

# PRESENTACIÓN CAPSTONE

Definición Proyecto



***Samira Awad - Constanza Flores - Astrid Gonzalez***



01

EQUIPO





**Samira Awad**  
**Development Manager**

Lidera la arquitectura y desarrollo backend, integra la IA y asegura la calidad técnica del código.



**Astrid González**  
**Product Owner + Developer**

Define y prioriza el backlog, válida entregas y asegura que el producto cumpla la visión del proyecto.



**Constanza Flores**

**Scrum Master + Developer**

Facilita las ceremonias Scrum, elimina impedimentos y vela por la aplicación correcta de la metodología.

02

## DESCRIPCIÓN





+4.000

Estudiantes de Derecho  
en Chile rinden el  
**examen de grado**  
anualmente.

30%

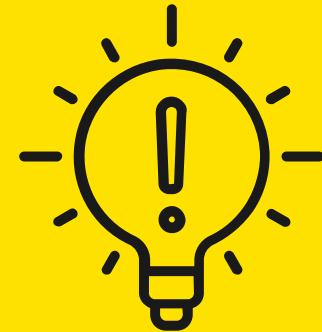
**Reaprueba** anualmente

Fuente: Estado Diario (2023)



## PROBLEMAS IDENTIFICADOS

- Falta de **constancia** en el estudio
- Dificultad para **identificar puntos débiles** específicos
- Bancos de preguntas estáticos sin **seguimiento personalizado**
- **Tiempos** de estudio



## PROPUESTA DE SOLUCIÓN

*App móvil que genera preguntas personalizadas, permite responder por texto o voz, corrige automáticamente y ofrece **seguimiento personalizado** actuando como un tutor digital.*

**GRADO CERRADO**

**CAPÍTULO TERMINADO**





**03**

**OBJETIVOS**

# OBJETIVO GENERAL

**Contribuir** a mejorar la preparación de los estudiantes de Derecho en Chile para el **examen de grado**, abordando la falta de herramientas personalizadas que faciliten la práctica constante, **la detección de debilidades y el seguimiento del progreso.**



# OBJETIVOS ESPECIFICOS



**Identificar las principales dificultades** académicas que enfrentan los estudiantes.



Proporcionar un sistema de **evaluación automática y retroalimentación inmediata.**



Ofrecer un **entorno accesible y práctico.**



Favorecer la continuidad del estudio mediante **recordatorios y recomendaciones**

04

# ALCANCE Y LIMITACIONES



# ALCANCE DEL PROYECTO

- **App móvil multiplataforma**
- Solo serán consideradas las materias **Civil y Procesal**
- **Generación automática de preguntas** con IA
- **Interacción por voz** (reproduce preguntas y recibe respuestas habladas)
- **Corrección automática** con feedback argumentativo
- **Seguimiento de progreso** por áreas temáticas con panel visual



# LIMITACIONES DEL PROYECTO

- **Dependencia de conexión a internet:** el uso de la API de OpenAI requiere conexión constante, lo que limita el funcionamiento offline.
- Corrección automática básica, **no sustituye a un profesor.**
- **Personalización inicial**, no tutor completo.
- Cobertura limitada de contenidos jurídicos: actualmente el sistema abarca **sólo las materias principales** del examen de grado y no incluye todas las áreas del Derecho.
- **Dependencia de APIs externas:** el funcionamiento depende de la disponibilidad y estabilidad de servicios como OpenAI, Azure Speech Service y Azure Database for PostgreSQL.

05

## METODOLOGIA



# SCRUM



## Framework Ágil

- Metodología iterativa e incremental
- 4 sprints de 2 semanas cada uno.
- Entregas incrementales de valor al final de cada sprint

## Ceremonias

Sprint Planning, Daily, Review y Retrospective.

## Distribución por Sprints

- **Sprint 1:** Funcionalidades base → visualizar preguntas, responder por texto, ver materias.
- **Sprint 2:** Feedback y progreso → retroalimentación, ver si es correcta, tablero de progreso, explicación de corrección.
- **Sprint 3:** Funcionalidades avanzadas → responder por voz, detectar debilidades, recomendaciones, escuchar corrección.
- **Sprint 4:** Personalización y cierre → recordatorios, notificaciones, login, puntaje final, selección de materias, perfil.



06

RELEASE



**Preguntas y  
Respuestas**

**Corrección  
y Feedback**

**Ver  
progreso**

**Recordatorios**

**Perfil**

## Release 1: Base Practica (octubre)

Visualizar preguntas

Responder preguntas  
por texto

Escuchar preguntas

Responder preguntas  
por voz

Ver si mi respuesta  
esta correcta

Leer explicación

Ver mi puntaje final

Ver mis debilidades

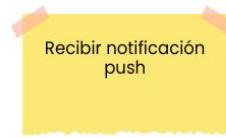
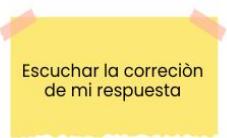
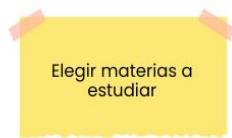
Dashboard de  
progreso

Ver el historial de  
sesiones y tests

Guardar preferencias



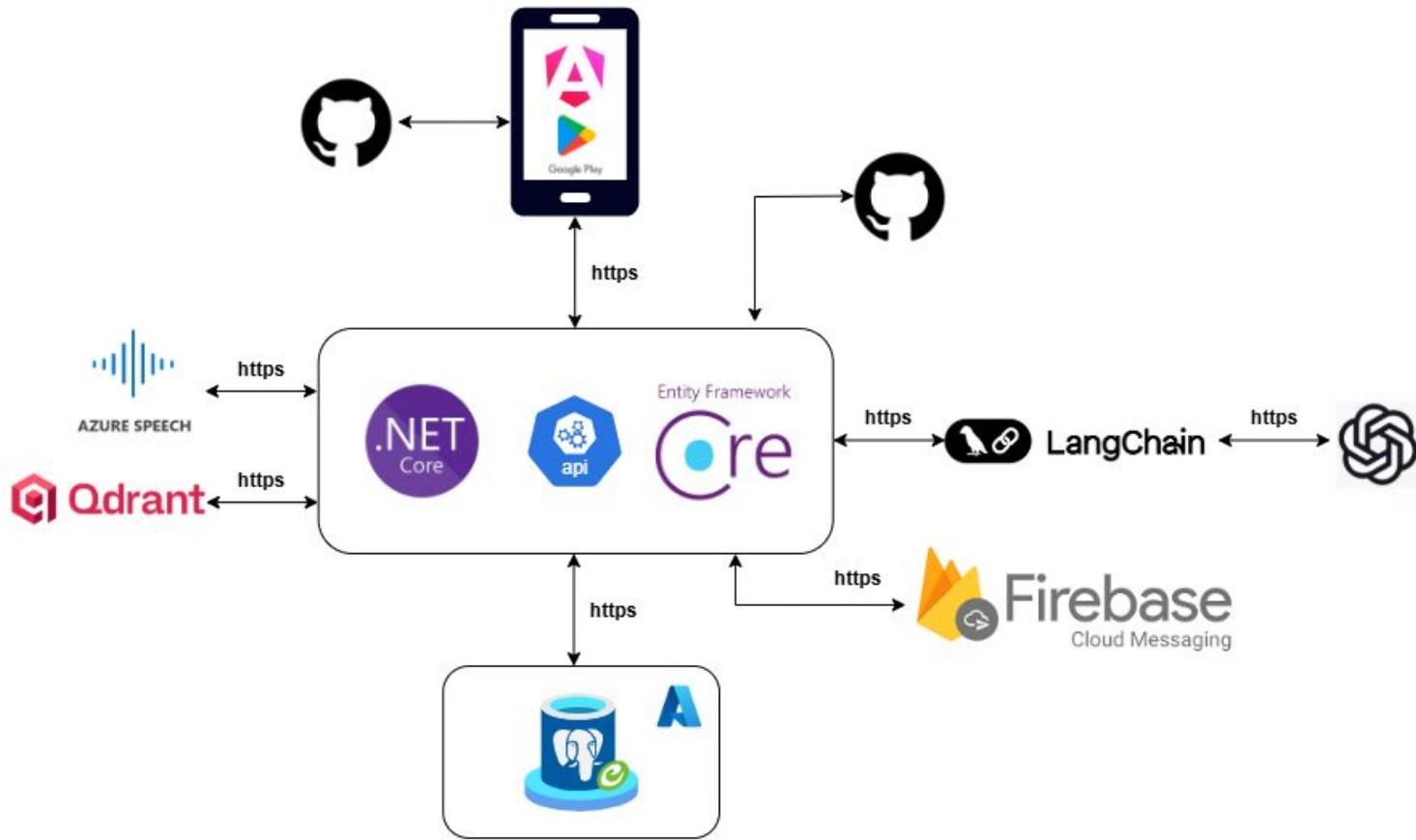
## Release 2: Personalización (Noviembre)

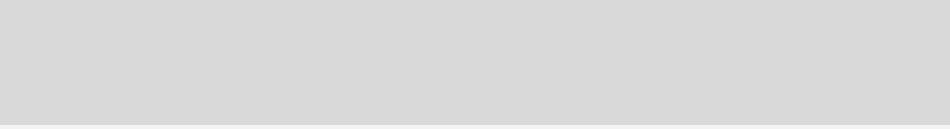


07

# ARQUITECTURA







08

MODELO  
DE DATOS

**12**

Tablas Core

**3**

Triggers Activos

**4**

Vistas Analíticas

**8**

Campos Calculados

## CONTENIDO

- **AREAS**  
Organización por áreas académicas
- **TEMAS**  
Temas organizados por área
- **SUBTEMAS**  
Granularidad con orden jerárquico

### JERARQUÍA

área → tema → subtema

## OPERACIONAL

- **ESTUDIANTES**  
Usuarios con modo adaptativo
- **TESTS**  
Sesiones de práctica realizadas
- **PREGUNTAS\_GENERADAS**  
Preguntas con IA y referencias documentales
- **TEST\_PREGUNTAS**  
Respuestas registradas con métricas
- **PREGUNTA\_OPCIONES**  
Opciones múltiples por pregunta

## ADAPTATIVO

- **ESTUDIANTE\_RENDIMIENTO\_TEMAS**  
Tracking automático por tema con triggers
- **ESTUDIANTE\_RENDIMIENTO\_SUBTEMAS**  
— Análisis granular de debilidades
- **VISTA\_DEBILIDADES\_TEMAS**  
Vista con nivel\_dominio calculado
- **VISTA\_DEBILIDADES\_SUBTEMAS**  
Análisis detallado multinivel

### AUTOMÁTICO

Se actualiza en cada respuesta

# FLUJO DE DATOS PRINCIPAL

Proceso automatizado de análisis y adaptación en tiempo real

1

## ESTUDIANTE RESPONDE PREGUNTA

El usuario completa una pregunta durante su sesión de test, registrando respuesta, tiempo y corrección.

Etiqueta: *tabla: test\_preguntas*.

3

## ACTUALIZACIÓN DE MÉTRICAS

Se calculan: tasa de acierto, racha consecutiva, tiempo promedio y días sin practicar.

Etiqueta: *rendimiento\_temas + rendimiento\_subtemas*

2

## TRIGGER SE ACTIVA AUTOMÁTICAMENTE

El sistema ejecuta *actualizar\_rendimiento\_tema\_completo()* sin intervención manual.

Etiqueta: *trigger AFTER INSERT*

4

## IDENTIFICACIÓN DE DEBILIDADES

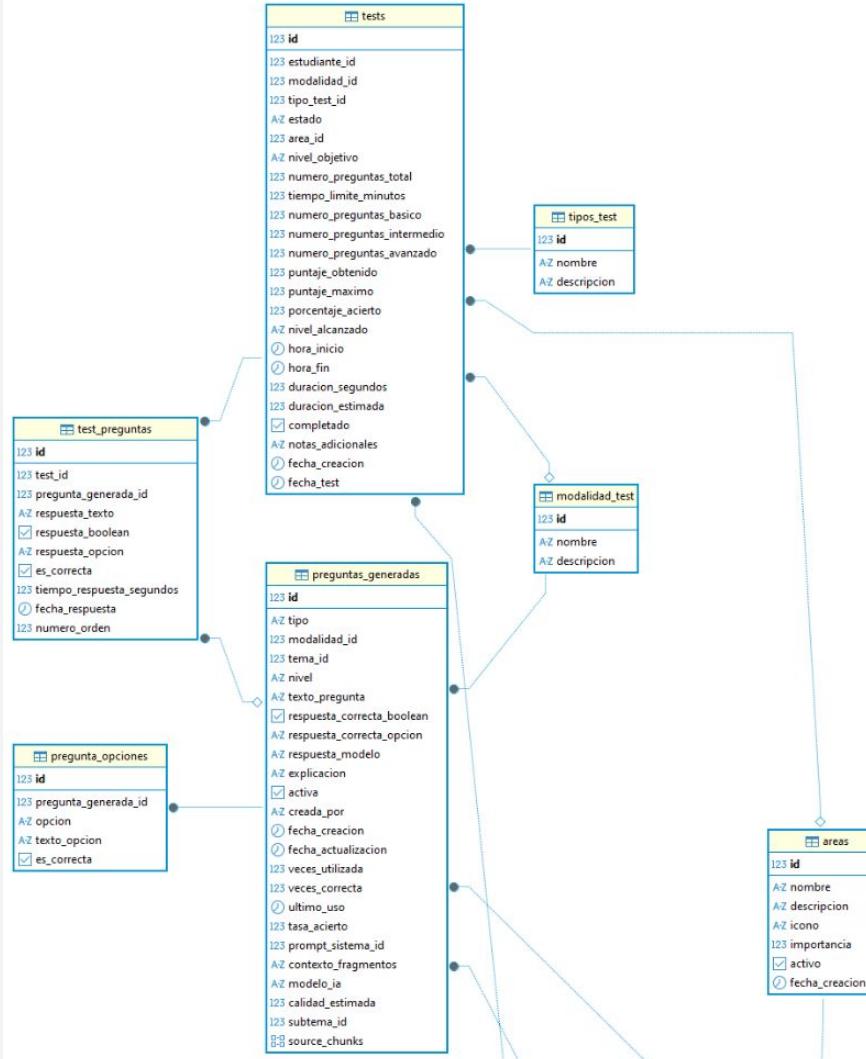
Las vistas calculan *nivel\_dominio* y priorizan áreas de mejora.

Etiqueta: *vista\_debilidades\_temas/subtemas*

## Modelo de datos completo

### Módulo operacional

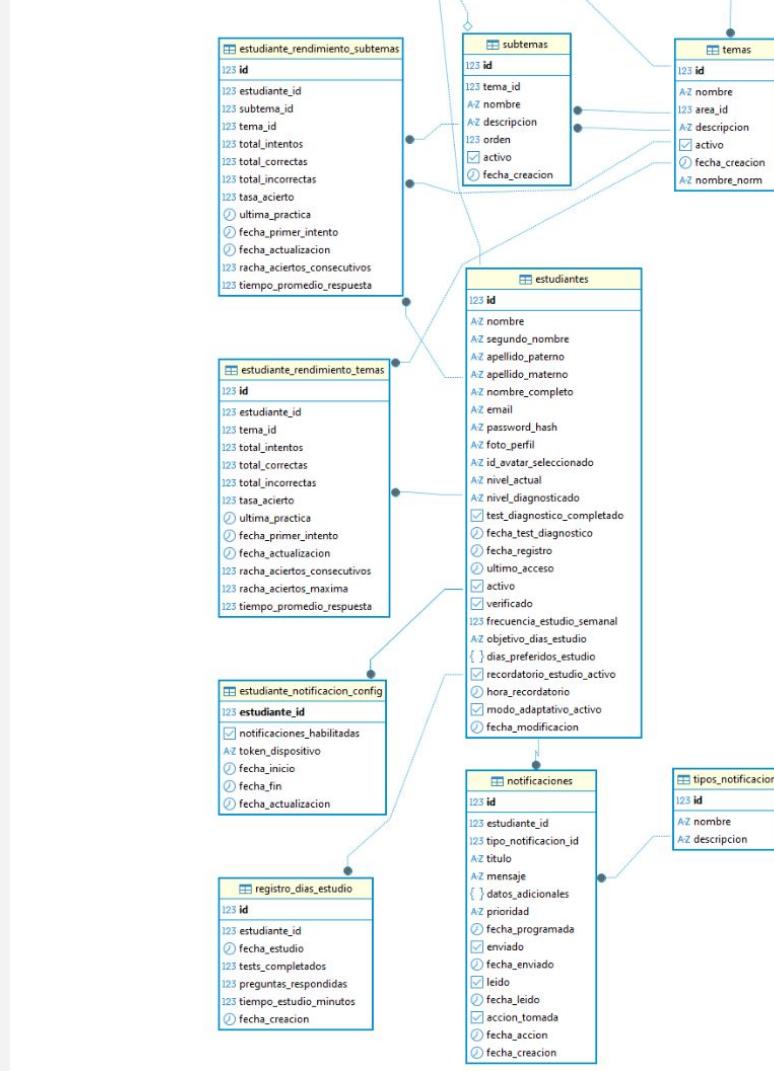
El estudiante responde preguntas en test\_preguntas. Cada respuesta activa triggers que actualizan automáticamente su rendimiento por temas y subtemas, identificando fortalezas y debilidades en tiempo real.



## Modelo de datos completo

### Módulo adaptativo

Triggers actualizan automáticamente el rendimiento por temas y subtemas. Identifica fortalezas, debilidades y patrones de estudio. Sistema de notificaciones personalizadas para mantener la constancia.



09

# TECNOLOGÍAS UTILIZADAS



## FRONTEND



**Angular 20**  
Framework principal



TypeScript  
Lenguaje



**Ionic/Capacitor**  
Desarrollo móvil



**RxJS**  
Programación reactiva

## BACKEND



**.NET Core 9.0**  
Framework backend



**Entity Framework Core**  
ORM



**ASP.NET Core  
Web API**  
API REST



## BASE DE DATOS



PostgreSQL

**PostgreSQL**  
Base de datos  
relacional principal



**Qdrant**  
Base de datos  
vectorial para IA

## LIBRERÍAS Y HERRAMIENTAS



**Langchain**  
**OpenAI**  
Generación de  
preguntas (GPT-4)



**iText7 8.0.2**  
**DocumentFormat.OpenXml 3.0.1**



**NAudio 2.2.1**  
Procesamiento de  
audio

## SERVICIOS NUBE



**Azure  
Database**  
Almacenamiento en  
la nube



**Azure Speech  
Service**  
Text-to-Speech y  
Speech-to-Text



**Azure App  
Service**  
Hosting del backend



**FirebaseAdmin  
3.4.0**  
Servicio de notificaciones push



10

DEMOSTRACIÓN





II

# RESULTADOS OBTENIDOS

## Objetivos Cumplidos

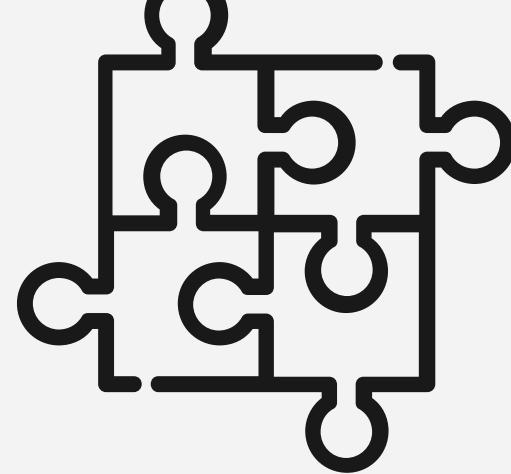
- 85% de las historias de usuario completadas
- 3 sprints ejecutados según planificación
- Release 1 entregado a tiempo

## Aprendizajes Técnicos

- Dominio de Angular y .NET Core
- Experiencia con servicios de Azure
- Integración de IA en aplicaciones móviles
- Gestión de proyectos ágiles

## MVP Completamente Funcional

- Todas las funcionalidades principales implementadas
- Integración exitosa con servicios de IA



## Calidad Técnica

- Arquitectura escalable
- Código documentado y versionado en Git
- Buenas prácticas de desarrollo aplicadas

## Trabajo en Equipo

- Coordinación efectiva bajo metodología Scrum
- Distribución equitativa de responsabilidades
- Comunicación constante y colaborativa

**12**

# OBSTÁCULOS

presentados durante el desarrollo

## OPERACIONAL

- **Integración de APIs de IA**  
Documentación exhaustiva y pruebas iterativas
- **Optimización de costos**  
Implementación de caché y prompts optimizados
- **Reconocimiento de voz**  
Configuración óptima Azure Speech Service

## DESAFIOS ORGANIZACIONALES

- **Coordinación de horarios**  
Daily meetings flexibles y comunicación asíncrona
- **Carga académica**  
Planificación realista en sprints
- **Curva de aprendizaje**  
Pair programming y documentación compartida

## APRENDIZAJES CLAVE

- Comunicación constante como factor crítico de éxito
- Priorización de MVP funcional sobre funcionalidades complejas
- Adaptación de Scrum al contexto académico
- Trabajo colaborativo como principal fortaleza del equipo



**¿PREGUNTAS?**

**GRACIAS**