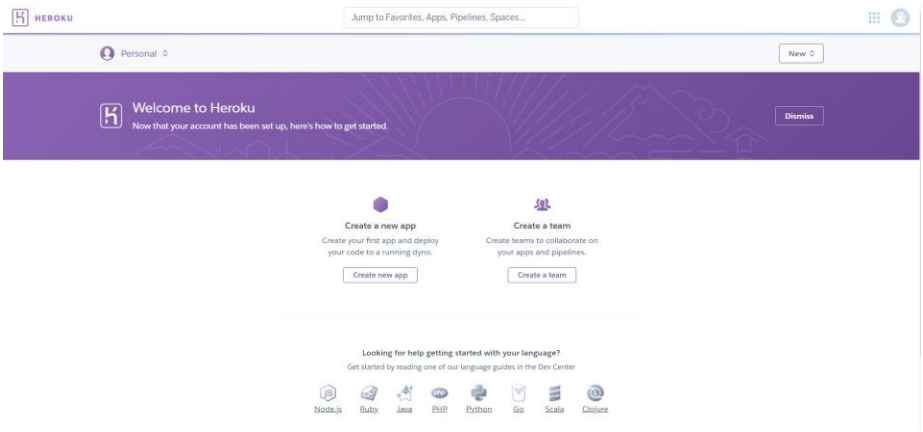
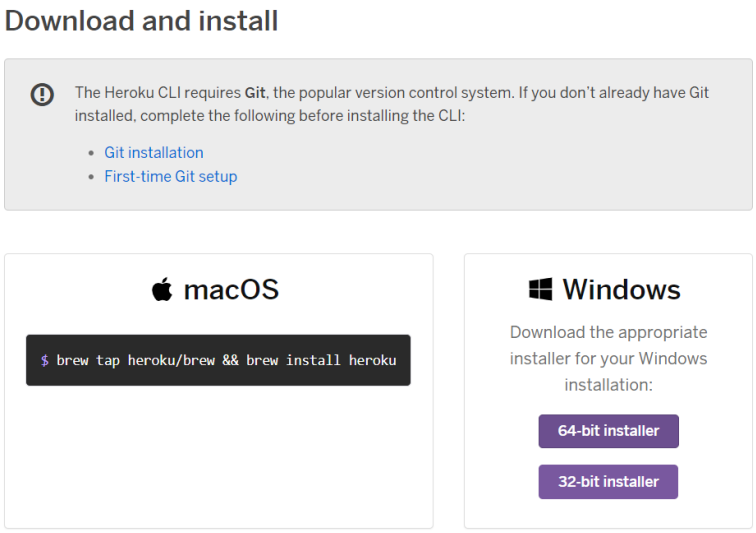
Heroku Deployment

Step 1: Create an account in Heroku.

Once you create and done your verification through mail , then main page looks like this



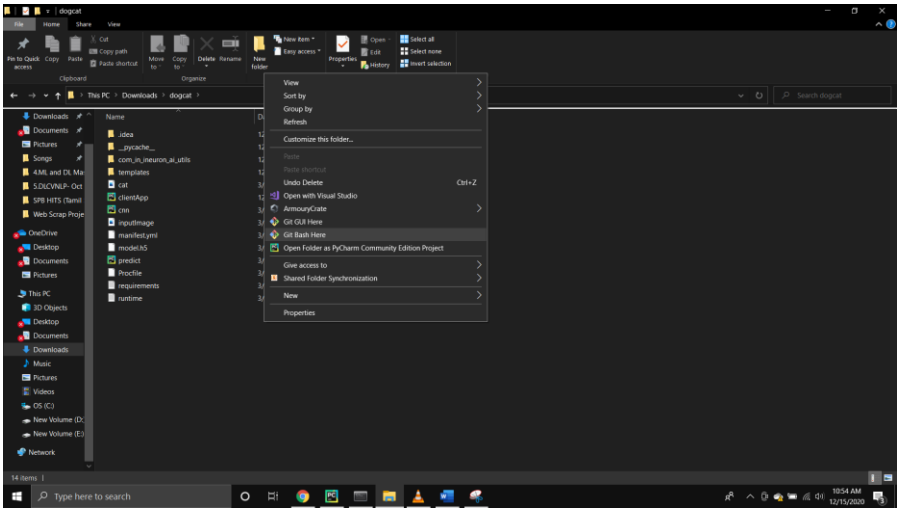
Step 2: Install the Heroku CLI



Step3 : Install the Git

Step 4:

Navigate to your project folder --- >right click ---- > choose Git Bash Here

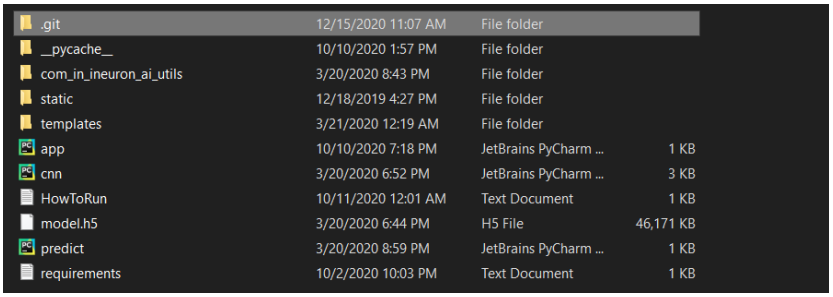


The page window looks like below

Step 5: Git init

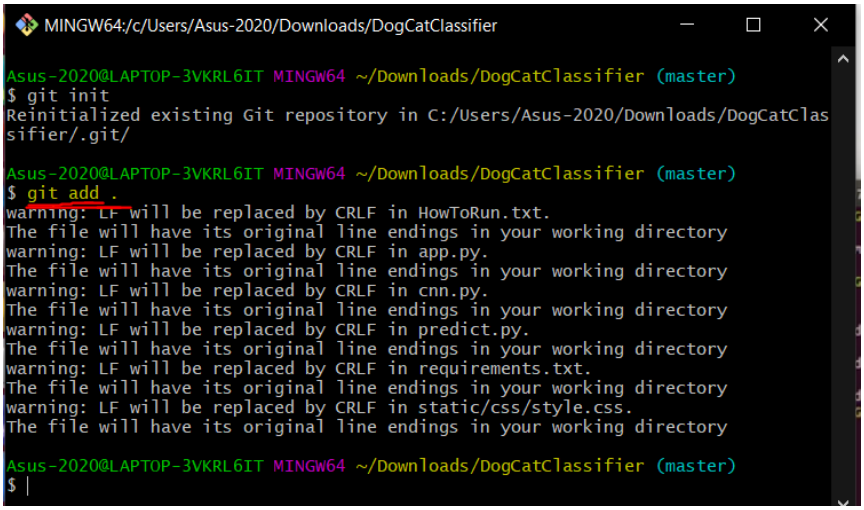
Then type : git init (to initialize the git(start tracking it – Changes, who did the changes, time, which file))

Git folder is created in our project folder

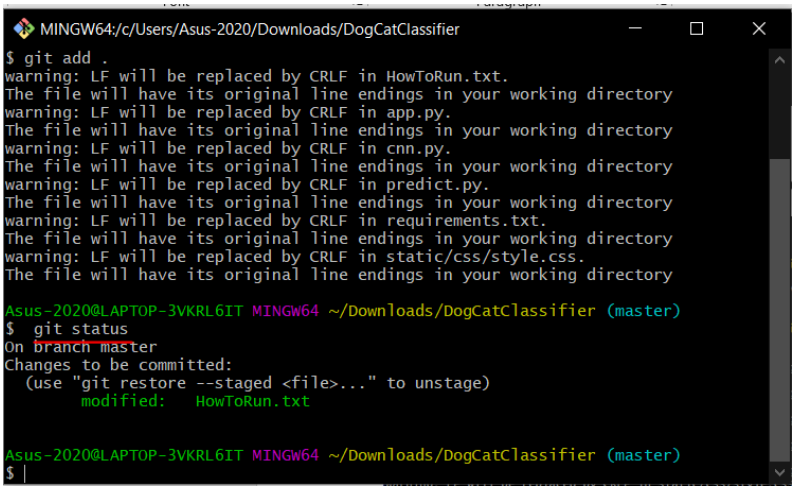


Step 6: git add .

Now add all the files which I have in my project folder.

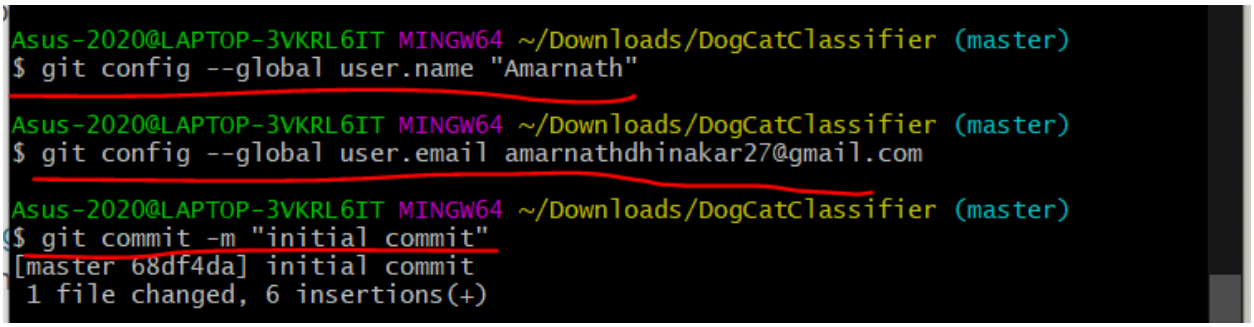


Step 7: git status

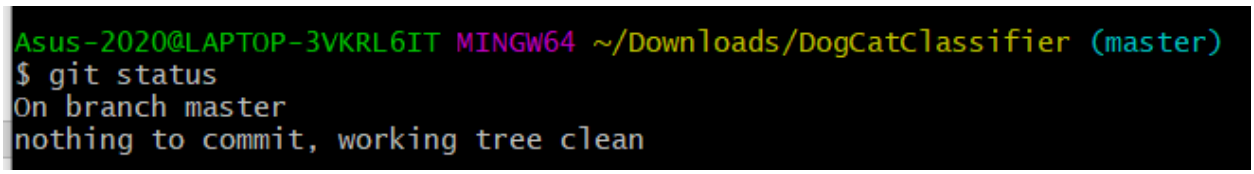


Step 8: git commit -m “initial commit”

If you use the git bash for the very first time then it will ask your username and mail once you done then try to commit again and you will the message like below.

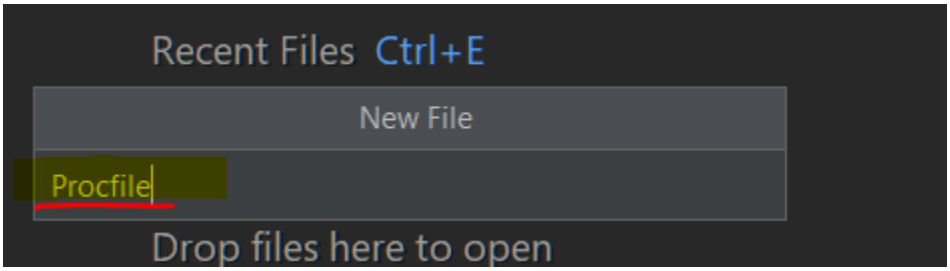


If I run the git status command again and it says that nothing to commit.



Step 9:

We need to add one file inside in our project folder and name it as Procfile without any extension.



And add the below line into the procfile

```
web: gunicorn app:app
```

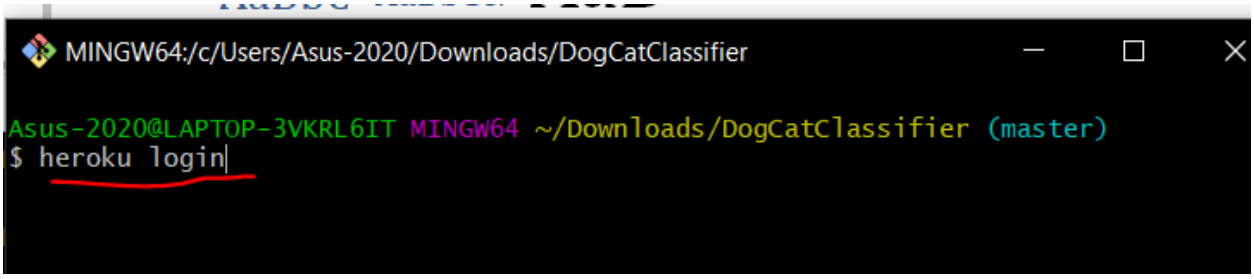
Here web is basically a web server and gunicorn is a kind of server and which will used on a cloud side. app:app (app is the file name : app is the flask object name)

Then go to git bash, check the git status and do the git add and commit for the changes that we had done in the project folder.

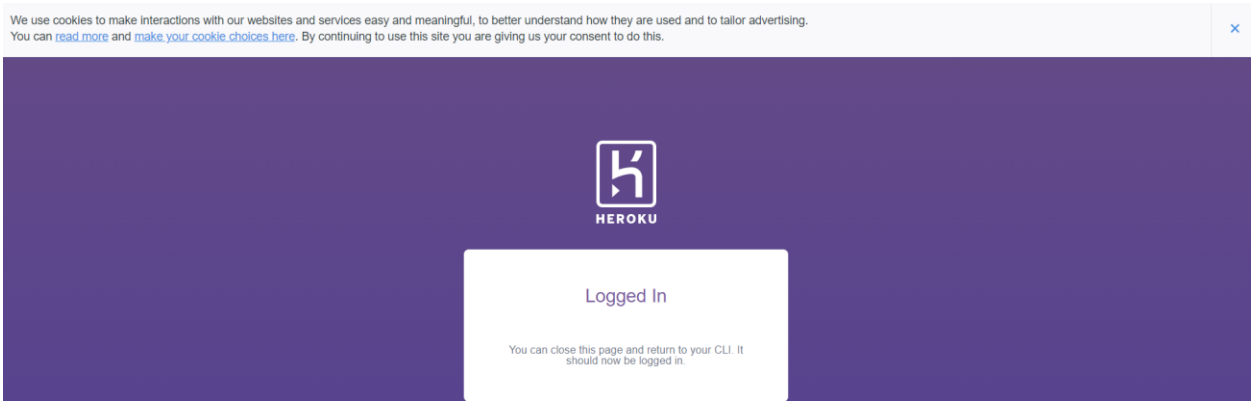
```
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git add .

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   .idea/.gitignore
        new file:   Procfile


Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git commit -m "initial commit"
[master bfecdc7] initial commit
 2 files changed, 3 insertions(+)
 create mode 100644 .idea/.gitignore
 create mode 100644 Procfile
```

Step 10:

Open the Git bash and type Heroku login



Then it ask us to hit any key to open the browser to login to Heroku account and once logged in the it will show the message like below

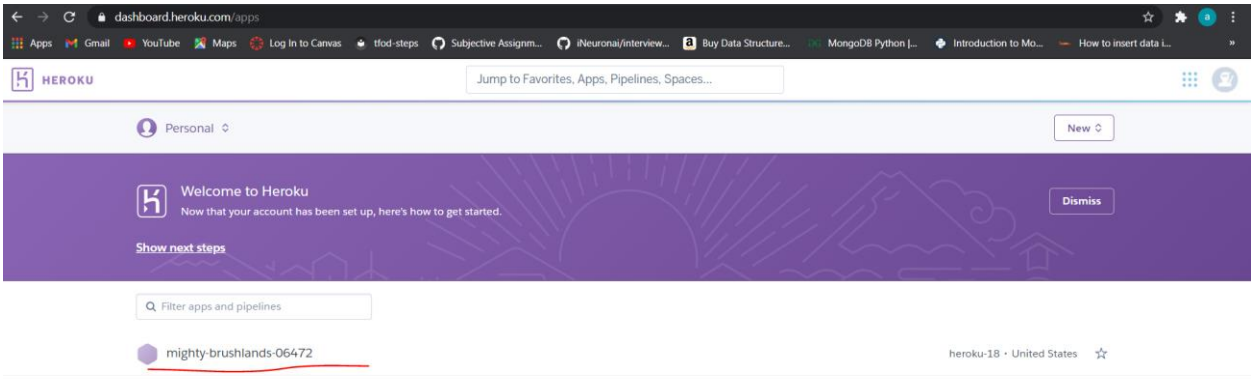


```
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ heroku login
 »   Warning: Our terms of service have changed:
 »   https://dashboard.heroku.com/terms-of-service
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/cli/browser/07524ac7-3ccd-41
07-8460-a148e10e89d9?requestor=SFMyNTY.g2gDbQAAAA0xMDMuMTAxLjE3Mi43bgYAmMInZXYBY
gABUYA.IOeuCo6RDwV8IH6r4rxKMTX26p-7BynvPuV4UQ_tEw8
heroku: Waiting for login...
Logging in... done
Logged in as amarnathdhinakar27@gmail.com
```

Step 11:

Type heroku create in git bash to create one blank Heroku on a Heroku cloud

```
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ heroku create
Creating app... done, mighty-brushlands-06472
https://mighty-brushlands-06472.herokuapp.com/ | https://git.heroku.com/mighty-b
rushlands-06472.git
```



Step 12:

Type : git remote -v

It gives me the url for push something to the cloud

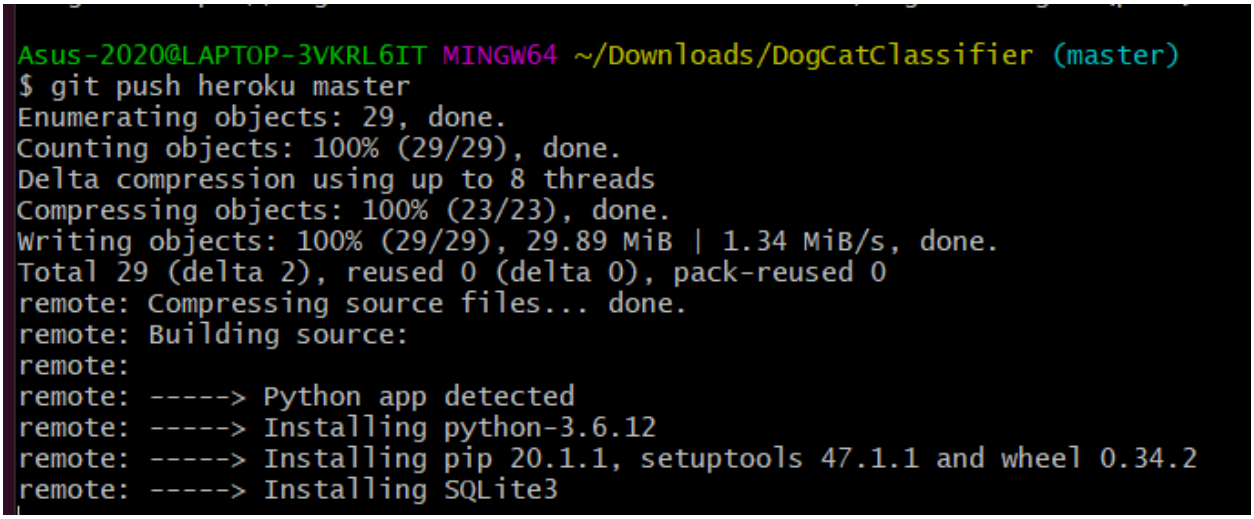
```
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git remote -v
heroku  https://git.heroku.com/mighty-brushlands-06472.git (fetch)
heroku  https://git.heroku.com/mighty-brushlands-06472.git (push)
origin  https://dogcattdl.scm.azurewebsites.net:443/dogcattdl.git (fetch)
origin  https://dogcattdl.scm.azurewebsites.net:443/dogcattdl.git (push)
```

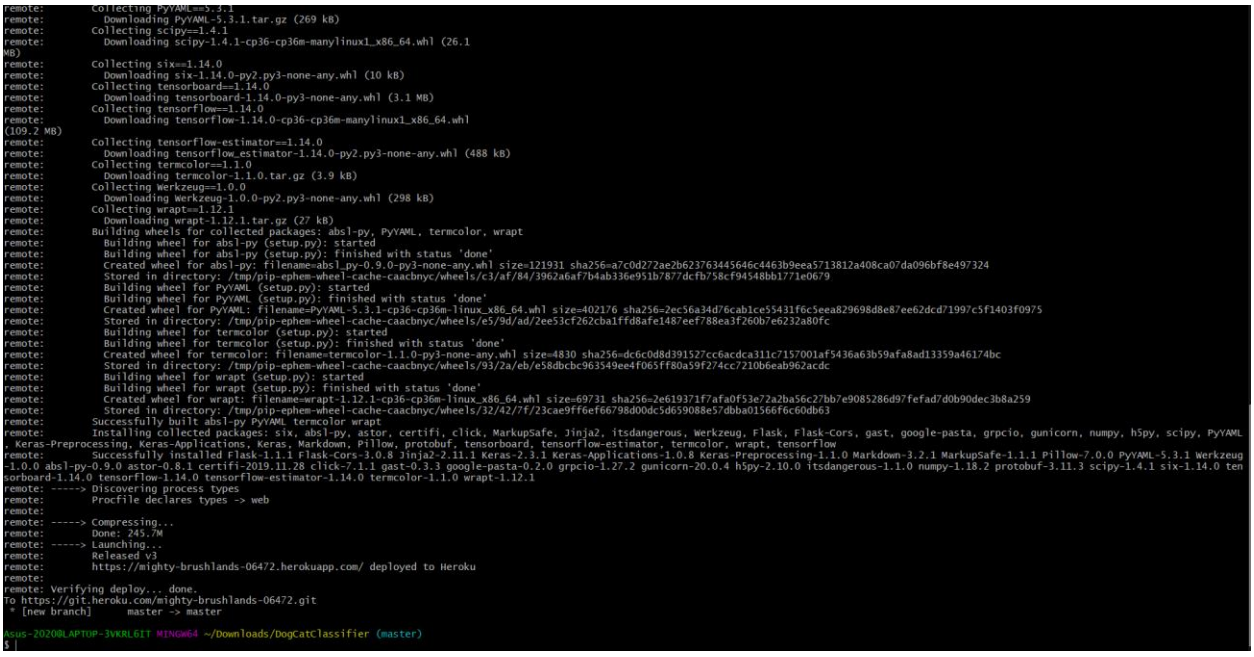
Step 13:

Type : git push heroku master

Whenever we try to push or fetch some data’s in the cloud instead of using the whole url I can type just Heroku push or fetch as here. Here Heroku is the alias for this whole url.

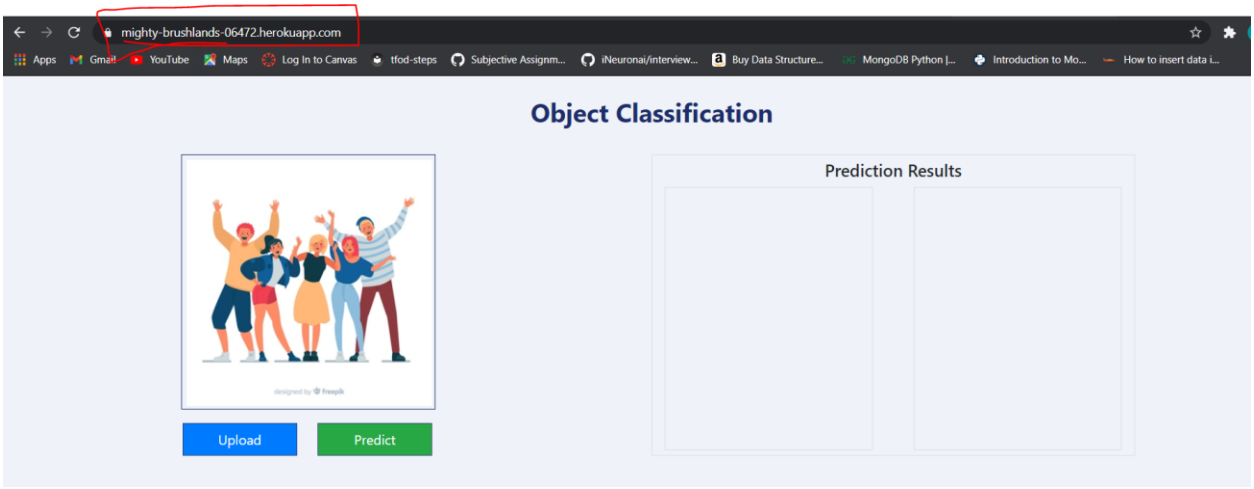
And it will install the necessary packages in the cloud app.





```
remote:
remote: -----> Compressing...
remote:        Done: 245.7M
remote: -----> Launching...
remote:        Released v3
remote:        https://mighty-brushlands-06472.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/mighty-brushlands-06472.git
 * [new branch]      master -> master
```

Finally we will be able to see this deployed our application url.



https://mighty-brushlands-06472.herokuapp.com/