

UNIVERZITA KOMENSKÉHO V BRATISLAVE

FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

# ONLINE APLIKÁCIA PRE VIZUALIZÁCIU DÁT

bakalárska práca

Tibor Stanko

Študijný program: **Matematika**

Študijný odbor: **1113 Matematika**

Vedúci bakalárskej práce: **Mgr. Matej Novotný, PhD.**

Bratislava 2012





Univerzita Komenského v Bratislave  
Fakulta matematiky, fyziky a informatiky

## ZADANIE ZÁVEREČNEJ PRÁCE

**Meno a priezvisko študenta:** Tibor Stanko  
**Študijný program:** matematika (Jednoodborové štúdium, bakalársky I. st., denná forma)  
**Študijný odbor:** 9.1.1. matematika  
**Typ záverečnej práce:** bakalárska  
**Jazyk záverečnej práce:** slovenský

**Názov:** Online aplikácia pre vizualizáciu dát

**Cieľ:** Vytvoriť webovskú aplikáciu, ktorá používateľovi umožní nahrat' alebo prilinkovať svoje dáta a následne z nich vytvoriť vizualizáciu (graf.) Výsledný graf je vo forme statického obrázku, obrázku dynamicky generovaného z dát alebo vo forme interaktívnej vizualizácie. Formát vstupných dát je XLS(X), CSV a XML. Implementácia bude obsahovať aspoň 4 rôzne vizualizácie z toho 1 interaktívnu a možnosť nastavenia základných parametrov vizualizácie (popis grafu, farby, orientácia osí...)

**Anotácia:** Vytváranie jednoduchých grafov - vizualizácií dát - je obsiahnuté napríklad v desktopových tabuľkových procesoroch akými sú Microsoft Excel alebo OpenOffice Calc. Publikovanie grafov na webe sa obvykle robí tak, že z dát vytvorí autor obrázok v niektorom spomenutom alebo podobnom programe a následne obrázok nahra na webovskú stránku. Takýto prístup má viacero nevýhod. Je nutné pracovať vo veľkom a komplikovanom softvéri, graf je nutné pri zmene dát manuálne aktualizovať, zmena vizuálnych parametrov je po vytvorení zložitá, nie je možné vytvoriť interaktívnu vizualizáciu. V tejto práci vznikne online prostredie, ktoré umožní ľuďom prispievajúcim na web, ale aj iným, vytvoriť zo svojich dát vizualizácie, ktoré uvedené nedostatky prekonávajú. Systém bude vedieť čítať dáta z Excelu, Calcu, XML a CSV formátov. Dôležitým atribútom bude možnosť dáta neposkytnúť službe priamo ale v podobe hyperlinky. Takýto spôsob vytvára priestor pre dynamickú automatickú aktualizáciu vizualizácie v prípade zmeny podkladových dát. Veríme, že takáto služba nájde uplatnenie medzi žurnalistami, bloggermi, a inými aktívnymi používateľmi internetu. Ako príklad použiteľnosti prostredia bude v práci implementovaných viacero techník pre vizualizáciu dát (koláčový diagram, stĺpcový graf, čiarový graf, scatterplot a iné) a minimálne jedna z techník bude vo forme interaktívneho appletu v technológii Flash, DHTML alebo HTML5. Vytvorený systém bude základom pre ďalšie rozširovanie o nové techniky vizualizácie.

**Vedúci:** Mgr. Matej Novotný, PhD.  
**Katedra:** FMFI.KAI - Katedra aplikovanej informatiky  
**Dátum zadania:** 05.10.2011

**Dátum schválenia:** 11.10.2011

prof. RNDr. Ján Filo, CSc.  
garant študijného programu

.....  
študent

.....  
vedúci práce

Čestne prehlasujem, že som túto bakalársku prácu vypracoval  
samostatne s použitím citovaných zdrojov.

.....

Bratislava, jún 2012

Ďakujem vedúcemu mojej práce Mgr. Matejovi Novotnému, PhD.  
za odborné vedenie a čas, ktorý mi venoval pri jej písaní. Jeho  
pripomienky a odpovede na moje otázky mnohé uľahčili.

Ďakujem aj svojej rodine a priateľom za veľkú podporu, ktorú  
mi prejavujú, a trpezlivosť, ktorú so mnou majú.

# Abstrakt

STANKO Tibor. *Online aplikácia pre vizualizáciu dát*. [Bakalárska práca] Univerzita Komenského v Bratislave. Fakulta matematiky, fyziky a informatiky. Vedúci: Mgr. Matej Novotný, PhD. Bratislava: UK, 2012.

Bakalárska práca prezentuje návrh a implementáciu novej aplikácie *Grafit*, ktorá umožňuje vizualizáciu dát v online prostredí. Dôraz je kladený na jednoduchosť a ľahkú rozširiteľnosť o ďalšie techniky vizualizácie. Cieľovou skupinou aplikácie sú bežní používatelia webu.

Kľúčové slová: online aplikácia, vizualizácia, dáta, graf

# Abstract

STANKO Tibor. *Online aplikácia pre vizualizáciu dát*. [Bakalárska práca] Univerzita Komenského v Bratislave. Fakulta matematiky, fyziky a informatiky. Vedúci: Mgr. Matej Novotný, PhD. Bratislava: UK, 2012.

This bachelor thesis presents design and implementation of new application *Grafit* intended for data visualisation in online environment. Emphasis is placed on simplicity and ease of expansion for further techniques of visualisation. Target audience for this application are common web users.

Keywords: online application, visualisation, data, graph

# Obsah

<b>1</b>	<b>Úvod</b>	<b>1</b>
<b>2</b>	<b>Prehľad problematiky</b>	<b>3</b>
2.1	Vizualizácia informácií . . . . .	3
2.2	Online aplikácie . . . . .	4
2.3	Podobné služby . . . . .	5
2.3.1	Desktopové softvéry . . . . .	5
2.3.2	Online aplikácie . . . . .	6
2.3.3	Jednoduché online nástroje . . . . .	9
2.3.4	Knižnice . . . . .	10
<b>3</b>	<b>Návrh riešenia</b>	<b>11</b>
3.1	Serverová časť . . . . .	11
3.1.1	Databáza . . . . .	11
3.1.2	Obsluha databázy . . . . .	15
3.1.3	Definícia typov vizualizácií . . . . .	17
3.1.4	Generovanie grafu . . . . .	20
3.2	Používateľská časť . . . . .	20



3.2.1	Registrácia a prihlásenie . . . . .	20
3.2.2	Import a správa dát . . . . .	20
3.2.3	Výber typu a vytvorenie grafu . . . . .	21
3.2.4	Nastavenie a export grafu . . . . .	21
3.2.5	Správa grafov . . . . .	21
3.2.6	Nastavenia účtu a vymazanie konta . . . . .	22
3.3	Administrácia aplikácie . . . . .	22
<b>4</b>	<b>Implementácia</b>	<b>23</b>
4.1	Serverová časť . . . . .	23
4.1.1	Jazyk PHP . . . . .	23
4.1.2	MySQL a prepojenie s PHP . . . . .	24
4.1.3	Spracovávanie dát . . . . .	24
4.1.4	.htaccess . . . . .	24
4.2	Klientská časť . . . . .	25
4.2.1	XHTML a CSS . . . . .	25
4.2.2	JavaScript, jQuery, Ajax . . . . .	27
4.2.3	Techniky vizualizácie a HTML5 . . . . .	28
<b>5</b>	<b>Výsledky</b>	<b>29</b>
<b>6</b>	<b>Záver</b>	<b>36</b>
	<b>Prílohy</b>	<b>38</b>

# Zoznam obrázkov

2.1	Tvorba koláčového diagramu v Exceli . . . . .	6
2.2	Online služba RichGraphs . . . . .	7
2.3	Jednoduchý online nástroj Chart Part . . . . .	10
3.1	Diagram väzieb v databáze . . . . .	15
3.2	Náhľad dát vygenerovaný triedou Dáta . . . . .	16
4.1	Príklad použitia CSS 3 . . . . .	25
4.2	Vizuálne vylepšenia cez jQuery . . . . .	27
	(a) Fancybox . . . . .	27
	(b) miniColors . . . . .	27
	(c) Uniform.js . . . . .	27
5.1	Úvodná stránka . . . . .	29
5.2	Správa dát . . . . .	30
5.3	Pridanie nových dát . . . . .	30
5.4	Správa grafov . . . . .	30
5.5	Detail existujúcich dát . . . . .	31
5.6	Vytvorenie nového grafu . . . . .	31
5.7	Nastavenie existujúceho grafu . . . . .	32

5.8	Stĺpcový graf . . . . .	33
5.9	Čiarový graf (interaktívny) . . . . .	33
5.10	Koláčový diagram (interaktívny) . . . . .	34
5.11	Relačný diagram (Scatter plot) . . . . .	34
5.12	Vloženie grafu do webstránky . . . . .	35

# Zoznam tabuliek

2.1	Rozloha kontinentov . . . . .	4
3.1	Prehľad väzieb v databáze . . . . .	14

# Zoznam zdrojových kódov

3.1	Príklad definície parametrov typu vizualizácie (PHP) . . . . .	18
3.2	Príklad konfiguračného súboru . . . . .	19
4.1	Použitie CSS 3 pre dosiahnutie efektu na obrázku 4.1 . . . . .	26
4.2	Použitie Ajaxu na načítanie náhľadu . . . . .	28

# Kapitola 1

## Úvod

Publikovanie informácií (najmä dát) na webe je často spojené s vytvorením vizualizácií, ktoré tieto informácie vysvetľujú či dopĺňajú. To však nemusí byť úplne triviálna úloha. Bežní používatelia majú v praxi dve možnosti - vytvoriť graf v desktopovom softvéri, alebo použiť niektorú z webových služieb, ktoré ponúkajú vytváranie vizualizácií. V drvivej väčšine prípadov je výstupom statický obrázok, ktorý sa nedá po uložení editovať - či už ide o zmenu zdrojových dát alebo úpravu vizuálnych parametrov grafu.

Táto práca opisuje návrh a implementáciu online aplikácie *Grafit*, ktorá bude slúžiť na takéto vizualizovanie informácií. *Grafit* bude prekonávať ho-  
re uvedený nedostatok - vizualizácie budú generované dynamicky, preto bude možná jednoduchá zmena zdrojových dát, resp. parametrov grafu aj po jeho vytvorení a uložení.

Okrem statického súboru bude systém vedieť načítavať tiež dáta zadané vo forme odkazu na online umiestnenie (URL) dátového súboru. Takto poskytnuté dáta sú pripravené na automatickú aktualizáciu v prípade ich zmeny.

Na ukážku funkčnosti budú v aplikácii implementované štyri techniky vizualizácie. Dve z nich majú na výstupe statický obrázok (stĺpcový graf, relačný diagram), ďalšie dve sú interaktívne s využitím technológie HTML5 (čiarový graf, koláčový diagram). *Grafit* bude ľahko rozšíriteľný o ďalšie techniky.

Služba bude určená pre všetkých používateľov webu, ktorí chcú oživiť svoje webstránky či blogy. Okrem ovládania základov práce s webom nevyžaduje ďalšie zručnosti.

# Kapitola 2

## Prehľad problematiky

### 2.1 Vizualizácia informácií

Moderná doba nás zahľcuje množstvom informácií, v ktorých sa často nedokážeme orietovať. Neustály prílív dát si vyžaduje spôsob ako ich efektívne skúmať tak, aby sme z nich získali podstatné informácie. Tu vstupuje do hry vizualizácia.

Vizualizáciu môžeme definovať ako „*kognitívnu aktivitu, ktorá prebieha v mysli človeka a jej výsledkom je sformovanie interného modelu, ktorý umožňuje pochopiť význam zobrazených dát a vzťahy medzi nimi.*“ [Paľ09] Delí sa na vedeckú, medicínsku, štatistickú, hudobnú, geovizualizáciu atď.

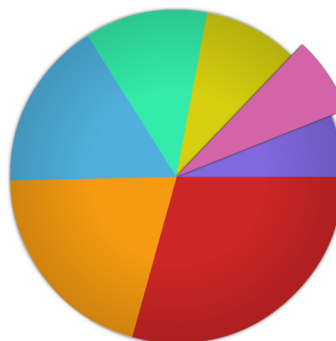
V tejto práci nás bude najviac zaujímať vizualizácia informácií, ktorej úlohou je sprostredkovanie informácie jednoduchou a zrozumiteľnou formou pre verejnosť, napr. v podobe diagramu (angl. *chart*).

O čo teda ide? Ako jednoduchý príklad si zoberme dáta v tabuľke 2.1. Na prvý pohľad nám čísla v pravom stĺpci veľa nehovoria. Informáciu, ktorú



v sebe nesú, oveľa rýchlejšie vyčítame z vytvoreného koláčového diagramu.

kontinent	rozloha v $km^2$
Ázia	43 820 000
Afrika	30 370 000
Severná Amerika	24 490 000
Južná Amerika	17 840 000
Antarktída	13 720 000
Európa	10 180 000
Austrália	9 008 500



**Tabuľka 2.1:** *Rozloha kontinentov*

## 2.2 Online aplikácie

Online alebo webová aplikácia je aplikácia, ktorá je prístupná cez sieť (napr. Internet) bez nutnosti jej inštalácie do lokálneho počítača. Zvyčajne využíva kombináciu skriptovania na strane servera (PHP, ASP) a na strane klienta (JavaScript).

Popularita webu a rozmach mobilných technológií (smartfóny, tablety) za posledné roky spôsobuje stále väčšiu obľúbenosť online aplikácií. Ich veľkou výhodou je to, že nie sú závislé od operačného systému - všetko, čo používateľ potrebuje na ich spustenie je internetové pripojenie a webový prehliadač.

## 2.3 Podobné služby

V tejto časti vyberáme niekoľko služieb podobných *Grafitu*. Rozdelené sú do štyroch skupín - desktopové softvéry, online aplikácie, jednoduché online nástroje a knižnice. Ich kompletný zoznam (spolu s URL) tvorí prílohu na strane 38.

### 2.3.1 Desktopové softvéry

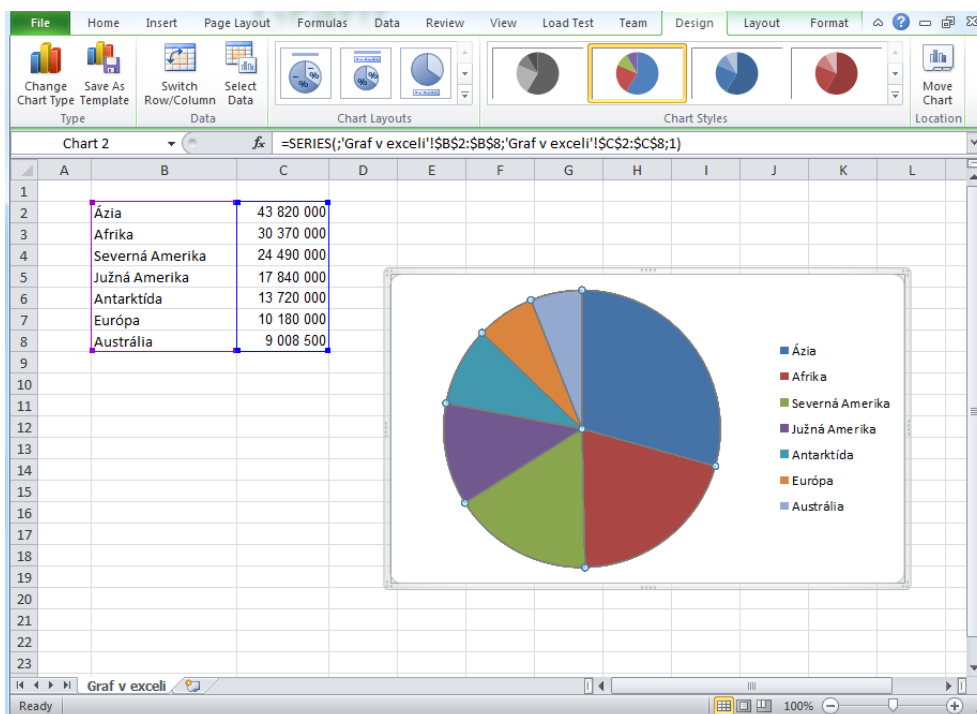
Medzi najznámejšie desktopové softvéry, ktoré umožňujú vizualizovať dáta, patria tabuľkové procesory **Microsoft Excel** a **OpenOffice Calc**. Medzi ich výhody patrí široká škála grafov, z ktorých má používateľ na výber, ich intuitívne vytváranie a tiež pridávanie väčšieho množstva vizualizácií do jedného dokumentu.

Ďalšími desktopovými softvérmi sú **Swift Chart**, **Grapher9** a **SmartDraw**. Narozdiel od Excelu a Calcu je vytváranie grafov ich primárnou funkciou. Vstupné dáta sa zadávajú manuálne alebo sa automaticky načítajú z textového súboru či excelovskej tabuľky. Grafy je možné upravovať vo WYSIWYG<sup>1</sup> editore a exportovať ako statický obrázok (JPG, PNG, SVG, ...), flashový súbor (SWF), prezentáciu (PPT) či dokument (PDF).

Všetky vyššie uvedené desktopové softvéry s výnimkou Open Office Calc sú platené - ich ceny sa približne pohybujú od 120 € za základný balík MS Office až po 270 € za Grapher 9. Pre bežného používateľa môžu byť tieto ceny privysoké.

---

<sup>1</sup>WYSIWYG = **W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **G**et - dosl. *čo vidíš, to dostaneš*.

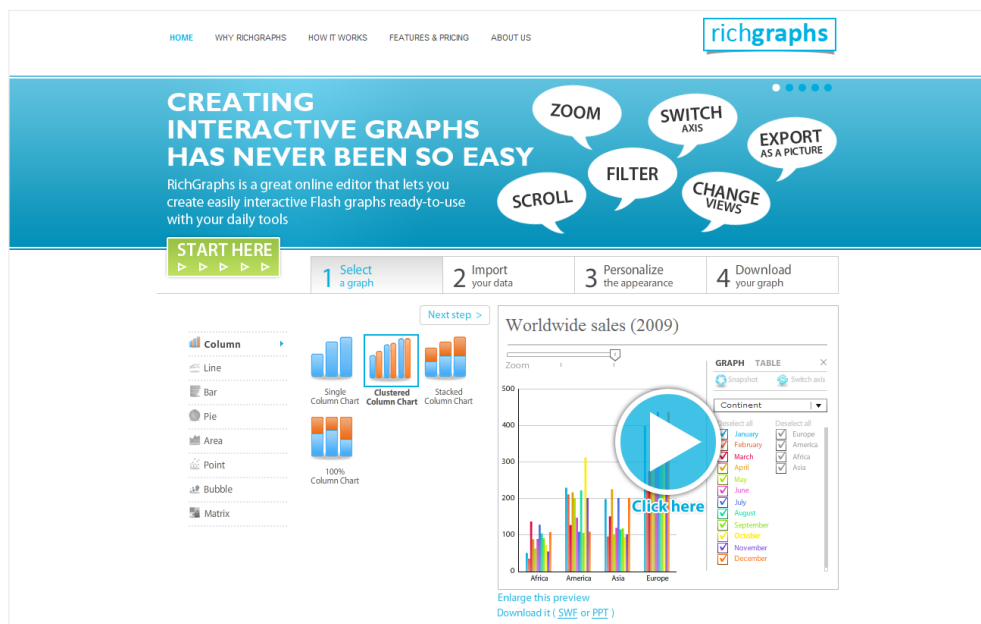


Obr. 2.1: Tvorba koláčového diagramu v Exceli

### 2.3.2 Online aplikácie

Na webe existuje množstvo webstránok s aplikáciami, ktoré ponúkajú tvorbu grafov z dát zadaných používateľom. Líšia sa od seba v type vstupu a výstupu, množstve grafov, z ktorých si používateľ môže vybrať, či v množstve parametrov, ktoré môže nastaviť.

Prvou skupinou online aplikácií sú služby bežiacie na flashovom rozhraní. **RichGraphs** je služba, umožňujúca vytvorenie grafu z dát uložených v excelovskej tabuľke. Výstupom je flashový súbor (SWF) a prezentácia (PPT). Služba ponúka na výber z 23 typov vizualizácií rozdelených do 8 kategórií. Vytvorenie grafu prebieha v štyroch jednoduchých krokoch.



**Obr. 2.2:** Online služba RichGraphs

Medzi flashové služby patrí aj **iCharts**. Jej použitie je podmienené registráciou. Používateľ môže dáta zadať manuálne, cez iCharts API alebo ich načítať z excelovskej či Google tabuľky. V ponuke je 14 typov vizualizácií v 4 kategóriách, výstupom je statický obrázok (JPG, PNG), platená verzia ponúka aj výstup vo forme PDF dokumentu.

Ďalšou flashovou službou je **Rich Chart Live**. Dáta je potrebné manuálne zadať alebo skopírovať z tabuľky (Excel, Open Office), automatický import služba neumožňuje. Vizualizácie sú rozdelené do 8 kategórií, výsledný interaktívny flashový súbor môže používateľ uložiť do svojho počítača alebo priamo vložiť do webstránky.

Druhou skupinou online aplikácií sú služby využívajúce kombináciu HTML a JavaScriptu. Prvou z nich je aplikácia **Create a Graph**. Jej najväčšou ne-

výhodou je, že neumožňuje načítanie dát z externého súboru - je potrebné zadať ich manuálne. Používateľ si môže vybrať z 5 typov vizualizácií a 6 typov výstupných súborov: PDF, SVG, PNG, JPG, EMF alebo EPS.

Ďalšia aplikácia má názov **DIY (Do It Yourself)**. Dáta sa zadávajú manuálne alebo sa načítajú automaticky z textového či CSV súboru. Táto služba je zároveň jedinou, ktorá umožňuje dynamické načítavanie dát z URL. Na výber ponúka 28 typov grafov, výstupom je statický obrázok vo formáte BMP, JPG, PNG alebo EMF, prípadne URL na PNG obrázok, ktorý sa môže dynamicky meniť ak sú aj dáta načítavané dynamicky.

**Chart Tool** umožňuje rýchle vytváranie statických grafov z manuálne zadaných dát. Ponúka 10 rôznych typov vizualizácií, ktoré sa dajú exportovať ako obrázok (PNG, JPG) alebo PDF dokument.

Zaujímavosťou medzi online aplikáciami je webstránka **StatCrunch**. Okrem načítavania dát z lokálneho súboru alebo URL adresy ponúka možnosť získať dáta z prieskumu, ktorý si používateľ sám vytvorí. Z nich potom systém vygeneruje vizualizáciu (statický PNG obrázok). Používateľ potom môže vytvorený graf skopírovať do schránky, vytlačiť, vložiť do webstránky alebo priamo zdieľať na sociálnych sieťach (Facebook, Twitter).

**ChartGizmo** je poslednou aplikáciou v tejto skupine. Dáta sa zadávajú podobne ako pri flashovej aplikácii Rich Chart Live - manuálne, alebo skopírovaním z tabuľky. Aplikácia ponúka na výber z 10 typov vizualizácií, výstupom je URL grafu vo forme statického obrázku alebo interaktívneho flashu.

Tretou skupinou sú Java aplikácie. Táto skupina má v našom prehľade len jedného zástupcu, ktorým je služba **Chartle.net**. Dáta je potrebné za-

dať manuálne, v ponuke je 25 grafových typov rozdelených do 9 kategórií. Výstupom aplikácie je URL grafu, ktorý si používateľ môže vložiť na svoj web.

Všetky spomenuté služby sú dostupné bezplatne. Výnimkou je aplikácia RichGraphs, ktorá je spoplatnená; bezplatná verzia pridáva na vytvorené vizualizácie vodoznak.

### 2.3.3 Jednoduché online nástroje

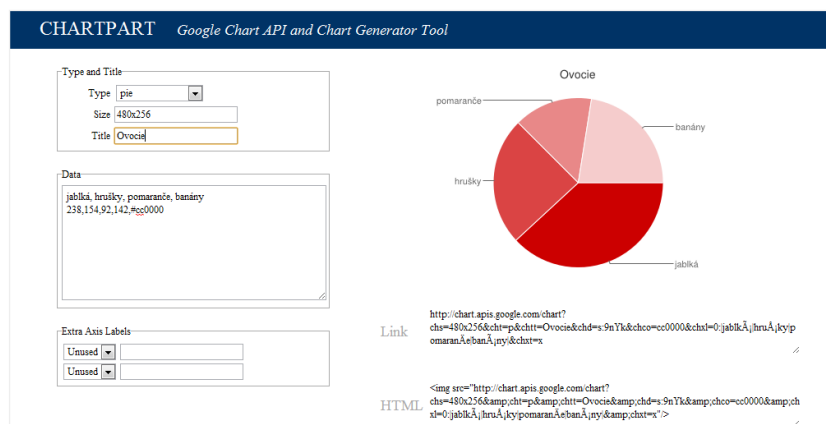
Do skupiny jednoduchých online nástrojov sú zaradené webstránky, ktoré umožňujú tvorbu statických obrázkových grafov, sú však príliš jednoduché na to, aby sa dali zaradiť medzi aplikácie. Ich spoločným znakom je nutnosť manuálneho zadávania dát. Vo väčšine prípadov sa v nich dá vytvoriť len jeden typ grafu, napr. koláčový diagram alebo stĺpcový graf. Všetky weby uvedené v tejto skupine ponúkajú tvorbu vizualizácií bezplatne.

Príkladmi jednoduchých online nástrojov sú **Pie Chart Tool**, **Pie Color**, **Math Warehouse's Chart Maker**, **CSS Chart Generator**, **ChartGo** alebo **OWTChart Generator**.

Špecifickú skupinu tvoria webstránky, ktoré na vykreslenie grafov využívajú Google Chart Tools<sup>2</sup>, ako napr. **Online Chart Generator**, **Online Charts Builder**, **Chart Part**, **Chart Maker** alebo **ChartTool**.

---

<sup>2</sup>Google Chart Tools je javascriptová knižnica pre webových vývojárov, ktorá slúži na vizualizáciu dát na webstránke.



Obr. 2.3: Jednoduchý online nástroj Chart Part

## 2.3.4 Knížnice

Poslednou kategóriou podobných služieb sú knižnice, ktoré však majú inú cieľovú používateľskú skupinu ako softvéry a aplikácie spomenuté v predchádzajúcich častiach (alebo ako samotný *Grafit*). Ich použitie je podmienené aspoň základnou znalosťou programovania, sú teda určené skôr pre vývojárov webových stránok. Väčšina z týchto knižníc beží na javascriptovom rozhraní resp. využíva javascriptovú knižnicu (jQuery, MooTools), napr. **Highcharts**, **JS Charts**, **D3.js**, **MilkChart**, **TufteGraph**, **amCharts** alebo už spomínaná knižnica **Google Chart Tools**. Ďalšie na vykreslenie grafov využívajú Flash (**FusionCharts**, **FlyCharts**) alebo PHP (**JpGraph**).

# Kapitola 3

## Návrh riešenia

Na nasledujúcich stranách opíšeme základnú koncepciu aplikácie *Grafit*.  
Návrh sa skladá z troch častí:

- serverová (funkčná) časť
- používateľská časť
- administrácia

### 3.1 Serverová časť

#### 3.1.1 Databáza

Všetky údaje o registrovaných používateľoch, nahratých dátach a vytvorených grafoch, ktoré aplikácia potrebuje, sa ukladajú do databázy. Kvôli ľahšej manipulácii sa zapisujú aj základné informácie o grafových typoch. Schéma databázy je v prílohe na strane 41.



### Tabuľka používateľov („users“)

Táto tabuľka obsahuje informácie o zaregistrovaných používateľoch. Pri registrácii používateľ zadá *e-mail* a *heslo*, pri zápise do databázovej tabuľky sa mu automaticky priradí číselný identifikátor používateľa *uID*, pomocou ktorého sa v databáze identifikujú dáta a grafy, ktoré používateľ nahral; *uID* je zároveň primárnym kľúčom tejto tabuľky. Do posledného stĺpca sa ukladá informácia o *právach*, ktorá má dve možné hodnoty: 0 (používateľ) alebo 1 (administrátor).

### Tabuľka nahraných dát („data“)

Ak chce zaregistrovaný používateľ vytvoriť vizualizáciu, musí najprv pridať dáta. Informácie o nich sa pridávajú do tabuľky dát. Týmito informáciami sú číselný identifikátor dát *dID* (primárny kľúč tabuľky), *uID*, *názov* dát, samotné *dáta* a *frekvencia aktualizácie*, teda informácia o tom, ako často sa dáta budú aktualizovať. Aktualizácia potom môže prebiehať

- (i) jednorázovo
- (ii) manuálne
- (iii) periodicky s nastaveným intervalom
- (iv) pri každom načítaní grafu, asociovaného s danými dátami

Používateľ môže dáta načítať buď zo súboru na svojom disku (i) alebo dynamicky z URL (ii - iv). Pri dynamickom načítavaní sa do databázy ukladá aj *URL* dátového súboru.

Posledným stĺpcom v tejto tabuľke je stĺpec *is\_deleted*, ktorý môže mať hodnotu 0 (dáta sú aktívne) alebo 1 (dáta sú neaktívne). Predvolená hodnota je 0; pri odstránení používateľom sa dáta hneď nevymažú, iba sa táto hodnota zmení na 1. Používateľ tým získava možnosť obnoviť odstránené dáta.

### **Tabuľka periodicky aktualizovaných dát („refresh“)**

V prípade, že sa dáta načítavajú periodicky sa navyše pridáva nový záznam aj do tabuľky periodicky aktualizovaných dát. Záznam obsahuje *dID*, *interval aktualizácie*, *čas poslednej* a *čas nasledujúcej aktualizácie* dát.

### **Tabuľka typov vizualizácií („types“)**

Existujúce typy sa ukladajú do tabuľky typov vizualizácií. Ukladanie sa realizuje cez administráciu aplikácie, opísané je v časti 3.3. Pre každý typ sa ukladá číselný identifikátor typu *tID* (primárny kľúč tabuľky) a *názov*.

### **Tabuľka vytvorených vizualizácií („graphs“)**

Po načítaní má používateľ možnosť vytvoriť z dát vizualizáciu. Pri uložení do databázy sa k nej priradí primárny kľúč *gID* a ukladajú sa ďalšie identifikátory, ktoré danú vizualizáciu prepoja s tabuľkami používateľov (*uID*), dát (*dID*) a typov (*tID*). Databázový záznam ďalej obsahuje *meno*, *parametre*, *rozmery* a *práva* (súkromný, na heslo, verejný) grafu, prípadne aj *heslo*.

## Stĺpce *tag* a *error*

Všetky tabuľky obsaujú navyše ďalší stĺpec: *error* pri tabuľke periodicky aktualizovaných dát, *tag* pri ostatných tabuľkách. Slúži na zaznamenávanie detailov interakcií aplikácie s databázovým záznamom, či už pre potreby samotnej aplikácie alebo používateľa. Takýmito detailami sú napr. čas poslednej aktualizácie dát, chyba pri načítavaní dát z URL, čas vymazania dát a podobne.

## Databázové väzby

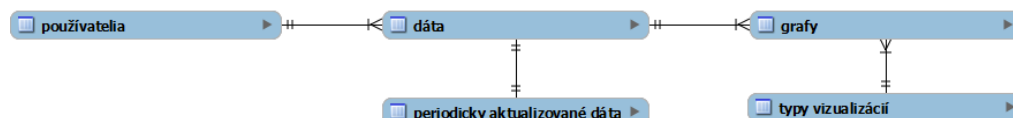
Jednotlivé databázové tabuľky sú navzájom prepojené cez tabuľkové indexy. Všetky väzby môžeme charakterizovať ako identifikačné (*identifying relationship*), čo znamená, že záznam v závislej tabuľke musí byť naviazaný na záznam v materskej tabuľke.

materská tabuľka	závislá tabuľka	typ väzby	index(y)
používatelia	dáta	1:N	uID
dáta	periodicky akt. dáta	1:1	dID
dáta	grafy	1:N	uID, dID
typy vizualizácií	grafy	1:N	tID

**Tabuľka 3.1:** *Prehľad väzieb v databáze (pozri aj diagram 3.1)*

Väzba **1:N** v prípade *používatelia*  $\rightarrow$  *dáta* znamená, že jeden používateľ môže nahrať viac dát, no ľubovoľné dáta patria práve k jednému používateľovi; analogicky to funguje aj pri ostatných väzbách tohto typu. Väzba **1:1**

pri tabuľkách *dáta*  $\rightarrow$  *periodicky aktualizované dáta* znamená, že v prípade, že sú dáta aktualizované periodicky, druhá menovaná tabuľka obsahuje na ne naviazaný záznam, ktorý sa už na iné dáta neviaže.



Obr. 3.1: Diagram väzieb v databáze

### 3.1.2 Obsluha databázy

Skripty obsluhujúce databázu sú súčasťou tried, ktoré v aplikácii reprezentujú tri hlavné typy objektov - používateľov, dáta a grafy. S databázou ďalej pracujú aj časti aplikácie, ktoré nie sú súčasťou žiadnej z týchto tried - napr. funkcie, starajúce sa o zapisovanie záznamov do tabuľky typov vizualizácií. Tieto sa spúšťajú v administrácii aplikácie (časť 3.3).

#### Trieda Používateľ

Trieda Používateľ má na starosti akcie súvisiace s používateľským kontom, ako napr. registrácia takéhoto konta, kontrola existencie, kontrola prihlasovacích údajov a prihlásenie, zmena e-mailu alebo hesla, odstránenie konta. Zabezpečuje tiež výstup zoznamu všetkých dát, ktoré konkrétny používateľ pridal, a grafov, ktoré vytvoril.

## Trieda Dáta

Trieda Dáta sa stará o databázové záznamy v tabuľke dát - o ich pridávanie, zmenu nastavení (názov, frekvencia aktualizácie, URL súboru), získavanie dát zo súborov alebo z URL, vymazanie dát používateľom a prípadné obnovenie po vymazaní. Obsahuje aj funkciu na výstup náhľadu dát vo forme tabuľky (obr. 3.2).

	A	B	C	D	E	F	G
1	40 184	Quebec	Jones	Pencil	95	1.99	189.05
2	40 201	Ontario	Kivell	Binder	50	19.99	999.5
3	40 218	Ontario	Jardine	Pencil	36	4.99	179.64
4	40 235	Ontario	Gill	Pen	27	19.99	539.73
5	40 252	Alberta	Sorvino	Pencil	56	2.99	167.44
6	40 269	Quebec	Jones	Binder	60	4.99	299.4
7	40 286	Ontario	Andrews	Pencil	75	1.99	149.25
8	40 303	Ontario	Jardine	Pencil	90	4.99	449.1
9	40 320	Alberta	Thompson	Pencil	32	1.99	63.68
10	40 337	Quebec	Jones	Binder	60	8.99	539.4
11	40 354	Ontario	Morgan	Pencil	90	4.99	449.1
12	40 371	Quebec	Howard	Binder	70	1.99	57.71

**Obr. 3.2:** Náhľad dát vygenerovaný triedou Dáta

Objekt triedy Dáta je možné inicializovať cez jednu z dvoch statických funkcií, podľa toho, či pridávame do databázy nové dáta alebo načítavame existujúce (pomocou *dID*).

Poslednou úlohou tejto triedy je vkladanie, aktualizácia a mazanie riadkov tabuľky periodicky aktualizovaných dát. Vkladanie do nej prebieha pri ich vytvorení alebo pri zmene iného typu na takéto dáta; záznam v nej sa aktualizuje pri automatickom načítaní dát z URL; vymazanie riadku sa deje pri vymazaní periodicky aktualizovaných dát alebo pri zmene typu takýchto dát na iný.

## **Trieda Graf**

Trieda Graf zabezpečuje komunikáciu aplikácie s tabuľkou grafov, t.j. vytváranie nových grafov, zmenu vlastností konkrétneho grafu (názov, rozmery, práva,...), aktualizáciu parametrov, zmazanie grafu. Zároveň je základom pre podtriedy, reprezentujúce typy vizualizácií.

Pri načítaní grafu trieda skontroluje jeho práva, podľa ktorých zobrazí príslušný výstup.

### **3.1.3 Definícia typov vizualizácií**

#### **Typ ako podtrieda**

Jedným z cieľov pri vytvorení aplikácie *Grafit* bola možnosť jej ľahkej rozšíriteľnosti o nové techniky vizualizácie. Aby sme to dosiahli, bolo potrebné vymyslieť spôsob akým sa budú jednotlivé typy vizualizácií definovať bez nutnosti zásahu do jadra samotnej aplikácie.

Riešením je definícia typu ako samostatnej triedy, ktorá bude potomkom (podtriedou) triedy Graf. Pre správne fungovanie a vykresľovanie potom stačí, ak takáto podtrieda obsahuje

1. **zoznam parametrov** grafu (spolu s predvolenými hodnotami) a
2. **funkciu na vykresľovanie** vizualizácie.

Pomocou parametrov si používateľ dokáže prispôbiť výsledný vzhľad vizualizácie. Trieda Graf je navrhnutá tak, aby jednotlivé typy mohli obsahovať ľubovoľný počet nastaviteľných vlastností. V podtriede sú parametre rozde-

lené do kategórií (farba, font, dátový stĺpec, textový reťazec,...) a v aplikácii sú reprezentované ako dvojrozmerné asociatívne pole (zdrojový kód 3.1).

Podtrieda môže predefinovať aj ďalšie časti triedy Graf. Ide najmä o funkcie na náhľad a export grafu, ktoré sú štandardne určené pre typy so statickým obrázkom na výstupe, alebo funkciu, ktorá sa spúšťa pri zmene dát, asociovaných s daným grafom. Tú je potrebné zmeniť napríklad vtedy, ak zmena zdrojových dát spôsobí zmenu počtu parametrov grafu.

```
1  class PieGraph extends Graph
2  {
3      public $param = array(
4          'data' => array( // dátové stĺpce
5              'items' => 0, // popisy
6              'values' => 1, // hodnoty
7          ),
8          'color' => array( // farby
9              'text' => '#333333', // text
10             'back' => '#ffffff' // pozadie
11         ),
12         'color_scheme' => array(
13             // farebná schéma - mení sa dynamicky podľa dát
14         ),
15         'options' => array( // nastavenia
16             'show_title' => 1, // ukázať nadpis?
17         ),
18         'font' => array( // nastavenia písma
19             'family' => "Calibri", // font
20             'legend' => "10", // veľkosť písma v legende
21             'title_font' => "20" // veľkosť písma nadpisu
22         ),
23     );
24     :
25 }
```

**Zdrojový kód 3.1:** *Príklad definície parametrov typu vizualizácie (PHP)*

## Ďalšie súbory

Typ vizualizácie tvoria okrem súboru s podtriedou opísanou v predošlej časti aj ďalšie dva súbory. **Konfiguračný súbor** (formát INI) obsahuje názov typu a označenie parametrov, ktoré uvidí používateľ pri ich editácii. **Statický náhľad** daného typu sa zobrazuje pri vytvorení nového grafu (časť 3.2.3).

```
1 ; konfiguračný súbor ku PieGraph.php
2 [meta]
3 name          = "Koláčový diagram"
4
5 [param]
6 data          = "Výber dátových stĺpcov"
7 color         = "Farby"
8 options       = "Možnosti"
9 font          = "Písmo"
10 color_scheme = "Farebná schéma"
11
12 [data]
13 items         = "položky (nečíselný stĺpec)"
14 values        = "číselné hodnoty (číselný stĺpec)"
15
16 [color]
17 text          = "texty"
18 back          = "pozadie"
19
20 [options]
21 show_title    = "zobraziť nadpis"
22
23 [font]
24 family        = "font"
25 legend        = "legenda"
26 title_font    = "veľkosť písma - nadpis"
```

**Zdrojový kód 3.2:** *Príklad konfiguračného súboru*



### 3.1.4 Generovanie grafu

Po vytvorení grafu aplikácia poskytne používateľovi URL adresu vizualizácie (napr. `grafit.tiborstanko.sk/graf-4.png`), pomocou ktorej si vizualizáciu môže vložiť na svoj web - ako obrázok, alebo - v prípade interaktívnej vizualizácie - ako objekt.

Keď prehliadač načíta vizualizáciu cez jej URL adresu, spustí sa na serveri skript, ktorý cez ID grafu zistí jeho typ a vytvorí objekt príslušnej podtriedy. Následne sa zavolá funkcia na „kreslenie“ grafu, ktorá podľa parametrov nastavených používateľom dodá výstup - obrázok alebo objekt.

## 3.2 Používateľská časť

### 3.2.1 Registrácia a prihlásenie

Prvým kontaktom používateľa s aplikáciou je úvodná stránka s prihlasovacím formulárom. Tu má používateľ dve možnosti. Ak už má v *Grafite* vytvorené konto, môže sa pomocou neho prihlásiť; ak ho nemá, pokračuje na registračnú stránku, kde si ho vytvorí. Po prihlásení sa zobrazí domovská stránka, na ktorej si používateľ vyberie svoju ďalšiu akciu.

### 3.2.2 Import a správa dát

Po vybraní podstránky s dátami sa používateľovi zobrazí zoznam všetkých dát, ktoré nahral, zoradených podľa poradia, v ktorom boli pridané. V zozname je okrem názvu dát aj frekvencia ich aktualizácie a čas posledného načítania dátového súboru. Používateľ má následne možnosť vybrať existu-

júce alebo pridať nové dáta. Po kliknutí na existujúce dáta je možné upraviť ich vlastnosti, manuálne ich aktualizovať alebo úplne vymazať. Zároveň sa zobrazí náhľad dát v podobe tabuľky.

### **3.2.3 Výber typu a vytvorenie grafu**

Pridané dáta sú pripravené na vizualizovanie. Pri vytvorení nového grafu používateľ vyplní jeho názov a rozmery v pixeloch - rozmery musia byť z intervalu [250; 1000]. Vyberie vizualizované dáta a typ grafu, nastaví práva a prípadné heslo. V spodnej časti stránky sa zobrazí náhľad vybraného typu v podobe statického obrázka - pri zmene vybraného typu sa automaticky zmení aj obrázok náhľadu.

### **3.2.4 Nastavenie a export grafu**

Po kliknutí na tlačidlo "Vytvoriť graf" je používateľ presmerovaný na podstránku s novovytvorenou vizualizáciou, kde môže upraviť jej vlastnosti (zadané pri vytvorení) a automaticky vygenerované parametre. Ponúkne sa mu náhľad a možnosti na export grafu - URL, HTML kód na vloženie do web stránky a pod. Typ exportu závisí od triedy typu grafu.

### **3.2.5 Správa grafov**

Podstránka so správou grafov je veľmi podobná správe dát. Používateľovi sa zobrazí zoznam všetkých doteraz vytvorených vizualizácií spolu s ich rozmermi, typom, právami a odkazom na prislúchajúce dáta.

### 3.2.6 Nastavenia účtu a vymazanie konta

Na poslednej podstránke má používateľ možnosť zmeniť nastavenia svojho konta, t.j. e-mail a heslo. Po zadaní aktuálneho hesla môže tiež vymazať svoje konto spolu so všetkými dátami a grafmi.

## 3.3 Administrácia aplikácie

Administračná časť aplikácie *Graft* je podobne ako používateľská časť dostupná priamo v prehliadači. Prístup do nej však majú len administrátori, teda používatelia, ktorí majú v databáze práva nastavené na hodnotu 1. Vďaka jednoduchosti aplikácie je administrácia tvorená iba jednou podstránkou, ktorá je rozdelená do dvoch častí.

**Správa typov vizualizácií** automaticky skenuje priečinok **factory** (nachádza sa v koreňovom adresári aplikácie), ktorý obsahuje definície typov. Jeho podpriečinky - konkrétne typy - potom zobrazí v tabuľke spolu s informáciou, či je daný typ aktívny (zapísaný ako záznam v tabuľke typov). Ak nie, administrátor ho môže aktivovať, čím sa stáva prístupným pre bežných používateľov.

**Správa používateľov** zobrazí zoznam všetkých zaregistrovaných účtov. Pri bežných používateľoch má administrátor možnosť vymazať ich konto, alebo z nich urobiť ďalšieho administrátora.

# Kapitola 4

## Implementácia

V tejto kapitole nájdete informácie o technológiách, použitých pri implementácii *Grafitu*. Všetky uvedené technológie sú open-source, ich zoznam je v prílohe na strane 42.

### 4.1 Serverová časť

#### 4.1.1 Jazyk PHP

Skriptovanie na strane servera (*server-side scripting*) zabezpečuje objektovo-orientovaný jazyk **PHP**,<sup>1</sup> vďaka svojej jednoduchosti a flexibilitě zrejme najrozšírenejší jazyk na tvorbu dynamických webov. Pre nás je dôležitá najmä jeho podpora objektov (tried) a zabudovaná funkcionálna na ovládanie databázy. *Grafit* beží na PHP 5.3.

---

<sup>1</sup>**PHP** pôvodne znamenalo Personal Home Page; dnes sa chápe ako rekurzívny akronym znamenajúci **PHP**: **H**ypertext **P**reprocessor.

### 4.1.2 MySQL a prepojenie s PHP

Databázovú vrstvu aplikácie tvorí relačný databázový server **MySQL** (verzia 5.1), bežiaci na populárnej technológii SQL.<sup>2</sup>

Na pripojenie k databáze využíva *Grafit* zabudované PHP rozšírenie **mysqli** (MySQL improved). Oproti štandardnej funkcionalite má toto rozšírenie niekoľko výhod, napr. objektovo-orientované rozhranie a podporu *prepared statements*, ktoré poskytujú lepšie zabezpečenie proti SQL injection.<sup>3</sup>

### 4.1.3 Spracovávanie dát

*Grafit* je schopný spracovať a uložiť dáta zo súborov rozličných typov. Využíva na to dve PHP knižnice. Spracovávanie formátov TXT, CSV a ODS (OpenDocument Spreadsheet) zabezpečuje **php-SpreadsheetReader**, excelovské súbory (XLS, XLSX) má na starosti **PHPExcel**.

### 4.1.4 .htaccess

Súbor **.htaccess** (*hypertext access*) je rozšírením konfiguračného súboru serveru, funguje na úrovni adresárov. *Grafit* ho využíva na

- blokovanie priameho prístupu do niektorých podadresárov
- presmerovanie www verzie na non-www
- prispôsobenie chybových reakcií (HTTP 404)
- vytváranie „pekných“ URL

---

<sup>2</sup>**SQL** = **S**tructured **Q**uery **L**anguage - štruktúrovaný vyhľadávací jazyk

<sup>3</sup>technika útoku na databázu prostredníctvom webstránky

## 4.2 Klientská časť

### 4.2.1 XHTML a CSS

Prezentačnú vrstvu aplikácie tvorí kombinácia štandardov **XHTML 1.1**<sup>4</sup> a **CSS 2.1**.<sup>5</sup> *Grafit* tiež využíva niektoré časti funkcionality **CSS 3**: gradienty ako pozadie (*background: linear-gradient*), priehľadnosť (*opacity*), tieň (*text-shadow*, *box-shadow*) a oblé rohy (*border-radius*).



**Obr. 4.1:** *Príklad použitia CSS 3 v aplikácii Grafit - tlačidlo vo formulári. Pri definovaní štýlu sme použili oblé rohy, lineárny gradient na pozadí, tieň textu a tieň boxu (zdrojový kód 4.1).*

V dobe písania tejto práce je väčšina z asi päťdesiat modulov špecifikácie CSS 3 v štádiu rozpracovania, čo znamená, že W3C<sup>6</sup> im zatiaľ nepridelilo štatút „odporúčanie“ (*recommendation*).

Napriek tomu väčšina moderných prehliadačov podporuje CSS 3 funkcionality. Množstvo ľudí však stále používa staršie verzie prehliadačov (najmä Internet Explorer 6 až 8), ktoré ju podporujú slabo alebo vôbec.

Tento problém rieši utilita **CSS3 PIE**,<sup>7</sup> ktorá pridáva podporu (pre nás podstatných) častí CSS 3 funkcionality do Internet Exploreru (od verzie 6). Celú utilitu tvorí iba jeden súbor a implementuje sa na pár riadkoch kódu.

---

<sup>4</sup>**XHTML** = **eXtensible HyperText Markup Language** - rozšíriteľný hypertextový značkový jazyk

<sup>5</sup>**CSS** = **Cascading Style Sheets** - kaskádové štýly

<sup>6</sup>**W3C** = **World Wide Web Consortium**

<sup>7</sup>**PIE** v tomto prípade znamená **P**rogressive **I**nternet **E**xplorer

```

1 input[type="submit"] {
2     /* pre prehliadače, ktoré nepodporujú CSS 3 */
3     background: #6fbb69;
4     /* podpora CSS 3: Chrome, Safari, Firefox, Opera, IE9 */
5     background: -webkit-gradient(linear,0 0,0 top,from(#6fbb69)
6         ,to(#38a92f));
7     background: -webkit-linear-gradient(#6fbb69, #38a92f);
8     background: -moz-linear-gradient(#6fbb69, #38a92f);
9     background: -ms-linear-gradient(#6fbb69, #38a92f);
10    background: -o-linear-gradient(#6fbb69, #38a92f);
11    background: linear-gradient(#6fbb69, #38a92f);
12    /* PIE pozadie */
13    -pie-background: linear-gradient(#6fbb69, #38a92f);
14    /* tieň boxu */
15    -webkit-box-shadow: #eee 0 0 5px;
16    -moz-box-shadow: #eee 0 0 5px;
17    box-shadow: #eee 0 0 5px;
18    /* tieň textu */
19    text-shadow: 1px 1px 0 #555;
20    /* použije sa PIE .htc súbor */
21    behavior: url("/PIE.htc");
22    :
23 }

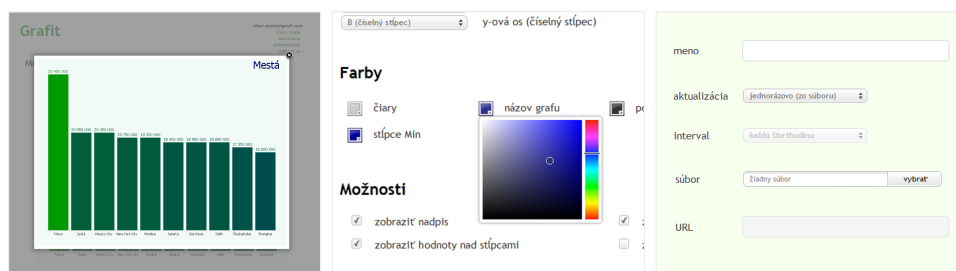
```

**Zdrojový kód 4.1:** Použitie CSS 3 pre dosiahnutie efektu na obrázku 4.1

## 4.2.2 JavaScript, jQuery, Ajax

Skriptovanie na strane klienta (*client-side scripting*) má na starosti programovací jazyk **JavaScript** a jeho rozšírenia, najmä populárna knižnica **jQuery**.<sup>8</sup>

Pri načítavaní statických náhľadov vizualizácií alebo dát využíva *Grafit* technológiu **Ajax**,<sup>9</sup> ktorá zabezpečuje dynamické načítavanie obsahu bez potreby obnovenia webstránky (zdrojový kód 4.2). Kvôli validácii textových polí formulárov v reálnom čase je v *Grafite* implemetovaný widget od Adobe (*Validation Text Field widget*), napísaný v ajaxovom frameworku Spry.<sup>10</sup>



(a) Fancybox

(b) miniColors

(c) Uniform.js

**Obr. 4.2:** Vizualne vylepšenia cez jQuery

Vizuálny dojem z aplikácie vylepšujú jQuery skripty **Fancybox** (zväčšené náhľady obrázkov a dát), **miniColors** (výber farby) a **Uniform.js** (krajšie HTML formuláre).

<sup>8</sup>jQuery v súčasnosti využíva viac ako polovica z top 10 000 najnavštevovanejších webov.

<sup>9</sup>**Ajax** = **A**ynchronous **J**avaScript and **X**ML

<sup>10</sup>[labs.adobe.com/technologies/spry/](http://labs.adobe.com/technologies/spry/)



```

1  $("form#create-new-graph select#select-type").change( function() {
2      // zobrazí sa animovaný GIF obrázok (prebieha načítavanie)
3      $("#type-preview .ajax-loader").show();
4      // uloženie ID vybraného typu
5      var selectedType = $(this).val();
6      // odošle sa Ajax požiadavka:
7      $.ajax({
8          url: "/ajax/get-type-preview.php", // (1) súbor, ktorý sa spustí
9          data: {tID: selectedType},          // (2) ID typu ako parameter
10         type: "post",                      // (3) použije sa HTTP metóda POST
11         success: function( url ){          // (4) po úspešnom načítaní...
12             // ...sa zmení pozadie...
13             $("#type-preview").css( "background", "url(" + url + ")" );
14             // ...a animovaný GIF obrázok sa schová (načítavanie skončené).
15             $("#type-preview .ajax-loader").hide();
16         }
17     });
18 });

```

**Zdrojový kód 4.2:** Použitie Ajaxu (ako funkcie v jQuery) na načítanie statického náhľadu grafu

### 4.2.3 Techniky vizualizácie a HTML5

Na ukážku funkčnosti sú v aplikácii implemetované štyri techniky vizualizácie - dve z nich (stĺpcový graf, scatterplot) majú na výstupe statický PNG obrázok, ďalšie dve (koláčový diagram, čiarový graf) sú interaktívne s využitím technológie **HTML5**, konkrétne elementu `<canvas>`.

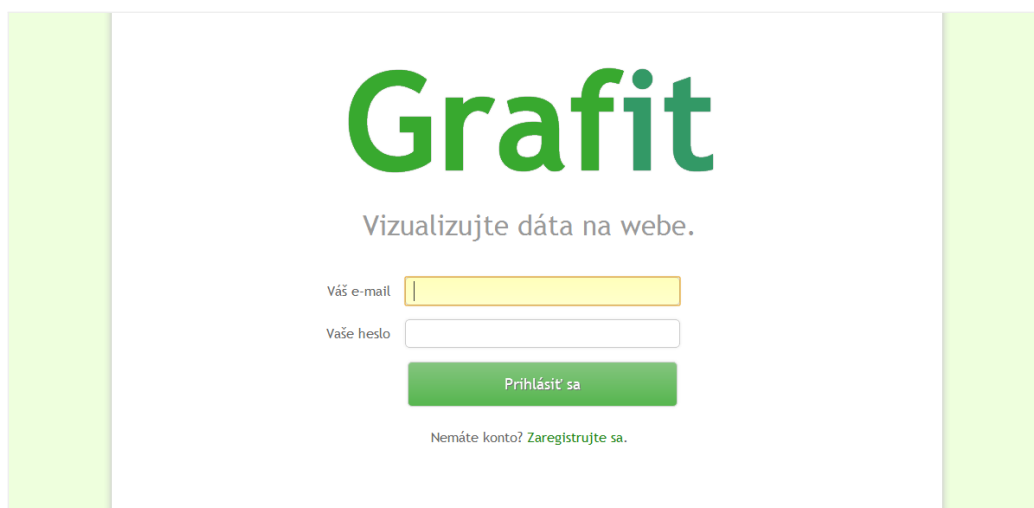
Tento element dokáže s pomocou JavaScriptu dynamicky renderovať rastrový grafický výstup. S jeho podporou je to podobné ako s CSS 3 - canvas podporujú všetky populárne moderné prehliadače: IE 9.0+, Firefox 3.0+, Safari 3.0+, Chrome 3.0+, Opera 10.0+. [Pil]

# Kapitola 5

## Výsledky

V tejto kapitole prezentujeme finálnu podobu prostredia aplikácie. Na príkladoch vytvorených grafov demonštrujeme čo dokáže.

*Grafit* je dostupný online na adrese `grafit.tiborstanko.sk`.



Obr. 5.1: Úvodná stránka

**Dáta**
pridať nové dáta

#	názov	aktualizácia prebieha	posledná aktualizácia
1	Teplota	automaticky, časový interval (URL)	2012-05-24 14:30:05
2	Kontinenty	automaticky, časový interval (URL)	2012-05-24 14:30:05
3	Scatter	automaticky, pri každom načítaní grafu (URL)	2012-05-22 22:54:45
4	Sample data	manuálne (URL)	2012-05-24 10:54:00
5	SK obyvatelia	manuálne (URL)	2012-05-24 11:12:06

Obr. 5.2: Správa dát

**Pridanie dát**

meno

Nové dáta

súbor

žiadny súbor

vybrať

aktualizácia

automaticky, časový interv...

interval

každú štvrťhodinu
každú štvrťhodinu
každú polhodinu
každú hodinu
každý deň
raz za týždeň
raz za mesiac

URL

http://tiborstanko.sk/data/SampleDz

Pridať dáta z URL

Obr. 5.3: Pridanie nových dát

**Grafy**
vytvoriť nový graf

#	názov	rozmery	typ	práva	dáta
1	Kontinenty	800 x 600	Koláčový diagram	súkromný	Kontinenty
2	Mestá	800 x 600	Stĺpcový graf	súkromný	Mestá
3	Teplota	800 x 600	Čiarový graf	súkromný	Teplota
4	Scatterplot	800 x 600	Scatterplot	súkromný	Scatter

Obr. 5.4: Správa grafov

## SK obyvatelia [posledné načítanie dát: 2012-05-24 14:42:16]

meno

súbor

aktualizácia

URL

interval

	A	B
1	Bratislavský kraj	602 436
2	Trnavský kraj	554 741
3	Trenčiansky kraj	594 328
4	Nitriansky kraj	689 867
5	Žilinský kraj	688 851
6	Banskobystrický kraj	660 563
7	Prešovský kraj	814 527
8	Košický kraj	791 723

Obr. 5.5: Detail existujúcich dát

## Vytvorenie grafu

meno

šírka

dáta

výška

typ grafu

rozmery sú v pixeloch a z intervalu [250,1000]

práva

heslo

Obr. 5.6: Vytvorenie nového grafu

Kontinenty

[dátá: Kontinenty]

Kontinenty

Podiel (%)	Continent	Area (km²)
29.3 %	Ázia	43 820 000
20.3 %	Afrika	30 370 000
16.4 %	Severná Amerika	24 490 000
11.9 %	Južná Amerika	17 840 000
9.2 %	Antarkída	13 720 000
6.8 %	Európa	10 180 000
6 %	Austrália	9 008 500

URL

http://grafit.tiborstanko.sk/graph-3.html

HTML

<object width="800" height="600" data="http://grafit.tiborstanko.sk/graph-3.html">Kontinenty</object>

Aktualizovať graf

Zmazať graf

Základné nastavenia

meno

Kontinenty

šírka

800

dáta

Kontinenty

typ grafu

Kolačový diagram

práva

súkromný

výška

600

heslo

rozmery sú v pixeloch a z intervalu [250,1000]

Vyber dátových stĺpcov

A (nečíselný stĺpec)

položky (nečíselný stĺpec)

B (číselný stĺpec)

číselné hodnoty (číselný stĺpec)

Farby

texty

pozadie

Farebná schéma

1

2

3

4

5

6

7

Možnosti

zobrazit' nadpis

Písmo

Calibri

font

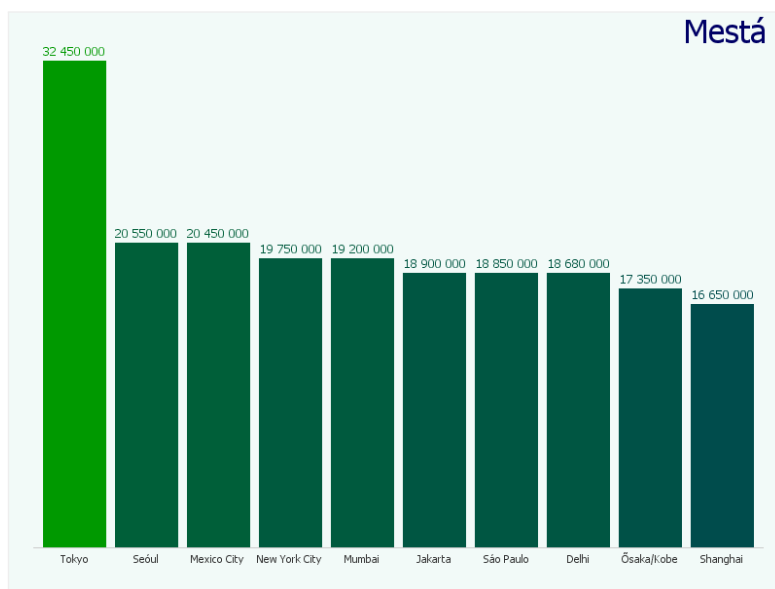
10

legenda

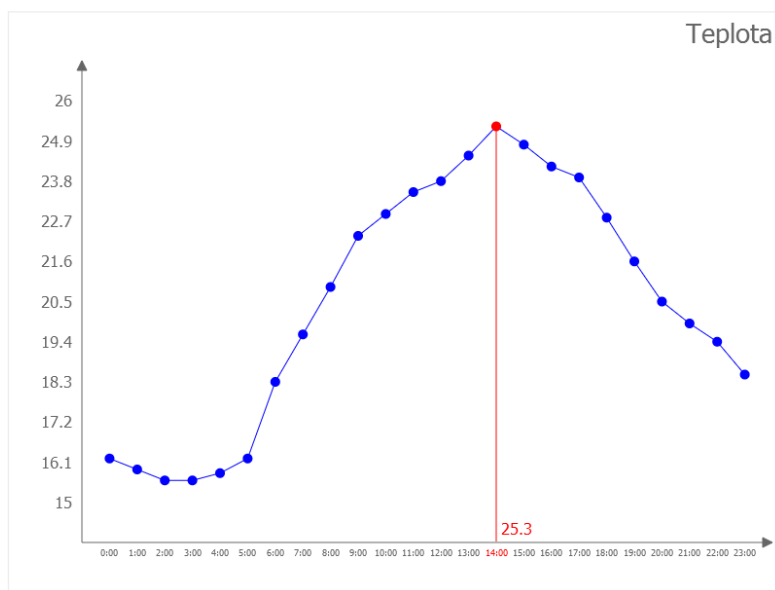
20

veľkosť písma - nadpis

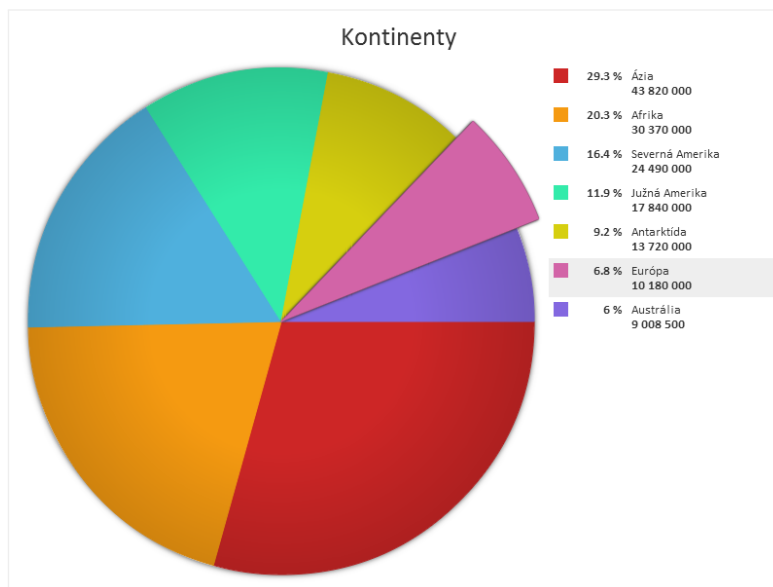
Obr. 5.7: Nastavenie existujúceho grafu



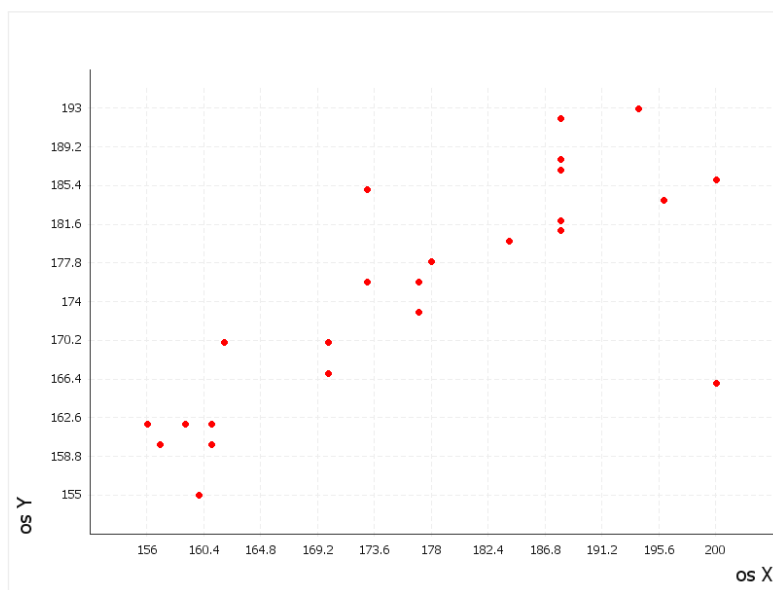
Obr. 5.8: *Stĺpcový graf*



Obr. 5.9: *Čiarový graf (interaktívny)*



**Obr. 5.10:** *Koláčový diagram (interaktívny)*



**Obr. 5.11:** *Relačný diagram (Scatter plot)*

1.

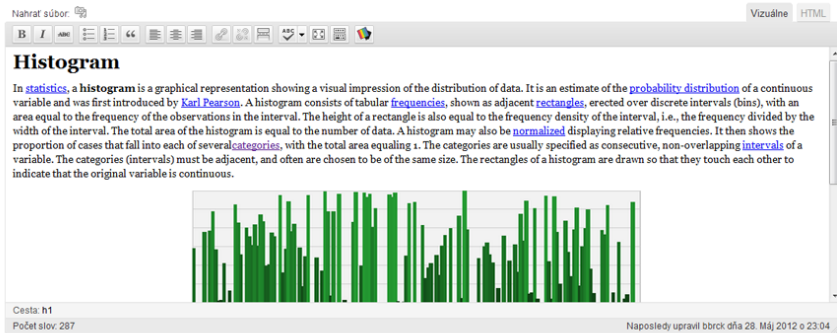
URL

`http://graft.tiborstanko.sk/graph-4.png`

HTML

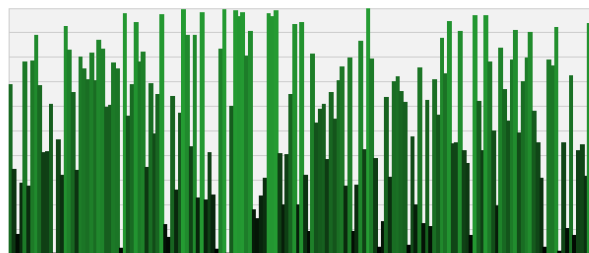
``

2.



### 3. Histogram

In statistics, a **histogram** is a graphical representation showing a visual impression of the distribution of data. It is an estimate of the **probability distribution** of a continuous variable and was first introduced by **Karl Pearson**. A histogram consists of tabular **frequencies**, shown as adjacent **rectangles**, erected over discrete intervals (bins), with an area equal to the frequency of the observations in the interval. The height of a rectangle is also equal to the frequency density of the interval, i.e., the frequency divided by the width of the interval. The total area of the histogram is equal to the number of data. A histogram may also be **normalized** displaying relative frequencies. It then shows the proportion of cases that fall into each of several **categories**, with the total area equaling 1. The categories are usually specified as consecutive, non-overlapping **intervals** of a variable. The categories (intervals) must be adjacent, and often are chosen to be of the same size. The rectangles of a histogram are drawn so that they touch each other to indicate that the original variable is continuous.



Histograms are used to plot density of data, and often for **density estimation**: estimating the probability density function of the underlying variable. The total area of a histogram used for probability density is always normalized to 1. If the length of the intervals on the x-axis are all 1, then a histogram is identical to a **relative frequency** plot.

An alternative to the histogram is **kernel density estimation**, which uses a **kernel** to smooth samples. This will construct a **smooth** probability density function, which will in general more accurately reflect the underlying variable.

The histogram is one of the **seven basic tools** of quality control.

[en.wikipedia.org/wiki/Histogram]

Obr. 5.12: Vloženie grafu do webstránky



# Kapitola 6

## Záver

V práci sme opísali proces vytvárania webovej aplikácie pre vizualizáciu dát, ktorá dostala názov *Grafit*.

Úspešne sa nám podarilo implementovať požadovanú funkcionality: aplikácia dokáže načítavať dáta z rozličných súborových formátov, podporuje automatické načítavanie z URL adresy, umožňuje editáciu vytvoreného grafu a jeho jednoduché vloženie do webstránky, poskytuje možnosť vytvoriť interaktívne vizualizácie.

Okrem dobre vybudovaného priestoru pre rozširovanie o nové techniky vizualizácie ponúka *Grafit* rozsiahle možnosti na svoje vylepšovanie.

V budúcnosti by preto mohol načítavať dáta vo formátoch XML alebo Google Spreadsheet, umožňovať prihlasovanie s použitím konta z inej služby (OpenID, Google alebo Windows Live) alebo byť prepojený s populárnymi sociálnymi sieťami (Facebook, Twitter) či redakčnými systémami (napr. ako plugin pre Wordpress).

# Literatúra

- [Bea10] Jason Beard. *Princípy krásného webdesignu*. Grada Publishing, 2010.
- [Gil10] W. Jason Gilmore. *Beginning PHP and MySQL: From Novice to Professional (Expert's Voice in Web Development)*. Apress, 2010.
- [Kos99] Jiří Kosek. *PHP - tvorba interaktivních internetových aplikací*. Grada Publishing, 1999.
- [Kru07] Steve Krug. *Webdesign - Nenutíte uživatele přemýšlet! 2. aktualizované vydání*. Computer Press, 2007.
- [Paľ09] Eva Paľovičová. *Vizuálna analýza dát*. [Bakalárska práca] Univerzita Komenského v Bratislave. Fakulta matematiky, fyziky a informatiky. Vedúci: Mgr. Matej Novotný, PhD. Bratislava: UK, 2009.
- [Pil] Mark Pilgrim. *Dive Into HTML5*. [diveintohtml5.info](http://diveintohtml5.info).
- [Tan10] Kay Tan. *22 Useful Online Chart & Graph Generators*, marec 2010. [www.hongkiat.com/blog/22-useful-chart-graph-diagram-generators](http://www.hongkiat.com/blog/22-useful-chart-graph-diagram-generators).
- [W3C] World Wide Web Consortium (W3C). [www.w3.org](http://www.w3.org).

# Prílohy

Súčasťou práce je CD so zdrojovým kódom aplikácie. *Grafit* je dostupný online na adrese `grafit.tiborstanko.sk`.

## Príloha 1: Zoznam podobných služieb

### Desktopové softvéry (2.3.1)

Microsoft Excel: `office.microsoft.com/excel`

OpenOffice Calc: `www.openoffice.org`

Swift Chart: `www.globfx.com/products/swfchart`

Grapher9: `www.goldensoftware.com/products/grapher/grapher.shtml`

SmartDraw: `www.smartdraw.com/specials/chart.htm`

### Online aplikácie (2.3.2)

*Flash*

RichGraphs: `www.richgraphs.com`

iCharts: `www.icharts.net`

Rich Chart Live: `www.richchartlive.com`

### *HTML, Javascript*

Create a Graph: [nces.ed.gov/nceskids/createagraph](http://nces.ed.gov/nceskids/createagraph)

DIY (Do It Yourself): [www.diychart.com](http://www.diychart.com)

Chart Tool: [www.onlinecharttool.com](http://www.onlinecharttool.com)

StatCrunch: [www.statcrunch.com](http://www.statcrunch.com)

ChartGizmo: [chartgizmo.com](http://chartgizmo.com)

### *Java*

Chartle.net: [www.chartle.net](http://www.chartle.net)

## **Jednoduché online nástroje (2.3.3)**

Pie Chart Tool: [www.mathcracker.com/pie\\_chart.php](http://www.mathcracker.com/pie_chart.php)

Pie Color: [piecolor.com](http://piecolor.com)

Math Warehouse's Chart Maker: [chartmaker.mathwarehouse.com](http://chartmaker.mathwarehouse.com)

CSS Chart Generator: [www.artviper.net/css-chart-generator](http://www.artviper.net/css-chart-generator)

ChartGo: [www.chartgo.com](http://www.chartgo.com)

OWTChart Generator: [www.maptools.org/owtchart/chartgen.phtml](http://www.maptools.org/owtchart/chartgen.phtml)

### *Google Chart Tools*

Online Chart Generator: [chartgen.blogspot.com](http://chartgen.blogspot.com)

Online Charts Builder: [charts.hohli.com](http://charts.hohli.com)

Chart Part: [chartpart.com](http://chartpart.com)

Chart Maker: [almaer.com/chartmaker](http://almaer.com/chartmaker)

ChartTool: [imagecharteditor.appspot.com](http://imagecharteditor.appspot.com)

## Knižnice (2.3.4)

### *Javascript*

Highcharts: [www.highcharts.com](http://www.highcharts.com)

JS Charts: [www.jscharts.com](http://www.jscharts.com)

D3 - Data-Driven Documents: [d3js.org](http://d3js.org)

MilkChart: [mootools.net/forge/p/milkchart](http://mootools.net/forge/p/milkchart)

TufteGraph: [xaviershay.github.com/tufte-graph](http://xaviershay.github.com/tufte-graph)

amCharts: [amcharts.com](http://amcharts.com)

Google Chart Tools: [developers.google.com/chart/](http://developers.google.com/chart/)

### *Flash*

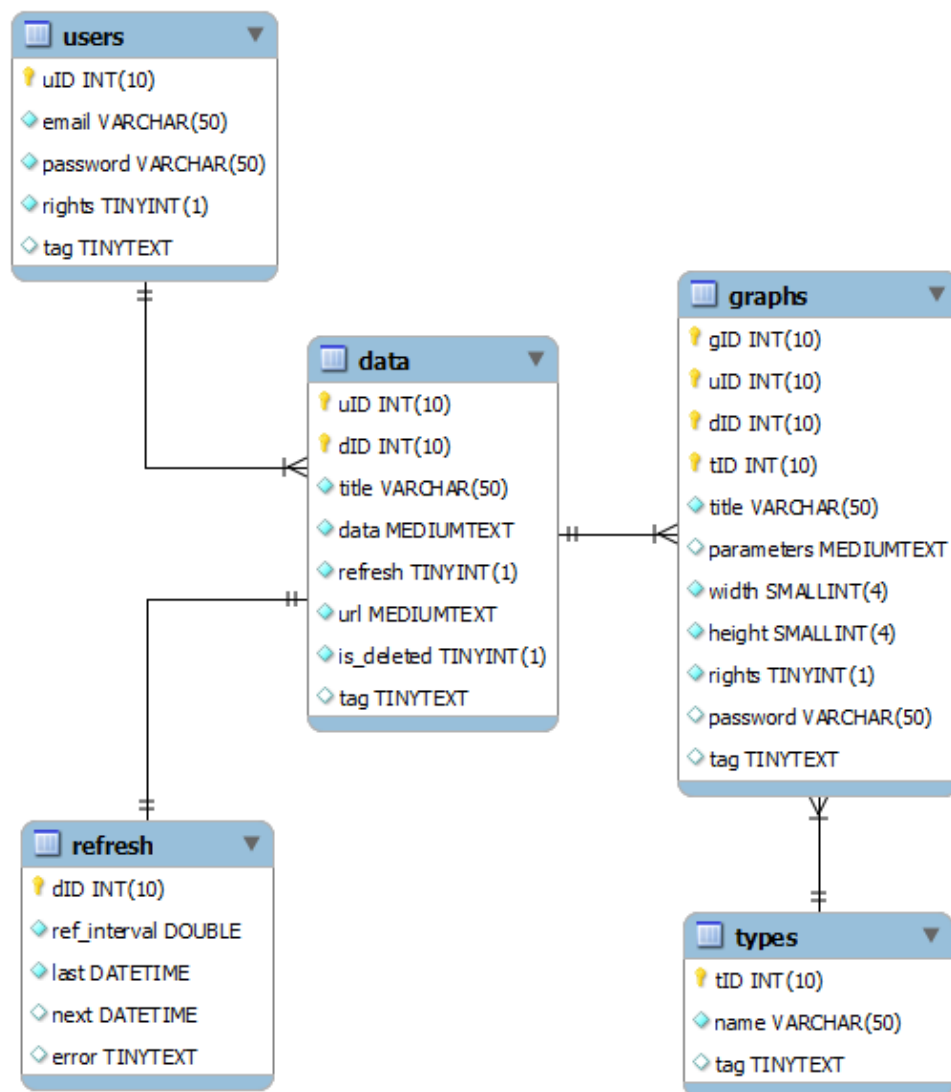
FusionCharts: [www.fusioncharts.com](http://www.fusioncharts.com)

Fly Charts: [www.flycharts.net](http://www.flycharts.net)

### *PHP*

JpGraph: [jpgraph.net](http://jpgraph.net)

## Príloha 2: Schéma databázy



## Príloha 3: Použité technológie

### Serverová časť (4.1)

PHP: [www.php.net](http://www.php.net)

MySQL: [www.mysql.com](http://www.mysql.com)

php-SpreadsheetReader: [code.google.com/p/php-spreadsheetreader/](http://code.google.com/p/php-spreadsheetreader/)

PHPExcel: [phpexcel.codeplex.com](http://phpexcel.codeplex.com)

### Klientská časť (4.2)

XHTML 1.1: [www.w3.org/TR/xhtml11/](http://www.w3.org/TR/xhtml11/)

HTML: [www.w3.org/html/](http://www.w3.org/html/)

CSS: [www.w3.org/Style/CSS/](http://www.w3.org/Style/CSS/)

CSS3pie: [css3pie.com](http://css3pie.com)

jQuery: [jquery.com](http://jquery.com)

Fancybox: [fancybox.net](http://fancybox.net)

miniColors: [www.abeautifulsite.net/blog/?p=1281](http://www.abeautifulsite.net/blog/?p=1281)

Uniform.js: [uniformjs.com](http://uniformjs.com)

Spry Validation Text Field widget:

[labs.adobe.com/technologies/spry/articles/textfield\\_overview/](http://labs.adobe.com/technologies/spry/articles/textfield_overview/)