# Development of an End-to-End Web Application Leveraging Retrieval Augmented Generation (RAG) and OpenAI's API with Enterprise Data

**Course**: Generative AI

**University**: PSL University

## Project Description

This project aims to develop a robust web application that integrates the principles of Retrieval Augmented Generation (RAG) and OpenAI's API. The application will utilize enterprise data to demonstrate the potential of Generative AI in enhancing productivity and task automatization, especially in customer interaction and customer service.

## Datasets

Simulated enterprise data will be provided, which includes:

* Company’s FAQ
* Twitter’s customer support data: tweets from customers and replies from customer service
* ~~Customer’s emails and customer service agent’s responses~~
* ~~Customer service call transcripts~~
* ~~Company Knowledge base PDF document~~

## Objectives

1. Product Brainstorming:

* Analysis and understanding of the available data.
* Identifying opportunities where Generative AI can provide value.
* Exploration of potential applications using RAG and Generative AI on those data.
* Choose an application to implement.

2. Data Retrieval and Formatting for RAG:

* Extracting, cleaning, and organizing the provided data.
* Formatting the data to be compatible with a RAG system.

3. Development and Coding:

* Architectural design of the web application.
* Building a functional back-end components of the application using Flask in Python, and integrating Azure OpenAI's API and a RAG system.
* Store data in a vector database, such as ChromaDB.
* Building a user-friendly front-end using HTML, CSS and Javascript.
* Version control with Git.

4. Web Application Deployment:

* Deploying the application on an appropriate web platform.

## Expected Outcomes

* A fully operational web application that demonstrates the practical application of Retrieval Augmented Generation and generative AI using enterprise data.
* Practical experience in developing and deploying a GenAI-integrated web application.

Project Timeline

The project’s Github need to be sent before 15/03/2024.

## Resources

1. [Guide from OpenAI to build a Q&A Chatbot using OpenAI API (back-end)](https://platform.openai.com/docs/tutorials/web-qa-embeddings)
2. Guide to use the Microsoft Azure OpenAI API
   1. [GPT model](https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/chatgpt?tabs=python&pivots=programming-language-chat-completions)
   2. [Embedding model](https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/embeddings?tabs=python)