1. Go to Sandbox (AWS Dashboard, link provided by the professor), Start AWS Lab, Click on Launch Instance

A screenshot of a computer

Description automatically generated

1. Select Ubuntu and AMI version 20.04 as shown in below screenshot.

A screenshot of a computer

Description automatically generated

1. Don’t change the Architecture & Instance type, keep it as it is as per the below screenshot.

A screenshot of a computer

Description automatically generated

1. Keep everything same and only change the configure storage to 28 and click on Launch Instance.

A screenshot of a computer

Description automatically generated

1. Click on (i-06b8cf85d5b8a9b84)

A screenshot of a computer

Description automatically generated

1. Check mark Oct3AME (Instance Name) and go to Security.

A screenshot of a computer

Description automatically generated

1. Then Click on Inbound Rules and select Edit Inbound Rules.

A screenshot of a computer

Description automatically generated

1. Modify the Inbound Rules as per below screenshot.

A screenshot of a computer

Description automatically generated

1. Go back to the previous page and click on Connect.

A screenshot of a computer

Description automatically generated

1. Once this window pops up, Click on Connect.

A screenshot of a computer

Description automatically generated

1. Once the EC2 instance is open, Go to Professors repository and click on the link (Install Node/Mongo on EC2)

A screenshot of a computer

Description automatically generated

1. Copy the “Curl | bash” and paste in the EC2 command line.

A screenshot of a computer

Description automatically generated

1. Paste the entire “Curl -s … | bash” here in the Instance command line as shown below.

A screenshot of a computer

Description automatically generated

1. Once the installation is complete it will show like this.

A screenshot of a computer

Description automatically generated

1. Once the installation is completed run this command: **sudo nano /etc/mongod.conf** (Change the bind IP to 0.0.0.0, Port should be 27017)
2. After that we run this command: **sudo service mongod start** to start the mongod.
3. The run this code: **sudo service mongod status** to check the status. It should show actively running if the service is started successfully.
4. After this we will have to install mongo skin and mongobd using following commands.
   1. **sudo npm install mongoskin@2**
   2. **sudo npm install** [**mongodb@2.2**](mailto:mongodb@2.2)
5. Once both of the above are installed then we will run the server using this command: **node serverwithdb.js**
6. Make sure that the port on EC2 and in the Arduino code is same. If not then change it accordingly. Below I changed the port from 1234 to 8080 using this command: **sudo nano serverwithdb.js** (Changed the port to 8080 & the bind IP to 0.0.0.0 if needed)

A screenshot of a computer program

Description automatically generated

1. Go to new tab in your browser and type [**http://18.212.152.139:8080/index.html**](http://18.212.152.139:8080/index.html) to open below window.

A screenshot of a computer

Description automatically generated

1. Make sure you use the same IP address in the MongoDB compass as well and connect.

A screenshot of a computer

Description automatically generated

1. Click on refresh and you will see sensorData, once you click on it you will be able to see the data.

A screenshot of a computer

Description automatically generated

1. This is what you need: **HTTP Response code: 200**. If you don’t get this then you have made some mistake and wont be able to open index.html file in the browser.

A screenshot of a computer program

Description automatically generated

1. Below are the commands that I followed for this Assignment.

* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ sudo nano /etc/mongod.conf (Change the bind IP to 0.0.0.0, Port should be 27017)
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ ls
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ sudo service mongod start
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ sudo service mongod status
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ node serverwithdb.js
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ sudo npm install mongoskin@2
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ node serverwithdb.js
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ sudo npm install [mongodb@2.2](mailto:mongodb@2.2)
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ node serverwithdb.js
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ sudo nano serverwithdb.js (Changed the port & the bind IP to 0.0.0.0 if needed)
* ubuntu@ip-172-31-37-153:~/ame498-598Fall2023/watchSensors/captureDataServer-TempHum$ node serverwithdb.js