

PROYECTO FINAL DE INGENIERÍA

Plataforma Visual para Mejorar la Gestión
de Causas Legales de Abogados Juniors

Autor/es:

Gini, Rafael - LU: 1131661

Suárez, Micaela - LU: 1136043

Carrera:

Ingeniería en Informática

Tutor/es:

Yaps, David

Año:

2025

PROYECTO FINAL DE INGENIERÍA

PLATAFORMA VISUAL PARA MEJORAR LA GESTIÓN DE CAUSAS LEGALES DE ABOGADOS JUNIORS

Gini, Rafael – LU 1131661

Ingeniería en Informática

Suárez, Micaela – LU 1136043

Ingeniería en Informática

Tutor:

Yaps, David, UADE

10 de diciembre de 2025

UADE

UNIVERSIDAD ARGENTINA DE LA EMPRESA

FACULTAD DE INGENIERÍA Y CIENCIAS EXACTAS

Agradecimientos

En primer lugar, queremos expresar nuestro profundo agradecimiento a nuestras familias, por su apoyo constante, su paciencia y su fortaleza para acompañarnos tanto en los buenos como en los momentos más desafiantes de este proceso. Deseamos también agradecer a nuestros amigos, quienes nos acompañaron incondicionalmente a lo largo de estos años.

Asimismo, extendemos nuestro sincero reconocimiento a nuestro tutor David Pablo Yaps, por su constante predisposición, por ofrecernos siempre una perspectiva objetiva y por orientarnos con profesionalismo en cada instancia de mejora.

Nuestro especial agradecimiento al Dr. Marcelo Fabián Martínez, quien nos brindó una guía experta y una mirada profundamente técnica sobre el dominio específico abordado en esta tesis.

Agradecemos también al equipo docente de la asignatura, cuyos conocimientos, devoluciones y motivación fueron determinantes para el desarrollo de esta tesis. Su compromiso con la enseñanza y su energía positiva hicieron del proceso de trabajo una experiencia sumamente enriquecedora.

Finalmente, queremos agradecer al Dr. Marcos Vilaplana y a la Dra. Michelle Trouchot por ser parte fundamental de la investigación y a todas las personas que participaron ya sea respondiendo encuestas, entrevistas o colaborando de manera directa o indirecta.

Resumen

La presente tesis desarrolla LexGO, una plataforma web diseñada para mejorar la gestión y comprensión de causas legales por parte de abogados juniors, con foco en los ámbitos laboral de la Provincia de Buenos Aires y la Ciudad Autónoma de Buenos Aires. A partir de entrevistas, encuestas y análisis del estado del arte, se identificaron necesidades relacionadas con la fragmentación de información, la falta de visualizaciones claras y la dificultad para acceder a jurisprudencia relevante. La solución integra una línea de tiempo interactiva, un grafo relacional, carga y análisis automático de documentos, un buscador de jurisprudencia basado en RAG y un modelo de machine learning para clasificar etapas procesales. La plataforma constituye una innovación dentro del ecosistema LegalTech argentino, orientada específicamente a las necesidades operativas y formativas de abogados juniors.

Abstract

This thesis presents LexGO, a web platform aimed at improving case management and comprehension for junior lawyers, focusing on labor law in the Province of Buenos Aires and the Autonomous City of Buenos Aires. Interviews, surveys, and literature review revealed challenges such as dispersed information, limited visual tools, and difficulty accessing relevant case law. The system incorporates an interactive timeline, relational graph, automated document analysis, a RAG-based jurisprudence search engine, and a machine learning model for procedural stage classification. The platform represents a novel contribution to the Argentine LegalTech ecosystem, addressing the specific operational and educational needs of junior legal professionals.

Contenido

1. Introducción 9

1.1. Objetivos 9

1.1.1. Objetivo general 10

1.1.2. Objetivos específicos 10

1.2. Alcance 10

2. Antecedentes 12

2.1. Marco Teórico 12

2.1.1. Marco Teórico Jurídico 12

2.1.1.1. Estructura procesal de una causa judicial y fundamentos para su representación visual 14

2.1.2. Marco Teórico Tecnológico 16

2.1.2.1. Clean Architecture como marco estructural 16

2.1.2.2. Componentes de MVC: 17

2.1.2.3. Aplicabilidad al caso 17

2.1.2.4. Arquitectura de sistemas con IA: centrados vs. integrados 18

2.1.2.5. Diseño Centrado en el Usuario (DCU) como Marco Teórico 18

2.1.2.6. Principios Fundamentales del DCU 18

2.2. Estado del Arte 19

2.2.1. Sistemas de gestión integrados: el caso de Lex-Doctor 20

2.2.2. Fundamentos de Visual Law y Legal Design 21

2.2.3. Herramientas de visualización y cronogramas interactivos 23

2.2.4. Transformación digital en despachos de abogados a nivel global 24

2.2.5. Brechas y oportunidades 25

2.2.6. Diferenciales tecnológicos de la propuesta 25

2.2.7. Conclusión 27

2.3. Competencia.....	29
2.3.1. Océano Azul	29
2.3.1.1. Matriz ERIC (Eliminar, Reducir, Incrementar y Crear):.....	29
2.3.1.2. Diccionario de variables	30
2.3.1.3. Curva de Valor:	31
2.3.2. Conclusión.....	33
3. Descripción	35
3.1. User Research	35
3.1.1. Entrevistas cualitativas	35
3.1.1.1. Entrevista a la Dra. Michelle Trouchet	35
3.1.1.2. Entrevista al Dr. Marcos Vilaplana	36
3.1.2. Encuestas cuantitativas.....	38
3.1.2.1. Síntesis y decisiones de diseño.....	38
3.2. Requerimientos del sistema	39
3.2.1. Tabla de Requerimientos Funcionales	40
3.3. Diseño y Arquitectura	43
3.3.1. Diagrama de Contexto.....	44
3.3.2. Vistas funcionales.....	45
3.3.2.1. Vista funcional: Gestión de Causas.....	46
3.3.2.2. Vista funcional: Causa específica.....	48
3.3.2.3. Vista funcional: Dashboard	52
3.3.2.4. Vista funcional: Asistente	53
3.4. Modelo de Datos y Arquitectura del Sistema	54
3.4.1. Modelo de Datos	55
3.4.2. Expansión del Modelo de Datos.....	57

3.4.3.	Modelo de Datos Integral	58
3.5.	Arquitectura del Sistema	59
3.5.1.	Arquitectura de la Persistencia	61
3.5.2.	Backend de Servicios	63
3.5.3.	Frontend.....	65
3.5.4.	Proceso de Desarrollo.....	65
3.5.5.	Despliegue en Terraform (IaC)	66
3.5.6.	Uso de Ngrok para habilitar tráfico HTTPS.....	67
3.5.7.	Arquitectura Ideal.....	68
3.6.	Modelo de Inteligencia Artificial.....	69
3.6.1.	Tavily.....	69
3.6.2.	Justificación de la elección del modelo	69
3.6.3.	Funcionalidades implementadas.....	70
3.6.4.	Funcionamiento técnico	71
3.6.5.	Asistente de Jurisprudencia Basado en Recuperación Aumentada (RAG).....	72
3.6.6.	Modelo de Machine Learning	73
3.6.7.	Construcción del conjunto de datos.....	74
3.6.8.	Entrenamiento supervisado	74
3.6.9.	Resultados del modelo.....	75
3.6.10.	Decisiones metodológicas en la construcción del modelo	76
3.7.	Identidad de marca	78
3.8.	Modelo económico.....	79
3.8.1.	Metodología del análisis económico	79
3.8.2.	Etapas de desarrollo del producto	80
3.8.2.1.	Etapa I: Desarrollo del MVP (primer año).....	80

3.8.2.2.	Etapa II: Consolidación del producto (segundo año).....	81
3.8.2.3.	Etapa III: Expansión y escalabilidad (a partir del tercer año).....	81
3.8.3.	Costos fijos y costos variables.....	82
3.8.3.1.	Costos fijos.....	82
3.8.3.2.	Costos variables.....	83
3.8.4.	Escenarios de uso y estimación de costos.....	84
3.8.4.1.	Supuestos de consumo por usuario.....	84
3.8.4.2.	Estimación de costos.....	85
3.8.4.3.	Análisis comparativo entre costos e ingresos.....	86
3.8.5.	Análisis de escenarios financieros.....	87
3.8.5.1.	Escenario I: Crecimiento moderado (100 → 500 usuarios).....	87
3.8.5.2.	Escenario II: Crecimiento acelerado (100 → 1.000 usuarios).....	88
4.	Metodología de desarrollo.....	90
4.1.	Enfoque general: metodología ágil adaptada.....	90
4.2.	Herramientas de gestión y colaboración.....	91
4.3.	Metodología de desarrollo técnico.....	91
5.	Pruebas Realizadas.....	92
5.1.	Pruebas Unitarias.....	93
5.1.1.	Enfoque y técnica utilizada.....	93
5.1.2.	Aplicación de las pruebas unitarias.....	93
5.1.2.1.	Prueba 1: Creación exitosa de una causa judicial.....	94
5.1.2.2.	Prueba 2: Resumen automático de un documento almacenado en S3.....	94
5.1.2.3.	Prueba 3: Registro correcto de una interacción usuario–asistente.....	94
5.2.	Pruebas de Usabilidad.....	95
5.3.	Conclusión de las Pruebas.....	96

6. Discusión.....	96
7. Conclusión	99
8. Bibliografía	101
9. Anexos	106
Anexo A. Estructura de una causa judicial y su interpretación en la App	106
Anexo B: Entrevista al Dr. Marcos Vilaplana.....	107
Anexo C: Entrevista a la Dra. Michelle Trouchet.....	112
Anexo D. Datos de la Encuesta	118
Anexo E. Glosario de requerimientos	122
Anexo F. Vistas Funcional: Causa Específica.....	124
Anexo G. Vista Funcional: DASHBOARD.....	126
Anexo H. Arquitectura ideal	127
Anexo I. Componentes del Asistente de Jurisprudencia Basado en Recuperación Aumentada (RAG).....	128
Anexo J. Resultados del modelo de Machine Learning	129
Anexo K. Trello.....	130
Anexo L. Entrevista a Lucía Suárez sobre prueba de la aplicación	131
10. Índice de Figuras	134
11. Índice de Tablas	136

1. Introducción

En los primeros años del ejercicio profesional, los abogados recién egresados enfrentan una compleja combinación de desafíos: la brecha entre teoría y práctica, la falta de entrenamiento procesal real, y condiciones laborales precarias o informales. Esta situación incrementa significativamente la carga cognitiva y emocional de los jóvenes profesionales, quienes deben organizar expedientes reales, interpretar notificaciones, cumplir con plazos procesales y redactar documentos legales sin contar con herramientas formativas adecuadas. Estudios internacionales ya han documentado el alto impacto de la sobrecarga laboral en la salud mental de los abogados jóvenes, destacando que la organización del trabajo y la claridad en la gestión de causas resultan factores clave para reducir el estrés y mejorar la productividad (Sinsheimer & Herring, 2016).

Sin embargo, esta problemática se ha profundizado en el contexto actual de digitalización acelerada. A partir de la pandemia de 2020, herramientas como la firma digital, los tokens de acceso y la Mesa de Entradas Virtual (MEV) transformaron la manera en que los abogados interactúan con los expedientes. Aunque estos cambios han agilizado procesos judiciales, también han generado nuevas dificultades: mayor volumen de información, dispersión de archivos, múltiples sistemas paralelos (Word, Drive, Excel, Lex-Doctor, WhatsApp), y una creciente dificultad para tener una visión clara, global y actualizada del estado de cada causa (Delgado Martín, 2021).

La entrevista realizada a la abogada Michelle Trouchot (2025), especializada en derecho e inteligencia artificial, confirma este diagnóstico desde la experiencia real. A pesar de utilizar una herramienta profesional como Lex-Doctor, Michelle construye sus propios listados en Excel y Word para poder organizar sus expedientes de forma más clara, ordenando causas según su estado procesal (por ejemplo: "inicio", "apertura a prueba", "sentencia"). Este “doble trabajo” revela que las herramientas actuales no logran adaptarse del todo a las necesidades cognitivas y operativas de los abogados jóvenes, especialmente en estudios pequeños o quienes trabajan de forma independiente.

1.1. Objetivos

La presente sección detalla los objetivos que guían el desarrollo de la aplicación web propuesta, definiendo tanto la meta general del proyecto como los resultados específicos

que se buscan alcanzar. Estos objetivos permiten estructurar el trabajo técnico y metodológico, asegurando que el sistema responda a las necesidades reales de los usuarios del ámbito jurídico y que su diseño, implementación y validación se orienten a mejorar la comprensión y el seguimiento de procesos legales en la Provincia de Buenos Aires y en la Ciudad Autónoma de Buenos Aires. A partir de esta formulación, se establecen criterios medibles que permiten evaluar el desempeño de la herramienta y la efectividad de las soluciones desarrolladas.

1.1.1. Objetivo general

Desarrollar una aplicación web orientada a la asistencia de abogados en etapas iniciales de su carrera profesional para facilitar la comprensión y el seguimiento de procesos legales en la Provincia de Buenos Aires (incluyendo la Ciudad Autónoma de Buenos Aires), Argentina, durante el año 2025.

1.1.2. Objetivos específicos

1. Diseñar una interfaz web responsive orientada a dispositivos web y móviles que priorice la usabilidad y la visualización clara de la información legal, evaluada mediante pruebas de usabilidad donde al menos el 80% de los usuarios comprendan correctamente la estructura de un caso en menos de 7 minutos.
2. Implementar funcionalidades de organización de causas legales, una línea de tiempo interactiva y un grafo de eventos y actores. Verificar su correcto funcionamiento a través de pruebas internas y retroalimentación de al menos 5 usuarios del ámbito jurídico.
3. Aplicar principios de diseño centrado en el usuario realizando al menos dos rondas de pruebas de usabilidad con usuarios reales del ámbito jurídico, obteniendo mejoras iterativas en base a los resultados de dichas pruebas.
4. Desarrollar un prototipo inicial de modelo predictivo basado en técnicas de machine learning que sugiera estructuras básicas de causas legales y alerte sobre posibles errores u omisiones, evaluando su desempeño en un entorno de prueba controlado.
5. Aplicar principios de diseño centrado en el usuario para validar la experiencia de uso mediante pruebas con usuarios reales del ámbito jurídico.

1.2. Alcance

El trabajo abarca el diseño, desarrollo y despliegue de un prototipo funcional de una aplicación web responsive mobile-first, orientada a la asistencia de abogados en etapas iniciales de su carrera profesional en la Provincia de Buenos Aires durante el año 2025.

El alcance del trabajo final incluye:

- Creación de proyectos que representen expedientes legales individuales.
- Carga y gestión de eventos clave asociados a cada proyecto.
- Carga de archivos y documentos en las causas.
- Creación automática de causas a partir de documentos PDF mediante un proceso de extracción y clasificación de contenido, complementado por IA para generar una primera estructura cronológica y sugerencias iniciales.
- Visualización cronológica de los eventos mediante una línea de tiempo interactiva.
- Representación gráfica de actores, hechos y documentos mediante un grafo relacional.
- Validación de la experiencia de usuario a través de pruebas controladas con al menos 5 estudiantes de abogacía o abogados juniors.
- Buscador de jurisprudencia vectorizada + asistente IA para el abogado.
- Generación de estructura inicial de una causa con IA mediante la carga de un primer documento.
- Detector y verificador de sugerencias, omisiones u posibles errores con IA.
- Gestor de tareas asociadas a una causa.
- Gestor de trazabilidad de movimientos dentro de una causa.

La solución es accesible desde navegadores web en dispositivos móviles y de escritorio, priorizando una experiencia fluida y adaptable.

Desde el punto de vista técnico, el desarrollo utilizará:

- Frontend: React.js + Next.js.
- Backend: Python con API RESTful.
- Base de datos: PostgreSQL.

- Infraestructura Cloud: AWS Free Tier (utilizando instancias EC2, almacenamiento S3 y base de datos RDS).

2. Antecedentes

La presente sección reúne los principales desarrollos, sistemas y enfoques previamente utilizados en la digitalización de procesos jurídicos y en la asistencia tecnológica para profesionales del Derecho. Se revisan tanto herramientas tradicionales de gestión de expedientes como soluciones recientes basadas en inteligencia artificial, visualización de información y automatización documental. Este recorrido permite identificar las limitaciones existentes en las plataformas actuales, especialmente en lo referido a usabilidad, trazabilidad y apoyo a profesionales en etapas iniciales de su carrera, y fundamenta la necesidad de una propuesta que integre modelos predictivos, interfaces intuitivas y mecanismos de representación estructurada de causas legales adaptados al contexto argentino.

2.1. Marco Teórico

El Marco Teórico expone los conceptos, enfoques y modelos analíticos que sustentan el desarrollo de la aplicación, incorporando nociones provenientes del diseño centrado en el usuario, la ingeniería de software, la visualización de información jurídica y las técnicas de inteligencia artificial aplicadas al procesamiento documental y la recuperación aumentada. Asimismo, se abordan fundamentos sobre procesos legales, lo que permite contextualizar la estructura del sistema y orientar sus funcionalidades a las prácticas reales del ámbito jurídico local. Este marco proporciona la base conceptual necesaria para comprender las decisiones metodológicas, tecnológicas y de diseño adoptadas a lo largo del proyecto.

2.1.1. Marco Teórico Jurídico

El pensamiento jurídico como proceso lógico estructurado se respalda en el proyecto con la teoría sobre cómo se enseña el razonamiento jurídico: identificación de hechos, normas aplicables, jurisprudencia y resolución. Además, las herramientas que permiten visualizar eventos, relaciones y documentación, pueden apoyar este proceso lógico y ayudan como marco para orientar el diseño visual. Dicho razonamiento constituye una actividad compleja que implica la identificación de hechos relevantes, la interpretación de normas aplicables y la aplicación de estas normas a los hechos del caso concreto. Este proceso,

conocido como subsunción, es fundamental en la práctica legal y requiere una comprensión profunda tanto de los hechos como del marco normativo (Magistratura del Perú, 2000).

Según la Academia de la Magistratura del Perú (2000), el razonamiento jurídico se vincula estrechamente con el tratamiento de los hechos del caso y la interpretación de las normas aplicables. Este proceso no es automático; requiere un trabajo inteligente de comprensión e interpretación normativa, así como una adecuada calificación jurídica de los hechos para dar solución a los problemas jurídicos del caso.

Se adopta dicho marco como base conceptual para construir una herramienta digital que favorezca la estructuración del pensamiento jurídico a través de visualizaciones interactivas. La aplicación está diseñada representar causas judiciales mediante líneas de tiempo interactivas y grafos relacionales, permitiendo al usuario:

- Descomponer una causa en eventos clave cronológicamente organizados, lo cual ayuda a identificar los hechos relevantes.
- Relacionar estos eventos con actores, documentos y normas, facilitando la interpretación de la situación legal.
- Visualizar todo el proceso como un mapa navegable, que favorece la comprensión global y estructurada del caso.

De esta manera, la herramienta se posiciona como un apoyo cognitivo al razonamiento jurídico, orientado especialmente a estudiantes avanzados y abogados juniors, quienes enfrentan por primera vez la complejidad del ejercicio profesional.

En la enseñanza del derecho, se ha destacado la importancia de métodos que fomenten el desarrollo del razonamiento jurídico práctico. Por ejemplo, el uso de esquemas y ensayos como herramientas pedagógicas permite a los estudiantes estructurar lógicamente las temáticas tratadas, facilitando el orden mental de las ideas y la comprensión de los criterios específicos de cada caso. En este contexto, las herramientas tecnológicas que permiten visualizar eventos, relaciones y documentación pueden apoyar significativamente este proceso lógico. La representación visual de los elementos de un caso jurídico facilita la identificación de patrones, la comprensión de la secuencia de eventos y la relación entre los diferentes actores y normas involucradas. Estas herramientas pueden servir como apoyo en la formación de abogados, especialmente en las etapas iniciales de su carrera profesional, al proporcionar una estructura clara y accesible para el análisis de casos complejos (Sancho Elizondo, 2014).

2.1.1.1. Estructura procesal de una causa judicial y fundamentos para su representación visual

Una causa judicial no es un documento único, sino una unidad compleja de información compuesta por hechos, actos procesales, documentos y decisiones judiciales, todos distribuidos a lo largo del tiempo y en constante transformación. En el sistema argentino, cada causa pasa por una serie de etapas: inicio, impulso, producción de prueba, resolución y ejecución, entre otras, dependiendo del fuero y tipo de juicio (Colegio de Abogados de San Isidro, 2015).

Sin embargo, no existe una forma única ni estandarizada de visualizar o resumir el estado de una causa, lo que genera barreras para su comprensión, especialmente entre profesionales con poca experiencia. Desde una perspectiva técnica, una causa puede conceptualizarse como una estructura secuencial y lógica de eventos, cada uno con un estado asociado (pendiente, cumplido, vencido, etc.), y relaciones entre actores, documentos y normas. La jurisprudencia, por su parte, actúa como un conjunto de antecedentes relevantes que pueden ser vinculados a esos eventos, o incluso sugerir caminos alternativos según el perfil del expediente (Magistratura del Perú, 2000).

En la práctica, los sistemas existentes como la MEV o Lex-Doctor presentan la causa como una lista de escritos y fechas, sin proporcionar una capa cognitiva que permita al usuario interpretar el estado del proceso en términos comprensibles. Esto fue señalado en entrevistas realizadas a profesionales del derecho, como Michelle Trouchet, quien afirmó:

"Hoy una causa se ve como una lista. Pero estaría bueno que te dé un resumen más visual, más parecido a cómo pensás la causa: qué está hecho, qué falta, qué venció. Como una historia" (Entrevista propia, 2025).

El modelo propuesto se basa en esa necesidad concreta, utilizando elementos de conocimiento jurídico y de diseño centrado en el usuario para generar representaciones interactivas de causas legales mediante interfaces modernas e intuitivas.

La visualización por la línea de tiempo o por estado de eventos no solo simplifica la lectura del expediente, sino que también favorece una mejor comprensión de su situación actual y de las acciones que deben emprenderse.

Además, la incorporación de jurisprudencia vinculada a eventos específicos (por ejemplo, mostrar fallos similares en etapas clave) permite contextualizar decisiones y anticipar

riesgos, integrando así el saber jurídico previo al razonamiento activo del usuario (Sancho Elizondo, 2014).

La estructura de una causa judicial se compone de actos procesales organizados en torno a un conflicto jurídico concreto. Estos actos, que incluyen presentaciones, audiencias, resoluciones y notificaciones, se encuadran en etapas procesales secuenciales que permiten el desarrollo ordenado del expediente judicial. Cada causa puede ser vista como una unidad sistémica en constante evolución, donde la acumulación de documentos, plazos y resoluciones configuran su estado actual (Fix-Zamudio citado por Mac-Gregor, 2012). La estructura completa de una causa judicial y su interpretación en la App puede observarse en Anexo A.

Partiendo de esta comprensión estructural de las causas judiciales, resulta indispensable considerar también el punto de vista del usuario final, en especial de aquellos con poca experiencia profesional. En este sentido, en el diseño de la herramienta se contempla no sólo la lógica jurídica del expediente, sino también las necesidades cognitivas, organizativas y prácticas de quienes interactúan con estas estructuras complejas a diario.

A continuación, se presentan las principales necesidades que surgieron de entrevistas, observación directa y análisis funcional, vinculadas al uso efectivo de la aplicación propuesta.

TABLA I: Principales necesidades de los abogados juniors. (Entrevista propia a Michelle Trouchet, Anexo C)

Necesidad	Qué implica	Interpretación en la App
Saber en qué etapa está la causa	Comprender el estado del proceso	Mostrar el estado actual con íconos o resúmenes
Saber qué falta por hacer	Identificar próximos pasos o vencimientos	Alertas o eventos marcados como “pendientes”
Resumir todo para un superior o cliente	Sintetizar información sin leer todo el expediente	Vista general de línea de tiempo + resumen automático
Relacionar hechos con fallos anteriores	Argumentar mejor o anticipar decisiones	Sugerencias de jurisprudencia en cada etapa

2.1.2. Marco Teórico Tecnológico

La presente sección establece los pilares técnicos que guiaron las decisiones de arquitectura, diseño e implementación del sistema, abarcando desde principios de ingeniería de software, como Clean Architecture, modularidad y desacoplamiento, hasta herramientas específicas empleadas para el procesamiento de documentos, almacenamiento en la nube e integración de modelos de inteligencia artificial. Asimismo, se analizan tecnologías clave que en conjunto posibilitan la construcción de un entorno funcional, escalable y orientado a la recuperación aumentada de información jurídica. Este marco teórico no solo contextualiza las elecciones tecnológicas adoptadas, sino que también permite comprender su pertinencia, limitaciones y aporte dentro del dominio jurídico-computacional en el que se inscribe el proyecto.

2.1.2.1. Clean Architecture como marco estructural

Se adopta como marco teórico estructural el enfoque de Clean Architecture, propuesto por Robert C. Martin (2017) para construir una solución tecnológica robusta, mantenible y escalable. Esta arquitectura organiza el sistema en capas concéntricas donde las dependencias siempre apuntan hacia el núcleo, permitiendo desacoplar la lógica de negocio de los detalles de implementación como frameworks, bases de datos o herramientas externas.

Las capas principales son:

- **Entidad / Dominio:** modelos conceptuales puros como Causa, Evento, Estado, Actor.
- **Casos de uso:** procesos como crear causa, visualizar línea de tiempo, asociar documento.
- **Adaptadores de interfaz:** controladores HTTP, API REST, interfaces visuales.
- **Frameworks y drivers:** React (UI), PostgreSQL (BD), AWS (infraestructura), y módulos IA.

Este enfoque garantiza que la lógica central del sistema permanezca estable, incluso si cambian las herramientas, tecnologías o servicios conectados. Además, favorece una estructura modular, donde nuevas funcionalidades (como IA) pueden integrarse como módulos aislados sin romper el resto de la aplicación.

Además del enfoque estructural propuesto por Clean Architecture, se considera también el patrón clásico Model-View-Controller (MVC) como marco teórico de referencia para aplicaciones web. MVC es uno de los modelos arquitectónicos más difundidos en el desarrollo de software y ha servido como base conceptual para muchos frameworks modernos (Django, Ruby on Rails, Laravel, etc.) (MDN Web Docs, 2023).

2.1.2.2. Componentes de MVC:

Model (Modelo): Representa los datos y las reglas de negocio. Ejemplo: una clase 'Causa-Judicial' que define sus atributos y lógica asociada (validaciones, relaciones, etc.).

View (Vista): Encargada de presentar la información al usuario. Una interfaz que muestra la línea de tiempo de una causa.

Controller (Controlador): Intermediario que coordina las acciones entre el modelo y la vista. Ejemplo: recibe un request para "crear causa", lo procesa, actualiza el modelo y devuelve una vista de confirmación.

2.1.2.3. Aplicabilidad al caso

Si bien MVC ofrece una estructura clara y sencilla para separar responsabilidades, su uso en sistemas con lógica compleja o múltiples canales de entrada/salida puede derivar en una excesiva concentración de responsabilidades en los controladores o en un acoplamiento indeseado entre capas.

Por esta razón, se considera MVC como un marco teórico útil para visualizar la separación básica de componentes, especialmente en la interfaz de usuario (frontend), pero se opta por Clean Architecture para el diseño integral del sistema, dado que:

- Clean Architecture extiende la separación propuesta por MVC hacia capas más profundas del dominio.
- Permite desacoplar el sistema de tecnologías concretas (como frameworks o bases de datos).
- Facilita la integración de componentes externos, como módulos de IA, de forma estructurada y estable.

En este sentido, MVC se utiliza como marco de referencia dentro del nivel de presentación de la aplicación (por ejemplo, en el diseño del frontend), pero no como estructura central del sistema.

2.1.2.4. Arquitectura de sistemas con IA: centrados vs. integrados

En el campo de la ingeniería de software para sistemas inteligentes, se distinguen dos modelos principales (Quantic School of Business & Technology, 2025):

Sistemas centrados en IA:

- El modelo de IA es el núcleo del sistema.
- Toda la lógica del negocio gira en torno a la inferencia, clasificación o generación del modelo.
- Ejemplo: asistentes conversacionales autónomos, análisis de texto sin estructuras previas.

Sistemas tradicionales con módulos IA integrados

- La lógica central es clásica (CRUD, reglas, flujos), y la IA funciona como componente auxiliar (IBM, s.f.)
- La IA mejora tareas específicas: OCR, extracción de datos, recomendaciones, predicciones.

Este segundo enfoque es el más adoptado en aplicaciones profesionales, según estudios recientes sobre arquitectura de IA aplicada:

"Las aplicaciones que incorporan IA como componente modular permiten validar el sistema sin IA, integrar modelos gradualmente y mantener la separación de preocupaciones" (Amershi et al., 2019).

2.1.2.5. Diseño Centrado en el Usuario (DCU) como Marco Teórico

El **Diseño Centrado en el Usuario (DCU)** es una filosofía y metodología de diseño que sitúa a las personas en el núcleo del proceso de desarrollo, garantizando que los sistemas creados respondan a sus **necesidades, expectativas y limitaciones reales**. Este enfoque busca optimizar la **usabilidad**, la **eficiencia** y la **satisfacción** en la interacción entre usuarios y sistemas interactivos. Su aplicación sistemática permite desarrollar productos, servicios y experiencias que resulten **funcionales, intuitivos y significativos** para los usuarios finales (ISO, 2019).

2.1.2.6. Principios Fundamentales del DCU

En base a lo definido en “Los Principios Clave del Diseño Centrado en el Usuario” (ISO, 2019), se estipulan los siguientes principios fundamentales:

- **Comprensión profunda de los usuarios:** Implica identificar quiénes son los usuarios finales, sus características demográficas, comportamientos y contextos. Se utilizan herramientas como entrevistas, encuestas y estudios etnográficos para comprender sus motivaciones, frustraciones y expectativas.
- **Diseño basado en la experiencia real del usuario:** Prioriza la funcionalidad y la facilidad de uso sobre la estética. Incluye simplificar la interfaz, crear soluciones accesibles y realizar iteraciones constantes basadas en el comportamiento y la retroalimentación de los usuarios reales.
- **Pruebas de usabilidad y retroalimentación continua:** El DCU es un proceso iterativo que depende de la retroalimentación constante de los usuarios. Esto implica crear prototipos, realizar pruebas de usabilidad y ajustar el diseño en base a los hallazgos obtenidos.
- **Diseño contextual:** Considera el entorno físico o digital en el que se utilizará el producto, las limitaciones técnicas y los factores culturales que pueden influir en la percepción y usabilidad.
- **Empatía como motor del diseño:** Reconoce que los usuarios son diversos y complejos. Diseñar con empatía significa escuchar activamente, evitar sesgos de diseño y crear soluciones que se ajusten a diferentes necesidades y circunstancias.

2.2. Estado del Arte

La última década ha transformado radicalmente la forma en que los profesionales del Derecho gestionan sus procesos y administran la información. La digitalización de expedientes, la proliferación de herramientas de visualización y el auge de la Inteligencia Artificial (IA) han marcado un punto de inflexión: los registros manuales y las agendas dispersas han dado paso a plataformas integradas que centralizan la documentación, aseguran

el seguimiento de plazos procesales y ofrecen representaciones gráficas intuitivas (SAKO & PARNHAM, 2021).

En esta sección se analiza la literatura más relevante relacionada con la visualización jurídica, el diseño legal, las herramientas computacionales para el procesamiento de textos legales, los sistemas de gestión de causas judiciales y la transformación tecnológica en despachos de abogados. Además, se presta especial atención a las aproximaciones para representar la dimensión temporal de los procesos, es decir, líneas de tiempo o cronogramas que permitan a los abogados gestionar de manera clara los hitos y plazos de cada causa.

2.2.1. Sistemas de gestión integrados: el caso de Lex-Doctor

Durante más de tres décadas, Lex-Doctor se ha consolidado como “*el software de gestión jurídica de escritorio más utilizado en Argentina*” (Entrevista propia a Michelle Troughot, 2025). Diseñado para centralizar y optimizar las tareas administrativas de un despacho legal, permite administrar expedientes, mantener un calendario de vencimientos procesales, generar escritos estandarizados, gestionar contactos y controlar la facturación, entre otras funcionalidades. Su adopción extendida responde a la necesidad de profesionalizar la administración interna de los despachos, sustituyendo registros manuales o soluciones fragmentadas por una plataforma integrada. Si bien carece de una capa analítica avanzada o de vistas gráficas interactivas, ofrece una base sólida para el manejo de la rutina procesal y documental de los abogados, estandarizando procesos como la confección de escritos formalizados, el seguimiento de plazos y la centralización de datos de cada caso, lo que redundará en una mejora significativa de la eficiencia operativa (LEX-DOCTOR, sin fecha).

En cuanto a su implementación y adopción, existen diversos documentos y experiencias que reflejan el uso de Lex-Doctor en distintos ámbitos judiciales y académicos. En el Informe Anual del Colegio de Abogados de San Isidro (2008), se detalla que la Suprema Corte de Justicia de la Provincia de Buenos Aires llevó a cabo, en junio de 2007, una prueba piloto para evaluar el funcionamiento de Lex-Doctor 8.0 en varios órganos jurisdiccionales (Juzgado Civil y Comercial, Tribunales de Familia y Cámaras de Apelaciones). Antes de 2004, muchos juzgados aún operaban en entornos MS-DOS, por lo que este proyecto buscaba modernizar la gestión procesal mediante la adopción de Lex-Doctor bajo Windows. En el ámbito académico, la Facultad de Derecho de la Universidad de Buenos Aires ofrece desde hace años un “Curso Lex-Doctor Nivel 1” destinado a auxiliares de estudio jurídico, secretarios

y profesionales que necesiten dominar la estructura de datos, los módulos de agenda y la generación de escritos automatizados del sistema (UNIVERSIDAD DE BUENOS AIRES, 2024). De igual manera, la Universidad Católica Argentina propuso un curso de formación continua titulado “Informática Jurídica: Herramientas para el manejo del sistema Lex-Doctor”, orientado a la capacitación de auxiliares de estudios jurídicos y funcionarios judiciales, lo que demuestra la prevalencia de esta herramienta como estándar de facto en múltiples ámbitos legales del país (UNIVERSIDAD CATÓLICA ARGENTINA, 2018).

Por su parte, en la provincia de San Juan, el Poder Judicial implementó “Lex-Doctor-Online” como servidor central para la consulta de expedientes en tiempo real, integrando datos de diversas bases locales y permitiendo la publicación de expedientes por Internet; esta iniciativa, documentada en informes internos del Centro de Información Judicial, muestra cómo Lex-Doctor se utilizó para disminuir el flujo de personas en los edificios judiciales y ofrecer acceso remoto a la información procesal (CENTRO DE INFORMACIÓN JUDICIAL, 2019). A pesar de ello, no se han encontrado estudios académicos sistemáticos ni publicaciones arbitradas que evalúen su impacto en términos de productividad, satisfacción del usuario o reducción de errores procesales. La mayoría de las referencias existentes son informes institucionales, memorias de cursos y experiencias piloto que descriptivamente posicionan a Lex-Doctor como la solución predominante en el mercado, sin profundizar en análisis cuantitativos sobre su eficacia o retorno de la inversión.

En cuanto al precio, Lex-Doctor suele cotizarse desde su página web, o mediante un proveedor. Datos de mercado obtenidos de su página oficial indican que la primera compra de Lex-Doctor LEX-11 ronda los ARS 432763, sumado a una licencia anual de ARS 173105, dando un total de ARS 575568, valor reportado en marzo de 2025 para un solo puesto de trabajo (LEX-DOCTOR, Sin Fecha). A estos costos se suman gastos adicionales por instalación y mantenimiento, y la contratación de personal técnico para configurar servidores y redes locales. Estos montos pueden representar una barrera de acceso para estudiantes recién egresados o para despachos de pequeño porte, que enfrentan limitaciones financieras al momento de adquirir y operar la plataforma.

2.2.2. Fundamentos de Visual Law y Legal Design

El concepto de Visual Law y su estrecha relación con el Legal Design han cobrado relevancia en los últimos años como formas de hacer más accesible y comprensible la

información jurídica. En este contexto, el Legal Design surge como una disciplina que aplica principios del diseño centrado en el usuario del ámbito jurídico, con el objetivo de reimaginar documentos, procesos y servicios legales para que resulten más claros, eficientes y útiles. Se fundamenta en metodologías de investigación etnográfica, prototipado y pruebas de usabilidad, buscando entender las necesidades reales de quienes interactúan con el Derecho. Por su parte, la Visual Law es la manifestación práctica de este enfoque: emplea recursos gráficos tales como diagramas de flujo, mapas de procesos, infografías, líneas de tiempo y otros esquemas visuales, para traducir textos legales complejos en representaciones intuitivas. Mientras el Legal Design define el “cómo” (la metodología de diseño y validación con usuarios), la Visual Law muestra el “qué” (los *deliverables* gráficos concretos) que permiten a abogados y a usuarios finales identificar de un vistazo sus derechos, obligaciones, plazos procesales y rutas de actuación, reduciendo la carga cognitiva y mejorando la transparencia del entramado normativo y procesal (BRUNSCHWIG, 2021).

A nivel internacional, los primeros trabajos en Visual Law destacaron la necesidad de diagramas que incorporen explícitamente la dimensión temporal. El artículo “Diagrammatics and the Proactive Visualization of Legal Information” introdujo los diagramas de flujo y mapas de fluidez para ilustrar la secuencia cronológica de trámites legislativos y contractuales, estableciendo las bases teóricas de lo que más tarde se denominaría “cronograma legal” (Murray, 2021). En años recientes, Brunshwig (2021) reivindicó el uso de cronogramas interactivos que permitan visualizar audiencias, vencimientos de plazos y etapas probatorias en un flujo continuo, capaz de adaptarse a usuarios con distinto grado de experiencia jurídica. Sin embargo, la doctora Eliza Mik advirtió que la simplificación excesiva de procesos complejos puede resultar engañosa y conducir a omisiones críticas, por lo que subrayó la importancia de preservar la fidelidad jurídica al diseñar estos esquemas (Mik, 2019).

En este contexto, McLachlan y Webley, en su artículo “Visualisation of Law and Legal Process: An Opportunity Missed”, realizan una revisión exhaustiva de la literatura jurídica contemporánea para identificar cómo se emplean diagramas de flujo y otros esquemas orientados a procesos en la enseñanza y práctica del Derecho. A través de un análisis de contenido, clasifican once tipos arquetípicos de diagramas y cuantifican su aparición por año, editorial y dominio legal, constatando que en ningún año se publicaron más de diez artículos que incorporan visualizaciones de este tipo. Su estudio revela que el diagrama de flujo conceptual es el formato más habitual y que el Lenguaje Unificado de Modelado (UML) es la

aproximación más frecuente para representar estructuras procesales. McLachlan y Webley concluyen planteando preguntas esenciales para educadores y profesionales: ¿cómo puede la inclusión sistemática de diagramas conceptuales, en la formación y la práctica, mejorar el acceso a la Justicia y la comprensión ciudadana de marcos legales cada vez más complejos? (McLachlan y Webley, 2021) Esta aportación refuerza los fundamentos de la Visual Law al mostrar, con datos empíricos, tanto su potencial como la escasa adopción actual en el ámbito jurídico.

2.2.3. Herramientas de visualización y cronogramas interactivos

En el terreno tecnológico, el proyecto LawMaps (McLachlan et al., 2023) empleó UML para exponer la estructura implícita de la legislación y las dependencias temporales entre artículos y disposiciones, demostrando que los nodos podrían representar etapas procesales (presentación de escritos, audiencias preliminares, sentencia) y las flechas los plazos obligatorios entre ellas. Por su parte, LegalViz (Onami et al., 2025), presentó un esfuerzo centrado en la generación automática de diagramas a partir de texto jurídico, avanzó un paso más al crear un dataset multilingüe de 7.010 pares “documento–visualización” y evaluar modelos de lenguaje para generar diagramas legales automáticos, incluidos cronogramas que ubican fechas de presentación de documentos y devoluciones de copias; sin embargo, los autores reconocieron que garantizar la coherencia de las fechas requería reglas heurísticas que agruparan eventos cercanos en el tiempo. En paralelo, LegalVis (Resck et al., 2022), un trabajo orientado a la exploración y explicación de citas de precedentes dentro de documentos jurídicos, ofreció cronologías de jurisprudencia que muestran la evolución histórica de precedentes relacionados con un caso dado, ilustrando cómo la visualización temporal enriquece el análisis doctrinal y facilita la justificación de decisiones judiciales.

La incorporación de la Inteligencia Artificial ha potenciado estas iniciativas. Se ha demostrado que la combinación de visión por computadora y Procesamiento de Lenguaje Natural (NLP) permite extraer “pistas visuales” (layout, estilo tipográfico, posicionamiento) de contratos extensos, mejorando la segmentación de cláusulas y habilitando líneas de tiempo contractuales que identifiquen firmas, fechas de cumplimiento y plazos de entrega (Hegel et al., 2021).

Por su parte, Górski, Ramakrishna y Nowosielski (2021) adaptaron la técnica Grad-CAM para “visualizar” en el texto legal las regiones decisivas que los modelos de NLP

consideran relevantes para clasificar párrafos como fechas de audiencia o plazos procesales, proporcionando un mecanismo de explicabilidad fundamental para validar la extracción automática de eventos temporales antes de incorporarlos a un cronograma (Górski et al., 2021).

En cuanto a otras implementaciones, Rodgers, Armour y Sako describen chatbots alimentados por bases de datos de normativas y plazos que sugieren acciones oportunas, integrándose con cronogramas interactivos para recordar audiencias próximas. En el plano predictivo, estudios fuera de este corpus utilizan machine learning para estimar probabilidades de éxito litigioso, pero requieren explicabilidad para generar confianza en el usuario (Rodgers et al., 2023).

2.2.4. Transformación digital en despachos de abogados a nivel global

La transformación digital en los despachos de abogados no se limita a la adopción de software para gestionar expedientes, sino que ha evolucionado hacia la creación de ecosistemas tecnológicos que integran asistentes conversacionales, dashboard's interactivos y analítica avanzada. Los chatbots conversacionales, por ejemplo, ofrecen a clientes y equipo interno la posibilidad de consultar procesos, plazos y requerimientos normativos en lenguaje natural, conectándose a bases de datos actualizadas de legislación y jurisprudencia. De este modo, se automatizan tareas repetitivas como la notificación de vencimientos o la generación de borradores de escritos, liberando tiempo para actividades de mayor valor añadido (Rodgers et al., 2023).

Paralelamente, los dashboard's interactivos se han convertido en la pieza central de los espacios de trabajo digital. Estas interfaces muestran en tiempo real indicadores clave de rendimiento (KPI) de cada causa (estado del expediente, etapas completadas, plazos próximos, carga de trabajo por abogado) y permiten filtrar por cliente, área de práctica o nivel de urgencia. Al incorporar visualizaciones de tipo Gantt o líneas de tiempo, los despachos logran una perspectiva global de sus proyectos legales, identificando cuellos de botella y anticipando riesgos de incumplimiento (McLachlan & Webley, 2021).

En el artículo "How Technology Is (or Is Not) Transforming Law Firms" se analizaron más de 50 despachos de EE. UU. y Reino Unido que habían integrado estas herramientas, y concluyeron que aquellos que confiaron plenamente en las interfaces gráficas y en los asistentes virtuales redujeron en un 20% la pérdida de plazos procesales durante su primer año de uso. Sin embargo, se subraya que el éxito de la transformación digital depende

tanto de la tecnología como del cambio cultural: es imprescindible capacitar a los abogados en el uso de estas plataformas y fomentar la colaboración entre perfiles tradicionales (abogados y secretarios) y los nuevos “legal technologists” responsables de configurar y optimizar los sistemas. Asimismo, la adopción de modelos de IA explicables (XAI) en los despachos ha permitido justificar las recomendaciones del sistema, por ejemplo, por qué un chatbot sugiere cierta acción o cómo un dashboard calcula un riesgo de incumplimiento, generando confianza en los usuarios. Estas explicaciones ayudan a que el equipo comprenda los criterios de priorización y corrija posibles errores en la interpretación de datos, como fechas mal registradas o audiencias reprogramadas (Rodgers et al., 2023).

2.2.5. Brechas y oportunidades

A pesar de estos avances, persisten carencias relevantes: no existen modelos híbridos consolidados que integren reglas jurídicas explícitas con IA para validar cada evento temporal; las interfaces colaborativas, capaces de permitir anotaciones y reprogramaciones automáticas, siguen siendo escasas; la adaptación de soluciones a contextos con datos no estructurados, propios de muchos despachos y tribunales latinoamericanos, carece de propuestas robustas; y falta investigación empírica a mediano plazo que cuantifique el impacto de estas tecnologías en la práctica diaria (Aguerre et al., 2021).

La plataforma propuesta en esta tesis, que combinará un pipeline híbrido de extracción de eventos temporales, un dashboard colaborativo con línea de tiempo interactiva y búsqueda de jurisprudencia inteligente, apunta a cerrar dichas brechas y a optimizar la gestión de causas, especialmente para abogados junior que enfrentan los retos de un entorno cada vez más complejo.

2.2.6. Diferenciales tecnológicos de la propuesta

La solución desarrollada presenta un conjunto de diferenciales tecnológicos que la distinguen sustancialmente de otras herramientas actualmente disponibles en el mercado jurídico. Su arquitectura modular, la integración de inteligencia artificial y las funcionalidades de visualización interactiva permiten mejorar la comprensión, seguimiento y organización de causas legales complejas.

A diferencia de sistemas basados en IA genérica (como asistentes conversacionales amplios), esta aplicación no depende exclusivamente de modelos de lenguaje

natural. En cambio, implementa módulos especializados de IA -entrenados y validados fuera del entorno crítico de uso- que aportan valor añadido sin desplazar el razonamiento del usuario. Cada sugerencia automatizada (como el orden de eventos o la vinculación de jurisprudencia) es acompañada de una explicación clara en lenguaje natural, incluyendo el criterio jurídico o documento que motivó dicha recomendación. Este enfoque contribuye a la transparencia y a la confianza en la tecnología, aspectos especialmente sensibles en entornos legales (Aguerre et al., 2021).

Asimismo, la incorporación de visualizaciones estructuradas (línea de tiempo interactiva y grafo relacional) posibilita una representación más clara del estado procesal de cada expediente, potenciando el razonamiento jurídico estructurado. Estas representaciones funcionan como soporte cognitivo, permitiendo que los usuarios comprendan la progresión del caso de manera visual y lógica (McLachlan & Webley, 2021).

Uno de los elementos diferenciales clave del sistema es el buscador de jurisprudencia basado en Recuperación Aumentada por Generación (RAG), diseñado para realizar consultas jurídicas en lenguaje natural y devolver respuestas contextualizadas con citas verificables de legislación, doctrina y jurisprudencia argentina dentro del ámbito del derecho laboral. Este módulo combina búsqueda semántica en la base de conocimiento legal con generación asistida por modelos de lenguaje, garantizando la trazabilidad de las fuentes y la pertinencia de los resultados. A su vez, el sistema cuenta con la opción de conectarse con modelos GPT para ampliar la comprensión de las consultas complejas y generar resúmenes explicativos o comparativos entre fallos, sin alterar la base jurídica original. Esta integración híbrida entre recuperación documental y generación controlada permite mantener el control humano sobre la interpretación legal, al tiempo que agiliza la investigación y redacción jurídica. En conjunto, representa una innovación relevante dentro del ecosistema LegalTech, al superar las limitaciones de los buscadores tradicionales basados en coincidencias léxicas mediante una comprensión semántica y contextual de las fuentes jurídicas (Ajevski et al., 2023).

Se resalta también que esta herramienta ha sido diseñada específicamente para el contexto jurídico de la Provincia de Buenos Aires y la Ciudad Autónoma de Buenos Aires en el fuero laboral, contemplando características del expediente digital local, las prácticas habituales del fuero y las necesidades de los abogados en formación. Esta focalización funcional y su costo accesible le otorgan ventajas frente a soluciones generalistas, consolidando así su valor como producto tecnológico original.

Finalmente, la seguridad de los datos constituye una prioridad central del diseño. Se implementaron protocolos de cifrado, control de accesos mediante autenticación por rol, y una política de privacidad clara que limita el procesamiento de datos por parte de los módulos de IA.

En conjunto, estos elementos consolidan a la propuesta como una herramienta jurídica inteligente, centrada en las necesidades reales de los abogados juniors, superando las funcionalidades de las plataformas actualmente disponibles y sentando las bases para una nueva categoría de software especializado en el análisis visual, seguro y colaborativo de causas legales.

2.2.7. Conclusión

La revisión evidencia que, a pesar de contar con plataformas consolidadas para la administración de expedientes (como Lex-Doctor) y con prototipos de vanguardia capaces de generar diagramas y cronogramas automáticos (LawMaps, LegalViz, LegalVis), persisten al menos tres carencias críticas. Primero, la coherencia temporal de los eventos extraídos mediante IA suele verse comprometida por errores en la ordenación y el solapamiento de fechas, lo que puede generar cronogramas poco fiables. Segundo, la baja precisión en resultados de modelos de lenguaje generativos cuando se emplean para buscar jurisprudencia, ya que tienden a producir coincidencias inexactas o referencias no verificables, fenómeno conocido como *alucinación* en inteligencia artificial (Katz et al., 2023); dichos modelos, al carecer de acceso directo a bases jurídicas estructuradas o actualizadas, pueden ofrecer información general o incluso errónea ante consultas específicas sobre derecho local argentino. Y tercero, la explicabilidad integrada de las sugerencias automatizadas (por ejemplo, por qué se propone un cambio de plazo o la asignación de una audiencia) no forma parte de la mayoría de las herramientas, lo que dificulta la confianza y la adopción por parte de los usuarios.

A continuación, se incorpora la propuesta de esta tesis, la cual será denominada LexGO en adelante, en la tabla comparativa ya desarrollada, evidenciando las diferencias sustantivas respecto de las herramientas existentes en el ecosistema LegalTech. La fila correspondiente resume sus aportes diferenciales en cuanto a visualización, automatización e interacción colaborativa.

TABLA II: Comparación entre aplicaciones jurídicas existentes.

Aplicación / Concepto	Visualización Temporal	Nivel de Automatización	Limitaciones Principales
Lex-Doctor	Calendario estático de vencimientos.	Baja (entradas manuales; sin IA).	No ofrece vistas gráficas interactivas ni análisis automático de plazos; requiere registros manuales.
LawMaps	Diagrama de flujo UML con etapas y dependencias	Baja-Media (manual con apoyo de UML)	Representaciones estáticas; no permite interacción ni validación directa de fechas.
LegalViz	Cronogramas simplificados (presentación y devolución de documentos)	Alta (modelos de lenguaje)	Precisión temporal limitada; requiere reglas heurísticas adicionales para agrupar eventos
LegalVis	Cronología interactiva de jurisprudencia	Media (clasificación + XAI)	Cobertura centrada en precedentes; no gestiona otros hitos procesales
Chatbots & Dashboards	Dashboards tipo Gantt y notificaciones de plazo	Media (alertas automáticas; chatbots con DB)	Depende del cambio cultural y capacitación; sin explicaciones integradas de por qué se generan ciertas alertas
Propuesta de esta Tesis (LexGO)	Línea de tiempo interactiva y grafo relacional dinámico	Alta (IA + explicabilidad integrada)	Solución integral orientada al Derecho Argentino, con sugerencias explicadas, búsqueda de jurisprudencia, y representación visual de causas

La propuesta de esta tesis está diseñada para cerrar estas brechas. Por un lado, se busca garantizar que las fechas y eventos se extraigan con precisión, se agrupan lógicamente y

se presenten en un orden cronológico coherente. Por otro, el dashboard principal incluye control de causas y representación visual de eventos más próximos.

Por su parte, los mecanismos incorporados explicarán en lenguaje claro y con visualizaciones interactivas el fundamento de cada alerta o recomendación, por ejemplo, señalando el fragmento de la causa que originó la sugerencia y el criterio legal asociado, fortaleciendo la confianza en el sistema. Por lo tanto, esta solución integral no solo supera las limitaciones identificadas, sino que optimiza la gestión de causas para abogados de todos los niveles, en especial aquellos que recién están ingresando al mundo laboral, reduciendo errores procesales, mejorando la transparencia y liberando tiempo para actividades de mayor valor estratégico.

En este sentido, la plataforma se consolida no sólo como una innovación tecnológica, sino como una respuesta adaptada a las particularidades del sistema judicial argentino, ofreciendo una solución específica para los desafíos operativos y formativos de los abogados en etapa inicial de su carrera.

2.3. Competencia

El análisis de la competencia permite identificar las soluciones existentes dentro del ecosistema LegalTech y evaluar en qué medida abordan problemas similares a los que resuelve el presente proyecto.

2.3.1. Océano Azul

En el ámbito legal tanto argentino como internacional existen diversas herramientas tecnológicas consolidadas. El enfoque del Océano Azul, propuesto por W. Chan Kim y Renée Mauborgne (2005), plantea que la verdadera innovación no surge de competir en mercados saturados (*océanos rojos*), sino de crear un nuevo espacio de mercado, donde la competencia se vuelve irrelevante. Esto se logra a través de la innovación en valor, es decir, ofrecer simultáneamente mayor diferenciación y menor costo, generando una propuesta atractiva tanto para usuarios actuales como para no-consumidores del sector.

A continuación, se desarrollan tres herramientas estratégicas complementarias que permiten analizar el posicionamiento de la propuesta bajo este marco conceptual.

2.3.1.1. Matriz ERIC (Eliminar, Reducir, Incrementar y Crear):

Permite identificar los factores que el proyecto eliminará, reducirá, incrementará o creará respecto a las soluciones existentes, definiendo así su diferencial competitivo (KIM & MAUBORGNE, 2005).

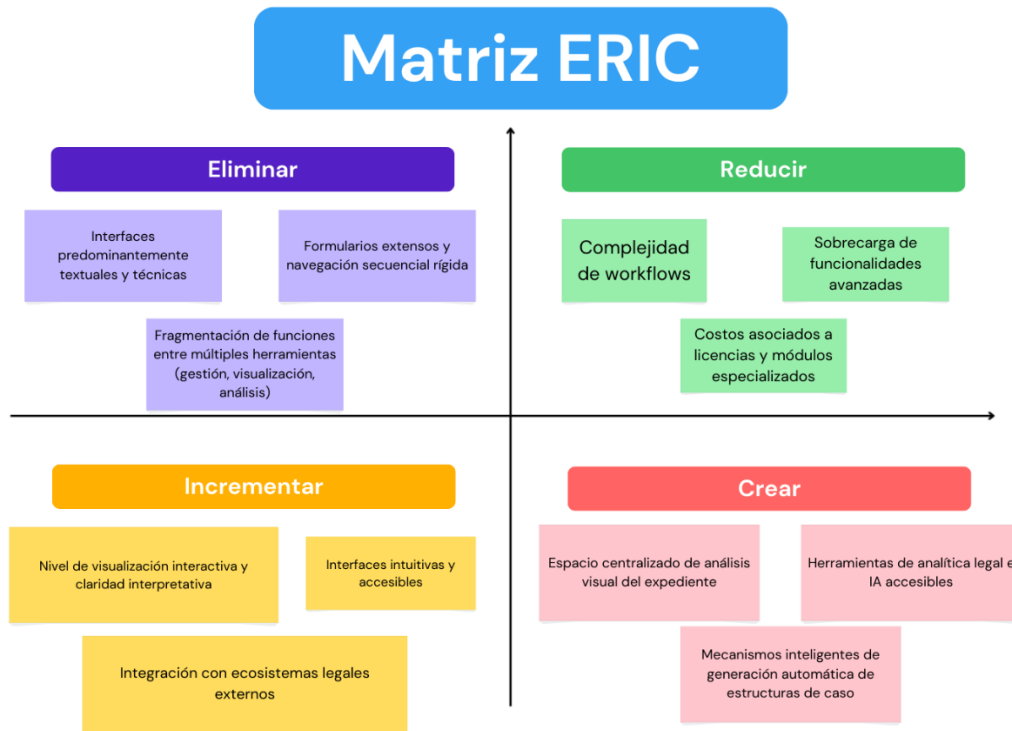


Figura 1. Análisis de los elementos existentes en la propuesta que definen su diferencial competitivo (Elaboración Propia, 2025).

2.3.1.2. Diccionario de variables

Eliminar:

- Interfaces predominantemente textuales y técnicas, ya que obligan al usuario a leer grandes volúmenes de texto y seguir flujos complejos que dificultan la interpretación global del expediente.
- Formularios extensos y navegación secuencial rígida, característicos de software jurídico tradicional, que consumen tiempo y generan fricción.
- Fragmentación funcional entre múltiples herramientas, donde el usuario debe alternar entre sistemas de gestión, plataformas de visualización y buscadores externos.

Reducir:

- Complejidad de workflows con numerosos pasos manuales, ya que aumentan el tiempo de trabajo y la probabilidad de error.
- Exceso de funcionalidades avanzadas orientadas a usuarios expertos, que termina saturando la interfaz y dificultando la adopción por parte de abogados juniors.
- Costos de licencias y módulos especializados.

Incrementar:

- Visualización interactiva del expediente (líneas de tiempo, grafos, paneles), para mejorar la comprensión inmediata del caso, destacar relaciones entre hechos y facilitar el análisis.
- Integración con ecosistemas legales externos, como jurisprudencia, expedientes digitales o documentos cargados por el usuario.
- Calidad de la experiencia de usuario, promoviendo interfaces intuitivas, accesibles y adaptadas a las necesidades de abogados con poca experiencia en tecnología.

Crear:

- Un espacio centralizado de análisis visual del caso, que combine cronología, relaciones entre actores y evidencia relevante.
- Herramientas de analítica legal e IA accesibles a abogados en etapas iniciales, democratizando capacidades avanzadas que antes estaban limitadas a organismos o software especializado.
- Mecanismos inteligentes para generar estructuras básicas del caso y sugerencias automáticas a partir de documentos.

2.3.1.3. Curva de Valor:

Representa gráficamente cómo se distribuyen las prioridades de valor entre las principales soluciones del mercado jurídico actual (Lex-Doctor, Prometea, TrialLine y MEV) y cómo *LexGO* redefine dichos factores para construir un nuevo espacio estratégico (Lex-Doctor, s.f.; Estévez et al., 2025; TrialLine, 2024, Portal Oficial del Estado Argentino, s.f).

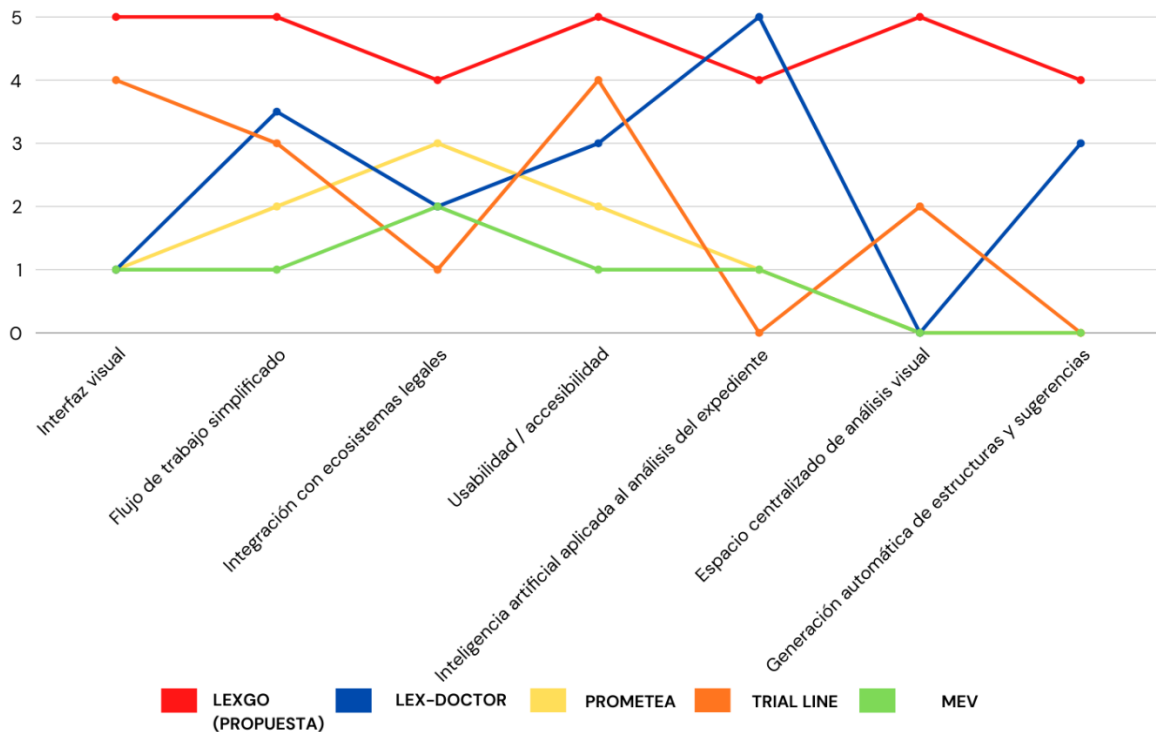


Figura 2. Comparación de la solución propuesta (LexGO) respecto a soluciones presentes en el mercado (Elaboración propia, 2025).

Con base en la matriz ERIC, se seleccionaron siete factores: nivel de visualización interactiva, simplificación del flujo de trabajo, integración con ecosistemas legales, usabilidad y accesibilidad, inteligencia artificial aplicada, existencia de un espacio centralizado de análisis visual y capacidad de generar automáticamente estructuras preliminares y sugerencias a partir de documentos. Estos factores reflejan directamente los atributos que la propuesta busca potenciar y que no se encuentran completamente desarrollados en las herramientas actuales. En el eje horizontal se presentaron las variables considerables para una inversión, siendo el eje vertical el indicador del nivel de ofrecimiento de esa variante, en donde 0 indica el nulo ofrecimiento, y el 5 un alto ofrecimiento de la misma.

Los resultados muestran que LexGO presenta un perfil de valor claramente diferenciado, con puntuaciones elevadas en visualización interactiva, usabilidad, integración, IA aplicada y creación de un espacio centralizado de análisis visual. Estas dimensiones representan una ruptura con el enfoque tradicional centrado en la gestión documental y

administrativa. Por el contrario, soluciones consolidadas como Lex-Doctor o MEV exhiben fortalezas en ámbitos administrativos, pero niveles bajos de visualización, usabilidad y capacidades de análisis. Asimismo, TrialLine muestra un desempeño elevado en visualización temporal, aunque carece de IA integrada y de un enfoque analítico completo. Por su parte, Prometea sobresale en automatización e inteligencia artificial orientada a dictámenes en organismos públicos, pero no aborda la representación visual integral del expediente ni la generación estructural del caso.

La comparación evidencia que la propuesta no compite en los mismos factores que dominan los actores tradicionales, sino que crea un nuevo espacio estratégico centrado en la comprensión visual y asistida por IA de expedientes legales, respondiendo a necesidades poco atendidas de abogados jóvenes y profesionales independientes.

2.3.2. Conclusión

La estrategia de LexGO se enmarca en un Océano Azul, es decir, la creación de un espacio de mercado nuevo y sin competir directamente con las ofertas existentes, orientándose hacia la innovación en valor, combinando atributos de forma única para atender necesidades desatendidas.

En primer lugar, LexGO no pretende ser “otro LexDoctor” ni “otro gestor de expedientes”. En lugar de entrar a competir en el saturado mercado de software de gestión jurídica integral (donde Lex-Doctor y unos pocos rivales dominan ofreciendo cada vez más funcionalidades similares para captar a los mismos abogados), la propuesta se posiciona en otra dimensión: la del análisis y visualización del conocimiento legal. Lex-Doctor, por ejemplo, se centra en el registro y control de expedientes, funcionalidades administrativas donde ya compite con varias alternativas (otros softwares locales o incluso con el propio sistema de gestión judicial). LexGO, en cambio, se enfoca en hacer inteligible la información, en generar comprensión visual, algo que no es el campo de batalla de Lex-Doctor ni de sus competidores tradicionales. Así, LexGO se sitúa aparte, aportando valor de otra naturaleza (cognitiva y estratégica) más que operativa.

Asimismo, no compite contra Prometea, puesto que no es una herramienta para acelerar trámites internos de un organismo, sino para empoderar a los profesionales legales en general con una nueva forma de ver sus casos. Prometea resolvió un problema concreto (tiempos de dictamen en fiscalías) dentro de un mercado muy acotado (justicia penal pública);

LexGO trasciende ese nicho: captura una oportunidad más amplia de mercado llevando capacidades inspiradas en IA y visualización a todos los abogados que hoy no tienen acceso a ellas (Estévez et al., 2025).

En lugar de pelear por ofrecer “la mejor IA para fiscalías” (mercado ya atendido por Prometea en Argentina), crea su propio mercado de “IA + visualización para abogados del ejercicio privado y otras áreas”, donde virtualmente no tiene competidores directos actualmente.

Respecto a otras herramientas de visualización, la propuesta presentada tampoco compite en sus términos, sino que los trasciende integrando y ampliando esa propuesta de valor. Toma la visualización (incluyendo cronologías) y la combina con análisis inteligente y modelado de conocimiento; el resultado es que no es solo una herramienta de presentación, sino un sistema de pensamiento visual para el abogado. En un entorno saturado de software jurídico en el que la mayoría de productos se concentran en la gestión documental, la facturación, la búsqueda de jurisprudencia o la automatización de documentos, casi ninguno ofrece una plataforma integral de visual analytics legal. Por tanto, LexGO abre un océano azul al no encasillarse en ninguna de las categorías existentes de productos legaltech, sino crear una categoría nueva.

Un factor clave de la estrategia de océano azul es la creación de nueva demanda: la aplicación propuesta apunta a atraer tanto a usuarios actuales insatisfechos con lo disponible, como a no-consumidores de tecnología legal (Kim & Mauborgne, 2005). Por ejemplo, muchos abogados juniors no usan software especializado porque lo ven complejo o costoso; se consideran no-clientes actualmente. LexGO, con su facilidad de uso y bajo costo, busca convertirlos en usuarios al ofrecerles algo completamente diferente (una forma visual de trabajar) que antes no contemplaban, y posicionándose colaborativamente más que en competencia ya que coexiste con las herramientas tradicionales, agregando valor donde éstas no llegan.

Por último, vale destacar que la tendencia hacia la visualización y simplificación de la información legal es cada vez más reconocida como necesaria en la profesión (Soto, 2018). Sin embargo, pocas soluciones concretas lo materializan en el mercado argentino. LexGO se adelanta a esta tendencia, creando su propio espacio como pionera en *Legal Data Visualization* local. Al hacerlo, se posiciona como primer candidato en un océano azul, con la posibilidad de establecer el estándar antes de que otros ingresen, ofreciendo un salto en valor novedoso,

evitando la competencia frontal y cambiando las reglas del juego en el ámbito de la tecnología legal.

3. Descripción

En esta sección se analiza al proyecto como solución, abarcando cada aspecto fundamental de su desarrollo, incluyendo la investigación sobre los potenciales usuarios, el diseño de la solución, y la implementación de algoritmos.

3.1. User Research

Como parte del proceso de validación temprana de la problemática, se llevaron a cabo entrevistas en profundidad y encuestas exploratorias a personas en distintas etapas del ejercicio profesional del derecho. Estas actividades permitieron contrastar la hipótesis inicial con datos cualitativos y cuantitativos, y refinar tanto los objetivos como las funcionalidades centrales de la solución propuesta.

3.1.1. Entrevistas cualitativas

La presente sección describe el proceso de recolección y análisis de información mediante entrevistas cualitativas realizadas a abogados. El objetivo de estas entrevistas fue comprender en profundidad las necesidades, dificultades y patrones de trabajo asociados al manejo de expedientes legales, así como evaluar la percepción de valor de las funcionalidades propuestas para la aplicación. Este enfoque permitió obtener información valiosa, contextual y no estructurada, fundamental para orientar el diseño centrado en el usuario, identificar oportunidades de mejora y validar hipótesis preliminares del proyecto. Las entrevistas completas se pueden observar en detalle en Anexo B y Anexo C.

3.1.1.1. Entrevista a la Dra. Michelle Trouchet

Una de las entrevistas significativas fue realizada a Michelle Trouchet, abogada egresada en 2017, con posgrado en inteligencia artificial y experiencia en el sector público y privado. A lo largo de la entrevista surgieron varios aspectos relevantes:

- A pesar de usar Lex-Doctor, Michelle confecciona sus propios listados en Excel para ordenar las causas por estados procesales. Esto sugiere que incluso quienes cuentan con software jurídico profesional **aún recurren**

a **sistemas caseros** para suplir limitaciones de visualización y organización.

- Recurre a **modelos de escritos legales** como punto de partida para su trabajo, y valora herramientas que le permitan adaptarlos fácilmente según el caso.
- Destacó como necesidad concreta la posibilidad de **ver todas las causas juntas**, segmentadas por estado, y recibir recordatorios de plazos.
- Si bien trabaja mayormente desde la computadora, **consideró útil tener acceso móvil** a un panel de consulta y notificaciones.
- Finalmente, consideró que una aplicación que integre estas funcionalidades, con una buena interfaz y un precio razonable, **podría reemplazar herramientas como Lex-Doctor** para muchos abogados jóvenes.

3.1.1.2. Entrevista al Dr. Marcos Vilaplana

Con el objetivo de profundizar el entendimiento del entorno profesional al que se dirige la herramienta propuesta, se llevó a cabo una entrevista con un referente del ámbito jurídico con amplia trayectoria, el Doctor Marcos Vilaplana, presidente del Colegio de Abogados de General San Martín desde hace más de quince años. Esta instancia cualitativa permitió relevar percepciones, desafíos y oportunidades en torno al uso de tecnologías en la práctica legal, particularmente en el contexto bonaerense.

Uno de los principales ejes abordados fue la accesibilidad tecnológica. El entrevistado destacó que la resistencia al uso de herramientas digitales no se limita a cuestiones etarias, sino que responde a factores culturales y formativos. En consecuencia, se enfatizó la necesidad de desarrollar interfaces altamente intuitivas, con lógica mobile-first, onboarding guiado y un lenguaje claro, de manera que puedan ser utilizadas tanto por profesionales jóvenes como por aquellos con menor familiaridad con entornos digitales. En palabras del entrevistado: *“Me transformé en un predicador de la tecnología, aunque insisto, todavía yo soy prehistórico”*.

Asimismo, señaló que la pandemia funcionó como catalizador de la digitalización del expediente judicial en la Provincia de Buenos Aires, impulsando la incorporación de herramientas como la MEV o Lex-Doctor. Este cambio fue descrito como un

proceso repentino e inevitable: *“La pandemia hizo obligatorio el uso”*. No obstante, se identificaron limitaciones en estas plataformas, las cuales, si bien permiten el acceso remoto a documentación, no ofrecen soporte cognitivo para interpretar la estructura lógica del caso. Este diagnóstico refuerza la hipótesis que motivó el desarrollo de una plataforma complementaria orientada a la visualización de eventos, relaciones y plazos.

A pesar de no haberse profundizado en tareas operativas concretas del abogado junior, se infirieron diversas dificultades comunes: desorganización en la recepción y archivo de documentos, uso fragmentado de herramientas, carencia de alertas procesales y errores frecuentes por omisión de pasos formales. A partir de estos datos, se propuso una solución que integre funcionalidades de gestión de múltiples causas, visualización temporal y organización semántica de elementos jurídicos. En relación con ello, el entrevistado expresó: *“Los plazos procesales son importantísimos. [...] Si no lo incorporaste en el momento oportuno y no tuviste en cuenta el plazo, te quedaste sin tener razón”*.

Otro aspecto abordado fue el uso de inteligencia artificial. Se reconoció su potencial para mejorar la eficiencia profesional, aunque también se advirtió sobre el riesgo de una posible “pereza intelectual” al delegar el razonamiento jurídico en modelos automatizados. En función de ello, se propuso diseñar mecanismos de IA que no sustituyan al profesional, sino que lo estimulen críticamente, por ejemplo, mediante preguntas orientadoras sobre la claridad, fundamentación y coherencia de sus escritos.

Finalmente, se identificaron barreras adicionales vinculadas a la formación académica, la seguridad de los datos y la confianza en las nuevas tecnologías. En este sentido, se destacó la importancia de garantizar la privacidad de la información mediante cifrado, control sobre los datos procesados por la IA y posibilidad de uso offline.

La entrevista aportó información valiosa para el diseño conceptual de la herramienta, validando la relevancia de funciones como la línea de tiempo interactiva, el grafo relacional y el revisor inteligente, así como la necesidad de una arquitectura flexible que contemple distintos niveles de experiencia tecnológica entre los usuarios. En referencia al manejo simultáneo de causas, se remarcó que *“el secreto está en no solamente ver una causa, sino que eso también se alimente con las otras causas que tenés y te prenda una luz amarilla o roja”*. La propuesta se consolidó así como una solución viable para asistir a abogados en etapas iniciales y, potencialmente, a profesionales de mayor trayectoria en proceso de adaptación digital.

3.1.2. Encuestas cuantitativas

Se diseñó una encuesta dirigida a estudiantes avanzados, egresados recientes y abogados en ejercicio. Se obtuvieron 101 respuestas (cantidad determinada por criterios de conveniencia y representatividad del universo de abogados juniors y estudiantes avanzados), distribuidas en diversos perfiles y universidades del país. En el contexto académico, este número resulta adecuado ya que equivale aproximadamente a dos salones completos de cursada universitaria de abogacía, considerando que el promedio habitual de estudiantes por comisión ronda los 60 alumnos. Los principales hallazgos fueron:

- **El 99% de los encuestados afirmó que una herramienta que visualice sus causas de forma estructurada les facilita el trabajo.**
- Las funcionalidades más priorizadas fueron:
 - Línea de tiempo con hitos procesales.
 - Alertas de errores comunes y fechas límite.
 - Visualización de relaciones entre actores, hechos y documentos.
- La **dispersión de información** fue el problema más reportado: los usuarios mencionaron usar simultáneamente Drive, Word, Excel, MEV y correos electrónicos, lo que complica el seguimiento de causas.
- Entre las tareas más difíciles se destacan: **leer y analizar expedientes, recordar plazos y buscar jurisprudencia.**
- En cuanto al acceso, la mayoría prefiere usar la computadora, aunque muchos valoran contar con una **versión móvil complementaria para consultas rápidas.**
- Las características más valoradas en una app legal fueron:
 - Interfaz simple y clara.
 - Foco local en el sistema judicial argentino.
 - Protección de datos personales.

Los resultados de las encuestas pueden observarse en Anexo C.

3.1.2.1. Síntesis y decisiones de diseño

A partir de estos insumos se tomaron decisiones importantes en el diseño funcional del producto, priorizando:

- **Visualización estructurada del estado procesal** mediante una línea de tiempo.
- **Panel de causas múltiples**, con filtros por estado y alertas de vencimientos.
- **Gestor de jurisprudencia**, que permita consultar jurisprudencia aplicable en el ámbito del derecho laboral argentino en la Provincia de Buenos Aires y en la Ciudad Autónoma de Buenos Aires.
- **Gráfico de relaciones** que mapee hechos, actos y documentos del caso.
- **Acceso móvil para consulta y notificación**, manteniendo la edición en versión escritorio.

Estas funcionalidades buscan aliviar la carga cognitiva, reducir la fragmentación de la información y acompañar al profesional joven en el desarrollo de su razonamiento jurídico y la gestión de sus causas.

3.2. Requerimientos del sistema

A partir del análisis de la problemática planteada, del marco teórico revisado junto con las encuestas realizadas a estudiantes avanzados y abogados en sus primeros años de ejercicio, y las entrevistas realizadas a profesionales, se identificaron un conjunto de necesidades comunes que orientan el diseño del sistema. Dichas necesidades se centran en la dificultad de organizar, comprender y comunicar de manera clara el estado de una causa judicial en entornos donde prevalece la sobrecarga documental y la fragmentación de herramientas digitales. En consecuencia, la aplicación propuesta se constituye en un espacio único, de acceso sencillo y disposición visual clara, que permite a los usuarios **registrar, consultar y administrar sus causas** en un entorno estructurado y consistente con sus prácticas profesionales cotidianas.

Por lo tanto, en términos generales, el sistema ofrece **mecanismos de autenticación y gestión de cuentas, administración de causas y expedientes con sus datos principales**, asociación de los eventos procesales más relevantes a las causas. Estos eventos se visualizan mediante una **línea de tiempo interactiva**, que funciona como apoyo cognitivo para la comprensión secuencial del caso. Asimismo, se considera fundamental contar con una **representación gráfica de actores, documentos y relaciones** mediante un grafo, así como con

funcionalidades de **resumen y recordatorio** que ayuden al usuario a retomar rápidamente el estado de un expediente después de un período de inactividad.

De igual modo, el sistema incluye un **panel de causas** que permite al profesional tener un pantallazo global de sus expedientes, identificar vencimientos próximos y detectar rápidamente posibles retrasos. La centralización en una sola plataforma, combinada con principios de diseño centrado en el usuario, apunta a reducir la dispersión de tareas entre múltiples aplicaciones y mejorar la productividad diaria. Asimismo, el prototipo incorpora **módulos de inteligencia artificial** destinados a asistir en la redacción de resúmenes y en la detección de inconsistencias, así como también la búsqueda de jurisprudencia aplicable, lo que fortalece la propuesta de valor al brindar apoyo cognitivo directo a los usuarios.

Por lo tanto, los requerimientos del sistema que se presentan a continuación surgen de la integración entre el **fundamento teórico**, el **estado del arte**, la **propuesta de valor aprobada** y la **investigación con usuarios reales**, y constituyen la base para la definición detallada de las funcionalidades y atributos no funcionales que guiarán el desarrollo del prototipo. El glosario de requerimientos completo se puede contemplar en el Anexo E.

3.2.1. Tabla de Requerimientos Funcionales

A partir del glosario presentado en la sección anterior, se enumeran a continuación los **requerimientos funcionales** que estructuran el sistema propuesto. Cada requerimiento se identifica con un código único (**ID**) y se acompaña de un **nombre**, una **descripción** sintética, los **datos principales involucrados** y la **etapa de release** en que se realizó su implementación.

Para simplificar la redacción en los casos de operaciones repetitivas, se emplea la abreviatura **CEE (Crear, Editar, Eliminar)**, equivalente al tradicional “alta, baja y modificación”. Esta convención se utilizará en todos aquellos requerimientos que impliquen la gestión completa de entidades del dominio (causas, eventos, documentos, nodos del grafo).

La tabla resultante refleja los requerimientos centrales del sistema y servirá de base para la trazabilidad hacia los objetivos planteados y la definición de criterios de validación.

TABLA III: Requerimientos centrales

<i>ID</i>	<i>Nombre</i>	<i>Descripción</i>	<i>Datos</i>	<i>Release</i>
<i>FR-01</i>	CEE Causas	Permite crear, editar y eliminar causas/expedientes como unidad de trabajo legal.	Título, descripción, fuero, jurisdicción, estado inicial	MVP
<i>FR-02</i>	Header de Causa	Muestra en la vista de detalle un encabezado con: título, descripción, estado y próximo evento más importante.	Título, desc, estado, fecha próximo evento	MVP
<i>FR-03</i>	CEE Eventos	Permite CEE eventos procesales asociados a una causa.	Título, tipo, fecha/hora, notas	MVP
<i>FR-04</i>	Línea de Tiempo	Visualiza los eventos de una causa en orden cronológico, con zoom, arrastre y botón para centralizar.	Eventos (id, fecha, tipo, estado)	MVP
<i>FR-05</i>	Panel Global de Causas	Home con listado de causas y resumen de estado; filtros por urgencia, estado, cliente.	Lista de causas, estados, metadatos	MVP
<i>FR-06</i>	Dashboard Semanal	Resumen general de semana: causas activas, eventos próximos, recientes y vencidos.	Eventos multi-causa, calendario	MVP
<i>FR-07</i>	CEE Documentos	Gestión documental: subir, previsualizar, renombrar, filtrar, ordenar y eliminar documentos vinculados a causas/eventos.	Archivos, metadatos (fecha, tipo, nombre)	MVP

FR-08	Grafo Relacional	Representa visualmente actores, hechos y documentos; permite agregar nodos, conectarlos, moverlos y eliminarlos.	Nodos (tipo, etiqueta), relaciones	MVP
FR-11	Resumen de IA Generativa	Asistente de IA genera resúmenes narrativos de la evolución de la causa.	Texto de eventos, documentos	MVP
FR-12	Asistente Conversacional	Interfaz para consultas en lenguaje natural: preguntas, verificación de documentos, sugerencias de pasos.	Texto libre del usuario	MVP
FR-13	Perfil y Ajustes	Gestión básica de usuario: editar datos, cambiar contraseña, preferencias.	Nombre, email, password, config.	MVP
FR-14	Autenticación	Registro, login, logout y recuperación de contraseña.	Email, password	MVP
FR-15	Buscador de Jurisprudencia Asistido por IA	búsquedas semánticas de jurisprudencia mediante embeddings y recuperación aumentada con LLM.	Texto de consulta, metadatos de jurisprudencia vectorizada	MVP
FR-16	Generación Automática de Causa desde Documento	Crea una causa inicial a partir del análisis de un documento cargado.	Archivo (PDF)	MVP
FR-17	Análisis Estructural con IA	Evalúa las estructuras de datos jurídicas (eventos, documentos, relaciones) para detectar errores, omisiones o inconsistencias, y sugiere	Estructuras JSON de causa	MVP

		correcciones basadas en patrones legales.		
R-18	Gestión de tareas centralizada	Permite agregar, editar y borrar tareas en las causas para mejorar la productividad del usuario	Estructuras JSON de causa	MVP
R-19	Visualizador de Trazabilidad	Permite ver todas las acciones y entidades que fueron agregadas, modificadas o eliminadas en una causa en el tiempo, lo que permite al usuario tener un control de todo lo que realizó.	Estructuras JSON de causa	MVP

3.3. Diseño y Arquitectura

En continuidad con los requerimientos funcionales definidos, se presenta a continuación la arquitectura de la solución propuesta. Desde un punto de vista tecnológico, la aplicación se estructura sobre un patrón estándar ampliamente utilizado en la industria: un Frontend web que implementa la interfaz de usuario, un Backend que concentra la lógica de negocio y la comunicación con servicios externos, y una base de datos alojada en la nube para persistencia y almacenamiento de información.

Si bien no se trata de una arquitectura disruptiva en términos tecnológicos, el valor diferencial radica en cómo se organiza la solución en torno a las necesidades específicas de los abogados junior, garantizando modularidad, mantenibilidad y la posibilidad de escalar en el tiempo. En este sentido, la arquitectura se describe mediante distintas vistas y diagramas que documentan tanto su estructura general como sus características no funcionales más relevantes, tales como capacidad de escalamiento, robustez y facilidad de integración.

Para favorecer la comprensión, la presentación comienza con diagramas de contexto que muestran la aplicación desde el punto de vista del usuario y de los sistemas externos con los que interactúa. Finalmente, se presentan consideraciones adicionales sobre

modularidad en Frontend y Backend, así como la manera en que la arquitectura materializa los requerimientos no funcionales definidos.

3.3.1. Diagrama de Contexto

El primer paso para comprender la arquitectura de la aplicación es situarla en su entorno más general. El **diagrama de contexto** de la Figura 3 muestra cómo los **usuarios principales**, abogados Jr. y estudiantes avanzados, interactúan con la plataforma, y cuáles son los **tres ejes de valor** que esta concentra.

La **Plataforma para abogados junior** funciona como un punto central de acceso a funcionalidades clave que responden a los problemas identificados en la investigación previa:

- **Organización temporal:** administración de causas mediante una línea de tiempo de eventos lo que permite una mejor gestión de plazos y vencimientos, trazabilidad cronológica de acciones del usuario y muchas entidades etiquetadas con fechas.
- **Soporte cognitivo:** incorporación de un asistente con IA generativa, grafos relacionales y elementos gráficos que favorecen la comprensión de las causas y la identificación de relaciones relevantes.
- **Productividad:** centralización de la documentación y seguimiento en un único espacio, tareas y trazabilidad de una causa, lo que evita la dispersión de herramientas y mejora la eficiencia del trabajo cotidiano.

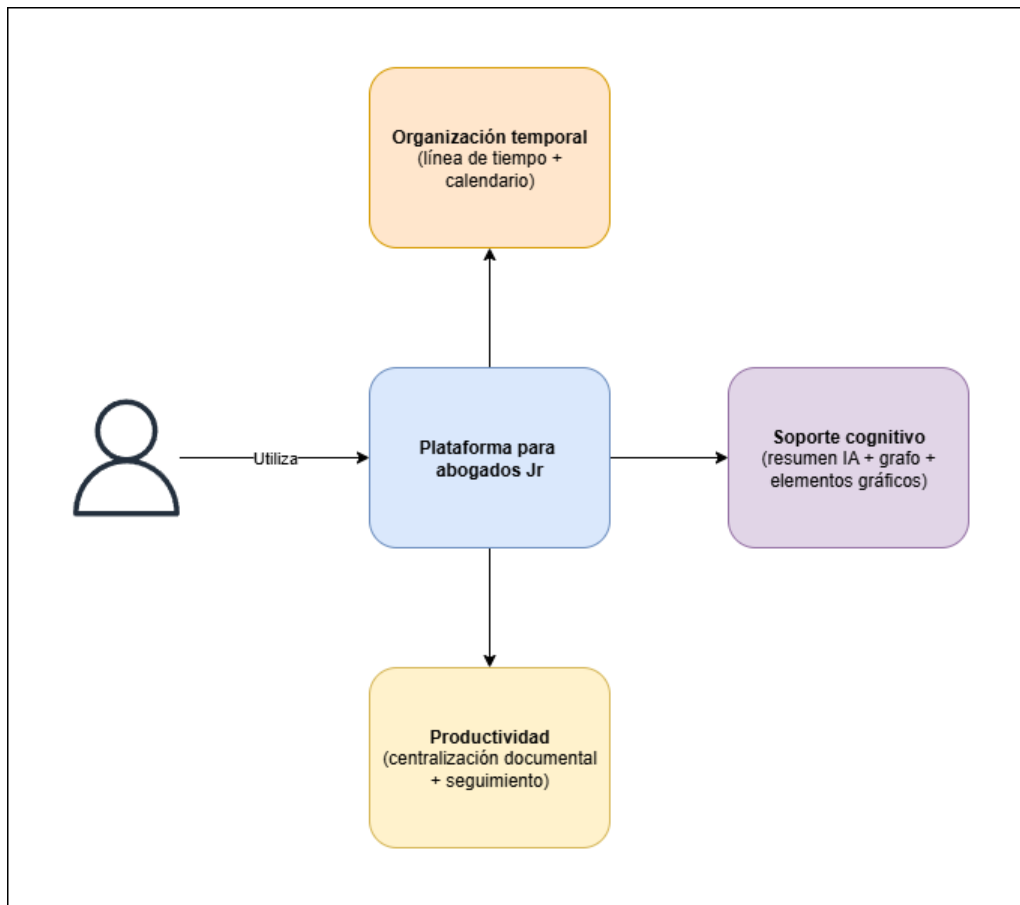


Figura 3. Diagrama de contexto narrativo: el usuario interactúa con la Plataforma para Abogados Junior, que brinda tres ejes de valor: organización temporal, soporte cognitivo y productividad, como respuesta a los requerimientos identificados en entrevistas y en el análisis del dominio. (Elaboración propia, 2025).

3.3.2. Vistas funcionales

Además del diagrama de contexto, resulta pertinente describir la aplicación desde una **perspectiva funcional**, es decir, mostrando cómo se organiza en módulos y vistas que el usuario efectivamente percibe al interactuar con ella. Estas vistas funcionales no reemplazan los diagramas de arquitectura técnica, sino que los complementan: permiten comprender la lógica modular de la interfaz y cómo cada componente responde a los requerimientos previamente definidos.

En el estado actual del prototipo, la plataforma se estructura en tres secciones principales de navegación: **Dashboard**, **Causas** y **Asistente**. Cada una concentra funcionalidades específicas y puede representarse a través de diagramas que resumen sus

componentes internos. La elección de este enfoque responde a que el valor del sistema radica, en gran medida, en la claridad de su interfaz y en cómo esta organiza la información legal en unidades comprensibles para el usuario final (abogado/a junior).

3.3.2.1. Vista funcional: Gestión de Causas

La gestión de causas constituye el núcleo principal de la aplicación, dado que representa la unidad de trabajo central de un abogado junior. Al acceder a esta vista, el usuario puede visualizar un listado de causas activas, junto con información clave que le permite identificar rápidamente el estado de cada expediente.

Tal como se ilustra en la Figura 4, la pantalla de “Causas” se organiza en los siguientes elementos:

- Sección principal: muestra una tabla con las causas registradas.
- Campos visibles: nombre de la causa, estado procesal, fecha de referencia, etiquetas asociadas (tags) y equipo responsable.
- Toolbar “CEE” (Crear, Editar, Eliminar): Crear causa en la parte superior, editar y eliminar causa en la tabla.
- Acceso a detalle: cada fila de la tabla enlaza a la vista de la causa específica, donde se despliegan módulos más complejos (línea de tiempo, grafo, documentos, etc.).

Su diseño busca asegurar simplicidad, legibilidad y escalabilidad, de modo que la vista pueda adaptarse a futuros atributos (como filtros avanzados o colaboración multiusuario).

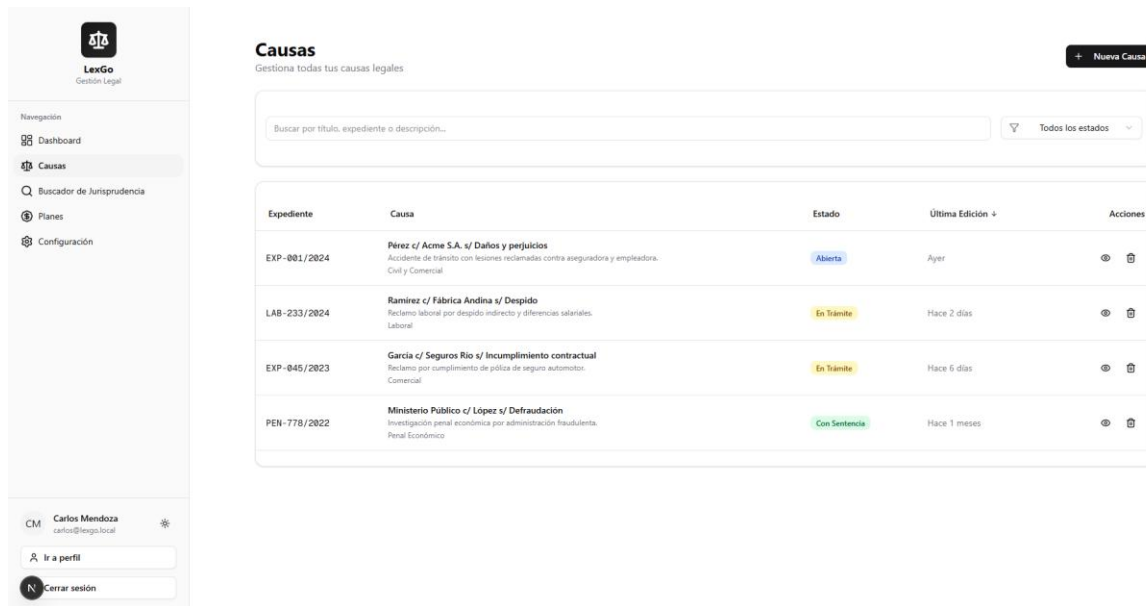


Figura 4. Vista funcional de gestión de causas. (Elaboración propia, 2025)

Uno de los desafíos más significativos en la gestión digital de información jurídica es la carga manual de datos. Este proceso consume tiempo, es propenso a errores y suele constituir un obstáculo para la adopción eficiente de herramientas tecnológicas en el ámbito legal. Con el fin de mitigar esta problemática, se incorpora un módulo orientado a la automatización inicial del registro de una causa a partir de su documento de origen, el cual generalmente constituye el punto de inicio del expediente.

Este módulo permite subir un documento y, mediante técnicas de Machine Learning (Sección 3.6), generar predicciones preliminares sobre eventos, estados procesales y otras entidades relevantes, utilizando para ello modelos previamente entrenados. Asimismo, el sistema es capaz de detectar títulos, identificar eventos ya existentes, reconocer actividades involucradas y construir representaciones gráficas coherentes de la estructura del caso.

Si bien los procesos judiciales presentan múltiples variantes y ramificaciones, lo que dificulta la construcción de un sistema completamente preciso para predecir todas las entidades pertinentes, la herramienta resulta altamente útil para proporcionar una base inicial de organización. Su función principal es ofrecer al usuario un punto de partida estructurado, reducir la necesidad de carga manual y acelerar la comprensión general de la causa. De esta forma, el abogado no comienza con una interfaz vacía, sino con una preconfiguración informada que mejora la experiencia de uso y optimiza los tiempos de trabajo.

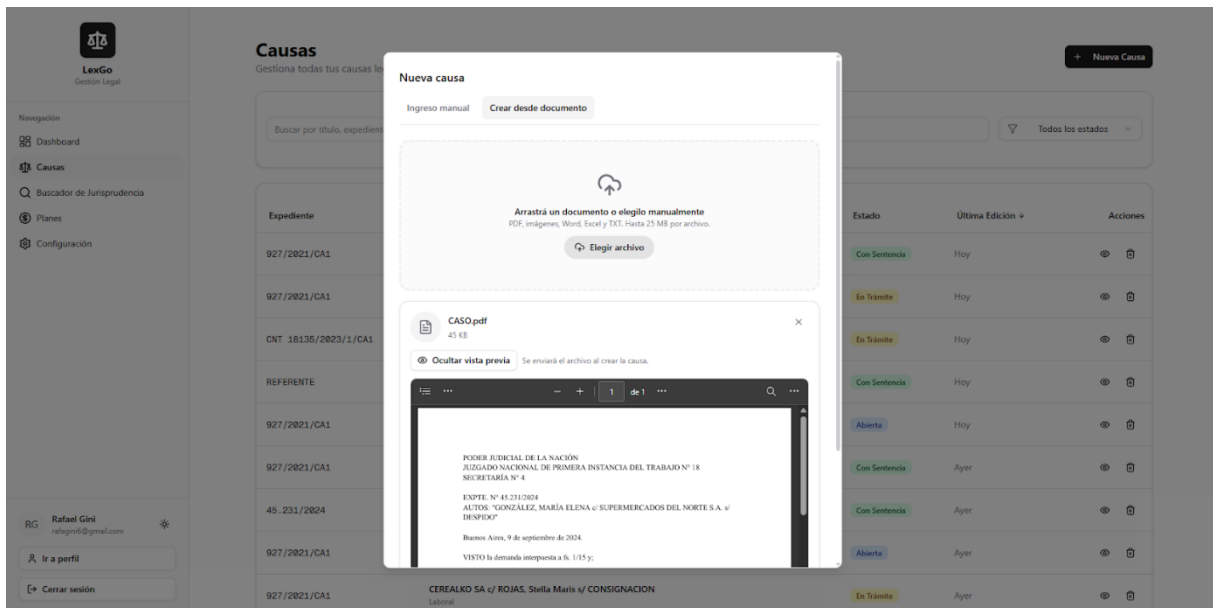


Figura 5. Vista funcional de gestión de causas. (Elaboración propia, 2025)

3.3.2.2. Vista funcional: Causa específica

Una vez seleccionada una causa del listado general, el usuario accede a la vista de detalle de la causa, que concentra toda la información y herramientas necesarias para comprender y gestionar el expediente. Esta sección representa la unidad de trabajo central y constituye la pantalla donde se integran los principales módulos funcionales definidos en los requerimientos.

En primer lugar, la interfaz presenta un encabezado de la causa que reúne la información esencial del expediente. Este encabezado incluye el título identificatorio, el número de expediente, datos relevantes para su seguimiento y el estado actual, el cual puede modificarse según corresponda. También se muestra un aviso destacado con el próximo evento relevante, junto con un conjunto de métricas generales, tales como la cantidad total de eventos registrados, la cantidad de partes involucradas y el número de documentos asociados.

Debajo del encabezado se organizan tres secciones principales del módulo, cada una presentada mediante widgets colapsables. La primera corresponde al resumen generado mediante técnicas de inteligencia artificial, el cual ofrece una síntesis automatizada de la causa. La segunda sección contiene las tareas vinculadas al expediente, permitiendo visualizar y gestionar los pendientes asociados. La tercera sección refiere a la transferibilidad de la causa y agrupa la información necesaria para compartir o trasladar el expediente a otros profesionales.

La disposición en widgets plegables contribuye a una navegación más clara y ordenada, favoreciendo la accesibilidad y la organización del contenido.

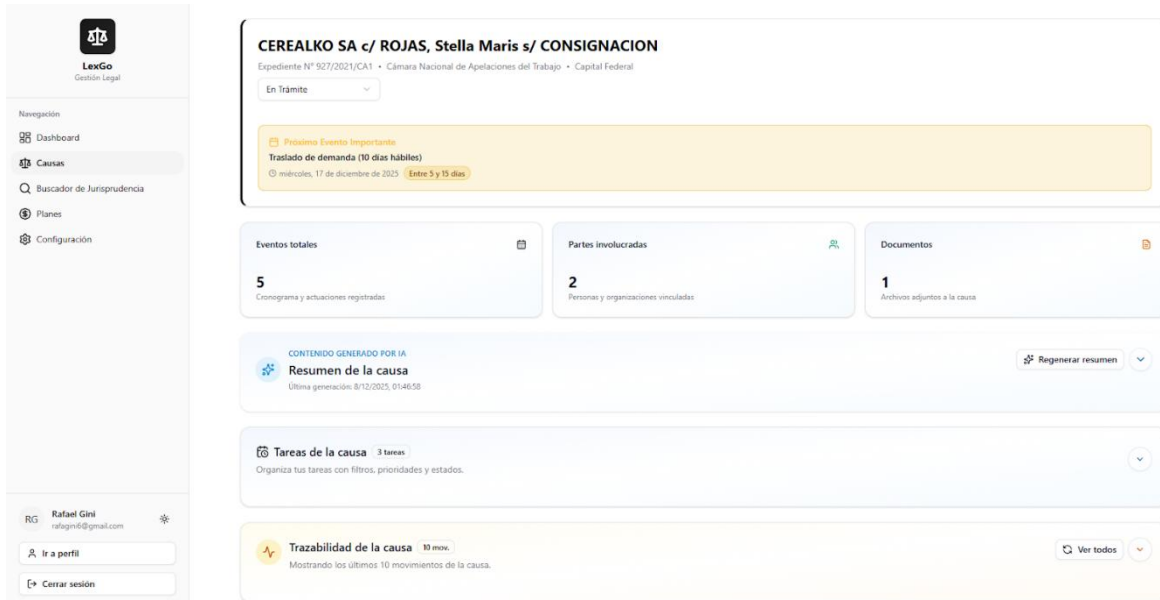


Figura 6. Vista funcional de causa específica: pantalla que integra header, resumen IA, módulo de tareas, trazabilidad. (Elaboración propia, 2025)

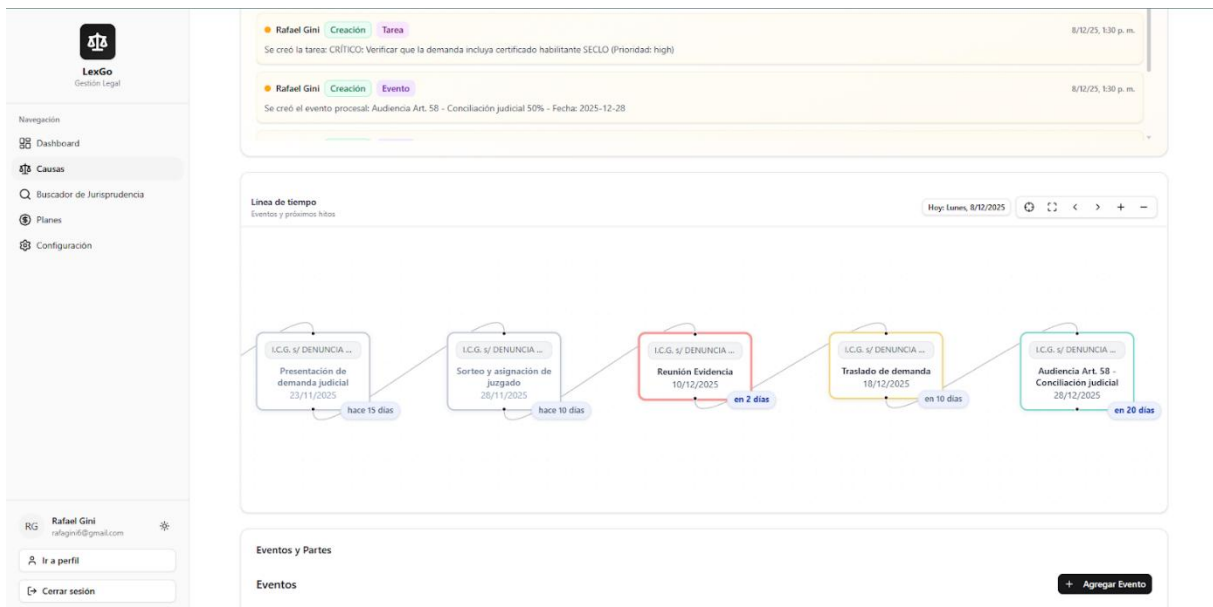


Figura 7. Vista funcional de causa específica: módulo de linea de tiempo. (Elaboración propia, 2025).

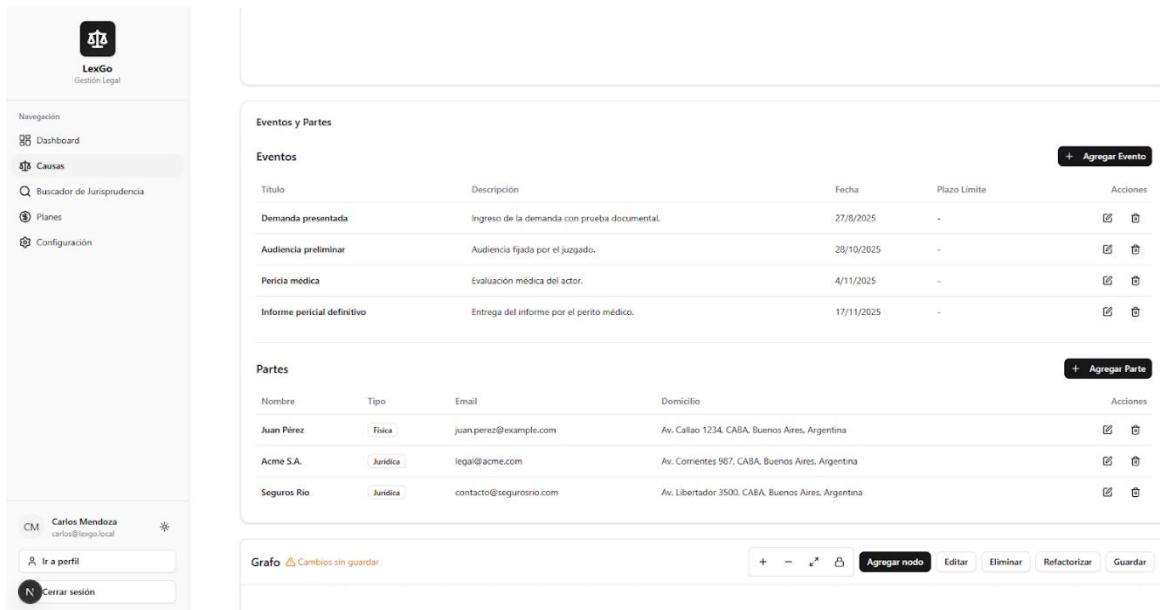


Figura 8. Vista funcional de causa específica: Gestor de actores y eventos.

(Elaboración propia, 2025)

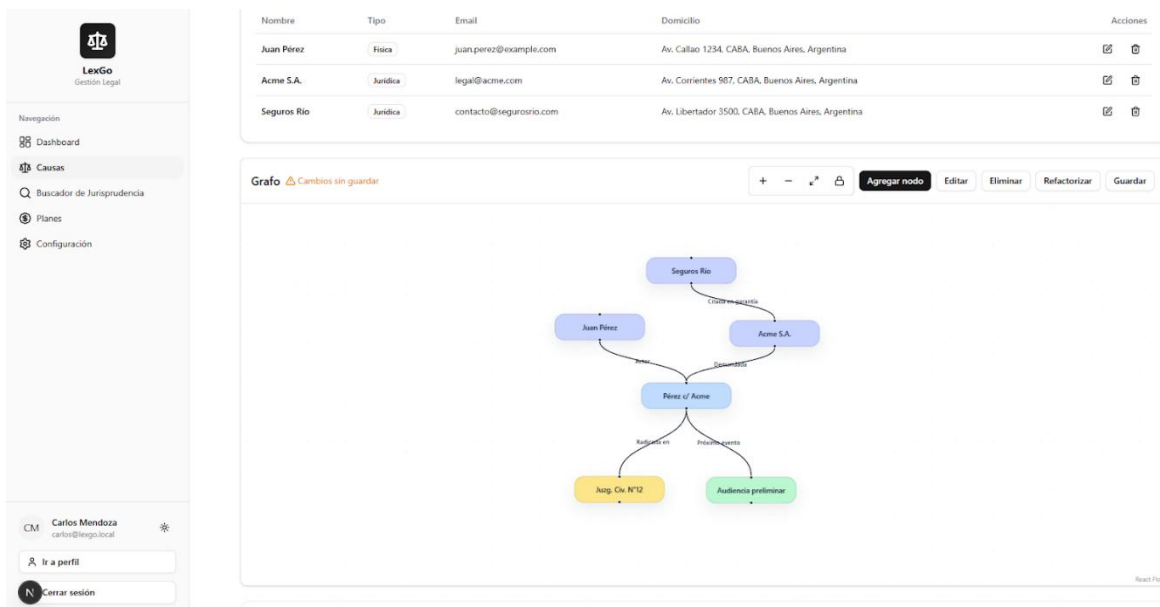


Figura 9. Vista funcional de causa específica: Grafo relacional de actores, eventos y documentos.

(Elaboración propia, 2025)

Tal como se ilustra en las anteriores figuras, la vista de una causa se organiza en diferentes componentes, los cuales son:

- **Header de causa:** muestra información básica y clave del expediente: título, descripción breve, estado procesal actual y próximo evento relevante. Su objetivo es brindar un pantallazo inmediato al usuario sobre la situación de la causa.

- **Resumen con IA generativa:** genera un texto narrativo que sintetiza los eventos, documentos asociados, ofrece asistencia de verificación para errores y omisiones, facilitando la comprensión global del expediente y sirviendo de apoyo cognitivo para retomar el contexto después de un período de inactividad.
- **Línea de tiempo:** visualización cronológica de eventos pasados, presentes y futuros de la causa. Incluye interacción mediante zoom, arrastre y un botón de centrado, además de resaltar vencimientos próximos o pasados.
- **Gestor de documentos y jurisprudencia:** permite CEE (crear, editar, eliminar) documentos asociados a la causa, organizarlos, buscarlos y vincularlos a eventos. Incluye la posibilidad de adjuntar jurisprudencia relevante.
- **Grafo relacional:** representa gráficamente los actores, hechos y documentos vinculados al caso, permitiendo agregar nodos, conectarlos y reorganizarlos para una mejor comprensión de las relaciones.
- **Módulo de tareas:** permite crear, editar y eliminar tareas asociadas a cada causa. Las tareas pueden clasificarse mediante prioridades y estados, lo que facilita la organización del trabajo y la gestión del flujo de pendientes. Este módulo contribuye tanto al seguimiento operativo del abogado como a la generación de métricas internas vinculadas a productividad, carga de trabajo y vencimientos importantes. Asimismo, su diseño permite integrarse con otros componentes del sistema, como la línea de tiempo y la trazabilidad, para ofrecer una visión coherente y estructurada de las responsabilidades del caso.
- **Módulo de trazabilidad:** registra y presenta de manera cronológica todas las acciones realizadas sobre la causa. Permite identificar qué usuario efectuó cada operación y visualizar detalles como la acción realizada: creación, actualización, eliminación, cambio de estado, alta o baja, la entidad afectada (causa, parte, documento, evento, tarea u otra entidad) y, cuando corresponde, el valor previo de la información modificada. Este módulo funciona como un historial integral que facilita la transparencia, evita olvidos y brinda soporte al control interno del caso, permitiendo al abogado revisar de forma clara y ordenada la evolución completa del expediente digital.

El diseño busca ofrecer un espacio centralizado y coherente, en el cual el usuario puede consultar la totalidad de la información de un expediente sin necesidad de alternar entre múltiples aplicaciones. La modularidad de la vista facilita la escalabilidad y la incorporación

progresiva de nuevas funcionalidades. Las ilustraciones de los componentes faltantes pueden observarse en Anexo F.

3.3.2.3. Vista funcional: Dashboard

El Dashboard constituye la vista de entrada principal a la aplicación y permite al usuario obtener un panorama global de su actividad. A diferencia de la vista de causa específica, que se centra en un solo expediente, el Dashboard reúne información consolidada de todas las causas, brindando indicadores rápidos y recordatorios útiles para la organización diaria.

Tal como se ilustra en la Figura 10 a continuación, la pantalla del Dashboard se compone de los siguientes elementos:

- **Header del Dashboard:** contiene el título de la sección y puede incorporar accesos rápidos a filtros o configuraciones generales.
- **Línea de tiempo general:** muestra de forma integrada los eventos de todas las causas, diferenciados por color o etiquetas, lo que facilita identificar vencimientos y audiencias relevantes en un único calendario visual.
- **Kpis:** módulo que presenta estadísticas globales (cantidad de causas activas, cerradas, en curso) junto con la cantidad de tareas “abiertas” que creó el usuario.
- **Listado de casos recientes:** acceso rápido a los últimos expedientes trabajados, facilitando retomar tareas sin necesidad de navegar por todo el listado.

El valor del Dashboard radica en proporcionar al abogado junior una visión integral y consolidada de su carga de trabajo, facilitando la planificación diaria y reduciendo el riesgo de omisiones procesales. Su diseño modular permite la incorporación de métricas adicionales en versiones futuras, como indicadores de productividad, tiempos promedio de resolución o niveles de carga de trabajo por cliente. Las ilustraciones de los componentes faltantes pueden encontrarse en Anexo G.

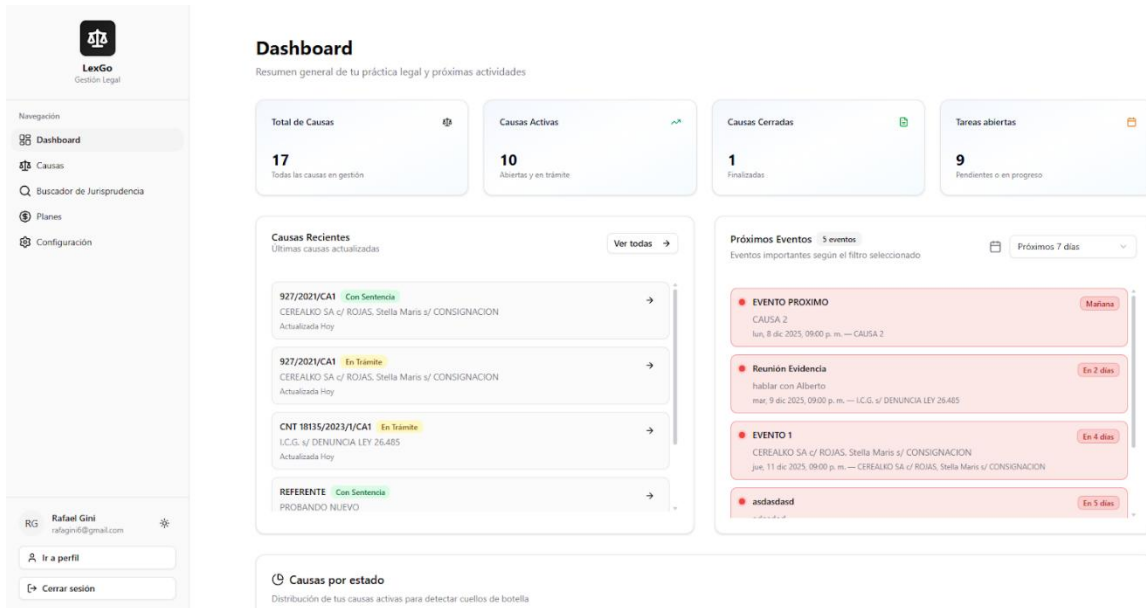


Figura 10. Vista funcional de Dashboard. (Elaboración propia, 2025)

3.3.2.4. Vista funcional: Asistente

El **Asistente** constituye uno de los módulos diferenciales de la plataforma, ya que integra funcionalidades de **inteligencia artificial generativa** para ofrecer soporte cognitivo en el trabajo diario del abogado junior. Esta vista fue diseñada para contemplar tanto **acciones rápidas** como **consultas libres**, combinando simplicidad y flexibilidad de uso.

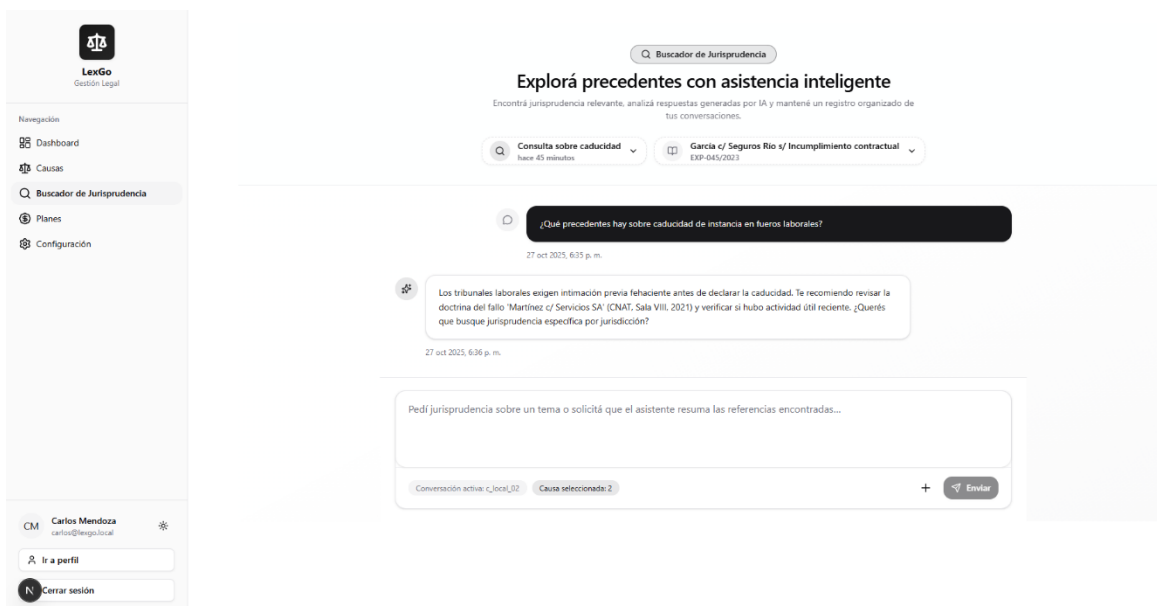


Figura 11. Vista funcional del Asistente. (Elaboración propia, 2025)

Tal como se ilustra en la Figura 11, la vista del Asistente se organiza en tres bloques principales:

- **Interacción libre con selector de causa:** campo de texto donde el usuario puede realizar consultas en lenguaje natural, vinculándolas a un expediente específico mediante un desplegable de selección de causa. Esto garantiza que las respuestas del asistente se enmarquen en el contexto correcto.
- **Historial de consultas:** sección que registra interacciones anteriores, permitiendo retomar conversaciones y mantener coherencia en el análisis de un caso. Además, favorece la trazabilidad de las recomendaciones y el acceso rápido a consultas pasadas.

El diseño de esta vista responde a los requerimientos funcionales **FR-11 (Resumen IA generativa)** y **FR-12 (Asistente conversacional)**, definidos en la sección 6.2. Si bien el esquema gráfico presentado es de carácter conceptual, permite visualizar la modularidad de la interfaz y documentar cómo se materializan en la práctica las funcionalidades innovadoras del sistema.

3.4. Modelo de Datos y Arquitectura del Sistema

En la sección anterior se presentaron los **requerimientos funcionales** de la aplicación, junto con wireframe y diagramas de interacción que ilustran cómo se espera que el sistema sea utilizado por un abogado junior. Ese análisis permitió establecer qué funcionalidades mínimas debe ofrecer la solución y cuál es la experiencia de uso deseada.

En esta sección se aborda el **aspecto técnico del sistema**, respondiendo a dos preguntas centrales:

1. ¿Cómo se modela y organiza la información necesaria para cumplir con los requerimientos?
2. ¿Cómo se despliega e integra esa información en una arquitectura tecnológica eficiente, escalable y de bajo costo?

Para responderlas, primero se describe el **modelo de datos**, que constituye la base estructural sobre la cual se almacenan y relacionan las entidades clave del dominio jurídico (usuarios, causas, partes, eventos y documentos). Este modelo, diseñado bajo criterios de simplicidad y normalización, permite representar la información de forma clara y coherente, garantizando la trazabilidad de cada acción dentro del proceso legal.

En segundo lugar, se introduce la **arquitectura del sistema**, concebida con un enfoque **serverless** y desplegada en la nube (AWS). La elección de esta estrategia técnica responde a los objetivos de bajo costo, alta disponibilidad y facilidad de mantenimiento, sin sacrificar eficiencia y escalabilidad en el corto plazo. Se explicará cómo los distintos servicios de backend y frontend interactúan con la base de datos y con el almacenamiento de documentos, y cómo se organiza el despliegue de los módulos en la nube.

De este modo, el presente capítulo conecta la visión funcional previamente definida con su concreción técnica, mostrando cómo la aplicación propuesta pasa de ser un conjunto de requerimientos a una solución tecnológica factible y sostenible.

3.4.1. Modelo de Datos

El funcionamiento esperado del sistema requiere organizar y relacionar información heterogénea: **usuarios, causas, eventos procesales, documentos y partes** vinculadas. Para dar soporte a estas funcionalidades, se definió un **modelo de datos relacional** que asegura persistencia, coherencia e integridad en el manejo de la información (IBM, [s. f.]).

En este modelo, cada **usuario** puede crear y gestionar sus propias **causas**. A cada causa se le pueden asociar **documentos** y **eventos**. Los documentos funcionan como evidencia o respaldo de los hechos procesales, mientras que los eventos son los hitos que permiten construir la **línea de tiempo** de cada causa. Adicionalmente, se incluyen las **partes**, que representan a los actores jurídicos (personas físicas o jurídicas) involucrados en el caso, y se vinculan con la causa correspondiente.

Este esquema básico garantiza que, a partir de las entidades nucleares, el sistema pueda cubrir las funcionalidades descritas en la sección anterior:

- **Línea de tiempo:** se construye a partir de los eventos registrados en cada causa, ya que incluyen fecha, descripción y vínculos con documentos.
- **Gestión documental:** los documentos se relacionan tanto con la causa como con el usuario que los incorporó, permitiendo un control organizado de escritos, cédulas u oficios.
- **Grafo relacional:** surge de la vinculación entre **partes** y **eventos**, permitiendo representar gráficamente quiénes intervienen en cada hito y cómo se conectan en la causa.

- **Próximos plazos y recordatorios:** se derivan de los eventos y sus fechas, habilitando la construcción de un calendario procesal y notificaciones de vencimientos.

En la Figura 12 se presenta el **modelo de datos simplificado**, que constituye el núcleo mínimo necesario para implementar las funcionalidades principales del prototipo. Dicho esquema incluye las entidades **Usuario, Causa, Parte, Documento y Evento**, además de sus relaciones clave (por ejemplo, la asociación entre evento y documento).

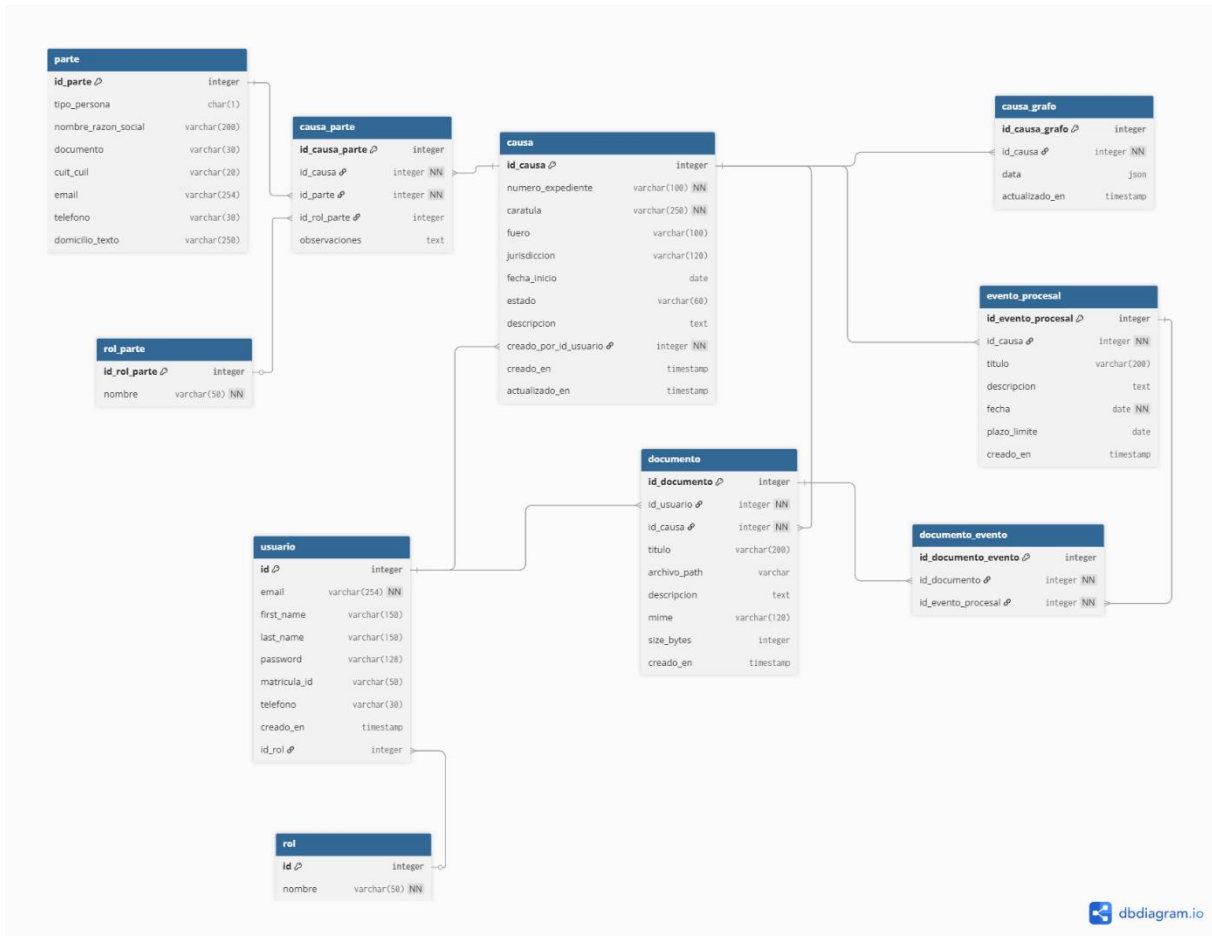


Figura 12. Modelo de datos simplificado del sistema. (Elaboración propia, 2025)

Este modelo inicial está preparado para expandirse en posibles versiones posteriores con entidades de soporte adicionales (como estudios jurídicos, profesionales, tipos de evento o domicilios). Sin embargo, aun en su versión inicial, permite demostrar que todas las funcionalidades principales del sistema pueden apoyarse de manera consistente en una estructura de datos bien definida.

3.4.2. Expansión del Modelo de Datos

El modelo de datos básico presentado en la sección anterior permite cubrir las funcionalidades esenciales del sistema: organización de causas, gestión de documentos, registro de eventos y visualización de partes intervinientes. Sin embargo, para garantizar la **escalabilidad** de la solución y pensar en un escenario de uso más amplio, se diseñó una **expansión del modelo** que incorpora entidades orientadas a contextos profesionales y colaborativos.

Esta ampliación introduce nuevas tablas que permiten representar la dinámica de trabajo en **estudios jurídicos** y su interacción con distintos profesionales. En primer lugar, la entidad **estudio_jurídico** agrupa información general sobre el estudio (nombre, CUIT, teléfono, página web), mientras que la entidad **estudio_usuario** permite vincular a distintos usuarios con un estudio concreto, asignándoles roles, permisos y períodos de vigencia. De esta forma, un mismo sistema puede soportar tanto el uso individual como el trabajo colectivo dentro de una organización legal.

Por otra parte, se incorpora la entidad **profesional**, que representa abogados, procuradores o colaboradores asociados a un estudio, incluyendo datos clave como matrícula, correo electrónico y teléfono de contacto. A través de la tabla **causa_profesional** se establece la relación entre los profesionales y las causas en las que participan, indicando además el rol específico que desempeñan (patrocinante, apoderado, colaborador, etc.). Este diseño habilita que múltiples profesionales puedan intervenir en una misma causa, respetando la lógica real de los estudios jurídicos.

En conjunto, este modelo expandido habilita escenarios de **colaboración entre usuarios**, delegación de responsabilidades y administración centralizada de información a nivel institucional. Aunque no forma parte del alcance inmediato del prototipo inicial, su definición temprana sienta las bases para futuras versiones de la aplicación, orientadas a **estudios jurídicos de mayor escala** y a la construcción de entornos compartidos donde se integren múltiples actores.

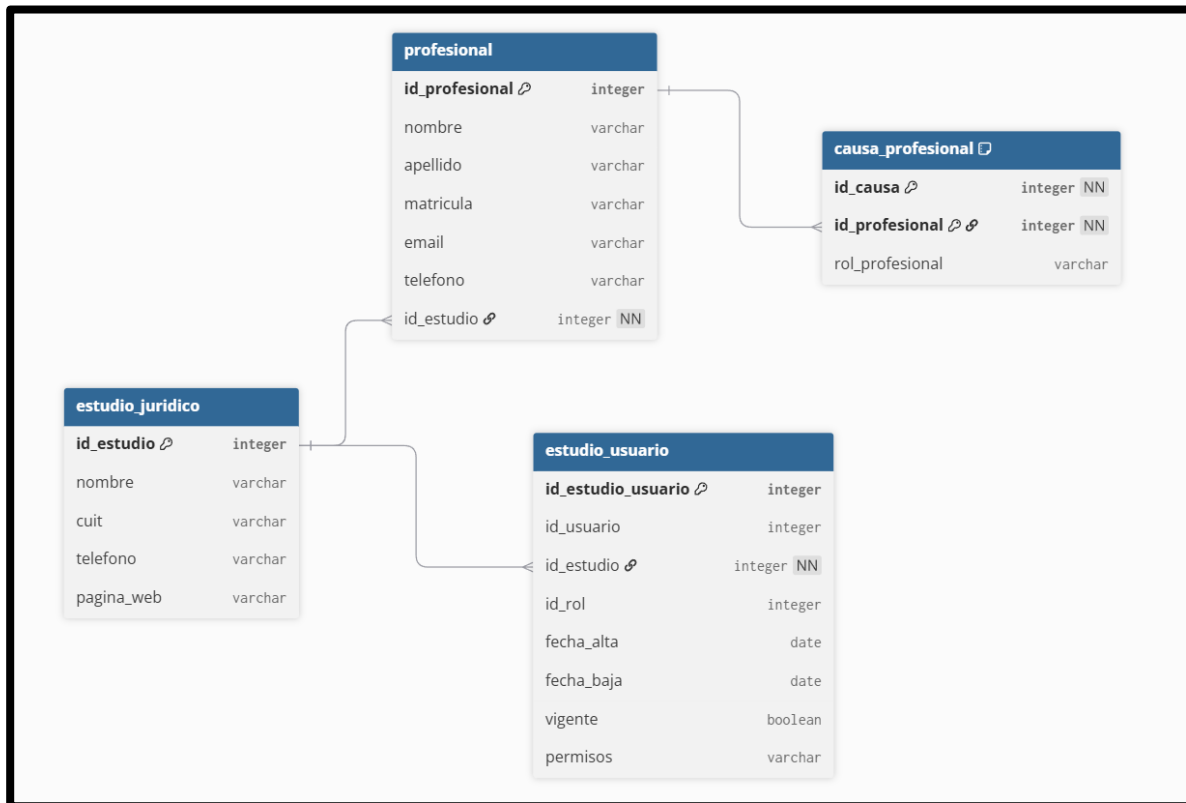


Figura 13. Expansión del modelo de datos. (Elaboración propia, 2025).

3.4.3. Modelo de Datos Integral

El modelo integral combina tanto las entidades nucleares (usuario, causa, evento, documento, parte, plazos y alertas) como las entidades de expansión (estudios jurídicos, profesionales y sus relaciones). De esta forma, se logra una visión completa del dominio de aplicación, que permite cubrir tanto las funcionalidades esenciales del prototipo como los escenarios de colaboración y escalabilidad futura.

En este esquema, los **usuarios** pueden crear causas, cargar documentos y registrar eventos procesales. Las **partes** representan a los actores involucrados en cada caso, mientras que las entidades **plazo** y **alerta** habilitan la gestión de vencimientos y notificaciones. A nivel organizacional, se incorporan **estudios jurídicos** y **profesionales**, con sus correspondientes vínculos a usuarios y causas, permitiendo así modelar entornos colaborativos y estructuras de trabajo más complejas.

La integración de estas entidades en un único diagrama refleja la coherencia y consistencia del diseño de datos, asegurando que cada funcionalidad pueda sustentarse en una base estructural clara y extensible.

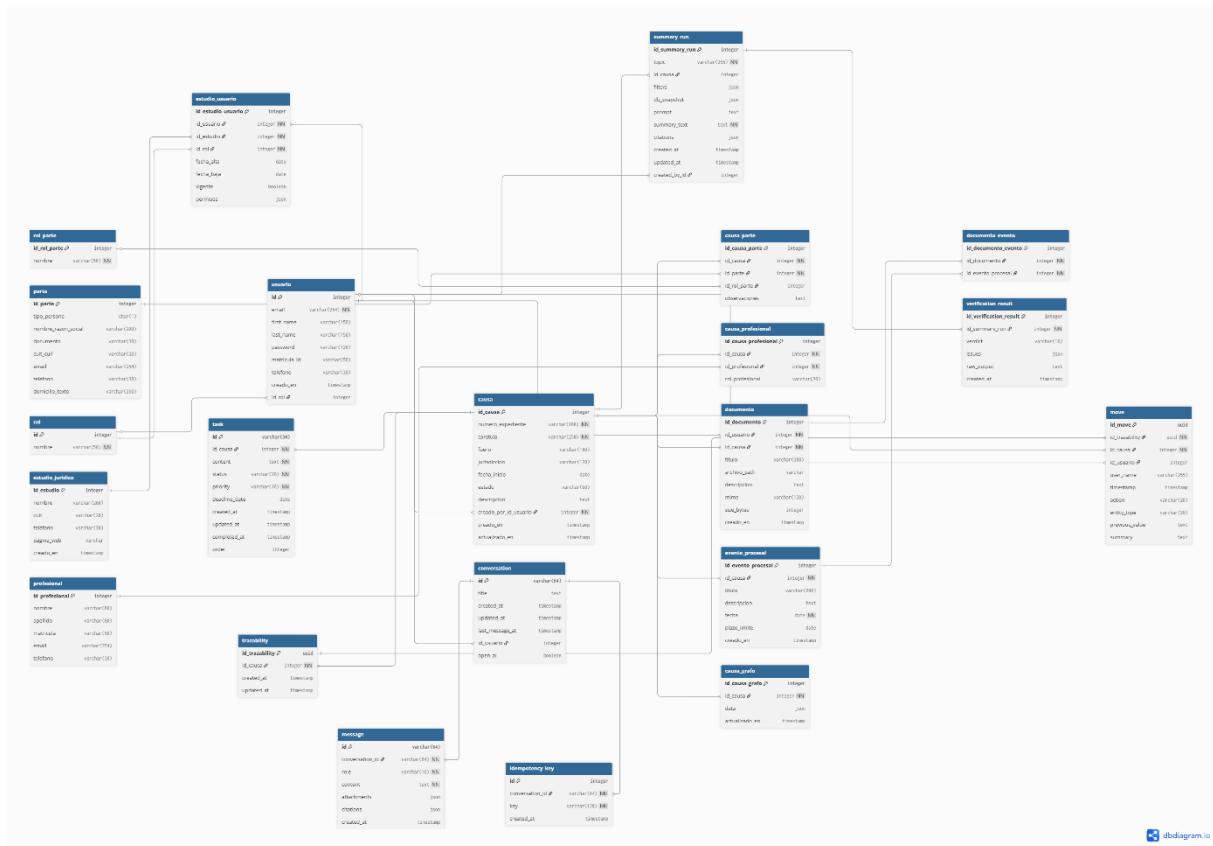


Figura 14. Modelo de datos integral del sistema. Diagrama completo que combina las entidades básicas y de expansión, mostrando cómo se relacionan entre sí para soportar tanto las funcionalidades principales como los escenarios de colaboración a nivel de estudios jurídicos. (Elaboración propia, 2025)

3.5. Arquitectura del Sistema

El diseño de la arquitectura se concibió bajo la premisa de construir un **sistema liviano, simple y eficiente**, capaz de acompañar el crecimiento de la herramienta a medida que se incorporen usuarios y se diversifiquen los patrones de uso.

En las secciones siguientes se diferenciarán dos niveles de arquitectura: la arquitectura efectivamente implementada para el desarrollo y validación del **prototipo (MVP)**, y la arquitectura **proyectada para un entorno de producción**. La primera corresponde a la infraestructura actualmente desplegada, concebida para garantizar el funcionamiento completo de la aplicación, la ejecución de pruebas con usuarios y la validación técnica del sistema bajo condiciones controladas. Esta configuración, de carácter liviano y económico, permite soportar sin inconvenientes un entorno de testing con un número acotado de usuarios simultáneos (entre

50 y 100, según la demanda y las optimizaciones aplicadas). La segunda arquitectura, en cambio, representa la versión ideal del sistema a mediano plazo, diseñada para escalar en capacidad, disponibilidad y resiliencia frente al uso intensivo por parte de múltiples estudios jurídicos o profesionales independientes. Esta proyección futura también constituye la base de referencia sobre la cual se elaboró el modelo económico y de costos del proyecto.

La **naturaleza del sistema** requiere soportar tanto la persistencia de datos estructurados como el manejo de documentos no estructurados. En este sentido, la información principal se almacena en una base de datos relacional, donde se modelan las entidades nucleares definidas en el capítulo anterior (usuarios, causas, eventos, documentos, partes, entre otras). Paralelamente, los archivos vinculados a las causas -escritos, cédulas, oficios o imágenes- se gestionan a través de un almacenamiento en buckets, lo que permite administrar volúmenes variables de información de manera económica y escalable.

En cuanto a la organización general, la arquitectura se divide en dos componentes principales. Por un lado, el **Backend**, que concentra la lógica de negocio y el acceso a los datos. En la versión implementada del MVP, se definen mecanismos de persistencia tanto relacional como documental a través de un servicio de base de datos gestionado, mientras que en la arquitectura proyectada para producción se prevé la adopción de servicios serverless que permitan una mayor elasticidad, disponibilidad y eficiencia en el uso de recursos. El backend expone sus funcionalidades a través de una interfaz pública de tipo API, que constituye el único punto de acceso externo y que, mediante políticas de seguridad y control, asegura la integridad del sistema.

Por otro lado, el **Frontend** representa la capa de interacción con el usuario. Se trata de una aplicación web cuyo código fuente se aloja en un entorno de hosting estático y que es ejecutado en el navegador del cliente. Desde allí, el Frontend se comunica con la API del backend para realizar operaciones sobre causas, eventos o documentos. La interfaz está diseñada con un enfoque responsive y desktop-first, lo cual permite que el sistema pueda utilizarse de manera fluida en escritorio, pero también en dispositivos móviles, respetando el contexto de uso de abogados jóvenes que trabajan de forma dinámica y muchas veces fuera de un escritorio fijo.

A continuación, se presenta la arquitectura desplegada correspondiente al entorno de desarrollo y validación del prototipo funcional (MVP). La arquitectura se diseñó

para mantener una correcta separación entre las capas de aplicación, garantizar la seguridad de los datos y facilitar el despliegue continuo mediante repositorios centralizados en GitHub.

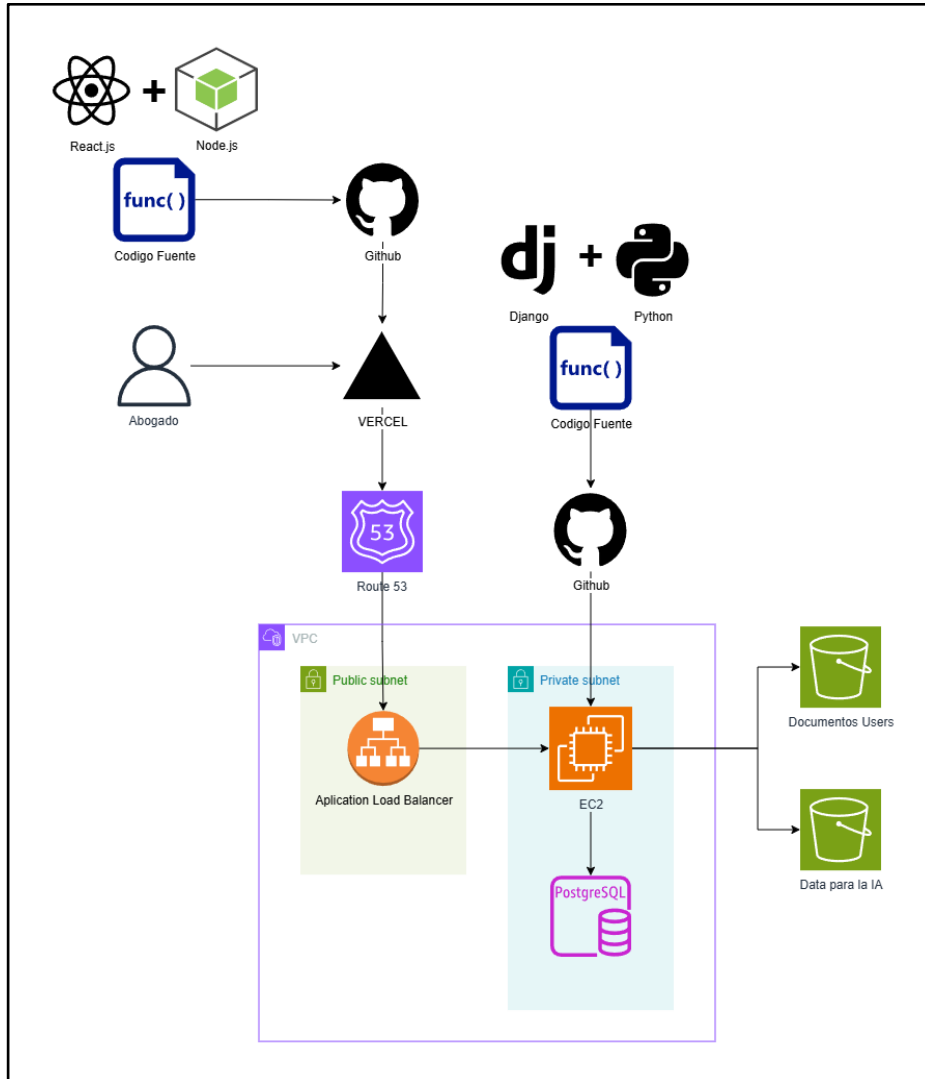


Figura 15. *Arquitectura del sistema implementada para el prototipo funcional (MVP). (Elaboración propia, 2025)*

3.5.1. Arquitectura de la Persistencia

En las secciones previas se describió cómo se modelan los datos del sistema. El paso siguiente consiste en definir dónde y cómo se almacenará esa información de manera confiable, escalable y económica. La elección de la arquitectura de persistencia no solo responde a una necesidad técnica, sino también a la estrategia de negocio: el sistema debe ser ligero en su etapa inicial, pero estar preparado para acompañar un eventual crecimiento en el número de usuarios y en el volumen de información gestionada.

Para el almacenamiento de datos estructurados: entidades como usuarios, causas, eventos, partes, profesionales y estudios jurídicos, se evaluaron distintas alternativas de bases de datos relacionales y no relacionales. La decisión final fue optar por **PostgreSQL v15**, un motor relacional robusto, desplegado dentro del entorno AWS. Esta elección se fundamenta en los siguientes aspectos:

- **Facilidad de despliegue y mantenimiento:** PostgreSQL permite una instalación rápida y estable dentro del entorno AWS, sin requerir configuraciones avanzadas ni costos adicionales de administración. Su integración nativa con servicios gratuitos del ecosistema (como EC2 o RDS en modo free tier) facilita su adopción en entornos de desarrollo y testing.
- **Compatibilidad con el modelo de datos:** la estructura relacional de PostgreSQL se ajusta a la naturaleza jurídica del dominio, que requiere integridad referencial, uniones entre múltiples entidades y consultas complejas sobre causas y eventos.
- **Rendimiento adecuado para pruebas controladas:** en la versión actual, el sistema puede soportar de manera estable un entorno de validación con entre 50 y 100 usuarios simultáneos, manteniendo tiempos de respuesta aceptables y consumo moderado de recursos.
- **Equilibrio entre madurez y disponibilidad:** frente a otras bases de datos gratuitas u open source, PostgreSQL ofrece un ecosistema consolidado, amplia documentación y compatibilidad con librerías ORM modernas, lo que favorece su mantenimiento y escalabilidad futura. raleza del prototipo y los recursos disponibles para un equipo de desarrollo reducido.

Se consideraron alternativas como Amazon RDS (instancias provisionadas) y soluciones no relacionales como DynamoDB. RDS hubiera implicado una infraestructura más rígida, con costos fijos incluso en períodos de baja utilización. DynamoDB, si bien ofrece gran escalabilidad y simplicidad para datos clave-valor, no resultaba adecuado para un modelo jurídico relacional donde las uniones y restricciones de integridad son críticas. En este contexto, PostgreSQL representó la opción más adecuada para la etapa de desarrollo, por su equilibrio entre solidez técnica, accesibilidad y alineación con las necesidades del MVP.

En cuanto al manejo de archivos no estructurados, el sistema utilizará **Amazon S3 (Simple Storage Service)** como repositorio para documentos, escritos y material adjunto a las causas. La elección de S3 responde a características diferenciales:

- **Escalabilidad prácticamente ilimitada**, sin necesidad de planificar capacidad de almacenamiento por adelantado.
- **Bajo costo por GB** y política de pago por uso, lo que resulta ventajoso para un sistema en fase inicial que puede crecer progresivamente.
- **Acceso eficiente y global**, con integración nativa a servicios de entrega de contenido y controles de permisos granulares.
- **Durabilidad del 99.99999999% (11 9's)** garantizada por replicación automática en múltiples instalaciones físicas (Amazon Web Services).
- **Integración sencilla con el backend**, que puede generar URLs firmadas para que los usuarios accedan a sus documentos de forma segura y controlada.

Desde el punto de vista arquitectónico, la base de datos **PostgreSQL** se encuentra **instalada y configurada dentro de una instancia EC2** que aloja los servicios del backend. Esta instancia se ubica en una **subred privada de la VPC**. Esto asegura que las consultas y operaciones sobre datos sensibles se realicen en un entorno controlado, aplicando principios de seguridad de red y aislamiento. Los **buckets de S3**, en cambio, se gestionan como un servicio global, por fuera de la VPC, aunque con políticas de acceso estrictas que restringen su uso exclusivamente al backend mediante credenciales seguras administradas en AWS Secrets Manager.

3.5.2. Backend de Servicios

El backend concentra la lógica de negocio del sistema y actúa como único intermediario entre la interfaz web, la base de datos relacional y el repositorio de documentos. Para este propósito se desarrolló una aplicación **Django (Python)** cuyas fuentes se mantienen en **GitHub** y se despliegan de forma automatizada sobre infraestructura administrada en AWS. Esta decisión prioriza simplicidad operativa, trazabilidad de cambios y un time-to-market reducido, a la vez que asegura una base sólida para escalar.

Flujo de CI/CD

Cada cambio en el repositorio activa un flujo de integración y entrega continua configurado en **GitHub Actions**. El pipeline ejecuta tareas de construcción, pruebas y despliegue automático del backend, garantizando la reproducibilidad y consistencia entre versiones.

Capa de cómputo y despliegue

La arquitectura implementada se apoya en una **instancia Amazon EC2 tipo t3.micro**, que aloja tanto el **servidor de aplicación Django** como la **base de datos PostgreSQL**, desplegados dentro de una **VPC privada**. Esta instancia cuenta con **2 vCPUs virtuales y 1 GB de memoria RAM**, y se encuentra configurada con acceso restringido, sin exposición directa al público. Su comunicación interna con los servicios de **Amazon S3** permite gestionar los documentos y archivos asociados a las causas legales, manteniendo el almacenamiento estructurado en PostgreSQL y los objetos multimedia en S3.

El acceso externo a la aplicación se realiza a través de un **Application Load Balancer (ALB)** ubicado en una subred pública, encargado de canalizar todas las solicitudes entrantes hacia la instancia EC2. El ALB aplica balanceo de carga básico, monitoreo de estado (*health checks*) y terminación **TLS/HTTPS**, garantizando una comunicación segura entre el cliente y el backend.

Esta configuración unificada: backend y base de datos en la misma instancia, resulta adecuada para **entornos de desarrollo y validación del MVP**, gracias a su **bajo costo operativo y simplicidad de mantenimiento**. En pruebas controladas, la instancia soporta de forma estable aproximadamente **20 a 30 usuarios concurrentes activos**, con operaciones mixtas de lectura, escritura y transferencia de documentos. Por encima de ese umbral, el rendimiento comienza a degradarse debido a la limitación de memoria y CPU compartidas entre la aplicación y la base de datos.

Para incrementar la capacidad y estabilidad del entorno, se contemplan distintas estrategias de **escalado progresivo** sobre la infraestructura actual:

- **Escalado vertical:** actualización a instancias de mayor capacidad, como **t3.small (2 GB de RAM)** o **t3.medium (4 GB de RAM)**, lo que permitiría duplicar o triplicar el número de usuarios concurrentes sin modificar la arquitectura lógica.
- **Separación de servicios:** migración de la base de datos a un servicio gestionado, como **Amazon RDS para PostgreSQL**, dejando la instancia EC2 dedicada exclusivamente a la aplicación.
- **Escalado horizontal:** incorporación futura de nuevas instancias de aplicación detrás del ALB para distribuir la carga de trabajo y mejorar la disponibilidad.

Estas alternativas se consideran parte del **plan de evolución técnica** del sistema, preservando la coherencia con la arquitectura actual y permitiendo una transición gradual hacia entornos de mayor concurrencia y resiliencia.

3.5.3. Frontend

La capa de presentación se desarrolló utilizando **React** con el framework **Next.js**, priorizando la modularidad, la reutilización de componentes y la mantenibilidad del código. La estructura del proyecto sigue el esquema de carpetas de Next.js, con una separación clara entre **componentes, contenedores y servicios (services)**.

Los **contenedores (containers)** concentran la lógica de cada vista y se comunican con la **capa de servicios**, la cual abstrae las llamadas a la API y actúa como un **adaptador** entre el frontend y el backend. Esta separación permite aislar la lógica de presentación del acceso a datos, facilitando el mantenimiento, la migración entre entornos y la incorporación futura de nuevas fuentes de información.

El diseño visual se basa en un sistema de **variables centralizadas** que definen la **paleta de colores y los temas de modo claro/oscuro**, garantizando coherencia en la interfaz y una experiencia consistente en distintos dispositivos. Los principales módulos del sistema, como el panel de causas y el dashboard, se estructuran mediante **widgets reutilizables**, lo que permite escalar la interfaz sin duplicar componentes y mantener una arquitectura visual coherente con el resto de la aplicación.

3.5.4. Proceso de Desarrollo

El desarrollo de la aplicación se llevó a cabo de forma **paralela e incremental**, donde el **frontend** y el **backend** se trabajaron de manera coordinada a partir de objetivos comunes. En las etapas iniciales se realizaron **sprints de ideación y diseño**, donde se propusieron **modelos de datos, diagramas preliminares y bocetos de interfaz**.

Durante la fase de planificación se adoptó un enfoque **iterativo**, orientado a la validación temprana de conceptos antes de avanzar hacia el desarrollo. Se comprendió que la planificación debía tener un alcance razonable, evitando la parálisis por análisis y priorizando la evolución constante del prototipo.

Una vez establecida la base conceptual, se definió la **interfaz de comunicación** como **contrato entre el frontend y el backend**. A partir de esta definición se estructuraron los **endpoints de la API**, los **tipos y modelos de datos**, y las **estructuras de respuesta** esperadas. Este acuerdo permitió avanzar en el desarrollo de ambas capas de manera simultánea, asegurando la coherencia en la transmisión de datos y reduciendo los conflictos de integración.

Este enfoque facilitó un proceso de desarrollo más ágil, en el que las modificaciones podían implementarse de forma controlada y predecible, manteniendo la independencia entre las capas y garantizando la escalabilidad del proyecto en futuras iteraciones.

3.5.5. Despliegue en Terraform (IaC)

La infraestructura necesaria para el funcionamiento del backend de la aplicación se define y administra mediante Terraform, siguiendo un enfoque de *Infrastructure as Code* (IaC). Este enfoque permite declarar todos los recursos de Amazon Web Services (AWS) a través de archivos de configuración versionables, reproducibles y auditables. La utilización de Terraform garantiza consistencia entre entornos, reduce errores de configuración y permite automatizar la creación, modificación y eliminación de recursos sin intervención manual sobre la consola de AWS (HASHICORP, 2025).

La arquitectura desplegada mediante Terraform replica de forma declarativa los componentes principales de la plataforma. En primer lugar, se define el proveedor de AWS, la región de operación y las variables necesarias para parametrizar el entorno, tales como el nombre del proyecto, el ambiente, la URL del Frontend y credenciales específicas. Esto permite mantener el archivo modularizado y adaptable a distintos contextos de despliegue (HASHICORP, 2025).

Terraform consulta la Virtual Private Cloud (VPC) predeterminada de AWS y obtiene sus subredes para distribuir los recursos en zonas de disponibilidad distintas, favoreciendo la tolerancia a fallos. Sobre esta red se crean los grupos de seguridad que regulan el tráfico permitido: uno para el Application Load Balancer (ALB), otro para la instancia que ejecuta el backend, y un tercero que restringe el acceso a la base de datos únicamente al backend. Este diseño minimiza la superficie de ataque y delimita correctamente los flujos internos de comunicación (HASHICORP, 2025).

El sistema de almacenamiento se estructura mediante dos buckets de Amazon S3: uno destinado a documentos de los usuarios y otro para datos asociados a futuras funcionalidades de inteligencia artificial. Ambos buckets se crean con reglas CORS orientadas a permitir interacción segura desde el Frontend desplegado en Vercel. Además, se incorpora un sufijo aleatorio mediante el proveedor `random`, asegurando nombres únicos y evitando colisiones con buckets existentes.

Para la capa de persistencia, Terraform declara un motor PostgreSQL administrado en Amazon RDS. Se define su *subnet group* para ubicarlo en las subredes internas previamente seleccionadas, se establecen las credenciales necesarias y se aplican medidas de seguridad tales como evitar el acceso público, habilitar un período de retención de backups y limitar el tráfico entrante al grupo de seguridad del backend. Esta configuración asegura la integridad y disponibilidad del repositorio de datos.

La capa de aplicación se implementa mediante una instancia EC2 que ejecuta el backend desarrollado en Django. Terraform obtiene la Amazon Machine Image (AMI) más reciente de Amazon Linux 2023 y la utiliza para crear la instancia, incorporando un *user data script* que automatiza la descarga del repositorio de GitHub, la instalación de dependencias y la configuración de variables de entorno como los parámetros de base de datos y los nombres de los buckets. La instancia se integra en un *target group* asociado a un Application Load Balancer configurado como punto de acceso público del sistema. El ALB distribuye el tráfico entrante hacia la aplicación mediante reglas definidas en el archivo, y se establece una política de health checks que permite monitorear la disponibilidad del servicio.

Finalmente, Terraform expone *salidas (outputs)* que permiten recuperar información relevante para otros componentes del proyecto, tales como la URL del backend, los endpoints de la API y las variables necesarias para configurar el ambiente del Frontend. Estas salidas facilitan la integración entre módulos y reducen la configuración manual en fases de despliegue.

3.5.6. Uso de Ngrok para habilitar tráfico HTTPS

Durante el proceso de despliegue se identificó una limitación estructural derivada del entorno AWS Student Lab utilizado para este proyecto. El backend se ejecuta en una instancia EC2 detrás de un Application Load Balancer (ALB) configurado exclusivamente sobre HTTP, debido a que el entorno académico no permite la creación de certificados TLS mediante AWS Certificate Manager (ACM) ni la integración con CloudFront. Esta restricción impide habilitar tráfico HTTPS hacia el backend sin la posesión de un dominio propio, condición necesaria para validar certificados y establecer conexiones seguras entre clientes y servidores.

El frontend se encuentra desplegado en Vercel, una plataforma que opera estrictamente bajo HTTPS. Los navegadores modernos bloquean cualquier intento de

comunicación entre un origen seguro (HTTPS) y un recurso expuesto únicamente en HTTP, aplicando la política de mixed content. Como resultado, el frontend no puede realizar solicitudes hacia el backend a través del ALB, aun cuando el backend funcione correctamente dentro de AWS. Este conflicto constituye un impedimento técnico relevante para la integración entre ambas capas de la aplicación.

Ante esta limitación, se adopta Ngrok como mecanismo temporal para exponer el backend mediante una URL pública con HTTPS. Ngrok es una herramienta que establece un túnel seguro entre un servicio local y un punto de acceso accesible desde Internet (NGROK INC., 2025). Al ejecutarse sobre la misma instancia EC2 que aloja el backend, Ngrok intercepta el tráfico entrante, gestiona el cifrado TLS y reenvía las solicitudes HTTP hacia el servicio interno que se ejecuta en el puerto 8000. El funcionamiento general puede expresarse del siguiente modo:

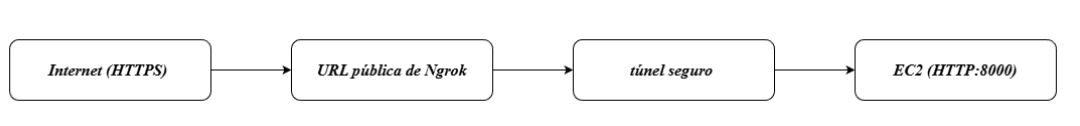


Figura 16. *Funcionamiento general de Ngrok.* (NGROK INC., 2025).

De esta manera, la aplicación obtiene una URL pública provista por Ngrok, completamente compatible con los requisitos del frontend en Vercel. Ngrok actúa como un terminador TLS externo al ecosistema de AWS, lo que permite sortear la necesidad de certificados y dominios, y evita el bloqueo por mixed content sin modificar la infraestructura original del proyecto (NGROK INC., 2025).

La utilización de Ngrok se limita a los entornos de desarrollo y demostración, dado que su versión gratuita presenta restricciones, como la rotación periódica de las URLs y desconexiones tras períodos de inactividad. No obstante, su incorporación resulta adecuada en este contexto académico, dado que permite validar la funcionalidad de la arquitectura, garantizar la comunicación segura entre frontend y backend, y presentar un prototipo operativo sin requerir recursos adicionales no disponibles en el entorno de AWS asignado.

3.5.7. Arquitectura Ideal

La arquitectura proyectada para un entorno de producción incorpora **servicios administrados de AWS** con el objetivo de optimizar la **escalabilidad**, la **disponibilidad** y la **eficiencia operativa** del sistema. Está dimensionada para soportar de forma estable un

volumen de entre 500 y 1000 usuarios concurrentes, con capacidad de **escalado automático** ante incrementos sostenidos de carga. Este diseño permite acompañar el crecimiento progresivo de la plataforma sin necesidad de rediseñar la infraestructura. Los detalles de esta arquitectura pueden encontrarse en Anexo H.

3.6. Modelo de Inteligencia Artificial

El componente de Inteligencia Artificial (IA) desarrollado para la presente aplicación constituye un módulo independiente dentro de la arquitectura general del sistema, siguiendo los principios de **Clean Architecture** y las recomendaciones para *AI-enhanced software systems* (Amershi et al., 2019).

Su propósito es asistir al usuario en tareas cognitivas del proceso legal mediante la generación automática de resúmenes, detección de omisiones y errores, así como la recomendación de jurisprudencia relevante en causas del derecho penal argentino.

Además, se incorpora la opción de realizar búsquedas extendidas en internet mediante la API de Tavily, exclusivamente cuando el usuario lo solicita de manera explícita, ampliando el alcance del sistema a información pública disponible en línea.

3.6.1. Tavily

Tavily es un motor de búsqueda orientado a aplicaciones de inteligencia artificial, diseñado para proporcionar respuestas precisas, verificables y basadas en fuentes reales. A diferencia de los buscadores tradicionales, Tavily está optimizado para integrarse con modelos de lenguaje y sistemas de Recuperación Aumentada por Generación (RAG), permitiendo obtener información actualizada, confiable y acompañada de referencias explícitas (Tavily, s.f.).

Su funcionamiento combina técnicas de crawling, indexación semántica y ranking de relevancia, lo que permite recuperar información de múltiples dominios públicos y devolver resultados ya estructurados para ser consumidos por una aplicación backend o un modelo de IA (Tavily, s.f.).

3.6.2. Justificación de la elección del modelo

Durante la fase exploratoria se evaluaron distintas alternativas de procesamiento del lenguaje natural (NLP), incluyendo modelos open-source como **LLaMA 3**, **Falcon** y **Mistral**, además de opciones propietarias.

La elección final de **GPT (Generative Pre-trained Transformer)** se fundamentó en tres criterios técnicos y cognitivos:

1. **Capacidad lingüística contextual:** GPT demostró un rendimiento superior en la comprensión de textos jurídicos en español, permitiendo mantener la coherencia entre párrafos extensos y adaptarse a terminología legal sin entrenamiento adicional.
2. **Versatilidad en tareas múltiples:** a diferencia de modelos especializados en clasificación o extracción, GPT puede realizar simultáneamente resúmenes, correcciones gramaticales y búsqueda de jurisprudencia semántica.
3. **Madurez tecnológica e integración API:** su disponibilidad mediante servicios REST seguros y escalables (OpenAI API / Azure OpenAI) facilitó su integración en la arquitectura backend del sistema, sin comprometer la privacidad ni el rendimiento.

Adicionalmente, la integración opcional con Tavily API permite complementar la base jurídica local con información actualizada disponible en la web, restringida a consultas en las que el usuario solicita explícitamente una "búsqueda extendida". Esta arquitectura híbrida provee un equilibrio entre recuperación estructurada (corpus local) y exploración abierta (búsqueda web dirigida).

De esta forma, GPT fue implementado como un *microservicio externo* accesible desde los *use cases* específicos del backend (por ejemplo, "Resumir Causa" o "Corregir Documento"), garantizando independencia modular y mantenibilidad (Amazon Web Services, s.f.).

3.6.3. Funcionalidades implementadas

El módulo IA se integra a través de tres funciones principales:

- **Resumen automático de causas:** Permite condensar textos procesales extensos (demandas, resoluciones, escritos) en párrafos sintéticos que mantienen las relaciones causales y procesales relevantes. El modelo

analiza los documentos almacenados en el bucket de S3, genera el resumen en lenguaje claro y lo devuelve como texto estructurado en JSON, siguiendo el esquema de metadatos diseñado para cada documento (.meta.json).

- **Corrección gramatical y de estilo jurídico:** Implementa un pipeline de revisión lingüística que identifica errores ortográficos, eufónicos y de puntuación, proponiendo mejoras en estilo sin alterar el contenido legal. El modelo detecta omisiones (como falta de conectores lógicos o de puntuación) y sugiere reformulaciones coherentes con el lenguaje jurídico formal.
- **Búsqueda y recomendación de jurisprudencia:** A partir del resumen generado o de palabras clave extraídas, GPT consulta una base documental local de fallos argentinos (almacenada en el bucket “jurisprudencia”) y devuelve los casos más similares semánticamente. Esta búsqueda se basa en *embeddings vectoriales* y *similaridad de coseno*, técnica ampliamente utilizada en sistemas de recuperación semántica (Reimers y Gurevych, 2019).
- **Búsqueda extendida mediante TAVILY:** Cuando el usuario activa explícitamente la opción de “búsqueda extendida”, el sistema envía la consulta jurídica al motor de búsqueda Tavily, que permite: obtener resúmenes de artículos, doctrinas o notas periodísticas relevantes, verificar si existen precedentes comentados fuera del corpus local, complementar el análisis con fuentes públicas de internet sometidas a filtros estrictos de dominio. Esta búsqueda extendida solo se ejecuta bajo requerimiento explícito del usuario para preservar trazabilidad y evitar interferencias con el corpus jurídico estructurado

3.6.4. Funcionamiento técnico

El flujo técnico del módulo IA se estructura en las siguientes etapas:

1. **Ingesta de datos:** los documentos procesales se almacenan en formato PDF en Amazon S3. Cada archivo cuenta con un metadato. meta.json que describe su tipo, fecha, partes involucradas y estado procesal.

2. **Extracción y preprocesamiento:** el backend utiliza Amazon Textract para convertir el texto del documento en formato legible por el modelo, eliminando encabezados y firmas redundantes.
3. **Procesamiento IA:** el servicio Django llama al endpoint `/api/causas/ia-summary/`, que envía el texto a GPT mediante una *prompt chain* especializada. Si el usuario solicita “Búsqueda Extendida”, se agrega una llamada secundaria a Tavily API antes de la generación final de la respuesta.
4. **Verificación y almacenamiento:** el resultado se evalúa mediante reglas heurísticas y se guarda en la base de datos, vinculado al documento original.
5. **Revisión manual:** el usuario puede aceptar, editar o descartar la salida del modelo, asegurando control humano sobre el resultado.

Esta arquitectura “**IA-as-a-Service**” permite que el módulo opere de manera asíncrona y escalable, sin interferir con las operaciones principales del sistema, y es compatible con integraciones vía AWS Bedrock o Hugging Face Inference API.

3.6.5. Asistente de Jurisprudencia Basado en Recuperación Aumentada (RAG)

El módulo Asistente de Jurisprudencia constituye una de las funcionalidades centrales del sistema, al posibilitar consultas jurídicas formuladas en lenguaje natural y la obtención de respuestas contextualizadas, acompañadas de citas verificables provenientes de legislación, doctrina o jurisprudencia argentina. El desarrollo de este componente responde a la necesidad de automatizar la búsqueda y síntesis de información jurídica, integrando fuentes normativas con inteligencia artificial generativa sin comprometer la trazabilidad ni la precisión legal (Fundar, 2023).

El módulo se integró dentro de la arquitectura general bajo un enfoque API REST + LLM Integration, utilizando Django REST Framework, servicios externos de IA y procesos automatizados con Selenium y búsqueda distribuida mediante Tavily cuando se habilita la función de búsqueda ampliada. Los componentes principales pueden observarse en Anexo I.

El funcionamiento del asistente sigue una secuencia estructurada de etapas. Primero, el usuario inicia o continúa una conversación identificada por un `conversation_id` único vinculado al token JWT. La solicitud se valida mediante el `AskJurisRequestUnionSerializer`, que registra el mensaje y sus metadatos en el modelo `Message`.

Luego, se ejecutan las funciones de recuperación (`search_chunks_strict` y `search_chunks`), que localizan fragmentos jurídicos relevantes según fuero, jurisdicción, tribunal y rango temporal (si se especificara). Tras la recuperación local de ambas funciones, si el usuario solicita búsqueda extendida el backend ejecuta una consulta paralela a Tavily, filtra los dominios permitidos, incorpora las referencias relevantes al prompt final y marca en el JSON qué fragmentos provienen de web externa; en caso contrario el asistente responde exclusivamente con bases locales y fuentes verificables. Esto permite mantener la precisión jurídica y, al mismo tiempo, ofrecer la posibilidad de ampliar el contexto cuando la consulta lo requiera.

Los textos recuperados se combinan con la pregunta del usuario para construir un prompt contextual, que se envía al modelo GPT-4o con baja temperatura (0,1) para asegurar coherencia y precisión técnica. La respuesta generada se almacena junto a las citas correspondientes en el modelo `Message` con rol `assistant`.

Cada respuesta incluye referencias en formato JSON estructurado (por ejemplo, “Ley 20.744 - Régimen de Contrato de Trabajo”), cuyas URLs se validan mediante un filtro de dominios autorizados. Finalmente, se actualizan los campos `updated_at` y `last_message_at` en el modelo `Conversation`, lo que permite ordenar las consultas por actividad reciente.

3.6.6. Modelo de Machine Learning

Con el objetivo de habilitar la creación automática de causas a partir de documentos judiciales, se diseñó e implementó un pipeline de procesamiento que integra técnicas de extracción de texto, clasificación supervisada y generación asistida mediante modelos de lenguaje de gran escala (LLMs). Este flujo permite transformar un archivo PDF de origen judicial en una representación estructurada compuesta por una línea de tiempo preliminar y sugerencias automatizadas para el análisis del caso. El proceso completo se articula en las siguientes etapas: extracción de texto, clasificación mediante modelos de machine learning, enriquecimiento semántico mediante un LLM y generación de timeline y alertas, lo que

conforma un mecanismo integral para asistir al profesional jurídico desde el inicio del expediente.

El flujo implementado sigue la secuencia: la extracción de texto se realiza sobre documentos judiciales en formato PDF, dando lugar a un cuerpo textual homogéneo sobre el cual opera el clasificador. El clasificador supervisado, entrenado en Python con la biblioteca scikit-learn, categoriza el documento según la etapa procesal a la que pertenece. Posteriormente, un modelo de lenguaje (OpenAI) analiza el texto ya clasificado, genera una estructura inicial del caso y produce una línea de tiempo preliminar acompañada de sugerencias inteligentes basadas en patrones frecuentes del fuero laboral.

3.6.7. Construcción del conjunto de datos

Para entrenar el clasificador supervisado se elaboró un conjunto inicial de datos compuesto por 50 documentos correspondientes a diferentes etapas posibles de la causa, con palabras clave para la correcta identificación del modelo. Cada documento fue revisado manualmente y etiquetado según la etapa procesal predominante mencionada en el texto. Las categorías definidas fueron:

- seclo: referencias a conciliación previa, audiencias de mediación laboral, certificados habilitantes.
- demanda inicial: indicación de demanda, traslado, contestaciones.
- prueba: menciones a apertura a prueba, producción de testimonios o pericias.
- sentencia: presencia de secciones decisorias, como “RESUELVO” o la parte dispositiva.
- desconocido: documentos cuya etapa no pudo identificarse con claridad.

3.6.8. Entrenamiento supervisado

Una vez concluido el etiquetado, se procedió a la preparación del conjunto de datos mediante:

- limpieza de texto.
- tokenización y vectorización TF-IDF.
- división en subconjuntos de entrenamiento y prueba (train/test split).

El modelo seleccionado fue un clasificador de Random Forest, debido a su robustez y capacidad para manejar conjuntos de datos pequeños sin requerir un ajuste extenso de hiperparámetros. El entrenamiento se realizó íntegramente en Python utilizando scikit-learn.

La evaluación del modelo incluyó:

- matriz de confusión.
- accuracy.
- recall por clase (especialmente relevante por la naturaleza desequilibrada entre categorías procesales).
- análisis de errores para retroalimentar mejoras del pipeline.

Una vez clasificado el documento, se integró un modelo LLM encargado de:

- interpretar el contenido.
- generar una estructura preliminar del caso.
- identificar actores, hechos relevantes y fechas.
- producir una línea de tiempo automática.
- elaborar sugerencias o advertencias sobre omisiones o puntos a revisar.

Este componente complementa el clasificador supervisado, aportando análisis contextual y generativo para facilitar la comprensión del expediente por parte del usuario final.

Este proceso permitió construir un módulo híbrido basado en ML y LLM que convierte documentos judiciales en estructuras comprensibles, generando automáticamente la etapa procesal, la línea de tiempo y un conjunto de sugerencias que potencian el análisis del expediente.

3.6.9. Resultados del modelo

La presente sección expone los resultados obtenidos durante el entrenamiento y la evaluación del modelo de clasificación diseñado para identificar la etapa procesal a la que pertenece cada documento jurídico. El análisis incluye métricas cuantitativas derivadas del rendimiento del modelo, como accuracy, F1-Score y matriz de confusión, así como una interpretación cualitativa de los aciertos y errores observados.

La matriz de confusión generada resume el desempeño del modelo de clasificación encargado de asignar cada documento procesal a una de las etapas del proceso judicial: demanda_inicial, prueba, secló y sentencia. Cada fila representa la etapa real, mientras

que cada columna indica la etapa predicha por el modelo. La diagonal principal refleja los aciertos, mientras que los valores fuera de ella representan errores de clasificación.

Los resultados obtenidos indican lo siguiente:

- **demanda_inicial:** el modelo clasificó correctamente los 3 casos pertenecientes a esta categoría. No se observaron confusiones con otras etapas, lo que sugiere que los patrones lingüísticos de esta fase son suficientemente distintivos para el clasificador.
- **prueba:** los 2 documentos reales de etapa de prueba fueron correctamente clasificados. Nuevamente, no se registraron errores, lo cual evidencia que el modelo identifica adecuadamente esta etapa a partir de los fragmentos textuales proporcionados.
- **seclo:** el modelo predijo correctamente 1 caso, correspondiente a la categoría real. Esto demuestra cierto nivel de diferenciación, aunque se requiere un conjunto de datos mayor para validar su estabilidad.
- **sentencia:** se observa un caso en el que un documento de sentencia fue erróneamente clasificado como prueba. Este resultado evidencia una superposición semántica entre ambas categorías, posiblemente asociada al uso de terminología jurídica común (resoluciones, valoración probatoria, considerandos). Este tipo de confusión es habitual en modelos entrenados con datasets reducidos.

En términos generales, la matriz refleja un buen desempeño inicial del clasificador, con una mayoría de predicciones correctas y errores aislados. No obstante, su efectividad se encuentra directamente condicionada por el tamaño reducido del conjunto de entrenamiento.

Los resultados completos se pueden observar en Anexo J.

3.6.10. Decisiones metodológicas en la construcción del modelo

La elaboración del modelo de clasificación supervisada y del pipeline de procesamiento documental requirió seleccionar técnicas y enfoques acordes al alcance del proyecto, a las restricciones del dominio jurídico y al tiempo disponible para la experimentación. En este contexto, las decisiones metodológicas adoptadas se fundamentan

tanto en criterios técnicos como en consideraciones prácticas, garantizando un equilibrio entre rigor académico, viabilidad operativa y alineación con los objetivos del sistema.

En primer lugar, la elección de un enfoque supervisado basado en la clasificación de etapas procesales responde a la necesidad de contar con un mecanismo capaz de orientar la posterior intervención del modelo generativo. Dado que la estructura de un expediente judicial posee hitos identificables (conciliación, demanda, prueba, sentencia), la categorización previa facilita que el LLM genere líneas de tiempo más precisas y relevantes. En este sentido, la definición de cinco clases discretas constituye un punto de partida razonable para capturar la progresión procesal sin introducir una complejidad excesiva para el tamaño del dataset.

La selección de TF-IDF como método de vectorización se fundamenta en su efectividad comprobada en escenarios donde los documentos son extensos, poseen vocabulario jurídico característico y el volumen de datos disponibles es limitado. Técnicas más recientes, como embeddings basados en transformadores, presentan ventajas semánticas, pero requieren conjuntos de datos más amplios y tiempos de entrenamiento significativamente mayores (Khan et al., 2025). En contraste, TF-IDF ofrece interpretabilidad, simplicidad y un desempeño competitivo en tareas de clasificación textual tradicional, especialmente cuando los textos incluyen términos distintivos como los presentes en resoluciones judiciales.

En cuanto al algoritmo, la elección de Random Forest responde a criterios de robustez, baja sensibilidad al ruido y buen rendimiento en datasets pequeños. A diferencia de modelos lineales o de redes neuronales, Random Forest permite manejar variabilidad entre textos, evita sobreajuste mediante el uso de múltiples árboles y no requiere ajustes hiperparamétricos complejos. En el contexto del proyecto, donde el objetivo del clasificador es proporcionar una etiqueta preliminar más que una decisión definitiva, este modelo ofrece un balance adecuado entre simplicidad y eficacia.

La decisión de trabajar con un dataset inicial de 50 documentos etiquetados manualmente se vincula con la disponibilidad de tiempo y con la necesidad de artefactos académicos verificables dentro del proyecto. El etiquetado manual garantiza consistencia conceptual, algo especialmente relevante en textos jurídicos donde los límites entre categorías pueden ser sutiles. Aunque el tamaño reducido limita la capacidad de generalización del modelo, resulta suficiente para construir un prototipo funcional y para generar métricas evaluables dentro del componente académico del trabajo. Además, este enfoque sienta las bases para futuras ampliaciones del conjunto de datos.

Finalmente, la integración del clasificador supervisado con un modelo de lenguaje (LLM) constituye una decisión estratégica orientada al valor práctico del sistema. El clasificador actúa como un mecanismo de orientación que reduce la ambigüedad inicial del documento, mientras que el LLM aporta análisis semántico profundo y generación de estructura narrativa. Esta división de responsabilidades permite aprovechar las fortalezas de cada tecnología: el ML tradicional ofrece velocidad, reproducibilidad y control; el LLM brinda flexibilidad interpretativa y capacidad generativa. El resultado es un pipeline híbrido que maximiza la utilidad para el usuario final sin comprometer la claridad metodológica del componente académico.

3.7. Identidad de marca

El nombre *LexGo* surge de la combinación de *Lex* (término latino que significa “ley”) y la expresión inglesa *Go*, asociada a dinamismo y acción. El resultado refleja una marca moderna y accesible que invita a “poner en marcha el derecho”, simbolizando la transición del estudio teórico a la práctica profesional.

Personalidad de marca:

LexGo se concibe como una herramienta confiable, clara y ágil, diseñada para acompañar a los abogados en sus primeros pasos profesionales. Su identidad transmite tres valores centrales: **claridad** (facilitar la comprensión de los procesos judiciales), **autonomía** (empoderar al profesional joven para organizar su trabajo sin depender de terceros) e **innovación** (introducir visualización y razonamiento asistido por IA en el ámbito jurídico argentino).

Identidad visual:

La estética se apoya en una **paleta azul-gris, negro profesional**, que comunica seriedad y confianza, complementada con acentos en **turquesa y blanco** para resaltar interactividad y frescura. La tipografía seleccionada es moderna y legible, con predominio de líneas limpias y sans-serif. La interfaz mantiene un estilo *minimalista*, priorizando la lectura rápida de eventos y relaciones.

Coherencia con la propuesta de valor:

La marca refleja la estrategia de *Océano Azul* adoptada por el proyecto: apartarse del paradigma del software de gestión documental (como *Lex-Doctor*) para posicionarse como

una plataforma de **pensamiento visual y analítico para abogados**, centrada en la comprensión de causas y no en la mera administración de expedientes.

3.8. Modelo económico

Una vez definida la arquitectura del sistema, resulta fundamental analizar la viabilidad económica del proyecto. La estructura técnica elegida, basada en servicios cloud y arquitectura serverless, permite optimizar costos al pagar únicamente por uso efectivo de recursos. En esta sección se evalúa el modelo económico, estimando los costos asociados, las fuentes de ingresos y los indicadores financieros que determinan la rentabilidad del proyecto.

3.8.1. Metodología del análisis económico

El análisis económico de este proyecto parte del reconocimiento de que se trata de una **solución SaaS (Software as a Service)**, es decir, un producto digital que se ofrece de manera continua a través de la nube y cuyo modelo de negocio se basa en suscripciones periódicas. Este enfoque implica que la estructura de costos y la generación de ingresos se encuentran **directamente relacionadas con el uso real de la plataforma**, y no con la venta de licencias únicas o la adquisición de infraestructura física.

En este contexto, la naturaleza **serverless** de la arquitectura, sustentada en servicios gestionados de Amazon Web Services (AWS) y en el hosting de Vercel, resulta determinante para la economía del sistema. Los costos principales provienen del **uso de cómputo, almacenamiento y transferencia de datos**, en lugar de equipamiento o mantenimiento. De este modo, el proyecto adopta un modelo altamente escalable y eficiente, en el cual los gastos crecen únicamente en función del número de usuarios activos y del volumen de operaciones realizadas.

El análisis económico se estructura con el propósito de evaluar la viabilidad financiera del proyecto en distintos escenarios de crecimiento, contemplando una evolución temporal coherente con las etapas técnicas del desarrollo. Para ello, se consideran tres situaciones representativas:

- Un escenario inicial de **100 usuarios activos**, correspondiente a la etapa de MVP
- Un escenario intermedio de **1.000 usuarios**, asociado al crecimiento y consolidación del producto.

- Y hasta un hipotético caso de **10.000 usuarios**, como escenario **extremo**, con el objetivo de justificar y proyectar un crecimiento explosivo de la plataforma.

Cada uno de estos escenarios permite observar cómo se comportan los costos fijos (independientes del número de usuarios) y los costos variables o escalables (proporcionales al uso del sistema), generando una visión integral sobre la sustentabilidad del modelo.

Finalmente, el análisis proyecta los flujos de caja esperados a lo largo de un horizonte de cinco años, estimando los indicadores financieros más relevantes -entre ellos el Valor Actual Neto (VAN), la Tasa Interna de Retorno (TIR) y el punto de equilibrio- con el fin de demostrar la viabilidad técnica y económica de la solución.

3.8.2. Etapas de desarrollo del producto

El análisis económico del proyecto se apoya en una visión temporal del negocio, estructurada en tres grandes etapas que acompañan el proceso natural de maduración de una solución SaaS. Cada una de ellas presenta un nivel distinto de inversión, riesgo y potencial de retorno, en consonancia con el grado de desarrollo tecnológico y de validación comercial alcanzado.

3.8.2.1. Etapa I: Desarrollo del MVP (primer año)

Durante el primer año se lleva adelante la construcción del **Producto Mínimo Viable (MVP)** y la validación técnica inicial de la plataforma. En esta etapa no se prevén ingresos, dado que se considera un **período financiado por los propios desarrolladores**, con una inversión estimada de alrededor de 4.000 USD correspondiente al uso de recursos de cómputo y almacenamiento en la nube.

Este primer año tiene por objetivo lograr una versión plenamente funcional del sistema, capaz de ser utilizada por un grupo reducido de usuarios en condiciones reales. Para facilitar la adopción inicial, se evalúan dos posibles mecanismos de acceso:

- Un **sistema de créditos o cupos gratuitos**.
- Un **modelo de suscripción con período de prueba limitado**.

Si bien ambos esquemas son válidos, se opta por priorizar el enfoque de **suscripción con prueba gratuita**, ya que contribuye a establecer desde el inicio una relación clara entre el usuario y el valor percibido del servicio. En experiencias previas, los sistemas

basados en créditos suelen generar resistencias al momento de la conversión a pago, mientras que un modelo transparente de suscripción facilita la comunicación y la percepción de profesionalismo.

En esta primera etapa, se proyecta una **base de aproximadamente 100 usuarios**, que permitirá obtener información empírica sobre patrones de uso, rendimiento y satisfacción, sin comprometer la estabilidad financiera del proyecto.

3.8.2.2. Etapa II: Consolidación del producto (segundo año)

La segunda etapa corresponde al proceso de **consolidación tecnológica y validación comercial**.

El software ya cuenta con dos años acumulados de desarrollo -el primero técnico y el segundo enfocado en la experiencia del usuario-, por lo que se espera disponer de una versión madura, estable y con funcionalidades avanzadas, incluyendo la integración de componentes de inteligencia artificial.

En esta fase se inicia la **monetización progresiva** del sistema. Los usuarios que participaron en la etapa inicial (early adopters) podrán acceder a planes promocionales, con tarifas preferenciales o descuentos por fidelidad. Este modelo busca reconocer su rol en el proceso de prueba y construcción del producto, al tiempo que genera ingresos iniciales que contribuyen a cubrir la inversión del año anterior.

Asimismo, se contemplan **dos escenarios posibles**:

- Una **reducción parcial de la base de usuarios**, donde parte de los testers no continúan pero los usuarios activos comienzan a pagar.
- O, un **crecimiento moderado de la base**, manteniendo o superando los 100 usuarios, pero ahora bajo modalidad de suscripción paga.

En ambos casos, el proyecto ingresa en una fase de equilibrio operativo: los costos de infraestructura continúan siendo bajos y los ingresos obtenidos permiten comenzar a recuperar la inversión inicial. El foco principal de este año se centra en la calidad del servicio, el soporte al cliente y la optimización de procesos.

3.8.2.3. Etapa III: Expansión y escalabilidad (a partir del tercer año)

La tercera etapa marca el inicio del **crecimiento sostenido** del proyecto. Con un producto sólido, una base de usuarios reales y un conjunto de funcionalidades diferenciadoras, se espera un aumento gradual de la adopción.

En esta fase se incorporan **estrategias de marketing digital**, acciones de posicionamiento y, eventualmente, **gastos administrativos** asociados a la gestión y soporte de una comunidad creciente. Se prevé un incremento constante de la base de usuarios, que podría alcanzar entre **500 y 1.000 usuarios al quinto año**, incluyendo potenciales clientes **B2B (Business to Business)**, como estudios jurídicos o instituciones que requieran licencias múltiples.

Este crecimiento está sustentado en un modelo de precios accesible (entre 5 y 7 USD mensuales por usuario) y en una infraestructura escalable de costos controlados. La combinación de bajo costo operativo y expansión progresiva permite proyectar márgenes de rentabilidad favorables a partir del tercer año.

En términos económicos, esta etapa representa el punto en el que la relación **ingresos/costos** se consolida, y el proyecto comienza a generar **flujos de caja positivos sostenidos**, alcanzando la **recuperación total de la inversión inicial** y configurando una base sólida para futuras fases de expansión o alianzas estratégicas.

3.8.3. Costos fijos y costos variables

La arquitectura técnica adoptada en el proyecto tiene un correlato directo en su estructura económica. El modelo serverless y basado en servicios gestionados de la nube permite que los costos se distribuyan en dos grandes categorías: costos fijos, que garantizan el funcionamiento continuo del sistema, y costos variables, que dependen del nivel de uso y de la cantidad de usuarios activos. Esta distinción resulta clave para entender la elasticidad económica del proyecto y su capacidad de escalar de manera eficiente sin requerir grandes inversiones iniciales.

3.8.3.1. Costos fijos

Los costos fijos corresponden a aquellos servicios indispensables para el funcionamiento básico del sistema, cuya magnitud no varía significativamente con el crecimiento de la base de usuarios. Estos gastos aseguran la operatividad permanente de la solución, la seguridad de acceso y la continuidad del ciclo de desarrollo:

- **Vercel Pro (hosting del frontend y despliegue continuo):** servicio que aloja la interfaz web de la aplicación y ofrece un pipeline de integración continua (CI/CD) automatizado, con un costo mensual estable.
- **Dominio y servicio de DNS (Amazon Route 53):** registro del dominio y mantenimiento de la zona hospedada, necesarios para garantizar disponibilidad y acceso seguro a la aplicación.
- **Herramientas de desarrollo colaborativo (GitHub Actions y repositorios):** incluidas dentro de planes gratuitos o de bajo costo, se consideran parte del gasto fijo vinculado al mantenimiento del ciclo de desarrollo.

Estos componentes conforman una **base económica constante** que permite sostener la infraestructura técnica sin depender de la cantidad de usuarios activos. En términos relativos, su peso sobre el costo total es reducido, pero son fundamentales para la continuidad operativa del producto.

3.8.3.2. Costos variables

Los costos variables representan los componentes más dinámicos del modelo económico.

Surgen de los **servicios cloud que se facturan según consumo**, por lo que su evolución está directamente vinculada al crecimiento de la plataforma. A medida que se incorporan usuarios o se incrementa la frecuencia de uso, estos costos aumentan de forma casi lineal, manteniendo una relación predecible entre inversión y demanda.

Los principales rubros variables son los siguientes:

- **Almacenamiento en Amazon S3:** espacio utilizado por los usuarios para guardar documentos y archivos asociados a las causas legales.
- **Requests y transferencia de datos (S3 egress):** volumen de operaciones de lectura, escritura y descarga de información desde la nube.
- **Aurora Serverless v2 (base de datos):** servicio de base de datos relacional bajo demanda, cuyos costos dependen del número de unidades de capacidad (ACUs) utilizadas, del tamaño de almacenamiento y de los respaldos automáticos.
- **AWS Lambda:** cómputo serverless que ejecuta funciones bajo demanda, con facturación por número de requests y tiempo de ejecución. En los escenarios iniciales, su costo es prácticamente nulo, pero crece gradualmente con el tráfico.

- **Elastic Container Registry (ECR, opcional):** almacenamiento de imágenes de contenedores, aplicable solo en fases avanzadas del proyecto si se integran componentes de machine learning o microservicios adicionales.
- **Servicios complementarios:** incluye herramientas opcionales como envío de correos electrónicos transaccionales, autenticación avanzada, métricas o servicios de inteligencia artificial, cuyo costo depende del uso real de cada API o integración.

En conjunto, esta estructura de costos evidencia que el modelo económico se apoya en una **base fija mínima y controlada**, mientras que los gastos variables responden de manera proporcional al uso y al crecimiento del sistema. Esta característica es central en los proyectos SaaS, donde la escalabilidad técnica y la sostenibilidad financiera avanzan de forma paralela: a medida que la plataforma crece, los costos aumentan, pero lo hacen acompañados por un incremento equivalente -o superior- en los ingresos (Microsoft. S.f).

3.8.4. Escenarios de uso y estimación de costos

Con el objetivo de evaluar la sostenibilidad financiera del proyecto en distintos niveles de adopción, se elaboraron tres escenarios de uso que representan **diferentes etapas de maduración del sistema y de crecimiento de la base de usuarios**. Cada escenario mantiene los mismos componentes de infraestructura, pero varía en función del volumen de usuarios, la cantidad de operaciones realizadas y el consumo de recursos en la nube.

Estos escenarios permiten analizar cómo el proyecto **escala progresivamente** sin comprometer su equilibrio económico, gracias a la estructura serverless y de pago por uso adoptada.

3.8.4.1. Supuestos de consumo por usuario

Para estimar los costos se definieron las siguientes **unidades de consumo promedio** por usuario activo, en base a las pruebas del MVP y las características esperadas de uso:

- **Almacenamiento:** 5 GB por usuario (documentos y archivos asociados).
- **Requests:** 1.000 operaciones mensuales promedio (lecturas, escrituras, descargas).
- **Transferencia de salida (egress):** entre 1 y 2 GB por usuario/mes.

- **Aurora Serverless v2:** consumo base de 2–3 ACUs para 100 usuarios, escalando de forma lineal con la carga.
- **Lambda: volumen proporcional** al número de requests, aunque sin impacto económico relevante en los primeros niveles de escala.

TABLA IV: Unidades de consumo promedio por usuarios activos.

Escenario	Usuarios	Almacenamiento total (GB)	Requests mensuales	Transferencia mensual (GB)	Aurora (ACUs)
MVP	100	500	100.000	100–200	2–3
Crecimiento	1.000	5.000	1.000.000	1.000–2.000	10–12
Expansión	10.000	50.000	10.000.000	10.000–20.000	20–30

3.8.4.2. Estimación de costos

A partir de los precios obtenidos en la calculadora de AWS y las proyecciones de uso, se estimaron los costos totales mensuales y anuales para cada escenario.

Cabe destacar que se trata de **valores referenciales**, utilizados con fines analíticos para evaluar la relación entre escala, costos y rentabilidad. Los precios unitarios se mantuvieron constantes, y los costos variables fueron escalados proporcionalmente al número de usuarios.

TABLA V: Costos mensuales aproximados.

Servicio / Escenario	100 usuarios	1.000 usuarios	10.000 usuarios (extremo)
Amazon S3	~USD 30/mes	~USD 300/mes	~USD 3.000/mes
Aurora Serverless v2	~USD 270/mes	~USD 2.700/mes	~USD 27.000/mes
ECR	~USD 5/mes	~USD 50/mes	~USD 500/mes
Lambda	~USD 0	~USD 1	~USD 20

Vercel + dominio + DNS	~USD 21/mes	~USD 21/mes	~USD 21/mes
Costo mensual total ~	~USD 307	~USD 3.070	~USD 30.700
Costo anual total ~	~USD 3.700	~USD 37.000	~USD 370.000

3.8.4.3. Análisis comparativo entre costos e ingresos

La tabla de costos presentada anteriormente muestra la evolución estimada del gasto operativo a medida que crece la base de usuarios. Sin embargo, para evaluar la verdadera sostenibilidad del modelo, es necesario **contrastar dichos costos con los ingresos esperados** en cada escenario.

A continuación, se presenta una nueva tabla comparativa que mantiene los mismos tres niveles de escala -**MVP, Crecimiento y Expansión**-, incorporando el valor de las suscripciones proyectadas a diferentes precios por usuario (5, 7 y 9 USD mensuales).

De este modo, se observa cómo la relación entre costos e ingresos se mantiene favorable incluso en escenarios conservadores, reforzando la viabilidad económica del proyecto.

TABLA VI: Proyección comparativa de resultados económicos por volumen de usuarios.

Concepto	100 usuarios	1.000 usuarios	10.000 usuarios
Costo anual estimado (USD)	\$3.700/año	\$37.000/año	\$370.000/año
Ingreso ~ anual (USD 5/mes)	\$6.000/año	\$60.000/año	\$600.000/año
Ingreso ~ anual (USD 7/mes)	\$8.400/año	\$84.000/año	\$840.000/año
Ingreso ~ anual (USD 9/mes)	\$10.800/año	\$108.000/año	\$1.080.000/año
Beneficio neto estimado (a 7 USD/mes)	+\$4.700/año	+\$47.000/año	+\$470.000/año

A medida que la plataforma escala hacia los escenarios de **crecimiento y expansión**, el **costo promedio por usuario** se mantiene estable o tiende a disminuir, mientras que los ingresos crecen en proporción directa al número de suscriptores.

Esto produce una **amplificación del margen operativo**, consolidando la viabilidad del modelo SaaS y demostrando que la estructura de costos no compromete la rentabilidad a medida que el sistema se masifica.

3.8.5. Análisis de escenarios financieros

El análisis financiero del proyecto se abordó considerando dos escenarios complementarios que reflejan posibles trayectorias de crecimiento de la base de usuarios durante los primeros cinco años de operación. Ambos escenarios suponen un modelo de ingresos basado en suscripción mensual y un esquema de costos dominado por servicios de cómputo en la nube y almacenamiento, cuya estructura es escalable y proporcional al uso.

3.8.5.1. Escenario I: Crecimiento moderado (100 → 500 usuarios)

En el primer caso se analiza una evolución conservadora, donde la base de usuarios crece desde 100 hasta 500 usuarios activos al quinto año. Este escenario representa un comportamiento de adopción paulatino, con un crecimiento lineal y costos controlados.

Los resultados financieros muestran un **Valor Actual Neto (VAN) de USD 11.875**, calculado a una tasa de descuento del **15 %**, y una **Tasa Interna de Retorno (TIR) del 40 % anual**, lo que indica una rentabilidad superior al costo de capital.

El **punto de equilibrio (PE)** se alcanza aproximadamente a los **3,1 años**, momento en el que el flujo acumulado pasa a ser positivo. El **Retorno sobre la Inversión (ROI)** alcanza un **400 %**, reflejando una relación favorable entre la inversión inicial (USD 3.960) y los beneficios proyectados.

Este escenario demuestra que incluso bajo condiciones moderadas, el proyecto es financieramente viable, con márgenes razonables y una capacidad clara de recuperar la inversión en el mediano plazo. Además, evidencia que la estructura de costos variables permite mantener una sostenibilidad económica sin requerir un volumen masivo de usuarios.

TABLA VII: Análisis económico del Flujo Nominal con crecimiento moderado.

Flujo Nominal					
Concepto	Año 1	Año 2	Año 3	Año 4	Año 5
Usuarios	100	70	150	300	500
Suscripcion (USD/mes)	\$ -	\$ 5,00	\$ 7,00	\$ 7,00	\$ 7,00
Costos variables	\$ 3.700,00	\$ 2.590,00	\$ 5.550,00	\$ 11.100,00	\$ 18.500,00
Costos fijos	\$ 260,00	\$ 260,00	\$ 260,00	\$ 260,00	\$ 260,00
Marketing / Overhead	\$ -	\$ -	\$ 5.000,00	\$ 5.000,00	\$ 7.000,00
Ingresos anuales	\$ -	\$ 4.200,00	\$ 12.600,00	\$ 25.200,00	\$ 42.000,00
Flujo neto	-\$ 3.960,00	\$ 1.350,00	\$ 1.790,00	\$ 8.840,00	\$ 16.240,00
Flujo acumulado	-\$ 3.960,00	-\$ 2.610,00	-\$ 820,00	\$ 8.020,00	\$ 24.260,00

TABLA VIII: Análisis del Flujo Financiero Descontado en los primeros 5 años con crecimiento moderado.

Flujo Financiero Descontado					
Concepto	Año 1	Año 2	Año 3	Año 4	Año 5
Usuarios	100	70	150	300	500
Suscripcion (USD/mes)	\$ -	\$ 5,00	\$ 7,00	\$ 7,00	\$ 7,00
Costos variables	\$ 3.700,00	\$ 2.590,00	\$ 5.550,00	\$ 11.100,00	\$ 18.500,00
Costos fijos	\$ 260,00	\$ 260,00	\$ 260,00	\$ 260,00	\$ 260,00
Marketing / Overhead	\$ -	\$ -	\$ 5.000,00	\$ 5.000,00	\$ 7.000,00
Ingresos anuales	\$ -	\$ 4.200,00	\$ 12.600,00	\$ 25.200,00	\$ 42.000,00
Flujo neto nominal	-\$ 3.960,00	\$ 1.350,00	\$ 1.790,00	\$ 8.840,00	\$ 16.240,00
Factor de descuento	0,870	0,756	0,658	0,572	0,497
Flujo descontado	-\$ 3.443,48	\$ 1.020,79	\$ 1.176,95	\$ 5.054,30	\$ 8.074,15
Flujo acumulado descontado	-\$ 3.443,48	-\$ 2.422,68	-\$ 1.245,73	\$ 3.808,57	\$ 11.882,72

TABLA IX: Métricas clave para la toma de decisiones de inversión.

Métricas	
Métrica	Valor
<i>Tasa</i>	15,00%
<i>VAN</i>	USD 11.875
<i>TIR</i>	40%
<i>ROI</i>	≈ 400%
<i>PE</i>	3,1 años

3.8.5.2. Escenario II: Crecimiento acelerado (100 → 1.000 usuarios)

El segundo escenario plantea una situación más optimista, en la cual el crecimiento de usuarios alcanza los 1.000 al quinto año. Este caso representa una adopción más sólida del servicio, sustentada en una propuesta de valor consolidada y una estrategia comercial activa.

Bajo estos supuestos, el **VAN asciende a USD 39.200**, lo que implica una creación de valor casi **cuatro veces superior** al escenario anterior. La **TIR estimada del 82 %**

anual muestra una rentabilidad excepcional, explicada por la alta escalabilidad del modelo SaaS: el aumento de usuarios genera ingresos adicionales significativos sin requerir una expansión proporcional en los costos fijos.

El **punto de equilibrio** se alcanza aproximadamente a los **2,2 años**, y el **ROI** proyectado ronda el **1.000 %**, es decir, por cada dólar invertido, se generarían diez dólares de ganancia neta.

Este escenario ilustra el potencial de crecimiento del proyecto y su capacidad para generar retornos sustanciales con una inversión inicial relativamente baja. Asimismo, confirma que la naturaleza serverless y la estructura modular de la arquitectura permiten acompañar el crecimiento sin comprometer la estabilidad financiera.

TABLA X: Análisis del Flujo Nominal en los primeros 5 años con crecimiento acelerado.

Flujo Nominal					
Concepto	Año 1	Año 2	Año 3	Año 4	Año 5
Usuarios	100	150	350	600	1000
Suscripcion (USD/mes)	\$ -	\$ 5,00	\$ 7,00	\$ 7,00	\$ 7,00
Costos variables	\$ 3.700,00	\$ 5.550,00	\$ 12.950,00	\$ 22.200,00	\$ 37.000,00
Costos fijos	\$ 260,00	\$ 260,00	\$ 260,00	\$ 260,00	\$ 260,00
Marketing / Overhead	\$ -	\$ -	\$ 5.000,00	\$ 5.000,00	\$ 7.000,00
Ingresos anuales	\$ -	\$ 9.000,00	\$ 29.400,00	\$ 50.400,00	\$ 84.000,00
Flujo neto	-\$ 3.960,00	\$ 3.190,00	\$ 11.190,00	\$ 22.940,00	\$ 39.740,00
Flujo acumulado	-\$ 3.960,00	-\$ 770,00	\$ 10.420,00	\$ 33.360,00	\$ 73.100,00

TABLA XI: Análisis del Flujo Financiero Descontado en los primeros 5 años con crecimiento acelerado.

Flujo Financiero Descontado					
Concepto	Año 1	Año 2	Año 3	Año 4	Año 5
Usuarios	100	150	350	600	1000
Suscripcion (USD/mes)	\$ -	\$ 5,00	\$ 7,00	\$ 7,00	\$ 7,00
Costos variables	\$ 3.700,00	\$ 5.550,00	\$ 12.950,00	\$ 22.200,00	\$ 37.000,00
Costos fijos	\$ 260,00	\$ 260,00	\$ 260,00	\$ 260,00	\$ 260,00
Marketing / Overhead	\$ -	\$ -	\$ 5.000,00	\$ 5.000,00	\$ 7.000,00
Ingresos anuales	\$ -	\$ 9.000,00	\$ 29.400,00	\$ 50.400,00	\$ 84.000,00
Flujo neto nominal	-\$ 3.960,00	\$ 3.190,00	\$ 11.190,00	\$ 22.940,00	\$ 39.740,00
Factor de descuento	0,870	0,756	0,658	0,572	0,497
Flujo descontado	-\$ 3.443,48	\$ 2.412,10	\$ 7.357,61	\$ 13.116,02	\$ 19.757,80
Flujo acumulado descontado	-\$ 3.443,48	-\$ 1.031,38	\$ 6.326,23	\$ 19.442,25	\$ 39.200,05

TABLA XII: Métricas para el análisis económico del crecimiento acelerado.

Metricas	
Metrica	Valor
<i>Tasa</i>	15,00%
<i>VAN</i>	USD 39.200
<i>TIR</i>	82%
<i>ROI</i>	≈ 1000%
<i>PE</i>	2,2 años

4. Metodología de desarrollo

El desarrollo del proyecto se llevó a cabo siguiendo un enfoque ágil, centrado en la iteración continua y la validación temprana con usuarios reales del ámbito jurídico. Esta metodología se seleccionó para asegurar que las funcionalidades implementadas respondieran de manera efectiva a las necesidades concretas de los abogados juniors y estudiantes de Derecho, principales usuarios objetivo de la aplicación.

4.1. Enfoque general: metodología ágil adaptada

Se adoptó una **metodología ágil tipo Scrum**, adaptada al contexto académico del Proyecto Final de Ingeniería. El trabajo se organizó en **sprints quincenales**, cada uno con entregables parciales y reuniones de revisión con el tutor cada tres sprints. Este enfoque permitió una integración progresiva de los módulos de backend, frontend e inteligencia artificial, garantizando coherencia entre las etapas de diseño, desarrollo y validación.

Cada sprint incluyó:

- **Definición de historias de usuario:** se identificaron funcionalidades desde la perspectiva del abogado junior (por ejemplo, “como usuario quiero ver la línea de tiempo del caso para conocer su estado actual”).
- **Planificación de tareas técnicas:** modelado de datos, endpoints API, interfaces de visualización.
- **Implementación incremental:** el código se desarrolló en módulos independientes para facilitar la integración.
- **Revisión y testing:** se realizaron pruebas unitarias, funcionales y de usabilidad.
- **Retrospectiva:** se documentaron hallazgos y se priorizaron mejoras para el siguiente ciclo.

Esta dinámica permitió un flujo iterativo de “**prototipar** → **probar** → **ajustar**”, maximizando la alineación entre los requerimientos del usuario final y los resultados técnicos.



Figura 17. *Historias de usuario realizadas para el proyecto. (Elaboración propia, 2025)*

4.2. Herramientas de gestión y colaboración

Para la coordinación del proyecto y el versionado del código se utilizaron herramientas modernas de colaboración:

- **GitHub:** control de versiones, issues y pull requests.
- **Notion y Google Drive:** documentación y seguimiento de avances.
- **Trello:** definición y seguimiento de tareas. Una instancia del tablero utilizado puede visualizarse en Anexo K.
- **Figma:** diseño colaborativo de interfaces y wireframes.
- **Postman:** pruebas de endpoints RESTful.

4.3. Metodología de desarrollo técnico

El desarrollo siguió una secuencia estructurada en cuatro etapas:

1. **Análisis y modelado de datos:** Se definieron las entidades principales (Causa, Evento, Parte, Documento, Profesional) y sus relaciones,

utilizando diagramas ER y modelos ORM de Django para estructurar la base de datos PostgreSQL.

2. **Diseño UX/UI centrado en el usuario:** Se aplicaron principios de **Design Thinking** y **Legal Design**. A partir de entrevistas con abogados se construyeron wireframes de baja fidelidad en Figma, que luego evolucionaron a prototipos interactivos evaluados con usuarios. Los criterios de evaluación incluyeron: comprensión de la línea de tiempo, claridad de la información y facilidad de navegación.
3. **Implementación backend y frontend:**
 - a. **Backend:** API REST desarrollada en Django REST Framework, estructurada en módulos (Usuarios, Causas, Eventos, Documentos).
 - b. **Frontend:** interfaz desarrollada con React.js, diseño *mobile-first*, integrando componentes visuales (timeline y grafo interactivo).
 - c. **Base de datos:** PostgreSQL alojado en AWS, con backups automáticos.
 - d. **Almacenamiento:** Amazon S3 para documentos.
4. **Integración del módulo de IA:** Se implementó un **módulo de inteligencia artificial** basado en modelos de lenguaje (GPT y Hugging Face Transformers), responsable de:
 - a. Generar resúmenes automáticos de causas.
 - b. Corregir errores gramaticales y eufónicos.
 - c. Sugerir jurisprudencia.

5. Pruebas Realizadas

El proceso de validación del sistema se llevó a cabo mediante dos enfoques complementarios: por un lado, pruebas unitarias, orientadas a verificar el correcto funcionamiento de los componentes técnicos del backend y frontend; y por el otro, pruebas con usuarios finales, destinadas a evaluar la usabilidad, comprensión y adecuación de la herramienta a las necesidades reales de profesionales y estudiantes del ámbito jurídico.

Este enfoque dual permitió asegurar tanto la solidez técnica de la plataforma como su pertinencia funcional para el dominio legal.

5.1. Pruebas Unitarias

Las pruebas unitarias constituyen una técnica fundamental dentro del proceso de aseguramiento de la calidad del software y se orientan a verificar, de manera aislada, el comportamiento correcto de las unidades más pequeñas del código (funciones, métodos o componentes individuales) antes de su integración en módulos superiores. Su objetivo principal es detectar errores de forma temprana, garantizar la estabilidad de los componentes y asegurar que cada unidad cumpla con su especificación funcional bajo diferentes condiciones de entrada. Esta metodología permite incrementar la mantenibilidad y la confiabilidad del sistema, ya que los fallos pueden identificarse y corregirse con menor costo durante las etapas tempranas del desarrollo. Según Pressman (2005), las pruebas unitarias representan el primer nivel del proceso de verificación y validación y proporcionan una base sólida para la integración posterior de componentes.

5.1.1. Enfoque y técnica utilizada

Para la especificación de los casos de prueba se adoptó la técnica **Given–When–Then**, proveniente del enfoque de *Behaviour-Driven Development* (BDD). Esta técnica propone estructurar cada escenario siguiendo tres componentes:

- Given (Dado): el contexto inicial o precondiciones del sistema.
- When (Cuando): la acción que realiza el usuario o el evento que dispara el comportamiento.
- Then (Entonces): el resultado esperado verificable.

Este formato facilita la traducción directa de requisitos funcionales a casos de prueba, reduce la ambigüedad y permite que tanto desarrolladores como expertos del dominio (en este caso, profesionales del derecho) compartan una comprensión común de lo que debe ocurrir en cada funcionalidad (ERTAS, 2024).

5.1.2. Aplicación de las pruebas unitarias

Las pruebas unitarias se aplicaron principalmente sobre:

- Módulo de gestión de causas (creación, edición, listado).
- Módulo de documentos (carga en S3, manejo de metadatos).
- Módulo de IA (resumen, corrección y análisis).

- Asistente de Jurisprudencia (RAG).
- Integración opcional con Tavily para búsqueda extendida.

En la implementación concreta, cada uno de estos escenarios fue traducido a pruebas que:

- verifican el código de estado HTTP retornado por los endpoints,
- validan la estructura de los datos de respuesta (campos obligatorios, tipos y formatos),
- comprueban la correcta interacción entre capas (por ejemplo, entre el servicio de IA y la persistencia),
- y simulan errores controlados (fallos en servicios externos, datos incompletos, permisos insuficientes) para asegurar un manejo robusto de excepciones.

A continuación, se presentan algunos ejemplos representativos de cómo se modelaron los escenarios de prueba utilizando la técnica Given–When–Then.

5.1.2.1. Prueba 1: Creación exitosa de una causa judicial.

GIVEN: Un usuario autenticado que completa los campos obligatorios de una Causa.

WHEN: Envía una solicitud POST al endpoint `/api/causas/`

THEN: Se crea un registro de causa en la base de datos.

5.1.2.2. Prueba 2: Resumen automático de un documento almacenado en S3.

GIVEN: Un documento existente, asociado a una causa almacenado en S3 y correctamente leído por TextExtract.

WHEN: El usuario solicita el resumen mediante el endpoint `/api/causas/ia-summary/`

THEN: El backend envía el texto al modelo GPT, se recibe un resumen en formato JSON con los campos esperados y el resumen se almacena vinculado al documento original.

5.1.2.3. Prueba 3: Registro correcto de una interacción usuario–asistente.

GIVEN: Una conversación activa identificada por un `conversation_id` válido

WHEN: el usuario envía una pregunta al asistente.

THEN: Se crea un registro Message con rol "user", una vez generada la respuesta, se crea un registro Message con rol "assistant" y se actualizan los campos `updated_at` y `last_message_at` de la conversación

5.2. Pruebas de Usabilidad

Se llevaron a cabo seis sesiones individuales con estudiantes avanzados de abogacía familiarizados con el uso de la MEV y Lex-Doctor. Cada participante realizó las siguientes tareas:

- Crear una causa y editar sus datos.
- Cargar un documento PDF y analizar el resumen automático.
- Interpretar la línea de tiempo interactiva.
- Explorar el grafo de actores y eventos.
- Consultar jurisprudencia mediante el asistente IA.
- Probar la búsqueda extendida con Tavily cuando les resultara relevante.

Los estudiantes reportaron:

- Alta claridad visual en la línea de tiempo y el grafo relacional.
- Ahorro de tiempo en la comprensión inicial de un caso gracias al resumen automático.
- Facilidad de uso, incluso sin experiencia previa con herramientas avanzadas.
- Relevancia de los resultados del asistente RAG, con apreciación especial por las citas verificables.
- Percepción de que la búsqueda extendida (Tavily) era útil para obtener contexto general o encontrar análisis doctrinarios externos.

Además de las seis sesiones individuales, una de las participantes, Lucía Suárez, estudiante de Derecho, brindó una entrevista en profundidad, cuyo desarrollo completo se presenta en el Anexo L. Esta instancia permitió obtener evidencia cualitativa más completa sobre la experiencia real de uso. La participante señaló de manera contundente que logró comprender la estructura completa del caso en aproximadamente tres minutos, tanto cuando

trabajó con una causa ya existente como cuando cargó un documento propio y utilizó la clasificación automática mediante machine learning. Destacó que el grafo de relaciones, la línea de tiempo y el dashboard inicial fueron determinantes para acelerar su comprensión, eliminando la necesidad de leer extensivamente el expediente original. Su testimonio reafirma los resultados obtenidos y confirma de forma explícita que la interfaz permite alcanzar el objetivo planteado: que al menos el 80% de los usuarios comprendan correctamente la estructura de un caso en menos de 7 minutos. Esta entrevista no solo valida el diseño, sino que además aporta ejemplos concretos y observaciones que reforzaron las decisiones de mejora introducidas en la plataforma.

A partir de estos aportes, junto con el resto del feedback recopilado durante las sesiones de prueba, se implementaron ajustes específicos orientados a optimizar la experiencia de uso, entre los cuales se destacan:

- Ajustes en el dashboard.
- Mejoras en las secciones de la interfaz.
- Refinamiento del color en nodos del grafo para mejorar el contraste.

5.3. Conclusión de las Pruebas

El conjunto de pruebas técnicas y de validación con usuarios confirmó que la plataforma cumple con los objetivos funcionales, técnicos y de usabilidad planteados en el proyecto.

La aplicación demostró: estabilidad técnica, consistencia en la integración de IA y módulos externos, claridad visual y facilidad de uso, pertinencia jurídica de las funcionalidades, aceptación positiva por parte de usuarios con distintos niveles de experiencia.

Los resultados de estas pruebas aportan evidencia sólida para sostener la viabilidad del sistema como herramienta de apoyo para abogados juniors y estudiantes en la gestión visual de causas judiciales.

6. Discusión

Una vez finalizada la implementación del prototipo funcional de LexGO, se identificaron diversas oportunidades de evolución tanto a nivel tecnológico como de producto. El desarrollo del sistema permitió constatar fortalezas, pero también limitaciones y aspectos susceptibles de ser optimizados en futuras iteraciones. La discusión de estos puntos resulta

fundamental para proyectar el crecimiento de la herramienta y anticipar los desafíos asociados a su escalabilidad y adopción en entornos jurídicos reales.

En primer lugar, uno de los desafíos más significativos se relaciona con la integración de técnicas de inteligencia artificial aplicadas al análisis documental y a la recuperación aumentada. La selección, prueba y ajuste de modelos (tanto propietarios como open-source) requirió tiempos de exploración mayores a los planificados, debido a la variabilidad de comportamientos entre modelos, a las diferencias en su capacidad de comprensión jurídica y a la necesidad de asegurar coherencia semántica en español. Asimismo, aunque el uso de servicios cloud como OpenAI API, Textract o Tavily permitió acelerar el desarrollo, la dependencia de servicios externos introduce factores de latencia, costos operativos y límites de cuota que deben ser considerados en un despliegue productivo. A futuro, la posibilidad de emplear modelos alojados en infraestructura privada o en servicios administrados como AWS Bedrock podría contribuir a mejorar la estabilidad, independencia y rendimiento del sistema.

Un segundo punto de discusión se vincula al diseño de la arquitectura distribuida y su potencial crecimiento. Si bien el sistema actual funciona en un esquema modular y desacoplado, una evolución hacia un entorno multiusuario y multiestudio demandaría mayor robustez en la gestión de autenticación, permisos, auditoría y almacenamiento seguro de expedientes. De igual manera, funcionalidades intensivas en recursos (como la generación de resúmenes o la inferencia de jurisprudencia) requerirían estrategias de caching, colas de procesamiento asíncrono o escalado automático para sostener mayores volúmenes de carga. La implementación de pipelines más complejos, tales como análisis comparado de fallos, extracción de entidades jurídicas o predicción de próximos pasos procesales, implicaría incorporar técnicas adicionales de NLP y modelos especializados.

En cuanto al módulo Asistente de Jurisprudencia, las pruebas realizadas evidencian que el enfoque RAG ofrece buenos resultados en una base documental acotada; sin embargo, a medida que la cantidad de documentos aumente, será necesario optimizar los índices de vectores, mejorar los filtros contextuales y evaluar la incorporación de embeddings entrenados específicamente en el dominio jurídico argentino. Además, la integración con Tavily, si bien agrega valor en búsquedas extendidas, introduce el desafío de evaluar la confiabilidad y pertinencia de fuentes externas, por lo que podría contemplarse en el futuro un

sistema de validación automática o clasificación de dominios para asegurar que la información complementaria cumpla criterios estrictos de calidad y autenticidad.

Otro aspecto relevante identificado durante la discusión del proyecto es la posibilidad de evolucionar hacia una plataforma multiusuario, como se ha mencionado previamente, especialmente considerando escenarios donde distintos abogados, estudios jurídicos o equipos de trabajo puedan colaborar en el análisis de causas. Si bien la versión actual de LexGO fue diseñada como un entorno individual y la interfaz web es completamente responsive, los participantes de las pruebas señalaron que no consideran necesario el desarrollo de una aplicación móvil nativa. En cambio, destacaron que una funcionalidad de mayor valor agregado sería la habilitación de espacios colaborativos donde múltiples usuarios puedan crear, editar y visualizar expedientes compartidos, con distintos niveles de permisos. Esta expansión implicaría replantear la arquitectura actual para incorporar módulos de gestión de roles, auditoría de acciones, control de acceso basado en políticas y mecanismos robustos de sincronización en tiempo real entre usuarios. La transición hacia un sistema multiusuario no solo ampliaría el alcance del producto, sino que también permitiría su adopción en entornos profesionales donde el trabajo colaborativo sobre expedientes resulta una necesidad operativa cotidiana.

Finalmente, se identifican oportunidades para ampliar el alcance funcional de LexGO incorporando capacidades adicionales tales como:

- predicción automática de plazos procesales,
- extracción estructurada de entidades jurídicas (personas, organismos, montos, fechas),
- análisis comparado de casos similares,
- integración futura con sistemas judiciales electrónicos (MEV, Lex-100),
- incorporación de recomendaciones estratégicas basadas en patrones detectados en fallos,
- entrenamiento personalizado a partir de expedientes previos del usuario.

Estas líneas futuras no solo representan una ampliación tecnológica, sino también un desafío ético y jurídico en cuanto a transparencia, protección de datos y explicabilidad del modelo. La consolidación de estas mejoras permitiría posicionar a LexGO

como una herramienta robusta, escalable y alineada con las necesidades reales del ecosistema jurídico contemporáneo.

En síntesis, el proyecto alcanzó sus objetivos funcionales y demostró su viabilidad como herramienta de apoyo para la comprensión visual de causas judiciales. No obstante, el análisis crítico del desarrollo permite identificar caminos de expansión que, abordados adecuadamente, podrían convertir a LexGO en un sistema más completo, colaborativo y adaptable a entornos profesionales complejos.

7. Conclusión

El desarrollo de LexGO permitió demostrar que la integración de visualización interactiva, ingeniería de software moderna e inteligencia artificial puede transformar significativamente la manera en que abogados y estudiantes comprenden, analizan y gestionan causas judiciales. A lo largo del proyecto se logró diseñar e implementar una plataforma capaz de abordar uno de los principales desafíos del ámbito legal: la fragmentación de la información procesal y la dificultad de obtener una visión clara y estructurada del estado de un expediente.

El sistema articula distintos componentes: línea de tiempo, grafo relacional, asistente jurídico basado en RAG, extracción automática de información y generación de resúmenes; para ofrecer una experiencia integral que reduce la carga cognitiva del usuario y mejora el acceso a datos relevantes. La arquitectura basada en principios de Clean Architecture, junto con la modularización del motor de IA, permitió garantizar escalabilidad, mantenibilidad y separación clara de responsabilidades. Asimismo, la incorporación de servicios externos como AWS Textract, OpenAI y Tavily evidenció la capacidad del sistema para integrarse con tecnologías avanzadas manteniendo un flujo controlado, verificable y seguro.

Las pruebas unitarias, junto con las pruebas con usuarios reales, validaron tanto la solidez técnica del sistema como su pertinencia funcional. El acompañamiento continuo de un abogado experto en IA resultó esencial para garantizar la fidelidad jurídica del modelo, mientras que las sesiones con estudiantes permitieron identificar oportunidades de mejora en la interfaz y confirmar el valor pedagógico del enfoque visual propuesto. Los usuarios destacaron la claridad de la línea de tiempo, la utilidad del grafo y la precisión del asistente de jurisprudencia, lo que indica que la herramienta satisface necesidades reales dentro del proceso de aprendizaje y práctica jurídica.

Si bien el prototipo cumple los objetivos planteados, el análisis crítico permitió identificar oportunidades para su evolución. La expansión hacia una plataforma multiusuario, la optimización del módulo RAG para grandes volúmenes de jurisprudencia, la mejora de mecanismos de ayuda interactiva y la incorporación futura de modelos especializados en derecho argentino conforman líneas claras de desarrollo. Estas posibilidades no solo fortalecen el producto, sino que abren el camino hacia una herramienta colaborativa que pueda integrarse en estudios jurídicos, clínicas legales universitarias o laboratorios de innovación judicial.

En términos más amplios, el proyecto demuestra que la aplicación responsable y contextualizada de IA puede generar beneficios concretos dentro del ecosistema jurídico, siempre que se mantenga un compromiso con la trazabilidad, la transparencia y la supervisión humana. La plataforma no busca reemplazar el criterio profesional, sino potenciarlo mediante herramientas que mejoran la comprensión, agilizan tareas y facilitan el acceso a información procesal y jurisprudencial.

En conclusión, LexGO constituye un aporte significativo al campo de las LegalTech, al proponer una solución innovadora y técnicamente robusta que combina análisis visual, automatización inteligente y recuperación aumentada para asistir a profesionales y estudiantes en la interpretación de expedientes judiciales. El trabajo realizado sienta bases sólidas para futuras ampliaciones y posiciona a la plataforma como un prototipo viable en la evolución digital del derecho en Argentina.

8. Bibliografía

ACADEMIA DE LA MAGISTRATURA DEL PERÚ y PASTOR, Dr.R.L., 2000. Razonamiento jurídico: educación a distancia [en línea]. Lima, Perú: s.n. ISBN 9972-779-04-3. Disponible en: <https://www.corteidh.or.cr/tablas/r32764.pdf>.
<https://www.corteidh.or.cr/tablas/r32764.pdf>

AGUERRE, CAROLINA, AMUNÁTEGUI, CARLOS, ARANGUIS, MATIAS, BUSTOS FRATI, GONZALO, CASTAÑO, DANIEL, MENDOZA ENRIQUEZ, OLIVIA ANDREA, MORENO GONZÁLEZ, JIMENA, GORGONE, BRUNO, LENS, JULIO, MADRID, RAÚL, MAQUEO RAMÍREZ, MARÍA SOLANGE, RENTERÍA MARÍN, CÉSAR, SEGREDO, SANDRA, y VARGAS, FERNANDO, 2021. Preparación del sector judicial para la inteligencia artificial en América Latina. [en línea], [consulta: 26 mayo 2025]. Disponible en: <http://hdl.handle.net/10908/18634>.

AJEVSKI, M., BARKER, K., GILBERT, A., HARDIE, L. y RYAN, F., 2023. ChatGPT and the future of legal education and practice. *The Law Teacher* [en línea], vol. 57, no. 3, pp. 352-364. [consulta: 26 mayo 2025]. ISSN 0306-9400. DOI [10.1080/03069400.2023.2207426](https://doi.org/10.1080/03069400.2023.2207426). Disponible en: <https://doi.org/10.1080/03069400.2023.2207426>.

AMAZON WEB SERVICES, [sin fecha]. Amazon S3 - Almacenamiento de objetos en la nube - AWS. *Amazon Web Services, Inc.* [en línea]. [consulta: 01 octubre 2025]. Disponible en: <https://aws.amazon.com/es/s3/>.

AMAZON WEB SERVICES, [sin fecha]. ¿Qué es la IA GPT? - Explicación de los transformadores generativos preentrenados - AWS. *Amazon Web Services, Inc.* [en línea]. [consulta: 01 octubre 2025]. Disponible en: <https://aws.amazon.com/es/what-is/gpt/>.

AMERSHI, S., BEGEL, A., BIRD, C., DELINE, R., GALL, H., KAMAR, E., NAGAPPAN, N., NUSHI, B. y ZIMMERMANN, T., 2019. Software Engineering for Machine Learning: A Case Study. *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)* [en línea]. S.l.: s.n., pp. 291-300. [consulta: 26 mayo 2025]. DOI [10.1109/ICSE-SEIP.2019.00042](https://doi.org/10.1109/ICSE-SEIP.2019.00042). Disponible en: <https://ieeexplore.ieee.org/abstract/document/8804457>.

BRUNSCHWIG, C.R., 2021. Visual Law and Legal Design: Questions and Tentative Answers. *Jusletter-IT* [en línea], no. 27- Mai-2021, [consulta: 26 mayo 2025]. ISSN

1664-848X. DOI [10.38023/8b70bb88-de0c-4034-a54c-68409bb9549e](https://doi.org/10.38023/8b70bb88-de0c-4034-a54c-68409bb9549e). Disponible en: https://jusletter-it.weblaw.ch/issues/2021/27-Mai-2021/05_rechtstheorie_02_d31aff7334.html.

CENTRO DE INFORMACIÓN JUDICIAL (CIJ), 2019. Mediación. [en línea]. [consulta: 26 mayo 2025]. San Juan, Argentina: Disponible en: <https://www.jussanjuan.gov.ar/jurisprudencia/wp-content/uploads/sites/3/2023/12/ACORDADAS-Y-RESOLUCIONES-ACERCA-DE-ORGANISMOS-V1-1.pdf>.

COLEGIO DE ABOGADOS DE SAN ISIDRO, [sin fecha]. Etapas del Proceso Civil | Colegio de Abogados de San Isidro (CASI). [en línea]. [consulta: 26 mayo 2025]. Disponible en: <https://www.casi.com.ar/content/etapas-del-proceso-civil-1>.

COLEGIO DE ABOGADOS DE SAN ISIDRO, 2007. *Memoria Anual 2007* [en línea]. [consulta: 26 mayo 2025]. 2007. S.l.: s.n. Disponible en: <https://www.casi.com.ar/sites/default/files/MEMORIA%2007.pdf>.

DELGADO MARTÍN, J., 2021. Tecnología para afrontar los efectos de la pandemia sobre la justicia. *Diario La Ley* [en línea], Disponible en: <https://apm3-9.studiweb.com/tecnologia.pdf>.

ERTAS, F., 2024. Enhancing Your Tests: The Given-When-Then Naming Technique for Unit Testing. *turkcell* [en línea]. [consulta: 6 noviembre 2025]. Disponible en: <https://medium.com/turkcell/enhancing-your-tests-the-given-when-then-naming-technique-for-unit-testing-4e821704dd8a>.

ENTREVISTA PROPIA A MICHELLE TROUCHOT, abogada especializada en IA. Buenos Aires, mayo de 2025. Registro interno del equipo de proyecto.

ENTREVISTA PROPIA AL DR. MARCOS VILAPLANA, presidente del Colegio de Abogados de General San Martín. Buenos Aires, mayo de 2025. Registro interno del equipo de proyecto.

ESTEVEZ, E., FILLOTTRANI, P. y LEJARRAGA, S.L., 2020. PROMETEA: Transformando la administración de justicia con herramientas de inteligencia artificial. *IDB Publications* [en línea], [consulta: 26 mayo 2025]. DOI [10.18235/0002378](https://doi.org/10.18235/0002378). Disponible en: <https://publications.iadb.org/es/prometea-transformando-la-administracion-de-justicia-con-herramientas-de-inteligencia-artificial>.

FUNDAR, [sin fecha]. Transformación digital en el sistema de justicia. *Fundar* [en línea]. [consulta: 26 mayo 2025]. Disponible en: <https://fund.ar/publicacion/transformacion-digital-en-el-sistema-de-justicia/>.

GÓRSKI, Ł., RAMAKRISHNA, S. y NOWOSIELSKI, J.M., 2021. Towards Grad-CAM Based Explainability in a Legal Text Processing Pipeline. Extended Version. En: V. RODRÍGUEZ-DONCEL, M. PALMIRANI, M. ARASZKIEWICZ, P. CASANOVAS, U. PAGALLO y G. SARTOR (eds.), *AI Approaches to the Complexity of Legal Systems XI-XII*. Cham: Springer International Publishing, pp. 154-168. ISBN 978-3-030-89811-3. DOI [10.1007/978-3-030-89811-3_11](https://doi.org/10.1007/978-3-030-89811-3_11).

HASHICORP, [sin fecha]. What is Terraform | Terraform | HashiCorp Developer. [en línea]. [consulta: 6 noviembre 2025]. Disponible en: <https://developer.hashicorp.com/terraform/intro>.

HEGEL, A., SHAH, M., PEASLEE, G., ROOF, B. y ELWANY, E., 2021. *The Law of Large Documents: Understanding the Structure of Legal Contracts Using Visual Cues* [en línea]. 16 julio 2021. S.l.: arXiv. [consulta: 01 agosto 2025]. arXiv:2107.08128. Disponible en: <http://arxiv.org/abs/2107.08128>.

IBM, 2024. AI in Software Development | IBM. [en línea]. [consulta: 10 octubre 2025]. Disponible en: <https://www.ibm.com/think/topics/ai-in-software-development>.

IBM, 2025. ¿Qué es el modelado de datos? | IBM. [en línea]. [consulta: 10 octubre 2025]. Disponible en: <https://www.ibm.com/es-es/think/topics/data-modeling>.

ISO, [sin fecha]. SIST EN ISO 9241-210:2020 - Ergonomics of human-system interaction - Part 210: Human-centred design for interactive systems (ISO 9241-210:2019). *iTeh Standards* [en línea]. [consulta: 01 octubre 2025]. Disponible en: <https://standards.iteh.ai/catalog/standards/sist/69d00dac-f209-4b17-acc7-715a0309fe4c/sist-en-iso-9241-210-2020>.

KAME-CODE.COM, 2025. Lex-Doctor 11 - Nivel 1 - Extensión Universitaria | Facultad de Derecho - Universidad de Buenos Aires. [en línea]. [consulta: 26 mayo 2025]. Disponible en: <https://www.derecho.uba.ar/extension/cursos/programa/lex-doctor-10/lex-doctor-11-nivel-1/2373>.

KATZ, D.M., HARTUNG, D., GERLACH, L., JANA, A. y BOMMARITO, M.J., 2023. *Natural Language Processing in the Legal Domain* [en línea]. SSRN Scholarly

Paper. 24 enero 2023. Rochester, NY: Social Science Research Network. [consulta: 26 mayo 2025]. 4336224. Disponible en: <https://papers.ssrn.com/abstract=4336224>.

KIM, W.C. y MAUBORGNE, R., 2005. Blue Ocean Strategy. *Harvard Business Review* [en línea], [consulta: 29 septiembre 2025]. ISSN 0017-8012. Disponible en: <https://hbr.org/2004/10/blue-ocean-strategy>.

LEX-DOCTOR, [sin fecha]. Lex-Doctor 11 - LEX-DOCTOR, Gestion Juridica | Software Juridico | Programa Juridico para Abogados | Software de Gestion Juridica para abogados. [en línea]. [consulta: 26 mayo 2025]. Disponible en: <https://www.lex-doctor.com/>.

LU, R., 2023. The design thinking behind Notion AI. *Notion* [en línea]. [consulta: 15 agosto 2025]. Disponible en: <https://www.notion.com/blog/the-design-thinking-behind-notion-ai>.

MAC-GREGOR, E.F., 2012. Aportaciones de Héctor Fix-Zamudio al derecho procesal constitucional. [en línea], Disponible en: <https://archivos.juridicas.unam.mx/www/bjv/libros/3/1088/8.pdf>.

MARTIN, R.C., GRENNING, J., BROWN, S. y HENNEY, K., 2018. *Clean Architecture: a craftsman's guide to software structure and design*. Boston Columbus Indianapolis New York San Francisco Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montreal Toronto Delhi Mexico City São Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo: Prentice Hall. Robert C. Martin series, ISBN 978-0-13-449416-6.

MCLACHLAN, S., KYRIMI, E., DUBE, K., FENTON, N. y WEBLEY, L.C., 2023. Lawmaps: enabling legal AI development through visualisation of the implicit structure of legislation and lawyerly process. *Artificial Intelligence and Law* [en línea], vol. 31, no. 1, pp. 169-194. [consulta: 30 mayo 2025]. ISSN 1572-8382. DOI [10.1007/s10506-021-09298-0](https://doi.org/10.1007/s10506-021-09298-0). Disponible en: <https://doi.org/10.1007/s10506-021-09298-0>.

MCLACHLAN, S. y WEBLEY, L.C., 2021. Visualisation of law and legal Process: An opportunity missed. *Information Visualization* [en línea], vol. 20, no. 2-3, pp. 192-204. [consulta: 30 mayo 2025]. ISSN 1473-8716. DOI [10.1177/14738716211012608](https://doi.org/10.1177/14738716211012608). Disponible en: <https://doi.org/10.1177/14738716211012608>.

MDN WEB DOCS, 2023. MVC - Glosario de MDN Web Docs | MDN. *MDN Web Docs* [en línea]. [consulta: 20 septiembre 2025]. Disponible en: <https://developer.mozilla.org/es/docs/Glossary/MVC>.

MICROSOFT, [sin fecha]. What is Software as a Service (SaaS)? | Microsoft Azure. [en línea]. [consulta: 01 octubre 2025]. Disponible en: <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-saas/>.

MIK, E., 2020. *The Limits of Visual Law* [en línea]. SSRN Scholarly Paper. 24 febrero 2020. Rochester, NY: Social Science Research Network. [consulta: 26 mayo 2025]. 3543730. Disponible en: <https://papers.ssrn.com/abstract=3543730>.

MURRAY, M., 2021. Diagrammatics and the Proactive Visualization of Legal Information. *University of Arkansas at Little Rock Law Review* [en línea], vol. 43, no. 3, pp. 1. Disponible en: <https://lawrepository.ualr.edu/lawreview/vol43/iss3/1>.

NGROK INC. *How Ngrok Works*. Ngrok Documentation. Disponible en: [How does ngrok work? - ngrok documentation](https://ngrok.com/docs)

ONAMI, E., MIYANISHI, T., MAEDA, K. y KURITA, S., 2025. *LegalViz: Legal Text Visualization by Text To Diagram Generation* [en línea]. 13 febrero 2025. S.l.: arXiv. [consulta: 26 mayo 2025]. arXiv:2502.06147. Disponible en: <http://arxiv.org/abs/2502.06147>.

PORTAL OFICIAL DEL ESTADO ARGENTINO, 2024. MEV | Mesa de Entradas Virtual. *Argentina.gob.ar* [en línea]. [consulta: 26 mayo 2025]. Disponible en: <https://www.argentina.gob.ar/mev-mesa-de-entradas-virtual>.

PRESSMAN, R.S., 2005. *Software Engineering: A Practitioner's Approach*. S.l.: Palgrave Macmillan. ISBN 978-0-07-301933-8.

QUANTIC SCHOOL OF BUSINESS AND TECHNOLOGY, 2025. AI-Focused Software Engineering: Real-World Impact - The Qumatic Blog %. *Quantic School of Business and Technology* [en línea]. [consulta: 01 octubre 2025]. Disponible en: https://quantic.edu/blog/2025/02/10/ai-focused-software-engineering-mastering-real-world-applications/?utm_source=chatgpt.com.

REIMERS, N. y GUREVYCH, I., 2019. *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks* [en línea]. 27 agosto 2019. S.l.: arXiv. [consulta: 10 octubre 2025]. arXiv:1908.10084. Disponible en: <http://arxiv.org/abs/1908.10084>.

RESCK, L.E., PONCIANO, J.R., NONATO, L.G. y POCO, J., 2023. LegalVis: Exploring and Inferring Precedent Citations in Legal Documents. En: arXiv:2203.02001 [cs], *IEEE Transactions on Visualization and Computer Graphics* [en línea], vol. 29, no. 6, pp. 3105-

3120. [consulta: 26 mayo 2025]. ISSN 1077-2626, 1941-0506, 2160-9306. DOI [10.1109/TVCG.2022.3152450](https://doi.org/10.1109/TVCG.2022.3152450). Disponible en: <http://arxiv.org/abs/2203.02001>.

RODGERS, I., ARMOUR, J. y SAKO, M., 2023. How Technology Is (or Is Not) Transforming Law Firms. *Annual Review of Law and Social Science* [en línea], vol. 19, no. 1, pp. 299-317. [consulta: 26 mayo 2025]. ISSN 1550-3585, 1550-3631. DOI [10.1146/annurev-lawsocsci-111522-074716](https://doi.org/10.1146/annurev-lawsocsci-111522-074716). Disponible en:

<https://www.annualreviews.org/doi/10.1146/annurev-lawsocsci-111522-074716>.

SAKO, M., PARNHAM, R., ARMOUR, J., RODGERS, I. y QIAN, M., [sin fecha]. Technology and Innovation in Legal Services: Final Report for the Solicitors Regulation Authority. [en línea], Disponible en: <https://contact.sra.org.uk/globalassets/documents/sra/research/full-report-technology-and-innovation-in-legal-services.pdf>.

SANCHO- ELIZONDO, Y.G., 2014. Propuesta metodológico-didáctica en busca del razonamiento jurídico: La Aplicación de técnicas didácticas comunes. *InterSedes* [en línea], vol. 15, no. 31, pp. 20-27. [consulta: 26 mayo 2025]. ISSN 2215-2458. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=9490740>.

SINSHEIMER, A.N. y HERRING, D.J., 2016. Lawyers at Work: A Study of the Reading, Writing, and Communication Practices of Legal Professionals. [en línea], Disponible en: https://scholarship.law.pitt.edu/fac_articles/139.

TAVILY. Tavily Docs [en línea], [sin fecha]. [consulta: 6 noviembre 2025]. Disponible en: <https://docs.tavily.com/documentation/about>.

TRIALLINE, [sin fecha]. TrialLine. *Clio* [en línea]. [consulta: 06 junio 2025]. Disponible en: <https://www.clio.com/app-directory/trialline/>.

UNIVERSIDAD CATÓLICA ARGENTINA, 2018. UCA - Universidad Católica Argentina. [en línea]. [consulta: 26 mayo 2025]. Disponible en: <https://uca.edu.ar/es/eventos/informatica-juridica.-herramientas-para-el-manejo-del-sistema-lex-doctor-1>.

9. Anexos

Anexo A. Estructura de una causa judicial y su interpretación en la App

TABLA XIII: Estructura de una causa judicial y su interpretación en la App.

Concepto	Qué es	Interpretación en la App
Causa judicial	Es el “caso” o conflicto que se tramita en la justicia. Puede ser laboral, civil, penal, etc.	Es el objeto central que el usuario va a cargar como “Proyecto” en su app.
Expediente	Es el conjunto de documentos y actuaciones que se van acumulando en una causa.	Son los “archivos adjuntos”, textos, escritos, y actuaciones legales.
Evento procesal	Cada acción que ocurre en una causa: presentación de demanda, contestación, audiencia, sentencia.	Son los nodos que van en la línea de tiempo o grafo.
Estado procesal	Es la etapa en la que está la causa: inicio, prueba, sentencia, etc.	Permite dar resúmenes como “Causa en etapa de prueba”.
Fallo	Resolución final de un juez sobre el conflicto.	Puede ser un evento clave o un cierre de la línea de tiempo.
Jurisprudencia	Fallos anteriores usados como referencia.	Se puede sugerir en la app como ayuda para pensar el caso.
Plazo o vencimiento	Fecha límite para hacer algo (ej: responder una demanda).	Se puede usar para alertas o como estado “vencido / cumplido”.

Anexo B: Entrevista al Dr. Marcos Vilaplana

Marcos, ¿alguna vez te entrevistaron para una tesis?

No, para tesis no. Entrevistas periodísticas y profesionales sí, pero en el marco de una tesis, no. Es la primera vez igual que ustedes.

Te contamos brevemente la idea de nuestra tesis: una plataforma visual para abogados jóvenes o estudiantes avanzados, con línea de tiempo, documentos agrupados por etapa procesal, conexión entre actores y documentos, y funciones de inteligencia artificial. ¿Qué opinás?

La idea está muy bien. Pero quiero hacer una observación importante: ustedes la piensan para abogados jóvenes, pero en realidad quienes más la necesitan son los abogados de mi edad o mayores, que no manejan bien la tecnología. Para ustedes, navegar plataformas es intuitivo. Para otras generaciones, no. Si desarrollan una interfaz muy amigable, pueden abarcar un público enorme.

Hoy, no incorporar tecnología es casi una sentencia de muerte profesional. El diferencial entre quien usa estas herramientas y quien no las usa es enorme.

Mencionamos que la tecnología apareció hace relativamente poco en el Derecho. ¿Cuál es tu percepción del cambio en estos años?

La transformación es muy reciente. Hace apenas unos años se hablaba por primera vez de expediente digital. Para que entiendas lo rápido que fue el cambio, te cuento una anécdota:

En 2015 o 2016, una delegación del Colegio viajó a un país a ver cómo funcionaba la justicia. Volvieron diciendo: “*No hay un solo papel*”. Ese país no era Estados Unidos ni Inglaterra: era Brasil. Mientras tanto, acá trabajábamos literalmente entre montañas de carpetas. Nosotros veíamos en los tribunales “casitas hechas de carpetas”. Cuando volvieron de ese viaje, entendimos que había que cambiar.

Después llegó la presentación electrónica: antes había que imprimir todo, llevarlo al juzgado, pedir sello, volver al estudio. Hoy lo presentás a distancia, incluso desde otro lugar del país.

Pero lo más fuerte fue la pandemia. En marzo de 2020, 15.000 abogados ya usaban token, pero ningún juez tenía puesto remoto habilitado. La pandemia obligó al sistema entero a digitalizarse de golpe. Hoy funciona muy bien. Y con IA va a pasar lo mismo: quien no se adapte, queda afuera.

¿Qué herramientas o experiencias tenés con IA aplicada al Derecho?

Empecé un posgrado en IA y Derecho, y en las primeras clases nos enseñaron herramientas como Notebook LM. Podés cargar: videos de audiencias, documentos del

expediente, audios, PDFs. Y la IA trabaja solo con esa información, sin buscar en Google ni inventar cosas. Puede: transcribir audios, resumir testimonios, enfocarse en ciertos temas, generar borradores de escritos, armar mapas conceptuales.

Es impresionante. Y todavía es la prehistoria de lo que viene. Me transformé en un predicador de la tecnología, aunque insisto, todavía yo soy prehistórico.

Para entender las necesidades reales, ¿qué tipo de documentos y datos manejan los abogados diariamente?

Siempre se parte de lo general a lo particular:

1. Constitución Nacional
2. Código de fondo (Laboral, Civil, etc.)
3. Normas procesales
4. Jurisprudencia (fallos previos)

Muchas situaciones se repiten: accidentes de tránsito, contratos, etc. Entonces los estudios acumulan modelos. Pero eso genera pereza intelectual: se usa el modelo, se cambian nombres, se adapta algo y listo. Antes, con la máquina de escribir, había que pensar cada palabra.

Te cuento algo personal: Hace muchos años buscábamos un escrito viejo nuestro. Lo encontramos y mi socio me dijo: “*Me sorprende lo bien que escribíamos*”. ¿Por qué? Porque antes pensábamos cada escrito; hoy a veces solo se copia y pega. Eso ya pasaba antes de la IA. La IA simplemente multiplica ese riesgo.

¿Cómo manejan hoy los abogados la información? ¿Qué está digitalizado?

En Provincia de Buenos Aires, casi todo es formato digital. Todavía hay papeles viejos en el juzgado, pero ya no se usan para trabajar.

Un acta de audiencia, por ejemplo, se registra digitalmente. Eso permite que la IA analice:

- qué dijo un testigo,
- qué hechos se mencionaron,
- qué contradicciones hay.

La mayoría de los abogados ni sabe que esto existe.

¿Una línea de tiempo con plazos y documentos realmente ayuda?

Muchísimo. En el Derecho procesal, los plazos son sagrados. Si no contestaste dentro del plazo, aunque tengas razón, perdiste.

Ejemplo típico: Un auto cruza en rojo y choca. Es obvio quién tuvo la culpa. Pero si el que cruzó en verde no contesta la demanda a tiempo, pierde igual. El sistema no espera.

La línea de tiempo ayuda a evitar errores graves. Y si además “cruza” todas las causas del estudio, es aún mejor. Porque a veces se juntan: audiencias, plazos de prueba, traslados, vencimientos, escritos obligatorios.

Vos querés irte de vacaciones en febrero, y el sistema te puede decir: “*Tenés un quilombo padre entre el 15 y el 18: se te junta todo.*”

Eso es oro.

¿Un grafo que conecte actores, hechos y documentos sería útil?

Sí. Notebook LM ya lo hace. Podés decirle: “Focalizá en la responsabilidad del conductor”, “Mostrá las líneas de razonamiento relevantes”, “Profundizá esta y descartá aquella”. Un grafo ayuda a pensar. Es una muy buena funcionalidad.

Muchos jóvenes quieren una función que “genere el escrito”. ¿Lo ves viable o peligroso?

Es viable y va a pasar. Yo reniego un poco por motivos filosóficos, no prácticos. Históricamente, los primeros sistemas jurídicos vendían modelos: contrato, cédula, contestación, etc. La IA va a potenciar eso: va a poder generar un borrador como si fueran los modelos, pero personalizados.

Lo importante es que el abogado lo revise críticamente, porque la IA puede incorporar sesgos o bases de baja calidad si no se controla.

¿Qué recomendaciones darías sobre privacidad al usar IA?

Primero: si lo mandaste por mail o WhatsApp, ya no es confidencial. Eso es un mito que hay que desterrar.

Segundo: hay excepciones como expedientes de familia, donde no se muestran datos de menores.

Pero en general, los expedientes son públicos (salvo que el juez los declare reservados).

La confidencialidad depende de qué pasó antes: si entró a Google, si se subió a un servidor, si se compartió en WhatsApp. Eso ya no lo controla el usuario.

¿Es válido que la IA sugiera causas similares, posibles escenarios o matices?

Sí, está muy bien. La ley debería ser predecible. De hecho, el sistema judicial se apoya en antecedentes.

Pero siempre recordando que hay tres jugadores:

- vos,
- la otra parte,
- el juez.

La herramienta predictiva debe estar entrenada con bases confiables. Si le cargás mal contenido (tesis malas, artículos incorrectos), la IA va a reproducir eso.

Por eso recomiendo acotar las fuentes: ejemplo: si analizás un choque de barcos en un lago, tenés que incluir:

- Código de Navegación,
- Ordenanzas locales del municipio,
- Reglamentaciones aplicables.

Si la IA no tiene esos datos, inventa.

¿Qué conviene usar como base para entrenar la IA? ¿Todas las causas? ¿Un estudio específico?

Puede ser un estudio específico. Lo importante es evitar el descontrol de fuentes.

Recomiendo:

- circunscribir territorio (ej.: San Martín)
- limitar temática (ej.: accidentes de tránsito),
- incorporar fallos de Cámara y de primera instancia,
- sumar normativa local.

Entre más acotado, mejor resultado. Si incluyen todo Buenos Aires, ya son 135 municipios con ordenanzas distintas.

Un ejemplo ideal es honorarios mínimos de abogados. Hace pocos años la Cámara no aplicaba el mínimo legal. Con trabajo institucional, empezaron a aplicarlo consistentemente. Eso genera una base: corta, predecible, verificable. Perfecta para entrenar un modelo y ver si predice correctamente.

Les sugiero mirar legislación ambiental. ¿Por qué? es muy actual, es clave para las próximas generaciones, tiene baja dispersión de normas, y los actores son pocos.

A diferencia del tránsito (millones de casos), residuos y ambiente tienen actores muy específicos: CEAMSE, municipios, empresas de disposición final, etc.

¿Creés que los estudios adoptarían nuestra herramienta?

Sí. En los años '90, El Lex Doctor fue un éxito, aun siendo primitivo. Si ustedes logran algo intuitivo, que:

- maneje tiempos,
- cruce causas,
- genere borradores de escritos,
- advierta vencimientos,
- organice información,

va a tener un enorme mercado.

El abogado deberá aportar su toque personal, pero la herramienta será muy útil.

Anexo C: Entrevista a la Dra. Michelle Trouchot

Michelle, para empezar, ¿podrías contarnos quién sos y cuál es tu recorrido profesional?

Me recibí en 2017, así que llevo ya varios años trabajando. En realidad, empecé en 2012, cuando entré a trabajar en un estudio mientras todavía cursaba la carrera. Desde entonces estuve siempre en lo civil. Después, en 2020, hice el posgrado de Derecho e Inteligencia Artificial en la UBA, que fue el primero que hubo en Latinoamérica. Lo dirigía Juan Corbalán, que es un referente enorme en IA y Derecho, con gente del Poder Judicial, abogados de estudios grandes, del interior, funcionarios, de todos lados.

Ahí me empecé a meter en serio en el mundo de la inteligencia artificial. Teníamos materias que mostraban cómo automatizar tareas, cómo aplicar modelos para análisis de expedientes, cómo se podrían acelerar procesos judiciales... incluso vimos experiencias reales que se estaban probando en juzgados.

Hoy tengo dos trabajos: uno en el sector privado, en un estudio, y otro en el Estado, en Fiscalía de Estado. Además, soy Presidenta de la Comisión de Jóvenes Abogados de San Martín. Por eso tengo mucho contacto con abogados jóvenes y sé muy bien qué usan, qué problemas tienen y qué soluciones necesitan.

¿Qué te motivó a meterte en inteligencia artificial?

En 2020 estábamos todos encerrados y yo sentía que todo cambiaba. Veías cómo la Justicia pasaba de lo presencial al mundo digital en cuestión de semanas. Me dije: “algo está pasando y yo no me lo quiero perder”.

El posgrado fue revelador. Hablábamos de automatización de escritos, análisis de jurisprudencia, extracción de hechos con modelos, scoring de expedientes, predicción de resultados, detección de patrones en demandas masivas... cosas que hoy ya son más comunes, pero en ese momento eran revolucionarias.

Lo que me impactó fue ver que la IA no venía a reemplazarnos, sino a potenciar lo que hacemos. El que sepa usarla, va a trabajar diez veces más rápido. El que no, va a quedar atrás.

¿Tenés colegas del área IA + Derecho que puedan servir para complementar nuestra investigación?

Sí, un montón. Entre el posgrado, los grupos de trabajo y la Comisión, tengo contactos de abogados jóvenes y no tan jóvenes que trabajan específicamente en IA para estudios jurídicos, para la Justicia, para empresas, incluso gente que construye modelos propios. Si quieren entrevistar a especialistas, tengo para recomendarles mínimo cinco o diez.

Además, administro un grupo de WhatsApp con 213 abogados jóvenes. Ahí hay gente que sabe muchísimo de tecnología legal, que seguro estaría encantada de participar.

¿Cómo usás la IA hoy, en tu día a día como abogada?

La uso muchísimo para tareas repetitivas. Por ejemplo, las negativas: antes me tomaban media hora, ahora le pongo: “Negame esto con este estilo, con esta estructura, con este tono”, y en un minuto lo tengo.

Para mí es una herramienta de productividad. También sirve para pliegos de posiciones. Te adelanta la estructura, vos lo revisás, corregís y listo.

No estoy explotando todo lo que podría, porque hay opciones mucho más avanzadas: hay modelos que entrenás con tus propios escritos y después te generan demandas enteras alineadas a tu estilo. Sé que estudios grandes ya los usan. Para mí es una herramienta que, bien usada, te hace ganar tiempo de vida.

¿Viste implementaciones reales de IA en juzgados?

Sí, y muy fuertes. En 2020 se hizo una prueba en un juzgado de Morón donde automatizaron declaratorias de herederos en sucesiones. Básicamente, vos tenías un expediente con toda la información cargada, la IA la procesaba y generaba automáticamente la declaratoria. Tardaba dos minutos. Después un humano lo revisaba. Y ya está. Eso acorta muchísimo los tiempos.

Y, ojo: cuando la Justicia mejora tiempos, nosotros también. Porque si ellos resuelven antes, avanzan nuestras causas. Por eso la IA en la Justicia no es solo para el Poder Judicial, sino para nosotros también.

Para entender los puntos de dolor, ¿cómo describirías tu flujo real de trabajo?

Todos los días arranco con las notificaciones electrónicas. Ahí puede haber desde un simple traslado hasta una sentencia. Después reviso expedientes. Después veo qué plazos están corriendo. Si hay que contestar, redacto. Después firmo digitalmente (en Provincia seguimos con token). Tengo que mandar cédulas, coordinar audiencias, hablar con clientes, con peritos, con el otro abogado... todo eso en paralelo.

Además está lo administrativo: cargar causas, seguir estados, anotar vencimientos. Es un trabajo que nunca termina. Cada vez que la Justicia acelera algo, te llegan más cosas. Con la digitalización, todos trabajamos más.

¿La digitalización aumentó la cantidad de causas que podés manejar?

Sí, muchísimo. Antes, si querías litigar en otro departamento judicial, tenías que viajar. Hoy entrás a una audiencia por Zoom y listo.

Entonces, colegas me dicen: “¿Podés cubrir una vista médica mañana que vos estás cerca?”, “¿Podés ingresar esta cédula porque tenés el token?”. Y eso hace que te lluevan causas.

Lo digital facilita el trabajo, pero también multiplica las demandas sobre vos.

¿Cómo hacés para no perderte entre tantas causas, plazos y documentos?

Con organización. Yo tengo mis listados propios, mis cuadros, mis resúmenes. Porque si dependés solo de la memoria, estás perdida.

En Fiscalía, por ejemplo, tengo 133 expedientes. Me armé un Excel donde pongo:

- etapa procesal,
- qué se presentó,
- qué respondió el contrario,
- si hay audiencia,
- si hay apertura a prueba,
- si hay sentencia pendiente.

Sin ese seguimiento, no podría trabajar. Literalmente, no me acuerdo nada de memoria. El secreto está en no solamente ver una causa, sino que eso también se alimente con las otras causas que tenés y te prenda una luz amarilla o roja.

¿Qué tan importante es clasificar por estados procesales?

Es fundamental. Sin estados, no podés priorizar. Para mí las categorías básicas son:

- Inicio / demanda / contestación
- Apertura a prueba
- Producción de prueba
- Sentencia
- Cámara

Cuando tengo mucha carga, lo primero que hago es aislar todas las que están en prueba, porque esas son las que más trabajo tienen. Si ustedes arman esto bien, ya ganan muchísimo.

Como usuaria avanzada de Lex Doctor, ¿qué funciones usás y qué le falta?

Lex Doctor para mí es muy bueno. Lo uso desde hace diez años. Tiene: estados de causa, agenda, carga de movimientos, posibilidad de trabajar varios usuarios, historial interno.

Pero tiene problemas:

- Es caro. La licencia hoy puede estar 300–400 mil pesos fácilmente.
- No es intuitivo para abogados jóvenes.
- No está pensado para integrar IA.
- Se quedó viejo en interfaz.

Si ustedes ofrecen algo más moderno y accesible, creo que hay muchísimo mercado, sobre todo entre abogados jóvenes.

¿Es importante poder compartir causas con distintos niveles de acceso?

Sí, clave. A mí me pasa que me voy de vacaciones o estoy enferma, y mi compañera tiene que tomar mis causas. Pero no quiero que vea todo el estudio. Solo quiero que vea mis causas. Esa granularidad es esencial para la práctica real.

¿Te imaginás trabajar desde el celular?

Para trabajar no. No voy a redactar una demanda en el teléfono. Pero para ver un resumen, sí.

Por ejemplo: Tengo audiencia en 10 minutos y quiero recordar qué dijo la otra parte. Quiero ver qué pasó ayer. Quiero revisar vencimientos.

Eso sí lo haría desde el celular.

¿Te pasa abrir un expediente y no recordar nada?

Todo el tiempo. Con 150 causas, es imposible recordar. Por eso me hago resúmenes, listas de movimiento, carpetas.

Una vista que te muestre de forma inmediata:

- los últimos movimientos,
- qué presentaste,
- qué contestó el otro,
- qué vencimientos hay,

sería espectacular.

¿Te serviría una visualización donde aparezcan actores, documentos y relaciones?

Sí, para mí eso es muy útil. Sobre todo, para causas complejas. Hacer clic en un actor y ver:

- qué presentó,
- qué documentos se relacionan,
- qué hechos lo involucran.

Es mucho más visual y rápido de entender.

¿Qué herramientas ves que usan los abogados jóvenes?

Word y Drive. Así, tal cual. No usan Lex Doctor porque es caro. Entonces arman carpetas: “Rafael c/ X”, “Micaela c/ Y”, “Expediente tal”. Pero para manejar 100 causas, eso no sirve. No escala. No está pensado para lo jurídico. Por eso creo que su herramienta tiene muchísimo sentido.

Para nosotros el público principal son abogados jóvenes y estudiantes avanzados. ¿Coincidís?

Sí. Desde 20 hasta 35 más o menos. Un estudiante de primer año capaz no lo usa porque no litiga aún. Pero desde cuarto año en adelante, sí.

¿Competiríamos contra Lex Doctor?

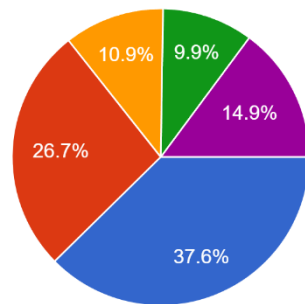
Sí, competirían. Y está perfecto.

Pero Lex Doctor es caro y viejo en interfaz. Si ustedes ofrecen algo: moderno, accesible, con IA, orientado a jóvenes, van a tener mucho mercado. Lex Doctor no cubre esa necesidad.

Anexo D. Datos de la Encuesta

¿Cuál es tu situación actual?

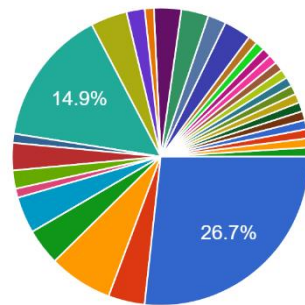
101 respuestas



- Estudiante intermedio de Derecho (3° año o menos)
- Estudiante avanzado/a de Derecho (4° año en adelante)
- Egresado/a reciente (menos de 2 años)
- Abogado/a en ejercicio (de 3 a 5 años de experiencia)
- Abogado/a en ejercicio (más de 5 años)

¿En qué universidad estudiaste o estudias actualmente?

101 respuestas

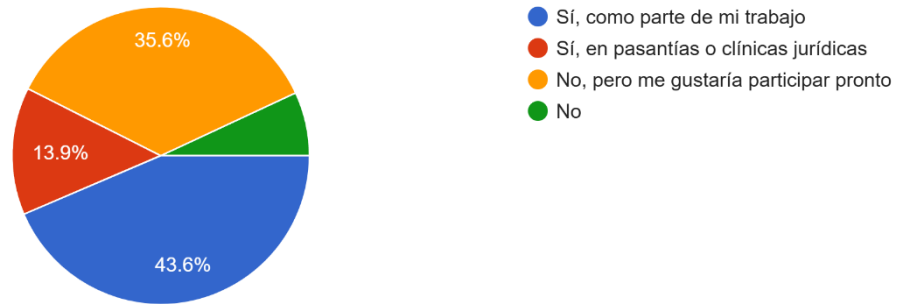


- Universidad Nacional de Buenos Aires...
- Universidad Nacional de Córdoba (UN...)
- Universidad Nacional de Cuyo (UNCU)
- Universidad Nacional de Gral. San Ma...
- Universidad Nacional de La Pampa (U...)
- Universidad Nacional de la Plata (UN...)
- Universidad Nacional del Litoral (UNL)
- Universidad Nacional de Lomas de Za...

▲ 1/5 ▼

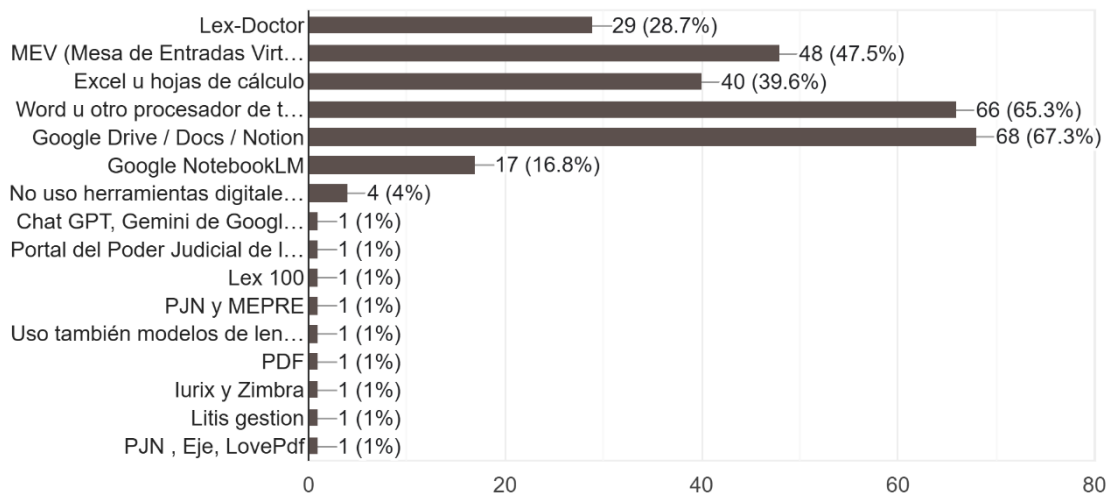
¿Participás actualmente en causas judiciales reales?

101 respuestas



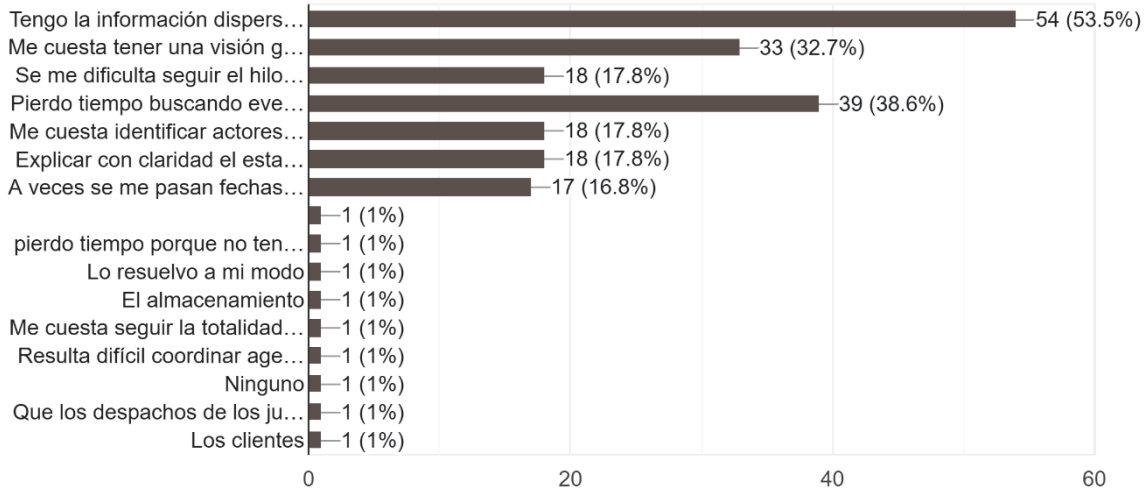
¿Qué herramientas usás actualmente para gestionar tu información legal?

101 respuestas



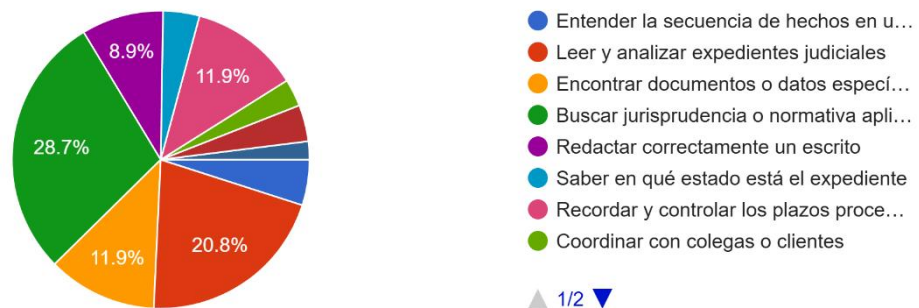
¿Qué tipo de problemas o desafíos encontrás en tu día a día?

101 respuestas



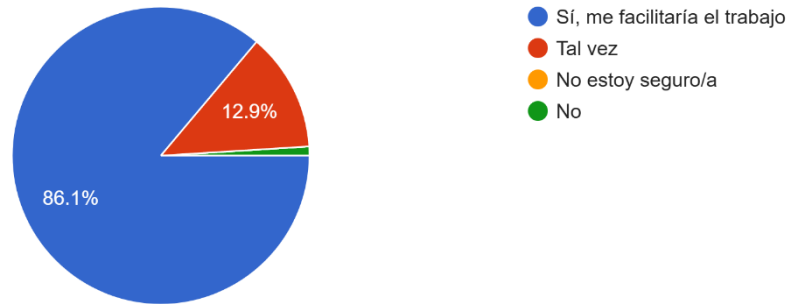
¿Cuál de estas tareas te resulta más difícil o te lleva más tiempo?

101 respuestas



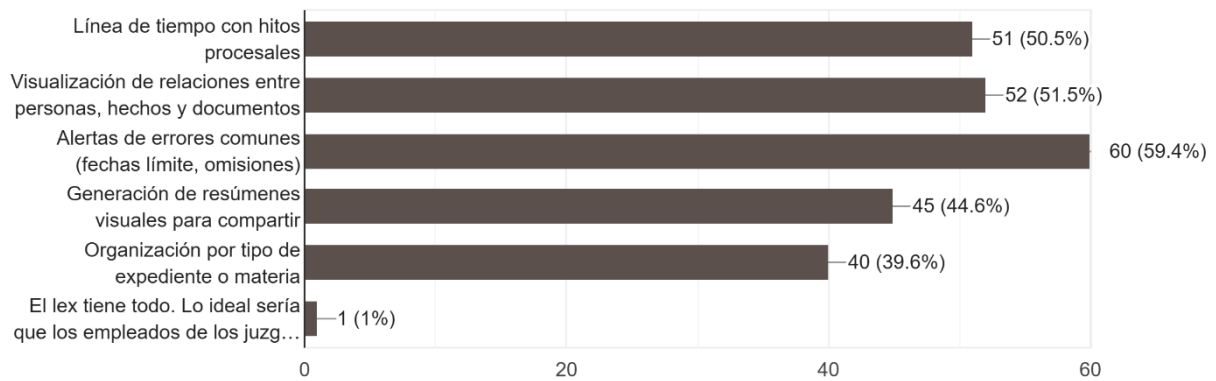
¿Consideras útil una herramienta digital que permita visualizar tus causas de forma estructurada?

101 respuestas



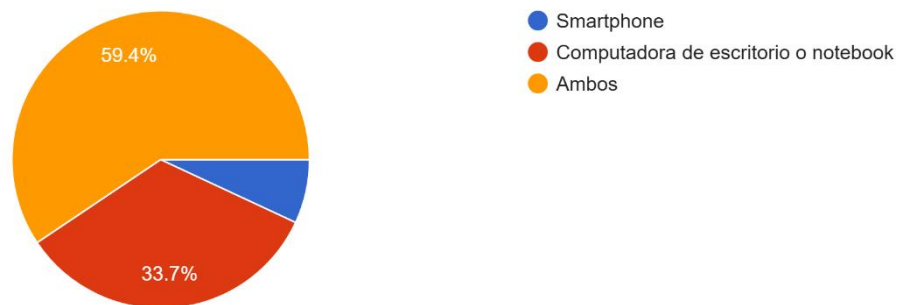
¿Qué funcionalidades priorizarías? (seleccioná hasta 3)

101 respuestas



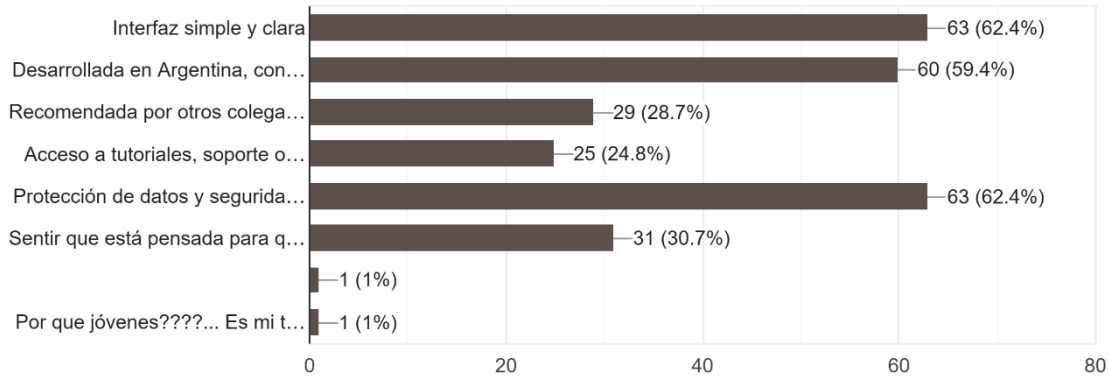
¿Desde qué dispositivo preferís acceder a una herramienta como esta?

101 respuestas



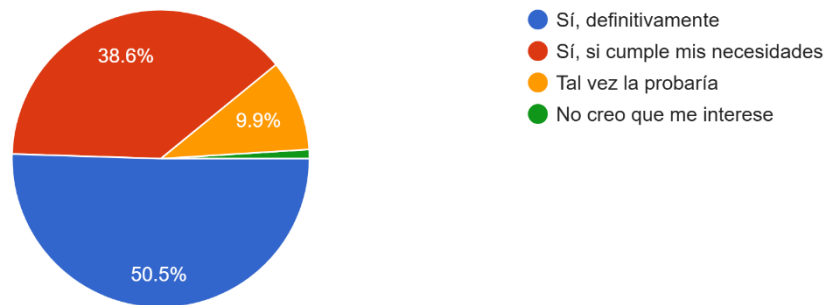
¿Qué características te darían confianza en una app legal para abogados juniors?

101 respuestas



¿Usarías una versión gratuita de esta app para organizar tus causas?

101 respuestas



Anexo E. Glosario de requerimientos

Con el fin de contextualizar los términos que estructuran los requerimientos funcionales del sistema, se presenta a continuación un glosario del dominio. Cada concepto se encuentra vinculado a una necesidad identificada en el análisis previo y se relaciona directamente con los módulos de la aplicación que se detallarán en los requerimientos.

Causa / expediente

Unidad de trabajo legal a gestionar dentro del sistema. Representa un expediente judicial o causa en curso, con sus datos básicos (carátula, fuero, jurisdicción, descripción). Los usuarios deben poder crear, editar y eliminar causas, así como centralizar en ellas todos los eventos, documentos y estados asociados.

Evento procesal

Hito que ocurre en el marco de una causa, asociado a una fecha y un tipo específico (ej.: presentación de demanda, audiencia preliminar, sentencia). Constituyen los elementos principales a representar en la **línea de tiempo**. Los eventos pueden ser creados manualmente por el usuario o sugeridos automáticamente por el sistema en función de reglas básicas o de IA Generativa.

Estado procesal

Etapa en que se encuentra la causa dentro de su ciclo de vida (ej.: iniciado, en prueba, con sentencia). El sistema debe poder reflejar dicho estado, actualizarlo según el avance de los eventos y ofrecerlo de manera visible tanto en el panel global como en la vista de detalle de cada causa.

Jurisprudencia

Fallos o antecedentes legales relevantes que pueden asociarse a una causa o a un evento específico. El sistema prevé la vinculación manual de jurisprudencia a expedientes, así como la integración con un módulo de recomendación conectado con IA.

Línea de tiempo

Visualización cronológica de los eventos procesales de cada causa. Permite al usuario comprender de manera secuencial lo ocurrido, lo que está pendiente y lo que ocurrirá. Es el eje principal del proyecto y debe ser interactiva (zoom, paneo, detalles emergentes).

Grafo relacional

Representación gráfica de los actores, hechos y documentos asociados a una causa. Facilita la comprensión de las relaciones entre los distintos elementos del expediente y ofrece una vista complementaria a la línea de tiempo.

Módulo de notificaciones

Subsistema encargado de alertar al usuario sobre eventos próximos o vencidos. Su objetivo es reducir el riesgo de omisiones procesales y mejorar la productividad diaria.

Asistente de IA generativa

Módulo que asiste al usuario mediante la generación de resúmenes automáticos de causas o documentos, la detección de inconsistencias y la sugerencia de pasos faltantes. Se apoya en técnicas de inteligencia artificial generativa para aportar un nivel cognitivo adicional al sistema.

Módulo de gestión de documentación

Espacio centralizado donde los usuarios pueden asociar documentos digitales (escritos, dictámenes, oficios) a las causas y eventos correspondientes. Complementa la jurisprudencia y el grafo relacional, permitiendo búsquedas rápidas y consultas estructuradas.

Módulo de inteligencia artificial asistida

Subsistema potenciado por modelos de lenguaje de gran escala (LLM) orientado a asistir al profesional en la redacción, revisión y análisis de causas judiciales. Este módulo permite detectar errores, sugerir pasos faltantes, generar resúmenes automáticos y agilizar la confección de estructuras de datos jurídicas. Las funcionalidades se integran dentro de distintos componentes del sistema, de modo que las estructuras internas de datos puedan aprovechar las capacidades de los LLM para mejorar la eficiencia y coherencia del flujo de trabajo. En determinados contextos específicos, se podrán emplear algoritmos predictivos complementarios que refuercen la capacidad de análisis del módulo.

Módulo de búsqueda jurisprudencial avanzada

Motor de búsqueda semántica basado en técnicas de vectorización de texto y recuperación aumentada por inteligencia artificial (RAG). Permite consultar jurisprudencia almacenada en una base de datos vectorizada, ofreciendo resultados relevantes más allá de coincidencias literales. A su vez, posibilita realizar preguntas en lenguaje natural mediante un modelo LLM que razona sobre los resultados y asiste al usuario en la interpretación de la información jurídica recuperada.

Anexo F. Vistas Funcional: Causa Específica

The screenshot displays a user interface for a legal case summary. At the top, it indicates 'CONTENIDO GENERADO POR IA' and 'Resumen de la causa' with a 'Regenerar resumen' button. Below this, a paragraph of text provides details about a legal case (CEREA LKO SA c/ ROJAS, Stella Maris s/ CONSIGNACION). The main section is titled 'VERIFICACIÓN AUTOMÁTICA' and contains four expandable items: 'OMISION CRITICA', 'INACTIVIDAD POTENCIAL', 'VENCIMIENTO IMPORTANTE', and 'DATO FALTANTE'. Each item includes a small icon and a count of findings.

Figura 18. Vista funcional del módulo de resumen: permite obtener un resumen general de la causa, redactado de manera clara y legible. Además, incorpora verificaciones automáticas que identifican posibles errores gramaticales, inconsistencias en la información y avisos relacionados con vencimientos relevantes, con el fin de mejorar la calidad y completitud del análisis inicial. (Elaboración propia, 2025)

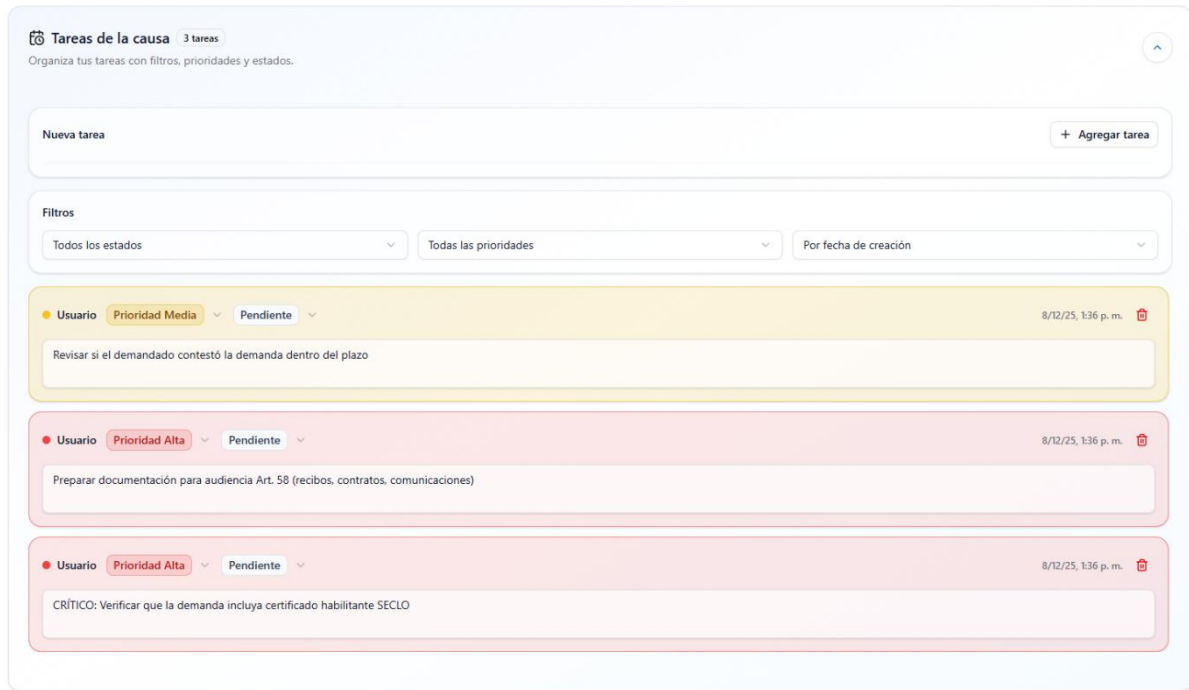


Figura 19. Vista funcional del módulo de tareas: El módulo permite realizar el seguimiento de las tareas asociadas a la causa, incluyendo sus estados y prioridades. Esta funcionalidad facilita la organización del trabajo y favorece la planificación efectiva de las actividades del profesional. (Elaboración propia, 2025)



Figura 20. Vista funcional del módulo de trazabilidad: El sistema muestra los últimos diez movimientos registrados en la causa mediante una interfaz visualmente clara y

estructurada. En cada registro se detallan la acción realizada, la entidad afectada, la persona responsable y la fecha correspondiente. Asimismo, se dispone de un botón “Ver todos” que permite acceder al historial completo de trazabilidad y revisar cada modificación efectuada a lo largo del proceso. (Elaboración propia, 2025)

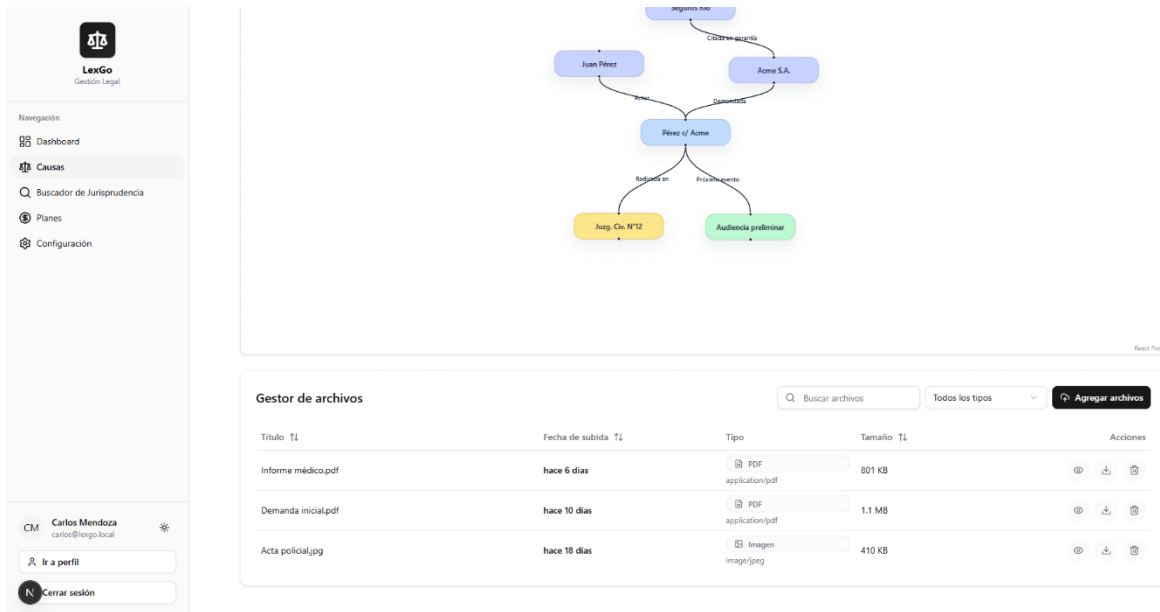


Figura 21. Vista funcional de causa específica: Gestor y visualizador centralizado de documentos, archivos y jurisprudencia. (Elaboración propia, 2025).

Anexo G. Vista Funcional: Dahnboard

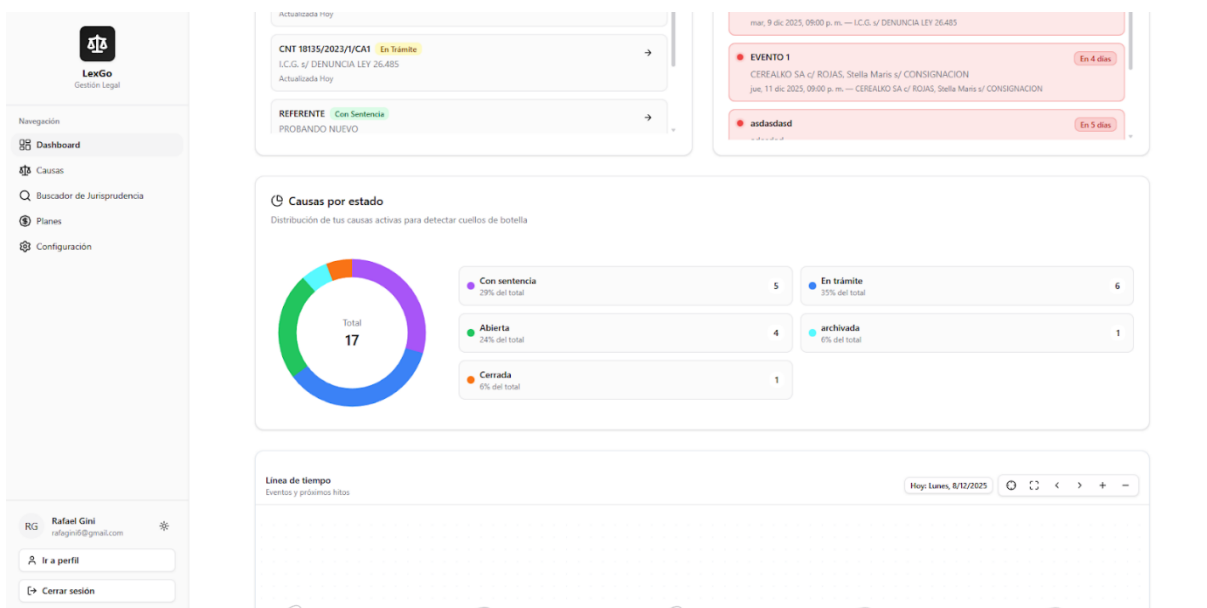


Figura 22. Vista funcional de Dashboard: Chart Pie de causas por estado, le da un vistazo rápido al abogado de la cantidad y tipos de causa que está trabajando. (Elaboración propia, 2025)

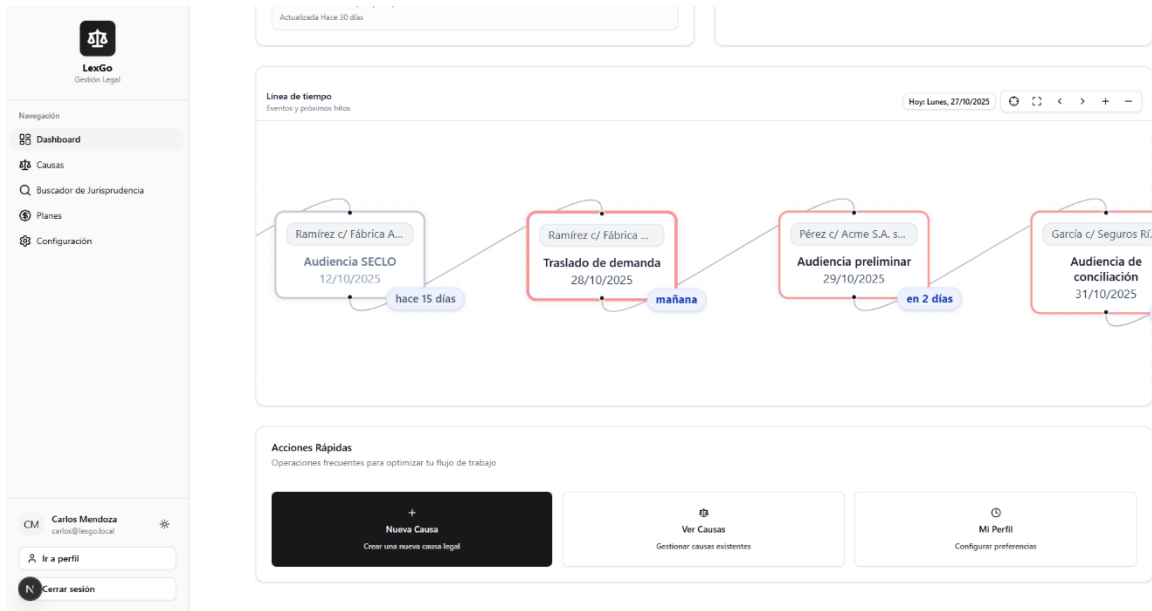


Figura 23. Vista funcional de Dashboard: Línea de tiempo general de todos los casos para resumir los próximos días del abogado y una barra de accesos rápidos. (Elaboración propia, 2025)

Anexo H. Arquitectura ideal

El **frontend** se despliega en **Vercel**, que ofrece distribución global mediante CDN y soporte integrado para **despliegue continuo** desde GitHub, garantizando baja latencia y versionado automático de entornos.

El **backend** se estructura sobre un conjunto de **servicios gestionados de AWS**. El código fuente se almacena en **GitHub** y se integra a través de un pipeline de **CI/CD** compuesto por **AWS CodePipeline** y **AWS CodeBuild**, que automatizan la construcción, pruebas y publicación de imágenes en **Amazon Elastic Container Registry (ECR)**. Dichas imágenes se ejecutan en **Amazon Elastic Container Service (ECS)** bajo el modo **Fargate**, eliminando la necesidad de administrar servidores.

El tráfico entrante es canalizado por un **Application Load Balancer (ALB)** ubicado en la **subred pública**, que aplica ruteo por contenido y distribuye las solicitudes entre **ECS** (para endpoints de aplicación) y **AWS Lambda** (para funciones serverless). Las funciones

Lambda se despliegan dentro de la **subred privada de la VPC**, lo que permite su acceso directo a **Amazon Aurora Serverless v2** y a los **buckets S3**, utilizados respectivamente para el almacenamiento de datos estructurados y documentos del usuario.

AWS Lambda actúa como componente de **procesamiento bajo demanda**, ejecutando tareas asíncronas y automatizaciones específicas como el análisis de documentos, la generación de *embeddings* y la actualización de datos de IA. Junto con **ECS Fargate** y **Aurora Serverless v2**, adopta un modelo **serverless** que **escala automáticamente en función de la demanda**, ajustando la capacidad de cómputo únicamente cuando existen solicitudes activas. Este enfoque permite **evitar costos por instancias ociosas, pagar solo por el tiempo efectivo de ejecución** y mantener un **alto nivel de disponibilidad** sin intervención manual, optimizando tanto el rendimiento como la eficiencia operativa del sistema.

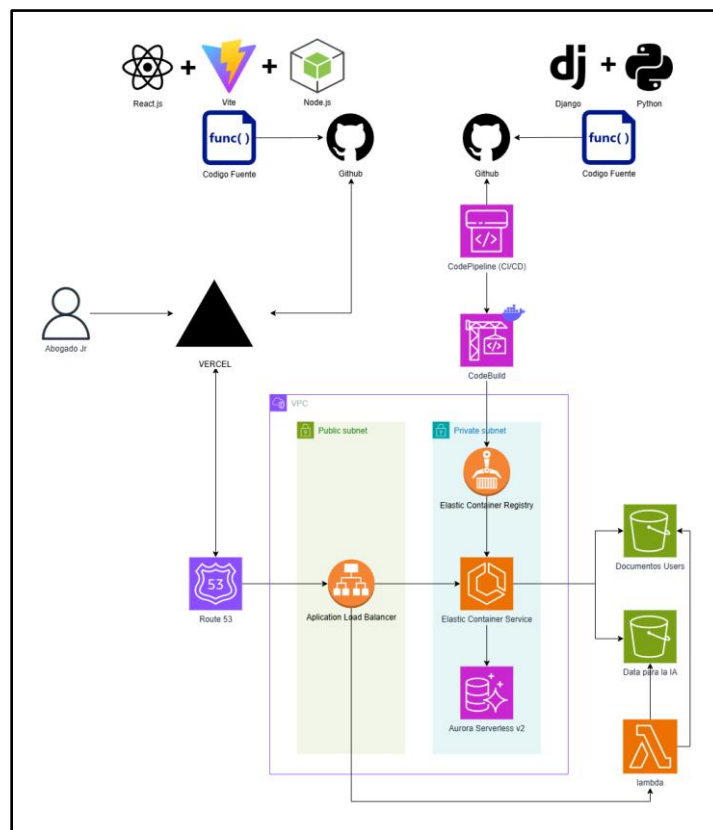


Figura 24. *Arquitectura proyectada para el entorno de producción.*
(Elaboración propia, 2025)

Anexo I. Componentes del Asistente de Jurisprudencia Basado en Recuperación Aumentada (RAG)

Los componentes principales son los siguientes:

- **Backend de recuperación semántica (R):** responsable de localizar fragmentos relevantes en el corpus jurídico empleando embeddings vectoriales y filtros contextuales.
- **Motor de generación (G):** basado en modelos de lenguaje (GPT-4o), encargado de sintetizar la información y elaborar respuestas explicativas.
- **Gestor de conversaciones:** mantiene el registro completo de las interacciones (usuario–asistente), preservando el contexto en sesiones sucesivas.
- **Control de idempotencia:** evita duplicaciones o inconsistencias en escenarios de reenvío o error de red.
- **Módulo de automatización Selenium:** permite la extracción dinámica de información desde fuentes jurídicas públicas, como InfoLeg o SAIJ, en los casos en que los documentos no se encuentran indexados previamente. Este proceso de web scraping automatizado opera bajo filtros de dominio autorizados, garantizando que los datos provengan exclusivamente de repositorios jurídicos argentinos.
- **Persistencia en base de datos:** mediante los modelos Conversation, Message e IdempotencyKey, se conserva el historial completo de interacciones, lo que facilita la trazabilidad y auditoría del sistema.

Anexo J. Resultados del modelo de Machine Learning

La matriz de confusión muestra que el clasificador identifica correctamente la mayoría de las instancias en cada una de las etapas principales. Se observa un desempeño perfecto en `demanda_inicial` (3 aciertos) y un muy buen rendimiento en prueba (2 aciertos), sin confusiones hacia otras clases. La etapa `secl` también fue clasificada correctamente en una ocasión. La única confusión relevante aparece en la clase `sentencia`, donde 1 caso fue erróneamente clasificado como `prueba`. En general, el modelo muestra una buena separación entre categorías, con errores mínimos y concentrados en clases conceptualmente cercanas.

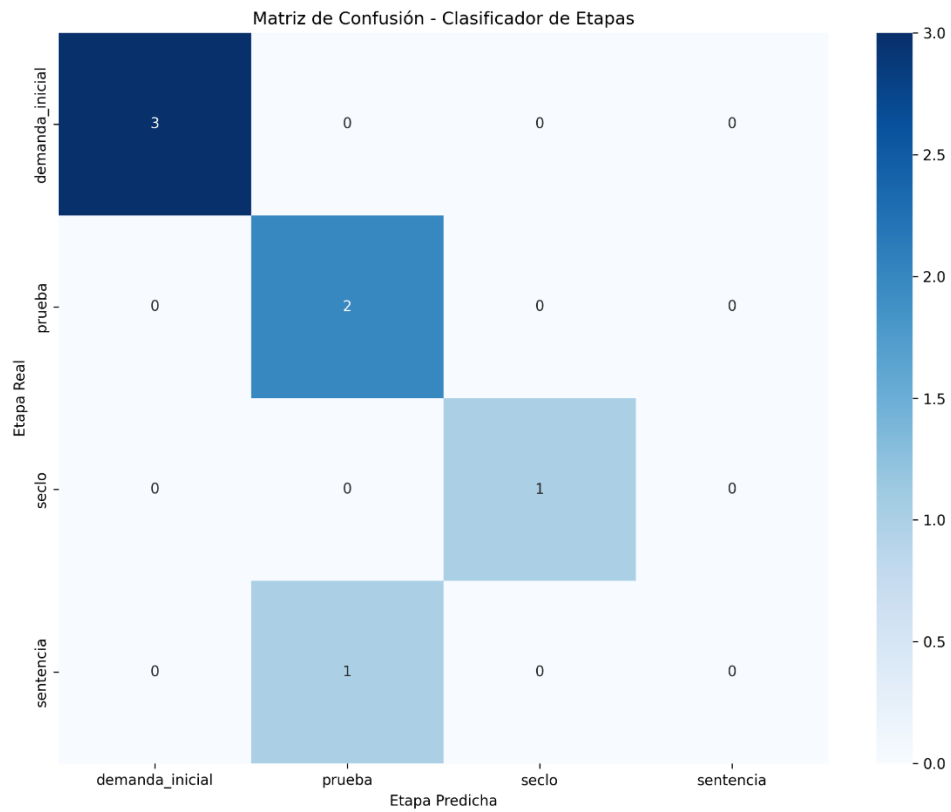


Figura 25. Matriz de Confusión. (Elaboración propia, 2025)

Anexo K. Trello

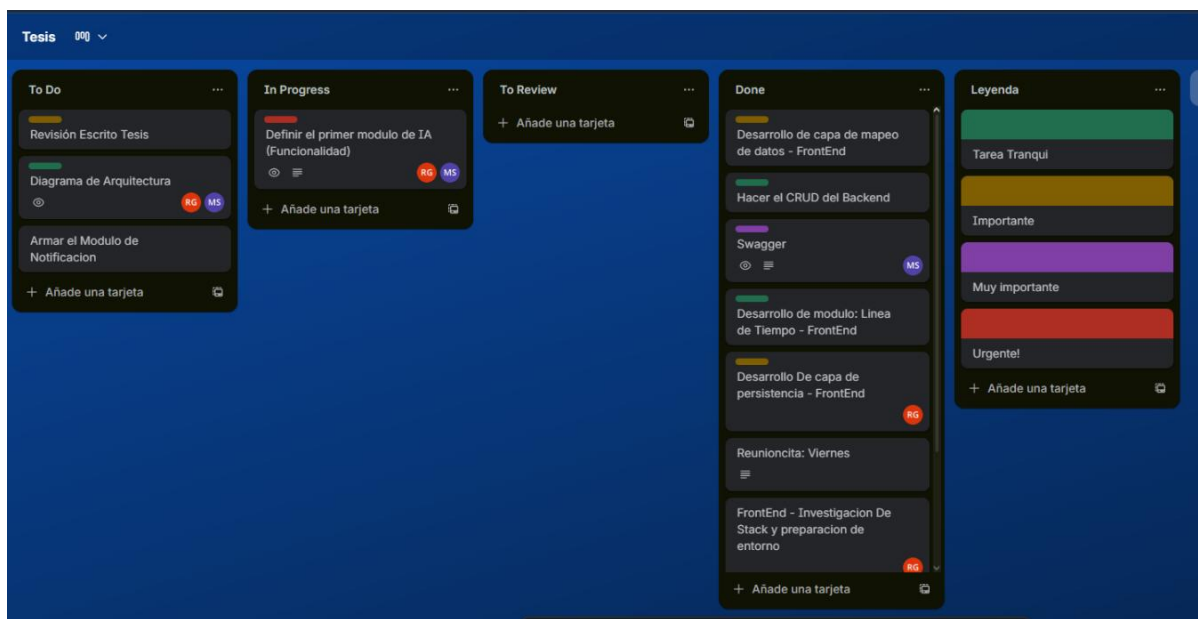


Figura 26. Trello utilizado durante el proyecto. (Elaboración propia, 2025)

Anexo L. Entrevista a Lucía Suárez sobre prueba de la aplicación

¿Qué te pareció el dashboard inicial? ¿Te ayudó a orientarte y a entender rápidamente el estado general de tus causas y la que estabas evaluando?

El dashboard me gustó porque me dio una vista general sin tener que entrar en cada expediente. Pude ver rápido qué causas estaban activas, en qué etapa estaban y cuáles tenían actualizaciones recientes. Eso para el estudio es clave, porque a veces manejamos muchas causas en paralelo y necesitamos un pantallazo rápido. Además, los colores y gráficos me ayudaron a identificar prioridades sin leer demasiado. Se siente como una herramienta que realmente organizaría mi trabajo del día a día.

¿Te resultó útil el grafo de relaciones para entender cómo se vinculan las partes, los documentos y los eventos dentro del caso?

Sí, me resultó útil. Al principio lo miré por arriba, pero cuando empecé a ver las conexiones me ayudó a entender rápido quién era quién en el expediente y cómo se relacionaban los eventos entre sí. En los estudios a veces tenemos que armar esos mapas mentalmente, así que verlo visualizado me ordenó bastante. Me sirvió sobre todo para identificar rápido los puntos clave del caso sin leer todo.

¿Sentís que, al ingresar al caso ya creado, la estructura general (partes, eventos procesales, documentos) se entiende rápidamente sin que necesites buscar demasiado?

Sí, la verdad es que me resultó bastante intuitivo. Veo todo ordenado por secciones y eso me ayuda mucho. En menos de un minuto ya entendí cuál era la etapa procesal, quiénes eran las partes y qué había pasado hasta ese momento. No me sentí perdida ni tuve que volver atrás para ubicarme.

¿Los nombres de las secciones y la jerarquía visual te ayudaron a identificar la estructura del caso sin leer todo el expediente?

Sí. Me ayudó especialmente que las secciones estén diferenciadas y tengan títulos claros. Por ejemplo, “Tareas”, “Eventos”, “Línea de Tiempo”, eso para nosotros es clave.

Siento que si tuviera que explicarle el caso a un abogado del estudio podría hacerlo sin necesidad de abrir el PDF original.

¿Cuánto tiempo sentís que tardaste en comprender de qué trataba el caso presentado?

Aproximadamente entre 2 y 3 minutos. Me sorprendió porque normalmente los casos llevan más tiempo de lectura, pero acá la interfaz ayuda a resumir lo importante.

Cuando subiste un documento que ya conocías, ¿el flujo para generar la causa te resultó claro y lógico?

Sí. Seleccioné el archivo, esperé que procese y después vi cómo se armó la estructura automáticamente. Me gustó el “caminito” para realizarlo porque en el estudio trabajamos mucho con PDFs y esto se parece a “subir un escrito al sistema”, así que me resultó familiar.

Cuando elegiste la opción de “Aplicar Machine Learning”, ¿entendiste qué iba a hacer el sistema?

Entendí que iba a clasificar partes del documento en etapas procesales y extraer lo más relevante. Aunque me parecería útil que agreguen un texto en donde se explique qué hace esa opción.

¿El resultado generado automáticamente coincidió con tu expectativa del caso que ya conocías?

Sí, bastante. La parte de demanda la detectó perfecto, también los eventos pasados. En algunos detalles menores capaz yo hubiera agregado algo más, pero en general estaba bien clasificado y me ayudó a ver el caso sin tener que releer todo.

**¿Lograste entender fácilmente lo que el sistema generó a partir del PDF?
Por ejemplo: la línea del tiempo, los eventos, las partes y los documentos.**

Sí. De hecho, creo que la línea del tiempo fue lo que más me ayudó. Ver todo organizado por fechas y etapas me dio una visión general muy rápida. Los documentos estaban en un orden razonable y no tuve que adivinar qué era cada cosa.

Probando en celular, ¿te resultó igual de claro que en la versión web? ¿O hubo algo que te costó más ver?

En celular está todo más compacto, pero igual entendí la estructura. Lo único que comentaría es que no sé si lo utilizaría tanto, me gusta más en computadora.

¿Te sentiste guiada por la interfaz o en algún momento dudaste qué hacer?

Me sentí guiada. El botón de subir documento está donde lo esperaba, y la indicación de “Generar causa” es clara. No tuve dudas sobre los pasos.

¿Considerás que, con esta interfaz, podrías analizar causas más rápido que como lo hacés hoy en tu estudio?

Sí, definitivamente. Nosotros abrimos varios PDFs, anotamos aparte, es más manual. Esto me permitió ver lo importante sin tanto trabajo. Me ahorraría bastante tiempo.

Si tuvieras que calificar tu nivel de comprensión de la causa en menos de 7 minutos, ¿dirías que fue: Muy alta, Alta, Media, Baja o Nula?

Diría alta. No tuve que releer nada y entendí el contexto completo.

¿Sentís que podrías recomendar esta herramienta a otro estudiante o abogado junior para entender un caso rápido?

Sí, seguro. Personalmente está hecha para mi manera de trabajar, y me ayuda a ubicarme rápido en un expediente.

10. Índice de Figuras

N° de Figura	Descripción	Origen
1	Análisis de los elementos existentes en la propuesta que definen su diferencial competitivo.	Elaboración Propia, 2025.
2	Comparación de la solución propuesta (LexGO) respecto a soluciones presentes en el mercado.	Elaboración propia, 2025.
3	Diagrama de contexto narrativo.	Elaboración propia, 2025.
4	Vista funcional de gestión de causas.	Elaboración propia, 2025.
5	Vista funcional de gestión de causas.	Elaboración propia, 2025.
6	Vista funcional de causa específica: pantalla que integra header, resumen IA, módulo de tareas, trazabilidad.	Elaboración propia, 2025.
7	Vista funcional de causa específica: módulo de línea de tiempo.	Elaboración propia, 2025.
8	Vista funcional de causa específica: Gestor de actores y eventos.	Elaboración propia, 2025.
9	Vista funcional de causa específica: Grafo relacional de actores, eventos y documentos.	Elaboración propia, 2025.
10	Vista funcional de Dashboard.	Elaboración propia, 2025.
11	Vista funcional del Asistente.	Elaboración propia, 2025.
12	Modelo de datos simplificado del sistema.	Elaboración propia, 2025
13	Expansión del modelo de datos.	Elaboración propia, 2025
14	Modelo de datos integral del sistema.	Elaboración propia, 2025

15	Arquitectura del sistema implementada para el prototipo funcional (MVP).	Elaboración propia, 2025
16	Funcionamiento general de Ngrok.	NGROK INC., 2025
17	Historias de usuario realizadas para el proyecto.	Elaboración propia, 2025
18	Vista funcional del módulo de resumen.	Elaboración propia, 2025
19	Vista funcional del módulo de tareas.	Elaboración propia, 2025
20	Vista funcional del módulo de trazabilidad.	Elaboración propia, 2025
21	Vista funcional de causa específica: Gestor y visualizador centralizado de documentos, archivos y jurisprudencia.	Elaboración propia, 2025
22	Vista funcional de Dashboard: Chart Pie de causas por estado.	Elaboración propia, 2025
23	Vista funcional de Dashboard: Línea de tiempo general de todos los casos.	Elaboración propia, 2025
24	Arquitectura proyectada para el entorno de producción.	Elaboración propia, 2025
25	Matriz de Confusión.	Elaboración propia, 2025
26	Trello utilizado durante el proyecto.	Elaboración propia, 2025

11. Índice de Tablas

N° de Figura	Descripción	Origen
I	Principales necesidades de los abogados juniors.	Entrevista propia a Michelle Trouchet, Anexo C
II	Comparación entre aplicaciones jurídicas existentes	Elaboración Propia, 2025
III	Requerimientos centrales	Elaboración Propia, 2025
IV	Unidades de consumo promedio por usuarios activos.	Elaboración Propia, 2025
V	Costos mensuales aproximados.	Elaboración Propia, 2025
VI	Proyección comparativa de resultados económicos por volumen de usuarios.	Elaboración Propia, 2025
VII	Análisis económico del Flujo Nominal con crecimiento moderado.	Elaboración Propia, 2025
VIII	Análisis del Flujo Financiero Descontado en los primeros 5 años con crecimiento moderado.	Elaboración Propia, 2025
IX	Métricas clave para la toma de decisiones de inversión.	Elaboración Propia, 2025
X	Análisis del Flujo Nominal en los primeros 5 años con crecimiento acelerado.	Elaboración Propia, 2025
XI	Análisis del Flujo Financiero Descontado en los primeros 5 años con crecimiento acelerado.	Elaboración Propia, 2025
XII	Métricas para el análisis económico del crecimiento acelerado.	Elaboración Propia, 2025
XIII	Estructura de una causa judicial y su interpretación en la App.	Elaboración Propia, 2025