- [7]. Unity. ¿Qué es CI/CD? <https://unity.com/es/solutions/what-is-ci-cd>
- [8]. IebsSchool. Cómo funciona la Metodología Scrum: Qué es y cómo utilizarla.
<https://www.iebschool.com/blog/metodologia-scrum-agile-scrum/>
- [9]. Universidad Industrial de Santander. *Manual de usuario sistema de extensión vicerrectoría de investigación y extensión (13)*. Recuperado de https://uis.edu.co/wp-content/uploads/2022/09/Manual-de-Usuario_-Propuestas-de-Extension.-pdf.pdf

- [10]. Universidad Industrial de Santander. *Fomento de las capacidades de extensión (4)*. Recuperado de <https://uis.edu.co/wp-content/uploads/2022/07/5.3.FOMENTO-DE-LAS-CAPACIDADES-DE-EXTENSION.pdf>
- [11]. Universidad Industrial de Santander. (7 de febrero). *Acuerdo No. 006 DE 2005 (7)*. Recuperado de <https://uis.edu.co/wp-content/uploads/2022/07/Políticas-de-Extension.pdf>
- [12]. *VIE: Creandosoft.com*. (s. f.). <http://extensionvie.uis.edu.co:8081/vie/>
- [13]. Northware. (2022, 29 julio). *Requerimientos en el desarrollo de software y aplicaciones*. Northware. <https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/>
- [14]. *Wireframe sive*. (S. f.). Figma. <https://www.figma.com/file/vYZeWDgqj1RsMZqwj1czB1/wireframe-sive?type=design&node-id=449-4115&mode=design&t=wH2gq5ZxBiKsiLKb-0>
- [15]. *Swagger UI*. (n.d.). <https://rsi.uis.edu.co/predev/extensionpropuestas/swagger-ui/index.html>
- [16]. *Swagger UI*. (n.d.). <https://rsi.uis.edu.co/predev/extensionsservicios/swagger-ui/index.html#/>

[17]. *Backlog - SI-VIE*. (n.d.). Taiga - Love Your Projects.

<https://chaquen.uis.edu.co/project/dgomezs-si-vie/backlog>

[18]. RSI / VIE / Extension / Back Servicios · GitLab. (n.d.). GitLab.

<https://gitlab.uis.edu.co/rsi/vie/extension/back-servicios>

[19]. RSI / VIE / Extension / Back Propuestas · GitLab. (n.d.). GitLab.

<https://gitlab.uis.edu.co/rsi/vie/extension/back-propuestas>

[20]. *25 Corner Design System / Estándares*. (s. f.). Figma.

<https://www.figma.com/proto/OxHmlXHiVzakU0pWJ4boZT/25-Corner-Design-System-%2F-Est%C3%A1ndares?type=design&node-id=4331-776&t=SXG4alqfIDnd4EHN-1&scaling=contain&page-id=4322%3A2977&starting-point-node-id=4331%3A776>

[21]. *Pruebas de QA y UX - Google Drive*. (n.d.).

https://drive.google.com/drive/folders/1uItfrO6C5BwQ_7I3fCmMP1X2aF6PYgUh

[22]. *Sign in* · GitLab. (s. f.-b). <https://gitlab.uis.edu.co/rsi/vie/extension/front-extension>

[23]. *Inicio | Extension*. (n.d.). <https://rsi.uis.edu.co/develop/extensionvie/#/home>

[24]. *Inicio | Extension*. (n.d.).

<https://rsi.uis.edu.co/develop/extensionvie/#/propuestas/consultar-propuestas>

Apéndice A. Pantallas del sistema actual vie extensión

- Se observa la primera interfaz con la que se encuentra el usuario al iniciar el flujo de creación de una propuesta de extensión, donde se anida un servicio provisional a la propuesta.

Figura A1.

Pantalla para la selección del servicio. Fuente (Manual y documentación oficial).

- Se observa la interfaz con la que inicia el formulario para crear una propuesta, siendo información básica la primera sección luego de la pantalla para la selección del servicio.

Figura A2.

Pantalla para el registro de la información básica de una propuesta. Fuente (Manual y documentación oficial).

- Luego de la pantalla anterior se observa la interfaz en la cual se ingresa la información general de las propuestas, relacionada al lugar de ejecución y tipo de iniciativa.

Figura A1.

Pantalla información general. Fuente (Manual y documentación oficial).

General Soporte Destinatario R Humano R Financiero Productos Documentos

Por favor diligencie en su totalidad la información solicitada. Esta corresponde a la información básica de la propuesta de extensión a presentar.

Tipo Iniciativa Otra, ¿Cuál?

Fecha Límite

Población atendida

Número Estudiantes

Lugar de ejecución

Campus o Sede Principal UIS

Una de las Sedes, ¿Cuál?

Por fuera de la Universidad

¿Cuál es el componente misional y la coherencia de la actividad con el Proyecto Institucional?

¿Por qué es pertinente este proyecto de extensión en el medio?

¿Ha concluido satisfactoriamente proyectos del mismo tipo?

¿El proyecto se deriva de otro trabajo realizado? ¿Cuál? Si No

Regresar Guardar Como Borrador Autorizar Envío Salir

- La siguiente interfaz corresponde a la información de soporte para la ejecución de la propuesta, donde se indican y relacionan los tipos de recursos físicos que se usarán durante la ejecución de la propuesta.

La siguiente interfaz corresponde a la información de soporte para la ejecución de la propuesta, donde se indican y relacionan los tipos de recursos físicos que se usarán durante la ejecución de la propuesta.

Figura A3.

Pantalla información de soporte. Fuente (Documentación oficial).

- La siguiente interfaz corresponde a Destinatarios, donde se ingresa información de entidades receptoras del servicio y sus responsables administrativos y técnicos.

Figura A5.

Pantalla de destinatarios. Fuente (Documentación oficial).

Apéndice B. Identificación de requerimientos funcionales y no funcionales**Tabla B1.**

Requerimientos funcionales sobre el manejo de la información básica de una propuesta de extensión.

ID	Descripción	Prioridad
RFB1	El usuario con rol de "Docente Planta" debe poder acceder al formulario para el registro de una propuesta de extensión desde el módulo correspondiente en el Sistema de Información Académica (SIA).	Alta
RFB2	Debe ser posible guardar la información ingresada en el formulario, permitiendo así que los usuarios puedan avanzar en el proceso de registro de la propuesta de extensión.	Alta
RFB3	El sistema debe permitir a los usuarios con rol de "Docente Planta" seleccionar un servicio de extensión asociado a la propuesta desde un campo específico.	Media
RFB5	En caso de no encontrar un servicio de extensión adecuado, el sistema debe permitir la creación de un servicio provisional proporcionando un mensaje de información sobre el campo.	Alta
RFB6	El servicio provisional que se crea solo puede estar asociado a una propuesta de extensión, luego de eso no saldría en la lista de servicios posibles a seleccionar dentro de una nueva	Alta

	propuesta.	
RFB6	El sistema debe permitir a los usuarios ingresar un título único en la base de datos para la propuesta de extensión, debe ser un campo de entrada de texto y obligatorio.	Alta

Tabla B2.

Requerimientos no funcionales sobre el manejo de la información básica de una propuesta de extensión.

ID	Descripción	Prioridad
RNFB1	La propuesta debe tener un servicio de extensión asociado, por ende, este campo debe ser obligatorio para avanzar en el proceso de registro.	Alta
RNFB2	Los usuarios deben de forma obligatoria ingresar una descripción del objetivo o los objetivos de la propuesta de extensión.	Media
RNFB3	Los usuarios deben de forma obligatoria seleccionar el marco legal asociado a la propuesta de extensión.	Media
RNFB4	Los usuarios deben de forma obligatoria seleccionar la unidad académica administrativa para la cual se desarrollará la	Media

	propuesta y su valor predeterminado debe ser la unidad del docente.	
RNFB5	Se debe mostrar automáticamente al director principal de la propuesta, estando deshabilitado para su edición y siendo esta la misma persona "docente planta" que está diligenciando y/o creando la propuesta.	Media
RNFB6	Los usuarios deben de forma obligatoria seleccionar un ordenador de gasto para la propuesta.	Media

Tabla B3.

Requerimientos funcionales sobre el manejo de las características del proyecto.

ID	Descripción	Prioridad
RFC1	El sistema debe permitir al usuario seleccionar el tipo de iniciativa asociada a la propuesta.	Alta
RFC2	El sistema debe permitir al usuario ingresar otro tipo de iniciativa no contemplada en la lista anterior, si selecciona "Otro" como opción en el campo tipo de Iniciativa.	Media
RFC3	El sistema debe permitir al usuario seleccionar la fecha límite para la presentación de la propuesta.	Media
RFC4	El sistema debe permitir al usuario ingresar otro lugar de ejecución si el lugar que desea no está en la lista de lugares de	Media

	ejecución y selecciona "Otro" como opción.	
RFC5	El sistema debe permitir al usuario indicar la población que será atendida por la propuesta.	Alta
RFC6	El sistema debe permitir al usuario indicar la clase de los beneficiarios.	Alta
RFC7	El sistema debe permitir al usuario indicar la clase de los beneficiarios.	Alta
RFC8	El sistema debe permitir al usuario llenar información sobre la misión y coherencia de la propuesta con el proyecto institucional.	Alta
RFC9	El sistema debe permitir al usuario llenar la razón por la cual el proyecto es pertinente en el medio.	Alta
RFC10	El sistema debe permitir al usuario indicar si ha realizado proyectos similares anteriormente.	Alta
RFC11	El sistema debe permitir al usuario indicar el número de proyectos previos del mismo tipo, si ha seleccionado "Sí" en el campo anterior.	Alta
RFC12	El sistema debe permitir al usuario indicar si este proyecto es derivado de otro proyecto anterior.	Alta
RFC13	El sistema debe permitir al usuario indicar el nombre del	Alta

proyecto del cual se origina, si ha indicado que este proyecto se origina de otro.
--

Tabla B4.

Requerimientos funcionales sobre el manejo de las características del proyecto.

ID	Descripción	Prioridad
RNFC1	Se debe implementar una validación para limitar el número de personas beneficiarias a un máximo de 5,000,000, conforme a los requisitos establecidos.	Media
RNFC2	Se debe validar que el campo de fecha límite sólo permita seleccionar fechas a partir de la fecha actual, para asegurar la coherencia temporal de las propuestas.	Media
RNFC3	Los usuarios deben de forma obligatoria seleccionar un lugar de ejecución para la propuesta.	Alta
RNFC4	Si se selecciona que sí ha concluido satisfactoriamente proyectos del mismo tipo, se debe validar que se seleccione un rango de proyectos ejecutados del mismo tipo.	Alta
RNFC5	Si se selecciona que, si el proyecto se deriva de otro, se debe validar que se ingrese el nombre del proyecto del cual se origina en el campo correspondiente.	Media

Tabla B5.

Requerimientos funcionales sobre la información de soporte de una propuesta de extensión.

ID	Descripción	Prioridad
RFS1	El sistema debe permitir a los usuarios con el rol de "Docente Planta" completar la información sobre soporte de la propuesta de extensión.	Alta
RFS2	El sistema debe permitir la búsqueda y selección de recursos físicos de la universidad, permitiendo agregar nuevos recursos si no existen.	Alta
RFS3	El sistema debe calcular correctamente el tiempo de ejecución de la propuesta a partir de las fechas de inicio y fin estimadas.	Alta
RFS4	El sistema debe habilitar o deshabilitar el campo de ingreso adicional "Otro" según la selección en el campo de participación de la universidad.	Baja

Tabla B6.

Requerimientos no funcionales sobre la información de soporte de una propuesta de extensión.

ID	Descripción	Prioridad
RNFS1	El sistema debe ser capaz de manejar la carga de trabajo asociada con la búsqueda, selección y gestión de recursos físicos de	Media

	manera eficiente y sin retrasos perceptibles para el usuario.	
RNFS2	El sistema debe validar que la fecha de inicio ingresada por el usuario sea igual o mayor que la fecha actual.	Media
RNFS3	El sistema debe validar que la fecha fin ingresada por el usuario sea mayor o igual a la fecha de inicio.	Media

Tabla B7.

Requerimientos funcionales sobre el destinatario de una propuesta de extensión.

ID	Descripción	Prioridad
RF1	El sistema debe permitir registrar la entidad destinataria.	Alta
RF2	El sistema debe permitir ingresar la información del responsable técnico, incluyendo nombre y cargo.	Alta
RF3	El sistema debe permitir agregar detalles de contacto del responsable técnico, como teléfono y email.	Alta
RF4	El sistema debe permitir ingresar la información del responsable administrativo, incluyendo nombre y cargo.	Alta
RF5	Se debe permitir indicar si se ha trabajado anteriormente con la entidad seleccionada.	Baja
RF6	El sistema debe permitir añadir proyectos/trabajos previos realizados con la entidad destinataria seleccionada.	Media

RF7	El sistema debe permitir agregar más de una entidad destinataria para la propuesta de extensión.	Baja
-----	--	------

Tabla B8.

Requerimientos no funcionales sobre el destinatario de una propuesta de extensión.

ID	Descripción	Prioridad
RNF1	El sistema debe validar que cuando "El destinatario de la propuesta es externo" sea "SI", todos los campos relacionados sean obligatorios y permitan ingresar información relevante.	Alta
RNF2	El sistema debe verificar que cuando "El destinatario de la propuesta es externo" sea "NO", no se permita ingresar datos en ningún campo relacionado con el destinatario externo.	Media
RNF3	Se debe implementar un comportamiento dinámico en el formulario que permita agregar y eliminar campos para otros proyectos/trabajos realizados con la entidad destinataria seleccionada, así como agregar otro formulario para otra entidad destinataria.	Media

Apéndice C. Pantallas diseñadas para el nuevo sistema

Figura C1.

Diseño para el manejo de la información básica de una propuesta de extensión. Fuente (Análisis y documentación).

The screenshot shows a mobile application interface for creating an extension proposal. The title is 'Propuesta extensión'. At the top, there is a progress indicator with four steps: 'Información básica', 'Características del proyecto', 'Soporte', and 'Destinatario'. Below this, there is a section for 'Unidad académica administrativa UAA' with fields for 'UAA', 'Jefe de unidad', 'Director Proyecto', and 'Ordenador de gasto'. The main section is 'Información básica de la propuesta extensión', which includes fields for 'Nombre del servicio', 'Título del proyecto', and 'Objetivo'. There are also radio buttons for 'Sentido ordinario' (Yes/No) and 'Fortalecimiento institucional' (Yes/No), along with a 'Marco legal' dropdown menu. A 'Guardar y seguir' button is located at the bottom right.

Figura C2.

Diseño de ventana modal para la creación de un servicio provisional. Fuente (Análisis y documentación).

The screenshot shows a modal window titled 'Modal crear servicio' with the subtitle 'Crear servicio para la propuesta de extensión'. It is divided into three sections: 'General', 'Clase', and 'Aplicación del servicio'. The 'General' section has fields for 'Nombre del servicio' and 'Descripción del servicio'. The 'Clase' section has fields for 'Modalidad', 'Sub modalidad', 'Clasificación por disciplina', and 'Línea de acción'. The 'Aplicación del servicio' section has a 'Todos' checkbox and a list of 'Sectores de aplicación del servicio' including 'Industrias Manufactureras', 'Administración Pública Y Defensa, Seguridad Social de Afiliación Obligatoria', 'Agricultura', and 'Industrias Manufactureras'. At the bottom, there are 'Cancelar' and 'Guardar' buttons.

Figura C3.

Diseño para el manejo de las características del proyecto. Fuente (Análisis y documentación).

Stepper 2 Características del proyecto

Propuesta extensión

Información básica | Características del proyecto | Soporte | Destinatario

Esta sección corresponde a los aspectos generales de la propuesta de extensión

Características del proyecto # Código del proyecto

Tipo iniciativa otro (se activa cuando se selecciona la opción "otros")

Fecha límite Lugar ejecución

otro (se activa cuando se selecciona la opción "otros")

Población atendida

0 de 1000 caracteres

Clase de personas beneficiarias Número aproximado de personas beneficiaria

Descripción de la propuesta

¿Cuál es el componente misional y la coherencia de la actividad con el Proyecto Institucional?

0 de 1000 caracteres

¿Por qué es pertinente este proyecto de extensión en el medio?

¿Ha concluido satisfactoriamente proyectos del mismo tipo? Sí No Seleccione el rango de proyectos del mismo tipo realizados previamente

¿El proyecto se deriva de otro trabajo realizado? Sí No Nombre del proyecto realizado del cual se deriva la propuesta de extensión

Anterior Guardar y seguir

Figura C4.

Diseño para el manejo de la información de soporte de una propuesta de extensión. Fuente (Análisis y documentación).

Stepper 3- Soporte

Propuesta extensión

Información básica | Características del proyecto | Soporte | Destinatario

Recursos Físicos de la Universidad a Utilizar

Esta corresponde a la capacidad institucional en recursos físicos (equipo, software e infraestructura), recursos académicos, talento humano y en logística, disponibilidad para comprometerla en los alcances y tiempos establecidos en la propuesta.

Nombre del recurso físico + + Agregar recurso físico

Filtrar registros

Tipo	Nombre	características	Cantidad	Acciones
Software	softworks	modelo 3d	<input type="text" value="1"/>	<input type="text"/>

Manejo Propiedad Intelectual

Fecha de inicio estimado de actividad Fecha fin estimado de actividad *Tiempo de ejecución de la propuesta

Todos

Participación de la universidad

Recurso humano recurso académica recurso financiero recursos logísticos recursos tecnológicos otro

Otro

Anterior Guardar y seguir

Figura C5.

Diseño de ventana modal para la creación de un recurso físico. Fuente (Análisis y documentación).



Modal crear recurso físico de la universidad

Crear recurso físico de la universidad

General

Nombre Tipo de recurso

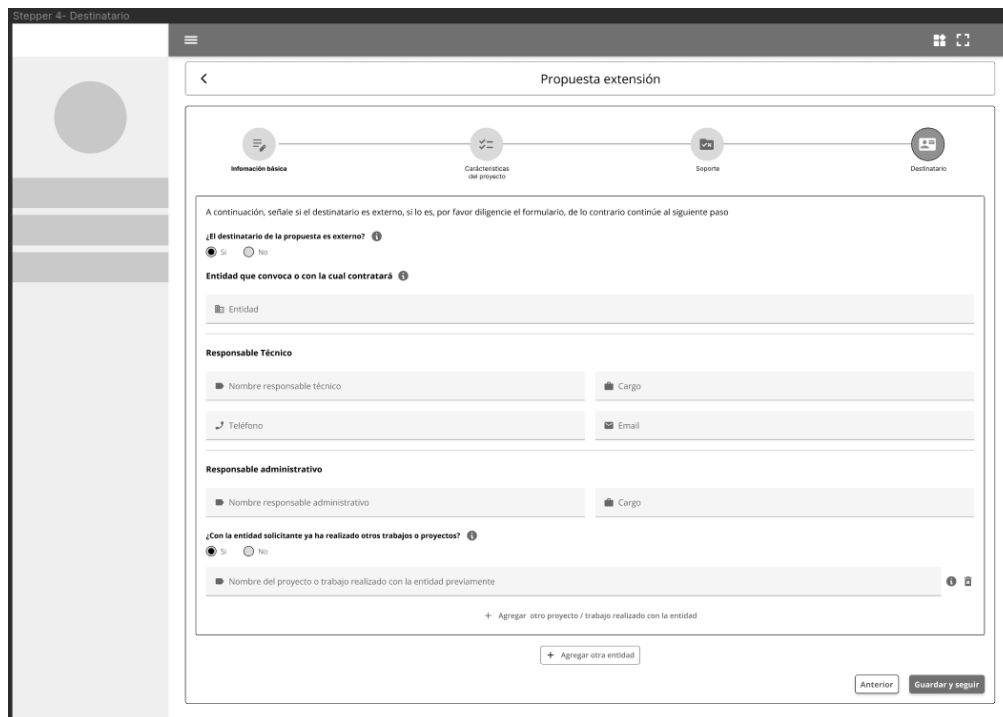
Características

0 de 1000 caracteres

Cancelar Guardar

Figura C6.

Diseño para el manejo del destinatario de una propuesta de extensión. Fuente (Análisis y documentación).



Stepper 4- Destinatario

Propuesta extensión

Información básica Características del proyecto Soporte Destinatario

A continuación, señale si el destinatario es externo, si lo es, por favor diligencie el formulario, de lo contrario continúe al siguiente paso

¿El destinatario de la propuesta es externo? Sí No

Entidad que convoca o con la cual contratará

Responsable Técnico

Nombre responsable técnico Cargo

Teléfono Email

Responsable administrativo

Nombre responsable administrativo Cargo

¿Con la entidad solicitante ya ha realizado otros trabajos o proyectos? Sí No

Nombre del proyecto o trabajo realizado con la entidad previamente

+ Agregar otro proyecto / trabajo realizado con la entidad

+ Agregar otra entidad

Anterior Guardar y seguir

Apéndice D. Sprints desarrollados

Sprint 1

→ **HU #675: Crear propuestas de extensión no solidarias: step 1 información básica**

En esta historia se desarrollaron endpoints relacionados a las funcionalidades del step 1, algunos de los endpoints que se desarrollaron son los siguientes:

- Endpoint para guardar la información básica de la propuesta.
- Endpoint para la edición de la información básica de la propuesta.
- Endpoint para la búsqueda del servicio por nombre.
- Endpoint para la creación de un servicio provisional.
- Endpoint para listar los sectores de aplicación de un servicio.
- Endpoint para listar las modalidades de un servicio.
- Endpoint para listar las submodalidades de un servicio.

● **Desarrollo de la capa Frontend.**

Desde la capa Frontend durante este sprint se desarrollaron las siguientes historias de usuario.

→ HU #814: Creación y estabilización del repositorio Front Extensión

Para el frontend, el proceso de creación del repositorio se lleva a cabo en Gitlab, donde se gestionan todos los repositorios de los proyectos de RSI. En este caso, se crea un repositorio llamado "Front Extensión", que funciona como el repositorio principal para el desarrollo del frontend de la aplicación web de extensión.

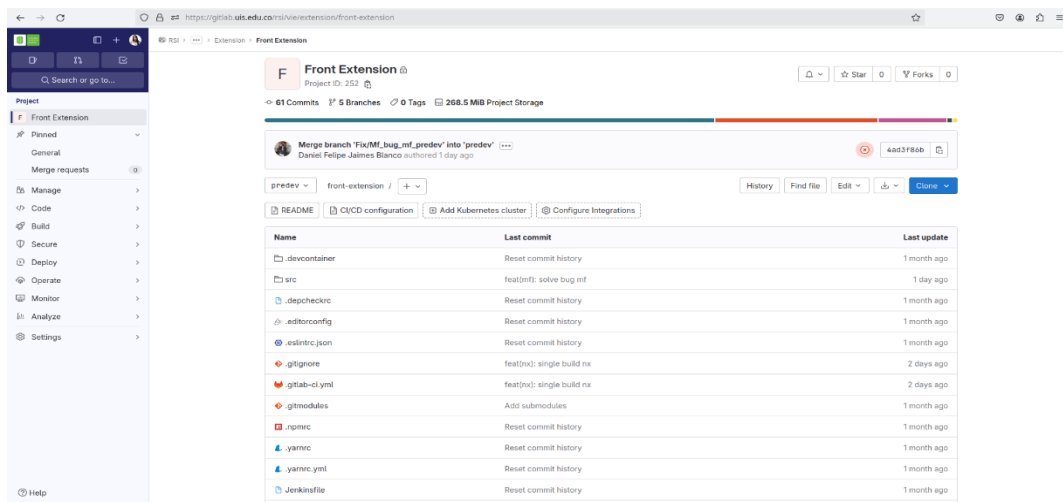
En este repositorio principal, se incluyen varios submódulos clave para organizar y gestionar el código de manera eficiente:

- **Environments:** Este submódulo se utiliza para cargar todas las URL de los diferentes backends en las diferentes ramas del proyecto.
- **Layout:** En este submódulo se encuentran los archivos CSS, imágenes y otros recursos necesarios para dar formato y diseño a las diferentes pantallas y componentes de la aplicación.
- **Karma:** Este submódulo está destinado a la ejecución de pruebas unitarias utilizando el framework de pruebas para Angular, Karma. Contiene configuraciones y scripts necesarios para ejecutar y automatizar las pruebas unitarias del código frontend.
- **Shared:** Este submódulo contiene información compartida y reutilizable en toda la aplicación. Aquí se incluyen componentes, servicios y otros recursos compartidos entre diferentes partes de la aplicación, lo que ayuda a mantener un código limpio y modular.

Una vez que se han configurado todos los submódulos y se ha probado el correcto funcionamiento de la aplicación en el entorno local, el código se sube al repositorio Gitlab para su revisión.

Figura D1.

Repositorio Front Extensión. Fuente (Gitlab UIS). [\[22\]](#)

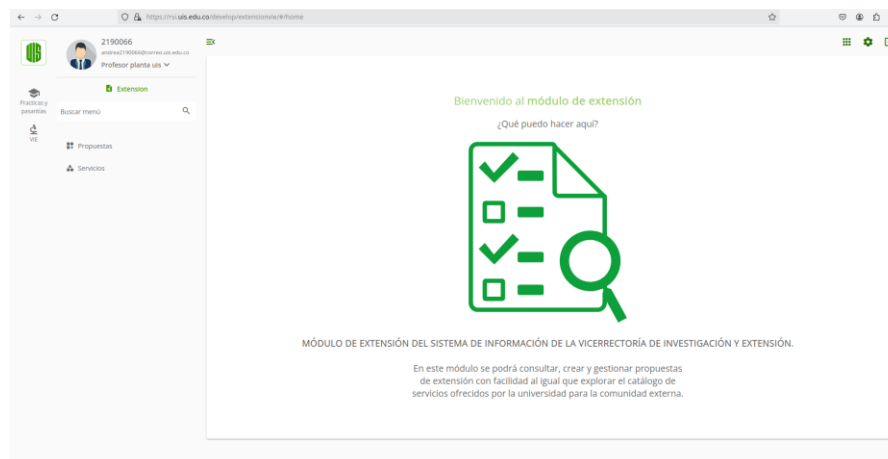


→ **HU #798: Crear portada para el módulo de extensión**

Como usuario del sistema de información de la Vicerrectoría de Investigación y Extensión, se requiere ver una portada atractiva e informativa al acceder al Módulo de Extensión para comprender de manera clara y rápida las funciones y opciones disponibles en este módulo.

Figura D2.

Portada módulo extensión. Fuente (Sistemas de información - SIA UIS). [\[23\]](#)



→ **HU #782: Crear propuestas de extensión no solidarias: CRUD y consulta Propuestas**




Como usuario del sistema con rol “docente planta” o “jefe de unidad” se requiere poder consultar las propuestas de extensión e iniciar un formulario para la creación de propuestas nuevas.

Figura D3.

Formulario para consultar propuestas de extensión. Fuente (Sistemas de información - SIA UIS). [24]

Figura D4.

Consulta y gestión de propuestas de extensión. Fuente (Sistemas de información - SIA UIS).

Código del proyecto	Título	UAA	Fecha de creación	Estado	Acciones
77.7	Pruebas	640 - Facultad De Ing. Química	07/04/2024	En borrador	  

→ HU #675: Crear propuestas de extensión no solidarias: Step 1 Información Básica

Como usuario del sistema con rol “docente planta” se requiere poder crear una propuesta de extensión y llenar la información básica de esta, además de poder asociar en este step el servicio (ya sea traído de catálogo o creando uno provisional) de la propuesta de extensión.

Figura D5.

Creación de la información básica de una propuesta de extensión. Fuente (Sistemas de información - SIA UIS).

The screenshot shows a web browser window with the URL <https://rsi.uis.edu.co/develop/extensionvie/#/propuestas/crear-propuesta>. The user is logged in as '2190066 profesor@correo.uis.edu.co' with the role 'Profesor planta uis'. The page title is 'Crear - Propuesta de extensión'. A progress bar at the top indicates the current step is 'Información básica', followed by 'Características del proyecto', 'Soporte', and 'Destinatario'. The form contains the following fields and options:

- Unidad académica administrativa UAA: A dropdown menu with 'UISA' selected.
- Ordenador de gasto: A dropdown menu.
- Información básica de la propuesta de extensión:
 - Nombre del servicio: A text input field with a '+' icon on the right.
 - Título del proyecto: A text input field.
 - Objetivo: A text input field with a height indicator of 0 to 100 caracteres.
- Marco legal: A dropdown menu.
- Sentido solidario: Radio buttons for 'Si' and 'No', with 'No' selected.
- Fortalecimiento institucional: Radio buttons for 'Si' and 'No', with 'Si' selected.
- A green 'Guardar y seguir' button is located at the bottom right of the form.

Figura D6.

Edición de la información básica de una propuesta de extensión. Fuente (Sistemas de información - SIA UIS).

The screenshot shows a web browser window with the URL <https://rsi.uis.edu.co/develop/extensionvie/#/proposas/editar-propuesta/7>. The user is logged in as '2190066' (Profesor planta us). The page title is 'Editar - Propuesta de extensión'. The main content area is divided into four sections: 'Información básica', 'Características del proyecto', 'Soporte', and 'Destinatario'. The 'Información básica' section contains several form fields: 'Unidad académica administrativa UAA' (set to '6456 - Escuela De Ing. Química'), 'Unidad' (set to 'Artes Chaves Guerrero'), 'Director proyecto' (set to 'Andrea Juliana Urrego Paredes'), and 'Destinatario' (set to 'Adriana Jose Bernal Gonzalez'). Below these are fields for 'Nombre del servicio' (set to '601 - Servicio De Pruebas'), 'Título del proyecto' (set to 'Prueba'), 'Objetivo' (set to 'Probando'), 'Marco legal' (set to 'Admisión Al Público'), 'Sentido solidario' (radio buttons for 'SI' and 'No', with 'No' selected), and 'Fortalecimiento institucional' (radio buttons for 'SI' and 'No', with 'SI' selected). A 'Guardar y seguir' button is at the bottom right.

Figura D7.

Ventana modal para la creación de un servicio provisional. Fuente (Sistemas de información - SIA UIS).

The screenshot shows a modal window titled 'Crear servicio único provisional para la propuesta de extensión'. The modal is overlaid on the 'Editar - Propuesta de extensión' page. It contains the following fields: 'Nombre del servicio', 'Descripción del servicio', 'Clase' (with sub-fields for 'Modalidad' and 'Sub modalidad'), 'Agrupación por disciplina', and 'Línea de acción'. Under 'Aplicación del servicio', there is a checkbox for 'Todos' and a section for 'Sector de aplicación del servicio'. At the bottom of the modal are 'Limpiar formulario' and 'Guardar' buttons.

→ HU #842: Rebajar deuda técnica proyectos de extensión

Los criterios de aceptación que se cumplieron en el Dev-Test fueron los siguientes:

1. Realizar una revisión exhaustiva de cada proyecto (back propuestas, back servicios y front extensión) utilizando SonarQube.
2. Identificar y abordar específicamente los puntos de deuda técnica en los proyectos mencionados.
3. Colocar en “predev” un nuevo merge request después de los ajustes, para que en cada nuevo despliegue (pipeline exitoso) se minimice la deuda técnica.
4. Verificar que las mejoras no introduzcan nuevos problemas o errores en los proyectos.
5. Dejar en SonarQube una deuda técnica en días de un valor que tienda a cero.

→ HU #798: Crear portada para el módulo de extensión

Los criterios de aceptación que se cumplieron en el Dev-Test fueron los siguientes:

1. Cuando acceda al sistema y seleccione el Módulo de Extensión, quiero ser recibido por una portada clara y bien diseñada.
2. La portada debe contener el mensaje "Bienvenido al Módulo de Extensión" para indicar claramente el propósito del módulo.
3. Debe haber una sección que explique de manera concisa las acciones y funcionalidades disponibles en el Módulo de Extensión.
4. Se deben utilizar elementos visuales (iconos, gráficos) para hacer la portada más atractiva

y fácil de entender.

→ **HU #818: Creación y estabilización del repositorio backend**

Los criterios de aceptación que se cumplieron en el Dev-Test fueron los siguientes:

1. Se debe crear un nuevo repositorio en el sistema de control de versiones con el nombre "Back Propuestas" y otro con el nombre "Back Servicios".
2. Se debe configurar un archivo de entorno para manejar configuraciones específicas del ambiente de desarrollo, como la conexión a la base de datos y configuraciones de servidor.
3. La estructura de carpetas y archivos iniciales debe seguir las mejores prácticas de desarrollo backend.
4. Se deben establecer carpetas específicas para módulos, rutas, controladores y servicios relacionados con la gestión del módulo de extensión.
5. Se debe crear al menos un endpoint inicial por cada backend que pueda manejar solicitudes relacionadas con la gestión del módulo de extensión.

→ **HU #814: Creación y estabilización del repositorio Front Extensión**

Los criterios de aceptación que se cumplieron en el Dev-Test fueron los siguientes:

1. Se debe crear un nuevo repositorio en el sistema de control de versiones con el nombre "Front-Extensión".
2. Se debe establecer la estructura de carpetas y archivos iniciales siguiendo las mejores prácticas de desarrollo frontend.

→ HU #782: Crear propuestas de extensión no solidarias: CRUD y consulta Propuestas

Los criterios de aceptación que se cumplieron en el Dev-Test fueron los siguientes:

1. Se debe poder digitar cualquiera de los campos para realizar una consulta, en caso de no seleccionarse un registro debe generar el error de requerido para el campo unidad gestora (ya que es obligatoria para la consulta).
2. No debe haber errores de consola y se debe validar la funcionalidad de todos los campos.
3. Se debe validar la consulta de los datos. Usar o crear datos de prueba en la base de datos para este criterio.
4. Verificar traducciones i18N en todos los labels y textos que no son de la base de datos.

→ HU #675: Crear propuestas de extensión no solidarias: Step 1 Información Básica

Los criterios de aceptación que se cumplieron en el Dev-Test fueron los siguientes:

1. Todos los campos deben estar debidamente ajustados a los lineamientos del mockup.
 2. Al guardar el servicio y propuesta se debe revisar los registros en base de datos verificando el almacenamiento correcto de estos.
 3. Verificar traducciones i18N en todos los labels y textos que no son de la base de datos.
- **Corrección de errores y depuración.**

Durante este sprint luego de pasar por el correspondiente Dev-Test realizado a cada

historia de usuario solo se obtuvo el siguiente issue, luego corregido satisfactoriamente:

Escenario 1: Cuando se crea un servicio provisional para la propuesta de extensión el botón de cancelar que se encuentra en el modal se comporta como un botón de limpiar formulario.

Figura D8.

Issue #880 Step 1 Información básica. Fuente (Sistemas de información - SIA UIS).

Escenario 1: cuando se crea un servicio provisional para la propuesta de extensión el botón de cancelar que se encuentra en el modal se comporta como un botón de limpiar formulario

Crear servicio único provisional para la propuesta de extensión ✕

General

Nombre del servicio *

Descripción del servicio *

0 de 1000 caracteres

Clase

Modalidad * Sub modalidad

AZ Clasificación por disciplina * Línea de acción

Aplicación del servicio

Todos

Sectores de aplicación del servicio

Cancelar Guardar

Sprint 2

Duración del Sprint: Del 29 de enero al 21 de febrero.

Figura D9.

Backlog del sprint 2. Fuente (Taiga).



- **Reestructuración de base de datos.**

Las tablas creadas y la migración de información realizada para este sprint son las siguientes:

1. **Vie_extension_tipos_iniciativa:** Tabla que almacena los diferentes tipos de iniciativas

relacionadas con la extensión.

2. **Vie_extension_lista_rangos_proyectos:** Tabla que contiene una lista de los distintos rangos de proyectos relacionados con la extensión.

3. **Vie_extension_propuestas_caracteristicas:** Tabla que guarda las características y detalles de las propuestas relacionadas con la propuesta.

La información que se migró corresponde a los tipos de iniciativa y lista de rangos de proyecto, se crearon relaciones para poder anidar el lugar de ejecución el cual corresponde a una sede de la universidad u otro lugar si se especifica.

- **Desarrollo de la capa Backend.**

Desde la capa Backend durante este sprint se desarrollaron las siguientes historias de usuario.

→ **HU #869: Acceso al módulo por roles y estados**

En esta historia se crearon ciertas validaciones para añadir restricciones de acuerdo al rol y estado actual de la propuesta. También se añadieron restricciones a las búsquedas de los usuarios, pues dependiendo del rol se muestran los estados para poder filtrar la información y dependiendo del estado se pueden realizar ciertas acciones sobre la propuesta. Estas restricciones pueden ser observadas en los criterios de aceptación en los Dev-Test realizados durante el sprint.

→ **HU #807: Propuestas de extensión no solidarias: Step 2 Características**

Se implementaron varios endpoints importantes para la gestión de características de una propuesta. Algunos de los más relevantes son:

- Endpoint para guardar la información de las características de una propuesta.
- Endpoint para la edición de la información de las características de una propuesta.
- Endpoint para obtener toda la información de características de una propuesta.
- Endpoint para listar los lugares de ejecución.
- Endpoint para listar los rangos de proyecto que se han realizado del mismo tipo.

- **Desarrollo de la capa Frontend.**

Desde la capa Frontend durante este sprint se desarrollaron las siguientes historias de usuario.

→ **HU #807: Propuestas de extensión no solidarias: Step 2 Características**

Como “docente planta” se requiere poder llenar el segundo step con la información de características de la propuesta de extensión, permitir guardar y editar la información, además de seguir al siguiente step:

Figura D10.

Creación de las características del proyecto. Fuente (Sistemas de información - SIA UIS).

Crear - Propuesta de extensión

En este módulo podrá ingresar las características del proyecto.

Información básica | **Características del proyecto** | Soporte | Destinatario

Esta sección corresponde a los aspectos generales de la propuesta de extensión

Características del proyecto Código de la propuesta: 77-10

Tipo iniciativa *

Fecha límite * Lugar ejecución *

Población atendida * 0 de 1000 caracteres

Clase de personas beneficiarias Número aproximado de personas beneficiarias *

Descripción de la propuesta

¿Cuál es el componente misional y la coherencia de la actividad con el Proyecto Institucional? *

0 de 1000 caracteres

¿Por qué es pertinente este proyecto de extensión en el medio? *

0 de 1000 caracteres

¿Ha concluido satisfactoriamente proyectos del mismo tipo?
 Sí No Seleccione el rango de proyectos del mismo tipo realizados previamente *

¿El proyecto se deriva de otro trabajo realizado?
 Sí No Nombre del proyecto realizado del cual se deriva la propuesta de extensión *

Anterior Guardar y seguir

Figura D11.

Edición de las características del proyecto. Fuente (Sistemas de información - SIA UIS).

Editar - Propuesta de extensión

En este módulo podrá editar las características del proyecto.

Información básica | **Características del proyecto** | Soporte | Destinatario

Esta sección corresponde a los aspectos generales de la propuesta de extensión

Características del proyecto Código de la propuesta: 77-8

Tipo iniciativa *

Fecha límite * Lugar ejecución *

Población atendida * 0 de 1000 caracteres

Clase de personas beneficiarias Número aproximado de personas beneficiarias *

Descripción de la propuesta

¿Cuál es el componente misional y la coherencia de la actividad con el Proyecto Institucional? *

Prueba 0 de 1000 caracteres

¿Por qué es pertinente este proyecto de extensión en el medio? *

Prueba 0 de 1000 caracteres

¿Ha concluido satisfactoriamente proyectos del mismo tipo?
 Sí No Seleccione el rango de proyectos del mismo tipo realizados previamente *
 20, más de 5 proyectos similares

¿El proyecto se deriva de otro trabajo realizado?
 Sí No Nombre del proyecto realizado del cual se deriva la propuesta de extensión *

Anterior Guardar y seguir

- **Dev Test - Test de funcionalidades.**

→ **HU #869: Acceso al módulo por roles y estados**

1. Validar que los permisos por acción y rol se implementen:
 - I. Un “docente planta” puede crear, editar, visualizar y eliminar propuestas.
 - II. Un “jefe de unidad” puede visualizar solamente.
 - III. Un “auxiliar de la VIE” puede visualizar.
 - IV. Un “profesional VIE” puede visualizar.
 - V. El “jefe DTC” puede visualizar.
2. Verificar que los permisos por acción y estado se apliquen:
 - I. En estado “borrador”, se permite editar, visualizar y eliminar.
 - II. En estado “en trámite”, solo se permite visualizar.
 - III. En estado “en corrección”, se permite visualizar y editar, pero no eliminar.
 - IV. En estado “en avales”, solo se permite visualizar.

→ **HU #807: Propuestas de extensión no solidarias: Step 2 Características**

1. Se debe poder acceder al step 2 desde el step 1 luego de guardar.
2. Todos los campos deben estar debidamente ajustados a los lineamientos del mockup.
3. Al guardar las características de la propuesta se deben revisar los registros en base de datos verificando el almacenamiento correcto de estos.

- **Corrección de errores y depuración.**

Para la tarea de corrección de errores y depuración, se realizó una revisión por parte del equipo de QA para identificar posibles problemas o fallos en el sistema. Sin embargo, durante este sprint, no se encontraron hallazgos por parte de QA. El sistema se sometió a pruebas, y todos los casos de prueba pasaron satisfactoriamente

Sprint 3

Duración del Sprint: Del 26 de febrero al 20 de marzo.

Figura D12.

Backlog del sprint 3. Fuente (Taiga).



- **Reestructuración de base de datos.**

Las tablas que se crearon y migraron en este sprint fueron las siguientes:

1. **Vie_extension_propiedad_intelectual:** Tabla para el manejo de la propiedad intelectual.
2. **Vie_extension_participacion_universidad:** Tabla para el manejo de la participación de la universidad.
3. **Vie_extension_tipos_recurso:** Tabla para el manejo de los tipos de recurso.
4. **Vie_extension_recurso:** Tabla para el manejo de los recursos.
5. **Vie_extension_propuestas_soporte:** Tabla para el manejo de las propuestas de soporte.
6. **Vie_extension_propuestas_rel_soporte_participacion:** Tabla para el manejo de la relación entre la propuesta de soporte y la participación de la universidad.
7. **Vie_extension_propuestas_rel_soporte_recurso:** Tabla para el manejo de la relación entre la propuesta de soporte y los recursos.

- **Desarrollo de la capa backend.**

Desde la capa Backend durante este sprint se desarrollaron las siguientes historias de usuario.

→ **HU #881: Propuestas de extensión no solidarias: Step 3 Soporte**

Se implementaron endpoints importantes para la visualización y guardado de la información necesaria para que el usuario ingrese la información del paso tres de soporte. Entre los endpoints implementados están los siguientes:

- Endpoint para buscar recursos dado un nombre.
- Endpoint para traer toda la información de un recurso en específico.
- Endpoint para crear un servicio en caso de que no exista.
- Endpoint para guardar toda la información del paso 3 de Soporte.

→ **HU #883: Propuestas de extensión no solidarias: Step 4 Destinatario**

Los endpoints implementados durante esta historia de usuario buscan permitir el ingreso de información de los destinatarios de la propuesta, generalmente si el destinatario de la propuesta es la misma universidad no hay necesidad de llenar todo el formulario.

En caso contrario, cuando el destinatario es externo, se debe seleccionar la entidad a la que va dirigida la propuesta, para ello se crearon endpoints como:

- Endpoint para la búsqueda de entidad por nombre.
- Endpoint para el registro de información de los proyectores anteriores.
- Endpoint para guardar información de los destinatarios.

● **Desarrollo de la capa frontend.**

Desde la capa Frontend durante este sprint se desarrollaron las siguientes historias de usuario.

→ **HU #881: propuestas de extensión no solidarias: step 3 soporte**

Como “docente planta” se requiere llenar el tercer stepper con la información de soporte de la propuesta de extensión, permitir guardar y editar la información, además de seguir al

siguiente stepper.

Figura D13.

Creación de la información de soporte de una propuesta de extensión. Fuente (Sistemas de información - SIA UIS).

Figura D14.

Edición de la información de soporte de una propuesta de extensión. Fuente (Sistemas de información - SIA UIS).

Tipo	Nombre	Características	Cantidad	Acciones
Infraestructura	Sala De Computo Ciencias Humanas	30 personas	1	

→ HU #883: propuestas de extensión no solidarias: step 4 destinatario

Como “docente planta” se requiere llenar el cuarto step con la información del destinatario de la propuesta de extensión, permitir guardar y editar la información, además de seguir al siguiente step.

Figura D15.

Creación del destinatario de una propuesta de extensión. Fuente (Sistemas de información - SIA UIS).

Crear - Propuesta de extensión

En este módulo podrá ingresar información sobre el destinatario de la propuesta.

Información básica Características del proyecto Soporte Destinatario

A continuación, señale si el destinatario es externo, si lo es, por favor diligencie el formulario, de lo contrario continúe al siguiente paso.

¿El destinatario de la propuesta es externo? Sí No

Entidad que comienza o con la cual contratará

Entidad *

Responsable técnico

Nombre responsable técnico * Cargo *

Teléfono * Correo electrónico *

Responsable administrativo

Nombre responsable administrativo * Cargo *

¿Con la entidad solicitante ya ha realizado otros trabajos o proyectos? Sí No

Nombre del proyecto o trabajo realizado con la entidad previamente *

+ Agregar otro proyecto / trabajo realizado con la entidad

+ Agregar otra entidad

Anterior **Guardar y seguir**

Figura D16.

Edición del destinatario de una propuesta de extensión. Fuente (Sistemas de información - SIA UIS).

- **Dev Test - test de funcionalidades.**

Para evaluar de manera efectiva si se cumplen los criterios de aceptación de cada historia de usuario, se cumplieron con los siguientes dentro de cada historia de usuario perteneciente al sprint 3:

→ **HU #881: propuestas de extensión no solidarias: step 3 soporte**

1. Se puede completar el tercer stepper con la información proporcionada.
2. Se puede guardar la información correctamente en la base de datos.
3. La tabla de detalle de recursos permite agregar y eliminar registros correctamente.
4. Los campos requeridos se validan adecuadamente antes de guardar la información.
5. El tiempo de ejecución de la propuesta se calcula correctamente a partir de las fechas de inicio y fin estimadas.

6. El campo "otro" se habilita/deshabilita correctamente según la selección en el campo de participación de la universidad.
7. Se puede crear un nuevo recurso y agregarlo a la tabla de recursos físicos.

→ **HU #883: propuestas de extensión no solidarias: step 4 destinatario**

1. Validar que el campo "El destinatario de la propuesta es externo" sea obligatorio y permita marcar solo una opción: "SI" o "NO".
2. Validar que, cuando "El destinatario de la propuesta es externo" sea "SI", todos los campos relacionados sean obligatorios y permitan ingresar información relevante.
3. Verificar que, cuando "El destinatario de la propuesta es externo" sea "NO", no se permita ingresar datos en ningún campo relacionado con el destinatario externo.
4. Implementar un comportamiento dinámico en el formulario que permita agregar y eliminar campos para otros proyectos/trabajos realizados con la entidad destinataria, así como agregar otro formulario para otra entidad destinataria.

- **Corrección de errores y depuración.**

Para la tarea de corrección de errores y depuración, se realizó una revisión por parte del equipo de QA para identificar posibles problemas o fallos en el sistema. Sin embargo, durante este sprint, no se encontraron hallazgos por parte de QA. El sistema se sometió a pruebas, y todos los casos de prueba pasaron satisfactoriamente.