

Universidad del Caribe

2000

CANCUN, QUINTANA ROO, MÉXICO

CONOCIMIENTO Y CULTURA PARA EL DESARROLLO HUMANO

Cómputo en la Nube para Dispositivos Ubicuos: Deploy your FastAPI app in serverless Google App Engine

Autores:

Jose David Almendra Pacheco

Valeria Del Carmen Guzman

Brisa Jael Lima Arenas

Frank Joseph Lopez Cruz

Freda Maria Perez Novelo

Ulises Gilberto Castellanos Flores

Docente:

Candy E. Sansores

Asignatura:

Cómputo ubicuo

23 de abril del 2024

Introducción

FastAPI es un moderno y rápido framework de web para construir APIs con Python 3.7+, basado en anotaciones de tipo estándar de Python1. Ofrece un rendimiento elevado y es fácil de usar, lo que permite escribir código eficiente y rápido. Por otro lado, Google App Engine (GAE) es una plataforma como servicio (PaaS) completamente administrada que permite a los desarrolladores construir y desplegar aplicaciones web escalables sin preocuparse por la infraestructura subyacente.

Desarrollo

Paso 1: Crear una aplicación FastAPI

Creemos un archivo llamado main.py, el cual contendrá un único punto de conexión raíz para devolver una respuesta JSON

```
main.py x
main.py > index
1
2 from fastapi import FastAPI
3 from fastapi.responses import JSONResponse
4 app = FastAPI()
5 @app.get("/")
6 def index():
7     return JSONResponse(
8         content={
9             "message": "Hello from Google App Engine!"
10        }
11    )
```

correremos nuestra aplicación con un servidor gunicorn con el siguiente comando:

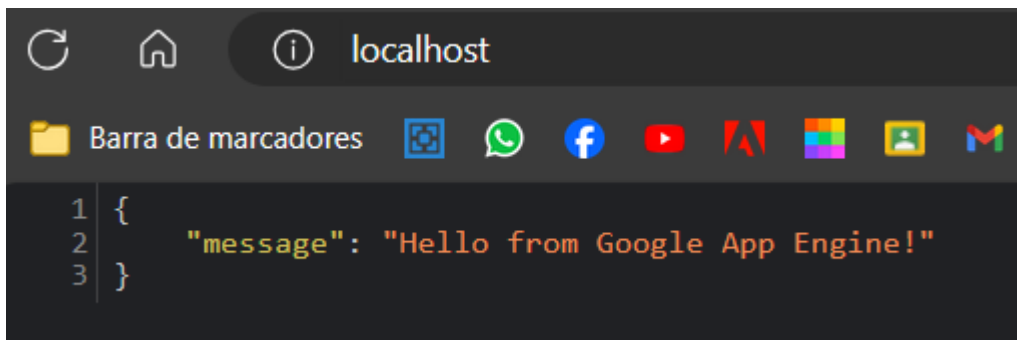
gunicorn -w 1 -k uvicorn.workers.UvicornWorker main:app

Si no también podemos usar este comando:

uvicorn main:app --host 0.0.0.0 --port 80

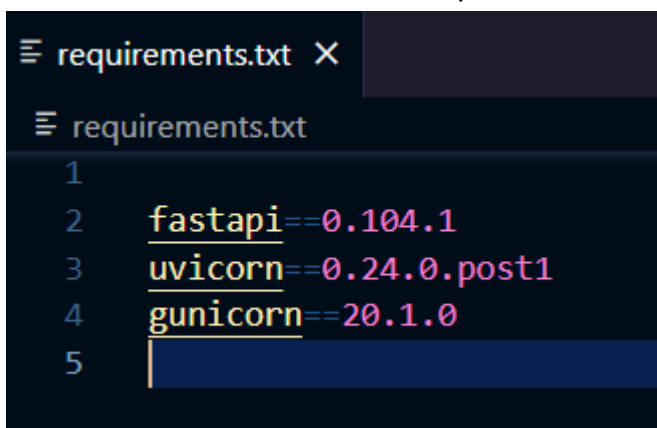
```
PS D:\Documentos\ubicuo\nube> uvicorn main:app --host 0.0.0.0 --port 80
INFO: Started server process [12280]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:80 (Press CTRL+C to quit)
INFO: 127.0.0.1:57172 - "GET /int HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:57172 - "GET /int HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:57229 - "GET / HTTP/1.1" 200 OK
```

Al correr nuestra aplicación podremos acceder a ella a través de localhost



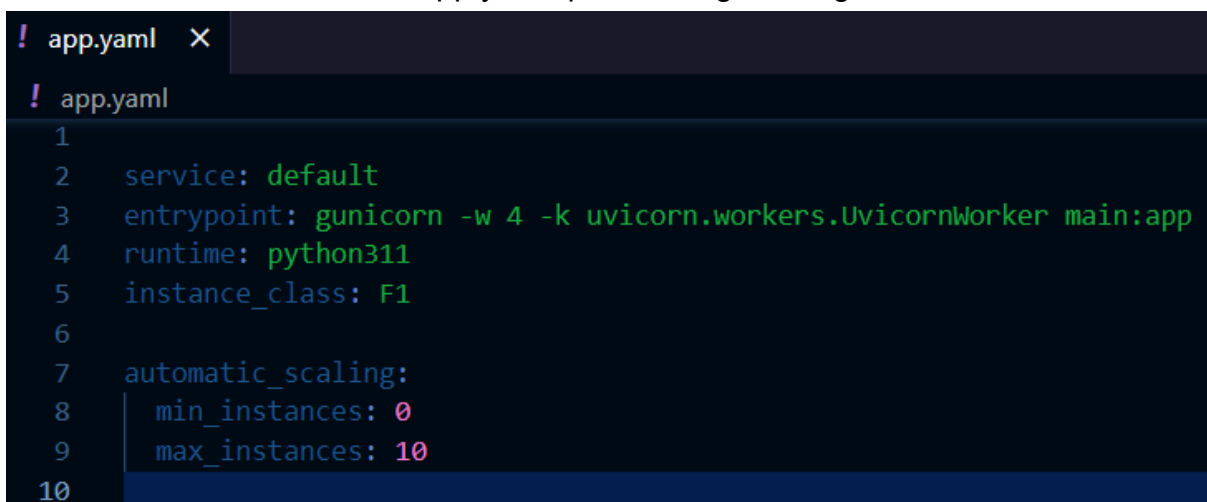
Paso 2: Crear dependencias para el tiempo de ejecución de Python

Creamos un archivo llamado requirements.txt que contenga lo siguiente:



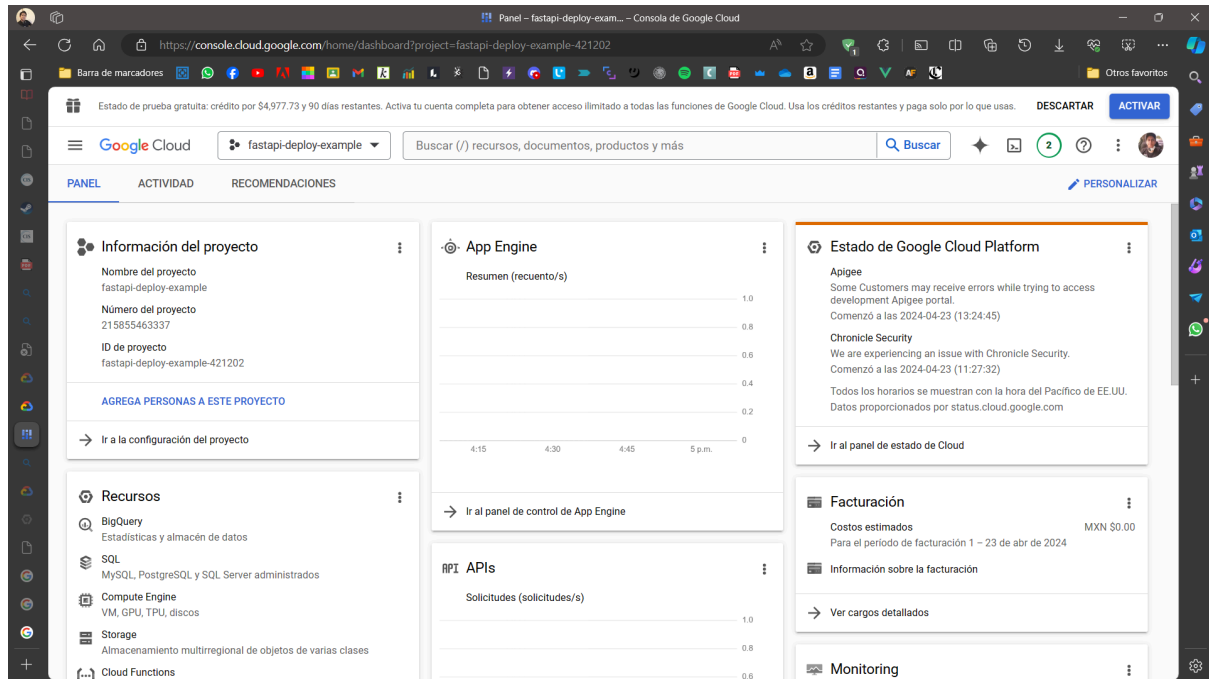
Paso 3: Crear el archivo de configuración YAML de Google App Engine

Creamos un archivo llamado app.yaml que contenga lo siguiente



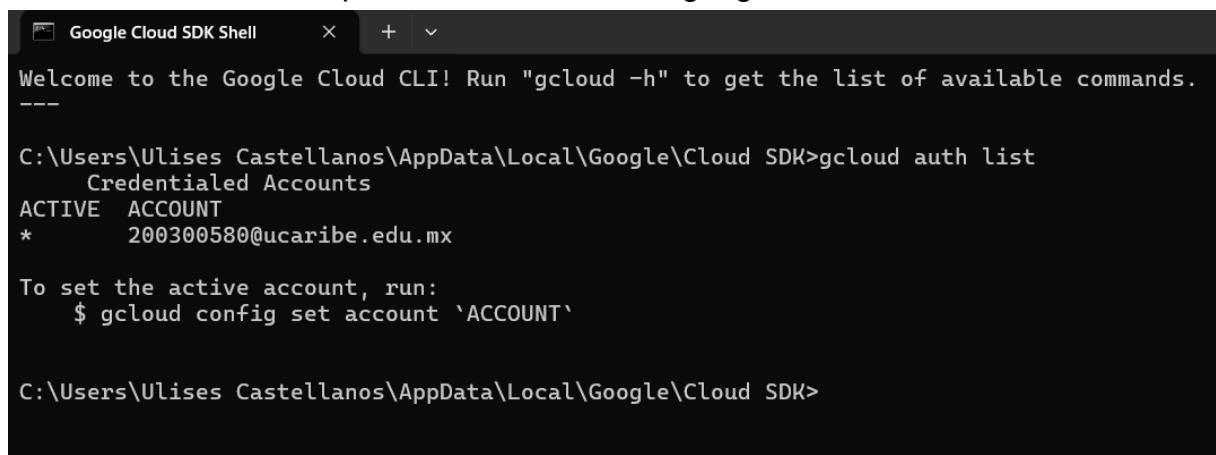
Paso 4: Crear un proyecto de Google Cloud

Creamos una cuenta en google cloud, creamos un proyecto el cual nombramos fastapi-deploy-example y dentro de este proyecto configuramos la facturación.



Paso 5: Instala la CLI de Google Cloud

Instalamos CLI desde la documentación oficial de google cloud, una vez instalado accedimos a la cuenta que estamos usando en google cloud



Paso 7: Implementar la aplicación FastAPI en Google App Engine

Finalmente implementamos la aplicación en google app engine usando el siguiente comando:

gcloud app deploy --project=fastapi-deploy-example-421202 --version=v1

Un punto importante a tomar es que la cuenta que se está utilizando debe de tener permisos de Implementador de App Engine

Al correr la aplicación nos mostrará lo siguiente, en el tendremos que responder con y

```

PS D:\Documentos\ubicuo\nube> gcloud app deploy --project=fastapi-deploy-example-421202 --version=v1
Services to deploy:

descriptor:          [D:\Documentos\ubicuo\nube\app.yaml]
source:              [D:\Documentos\ubicuo\nube]
target project:      [fastapi-deploy-example-421202]
target service:      [default]
target version:      [v1]
target url:          [https://fastapi-deploy-example-421202.uc.r.appspot.com]
target service account: [fastapi-deploy-example-421202@appspot.gserviceaccount.com]

Do you want to continue (Y/n)? y
s.
#=====#
#= Uploading 3 files to Google Cloud Storage      =#
#=====#
File upload done.
Updating service [default]...done.
Setting traffic split for service [default]...done.
Deployed service [default] to [https://fastapi-deploy-example-421202.uc.r.appspot.com]

You can stream logs from the command line by running:
$ gcloud app logs tail -s default

To view your application in the web browser run:
$ gcloud app browse

```

Una vez la aplicación este corriendo podremos usar el siguiente comando par abrir el servicio en el navegador:

gcloud app browse

mostrándonos lo siguiente:

