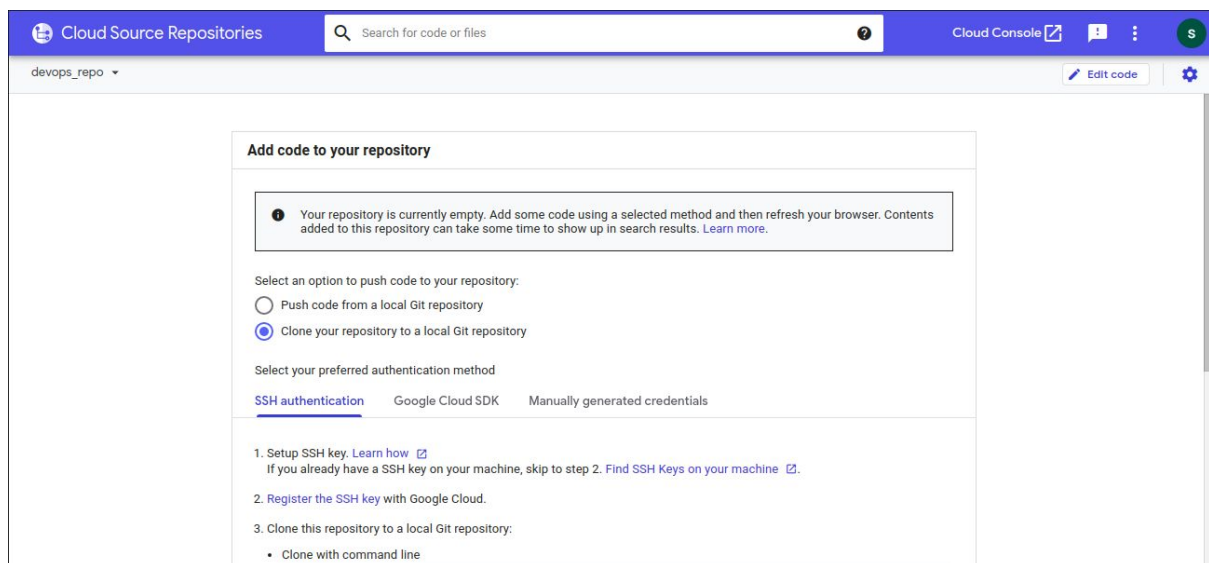


Building a Devops pipeline

Lab Objectives

- Create a Git repository
- Create a simple python application
- Test the web application in cloud shell
- Define a Docker build
- Manage Docker images with Cloud Build and Container registry
- Automate Builds with triggers
- Test our build changes

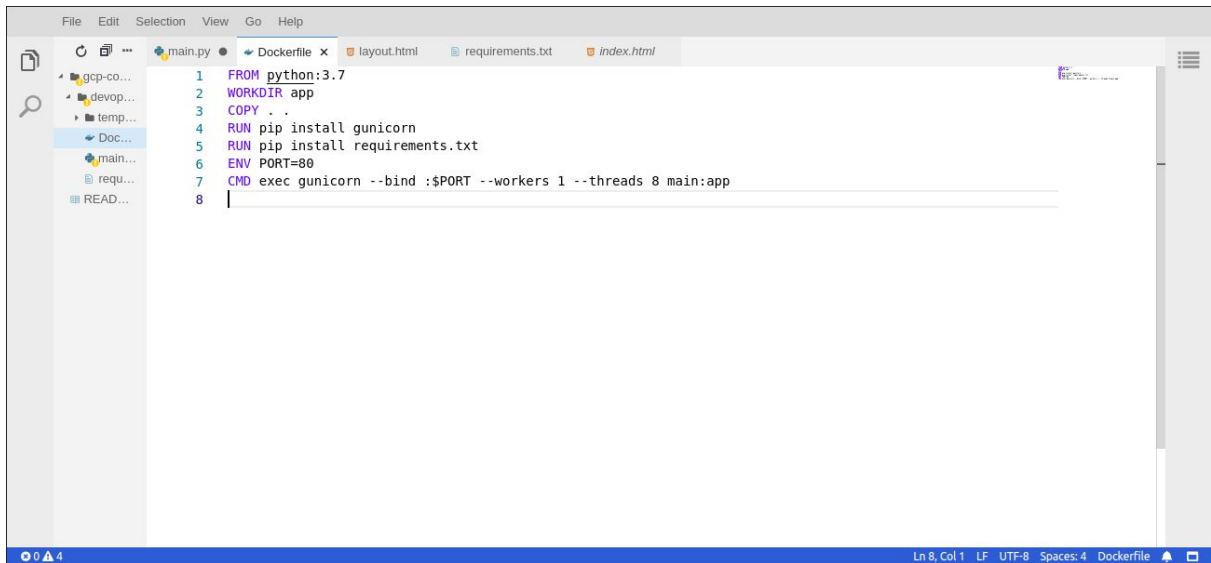
We started the lab by creating a Git repository that we cloned in our project. This repository was created in **Cloud Sources repositories**. Like Github, Gitlab or Bitbucket, the Cloud Sources Repositories is a git repositories server designed and owned by Google. It is integrated to GCP. Our repository was named *devops-repos* like shown on the screenshot below:



[Our repository]

We then created a simple python application. At this step, we just copied and pasted existing code in files that we saved and pushed in our repository. We then tested our web application in the cloud shell.

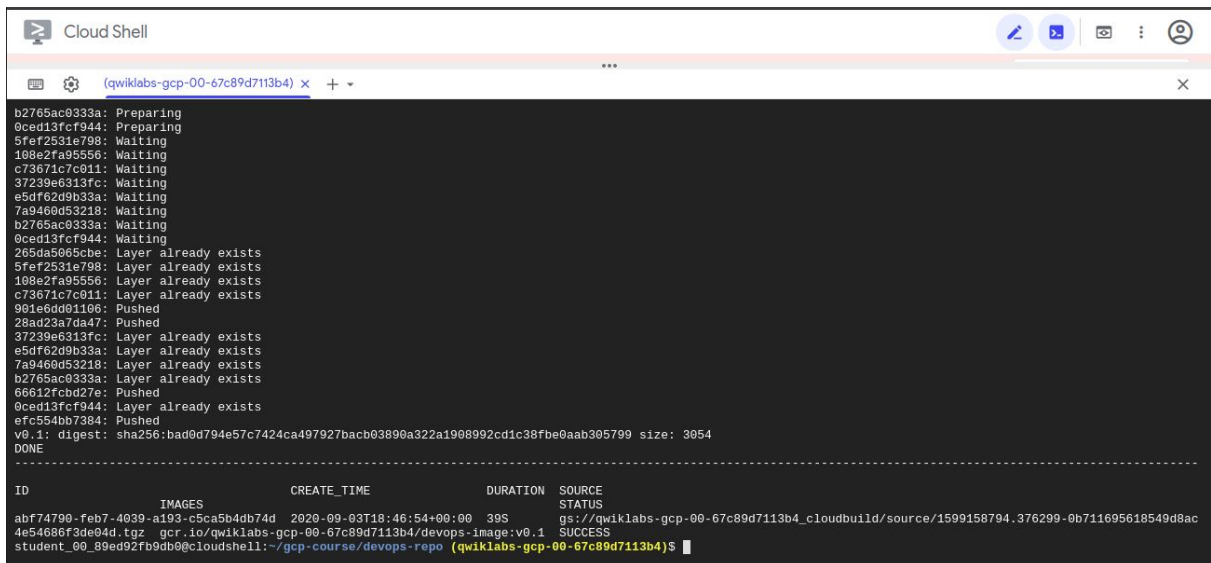
After the test, we made a little modification in the web application source code and pushed the code again. We then created a Dockerfile to containerize our web application.



```
1 FROM python:3.7
2 WORKDIR app
3 COPY . .
4 RUN pip install gunicorn
5 RUN pip install requirements.txt
6 ENV PORT=80
7 CMD exec gunicorn --bind :$PORT --workers 1 --threads 8 main:app
8 |
```

[Dockerfile content]

After this, we used Cloud Build to build our container image.



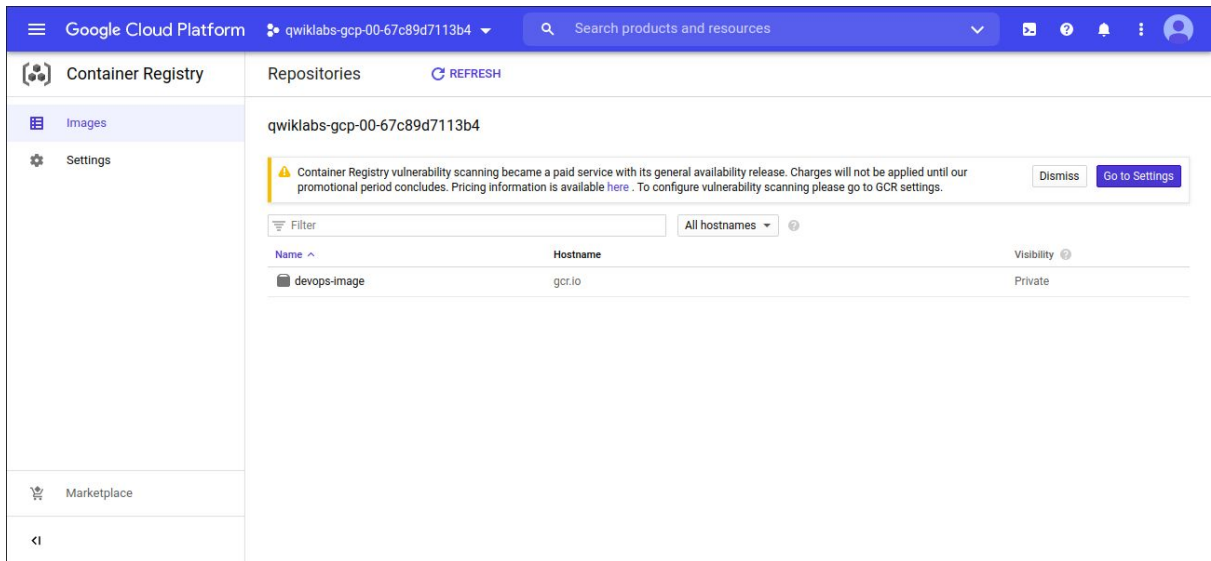
```
b2765ac0333a: Preparing
0ced13fcf944: Preparing
5fef2531e798: Waiting
108e2fa95556: Waiting
c73671c7c011: Waiting
37239e6313fc: Waiting
e5df62d9b33a: Waiting
7a9460d53218: Waiting
b2765ac0333a: Waiting
0ced13fcf944: Waiting
265da5065cbe: Layer already exists
5fef2531e798: Layer already exists
108e2fa95556: Layer already exists
c73671c7c011: Layer already exists
961e6dd01106: Pushed
28ad23a7da47: Pushed
37239e6313fc: Layer already exists
e5df62d9b33a: Layer already exists
7a9460d53218: Layer already exists
b2765ac0333a: Layer already exists
66612fcbd27e: Pushed
0ced13fcf944: Layer already exists
erc554bb7384: Pushed
v0.1: digest: sha256:bad0d794e57c7424ca497927bacb03890a322a1908992cd1c38f6e0aab305799 size: 3054
DONE
```

ID	IMAGES	CREATE_TIME	DURATION	SOURCE	STATUS
abf74790-feb7-4039-a193-c5ca5b4db74d	gcr.io/qwiklabs-gcp-00-67c89d7113b4/devops-image:v0.1	2020-09-03T18:46:54+00:00	39S	gs://qwiklabs-gcp-00-67c89d7113b4_cloudbuild/source/1599158794.376299-0b711695618549d8ac	SUCCESS

```
student_00_89ed92fb9db0@cloudshell:~/gcp-course/devops-repo (qwiklabs-gcp-00-67c89d7113b4)$
```

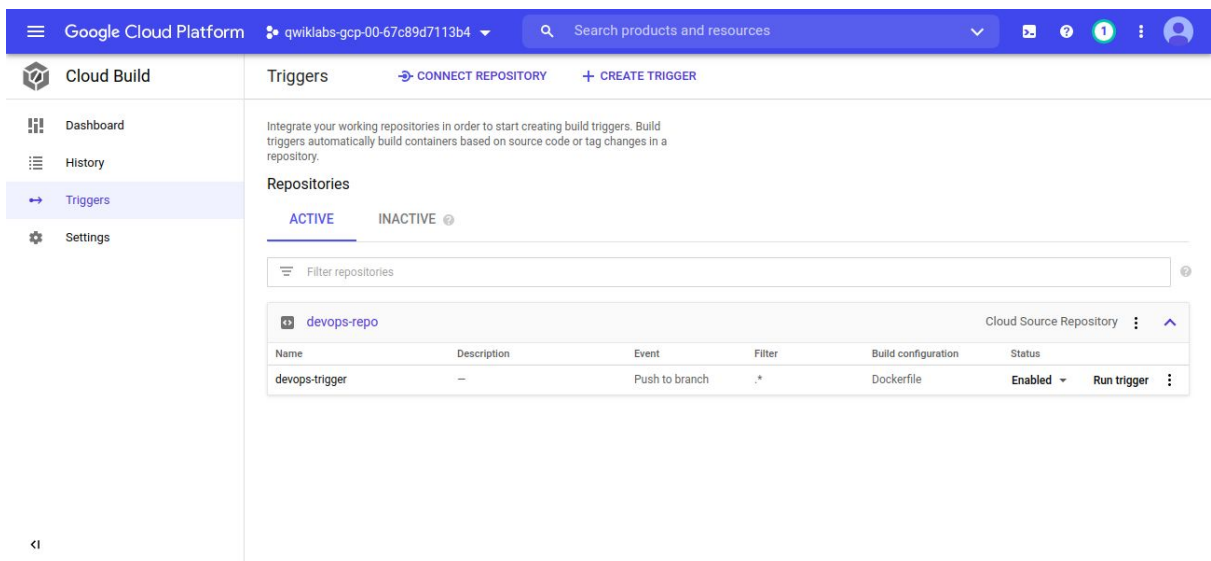
[Building image]

The image has been saved in the **Container Registry**. Container Registry is something like Docker hub. The one big difference between them is that Container Registry is only present on GCP while Docker Hub can be accessed by anyone.



[Our container Registry Content]

One of the features offered by the GCP is the automation of builds. The builds automation is assumed by triggers. In order to automate our builds, we created a trigger like shown below.



[The trigger we created]

With this trigger, as soon as we push a new code in our source repository, the Cloud Build automatically builds a new image for our container.

Finally, we created a Virtual machine in which we deploy our container, for testing.

Google Cloud Platform

qwiklabs-gcp-00-67c89d7113b4

Search products and resources

1

Compute Engine

VM instances

MANAGE ACCESS

SHOW INFO PANEL

LEARN

VM Instances

Instance groups

Instance templates

Sole-tenant nodes

Machine images

Disks

Snapshots

Images

TPUs

Migrate for Compute Engi...

Marketplace

<1

Filter VM instances

Columns

<input type="checkbox"/>	Name ^	Zone	Recommendation	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	devops	us-central1-a			10.128.0.2 (nic0)	35.223.237.73	SSH

[Vm instance created for test]