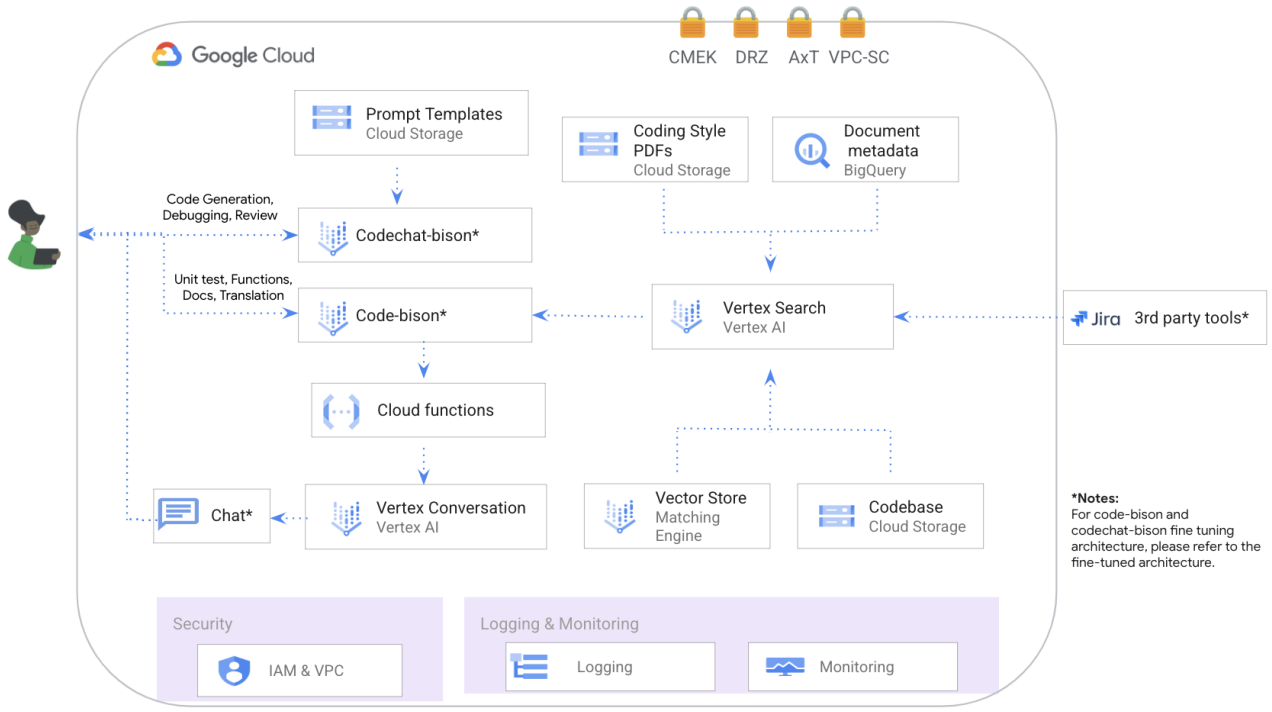
Developer Productivity with Codey E2E Demo Guide

Lei Pan 11/27



Overview

The purpose of this guide is to show you for partners and builders, how to set up resources and how to use the codey API integrations to complete all different tasks in software development life cycle in this notebook (github link). This is part of this developer productivity with GenAI.

At the end of the guide, there is a collection of demos, sample codes, and architecture to show you how to use Codey API to fulfill individual use journeys.

Setup

Step 1: Enable Vertex AI Search API and Vertex AI Conversation API

1. Go to the Google Cloud Platform (GCP) Console.
2. In the navigation pane, select APIs & Services > Enable APIs and Services.
3. In the search bar, type Vertex AI Search API and click on the result.
4. Click Enable.
5. Repeat steps 3 and 4 for Vertex AI Conversation API.

Step 2: Set Up Prompt Template in GCS Bucket

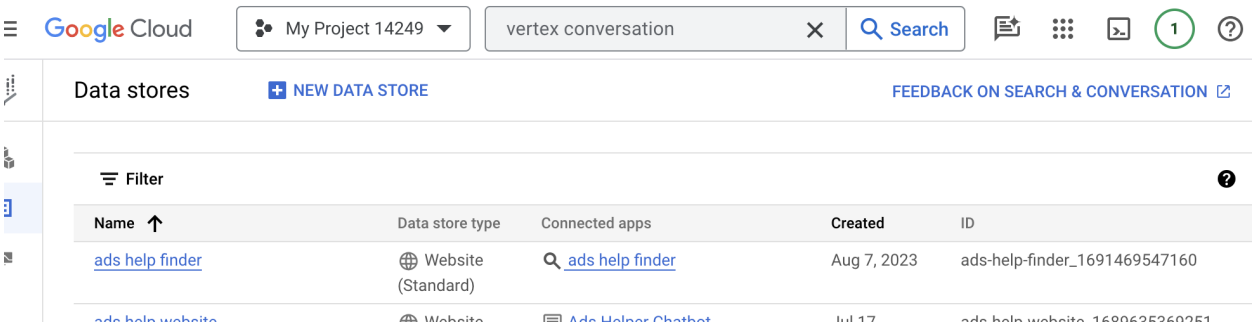
- Public doc: How to create a GCS bucket
- CSV files they can refer to: Github Link
- Store those prompt csv in your bucket to use them in the code or use your own prompts

Step 3: If you don't have a fine-tuned codey api, follow the doc to create it

- Public doc on fine tuning codey: how to fine tune codey

Step 4: Build Vertex AI Search Engine with PDFs (Coding Style PDFs)

- Public doc: How to set up unstructured data store in vertex ai search
- For PDFs: use your own pdfs including your coding best practices or use those examples and save them as PDFs and dump them to vertex AI search engine datastore.
- Once you set the datastore up, you will find the search engine id in the UI. Use that in the code.



Step 5: Build Another Vertex AI Search Engine with JIRA Issue Websites

- Public doc: How to set up JIRA in Vertex AI search
- For JIRA links: you can use your own JIRA links or use this public JIRA links to do the demo
- Once you set the datastore up, you will find the search engine id in the UI. Use that in the code. Refer to the screenshot in the step above.

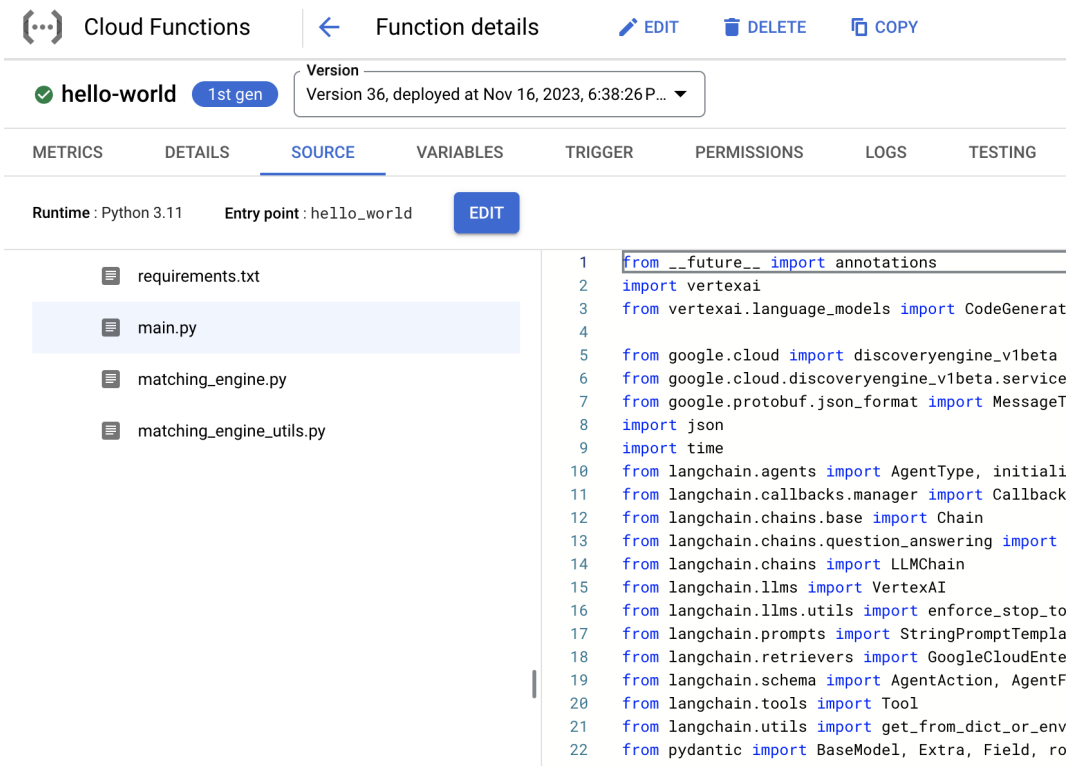
Step 6: Break Down Code Repository (Bank of Anthos) to Chunks and Store Indexes in the Vector Store (Matching Engine)

- Ingest the codebase to GCS bucket. Replace the `GCS_BUCKET_DOCS`with the bucket you use.
- Code reference to do RAG on codebase: github link

Step 7 (Optional - only if you want to demo Google Chat): Deploy Cloud Functions Code which Uses MultiRetrievalQAChain to Retrieve Information (Embedding Spaces + RAG + Codey) from 3 Different Retriever Embedding Spaces

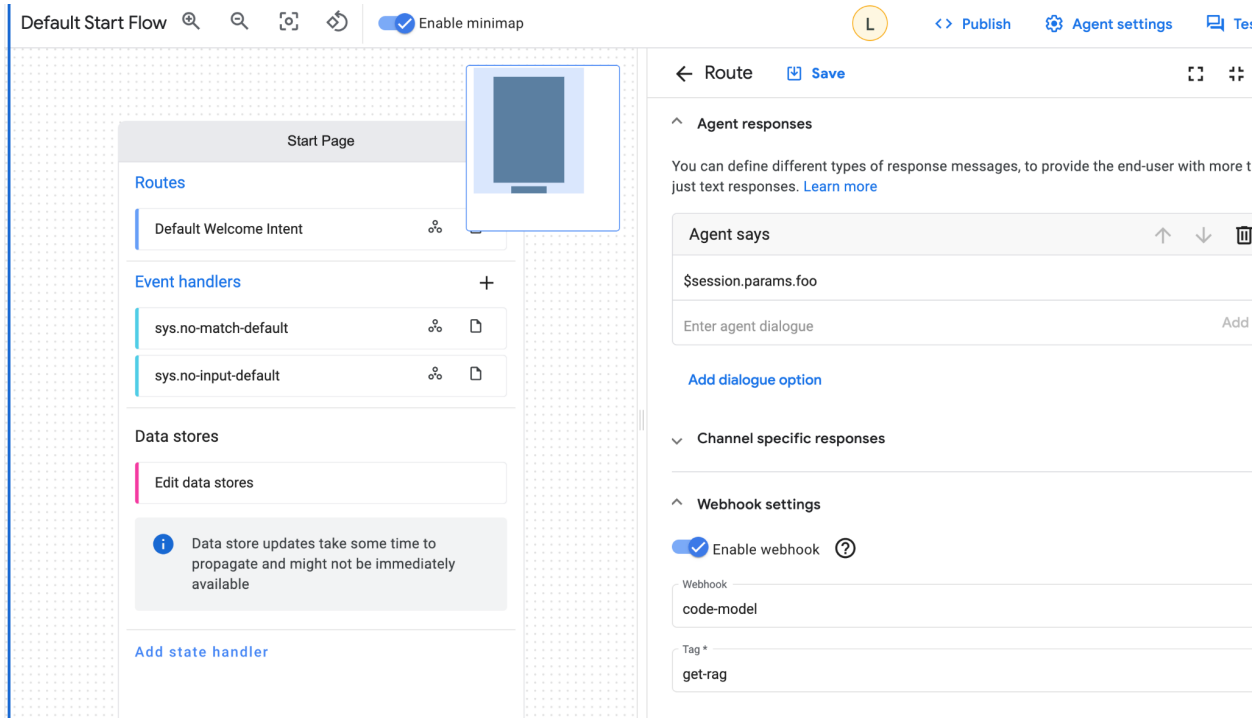
- Public doc: How to deploy cloud function
- Deploy below 3 sources to cloud functions as shown in the screenshot below
    - Webhook cloud function code. Copy the code here to cloud functions as main.py

- Requirements.text: below code is all you need.
  - Flask==2.2.2
  - Werkzeug==2.3.7
  - google-cloud-aiplatform
  - google-cloud-discoveryengine
  - langchain==0.0.236
- You can [download matching_engine.py and mathcing_engine_utils.py here](#).

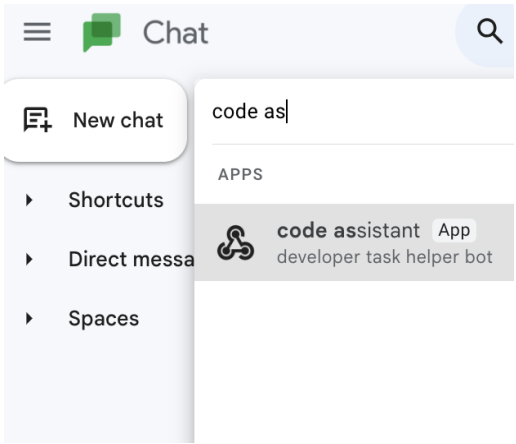


Step 8 (Optional - only if you want to demo Google Chat): Call Cloud Function in Webhook in a Dialogflow Project

- Public doc: [How to set up a dialogflow project](#)
- Public doc: [How to set up webhook with cloud function](#)
- Once you set it up, go to default welcome intent, sys.no-match-default, [sys.no-input-default](#) and set the agent response to the response from webhook (cloud function that you deployed), please refer to the screenshot below.



- Public doc: [how to deploy dialogflow project to Google Chat]
- After that, you should be able to search the chatbot in your google chat



Demo Steps

Step 1: Code Generation

Follow the zero-shot prompts in the notebook to do code generation, unit test, explanation, refactoring, and comment generations.

Step 2: Code Debugging

Follow the prompts from the prompt template to do debugging.

Step 3: Code Migration

Follow the prompts from the prompt template to do code migration from COBOL to JAVA. One partner had good experience when using Codey API to convert COBOL to JAVA.

Step 4: Codebase Search & Doc Search

Follow the prompts to test Different Retrievers in the notebook and Google chat if you set it up. If you don't know what RAG and Retrievers are, here is the paper.