Data Analytics on Elasticsearch for IoT

University of Camerino
School of Science and Technology
Data Analytics Course

Students:
Giulia Morelli
Jessica Piccioni

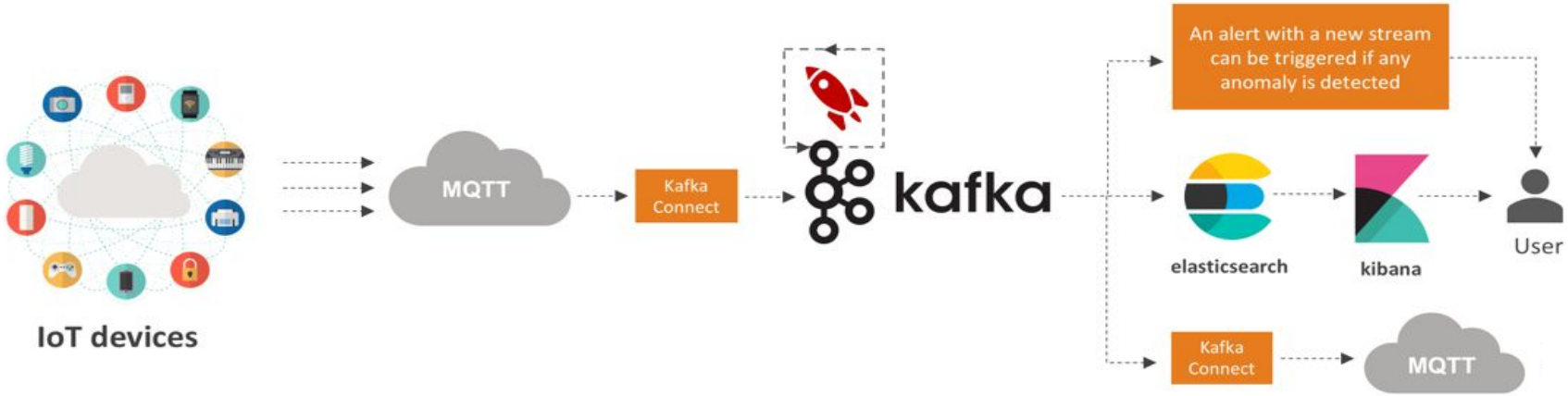
Supervisor:
Prof. Massimo Callisto

TABLE OF CONTENTS

PROJECT DESCRIPTION

PROJECT DESCRIPTION

The project aims to monitor IoT data coming from the sensors installed in the Computer Science department and classrooms of Camerino University. Data must be ingested with Kafka, sent in real time to Elasticsearch and displayed on a visualization dashboard defined in Kibana.



OBJECTIVES

The **main objective** of the project is to create an IoT data visualization dashboard based on Elasticsearch

Take MQTT data in real time from the Filippetti sensors and send it to a Kafka broker

Filter IoT data in order to make it understandable in Elasticsearch

Create a dashboard that displays the environmental data of the Filippetti sensors in Kibana.

CRITICAL POINTS

The **critical points** of the project were the following:

Taking sensor data using Filebeat with Logstash or Confluent Platform with Kafka?

Filtering data on Confluent Platform or on Elasticsearch?

What kind of data should be displayed on the dashboard?

OUR SOLUTION

The **solutions** of the critical points of the project were the following:

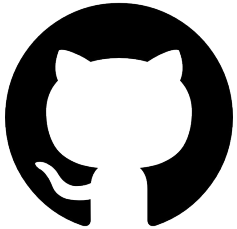
Confluent Platform is the most efficient solution.
Filebeat does not support
WSS protocol

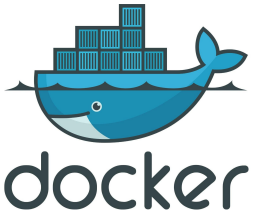
Filter the data with KsqlDB in Confluent Platform

Creating a dashboard showing the most significant data gathered in the classroom

TECHNOLOGIES

TECHNOLOGIES





- **Ubuntu 21.04 64 bit**
- **Java 11.0**
- **PuTTY 0.74**
- **Docker 20.10**
- **Elasticsearch 7.10**
- **Kibana 7.10**
- **KsqlDb 0.17**
- **Github**
- **Confluent Platform 6.2**

TECHNOLOGIES

Confluent Platform allows us to make a streaming analytics and filtering data that comes from FIippetti broker.

Elasticsearch allows us to save the data that comes from the kafka broker.

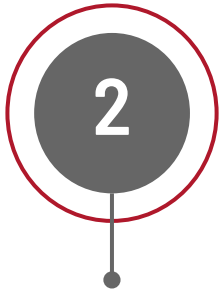
Kibana is used to create the dashboard for visualizing the data.

IMPLEMENTATION

IMPLEMENTATION PHASES

Set Elasticsearch and Kibana environment

Send the data from Kafka Broker to Elasticsearch and create Kibana Dashboard



Start Confluent Platform, create the topics, add the connector and filter mqtt data

1

SETTING ELASTIC AND KIBANA ENVIRONMENT

1) Start Elasticsearch and Kibana with docker-compose

```
docker-compose up
```

2) Create an index in Elasticsearch

```
curl -X PUT http://hostname:9200/index_name
```

1

SETTING ELASTIC AND KIBANA ENVIRONMENT

3) Create a Pipeline in Elasticsearch to convert the date field from a string format into a data format

```
curl -X PUT
"hostname:9200/_ingest/pipeline/my-pipeline?pretty" -H
'Content-Type: application/json' -d'
{
  "description" : "...",
  "processors" : [
    {
      "date" : {
        "field" : "TZ",
        "formats" : ["date_optional_time"],
        "timezone" : "Europe/London"
      }
    }
  ]
}'
```

4) Set the pipeline as default pipeline for the index

```
curl -X PUT
"hostname:9200/index_name/_settings" -H
'Content-Type: application/json' -d'
{
  "settings": {
    "index.default_pipeline": "my-pipeline"
  }
}'
```

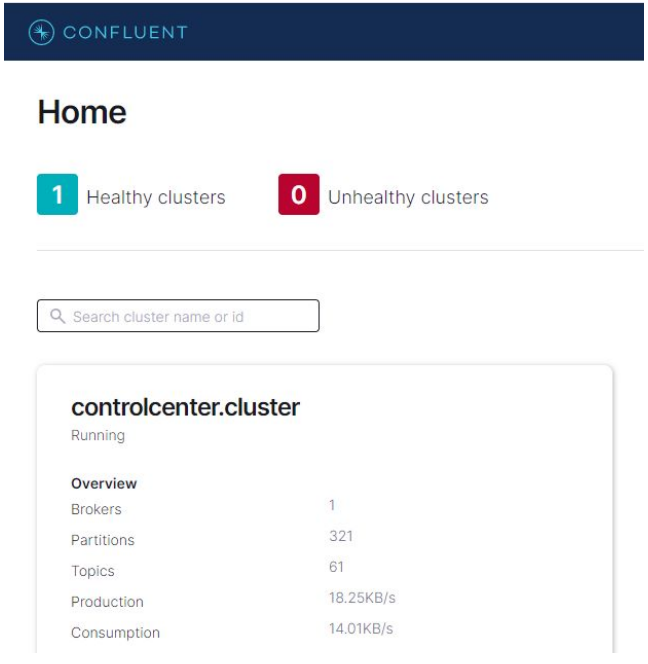
2

SETTING CONFLUENT PLATFORM ENVIRONMENT

1) Launch Confluent from any directory with the underneath command

```
confluent local services start
```

2) Once all services are [UP] we can proceed navigating to the Control Center web interface at http://hostname:9021/ and select the cluster



2

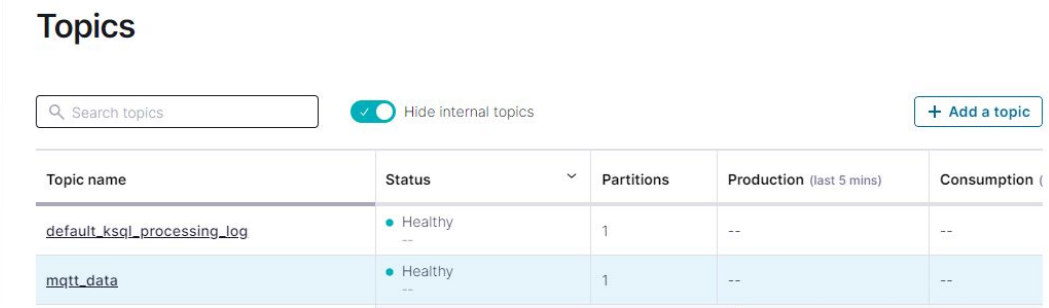
SETTING CONFLUENT PLATFORM ENVIRONMENT

3) Install the source and sink connectors with the commands

```
confluent-hub install
confluentinc/kafka-connect-mqtt:latest

confluent-hub install
confluentinc/kafka-connect-elasticsearch:latest
```

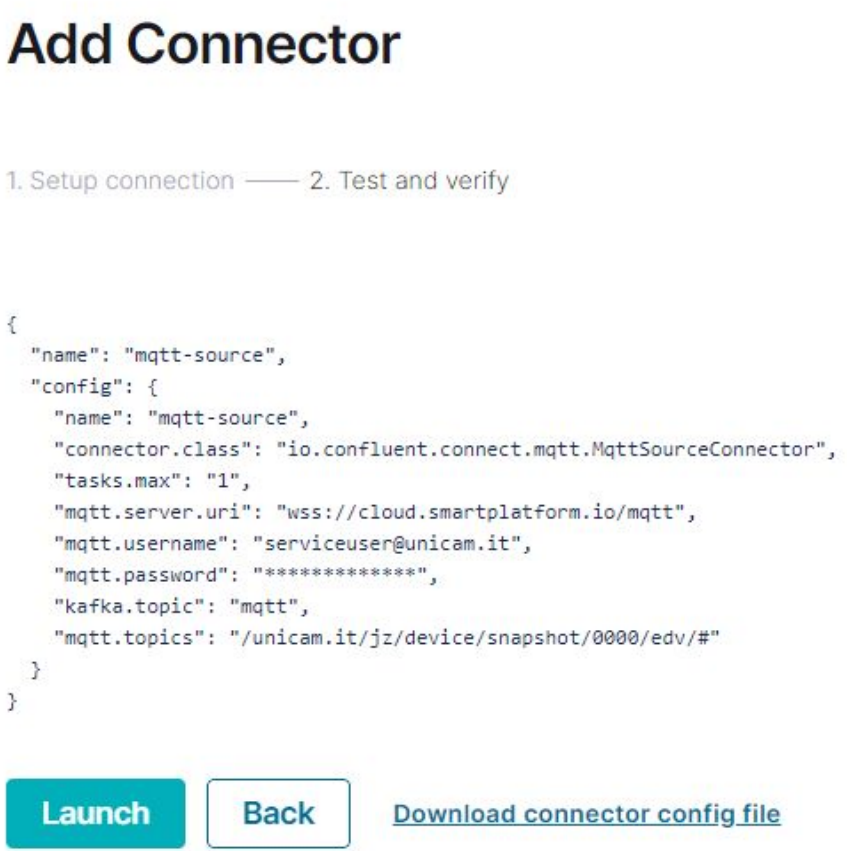
4) Create the topic you need



2

SETTING CONFLUENT PLATFORM ENVIRONMENT

5) Add the Mqtt source connector to take the data in real time from the Filippetti broker and put it into the new topic



2

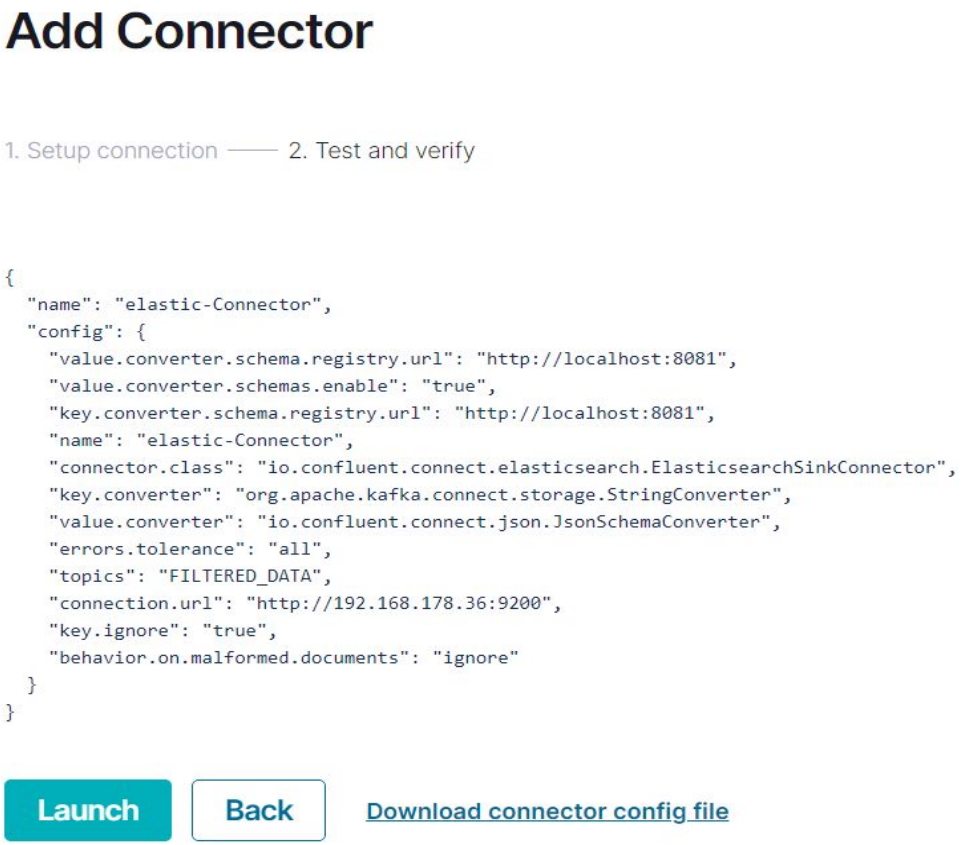
SETTING CONFLUENT PLATFORM ENVIRONMENT

6) Create and Write a Stream using KsqlDB to filter the mqtt data and create a new topic with the filtered data

3

SEND DATA TO ELASTICSEARCH AND CREATE DASHBOARD

1) Add the Elasticsearch sink connector to send filtered data to Elasticsearch.

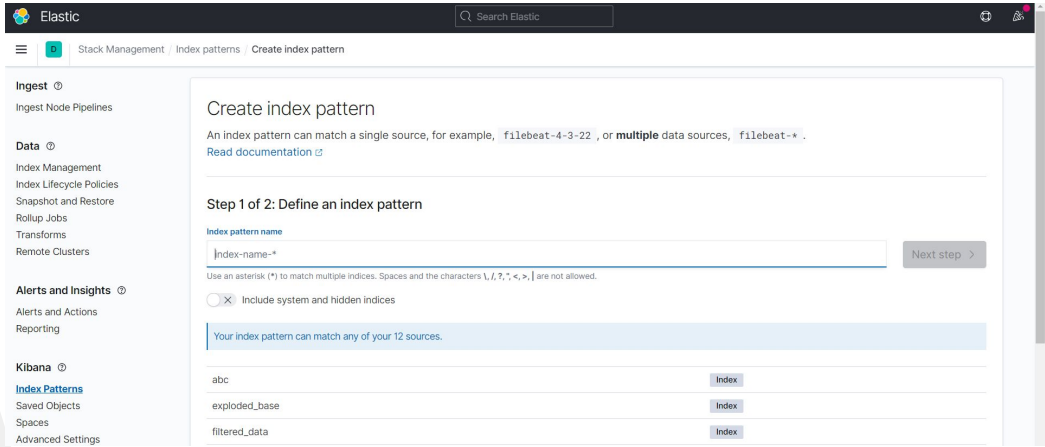


3

SEND DATA TO ELASTICSEARCH AND CREATE DASHBOARD

2) Open Kibana at http://hostname:5601/ and create the index pattern

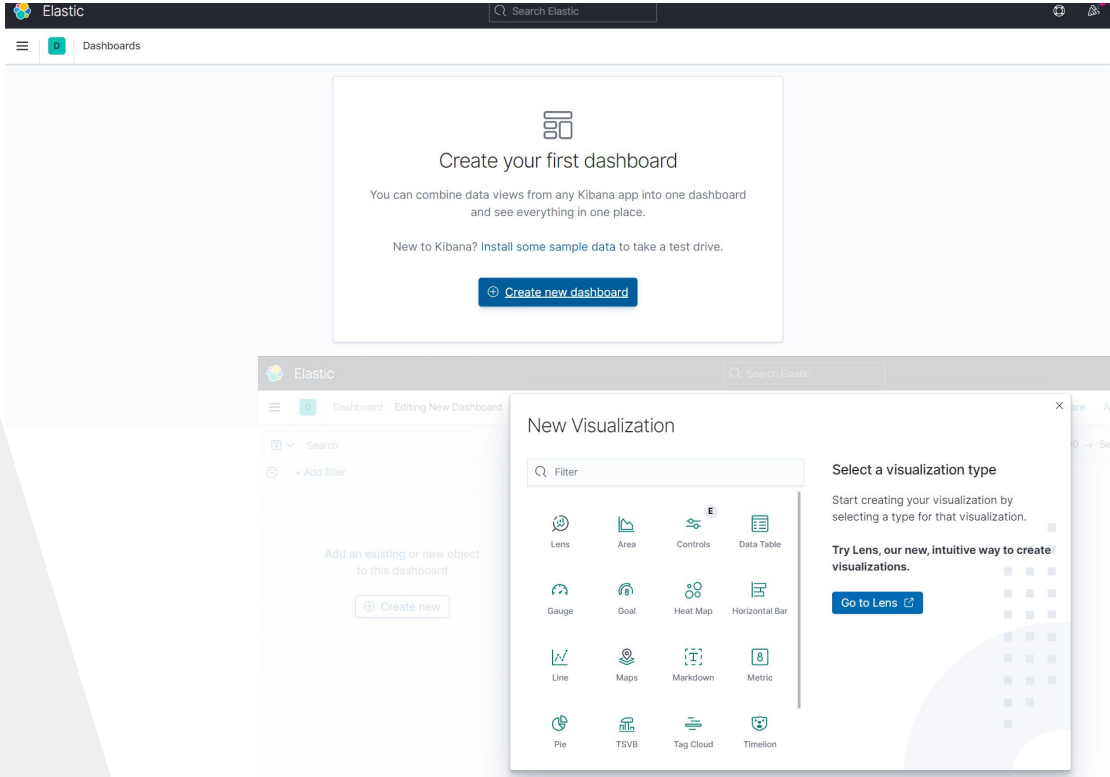
Stack Management -> Index Pattern -> Create index Pattern -> write the name with the same name of your index in which there are the sensors data -> select @timestamp as field.



3

SEND DATA TO ELASTICSEARCH AND CREATE DASHBOARD

3) After selecting the index pattern you can create your custom dashboard.



RESULTS AND FUTURE WORK

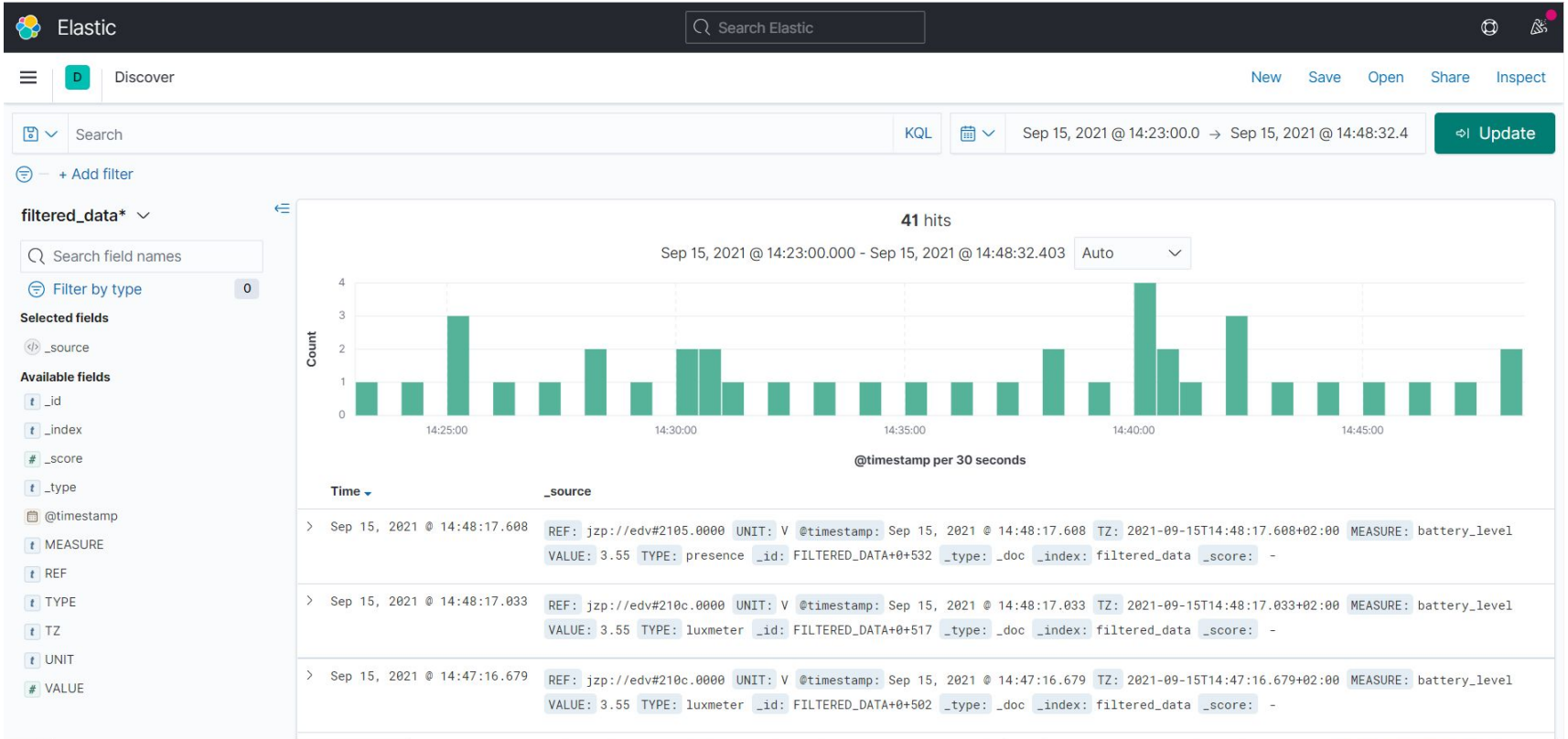
SENSORS DATA VS FILTERED DATA IN ELASTICSEARCH

```
{
"t": timestamp in secondi
"tz": timestamp in HHMMDDtHHMMSS
"uuid": identificativo univoco del messaggio
"cuid":
"ref": "jzp://edv#0503.0000" identificativo fisico del device
"type": "presence" tipo di messaggio del sensore
"cat": "0610"
"sn": "integer(0, 255)",
"m": [
 {
  "t": "nowTimestamp()",
  "tz": "now()",
  "k": "device_temperature" tipo di misura: temperatura del device
  "v": "double(0, 40)", valore della misura
  "u": "C" unità di misura
 },
```

200 - OK 128 ms

```
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 497,
      "relation" : "eq"
    },
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "filtered_data",
        "_type" : "_doc",
        "_id" : "FILTERED_DATA+0+0",
        "_score" : 1.0,
        "_source" : {
          "REF" : "jzp://edv#210c.0000",
          "UNIT" : "V",
          "@timestamp" : "2021-09-15T14:23:09.219+02:00",
          "TZ" : "2021-09-15T14:23:09.219+02:00",
          "MEASURE" : "battery_level",
          "VALUE" : 3.55,
          "TYPE" : "luxmeter"
        }
      },
```

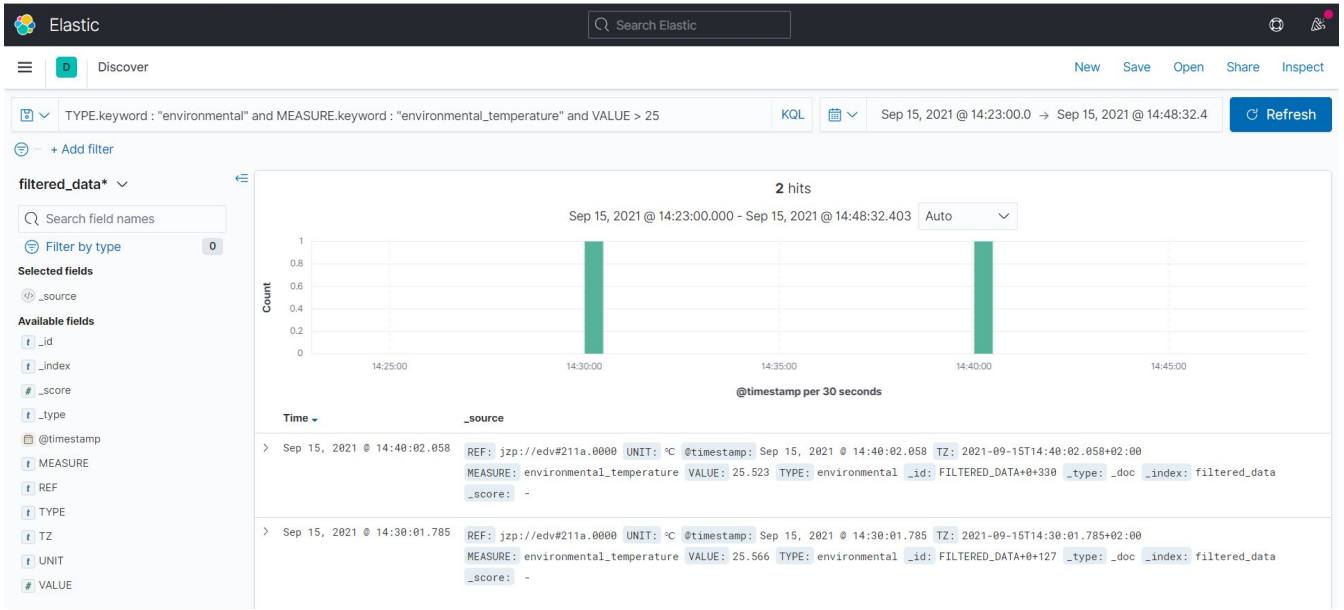
DISCOVER VIEW IN KIBANA



DISCOVER VIEW IN KIBANA

KQL QUERY 1

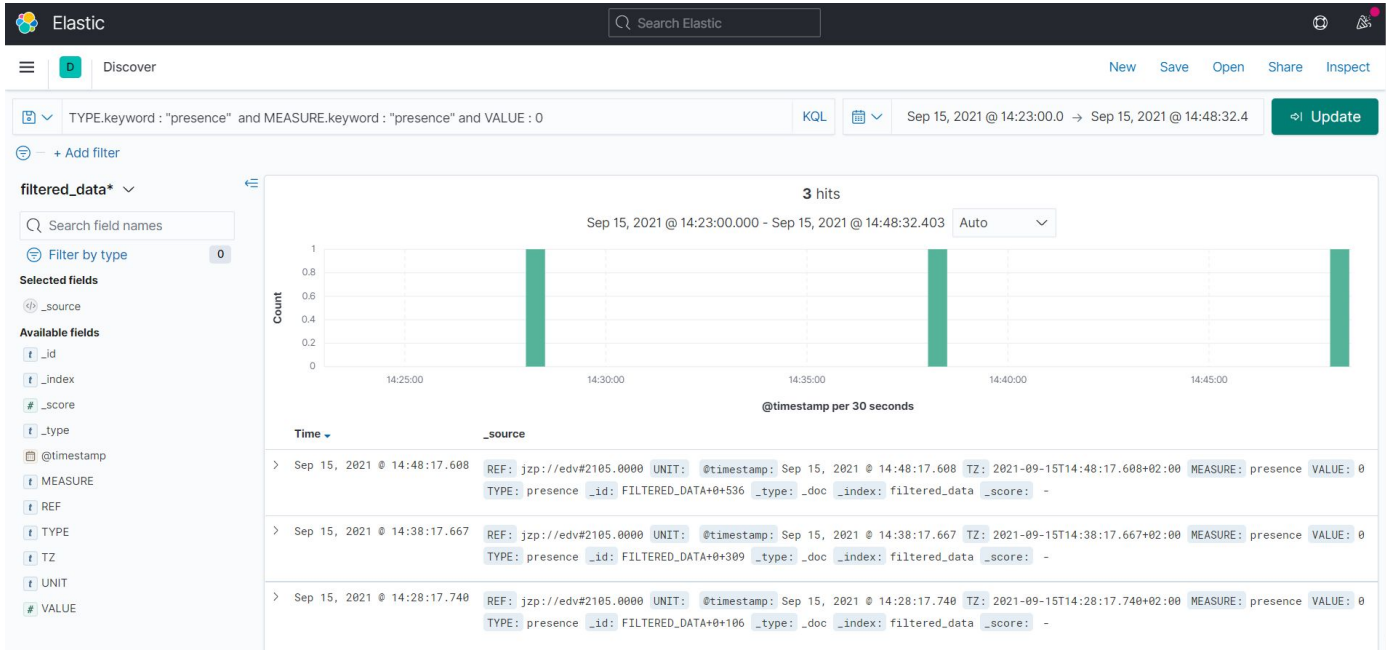
```
TYPE.keyword : "environmental" and
MEASURE.keyword : "environmental_temperature"
and VALUE > 25
```



DISCOVER VIEW IN KIBANA

KQL QUERY 2

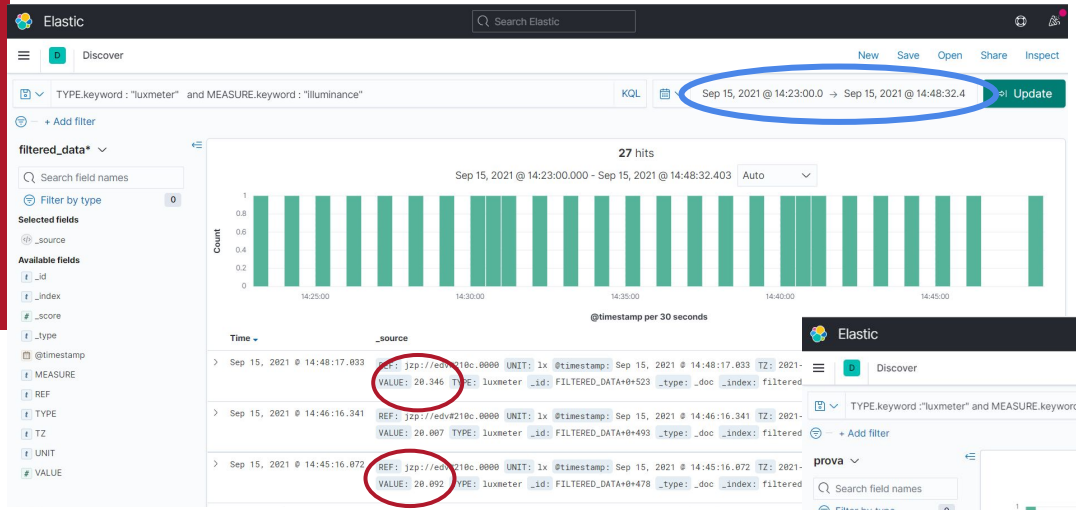
```
TYPE.keyword : "presence"  and
MEASURE.keyword : "presence" and VALUE : 0
```

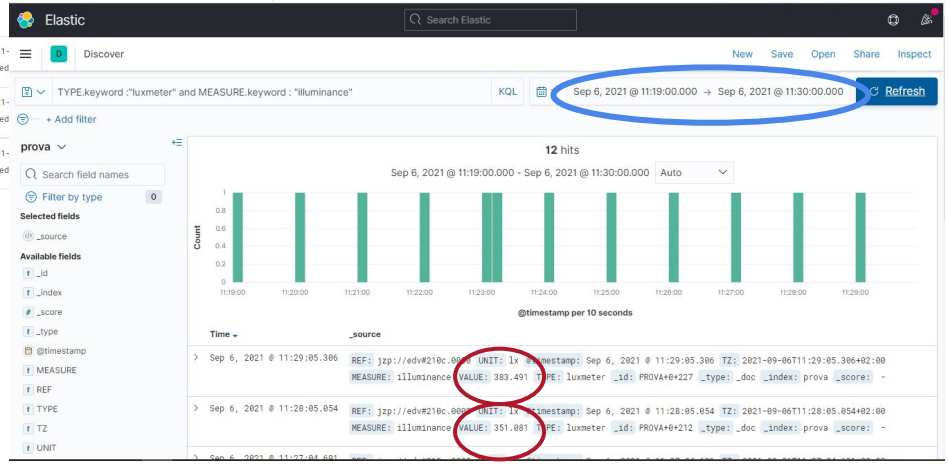


DISCOVER VIEW IN KIBANA

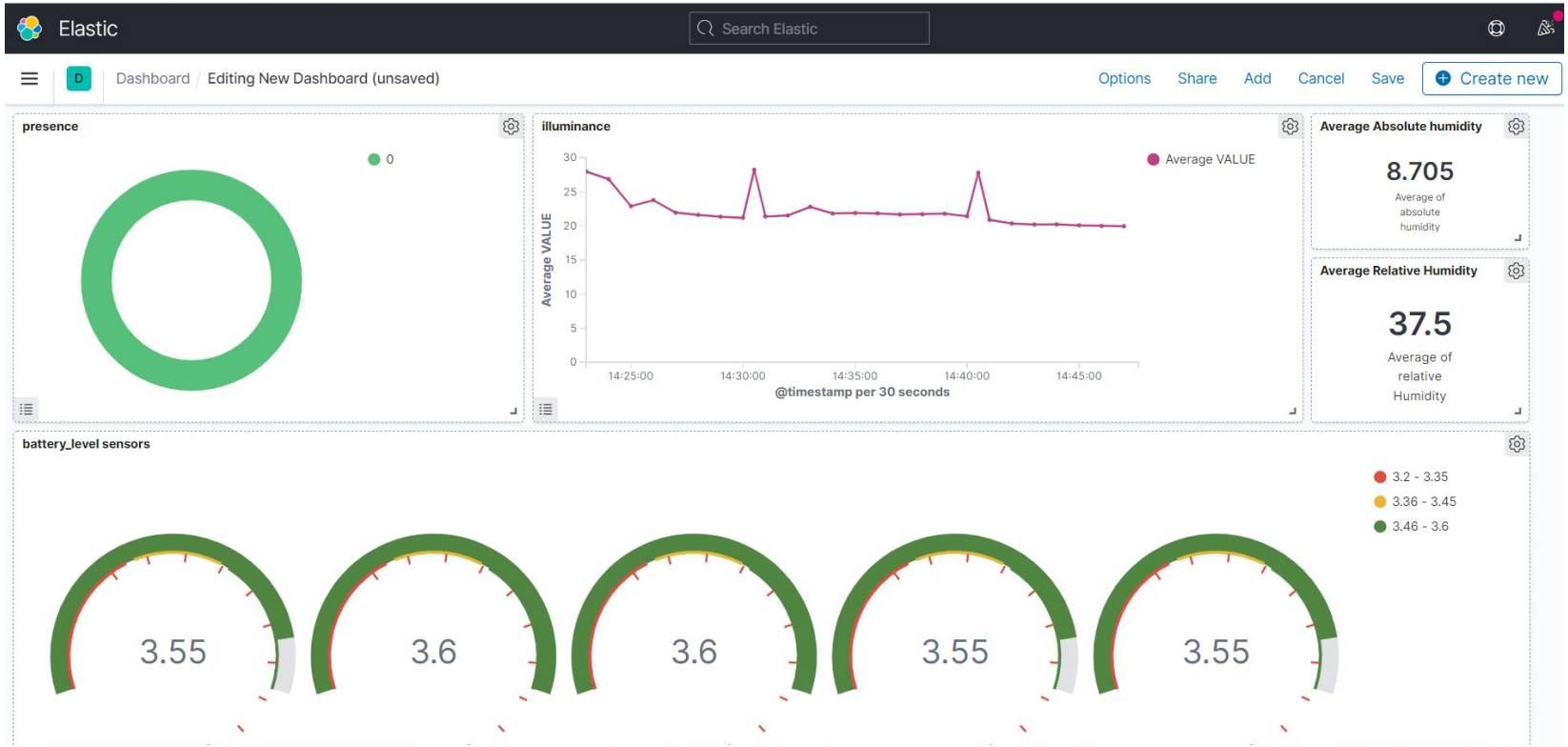
KQL QUERY 3

```
TYPE.keyword : "luxmeter"   and
MEASURE.keyword : "illuminance"
```





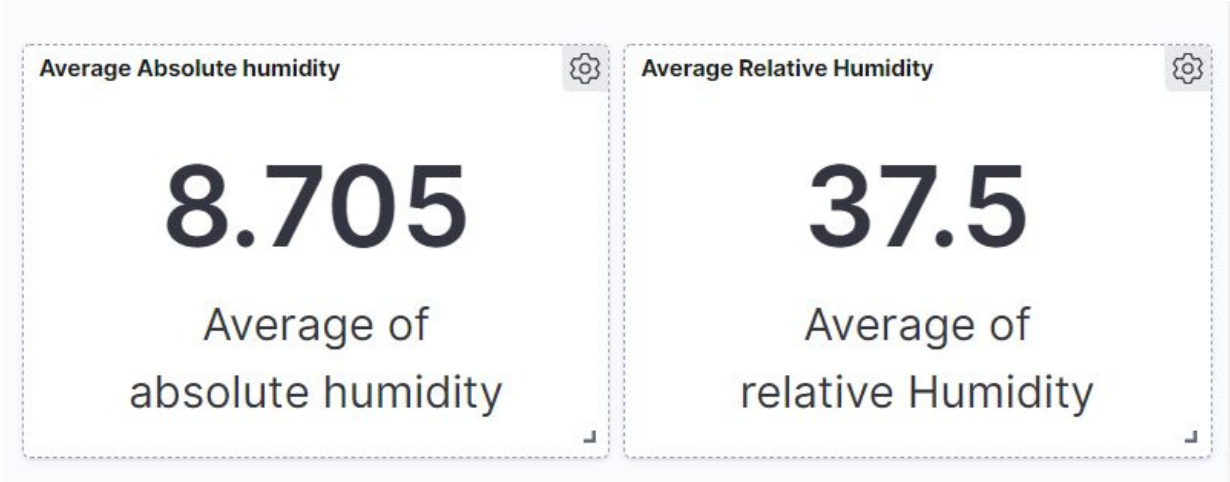
DASHBOARD IN KIBANA



DASHBOARD IN KIBANA

Graphic for the visualization of the absolute and relative humidity:

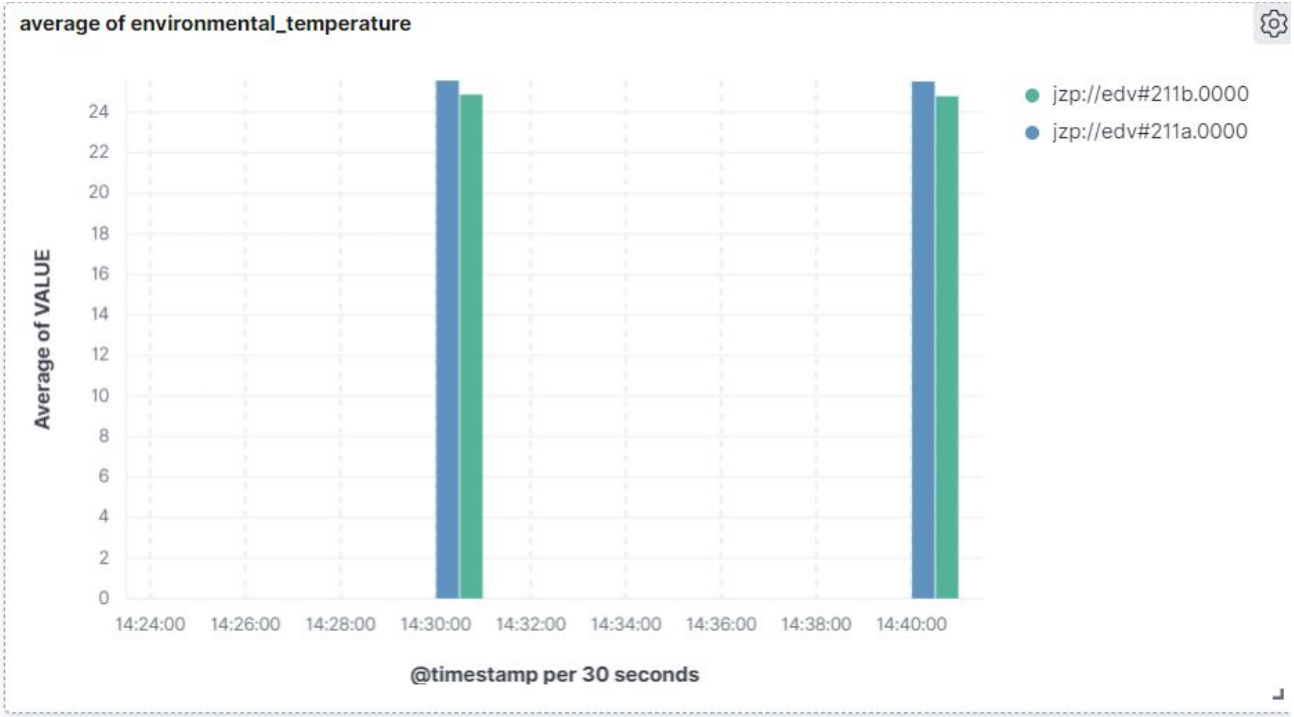
Use metric chart to display the absolute and relative humidity in the classrooms.



DASHBOARD IN KIBANA

Graphic for the visualization of the average of the environmental temperature in the classrooms

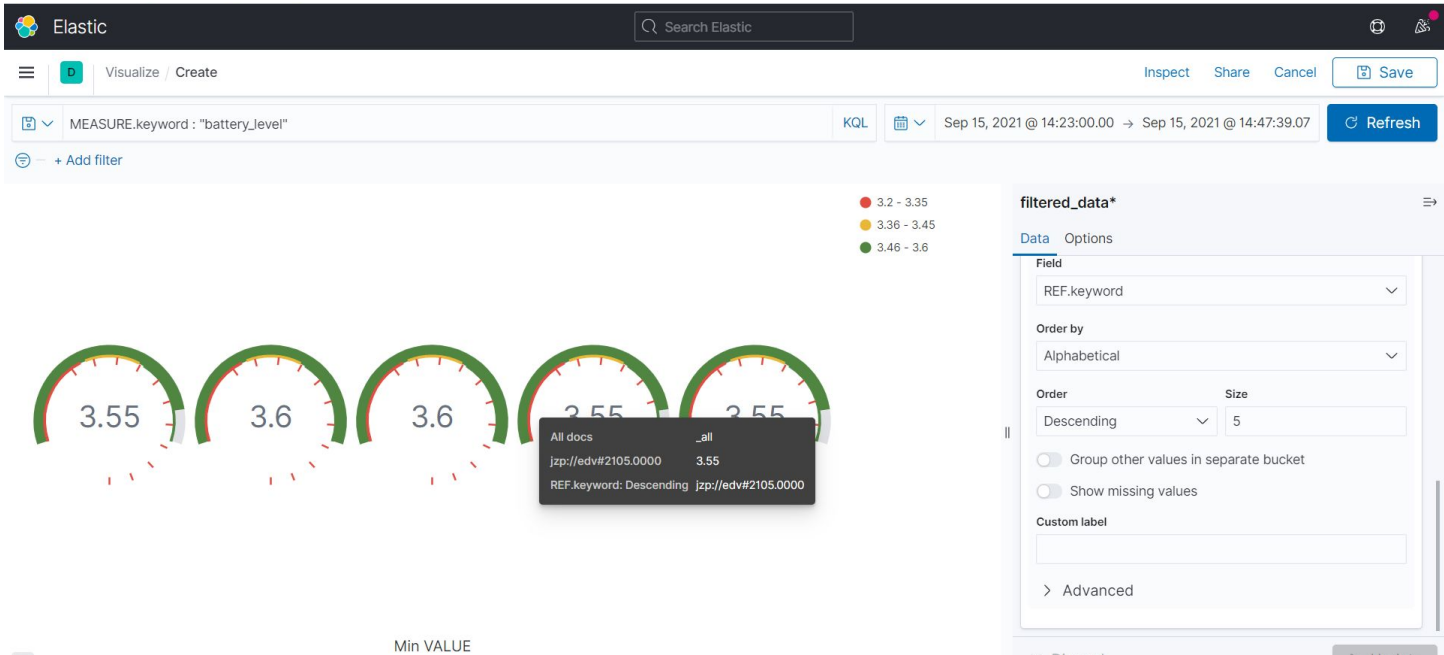
Use a Histogram chart to display the average of the environmental temperature in the classrooms, grouped by each sensor



DASHBOARD IN KIBANA

Graphic for the visualization of each sensor battery level

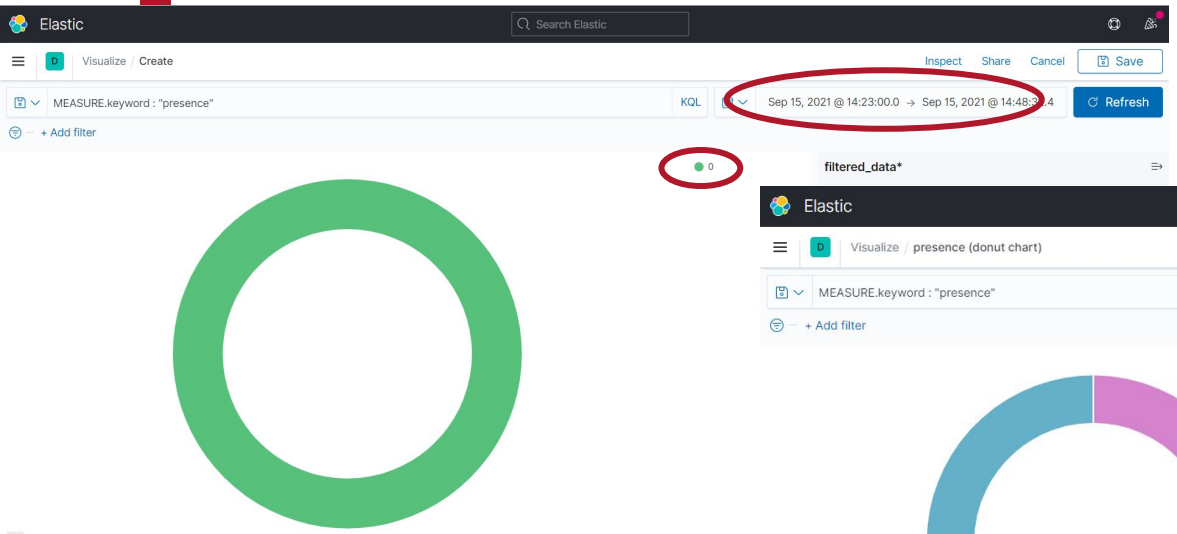
Use a Gauge graphic where we set 3 levels of battery with a color:
red (3.2-3.35) -> low battery level
yellow (3.36-3.45) -> medium battery level
green (3.46-3.6) -> high battery level

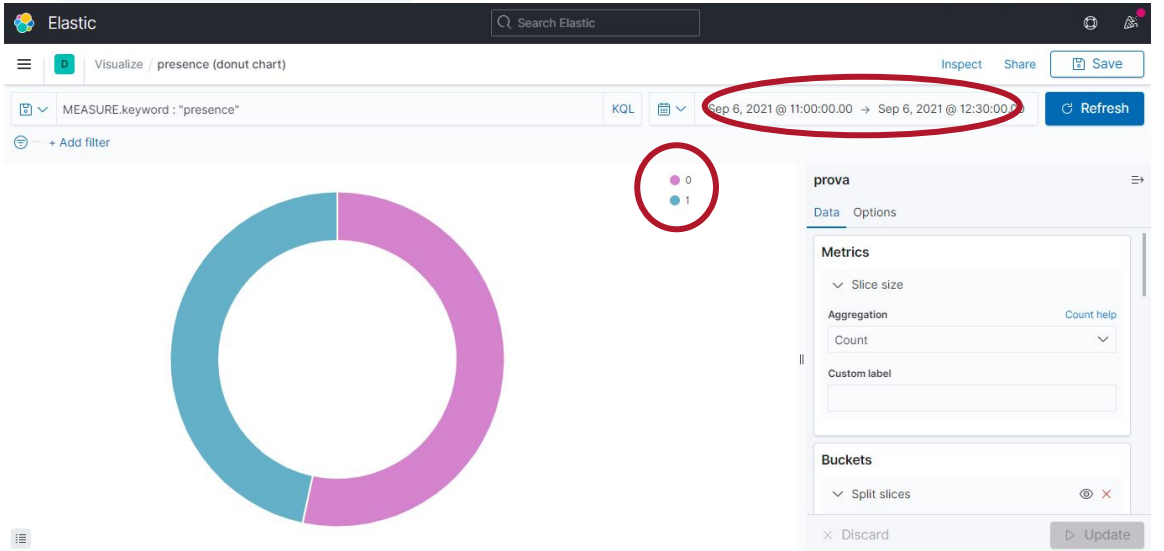


DASHBOARD IN KIBANA

Graphic for the visualization of the presence in the classrooms:

Use a Pie chart to show the presence in the classrooms. 0 is for no presence and 1 for the presence.

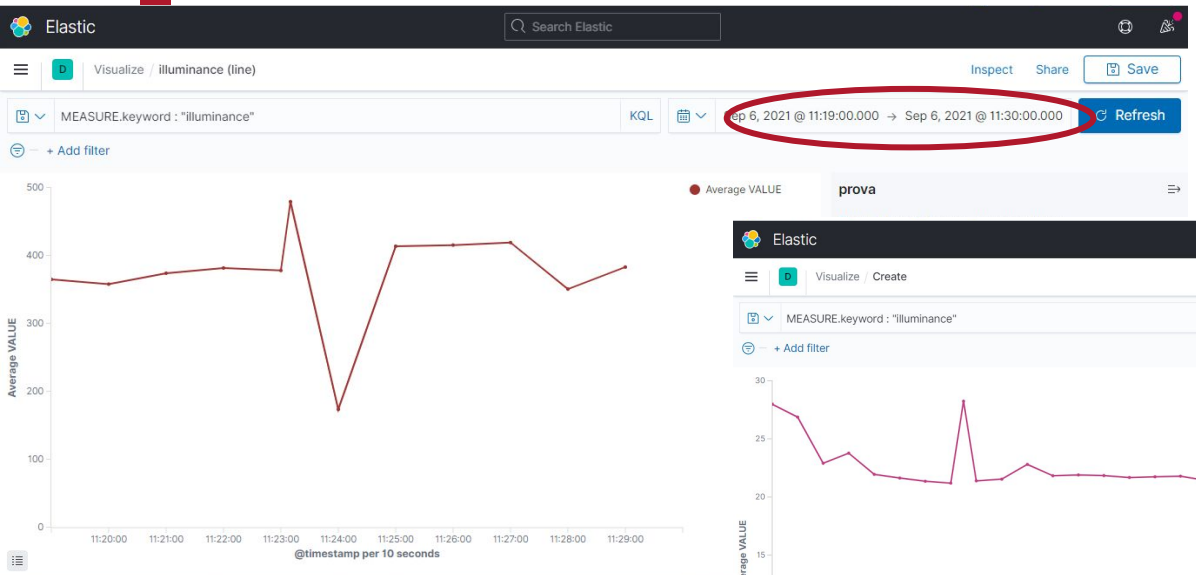


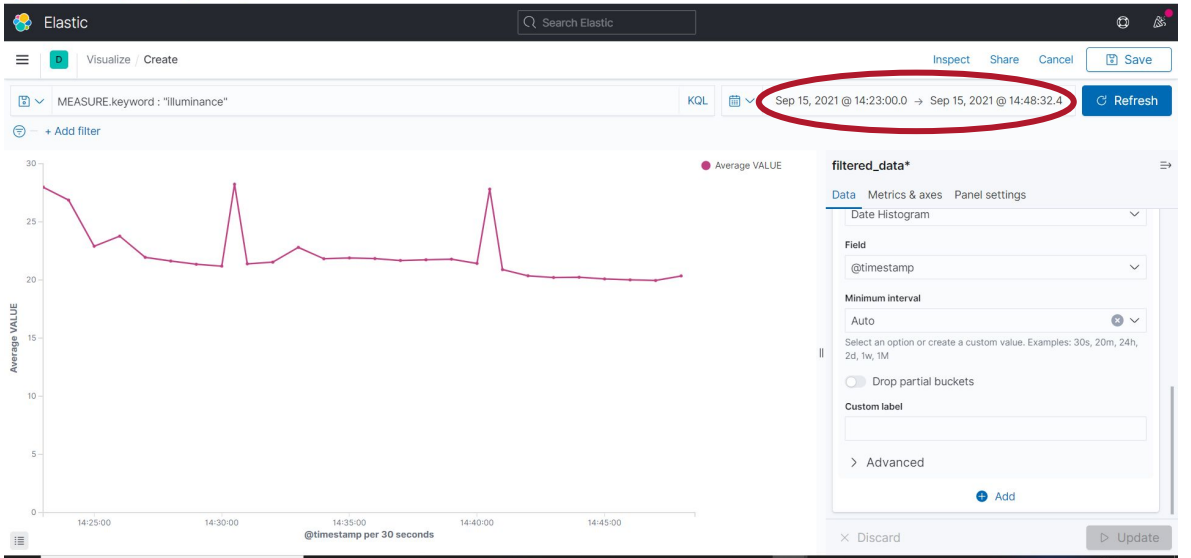


DASHBOARD IN KIBANA

Graphic for the visualization of the illuminance in the classrooms:

Use a Line chart to display illuminance in the classrooms.





FUTURE WORKS

Using Confluent Cloud with Elastic Cloud

Create new kind of dashboards in Kibana that focus on specific data.

If Confluent Platform supports the version 0.20 of KsqlDB, use directly the type timestamp without using the pipeline

Thanks For The Attention

Does anyone have any questions?


giulia.morelli@stundeti.unicam.it
jessica.piccioni@studenti.unicam.it


https://github.com/jessicapicc/elk-IoT