

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF

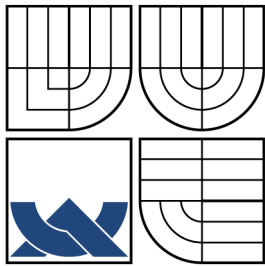
INTERAKTIVNÍ MAPOVÉ PODKLADY V SÍTI INTERNET

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

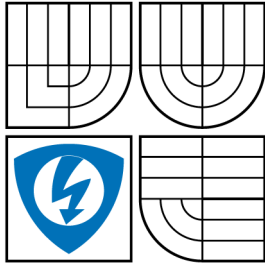
AUTOR PRÁCE
AUTHOR

MAREK ŠPETA

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF

INTERAKTIVNÍ MAPOVÉ PODKLADY V SÍTI INTERNET INTERACTIVE MAP RESOURCES ON THE INTERNET

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

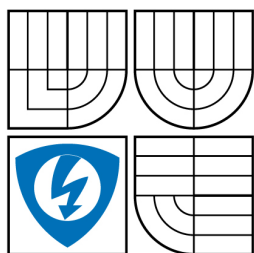
AUTOR PRÁCE
AUTHOR

MAREK ŠPETA

VEDOUCÍ PRÁCE
SUPERVISOR

ING. MARTIN KOUTNÝ

BRNO 2009



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Marek Špeta

ID: 72179

Ročník: 3

Akademický rok: 2008/2009

NÁZEV TÉMATU:

Interaktivní mapové podklady v síti Internet

POKYNY PRO VYPRACOVÁNÍ:

Podrobně se seznámte s možnostmi multiplatformního programování pro Internet. Ve Vámi vybraném vývojovém nástroji navrhnete a realizujete aplikaci pro zobrazování mapových podkladů. V aplikaci zahrňte možnost jednoduchého přidávání komentářů k vybraným místům mapy a tvorby cesty. Ve svém návrhu upřednostněte její možnou implementaci do webového prostředí a zabezpečte, aby bylo možné mapové podklady snáze implementovat.

DOPORUČENÁ LITERATURA:

[1] ZAKHOUR, Sharon, et al. Java 6 : Výukový kurz. 2007. 536 s. ISBN 978-80-251-1575-6.

[2] HEROUT, Pavel. Učebnice jazyka Java. 2006. 352 s. ISBN 80-7232-115-3.

Termín zadání: 9.2.2009

Termín odevzdání: 2.6.2009

Vedoucí práce: Ing. Martin Koutný

prof. Ing. Kamil Vrba, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

ABSTRAKT

Cílem této práce je uvedení přehledu dostupných technologií využívaných pro práci s mapovými podklady v síti Internet. Práce se také zabývá různými formáty vstupních dat a postupy pro získání mapových podkladů. Praktická část práce se soustřeďuje na návrh a realizaci webové aplikace využívající mapové podklady, klade se důraz na modularitu použití a jednoduchost vkládání dat a údajů. V realizaci se využívá skriptovacího jazyka PHP, databáze MySQL a pro doplňkové funkce technologie Ajax a jazyka JavaScript.

KLÍČOVÁ SLOVA

mapy, rastr, vektor, php, mysql, ajax, http, javascript, mapové podklady

ABSTRACT

The objective of this thesis gives an overview of available technologies used to work with map resources on the Internet. The thesis also contains overview of different input data formats and procedures for obtaining maps. Practical part is focused on designing and implementing web applications using map resources, puts the emphasis on the modularity and simplicity of data entry and data. The implementation uses the PHP scripting language, MySQL database and the additional functions uses Ajax technology and JavaScript language.

KEYWORDS

maps, raster, vector, php, mysql, ajax, http, javascript, map resources

ŠPETA M. *Interaktivní mapové podklady v síti Internet*. Brno: VUT v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav Telekomunikací, 2009. 53 s. Vedoucí bakalářské práce Ing. Martin Koutný.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Interaktivní mapové podklady v síti Internet“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce Ing. Martinu Koutnému za užitečnou odbornou, metodickou a organizační pomoc a cenné rady při zpracování této bakalářské práce.

Děkuji také Bc. Vladimíru Smitkovi za užitečné rozšíření povědomí v oblasti programování a jeho ochotu diskutovat nad technickými aspekty realizace této bakalářské práce.

V Brně dne

.....

(podpis autora)

OBSAH

Úvod	10
1 Elektronické Mapy	11
1.1 Typy elektronických dat	11
1.1.1 Rastrový datatyp	11
1.1.2 Vektorový datatyp	11
1.1.3 Výhody a nevýhody datatypů	12
1.2 Digitalizace map	13
1.3 Formáty digitálních map	14
1.3.1 ShapeFile	15
2 Interaktivní mapy na internetu	16
2.1 Milníky v rozvoji webových map	16
2.2 Technologie	17
2.3 Základní funkce	19
2.4 Klasifikace webových map	19
3 Návrh aplikace	21
3.1 Vstupní data	21
3.2 Vytvoření databáze	21
3.3 Rozvržení aplikace	22
3.4 Základní funkce	23
4 Výběr vhodné platformy	25
5 Pracovní jazyky a prostředky	26
5.1 PHP	26
5.1.1 Stručná historie	26
5.1.2 Použití	26
5.1.3 Syntax	27
5.2 MySQL	27
5.2.1 Stručná historie	28
5.2.2 Základní pojmy	28
5.3 JavaScript	28
5.3.1 Stručná historie	29
5.3.2 Použití	29
5.3.3 Syntax	29
5.4 Ajax	30

5.4.1	Stručná historie	30
5.4.2	Technologie	31
5.4.3	Výhody a nevýhody	31
6	Kyber-maps	32
6.1	Vstupní předpoklady	32
6.2	Vytvoření a návrh databáze	32
6.3	Administrátorský režim	34
6.3.1	Zadávání míst na mapě do databáze	34
6.3.2	Upload archivu s mapou na server	36
6.3.3	Zadávání mapového setu do databáze	37
6.3.4	Kontrola údajů a výpis údajů	37
6.3.5	Zabezpečení	38
6.3.6	Slicing tool	38
6.4	Uživatelský režim	39
6.4.1	Vyhledávání	40
6.4.2	Prohlížení	40
6.4.3	Kontextové tipy	41
7	Závěr	43
	Literatura	45
	Seznam symbolů, veličin a zkratk	48
	Seznam příloh	49
	A Vývojový diagram zobrazení mapy	50
	B Popis souborů na médiu	51
	C Softwarová konfigurace	53

SEZNAM OBRÁZKŮ

1.1	Ukázka rastrového datatypu.	12
1.2	Rozdíly mezi zvětšováním rastrového a vektorového textu. U rastru použito vyhlazování.	13
2.1	Ukázka typického ovládání webových map.	19
3.1	Získání jednotlivých segmentů z mapy.	21
3.2	Vytvoření databáze měst.	22
3.3	Návrh rozložení aplikace.	23
3.4	Princip funkce vyhledávání.	24
6.1	Struktura databázové tabulky „podklady“.	34
6.2	Struktura databázové tabulky „skupiny“.	34
6.3	Zadávání míst na mapě do databáze.	35
6.4	Síťová struktura mapy.	36
6.5	Upload archivu s mapou na server.	37
6.6	Zadávání mapového setu do databáze.	37
6.7	Okno programu Slicing tool.	39
6.8	Logo aplikace Kyber-maps.	40
6.9	Vyhledávání a jeho formulář.	40
6.10	Navigace v mapové aplikaci.	41
6.11	Ukázka kontextového tipu	42

ÚVOD

Náplní této bakalářské práce je seznámit se s dostupnými technologiemi na poli interaktivních map v síti internet.

Slovo interaktivní znamená „umožňující vzájemnou komunikaci“, tj. přímý vstup do činnosti stroje nebo programu. V našem případě to znamená, že uživatel či klient může vyžadovat a vyžaduje, aby aplikace jeho požadavky zpracovala a vykonala to, co si uživatel přál [20]. Tato vzájemná komunikace je jedním ze základních pilířů celosvětové sítě Internet.

Migrace na tuto globální síť se zdá být logickým vyústěním technologického pokroku na konci 20. století. V současnosti se na internetu nachází všemožné zdroje dat a pokud by geografické společnosti nevpuly do proudu ostatních odvětví, které se přesouvaly na síť, byla by to jistě brzda v rozvoji této vědy, která do nástupu digitalizace přešlapovala prakticky na místě.

Současné interaktivní mapové podklady na Internetu využívají multiplatformních vývojových nástrojů. Tomuto trendu napomáhá sílíci komunita uživatelů s počítači běžící na systému Linux a v USA je stále větší počet prodaných počítačů se systémem MacOS, než počítačů na bázi IBM PC s OS Windows. Jedním z cílů této práce je seznámit se s multiplatformními vývojovými nástroji a programovací jazyky, které implementují mapové podklady pro webové mapy.

Součástí práce je také návrh a realizace aplikace, jež poběží vzdáleně v síti Internet a bude využívat některé z technologií, které se běžně využívají. Při návrhu aplikace je důležité zamyslet se nad přínosem konkrétní aplikace pro uživatele, nad jejím ovládním, zda bude dostatečně ergonomické a intuitivní. Neméně důležitou vlastností aplikace by měl být samotný přístup ke zdrojovým datům, jejich jednoduchá změna a přidání.

1 ELEKTRONICKÉ MAPY

Pojem *mapa* fakticky znamená obrazovou reprezentaci určité oblasti. Tvorbou map se zabývá obor kartografie. Již naši dávní předkové tvořili mapy a jejich největší význam se přičítá středověkým objevným výpravám mimo tehdejší známý svět (Evropu). V průběhu let se v kartografii vymezipily různě složité a různě tématické druhy map. Důležitou vlastností každé takovéto mapy je její měřítko, které vyjadřuje poměr dílu velikosti mapy (většinou 1cm) ke skutečné velikosti dané oblasti na povrchu Země.

V poslední čtvrtině 20. století se důležitou pomůckou každého kartografa stal počítač a v dnešní době neexistuje kartografická firma nevyužívající digitálního zpracování dat [2]. V oblasti informačních technologií se také objevil nový pojem GIS (Geografický inf. systém – Geographic information system), což je informační systém, který integruje, uchovává, zpracovává, analyzuje, sdílí a zobrazuje geografická data.

1.1 Typy elektronických dat

V digitalizaci map se uvádějí dvě základní abstrakce reálného světa. Jedná se o diskrétní objekty (například domy) nebo velké pláne. Tyto abstrakce mohou být digitálně zobrazovány rastrovaně či vektorově.

1.1.1 Rastrový datatyp

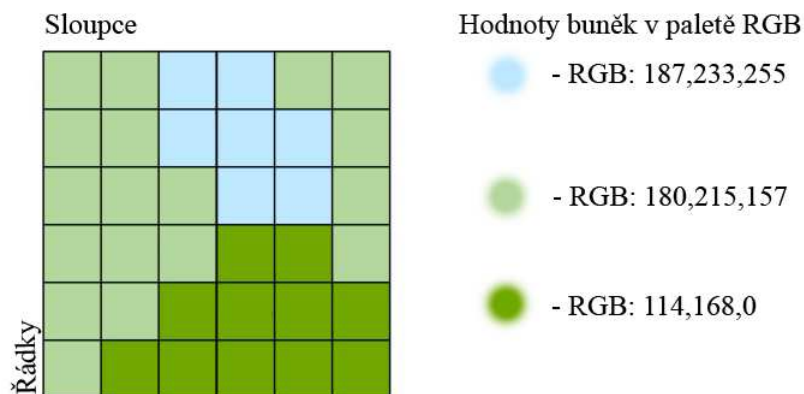
Datový typ rastr je v základě jakýkoliv typ digitálního obrazu, který je reprezentován v mřížce. Nejmenší jednotkou takového obrazu je jeden pixel. Vhodnou kombinací pixelů získáme výsledný obraz, což je velký rozdíl od poměrové vektorové grafiky. Pokud považujeme digitální obraz za reprezentaci reality, tak fotografie či mapa přenesená do počítače, se dá považovat za abstrakci reality.

Rastrový data typ se skládá z řádků a sloupců buněk, přičemž každá buňka má jednu hodnotu, tak jak to vidíme na obr. 1.1. Pokud za zdroj dat považujeme obrázek a za jednu buňku jeho pixel, tak tento pixel má hodnotu odpovídající barvě příslušné barevné palety.

Rastrové data jdou mohou být uloženy v rozličných formátech z nejznámějších můžeme hovořit o TIF, JPEG, BMP, PNG atd.

1.1.2 Vektorový datatyp

V GIS jsou mapy často reprezentovány vektory. Tyto vektory jsou základními prvky geometrických útvarů (tzv. shapes). Různé geografické rysy a objekty jsou pak vyjádřeny různými typy geometrických útvarů [1].



Obr. 1.1: Ukázka rastrového datatypu.

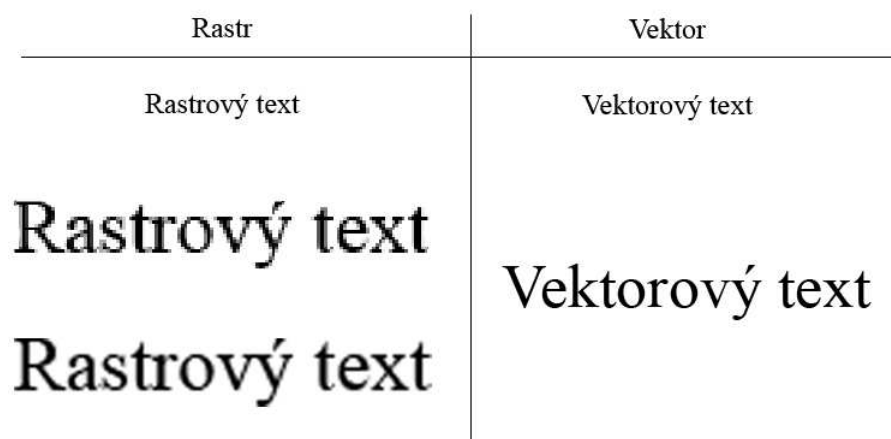
- **Body** – Bezrozměrné body jsou používány pro takové geografické útvary, které lze dobře vyjádřit jednobodovou referencí. Nejčastěji jsou to jednoduché lokace. Například můžeme hovořit o výškových bodech, památkách atp. Body tak na vektorových mapách zprostředkovávají nejmenší možnou množinu informací. Body jsou také vhodné pro použití ve velkých měřítcích, např. na mapě světa je vhodné použít jednoduché body pro zobrazení měst. U zvětšené mapy lze již při určitém měřítku uvažovat o použití polygonu.
- **Křivky** – Jednorozměrné přímky či křivky jsou používány pro vyjádření lineárních útvarů jako jsou, řeky, silnice, koleje a topografické křivky. Opět můžeme při určitém měřítku mapy některé objekty vyjádřit spíše křivkou, než polygonem. Křivky jsou měřitelné a můžeme díky nim měřit vzdálenosti.
- **Mnohoúhelníky (polygony)** – Dvourozměrné polygony jsou používány pro geografické útvary, které pokrývají určitou část zemského povrchu, např. jezera, parky, budovy, městské území atd. Polygony jsou nositeli největšího množství informací a můžeme jim měřit obvod či plochu.

Jakýkoliv vektorový soubor pak obsahuje databázi objektů s vlastními atributy objektů. Například může existovat databáze popisující horu, která obsahuje data o převýšení, zalesnění a zasněžení. Tyto data pak mohou být rozšířena o horské chaty, které se při určitých zvětšeních zobrazují jako body a při dalších jako polygony atp.

1.1.3 Výhody a nevýhody datatypů

Jako všechna hlediska i tato hlediska jsou relativní, vždy záleží na aplikaci a co přesně od dané aplikace chceme.

- Rastrová data povětšinou potřebují více úložného prostoru. Ze své podstaty data pokrývají celou mapu, přičemž u vektorových souboru jsou data jen tam, kde jsou potřeba. Rastrové data jsou až 100x větší než vektorová.
- Rastrová data také povolují jednodušší implementaci překryvných (vrstevných) operací, než jakou dovolují vektorová data.
- Vektorová data mohou být zobrazena jako vektorová grafika tradičních map a při zvětšování nedochází ke ztrátě kvality, kdežto rastrová data při zvětšování ztrácí ostrost a obraz tak blokovatí do typické rastrové mřížky (podle rozlišení samozřejmě).
- Vektorová data se snadněji přidávají, zvětšují a kombinují. Například postavili se nová silnice, lze ji jednoduchou editací přidat. V takovémto případě by musel být rastr kompletně, nebo přinejlepším částečně reprodukován.
- Vektorová data jsou kompatibilní s různými databázemi.



Obr. 1.2: Rozdíly mezi zvětšováním rastrového a vektorového textu. U rastru použito vyhlazování.

1.2 Digitalizace map

Na poli GIS je tento úkon jedním z nejnáročnějších. Existuje mnoho metod. Digitalizace skenováním je jednou z nejjednodušších a lze tak získat data z klasických papírových map či z fotografických filmů. Skenováním získáme rastrové data které mohou být dále použity pro získání vektorových dat.

Hlavním zdrojem digitálních dat však zůstávají fotografické interpretace leteckých fotografií. Nejužívanější technikou je zpracování dvou až tří rozměrné, kdy se pořizuje stereoskopicky dvojice fotek a zároveň se zaznamenávají data o výšce. S příchodem super-kvalitních snímacích čipů také odpadla nutnost skenování analogových filmů [1].

V profesionálních aplikacích je nutnost digitalizované data upravovat, odstraňovat chyby nebo použít obrazové filtry. U skenovaných dat se nejčastěji setkáváme se skvrnami na zdroji dat. Ty musí být odstraněny, protože může dojít k takovým chybám, že konkrétní skvrna spojí dvě silnice, které by neměly být spojeny.

1.3 Formáty digitálních map

V amatérských použitích se pravděpodobně nejčastěji setkáme s formáty, které dostaneme na výstupu skenovacího programu použitého skeneru. Jsou to tedy rastrové data ve formátech BMP, TIF, JPEG a PNG. Jednu charakteristiku mají tyto formáty pro skenované mapy společné, neobsahují žádné další informace. Nelze je (kromě zpracování) nijak dále interpretovat, musíme se tudíž spokojit jen s obrazovým výstupem a zpracováním našima očima [7].

V profesionální sféře GIS se setkáváme s mnoha formáty, dokonce některé firmy si dělají formáty vlastní. Následující výčet tak pokrývá jen ty nejpoužívanější formáty (vztaženo na evropský kontinent).

Rastrové formáty:

- **ECW** – Enhanced Compressed Wavelet. Komprimovaný waveletový formát, většinou ztrátový.
- **ESRI grid** – binární či ASCII rastrový formát.
- **GeoTIFF** – Varianta formátu TIFF, která obsahuje rozličná GIS metadata.
- **IMG** – Formát grafického editoru GIS dat ERDAS.
- **MrSID** – multiresolution seamless image database. Je to formát patentovaný společností LizardTech, je to komprimovaný waveletový formát, ztrátový.
- **JPEG2000** – Open-source rastrový komprimovaný formát, ztrátový i bez-ztrátový.

Vektorové formáty:

- **GML** – otevřený standard založený na XML.
- **Shapefile** – open-source formát, hybridní.
- **GeoMedia** – speciální formát vytvořený společností Microsoft pro prostředí Windows.

1.3.1 ShapeFile

Tento formát je asi nejpoužívanějším typem formátů pro systémy GIS. V současné době je používán mnoha českými geodetickými firmami a v hojnosti je používán i na internetu. Byl vyvinut společností ESRI (Environmental Systems Research Institute) jako open-source specifikace pro použití s produkty od ESRI a i jinými dalšími programy. Formát samotný byl představen v roce 1990. Slovo *shapefile* obecně označuje několik souborů s koncovkami *.shp, *.shx, *.dbf atp. Hlavním souborem shapefile je ten s koncovkou *.shp, ale jeden bez druhého jsou tyto soubory nepoužitelné. Proto také můžeme hovořit v souvislosti s shapefile o *kontejneru*.

Shapefile soubory popisují geometrické útvary, tak jak jsme si je popsali výše – body, křivky a polygony. Každá z těchto položek pak může mít atributy popisující každou položku, například jméno či velikost. Útvary spolu s atributy mohou společně vytvořit mnoho reprezentací geografických dat.

Jednotlivé soubory v kontejneru se rozdělují na hlavní a rozšířené. V hlavních jsou uloženy nejdůležitější data a v rozšířených indexovací data pro zlepšení výkonu. Tyto soubory také musí splňovat klasické MS DOS 8.3 pojmenování (osm písmen prefix a tři písmena koncovka). Tato konvence se udržuje i v moderních aplikacích a z toho stejného důvodu by měli být všechny soubory ve stejné složce.

Hlavní soubory:

- ***.shp** – shape formát, obsahuje samotnou geometrii.
- ***.shx** – shape index formát, prostorová indexace geometrie.
- ***.dbf** – formát atributů, sloupcové atributy pro každý útvar.

Limitace:

- Shapefile nemá možnost uchování topologických informací.
- Okraje křivek a polygonů jsou definovány body, což může při vyšších rozlišeních vyústit až v kostrbaté hrany. Z tohoto důvodu musí být využito dalších bodů pro získání hladkých okrajů. Pokud by bylo využito bézierových křivek, nepotřebovaly bychom tolik bodů, bohužel formát shapefile toto neumožňuje.

2 INTERAKTIVNÍ MAPY NA INTERNETU

Internet je v dnešní době nepostradatelným nástrojem pro přístup k elektronickým mapám. V současnosti si již nikdo nedokáže představit svět, ve kterém by neexistovali webové aplikace s mapami a jejím vyhledáváním, stejně jako si profesionální řidič nedokáže představit, že by vyměnil svůj GPS (Globální polohový družicový systém – Global Positioning System) přístroj za klasickou skládací papírovou mapu.

Vynechejme Internet jako prostředek pro přenos souborů (tedy i map v jednoduché neskenované formě) a zaměřme se pouze na vztah klient-server, kde se klient, s přístupem do celosvětové sítě Internet, prostřednictvím svého webového prohlížeče připojuje na vzdálený server, který nabízí různé webové aplikace s prací s mapovými podklady.

2.1 Milníky v rozvoji webových map

Tento krátký seznam nepostihuje vývoj Internetu jako takový, ale měl by ukázat nejdůležitější inovace, díky nimž by mapy, jak je známe teď, na Internetu nemohly existovat.

- **1989** – V Evropském výzkumném středisku CERN se zrodil WWW pro výměnu zpráv z výzkumu.
- **1990** – První webový prohlížeč a server.
- **1991** – HTTP verze 0.9, první použitelná verze protokolu pro komunikaci mezi serverem a prohlížečem.
- **1993** – HTML 1.0, první verze jazyka pro psaní webových stránek, jeho autory jsou T. Berners-Lee a Dan Connolly.
- **1993** – Xerox PARC Map Viewer. První ze statických stránek s webovými mapami byla vyvinut v červnu roku 93 Stevem Putzem. Ve své podstatě se jednalo o experiment dokazující jakých možností je Internet schopný a jak může být důležitý při získávání relevantních informací. Celý projekt byl naprogramován v jazyce Perl. Mapy byly generovány v rastrovém formátu GIF ze speciální geografické databáze. Od té doby je projekt ve stádiu pozastavení vývoje.
- **1994** – První online atlas zobrazující rastrové obrazy Kanady.
- **1995** – Java 1.0 a JavaScript 1.0, první skript pro základ interaktivity.

- **1996** – Mapquest, první webová služba pro hledání adres a ulic využívající map jako výstupu.
- **2000** – Společnost ESRI založila síť pro distribuci dat, včetně mapových podkladů.
- **2003** – NASA uvolnila program World Wind, virtuální globus, který sám o sobě neběžel v prohlížeči, ale načítal data z různých zdrojů po celém světě. Povrchy a budovy bylo možné prohlížet trojrozměrně. Uživatel mohl využít značkovací funkce, a tak vkládat své vlastní poznámky.
- **2005** – Google Maps. První verze asi nejznámější a nejpoužívanější webové služby. Je založena na rastrových dílcích organizovaných do kvadrantové stromové struktury. Tato mapová aplikace je úspěšná také díky tomu, že povoluje jiným subjektům na internetu integrovat službu do jejich vlastních stránek.
- **2005** – Google Earth. První verze dnes nejpoužívanější obdoby NASA World Wind. Tak jako u jejího předchůdce je možné globus prohlížet trojrozměrně. Mapové podklady jsou načítány ze stejné databáze jako v případě Google Maps, ale je nutno použít speciální software.

2.2 Technologie

Implementace map na web skýtá neomezené množství technologií. Může být využito jakékoliv programovací prostředí, jazyk, či programová struktura na straně serveru. Následuje strohý výčet nejpobulárnějších technologií pro implementaci mapových aplikací na straně serveru a klienta [7].

Technologie na straně serveru:

- **Web server** – je zodpovědný za správné zacházení s http žádostmi od webových prohlížečů. V nejjednodušším případě pracuje se statickými soubory jako jsou HTML stránky a obrazové soubory. Nejpobulárnější web servery jsou Apache a Microsoft Internet Information Server. Web servery také pracují s rozpoznáváním obsahu a předáváním požadavků dynamickým zdrojům jako jsou například CGI aplikace.

CGI (common gateway interface) aplikace běží v prostředí web serveru. Mohou být napsány v rozličných programovacích jazycích, či skriptovacích jazycích. CGI aplikace zpracovávají informace poslané klientem a výsledek posílají zpět ve formě elementu, jež lze na webu zobrazit. Jako příklad můžeme uvést děj, kdy prohlížeč pošle požadavek CGI aplikaci. Požadavek obsahuje příkaz pro

zobrazení mapy v příslušném stylu a s několika vrstvami. Výsledek je obrazový formát (JPEG, PNG), který je poslán zpět prohlížeči.

- **Webový aplikační server** – je tzv. middleware, program spojující několik softwarových komponent s webovým serverem. Typické aplikační servery jsou napsány v programovacích jazyk Java, C++ a C#. Tyto servery jsou velmi užitečné při budování mapových aplikací pracujících v reálném čase.
- **Prostorové databáze** – objektové databáze, které jsou rozšířeny o geografická data. Jsou využívány v případech, pokud jsou data dynamicky měněna.
- **WMS servery** – specializované servery pro webové mapy implementující CGI aplikaci, Java Servlet a jiné aplikační servery. Pracují buď jako samostatné web servery, nebo spolupracují s existujícími web servery. WMS servery generují mapy na požádání. Používají parametry jako pořadí vrstev, symboly, projekce atd. WMS je standard definující požadavky a výstupy. Typickým výstupním formátem jsou obrazové formáty PNG a JPEG. Existuje i open-source WMS server UMN Mapserver.

Technologie na straně klienta:

V našem případě uvažujeme pouze práci s webovým prohlížečem. Všechny současné prohlížeče podporují HTML a rastrové obrázky. Webový prohlížeč by měl podporovat tyto technologie pro bezproblémovou spolupráci s mapovými servery.

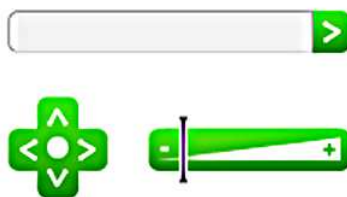
- **ECMAScript** – standardizovaná verze JavaScriptu. Důležitá část pro interakci mezi klientem a serverem.
- **Události** – webový prohlížeč by měl podporovat události jako je pohyb myši, stisk klávesy, atp.
- **Síťové požadavky** – jsou důležité pro nahrání dodatečných dat do webové stránky. Většina moderních prohlížečů nabízí XMLHttpRequest funkci, která dovoluje na již nahraných stránkách provádět určitou odezvu, již nestaví na HTTP protokolu. Tato funkce je nejčastěji reprezentována ECMAScriptem.
- **DOM** – podpora objektového modelu dokumentu, poskytuje jazykově nezávislé API pro manipulaci se strukturou dokumentu na webové stránce. Ve všech moderních prohlížečích je implementován. Podpora DOM spolu se podporou skriptování je také známa jako DHTML (Dynamic HTML). DHTML spolu s dalšími je využívána například Google Maps [21].
- **Java** – některé webové aplikace využívají tohoto programovacího jazyka. Je nutná instalace virtuálního stroje.

2.3 Základní funkce

Současné nejmodernější počítačové mapy mají mnohé interaktivní prostředky, díky nimž se může uživatel pohodlně dostat k informacím. Mezi takové funkce patří například přiblížení a oddálení mapy. Toho lze dosáhnout třemi způsoby:

- Záměna jedné mapy za podrobnější a naopak.
- Zvětšení mapy bez zvětšení pixelů (pokud je mapa původně zobrazena s méně detaily).
- Zvětšení mapy se zvětšením pixelů – nedochází k zobrazení dalších detailů. Tento způsob je nejméně vhodný. Může být také využito některých interpolačních algoritmů ke zlepšení kvality.

Další důležitou funkcí, bez které by žádný dnešní mapový server nemohl konkurovat, je vyhledávání ulic. Vyhledávání je většinou založeno na fulltextovém vyhledávání v databázi web serveru. Ve spolupráci s CGI je pak uživateli na výstupu zobrazen správný úsek mapy se zvýrazňující značkou daného místa.



Obr. 2.1: Ukázka typického ovládání webových map.

Dle statistiky každý druhý člověk prohlížející webové mapové aplikace také potřebuje znát cestu k danému cíli. Tato funkce se zakládá na měření vektorových křivek a jejich následném zpracování určitým matematickým algoritmem pro vyhodnocení té nejkratší cesty. Většina těchto algoritmů vychází z teorie grafů (např. heuristický algoritmus).

2.4 Klasifikace webových map

Téměř každé odvětví má svou vlastní klasifikaci. Ne jinak je tomu u webových mapových aplikací. Následující rozdělení je řazeno od nejjednodušší implementace k nejsložitější.

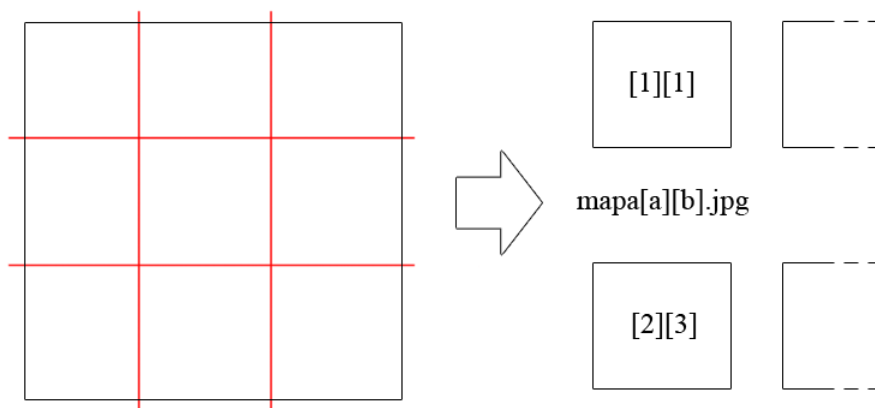
- **Statické webové mapy** – určeny pouze pro prohlížení, nemají animaci ani interaktivitu. Jsou generovány pouze jednou a musí se manuálně upravovat. Jsou prezentovány v rastrových formátech PNG, JPEG a GIF, nebo ve vektorových SVG, PDF či SWF. Nejčastěji jsou to pouze naskenované papírové mapy. Tyto mapy nejsou určené pro zobrazení na počítači, protože obsahují mnoho informací. V určitých použitích tak může docházet k nepřehlednosti a nemožnosti rozlišit i text v mapách [7].
- **Dynamické webové mapy** – tyto mapy jsou dynamicky vytvářeny dle požadavku pokaždé, když uživatel načte stránku. Používají dynamické zdroje dat, jako jsou databáze. Mapy jsou vytvářeny softwarem přímo na serveru.
- **Distribuované webové mapy** – založené na WMS protokolu. Mohou načítat data z mnoha zdrojů zároveň. Jeden server například poskytne topologická data a druhý tematickou vrstvu.
- **Animované webové mapy** – ukazují změny map v průběhu času. Například změna osídlení New Yorku v 20. století.
- **Webové mapy v reálném čase** – ukazují reálnou situaci téměř v tom stejném okamžiku, v jakém mapu sleduje uživatel. Typicky se jedná o monitoring provozu či mapy aktuálního počasí.
- **Přizpůsobitelné webové mapy** – tyto mapy dovolují uživateli filtrovat nabízené data a tak upřednostnit některé data na výstupu.
- **Interaktivní webové mapy** – interaktivita pomáhá prozkoumávat mapy, mění parametry map, pomáhá navigovat, zobrazuje další informace, odkazuje na další zdroje atp. Technicky je toho dosaženo kombinací událostí, skriptů a DOM manipulace.

3 NÁVRH APLIKACE

Navržená aplikace bude sloužit pro vyhledávání v mapových podkladech rastrového typu. V návrhu je kladen důraz na jednoduchost následné realizace a uživatelskou ergonomii. V neposlední řadě je důležitou částí vhodný systém pro vkládání údajů. V realizaci bude nejspíše použito dvou režimů: uživatelského a administrátorského.

3.1 Vstupní data

Zdrojovým souborem pro vstupní data je naskenovaná papírová mapa, tedy rastrová data. Pro použití na internetu je nutné takto velký obrazový soubor (může se jednat až o desetitisíce pixelů) rozdělit na menší bloky. Předpokládejme, že 50×50 pixelů je optimální rozlišení pro dynamické zobrazování v internetu. Pro přípravu takovýchto malých součástí obrazu je vhodné využít některý z volně dostupných programů, např. freewarový program „The Castle’s Split Image“ [18]. Již v tomto bodě je vhodné data správně pojmenovat např. podle matice s proměnnými a, b , které značí řádky a sloupce zdrojové mapy. Bez tohoto kroku nebude jednoduché zpětně mapu rekonstruovat v aplikaci.

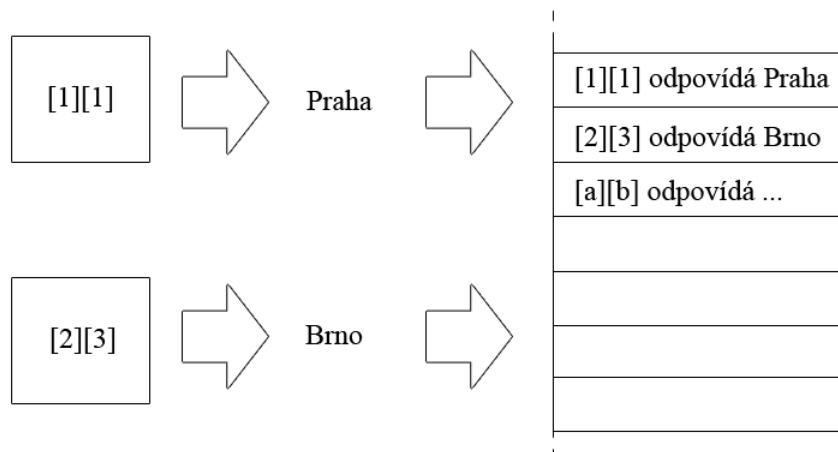


Obr. 3.1: Získání jednotlivých segmentů z mapy.

3.2 Vytvoření databáze

V této další části je nutný zdlouhavý zásah administrátora dat. Jak je v této práci již zmíněno, v oblasti GIS je tento krok složitý a nelze se mu vyhnout. Je nutné vytvořit databázi, která bude obsahovat přiřazení jednotlivých maticových souborů $mapa[a][b]$ městům, které se v dané oblasti vyskytují. Samozřejmě není potřeba přiřazovat všechny maticové soubory, pouze ty, které obsahují nějaké město. Není

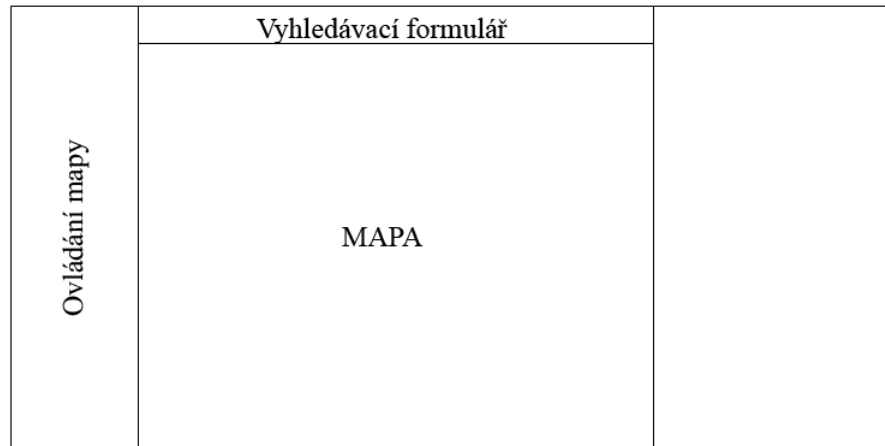
také nutná duplikace přiřazení. Stačí u větších měst (rozprostírající se na rozlehlejší ploše) vhodně vybrat maticový soubor, pro následné zobrazení v aplikaci (zobrazení na střed).



Obr. 3.2: Vytvoření databáze měst.

3.3 Rozvržení aplikace

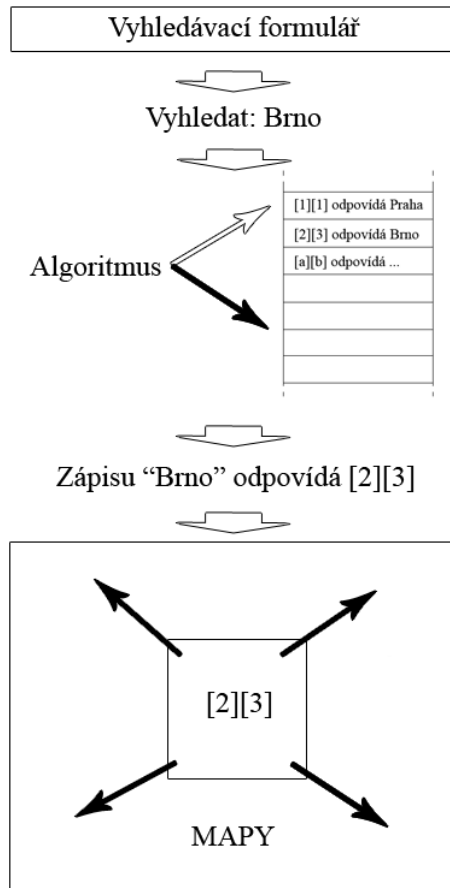
Důležitou součástí návrhu je bod věnující se rozvržení daného programu. Hlavní a největší oblastí bude samozřejmě část zobrazující mapu. Na obr. 3.3 vidíme, že nad touto centrální oblastí bude rámeček s dotazníkem pro hledaný výraz (v našem případě hledané město). Levý rámeček bude obsahovat standardní ovládací prvky jako přiblížení a pohyb v mapě. Levý rámeček je vyhrazen pro další případné funkce, které budou specifikované až při realizaci.



Obr. 3.3: Návrh rozložení aplikace.

3.4 Základní funkce

- **Zobrazení mapy** – Nejzákladnější funkcí je samotné zobrazení mapy. Uživatel vybere příslušné město a tedy i příslušný maticový soubor. Následně se tento zobrazí uprostřed matice s jedenácti řádky a jedenácti sloupci. Díky tomu bude mít výsledná mapa 550×550 pix, což je rozměr dostatečný a vhodný pro takto jednoduchou webovou aplikaci. Samotný algoritmus zobrazení matice bude počítat s krajními oblastmi zdrojové mapy a tyto bude zobrazovat prázdné (případně s upozorněním o konci mapy).
- **Ovládání mapy** – Nezbytnou funkcí webových map je posun základními směry. Toho je dosaženo pouhým posunem v maticových souborech (záleží na algoritmu a budou vyřešeny krajní oblasti mapy).
- **Přiblížení a oddálení** – Bude založeno na načtení detailnější mapy, případně zvětšení té stávající.
- **Vyhledávání** – Princip vyhledávání je zobrazen na obr. 3.4. Pokud tedy uživatel zadá do vyhledávání město Brno, algoritmus prohledne databázi a určí maticový soubor pro zobrazení ve středu matice 11×11 .



Obr. 3.4: Princip funkce vyhledávání.

4 VÝBĚR VHODNÉ PLATFORMY

Technologií a postupů v oblasti programování a implementace map do webového prostředí je velké množství. Dokážeme si představit aplikace běžící na platformách Java, .NET, Flash či Silverlight. Velkým problémem by také nebyla implementace programu napsaném v jazyce C#, či Visual C++ přes CGI do webového serveru. Samozřejmě lze uvažovat i o nejrozšířenější technologii na základě JavaScriptu, AJAX a PHP běžící nad databází MySQL.

Při výběru jazyka byl kladen důraz na tyto kritéria:

- Multiplatformní použití.
- Snadnost implementace do webového rozhraní.
- Efektivita pro zvolený typ aplikace.

Co se týče multiplatformního použití, tak všechny výše zmíněné technologie jsou v současnosti spustitelné na všech softwarových či hardwarových platformách pro osobní počítače. V tomto případě však má však největší výhodu aplikace založená na skriptovacích jazycích. Není třeba na počítač instalovat žádný další software kromě webového prohlížeče, u ostatních technologií je potřeba instalovat frameworky a virtuální stroje (u Javy).

U implementací jsou na tom frameworkové platformy velmi obdobně a mají malou výhodu nad skriptovými aplikacemi, zejména díky důrazu na vizualizaci.

Program k této práci by bylo neefektivnější vytvořit pomocí programovacího jazyka PHP nad databází MySQL. Cílem aplikace není konkurovat vyhlášeným aplikacím na internetu a vybrané technologie se vyznačují i jednoduchým syntax. Případné další prvky mohou být vyřešeny pomocí technologie Ajax resp. jazyka JavaScript.

5 PRACOVNÍ JAZYKY A PROSTŘEDKY

5.1 PHP

PHP je skriptovací jazyk původně určený pro generování dynamických webových stránek. Postupem času se však vyvinul do podoby, kdy je možné naprogramovat i samostatnou aplikaci s uživatelským prostředím. PHP byl vytvořen Rasmem Lerdorfem v roce 1995 [16], hlavní implementace je nyní tvořena skupinou The PHP Group a vydávána jako „de facto standard“ jelikož neexistuje žádná formální specifikace. PHP je volně šiřitelný software vydaný pod licencí PHP License, která však není slučitelná s licencí GNU General Public License.

PHP je široce užívaným skriptovacím jazykem pro webové vývojáře a je zabudovatelný do HTML. Běží na serverové straně. Vstup je tedy PHP kód a výstup je webová stránka. Může být umístěn na téměř jakémkoli serveru a na patrně všech operačních systémech a platformách.

5.1.1 Stručná historie

Rasmus Lerdorf vyvinul PHP jazyk nejprve za osobním účelem. Chtěl tím vylepšit svou osobní webovou stránku. PHP syntax je v mnohém založen na syntaxi jazyka C a Perl. Základní veřejná verze byla vydána v červnu roku 1995 zejména pro rychlejší vyhledání a odstranění chyb [15].

V roce 1997 dva izraelští programátoři Zeev Suraski a Andi Gutmans přepsali parser a formovali základ PHP 3.

V současnosti používanou verzí je verze PHP 5, vydaná roku 2004 a vyvíjená po boku PHP 6.

PHP zatím nemá kompletní nativní podporu Unicode kódování ani multibytových řetězců. Podpora má být až ve verzi 6.

5.1.2 Použití

PHP je jazyk určený především pro webové vývojáře, PHP běží na straně serveru. PHP se primárně chová jako filtr, kterému je na vstup přiváděn tok textu nebo PHP instrukcí, a z výstupu vychází jiný tok dat. Většinou je tím výstupem kód HTML. Ve své podstatě je PHP podobný dalším skriptovacím jazykům pracujícím na straně serveru, například Active Server Pages (ASP) [26], JavaServer Pages JSP [27].

V současnosti je PHP používán na cca 25mil webových stránkách. Mezi největší webové aplikace využívající mimo jiné i PHP patří Facebook [28], Yahoo! [29] a YouTube [30].

5.1.3 Syntax

PHP kód se zapisuje mezi omezovače. Jakýkoliv zápis mimo omezovače je poslán přímo na výstup a není zpracován PHP. Nejčastějšími oddělovači jsou počáteční oddělovač `<?php` a ukončovací `?>`. Pro zrychlení práce se používá `<?` jako počáteční oddělovač, nebo verze `<?=>`, která se používá pro výstup řetězců nebo proměnných.

Proměnné jsou rozlišeny symbolem dolar v prefixu a datatyp této proměnné nemusí být definován. Na rozdíl od názvů funkcí a tříd jsou názvy proměnných závislé na velkých a malých písmenech. Řetězce se do proměnných vkládají pomocí oddělovacích znaků `"` resp. `'`. PHP nakládá s novými řádky a prázdnými místy jako ostatní jazyky typu free-form (pokud nejsou uvnitř řetězce). Výrazy musí být zakončeny středníkem. PHP má tři druhy syntaxe komentářů: `/* */` pro blok textu a `//`, nebo `#` pro jednořádkové komentáře. Výraz `echo` se používá pro výstup proměnné, nebo textu jako textového výstupu. PHP následuje základní syntax jazyka C i v klíčových slovech. Je možné využívat podmíněk `if` i smyček `for` a `while` stejně, jako v ostatních programovacích jazycích [4].

PHP v základu obsahuje volně šiřitelné knihovny pro přístup k FTP serverům nebo databázovým SQL serverům.

5.2 MySQL

MySQL je relační databázový řídicí systém. Podporuje databáze, jež tvoří množina relací a tyto relace jsou tabulky. Program samotný se chová jako server poskytující mnohonásobný přístup do určitého počtu databází.

Zdrojový kód tohoto projektu je vydáván pod licencí GNU General Public License. Je-li to třeba, je k dispozici i komerční verze. MySQL systém je vlastněn a sponzorován švédskou firmou MySQL AB, která je jednou z dceřiných firem Sun Microsystems [3].

MySQL je populární ve webových aplikacích a je to databázový komponent softwarového balíku LAMP (Linux, Apache, MySQL, PHP). Jeho popularita stoupala s popularitou PHP. Mezi nejznámější hráče patří webové aplikace Facebook [28], Wikipedie, Google a Youtube [30].

MySQL je napsán v programovacích jazycích C a C++, pracuje na mnoha rozličných systémových platformách včetně AIX, BSDi, FreeBSD, HP-UX, i5/OS, Linux, Mac OS X, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, eComStation, OS/2 Warp, QNX, IRIX, Solaris, Symbian, SunOS, SCO OpenServer, SCO UnixWare, Sanos, Tru64 a Microsoft Windows.

Administrátor může použít pro administraci databáze pouze příkazového řádku nebo grafické prostředí MySQL Administrator a MySQL Query Browser. Existují i další nástroje pro správu databáze komerčního i nekomerčního rázu. Uvedeme pro

nás důležitý nástroj phpMyAdmin [31], což je webové grafické rozhraní pro správu a administraci MySQL (často používané webhostingovými poskytovateli).

5.2.1 Stručná historie

- V roce 1994 vyvinuta programátory Michaelem Wideniem a Davidem Axmarkem.
- První verze pro operační systém Windows 95/NT vyšla v lednu roku 1998.
- V roce 2003 vyšla verze 4.0 od tohoto data se datuje velký nárůst popularity systému.
- Sun Microsystems provedl akvizici s MySQL AB 26. února 2008.
- Do budoucnosti se počítá s verzí 6.0 a zavedením podpory paralelních procesů.

5.2.2 Základní pojmy

- **Entita:** představuje věc či osobu v reálném světě, o které budeme v databázi uchovávat nějaké informace [3].
- **Relace:** je vztah mezi dvěma a více entitami. Existují 4 typy relací 1:1 (spojuje 2 entity), N:1 a 1:N (spojení jedné entity s více entitami) a M:N (skupina entit ve vztahu s jinou skupinou).
- **Sloupec:** neboli atribut popisuje určitou část dat, kterou má každý záznam v tabulce.
- **Řádek:** neboli záznam je složen z hodnoty pro každý sloupec v tabulce.
- **Klíč:** je nejmenší superklíč. Podle klíče tedy sloupce můžeme identifikovat záznam v tabulce. Pokud je tato identifikace jednoznačná jedná se o tzv. primární klíč.
- **Schéma:** je struktura či návrh databáze, databáze bez jakýchkoli dat.

5.3 JavaScript

JavaScript je skriptovací jazyk používaný pro zpřístupnění objektů jiných aplikací. Je primárně využíván na straně klienta při vývoji dynamických webových stránek. JavaScript je dialekt standardizovaného ECMAScriptu [17]. JavaScript byl ovlivněn mnoha jazyky a byl vytvořen tak, aby vypadal jako jazyk Java, ale aby byl jednoduchý na pochopení pro vývojáře s nižším smyslem pro logiku programování.

S jazykem Java má JavaScript podobný syntax ovlivněný jazykem C, mají také podobné klíčové slova a konvence.

5.3.1 Stručná historie

JavaScript byl původně vytvořen Brendanem Eichem ze společnosti Netscape pod názvem *Mocha*. Prvně byla technologie implementovaná v prohlížeči Netscape Navigator v prosinci roku 1995. Jazyk se úspěšně rozšiřoval.

Díky velké oblibě Microsoft vyvinul kompatibilní dialekt nazvaný JScript, tento byl zahrnut v prohlížeči Internet Explorer 3.0, vydaném v srpnu roku 1996. Dialekty jsou tak podobné, že někdy může dojít k jejich záměně.

Netscape v roce 1997 nechal jazyk standardizovat pod názve ECMAScript.

Flexibilita a jednoduchost JavaScriptu může za to, že se stal tento jazyk jedním z nejvíce používaných jazyků pro webové vývojáře. Profesionály byl však zatracován až do doby nástupu AJAXu. Výsledkem tohoto zájmu jsou různé aplikační rámce a knihovny pro JavaScript [6].

5.3.2 Použití

Primárně je JavaScript určen pro psaní funkcí, které jsou zahrnuty do HTML stránek a ovlivňují DOM stránky. Nejčastěji je využíván na:

- Otevření nebo vyskočení nového okna. Toto nové okno má pak naprogramovanou velikost, pozici a další atributy.
- Validace a kontrola údajů a hodnot formulářů. Údaje jsou před vložením např. do databáze zkontrolovány, zda mají daný formát a odpovídají předpokladům.
- Změna obrázku při přejetí kurzoru. Tento efekt je používán zejména pro přitáhnutí uživatelovy pozornosti.

Jelikož je JavaScript spouštěn na klientské straně, odpovídá na požadavky rychle. Zdrojový kód je interpretován v JavaScript Engine.

5.3.3 Syntax

Do HTML kódu se skript vepíše buď mezi tagy:

`<script type="text/javascript"></script>`. Nebo se do počátečního tagu vepíše atribut `src="javascript"` s cestou k souboru s kódem. Java script je citlivý na velikost písmen, běžně se píše názvy objektů s počátečním velkým písmenem a názvy funkcí a proměnných malým. Výrazy jsou zakončeny středníkem. Komentáře jsou oddělovány podobně jako v PHP a to `//` pro jednořádkový komentář a `/* */` pro oddělení bloku textu [6].

Proměnné nemusí být deklarovány s určením datatypu. Proměnné deklarované uvnitř nějaké funkce pomocí slova `var` jsou lokální a přístupné jen uvnitř této funkce. Bez použití tohoto klíčového slova jsou proměnné globální a přístupné odkudkoli.

Příklad:

```
x = 0; // Globální proměnná
var y = 'globální'; // Globální proměnná (není uvnitř fce.)

function f(){
  var z~ = 'proměnná'; // Lokální proměnná
  dvacet = 20; // Gloální proměnná (bez var)
  return x; // x je zde přístupná
}
// Hodnota proměnné z~ již není přístupná
```

Textový výstup je posílán funkcí `alert`, tato však není ve standardní knihovně JavaScript, je podporována výhradně prohlížeči.

5.4 Ajax

AJAX (Asynchronous JavaScript and XML) je skupina vzájemně provázaných vývojových technik používaných k tvorbě interaktivních webových aplikací a tzv. „bohatých internetových aplikací“ (rich internet applications). S pomocí AJAXu mohou webové aplikace získat data asynchronně na pozadí, bez narušení zobrazení ve webovém prohlížeči. Data jsou získána užitím XMLHttpRequest nebo JSRS (JavaScript Remote Scripting). I přes svůj vlastní název tato technika nevyžaduje použití JavaScriptu ani XML, ani požadavky nemusí být asynchronní. Proto se již v dnešní době nepoužívá zkratka AJAX, ale název Ajax, který již nereprezentuje zmiňované technologie (ve zbytku textu tedy již jen Ajax) [5].

Ajax pracuje ve většině moderních prohlížečů a nevyžaduje žádný další software nebo hardware. Jednou z hlavních výhod tak je, že programátor není nucen se učit nový jazyk a zbavovat se technologií na straně serveru. Ajax představuje přístup ze strany klienta a je schopen komunikovat s J2SEE, .NET, PHP i CGI skripty a nezáleží tedy na serveru.

I když jsou myšlenky Ajaxu poměrně staré, představuje důležitou změnu pohledu na klasické schéma komunikace požadavek/odpověď.

5.4.1 Stručná historie

Termín Ajax se poprvé objevil na veřejnosti v roce 2005. Techniky asynchronního přístupu k datům jsou však mnohem starší. Java applety, představené v roce 1995, komunikovaly při běhu na klientské straně se serverem. V roce 1996 byl do Internet exploreru představen nový HTML element IFrame, který umožňoval totéž. Dnešnímu objektu XMLHttpRequest předcházela technika XMLHttpRequest ActiveX, představená v roce 1999. Velkým hnacím motorem pro rozvoj Ajaxu byla společnost

Google, která ho implementovala do několika svých oblíbených služeb, například Gmail, Google maps apod. Dalšími důležitými hráči jsou také společnosti Yahoo! a Amazon.

5.4.2 Technologie

V původním článku J. J. Garretta byly upřesněny technologie požadované pro správnou implementaci do webové aplikace.

- **XHTML** a **CSS** pro prezentaci obsahu.
- **DOM** pro dynamické zobrazování a interakci.
- **XML** a **XSLT** pro výměnu, manipulaci a zobrazení dat.
- **XMLHttpRequest objekt** pro asynchronní komunikaci.
- **JavaScript** pro seskupení předchozích technologií.

Od té doby však technologie pokročila a zjistilo se, že JavaScript není jediným použitelným skriptovacím standardem na klientské straně. Dále se došlo k poznatku, že XML/XSLT může být pro výměnu dat nahrazen jinými formáty [5].

5.4.3 Výhody a nevýhody

- **Snížení přenosu dat:** Rozsáhlé webové stránky obsahují spoustu stejného obsahu. Při tradičních požadavcích jsou tyto stejné elementy znovu a znovu načítány. Při použití Ajaxu se načítá právě jen nová část stránek.
- **Zrychlení a ergonomie:** Použitím Ajaxu lze na klientské straně vytvořit velmi elegantní ergonomické webové prostředí, které zrychlí uživatelské práce.
- **Problém s navigací:** Ajaxové aplikace jsou velmi složité a proto u nich nelze jednoznačně navigovat pomocí webového prohlížeče. Tyto jsou nahrazeny možností samotné navigace použitím složitějších technik.
- **Zvýšení HTTP požadavků:** Pokud je Ajax špatně implementován, může paradoxně zvýšený počet požadavků na server převýšit úsporu na přenosu dat.
- **Kompatibilita:** Na stolních počítačích je již kompatibilita plně vyřešena (nejnovější verze prohlížečů), avšak na mobilních zařízeních se můžeme setkat s problémy způsobenými nepřítomnou podporou JavaScriptu a XMLHttpRequest.

6 KYBER-MAPS

Tato kapitola se soustřeďuje na softwarovou část bakalářské práce, pracovně nazvanou Kyber-maps. Tato webová aplikace je fundamentálně postavena na skriptovacím jazyce PHP, který generuje HTML kód zobrazovaný uživateli ve webovém prohlížeči. Celá aplikace je postavena nad jednoduše navrženou databází MySQL, jejíž návrh počítá s možným dodatečným rozšířením. Dodatečnými technologiemi je Javascript pro kontrolu údajů zadávaných ve formulářích a Javascript kombinovaný s přístupem přes Ajax pro dodatečné získávání informací ze serveru.

Samotná aplikace je rozdělena na dvě samostatné webové stránky. Jedna je primárně využívána uživateli pro vyhledávání míst na mapě a druhá zabezpečená je určena pro administrátory vkládající jednotlivé mapové podklady a údaje.

6.1 Vstupní předpoklady

Pro správný běh aplikace je potřeba již v začátku dodržovat určité jednoduché pravidla. Předně se předpokládá, že existuje minimálně jedna osoba dohlížející na chod webové aplikace a starající se o vstupní data – administrátor. Vstupními daty se rozumí digitalizovaná mapa potřebného zaměření (topologická, politická apod.) a rastrového formátu. Tato pravděpodobně značně velká mapa (co se týká rozlišení) se musí pomocí dříve zmíněné utility „The Castle’s Split Image“ rozdělit na buňky či bloky stejného rozlišení [18]. Není nutné, aby měli buňky tvar čtverce. Aplikace je připravena i na obdélníkový tvar buněk, důležitá je pouze uniformita rozlišení buněk. Je možné pro tento krok použít „Kyber-maps slicing tool“, viz kapitola 6.3.6. Ideální velikost je však ze zkušeností doporučována na 100x100pix.

Buňky jsou ideálně pojmenovány ve formátu *nazev_cislo.jpg*, kde číslo je pořadí buňky čtené zleva doprava po řádcích dolů od čísla 1. Jako výstupní formát je striktně požadován rastrový komprimovaný formát JPEG s koncovkou *.jpg. Pokud nebudeme využívat FTP přístupu na server je vhodné nainstalovat nějaký běžně dostupný ZIP archivátor, nebo v případě OS Windows využít standardní komprimace ZIP v tomto systému.

Posledním a nejvíce důležitým předpokladem je dostatek času administrátora. Jak už bylo několikrát předesláno, vkládání mapových údajů je časově náročná činnost, a proto je vhodné tuto práci rozdělit mezi více administrátorů.

6.2 Vytvoření a návrh databáze

Webová aplikace Kyber-maps je navržena pro práci nad databází MySQL. Pokud využíváme služeb webhostingů je vhodné si předem zjistit informace o nabízené databázi. Kyber-maps využívá jedinou databázi, je tak vhodná i pro freehostingové

služby. Aplikaci lze samozřejmě provozovat na vlastním serveru, v tomto případě je nutné MySQL databázi nainstalovat dle návodu [8] a v případě OS Linux dle návodu [9].

Cílem je v naší pojmenované databázi vytvořit dvě tabulky *podklady* a *skupiny*. V těchto tabulkách budou vytvořeny sloupce *id_overall*, *jmeno_souboru*, *mesto*, *skupina*, *id*, *text*, resp. *id*, *vyska*, *sirka*, *prefixsouboru*, *velikost_sirka* a *velikost_vyska*. Tyto tabulky lze jednoduše vytvořit v příkazovém řádku po přihlášení do databáze (v případě vlastního serveru) nebo v PHPMyAdmin GUI (v případě webhostingu) těmito příkazy:

```
DROP TABLE IF EXISTS 'podklady';
CREATE TABLE 'podklady' (
  'id_overall' int(11) NOT NULL auto_increment,
  'jmeno_souboru' varchar(100) collate utf8_czech_ci NOT NULL
  default 'nic.jpg',
  'mesto' varchar(100) collate utf8_czech_ci default NULL,
  'skupina' int(11) NOT NULL default '0',
  'id' int(11) NOT NULL default '0',
  'text' varchar(200) collate utf8_czech_ci default NULL,
  PRIMARY KEY ('id_overall')
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_czech_ci
ROW_FORMAT=FIXED AUTO_INCREMENT=16 ;
```

```
DROP TABLE IF EXISTS 'skupiny';
CREATE TABLE 'skupiny' (
  'id' int(11) NOT NULL auto_increment,
  'vyska' int(11) NOT NULL default '0',
  'sirka' int(11) NOT NULL default '0',
  'prefix_souboru' varchar(50) collate utf8_czech_ci NOT NULL
  default 'a',
  'velikost_sirka' int(11) NOT NULL default '100',
  'velikost_vyska' int(11) NOT NULL default '100',
  PRIMARY KEY ('id')
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_czech_ci
ROW_FORMAT=FIXED AUTO_INCREMENT=9 ;
```

Příkaz *DROP TABLE IF EXISTS* zahodí tabulku se stejným názvem, pokud již existuje. *CREATE TABLE* příkaz vytvoří tabulku s určitým názvem a sloupci jako parametry uvedenými v závorce za názvem tabulky, jednotlivé sloupce jsou odděleny čárkou. Syntax je poměrně výstižný. Pro jakékoliv bližší informace navštivte české stránky s manuálem MySQL [10]. Výsledkem těchto příkazů jsou tabulky na obr. 6.1 a na obr. 6.2.

Na příloženém CD k bakalářské práci se v adresáři *Kyber-maps/mysql/* nachází soubor *mysql.sql*, což je textový skriptový soubor (obsahující posloupnost příkazů) sloužící k vytvoření požadované databáze. V PHPMyAdmin stačí tento soubor vybrat tlačítkem pod příkazovým polem. Na serveru v příkazovém řádku, viz [11].

Field	Type	Collation	Null	Key	Default	Extra	Privileges	Comment
id_overall	int(11)	NULL	NO	PRI	NULL	auto_increment	select,insert,update,references	
jmeno_souboru	varchar(100)	utf8_czech_ci	NO		nic.jpg		select,insert,update,references	
mesto	varchar(100)	utf8_czech_ci	YES		NULL		select,insert,update,references	
skupina	int(11)	NULL	NO		0		select,insert,update,references	
id	int(11)	NULL	NO		0		select,insert,update,references	
text	varchar(200)	utf8_czech_ci	YES		NULL		select,insert,update,references	

Obr. 6.1: Struktura databázové tabulky „podklady“.

Field	Type	Collation	Null	Key	Default	Extra	Privileges	Comment
id	int(11)	NULL	NO	PRI	NULL	auto_increment	select,insert,update,references	
vyska	int(11)	NULL	NO		0		select,insert,update,references	
sirka	int(11)	NULL	NO		0		select,insert,update,references	
prefix_souboru	varchar(50)	utf8_czech_ci	NO		a		select,insert,update,references	
velikost_sirka	int(11)	NULL	NO		100		select,insert,update,references	
velikost_vyska	int(11)	NULL	NO		100		select,insert,update,references	

Obr. 6.2: Struktura databázové tabulky „skupiny“.

6.3 Administrátorský režim

Na serveru je primárně umístěn ve složce *mapy/admin* pod skriptovým souborem *index.php*.

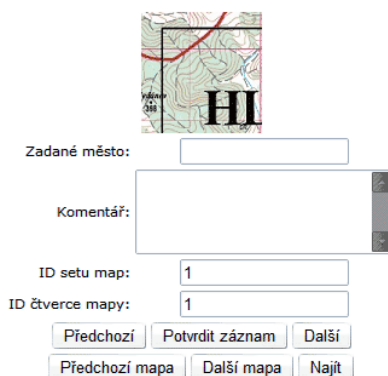
Pozn.: předpokládá se, že tato aplikace je doplňkem na serveru, a proto je primárně umístěna ve složce *mapy*. V jiném případě je třeba modifikovat zdrojový kód obou souborů *index.php*, přesněji jejich čtvrté řádky.

Samotná administrační stránka (načtená ve webovém prohlížeči), je rozdělena na tři funkční bloky.

6.3.1 Zadávání míst na mapě do databáze

Při prvním načtení stránky je v této sekci zobrazena první buňka prvního mapového setu v její plné velikosti (pokud je však nějaká mapa v databázi a na serveru nahrána – součástí bakalářské práce je jedna mapa okolí Hodonína). Obsah zobrazované buňky se pak mění dle požadavku administrátora. Na obr. 6.3 je tento blok zobrazen.

Zadávání míst na mapě do databáze:



The screenshot shows a web interface for entering map locations. At the top, there is a small map preview with a red line and the letters 'HI' overlaid. Below the map, the form consists of the following elements:

- Zadané město:** A single-line text input field.
- Komentář:** A multi-line text area.
- ID setu map:** A single-line numeric input field with the value '1'.
- ID čtverce mapy:** A single-line numeric input field with the value '1'.
- Buttons:** A row of three buttons: 'Předchozí', 'Potvrdit záznam', and 'Další'. Below this row are three more buttons: 'Předchozí mapa', 'Další mapa', and 'Najít'.

Obr. 6.3: Zadávání míst na mapě do databáze.

Následuje oblast s formulářem, jež je pevně spjatý s právě zobrazenou buňkou v databázi MySQL.

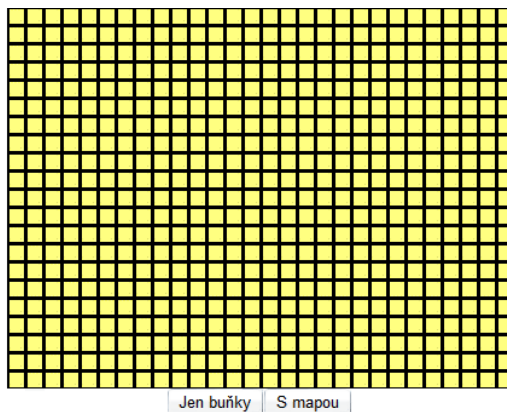
- **Zadané město:** Do pole se zadává město vyobrazené v buňce. Samozřejmě, že sem lze zadat i jiné místa na mapě. V databázi se jedná o sloupec *mesto* v tabulce *podklady*.
- **Komentář:** Víceřádkové textové pole, do kterého se vkládají dodatečné informace o místě, které se uživateli zobrazí jako kontextový tip. Při zadávání hypertextového odkazu je potřeba psát jeho celou formu i s *http://* a tento odkaz striktně oddělit složenými závorkami {}, jedině tak dojde ke správnému zobrazení tipu u uživatele. V tabulce *podklady* odpovídá sloupci *text*.
- **ID setu map a čtverce mapy:** Stavová textová pole, zobrazující číslo mapy, ze které je právě zobrazovaná buňka a taktéž číslo této buňky (odpovídá číslu v názvu souboru buňky). V tabulce *podklady* jsou svázány se sloupci *skupina* a *id*.

Pod oblastí formuláře se nachází šestice tlačítek pro práci s formulářem i samotnou mapou a její aktuální buňkou, tak jak je to vidět na obr. 6.3.

- **Předchozí a další:** Těmito tlačítky se dekrementuje, resp. inkrementuje, číslo čtverce mapy a dojde ke změně zobrazené buňky a údajů ve formuláři vypsaných z databáze.
- **Předchozí a další mapa:** Tato dvojice má obdobnou funkci. Rozdíl je pouze v tom, že zde se mění číslo setu map.
- **Potvrdit záznam:** Tlačítko zapíše informace z formuláře do databáze, v případě existujícího záznamu v databázi tento přepíše.

- **Najít:** Usnadňuje revizi a změnu údajů k příslušnému záznamu v databázi. Pokud se do pole *Zadané město* zadá existující místo v databázi a potvrdí se tímto tlačítkem, dojde k dohledání příslušných údajů v databázi, a je tak možná jejich okamžitá změna.

Další oblastí je oblast generovaná v závislosti na zvoleném mapovém setu. Jedná se vlastně o reprezentaci celkové zvolené mapy a aktuální buňka je zobrazena červeným čtvercem, tak jak ji můžeme vidět na obr. 6.4. Administrátor může pouhým kliknutím do této matice změnit aktuální zobrazovanou a popisovanou buňku a jedná se tak asi o neefektivnější způsob orientace při zadávání údajů. Dvěma tlačítky pod touto mapou se volí zda je mapa reprezentována pouze schématicky, nebo reálně zdrojovými daty. Tato volba však načte veškeré data mapy, což může být pro malé připojení celkem zdlouhavá záležitost.



Obr. 6.4: Síťová struktura mapy.

6.3.2 Upload archivu s mapou na server

V tomto funkčním bloku je administrátorovi umožněno nahrát mapový set, aniž by musel na server přistoupit přes FTP. Vybírá se soubor archivovaný algoritmem ZIP. V tomto archivu by se měli nacházet veškerá obrazová data spjatá s jednou mapou a to výhradně bez složkové struktury. Název tohoto archivu by měl být stejný jako název buněk z *název_číslo.jpg*. Po stisknutí tlačítka *upload* dojde k nahrání archivu na server a jeho automatickému rozbalení do složky *maps/název* se stejným názvem jako název archivu bez koncovky (z tohoto je tedy jasné, že při nahrávání dat přes FTP je nutné soubory jpg nahrát do takto pojmenované složky). Práce s archivy ZIP je poskytována externí knihovnou PlcZip speciálně vytvořenou pro práci s komprimovanými daty v prostředí PHP jazyka [22]. Ve zdrojovém kódu je obsažena jeho aktuální verze PlcZip 2.8.

Upload archivu s mapou na server:

Soubor:

Obr. 6.5: Upload archivu s mapou na server.

6.3.3 Zadávání mapového setu do databáze

Poslední funkční blok slouží pro vkládání údajů o mapových setech do databáze, přesněji do tabulky *skupiny*. Na obr. 6.6 vidíme, že tento formulář je opatřen jediným tlačítkem pro vložení či změnu údajů.

- **ID setu:** V případě vkládání nové mapy se nezadává. V případě změny je nutno zapsat číslo setu.
- **Výška a výška mapy:** Počet řádků a sloupců buněk.
- **Prefix souboru:** Zde se zapisuje prefix souborů jpg, již dříve zmiňovaný název v *název_číslo.jpg*.
- **Šířka a výška čtverce mapy:** Velikost jedné buňky v pixelech.

Zadávání mapového setu, změna mapového setu v databázi:

ID setu:
 (nezadávat v případě vložení)

Výška mapy
 (ve čtvercích):

Šířka mapy
 (ve čtvercích):

Prefix souboru:

Šířka čtverce mapy
 (v pix):

Výška čtverce mapy
 (v pix):

[Výpis existujících mapových setů](#) [Výpis zapsaných měst](#)

Obr. 6.6: Zadávání mapového setu do databáze.

6.3.4 Kontrola údajů a výpis údajů

Kontrola zadávaných údajů je zabezpečena pomocí JavaScriptové funkce *isNumeric*, která hlídá, zda se do textových polí formuláře, kde mají být zapsána pouze čísla, zapisují čísla. Dále JavaScript kontroluje, zda při kliknutí na tlačítko *Upload*, je zvolen soubor. V obou těchto případech při nesplnění podmínky dojde k vyskočení okna *ALERT* a zvýraznění polí, kterých se oznámení týká.

Kontrola údajů probíhá i na úrovni PHP skriptu, který hlídá především formát dat zapisovaných a čtených z databáze. V případě jakýchkoli problému na příslušných místech zobrazí chybové hlášení *Chyba zadávaných údajů*.

Na konci generované stránky se nacházejí dva hypertextové odkazy. První s názvem *Výpis existujících mapových setů* zobrazí tabulku s údaji o zapsaných mapách. Toto je vhodné pro kontrolu, pokud není přístup přímo k databázi. Druhý odkaz „*Výpis zapsaných měst*“ přejde na stránku, kde se vygenerují tlačítka s názvy mapových setů. Po kliknutí na jednotlivá tlačítka se generuje automaticky tabulka se zapsanými městy k příslušnému setu. Údaje v této tabulce obsahují hypertextové odkazy na uživatelský režim aplikace a dojde k zobrazení tohoto místa na mapě.

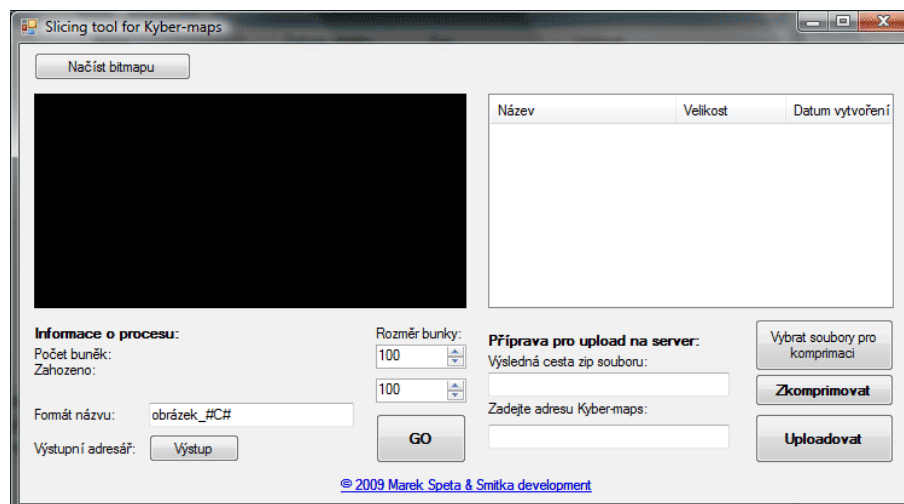
6.3.5 Zabezpečení

Není žádoucí, aby k tomuto administrátorskému režimu měl přístup každý, kdo ví, kde se nachází. K jednomu z nejjednodušších, avšak velice účinných, zabezpečení patří zabezpečení relace pomocí direktiv obsažených v souboru *.htaccess* a přihlašovacích údajů v souboru *.htpasswd*. Tyto soubory jsou na přiloženém CD také obsaženy. Je však nutné *.htaccess* modifikovat tak, aby byla zapsaná správná cesta k souboru *.htpasswd*, ve kterém jsou přihlašovací údaje ve formátu *jméno:heslo*. V přiloženém souboru je jméno **admin** a heslo **heslo**. Heslo je však zakódované algoritmem MD5 [14]. Na internetu existuje spousta stránek se zaměřením na generování souboru *.htpasswd* i se zakódováním hesla [23]. Stačí tyto dva soubory nahrát do složky *admin/* a veškeré soubory zde obsažené jsou nyní chráněny. Po zadání adresy administrátorského režimu v prohlížeči je administrátor vyzván k zadání jména a hesla, tato relace je pak uložena v paměti prohlížeče a není nutno je opětovně zadávat, dokud nedojde k zavření okna prohlížeče.

6.3.6 Slicing tool

V průběhu vývoje Kyber-maps se došlo k závěru, že na internetu neexistuje pohodlný software pro přípravu podkladů pro tuto aplikaci. Paralelně byla tedy vyvíjena utilita „Kyber-maps slicing tool“, která ve velké míře zjednodušuje administrátorovi práci. Utilita v jednom programu spojuje veškeré přípravné kroky od rozsekání mapy na jednotlivé buňky, přes jejich archivaci, až po automatické vyplnění formuláře pro zadávání údajů o novém mapovém setu. Utilita je napsána v jazyce C# [12] ve frameworku .NET [13] a vyvinuta ve vývojovém prostředí Microsoft Visual C# [24]. V OS Windows je pro správný chod potřeba mít nainstalovaný .NET framework 3.5 (v nižších verzích je program funkční, ale vizuálně nevypadá tak, jak byl zamýšlen) [25].

Utilita se nachází na CD v příslušném adresáři. Po jejím spuštění je zobrazeno okno se dvěma oblastmi. Z obr. 6.7 vyplývá, že první oblast nalevo slouží pro zpra-



Obr. 6.7: Okno programu Slicing tool.

cování vstupního souboru s mapou. Tlačítkem *Načíst bitmapu* se vybere soubor s mapou v rozměrech buňky se nastaví rozměr v pixelech. V textovém poli *Formát názvu* se změnou parametru *obrázek* zadá název výstupních souborů lišících se tedy pouze o pořadové číslo. Tlačítkem *Výstup* se nastaví výstupní adresář (nejlépe zcela nový). Po stlačení GO dojde k provedení operace.

V pravé části se připravuje samotný archiv pro nahrání na server. Administrátor vybere soubory pro komprimaci (primárně všechny soubory které byly v předchozí části vytvořeny) a tlačítkem *Zkomprimovat* se zadané soubory zkomprimují do archivu se stejným jménem jako je prefix souborů. Pokud administrátor zadá aktuální umístění webové adresy map dojde k otevření administrátorského režimu s již zapsanými všemi údaji ve formuláři vkládání údajů o nových mapových setech. Administrátorovi pak stačí jen archiv nahrán tak, jak bylo popsáno výše.

6.4 Uživatelský režim

Na serveru je umístěn v kořenovém adresáři aplikace v souboru *index.php*. Po zadání adresy v prohlížeči je uživateli zobrazena základní stránka s ovládacími prvky pro vyhledávání a posun v mapě. Dále pak oblast, ve které se vykresluje mapa. Bez zadání vyhledání je zde umístěno logo aplikace, tak jak je vyobrazeno na obr 6.8. Podobně jako v admin. režimu, je i zde možný přístup na stránku s výpisem měst. Údaje v generované tabulce obsahují hypertextové odkazy na uživatelský režim aplikace a dojde k zobrazení tohoto místa na mapě.

Kyber-maps

Projekt k bakalářské práci:
Interaktivní mapové podklady v síti Internet
Vedoucí práce:
Ing. Martin Koutný
Vypracoval:
Marek Špeta
VUT Brno 2009



Obr. 6.8: Logo aplikace Kyber-maps.

6.4.1 Vyhledávání

V levém horním rohu se nachází formulář pro vyhledávání měst. Je důležité ho napsat s diakritikou (pokud i v administrátorském režimu byly města zadány s diakritikou). Není však třeba dodržovat velké a malé písmena v názvech. Vyhledávání pracuje na principu vyhledání dané položky v databázi MySQL a zbylé údaje jsou ze stejného řádku načteny do potřebných proměnných PHP skriptu. Dle názvu se tak vyhledá zejména číslo buňky, tato buňka bude zobrazena uprostřed oblasti zobrazení. Vyhledáním se také zjistí *id setu map* a podle tohoto se v druhé tabulce zjistí potřebný počet sloupců a řádků pro vykreslení zbývajících buněk. Samotné zobrazování v HTML kódu je započato od levé horní buňky, jejíž číslo je pomocí údajů vypočteno a pokračuje doprava a po řádcích dolů. Po prvním vyhledávání je mapa zobrazena oddáleně v matici 13×13 buněk s naší hledanou buňkou uprostřed.

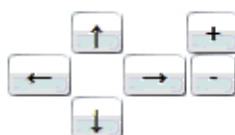
Vyhledej:

Obr. 6.9: Vyhledávání a jeho formulář.

6.4.2 Prohlížení

Pro prohlížení mapy slouží v pravém horním rohu čtveřice navigačních a dvojice tlačítek určených pro zvětšení a zmenšení mapy, viz obr 6.10.

- **Doleva a doprava:** Při stačení dojde k dekrementaci, resp. inkrementaci, čísla vyhledané buňky a znovu vykreslení oblasti zobrazení mapy. Podmínkami je ošetřeno případné načítání přes okraje.
- **Dolů a nahoru:** K číslu aktuální buňky se přičte, resp. odečte, počet sloupců zvoleného mapového setu. Tím dojde k opětovnému vykreslení mapy o řádek dolů, resp. nahoru. Opět je ošetřeno, aby zobrazení mapy skončilo na okraji zdrojových dat.
- **Přiblížení a oddálení:** V základu je mapa zobrazována oddálená. Po stisku přiblížení dojde k tomu, že se sníží počet zobrazovaných buněk na 7x7 a zvýší rozlišení buněk na požadovanou hodnotu tak, aby výsledný obraz měl stejné rozlišení jako oddálený. Pro jednoduchost jsou v programu pouze 2 stupně zoomu. Jednoduše lze však přidat další stupně přiblížení dle nastíněného algoritmu.



Obr. 6.10: Navigace v mapové aplikaci.

6.4.3 Kontextové tipy

Pokud v administrátorském režimu byl při zadávání místa vložen i příslušný komentář, je při vykreslení mapy k dané buňce přidána javascriptová událost (uživatel ji pozná změnou kurzoru myši ze šipky na ruku). Po kliknutí na buňku je pomocí Ajaxu odeslána informace o příslušném *id* buňky (jedná se o unikátní *id_overall* různé pro každou buňku, nehledě na mapu) do pomocného PHP skriptu, který pomocí tohoto *id* z databáze vyčte příslušný komentář. Následně je tento výstup PHP poslán do textového obsahu plovoucí bubliny kontextového tipu v podobě jako na obr. 6.11. Pokud uživatel na plovoucí kontextový tip klikne, tento se zavře.



Obr. 6.11: Ukázka kontextového tipu

7 ZÁVĚR

Tato práce se zaměřuje na elektronické mapy jako takové. První část práce je cílena na teorii map a jejich digitalizaci. Od prvních map k digitálním mapám je velmi dlouhá cesta. Mapy se buď digitalizují a složitě zařazují do systému GIS nebo pořizují přímo digitálně fotografováním a elektronickým snímáním. Ani v tomto případě ale neodpadá zdlouhavé vytváření databáze.

Elektronické mapy jsou dvojího druhu - buď vektorové nebo rastrové. V práci je vysvětlen rozdíl mezi těmito dvěma datatypy a je zde též rozebrána vhodnost použití v elektronické geografii. Ve stručnosti lze rastr popsat jako matici buněk s konkrétní hodnotou a to hodnotou barvy buňky v příslušné paletě. Vektorový datatyp je seznam objektů (bodů, křivek a mnohoúhelníků) a jejich velikostí, spolu s umístěním těchto objektů v prostoru.

Pokud hovoříme o jednotlivých formátech používaných v elektronickém mapování, lze vyzdvihnout rastrové formáty ECW, ESRI grid, GeoTIFF a IMG. Jsou to nejpoužívanější databázové rastrové formáty vedle JPEG2000, které se využívají nad GIS databázemi. V oblasti vektorových datatypů, které se používají v GIS nejčastěji (naproti tomu se rastrové formáty používají v komerčních webových mapách typu www.mapy.cz [19]), můžeme vyzdvihnout open-source formát ShapeFile. Tento formát má, i díky dobré podpoře programátorů, stále místo mezi amatérskými a profesionálními aplikacemi pro práci s mapami.

Kapitola, věnující se samotnému využití map na internetu, seznamuje s technologiemi multiplatformního programování a nejčastěji využívanými vývojovými prostředími. Díky CGI může webový server využívat jakýchkoli aplikací založených na různých technologiích a programovacích jazycích. Nejvyužívanější kombinací v současné komerční sféře webového mapování je kombinace MySQL databáze objektů a vrstvení rastrových a vektorových map s využitím AJAX a JavaScriptu.

V této práci byl také vybrán multiplatformní programovací jazyk pro aplikaci využívající mapových podkladů běžící v síti Internet. Byl zvolen skriptovací programovací jazyk PHP běžící nad databází MySQL, doplňkové funkce aplikace jsou založeny na jazyce JavaScript a využívá se i technologie Ajax.

Návrh aplikace se věnuje základním funkcím, které by měla realizovaná aplikace zahrnovat. Samozřejmostí tedy bylo navrhnout formální i datový formát vstupních dat, algoritmus pro vykreslení rastrové mapy z jednotlivých buněk a vytvořit si představu o uživatelské navigaci v aplikaci webových map.

Při realizaci se podařilo dosáhnout vytyčených bodů v návrhu aplikace a v mnohém je vylepšit a doplnit. Základní část webové aplikace je napsána ve skriptovacím jazyce PHP, který vytváří HTML webovou stránku zobrazovanou uživateli na základě jeho požadavků. Aplikace stojí nad databází MySQL, jejíž návrh je v textu také uveden. Aplikace je rozdělena na dvě části – administrátorskou a uživatelskou.

Administrátorský režim je zabezpečen přihlašovacími údaji tak, aby nikdo kromě určených osob nemohl měnit údaje v databázi. Administrátor zde zadává místa zobrazená v jednotlivých buňkách mapy a případně je může doplnit komentářem, který uživatel vyvolá jako kontextový tip. Součástí tohoto režimu je také formulář pro vkládání informací o novém mapovém setu a formulář pro nahrání mapy bez nutnosti na server přistupovat přes FTP. Údaje jsou validovány pomocí funkcí JavaScriptu i samotným PHP.

Uživatelský režim dovoluje uživateli vyhledat zvolené místo na mapách, toto místo je zobrazeno uprostřed oblasti s vykreslenou mapou. Touto mapou je umožněno procházet pomocí navigačních tlačítek a lze ji i přibližovat. Doplňkové informace jsou pomocí Ajaxu a JavaScriptu zobrazeny v plovoucím kontextovém poli ve tvaru bubliny. V případě nejasností lze zobrazit veškerá data obsažená v databázi – mapami počínaje a jednotlivými místy konče.

V průběhu realizace projektu nastal jediný problém a to nedostupnost freewarových aplikací, vhodných pro přípravu vstupních dat. Byl proto paralelně vyvíjen podpůrný program v jazyce C#, určený primárně pro ulehčení přípravných prací administrátorů. Aplikace rozděluje rastrové formáty a připravuje je pro nahrání na server.

Přínos bakalářské práce spočívá především ve výsledné aplikaci, která může velice dobře sloužit například jako webová mapa pro zobrazení mapy určitého areálu s jednotlivými místy areálu s poznámkami, důležitými pro orientaci v tomto areálu. Celý projekt je založen na freewarových softwarových komponentách a je tedy přístupný téměř každému. Implementace se může zdát poněkud složitější, ale veškerá příprava a realizace je v této práci popsána, aby aplikaci mohl spustit i méně zkušený administrátor. Výhodu můžeme vidět i v tom, že celá aplikace i podpůrný program jsou „vše v jednom“ a případnému administrátorovi stačí aplikaci zprovoznit na nějakém free-hostingovém serveru, podporujícím dané technologie, což ovšem v dnešní době podporují téměř všechny tyto hostingové služby.

LITERATURA

- [1] DAVIS, E.: *GIS for Everyone*. Esri Press, 2003. 196 s. ISBN 978-1589480568.
 - [2] VIRGA, V.: *Cartographia: Mapping Civilizations*. Little, Brown and Company, 2007. 272 s. ISBN 978-0316997669.
 - [3] WELLING, L., THOMSON, L.: *Průvodce základy databázového systému MySQL*. CP Books, 2005. 255 s. ISBN 80-251-0671-3.
 - [4] BRÁZA, J.: *PHP5 - začínáme programovat*. Praha, Grada, 2004. 244 s. ISBN 80-247-1146-x.
 - [5] ASLESON, R., SCHUTTA, N.: *Ajax – Vytváříme vysoce interaktivní webové aplikace*. Brno, Computer Press, 2007. 268 s. ISBN 80-251-1285-3.
 - [6] PÍSEK, S.: *JavaScript: Efektní nástroj oživení WWW stránek*. Praha, GRADA Publishing, 2001. 254 s. ISBN 80-247-0014-X
- Internetové články:**
- [7] VARIOUS.: *Map* [online]. 2003, poslední aktualizace 10. 12. 2008 [cit. 5. 12. 2008].
Dostupné z URL: <<http://en.wikipedia.org/wiki/Map>>.
 - [8] BRAŽINA, M.: *Instalace MySQL ve Windows* [online], 2000 [cit. 2. 5. 2009]
Dostupné z URL: <<http://www.builder.cz>>.
 - [9] BRAŽINA, M.: *Instalace MySQL v Linuxu* [online], 2000 [cit. 2. 5. 2009]
Dostupné z URL: <<http://www.builder.cz>>.
 - [10] SPOLEK AUTORŮ.: *MySQL manuál* [online], 2003-2005, posl. aktualizace 2005 [cit. 2. 5. 2009]
Dostupné z URL: <<http://mm.gene.cz/>>.
 - [11] LUKÁŠ, J.: *MySQL skripty* [online], 2001, [cit. 2. 5. 2009]
Dostupné z URL: <<http://www.supersvet.cz/>>.
 - [12] VARIOUS.: *C#* [online], 2004, poslední aktualizace 13. 5. 2009 [cit. 13. 5. 2009]
Dostupné z URL: <http://cs.wikipedia.org/wiki/C_Sharp>.
 - [13] VARIOUS.: *.NET* [online], 2005, poslední aktualizace 21. 3. 2009 [cit. 13. 5. 2009]
Dostupné z URL: <<http://cs.wikipedia.org/wiki/.NET>>.
 - [14] VARIOUS.: *MD5* [online], 2001, poslední aktualizace 3. 5. 2009 [cit. 10. 5. 2009]
Dostupné z URL: <<http://en.wikipedia.org/wiki/MD5>>.

- [15] VARIOUS.: *PHP skripting language* [online], 2001, poslední aktualizace 3. 4. 2009 [cit. 28. 4. 2009]
Dostupné z URL: <<http://en.wikipedia.org/wiki/PHP>>.
- [16] VARIOUS.: *PHP License* [online], 2006, poslední aktualizace 17. 3. 2009 [cit. 28. 4. 2009]
Dostupné z URL: <[http://en.wikipedia.org/wiki/PHP License](http://en.wikipedia.org/wiki/PHP_License)>.
- [17] VARIOUS.: *Standard ECMAScript* [online], 1998, poslední aktualizace 12. 12. 1999 [cit. 30. 4. 2009]
Dostupné z URL: <<http://www.ecma-international.org>>.
- WWW odkazy:**
- [18] *TheCastle: SplitImage* [online]. 2007. 1998, 13. 12. 2007 [cit. 10. 12. 2008].
Dostupné z URL: <www.thecastle.com>.
- [19] *mapy.cz* [online]. 2002, poslední aktualizace 5. 9. 2008 [cit. 10. 12. 2008].
Dostupné z URL: <www.mapy.cz>.
- [20] *Slovník cizích slov* [online]. 2002, poslední aktualizace 20. 11. 2008 [cit. 2. 12. 2008].
Dostupné z URL: <<http://slovník-cizich-slov.abz.cz/>>.
- [21] *Google maps* [online]. 2007, 2008 [cit. 2. 12. 2008].
Dostupné z URL: <<http://maps.google.com/>>.
- [22] *PlcZip - PHPConcept* [online]. 2003, 2009 [cit. 10. 5. 2009].
Dostupné z URL: <<http://www.phpconcept.net/plczip>>.
- [23] *HtacesTools* [online]. 2002 [cit. 10. 5. 2009].
Dostupné z URL: <<http://www.htacesstools.com/htpasswd-generator/>>.
- [24] *Microsoft Visual C#* [online]. 2008 [cit. 12. 5. 2009].
Dostupné z URL: <<http://www.microsoft.com/express/vcsharp/>>.
- [25] *Stahuj .NET framework* [online]. 2009 [cit. 12. 5. 2009].
Dostupné z URL: <<http://www.stahuj.centrum.cz>>.
- [26] *Microsoft Active Server Pages* [online]. 2007 [cit. 28. 4. 2009].
Dostupné z URL: <<http://www.asp.net>>.
- [27] *JavaServer pages* [online]. 2006 [cit. 28. 4. 2009].
Dostupné z URL: <<http://java.sun.com/products/jsp>>.
- [28] *Facebook* [online]. 2004 [cit. 28. 4. 2009].
Dostupné z URL: <<http://facebook.com>>.

- [29] *YAHOO!* [online]. 2000 [cit. 28. 4. 2009].
Dostupné z URL: <<http://yahoo>>.
- [30] *YouTube* [online]. 2004 [cit. 28. 4. 2009].
Dostupné
z URL: <<http://youtube.com>>.
- [31] *PHPMYAdmin* [online]. 2004 [cit. 28. 4. 2009].
Dostupné
z URL: <<http://www.phpmyadmin.net>>.

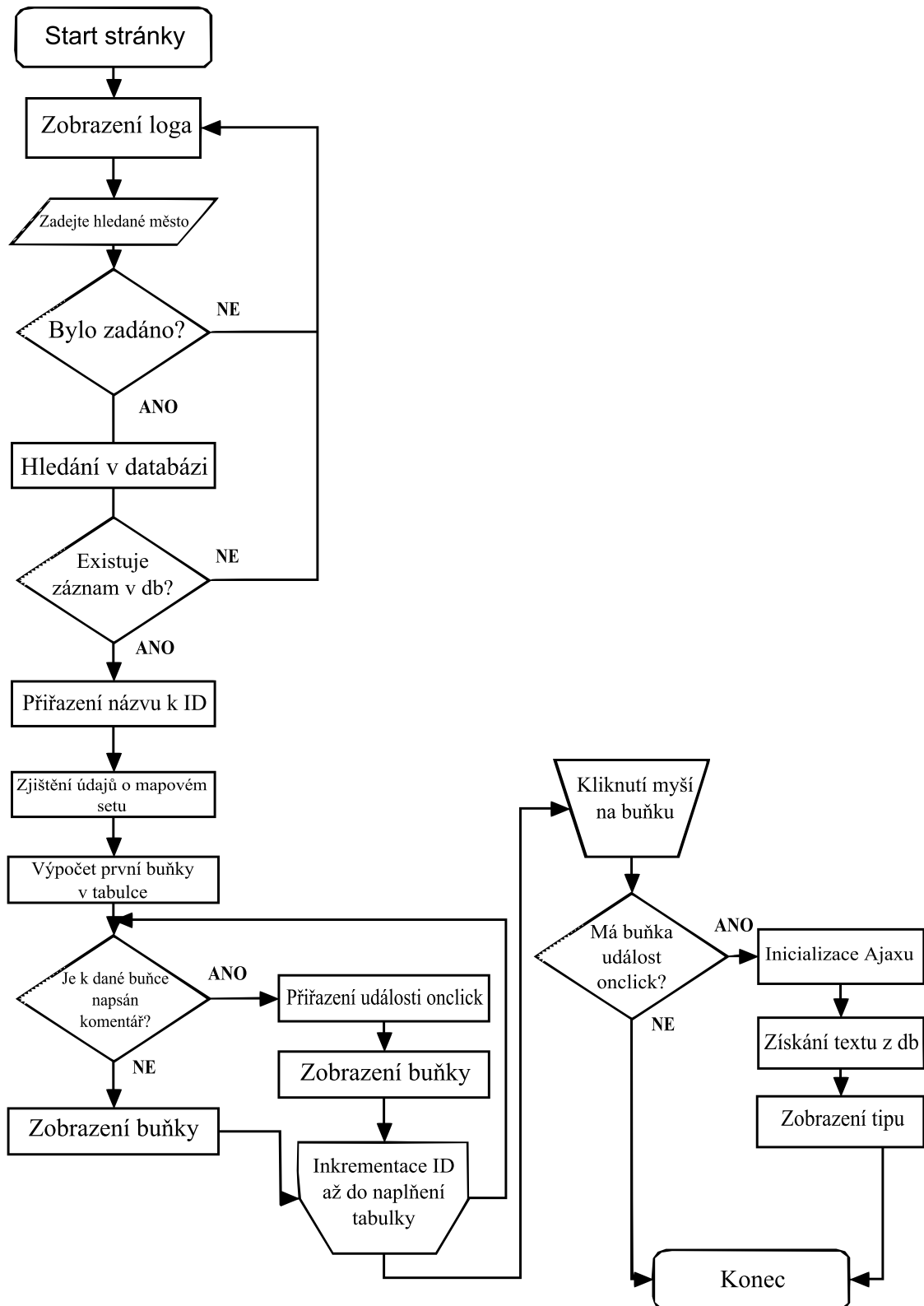
SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

AJAX	Asynchronous JavaScript and XML
API	Aplikační programové rozhraní Java – Application Programming Interface
CERN	Evropská organizace pro jaderný výzkum – European Organization for Nuclear Research
CGI	Common Graphic Interface
DOM	Objektový model dokumentu – Document Object Model
ESRI	Environmental Systems Research Institute
GIS	Geografický inf. systém – Geographic information system
GNU	Prjekt zaměřený na svobodný software – GNU's Not Unix (rekurzivní zkratka)
GPL	Všeobecná veřejná licence – General public license
GPS	Globální polohový družicový systém – Global Positioning System
GUI	Grafické uživatelské rozhraní – Graphic User Interface
HTML	Jazyk pro WWW stránky – Hypertext Markup Language
HTTP	Protokol pro přenos na internetu – Hypertext Transfer Protocol
IFrame	Vnořený plovoucí rámeček
MySQL	Databázový systém
NASA	Americká agentura zodpovědná za kosmický program a výzkum v letectví – National Aeronautics and Space Administration
PARC	Výzkumné centru Paola Alta – Palo Alto Research Center
PHP	Hypertextový procesor – Hypertext Preprocessor
RGB	Aditivní model založený na míchání červené, zelené a modré barvy
WWW	Celosvětová síť - World Wide Web
XML	Rozšířený značkovací jazyk – eXtensible Markup Language

SEZNAM PŘÍLOH

A Vývojový diagram zobrazení mapy	50
B Popis souborů na médiu	51
C Softwarová konfigurace	53

A VÝVOJOVÝ DIAGRAM ZOBRAZENÍ MAPY



B POPIS SOUBORŮ NA MÉDIU

Na přiloženém médiu jsou (kromě elektronické verze práce) umístěny zdrojové kódy realizované aplikace. Tato příloha slouží ke snadné orientaci v nich.

V adresáři **Kyber-maps** se nachází tyto složky a soubory:

- Adresář **admin** – obsahuje soubory:
 - *.htaccess* – definuje přístupovou relaci.
 - *.htpasswd* – zahrnuje přístupová jména a hesla.
 - *index.php* – PHP zápis administrátorského režimu aplikace.
 - *upload.php* – jednodušší verze administrátorského režimu spolupracující se „Slicing tool“.
 - *vypis.php* – generuje výpis mapových setů z databáze.
- Adresář **css** – obsahuje soubor *homestyle.css*. Kaskádové styly v něm formátují vizuálně generovanou stránku.
- Adresář **images** – obsahuje grafické soubory, které jsou využívány pro generování kontextových tipů, sítě mapy a samotné stránky. Je zde také zastoupen soubor *logo.jpg* s logem projektu Kyber-maps.
- Adresář **library** – obsahuje tyto soubory:
 - *_begin.php* – tento PHP kód se importuje na začátku HTML kódu obou režimů.
 - *_end.php* – obdobný účel. Importuje se na konci.
 - *bubbles.js* – JavaScriptový soubor s Ajaxovými funkcemi pro kontextové tipy.
 - *bubbles.php* – skript, využívaný Ajaxem k přístupu do databáze.
 - *class.mysql.php* – knihovna přístupových funkcí k databázi MySQL pomocí PHP skriptu.
 - *constants.php* – definice PHP konstant.
 - *functions.php* – definice funkcí využívaných aplikací.
 - *init.php* – inicializační knihovna.
 - *plczip.lib.php* – knihovna pro práci s archivovanými ZIP soubory.
 - *validation.js* – JavaScript určený k validaci údajů.
 - *vypis_mest.php* – generuje výpis měst v jednotlivých mapových setech.

- Adresář **maps** – na tomto médiu obsahuje složku **hodonin** s mapou obce Hodonín a jejího okolí.
- Adresář **mysql** – ve skriptu *mysql.sql* je posloupnost příkazů pro vytvoření a obnovení struktury databáze MySQL.
- Soubor *index.php* – PHP zápis uživatelského režimu aplikace.

V adresáři **Kyber-maps slicing tool** se nachází tyto soubory:

- *ICSharpCode.SharpZipLib.dll* – knihovna pro práci s archivy ZIP. Nutná k spuštění aplikace.
- *k-maps_tool.exe* – samotná zkompilevaná aplikace.
- *Slicing tool.sln* – projektový soubor pro Microsoft Visual C#. Vývojové prostředí po jeho otevření načte další potřebné prvky z adresáře **Kyber maps slicing tool**. Není proto potřeba se mu blíže věnovat.

C SOFTWAREOVÁ KONFIGURACE

Server:

- Databáze **MySQL 5.0.15**.
- Grafické prostředí pro správu databáze **phpMyAdmin 2.5.3-rc3**.
- Skriptovací jazyk **PHP 5.2.8**.
- Textový a kódový editor **PSPad 4.5.3**.
- Webový prohlížeč **Opera 9.62**.
- Operační systém **Microsoft Windows XP SP3**.

Pracovní stanice:

- Textový a kódový editor **PSPad 4.5.3**.
- Webové prohlížeče **Opera 9.62** a **Firefox 3.0.10**.
- Vývojové prostředí **Microsoft Visual C# 2008**.
- **.NET Framework 3.1**
- Operační systém **Microsoft Windows Vista SP1**.