

**Univerzita Hradec Králové
Fakulta informatiky a managementu
Katedra managementu**

Návrh a realizace vzdělávacího portálu o webové grafice

Diplomová práce

Autor: Vít Kašpar

Studijní obor: Aplikovaná informatika, AI

Vedoucí práce: Mgr. Lukáš Vacek

Hradec Králové

listopad 2016

Prohlášení:

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 7. 11. 2016

Vít Kašpar

Poděkování:

Děkuji Mgr. Lukáši Vackovi za vedení mé diplomové práce, za cenné rady a odbornou pomoc.

Anotace

Tato diplomová práce se zabývá návrhem a realizací vzdělávacího portálu o webové grafice. Součástí této práce je vysvětlení základních principů webového designu, použití kaskádových stylů a jejich aplikování k vytvoření responzivních stránek. Následují příklady nástrojů určených k tvorbě webové grafiky.

V praktické části je vytvořen vzdělávací portál o webové grafice. Jeho obsah je reprezentován jednotlivými tematickými články rozdělenými do kategorií, aby byla zachována jejich přehlednost. Z důvodu ověření nabytých znalostí jsou vybrané články doplněny o testovací otázky. Součástí portálu jsou také video návody sloužící k názorné ukázce práce s grafickými nástroji.

Anotation

Title: Design and implementation of an educational portal for web graphics

This Diploma Thesis is concerned with design and implementation of an educational portal for web graphics. Part of this work is explanation of fundamental principles of web design, the use of cascading style sheets and their application to create responsive site. The following are examples of tools for creating web graphics.

In practical part is designed an educational portal for web graphics. The content of portal is represented by individual thematic articles, which are divided into categories to maintain their clarity. In order to verify the acquired knowledge are selected articles supplemented with test questions. The part of the portal are also video tutorials which serve to visual examples of work in graphic tools.

Obsah

1	Úvod	1
2	Cíl a metodika práce	2
3	Základní principy webové grafiky	4
3.1	Uživatelský pohled na webové stránky	4
3.2	Definování účelu webových stránek	6
3.3	Volba fontu	7
3.4	Použití barev	8
3.5	Obrázky a jejich formáty	13
3.5.1	Obrazové formáty	15
3.6	Navigace na stránkách	18
3.7	Mřížové uspořádání obsahu	20
3.8	Prohledávající vzory na stránkách	22
3.8.1	Vzor F	22
3.8.2	Vzor Z	23
3.9	Responzivní a adaptivní design	24
3.9.1	Základní principy tvorby webového designu pro mobilní zařízení	25
4	Kaskádové styly	29
4.1	Aplikování CSS	29
4.2	Selektory, vlastnosti a hodnoty	30
4.3	Definice barev	31
4.4	Úprava textu	32
4.5	Vnitřní a vnější okraje prvků	33
4.6	Ohraničení prvků	33
4.7	Základní prvky responsivního webového designu v CSS	35
4.7.1	Viewport	35
4.7.2	Grid-View	37
4.7.3	Media Queries	38
4.7.4	Obrázky	42
4.7.5	Videa	45
4.7.6	Frameworky	45
4.8	Moderní technologie pro práci s CSS	47
4.8.1	CSS preprocesory	47
4.8.2	Buildování CSS	50

4.8.3	Testování funkčnosti CSS.....	52
5	Programy určené k tvorbě frontend	54
5.1	Aplikace pro psaní kódu	55
5.2	Nástroje k tvorbě grafických prvků a webového designu	56
5.3	Kompletní nástroj pro vývoj webových aplikací.....	59
5.4	Redakční systémy CMS.....	61
6	Příprava obsahu.....	63
7	Vlastní návrh portálu.....	65
7.1	Tvorba webového portálu	65
7.2	Hlavní prvky portálu	67
7.3	Vytváření video návodů	69
7.3.1	Zpracování textových podkladů	69
7.3.2	Namluvení doprovodného komentáře.....	70
7.3.3	Tvorba video návodů prací v aplikaci snímáním obrazovky.....	71
7.3.4	Sestřihání a vyrenderování výsledného video návodu.....	71
7.3.5	Nahrání video návodu na internet a vložení do webového portálu	72
7.4	Testy znalostí.....	72
7.5	Přístupové údaje na webový portál	73
8	Ověření.....	75
9	Shrnutí výsledků	76
10	Závěr a doporučení.....	77
11	Literární a internetové zdroje	78
12	Seznam obrázků	85
13	Seznam příloh.....	86

1 Úvod

V dnešní době jsou webové stránky téměř nutností pro jakýkoliv typ podnikání, jelikož většina zákazníků vyhledává produkty a služby na internetu. Jejich vytváření je nyní velmi snadné a může je vytvořit téměř kdokoliv s přístupem na internet. Přesto je nutné dodržovat určité zásady, aby se na webových stránkách návštěvník cítil spokojeně.

Hlavní motivací k vytvoření vzdělávacího webového portálu je poskytnutí základních znalostí o tvorbě webové grafiky všem studentům začínajícím vytvářet webové stránky. Tomuto tématu není, z časových důvodů, během studia věnována dostatečná pozornost.

Právě na studenty je cílen tento webový portál. Konkrétně se jedná o studenty oboru informačních technologií. U těchto studentů je již předpokládána základní znalost tvorby v HTML.

Těmto studentům může portál nabídnout seznámení se základními znalostmi, které se využívají při tvorbě webové grafiky. Jedním z nich jsou obecné principy, jež je nutné znát před samotnou tvorbou webových stránek. Další znalostí je vysvětlení a použití kaskádových stylů při vytváření moderních responzivních webových stránek. Nakonec se mohou studenti dozvědět jaké nástroje k tvorbě využívat.

Všechny tyto informace lze získat na webovém portále, který nabízí seřazení jednotlivých témat do kategorií a článků pro zachování jejich přehlednosti. Pro názorné ukázky vytváření webových stránek portál obsahuje video návody s mluveným komentářem. Znalosti z jednotlivých témat si může každý student otestovat pomocí krátkých testů.

2 Cíl a metodika práce

Cílem této práce je návrh a realizace vzdělávacího portálu o webové grafice. Ten návštěvníky seznámí s tvorbou webových stránek, nabízející přehledný a responzivní webový design, který lze použít na všech koncových zařízeních. Takové stránky musejí být zároveň správně navržené, aby je šlo snadno spravovat webovými designery dle potřeb uživatelů a správců stránek.

K vytvoření kvalitního webového designu je nejprve nutné znát základní principy uživatelského rozhraní. Tyto principy jsou rozděleny do jednotlivých tematických oblastí. Nejprve je rozebrán uživatelský pohled na webové stránky, který je nutný znát, protože všechny stránky jsou primárně vytvářeny pro uživatele. Další oblastí je definování účelu webových stránek, kterému se musí následná tvorba přizpůsobit. Dále použití barev, fontů a obrázků, aby stránka působila na uživatele cíleným dojmem. Pro správné rozvržení prvků na stránce jsou vytvořeny oblasti o výběru navigace, mřížového uspořádání a vysvětlení základních prohledávacích vzorů využívaných na stránkách. Poslední oblastí základních principů je vysvětlení pojmu responzivního designu.

Kromě základních principů je také vysvětlen význam a použití kaskádových stylů, jejichž znalost je nezbytná k vytváření moderního webového designu. Nejprve přehled nejčastěji využívaných prvků a poté prvky využívané při vytváření responzivního designu, včetně použití již vytvořených frameworků. K oblasti kaskádových stylů je také nutné zmínit moderní technologie pro práci s nimi, jako jsou preprocesory nebo nástroje k jejich buildování a testování.

Weboví designeři by také měli mít přehled o nástrojích využívaných k tvorbě webové grafiky. Tyto nástroje usnadní tvorbu a urychlí aplikování změn. Ať už se jedná o nástroje k úpravě kódu, grafických prvků nebo komplexní řešení nabízející vše potřebné k tvorbě webových stránek v jediném nástroji.

Všechny tyto poznatky jsou předány cílovým uživatelům pomocí vzdělávacího webového portálu. Ten obsahuje jednotlivé oblasti problematiky tvorby webové grafiky ve formě článků, které jsou pro přehlednost rozděleny do několika kategorií. Vybrané články jsou doplněny o testovací otázky, určené uživatelům k ověření nově získaných znalostí. Dalším prvkem portálu jsou komentované video

návody o tvorbě webových stránek pomocí vybraného nástroje, které uživatelům poskytnou názorné ukázky, jak s nástrojem pracovat. Tvorba video návodů probíhá nejprve zpracováním textových podkladů, které jsou následně namluveny a doplněny o video záznam vytvořený snímáním obrazovky při práci s daným programem.

3 Základní principy webové grafiky

Pro každého tvůrce webových stránek a designu je nutné znát základní principy a souvislosti. Z toho důvodů je důležité nejprve tyto principy ukázat a vysvětlit, poté se zaměřit na jejich aplikování v konkrétních případech.

3.1 Uživatelský pohled na webové stránky

Míra kvality a efektivnosti webového designu je vždy hodnocena především uživateli a nikoliv administrátory webových stránek. Existuje mnoho faktorů ovlivňujících použitelnost webových stránek jak z vizuálního, tak funkčního hlediska.

Hlavním předpokladem pro úspěšné webové stránky je jejich **použitelnost a užitečnost** pro cílové uživatele, proto by měl být využíván design orientovaný především na ně.

Pro správné užití principů efektivnosti webového designu je nejprve nutné pochopit to, jak uživatelé pracují se stránkami, jak přemýšlí a jaké jsou základní vzory jejich chování.

V podstatě jsou zvyky uživatelů na webu velmi podobné chování zákazníku v obchodech. Návštěvník rychlým pohledem pročte každou novou stránku, přečte si výrazný text a klikne na první odkaz, který ho zaujme nebo působí jako věc, kterou hledá. Na většinu obsahu stránky se často ani nepodívá.

Většina uživatelů hledá něco zajímavého, užitečného a nejlépe ve formě obrázku či odkazu, aby na to mohli kliknout. Jakmile naleznou, co hledají, tak kliknou. Jestliže nejsou spokojeni s výsledkem, vrátí se zpět a pokračují v hledání.

Uživatelé oceňují především kvalitu a důvěryhodnost. Jestliže stránka poskytuje kvalitní obsah, tak jsou ochotni přistoupit na kompromis obsahu s reklamou a případný horší vzhled. Z tohoto důvodu mohou mít webové stránky nabízející kvalitní obsah velkou návštěvnost, i když nebudou mít povedený webový design nebo mnoho reklam. To samé platí i v opačném směru, kdy kvalitní design nemusí být zárukou vysoké návštěvnosti, jestliže stránky nebudou nabízet hodnotný obsah. [1]

Uživatelé nečtou, ale pouze rychle prohlíží. Při hledání na webových stránkách využívají pevné body, které jim pomohou rychle projít obsah stránek. Typickým jevem rychlého prohlížení je hledání ve středu vět, které je dobře vidět na studii sledování pohybu očí uživatelů, provedené společností *Nielsen Norman Group* (Obr. 1). Uživatel nikdy nečte text na stránkách postupně, ale rychle prohlédne celé stránky a pozastaví se pouze na bodech, které ho zaujmou. [2]



Obrázek 1: Výsledek studie sledování očí uživatelů

Uživatelé jsou netrpěliví a požadují okamžité uspokojení. Například pokud webové stránky nenaplní návštěvníkovo očekávání, nekvalitním designem nebo neintuitivním vyhledáváním, tak návštěvník odejde hledat jinde.

Uživatelé nedělají správná rozhodnutí. Uživatelé nevyužívají nejrychlejší možné cesty k nalezení informace, kterou hledají. Nejdou kategorii po kategorii, aby našli to, co hledají, ale kliknou na první možný odkaz, který se zdá být zajímavý nebo si myslí, že je dovede k hledané informaci. Přemýšlení nad cestou a způsobem jakým by měli hledat je velmi zdlouhavé, a proto ho běžný uživatel ve většině případů nepoužije.

Uživatelé chtějí mít kontrolu. Chtějí mít kontrolu nad svým prohlížečem a přehled nad zobrazenými daty. Požadují jednotlivé komponenty tam, kde by je

očekávali, a to bez odlišného chování, jako jsou například vyskakovací okna. Tyto prvky mohou na první pohled působit zajímavě, ale uživatele většinou omezují v prohlížení, protože již nejsou schopni se z nově otevřeného okna vrátit zpět, kde skončili obvyklým způsobem pomocí tlačítka zpět.

Při tvorbě webových stránek je nutné brát v potaz všechna zmíněná tvrzení o chování uživatelů a využít těchto vědomostí k vytvoření uživatelsky přívětivým webovým stránkám.

3.2 Definování účelu webových stránek

Design webových stránek musí vždy záležet na potřebách cílových uživatelů. Ať webová stránka obsahuje vzdělávací informace, zábavu či propagaci společnosti, tak vždy musí všechny součásti stránky mít své opodstatnění a pokud možno efektivně splňovat požadavky návštěvníků.

V případě vytváření webového designu pro obchodní potřeby, ať už se jedná o stránky propagující firmu, její výrobky nebo nabízející služby, je velmi důležité, aby stránky dodávaly profesionální dojem a uměly potencionálního zákazníka dostatečně zaujmout. Zároveň musí být u takovýchto stránek mnohem více dbáno na bezpečnost, aby návštěvník stránkám důvěřoval a nebál se na nich nakupovat.

Jestliže se má jednat o webové stránky určené k poskytování informací, tak je velmi důležité rozhodnout se nejdříve, zda umožnit návštěvníkům komentovat jednotlivý obsah webu. V takovém případě musí být dále rozhodnuto, zda budou komentáře interní přímo na stránkách s možností jejich moderování nebo externě, formou sociálních sítí. Podle zvoleného typu komentářů na stránkách je nutné zohlednit zvýšený důraz na bezpečnost webových stránek a jejich napojení na databázi. Ovšem díky komentářům je autor obsahu schopen získat zpětnou vazbu od návštěvníků, která mu pomůže obsah nebo celé stránky zdokonalit.

Před začátkem vytváření stránek je tedy nutné nejprve definovat cílovou skupinu uživatelů a jejich potřeby. Tomu se následně musí celková tvorba webu přizpůsobit.

3.3 Volba fontu

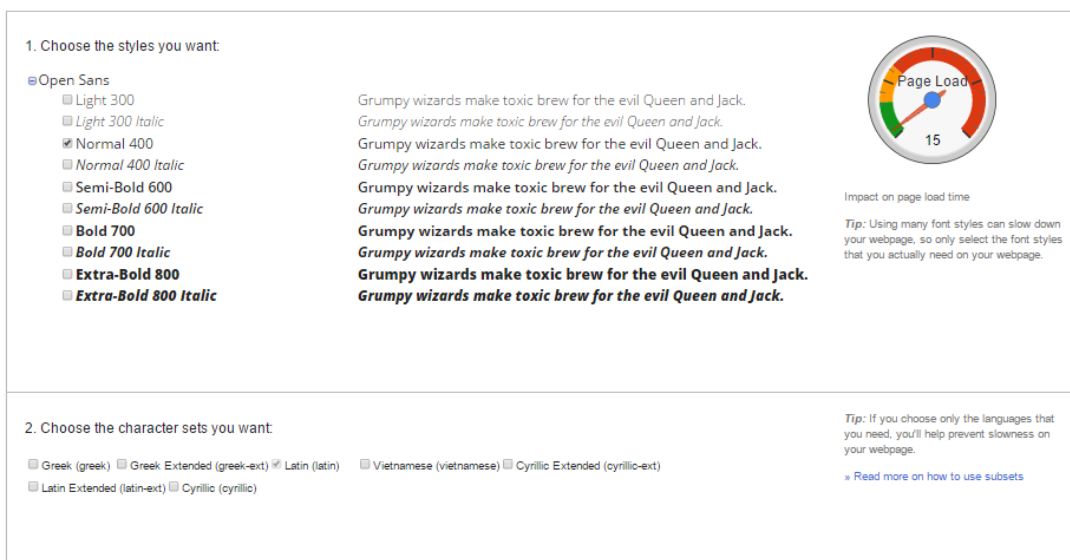
Font by se měl používat především bezpatkový, jako je *Arial* nebo *Verdana*, tyto fonty se čtou snadněji než fonty patkové. Ideální velikost použitého písma je šestnáct pixelů a maximálně tři různé typy fontů v rozmezí velikosti tři pixely, aby byl zachován jednoduchý a přehledný vzhled.

Uživatelé chtějí získat informace co nejrychleji, a proto je důležité vyjadřovat se na stránkách stručně a srozumitelně, aby byl uživatel schopen snadno přečíst a vstřebat všechny důležité informace. Účinné metody k zpřehlednění je využití nadpisů, podnadpisů, seznamů, krátkých vět a vyhnout se dlouhým odstavcům.

V dobách počátku internetu bylo pouze pár fontů, které byly podporovány webovými prohlížeči. Jakmile byl použit nestandardní typ, většina uživatelů viděla místo nestandardních písmen pouze náhodné znaky. V dnešní době stále existují fonty, které nejsou podporovány všemi prohlížeči, ale počet takzvaných "*web-safe fonts*¹" neboli bezpečných webových fontů velmi výrazně narostl, a to především díky použití CSS pravidla *@font-face*, který si umožňuje nastavit do stránky vlastní font a pojmenovat si ho. Vytvořené pojmenování lze následně použít u vlastnosti *font-family*, která určuje, v jakém fontu se text bude vykreslovat a *@font-face* ho načte ze stránky.[3], [4]

Při rozhodování jaký font použít pro webové stránky je velmi nápomocný *Google Fonts*, jenž nabízí rozsáhlé množství fontů k užití zdarma. Hlavní předností *Google Fonts* je především integrace nástrojů, které zobrazují náročnost vybraného fontu na načítání stránky, znázorní jeho kompatibilitu s jednotlivými znaky a zároveň vytvoří úsek kódu, který se jen vloží do CSS souboru (Obr. 2). [5]

¹ Web-safe fonts umožňují využití více rozšířených fontů. Více informací lze nalézt na stránkách http://www.w3schools.com/cssref/css_websafe_fonts.asp.



Obrázek 2: Prostředí webové aplikace Google Fonts

Při výběru a použití fontu je nutné dodržovat následující základní pravidla:

- Patková písma by se měla používat pouze pro nadpisy, protože při menších velikostech se obtížněji čtou; naopak hlavní text by měl být psán písmem bezpatkovým.
- Počet použitých typů fontů by nikdy neměl být vyšší jak dva až tři typy. Jakmile je použito více typů, tak se webové stránky stanou nepřehlednými.
- Některé soubory fontů mohou být náročnější na velikosti, a to má za následek pomalejší načítání webových stránek. Vždy je důležité nejprve otestovat rychlost s použitím nového fontu, například pomocí již zmíněného nástroje na stránkách *Google Fonts*.

Znalost správného výběru a použití fontu na stránkách je klíčovým znakem k úspěchu.

3.4 Použití barev

Dobře promyšlené použití palety barev může mít velký vliv na celkový dojem z webových stránek. Doplnkové barvy vytvářejí rovnováhu a uklidňují, naopak kontrastní barvy se využívají pro barvu textu a pozadí, aby usnadnily jeho čtení. S rozvahou je potřeba využívat zářivých barev, které zvyšují emoce a měly by být využity například pouze na tlačítka a upozornění. Důležitým prvkem často použí-

vaným u dnešních webových stránek jsou prázdné prostory, které dodají moderní a přehledný vzhled. Minimalistický design je v současnosti jednoznačným trendem.

Podobně jako u fontů je i u barev vhodné využít pouze dvě až tři různé pro celý web. V případě komerčních webů by stránky měly odrážet barvy firemní značky a zároveň by měly zvýrazňovat důležité oblasti webu. V případě mobilního rozhraní je nejvhodnější zvolit černý text na bílém pozadí, z důvodu dobré čitelnosti například i na ostrém slunečním světle.

Důležitá je také správná kombinace barev, protože celých 5 % mužské populace je buď úplně, nebo alespoň částečně barvoslepých. [6] Pro správné zvolení barev je možné využít nástrojů, které přímo nabízejí možnost zohlednit konkrétní poruchy vidění barev. Jedním takovým nástrojem je webová aplikace *paletton.com*. [7] Příklad špatného a správného použití barevné kombinace je znázorněn níže (Obr. 3).



Obrázek 3: Ukázka použití správných barev

Pro webový design jsou důležité **tři aspekty barev**:

- **Kontrast** - každá barva má několik různých odstínů daných rozdílným kontrastem, kterého lze využít na příklad k optickému rozdělení jednotlivých

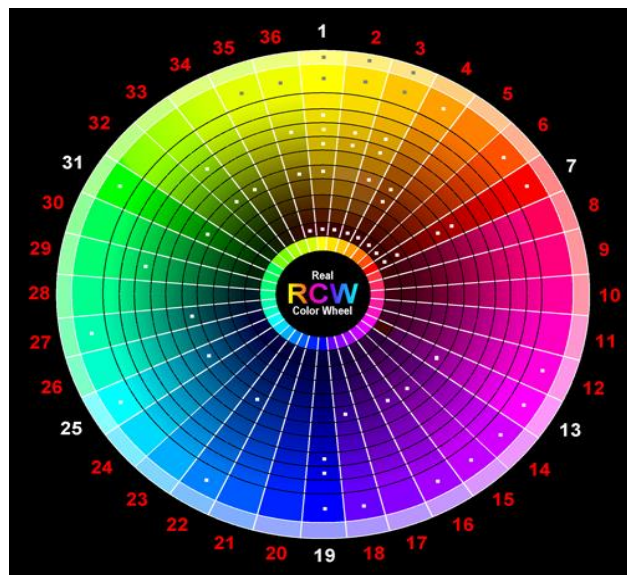
elementů webové stránky. Pozadí, hlavička, obsahová oblast i patička stránky by měly být vizuálně oddělené, k tomu se nejlépe hodí právě různý kontrast barvy použité v pozadí. [8]

- **Komplementace** – neboli doplňkové barvy, které v barevném kruhu stojí naproti sobě a dávají nejsilnější barevný kontrast. Komplementace konstatuje, že ta základní barva, která se nepodílela na vzniku barvy sekundární, je k ní komplementární. [9]
- **Živost** – každá barva vytváří různé nálady, a proto je nutné zvolit správnou barvu pro cílený vliv na návštěvníka stránek. Psychologický vliv barvy na lidské myšlení je různý pro jednotlivé kultury, proto musí být bráno v ohledu, zda se jedná o stránky určené pro konkrétní kulturu nebo o mezinárodní stránky, kde je lepší použít spíše neutrální barvy, které nemají velký vliv na změnu nálady. U lidí západní kultury mají základní barvy následující vliv na vyvolání nálady: [10]
- **Červená** – pocit síly, důležitosti a mládí.
- **Oranžová** – pocit přátelství, energie a jedinečnosti.
- **Žlutá** – pocit štěstí a nadšení.
- **Zelená** – pocit růstu, stability, financí a environmentálních témat.
- **Modrá** – pocit klidu, bezpečí, otevřenosti a u tmavšího odstínu tajemna.
- **Fialová** – pocit luxusu, romantiky a tajemství.
- **Černá** – pocit síly, podrážděnosti a sofistikovanosti.
- **Bílá** – pocit čistoty, cnosti a jednoduchosti.
- **Šedá** – pocit neutrality, formality a melancholie.

Barevné schéma

Při vytváření webové stránky je důležité správně zvolit barevné schéma. Jednak je nutné přemýšlet nad psychickým vlivem barvy na náladu návštěvníka stránky, ale zároveň musí barvy spolu ladit. Pro správný výběr barev, které spolu vytvářejí příjemnou kombinaci, se využívá sedmi hlavních typů schémat barev.

„Pro správné znázornění jednotlivých schémat se používá takzvané barevné kolo (Obr. 4). Jedná se o základní nástroj pro kombinování barev. První kruhový barevný diagram byl vytvořen už v roce 1666 Sirem Isaacem Newtonem. Barevné kolo bylo postupně několikrát upraveno, ale nejběžnější verzí je kolo o dvanácti barvách založených na RYB (červená-žlutá-modrá) barevném modelu.“ [11]



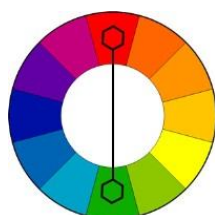
Obrázek 4: Barevné kolo

Monochromatické

Monochromatické schéma vychází z jednoho odstínu a využívá jeho stíny a tóny vzniklé změnou jasu a sytosti společně s černou a bílou barvou. Schéma poskytuje barevné kombinace příjemné pro oči ve všech odstínech. Nedostatkem je obtížnější zvýraznění a podtržení barevného tónu.

Kontrastní

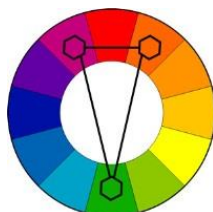
Jedná se o kombinaci barev, které jsou naproti sobě v barevném kole. Hlavním efektem použití kontrastních barev je jejich zářivý vzhled způsobený vysokým kontrastem, zejména při jejich plné sytosti. Tyto barvy by se měly používat pouze pro zvýraznění důležitých prvků, ale nikdy ne jako barva písma. (Obr. 5)



Obrázek 5: Kontrastní schéma

Dělené kontrastní

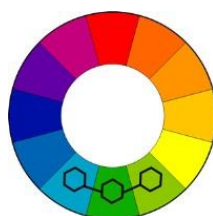
Dělené kontrastní nebo také měkké kontrastní barevné schéma je variací kontrastního schéma, kde se kromě základní barvy používají také dvě barvy přilehlé k jeho kontrastní barvě na druhé straně barevného kola. Podobně jako kontrastní schéma má vysoký vizuální kontrast, ale působí menší napětí. (Obr. 6)



Obrázek 6: Dělené kontrastní schéma

Analogické

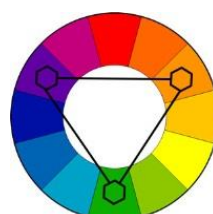
Analogické barevné schéma využívá barev, které jsou na barevném kole umístěné vedle sebe. Tyto barvy k sobě dobře ladí a vytvářejí klidný a komfortní dojem. Analogické barvy jsou velmi často k vidění v přírodě a jsou velmi příjemné pro oči. Při jejich použití je nutné použít dostatečný kontrast u jedné barvy, která bude dominantní a zbylé dvě jako barvy pomocné. (Obr. 7)



Obrázek 7: Analogické schéma

Triádní

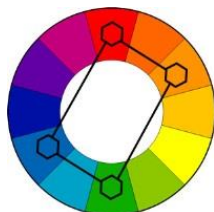
Triádní neboli trojúhelníkové barevné schéma využívá barev, které jsou rovnoměrně rozloženy po barevném kole. Tato kombinace vytváří živé barvy, které musejí být podobně jako u analogických opatrně vyváženy. Jedna barva by měla být dominantní a další dvě pouze jako pomocné. (Obr. 8)



Obrázek 8: Triádní schéma

Tetradické

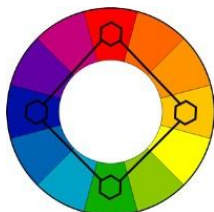
Tetradické neboli dvoj kontrastní schéma, využívá dvou kontrastních párů barev uspořádaných do obdélníku na barevném kole. Toto schéma nabízí bohatou varianci barev, ale nejlépe působí při použití pouze jedné dominantní barvy s vyváženým rozložením teplých a studených odstínů barev. (Obr. 9)



Obrázek 9: Tetradické schéma

Čtvercové

Čtvercové barevné schéma je podobné tetradickému. Také se využívá čtyř barev, které jsou ovšem rovnoměrně rozložené na barevném kole. [12] Použití je také stejné (Obr. 10)



Obrázek 10: Čtvercové schéma

Volba použitých barev na webových stránkách silně ovlivňuje vnitřní pocity většiny uživatelů, a proto je nutné využívat zmíněných znalostí o nich.

3.5 Obrázky a jejich formáty

Správné použití obrázků může pomoci se zviditelněním značky, prodejem výrobku nebo propojením s cílovou skupinou návštěvníků. Lidé mají v oblibě stránky, které obsahují obrázky, a preferují je před stránkami obsahujícími pouze strohý text. Důvod je především v pohodlnosti uživatelů, kteří místo čtení dlouhého textu se raději podívají na vysvětlující obrázek nebo graf. Zároveň obrázky vyvolávají více emocí, jež napomáhají k lepšímu zapamatování informací. V neposlední řadě obrazová sdělení pomáhají překonávat jazykové bariéry návštěvníků.

Při použití obrázku na webových stránkách je nutné řídit se několika hlavními zásadami:

- Vyhnout se použití pohyblivých obrazových prvků jako jsou GIFy nebo Flash videa, která se sama spustí a nelze je vypnout. Jakýkoliv vizuální pohyb bývá velmi rušivým a nepříjemným elementem, proto je vhodné se jim vyhnout.
- Nepoužívat obrázky pouze k vyplnění prázdných míst na obrazovce. Každý obrázek by měl mít své opodstatnění a předávat nějakou informaci. Například představovat produkt nebo názorně vysvětlovat popisovanou problematiku.
- Dát si pozor na licenční práva u použitých obrázků. Většina obrázků je chráněna autorskými právy, a proto je nutné k jejich užití za ně zaplatit.
- Velikost obrázků v bytech by měla být co nejmenší, protože často až 60 % velikosti webových stránek jsou právě obrázky. Jestliže bude velikost stránek příliš velká, tak se budou pomalu načítat, to návštěvníky od jejich užití odradí. [13]
- Použití obrázků místo textových odkazů je často vhodnější, a to díky získání větší pozornosti jednodušším použitím na dotykových displejích. Každý odkazující obrázek by měl být popsán stejně, jako kdyby se jednalo o textový odkaz.
- Vyhnout se vzhledu a umístění obrázku připomínající reklamy. Reklamy jsou velmi často umístěny na stejných místech a lidé je automaticky podvědomě ignorují, proto není vhodné umisťovat jakékoliv obrázky do horní nebo postranní části obrazovky, kde budou s největší pravděpodobností návštěvníky ignorováni. [14]
- Při použití obrázku v pozadí je nutné, aby se nejednalo o příliš rušivý motiv, který vytváří špatný celkový dojem. Zároveň je důležité, aby bylo na první pohled jasné, kde končí pozadí a kde začíná obsahová část stránek.
- Použití fotek s lidmi pomůže zaujmout většinu návštěvníků. Stránky působí na lidi více přátelsky, pokud vidí lidské tváře. Mají poté pocit, že se nejedná

pouze o naprogramované webové stránky, ale že jsou ve spojení se skutečnými lidmi. [15]

- Používat obrázky, které se zobrazí celé na stránce bez nutnosti posouvání dolů po stránce. Lidé se na stránkách neradi posouvají, protože chtějí získat informace co nejrychleji. V případě, že je nutné použít posouvání na stránkách, tak k tomu musí mít návštěvník důvod a informace obsažená v oblasti po posunutí by měla být více důležitá. [2]
- Obrázky s velkými rozměry musejí být pomocí CSS ošetřeny, aby velikost obrázku nikdy nepřekročila šířku obrazovky při zobrazení na mobilních zařízeních.
- Nejčastějším obrázkem na webových stránkách je logo společnosti, které by mělo být obsaženo v hlavičce stránek pokaždé na stejném místě, aby návštěvník dokázal poznat, o jaké stránky se jedná a že se na nich stále nachází. [16]
- Ke všem obrázkům nesoucím obsahovou informaci je vždy nutné doplňovat jejich popisky. Tyto popisky jsou nezbytné především pro uživatele s poruchou vidění, využívající čtečky stránek.

3.5.1 Obrazové formáty

Při vkládání obrázků nebo jakýchkoliv grafických prvků na webové stránky je velmi důležité zvolit jejich správný formát. Při použití obrázku v nejvyšší kvalitě může nastat problém s rychlostí načítání stránek, která se odvíjí od celkové velikosti stránek. Naopak při použití obrázku v příliš velké kompresi a malém rozlišení může mít dopad na kvalitu obrázku, který bude kazit celkový dojem z webových stránek. Existuje celá řada obrazových formátů, které nabízejí různé komprese a kvality. Díky tomu se každý hodí na jiné použití.

Výběr správného formátu pro optimalizaci stránek

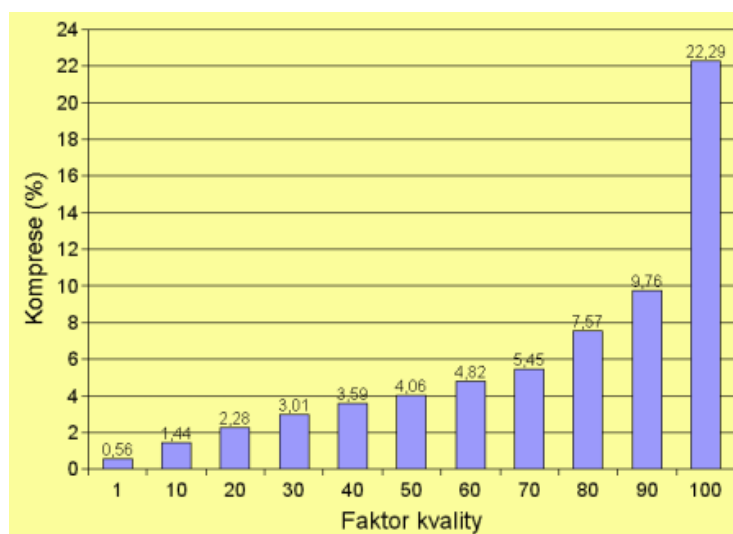
Nejběžnějším formátem používaným pro uložení obrázků na webové stránky je GIF. Soubory GIF jsou většinou malé velikosti a snadno se vytvářejí. Pro vytváření grafických elementů, jako jsou jednobarevné oblasti, text nebo liniové kresby, se používají formáty GIF a PNG-8. Fotografie se ukládají nejčastěji ve formátu JPEG, který nabízí velkou ostrost a kvalitní přechod mezi jednotlivými odstí-

ny barev. Velmi vysokou kvalitu nabízí také PNG-24, ale oproti formátu JPEG jsou mnohem náročnější na velikost. Vždy je vhodné každý grafický prvek nejprve otestovat v různých formátech a vybrat ten nejlepší kompromis mezi kvalitou a velikostí souboru. K optimalizaci komprese slouží mnoho nástrojů. Volně dostupným na internetu je například „*JPEG Resampler 2010*“ nebo placený profesionální nástroj *Adobe Photoshop*. [17][18][19]

JPEG

Formát JPEG je podporován většinou webových prohlížečů. Podporuje dvacet čtyřbitové barvy a nabízí širokou škálu jemných změn jasů, odstínů a přechodů barev.

JPEG využívá takzvanou ztrátovou kompresi, což znamená, že snižuje velikost souboru výběrovým odstraněním informací, které jsou v původním obrázku přítomné, ale postradatelné. Při ukládání formátu v JPEG se nejčastěji využívá faktor kvality v rozsahu nula až sto procent. „*Tento faktor ovlivňuje složení tabulek, které po provedené diskrétní kosinové transformaci redukuje spočtené hodnoty, které tak nemusí být ukládány. Čím nižší je nastavený faktor, tím větší je komprimační poměr, ale zvyšuje se také zkreslení. Pro mnoho aplikací se ukazuje, že se optimální hodnota pohybuje v rozmezí 70 až 80 %.*“ [20] Při vyšší hodnotě dochází k velmi vysokému nárůstu velikosti souboru, ale téměř nepozorovatelné změně obrazové kvality. Poměr komprese a faktoru kvality je znázorněn níže (Obr. 11).



Obrázek 11: Poměr komprese a faktoru kvality

Při použití JPEG souborů ve větší velikosti je vhodné využít progresivního JPEG souboru neboli obrázku v nízké kvalitě, který se načte okamžitě po načtení obrazovky, zatímco se načítá obrázek v plné kvalitě. Díky tomu uživatelé na první pohled vidí, že se obrázek stále načítá.

JPEG nepodporuje průhlednost, a proto jsou při vytváření průhledné body nahrazeny barvou podkladu. Pokud je známa barva pozadí, tak lze použít stejnou barvu po podklad, tím částečně nasimulovat průhlednost.

GIF

Podobně jako JPEG je i GIF podporován většinou moderních webových prohlížečů. Formát GIF využívá osmi bitových barev a komprimuje plné barevné plochy, zatímco zachovává ostrost detailů, proto je vhodný pro použití u textových ilustrací, logách nebo liniových kreseb.

GIF využívá bezztrátovou metodu komprese, to znamená, že nejsou odstranovány žádné informace při opakovaném ukládání. Hlavní nevýhodou je pouze osmi bitová podpora barev, na kterou pokud se přesune obraz s původně dvaceti čtyř bitovými barvami, se velmi sníží kvalita obrazu.

Hlavní výhodou formátu GIF je podpora animací, průhlednosti pozadí a podklad pozadí, díky kterému lze okraje pozadí prolnout společně s pozadím webových stránek.

PNG

Formát PNG je modernějším nástupcem formátu GIF. Stejně jako GIF využívá formát PNG-8 osmi bitových barev a komprimuje plné barevné plochy, zatímco zachovává ostrost detailů, proto je vhodný pro použití u textových ilustrací, log nebo liniových kreseb.

Formát PNG-24 má stejné vlastnosti jako PNG-8, ale navíc podporuje dvaceti čtyř bitové barvy a nabízí širokou škálu jemných změn jasů, odstínů a přechodů barev podobně jako formát JPEG. Nevýhodou toho formátu je větší velikost souboru.

Hlavní výhodou formátu PNG je plná podpora průhlednosti nazývaná *alpha kanál*. „Podle typu ukládaného obrázku je možné buď přímo každému pixelu umístě-

nému v rastrové mřížce, nebo každé barvě uložené v barvové paletě přiřadit hodnotu průhlednosti. Podporován je jak plný osmibitový či dokonce šestnáctibitový alfa kanál (256 či 65536 stupňů průhlednosti, tj. hodnot α), tak i „jednobitová“ průhlednost známá už z grafického formátu GIF.“ [21]

3.6 Navigace na stránkách

Navigace na webových stránkách je nezbytná pro snadné hledání a pohyb mezi jednotlivými stránkami. K navigaci se využívá několik různých metod například stromové uspořádání, drobečková navigace nebo využití takzvaného „tří klikového pravidla“, které znamená, že návštěvníci stránek by měli být schopni nalézt požadovanou informaci na tři kliknutí.

Webová navigace je vstupní branou k jednotlivému obsahu, a proto se je nutné držet základních pravidel při jejím vytváření [22]:

1. Navigaci by mělo být snadné nalézt

Uživatelé jsou často velmi netrpěliví, a pokud nejsou schopni na první pohled nalézt navigaci stránek, tak stránky opustí. Navigace by měla být hlavním elementem designu. Jelikož většinou zabírá mnohem méně místa než ostatní elementy na stránce, měla by být dostatečně zdůrazněna, aby vynikala nad zbytkem obsahu stránek. Nejběžnější oblastí umístění navigace je vrchní oblast stránek, proto je vhodné zachovat toto umístění, na které jsou uživatelé již zvyklí a zároveň se nemusí posouvat dolů na stránce.

Navigace u mobilního rozhraní by měla být umístěna v pravém horním rohu obrazovky, kde je nejbližší k palci pravé ruky při držení mobilního telefonu nebo tabletu.

2. Navigace by měla být konzistentní

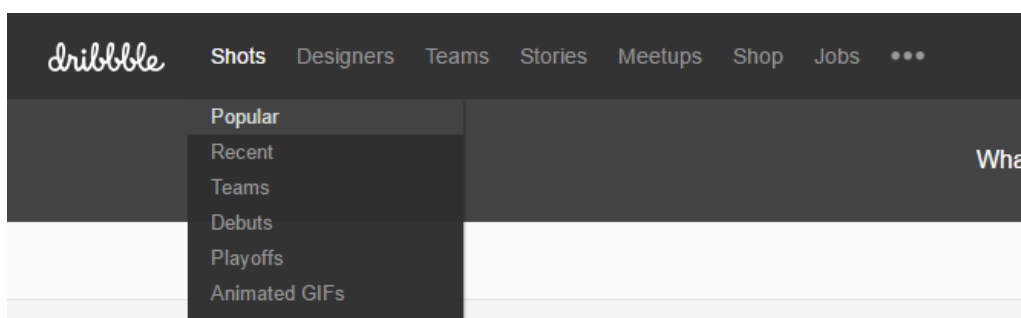
Stránková navigace by měla být stále na stejném místě na každé stránce webu. Zároveň by měl být zachován stejný styl, písmo a barvy. Díky tomu si na to uživatelé budou moci zvyknout a budou se cítit pohodlně při procházení jednotlivých stránek. Jestliže navigace nebude konzistentní a bude vždy na jiné pozici, v jiné barvě nebo úplně zmizí, tak to uživatele bude rozptylovat a nejspíše web opustí.

3. Používat přesně vypovídající názvy sekcí

Názvy jednotlivých oblastí webových stránek, především v oblasti navigace, by měly přesně definovat jejich obsah. Používání příliš obecných názvů způsobí, že uživatel bude nucen klikat na více tlačítek, než nalezne to, co hledá. Je důležité používat termíny, na které jsou uživatelé zvyklí, jako jsou například „Novinky“ nebo „Kontakty“.

4. Navigace by měla být stručná

Příliš mnoho navigačních tlačítek vedoucích do více sekcí ponechává návštěvníka s příliš mnoha možnostmi a udržuje je ho tak dál od nejdůležitějších oblastí webu. Důmyslnějším řešením pro obsáhlé navigace je použití rozbalovací nabídky pro jednotlivé sekce (Obr. 12), ale rozsah této nabídky by neměl být zbytečně velký.



Obrázek 12: Ukázka rozbalovací nabídky

Rozhraní s navigací pro mobilní zařízení by mělo být ve formě typické ikony menu, které se po kliknutí prstem vysune. Tato nabídka by měla být jednoduchá, přehledná s dostatečně velkými tlačítky, aby byla snadno ovladatelná prsty (Obr. 13).



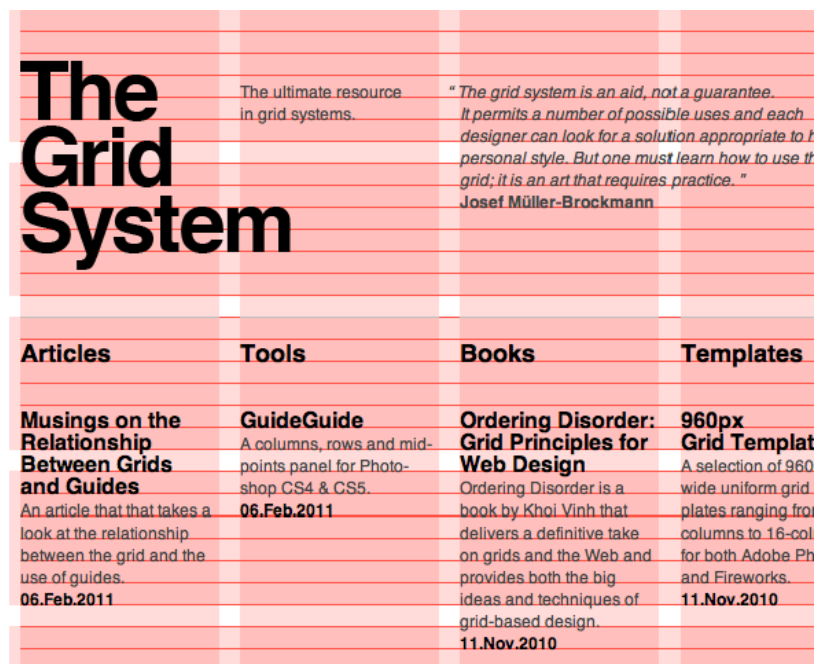
Obrázek 13: Navigace mobilní verze stránek

5. Zdůraznění aktuální sekce

Uživatel by měl mít stálý přehled, v jaké sekci na webu se právě nachází. Toho lze docílit použitím zvýraznění sekce v navigaci jinou barvou, ale takové zvýraznění by mělo být vždy konzistentní na všech stránkách. Zároveň je důležité, aby se uživatel mohl snadno vrátit na předešlou nebo domovskou stránku webu pomocí dobře viditelného tlačítka.

3.7 Mřížové uspořádání obsahu

Mřížové uspořádání se využívá k seřazení obsahu do jednotlivých sekcí a sloupců, díky kterým působí webové stránky zarovnaným a přehledným dojmem zlepšujícím celkový dojem na použitý design. Jedná se pouze o sérii protínajících se svislých a vodorovných čar rozmístěných v pravidelných intervalech (Obr. 14). [23]



Obrázek 14: Ukázka mřížového rozložení

Použití mřížového uspořádání je nutností v případě vytváření responzivního designu. Responzivní design znamená, že mřížka, která se běžně měří v pixelech, je měřena v procentech celkové šířky stránky. Skutečná vypočítaná šířka každého sloupce na responzivním webu se změní vždy podle velikosti okna prohlížeče, může tak být různá na každém zařízení, proto je nutné vždy využívat mřížové uspořádání.

Prakticky každý navrhovaný webový design by měl být založen na mřížovém uspořádání. Mřížky stránkám dodávají pocit struktury a zlepšují jejich použitelnost návštěvníkům. Vytváření designu pro takto vytvořené stránky je rychlejší, jelikož jsou možnosti omezeny předem danými vodítky díky mřížce a snadněji se určuje šířka a výška pro jednotlivé prvky na stránce.

Většina mřížových systémů je rozdělena do devíti, dvanácti, šestnácti nebo dvaceti sloupců. V případě méně rozsáhlého webu se není nutné za každou cenu držet daného počtu sloupců. Například v případě využití systému s dvanácti sloupci pro web, který využívá pouze čtyři, by vznikal zbytečně optický rušivý prvek. Pro efektivnější práci s volným místem je vhodné využívat také vodorovných vodících čar, které by měly mít stejný rozestup jako čáry svislé.

Většina moderních nástrojů pro návrh webového designu má v sobě již integrovaný systém mřížového uspořádání, proto se jeho užití dnes téměř nelze vyhnout.

3.8 Prohledávající vzory na stránkách

Ve většině západních kultur je zvykem číst ze shora dolů a zleva doprava. Tomuto zvyku musí být přizpůsoben i webový design. [24]

Během studie sledování pohybu očí na webu (Obr. 15) bylo zjištěno, že oči nejprve rychle prohlédnou stránku, aby zjistily, co je na ní zajímavé, než ji začnou číst. Studie zjistila, že prohledávací vzor stránky se drží dvou tvarů – „F“ a „Z“, proto by měl být každý kvalitní webový design těmto vzorům přizpůsoben. [25]

3.8.1 Vzor F

Tento vzor se vztahuje především na textově založené webové stránky. Čtenář nejprve prohlíží levou stranu stránky od shora dolů, kdy hledá zajímavá klíčová slova. Jakmile nějaké nalezne, tak pokračuje v čtení zleva doprava. Výsledkem je prohlížení ve tvaru písmena „F“ nebo také „E“.



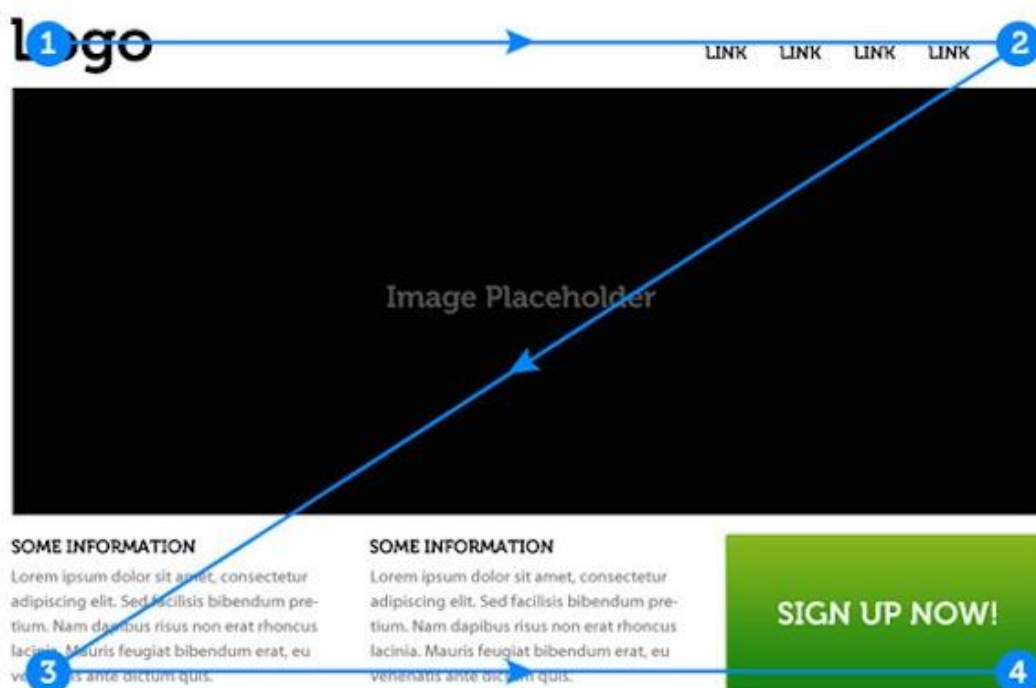
Obrázek 15: Výsledek studie sledování očí v "F" vzoru

Tento poznatek je dobré využít při návrhu webové stránky, a to díky použitím krátkých tučných nadpisů umístěných vlevo, nejlépe v seznamu s odrážky.

3.8.2 Vzor Z

Vzor Z se uplatňuje především na stránkách reklamních nebo propagujících nějaký výrobek, které obvykle nejsou prezentovány v dlouhých textových blocích. Návštěvník nejprve prohlédne horní oblast zleva doprava, kde jsou obvykle důležité informace, a poté pokračuje uhlopříčně do spodního rohu, kde prohlédá spodní oblast stránek.

Webový design je často přizpůsobován tomuto vzoru, jako v případě ukázky designu (Obr. 16). V tomto případě lze pozorovat, že nejdůležitější informace jsou umístěny v rozích horní a dolní lišty stránek. Těmito informacemi jsou postupně logo firmy v levém horním rohu, tlačítko registrace v pravém horním rohu, seznam informací v spodní levé části stránky a nakonec přihlašovací tlačítko v pravém dolním rohu. Všechny tyto prvky jsou rozestaveny tak, aby kopírovaly tvar písmena „Z“.



Obrázek 16: Ukázka rozložení stránky v "Z" vzoru

Zmíněnými informacemi o prohlídacích vzorech uživatelů na webových stránkách se je vhodné řídit a využít je při vytváření přehledného designu, ve kterém uživatel okamžitě získá potřebné informace.

3.9 Responzivní a adaptivní design

V dnešní době je velmi důležité, aby webové stránky byly přístupné a především použitelné i z mobilních zařízení. Mobilní zařízení velmi často používají menší a různě velké displeje, jež jsou nyní převážně dotykové, proto je nutné webové stránky přizpůsobit pro tato zařízení. Možností, jak toho dosáhnout je vytvoření responzivního nebo adaptivního designu.

Principem **responzivního designu** je posílání stejné stránky ze serveru na všechna zařízení. Stránka je vykreslena až na straně klienta podle typu zařízení a rozlišení displeje. Základem je využití „*media query*“² v CSS, kterými se nadefinuje, jaké styly se použijí při daném rozlišení pomocí breakpointů. Responzivní design využívá prvek „*Grid based layout*“ neboli mřížové uspořádání, které pomáhá definovat rozložení prvků na stránce. Rozlišují se tři typy layoutů:

- *Basic fluid layout* – plynulé přizpůsobení velikosti okna.
- *Adaptive layout* – nadefinováno několik layoutů pomocí breakpointů.
- *Responsive layout* – kombinace Basic fluid a Adaptive layout.

Při vytváření responzivního designu se využívají relativní jednotky, aby se jednotlivé prvky mohly přizpůsobovat velikosti obrazovky a bylo tím zajištěno správné zobrazení obsahu na všech typech zařízení.

Adaptivní design využívá stejných technik, ale již na straně serveru je rozhodnuto, jaký obsah bude doručen. Rozhodnutí se provádí pomocí detekce vlastností a schopností zařízení, která je prováděna buď již na serveru, nebo lokálně u uživatele. Adaptivní design je definován do třech vrstev:

- HTML dokument.
- Kaskádové styly.
- JavaScript upravující funkčnost stránek.

Při optimalizaci stránek pomocí adaptivního designu může být upravena jakákoliv vrstva, díky čemuž může vzniknout několik verzí webových stránek. [26]

² Media query je využíváno k deklaraci CSS dle nadefinovaných podmínek. Více informací lze nalézt na stránkách <http://www.vzhurudolu.cz/prirucka/css3-media-queries>.

3.9.1 Základní principy tvorby webového designu pro mobilní zařízení

Otestovat web na více typů zařízení

Každý webový design musí být otestován na různých typech zařízení, jako jsou tablety nebo mobilní telefony s odlišnými operačními systémy a velikostmi displeje. Jedním řešením je použití emulátorů na počítači, které nemusí vždy vytvářet stejný pocit, jaké zařízení poskytuje ve skutečnosti. Takovým emulátorem je například *mobilephoneemulator.com*³. Druhým řešením je otestování webu na skutečných zařízeních.[27]

Každý prvek musí být funkční

Při tvorbě webových stránek musí být dán důraz na použití prvků, které jsou funkční na nejrozšířenějších typech zařízení. Například podpora Flash je nyní velmi omezena pro většinu mobilních prohlížečů, i přestože je Flash často využíván na mnoha webech.

Používat krátké URL adresy

Psaní dlouhých a složitých URL adres na mobilních telefonech je nepohodlné a zdlouhavé, proto je vždy lepší používat pouze optimalizované krátké a výstižné URL.

Nedávat textové odkazy blízko sebe

Mobilní zařízení jsou v nynější době především dotyková, proto je nutné odkazy v textu umísťovat v dostatečných rozestupech, aby je bylo možné pohodlně otevřít dotykem. Zároveň tyto odkazy nesmí být příliš krátké, jinak bude v případě především menších displejů jejich použití složité. Odkazy tak mohou mít i tři až pět slov pro snadnější užití.

Používat skrývání navigace

Pro zobrazení většího obsahu na mobilních zařízeních je vhodné využívat skrytí navigační nabídky a jeho zobrazení bude pouze na vyžádání například pomocí CSS nebo JavaScriptu (Obr. 17).

³ *mobilephoneemulator.com* je online emulátorem mobilních zařízení dostupný na <http://mobilephoneemulator.com/>.



Obrázek 17: Skrytí navigační nabídky pomocí CSS a JS

Optimalizovat rozměry obrázků

Obrázky na webových stránkách musí být vždy nejprve otestovány a případně optimalizovány, aby se správně zobrazovaly i na mobilních zařízeních bez nutnosti posouvání do stran. Především se nesmí použít příliš velký obrázek ve vrchní části stránek, jež by mohl zakrýt důležité prvky stránek a celkově působit špatným dojmem.

Optimalizovat velikost obrázků

Velikost souboru a především obrazových musí být pro mobilní zařízení dobře optimalizována, jelikož se často využívá mobilního internetu, který nemusí být dostatečně rychlý, uživatel jej často využívá ve chvílích, kdy rychle potřebuje získat potřebné informace. Pokud budou obrázky příliš velké, bude se stránka příliš dlouho načítat.

Při vytváření webu pro mobilní zařízení je doporučeno se vyhnout následujícím prvkům:

- Flash – většina mobilních zařízení již oficiálně Flash nepodporují.
- Rámce – i když jsou široce podporovány, tak je jejich použití na mobilních zařízeních nepohodlné.
- Tabulky – velikost tabulek se velmi špatně optimalizuje pro menší velikosti displejů při jejich zachování dobré čitelnosti.
- Absolutní měřítka – velikost všech prvků by měla být uváděna pomocí procentuálního poměru a nepoužívat absolutní hodnoty jako pixely nebo milimetry. Na každém jinak velkém displeji by jinak zobrazovaly obrázky jinak.

- Fonty – používat pouze fonty, které mají širokou podporu napříč různými mobilními zařízeními.

Použití JavaScript

Při vytváření webu pro mobilní zařízení musí být použití JavaScript ke změně zobrazení nebo chování stránky využito vždy s rozvahou. JavaScript se obvykle využívá k rozhodnutí jaké zobrazit reklamy nebo jaká rozlišení obrázků zobrazit na stránce.

Na mobilních stránkách se JavaScript využívá třemi typy implementací:

- **Adaptivní JavaScript:** Všem zařízením je dodán stejný obsah HTML, CSS a JavaScript. Jakmile je spuštěn JavaScript na zařízení, tak je chování nebo zobrazení stránky změněno.
- **Dynamicky dodaný JavaScript:** Všechna zařízení získává stejné HTML, ale JavaScript je dodán z URL, která dynamicky dodává různý JavaScript kód podle uživatelského agenta použitého zařízení.
- **Kombinovaná detekce:** V této implementaci web využívá detekce typu zařízení jak pomocí JavaScriptu, tak ze strany serveru, díky kterému je dodán rozdílný obsah různým zařízením.

Adaptivní JavaScript

Při této implementaci URL dodává stejný obsah HTML, CSS, obrázků i JavaScript všem zařízením. Jakmile je spuštěn JavaScript na zařízení, změní se zobrazení nebo chování stránky.

Příkladem může být webová stránka dodávající všem zařízením stejné HTML obsahující `<script>` element, který vyžaduje externí URL dodávající JavaScript. Všechna zařízení vyžadující URL JavaScriptu získají stejný kód. Jakmile je kód spuštěn, tak JavaScript detekuje typ zařízení a případně upraví stránku, aby obsahovala například zmenšené obrázky určené pro mobilní zařízení. Tato metoda je doporučována vyhledavačem *Google*. [28]

Dynamicky dodaný JavaScript

U této implementace je všem zařízením dodáváno stejné HTML obsahující `<script>` element, který zahrnuje externí JavaScript soubor, který může mít odlišný

obsah. JavaScript kód je tak dynamicky dodáván podle požadavku uživatelského agenta. Tato metoda není již doporučována z důvodu obtížnější indexace obsahu a větší šance na duplikaci obsahu. [28]

Kombinovaná detekce

Kombinovaná detekce je nastavením, ve kterém server spolupracuje s JavaScript na straně klienta, aby získal informace o zařízení a případně upravil dodávaný obsah. [29]

Příkladem je webová stránka, jež dovoluje zvolit změnu zobrazení obsahu podle typu zařízení. V takovém případě stránka může obsahovat JavaScript detekující rozměry obrazovky, které jsou poslány serveru, ten upraví kód posílaný do zařízení. Obvykle JavaScript ukládá detekované informace o zařízení ve formě Cookies, jež se načtou při následujících návštěvách stránek ze stejného zařízení. [30]

Výše zmíněné informace jsou pouze základními principy při vytváření responzivního designu cíleného pro multiplatformní použití. Jeho praktické aplikování a tvorba je znázorněna v následujících kapitolách této práce.

4 Kaskádové styly

„CSS neboli kaskádové styly je jazykem pro popsání zobrazení webových stránek, zahrnující barvy, grafické rozvržení a fonty. Umožňuje přizpůsobit vzhled různým typům zařízení, jako jsou velké obrazovky, malé obrazovky nebo tiskárny. CSS je nezávislý na HTML, a proto může být použit s kterýmkoliv značkovacím jazykem na základě XML. Oddělení HTML od CSS umožňuje jednodušší správu stránek, sdílený stylů napříč stránkami a upravovat stránky na míru různým prostředím. Toto je označováno jako oddělení struktury nebo obsahu od prezentace.“ [31]

V dnešní době je téměř nemožné vytvářet moderní webový design, který by nevyužíval kaskádových stylů. Ty napomáhají k jednotnému definování vzhledu stránek a jednotlivých prvků. Jejich použití není složité a existuje mnoho nástrojů usnadňujících jejich tvorbu a aplikaci. Před podrobnějším popisem jejich aplikace je nejprve nutné se seznámit s jejich rozdělením, možnostmi a nejzákladnějšími prvky, které je nutné znát. Podrobné informace k jednotlivým prvkům kaskádových stylů, včetně praktických příkladů jsou obsaženy ve webovém portálu, jež je součástí této práce.

4.1 Aplikování CSS

K aplikování CSS do HTML se využívá tří metod:

- v textu (In-line);
- vnitřní
- a vnější.

V textu

Jedná se o metodu, která je aplikována přímo do HTML kódu pomocí speciálních atributů. Správný přístup k psaní HTML kódu je jeho oddělení od stylování, aby byl snadno čitelný a lehce editovatelný, proto není vhodné metodu in-line používat.

Vnitřní

Vložené, nebo vnitřní styly jsou použity pro celou stránku. Všechny styly jsou obklopeny stylovými tagy uvnitř hlavičky stránky.

Přestože se jedná o pokročilejší metodu vytváření stylu na stránce, tak přesto se nejedná o metodu, která by oddělila HTML kód a CSS do dvou separátních souborů, a proto není vhodné tuto metodu využívat u vícestránkových webových stránek.

Vnější

Vnější metoda je využívána pro vícestránkové weby. Tato metoda odděluje HTML a CSS kód do dvou samostatných souborů. Soubor s koncovkou „css“ obsahující CSS kód musí být nedefinovaný v hlavičce HTML kódu, aby byl použit. Jedná se nejčastěji preferovanou metodu práce s kaskádovými styly. [32]

4.2 Selektory, vlastnosti a hodnoty

Zatímco HTML využívá tagy, tak CSS má takzvané selektory. Selektory jsou názvy stylů použité ve vnějších i vnitřních definicích kaskádových stylů.

Každý **selektor** má ve složených závorkách definované **vlastnosti** pomocí slov jako „color“, „font-weight“ nebo „background-color“.

Hodnota jednotlivým vlastnostem se udává pomocí **dvojtečky** a následné **mezery**, nikoliv pomocí znaku „rovná se“.

Délky a procenta

V CSS se využívá mnoho specifických jednotek pro určité vlastnosti, ale pouze pár základních jednotek pro číselné hodnoty vlastností. Nejčastěji se využívají následující jednotky:

- px – jednotka pro pixely;
- pt – jednotka pro body;
- em – jednotka udávající násobek velikosti fontu
- % - jednotka v procentech.

Také se lze setkat s jednotkami cm (centimetr), mm (milimetr) nebo in (palec). Výjimku v definici tvoří hodnota „nula“, u které není nutné definovat jakoukoliv hodnotu. [33]

4.3 Definice barev

CSS nabízí škálu více než šestnácti milionů barev. Ty mohou být definovány pomocí předefinovaných názvů, RGB a RGBA hodnot, hexadecimálního kódu, HSL nebo HSLA.

Předefinované názvy barev

Jedná se o 140 barev, které jsou uloženy v HTML a CSS specifikaci. Příkladem jsou například: „*black, red, fuchsia*“ nebo také „*transparent*“ pro průhlednou barvu.

RGB a RGBA

RGB barva je specifikována pomocí tří barevných parametrů – červené, zelené a modré. Každý parametr definuje intenzitu dané barvy pomocí hodnot od 0 do 255 nebo pomocí procentuálních hodnot - *rgb(0,0,255)*. RGBA má o jeden parametr více, a to takzvaný Alfa kanál, který určuje průhlednost objektu a je udáván jako číslo v rozmezí od 0 do 1 - *rgba(255, 0, 0, 0.5)*

Hexadecimální kód

Jedná se o šestnáctkový číselný systém, který nabývá hodnot od 0 do f. U této metody se vždy využívá prefixu # následovaným tří nebo šestimístním kódem - *#RRGGBB*, kde RR značí červenou barvu, GG zelenou barvu a BB modrou barvu. Třímístná verze kódu je pouze zjednodušenou verzí té šestimístní, která nabízí více barevných odstínů.

HSL a HSLA

Od verze CSS3 je možné k definování barev použít také HSL barvy, jejíž parametry se skládají z odstínu, sytosti a světlosti - *hsl(120, 100 %, 50 %)*. Odstínem je barva na barevném kole, která nabývá hodnot od 0 do 360, kde 0 nebo 360 je červená, 120 je zelená a 240 modrá. Sytost je udávána v procentuální hodnotě od 0 % do 100 %, kde 100 % značí plnou sytost barvy. Posledním parametrem je světlost,

kteřá je také udávána v procentech a 0 % znamená černou barvu a 100 % barvu bílou.

Použití barev

K použití barev se využívá dvou vlastností, a to „*color*“ a „*background-color*“, oboje psané americkou angličtinou bez „u“. Tyto vlastnosti mohou být aplikovány na většinu elementů HTML. [34]

4.4 Úprava textu

Každý text umístěný na webové stránce může pomocí kaskádových stylů nabývat různých tvarů a velikostí se širokou škálou rozdílných vlastností.

font-family

Tímto parametrem se nastavuje samotný font písma, jako je například *Arial*, *Verdana* nebo *Times New Roman*.

Aby se uživateli zobrazil zvolený font, tak ho musí podporovat jeho internetový prohlížeč, proto není vhodné používat příliš originální nebo méně běžné písemné fonty. Při vytváření webových stránek se používá metody takzvaných bezpečných fontů. Hlavní logikou je specifikování více fontů oddělených čárkou, kdy je nejprve použit první uvedený font; jestliže není podporován, použije se ten následující uvedený. Tímto způsobem může být uvedeno neomezené množství fontů, ale z důvodu optimalizace a přehlednosti kódu jsou doporučovány tři typy fontů, z nichž tím posledním by měla být uvedena skupina fontů, z nichž prohlížeč sám vybere ten, který podporuje.

font-size

Parametr *font-size* slouží k nastavení velikosti fontu. Tento samotný parametr není vhodné používat k rozlišení odstavce a jejich nadpisů, k tomu by vždy mělo být použito také hlavičkování nadpisů pomocí *h1*, *h2*, *h3* a podobně.

font-weight

K definování tučného nebo normálního textu se využívá parametru *font-weight*. Nejčastějšími hodnotami jsou *bold* a *normal*, ale mohou být použity i jiné hodnoty jako například *bolder* nebo *lighter*, poté číselné hodnoty od 100 do 900.

font-style

Pro nastavení písma kurzívou se používá parametru *font-style*, který může nabývat tří základních hodnot – *normal*, *italic* a *oblique*.

text-decoration

Text-decoration určuje, zda bude text podtržený, přeškrtnutý nebo nadtržený a využívá se nejčastěji u odkazů.

text-transform

Parametr *text-transform* mění písmo na velké nebo malé. Jako hodnotu lze nastavit čtyři možnosti:

- *capitalize* – každé první písmeno slova bude velké;
- *uppercase* – každé písmeno bude velké;
- *lowercase* – každé písmeno bude malé
- *none* – velikost písma není předem specifikovaná.

Rozestup, řádkování a zarovnání textu

K nastavení rozestupu textu se využívá dvou parametrů, a to *letter-spacing* pro nastavení rozestupu písmen a *word-spacing* pro nastavení rozestupu slov. Oba parametry mohou nabývat hodnot *normal* nebo číselnou hodnotu pro normální a ručně definované rozestupy.

K definování řádkování textu slouží parametr *line-height*, který nastavuje výšku jednotlivých řádku bez změny velikosti písma. Parametr nabývá buď číselné hodnoty, jež udává násobek velikosti písma, vzdálenost, procenta, nebo hodnotu *normal*. K zarovnání textu je definován parametr *text-align*. [35]

4.5 Vnitřní a vnější okraje prvků

K odsazení jednotlivých prvků v textu se nejčastěji využívá parametrů *margin* a *padding*. Parametr *margin*, určuje vnější okraj prvku, zatímco *padding* určuje vnitřní okraj prvku. [36][37]

4.6 Ohraničení prvků

Ohraničení může být aplikováno na většinu prvků v těle HTML.

K vytvoření ohraničení prvku slouží parametr *border-style*.

K definování šířky ohraničení je definován parametr *border-width*, nejčastěji udáván v pixelech nebo procentech.

Posledním důležitým parametrem k definování ohraničení je *border-color*, sloužící k nastavení barvy. Hodnoty se nastavují stejně jako v případě definování barvy textu. [38]

Box Model

Vnější okraje, vnitřní okraje a ohraničení jsou součástí *takzvaného Box modelu*. *Box model* v podstatě obaluje každý HTML prvek, a to následujícím způsobem: uvnitř je základní prvek, okolo něj se nachází vnitřní okraj, okolo něj ohraničení a nakonec je vše obklopeno vnějším okrajem (Obr. 18).



Obrázek 18: Ukázka rozložení box modelu

- **Prvek** – základní prvek obsahující text či obrázek.
- **Vnitřní okraj** – vnitřní okraj základního prvku. *Padding* je průhledný.
- **Ohraničení** – hranice okolo základního prvku a vnitřního okraje.
- **Vnější okraj** – vnější okraj základního prvku. *Margin* je průhledný.

Není nutné použít všechny tyto parametry, ale dobré vědět, že je lze použít na každý prvek webové stránky.

4.7 Základní prvky responsivního webového designu v CSS

V předchozí kapitole byly vysvětleny a představeny základní prvky kaskádových stylů. V následující kapitole jsou představeny další základní prvky a informace o nich, ale v tomto případě se jedná o prvky kaskádových stylů používané při tvorbě responsivního designu. Každé webové stránky by měly být použitelné na všech platformách od stolních počítačů, tabletu až po mobilní telefony. Vytváření specializovaných webových stránek pro všechny platformy zbytečně pracné a obtížné na údržbu. Nejlepším řešením je právě použití responsivního designu, kterého lze docílit použitím následujících prvků v kaskádových stylech.

4.7.1 Viewport

Samotný viewport je pro uživatele viditelná oblast na stránce.

Před příchodem mobilních telefonů a tabletů byly webové stránky designovány pouze pro počítačové obrazovky, a proto bylo běžné používat statický design s pevnou velikostí.

Příchod mobilních zařízení donutil tvorbu webového designu ke změně. Stránky s pevnou velikostí byly příliš velké, aby se vešly na displeje mobilních zařízení, v té době byla jediným známým řešením změna měřítka celých webových stránek.

V HTML5 byla představena nová metoda, která umožňuje ovládat viditelnou oblast stránek neboli viewport, a to pomocí meta tagů.

Pro použití meta tagů je nutné zahrnout následující element na všechny webové stránky:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Stránky optimalizované pro různá zařízení musejí v hlavičce dokumentu obsahovat `<meta>` element viewport. Pomocí viewport tagů jsou prohlížeči předány instrukce, jak kontrolovat rozměry a změnu měřítka stránek.

Parametr `width=device-width` nastavuje šířku stránky podle šířky displeje zařízení, která se může u každého zařízení lišit. To dovoluje stránce přeskupit obsah, aby seděl přesně na různě velké obrazovky zařízení, a to ať se jedná o malé mobilní telefony nebo velké počítačové obrazovky.

Parametr *initial-scale=1.0* nastavuje počáteční úroveň přiblížení stránky po prvním načtení prohlížečem. Nastavením hodnoty na 1 se nastaví vazba 1:1 mezi pixely CSS a pixelů nezávislých na zařízení bez ohledu na orientaci přístroje. Díky tomu se obsah stránek naplno přizpůsobí při otočení přístroje na šířku či výšku.

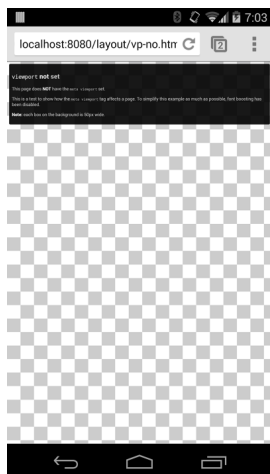
Při použití metody viewport je nutné dodržovat následující pravidla: [39]

- Nepoužívat velké prvky s pevnou šířkou.
- Obsah stránky by neměl být závislý na šířce zadané absolutní hodnotou.
- Používat CSS media queries pro použití různých stylů na malé i velké displeje.

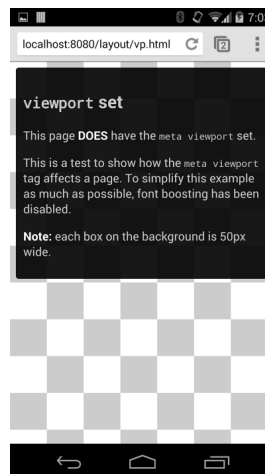
Při vytváření stránek pro mobilní zařízení se snadno může stát, že má obsah stránek (například obrázků) větší šířku než zrovna viditelná oblast. V takovém případě je uživatel nucen použít horizontální posouvání, které by se správně nemělo nikdy objevit. Proto je vhodné upravit obsah stránek, aby se celý vešel do viditelné oblasti stránek a uživatel nebyl nucen posouvat obsah do stran.

Jelikož se rozměry, šířka i počet pixelů displeje liší na většině mobilních zařízeních, tak by obsah stránek nikdy neměl být závislý na pevně dané šířce.

Nastavení velké absolutní hodnoty šířky některého z prvků na stránce může mít následek, že vzájemný velikostní nepoměr s ostatními prvky stránky, který způsobí například nečitelnost textu (Obr. 19). Místo pevně dané hodnoty by vždy měla být použita hodnota relativní, jako jsou například procenta. Použitím relativní hodnoty lze docílit čitelného textu na různě velkých mobilních zařízeních (Obr. 20).



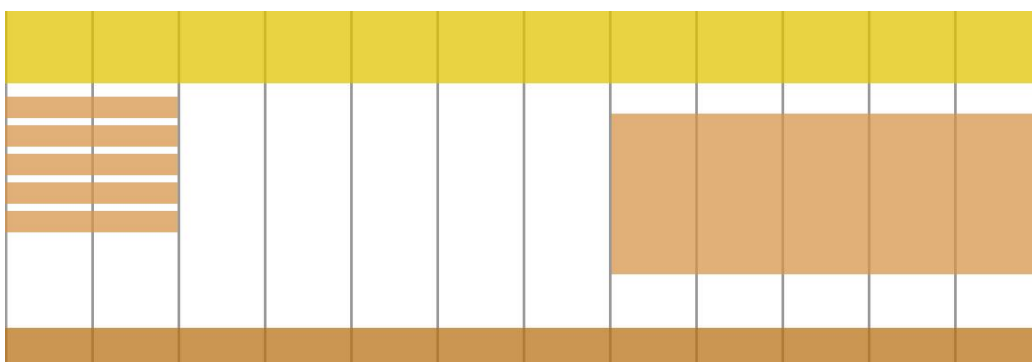
Obrázek 20: Použití absolutní hodnoty



Obrázek 19: Použití relativní hodnoty

4.7.2 Grid-View

Mnoho webových stránek je založeno na mřížovém uspořádání neboli *grid-view*. To znamená, že stránka je rozdělena do určitého počtu sloupců a každý prvek na stránce je zasazen do mřížky s relativní hodnotou šířky a výšky místo použití rozměrů na bázi obrazových bodů. Díky tomuto nastavení prvky při změně velikosti obrazovky upraví svoji velikost ve stanoveném poměru vůči svému rodičovskému kontejneru.[40] Použití této metody je velmi nápomocné, jelikož lze díky tomu snadněji umístit jednotlivé prvky obsahu na stránku, aby byl zachován čistý a snadno udržovatelný layout. Ukázka použití (Obr. 21):



Obrázek 21: Ukázka Grid-View

Typů responzivních grid-view je velké množství a nabízejí různé počty sloupců. Nejčastěji se využívá formát od dvou do dvanácti sloupců. Je ovšem celá řada designových nástrojů pro tvorbu *grid-view* z nichž každý nabízí různé počty

sloupců. Při tvorbě lze díky tomu zvolit různé počty sloupců pro různá zařízení, aby bylo zachováno správné zobrazení všech prvků i na menších obrazovkách. [41]

Vytvoření responzivního Grid-View

Nejprve se je nutné ujistit, že všechny HTML prvky mají nastaveny parametr *box-sizing* v CSS na hodnotu *border-box*. To zajistí zahrnutí ohraničení a vnitřního okraje do celkové šířky a výšky prvků.

Pro vytvoření obsahu s již dříve zmíněnými dvanácti sloupci, které jsou pro responzivní design nejčastější, je nutné nejprve spočítat procentuální hodnotu šířky jednotlivých sloupců, která se spočítá vydělením celkové šířky počtem sloupců:

$$100 \% / 12 = 8.33 \%$$

Poté se vytvoří třída pro každý sloupec pomocí `class="col-"` a čísla sloupce s hodnotou rozpětí úseku.

4.7.3 Media Queries

Media queries je technologie představená v CSS3. Jedná se o jednoduché filtry, které mohou být aplikovány na kaskádové styly. Díky nim lze snadno měnit styly založené na vlastnostech zařízení podle typu displeje, výšky, šířky, orientace nebo i rozlišení.

Filtr se nastavuje pomocí pravidla *@media*, který je aplikovaný pouze v případě splnění předem dané podmínky.

Použití *media query* v závislosti na velikosti viewport

Media queries umožňuje vytvářet responzivní design pomocí aplikování různých stylů podle velikosti displeje zařízení, na kterém je webová stránka zobrazována. *Media queries* je především pevně daná syntaxe k vytváření pravidel pro jednotlivá zařízení dle jejich vlastností.

Pravidel kaskádových stylů je poměrně vysoký počet, ale nejdůležitějšími a především nejčastějšími pravidly jsou *min-width*, *max-width*, *min-height* a *max-height* pro určení maximální nebo minimální výšky a šířky.

Použití relativních jednotek

Hlavním znakem responzivního designu je jeho plynulost a proporcionalita na rozdíl od klasických webů s pevně danou šířkou. Použití relativních jednotek rozměrů zjednoduší vzhled a především zabrání vytvoření prvků, které by byly příliš velké pro viditelnou oblast stránek.

Například nastavení šířky na 100 % pro nejvyšší úroveň div zajistí, že vždy přizpůsobí šířku viditelné oblasti a nikdy nebude na displeji příliš velký, nebo naopak malý. Použití relativních jednotek zároveň umožní webovým prohlížečům zobrazit obsah podle úrovně přiblížení, kterou si uživatel sám nastavil. Díky tomu není potřeba použití horizontálního posouvání na stránce. [42]

Breakpoint

Hlavním prvkem při použití *media queries* jsou takzvané *breakpointy*, díky kterým lze nastavit různý styl a chování designu podle vlastností zařízení, na kterém jsou webové stránky zobrazovány.

Při vytváření breakpointů by se měla dodržovat následující pravidla:

- Vytvářet breakpointy podle obsahu, nikoliv dle zařízení či značek.
- Nejprve vytvářet design pro nejmenší zařízení, poté postupně přidávat styly pro ty větší, která budou časem přibývat.
- Udržovat řádky textu o maximální délce 70 až 80 znaků.

Počet breakpointů by neměl být zbytečně vysoký. Vytvoření speciálního *breakpointu* podle zařízení, značky, modelu nebo operačního systému by bylo velmi náročné pro budoucí údržbu stránek. Místo toho by měl sám obsah definovat přizpůsobení rozložení jednotlivých prvků dle jejich kontejneru.

Umístění hlavních breakpointů

Přizpůsobit obsah nejprve malým displejům a poté rozšiřovat obraz dokud není nutný *breakpoint*. Tato technika umožňuje správně definovat použití breakpointů podle obsahu a vytvořit pouze nejmenší nutný počet *breakpointů*.

Umístění vedlejších breakpointů

Kromě vkládání hlavních *breakpointů*, kdy se rozložení stránky výrazně změní, je také vhodné vytvářet vedlejší *breakpointy* pro menší změny. Například mezi dva hlavní *breakpointy* může být vložen *breakpoint* upravující ohraničení, vnitřní okraje nebo velikost písma k docílení přirozenější a plynulejší změny rozložení.

Přizpůsobení textu pro čtení

„Cokoliv od 45 do 75 znaků je obecně považováno za uspokojivou délku řádku pro stránku obsahující patkový text zarovnaný do jednoho sloupce. Jako ideální délka jednoho řádku je považováno 66 znaků včetně mezer. Při použití více sloupců je lepší použít mezi 40 až 50 znaky.“ [43]

Možné následky špatného přizpůsobení délky řádku:

- **Příliš dlouhý řádek** – jestliže je řádek textu příliš dlouhý, tak se budou oči čtenáře těžce soustředit na text. To je způsobeno tím, že délka řádku neumožňuje snadno poznat, kde řádek začíná a kde končí. Zároveň může být obtížné pokračovat na správném řádku u větších bloků textu.
- **Příliš krátký řádek** – jestliže je řádek příliš krátký, tak se musí čtenářovy oči příliš často vracet, tím je narušen čtenářův rytmus. Příliš krátké řádky mají také za následek stresování čtenáře, jež ho nutí začít číst na dalším řádku, i když ještě nedokončil ten předchozí; potencionálně tak mohl vynechat některé důležité informace. [44]

Ukazuje se, že podvědomí je povzbuzováno při každém přechodu na nový řádek, pokud se to nestává příliš často. Na začátku každého řádku je čtenář soustředěn, ale soustředění postupně opadá s postupným čtením řádku.

Pro udržení pozornosti čtenářů je doporučováno udržovat text v rozmezí 50 až 75 znaků na řádek. [45]

Nikdy úplně neschovávat obsah

Opatrnost je nasnadě při určování, zda některý obsah zobrazit či skrýt z důvodu velikosti obrazu. Obsah by nikdy neměl být schován jen z důvodu, že se nevejde do viditelné oblasti. Velikost obrazovky často není konečným určením to-

ho, co uživatel pravděpodobně chce. Pokud bude chybět některá důležitá informace, kterou uživatel hledá, ale je skryta pouze kvůli velikosti obrazovky, tak návštěvník pravděpodobně odejde hledat jinam.

Vždy vytvářet design nejprve pro mobilní zařízení

Vytvářet design nejprve pro mobilní zařízení neboli „*Mobile first!*“ znamená, že by měl být design tvořen nejprve pro mobilní zařízení a až poté pro jiná větší zařízení. Díky tomuto pravidlu se stránka zobrazí rychleji na menších zařízeních.

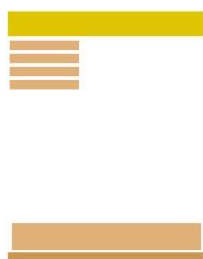
K aplikování toho pravidla je nutné provést některé změny v CSS. Místo pravidla, že se styl změní, jakmile je šířka obrazovky menší než 1224px, se použije pravidlo, aby se styl změnil při velikosti obrazovky větší než 1224px. Tímto způsobem se nejprve zobrazí styl určený pro menší zařízení.

Přidání nového breakpointu

Do CSS může být přidáno neomezené množství breakpointů. V následujícím příkladu jsou přidány breakpointy pro oddělení stolních počítačů, tabletů a mobilních telefonů. (Obr. 22, Obr. 23 a Obr. 24)



Obrázek 24: Rozložení na stolním počítači



Obrázek 23: Rozložení na tabletu



Obrázek 22: Rozložení na mobilním telefonu

Vytvoření nového breakpointu se provádí pomocí dalšího media queries odstavce s hodnotou větší než 768px. Nový odstavec bude mít za následek, že na obrazovkách s velikosti mezi 768px a 1224px se zobrazí nově definovaný styl pro tablety.

Orientace: na výšku / na šířku

Media queries mohou být také použity ke změně rozložení stránky podle orientace prohlížeče.

Například lze nastavit CSS vlastnosti, které budou aplikovány pouze, pokud okno prohlížeče je širší než vyšší neboli je zobrazeno na šířku.

CSS3 @media Rule

Jedná se o seznam pravidel pro různé styly a různá zařízení. Tato pravidla se nazývají media queries.

Media queries kontrolují možnosti zařízení a mohou být využita ke kontrolování dalších věcí, jako jsou šířka a výška *viewportu*, šířka a výška zařízení, orientace zařízení, rozlišení a další.

4.7.4 Obrázky

Jestliže je vlastnost *width* neboli šířka nastavena na 100 %, tak bude obrázek responzivní a bude měnit své měřítko.

Při použití hodnoty *width* na 100 % je možné zvětšit velikost obrázku nad jeho originální velikost. Vhodnějším řešením je přidání dalšího parametru, a to *max-width*, neboli maximální šířky.

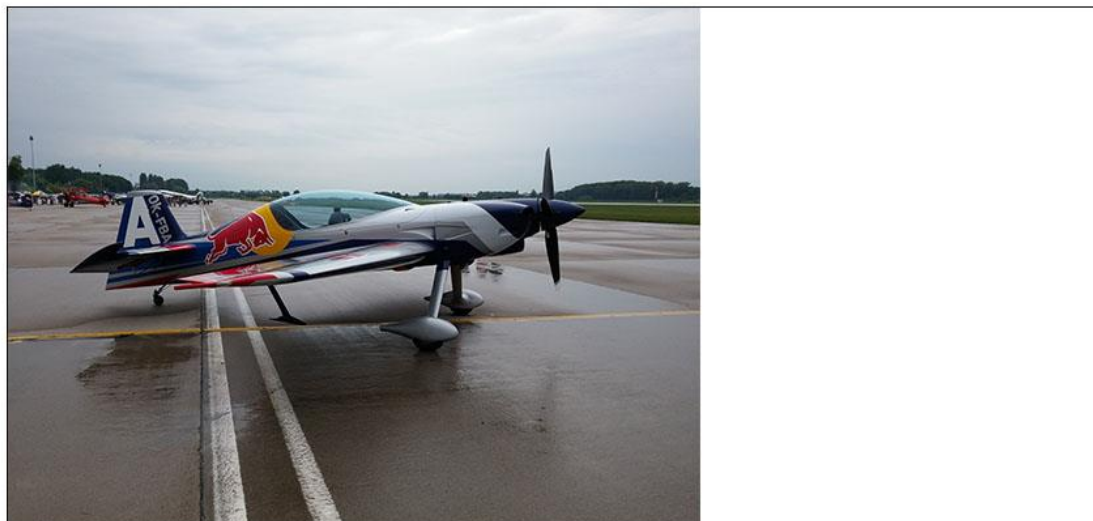
Použití vlastnosti max-width

Jestliže je *max-width* nastaven na 100 %, tak bude obrázek v případě potřeby zmenšen, ale nikdy nebude zvětšen na velikost větší, než je jeho původní.

Obrázky na pozadí

Obrázky na pozadí mohou také měnit svou velikost a poměr. K této změně se využívá u responzivního designu následujících metod:

1. Jestliže je vlastnost *background-size* neboli velikost pozadí nastavena na *contain*, bude obrázek na pozadí měnit své měřítko dle oblasti obsahu, avšak obrázek si ponechává původní poměr stran (Obr. 25).



Obrázek 25: Ukázka použití hodnoty *contain*

2. Jestliže je vlastnost *background-size* nastavena na „100 % 100 %“, tak bude obrázek roztažen do všech stran, aby zakryl celou oblast (Obr. 26).



Obrázek 26: Ukázka použití hodnoty 100 % 100 %

3. Jestliže je *background-size* vlastnost nastavena na *cover*, tak je obrázek na pozadí zvětšen nebo zmenšen tak, aby zakryl celou oblast. Hodnota *cover* ponechává původní poměr stran, ale některé části obrázku nemusí být vidět (Obr. 27).



Obrázek 27: Ukázka použití hodnoty cover

Použití různých obrázků podle zařízení

Velké obrázky mohou vypadat dobře na velkých obrazovkách, ale mohou být nepoužitelné na těch malých mobilních zařízeních. Zároveň je zbytečné stahovat a načítat obrázek, který se poté bude stejně muset zmenšit. Z těchto důvodů je vhodné použít *media queries* k zobrazení jiného obrázku pro různá zařízení.

Na tomto příkladu je zobrazen jeden obrázek, který se načte na velkých obrazovkách (Obr. 28) a druhý, který se načte na malých mobilních zařízeních (Obr. 29).



Obrázek 28: Obrázek pro velké obrazovky

Obrázek 29: Obrázek pro malá zařízení

Může být použit i parametr *min-device-width* místo *min-width*, který kontroluje šířku použitého zařízení místo šířky prohlížeče. Díky tomu se obrázek nezmění na jiný v případě změny velikosti prohlížeče.

HTML5 <picture> Element

HTML5 představilo nový element <picture>, který dovoluje definovat více než jeden obrázek.

Element <picture> pracuje podobně jako elementy <video> a <audio>. Nastaví se různé zdroje a první zdroj, který odpovídá požadavkům, je použit.

Atribut **srcset** je vyžadován, jelikož udává zdroj obrázku.

Atribut **media** je pouze volitelný a obsahuje hodnoty media queries uvedené v tabulce CSS3 @media Rule. [46]

Atribut <bimg> je také volitelný, ale měl by být vyplněn pro případ, kdy uživatel použije prohlížeč bez podpory elementu <picture>.

4.7.5 Videá

Použití vlastnosti **width**.

Jestliže je vlastnost width neboli šířka nastavena na 100 %, tak bude video přehrávač responzivní a měnit své měřítko.

Při použití hodnoty 100 % je možné zvětšit velikost přehrávače nad jeho originální velikost. Vhodnějším řešením je přidání dalšího parametru, a to *max-width* neboli maximální šířky.

Použití vlastnosti **max-width**.

Jestliže je *max-width* nastaven na 100 %, tak bude video přehrávač v případě potřeby zmenšen, ale nikdy nebude zvětšen na velikost větší, než je jeho původní.

4.7.6 Frameworky

Při vytváření webových stránek není nutné vše psát od úplného začátku, ale lze využít takzvané CSS frameworky, které jsou volně k užití, jsou snadno použitelné a nabízí podporu responzivního designu.

Následující přehled frameworků je vytvořen pro představu o nejznámějších typech a možnostech, které poskytují při vytváření webového designu.

W3.CSS

Jedním z CSS frameworků s podporou responzivního designu je *W3.CSS* vytvořený známým portálem *w3schools.com*. Tento Framework je založen pouze na standardech CSS bez použití jQuery nebo knihoven JavaScript a je volně k užití pro všechny vývojáře. Jeho hlavními výhodami jsou cena, snadné použití, optimalizace mobilních HTML aplikací a jedny kaskádové styly pro všechny typy zařízení ať už mobilní telefony nebo stolní počítače.

Jedinou podmínkou k použití je umístění odkazu webových stránek *w3.css* na webové stránky. [47]

Bootstrap

Pravděpodobně nejpoblárnějším frameworkem používaným k vytváření stránek určených primárně pro mobilní zařízení je v současnosti *Bootstrap*, jenž využívá HTML, CSS a jQuery. Jedná se o technologii, která je volně dostupná pro všechny uživatele.

Bootstrap využívá určitých HTML prvků a CSS vlastností, které vyžadují HTML doctype, proto musí být vždy obsažen na začátku každého projektu.

Podobně jako další moderní frameworky podporující responzivní design je i tento vhodný pro použití na mobilních zařízeních. Všechny styly pro mobilní zařízení jsou obsaženy přímo v knihovně framework a nikoliv v oddělených souborech. [48]

Bootstrap umožňuje nastavovat základní zobrazovací, typografické a odkazové styly. Pro zlepšení zobrazení mezi různými webovými prohlížeči je využíván projekt *Normalize.css*, který napomáhá prohlížečům zobrazit všechny prvky stejně podle moderních standardů. [49]

Celkově poskytuje *Bootstrap* responzivní, přizpůsobitelný *grid systém* s primárním zaměřením na mobilní zařízení. *Grid systém* je tvořen až dvanácti sloupci podle velikosti zařízení nebo viewportu. Dále zahrnuje předefinované třídy

pro jednoduché nastavení rozvržení a takzvané *mixiny*⁴ určené ke generování významových rozvržení.

Skeleton

Dalším oblíbeným frameworkem je *Skeleton*, který využívá pouze kaskádových stylů k vytvoření responzivních webových stránek. *Skeleton* je určen především pro menší projekty, u kterých není potřeba všech možností, jež nabízejí ostatní rozsáhlé frameworky. Díky tomu tento framework vyniká především svou jednoduchostí, kdy upravuje pouze hrstku ze standardních HTML prvků a rozloží stránku do mřížky neboli grid.

Skeleton má v základním nastavení dvanáctisloupcový *grid* s maximální šířkou 960px, který se zmenší při použití menšího zařízení nebo okna prohlížeče. Hodnota maximální šířky může být kdykoliv změněna v jediném řádku CSS a všechny sloupce se tomu přizpůsobí.

Zajištěna je také podpora media queries k obsluhování gridu, také nabízí list dotazů k pohodlné úpravě stylu stránek pro jednotlivé typy zařízení. Dotazy jsou vždy určeny nejprve pro mobilní zařízení. Na tomto principu je vytvořen celý *Skeleton* framework. To znamená, že tyto styly jsou aplikovány na všechna zařízení, ale pouze větším zařízením se styly přizpůsobují. Tato metoda zabraňuje tomu, aby malá zařízení nebyla zbytečně zpomalována nepoužitým CSS. [50]

4.8 Moderní technologie pro práci s CSS

4.8.1 CSS preprocesory

Vytváření kaskádových stylů je poměrně jednoduchý proces, ale s narůstajícím obsahem se stává složitějším a především obtížnějším ke správě. Z tohoto důvodu je vhodné používat *preprocesory*. Jejich užívání je především v posledních letech stále častějším trendem, a z toho důvodu by měly být také zmíněny.

Preprocesory usnadňují práci vývojářům s CSS. Jsou postaveny nad CSS a přidávají do něj zjednodušenou organizaci kódu a některé nové vlastnosti. Poté jsou přímo lokálně na vývojářově pracovní stanici kompilovány do „běžného“ CSS, kterému rozumí webové prohlížeče.

⁴ Mixiny jsou funkce, které umožňují znovupoužití vlastností v celém CSS souboru.

Nejznámějšími a nejpoužívanějšími preprocesory kaskádových stylů jsou nyní **Less** a **Sass**.

Sass je rozšířením CSS, které přidává nové vlastnosti, jako jsou například vnořená pravidla, proměnné, mixiny, dědičnost výběru a další. Vše je překládáno do standardního formátovaného CSS.

„Sass využívá dvou syntaxí. Novější a aktuální syntaxe, uvedená ve verzi Sass3, je známá jako „SCSS“ neboli „Sassy CSS“. Jedná se o podmnožinu syntaxe CSS. To znamená, že každý validní CSS kód je také validní v SCSS. Soubory SCSS využívají příponu „.scss“.

Druhá, starší syntaxe je známá jako členitá syntaxe neboli „Sass“. Je určena lidem, kteří preferují stručnost kódu, před jeho podobností s CSS. Místo závorek a středníků využívá odsazení řádků do konkrétních bloků. I když se již nejedná o primární syntaxi, tak zůstává být podporována. Soubory v členité syntaxi používají příponu „.sass.“ [51]

Less je dalším příkladem preprocesoru, jenž je rozšířením jazyku CSS. Podobně jako *Sass* přidává nové vlastnosti jako proměnné, mixiny, funkce a další techniky, které umožňují vytvářet CSS lépe udržovatelné a rozšiřitelné. [52]

Hlavním rozdílem mezi preprocesory *Less* a *Sass* je způsob, jakým jsou zpracovávány. *Less* je knihovnou JavaScriptu, to znamená, že je zpracováván na straně uživatele, zatímco *Sass* bývá zpracováván až na serveru. Přesto může být také *Sass* nainstalován lokálně a tam být také zpracováván. [53] Pro začátečníky s preprocesory písící hlavně CSS je vhodnější používat *Less*. Pro zkušenější programátory v JavaScriptu a PHP je vhodně využít *Sass*. [42]

Preprocesory *Less* i *Sass* přidávají CSS řadu nových vlastností:

Proměnné

Proměnné umožňují ukládat konstantní hodnotu, která může být později znovu použita v celém CSS.

Zanořování

Zanořování je stylem zápisu, který umožňuje vkládat CSS styly do sebe. Potom se zapíše podobně jako každá jiná vlastnost. Tato funkce preprocesoru je

velmi užitečná, jelikož ušetří mnoho nepříjemností s vypisováním desítek selektorů. Zanořovat do sebe lze jakkoli a kolikrát je potřeba, avšak nejlepší je uplatnit nepsané pravidlo nepřesáhnout maximální počet zanoření čtyři. [54]

Mixiny

Mixiny jsou funkce, které umožňují znovupoužití vlastností v celém CSS souboru. Díky tomu lze místo zdlouhavého prohledávání kódu a opakované změny určité vlastnosti změnit vlastnost pouze uvnitř mixinu. To může být velmi užitečné pro specifické úpravy prvků. Jakmile jsou zavolány mixiny uvnitř CSS selektoru, tak jsou rozpoznány jednotlivé mixin argumenty, a styly uvnitř *mixinu* jsou aplikovány na *selektor*. [55]

Partials

Pomocí *preprocesorů* je možné vytvářet takzvané *parciální* soubory, které obsahují CSS kód, jenž může být použit v jiném souboru. Díky tomu lze modulovat CSS a udržovat ho jednodušším k údržbě. *Partial* je například *Sass* nebo *Less* soubor, který má v názvu na začátku podtržítka. Pomocí podtržítka *preprocesor* zjistí, že se jedná pouze o *parciální* soubor a nemá být generován do CSS souboru. Tyto *parciální* soubory jsou využívány ve vlastnosti *@import*. [56]

Příklad použití *parciálních* souborů ve vlastnosti *@import*:

```
@import "partials/buttons";
```

```
@import "partials/typography";
```

```
@import "partials/textboxes";
```

@IMPORT

CSS má možnost importu, který umožňuje rozdělit CSS na menší a lépe udržovatelné části. Jedinou nevýhodou je, že s každým použitím *@import* v CSS je vytvořen nový HTTP požadavek, což má za následek zpomalení načítání stránek. *Sass* i *Less* staví na původním CSS *@import*, ale namísto vyžadování nového HTTP požadavku *preprocesor* použije soubor, který má být naimportovaný a kombinuje ho se souborem, do kterého má být naimportován. To znamená, že ve webovém prohlížeči bude obsloužen pouze jeden CSS soubor. [42]

4.8.2 Buildování CSS

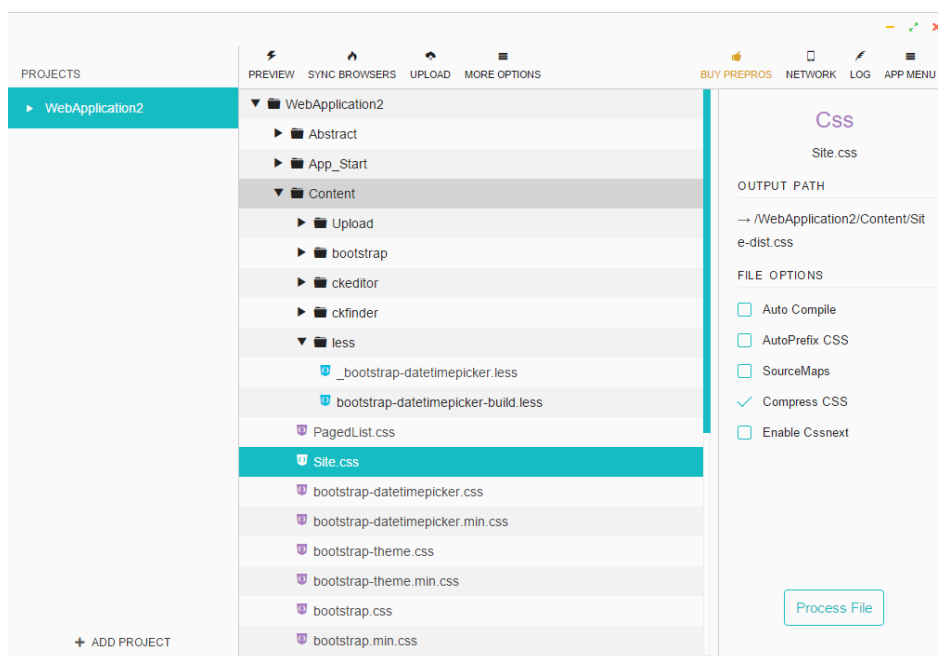
Při vytváření webových stránek využívá různých operací na zdrojovými kódy. Kompilace CSS z preprocesoru, minifikace CSS nebo optimalizace velikosti obrázků. Zároveň je potřeba vytvářet distribuční verze; k tomu všemu slouží sestavovací neboli buildovací nástroje. [42]

Sestavovací nástroje pro webový frontend tedy tu část webu, která je přístupná běžnému návštěvníkovi, jsou dvou typů:

- **Zjednodušené** – jednodušší k užití, ale s omezeným rozsahem funkcí. Takovým nástrojem je například *Prepros* a *CodeKit*.
- **Plnohodnotné** – s velkým rozsahem funkcí, ale zároveň složitějším pro začátečníky a neprogramátory. Takovým nástrojem je například *Grunt* a *Gulp*.

Prepros

Prepros je zjednodušený multiplatformní buildovací nástroj ke kompilování *Less*, *Sass*, *Stylus*⁵ a dalších preprocesorů CSS (Obr. 30). Jeho ovládání nevyžaduje žádné programování a poskytuje mimo podpory úkolů nad CSS, JavaScript a také FTP aktualizaci webových stránek.



Obrázek 30: Rozhraní nástroje Prepros

⁵ Stylus je jedním z preprocesorů. Více informací na <http://stylus-lang.com/>.

Nejedná se o nástroj zdarma, ale je pravděpodobně nejvhodnějším buildovacím nástrojem pro (ne)programátory, kteří nemají dostatek zkušeností a času pro učení se s plnohodnotnými nástroji, jako jsou *Grunt* nebo *Gulp*. [57]

Grunt

Nejpoužívanějším plnohodnotným buildovacím nástrojem je *Grunt*. Jedná se o opensource nástroj pod licencí *MIT* sloužící při vývoji k automatizaci jednoduchých a opakujících se úkolů.

Grunt obsahuje především velké množství pluginů, které mu umožňují provádět různé operace. Například v odvětví úpravy kaskádových stylů umožňuje spojování CSS, zpracování preprocesorového kódu do CSS, minifikaci CSS a JavaScriptových souborů. Jeho možnosti jsou téměř neomezené.

Celý nástroj *Grunt* se konfiguruje pomocí konfiguračního nástroje *Gruntfile.js*, který je možné také verzovat, tím pádem sdílet s ostatními vývojáři v týmu pro zachování integrity konfigurace. *Grunt* je přehlednější pro práci s CSS a začátečníky. [58]

Hlavními nedostatky nástroje *Grunt* jsou: [59]

- Obsáhlé konfigurace u jednotlivých pluginů.
- Celkově velmi dlouhé konfigurační soubory.
- Pomalé zpracovávání častých I/O operací při větším počtu souborů.
- Psaní JavaScript kódu nahrazeno konfigurací.
- Složitější konkurenční zpracování úloh pomocí pluginů.

GULP

Gulp je streamovací buildovací nástroj založený na JavaScriptu. Často bývá využíván pro streamování klientských souborů skrze řadu procesů ve chvíli, kdy je aktivován nějaký nastavený spouštěč. *Gulp* je také opensource nástroj pod licencí *MIT*, který poskytuje stejné možnosti jako *Grunt*, ale jeho konfigurace se provádí pomocí JavaScriptového kódu. Z tohoto důvodu je vhodný především pro programátory a velké projekty. [60]

Hlavními nedostatky nástroje *Gulp* jsou: [59]

- Nutnost znalosti používání streamů.
- Složitě řešené lineární závislosti jednotlivých úloh.
- Pluginy lze konfigurovat pouze minimálně.
- Oproti *Grunt* menší rozsah pluginů.

4.8.3 Testování funkčnosti CSS

Při vytváření webových stránek a především CSS kódu je nezbytné vždy otestovat jejich funkčnost a renderování na různých prohlížečích. Moderní desktopové prohlížeče nabízí v dnešní době ve většině případů stejné výsledky, ale jejich počet stále stoupá, a to především mobilní webové prohlížeče.

Provést test na všech typech zařízení ve všech webových prohlížečích není z důvodu jejich vysokého počtu možné, a proto lze využít následujících možností:

- Použití vývojářského desktopového prohlížeče, jako je například *Chrome* s rozšířením *DevTools*. Toto rozšíření nabízí emulátor různých mobilních zařízení, vizualizaci nadefinovaných *media queries* nebo také emulátor síťového přenosu pro otestování výkonu stránek.
- Další možností je použití specializovaných programů, emulátorů a simulátorů určených pro tyto testy. Tyto nástroje nabízejí vysoký počet zařízení, které lze simulovat a to včetně jejich operačních systémů a různých webových prohlížečů. To umožňuje se co nejvíce přiblížit funkčnosti skutečného zařízení.
- Posledním krokem je testování na skutečných přístrojích různých značek, operačních systémů a velikostí, ale to je časově i finančně velmi náročné.

V neoptimálnějších případech by se měly provést všechny tyto kroky. [42]

Browserstack

Jedním ze specializovaných nástrojů pro testování všech typů webových prohlížečů je *Browserstack*. Jedná se o online službu, která na svém serveru provo-

zuje přes tisíc různých desktopových i mobilních prohlížečů a pomocí vzdáleného připojení na nich lze otestovat funkčnost vytvořených stránek.

Jedná se o poměrně drahý placený nástroj, z toho důvodu je vhodný pro větší projekty, u kterých je správné zobrazení stránky na všech zařízeních nutností. [61]

Browsersync

Dalším nástroje, který je šikovným pomocníkem při vytváření a testování webového designu, je *Browsersync*. Jedná se o opensource *Node.js* komponentu pro lokální vývoj webů, jež umožňuje kompatibilitu s preprocesorovými nástroji jako například *Grunt* nebo *Gulp*.

Hlavní funkcí nástroje *Browsersync* je živé promítání změn provedených ve zdrojovém kódu přímo do prohlížeče. Díky tomu lze pozorovat provedené změny v CSS nebo HTML bez nutnosti stále obnovovat webovou stránku.

Browsersync využívá technologii *Weinre*, která je na podobném principu jako *DevTools* v prohlížeči *Chrome*. Díky *Weinre* je tak možné upravovat CSS, DOM a JavaScript přímo v konzoli prohlížeče, a to i napříč platformami desktopovými i mobilními. 0

YSlow!

Nástrojem sloužícím k otestování celkové rychlosti stránek je *YSlow!*. Jedná se o rozšíření, které je dostupné pro většinu nejznámějších webových prohlížečů. Nástroj analyzuje webové stránky a známkuje rychlost jednotlivých komponent podle 24 pravidel, které jsou vytvořeny společností *Yahoo!*.

Použití je velmi snadné, stačí nainstalovat rozšíření do webového prohlížeče. Následně spustit *YSlow!* a navštívit webovou stránku, kterou je cílem otestovat. Následně se zobrazí jednotlivé komponenty a jejich hodnocení s podrobným vysvětlením. Tímto způsobem lze předejít problémům, které by mohly zpomalovat chod webových stránek. [62]

5 Programy určené k tvorbě frontend⁶

K tvorbě frontend lze nyní použít velmi velký počet aplikací. Na trhu se jich nachází až nepřehledné množství, a proto je poměrně obtížné vybrat aplikaci, která bude nejvhodnější.

Tyto aplikace se dělí do čtyř hlavních skupin. První z nich jsou **aplikace určené k psaní kódu** s omezenými možnostmi vytváření grafického vzhledu. Druhou skupinou jsou naopak **aplikace určené primárně pro tvorbu webové grafiky**, často formou takzvaného „WYSIWYG“ neboli „*Co vidíš je to, co dostaneš*“. V těchto aplikacích často není nutné umět programovat a vše stačí pouze naklikat pomocí přehledného rozhraní. První a druhá skupina se většinou vzájemně doplňuje. Třetí skupinou jsou **kompletní nástroje pro vývoj webových aplikací**, které poskytují možnost vytváření kódu společně s vizuální úpravou webového designu. Součástí těchto programů je často možnost připojení databáze, okamžitá publikace na server a podpora vysokého počtu pluginů, rozšiřujících program o další možnosti. Čtvrtou skupinou jsou **systemy pro správu obsahu** neboli *CMS*. Ty nabízejí možnost využít již vytvořených šablon, do kterých se pouze doplní obsah stránek. Pro jeho správu tak nejsou potřeba téměř žádné technické znalosti. Tvůrci se v nich mohou zaměřit pouze na vytváření obsahu stránek, proto je CMS také označován jako redakční nebo publikační systém.

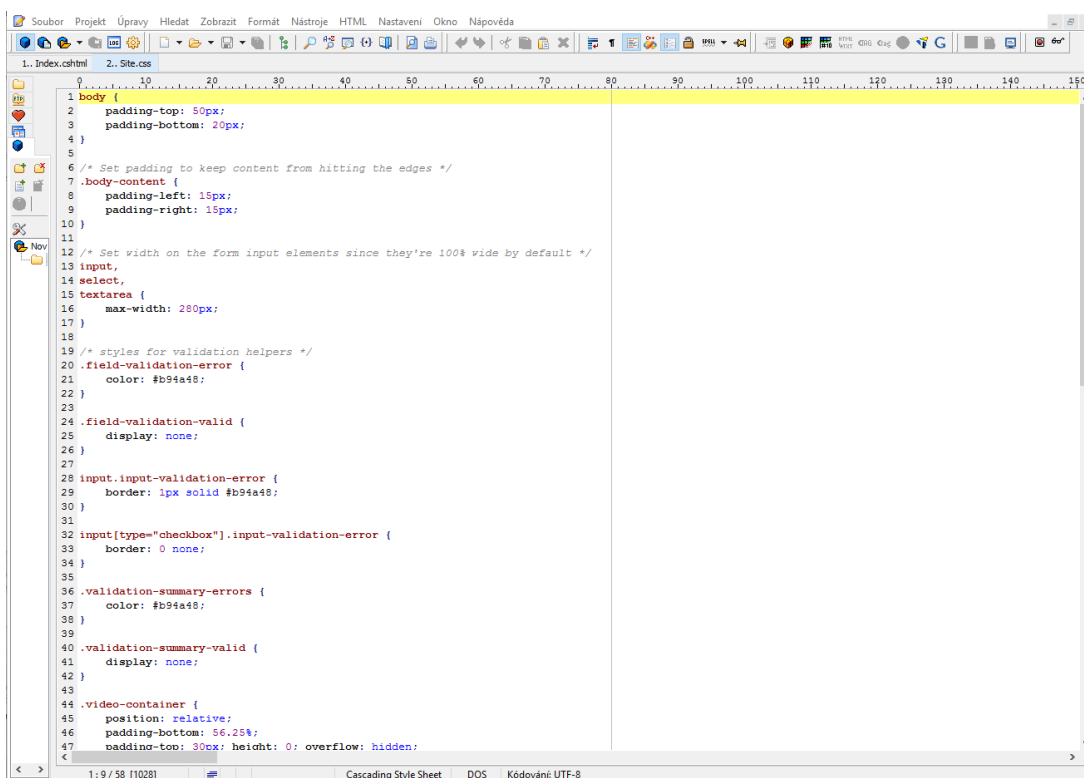
Dalším dělením aplikací pro úpravu designu je jejich cena. Při vytváření webových stránek pro soukromé účely si tvůrce často vystačí s aplikacemi, které jsou volně ke stažení na internetu. Nejčastěji se jedná o aplikace určené buď pro programování, nebo pro tvorbu webového designu. Problém nastává v případě tvorby pro komerční účely, kdy často tyto aplikace nelze použít z licenčních důvodů nebo pro jejich nedostatečné možnosti. V takovém případě je nutné zakoupit placenou aplikaci, která splňuje požadované vlastnosti. Tyto aplikace bývají prodávány v různých výhodných balíčcích společně s dalšími aplikacemi, které lze při tvorbě webových stránek využít a které se vzájemně doplňují.

⁶ Pojem frontend se využívá v oblasti vývoje webových aplikací k označení části webu viditelné návštěvníkům.

5.1 Aplikace pro psaní kódu

PSPad

PSPad je výkonný textový editor pro operační systémy Windows s mnoha nástroji pro tvorbu webových stránek s freeware licencí (Obr. 31). Program nabízí jednoduché prostředí, na dnešní dobu v již poněkud zastaralém vzhledu, ale to mu neubírá na funkčnosti, jelikož je určen především pro psaní kódu. *PSPad* nabízí šablony pro většinu nejpoužívanějších programátorských jazyků, integrovaného FTP klienta pro správu obsahu na webovém serveru, možnost úpravy CSS souborů a zobrazit jejich výstup nebo také tvorbu maker. Program je možné doplnit o velký počet rozšíření, které vytváří buď sám autor programu, nebo jakýkoliv jiný programátor s autorovým souhlasem.



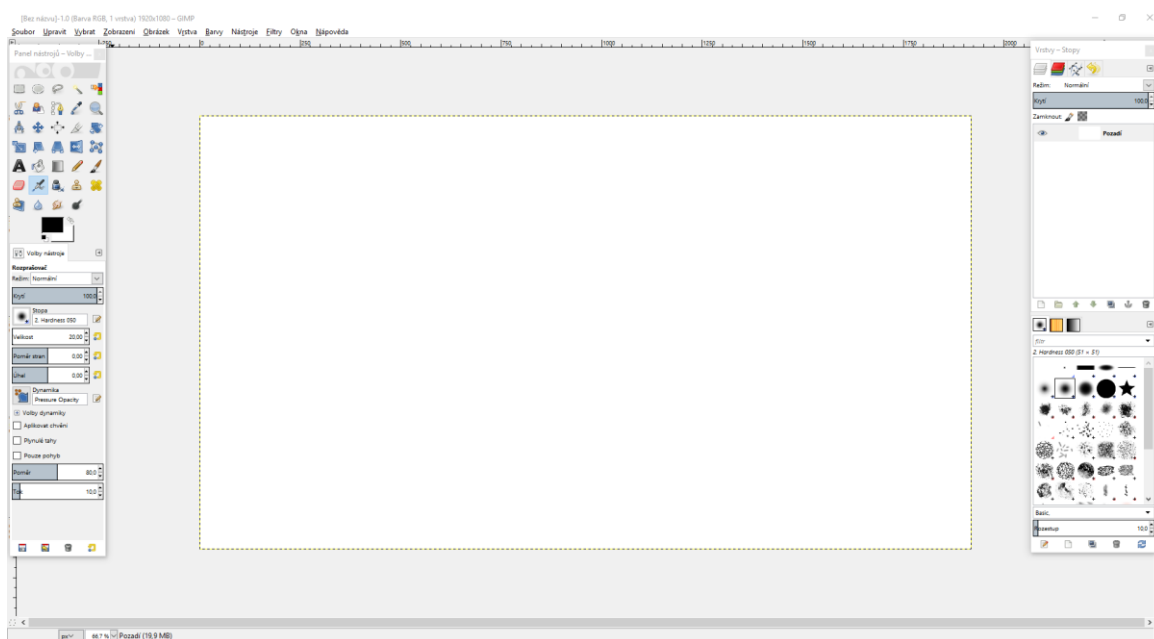
Obrázek 31: Rozhraní nástroje PSPad

Aplikace není určena primárně pro grafiky k tvorbě webového designu, ne nabízí žádné „WYSIWYG“ rozhraní, ale je šikovným pomocníkem při tvorbě webových stránek především pro programátory. [63]

5.2 Nástroje k tvorbě grafických prvků a webového designu

GIMP

GIMP je bezpochyby nejvýkonnějším volně dostupným programem pro úpravu fotek a jiných grafických prvků (Obr. 32). Jedná se o multiplatformní nástroj dostupný pro všechny desktopové platformy. Velmi často bývá srovnáván s programem *Adobe Photoshop®*, který nabízí více možností, ale právě *GIMP* je jeho nejlepší alternativou zdarma.



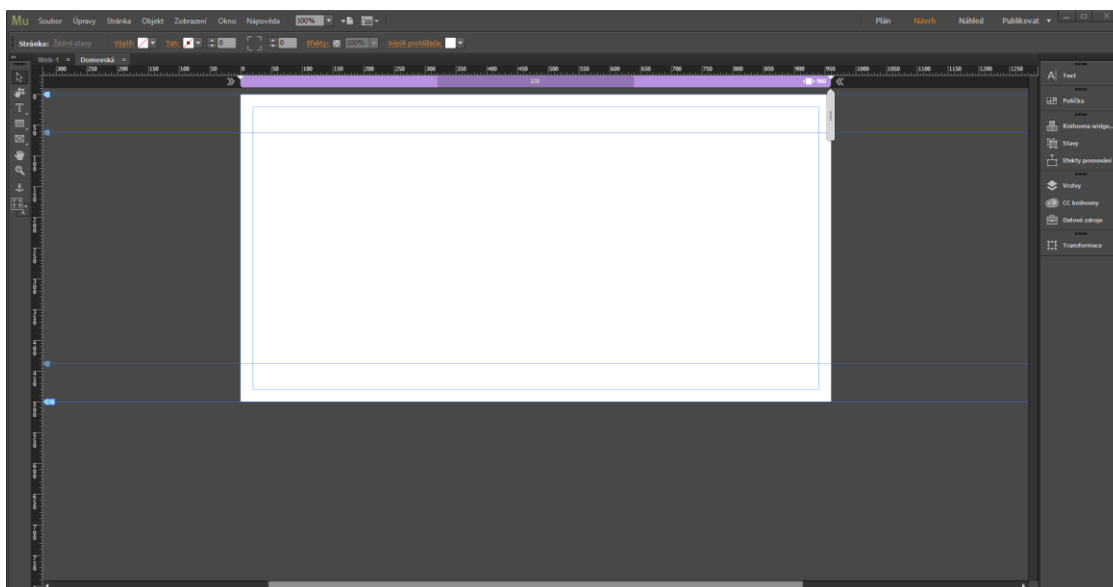
Obrázek 32: Prostředí nástroje GIMP

Program nabízí mnoho možností. Může být použit jako jednoduchý kreslicí program, k profesionálnímu retušování fotografií, hromadnému renderování obrázků nebo převodu formátů. V oblasti webové grafiky je nejčastěji využíván k tvorbě ikon, grafických designových prvků a celkové tvorbě uživatelského rozhraní.

Podobně jako *PSPad* lze i *GIMP* rozšířit vysokým počtem volně stažitelných rozšíření, díky kterým s tímto programem lze pracovat na plnohodnotné profesionální úrovni. [64]

Adobe Muse®

Aplikace *Adobe Muse®* nabízí možnost vytváření designu a publikování plnohodnotných responzivních HTML5 stránek bez nutnosti psaní jakéhokoliv kódu



Obrázek 33: Rozhraní nástroje Adobe Muse® (Obr. 33).

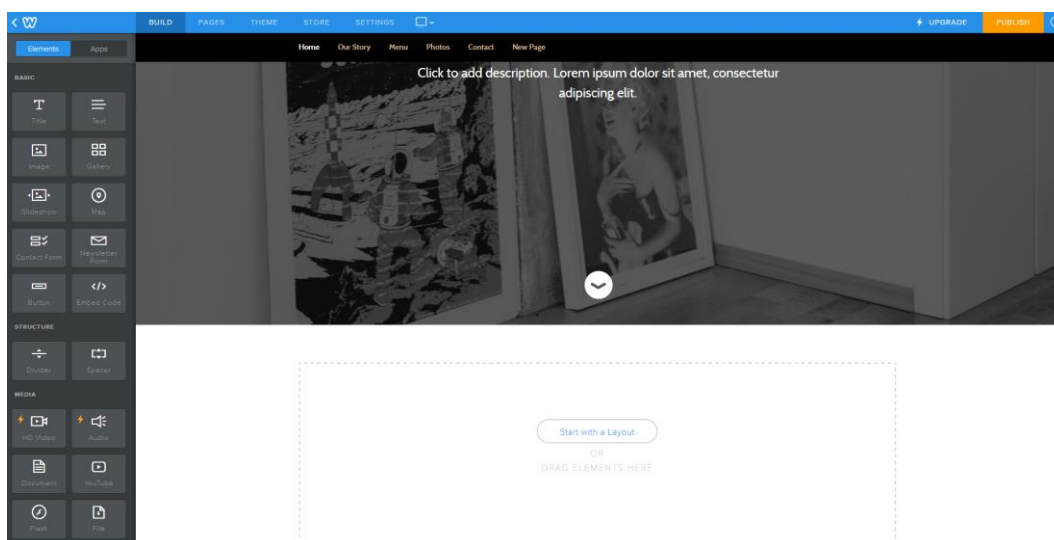
Aplikace *Adobe Muse®* využívá přístupu volné formy, kdy je tvorba responzivního webu omezena pouze tvůrčovou představivostí. Program nabízí podobné rozhraní, které využívají i ostatní *Adobe* aplikace, jako jsou *Photoshop®*, *Illustrator®* a *InDesign®*. Pomocí toho je možné vytvářet moderní webový design a vizuální efekty jako automatické prezentace obrázků, pohyby elementů při posouvání na stránce, interaktivní kontaktní formuláře nebo animované přechody bez nutnosti znalosti programování.

Přístup volné formy znamená, že tvorba není omezena vazbami a omezeními webových stránek. Aplikace umožňuje vytváření webu jednoduchým přetahováním elementů pomocí kurzoru, užíváním widgetů a formulářů ke konfiguraci rozhraní a jeho funkčnosti.

Adobe Muse® je vhodný pro rychlé vytváření designu a publikování vzhledově krásných webových stránek jakéhokoliv typu. Jakmile je design dokončen, tak aplikace vytvoří stránky a připraví je k publikování. [65]

Weebly

Weebly je jednoduchý grafický editor založený na principu „táhni a pusť“, který je vhodný především pro začátečníky (Obr. 34). Aplikaci není nutné instalovat, vše lze provádět pouze ve webovém prohlížeči po přihlášení do aplikace.



Obrázek 34: Rozhraní nástroje Weebly

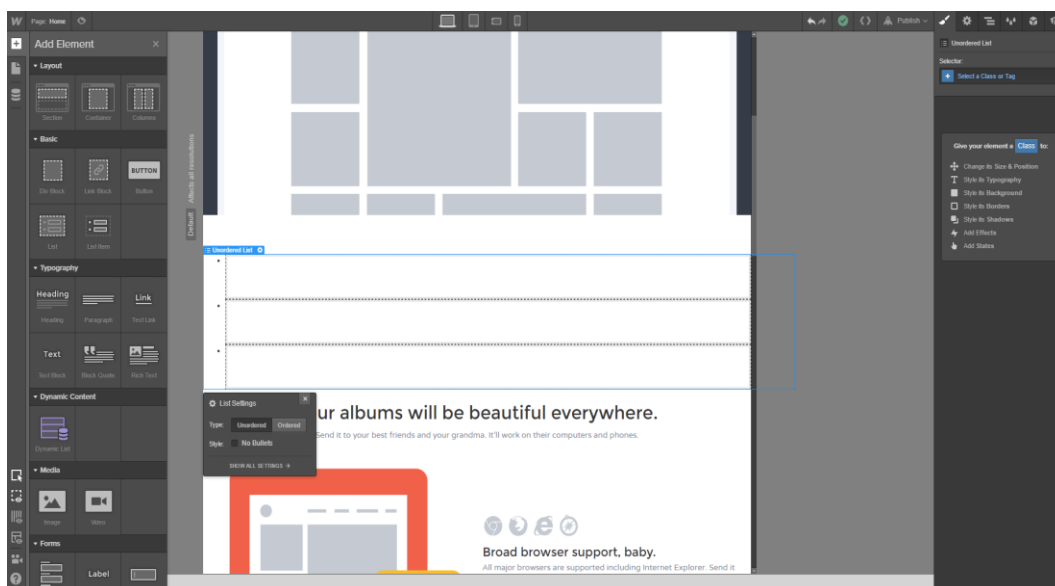
K vytvoření webových stránek není nutné znát nic o programování. Na začátku lze vybrat již vytvořenou šablonu, která se následně pomocí přehledného menu snadno upraví. Úprava se provádí přetahováním jednotlivých prvků z nabídky do stránky a doplněním požadovaného textu. Všechny vytvořené stránky poskytují responzivní design, pomocí kterého jsou dobře čitelná na všech typech zařízení.

Aplikace zároveň nabízí svůj vlastní hosting pro okamžité publikování na internet. V případě použití bezplatné licence je použita subdoména „weebly.com“, ale stránky lze tímto způsobem publikovat během několika sekund. Jedná se o nejrychlejší řešení vytvoření funkčních a dobře vypadajících webových stránek. [66]

Webflow

Další alternativou k aplikaci *Weebly* je aplikace *Webflow* (Obr. 35). Opět se jedná o online aplikaci ve webovém prohlížeči, která nabízí téměř stejné možnosti. Rozdíl je v možnostech úprav, které lze provádět. V případě *Webflow* je možné jednotlivé prvky velmi detailně upravovat, a to i přímo v CSS kódu. Bohužel vy-

generování kompletního CSS i HTML kódu pro celé webové stránky je možné pouze v placené verzi.



Obrázek 35: Rozhraní nástroje Webflow

Aplikace také nabízí vlastní hosting v neplacené verzi opět s vlastní subdoménou „weblow.io“. Oproti *Weebly* má verze zdarma další omezení, a to limitaci webu pouze na dvě stránky a omezenou technickou podporu.

Webflow aplikace je určena pro již zkušenější tvůrce webových stránek než v případě *Weebly*. Nabízí více možností, ale pro její plnohodnotné využití se pravděpodobně nelze vyhnout alespoň minimálním platbám. [67]

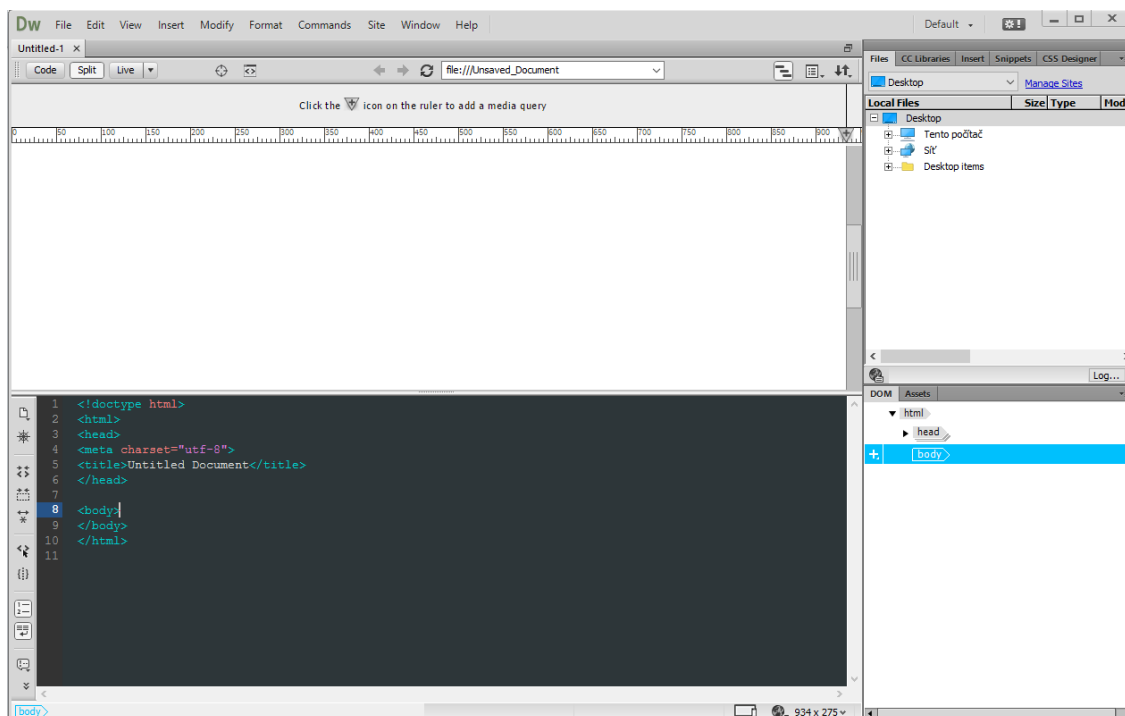
5.3 Kompletní nástroj pro vývoj webových aplikací

Adobe Dreamweaver®

Firma *Adobe Systém* nabízí profesionální aplikaci pro tvorbu webových stránek – **Adobe Dreamweaver®**. Aplikace umožňuje vytvářet responzivní web, který mohou uživatelé používat na stolních počítačích, tabletech nebo mobilních telefonech. Zároveň nabízí vytváření designu od úplného začátku až po publikování na internet v jediném programu.

Adobe Dreamweaver® je multiplatformní nástroj pro vývoj webových stránek (Obr. 36). Program byl vyvinut v roce 1997 společností *Macromedia*. [68] Ta jej spravovala do roku 2005, kdy byl získán společností *Adobe Systém*, která ho vlastní dodnes. [69]

Jedná se o nejkompexnější nástroj pro webové designery a vývojáře na světě. Aplikace *Adobe Dreamweaver®* nabízí jak rozhraní pro vizuální design, tak úpravu kódu; je určen designérům, kteří rozumí HTML i CSS. Při designování



Obrázek 36: Rozhraní aplikace *Adobe Dreamweaver®*

webových stránek v *Adobe Dreamweaver®* je struktura stránek definovaná pomocí HTML, zatímco rozložení a styling je definován v CSS.

Adobe Dreamweaver® využívá populární framework *Bootstrap* pro responzivní design a nabízí velkou škálu nástrojů, jako jsou fragmenty kódu, kontrola chyb za běhu programu nebo automatické doplňování slov, jež umožňují snadnější programování. Ke kódu je možné přistupovat kdykoliv během vytváření designu a zároveň jsou tyto dvě části propojeny. To znamená, že při úpravě kódu jsou změny okamžitě vidět v oblasti vizuálního designu a naopak - při úpravě designu se přizpůsobuje kód.

Ke zjednodušení práce napomáhá také úzké propojení *Adobe Dreamweaver®* s ostatními grafickými nástroji *Adobe*, které jsou součástí takzvaného „*Creative Cloud*“ balíku. Díky němu je například možnost pracovat s rozpracovaným *Adobe Photoshop®* projektem přímo v *Adobe Dreamweaver®*.

Nová verze aplikace také nabízí integrovaný CSS editor a možnost propojení mobilního zařízení pro okamžité otestování kompatibility přímo na daném zařízení. Stejně jako dříve zmíněné aplikace i *Adobe Dreamweaver*® nabízí možnost automatické publikace na webové servery.

Adobe Dreamweaver® je vhodný pro publikování rozsáhlejších webových stránek a pro tvorbu v profesionálních designerských týmech. [70]

5.4 Redakční systémy CMS

WordPress

WordPress, zdarma dostupný webový software založený na jazyku PHP. *WordPress* má velmi početnou komunitu vývojářů, díky kterým jsou k dispozici tisíce šablon a rozšíření. Díky nim je možné vytvářet blogy, webové stránky nebo aplikace přesně podle vlastních představ. Zároveň to ovšem může způsobit zbytečně pokročilé možnosti nastavení pro uživatele požadující pouze základní webové stránky. Nevýhodou řešení *WordPress* je také absence oficiální podpory, kdy se uživatelé musejí spolehnout pouze na uživatelská fóra. [71]

Joomla!

Joomla! je podobně jako *WordPress* zdarma dostupný webový software založený na jazyku PHP. Jedná se o velmi oblíbené řešení nabízející tisíce rozšíření a širokou uživatelskou komunitu. Jeho využití je jak pro klasické webové stránky a aplikace, tak také pro veřejně nepřístupné řešení využívané například uvnitř firem. Oproti jiným *CMS* nabízí menší výběr již vytvořených šablon a složitější vytváření stránek. [72]

Drupal

Dalším zdarma dostupným *CMS* nástrojem založeným na jazyku PHP je *Drupal*. Nabízí téměř totožné možnosti jako *WordPress* a *Joomla!*, ale jeho předností je silná komunita, která dokonce pořádá osobní setkání a to i na území České republiky. Díky tomu má široké zastoupení po celém světě. Uživatel má v tomto řešení také možnost si spravovat vlastní server. Slabou stránkou je omezené množství kvalitních šablon a složitější vytváření webových stránek. [73]

Webnode

Řešením od českých tvůrců je *Webnode*, který nabízí širokou nabídku moderních šablon. Ty jsou tvořeny ve stylu vizuálně dominantních stavebnic pro jejich snadnou úpravu. *Webnode* je zaměřen především na osobní a firemní webové stránky s responzivním designem. Nedostatkem *Webnode* je slabší komunita uživatelů a vývojářů, ale to je nahrazeno kvalitní dokumentací a oficiální podporou. [74]

6 Příprava obsahu

Praktickou částí této diplomové práce je webový portál o standardech webové grafiky a základních principech při jejím vytváření. Hlavním článkem tohoto portálu je především jeho obsahová forma, kterou tvoří tři základní prvky:

- **Články,**
- **testovací otázky,**
- **video návody.**

Před samotným vypracováním jednotlivých prvků je nutné prostudování aktuálních trendů v oblasti webové grafiky a její základní problematiky. Po dostatečném nastudování následuje výběr témat, které by v portálu měly být zmíněny. Při tomto úkolu je nejvhodnější vycházet z vlastních zkušeností, které určí, zda je dané téma vhodné zmínit ve vytvářeném portálu.

Prvním zpracovaným prvkem webového portálu jsou **články**, vytvořené po předchozím výběru z jednotlivých témat teoretické části této práce a následně seřazené do kategorií. Všechny informace jsou zpracovány z více zdrojů z důvodu ověření jejich správnosti a přesnosti. Obsah jednotlivých článků je doplněn o obrázky, praktické ukázky a citované zdroje.

Druhým prvkem doplňujícím vytvořené články jsou **testovací otázky**. Jedná se o výběr základních otázek k tématu, které bylo popsáno v daném článku. Tyto testy jsou především pro ověření uživatelských nově získaných vědomostí. Testy vždy uživateli poskytnou ihned po dokončení testu zpětnou vazbu, díky které má možnost zjistit, kde se dopustil chyby. Volba jednotlivých otázek je určena dle důležitosti daného problému a jeho obtížnosti, jelikož zopakování si termínu pomocí testu napomůže k jeho zapamatování.

Poslední obsahovou částí webového portálu jsou **video návody**. Ty jsou vytvořeny především k ukázce práce s aplikací *Adobe Dreamweaver®*. Tato aplikace je vybrána, protože se jedná o komplexní aplikaci na vytváření webových stránek i jejího designu. Proto lze na tomto jediném nástroji názorně demonstrovat tvorbu jak HTML schématu, kaskádových stylů, tak i vytvoření jednoduché responzivní

webové stránky. Video návod je vytvořen také pro ukázkou práce s nástrojem *Browsersync*, kde je názorně demonstrováno jeho použití.

K jednotlivým video návodům je nejprve nutné sepsat textové podklady, které poslouží pro mluvený komentář.

Popsaným způsobem je vytvořen textový podklad pro jednotlivá témata, který je následně namluven pomocí mikrofonu a podle potřeby sestřihán v editačním programu *Adobe Premiere®*. Pomocí namluveného podkladu je vytvořena ukázkou práce s aplikací *Adobe Dreamweaver®*, která je snímána pomocí aplikace *VidShot Capturer*⁷. Výsledkem je video záznam, jenž musí být v aplikaci *Adobe Premiere®* sestřihán a doplněn o namluvený komentář. Nakonec je celek vyexportován do video souboru, nahrán cloudové uložení a vložen do webového portálu.

⁷ VidShot Capturer je aplikací určenou k snímání obrazovky s freeware licencí

7 Vlastní návrh portálu

K prezentování informací o webovém designu slouží webový portál. Jedná se o nejčastější a především nejlépe přístupnou formu prezentování informací. Webový portál má hlavně tu přednost, že může být kdykoliv doplňován a aktualizován, díky čemu si stále zachovává svou aktuálnost v prezentovaném tématu. Další výhodou je již zmíněná přístupnost, kdy jej lze využít kdykoliv pomocí počítače nebo mobilního telefonu s internetovým připojením.

7.1 Tvorba webového portálu

Webový portál s názvem je vytvořen pomocí aplikace *Microsoft Visual Studio 2013* s použitím následujících technologií: *ASP.NET MVC*, *ASP.NET Identity*, *C#*, *JavaScript*, *HTML*, *CSS3*, *Bootstrap Framework* a rozšíření *CKeditor*.

Další možností k vytvoření webového portálu mohlo být použití systému pro správu obsahu neboli *CMS*. Ten nebyl zvolen z důvodu své jednoduchosti, která nabízí pouze omezené možnosti úprav a neumožňuje názorně demonstrovat problémy tvořící obsahovou část celého portálu.

Základem portálu je opensource framework *ASP.NET MVC 5* určený pro tvorbu dynamických webových aplikací. Pomocí softwarové architektury *MVC*, tedy zkratkou slov *Model* (model), *View* (pohled) a *Controller* (řadič), je aplikace rozdělena do třech komponent na datový model aplikace, uživatelské rozhraní a řídicí logiku. Toto rozdělení umožňuje snadnou architektonickou úpravu, otestování i průběžnou správu webového portálu. Z tohoto důvodu je tato architektura použita pro webový portál.

Projekt vychází ze šablony aplikace *Visual Studio 2013*, která využívá *Bootstrap* (Obr. 37). Jedná s nejrozšířenější framework pro vytváření responzivního designu webových stránek. *Bootstrap* je tvořen mnoha částmi především ze série *LESS* stylopisů, jež umožňuje vybrat pouze potřebné komponenty pro daný projekt. Hlavním výstupem jsou kompilované CSS a JavaScript soubory. Hlavním důvodem výběru tohoto frameworku je jeho schopnost zpracování responzivního designu a široká podpora ze strany mnoha vývojářů.

WebGraphic

Vítejte na portálu WebGraphic věnujícímu se základům teorie a tvorby webové grafiky.

Portál je cílen studentům Univerzity Hradec Králové studujícím Fakultu informatiky a managementu. Na oborech se zaměřením na informační technologie se vyučují předměty Technologie pro publikování na webu I a II, ale z omezených časových možností na nich není studentům možné dostatečně prezentovat problematiku tvorby webového designu. Právě to je hlavním podmětem k vytvoření portálu WebGraphic, který nabídne všem studentům základní znalosti o tvorbě webového designu, které budou moct kdykoliv uplatnit

Dělení do kategorií

Obsah portálu je tvořen především **články**, které jsou z důvodu přehlednosti rozděleny do několika **kategorií**.

Testy

Každý **přihlášený** uživatel má možnost otestovat své nově nabyté schopnosti pomocí krátkých testů v jednotlivých **kategoriích**. Pokud stále nemá vytvořený účet, tak se stačí **zaregistrovat**. To lze pomocí přihlašovacího jména a hesla nebo také pomocí účtu Google+ a Facebook.

Videotutoriály

Ukázky práce s grafickými nástroji jsou doplněny o video návody, které jsou přímo obsahem **článků**.

Obrázek 37: Úvodní stránka portálu WebGraphic

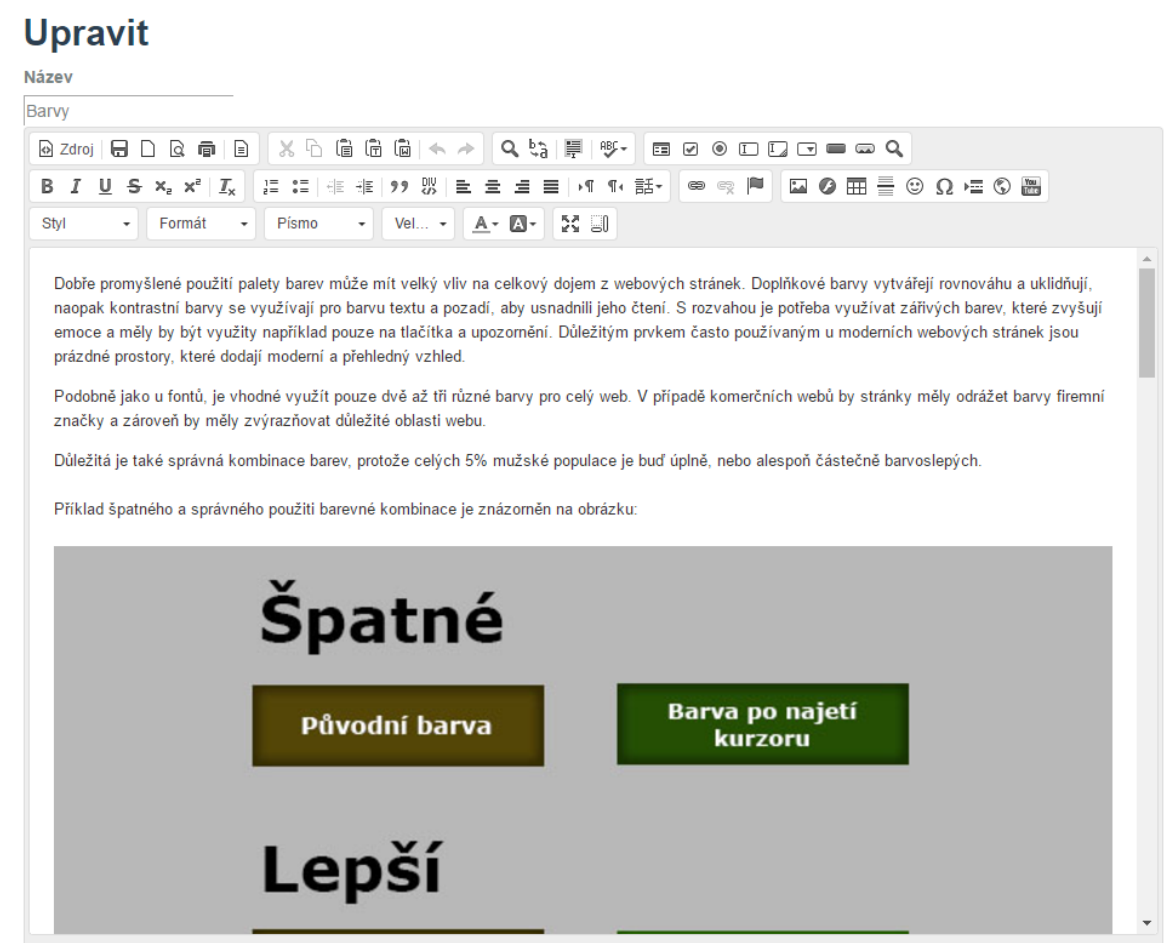
Z důvodu zabezpečení webového portálu a správy uživatelských rolí je použit systém *ASP.NET Identity*. Systém umožňuje uživatelskou autentizaci pomocí registrace, kde zadá uživatelské jméno a heslo nebo použitím již existujícího účtu ze sociální sítě *Google+* a *Facebook*.

Pro přihlášení účty sociálních sítí se využívá autorizační framework *Oauth 2.0*, který je již součástí *ASP.NET Identity*. K jeho použití jsou nakonfigurovány poskytovatelé sociálních sítí *Google+* a *Facebook*. Jelikož se jedná o dvě nejrozšířenější sítě v České republice, tak je webový portál využívá k možnosti přihlášení.

Oauth 2.0 framework pracuje na principu poskytování omezeného přístupu aplikaci třetích stran ke službě HTTP na požadavek. Přihlašovací údaje pro *ASP.NET Identity* jsou ukládány do databáze, jež umožňuje jejich správu a případný import a export při migraci projektu. Z důvodu ochrany uživatelských údajů jsou uživatelská hesla ukládána ve formátu hash.

Webový portál nabízí tři základní role. První z nich je „*Administrátor*“, která má možnost spravovat jednotlivé účty všech uživatelů a přiřazovat jim role. Druhou rolí je „*Moderátor*“, který má možnost vytvářet a upravovat obsah webového portálu. Jeho úkolem je především vytváření jednotlivých kategorií a článků. Třetí rolí je uživatel, který se přišel na Portál vzdělávat a má možnost číst články, skládat testy a prohlížet video návody.

Pro snadné vytváření článků je do portálu integrované rozšíření *CKEditor* (Obr. 38). Jedná se o opensource aplikaci, která poskytuje snadnou úpravu HTML textu v „WYSIWYG“ editoru. Pomocí editoru lze např. upravovat styl a velikost písma jako v běžných textových editorech. *CKEditor* má možnost integrace velkého počtu freeware rozšíření, které jsou v portálu využity především pro vkládání obrázků, videí nahraných na serveru „YouTube“ anebo testů k jednotlivým článkům pomocí HTML prvku „Iframe“.



Obrázek 38: Ukázka rozhraní rozšíření *CKEditor* na portálu *WebGraphic*

7.2 Hlavní prvky portálu

Webový portál je tvořen třemi hlavními prvky:

- **Tematickými články,**
- **video návody,**
- **krátkými testy.**

Všechny tyto prvky jsou vloženy formou jednotlivých článků. Pro jejich základní rozdělení jsou video návody označeny koncovým slovem „(video)“ a krátké testy slovem „(test)“. Z důvodu přehlednosti jsou všechny články rozděleny do několika kategorií. Seznam hlavních kategorií:

- Základy webové grafiky;
- Základy CSS;
- Responzivní CSS;
- Moderní technologie pro práci s CSS;
- Grafické nástroje;
- Dreamweaver – video návody.

Kategorie „*Základy webové grafiky*“ obsahuje články vysvětlující obecné principy, kterými se je nutné řídit při vytváření webové stránky. Tato kategorie je zařazena do portálu jelikož každý tvůrce webových stránek by se s nimi měl seznámit před samotným začátkem své vlastní tvorby.

Další kategorií jsou „*Základy CSS*“, která je určena především pro uživatele, kteří minimální zkušenosti s kaskádovými styly. Nejprve se uživatelé seznámí s možnostmi aplikování stylů a následně se základními prvky, jež je nutné znát.

Třetí kategorie „*Responzivní design*“ je určena uživatelům se základními znalostmi kaskádových stylů. Ve vybraných článcích se dozví, jaké jsou základní prvky responzivního designu a jaké je jejich praktické použití v kódu. Závěrečný článek této kategorie je věnován frameworkům určeným k tvorbě responzivního designu.

Čtvrtou kategorií jsou „*Moderní technologie pro práci s CSS*“. Hlavními tématy obsažených článků jsou moderní metody a nástroje pro komplexnější práci s kaskádovými styly.

V páté kategorii „*Nástroje pro webovou grafiku*“ se nachází přehled a popis konkrétních nástrojů pro základní vytváření webového designu. Představeny jsou jednoduché volně dostupné a profesionální programy běžně užívané tvůrci webových stránek.

Poslední šestá kategorie „*Dreamweaver – video návody*“ obsahuje speciální články tvořené formou video návodů. Ty se zabývají problematikou vytváření designu webových stránek v nástroji *Adobe Dreamweaver*.

7.3 Vytváření video návodů

Jedním z hlavních prvků webového portálu jsou video návody. Ty jsou použity z důvodu názorného zobrazení a vysvětlení problematiky tvorby webového designu v profesionálním nástroji. Tímto nástrojem je *Adobe Dreamweaver*, který umožňuje jak samotnou tvorbu HTML s podporou většiny frameworků, tak především poskytuje možnost úpravy webového designu stránek. Jelikož se jedná o profesionální nástroj, tak nabízí obsáhle a poměrně složité ovládání, především pro začínající tvůrce. Právě pro ně jsou vytvořené video návody určeny.

Video návody jsou tvořeny v těchto hlavních krocích:

- Zpracování textových podkladů.
- Namluvení doprovodného komentáře.
- Tvorba video návodů prací v aplikaci snímáním obrazovky.
- Sestřihání a vyrenderování výsledného video návodu.
- Nahrání video návodu na internet a vložení do webového portálu.

7.3.1 Zpracování textových podkladů

Zpracování textu pro video návody probíhá v následujícím postupu:

1. Sepsání seznamu cílů, které budou ukázány.
2. Otestování dosažení cílů v aplikaci.
3. Úprava seznamu cílů, podle poznatku z 2. Bodu.
4. Otestování upraveného seznamu cílů.
5. Sepsání podkladového textu během práce s aplikací.

Ukázka podkladového textu, sloužícího jako scénář k video návodu:

Před začátkem práce s aplikací *Adobe Dreamweaver®* se je nejprve nutné seznámit se základními prvky rozhraní.

Při prvním otevření programu se zobrazí uvítací okno. Toto okno se zobrazí pouze v případě, že není v programu otevřena žádná rozpracovaná práce. V levé části okna se zobrazují všechny projekty a položky, které byly otevřeny při přechodích spuštěních programu pro snadný přístup k rozpracovanému projektu. Prostřední oblast umožňuje vytvoření jakékoliv nové položky. Jestliže není vytvořena žádná položka, tak jsou všechny okolní panely zašedlé a nelze s nimi pracovat stejně tak, jako s většinou nabídek v hlavní nabídce. Po kliknutí na některou z položek seznamu se automaticky aktivují všechny panely, většina možností z hlavní nabídky a uvítací obrazovka zmizí.

Hlavní nabídka se nachází v horní oblasti obrazovky a na rozdíl od ostatních prvků je vždy viditelná. Tato nabídka umožňuje tradiční přístup ke všem jednotlivým možnostem programu.

Na pravé straně obrazovky se nachází několik typů pracovních panelů. Všechny panely se mohou libovolně přesouvat, schovat nebo naopak zobrazit jiné z hlavní nabídky. Pomocí těchto panelů lze například pohodlně upravovat design CSS, vkládat HTML prvky nebo zobrazit strukturu webových stránek.

Další důležitou oblastí je panel *Properties* neboli vlastnosti, který je v základní pracovní ploše nutné zobrazit. V tomto panelu se zobrazují vlastnosti právě vybraného prvku na stránce a zároveň lze jednotlivé vlastnosti upravovat.

Hlavní oblast umístěna uprostřed obrazovky se nazývá okno dokumentu. V této oblasti je zobrazen hlavní obsah stránek. V horní části této oblasti si lze zvolit, zda bude zobrazen pouze kód, pouze vizuální design anebo obojí pomocí možnosti *Split*. Aplikace má v sobě zabudovaný také builder pro debugování přímo ve zvoleném prohlížeči.

7.3.2 Namluvení doprovodného komentáře

Dalším krokem je namluvení textového podkladu z předchozího kroku. Nahraní hlasu proběhlo pomocí diktafonu v mobilním telefonu.

Pro tento způsob je rozhodnuto po otestování dvou dalších zařízení určených k záznamu zvuku, které jsou v rámci finančních možností k dispozici. Prvním testovaným zařízením je mikrofon integrovaný v notebooku. Zvuk není bohužel dostatečně čistý z důvodu rušení aktivním chlazením notebooku. Druhým testova-

ným zařízením je mikrofon umístěný v herních sluchátkách. V tomto případě zvuk dosahuje vyšších kvalit než v notebooku, ale hlavními problémy jsou zvukové otřesy způsobené pohybem hlavy a slabá výsledná hlasitost nahraného záznamu.

Posledním a zvoleným řešením je nahrávání hlasu pomocí osobního mobilního telefonu. Hlavními důvody pro zvolení mobilního telefonu jsou nejvyšší čistota zvuku v porovnání s ostatními testovanými zařízeními a možnost pořízení nahrávky v místnosti s nejlepší akustikou díky jeho malým rozměrům.

Záznam je pořízen pomocí integrované aplikace „*Diktafon*“ přímo od výrobce telefonu. Následný záznam je sestříhán a upraven v aplikaci *Adobe Premiere® CS6*. Vyexportovaný upravený záznam má formát MP3 s přenosovou rychlostí 256 kbps, která je dostačující pro poslech mluveného slova.

7.3.3 Tvorba video návodů prací v aplikaci snímáním obrazovky

Ve třetím kroku se vytváří pomocí **snímání obrazovky** samotný **video návod pro práci v nástroji Adobe Dreamweaver®**. Příprava pro tento krok probíhá již při vytváření textových podkladů, během kterého jsou jednotlivé kroky video návodu otestovány a připraveny.

Ke **snímání obrazovky** je využíván program *VidShot Capturer*, který umožňuje nahrávat pouze vybranou oblast obrazovky a poskytuje možnost nahrávání s 25 snímků za vteřinu, díky čemuž je výsledný záznam při přehrávání plynulý bez sekání.

Samotná práce v nástroji *Adobe Dreamweaver®* probíhá s puštěným namluveným komentářem, vytvořeným v předchozím kroku. Díky tomu je zachováno potřebné tempo práce v nástroji. Po dokončení všech kroků je záznam ze snímání obrazovky uložen a připraven pro další zpracování.

7.3.4 Sestříhání a vyrenderování výsledného video návodu

Ve čtvrtém kroku je **vytvořený záznam ze snímání obrazovky sestříhán**, doplněn o úvodní titulky a spojen se zvukovým záznamem namluveného doprovodného komentáře. Nakonec je celý projekt **vyrenderován do výsledného formátu „MP4“** v rozlišení „1920x1080“ pixelů s přenosovou rychlostí 9955 kbps pro video záznam a 188 kbps pro zvukový záznam. Přenosová rychlost je zvolena tak, aby byl zachován optimální poměr mezi vysokou kvalitou video záznamu

a nízkou výslednou velikostí souboru. Všechny tyto kroky jsou opět provedeny v nástroji *Adobe Premiere® CS6*.

7.3.5 Nahrání video návodu na internet a vložení do webového portálu

V posledním kroku je vyrenderovaný **video návod nahrán na server** YouTube. Tento server je zvolen z důvodu neomezeného limitu pro nahrávaná videa v jakékoliv kvalitě. Zároveň lze u nahraného videa zvolit možnost viditelnosti videa, která jsou v tomto případě nahrána jako soukromá s možností volby, kdo a kde se může na daná videa podívat. Nakonec jsou vytvořené **video návody vloženy** ve formě HTML prvku „`iframe`“⁸ **do webového portálu** jako články.

7.4 Testy znalostí

Dalším prvkem webového portálu jsou krátké testy znalostí. Tyto testy jsou vytvořeny pro většinu tematických článků na webovém portálu. Jejich cílem je dodat uživateli možnost zopakovat si právě získané znalosti z článku, ke kterému se daný test váže.

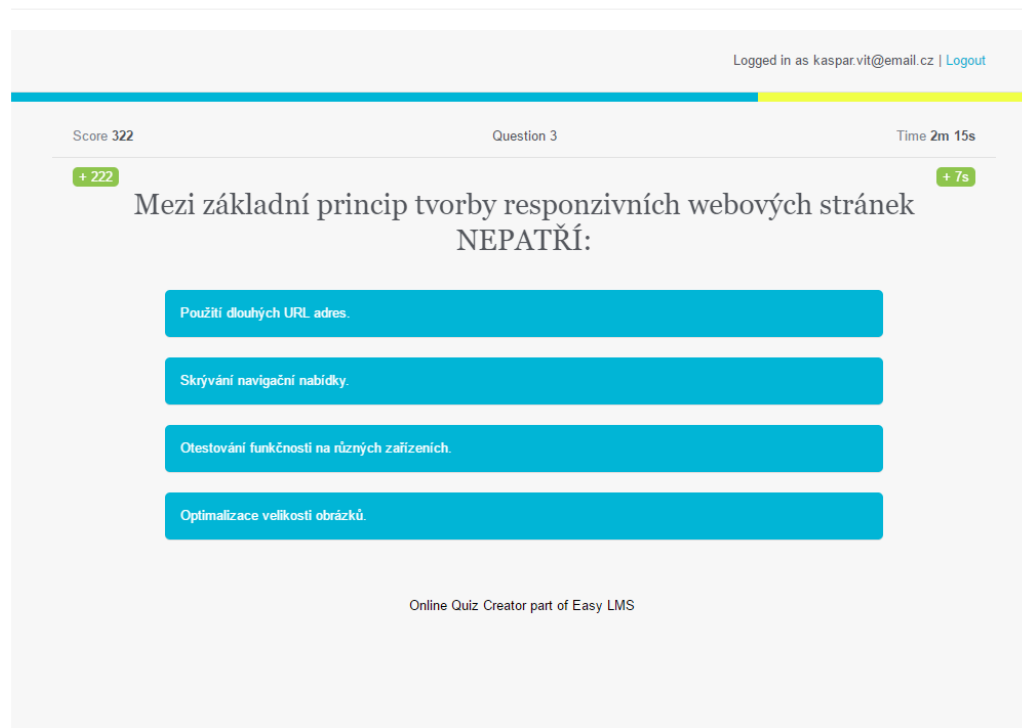
Jednotlivé testy jsou tvořeny formou samostatných článků, které mají vždy stejný název jako článek, ke kterému se váží, ale na konci názvu jsou doplněny o slovo „(test)“.

Podkladem pro všechny testy jsou právě vybrané články, proto uživatel vždy bude testován pouze z toho, co je obsaženo v článku. Pro vytvoření zpětné vazby test uživateli okamžitě po zadání špatné odpovědi oznámí chybu a zobrazí správnou odpověď.

Testy mají vždy formu jedné položené otázky a čtyř možných odpovědí, z nichž pouze jedna je správná (Obr. 39).

⁸ `Iframe` je tag umožňující vkládat takzvané vnořené stránky. Více informací na <https://www.jakpsatweb.cz/iframe.html>

Responzivní design (test)



Obrázek 39: Ukázka testovací otázky

Po hledání nejvhodnější formy testu byla vybrána možnost vložení testu z webové aplikace „www.onlinequizcreator.com“. Tato aplikace umožňuje vytvoření osobního účtu pro vytváření a správu testů. Tvůrce, nejčastěji moderátor, má tak možnost pohodlně vytvářet a upravovat testy k daným tématům.

Tato aplikace byla vybrána z důvodu volné licence užití, která sice omezuje počet možných otázek v jednom testu na 15, ten je však pro potřeby portálu vyhovující. Dalším důvodem je možnost vložení vytvořených testů do aplikace formou „*iframe*“ prvku. Díky tomu lze vytvořené testy pohodlně vkládat v CKEditoru do webového portálu.

7.5 Přístupové údaje na webový portál

Projekt WebGraphic je dostupný na internetové adrese:

<http://webgraphic.azurewebsites.net/>

Přihlašovací údaje na webový portál jsou následující:

Administrátorský účet:

Přihlašovací jméno: *admin@3xampLe.com*

Heslo: *admin@3xampLe.com*

Moderátorský účet:

Přihlašovací jméno: *moderator@3xampLe.com*

Heslo: *moderator@3xampLe.com*

Uživatelský účet:

Přihlašovací jméno: *user@3xampLe.com*

Heslo: *user@3xampLe.com*

Administrátorský účet může především spravovat účty a role. Moderátorský účet je určen především pro tvorbu a úpravu článků i kategorií. Uživatelský účet má právo pouze prohlížet jednotlivé články.

8 Ověření

Webový portál byl ověřen ve třech fázích. V první fázi bylo ověřeno zobrazení jednotlivých prvků portálu a jeho funkčnost v různých webových prohlížečích. Celkem bylo ověření provedeno na čtyřech nejrozšířenějších webových prohlížečích. První z nich byl *Google Chrome*, druhý *Mozilla Firefox*, třetí *Microsoft Edge* a poslední *Microsoft Explorer*. Všechny prohlížeče měly poslední dostupné aktualizace k datu testování. Během testu se neprojevyly žádné problémy se zobrazením portálu ani jeho funkčností.

Ve druhé fázi byla ověřena správná funkčnost portálu na různých typech zařízení. Prvním testovaným zařízením byl stolní počítač. Na tomto zařízení byl projekt primárně vyvíjen, a proto také nebyl shledán žádný problém v zobrazení ani ve funkčnosti. Dalším testovacím zařízením byl tablet s operačním systémem *Android* a integrovaným webovým prohlížečem, na kterém se objevily chyby v zobrazení hlavní nabídky, ale funkčnost portálu byla zachována. Posledním testovaným zařízením byl mobilní telefon s operačním systémem *Android*. Na mobilním telefonu se projevil stejný zobrazovací problém jako na tabletu, ale funkčnost byla zachována.

Před třetí fází byly opraveny nalezené chyby v zobrazení u tabletů a mobilních zařízení a teprve poté začala třetí fáze ověření. Ta byla provedena dvěma osobami, které provedly jak ověření zobrazení a funkčnosti webového portálu na různých typech zařízení, tak obsahovou formu webového portálu. K ověření byl použit mobilní telefon i tablet s operačním systémem *iOS* a nebyly nalezeny žádné problémy v zobrazení ani funkčnosti. Obě osoby mají povědomí o problematice informačních technologií, ale pouze základní nebo žádné zkušenosti s tvorbou webového designu. Proto byly vhodnými kandidáty pro ověření kvality obsahové části webového portálu.

Výtky testovaných osob byly pouze k typografickým chybám v textech a příliš dlouhých člancích, ale obě osoby se shodly, že díky webovému portálu získaly nové znalosti, které jim v problematice webové grafiky chyběly. Zároveň ocenily možnost okamžitého otestování popsané problematiky. Zmíněné chyby byly opraveny a dlouhé články byly rozděleny do více článků.

9 Shrnutí výsledků

Po vytvoření webového portálu, jeho ověření a následné úpravě je nutné shrnout získané poznatky. Nejprve k dosažení stanoveného cíle, kterým bylo zpracování principů webové grafiky a vytvoření vzdělávacího portálu o tvorbě webové grafiky pro studenty začínající s tvorbou webových stránek. Pomocí ověření dvěma osobami, které mají znalosti začínajících tvůrců webových stránek je zřejmé, že práce měla smysl.

Dosažení stanoveného cíle samozřejmě neznamená, že by vše bylo dokonalé. Webový portál je vhodné rozšířit o další okruhy webové grafiky s novými technologiemi. Toto rozšíření je bohužel nutné provádět inkrementálně v průběhu času. Zveřejněné články a postupy měly reflektovat potřeby studentů, na které je tento portál cílen. Pro jeho další rozšíření proto bude nejprve nutná zpětná vazba od většího počtu studentů.

Video návodům je třeba dodat vyšší kvalitu zvuku doprovodného komentáře, ale to bohužel není možné při použití levného nahrávacího zařízení. Samotný hlas komentující dění na obrazovce je také třeba zlepšit. To ovšem vyžaduje jiného, kvalitnějšího řečníka, který vše namluví. Video návody také mohou obsahovat úpravu s více grafickými prvky včetně doprovodného textu. K této úpravě nemohlo dojít z důvodu uplynutí třicetidenní zkušební lhůty, během které byla verze programu *Adobe Premiere®* zdarma.

Během tvorby webového portálu se připravená metodika tvorby práce osvědčila jako správná. Problémy nastaly pouze při hledání zdrojů, které jsou převážně v elektronické podobě a v anglickém jazyce. Naštěstí se knižní zdroje podařilo vyhledat hlavně díky vyhledávači *Google Books* a doporučení vedoucího práce. Další problémy, uváděné v „*Ověřování*“, byly odhaleny právě během v uvedené fázi a následně vyřešeny.

10 Závěr a doporučení

Během této diplomové práce byl vytvořený webový portál, dostupný na adrese <http://webgraphic.azurewebsites.net/>, s rozsáhlou obsahovou formou o tvorbě webové grafiky. Jeho zpracování i obsah bude jistě dostačující pro všechny, kteří začínají s tvorbou webových stránek a nevědí, jak správně zahájit svou práci.

Tato práce se může stát webovým portálem pro studenty, studující předměty s tematikou tvorby webových stránek, kde naleznou potřebné informace, které nebylo možné z dostupných studijních materiálů získat. Zároveň může být tento portál postupně rozšiřován o aktuální informace a trendy v oblasti webové grafiky. Jeho integrování do studijních materiálů předmětů *Technologie pro publikování na webu I. a II.* by jistě bylo přínosné pro všechny studenty, kteří nejsou seznámeni s touto problematikou.

Budoucí vývoj tohoto webového portálu je velmi pravděpodobný, jelikož vzdělávací portály podobného typu jsou stále častěji využívány k získávání nových znalostí v oboru informačních technologií. Portál by se měl v budoucnu více zaměřit na interakci s uživatelem, která napomůže k udržení pozornosti a snadnějšího pochopení vysvětlovaného problému. Takovéto interakce by mohlo být dosaženo rozsáhlejšími testy doplněnými o grafické prvky nebo možností vkládat komentáře k jednotlivým tématům pro získání zpětné vazby.

Závěrem je vhodné říct, že tvorba nejen praktické, ale i teoretické části přinesla mnoho nových zkušeností jak s programováním ve frameworku *ASP.NET*, tak s vytvářením vzdělávacích modulů. V budoucnu se tyto poznatky a zkušenosti jistě ještě nejednou uplatní.

11 Literární a internetové zdroje

- [1] FRIEDMAN, Vitaly. 10 Principles Of Effective Web Design. *Smashing Magazine* [online]. 2008 [cit. 2016-08-14]. Dostupné z: <https://www.smashingmagazine.com/2008/01/10-principles-of-effective-web-design/#how-do-users-think>
- [2] NIELSEN, Jakob. Scrolling and Attention. Nielsen Norman Group [online]. 2010 [cit. 2016-08-14]. Dostupné z: <https://www.nngroup.com/articles/scrolling-and-attention/>
- [3] CSS Web Safe Font Combinations. W3schools.com [online]. c1999-2016 [cit. 2016-08-14]. Dostupné z: http://www.w3schools.com/cssref/css_websafe_fonts.asp
- [4] CSS3 @font-face Rule. W3schools.com [online]. c1999-2016 [cit. 2016-08-14]. Dostupné z: http://www.w3schools.com/cssref/css3_pr_font-face_rule.asp
- [5] Google Fonts. Google Fonts [online]. c2016 [cit. 2016-08-14]. Dostupné z: <https://www.google.com/fonts>
- [6] BIGMAN, Alex. Why all designers need to understand color blindness. 99designs [online]. 2013 [cit. 2016-08-14]. Dostupné z: <https://99designs.com/blog/tips/designers-need-to-understand-color-blindness/>
- [7] STANÍČEK, Petr. Paletton - The Color Scheme Designer. Paletton.com [online]. c2002 [cit. 2016-08-25]. Dostupné z: <http://paletton.com/>
- [8] JONES, Henry. The Principle of Contrast in Web Design. Web Design Ledger [online]. 2010 [cit. 2016-08-14]. Dostupné z: <http://webdesignledger.com/the-principle-of-contrast-in-web-design/>
- [9] STAŇKOVÁ, Zuzana, Leona FOUSKOVÁ, Jana BOHÁČKOVÁ a Tomáš HAVRÁNEK. Průvodce bytového designéra. EDIKA, 2013, 216 s. ISBN 978-80-266-0360-3. Kniha
- [10] COUSINS, Carrie. Color and cultural design considerations. Webdesigner Depot [online]. 2012 [cit. 2016-08-14]. Dostupné z: <http://www.webdesignerdepot.com/2012/06/color-and-cultural-design-considerations/>

- [11] Basic color schemes: Introduction to Color Theory. Tiger Color [online]. c2010-2016 [cit. 2016-08-14]. Dostupné z: <http://www.tigercolor.com/color-lab/color-theory/color-theory-intro.htm>
- [12] Color Harmonies: Basic techniques for combining colors. Tiger Color [online]. c2010-2015 [cit. 2016-08-14]. Dostupné z: <http://www.tigercolor.com/color-lab/color-theory/color-harmonies.htm>
- [13] Interesting stats. HTTP Archive [online]. 2016 [cit. 2016-08-14]. Dostupné z: <http://httparchive.org/interesting.php#bytesperpage>
- [14] NIELSEN, Jakob. Banner Blindness: Old and New Findings. Nielsen Norman Group [online]. 2007 [cit. 2016-08-14]. Dostupné z: <https://www.nngroup.com/articles/banner-blindness-old-and-new-findings/>
- [15] DIHIANSAN, Jamie. Behind the scenes: A/B testing part 3: Finalé. Signal v. Noise [online]. 2011 [cit. 2016-08-14]. Dostupné z: <https://signalnoise.com/posts/2991-behind-the-scenes-ab-testing-part-3-final>
- [16] IDLER, Sabina. 10 Things To Know About Images In Web Design. Usabilla Blog [online]. 2012 [cit. 2016-08-14]. Dostupné z: <http://blog.usabilla.com/10-things-to-know-about-images-in-web-design/>
- [17] VAUGHN, Jason. File Types and Optimization. CreativePublic.com [online]. [cit. 2016-08-15]. Dostupné z: http://www.creativepublic.com/file_types_optimization.php
- [18] MACEK, David. JPEG Resampler. David Macek Development [online]. c2016 [cit. 2016-08-25]. Dostupné z: <http://software.macek.cc/resampler.php>
- [19] Photoshop CC zdarma.: Editor fotografií, zkušební verze. Adobe [online]. c2016 [cit. 2016-08-25]. Dostupné z: <http://www.adobe.com/cz/products/photoshop.html>
- [20] TIŠNOVSKÝ, Pavel. JPEG - král rastrových grafických formátů? Root.cz [online]. 2006 [cit. 2016-08-14]. Dostupné z: <http://www.root.cz/clanky/jpeg-kral-rastrovych-graficky-ch-formatu>
- [21] TIŠNOVSKÝ, Pavel. PNG is Not GIF. Root.cz [online]. 2006 [cit. 2016-08-14]. Dostupné z: <http://www.root.cz/clanky/png-is-not-gif/>

- [22] MILLER, Eric. Five Rules of Effective Website Navigation. About.com [online]. c2016 [cit. 2016-08-14]. Dostupné z: http://graphicdesign.about.com/od/effectivewebsites/a/web_navigation.htm
- [23] FRIEDMAN, Vitaly. Ultimate Guide To Grid-Based Web Design: Techniques and Tools. Noupe [online]. 2009 [cit. 2016-08-14]. Dostupné z: <http://www.noupe.com/design/ultimate-guide-to-grid-based-web-design.html>
- [24] BIGMAN, Alex. 6 principles of visual hierarchy for designers. 99designs [online]. 2014 [cit. 2016-08-14]. Dostupné z: <https://99designs.com/blog/tips/6-principles-of-visual-hierarchy/>
- [25] NIELSEN, Jakob. F-Shaped Pattern For Reading Web Content. Nielsen Norman Group [online]. 2006 [cit. 2016-08-14]. Dostupné z: <https://www.nngroup.com/articles/f-shaped-pattern-reading-web-content/>
- [26] TILP, Dominik. Responzivní a adaptivní design. Agile ICT [online]. 2013 [cit. 2016-08-25]. Dostupné z: <http://www.agile-ict.com/cs/blog/responzivni-a-adaptivni-design>
- [27] Mobile Phone Emulator. Mobile phone emulator [online]. c2016 [cit. 2016-08-25]. Dostupné z: <http://www.mobilephoneemulator.com/>
- [28] Responsive Web Design. Google Developers [online]. 2015 [cit. 2016-08-25]. Dostupné z: <https://developers.google.com/webmasters/mobile-sites/mobile-seo/responsive-design>
- [29] SEXTON, Patrick. Javascript & mobile seo. Varvy [online]. 2016 [cit. 2016-08-25]. Dostupné z: <https://varvy.com/mobile/javascript-seo.html>
- [30] What can you do with JavaScript. W3 [online]. 2011 [cit. 2016-08-14]. Dostupné z: https://www.w3.org/community/webed/wiki/What_can_you_do_with_JavaScript
- [31] HTML & CSS. W3 [online]. c2016 [cit. 2016-08-14]. Dostupné z: <https://www.w3.org/standards/webdesign/htmlcss#whatcss>
- [32] Applying CSS. HTML Dog [online]. c2003-2016 [cit. 2016-08-14]. Dostupné z: <http://www.htmldog.com/guides/css/beginner/applyingcss/>

- [33] CSS Units. W3schools.com [online]. c1999-2016 [cit. 2016-08-14]. Dostupné z: http://www.w3schools.com/cssref/css_units.asp
- [34] CSS Legal Color Values. W3schools.com [online]. c1999-2016 [cit. 2016-08-14]. Dostupné z: http://www.w3schools.com/cssref/css_colors_legal.asp
- [35] CSS/Properties. W3 [online]. 2012 [cit. 2016-08-14]. Dostupné z: <https://www.w3.org/community/webed/wiki/CSS/Properties>
- [36] CSS/Properties/margin. W3 [online]. 2011 [cit. 2016-08-14]. Dostupné z: <https://www.w3.org/community/webed/wiki/CSS/Properties/margin>
- [37] CSS/Properties/padding. W3 [online]. 2011 [cit. 2016-08-14]. Dostupné z: <https://www.w3.org/community/webed/wiki/CSS/Properties/padding>
- [38] CSS/Properties/border. W3 [online]. 2011 [cit. 2016-08-14]. Dostupné z: <https://www.w3.org/community/webed/wiki/CSS/Properties/border>
- [39] LEPAGE, Pete. Set the viewport. Google Developers [online]. 2016 [cit. 2016-08-14]. Dostupné z: <https://developers.google.com/web/fundamentals/design-and-ui/responsive/fundamentals/set-the-viewport?hl=en>
- [40] NIMESH, Rakhitha. Fluid Grid Layout Tutorial For Responsive Web Design Creation. 1stwebdesigner [online]. 2016 [cit. 2016-08-25]. Dostupné z: <http://1stwebdesigner.com/fluid-grid-layout/>
- [41] COUSINS, Carrie. Tips & Tools for Grid-based Layouts. Codrops [online]. 2012 [cit. 2016-08-25]. Dostupné z: <http://tympanus.net/codrops/2012/07/10/tips-tools-for-grid-based-layouts/>
- [42] MICHÁLEK, Martin. Vzhůru dolů [online]. 2016 [cit. 2016-08-14]. Dostupné z: <http://www.vzhurudolu.cz/ebook>
- [43] BRINGHURST, Robert. The Elements of Typographic Style. Hartley & Marks, 2013, 398 s. ISBN 978-0-88179-212-6.
- [44] HOLST, Christian. Readability: the Optimal Line Length. Baymard Institute [online]. 2010 [cit. 2016-08-14]. Dostupné z: <http://baymard.com/blog/line-length-readability>
- [45] RUDER, Emil. Typographie: A Manual of Design: A Textbook of Design. 4. vydání. Verlag Niggli, 2009, 274 s. ISBN 978-3-7212-0043-0.

- [46] CSS3 @media Rule. W3schools.com [online]. c1999-2016 [cit. 2016-08-14]. Dostupné z: http://www.w3schools.com/cssref/css3_pr_mediaquery.asp
- [47] W3.CSS Tutorial. W3schools.com [online]. c1999-2016 [cit. 2016-08-14]. Dostupné z: <http://www.w3schools.com/w3css/default.asp>
- [48] CSS: Global CSS settings, fundamental HTML elements styled and enhanced with extensible classes, and an advanced grid system. Bootstrap [online]. c2016 [cit. 2016-08-14]. Dostupné z: <http://getbootstrap.com/css/>
- [49] GALLAGHER, Nicolas a Jonathan NEAL. A modern, HTML5-ready alternative to CSS resets. Normalize.css [online]. c2016 [cit. 2016-08-14]. Dostupné z: <http://necolas.github.io/normalize.css/>
- [50] Responsive CSS Boilerplate: A dead simple, responsive boilerplate. Skeleton [online]. c2016 [cit. 2016-08-14]. Dostupné z: <http://getskeleton.com/>
- [51] Sass makes CSS fun again. GitHub [online]. c2016 [cit. 2016-08-14]. Dostupné z: <https://github.com/sass/sass>
- [52] Getting Started: An overview of Less, how to download and use, examples and more. Less [online]. c2016 [cit. 2016-08-14]. Dostupné z: <http://lesscss.org/>
- [53] HIXON, Jeremy. An Introduction To LESS, And Comparison To Sass. Smashing Magazine [online]. 2011 [cit. 2016-08-14]. Dostupné z: <https://www.smashingmagazine.com/2011/09/an-introduction-to-less-and-comparison-to-sass/>
- [54] BITTNER, Jan. 2. díl - Úvod do CSS preprocesoru Sass. It Network [online]. c2016 [cit. 2016-08-14]. Dostupné z: <http://www.itnetwork.cz/html-css/webove-portfolio/tutorial-moderni-webove-portfolio-sass/>
- [55] CROOM, Johnathan. Sass vs. LESS vs. Stylus: Preprocessor Shootout. Envato Tuts+ [online]. 2012 [cit. 2016-08-14]. Dostupné z: <http://code.tutsplus.com/tutorials/sass-vs-less-vs-stylus-preprocessor-shootout--net-24320>
- [56] Sass Basics. Sass [online]. c2006-2015 [cit. 2016-08-14]. Dostupné z: <http://sass-lang.com/guide>

- [57] Prepros : Compile Less, Sass, CoffeeScript and much more with live browser refreshing. Prepos [online]. 2016 [cit. 2016-08-15]. Dostupné z: <https://prepros.io/>
- [58] The JavaScript Task Runner. Grunt [online]. c2016 [cit. 2016-08-14]. Dostupné z: <http://gruntjs.com/>
- [59] OŽANA, Roman. Gulp vs. Grunt: souboj bez vítěze a poraženého. Zdroják [online]. 2014 [cit. 2016-08-14]. Dostupné z: <https://www.zdrojak.cz/clanky/gulp-vs-grunt-souboj-bez-viteze-a-porazeneho/>
- [60] REITAN, Erik, Scott ADDIE a Daniel ROTH. Using Gulp. ASP.NET [online]. c2016 [cit. 2016-08-14]. Dostupné z: <https://docs.asp.net/en/latest/client-side/using-gulp.html>
- [61] Live, Web-Based Browser Testing. BrowserStack [online]. c2011-2016 [cit. 2016-08-14]. Dostupné z: <https://www.browserstack.com/>
- [62] DURAN, Marcel. YSlow FAQ. YSlow [online]. c2016 [cit. 2016-08-25]. Dostupné z: http://yslow.org/faq/#faq_work
- [63] Textový editor PSPad. PSPad [online]. c2016 [cit. 2016-08-14]. Dostupné z: <http://www.pspad.com/cz/>
- [64] The Free & Open Source Image Editor. GIMP [online]. c2001-2016 [cit. 2016-08-14]. Dostupné z: <https://www.gimp.org/>
- [65] Seznámení s aplikací Adobe Muse, získání nápovědy a podpory. Adobe Muse CC [online]. c2015 [cit. 2016-08-14]. Dostupné z: <https://helpx.adobe.com/cz/muse.html>
- [66] Create a Unique Website and Sell Your Products Online. Weebly [online]. c2016 [cit. 2016-08-14]. Dostupné z: <https://www.weebly.com/features>
- [67] Design and develop at the same time. Webflow [online]. c2016 [cit. 2016-08-14]. Dostupné z: <https://webflow.com/designer>
- [68] History of Macromedia, Inc. FundingUniverse [online]. 2003 [cit. 2016-08-14]. Dostupné z: <http://www.fundinguniverse.com/company-histories/macromedia-inc-history/>
- [69] CAMPBELL, Holly. Adobe Completes Acquisition of Macromedia. Adobe Systems Incorporated [online]. 2005 [cit. 2016-08-14]. Dostupné z:

<https://www.adobe.com/aboutadobe/pressroom/pressreleases/pdfs/200512/120505AdobeAcquiresMacromedia.pdf>

- [70] Adobe Dreamweaver CC. Adobe [online]. c2016 [cit. 2016-08-14]. Dostupné z: <http://www.adobe.com/cz/products/dreamweaver.html>
- [71] Blog Tool, Publishing Platform, and CMS. WordPress.org [online]. c2016 [cit. 2016-08-25]. Dostupné z: <https://wordpress.org/>
- [72] About Joomla!. Joomla! [online]. c2016 [cit. 2016-08-25]. Dostupné z: <https://www.joomla.org/about-joomla.html>
- [73] SUCHÝ, Jakub. O systému Drupal. Dumpal.cz [online]. 2012 [cit. 2016-08-25]. Dostupné z: <https://www.drupal.cz/o-systemu-drupal>
- [74] Výhody tvorby webových stránek s Webnode. Webnode [online]. 2016 [cit. 2016-08-25]. Dostupné z: <https://www.webnode.cz/vyhody-systemu-webnode/>

12 Seznam obrázků

Obrázek 1: Výsledek studie sledování očí uživatelů (Zdroj: https://media.nngroup.com/media/editor/alertbox/eyetracking-scrolling-long.jpg)	5
Obrázek 2: Prostředí webové aplikace Google Fonts (Zdroj: https://www.google.com/fonts) ...	8
Obrázek 3: Ukázka použití správných barev (Zdroj: Vlastní tvorba)	9
Obrázek 4: Barevné kolo (Zdroj: http://joos.in/wp-content/uploads/2012/01/Real-Color-Wheel-Clean7x7-72dpi.png).....	11
Obrázek 5: Kontrastní schéma (Zdroj: Vlastní tvorba)	11
Obrázek 6: Dělené kontrastní schéma (Zdroj: Vlastní tvorba)	12
Obrázek 7: Analogické schéma (Zdroj: Vlastní tvorba)	12
Obrázek 8: Triádní schéma (Zdroj: Vlastní tvorba).....	12
Obrázek 9: Tetradické schéma (Zdroj: Vlastní tvorba).....	13
Obrázek 10: Čtvercové schéma (Zdroj: Vlastní tvorba).....	13
Obrázek 11: Poměr komprese a faktoru kvality (Zdroj: http://i.iinfo.cz/urs/jpeg1_c-preview-116541807821679.png)	16
Obrázek 12: Ukázka rozbalovací nabídky (Zdroj: https://dribbble.com/).....	19
Obrázek 13: Navigace mobilní verze stránek (Zdroj: http://michal.skrabalek.cz/images/clanky/navody/mobilni-verze/menu-mobilni-verze-stranek.jpg)	20
Obrázek 14: Ukázka mřížového rozložení (Zdroj: https://cdn.tutsplus.com/webdesign/uploads/legacy/tuts/185_designingForResponsiveWebDesign/designing_for_responsive_web_design/grids.png)	21
Obrázek 15: Výsledek studie sledování očí v "F" vzoru (Zdroj: https://media.nngroup.com/media/editor/alertbox/f_reading_pattern_eyetracking.jpg)	22
Obrázek 16: Ukázka rozložení stránky v "Z" vzoru (Zdroj: https://cdn.tutsplus.com/webdesign/uploads/legacy/004_Z_Layout/3.jpg)	23
Obrázek 17: Skrytí navigační nabídky pomocí CSS a JS (Zdroj: http://www.w3schools.com/howto/tryit.asp?filename=tryhow_js_topnav).....	26
Obrázek 18: Ukázka rozložení box modelu (Zdroj: Vlastní tvorba)	34
Obrázek 20: Ukázka použití relativní hodnoty (Zdroj: https://developers.google.com/web/fundamentals/design-and-ui/responsive/fundamentals/imgs/vp.png)	37
Obrázek 19: Ukázka použití absolutní hodnoty (Zdroj: https://developers.google.com/web/fundamentals/design-and-ui/responsive/fundamentals/imgs/no-vp.png).....	37
Obrázek 21: Ukázka Grid-View (Zdroj: Vlastní tvorba).....	37
Obrázek 24: Rozložení na mobilním telefonu (Zdroj: Vlastní tvorba)	41
Obrázek 23: Rozložení na tabletu (Zdroj: Vlastní tvorba)	41
Obrázek 22: Rozložení na stolním počítači (Zdroj: Vlastní tvorba)	41
Obrázek 25: Ukázka použití hodnoty contain (Zdroj: Vlastní fotografie).....	43
Obrázek 26: Ukázka použití hodnoty 100% 100% (Zdroj: Vlastní fotografie)	43
Obrázek 27: Ukázka použití hodnoty cover (Zdroj: Vlastní fotografie)	44
Obrázek 28: Obrázek pro velké obrazovky (Zdroj: Vlastní fotografie)	44

Obrázek 29: Obrázek pro malá zařízení (Zdroj: Vlastní fotografie).....	44
Obrázek 30: Rozhraní nástroje Prepros (Zdroj: Snímek z nástroje Prepros).....	50
Obrázek 31: Rozhraní nástroje PSPad (Zdroj: Snímek z nástroje PsPad)	55
Obrázek 32: Prostředí nástroje GIMP (Zdroj: Snímek z nástroje GIMP)	56
Obrázek 33: Rozhraní nástroje Adobe Muse® (Zdroj: Snímek z nástroje Adobe Muse®).....	57
Obrázek 34: Rozhraní nástroje Weebly (Zdroj: Snímek z nástroje Weebly)	58
Obrázek 35: Rozhraní nástroje Webflow (Zdroj: Snímek z nástroje Webflov)	59
Obrázek 36: Rozhraní aplikace Adobe Dreamweaver® (Zdroj: Snímek z nástroje Adobe Dreamweaver®).....	60
Obrázek 37: Úvodní stránka portálu WebGraphic (Zdroj: Snímek z webového portálu WebGraphic)	66
Obrázek 38: Ukázka rozhraní rozšíření CKEditor na portálu WebGraphic (Zdroj: Snímek z webového portálu WebGraphic).....	67
Obrázek 39: Ukázka testovací otázky (Zdroj: Snímek z webového portálu WebGraphic)	73

13 Seznam příloh

Příloha na přiloženém DVD: Webový portál WebGraphic ve formě projektu v aplikaci Visual Studio s přiloženým databázovým souborem a zdrojovými kódy.



UNIVERZITA HRADEC KRÁLOVÉ
Fakulta informatiky a managementu
Rokitanského 62, 500 03 Hradec Králové, tel: 493 331 111, fax: 493 332 235

Zadání k závěrečné práci

Jméno a příjmení studenta:

Vít Kašpar

Obor studia:

Aplikovaná informatika (2)

Jméno a příjmení vedoucího práce:

Lukáš Vacek

Název práce:

Návrh a realizace vzdělávacího portálu o webové grafice

Název práce v AJ:

Design and implementation of an educational portal for web graphics

Podtitul práce:

Podtitul práce v AJ:

Cíl práce: Zpracování principů tvorby webové grafiky. Vytvoření vzdělávacího portálu o tvorbě webové grafiky.

Osnova práce:

1. Úvod
2. Cíl a metodika práce
3. Teoretická část
 - 3.1 Základní principy webové grafiky
 - 3.2 Kaskádové styly
 - 3.3 Programy určené k tvorbě webové grafiky
4. Praktická část
 - 4.1 Příprava obsahu
 - 4.2 Vlastní návrh portálu
 - 4.3 Ověření
5. Shrnutí výsledků
6. Závěr a doporučení
7. Literární zdroje

Projednáno dne:

Podpis studenta

Podpis vedoucího práce