**Instructions after SSH into AWS machine**

1. sudo apt-get update
2. sudo apt-get install mysql-server python3-pip python3-dev libmysqlclient-dev ufw virtualenv docker.io nginx supervisor
3. pip3 install virtualenv gunicorn
4. sudo mysql\_secure\_installation  
     
   Root Password: P@$$w0rd  
   Remove anonymous user: Yes  
   Disallow root login: Yes  
   Remove test DB: Yes  
   Reload privilege tables: Yes
5. sudo mysql
   1. mysql> CREATE DATABASE hitme\_db;
   2. mysql> CREATE USER hitme\_user;
   3. mysql > GRANT ALL ON hitme\_db.\* TO 'hitme\_user'@'localhost' IDENTIFIED BY 'Mysql@123';
   4. mysql> ALTER DATABASE hitme\_db CHARACTER SET 'utf8';
   5. exit
6. cd ~
7. virtualenv --python=$(which python3) webenv
8. source webenv/bin/activate
9. Clone Github repo: git clone <https://github.com/csc667/csc667-su19-Team05.git>
10. cd csc667-su19-Team05/hitme/hitme
11. Create a local file called **local\_settings.py**. This file is used to store database credentials and allowed hostnames. The file should have the following content. Please replace hostnames with your AWS instance and also change DB credentials if your credentials are different
12. ALLOWED\_HOSTS = ['ec2-54-193-21-163.us-west-1.compute.amazonaws.com', '54.193.21.163']
13. DATABASE\_NAME = 'hitme\_db'
14. DATABASE\_USER = 'hitme\_user'
15. DATABASE\_PASSWORD = 'Mysql@123'
16. DATABASE\_HOST = 'localhost'
17. DATABASE\_PORT = ''
18. cd ~
19. cd csc667-su19-Team05/hitme
20. pip3 install -r requirements.txt
21. python manage.py makemigrations
22. python manage.py migrate
23. python manage.py collectstatic
24. sudo systemctl start docker
25. sudo systemctl enable docker
26. sudo docker run -p 6379:6379 -d redis:2.8
27. Run the server: python manage.py runserver 0.0.0.0:5000

**Instructions to set up Nginx + Daphne**

1. mkdir /run/daphne
2. sudo nano /etc/supervisor/conf.d/hitme.conf
3. [fcgi-program:asgi]
4. # TCP socket used by Nginx backend upstream
5. socket=tcp://localhost:8000
6. # Directory where your site's project files are located
7. directory=/home/ubuntu/csc667-su19-Team05/hitme/
8. # Each process needs to have a separate socket file, so we use process\_num
9. # Make sure to update "mysite.asgi" to match your project name
10. command=/home/ubuntu/webenv/bin/daphne -u /run/daphne/daphne%(process\_num)d.sock --endpoint fd:fileno=0 --access-log - --proxy-headers hitme.asgi:application
11. # Number of processes to startup, roughly the number of CPUs you have
12. numprocs=4
13. # Give each process a unique name so they can be told apart
14. process\_name=asgi%(process\_num)d
15. # Automatically start and recover processes
16. autostart=true
17. autorestart=true
18. # Choose where you want your log to go
19. stdout\_logfile=/home/ubuntu/asgi.log
20. redirect\_stderr=true
21. sudo supervisorctl reread
22. sudo supervisorctl update
23. sudo nano /etc/nginx/sites-available/hitme
24. upstream channels-backend {
25. server localhost:8000;
26. }
27. server {
28. listen 80;
29. server\_name ec2-54-193-101-52.us-west-1.compute.amazonaws.com;
30. location = /favicon.ico {access\_log off;log\_not\_found off;}
31. location /static/ {
32. root /home/ubuntu/csc667-su19-Team05/hitme;
33. }
34. location /media/ {
35. root /home/ubuntu/csc667-su19-Team05/hitme;
36. }
37. location / {
38. try\_files $uri @proxy\_to\_app;
39. }
40. location @proxy\_to\_app {
41. proxy\_pass http://channels-backend;
42. proxy\_http\_version 1.1;
43. proxy\_set\_header Upgrade $http\_upgrade;
44. proxy\_set\_header Connection "upgrade";
45. proxy\_redirect off;
46. proxy\_set\_header Host $host;
47. proxy\_set\_header X-Real-IP $remote\_addr;
48. proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;
49. proxy\_set\_header X-Forwarded-Host $server\_name;
50. }
51. }
52. sudo ln -s /etc/nginx/sites-available/hitme /etc/nginx/sites-enabled/hitme
53. sudo nano /etc/nginx/nginx.conf : **uncomment server\_name\_hash\_bucket\_size 64 and change it to server\_name\_hash\_bucket\_size 512**
54. sudo nginx -t
55. sudo ufw delete allow 5000
56. sudo ufw allow 'Nginx Full'
57. sudo supervisorctl restart all
58. sudo systemctl restart nginx