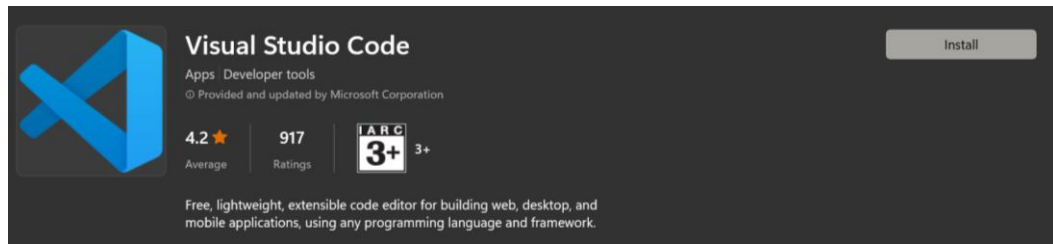


VS Code Installation

Step-01: Go to Microsoft Store and search for VS code and do install it



Step-02: Open VS code and go to **File** in left top corner and select **New Text File**

Paste the given code in the new text file (code give be provided in the zip file NAME)

Step-03: Go to **Terminal** in Top left corner and paste the below one by one and press enter and also install the python from Microsoft store.

```
Pip install paho-mqtt
```

```
Pip install request
```

Step-04: Go to the code and change the Broker ip address in line no. 77

To know broker IP address, go to command prompt and type ipconfig and enter, the Ipv4 address of your connected network is your Ip address.

Step-05: Save the code with extension .py (example receive.py) and run the code to check if any error in present.

Step-06: Again, to create another file, Go to **New Text File** and paste the given code below.

```
Cd (location of your previous saved file)
```

```
Timeout /t 1000
```

```
Python receive.py
```

And save this code in the following location in your PC with name **start.bat**
C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup

```
1 cd C:\Users\hp\Downloads
2 timeout /t 1000
3 python receive.py
```

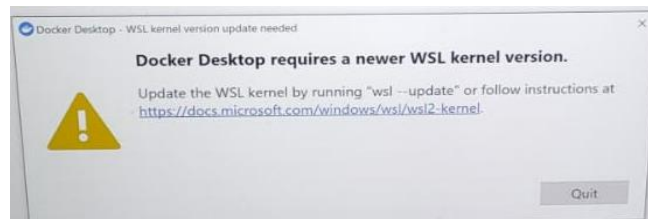
Docker Installation

Step-01: Docker Installation (window/Linux/intel)

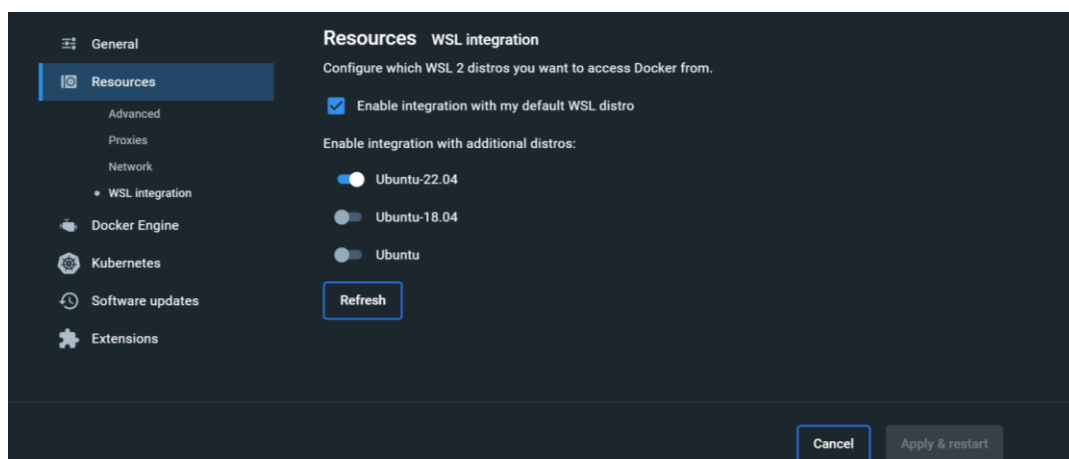
<https://www.docker.com/products/docker-desktop/>

if below popup is shown while installation download the following

https://wslstorestorage.blob.core.windows.net/wslblob/wsl_update_x64.msi



Step-02: Go to Setting at the top, go to Resources, WSL integration, enable and apply.



Ubuntu Installation

Step-01: Go to Microsoft store and install Ubuntu 22.04 2 LTS

Step-02: it will take some time to start. Set the password and username when it asks for.

Aws cli Installation

Step-01: Run the following commands one by one in Ubuntu

- ➔ `sudo apt install unzip`
- ➔ `curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"`
- ➔ `unzip awscliv2.zip`
- ➔ `sudo ./aws/install`

after all the commands done. To check the installation type `aws --version` and press and below line should appear if the installation is done successfully.

```
sunanda@sunanda:~$ aws --version
aws-cli/2.12.3 Python/3.11.4 Linux/5.15.90.1-microsoft-standard-WSL2 exe/x86_64.ubuntu.22 prompt/off
```

Note: Try again and again after some time gap if any error like directory not found is coming.

```
ramu@RAMU: ~$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left   Speed
20 55.3M  20 11.2M    0     0 2803k      0  0:00:20  0:00:04  0:00:16 2803k|

ramu@RAMU:~$ unzip awscliv2.zip
Archive:  awscliv2.zip
  creating: aws/
  creating: aws/dist/
  inflating: aws/THIRD_PARTY_LICENSES
  inflating: aws/README.md
  inflating: aws/install
    creating: aws/dist/awscli/
    creating: aws/dist/cryptography/
    creating: aws/dist/docutils/
    creating: aws/dist/lib-dynload/
  inflating: aws/dist/aws
  inflating: aws/dist/aws_completer
  inflating: aws/dist/libpython3.11.so.1.0
  inflating: aws/dist/_awsert.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/_cffi_backend.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/_ruamel_yaml.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/libz.so.1
  inflating: aws/dist/liblzma.so.0
  inflating: aws/dist/libbz2.so.1
  inflating: aws/dist/libffi.so.5
  inflating: aws/dist/libsqlite3.so.0
  inflating: aws/dist/base_library.zip
  inflating: aws/dist/lib-dynload/pickle.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/hashlib.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/sha3.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/blake2.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/sha256.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/md5.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/sha1.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/sha512.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/random.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/bisect.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/csv.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/binascii.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/grp.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/resource.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/lzma.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/bz2.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/pyexpat.cpython-311-x86_64-linux-gnu.so
```

Thingsboard Installation

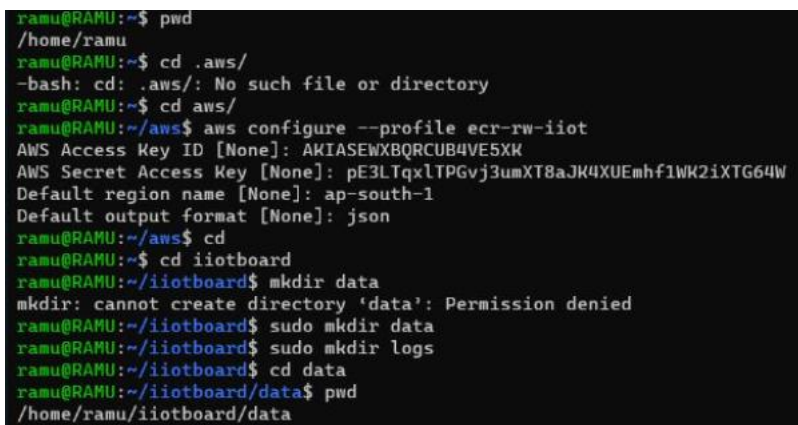
Step-01: Create a folder name Iiotboard in the following location in your PC

[\\wsl.localhost\Ubuntu-22.04\home\sunanda](#)

Step-02: Copy paste the docker-compose file in that folder
(docker_compose.yaml file will be provided)

Step-03: Run the following Commands in Ubuntu

```
pwd
cd aws/
aws configure --profile ecr-rw-iiot
->AWS Access Key ID: AKIASEWXBQRCUB4VE5XK
->AWS Secret Access Key: pE3LTqxITPGvj3umXT8aJK4XUEmh1WK2iXTG64W
->Default region name: ap-south-1
->Default output Format: json
cd
cd iiotboard
sudo mkdir data
sudo mkdir logs
cd data
pwd
```



```
ramu@RAMU:~$ pwd
/home/ramu
ramu@RAMU:~$ cd .aws/
-bash: cd: .aws/: No such file or directory
ramu@RAMU:~$ cd aws/
ramu@RAMU:~/aws$ aws configure --profile ecr-rw-iiot
AWS Access Key ID [None]: AKIASEWXBQRCUB4VE5XK
AWS Secret Access Key [None]: pE3LTqxITPGvj3umXT8aJK4XUEmh1WK2iXTG64W
Default region name [None]: ap-south-1
Default output format [None]: json
ramu@RAMU:~/aws$ cd
ramu@RAMU:~$ cd iiotboard
ramu@RAMU:~/iiotboard$ mkdir data
mkdir: cannot create directory 'data': Permission denied
ramu@RAMU:~/iiotboard$ sudo mkdir data
ramu@RAMU:~/iiotboard$ sudo mkdir logs
ramu@RAMU:~/iiotboard$ cd data
ramu@RAMU:~/iiotboard/data$ pwd
/home/ramu/iiotboard/data
```

Copy the output in docker-compose in last subsection Volumes instead of given location

```
volumes:
  - ~/.mytb-data:/home/sunanda/iiotboard/data
  - ~/.mytb-logs:/home/sunanda/iiotboard/logs
```

```
->aws ecr get-login-password --region ap-south-1 --profile ecr-rw-iiot
```

[illegible]

```
->aws ecr --region ap-south-1 | docker login -u AWS -p
eyJwYXlsb2FkljoiMHlaUkQvdTFvcnZTL21rTkxGTDZnckdrMU96ZTVWdzJ2M0xTZ
090cGVEQTRNUTdXbnJmNkMzOU95Qm1VaFZzNFBBT3RWbk5xQlc3ZjYxTzlwT
WFNZTVLRlhqKzVxSS9BRUwySUR0V2ZhrmtPNFZHVEIRUWszdkRmTWgrWFlmc
kZJNG8rY2ZHVWRvajM3SmRZVFFJSXU4clppcUR2ZHBURFJBWDUvZIAzc1NRd2d
wc1RRMVRfcmRKblZSMURGVm1LTDF2NFIOMEZoTG1KcmdRUWcVNVNBb1phV
npKVnVSZnBfOWpraTBkQ3dpdFNhdU5ldmNCWWNiSIBNZnFxUDFmMlhWn1Rz
U1JwMDMxaDA0dHVvaTFJhbVR4b2dyaG05UIRKT2kvaXdYaC9qUFE0VW43cmVx
YzhoSE94OG4yR3Rnb3RLS0VOV1NIVFA3Qk1JREMrSksyL05WYWNBVBIWWZ0
dFZ1RldpbIFuMjF2b0Q0c0pUdDIKYVFKR1lWdmJORKhZSIlFcEJnZTUwYXBjBk3
WkdmSXZGQ1J0WWJ3MHFTejRKSvVxRkRWR083Y29YWE5ldktyMUh1M2U5R0t
ET3BUczNkUm9GSXFiSm9zNTVVTGUzeW9LNkdXNjVGei9Galo3NExwT1A1NDlZ
NTZNQVBPNWFydDV6S3ISM1R2ak9XVGxDUE8xRnBLalRacm5VSnnjMkpoVWM
wWmFKclU3bjVDcThkak1ONDFITkFWRHdxZEIPQmhBcVVvcEJJdmk5bzZ3VzND
M0VqR1BvRzFBSUJFYUc5SktvYlNYQ2NhcVAzVTNjT0RwekVuWDBYsJy1cW5tNG
phQ0FGQ25McmxJcHVEQ1l4ekhGUjRzelVJckxTVTVqaHRSdURZaXo5d1Frdk1nW
Gd2VIFIMEphSDZ6RnA5TUVVY3FibGNGVzhYWkNuQzhwNk40Q0lCbC8xa2Z3Qk
0wZ1BXVk5iV2k1eFpVVIpxZnQ2emJjeGJNQWJZTDZkS0VlbTRVbINTVm5jL1hnTn
U5anB3V1INTGY0NEd3SzBvOUtKTTBHSEZmOGVxd0ZMRXl0enF5S0JoVDBZWk9
LaUswcVQ5dVZYNjRpeTM2QjFvR3hMQXE3VURKS2t0MVfwaktWWDc0MlIHRW
JwMGdLWkh4SWlTaHoyd2wvUmhrR0FOQmtwNzl0TFNIOWJFV2RLZGNjZC9XW
DB5Z3UzRFplRFh6NHU4NVdZd2RQUnlwQkdEZkZMMGZmTElvWVZzcTlZd3BwM
HYrdEM2MHFrBw55dm9qSXdQM0VITU40dElGdVI6TitXQm9VeHdocEdzWHZ0V
TRtcnRJejdqSFk2a1hKbmRINE93QVhJUfo5U0Z4THlmUVhyK3dpU0xhS1RUTFFJ
```



```
XUDFMtHlwH1R2u1JmIMDMyAd8dHJvzTF3bKvRbU52dyaga0SUTKt2kVaxdYc9qUFE09WuH33mVxYzhoES940GHyR3Rnb3RL59vOv1M1VFA3Qk13REHmSk5yL65YVmwBNW1b1Wz6DZF1R1dpb1f
uHfJ2b0q0c9pUDtLkVwFKR1u1JmIMDJKrHkZ51fEcJnZUwYXbJbHk3WkdKXVZGQ1JowM3JMHfTeJrKSVvXrKRW8R33Y29VWES1dktyMuh1M2U5R0eET3BUCzNKUmw9GSXf1Sme9ZntVTVGUzeW9
1mKdxNjYg91eGa1o3bG1m1A1NDJrZTnQVbPMWfydD6V53LSM1R2ak9XGQvDUE8XrNbLaLRacm5VSnN1JkPovWmWmFKcLU3bJvDCThkka1ONDf1KfWIRHdxZELPQmhBcVvucEJJdmk5bzZ
3vZNDM0vqR1eBvRzFBSUJYfUc5SK5tv1NYQ2Nh-cVzVTNjTRbWekVuDBySjY1cWStNGphQ0FQ25McmxjChVEQ114ekHGUjRze1VJckvT1VTVqaHRSDURZax0sD1Frdk1NW6D2V1LFMEphSDZ
6rNAStUvVY3fIBGNGVzHvYmKwQzhhNk4Q0B1CbC8kaZ23Qk0wZ1BXV51V2k1eFpVvLpxZnQ2emJjgEgJNQWJZT2k4S0vIDbTRV61NTVw5J1lhnTnU5anB3V11NTGvONED3S3zBv0utkTTBHSzE
mOGvXd2ZMRX10e9F5S0JvDVBZwK9LaUswcVQ5dVZYNjRpeTM20jFVr3hMQXE3VURHS2tBMVfwaktWmDc0M1LHRWJmWGLDkWh4SW1TAhoyd2vmvIhR0FQ0ntwN2LGTFNL0WJFV2RLZGNjZC9
XWDBS3ZuzRfPLRFh6HmUu4Wvxd2dRQUnLWkdqDEZkMGGZmTE1VwZzc1fZ2d3BmMHYdEEMZMHFRbW5Sdm9qSdXm0V1TUA0dE1GdV161f1XQm9VeHdocEdzWzh0vTRtcnR0ejdqSFk2a1hKbmR
IEN93vQhVJG5U0Z4TH1mUvnyh3k3dpU0xh51RUTFFJ3L1TUG6JaYnU0P51sInRhdGFzXkx1o1JBUULQCUhpsFdhvWruL1VXQ2Jue1s3THZNrytBUH2USHPiBzJUTV1GcUvTVj12QmR3d0ZyTHF
zMUfBY3tZbZUR0szV2Z2HXFBQUPBzmqIOwDcJna3F0aZ1H0kcwqk3JYVW1dekJ0gWdFQ11H20dD3FHU01M0RRRRhBVEFLQm5Z2hrZ0JaUu1FQWMed0VRUu0g0tHeCvZTMm1pM9F7xdQTQWd
F1WdEdWhKcGehJenMLUHL1RZC4QqkRVaM9IOwVd90ZmU1U00EzK0VGQWRB0R0RERCc1LnRHVPQ1L1bWm1ZUFCEdZC23Z0VUFqcl1DRjBEZU5mRnc9P5S1nZ1cnNpb24101iy1i1w1dHlwZSI6
1kRbVEFFS0Vzi1wiZXhwaXQgdG91bG1MTY0dE2Y2Mz1U30E= 147553616965.dkr.ecr.ap-south-1.amazonaws.com/i1ot-cassandra
WARNING! Using --password via the CLI is insecure. Use --password-file.
```

```
-> docker pull 147553616965.dkr.ecr.ap-south-1.amazonaws.com/iiot-cassandra
```

```
ramu@RAMU:~/iiotboard/data$ docker pull 147553616965.dkr.ecr.ap-south-1.amazonaws.com/iiot-cassandra
Using default tag: latest
latest: Pulling from iiot-cassandra
9e3ea8720c6d: Pull complete
066524c96da2: Pull complete
3f7d6085deb3: Pull complete
e67d3c362e6d: Pull complete
5f4441daa1a8: Pull complete
Digest: sha256:d11f72442bd2a3b2ae4dc27aee26d71aff0847bc40775975ecf58e0e7a0f776
Status: Downloaded newer image for 147553616965.dkr.ecr.ap-south-1.amazonaws.com/iiot-cassandra:latest
147553616965.dkr.ecr.ap-south-1.amazonaws.com/iiot-cassandra:latest
```

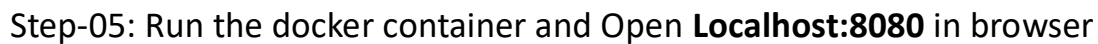
Rerun the following code multiple times until all the 3 containers are started.

```

Layers [0/10000]
7c18790 Pull complete
f2e78f4d7 Pull complete
0c47ef279 Pull complete
a3b0da998 Pull complete
d59c1c190 Pull complete
66d58a161 Pull complete
fc416dc85 Pull complete
uper 0 Layers [0/10000]
c490899d3 Pull complete
09d4c26a3 Pull complete
5d0703f5bc Pull complete
57626abf76 Pull complete
f6b4cfc5d2 Pull complete
8ef709f436 Pull complete
f3b69bf337c Pull complete
596ac63b197 Pull complete
nning 3/4
work iotboard-default Created
tainer iotboard-zookeeper-1 Started
tainer iotboard-kafka-1 Started
tainer iotboard-mysql-1 Starting
response from daemon: driver failed programming external connectivity on endpoint
(228ff6c1b): Bind for 0.0.0.0:8888 failed: port is already allocated
LAPTOP-935CBAL:/home/sai/iotboard/data docker compose up -d
Running 1/3
tainer iotboard-zookeeper-1 Running
tainer iotboard-kafka-1 Running
tainer iotboard-mysql-1 Starting
or response from daemon: driver failed programming external connectivity on endpoint
3a58f68a192): Bind for 0.0.0.0:8888 failed: port is already allocated
t/LAPTOP-935CBAL:/home/sai/iotboard/data docker compose up -d
Running 3/3
Container iotboard-zookeeper-1 Started
Container iotboard-kafka-1 Started
Container iotboard-mysql-1 Started
t/LAPTOP-935CBAL:/home/sai/iotboard/data |

```

```
-> docker compose logs -f mytb
```




Antar IIOT | Login



localhost:8080/login

Verilog HDL - Samir... MATLAB Filtering Using Infil... Courses :: NPTEL Machine Learning C... Machine learning KNN Algorithm - FL... colab nandu java

QANTAR IoT

 Username (email)*

Invalid email format

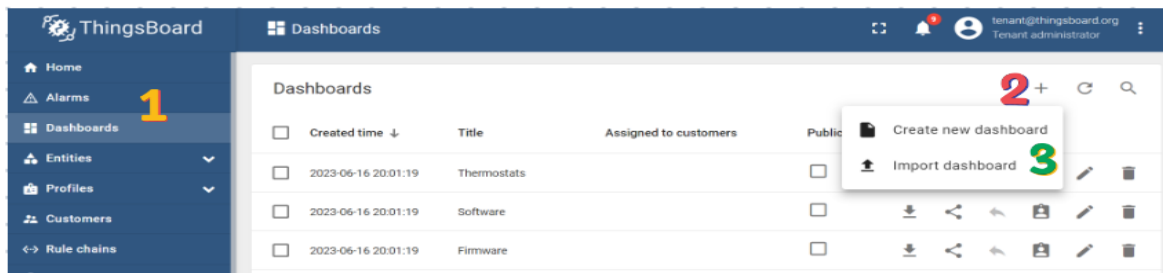
 Password 

[Forgot Password?](#)

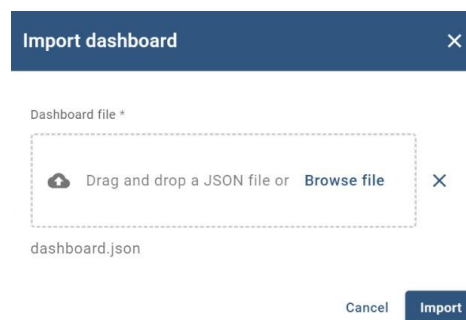
- Demo Accounts: login - **antarsmarthomestesting@gmail.com**
- password – **Antar@1234**

Setup in Thingsboard

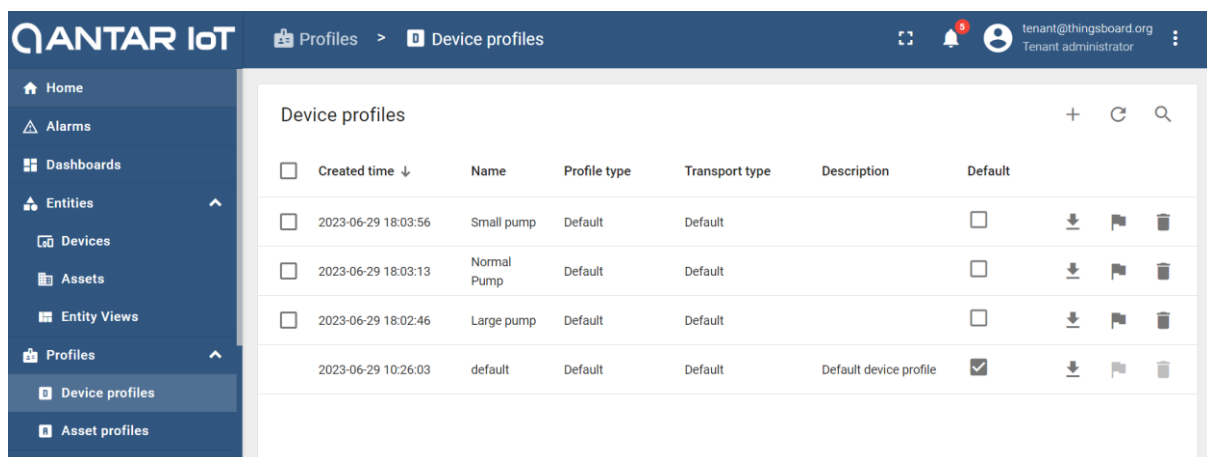
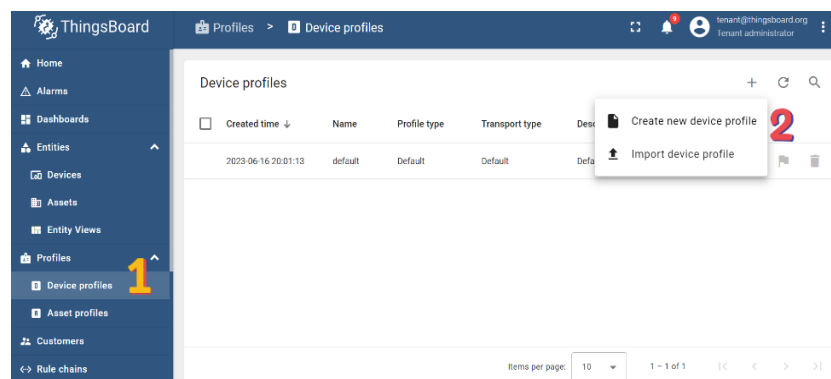
Step:01 - Import the dashboard (it will be provided)



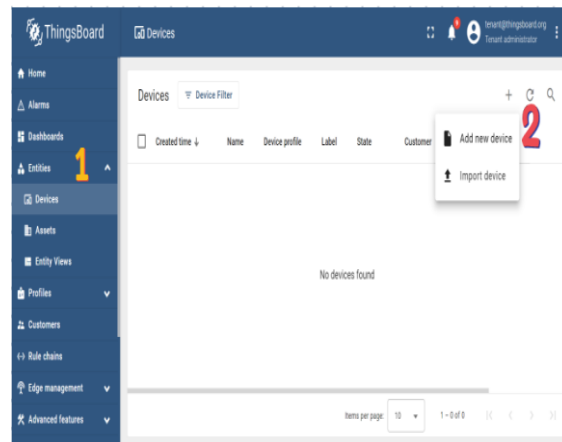
Step:02 – Upload the dashboard by clicking on **Browse File** and import



Step:03 – Import 5 device profiles namely for device1, device2 etc(1 device Profile will be provided for reference)

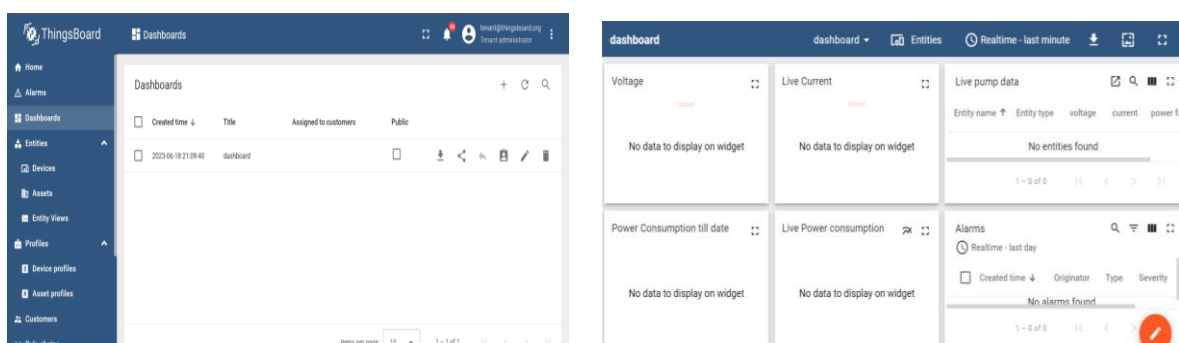



Step:04 – Import 5 Devices (namely Motor1, Motor2, Motor3, Motor4, Motor5) and add the device profile accordingly like motor1 of small pump, like motor2 of large pump




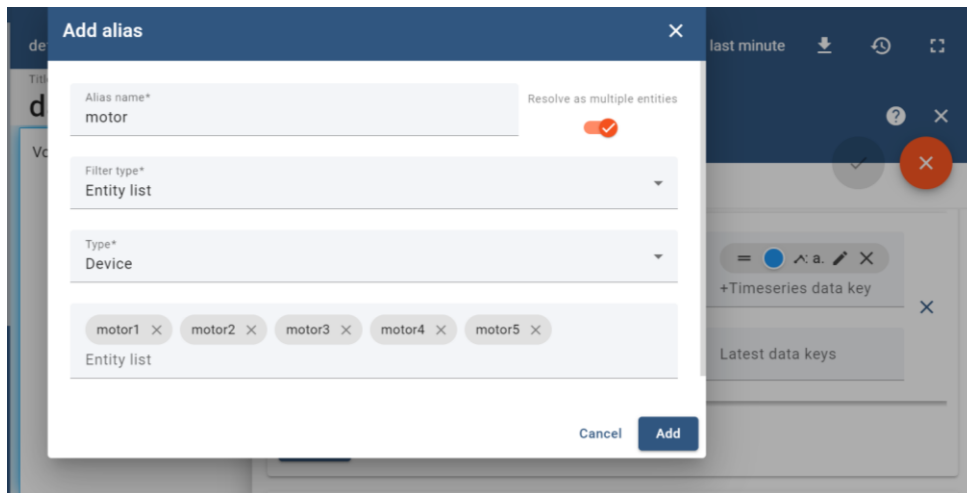
Created time ↓	Name	Device profile	Label	State	Customer
2023-06-18 22:39:25	motor5	motor		Inactive	
2023-06-18 22:39:13	motor4	motor		Inactive	
2023-06-18 22:38:59	motor3	motor		Inactive	
2023-06-18 22:38:47	motor2	motor		Inactive	
2023-06-18 22:38:32	motor1	motor		Inactive	


Step:05 – Setup the Main dashboard

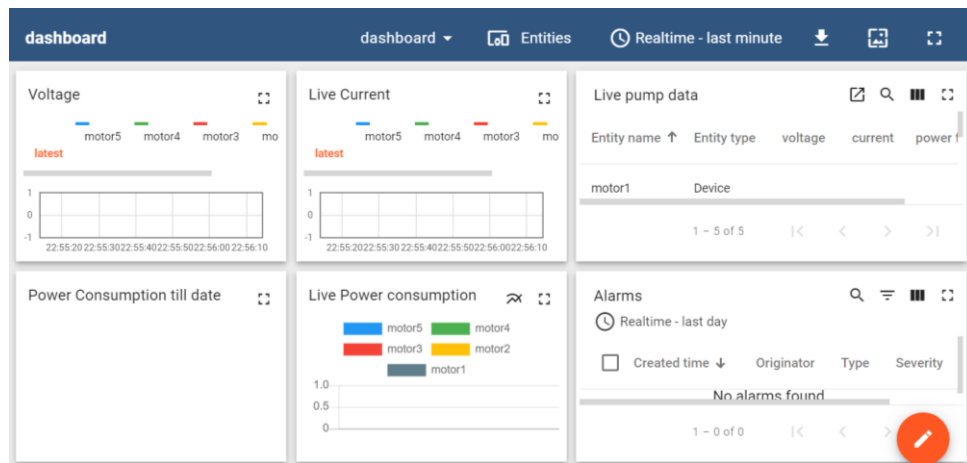


Click on pencil to edit . Edit all the widgets and setup and also add the following Entity alias

1st click on the below  and later the pencil icon on top of all widgets. And add the alias followingly.

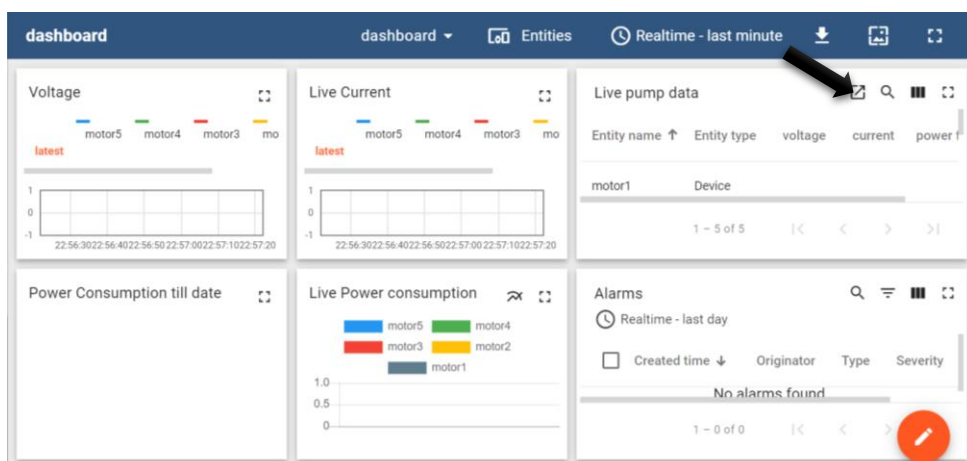


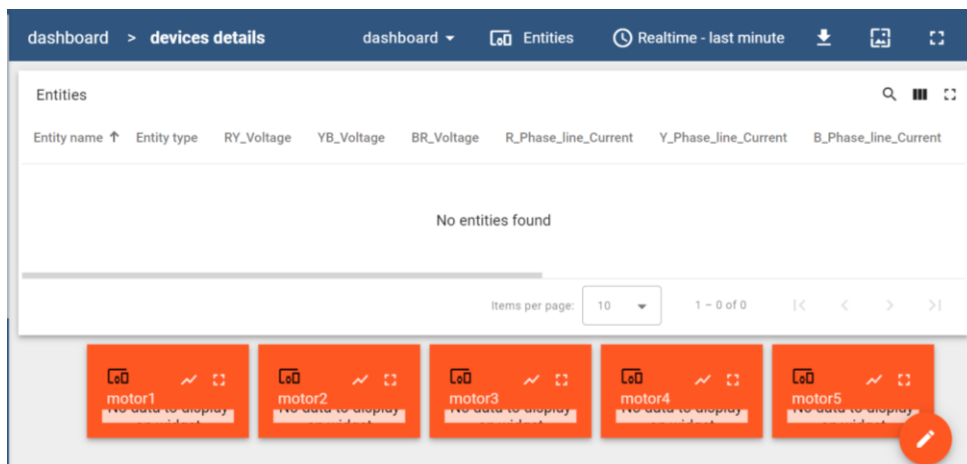
By setting up the alias in all the widgets. Save the changes by clicking the button 




Step:06- Setup the complete dashboard

Go to device details option in Live Pump data widget





Here also setup with the same entity alias (namely motor) to the Entities widget and save the change by clicking  and for Individual device details create another entity alias as follow.

Add alias

Alias name*

pump1

Resolve as multiple entities

☐

Filter type*

Single entity

Type*

Device

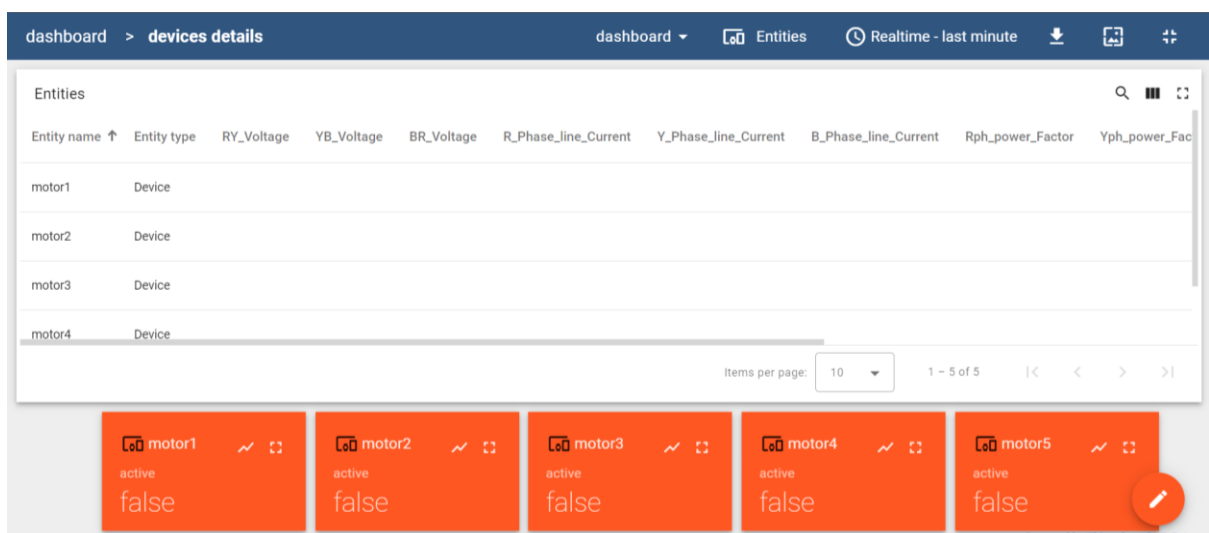
Device*

motor1

Cancel

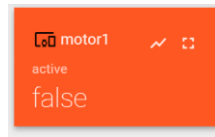
Add



Similarly define for all the motors namely pump1, pump2, pump3, pump4, pump5.

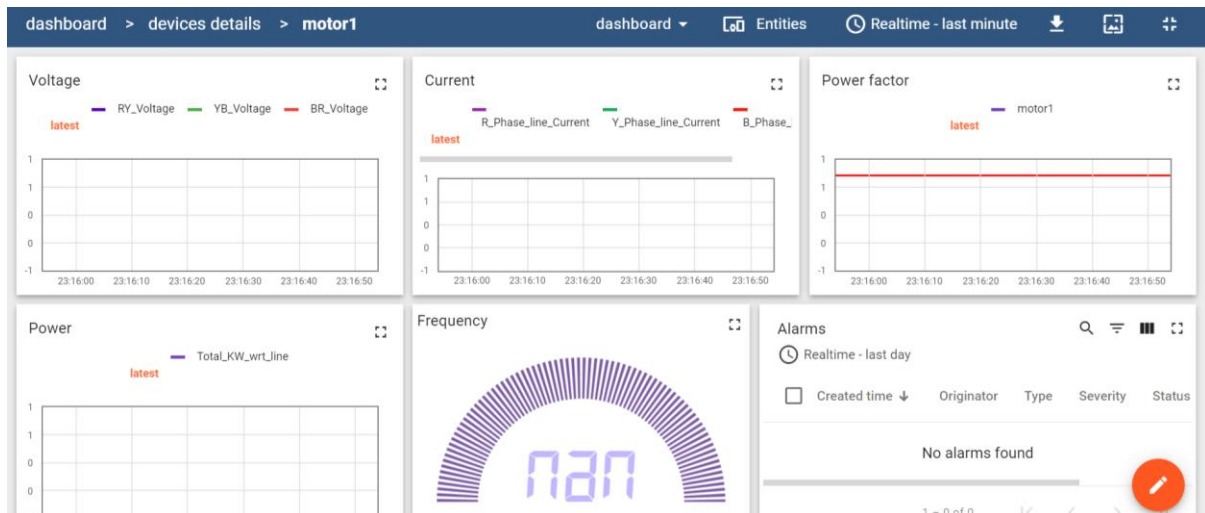


Step:07 – Setup for individual details






go to motor and setup/edit all the widgets with the specific entity (likely for motor1 all widgets with pump1 entity alias, motor2 all the widgets with pump2 entity alias and go on) and save the changes by clicking on . For edit click on pencil icon .

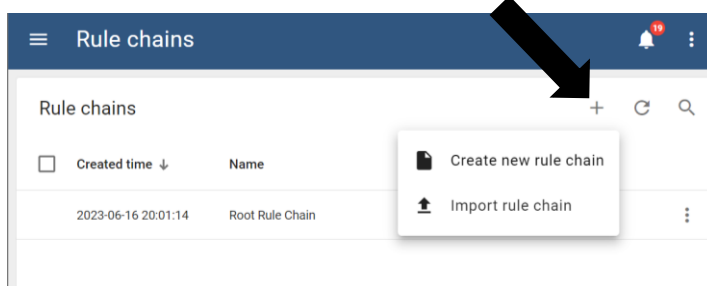


Similarly change for all the individual devices with their entity alias.

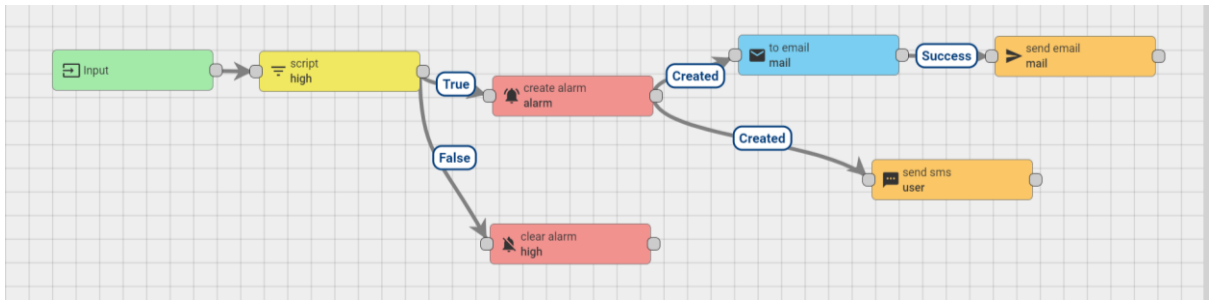
After this final dashboard is ready.

Step:08 – Rule chain setup

Go to root rule chain and import rule chain namely alarm of 3 and warehouse (Will be Provided). And save them by clicking .



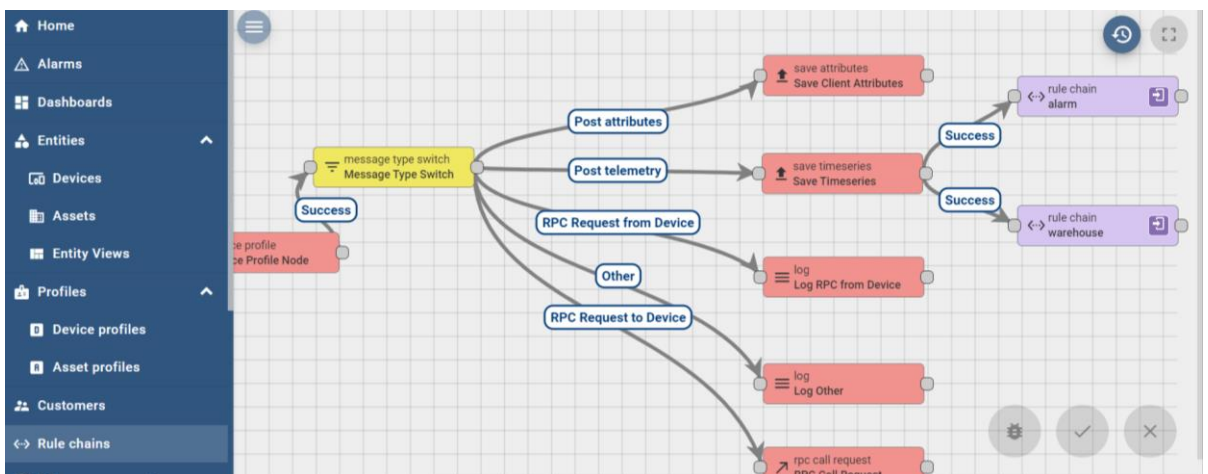
- 3 Alarm named rule chain



- Warehouse named rule chain



- 3 Root Rule Chains



Step-09: Assign Customer

Let's create a customer with title "My New Customer". Please see instruction below:

- Navigate to the Customers page;
- Click the "+" sign to add a customer;
- Input customer title and click "Add".

The first screenshot shows the 'My New Dashboard' interface. The 'Customers' menu item in the left sidebar is highlighted with a yellow box and labeled '1'. A callout box points to it with the text 'Navigate to "Customers" page'.

The second screenshot shows the 'Customers' page. The 'Add Customer' icon (a plus sign inside a circle) is highlighted with a yellow box and labeled '2'. A callout box points to it with the text 'Click "Add Customer" icon'.

Customers

3 Created time

Input title

Title*

My New Customer

Description

Country

City

State / Province

Zip / Postal Code

Address

Address 2

Phone

Cancel Add

4 Click "Add"

Items per page: 10 1 - 0 of 0

Let's assign device to the customer. The customer users will have ability to read and write telemetry and send commands to devices.

- Open "Devices" page, then select your device to open its details;
- Click "Assign to customer" button;
- Select the customer to whom you want to assign the device, and then click "Assign";
- You have changed the owner of the device. In the "Customer" column, you can see the owners name of the device;

Home

Alarms

Dashboards

Entities

Devices

Assets

Entity Views

Profiles

Customers

1 Open "Devices" page

Devices

Device Filter

Created time	Name	Device profile	Label	State	Customer	Public	Is gateway
2023-06-03 08:53:36	My New Device	default		Inactive			

2 Click on your device

My New Device

Device details

Details Attributes Latest telemetry Alarms Events Relations Audit log

Open details page Make device public Assign to customer Manage credentials

Delete device

Copy device Id Copy access token

3 Click "Assign to customer"

Assign Device(s) To Customer

Please select the customer to assign the device(s)

Customer*

My New Customer

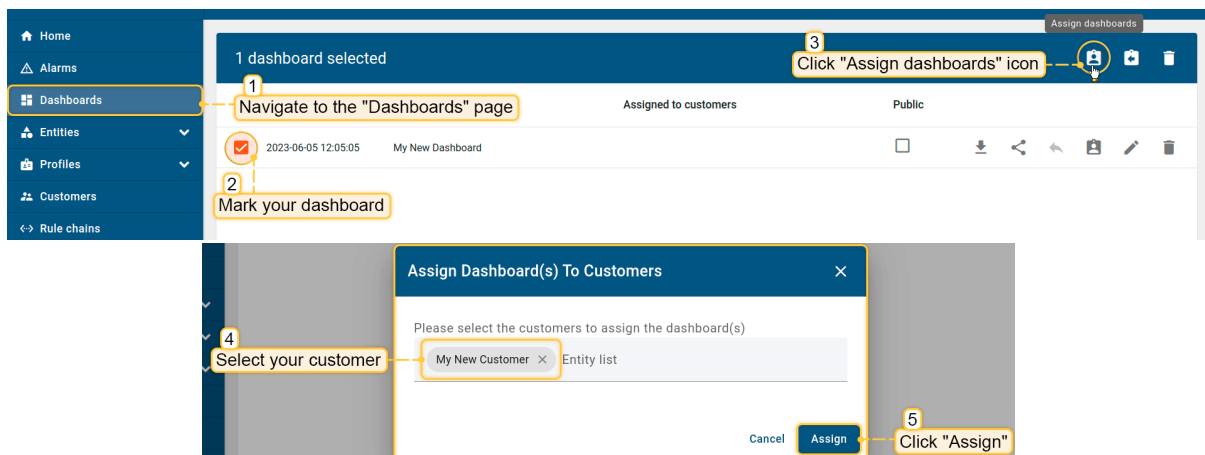
4 Select your customer

Cancel Assign

5 Click "Assign"

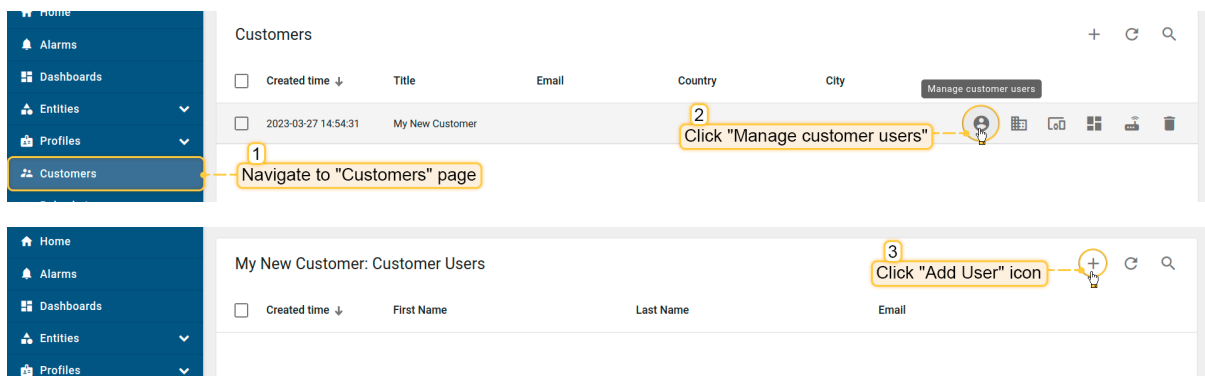
Let's share our dashboard with the customer. The customer users will have read-only access to the dashboard.

- Open "Dashboards" page. Mark your dashboard and click the "Assign dashboards" icon;
- Mark "My New Customer" and click "Assign";
- Navigate to the "Customers" page. Click "Manage customer dashboards" icon for "My New Customer";
- "My New Dashboard" is assigned to your customer.



Create Customer Account:

- Navigate to "Customers" page. Find your customer in the list of customers and then click on the "Manage customer users" icon;
- Click the "Add user" icon in the top right corner of the table;
- Specify email that you will use to login as a customer user and click "Add";
- Copy the activation link and save it to a safe place. You will use it later to set the password. Click "OK";
- Click on the created user to open details. Click "pencil" icon to enter edit mode;
- Select default dashboard and check "Always fullscreen". Apply changes.

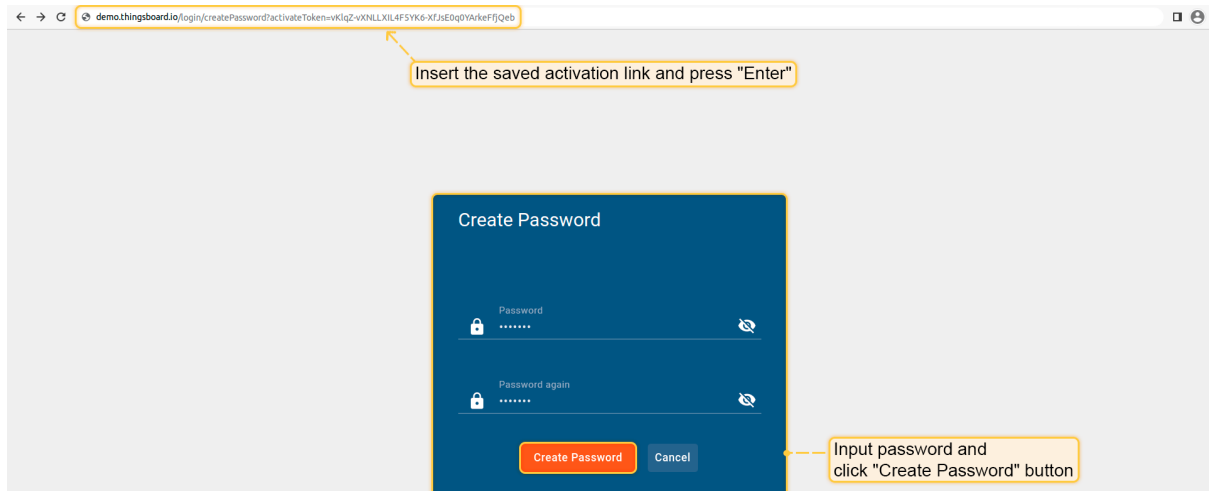


The sequence of screenshots illustrates the following steps:

- Add User:** A form is shown with fields for Email, First Name, Last Name, Phone, Description, and Activation method. Callout 4 points to the form fields, and callout 5 points to the 'Add' button.
- User activation link:** A modal displays a long activation link. Callout 6 points to the 'Copy activation link' button, and callout 7 points to the 'OK' button.
- My New Customer: Customer Users:** A table lists users. Callout 8 points to a user entry, and callout 9 points to the 'Enter Edit mode' button.
- User details:** A detailed view of the user's profile. Callout 10 points to the 'Default dashboard' dropdown, callout 11 points to the 'Always fullscreen' checkbox, and callout 12 points to the 'Apply changes' button.

- Paste the previously copied link into a new browser tab and press Enter. Come up with and enter a password twice, then press "Create Password". You will automatically login as a customer user;

- You have logged in as a Customer User. You may browse the data and acknowledge/clear alarms.



Set Threshold value:

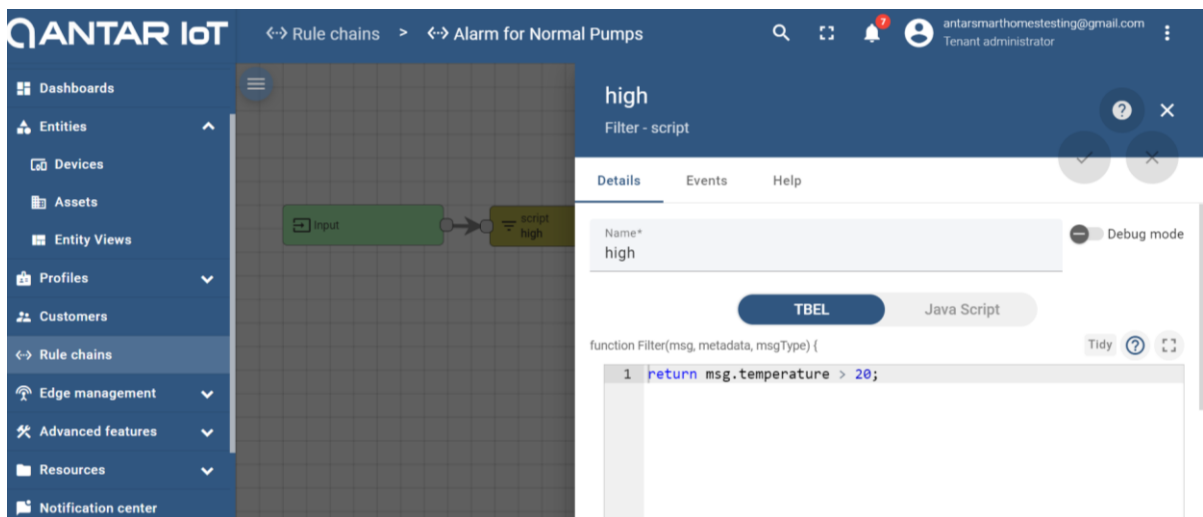
Step-01: Go to Rule Chains. And navigate to Alarm for large, Normal, small pump.

Step-02: Click on to script block and click on edit button. Write function on filter function according to your threshold values.

Eg: `return (msg.BR_Volatge > 220) || (msg.RY_Voltage >230) || (msg.RY_Voltage>240);`

In code for OR -> `||`

For and -> `&&`



Mail Setup in Thingsboard

In order to use Gmail, you will need to enable two-step verification

Follow the below steps to configure.

Google Account Search Google Account

Home Personal info Data & privacy **Security** People & sharing Payments & subscriptions About

1 Go to your Google Account and navigate to the 'Security' page

2 Select '2-Step Verification' tab

How you sign in to Google

Make sure you can always access your Google Account by keeping this information up to date

2-Step Verification	On since Nov 8, 2022	>
Password	Last changed Oct 24, 2022	>
Google prompt	1 device	>
2-Step Verification phones	093 212 2519	>
Recovery phone	093 212 2519	>
Recovery email	Verify johndoe@gmail.com	>

You can add more sign-in options

Security keys Authenticator Backup 2-Step Verification phones Backup codes

Your devices

Where you're signed in

1 session on Linux computer
Linux

Third-party apps with account access

You gave these sites and apps access to some of your Google Account data, including info that may be sensitive. Remove access for those you no longer trust or use.

2-Step Verification

Use the Authenticator app to get verification codes at no charge, even when your phone is offline. Available for Android and iPhone.

Security Key

A security key is a verification method that allows you to securely sign in. These can be built in to your phone, use Bluetooth, or plug directly into your computer's USB port.

Devices that don't need a second step

You can skip the second step on devices you trust, such as your own computer.

Devices you trust

Revoke trusted status from your devices that skip 2-Step Verification.

REVOKE ALL

App passwords

App Passwords aren't recommended and are unnecessary in most cases. To help keep your account secure, use "Sign in with Google" to connect apps to your Google Account.

App passwords
None

3 At the bottom of the page select 'App passwords'

Google Account

App passwords

App passwords let you sign in to your Google Account from apps on devices that don't support 2-Step Verification. You'll only need to enter it once so you don't need to remember it. [Learn more](#)

You don't have any app passwords.

Select the app and device you want to generate the app password for.

Select app

Select device

GENERATE

4 Select 'Other' app

Mail
Calendar
Contacts
YouTube
Other (Custom name)

← App passwords

App passwords let you sign in to your Google Account from apps on devices that don't support 2-Step Verification. You'll only need to enter it once so you don't need to remember it. [Learn more](#)

You don't have any app passwords.

Select the app and device you want to generate the app password for.

5 Enter app name → Thingsboard X

6 GENERATE → Click 'Generate' button

Generated app password

Your app password for your device

bqtk xvie ckal kgqw → Copy and save app password

How to use it

Go to the settings for your Google Account in the application or device you are trying to set up. Replace your password with the 16-character password shown above. Just like your normal password, this app password grants complete access to your Google Account. You won't need to remember it, so don't write it down or share it with anyone.

DONE

Email
securesally@gmail.com

Password
••••••••••

Once this is ready, you should be able to setup mail server using the information below:

- SMTP host: **smtp.gmail.com**;
- SMTP port: **465**;
- Username: **antarsmarthomestesting@gmail.com**;
- Password: previously created **app password**.

You can also enable/disable TLS using checkbox.

Click '**Send test mail**' button.

The screenshot shows the 'Mail Server' configuration page in the ThingsBoard interface. The left sidebar contains navigation links: Home, Tenants, Tenant profiles, Resources, Notification center, Settings (selected), and Security. The main content area is titled 'Outgoing Mail Server Settings' and includes a green success message: 'Test mail was successfully sent! Close'. The configuration fields are as follows:

- Mail From***: ThingsBoard<name@company.com> (Annotation 1: Update "Mail From" field)
- SMTP protocol**: SMTPS (Annotation 2: Select SMTP protocol)
- SMTP host***: smtp.gmail.com (Annotation 3: Specify SMTP host and port)
- SMTP port***: 465
- Timeout (msec)***: 10000
- Enable TLS**: ☒ (Annotation 4: Enable/disable TLS using checkbox)
- TLS version**: TLSv1.2
- Enable proxy**: ☐
- Username**: johndoe@thingsboard.io (Annotation 5: Enter your email as username)
- Change password**: ☒
- Password**: [masked] (Annotation 6: Enter previously copied app password)
- Buttons**: 'Send test mail' (Annotation 7: Click to send test mail) and 'Save' (Annotation 8: After you receive the test mail, save the Mail Server configuration)

Once you receive test mail on your email, save Mail Server configuration. In case of error in configuration, you should receive a popup with the error log.

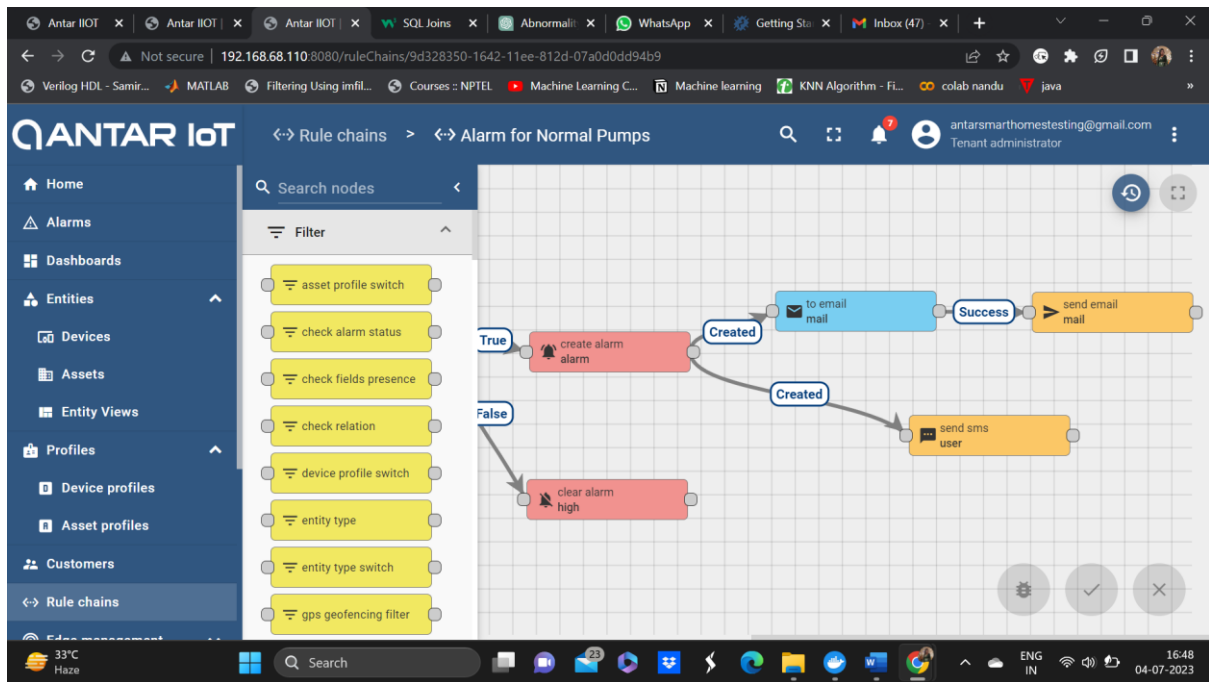
To setup mail for alarms:

Step-01: Go to Rule Chians and navigate to Alarm for large, small and normal pump.

Step-02: Click on to email block in the rule chain. change the to template and from template. And save.

The screenshot shows the 'Rule chains' editor in the ThingsBoard interface. The left sidebar contains navigation links: Dashboards, Entities, Devices, Assets, Entity Views, Profiles, Customers, Rule chains (selected), Edge management, and Advanced features. The main content area shows a rule chain with an 'Input' block connected to a 'script high' block. The 'mail' block is highlighted, and its configuration is shown in a modal window titled 'mail' with the subtitle 'Transformation - to email'. The configuration fields are as follows:

- Name***: mail
- From Template***: antarsmarthomestesting@gmail.com
- To Template***: jaswanthb@antariot.com
- Hint**: use \${metadataKey} for value from metadata, \${messageKey} for value from message body
- Buttons**: 'Details', 'Events', 'Help', and 'Debug mode' (toggle)



Mosquitto Broker Installation

Step-01: click the following link to download the Broker

<https://mosquitto.org/files/binary/win64/mosquitto-2.0.15-install-windows-x64.exe>

Step-02: Open services a check for status of mosquito broker and start the service.

Step-03: Create a text file with name **passwd** in the desktop. In format
Username:password

Ex: AntarloT:Antar@1234

Step-04: drag this folder on to the following location.

C:\Program Files\mosquitto

Step-05: Edit the conf file which is present in the following location C:\Program Files\mosquitto.

Add the following code to the conf file at the end

port 1883

listener 9001

protocol websockets

socket_domain ipv4

allow_anonymous false

password_file C:\Program Files\mosquitto\passwd.txt

Step-06: open cmd as administrator and enter the following cmd.

mosquitto_passwd -U password.txt

netsh advfirewall firewall add rule name="mqttTcpPort" dir=in action=allow
protocol=TCP localport=1883

Through this cmd the password will be changed to hash.

```

C:\Windows\System32>cd/

C:\>cd program files

C:\Program Files>cd mosquitto

C:\Program Files\mosquitto>dir
Volume in drive C has no label.
Volume Serial Number is 5CD6-A842

Directory of C:\Program Files\mosquitto

17-07-2023  11:16    <DIR>          .
08-07-2023  12:12    <DIR>          ..
16-08-2022  19:04                230 aclfile.example
16-08-2022  19:04            135,368 ChangeLog.txt
30-06-2023  13:24    <DIR>          devel
16-08-2022  19:04            1,568 edl-v10
16-08-2022  19:04            14,197 epl-v20
06-07-2022  03:13        3,415,552 libcrypto-1_1-x64.dll
06-07-2022  03:13        685,056 libssl-1_1-x64.dll
16-08-2022  19:04            40,449 mosquitto.conf
16-08-2022  19:05            87,040 mosquitto.dll
16-08-2022  19:11        382,464 mosquitto.exe
16-08-2022  19:05            18,432 mosquittopp.dll
16-08-2022  19:05            76,288 mosquitto_ctrl.exe
16-08-2022  19:07        122,880 mosquitto_dynamic_security.dll
16-08-2022  19:04            22,528 mosquitto_passwd.exe
16-08-2022  19:05            51,712 mosquitto_pub.exe
16-08-2022  19:05            79,872 mosquitto_rr.exe
16-08-2022  19:05            81,920 mosquitto_sub.exe
16-08-2022  19:04            1,886 NOTICE.md
17-07-2023  11:15            16 password.txt
16-08-2022  19:04            355 pwfile.example
16-08-2022  19:04            939 README-letsencrypt.md
16-08-2022  19:04            2,453 README-windows.txt
16-08-2022  19:04            3,768 README.md
30-06-2023  13:24        66,085 Uninstall.exe
                23 File(s)          5,291,058 bytes
                3 Dir(s)      395,962,576,896 bytes free

```

```

C:\Program Files\mosquitto>mosquitto_passwd -U password.txt

C:\Program Files\mosquitto>netsh advfirewall firewall add rule name="mqttTcpPort" dir=in action=allow protocol=TCP localport=1883
Ok.

C:\Program Files\mosquitto>ipconfig

Windows IP Configuration

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 10:

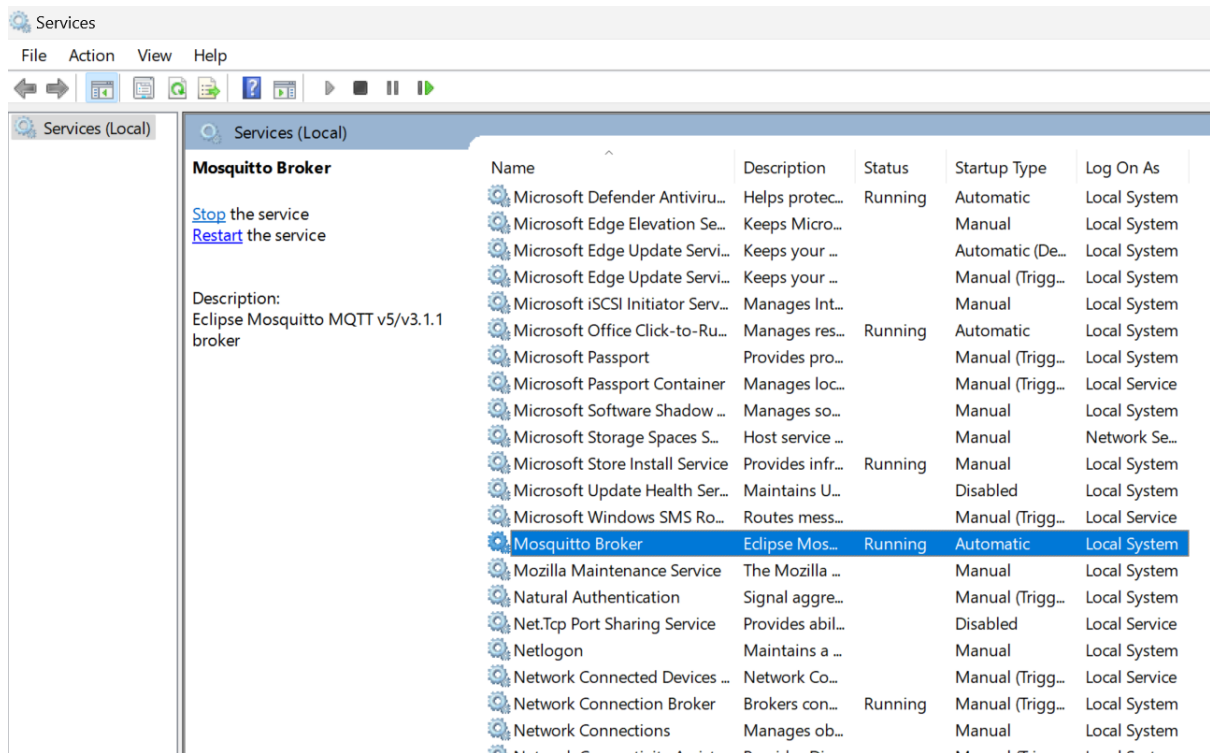
    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::485:4069:7549:8c9e%8
    IPv4 Address. . . . . : 192.168.68.113

```

Step-07: go the services and restart the mosquito service.



MQTT Explorer

Step-01: install using the following link

<https://github.com/thomasnordquist/MQTT-Explorer/releases/download/0.0.0-0.4.0-beta1/MQTT-Explorer-Setup-0.4.0-beta1.exe>

Step-02: Setup the MQTT Connection by below details and connect

Host -> mqtt.eclipse.org

Port -> 1883

Username -> antariot

Password -> admin@1234

