**Install Anaconda:**

1. Install Anaconda: <https://www.continuum.io/downloads>
2. Create a conda environment
   1. conda create –newEnvironment opencv-env python=(version number)
3. Activate the conda environment
   1. activate newEnvironment
4. Install the necessary dependencies
   1. pip install dependency
5. If pip can’t install a dependency, use conda to install
   1. conda install -c conda-forge av

**Set Up Web Interface:**

1. Install Python Flask using these instructions:
   1. https://linuxize.com/post/how-to-install-flask-on-ubuntu-18-04/Create a conda environment
2. Place the CSS and JS files in a folder named static, and html pages in a folder name template
   1. (View github repo for files)

**Set Up DSE Cluster on AWS EC2 Instance:**

1. Follow this tutorial to set up a 3-node cluster:
   1. https://academy.datastax.com/content/hands-tutorial-install-dse-aws-cloud-3-easy-steps
2. Connect to cqlsh using private ip and username and password
   1. cqlsh “privateIP” -u username -p password
3. Run this command to set up the keyspace in one instance (will replicate to all)

CREATE KEYSPACE competition WITH replication = {'class':'SimpleStrategy', 'replication\_factor':3};

USE competition;

CREATE TABLE positional (flight\_id TIMEUUID, ts TIMESTAMP, x DOUBLE, y DOUBLE, z DOUBLE, latest\_ts TIMESTAMP STATIC, station\_id UUID STATIC, num\_crashes INT STATIC, name TEXT STATIC, group TEXT STATIC, org\_college TEXT STATIC, major TEXT STATIC, valid BOOLEAN STATIC, PRIMARY KEY (flight\_id,ts)) WITH CLUSTERING ORDER BY (ts DESC);

1. To re set up after turning off instances, just adjust the node IPs and reinstall cluster. Keyspace will still exist.

**Open AWS Cluster to Remote Access:**

1. SSH into the node
2. Navigate to /etc directory (run the following command)

cd /etc

1. Create a file for the script

sudo touch setIP

1. Edit the new file

vi setIP

1. Paste the following script into the file (press “i” to go into insert mode)

#!/bin/bash

ip="$(sudo curl v4.ifconfig.co)"

grep -q "RPC\_ADDRESS: " /etc/dse/cassandra/cassandra.yaml &&

sudo sed -i s/"RPC\_ADDRESS: .\*"/"RPC\_ADDRESS: ${ip}"/ /etc/dse/cassandra/cassandra.yaml ||

sudo echo "RPC\_ADDRESS: ${ip}" >> /etc/dse/cassandra/cassandra.yaml

1. Press “Esc” to get out of insert mode
2. Enter the following command to save a read only file

:w !sudo tee %

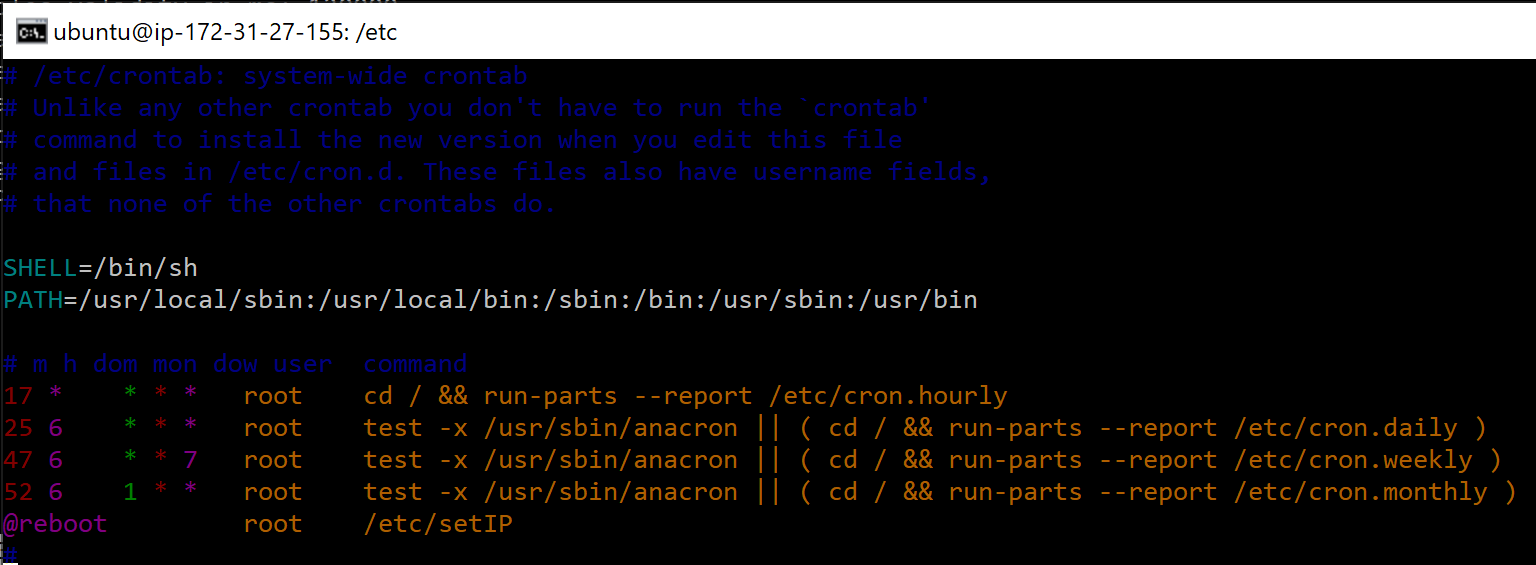
1. Press “O”
2. Press “Enter”
3. Enter the following command

:q!

1. Make the script executable

sudo chmod +x setIP

1. Edit crontab to make the script run on reboot…. For each node:
   1. vi crontab
   2. Press “i”
   3. @reboot root /etc/setIP



* 1. Press “Esc”
  2. :w !sudo tee %
  3. Press “O”
  4. Press “Enter”
  5. :q!

1. Reboot your instance

**Connect to AWS Cluster From Python Scrpt:**

1. Import the necessary packages
   1. from dse.cluster import Cluster
   2. from dse.auth import PlainTextAuthProvider
2. Create an auth\_provider with cluster credentials
   1. auth\_provider = PlainTextAuthProvider(username=”user”, password=”password”)
3. Create a cluster object using IP’s and auth\_provider
   1. cluster=Cluster(auth\_provider=auth\_provider, contact\_points=[‘ip1’, ‘ip2’, ‘ip3’])
4. Connect to the cluster

try:

session=cluster.connect(‘competition’)

**Connect to Drone and Cluster at the Same Time**

1. Connect laptop wifi to drone
2. Plug in wifi adapter dongle
3. Computer should then connect to wifi
4. Start Program
   1. Computer should be able to connect to both at the same time