Universidad de Costa

Rica Informática

Empresarial

Proyecto Final

Curso:	**Sistemas Expertos**
Docente:	**Rolando de Jesús Herrera Sánchez**
Valor porcentual del proyecto:	**12.5 %**
Fecha de asignación:	**22/05/2025**
Fecha de entrega:	**26/06/2025**

Objetivo General del Proyecto

Desarrollar un **asistente conversacional web** basado en tecnologías de Deep Learning que pueda responder preguntas sobre un dominio temático específico elegido por el estudiante, **Por ejemplo:** Historia, medicina, videojuegos, astronomía, etc.

Descripción del Proyecto

Este asistente utilizará modelos de lenguaje avanzados para comprender las consultas del usuario, recuperar información relevante del corpus de conocimiento del dominio y generar respuestas coherentes y precisas, ofreciendo una experiencia de interacción fluida y natural. Además, que el sistema (asistente) debe de integrar como **mínimo** lo siguiente:

- Un **modelo de lenguaje grande (LLM)** como backend generativo.
- Un mecanismo de **recuperación aumentada (RAG)** usando embeddings y búsqueda vectorial.
- Un **frontend web interactivo** que simule una experiencia de chat.

UNIVERSIDAD DE COSTA RICA

Requisitos Técnicos

- Implementación de **al menos una** técnica de Deep Learning vista en clase. Por ejemplo → Embeddings, transformers, codificadores etc.
- Mostrar cómo se construye el pipeline RAG y cómo este se integra al modelo desarrollado.
- Incluir un conjunto temático **personalizado** de documentos o preguntas de referencia para el modelo.
- Mostrar en la defensa cómo el sistema mejora la respuesta del modelo gracias al RAG.

Guía para desarrollo de un chatbot con RAG

1. Definir tema y dataset

1. Elegir un tema específico.
2. Reunir contenido de referencia tales como PDF´s, libros, Wikipedia etc.
3. Convertir el contenido a texto plano con ayuda de PDF to Text o algún script en Python.
4. Guardar todo el material convertido en archivos .txt o .md.

2. Generar los Embeddings necesarios y guardarlos en una base vectorial

- **Opción A:** Usar Vercel AI SDK + Embeddings de OpenAI.
- **Opción B:** Embeddings en Python + ChromaDB

3. Crear el Backend API para procesamiento RAG

1. Configurar la API en Next.js o Express.
2. Lógica de búsqueda vectorial incluyendo los embeddings generados anteriormente + búsqueda por similitud.

4. Construir el Frontend del chat

1. Hacer uso de React o Next.js para el desarrollo de este.

5. Despliegue del proyecto

- **Opción A:** Usar **Vercel** sí se decantaron por Next.js y solamente deben de hacer git push en GitHub y lo conectan a Vercel.
- **Opción B:** Presentarlo mediante **Render o Railway** sí se desarrollo con un backend en Express + un frontend en React.
- **Opción C:** Alguna otra investigada por cuenta propia.

Videos de referencias para desarrollo del proyecto

- **Video #1** → https://www.youtube.com/watch?v=E1-mUfpeRu0
- **Video #2** → https://www.youtube.com/watch?v=E4l91XKQSgw

Cuadro de evaluación para el proyecto

Entregables	**Valor porcentual**
Código fuente en repositorio Git (con instrucciones de despliegue).	**3%**
Aplicación desplegada de manera correcta.	**2%**
Informe técnico (máx. 5 páginas) - Arquitectura general. - Herramientas utilizadas. - Cómo se aplicaron los conceptos de Deep Learning.	**4%**
Presentación en vivo del asistente en funcionamiento.	**3.5%**
Total	**12.5%**