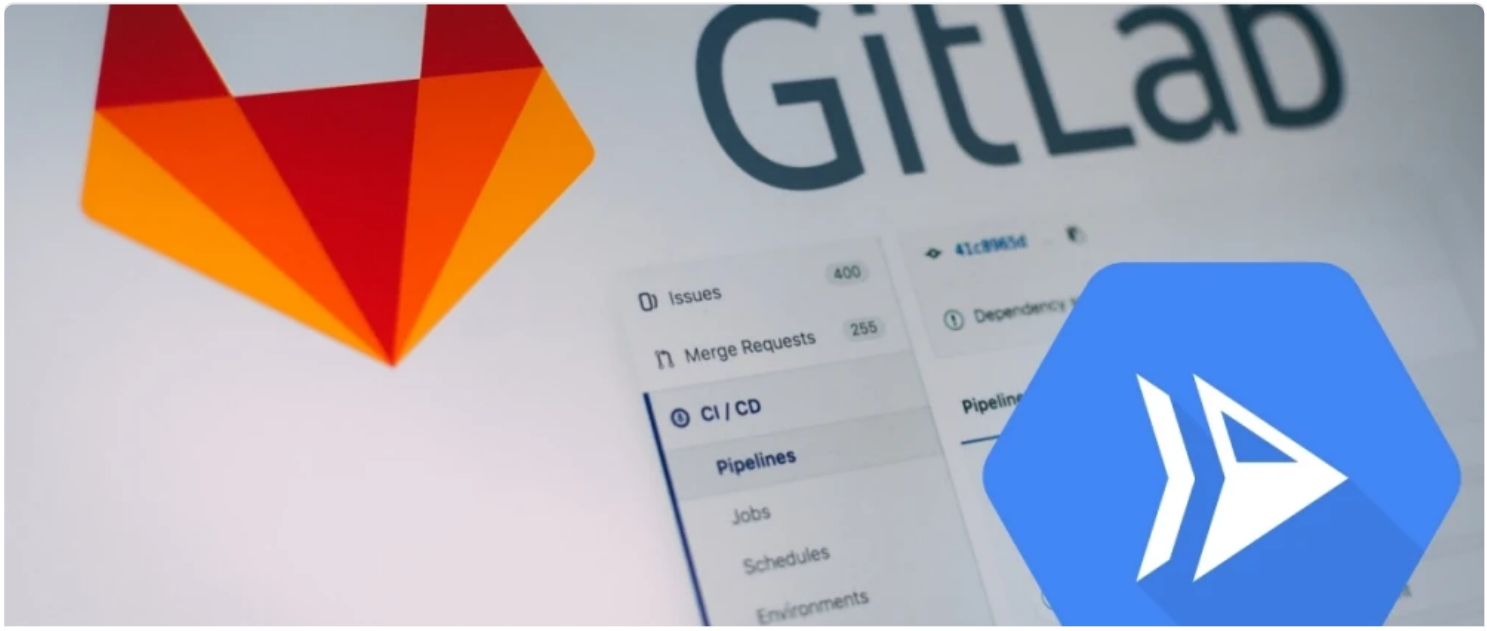


Set up automated deployments with Google Cloud Run and Gitlab

akinniyi May 7 · *Updated on May 9* · 4 min read

Let's look at how we can set up a continuous delivery pipeline for our Google Cloud Run projects with Gitlab CI/CD

Prerequisites
Google Cloud Run:

> Cloud Run is a serverless, managed compute platform that enables you to run stateless containers that are invocable via web requests or Pub/Sub events.
> To run a Cloud Run service, you need to:
>
> - Have a Google Project
> - Enable Cloud Run API
> - Enable Cloud Build API

Gitlab:

> GitLab is a web-based DevOps lifecycle tool that provides a Git-repository manager providing wiki, issue-tracking, and continuous integration and deployment pipeline

> features, using an open-source license, developed by GitLab Inc.
> You can create a Gitlab project here: New Gitlab project

Clone sample repo here:
https://gitlab.com/niyi/myhelloworldapp

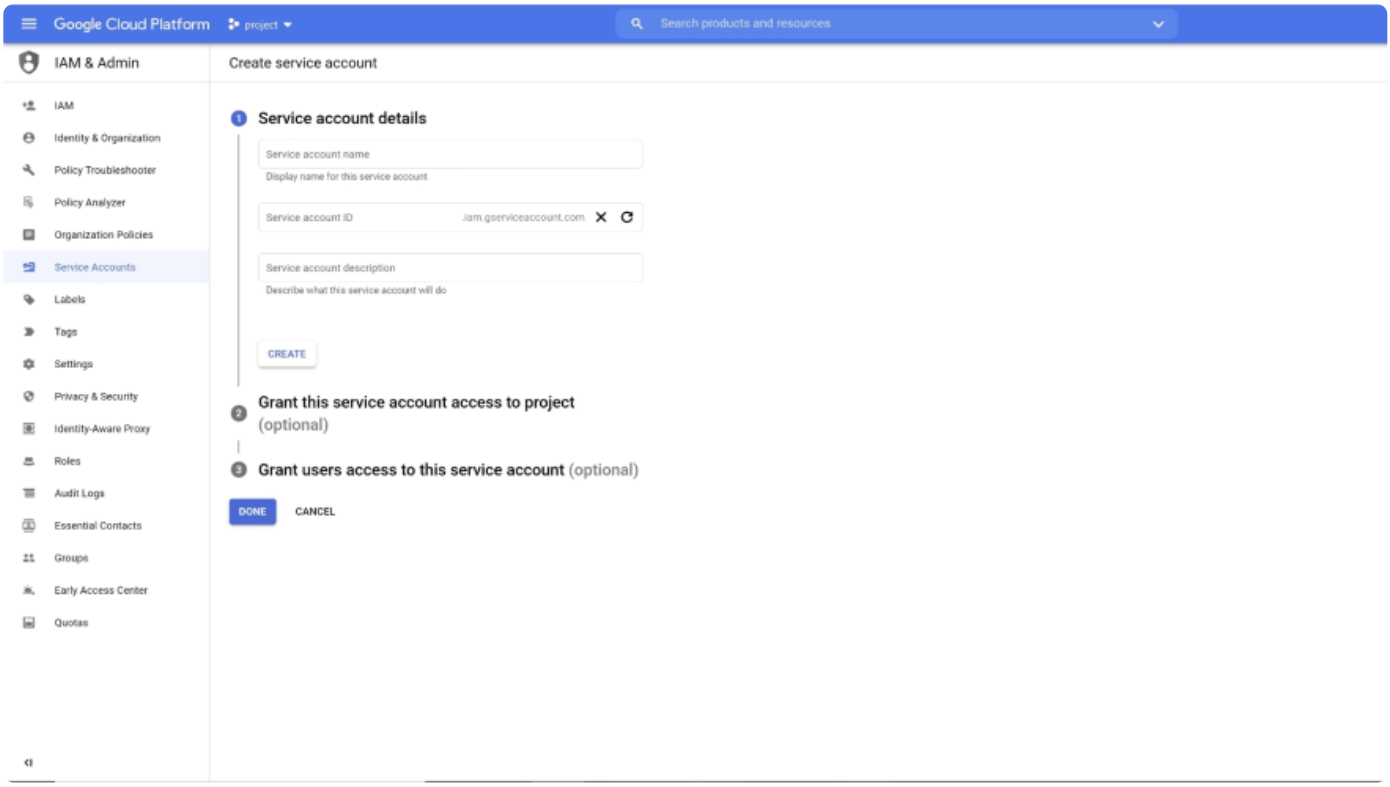
Step 1a: Enable Service Account

A service account is a special kind of account used by an application to make authorized API calls on the GCP platform.
On your Google Cloud project, navigate through Cloud Build > Settings.
Under Service account permissions, make sure both Cloud Run and Service Accounts are enabled

Step 1b: Create a Google Service Account

We'll create a new service account for your application to use

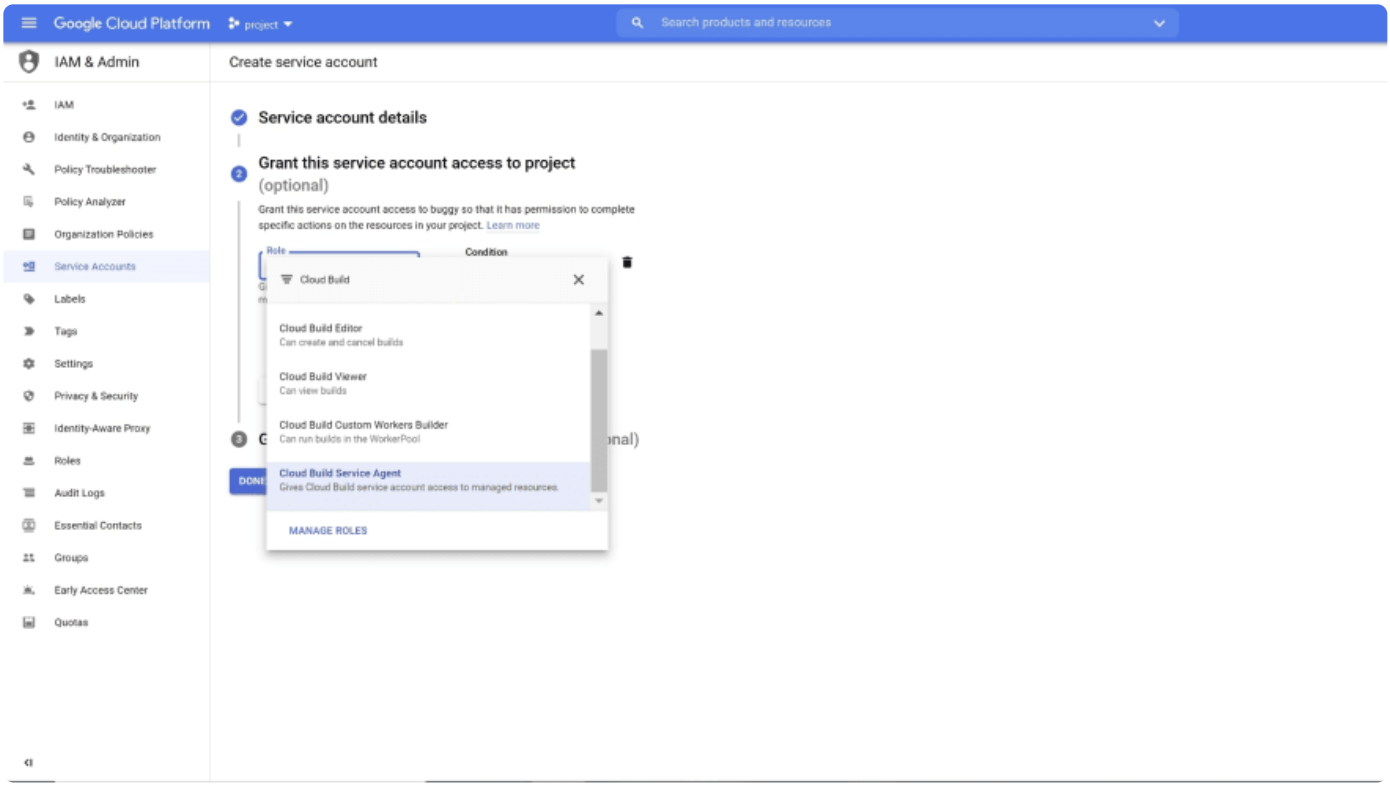
- On your Google Cloud project, navigate through IAM & Admin > Service Accounts > Click on CREATE SERVICE ACCOUNT



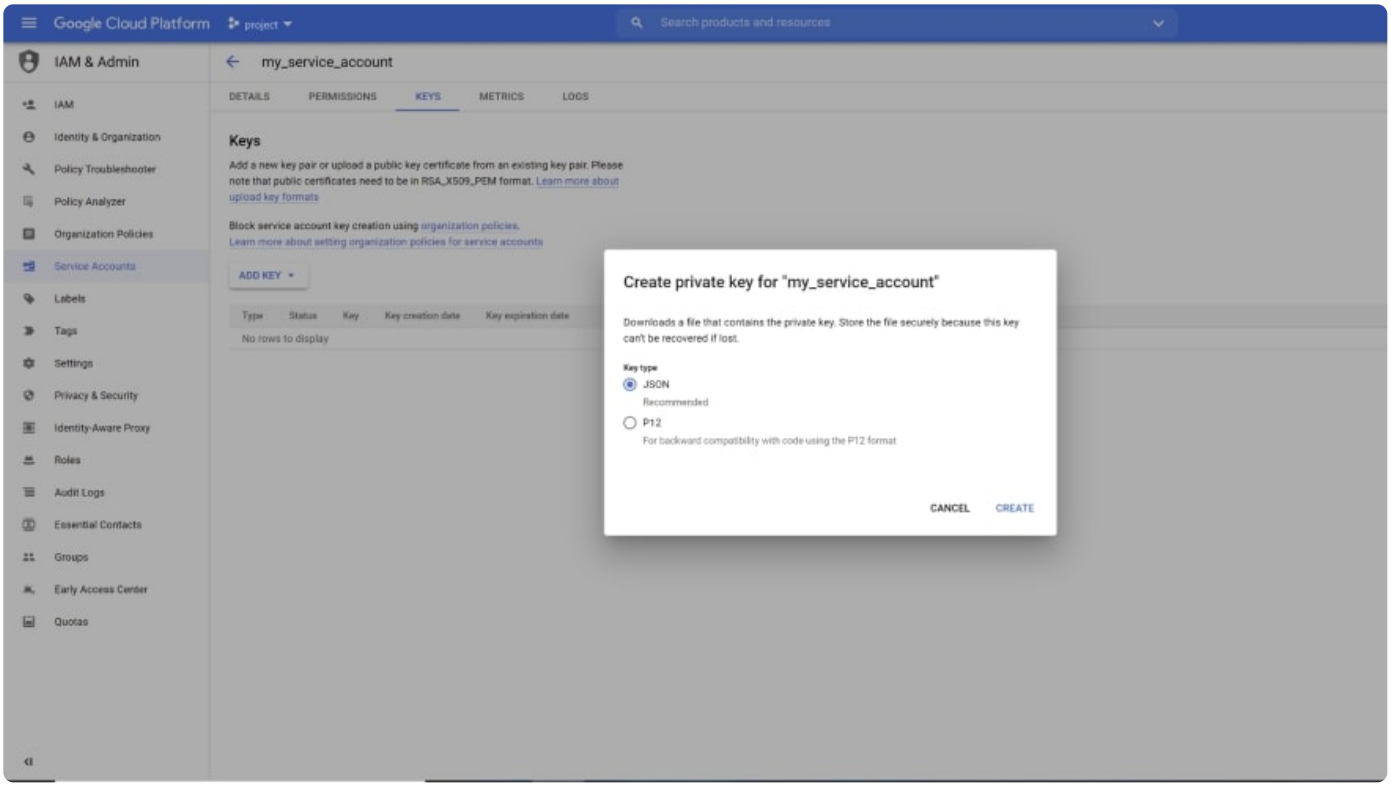
- Give your new service account any name you want and click CREATE
- Add the following roles to your service account by clicking Select Role input under task number 2
    1. Cloud Build Service Agent

2. Service Account User
3. Cloud Run Admin
4. Project Viewer

- Click Create then click Done to add the account.



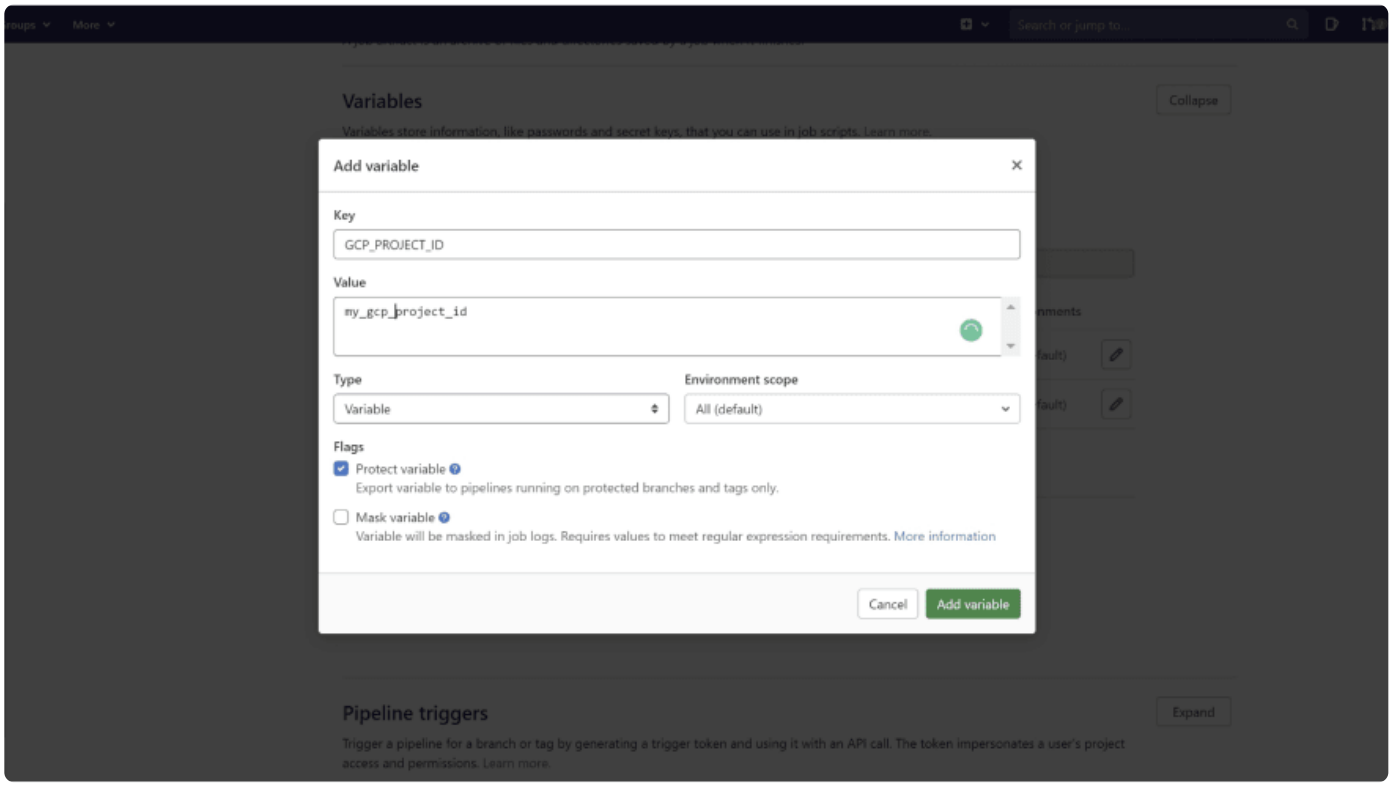
- Generate a credential file for this account by navigating to the newly created service account > Keys > Click on Add Key > Create New Key. Select JSON and click CREATE

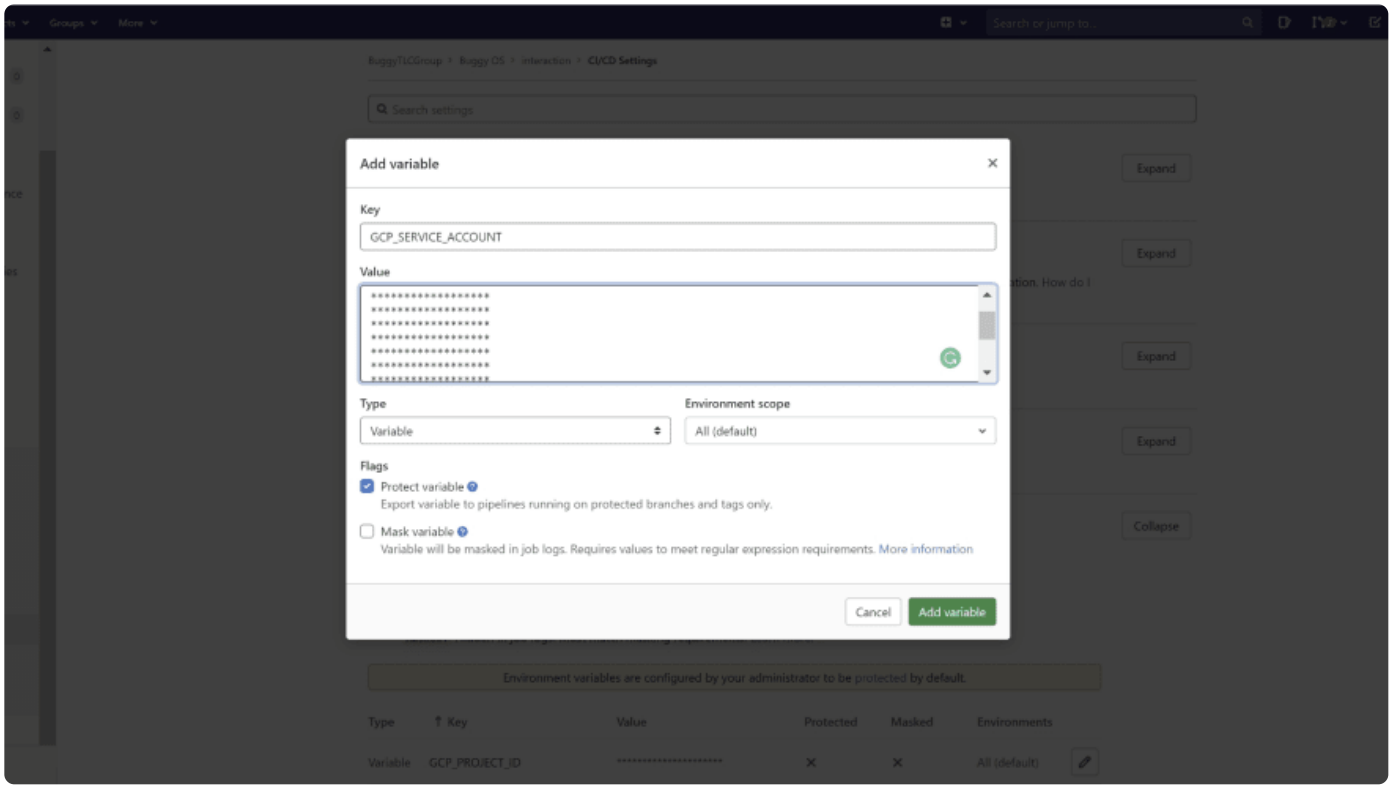


Step 2: Setup Gitlab CICD variables

In this step, we'll create variables that we'll use in our code. One for the GCP Project ID and another for the Service Account we created earlier

- Navigate to the project repository on Gitlab > Settings > CI/CD
- To add a variable, under the Variables section, click the Expand button and click on Add Variable We need to add two variables, one names GCP_PROJECT_ID with the value of our GCP Project ID and the other named GCP_SERVICE_ACCOUNT for the content of the JSON we downloaded earlier





Step 3: Setup Application code

We need to configure our code to connect to Gitlab CI/CD. We'll also use Docker to containerize our application so it runs the same across multiple platforms.

- We've added a Dockerfile in our application that will run on PORT 8080, which is Google Cloud Run's default port

- We've also added a .gitlab-ci.yml file which is the file the triggers our CI/CD pipeline on Gitlab *

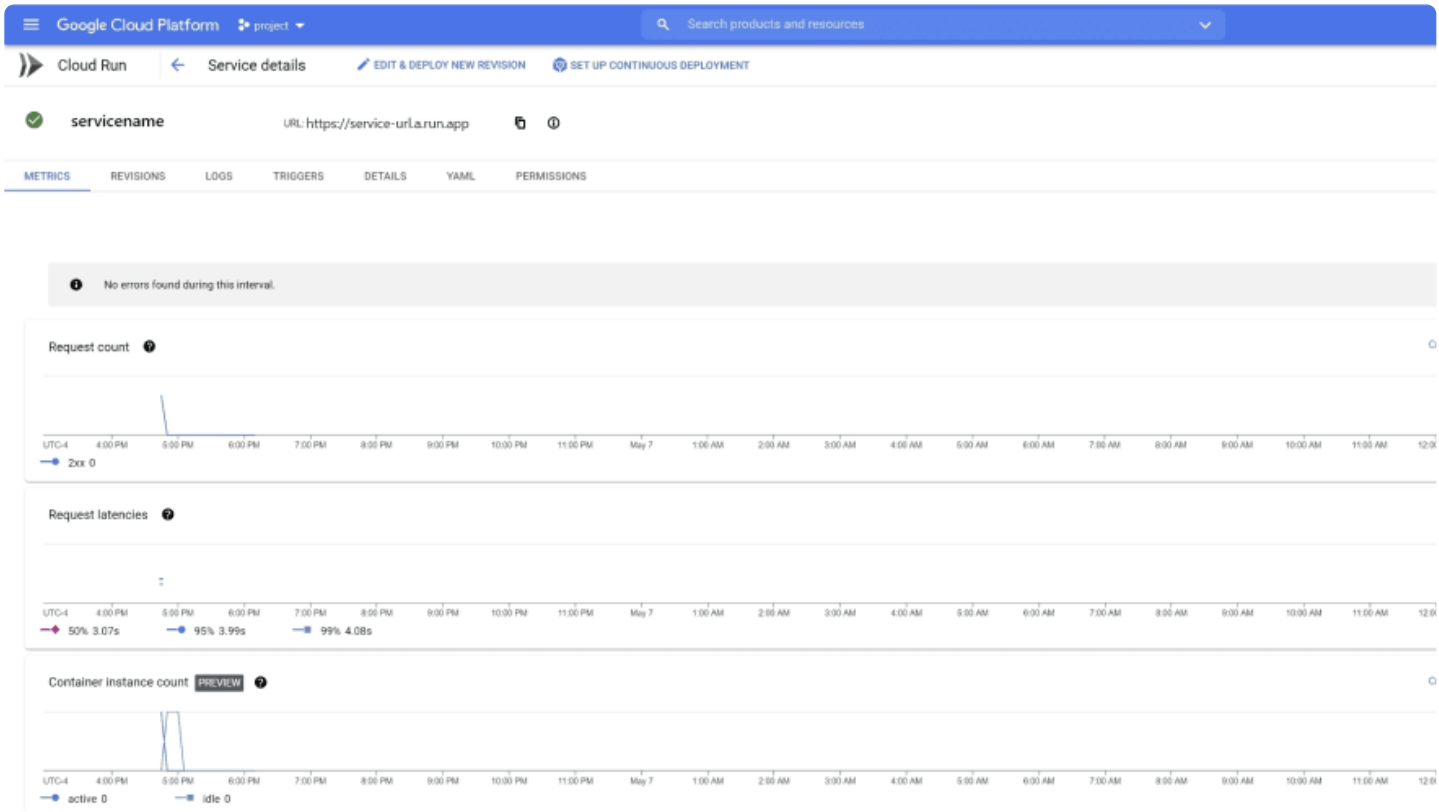
```
# File: .gitlab-ci.yml
variables:
  SERVICE_NAME: "myHelloWorldApp"

deploy:
  stage: deploy
  only:
    - master # This pipeline stage will run on this branch alone

  image: google/cloud-sdk:latest # We'll use Google Cloud SDK for Cloud Run relate
  script:
    - echo $GCP_SERVICE_ACCOUNT > gcloud-service-key.json # Save Google cloud cont
    - gcloud auth activate-service-account --key-file gcloud-service-key.json # Ac
    - gcloud auth configure-docker # Configure docker environment
    - gcloud config set project $GCP_PROJECT_ID #Set the GCP Project ID to the var
    - gcloud builds submit --tag gcr.io/$GCP_PROJECT_ID/$SERVICE_NAME #Run the gcl
    - gcloud run deploy $SERVICE_NAME --image gcr.io/$GCP_PROJECT_ID/$SERVICE_NAME
```

Replace the SERVICE_NAME value with the desired name for your application and save the changes.

At the end of the file, we're running the commands gcloud build and gcloud run deploy to build and deploy our application respectively.

Push your changes to the remote Gitlab repository and watch as your new baby is created.

To monitor the progress of your deployment on Gitlab navigate to CI/CD > Pipelines and click on the latest job.
To see your new application on Cloud Run, navigate to GCP > Cloud Run and search for the name of the service



Congratulations!

Discussion (0)

Code of Conduct • Report abuse

akinniyi

I'm a Learner

LOCATION
NYC

WORK
Full Stack DEv at Not Google

JOINED
Dec 21, 2018

Trending on DEV Community 🔥

6 Useful Tips from Visitors To Improve your Portfolio □□

#showdev #career #productivity #portfolio

Learn these awesome Javascript concepts.

#javascript #webdev #tutorial #career

I bought a tiny piece of internet history. [Plus a little puzzle to keep you busy this weekend!]

#watercooler #webdev #discuss