

Google cloud platform project

Deploy your flask app on Google Cloud (App Engine)

By Faisal Alkhayef

Guidelines:

1. Make your own GCP account.
2. Go to app engine and create your application.
3. deploy your flask app.

App Engine settings:

Make your first application:

Solutions

All products

Jump Start Solutions

Solution deployments

Categories






Management

Compute

Storage

Serverless

Build applications powered by serverless functions and containers

Name	Description
 Cloud Run	Serverless for containerized applications
 Cloud Functions	Event-driven serverless functions
 App Engine	Managed app platform
 API Gateway	API development, deployment, and management
 Endpoints	Cloud API gateway

App Engine

App Engine

Dashboard

Services

Versions

Instances

Task queues

Welcome to App Engine

Build scalable apps in any language on Google's infrastructure

[CREATE APPLICATION](#)

Settings:

- 1 **Configure application** —
- 2 **Get started**

Region

Select a region for your App Engine application. Please remember, once selected the region is permanently tied to the project.



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

Resources

Language
Python

Environment
Standard

Read App Engine Python Standard Environment [Documentation](#).

Visit [Github](#) for Python Standard Environment code samples.

[I'LL DO THIS LATER](#)

Deploy with Google Cloud SDK

[DOWNLOAD THE CLOUD SDK](#)

Initialize your SDK

```
$ gcloud init
```

Deploy to App Engine

```
$ gcloud app deploy
```

Cloud Shell(top right):

Now that the shell is open, we type git clone (insert your flask app repo)

```
CLOUD SHELL
Terminal (gcp-423112) x + v
afisal01994@cloudshell:~ (gcp-423112)$ git clone https://github.com/CJ-2/upload-app.git
```

cmd: ls

```
afisal01994@cloudshell:~ (gcp-423112)$ ls
README-cloudshell.txt  upload-app
```

cmd: cd upload-app

```
afisal01994@cloudshell:~ (gcp-423112)$ cd upload-app
```

cmd: ls

```
afisal01994@cloudshell:~/upload-app (gcp-423112)$ ls
README.md  storage
```

cmd: cd storage

```
afisal01994@cloudshell:~/upload-app (gcp-423112)$ cd storage
```

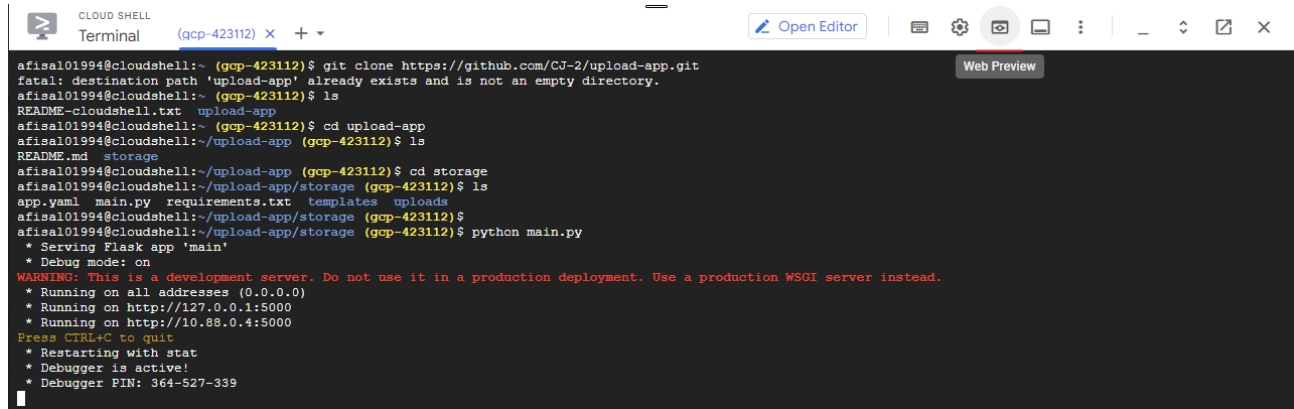
cmd: ls

```
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ ls
app.yaml  main.py  requirements.txt  templates  uploads
```

cmd: python main.py

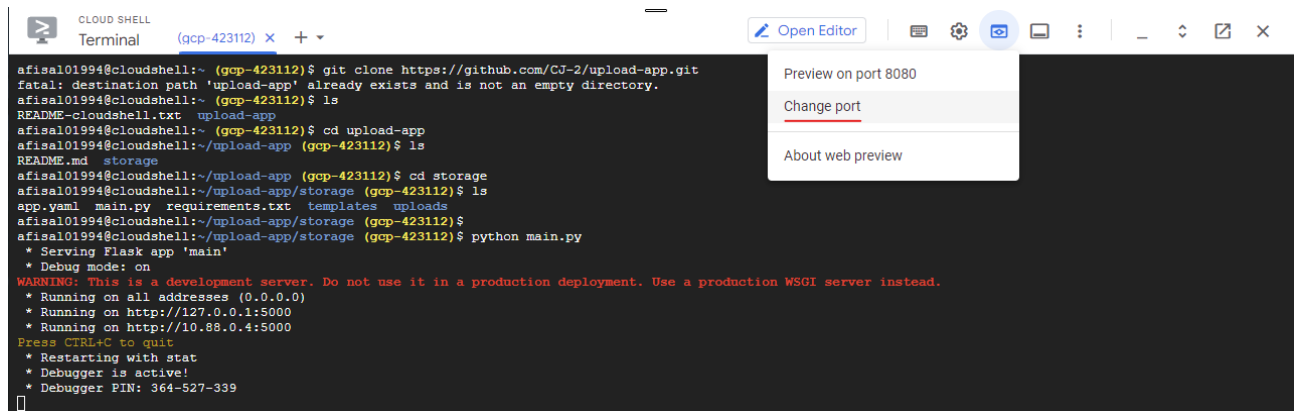
```
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ python main.py
```

Click on web preview and follow the steps:



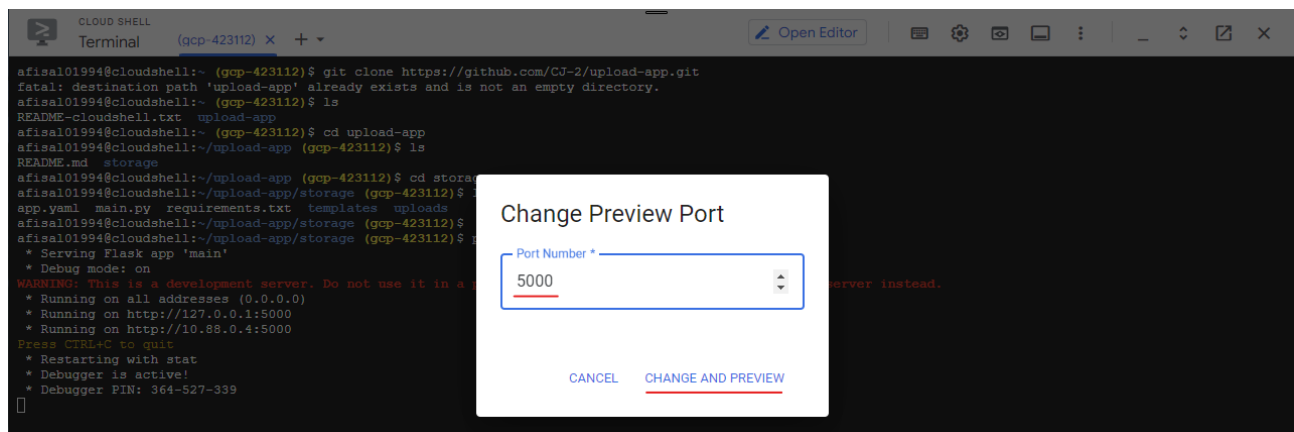
The screenshot shows a Cloud Shell terminal window with the following commands and output:

```
afisal01994@cloudshell:~ (gcp-423112)$ git clone https://github.com/CJ-2/upload-app.git
fatal: destination path 'upload-app' already exists and is not an empty directory.
afisal01994@cloudshell:~ (gcp-423112)$ ls
README-cloudshell.txt  upload-app
afisal01994@cloudshell:~ (gcp-423112)$ cd upload-app
afisal01994@cloudshell:~/upload-app (gcp-423112)$ ls
README.md  storage
afisal01994@cloudshell:~/upload-app (gcp-423112)$ cd storage
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ ls
app.yaml  main.py  requirements.txt  templates  uploads
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ 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: 364-527-339
```

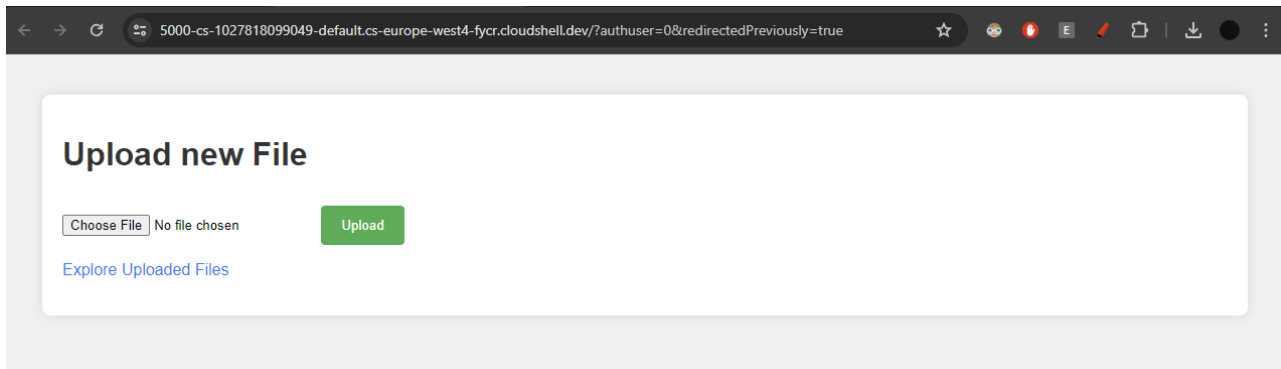


The screenshot shows the same Cloud Shell terminal window as above, but with a 'Web Preview' button in the top right corner. A dropdown menu is open, showing the following options:

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



The screenshot shows the same Cloud Shell terminal window as above, but with a 'Change Preview Port' dialog box open. The dialog box has a text input field labeled 'Port Number' with the value '5000' entered. Below the input field are two buttons: 'CANCEL' and 'CHANGE AND PREVIEW'.



Return to shell and cmd: CTRL+C

cmd: clear

```
Cloud Shell Terminal (gcp-423112)
afisal01994@cloudshell:~ (gcp-423112)$ git clone https://github.com/CJ-2/upload-app.git
fatal: destination path 'upload-app' already exists and is not an empty directory.
afisal01994@cloudshell:~ (gcp-423112)$ ls
README-cloudshell.txt  upload-app
afisal01994@cloudshell:~ (gcp-423112)$ cd upload-app
afisal01994@cloudshell:~/upload-app (gcp-423112)$ ls
README.md  storage
afisal01994@cloudshell:~/upload-app (gcp-423112)$ cd storage
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ ls
app.yaml  main.py  requirements.txt  templates  uploads
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ 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: 364-527-339
127.0.0.1 - - [12/May/2024 13:16:34] "GET /?authuser=0&redirectedPreviously=true HTTP/1.1" 200 -
127.0.0.1 - - [12/May/2024 13:16:34] "GET /favicon.ico HTTP/1.1" 404 -
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ clear
```

cmd: ls

```
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ ls
app.yaml  main.py  requirements.txt  templates  uploads
```

cmd: gcloud app deploy (you might need to authorize)

```
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ gcloud app deploy
```

cmd: enter Y

```
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ gcloud app deploy
Services to deploy:

descriptor:      [/home/afisal01994/upload-app/storage/app.yaml]
source:          [/home/afisal01994/upload-app/storage]
target project:  [gcp-423112]
target service:  [default]
target version:  [20240512t132239]
target url:      [https://gcp-423112.uc.r.appspot.com]
target service account: [gcp-423112@appspot.gserviceaccount.com]

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

```
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ gcloud app deploy
Services to deploy:

descriptor:          [/home/afisal01994/upload-app/storage/app.yaml]
source:              [/home/afisal01994/upload-app/storage]
target project:      [gcp-423112]
target service:      [default]
target version:      [20240512t132239]
target url:          [https://gcp-423112.uc.r.appspot.com]
target service account: [gcp-423112@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%
100%
File upload done.
Updating service [default]...working...☐
```

```
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%
100%
File upload done.
Updating service [default]...done.
Setting traffic split for service [default]...done.
Deployed service [default] to [https://gcp-423112.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
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ ☐
```

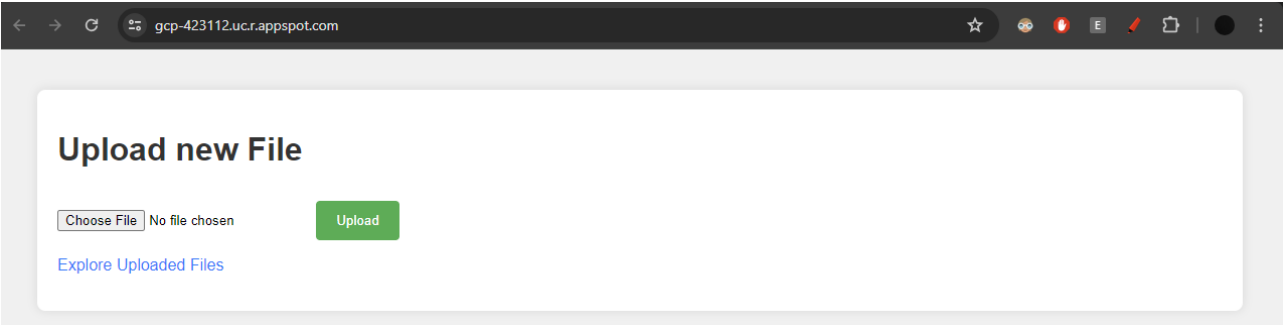
cmd: gcloud app browse

```
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ gcloud app browse
```

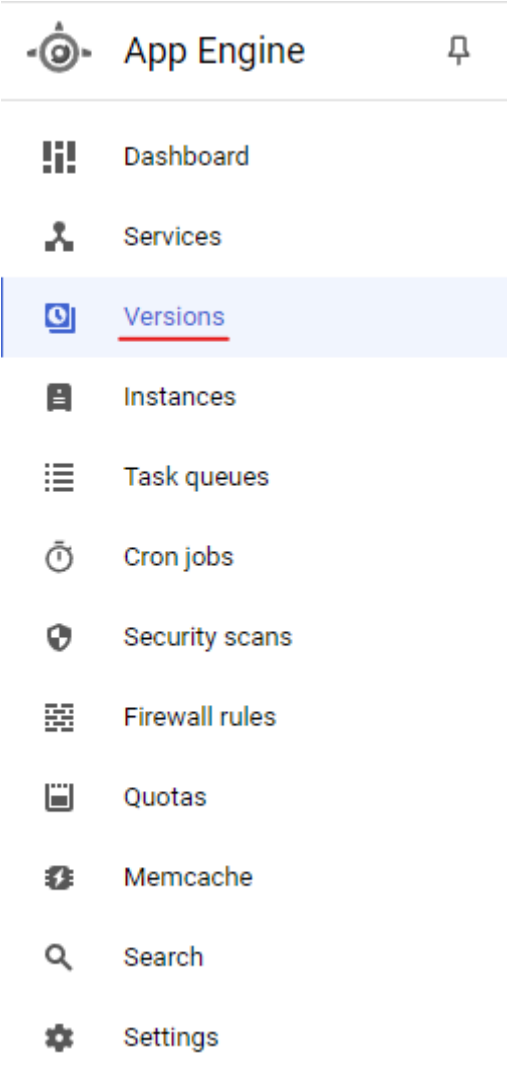
Then click on the link

```
afisal01994@cloudshell:~/upload-app/storage (gcp-423112)$ gcloud app browse
Did not detect your browser. Go to this link to view your app:
https://gcp-423112.uc.r.appspot.com
```

And now your app is deployed:



Go back to your app engine and select Versions:



And here is your version:

<input type="checkbox"/>	Version	Status	Traffic Allocation	Instances [?]	Runtime	Environment	Size
<input type="checkbox"/>	20240512t132239	Serving	<div><div></div></div> 100%	0	python39	Standard	6.9 MB

Hope this documentation helps.