

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INTERNETOVÝ AUTOBAZAR

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

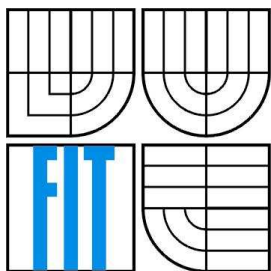
AUTOR PRÁCE
AUTHOR

ZDENĚK HORÁK

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INTERNETOVÝ AUTOBAZAR

INTERNET USED-CARS SHOP

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

ZDENĚK HORÁK

VEDOUCÍ PRÁCE
SUPERVISOR

ING. ROMAN SKŘIVÁNEK

BRNO 2007

Internetový autobazar

Internet Used-Cars Shop

Vedoucí:

Skřivánek Roman Ing., UITS FIT VUT

Oponent:

Jurka Pavel Ing., UITS FIT VUT

Přihlášen:

Horák Zdeněk

Zadání:

1. Seznamte se s požadavky, kladenými na informační systém autobazaru.
2. Za pomoci vhodných modelovacích nástrojů navrhnete systém pro elektronickou prezentaci autobazaru.
3. Prostudujte techniky programování webových informačních systémů (především PHP a SQL).
4. Navržený systém implementujte.
5. Zhodnoťte dosažené výsledky a diskutujte možnosti dalšího pokračování.

Kategorie:

Web

Literatura:

- Dle pokynů vedoucího

Licenční smlouva

Licenční smlouva je uložena v archivu Fakulty informačních technologií Vysokého učení technického v Brně.

Abstrakt

Tato bakalářská práce obsahuje analýzu, návrh a implementaci informačního systému internetového autobazaru. Systém eviduje zejména informace o automobilech, které autobazar nabízí k prodeji. O každém automobilu se uchovávají jeho základní údaje, údaje o bývalém majiteli, obecné informace o značce vozidla a údaje o změnách ceny automobilu. Do systému se můžou pomocí přihlašovacího jména a hesla přihlásit dva druhy uživatelů. Prvním z těchto uživatelů je klient autobazaru, kterému je dovoleno prohlížet katalog nabízených vozidel a případně provádět jejich rezervaci. A dále pak správce systému, který může všechny uchovávané záznamy editovat. Součástí autobazaru je diskusní fórum.

Klíčová slova

Internet, autobazar, internetový autobazar, www, webová aplikace, HTML, PHP, MySQL, Javascript, databáze, automobil, auto, klient-server.

Abstract

This bachelor's thesis contains the analysis, the design and the information system implementation of the Internet used-cars shop. The system files predominantly information about the cars which the used-cars shop offers for sale. The system saves basic data about each car, the data about the former keeper, general information about the car producer and about the car price changes. Two kinds of users can log into the system with a specific password. The customer of the used-cars shop who is allowed to view the catalogue of offered cars and make a reservation. The second is the system administrator who can edit saved data. The used-cars shop forum is also available.

Keywords

Internet, used-cars shop, internet used-cars shop, www, web application, HTML, PHP, MySQL, Javascript, database, car, client-server.

Citace

Zdeněk Horák: Internetový autobazar, bakalářská práce, Brno, FIT VUT v Brně, 2007

INTERNETOVÝ AUTOBAZAR

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Romana Skřivánka.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Zdeněk Horák
10. května 2007

Poděkování

Tímto bych chtěl poděkovat Ing. Romanu Skřivánkovi za odbornou pomoc, kterou mi při vytváření práce poskytoval.

© Zdeněk Horák, 2007.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..

Obsah

Obsah.....	1
1 Úvod.....	3
2 Analýza autobazaru.....	4
2.1 Analýza stránek z hlediska funkčnosti.....	4
2.1.1 Uživatelské cíle.....	4
2.1.2 Požadavky na systém.....	5
2.1.3 Integritní omezení.....	7
2.2 Systémové požadavky.....	8
2.3 Analýza stránek z pohledu návrháře.....	8
2.3.1 Vynechejte zbytečná slova.....	8
2.3.2 Hierarchie stránek.....	9
2.3.3 Logo.....	10
2.3.4 Odkazy.....	10
2.3.5 Navigace.....	11
3 Přístupnost webových stránek.....	12
3.1 Vlastnosti přístupných webových stránek.....	13
3.1.1 Obsah webových stránek je dostupný a čitelný.....	13
3.1.2 Práci s webovou stránkou řídí uživatel.....	14
3.1.3 Informace jsou srozumitelné a přehledné.....	14
3.1.4 Odkazy jsou zřetelné.....	15
3.1.5 Ovládání webové stránky je jasné a přehledné.....	15
3.1.6 Kód je technicky způsobilý a strukturovaný.....	15
4 Návrh databáze autobazaru.....	16
4.1 Vytvoření ER diagramu.....	16
4.2 Tvorba databáze a tabulek.....	18
4.2.1 Co je to databáze?.....	18
4.2.2 Co je to SQL?.....	18
4.2.3 Vytvoření databáze pomocí SQL skriptu.....	19
4.2.4 Použití fotografie v MySql.....	20
4.2.5 Vytvoření tabulek pomocí SQL skriptu.....	20
5 Návrh systému autobazaru.....	22
5.1 Základ.....	22
5.2 Uživatel.....	23
5.3 Správce.....	23

5.4	Diskuse.....	25
6	Použité prostředky a jazyky.....	26
6.1	HTML.....	26
6.1.1	Vývoj jazyka.....	26
6.1.2	Verze jazyka.....	27
6.2	CSS.....	27
6.2.1	Výhody.....	28
6.2.2	Nevýhody.....	28
6.3	PHP.....	28
6.4	Javascript.....	29
6.5	MySql.....	30
7	Závěr.....	31
	Literatura.....	32
	Seznam příloh.....	33

1 Úvod

Cílem této bakalářské práce je seznámit se s požadavky, kladenými na informační systém autobazaru. Požadovaný systém navrhnout a pomocí vhodných programovacích technik systém implementovat.

Informační systém je modelem systému reálného světa. Slouží k zpracování a uchování dat a informací. Informační systém internetového autobazaru získává informace především pomocí správce systému. Jedním z hlavních cílů je vytvořit systém, co nejjednodušší, aby činnost se systémem byla pro uživatele i správce snadná.

Informační systém internetového autobazaru uchovává především údaje o nabízených automobilech, o jejich bývalých majitelích, o průběhu cen vozidla, o značkách vozidel,.... Uživatelé systému mohou prohlížet katalog nabízených vozidel a v případě zájmu si mohou automobil rezervovat. Správce systému má pravomoc všechny informace o vozidlech editovat. Součástí informačního systému je i diskusní fórum, kde mohou přihlášení uživatelé a správce systému diskutovat na různá témata.

V kapitole č. 2 jsou popsány požadavky autobazaru na výsledný informační systém. Je zde provedena analýza autobazaru a to jak z hlediska funkčnosti tak z hlediska pohledu návrháře webového systému.

V kapitole č. 3 je popsán návrh databáze internetového autobazaru. Nejdříve je zde vytvořen ER diagram a poté je popsáno vytvoření databáze a tabulek databáze pomocí SQL skriptu. Celkový návrh tabulek databáze pomocí skriptu je uveden v příloze 1.

V další kapitole je popsán návrh webové aplikace informačního systému. Celý systém je zde rozdělen na 4 části a ty jsou zde probrány každá samostatně.

Kapitola č. 5 popisuje možnosti dalšího rozšíření autobazaru.

V kapitole č. 6 jsou uvedeny základní charakteristiky prostředků a jazyků použitých při implementaci systému.

Závěrem je výsledný systém zhodnocen.

2 Analýza autobazaru

2.1 Analýza stránek z hlediska funkčnosti

Autobazar má zájem o vytvoření informačního systému, který by umožňoval klientům prohlížení nabízených automobilů (bývalých majitelů, značky) a jejich případnou rezervaci.

Automobil, který si klient autobazaru rezervoval, si musí vyzvednout do týdne od data rezervace. Pokud tak neučiní, bude rezervace automaticky zrušena a začne platit v pořadí další rezervace (Klienti mohou provést rezervaci i v případě, že automobil má již někdo rezervovaný. V takovém případě začne jejich rezervace platit, když bude 1. rezervace resp. další rezervace zrušena nebo vyprší lhůta vyzvednutí).

Doba vyzvednutí je opět týden od data začátku platnosti rezervace. Autobazar v případě zrušení rezervace nebo vypršení lhůty musí dalším rezervovaným klientům poslat zprávu na email uvedený při registraci.

Pokud si klient automobil vyzvedne, autobazar zruší nabízený automobil z internetového katalogu a případným dalším rezervovaným klientům zašle zprávu a prodeji motorového vozidla.

Rezervaci můžou provádět pouze zaregistrovaní a přihlášení uživatelé. Přihlášený uživatel se může odhlásit. Po 10 minutách nečinnosti systém uživatele odhlásí automaticky. Uživatel má možnost změnit své registrační údaje.

Součástí autobazaru je i diskusní fórum, kde přihlášení uživatelé a správce systému můžou komunikovat v různých diskusních skupinách.

Sídlo autobazaru:

AutoNet

Závodní 4781

130 00 Praha

2.1.1 Uživatelské cíle

Cíle systému, které specifikuje uživatel, tedy klient autobazaru. Jedná se o stručný popis, co by měl systém provádět z pohledu uživatele. Jedná se o základní kámen při vytváření informačního systému vznikající na straně uživatele, který zcela neformálně, stručně, výstižně a obecně i ne zcela přesně popíše, co po informačním systému požaduje.

Uživatelské cíle internetového autobazaru jsou následující:

- přihlásit se jako uživatel systému
- prohlížení katalogu nabízených automobilů i bez přihlášení
- rezervace automobilu, o které má uživatel zájem
- zrušení rezervace automobilu, které má uživatel zaregistrované
- možnost dalších rezervací pro případ, že dřívější zájemci rezervaci zruší
- možnost zasílání připomínek správci autobazaru
- možnost komunikace pomocí diskusního fóra
- odhlášení se
- změnit své registrační údaje

2.1.2 Požadavky na systém

V této části se budu zabývat tím, co bude systém provádět. Nikoli jak to bude provádět. Jedná se především o požadavky na to, co by měl informační systém uchovávat za data, tedy jaké údaje by měl ve své databázi udržovat. Dále popíši možnosti správce autobazaru a automatické činnosti autobazaru.

Systém musí obsahovat:

a) údaje o klientech:

- jméno
- příjmení
- rodné číslo
- adresu
- email
- uživatelské jméno
- heslo
- datum a čas registrace
- datum posledního přihlášení
- oslovení
- platnost

b) údaje o motorových vozidlech:

- značka
- model
- rok výroby
- cena
- palivo

- výkon
 - karoserie
 - převodovka
 - tachometr
 - barva
 - výbava
 - majitel
- c) záznam o rezervacích:
- ID klienta, který provedl rezervaci
 - ID vozidla
 - pořadí rezervace u daného vozidla
 - datum a čas rezervace
 - datum a čas začátku platnosti rezervace
- d) údaje o bývalých majitelích jednotlivých vozidel:
- jméno
 - příjmení
 - rodné číslo
 - adresa
- e) údaje o výrobcích vozidel (o značkách vozidel):
- značka
 - sídlo
 - rok založení
 - domovská stránka
 - logo
- f) údaje o cenách vozidel:
- ID vozidla
 - pořadí ceny
 - datum a čas začátku platnosti
 - cena
- g) přihlašovací jméno a heslo správce systému:
- h) tabulka s komentáři od uživatelů:
- datum a čas komentáře
 - jméno a email uživatele
 - komentář
- i) tabulka s příspěvky z diskusního fóra:
- diskusní skupina

- datum a čas příspěvku
 - uživatel
 - ID komentáře, na který příspěvek reaguje
 - příspěvek
- j) tabulka se soubory a fotografiemi:
- datum a čas vložení souboru
 - ID vozidla
 - značka
 - pořadí
 - soubor

Možnosti správce autobazaru:

- editovat všechny tabulky.

Automatické činnosti autobazaru:

- označení nepřečtených komentářů od uživatelů v editační části autobazaru
- kontrola správnosti vyplnění všech formulářů v systému
- zasílání automatických emailů (např. při zrušení rezervace od správce, při změně údajů o automobilu)
- automatické zrušení rezervace (pokud si uživatel do týdne od začátku platnosti rezervace automobil nevyzvedne)
- zamezení registrace uživatele s uživatelským jménem, který již používá nějaký jiný uživatel nebo správce
- zamezení změny přihlašovacího jména správce na jméno, který již používá nějaký jiný uživatel

2.1.3 Integritní omezení

Jsou to omezení určující správnost dat, která existují v reálném světě.

Data musejí splňovat následující integritní omezení:

- primární klíče jednotlivých tabulek jsou unikátní
- rok výroby musí být menší nebo roven roku stávajícímu
- cena nesmí být záporné číslo
- tachometr nesmí být záporné číslo
- rok založení automobilky musí být menší nebo roven roku stávajícímu
- rezervaci může provést pouze zaregistrovaný a přihlášený uživatel

- zrušení rezervace může provést pouze zaregistrovaný a přihlášený uživatel nebo správce systému
-

2.2 Systémové požadavky

- Systém musí být implementován jako klient/server. Klientské počítače budou PC, na kterých se budou provádět aplikační programy.
- Uživatelské rozhraní bude grafické a jednoduše pochopitelné pro uživatele systému.
- Databáze bude s rozhraním SQL, která běží na dostupném serveru.

2.3 Analýza stránek z pohledu návrháře

Jedná se o požadavky na vyvíjený informační systém z hlediska návrhářů webových stránek. Zabýváme se zde základními principy rozvržení stránek zejména z pohledu jejich přehlednosti. A dalšími znaky, které by měla každá kvalitní webová stránka dodržovat.

Základní pravidlo zní: „Nenuťme uživatele přemýšlet!“ Tedy naučme uživatele pohybovat se na našich stránkách intuitivně bez přílišného namáhání mozkových buněk. Uživatelé to ocení a budou se rádi vracet.

2.3.1 Vynechejte zbytečná slova

Jedna z nejlépe dokumentovaných pravd o používání internetu říká, že lidé stráví jen velmi krátkou dobu čtením webových stránek. Místo čtení je vlastně jen prohlížíme (přelétáváme) a pátráme po slovech, která nás upoutají. Výjimku samozřejmě tvoří internetové noviny atd.

Proč stránky jen prohlížíme:

- čtení zdržuje
- hledáme jen to co nás zajímá
- zvyk (z novin, časopisů, knih, ...)

Odstranění zbytečných slov způsobí:

- užitečný obsah bude nápadnější
- stránky budou kratší – uživatelé uvidí víc, aniž by je museli posouvat
-

Informační systém internetového autobazaru musí být tedy stručný a výstižný (rozsáhle návody, přátelské monology atd. stejně nikdo nečte).

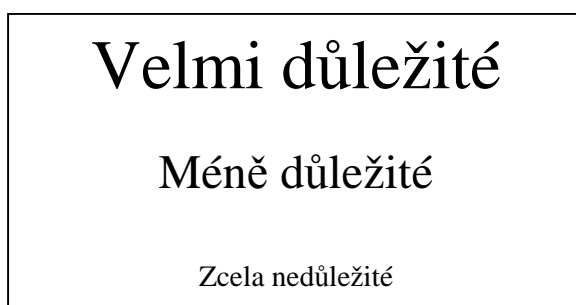
Držím se pravidla: *K vyjádření myšlenky mi stáčí 50% slov, kterých jsem použil poprvé.*

2.3.2 Hierarchie stránek

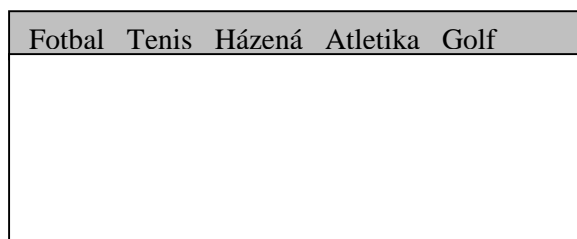
Má-li být stránka rychle a snadno pochopitelná, je důležité, aby vzhled věcí na stránce – všechny vizuální podněty – jasně a přesně zobrazovaly vztah mezi objekty na stránce: co spolu souvisí a co je součástí něčeho jiného. Jinými slovy každá stránka by měla mít jasnou vizuální hierarchii.

Tři základní vlastnosti:

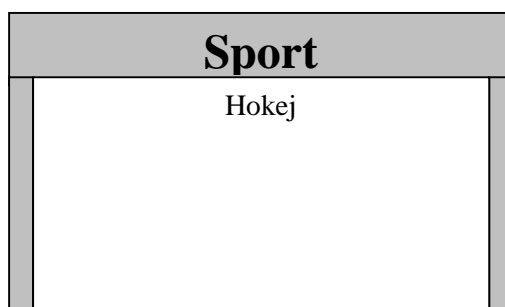
- čím je něco důležitější, tím je to nápadnější (Například nejdůležitější nadpisy jsou buď větší, tučnější, mají jinou barvu, jsou odděleny větší mezerou anebo se blíží k hornímu okraji stránky – případně různé kombinace)



- související objekty jsou propojeny také vizuálně (že spolu nějaké objekty souvisí můžeme demonstrovat například tím, že je seskupíme pod jeden titulek. Nebo k jejich zobrazení použijeme podobný vizuální styl anebo je všechny umístíme do vymezené oblasti)



- objekty jsou vizuálně „zanořeny“, aby bylo zřejmé, co je částí čeho (Například nadpis sekce „Sport“ by se měl objevit nad názvem konkrétního sportu, čímž vizuálně obsáhne celou stránku, protože konkrétní sport je součástí dané sekce.)



2.3.3 Logo

Při pohledu na jakoukoli internetovou stránku by nám mělo být hned zřejmé, kde se to nacházíme. K tomuto účelu slouží logo. První uživatelův pohled by měl padnout právě na logo. Nejčastěji (a i já se v systému držím tohoto pravidla) bývá umístěno v levém horním rohu. Logo může být doprovázeno krátkým a jednoduchým sloganem.

Logo by nemělo být příliš složité. Mělo by být především snadno zapamatovatelné. Webové logo je synonymem loga televizního. Když přepínáme televizní kanály chceme okamžitě vědět, na které stanici se nacházíme. Nemáme zájem hledat kde to jsme. A stejná situace platí i při „brouzdání“ po internetu.

V informačním systému internetového autobazaru použijí logo i slogan a to v rozvržení podle následujícího obrázku:

Logo	Slogan

2.3.4 Odkazy

Odkazy jsou křižovatkami na cestě webovou stránkou – proto je důležité vědět, jestli se dále dostanu „tudy“ nebo „tam tudy“. Měly by být na první pohled zřejmé (přemýšlení typu: „A můžu na to kliknout? Je to odkaz?“ jen zbytečně znepříjemňují uživateli život).

Jako odkaz můžeme použít prakticky cokoli na webové stránce (část textu, obrázek, tlačítko, atd.). Jak už jsem napsal v předchozím odstavci mělo by být na první pohled jasné na co můžeme kliknout (např. na stránce se nachází obrázek lidské postavy, ale my pomocí „klikací mapy“ vytvoříme odkaz je při kliknutí na nos člověka. Kdo z uživatelů tento odkaz objeví? Myslím jsi, že téměř nikdo a když už tak jen náhodou.).

Na odkaz se dá snadno upozornit např. těmito způsoby:

- Použitím zařítého prostředku, že odkaz je uveden v jiné barvě a podtržen.
- Před odkazem uvedeme text „Sem klikni.“ nebo jiný text stejného významu.
- Použití tlačítka jako odkazu.
- ...

2.3.5 Navigace

Uživatelé při „brouzdání“ internetem nevybírají nejlepší alternativu, ale pouze první rozumnou volbu. Neprovádí optimální výběr nýbrž dělají kompromisy.

Proč by měli internetoví uživatelé hledat nejlepší volbu, když ...

- obvykle spěchají,
- za špatný odhad následuje jen malý trest,
- zvažování voleb možná nezvýší naše šance (zejména na špatně navržených stránkách),
- hledání je zábavnější.

Samozřejmě tím nechci tvrdit, že uživatelé nikdy nezvažují různé alternativy, než klepnou na nějaký odkaz. Vždy bude záležet na jejich náladě, nakolik budou tlačeni časem a nakolik důvěry budou vkládat v příslušné stránky.

Znáte to z vaší vlastní zkušenosti webového uživatele. Pokud přijdete na nějaké stránky a nemůžete najít to, co hledáte, případně určit, jakým způsobem jsou stránky organizovány, pravděpodobně zde moc dlouho nezůstanete – a nikdy se už nevrátíte. Je tedy nutné vytvořit jasnou, jednoduchou a konzistentní navigaci.

Při vstupu na webové stránky procházíte podobným procesem jako v supermarketu. Zpravidla se snažíte něco najít a rozhodujete se, zda se máte nejprve zeptat a nebo začít hledat. Na webové stránce samozřejmě nepostává nikdo, kdo by vám poradil, kde hledat. Internetovým ekvivalentem dotazování na směr je vyhledávací políčko.

Lidem, kteří se rozhodnou použít vyhledávací políčko to musí být umožněno. Vyhledávací políčko tedy musí každá dobrá webová stránka obsahovat a to v takové pozici, která je dobře viditelná.

Naopak lidem, kteří se raději rozhodnou vyhledávat „na vlastní triko“ musí webová stránka tuto činnost, co nejvíce usnadnit. Musí obsahovat přehledné a srozumitelné sekce respektive podsekce.

Dále by měla každá stránka webového systému obsahovat odkaz na domovskou stránku. V informačním systému internetového autobazaru jsem to vyřešil způsobem, že v hlavním menu se nachází tlačítko „Úvod“.

3 Přístupnost webových stránek

Elektronicky publikované dokumenty a především ty, které jsou určeny pro internet se od klasických tj. tištěných v mnoha směrech liší. Jejich elektronická podoba nabízí mnoho výhod, které u tištěných dokumentů nejsou možné. Pro příklad bych uvedl menší finanční náročnost na pořízení knihy, skladnost, možnost používání odkazů na další zdroje, jednoduché vyhledávání klíčových slov i slovních spojení a mnohé další. Existují však některé nevýhody.

Až 30% uživatelů systému je nějakým způsobem omezeno. Existují specifické skupiny uživatelů, které mají i specifické potřeby. V praxi to jsou většinou zrakově a sluchově postižení, případně uživatelé se sníženou hybností rukou nebo poruchami soustředění. Tito uživatelé využívají pomocné technologie, například hlasové výstupy, které informace z WWW stránek zprostředkovávají. Aby tyto pomůcky fungovaly, je třeba, aby webová stránka splňovala kritéria přístupné webové stránky. Správně přístupná webová stránka navíc neslouží pouze zdravotně postiženým, ale i lidem využívajícím méně obvyklá zobrazovací zařízení, operační systémy nebo softwarové vybavení.

Uživatelé, kteří jsou nějakým způsobem omezeni patří sice do skupiny minoritních uživatelů, ale přesto je nutné systém vytvořit s ohledem na jejich problémy. Z tohoto důvodu musí informační systém internetového autobazaru splňovat i tzv. podmínky přístupné webové stránky.

Mezi minoritní uživatele patří:

- majitelé zastaralých počítačů
- dyslektici (nedokáží přečíst dlouhé a špatně strukturované texty)
- barvoslepí (špatně vnímají slabé kontrasty,)
- tělesně postižení (např. nemohou používat myš)
- a jiní

Pravidla pro přístupnost webových stránek vydávají různé organizace. Mezi nejznámější patří mezinárodní organizace W3C - Web Content Accessibility Guidelines 1.0 (WCAG) nebo české organizace SONS – Blind friendly web a MIČR – Pravidla pro tvorbu přístupného webu.

Při návrhu informačního systému internetového autobazaru jsem vycházel z dokumentu Ministerstva informatiky České republiky, který obsahuje doporučení pro tvorbu WWW stránek ve veřejné správě a stanovuje podmínky pro zveřejňování informací dálkovým přístupem ve shodě s takzvanými pravidly WAI (Web Accessibility Initiative, Iniciativa pro bezbariérový přístup). Pravidla mají doporučující charakter, jsou vodítkem pro zadavatele webových stránek a inspirací pro jejich správce.

3.1 Vlastnosti přístupných webových stránek

3.1.1 Obsah webových stránek je dostupný a čitelný

Každý netextový prvek nesoucí významné sdělení má svou textovou alternativu:

V případě systému internetového autobazaru se jedná především o obrázky a fotografie, které musí mít ve značce `` povinně obsažen atribut `alt`. Tento atribut je alternativním popisem pro případ, že se obrázek z nějakého důvodu v prohlížeči nezobrazí (např. pro starší prohlížeče, pro špatné technické vybavení počítače, ...).

V případě, že je textová informace pro použití v rámci atributu `alt` příliš rozsáhlá je přítomen navíc atribut `longdesc`, který odkazuje na speciální www stránku, která obsahuje kompletní popis daného obrázku.

Informace sdělované barvou jsou dostupné i bez barevného rozlišení:

Jestliže uživatel z nějakého důvodu nedokáže správně rozpoznat barvy, všechny informace uvedené na webové stránce mu zůstávají srozumitelné.

Webová stránka tedy neobsahuje např. odkazy odlišené od okolního textu pouze jinou barvou nebo položky formuláře, které musí být povinně vyplněny zobrazeny červeně a nepovinné položky zobrazeny modře.

Všechny informace uvedené na webové stránce musí být rovněž dostupné i při zapnutí funkci „Vysoký kontrast“, kterou si uživatel může aktivovat v prostředí operačního systému MS Windows.

Barvy popředí a pozadí jsou dostatečně kontrastní. Na pozadí není vzorek, který snižuje čitelnost:

Kombinace barvy nebo vzorku barvy pozadí a barvy popředí (zejména se jedná o písmo) musí tvořit dostatečný vzájemný kontrast proto, aby uživatelé, kteří mají zhoršený zrak popředí a pozadí nesplývalo.

Kontrast ve vytvářeném informačním systému budu kontrolovat pomocí dostupného on-line nástroje <http://www.sovavsiti.cz/kontrast/>. Minimální hodnota jasu pro rozdíl jasu je 125 bodů (maximum této hodnoty je 255 bodů) a pro rozdíl barev 500 bodů (maximum této hodnoty je 765 bodů), čím jsou tyto rozdíly větší, tím je kombinace barev více kontrastní a tudíž je i popředí na pozadí lépe čitelné.

3.1.2 Práci s webovou stránku řídí uživatel

Obsah www stránky se mění jen když uživatel aktivuje nějaký prvek:

Obsah stránky se nezmění pokud uživatel neprovede akci, která je k takové změně určena (např. kliknutí na odkaz, odeslání formuláře,...).

Kliknutí na prvek, který není odkazem, nebo pouhá změna formulářového prvku smí ovlivnit pouze stávající obsah webové stránky. Nesmí způsobit žádnou větší změnu (např. přesměrování na jinou stránku).

Mezi povolené změny patří i změna textu po najetí myši na daný text. V informačním systému internetového autobazaru toho budu využívat pro podtržení odkazů při najetí myši nad příslušné odkazy.

Webová stránka bez přímého příkazu uživatele nemanipuluje uživatelským prostředím:

Nic vně samotné webové stránky stránce nepatří. Toto je uživatelské prostředí a pouze uživatel má pravomoc s ním manipulovat. WWW stránka proto nesmí bez příkazu uživatele otvírat nová okna, jakkoli měnit ovládací prvky prohlížeče (např. nastavovat domovskou stránku), měnit velikost oken atd.

Toto je možné pouze pro předchozím zřetelném upozornění uživatele, který musí danou změnu odsouhlasit.

Nová okna se otevírají jen v odůvodněných případech:

Otevření nového okna by mělo být ve většině případu na volbě uživatele, existují však výjimky. V informačním systému internetového autobazaru bude využívat otevírání nových oken při kliknutí na odkaz na stránku automobilek, nebo v diskusním fóru při přidávání nových příspěvků.

Na webové stránce nic neblinká rychleji než jednou za sekundu:

Trvalé (více jak třikrát po sobě) blikání, pulzování a kmitání nesmí být s frekvencí větší než 1 Hz. Tyto periodické změny mohou způsobit potíže osobám s jistými psychickými obtížemi. Značka *blink* nesmí být použita vůbec, protože u ní nelze nastavit frekvenci blikání.

3.1.3 Informace jsou srozumitelné a přehledné

Tato kapitola byla již z velké části popsána v kapitole 2.3 Analýza stránek z pohledu návrháře. Pro doplnění uvedu jen následující pravidlo.

Na samostatné webové stránce je uveden kontakt na technického správce systému (autora systému). Na tuto stránku odkazuje každá stránka systému.

Na každé stránce informačního systému internetového autobazaru je uveden kontakt (email) na autora tohoto systému.

3.1.4 Odkazy jsou zřetelné

Tato část byla již popsána v kapitole 2.3 Analýza stránek z pohledu návrháře.

3.15 Ovládání webové stránky je jasné a přehledné

Tato kapitola byla již z velké části popsána v kapitole 2.3 Analýza stránek z pohledu návrháře. Pro doplnění uvedu jen následující pravidlo.

Každá webová stránka má smysluplný název, vystihující její obsah:

Především se jedná o správně zvolený nadpis stránky v hlavičce dokumentu mezi značkami `<title>` `</title>`.

3.1.6 Kód je technický způsobilý a strukturovaný

Kód webových stránek odpovídá nějaké zveřejněné finální specifikaci jazyka HTML nebo XHTML. Neobsahuje syntaktické chyby, které je správce webové stránky schopen odstranit:

Webové stránky jsou vytvořeny kódem nějaké mezinárodně uznávané verze značkovacího jazyka HTML nebo XHTML. Ve vytvářeném systému to bude jazyk HTML. Celý kód www stránky musí být bezchybný a korektní.

Jako vodítko pro kontrolu bezchybnosti použijí softwarový validátor <http://www.validator.w3.org/>. Výstup takového to programu je však pouze orientační a nelze jej považovat za 100% potvrzení správnosti.

V metaznačkách je uvedena použitá znaková sada dokumentu:

Aby se při uložení webové stránky na lokální disk neztratila informace o kódování stránky a stránka zůstala čitelná, musí být uvnitř sekce `<head>``</head>` uveden textový ekvivalent http hlavičky *Content-type*, specifikující použitou znakovou sadu.

Tato informace je zapsána jako metaznačka ve tvaru:

```
<meta http-equiv="content-type" content="text/html; charset=ANSI-KÓD" >
```

4 Návrh databáze autobazaru

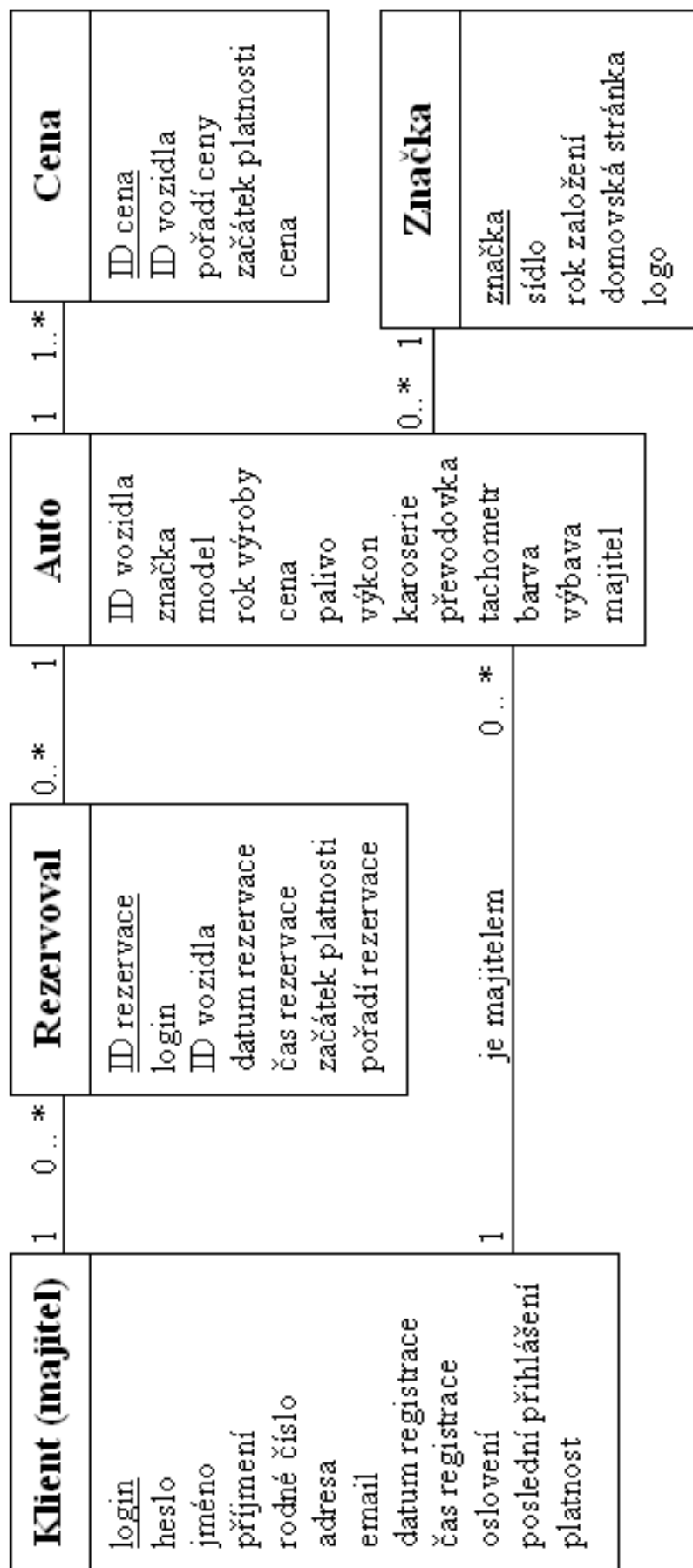
4.1 Vytvoření ER diagramu

Entity Relationship Diagram (ERD) slouží k modelování dat aplikační domény a jejich vztahů „v klidu“. ERD je síťový model popisující návrh uložených dat v systému na vyšší úrovni abstrakce. ERD modeluje data, která potřebujeme v systému uchovávat a vztahy mezi těmito daty.

Základní pojmy:

- *Entita* je „věc“ (objekt) reálného světa. Každá entita musí být rozlišitelná od jiných entit (= jedinečná). Např. konkrétní klient autobazaru (jednotlivé klienty lze od sebe rozlišit).
- *Entitní množina* je množina entit téhož typu, které sdílí stejné vlastnosti (atributy). Entitní množina je zachycena v ERD. Např. Klient
- *Vztah* vyjadřuje asociaci mezi entitami, tj. situaci, kdy dvě (nebo více) entity spolu logicky souvisejí. Např. klient autobazaru má rezervovaný automobil.
- *Vztahová množina* je množina vztahů téhož typu, které sdílí tytéž vlastnosti. V návaznosti na předchozí příklady existují entitní množiny Klient a Automobil, mezi nimi existující vztahová množina rezervovaný vůz.
- *Atribut* je vlastnost entity, která nás v kontextu daného problému zajímá. Atribut je definován jako prvek entitní množiny. Např. U entitní množiny Klient nás zajímá jméno klienta, příjmení, atd.
- *Kardinalita vztahu* je maximální počet vztahů daného typu (vztahové množiny), ve kterých může participovat jedna entita. Např. Klient autobazaru může mít rezervovaných více automobilů (obecně nekonečně mnoho).
- *Členství vztahu* je minimální počet vztahů daného typu (vztahové množiny), ve kterých musí participovat jedna entita. Typické hodnoty, které se u členství objevují jsou: 1 (povinné) a 0 (volitelné). Např. Klient autobazaru nemusí mít rezervovaný žádný automobil.

Ze základních pojmů je to vše. Nebudu se zde zabývat žádným detailnějším popisem vytváření ERD (to by bylo na další bakalářskou práci) a raději přistoupím k samotnému zobrazení ERD. ERD informačního systému internetového autobazaru je na následující straně:



Tabulky použité v ER diagramu, ale nejsou jediné tabulky použité v databázi informačního systému internetového autobazaru. Dále je potřeba vytvořit tabulku pro uložení komentářů od uživatelů, pro uložení informací o správci systému, pro uložení příspěvků z diskuse a pro uložení souborů.

4.2 Tvorba databáze a tabulek

V této kapitole nejdříve pár slovy popíši obecné informace o databázích a poté se budu věnovat vytvořením základních struktur MySQL: databázi a tabulek. Údaje, které musí systém uchovávat jsou uvedeny v kapitole 1.1.2 (Požadavky na systém). Databáze se skládá celkem z devíti tabulek.

4.2.1 Co je to databáze?

Databázi si můžeme představit jako prostor, do kterého se ukládají všechny potřebné údaje. Zpracováním a přístupem údajů v databázi bývá pověřen program, kterému se říká DBMS (DataBase Management System), nebo česky SŘBD (Systém Řízení Báže Dat). Celé je to tedy zařízeno tak, že veškeré zpracování, ukládání a získávání dat je prováděno DBMS programem. Jakákoli aplikace, která databázi používá vždy přistupuje přes tento DBMS program.

Mezi DBMS patří řada programů v nejrůznějších cenových kategoriích. Většina profesionálních a komerčních DBMS nejsou programy nikterak levné a jejich cena se pohybuje ve velmi vysokých částkách. Naštěstí existují v kategorii DBMS i programy, které jsou v nízké cenové relaci, nebo zcela zdarma. Představitelem těchto DBMS systémů jsou například MySQL, PostgreSQL, nebo Firebird. V informačním systému internetového autobazaru je použito MySQL.

Naprostá většina dnešních DBMS staví na tzv. relačním modelu dat. V tomto modelu jsou veškerá data uspořádána do databázových tabulek. Každá tabulka zpravidla uspořádává údaje o určitém objektu. Databázová tabulka je uspořádána do řádků a sloupců. Například budeme-li mít tabulku se seznamem osob, bude každý řádek odpovídat jedné osobě. Sloupce pak obsahují konkrétní údaje o osobách, v našem případě by to třeba mohlo být: Jméno, Rodné číslo, Adresa, Telefon.

4.2.2 Co je to SQL?

Jak už jsem zmínil, přístup k údajům uloženým v databázi obstarává DBMS program. Aby mohly jiné aplikace používat také údaje z databáze, musí DBMS program nabízet rozhraní, pomocí kterého budou moci ostatní aplikace s DBMS spolupracovat.

Při práci s DBMS programem se používá osvědčený model klient/server. V roli serveru vystupuje DBMS program nabízející své služby. DBMS program je tak nepřetržitě spuštěn a čeká na požadavky od klientů - jiných aplikací. Na tyto požadavky pak odpovídá a reaguje na ně. Protože DBMS program vystupuje v roli serveru, říká se mu také databázový server.

Pro zápis požadavků na databázový server se dnes nejčastěji používá SQL jazyk. SQL je anglická zkratka znamenající Structured Query Language. Tento jazyk dnes používá naprostá většina databázových serverů. Proto se takovým serverům říká zjednodušeně SQL servery.. Samotný SQL obsahuje vše potřebné pro práci s databázemi - ať už příkazy pro práci na vytvoření, rušení, modifikování tabulky, tak i pro práci s údaji samotnými - přidávání, změnu, rušení a vyhledávání údajů.

4.2.3 Vytvoření databáze pomocí SQL skriptu

Při vytváření databáze musíme po zhotovení návrhu samozřejmě nejprve sdělit systému MySql, že chceme vytvořit novou databázi. K tomu slouží příkaz SQL *create database*:

```
create database autobazar;
```

Zda se databáze opravdu vytvořila si můžeme ověřit následujícím příkazem:

```
show database;
```

Nyní by se mezi databázemi v systému měla objevit databáze *autobazar*. Takže máme prázdnou databázi, která čeká na vytvoření nějakých tabulek.

V případě, že databáze s názvem *autobazar* již existuje je potřeba před příkaz *create* přidat následující příkaz, který v případě existence dané databáze ji nejdříve vymaže:

```
drop database if exists autobazar;
```

Abychom mohli vytvořit nějaké tabulky nebo jinak pracovat s databází *autobazar*, musíme nejprve informovat MySql o tom, že chceme pracovat s naší novou databází. K tomu se používá příkaz *use*:

```
use autobazar;
```

Tímto jsme vybrali databázi *autobazar*, což znamená, že se nyní všechny příkazy, není-li zadáno jinak, aplikují na tuto databázi.

4.2.4 Použití fotografií v MySQL

Zabývám se zde základními principy práce s obrázky v MySQL. V databázi internetového autobazaru potřebujeme kromě základních údajů o automobilech ukládat i fotografii nabízeného vozidla (Kdo z nás by si rezervoval vozidlo aniž by ho předtím viděl?).

Spojení správné fotografie a dané položky lze dosáhnout v podstatě dvěma způsoby. Prvním způsobem je uložení obrázku do pole typu BLOB tabulky databáze. Je třeba vytvořit novou tabulku *obrázky*, do které uložíme obrázky jako binární soubory. A tuto tabulku propojit s tabulkou, v níž jsou uloženy textové informace.

U druhého způsobu je v příslušném poli tabulky adresa obrázku uloženého na serveru, obrázků se tedy načte jednoduchým dotazem na databázi.

Tento způsob má nespornou výhodu v minimálním zatížení databáze, nevýhodou však je komplikované zajištění samotného nahrání obrázku na server, například chceme-li jako autor aplikace internetového obchodu správu této aplikace včetně editace obrázků k položkám zboží, předat přímo majiteli obchodu.

Vzhledem k tomu, že v informačním systému internetového autobazaru potřebujeme obrázky editovat (editaci provádí správce přímo v systému) zvolil jsem použití druhého způsobu.

4.2.5 Vytvoření tabulek pomocí SQL skriptu

K tvorbě tabulek v databázi slouží příkaz SQL *create table*. Obecný tvar tabulky je následující:

```
create table nazev_tabulky ( definice tabulky ) [type = typ_tabulky];
```

Po slovech *create table* následuje název tabulky a množina definic sloupců. Na konci příkazu můžeme volitelně určovat jaký typ tabulky chceme vytvořit.

Definice tabulky:

Následující obecná ukázka definice tabulky je nejjednodušší ne však jedinou možností jak tabulky definovat.

```
název_sloupce typ [NOT NULL | NULL] [DEFAULT výchozí_hodnota]  
[AUTO_INCREMENT]  
[PRIMARY KEY] [seznam_sloupců]
```

U každé deklarace sloupce můžeme uvést následující volby:

- Každý sloupec je možné deklarovat jako NULL nebo NOT NULL, což znamená, že sloupec buď může nebo nesmí obsahovat hodnoty NULL. Standardně hodnoty mohou obsahovat hodnoty NULL.
- Pomocí klíčového slova DEFAULT a požadované výchozí hodnoty můžeme deklarovat výchozí hodnotu sloupce.
- Klíčové slovo AUTO_INCREMENT slouží ke generování pořadového čísla. Automaticky vygenerovaná hodnota bude o 1 větší než současná největší hodnota v tabulce. Řádek vložený jako první bude mít pořadové číslo 1. V tabulce může být pouze jeden sloupec AUTO_INCREMENT a ten musí být indexován.
- Lze deklarovat, že určitý sloupec je primárním klíčem (PRIMARY KEY) tabulky.
- S pomocí klauzule REFERENCES lze, definovat, že určitý sloupec představuje cizí klíč.

Tabulky vytvořené v kapitole 2.1 (Tvorba databáze a tabulek) jsou do SQL skriptu převedeny v příloze 1. Vytvoření databázových tabulek.

5 Návrh systému autobazaru

Informační systém internetového autobazaru si rozdělíme na 4 základní části, které si probereme postupně. Části si nazvěme **základ** (pro nepřihlášené uživatele), **uživatel** (pro přihlášené uživatele), **správce** (editační systém autobazaru po přihlášení správce systému) a **diskuse** (diskusní fórum autobazaru).

V této kapitole proberu činnosti, které mohou uživatelé (klienti) autobazaru resp. správce systému v jednotlivých částech provádět a dále pak automatické činnosti informačního systému internetového autobazaru v jednotlivých částích

5.1 Základ

Jedná se o úvodní část systému (bez přihlášení uživatele i správce).

V této části uživatelé mohou:

- prohlížet katalog nabízených automobilů
- vyhledávat vozidla pomocí vyhledávacího formuláře
- zasílat autobazaru komentáře
- přihlásit se
- nechat si zaslat na email uvedený při registraci zapomenuté heslo
- registrovat se

Automatické činnosti systému v části základ:

- kontrola správného vyplnění formuláře pro zasílání komentářů autobazaru (nevyplněná povinná pole)
- kontrola správného vyplnění formuláře pro registraci klientů (nevyplněná povinná pole, snaha o registraci již zaregistrovaného uživatelského jména včetně přihlašovacího jména správce,...)
- po registraci zaslání emailu na zadaný email
- zamezení přihlášení dokud na uživatel neklikne na příslušný odkaz zasláný v poslaném emailu (K tomu slouží sloupec *platnost* v databázi MySQL v tabulce *auto*. Po registraci je jeho hodnota nastavena na hodnotu 0, kliknutím na příslušný odkaz se jeho hodnota změní na hodnotu 1. Přihlášení je možné pouze tehdy, když se hodnota sloupce nerovná 0)
- zaslání zapomenutého hesla na daný email
- kontrola správných přihlašovacích údajů (přihlašovacího jména a hesla) a případné přihlášení do části uživatel resp. správce

5.2 Uživatel

Jedná se o část systému po přihlášení klienta autobazaru.

V této části uživatelé mohou:

- prohlížet katalog nabízených automobilů
- vyhledávat vozidla pomocí vyhledávacího formuláře
- zasílat autobazaru komentáře
- odhlásit se
- vstoupit do diskusního fóra
- rezervovat vozidla
- rušit své rezervace
- měnit své osobní údaje
- nastavovat oslovení

Automatické činnosti v části uživatel:

- kontrola správného vyplnění formuláře pro zasílání komentářů autobazaru (nevyplněná povinná pole)
- kontrola správného vyplnění formuláře pro nastavení osobních údajů klienta a oslovení (nevyplněná povinná pole)
- zamezení rezervace vozidla klientem, který již má dané vozidlo rezervované
- při zrušení rezervace vozidla předání platnosti rezervace dalšímu klientu v pořadí rezervací a zaslání emailu příslušnému klientu o začátku platnosti jeho rezervace
- odpočítávání počtu dní k vyzvednutí vozidla v případě platné rezervace
- spočítání celkového počtu rezervací vozidla
- spočítání pořadí rezervace automobilu u jednotlivých klientů
- automatické odhlášení po 10min nečinnosti

5.3 Správce

Jedná se o část systému po přihlášení správce systému.

V této části správce systému může:

- editovat (přidat, odstranit, vymazat) značky vozidel
- editovat majitele vozidel
- editovat klienty autobazaru

- odstraňovat komentáře od uživatelů
- měnit přihlašovací údaje správce
- editovat nabízené automobily
- rušit rezervace vozidel od klientů
- zadat novou cenu vozidla
- odstranit cenu vozidla
- vkládat a odstraňovat prezentace
- vstoupit do diskusního fóra
- odhlásit se
- přepínat barvy pozadí

Automatické činnosti v části správce:

- kontrola správného vyplnění formulářů při editaci značek vozidel (nevyplněná povinná pole, rok založení automobilky musí být menší než letošní rok), včetně zadání cesty k souboru při vkládání loga značky a zamezení vložení více než jednoho loga
 - kontrola správného vyplnění formulářů při editaci automobilů (nevyplněná povinná pole, ...), včetně zadání cesty k souboru při vkládání fotografie automobilu (možnost vložení libovolného počtu fotografií vozidla)
 - kontrola správného vyplnění formuláře pro změnu údajů správce (nevyplněná povinná pole, snaha o změnu přihlašovacího jména na přihlašovací jméno, které již má zaregistrovaný nějaký uživatel,...)
 - kontrola správného vyplnění formulářů při editaci bývalých majitelů a uživatelů (nevyplněná povinná pole)
 - označení komentářů od uživatelů, které byly přidány od posledního prohlížení komentářů správcem systému (komentáře jsou označeny následujícím obrázkem)
- NEW**
- nutnost zadat cestu k souboru při vkládání prezentací autobazaru
 - kontrola platných rezervací po přihlášení správce systému (rezervace jímž vypršela lhůta vyzvednutí budou zrušeny – daným klientům zaslány emaily o zrušení, a klientů s nově platnou rezervací systém zašle emaily o začátku platnosti jejich rezervace)

5.4 Diskuse

Jedná se o část systému, do které vstoupí přihlášení uživatelé resp. správce systému kliknutím na příslušný odkaz.

V této části uživatelé mohou:

- prohlížet příspěvky v jednotlivých diskusních skupinách
- řadit příspěvky ve skupinách od nejnovějších resp. nejstarších
- přidávat příspěvky do jednotlivých skupin
- reagovat na dané příspěvky
- odejít z diskusního fóra

V této části správce systému může:

- prohlížet příspěvky v jednotlivých diskusních skupinách
- řadit příspěvky ve skupinách od nejnovějších resp. nejstarších
- přidávat příspěvky do jednotlivých skupin
- reagovat na dané příspěvky
- odstraňovat příspěvky
- odejít z diskusního fóra

Automatické činnosti v části diskuse:

- spočítá celkový počet příspěvku v diskusním fóru a v jednotlivých skupinách
- při vložení reakce na příspěvek se reakce ve fóru zobrazí spolu s příspěvkem, na který klient nebo správce reaguje
- zobrazení datumu a času příspěvku
- zobrazení autora příspěvku
- možnosti činností ve fóru podle toho zda jsme vstoupili jako správce systému nebo klient autobazaru (správce systému má pravomoc kromě přidávání nových příspěvků a reakcí na příspěvky i příspěvky odstraňovat)

6 Použité prostředky a jazyky

V poslední kapitole popíši prostředky a jazyky, které jsem použil při vývoji informačního systému internetového autobazaru. Nezabývám se zde, co jsem k čemu použil, ale věnuji se především historii jazyka, jeho vývoji, jeho obecnému popisu, výhodám resp. nevýhodám jazyka, atd.

Jedná se o následující jazyky a prostředky:

- HTML
- CSS
- PHP
- Javascript
- MySql

6.1 HTML

HTML je zkratka z anglického *HyperText Markup Language*, značkovací jazyk pro hypertext. Je jedním z jazyků pro vytváření stránek v systému World Wide Web, který umožňuje publikaci stránek na internetu.

Jazyk je podmnožinou dříve vyvinutého rozsáhlého univerzálního značkovacího jazyka SGML (*Standard Generalized Markup Language*). Vývoj HTML byl ovlivněn vývojem webových prohlížečů, které zpětně ovlivňovaly definici jazyka.

6.1.1 Vývoj jazyka

V roce 1989 spolupracovali Tim Berners-Lee a Robert Caillau na propojeném informačním systému pro CERN, výzkumné centrum fyziky poblíž Ženevy ve Švýcarsku. V té době se pro tvorbu dokumentů obvykle používal TeX, Postscript a také SGML. Berners-Lee si uvědomoval, že potřebují něco jednoduššího a v roce 1990 byl tedy navržen jazyk HTML a protokol pro jeho přenos v síti - HTTP (*HyperText Transfer Protocol - přenosový protokol hypertextu*).

V roce 1991 CERN zprovoznil svoji webovou stránku. Současně NCSA (*National Center for Supercomputer Applications*) vyvíjelo Marca Andreessena a Erica Binu k vyvinutí prohlížeče Mosaic. Byl vyvinut v roce 1993 pro počítače PC a Macintosh a měl obrovský úspěch. Byl to první prohlížeč s grafickým uživatelským rozhraním. Došlo k velkému rozvoji webových stránek a bylo nutné pro HTML definovat standardy.

6.1.2 Verze jazyka

- Verze 2.0 zachycuje stav jazyka v polovině roku 1994. Standard vydala komunita IETF (*Internet Engineering Task Force*). Je to první verze, která odpovídá syntaxi SGML. Přidává k původní specifikaci interaktivní formuláře.
- Verze 3.2 byla vydána v květnu 1996 a zachycuje stav jazyka v tomto roce. Připravovaná verze HTML 3.0 nebyla nikdy přijata jako standard, protože byla příliš složitá a žádná firma nebyla schopna naprogramovat její podporu ve svém prohlížeči. Standard už vydalo konsorcium W3C, stejně jako následující verze. Přidává k jazyku tabulky, zarovnávání textu a stylové elementy pro ovlivňování vzhledu.
- Verze 4.0 byla vydána v roce 1997. Do specifikace jazyka přidala nové prvky pro tvorbu tabulek, formulářů a nově byly standardizovány rámy (*frames*). Tato verze se snaží dosáhnout původního účelu - prvky by měly vyznačovat *význam* (sémantiku) jednotlivých částí dokumentu, vzhled má být ovlivňován připojovanými styly. Některé prezentační elementy byly zavrženy.
- Verze 4.01 opravuje některé chyby verze předchozí a přidává některé nové značky. Je to poslední verze HTML, které se již dále nevyvíjí, protože je nahrazeno novějším XHTML, jehož základem je právě tato poslední verze HTML.

6.2 CSS

CSS je zkratka pro anglický název Cascading Style Sheets, česky tabulky kaskádových stylů. Je to jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML.

Jazyk byl navržen standardizační organizací W3C. Byly vydány zatím dvě verze specifikace CSS1 a CSS2 (plus CSS 2.1), pracuje se na verzi CSS3.

Hlavním smyslem je umožnit návrhářům oddělit vzhled dokumentu od jeho struktury a obsahu. Původně to měl umožnit už jazyk HTML, ale v důsledku nedostatečných standardů a konkurenčního boje výrobců prohlížečů se vyvinul jinak. Starší verze HTML obsahují celou řadu elementů, které nepopisují obsah a strukturu dokumentu, ale i způsob jeho zobrazení. Z hlediska zpracování dokumentů a vyhledávání informací není takový vývoj žádoucí.

Definice stylů může být vložena do stránky, nebo může být použita pro více stránek a definuje tak vzhled (styl) všech značek ve stránce, nebo všech stránek na serveru. Takhle můžeme jednoduše, rychle a efektivně měnit vzhled stránky. Pomocí CSS je možné měnit

cokoliv od velikosti, stylu a barvy textu po mezery mezi znaky a řádky, okraje a mezery kolem prvků, přesné umístění na stránce.

6.2.1 Výhody

Používání kaskádových stylů ve srovnání se samotným HTML v praxi přináší výhody:

- širší formátovací možnosti
- snadná tvorba a údržba konzistentního stylu
- oddělení struktury a stylu
- dynamická práce se styly
- formátování XML dokumentů
- větší kompatibilita alternativních webových prohlížečů
- kratší doba načítání stránky

Výhodou CSS oproti starému formátování v HTML je, že kód a obsah webových stránek je uložen v souboru html a veškerý design a formátování se načítá z jednoho souboru .css, který je většinou společný pro všechny webové stránky. To znamená, že pokud máte v plánu změnu designu webových stránek, stačí změnit pouze jeden soubor css a změna se aplikuje na všechny webové stránky.

Mohou také existovat různé styly pro různá výstupní zařízení. Návrhář webových stránek má tak možnost, prostřednictvím CSS stylů, dokumentu určit, jak bude vypadat na papíře, při projekci či na PDA apod. Specifikace CSS nezapomínají dokonce ani na zrakově postižené - je možno napsat styly pro hlasový syntetizátor nebo hmatovou čtečku Braillova písma.

6.2.2 Nevýhody

Hlavní nevýhodou CSS je zatím stále špatná podpora v majoritních prohlížečích. Různé prohlížeče interpretují stejný CSS kód jinak a je někdy velmi obtížné jej napsat tak, aby se na všech (resp. na několika vybraných) prohlížečích výsledek zobrazil stejně. Situace se ale v roce 2006 značně zlepšuje, v souvislosti s příchodem Internet Exploreru 7, který by měl postupně vytlačit svého předchůdce IE 6, který byl častým zdrojem problémů.

6.3 PHP

PHP (rekurzivní zkratka PHP: Hypertext Preprocessor, „PHP: Hypertextový preprocesor“, původně Personal Home Page) je skriptovací programovací jazyk, určený především pro

programování dynamických internetových stránek. Nejčastěji se začleňuje přímo do struktury jazyka HTML, což je velmi výhodné pro tvorbu webových aplikací.

PHP skripty jsou prováděny na straně serveru, k uživateli je přenášen až výsledek jejich činnosti. Syntaxe jazyka kombinuje hned několik programovacích jazyků (Perl, C, Pascal a Java). PHP je nezávislý na platformě, skripty fungují bez úprav na mnoha různých operačních systémech. Obsahuje rozsáhlé knihovny funkcí pro zpracování textu, grafiky, práci se soubory, přístup k většině databázových serverů (mj. MySQL, ODBC, Oracle, PostgreSQL), podporu celé řady internetových protokolů (HTTP, SMTP, SNMP, FTP, IMAP, POP3, LDAP, ...).

PHP se stalo velmi oblíbeným především díky jednoduchosti použití a tomu, že kombinuje vlastnosti více programovacích jazyků a nechává tak vývojáři částečnou svobodu v syntaxi. V kombinaci s databázovým serverem (především s MySQL nebo PostgreSQL) a webovým serverem Apache je často využíván k tvorbě webových aplikací.

S verzí PHP 5 se výrazně zlepšil přístup k objektově orientovanému programování. Přístup je velmi podobný jako v jazyce Java.

Od roku 1994 je PHP jedním z nejpoužívanějších způsobů tvorby dynamicky generovaných www stránek. Jeho tvůrce (Rasmus Lerdorf) jej vytvořil pro svou osobní potřebu přepsáním z Perlu do jazyka C.

6.4 Javascript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk, jehož autorem je Brendan Eich z tehdejší společnosti Netscape.

Nyní se zpravidla používá jako interpretovaný programovací jazyk pro www stránky, vkládaný přímo do HTML kódu stránky. Jsou jím obvykle ovládány různé interaktivní prvky GUI (tlačítka, textová políčka) nebo tvořeny animace a efekty obrázků.

Jeho syntaxe patří do rodiny jazyků C/C++ /Java. Slovo Java je však součástí jeho názvu pouze s marketingových důvodů a s programovacím jazykem Java jej vedle názvu spojuje jen podobná syntaxe. JavaScript byl v červenci 1997 standardizován asociací ECMA (European Computer Manufacturers Association) a v srpnu 1998 ISO (International Standards Organization). Standardizovaná verze JavaScriptu je pojmenována jako ECMAScript a z ní byly odvozeny i další implementace, jako je například ActionScript.

JavaScript byl původně obchodní název implementace společnosti Netscape, kde byl vyvíjen nejprve pod názvem Mocha, později LiveScript, ohlášen byl společně se společností Sun Microsystems v prosinci 1995 jako doplněk k jazykům HTML a Java. Pro verzi firmy Microsoft je použit název JScript.

Program v JavaScriptu se obvykle spouští až po stažení WWW stránky z internetu (tzv. na straně klienta), narozdíl od ostatních jiných interpretovaných programovacích jazyků (např. PHP a ASP), které se spouštějí na straně serveru ještě před stažením z Internetu. Z toho plynou jistá bezpečnostní omezení, JavaScript např. nemůže pracovat se soubory, aby tím neohrozil soukromí uživatele.

JavaScript je možné použít i na straně serveru. První implementací JavaScriptu na straně serveru byl LiveWire firmy Netscape vypuštěný roku 1996, dnes existuje několik možností včetně opensource implementace Rhinola založená na Rhino, gcj a Apache.

6.5 MySql

MySQL je databázovým systémem, vytvořený švédskou firmou MYSQL AB. Jeho hlavními autory jsou Michael „Monty“ Widenius a David Axmark. MySql je považován za úspěšného průkopníka dvojího licencování – je k dispozici jak pod bezplatnou licenci GPL tak pod komerční placenou licenci.

MySQL je multiplatformní databáze. Komunikace s ní probíhá – jak už název napovídá – pomocí jazyka SQL. Podobně jako u ostatních SQL databází se jedná o dialekt tohoto jazyka s některými rozšířeními.

Pro svou snadnou implementovatelnost (lze jej instalovat na Linux, MS Windows, ale i dalších operačních systémech), výkon a především díky tomu, že se jedná o volně dostupný software, má vysoký podíl na v současné době používaných databázích. Velmi oblíbená a často nasazovaná je kombinace MySQL, PHP a Apache jako základní software webového serveru.

MySQL byla od počátku optimalizována především na rychlost, a to i za cenu některých zjednodušení: má jen jednoduché způsoby zálohování, a až donedávna nepodporovala pohledy, trigger, a uložené procedury jsou doplňovány teprve v posledních letech, kdy začaly nejčastějším uživatelům produktu – programátorům webových stránek – již poněkud scházet.

Každá databáze v MySQL obsahuje *tabulky*, každá tabulka má *sloupce a řádky* – v každém řádku jsou záznamy předem určeného typu. Databáze MySQL je jeden z prvních hojně rozšířených systémů. Práce s tímto systémem se dá využít v C, C++, Java, Perl, PHP, Python, Tcl, Visual Basic nebo .NET.

Pro jednoduchou správu MySQL databází se používá nástroj PhpMyAdmin . PhpMyAdmin je Open Source program napsaný v PHP, který umožňuje zálohování, vytváření tabulek, vkládání, editaci a mazání záznamů v tabulkách, vytváření databází apod. PhpMyAdmin je pokročilý nástroj pro kompletní správu MySQL systému přes webové rozhraní.

7 Závěr

Závěrem bych rád pár slovy zhodnotil výsledný naimplementovaný informační systém internetového autobazaru a nastínil další možnosti vývoje.

Vývoj systému probíhal podle stanovených pravidel, tedy nejdříve analýzou celého systému a specifikací požadavků, poté návrhem databázové části systému a návrhem webové aplikace a na závěr samotnou implementací systému. Zejména v průběhu implementace však docházelo k novým poznatkům a zjištěním, které ovlivnily a někdy i zásadně změnily předchozí kroky (např. k navržené tabulce *klient* byl při implementaci přidán sloupec *platnost*, který zamezuje přihlášení zaregistrovaných klientů, kteří však na ještě neklikli na odkaz zaslaný klientovi emailem.).

Ve druhé kapitole této práce jsem celý systém rozebral z různých hledisek a popsal hlavní (i ty méně důležité) funkce systému a možnosti činností jednotlivých uživatelů autobazaru. Z tohoto hlediska je výsledný internetový autobazar plně funkční. A jeho webová aplikace splňuje základní pravidla dobré a kvalitní webové stránky.

Tato práce pro mě byla velkým přínosem, zejména proto, že návrh informačních systémů je oblast informatiky, která mě velmi zajímá. Naučil jsem se při ní aplikovat spoustu užitečných věcí (mimo jiné zasílání automatických emailů, práci s javascriptem, ...), které zcela bezpochyby v budoucnosti uplatním.

Možnosti dalšího vývoje práce je hned několik. Asi nejzásadnější je rozšíření autobazaru tak, aby nenabízel v katalogu vozidel jen vlastní automobily, ale umožnil přihlášeným klientům vkládat i jejich „inzeráty“ (informace o vozidlech), autobazaru by pak z takového prodaného autobazaru náležela určitá předem domluvená provize.

Další možností vývoje je rozšířit diskusní fórum tak, aby správce systému mohl editovat diskusní skupiny (přidávat nové skupiny, ...) to je v naimplementovaném systému možné jen programátorskou cestou.

Literatura

- [1] Steve Krug: Web design nenuťte uživatele přemýšlet, 2003.
- [2] Larry Ullman: PHP a MySQL, 2004.
- [3] Luboslav Lacko: PHP a MySQL hotová řešení, 2005.
- [4] Luke Welling, Laura Thomson: MySql Průvodce základy databázového systému, 2004.
- [5] Tom Negrino, Dori Smith: Javascript pro world wide web, 2003.
- [6] Jim Arlow, Ila Neustadt: UML a unifikovaný proces vývoje aplikací, 2003.
- [7] Ing. Bohuslav Křena, Ph.D., Ing. Radek Kočí Ph.D.. Úvod do softwarového inženýrství, IUS, Studijní opora, 2006.
- [8] Prof. Ing. Tomáš Hruška, CSc., Ing. Zbyněk Křivka: Informační systémy (IIS, PIS), Analýza, návrh, implementace informačního systému, Studijní opora, 2006.
- [9] Dokument Ministerstva informatiky České republiky – Pravidla pro tvorbu přístupného webu.

Seznam příloh

Příloha 1. Vytvoření databázových tabulek

Příloha 2. URL adresa webové aplikace

Příloha 3. CD/DVD ...

Příloha 1. Vytvoření databázových tabulek

- Klient

```
create table klient (  
    ID_klient mediumint(8) unsigned not null auto_increment,  
    login      varchar (20)      default null,  
    heslo      varchar (10)      default null,  
    datum_registrace date,  
    cas_registrace time,  
    jmeno      varchar (20)      not null,  
    prijmeni   varchar (20)      not null,  
    rodne_cislo int (10)         default null,  
    stat       varchar(15)       not null,  
    mesto      varchar (20)      not null,  
    ulice      varchar (20)      not null,  
    cislo_popisne int(6)        default null,  
    email      varchar (20)      not null,  
    posledni_prihlaseni date,  
    osloveni   varchar(20),  
    platnost   int (5)           not null,  
    primary key (ID_klient )  
);
```

- Cena

```
create table cena(  
    ID_cena mediumint(8) unsigned not null auto_increment,  
    ID_vozidla mediumint(9) not null  
        references auto(ID_vozidla),  
    poradi_ceny int (3)         not null,  
    platnost_od_darum date      not null,  
    platnost_od_cas time       not null,  
    cena       bigint (10)     not null,  
    primary key ( ID_cena )  
);
```


- Rezervoval

```

create table rezervoval (
  ID_rezervace   mediumint(8) unsigned not null auto_increment,
  Login   varchar(20) not null references klient(login),
  ID_vozidla mediumint(9) not null
                references auto(ID_vozidla),
  Datum_platnost_rok   int(4),
  Datum_platnost_mesic int(2),
  Datum_platnost_den   int(2),
  Cas_platnost_hodina  int(2),
  Cas_platnost_minuta  int(2),
  Datum_rezervace date           not null,
  Cas_rezervace   time           not null,
  poradi_rezervace int(3)           not null,
  primary key ( ID_rezervace )
);

```

- Auto

```

create table auto (
  ID_vozidla mediumint(8) unsigned not null auto_increment,
  znacka    varchar (20) not null references znacka(znacka),
  model     varchar (20)           not null,
  rok_vyroby int(4)                 not null,
  cena      bigint(10)           not null,
  palivo    varchar (20)           not null,
  vykon    varchar (20)           not null,
  karoserie varchar (20)           not null,
  prevodovka varchar (20)           not null,
  tachometr varchar(20)           not null,
  barva     varchar (20)           not null,
  vybava    varchar (100)          not null,
  majitel   mediumint(9) not null references klient(ID_klient),
  primary key ( ID_vozidla )
);

```

- Značka

```

create table značka (
    značka          varchar (20)      not null,
    sidlo           varchar (20)      not null,
    rok_zalozeni    int (5)           not null,
    odkaz           varchar (40)      not null,
    primary key ( značka )
);

```

- Zpráva

```

create table zprava (
    ID_zprava mediumint(8) unsigned not null auto_increment,
    jmeno     varchar (15),
    email     varchar (15),
    datum     date           not null,
    cas       time           not null,
    zprava    varchar (255)   default null,
    primary key (ID_zprava )
);

```

- Soubory a fotografie

```

create table soubor (
    ID_soubor mediumint(8) unsigned not null auto_increment,
    datum     date           not null,
    cas       time           not null,
    ID_vozidla mediumint(9) references auto(ID_vozidla),
    značka    varchar(20)   references značka(značka),
    poradi    int(3),
    soubor    blob         not null,
    primary key ( ID_soubor )
);

```

- Diskuse

```
create table diskuse (  
    ID_diskuse mediumint(8) unsigned not null auto_increment,  
    skupina      varchar (20)      not null,  
    datum       date              not null,  
    cas         time              not null,  
    login       varchar (20)      not null,  
    reakce      mediumint (8)      not null,  
    zprava      varchar (255)      not null,  
    primary key (ID_diskuse )  
);
```

- Správce

```
create table spravce (  
    login         varchar (15)      not null,  
    heslo        varchar (15)      not null,  
    primary key (login )  
);
```

Příloha 2 URL adresa webové aplikace

<http://www.stud.fit.vutbr.cz/~xhorak44/>