Analýza a vizualizace dat - Regulus

Magisterstký projekt - Jakub Tichý

# Úvod

V dnešní době je trendem využívat pro snížení spotřeby elektrické energie alternativní zdroje, primárně tepelná čerpadla a solární panely. Někteří poskytovatelé těchto služeb zároveň nabízí i monitorovací programy. Tyto programy jsou však pouze realtimové a nelze v nich dohledávat data zpětně. Pro tyto účely je už vytvořen scrapper, který data z monitorovacího systému scrappuje a nahrává do databáze.

Nad daty, která jsou dostupná z databáze je možné dělat analýzu a výstupy vizualizovat. Lze tedy pomocí patřičných nástrojů vytvořit přehledný reportovací nástroj, který by mohl uživateli zobrazovat jak se data měnila v čase.

# Cíl práce

Cílem práce je analyzovat stávající stav, problematiku a nabytých poznatků vytvořit adekvátní zadání pro diplomovou práci. Závěr práce by měl obsahovat: současný stav řešení problematiky, analýzu požadavků na systém, hrubý návrh architektury systému, implementaci, způsob testování.

# Současný stav

V současnosti se data ukládají na serveru do InfluxDB. Influx je databáze určená přímo pro ukládání časových řad v krátkém čase. Některá data zobrazují na serveru v grafaně *{url}*, tento nástroj umožňuje vykreslovat data přímo z DB, ale už nám neumožňuje provádět detailnější anaýzu nad daty. Oba tyto systémy běží na jednom serveru jako docker kontejnery.

# 

# Požadavky na systém

## Uživatelské

Abychom nedělali pouze kopii stávajícího řešení, bude potřeba systém navrhnout tak, aby uřivateli dával přidanou hodnotu. Měl by tedy obsahovat něco navíc oproti současnému řešení. Přidanou hodnotou by měli být nové metriky, které uživateli umožní sledovat jak hospodaří a na základě těchto dat činit další rozhodnutí.

## Technické

Analýza a zpracování dat bude provedena v jazyce python. Python jsem si vybral z důvodu vysoké množství knihoven a pro snadnou manipulaci s daty.

Data, která jsou předzpracována se musí někde vizualizovat. Pro tento účel jsem zvolil knihovnu Plotly Dash pro python. Plotly je open source framework pod licencí MIT.

Pomocí Plotly se dá relativně snadno vykreslovat data pomocí grafů bez nutnosti REST API mezi frontendovou a backendovou částí. Jedná se v podstatě o full-stack apliakci, která pro render grafů využívá React JS.

Pro získávání dat bude použita stávající databáze InfluxDB, která slouží i pro aktuální řešení.

Co se orchestrace a kontejnerizace týče, v rámci magisterského projektu by zatím mělo stačit spuštění systému na jednom kontejneru.

# Experimentální část

Jako experiment mě napadlo vytvořit takový systém, který by nesloužil pro vizualizaci dat pouze jednoho konkrétního zdroje ale aby mohl vizualizovat více na sobě nezávislých zdrojů dat.

Systém by pak nebyl omezen pouze na datovou sadu, kterou máme nyní k dispozici ale pomocí API by si do nové databáze uložil údaje k jinému datovému droji, který by obsahovala sadu naměřených dat.

Pro tyto účely bude zapotřebí vytvořit v aplikaci middleware pro získávání a uchovávání dat z různých typů databází. Tyto data pak ukládat do nové databáze. Typ databáze pro ukládání dat by mohl být SQL nebo MongoDB. Data, která by se do naší nové databáze ukládala by obsahovala tyto údaje:

* přístupové údaje do databáze(pouze čtení, typ DB)
* strukturu tabulek(relace, primární klíče, sekundární klíče)
* obsah tabulek(atributy, datové typy)

Při požadavku pro vykreslení těchto dat, by se klient v apliakci už pouze dotazoval do ddatového zdroje, který by byl uložen v naší databázi a data si získával stejně, jako si je získává z InfluxDB.

### Tato eperimentální část sebou nese i navýšení technických požadavků. Určitě bude potřeba vytvořit další databázi pro uchovávání informací k dalším datovým sadám. Pro tuto DB bude zapotřebí další docker kontejner.

### Otázky

Zda pro vizualizaci každé datové sady vytvářet kontejner nebo to zobrazovat na jednom a zybtek řešit přes middleware. Pokud bude vhodnější řešení přes docker kontejnery, do jaké míry bude potřeba kubernetes.