Django Post-Build Steps (needed on-time only unless you start from scratch)

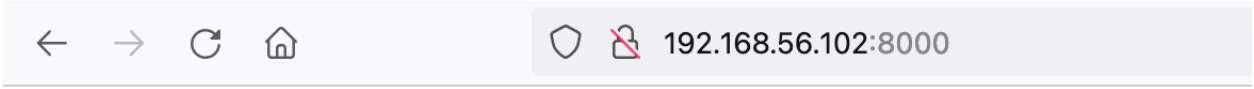
1. Cd to the db directory and run vagrant up
2. After the db server is up, cd to the ws1 directory and run vagrant up
3. After the ws1 server is up, run vagrant ssh
4. In the ws1 ssh session, run ./django-initialize.sh
5. The output of the previous command should look like:

```
vagrant@ubuntu-server:~$ ./django-initialize.sh

# Create Django tables in the db
cd /home/vagrant/website/
python3 manage.py migrate
Operations to perform:
  Apply all migrations: account, admin, auth, contenttypes, sessions, sites, socialaccount
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying account.0001_initial... OK
  Applying account.0002_email_max_length... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying sessions.0001_initial... OK
  Applying sites.0001_initial... OK
  Applying sites.0002_alter_domain_unique... OK
  Applying socialaccount.0001_initial... OK
  Applying socialaccount.0002_token_max_lengths... OK
  Applying socialaccount.0003_extra_data_default_dict... OK

# Create Django superuser
echo "from django.contrib.auth import get_user_model; User = get_user_model(); User.objects.create_superuser('vagrant', 'vagrant@iit.edu', '20
22-team01m')" | python3 manage.py shell
vagrant@ubuntu-server:~$
```

6. In the ws1 ssh session, run ./runserver.sh (you won't see any output)
7. Add the following line to your /etc/hosts file (for mac) or to your c:\Windows\System32\Drivers\etc\hosts file (for windows):
   `192.168.56.102 stackprj.com`
8. In a browser, go to http://stackprj.com:8000. You should see:



Food Question & Answer: Welcome Page

9. In the same browser window, go to http://stackprj.com:8000/admin. You should see a login page. For credentials, enter the user/password combination: vagrant/2022-team01m and press "Log in".
10. Next to Sites, click Add

11. For Domain name, type in stackprj.com
    For Display name, type in Stack Project
    Click Save.
12. In the upper click path, click Home

13. Next to Social applications, click Add

14. For Provider, select Google
    For Name, type Stack Google OAuth
    For Client id, type:
    336760590996-18um2itk2835mkq0vqgslomrs0rp17rr.apps.googleusercontent.com
    For App ID, or consumer key Secret key, type:
    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
    Under Key Sites, shift-select both example.com and stackprj.com
    Click Save
15. In the upper click path, click Log out
16. Go to http://stackprj.com:8000/login and click Login with Google
17. Click the Continue button
18. If you are already signed in via Gmail, you will be redirected back to the Welcome page. Otherwise, you will be prompted to login to Google and then redirected back to the Welcome page.