

**Západočeská univerzita v Plzni**  
**Fakulta aplikovaných věd**

Diplomová práce

Zpravodajský server o jezdeckví

Bc. Jiří Kubák

vedoucí: Ing. Ondřej Rohlík, Ph.D.

Plzeň 2013

## **Anotace**

Diplomová práce obsahuje kompletní návrh zpravodajského serveru o jezdeckví, jehož realizace je předlohou dalších částí práce, které jsou vytvořeny za cílem elektronické formy vzdělávání. Závěrem se práce věnuje reálnému provozu zpravodajského serveru o jezdeckví a úvodu do analytických nástrojů pro sledování webového projektu od společnosti Google.

## **Abstract**

The diploma thesis contains a complete design of horse riding news site, which implementation is a pattern for next parts of thesis, which are created for the purpose of e-learning. Final part of thesis deals with real working of the news site and with an introduction to analytical tools for web project monitoring from Google.

## **Prohlášení**

Prohlašuji, že svoji diplomovou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47 b zákona č. 111/1998 Sb. V platném znění souhlasím se zveřejněním své diplomové práce, a to v nezkrácené podobě – v úpravě vzniklé vypuštěním vyznačených částí archivovaných Fakultou aplikovaných věd - elektronickou cestou ve veřejně přístupné části databáze STAG provozované Západočeskou univerzitou v Plzni na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Plzni 14. srpna 2013

.....  
Jiří Kubák

## **Poděkování**

Děkuji vedoucímu práce Ing. Ondřeji Rohlíkovi, Ph.D. Za cenné připomínky při realizaci práce.

# Obsah

1. Úvod.....	8
2. Obecná východiska .....	9
2.1. Web a webové aplikace.....	9
2.1.1. Historie.....	9
2.1.2. Web 2.0.....	10
2.1.3. Webové aplikace.....	11
2.2. Návrh webových aplikací.....	12
2.2.1. Návrh architektury webu .....	12
2.2.2. Programování webových stránek.....	13
2.2.3. Grafický design.....	13
2.2.4. Schválení a uvedení webových stránek do provozu .....	13
2.2.5. Údržba.....	14
2.3. Počítačem podporovaná výuka .....	14
2.3.1. Historie.....	14
2.3.2. Výukové činnosti .....	15
2.3.3. Výhody a nevýhody e-learningu.....	15
2.3.4. Konstruktivismus a E-learning .....	16
2.4. Technologie pro tvorbu webových aplikací.....	17
2.4.1. HTML .....	17
2.4.2. CSS .....	17
2.4.3. JavaScript.....	17
2.4.4. PHP .....	18
2.4.5. ASP.NET .....	19
2.4.6. Java EE .....	20
2.4.7. Ruby on Rails.....	21
2.4.8. Django.....	22
3. Zpravodajský server o jezdeckví .....	24
3.1. Výběr technologie .....	24
3.2. Případy užití .....	25
3.3. Konkurence .....	26
3.4. ERA diagram.....	26
3.5. Obecné části projektu .....	27
3.6. Uživatelé .....	27
3.7. Soubory .....	28
3.8. Zprávy .....	30

3.9.	Články .....	31
3.10.	Akce.....	33
3.11.	Inzerce .....	36
3.12.	Stáj.....	38
3.13.	O nás.....	39
3.14.	Hlavní stránka.....	39
3.15.	Zhodnocení projektu.....	41
3.16.	Zhodnocení výběru technologie .....	41
4.	Technologie .....	42
4.1.	Spring framework.....	42
4.2.	Spring MVC .....	43
4.2.1.	Hello Word Spring MVC.....	44
4.2.2.	Spring MVC v demonstrátoru.....	45
4.3.	Ajax .....	45
4.3.1.	Hello Word Ajax.....	46
4.3.2.	Ajax v demonstrátoru.....	46
4.4.	Hibernate .....	47
4.4.1.	Hello Word Hibernate.....	47
4.4.2.	Hibernate v demonstrátoru.....	48
4.5.	jQuery.....	48
4.5.1.	Hello word jQuery .....	48
4.5.2.	jQuery v demonstrátor .....	49
4.6.	Google map API.....	49
4.6.1.	Hello Word Google Map API.....	50
4.6.2.	Google Map API v demonstrátoru.....	51
4.7.	Sociální síť .....	51
4.7.1.	Sociální plugin .....	52
4.7.2.	Facebook.....	52
4.7.3.	Twitter.....	57
4.7.4.	Google+ .....	60
5.	Demonstrátor .....	64
5.1.	Výukové aspekty.....	64
5.2.	Připojení k databázi.....	65
5.3.	Validace formulářových polí pomocí Ajax.....	66
5.4.	Uživatelé .....	69
5.5.	Soubory .....	70
5.6.	Zprávy a články.....	71

5.7.	Akce .....	72
5.8.	Inzerce .....	73
5.9.	Stáj.....	74
5.10.	Hlavní stránka.....	75
5.11.	Sitemap.xml.....	76
5.12.	Zprovoznění demonstrátoru.....	77
6.	Reálný provoz.....	78
6.1.	Hosting .....	78
6.1.1.	Výběr hostingu.....	79
6.2.	Zprovoznění hostingu.....	79
6.2.1.	Virtuální server .....	80
6.2.2.	J2EE Hosting .....	80
6.3.	Vynaložené finanční náklady na provoz projektu.....	82
6.4.	Výkon projektu.....	83
6.5.	Komplexní porovnání dvou reálně nasazených hostingů.....	85
6.6.	Google Analyst.....	86
6.6.1.	Zdroje návštěvnosti.....	86
6.6.2.	Cílové publikum .....	89
6.7.	Google AdWords .....	93
6.8.	Google Webmaster.....	98
7.	Závěr .....	102
	Seznam zkratk .....	103
	Seznam tabulek .....	104
	Seznam obrázků.....	105
	Seznam schémat.....	106
	Seznam zdrojových kódů.....	107
	Seznam použité literatury .....	108
	Přílohy.....	109

# 1. Úvod

Začátek práce se věnuje obecným východiskům, která se zabývají tvorbou webových aplikací, a elektronických výukových materiálů (e-learning). Dále je tato kapitola věnována analýze technologií pro tvorbu webových aplikací.

Následující část práce obsahuje návrh zpravodajského serveru o jezdeckví. Tento projekt je hlavní předlohou pro výukový program demonstrátor, který je nejkompexnější výukovou webovou aplikací této práce.

V předposlední části této publikace si představíme technologie, které jsou v projektu použity. Cílem této kapitoly je dané technologie představit a jejich popis doplnit příloženými funkčními miniaplikacemi. Na čtvrtou kapitolu volně navazuje kapitola pátá, která tvoří hlavní výukový text pro projekt demonstrátor. Kapitola záměrně neobsahuje výpisy zdrojových kódů, které jsou kompletně přiloženy.

Poslední kapitola této práce je věnována zkušenostem získaným z provozu reálného projektu zpravodajského serveru o jezdeckví, který se stal primární předlohou pro výukový projekt demonstrátor. Zpravodajský server je aktuálně spuštěn v ostrém provozu. Práce se v této kapitole dále zaměřuje na volně dostupné nástroje od společnosti Google, které zajišťují kompletní monitoring celého projektu.



## **2. Obecná východiska**

Tato kapitola pojednává o historii webu a webových aplikací, na tuto část volně navazuje pojednání o návrhu webových aplikací. Další část kapitoly je zaměřena na moderní techniku výuky tzv. e-learning a způsoby jeho využití ve výuce. Poslední část je věnována analýze technologií pro tvorbu webových aplikací.

### **2.1. Web a webové aplikace**

V současnosti používá internet velká část populace ve vyspělých zemích. Denní přístup k webovým aplikacím je již téměř zažitým stereotypem. V této kapitole je popsán vznik, vývoj a současné zásady pro tvorbu webových aplikací.

#### **2.1.1. Historie**

Internet vznikl jako armádní projekt v USA v 70 letech minulého století a jeho cílem bylo vybudovat decentralizovanou síť, která bude schopna fungovat i v případě výpadku některé z částí. V roce 1969 vznikl ARPANET, který byl považován za předchůdce dnešního Internetu. Pro vlastní komunikaci se zde začaly používat datové pakety. Od roku 1972 se začaly vyvíjet protokoly TCP/IP a v roce 1983 nahradil používaný protokol NCP – tato změna umožnila vzniknout prvnímu webu.

Základy webu položil Tim Berners-Lee, když v roce 1989 založil projekt na vytvoření distribuovaného hypertextového systému. Tim Berners-Lee je také autorem prvního internetového prohlížeče: WorldWideWeb. Jednalo se o první program, který nevyužíval pouze FPT, ale také Hypertext Transfer Protokol (HTTP). Po těchto krocích, se začal Internet i web exponenciálně rozšiřovat a to tak, že v roce 1996 měl přibližně 55 miliónů uživatelů a v současné době již přes 2,3 miliardy uživatelů po celém světě.

V tomto období umožňoval web uživateli pouze základní možnosti - čtení informací ze statických stránek vložených na web jejich autorem. Pro tvorbu webu se využíval jazyk HTML a začínalo se používat CSS. Později bylo možné využívat i JavaScript a Flash, které umožnili rozpohybovat jinak statické stránky. Návštěvníkům ovšem

přestávalo stačit být pouze pasivními konzumenty informací bez možnosti bezprostředně reagovat a komunikovat s ostatními uživateli připojenými k Internetu. Web se tedy začal postupně přizpůsobovat požadavkům uživatelů - začaly se vyvíjet technologie, které umožnily vývoj prvních webových aplikací, jak se známe nyní. Tento technologický postup dal vzniknout názvu Web 2.0.

### 2.1.2. Web 2.0

Definice pojmu Web 2.0 je velmi obtížná. Jedná se o termín, který v roce 2005 poprvé použil Tim O'Reilly. Nejedná se ovšem o žádný pevně stanovený standard, nýbrž pouze o souhrn nových postupů pro tvorbu webových aplikací a způsobů pro využití webu. Pojem Web 2.0 také zahrnuje technologie, které jsou známé již několik let, ovšem začali se hojněji využívat až v poslední době.

Nejčastěji zmiňované pojmy v souvislosti s pojmem Web 2.0:

- **„Many-to-many“**

*„Teoretik médií Vin Crosbie ve svém eseji *What is new media* používá slovo médium v ne zcela tradičním duchu a argumentuje, že existují pouze tři média, daná jejich povahou: interpersonální média (one-to-one), masová média (one-to-many) a nyní nová média (many-to-many). Ta umožňují vybírat si, personalizovat, a jsou technologicky podmíněná.“ [Zbi07]*

- **Producent/konzument**

*„S tím souvisí úzce další bod: když v 80. letech Alvin Toffler psal o vzniku tzv. prozumentů (prosumers – spojením slov producent a konzument), řada nástrojů, které postupně vedly k realizaci jeho myšlenek, ještě ani neexistovala. Jeho vize se naplnila v mnoha rovinách. Ať už je to samotný způsob používání a navigace v hypertextu umožňující nacházet obrovské množství cest (už jejich vybíráním je překročena předchozí lineární zkušenost a odpovídá to otevřenému textu, jak jej chápe Umberto Eco) nebo viditelnější přispívání komentáři (na weblozích, v diskusích či hodnocení produktů) až po snadné tvoření a distribuci hudby a videa – s nástroji technologické kvality před deseti lety vyhrazené pro špičková studia, dnes dostupnými komukoliv.“ [Zbi07]*

- **Wiki systémy**

*„Tento koncept je v zásadě do důsledku dovedenou aplikací předchozího: wiki systémy (wiki znamená v havajštině „rychle“) umožňují okamžitou editaci obsahu příslušné stránky kýmkoliv. Nejznámějším příkladem je nejrozsáhlejší encyklopedie na světě, Wikipedie, - ale není zdaleka jediná.“ [Zbi07]*

- **Reputační systémy**

*„Častým problémem na internetu je příliš velké množství informací, spojené s nemožností kvalifikovaně rozhodnout, která z nich jsou užitečná a která nikoliv. V těchto případech je nutné nasazení reputačních systémů, které agregují velké množství ohlasů a na jejich základě pak přidělují hodnocení.“ [Zbi07]* Jedna z nejznámějších stránek zabývající se a prosperující na této problematice je například heureka.cz.

- **Grafika a rozložení webu**

Grafika je důležitou součástí moderních webových aplikací a někdy i rozhodujícím faktorem pro uživatele při volbě mezi několika možnostmi. Design má být jednoduchý a přehledný. Mezi nejvíce využívané prvky se řadí výrazné logo, jednoduchá navigace, syté barvy či plastické plochy.

Také pro Web 2.0 je charakteristický neustálý vývoj. Již dnes se objevuje pojem Web 3.0, jež označuje směr, kterým by se mohl web i Internet celý do budoucna ubírat. Mezi nejčastěji skloňované termíny patří „*sémantický web*“, jež označuje web neobsahující pouze data, ale i jejich význam, což by mohlo napomoci strojovému čtení a zpracování informací. Od toho by již nemuselo být daleko k umělé inteligenci využívané na internetu. Ovšem až budoucnost ukáže, zda jsou tyto předpoklady pravdivé.

### **2.1.3. Webové aplikace**

Webová aplikace definuje aplikaci běžící na serveru, s níž uživatel komunikuje pomocí klienta na svém počítači, nejčastěji webovým prohlížečem. Ten sám o sobě nezná přesnou logiku aplikace, proto je možné použít označení tenký klient.

V poslední době se webové aplikace stávají čím dál tím více populární. K jejich největším výhodám patří možnost snadné aktualizace (distribuce) bez nutnosti instalace

či stahování SW. Díky neustále aktuální distribuci nových verzí je vždy nasazena pouze poslední verze aplikace. Další nespornou výhodou je vysoká dostupnost a naproti tomu nízká náročnost na HW a systém klienta. Kompatibilitu za tvůrce internetových aplikací zajišťují již internetové prohlížeče. Nevýhodou může být základní nedostupnost všech funkcí běžné známých z desktopových aplikací, jako je technika drag and drop, modální formuláře a další. K odstranění těchto omezení se v praxi používá skriptovacích jazyků, které se spouštějí na straně klienta (JavaScript). Ten umožňuje dynamicky modifikovat již načtenou stránku. Další nevýhodou jsou odlišné interpretace HTML a CSS v jednotlivých prohlížečích, které mohou zhoršit grafickou stránku aplikace. Výhody nad nevýhodami nicméně převažují a proto se stále častěji setkáváme s webovými aplikacemi jak pro externí tak intranetové aplikace.

## **2.2. Návrh webových aplikací**

Na samotném počátku každé aplikace, je idea, kterou se rozhodneme transformovat na aplikaci. Je však možné, že před námi již někdo takový nápad měl, je proto vhodné provést analýzu, zda již podobná aplikace neexistuje. Pokud ano, můžete z ní čerpat, namísto vývoje vlastní, což vede ke značné úspoře času při samotném vývoji. Při úvodní analýze je důležité si ujasnit představy a funkční požadavky na danou webovou aplikaci, dále cílovou skupinu zákazníků/návštěvníků, samozřejmě také smysl a cíl samotného projektu. Počítat musíme také s možnými budoucími úpravami a aktualizacemi. V neposlední řadě také musíte znát konkurenci, která bude spuštěnému webu konkurovat.

### **2.2.1. Návrh architektury webu**

Pro vytvoření úspěšné webové aplikace je zapotřebí při přípravě správně a účelně navrhnout její optimální architekturu. Už víte, co požadují budoucí návštěvníci a v čem musíte být lepší než konkurence.

Architekturou webu je nazýváno celkové uspořádání jednotlivých kategorií a podkategorií webových stránek, jejich logické uspořádání a celkové uspořádání informací v rámci webu, jejich vzájemnou provázanost a vhodně pojmenované odkazy. Všechny tyto skutečnosti ve svém důsledku ovlivňují použitelnost a přehlednost webových stránek a jsou základem pro zajištění kvalitní optimalizace pro vyhledávače.

### **2.2.2. Programování webových stránek**

Ve fázi, kdy už máte jasno, co přesně na webu bude, je vhodné si navrhnout funkční prototyp. Nejvhodnější je zhotovit prototyp v XHTML souborech s hotovým rozmístěním prvků a provázáním stránek. Samozřejmostí zhotovení prototypu stránek je jeho plná funkčnost. V této fázi vývoje provedeme první testování jako vývojáři a také zástupci cílové skupiny (uživatelské testování).

Správné kódování stránek neboli správné používání validního kódu HTML5 a CSS3, má vliv na zobrazení webových stránek v jednotlivých typech a verzích internetových prohlížečů, mobilních zařízeních a jejich prohlížečích, ovlivňují také přístupnost webu pro vyhledávače a také pro hlasové čtečky tělesně postižených uživatelů.

### **2.2.3. Grafický design**

V této fázi vývoje webové aplikace vyrobíme grafický návrh. Je důležité, si při návrhu grafiky stanovit případná omezení, kterými se budeme řídit. Jsou jimi například požadavky cílové skupiny budoucích uživatelů aplikace a firemní barvy a loga, v tomto případě respektuje a dodržujeme tedy corporate design. Po vytvoření grafického designu je potřeba vzhled převést do XHTML/CSS souborů a spojit s již existujícím prototypem.

### **2.2.4. Schválení a uvedení webových stránek do provozu**

Jestliže je kompletně hotové kódování a design aplikace, vystavíme aplikaci na speciální doméně na zvoleném serveru a následuje akceptační fáze. Během této fáze testujeme funkčnost, provádějí se obsahové korektury aj. V této fázi spuštění webové stránky k ní nemají přístup internetové vyhledávače, aby její obsah nemohl být indexován a nezobrazoval se ve výsledcích vyhledávání.

Pokud je vše řádně otestováno, plně funkční a schváleno námi, či klientem, který si webové stránky nechal zhotovit, přistoupí se k ostrému spuštění aplikace. Proveďte se tedy jeho umístění na hostingový server a uvolní se k registraci a indexaci internetovým vyhledávačům.

## 2.2.5. Údržba

Je pravděpodobné, že i po spuštění webových stránek bude nutné se jim do budoucna věnovat. Provádět aktualizace a vylepšení a také kontrolovat jeho umístění ve vyhledávačích, vzhledem k faktu, že jejich algoritmy se často mění. Samozřejmostí je také procházení chybových výpisů a případná reakce na případné méně závažné chyby, které se v první části testování nemusí vždy projevit.

## 2.3. Počítačem podporovaná výuka

E-learning je vzdělávací proces, využívající, informační a komunikační technologie k tvorbě výukových materiálů, jejich distribuci, komunikaci mezi studenty a pedagogy a k řízení studia. Clark a Myer [Clark, 2007] definují e-learning jako „*počítačovou výuku prostřednictvím CD-ROM, internetu nebo intranetu, která má následující rysy: její obsah odpovídá vzdělávacímu cíli; používá vyučující metody, které usnadňují vyučování (příklady a cvičení); používá různá média k předání obsahu; může být vedeno instruktorem (synchronní e-learning) nebo je určeno pro samouky (asynchronní e-learning), vytváří nové znalosti a dovednosti spojené s individuálními vzdělávacími cíli.*“ Multimediální texty, které jsou podkladem pro tvorbu e-learningových programů, v dnešní době nahrazují tištěné učebnice a skripta.

### 2.3.1. Historie

Počátky využívání výpočetní techniky ve výuce, respektive výuka probíhající za pomoci počítačů, se datují od šedesátých let minulého století. První představy o využití počítačů ve výuce byli však značně nerealistické, hlavně díky pomalému vývoji soudobých informačních technologií. Prvním mezníkem pro komplexní využití výpočetní techniky ve výuce se stal vznik multimediálních technologií. Současně se vznikem multimédií započal rychlý vývoj vysokokapacitních paměťových médií a komunikačních schopností počítačů, především díky spuštění celosvětové sítě Internet.

V současné době je ve výukovém procesu žádoucí přechod od přetrvávající encyklopedické výuky a neustále se zvyšujících nároků na paměť studentů, k naučení se a vstřípení využívání elektronických forem informací. Žádoucí je rozvíjení a podpora

zejména logického myšlení a schopnosti odvozovat, protože cílem moderního vzdělávání nejsou jen jisté základní znalosti, ale je kladen daleko větší důraz na samostatnost, tvořivost a hledání nových alternativ a pohledů.

### **2.3.2. Výukové činnosti**

Moderní koncepci výuky s pomocí výpočetní techniky rozdělujeme dnes do dvou skupin činností:

- počítačem podporovanou výuku
- počítačem řízenou výuku

Není možné tyto dvě oblasti brát jako disjunktivní, jelikož výuka podporovaná počítačem v sobě dost často nese i určité prvky řízení, které jsou zabezpečeny počítačem, a naopak. Výuka podporovaná počítačem, v cizí literatuře označovaná zkratkou CAI (Computer Assisted Instruction) nebo v poslední době CBI (Computer Based Instruction), obsahuje následující typy výukových činností:

1. procvičování
2. dialogová výuka
3. modelování a simulace
4. didaktické hry
5. testování (podporované nebo přímo prováděné počítačem)
6. řešení problémů
7. učení objevováním

### **2.3.3. Výhody a nevýhody e-learningu**

Pokud se uživatel rozhodne pro vzdělání e-learningovou formou, například volbou e-learningového kurzu, zjistí, že tento způsob čerpání informací s sebou přináší určité výhody a nevýhody.

#### **Výhody:**

- Studium přímo doma - odpadá cestování a s ním spojené finanční náklady.
- Možnost souběžného učení velké skupiny studentů.
- Volba vlastního tempa a času věnovaného studiu.

- Informace jsou čerpány kdekoli je to pro studenty výhodné, odpadá cestování za studiem, úspora času.
- Daný materiál lze procházet několikrát (opakovaně si poslechnout audio-záznam přednášky, spustit animaci, přečíst definici).
- Pestrost prezentace učební látky.

#### **Nevýhody:**

- Není vhodný pro všechny věkové kategorie (jeho použití je doporučováno až pro dospělé studenty) – spojitost s disciplínou a vnitřní motivací.
- Náročnost na tvorbu a realizaci kurzu. Počáteční nákladnost.
- Na tvorbě e-learningového kurzu by se mělo podílet více specialistů (učitel, designér, programátor), jinak hrozí, nepřipravenost autorů po stránce pedagogicko-didaktické, používání nevhodných autorských nástrojů typu kancelářských aplikací, neznalost standardů běžně používaných v oblasti internetu.
- Nutnost HW a SW, připojení k internetu.
- Chybí kontakt s učitelem, s ostatními spolužáky.

### **2.3.4. Konstruktivismus a E-learning**

Definici spojení konstruktivismu a e-learningu čtivě definuje Lucie Rohlíková:

*„Konstruktivistické pojetí je označováno jako ideální pedagogické východisko pro eLearning, neboť informační a komunikační technologie jsou nástrojem schopným vytvořit pro konstruktivisticky pojatý vzdělávací program velmi dobré podmínky. Konstruktivistická pedagogika staví studenta do centra vzdělávacího procesu. Stejně tak eLearning předpokládá samostatného studujícího, který řídí a organizuje své učení v on-line systému. Tradiční role učitele se v eLearningu přirozeně mění a učitel se stává konstruktivistickým tutorem, facilitátorem a průvodcem. ELearning ze své podstaty řízeného samostudia nutí studujícího převzít odpovědnost za své učení, kontrolovat své výsledky a hodnotit dosažené pokroky. Zároveň on-line technologie umožňuje studentům zaznamenávat a sledovat vlastní učení a zpětně svůj styl i metodiku reflektovat. Synchronní nebo asynchronní interakce mezi účastníky on-line vzdělávání jsou v eLearningu možné bez bariér místa a času. Komunikaci je však třeba povzbuzovat jednak vytvořením příjemné atmosféry otevřeného prostoru pro sdílení názorů a jednak*



*vhodnou koncepcí skupinové práce.*“ [Zbi07] Adam Zbiejczuk, Mediální Studia, Masarykovy univerzity v Brně 2007

[WRohl].

## **2.4. Technologie pro tvorbu webových aplikací**

Při tvorbě webové aplikace je nutné téměř vždy využít více než jednu technologii, stejně jako tomu je u desktopových aplikací.

### **2.4.1. HTML**

HTML je zkratka Hyper Text Markup Language. Jedná se o značkovací jazyk pro hypertext - hlavní jazyk pro tvorbu stránek v internetovém protokolu World Wide Web, který umožňuje publikování dokumentů na Internetu. Jazyk HTML je charakterizován množinou značek a jejich atributů. Názvy jednotlivých značek a jejich vlastností se uzavírají mezi úhlové závorky < >.

### **2.4.2. CSS**

CSS je zkratka kaskádové styly (v anglickém originálu *Cascading Style Sheets*). Jedná se o jazyk pro popis způsobů zobrazení stránek napsaných v jazycích HTML, XHTML. Tento jazyk navrhla standardizační organizace W3C<sup>1</sup>, autorem byl Håkon Wium Lie. Hlavním cílem tohoto jazyka je oddělit vzhled dokumentu od jeho struktury a tím zpřehlednit samotný HTML kód stránky. Vydány jsou specifikace CSS1, CSS2 a v roce 2011 byla dokončena revize CSS2.1. Aktuálně je možné setkávat se již i s CSS3 které se pevně váže se standardem HTML5. Pro tento jazyk nadále zůstává standardizační autorita W3C<sup>1</sup>.

### **2.4.3. JavaScript**

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk, jehož autorem je Brendan Eich. JavaScript se z pravidla používá jako interpretovaný programovací jazyk pro webové stránky. Do webových stránek je vkládán buď přímo,

---

<sup>1</sup> <http://www.w3.org/Consortium/>

nebo pomocí linkování JS souborů. Po stažení celé webové stránky na klientskou stanici dochází k jeho automatickému spuštění, na rozdíl od ostatních jiných interpretovaných programovacích jazyků (např. PHP, ASP), které se spouštějí na serveru ještě před stažením z internetu. Z toho vyplývají určitá bezpečnostní omezení, např. nemožnost práce se soubory klienta a jiné.

#### **2.4.4. PHP**

PHP je skriptovací programovací jazyk, který je určený k tvorbě dynamických webových stránek například ve formátu HTML či XHTML. Při použití PHP pro dynamické stránky jsou skripty prováděny na straně serveru – ke klientovi je tedy přenášen až výsledný HTML kód. Syntaxe jazyka je inspirována několika programovacími jazyky (Perl, C, Pascal a Java). PHP je na platformě nezávislý a případné rozdíly v různých operačních systémech se omezují pouze na malou část systémově závislých funkcí.

Aktuálně je PHP nejrozšířenějším skriptovacím jazykem pro vývoj webových aplikací. V kombinaci s operačním systémem Linux, databázovým systémem MySQL a webovým serverem Apache, je nejčastěji využívanou kombinací pro webové aplikace.

#### **Výhody PHP**

- PHP je specializovaný na webové stránky.
- Rozsáhlý soubor funkcí v základní knihovně PHP (přes pět a půl tisíce), další funkce v PECL.
- Nativní podpora mnoha databázových systémů.
- Multiplatformost (zejména Linux a Microsoft Windows).
- Možnost využití nativních funkcí operačního systému (možná nekompatibilita s jiným OS).
- Vysoká podpora na hostingových službách.
- Obrovské množství projektů a kódů, které lze zdarma využít (WordPress, phpBB a další).
- Poměrně dobrá dokumentace.
- Velmi svobodná licence, která (v protikladu k např. GPL) neobsahuje copyleft.

## **Nevýhody**

- Jazyk PHP je definován pouze svou jedinou implementací.
- Nekonzistentní pojmenování funkcí (strpos() atd.).
- Nejednotné pořadí parametrů.
- Malé použití výjimek v knihovnách jazyka.
- Slabší podpora Unicode.
- Ve standardní distribuci chybí ladící nástroj.
- Po zpracování požadavku neudrhuje kontext aplikace, vytváří jej vždy znovu.

## **2.4.5. ASP.NET**

ASP.NET je založen na Common Language Runtime, díky němuž je sdílen všemi aplikacemi postavenými na .NET Frameworku. Vývoj aplikací je tak možný realizovat v jakémkoliv jazyce podporujícím CLR, např. Visual Basic.NET, C#, Managed C++ a dalších. Koncept ASP.NET WebForms ulehčuje přechod od klasických aplikací pro Windows do prostředí webu: stránky jsou poskládány z objektů, ovládacích prvků, které jsou protějškem ovládacích prvků ve Windows. Při tvorbě webových stránek je tedy možné používat standardní prvky jako tlačítko label a další. S těmito prvky je možné dále pracovat jako při vývoji desktop aplikací: je možné jim nastavovat vlastnosti, odchyťávat události, nastavovat pozici na obrazovce atd. Dle nastavení jednotlivých prvků je poté automaticky generovaný HTML kód, který tvoří výslednou stránku posílanou dále na stranu klienta.

## **Výhody**

- Díky kompilovanému kódu běží aplikace rychleji a více chyb je zachyceno už při vývoji
- Uživatelsky definované ovládací prvky lze použít jako šablony, čímž se významně redukuje duplicitní kód
- Podobný přístup jako k aplikacím pro Windows zjednodušuje přechod od jednoho prostředí k druhému
- Bohatý výběr ovládacích prvků a knihoven tříd velmi zrychluje vývoj aplikací
- Programátoři mají na výběr velké množství programovacích jazyků

- Schopnost cachovat celou stránku nebo pouze její části podstatně zvyšuje výkon serveru
- Lze jej provozovat na různých operačních systémech i webových serverech, např. IIS (Windows), Apache
- Počínaje verzí 2 generuje ASP.NET validní HTML 4.0 / XHTML 1.0 / XHTML 1.1 kód a JavaScript

#### **Nevýhody**

- Menší podpora komerčních hostingů
- Složitější implementace

#### **2.4.6. Java EE**

Java Platform Enterprise Edition (dříve označovaná jako Java 2 Enterprise Edition nebo J2EE) je součástí platformy Java a je určená pro vývoj a provoz podnikových aplikací a informačních systémů. Jedná se o rozšíření platformy Java SE (Standard Edition) o podporu webových aplikací, webových služeb a distribuovaných vícevrstvých aplikací.

Java EE je velice robustní framework porovnatelný svojí robustností s ASP.NET, ovšem u Java EE nejsou programátoři limitováni pouze jednou technologií. Výhody a nevýhody Java EE tedy spíše souvisejí s použitými technologiemi, než se samotnou platformou.

Prezentační vrstva je zodpovědná na zpřístupnění funkcionality aplikace, uživatelé se v současné době nejvíce využívají technologie JavaServlets a JSP (JavaServer Pages). Webové aplikační rámce mohou být založené na požadavcích (příkladem jsou technologie Struts, Stripes, Spring MVC), nebo na vizuálních komponentách (například JSF nebo Tapestry).

V aplikační vrstvě, která zajišťuje vlastní funkcionality, se nejvíce používají EJB nebo Spring. Pro komunikaci aplikační vrstvy s databází se nejčastěji využívají nástroje Hibernate či Java Persistence API.

#### **Výhody**

- Vysoce výkonná platforma pro realizaci a běh výkonných řešení.
- Multiplatformost (zejména Linux a Microsoft Windows).
- Vytvořeny *design patterns*, techniky, metody a aplikační rámce, které podporují "dobrý" styl návrhu a vývoje řešení.
- Velké množství již hotových komponent.
- Velké množství stále rozšiřujících technologií.
- Dostupnost vývojového prostředí zdarma
- Platforma podporována velkými firmami (Oracle / Sun Microsystems, IBM, Spring Source).

#### **Nevýhody**

- Menší podpora komerčních hostingů
- Složitější implementace (nehodí se na menší projekty)

### **2.4.7. Ruby on Rails**

Ruby on Rails, zkráceně Rails, je Framework, díky němuž je možné poměrně rychle a pohodlně vytvářet moderní webové aplikace, napojené na databázi používající návrhový vzor Model-view-controller. V roce 2004 jej vytvořil dánský programátor David Heinemeier Hansson při práci na projektu Basecamp a dodnes je vylepšován tisíci přispěvateli. Používají ho společnosti jako Apple, Oakley, The New York Times, Twitter, Electronic Arts, Github nebo YellowPages.

Vše v Ruby on Rails stojí na jazyce Ruby, jako Ajax v pohledech, odpovědi v radičích i architektura aplikace, k jejímuž spuštění je potřeba jen databáze. Programátor zde konfiguruje pouze části aplikace lišící se od běžného nastavení, má tu tedy konvence přednost před konfigurací. Rails jsou postaveny na bázi návrhového vzoru MVC oddělující části aplikace zodpovědné za čtení a ukládání dat. Daty také manipuluje, zodpovídá za zobrazení grafického rozhraní aplikace, zobrazení dat na výstupu a za část přijímající vstupy od uživatele. Rails aplikace se mohou hostovat na vlastním VPS a celé řadě zahraničních hostingů. V České republice existuje pouze jeden hosting, který se na Rails specializuje – Railhosting.cz.

#### **Základní vlastnosti Ruby on Rails**

- Automaticky mapují URL na vnitřní řídicí prvky aplikace
- Zajišťují předávání dat mezi řadičem a modelem a mezi řadičem a pohledem
- Mapování záznamů z relační databáze na objekty obsahují přístup k datům v databázi
- Obsahují rozsáhle pomocné knihovny pro snadné generování HTML, pro práci s Ajax, formátování dat atd.

### **Výhody**

- Abstraktní vrstva pro práci s databází
- Přehledná implementace MVC architektury
- Zabudovaná podpora pro automatizované testování všech vrstev aplikace
- Obsahuje generátory kódu
- Obsahuje konzoli pro interaktivní práci s aplikací

### **Nevýhody**

- Malá podpora hostingových serverů v ČR

## **2.4.8. Django**

Django je vyspělý webový framework napsaný v jazyce Python, který podporuje rychlý vývoj a čisté, pragmatické konstrukce. Django byl vyvinut před několika lety společností The World Company v Lawrenci v Kansasu, později, v červnu 2005, bylo vydáno veřejně pod open-source licencí BSD. Framework byl pojmenován po jazzovém kytaristovi Django Reinhardtovi a mělo pomoci zvládat dva velké úkoly: intenzivní plnění termínů v redakci a dodržování přísných požadavků vývojářů — autorů framework. Také je další z řady webových framework, jako je např. Ruby on Rails, oproti kterému nabízí možnost automatické tvorby administrace projektu, která je generována dynamicky podle datového modelu.

Jádro frameworku Django obsahuje objektově-relační mapper zprostředkovávající přesun informací mezi relační databází a datovým modelem (definovaným jako třídy Python). Většina rozhodnutí je ponechána na programátorovi aplikace, který může použít jakoukoliv podmnožinu nabízené funkcionality (některé podmnožiny je možné aplikovat i při vývoji desktopových aplikací).

Jako entita je v Django vnímán projekt, který je charakterizován konfiguračním souborem `settings.py`, obyčejným modulem v Python, který je společný pro všechny aplikace v Django. Vzhledem k možnosti vytváření struktury projektu a aplikací jednoduchým skriptem, není potřeba používat sofistikovanější IDE a programátorovy postačí obyčejný textový dokument s podporou zvýrazňování syntaxe. Aplikace je vždy součástí jednoho nebo více projektů.

Framework Django může fungovat na Apache 2 s Mod Python, nebo pod jakýmkoliv webovým serverem vyhovujícím WSGI a je schopen spouštět server FastCGI v pozadí webového serveru, který jej podporuje.

### **Výhody**

- Překlad všech komponent do libovolného jazyka.
- Vygenerování plně okomentovaného konfiguračního souboru s významem jednotlivých konfiguračních klíčů.
- Možnost dědičnosti šablon – umožňuje v potomkovi předefinovat, či rozšířit sekce definované v předkovi.

### **Nevýhody**

- Málo dostupných informací v českém jazyce

## 3. Zpravodajský server o jezdeckví

Jedním z hlavních cílů této práce bylo vytvořit funkční reálný projekt, díky kterému by bylo možné propojit výukovou část práce (kapitola 5 Demonstrátor) s částí reálnou (kapitola 6 Reálný provoz). Realizovaný projekt [zesedla.cz](http://www.zesedla.cz)<sup>1</sup> se věnuje zpravodajství o koních a jezdeckví.

### 3.1. Výběr technologie

Dle technologií uvedených v kapitole 2 Obecná východiska by bylo možné určit jako neoptimálnější variantu vzhledem k očekávanému zatížení serveru a potřeby robustnosti projektu platformu PHP. Ovšem z hlediska zadání této práce (kde je definován Spring MVC) a dále z potřeby užití této práce, jako práce výukové, bylo vhodnější využít platformu Java EE.

---

<sup>1</sup> <http://www.zesedla.cz>



### 3.2. Případy užití

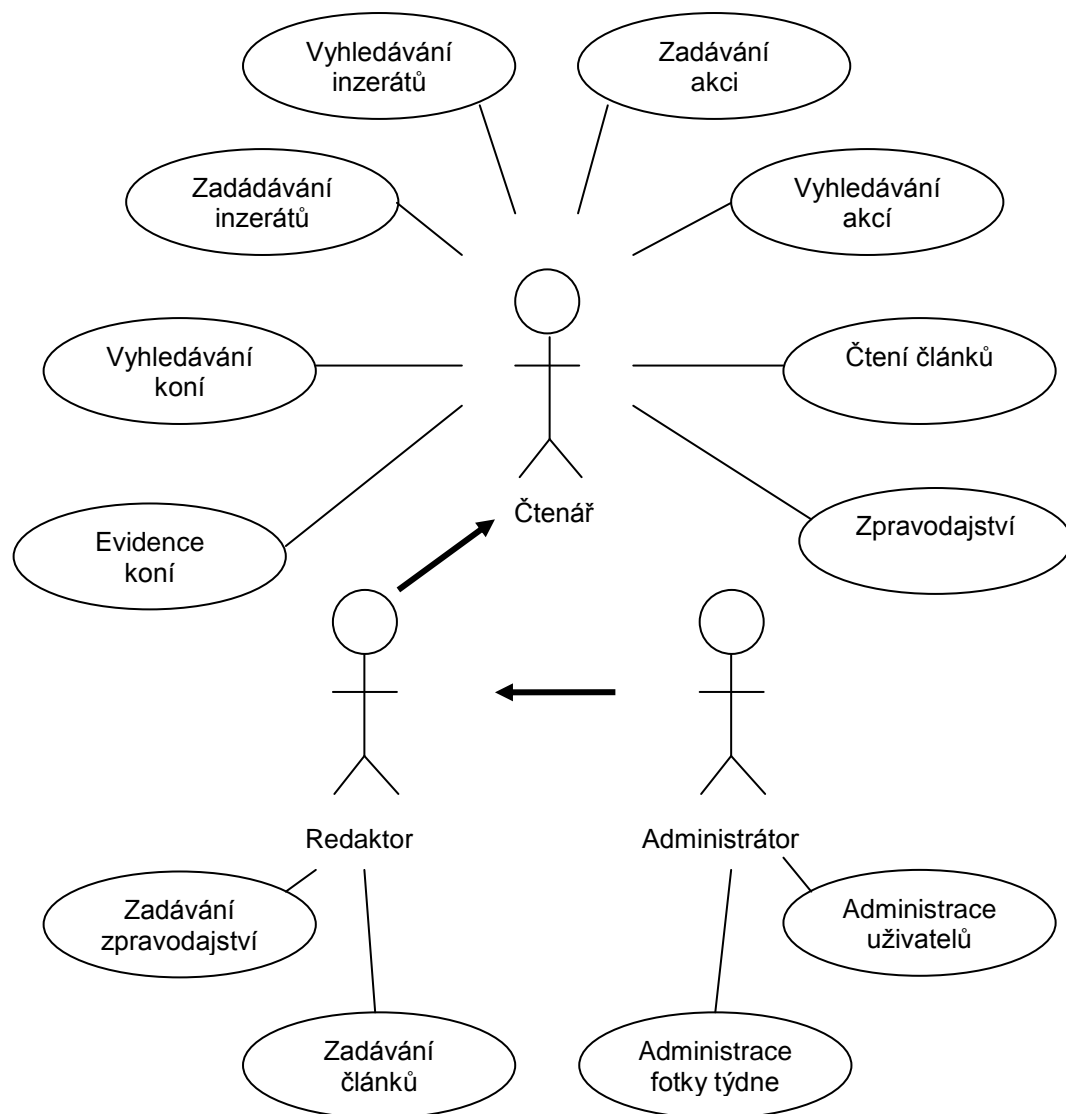


Schéma 3.1 Případy užití projektu zeSedla.cz

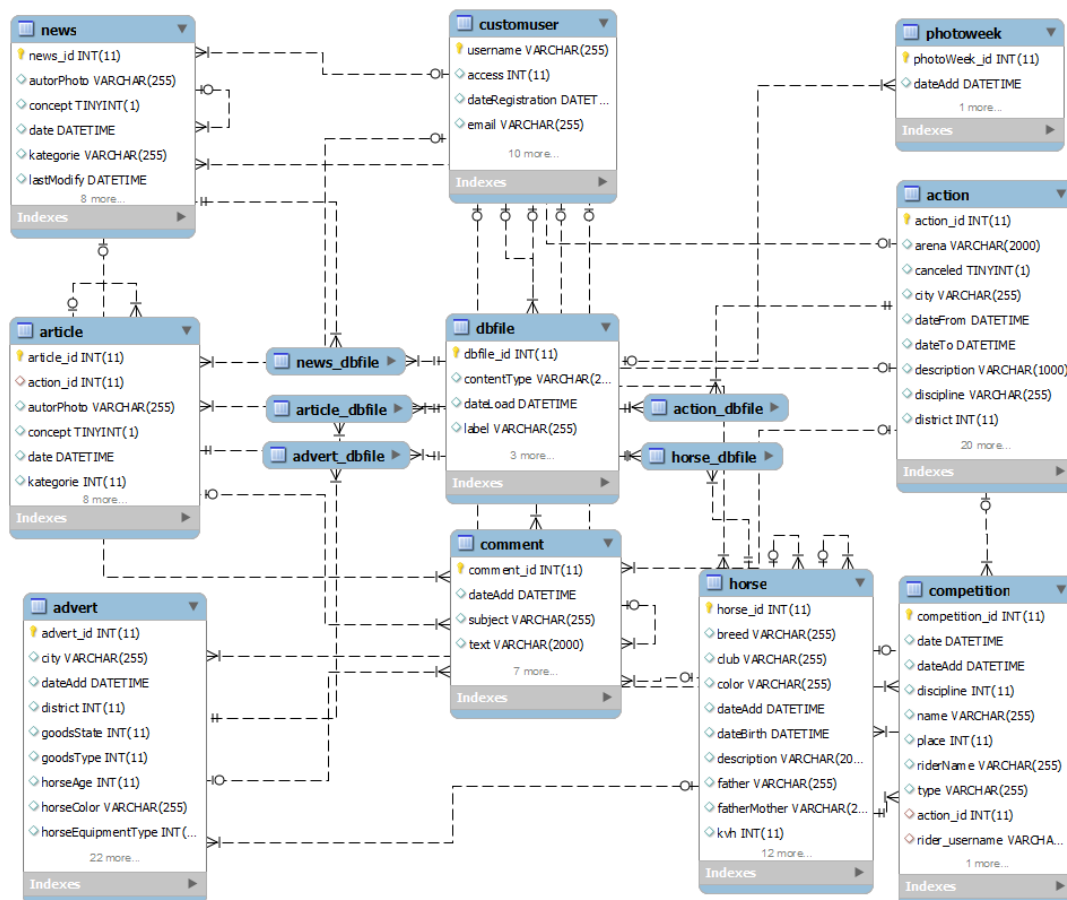
### 3.3. Konkurence

Hlavním cílem celého návrhu bylo vytvořit rozdílný web, který by nekopíroval jednotlivé již dostupné konkurenční weby, ale pouze načerpat drahocenné zkušenosti z již provozovaných webů. Díky podrobné analýze konkurenčních webů, bylo možné vyvarovat se realizaci chybných částí webu, které mají téměř nulovou návštěvnost:

- Aukce na distanc.cz – umožňuje provádět aukce za virtuální měnu
- Adresář na distanc.cz – eviduje firmy a hospodářství v oblasti jezdeckví

Opakem vyvarování se nenavštěvovaných částí webu bylo tedy možné sestavit nejvíce navštěvované části. Na tyto části a jejich kvalitu zpracování bylo možné se při realizaci prioritně zaměřit.

### 3.4. ERA diagram



Obrázek 3.1 ERA diagram zeSedla.cz

### 3.5. Obecné části projektu

Komplexní návrh projektu vychází z podrobné analýzy případů užití. Cílem je použití DAO vrstvy, která by dle návrhu tříd automaticky generovala databázové tabulky. Je tedy nutné navrhnout třídy komplexně včetně všech potřebných návazností na ostatní třídy.

Prioritně je nutné specifikovat třídy, které budou používány napříč celým projektem. Obecně je poměrně složité tyto moduly plnohodnotně navrhnout v první fázi vývoje, je ale možné vycházet z podrobné analýzy konkurenčních webů:

- Uživatelé – registrace, přihlášení, administrace.
- Soubory – fotky, dokumenty a jiné.

Díličí návrhy jednotlivých částí již budou zahrnuty níže v podrobném popisu těchto částí (4.3 – 4.10). Nyní lze již navrhnout jejich provázanost s výše uvedenými třídami.

Obecně lze tedy navrhnout, že jednotlivé části webu budou obsahovat odkaz na třídu uživatel. V prvním návrhu budeme předpokládat, že pouze regulérní uživatelé budou moci vkládat zprávy, články, inzeráty a akce. Tyto dílčí části webu budou dále nutně obsahovat odkaz na třídu soubor (minimálně pro vložení titulního obrázku).

### 3.6. Uživatelé

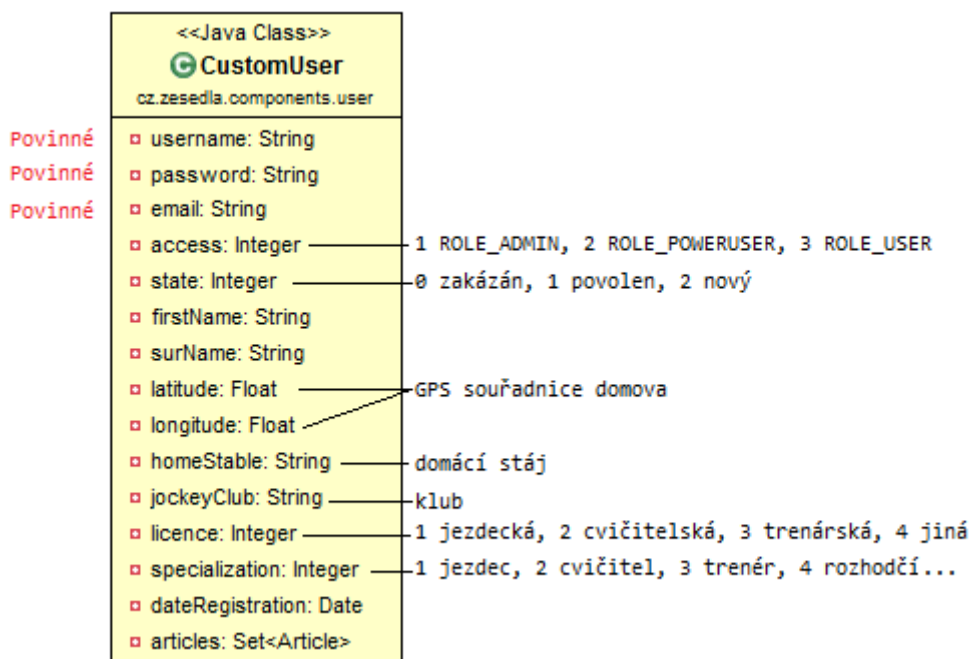
Třída uživatel bude používána napříč celým systémem. Je tedy nutné, aby obsahovala návaznosti na jednotlivé části webu. Další důležitou vlastností uživatele musí být rozdělení na typy uživatelů, pokud předpokládáme použití více typů účtů:

- Administrátor
- Uživatel
- Anonymní uživatel (bez přihlášení, anonymní návštěvník)

Účet typu administrátor bude použit pro přidělení maximálních privilegií a bude moci spravovat kompletní obsah webu. Bude tedy moci moderovat veškerý obsah i ostatních uživatelů.

Další potřebnou vlastností bude možnost deaktivovat účet. Pokud budeme předpokládat jedinečnou identifikaci pomocí emailového účtu, můžeme tuto možnost využít například pro zablokování uživatelského účtu (emailové adresy).

Ostatní vlastnosti uživatele už slouží pouze pro doplnění údajů o uživateli a jedná se o vlastnosti pevně svázané s jezdecktím. Při registraci se také jedná o nepovinné údaje.



Obrázek 3.2 Návrh třídy uživatele

### 3.7. Soubory

Třída Soubor se bude používat v dílčích částech webu. Je tedy nutné, aby obsahovala odkazy na tyto části. Teoreticky nebude nutné evidovat soubory, které například nepatří již k žádnému objektu v databázi. Prioritní pro nás tedy bude správné provázání s danou částí webu.

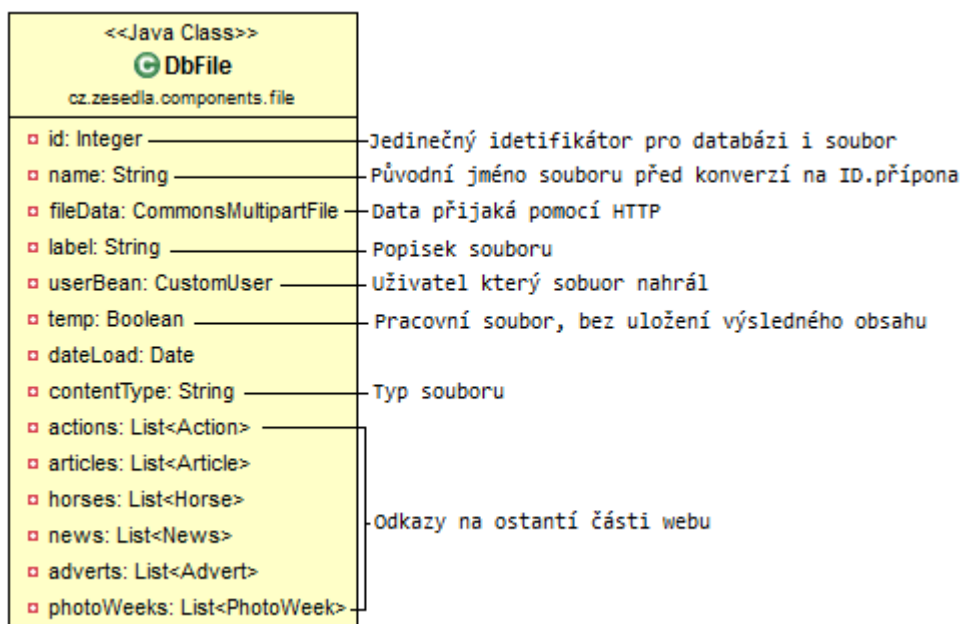
V dalším návrhu třídy musíme již zohlednit praktickou implementaci. Soubory mohou být uchovávány ve dvou hlavních úložištích:

- Na diskovém prostoru
- V databázi

Pro tento projekt vybereme možnost uchovávání souborů na diskovém úložišti vzhledem k tomu, že z nabízených kapacit hostingových služeb je kapacita databáze častěji omezována než kapacita diskového prostoru.

S tímto rozhodnutím, je tedy nutné nahranému souboru vygenerovat jedinečné ID, které bude svazovat objekt v databázi s fyzicky uloženým souborem na diskovém úložišti. Z důvodu usnadnění orientace, je při nahrání tento soubor pod vygenerované ID také uložen.

Jako poslední problém, který byl řešen až v rámci implementace ostatních částí, bylo použití Ajax načítání souboru. S touto rozšířenou funkcí bylo nutné vyřešit stav, kdy uživatel provede načtení nových souborů, ovšem již neprovede uložení vkládaného obsahu (článku, akce atd.). Kvůli tomuto problému byl do třídy souboru přidán příznak Temp. Temp příznak pracuje v návaznosti na uživatele tak, že pokud dojde k opětovnému přihlášení uživatele, dojde k automatickému smazání pracovních Temp souborů. Tímto krokem zamezíme existenci většího množství Temp souborů.



Obrázek 3.3 Návrh třídy soubor

### 3.8. Zprávy

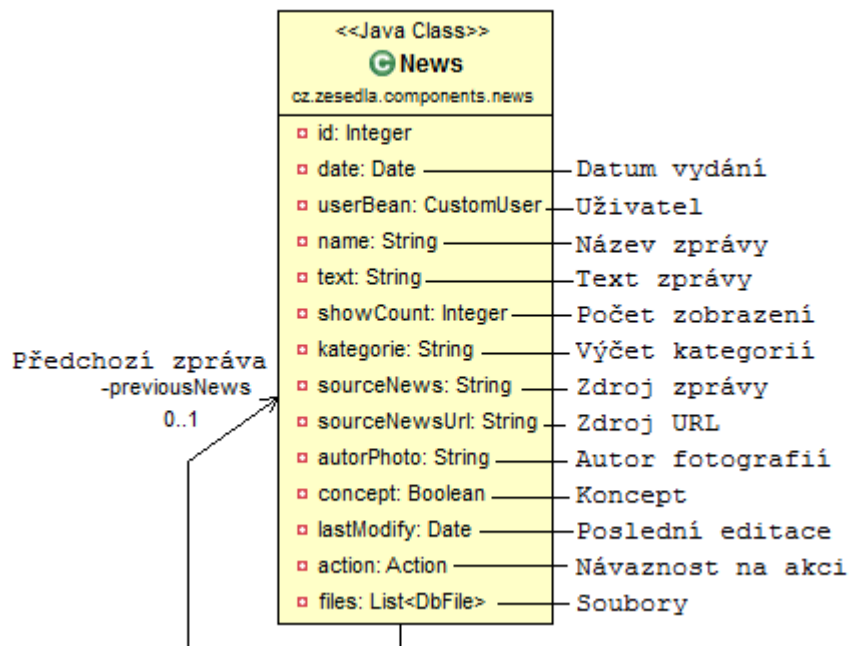
Část webu určená pro prezentaci zpráv je navržena podle obecného publicistického modelu, se kterým se můžeme setkat běžně na internetu. Prioritně je nutné sestavit nejdůležitější části, které částečně vychází i z povinnosti citací a šíření informací od externích zdrojů:

- Název zprávy
- Kategorizace (rozlišení pro filtrování)
- Datum vydání
- Text zprávy
- Zdroj zprávy (autor, URL adresa zdroje, autor fotografií)
- Soubory (úvodní obrázky, datové soubory, dokumenty, fotogalerie)

Další přidanou hodnotou naproti konkurenčním webům je umožnění propojení více zpráv vůči sobě, bez nutnosti definice další projektové větve – seriálů. Díky definici nadřazené zprávy je dynamicky vytvářen seriálový strom, který takto seskupuje skupinu zpráv do jednoho virtuálního seriálu. Prioritně tedy nemusíme definovat vazby jednotlivých zpráv mezi všemi ostatními, pouze mezi jedinou předchozí, která se dále odkazuje na vlastní předchozí zprávu.

Za poslední rozšířené vlastnosti lze také považovat možnost ukládání rozpracovaných konceptů – zpráv, které prozatím nejsou publikovány. Dále je možné provázat vydávanou zprávu s evidovanou akcí, čímž vznikne vazba například pro plnohodnotné provázání vydané fotogalerie s původními informacemi o akci (čas a místo konání).

Vzhledem k tomu, že je s cizími autorskými díly třeba patřičně zacházet, musí zde být evidována také sekce označená jako: zdroj zprávy. Blíže se této problematice věnuje práce z Masarykovy univerzity – Kurz práce s informacemi [WMuB].



Obrázek 3.4 Návrh třídy zpráva

### 3.9. Články

Tato sekce v projektu vznikla z potřeby dalšího třídění prezentovaných informací. Pokud bychom opět na projekt nahlíželi z pohledu návštěvníka, můžeme návštěvníka rozdělit z hlediska vyhledávání informací na dvě skupiny:

- Hledající obecné naučné informace (články)
- Hledající informace o aktuálním dění (zpravodajství)

Z tohoto důvodu bylo vytvoření sekce články velice důležité. Díky tomu je možné diverzifikovat návštěvníky do dvou skupin a daleko lépe celou koncepci projektu upravit tak, aby byli informace pro oba typy návštěvníků co možná nejrychleji dostupné.

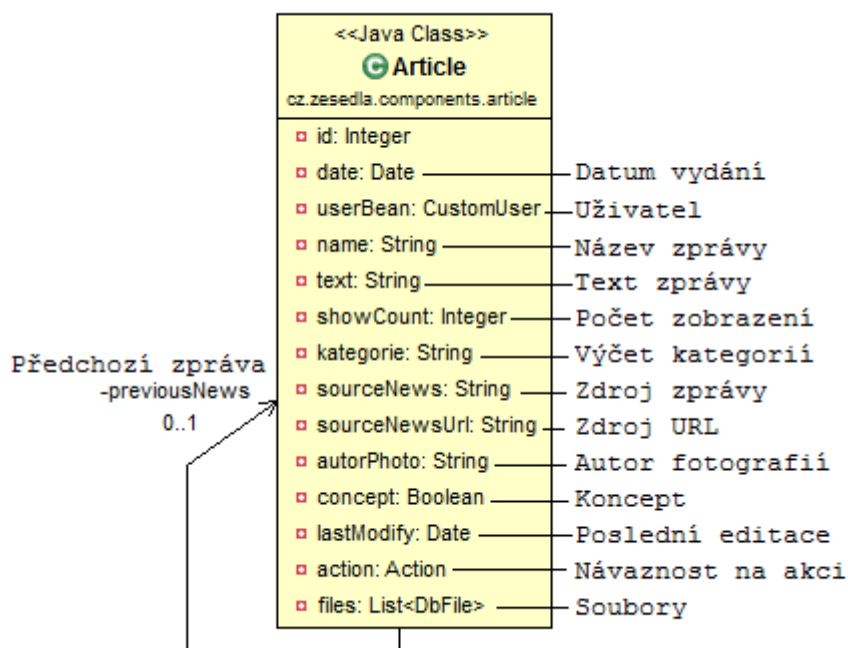
Z praktického hlediska by při sloučení těchto dvou sekcí byly naučné články zahrnuty do skupiny aktualit, čím by se zablokovala možnost v těchto článcích vyhledávat. Pokud bychom se dostali dále do reálného nasazení, aktuality (zprávy) čítají

v jezdecké sezóně zhruba dvě zprávy denně. Při této četnosti by zmíněné naučné články byli problematicky dosažitelné.

Cílem této sekce je tedy diverzifikovat tok informací tak, aby bylo možné vytvořit databázi naučných článků, které jsou pro čtenáře atraktivní i v časovém horizontu například 5 let.

Návrh článku vycházel prioritně z třídy zprávy. S sebou si převzal i možnost vytvářet generované seriály, které jsou složené z více článků a to za pomoci pouze evidence předchozí zprávy. Dále také obsahuje potřebné informace pro správnou publikaci článků jiných autorů. Problematické publikování článků jiných autorů i s jejich svolením jsme se zabývali v sekci zprávy.

Databázový návrh je opět řešen pomocí ORM vrstvy, která databázové tabulky navrhne dle nastavení třídy. Automaticky tedy například vytváří vazební tabulky pro spojení tabulek: články a soubory.



Obrázek 3.5 Návrh třídy článku



### 3.10. Akce

Tento modul je po zprávách / člancích jeden z nejdůležitějších. Jeho návrhu je tedy potřeba věnovat téměř nejvíce času. Akce je na konkurenčních webech řešena základním způsobem, akce tedy převážně obsahuje tyto vlastnosti:

- Název akce
- Datum konání (i vícedenní závody)
- Čas konání (případně rozpis jednotlivých závodů)
- Místo konání (převážně pouze adresa)
- Další soubory (startovní listiny)

Tyto základní informace jsou potřebné pro minimální informaci o konané akci. Pokud bychom se ale zaměřili opět na návštěvníky této sekce, mohli bychom je rozdělit na dvě hlavní skupiny: soutěžící a návštěvníci.

Pokud bychom provedli základní rozbor prvního typu – soutěžící – není pro tyto návštěvníky zcela prioritní místo konání, ani obecný popis závodů. Ale ve většině případů hledají informace o podrobném průběhu závodů. Tyto informace by mohli být uloženy v příložených souborech, ovšem pro tento případ je akci navržena část vlastností, která ulehčí získání těchto informací bez nutnosti procházení oficiálních dokumentů. Tyto vlastnosti jsou:

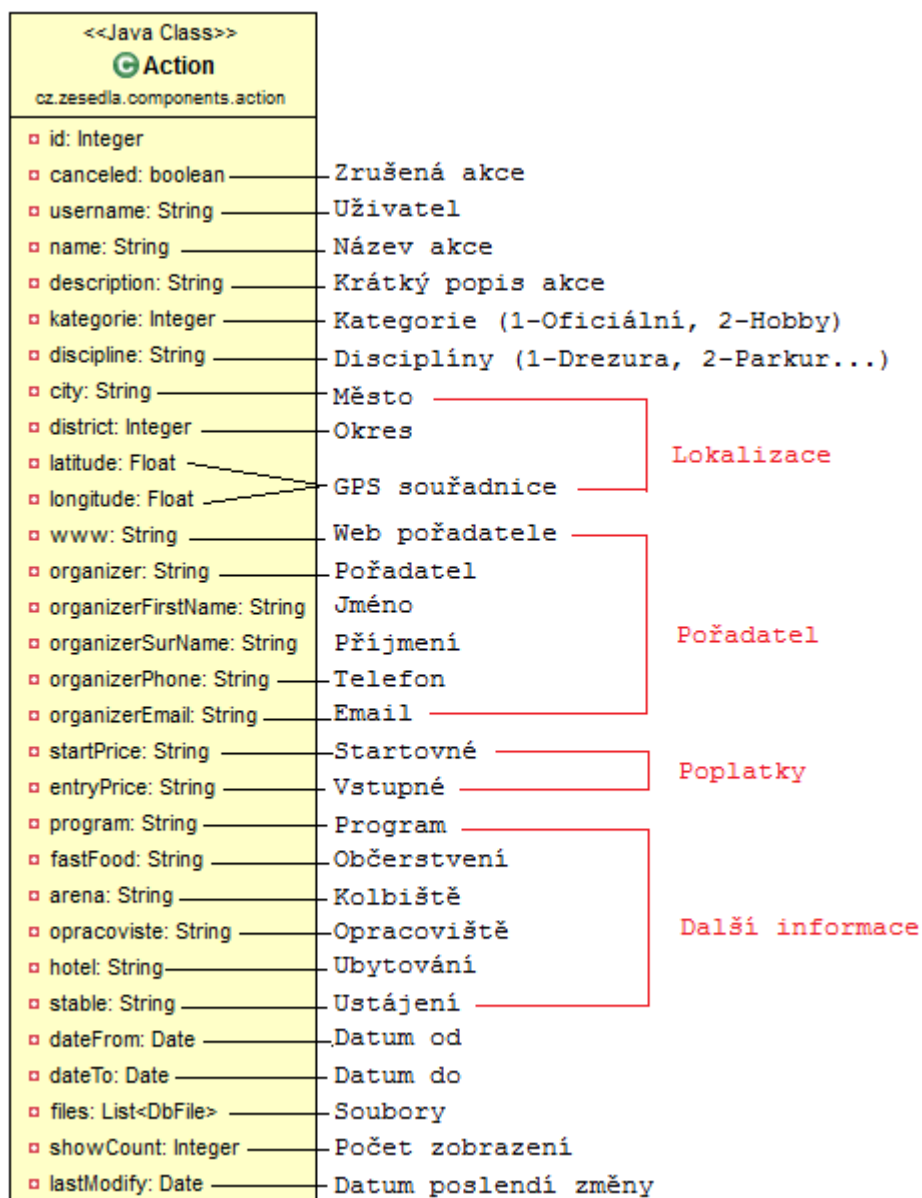
- Poplatky (startovné)
- Program
- Občerstvení
- Kolbiště
- Opracoviště
- Ubytování a ustájení

Druhá uvedená skupina – návštěvníci – naopak ocení jinou skupinu informací, které jim blíže upřesní, o jakou akci se jedná, kdo je pořadatel a kde danou lokalitu naleznou. Pro tuto skupinu byly prioritně vytvořeny tyto vlastnosti:

- Kategorie, disciplína
- Město, okres, GPS souřadnice – i s možností zobrazení v mapě
- Poplatky (vstupné)
- Program, občerstvení
- Termín akce

Standardním uvedením adresy je město a okres. Pokud bychom se ovšem podívali podrobněji na místa pořádaných akcí s jezdeckou tematikou, není tato lokalizace vždy stoprocentní. Ve většině případů by návštěvníci ocenili možnost přesné lokalizace místa, které jsou poměrně často na samotách. Z tohoto důvodu byly do třídy zapracovány GPS souřadnice, pomocí kterých je možné místo přesně lokalizovat a také zobrazit na mapě s vysokou přesností. V tomto modulu se dále využilo obdobné vlastnosti registrovaných uživatelů, kteří pokud mají vyplněné domovské souřadnice (opět GPS), zobrazí se jim routovatelná vzdálenost na konanou akci vypočtená od domovských souřadnic. Pomocí vzdálenosti lze také v seznamu akcí filtrovat.

K akci je také možné přikládat soubory – obrázky a fotky, případně dokumenty (startovní listiny a jiné informace). První vložený obrázek se také zobrazuje jako informační v seznamu a dále v hlavičce akce na jejím detailu.



Obrázek 3.6 Návrh třídy akce

Cílem je projekt dále odlišit od konkurenčních webů, z toho důvodu je pro přehled nadcházejících akcí navrženo dvojí zobrazení. První konkurenčními weby nepoužité řešení je zobrazení v kalendáři. Toto zobrazení nám umožňuje rychleji a přehledněji se orientovat v seznamu nadcházejících akcí. Díky zobrazení v kalendáři ihned návštěvník získá informaci o dni konané akce a také o době její trvání – například vícedenní akce – které jsou v kalendáři zobrazeny přes více dní.

Pro zachování určitého standardu, je možné přepnout se na zobrazení v seznamu, které používají všechny dostupné konkurenční weby. Dle aktuálních ohlasů návštěvníků, je daleko více využíváné nové zobrazení v kalendáři.

#### Koňské akce (Červen 2013)

Pondělí	Úterý	Středa	Čtvrtek	Pátek	Sobota	Neděle
27.	28.	29.	30.	31.	01.	02.
					<ul style="list-style-type: none"> <li>Červnová cena - mistrovství středočes</li> <li>49. ročník Memoriá</li> <li>O Pohár hl. města</li> <li>NCHA Affiliate Gra</li> <li>Drezurní závody Lu</li> <li>Drezurní závody Lu</li> </ul>	<ul style="list-style-type: none"> <li>Jarní jezdecké záv</li> <li>Jezdecké závody v</li> <li>Jezdecké závody v</li> </ul>
03.	04.	05.	06.	07.	08.	09.
		<ul style="list-style-type: none"> <li>Skokové závody K</li> </ul>	<ul style="list-style-type: none"> <li>Velká cena Litomyšle 2013 - IV kolo XVII. ročník Skokový pohár České pojišťovny</li> </ul>		<ul style="list-style-type: none"> <li>Jezdecké závody ve všestrannosti - 48. ročník kvalifikace Z</li> <li>Drezurní závody P</li> <li>Drezurní závody v</li> <li>Drezurní závody P</li> <li>Jezdecké závody v</li> <li>Drezurní závody Heroutice</li> <li>Drezurní závody Heroutice</li> <li>Hobby závody, po</li> <li>Jezdecké hobby z</li> <li>Jezdecké závody f</li> </ul>	<ul style="list-style-type: none"> <li>Hobby závody, po</li> <li>Jezdecké hobby z</li> <li>Jezdecké hobby z</li> </ul>

Obrázek 3.7 Zobrazení akcí v kalendáři

### 3.11. Inzerce

Sekce inzerce se v alespoň nějakém funkčním stavu objevuje na všech konkurenčních serverech. Jejím cílem rozhodně není profitovat z prodeje, tak jako fungují webové servery zaměřené čistě na prodej zboží, ale pouze poskytnout další možnost pro sdružování jedné komunity návštěvníků.

Po podrobnější analýze jednoho z nejnavštěvovanějších webů v české republice s touto tematikou<sup>1</sup> bylo zjištěno, že denně dojde k obnově desítek stávajících inzerátů. To je možné zjistit dle vícedenního pozorování inzerátů a jejich dat vydání. Tato informace má pro tento projekt dva významy:

- Sekce inzerce by mohla opět zvýšit návštěvnost
- Vkládání inzerátů by mělo být určitým způsobem omezeno, aby nedocházelo k neustálému znovu vydávání inzerátů a tím změnou defaultního výpisu řazení dle data vložení

<sup>1</sup> <http://www.equichannel.cz/>

Návrh inzerátu vycházel opět z obecně platných informací pro vytvoření inzerátu. Hlavními vlastnostmi jsou:

- Název
- Text inzerátu
- Kontaktní údaje (telefon, okres, souřadnice)

Pro definování polohy inzerenta byl použit již vytvořený systém použitý například v akcích: je možné zadat město, vybrat okres a pro nejpřesnější určení polohy zadat GPS souřadnice.

Poslední a nejrozsáhlejší část inzerátu je sekce pro třídění zboží. Zde bylo nutné správně sestavit hierarchický strom pro nabízené zboží, který by koncepčně nevybočoval ze zaměření projektu, ale také nebyl tvořený pouhým seznamem.

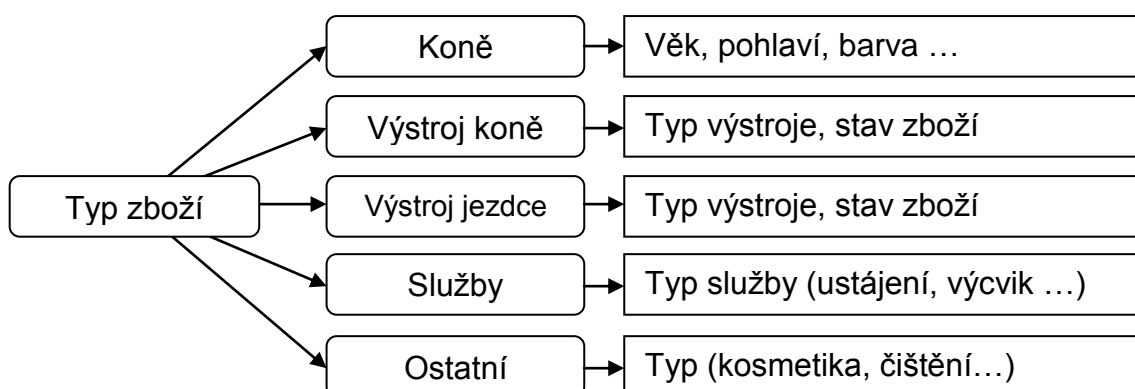
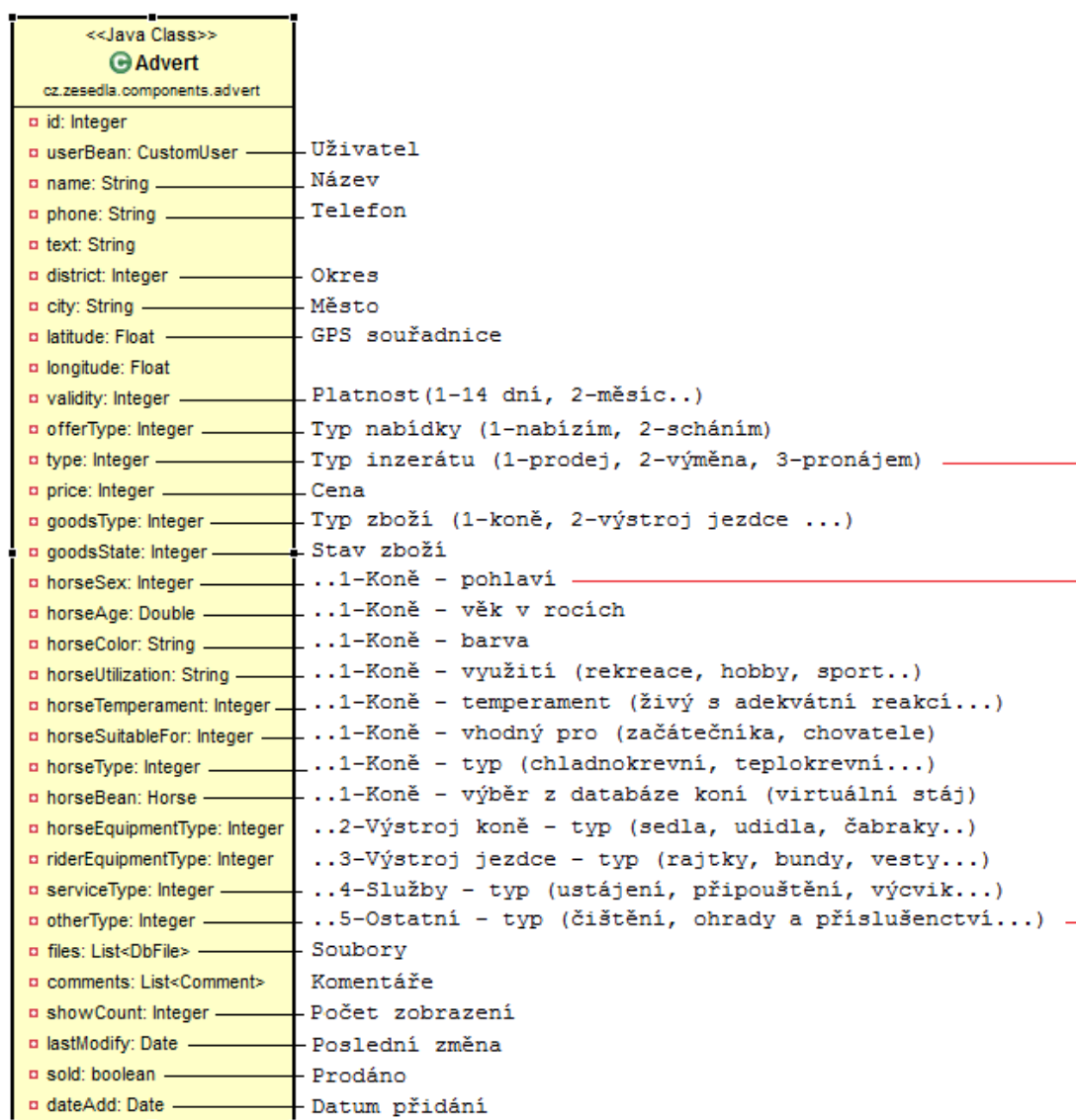


Schéma 3.2 Hierarchické členění typu zboží

Dále je možné nastavit typ nabídky, typ inzerátu, platnost a cenu zboží. Samozřejmostí je možnost připojovat soubory jako u předchozích částí webu.



Obrázek 3.8 Návrh třídy inzerátu

### 3.12. Stáj

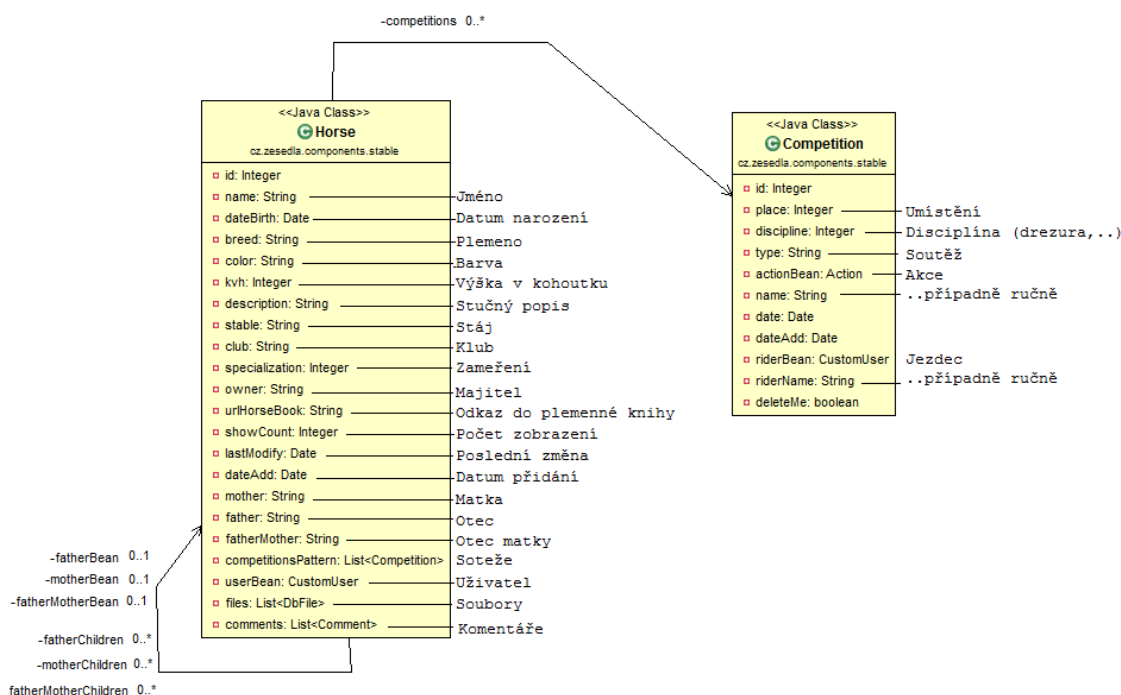
Sekce stáj dále rozšiřuje evidenci uživatele a to o evidenci vlastních či ježděných koní. Cílem této sekce je vytvořit databázi koní i s jejich sportovními úspěchy. Dle teoretického rozdělení sekce stáj obsahuje dvě vzájemně provázané třídy: kůň a soutěž.

Evidenze koně je rozdělena na hlavní část (jméno, plemeno, datum narození atd.), která daného koně blíže specifikuje. Druhá část jsou další informace (stáj, majitel, klub atd.). Třetí část se zaměřuje prvotně na evidenci původu koně (matka, otec, otec

matky) s možností výběru z databáze, či pouze vyplněním jmen. Poslední položka v této části je možnost zadat URL odkaz do plemenné knihy.

Soutěže jsou dynamickým nepovinným prvkem evidence koně. Je tedy možné evidovat koně bez jediné soutěže, je ale také možné přidat hned několik soutěží. Hlavním cílem bylo odlišit koně s evidovanými soutěžemi. Ze zadaných soutěží se tedy počítají dva průměrné indexy:

- Průměrné umístění
- Sportovní index (počet soutěží vzhledem k věku koně)



Obrázek 3.9 Návrh tříd koně a soutěže

### 3.13. O nás

Sekce o nás je již standardně věnována redaktorům webu. Dále je zde zařazena sekce na nabídku spolupráce. Odkazy na redaktory webu jsou také umístěny pro rychlejší orientaci v patičce webu.

### 3.14. Hlavní stránka

Hlavní stránka projektu je složena ze všech výše uvedených sekcí projektu. Největší prioritu má sekce zprávy, je tedy této sekci na hlavní stránce přidělena největší

část – zleva 2/3 stránky. Druhou nejnavštěvovanější částí webu je sekce akce. Pravá horní část hlavní stránky je tedy věnována práci této sekci. V seznamu jsou zobrazeny nejbližší konané akce.

Ve spodní části hlavní stránky je část stránky pro zobrazení článků. Toto zobrazení je totožné s formátem zobrazení zpráv. Inzeráty jsou poté umístěny na pravé straně hlavní stránky ihned pod nejaktuálnějšími akcemi. V seznamu inzerátů je také zobrazena miniatura fotografie. Pod inzeráty je část hlavní stránky se zobrazením nejnovějšího dění ve stáji.

Hlavní stránka také obsahuje uživatelské menu, které je umístěné po přihlášení v horní liště. Dále je možné okázat se z hlavní stránky na profily projektu na sociálních sítích. Pro větší dynamiku estetické stránky webu je možné definovat fotku týdne, která dobarvuje pozadí webu a je umístěna v pravém horní části hned pod odkazy na sociální sítě. Dále se tato stránka zobrazí na pozadí webu, čímž dochází k neustále aktuálnímu vzhledu.

The screenshot shows the website zeSedla.cz with a green navigation bar. The main content is divided into several sections:

- Zprávy (News):**
  - Hlasujte v anketě "Nejkrásnější kůň ČR 2013": Historická plemena** - Article about a survey for the most beautiful horse of the Czech Republic.
  - První český Extreme Horse Park** - Article about an extreme trail for horse riding.
  - Na Ptvřově se bude bojovat o mistrovské tituly v parkurovém skákání** - Article about a show jumping competition.
  - Pro české jezdce začíná v červenci mistrovské období** - Article about the start of the competitive season.
  - Závodů spřežení budou už příští víkend** - Article about harness racing events.
- Články (Articles):**
  - Rodeové disciplíny - Team Penning** - Article about the team penning discipline in rodeo.
- Sleďte nás na sociálních sítích** - Social media follow buttons for Facebook and Twitter.
- Fotka týdne** - A featured photo of a rider on a horse in a natural setting.
- Blížící se akce** - A list of upcoming events:
  - 05.07. pátek: 48. ročník kvalifikace, Závodů spřežení Pardubice, MČR ve skocích Ptvřov, Jezdecké závody Dobrošov, Jezdecké závody Dobrošov, Hobby parkury Dvůr Nová.
  - 06.07. sobota: Oblastní mistrovství, Drezurní závody Mělník, 3. kolo Klatovského poháru, Skokové závody Ústí nad, Jezdecké závody Chrástava, Skokové závody Bohuslavice, Sportovní hobby závody, Mistrovství ČR jízdy, HANÁCKÝ POHÁR.
- Inzerce** - Advertisements:
  - jízda na koni za spoluúčast** - Advertisement for horse riding with a price of 2,000 Kč.
  - Ustájení koní - koňský ráj Praha-Západ, Kladno** - Advertisement for horse stabling in Kladno.

### 3.9 Rozdělení hlavní stránky



### **3.15. Zhodnocení projektu**

Projekt vytvoření zpravodajského serveru o jezdeckví byl splněn. Server je provozován na hostingu a aktuálně vykazuje návštěvnost cca 70-100 unikátních návštěv denně. Jako návrh na vylepšení by nyní mohlo sloužit zaměření se na reklamu a propagaci. Další možností vylepšení návštěvnosti by bylo získání více partnerů, kteří by zpravodajský server zásobovali novými informacemi.

Z hlediska zájmu návštěvníků se jeví nejméně zdařilou sekcí sekce stáj, které i přes stabilní návštěvnost ostatních sekcí, nevykazuje téměř žádnou činnost. Aktuálně by bylo vhodné provést analýzu, z jakého důvodu návštěvníky tato sekce nezaujala. Z hlediska funkčního nebyl nalezen žádný problém, ani chybové kódy. Jako jeden z možných důvodů již může být aktuální popularita sociálních sítí, která naplňuje veškerou potřebu a volný čas uživatelů, kteří chtějí sdílet a prezentovat své informace na internetu. Jedná se pouze o myšlenku autora práce.

### **3.16. Zhodnocení výběru technologie**

Z hlediska náročnosti projektu, jeho celkové provázanosti a dalším technologickým aspektům, by se pro jeho vývoj jevílo neoptimálnější použít PHP. Aktuální zatížení provozovaného serveru také nedosahuje limitních stavů a ani zde není důvod pro použití platformy J2EE. Pokud bychom tedy práci vytvářeli bez výukových aspektů, bylo by vhodnější vyvíjet projekt v PHP.

## 4. Technologie

Tato kapitola představí technologie, které jsou v projektu použity. Cílem této kapitoly je použité technologie představit a pro snazší pochopení doplnit příloženými funkčními miniaplikace, které jsou přiloženy na CD.

### 4.1. Spring framework

Spring framework<sup>1</sup> je open-source framework pro vývoj J2EE aplikací. První informace o tomto framework byla napsána Rodem Johnsonem v rámci publikace jeho knihy Expert One-on-One J2EE Design and Development v roce 2002. Rod Johnson se ve své knize zabývá vývojem J2EE aplikací a stále více upozorňuje na problémy, se kterými se programátoři denně setkávají. Za pomoci Juergena Hoellera je později framework rozšířen a pod názvem Spring Framework poprvé uvolněn jako open-source.

Pro vývoj aplikací ve Spring framework je nejvhodnější použít upravenou verzi studia Eclipse - Spring Tool Suite<sup>2</sup>. Oproti standardnímu Eclipse pro J2EE aplikace Spring Tool Suite nabízí další rozšíření:

- Spring projekty, XML wizards
- Grafický editor konfiguračních souborů
- Rozšířené doplňování kódu, podpora Spring aplikací

Hlavní moduly Spring jsou navrženy tak, aby poměrně rychle umožnily vývojáři přístup i k pokročilým technologiím. Hlavním přínosem je tedy to, že si nemusíme všechny potřebné technologie sami programovat, ale postačí je správně konfigurovat či implementovat jejich části.

---

<sup>1</sup> <http://www.springsource.org/spring-framework>

<sup>2</sup> <http://www.springsource.org/sts>

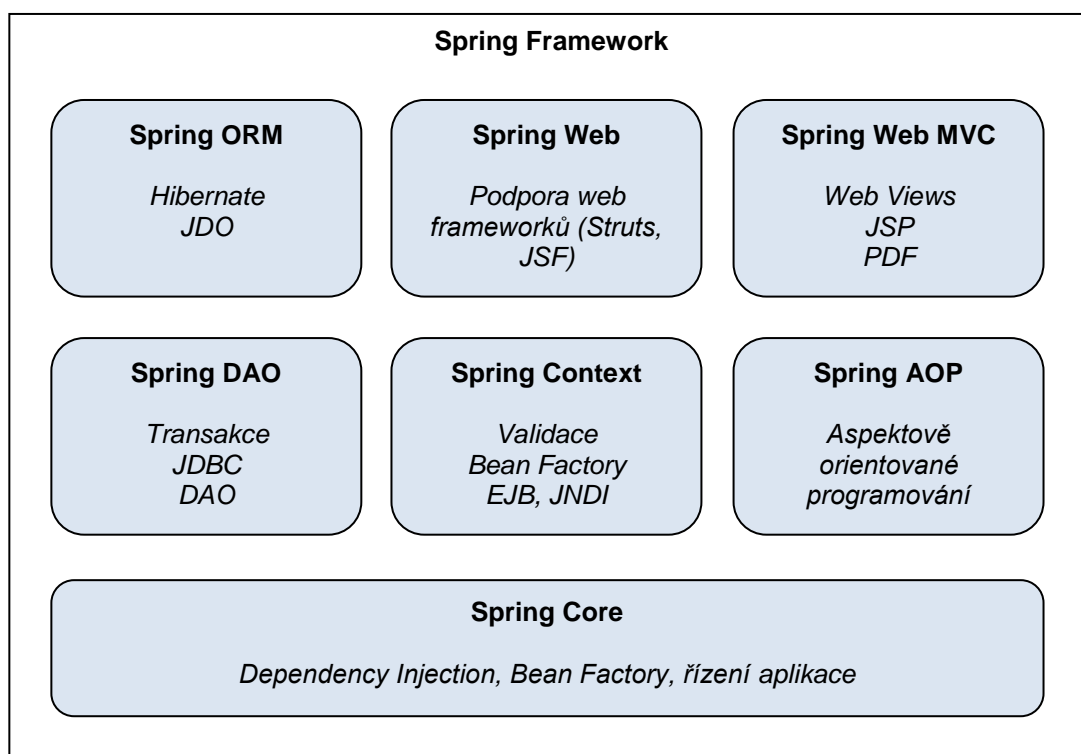


Schéma 4.1 Modulární popis Spring framework

## 4.2. Spring MVC

Spring MVC je aplikační rámec, který se specializuje na návrh zobrazovací vrstvy z pohledu: model – pohled - řadič<sup>1</sup>. Výhodou Spring MVC je přímá implementace vývojáři Spring, čím je zajištěna větší kompatibilita, než například při použití framework Struts.

Spring MVC je tedy možné rozdělit na 3 hlavní části:

- **Model** – je tvořen objekty nesoucími data a ve většině případů také odpovídá tabulkám v databázi
- **Pohled** (view) – převádí data reprezentovaná modelem do podoby vhodné k interaktivnímu zobrazení pro uživatele
- **Řadič** (controller) – reaguje na události pocházející od uživatele a zajišťuje změny v modelu nebo v pohledu

<sup>1</sup> Anglicky model-view-controller, známe pod zkratkou MVC

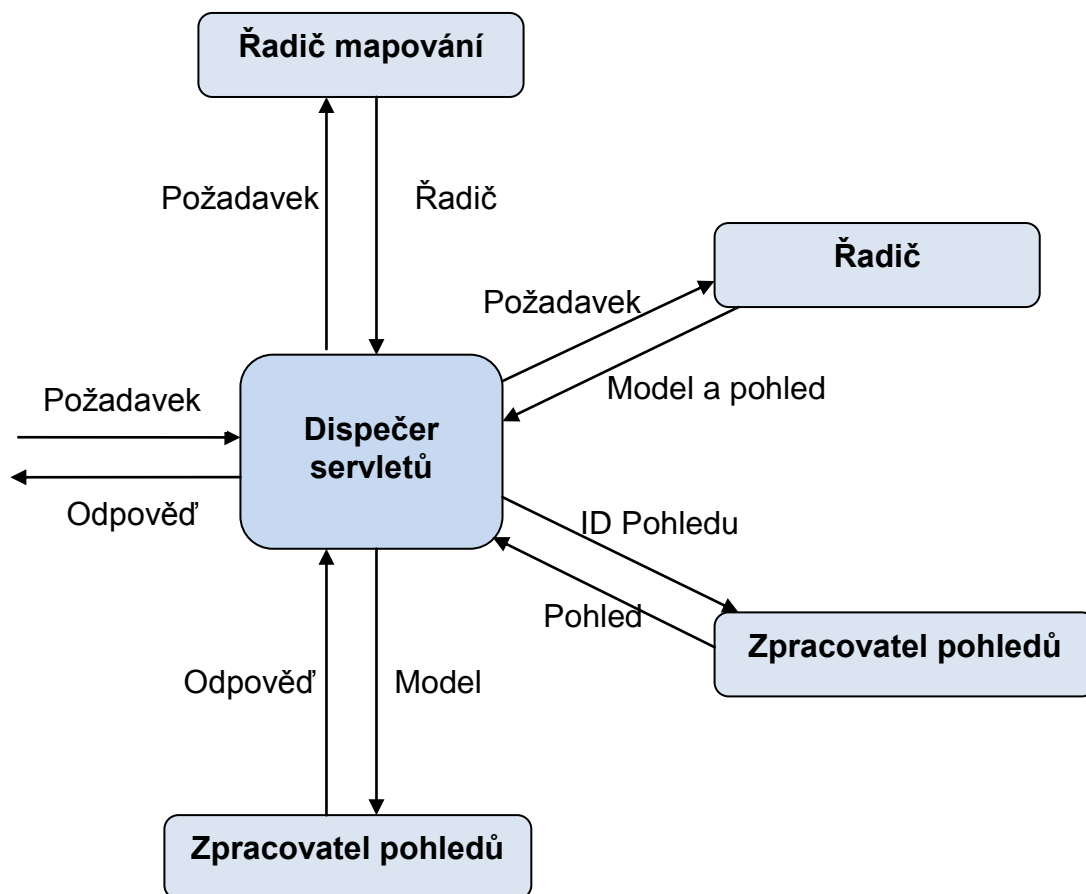


Schéma 4.2 Zpracovávání požadavků Spring MVC

### 4.2.1. Hello Word Spring MVC

Tato výuková miniaplikace obsahuje nejzákladnější možný návrh pomocí metodiky MVC. Cílem této miniaplikace je konfrontovat čtenáře s reálnou ukázkou funkčního programu, aby lépe pochopili princip MVC, který je při práci se Spring v návaznosti na J2EE základní. Na přiloženém CD najdou čtenáři tyto materiály:

[1] STS Projekt Hello Word Spring:

CD\1. Hello Word Spring MVC\Hello\_Word\_Spring\_MVC\\*.\*

[2] Dokumentace k Hello Word Spring:

CD\1. Hello Word Spring MVC\Hello\_Word\_Spring\_MVC.docx

[3] Programová dokumentace k Hello Word Spring:

CD\1. Hello Word Spring MVC\Programová dokumentace\index.html

[4] Spustitelná verze Hello Word Spring:

```
CD\1. Hello Word Spring MVC\root.WAR
```

### 4.2.2. Spring MVC v demonstrátoru

V demonstrátoru není implementována základní verze Views, kterou nabízí Spring. Pro více možností je implementováno Apache Tiles. Toto řešení nabízí možnost definice pohledů pohodlně v XML konfiguračním souboru. Tento konfigurační soubor poté obsahuje veškeré nastavení a kompletní definici možných pohledů. Samozřejmostí je možnost složení výsledného pohledu z více dílčích JSP stránek.

Všechny Views jsou nakonfigurovány v samostatném konfiguračním souboru: `layout.xml`. Tento soubor je poté přiřazen Tiles v konfiguračním souboru: `root-content.xml`. Postup zpracování požadavků a následná práce s pohledy se jinak dále od základní Spring MVC nijak neliší.

## 4.3. Ajax

Ajax (*Asynchronous JavaScript and XML*) je obecné označení technologií vývoje interaktivních webových aplikací, které dynamicky mění obsah stránek bez nutnosti jejich znovunačtení. Na rozdíl od klasických webových aplikací nabízejí v určitých případech uživatelsky příjemnější prostředí. Vhodné použití:

- potřeba načítání malého zlomku dat vzhledem ke komplexní velikosti a složitosti celé stránky
- online ověřování dostupné hodnoty formulářových polí například vůči databázi
- odeslání hlasu z ankety, kdy není nutné provést kompletní přesměrování

Možností vhodného použití je více, je ovšem dobré mít na paměti, že stále zůstávají oblasti, kde se použití Ajax nehodí. Například, pokud bychom prováděli příliš změn již načtené stránky, mohlo by dojít k dezorientaci uživatele, vzhledem k tomu, že URL adresa by se neměnila.

Další nevýhody spíše plynou z četnosti použití. Ne pro každý požadavek se Ajax hodí a ne vždy znamená použití Ajax úsporu komunikačního trafiku. Pokud bychom například načítali celou stránku pomocí Ajax na více částí, výsledný výkon bude z důvodu neustálého navazování spojení horší, než když načteme celou stránku najednou.

### 4.3.1. Hello Word Ajax

Tato miniaplikace ukáže, jakým způsobem je možné pomocí jQuery odeslat Ajax požadavek. Dále ukáže, jak takovýto požadavek ve Spring přijmout a dále s ním pracovat. V poslední řadě je zde také realizováno odeslání odpovědi na přijatý Ajax požadavek. Tato skalce plně zobrazuje komplexní Ajax komunikaci. Pokud tedy čtenář pochopí tuto miniaplikaci, je možné si pro další testování vyzkoušet Ajax komunikaci se složitější strukturou, pouhou změnou odesílaných / přijímaných dat. Na přiloženém CD najdou čtenáři tyto materiály:

[1] STS Projekt Hello Word Ajax:

CD\2. Hello Word Ajax\Hello\_Word\_Ajax\

[2] Dokumentace k Hello Word Ajax:

CD\2. Hello Word Ajax\Hello\_Word\_Ajax.docx

[3] Programová dokumentace k Hello Word Ajax:

CD\2. Hello Word Ajax\Programová dokumentace\index.html

[4] Spustitelná verze Hello Word Ajax:

CD\2. Hello Word Ajax \root.WAR

### 4.3.2. Ajax v demonstrátoru

Ajax je v demonstrátoru používán převážně pro komunikaci se serverem pro ověřování, editaci a ukládání modelů. Jeho použití v některých požadavcích není nezbytné, ovšem koncepce celého projektu využívá Ajax z důvodu udržení jednotnosti.

Spring MVC zatím nenabízí plnou podporu technologie Ajax, je tedy nutné v některých případech používané technologie částečně upravit či implementovat vlastní řešení. Jako příklad můžeme uvést ověření uživatele pomocí Spring Security.

## 4.4. Hibernate

Hibernate je ORM framework pro programovací jazyk Java. Při správném použití, nakonfigurování Hibernate, se již nemusíme starat o ukládání do databáze. Celý proces ukládání, včetně vytváření tabulek s kompletními návaznostmi (one-to-many adt.), nám zajistí ORM framework Hibernate.

Cílem ORM vrstvy je tedy programátora dostatečně odstínit od SQL databáze. Při správné implementaci ORM frameworku můžeme pracovat pouze s objekty a již nemusíme tvořit SQL kód. Obecně by se tedy dalo říci, že i bez znalosti jazyka SQL můžeme pracovat s SQL databází, není to ovšem možné vždy. Pokud se ale budeme pohybovat na úrovni základních operací (přidej, edituj, smaž), není nutné se při použití ORM frameworku SQL dotazy vůbec zabývat.

### 4.4.1. Hello Word Hibernate

Tato miniaplikace simuluje činnost registrace uživatele. Při zakládání uživatele se již seznámíme s validací vstupních dat pomocí Ajax. Takto realizovaná validace nás bude provázet celým projektem. Jedná se tedy o jednu z nejtěžejnějších miniaplikací. Čtenář by v této miniaplikaci měl pochopit práci s databází pomocí ORM frameworku a dále validaci formulářových polí pomocí Ajax požadavku. Jedná se tedy již o komplexnější řešení, než bylo ukázáno v předchozí miniaplikaci Hello Word Ajax. Na příloženém CD najdou čtenáři tyto materiály:

[1] STS Projekt Hello Word Hibernate:

```
CD\3. Hello Word Hibernate\Hello_Word_Hibernate\
```

[2] Dokumentace k Hello Word Hibernate:

```
CD\3. Hello Word Hibernate\Hello_Word_Hibernate.docx
```

[3] Programová dokumentace k Hello Word Hibernate:

```
CD\3. Hello Word Hibernate\Programová  
dokumentace\index.html
```

[4] Spustitelná verze Hello Word Hibernate:

```
CD\3. Hello Word Hibernate\root.WAR
```

## 4.4.2. Hibernate v demonstrátoru

Pro ORM vrstvu se v demonstrátoru používá implementace Hibernate, tak jako v ukázkové miniaplikaci Hello Word Hibernate. V jednotlivých modelech se nastavení Hibernate řeší pomocí anotací. Anotace nám také definují vztahy mezi jednotlivými modely (one-to-many, atd.)

Pro samotné příkazy pro práci s databází jsou v projektu použity jak základní Hibernate příkazy ze `SessionFactory` (`saveOrUpdate`), tak rozšířené pomocí Hibernate criteria. Veškerý zdrojový kód je vždy sjednocen do DAO třídy, jejíž název vychází z názvu modelu. Pro objekty s názvem Model by se jednalo o třídu `ModelDAO`.

## 4.5. jQuery

jQuery je JavaScript knihovna, která vylepšuje základní funkce JavaScript pro práci s HTML. jQuery má kvalitně zpracované interaktivní GUI. Díky jQuery je možné z HTML stránky vytvořit plnohodnotný formulář a to jak s dialogovými okny, tak s aktivními tlačítky, kalendáři a v neposlední řadě i interaktivními tabulkami.

jQuery dále díky své popularitě obsahuje mnoho dalších rozšíření, ať už částečně podporovaných (odkazovaných z oficiálních stránek) nebo zcela oddělených miniaplikací. Díky společnému jádru, je samozřejmostí velká podpora různých verzí prohlížečů.

### 4.5.1. Hello word jQuery

Tyto ukázkové aplikace nás pouze seznámí s celkovou koncepcí jQuery. Jak již bylo uvedeno v úvodu, celkový rozsah jQuery by nebylo možné prezentovat. Cílem těchto miniaplikací je tedy ukázat hlavní grafické prvky jQuery. Všechny projekty jsou koncipovány tak, že není nutné JavaScript knihovny jQuery fyzicky přikládat k projektu, ale je možné je získat online přímo ze serveru jQuery. Je tedy nutné připojení k internetu. Na přiloženém CD najdou čtenáři tyto materiály:

[1] HTML ukázky použití - Hello Word jQuery:

CD\4. Hello Word jQuery\01\_tlacitko.html

CD\4. Hello Word jQuery\02\_zalozky.html



CD\4. Hello Word jQuery\ 03\_accordion.html  
CD\4. Hello Word jQuery\ 04\_autocomplete.html  
CD\4. Hello Word jQuery\ 05\_dialog\_modalni\_formular.html  
CD\4. Hello Word jQuery\ 06\_menu\_ikony.html  
CD\4. Hello Word jQuery\ 07\_progressbar.html  
CD\4. Hello Word jQuery\ 08\_slider\_rozsah\_hodnot.html  
CD\4. Hello Word jQuery\ 09\_tooltip.html  
CD\4. Hello Word jQuery\ 10\_datepicker\_3\_mesice.html

[2] Dokumentace k Hello Word jQuery:

CD\4. Hello Word jQuery\Hello\_Word\_jQuery.docx

## 4.5.2. jQuery v demonstrátor

V demonstrátoru je jQuery knihovna v první řadě využita pro kompletní GUI:

- Kalendář
- Přejížděcí efekty (slide)
- Dialogy
- Tlačítka
- Události (tlačítka, prvky)
- Ajax

Dále jQuery používáme pro kompletní práci s HTML elementy, kde díky rozšířenému regulárnímu vyhledávání můžeme přistupovat k jednotlivým HTML elementům a dále s nimi pracovat.

Dále z knihovny jQuery používáme možnost zasílání Ajax požadavků. Pro tuto funkcionalitu má jQuery připravené konfigurovatelné API. Tato implementace tedy řeší i částečnou nekompatibilitu napříč prohlížeči, která se v řešení Ajax vyskytuje.

## 4.6. Google map API

Mapy Google je internetová mapová aplikace poskytovaná zdarma pro nekomerční použití – za komerční použití by se dalo považovat použití map

pro navigační účely či další prodáváný SW, kde by podpora Google Map byla stěžejní funkcionalitou nabízeného SW. Společnost Google pro práci s mapami vydává a postupně aktualizuje vývojářské API, díky kterému je možné mapy načítat, zobrazovat, vkládat do map různé body a dále s nimi pracovat.

Další ze zajímavých funkcí je možnost plánování a hledání tras. Díky tomu získáme možnost ve své aplikaci pracovat s plnohodnotnými mapovými podklady. Samozřejmostí je i podpora plánovače tras s rozšířeným JavaScript API. Díky Google map API tedy můžeme do své aplikace zapracovat kompletní funkčnost, kterou vidíme na oficiálním webu Google map.

Licenční použití u Google map je v době psaní této práce nastaveno takto: jako uživatel musíte vygenerovat jedinečný token, přes který k mapám přistupujete. Tím si Vás Google označí a může vám počítat počet požadavků. Obecně, pokud byste generovali příliš mnoho požadavků, můžete být od serveru odpojeni, případně by se Váš účet přepnul do placeného režimu. Zajímavé je, že aktuálně veškeré Google map API funguje i bez zadaného identifikačního tokenu. Obecně se ale o této možnosti na Google map API nepíše.

#### **4.6.1. Hello Word Google Map API**

V tomto příkladu je ukázáno, jakým způsobem je možné do webové stránky zapracovat podporu Google map. Ukázkový příklad zobrazí mapu zaměřenou na město Plzeň s přiblížením 10. Cílem této ukázky je tedy předvést čtenářům, že vložení Google map není nic složitého a díky její podpoře mohou svým budoucím uživatelům přinést zajímavý nadstandard. Na přiloženém CD najdou čtenáři tyto materiály:

[1] HTML Projekt Hello Word Google Map API:

CD\5. Hello Word Google Map API\01\_zakladni\_zobrazeni.html

CD\5. Hello Word Google Map API\02\_zadani\_souradnic.html

[2] Dokumentace k Hello Word Google Map API:

CD\5. Hello Word Google Map

API\Hello\_Word\_Google\_Map\_API.pdf

#### 4.6.2. Google Map API v demonstrátoru

Google map API se v projektu využívá k zadávání GPS souřadnic. Díky propojení s jQuery je možné souřadnice vybírat pohodlně z dialogového okna a tím uživatelům umožnit přesnou lokalizaci pomocí pouhého zadávání přes mapové podklady.

Dále se Google map API používá pro výpočet skutečné cestovní vzdálenosti od uživatelových domovských souřadnic k souřadnicím konané akce. Tato možnost zobrazuje skutečně reálnou vzdálenost po pozemních komunikacích dle automaticky vybrané nejvýhodnější cesty.

#### 4.7. Sociální sítě

Sociální sítě jsou služby na internetu, které umožňují registrovaným členům vytvářet své profily, sdílet informace a dále komunikovat s ostatními registrovanými členy. Přestože prvotní cíl sociálních sítí byl pouze sjednotit určité skupiny soukromých lidí, nyní se již původní cíl dále rozšiřuje.

Sociální sítě jsou svým potenciálem zajímavou oblastí i pro komerční společnosti, nekomerční sdružení a další různorodé skupiny. Nejzajímavější oblastí pro tyto skupiny je možnost sdílení informací. Sociální sítě obecně jsou postavené na sdílení informací, velký potenciál ovšem přináší možnost přijímat aktuální informace pouze od svých známých a přátel [Bed11].

Pokud bychom jako společnost vycházeli s předchozí informace, je pro nás velice zajímavé uvažovat tak, že pokud bychom si získali jednoho uživatele, který by dále sdílel naše informace, je zde možnost oslovit sdílenými informacemi uživatelovo známé a přátele. Ovšem nikoliv neosobně pomocí placené reklamy, ale cíleně přes uživatele, kterého všichni jeho známí a přátele dobře znají. Tím získá sdílená informace na větší důvěryhodnosti.

Z tohoto základního předpokladu vyplývá důležitost a také potenciál firemních profilů na sociálních sítích a jeho potvrzením jsou také finance utracené za propagaci

na sociálních sítích v roce 2012 - konkrétně 16,9 miliard dolarů, kde více jak polovina je tvořena příjmy z reklam – placené propagace.

Pro tento projekt, má podpora sociálních sítí zajímavý potenciál a to jak při vzniku – pro šíření prvotní informace o vzniku nového projektu, tak během provozu – pro sdílení aktuálních informací o aktuálním dění (nové články, akce, ankety a jiné). Obě možnosti, by projektu mohli přinést zvýšenou návštěvnost.

#### **4.7.1. Sociální plugin**

Sociální plugin jsou malé webové aplikace, které nám umožní implementovat funkce sociálních sítí na náš web. Díky těmto miniaplikacím, můžeme propojit náš web se sociální sítí a plnohodnotně tak využít potenciál těchto sítí.

Z pohledu provozovatelů sociálních sítí, je jejich maximální snahou co možná nejpohodlněji a nejjednodušeji zpřístupnit potřebné API, aby je i ne zcela technicky zdatný uživatel mohl integrovat do svého webu.

Pokud bychom se ale na veřejné API většiny známých sociálních sítí podívali podrobněji, nabízí nám poměrně dost miniaplikací, které můžeme do svého projektu integrovat. Integrace některých částí již většinou potřebuje větší zásah webového projektu a ne vždy postačí pouhé vložení JavaScript či HTML kódu – například tlačítko umožňující přihlášení uživatelů přes Facebook.

#### **4.7.2. Facebook**

Facebook je sociální síť, která byla spuštěna v roce 2004. Jeho zakladatelem je Mark Zuckerberg. Aktuálně se jedná o nejpoužívanější sociální síť na světě a také v ČR. Za první čtvrtletí roku 2013 má Facebook 700 miliónů denně aktivních uživatelů. V ČR se počet denně aktivních uživatelů odhaduje na 3,5 milionu.

Tato informace s tímto počtem denně aktivních uživatelů v sobě skrývá obrovský marketingový potenciál. Z tohoto důvodu je aktuálně sociální síť Facebook pro jakýkoliv projekt nejdůležitější a její podpora by se dala považovat za základní podporu sociálních sítí.

Podpora Facebook na webovém projektu by se dala rozdělit do několika částí, které jsou níže popsány a také dále obsahují i ukázky programového kódu pro jejich implementaci.

### Tlačítko „to se mi líbí“

Tlačítko „to se mi líbí“, nám umožňuje dát hlas určité stránce, společnosti či skupině. Toto tlačítko je generováno pomocí vloženého řádkového kódu, který je možné získat na stránkách Facebook. Umístění tlačítka je standardizováno na hlavní stránku převážně do její hlavičky či do horních okrajů. Použití tlačítka v patičce stránky je nejméně používané a také nejméně vhodné.

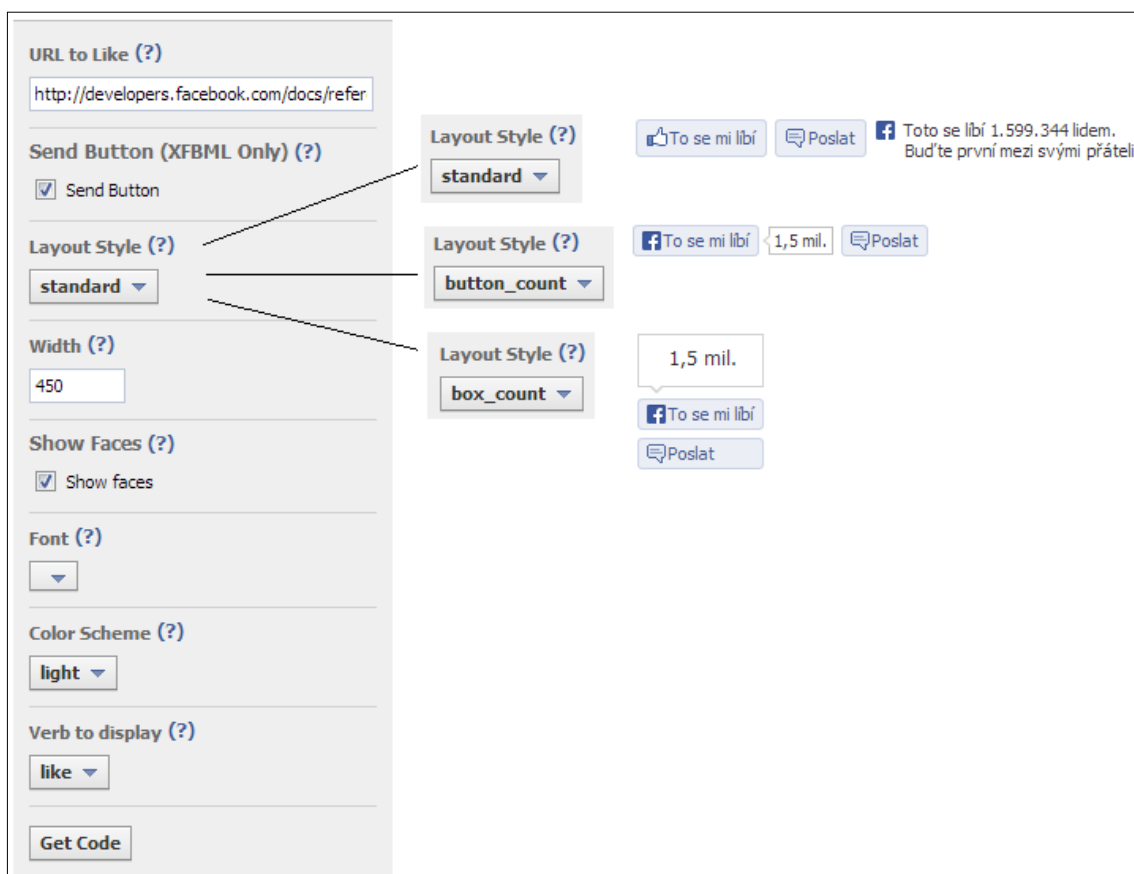
Při návrhu tlačítka máme možnost částečně upravit jeho vzhled (viz obrázek 5.1). Hlavní vlastnosti jsou šířka, velikost, barevné schéma a font. Po nastavení příslušných vlastností a odkazované stránky na Facebook můžeme vygenerovat kód.

```
<div id="fb-root"></div>
<script>
// Funkce pro vytvoření skriptu, který generuje tlačítko
(function(d, s, id) {
  // Načteme první skript, pro umístění nového skriptu
  var js, fjs = d.getElementsByTagName(s)[0];
  // Pokud již skript se vstupní ID existuje - nepokračuje
  if (d.getElementById(id)) return;
  // Vytvoří skript a nastaví vstupní ID
  js = d.createElement(s); js.id = id;
  // Nastaví adresu skriptu
  js.src =
"//connect.facebook.net/cs_CZ/all.js#xfbml=1&appId=1129446021209
17";
  // Vloží skript do stránky
  fjs.parentNode.insertBefore(js, fjs);
}
// Spustí výše uvedenou funkci
(document, 'script', 'facebook-jssdk'));
</script>
```

Výpis zdrojového kódu 4.1 Facebook - tlačítko to se mi líbí

```
<div class="fb-like" data-send="true" data-width="450" data-
show-faces="true"></div>
```

Výpis zdrojového kódu 4.2 Facebook - element pro umístění tlačítka to se mi líbí



Obrázek 4.1 Facebook - návrh tlačítka to se mi líbí

### Tlačítko odeslat

Tlačítko odeslat nám umožňuje na jedno kliknutí odeslat odkaz na profil. Opět tedy pouze usnadňuje práci uživatelům, kteří chtějí sdílet internetový obsah přes svůj profil na Facebook. Pokud si tedy vybereme vzhled a styl tlačítka (viz obrázek 4.2) program nám vygeneruje HTML kód, který si již vložíme do své stránky.

```

<div id="fb-root"></div>
<script>
// Funkce pro vytvoření skriptu, který generuje tlačítko
(function(d, s, id) {
  // Načteme první skript, pro umístění nového
  var js, fjs = d.getElementsByTagName(s)[0];
  // Pokud již skript se vstupní ID existuje - nepokračuje
  if (d.getElementById(id)) return;
  // Vytvoří skript a nastaví vstupní ID
  js = d.createElement(s); js.id = id;
  // Nastaví adresu skriptu
  js.src =
  "//connect.facebook.net/cs_CZ/all.js#xfbml=1&appId=1129446021209
  17";
  // Vloží skript do stránky
  fjs.parentNode.insertBefore(js, fjs);
}
// Spustí výše uvedenou funkci
(document, 'script', 'facebook-jssdk'));
</script>

```

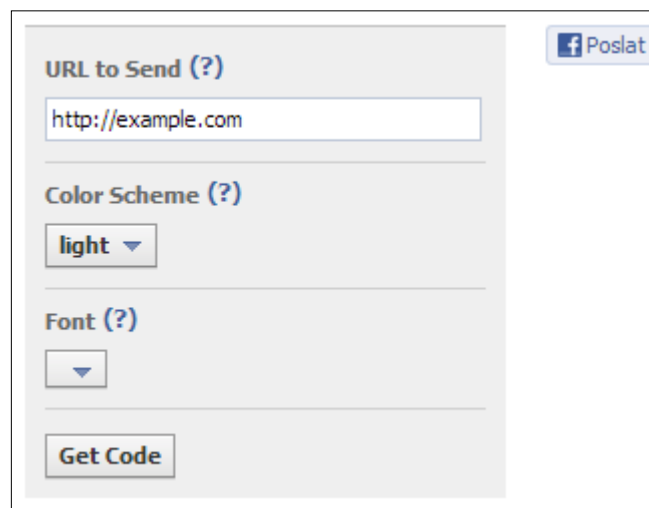
Výpis zdrojového kódu 4.3 Facebook - tlačítko odeslat

```

<div class="fb-send" data-href="http://example.com"></div>

```

Výpis zdrojového kódu 4.4 Facebook - element pro umístění tlačítka odeslat



Obrázek 4.2 Facebook - návrh tlačítka odeslat

## Plugin pro komentáře

Plugin pro komentáře nám umožňuje do projektu vložit aplikaci na vkládání komentářů bez nutnosti jakékoliv databázové evidence v projektu. Veškeré informace o komentářích a uživateli jsou uchovávány na sociální síti Facebook. Implementace komentářů začíná návrhem mini aplikace, kde si vybereme vzhled a styl (viz obrázek 4.3). Program nám dále vygeneruje HTML kód, který si již vložíme do své stránky.

```
<div id="fb-root"></div>
<script>
// Funkce pro vytvoření skriptu, který generuje tlačítko
(function(d, s, id) {
  // Načteme první skript, pro umístění nového
  var js, fjs = d.getElementsByTagName(s)[0];
  // Pokud již skript se vstupní ID existuje - nepokračuje
  if (d.getElementById(id)) return;
  // Pokud již skript se vstupní ID existuje - nepokračuje
  js = d.createElement(s); js.id = id;
  // Nastaví adresu skriptu
  js.src =
  "//connect.facebook.net/cs_CZ/all.js#xfbml=1&appId=112944602120917";
  // Vloží skript do stránky
  fjs.parentNode.insertBefore(js, fjs);
}
// Spustí výše uvedenou funkci
(document, 'script', 'facebook-jssdk'));
</script>
```

Výpis zdrojového kódu 4.5 Facebook - plugin pro komentáře

```
<div class="fb-comments" data-href="http://example.com" data-width="470" data-num-posts="10"></div>
```

Výpis zdrojového kódu 4.6 Facebook - element pro umístění komentářů





Obrázek 4.3 Facebook - návrh pluginu pro komentáře

Dále je možné od Facebook integrovat:

- Tlačítko sleduj – umožňuje sledovat příspěvky sledovaných uživatelů
- Nedávné aktivity přátel – zobrazuje nedávné aktivity přátel
- Doporučení přátel – informuje, kolik přátel stránku doporučuje
- Like box – rozšířená varianta Like button
- Tlačítko na přihlášení pomocí Facebook
- Plugin pro registrace – registrační formulář Facebook

### 4.7.3. Twitter

Twitter je další z řady oblíbených sociálních sítí. Twitter byl založen v roce 2006 třemi zakladateli J. Dorseyem, E. Williamsem a B. Stonem. Aktuálně je Twitter třetí nepoužívanější sociální síť za zcela jasně první sociální sítí Facebook. o druhé a třetí místo Twitter neustále soutěží se sociální sítí Google+. Twitter aktuálně navštíví 228 miliónů aktivních uživatelů. V České republice je počet denně aktivních uživatelů Twitter na 161 tisíc. Ve srovnání s nejpoblárnější sociální sítí Facebook, se jedná pouze o zlomek aktivních uživatelů ČR.

na této sociální síti můžete publikovat takzvané „tweety“ neboli příspěvky o délce 160 znaků. Jejím hlavním cílem je tedy rozšířit krátké zprávy s případným vloženým odkazem. Díky těmto omezením je možné šířit krátké a jasné informace bez

možnosti grafických úprav a dalších grafických prvků. Twitter je jedinou populární sociální sítí, která pracuje na tomto principu krátkých zpráv.

Twitter nabízí opět mnoho aplikací, které je možné vkládat do webových projektů. Níže jsou podrobněji uvedeny nejdůležitější z nich.

### Tlačítko Tweet

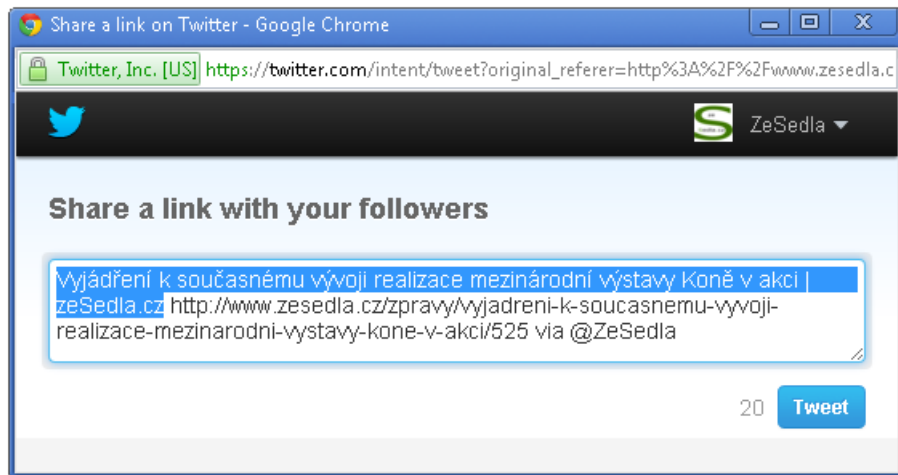
Tlačítko Tweet nám umožňuje odeslat zajímavou URL stránku na Twitter profil. Tlačítko se do webu vkládá pomocí vloženého JavaScript, který už obsahuje veškeré nastavení tlačítka: velikost, typ a další volitelné atributy. Pokud si tedy vybereme vzhled a styl tlačítka, program nám vygeneruje HTML kód, který si již vložíme do své stránky.

```
<script>
// Funkce pro vytvoření skriptu, který generuje tlačítko
!function(d,s,id)
{
  // Načteme první skript, pro umístění nového
  var js, fjs=d.getElementsByTagName(s)[0],
  p=/^http:/.test(d.location)?'http':'https';
  // Pokud již skript se vstupní ID existuje - nepokračuje
  if(!d.getElementById(id)){
    // Vytvoří skript a nastaví vstupní ID
    js=d.createElement(s);
    js.id=id;
    // Nastaví adresu skriptu
    js.src=p+'://platform.twitter.com/widgets.js';
    // Vloží skript do stránky
    fjs.parentNode.insertBefore(js,fjs);
  }
}
// Spustí výše uvedenou funkci
document, 'script', 'twitter-wjs');
</script>
```

Výpis zdrojového kódu 4.7 Twitter - tlačítko tweet

```
<a href="https://twitter.com/share" class="twitter-share-button"
data-via="ZeSedla">Tweet</a>
```

Výpis zdrojového kódu 4.8 Twitter - element pro umístění tlačítka



Obrázek 4.4 Twitter - sdílení obsahu webové stránky pomocí tlačítka Tweet

### Tlačítko sleduj

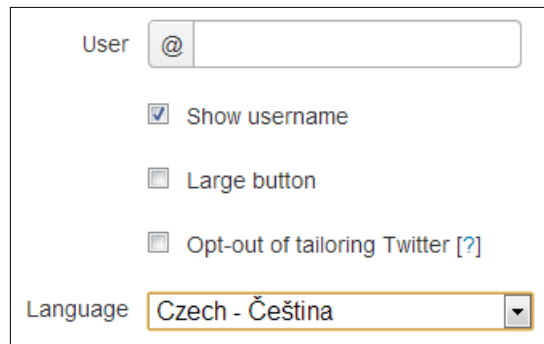
Tlačítko Sleduj nám umožňuje sledovat určitou stránku / profil na Twitter. Jeho cílem je tedy usnadnit sledování obsahu konkrétního projektu přímo z jeho stránek, bez nutnosti vyhledávání Twitter profilu. Má tedy pouze jediné využití a to zvyšovat počet sledujících uživatelů. Pokud si tedy vybereme vzhled a styl tlačítka (viz obrázek 4.5) program nám vygeneruje HTML kód, který si již vložíme do své stránky.

```
<script>
// Funkce pro vytvoření skriptu, který generuje tlačítko
!function(d,s,id)
{
  // Načteme první skript, pro umístění nového
  var js, fjs=d.getElementsByTagName(s)[0],
  p=/^http:/.test(d.location)?'http':'https';
  // Pokud již skript se vstupní ID existuje - nepokračuje
  if(!d.getElementById(id)){
    // Vytvoří skript a nastaví vstupní ID
    js=d.createElement(s);
    js.id=id;
    // Nastaví adresu skriptu
    js.src=p+'://platform.twitter.com/widgets.js';
    // Vloží skript do stránky
    fjs.parentNode.insertBefore(js,fjs);
  }
}
// Spustí výše uvedenou funkci
(document, 'script', 'twitter-wjs');
</script>
```

Výpis zdrojového kódu 4.9 Twitter - tlačítko sleduj

```
<a href="https://twitter.com/ZeSedla" class="twitter-follow-button" data-show-count="false" data-lang="cs">Sledovat @ZeSedla</a>
```

Výpis zdrojového kódu 4.10 Twitter - element pro umístění tlačítka sleduj



Obrázek 4.5 Twitter - návrh tlačítka Follow

Další pluginy Twitter:

- Twitter Cards – umožňují připojit multimediální obsah
- Embedded Timelines – časová osa příspěvků
- Embedded Tweets - vložené tweety umožní citovat přímý zdroj například prostřednictvím HTML kódu

#### 4.7.4. Google+

Google+ je nejmladší sociální síť, byl založen v roce 2011. Aktuálně se umístil Google+ s počtem uživatelů na druhém místě a na třetí posunul výše zmíněnou sociální síť Twitter. Aktivní uživatelé této sítě činí 359 miliónů. Využití a také firemní použití této sociální sítě prozatím není zcela jasné a dochází k poměrně velkému rozporu názorů co na Google+ zveřejňovat. Většina společností se při aktualizaci profilu na Google+ přiklání ke zveřejňování stejných dat jako na Facebooku. Počet aktivních uživatelů pro Českou republiku není uveden.

## +1 Button

Tlačítko +1 přidává +1 webovým stránkám, podobně jako tlačítko „*to se mi líbí*“ od společnosti Facebook. Dle některých názorů by tlačítko +1 mělo mít také vliv na vyhledávání a indexování. Teoreticky by stránka s více +1 měla být ve vyhledávání na Google upřednostňována. Přesná specifikace ovšem neexistuje. Pokud si tedy vybereme vzhled a styl tlačítka (viz obrázek 4.6) program nám vygeneruje HTML kód, který si již vložíme do své stránky.

```
<script type="text/javascript">
// Funkce pro vytvoření skriptu, který generuje tlačítko
(function() {
  // Vytvoří skript
  var po = document.createElement('script');
  // Nastaví parametry skriptu
  po.type = 'text/javascript'; po.async = true;
  // Nastaví adresu skriptu
  po.src = 'https://apis.google.com/js/plusone.js';
  // Načteme první skript, pro umístění nového
  var s = document.getElementsByTagName('script')[0];
  // Vloží skript do stránky
  s.parentNode.insertBefore(po, s);
})
// Spustí výše uvedenou funkci
();
</script>
```

Výpis zdrojového kódu 4.11 Google+ - tlačítko +1

```
<div class="g-plusone" data-annotation="inline" data-
width="300"></div>
```

Výpis zdrojového kódu 4.12 Google+ - element pro umístění tlačítka +1

<p>Velikost <input type="button" value="g+1"/> <input type="button" value="g+1"/> <input type="button" value="g+1"/> <input type="button" value="g+1"/></p> <p>Poznámka <input type="text" value="v řádku"/></p> <p>Šířka <input type="text" value="300"/></p> <p>Jazyk <input type="text" value="čeština"/></p>	<p>▼ Rozšířené možnosti</p> <p>Asynchronní <input checked="" type="checkbox"/></p> <p>Platná syntaxe HTML5 <input checked="" type="checkbox"/></p> <p>Analyzovat značky <input checked="" type="radio"/> Výchozí (při načtení) <input type="radio"/> Explicitní</p> <p>Funkce zpětného volání JS <input type="text"/></p> <p>+1 pro adresu URL <input type="text"/></p>
--	---

Obrázek 4.6 Google+ - grafický návrh tlačítka

### Tlačítko sdílej

Tlačítko *sdílej* se používá k odeslání URL adresy profil uživatele. Umístění tlačítka si můžete zvolit pomocí do řádku vloženého HTML kódu. Obecně bychom mohli používat pouze tlačítko +1, které nám jako uživatelům umožní takto ohodnocený obsah sdílet na svém profilu. Toto tlačítko to ovšem zvládne v jednom kroku. Pokud si tedy vybereme vzhled a styl tlačítka (viz obrázek 4.7) program nám vygeneruje HTML kód, který si již vložíme do své stránky.

```
<script type="text/javascript">
// Funkce pro vytvoření skriptu, který generuje tlačítko
(function() {
  // Vytvoří skript
  var po = document.createElement('script');
  // Nastaví parametry skriptu
  po.type = 'text/javascript'; po.async = true;
  // Nastaví adresu skriptu
  po.src = 'https://apis.google.com/js/plusone.js';
  // Načteme první skript, pro umístění nového
  var s = document.getElementsByTagName('script')[0];
  // Vloží skript do stránky
  s.parentNode.insertBefore(po, s);
})
// Spustí výše uvedenou funkci

();
</script>
```

Výpis zdrojového kódu 4.13 Google+ - tlačítko sdílej

```
<div class="g-plus" data-action="share"></div>
```

Výpis zdrojového kódu 4.14 Google+ - element pro umístění tlačítka sdílej

Poznámka <input type="text" value="V řádku"/>	Rozšířené možnosti
Velikost <input type="radio"/> Malá <input checked="" type="radio"/> Střední <input type="radio"/> Velká	Šířka <input type="text"/>
Jazyk <input type="text" value="čeština"/>	Asynchronní <input checked="" type="checkbox"/>
<input type="button" value="G+ Sdílet"/> +2804037 Sdílet na Google+	Platná syntaxe HTML5 <input checked="" type="checkbox"/>
	Analyzovat značky <input checked="" type="radio"/> Výchozí (při načtení) <input type="radio"/> Explicitní
	Adresa URL ke sdílení <input type="text"/>

Obrázek 4.7 Google+ - grafický návrh tlačítka sdílej

### Dále nám Google+ nabízí:

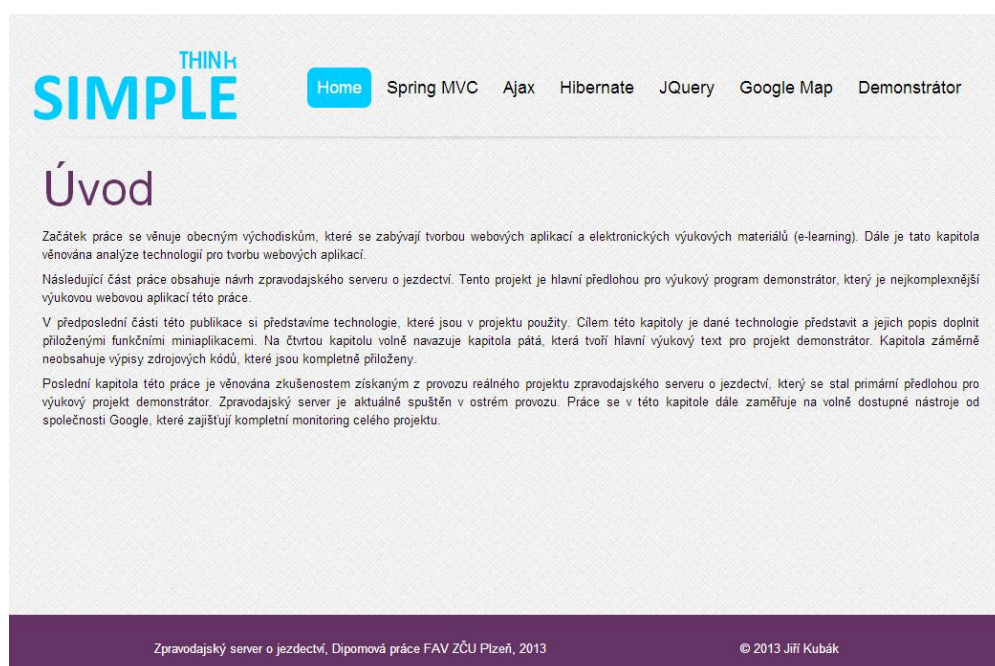
- Badge – zobrazí zvětšený profil Google+ účtu
- Interaktivní příspěvky – umožňují uživatelům sdílet obsah webu se připojením svých komentářů a jiných úprav
- Tlačítko na přihlášení
- Snippet – umožňuje vybrat zobrazovaný úryvek sdílené stránky, například ten, který sdílenou stránku nejlépe vystihuje

## 5. Demonstrátor

Tato kapitola obsahuje uživatelskou část dokumentace k výslednému komplexnímu projektu demonstrátor. Cílem této kapitoly je poskytnout čtenářům plnohodnotného průvodce pro konfiguraci a zprovoznění demonstrátoru a pochopení stěžejních částí celého projektu. Celá kapitola záměrně obsahuje vždy napřed naznačení fungování, poté ukázkou nejdůležitějšího zdrojového kódu a v poslední řadě informace o tom, kde je daná funkcionality použita. Tento postup byl navržen dle teoretického rozboru.

### 5.1. Výukové aspekty

Ve druhé kapitole této práce, pojednávající o e-learningu, bylo možné se o této vyučovací metodě dozvědět její podstatu a funkci v pedagogice. Veškeré praktické, ukázkové materiály a doprovodné texty, budou v souvislosti touto výukovou metodou umístěné na internetu, aby si studenti mohli vyhledat potřebné informace a podklady pro své samostudium. Veškeré programové kódy jsou také doplněny o vysvětlující komentáře, které přesahují rozsah běžných programových komentářů. Toto rozšíření nabízí možnost s projektem samostatně pracovat.



Obrázek 5.1 Náhled webu pro stahování elektronických materiálů



## 5.2. Připojení k databázi

Jak již bylo zmíněno v kapitole 4.4 Hibernate, případně ukázáno v mini aplikaci Hello Word Hibernate, k připojení do databáze se pro ORM vrstvu používá implementace Hibernate. Při použití Hibernate můžeme definovat dialekt, který nám určuje typ databáze. Při jeho změně je teoreticky možné přejít na jinou databázi bez jakékoliv změny zdrojového kódu. Samotné nastavení datového zdroje je umístěno v odděleném v konfiguračním souboru: *jdbc.properties*. Tento soubor obsahuje veškeré konfigurační hodnoty potřebné k připojení k datovému zdroji. Níže uvedené nastavení je poplatné databázi MySQL 5 a vyšší.

```
/* Nastavení ovladače pro připojení k databázi */
jdbc.driverClassName=com.mysql.jdbc.Driver
/* Nastavení dialektu - dle typu databáze */
jdbc.dialect=org.hibernate.dialect.MySQL5Dialect
/* Nastavení adresy databáze */
jdbc.databaseurl=jdbc:mysql://localhost:3306/zesedla
/* Přihlašovací údaje k databázi */
jdbc.username=root
jdbc.password=admin
```

### Výpis zdrojového kódu 5.1 Demonstrátor - konfigurace připojení k databázi

Samotné připojení do databáze nám zprostředkovává třída Spring `org.springframework.jdbc.datasource.DriverManagerDataSource`. Toto nastavení můžeme najít v souboru `root-context.xml`. Dále v souboru `root-context.xml` najdeme vytvoření `SessionFactory`, které nám do DAO tříd vkládá objekt s odkazem na databázi. Díky tomuto objektu můžeme z každé DAO třídy pracovat s databází bez nutnosti ručního předávání. Objekt má napříč celým systémem jednotný název: `SessionFactory` a jedná se o implementaci třídy: `org.springframework.orm.hibernate4.LocalSessionFactoryBean`. Pro jeho vytvoření je použit objekt `dataSource`, který nese informace o připojení k databázi. Inicializace `dataSource` probíhá za pomoci pomocného objektu `PropertyPlaceholderConfigurer`, který nám zajistí automatické načtení nadefinovaných parametrů z výše uvedeného souboru *jdbc.properties*. Díky tomuto způsobu načtení je možné se dále v konfiguračních souborech na tyto proměnné odkazovat pomocí anotací:

- `${jdbc.driverClassName}`

- `${jdbc.databaseurl}`
- `${jdbc.username}`
- `${jdbc.password}`

Vždy se jedná o první název souboru před tečkou a dále název proměnné uvedené v souboru. Díky tomuto způsobu odkazování je možné vytvořit jakýkoliv konfigurační soubor, díky němuž je možné vytvářet různé instalace bez nutnosti změny konfiguračních souborů.

```
<!-- Načítá nastavení ze souboru jdbc.properties do proměnných
${jdbc.driverClassName} atd. -->
<bean id="propertyConfigurer"
class=".beans.factory.config.PropertyPlaceholderConfigurer" >
  <property name="location"
    value="/WEB-INF/config/properties/jdbc.properties">
  </property>
</bean>
```

```
<!-- Vytvoření DataSource, který odkazuje na databázi a dále
předává informace o připojení -->
<bean id="dataSource"
class=".jdbc.datasource.DriverManagerDataSource">
  <property name="driverClassName"
value="${jdbc.driverClassName}"/>
  <property name="url" value="${jdbc.databaseurl}"/>
  <property name="username" value="${jdbc.username}"/>
  <property name="password" value="${jdbc.password}"/>
</bean>
```

Výpis zdrojového kódu 5.2 Demonstrator - předání parametrů připojení k databázi DS

### 5.3. Validace formulářových polí pomocí Ajax

V demonstrátoru používáme jednotný postup pro validaci formulářových polí založený na technologii Ajax. Spring prozatím plně verifikaci formulářových polí pomocí Ajax nepodporuje, museli jsme si část metod sami naprogramovat.

---

\* org.springframework

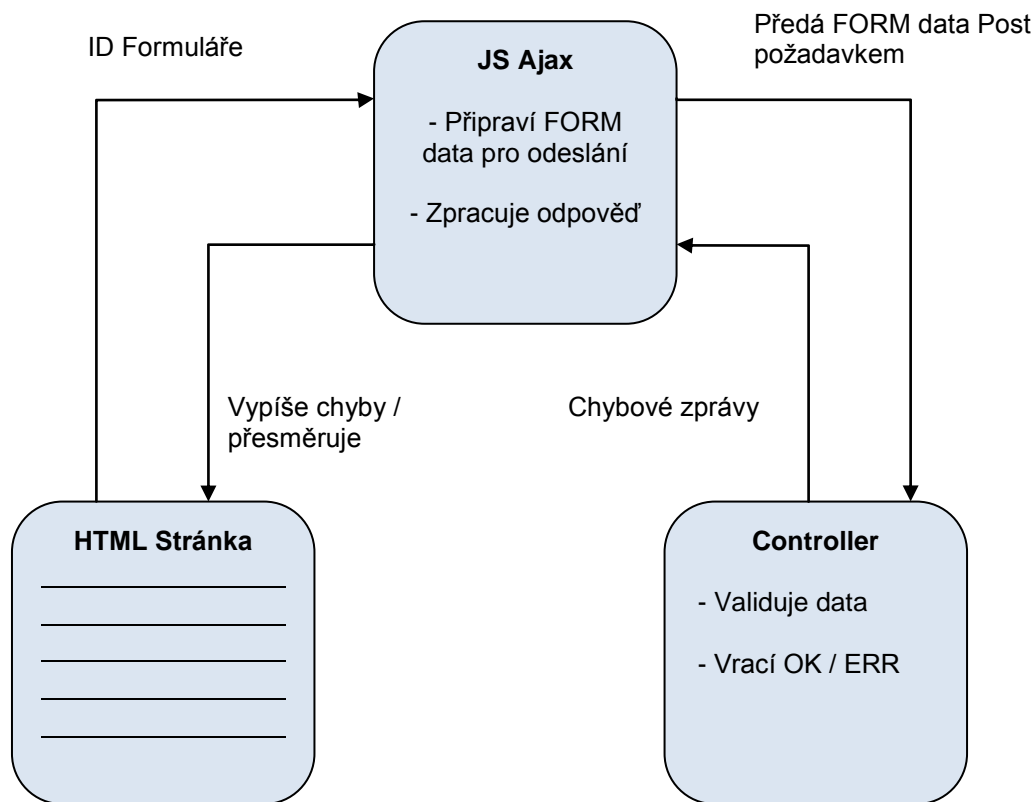


Schéma 5.1 Model validace dat pomocí Ajax

Na straně JavaScript se na zpracování dat a odeslání Ajax požadavku používá knihovna jQuery, která se tímto postará o celý proces vyslání Ajax požadavku a to napříč různými typy prohlížečů. Nám tedy stačí využít předpřipravené funkce pro vrácení dat, případně ošetření chybových stavů komunikace. Veškeré API najdeme v souboru `api.js`.

Řadič přijatá data validuje. Validace v řadiči probíhá ve dvou krocích. Případné první chyby vrací proměnná `BindingResult`, kterou nám Spring předává již v parametrech metody. Tato proměnná může obsahovat chyby při konverzi formátů:

- `typeMismatch.java.lang.Integer`
- `typeMismatch.java.lang.Double`

Druhý krok validace již validuje formát a rozsah proměnných modelu. Validáční nastavení můžeme najít přímo u proměnných v modelu:

```

@Entity
@Table(name="action")
public class Action implements Serializable
{
    @Transient
    private static final long serialVersionUID = 1L;

    @Id
    @Column(name="action_id")
    @GeneratedValue(strategy=GenerationType.AUTO)
    private Integer id;

    @Column(name="canceled")
    private boolean canceled;

    @Column(name="username")
    @NotEmpty(message="Pole je povinné")
    private String username;

    @Column(name="name")
    @Length(min=10, max=250, message="Pole musí mít od 10 do 250 znaků")
    private String name;

    @Column(name="description", length=1000)
    @Length(min=10, max=1000, message="Pole musí mít od 10 do 1000 znaků")
    private String description;

    @Column(name="kategorie")
    @NotNull(message="Pole je povinné")
    @Min(value = 1, message="Pole je povinné")
    private Integer kategorie;
}

```

Výpis zdrojového kódu 5.3 Demonstrátor - nastavení validačních parametrů modelu akce

Výstupem této validace je mapa, obsahující název proměnné a její chybovou zprávu. Tento výsledek si pouze doplníme o celkový stav validace a vrátíme ho pomocí mapy Ajax požadavku.

JavaScript dále požadavek předá metodě `doAjaxFormSuccess`, která přijatá data vypíše do připravených chybových HTML elementů. Například chybový element pro položku jméno:

```
<span id="nameError" class="errorMessage"></span>
```

Výpis zdrojového kódu 5.4 Demonstrátor - ukázka formulářového pole v JS

## 5.4. Uživatelé

Uživatelé tvoří základní část projektu vzhledem k tomu, že všechny ostatní části projektu s nimi pracují a jsou s nimi dále provázány. Prioritním prvkem celého modelu je `SpringSecurity`, přes který je vyřešena kompletní bezpečnost webu. Díky `SpringSecurity` je možné rozdělovat obsah projektu do kategorií dle práv uživatelů a dále celý projekt diverzifikovat: například zpřístupnit standardnímu uživateli jen část URL adres oproti administrátorovi a podobně. Díky provázanosti `SpringSecurity` je možné pracovat s uživateli v celém projektu a to jak v JSP stránkách, tak v CLASS souborech.

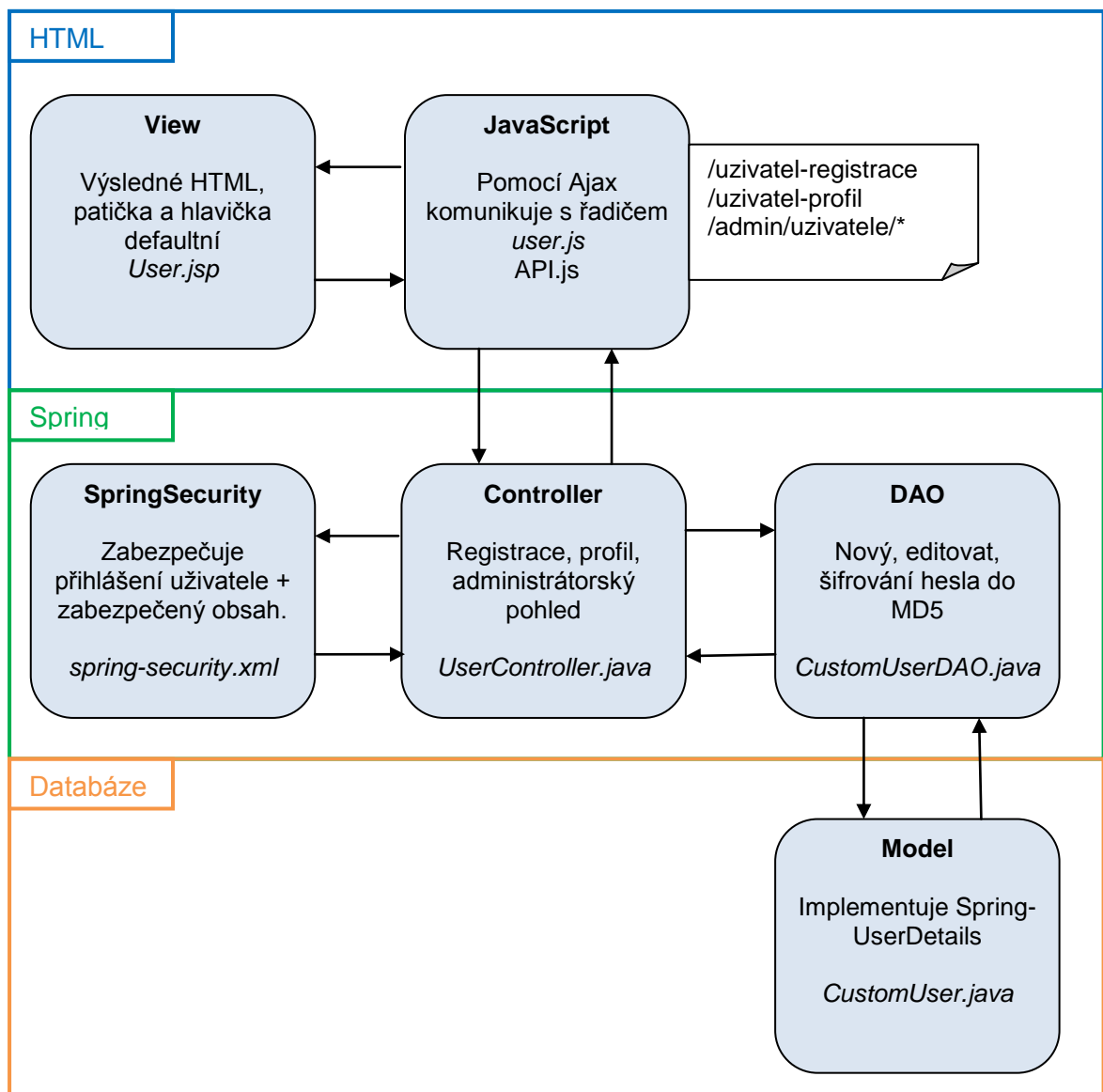


Schéma 5.2 Model postupu informací o uživateli

## 5.5. Soubory

Práce se soubory je v demonstrátoru vyřešena dostatečně obecně, aby bylo možné přidat načítání a stahování souborů bez větších změn do dalšího modulu. Díky tomu vznikl samostatný volně použitelný celek, který je nezávislý na modulu v demonstrátoru. Souborový modul tedy nepracuje v demonstrátoru samostatně, ale je vždy součástí většího celku: zpráv, článků, akcí, inzercí a stáje. Ke zmiňovaným modulům je tedy možné připojit libovolný počet souborů, který se již vnitřně člení na soubory a obrázky. Obrázky jsou zobrazované v hlavičce a v galerii a ostatní typy souborů jsou zobrazené v seznamu pro stažení.

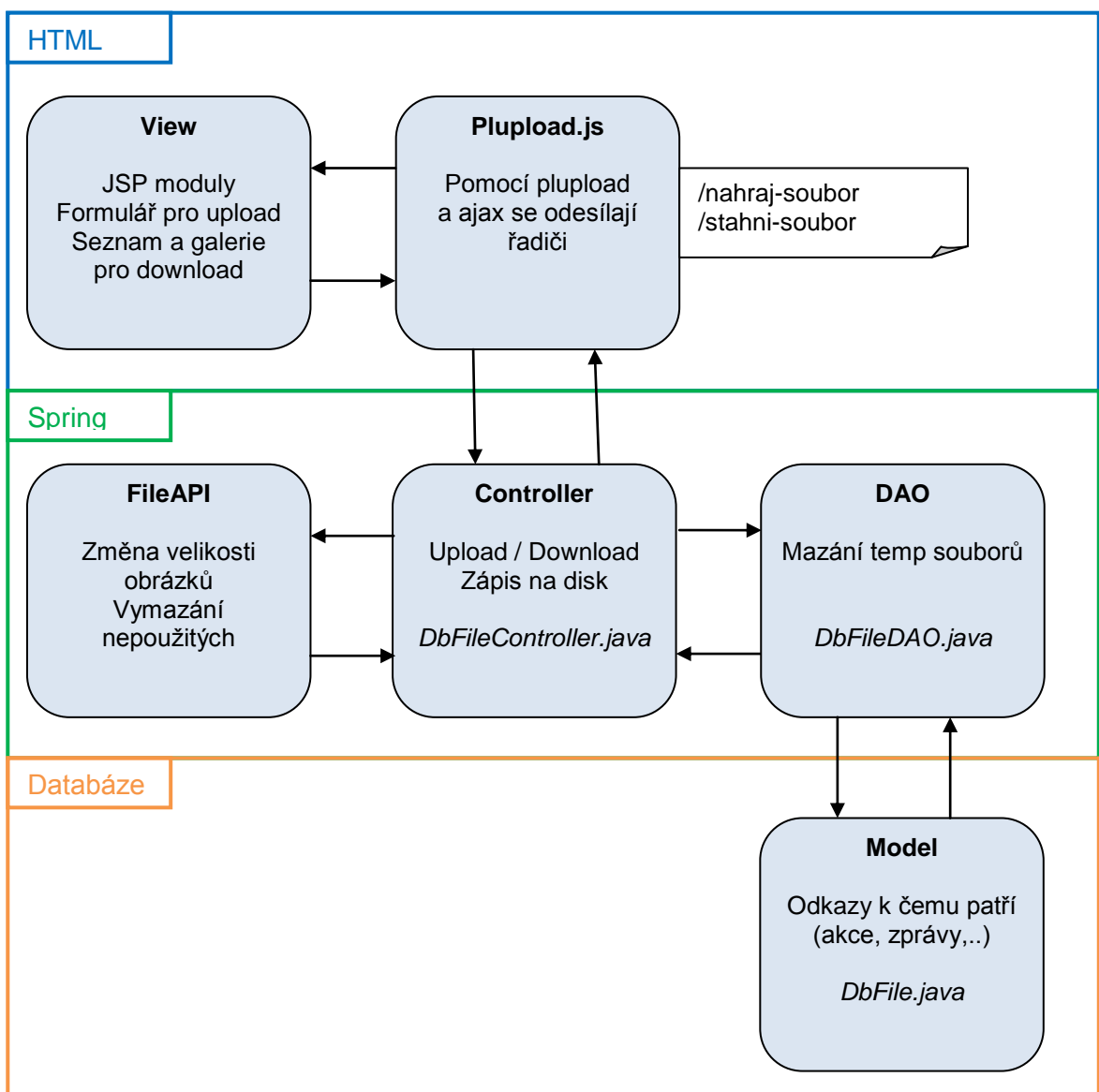


Schéma 5.3 Model postupu načítání / stahování souborů

## 5.6. Zprávy a články

Zprávy, potažmo články, tvoří samostatný menší celek – redakční systém. Cílem bylo umožnit uživatelům bezproblémový návrh stylizovaných článků a to včetně vkládání obrázků, bez nutnosti znalosti HTML kódu. Těchto možností jsme docílili hlavně díky jQuery a dalších kompatibilních knihoven – `wysiwyg`. Tato knihovna nám umožňuje do HTML stránky vložit textový editor, který umožňuje základní grafické úpravy textu.

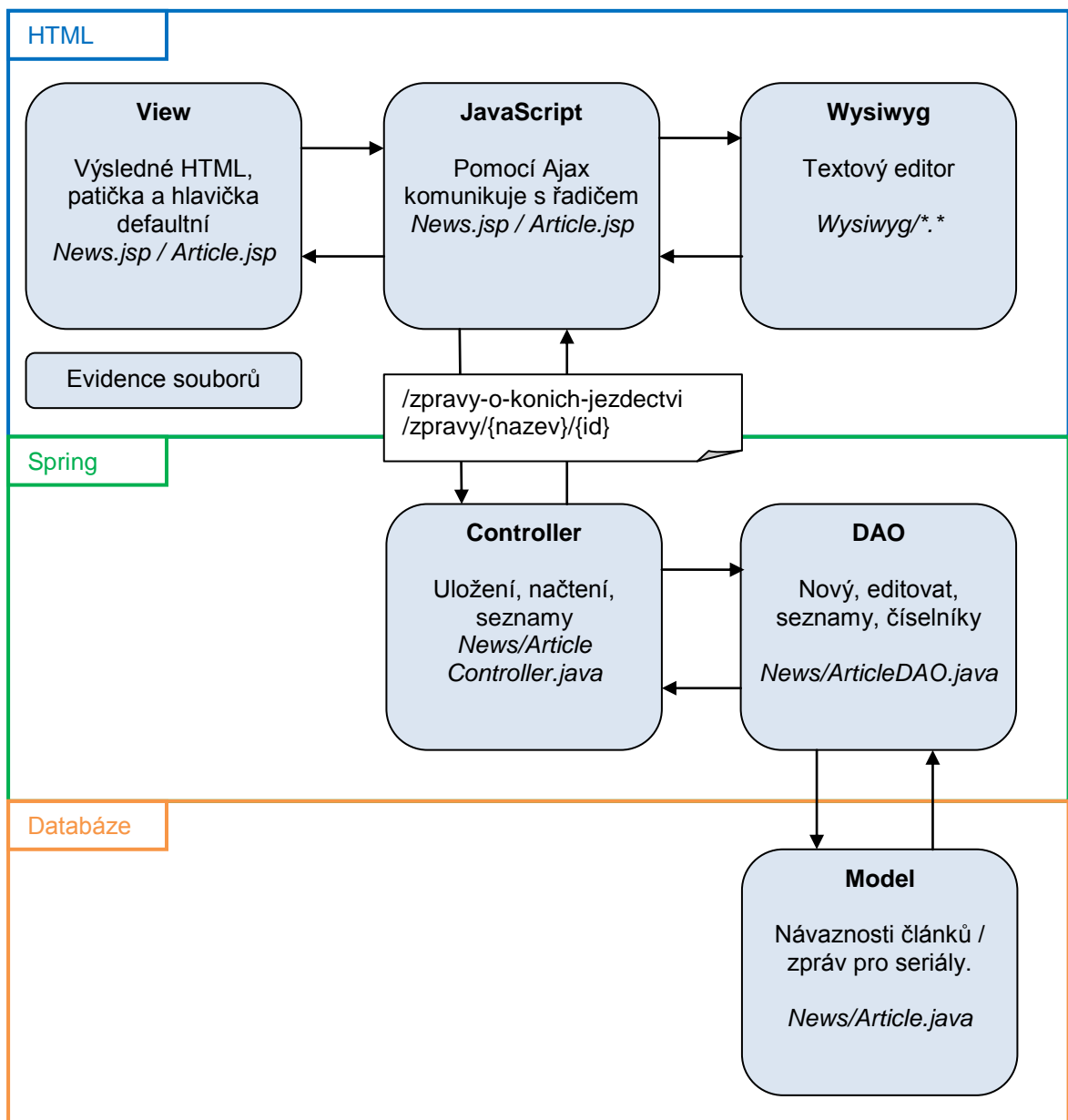


Schéma 5.4 Model postupu zpracování zpráv a článků

## 5.7. Akce

Sekce akce nám spojuje opět hned několik graficky zajímavých modulů v jeden funkční celek. Celý systém je založený na jednoduchém principu – uživatelé mohou zakládat / plánovat akce. Ostatní návštěvníci mohou jejich akce zobrazit a dále v nich vyhledávat. Díky integraci Google map API je možné při zakládání akce zvolit přesné souřadnice konané akce a tím je možné přesně určit místo konání akce i bez možnosti specifikovat toto místo adresou. Registrovaní uživatelé navíc mohou filtrovat akce podle skutečné vzdálenosti od jejich domovských souřadnic.

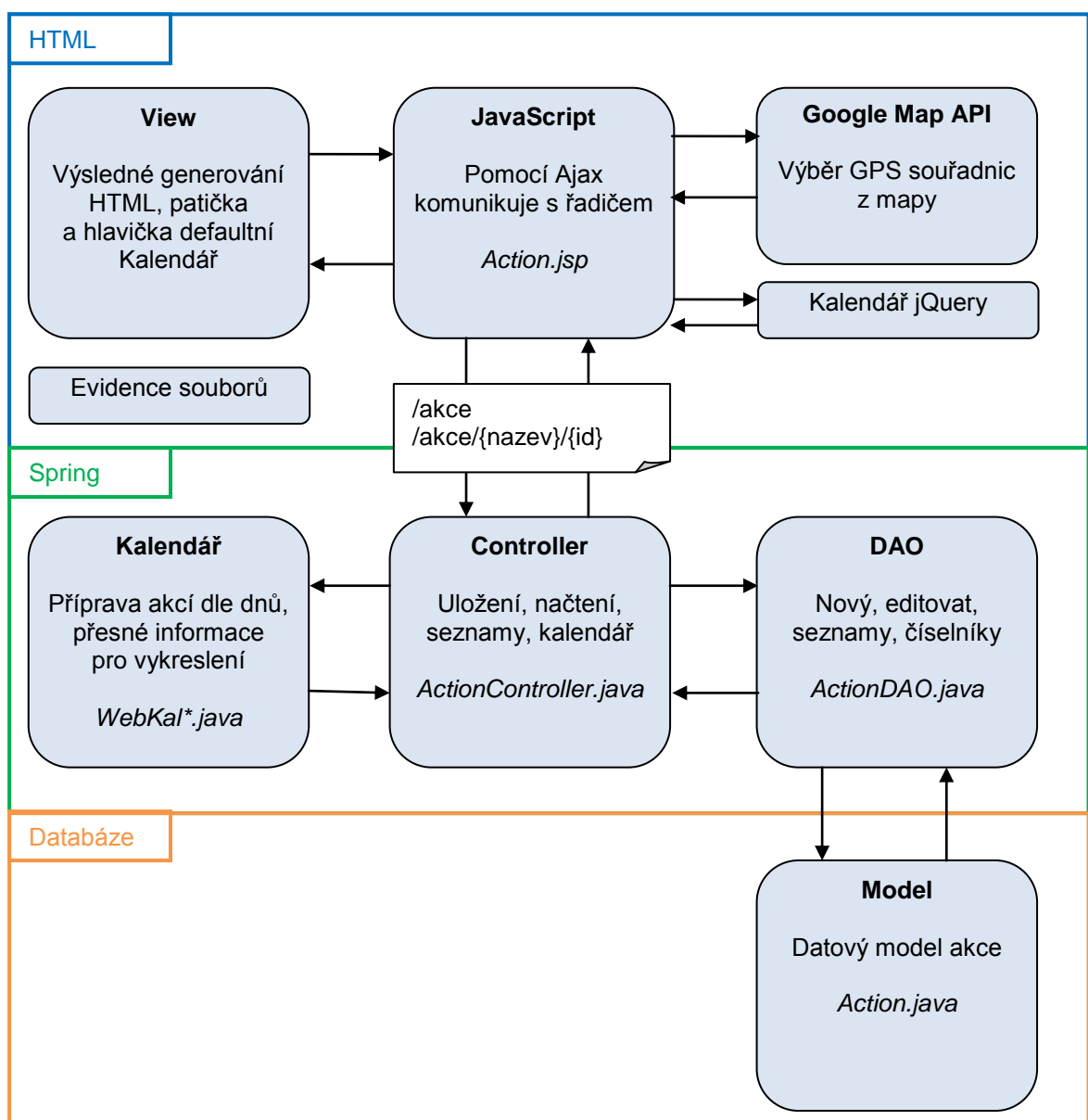


Schéma 5.5 Model postupu zpracování akcí



## 5.8. Inzerce

Inzerce opět propojuje již popisované moduly a k těmto modulům připojuje další svojí vlastní logiku. Pro evidenci inzerátů je tedy možné zadávat GPS souřadnice výběrem z mapy a připojovat libovolný počet obrázků. Novou a prozatím nepoužitou funkcionalitou je zde dynamické generování stránky při zadávání inzerátu. Díky tomuto rozšíření je možné mít pouze jednu JSP stránku. Dále je díky této změně možné zobrazovat uživateli vždy pouze ty formulářové pole, které jsou pro něj skutečně poplatné a tím se vyvarovat chybovosti a zbytečné nepřehlednosti. Celá koncepce inzerátů je časově omezena platností inzerátu. Je zde tedy zapracovaný jednoduchý systém na automatické vyřazování prošlých inzerátů.

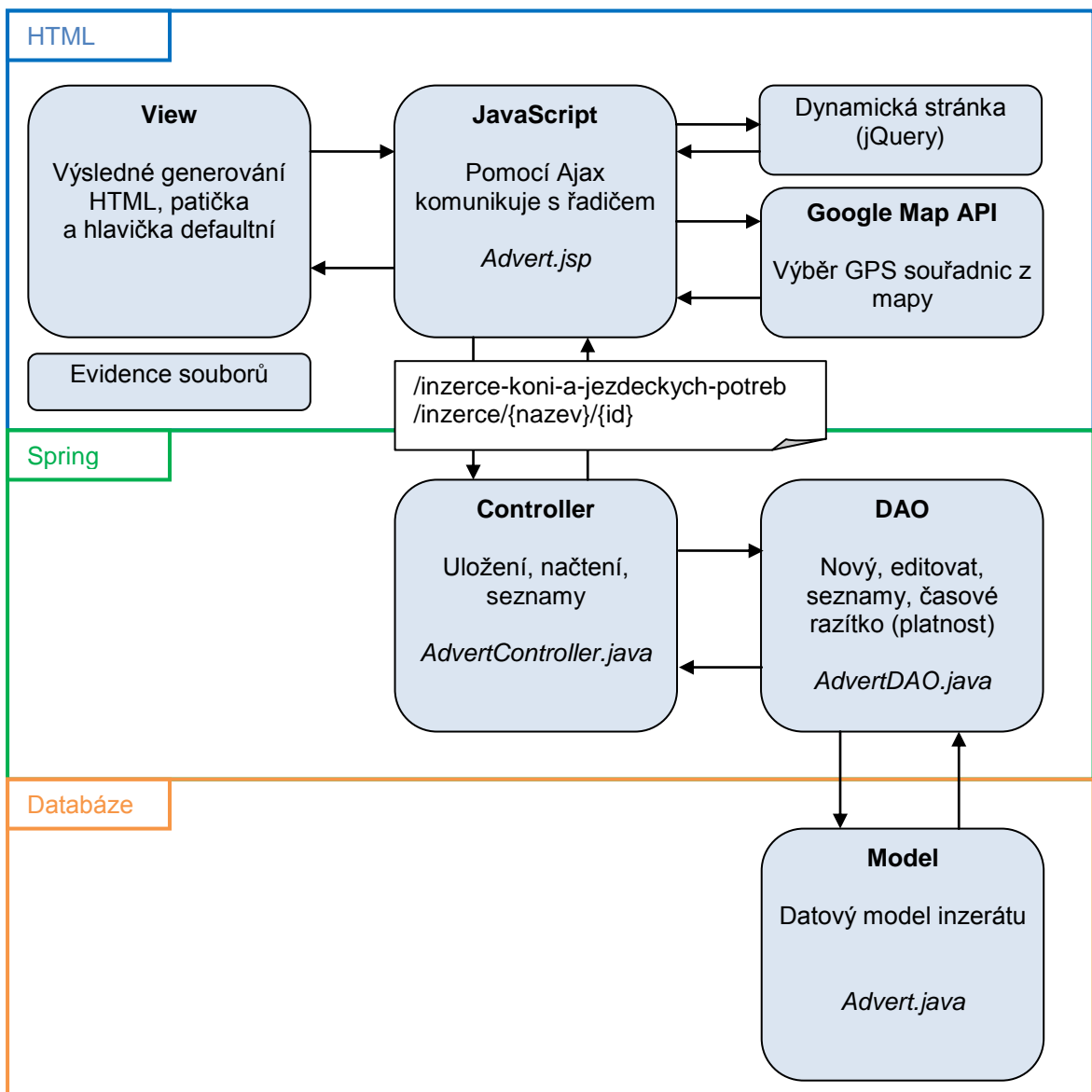


Schéma 5.6 Model postupu zpracování inzerátu

## 5.9. Stáj

Sekce stáj je opět dost specifická. Její základní funkcí je možnost založit si vlastního koně, k němuž je možné evidovat rodokmen provázaný s vlastními koňmi, či koňmi ostatních uživatelů. Rozšířenou funkcí evidence koní, je možnost evidence libovolného počtu zúčastněných se soutěží. Pro libovolný počet evidence soutěží je dynamicky rozšiřována zobrazená stránka o další objekty: soutěž. Tato funkcionalita je opět vyřešena pomocí jQuery. Nově je ovšem vyřešena předloha generovaného obsahu v JSP objektu pro soutěž, kde pomocí jQuery dochází pouze k duplikování skryté předlohy soutěže. Komplexním zhodnocení pořizovaných údajů je výpočet sportovního indexu – který je založený na počtu pořizovaných soutěží a stáří koně.

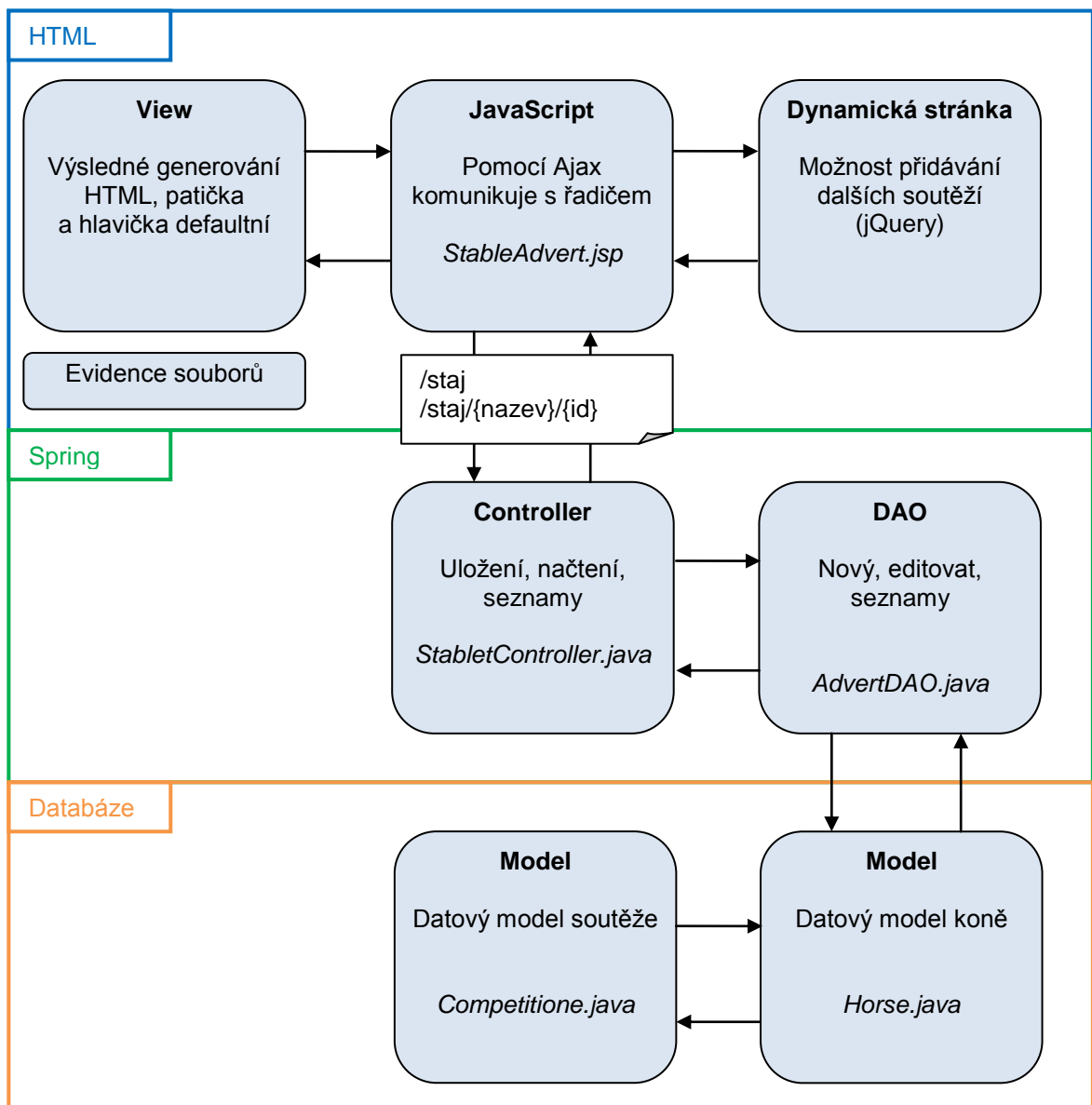


Schéma 5.7 Model postupu zpracování požadavků ve stáji

## 5.10. Hlavní stránka

Hlavní stránka je online generována z veškerého obsahu webu. Generováním hlavní stránky z nejaktuálnějšího obsahu webu je docílena neustálá aktuálnost a také atraktivnost hlavní stránky. Díky této koncepci nově přichodzí čtenáři mají možnost ihned přejít na nové příspěvky (zprávy, články, akce, inzeráty) bez nutnosti vyhledávání aktuálních informací. Koncepční návrh zobrazování zpráv a článků byl navržený dle nejnavštěvovanějších zpravodajských serverů<sup>1</sup>. Pro navržení formátu dalších zobrazovaných informací bylo postupováno dle analýzy konkurenčních webů a jejich přístupu k zobrazování informací. Dle dosavadních zkušeností je vhodné částečně těžit ze zažitých zvyklostí návštěvníků, čímž se neuživatelé na stránce lépe orientují hned při první návštěvě.

Pro větší podporu vkládání kvalitních fotografií k pořizovaným informacím, je do hlavní stránky také zapracován modul na zobrazení fotky týdne. Fotku týdne vždy nastaví administrátor webu. Tím je fotka zobrazena na hlavní stránce v levé horní části i s odkazem na informaci kde byla zveřejněna. Dále se tato fotka zobrazí na pozadí celého webu, čímž celý web získá nádech této fotografie. Jediný aktuálně nevyřešený problém je ten, že pokud se jedná o zajímavou fotografii, ovšem její kraje obsahují více rušivých elementů, není možné takovouto fotografií zobrazit i přes to, že by z hlediska zachycení hlavní informace či středového detailu byla v daném týdnu nejlepší.

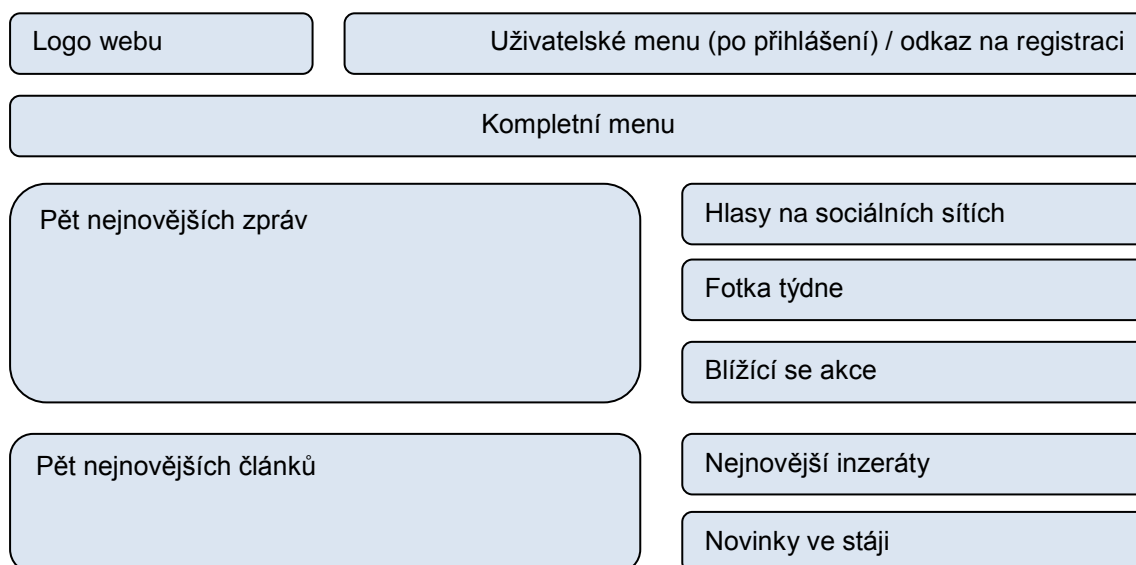


Schéma 5.8 Návrh rozložení hlavní stránky

<sup>1</sup> <http://www.novinky.cz/>, <http://www.idnes.cz/>

## 5.11. Sitemap.xml

Od dnešních webových projektů se očekává plná implementace pro indexování vyhledávačů. Demonstrátor tedy automaticky generuje `sitemap.xml` na adrese `/sitemap.xml`. Díky tomu je možné v Google Webmaster sledovat, zda Googlebot indexuje nové stránky pro plnohodnotný a aktuální obsah ve vyhledávání. Společnost Seznam tento přehled nenabízí. Toto dynamické generování zaručí stále aktuální `sitemap.xml` bez nutnosti jakéhokoliv ručního zásahu nebo dokonce opakovaného ručního generování. Tento modul je možné opět přesunout do libovolného projektu, kde dojde pouze ke změně DAO, potažmo modelových tříd.

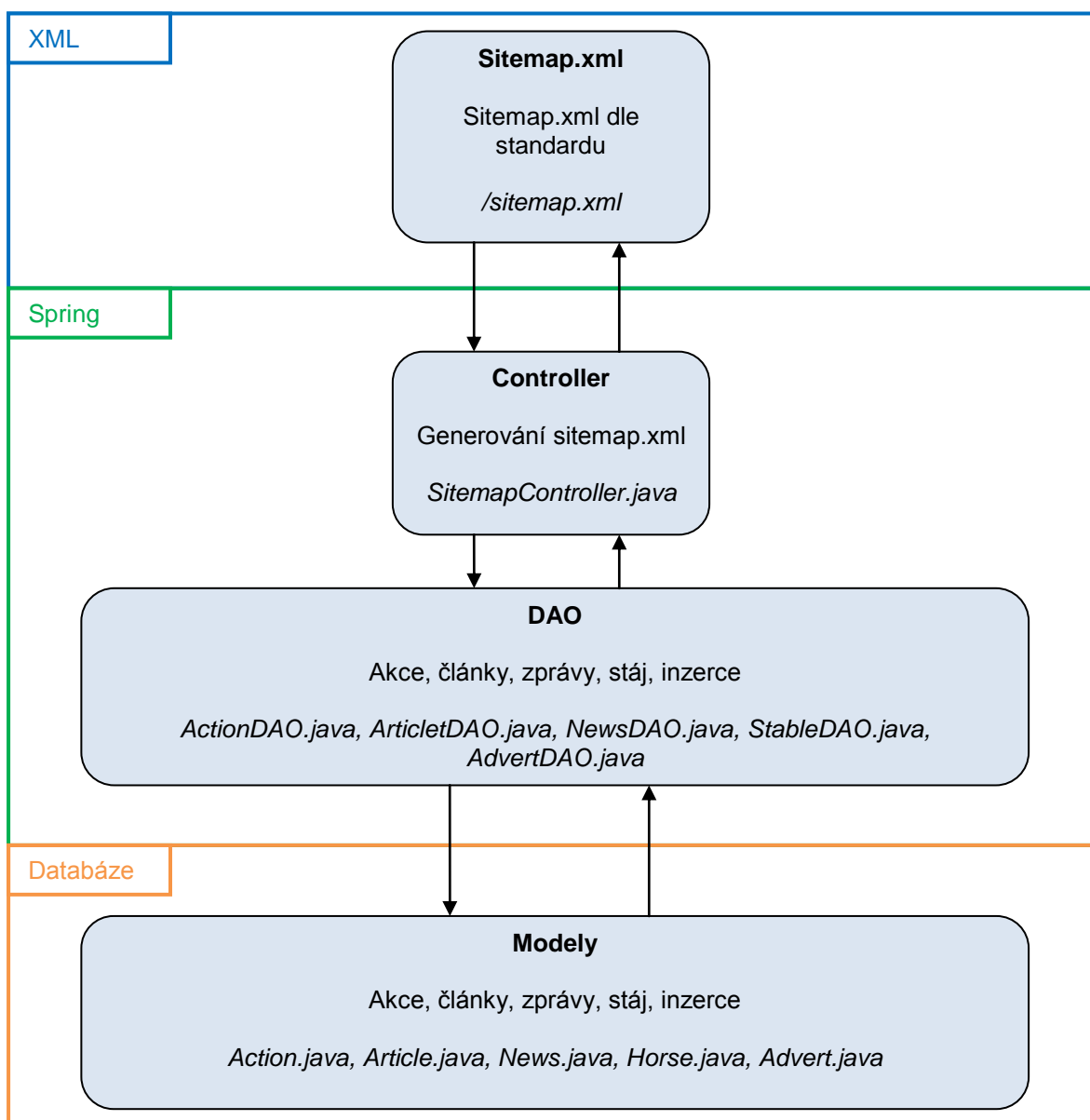


Schéma 5.9 Model postupu zpracování požadavku pro Sitemap.xml

## 5.12. Zprovoznění demonstrátoru

Pro správné zprovoznění demonstrátoru je potřeba:

- Spring Tool Suite verze 3.1. a vyšší
- MySql 5.1 a vyšší
- Tomcat v 7.0 a vyšší (nepovinně)

### Spring tool suite

Spring tool suite<sup>1</sup> již v základu obsahuje podporu Maven, všechny potřebné knihovny se tedy automaticky stáhnou po importu projektu do studia. Ostatní potřebné zdrojové kódy jsou přiložené v projektu.

### MySql

Teoreticky je možné projekt zprovoznit i na jiné databázi, ovšem vývoj i ostrý provoz probíhal na databázi MySql ve verzích 5.0 až 5.1. Nastavení připojení k databázi se nachází v souboru:

```
... \src \main \webapp \WEB_INF \config \properties \jdbc.properties
```

```
/* Nastavení ovladadače pro připojení k databázi */  
jdbc.driverClassName=com.mysql.jdbc.Driver  
/* Nastavení dialektu - dle typu databáze */  
jdbc.dialect=org.hibernate.dialect.MySQL5Dialect  
/* Nastavení adresy databáze */  
jdbc.databaseurl=jdbc:mysql://localhost:3306/zesedla  
/* Přihlašovací údaje k databázi */  
jdbc.username=root  
jdbc.password=admin
```

Výpis zdrojového kódu 5.5 Demonstrátor - definice připojení k databázi

### Kontejner

Projekt je možné spustit na standardním kontejneru studia od VMware. Ovšem kompletní vývoj a reálný provoz je otestovaný na kontejneru Tomcat od firmy Apache verze 6.0 a 7.0

---

<sup>1</sup> <http://www.springsource.org/sts>

## 6. Reálný provoz

Tato kapitola se bude zabývat reálnými tématy, které bylo možné vypracovat pouze díky tomu, že finální projekt – zpravodajský server o jezdeckví - byl reálně nasazen a cca ½ roku běží v ostrém provozu. Projekt demonstrátor uvedený výše z tohoto projektu striktně vychází a je pouze částečně upraven a to pro větší použitelnost ve výuce. Základní části, postupy zpracování a obecně celá konfigurace zůstala zcela totožná. Do budoucna je možné, že se tyto projekty více oddělí a to za předpokladu, že bude probíhat další vývoj jezdeckého serveru zeSedla.cz

Pro plnohodnotný běh projektu bylo nutné vynaložit také další, jak finanční, tak pracovní úsilí pro implementaci všech nástrojů pro analýzu provozovaného projektu. Pro sledování stavu indexování stránek a reakce na chybové hlášení (Google Webmaster) a dále pro podrobné sledování návštěvnosti (Google Analyst). V neposlední řadě také skutečná možnost placené reklamy pomocí Google AdWords.

### 6.1. Hosting

Hosting neboli webhosting je pronájem virtuálního prostoru pro webové stránky. Díky hostingu je možné své webové stránky umístit na servery s trvalým a stabilním připojením k internetu. Ceny webhostingu se pohybují od řádu korun pro tisíce korun za měsíc. K dispozici bývá i bezplatná varianta, jedná se ale pouze o základní typ služby. Typů rozdělení hostingů je celá řada, jedny z nejdůležitějších jsou:

- Podle typu podporované technologie (HTML, PHP, ASP, JSP)
- Podle poskytované databáze (MySQL, PostgreSQL, MS SQL)

Dle aktuálně nabízených služeb je pravidlem, že jsou k hostingu dodávány i ostatní služby, jako například: emailové schránky, FTP připojení, statistika návštěvnosti a dalších parametrů, administrační nástroje a mnohé další. Samozřejmostí již je technická podpora ať k zprovoznění či nastavení hostingu, tak i k základnímu nastavení ostatních služeb. Technická podpora v rámci hostingu ovšem končí při předání rady či informace, samotné nastavení služeb již tato základní podpora neprovádí.

Hlavní nevýhodou hromadných hostingů je to, že jeden fyzický HW server je sdílen pro stovky ne-li tisíce webových projektů. Z tohoto důvodu je možné, že jeden projekt může negativně ovlivnit ostatní. Prozatím se ale takovéto situace stávají spíše výjimečně.

### **6.1.1. Výběr hostingů**

Výběr hostingů pro J2EE platformy není zcela jednoznačný jako například výběr hostingů pro webové projekty napsané v PHP. Ceny PHP hostingů se pohybují v řádu 0–100 Kč, ovšem ceny hostingů pro J2EE projekty jsou většinou o řád výš. Vybrat tedy vhodný a únosně drahý hosting pro J2EE platformu obnáší delší analýzu trhu.

Po komplexní analýze můžeme rozdělit typy hostingů podle dvou základních kritérií:

- Hostiny hostované v ČR / Zahraničí
- Kompletní instalace a administrace, či pouze administrace SW řešení

#### **ČR / Zahraničí**

Hostiny v ČR mají výhodu v kompletní české podpoře. Ovšem ceny jsou daleko vyšší než u zahraničních hostingů. Jedním z bodů pro rozhodování by mohlo být i tvrzení některých uživatelů, že weby hostované v zahraničí jsou pro ČR pomalejší – dle technické analýzy a aktuálního stavu internetových připojení, by tato informace měla tvořit zanedbatelné hodnoty.

Hostiny v zahraničí mají výhodu větší profesionality – jejich zkušenosti v této oblasti služeb jsou na vyšší úrovni. Tuto výhodu bych přisuzoval větší a delší zkušenosti s provozem J2EE hostingů. Jak již bylo zmíněno výše, částečnou nevýhodou je většinou kompletní komunikace anglickém jazyce.

## **6.2. Zprovoznění hostingů**

Tato část práce bude popisovat jednotlivé kroky, které jsou stěžejní při zprovoznění J2EE webového serveru na dvou dostupných hostingech. Dle obecné analýzy dostupných možností / hostingů se jedná o dva typy z hlavního rozdělení: virtuální server a hosting.

### 6.2.1. Virtuální server

Pokud bychom se rozhodli pro vlastní řešení, například virtuální server, můžeme díky tomu získat plnou kontrolu nad projektem. Ovšem jednou poměrně zásadní nevýhodou je kompletní konfigurace celého serveru. Po objednání virtuálního serveru je virtuální server zpřístupněn v základní podobě pouze s operačním systémem dle výběru (Linux / Windows). Veškeré další serverové programy je nutné doinstalovat. Níže je uveden seznam nejdůležitějších kroků pro zprovoznění J2EE hostingu na základní verzi virtuálního serveru. Níže uvedené body popisují standardní instalaci serveru s doménou včetně emailového a FTP serveru.

- Přesměrování domény na vlastní virtuální server
- Instalace kompletního emailového klienta, včetně kompletního nastavení, přes spamové filtry až po oprávnění připojení
- Instalace Java
- Instalace a konfigurace kontejneru (pro tuto práci Tomcat 7.0)
- Instalace FTP serveru
- Instalace databáze MySql + případný vzdálený přístup (php admin)

Veškeré tyto instalační kroky mě jako začátečníkovi v této problematice zabrali zhruba 12 hodin čistého času. Ovšem poměrně velkou část mi také zabralo samotné vyhledání kvalitních freeware řešení. Pokud bych tuto instalaci prováděl po druhé, čas instalace a konfigurace bych odhadoval zhruba na 4 hodiny.

### 6.2.2. J2EE Hosting

Pokud bychom se rozhodli pro hosting, veškeré další nedílné součásti funkčního serveru řeší hosting sám. Na rozdíl tedy od instalace (virtuální server) je nutné provést pouze konfiguraci všech potřebných programů. Konfigurace emailových účtu probíhá v grafickém rozhraní, a pokud se čtenář v problematice alespoň částečně orientuje, nenarazí na žádný zásadní problém. Stejným způsobem probíhá i konfigurace FTP serveru, kde je pouze nutné vytvořit uživatelské účty.



Druhá část konfigurace hostingu je zaměřena na samotné nastavení JVM<sup>1</sup> a samotné nahrání WAR souboru s aplikací. Defaultní nastavení JVM není potřeba ve většině případů nijak modifikovat. Druhý krok – nahrání aplikace probíhá pomocí standardní HTTP komunikace a defaultně po jeho načtení dojde automaticky k restartu JVM. Čas potřebný pro konfiguraci bych určil zhruba na 2 až 4 hodiny čistého času.

Při prvním nasazení tohoto projektu na J2EE hostingu se objevily doposud neznámé problémy i přes to, že na virtuálním serveru projekt fungoval bez chyb. Po podrobné analýze chybových výpisů bylo zjištěno, že projektu v určitých různorodých časových úsecích dojde paměť a tím dojde k destabilizaci celého JVM. Jak bude níže uvedeno, problém byl způsobem omezeným prostorem Java Heap Size<sup>2</sup>, který činí 64 MB. Po další podrobné analýze se problém vyskytoval při práci se soubory – konkrétně při grafické úpravě obrázků. Z důvodu úspory komunikační linky dochází automaticky v Java ke zmenšování rozlišení úvodních obrázků dle HTTP požadavku. Pokud ovšem těchto požadavků přišlo například od internetových robotů více najednou, došlo k vyčerpání 64 MB paměti práce zpracovávanými obrázky a tím ve většině k znepřístupnění JVM, tím pádem i celého projektu. K částečnému řešení tohoto problému byla provedena optimalizace kódu pro zmenšení obrázků: striktní použití příkazů na vymazání proměnných a optimalizace počtu proměnných. Tato úprava chybu téměř odstranila. Chyba se ovšem ještě jednou vyskytla během psaní této práce, nebyl již ale čas na hlubší prozkoumání příčin této chyby.

---

<sup>1</sup> Java Virtual Machine

<sup>2</sup> v podstatě se jedná o paměť RAM pro JVM

### 6.3. Vynaložené finanční náklady na provoz projektu

Zde jsou uvedeny reálné náklady zahrnující veškeré náklady, které byly na projekt vynaloženy v rámci jeho testování, zkušebního a i ostrého provozu. Cílem těchto informací je shrnout veškeré vynaložené výdaje poplatné k datu realizace této diplomové práce. Finance se mohou aktuálně již lišit. Testovací provoz virtuálního serveru probíhal již z počátku této práce, aby bylo možné otestovat možnosti na v té době nejdostupnějším (nejlevnějším) a pro tento projekt dostatečně výkonným hostingem. V tabulkách se nacházejí určité řádky duplicitně z toho důvodu, aby bylo možné pracovat s každou tabulkou samostatně, čímž je možné vycházet z těchto nákladů pro provoz vlastního projektu už s konkrétním výběrem (virtuální server či hosting).

#### Virtuální server

Služba	Cena (za rok / týden)
Vývoj projektu (vlastní vývoj 200 hodin* 150 Kč/hod)	30 000 Kč
Virtuální server (za rok)	2 400 Kč
Doména (za rok)	250 Kč
Reklama (pro téma jezdeckví s limitem cca 100 Kč na den – zhruba 4-5 kliknutí na reklamu) – cena za měsíc	3 100 Kč
<b>Pořizovací náklady s ročním provozem</b>	<b>32 650 Kč</b>
<b>Cyklické roční náklady</b>	<b>2 650 Kč</b>

Tabulka 6.1 Vynaložené finanční náklady pro virtuální server

#### J2EE Hosting

Služba	Cena (za rok / týden)
Vývoj projektu (vlastní vývoj 200 hodin * 150 Kč/hod)	30 000 Kč
Hosting (za rok)	2 000 Kč
Doména (za rok)	250 Kč
Reklama (pro téma jezdeckví s limitem cca 100 Kč na den – zhruba 4-5 kliknutí na reklamu) – cena za měsíc	3 100 Kč
<b>Pořizovací náklady s ročním provozem</b>	<b>32 250 Kč</b>
<b>Cyklické roční náklady</b>	<b>2 250 Kč</b>

Tabulka 6.2 Vynaložené finanční náklady pro J2EE hosting

## 6.4. Výkon projektu

Pro sledování výkonnosti projektu jsme použili pokročilý nástroj Google Analyst. Díky tomuto nástroji můžeme pracovat jak s aktuálními daty, tak i s daty z minulých období. Samozřejmostí jsou exporty do XLS, PDF a dalších.

### HW specifikace

Wedos virtuální server <sup>1</sup>	Mochahost J2EE Hosting <sup>2</sup>
<b>HDD</b> 30 GB	<b>HDD</b> Neomezeno
<b>Garantovaná RAM DDR3</b> 1 GB	<b>Java Heap Size</b> 64 MB
<b>System</b> MS Win Server 2008 R2 Datacenter CZ 64bit	<b>System</b> Linux

Tabulka 6.3 Porovnání HW komponent virtuálního serveru vs J2EE hostingu

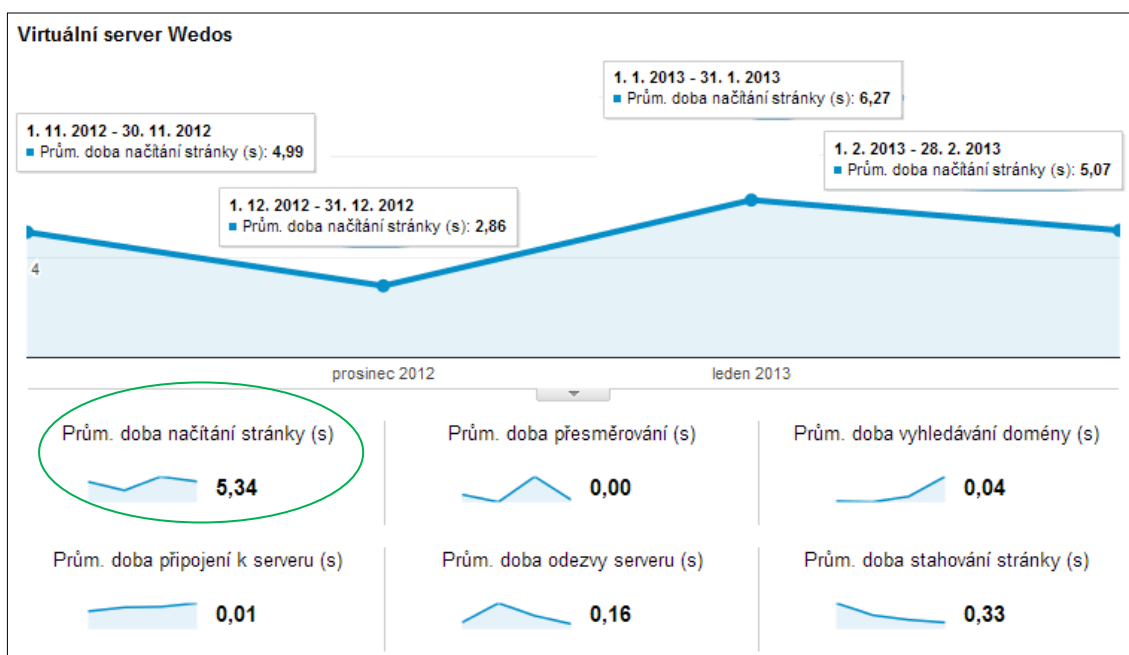
Technická specifikace by se mohla dále rozlišovat na použité verze databáze, kontejneru a dalších potřebných SW. Cílem této kapitoly není detailní technický rozbor nabízeného řešení, ale porovnání výkonů aktuálně nabízených dostupných hostingových služeb, ke kterým jsou výše vyčísleny finanční náklady a níže celkové porovnání.

Pro porovnání výkonnosti projektu byly použity výstupy z nástroje pro analýzu webových projektů Google Analyst. Na níže uvedených grafech jsou porovnávány 4 měsíce provozu na obou serverech. Delší časové období by nám mohlo přinést větší objektivnost posouzení rychlosti, ovšem z důvodu časových limitů na tuto práci není možné porovnávat delší časové období. Z níže uvedených obrázků je zřejmé, že Wedos virtuální server vyřizoval požadavky rychleji o celé 2,81 s. Dle technického rozboru můžeme tento zvýšený výkon přisuzovat větší paměti RAM, která i přes to, že na virtuálním serveru běžel Windows Server 2008, stále garantoval více prostoru pro JVM. Je tedy možné předpokládat, že i při více požadavcích reagoval rychleji.

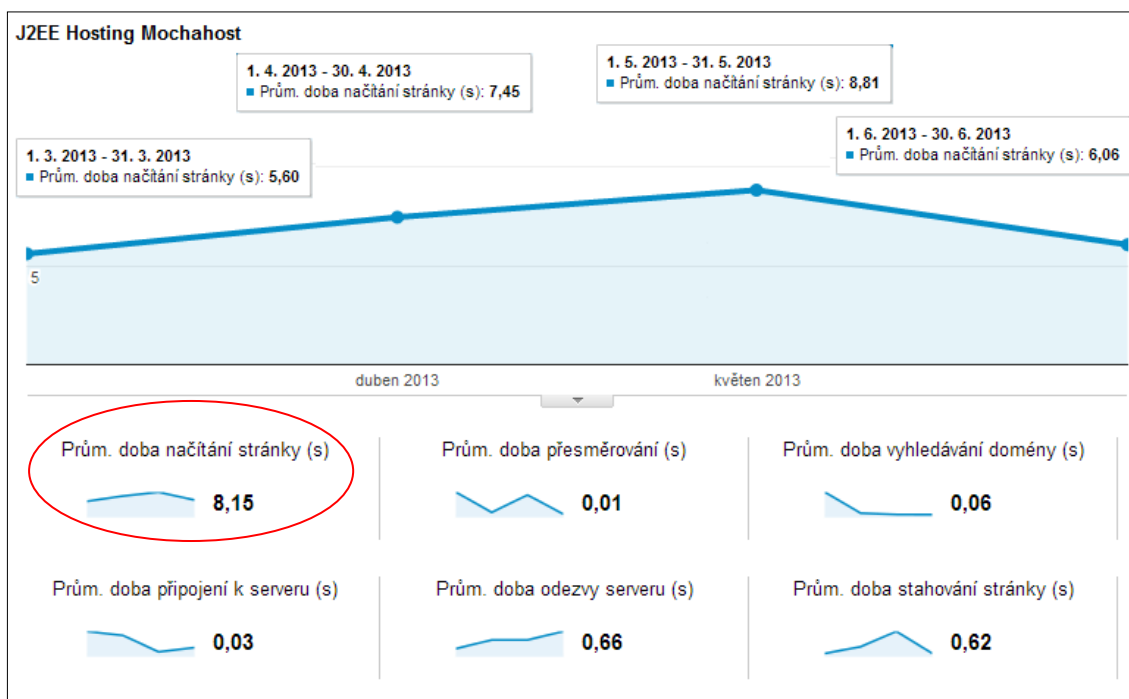
Závěrem je možné specifikovat, že oba dostupné servery splňují přijatelnou výkonnost. Dle dosavadního průzkumu uživatelé neregistrují žádný pokles výkonnosti, pokud se průměrná doba načítání prodloužila o zmiňovaných 2,81 s. Z hlediska výkonu je tedy možné obě řešení doporučit.

<sup>1</sup> <http://hosting.wedos.com/cs/virtualni-servery.html>

<sup>2</sup> <http://www.mochahost.com/java.php>



Obrázek 6.1 Přehled rychlosti webu za provozu na Wedos virtuálním serveru



Obrázek 6.2 Přehled rychlosti webu za provozu na Mochahost J2EE hostingu

## 6.5. Komplexní porovnání dvou reálně nasazených hostingů

Za dobu provozu reálného projektu jsme v první části používali virtuální server Wedos. Pro druhou část reálného provozu jsme si vybrali J2EE hosting od Mochahost.com. Provoz na virtuálním serveru trval zhruba rok, je ovšem nutné uvést, že z počátku se spíše jednalo o pokusný projekt, který zatím nebyl zcela provozuschopný a provoz virtuálního serveru sloužil spíše pro testovací účely. Z počátku vývoje projektu bylo nutné testovat jednotlivé části na virtuálním serveru, aby se předešlo vývoji kompletního projektu, který by poté nebylo možné reálně nasadit.

Hlavní výhodou i nevýhodou Wedos virtuálního serveru je kompletní administrace celého virtuálního serveru. Pokud bychom tedy zřizovali první projekt, je nutné věnovat zmíněných 12 hodin času ke konfiguraci plnohodnotných serverových programů. Rozdíl ceny je při porovnání s druhým hostingem zanedbatelný, při provozu virtuálního serveru se systémem Linux se jedná o roční rozdíl 400 Kč, při používání Windows se roční cena liší o 1600 Kč. Aktuálně je ovšem k dispozici licence Windows na první rok zdarma, je tedy možné pro roční testování zvolit i systém Windows za licenci systému Linux. Hlavní výhodou tohoto typu hostingu je možné určit jako vysoký potenciál budoucího rozšíření. Za předpokladu provozu více webových serverů, je možné stále pracovat pouze se základní konfigurací a tím výrazně ušetřit. Samozřejmostí virtuálního serveru je také možnost přikupovat další výkonnostní moduly, které nám bez nutnosti jakéhokoliv zásahu zvýší výkon celého systému.

Nasazení projektu na Mochahost J2EE hosting provázelo pár počátečních problémů, které byly popsány v kapitole 6.2.2. Samotná instalace a první spuštění bylo ovšem v porovnání s virtuálním serverem o mnoho snazší, nebylo nutné instalovat veškeré serverové programy, ale pouze nadefinovat uživatelské účty (email, FTP). Při porovnávání je možné specifikovat další náhradní řešení, která by vyřadila potřebnou optimalizaci části kódu na zmenšování obrázků:

- Vracet obrázky bez programového zmenšování
- Rozšířit Java Heap Space na 128 MB – 2800 Kč / Rok
- Rozšířit Java Heap Space na 256 MB – 3750 Kč / Rok

Výše uvedené varianty by poté jasně ukázali na tento typ hostingu, převládala by zde pouze kladná složka recenze, nebylo by nutné provést instalaci serverových programů.

Níže uvedená tabulka shrnuje všechny dosavadní parametry a další reálné zkušenosti z provozu. Oba typy testovaných hostingů splňují potřebnou funkcionalitu a není možné jeden z nich jednoznačně doporučit.

Wedos.cz Virtuální server	Mochahost.com J2EE Hosting
- Instalace veškerého SW	+ Pouze konfigurace SW
+ Průměrné načítání stránek o 2,8s rychlejší viz obrázek 7.1 a 7.2	- Průměrné načítání stránek o 2,8s pomalejší viz obrázek 7.1 a 7.2
+ Česká podpora	- Anglická podpora
+ Neomezeno Java Heap Size	- Omezeno Java Heap Size 64 MB
- Cena první rok 2400 Kč, další 3600 Kč viz kapitola 7.3	+ Cena 2000 Kč viz kapitola 6.3

Tabulka 6.4 Porovnání použitých hostingů

## 6.6. Google Analyst

Google Analyst je profesionální webový nástroj na kompletní sledování statistik webového projektu. Díky tomuto nástroji máte možnost sledovat návštěvnost do posledního detailu. Můžeme sledovat pohyb vašich uživatelů po vašem webu v grafickém náhledu a mnoho dalších funkcí. Jako poslední rozšíření byla do Google Analyst přidána možnost sledování návštěvníků v reálném čase, která vám přinese plnohodnotný přehled, kolik uživatelů je právě připojeno na váš web a kde se nacházejí.

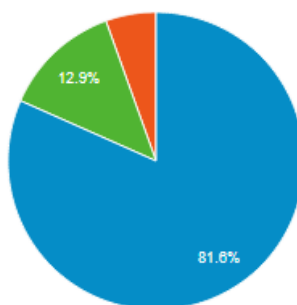
### 6.6.1. Zdroje návštěvnosti

Jedna z hlavních částí Google Analyst je část pro sledování návštěvnosti. Zdroje návštěvnosti se dále rozdělují na 4 hlavní části: provoz z vyhledávání, návštěvnost z odkazujících stránek, přímá návštěvnost a návštěvnost z placené reklamy (kampaně).

Na hlavní stránce této sekce můžeme pracovat s celkovými čísly (viz obrázek 7.3). Dále zde nalezneme celkový graf časového vývoje návštěvnosti. Tento přehled si můžeme zobrazit ve 4 hlavních rozlišeních: Každou hodinu, Den, Týden a Měsíc. Dále si pro celkový přehled můžeme zadávat časové období.

Tento web navštívilo 1 382 lidí.

■ Provoz z vyhledávání ■ Návštěvnost z odkazujících stránek ■ Přímá návštěvnost ■ Kampaně



Obrázek 6.3 Zdroje návštěvnosti

### Provoz z vyhledávání

Tato sekce má dvě hlavní rozdělení: zdroje a klíčová slova. V tabulce 6.5 je zobrazen přehled za zdroje. Ze zobrazených výsledků je zřejmé, že statistiky není nutné optimalizovat projekt pro jiné vyhledávače, než jsou Seznam.cz a Google.cz. Ostatní vyhledávače v celkovém počtu tvoří zanedbatelnou část.

Zdroj	Návštěvy	Návštěvy v %
1. seznam	614	54,43 %
2. google	499	44,24 %
3. search.centrum.cz	7	0,62 %
4. bing	3	0,27 %
5. conduit	2	0,18 %
6. ask	1	0,09 %
7. avg	1	0,09 %
8. images.google	1	0,09 %

Tabulka 6.5 Provoz z vyhledávání

### Návštěvnost z odkazujících stránek











V této sekci najdeme přehled všech stránek, které na náš web odkazují a odkud k nám přicházejí návštěvníci. Pokud se opět na tabulku 6.6 zaměříme podrobněji, je vidět, že se skutečně vyplatí publikovat obsah na sociálních sítích – převážně na Facebook. Návštěvnost z Facebook v tabulce zaujímá první místo. Návod, jak se sociálními miniaplikacemi pracovat jsme si ukázali v kapitole 4.7.

Zdroj	Návštěvy	Návštěvy v %
1. <a href="#">facebook.com</a>	137	76,97 %
2. <a href="#">miniaplikace.blueboard.cz</a>	11	6,18 %
3. <a href="#">m.search.seznam.cz</a>	9	5,06 %
4. <a href="#">google.cz</a>	8	4,49 %
5. <a href="#">plus.url.google.com</a>	4	2,25 %
6. <a href="#">m.facebook.com</a>	2	1,12 %
7. <a href="#">delta-search.com</a>	1	0,56 %
8. <a href="#">howrse.cz</a>	1	0,56 %
9. <a href="#">images.google.cz</a>	1	0,56 %
10. <a href="#">jutyna.own.cz</a>	1	0,56 %

Tabulka 6.6 Návštěvnost z odkazujících stránek

### Přímá návštěvnost

Přímá návštěvnost nám zobrazuje, jaké stránky uživatelé navštěvují jako vstupní stránku webového projektu. Z tohoto přehledu je možné dále získat informaci, které sekce webu jsou uživateli nejvíce oblíbené, případně jaké stránky si nechávají otevřené nebo jaké stránky si ukládají do záložek. Díky těmto informacím je možné specifikovat nejnavštěvovanější část webu a například, tuto část webu použít pro získání dalších pravidelných návštěvníků. Nebo také naopak, pokusit se odhalit nejméně navštěvované části webu a případně zvážit jejich další vývoj a provoz.

Vstupní stránka	Návštěvy	Návštěvy v %
1. / 	29	38,16 %
2. /akce 	9	11,84 %
3. /clanky/vycvik-kone-zakladni-prace-ze-zeme/17 	5	6,58 %
4. /akce/jarni-zavody-nemochovice-27-04-2013/575 	2	2,63 %
5. /akce/jezdecke-hobby-zavody-ve-vyskove-21-04-2013/579 	2	2,63 %
6. /clanky/vycvik-kone-par-veci-ktere-byste-meli-pred-zacatkem-vycviku-vedet/23 	2	2,63 %
7. /zpravy/tradicni-memorial-jana-papouska-vitezstvi-pro-jirho-luzu-s-land-rebelem-s/237 	2	2,63 %
8. /zpravy/vitez-ankety-nejkrasnejsi-kun-cr-2012-westernova-plemena-je-paint-horse/325 	2	2,63 %
9. /akce/13-rocnik-memorial-ing-karla-podarila-20-04-2013/588 	1	1,32 %
10. /akce/5-jezdecke-hry-v-arealu-pily-hrdejovice-19-05-2012/171 	1	1,32 %

Tabulka 6.7 Přímá návštěvnost



## Kampaně (placené reklamy)

Placené reklamy jsou vytvářené z Google AdWords. V Google AdWords si můžete vytvořit „kampaň“ – reklamy, finance a klíčová slova. Takto připravenou kampaň můžete aktivovat na určitý časový horizont nebo do vyčerpání finančních zásob. Díky propojení Google systémů máte poté možnost pracovat s těmito daty, jak v Google Analyst, tak v Google AdWords.

### 6.6.2. Cílové publikum

Sekce cílové publikum je pro změnu zaměřena na to, odkud pocházejí naši uživatelé, jaké používají technologie a jak se na webu chovají. Díky těmto sekcím můžeme analyzovat, co na našem webu převážně naši návštěvníci hledají, jaké adresy procházejí nejčastěji a také kolik času na webu tráví.



Obrázek 6.4 Přehled sekce cílové publikum za měsíc březen\*

\* Legenda

- **Návštěvy**  
Návštěvy představují počet návštěv webu.
- **Unikátní návštěvníci**

Unikátní návštěvníci představují počet neduplicitních (započítaných pouze jednou) návštěvníků webových stránek během konkrétního časového období.

- **Zobrazení stránek**

Zobrazení stránek představují celkový počet zobrazených stránek. Započítávají se opakovaná zobrazení jedné stránky.

- **Počet stránek na návštěvu**

Stránky/návštěva (průměrný počet stránek na návštěvu) je průměrný počet stránek zobrazených během návštěvy webu. Započítávají se opakovaná zobrazení jedné stránky.

- **Míra okamžitého opuštění**

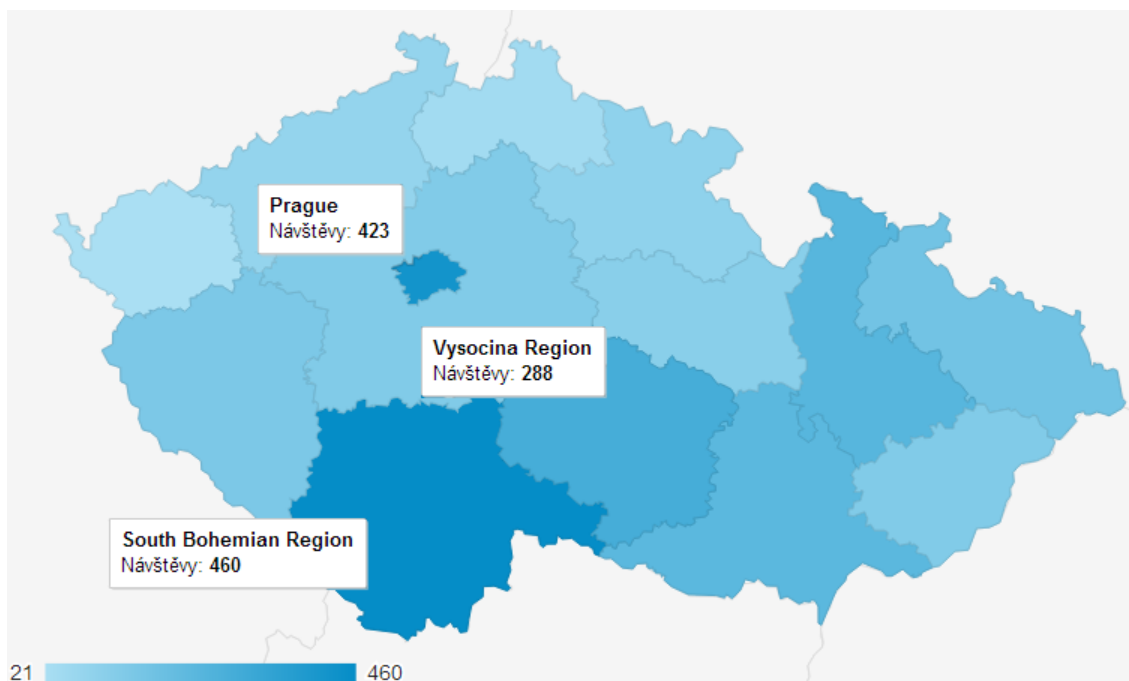
Míra okamžitého opuštění je procentuální podíl návštěv jedné stránky (tj. návštěv, při kterých uživatel opustil web již na vstupní stránce, aniž by uskutečnil interakci.

- **Procento nových návštěv**

Odhadované procento návštěvníků, kteří web navštívili poprvé.

## **Demografické údaje**

Můžeme předpokládat, že díky informacím poskytujícím přes webové brokery získává Google Analyst informaci o původu návštěvníků, nastaveném jazyku a mnoha dalších. Tyto údaje seskupuje v demografických údajích. Poměrně zajímavá je možnost zobrazení na mapě (viz obrázek 6.3).



Obrázek 6.5 Demografické údaje návštěvnosti za měsíc březen

## Chování

V sekci chování se pracuje s údaji zobrazujícími informace o době trvání návštěvy a o počtu vracejících se či nových uživatelů. Je tedy možné pracovat s informacemi o poměru nový vs. vracející se čtenář. Dále je možné získat křížovou tabulku o počtu opakovaných návštěv z procentuálního celkového počtu návštěv, čímž je možné získat informaci, kolik čtenářů se na web vrátilo a o jaké procento z celkové návštěvnosti se jedná.

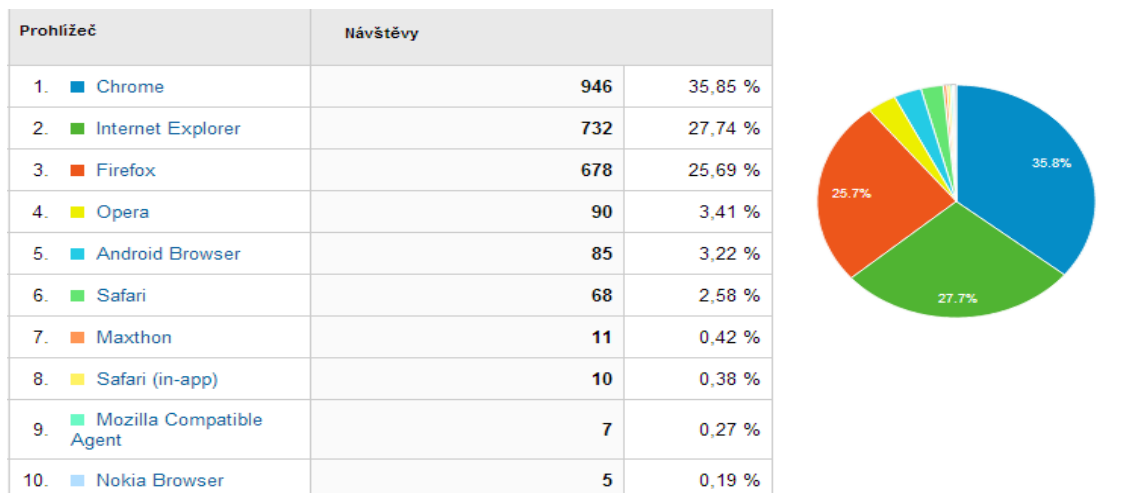
<input type="checkbox"/>	Typ návštěvníka	Návštěvy <sup>?</sup> ↓	Počet stránek na návštěvu <sup>?</sup>	Prům. doba trvání návštěvy <sup>?</sup>	Míra okamžitého opuštění <sup>?</sup>
		<b>2 639</b> Podíl z celku v %: 100,00 % (2 639)	<b>2,22</b> Průměr webu: 2,22 (0,00 %)	<b>00:01:23</b> Průměr webu: 00:01:23 (0,00 %)	<b>70,67 %</b> Průměr webu: 70,67 % (0,00 %)
<input checked="" type="checkbox"/>	1. New Visitor	1 930	1,88	00:00:58	74,25 %
<input checked="" type="checkbox"/>	2. Returning Visitor	709	3,15	00:02:31	60,93 %

Tabulka 6.8 Chování uživatelů – nový a vracející se uživatelé za měsíc březen

## Technologie

V této sekci si můžeme procházet používané technologie uživatelů, kteří navštěvují náš web a také z jaké sítě se připojují. Díky těmto údajům můžeme zvážit, zda webový projekt testovat i na dalších prohlížečích, než jsou nezákladnější. Pro tento

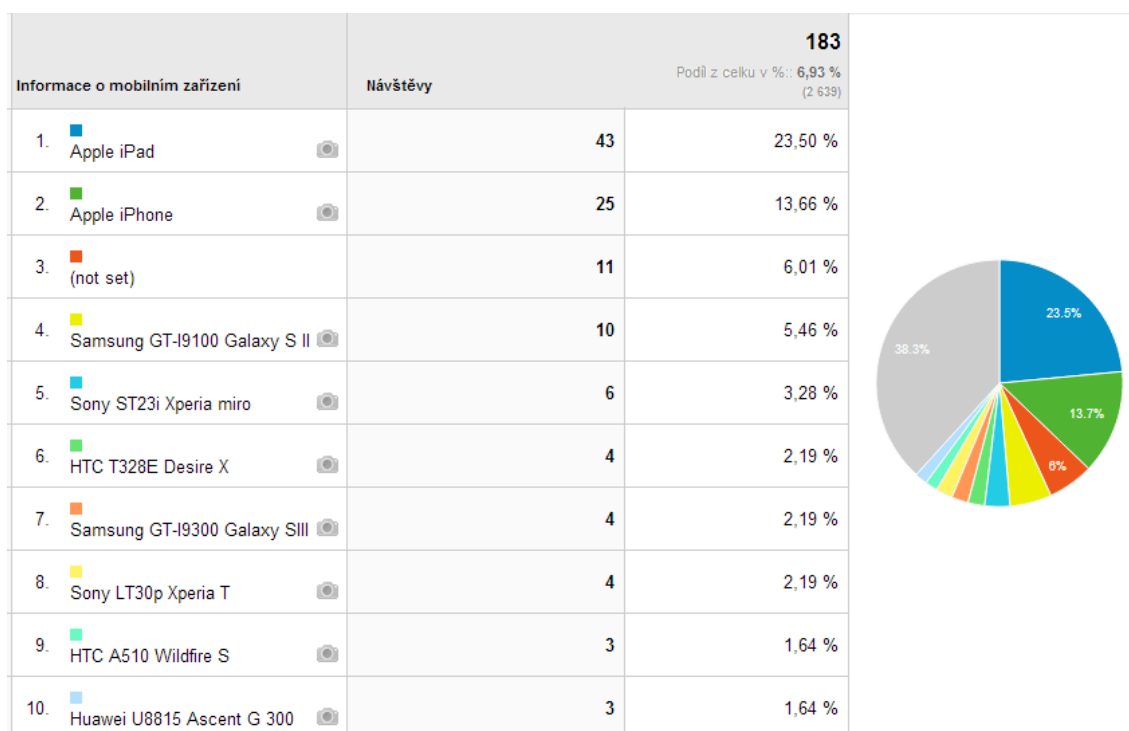
projekt je tedy stěžejní maximální optimalizace na prohlížeče Chrome, Internet Explorer a Firefox, kde tyto prohlížeče v níže uvedené tabulce obsazují první 3 místa. Ostatní prohlížeče zastávají v procentuálním počtu návštěv pouze zlomky z celkového počtu návštěv.



Tabulka 6.9 Používaná prohlížeče uživatelů za měsíc březen

## Mobil

V této sekci se můžeme zaměřit čistě na mobilní zařízení. První celkový přehled nám pouze rozděluje celkové návštěvy na mobilní a standardní. Pokud se ale podíváme na přehled ze zařízení, získáme informaci, jaká mobilní zařízení naši návštěvníci používají. Z těchto výsledků tedy můžeme získat potřebné údaje, abychom mohli určit, do jaké části se nám vyplatí web optimalizovat a dále jaké rozlišení vybrat jako výchozí. Další možností je otestovat provozuschopnost na menších rozlišeních, čímž získáme další údaj pro optimalizace našeho projektu.



Tabulka 6.10 Přehled návštěv z mobilních zařízení uživatelů za březen

## 6.7. Google AdWords

Google Adwords je reklamní systém, který je v současné době nejpoužívanějším PPC systémem v České republice i ve světě. Umožňuje uživatelům spravovat vlastní reklamní kampaně, přesné cílení, dobré nástroje pro přípravu a vyhodnocení. Důležitým faktorem pro užívání Google Adwords je také jeho příznivá cena.

Zobrazení reklamních bloků systému Google Adwords můžeme najít nejen při zobrazování výsledků hledání na Googlu, ale samozřejmě také u vyhledávačů a obsahových stránek partnerských webů a v dalších sítích. Díky tomuto celosvětovému pokrytí se zadané reklamy zobrazují i na síti Google Network – největší online reklamní síti, která svým pokrytím dosahuje až na 80 % uživatelů internetu v USA (patří do ní např. servery America online, Shopping.com, CompuServe, Netscape Netcentre). Dále se tyto reklamní bloky zobrazují také na tematických a zpravodajských serverech (např. Lycos, Weather.com, iVillage, InfoSpace, Bussines.com, The New York Times) a na Gmailu. Reklamy Google Adwords jsou k výsledkům vyhledávání na Googlu přiřazeny podle vyhledávaných klíčových slov, které jsou relevantní k danému

vyhledávání. Nejlépe ohodnocené reklamní bloky může přesunout pod nebo nad výsledky vyhledávání a také je zobrazit v modrém poli.

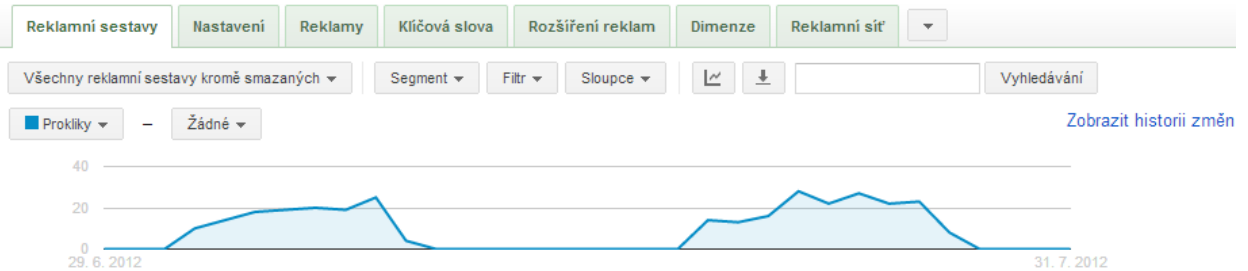
Pomocí Google Adwords je možné zacílit reklamu na určité lokality (země, regiony) nebo na stránky v požadovaném jazyce. Samozřejmostí je i vícebodové cílení. Aktuálně si může uživatel vybrat i cílení na určité webové stránky, které si může zvolit přímo. Sám si specifikuje, kde chce inzertní blok zobrazovat. Pro výběr těchto stránek je také možné použít automatické vyhledávání pomocí klíčových slov. V obou těchto případech již není reklama placena za klik, ale za 1000 zobrazení (CPM).

Pro kontrolu dosažených výsledků a také pro kontrolu výdajů nabízí Google AdWords různé statistiky a reporty, které jsou v sekci Report Center. Tato sekce umožňuje pracovat s různými statistikami, je zde možné generovat grafy a to vše až do nejnižší úrovně rozpadu – reklamy. Samozřejmostí jsou i sumační výstupy za celé reklamní sestavy či kampaně.

## **Kampaně**

Základním prvkem jsou kampaně. Kampaň tvoří ucelený celek, který obsahuje všechna potřebná data, aby jej bylo možné aktivovat a deaktivovat (začít zobrazovat/nezobrazovat reklamy). Pro kampaň je možné stanovovat denní rozpočet neboli denní limit finančních prostředků, které se mají za den investovat. Přímou úměrou investované částky je počet kliknutí, je tedy možné nastavovat denní limit s ohledem na to, kolik nových uživatelů je možné v jeden den přivést. Dále je možné pro kampaň zvolit, na kterých sítích Google se má reklama zobrazovat. V Google search se reklamy vypisují při vyhledávání na Google.cz. Pro Google network se reklamy zobrazují na partnerských webech používajících GoogleAdsense. Pro usnadnění tvorby kampaně obsahuje návrhář mnoho dalších vylepšení, například je možné si nechat vygenerovat alternativy ke klíčovým slovům a ty poté do reklamní kampaně pouze přidat.

● Aktivní Typ: Vyhledávací síť a reklamní síť – Všechny funkce Upravit Rozpočet: 100,00 Kč den Upravit Cílení: Česká republika Upravit  
Aktivní úpravy nabídek: Zařízení



+ Nová reklamní sestava Upravit Automatizovat Šítky

	Reklamní sestava	Stav	Výchozí max. CPC	Max. CPC pro reklamní síť	Prokliky	Zob.	CTR	Prům. CPC	Cena	Prům. pozice
	Reklamní sestava č. 1	OK	automatická: 10,00 Kč	automatická	302	420 007	0,07 %	5,51 Kč	1 664,24 Kč	2
	Celkem – všechny reklamní sestavy kromě smazaných				302	420 007	0,07 %	5,51 Kč	1 664,24 Kč	2
	Celkem – vyhledávací síť				214	41 197	0,52 %	4,92 Kč	1 053,04 Kč	1,7
	Celkem – reklamní síť				88	378 810	0,02 %	6,95 Kč	611,20 Kč	2,1
	Celkem – všechny reklamní sestavy				302	420 007	0,07 %	5,51 Kč	1 664,24 Kč	2

Obrázek 6.6 Google AdWords – ukázka reklamní kampaň č. 1

## Reklamní sestava

V zadané kampani si můžeme dále vytvořit reklamní sestavu. V této reklamní sestavě se již definuje vzhled reklamy a klíčová slova, pro která budeme chtít reklamu zobrazovat. Dále si zde také můžeme nastavit finanční limity, které nám zamezí zobrazování reklam na příliš „drahých“ místech a vyvarovat se tak utracení větších částek za jeden klik.

[Zpravodajství z Jezdeckví](#)  
www.zesedla.cz  
Nejnovější informace z jezdeckví  
Čerstvé reportáže a fotogalerie

● Aktivní

**Nabídky pro reklamní sestavy (Automatické)** ?

Výchozí nabídka **10,00 Kč**    Nabídka pro stránky v reklamní síti **automatická** ?

Aktivní úpravy nabídek: **Zařízení**

Nastavení	Reklamy	Klíčová slova	Publikum	Rozšíření reklam	Dimenze	Reklamní síť					
	Klíčové slovo	Stav <span style="font-size: small;">?</span>	↓ Max. CPC <span style="font-size: small;">?</span>	Prokliky <span style="font-size: small;">?</span>	Zobr. <span style="font-size: small;">?</span>	CTR <span style="font-size: small;">?</span>	Prům. CPC <span style="font-size: small;">?</span>	Cena <span style="font-size: small;">?</span>	Prům. pozice <span style="font-size: small;">?</span>	Štítky <span style="font-size: small;">?</span>	
	dostihy	<span style="color: red;">☐ Nízký počet zobrazení kvůli nízkému skóre kvality</span>	automatická: 10,00 Kč	31	1 545	2,01 %	3,53 Kč	109,53 Kč	1,2	--	
	jezdeckví	<span style="color: red;">☐ Nízký počet zobrazení kvůli nízkému skóre kvality</span>	automatická: 10,00 Kč	41	2 669	1,54 %	2,80 Kč	114,80 Kč	1,3	--	
	parkur	<span style="color: red;">☐ Nízký počet zobrazení kvůli nízkému skóre kvality</span>	automatická: 10,00 Kč	17	1 316	1,29 %	2,55 Kč	43,29 Kč	1,3	--	
	kůň	<span style="color: red;">☐ Nízký počet zobrazení kvůli nízkému skóre kvality</span>	automatická: 10,00 Kč	115	33 884	0,34 %	6,21 Kč	714,71 Kč	1,8	--	
	drezura	<span style="color: red;">☐ Nízký počet zobrazení kvůli nízkému skóre kvality</span>	automatická: 10,00 Kč	10	1 783	0,56 %	7,07 Kč	70,71 Kč	1,1	--	

Obrázek 6.7 Google AdWords – ukázka reklamní sestava č. 1.\*

\* Legenda

- **Stav**

Stav uvádí, zda klíčové slovo spouští reklamu či nikoliv. Co znamenají jednotlivé stavy: Stav, které ovládáte (např. pozastaveno) znamenají, že klíčové slovo reklamu nespouští. Stav, které souvisejí s naším schvalovacím procesem (např. aktivní nebo zamítnuto) uvádějí, zda se na dané klíčové slovo může zobrazovat reklama. Stav, které jsou ovlivněny dalšími faktory (např. nízký objem vyhledávání) znamenají, že klíčovému slovu brání v zobrazování reklamy jiné faktory.

- **Max. CPC**

Maximální cena za proklik (CPC) je nejvyšší částka, kterou jste ochotni zaplatit za to, že uživatel klikne na vaši reklamu. Proč je to důležité: Max. CPC představuje společně se skóre kvality a max. CPC vaší konkurence důležitý faktor při určování, kde se na stránce zobrazí vaše reklamy. Princip: za kliknutí nezaplatíte více, než kolik činí max. CPC, často zaplatíte ještě méně. Průměrnou částku, kterou platíte za kliknutí, zjistíte ve sloupci Prům. CPC.

- **Prokliky**

K prokliku dojde, pokud uživatel zareaguje na vaši reklamu kliknutím. To obvykle značí zájem získat další informace o vaší nabídce. Proč je to důležité: Počet prokliků může naznačit, nakolik vaše reklama oslovuje uživatele,



kterým se zobrazuje. Víte-li, kolik uživatelů kliklo na vaše reklamy ve srovnání s počtem zobrazení reklam, můžete měřit úspěšnost reklam. Prokliky schvaluje organizace Media Rating Council<sup>1</sup>.

- **Zobrazení**

Zobrazení uvádějí, jak často se vaše reklama zobrazila na stránce s výsledky vyhledávání nebo na webu v síti Google. Proč je údaj důležitý: Tento údaj vám napoví, jak často bylo předáno vaše reklamní sdělení. Můžete měřit potenciál reklamy přivádět na váš web návštěvníky. Provedete to tak, že porovnáte počet zaznamenaných zobrazení s mírou prokliku.

- **CTR**

Míra prokliku (CTR) je počet kliknutí na reklamu vydělený počtem zobrazení reklamy. Co je to: CTR vyjadřuje, jak často uživatelé kliknou na reklamu poté, co se jim zobrazí:

$$\text{CTR} = \frac{\text{Kliknutí na reklamy}}{\text{Zobrazení (zhlédnutí) reklamy}}$$

- **Průměrné CPC**

Průměrná cena za proklik (Prům. CPC) představuje průměrnou částku, která vám byla naúčtována za kliknutí na reklamu. Co je to: Tato částka představuje součet cen za všechny prokliky vydělený celkovým počtem zaznamenaných prokliků. Co to není: Průměrná CPC není totéž co maximální CPC – ta představuje nejvyšší částku, kterou jste ochotni zaplatit za kliknutí na reklamu. Za jednotlivý proklik vám nenaúčtujeme vyšší částku, než je vámi nastavená maximální CPC.

- **Cena**

Cena představuje součet všech nákladů vycházejících z ceny za proklik (CPC) a z ceny za tisíc zobrazení (CPM) během daného časového období. Co není zahrnuto: Zahrnuty nejsou další náklady, např. cena za volání. Součet všech nákladů naleznete ve sloupci Celkové náklady. Řízení nákladů: Nastavte na úrovni kampaně průměrný denní rozpočet, který vám vyhovuje, poté řiďte

---

<sup>1</sup> <https://support.google.com/adwords/answer/2616016&ctx=tltp>

útratu rozpočtu prostřednictvím nabídek na úrovni klíčových slov a reklamních sestav.

- **Průměrná pozice**

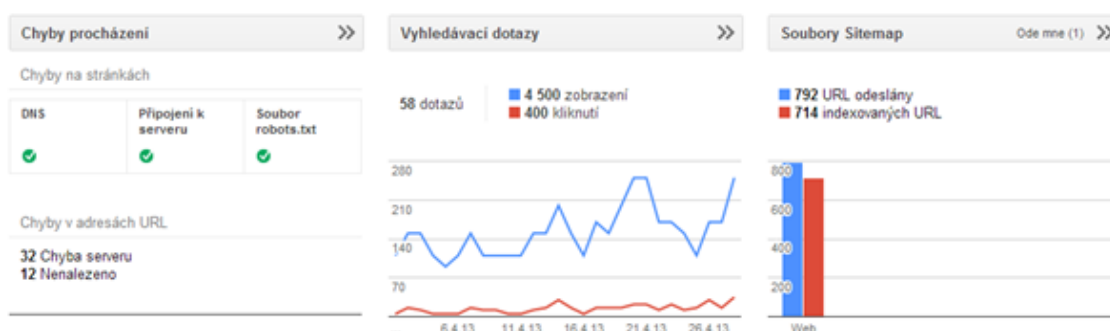
Průměrná pozice sděluje hodnocení reklamy v porovnání s jinými reklamami. Proč je to důležité: Tato metrika uvádí pozici reklamy ve vztahu k reklamám ostatních inzerentů. Další informace o hodnocení reklam v porovnání s ostatními inzerenty naleznete v přehledu Statistiky aukcí.

- **Štítky**

Pomocí štítků můžete uspořádat prvky v účtu do relevantních skupin. Budete tak moci rychle filtrovat a vytvářet přehledy s údaji, které vás zajímají nejvíce. Štítky můžete přiřadit klíčovým slovům, kampaním, reklamním sestavám či reklamám.

## 6.8. Google Webmaster

Google Webmaster je poslední aplikací od firmy Google, kterou můžeme pro sledování webu použít. Tento modul je zaměřený spíše na technické parametry webu, jako například SEO optimalizaci, chybové kódy a případně statistiku procházení Googlebot. Na níže uvedeném obrázku máme vyobrazený hlavní celkový přehled z této aplikace. Ihned z hlavní stránky tedy získáme informaci o chybách (nedostupné, chyba serveru – například 404 nenalezeno a podobně).

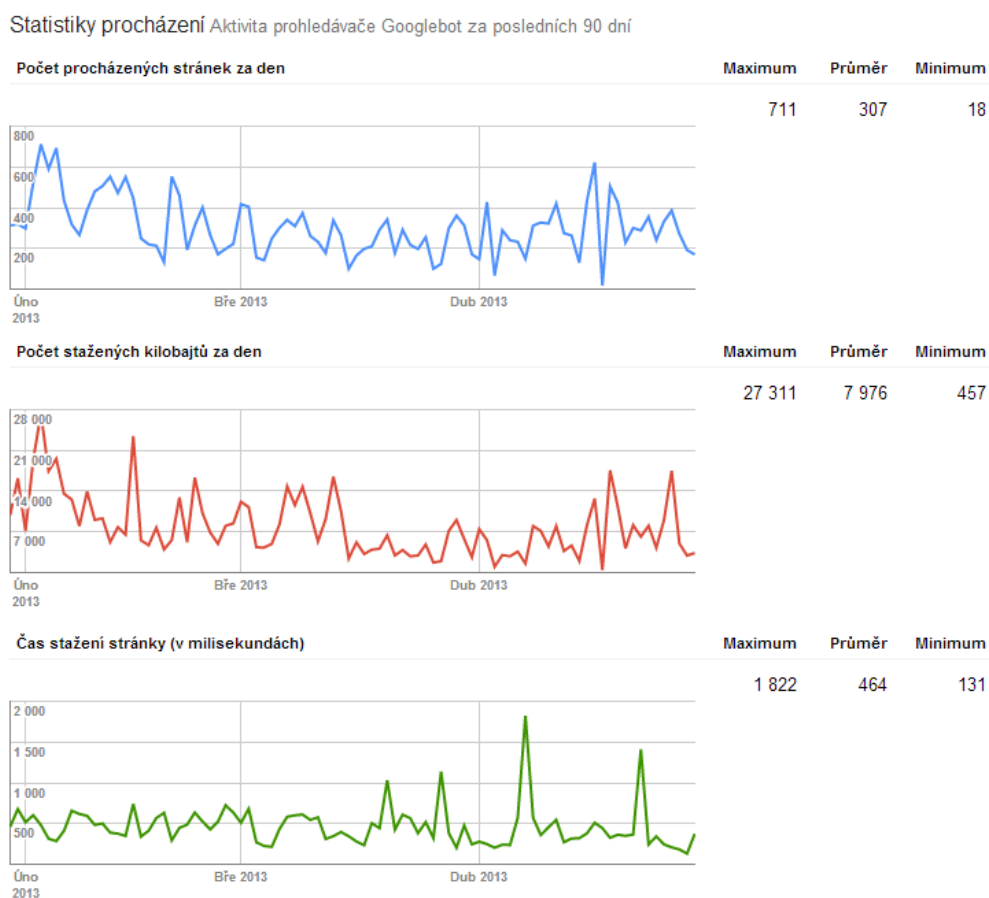


Obrázek 6.8 Google Webmaster - přehled

### Zdraví

Sekce, která pojednává o zdraví webu, nám dále nabízí chyby procházení, statistiky procházení Googlebot, přehled blokových URL adres a v neposlední řadě

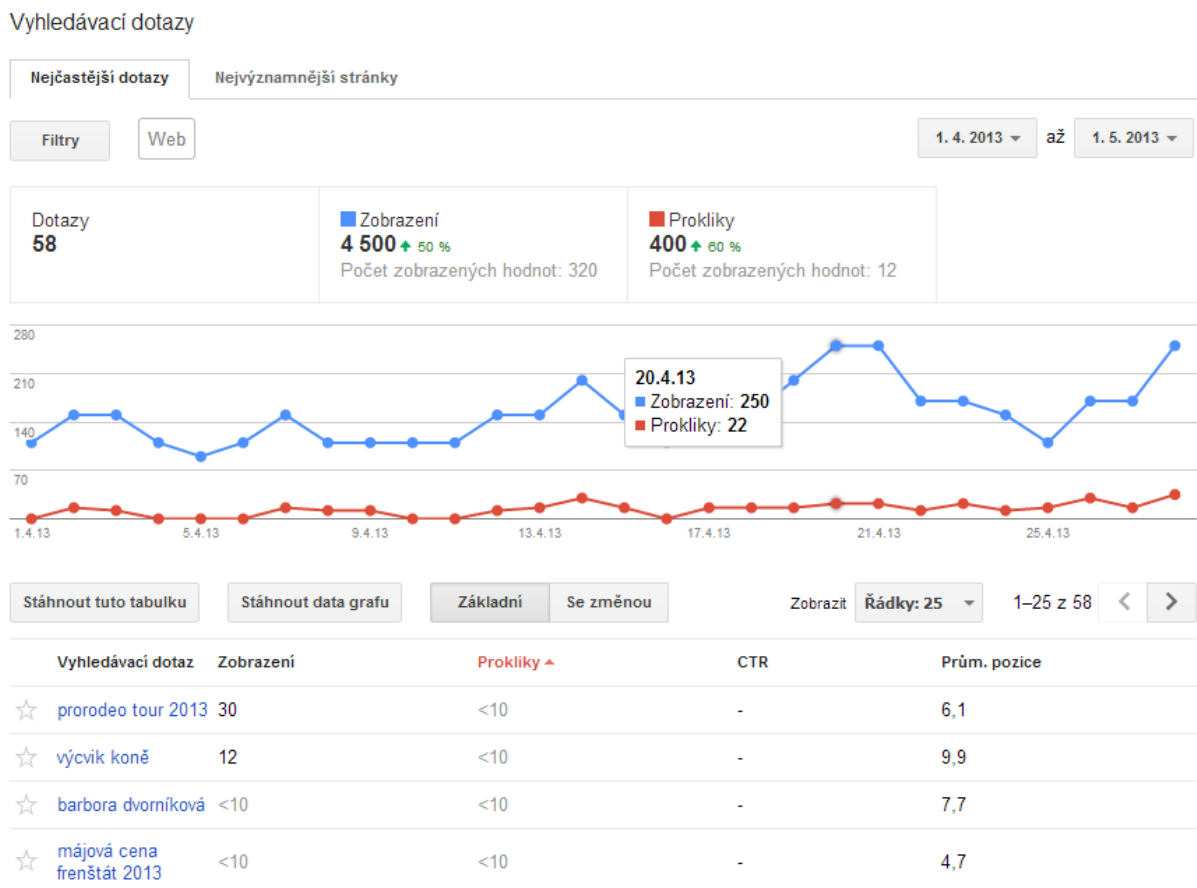
také stav indexu – indexování URL odkazů. Poměrně nová a možná spíše užitečná pro Google je možnost kontroly webu na malware.



Obrázek 6.9 Statistiky procházení Googlebot

## Provoz

Sekce provoz je zaměřena na vyhledávací dotazy – počet zobrazení, reálné umístění a počet odkazů. Dále si zde můžeme dopodrobna monitorovat odkazy na naše stránky a to i včetně porovnávání souborů (DOC, PDF). Díky této možnosti je snadné kontrolovat, zda někdo další nepublikuje naše vlastní soubory. Není ovšem k dispozici konkrétní popis, prozatím je tedy možné vycházet z reálných zkušeností – kde Google najde totožné soubory dle názvu a jejich velikosti. Díky tomu můžeme odhalit a případně nějakým způsobem reagovat na zveřejňování našeho obsahu bez výslovného souhlasu.



Obrázek 6.10 Provoz – vyhledávací dotazy na Google

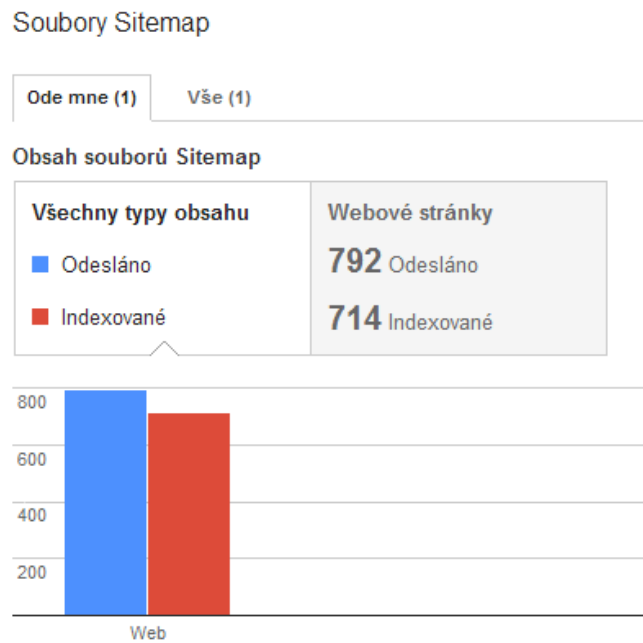
## Optimalizace

Tato sekce je jednou z nejrozsáhlejších. Jednou z nejdůležitějších funkcí je stav indexování Sitemap, díky níž můžeme pozorovat, zda je naše Sitemap správně nastavena a zda naše nové stránky bere Googlebot v úvahu – potažmo je zobrazuje ve vyhledávači. Dalším zajímavým přehledem této sekce jsou návrhy na vylepšení HTML kódu, díky kterému můžeme zjistit, zda máme správně umístěné elementy identifikující stránku (title, description). Poslední zde představenou částí bude část: obsahová a klíčová slova. Tato část nám ukáže, jaká slova mají na našem webu největší význam. V ideálním případě by se mělo jednat o slova, která náš web charakterizují a nejvíce se na něm vyskytují.

Z níže uvedeného přehledu můžeme získat informaci o stavu indexování webových stránek. V teoretické úvaze by se rozdíl odeslaných a indexovaných stránek měl rovnat 0. Pokud se ale podíváme na reálný přehled, indexovaných stránek je o něco méně. Důvody mohou být:

- Duplicitní URL adresy
- Zastaralé URL adresy
- Zakázaný obsah
- Nedostatečně strohý obsah

Teoreticky se tedy může jednat o všechny URL adresy, které nějakým způsobem porušují pravidla indexace nebo jsou pro Google nezajímavé či duplicitní.



Obrázek 6.11 Optimalizace – stav indexování Sitemap

## 7. Závěr

Na základě požadavků v zadání této práce byl vytvořen zpravodajský server o jezdeckví, který byl spuštěn do ostrého provozu. První část práce pojednává o obecném návrhu tohoto projektu a dále volně navazuje popisem jeho jednotlivých částí.

Výukový program byl inspirován již vytvořeným zpravodajským serverem o jezdeckví. Vzhledem k cílům této práce, byl k této webové aplikaci vytvořen doplňující výukový text. Dále bylo pro větší pedagogickou hodnotu této práce vytvořeno několik menších výukových programů opět i s doplňujícími texty. Veškeré přílohy jsou pro vysokou dostupnost koncipovány jako komplexní HTML stránky. Pro možnost rychlé orientace, jsou všechny přílohy koncipovány tak, aby byly použitelné i bez plného textu této diplomové práce.

S ohledem na rozsah zadání je do práce ještě zařazen úvod do analytických technologií od Google: Analyst, Webmaster a AdWords, které jsou v určitém rozsahu volně dostupné. Dle rozsahu zpracování výše uvedených analytických nástrojů je možné vyčlenit tuto část práce jako samostatnou a tím umožnit další studijní uplatnění této práce i pro studenty nezabývající se přímo J2EE, ale například jinou webovou technologií.

Díky včasnému zvládnutí webového projektu byl možný půlroční test projektu na dvou vybraných hostinzích. O tomto provozu pojednává poslední kapitola - reálný provoz, která tuto práci doplnila o zajímavé postřehy z ostrého provozu webové aplikace.

## Seznam zkratek

<b>AJAX</b>	Asynchronous JavaScript and XML
<b>API</b>	Application Programming Interface
<b>ASP</b>	Active Server Pages
<b>ASP.NET</b>	Active Server Pages. Network
<b>BSD</b>	Berkeley Software Distribution
<b>CAI</b>	Computer Asisted Instruction
<b>CBI</b>	Computer Based Instruction
<b>CLR</b>	Common Language Runtime
<b>DAO</b>	Data Access Object
<b>IDE</b>	Integrated Development Enviroment
<b>GPS</b>	Global Positioning System
<b>HTML</b>	Hyper Text Markup Language
<b>HTTP</b>	Hypertext Transfer Protocol
<b>HW</b>	Hardware
<b>ID</b>	Identifikátor
<b>J2EE</b>	Java 2 Enterprise Edition
<b>JDBC</b>	Java Database Connectivity
<b>JDO</b>	Java Data Objects
<b>JVM</b>	Java Virtual Machine
<b>JSF</b>	Java Server Faces
<b>JSP</b>	Java Server Pages
<b>MS SQL</b>	Microsoft Structured Query Language
<b>MVC</b>	Model View Controller
<b>MySQL</b>	My Structured Query Language
<b>ORM</b>	Object Relational Mapping
<b>PHP</b>	Personal Home Page
<b>SW</b>	Software
<b>VPS</b>	Virtual Private Server
<b>XHTML</b>	Extensible Hypertext Markup Language

## Seznam tabulek

Tabulka 6.1 Vynaložené finanční náklady pro virtuální server .....	82
Tabulka 6.2 Vynaložené finanční náklady pro J2EE hosting .....	82
Tabulka 6.3 Porovnání HW komponent virtuálního serveru vs J2EE hostingu .....	83
Tabulka 6.4 Porovnání použitých hostingů .....	86
Tabulka 6.5 Provoz z vyhledávání .....	87
Tabulka 6.6 Návštěvnost z odkazujících stránek .....	88
Tabulka 6.7 Přímá návštěvnost .....	88
Tabulka 6.8 Chování uživatelů – nový a vracející se uživatelé za měsíc březen .....	91
Tabulka 6.9 Používaná prohlížeče uživatelů za měsíc březen .....	92
Tabulka 6.10 Přehled návštěv z mobilních zařízení uživatelů za březen .....	93



## Seznam obrázků

Obrázek 3.1 ERA diagram zeSedla.cz.....	26
Obrázek 3.2 Návrh třídy uživatele.....	28
Obrázek 3.3 Návrh třídy soubor .....	29
Obrázek 3.4 Návrh třídy zpráva.....	31
Obrázek 3.5 Návrh třídy článku.....	32
Obrázek 3.6 Návrh třídy akce .....	35
Obrázek 3.7 Zobrazení akcí v kalendáři .....	36
Obrázek 3.8 Návrh třídy inzerátu .....	38
Obrázek 3.9 Návrh tříd koně a soutěže.....	39
Obrázek 4.1 Facebook - návrh tlačítka to se mi líbí .....	54
Obrázek 4.2 Facebook - návrh tlačítka odeslat.....	55
Obrázek 4.3 Facebook - návrh pluginu pro komentáře .....	57
Obrázek 4.4 Twitter - sdílení obsahu webové stránky pomocí tlačítka Tweet.....	59
Obrázek 4.5 Twitter - návrh tlačítka Follow.....	60
Obrázek 4.6 Google+ - grafický návrh tlačítka .....	62
Obrázek 4.7 Google+ - grafický návrh tlačítka sdílej.....	63
Obrázek 5.1 Náhled webu pro stahování elektronických materiálů .....	64
Obrázek 6.1 Přehled rychlosti webu za provozu na Wedos virtuálním serveru .....	84
Obrázek 6.2 Přehled rychlosti webu za provozu na Mochahost J2EE hostingu.....	84
Obrázek 6.3 Zdroje návštěvnosti .....	87
Obrázek 6.4 Přehled sekce cílové publikum za měsíc březen* .....	89
Obrázek 6.5 Demografické údaje návštěvnosti za měsíc březen.....	91
Obrázek 6.6 Google AdWords – ukázka reklamní kampaň č. 1 .....	95
Obrázek 6.7 Google AdWords – ukázka reklamní sestava č. 1.*.....	96
Obrázek 6.8 Google Webmaster - přehled.....	98
Obrázek 6.9 Statistiky procházení Googlebot .....	99
Obrázek 6.10 Provoz – vyhledávací dotazy na Google .....	100
Obrázek 6.11 Optimalizace – stav indexování Sitemap .....	101

## Seznam schémat

Schéma 3.1 Případy užití projektu zeSedla.cz .....	25
Schéma 3.2 Hierarchické členění typu zboží.....	37
Schéma 4.1 Modulární popis Spring framework .....	43
Schéma 4.2 Zpracovávání požadavků Spring MVC.....	44
Schéma 5.1 Model validace dat pomocí Ajax .....	67
Schéma 5.2 Model postupu informací o uživateli .....	69
Schéma 5.3 Model postupu načítání / stahování souborů.....	70
Schéma 5.4 Model postupu zpracování zpráv a článků.....	71
Schéma 5.5 Model postupu zpracování akcí .....	72
Schéma 5.6 Model postupu zpracování inzerátu .....	73
Schéma 5.7 Model postupu zpracování požadavků ve stáji .....	74
Schéma 5.8 Návrh rozložení hlavní stránky .....	75
Schéma 5.9 Model postupu zpracování požadavku pro Sitemap.xml .....	76

## Seznam zdrojových kódů

Výpis zdrojového kódu 4.1 Facebook - tlačítko to se mi líbí .....	53
Výpis zdrojového kódu 4.2 Facebook - element pro umístění tlačítka to se mi líbí .....	53
Výpis zdrojového kódu 4.3 Facebook - tlačítko odeslat.....	55
Výpis zdrojového kódu 4.4 Facebook - element pro umístění tlačítka odeslat .....	55
Výpis zdrojového kódu 4.5 Facebook - plugin pro komentáře.....	56
Výpis zdrojového kódu 4.6 Facebook - element pro umístění komentářů .....	56
Výpis zdrojového kódu 4.7 Twitter - tlačítko tweet .....	58
Výpis zdrojového kódu 4.8 Twitter - element pro umístění tlačítka .....	58
Výpis zdrojového kódu 4.9 Twitter - tlačítko sleduj .....	59
Výpis zdrojového kódu 4.10 Twitter - element pro umístění tlačítka sleduj.....	60
Výpis zdrojového kódu 4.11 Google+ - tlačítko +1 .....	61
Výpis zdrojového kódu 4.12 Google+ - element pro umístění tlačítka +1 .....	61
Výpis zdrojového kódu 4.13 Google+ - tlačítko sdílej .....	62
Výpis zdrojového kódu 4.14 Google+ - element pro umístění tlačítka sdílej .....	62
Výpis zdrojového kódu 5.1 Demonstrátor - konfigurace připojení k databázi.....	65
Výpis zdrojového kódu 5.2 Demonstrátor - předání parametrů připojení k databázi DS .....	66
Výpis zdrojového kódu 5.3 Demonstrátor - nastavení validačních parametrů modelu akce .....	68
Výpis zdrojového kódu 5.4 Demonstrátor - ukázka formulářového pole v JS.....	68
Výpis zdrojového kódu 5.5 Demonstrátor - definice připojení k databázi .....	77

## Seznam použité literatury

- [Zbi07] Adam Zbiejczuk, Mediální Studia, Masarykovy univerzity v Brně 2007
- [WRohl] PhDr. Lucie Rohlíková, Ph.D., Konstruktivismus v praxi vysokých škol, 2010  
<http://www.konstruktiv.zcu.cz/menu.php>
- [WMuB] Masarykova univerzita Brno, Kurz práce s informacemi  
[http://is.muni.cz/elportal/estud/ff/js07/informace/materialy/pages/citace\\_opora.pdf](http://is.muni.cz/elportal/estud/ff/js07/informace/materialy/pages/citace_opora.pdf)
- [Cra07] Craig Walls: *Spring in Action*, nakladatelství Manning Publications; Second Edition edition (August 23, 2007), ISBN: 978-1933988139
- [Chri04] Chrisian Bauer: *Hibernate in Action*, nakladatelství Manning Publications (August 1, 2004), ISBN: 978-1932394153
- [Bed11] BEDNÁŘ, Vojtěch. *Marketing na sociálních sítích: prosadte se na Facebooku a Twitteru*. nakladatelství CPress (2011). ISBN 978-80-251-3320-0
- [WSpr] Spring <http://www.springsource.org/>
- [WAp] Apache Tomcat <http://tomcat.apache.org/>
- [WHib] Hibernate <http://www.hibernate.org/>
- [WG+] Google + <https://support.google.com/plus/?hl=cs>
- [WFac] Facebook social plugin <https://developers.facebook.com/docs/plugins/>
- [WTwi] Twitter developeps documentation <https://dev.twitter.com/docs/>
- [WGA] Google Analytics <https://support.google.com/analytics/?hl=cs>
- [WGW] Google Webmaster <https://support.google.com/webmasters/?hl=cs>
- [WGA] Google AdWords <https://support.google.com/adwords/?hl=cs>

## **Přílohy**

**[1] STS Projekt Hello Word Spring:**

CD\1. Hello Word Spring MVC\Hello\_Word\_Spring\_MVC\\*.\*

**[2] Dokumentace k Hello Word Spring:**

CD\1. Hello Word Spring MVC\Hello\_Word\_Spring\_MVC.docx

**[3] Programová dokumentace k Hello Word Spring:**

CD\1. Hello Word Spring MVC\Programová  
dokumentace\index.html

**[4] Spustitelná verze Hello Word Spring:**

CD\1. Hello Word Spring MVC\root.WAR

**[5] STS Projekt Hello Word Ajax:**

CD\2. Hello Word Ajax\Hello\_Word\_Ajax\

**[6] Dokumentace k Hello Word Ajax:**

CD\2. Hello Word Ajax\Hello\_Word\_Ajax.docx

**[7] Programová dokumentace k Hello Word Ajax:**

CD\2. Hello Word Ajax\Programová dokumentace\index.html

**[8] Spustitelná verze Hello Word Ajax:**

CD\2. Hello Word Ajax\root.WAR

**[9] STS Projekt Hello Word Hibernate:**

CD\3. Hello Word Hibernate\Hello\_Word\_Hibernate\

**[10] Dokumentace k Hello Word Hibernate:**

CD\3. Hello Word Hibernate\Hello\_Word\_Hibernate.docx

**[11] Programová dokumentace k Hello Word Hibernate:**

CD\3. Hello Word Hibernate\Programová  
dokumentace\index.html

**[12] Spustitelná verze Hello Word Hibernate:**

CD\3. Hello Word Hibernate\root.WAR

**[13] HTML ukázky použití - Hello Word jQuery:**

CD\4. Hello Word jQuery\01\_tlacitko.html

CD\4. Hello Word jQuery\02\_zalozky.html

CD\4. Hello Word jQuery\ 03\_accordion.html  
CD\4. Hello Word jQuery\ 04\_autocomplete.html  
CD\4. Hello Word jQuery\ 05\_dialog\_modalni\_formular.html  
CD\4. Hello Word jQuery\ 06\_menu\_ikony.html  
CD\4. Hello Word jQuery\ 07\_progressbar.html  
CD\4. Hello Word jQuery\ 08\_slider\_rozsah\_hodnot.html  
CD\4. Hello Word jQuery\ 09\_tooltip.html  
CD\4. Hello Word jQuery\ 10\_datepicker\_3\_mesice.html

**[15] Dokumentace k Hello Word jQuery:**

CD\4. Hello Word jQuery\Hello\_Word\_jQuery.docx

**[16] HTML Projekt Hello Word Google Map API:**

CD\5. Hello Word Google Map API\01\_zakladni\_zobrazeni.html  
CD\5. Hello Word Google Map API\02\_zadani\_souradnic.html

**[17] Dokumentace k Hello Word Google Map API:**

CD\5. Hello Word Google Map  
API\Hello\_Word\_Google\_Map\_API.pdf

**[18] STS Projekt Demonstrátor:**

CD\Demonstrátor\demonstrator

**[19] Dokumentace a schémata k demonstrátoru:**

CD\Demonstrátor\Demonstrátor.docx

**[20] Programová dokumentace a schémata k demonstrátoru:**

CD\Demonstrátor\Programová dokumentace\index.html

**[21] UML diagramy k demonstrátoru:**

CD\Demonstrátor\UML\\*.\*

**[22] Spustitelná verze demonstrátoru:**

CD\ Demonstrátor\root.WAR