

Informe de Trabajo Final



Universidad Peruana de Ciencias Aplicadas

Ingeniería de Software

6to Ciclo

Código del curso: 1ACC0238

Nombre del curso: Aplicaciones para Dispositivos Móviles

NRC: 1795

Nombre del profesor: Jorge Luis Mayta Guillermo

Nombre del Startup: UniPafi

Nombre del Producto: InnoSpace

Relación de Integrantes

Código	Apellidos y Nombres
U20211B293	Cabrera Buitron, Diego Ivan
U202313172	Coca Lavado, Carlos Andres
U20231A500	Palomino Fiestas, Erick Leonardo
U20231D974	Rivera Ratachi, Renzo Sebastian
U202310988	Santur Tello, Andrea Elizabeth

Agosto 2025

Contenido

- [Student Outcome](#)
- [Capítulo I: Introducción](#)
 - [1.1. Startup Profile](#)
 - [1.1.1. Descripción de la Startup](#)
 - [1.1.2. Perfiles de integrantes del equipo](#)

- 1.2. Solution Profile
 - 1.2.1. Antecedentes y problemática
 - 1.2.2. Lean UX Process
 - 1.2.2.1. Lean UX Problem Statements
 - 1.2.2.2. Lean UX Assumptions
 - 1.2.2.3. Lean UX Hypothesis Statements
 - 1.2.2.4. Lean UX Canvas
- 1.3. Segmentos objetivo
- Capítulo II: Requirements Development and Software Solution Design
 - 2.1. Competidores
 - 2.1.1. Análisis competitivo
 - 2.1.2. Estrategias y tácticas frente a competidores
 - 2.2. Entrevistas
 - 2.2.1. Diseño de entrevistas
 - 2.2.2. Registro de entrevistas
 - 2.2.3. Análisis de entrevistas
 - 2.3. Needfinding
 - 2.3.1. User Personas
 - 2.3.2. User Task Matrix
 - 2.3.3. User Journey Mapping
 - 2.3.4. Empathy Mapping
 - 2.3.5. Ubiquitous Language
 - 2.4. Requirements Specification
 - 2.4.1. User Stories
 - 2.4.2. Impact Mapping
 - 2.4.3. Product Backlog
 - 2.5. Strategic-Level Domain-Driven Design
 - 2.5.1. EventStorming
 - 2.5.1.1. Candidate Context Discovery
 - 2.5.1.2. Domain Message Flows Modeling
 - 2.5.1.3. Bounded Context Canvases
 - 2.5.2. Context Mapping
 - 2.5.3. Software Architecture
 - 2.5.3.1. Software Architecture Context Level Diagrams
 - 2.5.3.2. Software Architecture Container Level Diagrams
 - 2.5.3.3. Software Architecture Deployment Diagrams
 - 2.6. Tactical-Level Domain-Driven Design
 - 2.6.X. Bounded Context:
 - 2.6.X.1. Domain Layer
 - 2.6.X.2. Interface Layer

- 2.6.X.3. Application Layer
 - 2.6.X.4. Infrastructure Layer
 - 2.6.X.5. Bounded Context Software Architecture Component Level Diagrams
 - 2.6.X.6. Bounded Context Software Architecture Code Level Diagrams
 - 2.6.X.6.1. Bounded Context Domain Layer Class Diagrams
 - 2.6.X.6.2. Bounded Context Database Design Diagram
 - Bibliografía
 - Anexos
-

Objetivos SMART

Coca Lavado Carlos Andrés

Objetivo 1: Desarrollar y lanzar una interfaz de usuario intuitiva para la plataforma InnoSpace con el objetivo de mejorar la experiencia de los usuarios, asegurando que los usuarios registrados tengan una navegación intuitiva y fluida en la aplicación.

Objetivo 2: Adquirir conocimientos y experiencia en el desarrollo de aplicaciones móviles, con el fin de seguir mejorando respecto a la accesibilidad y experiencia de usuario, mientras continúo desarrollando mis habilidades técnicas y profesionales en el ámbito del desarrollo móvil.

Capítulo I: Introducción

1.1. Startup Profile

1.1.1. Descripción de la Startup

Unipafi es una startup tecnológica que busca convertirse en un referente en la vinculación entre el talento joven y el ecosistema empresarial. Nacemos con la convicción de que la innovación no surge únicamente en los laboratorios corporativos, sino también en las aulas, los proyectos académicos y las ideas emergentes de estudiantes con visión de futuro. Desde Unipafi impulsamos un modelo de colaboración que une creatividad, impacto social y desarrollo profesional, construyendo un espacio en el que la innovación y el talento se encuentren para transformar realidades.

Visión:

En 4 años, consolidarnos como la startup líder en Latinoamérica en la creación de puentes entre universidades y empresas, reconocidos por fomentar el talento emergente, la innovación colaborativa y el crecimiento sostenible.

Misión:

Promover el desarrollo profesional de estudiantes universitarios y recién egresados a través de la creación de espacios donde puedan conectar con organizaciones, demostrar sus capacidades y generar soluciones de alto impacto que aporten al progreso económico y social.

1.1.2. Perfiles de integrantes del equipo

Fotos del Integrantes	Nombres y Apellidos	Código de estudiante	Conocimiento técnicos y habilidades
	Palomino Fiestas, Erick Leonardo	U20231A500	Hola, Soy Erick Leonardo Palomino Fiestas, un estudiante entusiasta de quinto ciclo en Ingeniería de Software. Mi conjunto de habilidades técnicas incluye C++, HTML, CSS, fundamentos de JavaScript y experiencia con bases de datos SQL y MongoDB. Me considero una persona responsable, con sólidas habilidades de comunicación para trabajar en equipo y una perspectiva optimista. También tengo un nivel básico de inglés.
	Renzo Sebastian Rivera Ratachi	U20231D974	Soy Renzo Sebastian Rivera Ratachi y soy estudiante de la carrera de Ingeniería de Software. Actualmente estoy cursando el 6to ciclo de mi carrera y tengo conocimientos intermedios de HTML, CSS, JavaScript y C++. Me considero una persona responsable y puntual.
	Carlos Andres Coca Lavado	U202313172	Mi Nombre es Carlos Andrés Coca, tengo 19 años, actualmente me encuentro cursando el sexto ciclo de la carrera de Ingeniería de Software. Cuento con conocimientos en C++, Python y HTML. y desde muy joven me ha interesado la ciberseguridad y el desarrollo de Software. Teniendo en cuenta el gran impacto que presentan a día de hoy.
	Andrea Elizabeth Santur Tello	U202310988	Estoy cursando el sexto ciclo de mi carrera Ingeniería de Software, soy una persona responsable que le gusta resolver desafíos a la par con el trabajo responsable y en equipo tengo la capacidad de líder y me gusta aprender nuevas cosas dia a dia.
	Diego Ivan Cabrera Buitron	U20211B293	Estoy cursando el sexto ciclo de mi carrera Ingeniería de Software, soy una persona responsable que le gusta resolver desafíos a la par con el trabajo responsable y en equipo tengo la capacidad de líder y me gusta aprender nuevas cosas dia a dia.

1.2. Solution Profile

1.2.1. Antecedentes y problemática

1.2.2. Lean UX Process

1.2.2.1. Lean UX Problem Statements

El propóstivo de InnoBridge es ser el puente entre el mundo académico y el empresarial en el Perú.

En el cual, por un lado, los estudiantes y profesores universitarios, quienes podrán subir proyectos, investigaciones, tesis o ideas innovadoras. Además, podrán explorar retos reales publicados por empresas y organizaciones, con la posibilidad de trabajar y mostrar su talento en un entorno competitivo y profesional.

Por otro lado, las empresas, ingenieros, administradores, técnicos y médicos, quienes podrán publicar problemas o necesidades que enfrenta su sector. De esta manera podrán tener acceso a un ecosistema de propuestas frescas e innovadoras de estudiantes y docentes, con lo que podrán evaluar un financiamiento o contratar soluciones concretas.

Y ante esta problemática nos surge la siguiente pregunta: ¿Cómo podríamos hacer que las personas universitarias puedan tener un ámbito laboral y estable, sin importar en el sector en el que estén?

- Domain: Personas con una visión laboral en base a la Universidad en la que se encuentran.
- Customer Segments: Estudiantes y profesores de una Universidad, que quieren dar el salto del estudio al centro laboral.
- Pain Points: Falta de personal joven que quieran tener un futuro en el ámbito laboral, falta de atención en el mundo laboral y bajos recursos para contratar profesionales de buen desempeño individual.
- Gap: No existe una solución en el mercado que se dedique al conocimiento y a la conexión entre personas universitarias y el ámbito laboral. Así como también, las empresas requieren de personal joven para mantener una estabilidad económica aceptable.
- Vision/Strategy: Crear una plataforma accesible cuyos estudiantes y profesores de una Universidad, puedan darse a conocer mediante sus trabajos, tesis, asignaciones y que tengan oportunidad en el mundo laboral, obteniendo múltiples propuestas.
- Initial Segment: Estudiantes y profesores universitarios que buscan dar el paso entre la Universidad y el centro laboral, también, contar con empresas de peso, para la contratación de nuevo personal.

1.2.2.2. Lean UX Assumptions

Business Assumptions

1. **Creo en que nuestros usuarios**, ya sean estudiantes o profesores universitarios, puedan dar el gran paso desde la Universidad a un centro laboral.
2. **Estas necesidades se pueden satisfacer** una plataforma web que conecte a estudiantes y profesores lanzando sus materiales importantes a personas que quieran sobresalir.
3. **Nuestros clientes iniciales serán** estudiantes y profesores universitarios, de carreras al azar, que buscan un trabajo en cualquier empresa.
4. **El valor más importante que un cliente quiere de nuestros servicios es** poder conseguir trabajos y publicar sus logros y materiales anteriores para poder tener un historial llamativo.
5. **El cliente también va a obtener** flexibilidad de horarios, ingresos adicionales, desarrollo de habilidades blandas y validación en su perfil.
6. **Se va a obtener a la gran mayoría de clientes mediante** las redes sociales (Facebook, X, Instagram, TikTok), alianzas con empresas peruanas e internacionales y campañas digitales en varios foros universitarios.
7. **Se van a obtener ingresos mediante** comisiones por publicación de materiales u obtener una suscripción premium para empresas de mayor jerarquía.
8. **Nuestra competencia en el mercado serán** plataformas como Indeed, LinkedIn Jobs, Glassdoor, etc.

9. **Vamos a tener ventaja frente a nuestra competencia debido a** que estamos centrados en el mercado nacional e internacional, sin barreras como un título o años de experiencia.
10. **El mayor riesgo del servicio es** la baja participación de estudiantes y profesores o la desconfianza de las empresas en contratar a personal sin tener preparación seria.
11. **Lo resolveremos realizando** incentivos por primeras contrataciones, sistema de reputación de candidatos, alianzas con universidades y empresas y casos de éxito visible.
12. **Otro riesgo que debemos considerar es que** los estudiantes y los profesores no tengan el tiempo, ni la motivación para postularse para otros centros laborales durante su carrera.

User Assumptions

1. ¿Quién es el usuario?

Estudiantes y profesores universitarios, con habilidades en los centros laborales, que apuntan a un futuro mejor y además, de contar con un trabajo en una empresa mostrando sus trabajos, tesis y demás.

2. ¿Dónde encaja nuestro producto en su vida?

Durante la última etapa universitaria. Permitiendo aplicar conocimientos en proyectos reales, ganar reputación y generar ingresos.

3. ¿Qué problemas resuelve nuestro producto?

InnoBridge se centra en resolver problemas como: Validación de trabajos y tesis, obtener oportunidades laborales a nivel nacional e internacional, poca preocupación en los nuevos empleados a las empresas, falta de contratación de plantel por bajos recursos.

4. ¿Cuándo y cómo se usa nuestro producto?

En el tiempo libre de los estudiantes y profesores, acceden a la aplicación, crean su perfil, suben sus trabajos, tesis y más, buscan oportunidades mediante esos archivos y se visualizan las empresas de quienes están interesados por contar con dicha persona.

5. ¿Qué características son importantes?

- Interfaz sencilla y fácil de aprender.
- Filtro de búsqueda de habilidades.
- Sistema de reputación.
- Mensajes internos.
- Perfil con insignias.

6. ¿Cómo debería lucir y comportarse el producto?

Debe ser moderno, amigable, accesible, con diseño responsive, colores neutros y una experiencia clara e intuitiva que no distraiga ni complique al usuario.

Feature Assumptions

- **Creemos que** la aplicación debe contar una interfaz de usuario clara, responsive y fácil de navegar que permitirá a estudiantes y empleadores a utilizar la plataforma sin necesidad de capacitación previa ni conocimientos técnicos avanzados.
- **Creemos que** la plataforma debe proporcionar notificaciones automáticas y personalizables que mantendrán informados a los usuarios sobre contrataciones recibidas, nuevas oportunidades, mensajes o estados de contratación.
- **Creemos que** la aplicación debe contar con un sistema de reputación y validación de desempeño, que permitirá a los clientes calificar los trabajos, las tesis tanto de otros estudiantes y profesores, para así, construir un historial profesional verificable.
- **Creemos que** la aplicación debe ofrecer una sección de perfil personal con un historial digital, donde los estudiantes puedan mostrar trabajos y tesis previas, habilidades destacadas y contar con una base preparada dentro y fuera de la plataforma.
- Creemos que la aplicación debe permitir la publicación rápida y sencilla de proyectos por parte de las empresas o clientes, incluyendo la posibilidad de establecer plazos, presupuestos y requerimientos específicos.

1.2.2.3. Lean UX Hypothesis Statements

- **Hypothesis Statement 01:**

Creemos que los estudiantes universitarios estarán dispuestos a adoptar la plataforma InnoBridge para compartir sus trabajos, tesis y proyectos, con el fin de obtener experiencia laboral y aumentar su visibilidad profesional.

Sabremos que hemos tenido éxito:

Cuando al menos un 70% de los estudiantes registrados completen su perfil y suban al menos un proyecto o tesis dentro del primer mes de uso.

- **Hypothesis Statement 02:**

Creemos que InnoBridge facilitará la conexión entre pequeñas y medianas empresas con talento universitario al ofrecer un entorno accesible, flexible y confiable, sin las barreras de experiencia previa exigidas por otras plataformas.

Sabremos que hemos tenido éxito:

Cuando al menos un 60% de los proyectos publicados sean completados exitosamente y reciban una calificación positiva por parte de los estudiantes, validando así la calidad del talento.

- **Hypothesis Statement 03:**

Creemos que la incorporación de un sistema de reputación, portafolio y mensajería interna permitirá a los empleadores seleccionar candidatos más efectivamente, sin necesidad de pasar por procesos de contratación tradicionales.

Sabremos que hemos tenido éxito:

Cuando el 75% de los empleadores indiquen que encontraron al candidato adecuado usando únicamente las funcionalidades de la plataforma (perfil, reputación, portafolio y mensajería).

- **Hypothesis Statement 04:**

Creemos que ofrecer notificaciones automáticas personalizables a los usuarios sobre nuevas oportunidades, contratación o mensajes de empresas incrementará la tasa de engagement y retención de la plataforma.

Sabremos que hemos tenido éxito:

Cuando el 80% de los usuarios registrados respondan a al menos un mensaje o postulación dentro del primer mes de uso.

- **Hypothesis Statement 05:**

Creemos que los incentivos iniciales, como el acceso a proyectos destacados y descuentos en funciones premium, motivarán a los estudiantes y empresas a adoptar la plataforma rápidamente.

Sabremos que hemos tenido éxito:

Cuando al menos el 50% de los estudiantes nuevos completen su perfil y postulen a proyectos dentro de los primeros 15 días después de registrarse, y al menos un 30% de las empresas paguen por suscripciones premium para acceder a perfiles destacados.

1.2.2.4. Lean UX Canvas



Lean UX Canvas - Elaboración propia

1.3. Segmentos Objetivo

Capítulo II: Requirements Elicitation & Analysis

2.1. Competidores

2.1.1. Análisis competitivo

Competitive Analysis Landscape					
¿Por qué llevar a cabo este análisis?	Analizar a la competencia permite entender el mercado en el que se introducirá nuestro producto, ofreciendo una visión clara de las funcionalidades que ofrecen y cómo satisfacen las necesidades de sus clientes.				
(Nombre y logo de cada competidor)	InnoSpace 	LinkedIn 	AngelList (Wellfound) 	Behance 	
Perfil	Overview	Plataforma móvil donde estudiantes publican ideas innovadoras y startups para atraer inversión, participar en proyectos empresariales y conectar con empresas.	Red social profesional global que conecta profesionales y empresas para networking, empleos y contenido.	Plataforma que conecta startups con inversionistas y talento especializado, muy centrada en ecosistema emprendedor.	Plataforma para creativos y diseñadores donde muestran proyectos y buscan oportunidades de colaboración.
	Ventaja competitiva ¿Qué valor ofrece a los clientes?	Conecta estudiantes con empresas de forma bidireccional (ideas <-> proyectos), fomenta innovación y aprendizaje colaborativo.	Gran base de usuarios, networking masivo y posicionamiento de marca fuerte.	Acceso directo a startups, talento e inversores, muy nichado en el ecosistema.	Visibilidad global para proyectos creativos y portafolios.
	Mercado objetivo	Estudiantes universitarios, jóvenes emprendedores, empresas que buscan innovación y talento emergente.	Profesionales de todos los sectores y empresas de cualquier tamaño.	Startups, inversionistas y profesionales del sector tecnológico y emprendedor.	Diseñadores, artistas, creativos y empresas del sector visual.

Competitive Analysis Landscape					
Perfil de Marketing	Estrategias de marketing	Marketing digital en universidades, incubadoras y redes sociales. Enfoque en innovación y talento joven.	Publicidad en la misma plataforma, marketing de contenido y presencia en medios profesionales.	Eventos de startups, aceleradoras, networking en ecosistemas emprendedores.	Colaboraciones con escuelas de diseño, concursos y portafolios online.
	Mercado objetivo	Estudiantes, startups en etapa temprana, empresas innovadoras.	Profesionales, empresas medianas y grandes.	Startups tecnológicas, inversores, emprendedores.	Creativos, diseñadores freelance, agencias.
Perfil de Producto	Productos & Servicios	Publicación de ideas y startups, proyectos de empresas, exploración de estudiantes con habilidades alineadas.	Ofertas de empleo, networking, cursos, grupos profesionales.	Ofertas de empleo en startups, conexión con inversores y talento.	Portafolios creativos, exposición de proyectos, búsqueda de empleo creativo.
	Precios & Costos	Modelo freemium (básico gratis, premium para empresas o estudiantes destacados).	Gratis con funciones premium de pago (LinkedIn Premium, LinkedIn Ads).	Gratis con servicios premium para empleadores e inversores.	Gratis con opciones premium para mayor visibilidad.
	Canales de distribución (Web y/o Móvil)	App móvil (iOS/Android) y versión web.	App móvil y web.	App móvil y web.	App móvil y web.
Análisis SWOT	Fortalezas	Innovación en el enfoque educación-empresa, espacio exclusivo para estudiantes.	Base de usuarios masiva, marca reconocida.	Nicho claro en startups e inversión.	Alta visibilidad para creativos y artistas.

Competitive Analysis Landscape					
	Debilidades	Startup nueva, sin base de usuarios inicial fuerte.	Demasiada competencia interna, saturación de usuarios.	Menor popularidad que LinkedIn, limitado fuera del sector tech.	No cubre sectores fuera del diseño/creatividad.
	Oportunidades	Aliarse con universidades, incubadoras y empresas innovadoras.	Expansión a nuevos servicios de educación y formación.	Crecimiento del ecosistema emprendedor global.	Mayor demanda de portafolios digitales y talento creativo.
	Amenazas	Competidores grandes con más recursos, barreras de entrada altas.	Pérdida de interés por parte de usuarios jóvenes.	Dependencia del ecosistema startup e inversión.	Alta competencia en redes creativas (Dribbble, ArtStation).

2.1.2. Estrategias y tácticas frente a competidores

Afrontando las fortalezas de nuestros competidores:

Fortalezas de competidores:

- LinkedIn: base de usuarios masiva y marca consolidada.
- AngelList: nicho en startups e inversión.
- Behance: visibilidad global para creativos.
- InnoSpace: innovación en el vínculo educación–empresa.

Comprendemos que nuestras fortalezas son:

- Enfoque exclusivo en estudiantes y talento emergente.
- Integración bidireccional: empresas publican proyectos, estudiantes aportan ideas.
- Nicho menos saturado, con alto potencial de crecimiento en universidades.

Estrategias:

- Diferenciarnos mediante el **posicionamiento educativo–empresarial**, no solo profesional.
- Crear comunidad cerrada y validada de estudiantes para garantizar **calidad y confianza**.
- Alianzas estratégicas con universidades e incubadoras.

Tácticas:

- Implementar programas piloto con universidades y ferias de empleo.
- Incentivar a empresas innovadoras a publicar proyectos reales.
- Marketing digital en redes sociales universitarias y canales académicos.

Afrontando las debilidades de nuestros competidores:

Debilidades de competidores:

- LinkedIn: saturación, competencia interna.
- AngelList: limitado fuera del ecosistema startup.
- Behance: no cubre sectores fuera de lo creativo.
- InnoSpace: aún sin masa crítica de usuarios.

Comprendemos que nuestras debilidades son:

- Startup nueva con poca base de usuarios.
- Recursos limitados frente a grandes plataformas.

Estrategias:

- Crecer a través de **nichos específicos** antes de expandir.
- Apoyarnos en **alianzas institucionales** para atraer usuarios sin grandes inversiones.

Tácticas:

- Campañas de captación en universidades.
 - Programas de referidos entre estudiantes y empresas.
 - Construir una primera comunidad sólida en un sector (ej. ingeniería, innovación tecnológica).
-

Afrontando las oportunidades de nuestros competidores:

Oportunidades de competidores:

- Expansión de la educación digital (LinkedIn).
- Crecimiento del ecosistema emprendedor (AngelList).
- Demanda de portafolios digitales (Behance).
- Integración universidad–empresa (InnoSpace).

Comprendemos que nuestras oportunidades son:

- Conectar talento joven con innovación empresarial.
- Posicionarnos como la **plataforma de early talent**.
- Aprovechar el auge de incubadoras, hackathons y proyectos de innovación abierta.

Estrategias:

- Crear un espacio que combine **empleabilidad temprana + innovación colaborativa**.
- Aliarnos con incubadoras y hubs de emprendimiento universitario.

Tácticas:

- Organizar retos de innovación con empresas patrocinadoras.
 - Promocionar historias de éxito de estudiantes para dar visibilidad a la plataforma.
 - Ofrecer planes gratuitos iniciales a startups emergentes para ganar tracción.
-

Afrontando las amenazas de nuestros competidores:

Amenazas de competidores:

- Recursos y posicionamiento de grandes plataformas (LinkedIn, Behance).
- Alta competencia en redes profesionales y creativas.
- Barreras de entrada y dependencia de ecosistemas externos.

Comprendemos que nuestras amenazas son:

- Riesgo de que grandes plataformas incorporen funcionalidades similares.
- Dificultad de escalar sin financiamiento fuerte.

Estrategias:

- Enfocarnos en ser **más ágiles y especializados** que los grandes competidores.
- Desarrollar una propuesta de valor difícil de replicar: **ecosistema académico–empresarial cerrado**.

Tácticas:

- Innovar en funcionalidades (ej. "match" entre ideas de estudiantes y retos empresariales).
- Fidelizar a la comunidad con beneficios exclusivos (certificados, visibilidad, mentorías).
- Monitorear tendencias para adaptar rápidamente la plataforma.

2.2. Entrevistas

2.2.1. Diseño de entrevistas

Preguntas Estudiantes universitarios innovadores

Preguntas Directas:

1. ¿Podrías proporcionarnos tu nombre completo, tu edad y qué carrera estás estudiando actualmente?
2. ¿Has tenido la oportunidad de desarrollar alguna idea o proyecto innovador durante tus estudios? ¿Cómo fue esa experiencia?
3. ¿Qué dificultades enfrentas actualmente para dar a conocer tus ideas o proyectos a empresas o posibles inversionistas?
4. ¿Has participado en concursos, hackathons o ferias de innovación? ¿Qué aprendiste de esas experiencias?
5. ¿Qué tan fácil o difícil ha sido para ti encontrar espacios donde mostrar tu talento y tus habilidades?
6. ¿Qué te motiva más: que te reconozcan por tus ideas, que te apoyen con mentoría o que te den financiamiento para llevarlas a cabo?
7. ¿Qué habilidades técnicas o personales tuyas consideras que serían más valiosas para una empresa que busque talento joven?
8. ¿Qué plataformas, redes sociales o medios digitales utilizas hoy para difundir tus proyectos o buscar oportunidades?
9. ¿Qué temores o dudas tendrías al publicar tus ideas en una aplicación que conecta estudiantes con empresas?

10. Si existiera una app donde pudieras subir tus ideas, explorar proyectos de empresas y conectar directamente con gerentes, ¿la usarías? ¿Por qué sí o por qué no?

Preguntas Complementarias:

11. ¿Qué tan importante es para ti que una plataforma valide o certifique tu participación en proyectos reales?
12. ¿Qué te gustaría aprender o reforzar al colaborar con empresas en proyectos de innovación?

*Preguntas Gerentes/profesionales de innovación y talento***Preguntas Directas:**

1. ¿Podrías darnos tu nombre completo, edad y cargo actual en la empresa?
2. Cuéntame un poco sobre tu empresa: ¿en qué sector operan y cómo gestionan actualmente los proyectos de innovación?
3. ¿Qué tan frecuente es que busquen nuevas ideas o startups jóvenes para resolver retos internos?
4. ¿Qué dificultades encuentras al identificar y contactar estudiantes con habilidades alineadas a tus proyectos?
5. ¿Han intentado colaborar con universidades o estudiantes en el pasado? ¿Qué resultados tuvieron?
6. ¿Qué características valoras más en un estudiante o joven innovador al momento de incluirlo en un proyecto?
7. ¿Qué riesgos o preocupaciones tendrías al usar una aplicación que conecta directamente con talento emergente?
8. ¿Qué tan dispuesto estarías a publicar proyectos o retos de tu empresa en una plataforma abierta a estudiantes?
9. ¿Qué funcionalidades considerarías indispensables en una app como InnoSpace para que realmente sea útil para tu empresa?
10. Si existiera una app local que te brinde acceso directo a ideas innovadoras y perfiles de estudiantes con portafolio visible, ¿la usarías? ¿Por qué sí o por qué no?

Preguntas Complementarias:

11. ¿Qué tan importante sería para tu empresa contar con métricas de impacto o informes sobre la colaboración con estudiantes?
12. ¿Qué expectativas tendrías de la relación con los estudiantes a través de una plataforma como esta (ideas, prototipos, contratación futura, etc.)?

2.2.2. Registro de entrevistas**2.3.5. Ubiquitous Language**

Término (Inglés)	Término (Español)	Definición
Student	Estudiante	Usuario que busca oportunidades para postular, compartir ideas y participar en proyectos innovadores.
Manager	Gerente	Usuario que representa a la empresa, encargado de publicar proyectos y seleccionar talento joven para colaborar.
Idea Submission	Publicación de ideas	Acción realizada por el estudiante para compartir propuestas o startups innovadoras en la plataforma.
Project Posting	Publicación de proyecto	Acción realizada por el gerente para dar a conocer convocatorias de proyectos disponibles.
Application	Postulación	Proceso por el cual un estudiante se inscribe a un proyecto disponible o a una convocatoria de empresa.
Profile	Perfil profesional	Información personal y académica que un estudiante o gerente edita para mostrar sus capacidades y experiencia.
Application Status	Estado de postulación	Información sobre si la postulación de un estudiante está en revisión, aprobada o rechazada.
Requirement Registration	Registro de requerimientos	Proceso en el que el gerente detalla las necesidades específicas del proyecto.
Candidate Validation	Validación de postulantes	Evaluación realizada por el gerente para determinar la pertinencia de los estudiantes postulados.

2.3. Needfinding

En esta sección, se crearán las User Personas basándonos en la información obtenida de las entrevistas con nuestros usuarios potenciales. Consideraremos aspectos fundamentales como la edad, la ocupación, las preocupaciones y las metas de los entrevistados para entender mejor sus necesidades, comportamientos y expectativas. Además, se examinarán las características y preferencias de los usuarios a través del análisis de nuestros competidores, lo que nos ayudará a identificar áreas de oportunidad para mejorar.

2.3.1. User Personas

- **User Persona Estudiante Universitario**

PERSONA: Jeremy Vega

NAME	MARKET SIZE	TYPE		
Jeremy Vega	 70 %	Rational		
	Background			
	<p>Jeremy es un joven estudiante de Ingeniería de Software en la Universidad Peruana de Ciencias Aplicadas (UPC). Está desarrollando sus habilidades en programación, bases de datos y administración de tecnologías en la nube, con la intención de contribuir al avance tecnológico tanto en su país como a nivel internacional.</p>			
Demographic	Goals			
<input checked="" type="radio"/> Male 22 years  Lima, Perú Student	<ul style="list-style-type: none"> perfeccionar sus habilidades en programación, especialmente en lenguajes como Java y Python. adquirir un conocimiento profundo sobre la gestión de bases de datos, optimizando consultas SQL. familiarizarse con las principales plataformas de la nube como AWS, Azure o Google Cloud. 			
Skills	Quote			
Programación en Java  Gestión de Base de Datos No SQL  Programación en Python  Manejo de tecnologías de la Nube 	“ <i>El conocimiento es el camino que conecta la solución a los problemas del futuro.</i> “			
	Motivations	Frustrations		
	<ul style="list-style-type: none"> Superarse Profesionalmente Conectar con personas del mundo tecnológico Contribución a la industria 	<ul style="list-style-type: none"> Falta de claridad Escasez de recursos Presión por destacar 		
	Technology	Channels		
	     	 LinkedIn	 WhatsApp	 Instagram

UXPRESSIA
 This persona was built in uxpressia.com

User Persona Estudiante Universitario - Elaboración propia

- **User Persona Gerente de empresa**

PERSONA: Carlos Fernández

NAME	MARKET SIZE	TYPE	
Carlos Fernández	 30 %	Rational	
	Background		
	<p>Carlos es un ingeniero con más de 15 años de experiencia en el sector tecnológico. Dirige una empresa que está ampliando su operación y, a menudo, necesita contar con nuevos talentos para desarrollar productos innovadores. Aunque tiene un equipo interno comprometido, en momentos de alta demanda, busca apoyo externo, pero ha tenido malas experiencias con freelancers que no cumplen con los plazos o estándares de calidad.</p>		
Demographic	Goals		
<input checked="" type="radio"/> Male 42 years  Lima, Perú Manager	<ul style="list-style-type: none"> encontrar ideas innovadoras de estudiantes. no perder proyectos por falta de talento o capacidad. mejorar la gestión de talento externo y freelancers. 		
Skills	Quote		
Gestión de equipos multidisciplinarios  Estrategia de negocio  Liderazgo en proyectos tecnológicos  Gestión de talento externo 	<p>“Busco ideas frescas y soluciones innovadoras, pero sobre todo, necesito que el talento cumpla con lo que promete. La fiabilidad es clave.”</p>		
	Motivations	Frustrations	
	<ul style="list-style-type: none"> Buscar innovación constante. Optimizar recursos humanos. Asegurar cumplimiento de plazos y calidad. 	<ul style="list-style-type: none"> Freelancers poco comprometidos y con retrasos. Dificultad para encontrar talento adecuado. Preocupación por el compromiso de los estudiantes. 	
Technology	Channels		
	 	 	 LinkedIn
		 Email	 Phone

UXPRESSIA

This persona was built in uxpressia.com

User Persona Gerente de empresa - Elaboración propia

2.3.2. User Task Matrix

En este apartado se presenta la User Task Matrix, que refleja las tareas de los dos segmentos principales: estudiantes universitarios que buscan financiar sus proyectos innovadores y participar en proyectos de empresas, y gerentes que buscan contratar talento joven para el desarrollo de sus proyectos.

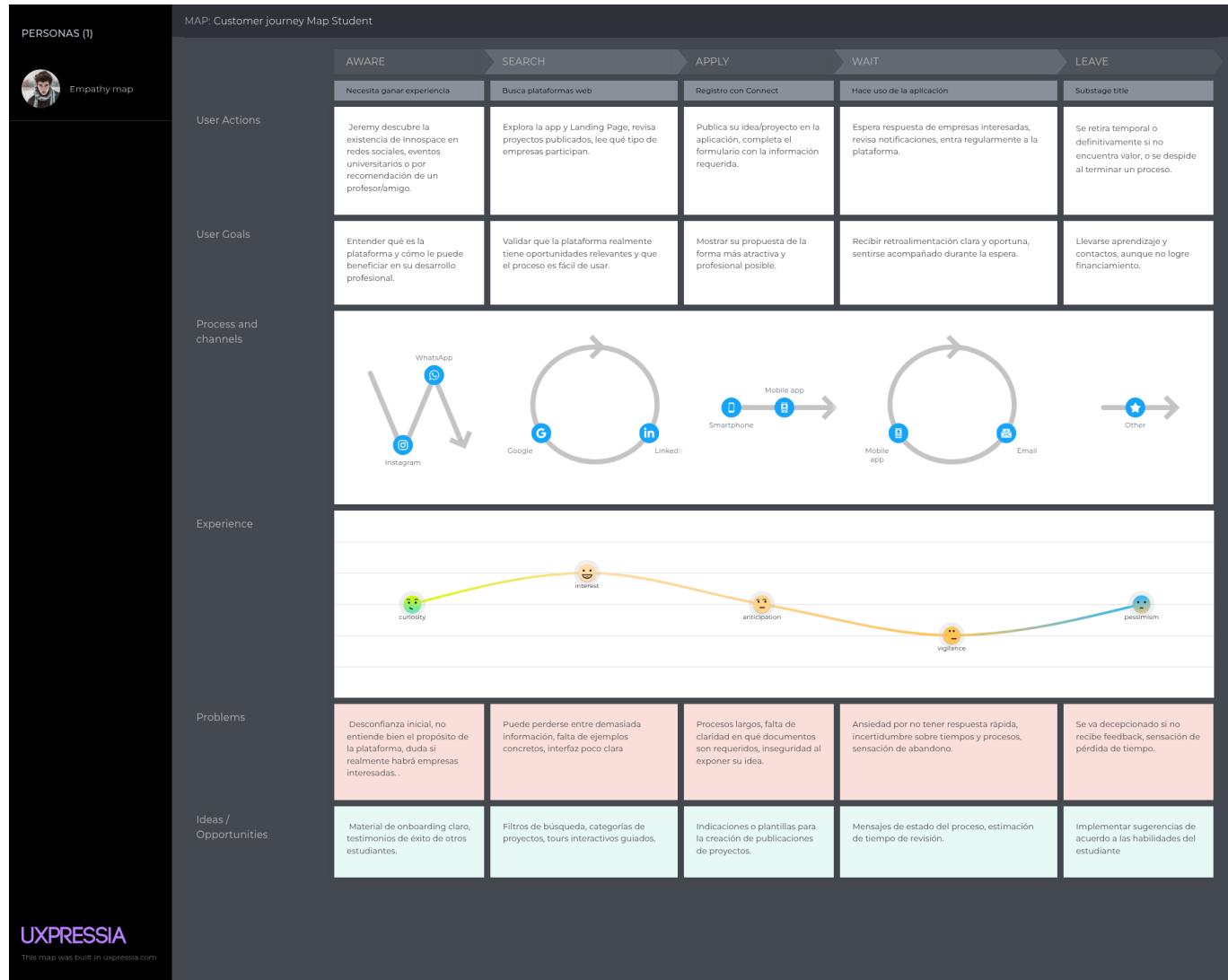
	Estudiante		Gerente	
TASK	FREQUENCY (Estudiante)	IMPORTANCE (Estudiante)	FREQUENCY (Gerente)	IMPORTANCE (Gerente)
Publicar ideas o startups innovadoras	Sometimes	High	Never	Low
Buscar proyectos disponibles	Often	High	Sometimes	Low
Postular a ofertas	Sometimes	High	Never	Low
Editar o actualizar su perfil profesional	Rarely	High	Sometimes	Medium
Consultar el estado de sus postulaciones	Sometimes	High	Never	Low
Asistir a entrevistas	Rarely	High	Sometimes	High
Asistir al daily scrum	Always	High	Rarely	Medium
Registrar requerimientos de proyecto	Never	Low	Sometimes	Medium
Publicar convocatorias de proyecto	Never	Low	Sometimes	Medium
Chatear con miembros del equipo	Sometimes	Medium	Sometimes	Medium
Ver postulaciones recibidas	Never	Low	Sometimes	Medium
Validar postulantes	Never	Low	Rarely	High
Aprobar entregables	Never	Low	Often	High
Planificación del sprint	Often	High	Often	High

2.3.3. User Journey Mapping

Esta sección muestra un Journey Map por cada segmento objetivo, detallando el proceso en el que descubren e interactúan con la aplicación.

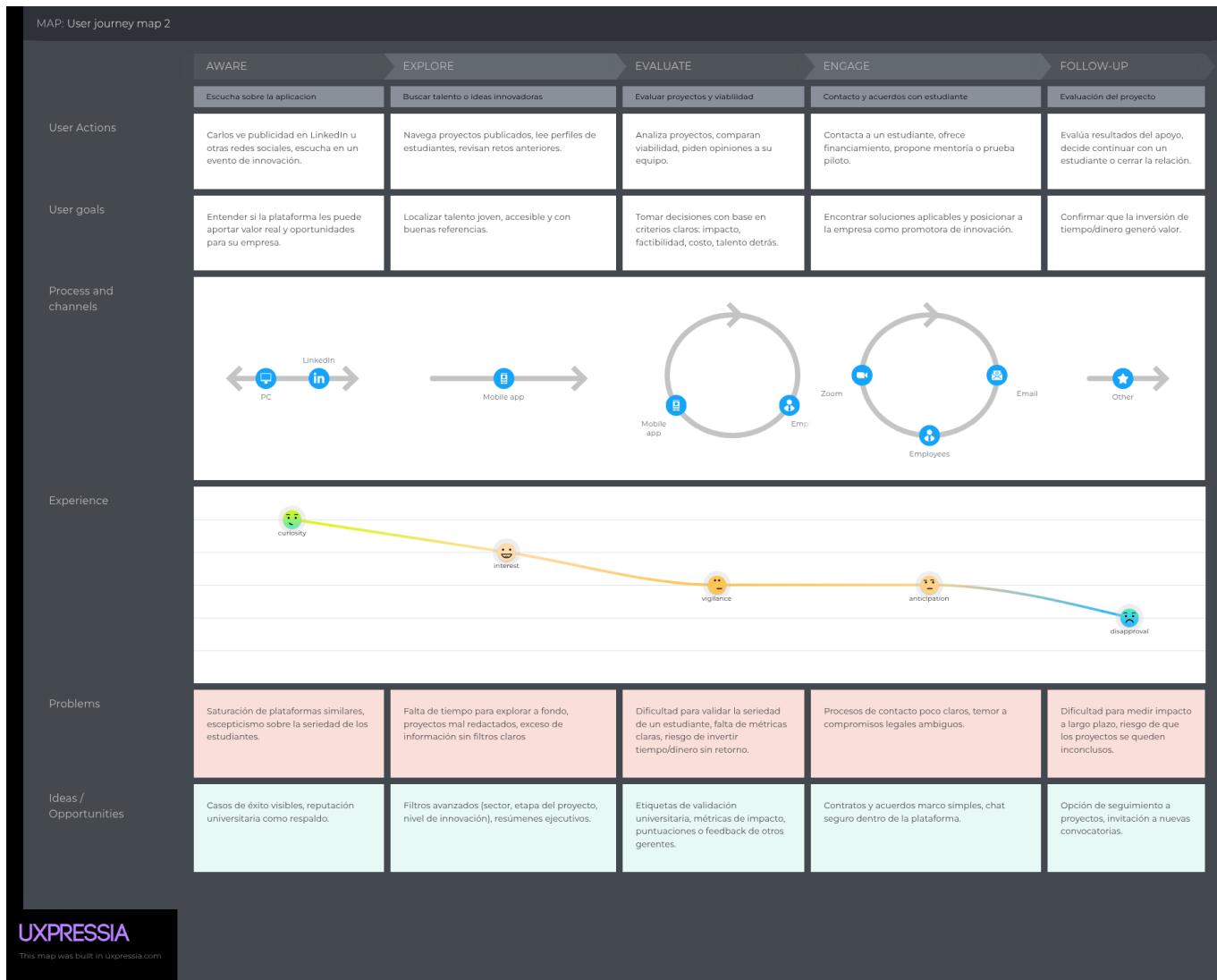
- User Journey Map Jeremy Vega:

Este Journey map detalla la experiencia de Jeremy Vega, un universitario de 22 años que descubre la aplicación móvil InnoSpace y la utiliza para presentar sus ideas.



- User Journey Map Carlos Fernandez:

Este Journey map detalla la experiencia de Carlos Fernández, un ingeniero dirigente de una empresa del sector tecnológico. Carlos escucha sobre la aplicación y decide probarla, explora perfiles de estudiantes e ideas de proyectos innovadoras.



2.3.4. Empathy Mapping

Empathy Map Segmento 1:

PERSONA: Empathy map

1.WHO are we empathizing with? <ul style="list-style-type: none"> Gerentes de innovación, operaciones o recursos humanos de medianas y grandes empresas. Personas con experiencia, poco tiempo y presión por resultados. Interesados en encontrar talento joven y soluciones innovadoras para problemas reales de la organización. 	7.What do they THINK and FEEL? <p>“<i>Si encontramos talento joven ahora, podemos formarlo según nuestras necesidades.</i>”</p> <p>“<i>La innovación es clave para mantenernos competitivos.</i>”</p>	2.What do they need to DO? <ul style="list-style-type: none"> Explorar proyectos fuera de sus canales tradicionales de innovación. Identificar talento emergente, captar ideas frescas, resolver problemas específicos de su sector. Seleccionar qué proyectos financiar, con qué estudiantes colaborar y cuánto invertir.
6.What do they HEAR? <p>Presión por ser competitivos y “no quedarse atrás”.</p> <p>Dudas sobre la viabilidad de trabajar con estudiantes.</p>		
5.What do they DO? <p>Buscan innovación vía consultoras, convocan concursos internos de ideas, contratan startups.</p> <p>Prefieren propuestas filtradas y resumidas, delegan a su equipo la primera revisión.</p>	PAINS <ul style="list-style-type: none"> Falta de tiempo para revisar propuestas extensas. Dificultad para evaluar el compromiso real de los estudiantes. Riesgo de invertir en ideas sin aplicación práctica. 	GAINS <ul style="list-style-type: none"> Acceso a un banco de ideas frescas y diversas. Descubrimiento de talento joven con visión innovadora. Posicionamiento de la empresa como referente en innovación y responsabilidad social.
		4.What do they SAY? <p>“What have we heard them say? “<i>Necesitamos soluciones frescas y rápidas.</i>”</p> <p>What can we imagine them saying? “<i>Quiero proyectos que tengan impacto y no solo ideas en papel.</i>”</p>

UXPRESSIA

This persona was built in uxpressia.com

Empathy Map Segmento 2:

PERSONA: Empathy map

1.WHO are we empathizing with? <ul style="list-style-type: none"> Estudiantes universitarios, de pregrado o recién egresados. Buscan visibilidad para sus proyectos, oportunidades de financiamiento o vinculación con empresas. Generalmente tienen entusiasmo, pero poca experiencia en cómo presentar o vender sus ideas. 	7.What do they THINK and FEEL? <p>“Quiero demostrar que puedo aportar soluciones reales.”</p> 	2.What do they need to DO? <ul style="list-style-type: none"> Aprender a presentar proyectos de forma profesional, clara y estructurada. Conseguir financiamiento, validación o contactos con empresas. Elegir qué idea compartir, cómo presentarla y a qué empresas apuntar.
6.What do they HEAR? <ul style="list-style-type: none"> Comentarios de profesores y mentores sobre “salir al mercado”. Opiniones de compañeros que dudan de que las empresas tomen en serio a estudiantes. Noticias sobre innovación, startups y convocatorias de financiamiento. 		3.What do they SEE? <ul style="list-style-type: none"> Competencia entre estudiantes, plataformas de innovación abierta, incubadoras. Profesores que recomiendan aplicar, compañeros que comparten experiencias en hackathons.
5.What do they DO? <ul style="list-style-type: none"> Aprender a presentar proyectos de forma profesional, clara y estructurada. Publican ideas en redes sociales, buscan becas o concursos. Usando la plataforma como portafolio de proyectos, aplicando varias veces hasta lograr un match con empresas. 		4.What do they SAY? <p>“No sé si mi proyecto es lo suficientemente bueno.” “Con esta oportunidad podría financiar mi idea.”</p>
PAINS <ul style="list-style-type: none"> Miedo a ser descartados sin explicación. Procesos largos o poco claros generan frustración. 	GAINS <ul style="list-style-type: none"> Retroalimentación directa de empresas y expertos. Posibilidad de financiamiento, prácticas o alianzas. Mayor visibilidad de su talento y proyectos. 	

UXPRESSIA
This persona was built in uxpressia.com

2.3.5. Ubiquitous Language

2.4. Requirements specification

2.4.1. User Stories

Epic /	Título	Descripción	Criterios de Aceptación	Relación con Epic

Story ID				
EP01	Registro y Onboarding de Usuarios	Como nuevo usuario de la plataforma, quiero registrarme y completar un perfil profesional guiado, para poder acceder a oportunidades o gestionar talento según mi rol.	No corresponde	No corresponde
EP02	Gestión de Oportunidades y Proyectos	Como gerente de una empresa, quiero publicar proyectos con requisitos personalizados, para atraer a estudiantes con el perfil adecuado y resolver tareas específicas de mi organización.	No corresponde	No corresponde
EP03	Exploración y Postulación a Proyectos	Como estudiante universitario, quiero explorar y postularme a proyectos compatibles con mis habilidades e intereses, para ganar experiencia práctica y construir un portafolio sólido.	No corresponde	No corresponde
EP04	Evaluación y Retroalimentación	Como gerente, quiero evaluar el desempeño de los estudiantes durante y al finalizar el proyecto, para asegurar la calidad del trabajo y fomentar una cultura de mejora continua.	No corresponde	No corresponde
EP05	Creación y Visualización de Portafolio	Como estudiante, quiero subir proyectos y evidencias de mis habilidades en un portafolio digital, para mostrar mi experiencia a futuras	No corresponde	No corresponde

		oportunidades laborales.		
US01	Registro de usuario	<p>Como usuario nuevo, quiero crear una cuenta, para poder acceder a las funciones de aplicación.</p>	<p>Escenario 1: Estudiante crea una cuenta Dado que el estudiante está en la pantalla de registro, Y selecciona la opción "Registrarse como Estudiante", cuando el estudiante ingresa un nombre de usuario, un correo electrónico y una contraseña válidos, Y presiona el botón "Registrarse", entonces la cuenta debe crearse correctamente.</p> <p>Escenario 2: Gerente crea una cuenta Dado que el gerente está en la pantalla de registro, Y selecciona la opción "Registrarse como Gerente", cuando el gerente ingresa un nombre de usuario, un correo electrónico y una contraseña válidos, Y presiona el botón "Registrarse", entonces la cuenta debe crearse correctamente.</p>	EP01
US02	Inicio de sesión	<p>Como usuario registrado, quiero iniciar sesión, para acceder a mi cuenta y explorar los proyectos.</p>	<p>Escenario 1: Estudiante crea una cuenta Dado que el usuario está en la pantalla de inicio de sesión y tiene una cuenta válida, cuando ingresa su nombre de usuario y contraseña correctos, Y presiona el botón "Iniciar sesión", entonces debería acceder a su cuenta correctamente.</p>	EP01
US03	Cierre de Sesión	<p>Como usuario, quiero cerrar sesión de mi cuenta, para proteger mi información personal.</p>	<p>Escenario 1: Usuario cierra sesión Dado que el usuario está autenticado y en su panel principal, cuando presiona el botón de "Cerrar sesión", entonces el usuario debe ser redirigido a la pantalla de inicio de sesión y la sesión debe cerrarse correctamente.</p>	EP01
US04	Recuperar contraseña	<p>Como usuario, quiero recuperar mi contraseña, para</p>	<p>Escenario 1: Usuario envía solicitud de recuperación Dado que el usuario está en la pantalla de inicio de sesión, cuando</p>	EP01

		poder acceder a mi cuenta si la olvido.	haga clic en el enlace de "¿Olvidaste tu contraseña?", Y proporciona su correo electrónico registrado, entonces debe recibir un correo electrónico con un enlace para restablecer su contraseña.	
US05	Explorar proyectos de empresa	Como estudiante, quiero explorar proyectos de empresas, para encontrar oportunidades de colaboración.	Escenario 1: Estudiante explora proyectos de empresas Dado que el estudiante está en la plataforma y ha iniciado sesión, cuando selecciona la opción "Explorar proyectos de empresas", entonces el estudiante debe ver una lista de proyectos de empresas disponibles para unirse o colaborar.	EP03
US06	Gerente publica nuevo proyecto	Como gerente, quiero completar un formulario con los detalles de un proyecto, para publicarlo y que los estudiantes puedan verlo y postularse.	Escenario 1: Gerente crea un proyecto exitosamente Dado que el gerente ha iniciado sesión y está en la sección "Publicar Proyecto", cuando Cuando complete los campos (título, descripción, habilidades requeridas, categoría), Y haga clic en "Publicar", entonces el proyecto debe mostrarse.	EP02
US07	Estudiante publica nuevo idea innovadora	Como usuario, quiero publicar una descripción de mi idea o startup innovadora, para atraer la atención e inversión de empresas interesadas.	Escenario 1: Estudiante pública una idea Dado que el estudiante ha iniciado sesión y navega a "Publicar mi Idea", cuando ingrese un título, descripción detallada y el sector de la idea, Y haga clic en "Publicar Idea", entonces la idea debe ser listada.	EP04
US08	Editar Perfil de Usuario	Como usuario registrado, quiero editar la información de mi perfil en cualquier momento, para mantener mi perfil actualizado.	Escenario 1: Edición exitosa exitosa Dado que el usuario está en la página de su perfil, cuando hace clic en el botón "Editar Perfil", modifica los diferentes campos de su perfil y presiona "Guardar", entonces se actualiza la información y se muestran los cambios en su perfil.	EP01
US09	Explorar Ideas de estudiantes	Como gerente, quiero explorar una lista de ideas innovadoras publicadas por estudiantes, para encontrar	Escenario 1: Navegación por la lista de ideas Dado que el gerente ha iniciado sesión y está en el dashboard, cuando navega a la sección "Ideas Innovadoras", Y el gerente puede	EP05

		<p>oportunidades potenciales en las que mi empresa pueda invertir.</p>	<p>hacer clic en "Ver Detalles" para obtener más información, entonces el sistema muestra una lista de ideas, cada una con su título, descripción corta y el nombre del autor.</p>	
US10	Búsqueda filtrada de convocatorias	<p>Como estudiante universitario, quiero buscar proyectos según filtros (habilidades, área académica, nivel de experiencia), para identificar oportunidades que se ajusten a mi perfil.</p>	<p>Escenario 1: Búsqueda con resultados disponibles Dado que el estudiante ha iniciado sesión y se encuentra en la sección "Proyectos", cuando aplica filtros por habilidades y área académica, entonces el sistema muestra una lista de proyectos compatibles, ordenados por relevancia, y el estudiante puede marcar algunos como favoritos.</p> <p>Escenario 2: Búsqueda sin resultados Dado que el estudiante ha iniciado sesión y aplica filtros demasiado específicos, Y selecciona la opción "Registrarse como Gerente", cuando se ejecuta la búsqueda, entonces el sistema no muestra proyectos disponibles y presenta el mensaje "No se encontraron proyectos" con la sugerencia de ampliar los filtros.</p>	EP03
US11	Visualización de detalles del proyecto	<p>Como estudiante universitario, quiero ver la descripción completa de un proyecto publicado, para decidir si postular o no.</p>	<p>Escenario 1: Proyecto vigente con opción de postulación Dado que el estudiante selecciona un proyecto activo desde la lista de proyectos, cuando abre la página de detalle, entonces el sistema muestra los requisitos, duración, beneficios, empresa y fecha límite, y el botón de "Postular" está habilitado.</p> <p>Escenario 2: Proyecto caducado sin opción de postulación Dado que el estudiante selecciona un proyecto cuya fecha límite de postulación ya venció, cuando abre la página de detalle, entonces el sistema muestra toda la información del proyecto pero el botón de "Postular" aparece deshabilitado junto al mensaje "Cerrado".</p>	EP03

US12	Postulación en línea	<p>Como estudiante universitario, quiero postular a un proyecto cargando mi CV o perfil digital, para para mostrar mis habilidades y experiencia al reclutador.</p>	<p>Escenario 1: Postulación exitosa con perfil completo cada bloque</p> <p>Dado que el estudiante tiene su perfil completo en la plataforma, cuando postula en "Postular" y adjunta su CV, entonces el sistema registra la postulación y muestra el mensaje "Postulación enviada con éxito".</p> <p>Escenario 2: Postulación rechazada por perfil incompleto</p> <p>Dado que el estudiante no completó la información mínima de su perfil, cuando intenta postular a un proyecto, entonces el sistema muestra el mensaje "Completa tu perfil antes de postular" y no registra la solicitud.</p>	EP03
US13	Seguimiento de postulaciones	<p>Como estudiante universitario, quiero ver el estado de mis postulaciones (en revisión, aceptado, rechazado), para organizar mis próximos pasos.</p>	<p>Escenario 1: Panel con postulaciones activas</p> <p>Dado que el estudiante ya postuló a varios proyectos, cuando ingresa al panel de postulaciones, entonces el sistema muestra cada proyecto con su estado actualizado (en revisión, aceptado o rechazado).</p>	EP03
US14	Construcción de portafolio	<p>Como estudiante universitario, quiero que los proyectos en los que participé exitosamente se registren en mi portafolio, para demostrar mi experiencia a futuras empresas.</p>	<p>Escenario 1: Proyecto validado agregado al portafolio</p> <p>Dado que el estudiante ha finalizado un proyecto y la empresa lo marcó como "Completado", cuando crevisa su portafolio, entonces el sistema muestra automáticamente el proyecto con el rol desempeñado, duración y competencias aplicadas.</p> <p>Escenario 2: Proyecto pendiente de validación</p> <p>Dado que el estudiante finalizó un proyecto pero la empresa aún no confirma la finalización, cuando revisa su portafolio, entonces el sistema no muestra el proyecto y aparece la nota "Pendiente de validación".</p>	EP03
US15	Guardado de Convocatorias	<p>Como estudiante, quiero guardar</p>	<p>Escenario 1: Guardar proyecto como favorito</p>	EP03

	Favoritos	proyectos como favoritos, para revisarlos más tarde antes de postular.	Dado que un estudiante navega por proyectos, cuando haga clic en "Guardar en Favoritos", entonces entonces el proyecto debe añadirse a su lista personal. Escenario 2: Postular desde el listado de favoritos Dado que el estudiante revisa su lista de favoritos, cuando da click en la opción "favoritos", entonces debe ver todos los proyectos marcados con la opción de postular directamente desde ahí.	
US16	Filtrado de estudiantes	Como gerente, quiero filtrar estudiantes por habilidades y nivel académico, para para identificar candidatos adecuados más rápido.	Escenario 1: Filtrado exitoso por habilidades Dado que el gerente aplica filtros, cuando selecciona habilidades específicas, entonces aparece la lista de estudiantes que cumplen. Escenario 2: Manejo de filtros sin resultados Dado que el gerente aplica filtros, cuando cuando no hay coincidencias, entonces entonces el sistema muestra el mensaje "No se encontraron estudiantes con estos criterios".	EP03
US17	Registro inmediato desde la landing	Como usuario nuevo, quiero acceder rápidamente al formulario de registro, para comenzar a usar la plataforma sin perder tiempo.	Escenario 1: Acceso directo desde el encabezado Dado que soy un usuario que visita la landing por primera vez, cuando shago clic en el botón "Empieza" ubicado en el encabezado, entonces soy redirigido al formulario de registro correspondiente (estudiante o gerente).	EP01
US18	Llamados a la acción claros en cada sección	Como usuario, quiero encontrar botones de acción relevantes a lo largo de la landing, para tomar decisiones sin tener que regresar al inicio.	Escenario 1: CTA visibles al finalizar cada bloque Dado que estoy navegando por secciones como "¿Cómo funciona?" o "Casos de éxito", cuando llego al final de una sección, entonces encuentro un botón como "Regístrate ahora" o "Publica tu primer proyecto" de forma clara y destacada.	EP01

2.4.2. Impact Mapping

2.4.3. Product Backlog

# Orden	User Story Id	Título	Descripción	Story Points (1 / 2 / 3 / 5 / 8)
1	US18	Llamados a la acción claros	Como visitante de la landing page, quiero contar con un encabezado visible en todo momento, para moverme fácilmente entre las secciones sin hacer scroll manualmente.	3
2	US17	Registro inmediato desde la landing	Como usuario nuevo, quiero acceder rápidamente al formulario de registro, para comenzar a usar la plataforma sin perder tiempo.	3
3	US01	Registro de usuario	Como usuario nuevo, quiero crear una cuenta, para poder acceder a las funciones de la aplicación.	3

# Orden	User Story Id	Título	Descripción	Story Points (1 / 2 / 3 / 5 / 8)
4	US02	Inicio de sesión	Como usuario registrado, quiero iniciar sesión, para acceder a mi cuenta y explorar los proyectos.	2
5	US04	Recuperar contraseña	Como usuario registrado, quiero recuperar mi contraseña, para poder acceder a mi cuenta si la olvido.	1
6	US08	Editar Perfil de Usuario	Como usuario registrado, quiero editar la información de mi perfil en cualquier momento, para mantener mi perfil actualizado.	3
7	US03	Cierre de Sesión	Como usuario, quiero cerrar sesión de mi cuenta, para proteger mi información personal.	1

# Orden	User Story Id	Título	Descripción	Story Points (1 / 2 / 3 / 5 / 8)
8	US06	Gerente publica nuevo proyecto	Como gerente, quiero completar un formulario con los detalles de un proyecto, para publicarlo y que los estudiantes puedan verlo y postularse.	5
9	US10	Búsqueda filtrada de convocatorias	Como estudiante universitario, quiero buscar proyectos según filtros (habilidades, área académica, nivel de experiencia), para identificar oportunidades que se ajusten a mi perfil.	5
10	US05	Explorar proyectos de empresa	Como estudiante, quiero explorar proyectos de empresas, para encontrar oportunidades de colaboración.	3

# Orden	User Story Id	Título	Descripción	Story Points (1 / 2 / 3 / 5 / 8)
11	US11	Visualización de detalles del proyecto	Como estudiante universitario, quiero ver la descripción completa de un proyecto publicado, para decidir si postular o no.	3
12	US12	Postulación en línea	Como estudiante universitario, quiero postular a un proyecto cargando mi CV o perfil digital, para mostrar mis habilidades y experiencia al reclutador.	5
13	US13	Seguimiento de postulaciones	Como estudiante universitario, quiero ver el estado de mis postulaciones (en revisión, aceptado, rechazado), para organizar mis próximos pasos.	3

# Orden	User Story Id	Título	Descripción	Story Points (1 / 2 / 3 / 5 / 8)
14	US15	Guardado de Convocatorias Favoritos	Como estudiante, quiero modificar la información de mi perfil para mantener mis datos actualizados en la plataforma.	3
15	US16	Filtrado de estudiantes	Como gerente, quiero filtrar estudiantes por habilidades y nivel académico para identificar candidatos adecuados más rápido.	5
16	US07	Estudiante publica nueva idea innovadora	Como usuario, quiero publicar una descripción de mi idea o startup innovadora, para atraer la atención e inversión de empresas interesadas.	4

# Orden	User Story Id	Título	Descripción	Story Points (1 / 2 / 3 / 5 / 8)					
17	US09	Explorar Ideas de estudiantes	Como gerente, quiero explorar una lista de ideas innovadoras publicadas por estudiantes, para encontrar oportunidades potenciales en las que mi empresa pueda invertir.	3	18	US14	Construcción de portafolio	Como estudiante universitario, quiero que los proyectos en los que participé exitosamente se registren en mi portafolio, para edemostrar mi experiencia a futuras empresas.	8

2.5. Strategic-Level Domain-Driven Design

2.5.1. EventStorming

Como equipo realizamos una sesión de EventStorming con el objetivo de comprender el dominio del problema y plantear una primera aproximación al modelado general de Innospace. La actividad tuvo una duración de alrededor de 1-2 horas, donde identificamos los principales eventos, actores y reglas que definen las interacciones entre estudiantes y empresas.

Durante la sesión usamos una herramienta colaborativa para organizar y visualizar los elementos, lo que nos permitió discutir, consensuar y delimitar los primeros bounded contexts del sistema. El resultado es un mapa inicial del dominio que servirá como base para el análisis y diseño detallado en las siguientes etapas.

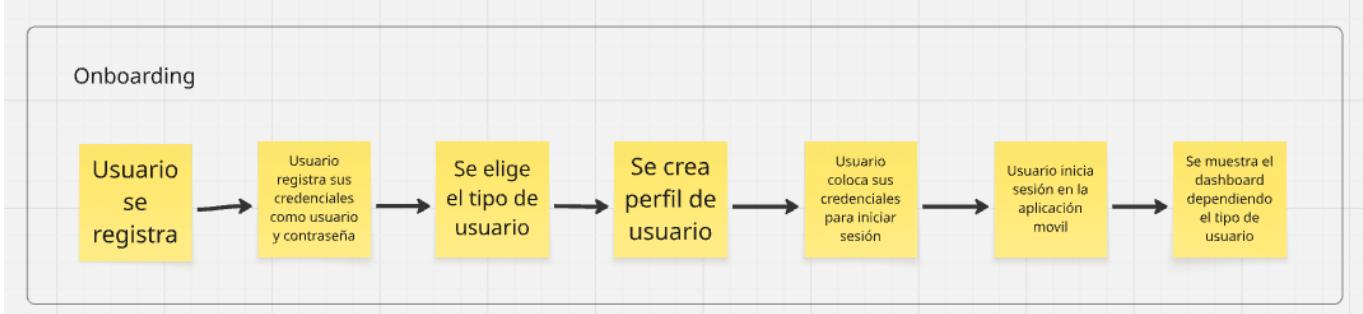
Se modifican los detalles de la convocatoria	Título y descripción fueron ingresados	Habilidades requeridas fueron establecidas	Se esperan postulantes para la convocatoria	Usuario coloca sus credenciales para iniciar sesión	Se publica la idea/proyecto
Gerente selecciona la opción "Crear convocatoria"	Usuario registra sus credenciales como usuario y contraseña	Se elige el tipo de usuario	Se crea perfil de usuario	Se recibe feedback acorde a la experiencia de trabajo	Empresa se contacta con el estudiante
Beneficios son establecidos	Se crea la convocatoria	Empresa propone apoyo financiero/mejoría.	Usuario se registra en la aplicación	Se muestra el dashboard dependiendo el tipo de usuario	Se comparte la información de contacto
Usuario inicia sesión en la aplicación móvil	Se explora el apartado de proyectos	Estudiante acepta la colaboración.	Estudiante selecciona la opción: "Crear Proyecto"	Se elimina la publicación del proyecto	Estudiante ingresa a la sección de explorar convocatorias
Activa las notificaciones de la convocatoria	Empresa selecciona un proyecto para colaborar.	Estudiante rechaza la oportunidad	Título, descripción y objetivos fueron ingresados	Se activan las notificaciones de la publicación	Postula a la convocatoria
Gerente borra la convocatoria	Se filtra la búsqueda proyectos	Acuerdo de colaboración se formaliza.	Se ingresa la categoría/tema en la que se basará el proyecto	Se modifica la información de la idea	Marca un proyecto como favorito
Revisa las habilidades y tecnologías requeridas	Se hace un seguimiento de la postulación	Es rechazado por no cumplir con requisitos	Es aceptado en la convocatoria	Su perfil se actualiza con la experiencia del trabajo	Accede a la información de contacto de la empresa

Event Storming - Elaboración propia

2.5.1.1. Candidate Context Discovery

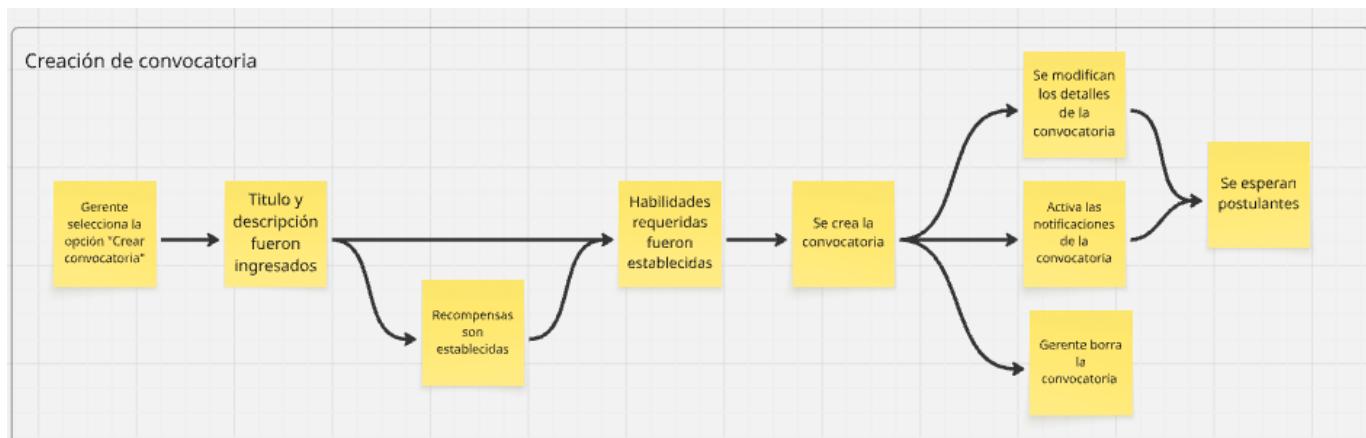
Tras concluir la sesión de Event Storming con la herramienta Miro, se llevó a cabo un análisis detallado de los eventos detectados con el fin de identificar los contextos candidatos más relevantes para el dominio. Este trabajo incluyó reconocer patrones y vínculos entre los eventos y creando flujos a partir de ellos. Como resultado, se organizaron conjuntos de eventos que correspondían a un mismo proceso dentro de la aplicación.

Onboarding



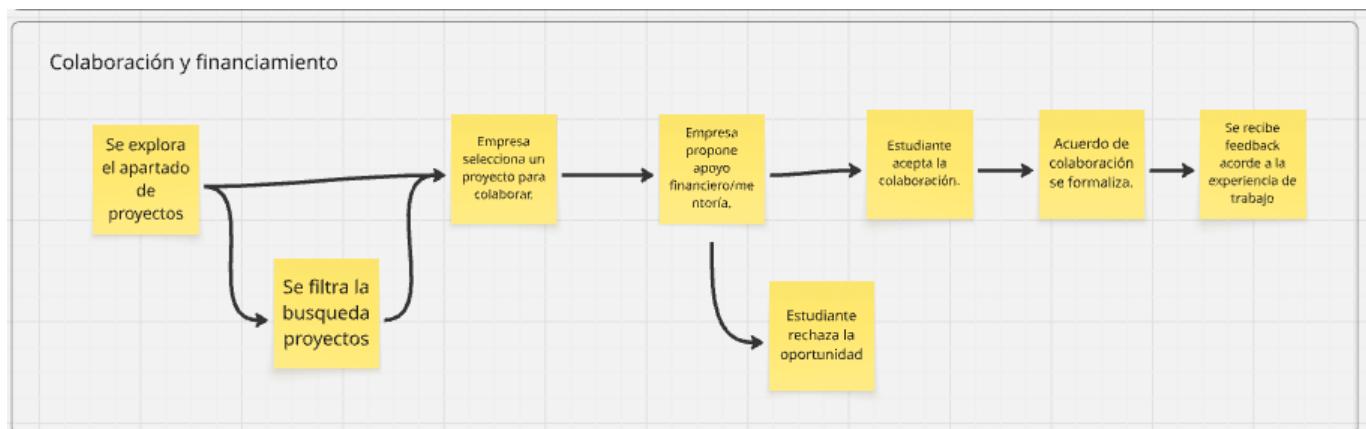
Elaboración propia

Creación de convocatoria



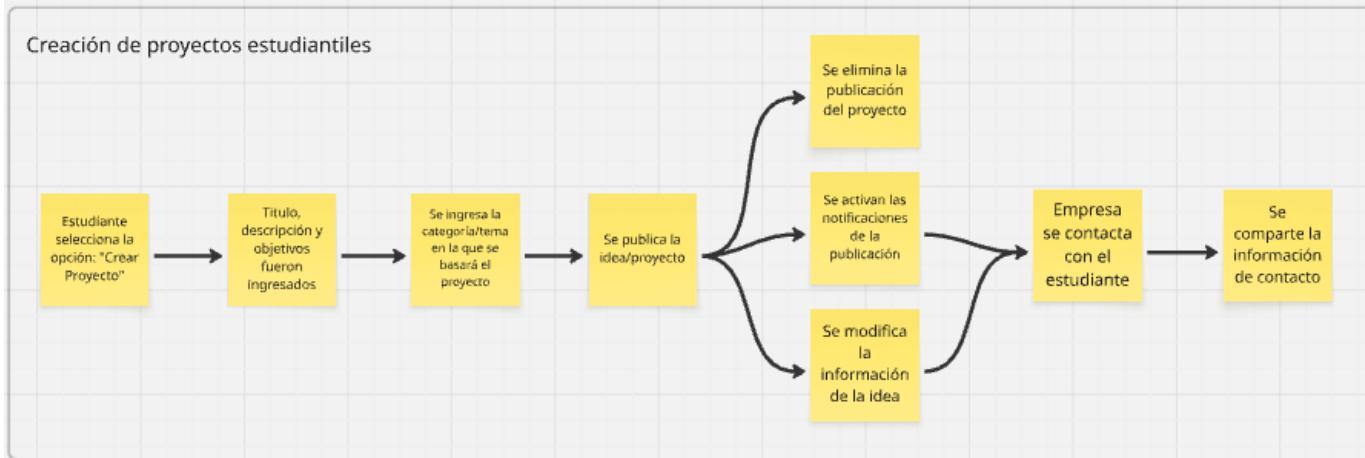
Elaboración propia

Colaboración y financiamiento



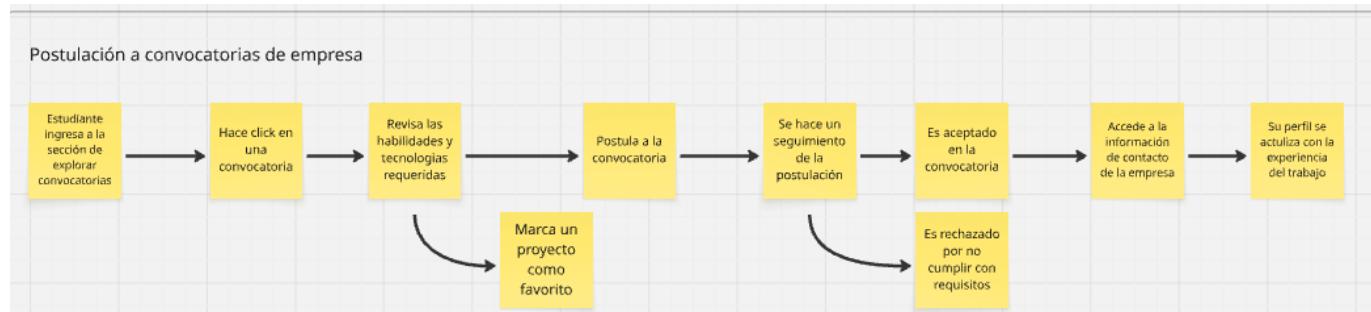
Elaboración propia

Creación de proyectos estudiantiles



Elaboración propia

Postulación a convocatorias de empresa



Elaboración propia

Después de mostrar los flujos de eventos que suceden dentro de la aplicación, los pivotal points (eventos que cambian el curso del sistema) son estos:

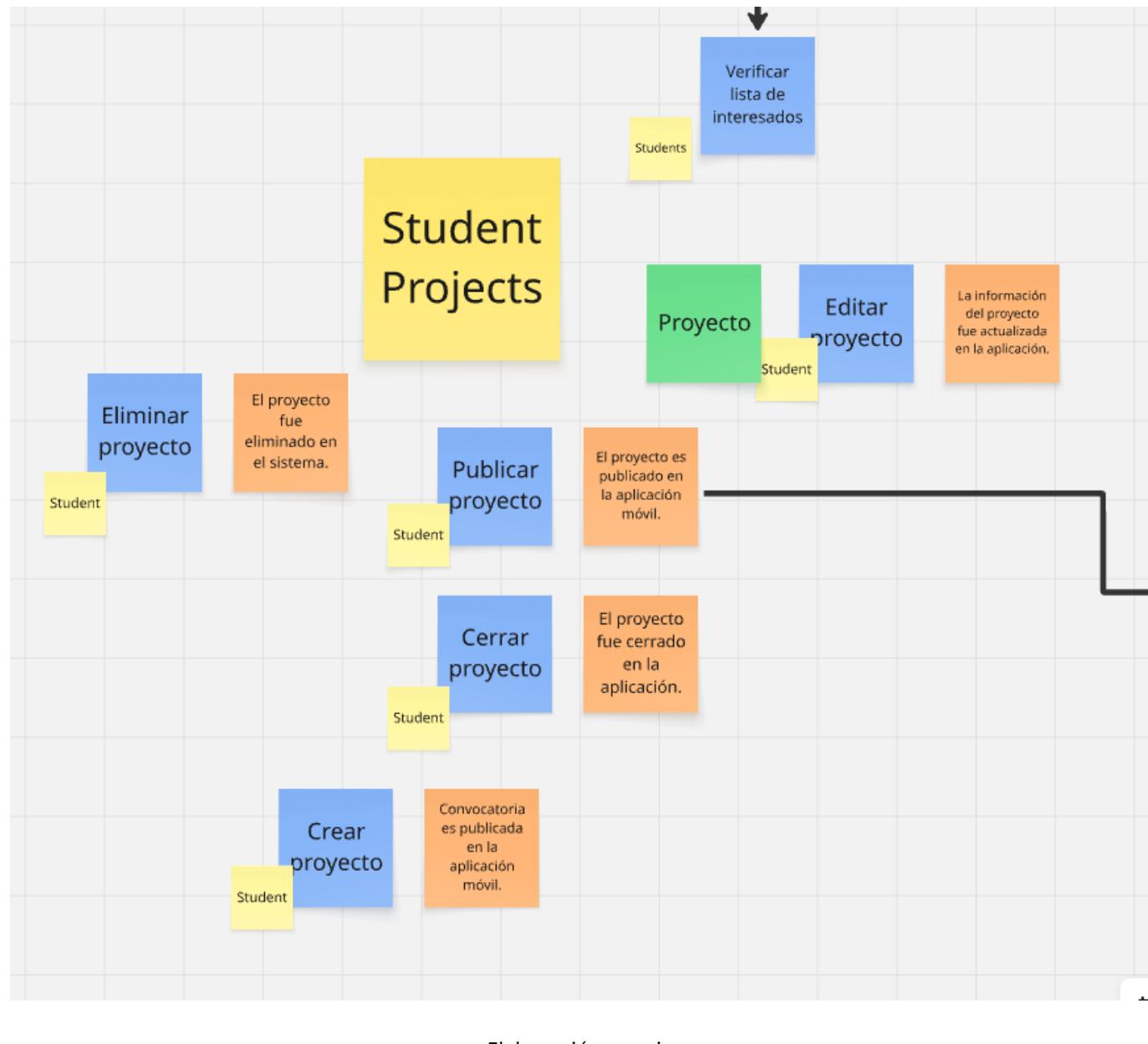
- La creación de un proyecto estudiantil
- La publicación de una oportunidad por parte de la empresa
- La postulación de un estudiante a una oportunidad
- La decisión de colaboración de una empresa sobre un proyecto
- La aceptación o rechazo de la colaboración por parte del estudiante

Al identificar estos pivotal points, se puede observar cómo los eventos se agrupan en contextos distintos, quedando definidos los siguientes bounded contexts:

- Student Projects: Gestión del ciclo de vida de proyectos estudiantiles.
- Company Opportunities: Creación y publicación de convocatorias de empresas.
- Student Applications: Administración de postulaciones de estudiantes a oportunidades.
- Project Collaboration: Decisiones de colaboración y respuestas de estudiantes.

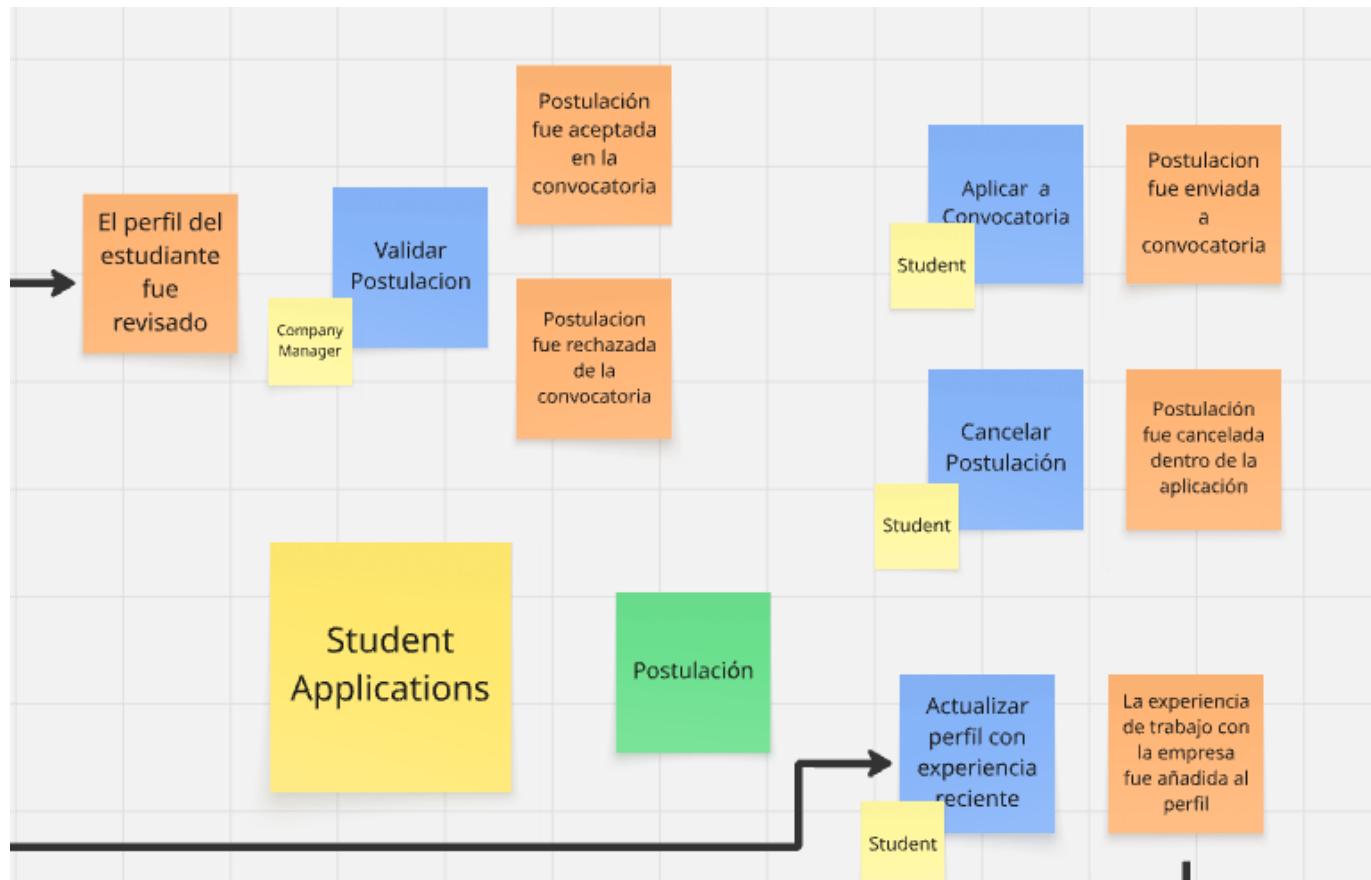
Luego de realizar estos pasos, se utilizó la herramienta Miro para dividir estos Bounded Contexts, detallando los agregados, eventos y entidades:

Student Projects:



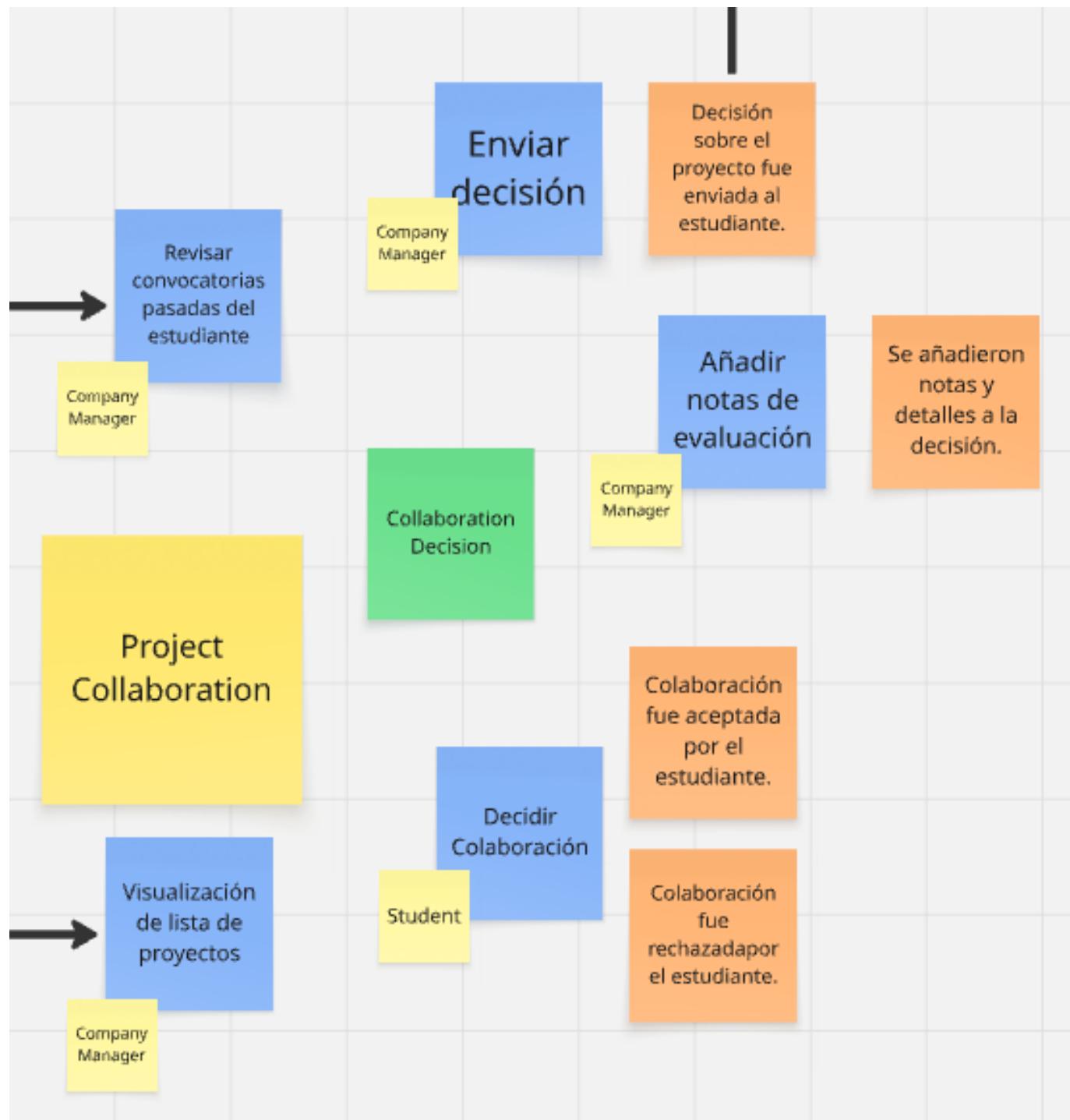
Elaboración propia

Student Applications



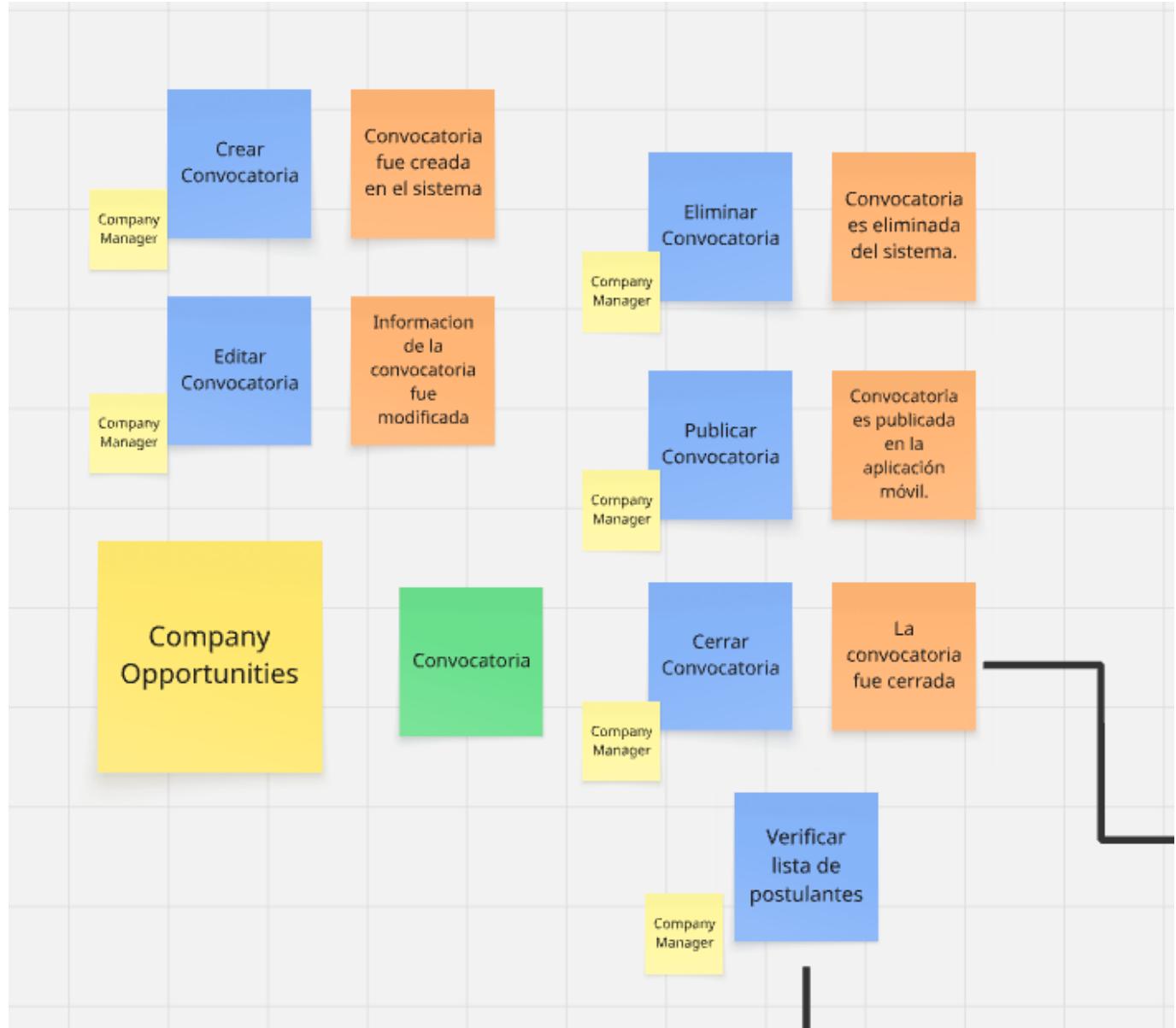
Elaboración propia

Project Collaboration:



Elaboración propia

Company Opportunities

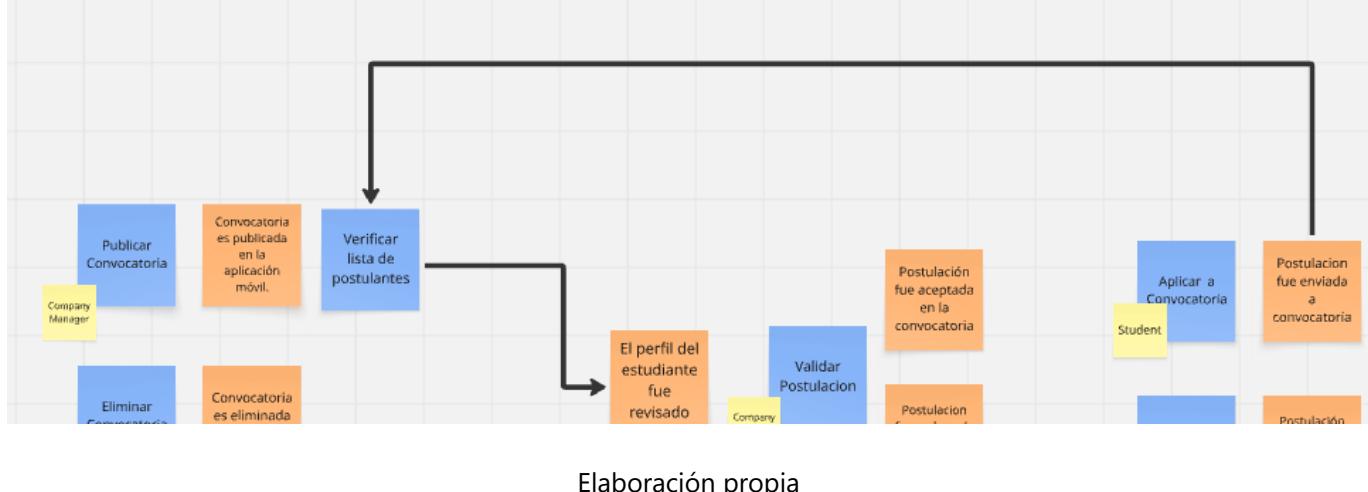


Elaboración propia

2.5.1.2. Domain Message Flows Modeling

Ahora se mostrarán los eventos en los cuales se comunican los bounded contexts hallados previamente.

Company Opportunities y Student Applications: Cuando un estudiante postula a una convocatoria, se mostrará en la lista de postulantes del gerente, luego el gerente podrá revisar el perfil del postulante y validar su postulación



Elaboración propia

Company Opportunities y Student Applications: Cuando una convocatoria acaba y el estudiante elegido participa en ella, entonces el perfil del estudiante se actualizará con la convocatoria en la que participó.



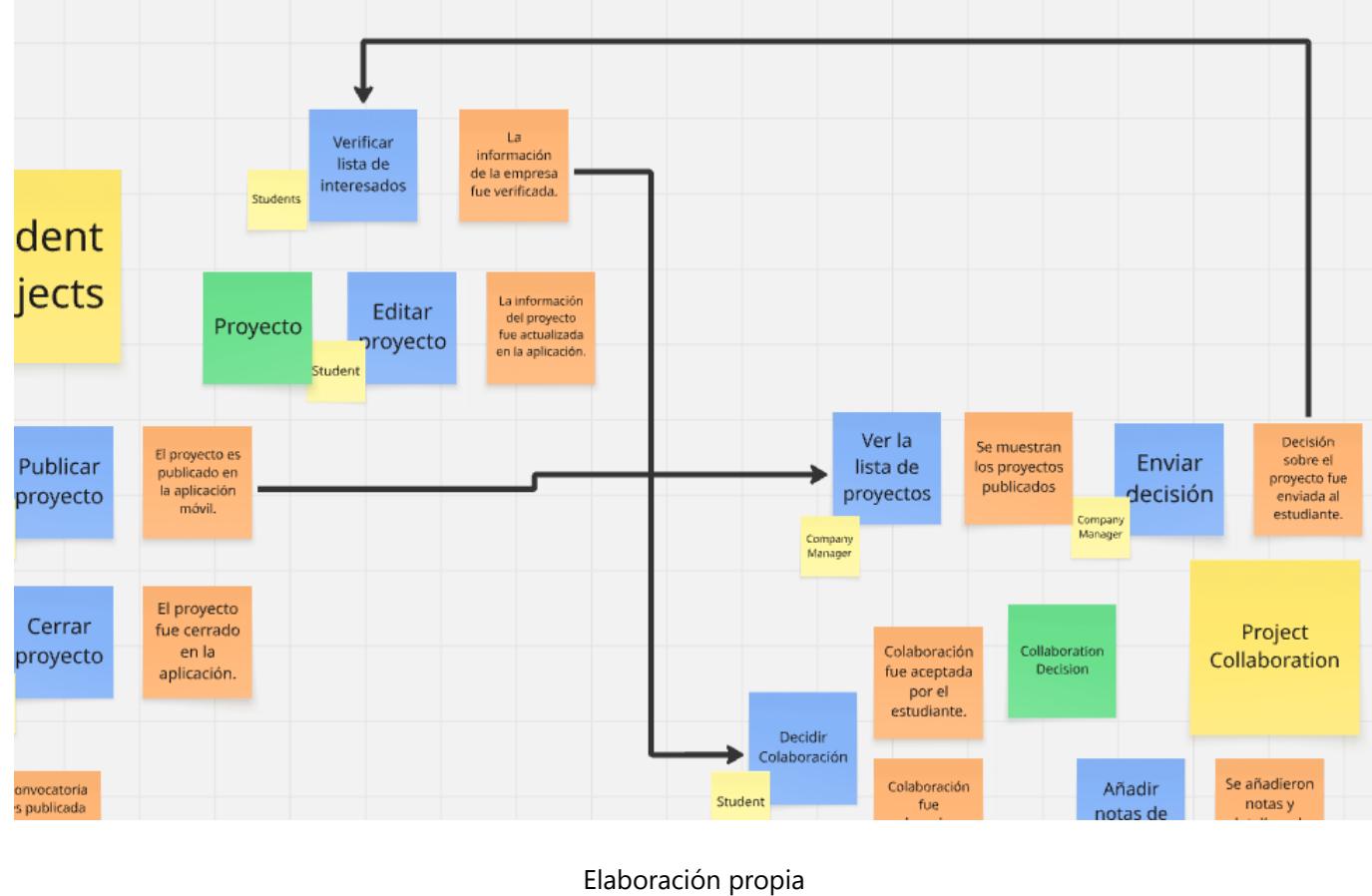
Elaboración propia

Project Collaboration y Student Applications: Cuando el estudiante gana experiencia después de una convocatoria, el gerente puede corroborar su experiencia antes de colaborar en un proyecto del estudiante.

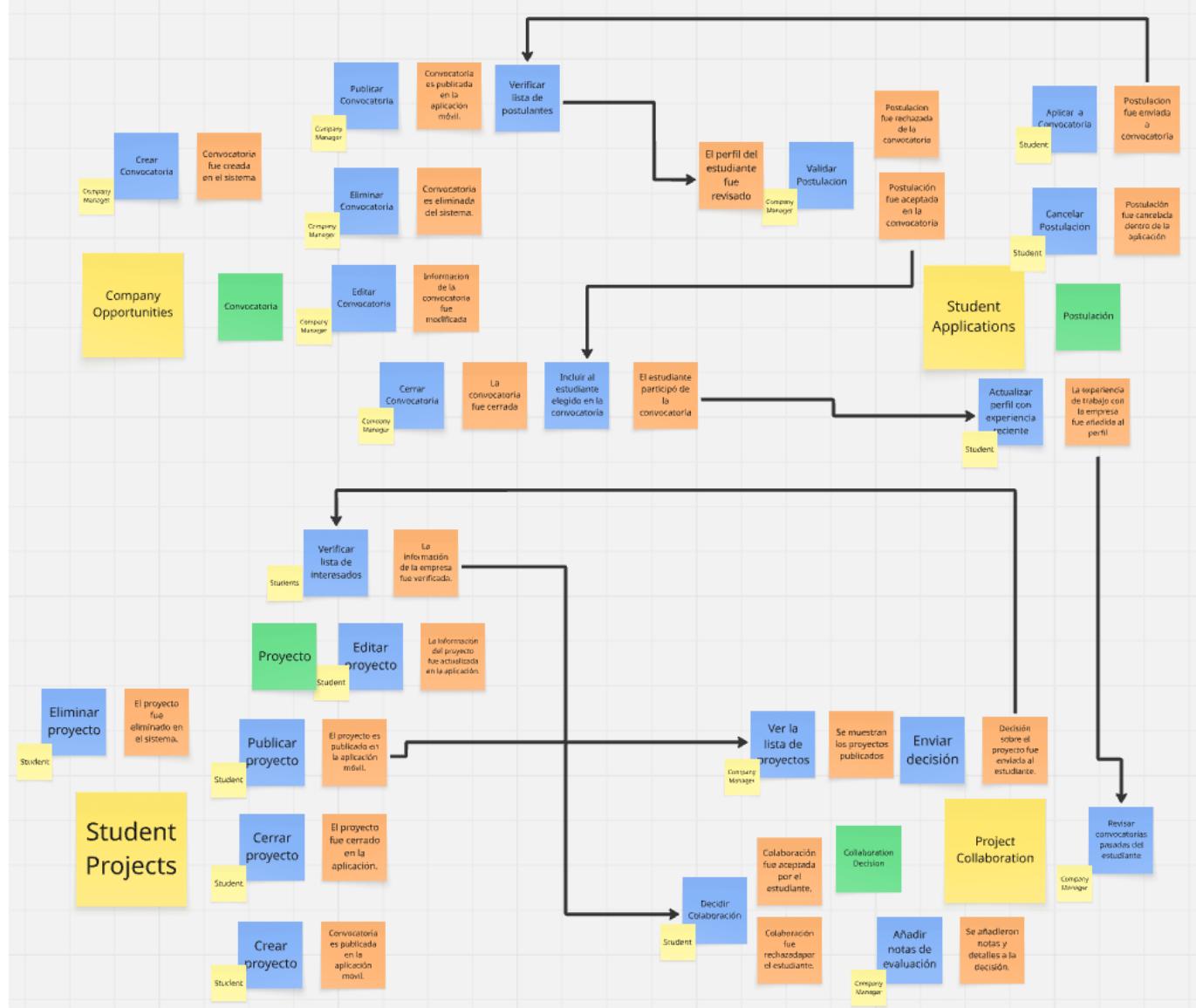


Elaboración propia

Project Collaboration y Student Projects: Cuando el estudiante publica un proyecto, un gerente podrá verlo en la aplicación y enviar la decisión al estudiante, finalmente el estudiante aceptará si decidir la colaboración o no.



Aca se muestra el diagrama completo que incluye los 4 Bounded Contexts:



Elaboración propia

Enlace al Miro Board: https://miro.com/app/board/uXjVJKRuXIw=/?share_link_id=238987388336

2.5.1.3. Bounded Context Canvases

En esta parte se lleva a cabo la descomposición estratégica del dominio a través de la construcción de Bounded Context Canvases, con el propósito de reconocer y delimitar áreas funcionales claras dentro del sistema. Esta actividad es clave en el enfoque de Domain-Driven Design (DDD), ya que ayuda a definir con precisión los límites semánticos y técnicos entre los distintos módulos, reduciendo la complejidad y permitiendo que cada componente pueda evolucionar de manera independiente.

Student Projects

Student Projects

Context Overview Definition		Capability Analysis		Business Rules Distillation & Ubiquitous Language Capture	
Propósito:	Límite:		-Crear proyectos con título y descripción. -Actualizar detalles del proyecto. -Publicar proyectos para visibilidad externa. -Finalizar proyectos cuando estén completos. -Eliminar proyectos obsoletos.	Reglas	<p>Un proyecto siempre pertenece a un estudiante propietario.</p> <p>Un proyecto puede estar en tres estados: <i>Draft</i>, <i>Published</i>, <i>Completed</i>.</p> <p>Solo el propietario puede modificar, publicar o finalizar un proyecto.</p> <p>No se puede publicar un proyecto sin título y descripción.</p>
Core	Support	Crear, publicar y finalizar proyectos.	Actualizar y eliminar proyectos.	Design Critique	<p>Lenguaje</p> <p>Project: Entidad principal que representa un trabajo creado por un estudiante.</p> <p>Draft: Estado inicial de un proyecto, aún no visible para otros usuarios.</p>
Dependencies Capture		Student Applications: las postulaciones se basan en proyectos publicados.	Company Opportunities: proyectos publicados pueden ser evaluados por empresas.	Debilidades	<p>Published: Estado en el que el proyecto se hace público para empresas y postulaciones.</p> <p>Completed: Estado final que indica que el proyecto ha sido finalizado.</p> <p>ProjectContent: Conjunto de información que describe al proyecto (título, descripción, archivos, links).</p>

Elaboración propia

Company Opportunities

Context Overview Definition		Capability Analysis		Business Rules Distillation & Ubiquitous Language Capture	
Propósito:	Límite:		-Registrar una nueva convocatoria en estado Draft. -Gestionar el cambio de estado de Draft a Published para hacerla visible a los estudiantes. -Finalizar la convocatoria y evitar nuevas aplicaciones. -Remover del sistema convocatorias en estado Draft o cerradas.	Reglas	<p>Solo una empresa registrada puede crear oportunidades.</p> <p>Una oportunidad debe tener título, descripción, requisitos y estado válido.</p> <p>Una oportunidad puede estar en estado <i>Draft</i>, <i>Published</i> o <i>Closed</i>.</p> <p>Solo la empresa propietaria puede modificar o cerrar sus oportunidades.</p> <p>Una vez <i>Closed</i>, la oportunidad no puede reabrirse.</p>
Core	Support	Creación y gestión de oportunidades (crear, publicar, cerrar).	Actualización de detalles, visibilidad de oportunidades.	Design Critique	<p>Lenguaje</p> <p>Opportunity: Publicación hecha por una empresa para atraer proyectos o estudiantes.</p> <p>CompanyOwner: Empresa creadora y responsable de la oportunidad.</p>
Dependencies Capture		Student Applications: Los estudiantes postulan a oportunidades creadas en este contexto.	Riesgo de duplicar reglas con StudentApplications si no se definen bien los límites.		<p>PublishedOpportunity: Oportunidad publicada y accesible a los estudiantes.</p> <p>ClosedOpportunity: Oportunidad finalizada y no disponible para nuevas postulaciones.</p>

Elaboración propia

Student Applications

Context Overview Definition		Capability Analysis	Business Rules Distillation & Ubiquitous Language Capture		
Propósito:	Límite:		Reglas		
Gestionar las postulaciones de los estudiantes a las oportunidades publicadas por las empresas, desde el registro de la postulación hasta su aceptación o rechazo.	Abarca únicamente el ciclo de vida de la postulación (crear, consultar, actualizar estado), sin involucrar la creación de oportunidades (Company Opportunities) ni la gestión de proyectos estudiantiles (Student Projects).	<ul style="list-style-type: none"> -Registrar una nueva aplicación de un estudiante a una oportunidad. -Obtener detalles de una aplicación específica. -Gestionar el cambio de estado de Pending a Accepted. -Gestionar el cambio de estado de Pending a Rejected. 	<p>Un estudiante solo puede postular a oportunidades en estado <i>Published</i>.</p> <p>Una postulación debe estar vinculada a un estudiante y una oportunidad válidas.</p> <p>Una postulación no puede modificarse una vez aceptada o rechazada.</p> <p>Solo la empresa propietaria de la oportunidad puede aceptar o rechazar postulaciones.</p>		
Core	Support	Design Critique	Lenguaje		
Capability Layering <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px; width: 50%;"> Core Crear postulación. Aceptar postulación. Rechazar postulación. Validar elegibilidad. </td> <td style="padding: 5px; width: 50%;"> Support Consultar postulación. Listar postulaciones de estudiante. Listar postulaciones de oportunidad. </td> </tr> </table>				Core Crear postulación. Aceptar postulación. Rechazar postulación. Validar elegibilidad.	Support Consultar postulación. Listar postulaciones de estudiante. Listar postulaciones de oportunidad.
Core Crear postulación. Aceptar postulación. Rechazar postulación. Validar elegibilidad.	Support Consultar postulación. Listar postulaciones de estudiante. Listar postulaciones de oportunidad.				
Dependencies Capture <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px; width: 50%;"> Company Opportunities: valida que la oportunidad esté publicada. </td> <td style="padding: 5px; width: 50%;"> Project Collaboration: se envían las postulaciones a este bounded context. </td> </tr> </table>				Company Opportunities: valida que la oportunidad esté publicada.	Project Collaboration: se envían las postulaciones a este bounded context.
Company Opportunities: valida que la oportunidad esté publicada.	Project Collaboration: se envían las postulaciones a este bounded context.				

Elaboración propia

Project Collaboration

Project Collaboration

Context Overview Definition		Capability Analysis	Business Rules Distillation & Ubiquitous Language Capture								
Propósito:	Límite:	<ul style="list-style-type: none"> -Registrar decisiones empresariales vinculadas a proyectos estudiantiles. -Gestionar la respuesta del estudiante frente a la decisión. 	Reglas <p>Una empresa puede marcar un proyecto como <i>Interested</i> o <i>NotInterested</i>. La decisión debe estar vinculada tanto a un proyecto como a la empresa que lo emite. El estudiante puede responder una decisión con <i>Accepted</i> o <i>Rejected</i>. Una vez que el estudiante acepta, la decisión no puede ser modificada.</p>								
Core	Support	Design Critique <table border="1"> <tr> <td>Fortalezas</td> <td>Riesgos</td> </tr> <tr> <td>Define claramente el flujo de negociación entre empresas y estudiantes con observaciones y estados.</td> <td>Possible solapamiento de funciones con Student Applications; requiere coordinación estricta para evitar inconsistencias.</td> </tr> <tr> <td>Debilidades</td> <td></td> </tr> <tr> <td>Alta dependencia de otros bounded contexts, lo que puede generar acoplamiento fuerte.</td> <td></td> </tr> </table>	Fortalezas	Riesgos	Define claramente el flujo de negociación entre empresas y estudiantes con observaciones y estados.	Possible solapamiento de funciones con Student Applications; requiere coordinación estricta para evitar inconsistencias.	Debilidades		Alta dependencia de otros bounded contexts, lo que puede generar acoplamiento fuerte.		Lenguaje <p>CollaborationDecision: Registro formal de la decisión tomada por una empresa sobre un proyecto estudiantil. CollaborationStatus (Interested, NotInterested): Estado que refleja si la empresa desea o no colaborar en el proyecto. StudentResponseStatus: Estado que representa la respuesta del estudiante, usados para dar contexto o justificar la elección.</p>
Fortalezas	Riesgos										
Define claramente el flujo de negociación entre empresas y estudiantes con observaciones y estados.	Possible solapamiento de funciones con Student Applications; requiere coordinación estricta para evitar inconsistencias.										
Debilidades											
Alta dependencia de otros bounded contexts, lo que puede generar acoplamiento fuerte.											
Dependencies Capture			<p>SendDecision: Acción de la empresa de enviar su decisión respecto a un proyecto. AcceptDecision: Acción del estudiante de aceptar la decisión emitida por la empresa.</p>								

Elaboración propia

2.5.2. Context Mapping

Esta sección documenta el proceso de análisis y diseño estructural de la aplicación, centrándose en la identificación y relación de los bounded contexts. Utilizando la metodología de Context Mapping de Domain-Driven Design, se evaluaron diversas alternativas para definir los límites de cada contexto y los patrones de interacción entre ellos.

El proceso involucró un análisis crítico de capacidades, considerando escenarios de descomposición, consolidación y reorganización de funcionalidades. A través de iteraciones y preguntas estratégicas, se establecieron las relaciones más apropiadas—Partnership, Customer-Supplier con Anti-corruption Layer y Shared Kernels—que garantizan autonomía, consistencia y escalabilidad del sistema.

A continuación se muestran los escenarios planteados:

Escenario 1: ¿Partir "Student Projects" en contextos más pequeños?

- Análisis: Evaluamos dividirlo en "Project Creation" y "Project Management".
- Decisión: No. El ciclo de vida del proyecto es cohesivo. La división añadiría acoplamiento innecesario y overhead de comunicación.

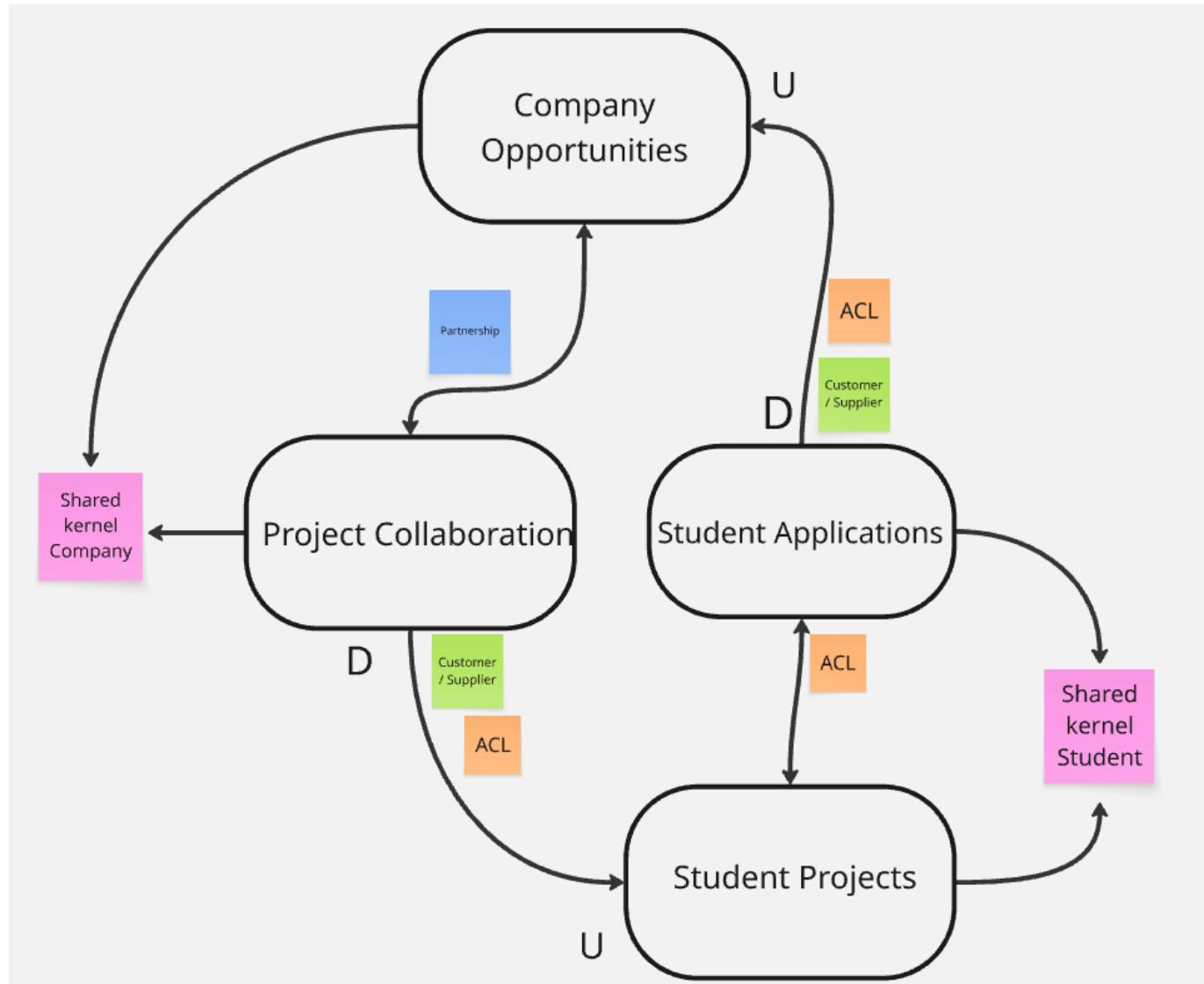
Escenario 2: ¿Duplicar el perfil del estudiante en "Student Applications"?

- Análisis: Consideramos duplicar datos para romper la dependencia con "Student Projects".

- Decisión: No. Se optó por un Anti-corruption Layer (ACL). La duplicación genera inconsistencia. El ACL mantiene la autonomía con una sola fuente de verdad.

Escenario 3: ¿Consolidar "Project Collaboration" y "Company Opportunities" en "Company Engagement"?

- Análisis: Evaluamos unirlos por ser ambos iniciativas de la empresa.
- Decisión: No. Sus capacidades centrales son distintas (convocatorias abiertas vs. contacto proactivo). Mantenerlos separados con una Partnership ofrece mayor cohesión.



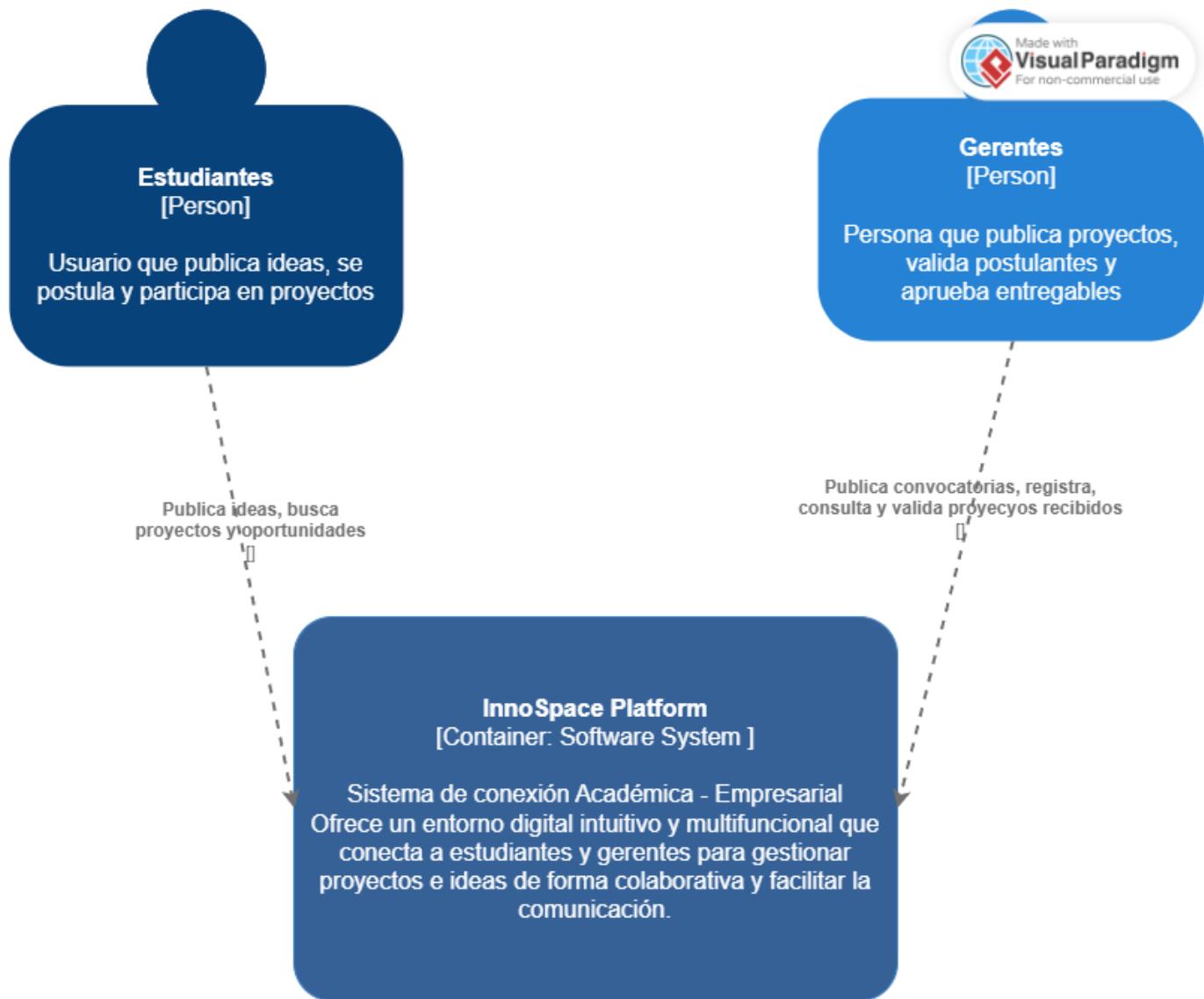
Elaboración propia

Destino (Downstream)	Origen (Upstream)	Tipo de Relación	Comentario
Project Collaboration	Student Projects	Customer/Supplier + ACL	Consumo de información de proyectos y estudiantes, traduciendo modelos mediante Anti-Corruption Layer
Student Applications	Student Projects	ACL	Utiliza datos de estudiantes y proyectos, protegiendo su modelo con ACL
Student Applications	Company Opportunities	Customer/Supplier + ACL	Consumo de convocatorias, adaptando el modelo upstream a sus necesidades con ACL

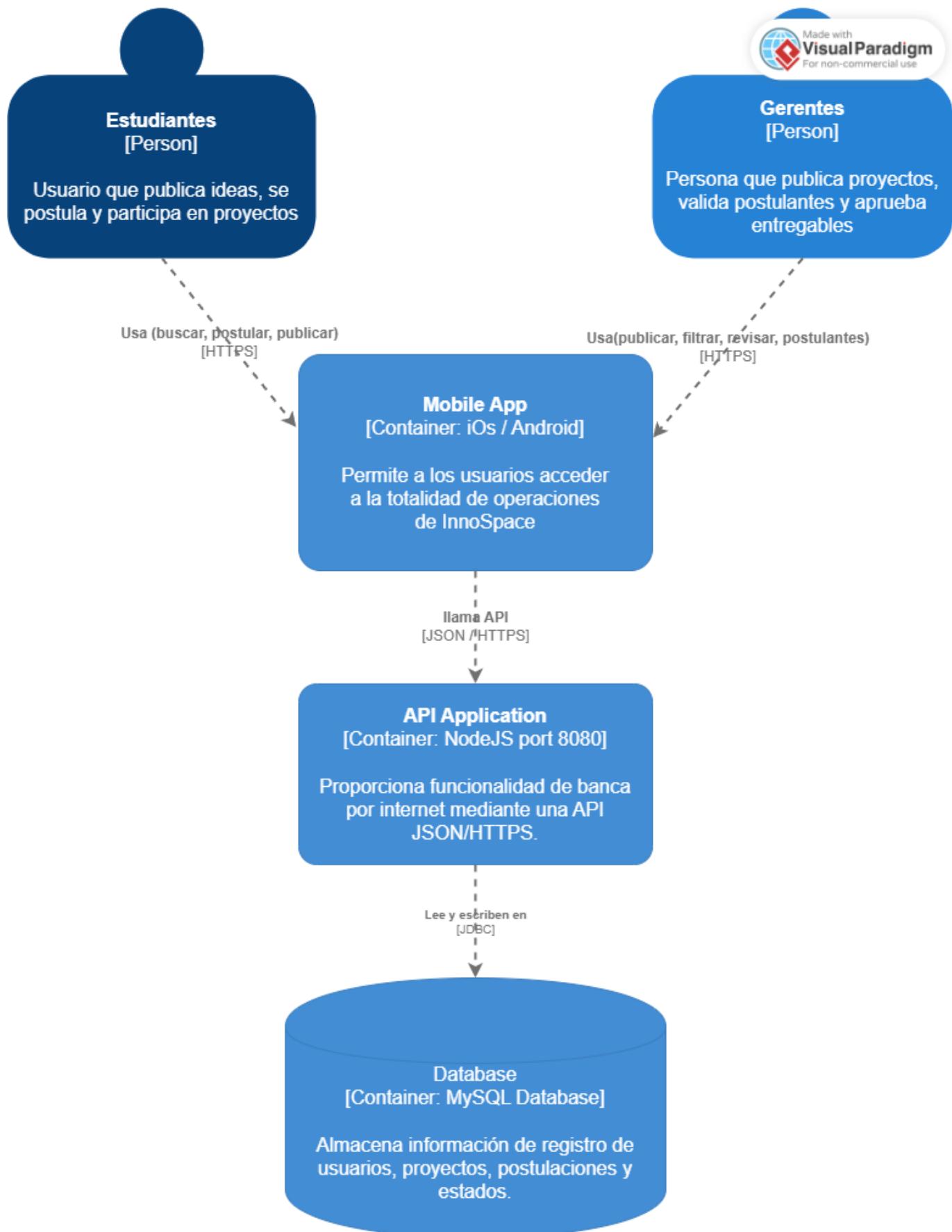
Destino (Downstream)	Origen (Upstream)	Tipo de Relación	Comentario
Project Collaboration	Company Opportunities	Partnership	Coordinación activa para gestión unificada de interacciones empresa-estudiante

2.5.3. Software Architecture

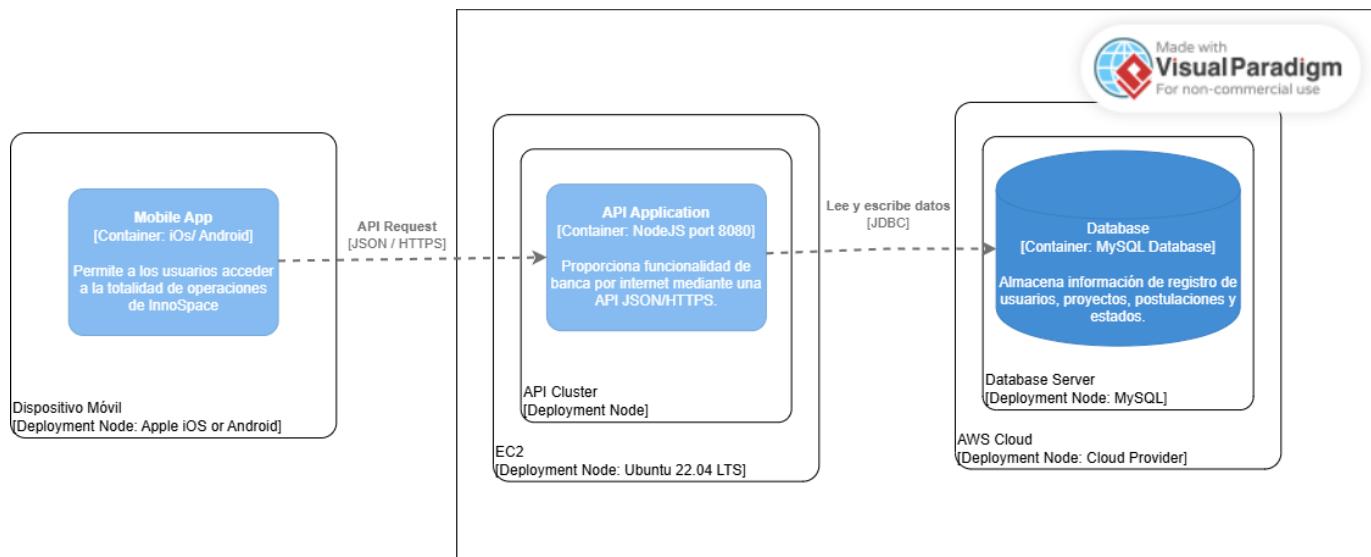
2.5.3.1. Software Architecture Context Level Diagrams



2.5.3.2. Software Architecture Container Level Diagrams



2.5.3.3. Software Architecture Deployment Diagrams



2.6.Tactical-Level Domain-Driven Design

2.6.1. Bounded Context: Student Projects

2.6.1.1. Domain Layer

El agregado Project encapsula el ciclo de vida de un proyecto estudiantil, permitiendo crearlo, actualizarlo, publicarlo y marcarlo como completado, asegurando que el estado (Draft, Published, Completed) y los contenidos asociados se mantengan consistentes y válidos.

Aggregate: Project

Descripción: Representa un proyecto o idea innovadora creada por un estudiante, con ciclo de vida completo desde borrador hasta finalizado, incluyendo contenido y estado.

Atributos	Tipo de dato	Visibilidad	Descripción
projectId	Long	Private	Identificador único del proyecto.
studentId	Long	Private	Identificador del estudiante propietario.
title	String	Private	Título del proyecto.
description	String	Private	Descripción detallada del proyecto.
status	ProjectStatus	Private	Estado del proyecto (Draft, Published, Completed).
creationDate	Timestamp	Private	Fecha de creación del proyecto.
lastUpdate	Timestamp	Private	Fecha de última actualización.
Métodos	Tipo de retorno	Visibilidad	Descripción
getProjectId()	Long	Public	Devuelve el ID del proyecto.
getStudentId()	Long	Public	Devuelve el ID del estudiante propietario.
getTitle()	String	Public	Devuelve el título del proyecto.
getDescription()	String	Public	Devuelve la descripción del proyecto.

Métodos	Tipo de retorno	Visibilidad	Descripción
getStatus()	ProjectStatus	Public	Devuelve el estado actual del proyecto.
publish()	void	Public	Cambia el estado del proyecto a Published .
update(title, description)	void	Public	Actualiza los campos modificables del proyecto.
finalize()	void	Public	Cambia el estado del proyecto a Completed .
getCreationDate()	Timestamp	Public	Devuelve la fecha de creación.
getLastUpdate()	Timestamp	Public	Devuelve la fecha de última actualización.

Value Objects

Value Object	Descripción
ProjectStatus	Representa el estado del proyecto: Draft , Published , Completed .
ProjectContent	Contiene archivos, imágenes, prototipos o documentación asociada al proyecto.

Clase: **ProjectQueryService**

Título	ProjectQueryService
Descripción	Interfaz de servicio de consultas para operaciones de lectura relacionadas con proyectos de estudiantes
Métodos	
Método	Descripción
handle(GetProjectByIdQuery)	Obtiene los detalles completos de un proyecto por su ID
handle(GetAllStudentProjectsQuery)	Obtiene todos los proyectos asociados a un estudiante
handle(ValidateProjectOwnershipQuery)	Verifica si un estudiante es propietario del proyecto

Clase: **ProjectCommandService**

Título	ProjectCommandService
Descripción	Interfaz de servicio de comandos para operaciones de escritura relacionadas con proyectos de estudiantes
Métodos	
Método	Descripción
handle(CreateProjectCommand)	Crea un nuevo proyecto con la información proporcionada
handle(UpdateProjectCommand)	Actualiza la información básica del proyecto (título, descripción, contenido)

Método	Descripción
handle(PublishProjectCommand)	Cambia el estado del proyecto a publicado
handle(FinalizeProjectCommand)	Cambia el estado del proyecto a completado
handle>DeleteProjectCommand	Elimina un proyecto del sistema

2.6.1.2. Interface Layer

El ProjectController expone endpoints REST que permiten a los estudiantes gestionar sus proyectos, desde la creación hasta el cierre, utilizando assemblers para mapear recursos REST a comandos de dominio y conectando las solicitudes del cliente con los servicios de aplicación.

Controlador: **ProjectController**

Título: ProjectController

Descripción: Controlador REST que maneja las operaciones CRUD y gestión de proyectos de estudiantes.

Métodos

Método	Ruta	Descripción
getProjectById	GET /api/v1/projects/{id}	Obtiene los detalles de un proyecto específico por su ID
getAllStudentProjects	GET /api/v1/projects/student/{studentId}	Obtiene todos los proyectos asociados a un estudiante
createProject	POST /api/v1/projects	Crea un nuevo proyecto a partir de los datos proporcionados
updateProject	PUT /api/v1/projects/{id}	Actualiza los detalles de un proyecto existente
publishProject	POST /api/v1/projects/{id}/publish	Cambia el estado del proyecto a publicado
finalizeProject	POST /api/v1/projects/{id}/finalize	Cambia el estado del proyecto a completado
deleteProject	DELETE /api/v1/projects/{id}	Elimina un proyecto del sistema

Dependencias

Dependencia	Descripción
ProjectQueryService	Servicio para consultas y recuperación de datos de proyectos
ProjectCommandService	Servicio para ejecutar comandos de creación, actualización, publicación y eliminación de proyectos
CreateProjectCommandFromResourceAssembler	Convierte recursos REST en comandos de creación de proyectos
UpdateProjectCommandFromResourceAssembler	Convierte recursos REST en comandos de actualización de proyectos

Dependencia	Descripción
DeleteProjectCommandFromResourceAssembler	Convierte recursos REST en comandos de eliminación de proyectos
ProjectResourceFromEntityAssembler	Convierte entidades de proyecto en recursos REST para la respuesta

2.6.1.3. Application Layer

Los servicios ProjectCommandServiceImpl y ProjectQueryServiceImpl implementan los casos de uso definidos en el dominio, coordinando la creación, actualización, publicación, finalización y consulta de proyectos, sin exponer detalles de persistencia ni de transporte.

Clase: [ProjectQueryServiceImpl](#)

Título	ProjectQueryServiceImpl
Descripción	Implementación del servicio de consultas para operaciones de lectura relacionadas con proyectos de estudiantes
Dependencias	
Dependencia	Descripción
ProjectRepository	Repositorio para acceso a datos de proyectos
ProjectQueryService	Interfaz de consultas definida en el dominio

Clase: [ProjectCommandServiceImpl](#)

Título	ProjectCommandServiceImpl
Descripción	Implementación del servicio de comandos para operaciones de escritura relacionadas con proyectos de estudiantes
Dependencias	
Dependencia	Descripción
ProjectRepository	Repositorio para acceso a datos de proyectos
ProjectCommandService	Interfaz de comandos definida en el dominio

2.6.1.4 Infrastructure Layer

El ProjectRepository provee los mecanismos de persistencia para proyectos, con operaciones CRUD y consultas específicas (por ID o por estudiante), integrando las entidades de dominio con representaciones en la base de datos mediante ProjectEntity.

Clase: [ProjectRepository](#)

Título	ProjectRepository
Descripción	Interfaz de persistencia para operaciones CRUD y consultas específicas de proyectos de estudiantes
Métodos	
Método	Descripción
save(ProjectEntity)	Persiste un nuevo proyecto o actualiza uno existente
deleteById(Long)	Elimina un proyecto por su ID
findById(Long)	Obtiene un proyecto por su ID
findAllByStudentId(Long)	Obtiene todos los proyectos asociados a un estudiante

Dependencias

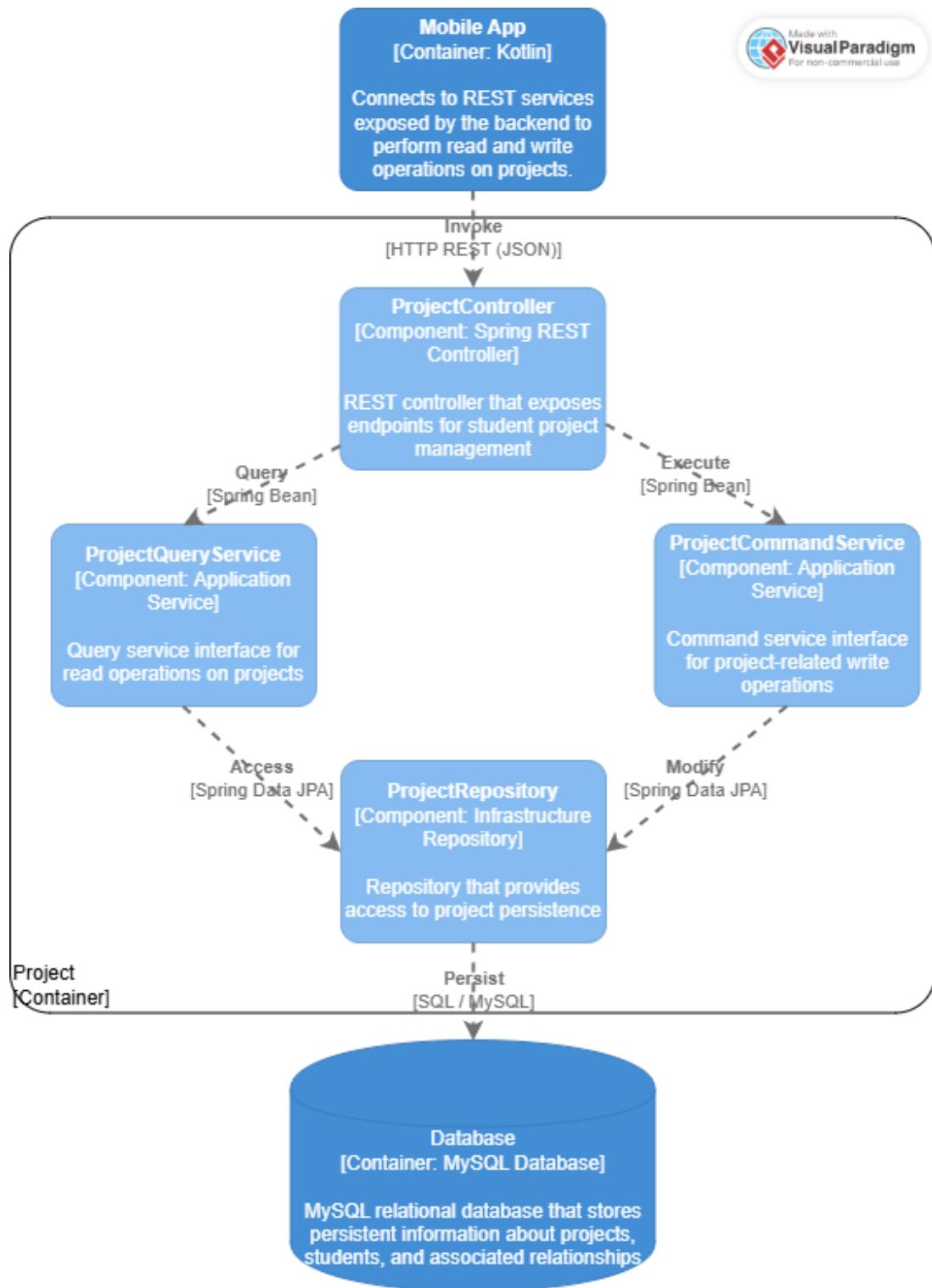
Dependencia	Propósito
Student	Clase que representa al estudiante en el sistema
ProjectEntity	Clase que representa la entidad de proyecto en la base de datos

2.6.1.5. Bounded Context Software Architecture Component Level Diagrams

Este diagrama de componentes representa un sistema monolítico que gestiona proyectos estudiantiles dentro de la plataforma. Una aplicación Movil interactúa con el [ProjectController](#).

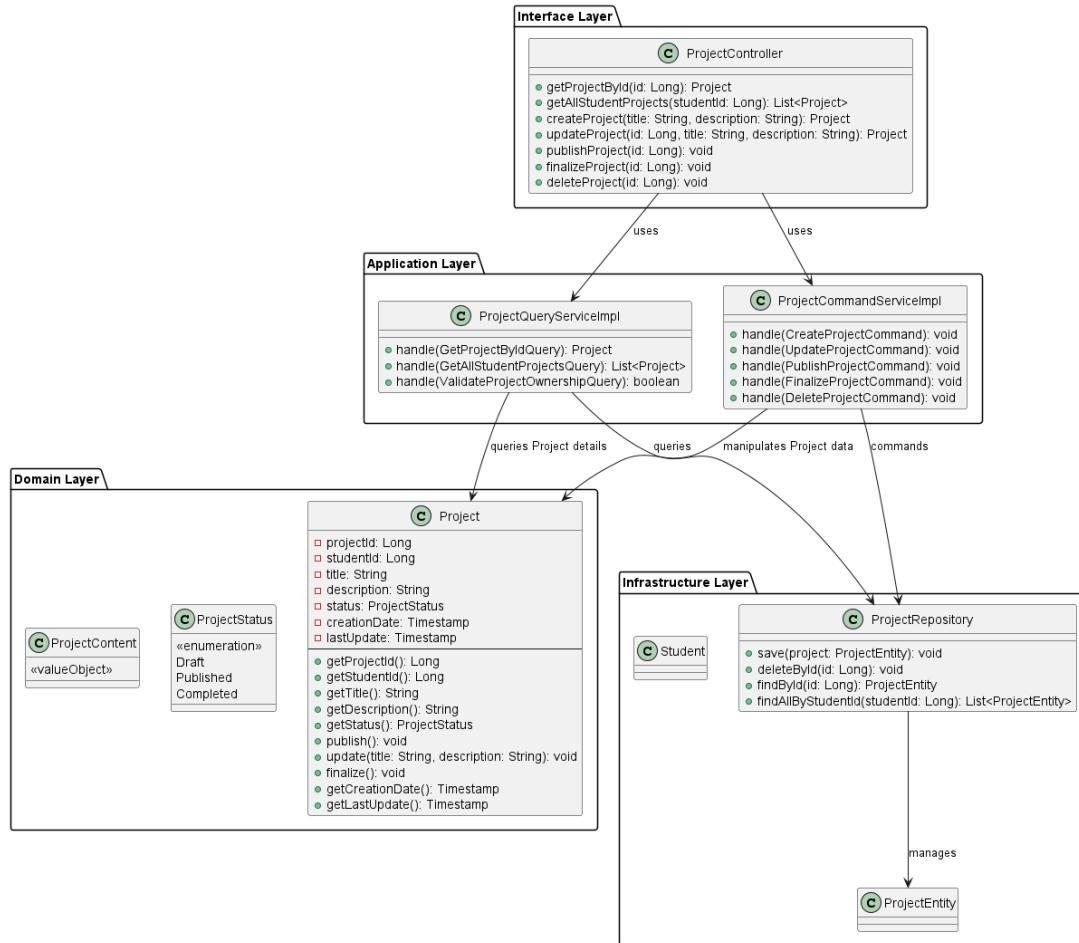
El controlador delega la lógica en dos servicios principales: [ProjectQueryService](#), para recuperar y validar proyectos, y [ProjectCommandService](#), para crearlos, actualizarlos, publicarlos o finalizarlos.

Ambos servicios utilizan el [ProjectRepository](#), que gestiona la persistencia mediante JPA en una base de datos MySQL.



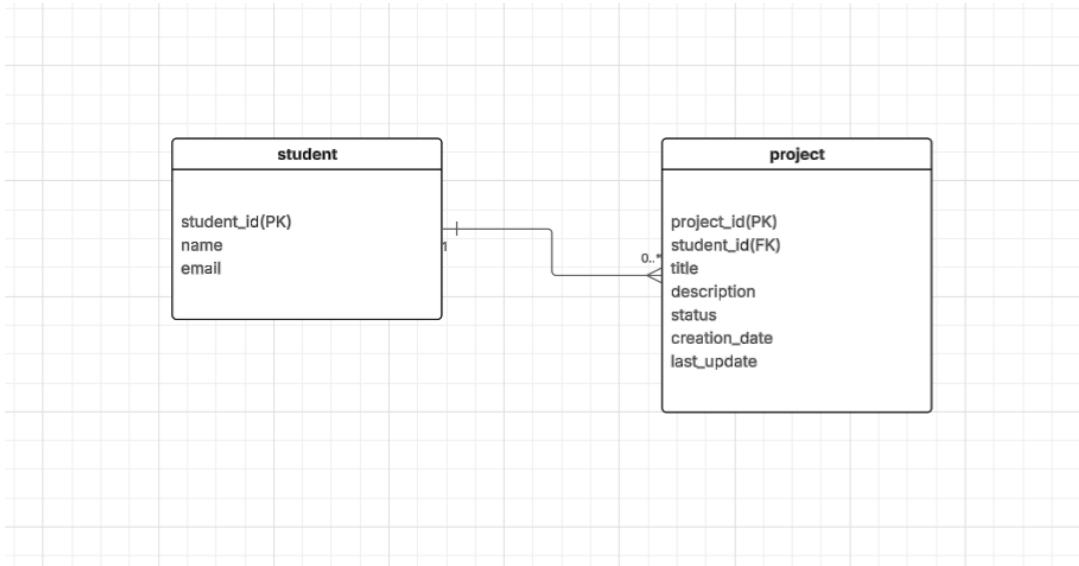
2.6.1.6. Bounded Context Software Architecture Code Level Diagrams

2.6.1.6.1. Bounded Context Domain Layer Class Diagrams



Bounded Context Class Diagram - Elaboración propia

2.6.1.6.2. Bounded Context Database Design Diagrams



2.6.2. Bounded Context: Company Opportunities

2.6.2.1. Domain Layer

El agregado Opportunity encapsula la lógica de negocio de una convocatoria publicada por una empresa, controlando su ciclo de vida (Draft, Published, Closed), validando criterios y actualizaciones, y garantizando consistencia mediante Value Objects como OpportunityStatus y OpportunityRequirements.

Aggregate: Opportunity

Descripción: Representa una convocatoria u oportunidad publicada por una empresa, con detalles, criterios de selección y estado.

Atributos	Tipo de dato	Visibilidad	Descripción
opportunityId	Long	Private	Identificador único de la oportunidad
companyId	Long	Private	Identificador de la empresa que publica la oportunidad
title	String	Private	Título de la convocatoria
description	String	Private	Descripción detallada de la oportunidad
requirements	List	Private	Lista de habilidades o criterios requeridos
status	OpportunityStatus	Private	Estado de la oportunidad (Draft , Published , Closed)
creationDate	Timestamp	Private	Fecha de creación
lastUpdate	Timestamp	Private	Fecha de última actualización

Métodos

Método	Tipo de retorno	Visibilidad	Descripción
getOpportunityId()	Long	Public	Devuelve el ID de la oportunidad
getCompanyId()	Long	Public	Devuelve el ID de la empresa
getTitle()	String	Public	Devuelve el título de la oportunidad
getDescription()	String	Public	Devuelve la descripción de la oportunidad
getRequirements()	List	Public	Devuelve los requisitos o habilidades
getStatus()	OpportunityStatus	Public	Devuelve el estado actual
publish()	void	Public	Cambia el estado de la oportunidad a Published
close()	void	Public	Cambia el estado de la oportunidad a Closed
update(title, description, requirements)	void	Public	Actualiza los datos básicos de la oportunidad
getCreationDate()	Timestamp	Public	Devuelve la fecha de creación
getLastUpdate()	Timestamp	Public	Devuelve la fecha de última actualización

Value Objects

Value Object	Descripción
--------------	-------------

Value Object	Descripción
OpportunityStatus	Representa el estado de la oportunidad: Draft , Published , Closed
OpportunityRequirements	Contiene las habilidades y criterios que la empresa espera de los postulantes

Clase: [OpportunityQueryService](#)

Título	
OpportunityQueryService	
Descripción	
Interfaz de servicio de consultas para operaciones de lectura relacionadas con oportunidades de empresa	
Métodos	
Método	Descripción
handle(GetOpportunityByIdQuery)	Obtiene los detalles completos de una oportunidad por su ID
handle(GetAllCompanyOpportunitiesQuery)	Obtiene todas las oportunidades asociadas a una empresa
handle(ValidateOpportunityOwnershipQuery)	Verifica si una empresa es propietaria de la oportunidad

Clase: [OpportunityCommandService](#)

Título	
OpportunityCommandService	
Descripción	
Interfaz de servicio de comandos para operaciones de escritura relacionadas con oportunidades de empresa	
Métodos	
Método	Descripción
handle(CreateOpportunityCommand)	Crea una nueva oportunidad con la información proporcionada
handle(UpdateOpportunityCommand)	Actualiza los detalles de la oportunidad (título, descripción, requisitos)
handle(PublishOpportunityCommand)	Cambia el estado de la oportunidad a publicada
handle(CloseOpportunityCommand)	Cambia el estado de la oportunidad a cerrada
handle>DeleteOpportunityCommand)	Elimina una oportunidad del sistema

2.6.2.2. Interface Layer

El OpportunityController expone endpoints REST que permiten a clientes externos crear, actualizar, publicar, cerrar y consultar convocatorias, utilizando assemblers para mapear recursos REST a comandos y entidades de dominio, y conectándose a los servicios de aplicación para ejecutar la lógica correspondiente.

Controlador: [OpportunityController](#)

Título: OpportunityController

Descripción: Controlador REST que maneja las operaciones CRUD y gestión de oportunidades/convocatorias de empresas.

Métodos

Método	Ruta	Descripción
getOpportunityById	GET /api/v1/opportunities/{id}	Obtiene los detalles de una oportunidad específica por su ID
getAllCompanyOpportunities	GET /api/v1/opportunities/company/{companyId}	Obtiene todas las oportunidades asociadas a una empresa
createOpportunity	POST /api/v1/opportunities	Crea una nueva oportunidad a partir de los datos proporcionados
updateOpportunity	PUT /api/v1/opportunities/{id}	Actualiza los detalles de una oportunidad existente
publishOpportunity	POST /api/v1/opportunities/{id}/publish	Cambia el estado de la oportunidad a publicada
closeOpportunity	POST /api/v1/opportunities/{id}/close	Cambia el estado de la oportunidad a cerrada
deleteOpportunity	DELETE /api/v1/opportunities/{id}	Elimina una oportunidad del sistema

Dependencias

Dependencia	Descripción
OpportunityQueryService	Servicio para consultas y recuperación de datos de oportunidades
OpportunityCommandService	Servicio para ejecutar comandos de creación, actualización, publicación, cierre y eliminación de oportunidades
CreateOpportunityCommandFromResourceAssembler	Convierte recursos REST en comandos de creación de oportunidades
UpdateOpportunityCommandFromResourceAssembler	Convierte recursos REST en comandos de actualización de oportunidades
DeleteOpportunityCommandFromResourceAssembler	Convierte recursos REST en comandos de eliminación de oportunidades
OpportunityResourceFromEntityAssembler	Convierte entidades de oportunidad en recursos REST para la respuesta

2.6.2.3. Application Layer

Los servicios OpportunityCommandServiceImpl y OpportunityQueryServiceImpl implementan los casos de uso de creación, actualización, publicación, cierre y consulta de oportunidades, actuando como orquestadores entre el dominio y la infraestructura, sin exponer detalles técnicos.

Clase: [OpportunityQueryServiceImpl](#)

Título	OpportunityQueryServiceImpl
Descripción	Implementación del servicio de consultas para operaciones de lectura relacionadas con oportunidades de empresa
Dependencias	
Dependencia	Descripción
OpportunityRepository	Repositorio para acceso a datos de oportunidades
OpportunityQueryService	Interfaz de consultas definida en el dominio

Clase: [OpportunityCommandServiceImpl](#)

Título	OpportunityCommandServiceImpl
Descripción	Implementación del servicio de comandos para operaciones de escritura relacionadas con oportunidades de empresa
Dependencias	
Dependencia	Descripción
OpportunityRepository	Repositorio para acceso a datos de oportunidades
OpportunityCommandService	Interfaz de comandos definida en el dominio

2.6.2.4. Infrastructure Layer

La clase OpportunityRepository gestiona la persistencia y recuperación de oportunidades en la base de datos, con operaciones CRUD y consultas específicas por empresa, asegurando la integración del dominio con entidades como Company y OpportunityEntity.

Clase: [OpportunityRepository](#)

Título	OpportunityRepository
Descripción	Interfaz de persistencia para operaciones CRUD y consultas específicas de oportunidades de empresa
Métodos	

Método	Descripción
save(OpportunityEntity)	Persiste una nueva oportunidad o actualiza una existente
deleteById(Long)	Elimina una oportunidad por su ID
findById(Long)	Obtiene una oportunidad por su ID
findAllByCompanyId(Long)	Obtiene todas las oportunidades asociadas a una empresa

Dependencias

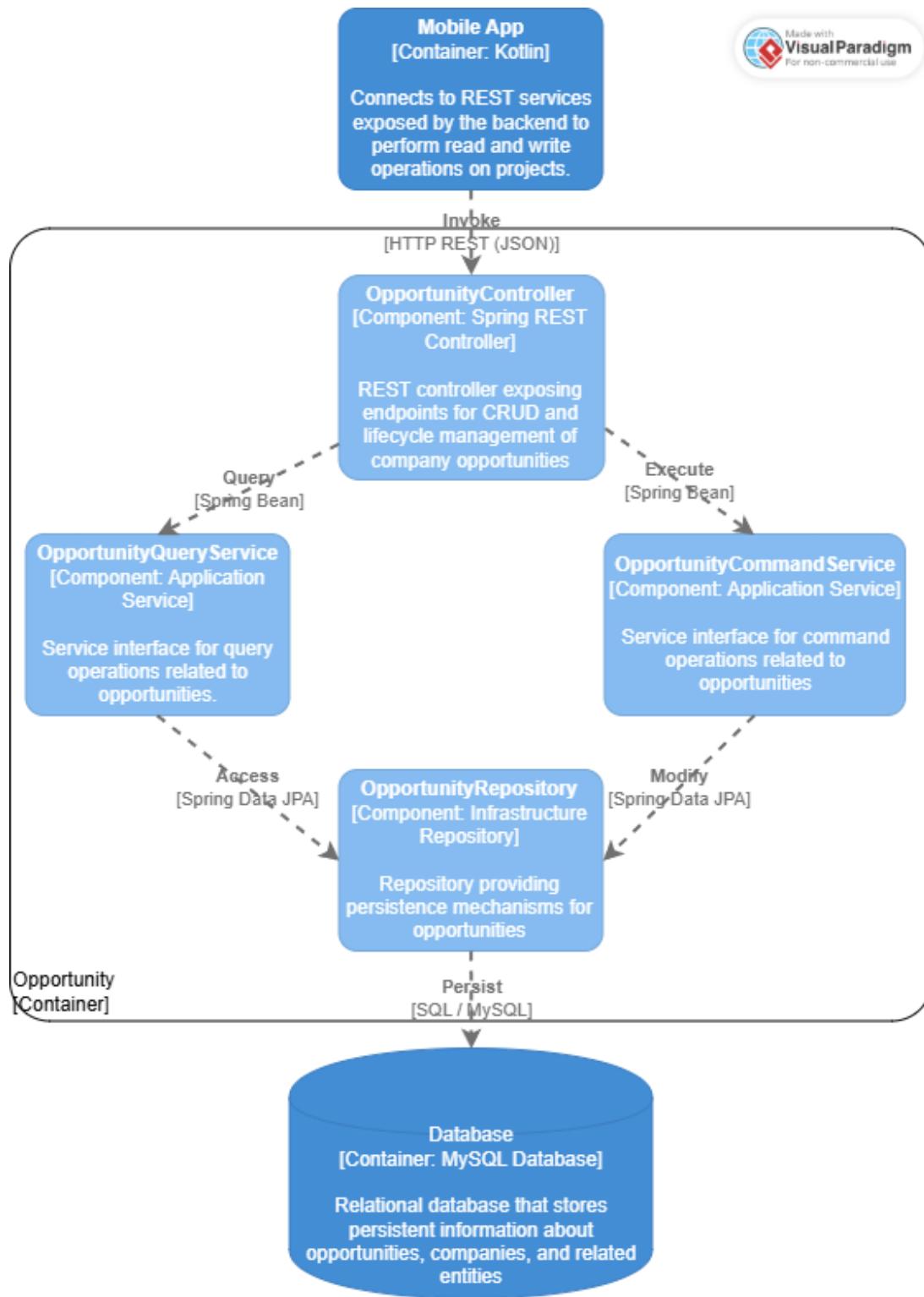
Dependencia	Propósito
Company	Clase que representa a la empresa en el sistema
OpportunityEntity	Clase que representa la entidad de oportunidad en la base de datos

2.6.2.5. Bounded Context Software Architecture Component Level Diagrams

Este diagrama de componentes representa un sistema monolítico que gestiona oportunidades empresariales dentro de la plataforma. Una aplicación Movil interactúa con el [OpportunityController](#).

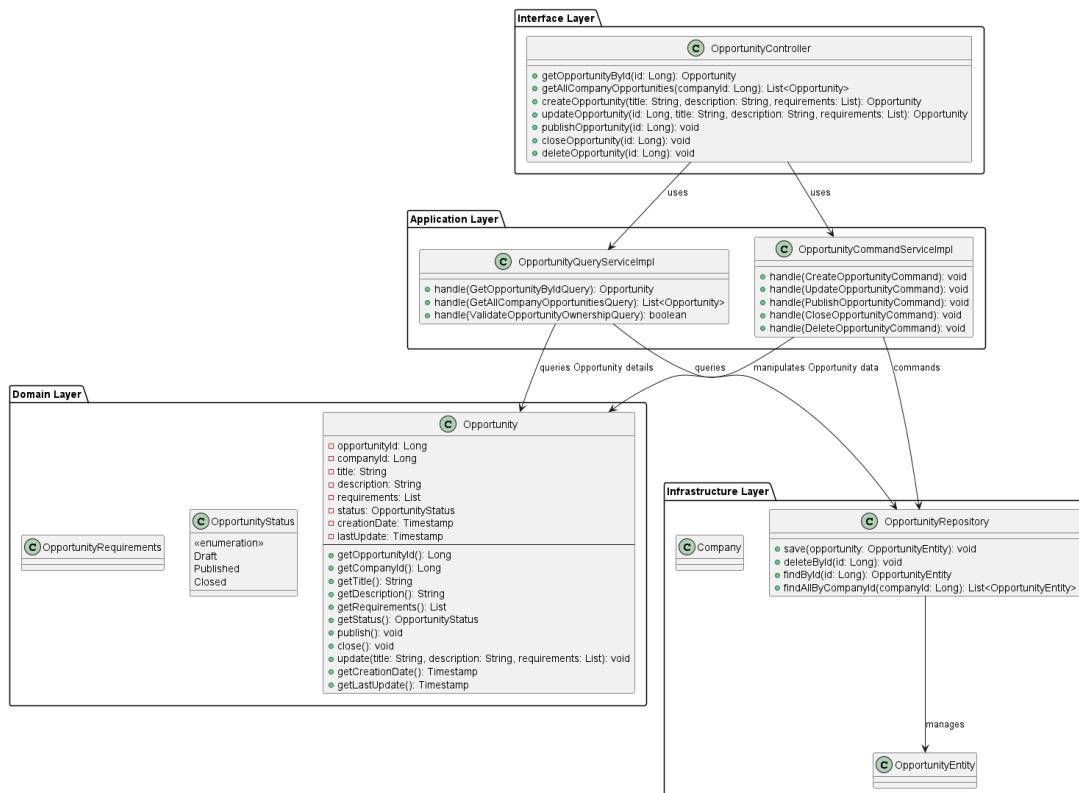
El controlador delega la lógica en dos servicios principales: [OpportunityQueryService](#), para consultas y validaciones, y [OpportunityCommandService](#), para creación, actualización, publicación y cierre de oportunidades.

Ambos servicios utilizan el [OpportunityRepository](#), que maneja la persistencia mediante JPA en una base de datos MySQL.



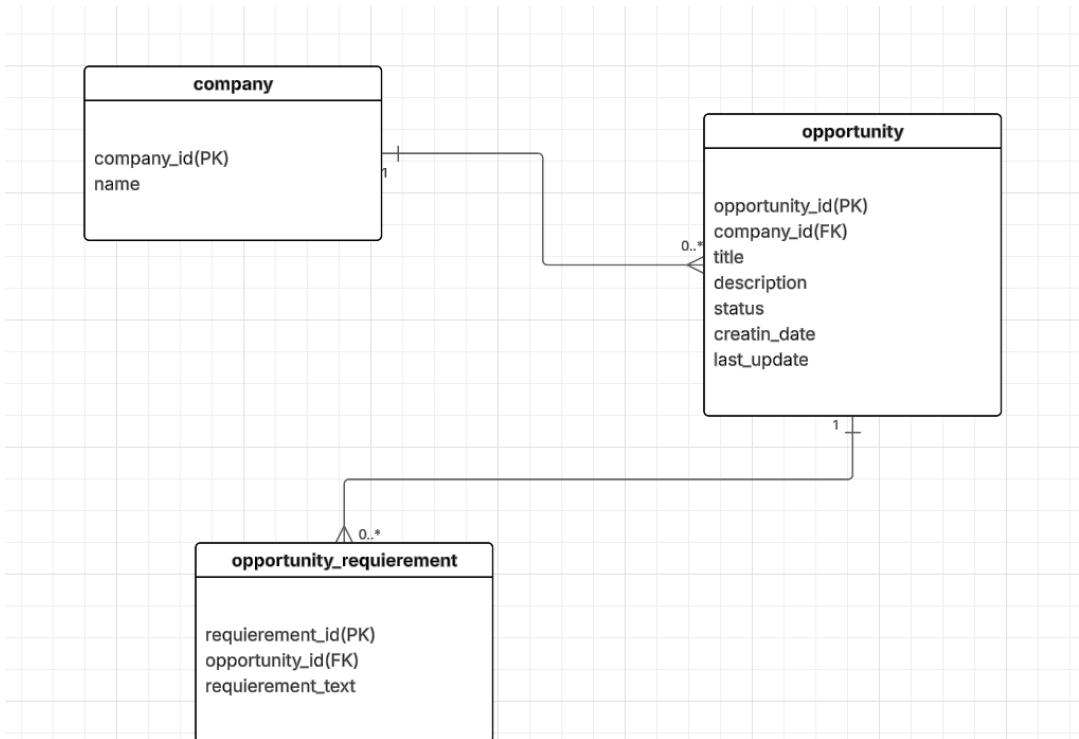
2.6.2.6. Bounded Context Software Architecture Code Level Diagrams

2.6.2.6.1. Bounded Context Domain Layer Class Diagrams



Bounded Context Class Diagram - Elaboración propia

2.6.2.6.2. Bounded Context Database Design Diagrams



2.6.3. Bounded Context: Project Collaboration

2.6.3.1. Domain Layer

Contiene la lógica principal sobre cómo las empresas (gerentes) toman decisiones respecto a proyectos estudiantiles. Incluye el agregado CollaborationDecision, con atributos como projectId, companyId, decisionStatus (interesado, aceptado, rechazado) y feedback. También maneja entidades de soporte como Manager y StudentProject.

Aggregate: **CollaborationDecision**

Descripción: Representa la decisión de un gerente sobre un proyecto estudiantil y la respuesta del estudiante, incluyendo estado de interés, aceptación/rechazo y observaciones.

Atributos	Tipo de dato	Visibilidad	Descripción
decisionId	Long	Private	Identificador único de la decisión
projectId	Long	Private	Identificador del proyecto asociado
companyId	Long	Private	Identificador de la empresa o gerente evaluador
status	CollaborationStatus	Private	Estado de la decisión del gerente (Interested , NotInterested)
studentResponse	StudentResponseStatus	Private	Respuesta del estudiante (Pending , Accepted , Rejected)
evaluationNotes	String	Private	Observaciones o criterios de evaluación del gerente
creationDate	Timestamp	Private	Fecha de creación de la decisión
lastUpdate	Timestamp	Private	Fecha de última actualización

Métodos

Método	Tipo de retorno	Visibilidad	Descripción
getDecisionId()	Long	Public	Devuelve el ID de la decisión
getprojectId()	Long	Public	Devuelve el ID del proyecto asociado
getCompanyId()	Long	Public	Devuelve el ID de la empresa evaluadora
getStatus()	CollaborationStatus	Public	Devuelve el estado de la decisión del gerente
markInterested()	void	Public	Cambia el estado del gerente a Interested
markNotInterested()	void	Public	Cambia el estado del gerente a NotInterested
addEvaluationNotes(notes)	void	Public	Agrega observaciones de evaluación

Método	Tipo de retorno	Visibilidad	Descripción
getEvaluationNotes()	String	Public	Devuelve las notas de evaluación
submitStudentResponse(response)	void	Public	Permite al estudiante aceptar o rechazar la colaboración
getStudentResponse()	StudentResponseStatus	Public	Devuelve el estado de respuesta del estudiante
getCreationDate()	Timestamp	Public	Devuelve la fecha de creación
getLastUpdate()	Timestamp	Public	Devuelve la fecha de última actualización

Value Objects

Value Object	Descripción
CollaborationStatus	Representa el estado de la decisión del gerente: Interested, NotInterested
StudentResponseStatus	Representa la respuesta del estudiante: Pending, Accepted, Rejected
EvaluationNotes	Contiene observaciones o criterios de evaluación del gerente

Interfaz: CollaborationDecisionQueryService

Título	CollaborationDecisionQueryService
Descripción	Contrato para operaciones de lectura relacionadas con decisiones de colaboración

Métodos

Método	Descripción
handle(GetDecisionByIdQuery)	Obtiene los detalles completos de una decisión por su ID
handle(GetDecisionsByProjectQuery)	Obtiene todas las decisiones asociadas a un proyecto
handle(GetDecisionsByCompanyQuery)	Obtiene todas las decisiones tomadas por una empresa

Interfaz: CollaborationDecisionCommandService

Título	CollaborationDecisionCommandService
Descripción	Contrato para operaciones de escritura relacionadas con decisiones de colaboración

Métodos

Método	Descripción

Método	Descripción
handle(CreateCollaborationDecisionCommand)	Crea una nueva decisión de colaboración (Interested / NotInterested)
handle(AddEvaluationNotesCommand)	Agrega notas de evaluación a una decisión
handle(SubmitStudentResponseCommand)	Permite al estudiante aceptar o rechazar la colaboración
handle>DeleteCollaborationDecisionCommand)	Elimina una decisión del sistema

2.6.3.2. Interface Layer

Se refleja en la aplicación móvil, mediante los formularios que puede enviar el gerente para colaborar con un proyecto. Los gerentes envían propuestas de colaboración a proyectos específicos, y los estudiantes pueden revisarlas y responder (aceptar o rechazar).

Controlador: **CollaborationDecisionController**

Título: CollaborationDecisionController

Descripción: Controlador REST que maneja la visualización de proyectos, registro de decisiones de gerentes y respuesta de estudiantes sobre colaboraciones.

Métodos

Método	Ruta	Descripción
getDecisionById	GET /api/v1/collaboration/{id}	Obtiene los detalles de una decisión específica por ID
getAllDecisionsByProject	GET /api/v1/collaboration/project/{projectId}	Obtiene todas las decisiones asociadas a un proyecto
getAllDecisionsByCompany	GET /api/v1/collaboration/company/{companyId}	Obtiene todas las decisiones tomadas por una empresa
createDecision	POST /api/v1/collaboration	Permite al gerente registrar su decisión (Interested / NotInterested)
addEvaluationNotes	POST /api/v1/collaboration/{id}/notes	Agrega observaciones o criterios de evaluación a la decisión
submitStudentResponse	POST /api/v1/collaboration/{id}/response	Permite al estudiante aceptar o rechazar la colaboración
deleteDecision	DELETE /api/v1/collaboration/{id}	Elimina una decisión de colaboración del sistema

Dependencias

Dependencia	Descripción

Dependencia	Descripción
CollaborationDecisionQueryService	Servicio para consultas y recuperación de decisiones de colaboración
CollaborationDecisionCommandService	Servicio para ejecutar comandos de creación, notas, respuesta y eliminación
CreateCollaborationDecisionCommandFromResourceAssembler	Convierte recursos REST en comandos para crear decisiones
AddEvaluationNotesCommandFromResourceAssembler	Convierte recursos REST en comandos para añadir notas
SubmitStudentResponseCommandFromResourceAssembler	Convierte recursos REST en comandos para que el estudiante acepte/rechace
DeleteCollaborationDecisionCommandFromResourceAssembler	Convierte recursos REST en comandos para eliminar decisiones
CollaborationDecisionResourceFromEntityAssembler	Convierte entidades en recursos REST para la respuesta

2.6.3.3. Application Layer

Orquesta los casos de uso para enviar, aceptar o rechazar decisiones de colaboración. Expone servicios de comandos (CollaborationDecisionCommandService) para procesar acciones y servicios de consulta (CollaborationDecisionQueryService) para obtener el estado de las decisiones.

Clase: **CollaborationDecisionQueryServiceImpl**

Título	CollaborationDecisionQueryServiceImpl
Descripción	Implementación del contrato de consultas definido en el dominio para decisiones de colaboración

Dependencias:

Dependencia	Descripción
CollaborationDecisionQueryService	Contrato definido en dominio
CollaborationDecisionRepository	Repositorio para acceso a datos de decisiones

Clase: **CollaborationDecisionCommandServiceImpl**

Título	CollaborationDecisionCommandServiceImpl
Descripción	Implementación del contrato de comandos definido en el dominio para decisiones de colaboración

Dependencias:

Dependencia	Descripción
CollaborationDecisionCommandService	Contrato definido en dominio
CollaborationDecisionRepository	Repositorio para acceso a datos de decisiones

2.6.3.4. Infrastructure Layer

Implementa la persistencia de las decisiones de colaboración y la integración con servicios externos. Incluye repositorios como CollaborationDecisionRepository y adaptadores para acceder a datos de proyectos y empresas.

Clase: [CollaborationDecisionRepository](#)

Título	CollaborationDecisionRepository
Descripción	Interfaz de persistencia para operaciones CRUD y consultas específicas de decisiones de colaboración

Métodos

Método	Descripción
save(CollaborationDecisionEntity)	Persiste una nueva decisión o actualiza una existente
deleteById(Long)	Elimina una decisión por su ID
findById(Long)	Recupera una decisión específica por ID
findByProjectId(Long)	Recupera todas las decisiones asociadas a un proyecto
findByCompanyId(Long)	Recupera todas las decisiones tomadas por una empresa

Dependencias

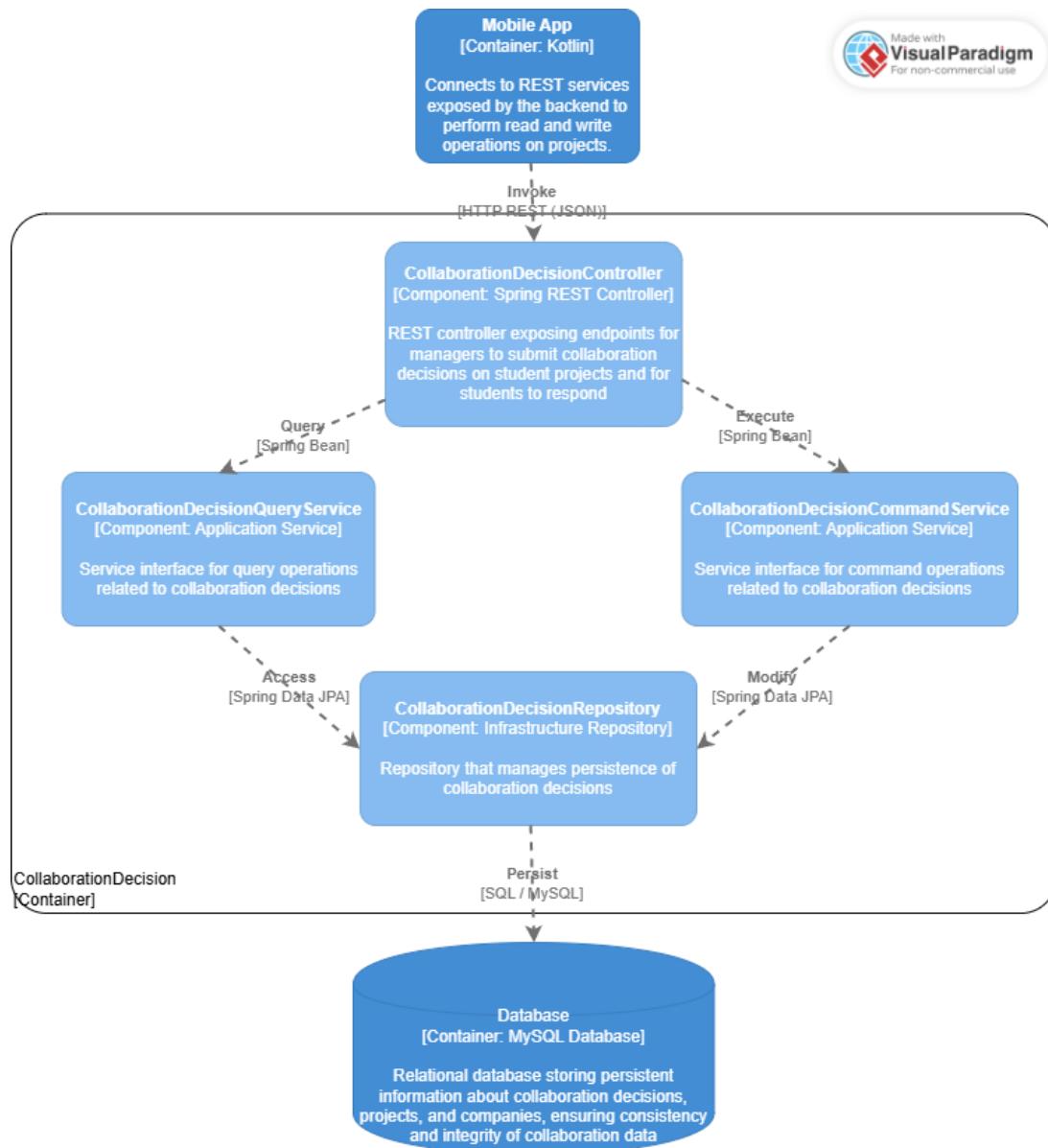
Dependencia	Propósito
CollaborationDecisionEntity	Entidad que representa la decisión en la base de datos
Project	Representa el proyecto asociado
Company	Representa la empresa o gerente evaluador

2.6.3.5. Bounded Context Software Architecture Component Level Diagrams

Este diagrama de componentes representa un sistema monolítico que gestiona decisiones de colaboración entre empresas y proyectos estudiantiles. Una aplicación Móvil interactúa con el [CollaborationDecisionController](#).

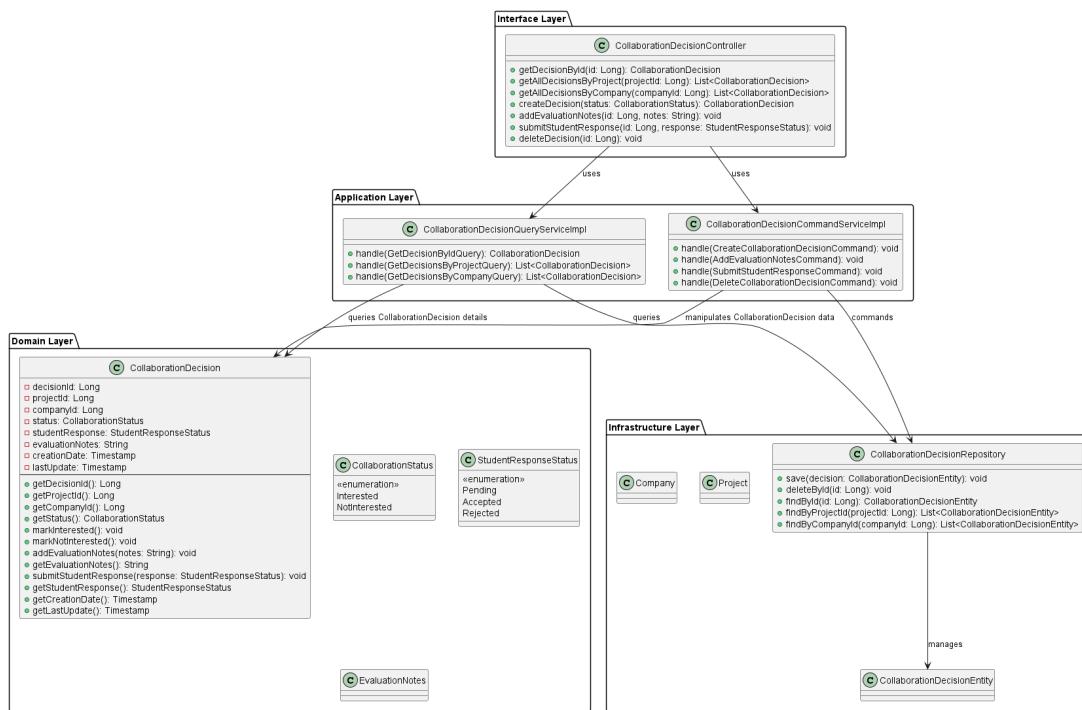
El controlador conecta con [CollaborationDecisionQueryService](#) para consultas y con [CollaborationDecisionCommandService](#) para registrar decisiones, añadir notas y gestionar respuestas de estudiantes.

Ambos servicios acceden al [CollaborationDecisionRepository](#), encargado de la persistencia mediante JPA en una base de datos MySQL.



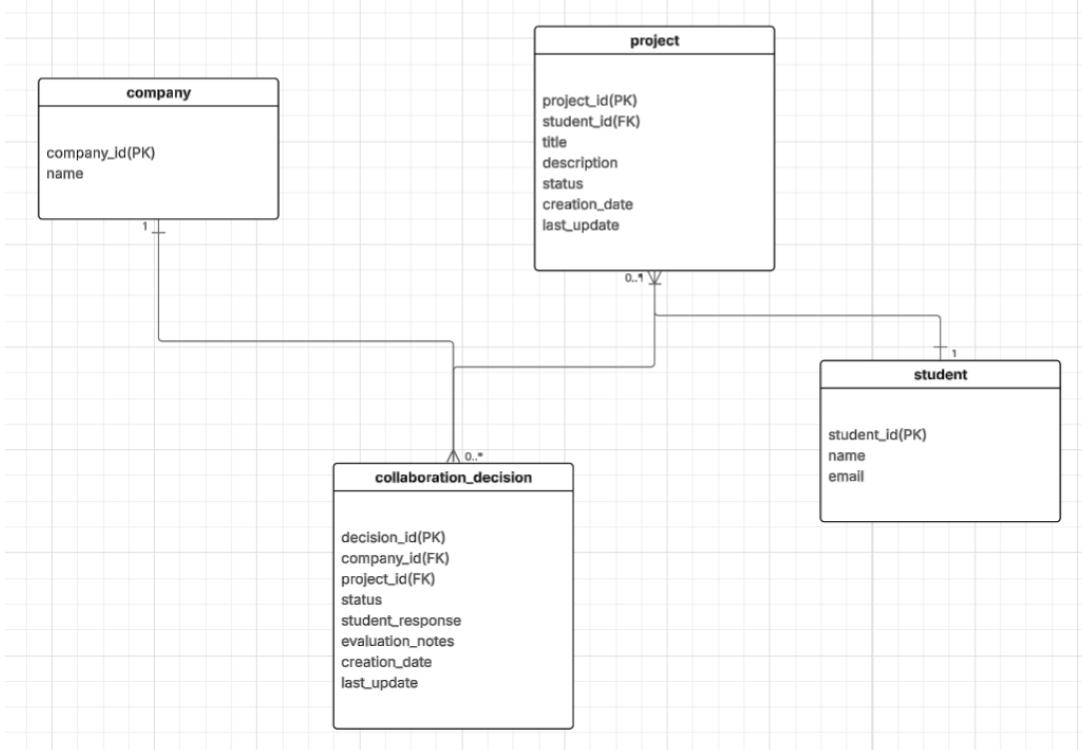
2.6.3.6. Bounded Context Software Architecture Code Level Diagrams

2.6.3.6.1. Bounded Context Domain Layer Class Diagrams



Bounded Context Class Diagram - Elaboración propia

2.6.3.6.2. Bounded Context Database Design Diagrams



2.6.4. Bounded Context: Student Applications

2.6.4.1. Domain Layer

Define el agregado Application, que encapsula los datos y reglas de negocio de una postulación (estado, fechas, vínculos a estudiante, proyecto y oportunidad), asegurando consistencia a través de Value Objects como `ApplicationStatus` y métodos de negocio (submit, accept, reject).

Aggregate: Application

Descripción: Representa la postulación de un estudiante a una oportunidad, incluyendo su proyecto asociado y el estado de la postulación.

Atributos	Tipo de dato	Visibilidad	Descripción
applicationId	Long	Private	Identificador único de la postulación
studentId	Long	Private	Identificador del estudiante que postula
opportunityId	Long	Private	Identificador de la oportunidad a la que postula
projectId	Long	Private	Identificador del proyecto que se postula
status	ApplicationStatus	Private	Estado de la postulación (Pending , Accepted , Rejected)
submissionDate	Timestamp	Private	Fecha de creación de la postulación
lastUpdate	Timestamp	Private	Fecha de última actualización de la postulación

Métodos

Método	Tipo de retorno	Visibilidad	Descripción
getApplicationId()	Long	Public	Devuelve el ID de la postulación
getStudentId()	Long	Public	Devuelve el ID del estudiante
getOpportunityId()	Long	Public	Devuelve el ID de la oportunidad
getProjectId()	Long	Public	Devuelve el ID del proyecto asociado
getStatus()	ApplicationStatus	Public	Devuelve el estado actual de la postulación
submit()	void	Public	Cambia el estado de la postulación a Pending
accept()	void	Public	Cambia el estado de la postulación a Accepted
reject()	void	Public	Cambia el estado de la postulación a Rejected
getSubmissionDate()	Timestamp	Public	Devuelve la fecha de creación de la postulación
getLastUpdate()	Timestamp	Public	Devuelve la fecha de última actualización

Value Objects

Value Object	Descripción
ApplicationStatus	Representa el estado de la postulación: Pending , Accepted , Rejected
ApplicationCriteria	Contiene los criterios de selección aplicados por la empresa

Interfaz: ApplicationQueryService

Título	ApplicationQueryService
Descripción	Contrato para operaciones de lectura relacionadas con postulaciones de estudiantes

Métodos

Método	Descripción
handle(GetApplicationByIdQuery)	Obtiene los detalles completos de una postulación por su ID
handle(GetAllStudentApplicationsQuery)	Obtiene todas las postulaciones asociadas a un estudiante
handle(GetAllOpportunityApplicationsQuery)	Obtiene todas las postulaciones a una oportunidad
handle(ValidateApplicationOwnershipQuery)	Verifica si un estudiante es propietario de la postulación

Interfaz: ApplicationCommandService

Título	ApplicationCommandService
Descripción	Contrato para operaciones de escritura relacionadas con postulaciones de estudiantes

Métodos

Método	Descripción
handle(SubmitApplicationCommand)	Crea una nueva postulación asociando estudiante, proyecto y oportunidad
handle(AcceptApplicationCommand)	Cambia el estado de la postulación a Accepted
handle(RejectApplicationCommand)	Cambia el estado de la postulación a Rejected
handle>DeleteApplicationCommand)	Elimina una postulación del sistema

2.6.4.2. Interface Layer

Expone un controlador REST (ApplicationController) que traduce las operaciones del dominio en endpoints accesibles al cliente, manejando rutas para crear, aceptar, rechazar o listar postulaciones, y apoyándose en assemblers para transformar entidades y comandos entre el mundo REST y el dominio.

Controlador: ApplicationController

Título: ApplicationController

Descripción: Controlador REST que maneja las operaciones CRUD y gestión de postulaciones de estudiantes a oportunidades de empresa.

Métodos

Método	Ruta	Descripción
getApplicationById	GET /api/v1/applications/{id}	Obtiene los detalles de una postulación específica por su ID

Método	Ruta	Descripción
getAllStudentApplications	GET /api/v1/applications/student/{studentId}	Obtiene todas las postulaciones de un estudiante
getAllOpportunityApplications	GET /api/v1/applications/opportunity/{opportunityId}	Obtiene todas las postulaciones a una oportunidad
submitApplication	POST /api/v1/applications	Crea una nueva postulación asociando estudiante, proyecto y oportunidad
acceptApplication	POST /api/v1/applications/{id}/accept	Cambia el estado de la postulación a Accepted
rejectApplication	POST /api/v1/applications/{id}/reject	Cambia el estado de la postulación a Rejected
deleteApplication	DELETE /api/v1/applications/{id}	Elimina una postulación del sistema

Dependencias

Dependencia	Descripción
ApplicationQueryService	Servicio para consultas y recuperación de datos de postulaciones
ApplicationCommandService	Servicio para ejecutar comandos de creación, aceptación, rechazo y eliminación de postulaciones
SubmitApplicationCommandFromResourceAssembler	Convierte recursos REST en comandos de creación de postulaciones
AcceptApplicationCommandFromResourceAssembler	Convierte recursos REST en comandos de aceptación de postulaciones
RejectApplicationCommandFromResourceAssembler	Convierte recursos REST en comandos de rechazo de postulaciones
DeleteApplicationCommandFromResourceAssembler	Convierte recursos REST en comandos de eliminación de postulaciones
ApplicationResourceFromEntityAssembler	Convierte entidades de postulación en recursos REST para la respuesta

2.6.4.3. Application Layer

Implementa los casos de uso a través de `ApplicationCommandServiceImpl` y `ApplicationQueryServiceImpl`, orquestando las operaciones sobre el dominio y gestionando el flujo entre la lógica de negocio y las dependencias externas.

Clase: ApplicationQueryServiceImpl

Título	ApplicationQueryServiceImpl
Descripción	Implementación del contrato de consultas definido en el dominio para postulaciones de estudiantes
Dependencias	
Dependencia	Descripción
ApplicationQueryService	Contrato definido en dominio
ApplicationRepository	Repositorio para acceso a datos de postulaciones

Clase: ApplicationCommandServiceImpl

Título	ApplicationCommandServiceImpl
Descripción	Implementación del contrato de comandos definido en el dominio para postulaciones de estudiantes
Dependencias	
Dependencia	Descripción
ApplicationCommandService	Contrato definido en dominio
ApplicationRepository	Repositorio para acceso a datos de postulaciones

2.6.4.4. Infrastructure Layer

Provee la persistencia mediante ApplicationRepository, con operaciones CRUD y consultas específicas por estudiante u oportunidad, conectando las entidades del dominio con la base de datos y gestionando la interacción con clases externas como Student, Project y Opportunity.

Clase: ApplicationRepository

Título	ApplicationRepository
Descripción	Interfaz de persistencia para operaciones CRUD y consultas específicas de postulaciones de estudiantes
Métodos	
Método	Descripción
save(ApplicationEntity)	Persiste una nueva postulación o actualiza una existente
deleteById(Long)	Elimina una postulación por su ID
findById(Long)	Obtiene una postulación por su ID

Método	Descripción
findAllByStudentId(Long)	Obtiene todas las postulaciones asociadas a un estudiante
findAllByOpportunityId(Long)	Obtiene todas las postulaciones a una oportunidad

Dependencias

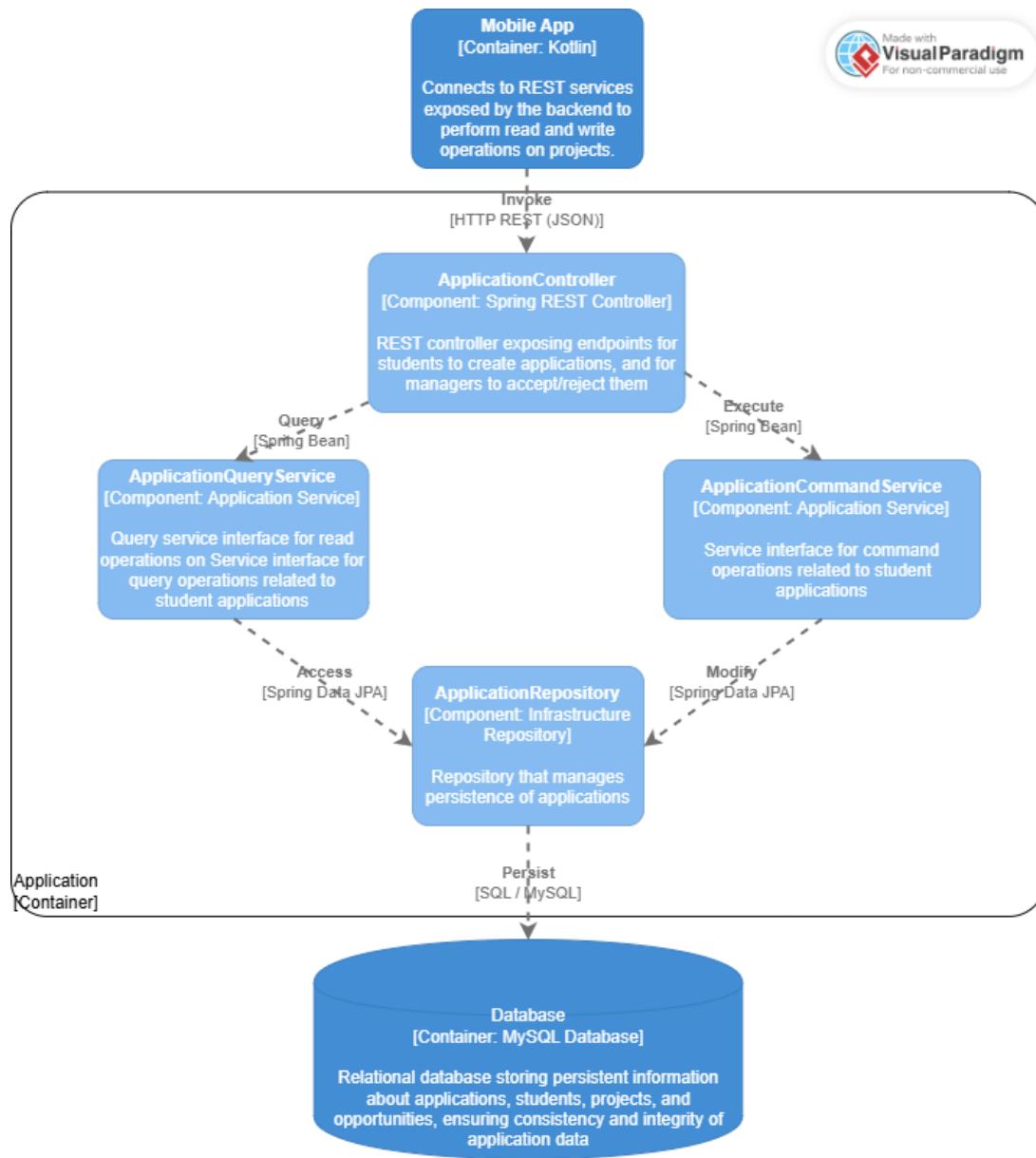
Dependencia	Propósito
Student	Clase que representa al estudiante en el sistema
Opportunity	Clase que representa la oportunidad en el sistema
Project	Clase que representa el proyecto asociado a la postulación
ApplicationEntity	Clase que representa la entidad de postulación en la base de datos

2.6.4.5. Bounded Context Software Architecture Component Level Diagrams

Este diagrama de componentes representa un sistema monolítico que gestiona postulaciones de estudiantes a oportunidades de empresas. Una aplicación Movil interactúa con el [ApplicationController](#).

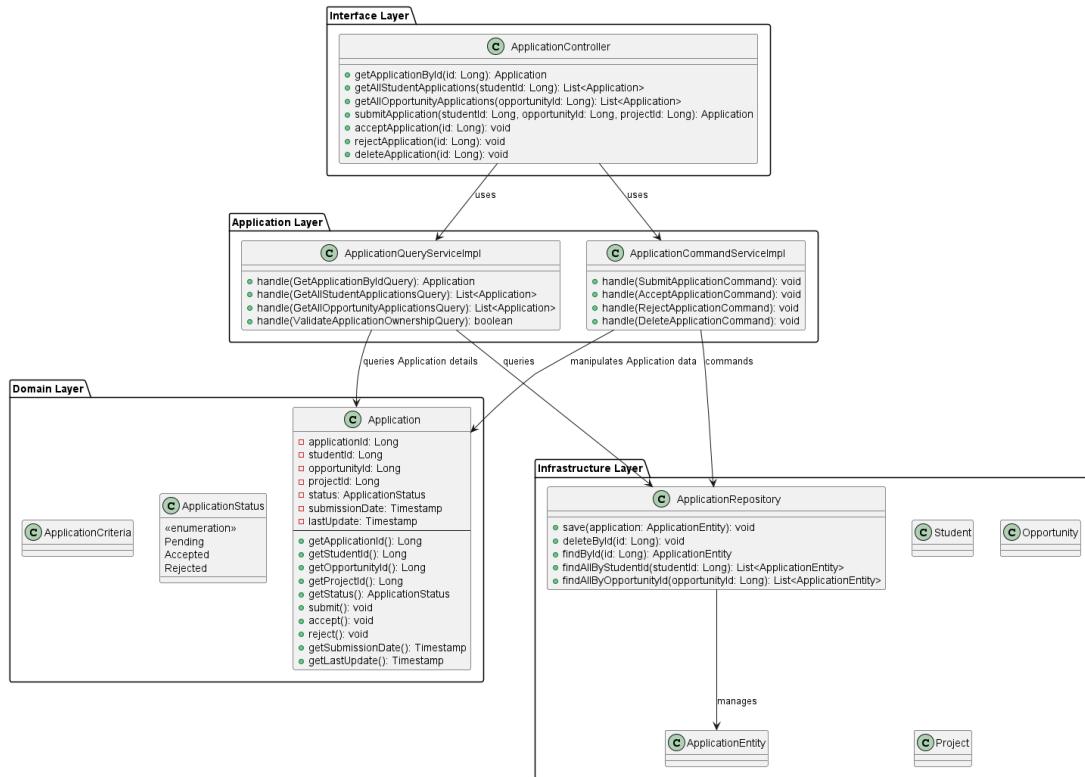
El controlador conecta con [ApplicationQueryService](#) para consultas y con [ApplicationCommandService](#) para registrar, aceptar, rechazar o eliminar postulaciones.

Ambos servicios utilizan el [ApplicationRepository](#), que gestiona la persistencia mediante JPA en una base de datos MySQL.



2.6.4.6. Bounded Context Software Architecture Code Level Diagrams

2.6.4.6.1. Bounded Context Domain Layer Class Diagrams



Bounded Context Class Diagram - Elaboración propia

2.6.4.6.2. Bounded Context Database Design Diagrams

