

ZADÁVACÍ PROTOKOL MAGISTERSKÉ PRÁCE

Student: Eva Adámková, Bc.

Obor – zaměření studia: Aplikovaná informatika

Katedra: Ústav aplikované informatiky

Školitel:

Jan Piša, Ing. (externí školitel)

FIEDLER AMS s.r.o., Lipová 1789/9, 370 05 České Budějovice

pisa@fiedler.company, +420 777 213 322

Garant z PřF: Rudolf Vohnout, Ing. Ph.D.

Téma magisterské práce: Tvorba systému interaktivních prvků pro uživatelské obrazovky

Cíle práce:

1. Proved'te analýzu aktuálně používaných prvků (widgetů) v různých systémech uživatelských obrazovek (dashboardů) a v systému uživatelských obrazovek zadavatele.
2. Proved'te analýzu datových toků vybraných widgetů.
3. Proved'te analýzu možností konfigurace vybraných widgetů.
4. Na základě analýzy vytvořte návrh a implementaci systému (frameworku), který zajistí správu celého životního cyklu widgetu, od jeho konfigurace, předzpracování dat, až po vykreslení těchto dat uživateli.
5. Proved'te implementaci několika widgetů za pomoci navrženého systému.
6. Vytvořte unity testy pro vytvořený systém.

Základní doporučená literatura:

Aktuální materiály dostupné na síti Internet.

Typescriptlang.org. (2017). Documentation · TypeScript. [online] Dostupné z: <https://www.typescriptlang.org/docs/> [cit. 2017-12-12].

Charrington, D. (2017). Aurelia For Real World Web Applications. [ebook] leanpub.com. Dostupné z: <https://leanpub.com/aurelia-for-real-world-applications> [cit. 2017-12-12].

Financování práce:

Vedoucí práce: Ing. Jan Píša

podpis:

U externích vedoucích fakultní garant práce: Ing. Rudolf Vohnout, Ph.D.

podpis:

Garant oboru mag. studia:
(nepožaduje se u zaměření „příprava na mag. studium biologie)

podpis:

Vedoucí katedry/ústavu, kde bude práce vypracovávána:

podpis:

Případný souhlas vedoucího ústavu AV

podpis:

V Českých Budějovicích dne 14. 12. 2017

Převzal/a dne podpis studenta:

Jihočeská univerzita v Českých Budějovicích
Přírodovědecká fakulta

**Tvorba systému interaktivních prvků pro uživatelské
obrazovky**

Diplomová práce

Bc. Eva Adámková

Vedoucí práce: Ing. Jan Píša, FIEDLER AMS s.r.o.

České Budějovice 2018

Adámková, E., 2018: Tvorba systému interaktivních prvků pro uživatelské obrazovky. [Creation of interactive widgets and dashboards system. Mgr. Thesis, in Czech.] – 120 p., Faculty of Science, University of South Bohemia, České Budějovice, Czech Republic.

Prohlášení

Prohlašuji, že svoji diplomovou práci jsem vypracovala samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své diplomové práce, a to v úpravě vzniklé vypuštěním vyznačených částí archivovaných Přírodovědeckou fakultou, elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

České Budějovice, 18. 4. 2018

Eva Adámková

Poděkování

Touto cestou bych chtěla poděkovat svému vedoucímu práce Ing. Janu Píšovi a kolegovi Mgr. Janu Tůmovi za trpělivost a užitečné rady při konzultačních schůzkách. Chtěla bych poděkovat také garantovi práce Ing. Rudolfovi Vohnoutovi, Ph.D. za ochotu a cenné komentáře.

Dále mé díky patří Ing. Jindřichovi Fiedlerovi za příležitost realizovat tuto práci pro společnost FIEDLER AMS s.r.o.

A v neposlední řadě děkuji své rodině za trpělivost a podporu.

Abstract

This thesis deals with the problem of interactive dashboards. It is focused on the analysis of existing widgets, their purpose and real usage in terms of configuration and final visualization. It also emphasizes the analysis of inputs and data streams within selected interactive elements. The aim of the thesis is to design and implement a system for managing the whole lifecycle of interactive elements based on this analysis.

Abstrakt

Diplomová práce se zabývá problémem systému interaktivních obrazovek (dashboards). Zaměřuje se na analýzu jejich interaktivních prvků (widgets), jejich účel a reálné užití z pohledu konfigurace a konečného zobrazení. Dává důraz také na analýzu vstupů a datových toků vybraných interaktivních prvků. Cílem práce je pak na základě této analýzy vytvořit návrh a implementaci systému pro správu celého životního cyklu interaktivních prvků.

Obsah

1	Úvod	1
2	Popis problému a motivace	3
2.1	Společnost FIEDLER AMS	3
2.2	Aplikace zadavatele	3
2.2.1	Původní aplikace „Web prohlížeč naměřených dat“	4
2.2.2	Aktuální aplikace „CloudFM“	5
2.2.3	Cíle práce	5
3	Analýza současného stavu	7
3.1	Widget	7
3.1.1	Konfigurace a řídicí jednotka widgetu	8
3.1.2	Kategorizace widgetů	12
3.2	Data a jejich kategorizace	14
3.2.1	Zdroj dat	15
3.2.2	Účel dat	16
3.2.3	Metadata	17
3.2.4	Využití dat	18
3.2.5	Formát dat	19
3.2.6	Transformace dat	20
3.2.7	Závislost dat na čase	22
3.3	Analýza současných widgetů	23
3.3.1	Kategorizace současných widgetů	24
3.3.1.1	Grafy	24

3.3.1.2	Tabulky	30
3.3.1.3	KPI	34
3.3.1.4	Mapy	38
3.3.1.5	Export dat	40
3.3.1.6	Ostatní	41
3.3.2	Analýza datových toků	43
3.3.2.1	Analýza dat ze serveru	45
3.3.2.2	Transformace dat na straně klienta	53
3.4	Analýza ostatních systémů	58
3.4.1	Web prohlížeč naměřených dat	58
3.4.1.1	Srovnání	64
3.4.2	Odoo	65
3.4.2.1	Srovnání	68
3.4.3	UNMS	69
3.4.3.1	Srovnání	70
3.4.4	Sensorthings	71
3.4.4.1	Srovnání	74
4	Návrh řešení	75
5	Realizace	77
6	Testování	79
6.1	Package-by-feature	79
6.2	Validace kódu	80
6.3	Unit testy	80
6.3.1	Řešené problémy	81
6.4	Ostatní testy	82
7	Náměty na rozšíření	83
8	Závěr	85

OBSAH

Literatura	87
A Seznam použitých zkratk	91
B Slovník pojmů	93
C Instalační příručka	95
D Obsah přiloženého CD	97

Seznam obrázků

2.1	Web prohlížeč naměřených dat	4
2.2	Portál CloudFM	5
3.1	Struktura widgetu	8
3.2	Naklonění časového rámce [4]	22
3.3	„Binární graf“ - chod motorů M1 - M3	25
3.4	Příklady čárových grafů - nalevo graf čárový jednoduchý, napravo čárový graf s transformací. Pro názornost zobrazují oba grafy hodnoty ze stejných kanálů a za stejné období.	26
3.5	Čárový graf „Limity“	27
3.6	Příklad sloupcového grafu s agregovanými daty	28
3.7	Příklad sloupcového grafu se spotřebou u sdruženého vodoměru	28
3.8	Tabulka „Počet změn binárních kanálů“	31
3.9	Tabulky „Hodnoty“ a „Spotřeba“	32
3.10	Tabulka s počtem výskytů jednotlivých hodnot	32
3.11	Tabulka „Log událostí“	33
3.12	Tabulka „Základní informace“	34
3.13	KPI „Událostí“	35
3.14	Několik možností použití „Univerzálního KPI“	37
3.15	Prvky - „KPI Odkaz“, „KPI Metadata“ a „KPI Ukazatel signálu“	37
3.16	KPI Limity	38
3.17	Mapa „Souřadnice“	39
3.18	Mapa „Limity“	40

3.19	Export dat	41
3.20	Ostatní widgety - „Interaktivní obrázek“	42
3.21	Volitelné webové komponenty „Graf 0 ... max“ a „Graf min ... max“	61
3.22	Volitelná webová komponenta „Limnigrafická stanice“	62
3.23	Volitelná webová komponenta „Větrná růžice“	63
3.24	Volitelná webová komponenta „Srážkoměr“	63
3.25	Odoe export	65
3.26	Odoe „Kanban“	66
3.27	Odoe „Graf“	67
3.28	Ubiquity „Graf“	70
3.29	SensorUp „Mapa“	72
3.30	SensorUp „Grafy“	72
3.31	SensorUp „Kalendář“	73
3.32	SensorUp „Animace“	74
6.1	Adresářová struktura „Package-by-feature“	79
6.2	Čárový graf bez osy x	81

Seznam tabulek

3.1	Maximální nastavení v uživatelské konfiguraci	10
3.2	Maximální nastavení v řídicí jednotce	11
3.3	Částečná konfigurace a nastavení	12
3.4	Značky datového slovníku	45
3.5	Analýza dat ze serveru - vybrané widgety (1. část)	51
3.6	Analýza dat ze serveru - vybrané widgety (2. část)	52
3.7	Analýza dat na straně klienta - vybrané widgety (1. část)	56
3.8	Analýza dat na straně klienta - vybrané widgety (2. část)	57

SEZNAM TABULEK

Kapitola 1

Úvod

S vývojem nového hardwaru a rostoucími požadavky zákazníků se společnost FIEDLER AMS s.r.o. rozhodla pro vytvoření nové webové aplikace pro prohlížení naměřených dat ze senzorů. V návaznosti na potřeby zákazníků byl zaveden a nasazen nový systém pro prohlížení dat - CloudFM. Na základě jeho používání byly vzneseny nové požadavky na jeho optimalizaci s cílem dosažení maximální flexibility v rámci celé škály nových řešení pro monitoring a řízení technologií.

Tato práce se zabývá stavem systémů zadavatele, které umožňují prohlížení a další práci s daty ze senzorů a měřících jednotek. Práce se konkrétně věnuje tvorbě interaktivních prvků uživatelských obrazovek v rámci těchto aplikací. Cílem této práce je navržení a implementace systému pro tvorbu interaktivních prvků, který zjednoduší jejich tvorbu a rozšíří oblast jejich využití.

V kapitole 2 „Popis problému, specifikace cíle“ jsou představeny aplikace, které zadavatel využívá k prohlížení naměřených dat.

Třetí kapitola 3 „Analýza“ se soustředí na rozsáhlou analýzu především stávající aplikace zadavatele, konkrétně struktury tvorby interaktivních prvků pro uživatelské obrazovky. V kapitole jsou dále srovnány přístupy dalších systémů pro zobrazení dat, včetně původní aplikace zadavatele.

Čtvrtá kapitola „Návrh“ je věnována návrhu na vylepšení stávající aplikace zadavatele. Zaměřuje se na systém pro správu celého životního cyklu používaných prvků, od jejich nastavení až po zobrazení dat uživateli.

Následuje kapitola 5 „Realizace“ popisující samotný postup při implementaci. Kapitola vyzdvihuje podstatné části implementace. V této fázi byly implementovány i konkrétní prvky zobrazující relevantní data.

Šestá kapitola „Testování“ je zaměřena na testování implementovaného systému. Stručně přibližuje použití nástrojů pro testování a popisuje vzniklé problémy a jejich řešení.

Následuje kapitola 7, která obsahuje náměty na možná rozšíření práce.

Závěrečná kapitola 8 shrnuje dosavadní práci a uvádí další možná vylepšení.

Kapitola 2

Popis problému a motivace

2.1 Společnost FIEDLER AMS

Práce je zpracována pro společností FIEDLER AMS s.r.o. Společnost se věnuje vývoji, výrobě, instalaci a servisu telemetrických jednotek, dataloggerů a řídicích jednotek, senzorů a celých řešení převážně pro monitorování životního prostředí a řízení průmyslových provozů. Konkrétně se jedná o meteorologické a limnigrafické stanice, lokální varovné systémy, monitorování a řízení sítě pro rozvod pitné vody, čerpacích stanic, čističek odpadních vod a další řešení.

Jednotky získávají data ze senzorů a zároveň i informace týkající se připojených motorů (například dmyhadla, míchadla, čerpadla apod.). Data jsou vyhodnocena, upravena a zobrazena v některých případech už v jednotkách (dle typu jednotky), především však je tento komplexní soubor dat zpracován a zobrazen ve webové aplikaci CloudFM. [2]

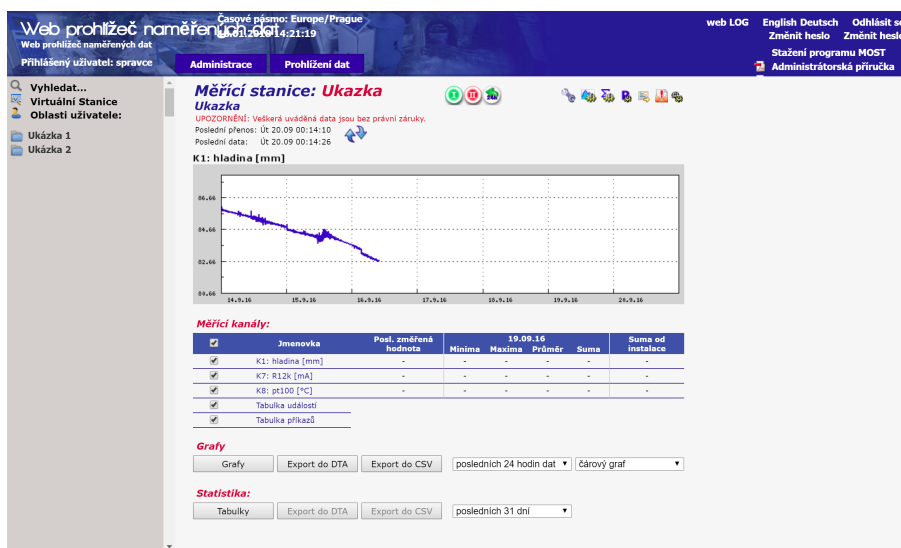
2.2 Aplikace zadavatele

Společnost využívá v současné době dvě webové aplikace pro správu a zobrazování dat:

2.2.1 Původní aplikace „Web prohlížeč naměřených dat“

„Tato aplikace nabízí zobrazení statických grafů a tabulek s nastavením časových parametrů. Dále je implementována možnost exportovat data do několika formátů. Mimo jiné také nabízí zobrazení několika stavů měřící stanice a informace o datových přenosech stanice. Aplikace pokrývá většinu zákaznických požadavků, nicméně existují i požadavky, kterým aplikace momentálně není schopna vyhovět.“ [1] Dále nepodporuje novou řadu telemetrických jednotek, čímž je znevýhodněna oproti nové verzi, která je zpětně kompatibilní se starou verzí jednotek.

Aplikace je napsaná v jazyce PHP a je stále využívána především pro vizualizaci a správu dat ze starších typů telemetrických jednotek společnosti. Využívají ji zejména stávající zákazníci, kteří potřebují některé funkce, které ještě nejsou v nové aplikaci přístupné (např. export na server hladiny.cz). Další skupina zákazníků je konzervativní a preferuje použití starší aplikace, protože si na ni zvykla a orientuje se v ní. Zákazníci nicméně postupně přecházejí na novou verzi.

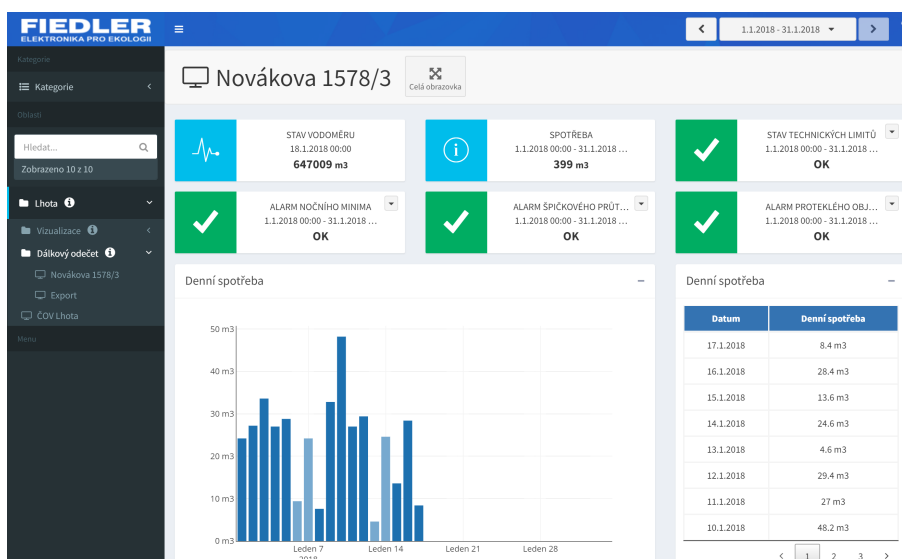


Obrázek 2.1: Web prohlížeč naměřených dat

2.2.2 Aktuální aplikace „CloudFM“

Společnost vyvíjí a postupně zavádí novou aplikaci pro správu a zobrazení naměřených dat. Tato aplikace je postavena na frameworku Aurelia.js a za využití moderních webových technologií, tj. REST API, Bootstrap apod. Data jsou zobrazena v uživatelských obrazovkách (dashboards) prostřednictvím řady interaktivních prvků (widgety), tj. různé typy grafů, tabulek, map, SCADA obrazovky apod.

Celá aplikace i jednotlivé widgety jsou vyvíjeny a editovány podle aktuální potřeby zákazníka. S rostoucími požadavky na jejich variabilitu dochází k potřebě udělat jejich podrobnou analýzu a vytvořit pravidla pro jejich tvorbu a správu. Motivací a cílem předložené práce je, aby byla aplikace vyvíjena co nejefektivněji a zároveň byla udržitelná, flexibilní vůči požadavkům uživatele, rozšiřitelná do budoucna, intuitivní a uživatelsky přívětivá. [1]



Obrázek 2.2: Portál CloudFM

2.2.3 Cíle práce

- Provedte analýzu aktuálně používaných prvků (widgetů) v různých systémech uživatelských obrazovek (dashboards) a v systému uživatelských obrazovek zadavatele.
- Provedte analýzu datových toků vybraných widgetů.

- Proveďte analýzu možností konfigurace vybraných widgetů.
- Na základě analýzy vytvořte návrh a implementaci systému (frameworku), který zajistí správu celého životního cyklu widgetu, od jeho konfigurace, předzpracování dat, až po vykreslení těchto dat uživateli.
- Proveďte analýzu, návrh a implementaci několika widgetů za pomoci navrženého systému.
- Vytvořte unit testy pro vytvořený systém.

Kapitola 3

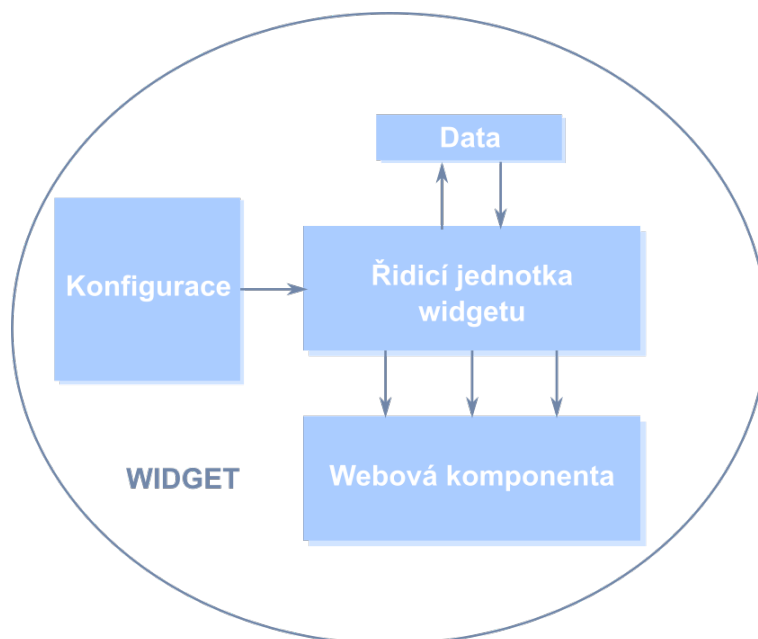
Analýza současného stavu

Tato kapitola obsahuje rozsáhlou analýzu widgetů, popisuje jejich strukturu a jednotlivé části. Zvláštní pozornost je věnována widgetům v současném aktuálně používaném systému (viz 2.2.2 CloudFM), který společnost využívá. Dále se práce zabývá užitím widgetů v dalších systémech vyhovujícím zadaným požadavkům.

3.1 Widget

Widget je prvek pro vlastní zobrazení dat, se kterým mohou uživatelé komunikovat. Jedná se o grafickou část uživatelského rozhraní. Libovolným kombinováním widgetů do bloků si může uživatel poskládat specifický dashboard, tj. uživatelskou obrazovku. [3]

Každý widget se sestává z řídicí jednotky, uživatelské konfigurace, dat a webových komponent (viz Obrázek 3.1). Řídicí jednotka widgetu požádá o data, která zpracuje podle svého nastavení a uživatelské konfigurace. Takto zpracovaná data pošle do webové komponenty společně s jejím nastavením. Webová komponenta slouží jako výstupní prvek, dojde tedy k zobrazení dat.



Obrázek 3.1: Struktura widgetu

3.1.1 Konfigurace a řídicí jednotka widgetu

Webová komponenta slouží k vlastnímu zobrazení dat a nastavení. Na základě potřeb zadavatele budeme v této práci uvažovat data a nastavení, která jsou jí předána prostřednictvím řídicí jednotky widgetu.

Uživatelská konfigurace vs. pevné nastavení widgetu

Nastavení a konfigurace slouží k předzpracování dat a obsahují volby pro úpravu a upřesnění vzhledu widgetu. Zařadit sem můžeme parametry jako je barva prvku, pozice a styl textu nebo zdroj dat.

- Nastavení řídicí jednotky (dále jen „Nastavení“)
 - Pro uživatele výchozí parametry.
 - Nastavení widgetu je zakomponované v řídicí jednotce.
 - Uživatel nemá možnost nastavení měnit.

- Uživatelská konfigurace (dále jen „Konfigurace“)
 - Konfigurace je prováděná uživatelem - pro uživatele přizpůsobené parametry.
 - Obsahuje nabídku pro úpravu parametrů.

PŘÍKLAD - Rozdíl mezi nastavením a konfigurací

Tento příklad slouží k bližšímu pochopení rozdílu mezi nastavením a konfigurací. Na příkladu lze pozorovat, jakým způsobem lze rozložit volby pro konfiguraci a nastavení řídicí jednotky.

Zpočátku budou diskutována okrajová řešení, tzn. maximální možnost konfigurace, poté maximální možnost nastavení s minimální možností konfigurace, následně obě možnosti v rovnováze.

Budeme uvažovat zobrazení widgetu popsaného níže:

VÝSTUP: ZOBRAZENÍ VÝŠKY HLADINY

WEBOVÁ KOMPONENTA: GRAF

TYP GRAFU: ČÁROVÝ

VLASTNOSTI GRAFU:

- ŠÍŘKA ČÁRY: 2 MM
- BARVA GRAFU: MODRÁ
- VYPLNĚNÝ GRAF: ANO
- HORNÍ LIMIT: 50 CM
- SPODNÍ LIMIT: 10 CM
- ZOBRAZIT HODNOTY: ANO
- ZOBRAZIT LEGENDU: ANO

VLASTNOSTI WIDGETU:

- NÁZEV: GRAF
- ZOBRAZIT NÁZEV: ANO
- FONT: ARIAL
- VELIKOST PÍSMO: 10 PX

DATA:

- ID STANICE: 123456
- DATOVÝ TOK: HLADINA

1. Maximální přizpůsobení na straně uživatelské konfigurace

- Tato varianta předpokládá, že se v nabídce nachází seznam widgetů dle základních webových komponent, např. graf. Následně se uživateli zobrazí nabídka pro ovlivnění vzhledu grafu.
- V tabulce 3.1 lze pozorovat, že uživatel má možnost graf pojmenovat, změnit barvu a šířku čáry grafu, vyplnit graf, vybrat datový tok apod. Ovlivnit nemůže to, že se u grafu vždy objeví i jeho název.
- Výhodou této kombinace je, že si uživatel navolí do posledního detailu vzhled komponenty. Nevýhodou je, že pro více widgetů bude opakovaně volit z velkého množství parametrů, často pro velmi podobné komponenty.

Vnitřní nastavení řídicí jednotky widgetu	Uživatelská konfigurace
<ul style="list-style-type: none"> - graf - zobrazit název grafu 	<ul style="list-style-type: none"> - čárový graf - šířka čáry 2 mm - zobrazit popisky u jednotlivých hodnot - datový tok Hladina - název grafu Hladina - vyplnit graf - barva modrá - horní limit 50 cm - spodní limit 10 cm

Tabulka 3.1: Maximální nastavení v uživatelské konfiguraci

2. Maximální nastavení na straně řídicí jednotky

- Nabídka obsahuje widgety s již přesně nadefinovanými vlastnostmi. Příkladem je levý sloupec tabulky 3.2, kdy výběr konkrétního widgetu odpovídá i výběru toho, o jaký typ grafu se bude jednat, jakou bude mít barvu, limitní hodnoty a v neposlední řadě jaké datové toky bude zobrazovat.

- Tato varianta umožňuje uživateli graf pojmenovat a povolit zobrazení jeho názvu.
- Výhodou diskutovaného řešení je, že uživatel není nucen vyplňovat a volit rozsáhlou řadu parametrů. Na druhou stranu nelze příliš ovlivňovat vzhled widgetu.
- V popisovaném návrhu dochází ke kritickému omezení dalšího použití widgetu. Kritické hodnoty jsou zvýrazněny v tabulce 3.2. Daný graf není přenositelný a použitelný pro jiné případy, neboť nelze změnit datový tok a limitní hodnoty. Dá se totiž očekávat, že uživatel bude chtít zobrazovat i jiné naměřené hodnoty s odpovídajícími limitními hodnotami.

Vnitřní nastavení řídicí jednotky widgetu	Uživatelská konfigurace
<ul style="list-style-type: none"> - graf - čárový graf - šířka čáry 2 mm - styl písma Arial, 10 px - zobrazit popisky u jednotlivých hodnot - datový tok Hladina - barva modrá - horní limit 50 cm - spodní limit 10 cm - zobrazit legendu 	<ul style="list-style-type: none"> - název grafu Hladina - zobrazit název grafu

Tabulka 3.2: Maximální nastavení v řídicí jednotce

3. Částečná konfigurace a nastavení

- Toto řešení se snaží najít rovnováhu v tom, jaká část nastavení widgetu bude skryta / odkryta uživateli.
- Součástí nastavení widgetu pro čárový graf je např. šířka čáry, styl písma a popisky u jednotlivých hodnot. Uživatel může ovlivnit, např. barvu grafu, datový tok a limitní hodnoty, viz tab. 3.3.

Vnitřní nastavení řídicí jednotky widgetu	Uživatelská konfigurace
<ul style="list-style-type: none"> - graf - čárový graf - tloušťka čáry 2 mm - styl písma Arial, 10 px - zobrazit popisky u jednotlivých hodnot 	<ul style="list-style-type: none"> - datový tok Hladina - název grafu Hladina - zobrazit název grafu - vyplnit graf - barva modrá - horní limit 50 cm - spodní limit 10 cm - zobrazit legendu

Tabulka 3.3: Částečná konfigurace a nastavení

Univerzální vodítko

Najít univerzální vodítko pro optimální rozdělení nastavení a konfigurací patří mezi cíle této práce.

Aplikace takového vodítka pak využívá výhod obou okrajových řešení, tzn. řešení je flexibilní požadavkům uživatele a zároveň jej nezatěžuje nastavováním opakujících se a podrobných voleb. Nabídka widgetů je pak dostatečně variabilní a je snazší se v ní orientovat, což vede k intuitivnímu použití celé aplikace. Konkrétní návrhy řešení jsou diskutovány v kapitole Návrh.

3.1.2 Kategorizace widgetů

Widgety lze kategorizovat dle webové komponenty a jejich typů, dle způsobu užití nebo typu využitých dat. Widgety lze kategorizovat:

- z pohledu uživatele
 - kategorie, které jsou viditelné uživateli.
- dle řídicí jednotky widgetu

– objektový návrh

Tato práce vychází ze systému modulárního rozšíření, viz práce [1].

Kategorizace dle webové komponenty

Rozdělení widgetů dle použitých typů webových komponent patří mezi typické klasifikace. Jedná se o kategorizaci, o kterou se do velké míry opírá současné řešení CloudFM, viz kapitola 3.3.1.

Z pohledu uživatele Taková klasifikace usnadňuje orientaci uživatelů při výběru vhodného widgetu. Na druhou stranu je tato klasifikace nedostatečná pro širokou škálu widgetů nabízených aplikací CloudFM.

- Přehledné pro uživatele
- Omezené na typy webových komponent

Dle řídicí jednotky Rozdílné webové komponenty znamenají rozdílný kód řídicí jednotky widgetu. Je proto vhodné tuto klasifikaci zahrnout do celkového objektového návrhu.

Kategorizace dle formátu dat a jejich transformace

Závisle na návrhu lze widgety kategorizovat dle formátu dat, bližší informace se nacházejí v kapitole 3.2.5. Příkladem je rozdělení na widgety, které zobrazují binární, analogová, nebo textová data.

Z pohledu uživatele Jedná se o řešení, které se může jevit, pokud aplikované samostatně, nepřehledné pro běžného uživatele.

Dle řídicí jednotky Všechny widgety pracují s daty. Formát dat, jejich transformace a často i jejich obsah se nastavuje v řídicí jednotce.

- Předpoklad znalosti formátů dat a operací, které je nad nimi potřeba provádět
- Pro jeden typ dat několik možných zobrazení

Kategorizace dle způsobu užití widgetu

Widgety lze použít k zobrazení nebo porovnání dat, k jejich změně a exportu, nebo např. jako odkaz na jinou webovou stránku. Pro jeden způsob užití se nabízí několik možných webových komponent a jejich typů. Zároveň lze do jednoho widgetu zakomponovat více způsobů užití.

Kategorizace dle způsobu užití stanic

Z pohledu uživatele Tato klasifikace je typická pro užití stanic zadavatele. Stanice lze užít pro management řady technologických řešení. Ve více případech užití se mohou objevovat stejné widgety, až na ty velmi specifické pro dané technologické řešení. U monitorování meteorologických veličin budou uživateli zobrazeny jiné widgety než u monitorování ČOV, nebo dálkového odečtu elektroměrů.

Nabízí se několik konkrétních řešení, podle kterých by bylo možné widgety rozdělit, nicméně v nabídce je rozsáhlá řada variabilních technologických řešení dle měnících se požadavků zákazníka.

- Přehledné pro zákazníky s typickými technologickými řešeními
- Předpoklad znalosti všech řešení
- Opakování widgetů pro různá řešení

3.2 Data a jejich kategorizace

Widgety zobrazují strukturovaná data, která jsou zpravidla předzpracovaná na straně serveru. Jejich další transformace probíhá na straně widgetu, resp. klienta. Tato kapitola se zaměřuje na kategorizaci dat a popisuje jejich formát, strukturu, trendy, vzorce a vzájemné vztahy.

Data lze rozdělit z několika úhlů pohledů:

3.2.1 Zdroj dat

Tato práce se zabývá především zdroji dat, která jsou relevantní k činnosti společnosti a mohou zákazníkům přinést užitečné informace. Jedná se především o data z měřicích a řídicích jednotek, připojených senzorů, z webových aplikací apod.

- Data surová ze senzorů
 - Surová data se začínají využívat ve značné míře pro tzv. IoT aplikace, kdy na server přichází velké množství dat s vysokou granularitou. Taková data lze snáze doplnit o další data a integrovat do jiného systému. Zároveň je jednodušší s nimi pracovat, pokud jsou zatím nejasné požadavky na jejich využití a lze očekávat specifické požadavky na jejich analýzu a následnou prezentaci. [7]
- Data ze senzorů upravená (transformovaná) v jednotce
 - Telemetrické jednotky a data loggery mohou data ze senzorů upravit na základě jejich typu. Např. je k jednotce připojen hladinoměr. Jednotka má informace o typu hladinoměru a je zkalibrována tak, aby vypočítala skutečnou výšku hladiny. Na server se pak dostanou komplexní data, ze kterých lze zjistit nejen výšku hladiny v udaných jednotkách s časovou, ale také typ měřeného média a přesnost sensoru.
- Data z jednotky
 - Jednotky neposílají pouze data ze senzorů. V závislosti na typu jednotky posílají řadu dalších informací, např. GPS souřadnice měrného místa, způsob komunikace, nebo identifikaci technika, který ji nastavoval.
- Data přidaná na serveru
 - Data přijatá na serveru mohou být modifikována, nebo rozšířená o další data, a to na základě nějaké podmínky. Takovou podmínkou může být např. typ jednotky, nebo měřeného média. Jedním z důvodů je zachování konzistence souboru dat k jednotlivým typům jednotek a také možnost doplnění metadat.

- Data z konfigurace
 - Mezi další data, která lze interpretovat a dále zpracovávat, patří nastavení jednotky. Zde je možné vyčíst a upravit, jak často jednotka odesílá data, co měří apod.

3.2.2 Účel dat

Účel dat je v této práci diskutován s návazností na řešený systém pro zobrazování dat.

- Naměřené hodnoty
 - Jednou z nejdůležitějších informací pro koncové uživatele aplikace je naměřená hodnota a její trend vývoje. Telemetrické jednotky posílají naměřenou hodnotu s časovou značkou, dále často změnu hodnoty (tzn. rozdíl proti předchozí hodnotě) a sumu za časový rámec.
- Informace o jednotce a měrném místě
 - Jednotka obsahuje identifikační a další údaje. Množství a typ údajů závisí na typu jednotky. Další metadata jsou k jednotce připojena na straně serveru, např. informace o měrném místě.
- Události
 - Další žádanou informací jsou události, tzv. eventy. Eventy obsahují informace o komunikaci jednotky se serverem, stavu baterie a dalších parametrů, které pomáhají zajistit celkový přehled nad měrným místem a dohledání problémů.
- Nastavení / Konfigurace jednotek
 - Nastavení slouží k ověření správného nastavení jednotky a prohlížení historie změn její konfigurace. V aktuální aplikaci se samotná konfigurace pak provádí v průvodcích konfigurací jednotek.
- Data vzniklá dalším zpracováním získaných dat

- S daty se nadále pracuje na straně serveru, kde jsou analyzována, transformována a rozšířena o další data. Na základě takového zpracování je možné prezentovat četnost dat, jejich závislosti, chybovost a další statistické údaje (viz kapitola 3.2.6). Jedná se to data, která přidávají k dostupným datům další informace na základě nějakého kontextu.

3.2.3 Metadata

Metadata jsou strukturovaná (v našem případě popisná) data o datech. Obsah metadat může být jednoduše zobrazen, nebo může ovlivnit finální zpracování a prezentaci naměřených dat.

- Informace o lokalitě
 - V našem případě se jedná především o data, která obsahují informace o lokalitě, odkud data pocházejí, tzn. adresa, GPS souřadnice, majitel zařízení, bližší informace o lokalitě apod. Tato data jsou přidělena vždy konkrétnímu měrnému místu, na kterém se telemetrická jednotka nachází. Často mohou sloužit k administrativním účelům.
- Metadata ze stanice / jednotky
 - Společnost vyrábí řadu telemetrických jednotek a data loggerů, které mají rozdílné vlastnosti. Základní informace jsou uloženy v metadatech. Metadata současně odkazují na webové stránky společnosti, kde lze zjistit detailní informace.
- Metadata o uživatelské obrazovce
 - Tato data pomáhají v orientaci v rámci nejen jednotlivých dashboardů (např. počty a typy widgetů, id relevantních jednotek, kanály apod.), ale také v hierarchii umístění dashboardů (např. související dashboardy, nadsložka/nadkategorie, ..)
 - Tato metadata mohou zahrnovat i informace o uživateli, kteří mají k dané uživatelské obrazovce přístup, a o jejich oprávněních.

- Stav dat

- Dalším typem metadat může být stav dat. V určitých situacích je potřeba hodnoty za časový úsek nějak označit - dát jim stav. Data můžeme označit na základě hodnot jiných dat, tzn. strojově, nebo uživatelsky. Data mají výchozí (předpokládaný) stav - závislé na kontextu. Data mohou být např.:

- * (Ne)Ověřená

- Data u kterých (ne)máme ověřeno, že jsou správná.

- * (Ne)platná

- Může se jednat o textová data, která je(není) nutné aktualizovat. Tuto informaci můžeme získat např. z metadat o lokalitě

- * Testovací vs. V provozu

- Data, která vznikla při testování funkčnosti systému.

- * Chybná vs. Správná

- Data s chybnou hodnotou. Například víme, že při vysoké teplotě, nebo vlhkosti ztrácí čidlo přesnost a naměřené hodnoty jsou tedy nedůvěryhodné.

- * Chybějící vs. Kompletní

- Chybějící data způsobí, že se v časové ose objeví mezera, chybějící hodnota. Tato informace je důležitá pro zobrazení dat, kde se chybějící hodnota vypočítává / odvozuje za využití definovaného vzorce.

3.2.4 Využití dat

Tato kapitola pojednává o tom, jakým způsobem se data dají využít. Využití dat se liší v závislosti na uživateli. V následujících bodech jsou nastíněny základní způsoby:

- Zobrazení

- Informace, které uživatele zajímají jsou zobrazována prostřednictvím widgetů v podobě různých webových komponent. Jedná se o tabulky, grafy, mapy, KPI.

- Vyvolání události
 - Existuje celá řada informací, které uživatele nezajímají do té doby, než jsou odchýlená od normálu. Často to pak pro uživatele znamená, že by měl řešit nějakou situaci. Příkladem takové informace je vybitá baterie, přerušení komunikace, nebo vysoká hladina v nádrži. Taková data jsou součástí systému varovných zpráv a mohou být zároveň i součástí prezentace.
 - Tato data mohou automaticky spouštět další akce a procesy, např. varování nebo automatické řízení motorů.

- Archivace dat
 - Nemalý soubor dat se pouze archivuje, tzn. nedochází k jejich zobrazení, ani nejsou využívána k indikování událostí, ale jsou uložena pro pozdější šetření. Je to často z důvodu, že mohou být nápomocná při hledání informací v budoucnu, nebo z legislativních důvodů.

- Další využití
 - Dále lze data použít, např. ke zkoumání, k predikci, zjištění souvislostí mezi daty nebo k vytvoření matematického modelu.

3.2.5 Formát dat

- Základní
 - * Binární
 - * Analogová
 - * Textová
 - * Obrázek

- Složený
 - * Blob

- Blob je binární řetězec dat, který lze využít například k ukládání obrázků a dalších multimediálních souborů, slouží jako obálka pro surová data.
- * Datový stream / tok (DS)
 - Datové toky jsou objemné soubory dat s vysokou granularitou.
 - Datový tok lze definovat jako neomezenou posloupnost datových položek, které spolu mohou a nemusí souviset. Datové toky rychle narůstají na velikosti a je neudržitelné je jako nekončící celek ukládat, analyzovat a zobrazovat za pomoci konvenčních architektur pro zpracování dat.
 - Každá datová položka je obvykle označena časovým razítkem a v některých případech geografickou značkou.
 - Uživatelé mají obvykle zájem o recentní data namísto celého souboru dat. Nabízí se několik možností, jak s datovými toky operovat, např. ukládat a zobrazovat vždy jen data za určitý časový úsek, případně poslední hodnotu a ostatní data nezachovávat. [4]

3.2.6 Transformace dat

Data získaná ze senzorů často procházejí nějakou úpravou. Data jsou upravena tak, aby poskytla ve vlastní prezentaci přehledné a relevantní informace. Pro efektivní zpracování a zobrazování dat je nutné data, jejich trendy a vztahy s dalšími daty pochopit do hloubky. V rámci této práce se uvažují případy užití a tedy i potřebné transformace dat, které splňují požadavky a možné potřeby zadavatele.

Transformovaná data jsou tedy data, která prošla úpravou. V následujících bodech jsou vyjmenované základní uvažované transformace:

- Součet / rozdíl
- Násobení / dělení
- Zaokrouhlení
- Konzumpční rovnice
- Matematické operace mezi daty se stejnou časovou značkou

– Např. rozdíl vnitřní a venkovní teploty

- Četnost za období a další statistika
 - Např. kolikrát bylo za uplynulý měsíc spuštěno čerpadlo apod.
- Agregace dat
 - Do agregace dat se především řadí:
 - * minimální, maximální a průměrná hodnota za časový rámec
 - * suma za období (např. motohodiny)
 - * nejčastější (nejméně častá) hodnota
- Filtrování dat
 - první / poslední hodnota za časový rámec
- Klouzavý součet / průměr a uplatnění dalších trendových funkcí

Datové toky (viz 3.2.5) jsou náročné na výpočetní výkon a čas. Pro jejich efektivní použití je vhodné nad nimi provádět transformace, které jejich nevýhody minimalizují. Snahou transformací je tedy i jejich optimalizace a snížení objemu dat za současného zachování jejich nosných informací, nebo jejich částečnou ztrátou.

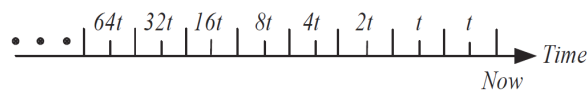
Možnosti transformací s cílem optimalizace datových toků (dle [4]) např.:

- Korekce dat
 - Nahrazení chybějících, nebo nesprávných hodnot odpovídající hodnotou.
- Metoda plovoucích oken
 - Zasílání pouze souboru dat, která náleží do časového rámce.
- Redukce duplicitních dat
 - Manipulace o datech, která mají zaznamenanou pouze změnu.
 - Výhodné u dat, která se často nemění, např. binární hodnoty
- Aproximace dat

- Redukce dat s podobnou hodnotou.
- Výhodné, pokud jsou pro zobrazení dostatečné orientační hodnoty.
- Kompresí časového rámce - „Tilted time frame“
 - U těchto metod dochází k naklonění časového rámce, např. přirozenému (viz obrázek 3.2 nahoře), nebo logaritmickému (viz obrázek dole).
 - Podrobněji jsou zobrazena data s časovou značkou nejbližší současnosti.



A tilted time frame with natural time partition



A tilted time frame with logarithmic time partition

Obrázek 3.2: Naklonění časového rámce [4]

3.2.7 Závislost dat na čase

Data mohou a nemusí být závislá na čase:

- Časově nezávislá data
 - Jedná se o data, která se s časem nemění, např. datum instalace a GPS souřadnice měrného místa.
- Časově závislá data
 - Hovoříme o nich, pokud se jedná o data, která se mění s časem a je tedy nutné, aby měly časovou značku.
 - * Data se s časem mění pravidelně
 - Při každém získání dat je očekávaná nová hodnota, např. měření teploty.

- Do této kategorie patří data, která přicházejí v pravidelných intervalech, ale i data, která chodí výjimečně, např. alarmový stav.
- * Data se mění nepravidelně
 - Data se mohou v určitých případech změnit. Například došlo k výměně sensoru na měrném místě a v metadatech je zapsána identifikace nového sensoru a nové datum jeho instalace.
- * Data aproximovaná na základě výpadku
 - Data lze měnit i do minulosti. V případech, kdy určitá část dat chybí je snaha data nahradit předpokládanou hodnotou (na základě statistických výpočtů). Jakmile jsou dostupná chybějící data, načtou se zpětně a reálné hodnoty nahradí ty předpokládané.
- * Data s omezenou vypovídající hodnotou
 - Data, která ztrácejí s časem na relevantnosti. Příkladem jsou datové toky, např. logy, které je vhodné indexovat za časový rámec, např. za rok, měsíc, den. Poté se s nimi pracuje jako s daty, které se pravidelně v čase mění.

[6]

3.3 Analýza současných widgetů

Současné uživatelské obrazovky využívají systému modulárních rozšíření: „*System modularních rozšíření je navržen tak, aby umožnil rychlý vývoj nových rozšíření, během kterého je možné využít co největší část dříve implementovaných funkcí, ale zároveň dodržuje nezávislost jednotlivých rozšíření.*“ [1].

Použité technologie:

- REST API, JSON, JavaScript, CSV

Tento systém klade důraz na flexibilitu aplikace a na znovupoužitelnost kódu. Využívá dědičnosti a je hierarchicky rozdělen do 3 generací tříd. V nejvyšší kategorii je widget, od kterého dědí obecně podle užití webové komponenty graf, tabulka a mapa, následují konkrétní užití (rozděleny často podle vlastností webové komponenty), např. čárový graf. Při

používání aplikace a jejím rozšiřování došlo i k rozšíření nabídky widgetů a jejich nastavení a konfigurace.[1]

3.3.1 Kategorizace současných widgetů

V současné době jsou widgety rozdělené do kategorií především podle typu použité webové komponenty:

- Grafy
- Tabulky
- Diagramy
- Obrázky
- KPI
- Mapy
- SCADA obrazovky
- Konfigurace
- Ostatní

Další rozdělení je především podle typu komponenty, podle použití, podle použité transformace dat, která jsou na webovou komponentu aplikována, př. (grafy) zobrazení spotřeby, čárový graf, binární graf apod. Pro každou takovou podkategorii je vytvořen samostatný widget.

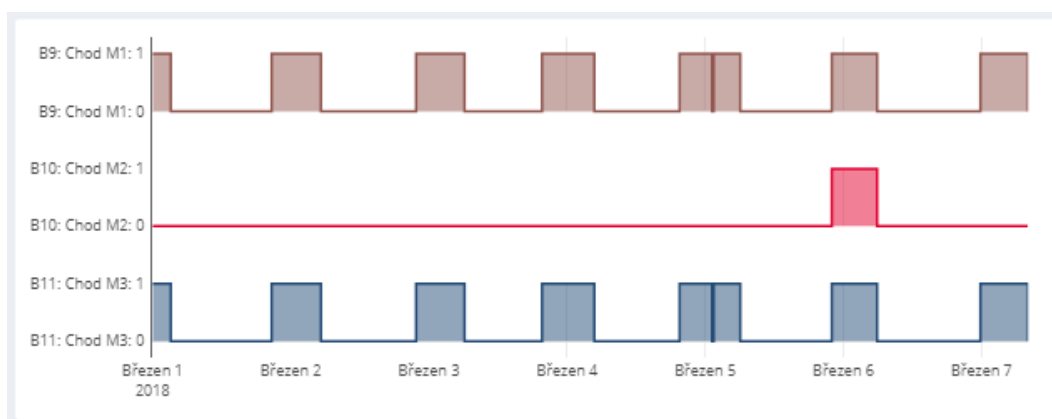
3.3.1.1 Grafy

Grafy slouží k zobrazení binárních, nebo analogových datových toků. Tato data se zpravidla mění v čase a obsahují tedy časovou značku. Můžeme zvolit, od jakého data chceme data zobrazovat (obvykle datum instalace). K vykreslení grafů používá aplikace otevřenou knihovnu Plotly.js (viz [14]).

- V konfiguraci volíme u grafů:
 - stanici a datové toky, které chceme prezentovat
 - název a popis widgetu
 - povolení zobrazení názvu widgetu

1. Binární graf

- Tento widget je určený pro zobrazení stavů v podobě binárních dat (0/1).
- Mezi parametry konfigurace patří například (mimo standardní parametry grafu):
 - změna pojmenování datových toků
 - zobrazení názvů datových toků na ose y¹



Obrázek 3.3: „Binární graf“ - chod motorů M1 - M3

2. Čárový graf

- Čárové grafy jsou určeny k rychlému přehledu nad trendy číselných hodnot z analogových kanálů. Konkrétní analogové kanály vybere uživatel v uživatelské konfiguraci.

(a) Čárový

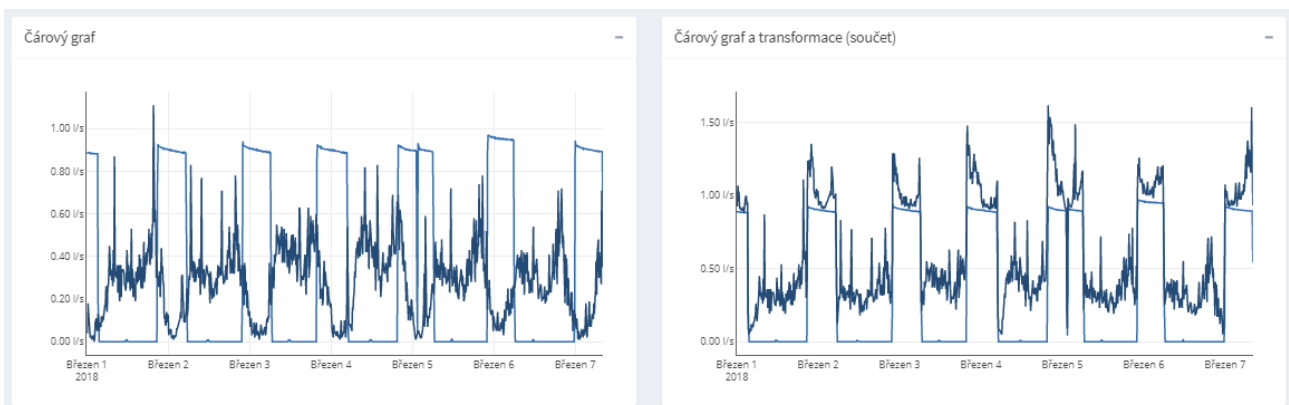
- Jedná se o čárový graf, u kterého jsou všechny zobrazované datové toky upraveny stejným způsobem, např. ke všem je přičtena hodnota 10.
- Mezi parametry konfigurace patří například (mimo standardní parametry grafu):

¹Názvy jednotlivých kanálů odpovídají názvům uloženým ve stanici, kde je jeho délka omezena na počet znaků.

- transformace dat
- vyplnění křivky (např. hladina)
- nastavení limitů
- přejmenování datových toků
- zobrazení tarifu (např. spotřeba energií)

(b) Čárový graf a transformace

- Tento čárový graf zobrazuje 2 trendy. Jedním je datový tok kanálu 1 a druhým je např. součet datových toků kanálů 1 a 2.
- Mezi parametry konfigurace patří například:
 - typ operace mezi jednotlivými datovými toky
 - nastavení krajních hodnot os
 - výška grafu
 - zaokrouhlení konečné hodnoty

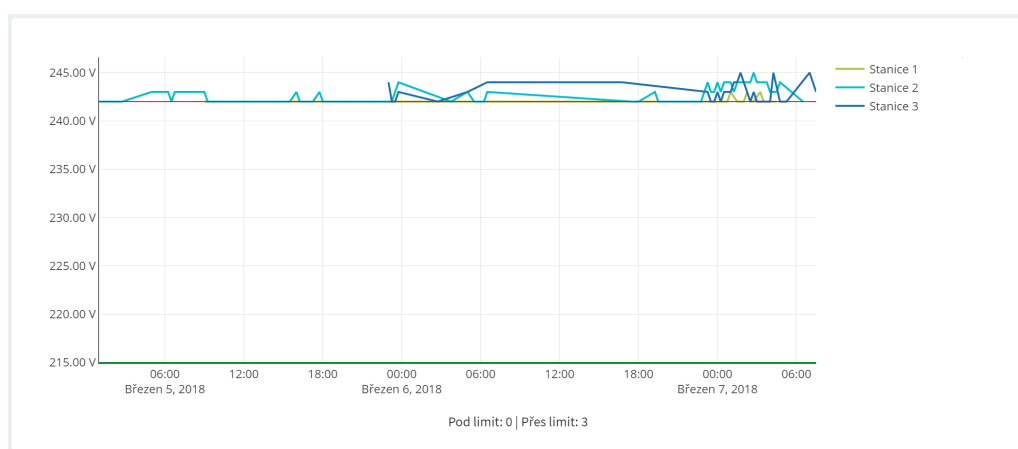


Obrázek 3.4: Příklady čárových grafů - nalevo graf čárový jednoduchý, napravo čárový graf s transformací. Pro názornost zobrazují oba grafy hodnoty ze stejných kanálů a za stejné období.

(c) Limity

- Tento graf zobrazuje pouze ty datové toky, jejichž hodnota přesáhla ve stanoveném časovém rámci nastavený limit. Na obrázku 3.5 jsou zobrazeny stanice, které naměřily vyšší napětí než je 242 V.
- Tento graf se odlišuje také v tom, že zde uživatel nevybírá jednotlivé stanice, ale přímo název skupiny, kam dané stanice patří. Widget pak zpracovává data a filtruje data ze všech stanic ze skupiny.

- Mezi parametry konfigurace patří například:
 - skupina stanic (jiná)
 - datový tok, např. napětí – transformace dat
 - hodnota limitu (může být několik) – měřicí jednotky
 - barva limitu (pro každý limit v datovém toku)

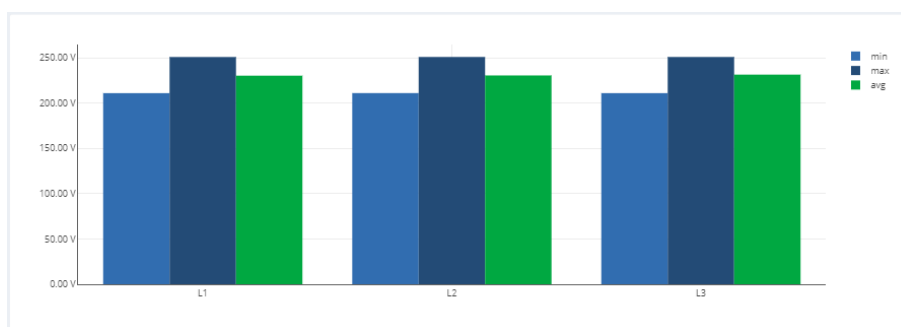


Obrázek 3.5: Čárový graf „Limity“

3. Sloupcový graf

(a) Minima/Maxima/Průměr

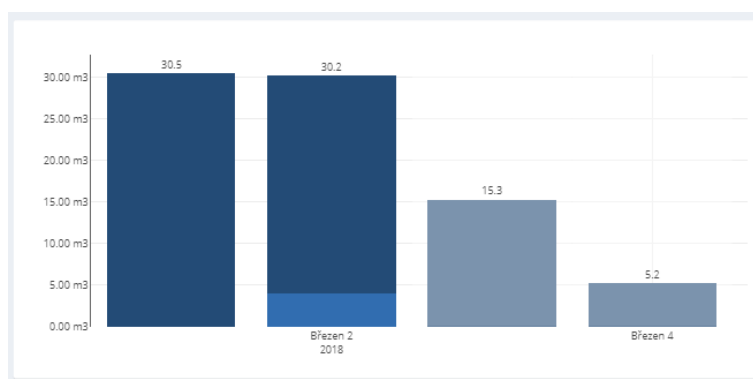
- Každý sloupec tohoto grafu za dané období reprezentuje agregovaná data, konkrétně maximální, minimální a průměrnou hodnotu kanálu.
- Na obrázku 3.6 zobrazuje graf agregované hodnoty 3 kanálů L1 - L3.
- Widget nabízí v uživatelské konfiguraci pouze nabídku platnou pro všechny grafy, tzn.:
 - název a popis widgetu – zobrazení názvu datového toku
 - stanice a datový tok – výška grafu
 - měřicí jednotky



Obrázek 3.6: Příklad sloupcového grafu s agregovanými daty

(b) Spotřeba

- Tento sloupcový graf se využívá především pro zobrazení spotřeby za období. Obvykle je v každém sloupci suma za den.
- Mezi parametry konfigurace patří například:
 - operace mezi datovými toky podle typu měřidla, např.
 - * vodoměr (1 tok)
 - * sdružený vodoměr (2 toky, každý sloupec zvlášť)
 - * vodoměr se zpětným průtokem (2 toky, které se od sebe odečítají)
 - transformace dat
 - měřicí jednotky
 - okrajové / mezní hodnoty (limity)



Obrázek 3.7: Příklad sloupcového grafu se spotřebou u sdruženého vodoměru

4. Výšečový

- Výšečový graf je vhodný pro znázornění poměru nebo procentuálního zastoupení u analogových datových toků.
- V konfiguraci widgetu lze nastavit, jestli budou zobrazeny popisky a procentuální zastoupení.
- Mezi parametry konfigurace patří například:
 - Zobrazení legendy
 - Procentuální zobrazení

5. Větrná růžice

- Jedná se o speciální graf pro zobrazení rychlosti a směru větru - analogové kanály.
- Mezi parametry konfigurace patří například:
 - Specifikace typu datového toku (rychlost / směr větru)
 - Intervalové koncové body
 - Zobrazení hodnot mimo interval

3.3.1.2 Tabulky

Podobně jako u grafů, tabulky slouží k prezentování binárních, nebo analogových dat, která se mění v čase a obsahují tedy časovou značku. Standardně můžeme zvolit, od jakého data chceme data zobrazovat (obvykle datum instalace). V konfiguraci volíme obvykle stanici a kanály s daty, která chceme zobrazit, název a popis grafu a zda chceme jeho název zobrazit.

Tabulky lze využít i k zobrazení dat, která na čase závislá nejsou a také textová data, např. popis lokality.

1. Tabulky s daty závislými na čase

- Tyto tabulky zobrazují často stejná data jako grafy. Z nabídky v uživatelské konfiguraci vybíráme:

- stanici a kanály s daty, která chceme prezentovat
- název a popis widgetu
- povolení zobrazení názvu widgetu

(a) Změny binárních hodnot

- Tabulka obsahuje informace o změnách hodnot (událostí) jednotlivých binárních kanálů, např. chod relé(zapnuto/vypnuto). Každá změna je znamenána ve formátu:

- | | |
|--------------------|--------------------------------|
| * Status, tzn. 0/1 | * Datum a čas začátku události |
| * Stanice | * Datum a čas konce události |

- Widget nabízí v uživatelské konfiguraci standardní nabídku pro tabulky, tzn.:

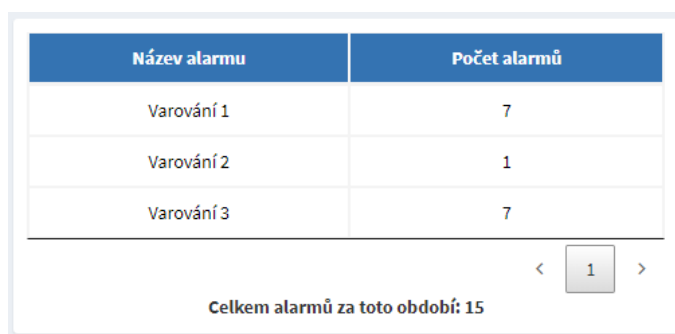
- * název a popis widgetu
- * stanice a datový tok

(b) Počet změn u binárních kanálů

- Tabulka zobrazuje počet případů za časový rámec, kdy byla hodnota binárního kanálu rovna 1, a to pro každý vybraný binární kanál zvlášť.

Tabulka zároveň obsahuje celkový počet pro všechny vybrané binární kanály dohromady.

- Na obrázku 3.8 se nachází ukázka typického užití této tabulky. Lze z ní jednoduše vyčíst, kolikrát došlo během časového úseku k alarmu.
- Widget nabízí v uživatelské konfiguraci standardní nabídku pro tabulky.



Název alarmu	Počet alarmů
Varování 1	7
Varování 2	1
Varování 3	7

< 1 >

Celkem alarmů za toto období: 15

Obrázek 3.8: Tabulka „Počet změn binárních kanálů“

(c) Hodnoty

- Tabulka zobrazuje všechny hodnoty (obvykle analogová data) jednoho či více datových toků za časový rámeček.
- Widget nabízí v uživatelské konfiguraci standardní nabídku pro tabulky.

(d) Spotřeba

- Každý řádek tabulky zobrazuje přírůstek za časový úsek, tzn. od poslední známé hodnoty. Jedná se o alternativní zobrazení sloupcového grafu, viz 3.7.
- Srovnání tabulek pro zobrazení jednotlivých hodnot a spotřeby obsahuje obrázek 3.9. Pro jednodušší srovnání vyobrazují tabulky data ze stejné jednotky (shodný analogový kanál).
- Mezi parametry konfigurace patří například:
 - * transformace dat
 - * měřicí jednotky

Čas	Hodnoty [m3]
23:00:00	33223200
22:00:00	33222800
21:00:00	33222300
20:00:00	33221900
19:00:00	33221300
18:00:00	33220600
17:00:00	33219100
16:00:00	33217500

Datum a čas	Spotřeba
6.3.2018 23:00:00	1.50 m3
6.3.2018 22:00:00	1.20 m3
6.3.2018 21:00:00	1.50 m3
6.3.2018 20:00:00	1.20 m3
6.3.2018 19:00:00	1.80 m3
6.3.2018 18:00:00	2.10 m3
6.3.2018 17:00:00	4.50 m3
6.3.2018 16:00:00	4.80 m3

Obrázek 3.9: Tabulky „Hodnoty“ a „Spotřeba“

(e) Průměr

- Tato tabulka vypisuje agregovaná data, konkrétně průměrné hodnoty za časový rámec. Obdobou této tabulky je sloupcový graf „Minima / Maxima / Průměr“, viz 3.6
- Widget nabízí v uživatelské konfiguraci standardní nabídku pro tabulky.

(f) Počet výskytů

- Tabulka obsahuje počet výskytů analogových hodnot u každé jednotky ve skupině (obvykle podle oblasti výskytu, DTS, ..).

Hodnota	Počet výskytů	%
214 V	6	0.02 %
215 V	12	0.04 %
216 V	21	0.07 %
226 V	655	2.20 %
227 V	807	2.71 %
229 V	1143	3.83 %
230 V	1331	4.46 %

Obrázek 3.10: Tabulka s počtem výskytů jednotlivých hodnot

- Na obrázku 3.10 je tabulka s počtem výskytů napětí u vybraných jednotek za stanovené období. Zvýrazněny jsou okrajové hodnoty, tj. < 215V,

a očekávaná hodnota, tj. 230V. Tabulka může dále zvýraznit průměrné hodnoty a hodnoty nad limitem.

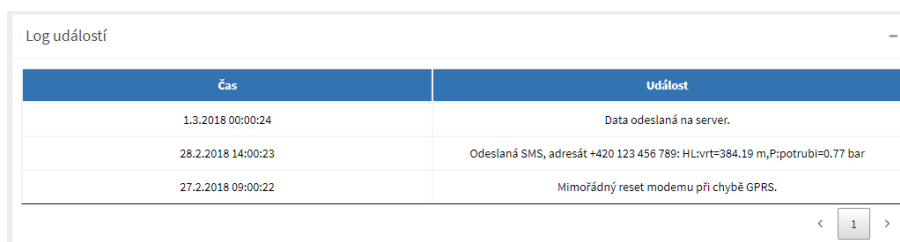
– Mezi parametry konfigurace patří například:

- * skupina stanic
- * měřicí jednotky
- * kanál, např. napětí
- * zobrazení prázdných intervalů
- * hodnoty limitů
- * rozptyl intervalu pro výpočet průměrné hodnoty
- * barvy limitů

2. Ostatní tabulky

- Log událostí

- Tato tabulka zobrazuje textová data o stavu jednotky a její komunikaci.
- Do logu událostí jsou zapisovány informace o každém odeslání dat na server, každém odeslání SMS, stavu signálu při zapínání modemu a také o výjimečných stavech, např. slabá baterie.
- Widget nabízí v uživatelské konfiguraci standardní nabídku pro tabulky.



Čas	Událost
1.3.2018 00:00:24	Data odeslaná na server.
28.2.2018 14:00:23	Odeslaná SMS, adresát +420 123 456 789; HLvrt=384.19 m,P:potrubí=0.77 bar
27.2.2018 09:00:22	Mimořádný reset modemu při chybě GPRS.

Obrázek 3.11: Tabulka „Log událostí“

- Základní informace

- Tabulka zobrazuje metadata, která jsou přidělena k měrnému místu.
- Příklad tabulky na obrázku obsahuje informace o lokalitě, na které je jednotka na instalovaná, o správci a majiteli měrného místa apod.
- Mezi specifické parametry konfigurace patří:
 - * položky s prioritou

- Uživatel může upravovat pořadí zobrazovaných položek.

Název	Hodnota
Měrné místo	104884-Sr
Adresa	Česká 10/1530
Číslo jednotky	103877877
Typ jednotky	H7 Hydro Controller
Majitel	Lubomír Novák
Měřená veličina	Hladina (m), Teplota vody (°C), Teplota vzduchu (°C)
Datum instalace	03/02/2018

Obrázek 3.12: Tabulka „Základní informace“

3.3.1.3 KPI

KPI jsou widgety, do kterých se vkládají nejdůležitější informace. Mají za úkol zajistit snadnou orientaci a rychlý přehled nad celkovou situací v rámci měřicí stanice, či lokality.

KPI nejsou omezené na formáty dat - mohou tedy zobrazovat data analogová, binární, textová apod. Slouží k zobrazení zpravidla jedné hodnoty z datového toku, nebo agregovaných dat, nebo krátkého textu.

Pokud zobrazují binární, či analogová data, lze u nich nastavit, že budou zobrazovat pouze hodnoty od data instalace.

1. KPI pro události

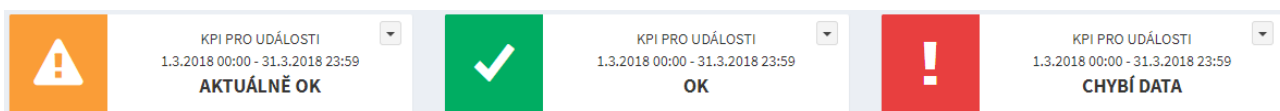
- KPI pro události jsou určeny pro binární data, která jsou závislá na čase. U těchto widgetů se jedná o kombinaci samotného KPI, binárního grafu a odkazu na jinou uživatelskou obrazovku.

– Zobrazení KPI

- * Vyhodnocuje na základě hodnot zvolených binárních kanálů (viz obrázek 3.13):

- Pokud je alespoň jeden v chybě (hodnota = 1), pak je widget zvýrazněn červeně a vypisuje chybu.

- Pokud jsou všechny v pořádku, ale ve sledovaném období byl alespoň jeden v chybě, je KPI zvýrazněno žlutě.
 - Pokud všechny binární kanály nabývaly hodnoty 0 za celý časový interval, pak je KPI zelené s popiskem "OK".
- Zobrazení binárního grafu
 - * KPI pro události jsou rozšířeny o binární graf, který zobrazuje hodnoty jednotlivých binárních kanálů za dané období.
 - * Uživatelská konfigurace odpovídá konfiguraci specifické pro binární graf, viz kapitola 3.3.1.1.
 - Odkaz
 - * Odkaz na uživatelskou obrazovku lze vybrat ze seznamu v uživatelské konfiguraci.
 - * V nabídce jsou odkazy na ty obrazovky, které obsahují průvodce pro konfiguraci konkrétních parametrů jednotky.



Obrázek 3.13: KPI „Události“

2. KPI pro události s kontrolou

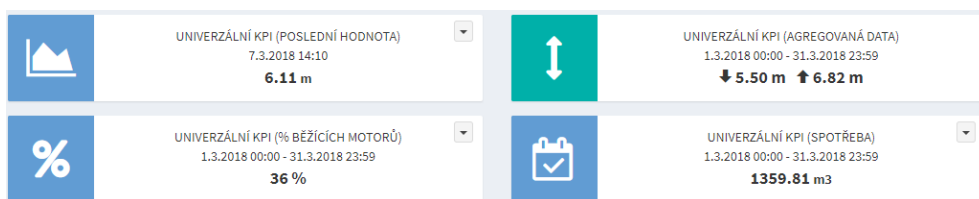
- Jedná se o období "KPI pro události", které je rozšířené o kontrolu, zda byla nastavena podmínka pro spuštění a ukončení události.
 - Pokud je podmínka nastavena, widget se chová stejně jako "KPI pro události".
 - Pokud podmínka nastavená není, widget má šedé pozadí a zobrazuje počet nenastavených událostí, jejich názvy pak v binárním grafu.

3. Univerzální KPI

- Univerzální KPI je další KPI, které je určené pro data závislá na čase. Zobrazuje

binární i analogová data. Vzhledem k flexibilitě obsahuje širokou škálu nabídky pro uživatelskou konfiguraci.

- Mezi specifické parametry konfigurace patří:
 - operace s daty
 - * poslední hodnota
 - poslední, obvykle analogová, hodnota z datového toku
 - * součet/odečet posledních hodnot
 - poslední, obvykle analogové, hodnoty z více datových toků
 - mezi hodnotami jsou provedeny zvolené operace a výsledná hodnota je zobrazena
 - * agregovaná data u 1 datového toku
 - minimální a maximální hodnota za časový interval
 - * rozdíl první a poslední hodnoty za období (1 datový tok)
 - přírůstek za časový interval
 - * rozdíl první a poslední hodnoty za období (2 datové toky)
 - celkový přírůstek, např. spotřeba u sdruženého vodoměru
 - * (invertované) procentuální zastoupení binární události
 - počet zastoupení určité hodnoty u binárního kanálu několika stanic
 - např. procento běžících motorů, nebo procento stanic, které odeslaly data
 - transformace dat
 - měřicí jednotky
 - ikona
 - * z dostupných knihoven (př. FontAwesome, Io-Icons, ...)
 - barva pozadí



Obrázek 3.14: Několik možností použití „Univerzálního KPI“

4. KPI Ukazatel signálu

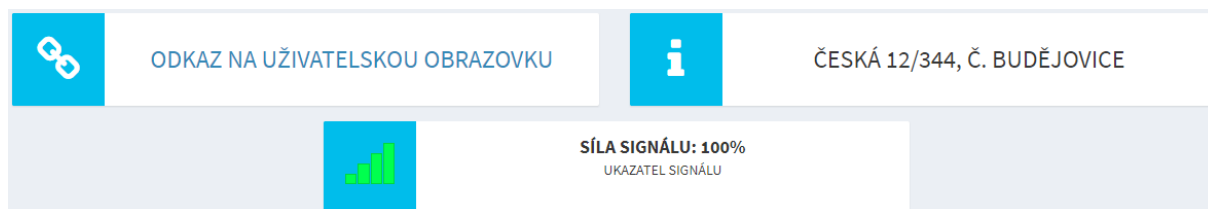
- Tento prvek obsahuje ikonu signálu, u které je její zbarvení závislé na síle signálu. Dále pak procentuální hodnotu síly signálu.
- Mezi parametry konfigurace patří:
 - Název a popis widgetu
 - Stanice a datový tok

5. KPI Odkaz

- Tento widget obsahuje odkaz na jinou uživatelskou obrazovku a text odkazu. To jsou také jediné položky, které lze upravit v uživatelské konfiguraci widgetu.

6. KPI Metadata

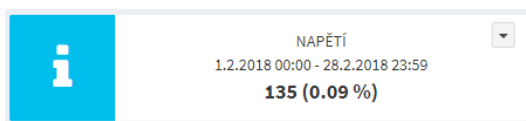
- Tento widget zobrazuje vybraná data z metadat.
- Mezi parametry konfigurace patří:
 - Soubor metadat
 - Pořadí



Obrázek 3.15: Prvky - „KPI Odkaz“, „KPI Metadata“ a „KPI Ukazatel signálu“

7. KPI Limity

- Tento widget zpracovává a zobrazuje statistická a agregovaná data získaná ze skupiny stanic. Konkrétně zobrazuje počet výskytů hodnot mimo limit (za daný časový úsek), obdobně jako tabulka 3.10.
- Na obrázku 3.16 je zobrazeno celkové a procentuální zastoupení počtu stanic, jejichž napětí přesahovalo v časovém intervalu nastavený limit.
- Mezi parametry konfigurace patří:
 - skupina stanic
 - datový tok
 - limity
 - ikonka, viz „Univerzální KPI“



Obrázek 3.16: KPI Limity

3.3.1.4 Mapy

Mapy slouží především k zobrazení umístění jednotky nebo měrného místa, nebo skupiny. Jednotlivé body mohou být odlišeny a mohou obsahovat stručný popis, nebo přímo naměřené hodnoty. K vykreslení map je použita knihovna Leaflet (viz [13]).

Využito je několik widgetů pro zobrazení v mapě. Všechny widgety lze v uživatelské konfiguraci pojmenovat.

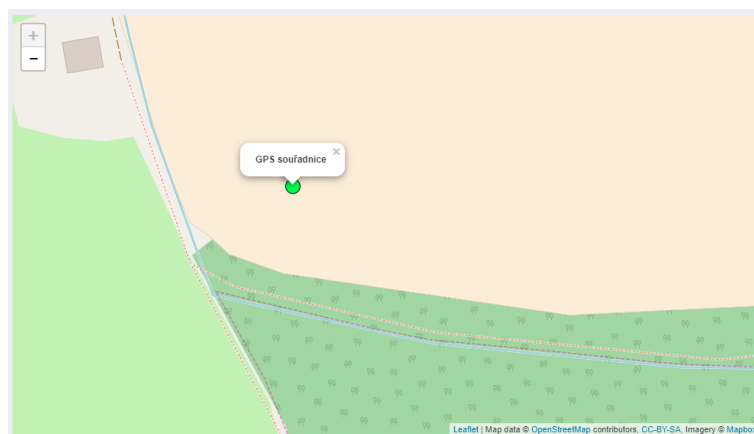
1. KML Mapa

- Tato mapa získává data z datové struktury KML.
- Na mapě se zobrazí body na souřadnicích vybraných stanic. U jednotlivých bodů je napsaná poslední hodnota vybraného datového toku.

- Mezi parametry konfigurace patří:
 - skupina stanic / stanice
 - datový tok
 - měřicí jednotky

2. Souřadnice

- Na mapě se zobrazí bod dle zadaných souřadnic. Popis u bodu je převzat z názvu widgetu.
- Mezi specifické parametry konfigurace patří GPS souřadnice.



Obrázek 3.17: Mapa „Souřadnice“

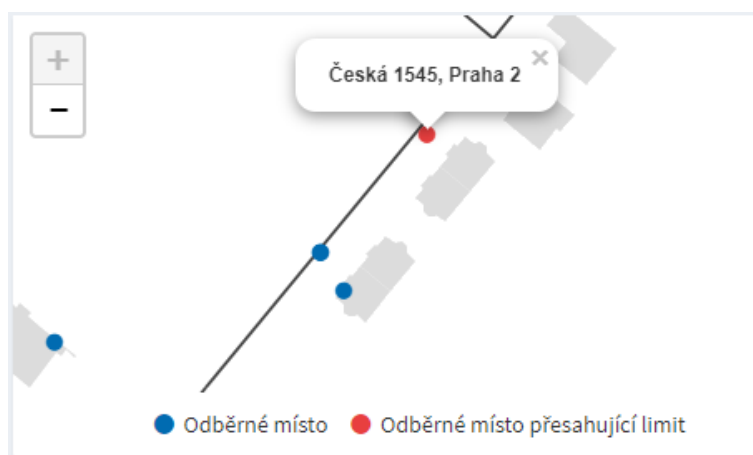
3. Agregovaná data na mapě

- Tato mapa zobrazuje agregovaná data na mapě.
- Na mapě se zobrazí body na souřadnicích vybraných stanic. U jednotlivých bodů je napsaná agregovaná hodnota vybraného datového toku.
- Mezi parametry konfigurace patří:
 - skupina stanic / stanice
 - datový tok
 - agregace dat, tzn. minimum, maximum, průměrná hodnota

- měřicí jednotky

4. Limity na mapě

- Obdobně jako tabulka na zobrazení počtu výskytů stanic mimo meze (viz Tabulky 1f), tak i tato mapa zvýrazňuje body stanic, u kterých se vybraný kanál nacházel v daném období mimo meze.
- Popis u bodu je převzat z metadat.
- Mezi parametry uživatelské konfigurace patří:
 - skupina stanic / stanice
 - limity
 - datový tok



Obrázek 3.18: Mapa „Limity“

3.3.1.5 Export dat

Widget „Export“ slouží k exportování vybraných dat. Exportovat lze binární, analogové i textové datové toky omezené časovým rámcem, a to do formátu CSV.

- Uživatelská konfigurace widgetů:
 - název a popis widgetu
 - povolení zobrazení názvu widgetu

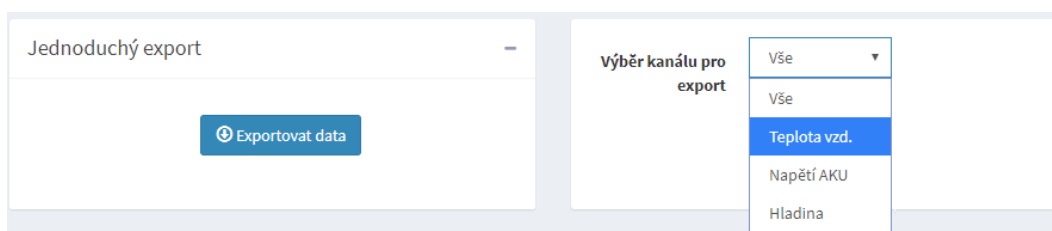
- stanice toků
- datové toky – povolení zobrazení názvů měřicích jednotek
- formát exportovaného souboru
- formát časové značky – povolení exportu dat pouze od data instalace
- povolení zobrazení názvů datových toků

1. Jednoduchý export

- Jednoduchý export zobrazuje pouze tlačítko na export, které vyvolá uložení souboru s daty do počítače, viz obrázek 3.19 vlevo.

2. Volitelný export

- Volitelný export obsahuje navíc nabídku pro výběr datového toku, který chceme exportovat.



Obrázek 3.19: Export dat

3.3.1.6 Ostatní

Tato kapitola pojednává o dalších widgetech, které nepatří do žádné z výše zmíněných kategorií.

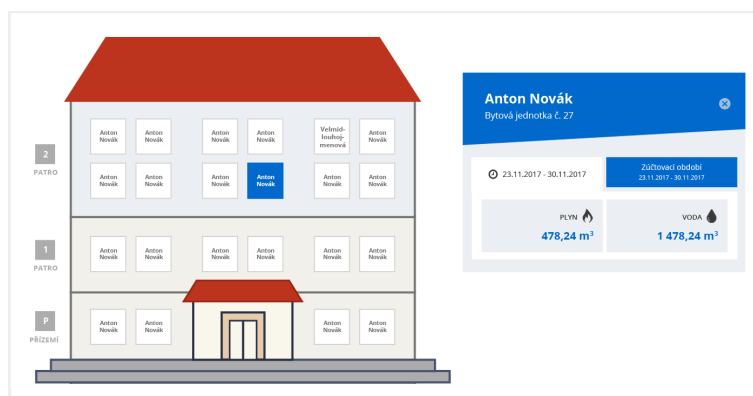
1. Obrázek / Foto

- Tento widget obsahuje především obrázek, zpravidla fotografii měrného místa.
- Mezi parametry uživatelské konfigurace patří:

- název a popis widgetu
 - rozměry obrázku
 - obrázek
- * adresa obrázku
 - * upload obrázku

2. Interaktivní obrázek

- Jedná se o interaktivní obrázek budovy ve formátu SVG, který se dynamicky mění na základě vybraných datových toků navázaných na jednotlivé svg prvky. Widget zajišťuje celkový přehled obvykle nad spotřebou vody nebo energií v budovách.
- Mezi parametry uživatelské konfigurace patří:
 - název a popis widgetu
 - počet podlaží
 - počet bytů na podlaží
 - stanice a datový tok (pro každý byt zvlášť)
 - měřicí jednotky
 - transformace dat
 - číslo uživatelské obrazovky
 - * slouží jako odkaz na obrazovku zobrazující spotřebu na daném měrném místě
 - zobrazení názvu widgetu



Obrázek 3.20: Ostatní widgety - „Interaktivní obrázek“

3. SCADA technologická obrazovka

- Tento widget funguje na podobném principu jako předchozí bod. Widget zobrazuje interaktivní SVG obrázek technického provozu (např. ČOV, vodovodní síť, lázně),

který se dynamicky mění na základě hodnot binárních a analogových hodnot datových toků.

- Součástí widgetu je lišta napojená na existující uživatelské obrazovky, které zobrazuje.
- Mezi parametry uživatelské konfigurace patří:
 - název a popis widgetu
 - seznam aktivních SVG prvků
 - stanice a datový tok (pro každý SVG prvek zvlášť)
 - měřicí jednotky
 - transformace dat
 - čísla uživatelských obrazovek
 - * pro každý SVG prvek zvlášť
 - * slouží pro zobrazení obsahu daných uživatelských obrazovek v liště
 - zobrazení názvu widgetu

3.3.2 Analýza datových toků

Tato kapitola se podrobně zabývá analýzou datových toků v aktuální aplikaci zadavatele.

Každá řídicí jednotka widgetu (viz 3.1) žádá o specifická data samostatně na základě svého nastavení a uživatelské konfigurace. Tento systém zajišťuje určitou variabilitu a flexibilitu. Na druhou stranu může docházet k tomu, že několik widgetů požaduje ze serveru shodný soubor dat, např. graf a tabulka zobrazující jednotlivé hodnoty konkrétního datového za shodné časové období.

Data mohou být již transformována, nebo k jejich transformaci dochází na straně řídicí jednotky, viz kapitola 3.2.1.

- Transformovaná data uložená v databázi
 - V současné době není aplikováno, uvažované do budoucna.
 - Výhody:
 - * Menší nároky na místo
 - * Menší nároky na výpočetní výkon při využívání dat

- Nevýhody:
 - * Složitější algoritmy využití dat (zobrazení, ...)
 - * Ukládání dat náročné na výpočetní výkon
 - * Nebezpečí nepřesných dat (závislé na zvolené transformaci)
 - Za předpokladu zvolení bezztrátové transformace je tato transformace výhodná především u dat, která vykazují dlouhodobě stabilní hodnoty, např. pouze výjimečně dochází k jejich změně. V takovém případě mohou být odstraněny duplicity a zaznamenány pouze změny. Naopak je taková transformace nevýhodná, pokud u hodnot dochází často ke změně.
- Data transformovaná na straně serveru
 - Aplikováno pro časté případy transformace, např. agregace.
 - Výhody:
 - * Menší zatížení sítě
 - Posílání menšího souboru dat (omezeno na potřebná data)
 - * Menší zatížení webového klienta
 - Nevýhody:
 - * Větší zatížení výpočetního výkonu serveru
 - Výhodné při velkém souboru dat a složitých operacích mezi několika datovými toky.
 - Ve všech případech je využita transformace plovoucích oken 20 pro zaslání pouze dat, která se nachází ve vybraném časovém rámci.
 - Data transformovaná v řídicí jednotce
 - Aplikováno pro méně časté případy a případy, kdy je potřeba data upravit na základě kontextu.
 - Příkladem takové transformace je převod jednotek, např. hodnoty hladiny v cm jsou převedeny na m.

- Výhody
 - * Menší zatížení výpočetního výkonu serveru
- Nevýhody
 - * Větší zatížení sítě
 - * Závislé na výpočetním výkonu klienta

3.3.2.1 Analýza dat ze serveru

Na základě své konfigurace a vnitřního nastavení žádá řídicí jednotka o data ze serveru. Jejich analýze je tato kapitola věnována. Prezentovaná data jsou především data získaná ze sensorů a telemetrických stanic a upravená na serveru. Diskutovaná aplikace využívá výhod RESTového přístupu, přijímá data ve formátu JSON.

Jednotlivé typy dat jsou pro snazší orientaci popsány za využití datového slovníku:

Datový slovník Data a datové toky jsou v této práci popsány za pomoci datového slovníku. Datový slovník popisuje data do detailu a dává tím hlubší náhled do celé problematiky dat, jejich struktury, stylu využití apod. Datové slovníky využívají k popisu specifické značky [8]:

... = ...	skládá se z	$AB = A + B$
... + ...	a	adresa = ulice + číslo popisné + město
[... ...]	nebo	ovoce = [jablko hruška]
{...} ⁿ	opakující se obsah	$PIN = \{\text{číslo}\}^4$
(...)	volitelná položka	(poznámka)
@...	klíčová položka	@idStanice
...	komentář	*Komentář pro upřesnění*

Tabulka 3.4: Značky datového slovníku

Pro účely této práce byla nadefinována základní pojmenování:

$D = \text{datový tok}$	$_1 = \text{první/pocateční}$
$T = \text{casova značka *UNIX format*}$	$_n = \text{poslední/koncova}$
$V = \text{hodnota}$	$_s = \text{pocet}$
$E = \text{indikátor chyby}$	$_x = 0\dots n$
$M = \text{metadata}$	$_{id} = \text{identifikační číslo}$

Datové toky (DS)

- Datové toky jsou omezeny časovým rámcem.
- Struktura datových toků a jejich transformací:

– Základní datové toky

- * 1 datový tok
- * všechny hodnoty za dané období

$$D = \{T_x + V_x + E_x\}^n$$

– Transformované datové toky - FILTROVÁNÍ

- * 1 datový tok
- * první známá hodnota

$$D = T_1 + V_1 + E_1$$

- * poslední známá hodnota

$$D = T_n + V_n + E_n$$

– Transformované datové toky - VÝPOČET

- * operace mezi hodnotami se stejnou časovou značkou
- * na vstupu pro 2 datové toky - výsledkem je 1 datový tok
- * datové toky musí být stejného typu
 - analogové - součet / rozdíl, násobení / dělení
 - binární - AND, OR, XOR

- * všechny výsledky za dané období

$$D = T_1 + T_n + V_s + E_s + \{D_x\}^n$$

$$D_x = T + V_r$$

$V_r = \text{vysledek matematicke operace}$

– Transformované datové toky - AGREGACE

- * 1 datový tok
- * agregovaná data za dané období

$$D = T_1 + V_1 + T_n + V_n + T_{min} + V_{min} + T_{max} + V_{max} + V_{avg} + V_{sum} + V_s + E_s$$

$T_{min}/T_{max} = \text{casova znacka min/max hodnoty}$

$V_{min}/V_{max} = \text{min/max hodnota za interval}$

$V_{avg} = \text{prumerna hodnota}$

$V_{sum} = \text{celkova suma}$

– Transformované datové toky - FREKVENCE

- * více datových toků
- * transformovaná data za dané období s mezními hodnotami - frekvence výskytu pro jednotlivé hodnoty

$$D = T_1 + T_n + V_s + E_s + \{D_x\}^n$$

$$D_x = V_f + V_c + V_p$$

$V_f = \text{frekventovana hodnota}$

$V_c = \text{pocet vyskytu}$

$V_p = \text{procentualni zastoupeni}$

Metadata

- Data napojená na ID číslo jednotky, nebo uživatelské obrazovky.
- Každá uživatelská obrazovka zpracovává následující metadata:
 - Metadata o uživatelské obrazovce (dashboard)
 - * obsahují seznam napojených stanic a souvisejících dashboardů, zařazení, strukturu a nastavení dashboardu a v neposlední řadě seznam widgetů a jejich nastavení

$$M = U_{id} + W_s + U_L + U_k + U_{set} + \{W_x\}^n + \{S_x\}^n$$

$$W = widget = W_{id} + W_p + W_{type} + \{W_{set_x}\}^n$$

$$S = stanice = S_{id} + S_{type}$$

$U = dashboard$

$U_L = struktura/rozlozeni\ dashboardu$

$U_k = zarazeni\ dashboardu$

$W_p = umisteni/pozice\ widgetu$

$set = nastaveni$

- Metadata o právech uživatele
 - * obsahují informace, jaká práva má aktuální uživatel
- Metadata o stanici
 - * obsahují vždy id stanice a její konfiguraci, např. frekvence odesílání dat, měřené veličiny,

$$M = S_{id} + \{K_x\}^n$$

$S = stanice$

$K = konfigurace\ stanice$

– Metadata o datovém toku

- * obsahují identifikační číslo a název datového toku, typ dat (analogová/binární), měrné jednotky a případné připojené datové toky

$$M = D_{id} + D_{name} + D_{type} + D_u + \{D_x\}^n$$

$$D = D_{id} + D_{tag}$$

$$D_{name} = \textit{nazev}$$

$$D_{type} = \textit{typ dat}$$

$$D_u = \textit{merne jednotky}$$

$$D_{tag} = \textit>tag$$

– Metadata o měrném místě

- * obsahují vždy technické číslo měrného místa a identifikační číslo stanice, dále pak volitelná data, např. adresu, datum instalace apod.

$$M = S_{id} + MM_{id} + \{I_x\}^n$$

$$S = \textit{stanice}$$

$$M = \textit{merne misto}$$

$$I = \textit{volitelna data}$$

- Pole metadat

– např. metadata ke všem relevantním datovým tokům v jednom souboru

$$P = \{M_x\}^n$$

$$P = \textit{pole}$$

Mapové podklady

- Speciální typ dat využívaný u map.

- Data získaná z URL adresy, ze které se získávají obrázky pro mapové podklady.
- Tato data se zpravidla nemění s časem, případně se mění jednorázově.

$$P = \text{mapovy podklad/obrazek}$$

Následující tabulky (Tab. 3.5 a Tab. 3.6) prezentují data, o které řídicí jednotka server žádá. Z tabulek je zřejmé, že server poskytuje kombinace výše popsaných základních datových toků (DS), jejich transformací a metadat. Zároveň je možné pozorovat, že některé widgety (např. „KPI s odkazem“) server o data nežádají.

Typ widgetu	Data ze serveru
Čárový graf	Základní DS
Čárový graf a transformace	Základní DS Transformované DS - VÝPOČET
Sloupcový graf Spotřeba	Základní DS Transformované DS - FILTROVÁNÍ
Binární graf	Základní DS Metadata o datovém toku
Tab. Základní informace	Metadata o MM
Tab. Hodnoty	Základní DS
Tab. Spotřeba	Základní DS Transformované DS - FILTROVÁNÍ
Tab. Počet výskytů	Metadata skupina stanic Transformované datové toky - FREKVENCE
Tab. Průměr	Základní DS
Tab. Počet změn u binárních DS	Základní DS DS z metadat

Tabulka 3.5: Analýza dat ze serveru - vybrané widgety (1. část)

Typ widgetu	Data ze serveru
KPI s poslední hodnotou	Transformované DS - FILTROVÁNÍ
KPI min/max	Transformované datové toky - AGREGACE
KPI pro události	Transformované datové toky - AGREGACE Metadata o datovém toku
KPI pro události s kontrolou	Transformované datové toky - AGREGACE Metadata o datovém toku
KPI s metadaty	Metadata o měrném místě
KPI s odkazem	
Mapa s více body	Metadata o měrném místě Mapový podklad
Agregovaná data na mapě	Transformované datové toky - AGREGACE Metadata o měrném místě Mapový podklad
Mapa s GPS souřadnicí	Mapový podklad
Jednoduchý export	Základní DS DS z metadat
Interaktivní obrázek	Transformované DS - FILTROVÁNÍ

Tabulka 3.6: Analýza dat ze serveru - vybrané widgety (2. část)

3.3.2.2 Transformace dat na straně klienta

Řídicí jednotka s daty dále manipuluje na základě svého vnitřního nastavení a uživatelské konfigurace. Tato kapitola pojednává o způsobech úpravy dat přijatých ze serveru, a to i v kombinaci s daty statickými. K finální transformaci dat před jejich zobrazením dochází v řídicí jednotce widgetu, tj. na straně klienta.

Statická data Data získaná z konfigurace a nastavení widgetu (viz str. 3.3), např. „Název widgetu“, která se zpravidla nemění s časem.

Kombinace V této práci je kombinace definována jako zobrazení více typů dat v jednom widgetu.

Mezi způsoby úpravy dat na straně klienta patří:

- Operace v rámci 1 datového toku
 - Jednoduché matematické operace
 - * výpočet přírůstku na časový interval - výsledkem je např. týdenní přehled s denní sumou
 - * suma za celý časový rámeček

$$Z = \{D\}^n$$

$$D = O + T_y + V_y + O + T_z + V_z$$

D = datový tok

Z = zobrazená hodnota

T = časová značka

V = hodnota

y/z = začátek/konec intervalu

$O = [+|-]$ = matematická operace

- Kombinace a operace se statickými daty

- Všechna data ze serveru jsou kombinována a zobrazena společně se statickými daty, např. „měrné veličiny“, „limitní hodnoty“, „nápověda“ apod.

- Datové toky a statická data

- * Jednoduché matematické operace - sčítání, odečítání, násobení a dělení.

$$Z = \{V_x + O + V\}^n$$

$V_x = \text{hodnota datoveho toku}$

$V = \text{staticka hodnota}$

$O = [+ | - | * | /] = \text{matematicka operace}$

$Z = \text{zobrazena hodnota}$

- * Složitější matematické operace dle vzorce, např. průměr nebo procentuální zastoupení, tzn. různé kombinace jednoduchých matematických operací.

- Metadata a statická data

- * V rámci metadat dochází především k jejich filtraci a řazení, např. přidání znaků mezi jednotlivými hodnotami metadat, výběr konkrétních metadat apod.

- * Nejsou podporovány matematické operace mezi statickými daty a metadaty.

- Kombinace datových toků a metadat

- Tato kombinace je využita často u binárních dat, kde jsou z metadat získány např. „názvy datových toků“.
- Nejsou podporovány matematické operace mezi datovými toky a metadaty.

- Kombinace a operace mezi datovými toky

- Kombinace
 - * Zobrazení více datových toků v jednom widgetu.
- Matematické operace

- * Aplikace podporuje jednoduché matematické operace mezi datovými toky, tzn. mezi hodnotami se společným znakem, např. časovou značkou nebo naměřenou hodnotou.

$$Z = \{(V_x + O + V_x) + O + V\}^n$$

$V_x = \text{hodnota datoveho toku}$

$V = \text{staticka hodnota}$

$O = [+ | - | * | /] = \text{matematicka operace}$

$Z = \text{zobrazena hodnota}$

- * Zpravidla dochází k matematické operaci mezi daty transformovanými na serveru.

- Např. při získání transformovaných datových toků „Frekvence“ získáme pro každý datový tok frekvenci jednotlivých hodnot za dané období. Operace v řídicí jednotce pak vypočítá průměrnou frekvenci hodnot přes všechny datové toky. Zároveň dochází k odstranění (filtrování) dat s nulovou frekvencí.

- Kombinace s mapovými podklady

- V rámci mapových podkladů dochází ke kombinaci se statickými daty, případně metadaty a datovými toky.

Následující tabulky (Tab. 3.7 a Tab. 3.8) demonstrují způsoby a rozmanitost využití dat u vybraných widgetů:

Typ widgetu	Kombinace a operace v řídicí jednotce
Čárový graf	Kombinace mezi DS Kombinace a operace DS se statickými daty
Čárový graf a transformace	Kombinace mezi DS Kombinace a operace DS se statickými daty
Sloupcový graf Spotřeba	Operace v rámci 1 DS Kombinace mezi DS Kombinace a operace DS se statickými daty
Binární graf	Kombinace DS, metadat a statických dat
Základní informace	Kombinace metadat a statických dat
Tabulka Hodnoty	Kombinace a operace DS se statickými daty
Tabulka Spotřeba	Operace v rámci 1 DS Kombinace mezi DS Kombinace a operace DS se statickými daty
Tabulka Počet výskytů	Kombinace a operace mezi DS Kombinace a operace DS se statickými daty
Tabulka Průměr	Operace v rámci 1 DS Kombinace a operace DS se statickými daty
Tabulka Počet změn u binárních DS	Operace v rámci 1 DS Kombinace mezi DS Kombinace DS se statickými daty
KPI s poslední hodnotou	Kombinace a operace DS se statickými daty
KPI s min/max	Kombinace a operace DS se statickými daty
KPI pro události	Kombinace mezi DS se statickými daty a metadaty
KPI pro události s kontrolou	Kombinace mezi DS se statickými daty a metadaty
KPI s metadaty	Kombinace metadat a statických dat
KPI s odkazem	Statická data

Tabulka 3.7: Analýza dat na straně klienta - vybrané widgety (1. část)

Typ widgetu	Kombinace a operace v řídicí jednotce
Mapa s více body	Kombinace metadat, statických dat a mapových podkladů
Agregovaná data na mapě	Kombinace mezi DS se statickými daty, metadaty a mapovými podklady
Mapa s GPS souřadnicí	Kombinace statických dat a mapových podkladů
Jednoduchý export	Kombinace mezi DS se statickými daty a metadaty
Interaktivní obrázek	Kombinace mezi DS se statickými daty a metadaty

Tabulka 3.8: Analýza dat na straně klienta - vybrané widgety (2. část)

Z výše uvedeného je zřejmé, že na straně serveru:

- dochází k základní transformaci datových toků za účelem snížení objemu dat:
 - Metoda plovoucích oken
 - * V případě, že widget zobrazuje data z datového toku, je tok vždy omezen časovým rámcem.
- jsou prováděny základní matematické operace mezi datovými toky.

Z výše uvedeného je zřejmé, že na straně klienta:

- dochází především ke kombinaci datových typů, jednoduchým matematickým operacím a zobrazení výsledků.

Současná aplikace nepodporuje, nebo podporuje omezeně:

- validaci a kontrolou dat a datových typů
- bližší využití metadat (podobně jako statická data), např.
 - matematické operace s metadaty - transformace datových toků na základě hodnot v metadatech nebo transformace hodnot z metadat na základě statických hodnot
 - automatizace pojmenování widgetů na základě názvů v metadatech
 - získání limitních hodnot z metadat

3.4 Analýza ostatních systémů

Kapitola se věnuje analýze dalších systémů pro zobrazování dat, především pak analýze datových toků a srovnání jejich využití oproti současné aplikaci CloudFM (viz kapitola 2.2.2). Jedním z analyzovaných systémů je původní systém zadavatele „Web prohlížeč naměřených dat“, dále jsou analýze podrobeny zástupci relevantních widgetů a celých systémů pro správu a vizualizaci dat.

3.4.1 Web prohlížeč naměřených dat

Jedná se o systém, který zadavatel stále používá (viz kapitola 2.2.1). Tento systém byl prozkoumán již v práci Ing. Jana Píši [1], na jejímž základě byla vytvořena aktuální aplikace CloudFM. V současné době zadavatel postupně přechází na aplikaci CloudFM.

Cílem této kapitoly je vydefinovat prvky, které se v původní aplikaci nacházely a byly využívány, a zároveň chybí v aplikaci CloudFM.

Webové komponenty

Pro každou stanici je automaticky vygenerována uživatelská obrazovka, existuje ale i možnost speciální uživatelské obrazovky zobrazující data z více stanic.

Každá uživatelská obrazovka obsahuje následující informace:

- Název stanice
- ID stanice
- Ikony - indikace alarmů
 - Ikona se zobrazí, pokud byla spuštěna podmínka pro alarm, např. „Výpadek napájení“ nebo „Data starší než interval odesílání“.
- Datum posledních dat
- Předem definované prvky

- Zobrazeny na všech uživatelských obrazovkách.
- Volitelné prvky
 - Pevný počet
 - Požná uživatelská konfigurace

Uživatelské obrazovky obsahují předem definované webové komponenty a omezený počet (maximálně 8) volitelných webových komponent, které lze většinou dále konfigurovat. Všechny webové komponenty jsou statické, např. grafy jsou zobrazeny jako obrázky.

Na uživatelské obrazovce se dále nachází následující odkazy:

- Operace s daty
 - Umožňuje smazat data naměřená v zadaném časovém intervalu.
 - Umožňuje pevné zadání data posledních dat.
- Sledovací agent
 - Slouží k nastavení akcí, které se spustí, pokud dojde k chybovému stavu.
 - Obsahuje možnost výběru chybového stavu a vyplnění adresy emailu, kam bude zaslána informace o události.
- Konfigurace parametrů stanice
 - Obsahuje seznam všech konfigurací stanice, které lze stáhnout do počítače a prohlédnout / editovat v příslušném počítačovém programu.

Předem definované webové komponenty:

1. Tabulka

- Systém využívá jednotný systém tabulek pro zobrazení dat ze všech jednotek.

2. Měřené veličiny

- Agregovaná data z hodnot jednotlivým měřených veličin.

3. Binární data

- Tabulka se všemi získanými binárními stavy.

4. Statistika

- Tabulka pro každou měřenou veličinu zvlášť.
- Obsahuje minimální, maximální a průměrnou hodnotu.

5. Export

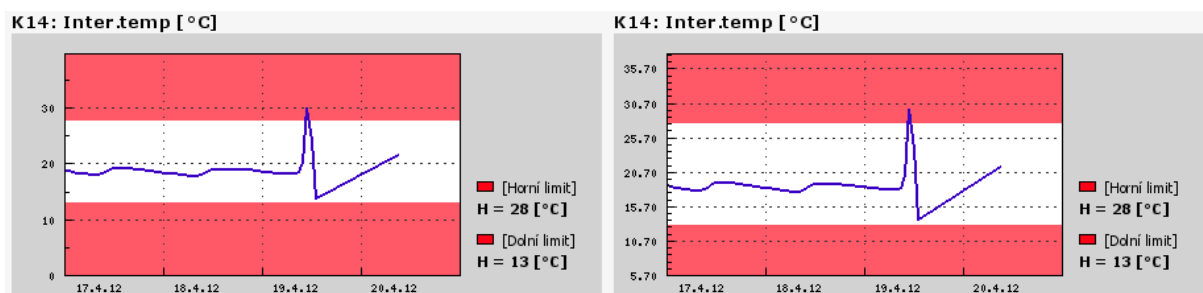
- Aplikace podporuje export dat do souborů DTA, CSV a TXT.

Volitelné webové komponenty:

1. Graf

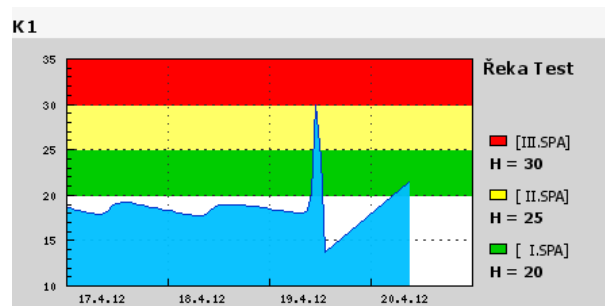
- Graf min ... max
 - Graf zobrazuje naměřené hodnoty a zvýrazňuje oblasti mimo nastavené limity, viz obrázek 3.21 napravo.
 - Uživatelská konfigurace obsahuje:
 - * Zobrazovaný kanál
 - Výběr jednoho analogového kanálu z nabídky.
 - * Typ grafu
 - čárový, bodový, sloupcový, vyplněný
 - * Rozsah grafu
 - pevně definovaný časový rámeček
 - 4 nebo 7 dní
 - * Popiska
 - * Horní / Dolní limit
 - * Korekce - nastavení korekce dat
 - podmínka (hodnota $<$, $>$, s chybou)
 - kanál

- nová hodnota (0, nebo poslední platná)
- VYHODNOCENÍ:
 - (a) Odpovídající komponenta v současné aplikaci CloudFM: **ANO**
 - (b) Prvek, který není v současné aplikaci zakomponován: **Korekce**



Obrázek 3.21: Volitelné webové komponenty „Graf 0 ... max“ a „Graf min ... max“

- Graf 0 ... max
 - Speciální případ grafu „Graf min ... max“. Osa y začíná od 0, viz obrázek 3.21.
 - Uživatelská konfigurace odpovídá konfiguraci „Grafu min ... max“.
- Graf + meze
 - Speciální případ grafu „Graf min ... max“.
 - Uživatelská konfigurace odpovídá konfiguraci „Grafu min ... max“, až na možnost nastavení mezí, tzn. rozsahu osy y:
 - * Horní rozsah
 - * Dolní rozsah
- Limnigrafická stanice
 - Graf určený pro zobrazení výšky hladiny, např. řek.
 - Graf odpovídá grafu „Graf + meze“ bez možnosti nastavení limitů. Navíc obsahuje 3 limity pro stupně povodňové aktivity.
 - Uživatelská konfigurace navíc obsahuje:
 - * 1. - 3. SPA



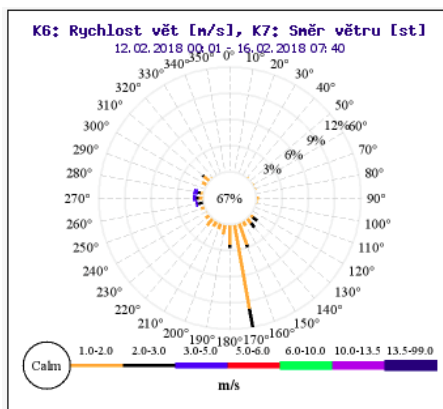
Obrázek 3.22: Volitelná webová komponenta „Limnigrafická stanice“

- Vícekanálový graf
 - Zobrazuje až 4 měřené veličiny v jednom grafu.
 - Uživatel může zvolit ze 4 barev pro zobrazení.
 - * 4 kolonky pro výběr měřených veličin
 - * Typ grafu
 - * Rozsah grafu
 - * Popiska
 - * Mezní (popř. limitní) hodnoty grafu

- Větrná růžice
 - Speciální graf pro zobrazení rychlosti a směru větru.
 - * Rozsah grafu
 - * Popiska
 - * Data pro rychlost větru
 - * Data pro směr větru

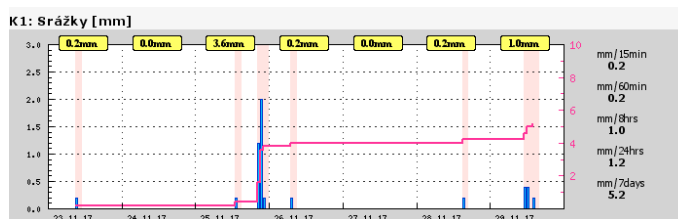
- Stav vrtu
 - Speciální čárový graf zobrazující stav vrtu.
 - Obsahuje stejnou konfiguraci jako „Graf min ... max“.

- Binární stavy
 - Graf pro zobrazení binárních kanálů.
 - Uživatelská konfigurace obsahuje:
 - * Rozsah grafu
 - * Popis grafu



Obrázek 3.23: Volitelná webová komponenta „Větrná růžice“

- * Binární kanál (výběr až 8 binárních kanálů)
- Srážkoměr
 - Speciální graf pro zobrazení srážek.
 - * Zobrazovaný kanál
 - * Rozsah grafu
 - * Sčítat data po
 - * Popiska
 - * Horní/Dolní rozsah
 - * Zobrazit sumu
 - * Rozlišit déšť



Obrázek 3.24: Volitelná webová komponenta „Srážkoměr“

2. Obrázek

- Bez uživatelské konfigurace v této nabídce.
- Zvolením této nabídky se zobrazí obrázek, který patří k dané stanici.
- Obrázek lze nahrát v nastavení stanice.

3.4.1.1 Srovnání

Cílem analýzy původní aplikace je vytvořit podklady pro návrh prvků a vlastností, které byly implementovány v původní aplikaci a v současné aplikaci chybí. Z analýzy vyplývá následující srovnání:

Současná aplikace CloudFM zobrazuje data dynamicky s flexibilním rozložením prvků (widgetů). Uživatelská obrazovka zpravidla obsahuje volitelné informace.

Původní aplikace používá uniformní systém pro statické zobrazení dat. Všechna naměřená data jsou zobrazena v tabulkách, volitelně v podobě grafů. Obrazovka může zobrazovat obrázky.

Na rozdíl od současné aplikace, původní aplikace

- vždy obsahuje implicitně datum posledních dat a případné varovné či chybové stavy.
- umožňuje ruční změnu data posledního odeslání dat
- umožňuje mazat naměřená data
- umožňuje korekci dat (nahrazení konkrétní hodnoty, či intervalu poslední platnou, nebo nulovou hodnotou)
- v nabídce webových komponent disponuje komponentou „Srážkoměr“, jež není v aktuální aplikaci implementována
- zobrazuje statistické informace ke všem měřeným veličinám
- automaticky zobrazuje hodnoty všech měřených hodnot
- umožňuje export do více datových formátů (ale neumožňuje ovlivnění obsahu souboru a jeho vzhled)

3.4.2 Odoo

Odoo je komplexní ERP a CRM systém, který mimo jiné nabízí zpracování a analýzu dat a je vybaven uživatelskými obrazovkami pro zobrazení dat. [10]

Použité technologie:

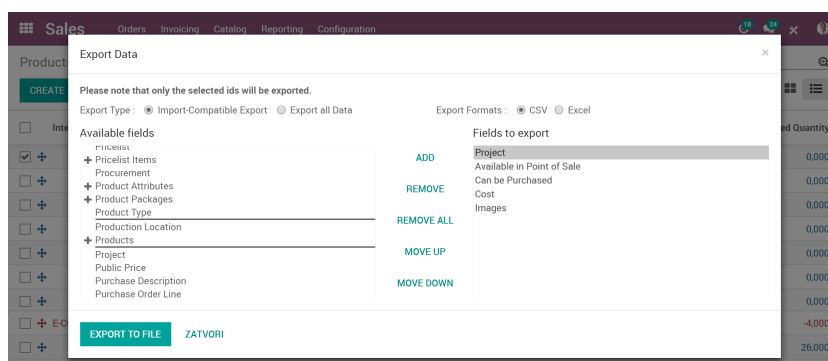
- Python, JavaScript, XML, REST API

V rámci této kapitoly je provedena analýza vybraných widgetů pro zobrazení dat, které nabízí systém Odoo na adrese <https://demo3.odoo.com>. Kapitola popisuje systém widgetů, jejich uživatelskou konfiguraci a datové toky. Na základě analýzy je vytvořeno srovnání oproti současné aplikaci CloudFM.

Základní typy widgetů

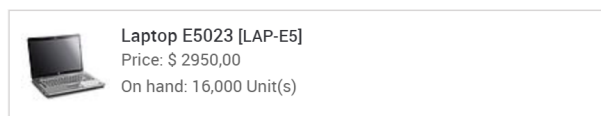
Systém Odoo rozlišuje widgety dle použité webové komponenty:

- Seznam
 - Jedná se o tabulku se vzhledem seznamu, která vypisuje metadata.
 - Jednotlivé položky lze řadit ručně (táhnutím), nebo abecedně podle sloupců.
 - Vybrané položky lze exportovat do XLS a CSV nebo tisknout do PDF (náhled v XML)
 - * Export obsahuje nabídku, kde lze vybrat exportované položky a nastavit jejich řazení, viz obrázek 3.25.



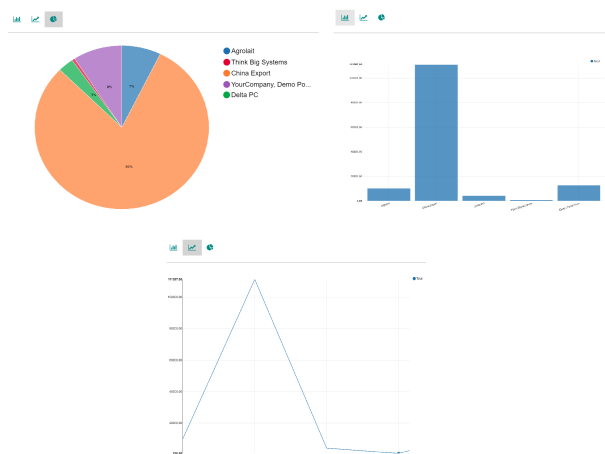
Obrázek 3.25: Odoo export

- Kanban (Cedule)
 - Tato komponenta zobrazuje klíčové položky firmy.
 - V základu odpovídá prvku „KPI“ v aplikaci CloudFM, viz str. 35.
 - Zobrazuje především data z metadat, včetně obrázků.



Obrázek 3.26: Odoo „Kanban“

- Kalendář
 - Zobrazuje metadata v oknech kalendáře.
 - Filtruje data podle jejich tagů (značek).
- Tabulka
 - Zobrazuje datové toky (základní i transformované) v kombinaci s metadaty.
 - Lze exportovat do XLS.
 - * Widget využívá k exportu již zasláná data.
 - * Exportovaná tabulka má stejný vzhled jako tabulka zobrazená online (včetně barev).
 - Umožňuje překlopení os x a y na straně klienta.
- Graf
 - Systém nabízí pro zobrazení dat webovou komponentu „Graf“.
 - Při samotném zobrazení komponenty lze měnit její typ (nezávisle na typu dat / datových toků).
 - * Widget při změně nežádá o nová data.
 - * Grafy na obrázku 3.27 zobrazují stejná data.



Obrázek 3.27: Odoo „Graf“

Každý widget žádá o data server. Využívané typy dat ze serveru:

- Datové toky (DS)
 - Obvykle transformované DS (viz kap. 3.3.2.1)
- Metadata
 - Metadata jsou závislá na typu zobrazované položky.
 - Jsou zobrazována ve všech typech použitých webových komponent.
- Kombinace DS a metadat
 - Obvykle pole metadat, kdy součástí metadat je transformovaný datový tok, např. poslední nebo agragovaná hodnota

3.4.2.1 Srovnání

CloudFM - postup při vytváření widgetu:

1. Typ widgetu / webové komponenty
2. Další nabídka
 - Datový tok a transformace nad ním, statická data
 - Časový rámeček (možná změna při zobrazení)

Odoo využívá (podobně jako CloudFM) REST API pro předávání zobrazovaných dat. Podobně jako CloudFM zobrazuje data dynamicky. Postup při vytváření widgetu:

1. Datový tok a transformace nad ním
2. Statická data a metadata
 - Časový rámeček (možná změna při zobrazení)
 - Typ webové komponenty (možná změna při zobrazení)

Na rozdíl od současné aplikace, Odoo

- posílá transformované datové toky a metadata v jednom datovém souboru
- umožňuje mazat data
- zobrazuje statistické informace (s možností exportu)
- umožňuje export do formátu XLS se zachováním formátu tabulky
- umožňuje tisk do PDF s náhledem XML
- umožňuje změnit typ webové komponenty při zobrazení dat
- umožňuje zobrazovat všechny datové typy ve všech webových komponentách
 - zobrazuje obrázky v KPI, metadata v grafech apod.
- kombinuje metadata a datové toky do jednoho datového souboru

3.4.3 UNMS

Jedná se o systém společnosti „Ubiquity Networks“ určený pro centralizované monitorování a řízení aktivních prvků v počítačových sítích v reálném čase. [11]

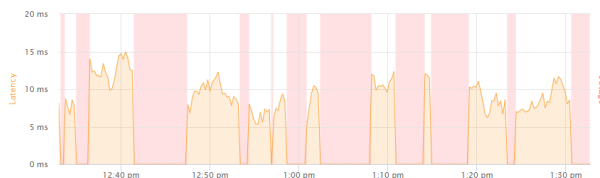
Aplikace UNMS zobrazuje data v reálném čase, a proto jsou zobrazovaná data neustále aktualizovaná. Využívané typy dat ze serveru:

- Metadata
 - Metadata jsou závislá na typu zobrazované položky.
 - Jsou zobrazována ve všech typech použitých webových komponent.
- Kombinace DS a metadat
 - Obvykle pole metadat, kdy součástí metadat je transformovaný datový tok, např. poslední nebo agragovaná hodnota

Systém UNMS má úzké zaměření na uniformní data z aktivních prvků. Aplikace má systém s unifikovanou strukturu uživatelských obrazovek a jejich widgetů. Odpovídá tomu i rozložení widgetů v rámci uživatelských obrazovek. Systém nabízí několik typů uživatelských obrazovek, které jsou vytvořeny automaticky, např.:

- Přehled aktivních prvků a výpadků
 - Obsahuje informace z metadat všech aktivních prvků ve skupině.
 - Lze filtrovat podle hodnot v metadatech.
 - * Tabulka
 - * Mapa
- Přehled statistik
 - Systém podporuje zejména čárové grafy pro zobrazení trendu.
 - Lze zobrazit několik trendů v jednom grafu.
 - Widget podporuje nastavení limitů a mezních hodnot.

- Lze nastavit zvýraznění období, kdy chybí data nebo jsou nulová, viz obrázek 3.28.
- Umožňuje zobrazit specifický časový rámec, např. hodina, den, měsíc, ..



Obrázek 3.28: Ubiqity „Graf“

- Log událostí
 - Tabulka s logem událostí (podobně jako aplikace CloudFM).
 - Využívá grafy pro zobrazení procentuálních hodnot.
 - Mění zabarvení textů v závislosti na jejich obsahu.
- Editace
 - Aplikace umožňuje přidávat nové aktivní prvky a upravovat jejich informace

3.4.3.1 Srovnání

Ačkoliv má aplikace UNMS jiné využití, nabízí přístupy, které by bylo možné aplikovat. Na rozdíl od CloudFM, umožňuje

- automatické vytvoření widgetů
- automatické propojení všech uživatelských obrazovek
- zobrazovat všechny datové typy ve všech webových komponentách, např.
 - zobrazuje metadata v grafech
 - zobrazuje grafy v tabulkách
- vytvářet nová, editovat a mazat data

- měnit typ webové komponenty při zobrazení dat
- kombinuje metadata a datové toky do jednoho datového souboru

Na druhou stranu, aplikace UNMS neumožňuje např.:

- přidání widgetu do přednastavené uživatelské obrazovky
- změnu vzhledu widgetu

3.4.4 Sensorthings

Sensorthings je systém společnosti SensorUp Inc. zaměřený na správu a zobrazení dat z IoT sensorů a jejich propojení s dalšími zařízeními a aplikacemi. [12]

Použité technologie:

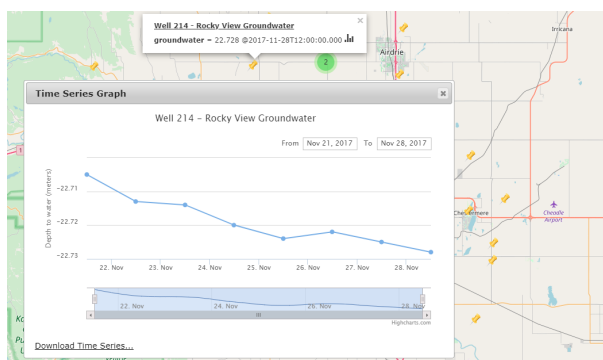
- Shell, Ruby, Python, JSON, JavaScript, JAVA, C#, REST API

Tato kapitola se zabývá především způsobem využití dat a vybranými widgety:

Interaktivní mapa

- Widget žádá server o omezený soubor dat, který bude zobrazen na mapě. Při najetí myši na konkrétní bod widget požádá o poslední naměřenou hodnotu z datového toku, viz <http://rockyview.sensorup.com/html/browse.html>.
- Data pro zobrazení mapy
 - Mapové podklady
 - Kombinace DS a metadat (pole metadat)
 - * Prvek v poli = Bod na mapě
 - Název
 - GPS souřadnice
 - Poslední známá hodnota - obdoba transformovaného datového toku - Filtrace.

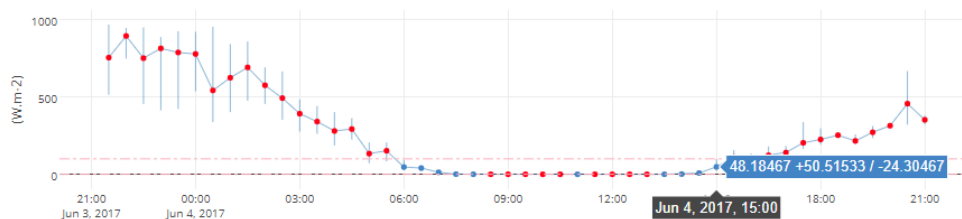
- Data pro konkrétní bod
 - KPI
 - * Výpis poslední hodnoty, obdoba transformovaného datového toku - Filtrace.
 - Čárový graf - zobrazení trendu
 - * Obdoba základního datového toku



Obrázek 3.29: SensorUp „Mapa“

Graf

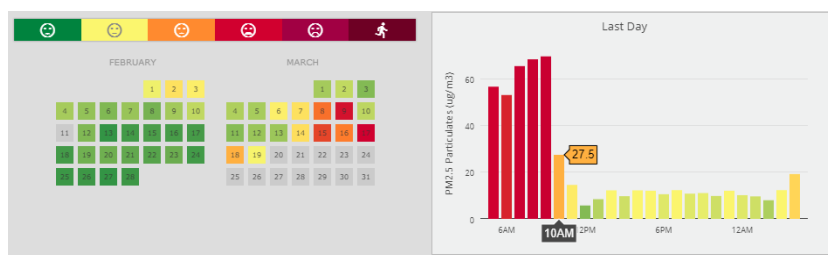
- Systém využívá k zobrazení dat širokou škálu grafů.
- Grafy využívají obdoby základních datových toků a pole metadat s transformovanými datovými toky (viz Filtrace / Agregace).
- Na obrázku 3.30 se nachází graf, který zvýrazňuje hodnoty mimo nastavené limity a zároveň zobrazuje agregované hodnoty v popisku (+ / -) a prostřednictvím horizontálních linek v grafu .



Obrázek 3.30: SensorUp „Grafy“

Kalendář

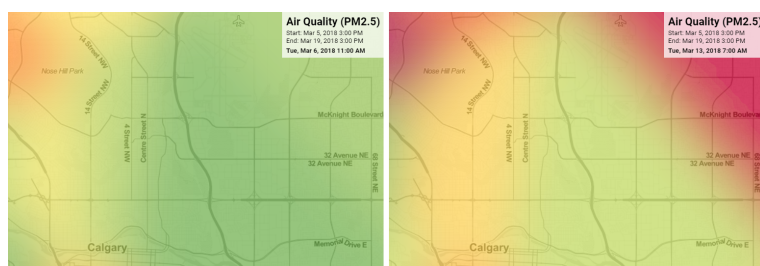
- Interaktivní kalendář s barevným označením jednotlivých prvků podle naměřených hodnot, viz obrázek 3.31.
- Data
 - Metadata k měrnému místu
 - Obdoba „Základního DS“



Obrázek 3.31: SensorUp „Kalendář“

Animace

- Grafické zobrazení trendů ze skupiny senzorů, konkrétně se jedná o změnu barev na mapě v průběhu času přes SVG filtr, např. znečištění ovzduší (viz <http://calgary-air.sensorup.com/animation/>)
- Data
 - Mapové podklady
 - Metadata o celé oblasti
 - Metadata o skupinách na mapě, např. městské oblasti
 - Rozsáhlá metadata pro celou oblast obsahující pole metadat ke každé stanici včetně filtrovaných hodnot z DS



Obrázek 3.32: SensorUp „Animace“

3.4.4.1 Srovnání

Systém „Sensorthings“ se oproti aplikaci CloudFM zaměřený na zobrazení dat z více senzorů v rámci jedné uživatelské obrazovky. Aplikace CloudFM se zaměřuje na zobrazení podrobných informací o jedné stanici / měřném místě. Od toho se odráží i použití dat.

Na rozdíl od CloudFM, systém „Sensorthings“:

- posílá v DS standardní datový formát
- používá kombinace DS a metadat
- u velkého množství dat používá více souborů s metadaty, které na sebe odkazují
- zobrazuje v jednom grafu trendy DS zároveň s agregovanými hodnotami
- ověřuje práva přístupu k datům před tím, než o data zažádá
 - posílá žádost o data metodou „OPTIONS“ (ověření) a až poté „GET“
- nabízí rozsáhlou škálu webových komponent pro zobrazení naměřených hodnot, např.:
 - Animace
 - Kalendář
 - Budík

Kapitola 4

Návrh řešení

Tato kapitola je věnována návrhu systému pro správu celého životního cyklu widgetu, od jeho konfigurace, předzpracování dat, až po vykreslení těchto dat uživateli.

V rámci kapitoly je navržena kategorizace jednotlivých widgetů, nastavení hranice pro parametry, jejichž hodnoty budou určeny již ve vnitřním nastavení řídicí jednotky widgetu nebo v nabídce uživatelské konfigurace. Dále se zabývá daty, která widgety zobrazují, a navrhuje jejich transformace na straně serveru a klienta.

Závisle na návrhu celého systému jsou pak navrženy vybrané widgety.

Následující pasáž o rozsahu 7 listů obsahuje utajované skutečnosti a je obsažena pouze v archivovaném originále diplomové práce uložené na Přírodovědecké fakultě JU. Neveřejná část práce se nachází v příloze ?? a ??.

Kapitola 5

Realizace

Kapitola „Realizace“ obsahuje popis implementace systému pro správu životního cyklu widgetu a několika konkrétních widgetů pro demonstraci funkčnosti návrhu.

Následující pasáž o rozsahu 3 listů obsahuje utajované skutečnosti a je obsažena pouze v archivovaném originále diplomové práce uložené na Přírodovědecké fakultě JU. Neveřejná část práce se nachází v příloze ??.

Kapitola 6

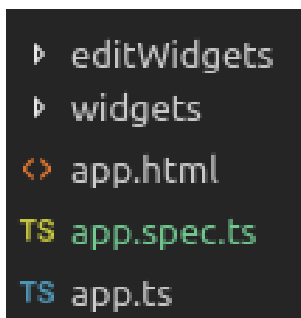
Testování

Tato kapitola se věnuje testování implementovaného systému.

6.1 Package-by-feature

Vyvíjená aplikace dodržuje adresářovou strukturu „package-by-feature“. Tato struktura mimo jiné usnadňuje testování jednotlivých tříd a celků, neboť soubory s jednotnou funkcionalitou jsou sdruženy do společných adresářů, a to včetně testů. Při objevení chyby v testu je snazší se zorientovat, ve kterém souboru k chybě došlo.

Adresářová struktura vychází z navržené hierarchie tříd. Na nejvyšší úrovni dochází k separaci na samotné widgety (*widgets*) a jejich editaci (*editWidgets*). Z následujícího obrázku 6.1 je zřejmé, že se testy (viz soubor s příponou *spec.ts*) nacházejí u testovaných objektů.



```
▶ editWidgets
▶ widgets
<> app.html
TS app.spec.ts
TS app.ts
```

Obrázek 6.1: Adresářová struktura „Package-by-feature“

6.2 Validace kódu

K vývoji aplikace byl použit jazyk TypeScript, nadstavba jazyka JavaScript. Výhodou zvoleného jazyka je, že TypeScript navíc obsahuje statické datové typy, třídy a dědičnost, generické datové typy a umožňuje definovat rozhraní nebo refaktorovat kód. Při psaní v TypeScriptu podléhá kód typové kontrole a základní validaci. [15]

Aplikace byla dále validována pomocí nástroje TSLint, který po celou dobu vývoje kontroloval syntaxi. Zároveň bylo kontrolováno použití správných typů, nebo řazení dle přístupových modifikátorů (například `private`, `protected`, `public`) a dle úrovní přístupnosti pro zachování konzistentního kódu. Byla využita verze, která je přizpůsobená standardům vývoje u zadavatele.[18]

Nástroj TSLint zamezil vzniku chyb, které by bylo zpětně složité odhalit. Zároveň bylo docíleno vzniku přehledného kódu se stejným formátováním.

Dále byly při implementaci dodržovány například konvence pro pojmenování všech objektů a metod, a to za cílem vytvoření tzv. „Self-documenting code“, aby byla funkčnost jednotlivých částí zřejmá z jejich názvu [17].

6.3 Unit testy

V rámci práce byly dále provedeny unit testy, které sloužily k ověření funkčnosti jednotlivých částí aplikace. K unit testům byl použit framework Jest[19].

Jest

Jedná se o nástroj, který je kompatibilní s frameworkem Aurelia.js, ve kterém byla aplikace implementována. Jest je přizpůsoben pro automatické testování v tomto frameworku, a proto pro jeho použití není nutné komplexní nastavování. Další výhodou je, že Jest podporuje Typescript a vypisuje do konzole nejen výsledky testů, ale i výpisy z konzole. Framework se schopný automaticky dohledat testovací soubory na základě jejich umístění, tj. složka `tests` nebo na základě názvu souboru ve formátu `*.spec.ts` nebo `*.test.ts`, což bylo využito

při vývoji, viz (6.1). Syntaxe testů je uživatelsky přívětivá, neboť připomíná běžný anglický text.[19]

Byly provedeny unit testy pro ověření správného nadefinování metod a odkazů. Dále byl simulován životní cyklus metod, a to provedením sekvencí akcí a porovnáním očekávaného a výsledného stavu.

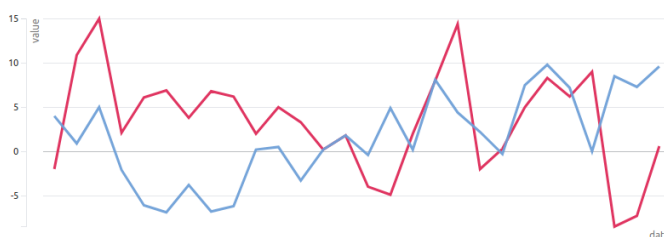
V následující ukázce je test, který ověřuje, že se v daném souboru datových toků vyskytuje analogový datový tok:

```
const expected: any = [{
  id: 'data',
  label: 'analog stream',
  tags: new Set(['analog'])
}];

it('matches', () => {
  expect(service.getAllStreams()).toEqual(expect.arrayContaining(expected));
});
```

6.3.1 Řešené problémy

Při implementaci grafů bylo řešeno několik problémů spojených se specifikací knihovny. Například se neprojevovalo nastavení pro zobrazení hodnot na ose x, viz obrázek 6.2. Po aplikování testů bylo zjištěno, že se jedná o problém css stylu, který byl předefinován.



Obrázek 6.2: Čárový graf bez osy x

Dále se grafy opětovně vykreslovaly po provedení editace, a to pod původní graf. Díky validaci a testům se podařilo odhalit, že nedocházelo k překreslení grafu, nýbrž vždy k vykreslení nového bez odstranění původního zobrazení.

6.4 Ostatní testy

Aplikace byla testována na zobrazení a responzivitu v následujících webových prohlížečích:

- Google Chrome, Version 65.0.3325.181 (Official Build) (64-bit)
- Firefox, Version 59.0.2 (64-bit)

Kapitola 7

Náměty na rozšíření

V rámci možného pokračování práce bylo navrženo:

- Odstínění frameworku
 - Současná implementace systému je úzce spojená s frameworkem Aurelia, ve kterém je implementována. Pro případ, že by se v budoucnu uvažovalo o využití odlišného frameworku, by bylo vhodné framework co nejvíce odstínit a zdokumentovat tyto závislosti.
- Komplexní operace s datovými toky
 - Širší využití operací mezi datovými toky a metadaty a jejich kombinace.
 - Čištění a korekce dat, optimalizace datových toků a časové osy.
 - Statistické operace, simulace A/D převodníku apod.
- Rozšíření funkcionality
 - V rámci implementace grafů by bylo vhodné zvážit využití knihovny pro dynamické přidávání dat do grafu bez jeho opětovného vytváření.
 - Užší využití metadat pro automatické doplnění dat do widgetů zobrazujících datové toky.
 - Sofistikovaná nabídka pro editaci widgetů, například automatický převod jednotek.

- Rozšíření nabídky widgetů
 - Tabulka se statistickými údaji
 - Srážkoměr
 - Kombinovaný graf - kombinace binárního a analogového grafu pro závislé sestavy, například monitorování dmychadla a nádrže, kde jsou sledovány hodnoty rozpuštěného kyslíku.
 - Volitelný export pro metadata a datové toky
 - Interaktivní mapy (s odkazem na jiný dashboard / webovou stránku, s metadaty nebo s grafy a tabulkami k jednotlivým bodům).
 - Standardizované interaktivní obrázky pro různé aplikace, například dálkový odečet vodoměrů ve vysokopodlažních budovách.
- Částečné převedení funkcionality na dashboardy
 - Nastavení barevného schématu pro celý dashboard.
 - Nastavení data, od / do kterého budou viditelné naměřené hodnoty, například datum instalace.

Kapitola 8

Závěr

Předložená diplomová práce se věnuje systému pro tvorbu interaktivních prvků (widgetů), které jsou součástí uživatelských obrazovek (dashboardů) společnosti FIEDLER AMS.

Nejprve byly analyzovány widgety v systémech zadavatele, které byly srovnány s widgety v konkurenčních systémech pro tvorbu dashboardů a zobrazování dat. Práce se dále zaměřila na možnosti konfigurace widgetů a jejich možností. Součástí kapitoly byla také analýza datových toků, jejich struktury a transformací. V kapitole 3 byl splněn první, druhý a třetí cíl práce stanovený v části 2.2.3.

V návaznosti na analýzu byly vyhodnoceny nedostatky současného systému a stanoveny následující cíle pro návrh systému pro správu životního cyklu widgetu.

Důležité body návrhu:

- Kategorizace widgetů
- Oddělené vykreslení widgetů od jejich editace
- Jednotná správa dat a datových toků
 - Unifikovaný datový formát
 - Filtrování relevantních dat
- Adaptér pro vykreslení widgetů

Následně byl implementován framework pro správu celého životního cyklu widgetu. Součástí frameworku jsou jednotlivé služby, které zajišťují úpravy widgetu, správu jeho metadat a datových toků i jeho finální vykreslení. Tímto byl splněn čtvrtý a pátý cíl práce stanovený v části 2.2.3.

Během implementace docházelo k průběžné validaci a testování tvořeného kódu pomocí unit testů. Tímto byl splněn i poslední (šestý) cíl.

Přínosy práce

Zadavatel volně přechází z původní aplikace pro zobrazení naměřených dat na novou aplikaci CloudFM. Porovnání těchto aplikací a specifikace potřebných vlastností, kterými nová aplikace nedisponuje, patří k nezanedbatelným přínosům této práce. Srovnání aplikací zadavatele se nachází v kapitole 3.4.1.1. Jedním ze zásadních přínosů této práce je optimalizace správy dat prostřednictvím služby *StreamService*, která automaticky poskytuje data widgetům zaregistrovaným pro odběr dat. Služba si zároveň pamatuje, o jaká data již žádala, a proto žádá server o stejná data pouze jednou, i pokud o ně žádá více widgetů.

Dále byla optimalizována správa pro editaci widgetů. V rámci práce byly vyhodnoceny společné vlastnosti potřebné pro editaci různých widgetů a byla vytvořena nová struktura.

Systém nově kontroluje podporované datové toky. Díky tomu nabízí v editaci widgetu pouze relevantní data, která je widget schopen zpracovat a zobrazit.

Má diplomová práce mě obohatila o mnoho znalostí a zkušeností v oblastech vývoje softwarového projektu. Průchod jednotlivými etapami, od analýzy, přes návrh, implementaci a testování vlastního frameworku, mi přinesl nový náhled na komplexní systémy pro správu a zobrazení dat. Během implementace jsem se seznámila s nejmodernějšími technologiemi pro tvorbu interaktivních webových aplikací.

Literatura

- [1] PÍŠA, Jan. *Vizualizační komponenta do CMS Drupal - verze 2.0* [online]. Praha, 2016 [cit. 2018-03-06]. Dostupné z: Databáze ČVUT DSpace. Magisterská práce. České vysoké učení technické v Praze, Fakulta informačních technologií, Katedra softwarového inženýrství. Vedoucí práce Jindřich Fiedler.
- [2] *FIEDLER: Elektronika pro ekologii* [online]. České Budějovice: FIEDLER AMS, 2017 [cit. 2018-03-13]. Dostupné z: <https://fiedlerams.cz/>
- [3] Chen, Alex Harper, Simon Lunn, Darren Brown, Andy. (2013). *Widget Identification: A High-Level Approach to Accessibility. World Wide Web*. 16. 73-89. 10.1007/s11280-012-0156-6.
- [4] AGGARWAL, Charu C. *Data streams: models and algorithms*. 1. New York: Springer, c2007. ISBN 978-0-387-47534-9.
- [5] JAROSCH, Martin. *Klasifikace v proudu dat pomocí souboru klasifikátorů* [online]. Vysoké učení technické v Brně. Fakulta informačních technologií, 2013 [cit. 2018-03-13]. Dostupné z: <http://hdl.handle.net/11012/52710>. Diplomová práce. Vysoké učení technické v Brně. Fakulta informačních technologií. Ústav informačních systémů. Vedoucí práce Martin Hlosta.
- [6] Time-Based Data: The Definitive Guide [2.x] | Elastic. *Open Source Search & Analytics · Elasticsearch / Elastic* [online]. Elasticsearch, 2018 [cit. 2018-03-13]. Dostupné z: <https://www.elastic.co/guide/en/elasticsearch/guide/current/time-based.html>

- [7] Pojem Big Data a principy distribuovaného zpracování dat. HOLUBOVÁ, Irena, Jiří KOSEK, Karel MINAŘÍK a David NOVÁK. *Big Data a NoSQL databáze*. Praha: Grada, 2015, s. 19-20. Profesionál. ISBN 978-80-247-5466-6.
- [8] The analysis phase Data Dictionary. DEMARCO, Tom. *Structured analysis and system specification*. Englewood Cliffs, N.J.: Prentice-Hall, c1979, 125 - 128. ISBN 0138543801.
- [9] *Open Source ERP and CRM: Odoo* [online]. Belgium: Odoo S.A. [cit. 2018-03-10]. Dostupné z: <https://www.odoo.com/>
- [10] *Open Source ERP and CRM: Odoo* [online]. Belgium: Odoo S.A. [cit. 2018-03-10]. Dostupné z: <https://www.ubnt.com/software/>
- [11] *Ubiquiti Networks - Democratizing Professional Network Technology* [online]. New York: Ubiquiti Networks, 2018 [cit. 2018-03-18]. Dostupné z: www.ubnt.com
- [12] *SensorUp: SensorThings Highchart and DataTable (HcDT)* [online]. Calgary: SensorUp, 2017 [cit. 2018-03-19]. Dostupné z: <https://www.sensorup.com/sensorthings-high-chart-and-data-table-hcdt>
- [13] *Leaflet: an open-source JavaScript library for interactive maps* [online]. OpenStreetMap contributors, 2017 [cit. 2018-03-19]. Dostupné z: <http://leafletjs.com/>
- [14] *Plotly.js: JavaScript Graphing Library* [online]. Plotly, 2015 [cit. 2018-03-19]. Dostupné z: <https://plot.ly/javascript/>
- [15] *Website Title TypeScript - JavaScript that scales.: Get TypeScript* [online]. Microsoft, 2012 [cit. 2018-04-10]. Dostupné z: <https://www.typescriptlang.org/index.html>
- [16] *Taucharts flexible javascript charting library for data exploration* [online]. Apache, 2015 [cit. 2018-04-10]. Dostupné z: <https://www.taucharts.com/>
- [17] Self-Documenting Code. STUEBEN, Michael. *Good habits for great coding: improving programming skills with examples in Python* [online]. New York, NY: Apress, 2018, s. 67-90 [cit. 2018-04-10]. ISBN 978-1-4842-3459-4. Dostupné z: https://link.springer.com/chapter/10.1007/978-1-4842-3459-4_6

- [18] *TSLint*[online]. Palantir Technologies, 2016 [cit. 2018-04-10]. Dostupné z: <https://palantir.github.io/tslint/>
- [19] *Jest Blog ATOM: Jest · Delightful JavaScript Testing* [online]. Facebook, 2018 [cit. 2018-04-10]. Dostupné z: <https://facebook.github.io/jest/>

Příloha A

Seznam použitých zkratk

AMS Automatic Monitoring Systems

API Application

BLOB Binary Large Object

COV Čistírna Odpadních Vod

CRM Customer Relationship Management

CSV Comma-Separated Values

DS Data Stream / Datový tok

DTA Stata dataset

GPS Global Positioning System

JSON JavaScript Object Notation

KML Keyhole Markup Language

KPI Key Performance Indicator

MM Měrné Místo

REST REpresentational State Transfer

SCADA Supervisory Control And Data Acquisition

SPA Stupeň Povodňové Aktivity

SVG Scalable Vector Graphics

UNMS Ubiquiti Network Management System

URL Uniform Resource Locator

XLS XML Spreadsheet

XML Extensible Markup Language

Příloha B

Slovník pojmů

Dashboard Uživatelská obrazovka, která se skládá z widgetů

Data logger Jednotka, která ukládá naměřená data

Jednotka Stanice, která zpravidla měří prostřednictvím sensorů nebo řídí motory

Limnigrafická stanice Jednotka, která slouží k monitorování stavu hladin vodních toků

Telemetrická stanice Jednotka, která vzdáleně odesílá data na server

Widget Interaktivní prvek pro zobrazení dat

Příloha C

Instalační příručka

Tato příloha popisuje postup instalace aplikace.

Prerekvizity pro spuštění aplikace:

- Aurelia.js
- Node.js (aktuální verze)

Příkazy pro instalaci, spuštění a otestování aplikace.

```
npm install // instalace aplikace
```

```
npm start // spuštění aplikace
```

```
npm test // spuštění testů
```

Aplikace běží na adrese *http : //localhost : 8080* Pokud aplikace běží, dochází při uložení změn k jejímu opětovnému načtení.

Příloha D

Obsah příloženého CD

Příložené CD obsahuje následující soubory:

<code>dp_adamkova.pdf</code>	text práce
<code>zadani.pdf</code>	zadávací protokol magisterské práce
<code>dp_implementace_adamkova.zip</code>	implementace