

Mendelova univerzita v Brně  
Provozně ekonomická fakulta

---

# **Evidenční systém pro podnikatele s viacerými podnikateľskými činnosťami**

**Bakalárska práca**

Vedúci práce:  
Ing. Jiří Lýsek, Ph.D.

Marek Valuch

Brno 2015



Týmto by som sa chcel poďakovať svojej rodine za podporu pri štúdiu a pri písaní bakalárskej práce. Taktiež sa chcem poďakovať vedúcemu práce Ing. Jiřímu Lýskovi, Ph.D. za pripomienky a návrhy.



Prehlasujem, že som túto prácu vyriešil samostatne s použitím literatúry, ktorú uvádzam v zozname.

V Brne dňa 19. mája 2015

.....



### **Čestné prehlásenie**

Prehlasujem, že som túto prácu: **Evidenčný systém pre podnikateľa s viacerými podnikateľskými činnosťami**

vypracoval samostatne a všetky použité pramene a informácie sú uvedené v zozname použitej literatúry. Súhlasím, aby moja práca bola zverejnená v súlade s § 47b zákona č. 111/1998 Sb., o vysokých školách v znení neskorších predpisov, a v súlade s platnou *Smernicou o zverejňovaní vysokoškolských záverečných prác*.

Som si vedomý, že sa na moju prácu vzťahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzatvorenie licenčnej zmluvy a použitie tejto práce ako školského diela podľa § 60 odst. 1 Autorského zákona.

Ďalej sa zaväzujem, že pred spísaním licenčnej zmluvy o využití diela inou osobou (subjektom) si vyžiadam písomné stanovisko univerzity o tom, že predmetná licenčná zmluva není v rozpore s oprávnenými záujmami univerzity, a zaväzujem sa uhradiť prípadný príspevok na úhradu nákladov spojených so vznikom diela, a to až do ich skutočnej výšky.

V Brne dňa 19. mája 2015

.....





**Abstract**

Valuch, M. System of evidence for businessman with multiple business activities. Bachelor thesis. Brno: Mendel University in Brno, 2015

This thesis is about designing and making web application for evidence employee, vehicle fleet, events, costs and revenues, while it is possible to evidence multiple companies in one system. Thesis describes designing and implementation of system in PHP language in nette framework.

**Keywords**

Web application, PHP, Nette Framework, evidence

**Abstrakt**

Valuch, M. Evidenčný systém pre podnikateľa s viacerými podnikateľskými činnosťami. Bakalárska práca. Brno: Mendelova univerzita v Brne, 2015

Táto práca sa zaoberá návrhom a tvorbou webovej aplikácie pre evidenciu zamestnancov, vozového parku, nákladov a výnosov, pričom aplikácia umožňuje evidenciu viacerých firiem zároveň. Práca popisuje návrh a implementáciu systému v jazyku PHP vo frameworku Nette.

**Klíčové slová**

Webová aplikácia, PHP, Nette Framework, evidencia



## Obsah

<b>1</b>	<b>Úvod a cieľ práce</b>	<b>13</b>
1.1	Úvod do problematiky . . . . .	13
1.2	Cieľ práce . . . . .	13
<b>2</b>	<b>Webové aplikácie</b>	<b>14</b>
2.1	Statické webové stránky . . . . .	14
2.2	Dynamické webové stránky . . . . .	14
<b>3</b>	<b>Webové technológie</b>	<b>15</b>
3.1	PHP . . . . .	15
3.1.1	História PHP . . . . .	15
3.1.2	Využitie jazyka PHP . . . . .	15
3.1.3	PHP a frameworky . . . . .	16
3.2	HTML . . . . .	16
3.3	CSS . . . . .	16
3.4	JavaScript . . . . .	17
3.4.1	jQuery . . . . .	17
3.4.2	AngularJS . . . . .	17
3.5	Databázové systémy . . . . .	17
3.6	SQL . . . . .	18
3.7	MySQL . . . . .	18
3.8	Oracle Database . . . . .	18
<b>4</b>	<b>Súčasný stav</b>	<b>20</b>
4.1	Súčasný stav . . . . .	20
4.2	Požiadavky majiteľa . . . . .	20
4.2.1	Funkčné požiadavky . . . . .	21
4.2.2	Nefunkčné požiadavky . . . . .	22
4.2.3	Používateľské role . . . . .	23
4.3	Existujúce systémy . . . . .	23
4.3.1	Potreba návrhu nového systému . . . . .	25
<b>5</b>	<b>Návrh riešenia</b>	<b>26</b>
5.1	Návrh grafického používateľského rozhrania . . . . .	26
5.1.1	Menu . . . . .	26
5.1.2	Zoznam automobilov . . . . .	26
5.1.3	Pridanie automobilu . . . . .	27
5.1.4	Náhľad automobilu . . . . .	27
5.1.5	Sumár . . . . .	28
5.2	Entitne-relačný diagram . . . . .	29

---

<b>6 Implementácia</b>	<b>32</b>
6.1 Nette . . . . .	32
6.2 MVC . . . . .	33
6.3 Verzovanie . . . . .	34
6.4 Šablóny . . . . .	34
6.5 Doplnky . . . . .	35
6.5.1 Datepicker . . . . .	35
6.5.2 DataTables . . . . .	36
6.5.3 Bootstrap Calendar . . . . .	37
6.6 Prihlasovanie . . . . .	37
6.6.1 Autentizácia . . . . .	37
6.6.2 Autorizácia . . . . .	37
6.7 Zamestnanci . . . . .	38
6.8 Automobily . . . . .	40
6.9 Udalosti . . . . .	41
6.10 Sumár . . . . .	42
6.11 Formuláre . . . . .	43
6.12 Validácie . . . . .	43
6.13 Rezervácia automobilov . . . . .	44
<b>7 Záver</b>	<b>45</b>
<b>8 Literatúra</b>	<b>46</b>
<b>Prílohy</b>	<b>48</b>
<b>A Priložené CD</b>	<b>49</b>

# 1 Úvod a cieľ práce

## 1.1 Úvod do problematiky

Je známe, že v dnešnej dobe počítačové aplikácie vedú k vyššej efektivite, či už v pracovnom, zábavnom alebo umeleckom prostredí. K desktopovým aplikáciám sa pridali webové. Tie dokážu konkurovať tým desktopovým, ako z pohľadu grafickej stránky, tak aj z pohľadu funkčnosti.

Evidencia patrí k jednej z oblastí, kde prechod do elektronickej podoby vo veľkej miere zrýchľuje a sprehľadňuje prácu mnohých osôb a podnikov. Z tohto dôvodu sa rozhodol majiteľ firiem Pizzeria Fantasia, Taxi Fantasia a Auto Žilina zmeniť svoj spôsob evidencie a prejsť tak k elektronickej podobe spracovania údajov. V tejto práci je na tento podnet vypracovaný návrh a realizácia webovej aplikácie, ktorá prechod z fyzickej podoby evidencie na podobu elektronickú umožní.

## 1.2 Cieľ práce

Cieľom tejto bakalárskej práce je návrh a implementácia evidenčného systému pre správu vozového parku, zamestnancov, udalostí a sledovanie majiteľom zvolených nákladov a výnosov pre vyššie spomínané firmy. Tento systém poskytne prehľad jednotlivých podnikateľských aktivít, ale zároveň umožní podnikateľovi združený prehľad všetkých podnikov ako jedného celku. Evidencia v elektronickej forme zjednoduší a urýchli prístup k potrebným dátam a zároveň klesne priestorová náročnosť na hmotné uchovávanie niektorých informácií. Všeobecne, pri webovej aplikácii používateľ nie je nútený inštalovať a aktualizovať software. Taktiež môže aplikáciu využívať na rôznych zariadeniach, pričom sa nemusí zaoberať synchronizáciou údajov, vďaka uchovávaniu údajov online.

Súčasťou práce je popísať využitie jazyka PHP pri tvorbe dynamických webových stránok s využitím MVC frameworkov, taktiež popísať súbežne používané technológie pri tvorbe tejto aplikácie ako CSS, HTML a databázové systémy.

## 2 Webové aplikácie

Webová aplikácia je aplikácia, ktorá je uložená na vzdialenom webovom serveri a je prístupná prostredníctvom internetu cez internetový prehliadač narozdiel od tradičných desktopových aplikácií, ktoré sú spúšťané na operačných systémoch (Web Application, 2015).

### 2.1 Statické webové stránky

Predstavujú súbor jednotlivých a pomocou odkazov vzájomne prepojených webových stránok. Každá webová stránka je samostatný dokument a nepoužíva žiadnu databázu alebo iné externé súbory. Nevýhodou tohto typu stránok je ich zložitá údržba pri nárokoch na ďalšie rozširovanie, a teda fakt, že nedokážu zobrazovať dáta uložené v databáze (Dynamické webové stránky, 2015).

### 2.2 Dynamické webové stránky

Základom je programovacia technológia, program a dáta uložené v databáze. Program (online obchod, redakčný systém, elektronická aukcia a podobne) umiestnený na internetovom serveri, potom na základe požiadavky na zobrazenie konkrétnej stránky webovú stránku dynamicky zostavuje a odosiela do prehliadača používateľa. Dynamické stránky majú tak omnoho väčšiu škálu funkcionality ako stránky statické. Mnoho dynamických stránok môže byť upravovaných prostredníctvom redakčných systémov. To umožňuje aktualizovať stránku bez nutnosti akejkoľvek znalosti HTML alebo iných vývojárskych prostriedkov. Dynamické stránky teda fyzicky neexistujú v uloženom stave, na rozdiel od statických stránok (Dynamické webové stránky, 2015).

## 3 Webové technológie

Táto kapitola poskytuje informácie o jazyku PHP a sním súbežne používaných technológiách ako HTML, CSS, JavaScript a databázových systémoch.

### 3.1 PHP

PHP (Hypertext Preprocessor) je jedným z programovacích jazykov, ktoré sú vykonávané na strane servera. Je určený najmä pre programovanie dynamických internetových stránok a webových aplikácií. Avšak PHP je možné použiť aj na tvorbu konzolových a desktopových aplikácií. Pri použití PHP pre dynamické stránky sú skripty vykonávané na strane servera, teda k používateľovi je prenášaný až výsledok ich činnosti. Syntax jazyka je inšpirovaná niekoľkými programovacími jazykmi ako Perl, C alebo Java. PHP jazyk je nezávislý na platforme, rozdiely v rôznych operačných systémoch sa obmedzujú na niekoľko systémovo závislých funkcií a skripty je možné väčšinou medzi operačnými systémami prenášať bez akýchkoľvek úprav. PHP nieje jediný zástupca jazykov vykonávaných na strane servera. Do tejto skupiny jazykov môžeme zaradiť taktiež jazyky ako ASP.net, Java, ColdFusion, Ruby, Perl alebo Python (PHP: What is PHP? - Manual, 2014).

#### 3.1.1 História PHP

Začiatky PHP spadajú do roku 1994 kedy sa Rasmus Lerdorf rozhodol vytvoriť jednoduchý systém pre počítanie prístupu ku svojim stránkam, tento systém bol napísaný v programovacom jazyku Perl. Neskôr bol tento systém prepísaný do jazyka C. V polovici roka 1995 sa jeho systém spojil s iným programom rovnakého autora, a to síce s nástrojom Form Interpreter, skrátene FI. Tak vznikol PHP/FI 2.0 systém, ktorý si postupne získal celosvetovú známosť a bol veľmi rozšírený. V súčasnosti je najnovšou verziou verzia PHP 5, avšak jazyk je naďalej aktívne vyvíjaný (PHP - Historie a budúcnosť -Linux Software, 2004).

#### 3.1.2 Využitie jazyka PHP

Jazyk PHP je použiteľný takmer pri všetkom, čo súvisí s dynamickými webovými aplikáciami. Svedčí o tom aj fakt, že patrí medzi najpoužívanejšie programovacie jazyky (TIOBE Index for March, 2015). Využívajú ho stránky ako napríklad sociálna sieť Facebook, či internetový vyhľadávač Yahoo. Medzi najčastejšie písané aplikácie patria napríklad:

- internetové obchody,
- podnikové informačné systémy,
- diskusné fóra,
- redakčné systémy,

- firemné prezentácie,
- dynamické osobné stránky,
- vyhľadávače a katalógy,
- drobnosti typu počítačta, ankety a mnoho ďalších. (PHP - Historie a budoucnost -Linux Software, 2004)

PHP je voľne dostupný produkt, teda nieje nutné platiť akékoľvek poplatky. Svoju obľúbenosť u webových vývojárov si získal aj vďaka dobrej spolupráci s jedným z najrozšírenejších webových serverov, serverom Apache. (Kosek, 1999)

### 3.1.3 PHP a frameworky

Framework je softwarová štruktúra, ktorá slúži ako podpora pri programovaní, vývoji a organizácii iných softwarových projektov. Môže obsahovať podporné programy, knižnicu API, návrhové vzory alebo doporučené postupy pri vývoji. Frameworky prevádzajú rutinné úlohy do všeobecných modulov, a tak uľahčujú ich opätovné používanie. Vďaka tomu programátor nieje nútený písať stále ten istý kód dookola. Využívanie frameworku je efektívnejšie ako používať čistý programovací jazyk. Využívať funkcie, ktoré už framework obsahuje ušetrí množstvo času a zároveň bývajú tieto funkcie mnohokrát elegantnejšie. Konkrétne frameworky využívajú architektúru, vďaka ktorej je pre webového vývojára jednoduchšie vykonávať zmeny v aplikácii. V súčasnej dobe existuje veľké množstvo dostupných frameworkov, ktoré využívajú jazyk PHP. Zvoliť si ten vhodný na vývoj svojej aplikácie môže byť náročné. Existujú rôzne kritéria, podľa ktorých má programátor možnosť sa rozhodnúť. Rozhodnúť sa môže napríklad podľa podporovaných technológií, výkonu, zložitosti inštalácie a v neposlednom rade veľkosťou komunity, ktorá tento framework využíva (PHP frameworky, 2008).

## 3.2 HTML

HTML (HyperText Markup Language) je štandardným značkovacím jazykom, ktorý sa používa pre vytváranie a zobrazovanie webových dokumentov v internetovom prehliadači. Vytvorený kód predstavuje návod, ako zobrazit dáta na obrazovke. HTML kód je teda presným a stručným vyjadrením toho, čo chceme v prehliadači zobrazit. Ako aj iné jazyky, HTML taktiež prešlo vývojom. Aktuálna oficiálna verzia, schválená konzorciom W3C je od októbra 2014 verzia HTML5. Táto verzia nadväzuje na verziu HTML 4.01 a prináša množstvo nových vlastností ako nové grafické možnosti, lepšiu podporu videí, animácií a mnoho ďalších nástrojov digitálnej publikácie, ktoré využívajú schopnosti dnešných a budúcich zariadení (Broža, 2003).



## 3.3 CSS

Kaskádové štýly alebo CSS(Cascading Style Sheets) sú všeobecné rozšírenie HTML. Štýly poskytujú tvorcom stránok oveľa širšie formátovacie možnosti, a to nielen čo sa týka formátovania textu. Tieto štýly sa používajú pre formátovanie vzhľadu webových stránok. Môžeme prostredníctvom nich definovať štýl textu, veľkosti tabuliek a ostatné aspekty webových stránok, ktoré predtým bolo možné definovať iba pomocou HTML. Samotný názov nám napovedá o tom, že jediný text môže ovplyvňovať viacej štýlov, ktoré ho formátujú (The Tech Terms Computer Dictionary, 2015).

## 3.4 JavaScript

JavaScript, ako už napovedá samotný názov, je skriptovací jazyk, ktorý sa používa v HTML stránkach. Skripty je možné zapísať priamo do HTML kódu stránky alebo pripojiť ako samostatný súbor. Tie sú potom prehliadačom interpretované až na strane klienta. To je dôvodom, prečo pre ich tvorbu nie je potrebný žiadny špeciálny nástroj alebo prekladač a zdrojové kódy sú všetkým prístupné. JavaScript je objektový jazyk. Jeho syntax vychádza z jazyka C++, preto je case sensitive. Tento jazyk je beztypový, preto nieje nutné pri deklarácií premenných špecifikovať, akého sú typu a konverzie hodnôt prebiehajú automaticky. Mnoho ľudí si myslí, že JavaScript je jazyk Java, poprípade nejaká jeho verzia, čo je ale omyl, s jazykom Java má spoločnú len časť mena. Má však podobnú syntax, pretože obidva jazyky vychádzajú z jazyka C++ (Václavek, 2003).

### 3.4.1 jQuery

jQuery je rýchla, malá multiplatformová knižnica jazyka JavaScript, ktorá kladie dôraz na interakciu medzi jazykom JavaScript a HTML. Syntax tejto knižnice je navrhnutá tak, aby zjednodušila prechádzanie HTML dokumentov, výber elementov, vytváranie animácií, obsluhu udalostí a vývoj aplikácií využívajúcich technológiu AJAX. Vďaka kombinácii všestrannosti a rozširiteľnosti, jQuery zmenilo spôsob, akým množstvo ľudí používa JavaScript (What is jQuery?, 2015).

### 3.4.2 AngularJS

AngularJS je JavaScriptová MVC (model view controller) knižnica, ktorá zjednodušuje a urýchľuje vývoj aplikácií, ktoré fungujú na desktopových ale aj mobilných platformách. AngularJS sa z neznámeho open source projektu za veľmi krátku dobu stal jedným z najznámejších a najpoužívanejších dostupných JavaScriptových knižníc. AngularJS v kombinácii s jQuery a Twitter Bootstrap ponúka všetko potrebné pre rýchlu tvorbu HTML5 a JavaScriptových frontendov, ktoré využívajú služby technológie REST pre backendové procesy (Ken Williamson, 2015).

### 3.5 Databázové systémy

Databázy sú špecializované štruktúry, ktoré umožňujú elektronickým systémom ukladať, spravovať a načítavať dáta vo veľmi krátkom čase. Databáza je zdieľaná, integrovaná počítačová štruktúra, ktorá ukladá sadu dát pozostávajúcu z:

- dát pre koncového používateľa - to sú dáta, o ktoré sa používateľ zaujíma a ktoré sa k nemu skutočne dostanú,
- metadát, alebo údajov o údajoch, cez ktoré sú dáta pre koncového používateľa integrované a spravované (Carlos CORONEL, Steven MORRIS, Peter ROB, 2013).

#### DBMS

DBMS (A database management system) je kolekcia programov, ktoré spravujú databázovú štruktúru a kontrolujú prístup k dátam uloženým v databáze (Carlos CORONEL, Steven MORRIS, Peter ROB, 2013).

### 3.6 SQL

SQL (Structured Query Language) je dopytovací neprocedurálny jazyk. Používa sa pre prístup k relačnej databáze. SQL poskytuje sadu príkazov pre ukladanie a načítanie dát z a do relačnej databázy. Tento jazyk naberal na popularitu, keď sa prvá relačná databáza začala celosvetovo rozširovať. SQL je dnes akceptovaný ako štandardný jazyk takmer pre všetky implementácie relačných databáz. Pre programátora SQL predstavuje rozhranie pre prístup k uloženým dátam v databáze. SQL dovoľuje vytvoriť databázové štruktúry ako sú tabuľky, pohľady a indexy. SQL dovoľuje vkladať dáta do databázy a získať uložené dáta v požadovanom formáte. Takisto umožňuje upravovať, vymazávať a inak manipulovať s uloženými dátami (MISHRA a BEAULIEU, 2004).

### 3.7 MySQL

MySQL je švédsky databázový systém založený na jazyku SQL. Je k dispozícii ako open source, teda program šírený zadarmo. Medzi výhody MySQL patrí aj podpora všetkých hlavných platforiem, vysoký výkon, rýchlosť a vynikajúca kompatibilita s inými systémami, hlavne so serverovým programom Apache a jazykom PHP. Vďaka týmto vlastnostiam sa systém MySQL presadil ako univerzálne riešenie používané na väčšine internetových projektoch a je automaticky dostupný takmer na všetkých typoch webhostingu. Nevýhody MySQL pramenia z jeho výhod. Nepodporuje zložitejšie programátorské konštrukcie a nemá dostatočný výkon v skutočne náročných webových aplikáciách. Pre tieto nároky je možné použiť konkurenčné databázy, napríklad PostgreSQL alebo Oracle. Aj napriek tomu MySQL dokáže vyhovieť vo väčšine prípadov (MySQL, 2015).

## 3.8 Oracle Database

Oracle je systém riadenia bázy dát(Oracle database management system - DBMS), moderný multiplatformový databázový systém s veľmi pokročilými možnosťami spracovania dát, vysokým výkonom a jednoduchou škálovateľnosťou. Databázový systém Oracle je vyvíjaný firmou Oracle Corporation. Aktuálnou verziou je Oracle Database 12c. Okrem toho, že tento systém podporuje štandardný dopytovací jazyk SQL, podporuje tiež imperatívny programovací jazyk PL/SQL, ktorý je firemným rozšírením Oracle, rozširujúci možnosti vlastného SQL. Ďalej podporuje objektové databázy a databázy uložené v hierarchickom modeli dát (XML databázy) (PROCHÁZKA, 2009).

## 4 Súčasný stav

Táto kapitola stručne predstavuje firmy Pizzeria Fantasia, Taxi Fantasia a Auto Žilina, pre ktoré má byť implementovaný evidenčný systém. Ďalej sa venuje analýze majiteľom kladených požiadaviek pre tvorbu tohto systému. Oboznamuje so súčasným stavom a existujúcimi systémami.

### 4.1 Súčasný stav

#### Pizza Fantasia

Je prvý z podnikov, ktorý bol založený majiteľom. Zaoberá sa výrobou a donáškou pizze a ponuku jedál rozširuje aj o iné druhy pokrmov. Firma pôsobí v meste Žilina a v jeho okolí. Objednávky prebiehajú telefonicky alebo prostredníctvom online objednávok z firemnej webovej stránky.

#### Taxi Fantasia

Taxi Fantasia je žilinská nonstop taxislužba, ktorá okrem klasickej prepravy osôb vykonáva aj služby na objednávku, napríklad rozvoz detí do školy alebo prepravu zamestnancov. Informácie o firme, rôznych akciách alebo kontakty na vodičov sú dostupné na statickej firemnej webovej stránke.

#### Auto Žilina

Je firma zaoberajúca sa prenájmom vozidiel pre rôzne krátkodobé aj dlhodobé účely. Firma má zriadenú webovú stránku s možnosťou online rezervácií vozidiel.

Majiteľ firiem nevyužíva žiadny elektronický systém pre uchovávanie informácií o zamestnancoch, stave vozového parku alebo nadchádzajúcich udalostiach. Taktiež majiteľ nepoužíva žiadnu aplikáciu, ktorá by mu poskytovala jednotný prístup k informáciám všetkých firiem zároveň.

### 4.2 Požiadavky majiteľa

Keď majiteľ týchto firiem začínal s firmou Pizza Fantasia s pár zamestnancami a mal k dispozícii zopár vozidiel nejavilo sa mu potrebné viesť si elektronickú evidenciu, keďže množstvo dát, ktoré bolo nutné evidovať nebolo až také veľké. Postupom času avšak pribúdali zamestnanci, automobily a začala sa rozrastať potreba efektívnejšieho prístupu k dátam a takisto manipulácia s nimi. Z týchto dôvodov sa majiteľ rozhodol založiť elektronickú evidenciu. Jedna z najdôležitejších fáz vývoja pri uskutočňovaní takéhoto projektu je kvalitný návrh a pre ten je dôležité mať čo najpresnejšie stanové požiadavky. Tieto požiadavky sa rozdeľujú do dvoch kategórií na funkčné a nefunkčné. Funkčné požiadavky určujú to, čo systém dokáže robiť, pri

interakcii s používateľom. Nefunkčné požiadavky sú obmedzenia kladené na systém alebo proces vývoja a teda dopĺňajú požiadavky funkčné.

#### 4.2.1 Funkčné požiadavky

Nasleduje zoznam funkčných požiadaviek:

- pridanie zamestnanca so základnými informáciami ako meno, priezvisko, email, telefónne číslo, pracovný úväzok, pracovná pozícia, dátum narodenia, poisťovňa, začiatok pracovného pomeru, koniec pracovného pomeru, rodné číslo a možnosť pridania profilovej fotografie,
- upravenie zamestnanca s možnosťou úpravy všetkých informácií spomenutých v bode vyššie,
- vymazanie zamestnanca,
- zobrazenie profilu zamestnanca,
- pridanie automobilu s informáciami o značke (meno automobilu), evidenčnom čísle vozidla, roku výroby, type motoru, dátume od kedy sa spláca leasing automobilu, dátume do kedy sa má leasing splácať, dátume platnosti stavu technickej kontroly, platnosti emisnej kontroly, dátume zakúpenia automobilu, dátume predaja automobilu, cene automobilu, možnosti zvoliť, či automobil patrí do vlastníctva alebo nie a pridanie reprezentačnej fotografie,
- upravenie automobilu s možnosťou úpravy všetkých spomenutých informácií v bode vyššie,
- vyradenie automobilu - informácia o vyradení automobilu z vlastníctva avšak automobil nieje vymazaný z evidencie,
- vymazanie automobilu,
- pridanie poistenia automobilu s informáciami o názve, dátume vzniku poistenia, dátume zániku poistenia a výške mesačnej splátky,
- upravenie poistenia automobilu s možnosťou úpravy všetkých spomenutých informácií v bode vyššie,
- vymazanie poistenia automobilu,
- pridanie poruchy automobilu s informáciami o názve, popise, dni vzniku poruchy, dni opravy poruchy, cene za opravu,
- upravenie poruchy automobilu s možnosťou úpravy všetkých informácií spomenutých v bode vyššie,
- pridanie udalosti s názvom, dátumom konania a popisom k danej udalosti,

- upravenie udalosti s možnosťou úpravy všetkých informácií spomenutých v bode vyššie,
- vymazanie udalosti,
- pridanie nákladu s názvom, dátumom, popisom a sumou,
- upravenie nákladu s možnosťou úpravy všetkých informácií spomenutých v bode vyššie,
- vymazanie nákladu,
- pridanie výnosu s názvom, dátumom, popisom a sumou,
- upravenie výnosu s možnosťou úpravy všetkých informácií spomenutých v bode vyššie,
- vymazanie výnosu,
- sumarizovať výnosy a náklady,
- možnosť vyhľadávať v dátach,
- zobrazíť celkový prehľad zamestnancov.
- zobrazíť celkový prehľad vozového parku,
- zobrazíť celkový prehľad udalostí,
- zobrazíť celkový prehľad nákladov a výnosov,
- v sekcii celkový prehľad možnosť filtrácie dát podľa firmy,
- pre firmu Auto Žilina požiadavka priradenia kalendára ku každému automobilu s informáciami o rezerváciách

Všetky požiadavky s výnimkou evidencie rezervácií pre automobily sa vzťahujú na všetky firmy. Rezervácie sa evidujú iba pri automobiloch firmy Auto Žilina.

#### 4.2.2 Nefunkčné požiadavky

Nasleduje zoznam nefunkčných požiadaviek:

- evidenčný systém bude vytvorený ako webová aplikácia,
- pri vývoji evidenčného systému bude použitý jazyk PHP, konkrétne framework Nette,
- evidenčný systém bude používať slovenský jazyk,
- pre budúcnosť by malo byť jednoduché, pridanie ďalšej firmy,
- informácie o rezerváciách automobilov v autopožičovni budú preberané z google kalendárov.

### 4.2.3 Používateľské role

Táto aplikácia má slúžiť hlavne potrebám majiteľa firiem. V súčasnom stave sa preto nekladie dôraz na rozlíšenie používateľských rolí, pretože sa predpokladá že jedna osoba bude schopná obsluhovať celý systém. Systém by mal teda mať iba jednu používateľskú rolu, ktorá má prístup ku všetkým funkciám v systéme bez akýchkoľvek obmedzení.

## 4.3 Existujúce systémy

Na trhu existuje mnoho produktov, ktoré integrujú a automatizujú veľké množstvo procesov súvisiacich s produkčnými činnosťami podniku. Typicky sa jedná o výrobu, logistiku, distribúciu, správu majetku, predaj, fakturáciu alebo účtovníctvo. Táto sekcia sa venuje existujúcim systémom, ktoré potenciálne vyhovujú požiadavkám stanoveným pre tvorbu evidenčného systému pre podnikateľa s viacerými podnikateľskými činnosťami.

### ABRA G4

Informačný systém ABRA poskytuje riešenia pre evidenciu podnikových procesov pokrývajúcich oblasti obchodu, ekonomických a účtovných agend a manažérskoho riadenia a plánovania. ERP (Enterprise Resource Planning) systém ABRA je vhodný pre malé aj veľké firmy, podnikateľov, živnostníkov, neziskové organizácie i štátnu správu.

Tento systém vyhovuje v nasledovných požiadavkách:

- evidencia vozidiel s detailnými informáciami vrátane evidencie STK alebo EK,
- možnosť evidovať zamestnancov, ich pracovné zaradenie a iné informácie,
- možnosť pristupovať k dátam online,
- systém je dostupný v slovenskom jazyku,
- umožňuje evidovať udalosti.

Tento systém nevyhovuje v nasledovných požiadavkách:

- systém je nutné poskladať z viacerých modulov pričom tieto moduly obsahujú príliš veľa funkcií a informácií, ktoré nie sú momentálne využiteľné pre potreby tejto firmy ako napríklad kniha jász, GPS lokácie vozidiel alebo fakturácie,
- tento systém je určený skôr pre väčší počet ľudí, ktorí s ním budú zaobchádzať,
- napriek tomu že niektoré informácie sú dostupné aj pomocou webového prehliadača, je nutné mať tento systém nainštalovaný aj lokálne,
- tento systém neposkytuje ucelený prehľad jednotlivých firiem do jedného celku,

- zložitost a finančná náročnosť tohto systému. (Informačné systémy | abra.sk, 2015)

Preziso	Meno	Titul	Popis pozície	Zamestna	Osobné Obs.	Telefón 1.	Fax	Neslo	PSČ	Ulica	Titul
	Dáňa	František	Ing.	Ne							
	Hajó	Milan	Jú.Úr.	podnik.zastupnik	Ne						
	Rachý	Martin	Ing.	obch.viediteľ	Ne						
	Müller	Johann		geschaeftsführer	Ne						
	Soldhan	Max		sales manager	Ne						
	Nezák	Vladimír		fin.nadzor	Ne						
	Kuzný	Karel		direktor	Ne						
	Čvachlavová	Mária		účtovníka	Ne						
	Čubáková	Jana		hlavná účtovníka	Ne						
	Pádek	Milan	Ing.	techn.	Ne						
	Surová	Johana	Ing.		Ne						
	Kradný	Peter	arch.		Ne						
	Zábranová	Lucia			Ne						
	Šteplná	Alžbeta			Ne						
	Oravá	Vladimír	Ing.	radca	Ne						
	Melichermý	Jar.		vedúci obchodu	Ne						
	Čížinský	Jaroslav	Ing.	vedúci	Ne						
	Kudláčková	Božena		účtovníka	Ne						
	Šesták	Marik	Ing.	magist	Ne						
	Novák	Jozef	PhDr.		Ne	+421 609 987 494		Přibruč	496 23	Za Kloboukom 13	
	Supervízor				Ne						
	Hegedý	Martin	Ing.	Áno	001	+421 211 672 351		Bratislava	821 05	U Pivovaru 13	
	Novák	Jaroslav	Ing.	Áno	002	+421 909 987 492		Zvolen 1	960 01	Za Kloboukom 15	
	Bisk	Peter	Áno	003	+421 602 748 961			Prasádky	496 23	Prasádky 322	
	Kučta	Václav	Áno	004	+421 903 744 981			Bratislava	821 01	Staráňa 322	
	Rasák	Daniel	Áno	005	+421 033 126 931			Prasádká vrata	496 12	Hrabá 6	
	Kúša	Eva	Áno	006	+421 033 444 621			Kopčinská cesta	496 40	Svetlá 11	

Obr. 1: ABRA online

## BYZNYS ERP

BYZNYS ERP sú podnikové informačné systémy a s nimi spojené nástroje určené pre podniky. Systém je možné zostaviť podľa odvetvia v akom firma podniká a ďalej ho rozširovať o rôzne moduly.

Tento systém vyhovuje v nasledovných požiadavkách:

- umožňuje evidovať jednotlivých zamestnancov,
- umožňuje evidovať informácie o stave vozového parku,
- systém je dostupný v slovenskom jazyku,
- má k dispozícii modul s kalendárom, vďaka čomu je možné evidovať rôzne udalosti.

Tento systém nevyhovuje v nasledovných požiadavkách:

- Moduly, ktoré sú potrebné pre zostavenie aplikácie podľa požiadaviek na tvorbu systému obsahujú veľa informácií a funkcií, ktoré majiteľ nevyužije,
- chýba prehľad všetkých firiem ako celku,
- systém je pomerne zložitý a cenovo náročný vzhľadom na požiadavky majiteľa (JKR - dodavateľ BYZNYS ERP, 2015)



The screenshot shows the 'Editace zakázky' (Edit Order) interface in the BYZNYS online system. The form is divided into several sections:

- Header:** Includes the BYZNYS logo and user information: 'Heslovník uživatele: Ing. Petr Anasov, Ph.D.' and 'HEURLINE: tel: 0987 654 321, e-mail: info@byznys.cz'.
- Navigation:** A sidebar on the left with a 'Zpět' (Back) button.
- Order Details:**
  - Číslo zakázky: 9804154874
  - Název: Velká zakázka
  - Zákazník: AUTOPORTE s.r.o.
  - Jméno: Petr Velký
  - Adresa: Krotkýhoava 58, 987 65, Brno, Česká republika, DIČ: 9807456548
  - Typ zakázky: Obchodní
  - Objednávka: Abonementová
  - Uzavřena: //
  - Termín: //
- Financial Summary:**
  - Plán. výnos: 0,00
  - Plán. náklady: 16,548
  - Plán. zisk / ztráta: 5,874
  - Výnos: 0,00
  - Náklady: 16,548
  - Zisk / ztráta: 5,874
- Buttons:** 'Zpět', 'Reset', and 'Kontrola' buttons are located at the bottom right of the form.

Obr. 2: snímka obrazovky zo systému BYZNYS online (BYZNYS online)

#### 4.3.1 Potreba návrhu nového systému

Dnes trh ponúka veľké množstvo produktov, ktoré poskytujú riešenia pre evidenciu, avšak tieto riešenia sú často spojené aj s inými podnikovými procesmi a je nutné ich skladať z rôznych modulov. Takisto sú tieto systémy primárne zamerané na väčší počet používateľov, poprípade zákazníkov. Evidenčný systém, ktorý je predmetom tejto práce je však určený pre majiteľa alebo inú osobu poverenú evidenciou. Nepredpokladá sa že by systém obsluhovalo veľké množstvo používateľov.

Jedným z dôvodov na vytvorenie nového systému je celkový prehľad všetkých firiem. Niektoré existujúce systémy ponúkajú možnosť evidovať na jednu licenciu viacero firiem ale separátne. Požiadavkou pre tvorbu tohto systému je možnosť zjednotiť pohľad na tieto firmy ako na jednu firmu. Ďalším dôvodom je, že pri väčšine systémov je nutná inštalácia softwaru na zariadenie. Pri vyššie spomínaných systémoch je síce možné pristupovať k dátam online, ale pre tento prístup je nutné pridať ďalší modul ako doplnok k desktopovej aplikácii.

V neposlednom rade je potrebné poukázať na cenovú náročnosť a potrebu školení k rozsiahlejším systémom, ktoré predstavujú pre majiteľa nezanedbateľnú záťaž.

## 5 Návrh riešenia

Táto časť práce sa venuje návrhu riešenia. Vysvetľuje postup pri grafickom návrhu riešenia a tvorbe databázového modelu.

### 5.1 Návrh grafického používateľského rozhrania

Návrh grafického rozhrania vznikol pri diskusiách s majiteľom firiem. Aplikácia ako taká nemá zložitú funkcionálnosť, zámerom bolo, aby sa táto jednoduchosť rovnako dodržala aj v používateľskom rozhraní. Na stránkach sa nenachádzajú články ani žiadne dlhé texty. Zobrazované majú byť primárne informácie spojené s evidenciou. Skôr ako sa vytvoril konečný prototyp v prostredí programu Balsamiq Mocup, celý návrh vznikol pomocou pera a papiera. Avšak prvotný dizajn prešiel pomerne veľkou sériou zmien pred samotnou implementáciou. Nasledoval grafický návrh aplikácie v elektronickej podobe. Jednotlivé stránky boli reprezentované mockupmi v programe Balsamiq Mocup. Tento program bol zvolený kvôli intuitívnemu rozhraniu, príjemnému vzhľadu. Pre každú stránku aplikácie bol osobitne vytvorený jeden mockup.

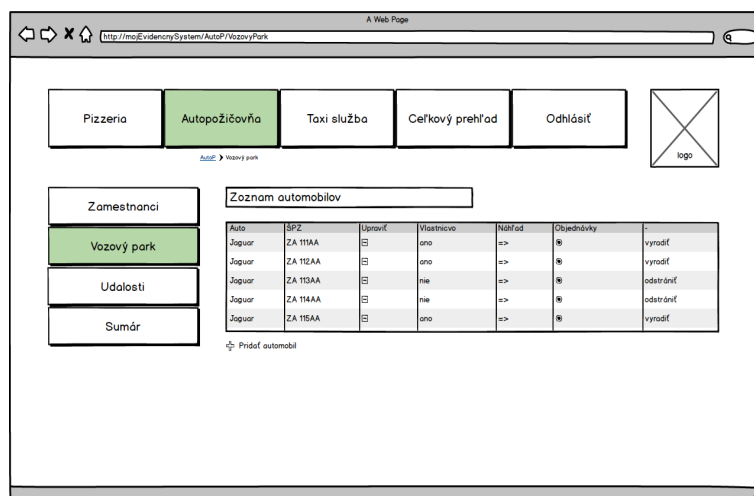
#### 5.1.1 Menu

Základným elementom na stránke je menu, ktoré sa skladá z dvoch častí. Prvou časťou je horizontálne menu v hornej časti stránky, pomocou ktorého môže používateľ prechádzať medzi firmami alebo celkovým prehľadom. Druhou časťou je vertikálne menu, ktoré sa nachádza v ľavej časti stránky. Toto menu umožňuje používateľovi prechod medzi hlavnými sekciami aplikácie, čo sú zamestnanci, vozový park, udalosti a sumár. Aktuálna pozícia v horizontálnom aj vertikálnom menu je farebne odlišená od ostatných segmentov menu, vďaka čomu je používateľ informovaný o aktuálnej pozícii v systéme.

Menu je zobrazované na všetkých stránkach aplikácie s výnimkou úvodnej, na ktorej sa nachádza rozcestník. Táto stránka sa zobrazí používateľovi ihneď po prihlásení, zobrazuje logá firiem a celkový prehľad, ktoré predstavujú odkazy do jednotlivých častí systému.

#### 5.1.2 Zoznam automobilov

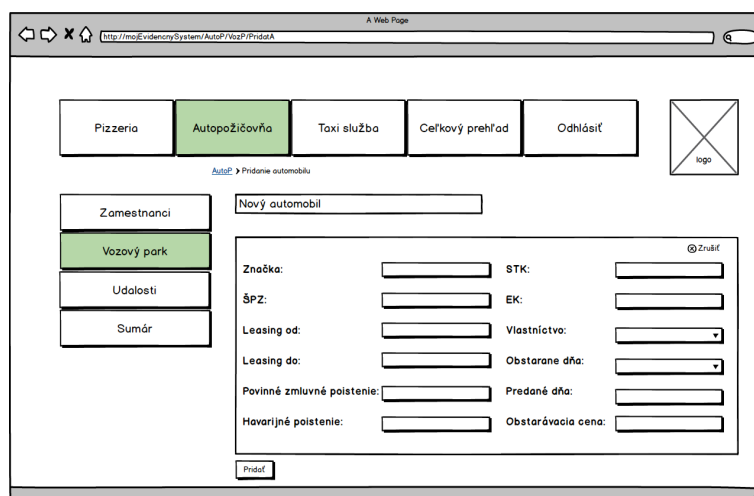
Výpis všetkých automobilov pre autopožičovňu s ich základnými informáciami. Používateľ má z tejto stránky taktiež možnosť editácie, zobrazenia profilu, vyradenia alebo vymazania. Rozhodujúcim faktorom je, či je automobil aktuálne vo vlastníctve alebo je už z vlastníctva vyradený.



Obr. 3: Zoznam automobilov

### 5.1.3 Pridanie automobilu

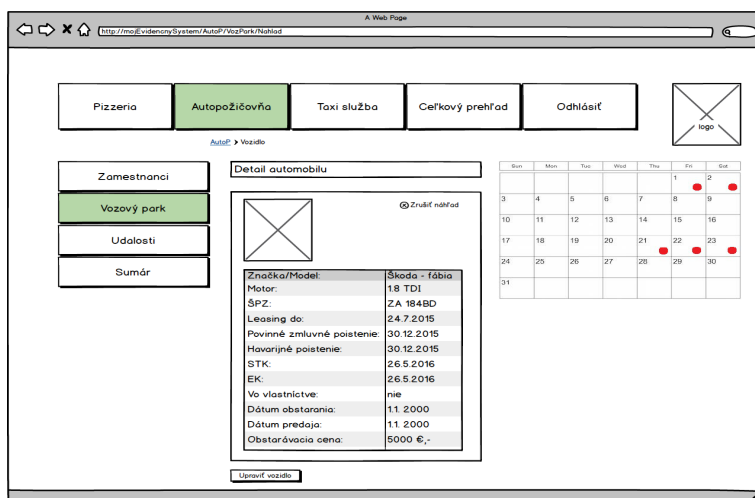
Používateľovi sa zobrazí formulár, obsahujúci pomerne veľké množstvo prvkov. Nie všetky prvky vo formulári sú povinné. Pre úpravy automobilu sa používa rovnaký formulár, kde sú vstupné polia vyplnené aktuálnymi dátami z databázy.



Obr. 4: Pridanie automobilu

### 5.1.4 Náhľad automobilu

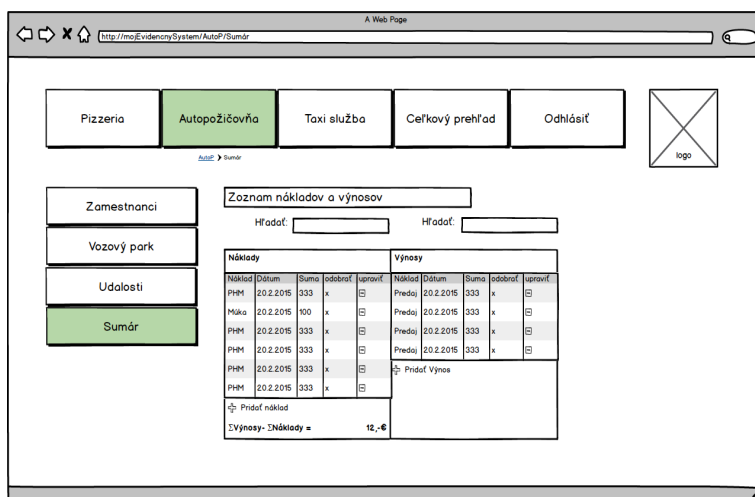
V detaile automobilu sa používateľovi zobrazujú všetky informácie o vozidle vrátane fotografie, ktoré sú uložené v databáze. Používateľ má možnosť editovať automobil z tejto stránky. Taktiež sa mu zobrazuje kalendár, s rezerváciami vozidla ku aktuálnemu mesiacu a zároveň má k dispozícii informácie o rezerváciách.



Obr. 5: Náhľad automobilu

### 5.1.5 Sumár

V sumári má používateľ k dispozícii informácie o určitých nákladoch a výnosoch konkrétnej firmy s možnosťou vyhľadávania medzi nimi oddelene. Stránka ďalej ponúka možnosť zobrazenia detailnejších informácií, vymazanie a pridanie. Na stránke sa tiež nachádza sumarizácia nákladov a výnosov.



Obr. 6: Sumár

Spomenuté webové stránky demonštrujú základné prvky používateľského grafického rozhrania vo webovej aplikácii. Rozhranie všetkých firiem je až na farebné označenie, ktoré je závislé na firemných farbách takmer rovnaké. V pizzerii a v taxislužbe detail auta neobsahuje kalendár nakoľko to nieje potrebné. Rovnakým spôsobom je navrhnutý aj celkový prehľad firiem, ktorý zoskupuje všetkých zamestnancov, automobily, udalosti a náklady s výnosmi.

## 5.2 Entitne-relačný diagram

Modelovanie dát je technika pre organizáciu a dokumentovanie systémových dát. Modelovanie dát je niekedy nazývané aj modelovanie databáz pretože dátový model sa nakoniec implementuje ako databáza. Skutočný model je často nazývaný ERD (Entity-Relationship Diagram), pretože znázorňuje dáta, v termínoch entita a väzba popísané dátami. ERD sa používa pre návrh dátových štruktúr a databázových systémov. Tri kľúčové koncepty ER diagramu sú entity, atribúty a väzby. (LI a CHEN, 2009)

### Entita

Entita interpretuje skutočný objekt, ktorý je možné odlíšiť od ostatných objektov. Entitou môže byť trieda osôb, miest, objektov, udalostí alebo konceptov, ku ktorým potrebujeme zachytiť a uložiť dáta. (LI a CHEN, 2009)

### Atribút

Atribút je popisujúca premenná alebo charakteristika, ktorá popisuje entitu. (LI a CHEN, 2009)

### Väzba

Väzba alebo vzťah je asociácia, ktorá existuje medzi jednou alebo viacerými entitami. Kardinalita definuje minimálne alebo maximálne číslo výskytov entity, ktorá je vo vzťahu k výskytom nejakej inej entity. Množiny vzťahov, ktoré zahŕňajú dve množiny entít sú binárne. Tento typ väzieb využívajú databázové systémy. Z tohto dôvodu musí byť kardinalita definovaná v oboch smeroch pre každý vzťah. V databázach sa používajú 3 typy väzieb:

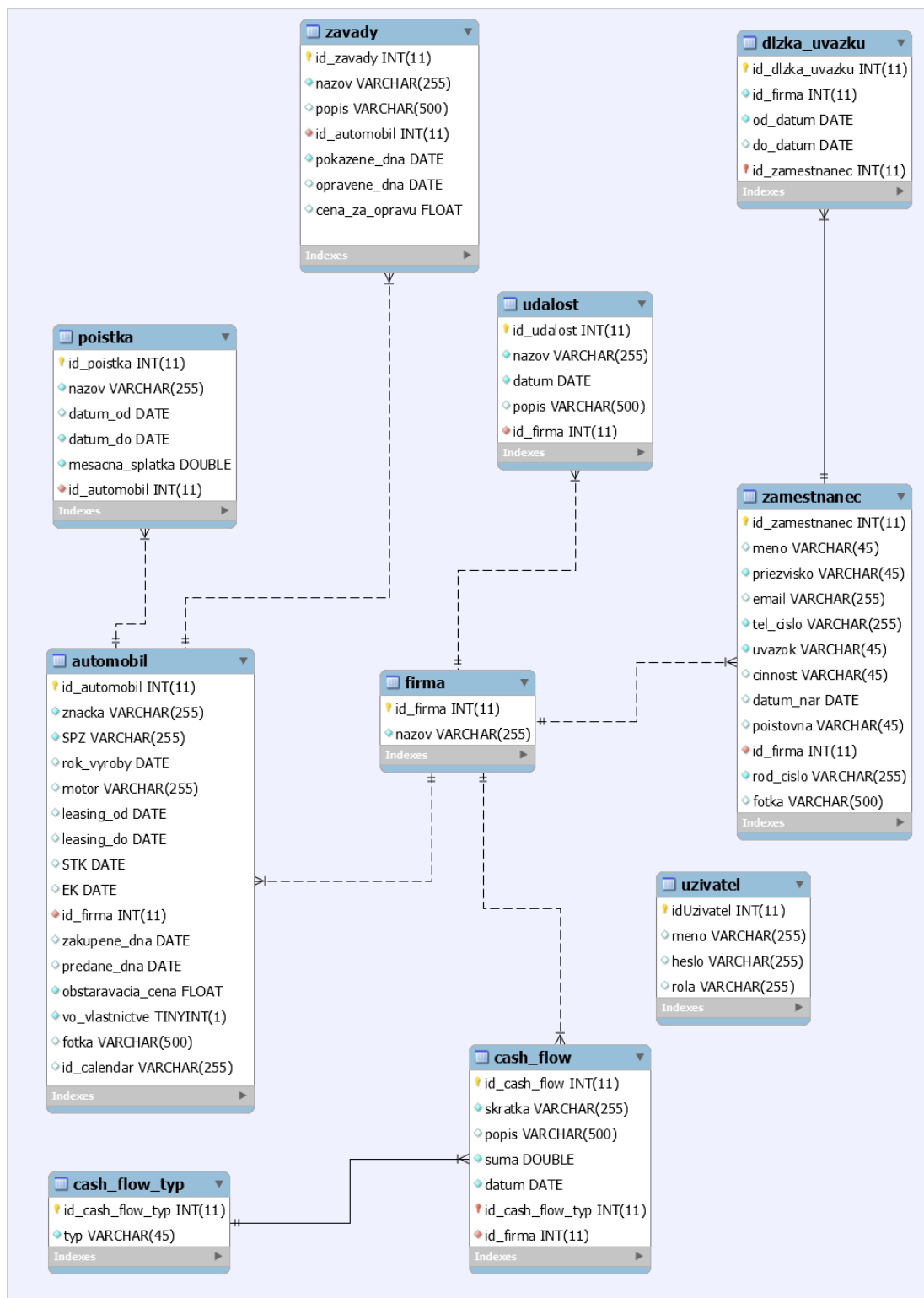
- 1:1,
- 1:N,
- M:N (LI a CHEN, 2009).

Všetky tabuľky a ich atribúty boli vytvorené pomocou programu MySQL Workbench, v ktorom bol vyexportovaný aj ER diagram so všetkými väzbami. MySQL Workbench je nástroj pre vizuálny návrh, správu a refaktoring databáze.

Databázový model sa skladá z 10 tabuliek. V každej tabuľke je primárnym kľúčom stĺpec ID. Entity so všetkými atribútmi sa nachádzajú v nasledujúcom zozname:

- **firma** - Entita, podľa ktorej sú kategorizované informácie do jednotlivých firiem. Táto entita obsahuje nasledujúce atribúty:
  - *id\_firma, nazov*

- **zamestnanec** - entita obsahuje základné informácie o zamestnancoch. Táto entita obsahuje nasledujúce atribúty:
  - *id\_zamestnanec, meno, priezvisko, email, tel\_cislo, uvazok, cinnost, datum\_nar, poistovna, rod\_cislo, fotka, id\_firma*
- **dlzka\_uvazku** - entita uchováva informácie o dĺžke pracovného úväzku zamestnancov. Táto entita obsahuje nasledujúce atribúty:
  - *id\_dlzka\_uvazku, id\_firma, od\_datum, do\_datum, id\_zamestnanec*
- **udalost** - táto entita zachytáva základné informácie o udalostiach v podniku, uchováva nasledujúce atribúty:
  - *id\_udalost, nazov, datum, popis, id\_firma*
- **cash\_flow** - entita uchováva informácie o nákladoch alebo výnosoch, ktoré sa rozhodne majiteľ evidovať. Táto entita obsahuje nasledujúce atribúty:
  - *id\_cash\_flow, skratka, (reprezentuje aktivitu, z ktorej vznikol náklad alebo výnos) popis, suma, datum, id\_cash\_flow\_typ, id\_firma*
- **cash\_flow\_typ** - entita, ktorá rozlišuje náklady od výnosov. Táto entita obsahuje nasledujúce atribúty:
  - *id\_cash\_flow\_typ, typ*
- **automobil** - entita uchováva informácie o automobiloch. Táto entita obsahuje nasledujúce atribúty:
  - *id\_automobil, znacka, SPZ (štátna poznávací značka), rok\_vyrobny, motor, leasing\_od, leasing\_do, STK (dátum platnosti technickej kontroly) EK (dátum platnosti emisnej kontroly), zakupene\_dna, predane\_dna, obstaravacia\_cena, vo\_vlastnictve, fotka, id\_calendar, (nepovinný atribút, platí iba pre vozidlá z autopožičovne, ku ktorým je priradený google calendar) id\_firma*
- **poistka** - entita, ktorá zaznamenáva informácie o poistení vzťahujúcich sa ku konkrétnemu automobilu. Entita uchováva nasledujúce atribúty:
  - *id\_poistka, nazov, datum\_od, datum\_do, mesacna\_splatka, id\_automobil*
- **zavady** - entita uchováva informácie o závadách automobilu. Entita uchováva nasledujúce atribúty:
  - *id\_zavady, nazov, popis, pokazene\_dna, opravene\_dna, cena\_za\_opravu, id\_automobil*



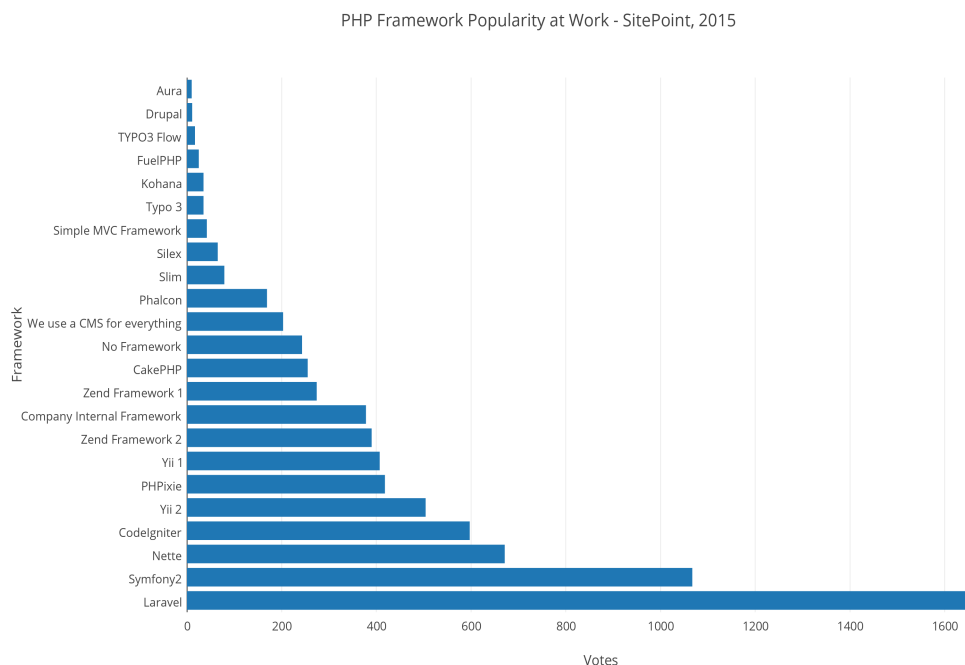
Obr. 7: ER diagram

## 6 Implementácia

Podmienkou pre tvorbu systému bolo vytvorenie evidenčného systému ako webovej aplikácie, v jazyku PHP. Pri implementácii systému bol zvolený framework Nette. Ako databázový systém pre ukladanie potrebných dát bol zvolený MySQL.

### 6.1 Nette

Využívanie frameworkov pre tvorbu aplikácii výrazne zjednodušuje implementáciu. Nette je webový framework, ktorý využíva jazyk PHP. Podľa magazínu SitePoint je Nette tretí najpopulárnejší framework hneď za Symfony2 a Laravel, čo je spôsobené predovšetkým jeho rozšírením na území Českej a Slovenskej republiky. (viď. obr. 8).



Obr. 8: Oblúbenosť frameworkov (Best PHP Framework for 2015, 2015)

Framework Nette sa svojou veľkosťou neradí medzi najväčšie PHP frameworky ako napríklad Symfony alebo Zend. Svoju oblúbenosť si získal aj vďaka svojej jednoduchosťi narozdiel od väčších frameworkov. Avšak Nette sa stále rozrastá a poskytuje dobré zabezpečenie, ktoré eliminuje výskyt bezpečnostných dier a ich zneužitie ako napríklad XSS alebo CSRF. Rozrastá sa aj oficiálna ponuka doplnkov a rozšírení, ktoré jednoducho riešia bežné problémy ako napríklad menu, pripojenie ku Google API a množstvo iných. Nette využíva moderné technológie, ku ktorým patrí aj architektúra MVC (Nette Framework, 2015).



## 6.2 MVC

MVC (Model-View-Controller) je návrhový vzor pre vývoj aplikácií. Prvá implementácia tohto vzoru, bola vytvorená Trygvom Reenskaugom v roku 1978. Zjednodušená MVC rozdeľuje aplikáciu do troch rôznych vrstiev. Prvá vrstva je tzv. Model, ktorý predstavuje logiku aplikácií. Vrstva View zobrazuje používateľské rozhranie a vrstva Controller spája Model a View (Bernard, 2009).

### Model (Model)

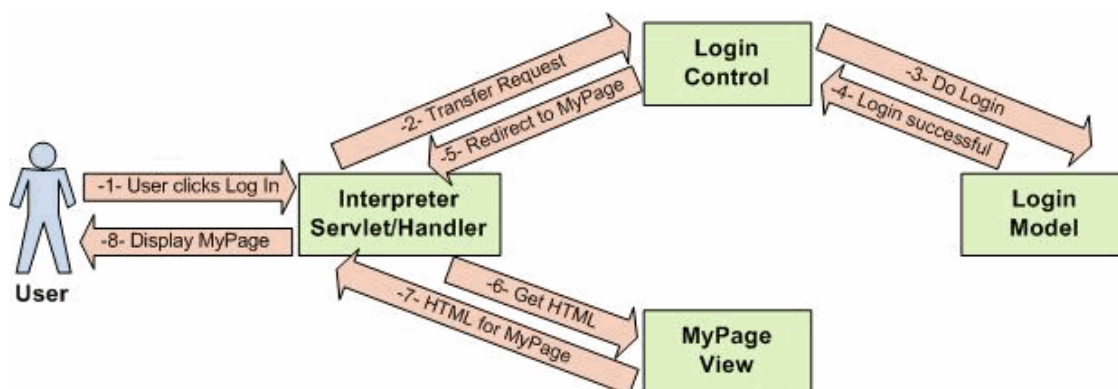
Predstavuje biznis logiku aplikácie. Táto vrstva v podstate zaisťuje základnú činnosť aplikácie. Umožňuje prístup k uloženým dátam a zároveň predstavuje výkonnú časť aplikácie. Býva najzložitejšou vrstvou aplikácie a často obsahuje ďalšie logické členenie (Portal UI Architecture, 2012).

### Pohľad (View)

View predstavuje používateľské rozhranie aplikácie. Je to vrstva zodpovedná za získavanie dát od používateľa a za zobrazenie dát (predstavovaných modelom) vhodnou formou používateľovi. Pre jeden model je možné vytvoriť viacero rôznych pohľadov, ktoré potom slúžia pre odlišnú prezentáciu rovnakých dát rôznym užívateľom alebo pre odlišnú prezentáciu dát na rôznych zariadeniach. Pokiaľ sa stav modelu zmení, potom sa pohľad aktualizuje (Portal UI Architecture, 2012).

### Radič (Controller)

Je umiestnený medzi vrstvami Model a Pohľad. Jeho hlavnou úlohou je odchytať a spracovávať požiadavky, ktoré zasiela používateľ na server. Na základe týchto požiadaviek sú volané jednotlivé akcie v modeloch a na základe výsledku tejto akcie je zobrazený výsledok používateľovi (Portal UI Architecture, 2012).



Obr. 9: Prihlásenie v MVC architektúre (Portal UI Architecture, 2012)

### 6.3 Verzovanie

Pred začiatkom implementácie bolo potrebné zvoliť systém, ktorý bude evidovať všetky zmeny, ktoré budú vykonané na jednotlivých súboroch. Pre tento účel bol zvolený verzovací systém Git. Verzovací systém zaznamenáva zmeny súboru alebo sady súborov v priebehu času, a používateľ tak môže kedykoľvek obnoviť ich konkrétnu verziu. Verzovací systém umožňuje vrátiť jednotlivé súbory alebo celý projekt do predchádzajúceho stavu, porovnávať zmeny uskutočnené v priebehu času, zistiť, kto naposledy zmenil daný súbor, čo možno spôsobuje problém alebo kto a kedy vložil akú verziu. Pri implementácii systému viacerými vývojármi verzovací systém GIT umožňuje vytvorenie viacerých vývojárskych vetiev, ktoré sú na sebe nezávislé a vývojári tak môžu vytvárať novú funkcionality bez toho, aby sa navzájom ovplyvňovali (Chacon, 2009).

Na trhu existuje niekoľko internetových bezplatných alebo spoplatnených služieb, ktoré poskytujú možnosť vytvorenia a prevádzkovania repozitára. Medzi tieto spoločnosti patrí napríklad GitHub alebo Bitbucket. Pre tento systém bola vybraná webová služba Bitbucket, ktorá ponúka bezplatný hosting pre open-source projekty (DjangoSuccessStoryBitbucket – Django, 2015).

### 6.4 Šablóny

Grafický návrh systému je založený na voľne dostupnej šablóne Arcana. Šablóna je vytvorená v jazyku HTML5 a CSS3. Taktiež využíva responzívny dizajn, čo umožňuje pracovať s aplikáciou na zariadeniach s rôznym rozlíšením. Táto šablóna bola upravovaná tak aby vyhovovala požiadavkám systému. Najvýraznejšie zmeny v šablóne boli vykonané vo vzdialenostiach elementov. Ďalej bol zmenený font písma na Century Gothic. Zmenená bola tiež aj farba dizajnu, ktorá sa mení v závislosti od aktuálnej pozície používateľa. Použité sú štyri rôzne farby:

- Pre pizzeriu odtieň oranžovej farby s hexadecimálnym kódom #F79621.
- Pre taxi službu bol použitý odtieň žltej farby s hexadecimálnym kódom #F1B519.
- Pre autopožičovňu bol použitý odtieň zelenej farby s hexadecimálnym kódom #46A748.
- Pre celkový prehľad bol zvolený odtieň modrej s hexadecimálnym kódom #37c0fb.

Tieto farby boli vybrané po konzultácii s majiteľom spoločností podľa farieb, ktoré využívajú logá jednotlivých podnikov. Farebné odlišenie taktiež napomáha používateľovi s navigáciou v systéme.

## Twitter Bootstrap

Pre vylepšenie užívateľskej prívetivosti bol použitý framework Twitter Bootstrap. Bootstrap je populárny framework, ktorý urýchľuje a zjednodušuje vývoj webových aplikácií. Zahŕňa šablóny založené na HTML a CSS, ktoré ponúkajú komponenty pre používateľské rozhranie ako ikony, formuláre, tlačidlá, tabuľky, navigáciu, rolovaciu menu, stavové správy, modálové okná a mnoho ďalších doplnených o rozšírenia JavaScriptu. Ďalšie výhody vyplývajúce z využívania Twitter Bootstrap frameworku:

- Šetrí mnoho času a úsilia využívaním preddefinovaných šablón a tried, takže je možné sústrediť sa na vývoj ostatných častí aplikácie.
- Používanie Bootstrapu uľahčuje tvorbu responzívneho dizajnu.
- Jednoduchosť používania Bootstrapu. Na použitie tohto frameworku stačí jednoduchá znalosť HTML a CSS.
- Kompatibilita s prehliadačmi. Bootstrap je kompatibilný vo všetkých moderných webových prehliadačoch ako Mozilla Firefox, Google Chrome, Safari, Internet Explorer a Opera.
- Framework je možné stiahnuť a používať úplne zdarma (Twitter Bootstrap, 2015).

V tomto systéme bol využitý Bootstrap napríklad pri flash správach, ktoré oznamovali určité akcie v podobe modálových okien, tlačidiel alebo kalendárov využívaných pri rezerváciách jednotlivých automobilov.

## 6.5 Doplnky

Táto aplikácia využíva niekoľko doplnkov pre ďalšie zlepšenie používateľského prostredia. Napriek tomu že vývojár musí tieto prvky pripojiť k aplikácii a následne nakonfigurovať podľa potrieb, je réžia potrebná pre použitie týchto doplnkov menšia, ako keby bolo nutné všetko od základov programovať.

### 6.5.1 Datepicker

Tento doplnok sa nachádza pri vstupných formulárových poliach s dátumom. Potom, ako používateľ klikne do jedného z týchto polí s dátumom zobrazí sa malý kalendár, kde je možné zvoliť dátum. Kalendár je implementovaný v jazyku JavaScript a využíva knižnicu jQuery.

Pri využívaní tohto doplnku bolo nutné definovať základné správanie tohto kalendára. Táto konfigurácia pozostávala z nastavenia slovenskej lokalizácie, nastavenia prvého dňa v týždni a formátu dátumu na D.M.YY.



Obr. 10: Datepicker

### 6.5.2 DataTables

DataTables je podobne ako Datapicker JavaScriptový doplnok, ktorý využíva knižnicu jQuery. Je to flexibilný nástroj, ktorého funkcionality sa stále rozširuje. Pridáva pokročilé možnosti s prácou na akýchkoľvek HTML tabuľkách.

Evidenčný systém tento doplnok vo veľkej miere využíva, keďže tabuľky sa vyskytujú takmer v každej hlavnej sekcii a tiež vo všetkých firmách. V systéme doplnok DataTables rozširuje funkcionality o možnosti filtrovania dát, zoradenia dát podľa zvoleného stĺpca, voľby počtu zobrazených záznamov na jednu stranu a v neposlednej rade možnosť aktivácie responzívneho dizajnu.

Zobraz  záznamov Hľadať:

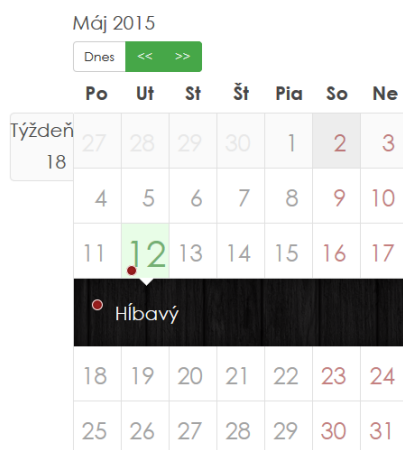
Firma	Priezvisko	Prac. pozícia	Upraviť	Profil	Vymazať
Pizzeria	Mladý	Šofér			Vymazať
Pizzeria	Pazucha	Upratovačka			Vymazať
Pizzeria	Kuchárik	Kuchár			Vymazať
Pizzeria	Včerajší	Upratovačka			Vymazať
Pizzeria	Dočohomansky	Šofér			Vymazať
Požičovňa	Uganda	Upratovačka			Vymazať
Taxi	Trstenský	Vodič			Vymazať

Záznamy 1 až 7 z celkom 7 Predchádzajúca  Nasledujúca

Obr. 11: DataTables

### 6.5.3 Bootstrap Calendar

Ďalším doplnkom je šablóna pre jednoduchý kalendár. Kalendár sa zobrazuje v detaile automobilov, ktoré patria k firme Auto Žilina. Obsahuje informácie o rezerváciách automobilov. Rezervácie sa zobrazujú ako farebné značky v jednotlivých dňoch mesiaca. Po kliknutí na konkrétnu položku sa používateľovi zobrazí modálne okno s detailnými informáciami o rezervácii. Kalendár využíva framework Twitter Bootstrap a tiež knižnicu jQuery.



Obr. 12: Bootstrap Calendar

## 6.6 Prihlasovanie

Táto sekcia je rozdelená na dve časti a to na autentizáciu a autorizáciu. Vstup do evidenčného systému je možný len po prihlásení, čo zabezpečí že informácie zo systému budú dostupné iba oprávneným osobám.

### 6.6.1 Autentizácia

Pod pojmom autentizácia sa rozumie prihlasovanie používateľov, teda proces, počas ktorého sa overuje, či je používateľ skutočne tým, za koho sa vydáva. Zväčša sa preukazuje používateľským menom a heslom (Nette Framework, 2014).

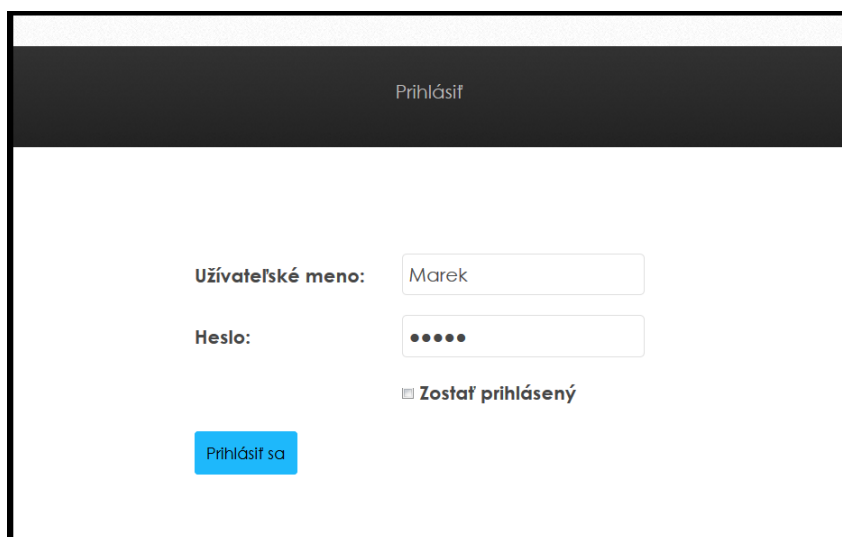
V súčasnosti nebola definovaná potreba vytvárať nových užívateľov systému pomocou systémového rozhrania nakoľko do systému bude mať prístup iba majiteľ spoločnosti. V budúcnosti, je možné rozšíriť systém o túto funkcionálnosť.

### 6.6.2 Autorizácia

Autorizácia zisťuje, či má používateľ dostatočné oprávnenia, pre uskutočnenie nejakej akcie. Autorizácia predpokladá že používateľ je úspešne prihlásený, teda úspešnú autentizáciu.

Nette pre autorizáciu používa Autorizátor. Autorizátor rozhoduje o tom či má používateľ oprávnenie vykonať určitú akciu. Ide o objekt implementujúci rozhranie `Nette\Security\IAuthorizator` s jedinou metódou `isAllowed()`, ktorej úlohou je rozhodnúť či ma daná používateľská rola povolenie uskutočniť určitú operáciu.

Nette Framework disponuje predpripravenou implementáciou autorizátora a to triedou `Nette\Security\Permission` poskytujúcou programátorovi ľahkú a flexibilnú ACL (access control list) vrstvu pre riadenie práv a prístupu. Práca s ňou spočíva v definícii rolí, zdrojov a jednotlivých oprávnení. Role a zdroje umožňujú vytvárať hierarchie. Používateľské role sa definujú metódou `$this->addRole()`. Zdroje sa definujú metódou `$this->addResource()`. Nakoniec zostáva priradiť práva konkrétnym používateľským roliam čo vykonáva metóda `$this->allow()`, v ktorej je prvým atribútom používateľská rola a druhým je pole s povolenými akciami. Kontrolu nad tým, či je používateľ aktuálne prihlásený poprípade či má práva na vykonať určitú akciu rieši trieda `StartupPresenter`. Od tejto triedy potom dedia ostatné prezentery (Nette Framework, 2014).



The image shows a login form with a dark header containing the text "Přihlásit". Below the header, there are two input fields: "Uživatelské meno:" with the value "Marek" and "Heslo:" with five dots representing a password. Below these fields is a checkbox labeled "Zostať prihlásený" which is currently unchecked. At the bottom left, there is a blue button labeled "Přihlásit sa".

Obr. 13: Prihlasovanie do systému

## 6.7 Zamestnanci

V sekcii zamestnanci sú opísané operácie, ktoré je možné vykonávať zo zamestnancami.

### Vytvorenie nového zamestnanca

Zamestnanca je možné pridať pomocou formulára, kde používateľ vyplní požadované atribúty. O správnosť vstupných dát sa starajú vytvorené validátory. Pre zamestnanca boli vytvorené validátory, ktoré kontrolujú správnosť:

- emailu,
- telefónneho čísla,
- dátumu narodenia,
- dátumov pracovného pomeru.

Spomenuté validátory sú popísané v sekcii o validátoroch nižšie.

**Nový zamestnanec** ✖

<b>Meno:</b>	<input type="text" value="Meno"/>	<b>*Priezvisko:</b>	<input type="text" value="Priezvisko"/>
<b>Email:</b>	<input type="text" value="Email"/>	<b>*Telefónne číslo:</b>	<input type="text" value="Telefónne číslo"/>
<b>*Pracovný úväzok:</b>	<input type="text" value="Úväzok"/>	<b>*Pracovná pozícia:</b>	<input type="text" value="Pracovná pozícia"/>
<b>*Dátum narodenia:</b>	<input type="text" value="Dátum narodenia"/>	<b>Poisťovňa:</b>	<input type="text" value="Poisťovňa"/>
<b>Začiatok pracovného pomeru:</b>	<input type="text" value="Začiatok prac..."/>	<b>Koniec pracovného pomeru:</b>	<input type="text" value="Koniec prac..."/>
<b>*Rodné číslo:</b>	<input type="text" value="Rodné číslo"/>		
<b>Fotografia:</b>	<input type="button" value="Prehľadávať..."/> Nie je zvolený súbor.		

Obr. 14: Zoznam zamestnancov v pizzerii

## Úprava zamestnanca

Úprava zamestnanca prebieha obdobne ako pri pridávaní nového zamestnanca. Používateľovi sa taktiež zobrazí formulár s predpripravenými hodnotami, ktoré sú aktuálne uložené v databáze.

## Zoznam zamestnancov

Na úvodnej stránke v sekcii zamestnancov je zobrazený zoznam zamestnancov. Z tejto stránky je potom možné pridať, upraviť, zobrazíť profil alebo vymazať jednotlivých zamestnancov. Zoznam je možné zoradovať podľa priezviska, úväzku, pracovnej pozície alebo rodného čísla. Pri zobrazovaní zoznamu zamestnancov bol využitý doplnok Datatables spomenutý v kapitole vyššie.

**Zoznam zamestnancov**

Priezvisko	Úväzok	Prac. pozícia	Rodné číslo	Upraviť	Profil	Vymazať
Dočohomansky	Dohoda	Šofér	123456/1234			Vymazať
Kuchárik	Plný úväzok	Kuchár	123456/7890			Vymazať
Mladý	Dohoda	Šofér	111111/1111			Vymazať
Pazucha	Dohoda	Upratovačka	121212/1212			Vymazať

Nový zamestnanec

Obr. 15: Zoznam zamestnancov v pizzerii

## 6.8 Automobily

Táto sekcia sa venuje evidencii vozidiel a akciám s ňou spojených.

### Vytvorenie nového automobilu

Automobil sa pridáva vyplnením formulára, v ktorom sú taktiež povinné a nepovinné polia. Vo formulári pre vozidlá správnosť vstupov kontrolujú nasledovné vstupy:

- rok výroby,
- dátumy leasingov,
- dátum obstarania automobilu,
- dátum predaja automobilu.

### Úprava automobilu

Používateľ upravuje automobil cez rovnaký formulár ako pri pridávaní nového automobilu. Avšak pri úprave sa používateľovi načítajú aktuálne informácie, ktoré sú uložené v databáze.

### Zoznam automobilov

V sekcii vozový park je na úvodnej stránke zobrazený zoznam vozidiel. Cez túto stránku je umožnený prístup k akciám upraviť automobil, zobraziť detail automobilu, vymazať alebo vyradiť automobil. Taktiež je možné zoradiť záznamy podľa značky, špz alebo vlastníctva. Majiteľovi ostávajú určité záväzky k automobilu aj potom ako automobil vyradí z prevádzky, preto vznikla možnosť automobil vyradiť. Ten potom ostane v databáze až pokiaľ používateľ vozidlo úplne nevymaže.



Pri vozidlách sa tiež evidujú poruchy a poistky, s ktorými sa dajú vykonať akcie upraviť, vymazať a pridať. Pridávanie týchto položiek sa vykonáva prostredníctvom formulárov. K zoznamom porúch a poistiek sa používateľ dostane cez detail jednotlivých automobilov. Takisto ako v hlavných sekciách sú zoznamy týchto údajov zobrazované pomocou tabuliek. Zoznamy si môže používateľ zoradovať.

## 6.9 Udalosti


Systém umožňuje jednoduchú evidenciu udalostí firiem, čomu sa venuje nasledujúca kapitola.

### Vytvorenie novej udalosti

Na vytvorenie novej udalosti sa používa taktiež formulár s informáciami o názve, dátume konania a k prípadným poznámkam tejto udalosti. Validácia sa tu vzťahuje iba na povinnosť atribútov, ktorými sú názov a dátum.

### Úprava udalosti

Udalosť je jednoduché zmeniť. Pre zmeny sa používa rovnaký formulár ako pri pridaní novej udalosti.

**Upraviť udalosť** 

\*Názov:

\*Dátum:

Popis:

Obr. 16: Úprava udalosti v pizzerii

### **Zoznam udalostí**

Udalosti sa zobrazujú používateľovi ako zoznam, ktorý je zobrazený pomocou tabuľky. Jeden záznam obsahuje informácie o názve a dátume udalosti. Zároveň môže vykonávať operácie ako úpravu, zobrazenie detailu a vymazanie.

## **6.10 Sumár**

Sumár poskytuje majiteľovi príležitosť na evidenciu ním vybraných nákladov a výnosov. Tieto náklady a výnosy systém najskôr jednotlivo sčíta a potom ponúkne majiteľovi informáciu o rozdiel medzi nimi.

### **Vytvorenie nákladov a výnosov**

Pretože náklady aj výnosy majú rovnaké atribúty je možné pre vytvorenie využívať rovnaký formulár. Pri nákladoch a výnosoch sa eviduje skratka do maximálnej dĺžky deviatich znakov ďalej popis, dátum a suma.

### **Úprava nákladov a výnosov**

Úprava prebieha rovnako ako v predchádzajúcich sekciách taktiež je tu využívaný rovnaký formulár ako pri vytváraní nových nákladov a výnosov s aktuálnymi informáciami uloženými v databáze.

### **Zoznam nákladov a výnosov**

Náklady a výnosy rozdeľujú obrazovku na dve časti. V ľavej časti sú náklady a v pravej výnosy. Rozloženie je podobné ako sa používa pri ilustrácii účtov v účtovníctve. V týchto záznamoch je možné vyhľadávať a taktiež sa používateľovi zobrazuje rozdiel nákladov a výnosov. V zoznamoch sú zobrazené informácie o skratke, dátume a sume. Používateľ má možnosť vykonať akcie zobraziť detail a vymazať náklady alebo výnosy.

**Zoznam nákladov a výnosov**

Hľadať:

Náklad ▲	Dátum ⇅	Suma ⇅		
Energie	04.03.2015	35 €		
PHM	02.03.2015	100 €		
PHM	16.03.2015	35.2 €		

náklad

Hľadať:

Výnos ▲	Dátum ⇅	Suma ⇅		
Predaj	04.03.2014	35.2 €		
Predaj	13.03.2015	100.1 €		

výnos

---

ΣV - ΣN -34.9 €

Obr. 17: Zoznam zamestnancov v pizzerii

## 6.11 Formuláre

Formulár sa pridáva do prezenteru pomocou takzvaných továrničiek. Továreň na komponenty predstavuje elegantný spôsob, ako komponenty vytvárať vo chvíli, keď sú skutočne potrebné. Nette využíva kľúčový názov funkcie `createComponent<Name>()`, kde `<Name>` je názov vytvárajúcej komponenty, ktorá komponentu vytvorí a vráti. Komponenta je potom vykreslená v šablóne. Metóde `createComponent<Name>` je predávaný voliteľný parameter s názvom komponenty, ktorú vytvára. V šablóne je možné získať a vykresliť komponentu pomocou makra `control` alebo ručne pomocou HTML tagov. Formulár po odoslaní a úspešnej validácii zavolá metódu `<Name>Succeeded()`. Ako prvý parameter tejto metódy je predaný samotný formulár. Do druhého parametra sa predávajú odoslané hodnoty formulára v objekte `ArrayHash`. V tele tejto metódy sa spracovávajú informácie z formulára pomocou funkcií v modeloch (Nette Framework, 2014).

## 6.12 Validácie

Kontrolu nad tým, či boli vyplnené povinné polia vykonáva v Nette metóda `setRequired()`, ktorej voliteľný atribút je text chybovej hlášky, ktorý potom informuje používateľa o nutnosti vyplnenia poľa, v prípade že dané pole nevyplnil.

Ďalšie validačné pravidlá sa pridávajú metódou `addRule()`, ktorej prvým argumentom je pravidlo hovoriace o tom, čo chceme overovať. V evidenčnom systéme sú využívané pravidlá `Form::PATTERN` a `Form::EMAIL`. Prvé pravidlo používa regulárny výraz pre kontrolu telefónneho čísla a druhé kontroluje správnosť emailu. Druhý argument funkcie `addRule()` je opäť text hlášky, ktorá sa zobrazí, pokiaľ hodnota validáciou neprejde. Tretím argumentom sa potom konkretizuje validačné pravidlo napríklad rozsah určitých hodnôt, alebo ako v prípade telefónneho čísla regulárny výraz.

Pre niektoré validácie preddefinované metódy nestačia, napríklad overenie správnej kombinácie hodnôt vo viacerých prvkoch formulára, a preto je nutné si vytvoriť vlastné validácie. V evidenčnom systéme sú to napríklad funkcie, ktoré kontrolujú vznik a zánik pracovného pomeru, začiatok a koniec splácania leasingu alebo dátum narodenia. Tieto funkcie sa potom na formulár naväzujú pomocou udalosti *onValidate*

Validácie využívajú triedu *Nette\Utils\Validators*, ktorá obsahuje potrebné funkcie pre validáciu dát (Nette Framework, 2014).

### 6.13 Rezervácia automobilov

Po dohode s majiteľom boli rezervácie vytvorené pomocou google kalendárov. Ku každému automobilu vo firme Auto Žilina bol vytvorený google kalendár. Každý google kalendár má ID, pomocou ktorého sa konkrétny automobil v evidenčnom systéme spároval s konkrétnym kalendárom. Pomocou google API sa aplikácia prepojila s google účtom. Rezervácia sa tak vytvára v google kalendári a informácie o nej sa zobrazujú u konkrétneho automobilu v evidenčnom systéme.

← ULOŽIŤ Zahodiť

**Valuch**

5/17/2015 2:30am do 5:30am 5/17/2015 Časové pásmo

Celý deň  Opakovať...

Podrobnosti udalosti [Vyhľadať čas](#)

Kde Žilina, Slovenská republika

Videohovor [Pridať videohovor](#)

Kalendár Mercedes ▾

Popis

Farba udalosti

Upozornenia Kontextové okno ▾ 30 min ▾ ×

[Pridať upozornenie](#)

Obr. 18: Vytváranie rezervácií pomocou google kalendára

## 7 Záver

Cieľom bakalárskej práce bolo vytvoriť evidenčný systém ako webovú aplikáciu pre majiteľa firiem Pizza Fantasia, Taxi Fantasia a Auto Žilina. Aplikácia má slúžiť ako systém pre elektronickú evidenciu informácií o zamestnancoch, stave vozového parku, udalostí, nákladov a výnosov zvolených majiteľom.

Prvá časť práce sa venuje prehľadu technológií, ktoré sú využívané pri tvorbe webových aplikácií ako sú jazyk PHP, HTML, JavaScript, kaskádové štýly a databázové systémy. Ďalšia časť je zameraná na súčasný stav, kde sú predstavené firmy majiteľa, požiadavky pre tvorbu evidenčného systému a dostupné aplikácie, ktoré by majiteľ mohol potenciálne využívať pre elektronickú evidenciu. Kapitola 4.3 obsahuje informácie o nesplnených požiadavkách vybraných systémov, vďaka čomu boli tieto systémy označené ako nevhodné vzhľadom k potrebám majiteľa. Na základe tejto analýzy bol navrhnutý nový systém.

Následne je opisovaný postup pri návrhu tohto systému. Databázový model bol navrhnutý pomocou ER diagramu. Na návrh používateľského rozhrania boli použité mockupy v programe Balsamiq Mockups.

Podľa návrhu bol vytvorený evidenčný systém ako webová aplikácia. Systém používa responzívny dizajn, čo umožňuje majiteľovi zjednodušený prístup k systému z rôznych zariadení. Pri implementácii bol využitý PHP framework Nette. Aplikácia bola sprevádzkovaná online na webovej adrese <http://es3f.php5.cz/> ako testovacia verzia s prihlasovacím menom „Majitel“ a heslom „heslo“.

Evidenčný systém spĺňa všetky požiadavky, ktoré majiteľ firiem požadoval, avšak po prezentácii výslednej práce majiteľovi vznikli nové pripomienky. Ide napríklad o úpravu niektorých atribútov, vzhľadu formulárov alebo odlišný spôsob akým je používateľ informovaný o vykonaných zmenách. Po vykonaní týchto zmien môže majiteľ začať využívať systém pre svoju evidenciu.

Do budúca je možné systém rozšíriť o nové funkcionality ako napríklad export jednotlivých záznamov, či celý zoznam údajov alebo pridať k evidencii novú firmu. Systém je možné upraviť tak, aby ho mohli používať aj zamestnanci podnikov, napríklad vytvoriť používateľské role, ktoré budú mať práva k evidencii iba jednej z firiem. S majiteľom spoločnosti sa o týchto možnostiach diskutuje a prípadne budú implementované.

## 8 Literatúra

- BERNARD, B. *Úvod do architektury MVC - Zdroják* Zdroják - o tvorbě webových stránek a aplikací [online]. 7.5.2009 [cit. 2014-12-05]. Dostupné z: <http://www.zdrojak.cz/clanky/uvod-do-architektury-mvc/>.
- BROŽA, P. A BURANSKÝ I. *Programování WWW stránek pro úplné začátečníky*. 3. aktualiz. a rozš. vyd. Brno: Computer Press, 2003, viii, 202 s. ISBN 80-722-6818-X..
- CORONEL C., MORRIS S., ROB P. *Database systems: design, implementation, and management*. 10e [edition]. Boston, MA: Course Technology/Cengage Learning, c2013, xxvi, 726 p. ISBN 11-335-2679-9..
- Dokumentace Nette Framework* [online]. 2014. [cit. 2015-05-12]. Dostupné z: <http://doc.nette.org/cs/2.2/>.
- CHACON S. *Pro Git* New York: Apress, xxi, 265 s. ISBN 978-1-4302-1833-3.
- Informačný systém a ERP pre každú firmu | abra.sk* [online]. © 2014 [cit. 2015-04-26]. Dostupné z: <http://www.abra.sk/informacne-systemy>.
- JKR - dodavatel BYZNYS ERP* [online]. 2015 [cit. 2015-04-28]. Dostupné z: <http://www.jkr.cz/>.
- JQuery* [online]. Copyright 2015 [cit. 2015-04-14]. Dostupné z: <https://jquery.com/>.
- KOSEK, J. *PHP - tvorba interaktivních internetových aplikací: podrobný průvodce*. Vyd. 1. Praha: Grada, 1999, 490 s. Průvodce (Grada). ISBN 80-716-9373-1..
- Linux Software* [online]. 27.5.2004 [cit. 2014-12-05]. Dostupné z: [http://www.linuxsoft.cz/article.php?id\\_article=171](http://www.linuxsoft.cz/article.php?id_article=171).
- LI A CHEN *Modeling and analysis of enterprise and information systems: from requirements to realization*. New York: Springer, xviii, 397 p. ISBN 978-354-0895-565..
- Oracle | Hardware and Software, Engineered to Work Together* [online]. Copyright © 2011-2012 [cit. 2014-12-06]. Dostupné z: [http://docs.oracle.com/cd/E23010\\_01/wci.1034/e14110/portaluiarchitecture.htm](http://docs.oracle.com/cd/E23010_01/wci.1034/e14110/portaluiarchitecture.htm).
- PHP: Hypertext Preprocessor* [online]. © 2001-2014 [cit. 2014-12-03]. Dostupné z: <http://cz1.php.net/manual/en/intro-what-is.php>.
- Programujte.com* [online][cit. 2015-05-08]. Dostupné z: <http://programujte.com/clanek/2008022000-php-frameworky/>.

- PROCHÁZKA, D. *Oracle: průvodce správou, využitím a programováním nad datábovým systémem* 1. vyd. Praha: Grada Publishing, 2009, 168 s. ISBN 978-80-247-2762-2..
- Quick 'n' Comfortable Web Development in PHP | Nette Framework [online]* 2015. [cit. 2015-05-09]. Dostupné z: <http://nette.org/cs/>.
- SANJAY MISHRA A ALAN BEAULIEU *Mastering Oracle SQL*. 2nd ed. Sebastopol, CA: O'Reilly, c2004, xvii, 472 p. ISBN 05-960-0632-2..
- ShopCentrik [online]*. 2015 [cit. 2015-04-16]. Dostupné z: <http://www.shopcentrik.cz/slovník/dynamicke-webove-stranky.aspx>.
- SitePoint PHP – Learn PHP, MySQL, SOAP and more [online]* [cit. 2015-05-09]. Dostupné z: <http://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/>.
- The Tech Terms Computer Dictionary [online]*. © 2015 [cit. 2015-04-13]. Dostupné z: <http://techterms.com/definition/css>.
- The Tech Terms Computer Dictionary [online]*. © 2015- [cit. 2015-04-09]. Dostupné z: [http://techterms.com/definition/web\\_application](http://techterms.com/definition/web_application).
- The Web framework for perfectionists with deadlines | Django [online]* [cit. 2015-05-09]. Dostupné z: <https://code.djangoproject.com/wiki/DjangoSuccessStoryBitbucket>.
- TIOBE Software: The Coding Standards Company [online]*. © 2015 [cit. 2015-04-12]. Dostupné z: <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>.
- Tvorba webu | Adaptic [online]*. © 2005–2015 [cit. 2015-04-14]. Dostupné z: <http://www.adaptic.cz/znalosti/slovnicek/mysql/>.
- Twitter Bootstrap [online]* 2015. [cit. 2015-05-10]. Dostupné z: <http://getbootstrap.com/>.
- VÁCLAVEK, P. *JavaScript: hotová řešení*. 1. vyd. Brno: Computer Press, 2003, 255 s. ISBN 80-722-6854-6..
- VHsoft - ERP třídy BYZNYS [online]*. © 1996 - 2015 [cit. 2015-04-30]. Dostupné z: <http://www.vhsoft.cz/moduly/byznys-online>.
- WILLIAMSON, K. *Learning AngularJS [online]* 1. vyd., © 2015 ISBN 978-1-491-91675..

# Prílohy



## A Priložené CD

Priložené CD obsahuje:

- Zdrojové kódy aplikácie.
- Mockupy vytvorené pri návrhu používateľského rozhrania.