

Docker + Django + Postgres Add-on + Heroku

So I'm running through the following:

7

- Created a Dockerfile and a docker-compose.yml to develop a django app locally, nothing to do here, everything works just fine.

- Dockerfile



5

```
FROM python:3
ENV PYTHONUNBUFFERED 1
RUN mkdir /code
WORKDIR /code
ADD requirements.txt /code/
RUN pip install -r requirements.txt
ADD . /code/
```

- All it does is fetches python3 image, copies the code from my source folder to the docker image
- Then I'm trying to deploy this to heroku, so I added postgres add-on on heroku apps, and configured it using `dj_database_url`, with the following command on my settings.py file :

```
DATABASES = {}
DATABASES['default'] = dj_database_url.config()
```

- After that I pushed my application to heroku using `heroku container:push web` and it got pushed to my app no problem at all...
- The final piece is when I open my application using `heroku open` it says I got an error:
Application error
An error occurred in the application and your page could not be served. If you are the application owner, check your logs for details.
- This is what get printed on the heroku website
- Now, when I log using `heroku logs -t`, it says that exited with code 0, then it crashes

What am I missing?

- Plus: when I run `heroku run web python manage.py migrate` it migrates the data on the heroku app
- Plus 2: when I run `heroku run web python manage.py runserver` it says is running on 127.0.0.1:8000

EDIT 1: If you want to reproduce the problem you can clone all my code in the following git repo:
<https://github.com/giovanniKleinCampigoto/djangoherokuproblem.git>

EDIT 2:

In this official tutorial: <https://devcenter.heroku.com/articles/deploying-python>, says to use virtualenv, but I'm using docker, so I don't really know how to proceed...

EDIT 3: Also, I noticed that when I push the image to heroku using: `heroku container:push web`, the server re-deploys, but instead of running whatever is on the procfile it just logs this: Starting process with command `python3 1`

with command `python3 j`

EDIT 4: When running `heroku local web`, and editing the Procfile to run `docker-compose up` works just fine, the problem is with the upload to heroku's server...not sure if I should edit the file to just `python manage.py runserver`...

`docker-compose.yml`

```
version: '2'

services:
  web:
    build: .
    env_file: composeexample/.env
    command: python3 manage.py runserver 0.0.0.0:8000
    volumes:
      - ./code
    ports:
      - "8000:8000"
```

django

heroku

edited Sep 14 '17 at 21:27

asked Sep 14 '17 at 18:14



Giovanni Klein
Campigoto

190 1 12

try to change running server on heroku to `heroku run web python manage.py runserver 0.0.0.0:5000` inside your ProcFile . I've some issues running flask and django on heroku, so, this worked for me. – Abe Sep 14 '17 at 18:18 ✎

Aren't we supposed to listen to the \$PORT variable? Wouldn't it be `0.0.0.0:$PORT`, plus I don't believe that my procfile get's ever executed... – Giovanni Klein Campigoto Sep 14 '17 at 18:25 ✎

2 Answers



How I did not find any answers to this I'm answering my own question, so others can use my solution...

8



Ok so after a lot of testing, coming and goings I have discovered the problem.



So what happened was the following: for local development you should use docker-compose, mount volumes and all that yummy stuff, however when going on production with heroku you need to set a command for server starting. So if you ever plan to setup a project with docker + django + postgres add-on on heroku. Do the following steps.

First off:

- Follow docker official setup for local development: <https://docs.docker.com/compose/django/>
- Then with docker-compose up command server should run locally at port 8000

- After all the development is done and you have written your Dockerfile, it's time to deploy it
- Follow this guide: <https://devcenter.heroku.com/articles/django-app-configuration>
- Be sure to add postgres add-on on your heroku app
- In settings.py under databases modify to the following: `DATABASES = {} DATABASES['default'] = dj_database_url.config(default='DATABASE_URL')`
- This DATABASE_URL is an environment variable that you can get in your heroku app, go to your dashboard on heroku, find your app, go to settings, and Config Variables, click on reveal, and you should get the DATABASE_URL, the default variable is needed to /admin on django work...(This is only a work around, you should find a way to get the DATABASE_URL from an environment var)
- Modify your Dockerfile to run the command to put the django server up adding to the last line:
CMD python3 manage.py runserver 0.0.0.0:\$PORT
- The \$PORT variable is the application port provided by heroku, so your application can run
- Last but not least add your heroku app url to ALLOWED_HOSTS, as our friend Alejandro Sánchez said
- And you're golden, your application should be running at the heroku app url

EDIT 1:

You don't need gunicorn as a dependency, the only command you should have in your Procfile is:
web: python3 manage.py runserver

edited Sep 14 '17 at 22:23

answered Sep 14 '17 at 22:15



**Giovanni Klein
Campigoto**

190 1 12

thanks for this. i was in a similar pinch and found this very helpful. – [Ryan Skene](#) Feb 11 '18 at 23:25

- 1 Any chance you used Caddy as the server and can share the configs? Finding it difficult to run for production using heroku's default app domain vs caddy's requirements – [Casper](#) Mar 23 '18 at 1:55 ✎

did you find installing django-heroku to be necessary? The module itself seems to break for me. – [Casper](#) Mar 23 '18 at 14:03

No, I used docker, I can't answer about caddy, because I don't have any knowledge about it. – [Giovanni Klein Campigoto](#) Mar 23 '18 at 14:12 ✎

I was finally able to only push it thanks to your descriptions. But I'm having trouble with the CMD python3 ... line. Heroku logs say python3: can't open file 'manage.py': [Errno 2] No such file or directory . Any idea? I would've thought with CMD, I'd run gunicorn. I've tried that too, but the following "app.wsgi" "module can't be found".. – [Casper](#) Mar 23 '18 at 14:42

That's probably because you are not in the right django repository, because your manage.py was not found, you should try to find the django directory that contains manage.py file, and open its path with CD, before executing CMD python3 – [Giovanni Klein Campigoto](#) Mar 23 '18 at 19:14

Try adding the heroku url to ALLOWED_HOSTS

Reference

answered Sep 14 '17 at 19:53

[Alejandro Sánchez](#)

31 1

That actually was some part of it, see my answer bellow – [Giovanni Klein Campigoto](#) Sep 14 '17 at 22:16
