Projekat 3

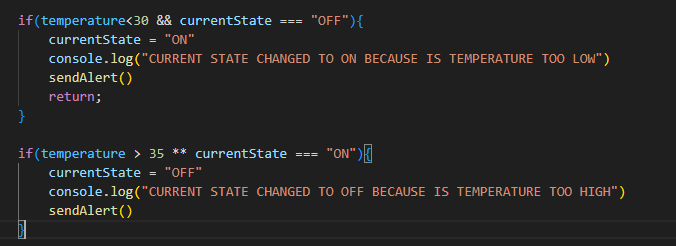
Cilj ovog projekta je da se kreiraju dva mikroservisa u različitim tehnologijama, kao i njihova integracija sa EdgeX platformom. Ovi mikroservisi predstavljaju northbound infrastrukturu na fog i cloud strani ove platforme. Ovi servisi su startovani kao Docker kontejneri.

# Visualization mikroservis

Ovaj mikroservis je implementiran u .NET-u 6.0. Njegov glavni zadatak je bio da dobije podatke od strane EdgeX-a preko MQTT protokola, pa je korišćen MQTT broker HiveMQ. Kada ovaj mikroservis primi podatke on treba da ih smesti u InfluxDB bazu podataka. Kako bi ovi podaci mogli da se vizuelizuju koristi se popularni alat za vizuelizaciju Grafana. Kao i InfluxDB i Grafana je pokrenuta kao Docker kontejner i povezana sa InfluxDB-em.

# Monitoring mikroservis

Ovaj mikroservis je implenetiran u Python-u. Njegov glavni zadatak je da dobija podatke na isti način kao i Visualization mikroservis i nakon prijema podataka on treba da primeni jednostavno pravilo za detekciju događaja na onsnovu tih podataka. Kada detektuje neki događaj on šalje poruku ili komandu mikroservisu na EdgeX platformi.

Ovaj mikroservis vrši proveru temperature da li prelazi određenu granicu:  


Pre pokretanja ovog servisa, potrebno je da pokrene jednostavan interfejs colorchanger. Interfejs colorchanger pokreće se sledećim komandama:

docker build -t colorChanger .  
docker run -d -p 5000:5000 --name colorChanger colorChanger:latest

# Python skripta

Ova skripta služi kao neki senzor na nekom od uređaja i vrši simuliranje slanja podataka na EdgeX platformu. Potrebno je da se startuje nakon što se pokrene ceo sistem i sve konfiguriše kako treba.

Komanda za pokretanje celog sistema je: docker-compose up -d  
Komanda za pokretanje skripte je: python script.py

# EdgeX konfiguracija

1. Na Data servisu, potrebno je izvrsiti HTTP POST metodu kako bi se kreirao ValueDescriptor:

adresa: http://localhost:48080/api/v1/valuedescriptor

body1:

{

"name": "ID",

"description": "data from .csv file",

"min": "",

"max": "",

"type": "String",

"uomLabel": "ID",

"defaultValue": "null",

"labels": [

"environment",

"ID"

]

}

body2:

{

"name": "timestamp",

"description": "data from .csv file",

"min": "",

"max": "",

"type": "String",

"uomLabel": "timestamp",

"defaultValue": "null",

"labels": [

"environment",

"timestamp"

]

}

body3:

{

"name": "temperature",

"description": "data from .csv file",

"min": "",

"max": "",

"type": "Int64",

"uomLabel": "temperature",

"defaultValue": "null",

"labels": [

"environment",

"temperature"

]

}

body4:

{

"name": "humidity",

"description": "data from .csv file",

"min": "",

"max": "",

"type": "Int64",

"uomLabel": "humidity",

"defaultValue": "null",

"labels": [

"environment",

"humidity"

]

}

2. Kreirati Device Profile pozivanjem HTTP POST metode na adresu http://localhost:48081/api/v1/deviceprofile/uploadfile

i selktovanjem fajla docker-profile.yaml

3. Treci korak je poteban da bi se kreirao device i to se radi slanjem POST zahteva na adresu http://localhost:48081/api/v1/device

Unutar body dela treba da se stavi sledece:

{

"name": "SensorValueCluster2",

"description": "Raspberry Pi sensor cluster",

"adminState": "unlocked",

"operatingState": "enabled",

"protocols": {

"HTTP": {

"host": "localhost",

"unitID": "1"

}

},

"labels": [

"Temperature sensor",

"DHT11"

],

"location": "Nis",

"service": {

"name": "edgex-device-rest"

},

"profile": {

"name": "SensorCluster2"

}

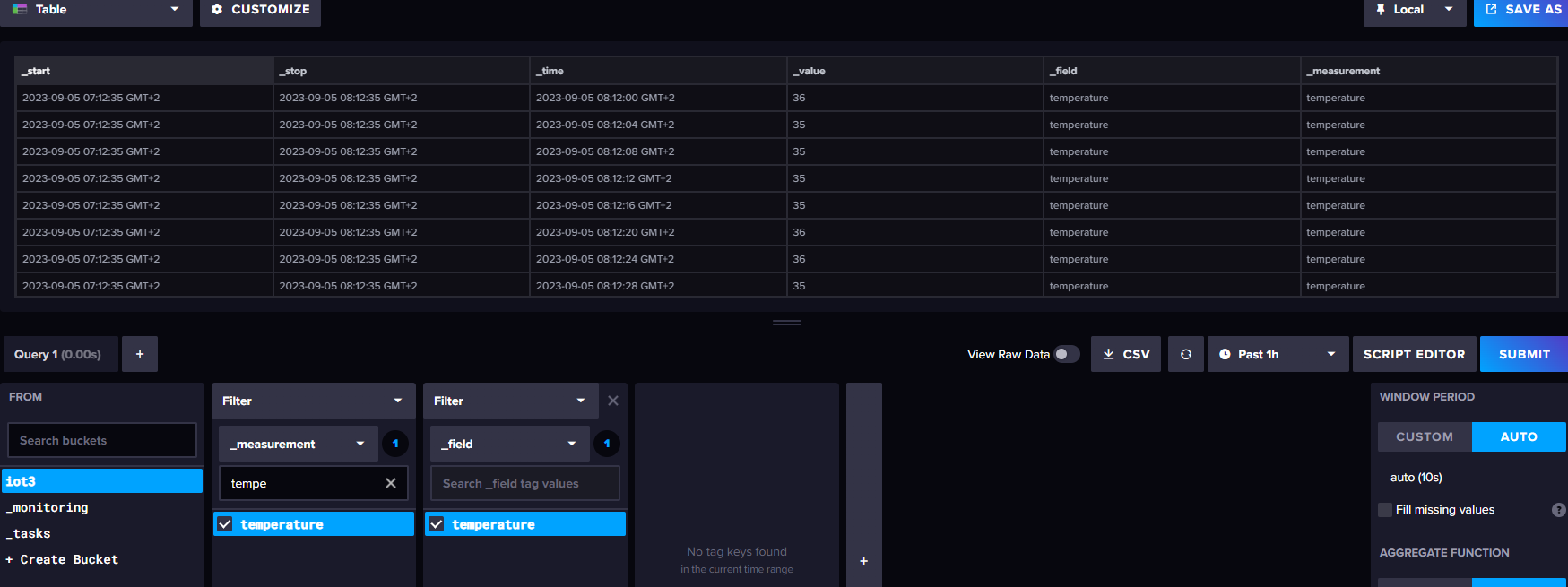
}

# InfluxDB

Nakon pokretanja docker-compose komande startuje se kontejner za InfluxDB na adresi 8086. Potrebno je da se zatim ode na adresu [**http://localhost:8086**](http://localhost:8086) i da se napravi novi nalog sa sledećim informacijama:

Username: admin  
Password: adminadmin  
Organization: organization  
Initial bucket: iot3

Sada je potrebno da se poveže InfluxDB sa Grafanom.



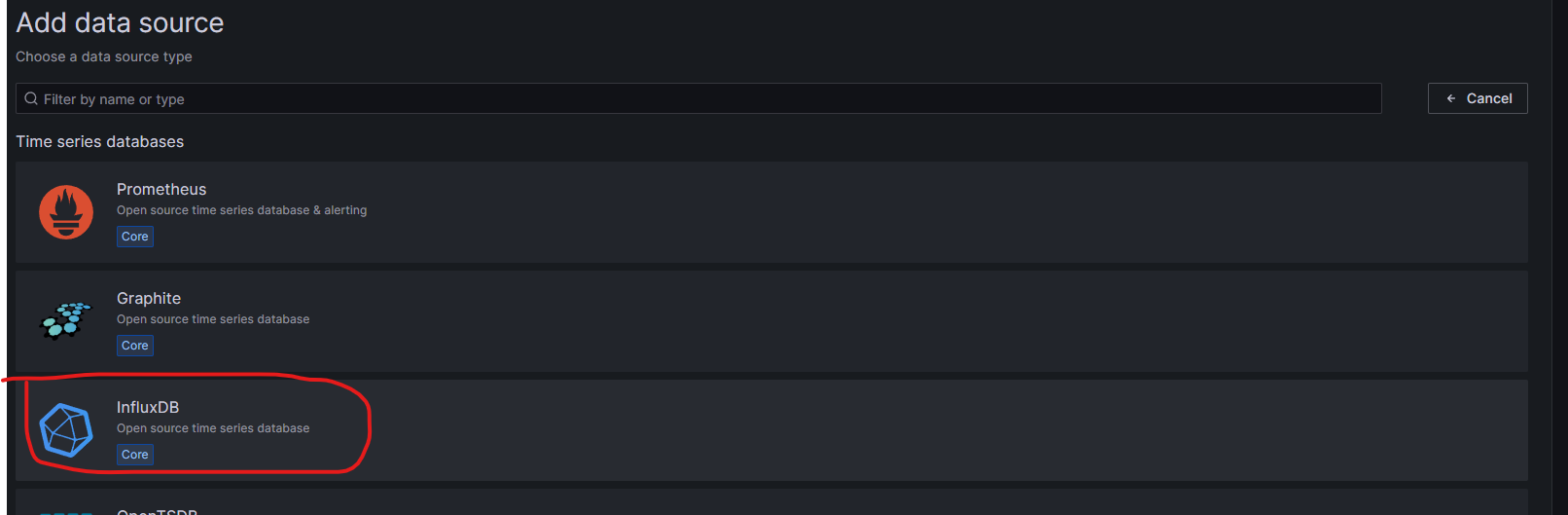
## Povezivanje InfluxDB i Grafane

Grafana se pokreće u kontejneru sa portom 3000, tako da je potrebno da se ode na adresu: [**http://localhost:3000**](http://localhost:3000).

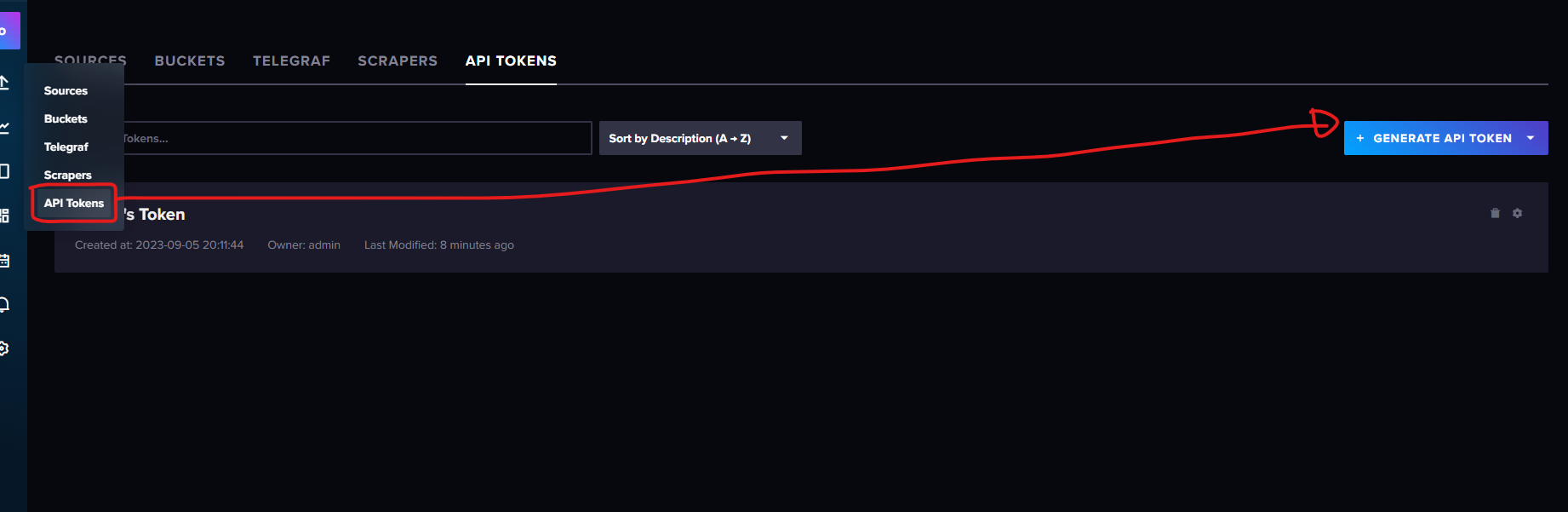
Kada se ode na ovoj adresi potrebno je da unesu sledeće informacije:

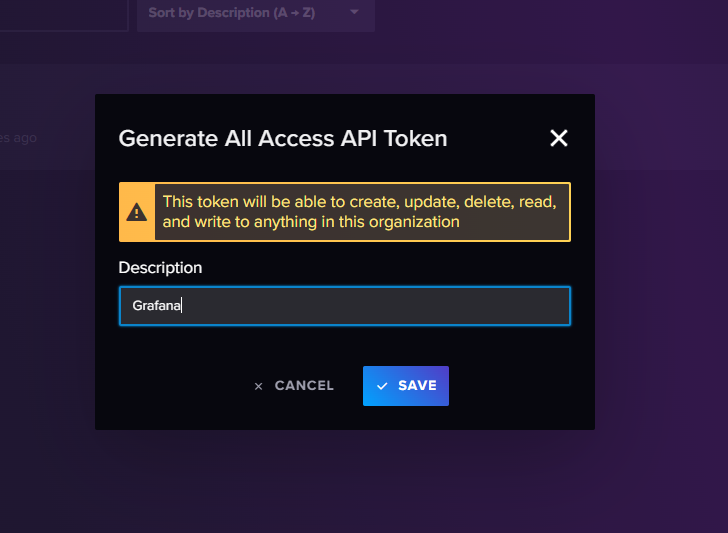
Username: admin  
Password: admin  
Zatim će biti ponuđeno da se promeni lozinka po želji ali taj korak može da se preskoči pritiskom na dugme Skip.

Nakon toga treba da se napravi konekcija sa InfluxDB-om tako što se ode na karticu Connections i izabere InfluxDB:

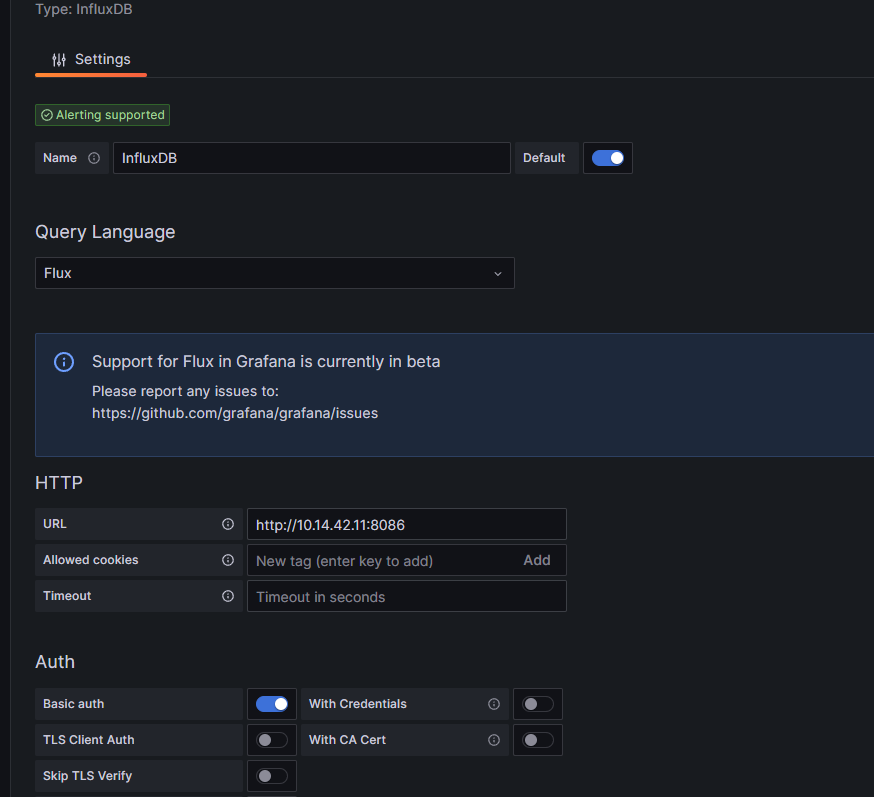
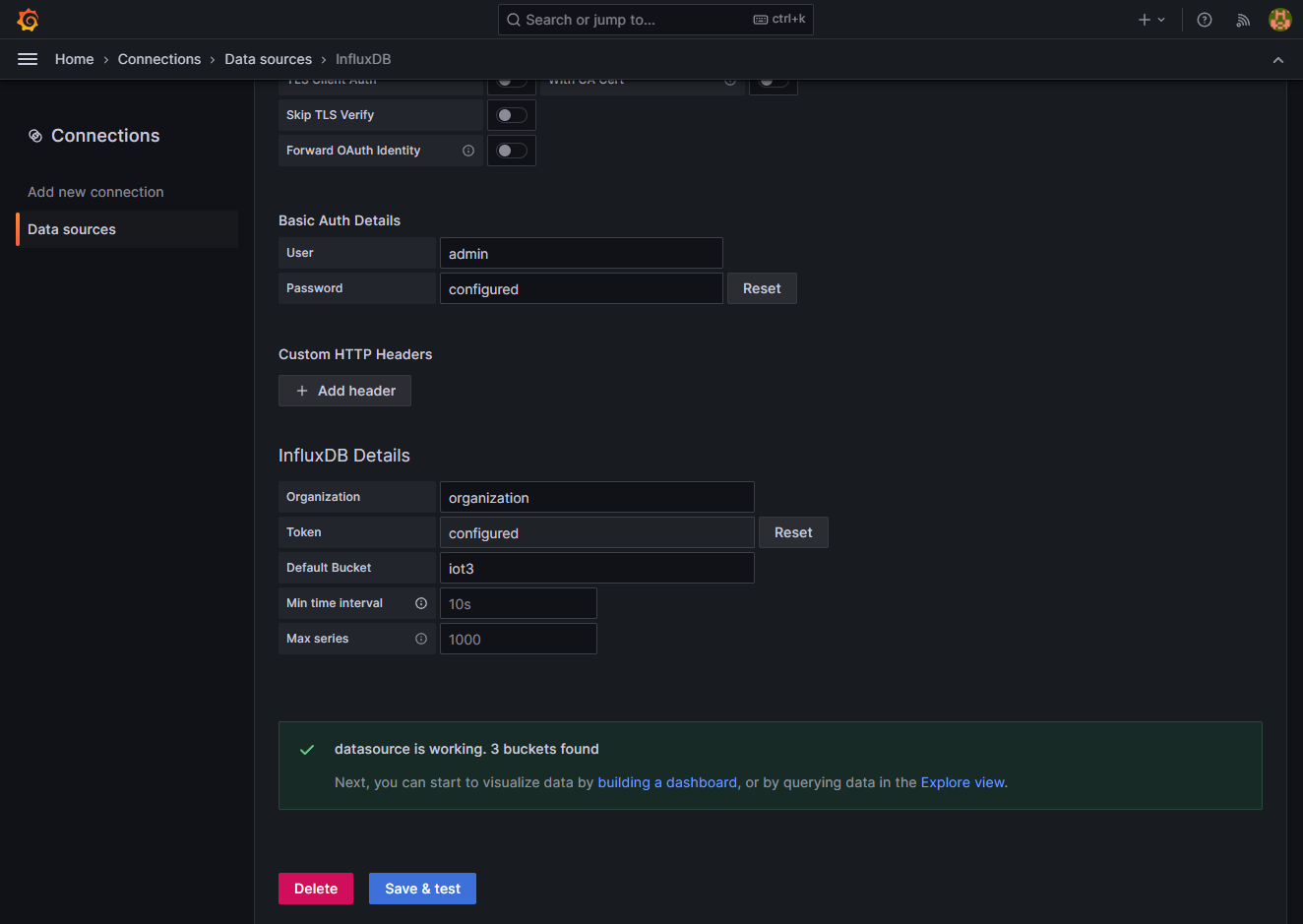


Sada izalzi forma koja treba da se popuni na sledeći način:

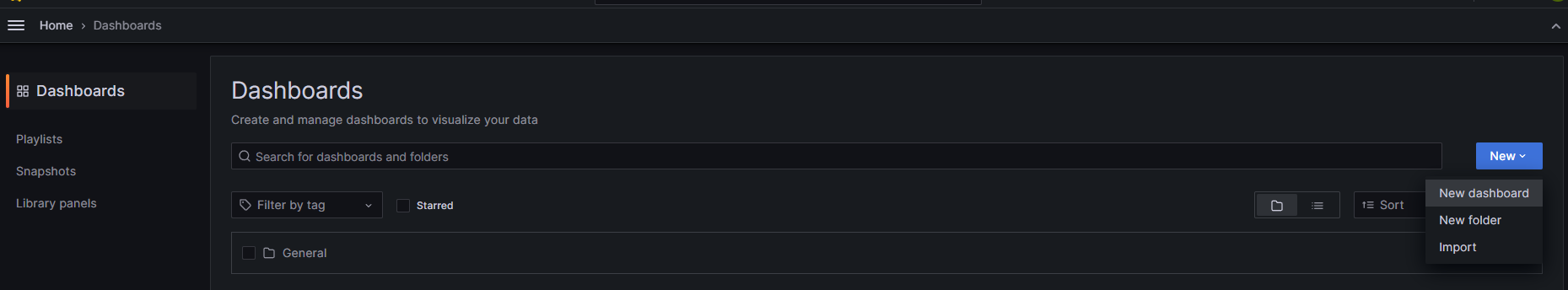
Query Language: Flux  
URL: <http://localhost:8086> (adresa gde se nalazi InfluxDB, localhost: adresa vašeg računara)  
Treba da se selektuje Basic Auth i popuni podacima za logovanje u bazu podataka:  
Username: admin  
Password: adminadmin  
Organization: organization  
Token: potreno je da se generiše All access API Token na InfluxDB stranici  




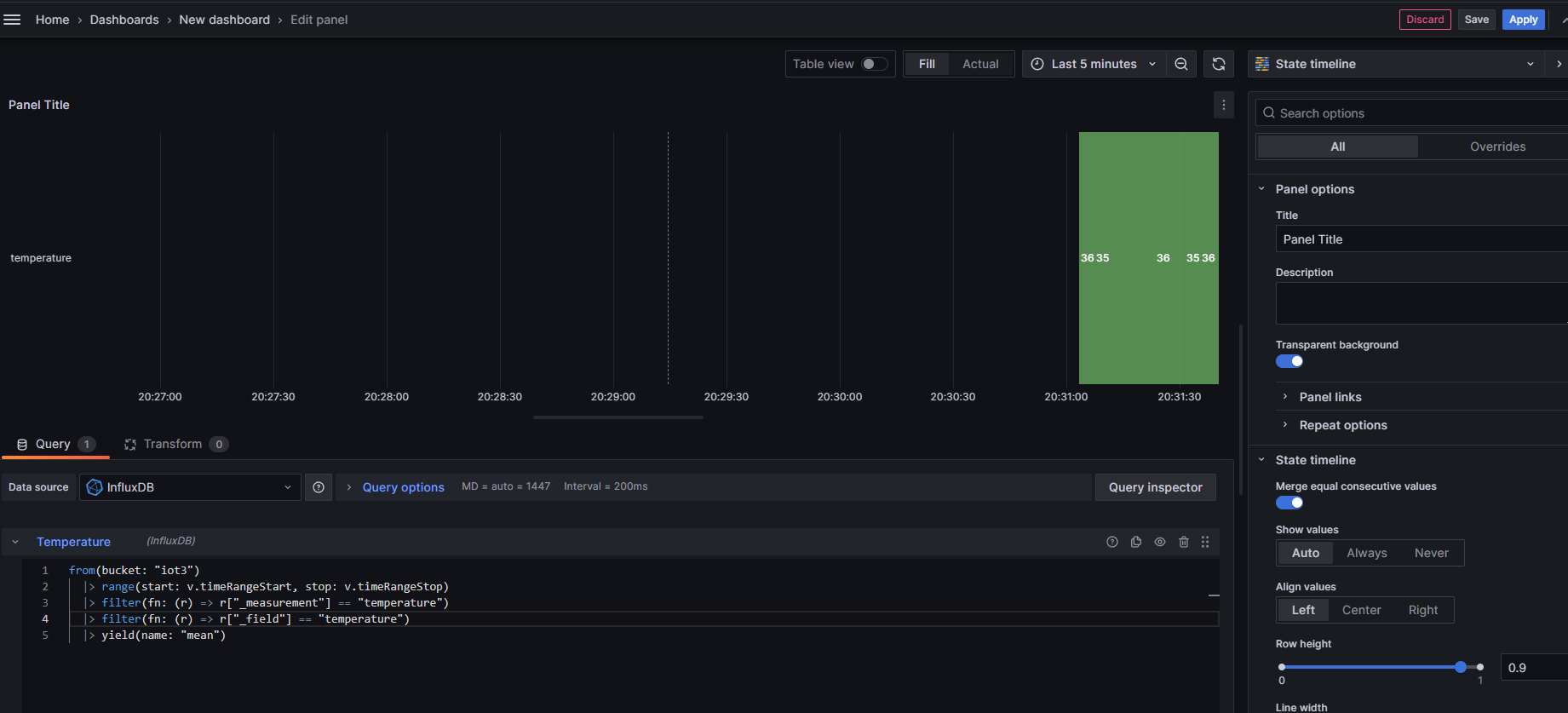
Nakon što se pritisne Save izlazi izgenerisani token koji treba da se unese na u formi kod Grafane.  
Default bucket: iot3

Nakon unošenja svih ovih podataka može da se pritisne dugme Save & test.

Sada je potrebno da se kreira novi Dashboard, tako što se ode na karticu Dashboard i pritisne dugme New Dashboard.   


Sada je potrebno da se unese upit za podatke koje želimo da prikažemo:



# Arhitektura sistema

Grafana

InfluxDB

Monitoring  
mikroservis

MQTT TOPIC:  
edgex/sensor\_value

EdgeX  
Foundary

Visualization  
mikroservis

PYTHON SKRIPTA

script.py