



Universidad **Mariana**

Gestión de la actividad investigativa estudiantil de la Universidad Mariana: Una solución basada en una plataforma web.

Andrés Esteban Rizo Zamora
Jeisson Fernando Caicedo Burbano

Universidad Mariana
Facultad de Ingeniería
Programa de Ingeniería de Sistemas
San Juan de Pasto
2023

Gestión de la actividad investigativa estudiantil de la Universidad Mariana: Una solución basada en una plataforma web.

Andrés Esteban Rizo Zamora
Jeisson Fernando Caicedo Burbano

Trabajo de grado como requisito para obtener el título de: Ingeniero de sistemas

Mg. Fabián Parra Pay
Asesor

Universidad Mariana
Facultad de Ingeniería
Programa de Ingeniería de Sistemas
San Juan de Pasto
2023

Artículo 71: los conceptos, afirmaciones y opiniones emitidos en el Trabajo de Grado son responsabilidad única y exclusiva del (los) Educando (s)

Reglamento de Investigaciones y Publicaciones, 2007

Universidad Mariana

Agradecimientos

Queremos expresar nuestro sincero agradecimiento a los profesores Fabian Parra Pay y Álvaro Alexander Martínez Navarro, por su orientación experta y paciencia infinita a lo largo de este proyecto. Su mentoría ha sido invaluable.

Agradecemos profundamente a nuestras familias por su constante apoyo y aliento. Su amor incondicional ha sido nuestra fuente de fortaleza.

Nuestra gratitud se extiende a la Universidad Mariana por proporcionar los recursos necesarios para llevar a cabo este trabajo.

Finalmente, a todos aquellos que, de una manera u otra, contribuyeron a este proyecto, gracias. Este logro no habría sido posible sin su colaboración.

Con aprecio,

Andrés Esteban Rizo Zamora y Jeisson Fernando Caicedo Burbano

Dedicatoria

Primero, agradezco a Dios por guiarme siempre en el camino de la vida.

A mis queridos padres, Andrés José y María del Pilar, les expreso mi profundo agradecimiento por su apoyo incondicional en todo momento. Aprecio sus valores, consejos y la confianza que siempre depositaron en mí, motivándome a seguir adelante. Gracias a ustedes.

A mis abuelos Gofrey y Amparo, les doy las gracias por toda la ayuda brindada, no solo durante mi formación académica, sino también en la vida misma. Su apoyo ha sido invaluable.

A mis hermanos, Juan Sebastian, Samuel, Laura y Camila, les agradezco por su apoyo incondicional y por ser parte fundamental de mi vida.

A mis familiares, en especial a mis tías Claudia y Sandra, y a mi primo Diego Fernando, agradezco su constante atención y cuidado hacia mí.

A mis maestros, les estoy agradecido por todas sus enseñanzas, las cuales han sido fundamentales para mi buen desempeño en mi vida académica.

Finalmente, agradezco a todos aquellos que han brindado su ayuda y han sido parte en este proyecto de vida.

Andres Esteban Rizo Zamora

A mis pilares fundamentales,

En este emocionante capítulo de mi viaje académico, dedico estas palabras a los seres extraordinarios que han sido mi constante fuente de inspiración y apoyo: mi amada mamá, mi eternamente optimista papá y mis queridos hermanos.

A ti, mamá, cuyo amor incondicional ha sido mi refugio en los días difíciles y mi inspiración en los momentos de éxito. Tus palabras alentadoras y tu sabiduría han sido faros luminosos que iluminan mi camino.

A ti, papá, el eterno portador de sonrisas y curador de cualquier tristeza. Tu alegría contagiosa ha sido mi luz en los días oscuros, recordándome que cada desafío es una oportunidad para aprender y crecer.

Y a mis amados hermanos, cómplices de risas y confidentes en todas las etapas de la vida. Su apoyo constante y complicidad han convertido los desafíos en experiencias compartidas y las victorias en triunfos familiares.

Este trabajo de tesis no solo es un logro académico, sino también el resultado de la red de amor y apoyo que ustedes han tejido a mi alrededor. Cada página lleva impresa la influencia de sus risas, la fuerza de sus abrazos y la alegría de compartir cada paso del camino juntos.

Agradecido infinitamente por su paciencia, por creer en mí cuando dudé y por celebrar cada pequeño logro como si fuera el más grande. Este logro es tan suyo como mío.”

Con amor y gratitud,

Jeisson Fernando Caicedo Burbano

Contenido

Introducción	14
1. Resumen del proyecto.....	16
1.1. Descripción del problema	17
1.1.1. Formulación del problema	19
1.2. Justificación	19
1.3. Objetivos	21
1.3.1. Objetivo general.....	21
1.3.2. Objetivos específicos	21
1.4. Marco referencial	22
1.4.1. Antecedentes	22
1.5. Método De Desarrollo.....	33
1.5.1. Etapas de la metodología XP	33
1.5.2. Artefactos	43
1.5.3. Roles	44
1.6. Línea y Áreas Temáticas.....	44
1.7. Presupuesto	45
1.8. Cronograma.....	46
1.9. Productos esperados.....	48
1.10. Condiciones de entrega	48
2. Presentación de resultados	49
2.1. Modelado de procesos.....	49
2.1.1. Oportunidad de intervención.....	52
2.1.2. Discusión.....	53

2.2. Desarrollo del aplicativo web	53
2.2.1 Etapa 1 – Planificación	53
2.2.2. Etapa 2 – Diseño	55
2.2.3 Etapa 3. Codificación.....	62
2.2.3.1. Backend.....	63
2.2.3.2. FrontEnd	72
2.2.4. Etapa 4. Pruebas.....	76
2.2.5 Construcción y despliegue imágenes docker	81
2.3. Evaluación de la plataforma.....	86
3. Conclusiones	100
4. Recomendaciones	101
Referencias bibliográficas.....	102

Índice de Tablas

Tabla 1	<i>VIPRI: Sistema de información de investigaciones de la Universidad de Nariño.</i>	22
Tabla 2	<i>SINFONI: Sistema de información de investigación de la Universidad Cooperativa de Colombia.</i>	23
Tabla 3	<i>HERMES: Sistema de información de la investigación Universidad Nacional de Colombia.</i>	25
Tabla 4	<i>π: Sistema de información de investigaciones de la Universidad EAFIT.</i>	27
Tabla 5	<i>FACULTYUP: Sistema de información de investigaciones de la Universidad del Pacífico.</i>	29
Tabla 6	<i>Plantilla para el diseño de Historias de Usuario</i>	33
Tabla 7	<i>Tecnologías utilizadas para el desarrollo de la plataforma.</i>	39
Tabla 8	<i>Artefacto de identificación de un escenario de prueba</i>	41
Tabla 9	<i>Artefacto para especificar un caso de prueba</i>	42
Tabla 10	<i>Presupuesto global del proyecto.</i>	45
Tabla 11	<i>Descripción de la Inversión en personal.</i>	45
Tabla 12	<i>Otros rubros.</i>	45

Índice de Figuras

Figura 1. <i>Búsqueda de proyectos estudiantiles en la plataforma VIPRI.</i>	23
Figura 2. <i>Búsqueda de proyectos estudiantiles en la plataforma SINFONI.</i>	25
Figura 3. <i>Página inicial de la plataforma HERMES.</i>	27
Figura 4. <i>Búsqueda de publicaciones en la plataforma π.</i>	29
Figura 5. <i>Buscar actividades de investigación en el sistema FACULTYUP.</i>	30
Figura 6. <i>Arquitectura monolítica de tres capas.</i>	35
Figura 7. <i>Patrón de descomposición strangler: Fases de migración.</i>	36
Figura 8. <i>API Gateway como único punto de enlace entre peticiones y microservicios.</i>	36
Figura 9. <i>Comunicación síncrona y asíncrona entre microservicios.</i>	38
Figura 10. <i>Despliegue de contenedores en el servicio ECS.</i>	43
Figura 11. <i>Modelado de procesos de la gestión de la actividad investigativa estudiantil de la Universidad Mariana - Primera fase vinculación de estudiantes</i>	50
Figura 12. <i>Modelado de procesos de la gestión de la actividad investigativa estudiantil de la Universidad Mariana - Segunda Fase Gestión de la información</i>	51
Figura 13. <i>Plan de entregas</i>	54
Figura 14. <i>Plan de iteración</i>	55
Figura 15. <i>Estructura del proyecto</i>	56
Figura 16. <i>Estructura BackEnd</i>	57
Figura 17. <i>Estructura de GraphQL</i>	58
Figura 18. <i>Estructura FrontEnd</i>	59
Figura 19. <i>Modelo ER</i>	61
Figura 20. <i>Modelo lógico</i>	62
Figura 21. <i>Entidades de dominio</i>	63
Figura 22. <i>Repositorios</i>	64
Figura 23. <i>Servicios para cada entidad</i>	65

Figura 24. Controladores.....	66
Figura 25. Controlador GraphQL para proyectos	67
Figura 26. Controlador SpringBoot para proyectos.....	67
Figura 27. Microservicios y Gateway	68
Figura 28. Enrutamiento microservicios desde el Gateway	70
Figura 29. Recurso de la plataforma en Google Cloud.....	71
Figura 30. ID Cliente OAuth.....	71
Figura 31. Controlador útil para la validación del token.....	72
Figura 32. Interfaces de entidades del BackEnd.....	73
Figura 33. Servicios Apollo y HTTP para comunicación con el BackEnd	74
Figura 34. Mutaciones y consultas de GraphQL	75
Figura 35. Operaciones GraphQL para cada entidad.....	75
Figura 36. Creación de componente para gestión de entidades	76
Figura 37. Resultados pruebas E2E en entorno de testing Cypress	79
Figura 38. Archivo para despliegue del microservicio a una imagen Docker	81
Figura 39. Archivo para el despliegue del frontend a una imagen Docker.....	82
Figura 40. Repositorios de Docker Hub	83
Figura 41. Cluster ECS	83
Figura 42. Configuración contenedores en tareas.....	84
Figura 43. Tareas para cada imagen Docker	84
Figura 44. Lanzamiento de servicios	85
Figura 45. IP pública del servicio FrontEnd	85
Figura 46. Arquitectura de la plataforma	86
Figura 47. Representación gráfica de la primera evaluación	88
Figura 48. Representación gráfica de la segunda evaluación.....	89

Figura 49. <i>Representación gráfica de la tercera evaluación</i>	89
Figura 50. <i>Representación gráfica de la cuarta evaluación</i>	90
Figura 51. <i>Representación gráfica de la quinta evaluación</i>	91
Figura 52. <i>Representación gráfica de la sexta evaluación</i>	92
Figura 53. <i>Representación gráfica de la séptima evaluación</i>	93
Figura 54. <i>Representación gráfica de la octava evaluación</i>	94
Figura 55. <i>Representación gráfica de la novena evaluación</i>	95
Figura 56. <i>Representación gráfica de la décima evaluación</i>	96
Figura 57. <i>Representación gráfica de la undécima evaluación</i>	97
Figura 58. <i>Representación gráfica de la Duodécima evaluación</i>	98

Índice De Anexos

Anexo A. Carta de justificación del desarrollo tecnológico	105
Anexo B. Formato de entrevista semi-estructurada	106
Anexo C. Aportes de la entrevista	107
Anexo D. Historias de usuario	111
Anexo E. Mockups pantallas	125
Anexo F. Escenarios de prueba.....	130
Anexo G. Casos de prueba.....	135
Anexo H. Resultados pruebas unitarias	149
Anexo I. Escenarios flujos de pruebas E2E	151
Anexo J. Plataforma web SIUMAR.....	154

Introducción

En la era de la información, la gestión eficiente de los datos se ha convertido en un desafío crucial para las instituciones educativas. La Coordinación de Investigación Estudiantil de la Universidad Mariana no es una excepción. Con el vertiginoso aumento de la actividad investigativa, el método actual de gestión de datos, basado en herramientas tradicionales como Excel, ha demostrado ser insuficiente para manejar la creciente carga de información generada por los estudiantes investigadores. Esta ineficiencia afecta la toma de decisiones y puede impactar negativamente en la reputación de la universidad.

Ante este escenario, surge la necesidad de una solución integral que supere las limitaciones tecnológicas actuales y establezca bases sólidas para la gestión de la actividad investigativa estudiantil. Este trabajo propone la implementación de una arquitectura de microservicios y la utilización de Docker, tecnologías innovadoras que están estableciendo un nuevo estándar para la gestión de información en entornos académicos.

Los microservicios son pequeñas unidades funcionales independientes que trabajan juntas para ofrecer la funcionalidad general de una aplicación. Esta arquitectura posibilita que cada servicio se actualice o escale de forma independiente, lo que resulta en una mayor eficiencia y flexibilidad. Docker, por otro lado, es una plataforma que permite contenerizar aplicaciones, facilitando su despliegue y escalabilidad. Esta combinación de tecnologías permite añadir funcionalidades según las necesidades que surgen para enfrentar los retos de la actualidad, lo que resulta en productos factibles a su evolución y mantenibilidad.

La adopción de estas tecnologías no solo aborda los desafíos inmediatos de la Universidad Mariana, sino que también establece un precedente para otras instituciones educativas. Este proyecto se presenta como un modelo a seguir para aquellos que buscan un futuro de gestión de datos más eficiente y sostenible.

Además, este proyecto no solo se limita a proponer una solución, sino que también se embarca en un viaje de descubrimiento y aprendizaje. A través de la implementación de la arquitectura de

microservicios y Docker, se exploran nuevas formas de gestionar los datos, descubrir nuevas posibilidades y aprender valiosas lecciones que pueden ser útiles para otros estudios en situaciones similares.

1. Resumen del proyecto

Este proyecto aborda la problemática de la gestión de datos en la Coordinación de Investigación Estudiantil de la Universidad Mariana. Se destaca la ineficiencia del sistema actual basado en Excel para manejar el creciente volumen de datos generados por los estudiantes investigadores, y se señalan las consecuencias que esto tiene en la toma de decisiones y la calidad de la información.

Como solución, se desarrolló una innovadora plataforma web basada en microservicios y contenedores Docker, respaldada por la metodología ágil eXtreme Programming (XP). Esta metodología permitió guiar el desarrollo desde la planificación inicial de historias de usuario hasta la implementación y pruebas finales. Inicialmente, se realizó un modelado de procesos mediante el uso del software BPMN, lo que permitió modelar el proceso actual de gestión de investigación de la Universidad Mariana. Este modelado permitió identificar los puntos débiles del sistema actual y la oportunidad de intervención a través del desarrollo de un aplicativo web. Posteriormente, se describe detalladamente el proceso de codificación desde la creación y migración de un aplicativo inicial con una arquitectura monolítica a microservicios mediante el patrón Strangler, destacando la importancia del modelo de base de datos por microservicio. Además, se especifican las tecnologías utilizadas, desde HTML y CSS hasta Java, Spring y Docker, subrayando su relevancia para la eficacia y eficiencia del nuevo sistema.

Las pruebas, tanto automatizadas como con usuarios finales, respaldan el éxito de la plataforma. Estas pruebas culminaron en un plan de despliegue en un entorno de pruebas en AWS utilizando servicios como ECS y EC2. Se realizaron pruebas con usuarios finales en este entorno, donde se evaluó la usabilidad y funcionalidad del aplicativo.

Las respuestas de los usuarios finales sirvieron para extraer conclusiones valiosas de cada sección de la evaluación. Estas conclusiones permitieron una introspección para la mejora continua del software. De esta manera, la implementación de la plataforma permitió mantener la alineación con las necesidades y dificultades de los Coordinadores de Investigación de la Universidad Mariana en su proceso de gestión de información.

1.1. Descripción del problema

En este capítulo se describirá el problema a la luz de cuatro elementos: síntomas, que hacen referencia a los problemas que actualmente se están presentando en el contexto de la temática a investigar; causas, entendidas como las razones por las que se producen los síntomas; diagnóstico, enunciado que expresa el problema específico que se abordará; y pronóstico, que enuncia los problemas futuros que pueden presentarse de continuar presentándose los síntomas, causas y diagnóstico.

La Coordinación de Investigación Estudiantil de la Universidad Mariana, en el análisis de sus mecanismos de gestión de datos, ha manifestado diferentes dificultades para un estudio óptimo del historial de actividades investigativas de sus estudiantes, esto debido al proceso de administración de la información obtenida. La gestión de datos se define como "el proceso de ingerir, almacenar, organizar, utilizar y mantener los datos creados y recopilados por una organización" (Stedman & Vaughan, 2021). De acuerdo con la anterior definición, entre los inconvenientes del procedimiento actual, se encuentra la cantidad de esfuerzo y tiempo perdido que ha dispuesto el personal encargado de digitalizar manualmente esta información, por ende, las generaciones tardías de estos informes han conducido a una toma de decisiones lenta y deficiente. Por otro lado, las ubicaciones de almacenamiento de datos y su estado de protección no han sido totalmente claros, además, las herramientas de recuperación a veces fallan al intentar reparar libros de trabajo dañados mientras que los archivos recuperados no siempre han incluido las últimas actualizaciones, así mismo, el intercambio de información con un sistema de datos no centralizados ha producido inconsistencias en los reajustes y actualizaciones de los archivos de registro.

La razón principal por la que el proceso de gestión de datos actual ya no se considera ideal, se debe al creciente recibimiento de un gran número de datos por cada estudiante investigador perteneciente a la Universidad Mariana desde los diferentes programas académicos que ofrece la institución. No sólo en el campus ha ocurrido este crecimiento, en realidad, para las fechas comprendidas entre los años 2017 a 2019 se dio un incremento masivo en cuanto al registro de grupos investigativos en Colombia, pasando de haber 5.207 en 2017, a tener 762.655 en 2019 (Ministerio de Ciencia y Tecnología e Innovación, s.f., 2021a, 2021b), con ello, se puede deducir

que estas estadísticas concluyen un aumento en las investigaciones no sólo a nivel de la Universidad Mariana sino a nivel nacional. Debido a este auge investigativo, la utilización de herramientas poco eficientes y limitadas como Excel, al ser el sistema de gestión actual en la institución educativa aludida, resulta ser un método escasamente sostenible, este hecho se refleja cuando, por ejemplo, en el año 2020 el uso inadecuado de este software fue la razón por la que casi 16.000 casos positivos de coronavirus no se informaron en Inglaterra, esto al no tener espacio en su hoja de Excel (Ventura, 2020). Por lo anterior, además de que Excel representa un gran problema en la escalabilidad de los datos, ocasiona una mala calidad de estos mismos y con ello produce posibles decisiones erróneas; lo anterior es corroborado por Contin (2021) cuando afirmó que, si los datos de una organización son incorrectos, puede generar un impacto negativo en su desempeño. En pocas palabras, los riesgos y pérdidas potenciales, esperados o imprevistos, asociados con la mala calidad de los datos pueden afectar de manera exponencial las decisiones de la empresa. Teniendo en cuenta esto, el limitado sistema actual para la gestión de información de las actividades investigativas de los estudiantes de la Universidad Mariana no es suficientemente efectivo ni eficiente.

De continuar con esta tendencia, el impacto positivo de las decisiones de investigación estudiantil mermaría exponencialmente y a su vez incrementaría la pérdida de credibilidad de sus reportes registrados. Figueroa y Lemoine (2018), por otra parte, afirman que "si se observa la jerarquía de las universidades de mayor reconocimiento a nivel mundial y regional se comprobará que las mejor posicionadas exhiben resultados significativos en la investigación y específicamente en la gestión de proyectos de investigación" (parr.4), por lo tanto, el no ofrecer un cambio significativo con el proceso de gestión de datos actual generaría la desventaja de la Universidad a nivel de reconocimiento frente a otras instituciones de educación superior, de modo que, pondría en duda el compromiso de la Universidad con el logro efectivo de su visión; pactada en el Plan de Desarrollo Institucional 2021-2028, donde estipula que: "La Universidad Mariana en 2028 será sostenible, innovadora y referente local, por la perspectiva global en la formación, investigación e innovación social y la adopción de las mejores prácticas de gobernanza en respuesta a las demandas de los grupos de interés en la región y el país" (Universidad Mariana, 2020, p. 34).

1.1.1. Formulación del problema

¿Cómo apoyar el proceso de gestión de la actividad investigativa estudiantil de la Universidad Mariana?

1.2. Justificación

La relevancia de este proyecto se presenta a continuación en términos de utilidad, que incluye los beneficios del desarrollo tecnológico y sus beneficiarios; novedad, que detalla lo innovador del proyecto; e interés que describe aspectos que cautivarán la atención de quien consulte el trabajo.

Los beneficiarios, inicialmente, serán los encargados de digitalizar los datos porque el sistema tendrá validaciones de campo que permitirá verificar que la información ingresada sea más sensata y razonable, además, con la creación de tablas relacionales junto a una interfaz cómoda y listados concisos, posibilitará una mejor entrada de datos, de esta manera, disminuye la probabilidad de que incorporen datos inexactos y el tiempo de registro sea alto. El personal dedicado al análisis de datos también se beneficiará con el proyecto porque va a permitir la producción de respuestas rápidas a consultas como: ¿cuál fue el volumen en actividades investigativas de los estudiantes por facultad durante los últimos seis meses?, por lo tanto, el acceso mejorado de los datos permitirá generar información con mayor eficiencia, obteniendo así capacidades de toma de decisiones más rápidas y nuevas estrategias que atiendan las necesidades del entorno. Finalmente, la Universidad tendrá beneficios porque la productividad avanzada en los datos e informes generados en conjunto a las habilidades de investigación de sus estudiantes podría permitir el posicionamiento superior de la labor investigativa de la institución en la perspectiva regional, nacional e internacional.

Evidentemente, hoy en día, las aplicaciones necesitan ser diseñadas de modo que puedan tolerar las fallas de los distintos servicios. A medida que las aplicaciones obtienen cada vez más funciones, la base de código subyacente aumenta en extensión y, por lo tanto, es más difícil de mantener y administrar. Para ello, este proyecto adopta una arquitectura de microservicios y la implementación de la plataforma de contenedorización Docker, que implica una gran novedad para el proceso de gestión de información del historial de actividades investigativas de los estudiantes de la Universidad Mariana, debido a que, es un proceso reciente cuyas características tecnológicas

son significativamente diferentes a las anteriores adquiridas por la institución en cuestión. Su importancia radica en que un microservicio es una unidad relativamente pequeña que interactúa con otras unidades sólo a través de interfaces de programación de aplicaciones (API) (Amazon Web Services, 2022) y Docker "encapsula el microservicio en un contenedor que luego se puede mantener e implementar de forma independiente" (Syed & Rao, s.f.). Por lo anterior, cada uno de estos contenedores será responsable de una funcionalidad específica. Las posibilidades de efectos secundarios inesperados de un cambio son menores y más fáciles de encontrar y corregir si ocurren. Su mayor clave es la separación de la interfaz de usuario de la funcionalidad. Por ejemplo, la forma en que una aplicación interactúa con los usuarios es un asunto independiente de cómo procesa sus datos y, a menudo, puede ser necesario cambiar uno sin afectar al otro. A veces, el objetivo es cambiar el aspecto de un sitio web sin modificar sus capacidades básicas. Y a veces es para reorganizar la base de datos sin afectar la experiencia del usuario. Los microservicios y Docker manejan las operaciones de datos sin importar cómo el usuario vea la información. Esto significa que los desarrolladores pueden cambiar diferentes partes del software, incluso al mismo tiempo, sin preocuparse por lo que están haciendo los demás equipos. Mientras todos usen la API correctamente, seguirá funcionando. Por lo tanto, es necesario implementar la arquitectura de microservicios conjunto a la plataforma Docker, con el fin de garantizar aportes en avances tecnológicos y computacionales que respondan a las exigencias del nuevo milenio, además, se convierte a la vez en una metodología atractiva y valiosa para las instituciones universitarias que manejan un sistema de información para la investigación estudiantil, porque a nivel internacional no existen suficientes plataformas que apliquen este tipo de arquitectura para la integración de gestión de información y control de actividades investigativas institucionales.

Teniendo en cuenta los objetivos específicos del proyecto, describir la oportunidad de intervención en la gestión de la actividad investigativa estudiantil de la Universidad Mariana mediante el modelado de procesos es útil e interesante, porque la institución se va dar cuenta de cuáles son los puntos culminantes que producen el retardamiento de las actividades en el proceso de administración de los datos con ayuda de un diagrama, que al ser información visual hace que sea más fácil de identificar que un texto, de esta manera, el instituto universitario sabrá dónde está el problema de manera exacta y qué decisiones debe tomar para resolverlo. Así mismo, el desarrollo de una plataforma web que permita la gestión de la actividad investigativa estudiantil

de la Universidad es valioso porque todos los beneficiarios van a notar hasta dónde pudo llegar la solución al resolver esas necesidades anteriormente identificadas, además, es interesante e innovador por el tipo de arquitectura tecnológica del sistema interno que estará particionado en microservicios y soportado por la tecnología de contenedores de Docker, lo cual no es común. Finalmente, evaluar la plataforma web desplegando los microservicios empaquetados en contenedores mediante la utilización de servicios en la nube con tecnologías tan importantes como Amazon Web Services puede servir como referente tecnológico y modelo para otras investigaciones que quieran hacer lo mismo.

1.3. Objetivos

1.3.1. Objetivo general

Construir una estrategia tecnológica para apoyar el proceso de gestión de la actividad investigativa estudiantil de la Universidad Mariana.

1.3.2. Objetivos específicos

- Describir la oportunidad de intervención en la gestión de la actividad investigativa estudiantil de la Universidad Mariana a través del modelado de procesos.
- Desarrollar una plataforma web mediante la implementación de una arquitectura de microservicios y contenedores Docker que permita la gestión de la actividad investigativa estudiantil de la Universidad Mariana.
- Evaluar la plataforma web construida para la gestión de la actividad investigativa estudiantil de la Universidad Mariana en un entorno de pruebas.

1.4. Marco referencial

1.4.1. Antecedentes

Tabla 1

VIPRI: Sistema de información de investigaciones de la Universidad de Nariño.

Objetivo	Recopilar y gestionar la información correspondiente a la investigación de estudiantes y docentes de la Universidad de Nariño.
Tiempo de existencia	Desde el año 2005.
Frecuencia de actualización	No presenta información.
Funcionalidad	La plataforma permite consultar proyectos de docentes, estudiantes, trabajos de grado o investigaciones, la búsqueda se puede realizar filtrando parámetros; se obtiene una lista de los documentos con su respectiva información (título, objetivo, valor aprobado, duración, estado). Los usuarios registrados pueden gestionar la información de sus proyectos, como el registro, evaluación, revisión de la propuesta, solicitudes de costes, compromisos, informes financieros, informes generales y el registro del estado de los investigadores.
Atributos de calidad	<p>Usabilidad: Plataforma poco intuitiva, cuenta con un sistema de menús confuso y un diseño poco atractivo.</p> <p>Funcionalidad: La plataforma cumple con sus propósitos al gestionar la información de los proyectos investigativos de estudiantes y docentes.</p> <p>Seguridad: No utiliza el protocolo HTTPS por lo tanto no hay seguridad en la transmisión de datos por otro lado protege la información mediante autenticación.</p> <p>Eficiencia: Optimiza todos los procesos manuales y automatiza la gestión de los datos.</p> <p>Mantenibilidad: Sistema monolítico, el mantenimiento de un servicio puede afectar todo el software.</p> <p>Portabilidad: Cuenta con un espacio en la web y un diseño responsivo por lo cual es accesible en todo dispositivo tecnológico.</p>
Licenciamiento	Software desarrollado internamente desde la universidad, uso exclusivo para la Universidad de Nariño.

Figura 1

Búsqueda de proyectos estudiantiles en la plataforma VIPRI.

SISTEMA DE CONSULTA DE PROYECTOS DE TRABAJO DE GRADO

Los proyectos de investigación se definen como la principal actividad de los Grupos de Investigación. Es su desarrollo lo que mantiene la constante producción de conocimiento científico, y una dinámica de injerencia real y de alto impacto para la región.

Así, Colociencias, y otras instituciones financiadoras, asumen los proyectos de investigación como marco de referencia para reconocer y medir la capacidad de trabajo de los investigadores y grupos de investigación, tomando como fundamento la producción de resultados tangibles y verificables.

Buscar por

No.	Código	Título	Objetivo General	Investigador Principal	Facultad	Departamento	Convocatoria	Tipo financiación	Valor aprobado	Estado	Ver
1	2331	ESENCIALIZACIÓN DE LA PRÁCTICA CONTROL DE CAMBIOS DE SOFTWARE DE RUP UTILIZANDO EL MODELO PARA LA DEFINICIÓN DE PRÁCTICAS EN INGENIERÍA DE SOFTWARE	Definir la práctica de Control de Cambios de Software de RUP aplicando el Modelo para la Definición de Prácticas en Ingeniería de Software de tal manera que se facilite entender, aplicar y evaluar la práctica.	Jairo Arevalo Acosta	Ingeniería	Sistemas	Trabajos de Grado 2020	Recursos Propios	8.500.000	En Ejecución	
2	2340	Estrategia lúdica computacional grupal para la estimulación de la memoria en adultos mayores	Contribuir a la estimulación de la memoria del adulto mayor, a través de una estrategia lúdica computacional.	CARLOS ANDRES GUZMAN NARVAEZ	Ingeniería	Sistemas	Trabajos de Grado 2020	Recursos Propios	8.700.000	Vencido	
3	1885	MOTOR DE BÚSQUEDA INTELIGENTE DE INFORMES DE INVESTIGACIÓN BASADO EN RECURSOS SEMÁNTICOS PARA EL SISTEMA DE INVESTIGACIONES DE LA UNIVERSIDAD DE NARIÑO	Soportar la búsqueda y recuperación inteligente de los documentos de los informes de investigación que se encuentran en formato digital en el Sistema de Investigaciones de la Universidad de Nariño, a través de la construcción de un motor de búsqueda inteligente, haciendo uso de recursos semánticos mediante el gestor de base de datos SPARQL, y el lenguaje RDF con el manejo de ontologías, con el fin de facilitar las consultas sobre proyectos de investigación realizados en la Universidad de Nariño y las posibles relaciones con otros proyectos de	FELIPE CUJAR ROSERO	Ingeniería	Sistemas	Trabajos de Grado 2019	Recursos Propios	8.281.000	Terminado	

Fuente: Adaptado de *Consulta de proyectos de trabajo de grado*, por Universidad de Nariño, Sistema de Información de Investigaciones VIPRI, (<https://bit.ly/3INmlu>).

Tabla 2

SINFONI: Sistema de información de investigación de la Universidad Cooperativa de Colombia.

Objetivo	Gestionar y visibilizar la investigación institucional, sus investigadores, los proyectos y resultados a través del trabajo articulado y la comunicación científica para el desarrollo de productos de alta calidad.
Tiempo existencia	Desde el año 2017.
Frecuencia de actualización	No presenta información.
Funcionalidad	El sistema permite inscribir, administrar y hacer seguimiento a convocatorias y proyectos de investigación desde cualquier lugar. SINFONI permite agilizar la revisión de los requisitos mínimos de un proyecto y gestionar la ejecución del mismo haciendo seguimiento al cronograma, a las tareas administrativas y a los gastos asociados cuando el proyecto de investigación está en desarrollo. SINFONI incluye alertas para avisar al investigador cuando debe realizar una actividad de carácter científico, administrativo o

	<p>presupuestal promoviendo el cumplimiento oportuno y evitando retrasos en los proyectos. El sistema de información SINFONI permite identificar los lugares donde se desarrollan las investigaciones y conocer el número de personas vinculadas a proyectos, así como, sus horas de dedicación y sus perfiles, también los recursos invertidos por parte de la universidad y por financiadores externos.</p>
Atributos de calidad	<p>Usabilidad: Plataforma muy intuitiva, con los servicios claros e instrucciones de uso en cada funcionalidad.</p> <p>Funcionalidad: Los servicios implementados son apropiados para la gestión de los proyectos de la universidad, por ende, cumple con los objetivos necesarios para una correcta utilización.</p> <p>Seguridad: La plataforma utiliza el protocolo HTTPS por lo tanto hay seguridad en la trasmisión de datos, por otro lado, protege la información mediante autenticación vía Outlook.</p> <p>Eficiencia: Optimiza todos los procesos manuales y automatiza la gestión de los datos.</p> <p>Mantenibilidad: Sistema monolítico, el mantenimiento de un servicio puede afectar todo el software.</p> <p>Portabilidad: Plataforma web con un diseño responsivo por ende es utilizable en cualquier dispositivo.</p> <p>La plataforma cuenta con certificaciones de calidad del software: ISO 9001, ISO 27001, UNE 166.002.</p>
Licenciamiento	<p>El sistema está soportado por la plataforma Fundanet con una licencia de pago para su uso.</p>

Figura 2

Búsqueda de proyectos estudiantiles en la plataforma SINFONI.

The screenshot shows the SINFONI web platform interface. At the top, there is a navigation menu with options like 'Recursos Humanos', 'Viajes', and 'Proyectos'. Below the menu, there is a search bar with the text 'MIS PROYECTOS' and a dropdown menu for 'ESTADO'. To the right of the search bar, there are buttons for 'Documentación', 'Exportar', and 'Mostrar Filtros'. A notification banner at the top right states 'Los importes aparecen en EUROS (EUR)'. Below the search bar, there is a table with the following columns: 'Código', 'Título', 'F.Comienzo', and 'F.Finalización'. The table contains several rows of project data, including codes like '04200017', '04200021', and titles like 'prueba traspaso 2', 'prueba carga de modelo de gastos', and 'Ensayo de investigación clínica'.

Código	Título	F.Comienzo	F.Finalización
04200017	prueba traspaso 2	01/01/2010	01/10/2010
04200021	prueba carga de modelo de gastos	01/01/2009	31/12/2009
002-2010-1	Ensayo de investigación clínica.	05/08/2010	
005-2010-2		05/08/2010	
007-2010-1	Título Nuevo Abreviado	03/09/2010	
007-2010-2	Otro de Gabriel		
04200049	prueba carga modelo y proyecto	01/01/2010	31/12/2010
ZZZ00026	Subvención ISCI 2010 Concesión Anual	01/01/2010	31/12/2010
2010-002-2	codigo bla bla bla bla bla bla bla bla bla bla bla b	21/10/2010	
2011-011-1	oprueba borrar		
2011-001-2	Estudio Observacional One		
2011-043-1	Prueba empresa 2		
2165165216			

Fuente: Tomando de *Guía Proyectos Sínfoni* (p.21), por Universidad Cooperativa de Colombia, Sistema de Información para la Investigación – Sínfoni, (bit.ly/36lrrA2).

Tabla 3

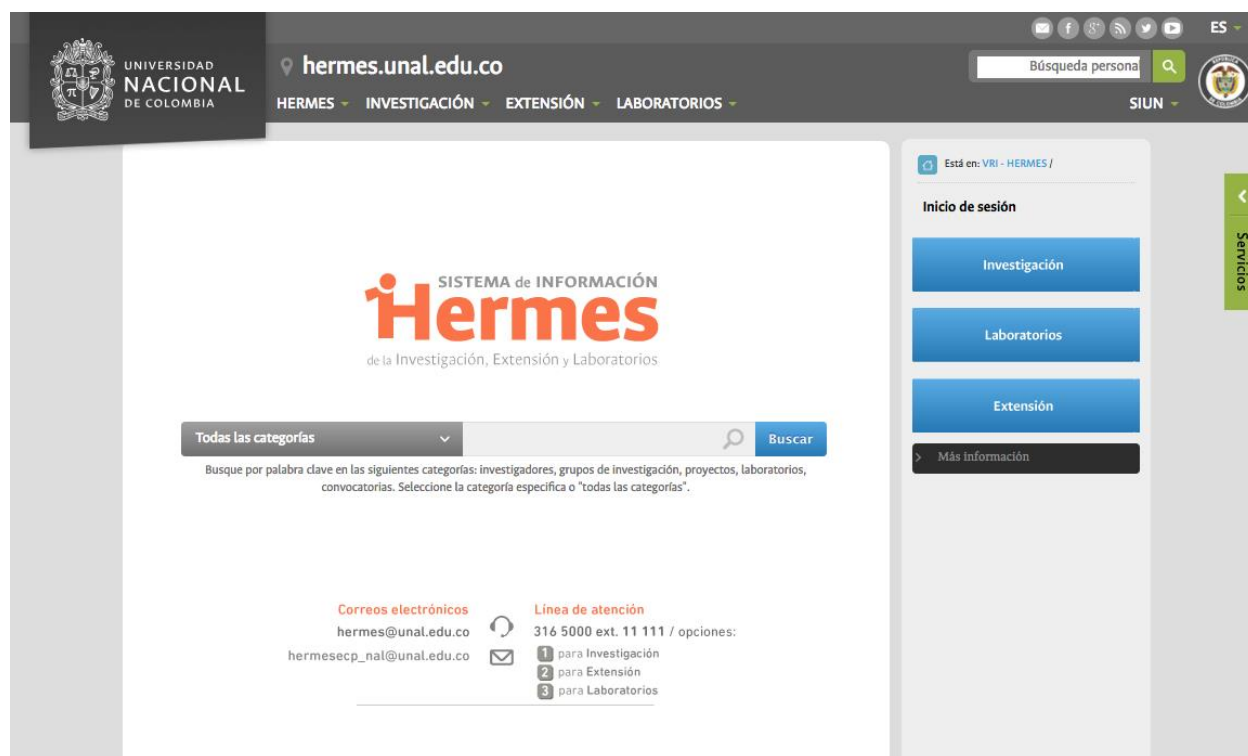
HERMES: Sistema de información de la investigación Universidad Nacional de Colombia.

Objetivo	Recopilar los datos relacionados con las actividades de investigación, laboratorios y extensión desarrolladas en la Universidad Nacional de Colombia.
Tiempo de existencia	Creado en el año 2005 y adoptado en el 2007.
Frecuencia de actualización	Sólo se ha registrado una actualización en el año 2014.
Funcionalidad	La plataforma permite a usuarios registrados y no registrados, consultar información acerca de los estados de los proyectos de Investigación, Laboratorios y Extensión. Para usuarios registrados el sistema permite según su rol: registrar convocatorias internas, registrar proyectos, solicitar avales, solicitar autorizaciones de biodiversidad, solicitar certificaciones, solicitar el registro de una propiedad intelectual, consultar documentos de nivel formativo como agendas de conocimiento o planes de política pública, consultar movilidades por internacionalización, permite registrar toda la información asociada a la hoja de vida del investigador, permite visualizar

	los grupos a los cuales el usuario está asociado, consultar información asociada al laboratorio al cual está inscrito, entre otros.
Atributos de calidad	<p>Usabilidad: El usuario puede navegar por el sitio fácilmente, plataforma intuitiva con un diseño básico.</p> <p>Funcionalidad: La plataforma permite a administrativos, docentes y estudiantes registrados consultar proyectos y movilidades, generar reportes de proyectos, de actividades de investigación, generar indicadores gráficos de convocatorias, avales, grupos de investigación y proyectos en general.</p> <p>Seguridad: No utiliza el protocolo HTTPS por lo tanto no hay seguridad en la transmisión de datos por otro lado protege la información mediante autenticación.</p> <p>Eficiencia: El proceso de acceso a la información que los usuarios desean es fácil, rápido y fluido, guiándolos sin problemas desde el ingreso a su sitio hasta su objetivo.</p> <p>Mantenibilidad: Sistema monolítico, el mantenimiento de un servicio puede afectar todo el software.</p> <p>Portabilidad: Cuenta con un espacio en la web y un diseño responsivo por lo cual es accesible en todo dispositivo tecnológico.</p>
Licenciamiento	Software desarrollado internamente desde la Universidad, uso exclusivo para la Universidad Nacional de Colombia.

Figura 3

Página inicial de la plataforma HERMES.



Fuente: Tomado de *Sistema de Información*, por Universidad Nacional de Colombia, Vicerrectoría de Investigación - HERMES, (<https://bit.ly/3NrwtfL>).

Tabla 4

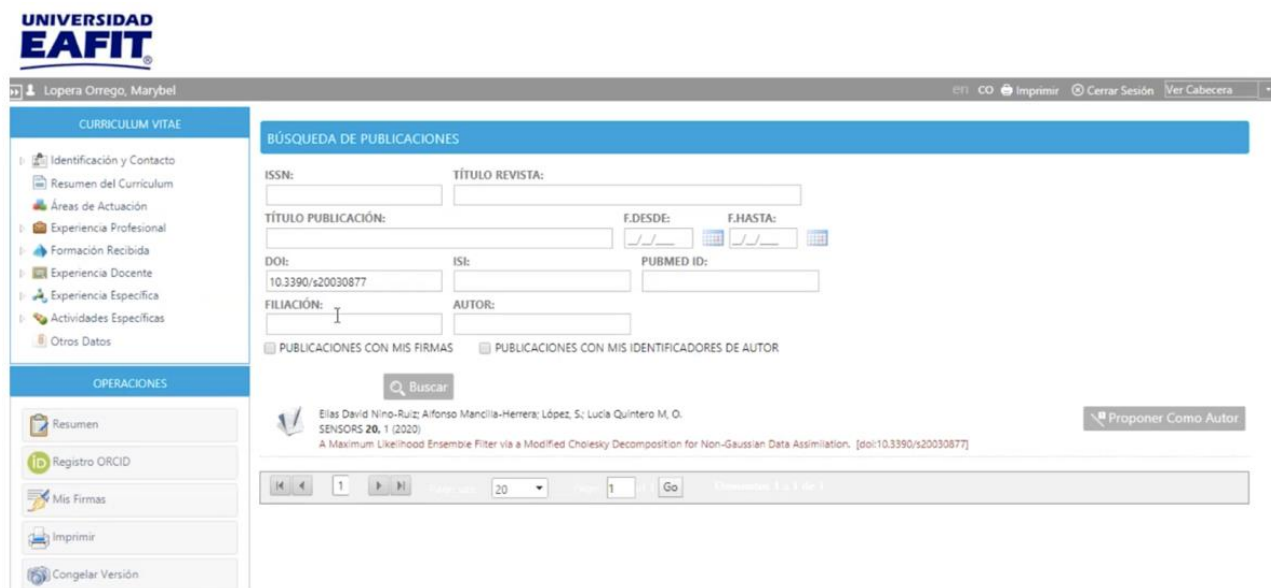
π: Sistema de información de investigaciones de la Universidad EAFIT.

Objetivo	Apoyar los procesos de gestión y divulgación de la investigación de la Universidad EAFIT.
Tiempo de existencia	Desde el año 2016.
Frecuencia de actualización	No presenta información.
Funcionalidad	Permite a los investigadores consultar, generar reportes y administrar la información relacionada con su producción académica y creativa. Pi consolida los datos más importantes de los currículos de los investigadores

	<p>como experiencia profesional, afiliación a grupos de investigación, producción científica y proyectos que pueden apoyar. Además, aporta a la certificación de la experiencia de los grupos de investigación para procesos de convocatorias, consultorías y proyectos cofinanciados.</p>
Atributos de calidad	<p>Usabilidad: La plataforma web está claramente etiquetada para el tipo de contenido que contiene, logrando que los usuarios puedan encontrar rápidamente la información que necesitan.</p> <p>Funcionalidad: El sistema permite consultar la producción científica y las capacidades investigativas de la Universidad, además de gestionar toda la información propia de los investigadores.</p> <p>Seguridad: Utiliza el protocolo HTTPS por lo tanto hay seguridad en la transmisión de información además protege los datos mediante autenticación.</p> <p>Eficiencia: La plataforma es coherente, compacta, secuencial y lógica. La búsqueda de producción académica de investigación se puede realizar filtrando parámetros, se obtiene la producción académica con su respectiva información (título, tipo de unidad, unidad, autores, anualidad, palabras clave, fuente, tipo de documento).</p> <p>Mantenibilidad: Sistema monolítico, el mantenimiento de un servicio puede afectar todo el software.</p> <p>Portabilidad: El sitio web trabaja con un diseño responsive por ende es capaz de adaptarse visualmente a cualquier dispositivo.</p>
Licenciamiento	<p>Software desarrollado internamente desde la Universidad, uso exclusivo para la Universidad EAFIT.</p>

Figura 4

Búsqueda de publicaciones en la plataforma π .



Fuente: Tomado de *Aprende a gestionar la Plataforma Pi*, por Universidad EAFIT, Sistema de información de investigaciones de la Universidad EAFIT , (<https://bit.ly/3uzKHm8>).

Tabla 5

FACULTYUP: Sistema de información de investigaciones de la Universidad del Pacífico.

Objetivo	Promover y organizar la información y conocimientos académicos e investigativos de la Universidad Del Pacifico, así como una intercomunicación e interacción entre sus investigadores y docentes, y sus pares nacionales e internacionales.
Tiempo de existencia	Desde el año 2021.
Frecuencia de actualización	No presenta información.
Funcionalidad	El sistema de información permite consultar perfiles de investigadores, visualizar redes de colaboración y los temas especializados en investigación de la institución. Además, para usuarios internos de la institución se puede publicar resultados de investigación y gestionar proyectos y fuentes de financiamiento.

<p>Atributos de calidad</p>	<p>Usabilidad: El sistema es práctico, a pesar de ello cuenta con una gran cantidad de servicios que pueden resultar confusos.</p> <p>Funcionalidad: Cumple con las necesidades requeridas para la investigación de la universidad, además tiene gráficos que aportan a la visualización de información.</p> <p>Seguridad: Cuenta con certificación ISO 27001.</p> <p>Eficiencia: Optimiza todos los procesos manuales y automatiza la gestión de los datos.</p> <p>Mantenibilidad: Sistema monolítico, el mantenimiento de un servicio puede afectar todo el software.</p> <p>Portabilidad: Cuenta con un espacio en la web y un diseño responsivo por lo cual es accesible en todo dispositivo tecnológico.</p>
<p>Licenciamiento</p>	<p>El sistema está soportado por la plataforma Elsevier con una licencia de pago para su uso.</p>

Figura 5

Buscar actividades de investigación en el sistema FACULTYUP.

The screenshot displays the search results for 'Encontrar actividades de investigación' on the FACULTYUP platform. The search bar is at the top, and the results are listed below. The first result is 'Anuario de Historia Regional y de las Fronteras (Revista)' by Beatriz Rodríguez Sotizabal, dated ene. 2022. The second result is 'BMC Public Health (Revista)' by Mario Gustavo Chong Chong & Manuel Fernando Barrón Aylón, dated 9 ene. 2022. The third result is 'SOMOS UP - Febrero' by a group of authors, dated 15 feb. 2022 - 17 feb. 2022. The interface also includes a sidebar with filters for 'Filtros para Actividades' and 'Año'.

Fuente: Adaptado de *Encontrar actividades de investigación*, por Universidad del Pacifico, Portal de Investigación FacultyUP, (<https://bit.ly/3uEnI9f>).

Las tecnologías analizadas en relación con el desarrollo del proyecto tienen como objetivo aportar inicialmente funcionalidades tales como recopilación, gestión, visualización y análisis de la información investigativa, además la generación de reportes estadísticos presentados gráficamente para una mejor visualización de la información; por otro lado adoptar tecnologías prácticas y esenciales para el desarrollo web, en concreto lenguaje de etiquetas de hipertexto (HTML), hojas de estilo en cascada (CSS), sistema gestor de base de datos (SGBD) y la utilización de interfaces de programación de aplicaciones (API's); finalmente las aplicaciones estudiadas permitirán establecer características no funcionales que definirán los atributos de calidad del software, entre ellos protocolos de transferencia de información segura (HTTPS) y protocolos de autenticación y acceso para garantizar la seguridad de la información; facilitar la usabilidad del aplicativo mediante un diseño sencillo y responsivo y, mejorar la eficiencia de los procesos a través de una construcción que vaya acorde a las necesidades identificadas.

Además de tener en cuenta las características clave de los sistemas de gestión de información anteriormente mencionados y sus aportes al estructuramiento del desarrollo de este proyecto, es factible implementar dos tecnologías cuyas características no han sido adoptadas en las plataformas revisadas, que se mencionan a continuación.

En primer lugar, se optará por aplicar una arquitectura de microservicios, mecanismo adecuado para la integración continua que permite mejorar la eficiencia de operación, mantenimiento, desarrollo y escalabilidad de un software. Lo anterior lo corrobora Benet (2019), la experta en Cloud y Big Data, al manifestar que: Los microservicios son unidades funcionales concretas e independientes, que trabajan juntas para ofrecer la funcionalidad general de una aplicación. Cada microservicio puede ser actualizado o escalado sin que esto afecte a la disponibilidad de las demás unidades y de la aplicación en su conjunto. (párr. 1)

Por lo tanto, cuando ocurre un problema en un software desarrollado con una arquitectura de microservicios, sólo se necesita separar la parte conflictiva sin afectar a todo el aplicativo, por tal razón es indispensable para una mayor responsabilidad, cumplimiento y seguridad de los datos.

Como segunda tecnología se abordará el desarrollo de la plataforma mediante la utilización de Docker, herramienta conocida como la "tecnología de código abierto" que puede implementar aplicaciones en varios entornos.

Docker mediante contenedores facilita la creación y la ejecución de arquitecturas de microservicios distribuidos, la implementación de código con canalizaciones de integración y entregas continuas estandarizadas, la creación de sistemas de procesamiento de datos altamente escalables y la creación de plataformas completamente administradas para sus desarrolladores. (Amazon Web Services, 2022).

Por lo anterior, Docker es una herramienta diseñada para beneficiar tanto a los desarrolladores como a los administradores de sistemas. Para los desarrolladores, significa que pueden concentrarse en escribir código sin preocuparse por el sistema en el que finalmente se ejecutará, también, les permite obtener una ventaja inicial mediante el uso de uno de los miles de programas ya diseñados para ejecutarse en un contenedor Docker como parte de su aplicación. Para el personal de operaciones, Docker brinda flexibilidad y reduce potencialmente la cantidad de sistemas necesarios debido a su tamaño reducido y sus gastos generales más bajos. De esta manera, se hace imprescindible la implementación de una herramienta como Docker. Además, los contenedores permiten a los desarrolladores crear y enviar simultáneamente diferentes microservicios; integrarlos con otros sistemas y organizarlos automáticamente utilizando reglas y procesos predefinidos. Esto se traduce en una mayor productividad, una mejor calidad de código y un tiempo de gestión más rápido.

1.5. Método De Desarrollo

Para el desarrollo de la plataforma web se empleará la metodología ágil eXtreme Programming (XP) elegida por sus características de comunicación estrecha entre el equipo de desarrollo y el cliente, respuestas óptimas a cambios continuos y retroalimentación constante para construir un producto que vaya en línea con los requerimientos del cliente. En la metodología aplicada se identificaron diferentes lineamientos entre ellos: etapas, artefactos y roles. A continuación, se describe cada uno de ellos.

1.5.1. Etapas de la metodología XP

Etapa 1 – Planificación

En la fase de planificación se construirán las historias de usuario (Tabla 6) basadas en entrevistas semiestructuradas con el cliente para especificar las funcionalidades del software, que se validarán con el cliente mediante sesiones de trabajo y, basado en la priorización de las mismas se establecerán 5 iteraciones y, los tiempos en los cuales se entregará cada funcionalidad será en un lapso de 2 a 4 semanas (de acuerdo a como se vaya a desarrollando el proyecto se ajustarán de manera dinámica).

Tabla 6

Plantilla para el diseño de Historias de Usuario¹

Historia de Usuario			
Código:			
Nombre:			
Actor:			
Descripción:			
Criterios de aceptación:	CID	Condición	Resultado

¹ Formato elaborado con base en la propuesta presentada por Villamizar, Katerine. Definición de equivalencias entre historias de usuario y especificaciones en UN-LENCEP para el desarrollo ágil de software. 2013. pp. 30-32.

- **Código:** es un valor único que identifica la historia de usuario de otras historias dentro de un proyecto. Se puede presentar en cualquier formato.
- **Nombre:** es el título de la historia de usuario como resumen de la funcionalidad que se describirá en ella. Generalmente se representa a través de un verbo en infinitivo como la acción que se desarrollará.
- **Actor:** es el rol que tendrá la persona que realiza la funcionalidad de la historia de usuario en el sistema a construir.
- **Descripción:** representa en una frase lo que quiere el usuario y para qué lo necesita.
- **Criterios de aceptación:** Detallan cómo se debe comportar el sistema para ejecutar la descripción de la historia de usuario. Se compone de CID (identificador del criterio de aceptación), y la condición representa lo que se debe cumplir para obtener dicho resultado.

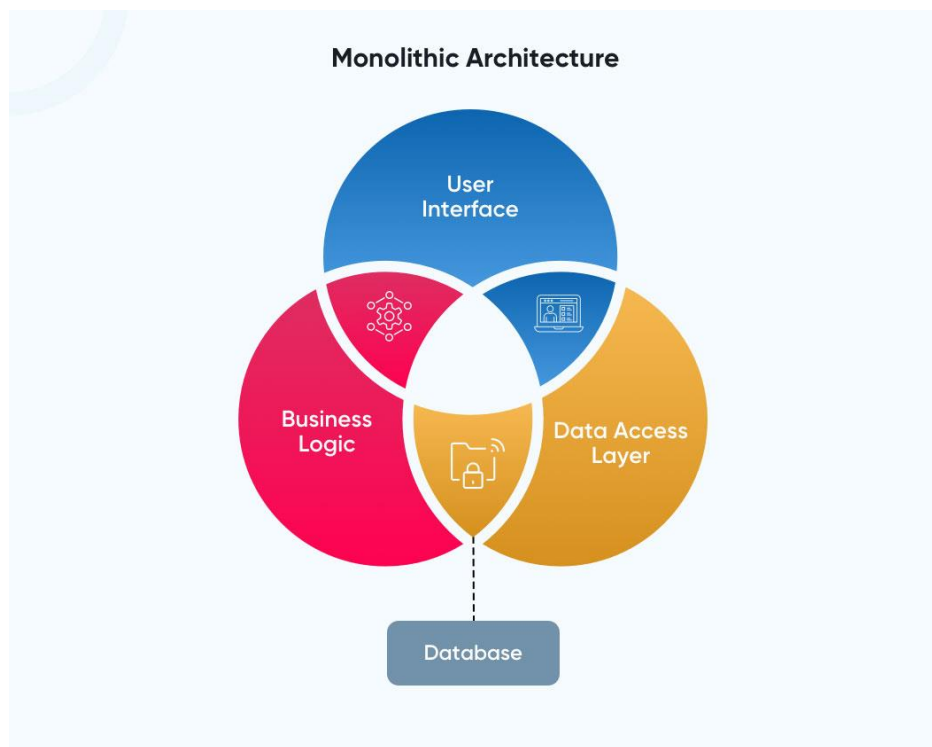
Etapas 2 – Diseño

Para el diseño del aplicativo se realizarán dos etapas que llevarán a una planificación, desarrollo e implementación correcta de la arquitectura de microservicios:

1. Inicialmente se construirá un aplicativo monolítico implementando una estructura de tres capas (Figura 6):
 - **Capa de presentación (UI):** Interfaz de usuario, es la parte gráfica del aplicativo y permite la interacción del usuario con las funcionalidades y servicios desarrollados en el software.
 - **Capa de negocio (Business Logic):** Lógica del aplicativo, todo el desarrollo y comunicación de procesos e información pasa por este nivel, al igual que las solicitudes y respuestas hacia las demás capas.
 - **Capa de datos (Data Access):** Base de Datos, almacena y gestiona la información de todo el sistema, procesa los datos y los devuelve al nivel de aplicación.

Figura 6

Arquitectura monolítica de tres capas.

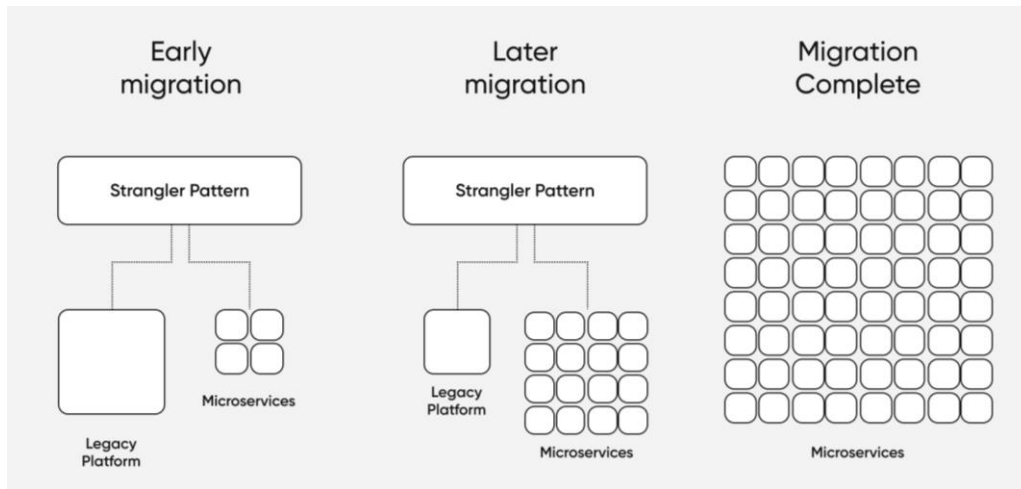


Fuente: Tomado de *Monolithic vs Microservices Diagram*, por Manchada A, 2021, net solutions (<https://bit.ly/38kEa7B>).

2. Para migrar la aplicación monolítica a una arquitectura de microservicios se aplicará el patrón de descomposición *Strangler* (Figura 7) el cual consiste en reemplazar gradualmente funcionalidades específicas por nuevos microservicios, se configurará una fachada mediante el patrón de integración *API Gateway* (Figura 8) para enrutar las peticiones de los usuarios con los microservicios y el monolito heredado (*API Gateway* aplica autenticación y control de acceso a las solicitudes de las API), una vez migradas todas las funcionalidades se eliminará la aplicación monolítica y se obtendrá un software basado en microservicios.

Figura 7

Patrón de descomposición strangler: Fases de migración.

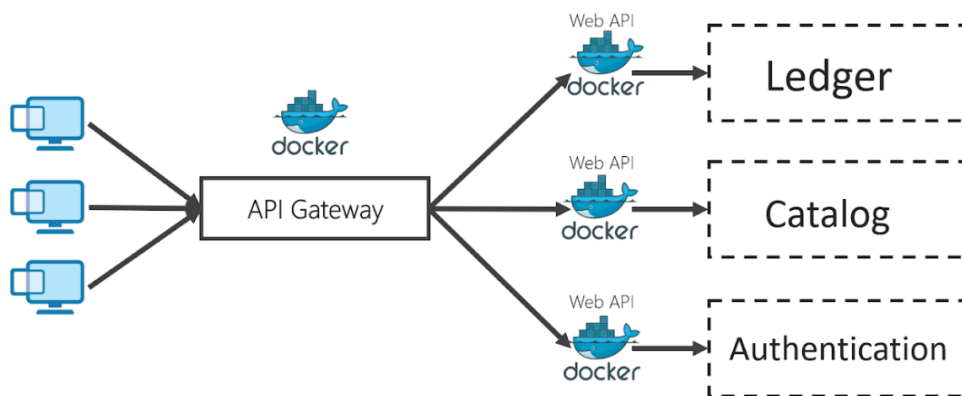


Fuente: Tomado de *What is Replatforming?*, por Zamoya D, 2021, fabric (<https://fabric.inc/blog/replatforming/>).

Figura 8

API Gateway como único punto de enlace entre peticiones y microservicios.

Dockerized Gateways



Nota: API Gateway y microservicios empaquetados en contenedores Docker

Fuente: Tomado de *Containerizing API Gateways*, por Chua A, 2018, PogsDotNet (<https://bit.ly/3FA9cV5>).

Para la construcción de la Base Datos Relacional se utilizará un diseño de tres modelos (Ruíz, 2001):

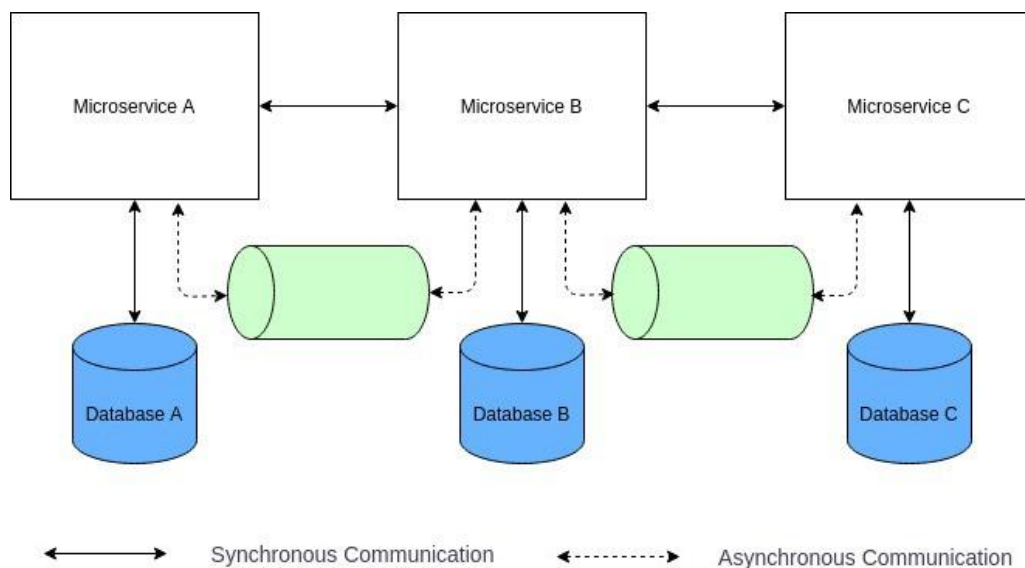
1. **Modelo Conceptual:** En esta fase se identificará y analizará los requisitos del sistema de información, para la representación del **esquema conceptual** se utilizará la metodología de diseño de bases de datos denominada Modelo Entidad-Relación (MER) con el fin de elaborar un diagrama que establezca un modelo de entidades, atributos, relaciones y cardinalidad.
2. **Modelo Lógico:** A partir del esquema conceptual, los requerimientos del sistema de información y el modelo lógico estándar (modelo relacional) se elaborará el **esquema lógico estándar** donde se transforman las entidades en tablas normalizadas, se definen las llaves primarias, las llaves foráneas y las interrelaciones N:M se convierten en nuevas tablas.
3. **Modelo Físico:** Se toma como entrada el esquema lógico y para el diseño del **esquema físico** se agregarán los tipos de datos y restricciones para cada atributo de cada una de las tablas, el esquema físico representará como se construirá el modelo en sistema gestor de base de datos (SGBD) mediante el lenguaje de definición de datos (DDL).

En este punto se implementará el patrón de base de datos *Database per Microservice* el cual consiste en proporcionar un único almacén de datos (esquemas, tablas separadas, colecciones, bases de datos) a cada microservicio, evitando las dependencias en la capa de base de datos; para la comunicación entre los microservicios (Figura 9) se realizará mediante las llamadas de API's (REST o gRPC²).

² REST utiliza métodos HTTP 1.1 como GET, POST, PUT y DELETE mientras que gRPC utiliza métodos HTTP/2 formato de Buffer de protocolo para intercambios de datos más veloz que REST

Figura 9

Comunicación síncrona y asíncrona entre microservicios.



Nota: Cada microservicio cuenta con su único almacén de datos inaccesible para los demás microservicios.

Fuente: Tomado de *Design Patterns for Microservice Architecture*, por Kamaruzzaman M, 2021, Towards Data Science (<https://bit.ly/3FwP0Ug>).

Etapa 3 - Codificación

Para la codificación del aplicativo se emplearán diferentes tecnologías (Tabla 7) especializadas para el desarrollo web. Se aplicarán los estándares de Java Language Specification (JLS)³ documento técnico que describe las reglas y la sintaxis del lenguaje de programación Java, y es utilizado como la referencia para el diseño, desarrollo y mantenimiento del lenguaje. Para el apartado de backend, se implementará el patrón de diseño de software Modelo-Vista-Controlador (MVC) con el fin de separar la lógica de negocios con su visualización; la vista del patrón de diseño MVC se presentará renderizada como un API REST o GraphQL.

³ Oracle Corporation. Java Platform, Standard Edition Java Language Specification.. (2022). Recuperado 8 May 2022, de <https://docs.oracle.com/javase/specs/>

Tabla 7

Tecnologías utilizadas para el desarrollo de la plataforma

Tipo	Nombre	Versión	Justificación
Lenguaje de etiquetas de hipertexto	HTML	5	Permite la maquetación de elementos de la interfaz gráfica mediante etiquetas de hipertexto. ⁴
Hojas de estilo en cascada	CSS	3	Estiliza las etiquetas HTML y permite realizar un diseño responsivo de la plataforma. ⁵
Lenguaje de programación	JavaScript	ES6 +	Lenguaje de programación especializado en el desarrollo web, cuenta con una variedad de Frameworks y librerías que facilitan el desarrollo con esta tecnología. ⁶
Framework Frontend de JavaScript	Angular	13.3.5	Proporciona una gran escalabilidad del aplicativo, aumenta la eficiencia por su arquitectura basada en componentes y, facilita el mantenimiento de la plataforma. ⁷
Lenguaje de programación	Java	19	Lenguaje de programación multiplataforma y orientado a objetos, seguro, rápido y flexible para crear aplicaciones multipropósitos. ⁸
Framework BackEnd de Java	Spring	6	Framework robusto que implementa el patrón de diseño de software MVC, automatización de construcción de código, facilita la utilización de microservicios y proporciona un sistema reactivo y servicios en la nube. ⁹
Lenguaje de consulta para API's	GraphQL	16.6.0	Es un lenguaje de consulta que permite un acceso eficaz a grandes volúmenes de datos extrayendo información precisa sin necesidad de realizar una consulta completa

⁴ Conceptos Básicos de HTML | MDN. (2022). Recuperado 2 May 2022, de <https://mzl.la/3MLZ5im>

⁵ CSS | MDN. (2022) Recuperado 2 May 2022, de <https://mzl.la/3y6cz4D>

⁶ JavaScript | MDN. (2022) Recuperado 2 May 2022, de <https://mzl.la/3vCo9mv>

⁷ Angular. (2022). Recuperado 2 May 2022, de <https://angular.io/docs>

⁸ Amazon Web Services. ¿Qué es Java?. Recuperado 2 May 2022, de <https://aws.amazon.com/es/what-is/java/>

⁹ Spring. Why Spring?. Recuperado 2 May 2022, de <https://spring.io/why-spring>

			y escalabilidad al agregar nueva información sin afectar las consultas existentes. ¹⁰
Gestor de Base de Datos	PostgreSQL	14	Es uno de los gestores de bases de datos más potentes, compatible con todas las tecnologías que se emplearán, estable, fácil de configurar y además de uso gratuito. ¹¹
Software de desarrollo de aplicaciones en contenedores	Docker	4.7.1	Estandariza el código, optimiza la ejecución de microservicios y automatiza el despliegue del aplicativo. ¹²
Editor de código	Visual Studio Code	1.66	Editor de texto flexible, personalizable y con variedad de extensiones que permiten la facilidad y comodidad del desarrollo del software ¹³
Editor de código	IntelliJ	17.0.5	Editor de texto que integra funcionalidades que permiten la asistencia en la codificación y personalización; cuenta con un gran soporte para el lenguaje de programación JAVA. ¹⁴
Software para testing de API Rest	Postman	9.16.0	Se utilizará para realizar las pruebas de las API REST desarrolladas en el BackEnd. ¹⁵
Servicios de computación en la nube	AWS	N/A	Plataforma donde se utilizarán servicios como EC2 y ECS para el despliegue y pruebas del aplicativo. ¹⁶
Plataforma de control de versiones	GitHub	2.35.1	Utilizado para alojar el proyecto, controlar las versiones y trabajar conjuntamente en la codificación del aplicativo. ¹⁷

¹⁰ GraphQL. Learn GraphQL. Recuperado 2 May 2022, de <https://graphql.org/learn/>

¹¹ PostgreSQL: Documentation. (2022). Recuperado 2 May 2022, de <https://postgresql.org/docs/>

¹² Docker Documentation. (2022). Recuperado 2 May 2022, de <https://docs.docker.com/>

¹³ Documentation for Visual Studio Code. (2022). Recuperado 2 May 2022, de <https://code.visualstudio.com/docs>

¹⁴ JetBrains. IntelliJ IDEA. Recuperado 2 May 2022, de <https://www.jetbrains.com/es-es/idea/>

¹⁵ Introduction | Postman Learning Center. (2022). Recuperado 2 May 2022, de <https://bit.ly/3OQS2XC>

¹⁶ AWS Documentation. (2022). Recuperado 5 May 2022, de <https://go.aws/3LTvNyg>

¹⁷ GitHub. (2022) *Get Started*. Recuperado 11 May 2022, de <https://docs.github.com/en/get-started>

Etapa 4 – Pruebas

Para el *testing* del software se implementarán dos tipos de pruebas:

1. Pruebas automatizadas con escenarios (Tabla 8) basados en casos de prueba (Tabla 9):

- **Pruebas Unitarias (*Unit Test*)**

Se efectuarán en cada microservicio para validar unidades pequeñas de código como funciones, objetos o eventos mediante la comprobación de entradas y resultados esperados

- **Pruebas End to End (*E2E*)**

Se enfoca en probar el flujo del aplicativo, la integra de inicio a fin mediante una simulación de experiencia de usuario con la interacción, integración y comunicación entre los microservicios.

Tabla 8

Artefacto de *identificación de un escenario de prueba*¹⁸

Código HU	
No	
Descripción	
Datos	
CID	

- **Código HU:** identificador único de la historia de usuario para la cual se elabora el escenario.
- **No:** valor único que identifica el escenario de prueba de acuerdo con la historia de usuario.
- **Descripción:** descripción del escenario de prueba.
- **Datos:** valores del estado inicial de el/los objeto(s) para los cuales que conforman el escenario de prueba.
- **CID:** identificador del criterio de aceptación de la historia de usuario para la cual se realiza la prueba.

¹⁸ Artefacto elaborado con base en la propuesta hecha por Villalobos, J. A. (2008). Introducción a las Estructuras de Datos. Aprendizaje Activo Basado en Casos. Editorial Prentice Hall.

Tabla 9

Artefacto para especificar un caso de prueba¹⁹

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
-------	--------	-------	--------	----	-----------	--------------------	--------------------

- **CPIId:** valor único que identifica el caso de prueba.
- **Nombre:** nombre descriptivo del caso de prueba, se recomienda usar el prefijo “test”.
- **Clase:** objeto al que pertenecen los métodos a validar.
- **Método:** método al que aplican los casos de prueba.
- **HU:** valor único que identifica la historia de usuario de otras historias dentro de un proyecto.
- **Escenario:** valor único que identifica el escenario de prueba de acuerdo con la historia de usuario.
- **Valores de entrada:** datos que se ingresan para ejecutar el caso de prueba.
- **Resultado esperado:** resultado que se debe obtener al ejecutar el caso de prueba, según el escenario al que corresponda.

2. Pruebas con usuarios finales en escenarios reales:

▪ Pruebas de aceptación

El usuario interactúa con el software validando que el funcionamiento satisfaga sus necesidades y el resultado sea acorde a sus expectativas.

Etapa 5 – Lanzamiento

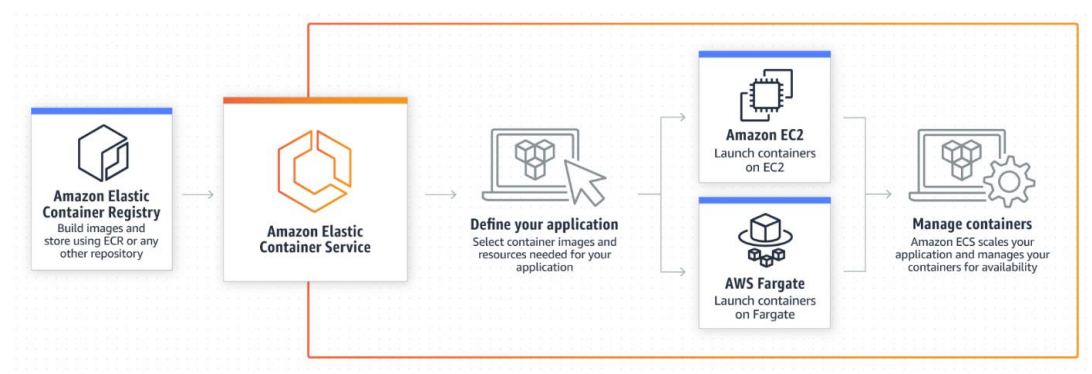
Para el despliegue del aplicativo en un entorno de pruebas se utilizará un servidor *cloud* de *Amazon Web Services* (AWS), para ello se empleará el servicio *Amazon Elastic Container Service* (ECS) (Figura 10), en él se creará un cluster cuyas tareas serán las imágenes de contenedores Docker de los microservicios desarrollados y del *API Gateway*, se ejecutarán en una infraestructura registrada al cluster mediante la utilización del servicio *Amazon Elastic Compute Cloud* (EC2)

¹⁹ Artefacto elaborado con base en la propuesta hecha por Villalobos, J. A. (2008). Introducción a las Estructuras de Datos. Aprendizaje Activo Basado en Casos. Editorial Prentice Hall.

para utilizar un servidor virtual configurable en aspectos como seguridad, redes, almacenamiento y especificaciones de hardware permitiendo un control específico a los requerimientos del software y una accesibilidad de la plataforma en cualquier dispositivo.

Figura 10

Despliegue de contenedores en el servicio ECS.



Fuente: Tomado de *Amazon Elastic Container Service (Amazon ECS)*, por Amazon Web Services, AWS (<https://aws.amazon.com/ecs/>).

1.5.2. Artefactos

Los productos tangibles resultados de las actividades del proyecto de desarrollo de software serán:

- **Historias de usuario:** Tablas con las especificaciones de los requisitos del software en las cuales se describe las características y funcionalidades que deberá tener el aplicativo.
- **Plan de entregas (*Release Plan*):** Cronograma que establece que historias de usuario serán agrupadas y ordenadas para conformar las entregas con base a estimaciones de tiempos realizadas por el equipo de desarrollo.
- **Plan de iteración (*Iteration Plan*):** Se presentarán el número de ciclos de iteración donde se desarrolla y prueba cada entrega de acuerdo al orden y tiempos establecidos en el plan de entregas.

- **Casos de prueba:** Conjunto de condiciones que validan el funcionamiento correcto de los servicios del aplicativo.
- **Diagrama Entidad-Relación:** Diseño del esquema de la base de datos con entidades, atributos, relaciones y cardinalidad elaborado mediante la identificación de requerimientos.
- **Diagrama Lógico:** Representación de la información por medio de tablas relacionadas y normalizadas.
- **Scripts de la Base de Datos:** Archivo de texto plano que contiene un conjunto de comandos de lenguaje de consulta estructurado (SQL) para la creación de tablas y gestión de información en la Base De Datos.

1.5.3. Roles

Los papeles que se desempeñarán en el desarrollo del aplicativo con sus respectivos responsables serán:

- **Cliente:** Coordinador de Investigación de la Universidad Mariana Dr. Harold Juajibioy Otero, definirá los objetivos y necesidades que debe suplir el aplicativo, además, probará las funcionalidades que se entreguen en cada iteración para verificar que cumplan con sus requerimientos.
- **Programadores:** Andres Esteban Rizo y Jeisson Fernando Caicedo, encargados de desarrollar y cumplir con los objetivos del software, además, estimar los tiempos y duraciones de cada iteración.
- **Testers:** Andres Esteban Rizo y Jeisson Fernando Caicedo, construirán los tipos de pruebas en todos los microservicios para verificar el correcto funcionamiento de los mismos y que los resultados estén alineados a los requisitos estimados.
- **Coach:** Mg. Fabián Parra Pay, encargado del asesoramiento y orientación constante del equipo de trabajo y del cliente; por otro lado, también asumirá el rol de tracker cuya función será realizar seguimiento y control del desarrollo del aplicativo.

1.6. Línea y Áreas Temáticas

Línea: Ingeniería, Informática y computación.

Áreas Temáticas: Innovación, modelamiento y desarrollo de software

1.7. Presupuesto

Tabla 10

Presupuesto global del proyecto

RUBROS	TOTAL (\$)
INVERSIÓN EN PERSONAL	\$ 7'466.368
OTROS RUBROS	\$ 5'900.000
TOTAL	\$ 13'366.368

Tabla 11

Descripción de la Inversión en personal.

NOMBRE INVESTIGADOR	Vr. Hora Investigador	DEDICACIÓN Número total de horas	VALOR
Fabián Parra Pay	\$ 16.666	64	\$ 1'066.624
Andrés Esteban Rizo Zamora	\$ 8.333	384	\$ 3'199.872
Jeisson Fernando Caicedo Burbano	\$ 8.333	384	\$ 3'199.872
TOTAL			\$ 7'466.368

Tabla 12

Otros rubros

RUBRO	JUSTIFICACIÓN	VALOR TOTAL
Equipos	Computadores: Necesario para el diseño, codificación, comunicación, pruebas y despliegue del aplicativo. Internet: Acceder a la documentación oficial de las tecnologías utilizadas, acceder al repositorio de GitHub para codificar el software en conjunto, ejecutar la aplicación en local host.	\$ 5'500.000
Software	Es necesario la utilización de los servicios de AWS para evaluar la plataforma web en un entorno de pruebas.	\$ 400.000
TOTAL		\$ 5'900.000

1.8. Cronograma

Actividades	Tiempo (Meses)											
	Ago	Sep	Oct	Nov	Feb	Mar	Abr	May	Ago	Sep	Oct	Nov
Objetivo específico 1	■	■	■	■								
Crear formato de recolección de información	■											
Aplicar formato de recolección de información		■	■									
Analizar la información			■									
Diseñar el modelado de procesos			■	■								
Describir la oportunidad de intervención			■	■								
Objetivo específico 2			■	■	■	■	■	■	■	■	■	■
Planificación			■	■	■							
Diseño				■	■							
Iteración 1					■	■	■					
Iteración 2						■	■	■				
Iteración 3							■	■	■			
Iteración 4								■	■	■		
Iteración 5									■	■	■	
Despliegue en entorno de pruebas										■	■	
Escribir un artículo para divulgar											■	■

1.9. Productos esperados

- Monografía (documento de informe final)
- Informe Técnico
- Carta de Innovación
- Código fuente completo
- Tutorial o video tutorial de la instalación de la solución

1.10. Condiciones de entrega

Para la entrega de documentos; como es la monografía, informe técnico y carta de innovación, al igual que, el video de instalación de la solución, serán entregados de manera digital. Mientras que para el código fuente se compartirá el proyecto del repositorio GitHub para su descarga.

2. Presentación de resultados

2.1 Modelado de procesos

Describir la oportunidad de intervención en la gestión de la actividad investigativa estudiantil de la Universidad Mariana a través del modelado de procesos.

Para el desarrollo de este objetivo se implementó un formato de recolección de información a través de entrevistas semi-estructuradas (Anexo B) para conocer el proceso de gestión de información de la investigación estudiantil desde las diferentes áreas específicas y generales que se van a abordar, se aplicó el formato al Coordinador de Investigación de la Universidad Mariana Dr. Harold Juajibioy Otero, Coordinador de Investigación del programa de Ingeniería de Sistemas Dr. Fabian Parra Pay, Coordinadora de Semilleros de Investigación del programa de Ingeniería de Sistemas Mgs. Leidy Marcela Gómez Melo y la Coordinadora de Co-investigación estudiantil Dra. Alejandra Zuleta Medina (Anexo C); mediante los resultados obtenidos se analizó los procesos en común y se creó un modelado de procesos unificado (Figura 12) que abarca cada una de las fases de gestión de cada área.

El modelado de procesos se desarrolló aplicando el estándar Business Process Model and Notation (BPMN)²⁰ y a través del software Bizagi Modeler²¹ escogido por soportar el estándar BPMN, ser software libre y por la alta variedad de herramientas que permiten modelar, simular y documentar detalladamente el proceso.

²⁰ *About the Business Process Model And Notation Specification Version 2.0.* (s. f.). Recuperado 9 de octubre de 2022, de <https://www.omg.org/spec/BPMN/2.0/>

²¹ *Bizagi, One Platform; Every Process.* (s. f.). Recuperado 9 de octubre de 2022, de <https://bit.ly/3rLojVP>

Figura 11

Modelado de procesos de la gestión de la actividad investigativa estudiantil de la Universidad Mariana - Primera fase vinculación de estudiantes

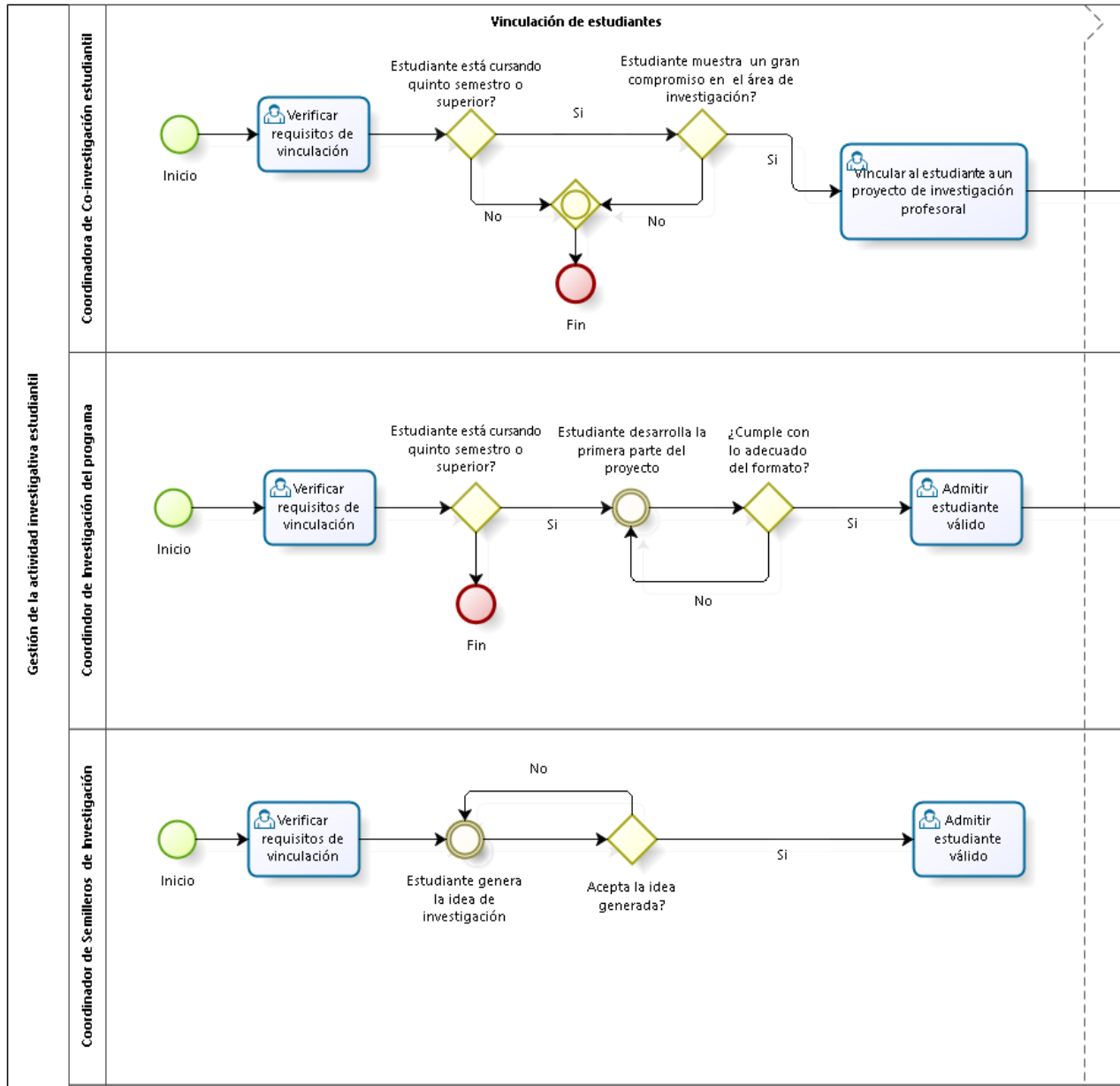
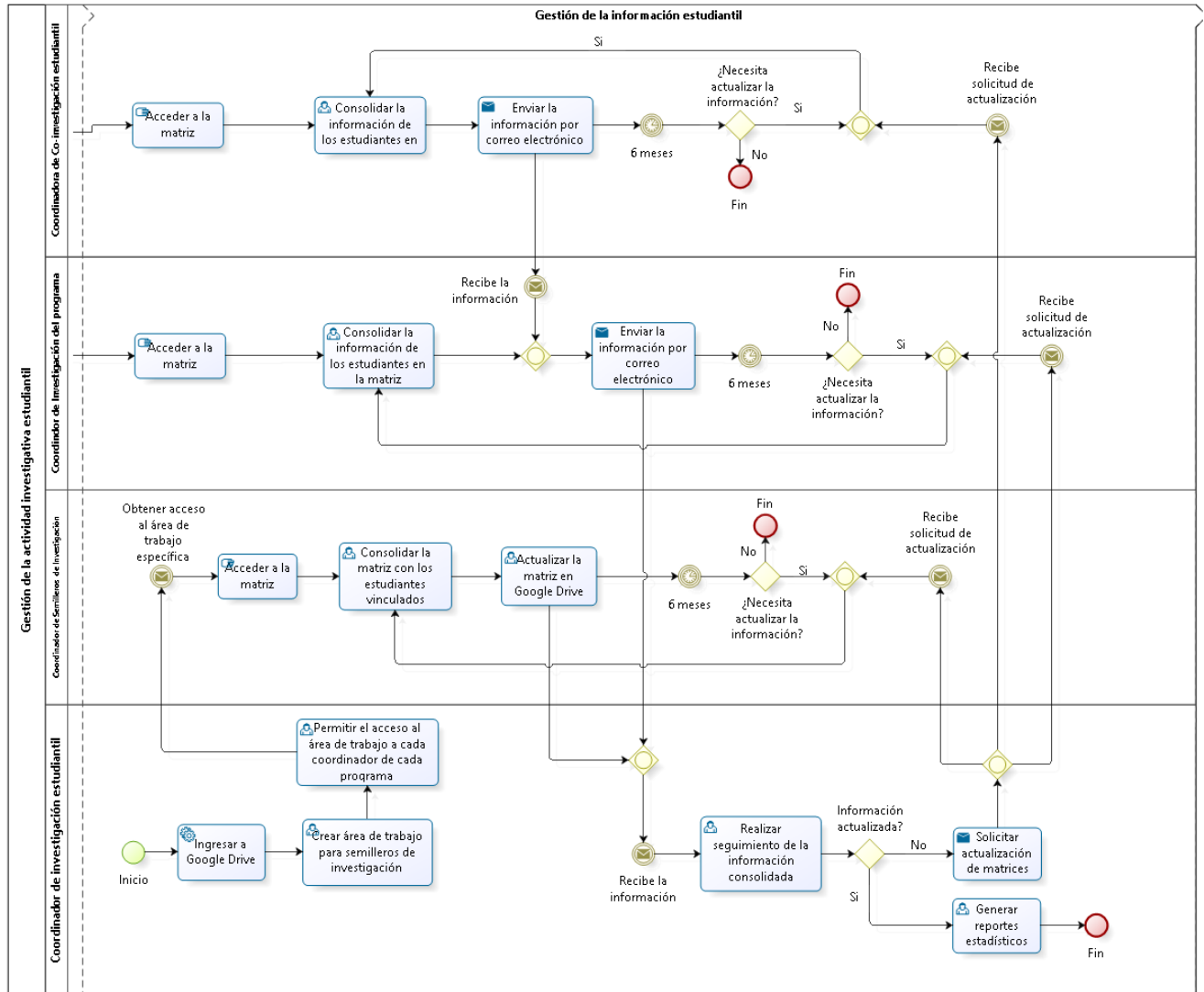


Figura 12

Modelado de procesos de la gestión de la actividad investigativa estudiantil de la Universidad Mariana - Segunda Fase Gestión de la información



Se puede evidenciar que el proceso se divide en cuatro actores, el primero es el Coordinador de investigación estudiantil de la Universidad Mariana encargado de asignar un espacio de trabajo a la modalidad de Semilleros de Investigación a través de la plataforma Google Drive y en el cual se proporciona el acceso al Coordinador de Semilleros de Investigación de cada programa encargado de la gestión estudiantil; cada espacio de trabajo lo subdivide en facultades y estos en programas en los que implementa la matriz específica para su respectivo diligenciamiento. El

Coordinador de Investigación de cada programa y el Coordinador de Co-Investigación estudiantil se encargan de identificar a través del análisis de proyectos e ideas, que estudiantes se vincularon para tramitar las matrices desde Excel y realizar los ajustes pertinentes cuando sea requerido; el proceso de envío de archivos y comunicación con el Coordinador de Investigación Estudiantil se realiza por medio de correo electrónico donde accede a las matrices para realizar un seguimiento y poder generar reportes estadísticos para la generación de informes; los datos de las matrices se actualizarán cada semestre para registrar a los nuevos estudiantes y los avances que va desarrollando el estudiante investigador; por otro lado el Coordinador de Investigación estudiantil puede solicitar la actualización de las matrices según vea conveniente.

2.1.1 Oportunidad de intervención

Al examinar el modelado de procesos desarrollado se observa que muchas de las actividades que lo componen son de tipo manual y el diligenciamiento, actualización, seguimiento y análisis de información resulta ser muy compleja debido a la gran cantidad de datos que se emplean, esto conlleva a una entrega tardía de las matrices, errores en la evolución de mantenimiento, confusiones en registros de datos y una lenta toma de decisiones. Analizando estas problemáticas se identificó que es posible desarrollar un software que automatice todos los procesos anteriormente mencionados con el fin de aumentar la productividad, disminuir sustancialmente la carga de actividades, reducir la posibilidad de generación de errores, facilitar la administración de la información estudiantil, por ende ayudar a la coordinación estudiantil a optimizar los flujos de trabajo y procesar registros más ágiles, lo que permite acomodar volúmenes mucho más altos a medida que aumenta la actividad investigativa estudiantil; por último se estandarizaría en primer lugar el formato de diligenciamiento de las matrices tanto para las modalidades de investigación que cubre el software como para todos los programas de las diferentes facultades que cuenten con estas modalidades facilitando la generación de informes y reportes estadísticos y en segundo lugar la comunicación entre Coordinadores, agilizando las consultas, revisiones y solicitudes entre ambas partes.

2.1.2 Discusión

Mediante el desarrollo de este objetivo, se logró concluir que:

A través de la creación e implementación de entrevistas semi-estructuradas (comunicación frente a frente con el cliente), fue posible describir de manera muy detallada un proceso complejo donde intervienen varios actores de diferentes modalidades.

Por medio de la aplicación de estándares como BPMN y el uso de software que soporte el estándar, fue factible crear un modelado de procesos de alta calidad con una documentación precisa permitiendo identificar de manera sencilla el ciclo del proceso diseñado.

A través del análisis del modelado de procesos fue posible determinar la oportunidad de intervención identificando las actividades que causan problemáticas e ideando soluciones funcionales que ayuden a optimizar y automatizar las acciones que actualmente se ejecutan.

2.2 Desarrollo del aplicativo web

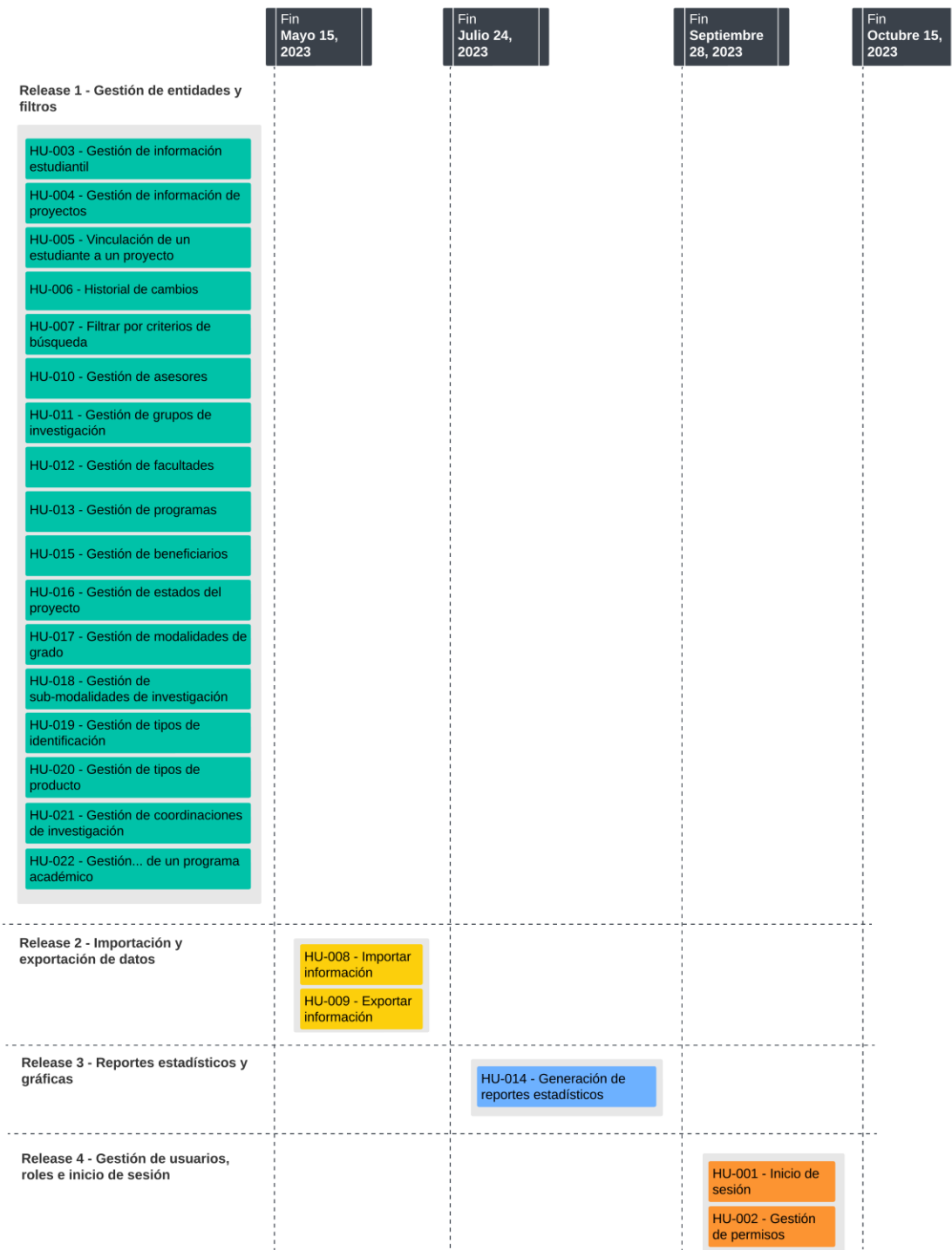
2.2.1 Etapa 1 – Planificación

Para identificar las funcionalidades que tendrá la plataforma, se realizaron las historias de usuario (Anexo D) a través del análisis de las matrices y las entrevistas semiestructuradas aplicadas al Coordinador de Investigación Estudiantil y a los Coordinadores de las distintas modalidades de investigación.

Al realizar las historias de usuario, se desarrolló el plan de entregas (Figura 13) el cuál agrupa por prioridad historias de usuario específicas que se realizarán en diferentes periodos de tiempo:

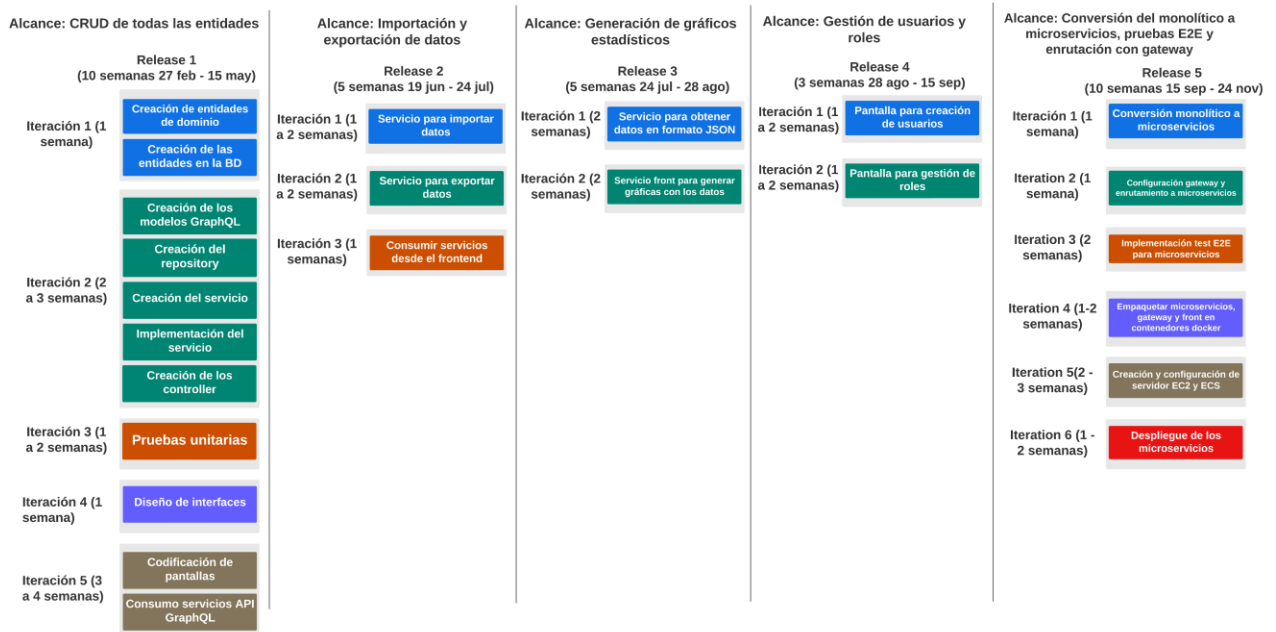
Figura 13

Plan de entregas



Cada entrega cuenta con un número de iteraciones (Figura 14) donde se desarrollan funcionalidades concretas y se realizan las pruebas de cada una de ellas:

Figura 14
Plan de iteración

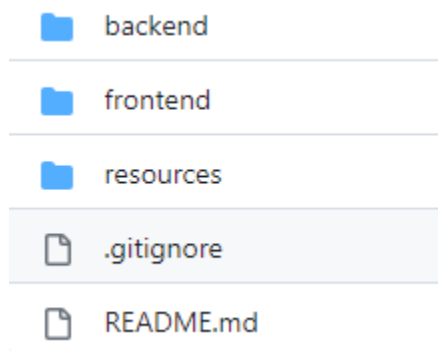


2.2.2. Etapa 2 – Diseño

Para el desarrollo del aplicativo se creó una estructura con los directorios presentados en la figura 15, la cual abarca el proyecto BackEnd y FrontEnd y un directorio de recursos donde se encuentra la documentación del proyecto.

Figura 15

Estructura del proyecto



▪ Estructura del proyecto BackEnd - Capa de negocio

El proyecto BackEnd cuenta con dos directorios principales:

- **Directorio Docker:** Se encuentran los scripts de la base de datos de PostgreSQL que automatizan la creación del schema y las tablas del proyecto y la importación de datos predefinidos, el archivo docker-compose el cual cuenta con los volúmenes, networks y servicios de los proyectos compilados en imágenes Docker.
- **Directorio siumar-be:** Se encuentra el proyecto creado con el framework Spring Boot de Java, su estructura (Figura 16) cuenta con paquetes que permiten la implementación de la arquitectura MVC.

Figura 16

Estructura BackEnd

```
- src
  - main
    - java
      - org.umariana.siumar
        SiumarApplication
      + org.umariana.siumar.config
      + org.umariana.siumar.controller
      + org.umariana.siumar.entity
      + org.umariana.siumar.entity.enums
      + org.umariana.siumar.filters
      + org.umariana.siumar.repository
      + org.umariana.siumar.service
      + org.umariana.siumar.service.impl
    + resources
  + test
```

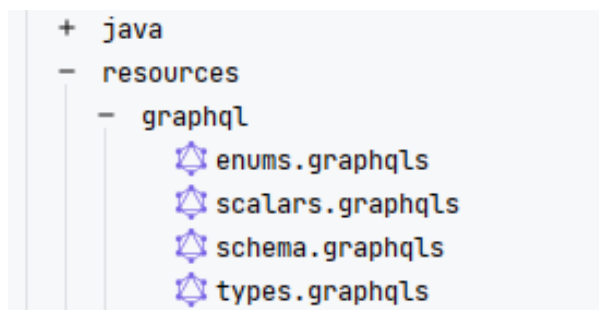
A continuación, se describen cada uno de los paquetes del proyecto:

- Las **entidades de dominio** son los objetos que le dan soporte a la lógica de negocio.
- El **repository** implementa los métodos básicos para la gestión de las entidades de dominio.
- El **service** es una interfaz en la que se definen métodos que serán ejecutados en los endpoints.
- La **implementación de servicio** es la lógica de negocio y le da funcionalidad a las interfaces del paquete service.
- El **controller** define los endpoints para realizar las peticiones desde el FrontEnd.

Adicionalmente, se implementó una API GraphQL (ver Figura 17), una tecnología valiosa para interactuar con servicios que suministran datos. Su API posibilita la desestructuración de la información, permitiendo la obtención de datos específicos de manera eficiente. Esta capacidad optimiza las consultas en conjuntos de datos diversos y extensos.

Figura 17

Estructura de GraphQL



A continuación, se describen cada uno de los archivos .graphql:

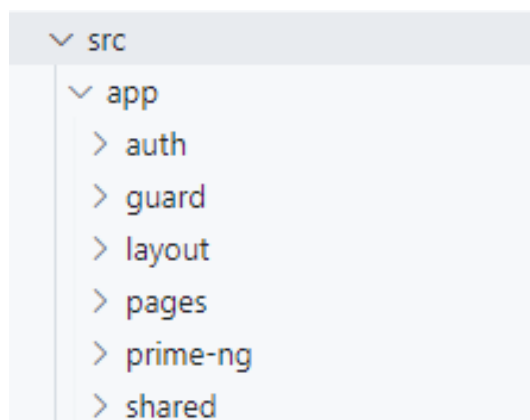
- Los **scalars**, en nuestro contexto, son tipos de datos personalizados que se configuran en el paquete de configuración para ser mapeados a tipos de datos primitivos de Java.
- En los **types**, se establecen las entidades que estarán disponibles al consumir la API de GraphQL.
- Los **enums**, representan conjuntos de valores predefinidos que respaldan el tipo de dato de un atributo en los types, brindando consistencia y control en la definición de atributos.
- Dentro del **schema**, se definen los métodos Query, que permiten realizar consultas sobre las entidades, y los Mutations, que posibilitan cambios directos en las entidades, como la creación, edición o eliminación.

▪ Estructura del proyecto FrontEnd - Capa de presentación

La estructura del FrontEnd cuenta con un directorio principal llamado “*siumar-fe*”, es el proyecto generado con el framework Angular (Figura 18) y un archivo “*docker-compose*” utilizado para generar la imagen de Docker del proyecto compilado.

Figura 18

Estructura FrontEnd



A continuación, se describen cada uno de los directorios del proyecto:

- El módulo **auth** contiene la autenticación de la plataforma, define las rutas que redireccionan a las páginas de login con Google, autenticación no autorizada y error interno del servidor.
- El directorio **guard** contiene una interfaz que es implementada por las rutas del aplicativo para bloquear su acceso cuando no existe un usuario autenticado o no cuenta con los privilegios de acceso.
- El módulo **layout** se encarga de orquestar los componentes del aplicativo para estructurar visualmente las pantallas.
- El módulo **pages** contiene los componentes que implementan las funcionalidades de CRUD de las entidades de dominio de manera visual a través de tablas y formularios.
- El módulo **prime-ng** centraliza las importaciones de los componentes de PrimeNG para que sean accesibles en todo el aplicativo.
- El módulo **shared** contiene los componentes compartidos por toda la plataforma, tales como el menú lateral, el menú superior y el pie de página.

- **Mockups de pantallas**

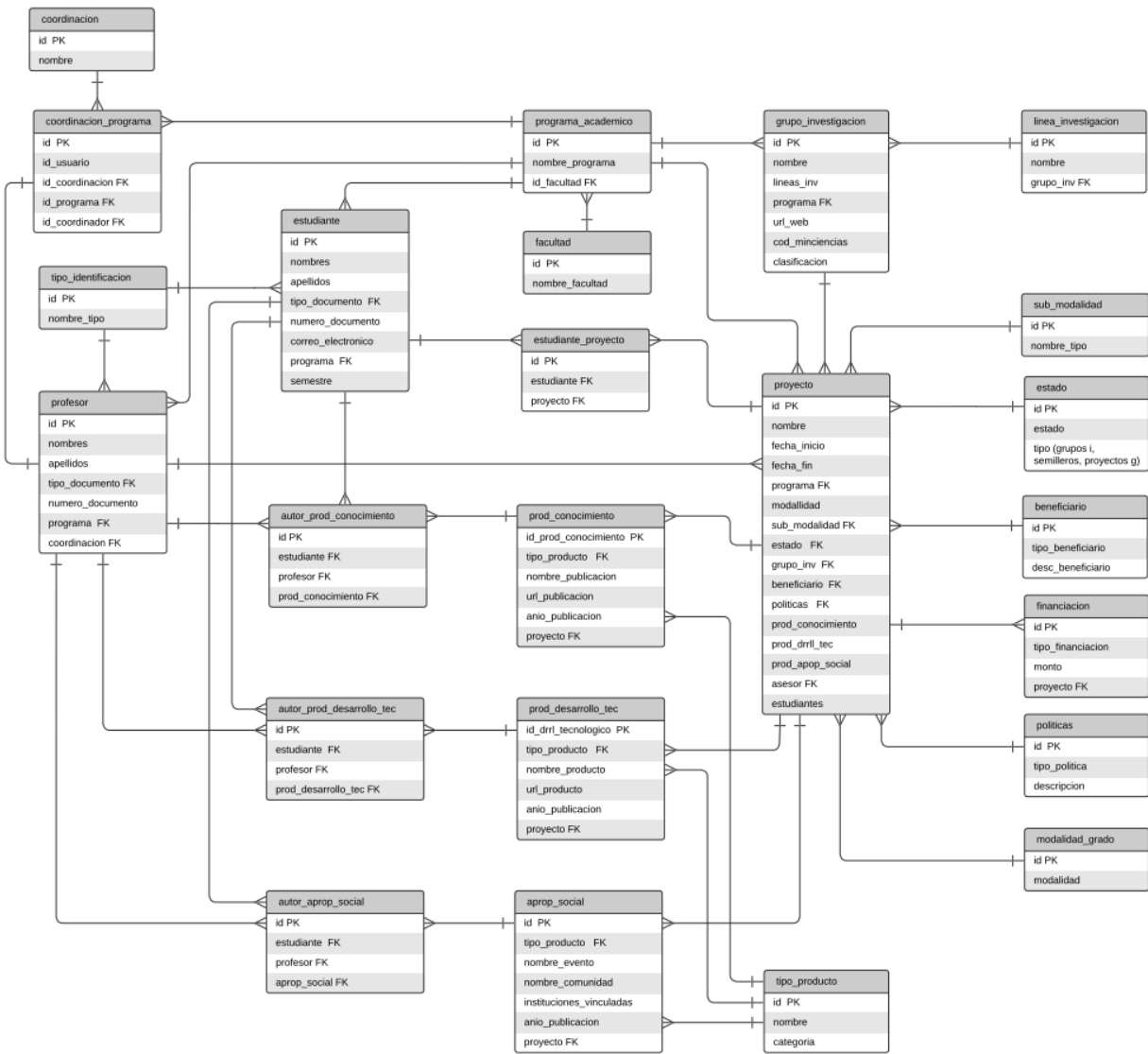
Se realizó el diseño de los mockups (Anexo E) para las pantallas desarrolladas que dan soporte a la interfaz visual de los componentes del aplicativo.

- **Capa de persistencia**

Para la implementación de la capa de persistencia, se ha desarrollado un modelo Entidad-Relación (ER) (Figura 19) y un modelo lógico (Figura 20). Estos modelos han permitido la identificación de diversas tablas necesarias para la gestión de proyectos presentes en la matriz.

Figura 20

Modelo lógico



2.2.3 Etapa 3. Codificación

En esta etapa del proceso de desarrollo, se llevó a cabo la implementación concreta del diseño detallado previamente. A lo largo de esta sección, se abordarán en detalle los aspectos fundamentales que delinearán la ejecución tanto del proyecto BackEnd, construido sobre Spring Boot, como del proyecto FrontEnd, desarrollado con Angular.

2.2.3.1. BackEnd

▪ Entidades

Se definieron entidades de dominio (Figura 21) utilizando anotaciones de Hibernate. Cada entidad representó una tabla en la base de datos y contiene atributos relevantes para la funcionalidad del sistema.

Figura 21

Entidades de dominio

```
- org.umariana.siumar.entity
  ○ AutorProdApropiacionSocial
  ○ AutorProdConocimiento
  ○ AutorProdDesarrolloTecnologico
  ○ Beneficiario
  ○ Coordinacion
  ○ Coordinador
  ○ Estado
  ○ Estudiante
  ○ EstudianteProyecto
  ○ Facultad
  ○ Financiacion
  ○ GrupoInvestigacion
  ○ LineaInvestigacion
  ○ ModalidadGrado
  ○ Politica
  ○ ProdApropiacionSocial
  ○ ProdConocimiento
  ○ ProdDesarrolloTecnologico
  ○ Profesor
  ○ ProgramaAcademico
  ○ Proyecto
  ○ SubModalidad
  ○ TipoIdentificacion
  ○ TipoProducto
```

▪ Repositorios

Se implementaron repositorios (Figura 22) para cada entidad que manejaría un controlador, se identificaron las entidades que podrían ser gestionadas por entidades padres con el fin de realizar su persistencia desde el mismo controlador; para la codificación de los repositorios, se utilizó *Spring Data JPA* (Java Persistence API) y se implementó la interfaz *JpaRepository* la cuál proporciona métodos que implementan las operaciones CRUD (Crear, Leer, Actualizar, Eliminar) en las entidades.

Figura 22

Repositorios

```
org.umariana.siumar.repository
├── AutorProdApropSocialRepository
├── AutorProdConocimientoRepository
├── AutorProdDesarrolloTecRepository
├── BeneficiarioRepository
├── CoordinacionRepository
├── CoordinadorRepository
├── EstadoRepository
├── EstudianteProyectoRepository
├── EstudianteRepository
├── FacultadRepository
├── LineaInvestigacionRepository
├── ModalidadGradoRepository
├── ProfesorRepository
├── ProgramaRepository
├── ProyectoRepository
├── SubModalidadRepository
├── TipoIdentificacionRepository
└── TipoProductoRepository
```

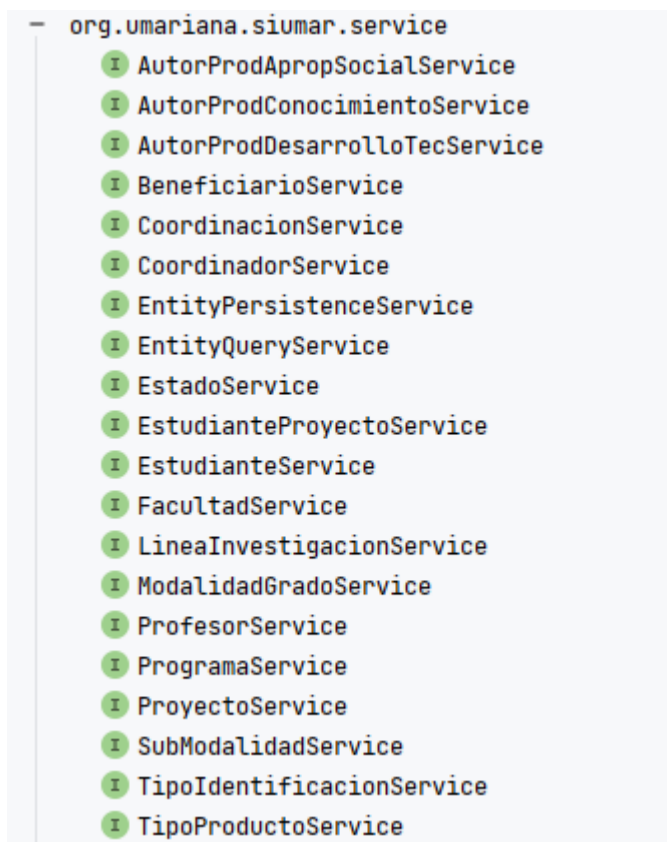
▪ Servicios

Los servicios (Figura 23) permitieron implementar la lógica de negocio de manera modular. Cada servicio corresponde a una entidad la cual permite su gestión o un conjunto de

funcionalidades como la exportación e importación de información o la generación de gráficos estadísticos.

Figura 23

Servicios para cada entidad



▪ Controladores

Para los controladores (Figura 24) se manejaron dos tipos de API'S, la primera API GraphQL utilizando la biblioteca graphql-java, estos controladores permiten la gestión de todas las entidades al igual que el filtrado de datos, para ello se utilizaron anotaciones como *@QueryMapping* y *@MutationMapping* que definen el comportamiento de los endpoints, la segunda API, REST que permite todo el manejo de información de archivos, en estos controladores se definieron los endpoint que permiten importar, exportar datos y generar gráficos estadísticos, se utilizó la anotación *@RestController* y las anotaciones básicas que soportan el protocolo HTTP (GET, POST, PUT, PATCH, DELETE).

Figura 24

Controladores

```
org.umariana.siumar.controller
├── AutorProdApropSocialController
├── AutorProdConocimientoController
├── AutorProdDesarrolloTecController
├── BeneficiarioController
├── CoordinacionController
├── CoordinadorController
├── EstadoController
├── EstudianteController
├── EstudianteProyectoController
├── FacultadController
├── LineaInvestigacionController
├── ModalidadGradoController
├── ProfesorController
├── ProgramaController
├── ProyectoController
├── SubModalidadController
├── TipoIdentificacionController
└── TipoProductoController
```

▪ GraphQL

Para la integración de GraphQL con nuestro proyecto de SpringBoot se crearon diferentes archivos que permiten mapear las entidades GraphQL con las definidas con JPA, en el archivo `types.graphql` se crearon todas estas entidades que serán gestionadas, el archivo `enums.graphql` mapea los enumeradores que definen opciones estáticas para un atributo, y el archivo `schema.graphql` (Figura 25) mapea los controladores de SpringBoot, en este punto los nombres de los controladores, parámetros y tipos de retorno de datos (Figura 26) deben ser iguales en ambos archivos.

Figura 25

Controlador GraphQL para proyectos

```
# Proyecto
type Query {
  """
  Retorna todos los proyectos registrados en el sistema
  """
  @return Lista de proyectos

  """
  buscarProyectos(filtros: [FiltroInput]): [Proyecto]
  """
  Busca un proyecto dado su id
  """
  @param id - Identificador del proyecto a buscar
  @return El proyecto con el id pasado por parámetro
  """
  buscarProyecto(id: ID): Proyecto
}

type Mutation {
  registrarProyecto(nuevoProyecto: ProyectoInput): Proyecto
  actualizarProyecto(proyecto: ProyectoInput): Proyecto
  eliminarProyecto(id: ID): Proyecto
}
```

Figura 26

Controlador SpringBoot para proyectos

```
@Controller
public class ProyectoController {

    @Autowired
    ProyectoService service;

    @QueryMapping
    public List<Proyecto> buscarProyectos(@Argument List<FiltroDTO> filtros) {
        return this.service.buscarPorFiltros(filtros);
    }

    @QueryMapping
    public Optional<Proyecto> buscarProyecto(@Argument Long id) { return this.service.buscarPorId(id); }

    @MutationMapping
    public Proyecto registrarProyecto(@Argument Proyecto nuevoProyecto) { return this.service.guardar(nuevoProyecto); }

    @MutationMapping
    public Proyecto actualizarProyecto(@Argument Proyecto proyecto) { return this.service.actualizar(proyecto); }

    @MutationMapping
    public void eliminarProyecto(@Argument Long id) { this.service.eliminar(id); }
}
```

- **Monolito a microservicios**

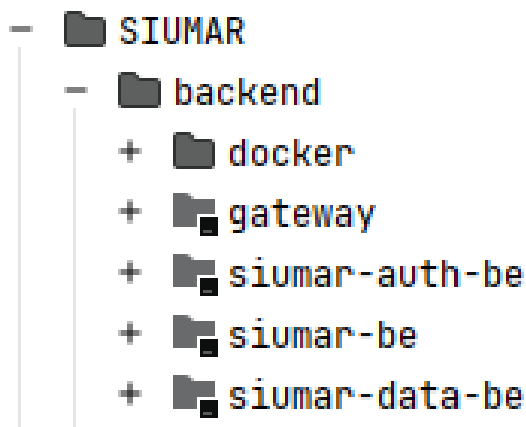
Al descubrir la coexistencia de dos APIs distintas en el BackEnd, GraphQL y REST, se destacó la importancia de preservar la independencia entre estos componentes. Esta observación ha impulsado la consideración de migrarlos a diferentes microservicios. GraphQL presenta la capacidad de solicitar únicamente los datos necesarios, reduciendo la sobrecarga de información no utilizada. Por otro lado, REST, con sus solicitudes tradicionales basadas en verbos HTTP, facilita el manejo de archivos y gráficos.

Adicionalmente, se incorpora un microservicio externo dedicado a la autenticación con la plataforma mediante OAuth 2.0 y la API de Google.

Para la comunicación entre los microservicios a nivel de BackEnd, se utilizó la clase **RestTemplate** y **HttpGraphQLClient** las cuáles permiten definir los parámetros de conexión y obtener los resultados de las peticiones hacia los demás servicios.

Figura 27

Microservicios y Gateway



- **Configuración de Gateway**

Para la configuración del Gateway se implementó **Spring Cloud Gateway** ya que ofrece una solución efectiva para gestionar las solicitudes entrantes y simplificar la comunicación entre el FrontEnd y los microservicios en una arquitectura de microservicios.

En el proceso de configuración del Gateway del aplicativo, se ha establecido un enrutamiento para dirigir las solicitudes entrantes a los microservicios correspondientes. A continuación, se describen los principales aspectos de esta configuración:

- **Microservicio de Gestión de Entidades (Puerto 8081):** Este microservicio se encarga de gestionar todas las entidades principales de nuestra aplicación. Se encuentra disponible en la ruta: *http://siumar:8085/api/gestion*.
- **Microservicio de Gestión de Archivos (Puerto 8082):** Este microservicio está especializado en la gestión de archivos (importación y exportación) y datos relacionados (generación de gráficos estadísticos). Desde el cliente, se puede acceder a través de la ruta: *http://siumar:8085/api/files/data*.
- **Microservicio de Autenticación (Puerto 8083):** El microservicio de autenticación se encarga de la seguridad y la autenticación de usuarios en la aplicación. Los usuarios pueden interactuar con este servicio utilizando la ruta: *http://siumar:8085/api/auth*.

Esta configuración (Figura 28) permite que las solicitudes dirigidas al Gateway se distribuyan de manera eficiente a los microservicios correspondientes, garantizando así un funcionamiento fluido y seguro de la aplicación. Cada microservicio cumple un rol específico y se encuentra disponible en su ruta respectiva, lo que facilita su acceso y utilización por parte de los clientes y usuarios de la aplicación.

Figura 28

Enrutamiento microservicios desde el Gateway

```
@Configuration
public class SpringCloudConfig {
    @Bean
    public RouteLocator customRouteLocator(RouteLocatorBuilder builder) {
        return builder.routes()
            .route("gestion", r -> r.path("/api/gestion/graphql")
                .filters(f -> f.rewritePath("/api/gestion", "/"))
                .uri("http://localhost:8081"))
            .route("fileData", r -> r.path("/api/data/**")
                .filters(f -> f.rewritePath("/api/data", "/"))
                .uri("http://localhost:8082"))
            .route("auth", r -> r.path("/api/auth/**")
                .filters(f -> f.rewritePath("/api/auth", "/"))
                .uri("http://localhost:8083"))
            .build();
    }
}
```

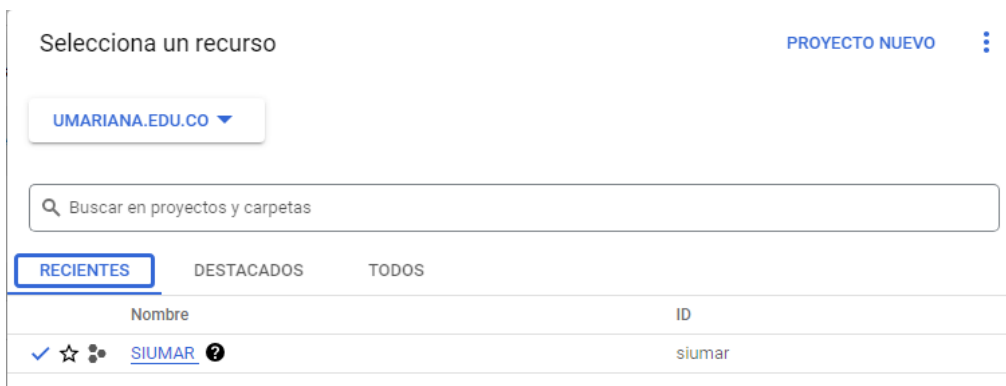
▪ Configuración microservicio autenticación con Google

Para la autenticación de la plataforma se utilizó la API OAuth2 de Google el cuál utiliza la especificación *OpenID Connect 1.0*

Con el fin de interactuar con los servicios de Google, se creó un recurso en Google Cloud denominado "Umariana" (Figura 29). Este recurso proporciona la infraestructura necesaria para la gestión segura de la autenticación.

Figura 29

Recurso de la plataforma en Google Cloud



Se llevó a cabo la creación de un ID OAuth (Figura 30) específico para la aplicación, desempeñando un papel fundamental en el proceso de obtención de tokens de acceso. Estas credenciales únicas identifican de manera inequívoca la aplicación ante Google, estableciendo así una conexión segura y autorizada.

Figura 30

ID Cliente OAuth

ID de clientes OAuth 2.0

<input type="checkbox"/>	Nombre	Fecha de creación ↓	Tipo	ID de cliente
<input type="checkbox"/>	siumar-web-app	25 oct 2023	Aplicación web	1050148638204-pco... 🗑️

En el ámbito del desarrollo, se configuró el proyecto de autenticación utilizando Spring Boot. Las credenciales del cliente OAuth2 recién creadas fueron integradas en el sistema, permitiendo así una interacción fluida con la API de Google.

Se diseñó un endpoint en el controlador del proyecto (Figura 31) para exponer el proceso de autenticación. Este endpoint facilita la obtención del token de autenticación y, simultáneamente, valida la sesión del usuario.

Figura 31

Controlador útil para la validación del token

```
+ org.umariana.siumarauthbe
- org.umariana.siumarauthbe.controller
  |
  |   AutenticacionController
+ org.umariana.siumarauthbe.dto
- org.umariana.siumarauthbe.utils
  |
  |   ValidarTokenUtil
```

Además, se implementó un mecanismo de almacenamiento en caché para guardar tanto el JSON Web Token como la información del usuario. Esta estrategia no solo optimiza el rendimiento, sino que también contribuye a la eficiencia del sistema.

Para gestionar adecuadamente la duración de las sesiones de usuario, se incorporaron validaciones de expiración. Estas verificaciones aseguran la coherencia temporal de las sesiones, permitiendo decisiones informadas sobre el cierre o mantenimiento de las mismas.

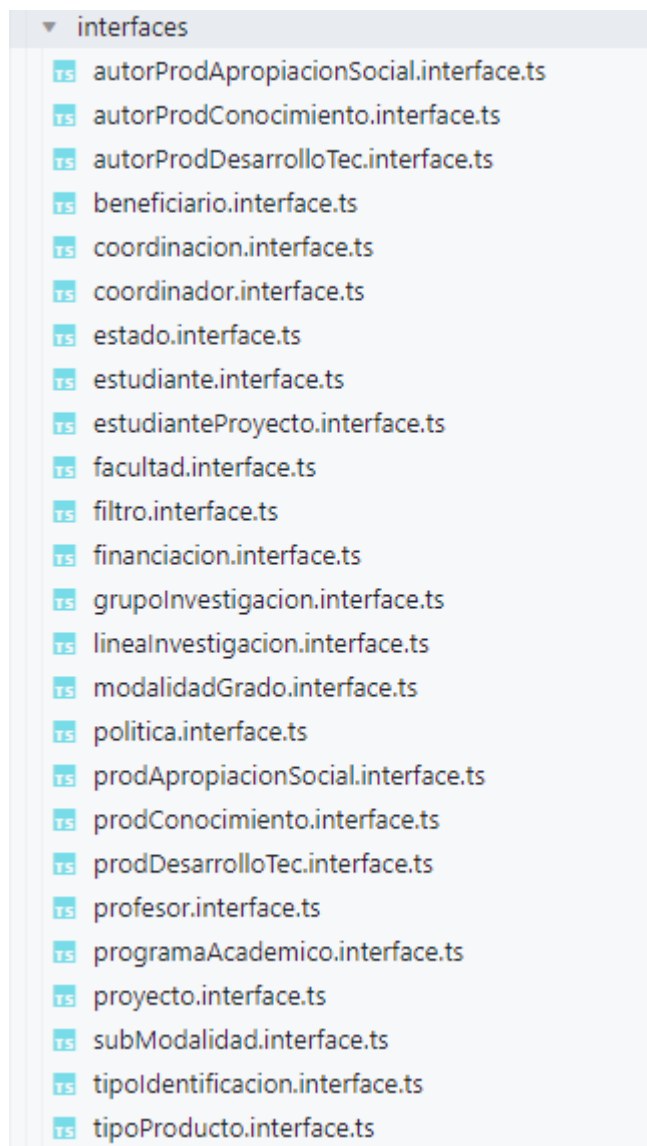
2.2.3.2. FrontEnd

▪ Interfaces de Entidades

Se crearon interfaces que representan las entidades definidas en el BackEnd (Figura 32). Estas interfaces actúan como contratos que definen la estructura de los datos de las entidades de dominio.

Figura 32

Interfaces de entidades del BackEnd

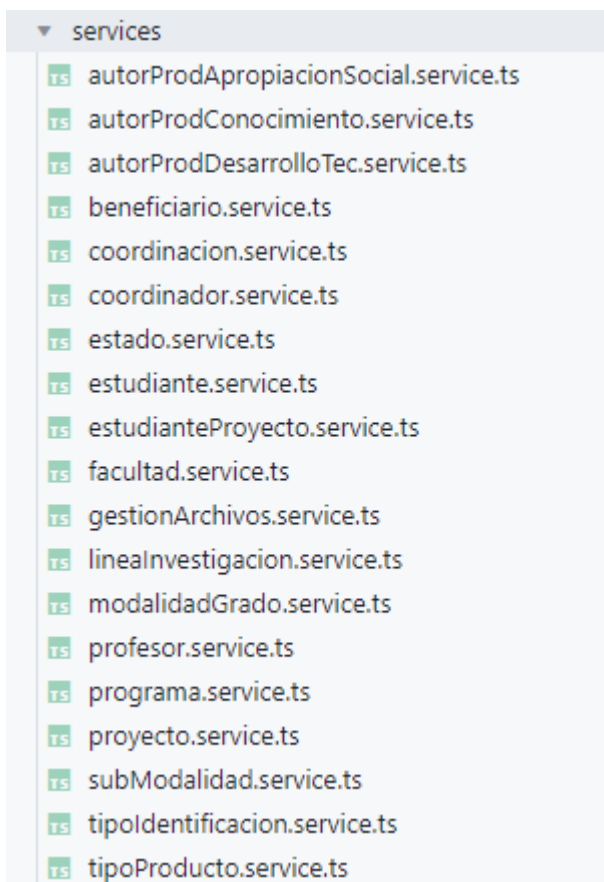


▪ **Servicios Apollo y HTTP**

Se implementaron servicios que se encargan de comunicarse con el BackEnd (Figura 33). Los servicios Apollo-Angular se utilizan para interactuar con los controladores de la API GraphQL, mientras que los servicios HTTP se utilizan para consumir los controladores de la API REST.

Figura 33

Servicios Apollo y HTTP para comunicación con el BackEnd



▪ Definición de Endpoints GraphQL

Se utilizó Apollo-Angular para interactuar con los servicios GraphQL, permite definir las consultas y mutaciones (Figura 34) en forma de tipos TypeScript fuertemente tipados. Esto garantiza una verificación en tiempo de compilación y evita errores relacionados con el formato de las consultas, también se encarga de la gestión del estado de los datos recuperados del BackEnd. Ofrece un sistema de caché eficiente que mejora la eficacia de la aplicación y evita solicitudes duplicadas al servidor.

Se definieron cada uno de los endpoints GraphQL que serán consumidos por la aplicación (Figura 35). En este punto, se especificaron las operaciones, parámetros y estructura de datos necesarios para cada interacción con el BackEnd.

Figura 34

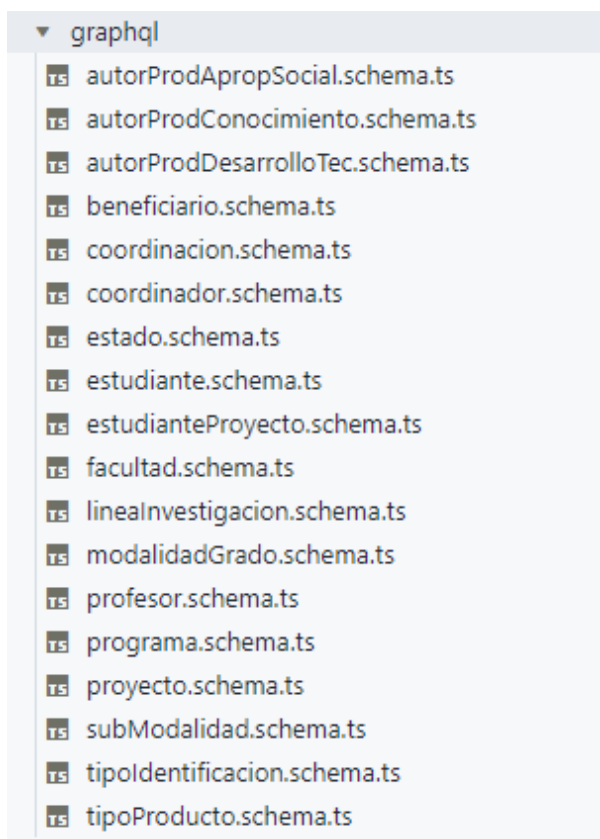
Mutaciones y consultas de GraphQL

```
export const buscarTipoProductoPorId = gql`
  query buscarTipoProductoPorId($id: ID!) {
    buscarTipoProducto(id: $id) {
      id
      nombre
      categoria
    }
  }
`;

export const registrarTipoProducto = gql`
  mutation registrarTipoProducto($tipoProducto: TipoProductoInput!) {
    registrarTipoProducto(nuevoTipoProducto: $tipoProducto) {
      id
      nombre
      categoria
    }
  }
`;
```

Figura 35

Operaciones GraphQL para cada entidad

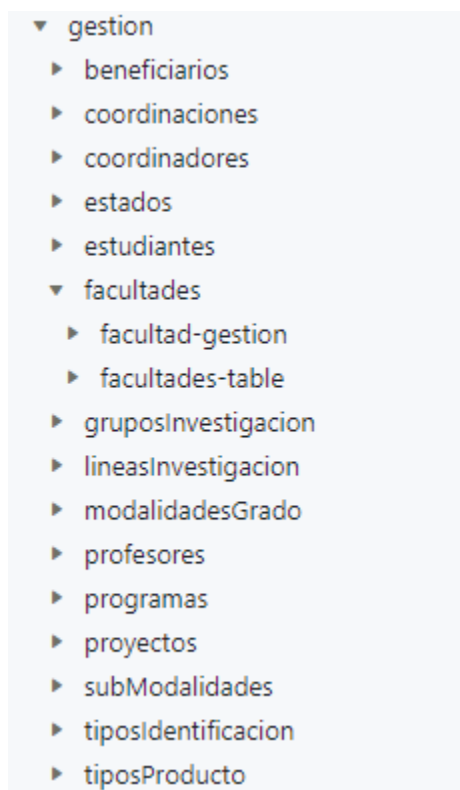


▪ **Gestión de Componentes**

Se creó un componente para permitir la gestión de cada una de las entidades (Figura 36), los componentes se dividen en dos partes, uno llamado “table” el cual indica los registros de la entidad en una tabla y cuenta con operaciones para registrar, editar o eliminar, además permite el filtrado de datos. Por otra parte, está el componente “gestión” el cuál es un modal con el formulario necesario para registrar o editar un registro.

Figura 36

Creación de componente para gestión de entidades



2.2.4. Etapa 4. Pruebas

Para la realización de las pruebas unitarias se crearon escenarios de prueba (Anexo F) para cada historia de usuario y los casos de prueba (Anexo G) para validar los criterios de aceptación, a continuación, se indican las tablas creadas para la implementación de los tests.

- **Resultados de las pruebas**

Durante el proceso de pruebas unitarias, se logró alcanzar una cobertura del 100% del código (Anexo H). Todas las secciones del proyecto, incluyendo los controladores, fueron exhaustivamente evaluadas. Además, se garantizó que todas las entidades fueron instanciadas y sometidas a pruebas con datos específicos para verificar su correcto funcionamiento.

Se observó un resultado positivo en la mayoría de los casos, ya que todas las pruebas fueron superadas con éxito. Sin embargo, se identificaron situaciones específicas en las pruebas de creación y actualización de Proyecto que resultaron en fallos. Tras un análisis detenido, se determinó que el problema residía en la implementación incorrecta del scalar de tipo Date en el bean de configuración correspondiente.

Una vez identificado este inconveniente, se procedió a corregir la implementación del bean de configuración del scalar de tipo Date. Después de la corrección, las pruebas de creación y actualización de Proyecto se ejecutaron satisfactoriamente.

Además, el tiempo de ejecución de las pruebas unitarias se optimizó significativamente mediante la implementación de mocks para representar los datos durante las pruebas. Este enfoque estratégico eliminó la necesidad de consultar la base de datos en cada iteración de las pruebas, permitiendo iteraciones rápidas y facilitando la identificación y resolución de problemas de manera oportuna.

En general, los resultados de las pruebas unitarias revelan una sólida calidad del código en el proyecto. La cobertura completa del 100% indica un esfuerzo metódico en la evaluación exhaustiva de todas las secciones, proporcionando una base robusta para la fiabilidad y estabilidad del sistema.

Aunque se identificaron problemas específicos en las pruebas de creación y actualización de Proyecto, la capacidad para abordar y corregir rápidamente estos problemas resalta la agilidad y la dedicación hacia la mejora continua.

La optimización del tiempo de ejecución mediante la implementación de mocks refleja una estrategia efectiva para iteraciones rápidas y una resolución ágil de problemas, contribuyendo así a la eficiencia del proceso de desarrollo.

En conjunto, estos resultados sugieren un enfoque sólido y comprometido hacia la calidad del código, respaldado por prácticas de prueba efectivas y una capacidad de respuesta efectiva ante desafíos identificados.

- **Pruebas E2E**

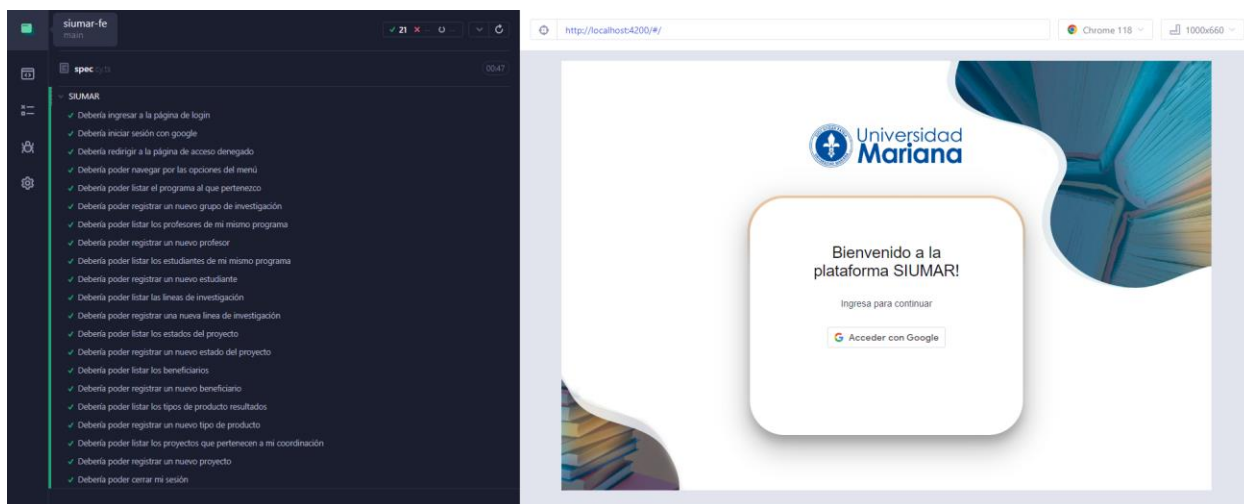
En las pruebas de extremo a extremo, se empleó el framework de JavaScript Cypress, una elección estratégica que permitió la configuración de un entorno de pruebas en tiempo real. Esto posibilitó la verificación del flujo de la aplicación mediante la ejecución de acciones directamente desde un navegador web, abarcando interacciones con elementos visuales, redirección de rutas, y la obtención y gestión de datos de la base de datos a través de los endpoints disponibles en los controladores de los microservicios.

El uso de Cypress proporcionó una visión integral y precisa del rendimiento de la aplicación, permitiendo cubrir aspectos cruciales como la interacción visual, la manipulación de rutas y el manejo eficiente de datos en la base de datos.

Para estructurar las pruebas, se definieron escenarios específicos (Anexo I) que encapsulan de manera representativa los flujos de trabajo en un entorno real. Estos escenarios proporcionan un marco detallado para evaluar la funcionalidad y la respuesta del sistema en situaciones diversas.

Figura 37

Resultados pruebas E2E en entorno de testing Cypress



▪ Resultados de las pruebas

Durante el proceso de pruebas end-to-end (E2E), se logró una cobertura considerable de los flujos de trabajo esenciales de la aplicación. Un flujo de trabajo clave que abarca las acciones críticas para un coordinador de investigación fue exhaustivamente probado. Esto incluye la capacidad de iniciar sesión, trabajar con información consolidada mediante operaciones de edición, adición, eliminación y filtrado de datos. Además, se examinaron a fondo los aspectos de seguridad del flujo, evaluando la autenticación, los privilegios asociados con las opciones del menú y el cierre de sesión.

Se obtuvieron resultados altamente positivos, ya que todas las pruebas fueron superadas de manera exitosa. Este logro se atribuye a la correcta configuración de las rutas del Gateway, lo cual permitió un flujo óptimo de emisión y recepción hacia los controladores.

Este éxito se vio respaldado aún más por la realización de pruebas unitarias exhaustivas en cada controlador encargado de la gestión de entidades. La implementación de estas pruebas unitarias contribuyó a evitar errores en las operaciones CRUD (Crear, Leer, Actualizar, Eliminar).

Cabe destacar que, las pruebas fueron ejecutadas de manera eficiente, contribuyendo a un proceso de desarrollo ágil y sin ralentizaciones significativas. Este rendimiento optimizado se logró gracias a dos aspectos fundamentales. En primer lugar, la utilización efectiva de un sistema de almacenamiento en caché para los datos desempeñó un papel clave. Este enfoque permitió evitar la necesidad de recuperar la información nuevamente en cada ejecución de prueba, acelerando significativamente el tiempo de ejecución al aprovechar datos previamente guardados en caché. En segundo lugar, la implementación de la API GraphQL desempeñó un papel crucial al permitir la obtención específica de los datos necesarios. En lugar de transmitir todos los objetos registrados, la API GraphQL posibilitó la recuperación precisa de datos específicos, reduciendo así la carga de datos transmitidos y optimizando aún más el tiempo de ejecución de las pruebas.

En un enfoque de calidad, la cobertura exhaustiva de los flujos de trabajo esenciales, especialmente aquellos críticos para un coordinador de investigación, refleja una sólida implementación de las operaciones clave. La capacidad de iniciar sesión, realizar operaciones CRUD en datos consolidados y gestionar la seguridad, incluyendo la autenticación y los privilegios, ha sido validada de manera exitosa. Además, la obtención de resultados altamente positivos en todas las pruebas sugiere una calidad robusta en la implementación. La correcta configuración de las rutas del Gateway y la ejecución eficiente de las pruebas unitarias en los controladores indican una atención detallada a la integridad y la funcionalidad del código. La utilización efectiva de almacenamiento en caché y la implementación estratégica de la API GraphQL para obtener datos específicos contribuyen a un rendimiento ágil y eficiente durante la ejecución de las pruebas.

Finalmente, según los resultados positivos de las pruebas E2E, se puede concluir que la aplicación ofrece una experiencia de usuario altamente intuitiva y fácil de usar. La cobertura exhaustiva de los flujos de trabajo esenciales, incluyendo operaciones clave como iniciar sesión, gestionar datos consolidados y garantizar la seguridad, respalda la eficacia de la interfaz de usuario. La falta de fallos en las pruebas y la ejecución exitosa de todas las operaciones CRUD sugieren que no hay pasos en los flujos de trabajo que sean confusos o poco intuitivos. La ausencia de problemas significativos en la simulación de la experiencia del usuario indica que los usuarios deben encontrar la aplicación intuitiva y amigable, facilitando la realización de tareas clave.

2.2.5 Construcción y despliegue imágenes Docker

▪ Construcción imágenes Docker de los artefactos de la plataforma

Para el despliegue de la base de datos, se definió un servicio de PostgreSQL en el archivo docker-compose, se configuraron sus variables de entorno para levantar la conexión y los volúmenes que permiten persistir la información en el propio contenedor, mediante estas rutas se cargaron los scripts SQL que crean las tablas y esquemas de la base de datos junto con los datos por defecto obtenidos del análisis de las matrices.

Para cada microservicio, se crea el archivo Docker file (Figura 38) el cual contiene una serie de instrucciones que permite construir una imagen Docker a través del JAR compilado del proyecto, las imágenes dockerizadas correrán la versión 17 de Java en el puerto que se especifica en cada uno de los archivos.

Figura 38

Archivo para despliegue del microservicio a una imagen Docker

```
Dockerfile x
1  FROM openjdk:17-jdk-alpine as builder
2
3  WORKDIR /app
4
5  COPY ./mvn ./mvn
6  COPY ./mvnw .
7  COPY ./pom.xml .
8
9  RUN ./mvnw clean package -Dmaven.test.skip -Dmaven.main.skip -Dspring-boot.repackage.skip && rm -r ./target/
10
11 COPY ./src ./src
12
13 RUN ./mvnw clean package -DskipTests
14
15 FROM openjdk:17-jdk-alpine
16
17 WORKDIR /app
18
19 COPY --from=builder /app/target/siumar-0.0.1-SNAPSHOT.jar .
20
21 EXPOSE 8081
22
23 ENTRYPOINT ["java", "-jar", "siumar-0.0.1-SNAPSHOT.jar"]
```

Nota: Cada microservicio y el Gateway tiene su archivo Docker file independiente para crear la imagen específica.

Para el FrontEnd, se crea la imagen Docker (Figura 39) con el directorio “dist” obtenido a través de la compilación del proyecto Angular, este directorio contiene archivos estáticos de HTML, CSS y JavaScript que pueden ser ejecutados por los navegadores web; la imagen correrá la versión 16 de Node JS junto con un servidor Nginx para cargar los archivos estáticos en el puerto 80.

Figura 39

Archivo para el despliegue del FrontEnd a una imagen Docker

```
Dockerfile > ...
1 FROM node:16-alpine AS build
2
3 WORKDIR /app
4
5 COPY . . .
6
7 RUN npm install
8
9 RUN npm run build
10
11 FROM nginx:alpine
12
13 COPY --from=build /app/dist/siumar-fe /usr/share/nginx/html
14
15 EXPOSE 80
```

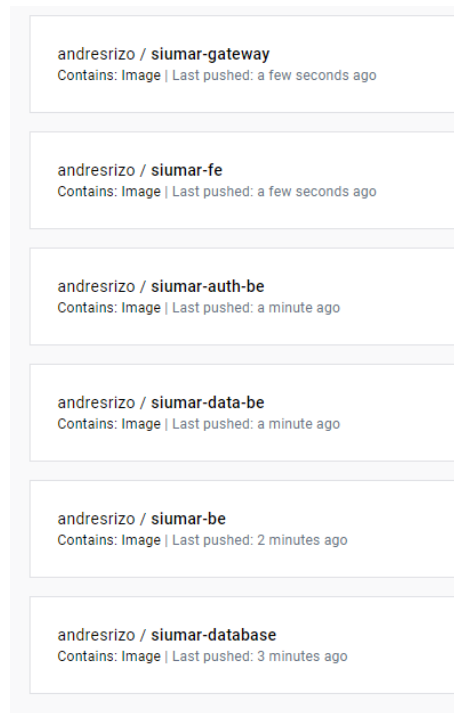
Al compilar todos los artefactos en imágenes Docker, fue necesario cambiar las direcciones “localhost” de todos los proyectos por el nombre correspondiente del servicio Docker creado al que referencia, de esta manera se establece un flujo correcto en la comunicación entre las imágenes Docker.

▪ Despliegue de las imágenes en Docker Hub

Se desplegaron las imágenes Docker compiladas en repositorios de Docker Hub (Figura 40) de esta manera se pueden descargar y configurar en un archivo Docker compose o en plataformas cloud como AWS.

Figura 40

Repositorios de Docker Hub



▪ Creación servicio ECS en AWS

Se configuró un cluster (Figura 41) el cuál funciona como una red general donde se comunican todos los servicios, estos ejecutan las tareas que contienen los contenedores con las imágenes de los artefactos desplegados.

Figura 41

Cluster ECS



Por cada una de las imágenes Docker se creó una tarea (Figura 42) definiendo los recursos necesarios para una óptima ejecución, de igual forma se configuraron variables de entorno, mapeo de puertos y URL de las imágenes publicadas en Docker Hub.

Figura 42

Configuración contenedores en tareas

The screenshot shows the configuration page for a container named 'Contenedor: 1'. It includes sections for 'Detalles del contenedor', 'Registro privado', and 'Mapeos de puertos'. The 'Detalles del contenedor' section has input fields for 'Nombre' (msv-siumar-auth), 'URI de imagen' (andresrizo/siumar-auth-be:latest), and a 'Contenedor esencial' dropdown set to 'Sí'. The 'Mapeos de puertos' section has a table with columns for 'Puerto del contenedor', 'Protocolo', 'Nombre del puerto', and 'Protocolo de la aplicación'. One mapping is shown: 8083, TCP, msv-siumar-auth-8083-tr, HTTP. There are 'Eliminar' buttons for both the container and the port mapping.

Figura 43

Tareas para cada imagen Docker

The screenshot shows the 'Definiciones de tareas (6)' page. It features a search bar and a table with columns for 'Definición de tarea' and 'Estado de la última revisión'. The table lists six tasks, all of which are in an 'ACTIVE' state.

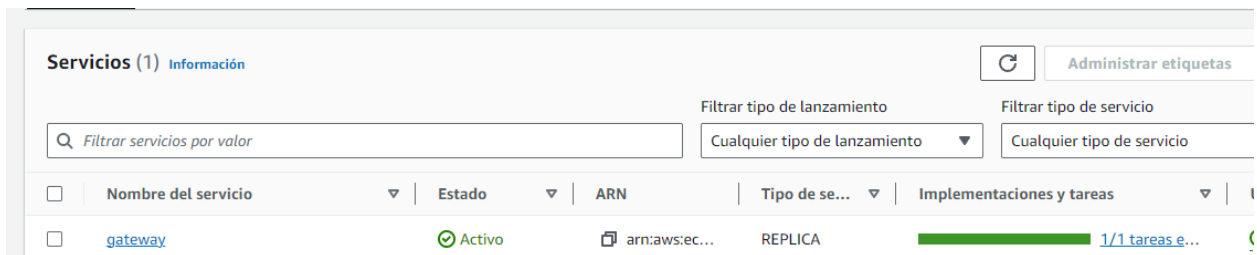
Definición de tarea	Estado de la última revisión
msv-gateway	ACTIVE
msv-siumar	ACTIVE
msv-siumar-auth	ACTIVE
msv-siumar-data	ACTIVE
siumar-database	ACTIVE
siumar-fe	ACTIVE

Por último, se configuraron los servicios (Figura 44) para cada tarea la cuál crea direcciones ip públicas que permiten su ejecución, se modificaron en las variables de entorno las IP'S específicas

que necesitan los microservicios permitiendo así una comunicación exitosa entre los contenedores desplegados en el cluster SIUMAR del servicio ECS.

Figura 44

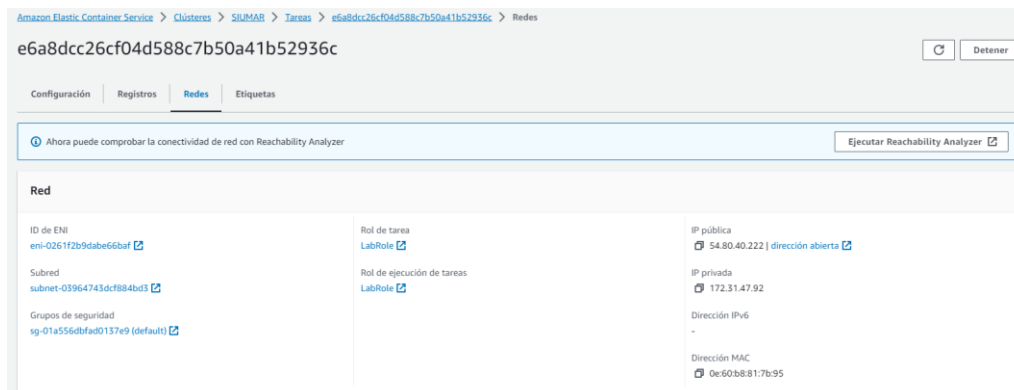
Lanzamiento de servicios



Para las pruebas del aplicativo desplegado en producción, se accede a los detalles del servicio que ejecuta la tarea del FrontEnd (Figura 45) y se identifica la ip pública que expone la red, a través de esta ip se logra el acceso a la plataforma web SIUMAR (Anexo J) .

Figura 45

IP pública del servicio FrontEnd

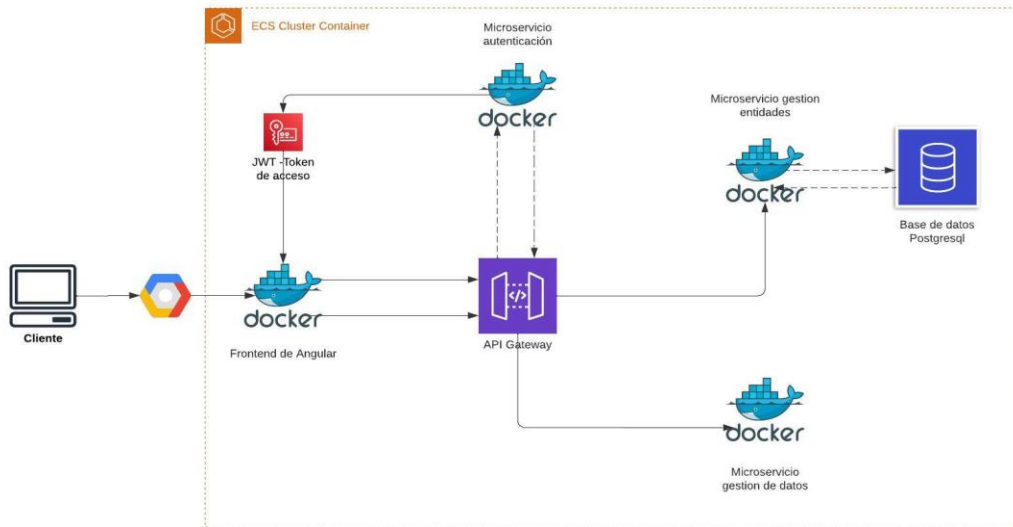


A través del desarrollo integral de la plataforma, que incluyó la implementación de microservicios, la configuración de la base de datos, la codificación del FrontEnd, la realización de pruebas exhaustivas y el despliegue del aplicativo, se logró identificar con éxito la arquitectura de la aplicación. Este proceso permitió una comprensión profunda de cómo interactúan los

diferentes componentes y cómo contribuyen al funcionamiento global de la aplicación. La arquitectura resultante (Figura 46) proporciona una base sólida para futuras mejoras y expansiones.

Figura 46

Arquitectura de la plataforma



2.3. Evaluación de la plataforma

Se llevó a cabo un recurso audiovisual que muestra el funcionamiento de la aplicación. Este material fue compartido con actores clave involucrados en la gestión de la actividad educativa estudiantil, incluyendo:

- Juan Pablo García López, Coordinador de Investigación Estudiantil de la Universidad Mariana.
- Giovanni Albeiro Hernández Pantoja, Coordinador de Investigación del Programa de Ingeniería de Sistemas.
- Gustavo Willyn Sánchez Rodríguez, Coordinador de Semilleros de Investigación del Programa de Ingeniería de Sistemas.
- Alejandra Zuleta Medina, quien anteriormente ocupó el cargo de Coordinadora de Investigación Estudiantil en la Universidad Mariana.

- Javier Mauricio López Moreno, Coordinador de Co-Investigación Estudiantil del Programa de Ingeniería de Sistemas.
- Carlos Gerardo Enríquez Ordoñez, Coordinador de Investigación del Programa de Administración de Negocios Internacionales.
- Nubia Del Rosario González Martínez, Coordinadora de Semilleros de Investigación del Programa de Administración de Negocios Internacionales.
- Teresita del Rocío Cánchala Nastar, Coordinadora de Investigación del Programa de Ingeniería Ambiental.

Se presentó y demostró la plataforma de gestión de la actividad educativa estudiantil mediante un ejemplo detallado de todo el proceso. Posteriormente, se distribuyó un formulario de evaluación que constaba de tres secciones principales, cada una centrada en aspectos clave de la plataforma web. Estas secciones incluían dos criterios de calidad fundamentales: la Usabilidad de la plataforma web y la Funcionalidad de la misma. Además, se incorporaron preguntas específicas sobre la optimización del proceso de gestión.

Para recopilar la retroalimentación de manera estructurada, se implementó una metodología de medición conocida como "escala de Likert". Esta escala, que utiliza cinco niveles de medición para cada afirmación o pregunta, permitió a los coordinadores expresar sus evaluaciones de la plataforma de manera precisa, desde "Totalmente de acuerdo" hasta "Totalmente en desacuerdo".

Adicionalmente, se incluyó un espacio dedicado a observaciones en cada sección del formulario, brindando a los coordinadores la oportunidad de ofrecer comentarios detallados y sugerencias específicas.

▪ **Sección inicial de evaluación centrada en la Usabilidad de la Plataforma web**

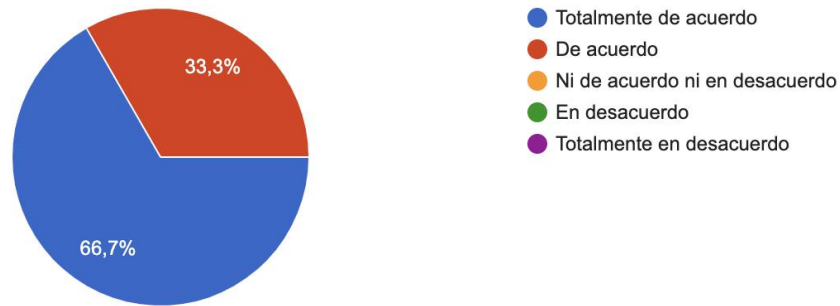
Figura 47

Representación gráfica de la primera evaluación

La plataforma web es fácil de navegar

 Copiar

9 respuestas



Según los resultados obtenidos, se evidencia que la mayoría del personal percibe el software desarrollado de manera positiva, con un alto nivel de aceptación (66.7%). Esto indica que la plataforma web cumple satisfactoriamente con sus objetivos y demuestra una funcionalidad efectiva. Sin embargo, un porcentaje de la población (33.3%) se muestra sólo de acuerdo. En respuesta, se llevará a cabo una revisión de la plataforma, tomando en consideración las observaciones proporcionadas por los participantes en la encuesta. El objetivo es identificar áreas específicas que puedan no estar funcionando de manera óptima y abordar cualquier problema relevante para mejorar la experiencia general.

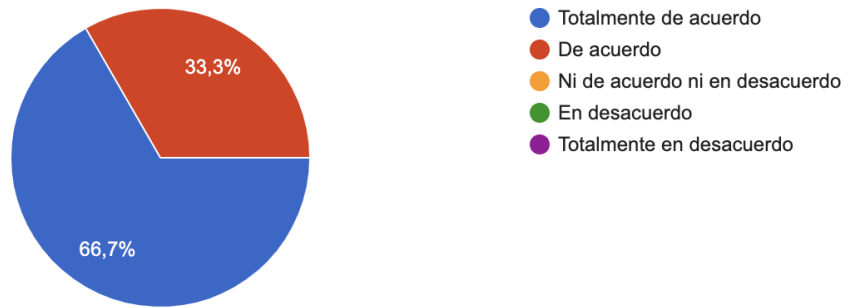
Figura 48

Representación gráfica de la segunda evaluación

La plataforma web presenta la información de forma clara

 Copiar

9 respuestas



Se puede evidenciar con el gráfico anterior que la forma de representación de los datos incluyendo los gráficos y formularios están bien definidos y ayudan al usuario a entender y obtener fácilmente resultados y conclusiones con la estructura de la información presentada.

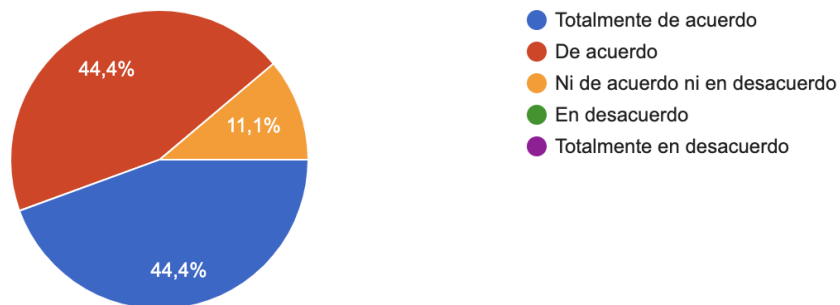
Figura 49

Representación gráfica de la tercera evaluación

Los mensajes de advertencia en la plataforma son fáciles de entender

 Copiar

9 respuestas



En general, este gráfico indica que los mensajes de advertencia que se muestran al realizar distintas acciones de la plataforma tales como eliminar, borrar o registrar son entendibles. A pesar de ello se identifica que la mayoría de los encuestados (55,5%) no están totalmente de acuerdo, significa que es necesario detallar más la información para aumentar la claridad de acciones ejecutadas.

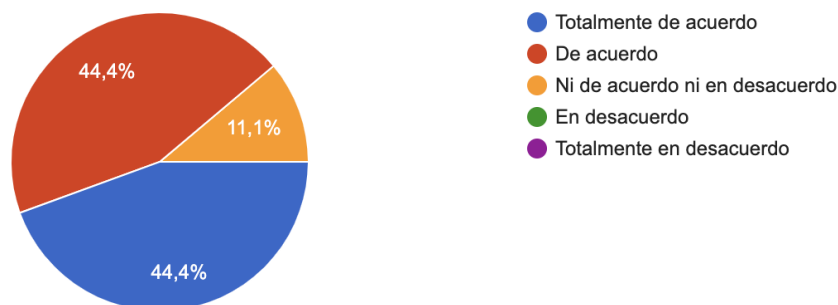
Figura 50

Representación gráfica de la cuarta evaluación

Las notificaciones en la plataforma son claras y comprensibles



9 respuestas



La evaluación refleja una respuesta equilibrada, con un 44.4% de participantes "Totalmente de acuerdo", un 44.4% "De acuerdo" y un 11.1% "Ni de acuerdo ni en desacuerdo" respecto a la claridad y comprensibilidad de las notificaciones en la plataforma. Aunque la mayoría de los participantes perciben las notificaciones como claras y comprensibles, es esencial abordar las sugerencias o inquietudes expresadas por el 11.1% que se encuentra en el medio. Agradecemos la aceptación positiva de la mayoría y nos comprometemos a revisar y mejorar cualquier aspecto que haya generado dudas o indecisiones en este grupo específico. Este enfoque asegura una comunicación efectiva y una experiencia del usuario optimizada.

La retroalimentación proporcionada destaca la importancia de una revisión continua para adaptarse a las necesidades y expectativas cambiantes de los usuarios. Además, se subraya la relevancia de una comunicación detallada y clara en todos los aspectos de la plataforma, desde

mensajes de advertencia hasta notificaciones, para garantizar una experiencia integral y satisfactoria para todos los usuarios.

▪ **Segunda sección de evaluación orientada a la Funcionalidad de la Plataforma web**

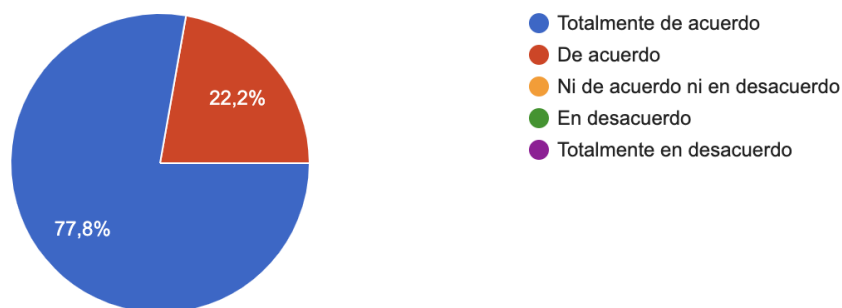
Figura 51

Representación gráfica de la quinta evaluación

La plataforma web presenta un buen tiempo de respuesta en sus funciones



9 respuestas



La evaluación refleja una respuesta positiva y significativa, con un 77.8% de participantes "Totalmente de acuerdo" y un 22.2% "De acuerdo" en lo que respecta al eficiente tiempo de respuesta de la plataforma web. Este respaldo resalta no solo la percepción positiva de los usuarios respecto a la agilidad de la plataforma, sino también su impacto directo en la eficiencia general y la experiencia positiva del usuario. Este hallazgo refuerza la idea de que la plataforma no solo cumple con sus funciones, sino que lo hace de manera eficaz y rápida, contribuyendo así a la optimización del proceso de gestión de la actividad investigativa estudiantil en la Universidad Mariana.

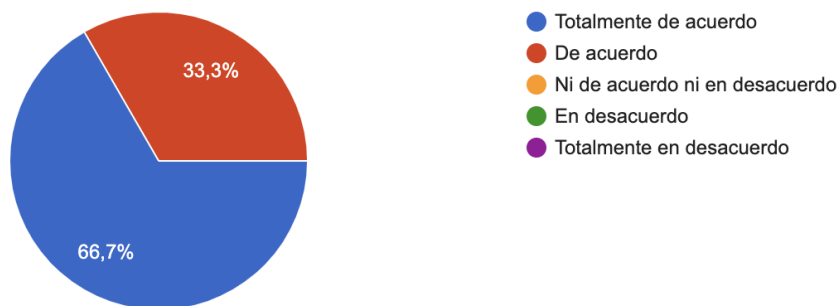
Figura 52

Representación gráfica de la sexta evaluación

La plataforma es eficiente (realiza o cumple adecuadamente sus funciones)

 Copiar

9 respuestas



La evaluación refleja una respuesta positiva, con un 66.7% de participantes "Totalmente de acuerdo" y un 33.3% "De acuerdo" en cuanto a la eficiencia de la plataforma al cumplir adecuadamente sus funciones. Estos resultados indican de manera clara que la plataforma no solo satisface las expectativas, sino que lo hace de manera eficiente según la percepción de los usuarios. En reconocimiento a este respaldo, se destaca la eficiencia como un atributo distintivo de la plataforma, reforzando su papel efectivo en la gestión de la actividad investigativa estudiantil.

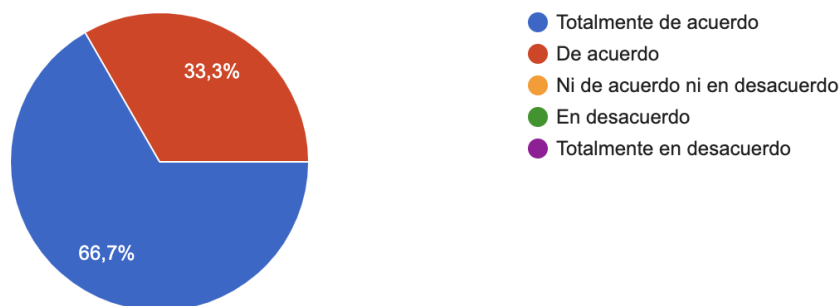
Figura 53

Representación gráfica de la séptima evaluación

La plataforma es eficaz (logra sus objetivos previstos y genera los resultados esperados)



9 respuestas



La sólida respuesta de los participantes, con un 66.7% totalmente de acuerdo y un 33.3% de acuerdo respecto a la eficacia de la plataforma en lograr sus objetivos y generar resultados esperados, subraya de manera elocuente el éxito de la plataforma en el cumplimiento de sus metas propuestas. Esta destacada aceptación por parte de los usuarios refuerza la idea de que la plataforma no solo ha sido bien recibida, sino que también ha superado las expectativas al demostrar consistentemente su capacidad para alcanzar los objetivos establecidos y producir los resultados deseados. Este respaldo positivo refleja un logro significativo en el marco de la gestión de la actividad investigativa estudiantil, respaldando la eficacia de la plataforma como una herramienta valiosa en este contexto.

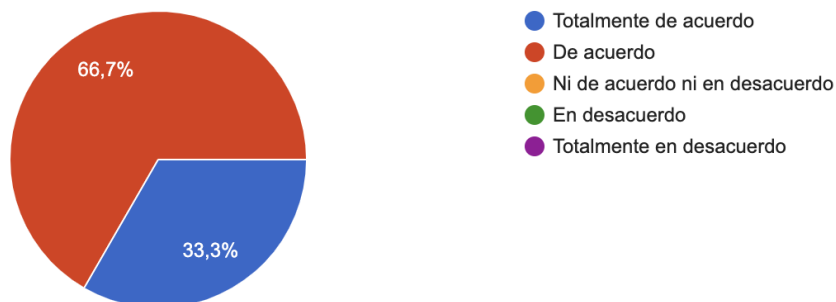
Figura 54

Representación gráfica de la octava evaluación

Sería beneficioso incorporar esta plataforma en el proceso de gestión de la actividad investigativa estudiantil



9 respuestas



La evaluación muestra una perspectiva favorable, ya que el 66.7% de los participantes está "De acuerdo" y el 33.3% está "Totalmente de acuerdo" con la idea de que incorporar la plataforma en el proceso de gestión de la actividad investigativa estudiantil sería beneficioso. Estos resultados indican un respaldo significativo a la idea de integrar la plataforma en el proceso existente. Para resaltar este respaldo positivo, se puede comunicar a los usuarios que expresaron acuerdo y total acuerdo la importancia de su perspectiva positiva y cómo sus opiniones respaldan la utilidad de la plataforma. Además, se pueden destacar las ventajas específicas que los participantes ven al incorporar la plataforma en el proceso de gestión, como eficiencia, accesibilidad o mejora general en la actividad investigativa estudiantil. Estos pasos ayudarán a resaltar y capitalizar la perspectiva positiva expresada por los participantes en relación con la incorporación de la plataforma.

En conjunto, estas evaluaciones reflejan un respaldo sustancial a la eficiencia, eficacia y utilidad de la plataforma en la gestión de la actividad investigativa estudiantil. La alta aceptación de la rapidez de respuesta, la eficiencia en el cumplimiento de funciones y la percepción positiva sobre la incorporación de la plataforma demuestran el impacto positivo de la herramienta en el proceso universitario. Estas observaciones respaldan la conclusión de que la plataforma no solo

cumple con sus funciones, sino que también supera las expectativas, consolidándose como un componente valioso en la gestión de la actividad investigativa estudiantil en la Universidad Mariana.

- **Tercera sección de evaluación que aborda preguntas sobre la optimización del proceso de gestión de actividad educativa estudiantil**

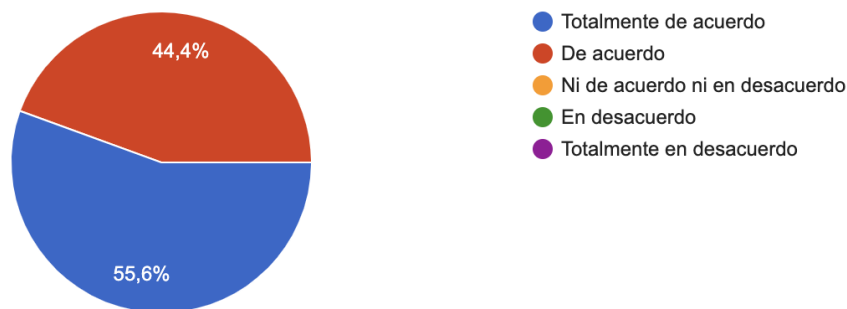
Figura 55

Representación gráfica de la novena evaluación

La plataforma mejoró al reducir tiempo y esfuerzo en comparación con la entrada de información a través de Excel



9 respuestas



La evaluación de la plataforma revela una respuesta mayoritariamente positiva entre los encuestados. Con un 55.6% indicando "Totalmente de acuerdo" y un 44.4% seleccionando "De acuerdo" en relación con la afirmación sobre la mejora en la eficiencia y reducción de esfuerzo en comparación con la entrada de información a través de Excel, se concluye que la plataforma ha logrado satisfactoriamente su objetivo. La alta aceptación y la proporción significativa de respuestas positivas validan la eficacia de la plataforma en optimizar procesos y superar las expectativas en comparación con el método previo basado en Excel.

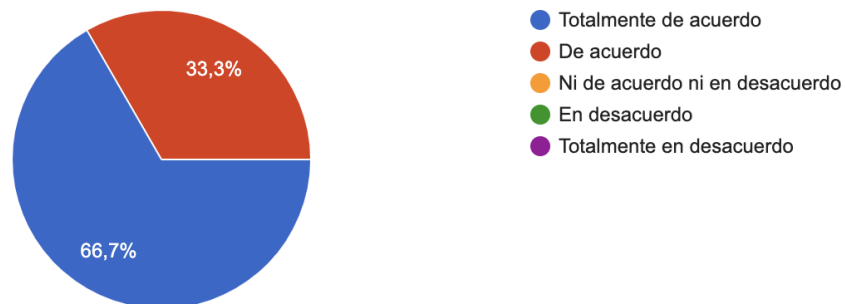
Figura 56

Representación gráfica de la décima evaluación

La velocidad de procesos, como la identificación y corrección de errores, mejoró, brindando agilidad al proceso de gestión de la actividad investigativa estudiantil



9 respuestas



La evaluación de la encuesta refleja una percepción positiva hacia la mejora en la velocidad de procesos en la gestión de la actividad investigativa estudiantil. Con un 66.7% de los participantes indicando "Totalmente de acuerdo" y un 33.3% seleccionando "De acuerdo", se evidencia un consenso general sobre la eficiencia mejorada y el impacto positivo en la agilidad del proceso. Estos resultados sugieren que la plataforma ha logrado satisfactoriamente su objetivo de optimizar la velocidad operativa en la gestión de proyectos de investigación estudiantil.

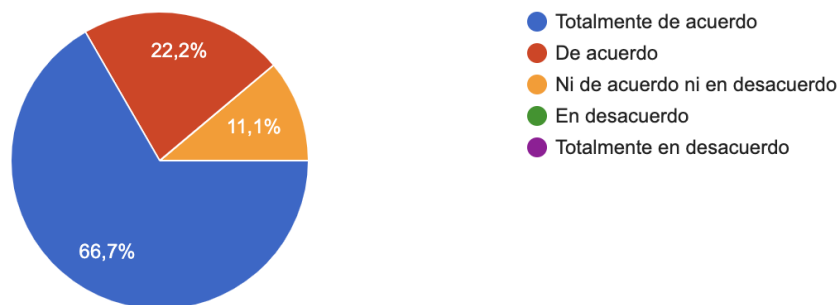
Figura 57

Representación gráfica de la undécima evaluación

La plataforma web cuenta con información actualizada acerca del historial de actividades investigativas de los estudiantes



9 respuestas



Dado que el 66.7% de los participantes expresó estar "Totalmente de acuerdo" y el 22.2% seleccionó "De acuerdo" respecto a la actualización de información en el historial de actividades investigativas, se destaca una percepción positiva. Para elevar aún más esta estadística, se sugiere revisar y abordar cualquier inquietud específica de aquellos que marcaron "Ni de acuerdo ni en desacuerdo". Además, de ser conveniente se podrían organizar sesiones de capacitación para garantizar que los usuarios estén plenamente informados sobre cómo acceder y utilizar la información en el historial de actividades investigativas, y evaluar la interfaz de la plataforma para asegurarse de que sea intuitiva y fácil de navegar. Estas acciones pueden fortalecer la confianza en la actualización del historial y mejorar la percepción general de los usuarios.

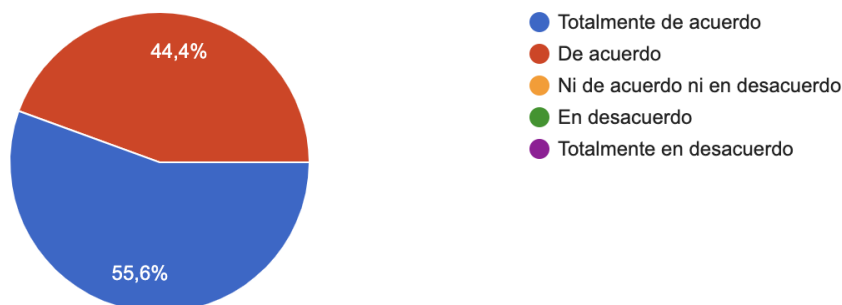
Figura 58

Representación gráfica de la Duodécima evaluación

Realiza un seguimiento de cada una de las actividades investigativas de los estudiantes que se están desarrollando de una manera sencilla y eficaz

 Copiar

9 respuestas



La evaluación refleja que un 55.6% de los participantes está "Totalmente de acuerdo" y un 44.4% está "De acuerdo" con la afirmación de que la plataforma realiza un seguimiento sencillo y eficaz de las actividades investigativas de los estudiantes. Esta respuesta positiva evidencia que la plataforma se percibe como exitosa en proporcionar un seguimiento efectivo y fácil de entender para las actividades investigativas. Para destacar aún más este aspecto positivo, se invita a aquellos que marcaron "De acuerdo" a compartir comentarios específicos sobre lo que encuentran más efectivo en el seguimiento de actividades, permitiendo resaltar características valoradas por los usuarios y compartirlas como buenas prácticas. Además, se puede promover el intercambio de experiencias positivas entre los usuarios para inspirar y fomentar la aplicación efectiva de las funciones de seguimiento. Estos pasos contribuirán a subrayar y consolidar la percepción positiva existente.

Las evaluaciones en esta sección destacan el éxito de la plataforma en optimizar diversos aspectos del proceso de gestión de la actividad investigativa estudiantil. Desde la mejora en la eficiencia y velocidad de procesos hasta el seguimiento efectivo de las actividades, la plataforma ha superado las expectativas y ha generado una percepción positiva entre los usuarios. Estas

observaciones respaldan la conclusión de que la herramienta no solo cumple con su propósito, sino que también contribuye significativamente a la eficacia y eficiencia del proceso educativo en la Universidad Mariana.

La evaluación integral de la plataforma revela un respaldo contundente a su eficiencia, funcionalidad y utilidad en la gestión de la actividad investigativa estudiantil en la Universidad Mariana. Desde la usabilidad hasta la optimización del proceso, los resultados reflejan una percepción mayoritariamente positiva entre los coordinadores y actores clave involucrados.

En la sección inicial, se destaca una aceptación mayoritaria, aunque se identifican áreas de mejora basadas en observaciones específicas. La representación gráfica de datos también recibe una evaluación positiva, señalando una comprensión efectiva de la información presentada.

En la sección de funcionalidad, la eficiencia en el tiempo de respuesta, la capacidad de cumplir funciones adecuadamente y la efectividad en lograr objetivos obtienen respuestas mayoritariamente positivas, respaldando la percepción de que la plataforma no solo cumple, sino que supera las expectativas.

La sección enfocada en la optimización del proceso destaca mejoras significativas en la eficiencia y reducción de esfuerzo en comparación con métodos anteriores. La velocidad de procesos y la actualización de información también reciben una evaluación positiva, evidenciando el impacto positivo de la plataforma en la gestión de proyectos de investigación estudiantil.

En conclusión, las evaluaciones colectivas respaldan la conclusión de que la plataforma no solo cumple con sus funciones, sino que también supera las expectativas, consolidándose como un componente valioso en la gestión de la actividad investigativa estudiantil en la Universidad Mariana. La retroalimentación positiva y las sugerencias específicas proporcionan una base sólida para futuras mejoras, asegurando una experiencia continua y satisfactoria para todos los usuarios involucrados.

3. Conclusiones

A lo largo del proceso de evaluación, se contó con la participación activa del Coordinador de Investigación Estudiantil de la Universidad Mariana, así como de los Coordinadores de Investigación y Coordinadores de Semilleros de Investigación de algunos programas.

Estos roles clave desempeñaron un papel fundamental al proporcionar sus opiniones, observaciones y sugerencias respecto a la plataforma web. Los resultados resaltaron aspectos positivos en términos de diseño, funcionalidad, usabilidad y aceptación, consolidando así la base para la implementación exitosa de la plataforma.

Una vez que el proceso quedó definido, se avanzó hacia la creación de historias de usuario, marcando así el comienzo de la programación de la plataforma web. La asesoría brindada fue esencial para superar cualquier dificultad. Cada funcionalidad de la plataforma se diseñó con el propósito de lograr eficazmente los objetivos, dejando atrás los métodos en la gestión de investigación estudiantil, donde la información se diligenciaba en un archivo Excel.

En conclusión, los resultados de la evaluación señalan que la implementación de plataformas puede ser fundamental para optimizar los procesos organizativos. En el contexto del panorama de Investigación Estudiantil, la plataforma de gestión de actividades investigativas de los estudiantes se proyecta como una herramienta integral para fortalecer la coordinación y trazabilidad de los procesos investigativos de los estudiantes, desde su concepción hasta su conclusión. Se espera que esta plataforma contribuya de manera significativa a la gestión eficiente de mencionados procesos.

4. Recomendaciones

Con el objetivo de realizar una evaluación exhaustiva y ajustes precisos, se recomienda que cada coordinador se enfoque en explorar aspectos específicos relevantes para su programa. Esto implica llevar a cabo pruebas detalladas de funcionalidades clave que se alineen con las particularidades de cada disciplina. Obtener perspectivas más amplias y diversificadas facilitará la identificación de áreas de mejora beneficiosas para todos los usuarios, contribuyendo así al desarrollo continuo de la plataforma.

Adicionalmente, se propone la participación de los coordinadores de investigación estudiantil de todos los programas en encuestas futuras. Este enfoque no solo brindará una visión continua y actualizada de las necesidades y expectativas, sino que también garantizará su participación activa en la evolución constante de la plataforma.

Referencias bibliográficas

- Amazon Web Services (2022). *Clústeres de Amazon ECS*.
https://docs.aws.amazon.com/es_es/AmazonECS/latest/developerguide/clusters.html
- Amazon Web Services. (2022). *¿Qué es Docker?* Amazon Web Services en español.
<https://aws.amazon.com/es/docker/>
- Amazon Web Services. (2022). *¿Qué son los microservicios?* Amazon Web Services en español.
<https://aws.amazon.com/es/microservices/>
- Araneda, P. (2021). Capítulo 4 Modelamiento de Datos | Base de Datos. Recuperado el 10 de mayo de 2022, de <https://bookdown.org/paranedagarcia/database/modelamiento-de-datos.html>
- Belkind, L. (2020). Does your microservice deserve its own database? <https://dev.to/lbelkind/does-your-microservice-deserve-its-own-database-np2>
- Benedet, M. (25 de abril de 2018). *¿Qué son los microservicios? Definición, características y retos*. MEDIACLOUD. <https://blog.mdcloud.es/que-son-los-microservicios-definicion-caracteristicas-y-retos/>
- Contin, D. (20 de octubre de 2021). *La baja calidad de los datos: cinco puntos sensibles de su negocio*. Validity. <https://www.validity.com/es/la-baja-calidad-de-los-datos-cinco-puntos-sensibles-de-su-negocio/>
- Digité. (2022). *Pruebas de aceptación: el qué y el porqué*.
<https://www.digite.com/es/agile/pruebas-de-aceptacion/>
- Figuroa, W. V., & Quintero, F. Á. L. (2018). *El enfoque de sistema, piedra angular para una gestión investigativa efectiva. Caso Bahía de Caráquez*. Etic@ net. Revista científica electrónica de Educación y Comunicación en la Sociedad del Conocimiento, 18(2), 398-414.

IBM (2020). *¿Qué es la arquitectura de tres niveles?*. <https://ibm.co/38etTKj>

Miguel Castaño, A., Piattini Velthuis, M., & Marcos Martínez, E. (2000). *Diseño de bases de datos relacionales*. Alfaomega, Rama.

Ministerio de Ciencia y Tecnología e Innovación. (26 de Julio de 2021). Grupos de Investigación reconocidos 2017 | Datos Abiertos Colombia. Datos.gov.co. Recuperado el 10 de mayo de 2022, de <https://www.datos.gov.co/Ciencia-Tecnolog-a-e-Innovaci-n/Grupos-de-Investigaci-n-reconocidos-2017/wz6c-k5fr>

Ministerio de Ciencia y Tecnología e Innovación. (26 de Julio de 2021). Grupos de Investigación reconocidos 2019 | Datos Abiertos Colombia. Datos.gov.co. Recuperado el 10 de mayo de 2022, de <https://www.datos.gov.co/Ciencia-Tecnolog-a-e-Innovaci-n/Grupos-de-Investigaci-n-reconocidos-2019/92tk-xn3q>

Stedman, C., & Vaughan, J. (2021). *Gestión de datos*. ComputerWeekly.es; TechTarget. <https://www.computerweekly.com/es/definicion/Gestion-de-datos>

Ventura, V. (5 de octubre de 2020). *Inglaterra anuncia 16.000 positivos de Covid-19 que no había contabilizado porque su hoja de Excel se quedó sin espacio*. elEconomista.es. <https://www.economista.es/internacional/noticias/10809139/10/20/Inglaterra-anuncia-16000-positivos-de-Covid19-que-no-habia-contabilizado-porque-su-hoja-de-Excel-se-queda-sin-espacio.html>

Rehkopf, M. (2022). Pruebas de software automatizadas para la entrega continua. <https://www.atlassian.com/es/continuous-delivery/software-testing/automated-testing>

SearchAppArchitecture. 2020. *What is the strangler pattern and how does it work?* <https://www.techtarget.com/searchapparchitecture/tip/A-detailed-intro-to-the-strangler-pattern>

Syed, A., & Rao, K. (n.d.). *Docker & the Rise of MicroServices*. Timber.io. Retrieved May 10, 2022, from <https://timber.io/blog/docker-and-the-rise-of-microservices/>

Universidad Mariana. (Diciembre de 2020). *Plan de Desarrollo Institucional 2021-2028*. <https://www.umariana.edu.co/docinstitucionales/plan-desarrollo2021-2028.pdf>

Metodología XP o Programación Extrema: ¿Qué es y cómo aplicarla?. <https://www.sinnaps.com/blog-gestion-proyectos/metodologia-xp>

¿Qué es Amazon EC2?. (2022). https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/concepts.html

Anexos

Anexo A. Carta de justificación del desarrollo tecnológico



UNIVERSIDAD MARIANA
NIT. 800.092.198-5

San Juan de Pasto, 28 de septiembre de 2021

Estimados
COMITÉ DE INVESTIGACIÓN
Programa de Ingeniería de Sistemas
Universidad Mariana
Pasto- Nariño

Un saludo cordial de paz y bien

Mediante la presente carta, expreso haber solicitado al grupo de Andrés Esteban Rizo y Jeison Fernando Caicedo, estudiantes de Ingeniería de Sistemas de la Universidad Mariana, el desarrollo de software "Plataforma para la gestión de Información de Estudiantes Investigadores", producto del trabajo de grado, con el cual se pretende solucionar tres aspectos centrales: i) cartografiar en plataforma virtual información básica y dinámica sobre los objetivos y alcances de los semilleros, co-investigadores, jóvenes investigadores, auxiliares de investigación y pasantes ii) generar información dinámica sobre estas modalidades de participación en investigación e innovación y sus proyectos y logros obtenidos, iii) presentación y difusión de convocatorias y eventos en tiempo real para que los estudiantes accedan a esta información.

Facilitaremos los recursos logísticos y dispondremos el tiempo acordado con el equipo de desarrollo.

Gracias por su atención, se suscribe


Harold Armando Juajibioy Otero
Coordinación de Investigación Estudiantil

Dirección de Investigación
InstituLAC
Universidad Mariana
hotero@umariana.edu.co
+52-7244460 Ext. 138

Anexo B. Formato de entrevista semi-estructurada



Programa de Ingeniería
de Sistemas

Formato de entrevista Semiestructurada

Objetivo del estudio

Entrevista aplicada como insumo para desarrollar el primer objetivo específico del proyecto de grado denominado “Gestión de la Actividad Investigativa Estudiantil de la Universidad Mariana: Una solución basada en una plataforma web” mediante la identificación del proceso detallado desde los diferentes campos que abarca la investigación.

Datos del informante

Nombre	
Cargo	
Programa	

Guión

N.º	Pregunta	Respuesta
1	¿Cómo recibe la información de los estudiantes de las distintas modalidades de investigación?	
2	¿ Quiénes llenan las matrices?	
3	¿ Por dónde recibe esas matrices?	
4	¿Qué información les resulta útil y cómo la utilizan?	
5	¿Cómo se vincula un estudiante al proyecto (semillero / proyecto de grado / coinvestigador)?	
6	¿En todos los programas de las diferentes facultades vinculan estudiantes a (semillero / proyecto de grado / coinvestigador)?	
7	¿Cómo diligencia a los estudiantes que se vinculan?	
8	¿Cómo se diligencia la matriz?	
9	¿A quién le envía la matriz?	
10	¿La matriz es la misma para todos los programas?	
11	¿Cada cuanto se solicita la actualización de datos?	
12	¿Con qué fin se diligencian las matrices?	
13	¿Para que se hace el seguimiento a las matrices?	

Anexo C. Aportes de la entrevista

Coordinador de Investigación Estudiantil Universidad Mariana



Universidad
Mariana



Programa de Ingeniería
de Sistemas

Formato de entrevista Semiestructurada

Objetivo del estudio

Entrevista aplicada como insumo para desarrollar el primer objetivo específico del proyecto de grado denominado “Gestión de la Actividad Investigativa Estudiantil de la Universidad Mariana: Una solución basada en una plataforma web” mediante la identificación del proceso detallado desde los diferentes campos que abarca la investigación.

Datos del informante

Nombre	Dr. Harold Juajibioy Otero
Cargo	Coordinador de Investigación Estudiantil Universidad Mariana

Guión

N.º	Pregunta	Respuesta
1	¿Cómo recibe la información de los estudiantes de las distintas modalidades de investigación?	Hay una estructura de trabajo para cada modalidad, obtenemos no en detalle información de los semilleros que se van consolidando en sus proyectos y que son agregados en las matrices, para los estudiantes coinvestigadores, el proyecto reposa en el grupo de investigación del programa y para los trabajos de grado, los proyectos pasan por acá, llegan para revisión técnica final.
2	¿Quiénes llenan las matrices?	Cada coordinador de semilleros diligencia la matriz, para el caso de coinvestigadores este año vamos a empezar el diligenciamiento de las matrices para la generación de los reportes, cuántos estudiantes coinvestigadores hay.
3	¿Por qué medio recibe esas matrices?	Con los coordinadores de semilleros tenemos un espacio de trabajo conjunto que es en Google drive, ahí están subidas las matrices para que las diligencien, y con los coinvestigadores les hacemos la solicitud al director de programa y al coordinador de investigación para que nos envíen los soportes de cómo está el proceso
4	¿Qué información les resulta útil y cómo la utilizan?	Nos interesan los datos estratégicos, cuántos estudiantes investigadores en cada modalidad son, cuántos proyectos hay, que énfasis tienen y así poder sacar estadísticas y propuestas de mejora

Coordinador de Investigación del programa de Ingeniería de Sistemas



Universidad
Mariana



Acreditación
INSTITUCIONAL

Programa de Ingeniería
de Sistemas

Formato de entrevista Semiestructurada

Objetivo del estudio

Entrevista aplicada como insumo para desarrollar el primer objetivo específico del proyecto de grado denominado “Gestión de la Actividad Investigativa Estudiantil de la Universidad Mariana: Una solución basada en una plataforma web” mediante la identificación del proceso detallado desde los diferentes campos que abarca la investigación.

Datos del informante

Nombre	Dr. Fabián Parra Pay
Cargo	Coordinador de Investigación
Programa	Ingeniería de Sistemas

Guión

N.º	Pregunta	Respuesta
1	¿Cómo se vincula un estudiante al proyecto (semillero / proyecto de grado / coinvestigador)?	Cuando un estudiante llega a quinto semestre, elige de una matriz con ideas propuestas su proyecto o genera una idea de investigación, se le asigna un asesor y realiza la primera fase del proyecto que culmina con el aval del asesor al finalizar el semestre.
2	¿En todos los programas de las diferentes facultades vinculan estudiantes a (semillero / proyecto de grado / coinvestigador)?	Si, pero en el programa de Ingeniería de Sistemas para proyectos de grado tenemos dos etapas, hay otros programas que manejan tres etapas
3	¿Cómo diligencia a los estudiantes que se vinculan?	En una matriz en Excel
4	¿Cómo diligencia la matriz?	Yo tengo un lista de todos los proyectos, quienes son los autores, quien es el asesor, quiénes son los jurados, fechas de entrega, diferentes etapas del proyecto y contabilizar como están los profesores en cuanto a carga de asesorías.
5	¿A quién le envía la matriz?	El informe de investigación del programa se lo remito a dirección de investigación y ellos se lo remiten al coordinador Harold.
6	¿Con qué fin se diligencian las matrices?	Para que mediante información consolidada sepan cuales son los resultados para saber cómo va la investigación dentro del programa.

Coordinadora de Semilleros de Investigación Programa de Ingeniería de Sistemas



Universidad
Mariana



Acreditación
INSTITUCIONAL

Programa de Ingeniería
de Sistemas

Formato de entrevista Semiestructurada

Objetivo del estudio

Entrevista aplicada como insumo para desarrollar el primer objetivo específico del proyecto de grado denominado “Gestión de la Actividad Investigativa Estudiantil de la Universidad Mariana: Una solución basada en una plataforma web” mediante la identificación del proceso detallado desde los diferentes campos que abarca la investigación.

Datos del informante

Nombre	Mgs. Leidy Marcela Gómez Melo
Cargo	Coordinadora de Semilleros de Investigación
Programa	Ingeniería de Sistemas

Guión

N.º	Pregunta	Respuesta
1	¿Cómo se vincula un estudiante como semillerista?	El estudiante participa en una serie de talleres y encuentros interdisciplinarios para generar una idea de investigación, cuando ya está la idea formulada se realiza la diligencia en una ficha de inscripción.
2	¿En todos los programas de las diferentes facultades vinculan estudiantes semilleristas?	Si, independientemente de si el estudiante esté trabajando o no el proyecto con estudiantes de otros programas, se hace la vinculación del semillerista al programa al que pertenece.
3	¿Cómo diligencia a los estudiantes que se vinculan?	A través de una gestión documental que es un plan detallado de trabajo de todas las actividades que se van a desarrollar en el semestre y también se diligencia una matriz donde está la lista de los semilleristas.
4	¿Cómo se diligencia la matriz?	El coordinador Harold envía un enlace de acceso a Google Drive donde están carpetas para cada programa y estas tienen una matriz que cada coordinador de semilleros debe diligenciar.
5	¿A quién le envía la matriz?	El coordinador accede a cada matriz mediante Google Drive para revisar los diligenciamientos y las actualizaciones que se van realizando.
6	¿La matriz es la misma para todos los programas?	Si, cada coordinador de semilleros tiene una carpeta en el drive donde está la misma plantilla de matriz que en las demás.
7	¿Cada cuánto se solicita la actualización de datos?	Cada semestre se actualiza la matriz.
8	¿Con qué fin se diligencian las matrices?	Para obtener reportes estadísticos como cantidad de estudiantes semilleristas, cantidad de semilleristas que participaron en x eventos.

Coordinadora de Co-Investigación Estudiantil del programa Ingeniería de Sistemas



Programa de Ingeniería
de Sistemas

Formato de entrevista Semiestructurada

Objetivo del estudio

Entrevista aplicada como insumo para desarrollar el primer objetivo específico del proyecto de grado denominado “Gestión de la Actividad Investigativa Estudiantil de la Universidad Mariana: Una solución basada en una plataforma web” mediante la identificación del proceso detallado desde los diferentes campos que abarca la investigación.

Datos del informante

Nombre	Dra. Alejandra Zuleta Medina
Cargo	Coordinadora de Co-investigación estudiantil
Programa	Ingeniería de Sistemas

Guión

N.º	Pregunta	Respuesta
1	¿Cómo se vincula un estudiante como coinvestigador?	En quinto semestre un estudiante que haya mostrado más compromiso y se vincule a una idea que esté relacionada con investigación docente empieza a formar parte como co-investigador para desarrollar uno de los objetivos de esa investigación.
2	¿Todos los programas de las diferentes facultades vinculan estudiantes co-investigadores?	No todos los programas lo incorporan, pero lo ideal para los indicadores de investigación es que todos los programas tengan estudiantes coinvestigadores.
3	¿Cómo diligencia a los estudiantes que se vinculan?	A través de una matriz
4	¿Cómo se diligencia la matriz?	Se diligencia la matriz y se envía por correo electrónico
5	¿A quién le envía la matriz?	A la profesora Marcela Cárdenas o al Profesor Harold
6	¿La matriz es la misma para todos los programas?	La matriz es la misma, pero ha venido cambiando a lo largo del tiempo, la última que se diligencio fue de coinvestigadores con productos.
7	¿Cada cuánto se solicita la actualización de datos?	Finalizando el semestre que es cuando se organizan los informes.
8	¿Con qué fin se diligencian las matrices?	Para reportar informes para procesos como la acreditación institucional o del programa.

Anexo D. Historias de usuario

Tabla 1.

Historia de usuario - Inicio de Sesión

Historia de Usuario			
Código:	HU-001		
Nombre:	Inicio de sesión		
Actor:	Coordinador de Investigación Estudiantil y coordinadores de las distintas modalidades de Investigación estudiantil		
Descripción:	Como coordinador quiero iniciar sesión con google vía correo institucional para acceder al apartado de gestión de información de mi modalidad.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se seleccione un correo institucional con acceso a la plataforma.	Se inicia sesión con el correo elegido y se carga la pantalla con la información propia de la modalidad.
	2	Cuando se seleccione un correo institucional que no está registrado en la plataforma.	Se mostrará un mensaje de error informando que el correo ingresado no tiene acceso a la plataforma.
	3	Cuando se seleccione un correo externo a la Universidad.	Se mostrará un mensaje de error informando que ese correo no pertenece a la Universidad Mariana.

Tabla 2..

Historia de usuario - Gestión de permisos

Historia de Usuario			
Código:	HU-002		
Nombre:	Gestión de permisos		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero dar acceso a los coordinadores de las distintas modalidades de investigación estudiantil para que puedan acceder a la plataforma y utilizar las funcionalidades.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se da acceso al correo de un coordinador de una modalidad de investigación estudiantil.	El coordinador al que se le dió acceso podrá iniciar sesión con su correo institucional en la plataforma y hacer uso de los servicios que en esta dispone.

Tabla 3.

Historia de usuario - Gestión de información estudiantil

Historia de Usuario			
Código:	HU-003		
Nombre:	Gestión de información estudiantil		
Actor:	Coordinadores de las distintas modalidades de Investigación estudiantil		
Descripción:	Como coordinador de una modalidad de investigación estudiantil quiero poder registrar, editar y eliminar la información de un estudiante que está vinculado a un proyecto para poder llevar un control de sus avances.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra un nuevo estudiante con datos válidos.	Se registra el estudiante en la base de datos y se añade a la lista de estudiantes mostrada en pantalla.
	2	Cuando se registra un estudiante que ya se encuentra registrado.	Se muestra un mensaje de error informando que el estudiante que intenta registrar ya está registrado.
	3	Cuando se edite la información de un estudiante registrado.	Se actualizarán los campos del estudiante y se informará de la correcta actualización.
	4	Cuando se seleccione un estudiante registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del estudiante seleccionado y se mostrará un mensaje informando de la correcta eliminación.

Tabla 4.

Historia de usuario - Gestión de información de proyectos

Historia de Usuario			
Código:	HU-004		
Nombre:	Gestión de información de proyectos		
Actor:	Coordinadores de las distintas modalidades de Investigación estudiantil		
Descripción:	Como coordinador de una modalidad de investigación estudiantil quiero poder registrar, editar y eliminar la información de un proyecto al que se vincula un estudiante para dar seguimiento a los avances del proyecto		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra un nuevo proyecto con la información válida y necesaria.	Se registra el proyecto en la base de datos y se añade a la lista de proyectos mostrada en pantalla.

	2	Cuando se registra un proyecto que ya se encuentra registrado.	Se muestra un mensaje de error informando que el proyecto que intenta registrar ya está registrado.
	3	Cuando se edita la información de un proyecto registrado.	Se actualizarán los campos del proyecto y se informará de la correcta actualización.
	4	Cuando se seleccione un proyecto registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del proyecto seleccionado y se mostrará un mensaje informando de la correcta eliminación.

Tabla 5.

Historia de usuario - Vinculación de un estudiante a un proyecto

Historia de Usuario		
Código:	HU-005	
Nombre:	Vinculación de un estudiante a un proyecto	
Actor:	Coordinadores de las distintas modalidades de Investigación estudiantil	
Descripción:	Como coordinador de una modalidad de investigación estudiantil quiero poder vincular a un estudiante registrado a su proyecto.	
Criterios de aceptación:	CID	Condición
	1	Cuando se vincula un estudiante registrado a un proyecto registrado.
	2	Cuando se vincula un estudiante que ya cuenta con un proyecto a otro de la misma modalidad.
3	Cuando se vincula un estudiante que ya cuenta con un proyecto a su mismo proyecto.	Resultado
		Se realiza la vinculación correctamente y se muestra que el estudiante está trabajando en ese proyecto.
		Se muestra un mensaje de error informando que el estudiante ya tiene un proyecto vinculado en esa modalidad.
		Se muestra un mensaje de error informando que el estudiante ya está vinculado a ese proyecto.

Tabla 6.

Historia de usuario - Historial de cambios

Historia de Usuario

Código:	HU-006		
Nombre:	Historial de cambios		
Actor:	Coordinadores de las distintas modalidades de Investigación estudiantil		
Descripción:	Como coordinador de una modalidad de investigación estudiantil quiero poder ver cuál ha sido mi historial reciente de cambios en la lista de proyectos y estudiantes para poder comparar y regresar a la información pasada.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se dé clic al botón de ver historial y se hayan realizado cambios.	Se mostrará una pantalla con los cambios más recientes que ha realizado.
	2	Cuando se dé clic al botón de ver historial y no se hayan realizado cambios.	Se mostrará una pantalla sin datos y se informará que no se han realizado cambios.

Tabla 7.

Historia de usuario - Filtrar por criterios de búsqueda

Historia de Usuario			
Código:	HU-007		
Nombre:	Filtrar por criterios de búsqueda		
Actor:	Coordinadores de las distintas modalidades de Investigación estudiantil		
Descripción:	Como coordinador de una modalidad de investigación estudiantil quiero poder filtrar la información de estudiantes y proyectos por diferentes criterios de búsqueda para que se facilite el acceso y gestión de la información.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se establezca un criterio de búsqueda.	Se mostrarán los resultados que cumplan con el criterio establecido.
	2	Cuando se eliminen los criterios de búsqueda.	Se mostrará toda la información sin ningún filtro.

Tabla 8.

Historia de usuario - Importar información

Historia de Usuario	
Código:	HU-008
Nombre:	Importar información
Actor:	Coordinadores de las distintas modalidades de Investigación estudiantil
Descripción:	Como coordinador de una modalidad de investigación estudiantil quiero poder importar la información de estudiantes y proyectos desde un archivo para que se carguen los datos en la plataforma.

	CID	Condición	Resultado
Criterios de aceptación:	1	Cuando se sube un archivo que tenga la misma estructura de datos que en la plataforma.	Se cargará la información que tengan los archivos correctamente en la plataforma.
	2	Cuando se sube un archivo que no tenga la misma estructura de datos que en la plataforma.	Se mostrará un mensaje de error informando que la estructura que tiene el archivo no es soportada.

Tabla 9.

Historia de usuario - Exportar información

Historia de Usuario			
Código:	HU-009		
Nombre:	Exportar información		
Actor:	Coordinadores de las distintas modalidades de Investigación estudiantil		
Descripción:	Como coordinador de una modalidad de investigación estudiantil quiero poder exportar la información de los estudiantes y proyectos a un archivo.		
	CID	Condición	Resultado
Criterios de aceptación:	1	Cuando se exporta una lista con datos.	Se descargará un archivo de tipo Excel con la información seleccionada.
	2	Cuando se exporta una lista vacía.	Se mostrará un mensaje de error informando que no se puede generar un archivo vacío.

Tabla 10.

Historia de usuario - Gestión de asesores

Historia de Usuario			
Código:	HU-010		
Nombre:	Gestión de asesores		
Actor:	Coordinador de Investigación		
Descripción:	Como coordinador de investigación de un programa quiero poder registrar, editar y eliminar la información de un profesor del programa que será asesor para poder vincularlo a un proyecto.		
	CID	Condición	Resultado
Criterios de aceptación:	1	Cuando se registra un nuevo asesor con la información válida y necesaria.	Se registra el asesor en la base de datos y se añade a la lista de asesores mostrada en pantalla.
	2	Cuando se registra un asesor que ya se encuentra registrado.	Se muestra un mensaje de error informando que el asesor que

		intenta registrar ya está registrado.
3	Cuando se edita la información de un asesor registrado.	Se actualizarán los campos del asesor y se informará de la correcta actualización.
4	Cuando se seleccione un asesor registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del asesor seleccionado y se mostrará un mensaje informando de la correcta eliminación.

Tabla 11.

Historia de usuario - Gestión de grupos de investigación profesoral

Historia de Usuario			
Código:	HU-011		
Nombre:	Gestión de grupos de investigación profesoral		
Actor:	Coordinador de Co-Investigación estudiantil.		
Descripción:	Como coordinador de co-investigación estudiantil, quiero poder registrar, editar y eliminar la información de grupos de investigación profesoral para que se le vincule un estudiante registrado.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra un nuevo grupo de investigación profesoral con la información válida y necesaria.	Se registra el grupo de investigación profesoral en la base de datos y se añade a la lista de grupos de investigación profesoral mostrada en pantalla.
	2	Cuando se registra un grupo de investigación profesoral que ya se encuentra registrado.	Se muestra un mensaje de error informando que el grupo de investigación profesoral que intenta registrar ya está registrado.
	3	Cuando se edita la información de un grupo de investigación profesoral registrado.	Se actualizarán los campos del grupo de investigación profesoral y se informará de la correcta actualización.
	4	Cuando se seleccione un asesor registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del asesor seleccionado y se mostrará un mensaje informando de la correcta eliminación.

Tabla 12.

Historia de usuario - Gestión de facultades

Historia de Usuario			
Código:	HU-012		
Nombre:	Gestión de facultades		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de una facultad.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra una nueva facultad con la información válida y necesaria.	Se registra la facultad en la base de datos y se añade a la lista de facultades mostrada en pantalla.
	2	Cuando se registra una facultad que ya se encuentra registrada.	Se muestra un mensaje de error informando que la facultad que intenta registrar ya está registrada.
	3	Cuando se edita la información de una facultad registrada.	Se actualizarán los campos de la facultad y se informará de la correcta actualización.
	4	Cuando se seleccione una facultad registrada para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información de la facultad seleccionada y se mostrará un mensaje informando de la correcta eliminación.

Tabla 13.

Historia de usuario - Gestión de programas

Historia de Usuario			
Código:	HU-013		
Nombre:	Gestión de programas		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de un programa.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra un nuevo programa con la información válida y necesaria.	Se registra el programa en la base de datos y se añade a la lista de programas mostrada en pantalla.
	2	Cuando se registra un programa que ya se encuentra registrado.	Se muestra un mensaje de error informando que el programa que intenta registrar ya está registrado.

3	Cuando se edita la información de un programa registrado.	Se actualizarán los campos del programa y se informará de la correcta actualización.
4	Cuando se seleccione un programa registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del programa seleccionado y se mostrará un mensaje informando de la correcta eliminación.

Tabla 14.

Historia de usuario - Generación de reportes estadísticos

Historia de Usuario			
Código:	HU-014		
Nombre:	Generación de reportes estadísticos		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder generar reportes estadísticos de los estudiantes y proyectos de investigación de las diferentes modalidades, facultades, programas y de manera general.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se genera un reporte estadístico de una modalidad.	Se muestran gráficamente las estadísticas de esa modalidad.
	2	Cuando se genera un reporte estadístico de una facultad.	Se muestran gráficamente las estadísticas de las diferentes modalidades que pertenezcan a la facultad.
	3	Cuando se genera un reporte estadístico de un programa.	Se muestran gráficamente las estadísticas de las diferentes modalidades que pertenezcan al programa.
	4	Cuando se genera un reporte estadístico general.	Se muestran gráficamente las estadísticas de todas las modalidades.

Tabla 15.

Historia de usuario - Gestión de beneficiarios

Historia de Usuario	
Código:	HU-015
Nombre:	Gestión de beneficiarios
Actor:	Coordinador de Investigación Estudiantil

Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de los beneficiarios del proyecto.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra un nuevo beneficiario con la información válida y necesaria.	Se registra el beneficiario en la base de datos y se añade a la lista de beneficiarios mostrada en pantalla.
	2	Cuando se registra un beneficiario que ya se encuentra registrado.	Se muestra un mensaje de error informando que el beneficiario que intenta registrar ya está registrado.
	3	Cuando se edita la información de un beneficiario registrado.	Se actualizarán los campos del beneficiario y se informará de la correcta actualización.
	4	Cuando se seleccione un beneficiario registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del beneficiario seleccionado y se mostrará un mensaje informando de la correcta eliminación.

Tabla 16.

Historia de usuario - Gestión de estados del proyecto

Historia de Usuario			
Código:	HU-016		
Nombre:	Gestión de estados del proyecto		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar el estado de un proyecto.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra un nuevo estado con la información válida y necesaria.	Se registra el estado en la base de datos y se añade a la lista de estados mostrada en pantalla.
	2	Cuando se registra un estado que ya se encuentra registrado.	Se muestra un mensaje de error informando que el estado que intenta registrar ya está registrado.
	3	Cuando se edita la información de un estado registrado.	Se actualizarán los campos del estado y se informará de la correcta actualización.
	4	Cuando se seleccione un estado registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del estado seleccionado y se

mostrará un mensaje informando de la correcta eliminación.

Tabla 17.

Historia de usuario - Gestión de modalidades de grado

Historia de Usuario			
Código:	HU-017		
Nombre:	Gestión de modalidades de grado		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de una modalidad de grado.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra una nueva modalidad de grado con la información válida y necesaria.	Se registra la modalidad de grado en la base de datos y se añade a la lista de modalidades de grado mostrada en pantalla.
	2	Cuando se registra una modalidad de grado que ya se encuentra registrada.	Se muestra un mensaje de error informando que la modalidad de grado que intenta registrar ya está registrada.
	3	Cuando se edita la información de una modalidad de grado registrada.	Se actualizarán los campos de la modalidad de grado y se informará de la correcta actualización.
	4	Cuando se seleccione una modalidad de grado registrada para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información de la modalidad de grado seleccionada y se mostrará un mensaje informando de la correcta eliminación.

Tabla 18.

Historia de usuario - Gestión de sub-modalidades de investigación

Historia de Usuario	
Código:	HU-018
Nombre:	Gestión de sub-modalidades de investigación
Actor:	Coordinador de Investigación Estudiantil

Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de una sub-modalidad de investigación.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra una nueva sub-modalidad de investigación con la información válida y necesaria.	Se registra la sub-modalidad de investigación en la base de datos y se añade a la lista de sub-modalidades de investigación de grado mostrada en pantalla.
	2	Cuando se registra una sub-modalidad de investigación que ya se encuentra registrada.	Se muestra un mensaje de error informando que sub-modalidad de investigación que intenta registrar ya está registrada.
	3	Cuando se edita la información de una sub-modalidad de investigación registrada.	Se actualizarán los campos de la sub-modalidad de investigación y se informará de la correcta actualización.
	4	Cuando se seleccione una sub-modalidad de investigación registrada para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información de la sub-modalidad de investigación seleccionada y se mostrará un mensaje informando de la correcta eliminación.

Tabla 19.

Historia de usuario - Gestión de tipos de identificación

Historia de Usuario			
Código:	HU-019		
Nombre:	Gestión de tipos de identificación		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de los tipos de identificación que podrán tener los estudiantes.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra un nuevo tipo de identificación con la información válida y necesaria.	Se registra el tipo de identificación en la base de datos y se añade a la lista de tipos de identificación mostrada en pantalla.
	2	Cuando se registra un tipo de identificación que ya se encuentra registrado.	Se muestra un mensaje de error informando que el tipo de identificación que intenta registrar ya está registrado.

3	Cuando se edita la información de un tipo de identificación registrado.	Se actualizarán los campos del tipo de identificación y se informará de la correcta actualización.
4	Cuando se seleccione un tipo de identificación registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del tipo de identificación seleccionado y se mostrará un mensaje informando de la correcta eliminación.

Tabla 20.

Historia de usuario - Gestión de tipos de producto

Historia de Usuario			
Código:	HU-020		
Nombre:	Gestión de tipos de producto		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de los tipos de producto generados por los proyectos estudiantiles.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra un nuevo tipo de producto con la información válida y necesaria.	Se registra el tipo de producto en la base de datos y se añade a la lista de tipos de producto mostrada en pantalla.
	2	Cuando se registra un tipo de producto que ya se encuentra registrado.	Se muestra un mensaje de error informando que el tipo de producto que intenta registrar ya está registrado.
	3	Cuando se edita la información de un tipo de producto registrado.	Se actualizarán los campos del tipo de producto y se informará de la correcta actualización.
	4	Cuando se seleccione un tipo de producto registrado para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información del tipo de producto seleccionado y se mostrará un mensaje informando de la correcta eliminación.

Tabla 21.

Historia de usuario - Gestión de coordinaciones de investigación

Historia de Usuario			
Código:	HU-021		
Nombre:	Gestión de coordinaciones de investigación		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de las coordinaciones de investigación de la Universidad Mariana.		
Criterios de aceptación:	CID	Condición	Resultado
	1	Cuando se registra una nueva coordinación de investigación con la información válida y necesaria.	Se registra la coordinación de investigación en la base de datos y se añade a la lista de coordinaciones de investigación mostrada en pantalla.
	2	Cuando se registra una coordinación de investigación que ya se encuentra registrada.	Se muestra un mensaje de error informando que la coordinación de investigación que intenta registrar ya está registrada.
	3	Cuando se edita la información de una coordinación de investigación registrada.	Se actualizarán los campos de la coordinación de investigación y se informará de la correcta actualización.
	4	Cuando se seleccione una coordinación de investigación registrada para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información de la coordinación de investigación seleccionada y se mostrará un mensaje informando de la correcta eliminación.

Tabla 22.

Historia de usuario - Gestión de coordinaciones de investigación de un programa académico

Historia de Usuario			
Código:	HU-022		
Nombre:	Gestión de coordinaciones de investigación de un programa académico		
Actor:	Coordinador de Investigación Estudiantil		
Descripción:	Como coordinador de investigación estudiantil quiero poder registrar, editar y eliminar la información de las coordinaciones de investigación de un programa académico.		
	CID	Condición	Resultado

Criterios de aceptación:	1	Cuando se registra un nueva coordinación de investigación de un programa académico con la información válida y necesaria.	Se registra la coordinación de investigación de un programa académico en la base de datos y se añade a la lista de coordinaciones de investigación del programa académico mostrada en pantalla.
	2	Cuando se registra una coordinación de investigación de un programa académico que ya se encuentra registrada.	Se muestra un mensaje de error informando que la coordinación de investigación del programa académico que intenta registrar ya está registrada.
	3	Cuando se edita la información de una coordinación de investigación de un programa académico registrada.	Se actualizarán los campos de la coordinación de investigación del programa académico y se informará de la correcta actualización.
	4	Cuando se seleccione una coordinación de investigación de un programa académico registrada para que su información sea eliminada.	Se eliminará de la base de datos y de la lista la información de la coordinación de investigación del programa académico seleccionada y se mostrará un mensaje informando de la correcta eliminación.

Anexo E. Mockups pantallas

Figura 1.

Mockup gestión de proyectos

Proyectos

Escriba información de proyecto Buscar Importar Exportar Registrar nuevo proyecto

Lista de **Proyectos**

		Nombre	Fecha de Inicio	Fecha de Finalización
Editar	Borrar	Gestión de la actividad investigativa estudiantil	13/02/2020	14/03/2021
Editar	Borrar	Desarrollo de un aplicativo móvil en Applinventor	15/05/2020	17/07/2021
Editar	Borrar	Gestión de la actividad investigativa estudiantil	16/06/2021	01/01/2022
Editar	Borrar	Gestión de la actividad investigativa estudiantil	20/08/2021	14/02/2022

Figura 2.

Mockup formulario edición de proyectos

Proyectos

Editar Proyecto

Nombre	Fecha de Inicio	Fecha de Finalización
Gestión de la actividad investigativa	13/02/2020	14/03/2021
Número de documento	Correo electrónico	Programa
1010134969	josueses.fernandez@umariana.edu.co	Ingeniería de Sistemas
Semestre		
IX		

Cancelar Editar

Figura 3.

Mockup diálogo registro líneas de investigación

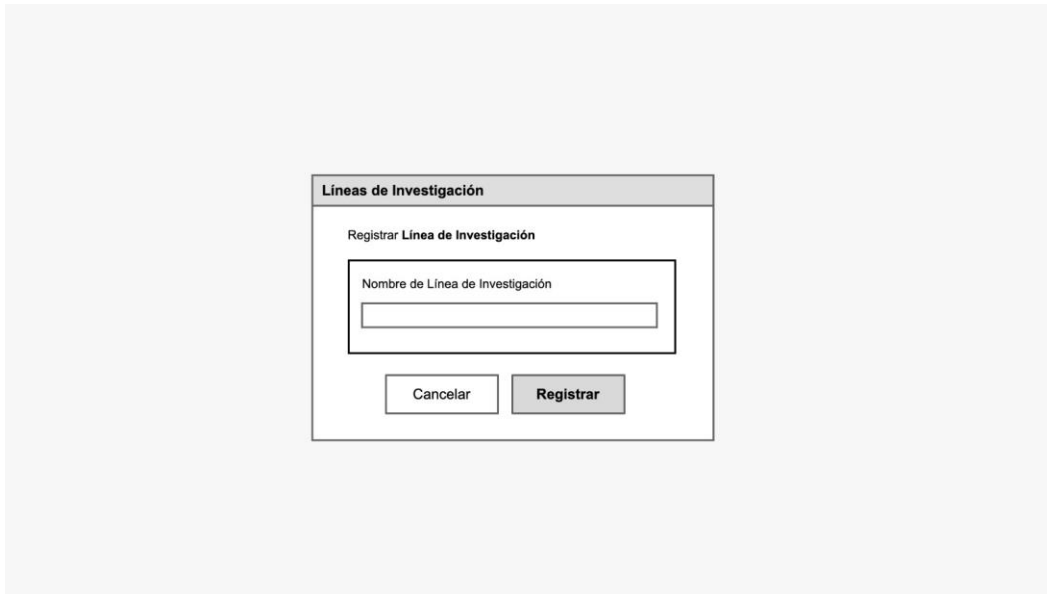


Figura 4.

Mockup gestión líneas de investigación

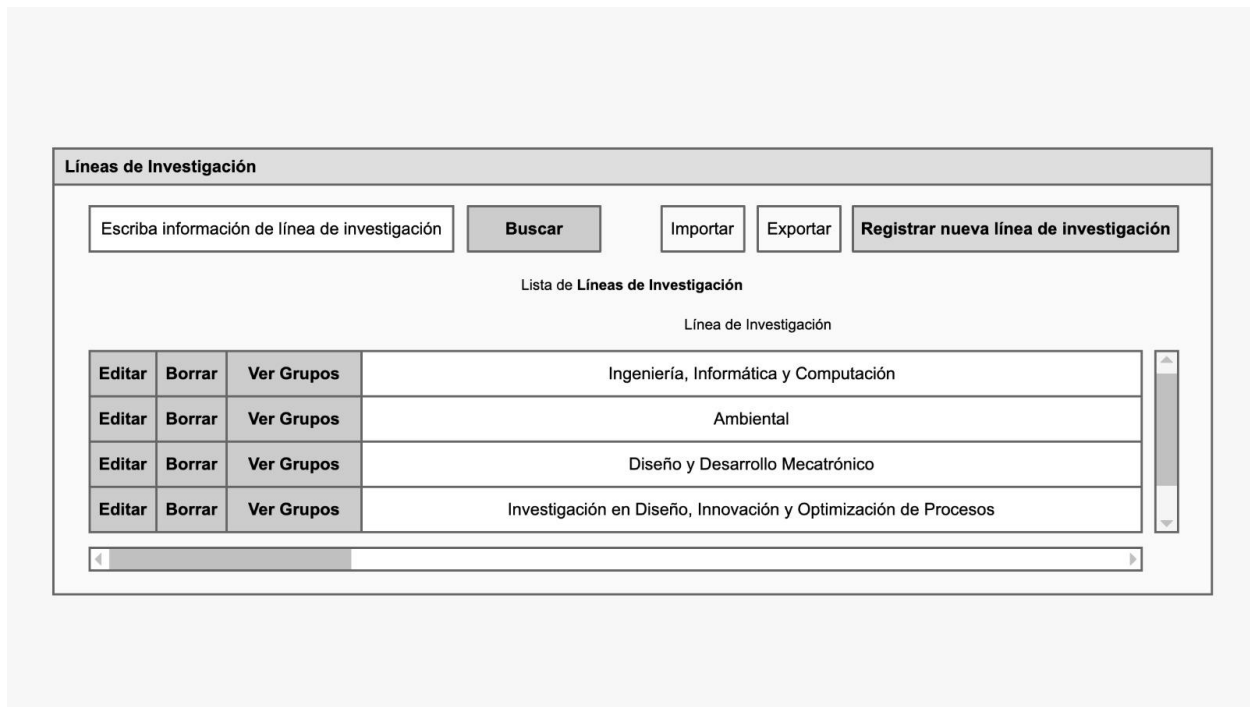


Figura 5.

Mockup gestión grupos investigación

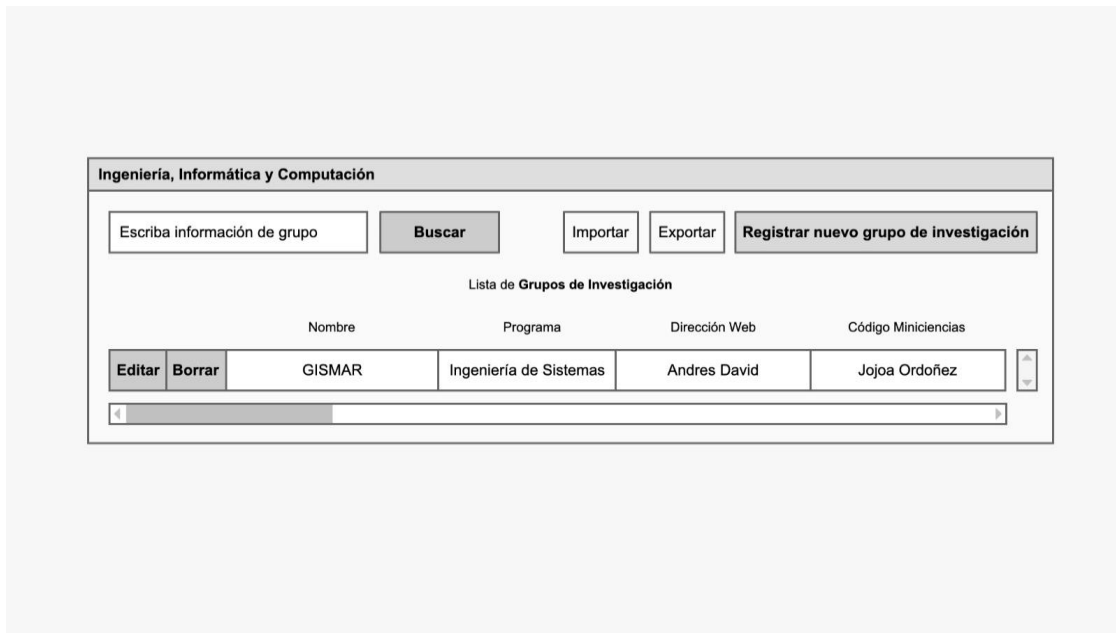


Figura 6.

Mockup gestión programas

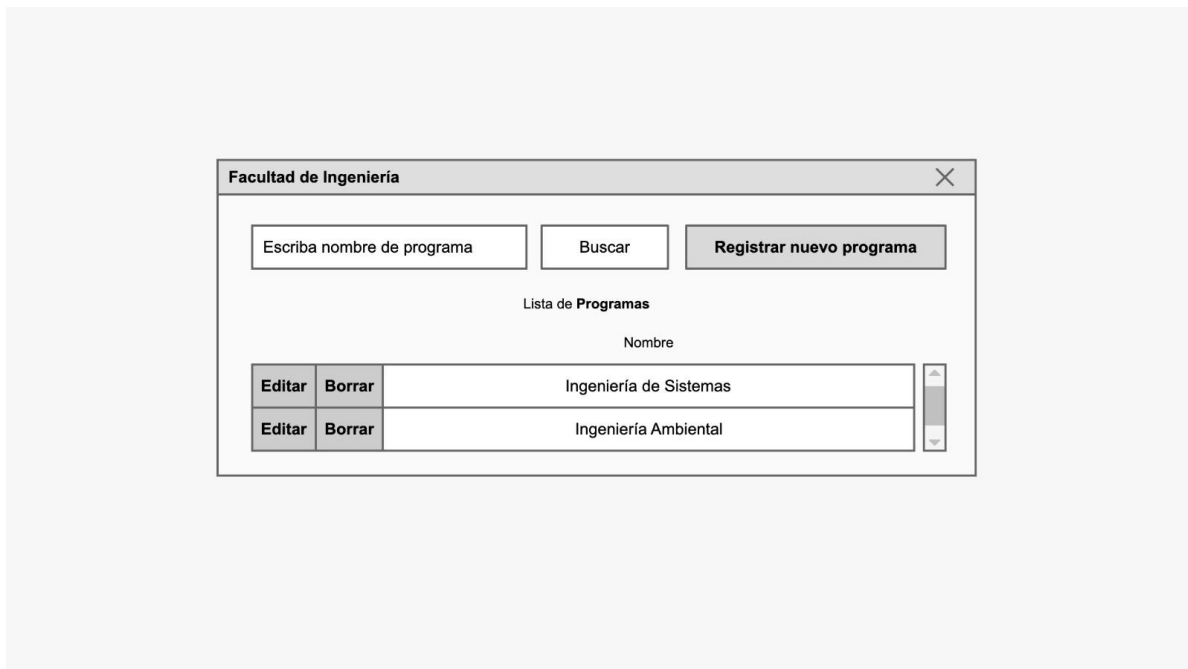


Figura 7.

Mockup gestión facultades

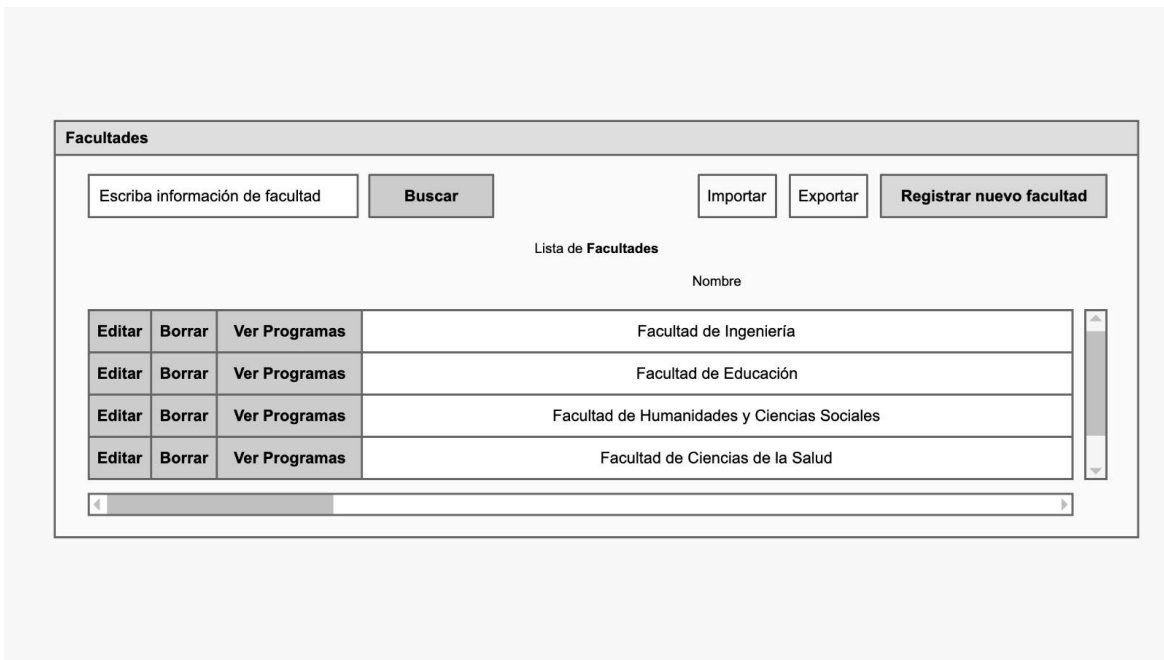


Figura 8.

Mockup gestión profesores

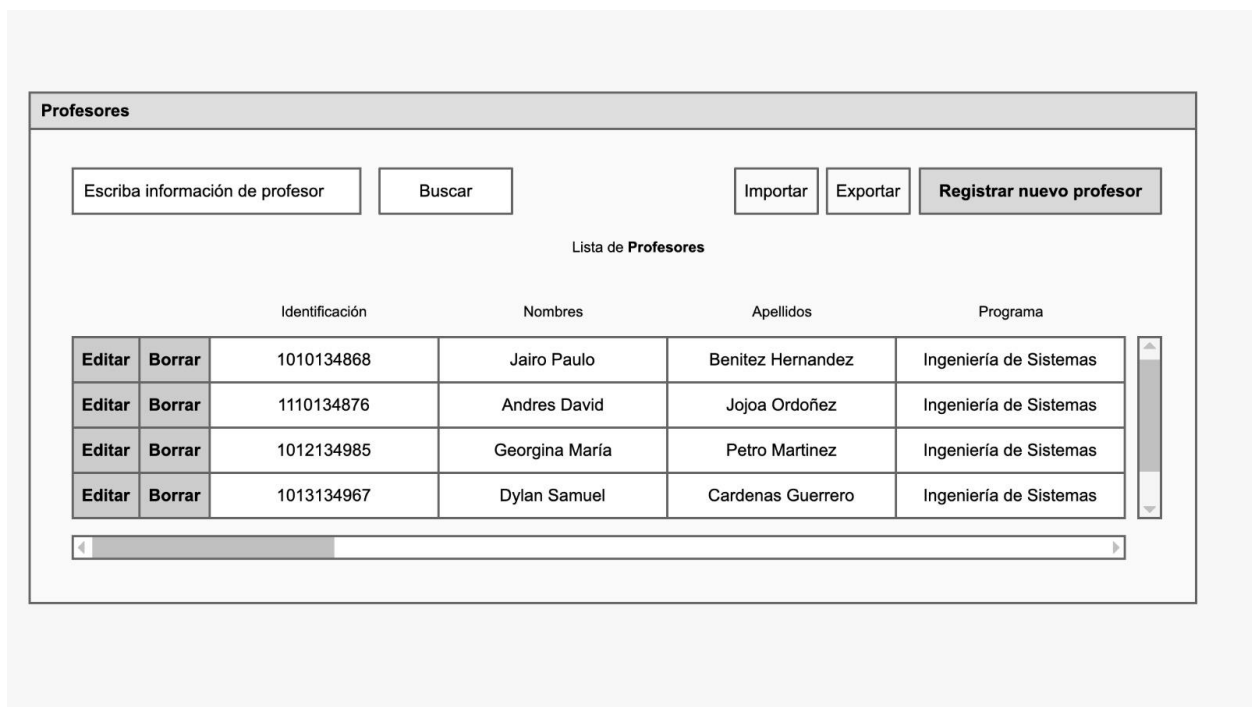


Figura 9.

Mockup gestión estudiantes

Estudiantes

Escriba información de estudiante Buscar Importar Exportar **Registrar nuevo estudiante**

Lista de **Estudiantes**

		Identificación	Nombres	Apellidos	Programa	
Editar	Borrar	1010134888	Jeisson Fernando	Caicedo Burbano	Ingeniería de Sistemas	↑ ↓
Editar	Borrar	1010134876	Andres Esteban	Rizo Zamora	Ingeniería de Sistemas	
Editar	Borrar	1010134989	Camilo David	Acosta Medina	Ingeniería de Sistemas	
Editar	Borrar	1010134969	Josué Esteban	Fernandez Díaz	Ingeniería de Sistemas	

◀ ▶

Anexo F. Escenarios de prueba

Tabla 1.

Escenario de prueba historia de usuario 003

Código HU	HU-003
No	1
Descripción	Escenario con dos estudiantes, un tipo de identificación, una facultad y un programa académico
Datos	<p>TipoIdentificación = { id: 1L, nombreTipo: "Cédula" }</p> <p>Facultad = { id: 1L, nombreFacultad: "Ingeniería" }</p> <p>ProgramaAcademico = { id: 1L, nombrePrograma: "Ingeniería de sistemas", facultad: 1L }</p> <p>Estudiante1 = {id: 1L, nombres: "Andres Esteban", apellidos: "Rizo Zamora", tipoIdentificacion: 1L, numeroDocumento: "1193", correoElectronico: arizo@umariana.edu.co, programaAcademico: 1L, semestre: "VIII"}</p> <p>Estudiante2 = {id: 2L, nombres: "Jeisson Fernando", apellidos: "Caicedo Burbano", tipoIdentificacion: 1L, numeroDocumento: "1008", correoElectronico: jcaicedo@umariana.edu.co, programaAcademico: 1L, semestre: "VIII"}</p>
CID	1, 2 , 3 ,4

Tabla 2.

Escenario de prueba historia de usuario 004

Código HU	HU-004
No	2
Descripción	Escenario con dos proyectos, dos estados, dos beneficiarios, una política, un tipo de identificación, una facultad, dos programa y dos profesores
Datos	<p>Estado1 = {id: 1L, estado: "En progreso", proyectoModalidad: "Proyecto de grado"}</p> <p>Estado2 = {id: 2L, estado: "En revisión", proyectoModalidad: "Proyecto de grado"}</p> <p>Beneficiario1 = {id: 1L, tipo: "Instituciones", descripción: "Instituciones"}</p> <p>Beneficiario2 = {id: 2L, tipo: "Universidades", descripción: "Universidades"}</p> <p>Politica = {id: 1L, tipo: "Política", descripción: "Política"}</p> <p>TipoIdentificación = { id: 1L, nombreTipo: "Cédula" }</p> <p>Facultad = { id: 1L, nombreFacultad: "Ingeniería" }</p> <p>ProgramaAcademico1 = { id: 1L, nombrePrograma: "Ingeniería de sistemas", facultad: 1L }</p> <p>ProgramaAcademico2 = { id: 2L, nombrePrograma: "Ingeniería ambiental", facultad: 1L }</p> <p>Profesor1 = {id: 1L, nombre: "Fabian", apellido: "Parra", tipoIdentificacion: 1L, numeroDocumento: "1006", programaAcademico: 1L}</p>

	<p>Profesor2 = {id: 2L, nombre: "Jaime", apellido: "Coral", tipoIdentificacion: 1L, numeroDocumento: "1008", programaAcademico: 2L}</p> <p>Proyecto1 = {id: 1L, fechaInicio: "03-03-2022", fechaFin: "03-03-2025", proyectoModalidad: "Proyecto de grado", modalidadGrado: null, subModalidad: null, estado: 1L, beneficiario: 1L, política: 1L, financiacion: null, grupoInvestigacion: null, asesor: 1L}</p> <p>Proyecto2 = {id: 2L, fechaInicio: "03-03-2021", fechaFin: "03-03-2024", proyectoModalidad: "Proyecto de grado", modalidadGrado: null, subModalidad: null, estado: 2L, beneficiario: 2L, política: 2L, financiacion: null, grupoInvestigacion: null, asesor: 2L}</p>
CID	1 , 2, 3 ,4

Tabla 3.

Escenario de prueba historia de usuario 010

Código HU	HU-010
No	3
Descripción	Escenario con un tipo de identificación, una facultad, un programa académico y dos profesores
Datos	<p>TipoIdentificación = { id: 1L, nombreTipo: "Cédula" }</p> <p>Facultad = { id: 1L, nombreFacultad: "Ingeniería" }</p> <p>ProgramaAcademico = { id: 1L, nombrePrograma: "Ingeniería de sistemas", facultad: 1L }</p> <p>Profesor1 = {id: 1L, nombre: "Fabian", apellido: "Parra", tipoIdentificacion: 1L, numeroDocumento: "1008", programaAcademico: 1L}</p> <p>Profesor2 = {id: 2L, nombre: "Giovanni", apellido: "Hernández", tipoIdentificacion: 1L, numeroDocumento: "1009", programaAcademico: 1L}</p>
CID	1, 2, 3 ,4

Tabla 4.

Escenario de prueba historia de usuario 011

Código HU	HU-011
No	4
Descripción	Escenario con dos líneas de investigación
Datos	<p>LineaInvestigacion1 = {id: 1L, línea: "Integración de tecnologías convergentes"}</p> <p>LineaInvestigacion2 = {id: 2L, línea: "Materia y energía: Fundamentos y mecanismos"}</p>
CID	1, 2, 3 ,4

Tabla 5.

Escenario de prueba historia de usuario 012

Código HU	HU-012
No	5
Descripción	Escenario con dos facultades
Datos	Facultad1 = {id: 1L, nombre: "Ingeniería"} Facultad2 = {id: 2L, nombre: "Ciencias de la salud"}
CID	1, 2, 3 ,4

Tabla 6.

Escenario de prueba historia de usuario 015

Código HU	HU-015
No	6
Descripción	Escenario con dos beneficiarios
Datos	Beneficiario1 = {id: 1L, tipo: "Instituciones", descripción: "Instituciones"} Beneficiario2 = {id: 2L, tipo: "Universidades", descripción: "Universidades"}
CID	1, 2, 3 ,4

Tabla 7.

Escenario de prueba historia de usuario 016

Código HU	HU-016
No	7
Descripción	Escenario con dos estados
Datos	Estado1 = {id: 1L, estado: "En progreso", proyectoModalidad: "Proyecto de grado"} Estado2 = {id: 2L, estado: "En revisión", proyectoModalidad: "Proyecto de grado"}
CID	1, 2, 3 ,4

Tabla 8.

Escenario de prueba historia de usuario 017

Código HU	HU-017
No	8
Descripción	Escenario con dos modalidades de grado
Datos	ModalidadGrado1 = {id: 1L, modalidad: "Tesis doctoral"} ModalidadGrado2 = {id: 2L, modalidad: "Práctica empresarial"}
CID	1, 2, 3 ,4

Tabla 9.

Escenario de prueba historia de usuario 018

Código HU	HU-018
No	9
Descripción	Escenario con dos sub modalidades de investigación
Datos	SubModalidad1 = {id: 1L, nombreSubmodalidad: “Joven Investigador”} SubModalidad2 = {id: 2L, nombreSubmodalidad: “Coinvestigador de grupos de investigación profesoral”}
CID	1, 2, 3 ,4

Tabla 10.

Escenario de prueba historia de usuario 019

Código HU	HU-019
No	10
Descripción	Escenario con dos tipos de identificación
Datos	TipoIdentificacion1 = {id: 1L, nombreTipo: “Cédula”} TipoIdentificacion2 = {id: 2L, nombreTipo: “Tarjeta de identidad”}
CID	1, 2, 3 ,4

Tabla 11.

Escenario de prueba historia de usuario 020

Código HU	HU-020
No	11
Descripción	Escenario con dos tipos de producto generado por el proyecto
Datos	TipoProducto1 = {id: 1L, nombre: “Desarrollo de Software”, tipoProductoCategoria: “Desarrollo tecnológico e innovación”} TipoProducto2 = {id: 2L, nombre: “Colección Científica”, tipoProductoCategoria: “Generación de conocimiento”}
CID	1, 2, 3 ,4

Tabla 12.

Escenario de prueba historia de usuario 021

Código HU	HU-021
No	12
Descripción	Escenario con dos coordinaciones
Datos	Coordinacion1 = {id: 1L, nombre: “Coordinación de investigación estudiantil”} Coordinacion2 = {id: 2L, nombre: “Coordinación de semilleros de investigación estudiantil”}
CID	1, 2, 3 ,4

Tabla 13

Escenario de prueba historia de usuario 022

Código HU	HU-022
No	13
Descripción	Escenario con un tipo de identificación, una facultad, dos programas académicos, dos profesores y dos coordinaciones de investigación
Datos	<p>TipoIdentificación = { id: 1L, nombreTipo: "Cédula" }</p> <p>Facultad = { id: 1L, nombreFacultad: "Ingeniería" }</p> <p>ProgramaAcademico1 = { id: 1L, nombrePrograma: "Ingeniería de sistemas", facultad: 1L }</p> <p>ProgramaAcademico2 = { id: 2L, nombrePrograma: "Ingeniería ambiental", facultad: 1L }</p> <p>Profesor1 = {id: 1L, nombre: "Fabian", apellido: "Parra", tipoIdentificacion: 1L, numeroDocumento: "1008", programaAcademico: 1L}</p> <p>Profesor2 = {id: 2L, nombre: "Jose", apellido: "Mafla", tipoIdentificacion: 1L, numeroDocumento: "1009", programaAcademico: 2L}</p> <p>Coordinacion1 = {id: 1L, nombre: "Coordinación de investigación estudiantil"}</p> <p>Coordinacion2 = {id: 2L, nombre: "Coordinación de semilleros de investigación estudiantil"}</p> <p>CoordinacionPrograma1 = {id: 1L, id_coordinacion: 1L, id_programa: 1L, id_coordinador: 1L}</p> <p>CoordinacionPrograma2 = {id: 1L, id_coordinacion: 2L, id_programa: 2L, id_coordinador: 2L}</p>
CID	1, 2, 3 ,4

Anexo G. Casos de prueba

Tabla 1.

Casos de prueba historia de usuario 003

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud estudiantes 1	EstudianteControlIT	debeRetornarUnaLongitudDeDosCuandoHayEstudiantes	HU-003	1	[Estudiante1 = {id: 1L, nombres: "Andres Esteban", apellidos: "Rizo Zamora", tipoIdentificacion: 1L, numeroDocumento: "1193", correoElectronico: arizo@umariana.edu.co, programaAcademico: 1L, semestre: "VIII"}, Estudiante2 = {id: 2L, nombres: "Jeisson Fernando", apellidos: "Caicedo Burbano", tipoIdentificacion: 1L, numeroDocumento: "1008", correoElectronico: jcaicedo@umariana.edu.co, programaAcademico: 1L, semestre: "VIII"}]	Debe retornar una longitud de dos estudiantes
2	Longitud estudiantes 2	EstudianteControlIT	debeRetornarUnaLongitudDeCeroCuandoNoHayEstudiantes		1	Estudiantes = []	Debe retornar una longitud de cero estudiantes
3	Estudiante por id 1	EstudianteControlIT	debeRetornarUnEstudianteCuandoElIdExiste		1	Estudiante1 = {id: 1L}	El estudiante no debe ser null
4	Estudiante por id 2	EstudianteControlIT	debeRetornarNullCuandoElIdNoExiste		1	Estudiante0 = {id: 1000L}	El estudiante debe ser null
5	Crear estudiante 1	EstudianteControlIT	debeCrearUnNuevoEstudiante		1	Estudiante3 = {id: 3L, nombres: "Pepe", apellidos: "Perez", tipoIdentificacion: 1L, numeroDocumento: "1009", correoElectronico: pperez@umariana.edu.co, programaAcademico: 1L, semestre: "VIII"}	Debe retornar los datos del estudiante creado
6	Crear estudiante 2	EstudianteControlIT	debeRetornarNullAlCrearEstudiante		1	Estudiante0 = {}	Debe retornar null

			anteConDatosIn correctos			
7	Actualizar estudiante 1	Estudian teContro llerIT	debeActualizar UnEstudianteEx istente	1	Estudiante3 = {id: 3L, nombres: "Pepe", apellidos: "Perez", tipoIdentificacion: 1L, numeroDocumento: "1009", correoElectronico: pepep@umariana.edu.co, programaAcademico: 1L, semestre: "VIII"}	Debe retornar el estudiante actualizado
8	Actualizar estudiante 2	Estudian teContro llerIT	debeRetornarNu llAlActualizarU nEstudianteInex istente	1	Estudiante0 = {id: 1000L}	Debe retornar null
9	Eliminar estudiante	Estudian teContro llerIT	debeEliminarUn EstudianteExist ente	1	Estudiante1 = {id: 1L}	La longitud de estudiantes debe disminuir en uno

Tabla 2.

Casos de prueba historia de usuario 004

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud proyectos 1	Proyecto Controll erIT	debeRetornarUn aLongitudDeDo sCuandoHayPro yectos	HU-004	2	[Proyecto1 = {id: 1L, fechaInicio: "03-03- 2022", fechaFin: "03-03-2025", proyectoModalidad: "Proyecto de grado", modalidadGrado: null, subModalidad: null, estado: 1L, beneficiario: 1L, política: 1L, financiacion: null, grupoInvestigacion: null, asesor: 1L}, Proyecto2 = {id: 2L, fechaInicio: "03-03- 2021", fechaFin: "03-03-2024", proyectoModalidad: "Proyecto de grado", modalidadGrado: null, subModalidad: null, estado: 2L, beneficiario: 2L, política: 2L, financiacion: null, grupoInvestigacion: null, asesor: 2L}]	Debe retornar una longitud de dos proyectos

2	Longitud proyectos 2	Proyecto Controll erIT	debeRetornarUnaLogintudDeCeroCuandoNoHayProyectos	HU-004	2	Proyectos = []	Debe retornar una longitud de cero proyectos
3	Proyecto por id 1	Proyecto Controll erIT	debeRetornarUnProyectoCuandoElIdExiste	HU-004	2	Proyecto1 = {id: 1L}	El proyecto no debe ser null
4	Proyecto por id 2	Proyecto Controll erIT	debeRetornarNullCuandoElIdNoExiste	HU-004	2	Proyecto0 = {id: 1000L}	El proyecto debe ser null
5	Crear proyecto 1	Proyecto Controll erIT	debeCrearUnNuevoProyecto	HU-004	2	Proyecto3 = {id: 3L, fechaInicio: "03-03-2022", fechaFin: "03-03-2025", proyectoModalidad: "Proyecto de realidad aumentada", modalidadGrado: null, subModalidad: null, estado: 1L, beneficiario: 1L, política: 1L, financiacion: null, grupoInvestigacion: null, asesor: 1L}	Debe retornar los datos del proyecto creado
6	Crear proyecto 2	Proyecto Controll erIT	debeRetornarNullAlCrearProyectoConDatosIncorrectos	HU-004	2	Proyecto0 = {}	Debe retornar null
7	Actualizar proyecto 1	Proyecto Controll erIT	debeActualizarUnProyectoExistente	HU-004	2	Proyecto3 = {id: 3L, fechaInicio: "03-03-2022", fechaFin: "03-12-2025", proyectoModalidad: "Proyecto de realidad aumentada con IA", modalidadGrado: null, subModalidad: null, estado: 1L, beneficiario: 1L, política: 1L, financiacion: null, grupoInvestigacion: null, asesor: 1L}	Debe retornar el proyecto actualizado
8	Actualizar proyecto 2	Proyecto Controll erIT	debeRetornarNullAlActualizarUnProyectoInexistente	HU-004	2	Proyecto = {id: 1000L}	Debe retornar null
9	Eliminar proyecto	Proyecto Controll erIT	debeEliminarUnProyectoExistente	HU-004	2	Proyecto1 = {id: 1L}	La longitud de proyectos debe disminuir en uno

Tabla 3.

Casos de prueba historia de usuario 010

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud asesores 1	Profesor Controll erIT	debeRetornarUnaLongitudDeDosCuandoHayProfesores	HU-010	3	[Profesor1 = {id: 1L, nombre: "Fabian", apellido: "Parra", tipoIdentificacion: 1L, numeroDocumento: "1008", programaAcademico: 1L}, Profesor2 = {id: 2L, nombre: "Giovanni", apellido: "Hernández", tipoIdentificacion: 1L, numeroDocumento: "1009", programaAcademico: 1L}]	Debe retornar una longitud de dos asesores
2	Longitud asesores 2	Profesor Controll erIT	debeRetornarUnaLogintudDeCeroCuandoNoHayAsesores	HU-010	3	Profesores = []	Debe retornar una longitud de cero asesores
3	Asesor por id 1	Profesor Controll erIT	debeRetornarUnProfesorCuandoElIdExiste	HU-010	3	Profesor1 = {id: 1L}	El asesor no debe ser null
4	Asesor por id 2	Profesor Controll erIT	debeRetornarNullCuandoElIdNoExiste	HU-010	3	Profesor0 = {id: 1000L}	El asesor debe ser null
5	Crear asesor 1	Profesor Controll erIT	debeCrearUnNuevoProfesor	HU-010	3	Profesor3 = {id: 3L, nombre: "Danny", apellido: "Cardenas", tipoIdentificacion: 1L, numeroDocumento: "1098", programaAcademico: 1L}	Debe retornar los datos del asesor creado
6	Crear asesor 2	Profesor Controll erIT	debeRetornarNullAlCrearProfesorConDatosIncorrectos	HU-010	3	Proyecto0 = {}	Debe retornar null
7	Actualizar asesor 1	Profesor Controll erIT	debeActualizarUnProfesorExistente	HU-010	3	Proyecto3 = {id: 3L, fechaInicio: "03-03-2022", fechaFin: "03-12-2025", proyectoModalidad: "Proyecto de realidad aumentada con IA", modalidadGrado: null, subModalidad: null, estado: 1L, beneficiario: 1L, política: 1L, financiacion: null, grupoInvestigacion: null, asesor: 1L}	Debe retornar el asesor actualizado

8	Actualizar asesor 2	Profesor ControllerIT	debeRetornarNullActualizarUnProfesorInexistente	HU-010	3	Asesor = {id: 1000L}	Debe retornar null
9	Eliminar asesor	Profesor ControllerIT	debeEliminarUnProfesorExistente	HU-010	3	Asesor1 = {id: 1L}	La longitud de asesores debe disminuir en uno

Tabla 4.

Casos de prueba historia de usuario 011

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud líneas investigación 1	LineaInvestigacionControllerIT	debeRetornarUnaLongitudDeDosCuandoHayLineasInvestigacion	HU-011	4	[LineaInvestigacion1 = {id: 1L, línea: “Integración de tecnologías convergentes”}, LineaInvestigacion2 = {id: 2L, línea: “Materia y energía: Fundamentos y mecanismos”}]	Debe retornar una longitud de dos líneas de investigación
2	Longitud líneas investigación 2	LineaInvestigacionControllerIT	debeRetornarUnaLongitudDeCeroCuandoNoHayLineasInvestigacion	HU-011	4	LineasInvestigacion = []	Debe retornar una longitud de cero líneas de investigación
3	Línea investigación por id 1	LineaInvestigacionControllerIT	debeRetornarUnaLineaInvestigacionCuandoElIdExiste	HU-011	4	LineaInvestigacion1 = {id: 1L}	La línea de investigación no debe ser null
4	Línea investigación por id 2	LineaInvestigacionControllerIT	debeRetornarNullCuandoElIdNoExiste	HU-011	4	LineaInvestigacion0 = {id: 1000L}	La línea de investigación debe ser null
5	Crear línea investigación 1	LineaInvestigacionControllerIT	debeCrearUnNuevoLineaInvestigacion	HU-011	4	LineaInvestigacion3= {id: 3L, línea: “Desarrollo de software”}	Debe retornar los datos de la línea de investigación creada

6	Crear línea investigación 2	LineaInvestigacionControllerIT	debeRetornarNullAlCrearLineaInvestigacionConDatosIncorrectos	HU-011	4	LineaInvestigacion = {}	Debe retornar null
7	Actualizar línea investigación 1	LineaInvestigacionControllerIT	debeActualizarUnaLineaInvestigacionExistente	HU-011	4	LineaInvestigacion3= {id: 3L, línea: "Desarrollo de hardware"}	Debe retornar la línea de investigación actualizada
8	Actualizar línea investigación 2	LineaInvestigacionControllerIT	debeRetornarNullAlActualizarUnaLineaInvestigacionInexistente	HU-011	4	LineaInvestigacion = {id: 1000L}	Debe retornar null
9	Eliminar línea investigación	LineaInvestigacionControllerIT	debeEliminarUnaLineaInvestigacionExistente	HU-011	4	LineaInvestigacion1 = {id: 1L}	La longitud de líneas de investigación debe disminuir en uno

Tabla 5.

Casos de prueba historia de usuario 012

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud facultades 1	FacultadControllerIT	debeRetornarUnaLongitudDeDosCuandoHayFacultades	HU-012	5	[Facultad1 = {id: 1L, nombre: "Ingeniería"}, Facultad2 = {id: 2L, nombre: "Ciencias de la salud"}]	Debe retornar una longitud de dos facultades
2	Longitud facultades 2	FacultadControllerIT	debeRetornarUnaLogintudDeCeroCuandoNoHayFacultades	HU-012	5	Facultad = []	Debe retornar una longitud de cero facultades
3	Facultad por id 1	FacultadControllerIT	debeRetornarUnaFacultadCuandoElIdExiste	HU-012	5	Facultad1= {id: 1L}	La facultad no debe ser null

4	Facultad por id 2	Facultad ControllerIT	debeRetornarNullCuandoElIdNoExiste	HU-012	5	Facultad0= {id: 1000L}	La facultad debe ser null
5	Crear facultad 1	Facultad ControllerIT	debeCrearUnNuevoFacultad	HU-012	5	Facultad3 = {id: 3L, nombre: "Ciencias contables"}	Debe retornar los datos de la facultad creada
6	Crear facultad 2	Facultad ControllerIT	debeRetornarNullAlCrearFacultadConDatosIncorrectos	HU-012	5	Facultad= {}	Debe retornar null
7	Actualizar facultad 1	Facultad ControllerIT	debeActualizarUnFacultadExistente	HU-012	5	Facultad3 = {id: 3L, nombre: "Ciencias contables y administrativas"}	Debe retornar la facultad actualizada
8	Actualizar facultad 2	Facultad ControllerIT	debeRetornarNullAlActualizarUnFacultadInexistente	HU-012	5	Facultad = {id: 1000L}	Debe retornar null
9	Eliminar facultad	Facultad ControllerIT	debeEliminarUnFacultadExistente	HU-012	5	Facultad1 = {id: 1L}	La longitud de facultades debe disminuir en uno

Tabla 6.

Casos de prueba historia de usuario 015

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud beneficiarios 1	BeneficiarioControllerIT	debeRetornarUnaLongitudDeDosCuandoHayBeneficiarios	HU-015	6	[Beneficiario1 = {id: 1L, tipo: "Instituciones", descripción: "Instituciones"}, Beneficiario2 = {id: 2L, tipo: "Universidades", descripción: "Universidades"}]	Debe retornar una longitud de dos beneficiarios
2	Longitud beneficiarios 2	BeneficiarioControllerIT	debeRetornarUnaLongitudDeCeroCuandoNoHayBeneficiarios	HU-015	6	Beneficiarios = []	Debe retornar una longitud de cero beneficiarios

3	Beneficiario por id 1	BeneficiarioControllerIT	debeRetornarUnBeneficiarioCuandoElIdExiste	HU-015	6	Beneficiario1 = {id: 1L}	El beneficiario no debe ser null
4	Beneficiario por id 2	BeneficiarioControllerIT	debeRetornarNullCuandoElIdNoExiste	HU-015	6	Beneficiario0 = {id: 1000L}	El beneficiario debe ser null
5	Crear beneficiario 1	BeneficiarioControllerIT	debeCrearUnNuevoBeneficiario	HU-015	6	Beneficiario3 = {id: 3L, tipo: "Gobernación", descripción: "Gobernación"}	Debe retornar los datos del beneficiario creado
6	Crear beneficiario 2	BeneficiarioControllerIT	debeRetornarNullAlCrearBeneficiarioConDatosIncorrectos	HU-015	6	Beneficiario0 = {}	Debe retornar null
7	Actualizar beneficiario 1	BeneficiarioControllerIT	debeActualizarUnBeneficiarioExistente	HU-015	6	Beneficiario3 = {id: 3L, tipo: "Alcaldía", descripción: "Alcaldía"}	Debe retornar el beneficiario actualizado
8	Actualizar beneficiario 2	BeneficiarioControllerIT	debeRetornarNullAlActualizarUnBeneficiarioInexistente	HU-015	6	Beneficiario = {id: 1000L}	Debe retornar null
9	Eliminar beneficiario	BeneficiarioControllerIT	debeEliminarUnBeneficiarioExistente	HU-015	6	Beneficiario1 = {id: 1L}	La longitud de beneficiarios debe disminuir en uno

Tabla 44

Casos de prueba historia de usuario 016

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud estados 1	EstadoControllerIT	debeRetornarUnaLongitudDeDo	HU-016	7	[Estado1 = {id: 1L, estado: "En progreso", proyectoModalidad: "Proyecto de grado"}]	Debe retornar una longitud de dos estados

			sCuandoHayEstados			Estado2 = {id: 2L, estado: "En revisión", proyectoModalidad: "Proyecto de grado"}]	
2	Longitud estados 2	EstadoController IT	debeRetornarUnaLongitudDeCeroCuandoNoHayEstados	HU-016	7	Estados = []	Debe retornar una longitud de cero estados
3	Estado por id 1	EstadoController IT	debeRetornarUnEstadoCuandoElIdExiste	HU-016	7	Estado1 = {id: 1L}	El estado no debe ser null
4	Estado por id 2	EstadoController IT	debeRetornarNullCuandoElIdNoExiste	HU-016	7	Estado0 = {id: 1000L}	El estado debe ser null
5	Crear estado 1	EstadoController IT	debeCrearUnNuevoEstado	HU-016	7	Estado3 = {id: 3L, estado: "Finalizado", proyectoModalidad: "Proyecto de grado"}	Debe retornar los datos del estado creado
6	Crear estado 2	EstadoController IT	debeRetornarNullAlCrearEstadoConDatosIncorrectos	HU-016	7	Estado0 = {}	Debe retornar null
7	Actualizar estado 1	EstadoController IT	debeActualizarUnEstadoExistente	HU-016	7	Estado3 = {id: 3L, estado: "Terminado", proyectoModalidad: "Proyecto de grado"}	Debe retornar el estado actualizado
8	Actualizar estado 2	EstadoController IT	debeRetornarNullAlActualizarUnEstadoInexistente	HU-016	7	Estado0 = {id: 1000L}	Debe retornar null
9	Eliminar estado	EstadoController IT	debeEliminarUnEstadoExistente	HU-016	7	Estado1 = {id: 1L}	La longitud de estados debe disminuir en uno

Tabla 45

Casos de prueba historia de usuario 017

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
-------	--------	-------	--------	----	-----------	--------------------	--------------------

1	Longitud facultades 1	ModalidadGradoControlerIT	debeRetornarUnaLongitudDeDosCuandoHayModalidadesGrado	HU.0-17	8	[ModalidadGrado1 = {id: 1L, modalidad: "Tesis doctoral"}, ModalidadGrado2 = {id: 2L, modalidad: "Práctica empresarial"}]	Debe retornar una longitud de dos modalidades
2	Longitud modalidades 2	ModalidadGradoControlerIT	debeRetornarUnaLogintudDeCeroCuandoNoHayModalidades	HU.0-17	8	ModalidadGrado = []	Debe retornar una longitud de cero modalidades
3	Modalidad por id 1	ModalidadGradoControlerIT	debeRetornarUnaModalidadGradoCuandoElIdExiste	HU.0-17	8	ModalidadGrado1 = {id: 1L}	La modalidad no debe ser null
4	Modalidad por id 2	ModalidadGradoControlerIT	debeRetornarNullCuandoElIdNoExiste	HU.0-17	8	ModalidadGrado0 = {id: 1000L}	La modalidad debe ser null
5	Crear modalidad 1	ModalidadGradoControlerIT	debeCrearUnNuevoModalidad	HU.0-17	8	ModalidadGrado3 = {id: 1L, modalidad: "Tesis pregrado"}	Debe retornar los datos de la modalidad creada
6	Crear modalidad 2	ModalidadGradoControlerIT	debeRetornarNullAlCrearModalidadConDatosIncorrectos	HU.0-17	8	ModalidadGrado = {}	Debe retornar null
7	Actualizar modalidad 1	ModalidadGradoControlerIT	debeActualizarUnaModalidadExistente	HU.0-17	8	ModalidadGrado3 = {id: 1L, modalidad: "Tesis profesional"}	Debe retornar la modalidad actualizada
8	Actualizar modalidad 2	ModalidadGradoControlerIT	debeRetornarNullAlActualizarUnaModalidadInexistente	HU.0-17	8	ModalidadGrado = {id: 1000L}	Debe retornar null
9	Eliminar modalidad	ModalidadGradoControlerIT	debeEliminarUnaModalidadExistente	HU.0-17	8	ModalidadGrado1 = {id: 1L}	La longitud de modalidades debe disminuir en uno

Tabla 46

Casos de prueba historia de usuario 018

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud sub-modalidades 1	SubModalidadControllerIT	debeRetornarUnaLongitudDeDosCuandoHaySubModalidades	HU-018	9	[SubModalidad1 = {id: 1L, nombreSubmodalidad: "Joven Investigador"}, SubModalidad2 = {id: 2L, nombreSubmodalidad: "Coinvestigador de grupos de investigación profesoral"}]	Debe retornar una longitud de dos sub-modalidades
2	Longitud submodalidades 2	SubModalidadControllerIT	debeRetornarUnaLogintudDeCeroCuandoNoHaySubModalidades	HU-018	9	SubModalidad = []	Debe retornar una longitud de cero sub-modalidades
3	Sub-modalidad por id 1	SubModalidadControllerIT	debeRetornarUnaSubModalidadCuandoElIdExiste	HU-018	9	SubModalidad1 = {id: 1L}	La sub-modalidad no debe ser null
4	Sub-modalidad por id 2	SubModalidadControllerIT	debeRetornarNullCuandoElIdNoExiste	HU-018	9	SubModalidad0 = {id: 1000L}	La sub-modalidad debe ser null
5	Crear sub-modalidad 1	SubModalidadControllerIT	debeCrearUnaNuevaSubModalidad	HU-018	9	SubModalidad3 = {id: 3L, nombreSubmodalidad: "Auxiliares"}	Debe retornar los datos de la sub-modalidad creada
6	Crear sub-modalidad 2	SubModalidadControllerIT	debeRetornarNullAlCrearUnaSubModalidadConDatosIncorrectos	HU-018	9	SubModalidad = { }	Debe retornar null
7	Actualizar sub-modalidad 1	SubModalidadControllerIT	debeActualizarUnaSubModalidadExistente	HU-018	9	SubModalidad3 = {id: 3L, nombreSubmodalidad: "Estudiantes Auxiliares"}	Debe retornar la sub-modalidad actualizada

8	Actualizar sub-modalidad 2	SubModalidadControllerIT	debeRetornarNullAlActualizarUnaSubModalidadInexistente	HU-018	9	SubModalidad = {id: 1000L}	Debe retornar null
9	Eliminar submodalidad	SubModalidadControllerIT	debeEliminarUnaSubModalidadExistente	HU-018	9	SubModalidad1 = {id: 1L}	La longitud de sub-modalidades debe disminuir en uno

Tabla 47

Casos de prueba historia de usuario 019

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud tipos identificación 1	TipoIdentificacionControllerIT	debeRetornarUnaLongitudDeDosCuandoHayTiposIdentificacion	HU-019	10	[TipoIdentificacion1 = {id: 1L, nombreTipo: "Cédula"}, TipoIdentificacion2 = {id: 2L, nombreTipo: "Tarjeta de identidad"}]	Debe retornar una longitud de dos tipos identificación
2	Longitud tipos identificación 2	TipoIdentificacionControllerIT	debeRetornarUnaLogintudDeCeroCuandoNoHayTiposDeIdentificacion	HU-019	10	TipoIdentificacion = []	Debe retornar una longitud de cero tipos identificación
3	Tipo identificación por id 1	TipoIdentificacionControllerIT	debeRetornarUnTipoIdentificacionCuandoElIdExiste	HU-019	10	TipoIdentificacion1 = {id: 1L}	El tipo identificación no debe ser null
4	Tipo identificación por id 2	TipoIdentificacionControllerIT	debeRetornarNullCuandoElIdNoExiste	HU-019	10	TipoIdentificacion0 = {id: 1000L}	El tipo identificación debe ser null
5	Crear tipo identificación 1	TipoIdentificacionControllerIT	debeCrearUnNuevoTipoDeIdentificacion	HU-019	10	TipoIdentificacion3 = {id: 3L, nombreTipo: "Pasaporte"}	Debe retornar los datos del tipo identificación creado

6	Crear tipo identificación 2	TipoIdentificacionControllerIT	debeRetornarNullAlCrearTipoDeIdentificacionConDatosIncorrectos	HU-019	10	TipoIdentificacion0 = {}	Debe retornar null
7	Actualizar tipo identificación 1	TipoIdentificacionControllerIT	debeActualizarUnTipoDeIdentificacionExistente	HU-019	10	TipoIdentificacion3 = {id: 3L, nombreTipo: "Cédula de extranjería"}	Debe retornar el tipo identificación actualizado
8	Actualizar tipo identificación 2	TipoIdentificacionControllerIT	debeRetornarNullAlActualizarUnTipoDeIdentificacionInexistente	HU-019	10	TipoIdentificacion0 = {id: 1000L}	Debe retornar null
9	Eliminar tipo identificación	TipoIdentificacionControllerIT	debeEliminarUnTipoDeIdentificacionExistente	HU-019	10	TipoIdentificacion1 = {id: 1L}	La longitud de tipos identificación debe disminuir en uno

Tabla 48

Casos de prueba historia de usuario 020

CPIId	Nombre	Clase	Método	HU	Escenario	Valores de entrada	Resultado Esperado
1	Longitud tipos identificación 1	TipoProductoControllerIT	debeRetornarUnaLongitudDeDosCuandoHayTiposProducto	HU-020	11	[TipoProducto1 = {id: 1L, nombre: "Desarrollo de Software", tipoProductoCategoria: "Desarrollo tecnológico e innovación"}, TipoProducto2 = {id: 2L, nombre: "Colección Científica", tipoProductoCategoria: "Generación de conocimiento"}]	Debe retornar una longitud de dos tipos producto
2	Longitud tipos identificación 2	TipoProductoControllerIT	debeRetornarUnaLogintudDeCeroCuandoNoHayTiposDeProducto	HU-020	11	TipoProducto = []	Debe retornar una longitud de cero tipos producto

3	Tipo producto por id 1	TipoProductoControllerIT	debeRetornarUnTipoProductoCuandoElIdExiste	HU-020	11	TipoProducto1 = {id: 1L}	El tipo producto no debe ser null
4	Tipo producto por id 2	TipoProductoControllerIT	debeRetornarNullCuandoElIdNoExiste	HU-020	11	TipoProducto0 = {id: 1000L}	El tipo producto debe ser null
5	Crear tipo producto 1	TipoProductoControllerIT	debeCrearUnNuevoTipoDeProducto	HU-020	11	TipoProducto3 = {id: 3L, nombre: "Página web", tipoProductoCategoria: "Desarrollo tecnológico e innovación"}	Debe retornar los datos del tipo producto creado
6	Crear tipo producto 2	TipoProductoControllerIT	debeRetornarNullAlCrearTipoDeProductoConDatosIncorrectos	HU-020	11	TipoProducto0 = {}	Debe retornar null
7	Actualizar tipo producto 1	TipoProductoControllerIT	debeActualizarUnTipoDeProductoExistente	HU-020	11	TipoProducto3 = {id: 3L, nombre: "Plataforma web", tipoProductoCategoria: "Desarrollo tecnológico e innovación"}	Debe retornar el tipo producto actualizado
8	Actualizar tipo producto 2	TipoProductoControllerIT	debeRetornarNullAlActualizarUnTipoDeProductoInexistente	HU-020	11	TipoProducto0 = {id: 1000L}	Debe retornar null
9	Eliminar tipo producto	TipoProductoControllerIT	debeEliminarUnTipoDeProductoExistente	HU-020	11	TipoProducto1 = {id: 1L}	La longitud de tipos producto debe disminuir en uno

Anexo H. Resultados pruebas unitarias

Figura 1

Resultados test controlador estudiantes

```
Run [org.umariana.simar.controller] 337ms
Tests passed: 9 of 9 tests - 337ms
EstudianteControllerIT (org.umariana.simar.controller)
  setUp() 214ms
  cuandoHayEstudiantes() 4ms
  cuandoNoHayEstudiantes() 7ms
  cuandoExiste() 7ms
  cuandoNoExiste() 16ms
  cuandoCrearEstudianteConDatosIncorrectos() 4ms
  cuandoActualizarEstudianteExiste() 4ms
  cuandoActualizarEstudianteNoExiste() 4ms
  cuandoEliminarEstudianteExiste() 4ms
```

Figura 2

Resultados test controlador estado

```
Run [org.umariana.simar.controller] 292ms
Tests passed: 9 of 9 tests - 292ms
EstadoControllerIT (org.umariana.simar.controller)
  setUp() 180ms
  cuandoHayEstados() 7ms
  cuandoNoHayEstados() 7ms
  cuandoExiste() 7ms
  cuandoNoExiste() 14ms
  cuandoCrearEstadoConDatosIncorrectos() 4ms
  cuandoActualizarEstadoExiste() 4ms
  cuandoActualizarEstadoNoExiste() 4ms
  cuandoEliminarEstadoExiste() 4ms
```

Figura 3

Resultados test controlador facultad

```
Run [org.umariana.simar.controller] 304ms
Tests passed: 9 of 9 tests - 304ms
FacultadControllerIT (org.umariana.simar.controller)
  setUp() 204ms
  cuandoHayFacultades() 7ms
  cuandoNoHayFacultades() 7ms
  cuandoExiste() 7ms
  cuandoNoExiste() 12ms
  cuandoCrearFacultadConDatosIncorrectos() 4ms
  cuandoActualizarFacultadExiste() 4ms
  cuandoActualizarFacultadNoExiste() 4ms
  cuandoEliminarFacultadExiste() 4ms
```

Figura 4

Resultados test controlador línea de investigación

```
Run [org.umariana.simar.controller] 290ms
Tests passed: 9 of 9 tests - 290ms
LineaInvestigacionControllerIT (org.umariana.simar.controller)
  setUp() 200ms
  cuandoHayLineasInvestigacion() 4ms
  cuandoNoHayLineasInvestigacion() 4ms
  cuandoExiste() 7ms
  cuandoNoExiste() 11ms
  cuandoCrearLineaInvestigacionConDatosIncorrectos() 4ms
  cuandoActualizarLineaInvestigacionExiste() 4ms
  cuandoActualizarLineaInvestigacionNoExiste() 4ms
  cuandoEliminarLineaInvestigacionExiste() 4ms
```

Figura 5

Resultados test controlador modalidad grado

Figura 6
Resultados test controlador profesor

Figura 7
Resultados test controlador proyecto

Figura 8
Resultados test controlador submodalidad

Figura 9
Resultados test controlador tipo identificación

Anexo I. Escenarios flujos de pruebas E2E

CPIId	Descripción del escenario	Pasos de la prueba	Resultado Esperado
1	Acceso a la página inicial de la plataforma SIUMAR	- Navegar a la página de la plataforma a través de la url http://localhost:4200	Se carga la página principal mostrando el login con Google
2	Inicio de sesión a través de google con una cuenta perteneciente a la Universidad Mariana	- Navegar a la página de la plataforma - Clickear el botón acceder con Google - Seleccionar la cuenta de inicio perteneciente a la Universidad Mariana	Se redirige a la página principal de la plataforma al usuario autorizado mediante las credenciales de Google
3	Redirección a la pantalla de acceso denegado	- Navegar a la página de la plataforma - Clickear el botón acceder con Google - Seleccionar la cuenta de inicio perteneciente a la Universidad Mariana	Se redirige a la página acceso denegado al usuario no autorizado
4	Navegar a través de las opciones de menú filtradas por los privilegios del usuario	- Iniciar sesión con Google en una cuenta autorizada - Navegar en las diferentes opciones del menú lateral	El usuario puede acceder a las secciones del menú que corresponden a los privilegios de su perfil
5	Listar el programa académico al que pertenece el usuario en sesión	- Dirigirse a la sección programas	El programa listado debe ser el mismo al que pertenece el usuario en sesión
6	Registrar un nuevo grupo de investigación	- Dirigirse a la sección programas - Visualizar los grupos de investigación del programa listado - Agregar un nuevo grupo de investigación al programa	El usuario puede agregar un nuevo grupo de investigación en el programa listado
7	Listar los profesores del mismo programa del usuario en sesión	- Dirigirse a la sección profesores	La lista mostrada debe contener solo profesores del mismo programa del usuario en sesión
8	Registrar un nuevo profesor	- Dirigirse a la sección profesores - Completar el formulario para agregar un nuevo profesor	El nuevo profesor es agregado y aparece en la lista de profesores

		- Confirmar el registro del nuevo profesor	
9	Listar los estudiantes del mismo programa del usuario en sesión	- Dirigirse a la sección estudiantes	La lista mostrada debe contener solo estudiantes del mismo programa del usuario en sesión
10	Registrar un nuevo estudiante	- Dirigirse a la sección estudiantes - Completar el formulario para agregar un nuevo estudiante - Confirmar el registro del nuevo estudiante	El nuevo estudiante es agregado y aparece en la lista de estudiantes
11	Listar las líneas de investigación disponibles para el programa del usuario en sesión	- Dirigirse a la sección líneas de investigación	La lista mostrada debe contener solo líneas de investigación disponibles para el programa del usuario en sesión
12	Registrar una nueva línea de investigación para el programa del usuario en sesión	- Dirigirse a la sección líneas de investigación - Completar el formulario para agregar una nueva línea de investigación - Confirmar el registro de la nueva línea de investigación	La nueva línea de investigación es agregada y aparece en la lista de líneas de investigación
13	Listar los estados del proyecto disponibles para el programa del usuario en sesión	- Dirigirse a la sección estados del proyecto	La lista mostrada debe contener solo estados del proyecto disponibles para el programa del usuario en sesión
14	Registrar un nuevo estado del proyecto para el programa del usuario en sesión	- Dirigirse a la sección estados del proyecto - Completar el formulario para agregar un nuevo estado del proyecto - Confirmar el registro del nuevo estado del proyecto	El nuevo estado del proyecto es agregado y aparece en la lista de estados del proyecto
15	Listar los beneficiarios disponibles para los proyectos del programa del usuario en sesión	- Dirigirse a la sección beneficiarios	- La lista mostrada debe contener solo beneficiarios disponibles para los proyectos del programa del usuario en sesión

16	- Registrar un nuevo beneficiario para los proyectos del programa del usuario en sesión	- Dirigirse a la sección beneficiarios - Completar el formulario para agregar un nuevo beneficiario - Confirmar el registro del nuevo beneficiario	El nuevo beneficiario es agregado y aparece en la lista de beneficiarios
17	Listar los tipos de producto resultados disponibles para los proyectos del programa del usuario en sesión	- Dirigirse a la sección tipos de producto resultados	La lista mostrada debe contener solo tipos de producto resultados disponibles para los proyectos del programa del usuario en sesión
18	Registrar un nuevo tipo de producto resultado para los proyectos del programa del usuario en sesión	- Dirigirse a la sección tipos de producto resultados - Completar el formulario para agregar un nuevo tipo de producto resultado - Confirmar el registro del nuevo tipo de producto resultado	El nuevo tipo de producto resultado es agregado y aparece en la lista de tipos de producto resultados
19	Listar los proyectos que pertenecen a la coordinación de investigación y programa académico del usuario en sesión	- Dirigirse a la sección proyectos	La lista mostrada debe contener solo proyectos que pertenecen a la coordinación y programa académico del usuario en sesión
20	Debería poder registrar un nuevo proyecto	- Dirigirse a la sección proyectos - Completar el formulario para agregar un nuevo proyecto - Confirmar el registro del nuevo proyecto	El nuevo proyecto es agregado y aparece en la lista de proyectos
21	Debería poder cerrar mi sesión	- Dirigirse a la página principal - Clickear el botón para cerrar sesión	El usuario es redirigido a la página de inicio de sesión

Anexo J. Plataforma web SIUMAR

Figura 1

Página de login



Figura 2

Página inicial del coordinador

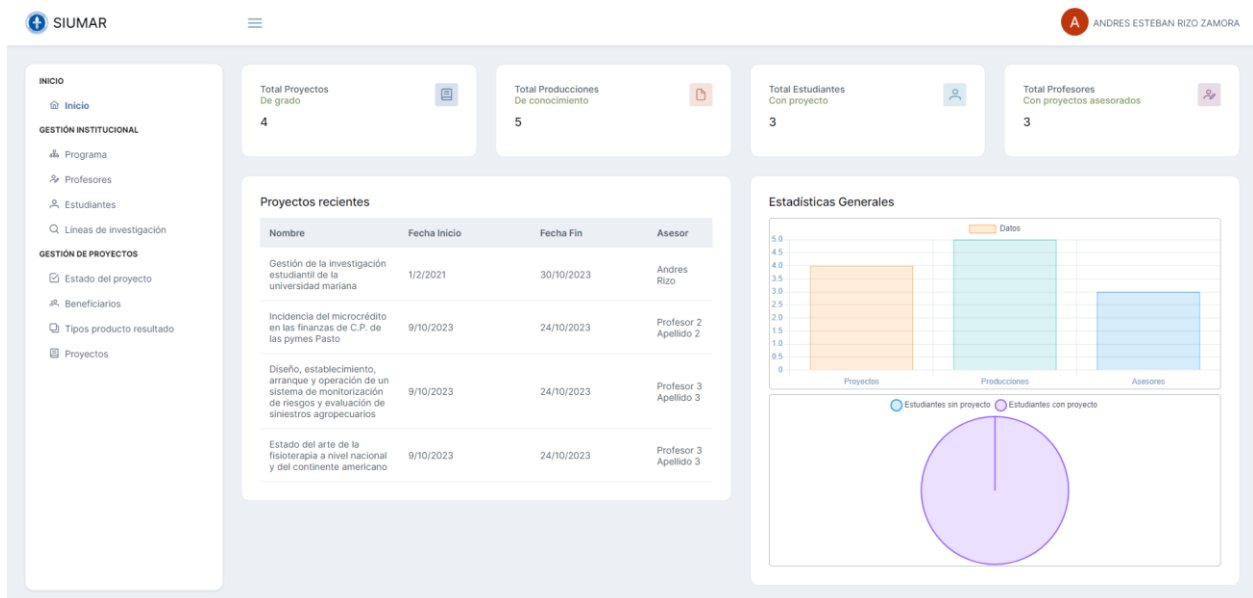


Figura 3

Sección estudiantes

The screenshot shows the 'Sección estudiantes' interface. On the left is a sidebar with a search bar and navigation menu. The main area displays a table of students with the following data:

Nombres	Apellidos	Tipo identificación	Número documento	Correo electrónico	Número de celular	Programa	Semestre	
Estudiante 1	Apellido 1	Cédula	123222	arizo@gmail.com	213	Ingeniería de sistemas	Cuarto	
Estudiante 2	Apellido 2	Cédula	111142	mips@gmail.com	213	Ingeniería de sistemas	Noveno	
Estudiante 3	Apellido 3	Cédula	11167	fpara@umariana.edu.co	3122	Ingeniería de sistemas	Noveno	

Below the table, it states: 'En total hay 3 estudiantes.' The page number is 'Página 1 de 1'.

Figura 4

Sección proyectos

The screenshot shows the 'Sección proyectos' interface. On the left is a sidebar with a search bar and navigation menu. The main area displays a table of projects with the following data:

Nombre	Fecha inicio	Fecha fin	Modalidad	Estado	Estudiantes	Asesor
Gestión de la investigación estudiantil de la universidad mariana	1/2/2021	30/10/2023	Proyecto de grado	En avances		Andres Rizo
Incidenca del microcrédito en las finanzas de C.P. de las pymes Pasto	9/10/2023	24/10/2023	Proyecto de grado	En avances		Profesor 2 Apellido 2
Diseño, establecimiento, arranque y operación de un sistema de monitorización de riesgos y evaluación de siniestros agropecuarios	9/10/2023	24/10/2023	Proyecto de grado	En avances		Profesor 3 Apellido 3
Estado del arte de la fisioterapia a nivel nacional y del continente americano	9/10/2023	24/10/2023	Proyecto de grado	En avances		Profesor 3 Apellido 3

Below the table, it states: 'En total hay 4 proyectos.' The page number is 'Página 1 de 1'.

Figura 5

Edición proyectos

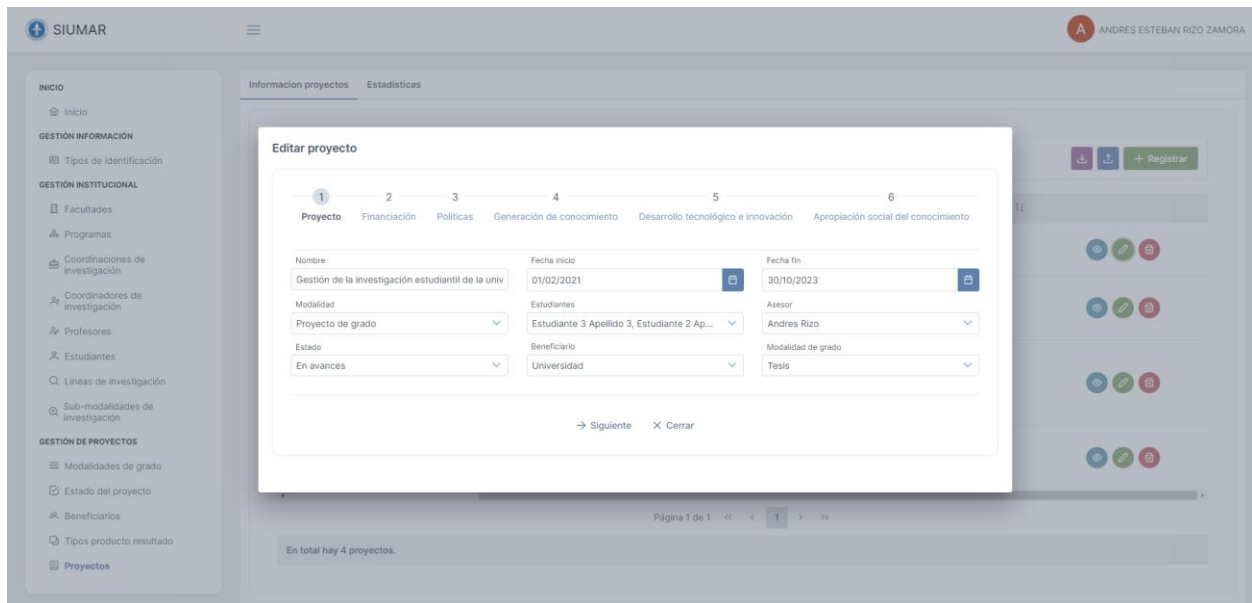


Figura 6

Sección gráficos estadísticos

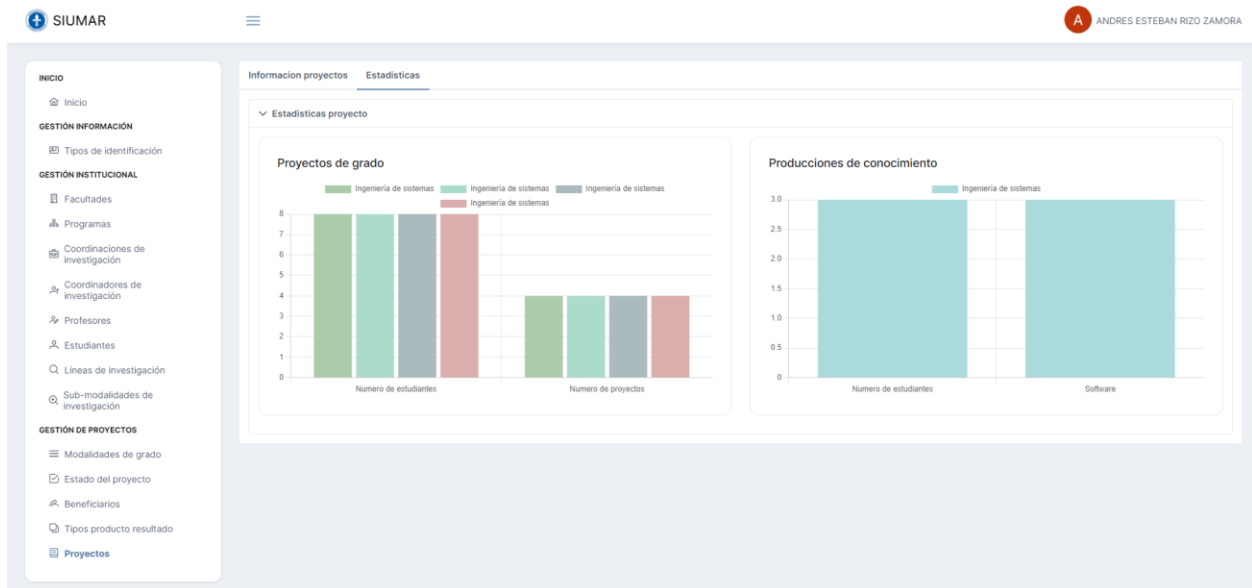


Figura 7

Detalles perfil de usuario

The screenshot shows the user profile details page. The sidebar on the left contains navigation options under 'INICIO', 'GESTIÓN INFORMACIÓN', 'GESTIÓN INSTITUCIONAL', and 'GESTIÓN DE PROYECTOS'. The main content area displays the profile of 'Andres Rizo' with the following details:

Coordinador	Correo electrónico	Programa	Coordinación
Andres Rizo	andres.rizo@umariana.edu.co	Ingeniería de sistemas	Coordinación de investigación

Below the table, it states 'En total hay 1 coordinadores.' A dropdown menu is open, showing details for 'Coordinador de investigación estudiantil':

- Facultad: Ingeniería
- Programa: Ingeniería de sistemas

Buttons for 'Cerrar sesión' and 'Editar' are visible.

Figura 8

Filtros de información

The screenshot shows the filters page. The sidebar on the left contains navigation options under 'INICIO', 'GESTIÓN INSTITUCIONAL', and 'GESTIÓN DE PROYECTOS'. The main content area displays a table of filters for 'Profesores' with the following columns:

Nombres	Apellidos	Tipo identificación	Número documento	Correo Electrónico	Programa
Andres	Rizo	Cédula	1193	andres.rizo@umariana.edu.co	Ingeniería de sistemas
Profesor 2	Apellido 2	Cédula	123222	jrizo@gmail.com	Ingeniería de sistemas
Profesor 3	Apellido 3	Cédula	1111	arizo@gmail.com	Ingeniería de sistemas
Profesor 4	Apellido 4	Cédula	11112	pepe@gmail.com	Ingeniería de sistemas

Below the table, it states 'En total hay 4 profesores.' Buttons for 'Registrar', 'Editar', and 'Eliminar' are visible.