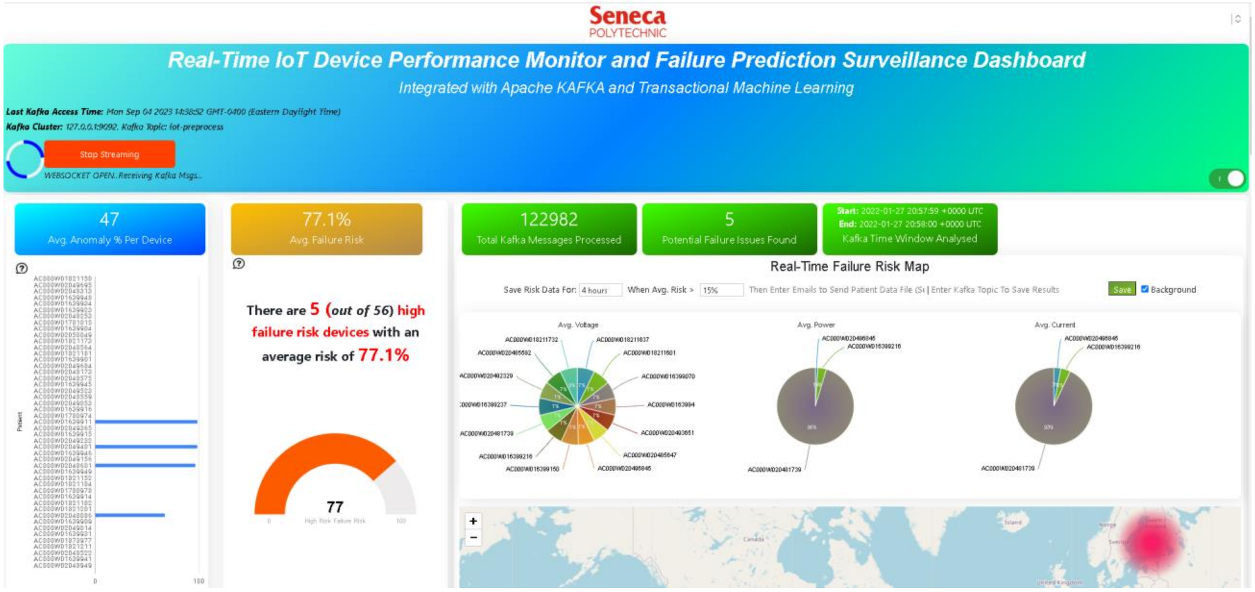
CSP 450 Fall 2023

PROJECT 5: Kubernetes, Docker, Python and TML Real-Time Dashboard Modifications

A. Make Modifications to TML Solution and Dashboard, Rebuild the Container, and Discuss Your New Solution

1. **Run your TML Dashboard:**
   a. **Open a browser i.e. Google Chrome**
   b. **Enter URL:** http://localhost:9005/iot-failure-seneca.html?topic=iot-preprocess2,iot-preprocess&offset=-1&groupid=&rollbackoffset=500&topictype=prediction&append=0&secure=1
2. **You should see:**



Seneca POLYTECHNIC

Real-Time IoT Device Performance Monitor and Failure Prediction Surveillance Dashboard

Integrated with Apache KAFKA and Transactional Machine Learning

Last Kafka Access Time: Mon Sep 04 2023 14:38:52 GMT-0400 (Eastern Daylight Time)

Kafka Cluster: 127.0.0.1:9092, Kafka Topic: iot-preprocess

WEBSOCKET OPEN..Receiving Kafka Msgs..

47 Avg. Anomaly % Per Device

77.1% Avg. Failure Risk

122982 Total Kafka Messages Processed

5 Potential Failure Issues Found

Start: 2022-01-27 20:57:59 +0000 UTC **End:** 2022-01-27 20:58:00 +0000 UTC Kafka Time Window Analysed

There are 5 (*out of 56*) high failure risk devices with an average risk of 77.1%

Real-Time Failure Risk Map

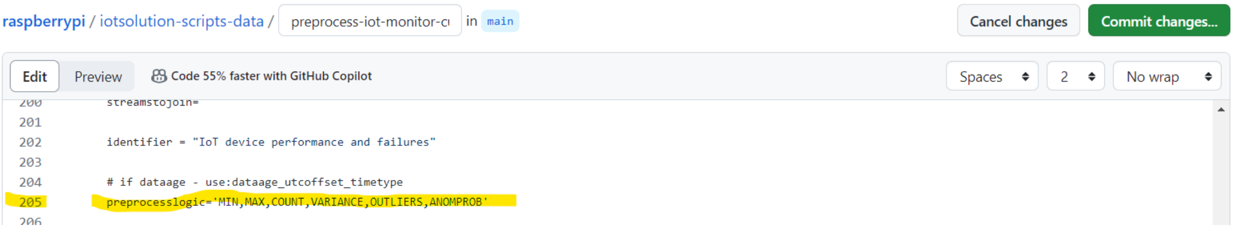
Save Risk Data For: 4 hours When Avg. Risk > 15% Then Enter Emails to Send Patient Data File (S| Enter Kafka Topic To Save Results

3. **Go to your raspberrypi repo**

   a. Open Python Script: **iotsolution-scripts-data/preprocess-iot-monitor-customdata.py** file

   b. Modify the preprocess types on **line 205 with the following:** MIN, MAX, COUNT, VARIANCE, OUTLIERS, ANOMPROB (shown below – then **Commit Change in Github**): For example:

      i. **preprocesslogic='MIN,MAX,COUNT,VARIANCE,OUTLIERS,ANOMPROB'**


raspberrypi / iotsolution-scripts-data / preprocess-iot-monitor-c
in main
Cancel changes
Commit changes...
Edit
Preview
Code 55% faster with GitHub Copilot
Spaces
2
No wrap

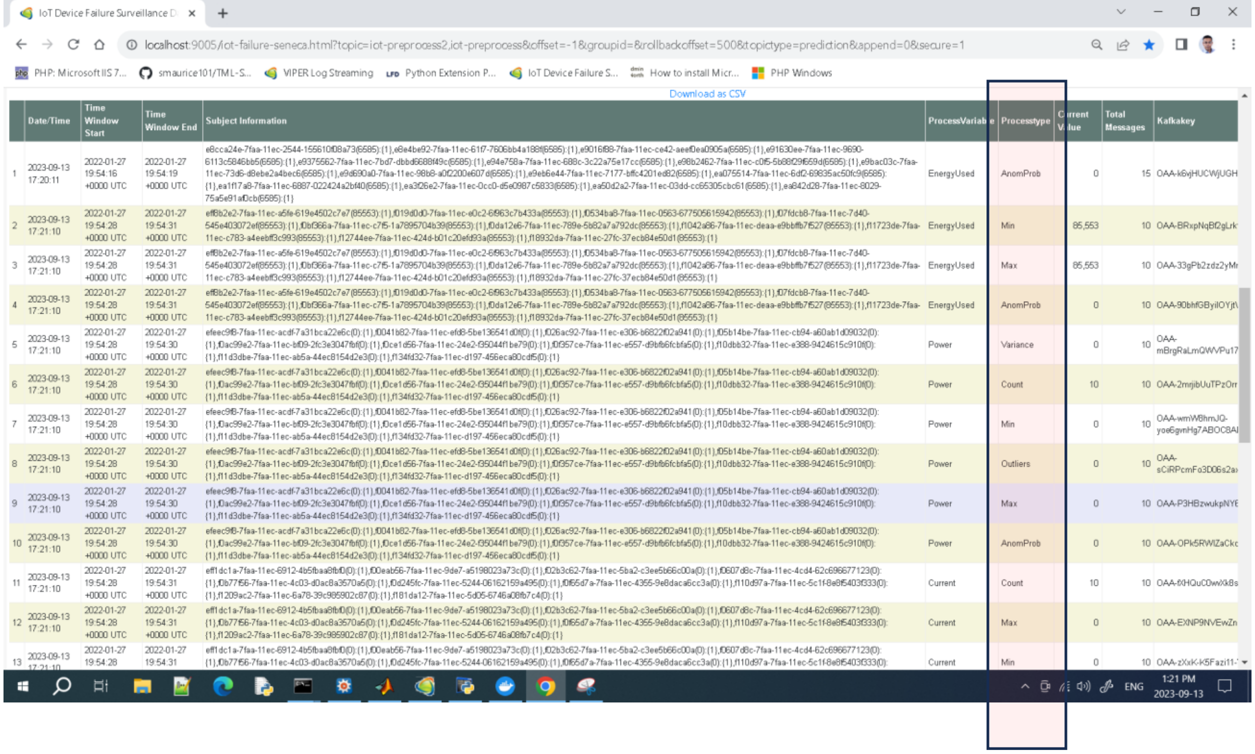
```
    streamstojoin=

    identifier = "IoT device performance and failures"

    # if dataage - use:dataage_utcoffset_timetype
    preprocesslogic='MIN,MAX,COUNT,VARIANCE,OUTLIERS,ANOMPROB'

```

4. **Create a new container and append "_modification" to the name of your container:**
   a. **Build your TML docker container:**
   a. NORMAL BUILD: docker build -t <your dockerhub id>/<container name>_modification --build-arg CHIP=AMD64 --no-cache --network=host .

   (**NOTE:** The '--no-cache' argument - this will build your container FROM SCRATCH (as you are doing now, and **takes about 20 minutes**)

5. **Push your container to Dockerhub:** docker push <your dockerhub id>/<container name>_modification

6. **Run your TML Dashboard and Show the Preprocess types in the TML Dashboard table:** MIN, MAX, COUNT, VARIANCE, OUTLIERS, ANOMPROB. For example:


IoT Device Failure Surveillance D
localhost:9005/iot-failure-seneca.html?topic=iot-preprocess2,iot-preprocess&offset=-1&groupid=&rollbackoffset=500&topictype=prediction&append=0&secure=1
PHP: Microsoft IIS 7...
smaurice101/TML-S...
VIPER Log Streaming
Python Extension P...
IoT Device Failure S...
How to install Micr...
PHP Windows
Download as CSV
ENG
1:21 PM
2023-09-13

	Date/Time	Time Window Start	Time Window End	Subject Information	ProcessVariable	Processtype	Current Value	Total Messages	Kafkakey
1	2023-09-13 17:20:11	2022-01-27 19:54:16 +0000 UTC	2022-01-27 19:54:19 +0000 UTC	e8cca24e-7faa-11ec-2544-155610f08a73(6585):{1},e8e4be92-7faa-11ec-61f7-7606bb4a188f(6585):{1},e9016f88-7faa-11ec-ce42-aeef0ea0905a(6585):{1},e91630ee-7faa-11ec-9690-6113c5846bb5(6585):{1},e9375562-7faa-11ec-7bd7-dbbd6688f49c(6585):{1},e94e758a-7faa-11ec-688c-3c22a75e17cc(6585):{1},e98b2462-7faa-11ec-c0f5-5b88f29f659d(6585):{1},e9bac03c-7faa-11ec-73d6-d8ebe2a4bec6(6585):{1},e9d690a0-7faa-11ec-98b8-a0f2200e607d(6585):{1},e9eb6e44-7faa-11ec-7177-bffc4201ed82(6585):{1},ea075514-7faa-11ec-6df2-69835ac50fc9(6585):{1},ea1f17a8-7faa-11ec-6887-022424a2bf40(6585):{1},ea3f26e2-7faa-11ec-0cc0-d5e0987c5833(6585):{1},ea50d2a2-7faa-11ec-03dd-cc65305cbc61(6585):{1},ea842d28-7faa-11ec-8029-75a5e91af0cb(6585):{1}	EnergyUsed	AnomProb	0	15	OAA-k6vjHUCWjUGH
2	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	Min	85,553	10	OAA-BRxpNqBf2gLrk
3	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	Max	85,553	10	OAA-33gPb2zdz2yMr
4	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	AnomProb	0	10	OAA-90bhfGByilOYjt
5	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Variance	0	10	OAA-mBrgRaLmQWVPu17
6	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Count	10	10	OAA-2mrjibUuTPzOrr
7	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Min	0	10	OAA-wmW8hmJQ-yoe6gvnHg7ABOC8AI
8	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Outliers	0	10	OAA-sCiRPcmFo3D06s2ax
9	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Max	0	10	OAA-P3HBzwukpNYE
10	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	AnomProb	0	10	OAA-OPk5RWlZaCkc
11	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):{1},f1209ac2-7faa-11ec-6a78-39c985902c87(0):{1},f181da12-7faa-11ec-5d05-6746a08fb7c4(0):{1}	Current	Count	10	10	OAA-fXHQuC0wXk8s
12	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):{1},f1209ac2-7faa-11ec-6a78-39c985902c87(0):{1},f181da12-7faa-11ec-5d05-6746a08fb7c4(0):{1}	Current	Max	0	10	OAA-EXNP9NVEwZn
13	2023-09-13 17:21:10	2022-01-27 19:54:28	2022-01-27 19:54:31	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):	Current	Min	0	10	OAA-zXxK-K5Fazi11-

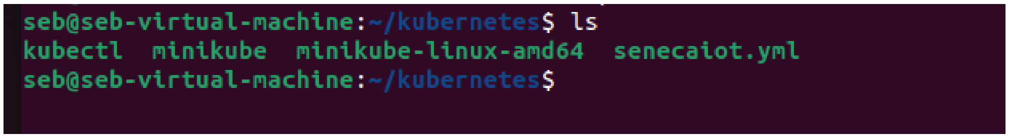
7. **Explain what the preprocessing is doing to the data; explain the importance of data preprocessing.**

B. Kubernetes: Run Docker container in Kubernetes

8. **Create a folder in your VM called kubernetes**
    a. **Note minikube is a ONE node Kubernetes cluster – it is the SAME functionality as a production grade Kubernetes cluster**
9. **cd to kubernetes folder**
10. **Go to: https://github.com/smaurice101/raspberrypi/blob/main/kubernetes/yaml**
    a. **Replace LINE 18 with your Docker image: image:** <your dockerhub id>/<container name>_modification
        i. **Commit the change**
    b. **Download yaml file to your kubernetes folder and rename it: senecaiot.yml**
11. **Now install Kubernetes and Kubectl:**
    a. **RUN:** `wget https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64`
    b. **RUN: sudo install minikube-linux-amd64 minikube**

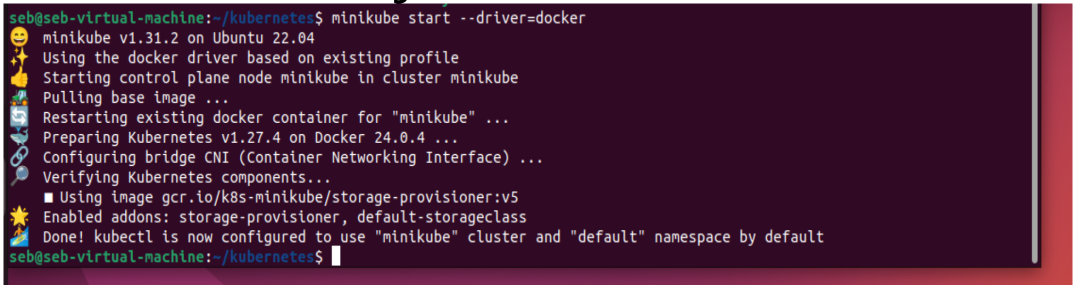
12. **Download: kubectl**
    a. `curl -LO https://storage.googleapis.com/kubernetes-release/release/`curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt`/bin/linux/amd64/kubectl`
    b. **RUN**: `sudo chmod +x kubectl`
    c. **RUN**: `sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl`

13. **Your kubernetes folder should look like:**



```
seb@seb-virtual-machine:~/kubernetes$ ls
kubectl  minikube  minikube-linux-amd64  senecaiot.yml
seb@seb-virtual-machine:~/kubernetes$
```

    a. **RUN Kubernetes: minikube start --driver=docker**
    b. `make sure docker engine is installed. If not run: sudo apt-get install docker.io`
    c. **RUN**: `sudo chmod 666 /var/run/docker.sock`
    d. **You should see a similar image**:

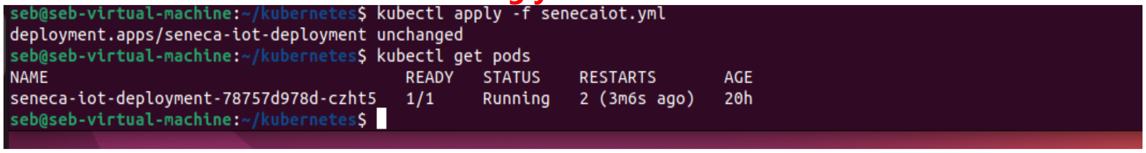


```
seb@seb-virtual-machine:~/kubernetes$ minikube start --driver=docker
😄  minikube v1.31.2 on Ubuntu 22.04
✨  Using the docker driver based on existing profile
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
🔄  Restarting existing docker container for "minikube" ...
🐳  Preparing Kubernetes v1.27.4 on Docker 24.0.4 ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: storage-provisioner, default-storageclass
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
seb@seb-virtual-machine:~/kubernetes$ 
```

    e.

14. **Create POD inside Kubernetes running your Docker Container**
    a. **RUN: kubectl apply -f senecaiot.yml**
    b. **RUN: kubectl get pods**

c. **You should see – You are now running your container inside Kubernetes:**

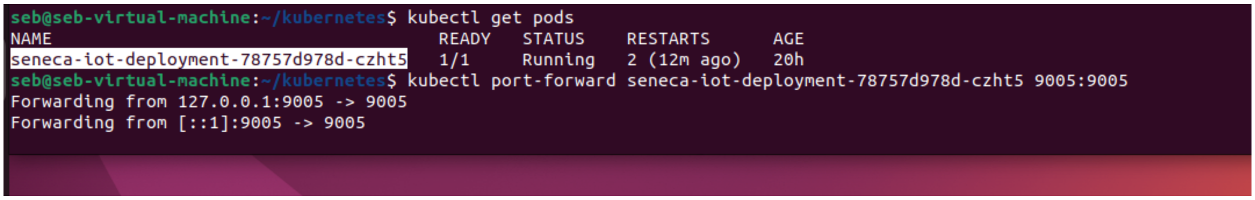


```
seb@seb-virtual-machine:~/kubernetes$ kubectl apply -f senecaiot.yml
deployment.apps/seneca-iot-deployment unchanged
seb@seb-virtual-machine:~/kubernetes$ kubectl get pods
NAME                                     READY   STATUS    RESTARTS       AGE
seneca-iot-deployment-78757d978d-czht5   1/1     Running   2 (3m6s ago)   20h
seb@seb-virtual-machine:~/kubernetes$ 
```

15. PORT Forward 9005:

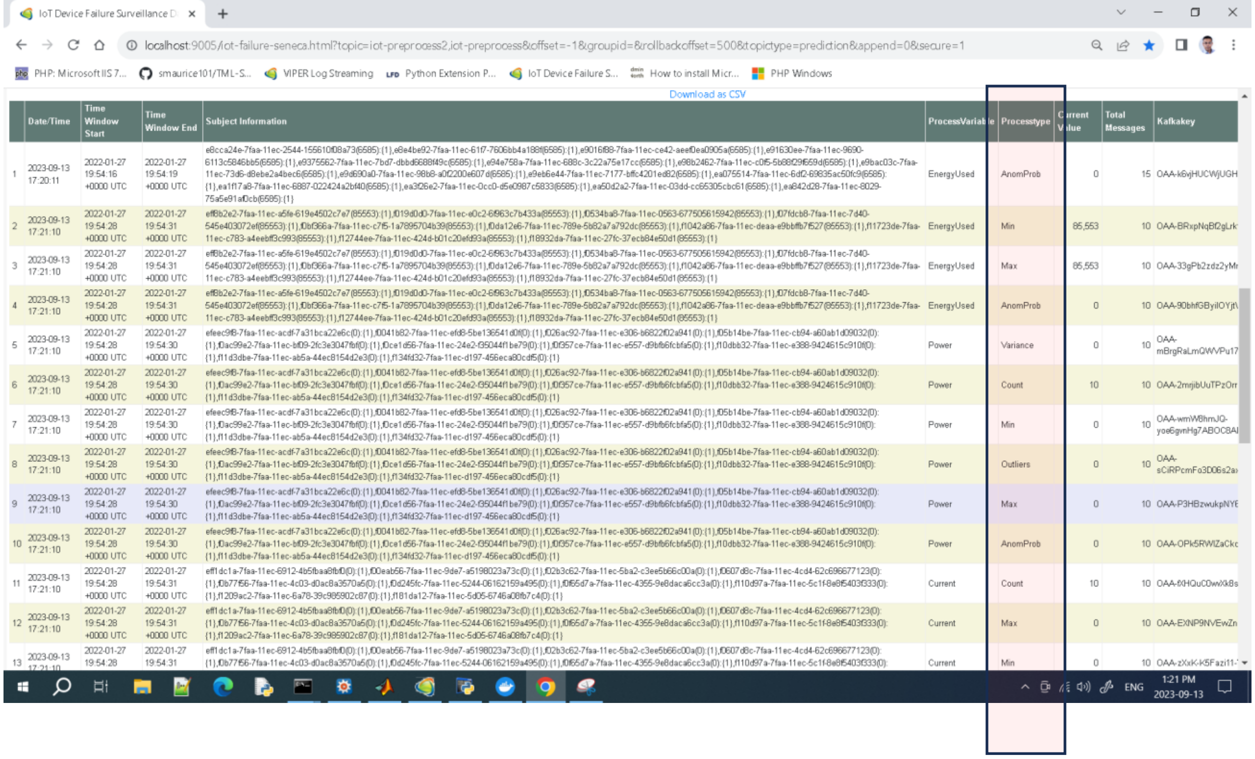
a. **RUN: kubectl port-forward <pod name> 9005:9005**

i. **For example: kubectl port-forward seneca-iot-deployment-78757d978d-czht5 9005:9005**



```
seb@seb-virtual-machine:~/kubernetes$ kubectl get pods
NAME                                     READY   STATUS    RESTARTS      AGE
seneca-iot-deployment-78757d978d-czht5   1/1     Running   2 (12m ago)   20h
seb@seb-virtual-machine:~/kubernetes$ kubectl port-forward seneca-iot-deployment-78757d978d-czht5 9005:9005
Forwarding from 127.0.0.1:9005 -> 9005
Forwarding from [::1]:9005 -> 9005
```

b. **RUN TML Dashboard: SEE SECTION A STEP 1 and take a screenshot as below:**

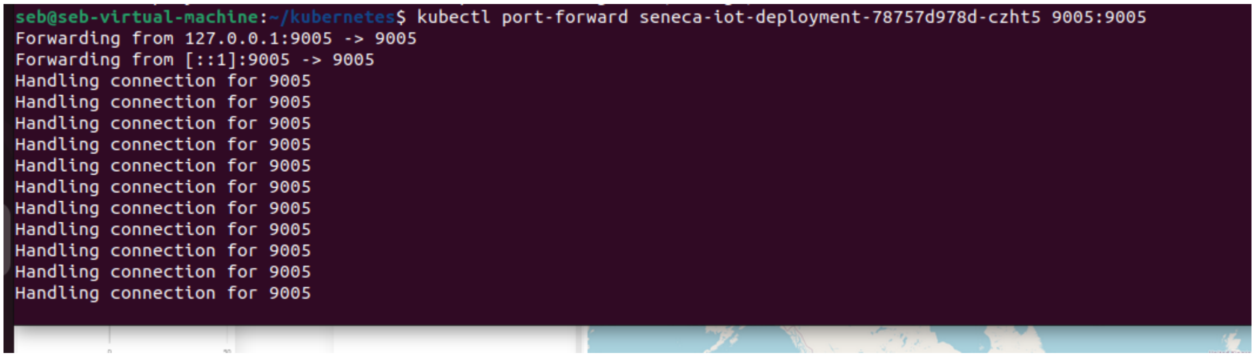


IoT Device Failure Surveillance D
localhost:9005/iot-failure-seneca.html?topic=iot-preprocess2,iot-preprocess&offset=-1&groupid=&rollbackoffset=500&topictype=prediction&append=0&secure=1
PHP: Microsoft IIS 7...
smaurice101/TML-S...
VIPER Log Streaming
Python Extension P...
IoT Device Failure S...
How to install Micr...
PHP Windows
Download as CSV
ENG
1:21 PM
2023-09-13

	Date/Time	Time Window Start	Time Window End	Subject Information	ProcessVariable	Processtype	Current Value	Total Messages	Kafkakey
1	2023-09-13 17:20:11	2022-01-27 19:54:16 +0000 UTC	2022-01-27 19:54:19 +0000 UTC	e8cca24e-7faa-11ec-2544-155610f08a73(6585):{1},e8e4be92-7faa-11ec-61f7-7606bb4a188f(6585):{1},e9016f88-7faa-11ec-ce42-aeef0ea0905a(6585):{1},e91630ee-7faa-11ec-9690-6113c5846bb5(6585):{1},e9375562-7faa-11ec-7bd7-dbbd6688f49c(6585):{1},e94e758a-7faa-11ec-688c-3c22a75e17cc(6585):{1},e98b2462-7faa-11ec-c0f5-5b88f29f659d(6585):{1},e9bac03c-7faa-11ec-73d6-d8ebe2a4bec6(6585):{1},e9d690a0-7faa-11ec-98b8-a0f2200e607d(6585):{1},e9eb6e44-7faa-11ec-7177-bffc4201ed82(6585):{1},ea075514-7faa-11ec-6df2-69835ac50fc9(6585):{1},ea1f17a8-7faa-11ec-6887-022424a2bf40(6585):{1},ea3f26e2-7faa-11ec-0cc0-d5e0987c5833(6585):{1},ea50d2a2-7faa-11ec-03dd-cc65305cbc61(6585):{1},ea842d28-7faa-11ec-8029-75a5e91af0cb(6585):{1}	EnergyUsed	AnomProb	0	15	OAA-k6vjHUCWjUGH
2	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	Min	85,553	10	OAA-BRxpNqBf2gLrk
3	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	Max	85,553	10	OAA-33gPb2zdz2yMr
4	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	AnomProb	0	10	OAA-90bhfGByilOYjt\
5	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Variance	0	10	OAA-mBrgRaLmQWVPu17
6	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Count	10	10	OAA-2mrjibUuTPzOrr
7	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Min	0	10	OAA-wmW8hmJQ-yoe6gvnHg7ABOC8AI
8	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Outliers	0	10	OAA-sCiRPcmFo3D06s2ax
9	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Max	0	10	OAA-P3HBzwukpNYE
10	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	AnomProb	0	10	OAA-OPk5RWIZaCkc
11	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):{1},f1209ac2-7faa-11ec-6a78-39c985902c87(0):{1},f181da12-7faa-11ec-5d05-6746a08fb7c4(0):{1}	Current	Count	10	10	OAA-fXHQuC0wXk8s
12	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):{1},f1209ac2-7faa-11ec-6a78-39c985902c87(0):{1},f181da12-7faa-11ec-5d05-6746a08fb7c4(0):{1}	Current	Max	0	10	OAA-EXNP9NVEwZn
13	2023-09-13 17:21:10	2022-01-27 19:54:28	2022-01-27 19:54:31	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):	Current	Min	0	10	OAA-zXxK-K5Fazi11-

16. Go to Port Forward terminal:

   a. You should see this:



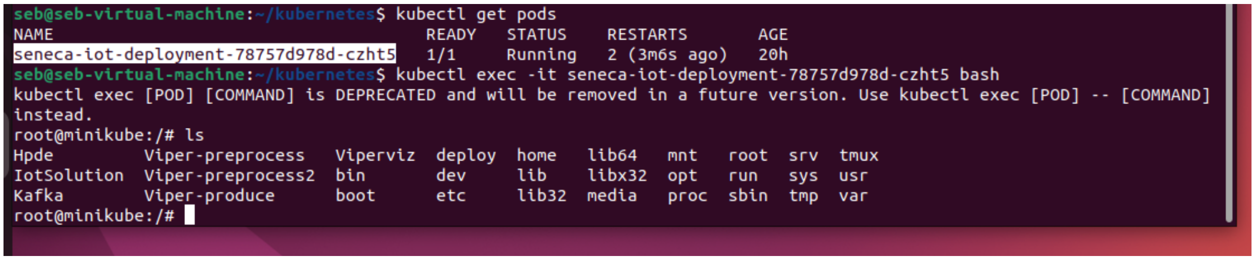
```
seb@seb-virtual-machine:~/kubernetes$ kubectl port-forward seneca-iot-deployment-78757d978d-czht5 9005:9005
Forwarding from 127.0.0.1:9005 -> 9005
Forwarding from [::1]:9005 -> 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
Handling connection for 9005
```

17. Open new Terminal: Go inside the running container:

   a. RUN: kubectl exec -it <NAME> bash

      i. For example: kubectl exec -it seneca-iot-deployment-78757d978d-czht5 bash

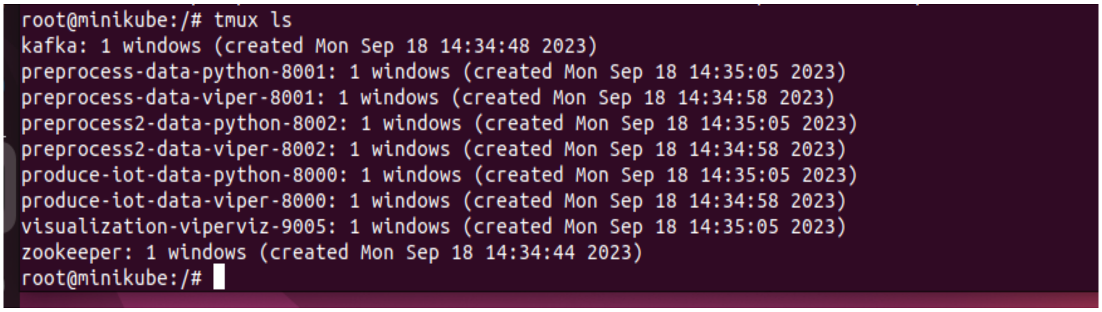
      ii. You should see similar screen:



```
seb@seb-virtual-machine:~/kubernetes$ kubectl get pods
NAME                                     READY   STATUS    RESTARTS       AGE
seneca-iot-deployment-78757d978d-czht5   1/1     Running   2 (3m6s ago)   20h
seb@seb-virtual-machine:~/kubernetes$ kubectl exec -it seneca-iot-deployment-78757d978d-czht5 bash
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND]
instead.
root@minikube:/# ls
Hpde         Viper-preprocess   Viperviz  deploy  home   lib64   mnt   root  srv  tmux
IotSolution  Viper-preprocess2  bin       dev     lib    libx32  opt   run   sys  usr
Kafka        Viper-produce      boot      etc     lib32  media   proc  sbin  tmp  var
root@minikube:/# 
```

   b. RUN: tmux ls

   c. You should see:

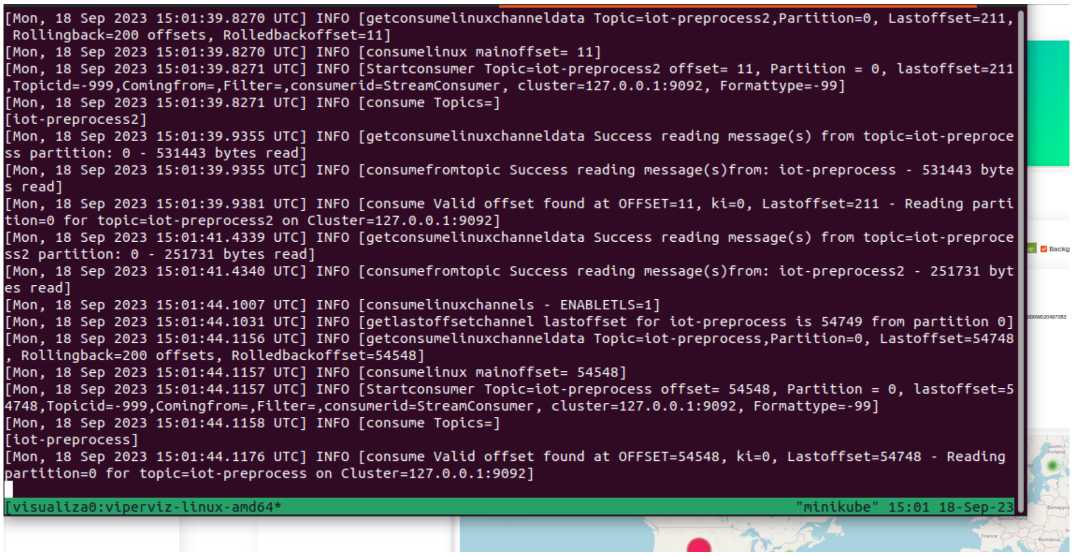


```
root@minikube:/# tmux ls
kafka: 1 windows (created Mon Sep 18 14:34:48 2023)
preprocess-data-python-8001: 1 windows (created Mon Sep 18 14:35:05 2023)
preprocess-data-viper-8001: 1 windows (created Mon Sep 18 14:34:58 2023)
preprocess2-data-python-8002: 1 windows (created Mon Sep 18 14:35:05 2023)
preprocess2-data-viper-8002: 1 windows (created Mon Sep 18 14:34:58 2023)
produce-iot-data-python-8000: 1 windows (created Mon Sep 18 14:35:05 2023)
produce-iot-data-viper-8000: 1 windows (created Mon Sep 18 14:34:58 2023)
visualization-viperviz-9005: 1 windows (created Mon Sep 18 14:35:05 2023)
zookeeper: 1 windows (created Mon Sep 18 14:34:44 2023)
root@minikube:/# 
```

   d. TMUX into visualization window:

      i. RUN: tmux a -t visualization-viperviz-9005

      ii. You should see:

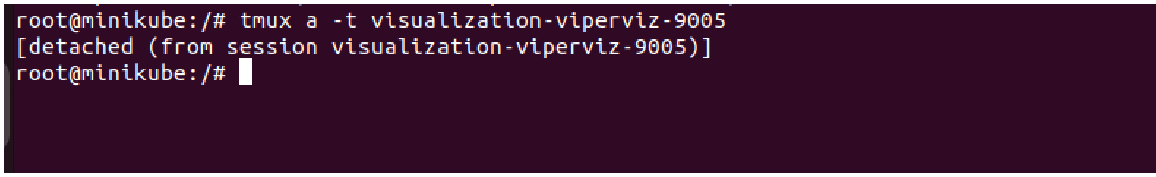


```
[Mon, 18 Sep 2023 15:01:39.8270 UTC] INFO [getconsumelinuxchanneldata Topic=iot-preprocess2,Partition=0, Lastoffset=211,
 Rollingback=200 offsets, Rolledbackoffset=11]
[Mon, 18 Sep 2023 15:01:39.8270 UTC] INFO [consumelinux mainoffset= 11]
[Mon, 18 Sep 2023 15:01:39.8271 UTC] INFO [Startconsumer Topic=iot-preprocess2 offset= 11, Partition = 0, lastoffset=211
,Topicid=-999,Comingfrom=,Filter=,consumerid=StreamConsumer, cluster=127.0.0.1:9092, Formattype=-99]
[Mon, 18 Sep 2023 15:01:39.8271 UTC] INFO [consume Topics=]
[iot-preprocess2]
[Mon, 18 Sep 2023 15:01:39.9355 UTC] INFO [getconsumelinuxchanneldata Success reading message(s) from topic=iot-preproce
ss partition: 0 - 531443 bytes read]
[Mon, 18 Sep 2023 15:01:39.9355 UTC] INFO [consumefromtopic Success reading message(s)from: iot-preprocess - 531443 byte
s read]
[Mon, 18 Sep 2023 15:01:39.9381 UTC] INFO [consume Valid offset found at OFFSET=11, ki=0, Lastoffset=211 - Reading parti
tion=0 for topic=iot-preprocess2 on Cluster=127.0.0.1:9092]
[Mon, 18 Sep 2023 15:01:41.4339 UTC] INFO [getconsumelinuxchanneldata Success reading message(s) from topic=iot-preproce
ss2 partition: 0 - 251731 bytes read]
[Mon, 18 Sep 2023 15:01:41.4340 UTC] INFO [consumefromtopic Success reading message(s)from: iot-preprocess2 - 251731 byt
es read]
[Mon, 18 Sep 2023 15:01:44.1007 UTC] INFO [consumelinuxchannels - ENABLETLS=1]
[Mon, 18 Sep 2023 15:01:44.1031 UTC] INFO [getlastoffsetchannel lastoffset for iot-preprocess is 54749 from partition 0]
[Mon, 18 Sep 2023 15:01:44.1156 UTC] INFO [getconsumelinuxchanneldata Topic=iot-preprocess,Partition=0, Lastoffset=54748
, Rollingback=200 offsets, Rolledbackoffset=54548]
[Mon, 18 Sep 2023 15:01:44.1157 UTC] INFO [consumelinux mainoffset= 54548]
[Mon, 18 Sep 2023 15:01:44.1157 UTC] INFO [Startconsumer Topic=iot-preprocess offset= 54548, Partition = 0, lastoffset=5
4748,Topicid=-999,Comingfrom=,Filter=,consumerid=StreamConsumer, cluster=127.0.0.1:9092, Formattype=-99]
[Mon, 18 Sep 2023 15:01:44.1158 UTC] INFO [consume Topics=]
[iot-preprocess]
[Mon, 18 Sep 2023 15:01:44.1176 UTC] INFO [consume Valid offset found at OFFSET=54548, ki=0, Lastoffset=54748 - Reading
partition=0 for topic=iot-preprocess on Cluster=127.0.0.1:9092]

[visualiza0:viperviz-linux-amd64*                                                            "minikube" 15:01 18-Sep-23
```

e. **Exit out of tmux**

   i. **RUN: Ctrl+B, then D**

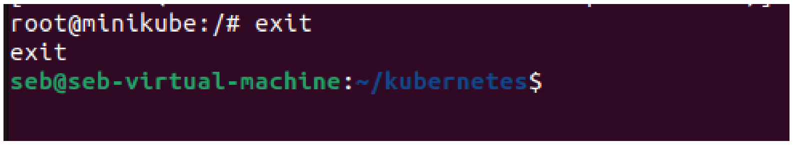
   ii. **You should see:**



```
root@minikube:/# tmux a -t visualization-viperviz-9005
[detached (from session visualization-viperviz-9005)]
root@minikube:/# 
```

f. **Exit from container:**

   i. **RUN: exit**



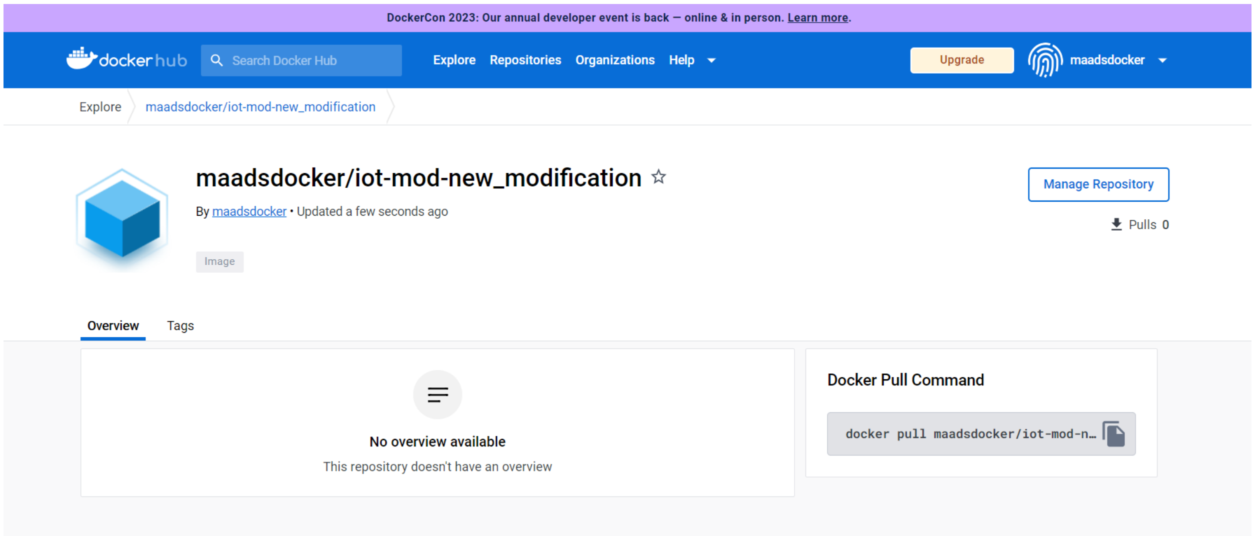
```
root@minikube:/# exit
exit
seb@seb-virtual-machine:~/kubernetes$
```

g. **DONE!**

Submit ONE (1) PDF Document to Black Board for Grading containing ALL of the commands/screenshots below:

1. **Provide the Docker Pull command to your modified container from Docker Hub**
   a) For example: **docker pull maadsdocker/iot-mod-new_modification**
   b) **Replace maadsdocker with your DockerHub ID**

2. **Provide a screenshot of your container in Dockerhub: For example**



DockerCon 2023: Our annual developer event is back — online & in person. Learn more.

maadsdocker/iot-mod-new_modification ☆

By maadsdocker • Updated a few seconds ago

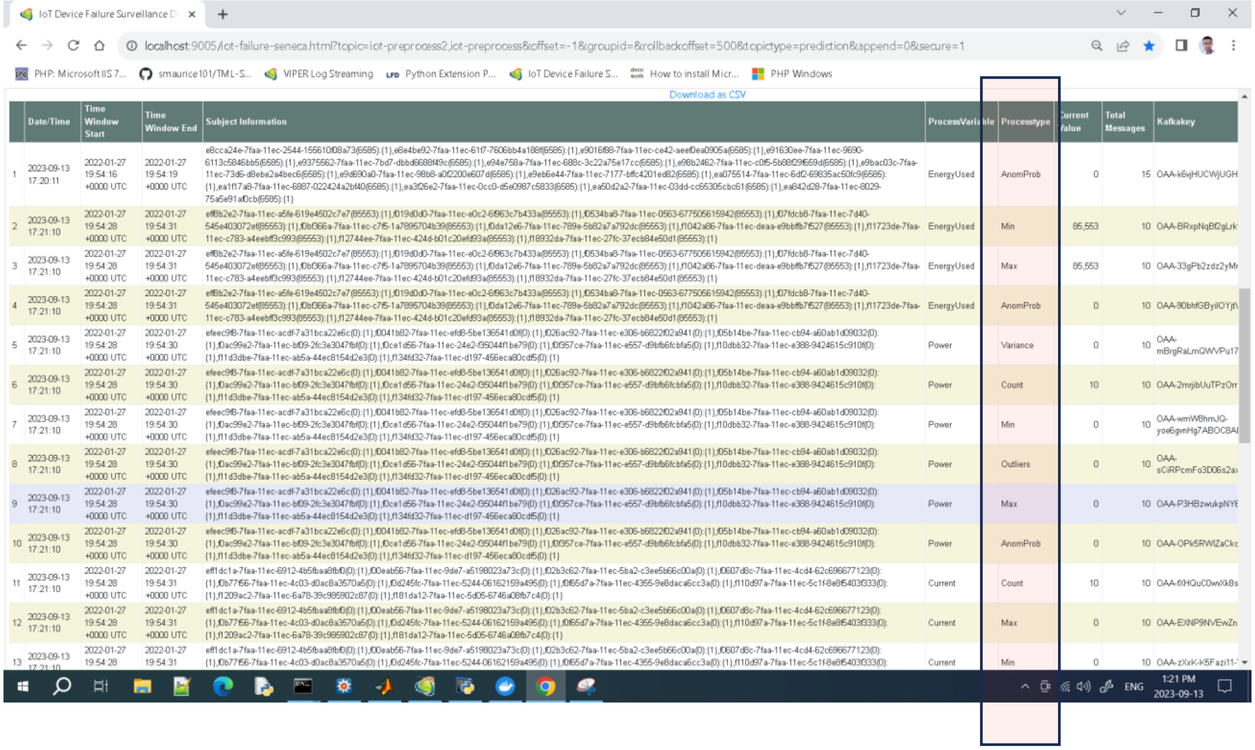
No overview available

This repository doesn't have an overview

Docker Pull Command

```
docker pull maadsdocker/iot-mod-n…
```

3. **Provide Screenshot of your dashboard running WITH the table showing each of the preprocess types in YOUR VM and machine browser:**
   a) Explain in **200 words or less** what the preprocess types are doing.

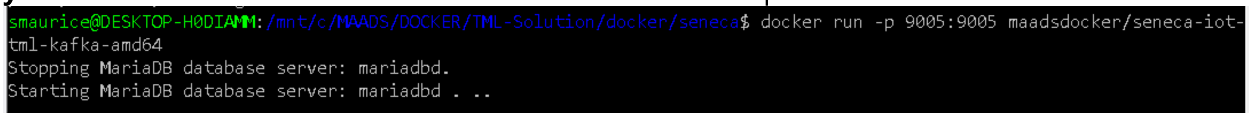


localhost:9005/iot-failure-seneca.html?topic=iot-preprocess2,iot-preprocess&offset=-1&groupid=&rollbackoffset=500&topictype=prediction&append=0&secure=1

Download as CSV

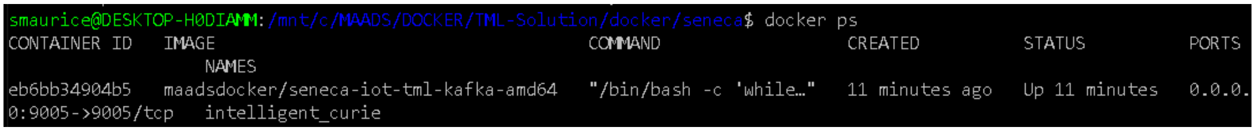
	Date/Time	Time Window Start	Time Window End	Subject Information	ProcessVariable	Processtype	Current Value	Total Messages	Kafkakey
1	2023-09-13 17:20:11	2022-01-27 19:54:16 +0000 UTC	2022-01-27 19:54:19 +0000 UTC	e8cca24e-7faa-11ec-2544-155610f08a73(6585):{1},e8e4be92-7faa-11ec-61f7-7606bb4a188f(6585):{1},e9016f88-7faa-11ec-ce42-aeef0ea0905a(6585):{1},e91630ee-7faa-11ec-9690-6113c5846bb5(6585):{1},e9375562-7faa-11ec-7bd7-dbbd6688f49c(6585):{1},e94e758a-7faa-11ec-688c-3c22a75e17cc(6585):{1},e98b2462-7faa-11ec-c0f5-5b88f29f659d(6585):{1},e9bac03c-7faa-11ec-73d6-d8ebe2a4bec6(6585):{1},e9d690a0-7faa-11ec-98b8-a0f2200e607d(6585):{1},e9eb6e44-7faa-11ec-7177-bffc4201ed82(6585):{1},ea075514-7faa-11ec-6df2-69835ac50fc9(6585):{1},ea1f17a8-7faa-11ec-6887-022424a2bf40(6585):{1},ea3f26e2-7faa-11ec-0cc0-d5e0987c5833(6585):{1},ea50d2a2-7faa-11ec-03dd-cc65305cbc61(6585):{1},ea842d28-7faa-11ec-8029-75a5e91af0cb(6585):{1}	EnergyUsed	AnomProb	0	15	OAA-k6vjHUCWjUGH
2	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	Min	85,553	10	OAA-BRxpNqBf2gLrk
3	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	Max	85,553	10	OAA-33gPb2zdz2yMr
4	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff8b2e2-7faa-11ec-a5fe-619e4502c7e7(85553):{1},f019d0d0-7faa-11ec-e0c2-6f963c7b433a(85553):{1},f0534ba8-7faa-11ec-0563-677505615942(85553):{1},f07fdcb8-7faa-11ec-7d40-545e403072ef(85553):{1},f0bf366a-7faa-11ec-c7f5-1a7895704b39(85553):{1},f0da12e6-7faa-11ec-789e-5b82a7a792dc(85553):{1},f1042a86-7faa-11ec-deaa-e9bbffb7f527(85553):{1},f11723de-7faa-11ec-c783-a4eebff3c993(85553):{1},f12744ee-7faa-11ec-424d-b01c20efd93a(85553):{1},f18932da-7faa-11ec-27fc-37ecb84e50d1(85553):{1}	EnergyUsed	AnomProb	0	10	OAA-90bhfGByilOYjt\
5	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Variance	0	10	OAA-mBrgRaLmQWVPu17
6	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Count	10	10	OAA-2mrjibUuTPzOrr
7	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Min	0	10	OAA-wmW8hmJQ-yoe6gvnHg7ABOC8AI
8	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Outliers	0	10	OAA-sCiRPcmFo3D06s2a
9	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	Max	0	10	OAA-P3HBzwukpNY6
10	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:30 +0000 UTC	efeec9f8-7faa-11ec-acdf-7a31bca22e6c(0):{1},f0041b82-7faa-11ec-efd8-5be136541d0f(0):{1},f026ac92-7faa-11ec-e306-b6822f02a941(0):{1},f05b14be-7faa-11ec-cb94-a60ab1d09032(0):{1},f0ac99e2-7faa-11ec-bf09-2fc3e3047fbf(0):{1},f0ce1d56-7faa-11ec-24e2-f35044f1be79(0):{1},f0f357ce-7faa-11ec-e557-d9bfb6fcbfa5(0):{1},f10dbb32-7faa-11ec-e388-9424615c910f(0):{1},f11d3dbe-7faa-11ec-ab5a-44ec8154d2e3(0):{1},f134fd32-7faa-11ec-d197-456eca80cdf5(0):{1}	Power	AnomProb	0	10	OAA-OPk5RWlZaCkc
11	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):{1},f1209ac2-7faa-11ec-6a78-39c985902c87(0):{1},f181da12-7faa-11ec-5d05-6746a08fb7c4(0):{1}	Current	Count	10	10	OAA-fXHQuCOwXk8s
12	2023-09-13 17:21:10	2022-01-27 19:54:28 +0000 UTC	2022-01-27 19:54:31 +0000 UTC	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):{1},f1209ac2-7faa-11ec-6a78-39c985902c87(0):{1},f181da12-7faa-11ec-5d05-6746a08fb7c4(0):{1}	Current	Max	0	10	OAA-EXNP9NVEwZn
13	2023-09-13 17:21:10	2022-01-27 19:54:28	2022-01-27 19:54:31	eff1dc1a-7faa-11ec-6912-4b5fbaa8fbf0(0):{1},f00eab56-7faa-11ec-9de7-a5198023a73c(0):{1},f02b3c62-7faa-11ec-5ba2-c3ee5b66c00a(0):{1},f0607d8c-7faa-11ec-4cd4-62c696677123(0):{1},f0b77f56-7faa-11ec-4c03-d0ac8a3570a5(0):{1},f0d245fc-7faa-11ec-5244-06162159a495(0):{1},f0f65d7a-7faa-11ec-4355-9e8daca6cc3a(0):{1},f110d97a-7faa-11ec-5c1f-8e8f5403f333(0):	Current	Min	0	10	OAA-zXxK-K5Fazi11-

4. **Provide Screenshot of your running container running in YOUR VM or machine that shows your Dockerhub ID in the Docker Run command.** For example:



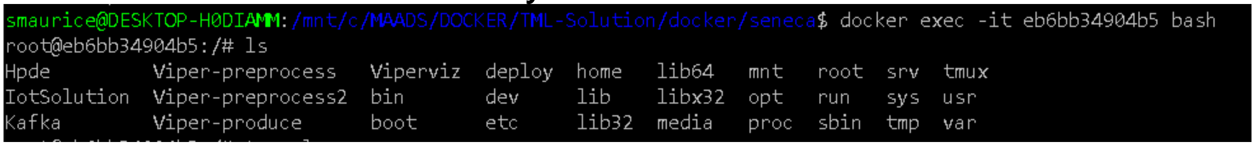
```
smaurice@DESKTOP-H0DIAMM:/mnt/c/MAADS/DOCKER/TML-Solution/docker/seneca$ docker run -p 9005:9005 maadsdocker/seneca-iot-
tml-kafka-amd64
Stopping MariaDB database server: mariadbd.
Starting MariaDB database server: mariadbd . ..
```

5. **Provide Screenshot of the containers AND Go inside your running container:**
   a) **Run**: docker ps
   b) You should see something similar to:



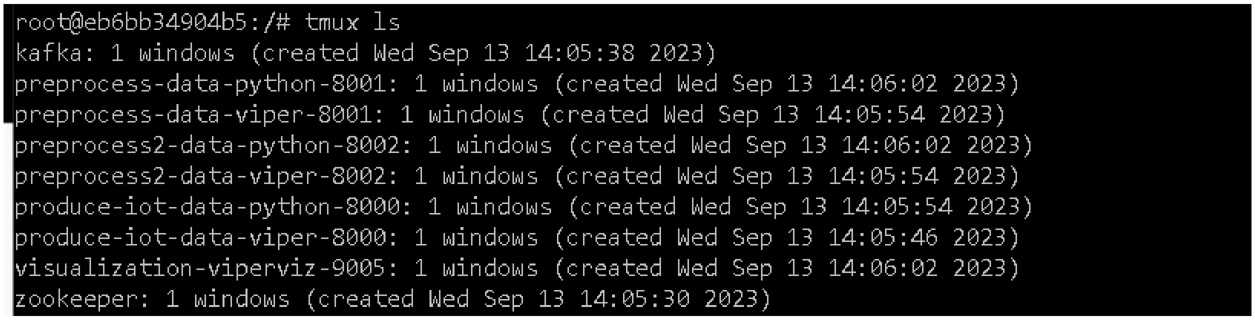
```
smaurice@DESKTOP-H0DIAMM:/mnt/c/MAADS/DOCKER/TML-Solution/docker/seneca$ docker ps
CONTAINER ID   IMAGE                                         COMMAND                  CREATED          STATUS          PORTS
                 NAMES
eb6bb34904b5   maadsdocker/seneca-iot-tml-kafka-amd64   "/bin/bash -c 'while…"   11 minutes ago   Up 11 minutes   0.0.0.
0:9005->9005/tcp    intelligent_curie
```

   c) **Take note of your Container ID – in the example – it is: eb6bb34904b5**
      1. **Run:** docker exec -it eb6bb34904b5 bash
      2. **Then type:** ls
      3. **Provide screenshot of inside your container - You should see:**



```
smaurice@DESKTOP-H0DIAMM:/mnt/c/MAADS/DOCKER/TML-Solution/docker/seneca$ docker exec -it eb6bb34904b5 bash
root@eb6bb34904b5:/# ls
Hpde          Viper-preprocess   Viperviz  deploy  home   lib64   mnt   root  srv   tmux
IotSolution   Viper-preprocess2  bin       dev     lib    libx32  opt   run   sys   usr
Kafka         Viper-produce      boot      etc     lib32  media   proc  sbin  tmp   var
```

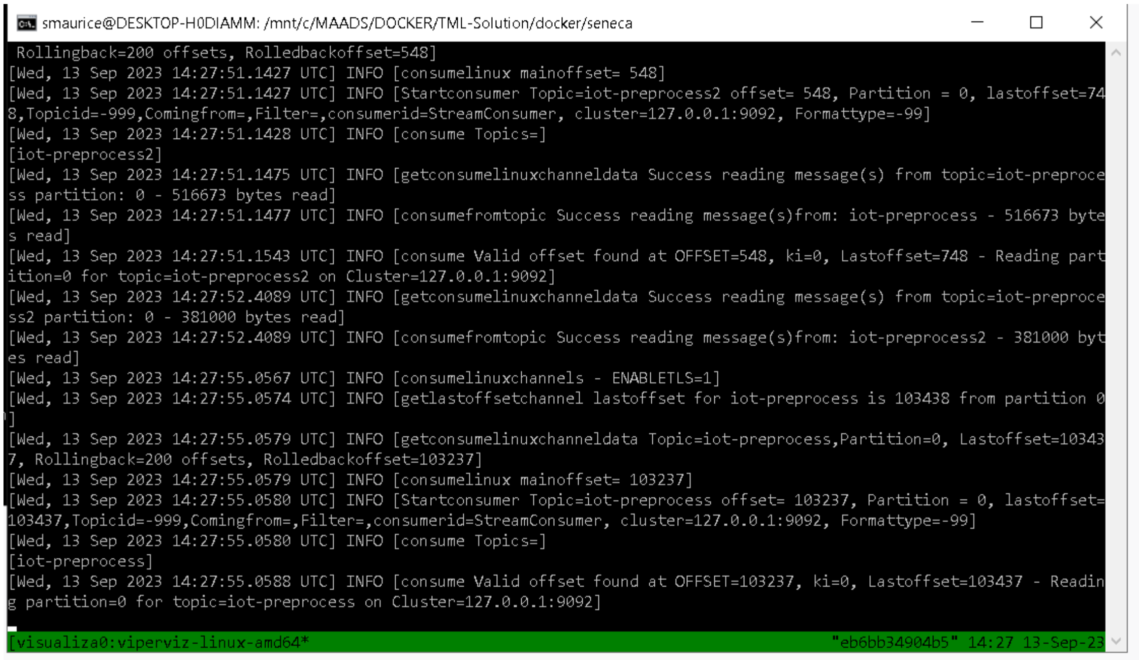
   d) **Now type:** tmux ls
   e) **Provide screenshot of TMUX windows - You should see the following TMUX windows:**



```
root@eb6bb34904b5:/# tmux ls
kafka: 1 windows (created Wed Sep 13 14:05:38 2023)
preprocess-data-python-8001: 1 windows (created Wed Sep 13 14:06:02 2023)
preprocess-data-viper-8001: 1 windows (created Wed Sep 13 14:05:54 2023)
preprocess2-data-python-8002: 1 windows (created Wed Sep 13 14:06:02 2023)
preprocess2-data-viper-8002: 1 windows (created Wed Sep 13 14:05:54 2023)
produce-iot-data-python-8000: 1 windows (created Wed Sep 13 14:05:54 2023)
produce-iot-data-viper-8000: 1 windows (created Wed Sep 13 14:05:46 2023)
visualization-viperviz-9005: 1 windows (created Wed Sep 13 14:06:02 2023)
zookeeper: 1 windows (created Wed Sep 13 14:05:30 2023)
```

f) **Provide screenshot of visualization-viperviz-9005 window: Go inside a TMUX window called: visualization-viperviz-9005**

1. **Run: tmux a -t visualization-viperviz-9005**
2. **You should see:**

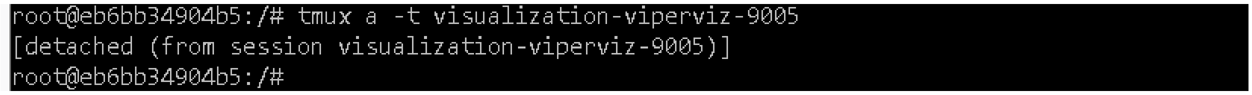


smaurice@DESKTOP-H0DIAMM: /mnt/c/MAADS/DOCKER/TML-Solution/docker/seneca

```
 Rollingback=200 offsets, Rolledbackoffset=548]
[Wed, 13 Sep 2023 14:27:51.1427 UTC] INFO [consumelinux mainoffset= 548]
[Wed, 13 Sep 2023 14:27:51.1427 UTC] INFO [Startconsumer Topic=iot-preprocess2 offset= 548, Partition = 0, lastoffset=74
8,Topicid=-999,Comingfrom=,Filter=,consumerid=StreamConsumer, cluster=127.0.0.1:9092, Formattype=-99]
[Wed, 13 Sep 2023 14:27:51.1428 UTC] INFO [consume Topics=]
[iot-preprocess2]
[Wed, 13 Sep 2023 14:27:51.1475 UTC] INFO [getconsumelinuxchanneldata Success reading message(s) from topic=iot-preproce
ss partition: 0 - 516673 bytes read]
[Wed, 13 Sep 2023 14:27:51.1477 UTC] INFO [consumefromtopic Success reading message(s)from: iot-preprocess - 516673 byte
s read]
[Wed, 13 Sep 2023 14:27:51.1543 UTC] INFO [consume Valid offset found at OFFSET=548, ki=0, Lastoffset=748 - Reading part
ition=0 for topic=iot-preprocess2 on Cluster=127.0.0.1:9092]
[Wed, 13 Sep 2023 14:27:52.4089 UTC] INFO [getconsumelinuxchanneldata Success reading message(s) from topic=iot-preproce
ss2 partition: 0 - 381000 bytes read]
[Wed, 13 Sep 2023 14:27:52.4089 UTC] INFO [consumefromtopic Success reading message(s)from: iot-preprocess2 - 381000 byt
es read]
[Wed, 13 Sep 2023 14:27:55.0567 UTC] INFO [consumelinuxchannels - ENABLETLS=1]
[Wed, 13 Sep 2023 14:27:55.0574 UTC] INFO [getlastoffsetchannel lastoffset for iot-preprocess is 103438 from partition 0
]
[Wed, 13 Sep 2023 14:27:55.0579 UTC] INFO [getconsumelinuxchanneldata Topic=iot-preprocess,Partition=0, Lastoffset=10343
7, Rollingback=200 offsets, Rolledbackoffset=103237]
[Wed, 13 Sep 2023 14:27:55.0579 UTC] INFO [consumelinux mainoffset= 103237]
[Wed, 13 Sep 2023 14:27:55.0580 UTC] INFO [Startconsumer Topic=iot-preprocess offset= 103237, Partition = 0, lastoffset=
103437,Topicid=-999,Comingfrom=,Filter=,consumerid=StreamConsumer, cluster=127.0.0.1:9092, Formattype=-99]
[Wed, 13 Sep 2023 14:27:55.0580 UTC] INFO [consume Topics=]
[iot-preprocess]
[Wed, 13 Sep 2023 14:27:55.0588 UTC] INFO [consume Valid offset found at OFFSET=103237, ki=0, Lastoffset=103437 - Readin
g partition=0 for topic=iot-preprocess on Cluster=127.0.0.1:9092]

[visualiza0:viperviz-linux-amd64*                                                  "eb6bb34904b5" 14:27 13-Sep-23
```

g) **Provide screenshot of exit screen: Exit out of the TMUX window by pressing: CTRL+B, THEN D**

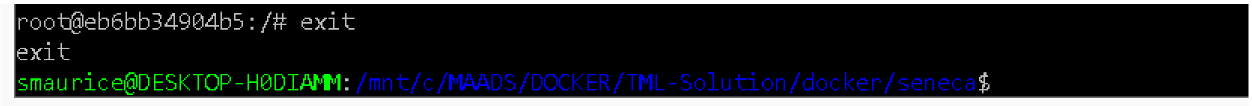
You should see:



```
root@eb6bb34904b5:/# tmux a -t visualization-viperviz-9005
[detached (from session visualization-viperviz-9005)]
root@eb6bb34904b5:/#
```

h) **Provide screenshot of exit container: Exit Out of the Container:**

1. **You should see:**

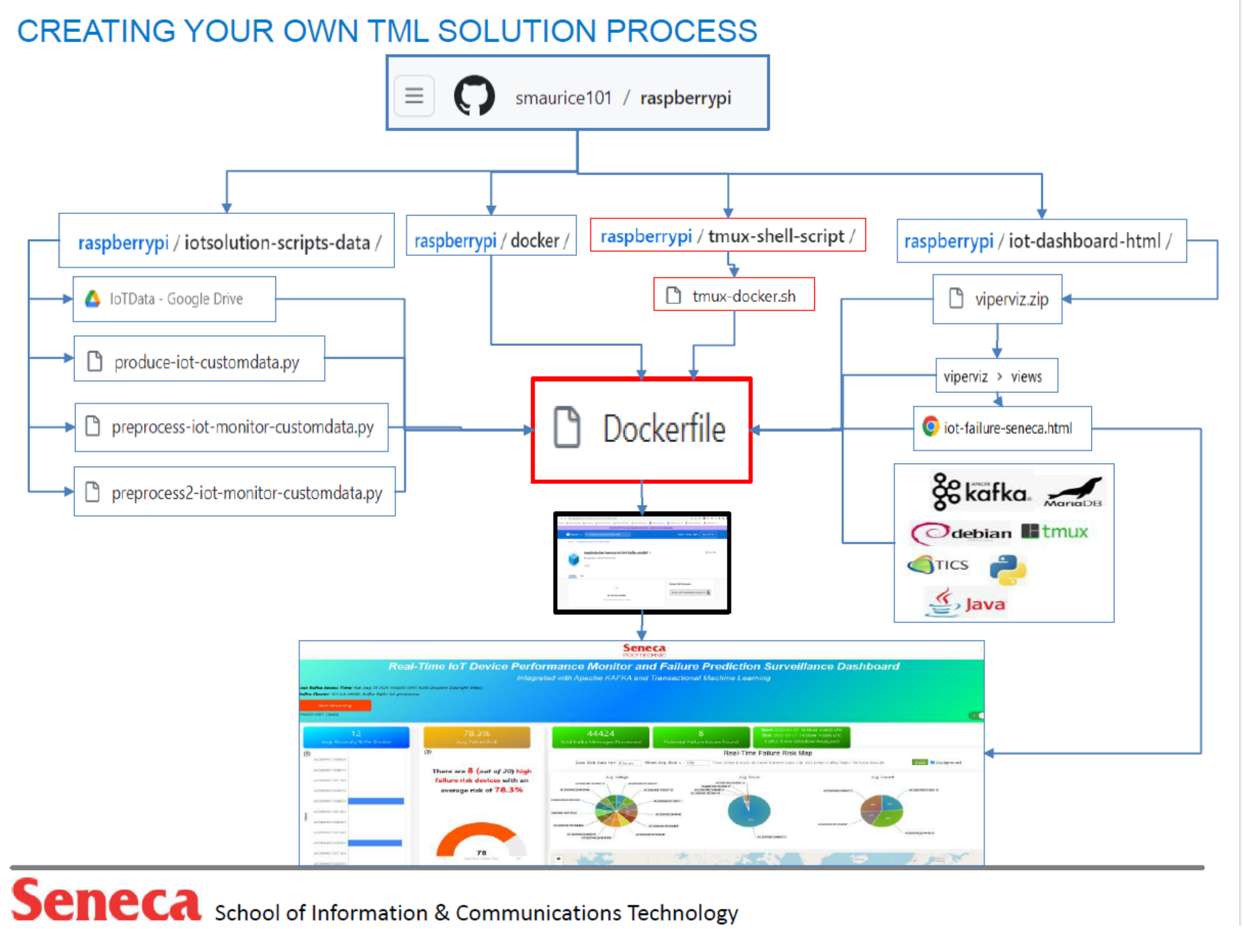


```
root@eb6bb34904b5:/# exit
exit
smaurice@DESKTOP-H0DIAMM:/mnt/c/MAADS/DOCKER/TML-Solution/docker/seneca$
```

6. **SUBMIT ALL KUBERNETES STEPS/SCREENSHOTS in SECTION B and ALL STEPS and SUB-STEPS in 1-17**

APPENDIX (for your reference)

(Source: How TML Works.PDF)



CREATING YOUR OWN TML SOLUTION PROCESS

smaurice101 / raspberrypi

raspberrypi / iotsolution-scripts-data /

IoTData - Google Drive

produce-iot-customdata.py

preprocess-iot-monitor-customdata.py

preprocess2-iot-monitor-customdata.py

raspberrypi / docker /

raspberrypi / tmux-shell-script /

tmux-docker.sh

Dockerfile

raspberrypi / iot-dashboard-html /

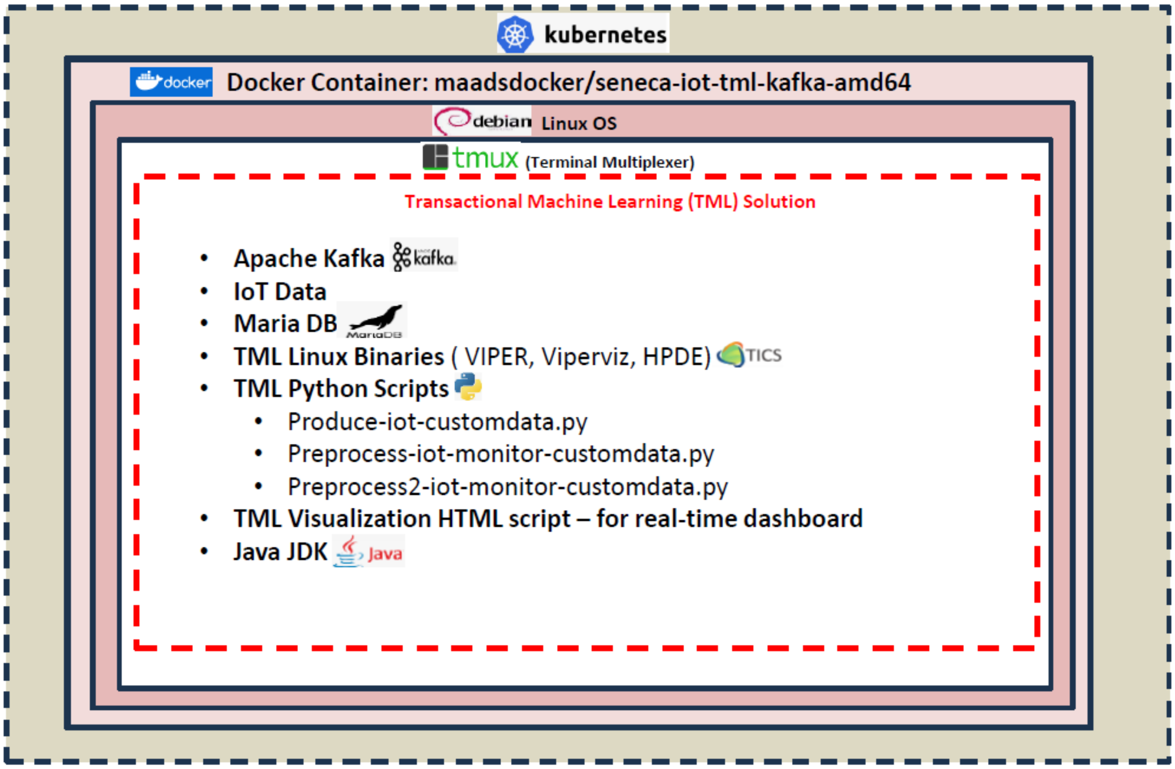
viperviz.zip

viperviz › views

iot-failure-seneca.html

kafka
MariaDB
debian
tmux
TICS
Java
Seneca POLYTECHNIC
Real-Time IoT Device Performance Monitor and Failure Prediction Surveillance Dashboard
Integrated with Apache KAFKA and Transactional Machine Learning
12
78.3%
44424
8
Real-Time Failure Risk Map
There are 8 (out of 20) high failure risk devices with an average risk of 78.3%
78

Seneca School of Information & Communications Technology



kubernetes

Docker Container: maadsdocker/seneca-iot-tml-kafka-amd64

debian Linux OS

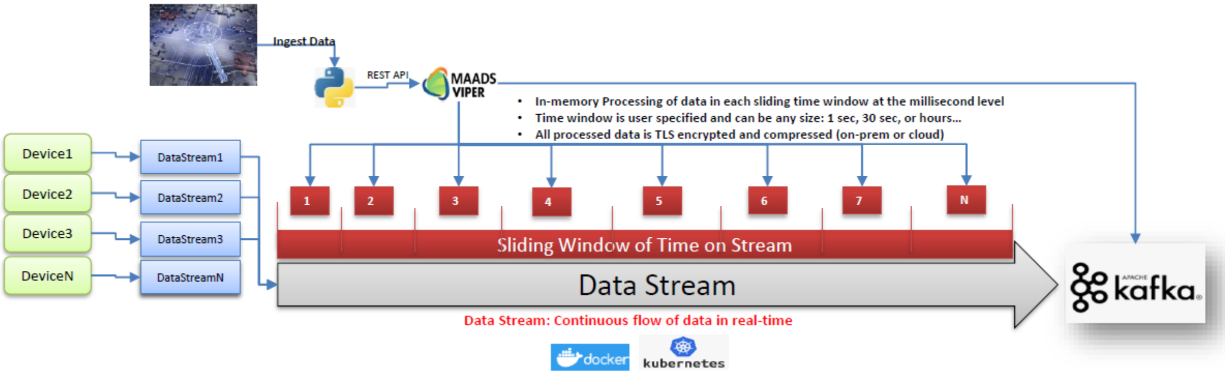
tmux (Terminal Multiplexer)

Transactional Machine Learning (TML) Solution

- **Apache Kafka** kafka
- **IoT Data**
- **Maria DB** MariaDB
- **TML Linux Binaries** (VIPER, Viperviz, HPDE) TICS
- **TML Python Scripts**
  - Produce-iot-customdata.py
  - Preprocess-iot-monitor-customdata.py
  - Preprocess2-iot-monitor-customdata.py
- **TML Visualization HTML script – for real-time dashboard**
- **Java JDK** Java

How TML Preprocessing works?

- All data streams from devices flow into Apache Kafka to a Kafka Topic
- TML performs **in-memory** processing of data in the Kafka Topic using TWO components across all **sliding time windows**
  - Python Script that uses the MAADSTML python library functions
  - MAADS-VIPER binary that can run in Linux, Mac, Windows (or other operating systems) on any Chip (32 or 64 bit) architecture (AMD, ARM, PPC, S390x, etc.)
- REST API connect MAADSTML python script to MAADS-VIPER
- 35+ different processing types: min, max, dataage, timediff, variance, anomaly prediction, outlier detection, etc…
- Apache Kafka is the central source of both input and output data **– *no external real-time database needed***
- ***No SQL queries are made for processing and machine learning***
- *Our technology can process unlimited number of devices (billions at high speed)*
- All TML solutions are containerized with Docker and scale with Kubernetes

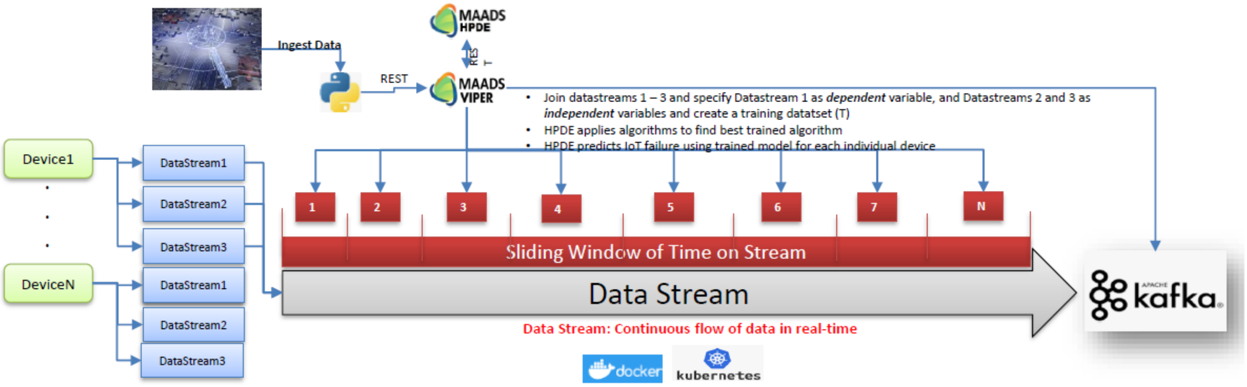


- In-memory Processing of data in each sliding time window at the millisecond level
- Time window is user specified and can be any size: 1 sec, 30 sec, or hours…
- All processed data is TLS encrypted and compressed (on-prem or cloud)

How TML Machine Learning works?

- All data streams from devices flow into Apache Kafka to a Kafka Topic
- TML performs **in-memory** machine learning of data in the Kafka Topic by **joining data streams** using THREE components across all **sliding time windows**:
  - Python Script that uses the MAADSTML python library functions
  - MAADS-VIPER binary that can run in Linux, Mac, Windows (or other operating systems) on any Chip (32 or 64 bit) architecture (AMD, ARM, PPC, S390x, etc.)
  - MAADS-HPDE binary that can run in Linux, Mac, Windows (or other operating systems) on any Chip (32 or 64 bit) architecture (AMD, ARM, PPC, S390x, etc.)

- REST API connect MAADSTML python script to MAADS-VIPER and MAADS-HPDE
- 5 different algorithm types: logistic regression, linear regression, gradient boosting, neural networks, ridge regression
- Apache Kafka is the central source of both input and output data for estimated parameters – ***no external real-time database needed***
- ***TML auto-creates individual machine learning models for each Device at the "entity" level and joins datastreams 1-3 for each device and user specifies "Dependent" variable streams, and "Independent" variables streams***
- *Our technology can build unlimited machine learning models (billions at high speeds) for unlimited number of devices (billions at high speed)*
- All TML solutions are containerized with Docker and scale with Kubernetes



- Join datastreams 1 – 3 and specify Datastream 1 as *dependent* variable, and Datastreams 2 and 3 as *independent* variables and create a training datatset (T)
- HPDE applies algorithms to find best trained algorithm
- HPDE predicts IoT failure using trained model for each individual device

Additional TML Resources:

1. **TML Crash Course Videos:** https://github.com/smaurice101/raspberrypi/tree/main/TML%20Crash%20course/Videos

2. **TML Binaries:**
   a) https://github.com/smaurice101/transactionalmachinelearning

3. **MAADSTML Python Library:**
   a) https://pypi.org/project/maadstml/

4. **TML Blogs:**
   a) **Stream Processing/Analytics Tools Like Apache Flink is NOT Transactional Machine Learning**
   b) **Data Quality Checking in Data Streams**
   c) **A Fast and Simple Way To Migrate Data Streams Between Kafka Clusters: An Alternative to MirrorMaker2**
   d) **NFT (Ethereum) Price Prediction with Transactional Machine Learning, Kafka (or Redpanda) and Blockchain**
   e) **Detecting Medication Fraud at Scale with Transactional Machine Learning and Blockchain**
   f) **TML and Cybersecurity**
   g) **Contextualizing ChatGPT with Healthcare Data Streams**