RWTH Hackathon Spring 2016

VIESSMANN TRACK

Idee

- Durch Analyse der Daten Einsicht gewinnen:
  - Potenzielle Ersparungsmöglichkeiten
  - Probleme aufgrund von falscher Konfiguration
  - …

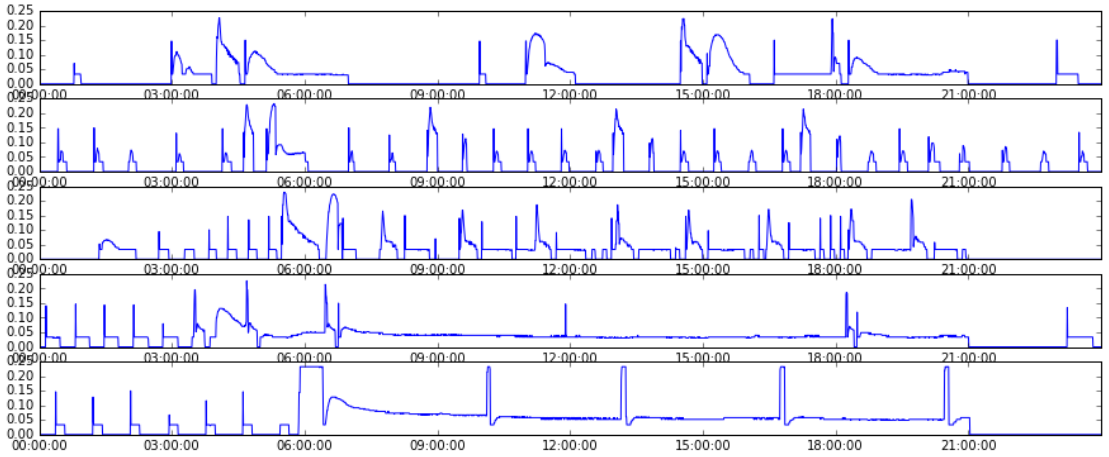
Data exploration mit Jupyter

um erste Analysen zu machen und den Datensatz zu verstehen

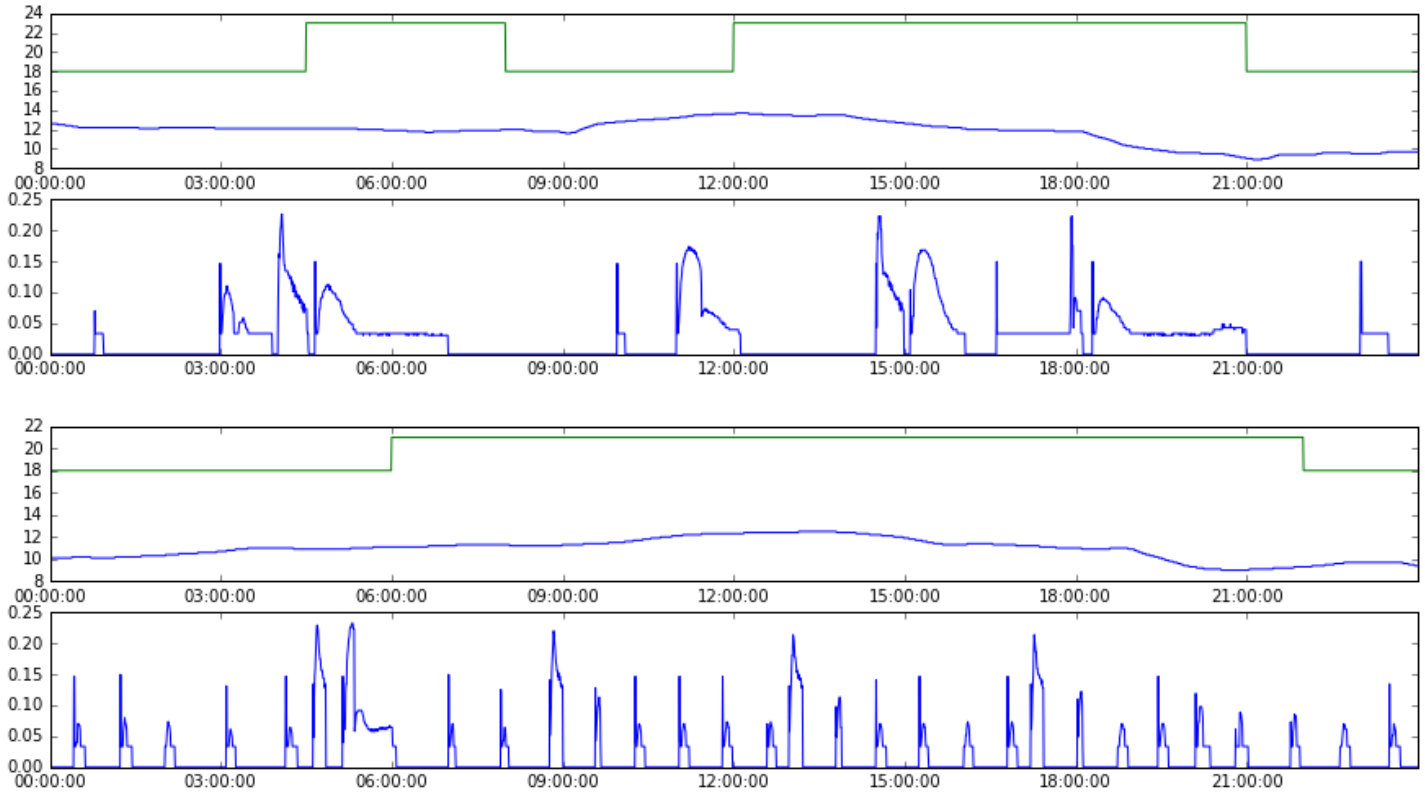
In [4]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as mpl
from IPython.display import display
%matplotlib inline
mpl.rcParams['figure.figsize'] = (15, 6)
```

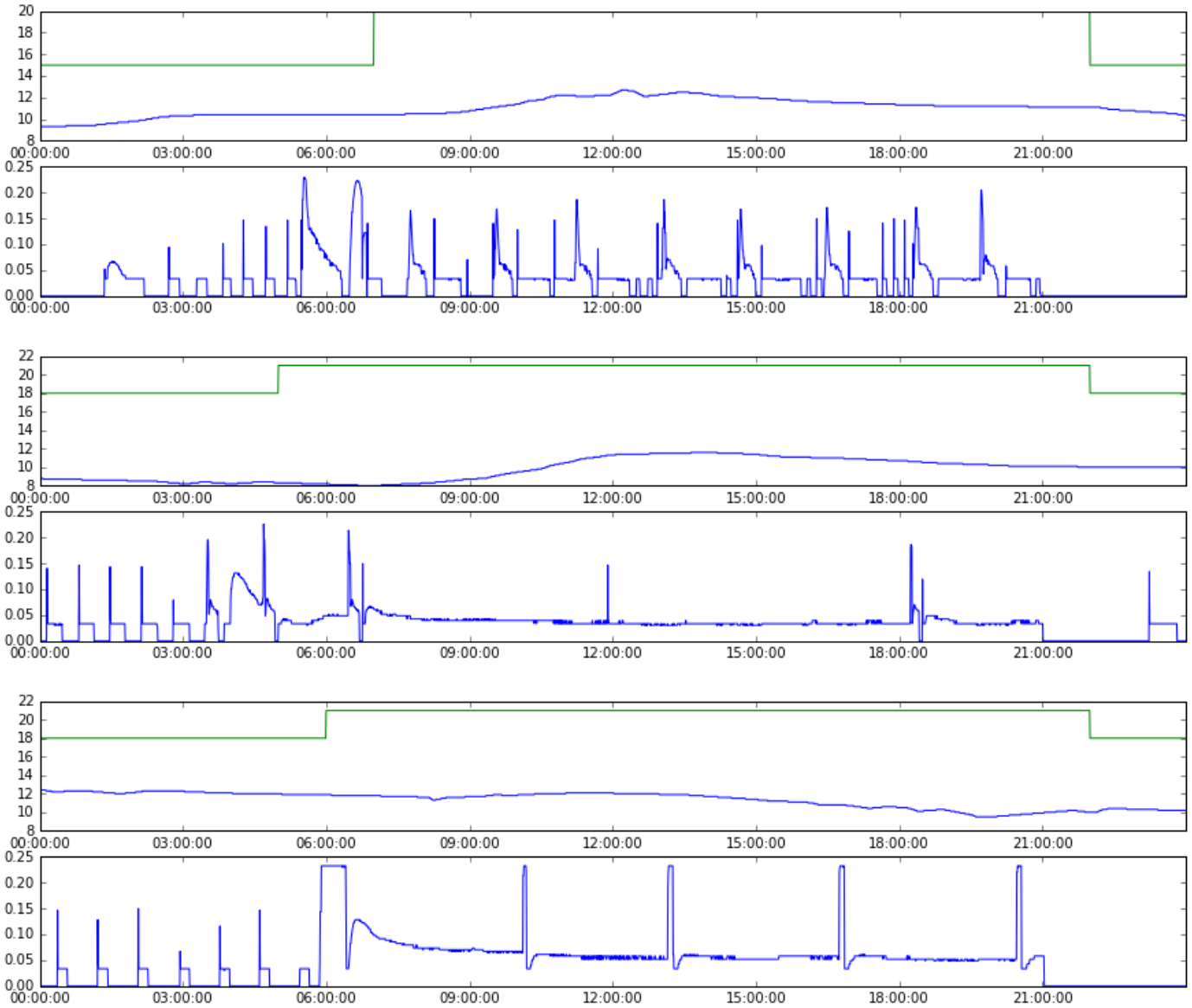
In [5]:
```
data1 = pd.read_csv("Anlage1.csv", sep=";", parse_dates=['timestamp'], index_col=['timestamp'])

data = []
for i in range(5):
    data.append(pd.read_csv("Anlage"+str(i+1)+".csv", sep=";", parse_dates=['timestamp'], index_col=['timestamp']))
for i in range(5):
    data[i].to_csv("Data_"+str(i)+".csv")
```

In [6]:
```
for i in range(5):
    mpl.subplot(5,1,i+1)
    mpl.plot(data[i].index[:2880], data[i]['energy_consumption_kwh'][:2880])
mpl.show()
```



In [8]:
```
mpl.rcParams['figure.figsize'] = (15, 20)
for i in range(5):
    mpl.subplot(10,1,i*2+1)
    mpl.plot(data[i].index[:2880], data[i]['outdoor_temperature_celsius'][:2880])
    mpl.plot(data[i].index[:2880], data[i]['indoor_target_temperature_celsius'][:2880])
    mpl.subplot(10,1,i*2+2)
    mpl.plot(data[i].index[:2880], data[i]['energy_consumption_kwh'][:2880])
    mpl.show()
```

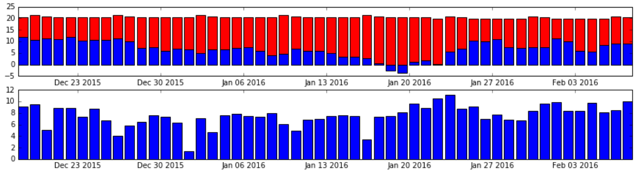



20
18
16
14
12
10
8
00:00:00
03:00:00
06:00:00
09:00:00
12:00:00
15:00:00
18:00:00
21:00:00
0.25
0.20
0.15
0.10
0.05
0.00
22

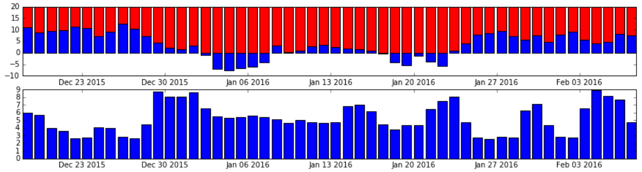
In [9]:

```python
mpl.rcParams['figure.figsize'] = (15, 20)
for i in range(5):
    gb = data[i].groupby(data[i].index.date)
    df = gb.agg({"outdoor_temperature_celsius": np.mean, "energy_consumption_kwh": np.sum, "indoor_target_temperature_
celsius": np.mean, "online_state": np.mean})
    df['avg_diff'] = df['indoor_target_temperature_celsius'] - df['outdoor_temperature_celsius']
    df['bla'] = df.energy_consumption_kwh / df.avg_diff
    df[df['online_state'] != 1]

    display(np.mean(df.bla))

    mpl.rcParams['figure.figsize'] = (15, 20)
    mpl.subplot(10,1,i*2+1)
    mpl.bar(df.index, df.indoor_target_temperature_celsius, color="red")
    mpl.bar(df.index, df.outdoor_temperature_celsius, color="blue")
    mpl.subplot(10,1,i*2+2)
    mpl.bar(df.index, df.bla, color="blue")
    mpl.show()
```

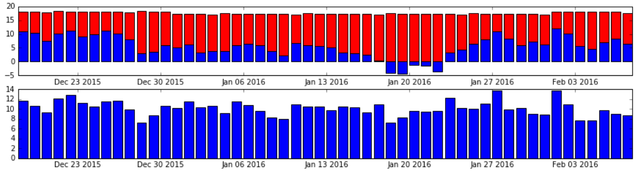
7.572035868053792



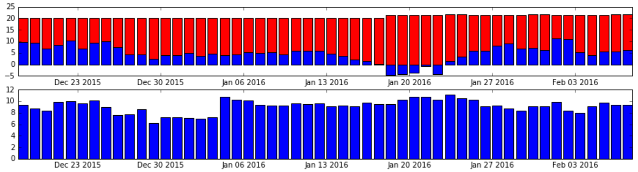
5.2481580471128808



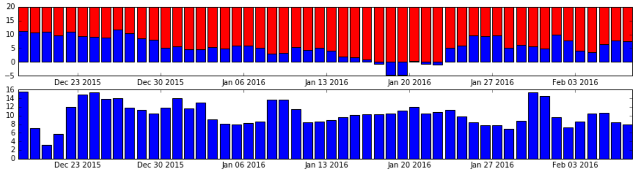
10.138937606316635



9.1565316796560303

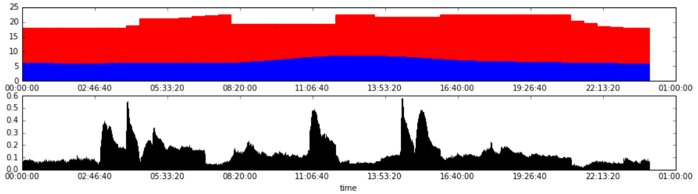


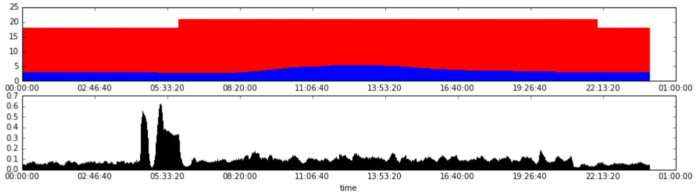
10.401384310776406

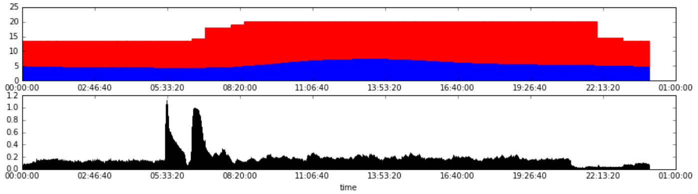


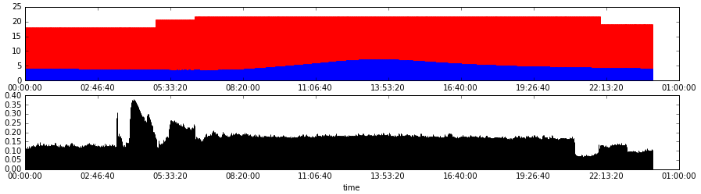
In [10]:

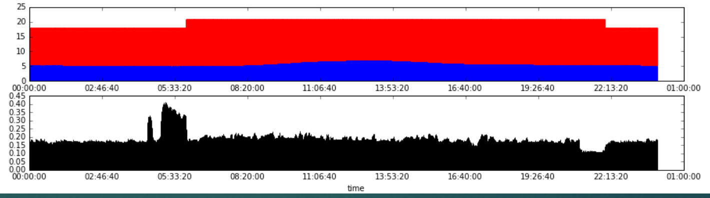
```python
mpl.rcParams['figure.figsize'] = (15, 20)
for i in range(5):
    gb = data[i].groupby(data[i].index.time)
    df = gb.agg({"outdoor_temperature_celsius": np.mean, "energy_consumption_kwh": np.sum, "indoor_target_temperature_
celsius": np.mean, "online_state": np.mean})
    df['avg_diff'] = df['indoor_target_temperature_celsius'] - df['outdoor_temperature_celsius']
    df['bla'] = df.energy_consumption_kwh / df.avg_diff
    #df[df['online_state'] != 1]

    mpl.rcParams['figure.figsize'] = (15, 20)
    mpl.subplot(10,1,i*2+1)
    mpl.bar(df.index, df.indoor_target_temperature_celsius, color="red", edgecolor="red")
    mpl.bar(df.index, df.outdoor_temperature_celsius, color="blue", edgecolor="blue")
    mpl.subplot(10,1,i*2+2)
    mpl.bar(df.index, df.bla, color="blue")
    mpl.show()
```





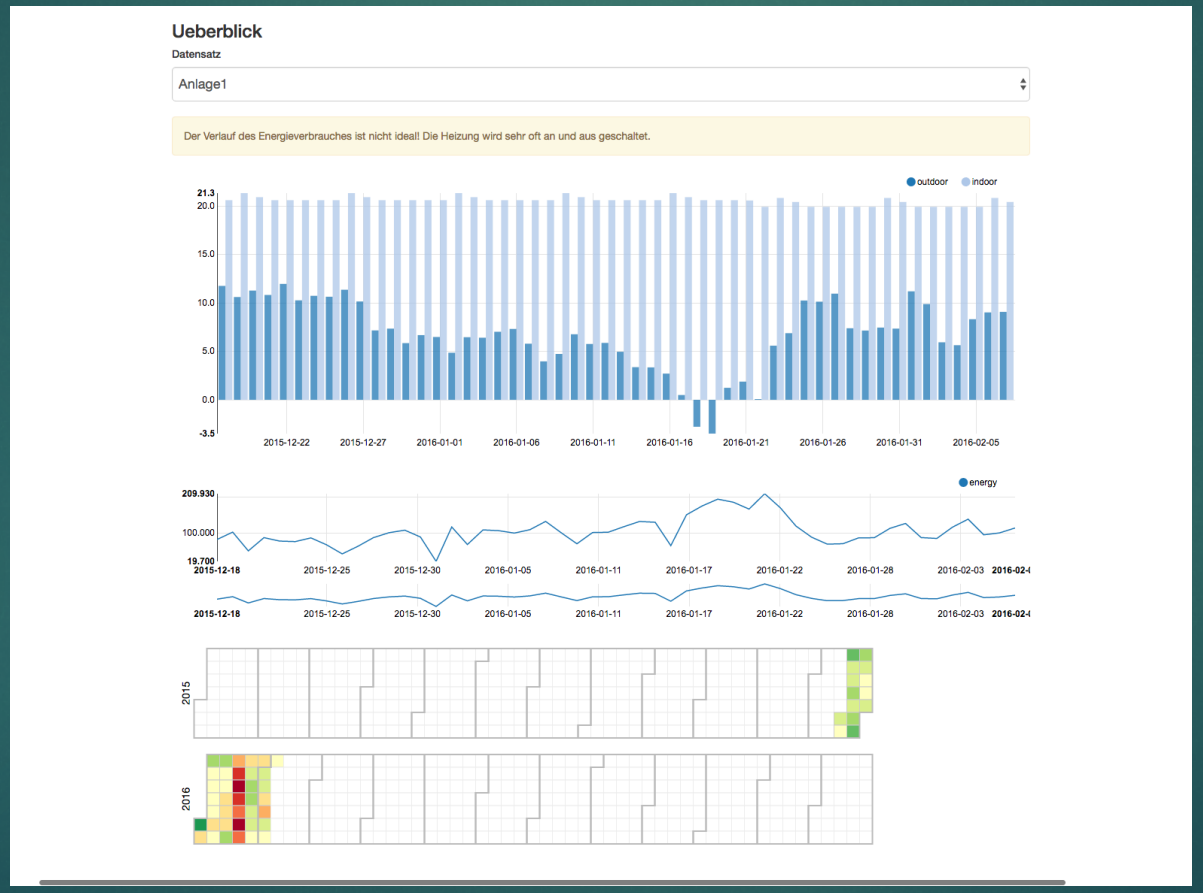






Prototypische Umsetzung

in Form einer Web App (HTML, CSS, JS)



Überblick über eine einzelne Anlage

- Durchschnittliche Temperaturen
- Summierter Energieverbrauch pro Tag
- “Heat Map” der Energieverbräuche

Detail Ansichten einzelner Tage

- Temperaturen über den Tagesverlauf
- Energieverbrauch im Tagesverlauf

Weitere Beispiele

Ueberblick

Datensatz

Anlage2

Der Verlauf des Energieverbrauches ist nicht ideal! Die Heizung wird sehr oft an und aus geschaltet.

Tag: 27. January 2016