**Heroku Deployment steps:**

Before deploying into heroku, we need to do heroku configuration.

1. Update the pipfile with latest python version **“3.7”**

**Python\_version = “3.7”**

Run the below command to generate the appropriate pipfile.lock

**Pipenv lock**

1. Create Procfile: touch Procfile

web: gunicorn nhsee\_project.wsgi --log-file -

1. In settings.py file,

**ALLOWED\_HOSTS = ['\*']**

1. Install gunicorn which we will use in production while still using Django’s internal server for local development use.

**pipenv install gunicorn==20.0.0**

1. For Static assets usage purpose, i have added below statements in setting.py file, before that i installed whitenose in project.

**pipenv install whitenoise**

1. Adding static root and static storage details

**STATIC\_URL = '/static/'**

**STATIC\_ROOT = os.path.join(BASE\_DIR, "staticfiles")**

**STATICFILES\_DIRS = [os.path.join(BASE\_DIR, 'static')]**

**STATICFILES\_STORAGE = 'whitenoise.storage.CompressedManifestStaticFilesStorage'**

1. middleware section,

MIDDLEWARE = [

'django.middleware.security.SecurityMiddleware',

**'whitenoise.middleware.WhiteNoiseMiddleware',**

1. Creating a requirement.text file,

**pip freeze > requirements.txt**

1. **pipenv install -r requirements.txt**
2. Commit the code, push into branch, after that merge into sprint and master as well
3. **Heroku login**
4. **Heroku create nhsee-team-4**
5. **Heroku git:remote -a nhsee-team-4**
6. DEBUG\_COLLECTSTATIC configuration,

**heroku config:set DEBUG\_COLLECTSTATIC=1**

1. Disable the collectstatic build step with the DISABLE\_COLLECTSTATIC configuration:

**heroku config:set DISABLE\_COLLECTSTATIC=1**

1. Push the code into heroku and add free scaling so it's actually running online.

**Git push heroku master**

**Heroku ps:scale web=1**

1. **Heroku open**

Issues:

1. Error like,

Scaling dynos… !

Count not find that process type(web)

Solution for the above issue is, need to merge code with master , after that i am successfully deployed code into heroku.

1. After 1st issue resolved, Successfully Web application running on heroku, but it was not displaying application in proper format because of css files issues, heroku was not allowed to use static folder files.

For Static assets usage purpose, i have added below statements in setting.py file, before that i installed whitenose in project.

pip install whitenoise (have not added whitenoise details into requirement.txt file)

Adding static root and static storage details

STATIC\_URL = '/static/'

STATIC\_ROOT = os.path.join(BASE\_DIR, "staticfiles")

STATICFILES\_DIRS = [os.path.join(BASE\_DIR, 'static')]

STATICFILES\_STORAGE = 'whitenoise.storage.CompressedManifestStaticFilesStorage'

MIDDLEWARE = [

'django.middleware.security.SecurityMiddleware',

'whitenoise.middleware.WhiteNoiseMiddleware',

Committed and pushed code into master.

DEBUG\_COLLECTSTATIC configuration,

heroku config:set DEBUG\_COLLECTSTATIC=1

Disable the collectstatic build step with the DISABLE\_COLLECTSTATIC configuration:

heroku config:set DISABLE\_COLLECTSTATIC=1

1. After fixing the 2nd issue and deployed code into heroku. Still got issue like application error on web page,

Too see the heroku logs,

**Heroku logs --tail**

Error:

2019-12-09T14:53:52.915211+00:00 app[web.1]: gunicorn.errors.HaltServer: <HaltServer 'Worker failed to boot.' 3>

2019-12-09T14:53:53.012276+00:00 heroku[web.1]: State changed from starting to crashed

2019-12-09T14:53:52.999155+00:00 heroku[web.1]: Process exited with status 1

2019-12-09T14:54:40.969928+00:00 heroku[router]: at=error code=H10 desc="App crashed" method=GET path="/" host=nhsee-project-5.herokuapp.com request\_id=879d78a2-9031-4ae2-93c1-8b521a3a6cc2 fwd="73.123.114.101" dyno= connect= service= status=503 bytes= protocol=https

H10 code -- app crashed

Solution: I added whitenoise with version in requirement.txt file and deployed code again into heroku master.

**Reference links:**

<https://devcenter.heroku.com/articles/django-assets>

https://www.codementor.io/jamesezechukwu/how-to-deploy-django-app-on-heroku-dtsee04d4