

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



Bakalářská práce

Webové standardy – HTML animace

Jan Kolář

© 2017 ČZU v Praze

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Jan Kolář

Informatika

Název práce

Webové standardy – HTML animace

Název anglicky

Web standards – HTML animations

Cíle práce

Hlavním cílem bakalářské práce je porovnat podporu Flash animací a HTML animací napříč všemi webovými prohlížeči, srovnat datovou náročnost a další skutečnosti, kterými se tyto dva způsoby tvorby animací odlišují.

Metodika

Metodika řešení problematiky bakalářské práce je založena na studiu a analýze odborných informačních zdrojů. Praktická část je zaměřena na využívání nástrojů pro tvorbu a prezentaci animací a poté na analýze a vzájemném porovnávání výstupů. Na základě syntézy teoretických poznatků a výsledků v praktické části jsou formulovány závěry práce.

Doporučený rozsah práce

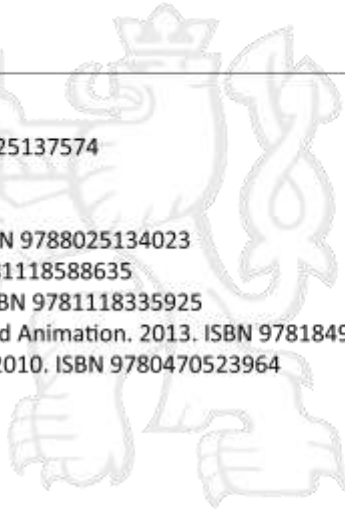
30-40 stran

Klíčová slova

HTML, CSS, animace, Flash, Edge, PPC

Doporučené zdroje informací

DOMES, M., Google Adwords Jednoduše. 2012. ISBN 9788025137574
DOMES, M. Sklik Jednoduše. 2012. ISBN 9788025137604
HOGAN, B. HTML5 a CSS3. 2012. ISBN 9788025135761
JANOUC, V. 333 tipů a triků pro internetový marketing. ISBN 9788025134023
KARLINS, D., HTML5 and CSS3 For Dummies. 2013. ISBN 9781118588635
ROHDE, M., Adobe Edge Animate CC For Dummies. 2013. ISBN 9781118335925
SHENOY, A., HTML5 and CSS3 Transition, Transformation, and Animation. 2013. ISBN 9781849519946
SCHAFFER, S M., HTML, XHTML, and CSS Bible (5th Edition). 2010. ISBN 9780470523964



Předběžný termín obhajoby

2016/17 LS – PEF

Vedoucí práce

Ing. Petr Benda, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 21. 10. 2016

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 24. 10. 2016

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 23. 11. 2016

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci „Webové standardy – HTML animace“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne _____

Poděkování

Rád bych touto cestou poděkoval panu Ing. Petru Bendovi, Ph.D., za konzultace, připomínky a odborné vedení mé bakalářské práce.

Webové standardy – HTML animace

Souhrn

Bakalářská práce se zabývá porovnáním Flash a HTML animací. Teoretická část obsahuje převážně úvod do HTML a CSS, kde jsou vysvětleny základní principy tvorby animací a syntaxe obou jazyků. Dále je stručně představena technologie Flash a velké reklamní systémy, které podporují zobrazení animovaných reklamních formátů.

Hlavním cílem praktické části bakalářské práce je porovnání výstupů vytvořených HTML a Flash animací, a to na základě datové náročnosti a zobrazení ve všech běžně dostupných prohlížečích na desktopových a mobilních zařízeních. Dílčím cílem je úspěšně validovat animaci jako banner v reklamním systému AdWords.

Klíčová slova: animace, HTML, CSS, Flash, Adobe Edge, Google Web Designer, reklamní systémy

Web standards - HTML animations

Summary

Bachelor thesis deals with comparison of Flash and HTML animations. The theoretical part contains introduction to HTML and CSS where basics of creating animations and syntax of both languages are explained. Another chapter of this part introduces Flash technology and big ad systems supporting displaying dynamic ad banners.

Main goal of the practical part of the bachelor thesis is comparison of created HTML and Flash animations based on data size and displaying in web browsers on desktop and mobile devices. Partial goal is successful validation of banner in AdWords ad system.

Keywords: animations, HTML, CSS, Flash, Adobe edge, Google Web Designer, ad systems

Obsah

1 Úvod.....	8
2 Cíl práce a metodika.....	9
2.1 Cíle práce	9
2.2 Metodika	9
3 Přehled řešené problematiky	10
3.1 HTML animace	10
3.1.1 HTML	10
3.1.2 CSS	12
3.1.3 WYSIWYG editory	20
3.2 Flash animace.....	21
3.2.1 Adobe Flash Player	21
3.2.2 ActionScript.....	21
3.2.3 Flash na mobilních zařízeních	21
3.3 Reklamní systémy podporující animované bannery	22
3.3.1 AdWords.....	23
3.3.2 AdForm.....	23
4 Vlastní práce.....	24
4.1 Charakteristika	24
4.2 Technické specifikace HTML banneru pro Google AdWords	24
4.3 Storyboard.....	26
4.4 Uložení a komprese obrázků.....	27
4.5 Tvorba HTML šablony	28
4.6 Příprava keyframů v CSS.....	29
4.7 Záložní podoba banneru pro prohlížeč, který nepodporuje keyframy	29
5 Zhodnocení.....	31
5.1 Test zobrazení animací v prohlížečích.....	31
5.1.1 Desktopová zařízení.....	31
5.1.2 Mobilní zařízení.....	32
5.2 Porovnání datové velikosti HTML a Flash výstupu	33
5.3 Validace HTML banneru pro reklamní systém AdWords	34
6 Závěr	35
7 Seznam použitých zdrojů	37
8 Seznam obrázků	41
9 Seznam tabulek	41

10 Slovník pojmů a zkratk.....	42
11 Přílohy.....	42

1 Úvod

Internet je dnes nepochybně místem, které dokáže oslovit celý svět. K webovým prezentacím také nedílně patří nabízení reklamních prostor pro inzerenty. Časy, kdy na internetu problikávaly jen GIF animace, již dávno minuly. S mnohem komplexnějšími funkcemi pro animování přišla technologie Flash. Zkušenosti uživatelé, a především webovými vývojáři, si mohli všimnout, že Flash animace se na webových stránkách v dnešní době zobrazují méně, než tomu bylo kdysi. Autor se v bakalářské práci snaží najít příčinu, která způsobila přechod standardu v tvorbě animací od Flash k HTML.

Webové animace často nacházejí využití v reklamních systémech namísto statických obrázků. Výhodou dynamických reklamních bannerů je větší potenciál zaujmout případného návštěvníka webových stránek samotným pohybem a schopnost pojmout více obsahu. Největší překážku pro tvorbu animací představuje jejich výrobní proces, který je oproti statickému obrázku složitější. Každý reklamní systém má navíc svou technickou specifikaci pro animace, která musí být dodržena.

Bakalářská práce je rozdělena na dvě části. V první části bakalářské práce se autor zabývá shrnutím nastudovaných poznatků o animacích a o reklamních systémech nabízejících inzerci animací. V druhé části pak autor tvoří HTML reklamní banner pro fiktivní cestovní kancelář, který by mohl být použit v reklamním systému Google AdWords. Zobrazení HTML animace je testováno ve všech běžných webových prohlížečích a porovnává se s totožnou Flash animací.

2 Cíl práce a metodika

2.1 Cíle práce

Hlavním cílem bakalářské práce je porovnat podporu Flash a HTML animací napříč všemi webovými prohlížeči, srovnat datovou náročnost a další skutečnosti, kterými se tyto dva způsoby tvorby animací odlišují. Dílčím cílem práce je příprava animace ve formě reklamního banneru dle technické specifikace pro vybraný reklamní systém a následně jeho úspěšná validace.

2.2 Metodika

Metodika teoretické části bakalářské práce vychází ze studia odborné literatury popisující HTML animace, technologii Flash a reklamní systémy působící na českém trhu, které zobrazují dynamické reklamy.

Metodika praktické části spočívá ve vytvoření totožné HTML a Flash animace a jejich vzájemném porovnání. Jako kritéria jsou zvoleny parametry datové velikosti a korektní zobrazení animace ve webovém prohlížeči na desktopovém a mobilním zařízení. Tvorbě animací předchází nastudování technické specifikace reklamního systému AdWords, jejíž splnění je otestované v oficiálním validátoru společnosti Google.

3 Přehled řešené problematiky

3.1 HTML animace

K tvorbě v HTML a CSS se nevyžadují žádné speciální nástroje. Může se používat jakýkoliv textový editor, a to dokonce program Poznámkový blok, jenž je součástí operačního systému Windows, nebo program TextWrangler, který lze stáhnout zdarma pro operační systém OS X (Castro a Hyslop, 2012).

Soubory musí mít příponu .html (případně .css), aby webové prohlížeče rozpoznaly, že se jedná o webovou prezentaci a věděly, jak mají interpretovat jejich kód (Castro a Hyslop, 2012).

3.1.1 HTML

Jazyk HTML (Hypertext Markup Language) je určen pro budování webových stránek. Aktuální používaná verze jazyka, kterou v roce 2014 představila organizace W3C, je HTML5. Má podporu všech předních prohlížečů, za které se považují Chrome, Internet Explorer, Firefox, Safari a Opera (Coremans, 2015).

HTML5 obsahuje nové elementy (např. audio a video) definující vkládání obsahu, což znamená, že se nemusí používat zdrojové kódy navíc nebo zásuvné moduly a další doplňky (Hogan, 2011).

Mezi předchozí verze, které už nyní k tvorbě webových stránek doporučovány nejsou, se řadí například velmi rozšířené XHTML (eXtensible Hypertext Markup Language) a HTML 4.0 (Coremans, 2015).

Konstrukce HTML kódu

Webová stránka je tvořena HTML elementy. Elementy většinou představují párové tagy, nicméně existuje i několik nepárových. Párové tagy se otevírají zápisem v ostrých závorkách a uzavírají se stejným zápisem, ale s přidaným lomítkem za první závorkou. Mezi otevření a zavření tagu je možné vložit obsah (Coremans, 2015).

Příklad párových tagů:

```
<head></head> // hlavička webu  
<div></div> // blok  
<p></p> // odstavec
```

Příklad vyplněného párového tagu:

```
<p>Toto je text v odstavci.</p>
```

Příklad nepárových tagů:

```
<img> // obrázek  
<br> // nový řádek  
<input> // element pro formulář
```

Do nepárových tagů se obsah vložit nedá, výjimkou je pouze přidání obsahu přes atribut.

Příklad výjimky, kdy se obsah doplňuje pomocí atributu:

```
<input type="submit" value="Odeslat">
```

Základní kostra HTML5 dokumentu:

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>Titulek webové stránky</title>  
    Do hlavičky zapisujeme obsah, který obvykle na webové  
    stránce není vidět (např. načítání externích souborů).  
  </head>  
  <body>  
    Do těla vkládáme obsah webové stránky, který je vidět.  
  </body>  
</html>
```

Zdroj: Coremans, 2015

Atributy

Každý HTML element může obsahovat atributy, které ho dokážou ovlivnit. Pro tvorbu animací mají význam především atributy **id** nebo **class**. Id je zkratka pro identifikátor, jehož hodnota může být na celé webové stránce použita pouze jednou. Oproti tomu stejnou hodnotu atributu class lze přiřadit jakémukoliv množství elementů. Atributy mají význam především pro propojení HTML elementů s kaskádovými styly, přes které jsou HTML elementům udávány vlastnosti stylů. Atributy mohou být přiřazeny párovým i nepárovým tagům (Coremans, 2015).

Castro a Hyslop (2012) uvádí: „Některé atributy, jako jsou atributy **href** a **src**, se odkazují na jiné soubory, a proto přijímají hodnoty ve formě adresy URL (Uniform Resource Locator), což je jedinečná adresa prostředku na Internetu. Spousta atributů ale přijímá číselnou hodnotu, a to zejména atributy popisující velikost a délku. Číselné hodnoty se

uvádí vždy bez jednotky a udávají počet pixelů (například šířka a výška obrázku nebo videa). Atributy obvykle přijímají hodnoty výčtového typu a předdefinované hodnoty. Jinými slovy – je potřeba si zvolit hodnotu z nějakého standardního seznamu možností.“

Příklad použití atributů:

```
<div id="stranka">
  <p class="odstavec"></p>
  <p id="unikatni-identifikator" class="odstavec"></p>
</div>
```

(Zdroj: vlastní zpracování)

Zanořování elementů

Nachází-li se element uvnitř jiného elementu, je považován za dceřiný element, obalen rodičovským elementem (Castro a Hyslop, 2012).

Když elementy obsahují jiné další elementy, je potřeba dbát na správné zanořování. Všechny elementy musí být vždy korektně zanořené. Křížení párových tagů není povolené a je považováno za závažnou chybu (Castro a Hyslop, 2012).

Příklad správného zanoření:

```
<div id="stranka">
  <p class="odstavec">Toto je zanořený element.</p>
</div>
```

Příklad chybného zanoření, kdy se elementy vzájemně kříží:

```
<div id="stranka">
  <p class="odstavec">Toto je špatně zanořený element.
</div></p>
```

3.1.2 CSS

CSS vzniklo kolem roku 1997. Jedná se o kolekci metod pro grafickou úpravu webových stránek. Zkratka CSS znamená Cascading Style Sheets, česky „kaskádové styly“. Výraz kaskádové vyjadřuje možnost vrstvení definic stylů (Janovský – CSS styly).

Hlavní účel CSS představuje poskytnutí čitelné metody oddělení stylů od obsahu prezentace. Oddělení umožňuje vývojářům použít jeden definovaný styl pro více stránek.

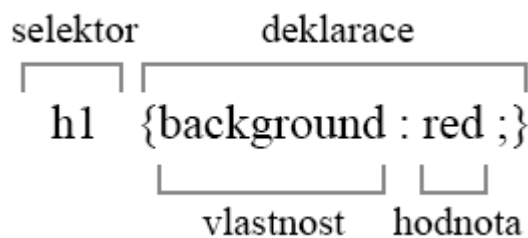
Pokud se vývojář rozhodne, že chce v celé webové prezentaci změnit například barvu, stačí ji změnit pouze na jednom místě a změny se projeví na všech stránkách, ke kterým je stylpis připojen (Mueller, 2014).

CSS dokáže propojovat různě definovaná pravidla s HTML elementy, které se objevují v prezentaci. Tato pravidla určují, jak se budou HTML elementy chovat a vypadat. CSS zápis pravidla se dělí na selektor a deklaraci, která se dále člení na vlastnost a hodnotu (Larsen, 2013).

Hohan (2011) k CSS uvádí: „*Kaskádové styly poskytují pokročilé selektory, grafická vylepšení a lepší podporu fontů, díky čemuž jsou weby vizuálně atraktivnější, aniž by bylo potřeba použít techniku nahrazování obrázků, složitý JavaScript nebo grafické nástroje.*“

Selektory a deklarace

Selektor slouží k označení konkrétních HTML elementů (Schafer, 2011).



Obrázek 1 - Zápis CSS pravidla

Zdroj: vlastní zpracování

- Selektor určuje, jaký element nebo elementy má pravidlo ovlivňovat.
- Deklarace zaobaluje vlastnost a její hodnotu, která je aplikována na selektor. Deklarací může být více.
 - Vlastnost značí vlastnost vybraného elementu, kterou chceme ovlivnit. Může to být například *background* (pozadí).
 - Hodnota je specifikace pro vlastnost. Ve spojení s vlastností *background* může být hodnota vyplněna například jako *red* (červená barva).

(Larsen, 2013)

Příklady použití základních selektorů

```
// Obarvení všech odstavců na červeně a nastavení tloušťky písma
p {color: red;font-weight: bold;}

// Obarvení všech odstavců a nadpisů první úrovně
p, h1 {color: red;}

// Označení všech elementů
* {color: red;}

// Označení párového tagu span uvnitř buňky v tabulce
table td span {color: red;}

// Označení elementu dle třídy
.nazevTridy {color: red;}

// Označení elementu dle identifikátoru
#nazevIdentifikatoru {color: red;}

// Označení tabulky dle atributu border s vyplněnou hodnotou 3
table[border="3"] {color: red;}

// Označení tabulky s atributem data-red (hodnota není povinná)
table[data-red] {color: red;}

// Označení elementu dle třídy a atributu border
table.nazevTridy[border] {color: red;}
```

(Schafer, 2011)

CSS zápisy

Při stylování HTML elementů se mohou používat 3 různé způsoby zápisu kaskádových stylů:

- a) in-line zápis
- b) interní zápis
- c) externí zápis

In-line zápis CSS

In-line (v překladu „v řádku“) zápisem se styluje přímo konkrétní HTML element. Na ostatní prvky daný zápis nemá vliv, respektive ostatním prvkům nenastavuje žádnou CSS vlastnost. Je zapotřebí zvážit, zdali je použití in-line CSS zápisu v dané situaci přínosné či

nikoliv. V případě, že je potřeba aplikovat stejnou CSS deklaraci na více HTML elementů, pak je použití in-line stylu z hlediska zápisu a přehlednosti kódu nevýhodné.

Pro použití in-line zápisu se používá atribut style, do kterého vkládáme CSS pravidla. Selektor zde neurčujeme, jelikož se deklarace vždy vztahuje k jednomu danému HTML elementu.

Příklad:

```
<p style="color: red;">Tento text má červenou barvu.</p>
```

(Janovský – Zápisy CSS)

Interní zápis CSS

V interním zápisu se využívá celý zápis CSS pravidla, tedy i včetně selektoru. Všechna CSS pravidla se vkládají do párového HTML tagu <style></style>, který se může v HTML souboru vyskytovat kdekoliv, ale jeho doporučené umístění je v hlavičce HTML dokumentu, tj. mezi tagy <head></head>.

Oproti in-line zápisu je interní zápis přehlednější a CSS vlastnosti HTML elementů se mohou spravovat centrálně z jednoho místa, ale pouze v rámci jednoho souboru.

Příklad:

```
<style>
  p {color: red;}
</style>
<p>Tento text má červenou barvu.</p>
<p>Tento také!</p>
```

(Janovský – Zápisy CSS)

Externí zápis CSS

Celý stylpis se ukládá jako externí soubor a do HTML dokumentu se umísťuje pouze odkaz, který stylpis automaticky načítá. Externí zápis je velmi výhodný, protože všechna CSS pravidla pro více HTML souborů mohou být uložena na jednom místě.

Pokud se ve všech HTML souborech rozhodneme, že odstavec bude mít červenou barvu, stačí dané CSS pravidlo zapsat pouze jednou do externího CSS souboru a změny se projeví na všech místech.

```
//Obsah CSS souboru style.css
p {color: red;}

//HTML soubor
<html>
<head>
<link rel="stylesheet" href="style.css" type="text/css">
</head>
<body>
    <p>Tento text má červenou barvu.</p>
</body>
</html>
```

(Janovský – Zápisy CSS)

Keyframy

Pro vyjádření přechodů v animaci z jednoho stavu do druhého se využívají keyframy. V každém keyframu se nastavuje, v jakém procentuálním bodě se mají různé vlastnosti měnit. Keyframe je svázán k selektoru, který nabývá dalších vlastností upřesňujících průběh animace:

- `animation-name`: povinná vlastnost, která určuje název keyframu, s nímž bude HTML element pracovat;
- `animation-duration`: povinná vlastnost, která určuje, jak dlouho bude animace probíhat;
- `animation-iteration-count`: nepovinná vlastnost, která určuje počet opakování animace. Pokud vlastnost není vyplněna, přehraje se animace pouze jednou;
- `animation-play-state`: nepovinná vlastnost, s jejíž pomocí se může animace pozastavovat a spouštět;
- `animation-timing-function`: nepovinná vlastnost, která určuje křivku rychlosti průběhu animace;
- `animation-delay`: nepovinná vlastnost, která dokáže animaci přehrát se zpožděním.

(Shenoy a Guarini, 2013; Karlins, 2014)

Michálek (2016) uvádí: „Keyframy definují začátek (klíčové slovo *from* nebo 0%), průběh (pomocí procent z průběhu) a konec (to nebo 100%) animace. Přechod mezi jednotlivými keyframy vypočítá prohlížeč sám. Začátek a konec je potřeba nastavit vždy, počet keyframů mezi nimi není nijak limitovaný. Syntaxe keyframu ve stylpisu vypadá takto:“

```
@keyframes _nazev_animace_ {  
  _cas_ { _deklarace_ }  
}
```

(Zdroj: Michálek, 2016)

Animation-timing-function

Vlastnost `animation-timing-function` určuje křivku rychlosti animace. Křivku rychlosti můžeme nastavit dle předdefinovaných hodnot (W3Schools: CSS3 animation-timing-function Property).

`Animation-timing-function` je vlastnost, která používá matematickou funkci zvanou křivka Cubic Biezer. Mimo předdefinované hodnoty je tedy možnost užívat svou vlastní křivku (W3schools.com: CSS3 animation-timing-function Property).

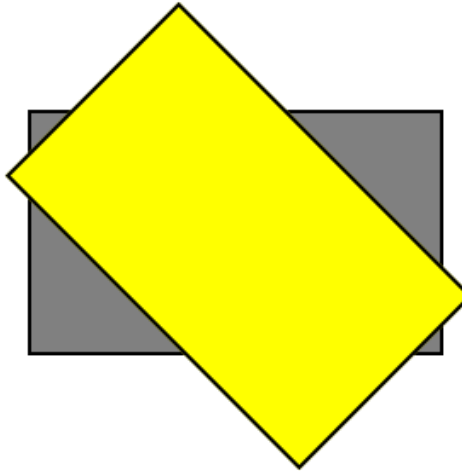
Syntaxe

```
animation-timing-function: linear|ease|ease-in|ease-out|ease-in-  
out|step-start|step-end|steps(int,start|end)|cubic-  
bezier(n,n,n,n)|initial|inherit;
```

CSS transformace

V CSS3 se může měnit pohled na HTML element v prostoru o dvou i třech dimenzích. V transformacích ve dvoudimenzionálním prostoru se může HTML element otáčet (`rotate`), překroutit (`skew`), posunout (`translate`) a zmenšit či zvětšit (`scale`).

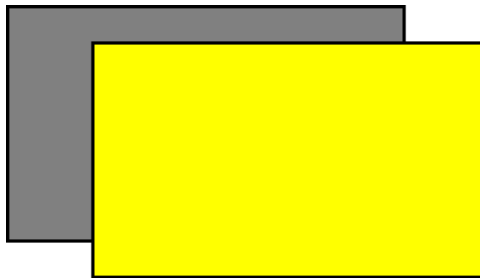
Rotate určuje otočení HTML elementu. Při použití kladné hodnoty se HTML element otáčí po směru hodinových ručiček.



Obrázek 2 - Ukázka transformace - rotate

Zdroj: vlastní zpracování

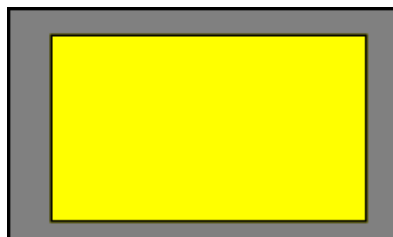
Translate mění pozici HTML elementu na ose X a Y.



Obrázek 3 - Ukázka transformace - translate

Zdroj: vlastní zpracování

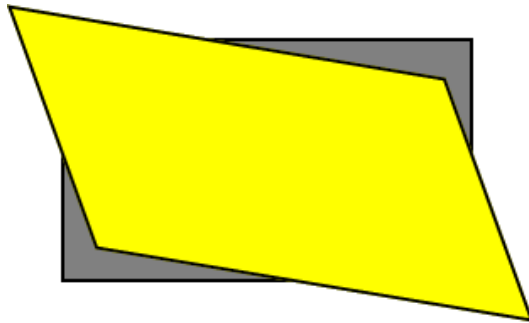
Scale zmenšuje či zvětšuje HTML element, aniž by se měnila jeho pozice.



Obrázek 4 - Ukázka transformace - scale

Zdroj: vlastní zpracování

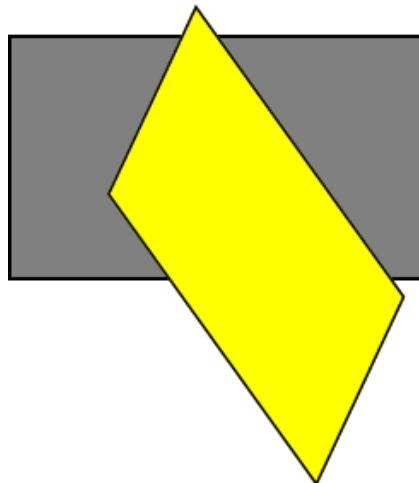
Skew mění úhel HTML elementu podél os X a Y.



Obrázek 5 - Ukázka transformace - skew

Zdroj: vlastní zpracování

Současné použití více transformací pro jeden element se navzájem nevyklučuje.



Obrázek 6 - Ukázka více transformací

Zdroj: vlastní zpracování

V transformacích v prostoru o třech dimenzích se pracuje stejně, ale nabízí se možnost využití dalšího prostoru Z. Pro zobrazení HTML elementu je nezbytné nastavit perspektivu. Perspektiva definuje, kolik pixelů z 3D HTML elementu má být umístěno pro zobrazení. Tato hodnota dovoluje měnit pohled na 3D HTML element (Shenoy a Guarini, 2013; Karlins, 2014).

3.1.3 WYSIWYG editory

WYSIWYG editor představuje takový editor, který vývojářům umožňuje vidět aktuální výstup práce už během procesu tvoření. Zkratka WYSIWYG je složena z anglického „What You See Is What You Get“, což v překladu znamená: „Co vidíš, to dostaneš“ (Rouse, 2011).

Google Web Designer

„Google Web Designer je pokročilá webová freeware aplikace vytvořená pomocí technologie HTML5, která umožňuje návrh a vytváření reklam HTML5 a jiného webového obsahu pomocí integrovaného vizuálního a kódového rozhraní. V zobrazení návrhu se v aplikaci Google Web Designer může vytvářet obsah pomocí nástrojů na kreslení, textu a 3D objektů a objekty je možné animovat na časové ose. Jakmile je vytváření obsahu dokončené, Google Web Designer vygeneruje čistý a pro lidi čitelný kód HTML5, CSS3 a Javascript“ (Nápověda Google Web Designer: Co je to Google Web Designer).

Během vytváření animace se může zobrazit její náhled v prohlížeči a zkontrolovat, jak bude vypadat, až bude publikována (Nápověda Google Web Designer: Zobrazení náhledu reklamy nebo stránky HTML).

Aplikace Google Web Designer je zatím k dostání pouze ve vývojové beta verzi. Pro diskuzi ohledně vývoje programu existuje fórum, kde se vývojáři aktivně podílejí na dotazech a návrzích od uživatelů (Oficiální stránky Google Web Designer).

Adobe Edge Animate CC

Boudreaux (2014) uvádí: *„Nástroj Edge Animate umožňuje webovým designérům snadno vytvářet interaktivní animace HTML pro web, digitální publikování, multimediální reklamu a mnohem více pro prostředí počítačových i mobilních prohlížečů.“*

Adobe Edge Animate CC je součástí placeného balíčku Adobe Creative Cloud (Rohde, 2013).

Aktivní vývoj aplikace Adobe Edge Animate CC byl v listopadu roku 2015 ukončen, nicméně je stále k dostání v balíčku Adobe Creative Cloud (Adobe.com – Edge).

3.2 Flash animace

V roce 1997 zakoupila společnost Macromedia grafický program zvaný FutureSplash od společnosti FutureWave. FutureSplash byl malý program se schopnostmi generovat kompaktní vektorovou grafiku a animace určené pro zobrazení na webových stránkách. V roce 2005 koupilo Adobe společnost Macromedia a během tří let úspěšně integrovalo Flash (přejmenovaný **FutureSplash**) do své softwarové rodiny (Perkins, 2010).

3.2.1 Adobe Flash Player

Pro přehrávání Flash animací (a také dalšího Flash obsahu) je potřeba mít ve svém zařízení nainstalovaný Adobe Flash Player.

„Aplikace Adobe Flash Player je zásuvný modul pro prohlížeč, který je v současné době nainstalovaný ve více než 1,3 miliardě zařízení.“ (Adobe Flash Player – Other Versions).

3.2.2 ActionScript

„ActionScript je programovací jazyk pro prostředí přehrávače Adobe Flash Player. Umožňuje interakce, zpracovávání dat a další možnosti v obsahu a v aplikacích Flash, Flex a AIR“ (Adobe TEAM, 2008).

Účelem jazyka ActionScript je umožnění vytváření balíků aplikací s velkým množstvím dat a kódem, který je objektově orientovaný – má schopnost opakovaného použití (Adobe TEAM, 2008).

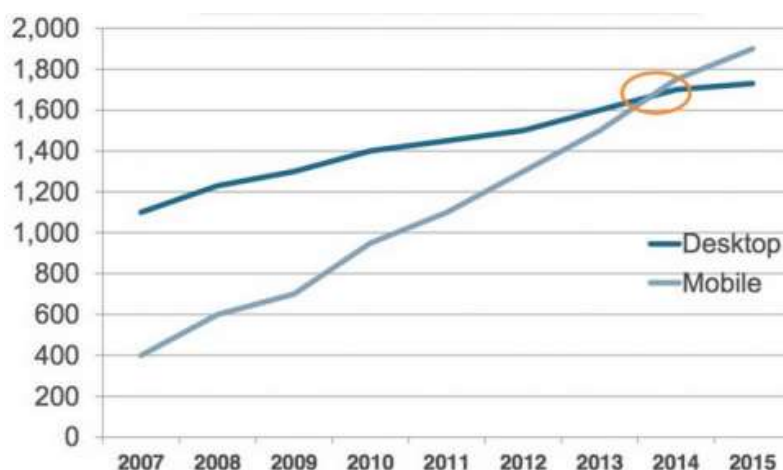
3.2.3 Flash na mobilních zařízeních

Adobe v listopadu roku 2011 vydalo prohlášení o ukončení vývoje Flash Playeru pro mobilní zařízení. Jediné, čím se bude Adobe v této oblasti zabývat, je oprava chyb v již vydaných verzích. Společnost oficiálně vyjádřila podporu HTML5 a přislíbila, že se bude snažit podílet na vývoji s klíčovými hráči na trhu, mezi které patří Google, Apple a Microsoft (Winokur, 2011).

Adobe pracuje s technologií Flash nadále, nicméně zaměřuje se už pouze na to umožnit flashovým vývojářům publikovat aplikace přes Adobe Air v největších obchodech s aplikacemi (Google Play, iStore). Zcela se tedy stahuje z podpory Flash technologie ve webových prohlížečích (Winokur, 2011).

„Odborníci se domnívají, že Adobe přestal podporovat mobilní Flash v prohlížeči proto, že se mu nepodařilo prosadit v zařízeních Apple, která patří ve svých kategoriích k nejprodávanejším“ (Vokáč, 2011).

Počet mobilních přístrojů oproti desktopovým neustále roste. V roce 2014 nastal bod zvratu, kdy bylo poprvé zaznamenáno více mobilů. Počet desktopových zařízení také roste, nicméně pomaleji než mobilních (Dave Chaffey, 2016).



Obrázek 7 - Celosvětový počet uživatelů (miliony)

Zdroj: Dave Chaffey, 2016

3.3 Reklamní systémy podporující animované bannery

Mezi velké reklamní systémy, které podporují animované bannery, můžeme zařadit:

- a) AdWords (reklamní systém společnosti Google)
- b) AdForm

Existuje několik forem plateb z reklamy, které motivují majitele webových stránek nabízet reklamní prostor k pronájmu. Nejčastější je PPC – Pay Per Click (platba za klik), PPV – Pay Per View (tedy platba za zhlédnutí), CPI – Cost Per Impression (cena za impresi), nebo PPA – Pay Per Action (platba za akci) (Domes - Sklik jednoduše, 2012).

3.3.1 AdWords

AdWords je reklamní systém provozovaný společností Google. Reklamy dokáže zobrazit ve svém vyhledávači a také v síti dalších webových stránek, např. na YouTube. (Nápověda AdWords).

Google se v únoru roku 2016 rozhodl, že od července ukončí podporu animací vytvořených ve Flashi a doporučuje tvořit animace pouze v HTML a CSS. Jako hlavní důvod uvádí dosavadní omezené zobrazování reklamních Flash bannerů. Flashové animace se totiž například standardně nezobrazují na mobilních zařízeních, jejichž podíl na trhu neustále roste (Google display ads go to 100% HTML5).

3.3.2 AdForm

AdForm byl založen v roce 2002 v Dánsku. Do České republiky se dostal v roce 2013 (AdForm – adform.com).

Společnost AdForm se specializuje na RTB (Real Time Bidding), což je technologie umožňující nákup impresí reklamy v aukci v reálném čase. Aukce proběhne při každém zobrazení imprese (RTB - Real Time Bidding).

„RTB se nejčastěji používá jako výplň nevyprodaných ploch. Pokud reklamní systém při zobrazení stránky nemá přímo prodanou reklamu, tak uspořádá aukci o její vyplnění. Vybrána je reklama, za jejíž zobrazení je nabídnuta nejvyšší cena. Celá tato aukce proběhne v řádech desítek milisekund. Imprese se prodá až po zobrazení stránky, ale tak rychle, že si toho uživatel nevšimne, nedochází k viditelnému prodlení“ (RTB - Real Time Bidding).

4 Vlastní práce

4.1 Charakteristika

Předmětem praktické části bakalářské práce je tvorba HTML animace pro reklamní systém AdWords a její následné porovnání s totožnou Flash animací na základě datové velikosti a korektního zobrazení ve webových prohlížečích.

Autor bakalářské práce se rozhodl, že bude vytvářet HTML animaci pro fiktivní cestovní kancelář, kde si stanovil následující cíle: 1) zobrazit na dostatečně dlouhou dobu fotografii Španělska a Rakouska, 2) na závěr zobrazit kontakt a logo společnosti, které zůstane viditelné po doběhnutí celé animace.

Všechny obrázky použité v animaci jsou dílem autora práce.

4.2 Technické specifikace HTML banneru pro Google AdWords

Google AdWords doporučuje pro tvorbu HTML bannerů používat WYSIWYG nástroj Google Web Designer. Nabízí ale i možnost manuálního vytvoření za předpokladu dodržení technických specifikací. Autor bakalářské práce se rozhodl zvolit manuální přípravu animace.

Nejprve je potřeba vybrat si jeden z požadovaných rozměrů banneru, který Google AdWords nabízí. Autor práce zvolil standardní obdélníkový rozměr 300 × 250.

Název	Rozměr (px)	Název	Rozměr (px)
Banner	468 × 60	Půlstránková reklama	300 × 600
Široký banner	728 × 90	Velký široký banner	970 × 90
Mobilní široký banner	320 × 50	Reklamní panel	970 × 250
Čtverec	250 × 250	Na výšku	300 × 1050
Malý čtverec	200 × 200	Panorama	980 × 120
Velký obdélník	336 × 280	Horní banner	930 × 180
Vkládaný obdélník	300 × 250	Trojnásobná širokoúhlá	250 × 360

Tabulka 1 - Formáty reklamy Google AdWords

Zdroj: Nápověda AdWords

Velmi důležité je dodržení maximální datové velikosti celého banneru **150 kB**, do které se započítávají všechny části HTML a CSS kódu včetně obrázků a případného JavaScriptu.

Google AdWords povoluje maximální délku animace 30 s. Autor bakalářské práce se rozhodl pro rozdělení HTML animace na dvě opakování, přičemž jedno celé opakování bude trvat 15 sekund. Na konci druhého opakování se animace zastaví a zobrazí se kontaktní údaje fiktivní cestovní kanceláře.

HTML dokument musí obsahovat HTML deklaraci. Do hlavičky HTML dokumentu je nutné vložit meta tag určující šířku a výšku HTML banneru.

```
<!DOCTYPE html>
<html>
<head>
...
<meta name="ad.size" content="width=300,height=250">
...
</head>
<body>
</body>
</html>
<head>
```

Veškerý obsah včetně obrázků je potřeba uložit do formátu ZIP.

4.3 Storyboard

Před tvorbou jakékoliv animace je standardně potřeba dodat kreativní podklady, jejichž součástí jsou všechny grafické prvky banneru a z nich sestaven tzv. storyboard celé animace.



Obrázek 8 - Storyboard 1. krok
Zobrazení obrázku Alp



Obrázek 9 - Storyboard 2. krok
Zobrazení titulku Rakousko
Přijetí kontaktu zezdola



Obrázek 10 - Storyboard 3. krok
Pomalé přibližování obrázku Alp



Obrázek 11 - Storyboard 4. krok
Prolnutí titulků Rakouska a Španělska
Změna obrázku a jeho přibližování



Obrázek 12 - Storyboard 5. krok
Zobrazení loga



Obrázek 13 - Storyboard 6. krok
Protáčení paprsků slunce

4.4 Uložení a komprese obrázků

Zmenšené fotografie Španělska a Rakouska byly kvůli velké četnosti barev uloženy do formátu JPG a všechny ostatní pak ve formátu PNG.

Pro hladký průběh a brzké načtení animace by měla směřovat datová velikost obrázků k minimu, obrázky tedy vyžadují kompresi. Jeden z mnoha on-line nástrojů, který umí zmenšit datovou velikost obrázků bez zjevné ztráty kvality, je www.tinypng.com.

Tabulka níže znázorňuje procentuální datovou úsporu každého obrázku po kompresi.

Název	Před	Po	Úspora
Logo společnosti (<i>logo.png</i>)	7,6 kB	2,4 kB	– 69 %
Sluneční záře (<i>slunce.png</i>)	2,4 kB	843 B	– 65 %
Spodní panel s kontakty (<i>txt-kontakt.png</i>)	5,1 kB	1.7 kB	– 67 %
Rakousko – horní štítek (<i>txt-rakousko.png</i>)	2,1 kB	753 B	– 64 %
Španělsko – horní štítek (<i>txt-spanelsko.png</i>)	3,2 kB	813 B	– 75 %
Rakousko - Alpy (<i>alpy.jpg</i>)	65,2 kB	30,9 kB	– 53 %
Španělsko - Bazén (<i>spanelsko.jpg</i>)	224 kB	25,9 kB	– 88 %
Celkem	309,6 kB	63,309 kB	– 80 %

Tabulka 2 - Srovnání datové náročnosti před a po kompresi

Zdroj: vlastní zpracování

Z původních 309,6 kB mají všechny obrázky datovou velikost 63,3 kB, ušetřilo se 80 % dat. Vzhledem k předpokladu, že HTML soubor bude zabírat pouze několik kB, je velikost všech obrázků 63,3 kB dostatečná a dá se s jistotou potvrdit, že stanovený 150kB limit pro velikost všech souborů v HTML banneru je splněn.

4.5 Tvorba HTML šablony

Celý proces začíná sestavením základní HTML šablony, do které postupně vložíme všechny HTML elementy reprezentující obrázky.

Rodičem všech obrázků musí být HTML element, který má pevnou výšku a šířku uvedenou v pixelech a zároveň má zakázané přetékaní. V bakalářské práci je označen rodičovský prvek identifikátorem **obal-animace**.

```
#obal-animace {  
    width: 300px;           // nastavení šířky na 300 px  
    height: 250px;        // nastavení výšky na 250 px  
    overflow: hidden;     // zakázání přetekání  
}
```

Do rodičovského elementu byly postupně vkládány všechny elementy (obrázky) HTML banneru.

Název obrázku	Identifikátor
alpy.jpg	#alpy
spanelsko.jpg	#spanelsko
txt-rakousko.png	#txt-rakousko
txt-spanelsko.png	#txt-spanelsko
txt-kontakt.png	#txt-kontakt
logo.png	#logo
slunce.png	#slunce

Tabulka 3 - Identifikátory obrázků v animaci

```
<a id="obal-animace">  
    <div id="alpy"></div>  
    <div id="spanelsko"></div>  
    <div id="txt-rakousko"></div>  
    <div id="txt-spanelsko"></div>  
    <div id="txt-kontakt"></div>  
    <div id="logo"></div>  
    <div id="slunce"></div>  
</a>
```

4.6 Příprava keyframů v CSS

Pro jednu iteraci banneru je k dispozici 15 s. Stylopis vyžaduje definovat tuto časovou délku každému elementu pomocí vlastnosti `animation-duration` (délka trvání animace). Příklady uvedené níže se týkají HTML elementu s identifikátorem `#alpy`.

```
#alpy {animation-duration: 15s;}
```

Pro počet iterací slouží CSS vlastnost `animation-iteration-count`. Jelikož technická specifikace Google AdWords povoluje maximální časovou délku banneru 30 s, počet iterací bude roven dvěma.

```
#alpy {animation-iteration-count: 2;}
```

K propojení keyframu s HTML elementem slouží vlastnost `animation-name` (název animace).

```
#alpy {animation-name: alpy;}
@keyframes alpy {
    0% {opacity: 1;transform: scale(0.85,0.85);} // 0 s
    33% {opacity: 1;transform: scale(1,1);} // 5 s
    100% {opacity: 0;} // 15 s
}
```

Pro definování konstantního přímočarého průběhu přechodu se využívá vlastnost `animation-timing-function` s hodnotou `linear`.

```
#alpy {animation-timing-function: linear;}
```

4.7 Záložní podoba banneru pro prohlížeč, který nepodporuje keyframy

Celosvětově méně než 0,4 % uživatelů používá prohlížeče Internet Explorer 9 a starší (W3Schools: The Internet Explorer Browser statistics). Tyto verze prohlížečů nepodporují CSS animace, a tak je potřeba vytvořit záložní zobrazení banneru ve statické podobě (W3Schools: CSS3 @keyframes Rule).

Před tvorbou statického banneru je potřeba zvážit, které obrázky z animace budou použity (zpravidla se rozhoduje na základě důležitosti jejich sdělení). Autor práce se rozhodl zvolit logo firmy, telefonní kontakt a název webové stránky. Zvoleným HTML elementům,

keré se mají zobrazit ve statické podobě, je potřeba nastavit v CSS viditelnost opacity (neprůhlednost).

```
#logo, #txt-kontakt {  
    opacity: 1;        // nastavení úplné neprůhlednosti  
}
```

Alternativou pro opacity může být zobrazování přes CSS vlastnost display (zobrazení), která však není pro animování příliš vhodná, protože nedokáže pracovat s postupným zapínáním a vypínáním v čase (může být buď zapnutá, nebo vypnutá – neexistuje nic mezi tím). Oproti tomu opacity může nabývat jakékoli hodnoty od 0 do 1.

```
#logo, #txt-kontakt {  
    display: block;    // blokové zobrazení elementů  
}
```



Obrázek 14 - Screenshot animace z IE 8

Zdroj: vlastní zpracování

5 Zhodnocení

5.1 Test zobrazení animací v prohlížečích

5.1.1 Desktopová zařízení

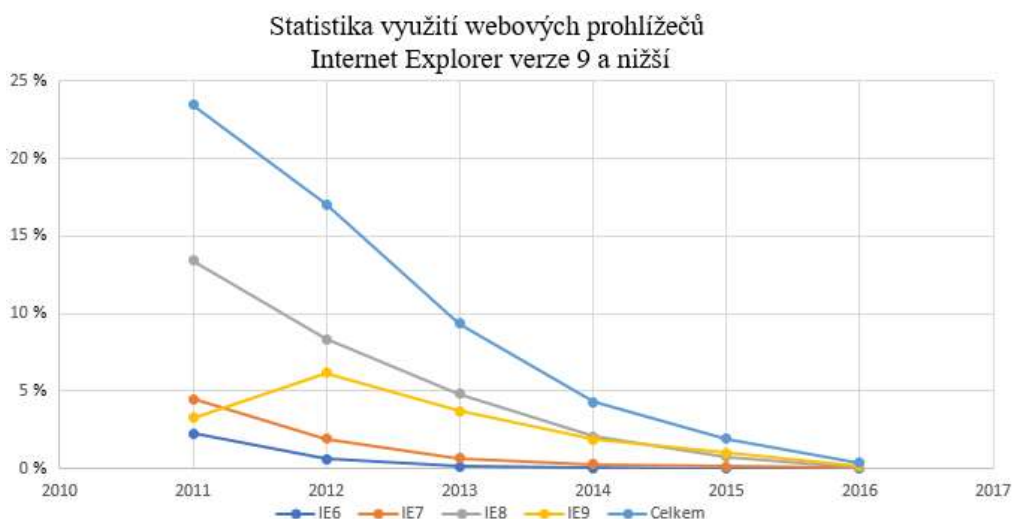
HTML a Flash animace byly otestovány ve všech běžně dostupných prohlížečích v operačním systému Windows 10. Kritérium pro testování představovalo korektní zobrazení animace.

Prohlížeč	Chrome	Firefox	Opera	Safari	Edge	IE 11	IE 10	IE 9 a starší
HTML	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Ne
Flash	Ano*	Ne	Ne	Ne	Ne	Ne	Ne	Ne

Tabulka 4 - Testování podpory animace v desktopových prohlížečích

* Prohlížeč má integrovaný doplněk Adobe Flash Player; Zdroj: vlastní zpracování

Pro úspěšné přehrání HTML animace není potřeba instalovat jakýkoliv doplněk do prohlížeče, naopak pro přehrání Flash animace je potřeba mít nainstalovaný Adobe Flash Player. Prohlížeče, které nepodporují CSS keyframy (např. IE 9 a starší verze), pracují s ostatními definovanými styly. V takových případech prohlížeč vykreslí pouze statický stav.



Obrázek 15 - Statistika IE9 a nižší

Zdroj: W3schools, Graf: vlastní zpracování

Ještě v roce 2013 činil zástup verzí IE9 a starších 10 %. Zastoupení těchto prohlížečů je už dnes ale prakticky nulové. V roce 2016 bylo zaznamenáno celkové zastoupení pouze 0,4 %. Takový výskyt můžeme považovat za zanedbatelný.

Animace byly testovány v základních instalacích webových prohlížečů. Flash animace nebyla přehrána v prohlížečích Firefox, Opera, Safari, Edge a IE, protože neobsahují v základním instalačním balíčku přehrávač Flash animací. Pro úspěšné přehrávání Flash animace v těchto prohlížečích je potřeba přidat doplněk Adobe Flash Player.

5.1.2 Mobilní zařízení

Bannery byly otestovány na mobilních zařízeních iPhone 6S (iOS), Nokia Lumia 530 (Windows Phone) a Samsung S5 (Android).



Obrázek 16 - Test animace na iPhone 6S / Safari



Obrázek 17 - Test animace na iPhone 6S / Chrome



Obrázek 18 - Test animace na Nokia Lumia 530 / Internet Explorer



Obrázek 19 - Test animace na Samsung S5 mini / Android Browser

HTML animace se úspěšně přehrála na všech testovaných mobilních zařízeních. Ani na jednom testovaném zařízení se však nezdařilo zobrazení Flash animace, neboť žádný prohlížeč neměl integrovaný Adobe Flash Player.

V oficiálních obchodech s aplikacemi (App Store, Google play a Windows Store) není možnost instalace Flash Playeru jako doplněk k prohlížeči. Existují pouze aplikace, které se popisují jako Flash přehrávače, nicméně přehrávání se u nich odehrává na vzdáleném serveru a do mobilního zařízení se pouze předávají data pro zobrazení. Mezi takové aplikace patří například Puffin nebo Photon.

5.2 Porovnání datové velikosti HTML a Flash výstupu

Jediný výstup Flash animace představuje soubor ve formátu swf, který do sebe zapouzdřil potřebný kód a všechny obrázky.

HTML animace		Flash Animace	
alpy.jpg	30,9 kB	animace.swf	67,1 kB
logo.png	2,4 kB	index.html	268 B
slunce.png	843 B		
spanelsko.jpg	25,9 kB		
txt-kontakt.png	1,7 kB		
txt-rakousko.png	753 B		
txt-spanelsko.png	813 B		
index.html	5 kB		
Celkem	68,309 kB		

Tabulka 5 - Porovnání datové náročnosti HTML a Flash výstupu

Zdroj: vlastní zpracování

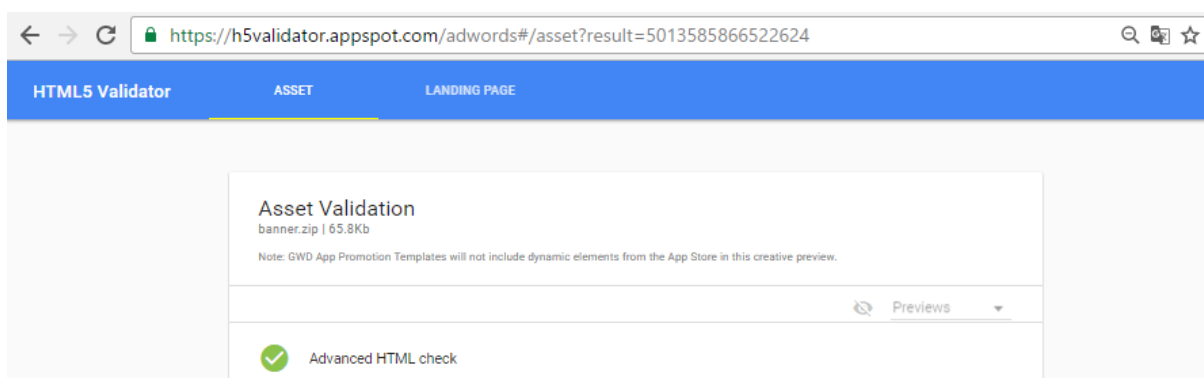
Z tabulky vyplývá, že HTML animace má přibližně o 1 kB větší datovou velikost než totožná Flash animace. Takový rozdíl nepředstavuje v současné době žádné větší zatížení, a tak ho lze považovat za zanedbatelný. Pokud bychom se rozhodli u HTML animace

minifikovat soubor index.html, pak bychom animaci ušetřili již zmíněný 1 kB a obě animace by byly s přesností na jeden kilobyte datově stejně veliké.

5.3 Validace HTML banneru pro reklamní systém AdWords

Google na adrese <https://h5validator.appspot.com> nabízí on-line validátor reklamních bannerů, který kontroluje dodržení všech technických specifikací.

Autor práce předal k testování všechny soubory (HTML dokument a obrázky) komprimované do formátu ZIP. Validátor označil nahraný banner za vyhovující, je tedy připraven k nasazení do reklamního systému AdWords.



Obrázek 20 - Google validátor

Zdroj: Google HTML5 validátor

K otestování Flash animace nedošlo, jelikož bylo předem známo, že reklamní systém společnosti Google od roku 2016 již tento formát animací nepodporuje.

6 Závěr

Hlavním cílem bakalářské práce bylo porovnání HTML a Flash animací. Zásadní příčinu neaktuálnosti Flash animací lze spatřovat ve stále menší podpoře zobrazování ve webových prohlížečích. Zde pak hraje roli absence Adobe Flash Playeru, a to především na mobilních zařízeních, jejichž zastoupení na trhu oproti desktopu neustále roste. Oproti tomu HTML animace se až na výjimky starých prohlížečů zobrazují ve většině webových prohlížečů.

Google se na začátku roku 2016 rozhodl Flash animace ve svých reklamních systémech definitivně zakázat. Tomuto rozhodnutí nepochybně předcházela skutečnost, že společnost Adobe ukončila vývoj přehrávače animací pro webové prohlížeče a vyjádřila podporu HTML5. Tato prohlášení vypovídají o tom, že technologie Flash animací se dala definitivně na ústup, do budoucna se s ní již nepočítá. Současnou technologií pro tvorbu webových animací je nepochybně HTML5.

Nástupu HTML animací také pomohlo stále menší zastoupení prohlížečů, které s nimi neumí pracovat (Internet Explorer 9 a jeho starší verze). V dnešní době je výskyt takových prohlížečů téměř nulový.

Za největší problém tvorby složitějších HTML5 animací, které by se obešly bez načítání externích JavaScript knihoven, je považován nedostatek kvalitního softwaru. Do budoucna směřují naděje především do aplikace Google Web Designer, která je zatím dostupná pouze v beta verzi.

V rámci bakalářské práce byly vytvořeny dva totožné HTML a Flash reklamní bannery, které lze nalézt v přílohách bakalářské práce. Tvorbě bannerů předcházelo vytvoření storyboardu, kde bylo definováno, jak se mají jednotlivé prvky chovat v čase. Všechny použité obrázky byly vystaveny kompresi z důvodu zmenšení datové náročnosti. V praktické části byla shrnuta specifikace pro reklamní bannery systému Google AdWords, na základě které se vytvářel HTML banner. Flash animace byla vytvořena pouze pro účely porovnání výstupů, jelikož bylo předem známo, že její formát již Google AdWords nepodporuje.

Oba vytvořené bannery byly vzájemně porovnány. Vytvořené reklamní bannery podstoupily vzájemné srovnání z pohledu datové velikosti, kde bylo zjištěno, že mezi HTML a Flash animací není výrazný rozdíl. HTML animace byla korektně zobrazena na desktopových i mobilních zařízeních, nezdařilo se však její zobrazení ve starších verzích Internet Exploreru, kde byla ke zhlédnutí jen statická alternativa banneru. Flash animace byla zobrazena pouze v desktopových prohlížečích s předinstalovaným modulem Adobe Flash Player. Na mobilních telefonech se animace nezobrazila, což se dá vzhledem k neustálému

nárůstu počtu mobilních zařízení obecně považovat pro Flash animace za největší handicap. Na základě studie provedené v této práci a s přihlédnutím ke slabé podpoře zobrazení ve webových prohlížečích již nemá smysl vytvářet Flash animace. Tvůrcům Flash animací je doporučeno přejít k HTML.

Popisovaný reklamní banner vznikl na podporu fiktivní cestovní kanceláře. Obsahoval fotografie slunného Španělska a rakouských Alp, čímž autor bakalářské práce dokázal představit oba produkty cestovní kanceláře v jednom reklamním formátu, což lze považovat za jednu z hlavních předností dynamických reklam. S dynamikou bannerů získává vývojář pomocí efektů větší možnost zaujmout potenciálního zájemce. Do procesu tvorby banneru se zapojil efekt postupného přibližování fotografie, zpomaleného vyjíždění bloku s kontaktními informacemi a mírné protočení slunce, jež bylo připraveno jako součást loga cestovní kanceláře.

7 Seznam použitých zdrojů

Seznam použité knižní literatury

CASTRO, Elizabeth a Bruce HYSLOP. 2012. *HTML5 a CSS3: názorný průvodce tvorbou WWW stránek*. ISBN 978-80-251-3733-8.

COREMANS, Chris. 2015. *HTML: a beginner's tutorial. First edition*. ISBN 978-1-77197-019-8.

DOMES, Martin. 2012. *Sklik: jednoduše*. ISBN 978-80-251-3760-4.

HOGAN, Brian. 2011. *HTML5 a CSS3: výukový kurz webového vývojáře*. ISBN 978-80-251-3576-1.

KARLINS, David. 2014. *HTML5 and CSS3 for dummies*. ISBN 978-1-118-63965-8.

LARSEN, Rob. 2013. *Beginning HTML & CSS*. ISBN 978-11-183-4018-9.

MUELLER, John. 2014. *CSS3 for dummies. First edition*. ISBN 978-1-118-61240-8.

PERKINS, Todd. 2010. *Flash Professional CS5 bible*. ISBN 978-04-706-0228-7.

ROHDE, Michael. 2013. *Adobe Edge animate CC for dummies*. ISBN 978-11-183-3592-5.

SCHAFER, Steven. 2010. *HTML, XHTML, and CSS bible*. ISBN 978-0-470-52396-4.

SHENOY, Aravind a Gianluca GUARINI. 2013. *HTML5 and CSS3 transition, transformation, and animation*. ISBN 978-1-84951-995-3.

Seznam použité on-line literatury

Adobe: Edge Animate CC [online]. [cit. 2016-10-27]. Dostupné z:

<https://www.adobe.com/cz/products/edge-animate.html>

Adobe Flash Player - Other Versions [online]. [cit. 2016-10-22]. Dostupné z:

<https://get.adobe.com/cz/flashplayer/otherversions/>

Adobe TEAM. *Programování v jazyce ADOBE ACTIONSCRIPT 3.0* [online]. [cit. 2016-10-22]. Dostupné z:

http://help.adobe.com/cs_CZ/ActionScript/3.0_ProgrammingAS3/flash_as3_programming.pdf

AdForm – About company [online]. [cit. 2016-10-19]. Dostupné z:

<http://site.adform.com/company/about/>

BOUDREAUX, Ryan. *Adobe Edge Animate CC review: Putting the animation and responsive options to the test* [online]. [cit. 2016-10-19]. Dostupné z:

<http://www.techrepublic.com/article/adobe-edge-animate-cc-review-putting-the-animation-and-responsive-options-to-the-test/>

Google display ads go 100% HTML5 [online]. [cit. 2016-10-22]. Dostupné z:

<https://plus.google.com/+GoogleAds/posts/dYSJRrrgNjk>

Google HTML5 validátor [online]. [cit. 2016-11-27]. Dostupné z:

<https://h5validator.appspot.com/adwords#/asset?result=5013585866522624>

CHAFFEY, Dave. *Mobile Marketing Statistics Compilation* [online]. [cit. 2017-01-17].

Dostupné z <http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/>

Oficiální webové stránky aplikace Google Web Designer [online]. [cit. 2017-01-16].

Dostupné z <https://www.google.com/webdesigner/>

JANOVSKÝ, Dušan. *CSS styly: úvod* [online]. [cit. 2016-10-14]. Dostupné z:
<https://www.jakpsatweb.cz/css/css-uvod.html>

JANOVSKÝ, Dušan. *Zápisy CSS* [online]. [cit. 2016-10-14]. Dostupné z:
<http://polopate.jakpsatweb.cz/?page=zapis-css>

MICHÁLEK, Martin. *CSS3 Animations: plnohodnotné animace* [online]. [cit. 2016-10-19].
Dostupné z: <http://www.vzhurudolu.cz/prirucka/css3-animations>

Nápověda AdWords [online]. [cit. 2016-10-22]. Dostupné z:
<https://support.google.com/adwords/answer/1722096>

Nápověda Google Web Designer: Co je to Google Web Designer? [online]. [cit. 2016-10-22].
Dostupné z:
https://support.google.com/webdesigner/answer/3184833?hl=cs&ref_topic=3249465

Nápověda Google Web Designer: Zobrazení náhledu reklamy nebo stránky HTML [online].
[cit. 2016-10-22]. Dostupné z:
https://support.google.com/webdesigner/answer/3261497?hl=cs&ref_topic=3178447

ROUSE, Margaret. *WYSIWYG* [online]. [cit. 2016-10-19]. Dostupné z:
<http://whatis.techtarget.com/definition/WYSIWYG-what-you-see-is-what-you-get>

RTB – real time bidding [online]. [cit. 2016-10-19]. Dostupné z:
<http://www.doublesense.cz/wiki/rtb-real-time-bidding>

VOKÁČ, Luděk. *Adobe končí s Flash Playerem v mobilních prohlížečích* [online]. [cit. 2016-10-22]. Dostupné z: http://mobil.idnes.cz/adobe-konci-s-flash-playerem-v-mobilnich-prohlizecich-p0y-/mob_tech.aspx?c=A111113_111712_mob_tech_vok

W3Schools. *CSS @keyframes Rule* [online]. [cit. 2016-10-19]. Dostupné z:
http://www.w3schools.com/cssref/css3_pr_animation-keyframes.asp

W3Schools. *CSS3 animation-timing-function Property* [online]. [cit. 2016-10-19]. Dostupné z: http://www.w3schools.com/cssref/css3_pr_animation-timing-function.asp

W3Schools. *The Internet Explorer Browser statistics* [online]. [cit. 2016-10-19]. Dostupné z: http://www.w3schools.com/browsers/browsers_explorer.asp

WINOKUR, Danny. *Flash focus* [online]. [cit. 2016-10-22]. Dostupné z: <https://blogs.adobe.com/conversations/2011/11/flash-focus.html>

8 Seznam obrázků

Obrázek 1 - Zápis CSS pravidla	13
Obrázek 3 - Ukázka transformace - rotate	18
Obrázek 4 - Ukázka transformace - translate.....	18
Obrázek 5 - Ukázka transformace - scale	18
Obrázek 6 - Ukázka transformace - skew	19
Obrázek 7 - Ukázka více transformací	19
Obrázek 8 - Celosvětový počet uživatelů (miliony)	22
Obrázek 9 - Storyboard 1. krok	26
Obrázek 10 - Storyboard 2. krok	26
Obrázek 11 - Storyboard 3. krok	26
Obrázek 12 - Storyboard 4. krok	26
Obrázek 13 - Storyboard 5. krok	26
Obrázek 14 - Storyboard 6. krok	26
Obrázek 15 - Screenshot animace z IE 8	30
Obrázek 16 - Statistika IE9 a nižší.....	31
Obrázek 17 - Test animace na iPhone 6S / Safari.....	32
Obrázek 18 - Test animace na iPhone 6S / Chrome	32
Obrázek 19 - Test animace na Nokia Lumia 530 / Internet Explorer.....	32
Obrázek 20 - Test animace na Samsung S5 mini / Android Browser	32
Obrázek 21 - Google validátor.....	34

9 Seznam tabulek

Tabulka 1 - Formáty reklamy Google AdWords	24
Tabulka 2 - Srovnání datové náročnosti před a po kompresi	27
Tabulka 3 - Identifikátory obrázků v animaci	28
Tabulka 4 - Testování podpory animace v desktopových prohlížečích.....	31
Tabulka 5 - Porovnání datové náročnosti HTML a Flash výstupu.....	33

10 Slovník pojmů a zkratek

Adobe AIR – Adobe Integrated Runtime. Program umožňují vývojářům publikovat aplikace se stejným zdrojovým kódem pro více zařízení s odlišným operačním systémem.

CSS – Cascading Style Sheets. Kaskádové styly, slouží pro stylování HTML elementů.

HTML – HyperText Markup Language. Značkovací jazyk.

RTB – Real Time Bidding. Prodej reklamního prostoru v reálném čase.

Storyboard – Popsání průběhu animace po jednotlivých částech.

W3C – World Wide Web Consortium. Konsorcium vydávající webové standardy.

WYSIWYG – What You See Is What You Get. V překladu: co chceš, to dostaneš. Zkratka označující editory, které mají připravený náhled výstupu po každé úpravě.

11 Přílohy

Příloha č. 1 – HTML animace

Dostupná on-line na adrese <http://kitlab.pef.czu.cz/~bendap/DP/Kolar/Animace-html/>

Příloha č. 2 – Flash animace

Dostupná on-line na adrese <http://kitlab.pef.czu.cz/~bendap/DP/Kolar/Animace-flash/>