



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



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

WEBOVÁ APLIKACE AUTOŠKOLY

WEB APPLICATION FOR A DRIVING SCHOOL

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ROMAN HALÍK

VEDOUcí PRÁCE

SUPERVISOR

Ing. RADEK BURGET, Ph.D.

BRNO 2016

Zadání bakalářské práce

Řešitel: **Halík Roman**

Obor: Informační technologie

Téma: **Webová aplikace autoškoly**
Web Application for a Driving School

Kategorie: Web

Pokyny:

1. Prostudujte existující technologie pro tvorbu webových aplikací se zaměřením na responsivní aplikace s JavaScriptovým klientem.
2. Seznamte se s požadavky zadavatele na aplikaci pro správu webu autoškoly a plánování jízd.
3. Navrhnete architekturu aplikace na základě požadavků zadavatele.
4. Po dohodě s vedoucím implementujte navrženou aplikaci s využitím vhodných technologií. Implementujte interaktivní systém rezervace jízd.
5. Optimalizujte aplikaci pro použití na mobilních zařízeních.
6. Proveďte testování vytvořené aplikace a zhodnoťte dosažené výsledky.

Literatura:

- Gutmans, A., Rethans, D., Bakken, S.: Mistrovství v PHP 5, Computer Press, 2012
- Žára, O.: JavaScript - Programátorské techniky a webové technologie, Computer Press, 2015
- Dokumentace k projektu Nette: <http://doc.nette.org/cs/2.2/>

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 3

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Burget Radek, Ing., Ph.D.**, UIFS FIT VUT

Datum zadání: 1. listopadu 2015

Datum odevzdání: 18. května 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav informačních systémů
602 00 Brno, Božetěchova 2

doc. Dr. Ing. Dušan Kolář
vedoucí ústavu

Abstrakt

Tato práce se věnuje problematice návrhu a implementace webové prezentace, redakčního systému a rezervačního systému jízd autoškoly. Hlavní funkcionalitou webové aplikace je možnost on-line plánování rozvrhu jízd, na které se mohou žáci autoškoly přihlašovat prostřednictvím svých uživatelských účtů. Interaktivní rozhraní rezervačního systému je optimalizováno pro pohodlnou obsluhu na chytrém telefonu, stejně jako webová prezentace. Této funkcionality je dosaženo použitím responzivního designu a JavaScriptového prostředí, ve kterém aplikace pracuje, za využití frameworku Bootstrap a knihovny jQuery. Komunikace s databází MySQL, ve které jsou uložena všechna data nezbytná pro běh redakčního a rezervačního systému, zabezpečují funkce jazyka PHP.

Abstract

This thesis aims to clarify and describe issues of design and implementation of web presentation and web application for administration system and reservation system for students of driving school. Main functionality of application is on-line scheduling of rides, which can be registered by students of driving school through their user accounts. Interactive interface of reservation system is optimized for comfortable use on smartphones as much as web presentation. This functionality is reached by using responsive design and JavaScript for application interface, with use of Bootstrap framework and jQuery library. Communication with MySQL database, which is used for store all data necessary for run administration and reservation system, is implemented with functions of PHP language.

Klíčová slova

Rezervační systém, Redakční systém, Webová prezentace, HTML, CSS, JavaScript, PHP, MySQL, Bootstrap, jQuery

Keywords

Reservation system, Administration system, Web presentation, HTML, CSS, JavaScript, PHP, MySQL, Bootstrap, jQuery

Citace

HALÍK, Roman. *Webová aplikace autoškoly*. Brno, 2016. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Burget Radek.

Webová aplikace autoškoly

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Radka Burgeta, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Roman Halík
16. května 2016

Poděkování

Děkuji panu Ing. Radku Burgetovi, Ph.D. za cenné rady, odborný dohled, a vedení mojí bakalářské práce, a paní Evě Vosinkové, majitelce autoškoly V&V, za spolupráci při návrhu a testování webové aplikace.

© Roman Halík, 2016.

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

1	Úvod	3
1.1	Cíle práce	3
1.2	Motivace	3
1.3	Struktura práce	3
2	Specifikace požadavků	5
2.1	Analýza původního informačního systému	5
2.2	Analýza původní webové prezentace autoškoly	6
2.3	Požadavky na nový informační systém	6
2.3.1	Systém rezervace jízd	6
2.3.2	Požadavky na systém správy obsahu webových stránek	7
2.4	Požadavky na webovou prezentaci	8
2.4.1	Moderní vzhled	8
2.4.2	Responzivní design	8
2.5	Cílová skupina uživatelů a jejich specifika	8
2.5.1	Návštěvníci webových stránek	8
2.5.2	Osoby pracující s rezervačním systémem	9
2.5.3	Osoby pracující s redakčním systémem	9
2.6	Požadavky na technologie	9
3	Analýza a návrh	10
3.1	Studium podobných aplikací	10
3.2	Vhodné technologie a jejich využití v aplikaci	10
3.2.1	Bootstrap	11
3.2.2	jQuery	11
3.3	Identifikace uživatelů systému	11
3.4	Návrh webové prezentace	11
3.4.1	Principy moderního designu	11
3.4.2	Barvy a písmo	12
3.5	Návrh grafického rozhraní redakčního a rezervačního systému	13
3.5.1	Redakční systém	13
3.5.2	Systém rezervace jízd	13
3.6	Ukládání a přenos dat	14
3.6.1	Komunikace se serverem – AJAX	14
3.6.2	Struktura tabulek v databázi	15
3.6.3	Přenos dat formátem JSON	17
3.7	Kritické prvky aplikace z pohledu obsluhy	17

4 Implementace	19
4.1 Vývojový cyklus	19
4.2 Principy komunikace klient-server	19
4.3 Implementační principy programových celků	20
4.3.1 Přihlašování do systému	20
4.3.2 Úprava textů a tabulek	21
4.3.3 Vyhledávání	21
4.3.4 Správa uživatelů	22
4.3.5 Správa rozvrhu	22
4.3.6 Přihlašování na jízdy – žáci	23
4.4 Grafické rozhraní systému a webové prezentace	23
4.4.1 Responzivní design	24
4.4.2 Interaktivní prvky	25
5 Testování a bezpečnost	27
5.1 Bezpečnost	27
5.2 Předpokládaná zátěž	27
5.3 Testování	28
5.3.1 Přihlašování	28
5.3.2 Správa žáků a učitelů	28
5.3.3 Změna textů a tabulek webových stránek	29
5.3.4 Správa rozvrhu	29
5.3.5 Práce se systémem – žák	29
5.3.6 Práce se systémem – učitel	30
5.4 Testování aplikace v internetových prohlížečích	30
6 Závěr	32
Literatura	33
Přílohy	34
Seznam příloh	35
A Obsah CD	36
B Instalace	37
C Manuál	38
C.1 Přihlášení	38
C.2 Rozvrh jízd	38
C.3 Editace webu	39
C.4 Žáci	39
C.5 Učitelé	40
C.6 Změna hesla	40
C.7 Žáci a učitelé	40
C.8 Odhlášení	40

Kapitola 1

Úvod

1.1 Cíle práce

Hlavním cílem této práce je vytvoření webové aplikace pro rezervaci jízd, kterou bude využívat Autoškola V&V. Tato aplikace bude umožňovat plánování jízd pro proměnný počet aut na týden dopředu, na které se budou žáci autoškoly přihlašovat on-line, prostřednictvím svých uživatelských účtů vytvořených v systému. Eliminuje se tak nutnost plánovat jízdy osobně v sídle autoškoly. To povede k menšímu administrativnímu vytížení osoby odpovědné za plánování rozvrhu (v současné době majitelka autoškoly), a také k vyššímu pohodlí plánování jízd pro všechny strany. Žáci autoškoly díky tomuto systému získají možnost měnit plánované jízdy dle svých potřeb a lépe je přizpůsobovat svým časovým možnostem. Současně lze předpokládat, že rezervační systém přidá na celkové atraktivitě autoškoly a poskytne jí tak vyšší konkurenceschopnost. Vedlejším, ačkoli neméně podstatným cílem je vytvoření nové, moderní webové prezentace autoškoly, a implementace pokročilejšího redakčního systému.

1.2 Motivace

Tvorba nové webové prezentace autoškoly je motivována zastaralým designem současných webových stránek. Vytvoření pokročilejšího redakčního systému je nutné především kvůli rozšíření možností editace obsahu webových stránek oproti původnímu redakčnímu systému, který neumožňuje provádění některých důležitých operací, jako například editace počtu řádků tabulek umístěných na webových stránkách.

Osobní motivací pro tvorbu webové prezentace, rezervačního a redakčního systému je rozvoj znalostí a zkušeností v oblasti problematiky informačních systémů a návrhu moderních webových prezentací.

1.3 Struktura práce

Nejprve bude popsána specifikace požadavků na webovou aplikaci a webovou prezentaci, součástí které je analýza původního redakčního systému autoškoly, rozbor cílové skupiny uživatelů a formulace požadavků na technologie. Následuje analýza požadavků, návrh webové prezentace a grafického rozhraní systému. Také bude zmíněn způsob ukládání a přenosu dat. Poté bude popsán princip implementace nejdůležitějších logických celků aplikace a webové

prezentace pomocí vhodných technologií zvolených na základě analýzy požadavků zadavatele. Bude také popsána komunikace mezi klientem a serverem a taktéž grafické rozhraní systému. Práci uzavírá kapitola věnovaná popisu testování aplikace, vyhodnocení výsledků a implementace změn.

Kapitola 2

Specifikace požadavků

Specifikace požadavků na nový informační systém a webovou prezentaci není jednoduchá záležitost. Především je třeba od zákazníka zjistit jeho potřeby a zhodnotit jejich proveditelnost, co se týká programové konstrukce. Není výjimkou, že zákazník neví, jaké možnosti a omezení dostupné technologie nabízí. Také se stává, že má zákazník buď nereálné požadavky, nebo se naopak bojí požadavky vyslovit, protože je názoru, že požaduje něco nemožného. Je tedy vhodné nejen od zákazníka zjistit jeho potřeby, ale také se nad danou problematikou zamyslet, a formulovat vlastní nápady na rozšíření/zjednodušení funkcionality.

Tato kapitola bude věnována analýze původního informačního systému a webové prezentace autoškoly. Následně budou formulovány požadavky na nový informační systém a webovou prezentaci, poté budou uvedena specifika cílové skupiny uživatelů, a nakonec požadavky na technologie, kterými bude nový systém vyvíjen.

2.1 Analýza původního informačního systému

Původní informační systém je klasický redakční systém, který vytváří pro administrátora prostředí pro správu obsahu webových stránek. Konkrétně umožňuje editaci většiny textů na stránkách, a také obsahu tabulek. Jeho zásadním nedostatkem je však nemožnost měnit počet položek v tabulkách. Například, pokud uvažujeme tabulku „Kondiční a zdokonalovací jízdy“, která obsahuje názvy vozidel a jim odpovídající ceny za hodinu kondiční jízdy, je problém zcela jasný. Jestliže autoškola koupí nový vůz, a chce pro něj nabídnout kondiční jízdy, neexistuje efektivní způsob, jak problém vyřešit. Samozřejmě je možné vepsat na jeden řádek dvě auta a ceny, ale tabulka tím ztrácí konzistentní způsob zápisu dat a tím se stává nepřehlednou.

Dalším nedostatkem původního informačního systému je absence možnosti on-line přihlašování na jízdy. V této funkcionalitě tedy spočívá hlavní přínos nového informačního systému. Současný systém plánování jízd spočívá v manuálním vypisování žáků a učitelů do předtištěných tabulek, které má dva hlavní nedostatky:

- nutnost osobního setkání žáka autoškoly s osobou, která je pověřená plánováním jízd (majitelka autoškoly),
- složité změny přihlášených žáků, které mohou vést k nepřehledným zápisům jízd a omylům při jejich plánování.

2.2 Analýza původní webové prezentace autoškoly

Nedostatkem původní webové prezentace autoškoly je dnes již zastaralý, nemoderní design, který nemá v dnešní době čím zaujmout a jak nalákat nové studenty do autoškoly. Místy je text na stránkách na hranici pohodlné čitelnosti. Jistým nedostatkem je také způsob členění informací na stránkách. Některé texty jsou příliš dlouhé a návštěvníka webových stránek autoškoly tak odrazují od čtení.

2.3 Požadavky na nový informační systém

Přední požadavek na nový informační systém byl formulován již v předchozí části tohoto textu – jedná se o rozšíření systému o možnost on-line rezervace jízd pro žáky autoškoly. S tímto rozšířením však souvisí celá řada funkcí, které musí nový informační systém umožňovat. Konkrétně se jedná o nutnost správy informací o žácích, učitelích, a počtu aut, pro která se bude rozvrh jízd plánovat. Je tedy třeba zajistit, aby režie spojená se správou těchto informací byla minimální a neodrazovala osobu odpovědnou za plánování jízd od používání systému.

Druhý zásadní požadavek je možnost editace počtu řádků tabulek na webových stránkách, jelikož tuto funkci původní systém postrádá.

2.3.1 Systém rezervace jízd

Požadavky na systém rezervace jízd se odvíjí od konkrétního uživatele (respektive skupiny uživatelů), který bude s touto částí systému pracovat. Pro všechny skupiny však platí, že je třeba jejich uživatele nějakým způsobem identifikovat. Je tedy třeba zvolit vhodné (jednoznačné) přihlašovací jméno a implicitní heslo, které si později uživatel musí mít možnost změnit.

Učitelé

Učitel zastupuje skupinu, která s rezervačním systémem přímo žádné akce neprovádí, tedy nemění jeho stav. Výjimku tvoří možnost změny hesla. Učitel by neměl mít právo se sám přihlašovat na termíny jízd, aby nedocházelo ke kolizím s ostatními učiteli, ve smyslu boje o nejlepší termíny. Je nutné tuto pravomoc přenechat pouze administrátorovi, který má de facto roli nezávislého arbitra, jenž rozhoduje, kterou jízdu kterému učiteli přiřadí. Zná totiž potřeby a časové možnosti všech učitelů současně, a může tak jízdy plánovat spravedlivě, ke spokojenosti všech.

Učitel by tedy měl mít právo vidět všechny své jízdy a žáky, kteří jsou na ně nahlášení, nebude mít však žádnou možnost rozvrh přímo změnit. V každém případě je ale vhodné vytvoření stylů pro tisk, aby si učitel mohl rozvrh jízd vytisknout v přehledné podobě.

Žáci

Hlavním požadavkem na funkcionalitu přístupnou pro žáky autoškoly je možnost se na jízdu přihlásit, nebo se z ní i odhlásit, přičemž je třeba zajistit, aby žáci neměli možnost vidět, ke kterému učiteli se nahláší. Mohlo by docházet k nežádoucímu upřednostňování některých učitelů.

Důležité je také znemožnit žákovi odhlášení z jízdy, respektive přihlášení na jízdu, jestliže se má jízda uskutečnit za méně než 24 hodin. V opačném případě by se mohlo stát, že žák,

který jízdu z různých důvodů nestíhá, si ji odhlásí například 10 minut před jejím začátkem. Tím by docházelo ke zbytečnému blokování vozidel i učitelů, a tedy i k finančním ztrátám. Rezerva 24 hodin pro přihlášení na jízdu je zase nutná kvůli dostupnosti učitelů. Uvažujme případ, kdy je učitel přihlášený například na pouhé dvě jízdy v celém dni, které ještě ke všemu začínají ráno v 6:30, a den před jízdou k nim není přihlášený žádný žák. Jaké by asi bylo nadšení učitele, který v 6 hodin zjistí, že se mu v průběhu noci někdo přihlásil na jízdy. Jedná se pochopitelně o extrémní případ, nicméně principiálně jde o záležitost, která značně zneprůjemňuje organizaci celého rozvrhu jízd. Je tedy třeba takové situace předcházet.

Mimo jiné je nutné znemožnit přihlášení více jízd v jednom dni (i pro různá auta), stejně jako překročení maximálního počtu jízd – ať už celkově (14), nebo týdně (3). Týdenní omezení je nutné z důvodu, aby měli všichni žáci autoškoly možnost využívat on-line přihlašování na jízdy. V reálném provozu se předpokládá, že běžný student si bude plánovat jednu až dvě jízdy týdně.

Administrátor

Administrátor musí mít možnost provádět nad rezervačním systémem veškeré možné operace – mít nad ním naprostou kontrolu. Kromě přihlašování/odhlašování žáků a učitelů na jízdy je nutné zajistit možnost některé jízdy pro přihlašování zcela zablokovat, pro případ nedostatečné kapacity aut nebo učitelů – z důvodu poruchy, respektive nemoci. Administrátor by také měl mít právo přihlásit žáka na více jízd v jednom dni, případně přihlásit/odhlásit žáka z jízdy bez časového omezení (24 hodin předem). Využití této funkce se sice nepředpokládá, ale administrátor by už z principu neměl být systémem příliš omezován v akcích, které nad ním chce provádět, pokud jimi neporuší integritu dat.

Také je třeba, aby měl administrátor možnost spravovat systémové údaje o žácích a učitelích, a také měnit počet aut, pro která se rozvrh plánuje. Tato funkcionalita je potřebná pro postupný přechod na systém, a také pro potřebu plánovat rozvrh pro více/méně aut v případě přírůstku do vozového parku autoškoly, nebo naopak zmenšení počtu dostupných aut při dlouhodobém servisu, nebo úplného vyřazení některého vozu autoškoly bez náhrady.

2.3.2 Požadavky na systém správy obsahu webových stránek

Je nutné, aby část informačního systému zabývající se správou obsahu webových stránek v každém případě pokrývala funkcionalitu původního systému, a dále ji vhodně rozšiřovala o funkce, které původní systém postrádal.

Úprava textů

Úprava textů na webových stránkách dle požadavků zadavatele postačuje na úrovni čistého textu – jinak řečeno, není třeba, aby systém umožňoval úpravu stylu textu. Kvůli nepříliš vysoké znalosti výpočetní techniky ze strany administrátora je vhodné systém editace textů navrhnout co nejjednodušší, tedy bez komplikací v podobě stylování, které by spíše než rozšířené možnosti mohlo způsobovat zmatení osoby obsluhující redakční systém. Častá úprava textů se ani nepředpokládá. Pokud by došlo k zásadním změnám textu a jeho stylování, řešil by tento úkon programátor.

Úprava tabulek

Nejdůležitějším požadavkem na úpravu tabulek je možnost přidání nového řádku. Tato funkcionality v původním systému chybí, a může za jistých okolností znamenat těžko řešitelný problém. Kromě přidání nového záznamu (řádku) do tabulky je nutností možnost úpravy textů v jednotlivých buňkách a mazání řádků tabulky.

2.4 Požadavky na webovou prezentaci

2.4.1 Moderní vzhled

Na webové stránky autoškoly přistupují v převážné většině mladí lidé, s věkovým průměrem mezi 20 a 25 lety, a těm je potřeba přizpůsobit vzhled stránek tak, aby pro byl líbivý, poutavý, ale současně jednoduchý a přehledný. Je obecně známý fakt, že jestliže uživatel na stránce nenajde to, co hledá, během několika jednotek až nízkých desítek vteřin ze stránky odchází. Webová prezentace musí zaujmout, současně ale umožnit rychlou orientaci v její struktuře a tedy i rychlé vyhledání požadovaných informací.

2.4.2 Responzivní design

Je třeba brát v úvahu, že mladí lidé velmi často přistupují na webové stránky skrz chytré telefony a tablety, které nemají tak velikou obrazovku, jako monitor stolního počítače, nebo notebooku. Design webové prezentace proto musí být responzivní, aby korektně reagoval na změnu velikosti obrazovky.

2.5 Cílová skupina uživatelů a jejich specifika

Cílová skupina uživatelů se liší v závislosti na části aplikace, kterou uvažujeme. Zjednodušeně je možné provést rozdělení na uživatele, kteří:

- prohlíží webové stránky,
- pracují s rezervačním systémem,
- pracují s redakčním systémem.

Příčemž pro každou skupinu platí jiná specifika, která ji charakterizují, a podle kterých je třeba upravit odpovídající část aplikace. Pro všechny skupiny platí, že je nutné uvážit možné zrakové indispozice a proto je vhodné zajistit dobrou čitelnost veškerých textů (vhodná velikost písma, dostatečný kontrast) a velikost ovládacích prvků. Navigační prvky je nutné volit jednoznačné, dobře vystihující zamýšlenou operaci. Ideální je, když je uživatel schopný používat aplikaci (provádět zamýšlené operace) aniž by mu byla práce s aplikací vysvětlena, přičemž uživatel sám si je jistý tím, co za operace provádí.

2.5.1 Návštěvníci webových stránek

Jsou zpravidla zájemci o výuku v autoškole. Jejich typický věk se pohybuje mezi 20 až 25 lety, a nezanedbatelná část z nich přistupuje na webové stránky prostřednictvím chytrého telefonu nebo tabletu. Této skutečnosti je třeba přizpůsobit návrh webové prezentace. Webovou stránku, která se na zmíněných zařízeních zobrazuje nevhodně – tedy nečitelně,

nebo dokonce s poškozeným rozmístěním prvků, uživatel pravděpodobně velmi rychle opustí a autoškola tím přichází o potenciálního zákazníka.

2.5.2 Osoby pracující s rezervačním systémem

Zastupují rozsáhlou věkovou skupinu, přibližně od 17 do 65 let. Jedná se o registrované studenty autoškoly, učitele, a administrátora. Kvůli věkové rozmanitosti této skupiny je třeba brát ohled na uživatele s nepříliš vysokými znalostmi výpočetní techniky. Tuto skupinu je vhodné dále rozdělit na registrované žáky autoškoly, učitele a administrátora.

Žáci

Žáci tvoří nejpočetnější skupinu, která bude s rezervačním systémem pracovat. U této skupiny je třeba počítat se skutečností, že značná část žáků bude využívat služby systému prostřednictvím svých chytrých telefonů a tabletů. Je tedy žádoucí odpovídajícím způsobem přizpůsobit grafické rozhraní aplikace. Prakticky to znamená použití responzivního designu, který přizpůsobí vzhled webové stránky velikosti obrazovky. Je nutné zajistit, aby byla aplikace stejně dobře použitelná na chytrém telefonu, tabletu, i stolním počítači.

Administrátor

Administrátor je osoba středního věku, která přistupuje k rezervačnímu systému prostřednictvím stolního počítače. Je tedy možné předpokládat dostatečný prostor (velikost obrazovky) pro pohodlné rozmístění všech ovládacích prvků, současně je ale nutné zabezpečit maximální možnou míru intuitivity ovládání kvůli nepříliš vysoké znalosti výpočetní techniky osobou administrátora. V praxi to znamená jednoznačné popisky ovládacích prvků, použití ikon a vhodných barev které napovídají, jakou operaci provede daná akce, a potvrzování operací nad systémem, které jsou nevratné – typicky mazání nějaké položky.

Učitelé

Typický učitel v autoškole je muž středního věku s průměrnou znalostí výpočetní techniky. Lze předpokládat, že někteří učitelé budou využívat služby informačního systému prostřednictvím svých chytrých telefonů a tabletů, a této skutečnosti přizpůsobit rozhraní části systému, ke které mají učitelé přístup.

2.5.3 Osoby pracující s redakčním systémem

Specifikace této skupiny uživatelů je popsána již v předchozí části textu, konkrétně v části „Osoby pracující s rezervačním systémem – Administrátor“, jelikož osoba odpovědná za plánování rozvrhu jízd (obsluhující rezervační systém v roli administrátora) je totožná s osobou obsluhující redakční systém.

2.6 Požadavky na technologie

Požadavky na technologie zvolené pro vývoj webových stránek, rezervačního a redakčního systému úzce souvisejí s náklady na webhosting. Proto je třeba volit takové technologie, které umožní financování pouze základní (současné) varianty hostingu, aby se předešlo dalším nákladům na provoz webových stránek autoškoly.

Kapitola 3

Analýza a návrh

Tato část textu se zabývá analýzou požadavků na aplikaci a návrhem řešení. Bude popsáno studium podobných aplikací, následně vhodné technologie pro splnění požadavků na informační systém a webovou prezentaci, způsob identifikace uživatele, poté grafický návrh webové prezentace a rozhraní redakčního a rezervačního systému, dále způsob ukládání a přenosu dat, a nakonec kritické prvky aplikace z pohledu obsluhy.

3.1 Studium podobných aplikací

Studium podobných aplikací je do jisté míry komplikované, jelikož z pochopitelných důvodů je přístup do rezervací jízd konkurenčních autoškol chráněný. Průzkumem internetových stránek konkurenčních autoškol lze zjistit, že ve velké většině autoškoly využívají k plánování jízd komerční rezervační systémy, které nemusí být nutně k rezervacím jízd autoškoly optimalizované, a v každém případě nejsou zdarma. Další běžnou praxí je plánování jízd pomocí aplikace „Google kalendář“, která je možná obecně vhodná pro plánování, není však v žádném ohledu optimalizovaná pro plánování jízd autoškoly.

Ve většině konkurenčních systémů platí podobná pravidla pro plánování jízd – tedy (od)registrace alespoň 24 hodin předem, a nejčastěji plánování jízd na týden dopředu. V komentářích o některých konkurenčních autoškolách v internetových diskuzích se absolventi zmiňují o systému rezervací, který je mezi studenty autoškoly natolik populární, že je nutné si jízdy registrovat s velkým časovým předstihem, protože jsou jízdy na dva týdny dopředu zcela zabrané. Z těchto tvrzení lze vyrozumět, že aplikace pro rezervaci neumožňovala žádné omezení počtu současně registrovaných jízd. Toto řešení bohužel nemusí být vždy vhodné a spravedlivé.

3.2 Vhodné technologie a jejich využití v aplikaci

Z požadavků na technologie přímo vyplývá, že je třeba se orientovat na takové technologie, které podporují i základní varianty webhostingů, jelikož je nežádoucí navyšování nákladů autoškoly na provoz webových stránek. Takové předpoklady splňují programovací jazyky HTML, CSS, PHP, databáze MySQL, a JavaScript. Použití těchto základních programovacích jazyků pro tvorbu webu zabezpečí jeho spustitelnost na téměř jakémkoli webovém serveru, poskytující i ty nejzákladnější služby, jako je právě PHP a databáze MySQL.

Jazyk HTML najde využití pro vytvoření kostry webové stránky, pomocí jazyka CSS bude stránka nastýlována. PHP [4] zabezpečí spolehlivou komunikaci se serverem. Na straně

druhé – klientské, bude pracovat JavaScript [8], díky kterému je možné vytvořit pro uživatele interaktivní prostředí. A konečně jazyk SQL poslouží pro operace nad databází MySQL, ve které budou uložena veškerá data. Pro pohodlné zajištění responzivního designu je vhodné využít volně dostupný framework – tzv. Bootstrap [2].

3.2.1 Bootstrap

Jedná se o knihovnu určenou pro webové aplikace, jejíž cílem je usnadnění práce a sjednocení postupů při tvorbě webových stránek. Mezi přední výhody Bootstrapu patří úspora času při tvorbě webové prezentace, responzivní design, kompatibilita se všemi moderními prohlížeči, a především fakt, že celý framework je tzv. „Open Source“ – je tedy volně přístupný ke stažení a jeho používání je zcela zdarma.

3.2.2 jQuery

K posílání dat bez obnovení stránky (dosažení interaktivního chování aplikace) poslouží volně dostupná JavaScriptová knihovna – jQuery. Tato knihovna vytváří efektivní rozhraní pro práci s technologií AJAX a značným způsobem usnadňuje dynamickou změnu obsahu a stylů stránky, případně různé animace.

3.3 Identifikace uživatelů systému

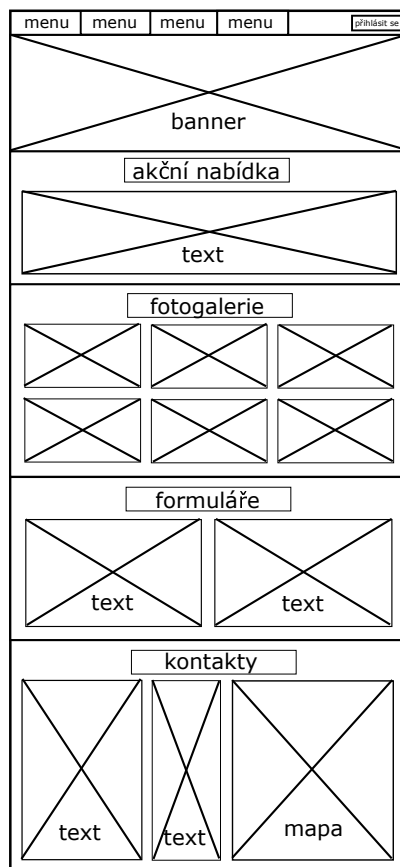
Každý uživatel systému musí mít přiřazeno unikátní přihlašovací jméno. Takový předpoklad splňuje například rodné číslo, které každý uchazeč vepisuje do formuláře žádosti o přijetí k výuce a výcviku, který se, jak název napovídá, odevzdává vedení autoškoly ještě před započítáním jízd. Podle informací uvedených v tomto formuláři pak administrátor zaregistruje žáka do systému a umožní mu tak plánování jízd. Na tuto skutečnost stačí žáky upozornit ať už na úvodní hodině výuky, nebo jiným způsobem, a při implicitním nastavení hesla na dohodnutý řetězec jsou přihlašovací údaje kompletní a jednoznačné.

3.4 Návrh webové prezentace

3.4.1 Principy moderního designu

Z požadavků na webovou prezentaci, uvedených v předchozí kapitole, jasně vyplývá potřeba autoškoly oživit současný vzhled stránek tak, aby byl pro nové návštěvníky (potenciální studenty autoškoly) poutavý. Vzhled stránky musí zaujmout, ale současně nesmí být na úkor čitelnosti textů a celkové přehlednosti stránek. Design je proto nutné volit tak, aby byl jednoduchý, přehledný, účelný, a přitom moderní. Taková kritéria splňuje jednostránkový design (anglicky „single page design“), viz obrázek 3.1, který je v současné době považován za moderní. Při jeho promyšleném návrhu získá webová prezentace líbivý vzhled, a uživatel dobrou schopnost orientace v obsahu webové stránky. Informace, které spolu nějakým způsobem souvisí, se sloučí do jedné dlouhé strany, která se nejčastěji pomocí barev a nadpisů rozčlení na dílčí části. Všechny důležité informace však zůstávají pospolu.

Velkou výhodou jednostránkového designu je, že uživatel již nemusí rozklikávat 20 položkové menu – namísto toho v například čtyřech záložkách na vrcholu stránky najde vše, co hledá. Je však důležité informace na stránce slučovat obezřetně, a vždy jen rozumný počet, aby se z výhody nestala nevýhoda. Jestliže uživatel narazí na webovou stránku tvořenou



Obrázek 3.1: Drátový model úvodní stránky webové prezentace.

stylem „single page design“, na které bude chybět jakákoli navigační struktura a strana jako taková bude obsahovat desítky nadpisů, uživatel rychle ztratí chuť na stránce vyhledávat informace.

Použití jednostránkového designu je v případě autoškoly velmi vhodnou volbou, jelikož je možné snadno seskupit informace, které mají být prezentovány, do čtyř disjunktních množin. Jedná se o množinu informací týkajících se klasických kurzů autoškoly (například výcvikové kurzy pro skupiny A a B), dále o množinu informací pojednávajících o kurzech profesní způsobilosti, poté obecné informace o autoškolě, a nakonec kontakty. Do těchto čtyř množin lze snadno seskupit takové množství informací, které vyplní dílčí stránku webové prezentace tak, aby se skládala ze tří až pěti podnadpisů – tedy takového množství, které umožní vyniknout jednostránkovému designu, ale současně zbytečně neprodlužuje stránku natolik, aby se stala pro návštěvníka příliš dlouhá, a demotivující ke čtení.

3.4.2 Barvy a písmo

Po ujasnění základní struktury vzhledu je důležité se zamyslet nad barevným schématem webové prezentace. Barvy je mimo jiné třeba volit s ohledem na jejich pocitové vnímání člověkem. Například žlutá v kombinaci s černou působí agresivně, červená unavuje oči a může působit nepřátelsky. Naproti tomu zelená je lidskému oku barvou nepříjemnější – neunavuje zrak, a na člověka působí přátelsky, dává pocit bezpečí, evokuje přírodu a ekologii [6]. Je ale nutné volit její odstíny citlivě – tedy takové, které nejsou příliš svítivé (agresivní). Spíše je

vhodné použití „teplejších“ odstínů zelené, tedy těch s převahou žluté barvy oproti modré. Do kombinace s odstínem zelené se velmi hodí tmavě šedá. Vytváří dostatečný kontrast (je tedy vhodná pro použití jako barva písma) a oproti černé barvě (jinak typicky používané pro text) působí měkce, moderně. Jestliže pak ke dvěma odstínům zelené a tmavě šedé přidáme bílou, dostaneme kombinaci čtyř barev, které dohromady tvoří dostatečný kontrast a upoutávají pozornost, ale nevynucují si ji. Jinými slovy, promyšlené kombinace těchto barev mohou velmi dobře usnadnit orientaci na stránce a pomoci tak zdůraznit podstatné prvky webové prezentace, ale neodvádí uživatelu pozornost od sdělené informace – nezpůsobují dezorientaci.

Písmo je vhodné volit bezpatkové, jelikož je v současném pojetí obecně považováno za moderní. Je však nutné následně zabezpečit, aby se v obsahu webové prezentace nevyskytovaly příliš dlouhé texty, jejichž četba je při použití bezpatkového písma obtížná. Velikost písma je třeba volit kompromisem. Příliš malé písmo demotivuje uživatele od čtení textu, jelikož mu čtení a celkovou orientaci na stránce značně znesnadňuje. Příliš velké písmo zase plýtvá místem a může působit rušivě.

3.5 Návrh grafického rozhraní redakčního a rezervačního systému

Grafické rozhraní redakčního a rezervačního systému by mělo vycházet z návrhu webové prezentace jako takové. Usnadní se tím orientace uživatele v systému – uživatel pracuje s něčím, co do jisté míry zná. Rychleji tak pochopí, jakým způsobem má se systémem pracovat, a kde hledat informace, se kterými chce nakládat.

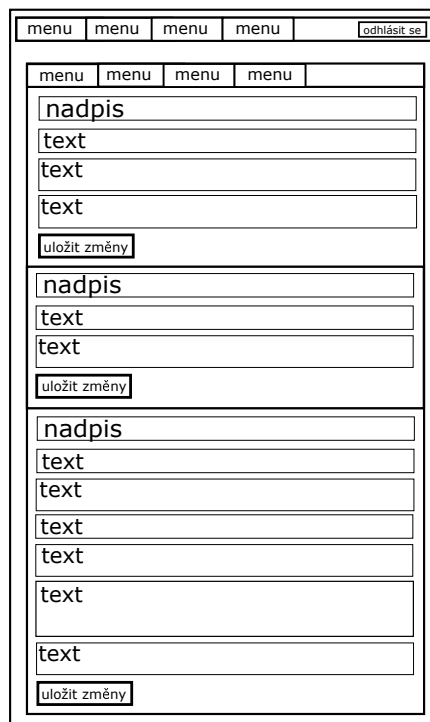
Obecně je poutavé použití různých interaktivních prvků, které usnadní komunikaci systému s osobou, která jej obsluhuje. Jako pozitivně vnímané se jeví barevné odlišení (případně podtržení nebo změna orámování) prvků, se kterými je v dané chvíli manipulováno. Stránka musí působit živě, a dávat najevo reakce na uživatelské podněty.

3.5.1 Redakční systém

Jak již bylo řečeno, grafický návrh redakčního systému je vhodné koncipovat tak, aby byl co nejvíce podobný grafické podobě samotných webových stránek, viz obrázek 3.2. Výrazným způsobem se tak pomůže osobě obsluhující systém k rychlé orientaci. V případě osoby, která využívá počítač jen na nejdůležitější úkony, to platí dvojnásob. Rozmístění ovládacích prvků (např. tlačítko pro potvrzení změn) je vhodné umisťovat konzistentně a na očekávaných místech. Tedy pod upravovanou tabulkou, respektive textem. Všechna tato opatření by měla vést ke snížení pravděpodobnosti chybného úsudku ohledně toho, jaký text na stránce je právě upravován, a jakým způsobem změny uložit.

3.5.2 Systém rezervace jízd

Pro grafický návrh systému rezervace jízd platí stejné principy, jako pro grafický návrh redakčního systému, s tím rozdílem, že pro každou skupinu uživatelů (žáci, učitelé, administrátor) je třeba návrh uzpůsobit tak, aby odrážel její potřeby a usnadnil způsob ovládání, který u konkrétního uživatele převažuje. V praxi to znamená, že je nutné zohlednit více ovládacích prvků v části systému pro administrátora, a adekvátně tomu uzpůsobit rozložení prvků a velikost stránky. Pro žáky autoškoly zase platí, že je možné v nezanedbatelné míře očekávat přístup do rezervačního systému skrz chytré telefony a tablety – je proto třeba



Obrázek 3.2: Drátový model redakčního systému.

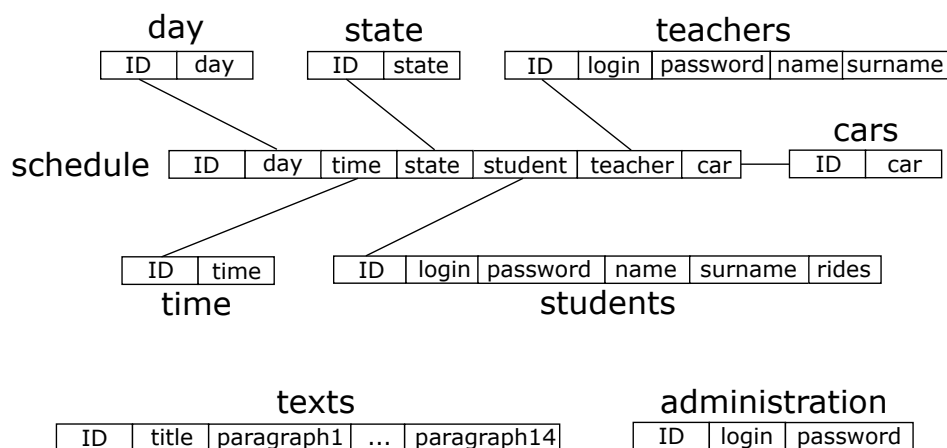
uzpůsobit ovládací prvky (především velikost) tak, aby byl systém snadno ovladatelný i na malé obrazovce chytrého telefonu. Pro skupinu učitelů je na místě zohlednit fakt, že někteří jedinci dosahují předdůchodového věku, a je tak nutnost počítat s možným zhoršeným zrakem, než je tomu u předchozích dvou skupin. Je tedy dobré zvolit spíše větší, a snadno čitelné (nezdobené) písmo, stejně jako ovládací prvky.

3.6 Ukládání a přenos dat

Vzhledem k požadavku na možnost změny textů a tabulek webové prezentace vzniká potřeba tato data uložit do databáze. Pro účely této aplikace naprosto postačuje využití databáze MySQL, která je k dispozici na serveru, který v současné době poskytuje autoškole hosting pro jejich webové stránky. K přenosu dat mezi uživatelem a serverem je možné využít množiny funkcí jazyka PHP pro práci s MySQL, avšak z požadavku na interaktivnost aplikace jasně vyplývá, že pouze s touto technologií si nelze vystačit. Pokud pomocí PHP odesíláme změněná data na server, změny se projeví až po aktualizaci stránky. Toto řešení příliš interaktivní není, je tedy třeba využít principů technologie AJAX a možností knihovny jQuery, která práci s AJAXem značně usnadňuje.

3.6.1 Komunikace se serverem – AJAX

Řešení komunikace klient-server pomocí AJAXu umožňuje dodat systému prvky interaktivity, které značně usnadní práci s ním. Technologie AJAX je jisté sjednocení technologií pro tvorbu interaktivních webových stránek, konkrétně se jedná o technologie HTML (XHTML), DOM model, XML, XMLHttpRequest objekt a Javascript [7].



Obrázek 3.3: Schéma tabulek databáze.

Použití AJAXu se hodí především v situacích, kdy potřebujeme změnit pouze určitou část obsahu webové stránky, a současně chceme dál pracovat s jejím rozhraním – chceme se vyhnout obnově stránky. Využití této funkcionality v navrhovaném informačním systému je rozsáhlé – od změny a ukládání textů a tabulek v redakčním systému, přes správu uživatelů systému v administraci, až po samotnou práci s rozvrhem, při jehož plánování (například přihlašování učitelů na jízdy) by neustálé obnovování stránky po přihlášení každého jednoho učitele nepříjemně prodlužovalo čas touto aktivitou strávený. Takový systém by příliš uživatelsky přívětivý nebyl, jelikož by všechny akce trvaly nepoměrně déle.

3.6.2 Struktura tabulek v databázi

Návrh struktury tabulek v databázi je jedním z kritických bodů celé aplikace. Vhodný návrh může pozdější implementaci systému usnadnit, zatímco špatný návrh velmi znepříjemní, až znemožní.

V navrhovaném informačním systému bude potřeba tabulka pro uložení všech textů webové prezentace, dále tabulky pro uložení všech tabulek z webové prezentace, a nakonec skupina tabulek umožňující uchovávat veškerá data týkající se rezervačního systému. Konkrétně bude třeba navrhnout a vytvořit tabulky pro uložení informací o všech skupinách uživatelů – žáci, učitelé, administrátoři. Dále je třeba vytvořit tabulky časů jízd, dnů jízd, stavů jízd a aut. Informace ze všech těchto tabulek potřebných v rezervačním systému budou následně použity pro naplnění tabulky rozvrhu. Strukturu a propojení tabulek databáze demonstruje obrázek 3.3.

Tabulky pro webovou prezentaci

Jedná se o tabulky zobrazené na webových stránkách, jejichž schéma odpovídá návrhu přejatého z původních stránek autoškoly. Jejich strukturu tedy není třeba více komentovat, nicméně jejich převedení do databázové podoby bylo nezbytné kvůli požadavku na možnost editace počtu záznamů (řádků) a textů v tabulkách.

Tabulky dat uživatelů systému

Tyto tabulky slouží pro uložení dat všech uživatelů systému, konkrétně se jedná o tabulky „administration“, „students“ a „teachers“. U všech typů uživatelů je třeba ukládat informace

o loginu (jednoznačné přihlašovací jméno) a heslu. Heslo je nutné ukládat zašifrované. Pro potřeby tohoto systému postačuje hašovací funkce sha1. Pro tabulku učitelů a žáků navíc uložíme jejich jméno a příjmení, jelikož práce s pouhým loginem by byla v systému velmi nepřehledná. Tabulka žáků navíc musí obsahovat ještě položku „rides“, která značí počet jízd zaregistrovaných daným žákem. Maximální počet jízd každého žáka je 14. Toto číslo se již řadu let nemění, proto není nezbytně nutné umožňovat jeho editaci v rezervačním systému.

Pomocné tabulky rozvrhu

Kromě tabulek žáků a učitelů spolupracují na naplnění tabulky rozvrhu ještě čtyři další tabulky. Konkrétně tabulka „cars“, která obsahuje auta, pro která se plánuje rozvrh. Počet záznamů této tabulky tedy určuje počet aut, pro která se rozvrh plánuje. Autoškola má auta očíslovaná, v systému jim proto není nutné (ani vhodné) přiřazovat jména, například podle tovární značky a modelu vozu. Autoškola totiž využívá více stejných modelů. Jejich odlišení podle čísel je tedy nejvhodnější. Kdyby však v budoucnu bylo v plánu tento systém přepracovat, stačilo by v tabulce „cars“ zaměnit jména (respektive čísla) aut za jejich nová (jednoznačná) označení, a následně změnit funkci pro přidání nového auta tak, aby se novému vozu nepřizovalo automaticky číslo, ale nové označení. Poté by rozvrh jízd by standardně fungoval dál. Proto je tabulka aut zcela samostatná, namísto přímého vepisování čísel aut do rozvrhu, i když se může zdát kvůli jejímu obsahu jako nadbytečná.

Další tabulkou nezbytnou pro tvorbu rozvrhu je tabulka „state“. Ta slouží pro uložení stavů jízd. Rozlišujeme 3 stavy:

- „empty“ – jízda je volná, je možné se na ni registrovat,
- „full“ – jízda je obsazená, není možné se na ni registrovat,
- „unavailable“ – jízda je zablokována, není možné se na ni registrovat (odblokovat ji může pouze administrátor).

Poslední dvě tabulky, které kompletují všechny nezbytné informace pro vytvoření tabulky rozvrhu jsou tabulky „days“ a „time“. První zmíněná tabulka slouží pro uložení názvů dní, na které se plánují jízdy, druhá tabulka slouží pro uložení časů, ve kterých probíhají jízdy. Opět je třeba poznamenat, že existence obou tabulek zaručuje snadnou změnu v plánování jízd bez nutnosti zcela přepisovat celý rozvrh.

Pokud uvažujeme plánování jízd na týden dopředu pro 6 aut (současný počet aut autoškoly), dostáváme pro 5 pracovních dní a 9 časových úseků celkem 270 jízd, které je možné v jednom týdnu naplánovat. Je snad zřejmé, že kdyby bylo třeba posunout čas všech jízd o půl hodiny, bude přepisování devíti záznamů v jednoduché tabulce méně časově náročné, než přepisování 270 záznamů v rozsáhlé tabulce.

Tabulka rozvrhu

Tabulka „schedule“ spojuje data tabulek „days, time, state, students, teachers“ a „cars“ do jedné velké tabulky. Z těchto dílčích tabulek se použijí pouze jednoznačné identifikátory, které definují položku dílčí tabulky. Tak se zajistí snadná úprava jakékoli hodnoty dílčí tabulky bez nutnosti přepisovat celý rozvrh, jelikož se změní pouze hodnota položky, nikoliv její identifikátor. Jízda je jednoznačně definována dnem, časem a autem.

3.6.3 Přenos dat formátem JSON

JSON (JavaScript Object Notation) je formát pro výměnu dat [5]. Oproti obecně dobře známému formátu XML je znatelně úspornější. Přenos dat pomocí formátu JSON je pro použití v této aplikaci velmi vhodným řešením v kombinaci s přenosem dat pomocí technologie AJAX. Jestliže provádíme dotaz nad databází, zpravidla kvůli aktualizaci hodnoty nějakého elementu na stránce, potřebovali bychom data vrátit nejlépe ve formátu „prvek:hodnota“, abychom mohli jednoznačně určit, který prvek na stránce máme aktualizovat získanou hodnotou. Přesně toto umožňuje formát JSON. Například, pokud výsledek dotazu nad databází (získaný ve formátu 1D nebo i 2D pole) převedeme na JSON řetězec, může výstup vypadat například takto:

```
[{"day": "1", "time": "1", "car": "1", "state": "3", "student": "1", "teacher": "0"}]
```

Pak víme, že:

- {"day": "1"}
– značí, že máme aktualizovat rozvrh pro pondělí,
- {"time": "1"}
– znamená začátek jízdy v 6:30,
- {"car": "1"}
– jízda proběhne v autě č. 1,
- {"state": "3"}
– říká, že je jízda obsazená,
- {"student": "1"}
– na jízdu je nahlášený student s identifikačním číslem 1,
- {"teacher": "0"}
– jízdě nebyl doposud přiřazen učitel.

Přičemž překlad identifikátorů na jejich skutečné hodnoty se provede následnými dotazy nad pomocnými tabulkami. Je proto zřejmé, že formát JSON je pro přenos databázových dat silnou technologií, která značně usnadní přenos databázových dat mezi serverem a klientem.

3.7 Kritické prvky aplikace z pohledu obsluhy

Kritickými prvky aplikace z pohledu obsluhy v kontextu této aplikace rozumíme takové prvky, které způsobí nevratnou změnu systému. Typicky se jedná o mazání jakýchkoli záznamů. Takovou operaci je nutno chránit před nechtěným spuštěním, a to takovým způsobem, který uživatele příliš nezdržuje. Důležitá je ale také prevence – tedy dát uživateli dostatečným způsobem najevo, že daná operace může způsobit nevratnou změnu systému, ještě než k spuštění dané operace dojde v reakci na kliknutí na tlačítko uživatelem. Pro příklad mazání je vhodné použití výstižných symbolů, které se nějakým způsobem zvýrazní (obarvení, zvětšení), jestliže se nad nimi nachází kurzor – tedy když se uživatel chystá operaci provést. Pro akci mazání se jako výstižné jeví použití červeného křížku, jehož intenzita

barvy se po najetí myši zvýší, a současně se objeví popisek, který oznamuje operaci, která po případném kliknutí nastane. Může se však stát, že uživatel přesto na zástupný symbol klikne, aniž by to zamýšlel. Proto je dobré vyžádat si od uživatele ještě potvrzení akce, například pomocí JavaScriptového modálního okna „prompt“. Pravděpodobnost, že uživatel omylem klikne na symbol mazání, a současně ještě poté akci omylem potvrdí, je malá, ačkoli ne nulová. Výrazně se však riziko nechtěného smazání položky snižuje.

Kapitola 4

Implementace

Tato kapitola se věnuje samotné implementaci informačního systému. Nejprve bude popsán vývojový cyklus, následně principy komunikace na straně serveru a klienta, poté budou popsány principy fungování důležitých logických celků aplikace. Konkrétně bude vysvětlen princip fungování přihlašování do systému, operace týkající se redakčního systému a správy uživatelů, a nakonec správa rozvrhu a přihlašování na jízdy pro žáky autoškoly. Poslední část této kapitoly se věnuje pouze rámcovému popisu implementace grafického rozhraní informačního systému a webové prezentace, kde budou popsány postupy dosažení poutavého a účelného vzhledu webové prezentace pomocí responzivního designu a interaktivních prvků.

4.1 Vývojový cyklus

Pro vývoj tohoto informačního systému byl použit klasický vodopádový model [3]. Známým problémem tohoto vývojového modelu je neschopnost uživatele na začátku přesně specifikovat všechny požadavky, proto bylo nutné klást velký důraz na podrobnou specifikaci požadavků před samotným návrhem systému. Nicméně hlavní funkcionality systému byla zcela jasná již od samého počátku vývoje, takže se neschopnost uživatele přesně definovat všechny požadavky projevila v přijatelné míře, která se odhalila při samotném testování aplikace a nepředstavovala žádný vážnější problém. Objevené nedostatky systému byly spíše charakteru implementačních detailů, které nijak zpětně neovlivnily jádro systému.

Ačkoli je vodopádový model nejjednodušším modelem vývoje softwaru, pro tento projekt byl dostačující díky znalosti rizik tohoto modelu, podrobné specifikaci požadavků, a také díky předvídatelným požadavkům na systém, které zadavatel nebyl schopen specifikovat, ale bylo možné je odvodit.

4.2 Principy komunikace klient-server

Komunikace mezi klientem a serverem probíhá, jak již bylo naznačeno v kapitole 3, pomocí funkcí knihovny jQuery, která značným způsobem zjednodušuje práci s AJAXem. V následujícím textu bude vysvětlen princip komunikace, který se v aplikaci využívá nejčastěji, na příkladu odregistrace jízdy žákem autoškoly.

Uvažujme, že víme, kterou jízdu chce žák odregistrovat – máme tedy informace o dni, času a autě, pro které byla jízda registrovaná. Nyní potřebujeme tyto informace poslat do php skriptu, který na základě parametrů a přijatých dat zajistí přepsání položky v databázi.

Tento požadavek vypadá následovně:

```
$.post("schedule_api.php", {action: 'unregister', time: time,
car: car, day: day}, function(data)
{
    if(data.status == "ok")
    {
        // zmenit podbarveni bunky rozvrhu
        // zmenit css tridu bunky rozvrhu
        // zobrazit moznost prihlaseni na vsechny volne jizdy
        // zobrazit potvrzujici hlasku o uspesne odregistraci
    }
    else
    {
        // zobrazit hlasku o nepovedene akci
    }
}, 'json');
```

Data tedy posíláme pomocí metody „post“. Proměnná „action“ slouží pro identifikaci prováděné akce v php skriptu „schedule_api“. Tento skript sdružuje operace pro všechny možné akce, které lze s rozvrhem provádět – od registrace a odregistrace, přes pomocné funkce pro kontrolu počtu jízd až po mazání jízd. Proměnné „time, car, day“ slouží pro identifikaci jednoho konkrétního záznamu, který chceme měnit.

V php skriptu poté budou v poli „post“ uložené všechny výše zmíněné proměnné, podle kterých spustíme provádění kódu pro manipulaci s databází. Nejprve se zkontroluje, jestli je časový odstup mezi odregistrací a jízdou alespoň 24 hodin, a pokud ano, provede se odstranění záznamu o žákovi z databáze. Jestliže i dotaz nad databází uspěje, dekrementujeme žákovi počet jízd a vrátíme informaci o úspěšném provedení celé operace. Tuto informaci zachytíme na straně klienta a můžeme ji následně pomocí „function(data)“ zpracovat a rozhodnout o provádění dalšího kódu.

4.3 Implementační principy programových celků

4.3.1 Přihlašování do systému

Přihlašování do systému je jedním z kritických bodů aplikace, je proto nutné mu věnovat zvláštní pozornost. Uživatel, který se nepřihlásí, nesmí mít možnost se systémem jakkoli manipulovat.

Princip autentizace uživatele spočívá ve vyhledání zadaného uživatelského jména v databázi, a následném porovnání otisku (haše) hesla uloženého v databázi s otiskem, který uživatel zadal do přihlašovacího formuláře. Uživatelé jsou v systému rozděleni do tří tabulek – „administration, students, teachers“, je tedy nutné vyhledávat zadaný login postupně ve všech tabulkách, dokud nenarazíme na shodu. Jistá optimalizace tohoto procesu spočívá v prohledávání tabulky administrátorů, která obsahuje nejméně záznamů, jako první. Následně se vyhledává v tabulce učitelů, která obsahuje řádově jednotky až nízké desítky záznamů, a nakonec se prohledává tabulka studentů, u které se předpokládá nárůst záznamů až na několik stovek. Velikost této tabulky silně závisí na iniciativě administrátora ze systému odstraňovat absolventy autoškoly, nicméně v každém případě je tato tabulka

nejobsáhlejší, je tedy vhodné ji prohledávat až jako poslední.

Důležité je také zmínit nutnost ošetření formulářových dat proti tzv. „SQL Injection“ [1], jelikož každý vstup od uživatele je potenciálně nebezpečný. Navíc, stránka s přihlášením je dostupná všem návštěvníkům. Více o této problematice obsahuje kapitola „Testování a bezpečnost“.

Jestliže autentizace uživatele proběhne korektně, tedy souhlasí otisk zadaného hesla a hesla z databáze příslušící zadanému loginu, je uživatel přesměrován na odpovídající stránku, která přísluší právům skupiny, do které patří. Administrátor je přesměrován do administrace, žák na zobrazení rozvrhu s možností přihlašování na jízdy, a učitel na stránku s rozvrhem jeho jízdy.

4.3.2 Úprava textů a tabulek

Úprava textů a tabulek náleží pouze administrátorovi. Rozhraní pro úpravu bylo navrženo s ohledem na intuitivnost ovládání, je proto implementováno tak, aby se co nejvíce podobalo webové prezentaci. Namísto textů se nachází tzv. „textareas“, tedy textová pole, jejichž velikost je možno dynamicky změnit tak, aby bylo možné zobrazit celý text, v tomto poli obsažený. Usnadní se tím orientace v textu, a tím se urychlí proces editace jako takové. Tabulky se zobrazují také téměř stejně, jako ve webové prezentaci, navíc jsou pouze sloupce s tlačítky pro potvrzení editace položky, případně její smazání.

Všechna textová pole a tabulky jsou na stránce vypsány pomocí dotazu nad databází, který je naplní daty z příslušné tabulky. V této fázi jsou současně veškerá vypsána data označena jednoznačnými identifikátory, které definují element obsahující data. Tato identifikace je nezbytná pro pozdější manipulaci s daty. Veškeré změny v textech a tabulkách se dějí interaktivně ve smyslu, že není třeba obnovovat stránku. Tato skutečnost značným způsobem urychluje správu textů a tabulek, a celkově zvyšuje pohodlí užívání redakčního systému. Děje se tak za pomoci AJAXové komunikace, která byla popsána výše, v části „Principy komunikace klient-server“. V principu se po stisknutí tlačítka „Uložit změny“ zavolá funkce, která zpracuje všechny hodnoty textových polí z dané části stránky (pomocí jejich identifikačních čísel), a odešle je ve formátu JSON do odděleného php skriptu, který se stará o zpracování (dekódování) přijatých dat a jejich uložení do příslušné tabulky v databázi. Operace s daty se neomezuje pouze na načtení (dotaz SELECT) nebo vložení (dotaz INSERT) do databáze – data je možné i upravit (dotaz UPDATE), nebo smazat (dotaz DELETE). Skript, který se stará o zpracování přijatých dat je typicky soubor funkcí, které se volají z rozhodovacího mechanismu „switch“, který volá obslužné funkce na základě parametrů. Jestliže se operace provede úspěšně, zobrazí skript příslušnou návratovou hodnotu, na základě které se vyvolá zobrazení potvrzujícího hlášení na stránce, ze které byl php skript zavolán. Neúspěch operace má za následek zobrazení chybového hlášení totožným principem.

4.3.3 Vyhledávání

Vyhledávání se v aplikaci objevuje na více místech, pokaždé v mírně odlišné podobě. Jedná se o následující případy:

- Vyhledávání uživatele na stránce správy žáků/učitelů – funguje jako dotaz nad databází, který prochází danou databázovou tabulku a hledá podobnost se jménem, příjmením nebo loginem. Výsledkem tohoto vyhledávání je barevné označení všech odpovídajících záznamů v tabulce výpisu žáků/učitelů.

- Vyhledávání loginu žáků v tabulce rozvrhu – automaticky nabízí výpis záznamů odpovídajících hledanému výrazu, za pomoci funkce „autocomplete“ knihovny jQuery. Seznam, ve kterém se vyhledává, je aktuální seznam všech žáků registrovaných v systému. Jestliže je navrhovaný výpis výsledků prázdný, daný žák v systému neexistuje.
- Vyhledávání učitelů v tabulce rozvrhu – rovnou nabízí seznam všech učitelů autoškoly. Nejedná se o vyhledání v pravém slova smyslu, jelikož neprobíhá žádná selekce záznamů odpovídajících hledanému výrazu, ale rovnou se nabízí všechny dostupné možnosti, tedy všichni učitelé registrovaní v systému. Toto řešení umožní rychlé přihlašování učitelů na jízdy, jelikož se jejich počet pohybuje přibližně okolo deseti. Administrátor nemusí do pole psát celé jméno nebo login učitele, pouze vybere ze seznamu.

4.3.4 Správa uživatelů

Podmnožina systému starající se o správu uživatelů funguje na podobných principech, které byly naznačeny v předchozí části textu. Opět platí, že data, která uživatel systému manipuluje, jsou zpracovávána JavaScriptovými funkcemi a následně posílána do php scriptu, který tvoří jisté rozhraní mezi systémem a databází.

Vložení nového uživatele probíhá totožným způsobem. Data z formuláře jsou převedena do formátu JSON a odeslána do php skriptu přes pomocnou funkci knihovny jQuery, kde se dále zpracovávají. Především se kontrolují veškerá integritní omezení. Jestliže dojde k chybě, je na ni uživatel upozorněn zobrazením odpovídajícího chybového hlášení, které konkrétně specifikuje povahu chyby. Formulace „Chyba dat ve formuláři.“ nenese potřebnou informační hodnotu pro specifikaci chyby. Z tohoto důvodu se rozlišují chyby následujících typů:

- některý z údajů je příliš dlouhý,
- zadaná hesla se neshodují,
- některé pole nebylo vyplněno,
- login musí být číslo,
- zadaný login již v databázi existuje u jiného uživatele,
- chyba spojení s databází.

Vyhledání uživatele a editace tabulky uživatelů také fungují na principech, které již byly uvedeny v předchozí části textu.

4.3.5 Správa rozvrhu

Správa rozvrhu je nejkomplexnější částí systému. Celý rozvrh je nejprve vypsán pro implicitní den (pondělí), přičemž každá buňka tabulky (rozvrhu) má své jednoznačné identifikační číslo. Celkem lze s rozvrhem provádět tři, respektive osm základních operací:

- zobrazit rozvrh pro následující/předchozí den,
- registrovat/odregistrovat žáka/učitele na jízdu,
- zablokovat/odblokovat jízdu.

Zobrazení rozvrhu pro následující den zajišťuje JavaScriptová funkce, která nejprve zjistí, jaký je aktuální den, na základě toho určí den následující (pro pátek je následující den v kontextu rozvrhu pondělí) a následně se pošle dotaz na php skript, který komunikuje s databází. Tento skript vrátí data z rozvrhu pro následující den ve formátu JSON, které je poté na straně klienta nutno zpětně dekodovat. Pro každý jeden záznam v rozvrhu se podle získaných dat přepíše stav jízdy a osoby, které jsou na ni registrovány. Pro zobrazení rozvrhu předcházejícího dne je situace analogická.

Registrování žáka/učitele začíná tím, že načteme data, která obsahuje vstupní pole, které editujeme. Současně uložíme identifikaci upravované buňky tabulky, přeložíme jméno z upravované buňky na ID, a nakonec uložíme informaci o dni, v němž se nachází jízda, kterou editujeme. Jestli se jedná o žáka nebo učitele poznáme z parametru, se kterým byla funkce zavolána. V tuto chvíli máme všechny potřebné informace pro aktualizaci záznamu. Data odešleme do php skriptu, který je dále zpracovává a odesílá dotaz na databázi. Jestliže vše proběhne v pořádku, je o této skutečnosti uživatel informován pomocí animace zeleného podbarvení upravovaného políčka. Jestliže jsme registrovali žáka a jízda byla předtím volná nebo blokována, stane se automaticky obsazenou. Při odregistraci žáka zase dochází k automatickému překlasifikování jízdy na volnou. Přihlášení učitele má na stav jízdy vliv pouze v případě, kdy byla předtím zablokována. V takovém případě se jízda stává volnou. Odregistrace učitele nemá na stav jízdy vliv.

Blokování/odblokování jízdy je také postavené na již zmíněných principech popsaných v (od)registraci žáka/učitele. Je však důležité zmínit, že zablokování jízdy, na kterou je někdo nahlášený, automaticky způsobí odhlášení těchto osob z jízdy. Na tuto skutečnost je administrátor předem upozorněn pomocí dialogového okna.

4.3.6 Přihlašování na jízdy – žáci

Přihlašování na jízdy je vázáno několika zásadními pravidly, které nesmí být při implementaci opomenuty. Především se jedná o časový limit (od)registrací jízd, který je pevně stanoven na 24 hodin. Dále možnost pouze jedné jízdy v jeden den, maximálně tři jízdy týdně, a celkově přesně 14 jízd, které definují rozsah standardního kurzu autoškoly.

Princip fungování rozvrhu pro žáky je velmi podobný principu, který byl popsán výše, v části „Správa rozvrhu“. Přepínání dní je zcela totožné, liší se pouze množina operací, které nad rozvrhem může uživatel provádět. V případě žáka se jedná pouze o změnu položky „student“ v databázové tabulce s rozvrhem, na základě výše definovaných pravidel. Žák si zvolí den, čas a auto, pro které chce jízdu registrovat, a jestliže splňuje všechny podmínky, jízda je mu připsána. Současně se zablokuje registrace ostatních jízd pro daný den. Pokud žák své stanovisko přehodnotí a jízdu si vzápětí odregistrová, opětovně se zobrazí možnost přihlášení na všechny zbývající volné jízdy ve zvoleném dni.

Celý rozvrh pracuje v JavaScriptovém prostředí, všechny změny se tedy projeví ihned, bez nutnosti obnovení stránky.

4.4 Grafické rozhraní systému a webové prezentace

Grafické rozhraní systému a webové prezentace odpovídá návrhu z kapitoly „Analýza a návrh“. Důraz byl kladen na přehlednost v kombinaci s moderním vzhledem v případě webové prezentace. U redakčního a rezervačního systému převládá potřeba přehlednosti a jednoduchosti – systém má být především funkční a jednoduchý na obsluhu, nikoliv nepřekonatelně krásný a prakticky nepoužitelný.

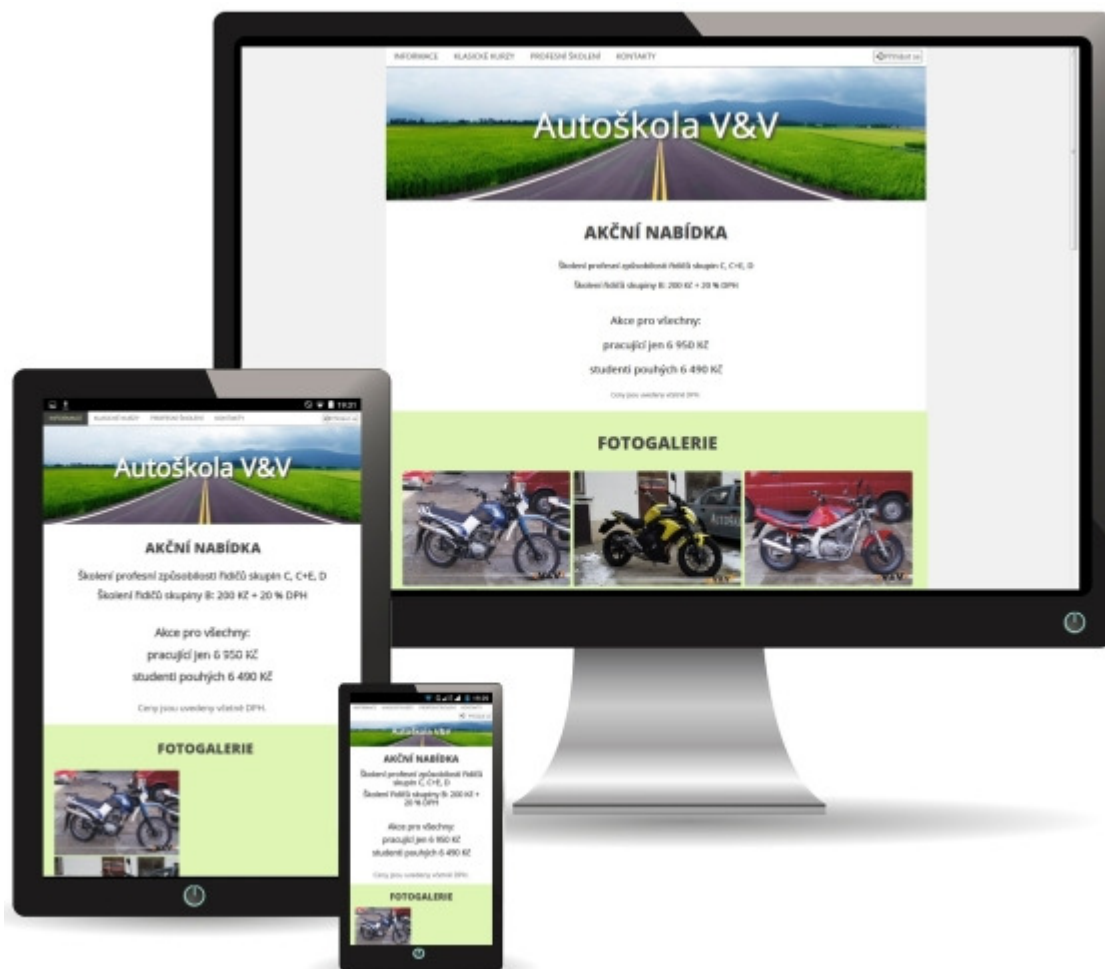


Obrázek 4.1: Zobrazení přihlašování na jízdy na různých velikostech obrazovek.

K dosažení výše zmíněného bylo využito dvou zásadních principů – responzivního designu, a použití interaktivních prvků za využití knihovny Bootstrap (responzivní design, stylování stránky) a knihovny jQuery (animace).

4.4.1 Responzivní design

Přizpůsobování stránky velikosti obrazovky je vidět na obrázcích 4.1 a 4.2. Dochází k zalomení obsahu a zvětšení písma tak, aby bylo bez problémů čitelné i na malé obrazovce chytrého telefonu. Mimo webovou prezentaci je vzhled stránky optimalizován pro chytré telefony i v části systému pro žáky a učitele autoškoly. Část systému určená administrátorovi je také responzivní, její optimalizace pro manipulaci na chytrém telefonu však není nutná, jelikož administrátor bude systém obsluhovat pouze prostřednictvím stolního počítače, který má po celý pracovní den k dispozici ve své kanceláři.



Obrázek 4.2: Zobrazení webové prezentace na různých velikostech obrazovek.

4.4.2 Interaktivní prvky

Použití interaktivních prvků se nevztahuje pouze na informační systém. Webová prezentace je jimi také obohacena tak, aby se návštěvníkovi usnadnila orientace na stránce. Konkrétně se jedná o barevné zvýraznění aktivní záložky (stránky), a dále o změnu podbarvení odkazů při najetí myši nad tento odkaz. Tyto interaktivní prvky nejsou příliš výrazné, nicméně u webové prezentace je důležité orientaci v ní uživateli zjednodušit, nikoli znepříjemnit použitím nadbytečného množství animací.

V rezervačním a redakčním systému je interaktivních prvků více. Především se jedná o chybová a potvrzující hlášení všeho druhu, která se pomocí animace zobrazí na čas nezbytně nutný k jejich přečtení, jestliže dojde k události, která je vyvolá. Jistým interaktivním prvkem je také barevné orámování aktivního políčka formuláře a umístění blikajícího kurzoru v něm. Dobrý příklad interakce s uživatelem je při zadání špatného hesla při přihlašování. Jestliže nastane taková situace, zobrazí se chybové hlášení upozorňující uživatele na nesprávně zadané heslo, a současně se do formuláře pro přihlašování doplní login (jestliže byl zadán správně), a druhé políčko pro zadání hesla se barevně zvýrazní (orámování) a umístí se do něj kurzor. Uživatel nemusí vůbec sahat po myši, pouze si přečte chybové hlášení, zjistí problém, znovu napíše heslo a potvrdí odeslání nových dat klávesou „enter“.

Další interaktivní prvky jsou:

- orámování tlačítek při najetí myši,
- podbarvení textových vstupů při najetí myši,
- podbarvení upravovaného řádku tabulky a zobrazení potvrzujícího hlášení po uložení změn,
- zobrazení ikony načítání při přepínání dnů rozvrhu,
- okamžité změny obsahu a animace podbarvení buněk rozvrhu při jeho editaci.

Všechny tyto interaktivní prvky přispívají k jednodušší obsluze systému, jelikož uživatel dostává jistá potvrzení o reakci systému na jeho akce. V případě interaktivní úpravy rozvrhu je přínos nejen v jednodušší obsluze a orientaci, ale také v šetření času, jelikož se stránka nemusí po každé akci znovu načítat.

Kapitola 5

Testování a bezpečnost

5.1 Bezpečnost

Kontrola bezpečnosti aplikace je velmi důležitá, zejména kontrola přístupu do části systému, která je přístupná pouze administrátorovi. Jiný uživatel do administrace nesmí mít přístup, ani nesmí mít žádnou jinou možnost ovlivnit databázová data, ke kterým z role návštěvníka stránky nebo žáka autoškoly nemá přímý přístup. Jmenovitě se jedná o známý problém, tzv. „SQL injection“, kdy útočník pomocí formuláře pošle data, která pozmění dotaz nad databází takovým způsobem, že ji mohou poškodit. Například do formuláře pro změnu hesla vepíše:

```
{nové_heslo;DROP TABLE uzivatele; --';}
```

Jestliže se pak v databázi skutečně nachází tabulka „uzivatele“, bude nenávratně smazána, jelikož doplněním tohoto řetězce do formuláře se uspokojí SQL dotaz pro přepsání hesla, středník dotaz ukončí, a začne se provádět následující dotaz, tedy smazání tabulky „uzivatele“.

Obrana proti SQL Injection je však vcelku jednoduchá – spočívá v ošetření vstupu od uživatele. Kritický je formulář pro přihlášení do systému, který je zcela dostupný všem návštěvníkům webových stránek. Řešení je jednoduché. Jediný vstup, který aplikace od uživatele přebírá pro použití v SQL dotazu je login (přihlašovací jméno), na základě kterého se v databázi vyhledá uživatel a jemu náležící otisk (hash) hesla, který se porovná s otiskem hesla, které uživatel napsal a odeslal formulářem. Vzhledem k tomu, že login je striktně ve formátu čísla (nebo roven řetězci „admin“), stačí kontrolovat tuto skutečnost. Jestliže se uživatel pokusí o SQL injection, aplikace řetězec nepropustí.

5.2 Předpokládaná zátěž

Nové kurzy autoškoly se otevírají přibližně každých 10 dní, přičemž do každého kurzu se může přihlásit až 20 žáků. Průměrná doba absolvování kurzu je přibližně 8 týdnů, to znamená, že současně studuje v autoškolě přibližně 90 žáků, jestliže uvažujeme průměrné naplnění kurzu ze 75 %.

Plánování rozvrhu na týden dopředu je tedy zcela dostačující, jelikož při počtu šesti aut (současný stav), je možné při devíti časových úsecích za den (po 1,5 hodině) a pěti pracovními dny, ve kterých jízdy probíhají, naplánovat 270 jízd. V jednom týdnu je tak možné naplánovat 3 jízdy pro všechny žáky autoškoly.

Pokud uvažujeme, že systém on-line přihlašování na jízdy bude využívat přibližně 80% žáků autoškoly, znamená to přibližně desítky až nízké stovky přístupů za den, které nemají šanci systém jakýmkoli způsobem zahltit.

Ani stoupající počet registrovaných žáků nepředstavuje zásadní problém na velikost dat v databázi. Pokud uvažujeme předchozí hodnoty, tak každoročně projde autoškolou přibližně 500 žáků, o kterých je třeba uchovávat informace. Nedá se předpokládat, že systém by beze změny (nebo údržby) běžel déle, než 5 až 10 let, z tohoto důvodu jsou i tato čísla naprosto zanedbatelná a nepředstavují zásadní datovou zátěž.

5.3 Testování

Možností testování systému je mnoho. Například pozorování práce uživatele se systémem, nebo vyplňování různých dotazníků s uzavřenými i otevřenými otázkami. Pro testování této aplikace se spíše hodí zadávání různých úkolů a následné pozorování uživatele při jejich řešení. Tímto způsobem je snadné zjistit, nad kterými akcemi uživatel déle přemýšlí, a mohly by tedy být více intuitivní, případně které akce uživatel vůbec nezvládne splnit.

Mimo testování schopnosti práce se systémem (intuitivní ovládní) je také třeba důkladně ověřit správnou funkčnost všech komponent systému. Zejména kritické je korektní ukládání všech změn systému do databáze. I jedna malá chyba může vést k naprosto nepředvídatelnému a chybnému chování aplikace jako celku.

Před samotným spuštěním aplikace v reálném provozu je vhodné takové podmínky co nejlépe nasimulovat. To znamená naplnění databáze pseudoreálnými daty (smyšlenými jmény) a následné provádění všech operací, které systém umožňuje, a paralelní kontrolování stavu databáze. Vzhledem k faktu, že systém reaguje na reálný čas (24 hodinové limity pro přihlášení a odhlášení jízdy), je nutné aplikaci testovat v ideálním případě celý týden, přestože je možné korektní chování odhadnout i na základě testování v průběhu jednoho dne.

Testování probíhalo bez sdělení jakýchkoli informací ohledně ovládní informačního systému. Testované osoby musely na postupy splnění úkolů přijít samy.

5.3.1 Přihlašování

S přihlašováním neměla žádná skupina uživatelů sebemenší problémy, nicméně někteří uživatelé projevili zájem o předvyplnění přihlašovacího jména v případě nesprávně napsaného hesla. Na základě tohoto požadavku byla daná funkce implementována. V případě překlepu při zadávání hesla již nebude muset uživatel zadávat znovu i uživatelské jméno, a celý proces přihlašování se při této chybě urychlí.

5.3.2 Správa žáků a učitelů

Správa žáků a učitelů je množina akcí, které může provádět pouze administrátor. Tato část testování byla prováděna pouze na dvou osobách, které budou mít k této části systému přístup. Úkoly byly zadávány pouze pro testování správy žáků, jelikož sekce týkající se správy učitelů je podmnožinou sekce správy žáků.

Úkoly, na kterých probíhalo testování této části systému byly následující:

- vložení nového žáka do systému,
- vyhledání informací o žákovi,

- úprava informací o žákovi,
- úplné odstranění informací o žákovi ze systému.

S výše popsanými úkoly neměly testovací subjekty žádný zásadní problém, nicméně byla vyslovena žádost o názornější potvrzení akce editace položky. Původně navržený systém reagoval na uložení změn v tabulce žáků pouze zeleným podbarvením řádku, který byl editován. Bylo proto přidáno hlášení s textovou informací, že se data povedlo úspěšně aktualizovat. Tato hlášení byla následně distribuována na podobná místa v systému, aby dávala jednoznačně najevo, že změna databázových dat byla úspěšná.

5.3.3 Změna textů a tabulek webových stránek

Změna textů a tabulek je vlastně veškerá funkcionalita redakčního systému, ke kterému má také přístup pouze administrátor, přičemž změna obsahu tabulek je založena na naprosto stejných principech, jako tabulky výpisu žáků a učitelů. Logicky ani zde nenastal žádný problém v manipulaci s těmito tabulkami.

Editace textů také proběhla bez jakýchkoli potíží. Administrátoři celkově oceňovali snahu o co nejpodobnější vzhled redakčního systému vzhledem k webové prezentaci. Díky této podobnosti jim orientace v redakčním systému nedělala žádné problémy a byli schopni velmi rychle vyhledat text, který měli za úkol upravit.

5.3.4 Správa rozvrhu

Testování schopnosti administrátora obsluhovat správu rozvrhu je zásadní. Pokud administrátor špatně upraví nějaký text na stránce, je možné chybu vcelku snadno odhalit a napravit. Pokud však dojde k omylu při plánování jízd (obsluze rozvrhu), můžou být následky mnohem vážnější. Z tohoto důvodu je nutné věnovat testování schopnosti administrátora spravovat rozvrh dostatečnou pozornost. Zadané úkoly byly následující:

- přihlášení libovolného učitele na jízdu ve středu, 9:00, auto č. 2,
- odhlášení žáka z jízdy v libovolný den a hodinu,
- zablokování všech jízd pro auto č. 1 v pátek,
- přihlášení žáka a učitele na zablokovanou jízdu,
- rozšíření plánování rozvrhu pro další auto.

Přihlášení učitele na jízdu proběhlo bez jakýchkoli komplikací. Odhlášení žáka z jízdy v libovolný den a hodinu proběhlo taktéž bez komplikací, ačkoli bylo zřejmé, že testovaná osoba váhá, jestli kliknutím na křížek, umístěný vedle jména žáka v rozvrhu například nedojde k úplnému vymazání žáka ze systému. V reakci na tuto nerozvážnost byl přidán popisek, který se zobrazí po najetí myši, a konzultována ikona představující odhlášení žáka z jízdy.

5.3.5 Práce se systémem – žák

Testování práce s rozvrhem jízd z pohledu žáků autoškoly probíhalo na vzorku osmi lidí různých věkových kategorií, kteří měli za úkol přihlásit se do rezervačního systému a následně registrovat dvě jízdy v jeden den, a ve zbývajících dnech pouze jednu jízdu. Sledovaly

se reakce uživatelů na chybová hlášení, která je upozorňovala na nemožnost provedení různých akcí – registrace jízdy méně, než 24 hodin dopředu, registrace dvou jízd ve stejný den, nebo registrace celkově více, než tří jízd za týden. Především se sledovalo, jestli má uživatel dostatek času chybová hlášení přečíst a také jim porozumět.

U některých chybových hlášení bylo zpozorováno, že pravděpodobně v důsledku jejich příliš krátkého zobrazení byli uživatelé nuceni opětovně provádět akce, které zobrazení hlášení způsobily. Na základě této skutečnosti byla následně doba zobrazení chybových hlášení upravena tak, aby byla doba zobrazení hlášení adekvátní délce a důležitosti sdělení. Změna hesla nepředstavovala pro žádného uživatele sebemenší problém.

Uživatelé oceňovali jasné a stručné instrukce k přihlašování jízd, naopak jim ale chybělo bližší vysvětlení jednotlivých stavů jízdy (volná, volná k přihlášení, obsazená jiným studentem, obsazená přihlášeným studentem, nedostupná). Na základě tohoto požadavku byla pod tabulku rozvrhu jízd přidána legenda, detailně vysvětlující všechny možné stavy jízd, které se v rozvrhu mohou vyskytnout. Tato legenda byla současně přidána i do stránky s rozvrhem pro učitele, jelikož se dal očekávat podobný návrh i od učitelů. Další připomínkou bylo zobrazování tlačítek pro přihlášení pro všechny volné jízdy, tedy i v případě, kdy si na daný den žák již jízdu registroval. Kliknutí na toto tlačítko pochopitelně vyvolalo chybové hlášení oznamující, že další jízdu již nelze v jednom dni registrovat. Někteří z testovaných žáků tuto možnost označili za matoucí a nadbytečnou, proto byla na základě této připomínky odstraněna všechna tlačítka pro přihlášení na volné jízdy, jestliže se žák v daný den na nějakou jízdu registruje. Jízdy jsou nadále označeny jako volné, aby si žák byl vědom toho, že odregistrováním své jízdy mu bude umožněno registrovat si jinou jízdu v daný den.

5.3.6 Práce se systémem – učitel

Testování schopnosti používat aplikaci z role učitele proběhlo na vzorku třech učitelů autoškoly různých věkových kategorií.

Učitelé dostali jediný úkol – změnit heslo. To se obešlo bez jakýchkoli komplikací a všichni učitelé úkol zvládli. Poté jim byl ponechán prostor pro orientaci v rozvrhu a případné námítky nebo dotazy. K samotnému vzhledu aplikace nebyly vzneseny žádné námítky, vyskytly se pouze dotazy ohledně doby, kdy je rozvrh aktuální (dále neměnný). Tato doba vyplývá ze zákazu (od)registrace jízd méně než 24 hodin před jejich uskutečněním, rozvrh pro následující den je tak postupně uzavíraný během předchozího dne, a zcela neměnný se stává po začátku poslední jízdy předešlého dne. Učitelé oceňovali přehlednost rozvrhu a barevné odlišení stavů jízd, které jim usnadňuje orientaci.

5.4 Testování aplikace v internetových prohlížečích

Aplikace byla testována v následujících internetových prohlížečích na desktopovém počítači a chytrém telefonu a tabletu:

- Mozilla Firefox verze 45.0 (desktop i smartphone/tablet),
- Google Chrome verze 49.0 (desktop i smartphone/tablet),
- Internet Explorer verze 11.0 (desktop),
- Opera verze 20.0 (desktop) a verze 36.1 (smartphone/tablet).

Webová prezentace i samotná aplikace se na všech zařízeních a velikostech obrazovek zobrazuje korektně, nedochází k žádnému překrývání prvků, nebo ořezávání textů. Zobrazení webové prezentace a aplikace na různých zařízeních (velikostech obrazovek) je vidět na obrázcích 4.1 a 4.2.

Kapitola 6

Závěr

Cílem práce bylo vytvořit webovou aplikaci pro on-line přihlašování na jízdy, dále redakční systém pro správu obsahu webové prezentace a v neposlední řadě vytvoření nové webové prezentace autoškoly. Vedlejším cílem této práce byl osobní rozvoj znalostí v oblasti problematiky návrhu a implementace informačních systémů a moderních webových prezentací. Tento cíl byl jednou z hlavních motivací pro volbu právě tohoto tématu práce.

Výsledkem práce je funkční webová aplikace a webová prezentace autoškoly, která má za cíl usnadnit žákům přihlašování na jízdy a osobě odpovědné za plánování rozvrhu ušetřit čas touto aktivitou strávený. Lze také předpokládat vyšší míru konkurenceschopnosti autoškoly po nasazení nové webové prezentace a rezervačního systému, jelikož on-line rezervace je velmi pohodlný způsob, jak si jízdy registrovat odkudkoli, kde je dostupné připojení k internetu, například i prostřednictvím chytrého telefonu, pro který je rezervační systém taktéž optimalizován. Vytvořená webová prezentace nahradí stávající webovou prezentaci, a rezervační a redakční systém bude také nasazen do reálného provozu. S nasazením rezervačního systému se počítá postupně, tedy nejprve bude spuštěno přihlašování na jízdy pro jedno auto, a postupně se bude přecházet na plánování rozvrhu pro všechna auta.

Aplikaci by bylo možné rozšířit například plánováním rozvrhu pro delší časový úsek, než jeden týden. V současné době je však týden dostačující doba a v nejbližších letech nelze předpokládat takový nárůst studentů autoškoly, aby bylo nutné k tomuto rozšíření přistoupit, vzhledem k obecně vysokému počtu autoškol v Brně, kde autoškola sídlí. Další možné rozšíření spočívá v rozesílání emailů žákům autoškoly o nastávající jízdě, například den předem, nebo případný sběr a vyhodnocení statistik o jízdách učitelů – například počet jízd za měsíc/rok/celkově.

Literatura

- [1] PHP: SQL Injection - Manual. ONLINE, 2016, [cit. 2016-5-5].
URL <http://php.net/manual/en/security.database.sql-injection.php>
- [2] W3Schools Online Web Tutorials. ONLINE, 2016, [cit. 2016-5-5].
URL <http://www.w3schools.com/>
- [3] Bohuslav Křena, R. K.: Úvod do softwarového inženýrství. FIT VUT v Brně, 2010, [cit. 2016-5-2].
URL https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/IUS-IT/texts/IUS_opora.pdf
- [4] Gutmansb Andi, R. D., Bakken Stig S.: *Mistrůství v PHP 5*. Computer Press, 2012, iISBN 978-80-251-1519-0.
- [5] Karl Swedberg, J. C.: *Mistrůství v jQuery*. Computer Press, 2013, iISBN 978-80-251-4103-8.
- [6] Pleskotová, P.: *Svět barev*. Albatros, 1987, praha.
- [7] Tomáš Hruška, A. K., Lukáš Máčel: Internetové aplikace (WAP) V. část AJAX. FIT VUT v Brně, 2012, [cit. 2016-4-28].
URL <https://www.fit.vutbr.cz/study/courses/WAP/private/opory/0poraWAP5Ajax.pdf>
- [8] Žára, O.: *JavaScript - Programátorské techniky a webové technologie*. Computer Press, 2015, iISBN 978-80-251-4573-9.

Přílohy

Seznam příloh

A	Obsah CD	36
B	Instalace	37
C	Manuál	38
C.1	Přihlášení	38
C.2	Rozvrh jízd	38
C.3	Editace webu	39
C.4	Žáci	39
C.5	Učitelé	40
C.6	Změna hesla	40
C.7	Žáci a učitelé	40
C.8	Odhlášení	40

Příloha A

Obsah CD

- /app – zdrojové soubory aplikace
- /doc
 - /src – zdrojové soubory technické zprávy
 - /doc.pdf – technická zpráva ve formátu PDF

Příloha B

Instalace

Skript pro inicializaci databáze a pro konfiguraci připojení k databázi je uložen v kořenové složce se zdrojovými soubory webové aplikace pod názvem „db_initialize.php“, respektive „config.php“.

Pro spuštění aplikace je nejprve potřeba všechny zdrojové kódy umístit na server, který podporuje databázi MySQL a jazyk PHP. Po nakopírování všech souborů je následně třeba spustit PHP skript „db_initialize“, který provede inicializaci tabulek databáze, a současně naplnění všech tabulek, které obsahují data zobrazovaná na webové prezentaci autoškoly, a také tabulek nezbytných pro běh systému. Ostatní tabulky („students, teachers, cars, schedule“) jsou implicitně prázdné. Poté stačí v konfiguračním souboru „config.php“ změnit přístupové údaje do databáze a aplikace je připravena k použití.

Příloha C

Manuál

C.1 Přihlášení

Stránka pro přihlášení je přístupná po kliknutí na tlačítko „Přihlásit se“, dostupné ze všech stránek webové prezentace, která je v době odevzdání této práce přístupná z adresy „<http://www.stud.fit.vutbr.cz/~xhalik01/VAV/>“. Pro první přihlášení do systému slouží implicitní přihlašovací údaje definované ve skriptu „db_initialize.php“:

- login: admin
- heslo: admin

Po přihlášení je uživateli přístupná administrace, se kterou může manipulovat pomocí šesti záložek – „Rozvrh jízd, Editace webu, Žáci, Učitelé, Auta, Změna hesla“.

C.2 Rozvrh jízd

Každá jízda (definovaná dnem, časem a autem) má barevné označení, kde:

- zelená značí, že je jízda volná,
- šedá znamená zablokovanou jízdu,
- červená značí obsazenou jízdu (je na ni přihlášený žák).

S každou jízdou je možné provádět následující operace:

- zablokování jízdy pomocí tlačítka „Blokovat“ – žákům se znemožní přihlášení na jízdu,
- odblokování jízdy pomocí tlačítka „Odblokovat“ – žákům se opětovně povolí přihlášení na jízdu,
- přihlášení žáka na jízdu – provádí se vepsáním loginu žáka do odpovídajícího pole a stisknutím ikony s popisem „registrovat žáka“,
- přihlášení učitele na jízdu – provádí se otevřením seznamu učitelů pomocí kliknutí na odpovídající pole, zvolením učitele ze seznamu a stisknutím ikony s popisem „registrovat učitele“,

- odhlášení žáka z jízdy – provádí se stisknutím ikony s popisem „odregistrovat žáka“ na odpovídajícím řádku,
- odhlášení učitele z jízdy – provádí se stisknutím ikony s popisem „odregistrovat učitele“ na odpovídajícím řádku.

Přihlášení žáka na zablokovanou jízdu způsobí její automatické odblokování a nastavení stavu jízdy na „obsazená“. Přihlášení učitele na zablokovanou jízdu také způsobí její odblokování a nastavení stavu jízdy na „volná“.

Přepínání dní se provádí stiskem odpovídající šipky vedle názvu dne nad tabulkou rozvrhu.

Další možnou operací v této části systému je „inicializace rozvrhu“. Stiskem tohoto tlačítka se rozvrh pro aktuální den přenastaví na výchozí hodnoty – každá jízda je volná, bez přihlášeného učitele i žáka. Slouží tedy pro uvolnění jízd pro nové plánování na následující týden. Posledním ovládacím prvkem této stránky je „Zobrazit rozvrh pro tisk“. Stisk tohoto tlačítka vyvolá otevření stránky s přehlednějším a kompaktnějším zobrazením rozvrhu, které může administrátorovi sloužit pro tisk rozvrhu, nebo pro kontrolu rozvrhu.

C.3 Editace webu

Editace textů se provádí prostým přepsáním textu v odpovídajícím textovém poli na dané stránce, vybrané z menu, a následným potvrzením uložení změn stisknutím tlačítka „Uložit změny“, které je umístěné na konci každého bloku textů. Editace obsahu tabulek umožňuje následující operace:

- smazání řádku tabulky,
- editace textu řádku tabulky,
- přidání nového řádku tabulky.

Smazání řádku se provádí stiskem ikony „smazat“. Editace textu tabulky se provede prostým přepsáním textu ve formulářovém poli a následným stiskem ikony „Upravit“. Editaci je třeba potvrdit pro každý řádek zvlášť. K přidání nového řádku slouží tlačítko „Vložit záznam“ umístěné v posledním řádku tabulky. K vepsání textů jednotlivých buněk nového řádku jsou připravena formulářová pole.

C.4 Žáci

Tato stránka slouží pro správu žáků autoškoly. Umožňuje přidání nového žáka, vyhledání žáka, nebo úpravu informací o žákovi v tabulce výpisu žáků. Přidání nového žáka se provádí vepsáním všech nezbytných informací do formulářového pole a potvrzením akce stiskem tlačítka „Přidat žáka“. V sekci vyhledávání je možno hledat žáka podle jména, příjmení, nebo loginu. Po odeslání dotazu pomocí stisku „Vyhledat“ se v tabulce výpisu žáků automaticky vyznačí všechny relevantní výsledky. Manipulace s tabulkou výpisu žáků funguje na stejných principech, jako editace tabulky webových stránek.

C.5 Učitelé

Stránka pro správu učitelů funguje zcela stejným způsobem, jako stránka pro správu žáků. Popis práce s ní tedy nebude více komentován.

Na této stránce je možné navolit počet aut, pro která se plánuje rozvrh, za pomoci tlačítek „+“ a „-“.

C.6 Změna hesla

V této záložce systému je možné měnit hesla všem uživatelům systému pomocí vepsání loginu a hesla do formuláře, a následným potvrzením akce stiskem tlačítka „Změnit heslo“.

C.7 Žáci a učitelé

Žáci pracují se systémem na podobných principech, jako administrátor, avšak nemají k většině akcí oprávnění. Žáci se přihlašují na jízdy stiskem odpovídajícího tlačítka v tabulce rozvrhu. Odhlášení funguje obdobně. Učitelé mohou pouze přepínat dny rozvrhu a prohlížet si jej. Změna hesla se provádí vepsáním nového hesla na odpovídající stránce systému a odesláním požadavku stiskem tlačítka pod formulářem.

C.8 Odhlášení

Odhlášení z aplikace se provádí stiskem příslušného tlačítka v pravém horním rohu stránky. Po odhlášení je uživatel přesměrován na úvodní stránku webové prezentace.