



Pedagogická  
fakulta  
Faculty  
of Education

Jihočeská univerzita  
v Českých Budějovicích  
University of South Bohemia  
in České Budějovice

Jihočeská univerzita v Českých Budějovicích  
Pedagogická fakulta  
Katedra informatiky

Bakalářská práce

# Geolokace ve webovém prohlížeči pomocí HTML5 Geolocation API

Vypracoval: Martin Laudát  
Vedoucí práce: PaedDr. Petr Pexa, Ph.D.

České Budějovice 2013

JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH

Fakulta pedagogická

Akademický rok: 2011/2012

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Martin LAUDÁT**  
Osobní číslo: **P10348**  
Studijní program: **B7507 Specializace v pedagogice**  
Studijní obor: **Informační technologie ve vzdělávání**  
Název tématu: **Geolokace ve webovém prohlížeči s využitím HTML5 Geolocation API**  
Zadávací katedra: **Katedra informatiky**

### Z á s a d y   p r o   v y p r a c o v á n í :

Cílem bakalářské práce bude rozbor aktuálního stavu vývoje nové a perspektivní technologie geolokace ve webovém prohlížeči, která je součástí HTML5. Tato technologie umožňuje určit pozici dle posláni požadavku přímo z webového prohlížeče (Geolocation API). Práce se bude zabývat výhodami této technologie především ve využití pro podnikání na internetu a také nevýhodami, které souvisí s možným zneužitím osobních údajů se zaměřením na aktuální geografickou polohu. Bude provedeno i testování podpory této nové technologie v dostupných verzích prohlížečů a především vytvořena sada praktických příkladů pro využití této technologie v reálném životě.

Rozsah grafických prací: **CD ROM**

Rozsah pracovní zprávy: **60**

Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

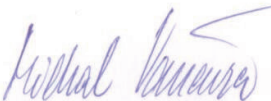
1. **HTML5: up and running**. 1st ed. Sebastopol, CA: O'Reilly, 2010. ISBN 05-968-0602-7.
2. **HTML5.cz - vše co potřebujete vědět o HTML5** [online]. Dostupné z: <http://www.html5.cz/>
3. **GOLDSTEIN, Alexis, Louis LAZARIS a Estelle WEYL. HTML5 a CSS3 pro webové designéry**. Vyd. 1. Brno: Zoner Press, 2011, 286 s. Encyklopedie webdesignera. ISBN 978-80-7413-166-0.
4. **Geolocation API Specification** [online]. 2011. Dostupné z: <http://dev.w3.org/geo/api/spec-source.html>
5. **Geolokace v prohlížeči**. MALÝ, Martin. Zdrojak.cz [online]. 2010. Dostupné z: <http://zdrojak.root.cz/clanky/geolokace-v-prohlizeci/>
6. **LUBBERS, Peter, Brian ALBERS a Frank SALIM. HTML5: programujeme moderní webové aplikace**. Vyd. 1. Brno: Computer Press, 2011, 304 s. ISBN 978-80-251-3539-6.
7. **HOLDENER, Anthony T. HTML5 Geolocation**. Sebastopol, CA: O'Reilly, 2011, 95 s. ISBN 14-493-0472-9.

Vedoucí bakalářské práce: **PaedDr. Petr Pexa, Ph.D.**

Katedra informatiky


Datum zadání bakalářské práce: **12. dubna 2012**

Termín odevzdání bakalářské práce: **26. dubna 2013**



Mgr. Michal Vančura, Ph.D.

děkan



doc. PaedDr. Jiří Vaníček, Ph.D.

vedoucí katedry

V Českých Budějovicích dne 12. dubna 2012

# Prohlášení

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

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

V Českých Budějovicích dne 23.6. 2013

Martin Laudát

## **Anotace**

Cílem bakalářské práce je rozbor aktuálního stavu vývoje nové a perspektivní technologie geolokace ve webovém prohlížeči, která je součástí HTML5. Tato technologie umožňuje určit pozici dle poslání požadavku přímo z webového prohlížeče (Geolocation API). Práce se zabývá výhodami této technologie především ve využití pro podnikání na internetu a také nevýhodami, které souvisí s možným zneužitím osobních údajů se zaměřením na aktuální geografickou polohu. Testování podpory této nové technologie je prováděno v různých prohlížečích a především je vytvořena sada praktických příkladů pro využití této technologie v reálném životě.

### **Klíčová slova**

Geolokace, HTML5, Geolokační API, Webový prohlížeč

## **Abstract**

The aim of the work is analysis of actual state of development of a new and perspective technology geolocation in the web browser, which is a part of HTML5. This technology enables to define position through sending requirement right from the web browser (geolocation API). The work deals with advantages of this technology mainly in using it for bussiness on the internet and it also deals with disadvantages which are connected with possible abuse of personal datas with the aim on actual geographical position. Testing of support of this new technology is done in different browsers and set of practical examples for using of this technology in real life is created.

### **Keywords**

Geolocation, HTML5, Geolocation API, Web browser

## **Poděkování**

Touto cestou bych rád poděkoval PaedDr. Petru Pexovi, Ph.D. za jeho cenné rady a trpělivost při vedení mé bakalářské práce.

# Obsah

<b>1</b>	<b>Cíle bakalářské práce</b>	<b>6</b>
<b>2</b>	<b>Metody bakalářské práce</b>	<b>8</b>
<b>3</b>	<b>Východiska bakalářské práce</b>	<b>9</b>
<b>4</b>	<b>Úvod</b>	<b>10</b>
<b>5</b>	<b>Novinky v HTML5</b>	<b>11</b>
5.1	Úvod . . . . .	11
5.2	Novinky v HTML5 . . . . .	12
5.2.1	Výběr nových elementů v HTML5 . . . . .	13
5.2.2	Nová API a technologie v HTML5 . . . . .	17
5.3	Podpora HTML5 ve starších prohlížečích . . . . .	19
<b>6</b>	<b>HTML5 Geolocation API</b>	<b>21</b>
<b>7</b>	<b>Funkce HTML5 Geolocation API</b>	<b>24</b>
7.1	Ověření podpory Geolocation API v prohlížeči . . . . .	24
7.2	Získání polohy ve webovém prohlížeči . . . . .	26
7.2.1	Parametr Showlocation . . . . .	27
7.2.2	Parametr ErrorHandler . . . . .	29
7.2.3	Parametr Options . . . . .	30
7.2.4	Ukázkový příklad jednorázového získání polohy . . . . .	31
7.3	Periodické získávání polohy ve webovém prohlížeči . . . . .	36
<b>8</b>	<b>HTML5 Geolocation API s různými mapovými podklady</b>	<b>38</b>
8.1	HTML5 Geolocation API a Google Maps . . . . .	39
8.2	HTML5 Geolocation API a Mapy.cz . . . . .	40
8.3	HTML5 Geolocation API a Bing Maps . . . . .	41
8.4	HTML5 Geolocation API a OpenLayers . . . . .	43
<b>9</b>	<b>Metody lokalizace mobilních zařízení</b>	<b>45</b>
9.1	Lokalizace pomocí GPS Hardware . . . . .	45

9.2	Lokalizace pomocí IP adresy . . . . .	46
9.3	Lokalizace pomocí datových sítí . . . . .	47
<b>10</b>	<b>Podpora HTML5 Geolocation API v prohlížečích</b>	<b>49</b>
<b>11</b>	<b>Využití geolokace na internetu</b>	<b>51</b>
11.1	Dáme jídlo . . . . .	53
11.2	Foursquare . . . . .	53
11.3	TripAdvisor . . . . .	54
11.4	Geocaching . . . . .	54
11.5	Různé sociální sítě . . . . .	55
<b>12</b>	<b>Nebezpečí zveřejňování své polohy na internetu</b>	<b>56</b>
<b>13</b>	<b>Praktické příklady</b>	<b>59</b>
13.1	Praktické příklady s mapovými podklady . . . . .	59
13.2	Hlavní praktický příklad . . . . .	60
<b>14</b>	<b>Závěr</b>	<b>61</b>
<b>15</b>	<b>Literatura a zdroje</b>	<b>62</b>



# 1 Cíle bakalářské práce

Cílem bakalářské práce je přiblížit odborné veřejnosti technologii Geolocation API obsaženou ve standardu HTML5. Tato práce se bude zabývat představením nejzásadnějších novinek HTML5, detailním rozбором možností této technologie v dnešní době, vysvětlením problematiky zjišťování polohy ve webovém prohlížeči a možnými riziky, které mohou hrozit při zveřejňování polohy na internetu, otestováním tohoto API v prohlížečích a vysvětlením, jak řešit nekompatibilitu s určitými prohlížeči. V rámci této práce bude veřejně na internetu dostupná webová stránka, kde se bude nalézat sada praktických příkladů a jeden hlavní praktický příklad. Jedná o sadu příkladů, které demonstrují spolupráci HTML5 Geolocation API s různými mapovými podklady a hlavním praktickým příkladem je vytvoření komplexnější webové aplikace využívající toto API.

Menší praktické příklady se zabývají využitím HTML5 Geolocation API s nejnámějšími mapovými podklady, které jsou na internetu dostupné. Mezi tyto podklady patří Google Maps, Bing Maps, Mapy.cz a OpenLayers. Dále jsou sepsány jednoduché návody, jak s jednotlivými podklady pracovat a samozřejmě vytvoření hotových ukázek.

Hlavní projekt by se měl zabývat využitím HTML5 Geolocation API v komplexnější aplikaci než-li pouze zobrazení naměřených výsledků na určitých mapových podkladech. Tato webová aplikace by měla využívat geolokaci pro zpříjemnění uživatelského zážitku a pro možné generování zisku. Mělo by se tedy jednat o aplikaci, která má reálný smysl a řeší konkrétní problém. Hlavní příklad by měl ukazovat, jak implementovat toto API mezi ostatní moderní webové technologie a vytvořit tak kompletní webovou aplikaci, jejímž jádrem jsou mapové podklady a HTML5 Geolocation API.

Celkově by tato práce měla sloužit jako průvodce HTML5 Geolocation API pro webové vývojáře, kteří toto API chtějí použít a jako zdroj informací běžných uživatelů k pochopení problematiky zjišťování polohy na internetu a k uvědomění rizik, která mohou při sdílení polohy na internetu nastat.

## 2 Metody bakalářské práce

V úvodu se zabývám zásadními novinkami v HTML5, a pak přecházím k technologii Geolocation API, kde detailně rozebírám funkce tohoto API, testuji funkčnost API ve všech moderních prohlížečích a samozřejmě také v mobilních prohlížečích. Popisuji výhody a nevýhody této technologie, bezpečnostní rizika spojená se zveřejňováním své polohy na různých sociálních sítích. Provádím čtenáře tvorbou základní aplikace pro zjišťování a vypisování polohy ve webovém prohlížeči. Poté popisuji mnou zvolený příklad a názorně ukazuji postup tvorby za pomoci HTML5/CSS3/JavaScript, popisuji nasazení této technologie do obchodní sféry a na závěr shrnuji celou práci.

### 3 Východiska bakalářské práce

Mým cílem je rozebrat a popsat novou technologii užitou v HTML5, a to technologii Geolokace ve webovém prohlížeči pomocí Geolocation API. V dnešní době, kdy se webové aplikace dostávají do mobilních zařízení, a to ať do mobilních telefonů či tabletů, se rozšiřují i možnosti využití aplikací na těchto zařízeních. Konkrétně na mobilních zařízeních je potřeba zjišťovat polohu zařízení větší a větší. Tuto potřebu mají hlavně různé sociální sítě (Facebook.com, Foursquare.com, atd.), ale také ostatní aplikace. Jednou z nejznámějších aplikací je GPS aplikace pro geocaching, kde se podle GPS souřadnic hledají schované poklady.

Do příchodu technologie HTML5 nebyla jiná možnost než vytvářet pro zjišťování polohy nativní aplikace pro konkrétní typy telefonu a konkrétní typ operačního systému. Tuto vlastnost technologie Geolocation API v HTML5 zcela ruší, protože se aplikace pro zjišťování polohy přesouvají z nativních mobilních aplikací do mobilních webových prohlížečů.

V dnešní době je tato technologie podporována ve všech moderních prohlížečích na PC a pomalu se začíná dostávat i do nejnovějších mobilních prohlížečů. S příchodem plné podpory této technologie a technologie HTML5 vznikne na webu prostor pro mnoho aplikací různých zaměření.

## 4 Úvod

V dnešní době se potřeby uživatelů na internetu neustále rozšiřují. Zvyšují se nároky na funkce, které mohou webové stránky nabízet. Díky tomuto vývoji vznikají i nové moderní webové technologie. Nejzákladnější metodou tvorby webu je dnes značkovací jazyk HTML, který ve své poslední specifikaci HTML5 přináší velké množství revolučních novinek.

Mezi tyto revoluční novinky patří například sémantičtější zápis, nové elementy a nová API, jako jsou například Fullscreen API, které dovoluje vytvářet aplikace pro celou obrazovku zařízení, ze kterého web prohlížíme. Dalším novým API je Drag and Drop API, které umožňuje intuitivně přesouvat prvky na webových stránkách v reálném čase. Dalším API je Geolocation API, o kterém bude tato práce pojednávat. Geolocation API je velice jednoduché API, které nám umožňuje zjišťovat zeměpisnou polohu zařízení, ze kterého se dotazujeme. A to vše pouze ve webovém prohlížeči, není tedy potřeba vytvářet nativní aplikace pro určitý OS, ale vše lze obsloužit ve webovém prohlížeči. Díky Javascriptu je možné pak dále s těmito údaji pracovat a vytvářet tak moderní webové aplikace.

Dnes se webové stránky a aplikace vytvářejí hlavně pro uživatele, který je bude používat. Existují způsoby, jak navrhnout dobré webové rozhraní, na kterém se uživatel bude chovat intuitivně a nebude bloudit. Takto navržený web dokáže uživatele na webu udržet, a dokonce může uživatele přimět k nějaké interakci, například nákupu. Díky novým technologiím lze takovéto weby vytvářet snadněji než dříve.

## 5 Novinky v HTML5

### 5.1 Úvod

Značkovací jazyk HTML od své první verze ušel již pořádný kus cesty. Stále je nejrozšířenějším jazykem pro popis dat nebo popis obsahu webových stránek. Jeho poslední verze nese název HTML5 a přináší množství nových funkcí, vylepšuje stávající funkce a obsahuje i několik důležitých API postavených na skriptech. Všechny tyto vlastnosti umožňují vytvářet moderní webové stránky s řadou užitečných funkcí a vylepšení. Díky tomu, že HTML5 obsahuje nové značkovací prvky, mohou být stránky přístupnější a srozumitelnější. Dnes nemusíme na hlavní prvky layoutu webových stránek používat pouze značku `div`, ale lze použít místo toho značky jako `header`, `nav`, `section`, `article`, `footer` a další. Použitím těchto značek se sémantika webových stránek značně zpřehlední.

HTML5 není zcela novou verzí předchozího HTML, respektive XHTML, obsahuje prvky, které byly použity v HTML4 a XHTML 1.0. Díky tomu je částečně HTML5 podporováno i ve starších prohlížečích, tedy elementy, které byly zároveň použity v předchozích specifikacích. Pro nové elementy v HTML5 je nutné podporu ve starších prohlížečích zařídit. Cílem bylo vytvořit značkovací jazyk, který je modernější než jeho předchůdci, bude kompatibilní s většinou dnešních platforem, bude umět správně obstarávat chyby a bude lehce udržovatelný.

Samotný termín HTML5 je často používán pro sadu nástrojů, různých API a nových technologií pro tvorbu moderních webů. Může to být Geolocation API, které slouží pro vyhledání polohy zařízení přímo ve webovém prohlížeči nebo to může být plátno `canvas`, které umožňuje ve webovém prohlížeči kreslit různé grafické prvky. Může se také jednat o novou podporu audia a videa, vytváření offline aplikací či Drag-and-drop API a další. Je třeba zmínit, že se pod názvem HTML5 ukrývají i technologie, které tam

patřily, ale časem se oddělily do vlastních vývojových větví nebo technologie, které do HTML5 nikdy nepatřily.

Technologie HTML5 původně vznikala jako dvě různé specifikace, které se nazývaly Web Forms 2.0 a Web Apps 2.0. Tyto specifikace byly reakcí na změny webových stránek a aplikací a staly se řešením pro moderní, svižné a efektivní webové stránky a aplikace. Je tedy logické, že se stejnými směry vydala i specifikace HTML5. Současně s vývojem HTML5 se vyvíjelo i XHTML 2.0, ale vývoj XHTML byl později ukončen. Je známým faktem, že za vývojem HTML5 stojí dvě organizace. W3C (World Wide Web Consortium), které se stará o původní specifikace HTML a CSS specifikace dalších technologií. A druhou organizací je WHATWG (Web Hypertext Application Technology Working Group), která se skládá ze zaměstnanců společností Apple, Mozilla a Opera. Organizace WHATWG vznikla proto, že se lidem, kteří tuto organizaci založili, nelíbilo, že organizace W3C ignoruje potřeby a požadavky společností, které vyvíjejí webové prohlížeče. Takže organizace WHATWG začala tvořit specifikace Web Forms 2.0 a Web Apps 2.0, které byly zmiňovány výše. Ovšem po určité době došlo ke sloučení těchto dvou specifikací, a to dalo za vznik nové specifikaci HTML5, na kterou později přešlo i W3C. Specifikace od WHATWG (<http://www.whatwg.com/html/>) obsahuje více nových a experimentálních funkcí než specifikace od W3C (<http://dev.w3.org/html5/spec/>). Obě zmíněné specifikace jsou si velice podobné. Obecně se dá říci, že se HTML5 skládá ze specifikace W3C a specifikace WHATWG. Závěrem je třeba říci, že HTML5 ještě není zcela dokončené a je třeba si uvědomit, že jeho použití je stále na experimentální úrovni.

## 5.2 Novinky v HTML5

Jak již bylo zmíněno, HTML5 přichází s několika novinkami a vylepšeními, které dovolují vytvářet modernější, více sémantický a celkově zajímavější web, než tomu bylo doposud. V následujícím textu uvádím ty nejzajímavější novinky trochu blíže.

## 5.2.1 Výběr nových elementů v HTML5

### Doctype

Definice dokumentu se s příchodem HTML5 razantně změnila. Při definování HTML5 dokumentu stačí prohlížeči říci, že se jedná o HTML dokument a víc nic už není nutné psát. Zápis definice dokumentu je tím pádem mnohem jednodušší.

Příklad zápisu definice dokumentu HTML 4.01 Transitional:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

Příklad zápisu definice dokumentu HTML5:

```
<!DOCTYPE html>
```

### Article

Element article reprezentuje část obsahu stránky, který tvoří samostatnou a nezávislou část, například článek nebo komentář.<sup>1</sup> Element může obsahovat i další elementy a je párový.

Ukázka zápisu:

```
<article>
  <!-- obsah např. článek pro blog -->
</article>
```

### Aside

Element aside slouží pro popis obsahového prvku webové stránky. Jedná se spíše o sekundární obsah. Například u nějakého článku může aside plnit

---

<sup>1</sup>HTML5 - nové vlastnosti: Article. In: ŠTĀSTNÝ, Jiří. *Programujte.com* [online]. 2011 [cit. 2013-02-21]. Dostupné z: <http://programujte.com/clanek/2010082200-html5-nove-vlastnosti/>



funkci doplňkových informací viz ukázka zápisu. Nejčastější použití bude pro deklaraci nějakého postranního panelu. Jedná se o párový tag.

Ukázka zápisu:

```
<article>
<!-- hlavní obsah nějakého článku -->
  <aside>
    <!-- sekundární obsah, doplňkové informace -->
  </aside>
</article>
```

## Canvas

Element canvas nemá sám o sobě žádnou funkci, tento element pouze obaluje grafiku. Jak název sám napovídá, jedná se o plátno (canvas = angl.plátno). Na toto pomyslné plátno můžeme kreslit za pomoci scriptů. Tento element je párový.

Ukázka zápisu:

```
<canvas></canvas>
```

## Figcaption a figure

Element figcaption obaluje nějaký prvek (například obrázek) a element figure přidává k tomuto obrázku popisek. Element figure je tedy potomkem elementu figcaption. Oba elementy jsou párové.

Ukázka zápisu:

```
<figure>
  
  <figcaption>Popisek obrázku</figcaption>
</figure>
```

## Header

Element header slouží pro popis hlavičky dokumentu nebo jen hlavičky úseku dokumentu - článku. Při vytváření layoutu webu již není nutné vytvářet `<div id="header">`, ale je možné použít element header. Jedná se o párový element.

Ukázka zápisu:

```
<header>  
  <!-- obsah hlavičky -->  
</header>
```

## Footer

Podobně jako header, element footer popisuje patičku dokumentu, nebo jen úseku dokumentu. Opět není nutné používat nesémantický zápis `<div id="footer">`, ale je možné použít element footer. Jedná se o párový element.

Ukázka zápisu:

```
<footer>  
  <!-- obsah patičky -->  
</footer>
```

## Hgroup

Element hgroup se používá pro seskupování více nadpisů (h1...h6). Tento element je párový a smí obsahovat pouze nadpisy.

Ukázka zápisu:

```
<hgroup>  
  <h1>Nadpis H1</h1>
```

```
    <h2>Podnadpis H2</h2>
      <h3>Podnadpis H3</h3>
</hgroup>
```

## Nav

Jak samotný název elementu napovídá, jedná se o element popisující navigaci. Tento element se používá pro popis hlavní navigace webových stránek. Jedná se o párový element.

Ukázka zápisu:

```
<nav>
  <ul>
    <li>Prvek navigace 1</li>
    <li>Prvek navigace 2</li>
    <li>Prvek navigace 3</li>
  </ul>
</nav>
```

## Section

Element section slouží pro popis části dokumentu, která je samostatná a její vyjmutí nenaruší logiku celého webu. Tedy, když obsah elementu section odstraní, logika stránky zůstane stejná a nic se tím nenaruší. Může se jednat například o článek na webu. Element je párový.

Ukázka zápisu:

```
<section>
  <h2>Nadpis</h2>
  <p>Obsah</p>
</section>
```

## Time

Time element slouží pro zápis časových údajů na webu. Umožňuje zvolit si vlastní formát času. Jedná se o párový tag.

Ukázka zápisu:

```
<time datetime="2013-02-25"> 25.listopadu 2013</time>
```

### 5.2.2 Nová API a technologie v HTML5

#### Geolocation API

Geolocation API umožňuje zjistit polohu zařízení, ze kterého se dotazujeme přímo ve webovém prohlížeči. Nemusí se tedy používat žádné nativní aplikace, ale stačí k tomu pouze webový prohlížeč s podporou HTML5 a javascriptu.

#### Drag and Drop API

Toto API umožňuje na úrovni webového prohlížeče přetahovat a přemisťovat objekty na webové stránce. Je možné si to představit tak, že na webové stránce jsou dva boxy - jeden modrý a druhý žlutý. V modrém boxu je obrázek, který díky tomuto API lze přetáhnout do žlutého boxu, kam se obrázek umístí a zůstane tam.

#### Full-screen API

Full-screen API umožňuje spouštět aplikace v plném zobrazení, tedy přes celou plochu. V minulých letech se toto řešilo pomocí Flash aplikací, ale příchodem HTML5 je už tento problém řešit přímo v prohlížeči a není potřeba žádné aplikace navíc.

## **File API**

V moderních webových aplikacích je potřeba spravovat uživatelské vstupy, které často obsahují i vkládání souborů. HTML5 File API umožňuje zpracovávat data, které uživatel vkládá přímo na straně uživatele. Lze si tedy tato data na straně uživatele připravit a až poté odeslat na server, kde s nimi pomocí nějakého serverového jazyka, jako je např. PHP, můžeme pracovat dál.

## **Offline web applications**

Díky této technologii, lze vytvářet webové aplikace, které mohou částečně fungovat i bez trvalého připojení k internetu. Takto lze získat webovou aplikaci, která při ztrátě připojení k internetu bude částečně fungovat dál, respektive lze zvolit data, která se budou ukládat na uživatelský počítač a budou dostupná i bez připojení k internetu – serveru.

## **Audio a video**

HTML5 audio a video je nová technologie, která dokáže přímo v prohlížeči přehrávat audio a video soubory bez aplikací třetích stran, jako je Adobe Flash, a to jen díky elementům audio a video. Takže pro přehrávání těchto souborů stačí pouze prohlížeč, který podporuje HTML5 audio a video technologii.

## **CSS3**

Společně s novou specifikací HTML5 přišla i specifikace CSS3, která přináší velké množství nových funkcí a vylepšení. Díky CSS3 mohou webové stránky dostat úplně jiný rozměr. Mezi hlavní novinky patří možnost vytváření animací (a to i 3D), nové efekty jako zaoblení rohů, vržený stín nebo barevný přechod, které lze použít na objektech. Další novinky jsou tzv. Media queries. Tato funkce umožňuje přizpůsobovat webovou stránku zařízením, se

kterými webové stránky prohlížíme. Jedná se například o chytré mobilní telefony nebo tablety. Je třeba ovšem říci, že specifikace CSS3 není ještě úplně hotová, a proto je CSS3 stále v experimentální fázi. S každou novou verzí prohlížečů přichází i podpora pro nové funkce CSS3. Ve starších prohlížečích můžeme CSS3 simulovat pomocí javascriptových knihoven.

### 5.3 Podpora HTML5 ve starších prohlížečích

Nové elementy v HTML5 jsou podporovány ve všech „moderních“ prohlížečích – v aktuálních verzích prohlížečů. Samozřejmě je podíl starších prohlížečů stále značně vysoký, proto je při vývoji moderních aplikací v HTML5 myslet i na uživatele, kteří používají starší neaktualizované prohlížeče.

Pro použití HTML5 a jeho nových elementů je nutné tedy nějak zařídit podporu těchto elementů i ve starších prohlížečích. Tento problém pomůže vyřešit javascriptová knihovna HTML5shiv<sup>2</sup>. Díky této knihovně lze používat HTML5 elementy v prohlížečích jako Internet Explorer 6-9, Safari 4.x, iPhone 3.x a Firefox 3.x. Instalace je velice jednoduchá. Uživatel stáhne tuto knihovnu, uloží ji do složky, kde má vytvořenou HTML5 webovou stránku nebo aplikaci a pak pouze v hlavičce webové stránky odkáže na tuto knihovnu. Nic víc není potřeba. A aplikací tohoto postupu lze vytvářet HTML5 stránky nebo aplikace, které budou fungovat i ve starších prohlížečích.

Jestliže ovšem chce uživatel přidat podporu nějakého HTML5 API, musí použít složitější postup. Základním prvkem je Modernizr<sup>3</sup> – knihovna, která zjišťuje podporu nových elementů a technologií v prohlížeči. Tedy, jestliže Modernizr zjistí, že prohlížeč nové technologie podporuje, tak nic neudělá a uživatel může pracovat dál. Jestliže ovšem Modernizr zjistí, že prohlížeč tyto technologie nepodporuje, vyvolá událost, která informuje uživatele, že jeho prohlížeč není kompatibilní s těmito technologiemi a funkcemi. Tohoto lze

---

<sup>2</sup>Více na: <https://github.com/aFarkas/html5shiv>

<sup>3</sup>Více na: <http://www.modernizr.com>

využít a při zjištění, že prohlížeč danou technologii nepodporuje, je možné podstrčit alternativní řešení – tzv. polyfills<sup>4</sup>. Jedná se o většinou javascriptové knihovny, které simulují moderní technologie, jako je například podpora formátu \*.SVG ve starších prohlížečích. Lze tak docílit toho, že v moderních prohlížečích budou použity nativní technologie obsažené v HTML5 a ve starších prohlížečích budou tyto technologie simulovány pomocí těchto polyfillů.

---

<sup>4</sup>Seznam polyfillů: <https://github.com/Modernizr/Modernizr/wiki/HTML5-Cross-Browser-Polyfills>

## 6 HTML5 Geolocation API

Hlavním cílem této práce je představit a popsat novou technologii obsaženou v HTML5, a to geolokační rozhraní – Geolocation API. Účelem HTML5 Geolocation API je zjišťovat polohu zařízení, ze kterého se uživatel dotazuje přímo ve webovém prohlížeči a s těmito informacemi dále pracuje. Je možné zjistit aktuální polohu zařízení (v souřadnicovém systému), aktuální nadmořskou výšku, přesnost získaných dat, natočení zařízení, ze kterého se dotazujeme, rychlost pohybu zařízení, ale je možné zjišťovat polohu i průběžně a vytvářet tak záznam o pohybu. Dále tato technologie umožňuje jednoduše obsluhovat chyby pro příjemný uživatelský zážitek. K těmto informacím lze přistupovat pomocí javascriptu a v javascriptu s nimi i dále pracovat. Toto API není nijak složité, jeho hlavní metodou je metoda `getCurrentPosition`, která obsahuje tři parametry.

Těmito parametry jsou:

- Parametr `showLocation` – určuje callback metodu, kterou prohlížeč zavolá po získání informace o poloze<sup>5</sup>.
- Další parametr je `ErrorHandler` – určuje callback metodu, která se zavolá při vzniku nějakého problému.
- Posledním parametrem je `Options` – tento parametr umožňuje nastavit další možnosti.

Zápis metody `getCurrentPosition` bude vypadat tedy nějak takto:

```
getCurrentPosition (showLocation, ErrorHandler, options);
```

Geolocation API je revoluční technologií, protože již není nutné být vázán na nativní aplikace, ale lze pracovat s polohou zařízení přímo ve webovém

---

<sup>5</sup>HTML5 – geolokační rozhraní. *Programujte.com - web o programování, webdesignu, počítačové grafice, databázích, elektrotechnice a designu* [online]. 2012 [cit. 2013-03-14]. Dostupné z: <http://programujte.com/clanek/2011052400-html5-geolokacni-rozhrani/>



prohlížeči. Před uvedením HTML5 a tohoto API bylo nutné vytvářet nativní aplikace pro každou platformu zvlášť, což bylo velice komplikované a složité. Dnes stačí jedna webová aplikace, která běží v prohlížeči pro všechny dnešní běžné platformy, jako jsou Windows, Android a iOS. Geolocation API je celkem připravené na nasazení do ostrého provozu, ovšem je třeba si uvědomit, že je to technologie na experimentální úrovni, a i když je podpora v moderních prohlížečích velice slušná, tak ve starších prohlížečích tato technologie nemusí fungovat vůbec. Proto při vývoji aplikace, která bude Geolocation API používat, musíme na tuto skutečnost myslet a nechat si otevřená zadní vrátka, do kterých bude možné podstrčit nějakou simulaci Geolocation API v případě, že na zařízení, ze kterého se budou uživatelé dotazovat, nebude Geolocation API plně funkční.

HTML5 Geolocation API se ovšem nestará o proces zjištění aktuální polohy. O tento proces se stará samo zařízení, ze kterého se dotazujeme. Geolocation API dá pouze podnět k získání polohy ze zařízení a o samotné zjištění se pak postará zařízení, ze kterého se dotazujeme. Z toho vyplývá, že technologii pro určení aktuální polohy si volí samo zařízení z jeho dostupných zdrojů. Takže způsob, jakým je zjištěna aktuální poloha, závisí pouze na zařízení. Způsobů zjišťování polohy je několik, mezi hlavní tři způsoby patří: zjištění polohy skrze IP adresu, zjištění polohy přes GPS modul a posledním způsobem je zjištění polohy pomocí dat z přístupových bodů, jako jsou wifi sítě, mobilní sítě atd. Po vytvoření požadavku na získání aktuální polohy si zařízení samo vybere, jakou metodu zvolí, respektive má-li k dispozici GPS modul, zjistí polohu pomocí tohoto modulu, pokud zařízení GPS modul nemá, zjistí polohu pomocí dalších způsobů – IP adresy nebo datové sítě. Samozřejmě nejpřesnější metodou je zjišťování polohy pomocí GPS modulu. Zbylé dva způsoby již tak přesné nejsou. Zjišťování polohy přes IP adresu zařízení se provádí tak, že se zjistí fyzická adresa providera internetového připojení, tudíž jestliže má provider sídlo na druhé straně republiky, získaná poloha bude právě z tohoto místa. Tato metoda je proto velice nepřesná. Posledním způsobem je zjišťování polohy pomocí datových sítí, jako jsou WIFI sítě nebo mobilní sítě operátorů. Zjištění probíhá tak, že se zjistí

vzdálenost zařízení z několika těchto přístupových bodů – vysílačů. Přesnost této metody je ovlivněna počtem těchto vysílacích stanic v okolí zařízení, ze kterého se dotazujeme. Čím více přístupových bodů, tím větší přesnost získané polohy.

## 7 Funkce HTML5 Geolocation API

V předchozí kapitole jsem představil HTML5 Geolocation API. V této kapitole ukáži, jak s tímto API pracovat. Použití tohoto API je velice snadné a dobře se s ním pracuje.

HTML5 Geolocation API lze ovládat pomocí javascriptu. Jelikož se jedná o webovou technologii, je potřeba chápat základní vztahy a pravidla tvorby webu. Opět je důležité si uvědomit, že toto API a celkově celé HTML5 je v experimentální fázi, takže při použití je třeba přemýšlet, zda-li je možné danou technologii použít nebo ne.

### 7.1 Ověření podpory Geolocation API v prohlížeči

HTML5 dovoluje jednoduše a elegantně otestovat podporu HTML5 Geolocation API v prohlížeči. Je možné použít javascriptovou knihovnu Modernizr, ale v tomto případě je mnohem snazší použít jednoduchou javascriptovou podmínku. Vždy je důležité otestovat, zda-li uživatelův prohlížeč podporuje Geolocation API a zda-li uživatel dal svolení ke zjištění jeho polohy. V případě, že nastane jedna z těchto událostí, je třeba o tom dát uživateli jasně vědět.

*Příklad 1 – Ověření podpory Geolocation API*

```
if (navigator.geolocation) {  
    /* Prohlížeč podporuje Geolocation API a uživatel dal  
    svolení pro zjištění polohy */  
} else {  
    /* Prohlížeč nepodporuje Geolocation API nebo uživatel  
    nedal svolení pro zjištění polohy */  
}
```

*Příklad 2 – Funkční zápis ověření podpory Geolocation API*

```
if (navigator.geolocation) {  
    var options = {timeout:60000};  
    navigator.geolocation.getCurrentPosition(showLocation, errorHandler,  
    options);  
} else {  
    document.getElementById("error").innerHTML = "Podpora pro geolokaci  
    není dostupná."  
}
```

Na příkladu číslo 1 je uvedeno, jak správně zapsat podmínku pro ověření podpory Geolokace. Na příkladu číslo 2 je funkční zápis tohoto ověření. V podmínce se kontroluje podpora Geolokace. Objekt navigator je objekt, který uchovává informace o prohlížeči, ze kterého se dotazujeme. Tento objekt obsahuje informace, jako jsou název prohlížeče, verze prohlížeče, pro jakou platformu je prohlížeč kompilován atd.<sup>6</sup> Pokud tedy prohlížeč podporuje Geolocation API a uživatel dal svolení ke zjištění jeho polohy, lze zavolat funkci pro zjišťování polohy - `getCurrentPosition`, která obsahuje 3 parametry (`showLocation`, `errorHandler`, `options`). Toto celé se musí stát do 60000 milisekund, pokud se tak nestane, automaticky je podmínka nesplněna, protože nebyl splněn časový limit. Pokud tedy nastane jedna z výše popsaných možností, tak do divu, který bude mít identifikátor „error“, vpiše se hláška: “Podpora pro geolokaci není dostupná.” Lze si vymyslet cokoli, je možné použít objekt

---

<sup>6</sup>Více na: [http://www.w3schools.com/jsref/obj\\_navigator.asp](http://www.w3schools.com/jsref/obj_navigator.asp)

alert a vyhodit hlášku do message boxu, ale já jsem zvolil metodu zápisu do HTML stránky. Pokud uživatel nechce použít pouze chybovou hlášku, ale chce geolokaci simulovat, může použít polyfill, který nahradí nativní Geolocation API. Mezi tyto polyfilly patří například Geolocation polyfill<sup>7</sup>. Tato javascriptová knihovna zajistí funkčnost zjišťování polohy i v prohlížečích, které nativní HTML5 Geolocation API nepodporují. Je třeba ovšem počítat s tím, že funkčnost nemusí být stejná jako u HTML5 Geolocation API.

## 7.2 Získání polohy ve webovém prohlížeči

Po úspěšném ověření podpory HTML5 Geolocation API lze přejít do fáze získání aktuální polohy. O vyvolání požadavku na zjištění polohy se stará metoda `getCurrentPosition`. Tato metoda zajistí získání polohy od prohlížeče, který využije své dostupné zdroje. Metoda `getCurrentPosition` obsahuje tři parametry: `showLocation`, `errorHandler` a `options`.

*Příklad 3 – Zápis metody `getCurrentPosition()`*

```
getCurrentPosition(showLocation, ErrorHandler, options);
```

---

<sup>7</sup>Podrobnosti: <https://github.com/inexorabletash/polyfill/blob/master/geo.js>

### 7.2.1 Parametr Showlocation

Parametr showlocation je callback metoda, které je předán objekt po zjištění polohy. Objekt, který je předán callbacku (v našem případě showLocation()), je potomek třídy, která implementuje rozhraní Position<sup>8</sup>. Rozhraní Position obsahuje dva atributy, které jsou pouze pro čtení. Tyto atributy jsou timestamp a coords. Atribut timestamp v sobě uchovává časový otisk zjištění aktuální polohy. Atribut coords implementuje další rozhraní, které nese název Coordinates. Rozhraní Coordinates obsahuje tyto atributy:

- Latitude
- Longitude
- Altitude
- Accuracy
- Altitude accuracy
- Heading
- Speed

Atribut Latitude nese informace o zeměpisné šířce a formátem tohoto údaje je desetinné místo. Tudíž pokud chceme získat přesnou délku, je třeba tento údaj převést na minuty a sekundy. Stejně tak atribut Longitude, který nese informace o zeměpisné délce, je třeba převést, pokud chceme získat přesný formát s minutami a sekundami. Další atribut je Altitude, který obsahuje údaj o nadmořské výšce zařízení, ze kterého se dotazujeme. Jestliže zařízení nemůže získat údaj o nadmořské výšce, tak atribut vrátí prázdnou hodnotu, respektive hodnotu null. Atribut Accuracy udává přesnost zaměření zeměpisné šířky a délky v metrech. Altitude accuracy, jak už název napovídá, plní stejnou funkci jako Accuracy, ale Altitude accuracy udává

---

<sup>8</sup>Geolokace v prohlížeči — Zdroják. In: *Zdroják — o tvorbě webových stránek a aplikací* [online]. 2010 [cit. 2013-03-20]. Dostupné z: <http://www.zdrojak.cz/clanky/geolokace-v-prohlizeci/>

přesnost naměřené nadmořské výšky a hodnota je opět v metrech. Atribut Heading vrací údaj o směru natočení zařízení, ze kterého se dotazujeme. Získání údaje o natočení funguje tak, že vrátí hodnotu úhlu, o který je zařízení natočeno od severu, tedy pokud budeme natočeni na jih, hodnota pootočení bude 180°. Pokud zařízení nedokáže zjistit polohu otočení nebo se jedná o statické zařízení, hodnota atributu bude null<sup>9</sup>. Posledním atributem rozhraní Coordinates je atribut speed. Tento atribut sleduje rychlost horizontálního pohybu zařízení. Tedy, pokud se zařízení pohybuje, tak tento atribut vrátí hodnotu rychlosti pohybu v metrech za sekundu.

#### *Příklad 4 – Jednoduchý výpis callback funkce showLocation*

```
function showLocation(position) {
    var latitude = position.coords.latitude;
    var longitude = position.coords.longitude;
    var altitude = position.coords.altitude;
    var accuracy = position.coords.accuracy;
    var altitudeAccuracy = position.coords.altitudeAccuracy;
    var heading = position.coords.heading;
    var speed = position.coords.speed;
    document.getElementById("result").innerHTML = "Šířka: " + latitude +
    "Délka: " + longitude + "Výška: " + altitude + "Přesnost: " + accuracy +
    "Přesnost Výšky: " + altitudeAccuracy + "Otočení: " + heading +
    "Rychlost: " + speed;
}
```

Funkce v příkladu 4 do elementu s identifikátorem result vypíše zjištěné hodnoty. Funkce zobrazí údaje o zeměpisné šířce a délce, nadmořské výšce, přesnosti naměřených údajů, pootočení zařízení vůči severu a rychlosti pohybu zařízení.

---

<sup>9</sup>5.4 Coordinates interface. In: *Geolocation API Specification* [online]. 2012 [cit. 2013-03-20]. Dostupné z: <http://dev.w3.org/geo/api/spec-source.html>

## 7.2.2 Parametr ErrorHandler

Dalším parametrem metody `getCurrentPosition` je parametr `ErrorHandler`. Jedná se o callback metodu, která je zavolána, pokud dojde k nějakému problému při zjišťování polohy zařízení. Jedná se o nepovinnou metodu, která umožňuje zachytit a ošetřit chyby, které mohou při zjišťování polohy nastat. Je důležité tuto metodu správně a pokaždé použít, protože díky ní můžeme uživatele elegantně a srozumitelně informovat o vzniklé chybě při pokusu zjistit polohu zařízení. `ErrorHandler` implementuje rozhraní `PositionError`, které obsahuje dva argumenty, a to `code` s kódem chyby a `message` s detailním popisem chyby<sup>10</sup>. Rozhraní `PositionError` obsahuje tyto atributy:

- `PERMISSION_DENIED = 1`
- `POSITION_UNAVAILABLE = 2`
- `TIMEOUT = 3`
- `Code`
- `Message`

Atribut `code` musí vždy nést číselný údaj, který patří dané chybě, tedy prvním třem atributům. První atribut `PERMISSION_DENIED` znamená, že uživatel nedal svolení ke zjištění jeho polohy.

Druhý atribut `POSITION_UNAVAILABLE` znamená, že aktuální pozice je nedosažitelná, tedy že jí nelze zjistit. A poslední atribut `TIMEOUT` znamená, že zjišťování trvá příliš dlouho. Ošetření chyb lze jednoduše ošetřit za pomoci konstrukčního příkazu `SWITCH`.

---

<sup>10</sup>HTML5 – geolokační rozhraní. *Programujte.com - web o programování, webdesignu, počítačové grafice, databázích, elektrotechnice a designu* [online]. 2012 [cit. 2013-03-14]. Dostupné z: <http://programujte.com/clanek/2011052400-html5-geolokacni-rozhrani/>



### *Příklad 5 – Jednoduchý výpis callback funkce ErrorHandler*

```
function ErrorHandler(error) {
    switch (error.code) {
        case 1:
            document.getElementById("error").innerHTML =
                "Přístup byl odepřen!"
            break;
        case 2:
            document.getElementById("error").innerHTML =
                "Pozici nelze zjistit!"
            break;
        case 3:
            document.getElementById("error").innerHTML =
                "Vypršel časový limit žádosti!"
            break;
        default:
            alert("Chyba: " + error.message);
            break;
    }
}
```

Tato funkce se stará o obstarávání chyb. Při vzniku nějaké chyby uživatele informuje o tom, co se stalo. Při vzniku chyby `PERMISSION_DENIED` funkce vypíše do HTML elementu s ID `error` chybovou hlášku – “Přístup byl odepřen!” Pokud nastane chyba `POSITION_UNAVAILABLE`, do stejného HTML elementu vypíše hlášku - “Pozici nelze zjistit!” A pokud bude zjišťování trvat nepřiměřeně dlouho, funkce vypíše - “Vypršel časový limit žádosti!”

### **7.2.3 Parametr Options**

Dalším parametrem metody `getCurrentPosition` je parametr `option`. Tento parametr implementuje rozhraní `PositionOptions`, který umožňuje zvolit si

rozšířené nastavení geolokace ve webovém prohlížeči. Rozhraní `PositionOptions` obsahuje tyto rozšiřující atributy nastavení:

- `enableHighAccuracy`
- `timeout`
- `maximumAge`

Atribut `enableHighAccuracy` je typu `boolean`, a jestliže je nastaven na hodnotu `TRUE`, tak při zjišťování polohy se bude aplikace vždy snažit o využití nejpřesnějšího zdroje pro zjištění polohy – GPS modulu, pokud bude k dispozici. Atributem `timeout` lze nastavit hodnotu času v milisekundách pro zjištění polohy. Pokud je tento časový limit překročený, vyvolá se chyba `TIMEOUT`. Poslední atribut `maximumAge` nese časový údaj opět v milisekundách o tom, jak starou informaci o poloze může prohlížeč použít z cache paměti prohlížeče. Použití parametru `options` demonstruje následující příklad.

*Příklad 6 – Použití parametru `options`*

```
getCurrentPosition (showLocation, ErrorHandler,  
{ enableHighAccuracy: true, timeout: 500000, maximumAge: 0 });
```

#### 7.2.4 Ukázkový příklad jednorázového získání polohy

Všechno nejlépe vysvětlí ukázkový příklad. Pro tvorbu ukázkových příkladů jsem se rozhodl použít front-end responzivní framework `Foundation`<sup>11</sup>. Jedná se o připravenou šablonu HTML5 webové stránky. `Foundation` ušetří při vývoji spoustu práce, protože obsahuje připravenou stromovou strukturu webové stránky. Dále obsahuje připravené styly pro globální HTML elementy, knihovnu `jQuery` a samozřejmě také obsahuje styl `normalize.css` – moderní náhradu za `CSS reset`. Dále v sobě implementuje knihovnu `Modernizr`, kterou jsem již zmiňoval a mnoho dalších podpůrných funkcí, vlastností a knihoven.

<sup>11</sup>Ke stažení: <http://foundation.zurb.com/>

*Příklad jednorázového zjištění polohy (příklad-1\_Jednorazove-zjistení)*

```
<script type="text/javascript">
var map;

// Zjištění polohy =====
function showLocation(position) {
    var latitude = position.coords.latitude;
    var longitude = position.coords.longitude;
    var accuracy = position.coords.accuracy;
    var altitude = position.coords.altitude;
    var altitudeAccuracy = position.coords.altitudeAccuracy;
    var heading = position.coords.heading;
    var speed = position.coords.speed;
    document.getElementById("result").innerHTML =
    "Šířka: " + latitude +
    "<br/>Délka: " + longitude +
    "<br/>Výška: " + altitude +
    "<br/>Přesnost: " + accuracy +
    "<br/>Přesnost Výšky: " + altitudeAccuracy +
    "<br/>Otočení:" + heading +
    "<br/>Rychlost: " + speed;
}

// Ošetření chyb =====
function errorHandler(error) {
    switch (error.code) {
        case 1:
            document.getElementById("error").innerHTML =
            "Chyba: Přístup byl odepřen.";
            break;
        case 2:
```

```

        document.getElementById("error").innerHTML =
            "Chyba: Poloha není k dispozici.";
        break;
    case 3:
        document.getElementById("error").innerHTML =
            "Chyba: Vypršel časový limit.";
        break;
    default:
        document.getElementById("error").innerHTML =
            "Chyba: " + error.message;
        break;
    }
}

// Volání polohy =====
function getLocation(){
    if (navigator.geolocation) {
        var options = maximumAge:600000, timeout:15000,
            enableHighAccuracy: true;
        navigator.geolocation.getCurrentPosition(
            showLocation, errorHandler, options);
    } else {
        document.getElementById("error").innerHTML =
            "Podpora pro geolokaci není dostupná.";
    }
}

</script>
<body onload="getLocation()">

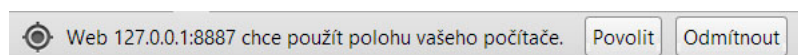
<div class="row">
<div class="large-12 columns">
<h2>Příklad 1 - Jednorázové zjištění polohy</h2>

```

```
<p class="small">Geolokace ve webovém  
prohlížeči pomocí HTML5 Geolocation API</p>  
<hr />  
</div>  
</div>  
  
<div class="row">  
<div class="large-12 columns">  
    <p id="result"></p>  
    <p id="error"></p>  
<a href="" class="button">Zpět</a>  
</div>  
</div>  
</body>
```

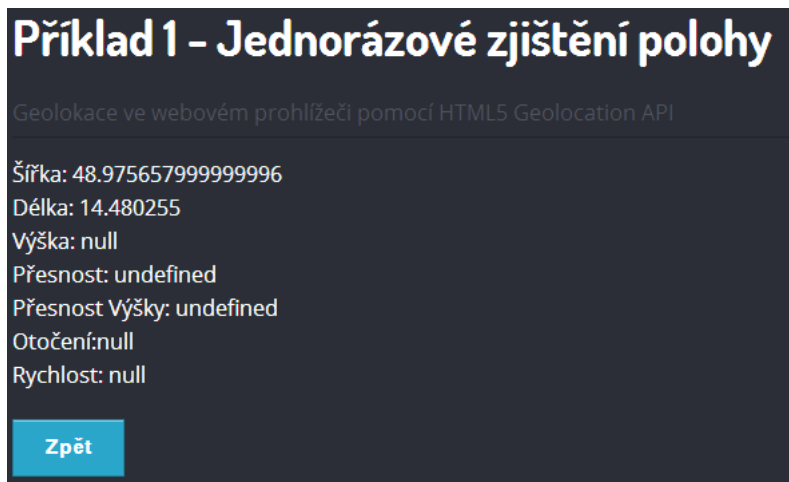
Tento příklad demonstruje jednorázové zjištění polohy ve webovém prohlížeči. Jedná se o nejzákladnější ukázkou geolokace ve webovém prohlížeči. Příklad zajišťuje zjištění polohy a její výpis do těla dokumentu. V případě výskytu chyby se vypíše popis chyby.

Po spuštění příkladu musíme potvrdit, že souhlasíme se zjištěním naší polohy.



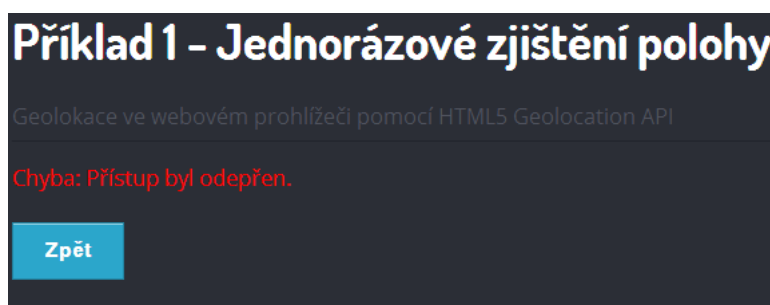
Obrázek 1: Požadavek prohlížeče o povolení zjištění vaší polohy

Po kliknutí na tlačítko se nám zobrazí výsledek zjišťování polohy.



Obrázek 2: Výpis výsledku zjišťování polohy

Jelikož jsem testoval tento příklad na stolním počítači bez GPS modulu, tak se nepodařilo zjistit nadmořskou výšku, přesnost, přesnost nadmořské výšky, otočení a rychlost. Ale zeměpisnou šířku a délku se podařilo získat bez problému. Pokud by se vyskytla jakákoliv chyba, aplikace vypíše zprávu o chybě.



Obrázek 3: Chybové hlášení

Zde se nepodařilo získat autorizaci ke zjištění polohy, proto aplikace vyhodila chybovou hlášku – "Přístup byl odepřen."

### 7.3 Periodické získávání polohy ve webovém prohlížeči

Polohu zařízení nemusíme zjišťovat pouze jednorázově, ale díky metodě `watchPosition()` můžeme polohu zaznamenávat i v periodických intervalech. Tato metoda plní stejnou funkci jako metoda `getCurrentPosition` jen s tím rozdílem, že aktuální polohu získává pravidelně. Zápis metody je stejný, jako je u metody `getCurrentPosition`.

*Příklad 7 – Zápis metody `watchPosition()`*

```
watchPosition (showLocation, ErrorHandler, options);
```

Použití je tedy stejné jako u metody `getCurrentPosition()`. Jediný rozdíl je v tom, že metoda `watchPosition` jako hodnotu vrací identifikátor požadavku kvůli pozdějšímu zrušení.<sup>12</sup>

Postup tvorby aplikace, která bude periodicky zjišťovat polohu zařízení, je podobný jako při tvorbě aplikace pro jednorázové zjišťování. Je zde jen několik věcí navíc. Za prvé je do těla dokumentu přidáno tlačítko, které bude zapínat zjišťování.

```
<input type="button" onclick="getLocationUpdate()" value="Start">
```

---

<sup>12</sup>HTML5 – geolokační rozhraní. *Programujte.com - web o programování, webdesignu, počítačové grafice, databázích, elektrotechnice a designu* [online]. 2012 [cit. 2013-03-14]. Dostupné z: <http://programujte.com/clanek/2011052400-html5-geolokacni-rozhrani/>

Na začátku bloku s javascriptem se nadefinují dvě proměnné watchID a navGeo.

```
<script type="text/javascript">
var watchID;
var navGeo;
```

A pak pouze stačí změnit funkci getLocationUpdate, která by měla vypadat takto:

```
function getLocationUpdate() {
  if (navigator.geolocation) {
    navGeo = navigator.geolocation;
    watchID = navGeo.watchPosition(showLocation);
  }else{
    document.getElementById("error").innerHTML =
      "Podpora pro geolokaci není dostupná.";
  }
}
```



## 8 HTML5 Geolocation API s různými mapovými podklady

HTML5 geolocation API toho samo o sobě příliš nedokáže. Teprve s použitím mapových podkladů dostává toto API úplně jiný rozměr, a to protože dokážeme zjištěné výsledky promítnout do mapových podkladů a zobrazit tak mnohem přívětivější informace než pouhé zeměpisné souřadnice. Další výhodou je možnost použití samotného API podkladových map. Je tedy možné aplikace přizpůsobit přesně požadovaným potřebám.

Práce s HTML5 Geolocation API a mapovými podklady je velice snadná. Stačí pouze webová stránka, která bude obsahovat blok s názvem *map\_canvas*. Název je samozřejmě možné zvolit libovolně.

```
<div id="map_canvas"></div>
```

Do tohoto bloku bude nahrávána mapa a vypisovány získané údaje. Tento blok samozřejmě lze pomocí CSS stylovat. Je možné mu například dát určitou šířku a délku, barvu a velikost rámečku nebo nějakou CSS3 vlastnost jako například vržený stín apod.

Jako mapové podklady, na kterých budu demonstrovat použití s HTML5 Geolocation API, jsem vybral následující:

- Google Maps
- Mapy.cz
- Bing Maps
- OpenLayers

Jedná se o výběr nejznámějších dostupných mapových podkladů, které se dají na internetu nalézt. Veškerý problém spočívá pouze v tom, že ve funkci

showLocation přidáme kód, který vytvoří, nastaví a zobrazí mapu. A pak už pouze do pozice, kterou HTML5 Geolocation vrátí, vypíše grafický ukazatel.

## 8.1 HTML5 Geolocation API a Google Maps

Začnu tedy nejrozšířenějšími mapovými podklady na internetu, a to Google Maps. Veškeré informace o Google Maps API nalezneme zde:

<https://developers.google.com/maps/documentation/javascript/?hl=cs>

Nejprve je nutné nalinkovat Google Maps API do naší stránky:

```
<script src="https://maps.googleapis.com/maps/api/  
js?v=3.exp&sensor=false"></script>
```

Poté se do funkce showLocation přidají následující řádky kódu:

```
var myLatLng = new google.maps.LatLng(latitude, longitude);
```

```
    var mapOptions =  
        zoom: 16,  
        center: myLatLng,  
        mapTypeId: google.maps.MapTypeId.HYBRID
```

```
var map = new google.maps.Map(document.getElementById  
("map_canvas"), mapOptions);
```

```
var marker = new google.maps.Marker(  
    position: myLatLng,  
    map: map,  
    title:"Hello World!");
```

V prvním kroku jsou přebrány údaje naměřené HTML5 Geolocation API a jsou uloženy do proměnné. V dalším kroku se nastavuje již samotná mapa.

Zoom znamená přiblížení mapy, center určuje, kam se umístí střed mapy, v tomto případě se mapa vycentruje na souřadnice naměřené Geolocation API. Poslední parametr nám určuje typ mapy. Další možnosti lze nalézt v dokumentaci. V dalším kroku je mapa „vepsána“ do vytvořeného oddílu, který byl nazván `map_canvas`. A v poslední části kódu se pouze vytvoří grafický ukazatel pozice, který je umístěn na naměřených souřadnicích. Výsledkem je tedy mapa s ukazatelem aktuální pozice.<sup>13</sup>

## 8.2 HTML5 Geolocation API a Mapy.cz

Dalšími mapovými podklady jsou, v České republice dobře známé, Mapy.cz. Dokumentace je k nalezení zde:

<http://api4.mapy.cz/>

Opět je nutné nalinkovat toto API do vytvořené stránky:

```
<script src="http://api4.mapy.cz/loader.js"></script>
<script>Loader.load()</script>
```

Funkce `showLocation` pak bude vypadat takto:

```
var lonD = longitude;
var latD = latitude;
var center = SMap.Coords.fromWGS84(lonD, latD);
var m = new SMap(JAK.gel("map_canvas"), center, 14);

    m.addDefaultLayer(SMap.DEF_OPHOTO);
    m.addDefaultLayer(SMap.DEF_OPHOTO0203);
    m.addDefaultLayer(SMap.DEF_OPHOTO0406);
    m.addDefaultLayer(SMap.DEF_TURIST);
    m.addDefaultLayer(SMap.DEF_HISTORIC);
    m.addDefaultLayer(SMap.DEF_BASE).enable();
    m.addDefaultControls();
```

---

<sup>13</sup>Praktická ukázka: <http://bp.martinlaudat.cz/priklady/3-google-maps-api/>

```

var layerSwitch = new SMap.Control.Layer();
layerSwitch.addDefaultLayer(SMap.DEF_BASE);
layerSwitch.addDefaultLayer(SMap.DEF_OPHOTO);
layerSwitch.addDefaultLayer(SMap.DEF_TURIST);
layerSwitch.addDefaultLayer(SMap.DEF_OPHOTO0406);
layerSwitch.addDefaultLayer(SMap.DEF_OPHOTO0203);
layerSwitch.addDefaultLayer(SMap.DEF_HISTORIC);
m.addControl(layerSwitch, left:"8px", top:"9px");

var layer = new SMap.Layer.Marker();
m.addLayer(layer);
layer.enable();

var options = ;
var marker = new SMap.Marker(center, "myMarker", options);
layer.addMarker(marker)

```

Princip je naprosto stejný jako u Google Maps. Opět jsou naměřené hodnoty předány do proměnných, mapa je vycentrována přesně na naměřenou pozici. Je vytvořena mapa s individuálním nastavením a je vytvořen a umístěn grafický ukazatel na naměřené souřadnice. Na první pohled je vidět, že kód je delší než u Google Maps, ale jedná se o naprosto stejný postup, který se liší jen zápisem kvůli odlišnosti Google Maps API a Mapy.cz API. Opět by měla být výsledkem mapa s ukazatelem aktuální pozice.<sup>14</sup>

### 8.3 HTML5 Geolocation API a Bing Maps

Předposlední ukázkou je spojení HTML5 Geolocation API s mapovými podklady od Microsoftu – tedy mapami Bing Maps. Dokumentace Bing Maps je zde:

---

<sup>14</sup>Praktická ukázka: <http://bp.martinlaudat.cz/priklady/4-mapy.cz-api/>

<http://msdn.microsoft.com/en-us/library/dd877180.aspx>

Jako v předchozích příkladech i zde se musí nalinkovat API:

```
<script src="http://ecn.dev.virtualearth.net/mapcontrol/  
mapcontrol.ashx?v=6.3"></script>
```

Kód ve funkci showLocation bude vypadat následovně:

```
var map = null;  
var pinPoint = null;  
var pinPixel = null;  
  
var LA = new VELatLong(latitude, longitude);  
  
    map = new VEMap('map_canvas');  
    map.LoadMap(LA, 15, VEMapStyle.Road, false,  
    VEMapMode.Mode2D, true, 1);  
  
    pinPoint = map.GetCenter();  
    pinPixel = map.LatLongToPixel(pinPoint);  
    map.AddPushpin(pinPoint);  
    // Převzato z http://gazpo.com/gazpo-files/  
html5\_geolocation/bing.html
```

Také na tomto příkladě je vidět, že práce s různými mapovými podklady je velice snadná a podobná. A samozřejmě i zde je výsledkem mapa s grafickým ukazatelem aktuální pozice.<sup>15</sup>

---

<sup>15</sup>Praktická ukázka: <http://bp.martinlaudat.cz/priklady/5-bing-maps-api/>

## 8.4 HTML5 Geolocation API a OpenLayers

Závěrečnou ukázkou je kombinace HTML5 Geolocation API a OpenLayers. Dokumentace k OpenLayers je zde:

<http://trac.osgeo.org/openlayers/wiki/Documentation>

Také zde se musí nalinkovat API:

```
<script src="http://www.openlayers.org/api/OpenLayers.js"></script>
```

Kód ve funkci showLocation bude vypadat následovně:

```
var lon = longitude;
var lat = latitude;
var map = new OpenLayers.Map('map_canvas');

    map.addLayer(new OpenLayers.Layer.OSM());
    var lonLat = new OpenLayers.LonLat( lon ,lat )
        .transform(
            new OpenLayers.Projection("EPSG:4326"), //převod
            souřadnicových soustav pro pozdější použití
            map.getProjectionObject()
        );

    var markers = new OpenLayers.Layer.Markers( "Markers" );
    map.addLayer(markers);

    markers.addMarker(new OpenLayers.Marker(lonLat));

    map.setCenter (lonLat, 16);
}
```

Jak je vidět, tak princip nastavení je opět velice podobný předchozím příkladům. Jediné, co je trochu jiné, je to, že je nutné převést získané souřadnice do srozumitelnějšího formátu pro OpenLayers. Tento příklad by měl opět zobrazit mapu s grafickým ukazatelem aktuální pozice.<sup>16</sup>

Jak je vidět na všech příkladech, práce s mapovými podklady je velice jednoduchá a princip je u všech zmíněných podkladů stejný. Díky tomu je pouze na uživateli, jaké mapové podklady zvolí. Rozhoduje tedy spíše API samotných mapových podkladů. Ve většině případů bude potřeba udělat aplikaci složitější než jenom výpis polohy, a proto budou rozhodující možnosti konkrétního API, které mapové podklady nabízí.

---

<sup>16</sup>Praktická ukázka: <http://bp.martinlaudat.cz/priklady/6-openlayers-api/>

## 9 Metody lokalizace mobilních zařízení

Pro tvorbu aplikací, které budou zjišťovat polohu zařízení, je nutné nejdříve zjistit polohu zařízení, tj. kde se právě nachází. Bez těchto údajů nelze dále pracovat s HTML5 Geolocation API. Je hned několik možností, jak zjistit aktuální polohu zařízení. Mezi ty hlavní patří zjišťování polohy pomocí GPS hardware, zjištění polohy pomocí IP adresy zařízení a zjištění polohy pomocí datových sítí – WIFI a GSM sítě. Nejpřesnější výsledky lze získat se zařízením, které disponuje GPS hardwarem. V dnešní době je to většina chytrých telefonů a tabletů. Pokud ovšem zařízení nedisponuje GPS hardwarem, je nutné se pokusit o zjištění pomocí alternativních metod. Vyhledání zařízení pomocí datových sítí je relativně přesné hlavně v oblasti s vysokou silou signálu, a to ať už WIFI signálu, nebo signálu GSM sítí. Zjišťování polohy pomocí IP adresy je nejméně přesné, protože záleží na typu připojení. Může se tedy stát, že místo IP adresy se bude zobrazovat IP adresa poskytovatele připojení.

Je důležité ještě zmínit, že samotné Geolocation API se nestará o zjišťování polohy zařízení. Geolocation API pouze požádá prohlížeč a zjištění polohy a zbytek je na prohlížeči. Prohlížeč sám určí nejpřesnější zdroj lokace, který má k dispozici a ten využije pro zjištění polohy. Zjištěné údaje pak pouze předá Geolocation API, kde s nimi lze pracovat dál.

### 9.1 Lokalizace pomocí GPS Hardware

Jedná se o nejpřesnější metodu zjištění polohy. Tato metoda funguje na principu komunikace zařízení se satelity, které obíhají kolem Země. Jedná se o soustavu 27 družic z toho tři jsou záložní pro případ poruchy některé z družic. Družice obíhají kolem Země v šesti různých dráhách, ve výšce zhruba 20 000 km. Každá družice oběhne Zemi dvakrát denně (jeden oběh trvá 11 h 58 min )<sup>17</sup>.

---

<sup>17</sup>Jak funguje GPS?. In: *Svět hardware* [online]. 2006 [cit. 2013-06-05]. Dostupné z: <http://www.svethardware.cz/jak-funguje-gps/21826-4>



K určení polohy na Zemi stačí signál ze tří družic. Pro zjištění nadmořské výšky bude potřeba signál alespoň ze čtyř družic. Logicky platí pravidlo, že čím více družic se zařízením komunikuje, tím přesněji se poloha určí.

Každá družice vysílá elektromagnetický signál – mikrovlnný paprsek, který oznamuje její přítomnost každému, kdo má přijímač schopný tento signál přijmout.<sup>18</sup> Díky vzdálenosti družic existuje malé zpoždění od odeslání signálu družicí po příjem signálu zařízením. Díky tomuto zpoždění je možné určit vzdálenost zařízení od družice, a pokud jsou tato zpoždění spočítána alespoň ze tří družic, tak lze získat údaje zeměpisné polohy (šířka a délka) s přesností na několik metrů.

GPS přijímače se často označují jako pasivní. To proto, že tyto přijímače pouze přijímají signál z družic a zpět družicím nic neodesílají. GPS se dnes používá v mnoha oborech lidské činnosti, například v letectví, dopravě, zemědělství nebo pro vojenské účely. Ale oborů, kde se GPS používá, je samozřejmě mnohem více a další budou přibývat. A také díky moderním informačním technologiím mohou obyčejní lidé používat GPS pro běžné úkony, jako je například automobilová navigace.

## 9.2 Lokalizace pomocí IP adresy

Lokalizace pomocí IP adresy je nejméně přesná metoda lokalizace, protože se nezakládá na určení polohy, ale pouze zjišťuje údaje o držiteli IP adresy. Princip je takový, že po získání IP adresy se dotazujeme na databázi IP adres, komu testovaná adresa patří. Nejznámější takovou databází je databáze WHOIS<sup>19</sup>. Tato databáze je nejznámější a nejvíce rozšířená. Existují i další databáze, ale většinou jsou zpoplatněné. Dále se lze dotazovat na záznamy z DNS serverů nebo informace o ISP poskytovatele připojení. Největší nevýhoda je, že pokud uživatel nemá statickou IP adresu, ale na internet se připojuje přes IP adresu (bránu) poskytovatele připojení, tak výsledky vy-

---

<sup>18</sup>O technologii GPS. In: *Technologie Mio* [online]. 2012 [cit. 2013-06-05]. Dostupné z: [http://eu.mio.com/cs\\_cz/global-positioning-system\\_jak-gps-funguje.htm](http://eu.mio.com/cs_cz/global-positioning-system_jak-gps-funguje.htm)

<sup>19</sup>Adresa databáze: <http://whois.net/>

hledávání budou směřovat právě na poskytovatele a ne na uživatele. Přesnost se dá těžko určit, ale většinou lokalizace pomocí IP adresy trefí alespoň stát, ze kterého se dotazujeme.

### 9.3 Lokalizace pomocí datových sítí

Dalším způsobem jak lokalizovat zařízení, je lokalizace pomocí datových sítí. Jedná se o lokalizaci pomocí WIFI sítí a GSM. Lokalizace pomocí WIFI sítí probíhá tak, že se otiskne seznam všech viditelných hotspotů, které se nacházejí v okolí zařízení. Tyto informace se porovnají s databází a na základě shody MAC adres sítí se zjistí pozice zařízení. Na internetu je několik databází hotspotů, z nichž nejrozsáhlejší vytvořil Google. Velice záleží na počtu sítí v okolí a na síle signálu těchto sítí. Čím více bude v okolí sítí s dobrým signálem, tím přesnější výsledky získáme.

Mimo WIFI sítí lze lokalizovat zařízení i pomocí GSM sítí. Ovšem hlavní podmínkou je, aby zařízení, ze kterého se dotazujeme, mělo potřebný hardware pro podporu GSM sítí. Jedná se tedy hlavně o mobilní telefony a některé tablety, které nativně GSM síť podporují. Jsou tři základní způsoby zjištění polohy zařízení v síti GSM. První metodou je tzv. CGI (Cell Global Identity). Tento způsob využívá znalostí operátora o rozmístění svých vysílačů, tedy BTS stanic. Operátor, který GSM síť provozuje, zná přesné polohy svých BTS a díky tomu dokáže určit polohu zařízení. Bohužel je tato metoda nepřesná.



Obrázek 4: Zjištění polohy zařízení metodou CGI

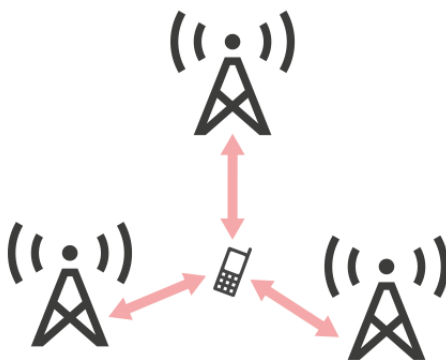
Jak je vidět na obrázku číslo 4, tato metoda zjištění polohy nám určí pouze oblast, ve které se zařízení nachází. Proto se nejedná o přesnou metodu.

Další a mnohem přesnější metodou je metoda CGI+TA (Cell Global Identity + Timing Advance) která, jak již název napovídá, používá pro zjištění polohy ještě jeden parametr - TA. Tedy Timing advance, což je časový údaj zpoždění signálu mezi zařízením a BTS stanicí. Díky tomuto parametru je možné polohu určit mnohem přesněji, protože lze určit i vzdálenost zařízení od BTS, a to dokonce s přesností zhruba 550m.



Obrázek 5: Zjištění polohy zařízení metodou CGI+TA

Poslední metodou, o které bych se chtěl zmínit je metoda UL-TOA (Uplink Time of Arrival). Tato metoda zjišťuje pouze časový interval komunikace zařízení se sítí, tedy časový rozdíl, kdy zařízení začalo vysílat a kdy dorazil signál k vysílači. Pokud je zařízení v dostupnosti alespoň tří vysílačů, lze určit přesnost zhruba na 150 metrů. Jak to zhruba probíhá, je vidět na obrázku číslo 6.



Obrázek 6: Zjištění polohy zařízení metodou UL-TOA

Je ovšem třeba zdůraznit, že přesného zaměření lze dosáhnout pouze v ideálních podmínkách. Přesnost měření například ovlivňují odrazy nebo zlomení signálu. Proto tato metoda není moc vhodná v městském prostředí.

## 10 Podpora HTML5 Geolocation API v prohlížečích

Při tvorbě každé webové aplikace nebo webové stránky je důležité myslet na to, že ne všichni používají stejný operační systém, stejný prohlížeč nebo stejné rozlišení. Proto je nutné dbát na to, aby to, co vznikne, bylo funkční na většině zařízení, ze kterých mohou uživatelé na web přistupovat. Pro nás je nejdůležitější podpora prohlížečů. Jelikož je Geolocation API a HTML5 relativně mladá technologie, tak je možné, že nebude fungovat ve všech prohlížečích, respektive ve starších prohlížečích.

Dnes se na trhu webových prohlížečů pohybuje několik velkých hráčů, o kterých se má smysl bavit. Jedná se Google Chrome, Mozilla Firefox, Opera, Safari a Internet Explorer. Dále tyto prohlížeče můžeme rozdělit na moderní a zastaralé. Do kategorie moderních prohlížečů patří pouze poslední verze zmíněných prohlížečů a do kategorie zastaralých logicky prohlížeče neaktuální.

Nejdříve popíši podporu v moderních prohlížečích. HTML5 Geolocation API je podporováno ve všech moderních prohlížečích. Testování jsem provedl v následujících verzích, které jsou právě aktuální:

- Google Chrome 27
- Mozilla Firefox 21
- Opera 12.15
- Safari 6.0.5
- Internet Explorer 10

Ve všech těchto prohlížečích byla funkčnost naprosto bezproblémová. Dá se tedy říci, že podpora geolokace je velmi dobrá. Dá se tedy předpokládat, že v budoucích verzích prohlížečů bude podpora také, pokud se výrobci prohlížečů z nějakých důvodů nerozhodnou geolokaci odebrat, ale to je velice nepravděpodobné.

A jaká je podpora v zastaralých verzích prohlížečů? Rozhodl jsem se testovat pouze starší verze Internet Exploreru, protože díky automatickým aktualizacím není potřeba testovat ostatní prohlížeče. Je jasné, že někteří uživatelé mají automatické aktualizace vypnuté, ale to je tak zanedbatelná část, že není nutné se tím zabývat. Pro tuto práci tedy bude rozhodující Internet Explorer, jehož starší verze se stále ještě hojně používají. Je to logické, protože většina uživatelů nepotřebuje jiný prohlížeč nebo z nějakých důvodů nemůže svůj prohlížeč aktualizovat. A proto se dnes na počítačích stále vyskytují verze Internet Exploreru 7 nebo 8. Dle serveru [statcounter.com](http://statcounter.com)<sup>20</sup> je podíl Internet Exploreru 8 i dnes uctihodných 11,83 % a podíl Internet Exploreru 9 16,14 %. Je tedy potřeba se těmito prohlížeči i nadále zabývat a připravovat webové stránky nebo aplikace i pro tyto prohlížeče.

Provedl jsem tedy testy v Internet Exploreru 8 a 9. Ve verzi 9 funguje geolokace bez jakýchkoliv problémů. V Internet Exploreru 8 už bohužel tak pozitivní výsledky nejsou, protože zde geolokace nefunguje vůbec. Takže pro Internet Explorer 8 je nutno používat nějakou simulaci - polyfill. Je možné zvolit například polyfill [Geo.js](https://github.com/inexorabletash/polyfill/blob/master/geo.js)<sup>21</sup>. Jedná se o javascriptovou knihovnu, která simuluje HTML5 Geolocation API. Je jasné, že funkčnost nebude úplně stejná jako s použitím samotného Geolocation API, ale ve většině případů to může pomoci vytvořit aplikace funkční i v Internet Exploreru 8 a starších verzích.

---

<sup>20</sup>Veškeré statistiky: [gs.statcounter.com](http://gs.statcounter.com)

<sup>21</sup>Stážení a informace: <https://github.com/inexorabletash/polyfill/blob/master/geo.js>

## 11 Využití geolokace na internetu

Tato kapitola se bude zabývat možnostmi využití geolokace na internetu, které ukáží na reálných službách. S příchodem moderních technologií pro tvorbu webových stránek se rozšířily možnosti tvorby webových aplikací, a proto vzniklo spousta nových aplikací a služeb. Vývojáři si dobře uvědomují sílu nových možností, které mají. A právě díky těmto vývojářům vznikají aplikace, o kterých se nikomu před několika lety ani nezdálo. A vývoj webových technologií jde stále dopředu, a proto budou dále vznikat nové a modernější aplikace než doposud. Mezi dnešní moderní aplikace se dají zařadit sociální sítě, aplikace nahrazující desktopové programy, různá webová uložení a v neposlední řadě několik služeb využívajících geolokaci. Je to důsledkem toho, že si spousta lidí uvědomila, že dokáží vytvořit aplikace, které budou užitečné a dokáží generovat nějaké zisky. Samozřejmě ne všichni to dělají pro peníze, ale ti největší hráči na "dnešním" webu - sociální sítě a různé služby - generují miliardové zisky. Je tedy zřejmé, že tvořit moderní aplikace má v dnešní době opravdu smysl.

Jako další využití geolokace na internetu vidím usnadnění uživatelského zážitku z webu. Dnešní webové stránky a aplikace by měly být tzv. user-centered, tedy měly by to být takové aplikace, které se uživateli dobře ovládají a uživatel z nich má dobrý pocit. Rozepsat všechny kroky návrhu kvalitního rozhraní a uživatelského zážitku by zaplnilo minimálně ještě jednu bakalářskou práci. Jedná se totiž o několik komplikovaných kroků, které ovšem přinášejí jasné výsledky, pokud je dokážeme správně využít. Tyto kroky dnes bezpodmínečně patří do moderního workflow vytváření webových aplikací. HTML5 Geolocation API při vytváření kvalitního uživatelského zážitku může velice pomoci, ale také uškodit. Pokud se nedokážeme zamyslet nad uživatelskými potřebami, pak tato technologie může uškodit. V opačném případě může velice pomoci. Díky této technologii se může uživateli zobrazit spousta informací, aniž by musel odejít z webu na jiný web. Například je možné přímo ve webové stránce vygenerovat trasu od uživatele do cíle (např. sídlo firmy),

a to vše na našem webu a na všech zařízeních. Díky tomu získá uživatel okamžitě výsledek jeho interakce a hned se dozví, co potřebuje, aniž by opustil web. A toto byl pouze jeden příklad z mnoha, kdy tyto technologie dokáží uživateli pomoci. Díky napojení na mapové podklady lze vytvořit aplikace, se kterými se bude uživateli snadno pracovat. A je jenom na nás, abychom toto všechno dokázali využít v náš prospěch.

V dnešní době tvoří jeden z největší zisků na internetu zisk z reklam. Reklamy jsou všude a dokonce je dnes více reklam než prostoru, kde mohou být zveřejněny. V reklamách na internetu se točí obrovské množství peněz, a proto je každá dobře cílená reklama velmi cenná. Díky HTML5 Geolocation API je možné vytvořit reklamní systém, který bude zobrazovat reklamy na základě pozice zařízení. Lze přesně určit polohu uživatele a podstrčit mu reklamy služeb či podniků z jeho bezprostředního okolí. Toto řešení má určitě mnohem větší smysl a bude mnohem účinnější než dosavadní reklamní systémy, i když například Google Adwords takto pracují také.

Cílem této práce je rozebrat a vysvětlit problematiku HTML5 Geolocation API, proto mě budou zajímat aplikace, které tuto technologii využívají. Takových to aplikací je spousta, ale já jsem se snažil vybrat ty nejdůležitější a nejpoužívanější v České republice. Vybral jsem tedy z českých zástupců službu Dáme jídlo a ze zahraničních populární sociální síť Foursquare, službu TripAdvisor a ještě jednu hru která je založená na geolokaci - Geocaching. Jak jsem již zmiňoval, aplikací a služeb je na internetu spousta, ale já se pokusil vybrat nejpoužívanější. Dále bych ještě rád zmínil hru od společnosti Google, která se jmenuje Ingress<sup>22</sup>. Tato hra zatím ještě není veřejně dostupná, ale mělo by se jednat o zajímavou geolokační hru o objevování portálu. Dle prvních recenzí si troufám říct, že něco takového dosud neexistuje a rozhodně se bude jednat o zajímavou hru.

---

<sup>22</sup>více na: <http://www.ingress.com>

## 11.1 Dáme jídlo

Dáme jídlo (dříve PizzaTime.cz) je služba, za kterou stojí Tomáš Čupr, který také založil Slevomat. Jedná se o agregátor dovozu hotových jídel. Tato služba tedy agreguje restaurace a zprostředkovává rozvoz jídel k zákazníkům. Díky tomu tedy nemusím složitě prohledávat webové stránky restaurací, ale mám pěkně všechno na jednom místě. Samozřejmostí jsou mobilní aplikace pro všechny dnešní platformy. A právě v těchto mobilních aplikacích se používá geolokace, která zajistí to, že po spuštění aplikace se zobrazí nabídky restaurací v okolí zařízení, ze kterého se dotazujeme. To tedy znamená že se k uživateli dostanou pouze restaurace v okolí, a tudíž není potřeba filtrovat ze všech výsledků jen ty restaurace, které jsou v blízkosti zařízení. Tato služba nabízí i další funkce jako hodnocení, sbírání kreditů atd. Ale pro účel této práce je důležitý pouze smysl geolokace, kterou aplikace používá. Více podrobností lze dohledat na oficiálních stránkách - <http://www.damejidlo.cz>.

## 11.2 Foursquare

Foursquare je asi neznámější službou, která využívá geolokaci. Jedná se o sociální síť, která je založená na zveřejňování polohy. Tím, že zveřejníte svou polohu (tzv. check-in) na určitém místě, získáte body. Pokud například některé místo navštívíte několikrát, můžete odemknout jednu z tzv. badges, tedy odznak. Například odznak za navštívení místa, na kterém jste ještě nikdy nebyl. A pokud chodíte na nějaké místo často, například do oblíbeného klubu, a pravidelně se tam checkujete, tak pokud budete mít nejvíce check-inů z osob, které ten klub navštěvují, tak se tam stanete starostou. Tedy osobou, která tam chodí nejčastěji a budete zveřejněni u profilu podniku. Můžete si také přidat přátele, kteří Foursquare rovněž používají a soutěžit s nimi o to, kdo nasbírá co nejvíce bodů. Foursquare je samozřejmě také dostupný jako aplikace pro všechny moderní platformy. Oficiální web je zde - <http://www.foursquare.com>.



### 11.3 TripAdvisor

TripAdvisor je služba především pro lidi, kteří často cestují. Díky této aplikaci lze dohledat recenze na hotely, restaurace nebo třeba letoviska. Díky dostupnosti mobilní aplikace na všechny dnešní mobilní platformy se z této služby stává ideální společník na cesty. Právě v mobilních aplikacích díky geolokaci lze rychle zjistit recenze či názory jiných lidí na služby či restaurace v okolí. Bohužel tato služba není moc známa v České republice a recenze na české podniky jsou napsány především v anglickém jazyce. Ale při cestách do zahraničí může být tato služba velice užitečná a může ušetřit i nějaké finance. Všechny informace lze nalézt na oficiálních stránkách - <http://www.tripadvisor.com>.

### 11.4 Geocaching

Geocaching je turistická hra, kde je hlavním cílem najít poklady (tzv. kešky). Princip je v tom, že si na webu vyhledáme souřadnice nějaké kešky, pomocí navigace se dostaneme na toto místo a zkusíme najít kešku. Většinou bývají kešky dobře schované a není úplně jednoduché je najít. Po nalezení je možné vyměnit obsah kešky a zapsat se do logbooku - deníku nálezců, který často bývá součástí kešky. Pak kešku znovu schováme pro další hledače. Každý může kešky vytvářet a po vytvoření a schování kešky autor zveřejní její polohu na jednom z mnoha serverů zabývajících se geocachingem, kde si ostatní dohledají souřadnice. I v tomto případě existují mobilní aplikace pro všechny moderní platformy a díky nim lze hledat kešky nedaleko zařízení, na kterém aplikaci spustíme. Podrobnosti lze najít na oficiálních webových stránkách - <http://www.geocaching.com>.

## 11.5 Různé sociální sítě

Nelze opomenout zjišťování polohy na různých sociálních sítích, jako je Facebook nebo Twitter. Na těchto sítích je možné u svých příspěvků zveřejnit svojí polohu.

## 12 Nebezpečí zveřejňování své polohy na internetu

Geolokace na internetu v mnoha případech dokáže usnadnit život, a to jak soukromý tak i profesionální. Ale geolokace dokáže také uškodit, díky tomu, že lidé zveřejňují svojí polohu na internetu, zveřejňují i svoje soukromí s přesností na metry. Mnoho uživatelů nad tímto vůbec nepřemýšlí a bezmyslenkovitě svojí polohu zveřejňuje na internetu. Toto chování může způsobit celou řadu problémů a vystavit uživatele nebezpečí.

Lidé svojí polohu nezveřejňují pouze na sociálních sítích, ale je potřeba si uvědomit, že GPS informace se dnes ukládají i k fotografiím. Tato funkce může být užitečná pro pozdější zorientování se v pořízených fotografiích, ale také nevědomky prozradí naši polohu. Dnešní chytré telefony obsahují kvalitní fotoaparáty a ve většině případů tyto telefony automaticky ukládají GPS informace k pořízeným fotografiím. Poté stačí pouze takovouto fotografii zveřejnit na veřejnou fotogalerii nebo sociální síť a každý může mít přístup k přesné poloze, kde byla fotografie pořízena a kde se právě nachází uživatel, který ji pořídil. Je znám případ amerického vojáka, který vyfotil své kolegy na hlídce v Afghánistánu a okamžitě tyto fotky zveřejnil na webové galerii Google Picasa. U těchto fotografií, které byly veřejně dostupné byly uloženy i GPS informace o pozici.

Stačilo jen, aby tyto fotografie našel nepřítel na internetu a okamžitě věděl přesnou pozici této hlídky, a tím se dostal do značné výhody. Zde se jednalo o extrémní nebezpečí, protože voják, který tyto fotografie zveřejnil, tím vystavil sebe a svoje spolubojovníky do ohrožení života.

Dalším způsobem je zveřejňování polohy na sociálních sítích. Na nejrozšířenější sociální síti dneška Facebooku je možné zveřejňovat příspěvky jako statusy, fotografie, videa a další. Jenže s těmito příspěvky uživatelé zveřejňují i svojí polohu. Dokonce Facebook mobilní aplikace pro operační

system Android zveřejňuje polohu i při odeslání zprávy na chatu. Proto je třeba si uvědomit, co všechno vlastně sdílíme a kdo všechno se k těmto informacím může dostat.

Tím, že zveřejňujeme svojí polohu se můžeme dostat jednoduše do problémů. Jedna žena z USA zveřejnila polohu restaurace, kde byla právě na večeři se svými přáteli. Za několik minut obdržela zprávu o tom, že dotyční přesně ví, kde se žena právě nachází a že jí jdou vykrást dům. Žena se svými přáteli okamžitě jela zkontrolovat dům. V tomto případě se jednalo pouze o žert, ale vždy to takto dobře dopadnout nemusí. Nebo se díky častému zveřejňování své polohy můžeme stát obětí sledování, kdy si pachatel může zaznamenávat, kudy pravidelně chodíme nebo kam často chodíme a může nás sledovat. V horším případě si na nás pachatel může někde počkat a dokonce nás napadnout nebo znásilnit. Může také dojít ke zneužívání online komunikace k obtěžování, zastrašování nebo šikaně vybraných osob, což spolu s tím toho, že osoba ví, kde se zastrašovaný nachází nebo často pohybuje, může být velmi efektivním faktorem při nahnání strachu.<sup>23</sup> Dalším rizikem může být vydírání, kdy zveřejníme svou polohu na místě, kde bychom neměli být. V neposlední řadě může dojít dokonce ke krádeži identity, a to může být veliký problém. Díky geolokaci může pachatel vysledovat bydliště a dalším sledováním může získat další osobní informace pro krádež identity.

Jak je vidět rizik zveřejňování polohy na internetu existuje celá řada a další budou zcela jistě přibývat. Je proto dobré si svoje soukromí chránit, jak to jen jde a přemýšlet o tom, co všechno na internetu zveřejňujeme. V mnoha případech geolokace může usnadnit život, ale pokud si nedokážeme uvědomit rizika, tak geolokace může uškodit a to dokonce i tak, že se díky ní můžeme dostat do ohrožení života. A to by byla cena nejvyšší za neznalost. Snad se bezpečnost na internetu bude zlepšovat a nad podobnými scénáři nebudeme

---

<sup>23</sup>Chytrý telefon, který ví, kde je aneb Smartphone a jeho úskalí v oblasti soukromí. In: SLOUPENSKÁ, Hana. *Inflow Roč.6, no.5 (2013)* — Inflow.cz [online]. 2013 [cit. 2013-06-19]. Dostupné z: <http://www.inflow.cz/chytry-telefon-ktery-vi-kde-je-aneb-smartphone-jeho-uskali-v-oblasti-soukromi>

muset přemýšlet, ale do té doby je to hlavně na uživateli, co všechno o sobě na internetu zveřejní.

## 13 Praktické příklady

Zároveň s touto bakalářskou prací vznikla i webová stránka, kde jsou dostupné všechny příklady a bude tam dostupná i tato práce. Adresa této webové stránky je:

<http://bp.martinlaudat.cz>

Na této webové stránce jsou k dispozici veškeré praktické příklady, které byly vytvořeny. Zdrojové kódy těchto příkladů jsou nekomprimované a kdykoliv si je může zobrazit.

Celý tento web i praktické příklady jsou vytvořeny pomocí responzivního HTML5/CSS3/JS frameworku Zurb Foundation<sup>24</sup>. Jedná se o framework, který umožňuje vytvářet webové stránky, které se přizpůsobují zařízení, ze kterého na tento web přistupujeme. Takže by tento web měl být dobře ovladatelný, jak na počítačích, tak i na mobilních zařízeních, jako jsou tablety či chytré mobilní telefony. Tuto metodu jsem zvolil protože si myslím, že geolokace ve webovém prohlížeči má smysl hlavně na mobilních zařízeních.

### 13.1 Praktické příklady s mapovými podklady

První část praktické části práce je sadou příkladů, které ukazují základní práci s HTML5 Geolocation API a využití tohoto API s různými mapovými podklady. Jedná se celkem o šest ukázek. První ukázka demonstruje pouze jednorázové zjištění polohy, kde se po spuštění příkladu zobrazí hodnoty naměřené Geolocation API. Druhá ukázka je velice podobná první ukázce jen s tím rozdílem, že se jedná o periodické zjišťování polohy. Zbytek příkladů se zabývá využitím Geolocation API s mapovými podklady Google Maps, Mapy.cz, Bing Maps a OpenLayers. Po spuštění těchto příkladů se zobrazí mapa s grafickým ukazatelem naměřené polohy a naměřené údaje se zobrazují i pod mapou jako textové údaje.

---

<sup>24</sup>Podrobnosti a stažení: <http://foundation.zurb.com>

Všechny tyto příklady slouží jako hotové řešení popisované problematiky v kapitolách 7 a 8.

## 13.2 Hlavní praktický příklad

Jedná se o ukázkou implementace HTML5 Geolocation API do komplexnější webové aplikace. Jako tuto aplikaci jsem zvolil projekt, na kterém spolupracuji s kolegy ve společnosti INIZIO Internet Media s.r.o., kde aktuálně pracuji. Jedná se o aplikaci, která se nazývá Předzahrádka a tato aplikace je dostupná jako odkaz z mého webu - <http://bp.martinlaudat.cz> nebo na samostatné adrese:

<http://www.predzahradka.cz>

Tato aplikace slouží jako průvodce venkovními zahrádkami u restaurací v Českých Budějovicích. Cílem bylo vytvořit aplikaci, která bude mapovat možnosti těchto venkovních zahrádek. Jedná se především o možnosti připojení k internetu a připojení k elektrické energii. Díky této aplikaci si uživatel může snadno vyhledat venkovní zahrádku, kam může jít pracovat s notebookem, protože si předem dohledá informace o dostupnosti připojení k internetu a k elektrické síti. Dále se může podívat na oficiální web restaurace, na všechny dostupné sociální sítě, kde daná restaurace vystupuje a na fotografii restaurace.

Tato aplikace využívá moderní webové technologie, jako jsou HTML5, PHP, MySQL, jQuery a další. Díky HTML5 Geolocation API aplikace dokáže zjistit polohu uživatele, který aplikaci právě používá a díky tomu dokáže zobrazit zahrádky, které jsou v blízkém okolí uživatele. Ten si navíc může výsledky nechat vyfiltrovat pomocí jednoduchých přepínačů. Díky těmto vlastnostem je hledání velice rychlé a uživatel nemusí dlouho pátrat, které venkovní zahrádky jsou nejbližší. Dříve by se podobný problém musel řešit nativní aplikací pro každé zařízení zvlášť, ale s příchodem HTML5 už tomu tak není. Stačí nám pouze zařízení s webovým prohlížečem, které HTML5 podporuje.

## 14 Závěr

Nové moderní webové technologie, mezi které patří i HTML5, nám umožňují vytvářet vyspělé webové aplikace. HTML5 Geolocation API je pouze malá část toho, co dnes tvoří moderní webové aplikace. Přesto díky tomuto API můžeme dělat aplikace, které zpříjemní uživatelský zážitek z webové stránky či aplikace. Práce s HTML5 Geolocation API je velice jednoduchá a zvládne ji každý, kdo se o tvorbu webů alespoň trochu zajímá. Toto API toho samo o sobě moc neumí, dokáže nám zjistit pouze několik údajů o pozici, kde se nacházíme. S těmito údaji můžeme dále pomocí javascriptu pracovat. Ovšem ten pravý rozměr dostává HTML5 Geolocation API při použití s mapovými podklady. Díky tomu, že můžeme použít mapové podklady, jako například Google Maps nebo Mapy.cz, můžeme vytvářet komplexní webové služby, které toto API budou využívat. Na internetu existuje hned několik služeb a aplikací využívajících geolokaci a další budou určitě přibývat. I když se jedná o moderní technologii, tak podpora v prohlížečích je velmi solidní a díky několika metodám zjišťování polohy, nejsme vázáni pouze na GPS hardware.

V rámci této bakalářské práce byly představeny zásadní novinky, které obsahují standard HTML5. Bylo detailně rozebráno HTML5 Geolocation API, bylo popsáno využití tohoto API s různými mapovými podklady a následně vytvořeny praktické ukázky. V práci jsem rozebral nejdůležitější způsoby zjišťování polohy na internetu a jejich výhody a nevýhody. Otestoval jsem podporu tohoto API v prohlížečích a doporučil metody pro simulaci geolokace ve starších prohlížečích. Dále jsem popsal využití tohoto API na internetu a vysvětlil jsem toto využití na nejznámějších aplikacích, které toto API používají. Vypsal jsem hlavní rizika, která může časté a bezhlavé zveřejňování své polohy na internetu způsobit a na závěr jsem vypracoval praktické příklady, které jsem umístil na veřejně dostupnou webovou stránku.

Všechny cíle, které byly na začátku práce stanoveny, byly splněny.



## 15 Literatura a zdroje

[1] *HTML5: up and running*. 1st ed. Sebastopol, CA: O'Reilly, 2010. ISBN 05-968-0602-7.

[2] *HTML5.cz - vše co potřebujete vědět o HTML5* [online]. Dostupné z: <http://www.html5.cz/>

[3] GOLDSTEIN, Alexis, Louis LAZARIS a Estelle WEYL. *HTML5 a CSS3 pro webové designéry*. Vyd. 1. Brno: Zoner Press, 2011, 286 s. Encyklopedie webdesignera. ISBN 978-80-7413-166-0 (BROŽ.).

[4] *Geolocation API Specification* [online]. 2011. Dostupné z: <http://dev.w3.org/geo/api/spec-source.html>

[5] LUBBERS, Peter, Brian ALBERS a Frank SALIM. *HTML5: programujeme moderní webové aplikace*. Vyd. 1. Brno: Computer Press, 2011, 304 s. ISBN 978-80-251-3539-6 (BROŽ.).

[6] HOLDENER, Anthony T. *HTML5 Geolocation*. Sebastopol, CA: O'Reilly, 2011, 95 s. ISBN 14-493-0472-9.

[7] *Geolokace v prohlížeči*. MALÝ, Martin. Zdrojak.cz [online]. 2010. Dostupné z: <http://zdrojak.root.cz/clanky/geolokace-v-prohlizeci/>

[8] *HTML5 - nové vlastnosti*: Article. In: ŠŤASTNÝ, Jiří. Programujte.com - web o programování, webdesignu, počítačové grafice, databázích, elektrotechnice a designu [online]. 2011 [cit. 2013-02-21]. Dostupné z: <http://programujte.com/clanek/2010082200-html5-nove-vlastnosti/>

- [9] *HTML5 - geolokační rozhraní*. Programujte.com - web o programování, webdesignu, počítačové grafice, databázích, elektrotechnice a designu [online]. 2012 [cit. 2013-03-14]. Dostupné z: <http://programujte.com/clanek/2011052400-html5-geolokacni-rozhrani/>
- [10] *O technologii GPS*. In: Technologie Mio [online]. 2012 [cit. 2013-06-05]. Dostupné z: [http://eu.mio.com/cs.cz/global-positioning-system\\_jak-gps-funguje.htm](http://eu.mio.com/cs.cz/global-positioning-system_jak-gps-funguje.htm)
- [11] *Chytrý telefon, který ví, kde je aneb Smartphone a jeho úskalí v oblasti soukromí*. In: SLOUPENSKÁ, Hana. Inflow Roč.6, no.5 (2013) — Inflow.cz [online]. 2013 [cit. 2013-06-19]. Dostupné z: <http://www.inflow.cz/chytry-telefon-ktery-vi-kde-je-anebsmartphone-jeho-uskali-v-oblasti-soukromi>