

Mendelova univerzita v Brně
Provozně ekonomická fakulta

Aplikace zpracovávající geografická data pro využití ve veřejné správě

Bakalářská práce

Vedoucí práce:
Ing. Pavel Turčínek, Ph.D.

Ondřej Šlégl

Brno 2017

Rád bych poděkoval Ing. Pavlu Turčínkovi, Ph.D. za skvělé vedení, cenné rady a zapůjčenou literaturu. A dále Ing. Jaromírovi Landovi, Ph.D. za podnětné připomínky a náměty.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Aplikace zpracovávající geografická data pro využití ve veřejné správě**

vypracoval samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací*.

Jsem si vědom, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.

Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

Brno 3. ledna 2017

.....

Abstract

Šlégl, O. Application for processing geographical data for use in public administration. Bachelor thesis. Brno: Mendel University, 2017.

The bachelor thesis deals with the development of web application for playgrounds in Brno. The partial analysis of similar applications in the Czech Republic is presented in the introductory part. Each phase of the application starting from the data collection to the final implementation of the application is explained in the following parts.

Keywords

Playgrounds, spatial data, web application, Leaflet, GIS

Abstrakt

Šlégl, O. Aplikace zpracovávající geografická data pro využití ve veřejné správě. Bakalářská práce. Brno: Mendelova univerzita, 2017.

Bakalářská práce se zabývá tvorbou webové aplikace pro dětská hřiště v Brně. V úvodní části je provedena analýza podobných aplikací v České republice. Dále jsou vysvětleny jednotlivé fáze vývoje aplikace od sběru dat až po konečnou implementaci aplikace.

Klíčová slova

Dětská hřiště, prostorová data, webová aplikace, Leaflet, GIS

Obsah

1	Úvod práce	6
1.1	Úvod	6
2	Cíl práce	7
2.1	Cíl	7
3	Literární přehled	8
3.1	Analýza podobných aplikací	9
	VenKaždýDen.cz - dětská hřiště v Česku	9
	JakNaMiminka.cz - Dětská hřiště v ČR	10
	HristePraha.cz - dětská hřiště Praha	10
	Webové stránky jednotlivých městských částí Brna	11
	Srovnání aplikací	12
3.2	Relační databáze	13
	Představení pojmu	13
	Terminologie relačních databází	13
3.3	Prostorové databáze	13
	Představení pojmu	13
	Odvětví, využívající data v prostorových databázích	14
	Prostorové datové typy	14
	Prostorové funkce	15
4	Metodika	16
4.1	Fáze řešení	16
	První fáze - sběr dat	16
	Druhá fáze - výběr databáze a zavedení dat	16
	Třetí fáze - implementace webové aplikace	17
4.2	Frameworky a použité technologie	17
	HTML5	17
	CSS	18
	Bootstrap	18
	JavaScript	19
	Leaflet	20
	Carto	21
4.3	Použité programové nástroje	21
	CARTO SQL editor	21
	XAMPP	22
	Notepad++	22
	WinSCP	22
	Internetové prohlížeče	23

5	Vlastní práce	24
5.1	Databáze	24
	Popis atributů	24
5.2	Frontend aplikace	25
	Úvodní stránka a bar pro navigaci	25
	Mapa hřišť	25
	Formulář pro přidání nového hřiště	26
5.3	Backend aplikace	27
	Načtení mapy	27
	Připojení k databázi	27
	Vykreslení záznamů do mapy	28
	Kontrola číselného kódu ve formuláři	29
	Vložení nového záznamu přes formulář do databáze	29
	Automatické vyplnění souřadnicových polí	29
6	Diskuze	31
6.1	Možná rozšíření	31
6.2	Nevýhody aplikace a problémy při vývoji	31
7	Závěr	32
8	Reference	33

Seznam obrázků

1	Úvodní stránka webu venkazdyden.cz (Matějka, 2016b).	9
2	Hřiště zobrazená na mapě webu jaknamiminka.cz (JakNaMiminka, 2016).	10
3	Informace o jednom hřišti na webu hristepraha.cz (Vašák, 2014b).	11
4	Bootstrap grid systém (Bootstrap, 2016b).	19
5	Ukázka značky, vytvořené pomocí knihovny Leaflet (Agafonkin, 2015b).	20
6	Carto SQL editor (vpravo) a jeho výstup (vlevo)	21
7	Prosředí editoru notepad++.	22
8	Úvodní stránka a navigační bar.	25
9	Mapa hřišť s informacemi.	26
10	Formulář pro přidání hřiště.	27
11	Diagram přenostu dat mezi klientem a databází prostřednictvím PHP souboru.	28
12	Okno se souřadnicemi vyplňuje formulář.	30

Seznam tabulek

1	Relační databáze jako tabulka zákazníků (Ritchie, 2002b)	13
2	Prostorové datové typy (Rigaux, 2001)	14
3	Atributy v databázi	24

1 Úvod práce

1.1 Úvod

Využití geografických informačních systémů ve veřejné správě je ve městech s rozsáhlou infrastrukturou obrovský. Města z velké části pracují právě s prostorovými daty, tedy s daty, vztahující se k jejich území. Mohou to být data jako například dopravní značky, reklamní billboardy, pískoviště a dětská hřiště, nákupní centra, kontejnery na tříděný odpad a mnoho dalších. Všechna tato data spadají pod správu města a to rozhoduje o jejich umístění do vhodných lokací, zda funkce, kterou vykonávají, postačuje na množství lidí v dané lokalitě a jestli mají lidé dost informací o těchto místech a mohou je bez zábrán využívat. K těmto všem rozhodnutím napomáhají veřejné správě měst právě geografické informační systémy.

Město Brno se dělí na 29 městských částí. Každá část si data na svém území spravuje samostatně. Problémem ale bývá, že ne všechna data jsou aktuální, některá chybí nebo jsou těžko dohledatelná na stránkách městských částí a navíc, je stejný typ dat jednoho města rozdělen do 29 různých míst.

Člověk, který chce například se svým dítětem navštívit hezká hřiště v Brně, musí nejdříve pročíst všechny stránky městských částí. Zjistit, jestli se zde alespoň nějaký seznam nachází a udělat si přehled, kde je jaké hřiště. Dále často neví, jestli informace, které vyhledal, jsou aktuální, jestli hřiště není zavřené, jak vůbec vypadá apod.

Lidé, pracující například s webovou mapou očekávají, aby informace, které potřebují, byly jednoduše a rychle dohledatelné, aktuální a aplikace od nich nevyžadovala žádné předešlé zkušenosti k obsluze. Některé aplikace však kladou na uživatele určité nároky, aby si data, která vyžaduje, mohl obstarat. Může to být například nutnost zdlouhavé registrace pro otevření práv nebo jsou určitá data nezobrazitelná uživateli bez zvláštního typu přístupu.

Práce se zaměřuje na zmapování a prezentaci dětských hřiště v Brně. Důvodem výběru bylo, že podobné a velmi dobré aplikace již existují, ale pouze pro jiná města, například Prahu. Nebo jsou udělána pro celý stát, což má však za následek méně podrobná či neaktuální data nebo chybějící lokace vzhledem k velké rozloze, kterou se snaží zahrnout. Byla snaha o přívětivé a jednoduché uživatelské rozhraní a co nejmenší nároky a omezení práv uživatele v aplikaci.

2 Cíl práce

2.1 Cíl

Cílem práce je v první řadě analyzovat a srovnat stávající přístupy, které se zabývají zpracováním a prezentací dětských hřišť v České republice. V druhé fázi pak s nasbíranými daty z oblasti Brno-město implementovat funkční webovou aplikaci.

Aplikace by měla poskytovat informace návštěvníkům dětských hřišť v Brně a může i pomoci městským částem s rozhodováním o lokacích pro výstavbu hřišť nových, či s aktualizací jejich stávajících dat.

Tato aplikace má za úkol co nejméně zatěžovat uživatele a umožňovat jim rychle vkládat přes formulář do interaktivní mapy nové lokace, popisy a případně fotografie dětských hřišť.

3 Literární přehled

Vývoj a samotnou implementaci práce výrazně ovlivnily a inspirovaly některé jiné aplikace pro dětská hřiště, ať už v Brně, nebo jiném českém městě. Bude zde popsáno chování, vzhled, uživatelská přívětivost a několik dalších parametrů typických pro danou aplikaci. Následně bude provedeno srovnání a případné odlišnosti všech přístupů.

V další části jsou pak vysvětleny dílčí pojmy, se kterými aplikace pracuje a je odkázáno na příslušnou literaturu a zdroje, které je zavádí.

3.1 Analýza podobných aplikací

VenKaždýDen.cz - dětská hřiště v Česku

Pravděpodobně největší portál dětských hřišť v České republice. Mobilní aplikace k dispozici pro Android a Windows.

Windows aplikace je univerzální, což znamená, že funguje jak na telefonu, tak na počítači či tabletu s Windows 10. Verze pro iPhone je plánována do konce roku 2016. (Matějka, 2016a). Jako podkladová mapa pro tuto aplikaci slouží openstreetmap.org.

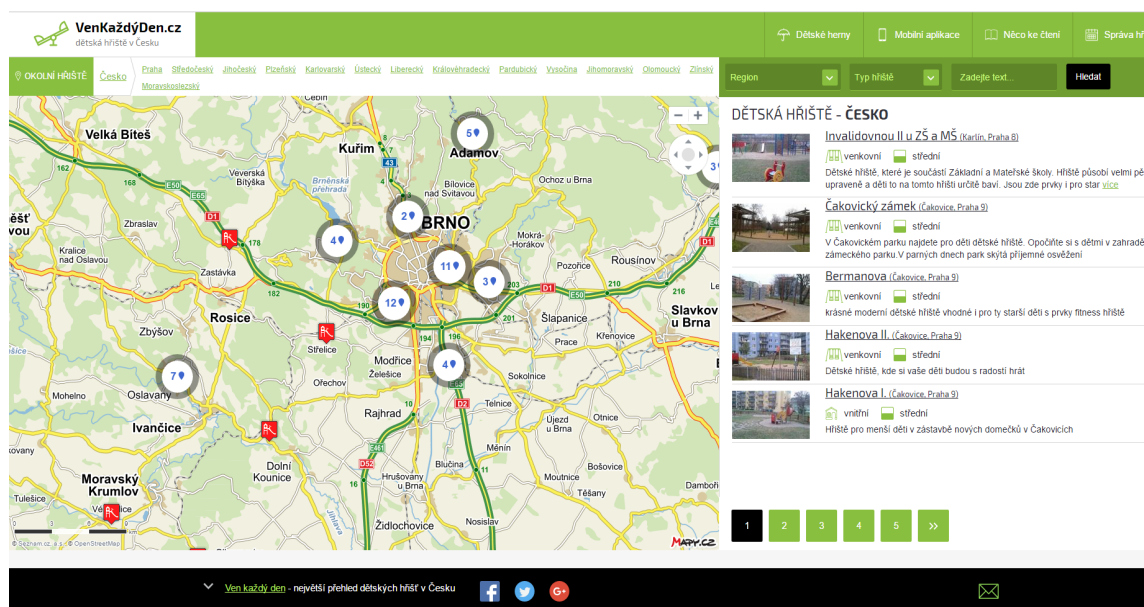
Vzhled a funkce: Jednotlivé ukazatele hřišť se přizpůsobují podle oddálení na mapě. Po kliknutí na tento ukazatel se zobrazí odkaz, který přesměruje uživatele na stránku s informacemi o daném hřišti. Kromě informací zde občas bývá fotografie a možnost přidání komentáře a hodnocení.

Hřiště lze zobrazovat podle výchozí polohy uživatele či řadit podle krajů. Přidání nového hřiště funguje vyplněním formuláře s informacemi k hřišti a zasláním údajů administrátorovi k potvrzení či zamítnutí.

Práva uživatelů: Neregistrovaný uživatel má možnost prohlížet si hřiště, přidávat komentáře a hodnocení. Přidávat nová hřiště mohou pouze registrovaní uživatelé.

Za přidání nového hřiště je možnost peněžní odměny. To však vyžaduje vyplnění dalších údajů, jako číslo bankovního účtu.

Webová stránka: <https://www.venkazdyden.cz>



Obrázek 1: Úvodní stránka webu venkazdyden.cz (Matějka, 2016b).

JakNaMiminka.cz - Dětská hřiště v ČR

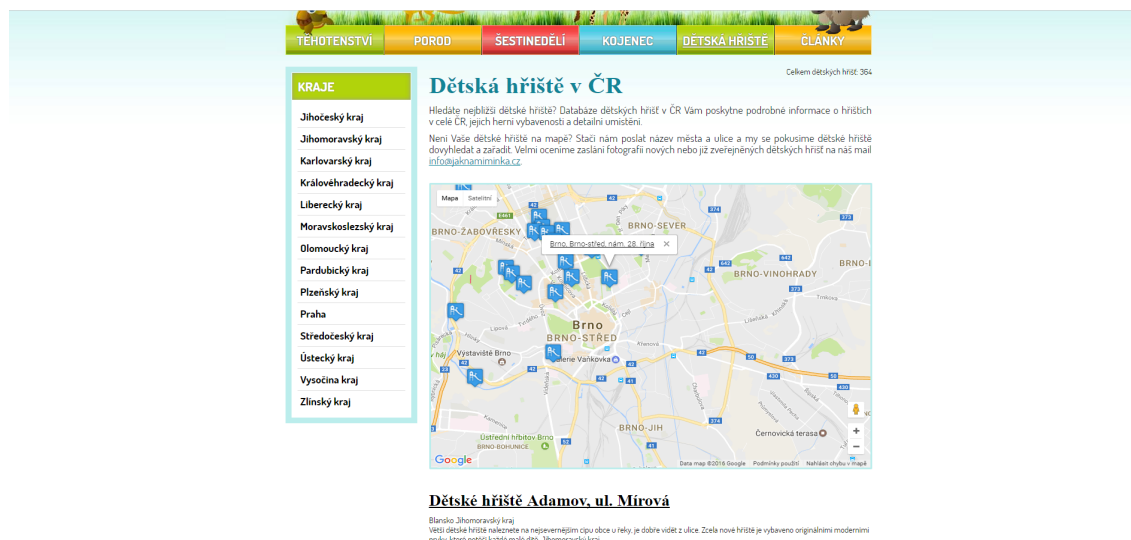
Databáze dětských hřišť na území ČR. Hřiště jsou pouze jednou ze sekcí celé webové stránky. Podkladovou mapu je Google Maps/Google Earth.

Vzhled a funkce: Je možné zvolit mapu terénní nebo satelitní. Značky hřišť odkazují na stránky s informacemi, polohou, někdy fotogalerií. Přidání nového hřiště je podmíněno kontaktem administrátora stránek na uvedenou adresu.

Po zaslání adresy a případně fotografie je hřiště vyhledáno a přidáno lidmi, kteří stránku udržují.

Práva uživatelů: Uživatelé se neregistrují ani nepřihlašují. Všichni mohou procházet hřiště a přidávat nové.

Webová stránka: <http://www.jaknamiminka.cz/detska-hriste/>



Obrázek 2: Hřiště zobrazená na mapě webu jaknamiminka.cz (JakNaMiminka, 2016).

HristePraha.cz - dětská hřiště Praha

Dětská hřiště na území Prahy. Aplikace se však nesnaží zobrazit úplný výčet všech hřišť, ale pouze vybírá ty nejzajímavější. Podkladová mapa Google Maps/Google Earth.

Hřiště blízko sebe jsou zpracovávány v rámci jedné lokality a jsou kromě čísla navíc označeny písmenem, např. 4.A a 4.B. (Vašák, 2014a)

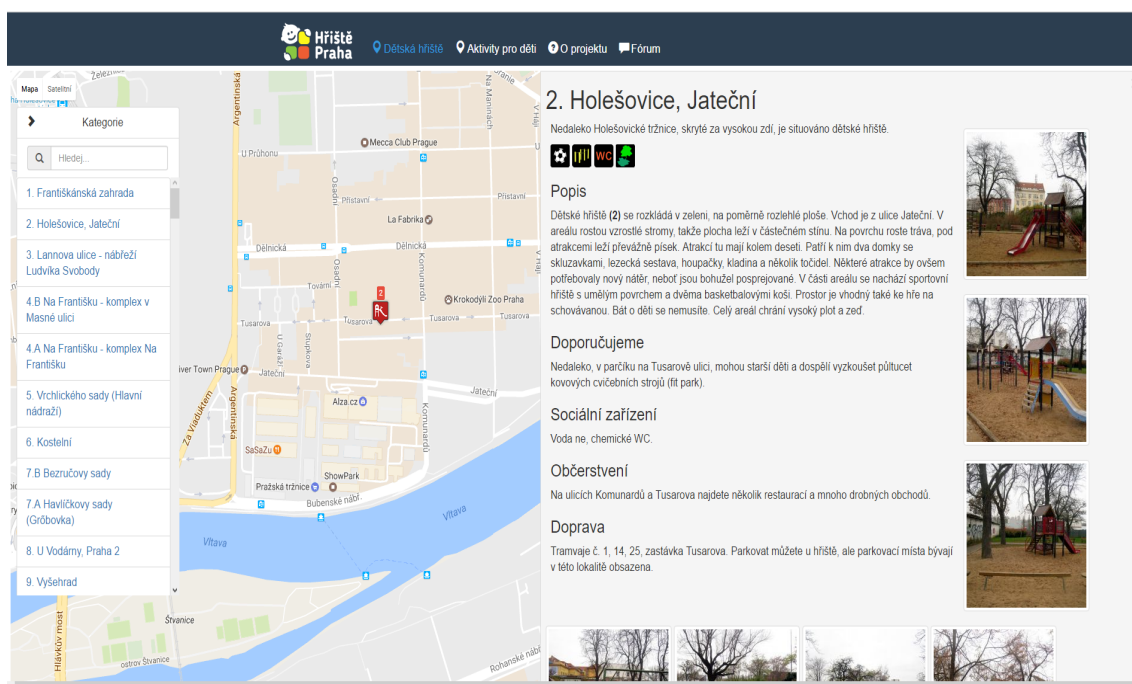
Vzhled a funkce: Mapa je zobrazitelná jako satelitní snímky nebo terénní mapa

s ulicemi. Po kliknutí na značku hřiště se přímo na stránce zobrazí v boční liště informace a fotogalerie o této lokaci. Aplikace také umožňuje vyhledávání hřišť podle kritérií.

Pro přidání nového hřiště je nutné vyplnit povinné položky s popisem, následně zjistit souřadnice a napsat je ve formátu šířka/délka do formuláře a vložit hřiště do systému pod uživatelskou emailovou adresou. Souřadnice jsou potřeba získat z jiné aplikace, například google maps.

Práva uživatelů: Uživatelé se neregistrují a nepřihlašují. Všichni mohou procházet hřiště a přidávat nové. Pouze u přidání nového hřiště je potřeba uvést svojí emailovou adresu.

Webová stránka: <http://www.hristepraha.cz>



Obrázek 3: Informace o jednom hřišti na webu hristepraha.cz (Vašák, 2014b).

Webové stránky jednotlivých městských částí Brna

Každá městská část má svoje webové stránky, na kterých můžou, ale i nemusí být informace, kde se nacházejí dětská hřiště na jejich území. Městské části poskytují informace svým občanům odlišným způsobem.

Vzhled a funkce: Informace polohy hřišť je zobrazována těmito způsoby:

1. Mapa - na stránkách je přímo mapa s body hřišť. Například Židenice nebo Líšeň.
2. Textový seznam hřišť - hřiště a jejich polohy jsou vypsané v samostatném souboru ke stažení nebo přímo na stránce. Královo Pole, Bystrc, Kohoutovice.
3. Seznam se na stránce nenachází - jednotlivá místa, kde jsou umístěny dětská hřiště, se na stránce nenachází. Vinohrady, Bohunice.

Práva uživatelů: Uživatelé se neregistrují a nepřihlašují. Je umožněno pouze prohlížení aktuálních hřišť, nikoliv přidávání nových.

Srovnání aplikací

Jednotlivé aplikace se od sebe odlišují především způsobem a náročností přidání nových hřišť ze strany uživatelů. Dále množstvím funkcí, zobrazitelných informací o jednotlivých místech a množstvím dat.

Množství dětských hřišť Pokud se podíváme na množství dat, největší databázi má *VenKazdyDen.cz*. Obsahuje zhruba 2700 hřišť v celé České republice. *JakNaMiminka.cz*, které zpracovává také celé naše území, obsahuje pouze 364 lokací. Na druhou stranu velké množství hřišť je v aplikaci *VenKazdyDen.cz* soustředěno do jedné oblasti.

Můžeme srovnat dvě největší města Prahu a Brno. Praha obsahuje cca 600 lokací zatímco Brno pouze 32. Je vidět že zmapování hlavního města je zde mnohem podrobnější. *HristePraha.cz* má asi 130 záznamů, ale ani se nesnaží o zmapování všech, pouze vybírá ty zajímavé. Městské části, pokud nabízejí mapu nebo seznam, většinou obsahují všechny známé lokace svého území.

Požadavky na uživatele Přidání hřišť od uživatelů *VenKazdyDen.cz* odměňuje symbolickou peněžní částkou. To však sebou přináší i určité nevýhody, jako uvedení v registraci nejenom jména a hesla, ale i vložení dalších doplňujících údajů, jako číslo bankovního účtu. Dále to výrazně zpomaluje proces a uživatel déle čeká na schválení ze strany správce.

U *HristePraha.cz* je proces přidání hřiště o něco kratší. Při vyplňování formuláře je však stále požadováno velké množství povinných informací o lokalitě. Je také nutné dohledávat souřadnice jinde, než na stránce samotné.

JakNaMiminka.cz má proces přidání hřiště nejméně náročný ze strany uživatele. Je pouze odeslán email s adresou a případně fotografií. U všech přístupů však záleží na tom, jak často správci stránek aktualizují svá data a čtou nové příspěvky.

Funkce a možnosti aplikací Nejrozsáhlejší funkcionalitu má pravděpodobně *VenKazdyDen.cz*. Je zde možné zobrazovat hřiště v blízkém okolí od své pozice, vyhledávat podle regionu a typu, komentovat, hodnotit či sdílet hřiště a mnoho dalšího.

Velmi dobré možnosti má i aplikace *HristePraha.cz*. Umožňuje nám přepínání mapových podkladů, vyhledávání podle kategorií a adres a zobrazení informací a fotogalerie o lokalitách hned vedle, v příjemné postranní liště. *JakNaMiminka.cz* nabízí

pouze přepínání mapových podkladů a jednoduchý odkaz na popis lokalit a fotogalerie.

U městských částí, kde je nabízena interaktivní mapa a můžeme očekávat alespoň nějakou funkcionalitu, většinou najdeme základní operace s mapou, pár různých mapových podkladů a někdy několik fotografií.

3.2 Relační databáze

Představení pojmu

Nejjednodušší pohled na relační databáze je, jakožto na reprezentaci aplikačních dat v tabulce se dvěma prostory. Sloupce reprezentující typ dat a řádky samotná data. (Ritchie, 2002a) Za relační databáze podle (Harrington, 2002) můžeme považovat místo, kde jsou uloženy data samotná a zároveň jsou zde i informace o vzájemných vztazích mezi těmito daty.

Terminologie relačních databází

Tabulka 1: Relační databáze jako tabulka zákazníků (Ritchie, 2002b)

ID zákazníka	Jméno	Adresa	Dlužný zůstatek
5567	Jones	Cross Road	2.50
2913	Anderson	River Lane	0.00
4890	Murray	West Street	1.50
1622	Richards	Mill Lane	3.00

Každý sloupec tabulky má nadpis jako například Jméno, Dlužný zůstatek atd. To nám specifikuje, co bude celý sloupec obsahovat. Nadpis není součástí dat samotných.

Tabulka je někdy také označována jako relace, sloupce jako atributy a řádky jako záznamy.

V každé tabulce se většinou nachází jeden nebo i více speciálních sloupců s unikátní hodnotou, které fungují jako identifikátory jednotlivých řádků. Tato množina sloupců se označuje jako *primární klíče*. V tabulce zaměstnanců je to například ID zákazníka. (Ritchie, 2002c)

3.3 Prostorové databáze

Představení pojmu

Prostorové databáze jsou systémy, které nabízí prostorové datové typy a jejich implementaci. Poskytují minimálně prostorovou indexaci a spojující funkce. (Güting,

1994).

Jsou rozšířením relačních databází, určené pro manipulaci s prostorovými daty. (Arctur, 2004) Prostorové databáze uchovávají a manipulují s prostorovými objekty, stejně jako s jakýmkoli jiným objektem v databázi. (Boundless, 2015a)

Odvětví, využívající data v prostorových databázích

Příklady odvětví lidí podle (Shekhar, 2003), kteří používají prostorová data:

1. **Uživatel mobilního zařízení** - nejbližší benzínová stanice od aktuální pozice.
2. **Armádní velitel** - pohyb nepřátelských jednotek.
3. **Pojišťovací úředník** - potenciální objekty, které mohou být v budoucnu zasaženy přírodní katastrofou.
4. **Klimatolog** - testování nového modelu pro globální oteplování.
5. **Manažer logistiky** - určení nejlepších míst pro nová skladiště.
6. **Záchranná služba** - nejkratší cesta k raněnému.
7. **Zemědělec** - minimalizace ujeté vzdálenosti kombajnů, při setí obilí.

Prostorové datové typy

Obyčejná databáze umožňuje ukládat textové řetězce, čísla a časové údaje. Prostorová databáze přidává další typy pro reprezentaci geografických rysů (Boundless, 2015b).

Tabulka 2: Prostorové datové typy (Rigaux, 2001)

Typ geometrie	Reprezentace	Popis
Bod	(x,y)	Bod v prostoru
Linie	$((x1, y1), (x2, y2))$	Nekonečná přímka definovaná dvěma body
Kružnice	$\langle (x, y), r \rangle$	Kruh se středem a poloměrem
Polygon	$((x1, y1), \dots)$	Mnohoúhelník

Bod (point) - Bod je objekt reprezentující určitou lokaci a obsahuje x (zeměpisná šířka) a y (zeměpisná délka). Tyto hodnoty jsou udávány ve stupních, minutách a vteřinách nebo jako desetinné číslo. Pro stupně leží hodnoty zeměpisné šířky vždy v intervalu $[-90, 90]$. Hodnoty pro zeměpisnou délku leží v intervalu $[-180, 180]$. (Microsoft, 2015)

Linie (line) - Linie je objekt v jednorozměrné rovině, reprezentující sekvenci bodů a čáru, která je spojuje.

Kružnice (circle) - Křivka reprezentovaná třemi body v dvourozměrné rovině. Vzdálenost mezi jakýmkoliv bodem a středem se nazývá poloměr.

Mnohoúhelník (polygon) - Polygon je dvourozměrný povrch, znázorněn jako sekvence bodů, určující vnitřek a vnějšek. Bývá vytvořen z objektu, který má alespoň tři rozdílné body. Objekt polygonu může být také prázdný. Vnější a vnitřní objekty polygonu definují jeho hranice.

Prostorové funkce

Kromě jednořádkových a víceřádkových funkcí prostorové databáze přidávají další, tzv. prostorové funkce. Většinu prostorových funkcí lze rozdělit do pěti kategorií. (Boundless, 2015c)

Konverzní funkce - Funkce, které převádějí mezi externími datovými formáty a geometriemi. Př. (PostGIS, 2015):

- `ST_AsText(geometry)` - vrací text
- `ST_AsBinary(geometry)` - vrací binární data
- `ST_GeomFromKML(text)` - vrací geometrický prvek

Funkce řízení - Spravují informace o prostorových tabulkách a provádějí PostGIS administraci.

- `AddGeometryColumn` - přidává sloupec do existující tabulky atributů

Výpočtové funkce - Počítají rozměry a získávají vlastnosti geometrických objektů.

- `ST_Area(geometry)` - vrací plochu polygonu.
- `ST_Perimeter(geometry)` vrací obvod všech hranic.

Porovnávací funkce - Porovnávají dvě a více geometrií s ohledem na prostorový vztah.

- `ST_Equals(geom A, geom B)` - testuje, zda jsou dva prvky shodné
- `ST_Overlaps(geom A, geom B)` - porovnává dva prvky stejných rozměrů. Vrací PRAVDU pokud je průnik rozdílný, ale shodných rozměrů, jako oba prvky.

Generující funkce - Funkce, které vytvářejí nové geometrie z jiných geometrií.

4 Metodika

4.1 Fáze řešení

Pro dosažení výsledku se postupovalo v několika fázích. Jednotlivé kroky byly rozdílné především v časové náročnosti a složitosti řešení.

První fáze - sběr dat

Časově jednoznačně nejdelší fáze. Bylo nutno zjistit co nejvíce potřebných dětských hřišť v Brně, a to v odpovídajícím formátu. V prvním kroku byla provedena analýza všech 29 městských částí Brna a vyhodnoceno:

1. Jestli vůbec existují na stránce dané městské části nějaká data.
2. Pokud ano, v jakém formátu jsou tyto data prezentována. Např. pouze jako textový seznam nebo již body na mapě apod.
3. V jakém stavu jsou a posouzení jejich aktuálnosti.

V druhém kroku, po zhodocení a výběru vhodných dětských hřišť, proběhl kontakt některých městských částí. Bylo zažádáno o data pro potřeby této práce. Komunikace probíhala nejčastěji přes e-mail. Zde se vyskytl problém s poměrně malou komunikací a zpětnou vazbou ze strany úřadů a díky tomu se tato fáze značně protáhla. Více bude problém rozveden v kapitole **diskuze**.

Bylo zjištěno, že dětských hřišť, poskytnutých od městských částí není mnoho. Pokusil jsem se tedy některá chybějící hřiště obejít, zjistit nejdůležitější informace a vyfotografovat je. U fotografií se však vyskytl právní problém.

Nový občanský zákoník § 84 říká: *"Zachytit jakýmkoli způsobem podobu člověka tak, aby podle zobrazení bylo možné určit jeho totožnost, je možné jen s jeho svolením."* (Zákon č. 89, 2012).

Některé mnou pořízené fotografie zobrazovali úplně nebo částečně obličej lidí, nacházející se v době pořízení snímku na hřišti. U některých snímků se sice podařilo obličej dané osoby upravit tak, aby byla zajištěna její úplná anonymita, ale nešlo to vždy. Navíc, některé fotografie byly v poměrně nízké kvalitě.

Bylo učiněno rozhodnutí, že budoucí samotná aplikace bude schopna vkládat nová hřiště sama do databáze prostřednictvím uživatelů. To zaplní mezery v chybějících datech a zajistí jejich aktuálnost.

Druhá fáze - výběr databáze a zavedení dat

Nejdříve bylo zhodnoceno, jaký objem a typ dat se bude ukládat do databáze. Jednotlivá hřiště budou ukládána jako body v prostoru. Jak vyplývá z kapitoly **Rešerše - prostorové databáze** je tedy vhodné použít rozšíření databáze relační, tedy prostorovou.

Na výběr bylo poměrně velké množství prostorových databází. Byl zvolen systém Carto, nabízející databázi, vycházející z objektově-relační databáze PostgreSQL s rozšířením PostGIS. Obsluha a používání Carto je zcela zdarma. Jak píší licenční podmínky (Carto, 2012b), jsou tu však jistá omezení:

1. *Dovolená velikost údajů v tabulce je maximálně 1000 řádků nebo 20MB dat.*
Vzhledem k tomu, že se do databáze budou vkládat především textové údaje s malou velikostí a fotografie budou mít úložiště jinde, je velikost 20MB dostatečující.
2. *Uložená data musí být veřejná.*
Všichni uživatelé mohou vidět data vložená v databázi a údaje do ní vkládat. Zde to sice není takový problém, protože není možné data z vnějšku nijak editovat, je ale možné například použít robota pro zahlcení databáze. Vyvarování se něčemu podobnému je nutno ošetřit například ověřením, že uživatelem je skutečně člověk, a ne stroj.

Žádné omezení pro potřeby této aplikace nebylo nepřekonatelné a Carto bylo zvoleno jako vhodné řešení pro uchování dat.

Třetí fáze - implementace webové aplikace

Nakonec přišla samotná implementace webové aplikace. Díky tomu, že jak databáze, tak i potřebná data, už byly k dispozici, mohlo být testování funkcí aplikace prováděno přímo na reálných datech. Podrobný popis implementace viz část **Vlastní práce**.

4.2 Frameworky a použité technologie

V aplikaci bylo využito několik frameworků a řada knihoven. Výběr vhodných technologií záleží především na typu aplikace a pro koho je určená. Správná volba pak může výrazně zlepšit rychlost, přehlednost, design a funkčnost výsledné aplikace.

Pro vzhled a uživatelské rozhraní webové stránky byl použit jazyk **HTML5** s **CSS** styly ve frameworku **Bootstrap**. **JavaScript** a knihovna **Leaflet** zajišťovali vykreslení mapy, obslužné funkce s mapou apod. Jazyky **php** a **sql** byly využity především pro připojení a komunikaci s databází. Databáze samotná pak byla vytvořena v software **Carto**.

HTML5

HTML je zkratka pro HyperText Markup Language. Používá se ve vytváření elektronických dokumentů, které jsou zobrazovány na www stránkách. HTML zajišťuje rozložení textu a obrázků, aby je internetový prohlížeč dokázal správně zobrazit. Udává tedy strukturu každé webové stránky.

HTML5 je aktualizací k HTML4. Používá stejné základní pravidla, avšak přidává nové tagy a atributy, díky kterým je zajištěna lepší sémantika a ovládání dynamických elementů přes JavaScript (ComputerHope, 2016).

Ukázka rozložení tagů v HTML5:

```
1 <html>
2 <head>
3 <meta charset="utf-8">
4 <title>Příklad stránky</title>
5 </head>
6 <body>
7 <h1>Nadpis první úrovně</h1>
8 <p>Text</p>
9 </body>
10 </html>
```

CSS

CSS je zkratka pro Cascading Style Sheets (kaskádové styly). Definuje rozložení a vzhled HTML dokumentu. Použitím CSS můžeme například kontrolovat barvu textu, fonty, řádkování mezi odstavci, velikost obrázků, celkový design apod. Většinou je CSS kombinován s HTML.

Na různé HTML stránky lze použít jeden, předem definovaný CSS styl a na více HTML tagů zase jedno CSS pravidlo. To šetří čas a navíc zrychluje načítání stránek eliminací značného množství kódu (Tutorialspoint, 2016).

Ukázka CSS v samostatném souboru:

```
1     background-color: blue;
2 }
3
4 h1 {
5     color: white;
6     text-align: center;
7 }
8
9 p {
10    font-family: verdana;
11    font-size: 20px;
12 }
```

Bootstrap

Bootstrap je open-source Javascript framework vytvořený a udržovaný týmem na Twitteru. Kombinuje HTML, CSS a Javascript kód, tak aby pomáhal vytvářet uživatelské rozhraní. Je to kolekce nástrojů pro vytváření webových stránek a aplikací. Obsahuje šablony pro formuláře, tlačítka, navigaci, stránkování, obrázky, tabulky a mnoho dalších komponent rozhraní.

Je to responzivní framework, může být tedy použit na širokou řadu platform. Přizpůsobuje se totiž automaticky velikosti okna zařízení, na kterém je webová stránka používající Bootstrap otevřena.

Používá tzv. grid systém. Webové stránky jsou většinou strukturovány jako sloupce dat - navigační sloupec vlevo a reklamní sloupec vpravo (*span 2 - span 8 - span 2*). Bootstrap grid systém dovoluje až 12 sloupců na stránce.

Existují čtyři typy grid systémů se zadávaným parametrem šířky (width):

- Extra malé zařízení - telefony (< 768px), zkratka col-xs-
- Malé zařízení - tablety (>= 768px), zkratka col-sm-
- Střední zařízení - desktopy (>= 992px), zkratka col-md-
- Velká zařízení - desktopy (>= 1200px), zkratka col-lg-

(Bootstrap, 2016a)

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

Obrázek 4: Bootstrap grid systém (Bootstrap, 2016b).

JavaScript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk. Uvnitř svého prostředí např. webového prohlížeče, se JavaScript napojuje do HTML kódu stránky a zajišťuje programové ovládání jednotlivých prvků např. tlačítek, animací, různých efektů na stránce apod.

Obsahuje standardní objektovou knihovnu, v níž nalezneme např. pole, operátory, funkce pro práci s daty, matematické funkce apod. Tento základ je však rozšířen ještě o další funkce:

JavaScript na straně klienta přidává navíc objekty pro kontrolu prohlížeče a jeho document object model (objektový model dokumentu). Toto rozšíření zajišťuje například aplikaci, aby si ukládala údaje z HTML formuláře a následně reagovala daným způsobem na události jako kliknutí myši, odeslání formuláře přes tlačítko submit apod.

JavaScript na straně serveru aplikacím nabízí možnosti a funkce pro komunikaci s databázemi, poskytuje manipulaci se soubory nebo zajišťuje informační tok mezi aplikacemi. (Patonnier, 2016)

Ukázka JavaScriptu v samostatném souboru:

```
1 var popup = L.popup();
2
3 function onMapClick(e) {
4     popup
5     .setLatLng(e.latlng)
6     .setContent("Poloha: " + '' + popup.getLatLng().lat.toString() +
7         '' + popup.getLatLng().lng.toString())
8     .openOn(map);
9     document.getElementById('latitude').value = popup.getLatLng().lat;
10    document.getElementById('longitude').value = popup.getLatLng().lng;
11 }
12 map.on('click', onMapClick);
13 };
```

Leaflet

JavaScriptová open-source knihovna pro vytváření interaktivních map, jak pro osobní počítače, tak mobilní zařízení. Obsahuje většinu funkcí, potřebných pro vytváření map a je snadno rozšiřitelná množstvím samostatných modulů.

Další výhodou je velmi dobrá a obsáhlá dokumentace a dobře čitelný kód. (Agafonkin, 2015a)



Obrázek 5: Ukázka značky, vytvořené pomocí knihovny Leaflet (Agafonkin, 2015b).

Carto

Carto je cloudová platforma, poskytující nástroje pro vytváření webových map a vizualizaci prostorových dat. (GIS, 2016)

Je to model Software as a Service (software jako služba), což znamená, že celá aplikace je hostována provozovatelem služby. Není tedy potřeba instalace a udržování ze strany zákazníka na vlastním zařízení. (Gartner, 2016) Carto je nabízena jako freemium služba. Účty jsou nejdříve zcela zdarma, ale od určité velikosti, například množství dat v databázi, je vyžadován poplatek.

4.3 Použité programové nástroje

CARTO SQL editor

Součástí platformy Carto. Nástroj pro manipulaci s databází pomocí SQL příkazů. Můžeme filtrovat, řadit, vyhledávat atd. jakákoliv data v dané tabulce. Výsledek bude zobrazen, za předpokladu korektní SQL syntaxe, přímo v tabulce vedle. Nástroj má také velmi inteligentní našeptávač, který nám pomůže například pokud máme podobné názvy sloupců, nepamatuje si přesnou syntax apod.

address -
string
Slívova 1 615 00 Brno-Židenice
Krásného 37 636 00 Brno-Židenice
Nové náměstí 1-8 Brno-Řečkovice
Jubilejní 79 Brněnské Ivanovice, 620 00 Brno
Turgeněvova 6 618 00 Brno
Elišky Krásnohorské 20 618 00 Brno
U Velké ceny 6 623 00 Brno
Bubeničkova 28 615 00 Brno
Šámalova 9 615 00 Brno
Kosmákova 38 615 00 Brno-Židenice

```
SELECT address FROM playgrounds where cartodb_id=52 or confirmed=true
```

Obrázek 6: Carto SQL editor (vpravo) a jeho výstup (vlevo)

XAMPP

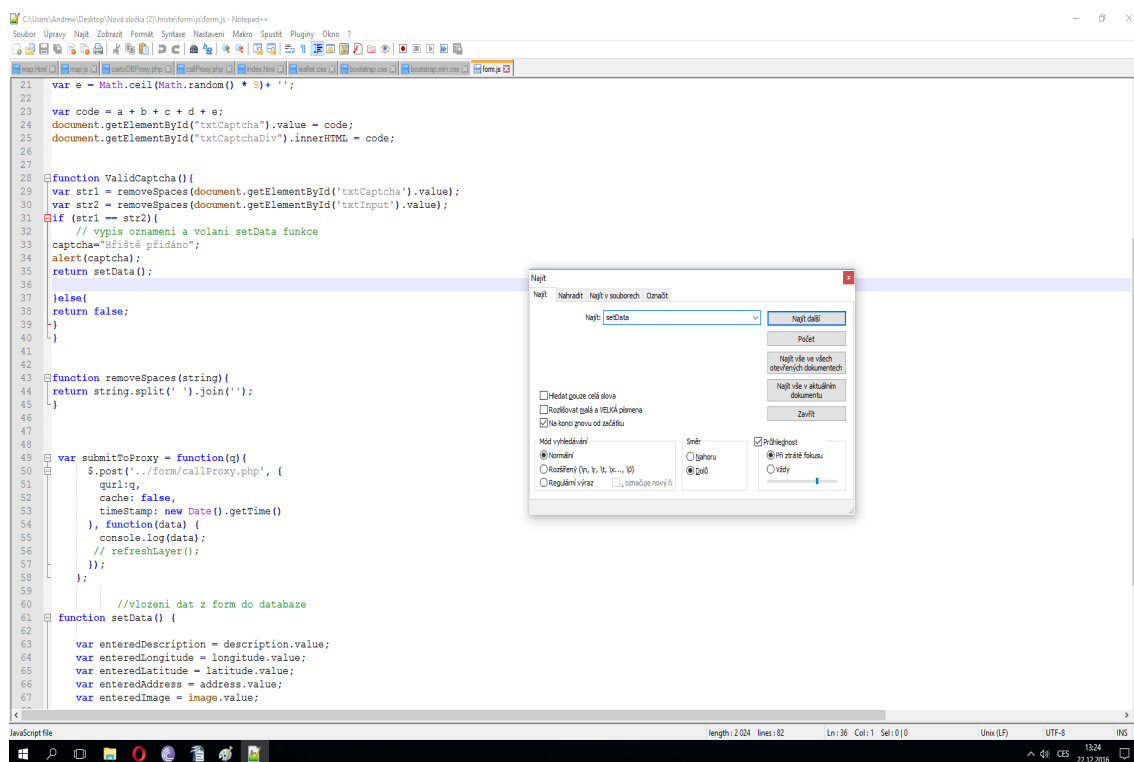
Nástroj pro vývojáře na vytvoření lokálního webového prostředí. XAMPP je zkratkou pro multi-platformní - funkční na všech OS (X), Apache - webový server (A), MySQL - databáze (M), PHP - skriptovací jazyk (P) a Pearl - programovací jazyk (P). (Mikoluk, 2013)

Byl používán především na testování, předtím než se části aplikace bez bugů a chyb přesunuly do prostředí akely.

Notepad++

Nástroj pro psaní a editaci zdrojového kódu i obyčejného textu v prostředí MS Windows. Je zcela zdarma a podporuje velké množství programovacích jazyků. Byl zvolen místo jiných editorů protože má jednoduché uživatelské rozhraní a dobré barevné značení syntaxe.

Umožňuje také editaci velkého množství souborů s rozdílnými příponami. Pracuje rychle a i velké projekty zabírají jen velmi málo paměťového prostoru.



Obrázek 7: Prosděří editoru notepad++.

WinSCP

WinSCP je open source klient pro Windows. Podporuje čtyři protokoly (WinSCP, 2016):

1. SFTP (SSH File Transfer Protocol)
2. FTP (File Transfer Protocol)
3. SCP (Secure Copy Protocol)
4. WebDAV (Web Distributed Authoring and Versioning)

Jeho hlavní funkcí je přenos souborů a dat mezi lokálním a vzdáleným serverem. Poskytuje také základní funkce pro správu souborů. (WinSCP, 2014)

Byl používán hlavně k přenosu hotových částí webové aplikace na školní server akela.

Internetové prohlížeče

Aplikace byla testována v prohlížečích:



Opera verze: 42.0.2393.85



Firefox verze: 50.1.0



Safari verze: 5.1.7



Chrome verze: 42.0.2311.90

5 Vlastní práce

5.1 Databáze

Tabulka 3: Atributy v databázi

Název atributu	Datový typ
cartodbId	number
theGeom	geometry
address	string
confirmed	boolean
description	string
image	string

Popis atributů

- **cartodbId** - primární klíč. Tímto číslem se můžeme odkázat na kterýkoliv záznam v databázi. Je unikátní, nemohou existovat dva záznamy se stejným číslem.
- **theGeom** - Geometrie. Tento sloupec Carto vytváří automaticky pro každou tabulku. Zde se ukládá typ geometrie pro záznam z výběru: bod, linie nebo mnohoúhelník (polygon). Můžeme mít pouze jeden typ geometrie pro daný sloupec. (Carto, 2012a) Tento sloupec také slouží pro ukládání souřadnic. Pokud pracujeme s body, souřadnice vkládáme v pořadí zeměpisná délka, zeměpisná šířka.

Určení typu a vložení souřadnic do sloupce theGeom:

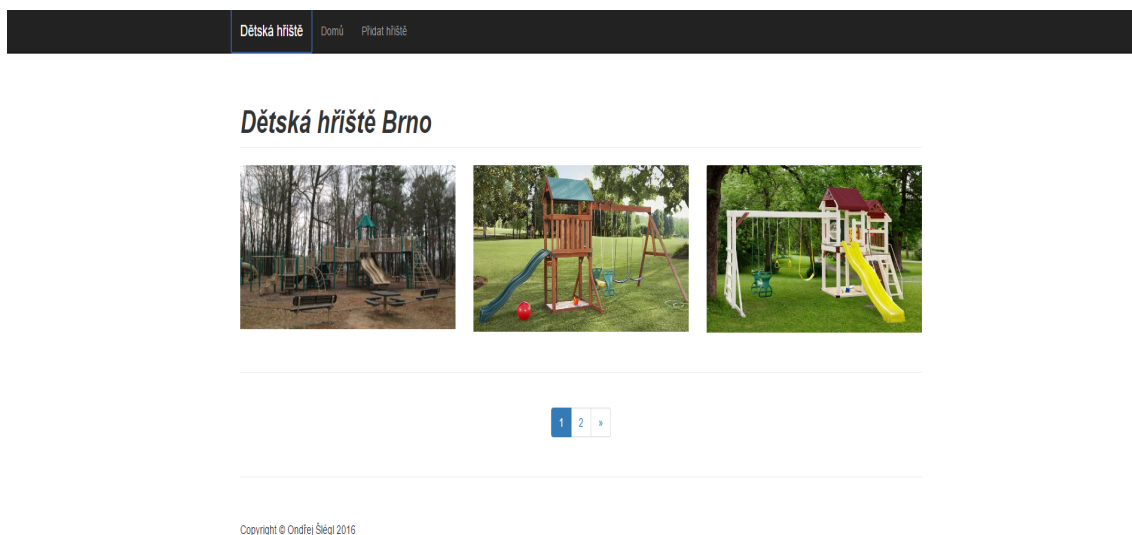
```
1 {"type": "Point",
2  "coordinates": [16.583715, 49.24822]}
```

- **address** - adresa dětského hřiště jako textový řetězec ve formátu: název ulice, číslo orientační, poštovní směrovací číslo, případně městská část.
- **confirmed** - potvrzení záznamu. Po vložení nového je defaultní hodnota nastavena na false. Může být později změněna.
- **description** - popis hřiště. Druh atrakcí, otevírací doba, oplocení, povrch atd. Může mít libovolnou délku. Jediné omezení je maximální velikost databáze viz kapitola **Metodika - výběr databáze a zavedení dat**.
- **image** - URL fotografií hřiště. Kvůli omezení velikosti databáze se obrázky nekládají přímo. Jsou nahrány na jiném úložišti a v databázi se ukládá pouze odkaz.

5.2 Frontend aplikace

Úvodní stránka a bar pro navigaci

Za pomoci šablony frameworku Bootstrap byla vytvořena úvodní stránka s navigací, což jsou odkazy na mapu dětských hřišť, úvodní stránku a formuláře přidání hřiště nového.

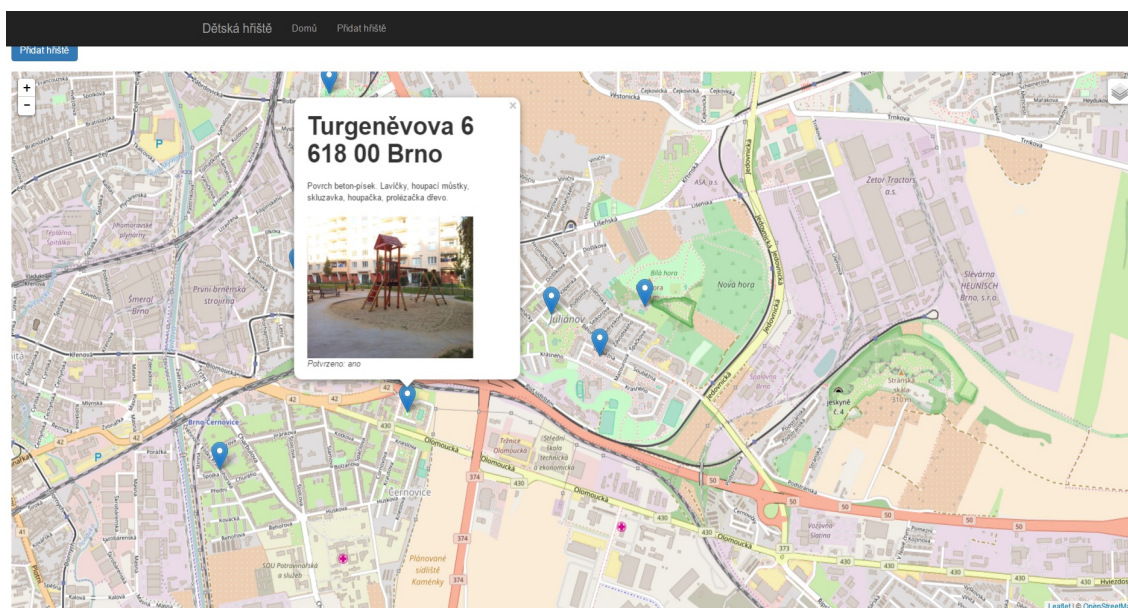


Obrázek 8: Úvodní stránka a navigační bar.

Mapa hřišť

Po mapě je možné se posouvat, přibližovat a oddalovat. Lze měnit mezi třemi mapovými podklady - základní, satelitní a bez popisků. Při kliknutí kamkoliv na mapu, mimo značky hřišť, se v okně zobrazí souřadnice této lokace.

Jakmile je rozkliknuto hřiště, v okně nad značkou se zobrazí informace adresy, popisu hřiště, fotografie a stav potvrzení hřiště.



Obrázek 9: Mapa hřišť s informacemi.

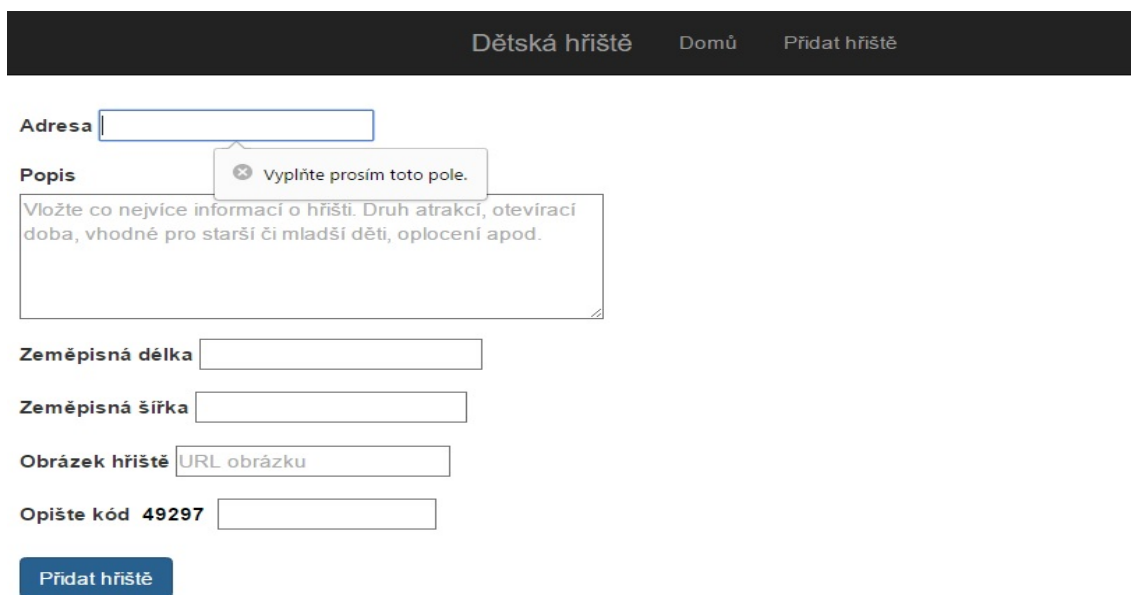
Formulář pro přidání nového hřiště

Uživatel může přidat nové hřiště po vyplnění několika povinných polí ve formuláři. Povinná pole jsou:

- Adresa
- Zeměpisná délka
- Zeměpisná šířka
- Číselný kód

Popis a fotografie hřiště mohou být vynečány, případně doplněny později. Po odeslání a kontroly správnosti kódu je hřiště ihned přidáno a vykresleno na mapě, zatím se statusem nepotvrzeno.

Uživatel také nutně nemusí dohledávat souřadnice a přesnou polohu hřiště. Je možné kliknout do mapy na místo, kde se budoucí hřiště nachází. Automaticky se pak vyplní pole souřadnic.



Dětská hřiště Domů Přidat hřiště

Adresa

Popis
Vložte co nejvíce informací o hřišti. Druh atrakcí, otevírací doba, vhodné pro starší či mladší děti, oplocení apod.

Zeměpisná délka

Zeměpisná šířka

Obrázek hřiště

Opište kód 49297

Obrázek 10: Formulář pro přidání hřiště.

5.3 Backend aplikace

Načtení mapy

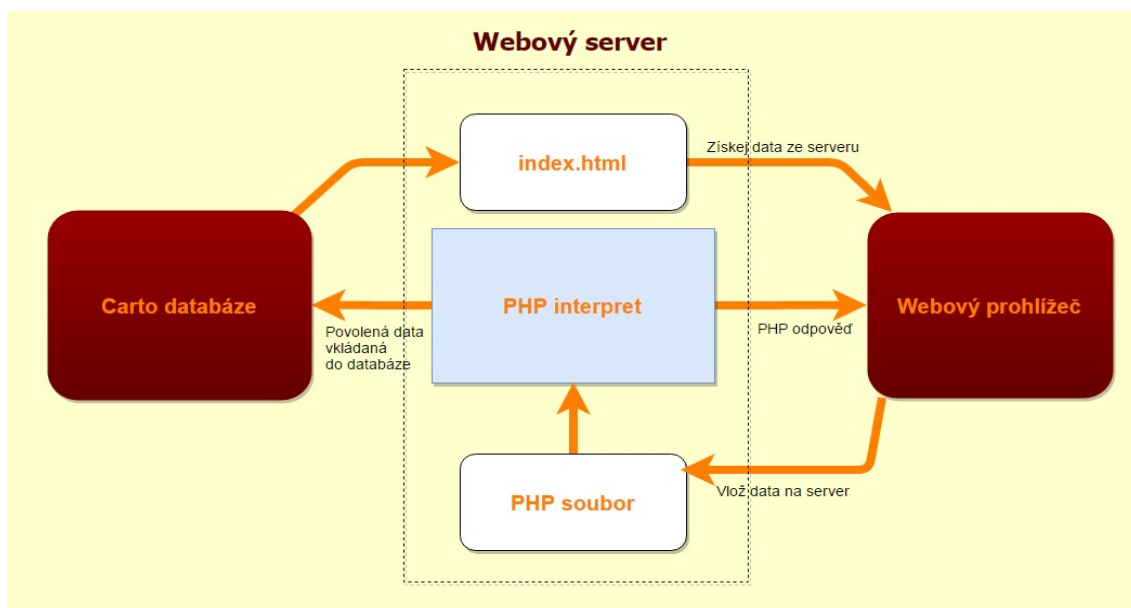
Celé ovládání mapy je prováděno přes JavaScriptové soubory. JS knihovna Leaflet nejdříve načte základní podkladovou mapu ze serveru openstreetmap.org, kde si můžeme zvolit, jaký typ podkladů nám vyhovuje. Ne všechny jsou však zdarma, jak tomu je u OpenStreetMap.

```
1 var basic = L.tileLayer('http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.  
2   png', {  
3     id: 'MapID',  
4     attribution: '© OpenStreetMap'  
5   });
```

Podobným způsobem jsou pak načteny i satelitní(server ArcGisOnline) a bezpopis-
kové(OpenStreetMap) podklady, mezi kterými se lze přepínat. Mapa je následně
vycentrována a přiblížena na defaultní pozici.

Připojení k databázi

Pro připojení k databázi jsou vytvořeny dva PHP soubory. Tyto soubory představují
zprostředkovatele komunikace mezi databází a aplikací. PHP soubory nejsou posí-
lány nikdy klientovi, to znamená, že aplikační klíč pro přístup do databáze bude
vložen v tomto souboru kvůli bezpečnosti.



Obrázek 11: Diagram přenosu dat mezi klientem a databází prostřednictvím PHP souboru.

Vykreslení záznamů do mapy

Do prázdné podkladové mapy se následně musí vykreslit všechny uložené záznamy z databáze. Bod se vykreslí do tzv. popup, což je rozkliknutelná značka na mapě s vloženými informacemi. Nejdříve vytvoření třech proměnných:

```

1 var playgroundLocations=null;
2
3 var cartoDBUserName = "xslegl";
4
5 var query="SELECT * FROM playgrounds";

```

Vygenerování URL dotazu, který se připojí na účet xslegl a vrátí všechny údaje z databáze playgrounds ve formátu GeoJSON. Data jsou pak zformátována k zobrazení a nastavena, jak se budou vypisovat, pokud je sloupec nevyplněn.

Následně pomocí `layer.bindPopup` jsou údaje vykresleny v okně nad značkou hřiště. Všechny značky pak funkce `.addTo(map)` vloží do mapy.

```

1 $.getJSON("https://" + cartoDBUserName + ".carto.com/api/v2/sql?format=
  GeoJSON&q="+query, function(data) {
2   playgroundLocations = L.geoJson(data, {
3     onEachFeature: function (feature, layer) {
4     if (feature.properties.confirmed===false){
5       feature.properties.confirmed='ne';
6     } else {
7       feature.properties.confirmed='ano'
8     }
9     if (feature.properties.image==''){

```

```

10         feature.properties.image = '../map/images/nophoto.jpg';
11     }
12     layer.bindPopup( ""+feature.properties.address+"" + ""+
13         feature.properties.description+
14         "" + "" +
15         " Potvrzeno: "+feature.properties.confirmed + "" );
16     layer.cartodb_id=feature.properties.cartodb_id;
17     }
18     }).addTo(map);
19 });

```

Kontrola číselného kódu ve formuláři

Data jsou z formuláře, po stisknutí tlačítka *přidej hřiště*, předávána metodou **POST**. Nejdříve se vygeneruje pět náhodných čísel od 0 po 9, která jsou zaokroulena na celá. Všechny číslice jsou pak vloženy do proměnné `code` a ta je vložena do html formuláře. Automaticky je tedy vždy vygenerován nový kód, pokud obnovíme stránku.

```

1 var prvni = Math.ceil(Math.random() * 9)+ '';
2 var druhe = -//-;
3 ...
4 var code = prvni + druhe + treti + ctvrte + pate;
5 document.getElementById("captcha").value = code;

```

Funkce pak porovnají, jestli vložený kód od uživatele je stejný jako proměnná `code`. Pokud je prázdný, nebo nesprávný vypíše se hlášení. Pokud se oba shodují, zavolá se funkce obsluhující přidání hřiště do databáze `setData()`.

Vložení nového záznamu přes formulář do databáze

Jakmile jsou všechny údaje vložené uživatelem korektní, zavolá se funkce `setData()`. Ta si uloží hodnoty formuláře do svých proměnných a vygeneruje SQL příkaz `INSERT INTO` pro vložení záznamu do databáze. Například pro souřadnice:

```

1 var enteredLongitude = longitude.value;
2 var enteredLatitude = latitude.value;

```

```

1 "INSERT INTO playgrounds(the_geom)
2 VALUES(ST_SetSRID(ST_Point('" + enteredLongitude + "','" +
    enteredLatitude + '')));

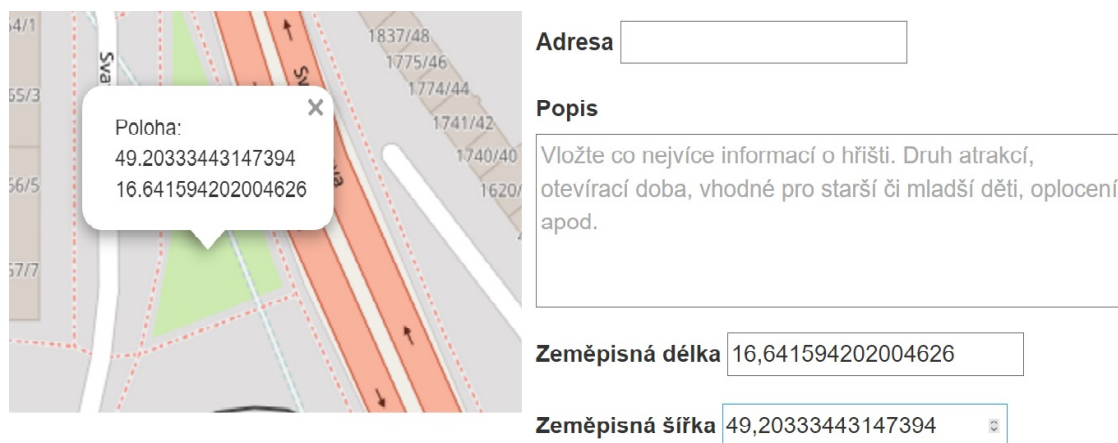
```

Automatické vyplnění souřadnicových polí

Pro jednodušší a rychlejší přidání hřiště je uživateli umožněno, aby pole zeměpisná šířka a délka byly vyplněny hodnotou podle místa kliknutí myši. Jakmile je označeno místo na mapě, vytvoří se okno se souřadnicemi.

Obsah tohoto okna je následně přenesen do formuláře. Vytvoření okna a přenesení hodnot do formuláře, zajišťuje funkce `onMapClick()`.

```
1 var popup = L.popup();
2 function onMapClick(e) {
3     popup
4         .setLatLng(e.latlng)
5         .setContent("Poloha: " + ' ' + popup.getLatLng().lat.toString() +
6             ' ' + popup.getLatLng().lng.toString())
7         .openOn(map);
8     document.getElementById('latitude').value = popup.getLatLng().lat;
9     document.getElementById('longitude').value = popup.getLatLng().lng;
10 }
map.on('click', onMapClick);
```



Obrázek 12: Okno se souřadnicemi vyplňuje formulář.

6 Diskuze

6.1 Možná rozšíření

Do aplikace je možné přidat velké množství nové funkcionality. Hřiště jsou například velmi často rekonstruována. Pokud je hřiště v určité době z tohoto důvodu uzavřeno, uživatel by mohl přidat upozornění a časový údaj, do kdy se provádí příslušné práce. U značení hřiště by pak byl například červený vykřičník.

Jak bude množství hřišť narůstat, je vhodné vytvořit vyhledávání podle určitých kritérií, které zobrazí na mapě jen hřiště, které uživatel vyžaduje. Může to být podle městské části, typů atrakcí, povrchu apod.

6.2 Nevýhody aplikace a problémy při vývoji

Při vytváření aplikace se často naráželo na problém **uživatelské přívětivosti** na úkor **bezpečnosti**. V aplikaci je snaha o co nejmenší požadavky na uživatele při využívání funkcí.

Zavedení autentizace a autorizace uživatelů a určení práv, kdo může přidávat nová hřiště, by zvedlo bezpečnost. Mohlo by to však odradit spousty nových návštěvníků od jejich úmyslu přidat nové záznamy.

Stejný problém byl u anti-spam systému. Čím hůř rozluštitelný je kód, který vkládá uživatel, tím menší šance, že samotný robot ho bude schopen opsat a zvyšuje se bezpečnost. Kód může být například několik špatně čitelných slov nebo kombinace slov a písmen. Často například nemusí být jasné, zda je rozlišováno mezi malými a velkým písmeny. Uživatel pak u opisu kódu stráví dlouhou dobu a pokud udělá někde chybu, musí opisovat úplně nový.

Nevýhodou je také neúplnost aktuálních dat. Ve fázi **sběru dat**, bohužel, ne všechny městské části komunikovaly. Některé neodpověděly, další reagovaly až po dlouhé době. To má za následek, že se více spoléhá na uživatele, kteří chybějící dětská hřiště v průběhu času doplní.

7 Závěr

Cílem práce bylo analyzovat a srovnat stávající přístupy aplikací zpracovávající data ve veřejné zprávě. Následně pak navrhnout a implemetovat funkční aplikaci.

Během přípravy byla nastudována problematika uchovávání dat v relačních a prostorových databázích. Byly stanoveny rozdíly, a poznatků o prostorových databázích, pak využito při výběru databáze vlastní. Dále proběhlo srovnání a charakteristika stávajících aplikací a přístupů, zabývajících se podobnou tematikou.

Vývoj pak probíhal ve třech fázích - sběr dat, výběr databáze a zavedení dat a samotná implemetace webové aplikace. Tyto fáze byly podrobně popsány v kapitole **metodika**.

Aplikace a její funkčnost byla následně popsána v kapitole **vlastní práce** a její zhodnocení proběhlo v sekci **diskuze**.

Podařilo se vytvořit aplikaci pro dětská hřiště v Brně, která nabízí běžným uživatelům základní informace o těchto lokalitách a umožňuje jim také přidávat záznamy nové. Aplikace je dostupná na adrese <https://akela.mendelu.cz/~xslegl/hriste/>.

8 Reference

- AGAFONKIN VLADIMIR. *Leaflet - example*. leafletjs.com. [online]. 2015b [cit. 2016-12-23]. Dostupné z: <http://leafletjs.com/examples/quick-start/example.html>.
- AGAFONKIN VLADIMIR. *Leaflet - Overview*. leafletjs.com. [online]. 2015a [cit. 2016-12-21]. Dostupné z: <http://leafletjs.com>.
- ARCTUR, DAVID A MICHAEL ZEILER. *Designing geodatabases : case studies in GIS data modeling*. 1. New York Street, USA: Redlands, Calif. : ESRI Press, 2004, s. 393. ISBN 1-58948-021-X..
- BOOTSTRAP. *Bootstrap example*. UMKC - School of Computing and Engineering. [online]. 2016a [cit. 2016-12-19]. Dostupné z: <http://sce2.umkc.edu/BIT/burrise/it222/notes/bootstrap.html>.
- BOOTSTRAP. *Bootstrap example*. UMKC - School of Computing and Engineering. [online]. 2016b [cit. 2016-12-21]. Dostupné z: <http://sce2.umkc.edu/BIT/burrise/it222/notes/bootgrid.png>.
- BOUNDLESS. *Introduction to PostGIS - What is a Spatial Database?*. Boundlessgeo. [online]. 2015a [cit. 2015-12-20]. Dostupné z: <http://workshops.boundlessgeo.com/postgis-intro/introduction.html>.
- BOUNDLESS. *Introduction to PostGIS - Spatial Data Types*. Boundlessgeo. [online]. 2015b [cit. 2015-12-20]. Dostupné z: <http://workshops.boundlessgeo.com/postgis-intro/introduction.html>.
- BOUNDLESS. *Introduction to PostGIS - Spatial Functions*. Boundlessgeo. [online]. 2015c [cit. 2015-12-20]. Dostupné z: <http://workshops.boundlessgeo.com/postgis-intro/introduction.html>.
- CARTO. *Carto documentation: Data types*. Carto. [online]. 2012a [cit. 2016-12-25]. Dostupné z: <https://carto.com/docs/tips-and-tricks/data-types>.
- CARTO. *Terms of service*. Carto. [online]. 2012b [cit. 2016-12-14]. Dostupné z: <https://carto.com/terms/#limitations-and-restrictions>.
- COMPUTERHOPE. *Computer Dictionary - HTML*. ComputerHope. [online]. 2016 [cit. 2016-12-19]. Dostupné z: <http://www.computerhope.com/jargon/h/html.htm>.
- GARTNER. *Software as a Service (SaaS)*. Gartner. [online]. 2016 [cit. 2017-01-01]. Dostupné z: <http://www.gartner.com/it-glossary/software-as-a-service-saas/>.
- GIS. *What is cartoDB*. jesidea.com. [online]. 2016 [cit. 2017-01-01]. Dostupné z: <http://jesidea.com/pakdb/gis-what-is-cartodb/>.

- HARRINGTON L. JAN. *Relational Database Design and Implementation: Clearly Explained*. [online]. 3. USA: Morgan Kaufmann, 2009, s. 5 [cit. 2016-12-10]. ISBN 0-0808-8501-2. Dostupné z: https://books.google.cz/books?id=O3Qdc2yb_JoC.
- GÜTING HARTMUT, RALF. *An introduction to spatial database systems*. 3. New York, USA: Springer-Verlag, 1994, s. 399. ISBN 1066-8888.
- JAKNAMIMINKA. *Dětská hřiště*. JakNaMiminka. [online]. 2012 [cit. 2016-12-05]. Dostupné z: <http://www.jaknamiminka.cz/detska-hriste/>.
- MATĚJKA JIŘÍ. *Dětská hřiště - Česko*. VenKazdyDen. [online]. 2016b [cit. 2016-12-01]. Dostupné z: <https://www.venkazdyden.cz>.
- MATĚJKA JIŘÍ. *Mobilní aplikace*. VenKazdyDen. [online]. 2016a [cit. 2016-12-02]. Dostupné z: <https://www.venkazdyden.cz/home/mobilni-aplikace>.
- MICROSOFT. *Spatial Data Types Overview*. Msdn.microsoft. [online]. 2015 [cit. 2015-12-20]. Dostupné z: <https://msdn.microsoft.com/en-us/library/bb964711.aspx>.
- MIKOLUK KASIA. *XAMPP Tutorial*. blog.udemy.com. [online]. 2013 [cit. 2016-12-23]. Dostupné z: <http://www.gartner.com/it-glossary/software-as-a-service-saas/>.
- PATONNIER JÉRÉMIE. *Introduction - JavaScript*. Mozilla Developer Network. [online]. 2016 [cit. 2016-12-21]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Introduction>.
- POSTGIS. *PostGIS 2.2.1dev Manual*. PostGIS. [online]. 10.7. 2015 [cit. 2015-12-20]. Dostupné z: <http://postgis.net/docs/>.
- RIGAUX PHILIPPE, MICHEL SCHOLL A AGNES VOISARD. *Spatial Databases with application to GIS*. 1. San Francisco, USA: The Morgan Kaufmann, 2001, s. 410. ISBN 1-55860-588-6..
- RITCHIE, COLIN. *Relational Database Principles*. [online]. 2nd edition. London: Cengage Learning EMEA, 2002a, s. 8 [cit. 2016-12-10]. ISBN 0-8264-5713-4. Dostupné z: <https://books.google.cz/books?id=vYBzeAJog7IC>.
- RITCHIE, COLIN. *Relational Database Principles*. [online]. 2nd edition. London: Cengage Learning EMEA, 2002b, s. 10 [cit. 2016-12-10]. ISBN 0-8264-5713-4. Dostupné z: <https://books.google.cz/books?id=vYBzeAJog7IC>.
- RITCHIE, COLIN. *Relational Database Principles*. [online]. 2nd edition. London: Cengage Learning EMEA, 2002c, s. 8-9 [cit. 2016-12-10]. ISBN 0-8264-5713-4. Dostupné z: <https://books.google.cz/books?id=vYBzeAJog7IC>.
- SHEKHAR, SHASHI A SANJAY CHAWLA. *Spatial Databases: A Tour*. 1. University of Minnesota: Prentice Hall, 2003, s. 262. ISBN 978-0130174802..

- TUTORIALSPPOINT. *What is CSS*. Tutorialspoint. [online]. 2016 [cit. 2016-12-19]. Dostupné z: https://www.tutorialspoint.com/css/what_is_css.htm.
- V AŠÁK MARTIN, KUTHAN JAKUB, RAJCHLOVÁ ALŽBĚTA, PEKÁREK VÁCLAV, SHTIBEN OXANA. *Dětská hřiště Praha - O projektu*. Hřiště Praha. [online]. 2014a [cit. 2016-12-05]. Dostupné z: <http://www.hristepraha.cz/o-projektu>.
- V AŠÁK MARTIN, KUTHAN JAKUB, RAJCHLOVÁ ALŽBĚTA, PEKÁREK VÁCLAV, SHTIBEN OXANA. *Mapa hřišť v Praze*. Hřiště Praha. [online]. 2014b [cit. 2016-12-06]. Dostupné z: <http://www.hristepraha.cz/hriste/mapa/>.
- WINSCP. *Introducing WinSCP*. winscp.net. [online]. 2014 [cit. 2016-12-25]. Dostupné z: <https://winscp.net/eng/docs/introduction>.
- WINSCP. *WinSCP: Supported File Transfer Protocols*. winscp.net. [online]. 2016 [cit. 2016-12-23]. Dostupné z: <https://winscp.net/eng/docs/protocols>.
- ZÁKON Č. 89 ze dne 1. ledna 2014 Podoba a soukromí. In: Nový občanský zákoník. [online]. 2012 [cit. 2016-12-16]. Dostupné z: <http://zakony.kurzy.cz/89-2012-obcansky-zakonik/paragraf-84/>.