

Redakční systém pro Cestovatelský magazín obsahující katalog ubytovacích zařízení

Vojtěch Reiner

Bakalářská práce
2018



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Vojtěch Reiner**
Osobní číslo: **A14204**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**
Forma studia: **kombinovaná**

Téma práce: **Redakční systém pro Cestovatelský magazín obsahující katalog ubytovacích zařízení**

Téma anglicky: **An Editorial Suite for the Travel Magazine Containing a Catalogue of Accommodation Facilities**

Zásady pro vypracování:

1. Vypracujte literární rešerši analyzující a popisující webové servery zaměřené na cestování.
 2. Provedte návrh struktury redakčního serveru pro Cestovatelský magazín.
 3. Vytvořte grafický návrh webu.
 4. Návrhy z bodů 2 a 3 implementujte.
 5. Vytvořte specifický modul realizující katalog ubytování s možností filtrace podle různých parametrů.
 6. Provedte analýzu celého systému z hlediska bezpečnosti.
-

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **Larry Ullman: PHP a MySQL – Názorný průvodce tvorbou dynamických WWW stránek, Computer Press, červen 2004.**
2. **Marek Prokop: CSS kaskádové styly pro webdesignéry (2. vydání), Computer Press, Praha 2006.**
3. **Mach, J.: PHP pro úplné začátečníky, Computer Press, Praha 2002.**
4. **Jiří Kosek: PHP – tvorba interaktivních internetových aplikací, Grada Publishing 1999.**
5. **W. Jason Gilmore: Velká kniha PHP 5 a MySQL, Zoner Press, 2011.**
6. **David LeBlanc, Michael Howard: Bezpečný kód, COMPUTER PRESS, 2008.**

Vedoucí bakalářské práce:

Ing. Tomáš Sysala, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

15. prosince 2017

Termín odevzdání bakalářské práce:

25. května 2018

Ve Zlíně dne 15. prosince 2017



doc. Mgr. Milan Adámek, Ph.D.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu


Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 23.5. 2018


.....
podpis diplomanta

ABSTRAKT

Cílem bakalářské práce je vytvoření kompletních www stránek včetně vlastního redakčního systému. Jedná se o specifický webový portál zaměřený na cestování. Ke standardnímu redakčnímu systému (správa statických článků, kategorií, komentáře jak na frontendu tak jejich správa v administraci) jsem vytvořil i specifickou část pro katalog ubytovacích zařízení, který je s webem propojen pomocí horizontálního navigačního menu. Součástí katalogu ubytovacích zařízení je filtr, kde probíhá vyhledávání dle kraje, ceny, typu zařízení a dalších parametrů. Každé ubytovací zařízení má vlastní podstránku (nebo-li detail) na které je popis objektu, fotogalerie, atributy (zda je povolen pes, je-li objekt bezbariérový, do budoucna můžeme případně zařadit další parametry). Součástí www stránek je formulář pro registraci nového zařízení a také administrační rozhraní pro správce webu, kde je možné nová zařízení schvalovat, případně mazat. Stránky jsou napsané v jazyce HTML s využitím kaskádových stylů. Redakční systém s administrací v jazyce PHP s využitím MySQL pro ukládání dat (článků, údajích k ubytovacím zařízením atp.). Pro některé pomocné funkce jsem využil JavaScript + jeho knihovnu jQuery.

Klíčová slova: katalog ubytovacích zařízení, cestovatelský magazín, www technologie

ABSTRACT

The aim of the bachelor thesis is create complete www pages including editorial system. It is a specific web portal of traveling. To the standard editorial system (managing static articles, categories, comments on both the front and their management in the administration) I created a specific section - the catalog of accommodation facilities, which is linked to the web via the horizontal navigation menu. Part of the accommodation catalog is a filter, where is a search section. Search is possible by region, price, type of equipment and other parameters. Each accommodation facility has its own sub-page (detail) on which the object is fully described, photo gallery, attributes (dog is allowed, if the object is barrier-free, we may possibly include other parameters in the future). The web site includes a new device registration form and an administrative interface for webmasters, where it is possible to approve, delete. The web page is written in HTML using cascading styles. Content management system is created by PHP with MySQL for storing data (articles, data on accommodation facilities, etc.). For some helper features, I used JavaScript + jQuery library.

Keywords: catalog of accommodation facilities, travel magazine, www technology

Děkuji vedoucímu práce panu Ing. Tomáši Sysalovi, Ph.D za vedení, rady a doporučení při realizaci této bakalářské práce.

Prohlašuji, že odevzdaná verze bakalářské/diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	8
I. TEORETICKÁ ČÁST	9
1. WEBOVÉ SERVERY ZAMĚŘENÉ NA CESTOVÁNÍ	10
1.1 SOUČASNÝ STAV WEBOVÝCH SERVERŮ ZAMĚŘENÝCH NA CESTOVÁNÍ.....	10
1.2 KONKRÉTNÍ PŘÍKLADY.....	11
1.3 MOJE VIZE.....	15
2. PROGRAMOVÉ VYBAVENÍ	17
2.1 VYTVOŘENÍ GRAFICKÉHO NÁVRHU.....	17
2.2 EDITOR KÓDU (IDE).....	17
2.3 WEBOVÝ SERVER.....	18
2.4 PHPMYADMIN.....	18
3. POUŽITÉ TECHNOLOGIE	20
3.1 HTML.....	20
3.2 CSS.....	20
3.3 PHP.....	21
3.4 ARCHITEKTONICKÉ VZORY.....	23
3.5 SQL.....	25
3.6 JAVASCRIPT.....	25
4. WEBDESIGN	26
4.1 O WEBDESIGNU.....	26
4.2 ZÁKLADNÍ FORMÁTY WEBOVÉ GRAFIKY.....	26
II. PRAKTICKÁ ČÁST	29
1. NÁVRH STRUKTURY REDAKČNÍHO SYSTÉMU	30
2. NÁVRH DESIGNU	31
2.1 VÝSLEDNÝ NÁVRH.....	33
2.2 VYTVOŘENÍ ŠABLONY.....	34
2.3 RYCHLOST NAČÍTÁNÍ.....	34
3. REDAKČNÍ SYSTÉM	35
3.1 FRONTEND.....	36
3.2 ADMINISTRAČNÍ ROZHRANÍ (BACKEND).....	37
4. DATABÁZE	38
5. ZABEZPEČENÍ	43
ZÁVĚR	47
SEZNAM POUŽITÉ LITERATURY	48
SEZNAM POUŽITÝCH ZKRATEK A POJMŮ	49
SEZNAM OBRÁZKŮ	50
SEZNAM PŘÍLOH	51

ÚVOD

Webové stránky zažívají několik posledních let velký rozvoj a už není důležité pouze to, aby se web jakýmkoliv způsobem vykreslil a předal uživateli informace, ale vzhledem k opravdu velkému množství www stránek je také důležité aby web návštěvníka zaujal a určitým způsobem i vyčníval. Ruku v ruce s designem a funkčností webu jde obsah webu, který i nadále zůstává nejdůležitějším faktorem. Ovšem důležité je jakým způsobem je obsah prezentován. Ve své práci jsem neřešil pouze vzhled a funkčnost systému, ale svým způsobem také obsahovou část, protože rozložení některých prvků je dané právě šablonou webu a pro autory příspěvků i uživatele je neměnné.

Cílem této bakalářské práce je analýza současného stavu webových magazínů zaměřených na cestování a vytvoření magazínu nového. Součástí magazínu bude katalog ubytovacích zařízení, který bude sloužit uživatelům se zájem o cestování k jednoduchému vyhledání ubytovacího zařízení v určité lokalitě. Katalog bude obsahovat filtr, kde si mimo lokality můžeme určit také cenu, typ ubytovacího zařízení a další parametry. Web tak bude spojovat tyto dvě části – obsahovou a vyhledávací.

Správa objektů (ubytovacích zařízení) a přidávání + editace článků v magazínu bude probíhat z administrace webu a ani od správce objektu, ani od redaktora tak nebude vyžadována znalost použité technologie, ani žádných speciálních dovedností.

I. TEORETICKÁ ČÁST

1. Webové servery zaměřené na cestování

1.1 Současný stav webových serverů zaměřených na cestování

Podělit se o své zážitky s cestování mohli lidé už dlouhá staletí. S rozvojem technologií bylo možné cestovat jednodušeji na vzdálenější a vzdálenější místa a z pověstných Foghových 80 dní na procestování zeměkoule se stalo jen „pár“ hodin v moderním letadle. Až donedávna mohla být touha podělit se o své cestovatelské zážitky rozvinuta pouze u profesionálních cestovatelů, dá se říci redaktorů cestujících pod nějakým časopisem, později i pro film a televizi. Přes velký rozvoj těchto technologií tyto možnosti mělo jen několik vyvolených. Vydat na vlastní náklady třeba cestopis není a nikdy nebyla levná záležitost. Navíc muselo být dílo ucelené, dobře připravené a při touze podělit se o nový zážitek – vydání nového dílu. S nástupem internetu se situace razantně změnila. Svůj blog, osobní web, cestovatelskou stránku může mít víceméně každý člověk. Založení takového blogu je snadné a rychlé, dokonce může být i bez vynaložení finančních prostředků. Během pár minut tak může dotyčný sdílet svoje zážitky z cest nejen formou psaného textu, ale i za pomoci fotografií a videí. To je jedna, řekl bych majoritní kategorie cestovalských webů.

Nevýhod takových www stránek je hned několik. Jsou vesměs všechny vybudovány na šablonovitém systému, ve většině případů na doméně 3. řádu na nějakém freehostingu, nebo rovnou „webu na klik“, čímž myslím vytvoření tuctového webu doslova na několik kliknutí bez jakýchkoliv znalostí v této oblasti. Taková šablona často vůbec neodpovídá tématu, má špatnou navigační strukturu, má chybně nazvané kategorie, nebo i třeba kategorie, či jiné prvky které s webem vůbec nesouvisí, ale autor netuší jak je v šablonovitém systému „na klik“ odstranit, nebo vyměnit. Stránky jsou plné netématické reklamy, často PR článků a blogů které mají kvalitní obsah je jako šafránu. Drtivá většina takto založených blogů končí s méně než 10 narychlo sepsanými články s fotografiemi staženými s internetu, často s logotypem jiného serveru.

Druhou, ve své podstatě menší kategorií co se počtu webů týče jsou cestovatelské servery. Budu tak nazývat weby které jsou řekněme profesionální, nebo poloprofesionální. Obvykle na ně přispívá více autorů, jsou na nich stovky, nebo tisíce článků, mívají prostor pro placenou reklamu, doménu 2. řádu a vlastní redakční systém.

Cestovatelských serverů už není takové množství jako blogů, většinou má autor nějaký podnikatelský, nebo osobní záměr, nicméně je třeba investovat mnoho úsilí a často také nemalé finanční prostředky. Z toho důvodu takových magazínů neexistují tisíce, ale desítky. Mnohé mají také přidružené diskusní fórum, nebo jsou součástí většího portálu, který se nevěnuje pouze cestování, ale zprávám z různých kategorií.

Většina takových portálů je špatně koncipovaných, mají zastaralou grafickou úpravu, jsou nepřehledné a doslova přeplněné mnohdy netematickou reklamou. Při mém hledání na internetu jsem naopak nenašel žádný, který by byl přímo propojen s vyhledáváním ubytovacích zařízení. Což je poměrně důležitá část webu, tematická a pro uživatele přínosná. Navíc, co je velmi důležité, web má velký potenciál dosáhnout zisku nenásilnou formou – vůbec se nejedná o obtěžující reklamu, ale naopak o něco co uživatelé třeba hledají, nebo je to zaujme. Pokud se uživatel třeba na základě poutavého článku rozhodne lokalitu navštívit a není mu právě příjemné spát pod širákem, bude jistě hledat nějaké ubytování v okolí. Musí tedy opustit stránku, vyhledat neprověřený web, samostatný katalog těchto zařízení a mnohdy složitě a bezparametricky ubytovací zařízení vyhledat.

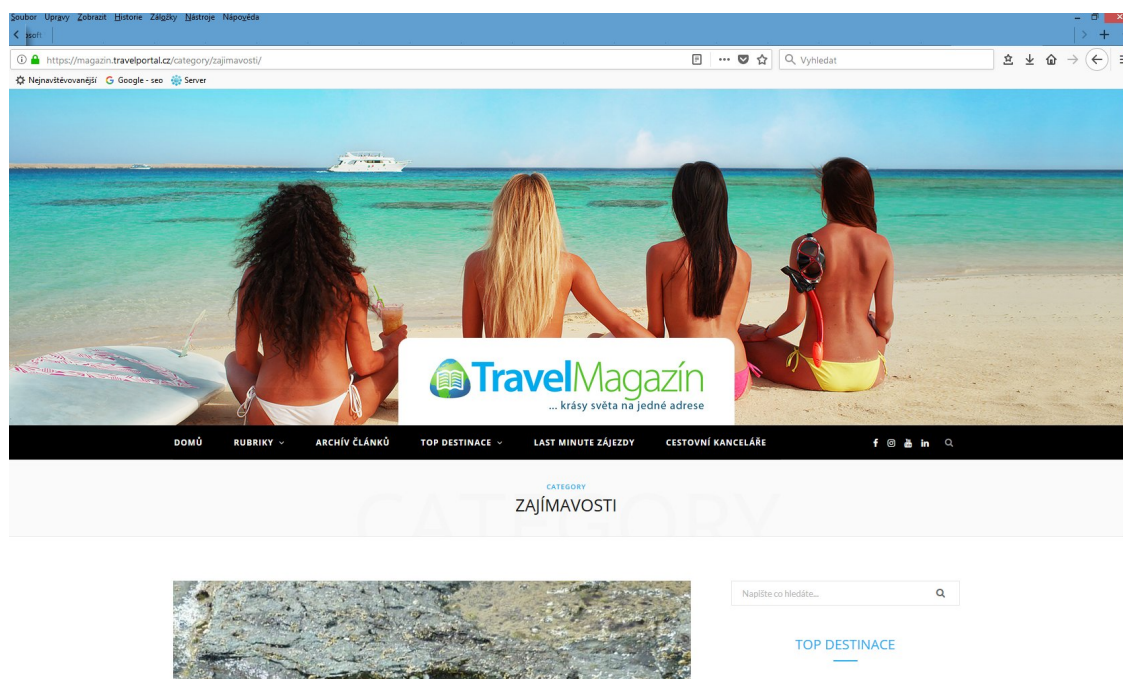
1.2 Konkrétní příklady

Magazín travelportal

URL: <https://magazin.travelportal.cz>

Jedná se v podstatě o propagační magazín cestovatelské kanceláře. Menu je společné jak pro magazín, tak pro cestovní kancelář, což považuji za nesprávné. Jako čtenář netuším že budu přesměrován na jiný web, ačkoliv je součástí domény, vypadá úplně jinak a jeho obsah je také odlišný od magazínu. Takže co se týče samotného webu, obsahuje velmi omezené množství kategorií, které jsou pouze obecné, není zde možnost výběru konkrétní destinace, určitého typu památek atp. Pouze obecné rubriky typu „zajímavosti, novinky“.

Design webu je jednoduchý, moderní a intuitivní. V této oblasti nemám co vytknout. Snad jen absentující hover efekt víceméně u všech prvků mimo hlavní navigace.



Obrázek 1: Náhled webu magazin.travelportal.cz [14]

Magazín o cestování

URL: <https://magazin-cestovani.cz/>

Obsahově hodnotný magazín, ačkoliv některé poslední články nejsou evidentně klasickými články, ale tzv. PR články. Jedná se o placené příspěvky v jejichž obsahu je jeden, nebo více odkazů na inzerenta. Obecně je tento web poměrně plný reklamy – má svůj cca 400x400px velký blok v levém sloupci, 120x600px po pravé straně a ještě 500x80px přímo v článku.

Design webu působí zastarale, v roce 2000-2010 byl aktuální. Postrádám hover efekty u nadpisů článků, jako plus hodnotím související články, ikdyž mi připadá že název „související“ je přinejmenším zavádějící, vypadá to že články jsou generované náhodně.



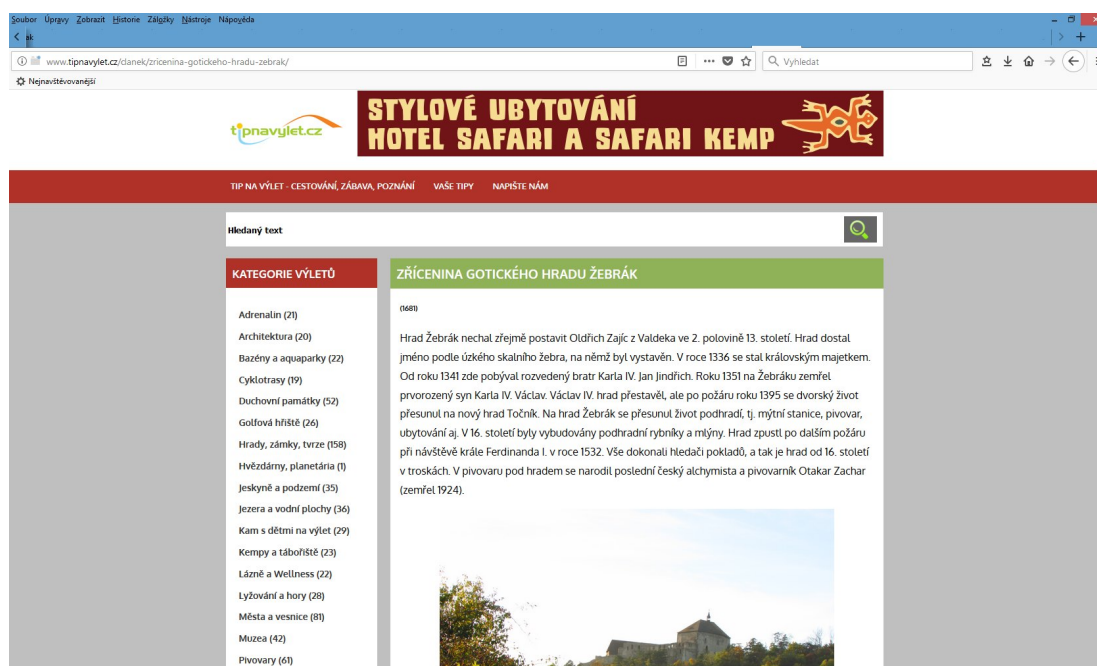
Obrázek 2: Náhled webu magazin-cestovani.cz [15]

Tipy na výlet.cz

URL: <http://www.tipnavylet.cz>

Velmi jednoduchý web co se grafického zpracování týče. Jednobarevné hlavní menu, rozdělení webu na 2 segmenty s velmi jednoduchou grafikou tvořenou obdelníky a kombinací 2 hlavních barev. Povedená struktura URL adres.

Vzhledem k tomu že web s největší pravděpodobností nemá redakční část pro více autorů (vycházím z toho, že neobsahuje žádné informace o možnosti přispívání, u článku není uveden autor), působí některé články chaoticky, některé články jsou převzaty z oficiálních zdrojů lokalit, či destinací. Některé články jsou z hlediska rozsahu skromější, slouží pro stručné informování o dané lokalitě/místě. Každopádně web z hlediska obsahu určitě má co nabídnout a každý článek obsahuje také krátkou fotogalerii. Reklama je na webu rozumnou formou – v hlavičce a dole v levém navigačním sloupci. Co jsem měl možnost shlédnout aktuálně zobrazené bannery, nepůsobili nijak rušivě a byli víceméně tematické. Další částí webu je sekce „Vaše tipy“ kde jsou pravděpodobně odděleny články externích, nahodilých autorů.



Obrázek 3: Náhled webu tipnavylet.cz [16]

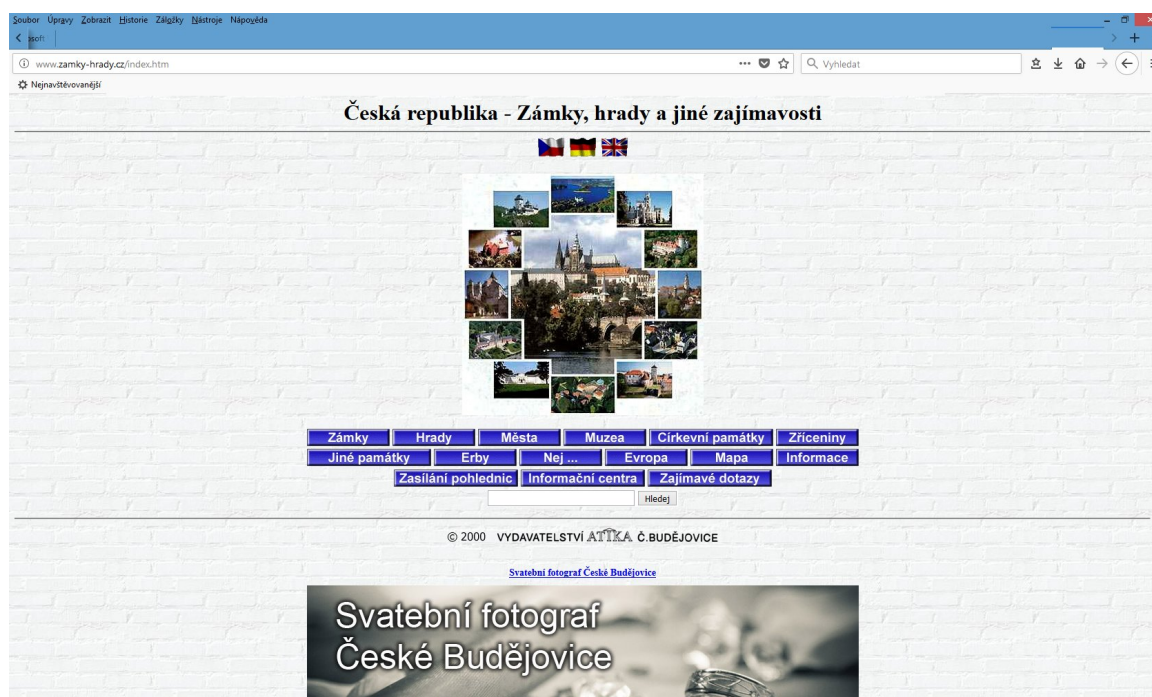
Zámky, hrady.cz

URL: <http://www.zamky-hrady.cz/>

Velmi výstižná doména, která by si jistě zaloužila aktuálnější prezentaci. V patičce je uveden rok 2000, čili s tím, že je webdesign aktuální k tomuto roku plně souhlasím. K zastaralému stylu webu se nemá smysl vyjadřovat, vývoj jde neustále kupředu a na rok 2000 byl tento webdesign jistě naprosto v pořádku. Úvodní obrázek by měl být typu png, případně gif, jelikož použitý jpg nepodporuje průhlednost. Z toho důvodu jsou cihly na pozadí překryté bílou barvou v okolí hlavního obrázku. Ten je svým způsobem povedený, nicméně i kvalitou neodpovídá a určitě by šel vyexportovat lépe. U hlavní navigace postrádám hover efekt, ten stejný problém je po kliku na položku v podkategorii.

Obsahově je web poměrně rozsáhlý, každá položka zahrnuje korespondenční adresu objektu, telefonní kontakt a případné webové stránky. Součástí je vždy fotografie a kratší, či delší popis. Navzdory zastaralému provedení přes odkazy (aktuálně můžu vybírat jen z jedné podkategorie, na detailu není k dispozici vůbec žádné menu) je web poměrně přehledný a zajisté má svou informační hodnotu.

Reklama je pouze v druhé části úvodní stránky a to nenásilnou formou – jeden banner a několik textových odkazů. Web je dokonce k dispozici ve třech světových jazycích.



Obrázek 4: Náhled webu zamky-hrady.cz [17]

1.3 Moje vize

Ve svém projektu se chci držet jednoduché koncepce magazínu, moderní a přehledné grafiky a intuitivního propojení s katalogem ubytování. Náš portál tedy skloubí obě dvě části – informační a vyhledávací. V informační části uživatelé najdou zajímavé informace, tipy na výlety, rady doporučení a krásné fotky z cest. Zdrojem těchto článků nebudou pouze zkušenosti a zážitky hlavního administrátora, ale přispět může kdokoliv z řad široké veřejnosti – postačí k tomu krátká registrace a ověření emailové adresy kvůli případnému zneužívání k reklamě, spamu atp. Ve vyhledávací části bude možnost vyhledat si ubytovací zařízení přesně podle zadaných parametrů a na vyfiltrované objekty se detailně podívat a případně si pobyt objednat. Součástí katalogu ubytování bude také hodnocení předchozích zákazníků, aby bylo možné vybírat z prověřených objektů.

Co se týče struktury, většina mnou testovaných cestovatelských magazínů na úvodní stránce obsahuje obecné informace o portálu, k čemu slouží, obsahují rozcestník, který je identický z hlavní nabídkou, neboli menu webu. Některé obsahují statistiky kolik magazín obsahuje článků, fotogalerií atp. Uveden bývá také kontakt pro případné dotazy a podněty čtenářů.

Já zvolím jednoduchý styl hlavního menu – horizontální navigace, která bude sloužit pro rozdělení webu na stěžejní části. Nabízí se položky „úvod do magazínu, katalog ubytovacích zařízení, o webu, kontakt“. Tyto hlavní kategorie jsou plně dostačující pro provoz webu a navigace bude k dis-

pozici z každého místa uživatelské části webu. Obsah bude rozdělen na dvě části – levou, podstatně menší část bude tvořit menu magazínu, kde budou příslušené kategorie pro zveřejňované články. Kategorie se mohou v průběhu času měnit dle aktuálních potřeb zákazníků. Měnit by je měl pouze správce webu, jde o poměrně zásadní změnu, která se neprovádí denně. Pravá část bude vyhrazena pro obsah – tedy pro články magazínu, ale také pro výpis katalogu ubytovacích zařízení, detaily ubytovacích zařízení, nebo obecné položky z hlavní horizontální navigace. Ve dolní části webu bude cca 40px vysoké zápatí pro informaci o tvůrci webu a bude zde také místo pro krátké odkazy sloužící pro rychlou navigaci, které nemají být viditelné v hlavním menu (přidávání ubytovacího zařízení, administrace atp.).

Informace o statistikách nepovažuji pro návštěvníky za důležité, proto je nikde na webu uvádět nebudu. Mnohem důležitější je, zda se uživatel dokáže k článku pomocí navigace webu pohodlně dostat a není pro něj zásadní informace kolik je na webu článků, kolik fotografií v galerii atp. Pokud má taková stránka na webu být, nemělo by to být v hlavní navigaci a už vůbec ne v úvodním článku.

Technické řešení

Tvorbu webových stránek v dnešní době bych rozdělil na tyto způsoby:

- Uživatelské vytvoření webu (blogu) na některé z těchto služeb
- Vytvoření webu za pomoci šablony (a případně opensource RS)
- Naprogramování a nakódování celého webu samostatně

K tvorbě cestovatelského magazínu by bylo reálné použít (2) a (3) variantu. Druhou variantu jsem v tomto případě vyloučil z toho důvodu, aby byl web vytvořený kompletně na míru, bez zbytečných funkcí, komponent a ostatních částí systému, které nevyužiji. Jistě by byla tvorba některých částí v případě hotového CMS jednodušší, v případě funkcí na míru by ale byla implementace složitější a vyžadovalo by to podrobné znalosti konkrétního CMS systému, případně frameworku. Šablonu pro vzhled www stránky nepoužívám ze zřejmého důvodu – odlišení od konkurence. Jednak by určitě nebylo dobré, kdyby měl stejný vzhled náš magazín s katalogem jako třeba obecní stránky. Také je důležité aby vzhled návštěvníka zaujmul, cítil se na něm příjemně a byl pokud-možno tematický. Proto jsem zvolil jednak naprogramování celého systému samostatně – za pomoci PHP a architektu-

ry MVC – tu jsem zvolil z důvodu lepší přehlednosti a jednodušších úprav v případě jiného programátora. Šablonu si také vytvořím vlastní – grafický návrh v programu pro rastrovou grafiku a funkční šablonu za pomoci HTML a CSS.

Postup naprogramování a nakódování celého webu bych rozdělil opět do tří částí:

- Vytvoření grafického návrhu (designu, vzhledu)
- Nakódování šablony (převod z „obrázkové“ formy do funkční šablony)
- Programování systému a administrace webu

2. PROGRAMOVÉ VYBAVENÍ

2.1 Vytvoření grafického návrhu

K vytvoření grafického návrhu existuje velké množství editorů. Pro tvorbu logotypu je vhodnější vektorový editor, z důvodu že lze jednoduše navrhnout logo ve křivkách, většina logotypů je složena z různých tvarů, stejně tak jako naše logo. Důležité také je aby bylo možné logotyp vyexportovat např. pro tisk, nebo další zpracování a k tomu je vhodný jednoznačně vektorový formát. Jako editor jsem zvolil Inkscape, který je dostupný ve freeware verzi a dokonce je distribuován jako opensource. Z hlediska rastrové grafiky a tedy grafického návrhu webu mimo loga, jsem se rozhodl pro použití programu Photofiltre, který je do verze 7.0 zdarma pro jakékoliv použití. Program pro rastrovou grafiku můžu doporučit, obsahuje všechny základní funkce pro počítačovou grafiku.

2.2 Editor kódu (IDE)

Jako vývojové prostředí (Integrated Development Environment) jsem zvolil Notepad++ mající GPL licenci:

GNU General Public License. Software šířený pod licencí GPL je možno volně používat, modifikovat i šířit, ale za předpokladu, že tento software bude šířen bezplatně (případně za distribuční náklady) s možností získat bezplatně zdrojové kódy. Toto opatření se týká nejen samotného softwaru, ale i softwaru, který je od něj odvozen. Na produkty šířené pod GPL se nevztahuje žádná záruka. Licence je schválena sdružením OSI a plně odpovídá Debian Free Software Guidelines.

Tento program obsahuje velké množství pluginů, mimo jiné např. kompilátor pro C++, nebo FTP klienta, což jsem využil právě já při programování magazínu. Pracoval jsem online na vlastním serveru, takže jsem potřeboval být neustále připojený, což mi Notepad++ společně s pluginem „NppFTP“ umožnili. Editor můžu doporučit, po jednoduchém nastavení jsem si na něj zvyknul a s jiným editorem už nepracuji.

2.3 Webový server

Pro ladění aplikací se používá tzv. virtuální server Apache:

Apache HTTP Server je softwarový webový server s otevřeným kódem pro GNU/Linux, BSD, Solaris, Mac OS X, Microsoft Windows a další platformy. V současné době dodává prohlížečům na celém světě většinu internetových stránek.

Pokud se rozhodnete na svém serveru provozovat webové stránky, bude Apache logická volba. Apache samotný slouží pro zobrazování stránek a pokud potřebujete server na složitější prezentaci, například CMS systémy WordPress a Drupal, potřebujete i podporu PHP a databázi. Tyto nejpoužívanější komponenty se označují jako LAMP server podle počátečních písmen Linux, Apache, MySQL a PHP.

[1]

Ten lze jednoduše nainstalovat na každém počítači a ladit tak web bez připojení k internetu. Já jsem se rozhodl ladit web přímo na skutečném webovém serveru, kde bude odzkoušený a ostrá verze může fungovat na totožném nastavení, případně přímo na stejném serveru. Mám nastavenou tuto konfiguraci:

Verze PHP: 5.6

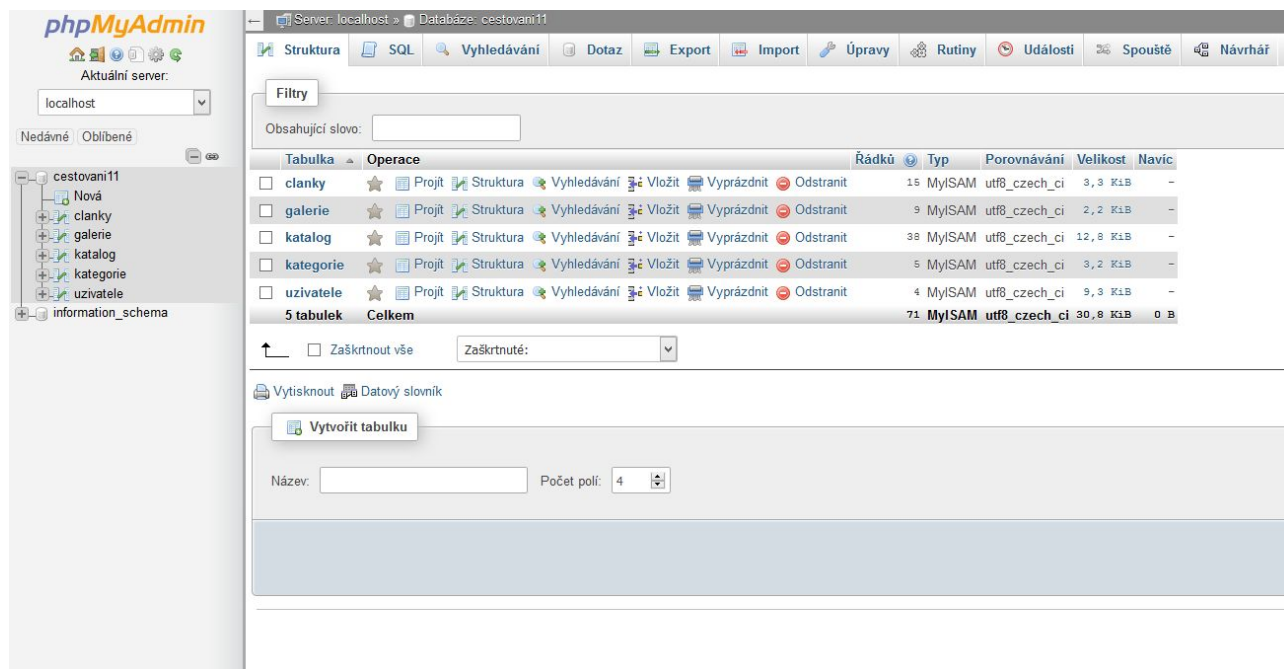
Typ databáze: MySQL 5 MyISAM

Verze PHPmyAdmin: 4.7.7

2.4 PHPMYADMIN

PHPmyAdmin je aplikace vytvořená v PHP a slouží pro snadnou obsluhu databáze MySQL. Licencovaný je jako Open Source. Je tedy volně šiřitelný vč. zdrojového kódu. Bez znalosti programování lze vytvářet databáze, tabulky, měnit jejich strukturu, nebo i samotný obsah. Aplika-

ce funguje přes webové rozhraní v podstatě jako běžná stránka. PHPmyAdmin obsahuje také pokročilé funkce pro zálohování, opět v uživatelsky přívětivé podobě.



Obrázek 5: Ukázka prostředí phpMyAdmin

Alternativou je nástroj českého autora Jakuba Vrány nazvaný Adminer, také velmi rychlá a přehledná aplikace pro snadnou zprávu databáze. Jedná se o jednodušší verzi phpMyAdminu, která je při své práci ale téměř 3x rychlejší.

MYSQL

MYSQL je zkratka z anglického „My Structured Query Language“ a jedná se o jeden z nejoblíbenějších systémů pro správu databáze, zejména v prostředí internetu. Systém je šířen pod Open Source licencí, je velmi rychlý a jednoduchý pro správu. Mimo PHP se tento systém dá využít jako databáze k programům v jazycích C, Java, .NET, C++ a další. Oproti velkým databázovým systémům jako třeba Oracle má určitá omezení, ale pro běžné webové aplikace plně dostačuje a je velmi rychlý.

3. POUŽITÉ TECHNOLOGIE

3.1 HTML

HTML (Hypertext Markup Language) je jednoduchý značkovací jazyk, který slouží k tvorbě webových dokumentů. Výhodou je jeho velká univerzálnost, je použitelný na Všechných platformách. Dokument je složen z tagů, které píšeme do ostrých závorek < >. Každý tag má svůj název a může, ale také nemusí mít atributy. Existují také párové a nepárové tagy. Příkladem nepárového je tag
, který slouží pro vytvoření pevné vertikální mezery. Naopak třeba tag <a> sloužící jako odkaz je párový a musíme jej uzavřít pomocí lomítka . Jazyk se stále vyvíjí a některé nepoužívané tagy už jsou téměř vypuštěny. Jedná se zejména o tagy a atributy určující velikosti – písma, buňek, ale třeba i sloupců v tabulce. Tyto věci se řeší přes CSS v externím souboru.

HTML se se vyvinul ze SGML a stal se víceméně jediným používaným jazykem pro tvorbu www stránek. Aktuální verze jazyka je HTML 5. Z HTML se vyvinulo také XHTML (extended hypertext markup language) jako aplikace XML, které se nepoužívá pro samotné webové stránky, ale třeba pro RSS, nebo feedy vyhledávačů, dodavatelů atp.

3.2 CSS

Zkratka pochází z anglického Cascading Style Sheets, nebo-li kaskádové styly. Společně s HTML tvoří stěžejní jazyk pro statické dokumenty na webu. Pomocí stylů můžeme hromadně upravovat více stránek bez nutnosti úprav na každé stránce zvlášť, stačí importovat stejný CSS dokument. CSS lze také použít přímo v HTML dokumentu, kdy se zapíše do hlavičky pomocí tagu <style>, případně přímo do atributu „style“ konkrétního prvku. Poté se jedná o tzv. Inline-zápis, který je nutné používat s mírou a v drtivé většině je lepší nahradit jej externím souborem. CSS souborů vložených do dokumentů může být i více, např. jeden pro pozicování prvků na webu, jeden pro barevné odlišení, jeden pro určitý modul atp.

Pomocí CSS definujeme vlastnosti obecně, např. nastavení pozadí na zelenou barvu:

```
body {  
    background: green;  
}
```

S tím by si žádný kodér samozřejmě nevystačil a je tedy možné stylovat konkrétně jednotlivé prvky stejného typu – ačkoliv body je v dokumentu vždy pouze 1x, např. tag sloužící pro jakýkoliv seznam lze jednoduše nastylovat pomocí:

```
li {  
    color: red;  
}
```

Tím nastavíme všem tagům červenou barvu písma. Mimo stylování tagů můžeme také přidávat třídy. Ty mohou být společné pro více prvků, např. třída „.maly-blok“ může být společná pro více tagů, které budou mít stejné css vlastnosti. V případě jednoho unikátního prvku na stránce použijeme indentifikátor označený mřížkou - #nas-identifikator.

CSS je společně s HTML stránkou staženo k uživateli a společně tvoří statickou stránku.

3.3 PHP

Věk statických webových stránek je pryč. Po mnoho let byla síť WWW oblastí v níž několik jednoduchých, vzájemně propojených stránek HTML utvářelo webový server. Dnešní uživatelé však očekávají stránky, jež jsou průběžně aktualizovány a nabízejí vlastní zážitky. Kromě toho správci webových serverů požadují, aby byla aktualizace a údržba stránek snazší. Z těchto a mnoha dalších důvodů už je tvorba webového serveru pomocí statických souborů HTML nepřijatelná. Síť WWW je nyní místem pro dynamické, často databázemi řízené webové aplikace.

[3]

PHP je programovací jazyk, který pracuje na straně serveru. Jakmile uživatel zadá požadavek na určitou stránku, webserver nejdříve vygeneruje stránku pomocí CGI scriptu a tu jako HTML šablonu předá uživateli.

Jeho zakladatel Rasmus Lerdorf jej v roce 1994 nazval PHP jako „Personal Home Page“, později se název změnil na Hypertext Preprocessor, zkratka ovšem zůstává. V samotných začátcích PHP v roce 1994 byly aplikace psané v jazyce C. V dnešní době je jedním z nepoužívanějších jazyků pro generování dynamických stránek webu je právě PHP. Jazyk se začal utvářet v roce 1995 a vycházel

právě s C a Perlu. V květnu 1998 běželo PHP na serverech pro téměř 60 000 domén. Jednalo se tedy teprve o 2.0 verzi PHP, přesněji PHP/FI Code. Ukázka zdrojového kódu:

```
<!--include /text/header.html-->

<!--getenv HTTP_USER_AGENT-->
<!--ifsubstr $exec_result Mozilla-->
  Hey, you are using Netscape!<p>
<!--endif-->

<!--sql database select * from table where user='$username'-->
<!--ifless $numentries 1-->
  Sorry, that record does not exist<p>
<!--endif exit-->
  Welcome <!--$user-->!<p>
  You have <!--$index:0--> credits left in your account.<p>

<!--include /text/footer.html-->
```

Tento kód se dnešnímu PHP příliš nepodobá, tyto změny začali až od verze 3.0, kdy se PHP oficiálně začalo jmenovat „PHP“ jako Hypertext Preprocessor. Nyní je aktuální nejnovější verze PHP 7.2.

[2]

Kód se odděluje značkami „<?php“ a „?>“. Obsah mezi těmito značkami Apache předává ke zpracování interpreteru PHP, který je zpracuje a vrátí výstup. Ukázka jednoduchého kódu:

```
<?php
if($uzivatel == "1"){
    echo „Vítejte v systému!“;
} else {
    echo „Je mi líto, při přihlašování došlo k chybě.“;
}
?>
```

3.4 Architektonické vzory

Samotný procedurální PHP kód je pro tvorbu systému dostatečný, nicméně pokud je PHP souborů více a celkový objem kódu je také veliký, stává se zdrojový kód navzdory správnému kódu a dostupnému manuálu nepřehledný. Často se takovému kódu příznačně říká „špagety kód“. Bez správně okomentovaného kódu a manuálu lze jen velmi těžko dohledat a pochopit některé dílčí části systému. Z toho důvodu byl zaveden (a nejen v PHP, ale i v jiných programovacích jazycích jako Java, C++ atp.) objektově orientovaný přístup (OPP).

Někteří programátoři jsou striktně proti OOP, jelikož tato myšlenka jde proti principu fungování číslicových počítačů a já jsem byl velmi dlouhou dobu jedním z nich. Pravdou však zůstává, že pro tvorbu náročnějších aplikací je objektový přístup nevyhnutelný. Je třeba jistá míra abstrakce, objektový přístup je blíže ke skutečnému světu, než k principu fungování počítače. Proto je potřeba (a pro zatvrzelé procedurální programátory to není jednoduché) naučit se objektově myslet. Program musíme brát s jistým nahladem, nikoliv jako strukturovaný kód vykonávaný od prvního po poslední řádek. Důležitým prvkem je zde znovupoužitelnost. Spousta věcí lze naprogramovat univerzálně, tak abychom nemuseli psát vždy novou funkci (objekt), ale můžeme využít jednu třídu, resp. jednu členskou funkci pro více objektů (instancí).

Jelikož samotné OPP stále nemělo přesný řád jak by měla struktura aplikace vypadat, postupem času vznikli architektonické vzory. Ty nám definují jak by měla aplikace vypadat z hlediska struktury souborů a pro větší projekty rozhodně doporučuji některý arch. vzor použít. Nejpoužívanější jsou v podstatě MVC a Front Controller.

MVC

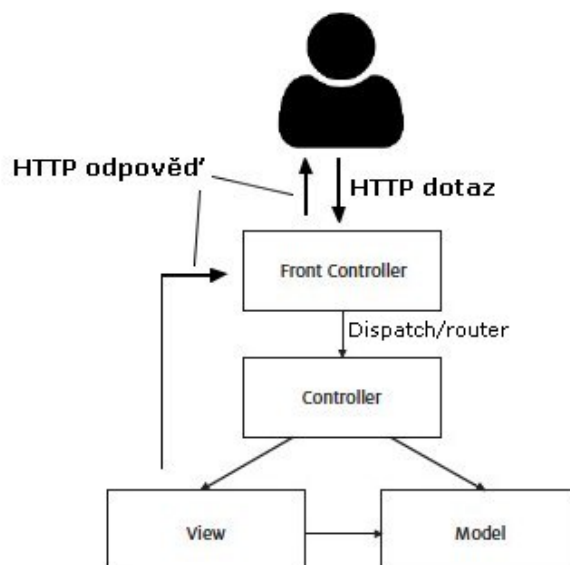
Jedná se o architekturu, neboli návrhový vzor podle kterého je možné PHP scripty členit, udělat si na serveru jakou strukturu, nebo pořádek podle jednotného stylu, tak aby každý programátor pochopil o co se jedná a věděl kde má co hledat. Zkratka MVC v sobě skrývá tři části, ze kterých je tato architektura složena. Jde o Model, View (Pohled) a Controller (Kontroler).

Model funguje jako script který provádí konkrétní úkoly – výpočty, dotazy do databáze atp. Pohled slouží ke zobrazení uživateli, jedná se tedy o jakousi šablonu. Kontroler to všechno řídí a stará se o komunikaci mezi modelem a pohledem. Zjednodušeně řečeno přijme požadavek od uživatele, pošle do kontroleru na provedení a navrácené hodnoty vypíše do šablony a příslušnou šablonu zavolá.

Front Controller

Front Controller je další typ architektury, je založený na jednom přístupovém bodu do aplikace, který obstarává předávání požadavku specifickým obsluhám, ale také autorizaci, autentizaci, nebo logování požadavků. Součástí jsou:

- Front Controller
 - obsluhuje příchozí požadavky
- Router (dispatch)
 - předává požadavky ke specifické obsluze
- View
 - zajišťuje sestavení šablony a výstup pro uživatele
- Model
 - stejně jako v případě MVC je model ta „výkonová“ část kde se zpracovávají všechny výpočty, získávají data atd.



Obrázek 6: Architektura Front Controller

3.5 SQL

Název SQL jazyka pochází z anglického Structured Query Language, čili strukturovaný jazyk dotazů. Označení je poměrně přesné, protože v SQL píšeme opravdu dotazy, pomocí kterých komunikujeme s naší databází. Pomocí dotazu můžeme jednak data z databáze číst, výsledky různě řadit, oříznout, vytvořit třeba věkový průměr z tabulky uživatelů atp. Aby jsme z databáze mohli číst, je třeba tam nějaká data zapsat a to lze také jednoduchým dotazem jazyka sql. Stejně tak dobře lze data editovat, tabulku vyprázdnit, nebo celou smazat. Jazyk SQL je tedy jakýmsi prostředníkem mezi databází a naším programem. Dotaz pro výpis všech uživatelů z tabulky uživatelé a seřazení podle příjmení by vypadal například takto:

```
SELECT * FROM uzivatele ORDER BY prijmeni
```

Jak můžeme vidět, jazyk je poměrně jednoduchý a opět vychází z angličtiny. Samozřejmě lze psát i delší a složitější dotazy, například spojovat tabulky – na základě dat z jedné tabulky vybírat data z jiné tabulky.

3.6 JAVASCRIPT

Pomocí Javascriptu lze vytvořit dynamičnost, která pracuje na straně klienta. Klient ji získá společně s HTML šablonou a není tak třeba odesílat data na server při každém požadavku, jako je to u PHP. Pomocí javascriptu se řeší části webu které není třeba nikam ukládat, ale přesto je potřeba aby byly dynamické. Například nějaký jednoduchý výpočet, změna dokumentu, nebo stylu, vyskakovací okno atp. Javascript obvykle reaguje na nějakou událost, například najetí myši na objekt, klik, opuštění objektu, rolování atp. Podobně jako CSS může být definován jako externí, interní, nebo také inline zápisem.

Scripty jsou krátké programy, nebo aplikace, které nějakým způsobem vytvářejí interaktivitu internetových stránek, nebo mohou sloužit k navigaci, či jiným složitějším operacím. Scripty lze psát pomocí dvou programovacích jazyků. Jsou to JavaScript a VBScript (Visual Basic Scripting edition). JavaScript vyvinula společnost Netscape Communications, je tedy podporován prohlížeči Internet Explorer, Netscape Navigator, Netscape Communicator. Scripty podporuje také prohlížeč Opera. Možná jste se setkali s názvem Jsript, Je to taktéž scriptovací jazyk, který je totožný s JavaScriptem a je implementován do prohlížečů Internet Explorer. VBScript vyvinula společnost Microsoft a není podporován v žádném jiném prohlížeči než v Internet Exploreru.

4. WEBDESIGN

4.1 O webdesignu

Webdesign je mezioborová disciplína, která využívá poznatky vizuální komunikace, interakčního designu, psychologie (především ovlivňování a rozhodování lidí), marketingu, branding, copywritingu, gamifikace a dalších oborů. Cílem webdesignu je vytvořit funkční webové stránky nebo webové aplikace. Funkční znamená, že splňují především následující kritéria:

- *Na web přicházejí relevantní návštěvníci*
- *Web je ovlivňuje/pomáhá jim tak, že provedou konverzní akci (akce), která je v souladu se záměrem tvůrce webu*
- *Návštěvníci se na web vrací a provádějí stejnou nebo další akce*
- *Web návštěvníky zaujme natolik, že o něm mluví sami od sebe s dalšími členy cílové skupiny webu*

[1]

Pro moderní web je zásadní aby byl intuitivní a aby uživatel vždy jednoduše našel co hledá. S tím souvisí jednoduchost – už neplatí, čím více objektů na webu bliká, jezdí a svítí, tím je web zajímavější. Důležité je, aby se uživatel na webu cítil příjemně a měl vše takřikajíc po ruce.

4.2 Základní formáty webové grafiky

Ve zkratce se pokusím popsat nejpoužívanější formáty dvojrozměrné rastrové grafiky, které je možné použít na webu.

BMP

Grafický formát BMP (BitMaP) patří v současnosti mezi nejpoužívanější grafické formáty, což je z technologického pohledu docela paradoxní, protože je poměrně složitý na zpracování a přitom nabízí pouze minimum užitečných vlastností. Oba dva dříve popisované formáty, tj. jak GIF, tak i PNG, umožňují ukládání obrázku v mnohem více variantách a přitom používají lepší komprimační schéma. Postup použitý při komprimaci v BMP je tak špatný, že se používá pouze u minima obrázků a mnoho aplikací s komprimovanými obrázky ani neumí korektně pracovat.

[5]

BMP není vhodný pro obrázky s větší barevnou hloubkou.

GIF

Gif je poměrně rozšířeným formátem pro webovou grafiku především pro svou průhlednost a možnost tzv. animovaných gifů. Používá LZW kompresi, je rychlý, dobře čitelný formát. Hlavní nevýhoda je nízká barevná hloubka – 4 až 8b barevné palety v oblasti jednoho rámce.

PNG

Dalo by se říci že formát PNG je nástupcem GIFu. Oproti GIFu má vylepšenou LZW kompresi (cca o 15%) a podporuje RGBA barevnou paletu. Na jeho kód se nevztahují žádná autorská práva. Dalšími výhodami jsou flexibilita (další rozšíření v budoucnosti) a velká robustnost (úplná kontrola integrity).

Oproti GIFu má zásadní nevýhodu – nepodporuje více než jeden obraz v jednom souboru, tudíž animace nejsou možné.

JPG

JPG je vůbec nepoužívanějším formátem webové grafiky. Spousta tvůrců webových stránek bez patřičných znalostí používá právě tento formát pro veškerou grafiku na webu.

Používá rychlý a spolehlivý komprimační způsob ve 24bitovém rozlišení (rozlišení rozpoznatelné lidským okem). Má velmi dobrý kompresní poměr, který lze nastavit. Používá ztrátovou kompresi, ale při použití hodnot mezi 70-80% komprese dosahují dobrý výsledek, víceméně nerozpoznatelných změn v obrázku, přičemž změna velikosti je patrná.

Diskrétní Kosinová transformace (dct)

Při ztrátové dochází k odstranění nepodstatných detailů, které lidské oko nedokáže rozpoznat. Podstatou DCT je nalezení korelace mezi sousedními (případně i vzdálenějšími) pixely ve dvojrozměrném objektu – v našem případě obrázku. DCT se aplikuje na čtvercové matice 8x8px, které se zpracovávají samostatně. Při kompresi se využívá toho že lidské oko není citlivé na funkce vysokých kmitočtů. Kvantizační matice nám pomocí svých hodnot ukáže kde jsou tyto frekvence nejvyšší, Dělením matic se odstraní vysoké kmitočty a zbydou nuly – tento obsah se poté zredukuje neztrátovou kompresní metodou.

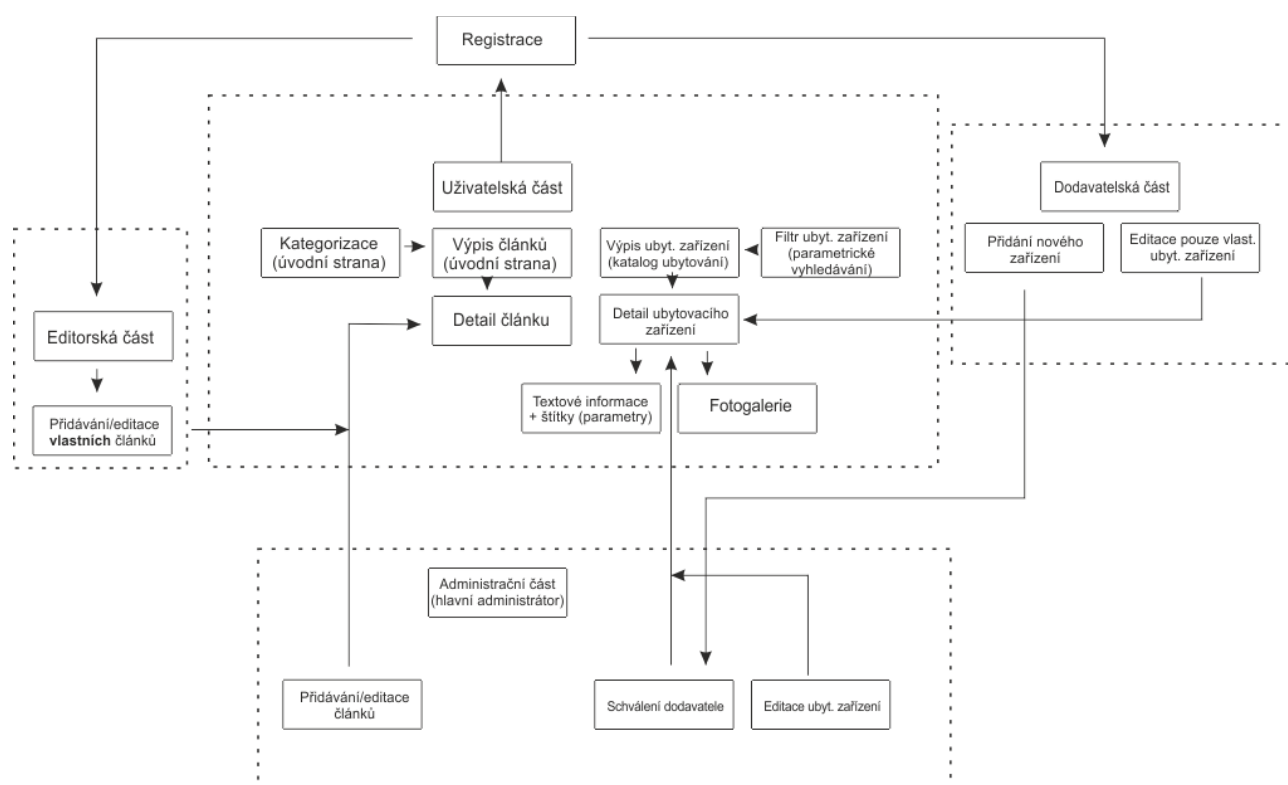
Při dekompresi se zredukovaná DCT matice opět vynásobí kvantizační maticí a pak se aplikuje I-DCT. Tím dostaneme aproximaci původního čtverečku (bloku). Míra komprese a tím taky kvalita výstupu je dána faktorem kvality (nazývá se Q-Faktor), který v podstatě stupňuje hodnoty v kvantizační matici. Funkce vyšších frekvencí obvykle nejsou tak viditelné, ovšem ostré hrany jsou často výjimkou. Tady potom dochází k nepříjemnému zvlnění (Gibbsův jev) a vůbec lehkému rozostření celého obrázku. Proto není JPEG ani žádná jiná metoda založená na DCT vhodná pro kompresi technických výkresů a prezentačních materiálů, naopak je výborná pro zpracování digitálních fotografií a videa (DCT využívají všechny MPEG kodeky).

[8]

II. PRAKTICKÁ ČÁST

1. Návrh struktury redakčního systému

Strukturu redakčního systému jsem rozdělil do čtyř částí. První z nich je frontend webu, který uvidí každý návštěvník. Další je dodavatelská část, což je sekce pro správce ubytovacích zařízení, které mohou přidat nové zařízení, nebo objekt který mají ve správě mohou editovat, či upravovat galerii. Poslední běžnou částí je editorská část, kam mají přístup „redaktoři“ našeho magazínu – uživatelé, kteří mají zájem podílet se na tvorbě obsahové části. Speciální část tvoří administrátorská část – prostředí pro superadministrátora, který může editovat články ostatních autorů a také ubytovací zařízení všech správců.



Obrázek 7: Blokové schéma navrhovaného webu

Poslední částí samotného cestovatelského magazínu jsou statické stránky, nazývané také statické články. Pro jednoduchou správu jsem se rozhodl zařadit tam články se speciální kategorií. Je tedy možné je vkládat standardně přes administraci a přístup k nim má pouze hlavní administrátor.

Články zobrazované na frontendu webu budou samozřejmě plně editovatelné z administrace (za předpokladu že jste autorem článku, nebo hlavní administrátor), vkládat lze text doplněný o případné obrázky. Ty lze nahrát přímo v editoru a ihned vložit do článku.

Samostatnou částí integrovanou do magazínu bude katalog ubytovacích zařízení. Tento modul bude součástí webu, i hlavního menu. Tato speciální sekce bude obsahovat výpis ubytovacích zařízení které je možné filtrovat. Filtrovat budeme podle běžných požadavků uživatelů:

- dle kraje
- dle ceny za osobu
- dle typu zařízení (hotel, pension, chalupa, ubytování v soukromí)

Další filtr bude poskytovat možnost volby ano/ne u následujících položek:

- pes
- bezbariérový přístup

Jako další můžeme do budoucna přidat například tyto filtry (ale samozřejmě i jiné):

- kouření povoleno
- parkování u objektu
- zábava pro děti (dětské hřiště, bazén atp.)

Filtrace probíhá dynamicky a pokud není uživatel s aktuální nabídkou spokojen, může například zvýšit maximální cenu, vynechat bezbariérový přístup atp. Filtr je k dispozici na stránce vždy nahoře nad výpisem jednotlivých ubytovacích zařízení. Ve výpisu je fotografie zařízení, krátký popis, cena a tagy, dle toho čím dané zařízení disponuje. Vpravo dole je akční tlačítko přejít na detail, které nás přeměruje na samostatnou stránku příslušného ubytovacího zařízení. Na detailu se nám informace rozšíří o kompletní popis objektu a fotogalerii. Máme také možnost hodnotit objekt, nebo shlédnout hodnocení bývalých nájemníků objektu.

2. Návrh designu

Grafický návrh pro webovou stránku jsem prováděl v programu Photofiltre, Vycházel jsem z moderních trendů webdesignu a přesto že v této oblasti nemám velkou praxi, web vyhovuje všem standardům, je použitelný, jednoduchý a v rámci možností a mého grafického cítění i hezký.

Vzhledem k tomu že web budu i kódovat, volil jsem prvky tak aby byly jednoduše realizovatelné a co nejvíce univerzální, aby bylo možné v případě potřeby některý odebrat, případně aktualizovat.

Na pozadí webu je souvislý tematický obrázek hor, který je dostatečně velký i pro HD rozlišení – 1920x1280px. Obrázek navrhuji jako statický, při rolování webem bude stále na stejném místě.

Hlavičku o výšce 280px tvoří logotyp a tři tematické fotografie z fotobanky. Působí jako prvek „chaotičnosti“, aby vše na webu nebylo rovné a uniformní. Mezi hlavičkou a obsahem působí jako předěl 60px vysoké hlavní menu kde je použito tmavší písmo k docílení efektu plastičnosti, kdy písmo vypadá jako reliéf „vyrytý“ do menu. Na pravé straně je opět prvek navíc – boty zavěšené na hřebíku, aby se na webu každý cestovatel cítil „jako doma“.

V obsahu webu je levé podružné menu kde by měli být kategorie magazínu jednak pro navigaci na webu a jednak v administraci může sloužit jako editační menu kategorií. Pravá část by měla tvořit asi 80% šířky a budou v ní články magazínu, statické články, filtrace katalogu a výpis ubytovacích zařízení, případně jejich detail. V případě specifických podstránek jako registrace ubytovacího zařízení, nebo filtr katalogu navrhuji tmavé průhledné pozadí a bílý text.

Závěrem je na webu jednoduchá patička ve stejném stylu jako horní menu, s tím že u písma už není plastický efekt, písmo je standardní, černé barvy zarovnané na střed stránky. Patička je přes celou šířku stránky a uzavírá tak celý web.

Co se týče typů obrázků, pro webové prostředí jsou nejpoužívanější formáty které používám i já JPG a PNG. Oba formáty mají své opodstatnění. JPG jsem použil pro celistvé plochy vč. pozadí, které mají podobné pixely. Používá totiž ztrátovou kompresi – Diskrétní kosinovu transformaci (DCT). Metoda spočívá v nalezení podobnosti (korelace) zejména mezi sousedními, nebo nejbližšími pixely. Tyto pixely jsou sloučeny do takové míry že lidské oko nepozná rozdíl a velikost souboru je podstatně snížena.

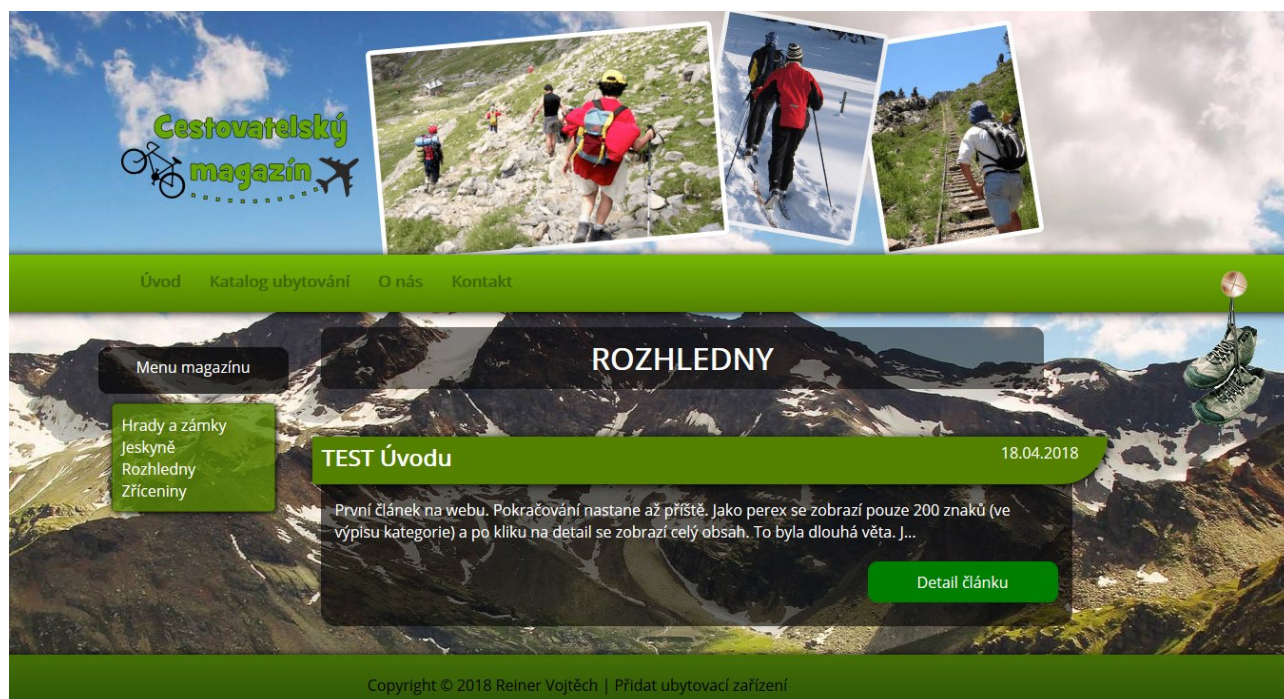
Z toho důvodu navrhuji použití JPG formátu jednoznačně pro obrázek na pozadí, i fotografie v hlavičce. Pro logotyp naopak doporučuji použití PNG vzhledem k možnosti průhledného pozadí. Pro běžné použití v hlavičce, případně i na fakturách atp. stačí velikost 250x100px, což má orientačně 20kB. Posledními grafickými prvky na stránce jsou šipky a navigační tlačítka v galerii (v pop-up okně při kliku na obrázek galerie), které jsou kvůli průhlednosti ve formátu GIF a mají velikost v řádu jednotek kB, čili jsou zanedbatelné.

Orientační seznam velikostí grafických prvků na stránce:

- Pozadí: 401kB
- Logotyp 22kB
- Fotografie v hlavičce 105kB
- Obrázek „boty“ 23kB

Čili celková velikost statických grafických prvků na webu je 551kB, což považuji za optimální.

2.1 Výsledný návrh



Obrázek 8: Výsledný návrh stránek pro cestovatelský magazín

Cestovatelský magazín je sám o sobě z pohledu systému velmi podobný všem ostatním internetovým magazínům na jiná témata. Na úvodní stránce bude výpis nejnovějších článků

2.2 Vytvoření šablony

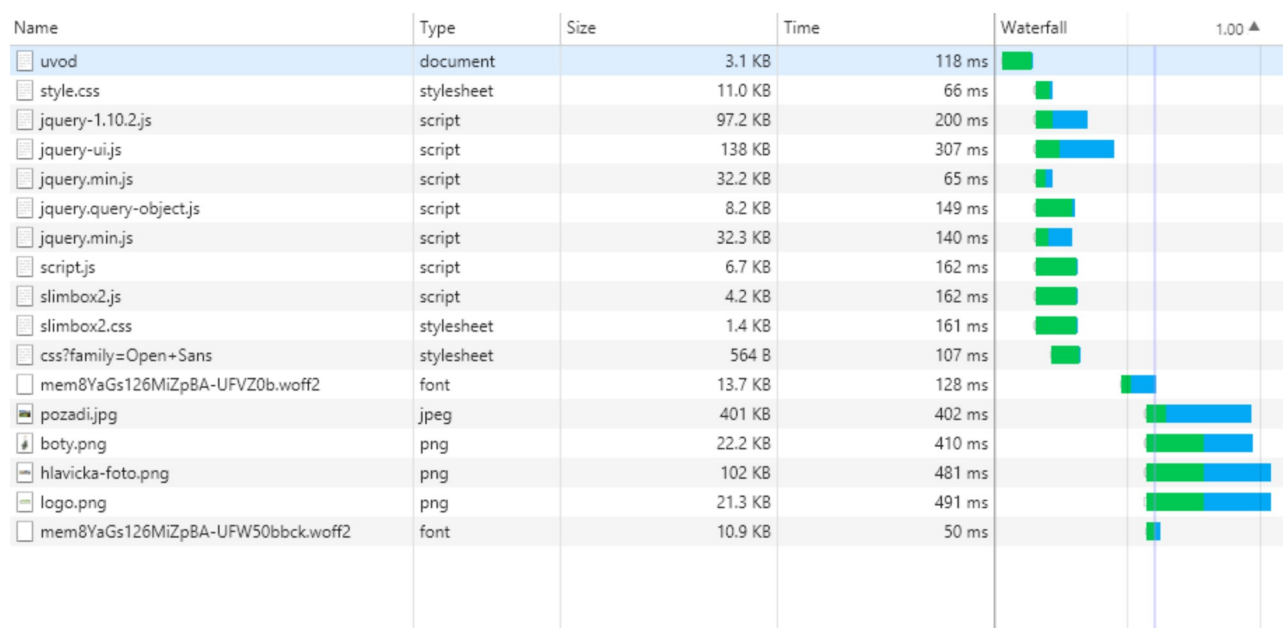
Šablona je vytvořena za pomoci HTML v.5 a CSS v.3. Web je optimalizován pro šířku 1024px a vyšší. Z toho vychází základní blok o velikosti 980px, který je centrován na střed. Snažil jsem se dodržet čistý a přehledný kód.

Pro nadpisy používám implicitní tagy <h1>, <h2> atp. kde je nadpis první úrovně vyhrazen pro název webu v hlavičce a nadpisy druhé úrovně jsou jako nadpisy článků/názvy objektů. Pro menu, jelikož jde o seznam položek používám tag pro seznam , pro větší bloky, jako například výpis článků, nebo ubytovacích zařízení (objektů) používám tag <div> a pro menší textové části tag . Inline stylům jsem se nažil vyvarovat, všechny CSS jsou v jednom externím souboru (css/style.css) a druhý doplňkový CSS soubor je pro „slimbox“, který slouží ke stylování obrázků při rozkliknutí náhledu (v článku, v galerii).

2.3 Rychlost načítání

Celkem se k načtení webu stahuje 17 souborů:

- 1 dokument
- 3 kaskádové styly (jeden hlavní, jeden pro prohlížeč obrázků a jeden pro písmo)
- 7 javascriptů (jeden hlavní a ostatní pro jQuery)
- 4 obrázky
- 2 soubory s použitým fontem (písmem)

Výsledek mého subjektivního testu:

Obrázek 9: Výsledek testu měření rychlosti načítání webu

Celá úvodní stránka je načtena za cca 1,2s. Při opakovaném načtení se doba načítání pohybuje kolem 0,8s. Tyto výsledky jsou pro web tohoto typu naprosto dostačující a možná i mírně nadprůměrné.

3. Redakční systém

Redakční systém (dále jen RS), označovaný také publikační systém, nebo CMS (z anglického content management systém) slouží k řízení celého webu. Bez něj by byla stránka pouze statická, fungovala by pouze pro čtení, nemohla by mít žádnou zpětnou vazbu od uživatelů, ukládat data do databáze, pamatovat si oblíbené destinace uživatelů atp.

Bez RS by také nebylo možné přidávání nových ubytovacích zařízení samotnými majiteli/provozovateli, musel by to dělat administrátor, tzv. webmaster za pomoci značkovacího jazyka HTML, což je značně nepohodlné a při větším množství časově náročné a nepřehledné.

Pro jednoduchost jsem se rozhodl redakční systém nestavět na žádném modelu, RS nebude využíván pro jiné weby jako většina univerzálních CMS a má své specifické funkce. Z toho důvodu je naprogramován na míru pro tento webový portál. Na webu bylo potřeba programovat spoustu specifických věcí a jejich implementace do hotového CMS by vyžadovala dobré znalosti tohoto konkrétního systému.

3.1 Frontend

Na frontendu (nebo-li uživatelské části webu, viditelné bez registrace/přihlášení) systém vypisuje články, kategorie a ubytovací zařízení. Dle MVC architektury vše řídí směřovač „SmerovacKontroler“ a všechny kontrolery pro dílčí části webu dědí z třídy „Kontroler“, která obsahuje univerzální funkce, jako ověření uživatele, funkce pro zjištění práv (je-li uživatel administrátor, správce atp.), funkce pro zjištění čísla stránky atp.

Kategorie a články

Univerzální kontroler je „KategorieKontroler.php“, který je načten v případě že neexistuje kontroler daného jména volaného jako první parametr v URL. Čili pokud se pokusíme zobrazit adresu `www.mvc.webstav.com/neco` a kontroler „NecoKontroler.php“ neexistuje, je načten tento univerzální – kontroler kategorií (resp. článků).

Kontroler kategorií pomocí prvního parametru v URL vyhledá v databázi příslušnou kategorii a pokud se taková v databázi nenachází, vypíše chybu, protože stránka neexistuje (ani jako kontroler, ani jako kategorie v databázi). Když se jedná o kategorii, vypíše se seznam článků dané kategorie. Pokud URL obsahuje i druhý parametr – název článku, zobrazíme detail článku. V kontroleru kategorií je ošetřeno také to, že by uživatel mohl zadat jinou kategorii a stejný název článku. Např. originální url bude `hrady/Rabstejn/` a uživatel by mohl zadat `jeskyne/Rabstejn/`. Proto systém kontroluje i to, zda se snažíme vypsát článek ve správné kategorii. Kdyby to ošetřeno nebylo, vznikl by tak vysoký počet (celkový počet kategorií mínus 1) duplicitních článků.

Ostatní informační položky v menu – jako prozatímní články o nás, kontakt jsou ve speciální kategorii `clanek`.

Katalog ubytování

Kontroler pro katalog ubytování vrací seznam ubytovacích zařízení a v záhlaví filtrační okno. Při kliku na některý z filtrů se pomocí javascriptu změní URL adresa – zapíše se do ní filtrační parametr, případně se odstraní. Současně se obnoví stránka a PHP vygeneruje nový seznam objektů. Zvažoval jsem možnost použití AJAXu, kdy by se obsah měnil bez nutnosti obnovení stránky, ale nakonec jsem zvolil tuto logičtější variantu. Uživatelé si tak odkaz s konkrétním filtrem mohou uložit, odeslat známému atp. I z hlediska systému je tato varianta rozumnější, musel bych vytvářet pomocný soubor, který by data ajaxu předával.

Na detailu ubytovacího zařízení je mimo popisu a fotogalerie také sekce pro recenze daného objektu. Recenze smí přidávat pouze registrovaní uživatelé – je to z toho důvodu, aby recenzi nemohl přidat každý a zneužívat tak hodnocení ve prospěch objektu, ale i naopak konkurence pro jeho poškzení. Není to samozřejmě 100% řešení, ale určitě to eliminuje velké množství případů. Takto s emailem jde v případě nejasností recenzi napadnout a prostřednictvím hlavního administrátora požádat o smazání, nebo otevření sporu. Díky emailu použitému v registraci lze hodnotitele dohledat a vyzvat k vysvětlení, resp. dialogu s provozovatelem objektu.

3.2 Administrační rozhraní (backend)

Administrační rozhraní pro správce webu slouží k:

- správě článků (přidání nového článku, editaci, smazání)
- správě novinek (přidání novinky, editace a smazání)
- správě statických článků – jedná se zejména o články v menu jako kontakt, o nás atp.

Pro část webu „katalog ubytovacích zařízení“ slouží administrace ke schvalování administrátorem a případně editaci uvedených údajů o konkrétním ubytovacím zařízení jeho majitelem (správcem).

Uživatel, který získá přístup do administrace může mít některou z následujících rolí:

Správce ubytovacího zařízení

- Tuto roli uživatel získá po přidání ubytovacího zařízení, pomocí svého účtu má přístup k editaci zařízení a jeho fotogalerie

Administrátor (editor)

- Uživatelé, kteří získají tuto roli smí přidávat nové články a editovat své starší články. Jedná se o články přidávané do obsahové části webu (do magazínu)

Superadministrátor

Roli hlavního administrátora má přiřazený pouze úzký okruh uživatelů, kteří zodpovídají za správný chod webu – schvalují nové objekty, udělují administrátorská práva a mají přístup k editaci všech článků na webu.

Editaci objektů přímo superadministrátorem nepovažuji za důležitou z toho důvodu, že informace, resp. Podstránka každého ubytovacího zařízení by měla být vizitkou každého správce objektu a hlavní administrátor by měl mít možnost objekt zneveřejnit, ale nikoliv měnit fotogalerii, nebo znění jejich textů.

Mazání článků je povoleno pouze superadministrátorovi a tlačítko pro smazání se nachází v administraci, ve výpisu článků.

Původně jsem chtěl udělat administraci katalogu oddělenou od frontendu, nicméně nakonec jsem usoudil že lepší varianta pro schvalování a mazání katalogu bude přímo na frontendu s tím, že se administrátorovi zobrazí tlačítka pro obsluhu. Upravil jsem tedy funkci pro výpis objektů, tak aby se zobrazovali i ty neschválené (na frontendu se zobrazují pouze ty, které mají v databázi ve sloupci schváleno 1) a přidal funkci pro mazání a změnu stavu.

4. Databáze

Níže uvádím jednotlivé tabulky databáze s popisem jejich využití.

Články

#	Název	Typ	Porovnávání	Vlastnosti	Nulový	Výchozí	Komentáře	Další
<input type="checkbox"/> 1	clanky_id 	int(11)			Ne	Žádná		AUTO_INCREMENT
<input type="checkbox"/> 2	nazev	varchar(255)	utf8_czech_ci		Ano	NULL		
<input type="checkbox"/> 3	obsah	text	utf8_czech_ci		Ano	NULL		
<input type="checkbox"/> 4	url	varchar(255)	utf8_czech_ci		Ano	NULL		
<input type="checkbox"/> 5	popisek	varchar(255)	utf8_czech_ci		Ano	NULL		
<input type="checkbox"/> 6	id_kategorie	int(11)			Ne	Žádná		
<input type="checkbox"/> 7	autor_id	int(11)			Ne	Žádná		
<input type="checkbox"/> 8	zobrazit	tinyint(1)			Ne	0		

Obrázek 10: Databázová tabulka "clanky"

Každý článek obsahuje unikátní identifikátor, který zároveň slouží jako primární klíč. Identifikátor se generuje v databázi automaticky dle posledního vytvořeného záznamu.

Název článku je typu varchar s omezením na 255 znaků, protože názvy mohou být různě dlouhé. Datový typ varchar je použitelný pro kratší textové záznamy vzhledem k rychlejšímu vyhledávání – tento datový typ je uložený přímo ve sloupci tabulky.

Obsah bude samozřejmě delší než 255 znaků, proto jsem použil dat. typ „text“, který ve sloupci tabulky ve skutečnosti obsahuje pouze ukazatel na skutečné místo uložení. Limit 65 535 znaků je plně dostačující. Do tohoto limitu se započítává nejen čistý text, ale i formátovací značky, případně obrázky atp.

URL bude víceméně vycházet z názvu, proto také raději volím max. délku 255 znaků. Popisek je krátký text, který se zobrazí ve výpisu článků, tzv. perex, pro ten bude délka dostačující. Současně slouží jako obsah pro meta tag description, s tím že je oříznutý na 150 znaků a přidány tři tečky, jako pokračování.

Id_kategorie je navázána na ID kategorie v tabulce kategorií. Typ databáze který používám neumožňuje relace mezi tabulkami (nejedná se o relační db, není možné vytvořit cizí klíč), takže propojení je zajištěno pouze v PHP. Id autora článku je podobně navázáno na tabulku uživatele a slouží k jasné identifikaci autora daného článku.

Poslední položka „zobrazit“ je typu „tinyint“ o velikosti 1. Chtěl jsem použít datový typ boolean, ale podle dokumentace MYSQL se jedná o ekvivalent datového typu boolean a ten jako takový použít nelze. Při pokusu o jeho vytvoření se automaticky vytvoří datový typ tinyint. Tato položka bude nabývat hodnot 0/1, dle toho zda je článek schválený hlavním administrátorem, či nikoliv. Je to z důvodu kontroly nevhodného obsahu článků, zejména u nových autorů.

Aktuálně je schvalování vypnuté a zveřejňovány jsou všechny články. Plánovaná funkčnost je na manuálním principu, ale bylo by možné nastavit v tabulce uživatelů sloupec „ověřený redaktor“, na základě kterého by PHP script automaticky v této tabulce (články) označil článek jako schválený. Případně by se to mohlo posoudit podle množství schválených článků daného autora (pokud má více než 10 schválených → jsou jeho články automaticky schvalovány). Sloupec současně slouží pro zneveřejnění článku v případě potřeby (neaktuální informace, nutnost úpravy, zveřejnění až později, atp.). Možností je více, rozhodnutí bych nechal až na ostrý provoz webu, třeba schvalování vůbec nebude potřeba a bude stačit pouze odebrat práva nevyhovujícím autorům.

Kategorie

#	Název	Typ	Porovnávání	Vlastnosti	Nulový	Výchozí	Komentáře	Další
<input type="checkbox"/> 1	id	int(11)			Ne	Žádná		AUTO_INCREMENT
<input type="checkbox"/> 2	nazev	varchar(50)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 3	popis	text	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 4	url	varchar(50)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 5	description	varchar(250)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 6	zobrazit	tinyint(1)			Ne	0		

Obrázek 11: Databázová tabulka "kategorie"

Oproti článkům tabulka kategorií obsahuje navíc sloupec „description“, který v systému používám pro automatické vyplnění meta tagu description v každé kategorii. Jedním důvodem je že by tento tag nebylo možné z ničeho jiného generovat a druhým důvodem že kategorie bude mít ve správě hlavní správce webu a nikoli „laický“ pisatel, je možné tento vyplnit tak aby byl optimální v rámci SEO.

Katalog

#	Název	Typ	Porovnávání	Vlastnosti	Nulový	Výchozí	Komentáře	Další
<input type="checkbox"/> 1	id	int(11)			Ne	Žádná		AUTO_INCREMENT
<input type="checkbox"/> 2	nazev	varchar(200)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 3	kratky_popis	varchar(500)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 4	popis	varchar(1000)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 5	typ	varchar(30)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 6	kraj	tinyint(1)			Ne	Žádná		
<input type="checkbox"/> 7	cena_luzko	int(10)			Ne	Žádná		
<input type="checkbox"/> 8	pes	tinyint(1)			Ne	Žádná		
<input type="checkbox"/> 9	bez_barier	tinyint(1)			Ne	Žádná		
<input type="checkbox"/> 10	kontakt_osoba	varchar(200)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 11	email	varchar(200)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 12	telefon	varchar(9)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 13	datum_vytvoreni	datetime			Ne	CURRENT_TIMESTAMP		
<input type="checkbox"/> 14	url	varchar(300)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 15	foto_url	varchar(50)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 16	schvaleno	tinyint(1)			Ne	Žádná		

Obrázek 12: Databázová tabulka "katalog"

Sloupec ID slouží opět jako unikátní identifikátor každého ubytovacího zařízení, hodnota je vždy automaticky inkrementována. Následují údaje o objektu, u položek „pes“ a „bez_barier“ jsem chtěl jsem použít datový typ boolean, ale používám tinyint(1) - dle dokumentace MYSQL se jedná o ekvivalent datového typu boolean a ten jako takový použít nelze. Sloupec „foto_url“ slouží pro hlavní fotografii objektu. Ostatní fotografie jsou uloženy v tabulce „galerie“, viz. níže.

Galerie

#	Název	Typ	Porovnávání	Vlastnosti	Nulový	Výchozí	Komentáře	Další
<input type="checkbox"/> 1	id 	int(11)			Ne	Žádná		AUTO_INCREMENT
<input type="checkbox"/> 2	id_katalog	int(11)			Ne	Žádná		
<input type="checkbox"/> 3	url	varchar(20)	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 4	razeni	smallint(3)			Ne	Žádná		

Obrázek 13: Databázová tabulka "galerie"

Tato tabulka je (na úrovni PHP) vázána na tabulku katalogu přes sloupec „id_katalog“ - tím je dané k jakému objektu fotografie patří. Unikátnost názvu souboru je zajištěna přidáním posledního ID položky z galerie u daného objektu. Sloupec řazení je číselný a slouží pro řazení fotografií v galerii, které je možné provádět jako správce z administrační části webu.

Uživatelé

#	Název	Typ	Porovnávání	Vlastnosti	Nulový	Výchozí	Komentáře	Další
<input type="checkbox"/> 1	uzivatele_id 	int(11)			Ne	Žádná		AUTO_INCREMENT
<input type="checkbox"/> 2	email 	varchar(255)	utf8_czech_ci		Ano	NULL		
<input type="checkbox"/> 3	heslo	varchar(255)	utf8_czech_ci		Ano	NULL		
<input type="checkbox"/> 4	spravce	tinyint(1)			Ne	0		
<input type="checkbox"/> 5	admin	tinyint(1)			Ne	0		
<input type="checkbox"/> 6	superadmin	tinyint(1)			Ne	0		
<input type="checkbox"/> 7	overeno	tinyint(1)			Ne	Žádná		
<input type="checkbox"/> 8	overovací_kod	char(12)	utf8_czech_ci		Ne	Žádná		

Obrázek 14: Databázová tabulka "uzivatele"

Mimo klasické údaje jako email, heslo obsahuje tabulka informace o tom, zda je uživatel správce, administrátor, nebo superadministrátor. Podle toho má přístup do různých administračních částí. Ověřovací kód slouží jako dočasná položka při ověřování uživatele, případně jako kontrolní kód při ztrátě hesla. Sloupec ověřeno je kontrolován u autorů článků, ověření probíhá automaticky zasláním kontrolního – ověřovacího odkazu na zadaný email.

Recenze

#	Název	Typ	Porovnávání	Vlastnosti	Nulový	Výchozí	Komentáře	Další
<input type="checkbox"/> 1	id 	int(11)			Ne	Žádná		AUTO_INCREMENT
<input type="checkbox"/> 2	id_katalog	int(11)			Ne	Žádná		
<input type="checkbox"/> 3	id_uzivatele	int(11)			Ne	Žádná		
<input type="checkbox"/> 4	text	text	utf8_czech_ci		Ne	Žádná		
<input type="checkbox"/> 5	body	tinyint(1)			Ne	Žádná		
<input type="checkbox"/> 6	datum	datetime			Ne	CURRENT_TIMESTAMP		

Obrázek 15: Databázová tabulka "recenze"

Vazby v databázi

Jelikož v databázi typu MyISAM, kterou používám není možné vytvořit cizí klíč a vytvořit tak vztahy mezi tabulkami (na rozdíl třeba od InnoDB), tyto relace řeším pouze na úrovni PHP, kde propojují jednotlivé tabulky zejména přes jejich identifikátory.

5. Zabezpečení

XSS – z anglického Cross-site scripting je metoda, která využívá neošetřených vstupů k vložení vlastního škodlivého kódu. Ten může mít za cíl třeba podstrčení formuláře vyžadující heslo, které se následně odešle útočnickovi, může zjistit cookies uživatele atp. Nejjednodušším ošetřením vstupů je funkce PHP „htmlspecialchars()“, která tagy převede na entity, které prohlížeč neprovede a kód tak není nebezpečný. U funkce je vhodné použít ještě parametr „ENT_QUOTES“, který vytvoří entitu i z jednoduchých uvozovek - „'“.

Jelikož je v systému spousta výstupů, bylo by poněkud nepřehledné každý potřebný výpis ošetřovat touto funkcí. Navíc by se na ošetření mohlo lehce zapomenout. Z toho důvodu v hlavním kontroleru používám vlastní funkci, která každé pole (jeho řetězce), které slouží k výstupu k uživateli ošetří pomocí entit – pomocí funkce htmlspecialchars. Když pole obsahuje další vnitřní pole, ošetří se rovněž.

SQL Injection – u této metody se jedná o rozbití dotazu do databáze a vložení vlastního, potenciálně škodlivého dotazu – například vymazání všech dat, editace atp. V klasickém dotazu lze dotaz zabezpečit funkcí pro zaescapování škodlivých znaků. Já v systému využívám tzv. Prepared statements, kde nevkládáme surová data přímo, ale pomocí zástupného znaku „?“ . Při provádění dotazu se namísto zástupných znaků dosadí hodnoty z pole parametrů a nelze tak ovlivnit samotný dotaz.

Brutal force attack – proti tomuto útoku jsem zavedl jednoduchou otázku při registraci. V případě potřeby ji lze vyměnit za sadu náhodně generovaných otázek, případně klasickou captchu

Ověření uživatelů

Projekt je navržen tak že se do systému může zaregistrovat každý, tedy teoreticky i spamový robot. Mimo jednoduché otázky web používá i ověření uživatele pomocí emailu. Při registraci systém vygeneruje speciální odkaz, který odešle na zadaný email a až po kliku na něj v emailovém klientovi dojde k ověření uživatele a zpřístupnění funkcí.

Je to výhodné z hlediska registrace robotů, ale i zneužití cizího emailu, případně proti variantě kdy si uživatel email jednoduše vymyslí.

Zabezpečení hesla

Používané hashovací algoritmy (md5, sha), lze snadno rozluštit porovnáním, pomocí slovníkové metody atp.

Použití šifrování namísto hashování je vyloučené, heslo, případě jiný údaj který potřebujeme zabezpečit nesmí znát ani provozovatel aplikace, nebo útočník, který nějakým způsobem databází získá. Provozovatel má přístup k algoritmům a mohl by si heslo zpětně dešifrovat, nebo toho jinak zneužít. Jejich nevýhoda je v tom, že jsou příliš rychlé.

Víme už tedy, že hesla musíme hašovat a že k nim musíme přiložit náhodnou sůl. Zbývá výběr hašovací funkce. Běžné hašovací funkce jako MD5, SHA-1 nebo SHA-2 jsou pro ukládání hesel nevhodné proto, že jsou příliš rychlé. MD5 hesla složeného z osmi malých písmen prolomí oclHashcat na celkem běžné grafické kartě průměrně za 10 sekund ($26^8 / 10.742e9 / 2$)! Pokud požadavek na heslo zpřísníme třeba tím, že budeme vyžadovat použití i velkých písmen a čísel, tak řada uživatelů dá velké písmeno na začátek a číslo přidá na konec. Takto zkonstruované heslo útočník prolomí dokonce za 4 sekundy. A to vůbec nemluvíme o použití slovníků nebo hesel získaných z dříve uniknutých databází.

[6]

Za bezpečné hashování hesla považuji `crypt()` a `password_hash()`. Vzhledem k tomu že jsem v nedávné době aktualizoval verzi PHP na serveru na 5.6, mám k dispozici `password_hash()` (od verze PHP 5.5). Tato funkce používá algoritmus BCrypt a automaticky k heslu generuje „sůl“ - pseudo-náhodný řetězec přidaný k heslu.

Funkce `password_hash()` je dostupná až od verze PHP 5.5, kterou na serveru prozatím nemám, z tohoto důvodu použiji funkci `crypt()`.

Pokud bych měl starší verzi PHP, použil bych funkci `crypt()`, nicméně je zde ještě jeden problém – starší verze PHP (<5.6) neobsahuje funkci `hash_equals`, museli by jsme ji nahradit:

```
if(!function_exists('hash_equals'))
{
    function hash_equals($str1, $str2)
    {
        if(strlen($str1) != strlen($str2))
        {
            return false;
        }
        else
        {
            $res = $str1 ^ $str2;
            $ret = 0;
            for($i = strlen($res) - 1; $i >= 0; $i--)
            {
                $ret |= ord($res[$i]);
            }
            return !$ret;
        }
    }
}
```

HTTPS - SSL certifikát

SSL (secure socket layer) není zabezpečením samotného systému, ale celé webové stránky. Jedná se o šifrování komunikace mezi webovým (případně emailovým) klientem a serverem. Pokud má daný web SSL certifikát, máme jistotu že skutečně odesíláme data (hesla, osobní informace) na web na který je poslat chceme. Další věcí je šifrování během přenosu – pokud by se podařilo útočníkovi odposlednout data někde „po cestě“, získá pouze šifrovaná data. Bez soukromého klíče je nedokáže dešifrovat.

Certifikátů existuje celá řada. Od komerčních, až po ty poskytované zdarma. Já jsem zvolil Let's encrypt, který je k dispozici zdarma a pro účely magazínu plně dostačuje.

RegistraceKontroler

U tohoto kontroleru není ošetřen vstup z metody post, proto ošetřuji „ručně“ pomocí funkcí: htmlspecialchars() a trim().

Ukázka použití:

```
htmlspecialchars(trim($_POST['email']));
```

Ošetřit je nutné pouze vstupní proměnnou pro email. S ostatními proměnnými pracujeme v členské funkci „registruj“, kde je SQL Injection ošetřená způsobem popsáným výše v této kapitole.

Zapomenuté heslo

V případě že uživatel zapomene heslo, je v navigaci, která je umístěna v patičce odkaz na stránku s možností nastavení nového hesla. Uživatel zadá svůj email na který je zaslán potvrzovací odkaz. Po jeho rozkliknutí je přesměrován opět na stránku zapomenutého hesla a v případě že je odkaz správný je zobrazena možnost nastavení nového hesla.

Bezpečnostní kód je složen z 11 znaků a může obsahovat písmena anglické abecedy – malá i velká + číslice. To znamená že máme pole 62 znaků, což je:

$62^{11} = 751 \cdot 10^{16}$, čili téměř 10000000 miliard možností. Takový bezpečnostní kód považuji za dostatečně dlouhý a víceméně neprolomitelný hard force útokem.

ZÁVĚR

Cílem bakalářské práce bylo vytvořit funkční stránky pro cestovatelský magazín s katalogem ubytovacích zařízení. V teoretické části jsem se zabýval rozбором technologií, potřebného programového vybavení, možnostmi vývoje, teoretickou částí webdesignu a také rozбором aktuálních webových serverů zaměřených na cestování.

V druhé – praktické části jsem popsal vytvoření grafického návrhu a následné šablony webové stránky. Dále jsem se zabýval popisem vývoje systému na MVC architektuře, návrhem databázových tabulek a jako poslední bod bezpečností aplikace.

K vytvoření webu jsem použil zejména technologií HTML, CSS, JavaScriptu a pro systémovou část PHP + pro obsluhu databáze jazyk SQL. Web jsem spustil na reálném webhostingu a otestoval z hlediska rychlosti načítání – výsledky byly naprosto v pořádku. Magazín i s katalogem ubytování je tedy plně funkční a přístupný odkudkoliv kde je k dispozici připojení k internetu. Aktuální pracovní adresa webu je <https://mvc.webstav.com>

SEZNAM POUŽITÉ LITERATURY

- [1] Interval: Svět Internetu, Technologií a Bezpečnosti [online]. [Cit. 12.4.2018]. Dostupné z: <https://www.interval.cz/>
- [2] PHP: Manuál [online]. [Cit. 18.4.2018]. Dostupné z: <http://php.net/manual/en/history.php.php>
- [3] Ulman Larry, PHP a MySQL, Computer Press, Brno , 2004. ISBN 80-251-0063-4
- [4] Stejskal Jan, Vytváříme WWW stránky pomocí HTML, CSS a Javascriptu, Computer Press, a.s., Brno , 2006. ISBN 80-251-0167-3
- [5] Root: Informace nejen ze světa Linuxu [online]. [Cit. 24.4.2018]. Dostupné z: <https://www.root.cz/>
- [6] PHP triky: Weblog o elegantním programování v PHP pro mírně pokročilé [online]. [Cit. 2.5.2018]. Dostupné z: <https://php.vrana.cz/>
- [7] Slunecnice.cz: Programy rychle a zadarmo [online]. [Cit. 12.5.2018]. Dostupné z: <https://www.slunecnice.cz/>
- [8] Libca.cz: [online]. [Cit. 14.5.2018]. Dostupné z: <http://libca.cz/>
- [9] Marek Prokop: CSS kaskádové styly pro webdesignéry (2. vydání), Computer Press, Praha 2006
- [10] Mach J.: PHP pro úplné začátečníky, Computer Press, Praha 2002
- [11] Jiří Kosek: PHP – tvorba interaktivních internetových aplikací, Grada Publishing, 1999
- [12] W. Jason Gilmore: Velká kniha PHP 5 a MySQL, Zoner Press, 2011
- [13] David LeBlanc, Michael Howard: Bezpečný kód, Computer Press, 2008
- [14] Travelportal.cz: Magazín – Celý svět na jedné adrese [online]. [Cit. 12.5.2018]. Dostupné z: magazin.travelportal.cz
- [15] Magazín-cestovani.cz: Magazín o cestování [online]. [Cit. 12.5.2018]. Dostupné z: <https://magazin-cestovani.cz/>
- [16] Tipnavylet.cz: Tip na výlet - cestování, zábava, poznání [online]. [Cit. 12.5.2018]. Dostupné z: <http://www.tipnavylet.cz>
- [17] Zamky-hrady.cz: Česká republika - Zámky, hrady a jiné zajímavosti [online]. [Cit. 12.5.2018]. Dostupné z: <http://www.zamky-hrady.cz/>

SEZNAM POUŽITÝCH ZKRATEK A POJMŮ

URL	Uniform Resource Locator
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
PHP	HyperText Preprocesor
SQL	Structured Query Language
OPP	Objektově orientované programování
MVC	Model-view-controller (architektonický vzor)
Frontend	Část webu, kterou vidí každý návštěvník stránek
Backend	Část webu, která je viditelná po přihlášení
Perex	Úvodní část článku (náhled článku)

SEZNAM OBRÁZKŮ

Obrázek 1: Náhled webu magazin.travelportal.cz.....	12
Obrázek 2: Náhled webu magazin-cestovani.cz.....	13
Obrázek 3: Náhled webu tipnavylet.cz.....	14
Obrázek 4: Náhled webu zamky-hrady.cz.....	15
Obrázek 5: Ukázka prostředí phpMyAdmin.....	19
Obrázek 6: Architektura Front Controller.....	24
Obrázek 7: Blokové schéma navrhovaného webu.....	30
Obrázek 8: Výsledný návrh stránek pro cestovatelský magazín.....	33
Obrázek 9: Výsledek testu měření rychlosti načítání webu.....	34
Obrázek 10: Databázová tabulka "clanky".....	38
Obrázek 11: Databázová tabulka "kategorie".....	40
Obrázek 12: Databázová tabulka "katalog".....	40
Obrázek 13: Databázová tabulka "galerie".....	41
Obrázek 14: Databázová tabulka "uzivatele".....	41
Obrázek 15: Databázová tabulka "recenze".....	42

SEZNAM PŘÍLOH

Příloha 1: CD se zdrojovými kódy webové stránky + kódem (dotazem) pro naplnění databáze