



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Platforma průmyslové spolupráce

CZ.1.07/2.4.00/17.0041

Název

Metodiky návrhu a realizácie mobilného webu

Popis a využití

- studijní materiál pro úvod do designování mobilního webu
- výuka: vývoj pro mobilní platformy

Jazyk textu

- slovenský

Autor (autoři)

- Monika Lukáčová

Oficiální stránka projektu:

- <http://lasaris.fi.muni.cz/pps>

Dostupnost výukových materiálů a nástrojů online:

- <http://lasaris.fi.muni.cz/pps/study-materials-and-tools>

Metodiky návrhu a realizácie mobilného webu

Mgr. Monika Lukáčová

Dáta sa dajú mobilným užívateľom sprístupniť rôznymi spôsobmi. Pomocou prehliadačov, ako sú dostupné všetkým ostatným užívateľom, alebo pomocou aplikácii, ktoré umožňujú plné využitie schopností daného zariadenia s maximálnou mierou použiteľnosti a špecifickosťou pre konkrétnu platformu. Nasledujúci text sa zameriava na prvý variant, ktorý je univerzálny, prístupný na ľubovoľnom zariadení podporujúcom prehliadanie webu a najmenej zložitý čo sa týka učenia nových technológií a programovacích jazykov. Ako vytvoriť webové prezentácie s univerzálnym charakterom popisuje najmä metodika prispôsobivý webdesign a responzívny webdesign. Posledná metodika, mobil na prvom mieste, sa zameriava prioritne na návrh a vizuálnu prístupnosť dizajnu cieľeného pre mobilné a dotykové použitie.

1 Prispôsobivý webdesign

Prístup prispôsobivý webdesign (z angl. *Adaptive Web Design*) je tvorený koncepciou postupného zlepšovania, ktorá sa snaží neviazať na aktuálne trendy v technológiách, nové prístupy alebo vymoženosti nových prehliadačov, ale na schopnosť poskytnúť prístup k obsahu bez závislosti na zariadení. Prispôsobivý webdesign je postavený na princípe odolnosti voči chybám (z angl. *Fault tolerance*) a upozorňuje na elegantné znehodnotenie (z angl. *Graceful degradation*).

1.1 Postupné zlepšovanie

„Postupné zlepšovanie je filozofia s cieľom vybudovať zážitok, ktorý užívateľovi sprístupní obsah nezávisle od technologických obmedzení.“ [3]

Snahou postupného zlepšovania (z angl. *Progressive enhancement*) je zabezpečiť, aby boli informácie obsiahnuté vo webovej prezentácii každému dostupné a prístupné bez ohľadu na zariadenie, software alebo polohu. Keďže zobrazovanie webu mobilnými zariadeniami je odlišné, treba myslieť aj na menšiu veľkosť obrazovky, prípadne viac obmedzujúce ovládanie zariadenia pomocou prstov.

Postupné zlepšovanie svojím prístupom nemá za úlohu obmedzovať vývoj technológií a ich využitie. Inováciu podporuje, ak sa spojí s vývojom zameraným na užívateľa a nájde sa správne riešenie dostupné pre každého. Hlavné je uvedomiť si, že postupné zlepšovanie nevyžaduje dosiahnutie rovnakého zážitku zo získavania informácií v rôznych prehliadačoch, alebo rôznych zariadení, taktiež neurčuje používanie najnovších technológií.

Koncept postupného zlepšovania vytvorí v prvom rade základnú štruktúru zážitku dostupnú zo všetkých prehliadačov a na to po vrstvách pridáva lepšie zážitky pre tie prehliadače, ktoré majú lepšiu podporu pre nové technológie [4].

1.2 Odolnosť voči chybám

Schopnosť systému pokračovať v činnosti aj po výskyte nečakanej chyby označujeme ako odolnosť voči chybám. Prehliadače sú schopné ignorovať tie značky HTML a CSS, ktorým nerozumejú a pokračovať vo vykresľovaní obsahu aj po prečítaní nepochopených príkazov. Ak nie je prehliadač schopný prečítať značky jazyka JavaScript, nastane chyba a činnosť vykresľovania nepokračuje.

1.3 Elegantné znehodnotenie

„Pri elegantnom znehodnotení je pozornosť zameraná na funkčnosť webu v moderných prehliadačoch s najvyšším podielom na trhu. Testovanie webu na starších zariadeniach je až na konci zoznamu priorit, ak sa na tom zozname vôbec nachádza.“ [3]

Zameranie sa na najrozšírenejšie a na najaktuálnejšie funkcie prehliadačov bez ohľadu na slabú až žiadnu podporu v iných prehliadačoch, sa označuje elegantné znehodnotenie. Tento prístup podceňuje dostupnosť obsahu pre každé zariadenie, aj prístupnosť pre každého, kto má o obsah webu záujem.

2 Responzívny web dizajn

„Pre web je prirodzené, že je flexibilný. Dizajnér aj developer by sa mal chopiť tejto pružnosti a produkovať prezentácie, ktoré sú vďaka svojej prispôsobivosti každému prístupné.“ [1]

Responzívny webdesign (z angl. *Responsive Web Design*, skratka RWD) je koncept, ktorého autorom je Ethan Marcotte. RWD zabezpečuje schopnosť webu prispôbiť sa tak, aby bolo čítanie a navigácia počas prezerania čo najintuitívnejšie na každom zo zariadení podporujúcim jeho zobrazovanie. Prezeranie je pritom umožnené aj napriek obmedzeniam, ktoré každé zariadenie so sebou prináša. Koncept RWD tvoria tri piliere: flexibilné rozmiestnenie prvkov, flexibilné médiá a otázky o médiu, z ktorých už každý existoval pred RWD, ale spolu dokážu vytvoriť optimálny zážitok pri prehliadaní webu. Preto už nie je potrebné hovoriť o separátnej verzii prezentácie pre desktop, mobil alebo tablet, stačí mať jednu prezentáciu, ktorá sa prispôbí na každom zariadení podľa stanovených obmedzení a možností. [4]

2.1 Flexibilné rozmiestnenie prvkov

Flexibilné rozmiestnenie prvkov (z angl. *Flexible Layout* alebo *Fluid Layout*), nie je určené absolútnymi hodnotami (px, cm). Dimenzie sú určené percentami (%). Ak je obsah nastavený na 60 %, potom nezáleží, či je prezeraný na stolnom počítači, alebo mobilnom zariadení, vždy bude zaberat' 60 % danej obrazovky. Výhodou je taktiež zaručené odstránenie horizontálnej posuvnej lišty.

Dynamické písmo

Stanovenie absolútnych hodnôt pri veľkosti písma obmedzí prístup k obsahu webu znevýhodneným užívateľom, keďže veľkosť textu určená v pixeloch sa v niektorých prehliadačoch nedá zväčšiť podľa potreby slabozrakých. Ďalším problémom je rozlíšenie koncových zariadení. Keďže každé zariadenie disponuje rozdielnym rozlíšením a rôznym množstvom PPI (*Pixels per inch*), písmo o veľkosti 12 px môže byť na zariadeniach s malou obrazovkou a vysokým rozlíšením nečitateľné. [4]

Dynamickú variantu určovania veľkosti písma je v RWD možné dosiahnuť kombináciou percent a jednotky em. Nastavením písma v tele prezentácie na 100 % sa určí jeho veľkosť na prehliadačom určenú hodnotu, čo je vo väčšine prípadov ekvivalentné 16 px. Pre dosiahnutie veľkosti písma 24 px pre nadpis (h1) sa použije vzorec:

$$\text{Ciel'ová hodnota} / \text{kontext} = \text{výsledok}$$

kde cieľová hodnota je požadovaná veľkosť (24 px), kontext je veľkosť textu nadradeného elementu (16 px) a výsledok je veľkosť textu v jednotke em. Dosadením nadobudne hodnotu 1,5 em (24 px / 16 px).

Ak je výsledkom dlhé číslo, odporúča sa neskracovať ho a uviesť celú hodnotu. Každý prehliadač disponuje vlastným spôsobom zaokrúhľovania čísiel. Poskytnutie väčšieho množstva informácií prehliadaču pomôže rozhodnúť, ktorým smerom zaokrúhliť, čo zabezpečí lepšiu výsledok zobrazovania webu. [5]

Flexibilná mriežka

„... mriežka umožňuje vytvárať konzistentný dizajn celého webu aj napriek veľmi odlišnému obsahu webovej prezentácie a dodáva prvkom webu poriadok a vizuálnu previazanosť.“ [7]

Aby bola mriežka flexibilná (z angl. *Flexible Grid*), použije sa vzorec

$$\text{Ciel'ová hodnota} / \text{kontext} = \text{výsledok}.$$

Ak je napríklad dizajn prezentácie navrhnutý na šírku 980 px a bočný panel v ňom umiestnený zaberá 300 px, jeho šírka sa nastaví na 30,612244898 % (300 px / 980 px). Celkovému obsahu sa nastaví šírka 95 %, čo nie je žiadne stanovené pravidlo, iba hodnota určujúca, koľko percent z dostupnej obrazovky bude obsah zaberáť.

Pri nastavovaní vonkajších a vnútorných okrajov (z angl. *margin s padding*) platí rovnaký vzorec. Každý okraj je rátaný na základe kontajnera, ktorému prináleží. Vonkajší prináleží nadradenému elementu, vnútorný sa počíta na základe šírky daného elementu.

2.2 Flexibilné média

Ak nie sú stanovené hranice obrázku pri jeho zobrazovaní vo flexibilnej mriežke sa môže kedykoľvek stať, že obrázok pretečie nadradený element a prekrýva iný obsah. Prvým variantom ako tento problém vyriešiť je nastaviť maximálnu šírku obrázku pomocou atribútu `max-width: 100 %`, čo zabezpečí, že nedôjde ku žiadnemu pretečeniu a obrázok sa proporčne prispôsobí na potrebnú veľkosť. Toto riešenie platí taktiež pre Flash objekty, videá a iné média, kde je potrebné zobrazit' všetky informácie obsiahnuté v médiu (tj. celý obrázok bez orezania). Druhým variantom je nezobrazit' pretekajúcu časť obsahu pomocou `overflow: hidden`. Hoci výsledný obrázok je orezaný, aby sa zmestil do kontajnera, aj takéto riešenie môže byť niekedy užitočné.

2.3 Otázky o médiu

Spolu s Media pomáhajú otázky o médiu (z angl. *Media Queries*, skratka MQ) webovému prehliadaču určiť druh koncového zariadenia a potrebné parametre tak, aby bolo možné pre každé zariadenie zmeniť rozloženie obsahu a jeho prvky reagujúc na obmedzenia, ktoré každé zariadenie so sebou prináša.

Media

Funkcia Media umožňuje zobrazovať rôzne formátovanie podľa druhu média, na ktorom sa bude obsah zobrazovať. Rozlišuje 8 typov médií [10], najpoužívanejšími sú tieto 4:

- *all* – zjednocuje všetky typy médií
- *print* – médium určené k zobrazovaniu pre tlač
- *handheld* – charakterizuje mobilne prenositeľné zariadenia
- *screen* – určené k zobrazovaniu na obrazovkách počítačov; ak typ nie je určený, pre prehliadač je to prednastavená hodnota

Základný popis MQ

MQ je funkcia zavedená do CSS3. Považuje sa za základný pilier RWD. Nadväzuje na funkciu Media a rozširuje jej schopnosti použitím niektorých pravdivostných operácií v kombinácii s vlastnosťami médií (z angl. *Media Features*). Podporovanými operáciami sú konjunkcia (AND), disjunkcia syntakticky značená čiarkou (,) a negácia (NOT). Konkrétny dotaz môže vyzerat' takto:

```
©media screen and (max-width : 800 px) { ... }
```

kde do zložených zátvoriek sú umiestnené CSS vlastnosti, ktoré budú aplikované iba ak bude web zobrazovaný na obrazovke počítača a okno prehliadača nebude širšie ako 800 px.

Vlastnosti médií poskytujú väčšie množstvo pravidiel ako prispôbiť dizajn pre rôzne zariadenia rovnakého typu. Rozlišujú sa vlastnosti [9]:

- šírka (width)
- výška (height)
- šírka zariadenia (device-width)
- výška zariadenia (device-height)
- orientácia (orientation) - napríklad portrét, krajina
- pomer strán (aspect-ratio)
- pomer strán zariadenia (device-aspect-ratio)
- farba (color) – počet bitov farby výstupného zariadenia (v prípade čierneho-bieleho zariadenia nadobúda hodnotu 0)
- index farby (color-index)
- jednofarebnosť (monochrome) – opak farby (v prípade čiernobieleho zariadenia nadobúda hodnotu 1)
- rozlíšenie (resolution)
- zobrazenie (scan) – použiteľné pre televízie
- mriežka (grid)

Väčšina uvedených pravidiel ako napr. výška a šírka podporujú aj prefixy max- a min-. Rozdiel medzi šírkou a šírkou zariadenia je ten, že prvé berie do úvahy celý priestor obrazovky vrátane posuvnej lišty. Obdobné platí aj v prípade výšky a výšky zariadenia.

Medzníky MQ

Medzníky zmeny rozloženia obsahu (z angl. *Breakpoints*) je možné určiť na základe najbežnejších zariadení na trhu, ako to robí väčšina vývojových prostredí, alebo aplikácií podporujúcich RWD. Tento prístup je síce menej pracný, keďže nie je potrebné všetky medzníky nastavovať, ale zároveň je menej flexibilný, pretože rozlíšenie najbežnejších moderných technológií sa mení veľmi často. Odporúčanejším variantom je nastaviť vlastné medzníky tak, aby vyhovovali konkrétnemu dizajnu.

Vytvorenie webovej prezentácie pre stolný počítač a postupné zmenšovanie pre ostatné zariadenia je síce rozšírený spôsob, ale pre staršie mobily bez podpory MQ je nefunkčný. Preto sa odporúča budovať web od najmenej schopných zariadení čo sa podpory funkcií vykreslenia prezentácie týka, až po tie najaktuálnejšie a s najlepšou výbavou.

Štýl umiestenia MQ

Pravidlá MQ sa vkladajú do súborov CSS a teda ich umiestenie má obdobné varianty. Môžu sa umiestniť do hlavičky dokumentu, uzavreté elementom style v ostrých zátvorkách. Druhým variantom je každé nové vymedzenie pravidiel volať v samostatnom súbore cez odkaz. Tretím variantom je umiestniť ich na koniec CSS dokumentu, spolu so všetkými ostatnými pravidlami. Posledná možnosť je najefektívnejšia, pretože nie je potrebné posielat' mnohonásobné požiadavky na stiahnutie súborov. Taktiež je aj najprehľadnejšia, keďže všetky formátovacie pravidlá sú pohromade, a preto sa tento variant aj najviac odporúča [8].

3 Mobil na prvom mieste

V období pred mohutným rozšírením využitia mobilných telefónov pre viac ako telefonovanie sa návrh webu príliš nezaoberal štruktúrou obsahu, prioritizáciou informácií alebo organizovaním prvkov ako takých. Aj napriek tomu, že existovali rôzne prístupy ako sa nad obsahovou stránkou webu zamyslieť a prístupy ako užívateľská

použitelnosť, existovalo aj existuje mnoho webových prezentácií, ktoré tieto prístupy pri návrhu nezohľadňovali.

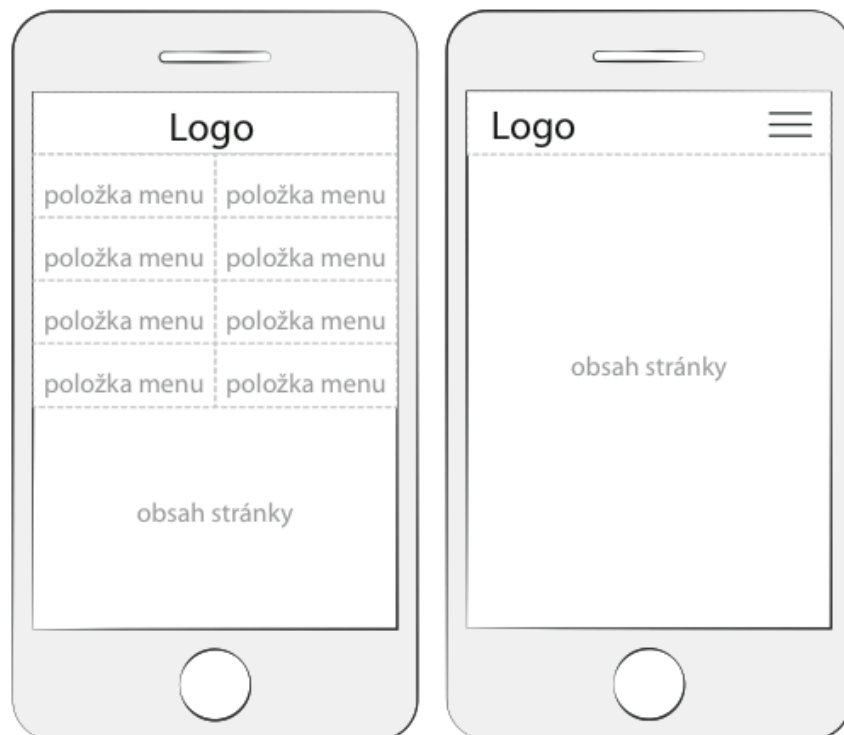
3.1 Obsah

Rozšírenie mobilných telefónov prinieslo množstvo obmedzení i výhod. Kvôli obmedzeniam ako rýchlosť pripojenia, výkon zariadenia a veľkosť obrazovky mobilu prišla myšlienka, že ak sa má na zariadení zobrazovať obsah, stojí za zváženie usporiadať ho podľa priority a stanoviť hierarchiu pre užívateľa najprístupnejšiu a intuitívnejšiu, čo priamo nadväzuje na informačnú architektúru a zážitok užívateľa.

Mobil na prvom mieste (z angl. *Mobile First*) je prístup, ktorý začína návrh webovej prezentácie pri mobilných zariadeniach, pretože takto sa pri návrhu zistia požiadavky užívateľa, obsah je redukovaný iba k tomu najdôležitejšiemu a nepodstatné informácie sú odstránené [11]. Tým sa z obmedzení stali výhody, pretože pomocou tohto prístupu je možné revidovať zobrazované informácie a znížiť množstvo stiahnutých dát, čím sa zníži čas potrebný na prechádzanie všetkých informácií.

Relevantnosť obsahu

Aby bol obsah webových prezentácií zameraný na potreby užívateľa, je potrebné zistiť, ako, kedy a za akých okolností užívateľia používajú svoj telefón. Pri stolnom počítači a notebooku, nebolo potrebné riešiť kontext použitia, pretože práca bola sústredená vždy za stolom, s vyhradeným časom a bez vyrušení, ktoré mobilné zariadenia so sebou prinášajú. Prípady použitia pomáhajú určiť, kedy sú ktoré informácie užitočné a ktorý obsah je pre mobilné zariadenia relevantný.



Obr. 1: Porovnanie dostupného miesta na obrazovke inteligentného telefónu pri schovanej a zobrazenej navigácii

Navigácia vs. Obsah

Zobrazovaním navigácie webu bežným spôsobom sa vyčerpáva miesto na obrazovke zariadenia. Pri akomkoľvek kroku, ktorý užívateľ vykoná, dostane najskôr informácie prináležiace k navigácii a až následne tie, o ktoré žiada. Luke Wroblewski zavádza

pravidlo, že obsah má prednosť pred navigáciou. Jednoduchšia hlavička, ktorá obsahuje iba logo webu a odkaz k navigácii vyrieši problém a ušetrí množstvo miesta na obrazovke, ukážka (Obr. 1). Variantom je taktiež umiestnenie navigačných prvkov až pod obsah, na koniec webu.

3.2 Vzhľad elementov

Veľkosť tlačidiel

Menšie rozmery obrazovky neznamenaajú, že obsah je zobrazovaný v menšej veľkosti. Tým, že sa užívateľ nenachádza v statickom prostredí a jeho pozornosť je iba čiastočná, cieľ jeho akcií musí byť jednoducho a bez omylov prístupný za akýchkoľvek okolností. Dotykové ovládanie zariadenia pomocou prstov vyžaduje väčšiu plochu ako doterajšie ovládače, ako napríklad myš alebo tablet. To znamená, že všetky tlačidlá, ktoré vyžadujú akciu od užívateľa musia spĺňať isté podmienky, aby sa užívateľovi zabezpečilo jednoduché ovládanie.

Každá zo spoločností, ktorá sa zaoberá vývojom mobilných zariadení si vytvorila vlastné štandardy, aká veľkosť tlačidiel je ideálna. Apple odporúča tlačidlá vo veľkosti minimálne 44 x 44 bodov [2]. Microsoft stanovil minimálnu veľkosť na 7 mm, ideálnu na 9 mm, pričom odporúča pre frekventovane používané tlačidlá práve 9 mm, aby sa obmedzilo množstvo neúmyselných stlačení [6]. Aj ďalšie spoločnosti majú vlastné štandardy, ktoré sú v hrubom merítku porovnateľné. Nie je potrebné dodržiavať ich striktné, treba brať ale do úvahy, že aplikovaním týchto rozmerov sa obmedzí chybovosť užívateľa a tým sa zníži aj množstvo faktorov, ktoré užívateľa môžu odrádzať od prezerania webu. Pri návrhu je tiež potrebné myslieť na dostatočné odsadenie jednotlivých tlačidiel medzi sebou, aby tie, ktoré sú blízko nespôsobovali užívateľovi ťažkosti.

Neexistujúci aktívny prvok

Keďže mobilné zariadenia nemajú žiaden kurzor, nie je ani rozoznateľný stav, kedy užívateľ prechádza kurzorom nad aktívnym prvkom (z angl. *hover*). Ak sa v desktopovej variante webu zobrazuje po prejdení myšou podstatná informácia, k užívateľovi s dotykovým zariadením sa bez úprav nedostane. Tieto zariadenia ale rozlišujú akcie, ktoré na desktope nie sú, a preto existujú možnosti ako informácie sprístupniť všetkým, pričom každé zariadenie bude mať svoju vlastnú, pre seba intuitívnejšiu cestu.

Možnosťou ako nahradiť informácie aktívneho prvku je zobrazit' ich bez akejkoľvek akcie na obrazovke. Ak sú informácie doplnkového charakteru, s prístupom iba na vyžiadanie, používa sa gesto *Tap*, ktoré je v niektorých mobilných zariadeniach automaticky aplikované na aktívny prvok. Zobrazenie dát na separátnej obrazovke je odporúčané v prípade rozsiahleho množstva informácií. Poslednou možnosťou ostáva informácie odstrániť, čím sa zjednoduší užívateľské rozhranie. V prípade používania dotykových gest je vhodné použiť známe a rozšírené gesto, ktoré nie je komplikované rozoznať.

4 Porovnanie metodík

4.1 Poradie zariadení

Každá z metodík popísaných v predchádzajúcom texte má vlastnú skupinu zariadení, na ktoré sa zameriava prednostne. Prispôsobivý web dizajn stavia na prvé miesto najstaršie a najmenej schopné zariadenia a prehliadače, pretože ak je obsah možné zobraziť na nich, s určitosťou je možné zobrazenie na každom ďalšom zariadení. Zážitok užívateľa stavia na druhé miesto. Je ale možné dobudovať ho ako nadstavbu, ak sa použije JavaScript alebo funkcie CSS, ktoré v starších prehliadačoch podporované nie sú a preto počas vykresľovania obsahu budú ignorované a nebudú ich zaťažovať.

Metodika mobil na prvom mieste odporúča navrhovať webové prezentácie najprv pre mobilné zariadenia, s použiteľnosťou typickou pre dotykové ovládanie. Pre ostatné zariadenia je zobrazovanie ponechané buď v mobilnej variante, čo sprístupňuje užívateľovi revidovaný a najviac relevantný obsah, ale aj nedostatočné využitie dostupného miesta. Druhou variantou je použitie a skombinovanie iných metodík k rozloženiu webovej prezentácie podľa veľkosti obrazovky konkrétneho zariadenia. Responzívny webdesign nestanovuje zariadenie, ktoré má prednosť a ponecháva to na tvorcovi prezentácie. Ten si sám musí vybrať, pre ktoré zariadenie bude návrh prvotný a ako sa bude obsah pomocou MQ prispôsobovať na iných zariadenia. Ak nejaké zariadenie nepodporuje MQ, je mu zobrazované rozloženie prvotného návrhu.

4.2 Výkon

Keďže prispôbovateľný webdesign vykresľuje štandardným spôsobom iba základnú formu zobrazenia obsahu, všetky nadstavby sú dobudované podľa schopností zariadenia za pomoci skriptovacieho jazyka JavaScript a formátovacích značiek CSS. To vedie k zaťažovaniu na strane klienta a vykresľovanie pri lepších zariadeniach môže byť pomalšie, vyťažujúc výkon zariadenia. Responzívny webdesign zabezpečuje, aby sa obsah prispôboval veľkosti zariadenia. Pri zobrazovaní je potrebné sťahovať všetky dáta webovej prezentácie, informácie týkajúce sa aktuálnej veľkosti aj tie, ktoré sú určené pre iné zariadenia. Obrázky zobrazované na veľkých obrazovkách s rozlíšením 1980x1080 px sú taktiež zobrazované aj na mobilných obrazovkách, kde je kvalita mnohonásobne väčšia než môže byť zobrazená. Pri tomto prístupe musí užívateľ prezerajúci web sťahovať omnoho viac dát ako v skutočnosti potrebuje a využije.

	Prispôbovateľný webdesign	Responzívny webdesign	Mobil na prvom mieste
hlavná myšlienka	layout postupne dokresľovaný podľa schopností prehliadača	layout reagujúci na zmenu rozlíšenia a iných podmienok	návrh pre dotykové mobilné zariadenia
prístup metodiky	implementačný	implementačný	návrhový
podpora prehliadačov	áno	desktopový návrh neprispôbovateľný na starých zariadeniach	-
revízia obsahu	nie	nie	áno
výkon	zaťaženie na strane klienta dokresľovaním zážitku (UX)	rovnako veľké súbory sťahované pre všetky zariadenia	-
zohľadnená použiteľnosť	áno	nie	áno

Obr. 2: Prehľad porovnania metodík pre návrh a realizáciu mobilného webu

4.3 Prehľad porovnania

Porovnanie popísaných metodík je zhrnuté na obrázku (Obr. 2). Porovnávané sú na základe prvotných zdrojov a nezohľadňujú rôzne kombinácie a vylepšenia s inými prístupmi. Keďže sa metodika mobil na prvom mieste zameriava prioritne na návrh pre mobilné zariadenia, implementačná časť je závislá úplne od tvorcu webu a preto nie je uvedená v tabuľke.

Literatúra

- [1] Allsopp, John. *A Dao of Web Design* [online]. 2000 [cit. 2013-02-09]. URL: <<http://alistapart.com/article/dao>>.
- [2] Apple, Inc. *iOS Human Interface Guidelines* [online]. 2013 [cit. 2013-03-02]. URL: <<http://developer.apple.com/library/ios/#documentation/userexperience/conceptual/mobilehig/Introduction/Introduction.html>>.
- [3] Gustafson, Aaron. *Adaptive Web Design: Crafting Rich Experiences with Progressive Enhancement*. Easy Readers, LLC, 2011. 137 s. ISBN 978-0-9835895-2-5.
- [4] Kadlec, Tim. *Implementing Responsive Design*. New Riders, 2012. 271 s. ISBN 978-0-321-82168-3.
- [5] Marcotte, Ethan. *Responsive Web Design*. A Book Apart, 2011. 157 s. ISBN 978-0-9844425-7-7.
- [6] Microsoft. *Interactions and usability with Windows Phone* [online]. 2013 [cit. 2013-03-02]. URL: <<http://msdn.microsoft.com/library/windowsphone/design/hh202889%28v=vs.105%29>>.
- [7] Řezáč, Jan. *Vývoj grafického rozhraní webového portálu* [online]. 2010 [cit. 2013-02-10]. URL: <https://is.muni.cz/auth/https://is.muni.cz/auth/th/140493/fi_m/rezacthesis.pdf>.
- [8] Smashing Magazine. *Mastering CSS for Web Developers*. Smashing Media GmbH, 2011. 351s. ISBN 9783943075137.
- [9] W3C. *Media Queries* [online]. 2012 [cit. 2013-02-10]. URL: <<http://www.w3.org/TR/2012/REC-css3-mediaqueries-20120619/>>
- [10] W3C. *Media Types* [online]. 2011 [cit. 2013-02-10]. URL: <<http://www.w3.org/TR/CSS21/media.html>>.
- [11] Wroblewski, Luke. *Mobile First*. A Book Apart, 2011. 138 s. ISBN 978-1-937557-02-7.