

GOOGL CLOUD PLATFORM PROJECT

DEPLOY YOUR FLASK APP ON GOOGLE CLOUD (APP ENGINE)

BY SHAHAD BAELAYAN

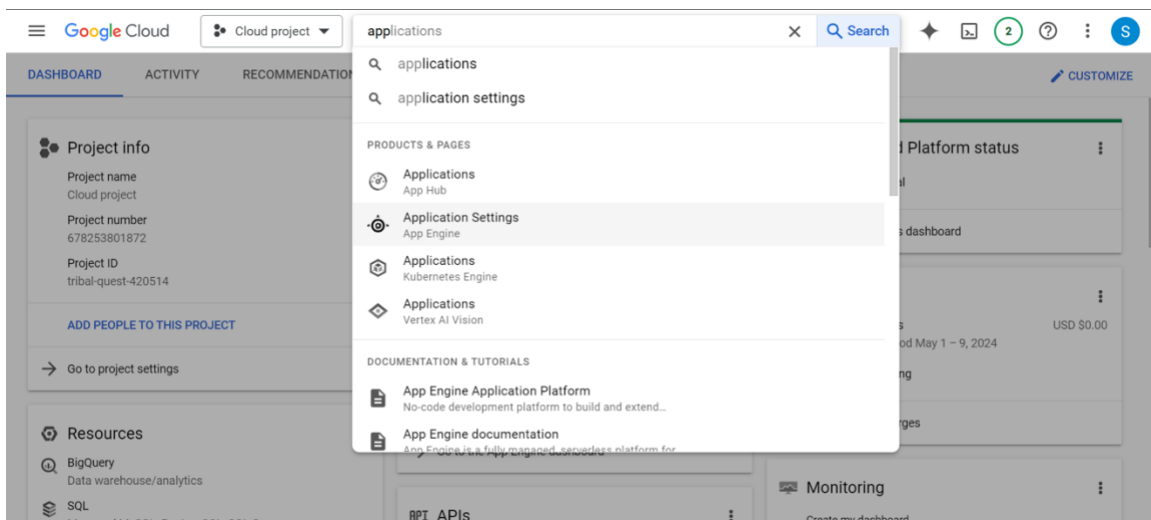
GUIDELINES:

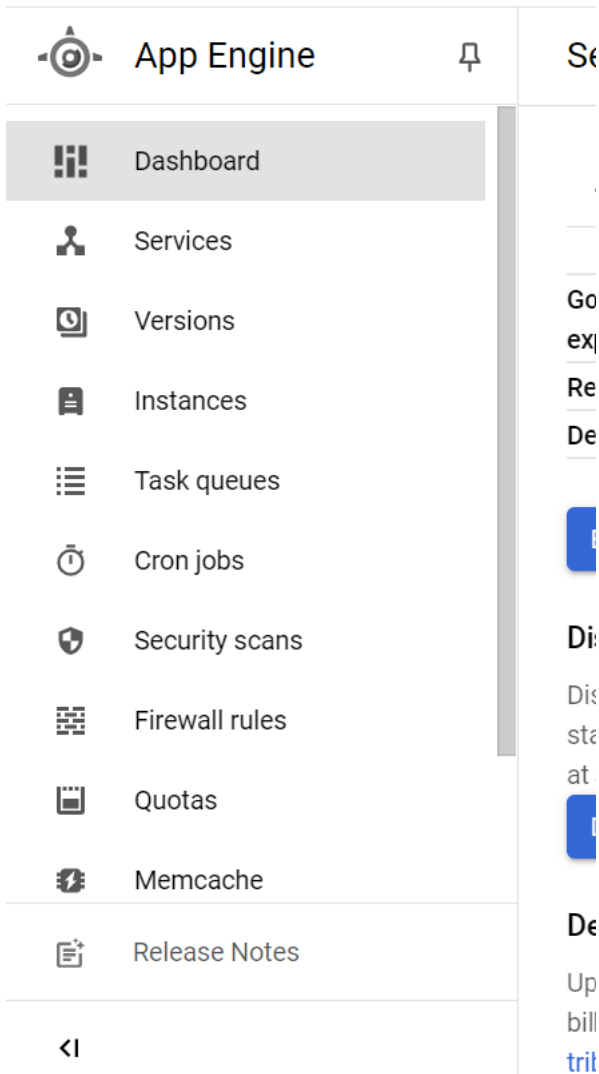
- First make your own GCP account.
- second after you make your account go to app engine and create your application.
- third deploying your flask app.

HANDS-ON-DECK \$:~

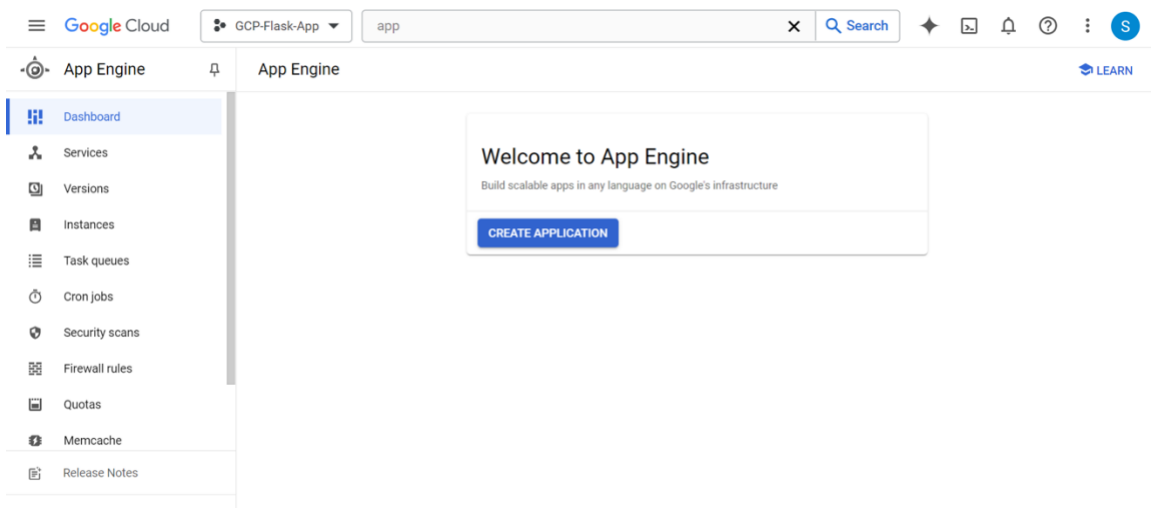
App Engine settings:

here you make your first application.





And click on create.



The application settings.

Google Cloud | GCP-Flask-App | app | Search

App Engine | Create app | LEARN

Dashboard | Services | Versions | Instances | Task queues | Cron jobs | Security scans | Firewall rules | Quotas | Memcache | Release Notes

Select a region *
us-central

Identity and API access
Select a service account
App Engine default service account

If no service account is selected the default App Engine service account will be used.

NEXT

Choose the API access and click create.

Google Cloud | GCP-Flask-App | app | Search

App Engine | Create app | LEARN

Dashboard | Services | Versions | Instances | Task queues | Cron jobs | Security scans | Firewall rules | Quotas | Memcache | Release Notes

Select a region *
us-central

Identity and API access
Select a service account
App Engine default service account

NEW SERVICE ACCOUNT | REFRESH LIST

If no service account is selected the default App Engine service account will be used.

NEXT

Click I'll do this later.

Google Cloud | GCP-Flask-App | app | Search

App Engine | Get started | LEARN

Dashboard | Services | Versions | Instances | Task queues | Cron jobs | Security scans | Firewall rules | Quotas | Memcache | Release Notes

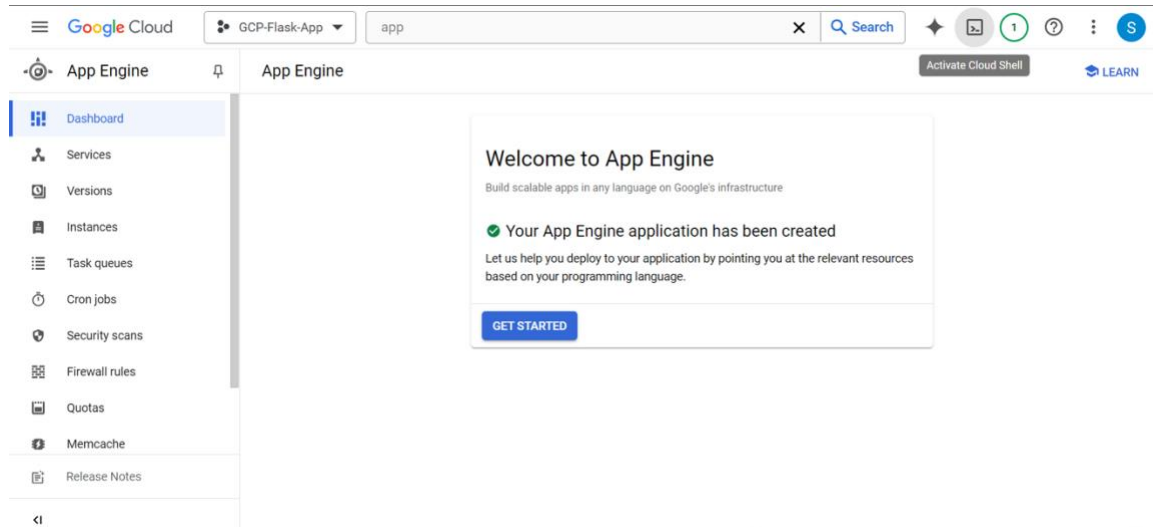
Resources
Language: Python
Environment: Standard
Read App Engine Python Standard Environment [Documentation](#).
Visit [Github](#) for Python Standard Environment code samples.

Deploy with Google Cloud SDK
DOWNLOAD THE CLOUD SDK
Initialize your SDK
\$ gcloud init
Deploy to App Engine
\$ gcloud app deploy

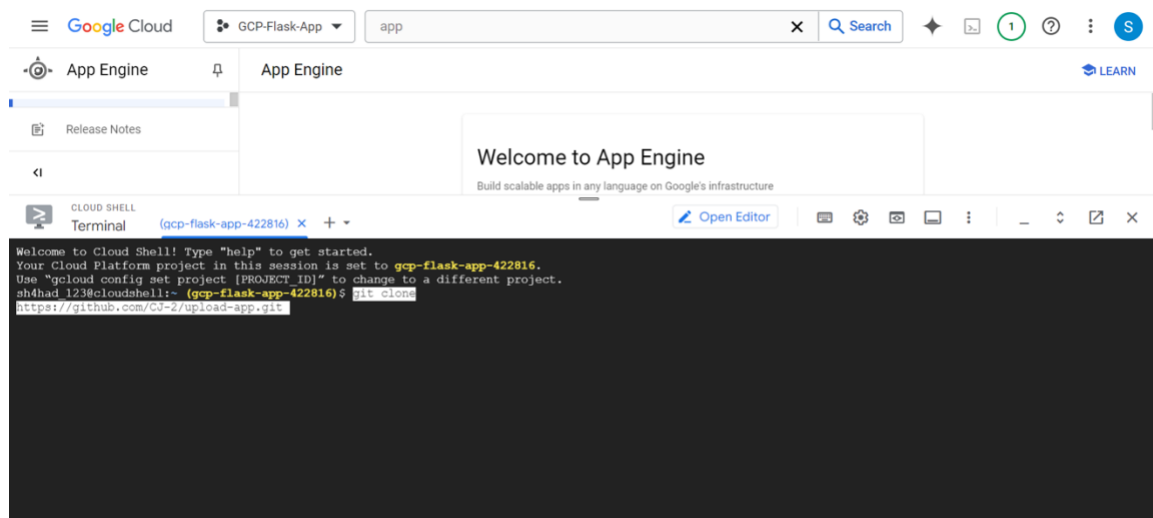
I'LL DO THIS LATER

Cloud Shell:

After we create our first application we will activate cloud shell.



After we open the shell cmd: `git clone https://github.com/CJ-2/upload-app.git`



cmd: `ls`



cmd: `cd upload-app`



cmd: **ls**

```
sh4had_123@cloudshell:~/upload-app (gcp-flask-app-422816)$ ls
README.md storage
sh4had_123@cloudshell:~/upload-app (gcp-flask-app-422816)$
```

cmd: **cd storage**

cmd: **ls**

```
sh4had_123@cloudshell:~/upload-app (gcp-flask-app-422816)$ cd storage
sh4had_123@cloudshell:~/upload-app/storage (gcp-flask-app-422816)$ ls
app.yaml main.py requirements.txt templates uploads
sh4had_123@cloudshell:~/upload-app/storage (gcp-flask-app-422816)$
```

cmd: **python main.py**

The screenshot shows the Google Cloud App Engine console. The 'Terminal' tab is active, displaying the following output:

```
sh4had_123@cloudshell:~ (gcp-flask-app-422816)$ ls
README-cloudshell.txt upload-app
sh4had_123@cloudshell:~/upload-app (gcp-flask-app-422816)$ cd upload-app
sh4had_123@cloudshell:~/upload-app (gcp-flask-app-422816)$ ls
README.md storage
sh4had_123@cloudshell:~/upload-app (gcp-flask-app-422816)$ cd storage
sh4had_123@cloudshell:~/upload-app/storage (gcp-flask-app-422816)$ ls
app.yaml main.py requirements.txt templates uploads
sh4had_123@cloudshell:~/upload-app/storage (gcp-flask-app-422816)$ python main.py
* Serving Flask app 'main'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://10.88.0.4:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 864-516-226
```

Click on web preview and follow the steps.

The screenshot shows the Google Cloud App Engine console with the 'Web Preview' button highlighted. A dropdown menu is open, showing the following options:

- Preview on port 8080
- Change port
- About web preview

Google Cloud GCP-Flask-App app

App Engine

Welcome to App Engine

Build scalable apps in any language on Google's infrastructure

Terminal (gcp-flask-app-422816)

```
sh4had 123@cloudshell:~ (gcp-flask-app-422816) $ ls
README-cloudshell.txt  upload-app
sh4had 123@cloudshell:~ (gcp-flask-app-422816) $ cd upload-app
sh4had 123@cloudshell:~/upload-app (gcp-flask-app-422816) $ ls
README.md  storage
sh4had 123@cloudshell:~/upload-app (gcp-flask-app-422816) $ cd storage
sh4had 123@cloudshell:~/upload-app/storage (gcp-flask-app-422816) $ ls
app.yaml  main.py  requirements.txt  templates  uploads
sh4had 123@cloudshell:~/upload-app/storage (gcp-flask-app-422816) $ python main.py
* Serving Flask app 'main'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://10.88.0.4:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 864-516-226
```

Preview on port 8080
Change port
About web preview

Google Cloud GCP-Flask-App app

App Engine

Welcome to App Engine

Build scalable apps in any language on Google's infrastructure

Terminal (gcp-flask-app-422816)

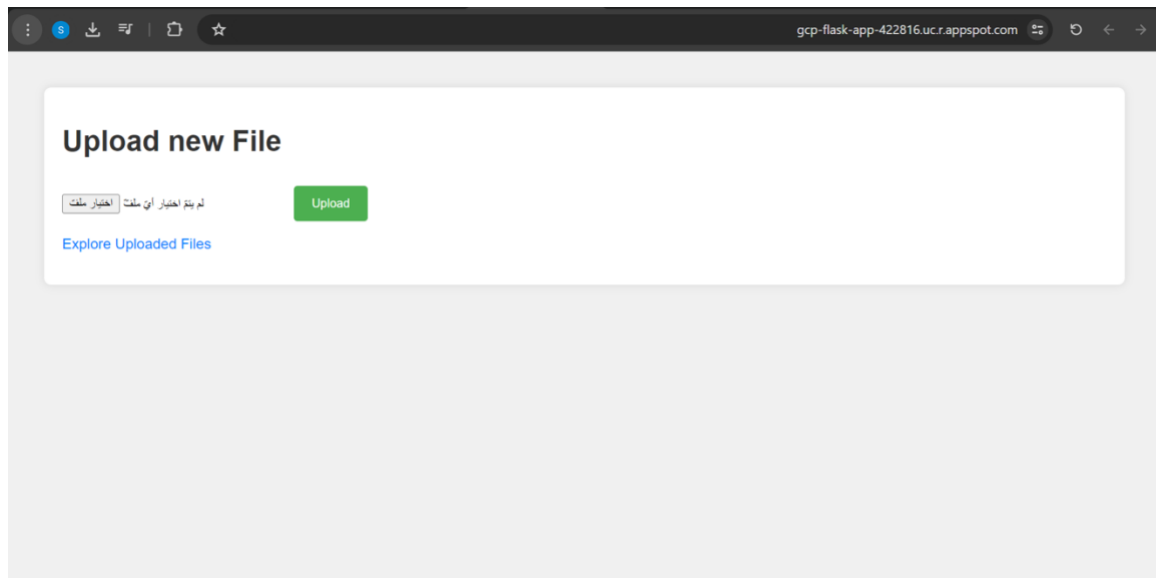
```
sh4had 123@cloudshell:~ (gcp-flask-app-422816) $ ls
README-cloudshell.txt  upload-app
sh4had 123@cloudshell:~ (gcp-flask-app-422816) $ cd upload-app
sh4had 123@cloudshell:~/upload-app (gcp-flask-app-422816) $ ls
README.md  storage
sh4had 123@cloudshell:~/upload-app (gcp-flask-app-422816) $ cd storage
sh4had 123@cloudshell:~/upload-app/storage (gcp-flask-app-422816) $ ls
app.yaml  main.py  requirements.txt  templates  uploads
sh4had 123@cloudshell:~/upload-app/storage (gcp-flask-app-422816) $ python main.py
* Serving Flask app 'main'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://10.88.0.4:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 864-516-226
```

Change Preview Port

Port Number *

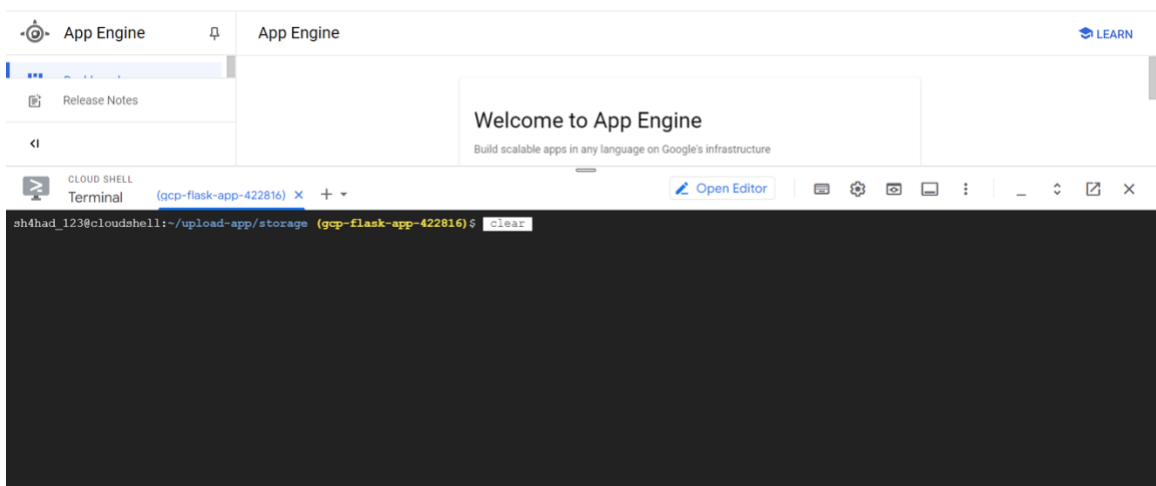
5000

CANCEL CHANGE AND PREVIEW



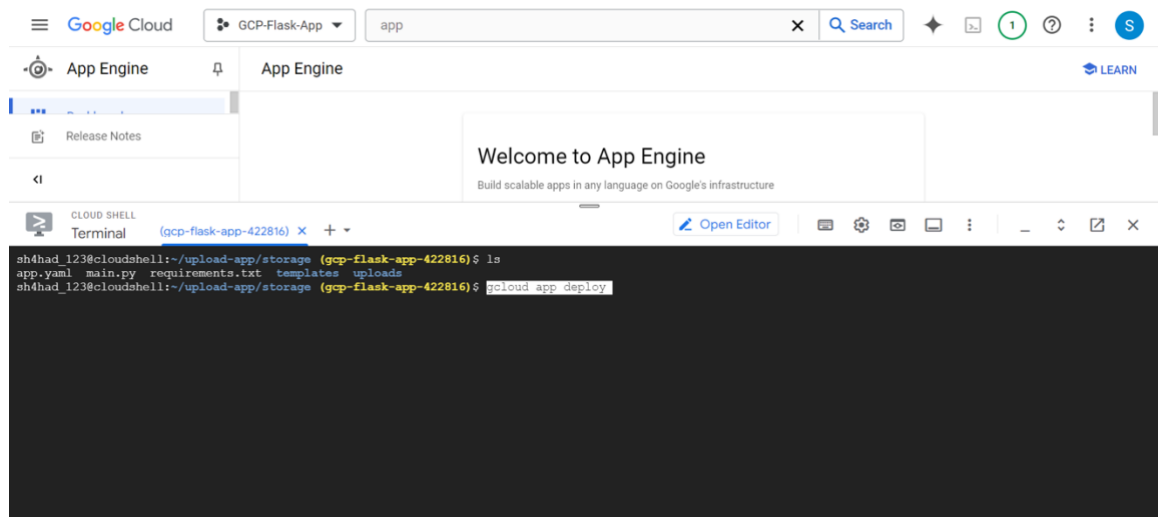
Return to shell and cmd: **CTRL+C**

cmd: **clear**



cmd: **ls**

cmd: **gcloud app deploy**



cmd: enter **y**

```
sh4had_123@cloudshell:~/upload-app/storage (gcp-flask-app-422816)$ gcloud app deploy
Services to deploy:

descriptor:      [/home/sh4had_123/upload-app/storage/app.yaml]
source:          [/home/sh4had_123/upload-app/storage]
target project:  [gcp-flask-app-422816]
target service:  [default]
target version:  [20240509t171516]
target url:      [https://gcp-flask-app-422816.uc.r.appspot.com]
target service account: [gcp-flask-app-422816@appspot.gserviceaccount.com]

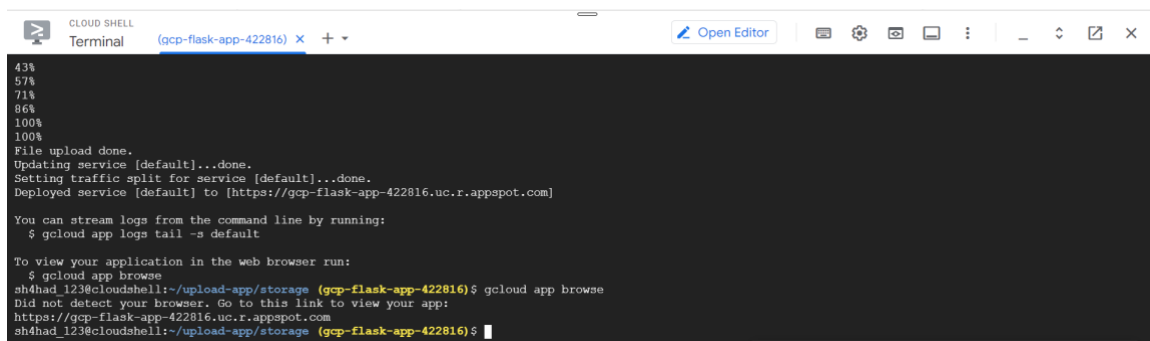
Do you want to continue (Y/n)?
```

```
target service:  [default]
target version:  [20240509t171516]
target url:      [https://gcp-flask-app-422816.uc.r.appspot.com]
target service account: [gcp-flask-app-422816@appspot.gserviceaccount.com]

Do you want to continue (Y/n)? y

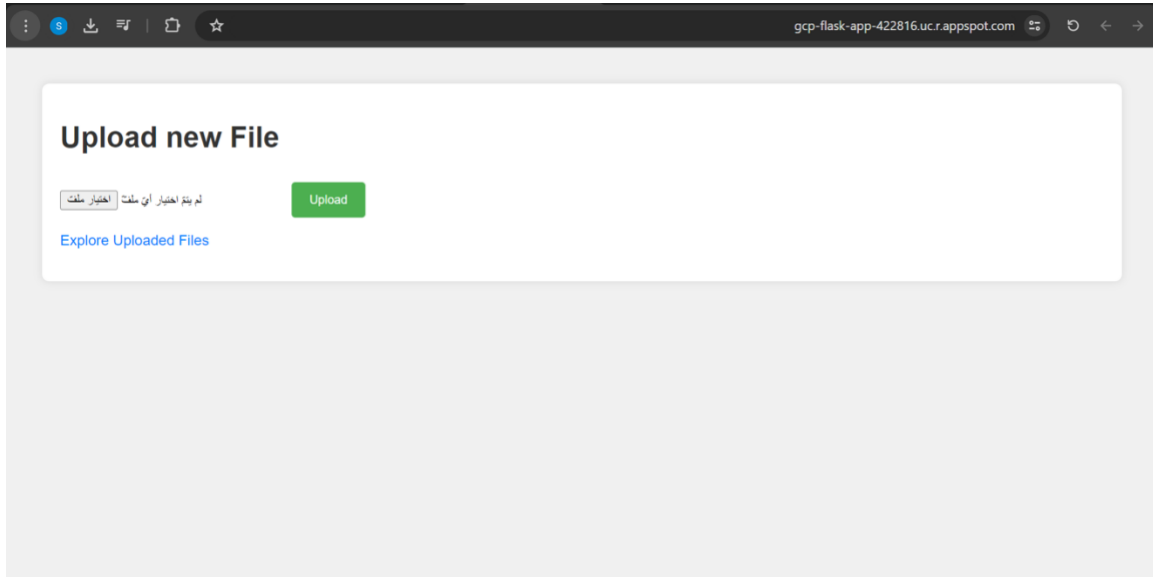
Beginning deployment of service [default]...
Uploading 7 files to Google Cloud Storage
14%
29%
43%
57%
71%
86%
100%
File upload done.
Updating service [default]...working...
```

cmd: **gcloud app browse**

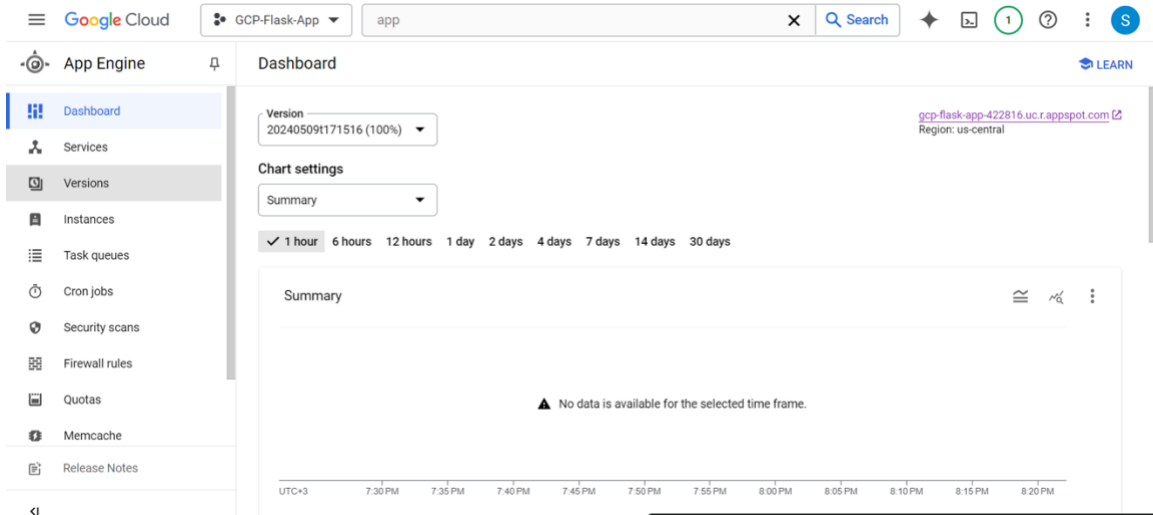


After that click on the link.

And now your app is deployed congratulations!



Go back to dashboard and select Versions.



Here's your version.

Google Cloud

GCP-Flask-App

app

Search

App Engine

Dashboard

Services

Versions

Instances

Task queues

Cron jobs

Security scans

Firewall rules

Quotas

Memcache

Release Notes

Filter

Filter versions

Version	Status	Traffic Allocation	Instances	Runtime	Environment	Size
202405091171516	Serving	100%	0	python39	Standard	7 MB

I hope that the project will help you