# Transformers con arquitectura RAG: Generación de exámenes

Uriel Arias, Catalina Müller, Manuel Quesada

### Contenido

01

02

03

#### Introducción

Descripción de la aplicación desarrollada

#### **Datos**

Datos utilizados y su formato

#### **Arquitectura**

Diseño de la arquitectura RAG

04

05

06

#### Resultados

Algunos exámenes generados

#### Métricas

Evaluación del desempeño del modelo

#### **Conclusiones**

Algunas conclusiones del proyecto

## Objetivo de la aplicación

#### Generación de exámenes

- Selección de la materia y tipo de examen
- Usando el formato de exámenes viejos de la materia seleccionada
- Basándose en apuntes específicos de la materia

### **Dataset**

#### **Exámenes viejos**

PDFs de exámenes viejos de las distintas materias

#### **Apuntes**

Contenido teórico de las materias con contenido propio del material visto en la universidad

Formato de Markdown, con los temas separados por headings

## Formato de los exámenes



#### Prompt a un LLM

Se proporcionan PDFs de exámenes viejos para que describa su estructura



#### Archivo .json que describe la estructura

Indica la cantidad de ejercicios promedio, la frecuencia de aparición de temas, el formato típico de los ejercicios



#### Almacenamiento en S3

Se almacena el formato del examen para la materia en un archivo JSON



# Retrieval Augmented Generation

Arquitectura

## Procesamiento de los apuntes





## Markdown headers

De Langchain: MarkdownHeader TextSplitter

#### Subject

Para filtrar al hacer el retrieve según la materia solicitada

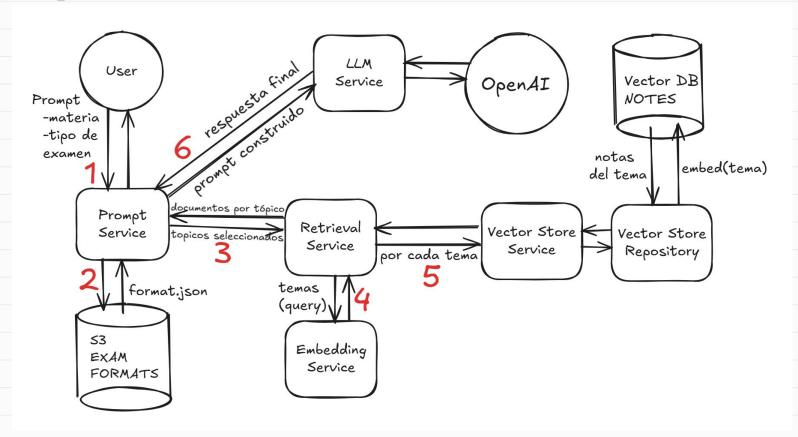
#### **OpenAl**

text-embedding -3-small

#### **Pinecone**

Almacenamiento de los embeddings

## **Arquitectura**



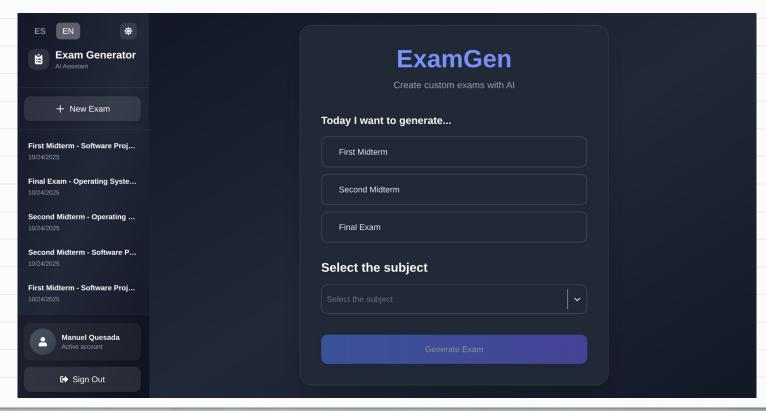


- Cómo transmitir el formato del examen? Oué almacenar en la BD vectorial?

#### Prompt del usuario

- Cómo obtener el nombre de la materia? Distintos nombres para una misma materia? Errores de tipeo? Materias para las que no tenemos data almacenada?
- Calculo de métricas
- Formato del prompt y de las respuestas

## Aplicación: frontend y deployment

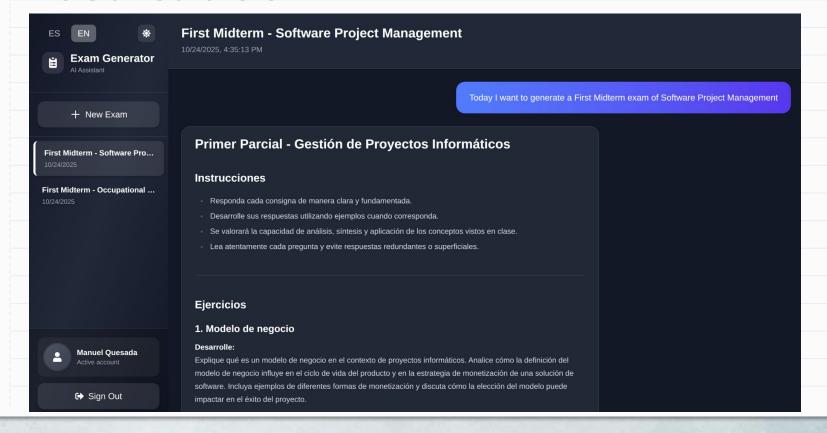




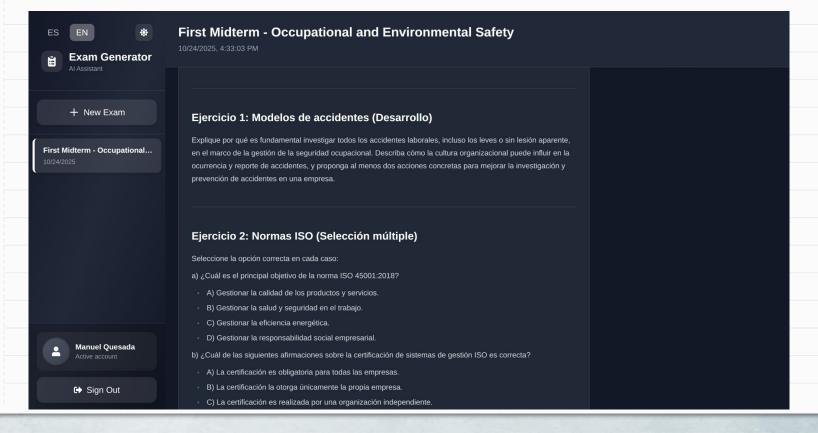
# Resultados

Algunos exámenes generados

## Resultados



## Resultados





# Métricas

Desempeño del modelo

# Métricas Context Precision

#### **Faithfulness**

**Answer Relevance** 

Si los documentos recuperados están relacionados al tema de examen solicitado

Si se relaciona el ejercicio generado con los documentos proporcionados en el contexto Si el ejercicio generado se relaciona con el tema correspondiente

Examen	Context Precision	Faithfulness	Answer Relevance
GPI-midterm1 (small embedding)	0,366	0,343	0,818
GPI-midterm1 (large embedding)	0	0	0,474
GPI-midterm1 (manualmente)	0,697	0,840	0,960
SO-final	0,066	0,020	0,804



# Conclusiones

## Conclusiones

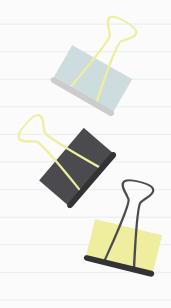
- La arquitectura diseñada permite generar preguntas directamente relacionadas a los temas específicos vistos en la materia
- Un menor tamaño de embedding resultó en mejores resultados (y menor costo)

#### **Algunas limitaciones**

- La arquitectura es dependiente de la obtención y el parseo de apuntes de las materias que se quieran incluir
- Se pierden detalles al armar y describir el formato del examen

#### Posibles expansiones

- Mayor cantidad de materias
- Mejorar la precisión al transmitir el formato del examen
  - o Proveer un examen completo a modo de exemplo/referencia
  - o Incluir preguntas de desarrollo de exámenes en la base de datos vectorial?



# Gracias!

**CREDITS:** This presentation template was created by <u>Slidesgo</u>, and includes icons by <u>Flaticon</u>, and infographics & images by <u>Freepik</u>