

Univerzita Hradec Králové

Fakulta informatiky a managementu

Katedra informatiky a kvantitativních metod

**Migrace statického webu na CMS za účelem zefektivnění
správy**

Bakalářská práce

Autor: Ondřej Mlejnek
Studijní obor: Aplikovaná informatika

Vedoucí práce: Ing. Andrea Vokálová

Hradec Králové

srpen 2017

Prohlášení:

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

.....

V Hradci Králové dne 13.8.2017

Ondřej Mlejnek

Poděkování:

Rád bych tímto poděkoval vedoucí bakalářské práce Ing. Andree Vokálové za metodické vedení práce, ochotu a trpělivost.

Anotace

Tato bakalářská práce se zabývá problematikou tvorby šablony pro redakční systém WordPress. Teoretická část přibližuje čtenáři ucelený náhled na redakční systémy. Důraz je kladen na zamyšlení se nad vhodností usnadnění tvorby nových projektů za pomoci redakčních systémů. Dále je možné nahlédnout do zákoutí responzivního designu a tvorby webových stránek s obsahem, který se přizpůsobí zobrazovanému zařízení. Praktická část je zaměřena na udržitelnost a zlepšení komfortu správy webové prezentace pro uživatele bez znalosti kódování. Práce prakticky popisuje tvorbu nové šablony redakčního systémů se zachováním některých funkčních prvků původní prezentace a následnou migraci stávajícího statického webu.

Klíčová slova

WordPress, CMS, šablona, responzivní design, Custom Post Type

Annotation

Title: Migration of static website to CMS for effective administration

This bachelor thesis deals with the creation of a template for the WordPress content management system. The theoretical part brings the reader a comprehensive overview of the editorial systems. Emphasis is placed on considering the appropriateness of facilitating the creation of new projects with the help of editorial systems. It is also possible to look into the responsive design and creation of web pages with content that adapts to the displayed device. The practical part is focused on the sustainability and improvement of the web presentation administration for users without the knowledge of coding. The work describes practically the creation of a new template of the editorial system with preservation of some functional elements of the original presentation and subsequent migration of the existing static web.

Keywords

WordPress, CMS, template, responsive design, Custom Post Type

Obsah

Úvod.....	1
1 Systémy pro správu obsahu	2
1.1 Nejčastější zástupci redakčních systémů.....	2
1.1.1 WordPress.....	4
1.1.2 Joomla!.....	5
1.1.3 Drupal.....	6
1.2 Potřeba redakčního systému	7
2 WordPress.....	9
2.1 Historie.....	9
2.2 Verzování.....	12
2.2.1 Major.....	12
2.2.2 Minor	12
2.3 Komunita	12
2.3.1 Vývojářské týmy.....	13
2.3.2 Komunita v České republice.....	14
2.4 Ekosystém	16
2.4.1 Šablony	16
2.4.2 Pluginy	17
2.4.3 E-komerce.....	18
2.5 Vývojový cyklus	19
2.6 Významné weby založené na WordPressu.....	20
3 Responzivní design.....	21
3.1 Vývoj rozlišení obrazovek.....	21

3.2	Základní pilíře responzibility	22
3.2.1	Flexibilní obrázky	23
3.2.2	Dotazy na médium	24
3.2.3	Layouty	25
3.2.4	Fluidní mřížky	27
3.3	CSS frameworky	28
3.3.1	Foundation	29
3.3.2	Skeleton	29
3.3.3	Bootstrap	30
4	Analýza stávajícího stavu webu	32
4.1	Cíl praktické části	32
4.2	Stávající stav webu	32
4.2.1	Google Analytics	35
4.2.2	Sledované parametry	36
5	Realizace webové prezentace	40
5.1	Instalace CMS	40
5.2	Příprava HTML šablony	42
5.2.1	Rozdělení webové prezentace	42
5.2.2	Očištění šablony	45
5.3	Tvorba šablony pro redakční systém	45
5.3.1	Hlavička stránky	47
5.3.2	Patička stránky	48
5.3.3	Výpis obsahu hlavní stránky	48
5.3.4	Používání funkcí v šabloně	49

5.3.5	Tvorba zobrazení stránek vlastních příspěvků	54
5.3.6	Chybová stránka.....	55
6	Migrace webu.....	57
6.1	Ruční migrace	57
6.2	Automatizovaná migrace.....	58
	Závěr	60
	Seznam použité literatury	61
	Seznam obrázků	65
	Seznam tabulek.....	66
	Seznam grafů	67
	Seznam zdrojových kódů.....	68
	Seznam zkratek.....	69
	Přílohy.....	70

Úvod

V posledních letech se stal internet nedílnou součástí našich životů. Rozšířil se z původní vojenské dálkové komunikace pro radarové systémy do celosvětového systému propojených počítačových sítí. Tato síť se stala velice důležitým prvkem pro prezentaci informací. Ne každý má znalosti kódování a programování webových stránek. Právě na tyto uživatele jsou cíleny systémy pro správu obsahu. Tyto systémy lze snadno spravovat přímo ve webovém rozhraní, obsahují editory pro práci s textem, podobné těm, které uživatelé znají z desktopových textových procesorů. Jedním z jejich zástupců je WordPress, kterým se bude práce z velké části zabývat.

Autorovo setkání s WordPressem bylo poprvé v roce 2011, do té doby využíval pro tvorbu webu výhradně kompletní ruční kódování vzhledu i obsahu. Využití redakčního systému mu ušetřilo spoustu času stráveného vývojem webů. Důležitou součástí pro práci se systémem, který nevyvíjíte osobně je porozumění jeho funkcionalitě a uvědomění si, kde všude je vhodné systém použít a kde je třeba nasadit již vlastní řešení.

Kvalitní webové prezentace odlišuje jak vnitřní programové zpracování, tak z velké části vzhled, kterým se prezentují návštěvníkům navenek. Redakční systémy nabízejí mnoho dostupných šablon od předinstalovaných zdarma ke stažení, až po perfektně zpracované placené. Pokud se však chceme odlišit od ostatních a rádi bychom byli originální a jedineční nezbyvá než přistoupit k tvorbě vlastní šablony. V případě použití stažené nebo koupené šablony se může stát, že stejnou šablonu využívají i jiné weby.

Tématem tvorby vlastní šablony se česká literatura příliš nezabývá, je tedy nutné sáhnout po zahraničních publikacích, nebo využít internetových návodů. Tato práce nabízí čtenářům ucelený návod tvorby vlastní responzivní šablony pro redakční systém WordPress.

1 Systémy pro správu obsahu

Název vychází z anglického (Content management system, CMS), také označovány jako redakční nebo publikační systémy. Jde o systémy určené pro spravování obsahu webových stránek. Jedná se o ucelené řešení pro publikaci obsahu na internetu. Tyto systémy mohou posloužit pro blog, malou firemní prezentaci, ale i velké zpravodajské portály či internetové obchody. Hlavní myšlenkou redakčních systémů je přiblížit a usnadnit správu obsahu webových stránek uživatelům s minimálními znalostmi problematiky tvorby webových stránek.

Využívají oddělení obsahu od vzhledu a funkcionality, díky tomu mohou být spravovány a doplňovány bez zásahu do zdrojových kódů. Samotný publikovaný obsah a nastavení je uloženo v databázi. Je zde kladen důraz na intuitivnost ovládání a práci s obsahem. Práce s texty a dalšími prvky se snaží přiblížit práci v textových editorech. Problém v dnešní době není ani s instalací a nastavením. Existují webové služby, kde se pouhou registrací redakční systém nainstaluje, odpadá tak potřeba znalosti připojení databáze, nahrání redakčního systému na webový hosting a další počáteční znalosti, které mohou začínající uživatele od vlastní instalace odradit.

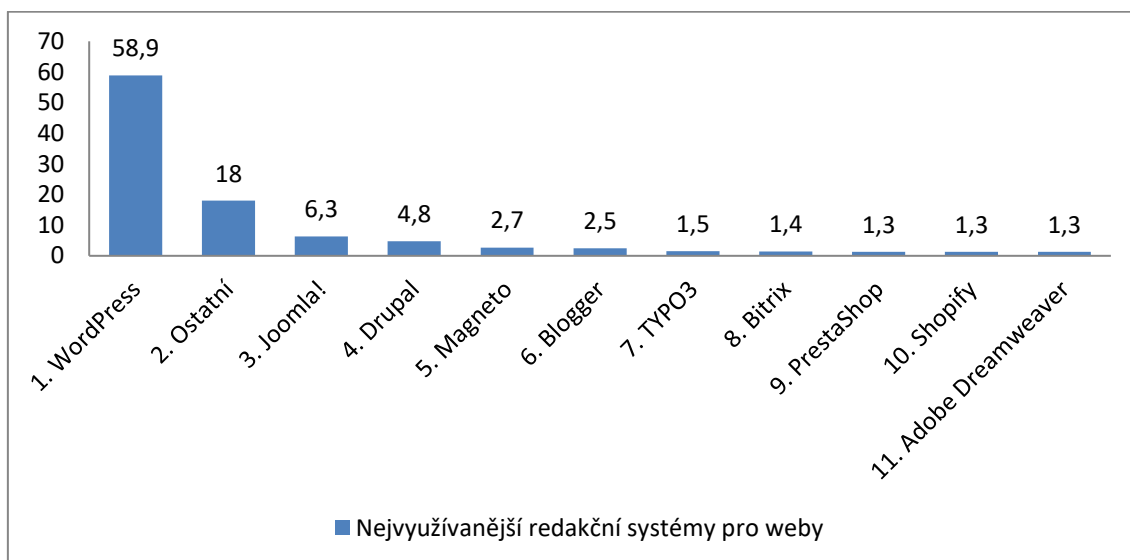
Redakční systémy je možné rozdělit do dvou skupin. První jsou redakční systémy s otevřeným zdrojovým kódem, ty jsou zdarma k použití vyvíjeny komunitou. Druhou skupinu tvoří komerční systémy prodávané jako předpřipravený produkt k použití. Studia vytvářející webové stránky pro klienty využívají buď vlastní naprogramovaný CMS systém nebo se spoléhají na open source projekty. V dalších kapitolách si rozebereme tři nejpoužívanější zástupce nekomerčních redakčních systémů.

1.1 Nejčastější zástupci redakčních systémů

Pro graf níže byl vybrán denně aktualizovaný průzkum W3Techs, který zkoumá deset milionů nejnavštěvovanějších webových stránek podle rankingu Alexa od společnosti Amazon používající tříměsíční průměrný ranking. Průzkum hledá ve stránkách specifické vzory identifikující použití daných technologií. Obvykle je prohledávána hlavní stránka dané domény, případně některé její podstránky.

Subdomény nejsou v průzkumu zahrnuty. Stránky jsou obvykle skenované přibližně jednou za měsíc, denně jsou aktualizovány výsledky a přidávány informace do databáze.

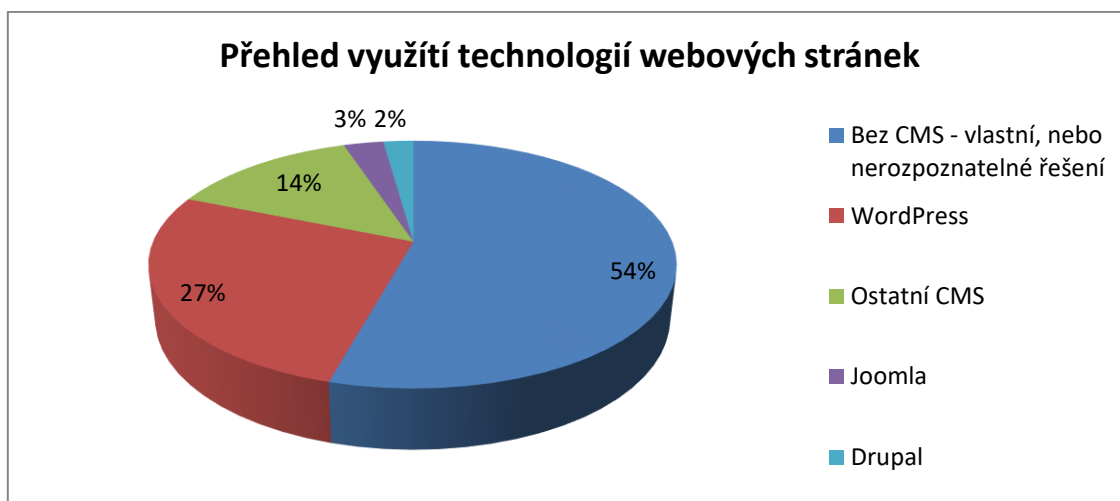
Do měření je zahrnuto přibližně 350 různých CMS systémů. Na grafu níže kategorie ostatní zahrnuje přibližně 340 méně známých redakčních systémů s podílem pod 1 % (1).



Graf 1. Procentuální podíl zástupců nejvyužívanějších redakčních systémů na internetu

Zdroj: Vlastní zpracování, data vycházejí z (2)

Následující graf vyobrazuje procentuální podíl technologií webových stránek na internetu. Z grafu je patrné, že nejčastěji použité řešení pro publikování obsahu na webu je neidentifikovatelného typu. Jedná se o stránky, které nemají typické znaky CMS, podle kterých by mohly být identifikovány. V této nejširší kategorii se mohou skrývat jak vlastní CMS řešení tvůrců webových stránek, tak stránky, na kterých žádný redakční systém nasazen není.



Graf 2. Podíl využití technologií webových stránek na internetu

Zdroj: Vlastní zpracování, data vycházejí z (2)

1.1.1 WordPress

Je jeden z nejoblíbenějších redakčních systémů vyvíjený pod licencí GNU GPLv2 (nebo novější) od Free Software Foundation (3).

WordPress začal v roce 2003 jako malá část kódu, na které pracovalo jen pár lidí. Od té doby se rozrostl na největší blogovací nástroj na světě. Vše okolo tohoto systému tvoří pouze komunita uživatelů. Stovky lidí na celém světě pracují nejen na samostatném redakčním systému, ale i dokumentaci, rozšířeních a motivech. WordPress byl v počátku vyvíjen jako blogovací systém, který vznikl jako projekt vysokoškolského studenta Matta Mullenwega. Matt z části použil původní systém b2/cafelog jehož autor přestal na systému pracovat a komunita tak přišla o svého vůdce (4).

První verze WordPressu vyšla 27. května 2003. Jednalo se verzi 0.7. Na začátku roku 2004 byla spuštěna verze 1.0. Tato verze již připomínala WordPress jak ho známe dnes, jen použití pokročilejších funkcí vyžadovalo znalosti z programování. Zásadní převrat přišel ve verzi 3.0 v červnu 2010, kdy byla přidána možnost vlastních typů příspěvků. Tato funkcionality dodala WordPressu široké spektrum použitelnosti. Díky tomu je nyní možné ho využít i k jiným účelům, než pouze jako blogovací systém. (5)

Matt má slabost pro jazz a ta se odráží i v jeho tvorbě, proto jsou všechny velké aktualizace pojmenovány podle jazzových velikánů. Již od první verze je zde dostupný plugin Hello Dolly. Dle slov autora to není pouze plugin, ale je to symbol naděje a nadšení celé generace shrnuté ve dvou slovech písně od Louise Armstronga. Jeho aktivací se v administračním rozhraní objeví náhodná část textu této písně. (5)

System lze používat dvěma způsoby. Podle toho se dělí i oficiální stránky na doméně prvního řádu:

- wordpress.com – hostování redakčního systému,
- wordpress.org – komunita okolo redakčního systému.

Na doméně wordpress.com je redakční systém k dispozici jako služba nejčastěji pro osobní blogy. Výhodou je, že nemusíte mít žádné zkušenosti s tvorbou stránek. Vše je již automaticky nastavené a instalace se provede pouhým vyplněním formuláře. Vytvořením takového webu vás provede průvodce krok za krokem. Ve verzi zdarma je poté web hostovaný na subdoméně wordpress.com. Oproti tomu web wordpress.org je klíčovým jádrem pro všechny vývojáře. Je zde celý ekosystém komunity, kde je možné CMS stáhnout. Najdeme tu také kompletní dokumentaci, doplňky ke stažení a podporu v podobě fóra.

1.1.2 Joomla!

Druhým nejpoužívanějším redakčním systémem je Joomla!. Obdobně jako WordPress vznikla odčleněním větve z projektu dnes již nevyvíjeného CMS Mambo. Název joomla vychází z fonetického přepisu svahilského slova *jumla*. To v překladu znamená „všichni dohromady”. (6)

Jedná se o systém s otevřeným zdrojovým kódem napsaném ve skriptovacím jazyce PHP. Oproti Wordpressu je možné využít i jiných databázových systémů než pouze MySQL (PostgreSQL, Oracle, SQLite, MSSQL). Tento systém se těší velké oblíbenosti také díky tomu, že je přeložen do 64 jazyků. První verze byla vypuštěna 22. září 2005. Tato verze byla identická s Mambo 4.5.2.3, byly v ní opraveny pouze některé bezpečnostní chyby. (6)

Joomla! se také dlouhou dobu odlišovala pro použití dvou druhů verzí (7):

- STS – Short Term Support je verze určená výhradně pro vývojáře. Doba podpory je přibližně kolem jednoho roku.
- LTS – Long Term Support je verze s dlouhodobou podporou minimálně dva roky.

Pro velké stížnosti komunity bylo od tohoto systému verzí nakonec v dubnu 2014 upuštěno a nyní jsou již všechny vydané verze typu LTS (8).

Již od počátku byl systém navržen tak, aby ho bylo možné rozšiřovat. Je ho možné ohýbat potřebným směrem, aby pracoval tak, jak my potřebujeme. Nejšetrnějším způsobem jak toho docílit, je využít rozšíření, která systém dělí na dva druhy:

- klíčová rozšíření – rozšíření je součástí instalace systému Joomla!,
- rozšíření třetích stran – rozšíření pochází z jiných zdrojů. Oficiálním zdrojem je <https://extensions.joomla.org/>.

Důležité je, že se systém chová po integraci ke všem rozšířením, jako by byla klíčová. Pokud jsou tedy rozšíření dobře zpracována, je těžké rozeznat, kde je hranice klíčového balíčku a začíná rozšíření. (9)

1.1.3 Drupal

Stříbrnou medaili za nejpoužívanější CMS si vysloužil Drupal. Na začátku zrodu tohoto systému stál v roce 2000 student Dries Buytaert. Chtěl vytvořit jednoduchý web, který by umožňoval studentům sdílet poznámky na univerzitě. Vytvořil proto jednoduchou webovou nástěnku, na které si uživatelé zanechávali vzkazy. Dries se později rozhodl umístit tuto aplikaci veřejně na internet, aby si skupina spolužáků mohla stále předávat informace ze života i po dokončení školy. Když registroval doménu pro tento projekt, udělal překlep. Místo slova dorp, jež v nizozemštině znamená vesnice, zaregistroval doménu s názvem drop. Později se komunita ohledně projektu rozrostla a o rok později se ho autor rozhodl uvolnit pod názvem Drupal.

System je postaven na modulární koncepci. Díky tomu je možné si ho upravit podle svých potřeb. Není totiž potřeba mít na firemní prezentaci skripty pro podporu diskuzního fóra, pokud se tam nepoužívá. Podobně zbytečné je mít v jádře podporu pro fotogalerii nebo e-shop. Z tohoto důvodu jsou jednotlivé funkce včetně těch v základní instalaci rozděleny do modulů, které je možné zapínat podle aktuální potřeby. Při tvorbě webu nezapomínejte vypínat nepotřebné moduly. Web poté nespotřebuje tak mnoho systémových prostředků, což ve výsledku přinese rychlejší zobrazování.

Drupalu byla dlouhou dobu vyčítána uživatelská přívětivost. Z pohledu mnoha lidí nebyl systém ihned po instalaci plnohodnotným nástrojem pro správu obsahu. Ač se to může zdát zvláštní, Drupalu až do verze 8 chyběl vizuální editor pro psaní formátovaného textu a text bylo nutné psát s využitím prvků HTML kódu. Od verze 8 je systém nově vybaven WYSIWYG editorem využívajícím open source CKEditor. Samozřejmě bylo ve starších verzích možné WYSIWYG editory používat, ale bylo nutné je doplnit pomocí dodatečných modulů. (10)

1.2 Potřeba redakčního systému

Zákazníci mají různé požadavky a dle konkrétních specifikací by měla být volena konkrétní technologie a systém. Proč pro jednoduchý publikační web zdlouhavě vytvářet na zelené louce vlastní systém pro správu obsahu, když je možné využít dostupné nástroje? Ušetří se prostředky pro tvorbu a údržbu. Ve výsledku se sníží i cena. Položme si proto otázku, je nutné znovu vynalézat kolo?

Existuje mnoho řešení a je vysoce pravděpodobné, že některé z nich je možné vhodně použít pro dané požadavky. Pokud stojíme před otázkou, jaký systém pro své stránky zvolit, měli bychom si nejprve uvědomit, jestli potřebujeme redakční systém či nikoliv.

Zde jsou hlavní body, které jsou typickými znaky poukazujícími na vhodnost použití redakčního systému:

- Chcete často aktualizovat obsah?

- Opakovaně publikujete nové příspěvky?
- Nemáte znalosti HTML a CSS?
- Chcete přidávat multimediální obsah?
- Rádi byste jednoduchou správu obsahu přímo z webového prohlížeče?
- Nechcete kvůli každé změně kontaktovat správce?

Pokud jste na některou z otázek odpověděli ano nebo si myslíte, že v budoucnu byste mohli některý z výše uvedených bodů potřebovat, je pro Vás vhodné využít redakční systém. Počáteční investice může být vyšší, ale při správném výběru se v budoucnu jistě vyplatí.

Obecně lze zjednodušeně říci, že použití statických stránek bez redakčního systému se hodí pro jednoduchý web, kde v budoucnosti neplánujeme aktualizace ani rozšiřování o další obsah, tzv. jednou se web vytvoří a ve stejné podobě zůstává po dlouhou dobu.

Například pokud chceme přidat obrázek na hlavní stranu webu se zaměřením na prezentaci firmy, na kterém není použit redakční systém, musíme provést mnohem složitější operaci. Nejprve fotografii v počítači upravíme na dané rozměry, poté ji nahrajeme přes FTP a vložíme HTML tag s přesnými rozměry na dané místo v kódu stránky.

Oproti tomu redakční systém za nás skrz webové rozhraní zprostředkuje vše od nahrání obrázku až po jeho umístění na stránce. Vše je řešeno vizuálně, tzv. „naklikáním“ v editoru. Díky tomu jsou okamžitě vidět změny.

2 WordPress

Oficiální stránky: <https://wordpress.org/>

Oficiální české stránky: <https://cs.wordpress.org/>

2.1 *Historie*

Historie WordPressu byla z počátku psána jako většina ostatních. Na začátku byl nápad jednoho člověka, který se poohlížel po řešení jeho problému. Psal se rok 2002, když si student Matt Mullenweg nainstaloval blogovací systém b2 neboli cafelog blogging system. Naneštěstí oficiální tvůrce tohoto systému přestal z osobních důvodů na systému pracovat a komunita tak přišla o svého vůdce.

Následně 1. dubna 2003 vydal Matt novou větev b2 na SourceForge s použitím původních částí z cafelogu. S vydáním vlastní verze Mattovi pomáhal Mike Little. Nový název WordPress vymyslela Mattova kamarádka Christine Tremoulet. Tento název se uchytil a používá se doposud.

První verze systému spatřila světlo světa po stovkách commitů v oficiálním SVN repozitáři. Jednalo se o verzi 0.7 vydanou 27. května 2003.

Post / Edit **Team** Options Categories Template Manage Links My Profile View site Logout

Post / Edit

Title:	Category :	Post Status:	Comments:	Pings:	Post Password:
<input type="text"/>	General	Publish	Open	Open	<input type="text"/>

Excerpt:

Post:

B *i* u ~~strik~~ `<p>` `<i>` `b-quote` `image` `link` `X`

PingBack the URLs in this post

TrackBack an URL: (Seperate multiple URLs with commas.)

Obr. 1 Administrační rozhraní WordPressu verze 0.7

Zdroj: (4)

Začátkem roku 2004 se objevila verze 1.0 známá pod označením Davis. Byla to první verze, která připomínala WordPress tak, jak ho známe dodnes. Obsahovala uživatelsky přívětivé URL adresy, možnost dělení příspěvků do kategorií, jednoduchou instalaci, možnost aktualizací a moderování komentářů. Jednalo se o blogovací systém se vším všudy v pravém slova smyslu, i když v té době byly některé funkce vyhrazeny pouze uživatelům se zkušenostmi s programováním. Ve stejném roce změnil konkurent Movable Type cenovou politiku a spousta jeho příznivců shánějících vhodnou alternativu přešla k WordPressu.

Později v říjnu 2004 najal společnost CNET Matta, aby jim pomohl s jejich blogy. Kvůli této práci zanechal školy a odstěhoval se do San Francisca. Ve společnosti setrval do října 2005. Tento měsíc byl velmi rušný, Matt vytvořil antispamový komentářový filtr Akismet. Po odchodu založil společnost Automattic, která při prvním kole financování získala 1,1 miliónů dolarů. Mezi první investory patřili True Ventures, Radar Partners, Polaris Ventrues a samotný CNET. To bylo jasným signálem rostoucí důvěry v tuto platformu. Když Matt skončil v CNET, měl více času

věnovat se vývoji nové verze WordPressu. Aktualizace se uživatelé dočkali v prosinci. Hlavními změnami bylo předělání nástěnky, zefektivnění a zrychlení práce s příspěvky a nahráváním obrázků.

Během roku 2006 pokračoval Automattic v posilování svého jména a upevnění na trhu. Byl přijat nový výkonný ředitel Toni Schneider, bývalý CEO Yahoo!. I když počet uživatelů dosahoval tisíců, společnost měla pouze pět zaměstnanců. V srpnu tohoto roku se konal první WordCamp v San Franciscu. O rok později se konal první WordCamp mimo San Francisco. V současné době se koná 346 WordCampů ve 48 zemích světa.

V lednu 2008 proběhlo druhé kolo financování. Společnost Automattic získala 29,5 miliónu dolarů. Významnými investory byli Polaris, Radar Partners, the New York Times. Jednalo se o velmi důležité investice do budoucna. I když je WordPress open source, hranice mezi ním a společností Automattic je velmi tenká. V říjnu 2008 společnost Automattic koupila službu Gravatar. Jedná se o službu zobrazování avatarů například u komentářů. Tato služba je zdarma dostupná nejen na platformě WordPress. To byl první ze série nákupů souvisejících technologií za účelem zlepšení WordPressu pro uživatele. V tomto roce byl také zprovozněn adresář šablon. Kdokoliv tak může vytvářet a publikovat šablony zdarma.

Velice zásadní aktualizace vyšla v červnu 2010. Byla vydána verze Thelonus 3.0, která obsahovala vlastní typy příspěvků. Tato funkcionalita velice pomohla přizpůsobivosti a univerzálnosti WordPressu. Díky tomu je možné pracovat s různými druhy příspěvků, které jsou po přidání přímo integrované v administraci.

Rok 2011 přinesl WordPress 3.1 a s ním administrátorskou lištu. Pokud prohlídí web přihlášený správce, může tak pomocí lišty upravit danou stránku, na které se nachází. Odpadla tak nutnost hledat v administraci stránky, které bylo třeba upravit. V tomto roce již na WordPressu běželo 12 % všech webů.

V prosinci 2013 přišla velká změna s verzí WordPress 3.8 Parker. Poprvé po takřka 9 letech od verze Davis se změnila nástěnka v administraci. Další vítanou novinkou

byla změna administračního rozhraní pro responzivní zobrazení. Web byl poté snadněji spravovatelný i na mobilních zařízeních.

V dubnu 2014 došlo ke změně provádění aktualizací. Z důvodu zvýšení bezpečnosti jsou aktualizace na vyšší verzi prováděny automaticky. V květnu bylo další kolo financování, kde bylo získáno 160 miliónů dolarů. Poté bylo potvrzeno, že hodnota společnosti Automattic je přes jednu miliardu dolarů. Ve společnosti je zaměstnáno pouze 256 zaměstnanců. (4) (5)

2.2 Verzování

Veřejnosti jsou publikovány dva druhy stabilních verzí major a minor. Ke dni 6. srpna 2017 bylo vydáno celkem 251 stable verzí, z toho 34 je hlavních.

2.2.1 Major

Hlavní verze je udávána pomocí dvojčíslí například 4.6. Tyto verze přidávají nové uživatelské funkce a vývojářské rozhraní. U většiny vyvíjených aplikací se standardně nedrží zpětná kompatibilita hlavních verzí. WordPress však, kvůli usnadnění migrace uživatelů a vývojářů, zpětnou kompatibilitu ke starším verzím dodržuje. Všechny hlavní verze mají jméno po jazzových hudebnících. (11)

2.2.2 Minor

Vedlejší verze jsou číslovány pomocí trojčíslí například 4.6.1. Tyto verze jsou vydávány pro zlepšení bezpečnosti. Opravují zranitelnost a kritické chyby (11).

2.3 Komunita

V posledních 11 letech se komunita okolo WordPressu značně rozrostla. Komunita je vedena dvěma hlavními skupinami, vnitřním vedením a dobrovolníky komunity. Do komunity se může zapojit každý, od vývojářů jádra přes moderátory oficiálního fóra až po organizátory WordCampů.

2.3.1 Vývojářské týmy

Na poslední hlavní větvi 4.6 Pepper Adams pracovalo 272 vývojářů jádra. Vývojář jádra je člověk, který pravidelně přispívá k aktualizacím a vylepšením WordPressu. Projekt je řízen manažerským týmem pod vedením spoluzakladatele Matta Mullenwega. Jeho tým řídí celý projekt včetně vývoje jádra, oficiálního webu a aktivit komunity. Vedoucí role jsou zaslouženě udělovány na základě schopností a respektu v komunitě. (12)

Vývojáři se dělí na níže uvedené skupiny. Z důvodu zachování autenticity jsou jména týmů v originálním znění.

The WordPress Core Leadership Team

Je veden hlavním vedoucím týmu, Mattem Mullenwegem a pěti řídicími vývojáři. Jedná se o hlavní autoritu technických rozhodnutí.

Contributing Developers

Přispívající vývojáři se skládají z bývalých nebo aktuálních vývojářů, kteří si získali respekt v komunitě. Do této skupiny se řadí také hostující přispěvatelé, kteří se podílejí jen na specifických částech a komponentách.

Core Contributors

Skupina stovek dobrovolníků přispívajících k vývoji jádra.

UX and Design

Tým starající se o design a uživatelské rozhraní.

Support

Podpora fóra je složena z týmu dobrovolníků starajících se o moderování, mazání spamu a udržování přehlednosti.

Documentation

Dokumentační tým je zodpovědný za vše okolo dokumentace napříč WordPressem. Spravuje a udržuje kodex a k němu i vývojářský web. Kodex je základním zdrojem informací o WordPressu. Vývojářský web spravuje komunita. Do kodexu může

přispívat každý, kdo má vytvořený účet u wordpress.org, pro lepší srozumitelnost jsou některé stránky přeloženy do mnoho jazyků.

Mobile

Jedná se o open source software, vyvíjený týmem dobrovolníků. V současné době je možné sehnat aplikaci jak pro Android, tak i pro iOS. Obě aplikace jsou dostupné na GitHubu a k jejich vývoji může přispět každý.

Theme Reviewers

Témata odesílána do oficiálního webového rozhraní kontroluje tým dobrovolníků. Témata musí být v souladu se standardy WordPressu.

Plugin Reviewers

Tým dobrovolníků kontroluje také pluginy, které jsou vkládány do oficiálního adresáře.

Community Blogs and Communication

Tým spravuje komunitní blog WordPressu. Na blog se nejčastěji umísťují informace o nově vydaných verzích nebo postupu vývoje.

Bez těchto všech týmů složených z dobrovolných členů by platforma nemohla existovat. Tito lidé opravují bezpečnostní chyby, stále přidávají nové funkce, testují a udržují CMS moderní, píší dokumentace, vytváří šablony a pluginy, organizují setkání a mnoho dalšího. Patří jim za to velké díky.

2.3.2 Komunita v České republice

Oficiální stránky české komunity: <http://naswp.cz/>

Okolo WordPressu je v České republice stále rozrůstající se komunita. Konají se zde různé typy srazů.

WordCamp konference

Jedná se o oficiální lokální akci pořádanou komunitou. V České republice proběhl již 3. ročník pořádaný komunitou Náš WordPress. Akce se pořádá jednou ročně v Praze. V roce 2014 se na první český WordCamp registrovalo 230 lidí. Cílem bylo

nastartovat komunitu v Česku. O rok později se akce zúčastnilo opět 230 lidí. Poslední WordCamp, který se konal v únoru 2016, navštívilo již 300 návštěvníků. Změnou oproti předchozím ročníkům bylo rozdělení bloku přednášek na dva nezávislé proudy. Ve stejný čas se konaly přednášky na oddělených sálech. První se zaměřil na uživatele, na své si přišli jak začátečníci, tak pokročilí uživatelé. V druhém se konaly přednášky pro vývojáře. Tento systém se osvědčil a organizátoři v něm budou pokračovat i v dalším roce. Díky tomu se akce otevřela pro více lidí. Na příští rok si dávají za cíl přitáhnout 400 lidí.

Informace ohledně návštěvnosti byly získány díky ochotě jednoho z hlavních členů komunity Radka Kučery, za to mu patří poděkování.

WP Pivo

Tento sraz je neformální posezení, kde se scházejí příznivci tohoto redakčního systému a diskutují. Akce nemá přesně danou koncepci, nejsou předem dána témata. Pouze přijdete, něco se nadhodí a okolo toho se diskutuje. Cílem je setkat se s lidmi se společným zájem, ať jste programátor, nebo jen správce stránek. Vítáni jsou všichni.

WP konference pro začátečníky

Celodenní akce zaměřená na začínající uživatele je cílená na základy práce, vytvoření správných návyků. Program návštěvníka provede od základů plánování webu, instalaci, správu až po konečnou propagaci stránek. Již proběhlo šest ročníků. Cena se obvykle pohybuje kolem symbolických 100Kč a je v ní zahrnuto i občerstvení.

WP konference pro pokročilé

Zaměřuje se na zkušenější uživatele WordPressu. Cílem těchto konferencí je něco se přiučit a získat kontakty na lidi okolo tohoto CMS. Již proběhlo sedm ročníků.

WP Weekend

Jedná se o výhradně vývojářskou akci. Proběhl zatím první ročník s omezeným počtem 30 účastníků. Cílem akce je sdružovat a rozšiřovat komunitu vývojářů, sdílet jejich zkušenosti a nápady. Akce se skládá z workshopů a přednášek.

Další meetupy a konference

Dále se konají konference na různých místech České republiky, například na univerzitě v Opavě, nebo meetupy v Brně. Každý se může připojit ke komunitě a uspořádat akci. Cílem je rozšiřovat povědomí a komunitu uživatelů WordPressu.

2.4 Ekosystém

Tato open source platforma pomohla k úspěchu mnoha firmám nejenom v zahraničí. S popularitou tohoto řešení přibývá lidí, které práce okolo WordPressu živí. Zdaleka se nejedná jen o vývojáře tohoto CMS systému. Je mnoho pozic více či méně spjatých s tímto řešením. Podle rozhovoru s Mattem Mullenwegem k desátému výročí vzniku živí WordPress dvacet tisíc lidí a díky tomu setrval až dodnes. Dle Marka Gavaldy z hostingové společnosti Kinsta je toto číslo velice těžké odhadnout a může být i několikanásobně vyšší (13). Jedná se například o vývojáře, blogery, grafiky, společnosti poskytující hostingové služby, tvůrce pluginů, konzultanty, freelancery, komerční společnosti zabývající se tvorbou webu a mnohé další. V dalších částech bude zacíleno na nejsilnější milníky ekosystému.

2.4.1 Šablony

Šablona je v redakčním systému prostředek pro zobrazení dat uživateli. Data jsou z aplikační vrstvy zobrazena přes šablonu směrem k uživateli. Je to základní designový prvek. Tvorbou šablony krok za krokem se zabývá praktická část této práce.

Tržiště šablon tvoří značnou část komerční sféry ekosystému WordPressu. Po instalaci lze nastavit jednu z výchozích šablon, nicméně drtivá většina uživatelů se chce odlišit. Samozřejmě existují šablony volně ke stažení, ale pokud jsou kvalitní, najdou si je velké množství uživatelů. A stále vyvstává otázka prestiže a originality. Nikoho nepotěší, když narazí na konkurenční webovou prezentaci a zjistí, že má shodný design stránek.

S rostoucí popularitou je i rostoucí poptávka po prémiových šablonách. Z mnoha důvodů je jednou za čas potřeba modernizovat vzhled webu, dodržovat aktuální

trendy. Na tvorbě komerčních šablon postavilo svůj obchodní plán mnoho společností i designerů. Ekosystém okolo šablon je tak velký, že například společnost Envata založila tržiště s šablonami Themeforest, které na začátku roku 2017 obsahovalo přes 8500 šablon s cenou začínající na 13 \$. Nejdražší šablona pak vychází na 350 \$ (14). Na úspěchu společnosti je dobře vidět, jak velký ekosystém okolo WordPressu vznikl. (5)

2.4.2 Pluginy

Stejně jako se různé projekty chtějí odlišit vzhledem díky šablonám, tak se často liší i ve funkcionalitě. K tomu slouží právě pluginy. Kouzlo WordPressu spočívá v jeho rozšiřitelnosti a univerzálnosti. Někteří vedoucí vývojáři tvrdí, že nejzajímavějších funkcí dosáhne WordPress až po dodatečné instalaci pluginů. Tím se otevírají dveře pro velké množství vývojářů a firem, díky kterým je používání systému opět o kousek pohodlnější. Tato vylepšení, která se instalují jednoduše z administrace, mohou změnit prakticky celý systém. Pokud chceme něco, co v základu WordPress neumí, stačí prohledat databázi pluginů a jistě najdeme vhodného kandidáta, který nám danou funkci zpřístupní. Tato rozšíření lze snadno hledat jak přes oficiální stránky, tak i v administraci WordPressu. (15)

V lednu 2017 jich je na stránkách ke stažení přes čtyřicet osm tisíc s více jak miliardou a půl stažení. (16) Každý může přidat rozšíření do oficiálního úložiště. Mnoho jich je zdarma ke stažení, některé velmi pokročilé jsou placené.

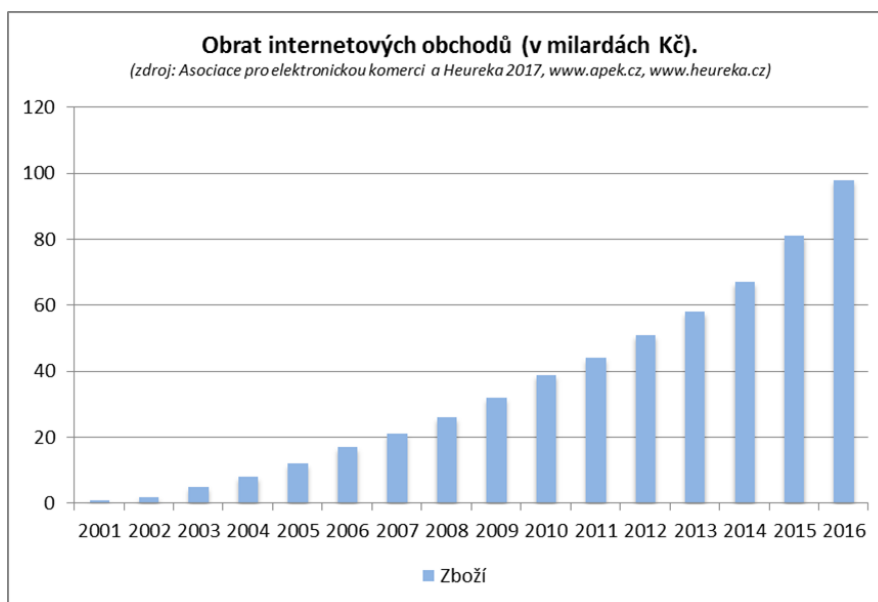
Napadlo Vás někdy mít vícejazyčný web? I toto je u WordPressu možné. Slouží k tomu placený plugin WPML od společnosti OnTheGoSystems, který používá již přes čtyři sta tisíc webových stránek. Nejpopulárnější je v Evropě a Severní Americe a mezi nejpoužívanější jazyky patří angličtina, španělština, francouzština, němčina. (17)

Nestačí Vám standartní WYSIWYG editor? Jednoduše nainstalujete plugin TinyMCE Advanced a můžete si tlačítka v editoru skrývat přesouvat a nastavit podle svých představ. Jedná se o jeden z nejstahovanějších pluginů s více než miliónem aktivních instalací. (18)

Pro názornost jsou výše uvedené dva pluginy. První je velmi pokročilý a placený pro vytvoření vícejazyčného webu, druhý je jen jednoduchá utilita, která nám umožní změnit podobu editoru textu. Konkrétními použitými pluginy se bude zabývat praktická část práce.

2.4.3 E-komerce

Internet rozšířil možnosti prodeje do výšin, o kterých se nám předtím ani nesnilo. Z průzkumu Asociace pro elektronickou komerci, která provádí každoročně výzkum ročního obrátu v České republice u zboží a služeb prodaných přes internet vyplývá, že je toto odvětví stále na vzestupu. Počet uživatelů se zkušenostmi s nakupováním na internetu se již příliš nezvyšuje, z toho vyplývá, že uživatelé mají k nakupování přes internet stále větší důvěru a utrací za něj každoročně více finančních prostředků. (19)



Obr. 2 Graf vývoje obrátu internetových obchodů za zboží

Zdroj: (19)

Velká část internetu se zaměřuje na prodej různých druhů služeb, produktů a digitálního obsahu. WordPress jako univerzální CMS se rozšířil i do této oblasti a je možné ho díky pluginům přestavět na plnohodnotný internetový obchod. Nejznámějším pluginem v této oblasti je WooCommerce. Se přibližně dvaceti

milióny stažení pohání přes 39 % všech internetových obchodů. Jedná se o open source řešení od společnosti Automattic. (20)

S příchodem velmi diskutované EET (elektronické evidence tržeb) do České republiky nezůstávají pozadu ani vývojáři pluginů pro WooCommerce a přinášejí řešení v podobě Toret EET, které umožní propojení internetového obchodu s portálem ministerstva financí pro evidování tržeb z prodeje. Plugin stojí jednorázově 1000 Kč. (21) To je příkladem, že i na open source plugin WooCommerce je možné dodělat další rozšíření a ta prodávat dále. Jedná se pouze o jeden příklad z mnoha, díky kterým se ekosystém může stále rozrůstat.

2.5 Vývojový cyklus

Každý cyklus vydání major verze vede jeden nebo více hlavních vývojářů WordPressu. Celý cyklus trvá okolo čtyř měsíců od prvního zasedání (22). Kroky vývojového cyklu:

1. Plánování a zabezpečení vedení týmu

Hlavní tým se sejde v chatovací místnosti #wordpress-dev. Diskutuje se o novinkách v příští verzi. Další přispívající vývojáři se zapojují do diskuze.

2. Start vývojových prací

Hlavní členové jsou rozděleni jako vedoucí vlastních týmů. Startuje vývoj, každý pracuje na svém přiděleném úkolu.

3. Beta

Jsou uvolňovány beta verze. Testeři se snaží hledat chyby a zranitelnost. Nejsou vyvíjeny žádné další novinky. Pouze se opravují nalezené chyby. Vývojáři doplňků a šablon jsou vyzváni k otestování kompatibility.

4. Finální verze

Veškerý vývoj je zastaven. Práce probíhá pouze na překladech a částech nezasahujících do kódu jádra.

5. Zpřístupnění finální verze

Nová verze je oficiálně zpřístupněna. Administrátoři jsou upozorněni na možnost aktualizace v administraci.

2.6 Významné weby založené na WordPressu

Jak již bylo zmíněno v kapitole 1.1 Nejčastější zástupci redakčních systémů WordPress využívá 27 % všech webů. Některé weby možná navštívíte a ani nevíte, že na jejich pozadí tepe redakční systém WordPress. Pojd'me se tedy podívat na pár nejznámější webů nebo renomovaných značek, které pro svoji internetovou propagaci zvolili tento velmi populární CMS. Jedná se o weby jak ze zahraniční, tak i z České republiky:

- [The New Yorker](#) – web Magazínu New Yorker,
- [Playstation](#) – oficiální blog platformy Playstation,
- [UPS](#) – oficiální blog americké přepravní služby,
- [IBM](#) – oficiální blog společnosti,
- [Facebook](#) – oficiální zpravodajská místnost,
- [Mercedes-Benz](#) – oficiální stránky výrobce automobilů,
- [BlaskBerry](#) – oficiální blog výrobce mobilních telefonů,
- [Sony music](#) – oficiální web hudebního vydavatelství Sony,
- [Lesy ČR](#) – oficiální web státního podniku Lesy České republiky,
- [Mobilizujeme](#) – web o mobilních telefonech,
- [Play](#) – magazín o hudbě,
- [Mobility UHK](#) – informační web o studijních pobytech.

3 Responzivní design

Po delší době strávené vývojem webových stránek si začnete uvědomovat, kolik se v tomto odvětví děje změn, jak rychle se mění trendy. Co bylo moderní nedávno, nemusí již nutně platit dnes. S rozšířením mobilních zařízení a dostupnosti bezdrátových sítí, které dnes nabízí špičkové rychlosti, přistupuje tímto způsobem na web stále více uživatelů. Ruku v ruce s technologickým pokrokem narůstá i standardní rozlišení obrazovek. Vysoké rozlišení však nemusí znamenat velkou obrazovku. Není tomu tak dávno, kdy byly moderní obrazovky s poměrem stran 4:3. Těmto trendům je třeba věnovat pozornost a nenechat si tzv. ujet vlak. Nedávno nám stačilo přizpůsobit web na dané rozlišení většinou 1024 pixelů a s tím jsme si vystačili. Nyní je však nutné počítat s širokou škálou různorodých zařízení od mobilních telefonů přes tablety, notebooky, desktopové počítače až po televizory. Některá zařízení také mohou pracovat jak ve vodorovném, tak ve svislém režimu.

Cestou ke stejnorodému zobrazení na jakémkoli zařízení může být responzivní design. Nezměnily se pouze rozměry zařízení, které uživatelé používají k prohlížení, ale i způsob jakým je ovládají. Základní práci s myší doplnily dotykové displeje s nimi přišla gesta a v další řadě hlasové ovládání. Responzivní webdesign je sada technik a technologií s cílem zpřístupnit aplikace co nejvíce uživatelům na různých zařízeních. I bez použití těchto technologií je možné na zařízeních weby zobrazit, ale chybí tu uživatelský komfort v ovládání, který může v době velké konkurence mnohé zákazníky odradit. (23)

3.1 Vývoj rozlišení obrazovek

S implementací webových prohlížečů do různorodých zařízení přibývá i různorodost rozlišení obrazovek uživatelů.

V následující tabulce je zobrazen vývoj rozlišení obrazovky návštěvníků stránky W3Schools za posledních 5 let. W3School upozorňuje, že jejich statistiky se mohou lišit od jiných webů, protože každé stránky navštěvují odlišní lidé.

Je zde vidět trend přibývajících mobilních zařízení tzv. přibývá návštěvníků s nižším rozlišením než 1024x768. Zároveň si můžeme všimnout, že přibývá návštěvníků s širokoúhlými obrazovkami, poměrem stran zařízení 16:9 na úkor poměru 4:3. V kategorii vyšší jsou obsažena všechna rozlišení vyšší než 1024x768, která by se do tabulky nevešla. V této kategorii jsou v roce 2017 zástupci 1536x864, 1680x1050, 1920x1200, 1280x720, 1360x768, 2560x1440 a další. V minulých letech se v této kategorii rozlišení příliš neměnila. V kategorii nižších rozlišení, než je 1024x768 nejsou konkrétní dále specifikována.

Stránky W3School jsou zaměřené především na výuku tvorby webových stránek, proto se předpokládá, že většina příchozích přistupuje na web z počítače. I zde je však vidět posun. Za posledních 5let se počet uživatelů přistupujících na web z mobilních zařízení takřka ztrojnásobil. (24)

Rozlišení	2017	2016	2015	2014	2013
Vyšší	20,6 %	18,7 %	19,7 %	21 %	24 %
1920x1080	17 %	18 %	16 %	13 %	11 %
1600x900	5 %	6 %	6 %	6 %	5 %
1440x900	6 %	6 %	7 %	7 %	7 %
1366x768	35 %	35 %	33 %	31 %	25 %
1280x1024	5 %	6 %	7 %	8 %	10 %
1280x800	4 %	4 %	5 %	7 %	8 %
1024x768	3 %	3 %	4 %	6 %	9 %
Nižší	4,4 %	3,3 %	2,3 %	1 %	1 %

Tab. 1 Trend rozlišení obrazovky na webu W3School za posledních 5 let.

Zdroj: Vlastní zpracování, data (24)

3.2 Základní pilíře responzibility

Poprvé termín *Responsive Web Design* použil Ethan Marcotte ve svém stejnojmenném článku na webu A List Apart. V článku pojednává o potřebě inovací a pomíjivosti webového designu v kontrastu s architekturou, která je takřka věčná.

Zaměřuje se také na základní pilíře responzivních webů a na potřeby nového přístupu k obsahu. Cílem těchto pilířů je dosáhnout zobrazení, které se přizpůsobí zařízení. Vše napíšeme pouze jednou a je možné to spustit na jakémkoliv zařízení. Díky tomu je možné se více soustředit na uživatelský obsah. (25)

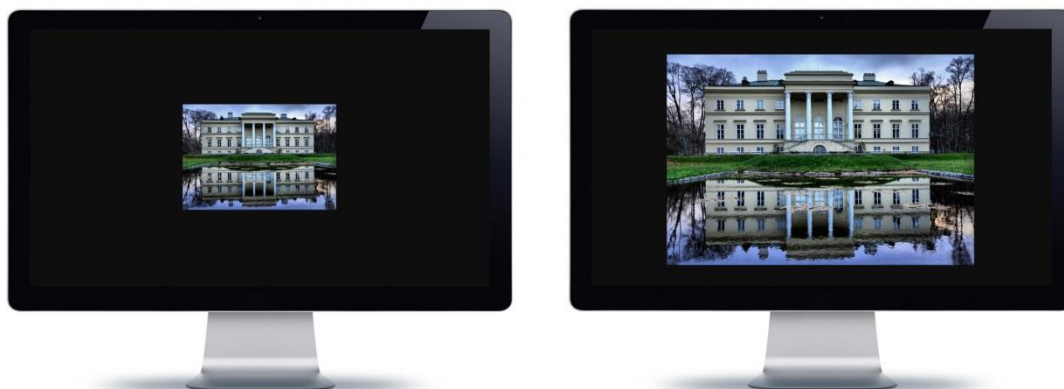
3.2.1 Flexibilní obrázky

Jednu z hlavních částí webového obsahu tvoří obrázky. Mohou být vhodným doplňkem k textům nebo mají funkci pouze vizuálního designového charakteru. Adaptivní obrázky je druhé možné označení používané specifikací HTML5. U obrázků mohou nastat problémy s předurčením rozlišení, v kterém se obrázek bude zobrazovat. Například obrázek o dané šířce 1280 pixelů se nám na zařízení s rozlišením 320 pixelů nezobrazí celý. Aby se velikost obrázku přizpůsobila zařízení, je možné aplikovat maximální šířku na 100 %. Tím se šířka obrázku bude nastavovat podle obsahu až do plné velikosti originálu. Tento postup však neřeší problém datové náročnosti. Pokud v tomto případě použijeme obrázek ve vysokém rozlišení, bude se muset vždy stáhnout celý.

Druhým problémem je hustota pixelů, která stále narůstá. Není tomu tak dávno, co se vysoká rozlišení používala především pro zařízení s velkým displejem typicky televizory. Dnes se však rozlišení Full HD i 4K používá pro tablety a mobilní telefony. Tím dochází k velkému nárůstu obrazových bodů. Podle Borise Smuse byla standardem do roku 2008 hustota 150 ppi v roce 2012 již 300 ppi. (26)

Konkrétně Sony Xperia Z5 Premium již dosahuje hodnoty 801 ppi při velikosti obrazovky 5,5 palce, takto vysoké hodnoty je dosaženo použitým rozlišením 4K.

Na následujícím obrázku je vyobrazeno zobrazení obrázku s šířkou 1280 pixelů na monitorech s vysokým rozlišením 4K vlevo a Full HD vpravo.



Obr. 3 Porovnání rozlišení obrazovek 4K a Full HD při zobrazení stejného obrázku s šířkou 1280 pixelů.

Zdroj: monitor (27), foto a zpracování autor

Častou otázkou na toto téma je zachování kvality zobrazení obrázků pro co nejširší škálu zařízení a s tím i optimalizování datové náročnosti. Přímá cesta v tomto ohledu neexistuje, prohlížeče zatím tyto funkce plně neimplementují. Pro řešení této problematiky existuje polyfill. Polyfily jsou kódy nejčastěji psány v JavaScriptu, které mají za úkol překlenout nedostatky prohlížečů v implementaci nových funkcí. Díky nim je možné například zobrazit kulaté rohy v prohlížečích, které nepodporují tuto vlastnost.

Jeden takový polyfill řeší i problém efektivity načítání obrázků podle rozlišení zařízení. Jedná se o Picturefill od Scotta Jehla, který rozšiřuje možnosti používání elementu *picture*. (23) (28)

3.2.2 Dotazy na médium

Jedná se o zásadní pilíř při návrhu responzivní webové stránky. Dotazy na médium jsou pilířem nejlépe podporovaným webovými prohlížeči. Díky nim lze klást dotazy na vlastnosti zařízení. Podle těchto vlastností můžeme poté reagovat určitou sadou pravidel. Nejznámější dotazem na médium již z předchozích let jsou tiskové výstupy. Na webu je žádoucí aplikovat optimalizaci pro tisk. Když na webové stránce nepoužíváme odlišné styly pro tisk, dojde k vytisknutí ve formě, jakou vidíme na obrazovce.

```
<link type="text/css" rel="stylesheet" href="tisk.css"
media="print">
```

Kód 1 Příklad typu média v tomto případě tiskárna

Zdroj: Autor

Příklad uvedený výše nám umožní ovlivnit tiskový výstup. Jedná se o připojení CSS stylu tisk.css, který v sobě obsahuje sadu pravidel pro formátování a zobrazení webové stránky. Tato pravidla se aplikují pouze v případě, že použité medium bude tiskárna, to zajišťuje použití atributu *media="print"*.

Obdobným způsobem jako s tiskárnou můžeme sadu pravidel použít pro určité dotazy na média. Hlavním ukazatelem je v tomto případě rozlišení displeje zařízení. Vlastnosti daného zařízení poté ovlivní, jaká šablona se načte. (23)

```
<link type="text/css" rel="stylesheet" href="mobilni.css"
media="screen and (max-width:640px)">
```

Kód 2. Příklad použití dotazu na media s maximální šířkou 640 pixelů

Zdroj: Autor

Tento příklad aplikuje sadu stylů ze souboru mobilni.css pro všechna rozlišení s šířkou maximálně 640 pixelů.

3.2.3 Layouty

Jádrem každého dobrého webu je smysluplné rozvržení základních prvků na stránce. Dlouhou dobu byly budovány weby, které byly neohebné, kvůli tomu nebylo možné přizpůsobit se narůstající nevyočitatelnosti zobrazovacích zařízení. John Allsopp již v roce 2000 v článku Tao webového designu pro A List Apart přišel s myšlenkou: „Místo abychom s nevyočitatelností bojovali, je třeba, abychom ji přijali jako fakt“ (29). V rychle se rozvíjejícím odvětví jako je webový design se nakonec ukázalo, že Allsoppův náhled na věc předběhl dobu a je i dnes stále aktuální. Domníval se, že se na nejsilnější stránku webu, kterou je flexibilita díváme jako na omezení. Web je ve své podstatě flexibilní a je jen na vývojářích a designérech tuto flexibilitu přijmout a udělat z ní cnost.

V oblasti webů existují čtyři různé typy layoutů. Každý má své silné i slabé stránky, které si rozebereme v následujících podkapitolách.

Layout s pevnou šířkou

Rozvržení s pevnou šířkou je omezeno konkrétní danou hodnotou v pixelech. Nejběžněji volenou šířkou je dnes 960 pixelů. Layout s pevnou šířkou je na webu nejčastěji používán. Vzbuzuje u autorů milnou iluzi, že mají web kompletně pod kontrolou. Díky tomu je možné vytvářet graficky propracované návrhy.

Největším problémem je potřeba optimalizovat web pro nějaké rozlišení. Při rozhodování, jak široký web vlastně bude, je možné vycházet ze statistik návštěvnosti a zaměřit se na nejčtenější rozlišení. I tento údaj však může být nevyovídající. Daná skutečnost je složitější, než se může zpočátku zdát. Uživatel může mít okno otevřené pouze na části monitoru. Někdo používá postranní panely, nebo různé pluginy, které se zobrazují na straně a zredukují tak prostor pro zobrazení plné webové stránky, která se poté zobrazuje s vertikálním posuvníkem. Nicméně ani velké rozlišení na straně návštěvníka stránek není výhodou. Typické je poté zobrazení se stránkou uprostřed a velkým prázdným místem po stranách, které je známe z mnoha webových stránek. Největším problémem tohoto typu rozvržení je nedostatečná přizpůsobivost zobrazení na dnešních rozmanitých typech zařízení.

Plovoucí layout

U plavoucích rozvržení jsou rozměry určovány v procentech, ne pevně v pixelech. Tím pádem může být layout mnohem flexibilnější. Například hlavní okno pro obsah může být s šířkou 70 % postranní panel 20 % a zbylý volný prostor 10 %, tím je vyplněno 100 % šířky obrazovky. Poté již nezáleží, na jakém zařízení návštěvník pracuje. Web se vždy přizpůsobí, daná procenta se přepočítají na konkrétní pixely podle fyzického rozlišení zařízení. Toto rozvržení odstraňuje nutnost horizontálních posuvníků. Lépe je využít dostupný prostor a nevzniká velké množství prázdného prostoru u vysokých rozlišení. Samotný plavoucí layout není sám o sobě postačujícím řešením responzivního designu. Nestačí ho pouze aplikovat a stránka bude vypadat všude dobře. Šířka řádku může být na vysokých rozlišeních moc široká, nebo naopak při malých rozlišeních úzká.

Elastické layouty

Tato rozvržení jsou z velké míry podobná plovoucím layoutům s rozdílem, že pro určování rozměrů je používáno místo procent relativních jednotek. Typicky se používá jednotka em. Jedná se o velikost základního písma definovaného v CSS. Pokud použijeme výšku písma 16 px bude poté 1 em 16 pixelů. Použití tohoto typu layoutů napomáhá držet pod kontrolou typografii. Pro nejlepší čitelnost je doporučená délka řádku mezi 45-70 znaky. Když chceme dodržet toto pravidlo, můžeme si definovat šířku kontejneru pro obsah na 60 em. To zajistí, že bude dodržena délka řádku. Výhodou elastických layoutů je, že elementy definované tímto stylem se při zvětšování velikosti písma proporcionálně mění také. Tato rozvržení však neomezí zobrazení horizontálního posuvníku ani prázdná místa po stranách.

Hybridní layouty

Jak už z názvu vyplývá, tento layout je kombinací dvou a více druhů rozvržení. Používá se za předpokladu, že některý element chceme na stránkách nechat v přesně definované velikosti. Kupříkladu se může jednat o službu třetí strany, jejíž velikost nejsme schopni ovlivnit. Zbytek webu může používat plovoucí layout. Tento přístup může způsobit komplikace a může docházet k zobrazení horizontálního posuvníku.

Jaký layout zvolit

Obecně platí, že není žádné univerzální správné řešení a je třeba volit layout podle konkrétního projektu. Z hlediska responzivity je vhodné nepoužívat pevný layout. Za použití dotazů na media je možné i takovýto web udělat částečně responzivní, ale poté postrádá univerzálnost. S použitím flexibilních layoutů se vydáváme responzivitě naproti a s použitím dalších technik je možné dosáhnout přímou cestou univerzálnosti webového zobrazení. (30)

3.2.4 Fluidní mřížky

Stále narůstá různorodost zařízení přistupujících na web, různé velikosti obrazovek, množství rozlišení. Na tento fakt je třeba reagovat. Stále více potřebujeme řešení,

kteře by vyhovovalo narůstající škále zařízení. V tomto ohledu nám mohou pomoci fluidní mřížky. Použitím mřížky můžeme rozvrhnout stránku na části, do kterých se umisťují jednotlivé elementy webové stránky. Vše je založeno na flexibilních layoutech, které zajistí, že se dané elementy přizpůsobí konkrétnímu zobrazení. Nejjednodušší mřížky bývají tvořeny jedním, dvěma nebo třemi sloupci.



Obr. 4 Fluidní mřížky v rozvržení webu zobrazené na různých zařizních.

Zdroj: (31)

3.3 CSS frameworky

Framework je obecné označení pro softwarové řešení určitých problematických oblastí vývoje softwaru. V oblasti CSS se pro tato řešení užívá název Boilerplate, nebo jednoduše CSS frameworky. Jedná se o předpřipravené znovupoužitelné kódy, které mohou být využívány jako základ pro nové projekty. Mají za úkol usnadnit a zrychlit vývoj. Dobré frameworky jsou odzkoušené webovou komunitou, takže jsou optimalizované. Díky tomu lze začít práci opravdu rychle a ušetřit tak čas, protože v začátcích projektu není nutné vše pracně psát znovu. V základu již zvládají responzivitu a při použití stačí dodržovat zásady daného frameworku. Odměnou může být snadnější tvorba responzivních layoutů.

I když se jedná o užitečnou věc, není radno to s použitím frameworků přehánět. Vždy je dobré se zamyslet, zda takové řešení potřebuji, co mi poskytne a jestli dostatečně

využiji jeho potenciál. Naimplementovat Framework, z kterého je využita pouze malá výseč jeho funkcionality je plýtvání prostředky. (25)

V dnešní době již existuje velké množství CSS frameworků. Aby bylo možné vypsát jejich základní vlastnosti, bude se práce zabývat pouze nejčastějšími z nich.

3.3.1 Foundation

Je responzivní CSS framework od společnosti Zurb. Obsahuje mřížkový systém, který se umí přizpůsobit různým zařízením. Využívá přístupu mobile-first a netrpí častým jevem, kterým je viditelnost použitého frameworku na první pohled. Jeho přítomnost není na webech, které ho používají, znát. Umožňuje použití responzivní mřížky, která obsahuje 12 sloupců. Je také možné využít změny obsahu pro jednotlivá zařízení. Díky tomu je možné doručit na mobilní zařízení jiné méně datově náročné obrázky. Pro rychlý začátek je možné si vybrat z některých základních šablon. O jeho dobrém jméně vypovídají velcí hráči, kteří jej využívají. Je možné jmenovat například Ebay, Yahoo!, National Geographic a mnoho dalších. (23) (32)

3.3.2 Skeleton

Jedná se o minimalistický framework, který je přizpůsobený mobilním telefonům, ale je možné ho využít i na větších zařízeních. Základní myšlenkou Skeletonu je být minimalistický a stavěný pro mobilní zařízení. Celý framework je pouze na 400 řádků kódu. I design je v duchu minimalizace s všudypřítomným nádechem méně je někdy více. Vychází z mřížky o 12 sloupcích s šířkou 960 pixelů. Na menších rozlišeních je možné některé prvky odstraňovat. Skeleton je vhodný pro tvorbu stránek zaměřených na mobilní zařízení. Nabízí předdefinované stylování základních HTML prvků webové stránky. Může být také dobrým základem pro webdesignery, protože je v základě velmi minimalistický a jednoduchý. Naučit se s ním pracovat lze velmi rychle. (23) (33)

3.3.3 Bootstrap

Je jedním z nejpopulárnější open source frontend frameworků, který se používá pro vývoj responzivních webových stránek. Byl vyvinut designery a vývojáři společnosti Twitter známý pod jménem Twitter Blueprint. Jedná se o mocný frontendový nástroj pro snadnější a rychlejší nasazení responzivity. Framework Bootstrap řeší mnoho věcí za uživatele. Nastavuje za nás například i velikost písma. Oproti Skeletonu nabízí mnohem širší spektrum funkčnosti. Je založen na 12 sloupcové mřížce s předpřipravenými body zlomu. Ty jsou děleny podle druhů zařízení viz následující tabulka. (23)

Pro zařízení	Velmi malá	Malá	Střední	Velká	Velmi velká
Šíře okna	< 576px	≥ 576px	≥ 768px	≥ 992px	≥ 1200px
Třída	.col-xs	.col-sm-	.col-md-	.col-lg-	.col-xl-

Tab. 2 Body zlomu a jejich třídy dle šířky obrazovky.

Zdroj: Vlastní zpracování, data (34)

Následující tabulka zobrazuje rozvržení layoutu na stránce a některé možné kombinace jeho použití. Zde se jedná o rozvržení pro střední rozlišení. V praxi je možné používat u jednoho sloupce nastavení pro různá rozlišení. Například použití .col-md-3 a ve stejném elementu také .col-sm-6. Díky tomu se na menších rozlišeních zobrazí pouze dva sloupce vedle sebe, při vyšších rozlišeních se vedle sebe zobrazí čtyři.

.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-3			.col-md-3			.col-md-3			.col-md-3		
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					
.col-md-4				.col-md-8							
.col-md-12											

Tab. 3 Grid a jeho dělení v různých kombinacích.

Zdroj: Vlastní zpracování

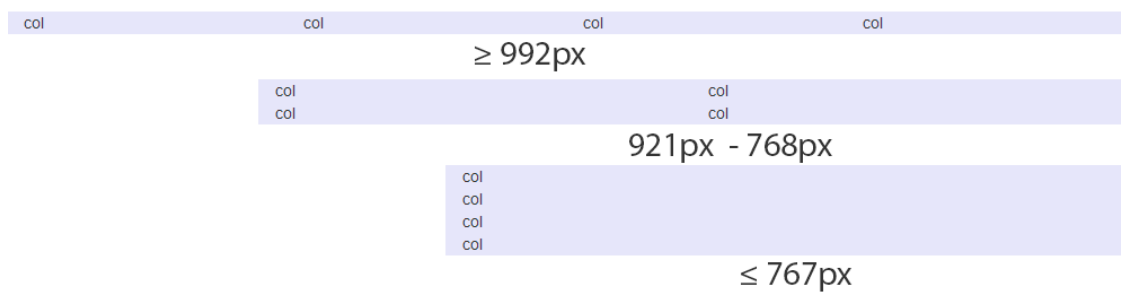
Ukázka kódu 3 zobrazuje použití dvou rozvržení v jednom elementu. Výsledek tohoto zdrojového kódu je vyobrazen na obrázku číslo 5.

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

Kód 3. Příklad rozvržení pro malé a střední obrazovky

Zdroj: Autor

Následující obrázek znázorňuje přeskupení řádků při různých rozlišeních obrazovky. Z toho lze odvodit, že při vyšších rozlišeních než 992 px se uživateli zobrazí rozvržení o čtyřech sloupcích.



Obr. 5 Zobrazení řádku při různých rozlišeních viz kód 3.

Zdroj: Vlastní zpracování

4 Analýza stávajícího stavu webu

Tato kapitola je první kapitolou praktické části práce a popisuje postup analýzy stávající webové stránky Beruška Zdravíčko před tvorbou šablony pro redakční systém. Stránka se nachází na adrese <http://www.masaze-zdravavyziva.cz>. V praktické části jsou využívány poznatky z části teoretické. Původní webové stránky jsou přílohou této práce.

4.1 Cíl praktické části

Cílem praktické části je převést kompletní stávající statický web na redakční systém WordPress. Zachována bude jak funkčnost, tak i stávající web design, který je tváří společnosti. Zároveň bude vytvořena i druhá šablona v jiném modernějším pojetí. Důvodem k migraci na redakční systém je špatná udržitelnost webu ve stávající podobě. Při jakékoliv změně na webu musí vlastník kontaktovat administrátora. Je také nutné optimalizovat stránky pro mobilní telefony. Hlavní motivací ke změně je vlastníková snaha o samostatnost, možnost častější obměny webu a tím i snížení nákladů na provoz.

4.2 Stávající stav webu

Stávající webová stránka je z velké části stránkou statickou. Je psána ve skriptovacím jazyce PHP, ale z něho využívá pouze funkci *include* pro vložení opakujících se částí do stránky a také přepis URL adresy pro uživatelsky přívětivější odkazy. Web není nijak optimalizován pro zobrazení na mobilních zařízeních. Na desktopových prohlížečích je web uzpůsobený pro rozlišení 1024x768 pixelů a vyšší. Při nižších rozlišeních dochází k nutnosti používat horizontální posuvník, což je pro uživatele velice nekomfortní. Při vyšších, dnes typicky používaných širokoúhlých rozlišeních, dochází k centrování stránky a velkým prázdným místům na okrajích. To je dáno použitým layoutem webu se zarovnáním na střed. Pokud bychom chtěli velikost webu měnit v závislosti na rozlišení monitoru, musel by být použit pružný layout.

Na wireframu níže jsou viditelná velmi výrazná prázdná místa. Obsah je uživateli zobrazen pouze v omezené míře. Čím má uživatel vyšší rozlišení, tím více prázdného obsahu se mu po okrajích stránky zobrazí.



Obr. 6 Wireframe úvodní stránky webu při FullHD rozlišení

Zdroj: (27) (35) zpracování autor

Na následujícím kódu je stylován hlavní div, ve kterém se zobrazuje obsah. Šířka tohoto kontejneru je neměnná. Ve všech rozlišeních je napevno nastavena na 820 px. Tím vzniká problém při zobrazování na nižších rozlišeních. Prohlížení je nekomfortní s nutností vertikálního posuvu. Opačným problémem mohou být vysoká rozlišení, kde je obsah vycentrovaný a zabírá pouze malou část okna.

```
#obsah {  
  width: 820px;  
  float: left;  
  min-height: 500px;  
  padding: 25px 50px 50px 50px;  
}
```

Kód 4. Stylování hlavního kontejneru, ve kterém je zobrazován obsah

Zdroj: Autor

Původní webová stránka je stylována do teplých barev. Menu se nachází v horní části obrazovky a má obsáhle členění. Žádná ze stránek neobsahuje podstránku. Nově příchozí návštěvník může v touze po rychlém nalezení informací stránky opustit. Na hlavní stránce se nachází upozornění na nadcházející akce, novinky a popis jednotlivých záložek menu. Součástí hlavního obrázku jsou i náhodně zobrazovaná mota, která byla v původním konceptu zadávána ručně do pole v PHP. Zobrazení mot na úvodní stránce bude zachováno, ale mota budou zadávána pomocí rozhraní přímo do databáze.



Obr. 7 Vzhled úvodní webové stránky při rozlišení 1280 x 1024

Zdroj: (36)

Web využívá uživatelsky přívětivé odkazy, které web zpřehlední a jsou i vhodné pro SEO optimalizaci. Je toho docíleno pomocí RewriteRule v soboru `.htaccess`. Každá stránka má svůj vlastní PHP soubor, ve kterém je pouze formátovaný text a ten se poté zobrazuje do kostry stránky, kterou tvoří soubor `index.php`.

Název souboru	Velikost souboru	Typ souboru
..		
fonty		Složka souborů
highslide		Složka souborů
img		Složka souborů
include		Složka souborů
obrazky		Složka souborů
.htaccess	644	Soubor HTACCESS
cenik.php	2 053	Soubor PHP
clanky.php	1 700	Soubor PHP
detoxikace.php	5 007	Soubor PHP
index.php	1 079	Soubor PHP
konzultace.php	435	Soubor PHP
kostelec-nad-orlici.php	1 936	Soubor PHP
masaze-kostelec-nad-orlici.php	1 671	Soubor PHP
sitemap.xml	1 186	Dokument ve formátu XML
tarotove-karty.php	604	Soubor PHP
uvod.php	3 782	Soubor PHP
vedome-prozivani.php	665	Soubor PHP
vyrobky.php	1 714	Soubor PHP

Obr. 8 Členění souborové struktury

Zdroj: Autor

4.2.1 Google Analytics

Web využívá pro sledování návštěvnosti a chování uživatelů službu Google Analytics. Jedná se o velice mocný statistický nástroj, díky kterému je možné vidět aktuální i historickou návštěvnost, sledovat tok návštěvníků, jejich chování a také údaje o koncovém zařízení, z kterého návštěvníci přistupují na webovou stránku. Hlavním atributem, který je vhodné sledovat je nejčastěji používané rozlišení obrazovky, to je důležité z hlediska optimalizace zobrazení.

Služba ve stránce pracuje pomocí vloženého JavaScriptu. Na každé stránce, na které chceme sbírat údaje o návštěvnících, musí být vložen kód z ukázky číslo 5. Kód je stejný pro všechny stránky, které chceme mít spravované pod jedním účtem. Samotný kód je univerzální. Mění se pouze jeho jedinečný identifikátor, který je vytvořen při založení účtu v Google Analytics a určuje, pod který účet se budou sbíraná data ukládat.

```

<script type="text/javascript">
  var _gaq = _gaq || [];
  _gaq.push(['_setAccount', 'Jedinečný identifikátor']);
  _gaq.push(['_trackPageview']);
  (function() {
    var ga = document.createElement('script'); ga.type='text/javascript';
    ga.async = true;
    ga.src = ('https:' == document.location.protocol ? 'https://ssl' :
    'http://www') + '.google-analytics.com/ga.js';
    var s = document.getElementsByTagName('script')[0];
    s.parentNode.insertBefore(ga, s);
  })();
</script>

```

Kód 5. Script sbírající data o návštěvnících do služby Google Analytics

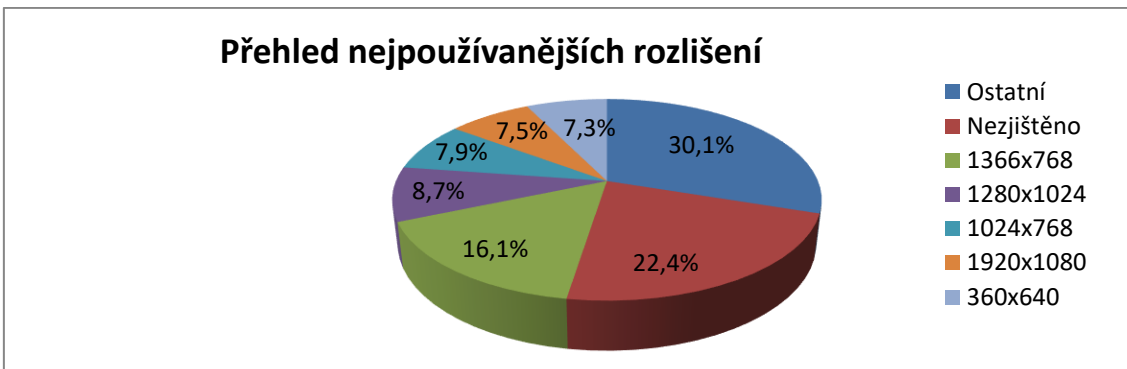
Zdroj: (37)

4.2.2 Sledované parametry

Pro sledování byly vybrány následující klíčové indicie. Pomocí exportů ze služby Google Analytics bude výsledný web optimalizován podle parametrů nejpočetnějších skupin návštěvníků. Sledovaná data jsou použita za celý kalendářní rok v období od 7. února 2016. Za sledované období navštívilo stránky 1127 návštěvníků. Export je obsažen jako součást přílohy na CD.

Sledované parametry vyobrazené na následujících grafech:

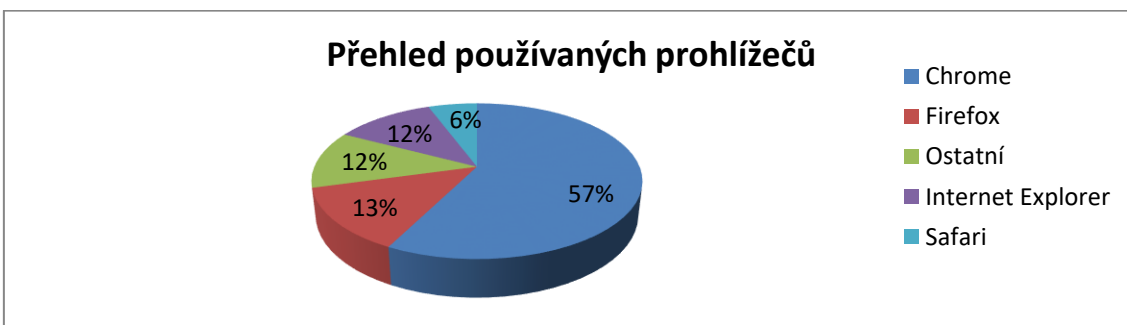
- rozlišení obrazovky,
- prohlížeč internetu,
- koncové zařízení uživatele.



Graf 3. Nepoužívanější rozlišení obrazovky

Zdroj: Vlastní zpracování, data vycházejí z přílohy Google Analytics

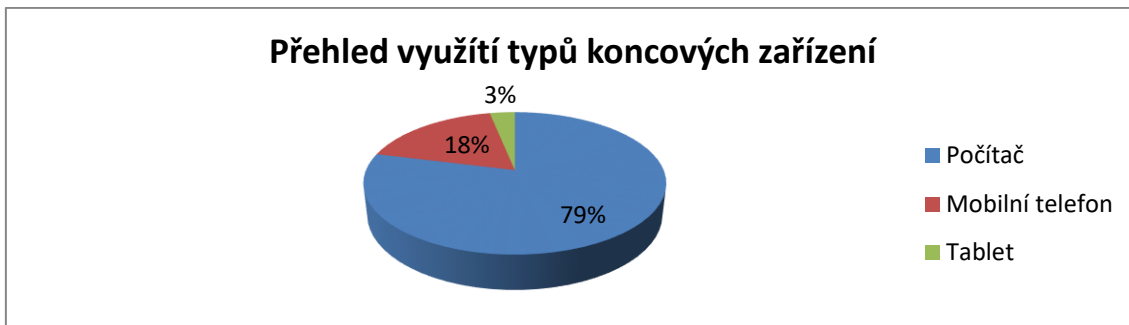
Graf č. 4 ilustruje přehled nepoužívanějších prohlížečů návštěvníků webové stránky.



Graf 4. Nepoužívanější webový prohlížeč

Zdroj: Vlastní zpracování, data vycházejí z přílohy Google Analytics

Na následujícím grafu je znázorněno zastoupení používaných koncových zařízení. Nejčastěji používají zákazníci pro zobrazení stránek počítače.



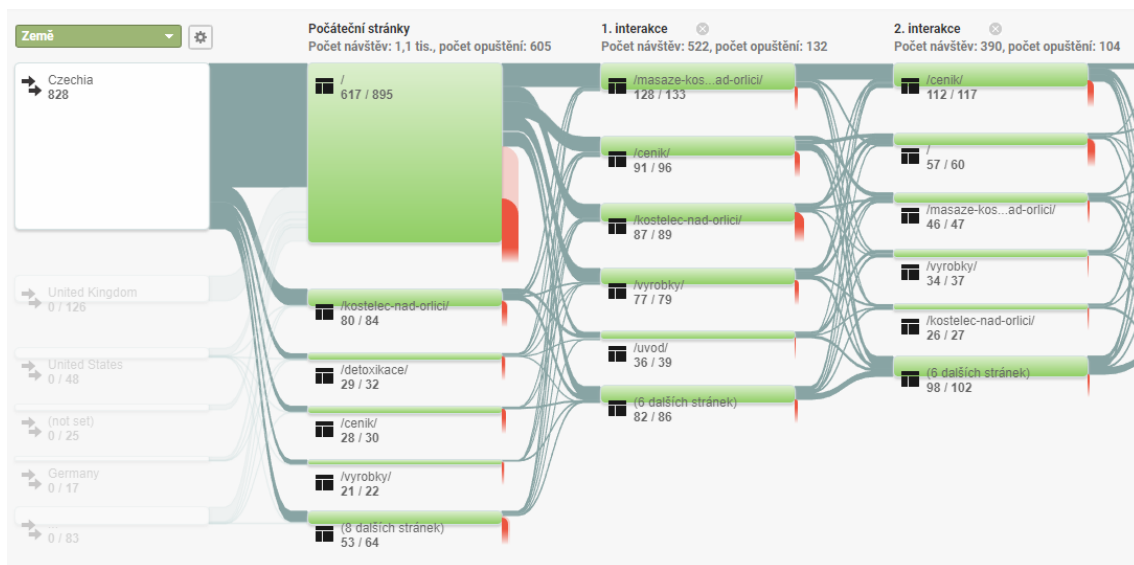
Graf 5. Koncová zařízení, přes které uživatel přistupuje na web

Zdroj: Vlastní zpracování, data vycházejí z přílohy Google Analytics

Z přehledů jasně vyplývá, že není možné zanedbat návštěvníky používající mobilní zařízení, kterých je 21 %. Z důvodu četnosti rozlišení je nutné volit responzivní design, který je v dnešní době nutnou součástí moderní webové prezentace. Vyhledávač Google dokonce upřednostňuje optimalizované mobilní weby ve vyhledávání. Nejdůležitější optimalizace by měla být věnována prohlížeči Google Chrome.

Tok návštěvníků

Stránky za rok navštívilo celkem 895 uživatelů z České republiky. Největší počet návštěvníků přichází na hlavní stránku, na které by měl být stručný a přehledný souhrn základních informací. Zároveň by tato stránka měla uživatele zaujmout. Na stávajícím webu 53,6 % návštěvníků z úvodní stránky web ihned opustí. Z úvodní stránky uživatelé v největší míře navštěvují stránku s rozpisem masáží a z ní přechází na ceník. Velmi často navštěvovanou stránkou je také kontakt. Úkolem šablony je také stylizovat a zpřehlednit obsah pro uživatele, aby jeho orientace na stránce byla co nejjednodušší.



Obr. 9 Přehled toku návštěvníků na webu a jejich první dvě interakce

Zdroj: Vlastní zpracování, data vycházejí z (37)

5 Realizace webové prezentace

Kapitola se zabývá návrhem nové webové prezentace, který vychází z poznatků stávajícího webu analyzovaného v předchozí kapitole.

5.1 Instalace CMS

Než začneme samotnou instalaci redakčního systému je potřeba si uvědomit, kam ho budeme nasazovat. Při nasazení do stávajícího provozu musíme brát v potaz tento fakt a rozhodnout se kam systém nasadit, aby nebyl omezen přístup návštěvníků. V dnešní době je takřka nepřijatelný stav, který býval dříve zvykem, to jest vystavit na webové stránky oznámení, že jsou stránky v rekonstrukci a odstavit přístup na web uživatelům. Používané jsou dva způsoby přípravy webu bez dlouhodobého omezení provozu. Web připravíme bokem a po jeho dokončení provedeme v nejméně vytíženém čase migraci.

První způsob, který byl zvolen, je vytvářet web na subdoméně původní webové stránky. Tento způsob lze použít pouze pokud máme přístupová práva k danému hostingu. Výhodou je, že nově budovaný projekt běží přímo na serveru, na kterém bude poté nasazen. Nevýhodou je, že vývoj je pomalejší vše se ukládá přes internet na server.

Druhým způsobem je příprava na tzv. localhostu. Vytvoříme si webový server na svém počítači. Tento web je pro internet plně skrytý a všechna nastavení serveru si řídíme na svém počítači. Nevýhodou může být, že nepoužíváme totožnou konfiguraci jako na serveru v hostingu a po migraci webu se mohou vyskytnout chyby.

Samotná instalace začíná stažením WordPressu z oficiálních stránek. Instalace je dodávána zabalená v archivu, jehož obsah rozbalíme na webový server. Po nahrání souborů přistoupíme na webovou adresu, kde je web umístěný. Instalací nás už krok za krokem provede průvodce. (38)

Nejdůležitější částí je zadání správných přístupových údajů k databázi. WordPress využívá pro ukládání dat databázi, ze které potom data načítá a zobrazuje. Pro instalaci je nutné mít vytvořenou databázi, znát její název, adresu serveru, na kterém databáze běží a přístupové údaje do ní. Jsou podporovány databáze MySQL nebo MariaDB. Aktuální podporovanou verzi si lze ověřit na <https://wordpress.org/about/requirements/>.

Pro instalaci budete muset znát následující údaje:

- název databáze – název vytvořené databáze, do které chcete databázi přidat,
- uživatelské jméno – účet s přístupovým oprávněním k databázi,
- heslo – heslo k uživatelskému účtu,
- databázový server – název databázového serveru,
- předpona tabulek – názvy tabulek v databázi vytvoří s předepsanou příponou.

Zadejte prosím následující informace o připojení k databázi. Pokud uvedené údaje neznáte nebo něčemu nerozumíte, kontaktujte poskytovatele hostingu.

Název databáze	<input type="text" value="wordpress"/>	Název databáze, kam chcete instalovat WordPress.
Uživatelské jméno	<input type="text" value="uzivatelske-jmeno"/>	Uživatelské jméno k databázi.
Heslo	<input type="text" value="heslo"/>	Heslo pro přístup k databázi.
Databázový server	<input type="text" value="localhost"/>	Pokud nebude fungovat výchozí hodnota localhost, tak musíte použít informace od poskytovatele hostingu.
Předpona tabulek	<input type="text" value="wp_"/>	Pokud chcete provozovat několik různých instalací WordPressu pod jednou databází, tak musíte zvolit různé předpony tabulek.

Obr. 10 Instalační průvodce nastavení přístupu k databázi

Zdroj: Vlastní zpracování, instalátor WordPressu

Jestliže bylo vše zadáno v pořádku a WordPressu se podařilo spojit s databází, je nutné zadat ještě nové přihlašovací údaje pro správu webu. Tím se vytvoří uživatelský účet s administrátorskými oprávněními.

Po dokončení instalace je možné se hned přihlásit na adrese <http://doména/wp-login> a spravovat web.

Je vhodné provést aktualizace pluginů a šablon. Pokud bychom měli již hotovou šablonu nebo si vybrali z výchozích, bylo by možné začít web ihned plnit obsahem.

5.2 Příprava HTML šablony

V případě vlastního návrhu šablony bychom nejprve vytvořili vzorovou šablonu pouze za pomoci HTML a kaskádových stylů a poté tuto šablonu předělávali na redakční systém. Pro začínající uživatele je tato možnost jednodušší na debugování. Snadněji se hledá chyba v HTML stránce než v šabloně redakčního systému, kde je mnohem více návazností mezi jednotlivými soubory.

Pokud si chceme tvorbu šablony usnadnit a zaujme nás některá volně použitelná HTML šablona, je možné si tím usnadnit práci a nevytvářet kompletní šablonu na zelené louce. Stejně tak tomu bylo v autorově případě. Rozhodl se vycházet z demo Bootstrap šablony Agency, která je volně ke stažení pod MIT licencí umožňující její další použití a úpravy. Jedná se pouze o HTML a CSS kostru, která bude v dalších částech předělána do formy použitelné v redakčním systému. Šablona je součástí přílohy této práce.

5.2.1 Rozdělení webové prezentace

Bylo zvoleno rozdělení webu podle jiné struktury, než byl původní koncept. V menu bude umístěno pouze několik nejdůležitějších kategorií. Je zbytečné přehlcovat uživatele položkami. Web bude více propagovat osobní přístup, nebude zaměřený pouze na jméno společnosti, ale na konkrétní osobu, která veškeré služby zajišťuje. Hlavní stránka bude rozložena do tří hlavních sekcí.

Hlavní část zobrazená uživateli při návštěvě bude cílena na fotografii přes celou obrazovku a moto dne. Fotografie bude ve všech rozlišení zobrazena vždy přes celou obrazovku zařízení.

Druhá část stránky bude zaměřena na novinky, aktuality a oznámení. Tyto novinky budou zobrazeny maximálně čtyři a bude se jednat o stručný popis, který nebude dále nikam odkazovat.

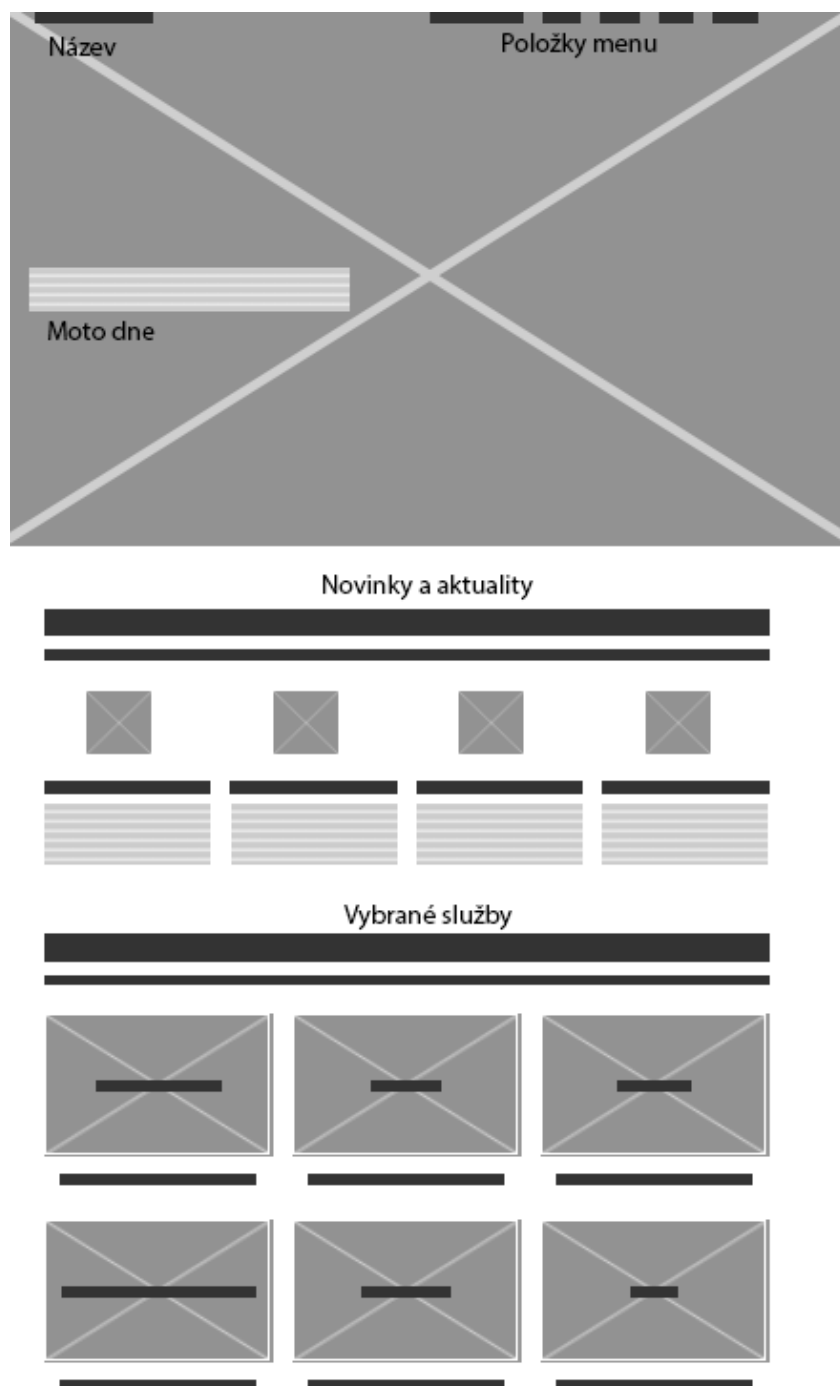
V třetím oddílu budou umístěny služby. Každá služba bude identifikována vlastním výstižným obrázkem a názvem služby. Zde již bude propojení a každá služba bude odkazovat na vlastní stránku s jejím popisem. Na hlavní straně bude zobrazeno vybraných šest služeb s možností jejich řazení.

```
<header> Hlavní část s fotografií přes celou obrazovku </header>  
<section id="news"> Novinky a aktuality </section>  
<section id="services"> Služby </section>
```

Kód 6. Zjednodušené rozdělení hlavních elementů stránky

Zdroj: Autor

Následující obrázek znázorňuje rozvržení hlavní stránky webu, skládajícího se ze třech hlavních částí vyobrazených na následujícím obrázku. Nic více nebude na úvodní stránce zobrazováno.



Obr. 11 Rozvržení elementů hlavní stránky

Zdroj: (35) zpracování autor

V prvním stádiu přípravy došlo k rozvržení elementů v původní Agency šabloně podle Wireframe na obrázku 11. Byl přidán sloupec v kontejneru novinek a byla změněna pravidla pro vykreslování úvodního obrázku. Došlo k odebrání dalších sekcí na úvodní stránce.

5.2.2 Očištění šablony

Velmi důležitým krokem bylo očištění šablony ve všech rovinách, jak v HTML, CSS tak i připojeném JavaScriptu od přebytečného kódu, který nebyl využíván. Pro co nejrychlejší vykreslení a zjednodušení byly odstraněny všechny připojené JavaScriptové knihovny. Celkem se jednalo o 7 knihoven o celkové velikosti 196 kB a 201 783 znacích. To vše by musel prohlížeč načíst do paměti a zpracovat zcela zbytečně. Veškerá JavaScriptová funkčnost byla vnořena do těla stránky a obsahuje pouhých 1 976 znaků. Jediná funkce, co potřebovala JavaScript bylo responzivní menu. Využívat pouze kvůli menu jQuery a jiné podpůrné knihovny by bylo naprosto zbytečné, a proto byla jejich funkce nahrazena čistým JavaScriptem.

Po dokončení a připravení HTML šablony následuje její převedení do redakčního systému WordPress.

5.3 Tvorba šablony pro redakční systém

Prvním krokem k vytvoření šablony je vytvoření adresáře v adresářové struktuře WordPressu. Šablony se nacházejí v adresáři wp-content/themes. Po zvolení vhodného názvu pro šablonu je doporučeno složku pojmenovat obdobným názvem, který pak použijí ve vlastnostech šablony.

Pro základní fungování šablony jsou v jejím adresáři nutné pouze dva základní soubory:

- index.php,
- style.css.

Ze souboru kaskádových stylů se generuje název a podrobnosti o šabloně. Tento soubor slouží k účelům inicializace šablony, lze jej také využít ke stylování stránek. Autor se přiklání k řešení tyto soubory nemíchat.

```
/*
Theme Name: Agency
Theme URI: http://www.ondrejmlajnek.cz
Author: Ondřej Mlejnek
Author URI: http://www.ondrejmlajnek.cz
Description: Je šablona psaná přímo na míru webu masaze-
zdravavyziva.cz.
Version: 1.0
License: for masaze-zdravavyziva.cz
License URI: http://www.gnu.org/licenses/gpl-2.0.html
Tags: menu, editor-style, featured-images, microformats, post-
formats
Text Domain: masaze-zdravavyziva.cz
*/
```

Kód 7. Soubor style.css určuje název a další informace o šabloně

Zdroj: Autor

Soubor index.php je hlavním souborem, dle kterého se vykreslují stránky redakčního systému. V tomto souboru je řešeno vyčítání dat z databáze a jejich prezentace uživateli.

Šablony typicky používají pro stránky rozdělení souborů na logické oblasti. Soubor index.php může být poté složen z:

- header.php – hlavička stránky,
- footer.php – patička stránky.

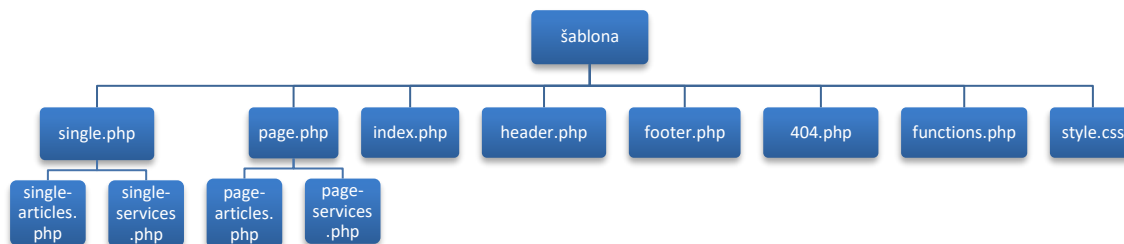
Výhodou je oddělení opakujících se částí od obsahu. Tyto soubory se poté linkují do stránky. Díky tomu není potřeba pro každou stránku, která potřebuje vlastní soubor šablony vytvářet hlavičku a patičku. Ta bývá většinou ve všech souborech stejná.

```
<?php get_header(); ?> // hlavička stránky
OBSAH SOUBORU index.php // obsah stránky
<?php get_footer(); ?> // patička stránky
```

Kód 8. Zjednodušené uspořádání souboru index.php

Zdroj: Autor

Na následujícím obrázku je rozkreslena souborová struktura hotové šablony.



Obr. 12 Struktura souborů šablony

Zdroj: Vlastní zpracování

5.3.1 Hlavička stránky

Soubor: header.php

Je důležitou součástí šablony, její obsah je možné psát ručně do všech požadovaných částí. Vhodnější je použít ji jako samostatný soubor. V hlavičce jsou seskupena data uvozující stránku. Jedná se především o HTML tagy. Typicky se používá pro následující úkony:

- linkování CSS a JavaScriptu,
- uvozování HTML tagů,
- nastavení metadat,
- nastavení titulku stránky,
- zobrazení menu.

Funkce pro zobrazení menu viz ukázka kódu číslo 9 najde primární menu šablony a použije pro něj třídu *nav navbar-nav navbar-right*. Menu se v defaultním nastavení vytváří pomocí odrážkového seznamu ``, kde každá jeho položka je ``.

```
<?php
wp_nav_menu(array('menu_class' => 'nav navbar-nav navbar-right',
'theme_location' => 'primary' ));
?>
```

Kód 9. Slouží k zobrazení menu

Zdroj: Autor

5.3.2 Patička stránky

Soubor: footer.php

Její obsah je podobně jako u hlavičky možné psát ručně do všech potřebných souborů, ale vhodnější je linkování souboru viz kód 8. Patička uzavírá HTML elementy stránky a je to koncová část šablony stránky. Patička se používá k následujícím úkonům:

- ukončuje otevřené HTML tagy,
- připojení koncových JavaScriptů,
- zobrazení informací ve spodní části stránky.

5.3.3 Výpis obsahu hlavní stránky

Soubor: index.php

Jedná se o hlavní a nejdůležitější soubor celé šablony. Pokud šablona nemá jiné definované soubory pro vykreslení obsahu konkrétních stránek, je vždy použit jako výchozí index.php. Proto je tento soubor povinný a bez něho šablona nebude vůbec fungovat. Primární použití je pro zobrazení obsahu hlavní stránky webu. Úkolem souboru index.php je zprostředkovat následující funkce:

- zobrazení obsahu úvodní stránky,
- používá se jako výchozí kostra pro zobrazení všech stránek webu.

Pro vyčítání dat z databáze využívá WordPress funkci *WP_Query* jejíž argumentem je pole parametrů, které definují, k jakým datům budeme v databázi přistupovat. Strukturu zápisu lze vidět v následujícím kódu. Pomocí parametrů v proměnné

`$args` nastavíme následující parametry. Vybírej z `post_typu` mota, seřad' je náhodně a omez výpis pouze na jedno. Výsledek tohoto dotazu poté pomocí cyklu `while` vypíšeme na obrazovku. O samotné propojení s databází se už postará jádro WordPressu. Obdobným stylem provádíme všechny dotazy na databázi.

```
<?php $args = array( 'post_type' => 'mota', 'orderby' => 'rand',
'posts_per_page' => '1' ); //výber náhodně , pouze jedno moto
$loop = new WP_Query( $args );

while ( $loop->have_posts() ) : $loop->the_post();
    the_title(); //vypíše text mota
    echo get_field('mota_autor'); //vypíše autora mota
endwhile; ?>
```

Kód 10. Výpis náhodného mota z databáze

Zdroj: Autor

5.3.4 Používání funkcí v šabloně

Soubor: `functions.php`

Je místo v šabloně, kde můžeme přidávat nové funkce a vlastnosti tématu. Z důvodu, že není vhodné zasahovat do jádra samotného WordPressu existuje soubor `functions`, který přidává nové možnosti bez zásahu do výchozího zdrojového kódu. Zásahy do jádra nejsou vhodné jak z hlediska bezpečnosti, mohli bychom vytvořit bezpečnostní díru, tak z hlediska aktualizací. Aktualizace přepíše zdrojové soubory a o všechny změny bychom tím přišli. Tento soubor je využit k následujícím úkonům:

- zápis nových funkcí do šablony,
- registrace součástí,
- úprava a nastavení parametrů.

V tomto souboru shromažd'ujeme veškeré důležité funkce. Jedná se o aplikační část šablony. Pro téma popisované v této práci byl soubor `functions` použit pro následující funkcionality.

Registrace menu

Vytvoření hlavního menu pro danou šablonu, toto menu je poté v administraci webu nabízeno pro přidání nových položek. První parametr funkce určuje identifikátor menu, druhý jeho popis v šabloně.

```
function register_my_menu() {
    register_nav_menu( 'primary', 'Hlavní menu' );
}
add_action( 'init', 'register_my_menu' );
```

Kód 11. Registrace menu pro šablonu

Zdroj: Autor

Přidání shortcode

Shortcode je užitečná věc, která velice usnadní práci obyčejnému uživateli. Vytvoří se funkce, kterou zaregistrujeme jako shortcode. Poté můžeme funkci vyvolat ve WYSIWYG editoru při úpravě libovolné stránky zadáním *[název shortcode]*. Tato funkcionality byla v práci využita pro zobrazení kontaktního formuláře, pro jehož tvorbu autor nechtěl využít pluginu. Formulář pro odesílání emailu musí být ošetřený v mnoha ohledech na validitu dat. Proto byl formulář vytvořen v soboru functions a pro jeho zobrazení je vytvořen shortcode, který se vloží jako HTML znak do stránky a zbylá část kontaktní stránky může být upravována uživatelem. Prvním parametrem funkce je název, pod kterým bude shortcode zobrazen. Druhým parametrem je funkce, kterou bude shortcode volat.

```
add_shortcode( 'email', 'formContacs' ); //registrace shortcode
function formContacs() { //funkce, kterou volá shortcode
    Tělo volané funkce
}
//Část HTML z těla stránky pro kontakt
//[email] zajistí spuštění obsahu funkce formContacs
<div id="email" class="col-lg-6">[email]</div>
```

Kód 12. Vytvoření a použití shortcode

Zdroj: Autor

Úprava administrátorského menu

Nechceme uživateli zobrazovat položky v menu, které nepotřebuje vidět. Chceme redaktorům co nejvíce zjednodušit práci s redakčním systémem. Na co si uživatel neklikne, jako by neexistovalo. Funkce pro odstranění položek administrátorského menu pouze odebírá položky menu. Při zadání odkazu je stránka nadále funkční. Tuto funkci nelze dostatečně použít pro odebrání přístupu uživateli ke konkrétním položkám. Pro vizuální odstranění přebytečných položek však poslouží dobře. Ve funkci pouze jako parametr předáváme název stránky.

```
function remove_menus(){
    remove_menu_page( 'edit-comments.php' );
    remove_menu_page( 'index.php' );
    remove_menu_page( 'edit.php' );
}
add_action( 'admin_menu', 'remove_menus' );
```

Kód 13. Odstranění položek menu

Zdroj: Autor

Tvorba miniatur obrázku

Při každém uploadu obrázku do redakčního systému jsou vytvářeny zmenšeniny originálu. Ve výchozím stavu jsou vytvářeny následující velikosti v pixelech:

- miniatura 150 x 150,
- střední 300 x 300,
- velký 640 x 640,
- originál.

U velikostí větších než miniatura je zachován poměr stran a větší strana má maximální rozměr.

Následující kód vytváří nový ořezávací vzor pro miniatury. Každý nově nahraný obrázek bude mít vytvořenou kopii podle těchto parametrů. Může se stát, že si uvědomíme potřebu vytvářet miniaturu v jiném rozměru, než jsou stávající a máme nahrané už velké množství obrázků. Vzor velikosti obrázků se aplikuje pouze na nově nahrané od doby jeho přidání. V tomto případě můžeme pro aktivování znovu

ořezání využít plugin *Regenerate Thumbnails*, který znovu vyvolá proces vytváření miniatur obrázků.

```
add_image_size( 'crop-services', 400, 250, array( 'left', 'top' ) );  
// vytvoří vzor miniatur obrázku v dané velikosti s oříznutím vlevo  
odshora
```

Kód 14. Vytváření šablony pro nové miniatury

Zdroj: Autor

Přidání Custom Post Type

Post Types jsou velice mocnou zbraní WordPressu, díky které je možné vytvořit takřka libovolný druh obsahu. Základní výchozí typ obsahu je stránka, příspěvek, příloha, my však můžeme díky vlastním typům příspěvků tuto škálu rozšířit o libovolný typ obsahu. Chceme zadávat typy příspěvku položka e-shopu, není problém, zaregistrujeme post type a vkládáme do redakčního systému zboží.

Pro snadnější tvorbu a úpravy vlastních polí autor doporučuje nainstalovat plugin *Advance Custom Fields* zkráceně *ACF*. Po instalaci pluginu je již správa a tvorba nových typů příspěvku velice jednoduchá. Vytvoření nového druhu příspěvku provedeme následujícím kódem, který registruje typ příspěvku služba. Bylo použito pouze pár následujících parametrů.

```
add_action( 'init', 'create_post_type' );  
function create_post_type() {  
    register_post_type( 'services', //název pro databázi  
        array(  
            'labels' => array(  
                'name' => ( 'Služby' ), //název druhu příspěvků, bude v menu  
            ),  
            'public' => true, //nastavení viditelnosti příspěvku  
            'has_archive' => false, //bude možné archivovat  
            'menu_icon' => 'dashicons-clipboard', //ikona v menu  
        )  
    );  
}
```

Kód 15. Registrace nového typu příspěvku

Zdroj: Autor

V následujícím obrázku je možné vidět vytváření parametrů vlastního příspěvku v pluginu ACF. V tomto případě se jedná o služby. Je nutné, aby se v pravidlech typ příspěvku rovnal typu vytvořeném v souboru funkcí. Poté je možné vytvořit libovolný počet polí pro daný typ příspěvků. Každé toto vlastní pole je možné parametrizovat. Výpis důležitých parametrů vlastního pole:

- název pole – název zobrazený uživateli,
- jméno pole – jméno, pod kterým budeme toto pole vypisovat v šabloně,
- typ pole – volíme text, obrázek, WYSIWYG editor, přepínače a mnoho dalšího,
- instrukce pole – nápověda pro uživatele, je zobrazena při vyplňování pole,
- požadováno – povinnost vyplnění pole,
- další parametry – liší se podle typu použitého pole.

Upravit skupinu polí [Přidat novou](#)

Služby

Jméno pole, podle kterého pak v dotazech vypisují obsah

Pořadí pole	Název pole	Jméno pole	Typ pole
1	Obrázek	services_img	Obrázek
2	Zobrazit na hlavní straně	services_view	Přepínací tlačítko
3	Stručná charakteristika	services_small_text	Text
4	Popis	services_text	Wysiwyg Editor
5	Cena	services_price	Číslo

Chytněte a táhněte pro změnu pořadí [+ Přidat pole](#)

Umístění

Typ příspěvku se musí rovnat vytvořenému post typu ve functions.php

Pravidla

Vytváří sadu pravidel pro určení, na kterých stránkách úprav budou použita tato vlastní pole

Zobrazit tuto skupinu pole pokud

Typ příspěvku je rovno services a

nebo

[Přidat skupinu pravidla](#)

Obr. 13 Tvorba vlastního příspěvku v pluginu ACF

Zdroj: Vlastní zpracování

Díky použití ACF a registraci typu příspěvku máme nyní v menu položku Služby. Práce s položkami typu Služby je stejná jako například se stránkami, jen s tím rozdílem, že položka typu Služby má jiná vlastní pole, která jsme definovali pomocí ACF. Uživatel vytváří vyplněním požadovaných parametrů novou službu. O její zobrazení a styl výpisu se již starají definice v šabloně.

5.3.5 Tvorba zobrazení stránek vlastních příspěvků

Soubor: `page-nazev_stranky.php`, `single-nazev_stranky.php`

Máme vytvořený vlastní typ příspěvku a přidané příspěvky tohoto typu, ale jak je nyní zobrazit? WordPress umožňuje vytvořit šablonu pro zobrazení konkrétních typů příspěvku. Jedná se o dvě různé stránky pro konkrétní typ zobrazení. Vždy existuje šablona pro výpis všech příspěvků *page* a k ní ještě pro výpis jednoho konkrétního příspěvku *single*.

V šabloně je v základu možné vytvořit soubory `page.php` a `single.php`, podle kterých se zobrazují všechny typy příspěvků včetně článků. Pokud jsme v ACF vytvořili nový typ příspěvku není nutné, aby měl vlastní stránku pro výpis. V naší šabloně je například vytvořen vlastní typ příspěvku pro zobrazení mota dne. Tato mota jsou však zobrazena pouze jako text na úvodní straně webu, proto je zcela zbytečné vytvářet pro ně vlastní stránku v šabloně.

Vlastní stránku v šabloně je potřeba vytvářet v případě, že chceme vlastní typ příspěvku formátovat odlišně než zbytek webu. Například stránka pro zobrazení kontaktu je zobrazena pomocí *page.php*, protože nemá vytvořenou vlastní stránku a veškeré kódování je napsané přímo ve stránce přes administraci. Oproti tomu stránka služby má definovanou vlastní stránku pro výpis. Je nutné zobrazit všechny služby na jedné stránce a také po rozkliknutí dané služby zobrazit informace o ní. Definujeme tedy *page-services.php* i *single-services.php*. Každá vlastní šablona stránky musí být na začátku definovaná názvem a popisem. Definice se provádí pouze v souboru *page-nazev.php*. Soubor *single* je nutné pojmenovat stejně *single-nazev.php*. WordPress díky jmenné konvenci pozná, že se jedná o stránku, která patří k výpisu podle dané šablony. Pro výpis konkrétních služeb poté automaticky

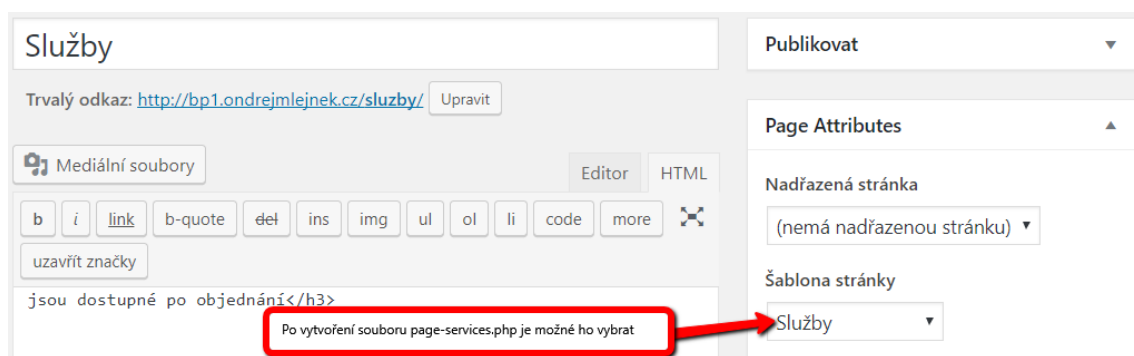
hledá pro zobrazení soubor *single_services.php*. Pokud ho najde formátuje obsah podle něho. Pokud soubor nenalezne pro prezentaci obsahu použije *single.php*.

```
/*  
Template Name: Služby  
Description: This template can use for viewing all services  
*/
```

Kód 16. Definice šablony stránky

Zdroj: Autor

Pokud chceme pro stránku použít konkrétní šablonu musíme tuto skutečnost nastavit v atributech stránky. Ve výchozím stavu se všechny stránky vykreslují podle souboru *page.php* pro výpis všech vlastních typů příspěvku a *single.php* pro konkrétní jeden. Na následujícím obrázku je vidět přiřazení šablony pro konkrétní stránku.



Obr. 14 Přiřazení šablony stránky konkrétní stránce

Zdroj: Vlastní zpracování

5.3.6 Chybová stránka

Soubor: 404.php

Stránka se použije v případě, že jsme odkázaní na stránku, která již neexistuje. Typickým příkladem v praxi je indexace stránek vyhledávačem. Vyhledávací boti procházejí stránky a indexují jejich obsah. Návštěvník ve vyhledávání podle klíčových slov najde naši stránku, ale ta je na serveru již smazaná pouze vyhledávač

to ještě nezaregistroval. Uživatel přistoupí na tuto stránku, WordPress tuto chybu odchytlí a přesměruje uživatele na chybovou stránku 404, která je uložena v šabloně. Pro tvorbu této stránky se nejčastěji použije univerzální text, který chceme uživateli zobrazit, aby měl zpětnou vazbu, že je něco špatně.

6 Migrace webu

Webová stránka byla tvořena za provozu stávajícího webu na subdoméně z důvodu zachování stávajícího provozu. Po kompletním dodělání webu do podoby, kterou je možné publikovat na hlavní doméně, přichází fáze migrace. Na původní subdoméně je vytvořena šablona a celý web naplněn daty. Veškeré použité texty a obrázky je nutné ručně kopírovat do nových stránek přes webové rozhraní. Data je tedy nutné vkládat do WYSIWYG editoru. Tento editor umí přes schránku zachovat i daný styl webové stránky. Při kopírování se formátování neztrácí a zachovají se HTML tagy. Je nutné pamatovat na to, že pokud vkládám externí obsah všechny cesty k tomuto obsahu jsou kopírovány jako cesty k umístění. To může být problém při smazání původního obsahu. Pokud chceme u editoru zachovat všechny funkce, například správné obtékání obrázků textem, je nutné mít v CSS šablony definované výchozí styly, které nalezneme zde <https://codex.wordpress.org/CSS>.

Jakmile máme hotový celý web, jak šablonu, tak obsah, můžeme přistoupit k přesunu ze subdomény. Autor doporučuje vybrat pro přesun čas, kdy má web nejmenší návštěvnost. Přesun se neobejde bez výpadku.

Existují dva způsoby, jak WordPress migrovat na jinou doménu:

1. ruční migrace,
2. automatizovaná migrace.

6.1 Ruční migrace

Problém migrace je, že nelze udělat pouhým přesunutím dat a databáze na novou doménu, jak tomu bývá u přesunu statických webových stránek bez databáze. V databázi totiž existují řádky spjaté s doménou, na které je web umístěn například relativní cesty a podobně. Tyto řádky by po pouhém importu databáze odkazovali do míst, která v daném hostingu nemusí být viditelná. Proto je u ruční migrace nutné tato místa v databázi opravit.

Prvním krokem je export databáze. Je možné ho provést přes správce databáze nejčastěji u hostingů skrze phpMyAdmin. Nyní záleží, jak přistoupíme k vyhledání a přepsání původních cest na nové. Nejjednodušší cestou je funkce najít a nahradit dostupná v textových editorech. Nahrazovat budeme takto: *puvodnidomena.cz* za *novadomena.cz*. Pokud máme v hostingu možnost tvorby více databází autor doporučuje založit novou databázi a do ní provést import. V případě, že tato možnost zde není, je možné vrátit se zpět k záloze, kterou jsme exportovali. Dále je nutné přesunout kompletní souborový systém WordPressu na nový hosting. V případě použití jiné databáze je nutné v root složce WordPressu upravit soubor *wp-config.php* a v něm vyplnit nové přihlašovací údaje pro databázi. Nyní by měl být web z migrovaný a funkční. Mohl by nastat problém při tvorbě pěkných URL adres. Web je funkční, ale při rozkliknutí odkazů píše že stránka neexistuje. V tomto případě je možné smazat soubor *.htaccess* a v nastavení WordPressu přepnout zobrazení adres. Přepnutím se znovu vytvoří htaccess již přizpůsobený pro nový web.

6.2 Automatizovaná migrace

Automatická migrace je mnohem snazší nežli ruční. Ruční byla zmíněna hlavně z důvodu, abychom měli povědomí ohledně toho, co se vlastně při automatické migraci zjednodušeně děje, nebo se automatizovaná migrace zasekla a nebylo ji možné dokončit.

Automatizovanou migraci lze provést pomocí pluginů. V našem případě byl použit osvědčený plugin *Duplicator*, který existuje i v placené verzi, ale pro tento účel si vystačíme pouze se základní verzí. Plugin stačí nainstalovat, poté v jeho správě vytvořit nový balíček. Průvodce nás provede tvorbou balíčku a vytvoří *snapshot* což je vlastně zabalená adresářová struktura redakčního systému, exportovaná databáze a k tomu je vytvořen instalační soubor. Tyto dva soubory poté nahrajeme na nový hosting. Na něm pouze pomocí webového prohlížeče přejdeme na adresu instalačního souboru. Průvodce nás provede krok za krokem instalací. Vyplníme nutné údaje, web se rozbalí a zapíše do databáze potřebná data. Obdobným

způsobem lze dělat i kompletní zálohu webu, ke které se lze poté snadno vrátit. Pro zálohování však existují i jiné pluginy zaměřené pouze na tuto činnost.

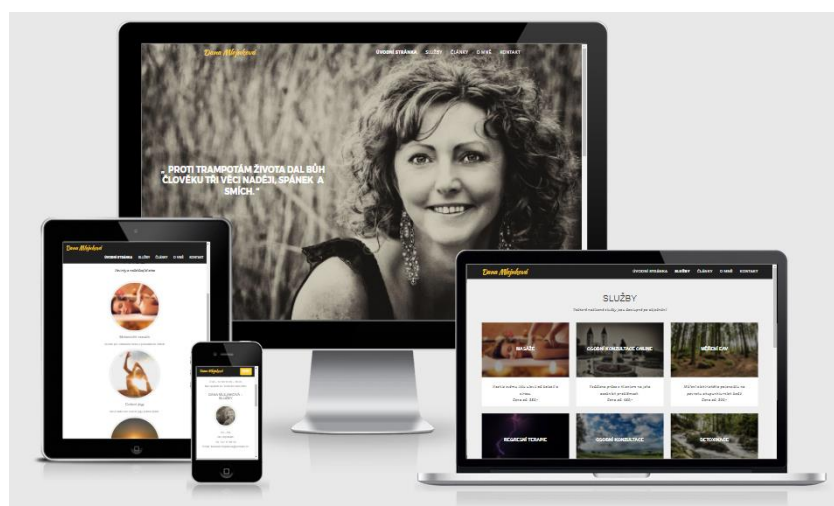
Na obrázku číslo 15 je zobrazen web na různých zařízeních před migrací.



Obr. 15 Zobrazení původní stránky na různých zařízeních

Zdroj: (39), (36)

Následující obrázek znázorňuje podobu webu po migraci na redakční systém WordPress a nasazení nově vytvořené šablony Agency.



Obr. 16 Podoba webu pro migraci

Zdroj: (39), (36)

Závěr

Cílem bakalářské práce bylo seznámit čtenáře s problematikou uživatelské správy webových stránek a možnosti migrace statických stránek na redakční systém za účelem dostupnější aktualizace obsahu. V teoretické části byly představeny dostupné redakční systémy, ze kterých byl vybrán WordPress pro svoji jednoduchost. Autor popisuje tento systém od počátku jeho vzniku až po současnost. Práce přináší shrnutí ekosystému včetně společných akcí pořádaných komunitou v České republice. Poslední kapitola teoretické části umožňuje čtenáři nahlédnout pod pokličku stále více populárnějšímu responzivnímu designu v jehož stylu se ponese i vytvořená šablona.

Praktická část nejprve analyzuje sbírané statistiky návštěvnosti za poslední rok, podle kterých jsou poté voleny parametry nové šablony. Autor si kladl na srdce přinést uživateli komplexní náhled na tvorbu nové šablony a provést ho krok za krokem postupem jejího vytváření. Samotné téma tvorby šablony je velmi obsáhlé a vzhledem k rozsahu práce nebylo možné popsat zde veškeré detaily. Z tohoto důvodu je součástí práce přiložené CD, na kterém se nachází obě vytvořené šablony a je možné si projít jejich zdrojové kódy.

V první fázi migrace byla vytvořena šablona kopírující původní vzhled a přenesena veškerá data na CMS. Z důvodu zastaralého formátování původních stránek a nízké možnosti optimalizace pro mobilní zařízení bylo přistoupeno k tvorbě nové modernější šablony postavené na Bootstrap templates. Tato šablona je nyní využívána pro prezentaci firmy. Její tvorba a přizpůsobení na míru danému webu je popsána v praktické části práce.

Jako budoucí vylepšení by bylo možné doplnit online objednávání, o které nyní nebyl zájem. Dále by bylo možné rozšířit administraci o jednotnou stranu pro úpravy cen všech služeb. Při stávajícím stavu je nutné pro změnu ceny upravit tuto položku u každé služby samostatně.

Majitelka webové prezentace má nyní stránky pod svojí správou skrze administraci a případné změny obsahu si již provádí sama.

Seznam použité literatury

Literární zdroje

7. SHREVES, Ric. *Joomla! Bible*. 2nd Edition. Indianapolis : John Wiley & Sons, Inc., 2013. str. ISBN 744. 978-1-118-47491-4.
9. DEXTER, Mark a Louis LANDRY. *Mistrovství v Joomla!* Brno : Computer Press, 2013. str. 560. ISBN 978-80-251-3740-6.
10. POLZER, Jan. *Drupal 8*. Brno : Computer Press, 2016. str. 360. ISBN 978-80-251-4147-2.
15. BRAZELL, Aaron. *WordPress Bible*. Indianapolis : Wiley Publishing, Inc., 2010. str. 675. ISBN 978-0-470-56813-2.
23. SHARKIE, Craig a Andrew FISHER. *Responzivní webdesign okamžitě*. Brno : Computer Press, 2015. str. 144. ISBN 978-80-251-4384-1.
30. KADLEC, Tim. *Responzivní design profesionálně*. Brno : ZONER Press, 2014. str. 248. ISBN 978-80-7413-280-3.
38. KUDLÁČEK, Lubomír. *WordPress : podrobný průvodce tvorbou a správou webů*. Brno : Computer Press, 2010. str. 261. ISBN 978-80-251-2734-6.

Ostatní zdroje

1. *W3Techs.com*: Extensive and reliable web technology surveys. Web Technologies Statistics and Trends. [Online] [Citace: 10. září 2016.] Dostupné z: <<https://w3techs.com/technologies>>.
2. *W3Techs.com*: Usage Statistics and Market Share of Content Management Systems for Websites, October 2016. Web Technologies Statistics and Trends. [Online] [Citace: 12. září 2016.] Dostupné z: <https://w3techs.com/technologies/overview/content_management/all/>.
3. *WordPress.com*: About » License – WordPress. [Online] [Citace: 9. září 2016.] Dostupné z: <<https://wordpress.org/about/license/>>.

4. H., Martin. Podrobná historie WordPressu. wplama. [Online] 10. červenec 2014. [Citace: 12. září 2016.] Dostupné z: <<http://www.wplama.cz/podrobna-historie-wordpressu/>>.

5. GAVALDA, Mark. The History of WordPress, its Ecosystem and Community. Kinsta. [Online] 8. červenec 2014. [Citace: 13. září 2016.] Dostupné z: <<https://kinsta.com/learn/wordpress-history/>>.

6. *Joomlaportal.cz*: Co je to Joomla? [Online] [Citace: 10. září 2016.] Dostupné z: <<http://www.joomlaportal.cz/dotazy/co-je-joomla>>.

8. *Joomla.org*: Release and support cycle – Joomla! Documentation. [Online] [Citace: 18. září 2016.] Dostupné z: <https://docs.joomla.org/Release_and_support_cycle>.

11. *WordPress.org*: Version Numbering – Make WordPress Core. WordPress. [Online] [Citace: 15. listopad 2016.] Dostupné z: <<https://make.wordpress.org/core/handbook/about/release-cycle/version-numbering/>>.

12. *WordPress.org*: Project Organization – Make WordPress Core. Get Involved – WordPress. [Online] [Citace: 15. listopad 2016.] Dostupné z: <<https://make.wordpress.org/core/handbook/about/organization/>>.

13. *Sfgate.com*: WordPress celebrates 10 years of growth – SFGate. SFGATE. [Online] 25. červenec 2013. [Citace: 13. leden 2017.] Dostupné z: <<http://www.sfgate.com/business/ontherecord/article/WordPress-celebrates-10-years-of-growth-4687933.php>>.

14. *Themeforest.net*: Envato market. 2017's Newest Premium Wordpress Themes from Themeforest. [Online] [Citace: 1. leden 2017.] Dostupné z: <<https://themeforest.net>>.

16. *WordPress.org*: WordPress Plugins. [Online] WordPress. [Citace: 15. leden 2017.] Dostupné z: <<https://wordpress.org/plugins/>>.

17. *Wpml.org*: About WPML – WPML. WPML. [Online] onTheGoSystem. [Citace: 16. leden 2017.] Dostupné z: <<https://wpml.org/home/about-us/>>.
18. *WordPress.org*: TinyMCE Advanced – WordPress Plugins. WordPress Plugins. [Online] WordPress. [Citace: 16. leden 2017.] Dostupné z: <<https://wordpress.org/plugins/tinymce-advanced/>>.
19. *Apek.cz*: Pro novináře | Asociace pro elektronickou komerci – APEK. Asociace pro elektronickou komerci. [Online] [Citace: 23. leden 2017.] Dostupné z: <<https://www.apek.cz/pro-novinare>>.
20. *WooCommerce.com*: WooCommerce – The Best eCommerce Platform for WordPress. [Online] [Citace: 23. leden 2017.] Dostupné z: <<https://woocommerce.com/>>.
21. *Toret.cz*: Toret EET – Toret.cz. Toret. [Online] [Citace: 23. leden 2017.] Dostupné z: <<http://toret.cz/produkt/toret-eet/ref/3/>>.
22. *WordPress.org*: How the Release Cycle Works – Make WordPress Core. Get Involved – WordPress. [Online] [Citace: 15. listopad 2016.] Dostupné z: <<https://make.wordpress.org/core/handbook/about/release-cycle/>>.
24. *W3Schools.com*: Browser Statistics. W3Schools. [Online] [Citace: 12. únor 2017.] Dostupné z: <<http://www.w3schools.com/browsers/>>.
25. MARCOTTE, Ethan. Responsive web design. A List Apart. [Online] 25. květen 2010. [Citace: 12. únor 2016.] Dostupné z: <<http://alistapart.com/article/responsive-web-design>>.
26. SMUS, Boris. High DPI Images for Variable Pixel Densities – HTML5 Rocks. HTML5 Rocks. [Online] 22. srpen 2012. [Citace: 14. únor 2017.] Dostupné z: <<https://www.html5rocks.com/en/mobile/high-dpi/>>.
27. *Pngimg.com*: Team PngImg. [Online] [Citace: 15. únor 2017.] Dostupné z: <<http://pngimg.com/download/5893>>.

28. JEHL, Scott. Picturefill. [Online] [Citace: 14. únor 2017.] Dostupné z: <<https://scottjehl.github.io/picturefill/>>.
29. ALLSOPP, John. A Dao of Web Design. A List Apart. [Online] 7. květen 2000. [Citace: 20. únor 2017.] Dostupné z: <<https://alistapart.com/article/dao>>.
31. *Gdswebdesign.co.uk*: GDS Web. How does Responsive Website Design work? [Online] 26. leden 2015. [Citace: 14. únor 2017.] Dostupné z: <<http://www.gdswebdesign.co.uk/blog/coding/responsive-website-design/>>.
32. GERCHEV, Ivaylo. The 5 Most Popular Frontend Frameworks of 2014 Compared – SitePoint. SITEPOINT. [Online] 11. prosinec 2014. [Citace: 4. březen 2017.] Dostupné z: <<https://www.sitepoint.com/5-most-popular-frontend-frameworks-compared/>>.
33. *Getskeleton.com*: Responsive CSS Boilerplate. [Online] Skeleton. [Citace: 4. březen 2017.] Dostupné z: <<http://getskeleton.com/>>.
34. *Getbootstrap.com*: Grid system · Bootstrap. Bootstrap. [Online] [Citace: 3. únor 2017.] Dostupné z: <<https://v4-alpha.getbootstrap.com/layout/grid/>>.
35. *Wirify.com*: Wirify – The web as wireframes. [Online] Volkside Pty Ltd. [Citace: 8. únor 2017.] Dostupné z: <<https://www.wirify.com/>>.
36. *Masaze-zdravavyziva.cz*: Masáže Kostelec nad Orlicí. Beruška Zdravíčko. [Online] [Citace: 17. leden 2017.] Dostupné z: <<http://www.masaze-zdravavyziva.cz/>>.
37. *Analytics.google.com*: Google Analytics. [Online] Google. [Citace: 4. březen 2017.] Dostupné z: <<https://analytics.google.com/analytics/>>.
39. *Ami.responsivedesign.is*: Am I Responsive? Responsive Design. [Online] [Citace: 4. červenec 2017.] Dostupné z: <<http://ami.responsivedesign.is/>>.

Seznam obrázků

Obr. 1 Administrační rozhraní WordPressu verze 0.7	10
Obr. 2 Graf vývoje obratu internetových obchodů za zboží	18
Obr. 3 Porovnání rozlišení obrazovek 4K a Full HD při zobrazení stejného obrázku s šířkou 1280 pixelů.....	24
Obr. 4 Fluidní mřížky v rozvržení webu zobrazené na různých zařízeních.....	28
Obr. 5 Zobrazení řádku při různých rozlišeních viz kód 3.	31
Obr. 6 Wireframe úvodní stránky webu při FullHD rozlišení	33
Obr. 7 Vzhled úvodní webové stránky při rozlišení 1280 x 1024.....	34
Obr. 8 Členění souborové struktury	35
Obr. 9 Přehled toku návštěvníků na webu a jejich první dvě interakce	39
Obr. 10 Instalační průvodce nastavení přístupu k databázi	41
Obr. 11 Rozvržení elementů hlavní stránky	44
Obr. 12 Struktura souborů šablony	47
Obr. 13 Tvorba vlastního příspěvku v pluginu ACF	53
Obr. 14 Přiřazení šablony stránky konkrétní stránce	55
Obr. 15 Zobrazení původní stránek na různých zařízeních	59
Obr. 16 Podoba webu pro migraci	59

Seznam tabulek

Tab. 1 Trend rozlišení obrazovky na webu W3School za posledních 5 let.	22
Tab. 2 Body zlomu a jejich třídy dle šířky obrazovky.....	30
Tab. 3 Grid a jeho dělení v různých kombinacích.....	30

Seznam grafů

Graf 1. Procentuální podíl zástupců nejvyužívanějších redakčních systémů na internetu.....	3
Graf 2. Podíl využití technologií webových stránek na internetu.....	4
Graf 3. Nejpoužívanější rozlišení obrazovky.....	37
Graf 4. Nejpoužívanější webový prohlížeč.....	37
Graf 5. Koncová zařízení, přes které uživatel přistupuje na web.....	38

Seznam zdrojových kódů

Kód 1. Příklad typu média v tomto případě tiskárna.....	25
Kód 2. Příklad použití dotazu na media s maximální šířkou 640 pixelů	25
Kód 3. Příklad rozvržení pro malé a střední obrazovky	31
Kód 4. Stylování hlavního kontejneru, ve kterém je zobrazován obsah.....	33
Kód 5. Script sbírající data o návštěvnících do služby Google Analytics.....	36
Kód 6. Zjednodušené rozdělení hlavních elementů stránky	43
Kód 7. Soubor style.css určuje název a další informace o šabloně	46
Kód 8. Zjednodušené uspořádání souboru index.php	46
Kód 9. Slouží k zobrazení menu	48
Kód 10. Výpis náhodného mota z databáze	49
Kód 11. Registrace menu pro šablonu	50
Kód 12. Vytvoření a použití shortcode	50
Kód 13. Odstranění položek menu	51
Kód 14. Vytváření šablony pro nové miniatury	52
Kód 15. Registrace nového typu příspěvku	52
Kód 16. Definice šablony stránky	55

Seznam zkratek

CMS – Content Management System – systém pro správu obsahu

FTP – File Transfer Protocol – protokol pro přenos souborů

HTML – HyperText Markup Language – značkovací jazyk pro tvorbu webových stránek

WYSIWYG – What you see is what you get – zkratka co vidíš to dostaneš používaná pro vizuální editory

SVN – Subversion – open source nástroj pro správu verzí

URL – Uniform Resource Locator – adresa zdroje

CEO – Chief executive officer – výkonný ředitel společnosti

SEO – Search Engine Optimization – optimalizace pro vyhledávače

CSS – Cascading Style Sheets – kaskádové styly

MIT – Massachusetts Institute of Technology – licence se jmenuje podle místa vzniku

PHP – Personal Home Page – skriptovací programovací jazyk

PPI – body na palec

Px – pixel

Em – čtverčik

\$ - dolar

Přílohy

Následující přílohy jsou součástí dodávaného CD:

- 1) šablona Agency pro WordPress,
- 2) šablona vytvořená podle designu původního webu pro WordPress,
- 3) šablona Agency originál HTML verze,
- 4) šablona Agency upravená HTML verze,
- 5) kompletní záloha původního webu,
- 6) export ročních sledovaných dat ze služby Google Analytic.

Podklad pro zadání BAKALÁŘSKÉ práce studenta

PŘEDKLÁDÁ:	ADRESA	OSOBNÍ ČÍSLO
Mlejnek Ondřej DiS.	Na Lávkách 55, Kostelec nad Orlicí	I1300565

TÉMA ČESKY:

Migrace statického webu na CMS za účelem zefektivnění správy

TÉMA ANGLICKY:

Migration of static website to CMS for effective administration

VEDOUCÍ PRÁCE:

Ing Andrea Vokálová - KIKM

ZÁSADY PRO VYPRACOVÁNÍ:

Cílem mé práce je kompletně převést daný web ze statické aktuální podoby do CMS za účelem lepší a jednodušší správy přímo vlastníkem. Nový web bude využívat redakční systém WordPress, kterým se bude tato práce z velké části zabývat.

OSNOVA:

Úvod

1. CMS systémy

2. WordPress

3. Praktická část

4. Výsledky

Závěr, Zdroje

SEZNAM DOPORUČENÉ LITERATURY:

WordPress podrobný průvodce tvorbou a správou webů
Luboš Kudláček ISBN 978-80-251-2734-6

333 tipů a triků pro WordPress
Michaela Horňáková ISBN 978-80-251-3443-6

Podpis studenta: 

Datum: 12.10.2016

Podpis vedoucího práce: 

Datum: 12.10.2016