

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



Bakalářská práce

Respozivní design

Jan Belšán

© 2017 ČZU v Praze

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Jan Belšán

Informatika

Název práce

Responzivní design

Název anglicky

Responsive design

Cíle práce

Hlavním cílem bakalářské práce je analyzovat možnosti tvorby webových aplikací, které vyhovují požadavkům na současný responzivní design.

Díložními cíli práce jsou:

- charakterizovat technologie pro tvorbu webových aplikací
- analyzovat požadavky na responzivní design
- porovnat programová řešení (frameworky) na straně klienta
- v praktické části vytvořit responzivní webovou aplikaci
- formulovat obecné i specifické závěry

Metodika

Při zpracování BP budou využívány poznatky z odborné literatury a z relevantních webových stránek.

V teoretické části bude zajištěno porovnání frameworků a analýza zařízení, s kterými uživatelé nejčastěji pracují.

V praktické části bude vytvořena vzorová webová aplikace.

Na závěr práce bude zajištěna syntéza získaných poznatků a formulovány obecná i specifická doporučení.

Doporučený rozsah práce

35

Klíčová slova

Responzivní design, www aplikace, web design

Doporučené zdroje informací

CASTRO, E. – HYSLOP, B. *HTML5 a CSS3 : názorný průvodce tvorbou WWW stránek*. Brno: Computer Press, 2012. ISBN 978-80-251-3733-8.

KADLEC, T. *Responzivní design profesionálně*. Brno: Zoner Press, 2014. ISBN 978-80-7413-280-3.

LUBBERS, P. – ALBERS, B. – SALIM, F. *HTML5 : programujeme moderní webové aplikace*. Brno: Computer Press, 2011. ISBN 978-80-251-3539-6.

Předběžný termín obhajoby

2016/17 LS – PEF

Vedoucí práce

doc. Ing. Zdeněk Havlíček, CSc.

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 12. 03. 2017

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Responzivní design" 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 14. března 2017

Poděkování

Rád bych touto cestou poděkoval doc. Ing. Zdeňku Havlíčkovi, CSc. za odborné vedení, pomoc a cenné rady při zpracování bakalářské práce.

Responzivní design

Souhrn

Předmětem této bakalářské práce je analyzovat možnosti tvorby webových aplikací, které vyhovují požadavkům na současný responzivní design. Dnešní elektronická zařízení se začínají čím dál tím více zmenšovat. Zároveň je kladen důraz na správné zobrazování obsahu internetu na všech platformách naprosto bez problému.

Práce je rozdělena na 3 částí. První část, teoretická, se zabývá pojmy spojené s tvorbou webových aplikací, které jsou nastudovány a čerpány z odborné literatury. Dále vysvětluje požadavky na vytvoření responzivní www stránky. Také se zde zaměřuji na porovnání stávajících řešení (Frameworků). Bakalářská práce zohledňuje jejich využitelnost a praktičnost. Druhá část se zabývá praktickou stránkou a představím zde responzivní web a veškerou jeho funkčnost. V posledním úseku se zaměřím na shrnutí závěru bakalářské práce a vyhodnotím nejvhodnější variantu pro tvorbu responzivního webu.

V práci jsou zhodnoceny současné technologie vývoje responzivního webu.

Klíčová slova: web, responzivita, přizpůsobivost, html, css, framework, optimalizace

Responsive design

Summary

The subject of this bachelor work is to analyze possibilities of creating web applications which meet requirements of today's responsive webs. Today's electronic devices begin to be smaller. The emphasis is placed on proper content displaying without any problems.

The Bachelor work is divided into three parts. First part, theoretical, explains the main concepts associated with the creating web applications that are drawn from the literature. This work also describes the requirements for creating www pages. Next words are focused on the comparison of existing solutions (Framework). Furthermore, taking their usefulness and practicality. The second part deals with the practical aspect and I introduce there my website and all its functionality. The last section will focus on summary and conclusion, the thesis will evaluate the best option for creating responsive site.

The paper reviewed the current technology developments responsive site.

Keywords: web, responsivity, flexibility, html, css, framework, optimization

Obsah

1 Úvod.....	11
2 Cíl práce a metodika	12
2.1 Cíl práce	12
2.2 Metodika práce.....	12
3 Teoretická východiska	13
3.1 Responzivní web design.....	13
3.1.1 Trend webových stránek	13
3.1.2 Responzivita.....	14
3.2 Základní pilíře responzivního webu.....	14
3.2.1 Tekuté mřížka	14
3.2.2 Obrázky, které mění velikost	15
3.2.3 CSS Media Queries.....	15
3.3 Důležité pojmy spojené s responzivitou webu.....	16
3.3.1 Relativní jednotky	16
3.3.2 Procentuální velikost.....	16
3.3.3 Breakpoint.....	17
3.3.4 Způsoby rozložení.....	18
3.4 Princip responzivního webu.....	20
3.4.1 Responzivní rozložení.....	20
3.4.2 Implementace	21
3.5 Požadavky na responsivní design.....	22
3.5.1 Nástroje pro tvorbu webové stránky	22
3.5.2 Webové prohlížeče.....	24
3.5.3 Desktop First princip.....	25
3.5.4 Mobile First princip	26
3.5.5 Content First princip	27
3.5.6 Optimalizace výkonu webové stránky	27
3.6 Frameworky	28
3.6.1 Bootstrap	28
3.6.2 Foundation	28
3.6.3 Skeleton.....	29
3.6.4 Porovnání	29
4 Vlastní práce	31
4.1 Návrh.....	31
4.1.1 WireFrame	31
4.2 Kódování.....	37
4.3 Testování.....	42

4.3.1	Mobilní zařízení	42
4.3.2	Desktopová zařízení	43
5	Závěr.....	44
5.1	Charakterizovat technologie pro tvorbu webových aplikací.....	44
5.2	Analyzovat požadavky na responzivní design	44
5.3	Porovnat programová řešení (frameworky)	44
5.4	V praktické části vytvořit responzivní webovou aplikaci	45
5.5	Formulovat obecné i specifické závěry.....	45
6	Bibliografie	46
7	Zdroje obrázků a tabulek.....	47

Seznam obrázků

OBRÁZEK 1 - ZOBRAZENÍ NA ODLIŠNÝCH ZAŘÍZENÍCH.....	13
OBRÁZEK 2 – ROZLOŽENÍ STRÁNKY V ZÁVISLOSTI NA ŠÍŘKY ZOBRAZENÍ ZAŘÍZENÍ.....	17
OBRÁZEK 3- ROZDÍLNÉ ZÁPISY HTML SPECIFIKACÍ	23
OBRÁZEK 4- WIREFRAME MOBILNÍ VERZE WEBOVÉ STRÁNKY - ČÁST 1	33
OBRÁZEK 5- WIREFRAME MOBILNÍ VERZE WEBOVÉ STRÁNKY - ČÁST 2	34
OBRÁZEK 6- WIREFRAME DESKTOPOVÉ VERZE WEBOVÉ STRÁNKY - ČÁST 1	36
OBRÁZEK 7- WIREFRAME DESKTOPOVÉ VERZE WEBOVÉ STRÁNKY - ČÁST 2	37
OBRÁZEK 8- SPOJENÍ CSS SOUBORŮ S HTML SOUBOREM	38
OBRÁZEK 9- KÓD PRO VYTVOŘENÍ NAVIGAČNÍ LIŠTY, KTERÁ MÁ SVOU POLOHU NA HORNÍ ČÁSTI STRÁNKY	38
OBRÁZEK 10 - KÓD NAVIGAČNÍ LIŠTY A MENU (BOOTSTRAP)	39
OBRÁZEK 11- KÓD NAVIGAČNÍ LIŠTY A MENU (FOUNDATION).....	39
OBRÁZEK 12 - KÓD NAVIGAČNÍ LIŠTY A MENU (SKELETON).....	40
OBRÁZEK 13 - HLAVNÍ STRANA PO VYTVOŘENÍ NAVIGAČNÍ LIŠTY A HEADERU	40
OBRÁZEK 14 - KÓD PRO JEDNU POLOŽKU SLUŽBY	41
OBRÁZEK 15 - ZOBRAZENÍ GALERIE	41
OBRÁZEK 16 - KÓD VZTAHUJÍCÍ SE KE KONTAKTŮM	42
OBRÁZEK 17 - ZOBRAZENÍ WEBOVÉ STRÁNKY NA MOBILNÍCH ZAŘÍZENÍCH (IPHONE 6S, IPHONE 6S, NOKIA LUMIA 520)	42
OBRÁZEK 18 - ZOBRAZENÍ WEBOVÉ STRÁNKY V PROHLÍŽEČI MOZILLA FIREFOX	43

Seznam tabulek

TABULKA 1 - VÝČET POUŽÍVANÝCH ROZLIŠENÍ DLE ZAŘÍZENÍ	18
TABULKA 2 - POROVNÁNÍ FRAMEWORKŮ VÍCEKRITERIÁLNÍ METODOU	29

1 Úvod

Responzivní web a web jako takový se neustále vyvíjí. Má to za následek čím dál více zařízení, které mají různou velikost, různé rozlišení a různé způsoby zobrazování webového obsahu.

Webové stránky jsou základem celého internetu. Jsou na nich zobrazovány informace, které uživatel vyhledává. Dnešní funkce jsou rozmanité od pouhého zobrazení textu, až po spuštění plnohodnotné hry, či spravování svého bankovního účtu. Informace na webových stránkách jsou prezentovány ve formě hyper textu, který je vytvořen pomocí HTML značek a dalších technologií. Stránky mohou být buďto statické, nebo dynamické. Statické webové aplikace obsahují stále stejný obsah a nemění se v čase. Dynamické naopak mění svou podobu v čase a může tak interagovat s uživatelem.

Tato bakalářská práce se zabývá tvorbou responzivního webu v roce 2016. K tomuto tématu má autor blízký vztah, protože ho tato problematika zajímá a zabývá se s ní v každodenním životě.

2 Cíl práce a metodika

2.1 Cíl práce

Hlavním cílem bakalářské práce je analyzovat možnosti tvorby webových aplikací, které vyhovují požadavkům na současný responzivní design.

Díličními cíli práce jsou:

- charakterizovat technologie pro tvorbu webových aplikací
- analyzovat požadavky na responzivní design
- porovnat programová řešení (frameworky) na straně klienta
- v praktické části vytvořit responzivní webovou aplikaci
- formulovat obecné i specifické závěry

2.2 Metodika práce

Při zpracování BP budou využívány poznatky z odborné literatury a z relevantních webových stránek. V teoretické části bude zajištěno porovnání frameworků pomocí vícekritériální analýzy variant a analýza zařízení, s kterými uživatelé nejčastěji pracují. V praktické části bude vytvořena vzorová webová aplikace.

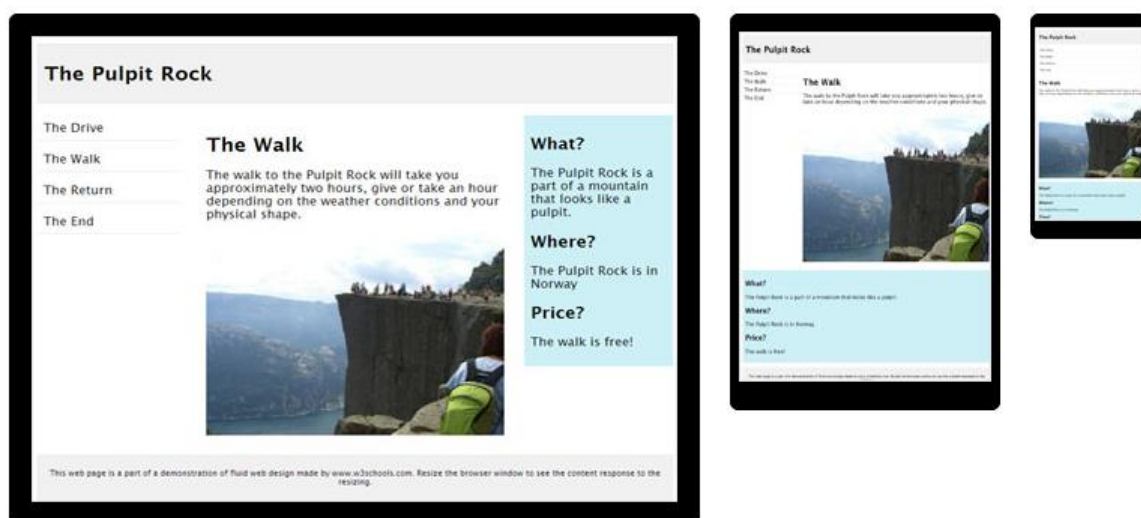
Na závěr práce bude zajištěna syntéza získaných poznatků a formulovány obecná i specifická doporučení.

3 Teoretická východiska

3.1 Responzivní web design

Samotné designování a přizpůsobení stránky k flexibilitě a adaptaci je nazýváno jako Responzivní Web Design (RWD). Tento termín vyslovil Ethan Marcotte, web designer na volné noze, v květnu roku 2010 v magazínu A List Apart. RWD je princip, který se zaměřuje na optimalizaci webové stránky tak, aby poskytla uživateli co možná nejlepší prožitek s návštěvou webu.

Je zde několik důvodů, proč vybrat responzivní design místo klasického, kde se pro rozdílná zařízení vytváří rozdílné stránky a nastavení. Vývojář nemusí vytvářet několik kopií, které se dále špatně spravují. Uživatel najde vždy, co potřebuje, a to včetně obsahu, který zrovna očekává. (Duckett, 2008)



Obrázek 1 - Zobrazení na odlišných zařízeních (Zdroj: https://www.w3schools.com/html/html_responsive.asp)

3.1.1 Trend webových stránek

Na konci minulého století se ve světě ukázal internet jako součást mobilních zařízení. Ve většině případu fungoval pouze jako strohý přenašeč textu a tištěných informací.

V roce 2007 však přišla firma Apple na trh s revolučním zařízením. Byl jím iPhone, který se stal jedním z prvních smartphonů na světě a web mohl přenášet nejen text, ale podobný obsah jako tehdejší počítače. Do této doby se stránky zobrazovaly pouze na

větších monitorech, než je velikost mobilního zařízení. S tímto přišla i potřeba zobrazovat webové stránky jinak a lépe na menším zařízení. Jako první řešení problému se nabídlo udělat odlišnou stránku pro iPhone a pro počítače. Nicméně netrvalo dlouho a na trh vnikly ostatní mobilní zařízení s podobným potenciálem k zobrazování webového obsahu. Nebylo technicky možné vytvářet tolik verzí webu, a tak se muselo přijít na jinou metodu. Nejlepším řešením se zdálo vytvořit design, který by se přizpůsoboval všem velikostem zařízení. Toto řešení se pojmenovalo jako Responzivní web design.

Nejen že se responzivní design neustále vylepšuje, ale zároveň se snaží vyhovět i výkonem všem zařízením. Tento trend neustále roste a s příchodem masivní veřejnosti na pole internetu byly tyto potřeby mnohem vyšší. V 21. století pak vlastnilo chytrý mobil již 58 % dospělých lidí v Americe. (Peterson, 2014)

3.1.2 Responzivita

Jedná se o způsob stylování HTML dokumentu, které zaručí, že zobrazení stránky bude optimalizováno pro všechny druhy nejrůznějších zařízení (mobily, notebooky, netbooky, tablety atd.). Především díky vlastnosti Media Queries, která je zahrnuta ve specifikaci CSS3, lze rozpoznat vlastnosti zařízení, na kterém je stránka prohlížena a přizpůsobit tak samotnou stránku a její obsah.

3.2 Základní pilíře responzivního webu

3.2.1 Tekuté mřížka

V angličtině Fluid web design, můžeme přeložit jako tekutá, nebo plovoucí mřížka. Je to první pilíř responzivního web designu. Tato myšlenka vznikla v návaznosti na nápad vytvoření plovoucího rozložení, kdy se v roce 2006 hodně debatovalo, jaká je optimální šířka rozložení pro zobrazování.

Tekutá mřížka se dá interpretovat jako rozložení jednotlivých částí webové stránky. Je to způsob, jak se postaví layout stránky. Mřížka se skládá ze sloupců, kterých může být různé množství a od toho se odvíjí složitost dané stránky. Nejjednodušší stránky mají do tří sloupců. Nejsložitější mohou mít naopak až dvanáct takových útvarů. Sloupce mohou být orientovány jak horizontálně, tak i svisle.

Dříve bylo běžnou praxí vytvořit mřížku s pevnými rozměry ve fixních jednotkách (px). Plovoucí mřížka však uvádí pouze relativní šířky sloupců. Podle počtu sloupců se vypočítá šířka pro jednotlivé sloupce, která se udává v procentech. Pokud se změní velikost zobrazení, mřížka se přepočítá a web se tak přizpůsobí nové velikosti. (Marcotte, & [foreword by Jeremy Keith]., 2011), (Craig Sharkie, 2013)

3.2.2 Obrázky, které mění velikost

Dalším pilířem responzivního web designu je vlastnost obrázku měnit svou velikost. Obrázek má jako základní vlastnost rozměr a hustotu pixelů, která je nejčastěji udávaná na palec čtvereční. V tekuté mřížce představují tyto dvě vlastnosti poměrně velký problém.

Pilíře responzivního webu jsou zaměřené na uživatele. Uživatel požaduje hned po funkční stránce věci, také grafickou. Skrz grafické prvky probíhá komunikace mezi uživatelem a webovým obsahem. Flexibilní obrázek tudíž musí při změně mřížky reagovat a být pro uživatele stále čitelný. Určitě by se dalo navrhnout, že pokud zvolíme dostatečně velké rozlišení obrázku, neměl by nastat problém. Avšak problém nastává při zobrazení na malých zařízeních, kde se stává takto kvalitní obrázek nečitelným. Nemluvě o datové velikosti takového obrázku, který je nešetrný k velikosti přenášených dat. Řešením je tedy HTML5 CSS3, které umí načítat různé zdroje obrázků pro různé velikosti obrazovek a dále s nimi pracovat. (Marcotte, & [foreword by Jeremy Keith]., 2011), (Craig Sharkie, 2013)

3.2.3 CSS Media Queries

Jedná se o stěžejní součást a třetí pilíř responzivního webu. Procentuální vyjádření a proporcionální design nemusí dostačovat k správnému zobrazení jako jsou třeba mobilní zařízení nebo obrovské obrazovky stolních počítačů. Rozložení pak může vypadat jako zúžená obdoba originálního návrhu.

Na stolním PC funguje rozložení naprosto bez problému například při 85 % šířce. Avšak na malém displeji telefonu je obsah tohoto okna úzký a skoro nečitelný a obecně neesteticky vypadající. V tomto případě je nutné využít breakpoint pomocí media queries. Nadefinují se nové nastavení CSS souboru a šířka zobrazení se změní.

Tato funkce byla představena v roce 2012 organizací World Wide Web (W3C) jako součást standardu CSS verze 3. Do té doby nešlo nadefinovat určité CSS nastavení na

určitou velikost viewportu. Nyní tato funkce obsahuje několik předem nadefinovaných viewportů zařízení. Mezi ně patří například projektor, TV, nebo speciální zobrazení pro handicapované uživatele.

Tato technologie pracuje na bázi dotazů na medium. Ve výsledku toto znamená dotaz na aktuální zobrazovací plochu a poté přiřazení správné části kódu. (Marcotte, & [foreword by Jeremy Keith]., 2011), (Craig Sharkie, 2013)

3.3 Důležité pojmy spojené s responzivitou webu

3.3.1 Relativní jednotky

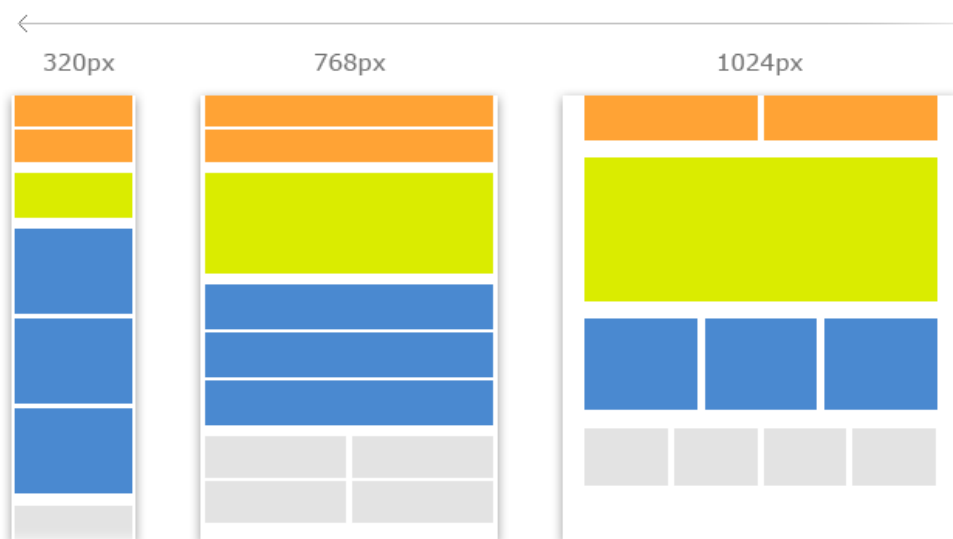
K optimálnímu zobrazení webové stránky se používají takzvané relativní jednotky. Existuje několik druhů:

- em – relativní velikost vztažená k velikosti konkrétního písma pro HTML element. Také nazývána jako čtverčík. Čtverčík reprezentuje rozměr čtverce daného velikostí kuželky písma v bodech
- rem – tato jednotka je podobná jako em, avšak vztahuje se k velikosti písma kořenového elementu dokumentu, tudíž je její relativita konstantní pro celý dokument. Podpora tohoto způsobu je ovšem některými prohlížeči omezena.
- ex – reprezentace relativní velikosti střední výšky použitého písma. Většinou platí, že $2 * ex = 1em$
- ch – poslední a méně využívaná varianta, která reprezentuje šířku znaku 0 právě použitého písma

(Craig Sharkie, 2013)

3.3.2 Procentuální velikost

Tento pojem je hojně využívaný v oblasti responzibility. Určuje se jím velikost konkrétního prvku na webové stránce na určitém zařízení. Vypočítává se jako (požadovaná šířka / kontext) * 100. Tímto způsobem se obsah webové stránky přizpůsobí jakémukoli zařízení a není zde omezení značkou, velikostí displeje ani výkonem konkrétního zařízení. (Gasston, 2014)



Obrázek 2 – Rozložení stránky v závislosti na šířky zobrazení zařízení
 (Zdroj: <http://www.speeli.com/articles/view/Css-media-queries-for-standard-devices>)

3.3.3 Breakpoint

Jeden z nejzákladnějších pojmů vázající se k responzivnímu designu. Je to určitý bod, který značí šířku stránky. Při překročení této šířky se změní layout webové stránky. Šířka stránek se mění pomocí Media Queries. Ve většině případů se používají jako breakpointy šířky často používaných zařízení, ale toto pravidlo není tolik striktní a vývojář může použít jakoukoli šířku. Po překročení breakpointu se nemusí měnit webová stránka nijak důrazně, občas se pouze přeskupí prvky, nebo se změní velikost textu. Nejpoužívanější breakpoint systém vypadá následovně.

- Velmi malá zařízení (<768px) – například smartphony, PDA
- Malá zařízení (768px – 992px) – například tablety, netbooky
- Středně velká zařízení (992px – 1200px) – notebooky, monitory počítačů
- Velká zařízení (>1200px) – velké monitory počítačů, HD displeje, Smart TV

(Peterson, 2014)

3.3.4 Způsoby rozložení

Rozložení webové stránky je jednou z nejdůležitějších vlastností webu, protože přesně určuje, v jakém pořadí se webové prvky budou zobrazovat. Nejde zde jen o rozložení elementů stránky, ale důraz je kladen i na uživatelskou přívětivost a přístupnost. Cíl je vytvořit design vyhovující uživateli a zároveň atraktivní a hezký na pohled.

Screen resolution	Display ratio	Usage	Screen size / type
1366x768	16:9	19.1%	14" Notebook / 15.6" Laptop / 18.5" monitor
1920x1080	16:9	9.4%	21.5" monitor / 23" monitor / 1080p TV
1280x800	8:5	8.5%	14" Notebook
320x568	9:16	6.4%	4" iPhone 5
1440x900	8:5	5.7%	19" monitor
1280x1024	5:4	5.5%	19" monitor
320x480	2:3	5.2%	3.5" iPhone
1600x900	16:9	4.6%	20" monitor
768x1024	3:4	4.5%	9.7" iPad
1024x768	4:3	3.9%	15" monitor
1680x1050	8:5	2.8%	22" monitor
360x640	9:16	2.3%	
1920x1200	8:5	1.7%	24" monitor
720x1280	9:16	1.6%	4.8" Galaxy S
480x800	3:5	1.1%	
1360x768	16:9	0.9%	
1280x720	16:9	0.9%	720p TV

Tabulka 1- Výčet používaných rozlišení dle zařízení. (Zdroj: <http://www.rapidtables.com/web/dev/screen-resolution-statistics.htm>)

Tabulka výše zobrazuje používaná rozlišení k roku 2015. Rozlišení zařízení určuje, jaká část obsahu se bude zobrazovat a jestli tento pohled bude obsahovat všechny prvky, nebo se naopak musí některé přesunout pod sebe.

Dříve se hojně využívalo rozlišení 1024x768 jako stavební kámen pro každou novou webovou stránku. Toto ovšem způsobovalo, že uživatelé s menším rozlišením než s právě zmiňovaným byli nuceni posunovat web ze strany na stranu, aby dosáhli na úplný kraj. Tento fenomén se však podařilo téměř všude vymýtit a v dnešní době je už neakceptovatelný. (Peterson, 2014), (Marcotte, & [foreword by Jeremy Keith]., 2011)

3.3.4.1 Fixované rozložení

Fixované nebo také statické je rozložení stránky, kde je šířka obsahu nastavena v pixelech a nemění se v závislosti na velikosti okna prohlížeče.

Fixované rozložení je zastaralé řešení, přesto je dnes poměrně lehké najít webové stránky, které jsou touto metodou tvořeny. Nicméně se tato metoda přestala používat hned z několika důvodů:

- Fixované rozložení vytváří spoustu prázdného místa a tím porušuje několik principů designových pravidel
- Při nízkém rozlišení se zobrazí horizontální posuvník a stránka je esteticky nehezká
- Grafické prvky vyžadují vysoké rozlišení, i když není potřeba
- Použitelnost je velmi malá

(Fielding, 2014)

3.3.4.2 Tekuté rozložení

V tekutém rozložení se obsah stránky přizpůsobuje přímo velikosti konkrétního prohlížeče. Tento typ rozložení je v jeho základní formě také přežitý, protože neuvažuje, jak se obsah bude vykreslovat na extrémně velkých rozlišeních, a naopak na extrémně malých. Několik nevýhod tohoto rozložení:

- Je skoro nemožné navrhnout design, který by zachytil zobrazení na všech možných velikostech displeje a rozlišení
- Široké obrazovky vytvářejí dlouhé části textu, které jsou těžko čitelné
- Grafické prvky musí mít více nastavení šířek, pro různá rozlišení

(Fielding, 2014)

3.3.4.3 Adaptivní rozložení

Toto rozložení je velmi blízko responzivnímu typu. Základem jsou takzvané media queries, které zajistí adaptování obsahu různým obrazovkám a rozlišení.

Media queries jsou částí CSS3 specifikace, která je jednotkou velikosti výstupního zařízení. Hlavní rozdíl mezi adaptivním a responzivním rozložením je způsob zvětšování a zmenšování jednotlivých elementů na stránce. V adaptivním stylu se velikost mění pouze na nějakém bodě, což způsobuje „skoky“ v zobrazení. Výsledkem je totiž několik nastavení na různá rozpětí rozlišení.

Nevýhoda tohoto zobrazení bude hlavně nepředvídatelná velikost na jednotlivých zařízeních. Body, u kterých se obsah změní mohou být položeny daleko od sebe, a tudíž se může obsah jevit moc malý, nebo naopak moc velký. (Fielding, 2014)

3.3.4.4 Hybridní rozložení

Je samozřejmostí, že je mnoho způsobů, jak tvořit webové rozložení. Tyto styly se mohou kombinovat a vznikají tak hybridní verze. Pokud se responzivita doplňuje už k hotovému webu, často vznikají jen částečně responzivní weby. (Marcotte, & [foreword by Jeremy Keith]., 2011)

3.4 Princip responzivního webu

Responzivní web je webová stránka vytvořená za účelem předkládat uživateli co nejvíce přívětivý design a zároveň vykreslit stránku správně s co nejmenším počtem překreslení. (Wroblewski, 2011)

3.4.1 Responzivní rozložení

Responzivní rozložení se na rozdíl od adaptivního chová jinak mezi body změny rozlišení. Zatímco se adaptivní mezi body nemění, u responzivního rozložení se sice nemění uspořádání elementů, ale automaticky se zvětšují a zmenšují podle potřeby. Tudíž se vyeliminují „skoky“ mezi určitými body a přechod vypadá plynule. (Fielding, 2014)

3.4.2 Implementace

- Generátory responzivních stránek
 - Tento způsob není doporučovaný, protože veškerá optimalizace je zpracovaná strojově, a tak přesně nevíme, jak se stránka, v jaké situaci bude chovat a zároveň může obsahovat mnoho chyb.
- Flexibilní prvky margin a padding
 - Při zadávání hodnot do těchto atributů je potřeba zadávat relativní jednotky. Pokud vyplníme jednotky pixelů, budou se tvořit prázdná místa.
- Viewport
 - Viewport je zjednodušeně viditelná oblast v internetovém prohlížeči, a to bez horizontálních a vertikálních posuvníků. Problém zde nastává, kvůli relativnosti pixelů. Pixel na mobilním zařízení má například jinou velikost než CSS pixel.
- Media queries
 - Media queries jsou prostředky, které částečně zresponzivní stránku, použitím odlišného nastavení pro odlišné rozlišení. Toto umožní designérům vytvořit web, který je hezký na pohled a přesně víme, jak bude v různorodém rozlišení vypadat.
- Responzivní text
 - Hlavním účelem textu je, aby byl čitelný. Důležitý prvek v rozhodování o velikosti textu je vzdálenost uživatele od zařízení. CSS disponuje hned několika jednotkami velikosti textu, mezi které patří pixels, em, pt, %. Jejich použitelnost je individuální a dodnes se vedou velké diskuze, zda jsou všechny potřeba.
- Flexibilní obrázky
 - HTML a CSS soubory jsou velmi malé a nenáročné na datový objem. To se nedá říct o obrázcích a videu, popřípadě i jiných grafických prvcích. Mobilní zařízení mají skvělé vykreslovací schopnosti. Text zde vypadá ostrý, avšak obrázky jsou často rozmazané. Proto se k aktuálnímu zobrazení písma přidávají obrázky dvakrát tak větší, aby srovnaly krok s textem a byly stejně ostré.

(Duckett, 2008), (Fielding, 2014)

3.5 Požadavky na responsivní design

V roce 2009 napsal Ethan Marcotte první článek o responzivním rozložení a po 8 letech jsou tyto pravidla skoro identická.

“Recently, an emergent discipline called “responsive architecture” has begun asking how physical spaces can respond to the presence of people passing through them. Through a combination of embedded robotics and tensile materials, architects are experimenting with art installations and wall structures that bend, flex, and expand as crowds approach them.”

- Ethan Marcotte, Responsive Web Design, 2010

3.5.1 Nástroje pro tvorbu webové stránky

3.5.1.1 Webová stránka

Určité informace uspořádané čitelně a přehledně na internetu. K přehlednosti slouží několik pomůcek, mezi které patří například rozdělení nadpisy, tabulkami a seznamy. Tyto funkce jsou implementovány do hypertextových jazyků. Patří mezi ně například XHTML nebo HTML. Pomocí těchto jazyků vypisujeme značkami strukturu webu, kterou následně naplňujeme konkrétními informacemi. (Duckett, 2008)

3.5.1.2 HTML5

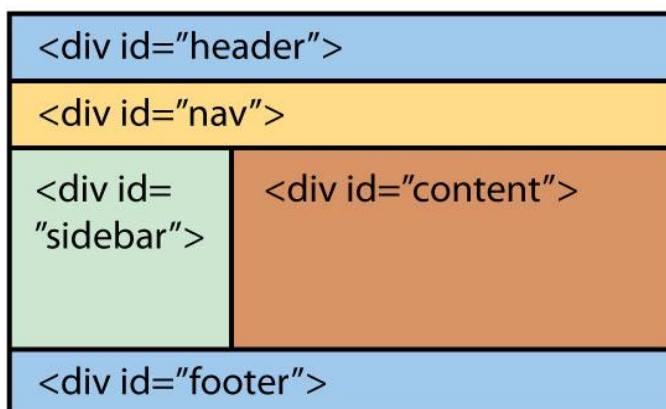
Takzvaný HyperText Markup Language je ve volném překladu hypertextový značkovací jazyk. Skládá se z něj prakticky každá dnešní webová stránka. Verze 5 byla dokončena v roce 2014 a obsahuje obsáhlé množství nových funkcí a inovací. Tyto novinky vycházejí hlavně z požadavků uživatelů jazyka a snaží se, co nejvíce vyhovět novým trendům.

Byly zde definovány nové tagy a atributy, které jsou kratší a mnohem rychlejší na zápis. Dále byl kladen důraz na přehlednost kódu jako takového, tudíž i vývojář, který nepracoval na vývoji stránky, má možnost se lehce zorientovat v kódu.

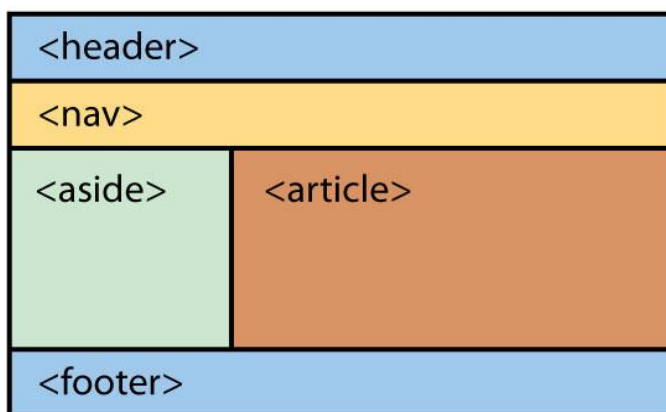
K velkému průlomů došlo v oblasti zobrazování multimediálního obsahu. Nová specifikace umožňuje vkládat multimediální obsah přímo a návštěvníci webu by tak neměli mít problém se zobrazováním tohoto obsahu, jako tomu bylo v předchozích verzích. Do

nové verze byl také přidán Canvas, což je vektorová grafika, která je nativní pro HTML5, a tak je možné vykreslovat grafy, či jednoduchou grafiku přímo pomocí HTML5. Další výhodou je možnost pracovat offline v aplikacích této specifikace. Také byl kladen důraz na samostatnost celé specifikace, a tak již není potřeba přidávat zásuvné moduly pro vytvoření několika efektů. To také napomáhá při výkonnostní optimalizaci. Tato skutečnost se projeví hlavně na méně výkonnějších zařízeních a chytrých TV a stanou se tak více využitelné. (Hogan, 2010), (MacDonald, 2013)

HTML4



HTML5



Obrázek 3- rozdílné zápisy HTML specifikací (Zdroj: <http://electronicdesign.com/embedded/whats-difference-between-html-4-and-html-5>)

3.5.1.3 CSS3

Cascading Style Sheets je ve volném překladu skupina kaskádových stylů. Je to popis pro určitý jazyk, který označuje, jakým způsobem se mají zobrazovat elementy a prvky značkovacích jazyků.

Tyto dvě skupiny k sobě mají velmi blízko a přímo spolu souvisí. Vytváří se již od roku 2005 a v posledních letech se objevuje už i nástupce, a to je CSS4.

HTML5 a CSS3 spolu velice úzce souvisí a byly vyvíjené společně, tak aby se doplňovaly. Ohled byl brán hlavně na stránku vzhledovou, kdy CSS3 umožňuje grafikům snadnější práci s obrázky, pozadím a různými grafickými prvky. Mezi nejdůležitější patří zaoblování hran, stínování a několik akcí pro text. Další grafickou vychytávkou je přidání průhlednosti barvy, které je realizováno pomocí alpha kanálu, který se nachází za kódem barvy a určuje tak procentuálně průhlednost. CSS3 je tak základním pilířem pro práci s responzivním designem. (Fielding, 2014), (Powers, 2012)

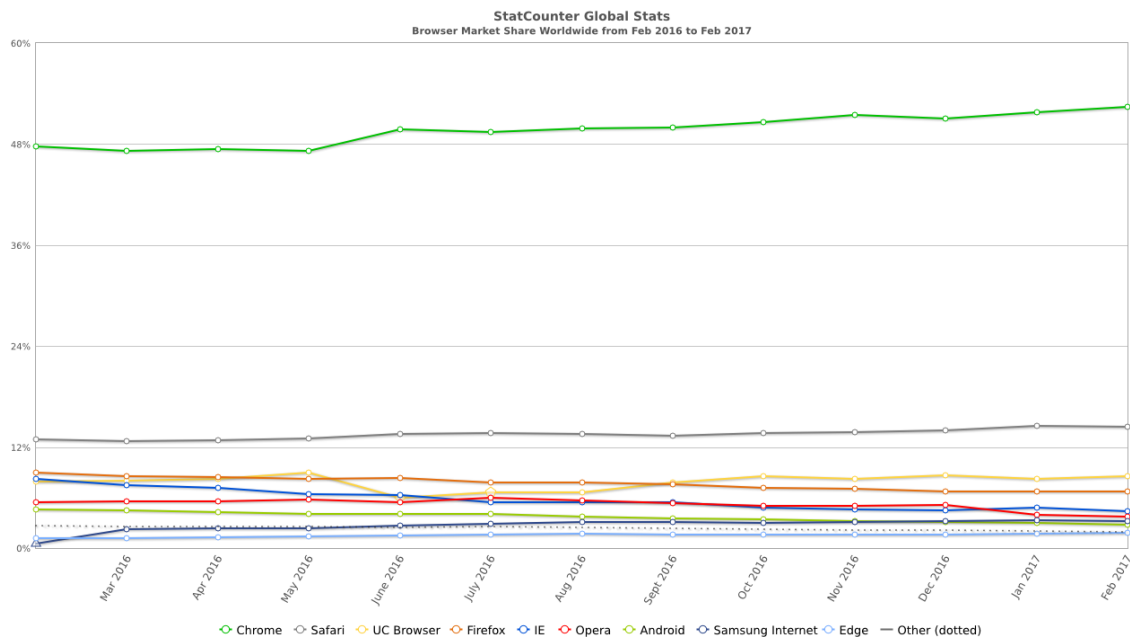
3.5.1.4 JavaScript

I když bylo řečeno, že se moderní prostředky většinou obejdou bez většiny zásuvných modulů a podpůrných komponentů z třetí strany, stále jsou některé prvky, které využívají tyto prostředky. Jedním z nich je právě JavaScript, který většinou zaručuje zpětnou kompatibilitu ve starších prohlížečích. Pomocí knihovny je vznesen dotaz na prohlížeč a pokud dotaz vrátí hodnotu nepodporovaného prohlížeče, jsou kusy kódu nahrazeny jiným, nebo je prvek nedostupný. Touto metodou můžeme nahradit HTML ale i CSS kód a tak změnit zobrazení uživateli. (Peterson, 2014)

3.5.2 Webové prohlížeče

K jednomu z největších problémů v rámci responzivního designu patří nastavení webové stránky na všechny webové prohlížeče. Reálně je tato věc těžko dosažitelná, hlavně proto, že uživatelé mohou používat starší, již nepodporované, webové prohlížeče. Není opravdu žádný nástroj ani technika, jak jim v takovém jednání zabránit.

Předejít takovým problémům se dá použitím podmínkových formulí. Ty zamezí špatnému zobrazení prvků. Tudiž pokud se nadetekuje starý typ prohlížeče webu, stránka automaticky použije jiný zdroj CSS souboru a tím přepíše původní nastavení stránky a zjednoduší ho na zobrazitelnou podobu. (Craig Sharkie, 2013)



Graf 1- Procentuální použití jednotlivých prohlížečů (Zdroj: <http://gs.statcounter.com/>)

3.5.3 Desktop First princip

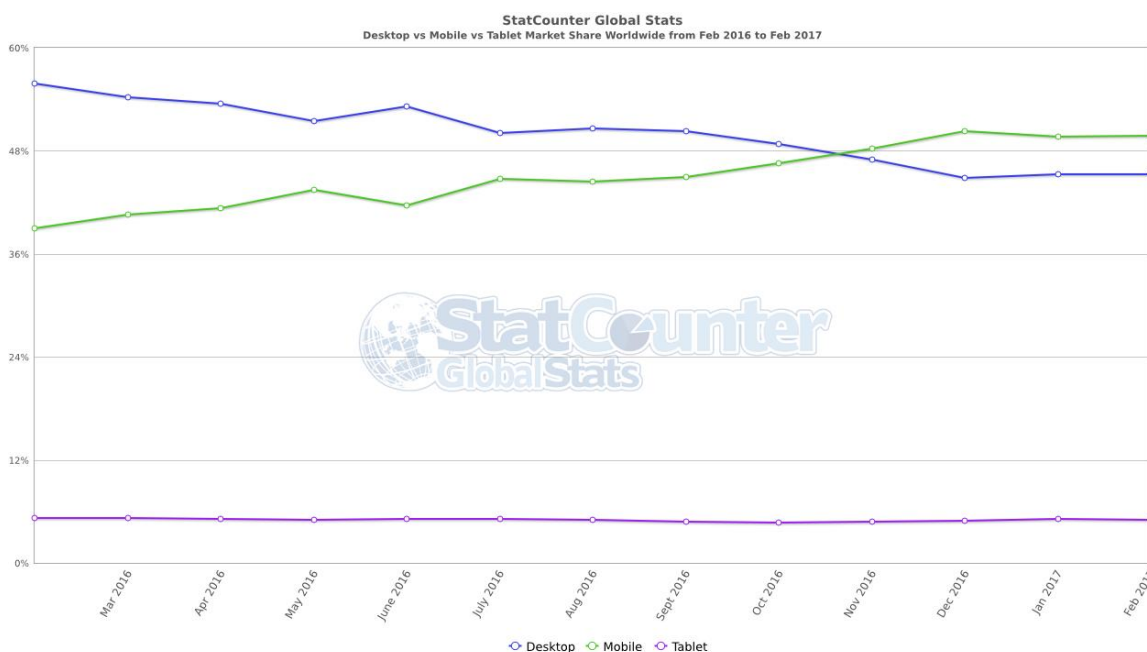
Tento způsob tvorby webu se používal ještě před příchodem chytrých telefonů. Vývojář vytvořil webovou stránku přesně na obrazovky počítačů. Většinou měla fixní šířku a hlavní záměr byl, aby se web správně zobrazil na rozlišeních, které se používaly nejvíce. Tento přístup byl dříve naprosto normální a žádný jiný ani neexistoval. S příchodem smartphonů se z tohoto přístupu stala minulost a jen staré weby jsou tvořeny touto metodou. Vychází totiž z předpokladu, že nejdříve vytvoříme stránku pro desktopové uživatele a až později upravíme k zobrazení na telefonu.

Ve výsledku vývojář vytvoří webovou aplikaci na zobrazení na stolním počítači a poté se zvolí breakpoints, které slouží pro zobrazení na tabletu a telefonu. Tyto úpravy se provádí pomocí media queries. Některé prvky jsou dokonce na menších zařízeních skrývány, kvůli obtížnosti a složité implementaci na menší zařízení. Odtud vzniklo přívlastko tohoto postupu, a to je degradační. Ze stránky se prakticky odebírají prvky až na mobilní úroveň, která logicky nemůže být plnohodnotná. Mezi nevýhody patří hlavně náročnost změn pro všechny zobrazení. Kód pak začne být příliš dlouhý a špatně přehledný. (Clark, 2012), (McFarland, 2013), (Fielding, 2014)

3.5.4 Mobile First princip

Od roku 2010 zde bylo několik změn v oboru responzivního designu. Z počátku byly webové stránky navrhovány tak, aby co nejlépe vypadaly na stolních počítačích. Avšak po velkém rozšíření mobilních zařízení, známých jako smartphone, se priority začaly měnit.

V tomto období byl přístup na web skrz mobilní zařízení velice těžkopádný. Hlavní problém byl v pomalé rychlosti připojení a příliš mnoho informací, které měly být zobrazovány na displeji mobilního telefonu. Dnešní trend však ukazuje, že se oblíbenost mobilních zařízení pro sledování webového obsahu zvyšuje a v roce 2016 dokonce překročila hranici stolních počítačů.



Graf 2 - Prodejnost zařízení dle typu zařízení (Zdroj: <http://gs.statcounter.com/>)

Z Grafu vidíme, že nárůst mobilních zařízení je opravdu značný, a to značí i větší popularitu těchto zařízení. Pohledem na graf se může tvrdit, že používání smartphonů pro prohlížení webu bude stále vysoké, nebo bude ještě růst. Pro vývojáře je doporučeno začínat vyvíjet aplikace právě od mobilní platformy a poté přikročit k ostatním platformám.

K plánování mobilního designu pro webové rozhraní slouží několik pravidel, kterými by se měl vývojář řídit. Je to hlavně pochopení, co uživatel od stránky očekává a

jaké informace chce vidět jako první. Vytvářet design pro menší či střední displeje je těžká práce, protože se musí vybrat nejdůležitější prvky stránky a ty s prioritou umístit nejvíce na oči. Hledět při tom na ovládání zařízení, jeho možnosti a zároveň znát jeho hranice. (Clark, 2012), (McFarland, 2013), (Craig Sharkie, 2013), (Peterson, 2014)

3.5.5 Content First princip

Dnešní webové stránky jsou stavěny pro specifické viewporty. Nicméně variací těchto stránek přibývá velmi rychlým způsobem, díky mnoha zařízením s různými rozměry. Při tomto přístupu je nejdůležitější, jaký obsah uvidí uživatel jako první. Design na stolní počítač by musel uživatel zoomovat a hledat co potřebuje a tomu je nutné se v responzivním designu vyhnout.

Responzivní design se snaží zobrazit co nejvíce relevantní data. Bohužel se stejný viewport nedá aplikovat skrz stejná rozložení webu. Na druhou stranu je důležité se zaměřit na zprostředkování obsahu a snadnosti dostat se ke správnému obsahu uživatelem.

Tento obsah může být skoro vše, například text, obrázek, či video. Uživatel tento obsah očekává hned a co nejpřehledněji. Tudiž se responzivní web nezabývá jen rozložením webu a media queries, ale také jaký obsah zobrazuje v určitých situacích.

Toto se vztahuje k otázce vzhledu a použitelnosti. Cílem webové stránky je předat informaci a je nutné najít kompromis mezi funkčností webové aplikace a její krásou. Pokud například uživatel s pomalým připojením hledá kontakt na webu a načítá se mu několik obrázku na hlavní straně, je pro něj web nepoužitelný. V této situaci uživatele nezajímá, jak se chovají tlačítka, když na ně klikne, nebo zda má menu animaci rozrollování. (Peterson, 2014)

3.5.6 Optimalizace výkonu webové stránky

Další z důležitých aspektů responzivity je použitelnost z hlediska výkonu. Pro uživatele přistupujícího na webovou stránku je tato věc stěžejní. Podle společnosti Google způsobuje pomalá odezva a prodlevy až o 0.6 % méně aktivity uživatele. Návštěvníci přistupující z mobilního telefonu často odchází z webové stránky, pokud načítání trvá příliš dlouhou dobu. Výkon webové prezentace by se měl stavět do popředí, pokud jde o vývoj stránky.

Dnešní situace už nám zajišťuje rychlé připojení i na mobilním zařízení, avšak ne všude je tomu stejně. Například severní část Evropy má mnohem vyšší průměrné rychlosti nežli rozvíjející se země, kde je internet stále velmi pomalý. Tudíž je nutno uvážit jaká bude cílová skupina uživatelů webové prezentace. Stránka, která se chová skvěle na rychlém připojení, může působit naprosto opačně při nízké rychlosti internetu. Tyto optimalizace je možné provádět pomocí zmenšení nebo odebrání prvků jako obrázky, komprese obrázků, zbytečné stahování souborů v pozadí, odpovídání kvůli protokolům. Dále se dá webová stránka optimalizovat skrz HTML a CSS soubory, které by měly být co nejmenší pro nejvyšší účinek. (Craig Sharkie, 2013)

3.6 Frameworky

3.6.1 Bootstrap

Bootstrap je opensource webový Framework, který umožňuje vytvářet webové stránky a aplikace. Skládá se z HTML a CSS souborů. Popřípadě obsahuje i sadu písma a ikon. Je v něm obsaženo několik přednastavených prvků jako formuláře, tlačítka, odkazy, navigační menu a dokonce i JavaScriptové rozšíření.

Bootstrap byl původně nazvaný jako Twitter Blueprint. Sloužil nejdříve jako nástroj pro samotný Twitter. Později se developři rozhodli uvést tento způsob stavby webu mezi veřejnost a přejmenovali ho na dnešní jméno.

S příchodem nové verze Bootstrapu s pořadovým číslem 4 je očekáváno několik změn. Mezi hlavní patří zrušení podpory pro některé verze Internet Exploreru a starý iOS verze 6. Přejít z jednotek pixelů na jednotky rems. Zvětšení globální velikosti písma na 16px z 14px. Několik prvků bude zrušeno nebo budou nahrazeny modernější verzí. („Bootstrap“, 2014)

3.6.2 Foundation

Tento Framework byl vytvořen společností ZURB. Je to jeden z nejrozšířenějších aktuálních frontend Frameworků. Od verze 2.0 je zdrojový kód licencován pod značkou open-source. Je plně semantický, podporuje Mobile First přístup, tudíž můžete vytvářet webovou stránku přímo od mobilního zařízení.

Pro vývojáře nabízí čtyři možnosti vývoje, a to namely complete, essential, custom a sass. Každý si pak může vybrat, co je pro jeho potřeby nejlepší a shodující se s aktuálním projektem. Nabízí jednoduchý systém mřížky s jednoduchým pojmenováním. („Foundation”, 2016)

3.6.3 Skeleton

Skeleton je velice lehký Framework vytvořený Davem Gamachem. Jedná se prakticky o dva CSS soubory, které obsahují všechny potřebné prvky pro tvorbu webu. Je velmi podobný dvou zmíněným Frameworkům. Používá stejný přístup k tvorbě, a to princip Mobile First, což znamená, že se stránka tvoří z mobilního zařízení a poté až na desktopové zařízení. Sám se prezentuje jako „dead simple“, možno přeložit jako jednoduchý jako smrt. Není tak bohatý v CSS nastaveních jako jeho dva konkurenti, vývojáři stráví více času při jeho nasazování na web. („Getting Started with Skeleton, the Simple CSS Boilerplate”, 2015)

3.6.4 Porovnání

Vzhledem k velké podobě zmiňovaných Frameworků jsem se rozhodl porovnat je pomocí vícekritériálního rozhodování s vahami. Nejprve si stanovím kritéria, která se budou vázat ke každému Frameworku. Poté si zvolím váhy jednotlivých kritérií podle významnosti každého z nich. Dalším krokem je sestavení tabulky a obodování každého kritéria od 1-10, přičemž 10 je nejvíce vyhovujících a 1 nejméně vyhovující.

Framworl	Uživatelská komunita	Předpřipravené prvky	Rychlost implementace	Srozumitelnost prvků	Body s vahami	Pořadí
Bootstrap	9	8	7	5	7,9	1.
Foundation	6	8	7	4	6,6	2.
Skeleton	3	6	8	8	5,4	3.
Váhy	0,4	0,3	0,2	0,1	1	

Tabulka 2- Porovnání frameworků vícekritériální metodou

V následující tabulce mám zanesené výsledné body pro konkrétní Frameworky. Výsledkem skalární součet vah a bodů a následné vyhodnocení variant. Uživatelská komunita má v tomto případě vysokou váhu z důvodu velmi rychlého vyhledání prvků, či řešení daného problému, které jinak zabere několik hodin až dnů řešení. Naopak nejnižší váhu má srozumitelnost prvků. Poté co s Frameworkem začneme opravdu pracovat, neznámý prvkem si vyhledáme a poté už ho používáme repetitivně a není tudíž nutností ho příště znovu vyhledávat.

Výsledkem mé tabulky je tedy největší perspektivnost a využitelnost Frameworku Bootstrap, za ním je na druhém místě Framework Foundation a na poslední příčce je Skeleton.

4 Vlastní práce

Cílem této bakalářské práce je vytvořit responzivní web s použitím vhodného Frameworku. Zobrazit tím principy a základy tvorby webových stránek. Tomuto předchází několik fází od prvního náčrtu stránky, až po zhotovení. Stránka musí být pohyblivou šířku a výšku, media by měla reagovat na změnu velikosti a měla by být použita CSS media queries.

4.1 Návrh

První krok k vytvoření webové stránky je stanovení představy o tom, jak by stránka mohla vypadat a fungovat. Zde je potřeba přihlédnout k několika aspektům. Pro mé účely jsem se rozhodl vytvořit stránku, která má sloužit jako prezentace autoservisu. Stránka se vyznačuje tím, že je takzvaně Jednostránková. Uživatel tak neproklíkává a nenačítá ostatní stránky zvlášť, ale obsah se kumuluje za sebou a po kliknutí na určitou sekci se uživateli daný oddíl odscrolluje. Jedná se o dnes velmi populární variantu tvorby webových stránek.

Webová stránka autoservisu by měla poskytovat relevantní informace a obsah, který zajímá uživatele. Obsahem bude jednoduché představení autoservisu, poté se uživatel dozví, jakým odvětvím se servis zabývá a jaké poskytuje služby. Další částí je galerie obrázků, která slouží uživateli jako vyobrazení pracovního nasazení servisu a zároveň je zde prezentováno prostředí autoservisu. V poslední sekci jsou veškeré kontakty na autoservis, včetně adresy.

4.1.1 WireFrame

Před každým návrhem stránky je potřeba vytvořit takzvaný wireframe, což je jednoduchý nákres dané webové stránky. Je spoustu způsobů, jak tohoto dosáhnout. Je možné wireframe nakreslit tužkou na papír, vystříhnout z papíru, či vytvořit ve webové aplikaci. Pro mé účely jsem zvolil třetí variantu a vybral webovou stránku www.wireframe.cc, která nabízí přehledné, jednoduché, avšak plně dostačující možnosti tvorby wireframu. Je zde možné vkládat text, vytvářet obrazce, popřípadě vkládat rozdělovníky.

Návrh je zde rozdělený na několik variant. Jedná se o návrh na mobilní zařízení, návrh na tablet a návrh na desktopové prohlížeče. Pro každé zařízení bude mít stránka trochu jiné zobrazení a je nutné s tímto počítat již u návrhu.

Zvolil jsem si způsob Mobile First, který je popsán v teoretické části. Tudíž jsem si jako první navrhl design na mobilní zařízení. Menu bylo umístěno v rolovacím prvku na pravé horní straně obrazovky. Dále se zde nachází hlavní stránka, která má obrázek dílny, či automobilu na pozadí. Obrázek musí být flexibilní na všech zařízeních. Na obrázku návrhu pro mobilní zařízení můžeme vidět rozdíl oproti desktopové aplikaci v rozložení některých prvků. V sekci Služby se jednotlivé služby řadí pod sebe, místo vedle sebe, díky zúženému prostoru. Vypovídací hodnota této sekce však zůstává stejná. Dále je zde galerie, kde se obrázky řadí také pod sebe, pokud nemají vedle sebe dostatečné místo. Toto může vést až k zobrazení pouze jednoho obrázku na řádku, ovšem všechny obrázky budou zobrazené, tudíž uživatel nebude ochuzen o obsah.

Autoservis Novák

Nejlepší autoservis v okolí

O nás

Lorem ipsum dolor sit amet et delectus
accommodare his consul copiosae legendos at vix
ad putent delectus delicata usu. Vidit dissentiet eos
cu eum an brute copiosae hendrerit. Eos erant
dolorum an. Per facer affert ut. Mei iisque mentitum

Služby

Lorem ipsum dolor sit amet et
delectus accommodare his consul
copiosae legendos at vix ad putent

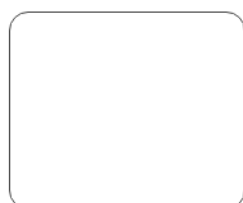
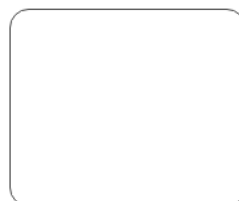
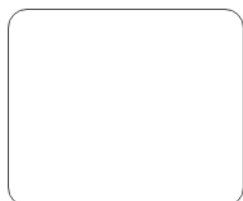
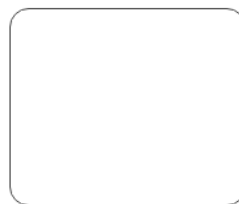
Lorem ipsum dolor sit amet et
delectus accommodare his consul
copiosae legendos at vix ad putent

Lorem ipsum dolor sit amet et
delectus accommodare his consul
copiosae legendos at vix ad putent

Lorem ipsum dolor sit amet et
delectus accommodare his consul
copiosae legendos at vix ad putent

Obrázek 4- Wireframe mobilní verze webové stránky - část 1

Galerie



Kontakt

Lorem ipsum dolor sit amet et delectus accommodare his consul copiosae legendos at vix ad putent delectus delicata usu. Vidit dissentiet eos cu eum an brute copiosae hendrerit. Eos erant dolorum an. Per facer affert ut. Mei iisque mentitum moderatius cu. Sit munere facilis accusam eu dicat falli consulatu at vis. Te facilisis mnesarchum qui posse omnium mediocritatem est cu. Modus argumentum ne qui tation efficiendi in eos. Ei mea falli legere efficiantur et tollit aliquip debitis mei. No deserunt mediocritatem mel. Lorem ipsum dolor sit amet et delectus accommodare his consul copiosae legendos at vix ad putent delectus

Telefon

Facebook

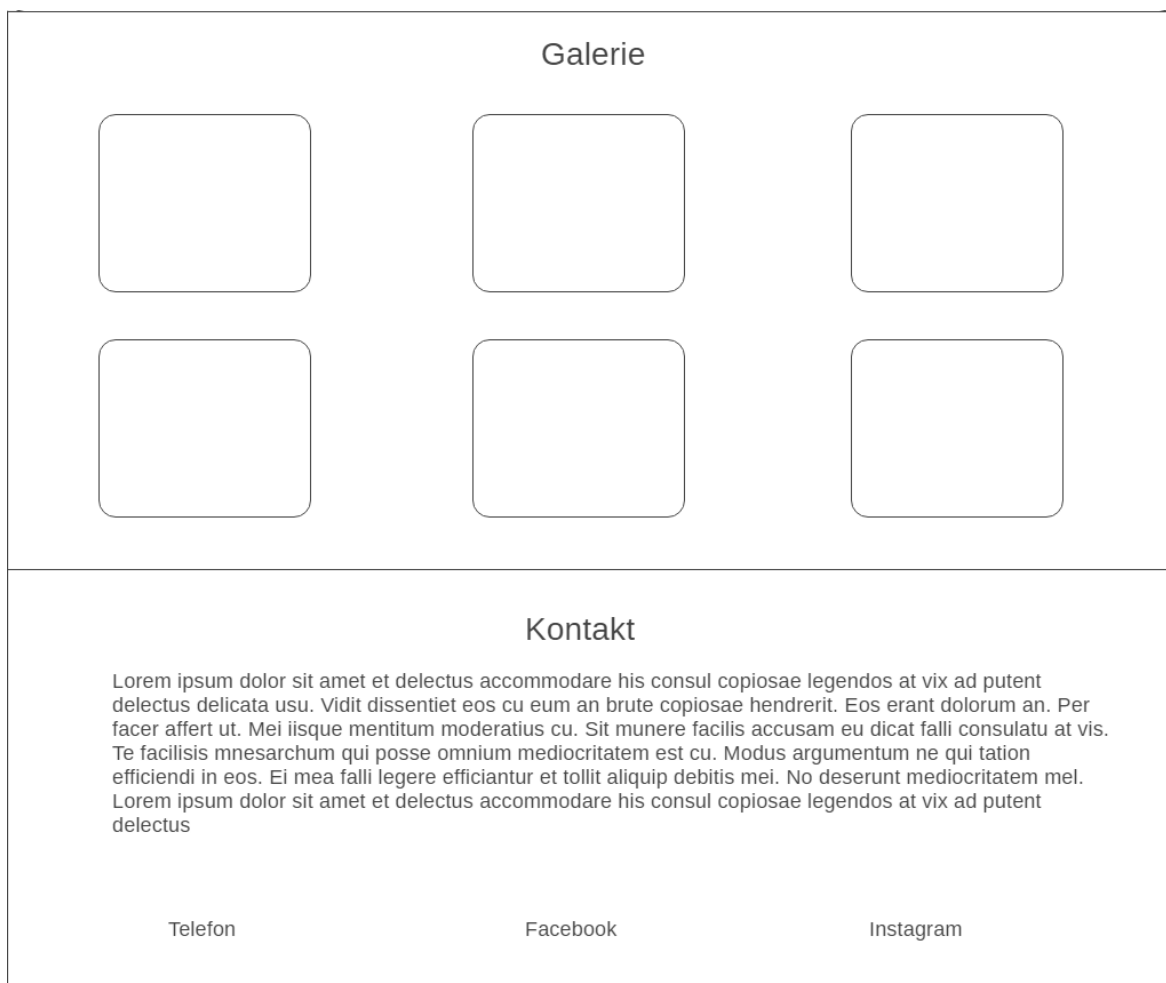
Instagram

Obrázek 5- Wireframe mobilní verze webové stránky - část 2

Verze pro stolní počítače vypadá dost podobně, ale skrývá několik změn. Zde máme více prostoru na zobrazení veškerých prvků. Horní lišta tvoří název autoservisu společně s navigací, která obsahuje položky „O nás, Služby, Galerie, Kontakt“. Úvodní stránka poskytuje pohled na základní info – název a slogan autoservisu. Stránka „O nás“ obsahuje krátký odstavec, který popisuje původ a účel autoservisu. Další položkou jsou služby. Zde je stránka rozložena do 4 sloupců, při čemž v každém sloupci je služba. Na stránce se v další sekci objevuje galerie, která čítá dohromady 6 nejaktuálnějších fotek. Zde jsou použity 3 sloupce, které rozdělují galerii na 2 řádky. Na konci stránky se nacházejí kontakty, kde je stručný popis ohledně otevírací doby a domluvy se zákazníkem. Dále můžeme vidět odkazy na sociální media a telefon, které jsou rozděleny také do tří sloupců.



Obrázek 6- Wireframe desktopové verze webové stránky - část 1



Obrázek 7- Wireframe desktopové verze webové stránky - část 2

4.2 Kódování

Pokud jsme rozhodnutí vytvářet webovou aplikaci, musíme si připravit několik věcí. Mimo wireframe návrh je potřeba stáhnout Framework, s kterým budeme pracovat. V našem případě se mluví o Bootstrapu. Celý Framework je tvořen základem, což je CSS soubor, společně provázaný s HTML souborem. Další doplňky jsou tvořeny Javascriptem a různými doplňkovými písmy a ikonami.

Nejdříve si tedy stáhneme potřebné soubory pro práci s Frameworkem. Poté, co máme CSS soubor, je nutné zkontrolovat, jestli je správně svázaný s HTML souborem a to tagem `<link>`. Některé Frameworky používají dva soubory CSS, při čemž jeden tvoří samotný Framework ve kterém jsou všechny prvky a deklarace a druhý je naopak prázdný a je nalinkovaný až po prvním. Do druhého souboru pak vkládáme změny oproti původnímu souboru a máme tak přehledně oddělené úpravy na specifickou stránku a nemusíme hledat v původním souboru. Tento způsob je výhodný při realizaci velkých

webů. Poté prohlížeč postupuje od shora dolů a hodnoty v původním CSS souboru jsou přepsány novým, upraveným. Dále je možné nalinkovat písma, která nejsou v původní sadě.

```
<!-- Link na CSS soubor -->
<link href="vendor/bootstrap/css/bootstrap.css" rel="stylesheet">
<!-- Link na písmo-->
<link href='https://fonts.googleapis.com/
css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800' rel='stylesheet' type='
text/css'>
```

Obrázek 8- Spojení CSS souborů s HTML souborem

Nyní přistoupíme k deklaraci navigační lišty, která se nachází na horním okraji stránky. Pomocí tagu `<nav>` podporovaného v nejnovější verzi HTML vytvoříme menu a vložíme do něj několik položek. Nyní je čas na využití možností frameworku, a tak můžeme najít na webu bootstrapu možnost, jak vytvořit takovéto menu. Menu by mělo být na horním okraji, tudíž použijeme variantu „Fixed to top“ z webu Bootstrapu.

Fixed to top

Add `.navbar-fixed-top` and include a `.container` or `.container-fluid` to center and pad navbar content.

```
Brand Home Link Link
<nav class="navbar navbar-default navbar-fixed-top">
  <div class="container">
    ...
  </div>
</nav>
```

Obrázek 9- Kód pro vytvoření navigační lišty, která má svou polohu na horní části stránky. (Zdroj: <https://getbootstrap.com/components/#navbar>)

Po tomto kroku naplníme menu položkami, které jsme si dříve nadefinovali a přidáme tlačítko pro případ, že bude zobrazení příliš úzké na vykreslení plnohodnotného menu.

```

<nav class="navbar navbar-default navbar-fixed-top">
  <div class="container-fluid">
    <!-- Tlačítko zobrazované na menších displayích -->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1">
        <span class="sr-only">Toggle navigation</span> Menu <i class="fa fa-bars"></i>
      </button>
      <a class="navbar-brand page-scroll" href="#page-top">BPM Motorsport</a>
    </div>

    <!-- Položky menu -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
      <ul class="nav navbar-nav navbar-right">
        <li>
          <a class="page-scroll" href="#about">O nás</a>
        </li>
        <li>
          <a class="page-scroll" href="#services">Služby</a>
        </li>
        <li>
          <a class="page-scroll" href="#portfolio">Galerie</a>
        </li>
        <li>
          <a class="page-scroll" href="#contact">Kontakt</a>
        </li>
      </ul>
    </div>
  </div>
</nav>

```

Obrázek 10 - Kód navigační lišty a menu (Bootstrap)

Pro zajímavost je zde vidět kód, potřebný k podobnému výsledku ve Frameworku Foundation a také řešení dle Skeletonu. Je vidět, že kód Skeletonu je nejkratší, ale zároveň je nutno brát v potaz, že je náročnější na správu díky duplicitám pojmenování.

```

<div class="title-bar" data-responsive-toggle="example-menu" data-hide-for="medium">
  <button class="menu-icon" type="button" data-toggle></button>
  <div class="title-bar-title">Menu</div>
</div>

<div class="top-bar" id="example-menu">
  <div class="top-bar-right">
    <ul class="menu">
      <li class="menu-text">Foundation Responsive <span class="fa fa-bars"></span>
    </ul>
  </div>
  <div class="top-bar-left">
    <ul class="medium-horizontal vertical dropdown menu" data-responsive-menu="accordion medium-dropdown">
      <li><a href="#">O nás</a></li>
      <li><a href="#">Služby</a></li>
      <li><a href="#">Galerie</a></li>
      <li><a href="#">Kontakt</a></li>
    </ul>
  </div>
</div>

```

Obrázek 11- Kód navigační lišty a menu (Foundation)

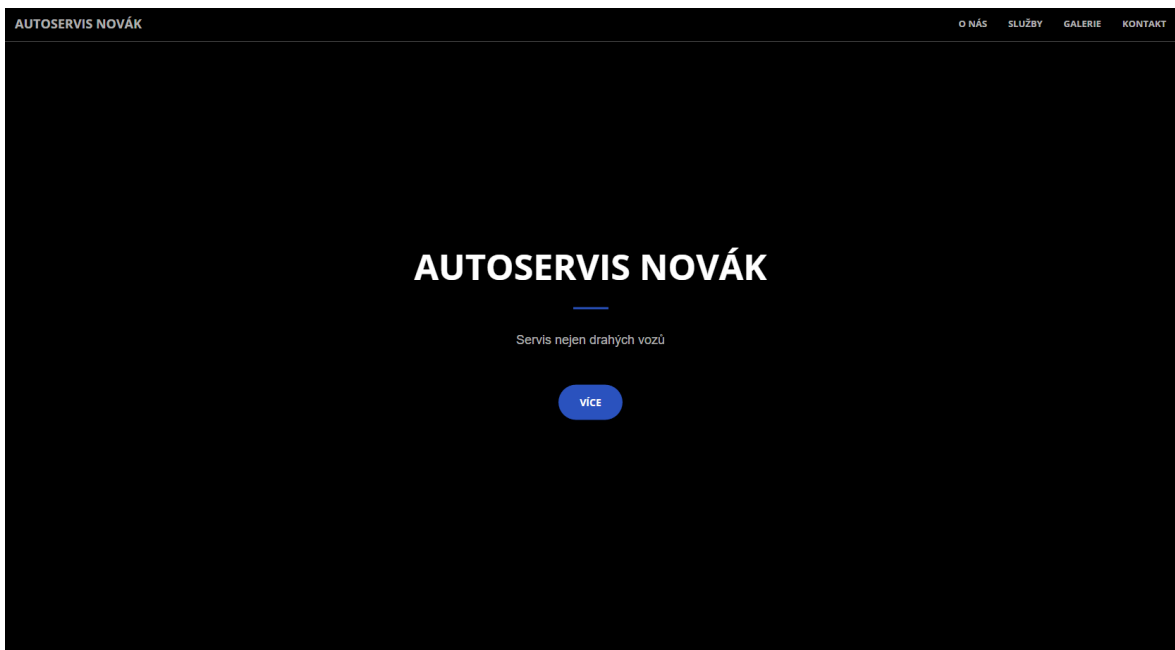
```

<div class="band navigation">
  <nav class="primary container">
    <ul>
      <li><a href="#">O nás</a></li>
      <li><a href="#">Služby</a></li>
      <li><a href="#">Galerie</a></li>
      <li><a href="#">Kontakt</a></li>
    </ul>
    <select><option selected="selected" value="">Go to...</option><option value="#">O nás</option><option value="#">
    Služby</option><option value="#">Galerie</option><option value="#">Kontakt</option>
  </nav>
</div>

```

Obrázek 12 - Kód navigační lišty a menu (Skeleton)

Nyní vytvořím úvodní stranu, která má na pozadí obrázek a v popředí je název autoservisu a slogan. Takto stylizovaná úvodní strana je tvořena pomocí tagu <header>, který se používá pro deklarování úvodních informací a zobrazuje většinou jen několik málo textů, či obrázků. Zde přiřadíme správné třídy CSS souboru dle specifikací frameworku a doplníme texty. Nyní máme Hlavní stranu s navigační lištou a vše již plně responzivní. Nyní webová stránka vypadá jako na obrázku č. 13.



Obrázek 13 - Hlavní strana po vytvoření navigační lišty a headeru

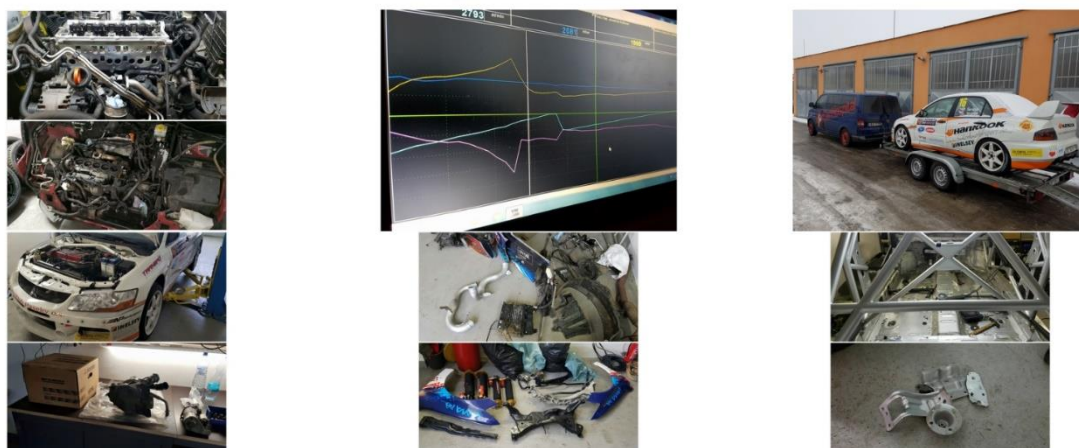
Dalším krokem je vytvoření responzivní sekce s výčtem služeb. Zde musíme vytvořit 4 sloupce, které reprezentují služby. Nejdříve vytvoříme řádek, do kterého zapouzdříme všechny sloupce. Zde určíme, že počet sloupců je 4, v případě velkého displeje, na který se tento obsah vejde. V tomto případě se jedná o šířku zobrazení větší než 992px. Toto nastavení je zprostředkované speciální třídou col-lg-3, která přímo určuje,

že sloupec bude rozdělen po třech jednotkách. Vzhledem k tomu, že celá šířka rozložení tvoří 12 těchto jednotek, vytvoří se 4 sloupce. Dále zde využijeme takzvaný service-box, což je styl určený přímo pro vkládání podobných formátů obsahu.

```
<div class="col-lg-3 text-center">
  <div class="service-box">
    <h3>Úpravy řídicích jednotek</h3>
    <p>Úpravy řídicích jednotek na míru celému vozu.</p>
  </div>
</div>
```

Obrázek 14 - Kód pro jednu položku služby

Důležitou součástí tohoto konkrétního webu je galerie. Byla navržena pro zobrazování 6 nejaktuálnějších obrázků. Na mobilním zařízení se obrázky budou zobrazovat pod sebou, na větším displeji ve dvou sloupcích a na plnohodnotných zařízeních se zobrazí tři sloupce a dva řádky obrázků. Tohoto opět docílíme přiřazením správného čísla rozložení dle frameworku. Na úložiště webové stránky pak vložíme dvě velikosti obrázků. Jednu náhledovou a druhou plnohodnotnou, kterou uživatel uvidí poté, co obrázek rozklikne. Obrázková galerie bude vypadat podobně jako na obrázku č. 15.



Obrázek 15 - Zobrazení galerie

Posledním krokem k dokončení kódování stránky je vytvořit sekci Kontakt, která obsahuje pouze text a pod textem se nachází kontakty umístěné ve třech sloupcích, které jsou opět plně responzivní. K takovému výsledku je možno dostat se opět deklarací velikosti jednotlivých sloupců. Každému sloupci tedy přiřadíme šířku 4 jednotek, aby dohromady tvořily 12 jednotek. Dále můžeme přidat několik ikon, které přidají na přehlednosti a srozumitelnosti zobrazených informací.

```

<div class="col-lg-4 text-center">
  <i class="fa fa-envelope-o fa-3x sr-contact"></i>
  <p><a href="mailto:your-email@your-domain.com">ServisNovak@gmail.com</a></p>
</div>

```

Obrázek 16 - Kód vztahující se ke kontaktům

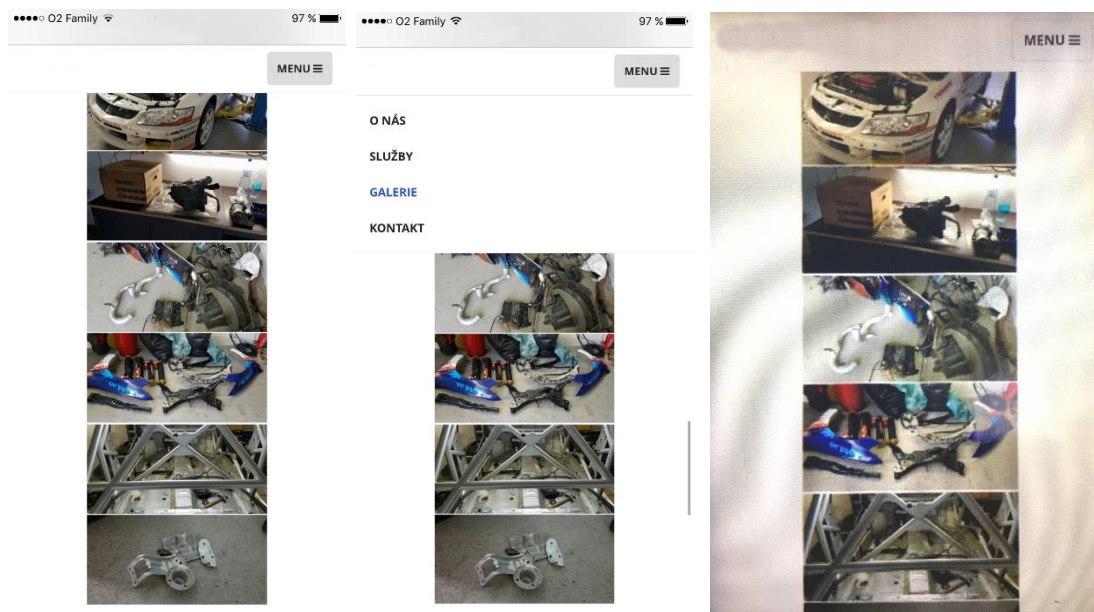
4.3 Testování

Testování webové stránky je velmi rozsáhlé téma. V mém případě je důležité otestovat funkčnost stránky v závislosti na používaném zařízení a prohlížeči. Vytvořenou webovou aplikaci tedy otestuji na několika platformách.

Nejcitlivější část webové stránky je určitě galerie, díky zobrazování multimediálního obsahu a zarovnání do sloupců. Proto jsem vybral právě tuto část a pokusím se ji otevřít na co nejvíce zařízeních.

4.3.1 Mobilní zařízení

Na obrázku níže můžeme vidět obsah, který se přizpůsobil obrazovce mobilního zařízení a zároveň funkčnost navigačního menu, které je schované v dropdown tlačítku.

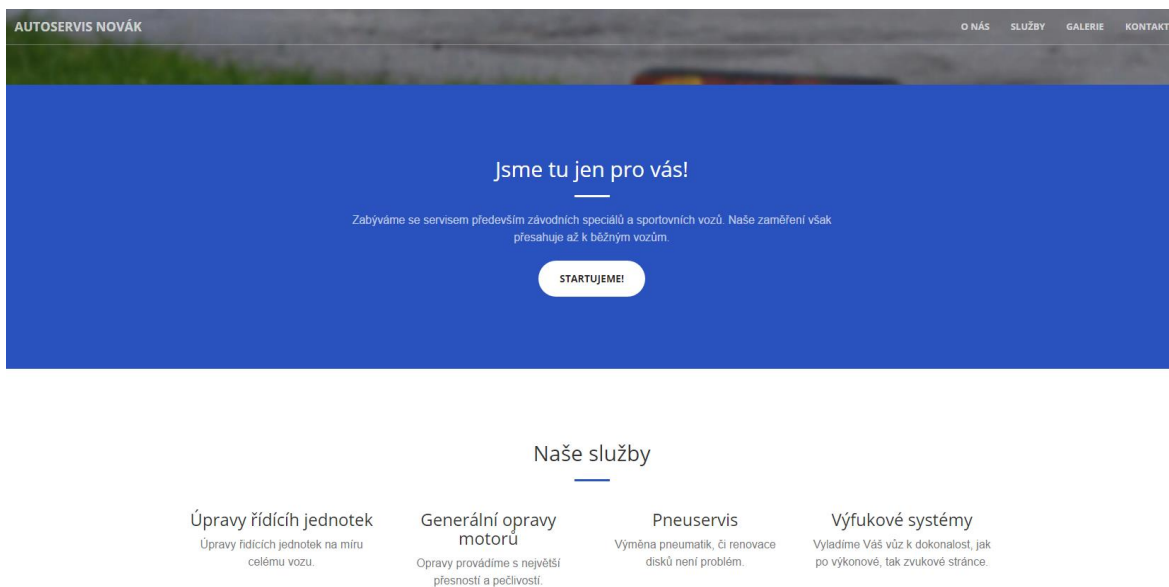


Obrázek 17 - Zobrazení webové stránky na mobilních zařízeních (iPhone 6s, iPhone 6s, Nokia Lumia 520)

Testován byl iPhone 6s s výchozím prohlížečem Safari. (první dvě části obrázku) Poté jsem stránku otevřel na starší Nokii Lumii 520, kde je výchozím prohlížečem Internet Explorer, který nebyl nejaktuálnější, a i přesto se stránka zobrazila správně a měla všechny funkčnosti.

4.3.2 Desktopová zařízení

Testování jsem prováděl na několika zařízeních a webových prohlížečích. Na platformě Microsoft Windows jsem nainstaloval nejnovější verze používaných prohlížečů jako Mozilla Firefox, Opera, Google Chrome a také je doplnil o výchozí prohlížeč Internet Explorer. Na všech těchto prohlížečích se stránka chovala naprosto předvídatelně a nedošlo k žádnému problému se zobrazením.



Obrázek 18 - Zobrazení webové stránky v prohlížeči Mozilla Firefox

5 Závěr

Hlavní cíl předkládané bakalářské práce byl splněn. Na jednotlivé dílčí cíle byly získány tyto odpovědi – výsledky.

5.1 Charakterizovat technologie pro tvorbu webových aplikací

Mezi technologie pro tvorbu webových aplikací patří několik prvků. Mezi nejzásadnější se řadí hypertextový jazyk HTML5 a k tomu soubor kaskádových stylů CSS ve verzi 3. Dále je zde mnoho pomocných technologií, které přímo i nepřímo ovlivňují vzhled a funkčnost stránky. Mezi tyto prvky se řadí například Javascript, či dnes už méně používaný Flash.

Ke tvorbě se používají takzvané Frameworky, které z velké míry usnadňují práci, a hlavně je jimi zamezeno opakovanému psaní stejného kódu. Dále již obsahují podporu mnoha prohlížečů, a tak vývojář nemusí řešit další nastavení.

5.2 Analyzovat požadavky na responzivní design

Požadavky na dnešní web design jsou velmi striktní a měly by být dodrženy v plné míře. Webová stránka musí být zobrazitelná prakticky na libovolném zařízení bez problémů se zobrazováním grafického obsahu, či videa. Jedná se o zobrazení od mobilních zařízení až po velkoplošné televize. Obsah takové webové aplikace musí být vždy čitelný a měl by být plnohodnotný. Uživatelé menších zařízení by neměli být ochuzeni o informace. Mohou očekávat maximálně ztrátu grafických prvků a animací. Nejmodernější způsob tvorby webové stránky je principem Mobile First, který zaručí správné zobrazování na mobilních zařízeních. Nejprve se navrhne grafický koncept pro mobilní zařízení a poté z něho vyplyne vzhled stránky na desktopové zařízení.

5.3 Porovnat programová řešení (frameworky)

Pro porovnání Frameworků bylo zvoleno několik nejpoužívanějších zástupců tohoto oboru. Jedná se o Framework Bootstrap, Skeleton a Foundation. Srovnání bylo prováděno pomocí několika faktorů, mezi které patří například rychlost práce s Frameworkem, či uživatelská podpora. Nejperspektivnější Framework z mé studie je Bootstrap. Jedná se o spojení uživatelské přívětivosti, velmi jednoduchého a intuitivního vkládání prvků a zároveň zažitého systému sloupcování stránek. Podobné vlastnosti mají i

dva ostatní Frameworky, avšak Bootstrap má neskutečnou výhodu v obrovské komunitě vývojářů a uživatelů tohoto Frameworku. Pokud zde nastane problém, nebo je potřeba vytvořit nějaký nový prvek, je velmi pravděpodobné, že na internetových stránkách najdete řešení daného problému.

5.4 V praktické části vytvořit responzivní webovou aplikaci

V praktické části bakalářské práce jsem vytvořil funkční webovou aplikaci, která splňuje dnešní požadavky na responzivní design. Jedná se o webovou stránku, která má několik základních a běžných prvků. Webová stránka prezentuje blíže neurčený autoservis. Zároveň je tvořena pomocí principu Mobile First, kdy byl nejdříve vytvořen design webové stránky zobrazující se na mobilním zařízení a poté se z takového prototypu začal tvořit design webové stránky zobrazující se na stolních počítačích.

Webovou stránka byla tvořena s pomocí programu Sublime Text 3, který také ulehčuje práci a nabízí různé funkce našeptávače, mnoho plugin modulů apod. V praktické části byly modulárně testovány všechny zmiňované Frameworky, avšak základ tvořil Framework Bootstrap. Bylo použito externí písmo a několik Javascriptů na zobrazení animovaného obsahu a vzhlednější funkčnost stránky.

5.5 Formulovat obecné i specifické závěry

Responzivní design je jedno z nejzajímavějších témat v poli internetové problematiky. Je zde mnoho benefitů, ale zároveň se k němu váže i složitost a určitá časová náročnost. Ovšem pokud se chceme držet nejnovějších trendů a držet svou technologickou znalost na určité úrovni, je potřeba tyto principy dodržovat. V mé studii bylo zjištěno, že pomocí Frameworku Bootstrap a principem Mobile First je možné vytvořit plně responzivní webovou aplikaci, která se zobrazuje správně na všech zařízeních.

6 Bibliografie

1. *Bootstrap*. (2014). Získáno 2017-03-13, z Bootstrap: <http://getbootstrap.com/>
2. Clark, R. (2012). *Beginning HTML5 and CSS3: the Web evolved : next generation Web standards* (First. vyd.). New York: Distributed to the book trade worldwide by Springer Science Business Media.
3. Craig Sharkie. (2013). *Jump Start responsive web design* (First. vyd.). VIC Australia: SitePoint Pty. Ltd.
4. Duckett, J. (2008). *Beginning web programming with HTML, XHTML, and CSS* (2nd ed.. vyd.). Indianapolis, IN: Wiley Pub.
5. Fielding, J. (2014). *Beginning responsive web design with HTML5 and CSS3 / Jonathan Fielding* (First. vyd.). New York: Apress.
6. *Foundation*. (2016). Získáno 2017-03-13, z <http://foundation.zurb.com/showcase/about.html>
7. Gasston, P. (2014). *Book of css3: a developer's guide to the future of web design* (2nd Edition. vyd.). S.l.: O'Reilly Media.
8. *Getting Started with Skeleton, the Simple CSS Boilerplate*. (2015). Získáno 2017-03-13, z SitePoint: <https://www.sitepoint.com/getting-started-with-skeleton-simple-css-boilerplate/>
9. Hogan, B. (2010). *HTML5 and CSS3: develop with tomorrow's standards today* (First. vyd.). Raleigh, N.C.: Pragmatic Bookshelf.
10. MacDonald, J. (2013). *HTML5: the missing manual : the book that have been in the box* (2nd Edition.. vyd.). USA: O'Reilly Media.
11. Marcotte, E., & [foreword by Jeremy Keith]. (2011). *Responsive web design* (First. vyd.). New York: A Book Apart.
12. McFarland, D. (2013). *CSS3: the missing manual* (3rd ed.. vyd.). Sebastopol, CA: O'Reilly.
13. Peterson, C. (2014). *Learning responsive Web design: a beginner's guide* (First edition.. vyd.). USA: O'Reilly Media.
14. Powers, D. (2012). *Beginning CSS3* (First. vyd.). New York, NY: Distributed by Springer Science Business Media.
15. Wroblewski, L. (2011). *Mobile first* (First. vyd.). New York: A Book Apart.

7 Zdroje obrázků a tabulek

[Obrázek 1] - https://www.w3schools.com/html/html_responsive.asp

[Obrázek 2] - <http://www.speeli.com/articles/view/Css-media-queries-for-standard-devices>

[Obrázek 3] - <http://electronicdesign.com/whats-difference-between-html-4-and-html-5>

[Obrázek 9] - <https://getbootstrap.com/components/#navbar>

[Tabulka 1] - <http://www.rapidtables.com/web/dev/screen-resolution-statistics.htm>