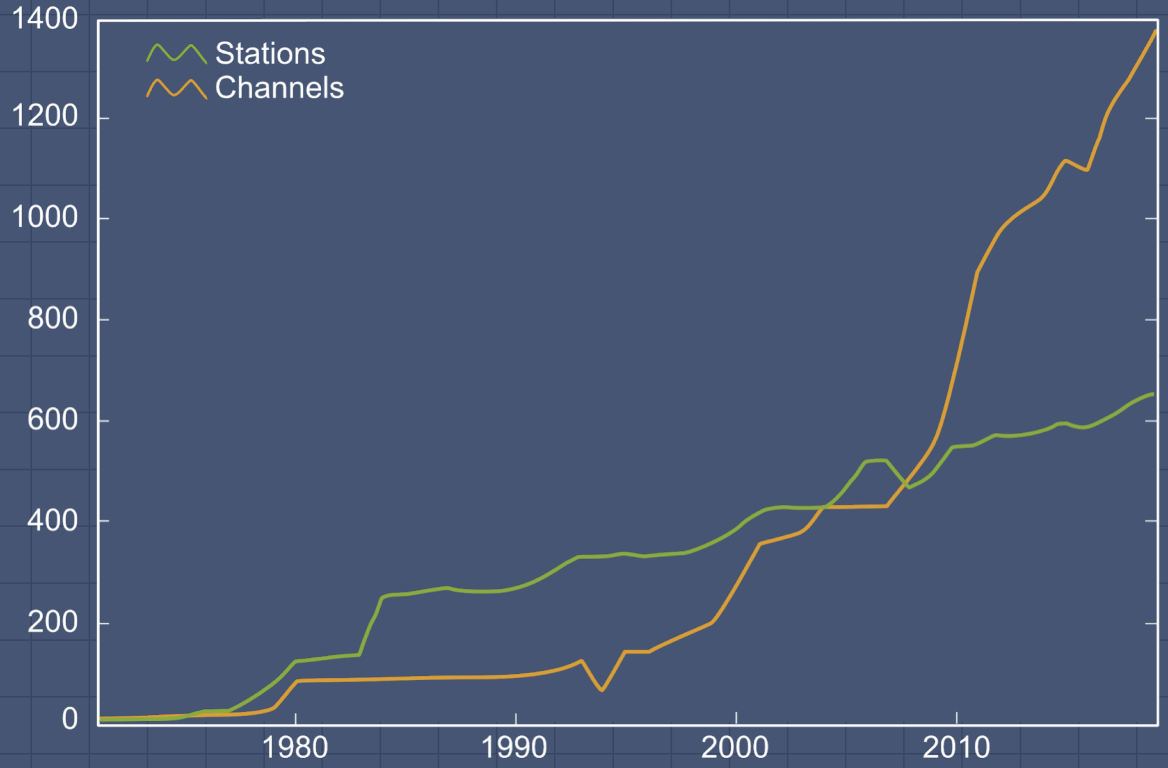
Unmet Performance Monitoring Needs

NetOpsX - Memphis 11/4/2019

Jon Connolly
Kyla Marczewski
Alex Hutko
Renate Hartog
Natalie Chow
Justin Clayton

PNSN
Pacific Northwest Seismic Network



PNSN growth over time

Motivation

PNSN needs an HTTPS dashboard that provides immediate insight and alerting on overall network performance to quickly identify and triage bad performing channels. While there are many existing well designed solutions that partially solve this need, no single application fully satisfy it.

What does the PNSN want?

User Roles (Admins, Reporters, Viewers, Data Contributors)

User Defined Dashboards, Metrics, Thresholds, and Alarms

Responsive HTTPS UI

Low Latency (hour)

Weighted Metric Aggregates

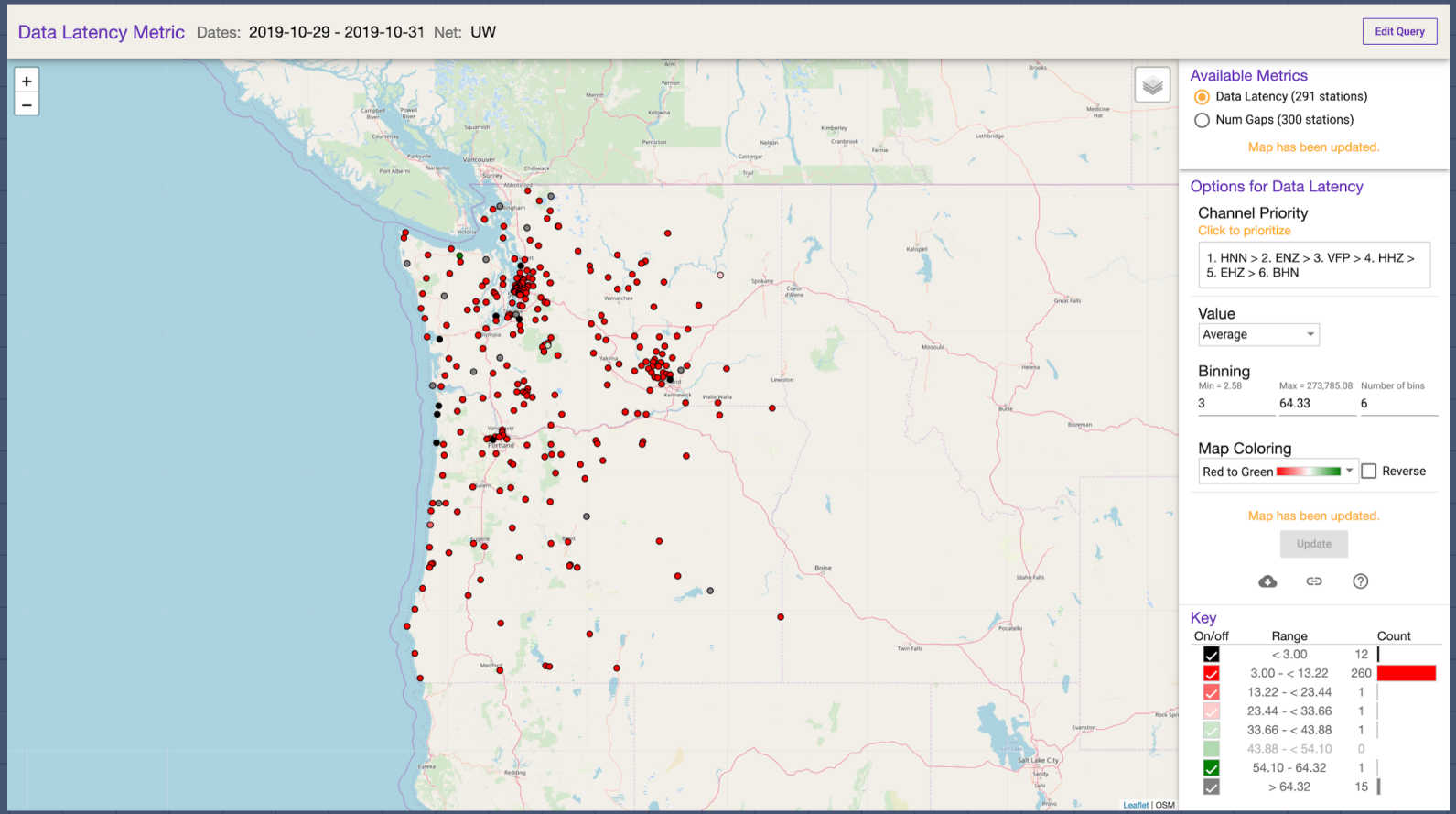
Notification by SMS, Email

Third party integration (Slack, ticketing, Trello)

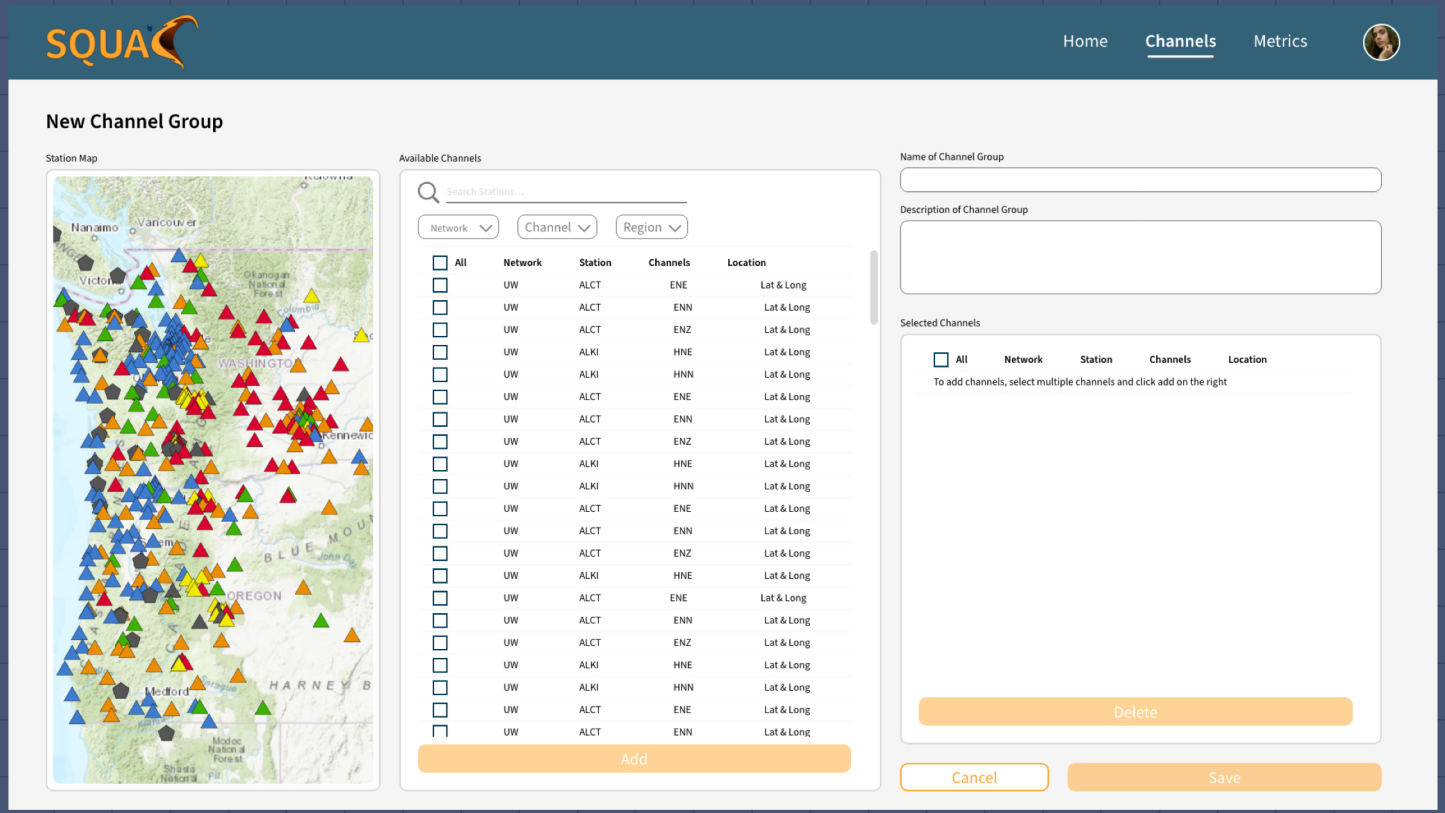
HTTPS API with simple wrapper

		MUSTANG	DQA	SeisNetWatch	Nagios
Features	User Defined Dashboards			■	
	User Roles				■
	Responsive HTTPS UI	■	■		
	Low Latency (hour)			■	■
	Weighted Metric Aggregates		■		
	User Defined Thresholds and Alarms		■	■	
	SMS, Email, Notifications				■
	Third party integration (ticketing, Slack)				■
	HTTPS API with wrapper library	■			■

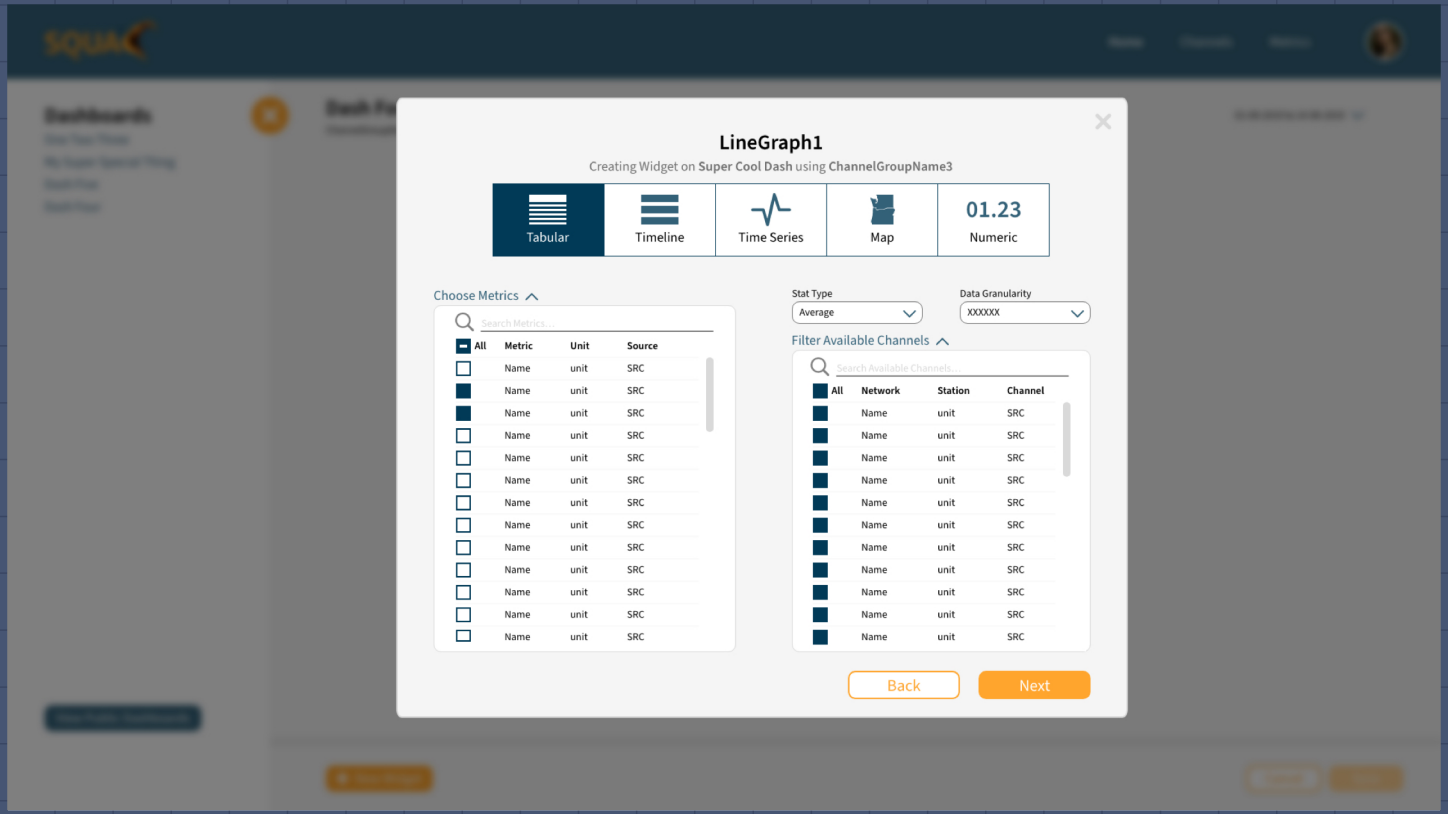
Desired features for SQUAC



Mustang + Angular = Mustangular

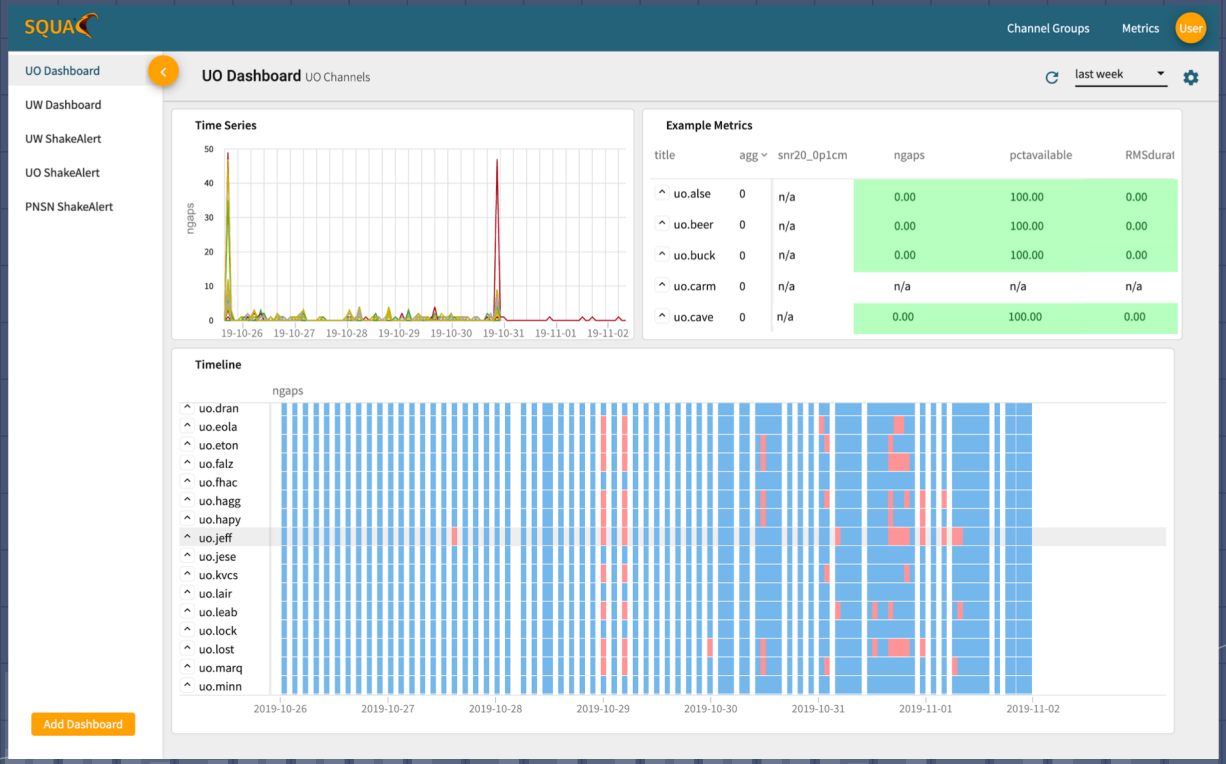


SQUAC Channel Group

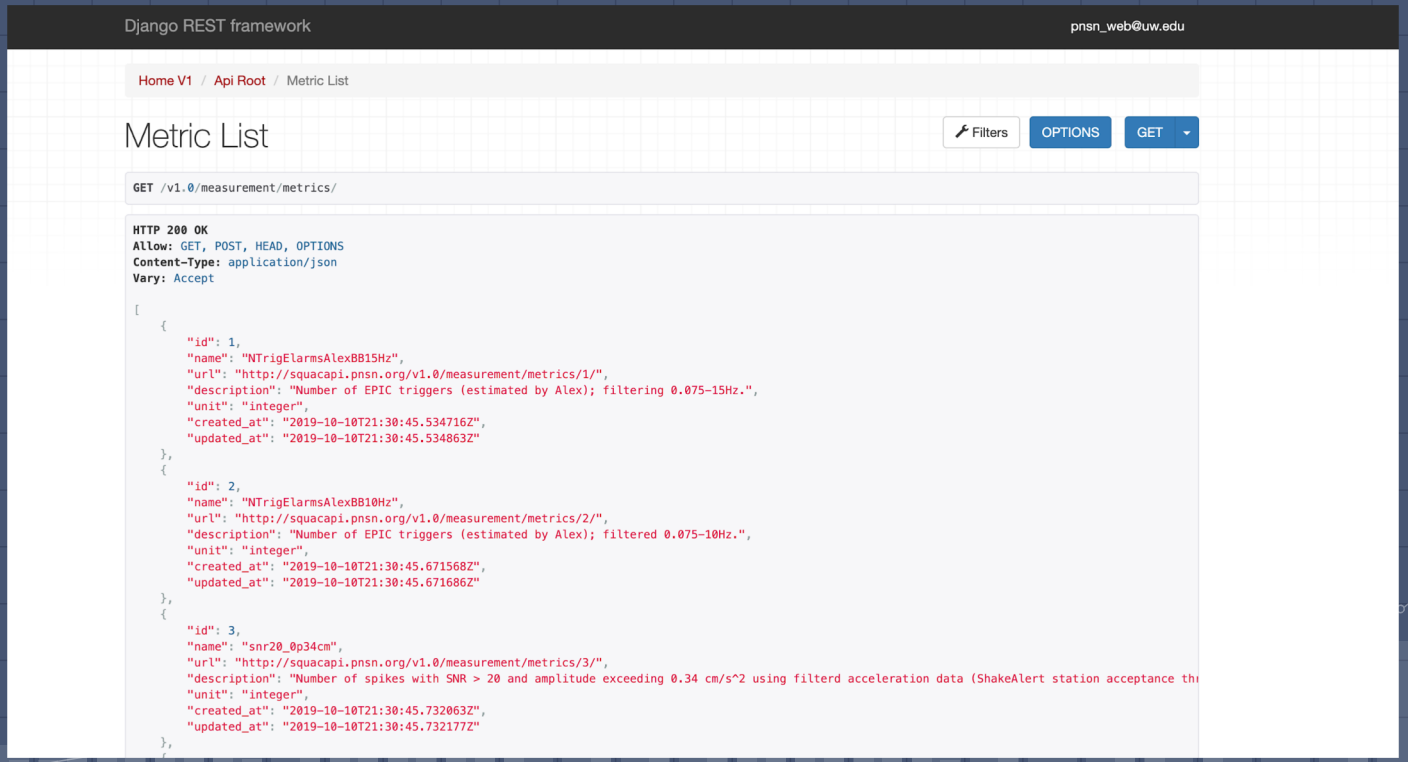


Widget Configuration

SQUAC UI


SQUAC
Channel Groups
Metrics
User
UO Dashboard
UW Dashboard
UW ShakeAlert
UO ShakeAlert
PNSN ShakeAlert
Add Dashboard
UO Dashboard UO Channels
last week
Time Series
ngaps
50
40
30
20
10
0
19-10-26
19-10-27
19-10-28
19-10-29
19-10-30
19-10-31
19-11-01
19-11-02
Example Metrics
title
agg
snr20_0p1cm
ngaps
pctavailable
RMSdurat
uo.alse 0 n/a 0.00 100.00 0.00
uo.beer 0 n/a 0.00 100.00 0.00
uo.buck 0 n/a 0.00 100.00 0.00
uo.carm 0 n/a n/a n/a n/a
uo.cave 0 n/a 0.00 100.00 0.00
Timeline
ngaps
uo.dran
uo.eola
uo.eton
uo.falz
uo.fhac
uo.hagg
uo.hapy
uo.jeff
uo.jese
uo.kvcs
uo.lair
uo.leab
uo.lock
uo.lost
uo.marq
uo.minn
2019-10-26
2019-10-27
2019-10-28
2019-10-29
2019-10-30
2019-10-31
2019-11-01
2019-11-02

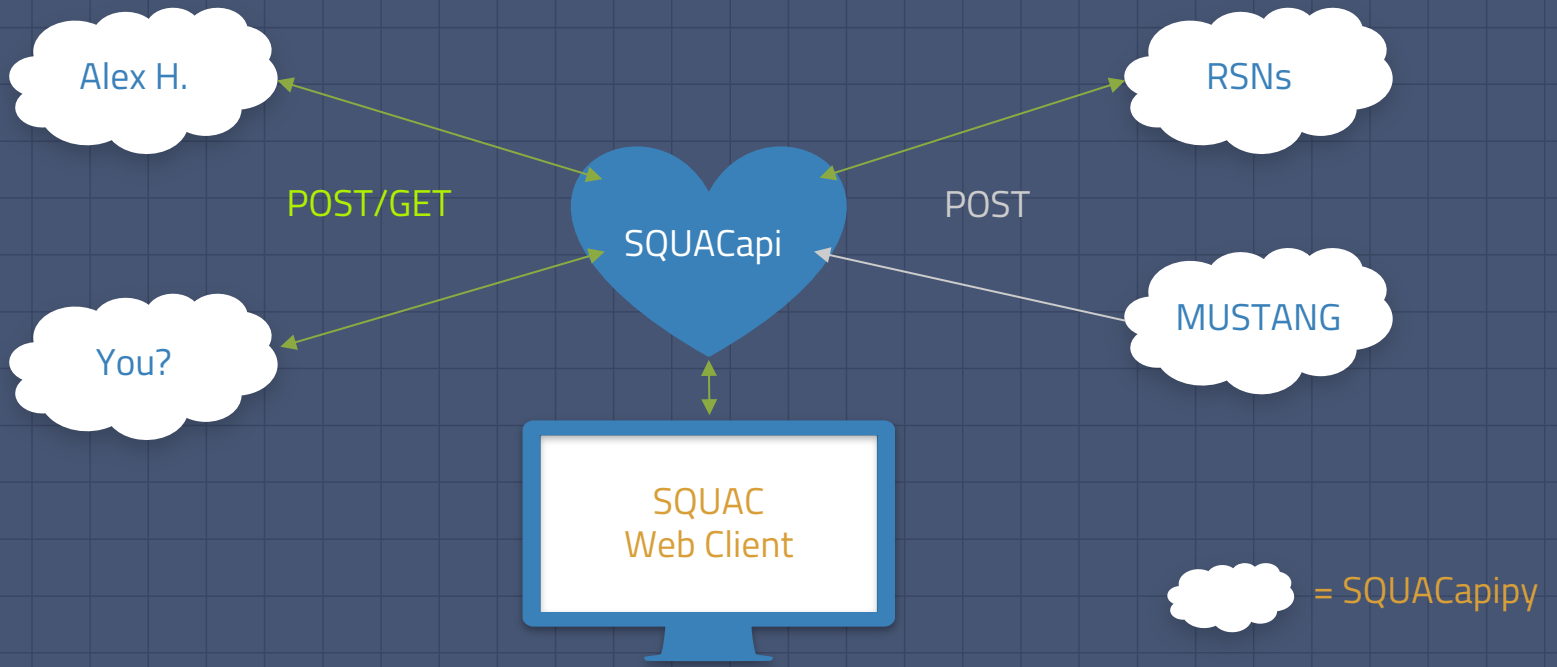
SQUACapi


Django REST framework
pnsn_web@uw.edu
Home V1 / Api Root / Metric List
Metric List
Filters
OPTIONS
GET
GET /v1.0/measurement/metrics/
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept
[
{
"id": 1,
"name": "NTrigElarmsAlexBB15Hz",
"url": "http://squacapi.pnsn.org/v1.0/measurement/metrics/1/",
"description": "Number of EPIC triggers (estimated by Alex); filtering 0.075-15Hz.",
"unit": "integer",
"created_at": "2019-10-10T21:30:45.534716Z",
"updated_at": "2019-10-10T21:30:45.534863Z"
},
{
"id": 2,
"name": "NTrigElarmsAlexBB10Hz",
"url": "http://squacapi.pnsn.org/v1.0/measurement/metrics/2/",
"description": "Number of EPIC triggers (estimated by Alex); filtered 0.075-10Hz.",
"unit": "integer",
"created_at": "2019-10-10T21:30:45.671568Z",
"updated_at": "2019-10-10T21:30:45.671686Z"
},
{
"id": 3,
"name": "snr20_0p34cm",
"url": "http://squacapi.pnsn.org/v1.0/measurement/metrics/3/",
"description": "Number of spikes with SNR > 20 and amplitude exceeding 0.34 cm/s^2 using filterd acceleration data (ShakeAlert station acceptance th
"unit": "integer",
"created_at": "2019-10-10T21:30:45.732063Z",
"updated_at": "2019-10-10T21:30:45.732177Z"
},

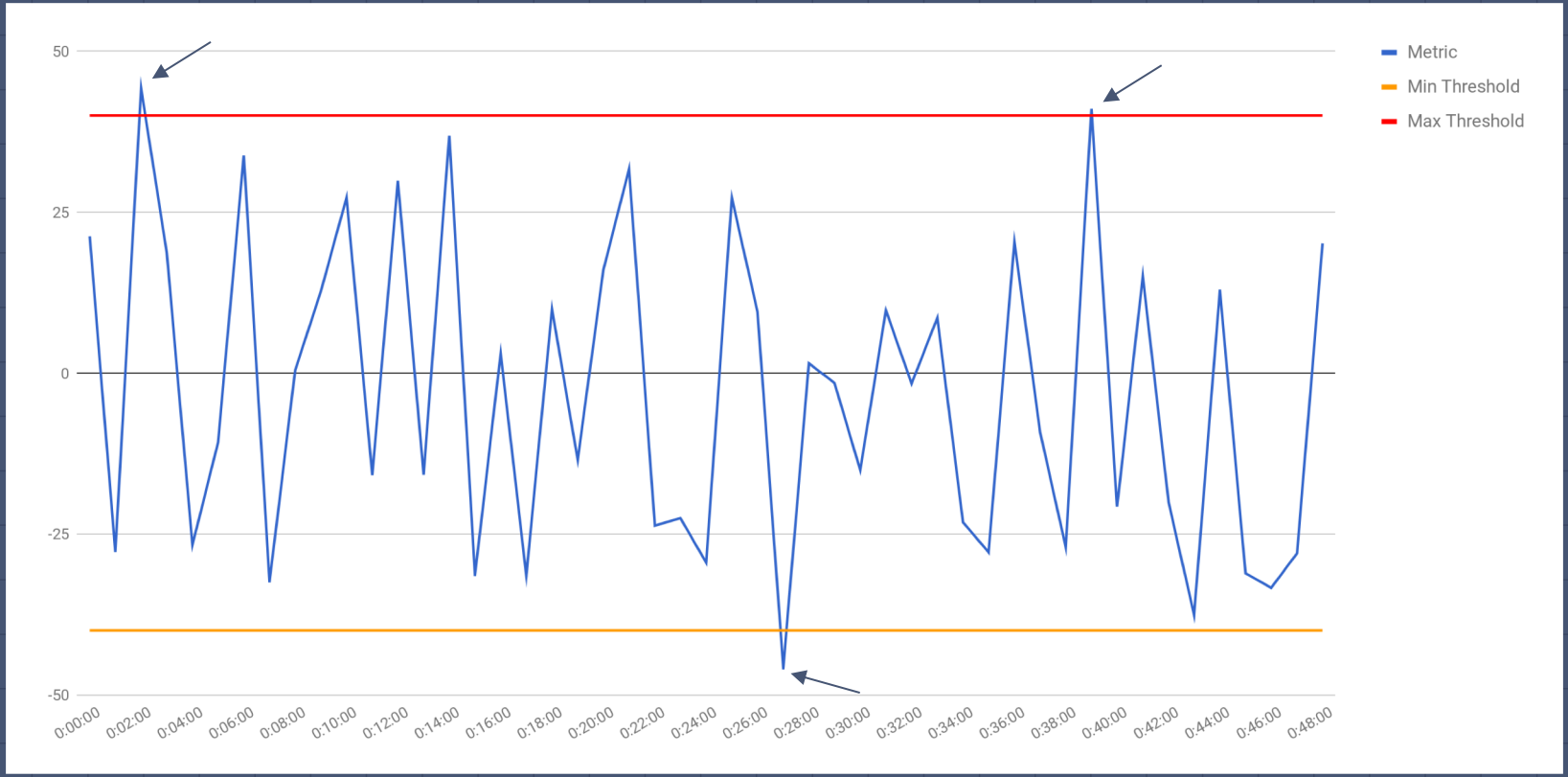
SQUACapipy

```
pip install squacapipy
```

```python
import os
from squacapipy.squacapi import Metric, Channel, Measurement

API_TOKEN = os.getenv('SQUAC_API_TOKEN')
API_BASE_URL = os.getenv('SQUAC_API_BASE')

chan = Channel().get(network='uw', station='RCM', loc='--', channel='HHZ')

metric = Metric().get(name='latency')

payload = {
    'channel': chan.id,
    'metric': metric.id,
    'value': 10.0,
    'starttime': datetime,
    'endtime': datetime

}

m = Measurement.post(payload)
if m.status_code != 200:
    print('Error {}'.format(m.body))
    exit(1)
```



SQUAC trinity



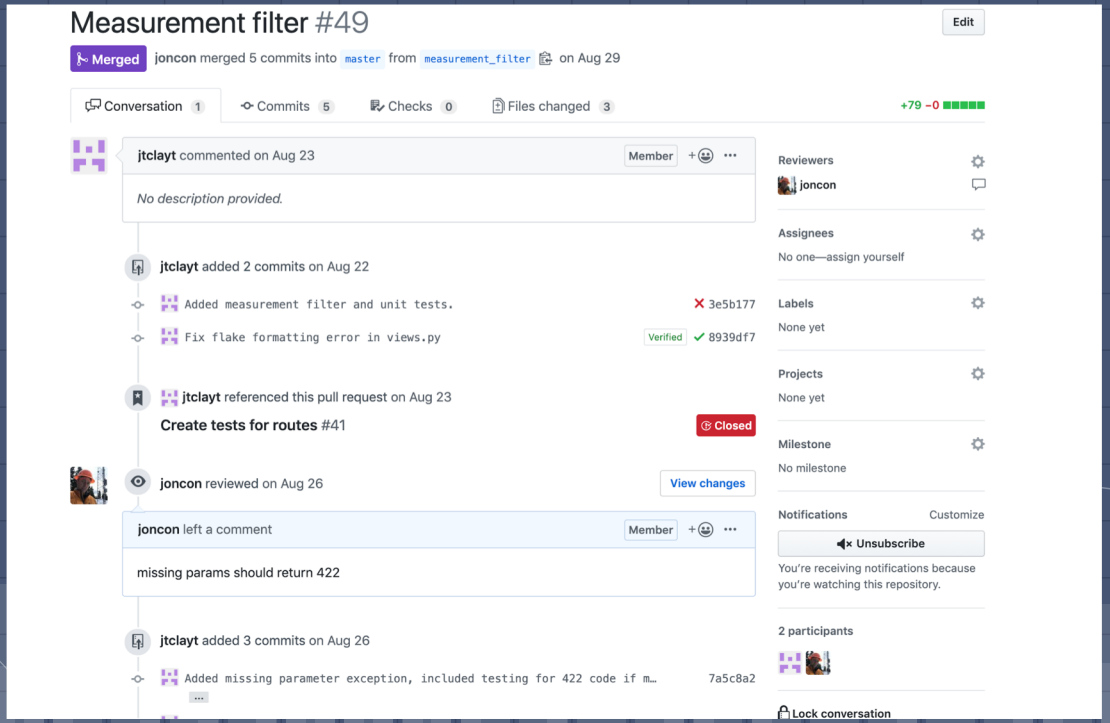
Example of metric thresholds

Contributors wanted

SQUAC

SQUACapi

SQUACapipy



Thanks to:

Kyla Marczewski - Front end development

Alex Hutko - Algorithms and data population

Renate Hartog - Product development

Natalie Chow - UI/UX design

Justin Clayton - Front and backend dev (student)