Day 1: Building Generative AI Applications with Vertex AI Challenge Labs

Overview

This set of case studies is to test your understanding of what was covered in class today. They have limited instructions, and you are expected to use the course materials and your experience to complete them. This sandbox project environment will allow you to work on the case studies without the need for your own project. If you prefer, you can use a Google project that you own, so you do not lose your work when this Qwiklab project ends.

Warning: This Qwiklabs environment will run for day 1 of this training class. Then it will be shut down. Make sure you save your work outside of the environment. You can either save your code to an external Git repository, or download the code to your computer. The case studies on days 2 and 3 build on the work you do today, so make sure you don't lose your work!

Day 1 Case Studies Objectives

You will learn how to:

- Write generative AI prompts using advanced prompt engineering techniques
- Prototype GenAI applications using Jupyter notebooks
- Integrate Google GenAI APIs into you own projects

Warning: Resources that you create in this Qwiklabs project are not free and are limited. Please, do not use resources that are not required to complete the case studies.

Case Study 1: Prompt Engineering - Social Media Posts

Your boss asked everyone to write articles about the latest technology trends and post them on social media. They want to create some buzz around the company and increase its social media footprint. You want to be a team player, but don't really have the time.

Write a prompt that allows you to specify a topic, and write a compelling article in a consistent style that will attract readers and get good reviews. Ask Bard for some tips on writing effective technology articles and social media posts.

Do this in Vertex AI Studio and save the prompt for later use.
Experiment with context, examples, temperature, and advanced prompting techniques like chain of thought to get the best results possible.

Case Study 2: Prompt Engineering - Sales Proposals

You work for a technology consulting company that specializes in enterprise software development and cloud migrations. The sales team spends a huge amount of time generating sales proposals. They have asked you if generative AI could help accelerate the process.

A typical proposal includes the following sections:

- Title
- Date
- Customer
- Salesperson
- Executive Summary
- Problem Statement
- Proposed Solution
- Benefits
- Pricing and Terms
- Conclusion

Your task is to create a prompt that allows salespeople to pass in a minimal amount of data and have the model fill in the details. They want the output to be returned in HTML format, so it can be copy and pasted into a document or email.

Log into your Google Cloud sandbox project, and work on the prompt in Vertex AI Studio. Save it for later use. Eventually, this will be added into a program. Someone suggested the variable data could be passed to the model in JSON format.

Experiment with context, examples, temperature, and advanced prompting techniques like chain of thought to get the best results possible.

Case Study 3: Programming the Sales Proposals Example

You've been working with the sales team and they decided they want a wizard-style approach for building the sales proposals. Sections of the proposal are created one step at a time, with the opportunity for the salesperson to make changes. Each section should have its own prompt and examples to optimize the output. The output of prior steps is used to help generate subsequent sections.

Create the Jupyter Notebook and code this Sales Proposals example in it.

Remember, the generated sales proposal should be returned in HTML so it can be used in another application.

Use Vertex AI Workbench to create the Jupyter Notebook in your sandbox project.

Case Study 4: Programming the Social Media Posts Example

Create a web site that allows you to enter a topic and have it generate an article for you as described earlier. Add the following features:

- Allow the temperature property to be set in the UI.
- Have the model return 2 examples, so you can pick the one you like best.
- Add a refresh button to allow the results to be regenerated.
- Allow the generated posts to be editable.

Work on this using the Cloud Shell Editor.

Bonus: Write the code to automatically post the article to your favorite Social Media site.

Reminder: This project will be shut down after class. Make sure to save your work from today either to an external Git repo, or download it to your computer.