ML Web Application:

* Launch a Ubuntu Server 16.04 LTS (HVM), SSD Volume Type instance.
* Attach an IAM role with the appropriate permissions that your web app will use. This can be updated later if needed.
* Create a .pem file and save it. This will be how you ssh into your instance for ongoing maintenance.
* SSH into your newly launched instance and perform the bellow commands:
  + sudo apt-get update
  + sudo apt-get upgrade
  + sudo timedatectl set-timezone UTC
  + sudo apt-get install apache2
  + sudo apt-get install libapache2-mod-wsgi
  + sudo apt-get install postgresql postgresql-contrib
  + sudo -u postgres createuser -P catalog
  + sudo -u postgres createdb -O catalog catalog
    - this will require a password so I chose ‘password’
  + sudo apt-get install python-sqlalchemy python-pip
  + sudo apt-get install python-psycopg2 python-flask
  + sudo apt-get install python-pandas
  + sudo apt-get install python-boto3
  + sudo apt-get install git
  + cd /var/www/
  + sudo mkdir mlwebapp
  + sudo chown www-data:www-data mlwebapp/
  + sudo -u www-data git clone https://github.com/zacktwp/MLWebApp.git mlwebapp
  + cd mlwebapp
    - replace the files that have engine = create\_engine('sqlite:///polution.db') with engine = create\_engine('postgresql://catalog:password@localhost/catalog')
  + sudo mv webapp.py \_\_init\_\_.py
  + cd ..
    - this should be in the bring you back to the /var/www/ directory
  + sudo nano flaskapp.wsgi
    - #copy the below code and save it in this file

import sys

import logging

logging.basicConfig(stream=sys.stderr)

sys.path.insert(0,"/var/www/")

from mlwebapp import app as application

application.secret\_key = 'Add your secret key'

* sudo nano /etc/apache2/sites-available/FlaskApp.conf
  + #copy VirtualHost block below and replace ServerName and ServerAdmin with your ec2 ip address past in the FlaskApp.conf file

<VirtualHost \*:80>

ServerName mywebsite.com

ServerAdmin admin@mywebsite.com

WSGIScriptAlias / /var/www/flaskapp.wsgi

<Directory /var/www/mlwebapp/>

Order allow,deny

Allow from all

</Directory>

Alias /static /var/www/mlwebapp/static

<Directory /var/www/mlwebapp/static/>

Order allow,deny

Allow from all

</Directory>

ErrorLog ${APACHE\_LOG\_DIR}/error.log

LogLevel warn

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

* sudo a2dissite 000-default.conf
* sudo a2ensite FlaskApp
* sudo service apache2 restart
* sudo pip install awscli –upgrade
* sudo pip install boto3 –upgrade
* sudo service apache2 restart

Congratulations, you’ve just set up an apache linux server that runs your flask web app. You can now start interacting with your web application by typing in the public ip address of the server instance in your browser.

If you encounter an error, feel free to ssh back into your server instance and type:

sudo nano /var/log/apache2/error.log

to error track.