

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



Bakalářská práce

Tvorba webových stránek pomocí HTML5

Dmitry Arkhipenko

© 2017 ČZU v Praze

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Dmitry Arkhipenko

Informatika

Název práce

Tvorba webových stránek pomocí HTML5

Název anglicky

Creating a Website using HTML5

Cíle práce

Bakalářská práce se bude zabývat tvorbou webových stránek pomocí jazyka HTML. Cílem teoretické části bude obecný přehled vývoje jazyka, popis nových elementů a jejich použití, a také porovnání HTML 4.01 a HTML 5.

Metodika

Bakalářská práce je založena na analýze již existujících webových stránek. Znalosti a dovednosti získané studiem odborné literatury a kódu existujících webových stránek budou použity v dané práci, na základě kterých budou popsány tagy HTML, vysvětleny postupy tvorby webových stránek a také poskytnuta další nezbytná informace, která bude souviset s přidáváním kódu CSS.

Doporučený rozsah práce

30 – 40 stran

Klíčová slova

HTML, CSS, tvorba webu, webové stránky

Doporučené zdroje informací

BROWN, Tiffany B., Kerry BUTTERS a Sandeep PANDA. HTML5 okamžitě: [ovládněte HTML5 za víkend].

Brno: Computer Press, 2014. ISBN 978-80-251-4296-7.

GOLDSTEIN, Alexis, Louis LAZARIS a Estelle WEYL. HTML5 a CSS3 pro webové designéry. Brno: Zoner Press, 2011. Encyklopedie webdesignera. ISBN 978-80-7413-166-0.

KOSEK, J. *HTML : tvorba dokonalých www stránek : podrobný průvodce*. Praha: Grada, 1998. ISBN 80-7169-608-0.

Předběžný termín obhajoby

2016/17 LS – PEF

Vedoucí práce

Ing. Mgr. Vladimír Očenášek, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 18. 10. 2016

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 24. 10. 2016

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 07. 03. 2017

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Tvorba webových stránek pomocí HTML5" 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 použitých zdrojů 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 11. března 2017

Poděkování

Rád bych touto cestou poděkoval Mgr. Ing. Vladimíru Očenáškoví, Ph.D. za cenné rady, podporu a připomínky, které mi poskytl v průběhu zpracování této bakalářské práce. Rád bych také poděkoval své rodině a blízkým za podporu při studiu.

Tvorba webových stránek pomocí HTML5

Souhrn

Bakalářská práce se zabývá tvorbou webových stránek pomocí HTML5. Práce je rozdělaná do dvou částí: teoretická východiska a vlastní práce. V teoretických východiscích jsou popsány nové funkce, které přináší specifikace HTML5, a jejich použití v praxi. Dále je provedeno porovnání dvou verzí HTML 4.01 a HTML5 pomocí zápisu hlavní struktury zdrojového kódu dokumentu. Také je součástí analýza již existujících webových stránek, kde byla vybrána jedna webová stránka a následně provedena kontrola, jestli ve zdrojovém kódu jsou použity nové funkce HTML5, zda zdrojový kód je validní a jestli daný web má responzivní design na různých typech zařízení. Ve vlastní práci jsou popsány jednotlivé kroky tvorby webové stránky. Po vytvoření webové stránky jsou provedeny kontrola validity zdrojového kódu a testování nových funkcí HTML5, použitých při tvorbě webu, v různých verzích internetových prohlížečů. Součástí testování vytvořené stránky je také responzivní design na mobilním zařízení a tabletu.

Klíčová slova: HTML, CSS, tvorba webu, webové stránky, responzivní design, HTML5 audio a video.

Creating a Website using HTML5

Summary

Bachelor thesis is following up creating a website using HTML5. The thesis is divided into two parts: theoretical points and practical part. In the theoretical part are described the new features that brings HTML5 and their use in practice. Next, it is a comparison of the two versions HTML 4.01 and HTML5 by writing the main structure of the source code of document. It is also part of analysis already existing websites where has been selected one website and then checked if the source code is used new features of HTML5, if the source code is valid and if the website has a responsive design for different types of devices. In practical part are described individual steps of creating a website. After creating a website were checked the validation of source code and tested new features of HTML5 in the various versions of web browsers. Also, part of website testing was responsive design on mobile device and tablet.

Keywords: HTML, CSS, web development, website, responsive design, HTML5 audio and video.

Obsah

1 Úvod	9
2 Cíl práce a metodika	10
2.1 Cíl práce	10
2.2 Metodika	10
3 Teoretická východiska	11
3.1 HTML co to je?	11
3.1.1 Vývoj jazyka	11
3.1.2 HTML5 vývoj	12
3.1.3 Rysy HTML5	13
3.2 Adresy ve WWW – URL	15
3.3 MIME-typy	16
3.4 DOCTYPE	17
3.5 W3C	18
3.6 Sémantika v HTML5	19
3.7 Nové elementy ve verzi HTML 5.0	20
3.8 Podpora HTML5 webovými prohlížeči	24
3.9 Porovnání HTML 4.01 a HTML5	25
3.10 HTML5 audio a video	27
3.11 HTML formuláře	30
3.12 HTML5 vs Adobe Flash	31
3.13 Analýza vybraného webu	32
3.14 CSS	33
3.14.1 Syntaxe	33
3.14.2 Způsoby deklarování kaskádových stylů	34
3.14.3 Bootstrap	35
4 Vlastní práce	37
4.1 Tvorba webové stránky	37
4.1.1 Struktura	37
4.1.2 Kódování stránky	38
4.1.3 Grafická část	39
4.2 Validita zdrojového kódu	43
4.3 Testování použitelnosti HTML5	43
4.3.1 Testování audio a video	43
4.3.2 Nové atributy formuláře	44
4.3.3 Zobrazení v Internet Explorer	45

4.3.4 Responzivní design	45
5 Výsledky a diskuze.....	48
6 Závěr	49
7 Seznam použitých zdrojů	50
8 Přílohy.....	51

Seznam obrázků

Obrázek 1 - Sémantika HTML5	20
Obrázek 2 - Test TOP 5 prohlížečů	24
Obrázek 3 - Vzhled audio přehrávače v prohlížeči "Google Chrome"	27
Obrázek 4 - Vzhled video přehrávače v prohlížeči "Google Chrome"	30
Obrázek 5 - Syntaxe CSS.....	33
Obrázek 6 – Struktura stránky	38
Obrázek 7 - Rozdělení do čtyř sloupců.....	40
Obrázek 8 - Rozbalovací navigační panel	40
Obrázek 9 - Fotografie ve formě kruhu	41
Obrázek 10 - Efekt zasvícení obrázku	41
Obrázek 11 - Formulář do vyplnění.....	42
Obrázek 12 - Formulář po vyplnění.....	42
Obrázek 13 - Tlačítko před najetím myši	42
Obrázek 14 - Tlačítko po najetí myši.....	42
Obrázek 15 - Kontrola vyplnění formuláře.....	44
Obrázek 16 - Zobrazení v Internet Explorer 6	45
Obrázek 17 - Mobil (360×640).....	46
Obrázek 18 - Mobil (640×360).....	46
Obrázek 19 - Tablet (768×1024)	46
Obrázek 20 - Tablet (1024×768)	46
Obrázek 21 - Notebook (1366×768).....	47

Seznam tabulek

Tabulka 1 - Sémantické tagy HTML5	19
Tabulka 2 – Deklarace typu dokumentu	25
Tabulka 3 - Metadata	25
Tabulka 4 - Tělo dokumentu.....	26

1 Úvod

S každým dnem internet je stále více součástí běžného života člověka. Všechny informace na internetu člověk získává pomocí webových stránek, většina z kterých používá pro navržení své struktury značkovací jazyk HTML. HTML slouží stavebním kamenem pro vytváření webové prezentace, na kterém již lze dále stavět grafickou část a přidávat další potřebné funkce.

V celosvětové síti po dlouhou dobu se používaly standardy HTML 4.01, které měly v poslední době řadu vylepšení. Však se technologie neustále rozvíjejí a tím vznikají nové požadavky na web, které nutí vylepšovat daný značkovací jazyk. První myšlenka vytvořit novou verzi značkovacího jazyka vznikla v roce 2004, kdy se aktivně vyvíjely v oblasti informačních technologií takové významné firmy jako jsou Google, Apple, Microsoft.

Novým standardem se objevuje verze HTML5, která přináší mnoha nových funkcí. Hlavním průlomem nové verze jsou zavedení nových strukturálních značek, které mají sémantický význam pro vyhledávací systémy, a možnost vkládat do webových stránek audio a video soubory bez nutnosti používání doplňkových modulů pro jejich přehrávání.

2 Cíl práce a metodika

2.1 Cíl práce

Bakalářská práce se bude zabývat tvorbou webových stránek pomocí jazyka HTML. Cílem teoretické části bude obecný přehled vývoje jazyka, popis nových elementů a jejich použití, a také porovnání HTML 4.01 a HTML5.

2.2 Metodika

Bakalářská práce je založena na analýze již existujících webových stránek. Znalosti a dovednosti získané studiem odborné literatury a kódu existujících webových stránek budou použity v dané práci, na základě kterých budou popsány tagy HTML, vysvětleny postupy tvorby webových stránek a také poskytnuta další nezbytná informace, která bude souviset s přidáváním kódu CSS.

3 Teoretická východiska

3.1 HTML co to je?

HTML (z angličtiny "HyperText Markup Language") – je standardizovaný značkovací jazyk dokumentů ve WWW (World Wide Web). Většina webových stránek obsahují značky jazyka HTML (nebo XHTML). Jazyk HTML je interpretován prohlížečem; získaný po interpretaci formátovaný text se zobrazí na monitoru počítače nebo mobilního zařízení. [3]

Jazyk HTML je aplikací SGML (z angličtiny "Standard Generalized Markup Language") a splňuje mezinárodní standard ISO 8879. Jazyk XHTML (z angličtiny "eXtensible HyperText Markup Language") je přísnější alternativou HTML, splňuje všechna omezení XML (z angličtiny "eXtensible Markup Language") a ve skutečnosti lze tento jazyk vnímat jako aplikace jazyka XML do oblasti hypertextových značek. [3]

Ve WWW HTML stránky jsou přenášeny do prohlížeče ze serveru na základě protokolu HTTP nebo HTTPS, v podobě prostého textu nebo s použitím šifrování. [3]

3.1.1 Vývoj jazyka

První verze HTML byla vyvinuta v roce 1989 Timem Berners-Lee a Robertem Cailliau pro populární v minulosti prohlížeč Mosaic. Ale v té době ani jazyk, ani prohlížeč nebyly použity v praxi. V roce 1993 se objevil HTML+ a tato verze také zůstala prakticky bez povšimnutí. Začátek širokého použití hypertextu dala verze 2.0, která vznikla v červnu roku 1994. To byl rok růstu popularity WWW po celém světě. Prvky, zahrnuté ve verzi 2, většinou se používají až dodnes. [3]

Ve verzi 3.0 HTML, která se objevila o rok později, byla implementována možnost vytvářet matematické symboly (značky intervalu, nekonečnosti, zlomku, závorky atd.) pomocí prvků jazyka. V této verzi byl vyvinut prohlížeč "Arena". Ale tento projekt se dostal do slepé uličky a neobdržel dalšího šíření. [3]

V roce 1996 se objevil HTML 3.2. To bylo inovativní řešení, ve specifikaci jazyka byly zavedeny rámy (značka <frame>), které se nyní staly velmi populární mezi vývojáři webových stránek. Až do teď na základě této specifikace je možné realizovat zajímavé designové řešení. Skoro všechny moderní prohlížeče podporují tuto verzi. [3]

Spolu s oficiálními specifikacemi jazyka, které vyvíjela organizace W3C, firmy-výrobci prohlížečů vytvářely vlastní prvky (rozšíření). Následně, některé z těchto prvků, po

získání všeobecného uznání, byly zahrnuty ve specifikaci následujících verzí jazyka. Ale inovativní řešení – rámy, nebyly zahrnuty ve specifikaci 3.2. Však prohlížeče podporovaly rámy a mnohé knihy o HTML obsahovaly popis rámu bez zmínky o tom, že jsou to nestandardní prvky. Později, rámy se staly de facto standardem. Ve verzi 4 již byly plně zahrnuty. [3]

Naopak, prvky applet (značka `<applet>`) a script (značka `<script>`), nezbytné pro rozšíření HTML dalšími softwarovými kódy verze 3.2, nezahrály takové role, které by měly zahrát. To je způsobeno tím, že webové prohlížeče různých verzí odlišně interpretovaly programy v různých jazycích jako Java, Javascript, Visual Basic. V důsledku toho nepodařilo se získat dostatečně spolehlivý funkční kód a dané jazyky amatéry HTML používali pro experimentování ve vlastních zájmech. [3]

Oficiální specifikace HTML 4 (Dynamic HTML) se objevila v roce 1997. V této době již bylo zřejmé, že další rozvoj hypertextu bude probíhat prostřednictvím script-programování. To se ukázalo o něco účinnější než zadat do jazyka všechny nové prvky. Vzniklé v té době prohlížeče (Netscape Navigator 4, Microsoft Internet Explorer 4 atd.) již poměrně bezpečně interpretovaly programový kód (byla nastavena na určitou úroveň standardizace). Problémy však vývojářům ještě zůstali. Jako příklad lze uvést, že mnoho skriptů začínají určením verze prohlížeče, aby potom použít ten či onen kus kódu. Je zřejmé, že v tomto případě programátor by měl provést testování stránek ve všech populárních v současné době prohlížečích. V důsledku toho, použití všech funkcí Dynamic HTML stalo údělem programátorů poměrně velkých organizací, kde jsou podmínky pro vývoj složitých programů a jejich komplexní testování. [3]

3.1.2 HTML5 vývoj

Standard HTML má za sebou dlouhou evoluční cestu. Období rychlého rozvoje se změnilo klidem, budoucnost HTML připadalo poměrně nejasným. Zaměření webové standardizace se posunulo směrem k XML a XHTML, zatímco HTML byl zatlačen do pozadí. Nicméně HTML vůbec neměl v úmyslu zemřít, na něm stále stavěla značná část obsahu webu. Pro přechod k webovým aplikacím nového typu a překonání stávajících nedostatků HTML potřeboval v nových prostředcích a specifikacích. [1]

V roce 2004 z iniciativy malé skupiny odborníků, kteří byli představiteli řady významných výrobců prohlížečů a měli zájem o zvedání web-platforem na novou technologickou úroveň, byla založena komunita WHATWG (Web Hypertext Application

Working Group). Díky úsilí její členů se objevila specifikace HTML5. Tato komunita zahájila vývoj nových nástrojů, které jsou určeny speciálně pro webové aplikace, protože jasně věděli, jak velká je potřeba v takových prostředcích. Právě v té době začal používat termín "Web 2.0". Všechno to má důvod se domnívat, že jsme skutečně svědci druhého obrození Internetu, pokud první považovat za období, kdy statické webové stránky byly nahrazeny dynamickými a sociálně orientovanými webovými stránkami, které potřebovaly mnohem větší zpestření různými nástroji v jejich arzenálu. [1]

V roce 2006 pracovat nad HTML byl opět připojen Konsorcium W3C a začaly se objevovat trvalé nebo dočasné odkazy na různé zdroje, v důsledku čehož v roce 2008 svět uviděl první pracovní verzi projektu doporučení na HTML5, zatímco pracovní skupina, která se zabývala vývojem specifikace XHTML 2, v roce 2009 ukončila svou činnost. Jelikož HTML5 řeší řadu důležitých praktických problémů, výrobci prohlížečů vedou aktivní práci na zavádění nových příležitostí, a to i přes to, že do vydání finální verze specifikace je stále velmi daleko. Zase aprobace specifikace využívaných prohlížečích slouží zdrojem cenných informací, který se používá pro zlepšení samotné specifikace. HTML5 se rychle vyvíjí a jde spolu se všemi rostoucími požadavky na webové platformy. [1]

3.1.3 Rysy HTML5

HTML5 je nová otevřená platforma, která je určena k vytváření webových aplikací pomocí audia, videa, grafiky, animace a mnoho dalšího. Základní myšlenkou vývojářů HTML5 bylo udělat životaschopný značkovací jazyk, který by fungoval v praxi a přinášel užitek návštěvníkům a vývojářům webových stránek. Princip je takový: na prvním místě stojí zájmy skutečných uživatelů, pak tvůrce webových stránek, pak vývojářů prohlížečů a dalších aplikací, pak autorů samotného standardu, a teprve pak teoretické a ideologické úvahy, které Konsorcium neúspěšně podporoval posledních deset let. HTML5 by se měl stát nástupcem HTML 4.0 a XHTML 1.0. XHTML je značkovací jazyk, který je založen na XML. Specifikace XHTML 1.0 a XHTML 1.1 jsou doporučení konsorcia World Wide Web. Hlavní rozdíl mezi XHTML a HTML spočívá ve zpracování dokumentu. Dokumenty XHTML jsou zpracovány svým modulem (parser) podobně jako dokumenty XML. V průběhu tohoto zpracování chyby, které se dopustili vývojáři, nejsou opraveny, proto v standardu je kladen velký důraz na zpětnou kompatibilitu. [5]

HTML5 existuje ve dvou variantách: první jako XHTML a druhá jako modifikace historického HTML. Tvůrce webových stránek mohou použít některou z těchto variant. V

první variantě nový jazyk je zcela sjednán s XML 1.0, takže vývojáři mají k dispozici všechny výhody XML a vedlejších technologií. V druhé variantě HTML5 se zbavil děsivým dědictvím minulosti – formální kompatibilitou s SGML. SGML (z angličtiny "Standard Generalized Markup Language") – je metajazyk, na kterém je možné definovat značkovací jazyk pro dokumenty. SGML je dědic vyvinutého v roce 1969 v IBM jazyka GML (Generalized Markup Language), který nelze plést s Geography Markup Language od Open GIS Consortium. Všechny předchozí standardy HTML měly nárok na tuto kompatibilitu, ale ani jeden prohlížeč nezpracovával HTML kód podle pravidel SGML. Místo toho byly použity jednoduché algoritmy, které rozpoznávaly soubor konkrétních elementů a atributů. HTML5 legalizoval toto chování prohlížečů. Nový standard se neohlíží na SGML, ale popisuje soukromá pravidla pro interpretaci konkrétních značek a jiných syntaktických konstrukcí. Ve skutečnosti HTML5 dokumentuje principy, podle kterých prohlížeče zpracovávají kód již posledních patnáct let. Na rozdíl od předchozích standardů, určených především pro vývojáře kódu, HTML5 přísně a podrobně popisuje chování prohlížečů. Vyžaduje od prohlížečů zpracovávat nejen příslušný kód, ale i na možnosti upravování chyb, až na použití chybějících v normě elementů. Například HTML5 neumožňuje autorům kódu použít prvky `nobr` nebo `marquee`, ale popisuje, jak prohlížeč by se měl chovat, pokud se objeví v dokumentu. To odpovídá dalšímu obecnému principu HTML5 – maximální kompatibilita s již existujícím obsahem sítě. Miliony stávajících webových stránek by neměli utrpět kvůli vzniku nového jazyka a podporujících ho prohlížečů. [5]

Je ještě jedna vlastnost nového standardu, která spočívá v popisu dokumentu HTML pomocí objektových modelů DOM (Document Object Model). DOM – je soubor objektů, předložených HTML prvky na stránce. Před tím než zobrazit webovou stránku, prohlížeč vytvoří stromovou struktura její prvků a jiných objektů. Předchozí standardy nic o tom neříkaly, ale HTML5 naopak je do značné míry založen právě na této reprezentaci dokumentu. To umožňuje zahrnout do specifikaci objekty, vlastnosti a metody DOM, které jsou široce používány, ale nikdy nebyly standardizovány: `window.open()`, `alert()`, `prompt()`, `document.links`, `document.images`, `innerHTML`, `contentEditable` a tak dále. Konečně v HTML5 je oficiálně zapsán element `embed`. Značka `<embed>` se používá pro stažení a zobrazení objektů (např. video soubory, flash-animací, některé audio soubory a tak dále), kterým prohlížeč prvotně nerozumí. Obvykle takové objekty vyžadují

připojení k prohlížeči speciálního modulu, který se jmenuje plugin, nebo spuštění pomocného programu. [5]

Nicméně v HTML5 zůstává ne vše, co je podporováno v prohlížečích. Některé prvky naopak byly odstraněny z normy: `acronym`, `applet`, `basefont`, `big`, `center`, `dir`, `font`, `isindex`, `s`, `strike`, `tt`, `u`. Odstraněny rámce (`frame`, `frameset`, `noframes`), i když zůstal prvek `iframe`. Nejsou další atributy, které téměř nikdo nepoužíval (`accesskey`, `longdesc` atd.) nebo které neměly smysl používat (`align`, `border`, `valign`, `cellpadding`, `cellspacing` atd.). Položka `menu` nyní umožňuje vytvářet skutečné menu, a `address` je určen výhradně pro kontaktní informace v užším kontextu. [5]

3.2 Adresy ve WWW – URL

Každý dokument ve World Wide Web má svou vlastní speciální adresu, která se nazývá URL adresa (Uniform Resource Locator – "jednotná adresa zdroje"). Tim Berners-Lee (člen Evropské rady pro jaderné vojenské problémy v Ženevě) v roce 1990 vynalezl URL, který v té době byl prostě adresou umístění souborů v systému. URL adresy stránek se zapisují v podobě absolutní nebo relativní. [3]

Absolutní adresa

Obecné schéma nebo struktura absolutní URL adresy vypadá následujícím způsobem:

«schéma» : // «uživatel» : «heslo» @ «počítač» : «port» / «cesta» ? «dotaz» # «fragment»

Schéma – to je ten protokol pro přenos dat, s jehož pomocí chceme odkazovat na zdroj.

Uživatel a heslo – uživatelské jméno a heslo se používají pro přístup k prostředkům. Ne vždy budou tyto parametry použity. Například pro přístup k jakékoli webové stránce, přes protokol http – zpravidla, tyto údaje nepoužívají.

@ – oddělovač mezi uživatel-heslo a počítačem. V případě, že uživatel-heslo nejsou zadány, pak oddělovač lze rovněž vynechat.

Počítač – doménové jméno nebo IP adresu zdroje, na které je třeba se obrátit.

Port – unikátní číslo, které se poskytuje aplikaci, která bude zpracovávat žádost. Při práci přes protokol http se nejčastěji nastaví automaticky na 80 nebo 8080.

Cesta – zde se uvádí přesná informace o umístění zdroje. Závisí na použitém protokolu. V případě protokolu HTTP se cesta zapisuje s uvedením adresáře a podadresáře, kde je umístěn zdroj.

Dotaz – žádost s určitými parametry, které jsou předávány na server metodou GET. Pro rozdělení parametrů se uplatňuje znak &.

Fragment – je unikátním řetězcem, který se skládá z písmen a/nebo číslic. Odkazuje na určitou oblast otevíracího webového dokumentu. S jeho pomocí lze udělat tak, aby se uživatel dostal ne na začátek HTML dokumentu, ale hned na konkrétní místo v tom dokumentu.

Každý formát URL adresy je především symbolický řetězec, v němž mohou být zahrnuty pouze deklarované symboly. Mezi nimi patří písmena abecedy, arabské číslice (0-9) a speciální znaky ("+", "=", "!", "?", "&", ":", "#", "@"). [3]

Relativní adresa

Relativní adresa představuje sebou zkrácenou cestu k souboru. Často se může stát, kdy vypisování absolutní adresy není vhodné, protože je příliš dlouhá, a někdy to bývá zbytečné. Myšlenkou relativní adresy je to, že většinou soubory, které se odkazují na jiný soubor, leží na jednom serveru. V tom případě, když se bude jeden soubor odkazovat pomocí URL na jiný soubor, není nutné vyplňovat celou absolutní adresu, stačí uvést cestu k danému souboru (všechna oddělení se provádí pomocí lomítka) a název souboru. Takovým způsobem vytvoříme relativní URL adresu. V případě, když několik souborů leží ve stejné záložce, není potřeba uvádět cestu, stačí uvést název souboru. [3]

Takovým způsobem lze zapsat relativní adresu: `/images/picture.png`

3.3 MIME-typy

MIME (Multipurpose Internet Mail Extensions) – standard, který popisuje přenos různých typů dat e-mailem, a také je specifikace pro kódování informací a formátování zpráv tak, aby bylo možné je zasílat na Internetu. MIME definuje mechanismy pro přenos různých druhů informací uvnitř textových dat (zejména prostřednictvím e-mailu) a to právě: text v jazycích, pro které se používá kódování, které se liší od ASCII a netextová data, jako jsou obrázky, programy, filmy a hudba. MIME je také základní složkou komunikačních protokolů, jako jsou HTTP, který potřebuje, aby údaje byly předávány v kontextu zpráv, jako je e-mail, i pokud data opravdu nejsou e-mailem. [7]

Základní formát e-mailových zpráv je definován v RFC 5322, který je aktualizovanou verzí RFC 2822. Tyto normy definují podobné formáty pro textové e-mail-hlavičky, obsah a předpisy vztahující se k společným pro použití polím, jako jsou *To:*, *Subject:*, *From:*, *Date:*. MIME definuje sadu e-mail-záhlaví pro stanovení dalších atributů zpráv, včetně typu obsahu, a určuje řadu znakových sad, které mohou být použity k reprezentaci 8-bit binárních dat pomocí znaků 7-bit ASCII. MIME také definuje pravidla pro kódování znaků z rozšířené ASCII (kódy 128-255) v záhlaví e-mailové zprávy, jako je *Subject:*. Například textový soubor má typ `text/plain` a soubor HTML – `text/html`. Odeslání hlavičky se obvykle provádí na základě přípony souboru webového serveru. [7]

Dokumenty XHTML se standardně odesílají jako `text/html`, což ve skutečnosti mluví o tom, že pracujeme s HTML, ale ne s XHTML souborem. Aby využít možnosti XHTML, je nutné dát soubor s typem `application/xhtml+xml`. Typ `application/xhtml+xml` je nutným v případě, když se v dokumentu používá MathML (Mathematical Markup Language), určený pro přidání vzorců nebo SVG (Scalable Vector Graphics). SVG je značkovací jazyk pro vytváření na stránce vektorové kresby. Pokud nevíme nic o těchto technologiích a nechystáme se je použít, je lepší dát dokument jako `text/html`. To umožní zahrnout co největší počet prohlížečů a vyhledávačů. [7]

3.4 DOCTYPE

DOCTYPE deklarace, která by měla být prvním tagem ve zdrojovém kódu jakékoli webové stránky, je využívána webovým prohlížečem k identifikaci verze značkovacího jazyka, ve kterém je stránka napsána. To může obsahovat odkaz na Definicí Typu Dokumentu nebo zkráceně DTD. DTD stanoví pravidla a gramatiku, což umožňuje prohlížeči zobrazit obsah odpovídajícím způsobem. Mnohé rysy tohoto tagu jsou díky společnosti Microsoft, která při vývoji aplikace Internet Explorer 5 zjistila, že se zlepšila podpora standardů natolik, že se starší stránky nezobrazovaly správně v prohlížečích. [8]

Bez DOCTYPE (nebo při špatném DOCTYPE) prohlížeče zobrazují stránky v režimu kompatibility se staršími prohlížeči, až po simulaci jejich chyb. Kromě toho, moderní funkce (například podpora CSS vlastnosti `padding` pro element `IMG`) je k dispozici pouze v režimu dodržování standardu. Je třeba vždy opatřit dokumenty HTML DOCTYPE-oznámením. [8]

Existuje několik druhů `<!DOCTYPE>`, které se liší v závislosti na verzi jazyka, na kterého jsou zaměřeny. Ve verzi HTML5 je zajištěna XML notace, tzv. XHTML5. V XML klíčová slova (`DOCTYPE` v tomto případě) se zapisují velkými písmeny a názvy prvků jsou citlivé na velikost písma (například `<p></p>` a `<P></P>` – není totéž) a v XHTML musí být zapsány malými písmeny. HTML ve složení `DOCTYPE` – jméno kořenového elementu dokumentu, a proto pro ni je třeba použít malá písmena: `<!DOCTYPE html>`. Tato varianta je kanonická a výhodnější bez ohledu na používanou v HTML5 dokumentu notaci. [8]

3.5 W3C

World Wide Web Consortium (W3C) – organizace, která vyvíjí jednotné principy a normy pro web (tzv. "doporučení", W3C Recommendations), které se pak zavádí výrobci softwaru a hardwaru. Tímto způsobem se dosahuje kompatibilita mezi softwarem a hardwarem různých firem, co dělá celosvětovou síť dokonalejší a univerzálnější. Konsorcium řídí Tim Berners-Lee, autor mnoha vývojových projektů v oblasti informačních technologií. [6]

Cíle W3C

Cílem W3C je pomáhat počítačovým programům dosáhnout schopnosti ke komunikaci v síti (tzv. "síťová interoperabilita"). Použití jednotných standardů v síti – to je klíčový krok pro dosažení takové interoperability. Dalším hlavním úkolem konsorcia je zajistit kompletní "internacionalizaci sítě" a vytvořit síť, která bude dostupná pro osoby se zdravotním postižením. Pro řešení prvního úkolu konsorcium aktivně spolupracuje s organizací "Unicode" a řadou dalších pracovních skupin zabývajících se mezinárodní spoluprací na webu a jazykovými technologiemi. Pro řešení druhého úkolu konsorcium nejen spolupracuje s institucemi daného profilu, ale také vyvinul svá vlastní doporučení, která nyní aktivně získává popularitu. [6]

Implementace doporučení

Doporučení konsorcia World Wide Web jsou otevřené, to znamená, že nejsou chráněny patenty a mohou být zaváděny jakoukoli osobou bez jakýchkoliv finančních příspěvků konsorciu. Na rozdíl od jiných organizací zabývajících se vývojem standardů pro web, konsorcium World Wide Web nemá certifikačních programů na soulad doporučením konsorcia a nemá plánuje jejich zavádění, proto doporučení W3C získaly mnohem větší šíření než normy jiných organizací. Ale kvůli nedostatku certifikace mnozí výrobci následují

doporučení jen částečně. Doporučení konsorcia jsou postavena tak, že částečné zavedení není v rozporu se společnými normami. Některé populární doporučení mají několik stupňů implementaci. Míry implementace – to je nové slovo v síťových standardech, které přineslo konsorciu World Wide Web a jeho doporučení zaslouženou popularitu. [6]

Doporučení W3C často dobře a podrobně vypracovány. Většina doporučení je k dispozici pro všechny kategorie uživatelů od odborníků-programátorů do začínajících webmasterů. Kromě technických specifikací konsorcium také publikuje mnoho návodů a tipů, které usnadňují zavedení doporučení. [6]

Proces standardizace

Každý standard W3C probíhá 5 fází schválení:

- Koncept specifikace (Draft);
- Pracovní návrh (Working Draft);
- Poslední svolání (Last Call);
- Případné doporučení (Candidate Recommendation);
- Navrhovaná doporučení (Proposed Recommendation);

a teprve poté se oficiálně stává doporučením W3C. [6]

Doporučení se mohou čas od času být aktualizovány. K doporučením zveřejňují zprávy o nalezených chybách. Až se nahromadí dost zjištěných chyb, vyjde nová, novelizovaná, dokončená verze doporučení (například "verze 1.1"). Ve výjimečných případech se celé doporučení může být odvoláno konsorciem k přepracování. [6]

3.6 Sémantika v HTML5

HTML5 nabízí řadu strukturních tagů, které rozdělují webovou stránku na logické části: hlavička, navigace, hlavní obsah stránky, patička apod. Toto rozdělení umožňuje strojům a robotům lépe procházet obsah webové stránky.

Přehled sémantických strukturních tagů: [4]

Tabulka 1 - Sémantické tagy HTML5

Tag	Popis	Párový
<article>	nezávislá část webové stránky (např. článek nebo komentář)	ano
<aside>	vedlejší části webové stránky, které s ní souvisí	ano
<details>	slouží pro dodatečnou informaci, kterou uživatel může zobrazit nebo skrýt kliknutím	ano

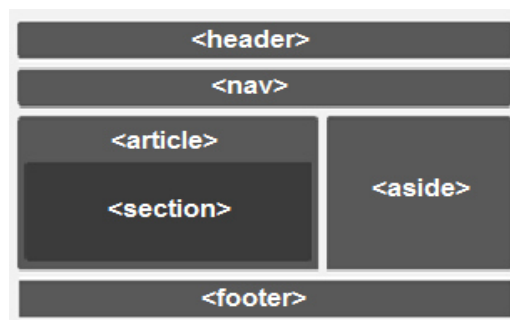
<code><figcaption></code>	nadpis pro <code><figure></code>	ano
<code><figure></code>	samostatný obsah, který slouží jako obalení např. obrázku, diagramu, videa	ano
<code><footer></code>	patička webové stránky	ano
<code><header></code>	hlavička, obsahuje nadpis nebo odkaz	ano
<code><main></code>	hlavní obsah webové stránky	ano
<code><mark></code>	označení/zvýraznění textu	ano
<code><nav></code>	seznam odkazů, které slouží jako navigace webové stránky	ano
<code><section></code>	části/kapitoly webové stránky	ano
<code><summary></code>	legenda pro <code><details></code>	ano
<code><time></code>	čas a datum	ano

Struktura kódu v HTML5 vypadá takto:

```

<body>
  <header>...</header>
  <nav>...</nav>
  <article>
    <section>...</section>
  </article>
  <aside>...</aside>
  <footer>...</footer>
</body>

```



Obrázek 1 - Sémantika HTML5

3.7 Nové elementy ve verzi HTML 5.0

Tag `<section>`

Tag `<section>` definuje hlavní dokument anebo část přílohy. Slouží jako logický kontejner, který spojuje obsah ve smyslu. Lze ho používat pro sekci novin, kapitol, kontaktní informaci. Pokud bychom odstranili jednu z těchto částí, tak tím obsah webové stránky nebude narušen. Tento tag se používá stejně tak, kdybychom rozdělili knihu na kapitoly. Tag `<section>` je velmi obecným elementem, a proto vždy je potřeba zvážit, jestli byste mohli použít nějaký specifitější element. [1]

Příklad použití:

```
<section>
  <h1>Zde bude nadpis (úroveň 1 až 6)</h1>
  <p>Obsah</p>
</section>
```

Tag <article>

Tag <article> je speciálním druhem tagu <section>, který představuje samostatnou část stránky. Je vhodný v takové situaci, kdy je těch článků na webové stránce více. Mohou to být např. noviny, zprávy fóra, časopisy, komentáře, příspěvky z blogu atd. Místo tagu <article> lze použít tag <section>, ale <article> přidává více sémantický význam. Daný tag také může obsahovat jak hlavičku, tak i patičku, ale není to povinné. Do tagu <article> lze vnořovat další tagy jako např. <article>, <section> apod. [1]

Příklad použití:

```
<section>
  <header>
    <h2>Nadpis</h2>
  <header>
    <p>Text článku nebo komentáře</p>
</section>
```

Tag <nav>

Tag <nav> reprezentuje část webové stránky, která odkazuje na jiné webové stránky anebo části této stránky: v podstatě je oddílem s navigačními odkazy nebo menu. Ne všechny skupiny odkazů na webové stránce by se měly umísťovat do tagu <nav> – pouze oddíly, které se skládají z bloků hlavní navigace, jsou ideální pro využití tohoto tagu. Vztahuje se to také i na spodní část stránky, která často obsahuje stručný seznam odkazů na různé webové stránky jako úvodní stránka, podmínky používání, kontakty, pro média apod. V tomto případě není nutné používat tag <nav>, stačí použít tag <footer>. Také nelze vkládat tag <nav> do tagu <address>. [2]

Příklad použití:

```
<nav>
  <a href="http://www.w3schools.com/html/">HTML</a> |
```

```
<a href="http://www.w3schools.com/css/">CSS</a> |  
<a href="http://www.w3schools.com/js/">JAVASCRIPT</a>  
</nav>
```

Tag <aside>

Tag <aside> obsahuje obsah, jako jsou boční panely, vložení nebo stručné vysvětlení, které nepřímo souvisí s okolím obsahu. Například při použití uvnitř tagu <article>, tag <aside> může určit obsah související s autorem článku. Můžeme si představit tento daný jako prvek, který se zobrazí vlevo či vpravo na stránce. Nicméně je třeba brát v úvahu, že všechny strukturní tagy včetně <aside> jsou blokové a zobrazují se na novém řádku, také zabírají celou dostupnou šířku stránky nebo položky, do které jsou vloženy. Daný tag lze využít pro výstup citací, reklamy, bočních panelů, navigace přes <nav> a dalšího obsahu, který je považován za oddělený od hlavního obsahu stránky. [2]

Příklad použití:

```
<aside>  
  <nav>  
    <h3> Můj krásný blog</h3>  
    <ul>  
      <li><a href="/hudba/"> Hudba </a></li>  
      <li><a href="/fotografie/"> Fotografie </a></li>  
      <li><a href="/hobby/"> Hobby </a></li>  
    </ul>  
  </nav>  
</aside>
```

Tag <header>

Tag <header> představuje skupinu úvodních nebo navigačních prostředků. Daný tag obvykle obsahuje záhlaví oddílu (tag <h1> až <h6>, anebo <hgroup>), ale není to povinné. Z názvu plyne, že se termín "header" vztahuje k hlavičce webové stránky, kde obvykle umístěn název a logo webu, menu, formulář pro vyhledávání atd. Když se tento tag vztahuje na libovolný element, který rozděluje stránku na části, tak sémanticky znamená titulek či úvod pro obsah svého nadřazeného elementu. [2]

Příklad použití:

```
<article>
  <header>
    <h2> Název článku </h2>
  </header>
  <p> Text článku </p>
</article>
```

Tag <footer>

Tag <footer> definuje zápatí pro celou webovou stránku anebo pro určitý oddíl. Obvykle obsahuje další informace o obsahu svého nadřazeného elementu, např. jméno a příjmení autora, kontaktní údaje, podmínky používání, další navigaci apod. Termín "footer" zahrnuje podkapitolu, která se nachází v dolní části svého nadřazeného elementu, ale nemusí to být poslední dceřiný element svého nadřazeného. Jeho sémantickým významem je doplňující informace k oddílu, ale odkaz na místo umístění. [2]

Příklad použití:

```
<footer>
  <p> Autor: Dmitry Arkhipenko </p>
  <p> E-mail: <a href="mailto:login@abc.com"> login@abc.com</a></p>
</footer>
```

Tag <time>

Tag <time> unožňuje zadat čas v 24 hodinovém formátu a datum. Daný tag má atribut "datetime", ve kterém se zadává čas a datum ve strojově čitelném formátu a pak se to zobrazí v už námi známé podobě. To může být užitečné pro uspořádání událostí v kalendáři uživatele anebo to může pomoci vyhledávači přesněji provést výsledky vyhledávání. [1]

Příklad použití:

```
<article>
  <p> Čas publikace článku: <time> 10:00 </time> </p>
  <time datetime="2015-09-30"> 30. září 2015 </time>
  <time datetime="2007-12-17 17:00+01:00"> 17. prosince 2007 17:00
  UTC+1 hodina </time>
</article>
```


Tag <mark>

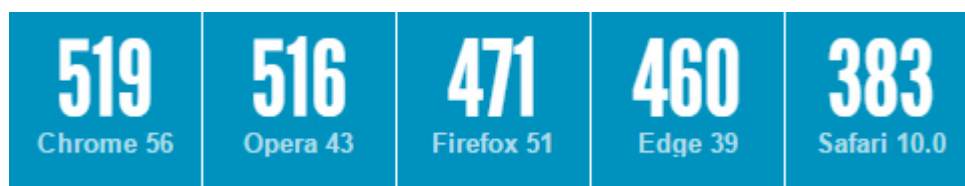
Tag <mark> se používá pro zvýraznění textu z důvodu jeho významu v daném kontextu. Například by to mohlo být zvýraznění slova v textu, které uživatel hledal. Tag <mark> nenes v sobě žádný logický kontext na rozdíl od tagů nebo , ale je více významově smysluplnější než tag . Jeho přímým cílem je věnovat pozornost na to, co je relevantní v daném okamžiku pro uživatele. [2]

Příklad použití:

```
<article>
  <header>
    <h2> HTML - co to je? </h2>
  </header>
  <p><mark>HTML</mark> - je HyperText Markup Language </p>
</article>
```

3.8 Podpora HTML5 webovými prohlížeči

Pro zjištění podpory HTML5 webovými prohlížeči byl proveden test na webové stránce "html5test.com". Daný test hodnotí webové prohlížeče podle takových vlastností jako jsou sémantika, animace, multimédia, responzivní obrázky a další. Výsledek se skládá ze součtu všech bodů za jednotlivé vlastnosti. Maximální skóre je stanoveno ve výši 555 bodů.



Obrázek 2 - Test TOP 5 prohlížečů

Přiložené výsledky uvádí dosažená skóre vybraných prohlížečů aktuálních verzí v okamžiku provádění testu. Nejvyšší počet bodů získal prohlížeč "Google Chrome" a těsně vedle je prohlížeč "Opera", který získal o 3 body méně. Nejhorší výsledek prokázal prohlížeč "Safari".

3.9 Porovnání HTML 4.01 a HTML5

Pro porovnání dvou verzí je potřeba mít představu o tom, jak vypadá struktura zápisu dokumentu. Proto bylo rozhodnuto uvést zápis struktury dokumentu obou verzí a pomocí něj provádět porovnání.

Každý HTML soubor na začátku zdrojového kódu by měl mít uvedenou deklaraci typu dokumentu, kterou využívá webový prohlížeč pro identifikaci. V nové verzi HTML5 se tento zápis výrazně změnil. Následující tabulka (Tab.2) představuje zápis deklarace v obou verzích a je vidět, že zápis ve verzi HTML5 je mnohem jednodušší a snadno zapamatovatelný pro vývojáře.

Tabulka 2 – Deklarace typu dokumentu

HTML 4.01	HTML5
<pre><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"></pre>	<pre><!DOCTYPE html></pre>

Další výraznou změnou je zápis tagu <meta>. Tento tag se používá pro uchování informací webovými prohlížeči a vyhledávacími systémy. Například vyhledávací systém používá metadata pro zjištění obsahu dokumentu, klíčových slov a dalších dat. V následující tabulce (Tab.3) je vidět, že zápis se ve dvou verzích liší a ve verzi HTML5 je kratší.

Tabulka 3 - Metadata

HTML 4.01	HTML5
<pre><meta http-equiv="Content-Type" content="text/html; charset=UTF-8"></pre>	<pre><meta charset="UTF-8"></pre>

Dále je potřeba srovnat zápis obsahu dokumentu, který je umístěn do tagu <body>. V tomto tagu je umístěno tělo dokumentu, to, co se zobrazuje webovým prohlížečem na obrazovce. Přehled této struktury je uveden v následující tabulce (Tab.4). Verze HTML 4.01 využívá tag <div> pro umístění veškerého obsahu dokumentu a pomocí atributu id a class se provádí identifikace tohoto obsahu. Například id="header" určuje, že se jedná o hlavičku webové stránky, do id="nav" je umístěna navigace, id="article" obsahuje

název a text článku, následně `id="section"` obsahuje fotografii k tomuto článku a nakonec `id="footer"`, který uvádí autorská práva.

Tabulka 4 - Tělo dokumentu

HTML 4.01	HTML5
<pre><div id="header"> <h1>Název webu</h1> </div></pre>	<pre><header> <h1>Název webu</h1> </header></pre>
<pre><div id="nav"> Domů Lidé Kontakt </div></pre>	<pre><nav> Domů Lidé Kontakt </nav></pre>
<pre><div id="article"> <h2>Název článku</h2> <p>Popis ...</p> Odkaz <div id="section"> <h3>Foto</h3> </div> </div></pre>	<pre><article> <h2>Název článku</h2> <p>Popis ...</p> Odkaz <section> <h3>Foto</h3> </section> </article></pre>
<pre><div id="footer"> <p>Copyright © 2016</p> </div></pre>	<pre><footer> <p>Copyright © 2016</p> </footer></pre>

Po prohlížení zápisu struktury těla dokumentu ve dvou verzích je dobře vidět, že ve verzi HTML 4.01 je nutností upřesňování, k čemu konkrétní element slouží, aby bylo možné rozpoznat, jestli je to hlavička anebo zápatí. To může činit problémy pro vyhledávací roboty, které procházejí obsahem dokumentu, protože tag `<div>` nenesou v sobě sémantický význam, a proto vývojář by měl vždy jednoznačně identifikovat typ elementu. Nová verze HTML5 řeší tento problém tak, že byly zavedeny speciální tagy, které rozdělují na jednotlivé části dokument a tím nesou sémantický význam pro vyhledávací roboty. Zdrojový kód dokumentu, který je napsán pomocí HTML5, je přehlednější a snadno orientovatelný.

3.10 HTML5 audio a video

Audio

HTML5 audio poskytuje vylepšené možnosti práce s audio soubory. Až do nedávné doby, jediný způsob, jak přidat zvukové soubory na webové stránky byla integrace na pozadí zvuku pomocí tagu `<bgsound>`, který se přehrával při prohlížení uživatelem stránky bez možnosti vypnutí. Díky tomu, že do specifikaci HTML5 byl zahrnut nový tag `<audio>`, se objevila možnost přidat audio soubory se zabudovaným přehrávačem bez použití doplňujících pluginů.

V současné době neexistuje audio formát, který by fungoval ve všech prohlížečích, proto se pro zajištění dostupnosti obsahu co nejširšímu publiku doporučuje zahrnout více zdrojů zvuku, prezentované pomocí atributu `src` tagu `<source>`. Současně je možné přidat záložní obsah pro prohlížeče, které nepodporují tag `<audio>`.

```
<audio controls>
```

```
  <source src="music.ogg" type="audio/ogg">
```

```
  <source src="music.mp3" type="audio/mpeg">
```

Váš prohlížeč nepodporuje přehrávání zvukových souborů, daný soubor si můžete stáhnout [zde](music.mp3).

```
</audio>
```



Obrázek 3 - Vzhled audio přehrávače v prohlížeči "Google Chrome"

Atributy:

- **autoplay** – automatické přehrávání audio souboru ihned po načtení stránky;

```
<audio autoplay></audio>
```

- **controls** – zobrazuje ovládací prvky přehrávání, včetně: spuštění, pozastavení, posuvníku a hlasitost zvuku. Ve výchozím nastavení při absenci atributu ovládací panel zvukového souboru nebude zobrazena;


```
<audio controls></audio>
```
- **loop** – opakované přehrávání zvukového souboru po jeho spuštění;


```
<audio loop></audio>
```
- **muted** – vypnutí zvuku při přehrávání zvukového souboru;


```
<audio muted></audio>
```
- **preload** – určuje, jaké akce musí provést prohlížeč, pokud v kódu není specifikován atribut **autoplay**. Pokud atribut **autoplay** byl uveden, atribut **preload** bude ignorován. Daný atribut může mít jednu z následujících tří hodnot:
 - *auto* – nahrávání zvukového souboru začne hned po načtení stránky;
 - *metadata* – sběr informací o zvukovém souboru: velikost, seznam skladeb, délka;
 - *none* – nahrávání zvukového souboru začne až po kliknutí na tlačítko přehrávání;


```
<audio preload="auto"></audio>
```

```
<audio preload="metadata"></audio>
```

```
<audio preload="none"></audio>
```
- **src** – obsahuje absolutní nebo relativní URL adresu zvukového souboru.


```
<audio src="music.mp3"></audio>
```

Video

Dříve, když bylo potřeba přidat video na webovou stránku, používal se tag `<object>`, který sloužil univerzálním obalem. Druhý přístup se spočíval v použití pluginu prohlížeče, jako je například Flash nebo Silverlight. I když technologie Flash umožňovala použít předdefinovaný video přehrávač nebo vytvořit vlastní, ale její použití způsobilo velké hromadění HTML kódu a video soubory bylo nutné kódovat na požadovaný formát.

HTML5 video je novým standardem pro vysílání multimediálních souborů v síti s původním programovým rozhraním bez použití doplňkových modulů. Pomocí tagu `<video>` se objevila možnost přidat video soubory na webové stránky a stylizovat vzhled video přehrávače pomocí CSS stylů.

Všechny prohlížeče nepodporují stejný formát videa, proto je nutné poskytnout video soubor do několika formátů najednou, pro tento účel se používá tag `<source>`, který musí být umístěn uvnitř tagu `<video>`. Jakýkoli text uvnitř tagu `<video>` bude zobrazen prohlížečem v tom případě, když tag sám nebo formát použitého videa není podporován prohlížečem.

Tag `<video>` má atributy, které mohou být použity pro různé účely. Část z nich je stejná jako u tagu `<audio>` (`controls`, `autoplay`, `loop`, `muted`, `preload`, `src`), proto budou uvedeny jenom ostatní atributy:

- **height** – nastavuje výšku okna pro zobrazení videa, hodnota musí být uvedena v pixelech a bez použití "px";

```
<video height="320"></video>
```

- **width** – nastavuje šířku okna pro zobrazení videa, hodnota musí být uvedena v pixelech a bez použití "px";

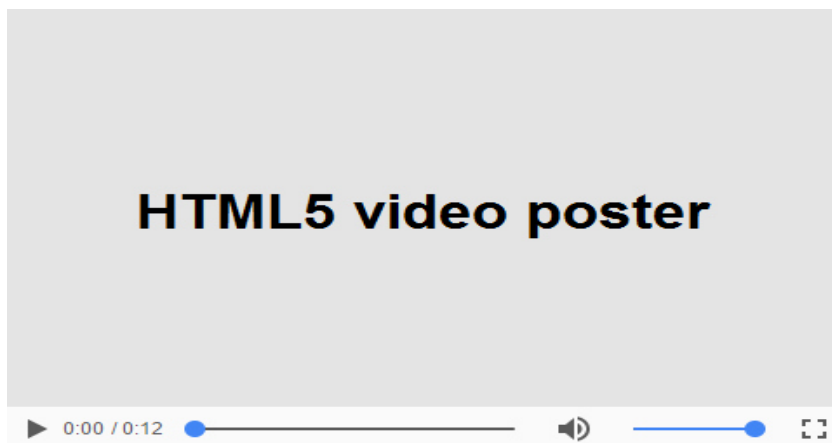
```
<video width="480"></video>
```

- **poster** – určuje URL adresu obrázku, který bude zobrazen až do úplného nahrání souboru nebo dokud uživatel klikne na tlačítko pro přehrávání. Pokud atribut není nastaven, pak prohlížeč zobrazí první snímek videa.

```
<video poster="logo_video.jpg"></video>
```

Příklad použití tagu `<video>` pro zobrazení videa na webové stránce:

```
<video width="480" height="320" poster="logo_video.jpg" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  <source src="movie.webm" type="video/webm">Váš prohlížeč
  nepodporuje přehrávání video souborů, daný soubor si můžete
  stáhnout
  <a href="movie.mp4">zde</a>.
</video>
```



Obrázek 4 - Vzhled video přehrávače v prohlížeči "Google Chrome"

3.11 HTML formuláře

Tag `<form>`

Tag `<form>` stanovuje formulář na webové stránce. Formulář je určen pro výměnu dat mezi uživatelem a serverem. Oblast použití formulářů není omezena odesláním dat na server, pomocí klientských skriptů je možné přistupovat k libovolnému prvku formuláře, upravovat ho a aplikovat podle vlastního uvážení. Dokument může obsahovat libovolný počet formulářů, ale současně se na server může být odeslán pouze jeden formulář. Z tohoto důvodu údaje z formulářů musí být navzájem nezávislé.

Pro odeslání formuláře na server slouží tlačítko "Submit", stejně tak toho lze dosáhnout stisknutím klávesy "Enter" v rámci formuláře. Pokud je tlačítko "Submit" chybí ve formě, klávesa "Enter" napodobuje jeho použití. Ve chvíli kdy je formulář odeslán na server, správa dat je předána programu, který nastaven atributem "action" tagu `<form>`. Předem prohlížeč připravuje informace v podobě dvojice "název=hodnota", kde název je určen atributem "name" tagu `<input>`, a hodnota je zadána uživatelem nebo nastavena výchozí v poli formuláře. Je povoleno uvnitř tagu `<form>` vkládat jiné tagy, přičemž samotná forma není zobrazena na webové stránce, jsou viditelné pouze její prvky a výsledky vnořených tagů.

Atributy:

- **action** – adresa dokumentu nebo skriptu, který zpracovává data z formuláře;
- **enctype** – způsob kódování dat formuláře (hodnota pro zachování českých znaků: multipart/form-data);
- **method** – metoda protokolu HTTP (hodnoty: GET nebo POST);

- **name** – název formuláře;
- **target** – okno nebo rámeček, kam bude vrácen výsledek (výchozí hodnota: `_self`).

Tag `<input>`

Tag `<input>` je jedním z nejpoužívanějších elementů formuláře a umožňuje vytvářet různé prvky uživatelského rozhraní a zajistit interakci s uživatelem. Hlavně `<input>` je určen pro vytváření textových polí, různých tlačítek, přepínačů a zatrhvacích políček. I když tag `<input>` není nutné umísťovat uvnitř tagu `<form>`, který definuje formulář, ale pokud zadané údaje by měly být odeslány na server, kde budou zpracovány, pak je tag `<form>` povinným.

Hlavní atribut tagu `<input>`, který určuje jeho typ – "`type`". Tento atribut umožňuje nastavit následující položky formuláře: textové pole (`text`), pole s heslem (`password`), přepínače (`radio`), poštovní schránku (`email`), telefonní číslo (`tel`), webovou adresu (`url`), zatrhvací políčko (`checkbox`), skryté pole (`hidden`), tlačítko (`button`), tlačítko pro odesílání formuláře (`submit`), tlačítko pro vymazání formuláře (`reset`), pole pro odesílání souboru (`file`), tlačítko s obrázkem (`image`).

Atributy:

- **maxlength** – maximální počet znaků povolených v textu;
- **name** – jméno pole formuláře pro identifikaci;
- **pattern** – šablona vstupu, konkretizuje povolené znaky ke vstupu;
- **placeholder** – nápověda v poli, která hned zmizí při kliknutí;
- **required** – označuje, že toto pole je povinné;
- **size** – je šířka textového pole;
- **value** – hodnota.

3.12 HTML5 vs Adobe Flash

HTML5 a Adobe Flash jsou dvěma technologiemi, které se v poslední době neustále srovnávají. Nová specifikace HTML5 podporuje umístění audia a videa na webové stránce. Do té doby, než byla aktualizovaná specifikace HTML5, mnoho let Adobe Flash byl jediným způsobem pro umístění videa na webových stránkách. Dnes HTML5 je lepší než Flash minimálně ve dvou věcech: použitelností v mobilním prostředí a existence sémantiky. Růst používání mobilních zařízení a rozšíření uplatnění interaktivních videí v oblastech zábavy a

nákupů dokonale ladí s otevřenou strukturou HTML5. Po aktualizaci HTML5, jednou z hlavních výhod nového jazyka byla považována schopnost pracovat na mobilních zařízeních. Realita je taková, že některé Android a iOS zařízení nepodporují Flash. Majitelů mobilních zařízení s přístupem na internet dnes je docela hodně, ale ne všichni z nich mají moderní chytré telefony od topových výrobců, a proto společnosti, vytvářející interaktivní videa, které využívají technologie Flash, automaticky ztrácejí tuto skupinu, která by mohla prohlížet tento obsah na svých zařízeních.

Dnes zástupci různých oblastí od reklamy do b2b služeb věnují více pozornosti mobilnímu publiku. Byly takové situace, kdy se uživatelé rozhodli koupit si něco na e-shopu Amazon přes iPad, ale nebyli moci si prohlédnout video, které popisovalo určitý produkt, z důvodu absence Flash-pluginu. Dá se říci, že část z těchto uživatelů kvůli tomu nedokončili nákup zboží a opustili webovou stránku. Pro společnost je zde zřejmé promarněná příležitost.

Uplatnění v mobilním prostředí není jedinou výhodou HTML5 nad Flash. Použití struktury nového standardu umožňuje vyhledávačům snadněji procházet vytvořené aplikace, což nelze říci o Flash. Pomocí HTML5 vývojáři mohou vytvářet projekty, skládající se z mnoha vzájemně propojených částí, a nyní vyhledávače mohou porozumět těmto souvislostem. Projekty, vytvořené pomocí Flash, jsou pro roboty Google černou skříňkou s matoucím obsahem.

HTML5 je nová technologie v porovnání s Flash. Vývojáři HTML5 neustále pracují nad zlepšováním produktu a možná v blízké budoucnosti HTML5 bude vedoucím hráčem v této oblasti.

3.13 Analýza vybraného webu

Již uplynulo pár let od data, kdy byla dokončena specifikace HTML5, a myšlenkou bylo podívat se na strukturu nově vytvořených webových stránek, jelikož starší webové stránky dosud ne zcela mají použité nové elementy. Pro analýzu byla vybrána následující webová stránka "html5accessibility.com".

Zprv je nutné otevřít zdrojový kód vybraného webu, a to lze udělat přímo v prohlížeči na daném webu. Kliknutím pravého tlačítka myši je nutné zobrazit zdrojový kód této stránky. Zdrojový kód se otevře v novém okně a je k dispozici pro čtení.

Po přečtení zdrojového kódu bylo zjištěno, že dokument byl plně vytvořen pomocí verze HTML5. Dokument má správnou strukturu a využívá sémantické značky.

Pro ověření správnosti kódu dokumentu vybraného webu bylo rozhodnuto provést validaci zdrojového kódu pomocí online nástroje, který poskytuje web konsorcium "W3C" a to na adrese "validator.w3.org". Po validaci bylo zjištěno, že kód je správný a validní, nástrojem nebyly nalezeny žádné chyby a varování.

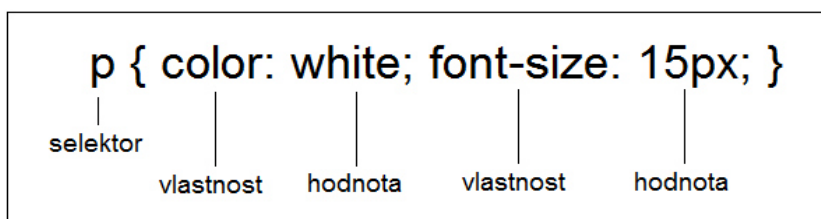
Posledním krokem byla kontrola na responzivní design, jestli ten web je použitelný i v případě, když se na něj dostanou z jiných zařízení než počítač nebo notebook. Pomocí použití mobilu a tabletu s displeji 5.5 a 9.7 palců byl otevřen daný web a celý obsah byl zobrazen správně v rámci šířky zařízení a tím nevznikly žádné rolovací lišty.

3.14 CSS

CSS (Cascading Style Sheets), česky "kaskádové styly", je kolekce metod, která slouží pro grafickou úpravu obsahu webových stránek. Obvykle CSS se používá ke změně stylu prvků webové stránky a uživatelského rozhraní, které jsou napsané v jazycích HTML a XHTML, ale také mohou být použity pro jakýkoliv druh dokumentu XML. Až textové dokumenty HTML jsou zobrazeny na obrazovce, CSS modeluje dokument ve formě hierarchie kontejnerů, které obsahují slova, řádky, odstavce, tabulky atd., z nichž každý má vlastnosti, jako je velikost, barva, písmo a další.

3.14.1 Syntaxe

Syntaxe CSS se skládá ze dvou částí: prvek webové stránky – selektor, a příkaz pro formátování tohoto prvku. Selektor sděluje prohlížeči, jaký přesně prvek má formátovat a následně zpracovává kód uvnitř složených závorek, který obsahuje vlastnosti a jejich hodnoty. Následující obrázek znázorňuje, jak vypadá syntaxe CSS příkazu.



Obrázek 5 - Syntaxe CSS

Výsledek příkazu z výše uvedeného obrázku zobrazí text, který je umístěn v tagu `<p>`, s velikostí písma 15 bodů a bílou barvou.

Selektory ID a CLASS

Kromě typu selektoru v CSS existuje možnost výběru konkrétního prvku bez ohledu na jeho typ. K tomu lze využít buď identifikátor `id`, nebo třídu `class`.

Selektor `id` je určen pro použití stylu na unikátní prvky webové stránky, znamená to, že tento prvek se používá na stránce jenom jednou, v takové roli mohou být například hlavička, zápatí, navigační menu. Pro použití selektoru `id` je nutné zadat unikátní identifikátor v atributu HTML tagu, potom v popisu stylu zapsat daný selektor pomocí `#` a hned po něm název identifikátoru. [2]

Selektor `class` umožňuje také jako selektor `id` stylizovat konkrétní prvek stránky, ale na rozdíl od `id`, selektor `class` umožňuje aplikovat svůj styl na několika prvky webové stránky. Pro použití selektoru `class` stejně jako pro selektor `id` je nutné určit, k jakému prvku se tento selektor bude aplikovat. Je třeba přidat do HTML tagu atribut `class` a zadat název této třídy, potom již lze zapsat ve stylopisu tento selektor pomocí tečky a hned po ní název třídy. [2]

Pravidla @media

Funkce `@media`, která byla zavedena do třetí verze CSS, umožňuje použít speciální styly pro jednotlivé výstupní zařízení. Při zpracovávání HTML dokumentu prohlížeč určuje typ nosiče informace. Pokud je zadaný typ nosiče odpovídá typu nosiče, na kterém běží prohlížeč, v tomto případě budou použity styly pro konkrétní prvky, které jsou uvedeny v `@media`.

Pravidla `@media` kontrolují takové vlastnosti, jako je šířka a výška okna prohlížeče, šířka a výška obrazovky zařízení, orientace tabletu nebo telefonu (na výšku či šířku) a mnoho dalších, ale nejčastěji se kontrolují šířka zobrazovací plochy a šířka obrazovky přístroje. Zápis vypadá následujícím způsobem:

```
@media (min-width: 768px) and (max-width: 992px) {  
    p { font-size: 12px; }  
}
```

3.14.2 Způsoby deklarování kaskádových stylů

Externí stylopis

Externí stylopis představuje textový soubor s příponou `.css`, ve kterém se nachází styly prvků. Soubor se vytváří v editoru kódu stejně jako HTML stránka. Uvnitř souboru mohou být pouze styly, nesmí tam být HTML kód. Externí stylopis lze připojit k webové stránce pomocí nepárové značky `<link>`, která se nachází uvnitř sekce `head`. V tagu `<link>` je nutné uvést atributy `rel` a `href`.

```
<head>
  <link rel="stylesheet" href="css/style.css">
</head>
```

Interní stylopis

Interní styly jsou vloženy do sekce head HTML dokumentu a definovány uvnitř párového tagu `<style>`. Interní styly mají přednost před externími. Tento způsob formátování obsahu není vhodný pro použití ve vícestránkovém dokumentu.

```
<head>
  <style>
    p {font-size: 15px; font-family: Georgia, Serif;}
  </style>
</head>
```

Přímý stylopis

Přímý stylopis se zapisuje pomocí atributu `style` do tagu, který je třeba stylizovat. Do atributu se zapisují vlastnosti a jejich hodnoty. Takový způsob se používá jenom ve výjimečných případech, kdy je nutné určit styl pouze pro daný prvek.

```
<p style="font-weight: bold; color: green;">Ahoj</p>
```

3.14.3 Bootstrap

Bootstrap neboli "Twitter Bootstrap" je HTML/CSS framework (aplikační rámec) pro vytváření webových stránek. Jinými slovy, je to sada nástrojů pro vývoj webu. Využití daného frameworku může ušetřit spoustu času a úsilí vývojářům při navrhování webových stránek. Hlavními nástroji Bootstrapu jsou grid (česky "mřížkový") systém, šablony, responzivní obrázky, tabulky, tlačítka, formuláře, navigační panely, ikonky a mnoha dalších užitečných věcí. Všechny tyto funkce také pomáhají udělat webové stránky responzivními pro různé typy zařízení, což je jednou z výhod, kdy cílem je dosáhnout co největšího počtu návštěvníků nebo zákazníků.

Mřížkový systém daného frameworku je silným nástrojem pro vývoj responzivních webových stránek. Pomocí prefixů můžeme indikovat, jak je třeba změnit zobrazení bloků v závislosti na typu zařízení, na kterém se zobrazí webová stránka. Bootstrap rozděluje šířku nadřazeného bloku na 12 stejných částí, které vývojář může použít, jak je potřeba. Vytvoření takového kontejneru se provádí tak, že nejdříve je nutné vytvořit tag `<div`

`class="container">` a do něj vnořit tag `<div class="row">`, dále již lze nastavit rozdělení bloků dle potřeb pomocí prefixů, které je nutné zapsat do atributu `class`.

Prefixy, které jsou k dispozici pro použití:

- **col-xs-2** – je určen pro malé typy zařízení, jako je mobil (šířka $<768\text{px}$);
- **col-sm-3** – je určen pro typy zařízení, jako je tablet (šířka $\geq 768\text{px}$);
- **col-md-6** – je určen pro střední typy zařízení, např. notebook (šířka $\geq 992\text{px}$);
- **col-lg-6** – je určen pro velké typy zařízení, např. počítač (šířka $\geq 1200\text{px}$).

Příklad použití:

```
<div class="container">
  <div class="row">
    <div class="col-md-3"></div>
    <div class="col-md-6"></div>
    <div class="col-md-3"></div>
  </div>
</div>
```

4 Vlastní práce

Předchozí teoretická část této práce se věnovala novinkám HTML5 a porovnání dvou verzí HTML 4.01 a HTML5. Vlastní práce je zaměřena na použití nových možností značkovacího jazyka HTML5. Nejprve je vytvořena demonstrativní webová stránka a dále je provedeno testování zobrazení webové stránky v různých prohlížečích a na různých typech zařízení.

4.1 Tvorba webové stránky

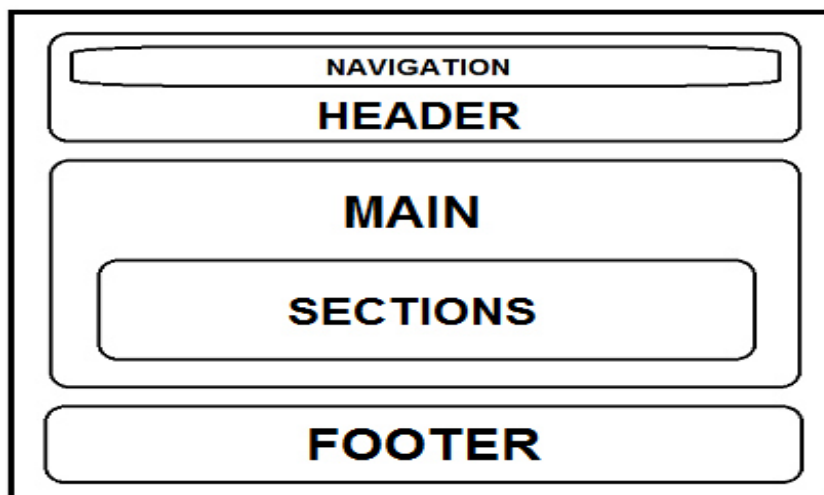
Pro demonstraci nových možností HTML5 bylo rozhodnuto vytvořit jednostránkový web fiktivní firmy, která nabízí služby vývoje webových stránek, blogů a elektronických obchodů.

Nejprve pro vytvoření webové stránky je nutné nainstalovat na počítač programátorský editor. Z vlastních zkušeností byl zvolen editor "Sublime Text 3", který je možné stáhnout zdarma na adrese "sublimetext.com/3" a také je dostupný pro různé platformy operačních systémů.

Aby webová stránka byla dostupná k prohlížení nejen lokálně nebo při nahrávání ze souboru, je potřeba umístit zdrojový kód na webhosting (česky "webový prostor") a zakoupit doménové jméno. Podle vlastního názoru byl vybrán webhosting, který poskytuje společnost "000webhost" a je k dispozici na adrese "000webhost.com", a také této společnosti bylo rezervováno doménové jméno třetího řádu. Finální adresa, na kterém je umístěna webová stránka, je "rwd.host56.com".

4.1.1 Struktura

Webová stránka je určena pro firmu, která nabízí služby v oblasti vývoje webu, proto bylo zvoleno navrhnout takovou strukturu, která by byla pro návštěvníky a potenciální zákazníky jednoduchá a přehledná. Prototyp struktury slouží jako ukázka, jak výsledný web by měl vypadat. Daný návrh slouží jako dobrá pomůcka při vytváření HTML struktury webové stránky.



Obrázek 6 – Struktura stránky

Struktura se skládá ze tří hlavních částí: hlavička (header), hlavní obsah (main) a zápatí (footer). Navigace je umístěna nahoře zobrazovací plochy, hlavní obsah zahrnuje v sobě dílčí sekce, které souvisí se službami firmy, a zápatí obsahuje kontaktní informace a odběr novinek.

4.1.2 Kódování stránky

Prototyp webové stránky je hotový a již je možné přistoupit ke kódování HTML struktury. Každá správně navržená webová stránka by měla začínat deklarací typu dokumentu. Pak následuje tag `<html>`, ve kterém bude umístěn celý obsah dokumentu včetně doplňujících skriptů. Dalším tagem musí být zapsán `<head>`, který zahrnuje v sobě tagy `<meta>`, `<title>` a `<link>`. Metadata používají vyhledávací roboty pro zjištění obsahu webu a klíčových slov, do tagu `<title>` se zapisuje název webové stránky, který bude zobrazen v horní části webového prohlížeče, pomocí tagu `<link>` se provádí připojení kaskádových stylů z externího souboru.

Další částí je návrh struktury dokumentu. Pro zapsání hlavního obsahu je třeba se podívat do prototypu, který byl vytvořen pro přehlednost. Poté již je možné naplnit dokument textem a další potřebnou informací. Pro znázornění je uveden zdrojový kód sémantické struktury dokumentu.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
```

```

<meta name="viewport" content="width=device-width, initial-
scale=1">
<title>...</title>
<link rel="stylesheet" href="css/bootstrap.min.css">
<link rel="stylesheet" href="css/style.css">
</head>
<body>
  <header>...</header>
  <nav>...</nav>
  <main>
    <section id="services">...</section>
    <section id="pricing">...</section>
    <section id="team">...</section>
    <section id="clients">...</section>
    <section id="contact">...</section>
  </main>
  <footer>...</footer>
</body>
</html>

```

Uvedená sémantická struktura dokumentu dobře znázorňuje, že kód, napsaný pomocí nových značek HTML5, je mnohem přehlednější a lze se v něm lehce orientovat. Kdyby struktura dokumentu byla zapsána pomocí tagu `<div>` s příslušnou identifikací každého prvku a jednotlivých sekcí v části "main", zdrojový kód by byl pro vývojáře těžce čitelný a také by obsahoval větší počet znaků.

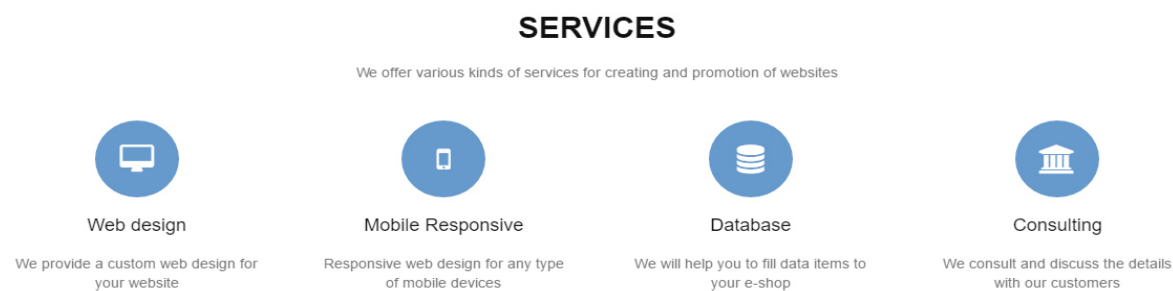
4.1.3 Grafická část

Zápis pouze HTML dokumentu nestáčí pro vytvoření webové stránky a je potřeba přidat také grafickou část. Ve zdrojovém kódu jsou uvedeny odkazy pomocí tagu `<link>` na externí soubory kaskádových stylů, které slouží pro grafické formátování dokumentu.

Pro formátování stylu dokumentu je také využít framework "Bootstrap", které nabízí šablony kaskádových stylů dílčích prvků. Nejprve je třeba stáhnout z oficiální webové

stránky "getbootstrap.com" balíček, který obsahuje soubory s již definovanými styly. Dále už lze přistoupit k formátování dokumentu.

Pro rozdělení obrazovky do více sloupců je využít mřížkový systém. Dělení se provádí pomocí postupně do sebe vnořených tříd ".container", ".row" a prefix ".col-md-3" v případě rozdělení obrazovky do čtyř sloupců. Takový systém je použit v každé sekci vytvořeného dokumentu. Příklad sekce "services" (služby) demonstruje, jak vypadá takové rozdělení.



Obrázek 7 - Rozdělení do čtyř sloupců

Aby se sloupce postupně přenášely dolů při menší šířce obrazovky a tím tvořily dobrou přehlednost, v souboru `style.css` je nutné definovat pomocí funkce `@media` konkrétní šířku obrazovky zařízení, určit prvek pomocí selektoru a dále nastavit styl zobrazení. Následující příklad znázorňuje použití dané funkce.

```
@media(max-width:992px) {  
  #services .col-md-3 {  
    display: inline-block;  
    max-width: 235px;  
  }  
}
```

Navigační panel je vytvořen pomocí šablony, která v případě nedostatečné šířky, pokud je webová stránka otevřená na takových zařízeních jako jsou mobily a tablety, zobrazí kompaktnější navigaci a lze ji rozbalit kliknutím myši na tlačítko rámečku. Po rozbalení se zobrazí všechny odkazy na jednotlivé části stránky. Při otevření stránky v prohlížeči tabletu nebo mobilu navigace vypadá následovně.



Obrázek 8 - Rozbalovací navigační panel

V sekci "team" (tým) jsou umístěny fotografie jednotlivých členů týmu. Pro atraktivní zobrazení fotografií je použita třída "img-circle" v tagu , která definuje, aby se vložená fotografie zobrazovala ve formě kruhu. Na obrázku (Obr.9) je vzhled takového formátování fotografií.

MEET OUR TEAM

Right here you can see our good team, each one has professional skills and extensive experience.



John Taylor



Andrea Neuer



Kevin Johnson



Megan Cutler

Obrázek 9 - Fotografie ve formě kruhu

Sekce "clients" (klienti) zahrnuje v sobě logotypy zákazníků, které využily služby dané firmy. Při otevření dané sekce logotypy jsou zatemněny, až po najetí myši jednotlivý logotyp bude zasvícen. Takový efekt se dělá pomocí nových možností kaskádových stylů, a to je pomocí pseudotřídy ":hover" a vlastností "opacity". Vlastnost "opacity" umožňuje zatemnit vybrané prvky ve standardním zobrazení a pak pseudotřída ":hover" definuje novou vlastnost zatemnění, když se provede najetí myši na dané prvky. Najetí myši je provedeno na první zleva logotyp a vypadá takový efekt následovně.



Obrázek 10 - Efekt zasvícení obrázku

```
#clients img {  
    opacity: 0.5;  
}  
#clients img:hover {  
    opacity: 1;  
}
```

Kontaktní formulář webové stránky se skládá z polí "Full Name" (Jméno a Příjmení), "Email Address" (Emailová Adresa), "Phone Number" (Telefonní Číslo) a "Message" (Zpráva). Každé pole má nápovědu, co v něm konkrétně by měl návštěvník webové stránky vyplnit. Dělá se to pomocí nového atributu "placeholder", který byl zaveden do specifikace HTML5. Daný atribut se využívá ve formulářových polích jako nápověda pro návštěvníka webové stránky, jaké informace by měl v jednotlivých polích zapsat. V okamžiku kliknutí na pole jsou podsvíceny okraje daného pole pro pohodlí, aby návštěvník si byl jistý, že vybral to pole, které potřeboval, a také je zobrazena nápověda, jakmile bude zapsán alespoň jeden symbol, nápověda zmizí. Následující ukázka znázorňuje, jak vypadají pole do (Obr.11) a po (Obr.12) vyplnění.

The image shows three input fields in a contact form. Each field has a small icon on the left and placeholder text. The first field has a person icon and the text 'Full Name'. The second field has an envelope icon and the text 'Email Address'. The third field has a telephone icon and the text 'Phone Number'.

Obrázek 11 - Formulář do vyplnění

The image shows the same three input fields as in the previous image, but now they are filled with data. The first field contains 'Dmitry Arkhipenko', the second contains 'info@example.com', and the third contains '+ (420) 777 889 900'.

Obrázek 12 - Formulář po vyplnění

```
<form>
```

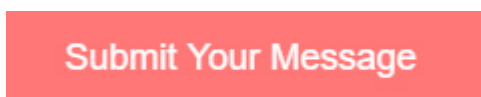
```
  <input type="text" placeholder="Full Name" required>
```

```
  <input type="email" placeholder="Email Address" required>
```

```
  <input type="tel" placeholder="Phone Number" required>
```

```
</form>
```

Odesílací tlačítko "Submit Your Message" (Odeslat zprávu) je definováno tak, že se při najetí kurzorem myši barva pozadí změní na jinou barvu. K tomu se využívají výše zmíněné pseudotřída ":hover" a vlastnost "opacity" jako v případě sekce "klienti". Používá se takový efekt pro lepší přehlednost, když se návštěvník webové stránky rozhodne vyplnit formulář a poslat zprávu. Vypadá takový efekt následovně.



Obrázek 13 - Tlačítko před najetím myši



Obrázek 14 - Tlačítko po najetí myši

```
#contact .btn {
```

```
  background-color: #ff7777;
```

```
        color: #fff;
    }
    #contact .btn:hover {
        background-color: #ff3333;
    }
```

4.2 Validita zdrojového kódu

Po dokončení vytvoření webové stránky je potřeba ověřit validitu zdrojového HTML kódu. Pro daný účel slouží speciální nástroj na oficiálním webu W3 konsorcia, který je umístěn na adrese "validator.w3.org". Nabízí se několik možností: buď uvést URL adresu webové stránky, nahrát soubor `.html` anebo vložit zdrojový kód do speciálního pole. Také je možné zvolit, jaký typ deklarace dokumentu se má kontrolovat, v případě kontroly kódu napsaného pomocí HTML5 je třeba vybrat typ deklarace "HTML5 (experimental)". Po odeslání požadavku se hned zobrazí výsledek kontroly.

Výsledek ukázal, že vytvořená webová stránka je zcela validní.

4.3 Testování použitelnosti HTML5

V této části je provedeno testování použitelnosti nových tagů a atributů specifikace HTML5, zobrazení vytvořené webové stránky v prohlížeči starší verze a testování responzivního designu na mobilu a tabletu.

4.3.1 Testování audio a video

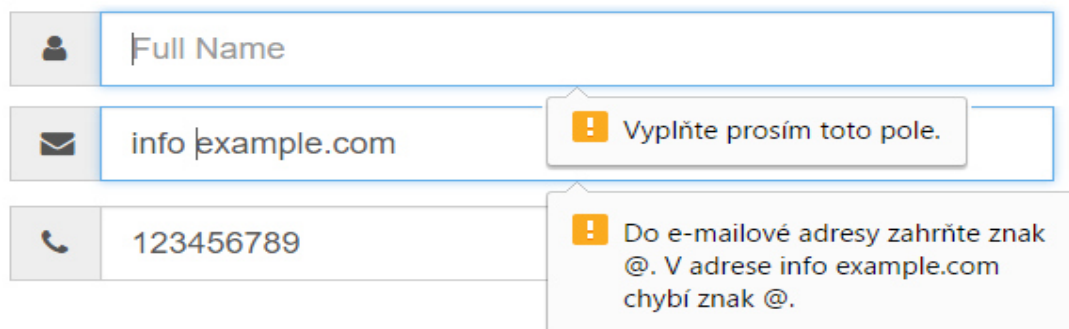
Elementy `<audio>` a `<video>` slouží podporou pro přehrávání audio a video souborů bez nutnosti instalace doplňujících pluginů. V dnešní době většina moderních verzí internetových prohlížečů podporují přehrávání audio a video souborů pomocí tagů `<audio>` a `<video>`, ale pro některé starší verze je potřeba zajistit různé kodeky jak videa, tak i audia. Například pro přehrávání audia souboru by bylo vhodné mít nejenom populární kodek mp3, ale i další kodeky jako jsou `ogg/vorbis`, `wav` a `AAC`. V případě video souboru je vhodné mít kodeky `ogg/theora`, `H.264` a `WebM`. Takovým způsobem se lze vyhnout problémům přehrávání audia a videa v různých prohlížečích.

Situace s nejstaršími verzemi prohlížečů je odlišná od verzí moderních. Například internetový prohlížeč "Internet Explorer 6" je známý jako jeden z nejvíce problémových prohlížečů. Po instalaci této verze "IE6" byl proveden pokus o přehrávání jak videa, tak i

audia, které byly umístěny pomocí nových tagů HTML5. Po otevření dokumentu v prohlížeči se stalo zřejmé, že daná verze nepodporuje nové tagy `<audio>` a `<video>`. Nakonec bylo zjištěno pomocí webu "htmlbook.ru", že nové tagy multimédií prohlížeč "Internet Explorer" podporuje jenom od verze 9 a výš.

4.3.2 Nové atributy formuláře

V nové specifikaci HTML5 byly přidány nové atributy v tagu `<input>`, které nabízí sadu možností ovládání polem. Při vytvoření webové stránky byly použity atributy `placeholder` a `required`. První atribut slouží pro nápovědu, co má návštěvník zapsat do příslušného pole, druhý atribut kontroluje, aby všechna pole, která jsou povinná, byla vyplněna. Daná kontrola vyplnění povinných polí se provede, až proběhne kliknutí na tlačítko "Odeslat". Také byly použity nové HTML5 hodnoty atributu `"type"`, které určují, pro jaký účel slouží příslušné pole. Byly použity hodnoty `"tel"` (mobilní číslo) a `"email"` (emailová adresa). Následující obrázek (Obr.15) znázorňuje, jak vypadá kontrola při vyplnění formuláře.



The image shows a web form with three input fields. The first field is labeled 'Full Name' with a person icon. The second field is labeled 'info@example.com' with an envelope icon and a validation message: 'Vyplňte prosím toto pole.' The third field is labeled '123456789' with a telephone icon and a validation message: 'Do e-mailové adresy zahrňte znak @. V adrese info@example.com chybí znak @.'

Obrázek 15 - Kontrola vyplnění formuláře

V průběhu testování bylo zjištěno, že hodnota `"email"` atributu `type` testuje emailovou adresu pouze na přítomnost zavináče, netestuje se přítomnost tečky. Takové řešení je podmíněno tím, že existují také adresy typu `"example@localhost"`, které neobsahují tečku.

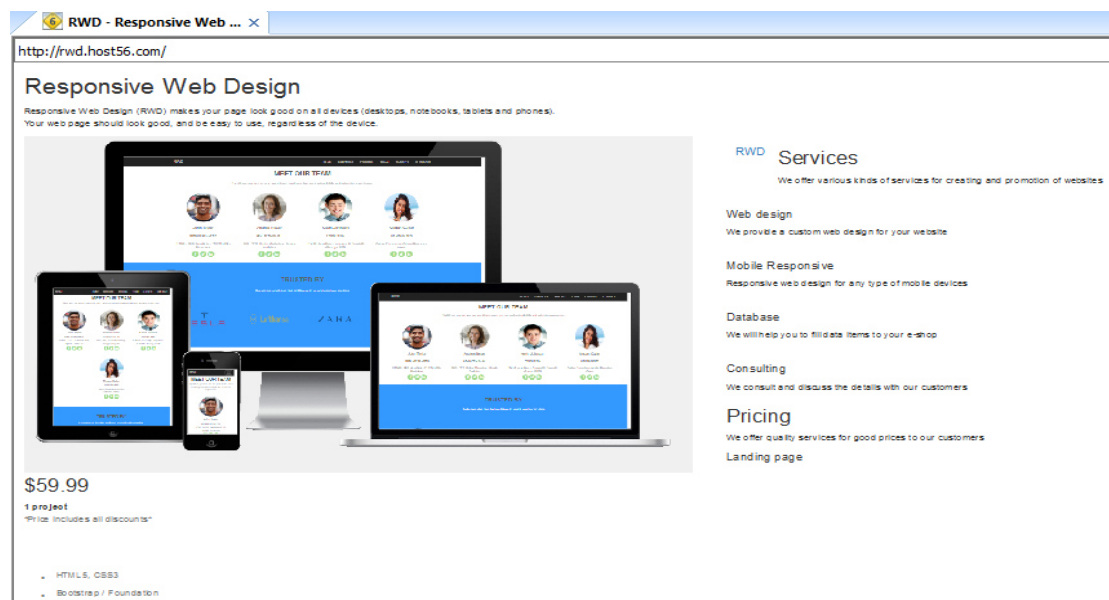
Také bylo provedeno testování kontroly vyplnění formuláře v internetovém prohlížeči staré verze, a to je v "Safari 5.1". Po nahrání stránky bylo vyplněno pouze pole "telefonní číslo" a dále stisknuto tlačítko "Odeslat". Výsledkem bylo odeslání celého formuláře i když nebyla vyplněna všechna pole. Z toho vyplývá, že atribut `required` a nové hodnoty atributu `type` nejsou podporovány danou verzí prohlížeče.

4.3.3 Zobrazení v Internet Explorer

Prohlížeče "Internet Explorer" starších verzí mají problém se zobrazením nejen nových funkcí specifikace HTML5, jak bylo zjištěno v průběhu testování "audio" a "video" značek, ale také i s grafickou částí webové stránky.

Po prostudování informací ohledně podpory různých funkcí HTML a CSS, které byly využity při tvorbě webové stránky, že verze "Internet Explorer" starší než 9 mají slabou podporu některých funkcí jak HTML, tak i CSS.

Pro kompletnější představu, jak bude vypadat navržená webová stránka v dané situaci, byla nainstalována verze 6 prohlížeče "Internet Explorer". Po nahrání stránka vypadala následovně.



Obrázek 16 - Zobrazení v Internet Explorer 6

Výše uvedený obrázek demonstruje, že webová stránka vypadá jako kdyby byl odpojen soubor kaskádových stylů. Z toho vyplývá, že prohlížeč "Internet Explorer 6" nepodporuje ani nové značky HTML5, ani většinu pokročilých funkcí kaskádových stylů, proto by bylo vhodné nastavit alternativní variantu zobrazení prvků, například v případě pozadí místo obrázku lze nastavit nějakou barvu, která bude sloužit náhradní variantou pro dekoraci.

4.3.4 Responzivní design

Responzivní design je nedílnou součástí v případě, když je potřeba mít webovou stránku dostupnou na všech typech zařízení. Responzivní design se vytváří pomocí

kaskádových stylu, které definují, jak mají jednotlivé sekce webu vypadat při různých šířkách obrazovky, na které je zobrazena stránka.

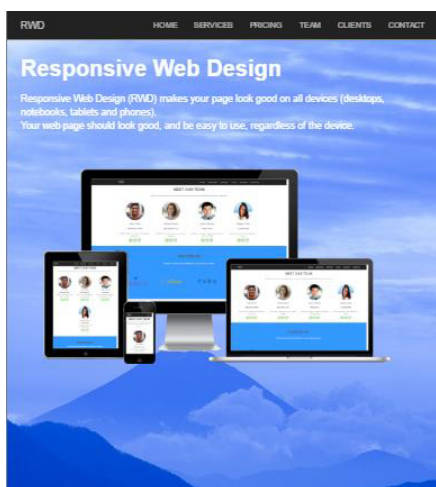
Testování probíhalo dvěma způsoby. Prvním způsobem byly použity v prohlížeči "Google Chrome" nástroje pro vývojáře, kde je možné otestovat přímo v prohlížeči, jak bude vypadat webová stránka na mobilu, tabletu nebo notebooku, a také daný nástroj umožňuje nastavit vlastní libovolnou šířku a výšku obrazovky a také je možné změnit orientaci zobrazení v případě mobilu a tabletu. Druhý způsob spočíval v testování responzivního designu přímo na zařízeních s menší šířkou obrazovky: mobil a tablet. Uvedené obrázky demonstrují, jak vypadá vytvořená webová stránka po otevření na mobilu a tabletu ve dvou způsobech orientace obrazovky.



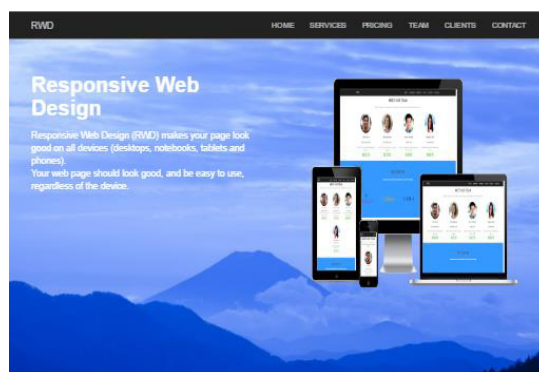
Obrázek 17 - Mobil (360×640)



Obrázek 18 - Mobil (640×360)

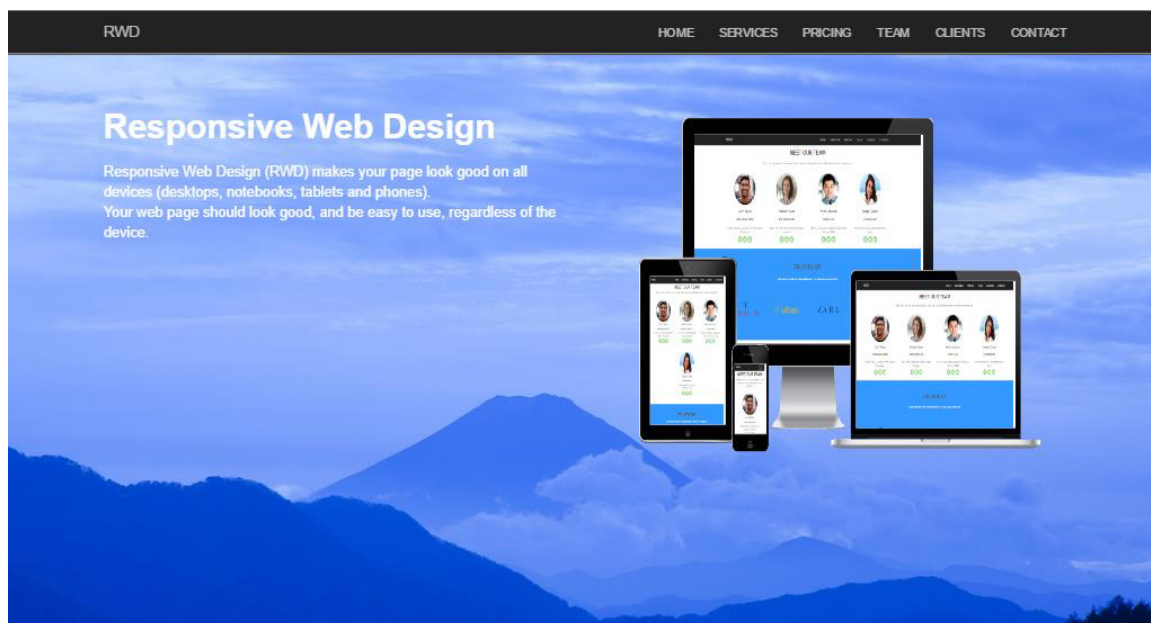


Obrázek 19 - Tablet (768×1024)



Obrázek 20 - Tablet (1024×768)

Finální verzi vytvořené webové stránky pro stolní zařízení jako jsou počítače a notebooky demonstruje následující obrázek.



Obrázek 21 - Notebook (1366×768)

5 Výsledky a diskuze

Webová stránka v praktické části bakalářské práce byla vytvořena pomocí nových technologií, které přináší specifikace HTML5, a také byly použity funkce kaskádových stylů pro grafické formátování dokumentu. Díky jejich použití při tvorbě jednostránkového webu byly otestovány teoretické znalosti nových technologií.

Testování podpory přehrávání audio a video souborů, použití nových atributu a hodnot při vytváření kontaktního formuláře proběhlo jak v moderních verzích internetových prohlížečů, tak i v starších verzích. Většina moderních prohlížečů bezproblémově podporuje nové funkce HTML5. Se staršími verzemi to tak není. Při testování přehrávání audio a video souborů v prohlížeči "Internet Explorer 6" bylo zjištěno, že daná verze a ostatní až do verze 9 nepodporují nové tagy <audio> a <video>. Testování nových atributu formuláře ukázalo na příkladu webového prohlížeče "Safari 5.1", že dané atributy také nejsou propojovány staršími verzemi. Problémy s podporou nových funkcí HTML5 a CSS se také ukázaly na příkladu otevření vytvořené webové stránky v prohlížeči "Internet Explorer 6", který demonstroval nesprávné zobrazení stránky. Testování responzivního designu proběhlo úspěšně po nahrání webové stránky na mobilním zařízení a tabletu.

6 Závěr

Cílem porovnání dvou verzí HTML 4.01 a HTML5 v teoretické části byl přehled změn, které přináší nová specifikace. Z představeného zápisu kódu obou verzí je patrné, že zapsaný kód pomocí HTML5 je mnohem přehlednější a kompaktnější díky novým sémantickým značkám na rozdíl od předcházející verze HTML 4.01.

V průběhu zkoumání zdrojových kódů již existujících webových stránek bylo zjištěno, že většina z nich využívají nové funkce specifikace HTML5. Zřejmě to bylo především z nového typu zápisu deklarace dokumentu a využití nových sémantických značek. Také se často objevovaly ve zdrojovém kódu značky pro umístění audio a video souborů.

Vytvořená webová stránka ukázala použití pokročilých možností nové specifikace. Po otestování nových funkcí v různých internetových prohlížečích se dá říci, že moderní verze prohlížečů prokazují dobrou připravenost podpory novinek od specifikace HTML5. Starší verze prohlížečů mohou činit problémy s korektním zobrazením webu.

Hlavními přínosy specifikace HTML5 lze jmenovat zavedení přehrávání video souborů bez nutnosti instalace doplňkových modulů a funkce nápovědy a kontroly vyplnění formuláře, kde již není nutné k tomu používat JavaScript. Za další přínos lze také považovat zavedení nových strukturních značek, které mají sémantický význam pro vyhledávací systémy.

7 Seznam použitých zdrojů

Literatura:

- [1] BROWN, Tiffany B., Kerry BUTTERS a Sandeep PANDA. *HTML5 okamžitě: [ovládněte HTML5 za víkend]*. Brno: Computer Press, 2014. ISBN 978-80-251-4296-7.
- [2] GOLDSTEIN, Alexis, Louis LAZARIS a Estelle WEYL. *HTML5 a CSS3 pro webové designéry*. Brno: Zoner Press, 2011. Encyklopedie webdesignera. ISBN 978-80-7413-166-0.
- [3] KOSEK, Jiří. *HTML: tvorba dokonalých www stránek: podrobný průvodce*. Praha: Grada, 1998. Průvodce (Grada). ISBN 80-7169-608-0.

Internetové zdroje:

- [4] W3Schools Online Web Tutorials. *HTML5 Semantic Elements*. [online]. [cit. 2016-04-10]. Dostupné z: <http://www.w3schools.com/html/html5_semantic_elements.asp>
- [5] Optimalizace webových stránek. *HTML5*. [online]. [cit. 2016-06-26]. Dostupné z: <<https://www.optimism.ru/agency/articles/270/>>
- [6] W3C. *About W3C*. [online]. [cit. 2016-07-10]. Dostupné z: <<https://www.w3.org/Consortium/>>
- [7] HTMLbook. *MIME-typ dokumentu*. [online]. [cit. 2016-08-20]. Dostupné z: <<http://htmlbook.ru/samlayout/rezhimy-brauzerov/mime-tip-dokumenta>>
- [8] HTMLbook. *Element <!DOCTYPE>*. [online]. [cit. 2016-08-20]. Dostupné z: <<http://htmlbook.ru/html/!doctype>>

8 Přílohy

Vytvořená webová stránka je umístěna na adrese **rwd.host56.com**