**Deploying Flask application to Heroku**

Heroku requires Procfile, which it uses to set up a web server.

1. From Git bash terminal:
2. source activate pet\_pals (conda create –n pet\_pals python=3.6)
3. pip install -r requirements.txt (install what you know you need, after running the program you can update this file.)
4. python initdb.py
5. sh run.sh (run.sh file contains FLASK\_APP=pet\_pals/app.py flask run, you can run this from the command prompt.)
6. Basically need to verify application is working locally
7. Update and correct requirements file pip freeze > requirements.txt or just edit in vsc or some other editor. (when compiling in Heroku there are several things that can cause it to kick out, be aware maybe all requirements not installed.)
8. We also must create a Procfile file
   1. Double check you are in the right environment then command is touch Procfile
   2. Use an editor to enter this command “web: gunicorn pet\_pals.app:app” pet\_pals is the directory your \_init\_.py file is in
9. Create git repo
10. Go to Git bash terminal and directory where our code is and verify that we are in peta\_pals environment
11. git init (this creates a Git repository in this folder)
12. git add . (adds all these files to the newly created Git repository)
13. git commit -am “Pet Pals created” (this finishes creating local Git repository)
14. Go to Github
15. Click + new repository (to create a new repository)
16. Type in repository name like pet\_pals, click on Public, don’t initialize with a README, and click on “Create repository.”
17. Copy the line on Github screen “git remote add origin [git@github.com:manlara/pet\_pals.git](mailto:git@github.com:manlara/pet_pals.git)”
18. Paste the line “git remote add origin [git@github.com:manlara/pet\_pals.git](mailto:git@github.com:manlara/pet_pals.git)” into our Git bash terminal and hit Enter in the Git bash terminal.
19. Copy the line on Github screen “git push -u origin master”
20. Paste the line “git push -u origin master” into our Git bash terminal and hit Enter in the Git bash terminal.
21. This should push code up to Github.
22. Click refresh on our Github page (<https://github.com/manlara/pet_pals>) and the pet\_pals project (and all files) should be visible there on Github.
23. Go to Heroku.com
24. Click New and create new app
25. Call App name something unique like manlara-pet-pals and click “Create app”
26. Then on the “Deploy” tab, click on “Connect to GitHub”
27. Next to our username manlara, type in repository name “pet\_pals” and press Enter.
28. You should see manlara/pet\_pals show up and then click the “Connect” button.
29. After you see “Connected to manlara/pet\_pals”, scroll down and click “Enable Automatic Deploys.”
30. Then click on “Deploy branch” button to deploy master branch.
31. In the “Receive code from Github” section, it should start downloading files from our requirements.txt and eventually should see “(this is also where you will begin to see errors if you have any,) Launching… Released v3 <https://manlara-pet-pals.herokuapp.com/> deployed to Heroku” that same section.
32. Now we need to connect this to a database. So in Heroku, go to the “Resources” tab
33. In Add-ons, type in postgres and select Heroku Postgres.
34. Click on Provision button
35. Need to initialize database before opening our application. In Heroku, go to right of screen and click on “More”
36. Click on Run console
37. Next to the heroku run part, type in “python initdb.py” and click the Run button, (another place errors may show up.)
38. Look for phrase at bottom of Console that says “Process exited.”
39. Then on Heroku, go to right of screen and click on “Open app”
40. This should be our application just like we ran locally on our laptop before.
41. You should be able to run everything properly from there (and the url for the app should be <https://manlara-pet-pals.herokuapp.com>)
42. That’s it!