

ŽILINSKÁ UNIVERZITA V ŽILINE
FAKULTA RIADENIA A INFORMATIKY

BAKALÁRSKA PRÁCA

Študijný odbor: **Informatika**

Oľga Chovancová

**Vizualizácia dát získaných pomocou SCADA
systémov s využitím HTML 5 štandardov**

Vedúci: **Ing. Juraj Veverka**

Reg.č. xxx/2015

Máj 2015

Abstrakt

CHOVANCOVÁ OEGA: *Vizualizácia dát získaných pomocou SCADA systémov s využitím HTML 5 štandardov* [Bakalárska práca]

Žilinská Univerzita v Žiline, Fakulta riadenia a informatiky, Katedra TODO.

Vedúci: Ing. Juraj Veverka

Stupeň odbornej kvalifikácie: Inžinier v odbore IPESOFT.... Žilina ...TODO .

FRI ŽU v Žiline, 2015 — ?? s.

Obsahom práce je...

Obsahom práce je vzorová sada grafických komponentov na vizualizáciu technologických procesov s využitím HTML 5 štandardov. Jedná sa o grafické komponenty, ktoré nie sú bežne dostupné na tvorbu interaktívnych webových aplikácií ako napríklad vizualizácie mechanických súčasti hydraulických systémov, technologických liniek, silových a výkonových častí automatizačných sústav. Návrh interface, pomocou, ktorého budú tieto komponenty komunikovať so serverovou časťou SCADA systému. Cieľová platforma pre výslednú webovú aplikáciu bude kompatibilná s rodinou štandardov HTML 5 pre každý webový prehľadávač.

Abstract

CHOVANCOVÁ OLGA: *Data visualization acquired by SCADA systems using HTML5 standarts*
[Bacalar thesis]

University of Žilina, Faculty of Management Science and Informatics, Department of TODO.

Tutor: Ing. Juraj Veverka.

Qualification level: Engineer in field Žilina: TODO

FRI ŽU v Žiline, 2009 — ?? p. TODO

The main idea of this ... TODO

Prehlásenie

Prehlasujem, že som túto prácu napísala samostatne a že som uviedla všetky použité pramene a literatúru, z ktorých som čerpala.

V Žiline, dňa DD.MM.2015

Oľga Chovancová

Obsah

Úvod	4
1 ...	6
1.1 SVG grafické editory	6
1.1.1 Adobe Illustrator	6
1.1.2 CorelDraw	6
1.1.3 Inkscape	7
1.2 JavaScript knižnice SVG	8
1.2.1 D3	8
1.2.2 Raphaël	9
1.2.3 Snap.svg	9
1.2.4 SVG.JS	10
2	11
3	12
4	13
5 Úvod	14
6 Nadpis	16
6.1 Obsah práce	17
6.2 Další forma	18

	3
6.3 5 CITOVANIE A ZOZNAM BIBLIOGRAFICKÝCH ODKAZOV	18
7 Záver	20
Literatúra	21

Úvod

Úvod sa obyčajne píše **až po napísaní jadra práce** – uvádza sa to, čo je napísané!!

..... todo todo TODO TODO
todo todo
TODO TODO todo todo TODO TODO
..... todo todo TODO TODO todo todo TODO
TODO todo todo TODO TODO
todo todo TODO TODO todo todo TODO TODO
..... todo todo TODO TODO todo todo TODO
TODO

Téma práce je vizualizácia technologických dát zo SCADA systémov na webe. Produktom Bakalárskej práce je vzorová sada grafických komponent na vizualizáciu technologických procesov s využitím HTML 5 štandardov. Jedná sa o grafické komponenty, ktoré nie sú bežne dostupné na tvorbu interaktívnych webových aplikácií ako napríklad vizualizácie mechanických súčastí hydraulických systémov, alebo technologických liniek, vizualizácie silových a výkonných častí automatizačných sústav. Návrh interface, pomocou ktorého budú tieto komponenty komunikovať so serverovou časťou SCADA systému.

V súčasnosti je v IPESOF s.r.o. software, ktorý dokáže vizualizovať dáta z technológií pomocou "hrubých klientov", čo sú natívne (exe) Windows aplikácie a je technológia, ktorá dokáže rovnaké dáta zobrazovať na webe.

Aktuálna webová prezentácia takýchto dát nespĺňa súčasne štandardy pre moderne webové aplikácie a preto je potrebné nájsť nový spôsob vizualizácie na webe, ktorá bude v budúcnosti použiteľná na rôznych platformách, nielen na PC.

Cieľová platforma pre výslednú webovú aplikáciu bude každý web prehliadač kompatibilný s rodinou štandardov HTML 5. Riešenie bude využívať výhradne open-source knižnice s licenciami typu MIT, GNU GPL, BSD. Zdrojové kódy práce budú udržiavané v Git repository.

Predbežný postup práce:

1. Analýza požiadaviek, prieskum možnosti využitia wyswing editorov na tvorbu grafických komponent s možnosťou exportu do formátov SVG, JSON, XML, alebo JavaScript.
2. Výber vhodných open-source knižníc na tvorbu grafických komponent kompatibilných s HTML 5.
3. Návrh REST API na prepojenie grafických komponent so SCADA serverom.
4. Analýza možnosti automatického mapovania API grafických prvkov pomocou metadát na existujúce API dostupné pre SCADA server D2000.
5. Implementácia vzorovej sady grafických komponent.
6. Analýza výkonnosti a výkonnostné obmedzenia.

Kapitola 1

...

Postup práce:

- Nájsť spôsob ako nakresliť vizualizáciu v open-source SVG grafickom editore, kde výstupom sú grafické objekty vo formáte SVG.
- Nájsť spôsob ako výsledné vytvorené SVG objekty, použiť na webovej stránke a prepojiť s funkciami napísanými v JavaScripte na ovládanie animácií, nastavovanie farieb a podobne.

1.1 SVG grafické editory

1.1.1 Adobe Illustrator

Adobe Illustrator is a vector graphics editor developed and marketed by Adobe Systems. It's not free and not open source. <http://www.adobe.com/products/illustrator.html>

1.1.2 CorelDraw

CorelDraw (styled CorelDRAW) is a vector graphics editor developed and marketed by Corel Corporation of Ottawa, Canada. CorelDRAW X7, a vector-illustration and page-layout application. It's not free and not open source. <http://www.coreldraw.com/rw/> <http://www.coreldraw.com/rw/products/design-software/>

1.1.3 Inkscape

Inkscape is a professional vector graphics editor for Windows, Mac OS X and Linux. It's free and open source. <http://www.inkscape.org/en/about/features/>

Features of Inkscape

Object creation

- Drawing: pencil tool (freehand drawing with simple paths), pen tool (creating Bézier curves and straight lines), calligraphy tool (freehand drawing using filled paths representing calligraphic strokes)
- Shape tools: rectangles (may have rounded corners), ellipses (includes circles, arcs, segments), stars/polygons (can be rounded and/or randomized), spirals
- Clones ("live" linked copies of objects), including a tool to create patterns and arrangements of clones

Object manipulation

- Transformations (moving, scaling, rotating, skewing), both interactively and by specifying exact numeric values
- Z-order operations (raising and lowering)
- Grouping objects ("select in group" without ungrouping, or "enter the group" making it a temporary layer)
- Layers (lock and/or hide individual layers, rearrange them, etc; layers can form a hierarchical tree)
- Alignment and distribution commands

Operations on paths

- Node editing: moving nodes and Bezier handles, node alignment and distribution, etc.
- Converting to path (for text objects or shapes), including converting stroke to path

- Boolean operations
- Path simplification, with variable threshold
- Path inseting and outseting, including dynamic and linked offset objects

Rendering

- Fully anti-aliased display Alpha transparency support for display and PNG export Complete “as you drag” rendering of objects during interactive transformations

Misc

- Live watching and editing the document tree in the XML editor PNG, OpenDocument Drawing, DXF, sk1, PDF, EPS and PostScript export formats Command line options for export and conversions Perfectly compliant SVG format file generation and editing

1.2 JavaScript knihovnice SVG

1.2.1 D3

D3 - **D**ata **D**riven **D**ocument - <http://d3js.org/>

D3.js is a JavaScript library for manipulating documents based on data. D3 helps you bring data to life using HTML, SVG and CSS. D3’s emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.

D3 allows you to bind arbitrary data to a Document Object Model (DOM), and then apply data-driven transformations to the document. For example, you can use D3 to generate an HTML table from an array of numbers. Or, use the same data to create an interactive SVG bar chart with smooth transitions and interaction.

D3 is not a monolithic framework that seeks to provide every conceivable feature. Instead, D3 solves the crux of the problem: efficient manipulation of documents based on data. This avoids proprietary representation and affords extraordinary flexibility, exposing the full

capabilities of web standards such as CSS3, HTML5 and SVG. With minimal overhead, D3 is extremely fast, supporting large datasets and dynamic behaviors for interaction and animation. D3's functional style allows code reuse through a diverse collection of components and plugins.

1.2.2 Raphaël

[http://raphaeljs.com/](http://dmitrybaranovskiy.github.io/raphael/)

Raphaël is a small JavaScript library that should simplify your work with vector graphics on the web. If you want to create your own specific chart or image crop and rotate widget, for example, you can achieve it simply and easily with this library.

Raphaël uses the SVG W3C Recommendation and VML as a base for creating graphics. This means every graphical object you create is also a DOM object, so you can attach JavaScript event handlers or modify them later. Raphaël's goal is to provide an adapter that will make drawing vector art compatible cross-browser and easy.

Raphaël currently supports Firefox 3.0+, Safari 3.0+, Chrome 5.0+, Opera 9.5+ and Internet Explorer 6.0+.

1.2.3 Snap.svg

[http://snapsvg.io/](http://dmitrybaranovskiy.github.io/snap.svg/)

Snap.svg is a brand new JavaScript library for working with SVG. Snap provides web developers with a clean, streamlined, intuitive, and powerful API for animating and manipulating both existing SVG content, and SVG content generated with Snap.

Currently, the most popular library for working with SVG is Raphaël. One of the primary reasons Raphaël became the de facto standard is that it supports browsers all the way back to IE 6. However, supporting so many browsers means only being able to implement a common subset of SVG features. Snap was written entirely from scratch by the author of Raphaël (Dmitry Baranovskiy), and is designed specifically for modern browsers (IE9 and up, Safari, Chrome, Firefox, and Opera). Targeting more modern browsers means that Snap can support features like masking, clipping, patterns, full gradients, groups, and more.

Another unique feature of Snap is its ability to work with existing SVG. That means your SVG content does not have to be generated with Snap for you to be able to use Snap to work with it (think “jQuery or Zepto for SVG”). That means you create SVG content in tools like Illustrator, Inkscape, or Sketch then animate or otherwise manipulate it using Snap. You can even work with strings of SVG (for example, SVG files loaded via Ajax) without having to actually render it first which means you can do things like query specific shapes out of an SVG file, essentially turning it into a resource container or sprite sheet.

Finally, Snap supports animation. By providing a simple and intuitive JavaScript API for animation, Snap can help make your SVG content more interactive and engaging.

Snap is free and open-source (released under an Apache 2 license).

1.2.4 SVG.JS

<http://www.svgjs.com/>

A lightweight library for manipulating and animating SVG.

- easy readable uncluttered syntax
- animations on size, position, transformations, color, ...
- painless extension thanks to the modular structure
- various useful plugins available
- unified api between shape types with move, size, center, ...
- binding events to elements
- full support for opacity masks and clipping paths
- text paths, even animated
- element groups and sets
- dynamic gradients

Kapitola 2

Kapitola 3

Kapitola 4

Kapitola 5

Úvod

Úvod sa obyčajne píše **až po napísaní jadra práce** – uvádza sa to, čo je napísané!! Jeho význam spočíva v tom, že poskytuje čitateľovi jasné informácie o riešenom probléme a o dôvodoch, prečo sa ho autor rozhodol riešiť. Obsahuje najmä:

- výstižný opis skúmanej problematiky, jej významu, stručnú charakteristiku stavu poznania problematiky, špecifiká skúmanej oblasti,
- stručné oboznámenie s účelom, autorským zámerom a cieľmi práce, zdôvodnenie jej dôležitosti a významu, prípadne vzťah k iným prácam podobného zamerania,
- stručný prehľad najpodstatnejších zdrojov, literárnych i iných, využitých pri spracovaní práce.

Tu by bolo dobré zoznámiť a zaradiť problematiku práce. Je dobré mať na pamäti, na základe akých kritérií bude oponent hodnotiť túto prácu. Náročnosť zadania sa hodnotí slovne ako malá, stredná, veľká na základe nasledujúcich kritérií:

- ♡ teoretické znalosti,
- * invenčnosť, tvorivosť,
- experimentálna činnosť
- technické práce vrátane programovania,

- návrh algoritmu, datových štruktúr,
- informačno rešeržný prieskum a syntéza.

Dôležitejšie pre záverečné hodnotenie je však je bodové hodnotenie na základe nasledujúcich kritérií:

1. Hĺbka analýzy vo vzťahu k téme [10b]
- 1b) Adekvátnosť použitých metód [15b]
2. Splnenie cieľov zadania [20b]
3. Kvalita riešenia [15b]
4. Logická stavba, nadvúznosť, úplnosť, zrozumiteľnosť [10b]
5. Formálna gramatická úroveň práce, dokumentácie a prezentácie 10b

Ak by oponent robil toto hodnotenie v Exelovskej tabuľke a v bunke D21 by mal súčet bodov, potom výsledná známka bude

```
=if(D21>=90;"A";if(D21>=80;"B";if(D21>=70;"C";if(D21>=60;"D";if(D21>=50;"E";"Fx")))).
```

Kapitola 6

Nadpis

Tu je potrebné popísať doteraz získané poznatky z problematiky. Nezabudnúť dôsledne citovať autorov článkov, kníh aj internetových publikácií napr. monografia [2]. V prameňoch – spravidla posledná kapitola – treba uviesť všetku použitú literatúru. **Nemala by obsahovať tie zdroje, ktoré nie sú v práci citované.** A tiež nie je vhodné citovať nedôveryhodné zdroje ako sú Wikipédia ap.

Jadro práce sa člení na kapitoly a podkapitoly – t.j. podčasti druhej, prípadne tretej úrovne. Vlastný text obsahuje vlastnými slovami formulované úvahy, analýzy, výpočty, vlastné myšlienky – závery a návrhy, parafrázy (vlastnými slovami prerozprávaný originálny text – treba uviesť odkaz na zdroj), výťahy (krátke zhrnutia originálnej pasáže – uviesť odkaz na zdroj), citáty (doslovne prevzatý text – píše sa vždy v úvodzovkách s uvedením odkazu na zdroj), ilustrácie (grafy, obrázky, tabuľky, schémy – u prevzatých je treba vždy uviesť zdroj).

Jadro záverečných prác spravidla obsahuje tieto časti:

- charakteristika a súčasný stav riešenej problematiky v SR a vo svete,
- cieľ záverečnej práce a použité metódy práce (charakteristika použitých metód, techník a postupov, ich využitie),
- výsledky práce a diskusia.

Jadro bakalárskej a diplomovej práce je vhodné rozdeliť na 2 - 4 kapitoly (podľa potreby a dohody s vedúcim - ich počet nie je direktívne stanovený). Vyššie uvedené časti môžu

byť samostatnými kapitolami, alebo sa môžu spájať, napr. cieľ a metódy alebo výsledky a diskusia. Tie môžu byť ďalej členené na podkapitoly druhej, výnimočne tretej úrovne. Názvy jednotlivých kapitol a podkapitol musia vystihovať podstatu ich obsahu.

6.1 Obsah práce

b) **Obsah a štruktúra jadra praktickej, resp. prípadovej štúdie:**

- Charakteristika riešeného problému a opis najdôležitejších poznatkov z odbornej a vedeckej literatúry, ktoré sa vzťahujú k danému problému, či prípadu (cca 10 – 15 strán).
- Ciele a metódy práce – presná a výstižná charakteristika riešenej problematiky a zdôvodnenie zvoleného postupu a použitých metód (cca 1 – 3 strany).
- Výsledky práce predstavujú výsledky skúmania objektu, ktorého sa téma týka a výsledky analýzy konkrétneho skúmaného prípadu.
 - **Analýzou objektu** (organizácie, podniku, odvetvia, javu, procesu, dokumentácie a i.) rozumieme proces identifikácie, opisu, rozboru prvkov, funkcií a vzťahov objektu pomocou zvoleného postupu.
 - **Analýza prípadu** predstavuje systematický, primerane podrobný a výstižný opis konkrétneho prípadu a jeho súvislostí z praxe krízového manažmentu (bezpečnostného manažmentu, riadenia záchranných prác a pod.), identifikácia a interpretácia podstatných zistení (cca 10 strán).
- Diskusia má v tomto prípade podobu zhrnutia a návrhov opatrení, t.j. obsahuje zhrnutie pozitívnych a negatívnych poznatkov o objekte, procese či prípade, porovnanie s inými podobnými prípadmi a poznatkami z preštudovanej literatúry, spracovanie súboru opatrení alebo uceleného návrhu na zvýšenie účinnosti, na zefektívnenie činností a operácií, či na odstránenie problémových miest (5 – 10 strán).

6.2 Ďalšia forma

V prípade realizácie vlastného prieskumu (sociologického, ekonomického a pod.) je nevyhnutné vypracovať jeho projekt. V ňom je potrebné vymedziť:

- cieľ, pracovné hypotézy, úlohy prieskumu,
- použité metódy a techniky prieskumu,
- organizáciu a priebeh vykonania prieskumu,
- popis prieskumnej vzorky (objektu),

Podľa projektu realizovať plánované etapy prieskumu, ktorými sú najmä:

- zber materiálov (faktov) pomocou metód a techník výskumu,
- spracovanie informácií, ich výber a členenie na jednotlivé zložky, kvantifikácia získaných dát, ich matematicko-štatistické spracovanie, grafické znázornenie,
- prehľad výsledkov prieskumu - interpretácia získaných údajov, predstavujúca kvalitatívne hodnotenie materiálu s formuláciou čiastkových a celkových záverov, doložených logickou argumentáciou a dokumentovaných spracovaným materiálom z prieskumu a vyvodenie teoretických záverov i odporúčaní pre prax,
- celkové spracovanie správy o priebehu a výsledkoch prieskumu je súčasťou záverečnej práce. Interpretácii výsledkov prieskumu je možno venovať jednu kapitolu, alebo jej časť.

6.3 5 CITOVANIE A ZOZNAM BIBLIOGRAFICKÝCH ODKAZOV

Citovanie a citáty slúžia na argumentáciu. Vhodné sú najmä tam, kde svojou výstižnosťou ušetria zdĺhavý výklad a v prípade, keď vlastné názory a argumenty možno podprieť citátom

autority v odbore. Osamotený citát bez vlastnej argumentácie nemá platnosť vecného dôkazu. Citáty využívame aj na prezentáciu názorov, s ktorými polemizujeme, alebo na porovnanie so zistenými faktami alebo inými názormi.

Pri používaní citátov treba rešpektovať nasledujúce pravidlá:

- citát musí byť uvedený doslovne a presne so všetkými typografickými zvláštnosťami (vrátane možných chýb - preklepov) a vždy v úvodzovkách,
- citát treba vybrať tak, aby vyjadroval ucelenú myšlienku a nebol po vyňatí z pôvodného kontextu porušený jeho zmysel,
- prevzaté citáty overiť v origináli (ak je dostupný), u neoverených citátov uviesť zdroj, z ktorého boli prevzaté,
- citáty treba používať priamo v texte záverečnej práce (nie pod čiarou),
- odlišujeme citát (doslovný text) od jeho výkladu vlastnými slovami. Pri použití voľného výkladu myšlienky autora (jej parafrázy) nedávame text do úvodzoviek, ale označíme ho uvedením pôvodného zdroja podľa niektorého z nižšie uvedených príkladov (podľa STN ISO 690 a STN ISO 690-2).

Technikou citovania rozumieme spájanie miesta v texte so záznamami o dokumentoch, ktoré sú v zozname bibliografických odkazov pri rešpektovaní medzinárodných noriem ISO (Kucianová, 2008).¹

Kapitola 7

Záver

Záver musí byť vyústením výkladu, úvah a argumentov uvedených v jadre práce. Mal by obsahovať najmä:

- vecné zhrnutie hlavných myšlienok a dosiahnutých výsledkov,
- vyjadrenie a sumarizáciu vlastného pohľadu na riešený problém a vlastného prínosu,
- zhodnotenie splnenia cieľa záverečnej práce (overenia pracovných hypotéz),
- naznačenie sporných otázok, otázok do diskusie, otvorených problémov,
- formuláciu praktických odporúčaní nadväzujúcich na výklad, analýzy a argumenty obsiahnuté v jadre práce.

Literatúra

- [1] Bartsch H. J., *Matematické vzorce*, 3. revidované vydání, Praha, Mladá fronta 2000, ISBN 80-204-0607-7.
- [2] Berman G. N., *Zbierka úloh z matematickej analýzy*, Bratislava, ŠNTL 1955.
- [3] Peško, Š., *Pohodlná optimalizácia reálnych úloh v tabuľkových procesoroch*, Slovak Society for Operations Research, 7th international seminar, Application of Quantitative Methods in Research and Practice (2005), pp. 29–35, Remata, ISBN 80-225-2079-9.
- [4] World of mathematics, A Wolfram Web Resource, <http://mathworld.wolfram.com/>,
WolframAlpha – computational knowledge engine, <http://www.wolframalpha.com/>.