

# **Webová aplikace pro organizaci práce ve společnosti zabývající se inventurami**

**Bakalárska práca**

**Vedúci práce:**

**Ing. Pavel Turčínek, Ph.D.**

**Martin Bendík**

**Brno 2016**



V prvom rade by som sa chcel poďakovať svojmu školiteľovi Ing. Pavlovi Turčínkovi Ph.D. za odbornú pomoc, cenné informácie a rady, a v neposlednom rade za ochotu. Taktiež by som sa chcel poďakovať rodine a blízkym osobám, ktoré ma pri celom štúdiu a pri písaní tejto práce motivovali a podporovali.



### **Čestné prehlásenie**

Prehlasujem, že som prácu: **Webová aplikace pro organizaci práce ve společnosti zabývající se inventurami**

vypracoval samostatne a všetky použité zdroje a informácie uvádzam v zozname použitej literatúry. Súhlasím, aby moja práca bola zverejnená v súlade § 47b zákona č. 111/1998 Sb., o vysokých školách v znení neskorších predpisov a v súlade s platnou *Směrnici o zveřejňování vysokoškolských závěrečných prací*.

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 použití diela inou osobou (subjektom) si vyžiadam písomné stanovisko univerzity, že predmetná licenčná zmluva nie je 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 15. mája 2016

---



## **Abstract**

Bendík, M. Web application for organizing work in company engaged in inventories. Bachelor thesis. Brno: Mendel University in Brno, 2016.

Bachelor thesis is focused on analysis and design of information system in the form of a web application in order to work organization in company engaged in inventories. The literature review explains the basic concepts necessary for the development of information systems and an overview of implementing tools and technologies. The practical part describes the analysis tools to support the creation of web applications, analysis of the company, the design of the information system, implementation, deployment and testing of current solution.

## **Keywords**

Information system, PHP, JavaScript, SQL, ERD, activity diagram, use-case diagram, MVC.

## **Abstrakt**

Bendík, M. Webová aplikace pro organizaci práce ve společnosti zabývající se inventurami. Bakalářská práce: Mendlova univerzita v Brně, 2016.

Bakalářská práce je zaměřená na analýzu a návrh informačního systému ve formě webové aplikace za účelem organizace práce v společnosti zabývající se inventurami. V literární rešerši jsou vysvětlené základní pojmy potřebné při vývoji informačních systémů a přehled implementačních nástrojů a technologií. Praktická část práce popisuje analýzu nástrojů pro podporu tvorby webových aplikací, analýzu společnosti, návrh informačního systému, implementaci, zavedení a testování stávajícího řešení.

## **Klíčové slová**

Informační systém, PHP, JavaScript, SQL, ERD, diagram aktivit, diagram případů užití, MVC.





# Obsah

<b>1</b>	<b>Úvod, cieľ a metodika práce</b>	<b>11</b>
1.1	Úvod .....	11
1.2	Cieľ práce.....	11
1.3	Metodika práce .....	12
<b>2</b>	<b>Literárna rešerš</b>	<b>13</b>
2.1	Informačný systém .....	13
2.2	Diagram prípadov užitia (Use-case diagram) .....	15
2.3	Diagram aktivít (Activity diagram) .....	15
2.4	Entitne-relačný diagram (ERD) .....	17
2.5	Jazyk PHP a dynamické webové aplikácie .....	18
2.6	PHP Framework a architektúra MVC .....	18
2.7	Jazyk SQL, MySQL a phpMyAdmin.....	19
2.8	Javascript .....	20
<b>3</b>	<b>Analýza a návrh</b>	<b>21</b>
3.1	Porovnanie PHP frameworkov.....	21
3.2	Analýza spoločnosti.....	24
3.3	Špecifikácia požiadaviek .....	27
3.4	Návrh entitne-relačného diagramu.....	29
3.5	Návrh diagramov aktivít .....	30
3.6	Návrh webového rozhrania .....	32
<b>4</b>	<b>Implementácia a výsledky</b>	<b>33</b>
4.1	Prihlasovanie a odhlasovanie užívateľ'ov.....	33
4.2	Privilégia užívateľ'ov .....	33
4.3	Zoznamy a vyhľadávanie v nich .....	34
4.4	Vytváranie inventúr.....	35
4.5	Detail zákazníka .....	36
4.6	Prihlasovanie na inventúry.....	37

---

4.7	Vytváranie a vkladanie reportov zamestnancov .....	37
4.8	Štatistiky a grafy .....	39
4.9	Kontrola aktivity zamestnancov .....	40
<b>5</b>	<b>Testovanie a zavedenie</b>	<b>41</b>
5.1	Testovanie .....	41
5.2	Zavedenie .....	42
<b>6</b>	<b>Diskusia</b>	<b>45</b>
6.1	Ekonomické zhodnotenie a prínosy .....	45
6.2	Možné rozšírenia .....	45
6.3	Iné využitie aplikácie .....	46
<b>7</b>	<b>Záver</b>	<b>47</b>
<b>8</b>	<b>Literatúra</b>	<b>49</b>
<b>9</b>	<b>Zoznam obrázkov</b>	<b>53</b>
<b>10</b>	<b>Zoznam tabuliek</b>	<b>55</b>
<b>A</b>	<b>Obsah priloženého DVD</b>	<b>58</b>
<b>B</b>	<b>URL adresa a prihlasovacie údaje</b>	<b>59</b>
<b>C</b>	<b>Diagram prípadov užitia</b>	<b>60</b>
<b>D</b>	<b>Rezervácie a vozidlá</b>	<b>61</b>
<b>E</b>	<b>Profil zamestnanca</b>	<b>62</b>
<b>F</b>	<b>Detail inventúry</b>	<b>63</b>
<b>G</b>	<b>Výplatné pásy</b>	<b>64</b>
<b>H</b>	<b>Výstup pre archiváciu</b>	<b>65</b>

# 1 Úvod, cieľ a metodika práce

## 1.1 Úvod

V súčasnej dobe si väčšina ľudí myslí, že za peniaze sa dá kúpiť skoro všetko. Jeden si myslí, že keď má veľa peňazí, zaplatí si drahú zdravotnú starostlivosť, iný si myslí, že za peniaze si kúpi šťastie. Ale jednu vec si nemôže kúpiť nikto, a to čas. Čas sa nedá zväčšiť, zmenšiť, ale ani predať, alebo kúpiť. Dnešným trendom je vytvárať alebo produkovať veci viac a viac efektívnejšie, aby sme z rovnakého času dokázali vyťažiť maximum.

Niektoré spoločnosti sa napríklad zaoberajú pásovou výrobou. Snažia sa nájsť také výrobné optimum, aby im to maximalizovalo zisky. Iné spoločnosti sa zameriavajú na marketing a reklamu, iné zase na kvalitu vlastných výrobkov. V tejto modernej dobe, kde sa už málo vecí robí písomnou papierovou formou, je jednou z efektívnejšieho využívania zdrojov v spoločnostiach aj kvalitný a výkonný informačný systém a webová aplikácia, ktoré plnia svoju úlohu dokonalo. Jednak dokážu ušetriť nie malé finančné prostriedky, alebo zvýšiť produktivitu práce, ale hlavne dokážu šetriť tým najdôležitejším, a to časom. Jednou z týchto spoločností ktoré nevyužívajú túto možnosť informačných systémov, je aj spoločnosť, pre ktorú bude tento informačný systém vytváraný.

Táto spoločnosť sa zaoberá kompletnými inventúrami všetkých druhov tovarov, či už potravín, oblečenia, ale aj interných zariadení nehnuteľných objektov. V tejto spoločnosti pracuje aktuálne niekoľko typov zamestnancov, ktorí sa môžu prihlasovať na inventúry. Všetky informácie týkajúce sa či už inventúr, spomínaných zamestnancov, ale aj vozidiel, sú uložené v súboroch a v priečinkoch, rozdelených podľa jednotlivých rokov. Samozrejme existuje jednoduchý hierarchický systém uloženia týchto dokumentov. Problémom však je vyhľadávanie v jednotlivých dokumentoch a integrita dát, ktoré obsahujú.

## 1.2 Cieľ práce

Keďže všetky akcie medzi zamestnancami v spoločnosti prebiehajú zvyčajne pomocou e-mailovej komunikácie, je cieľom tejto bakalárskej práce vytvoriť informačný a rezervačný systém, za účelom efektívnejšieho využívania pracovného času. Súčasťou tohto cieľa je taktiež zabezpečiť integritu dát, a to z toho dôvodu, že dáta sú zvyčajne uložené v súboroch tabuľkového softwaru Excel, kde často dochádzalo k chybám vo vyplňovaní. Aplikácia musí taktiež poskytovať rôzne evidencie, či už zákazníkov, zamestnancov, ale aj vozidiel. Táto webová aplikácia nemá slúžiť na prezentáciu spoločnosti, preto je potrebné, aby jednotlivé rozhrania boli graficky a vizuálne minimalistické, ale zároveň musia poskytovať intuitívne ovládanie.

Rozširujúcou funkciou aplikácie budú rôzne štatistiky v grafickom zobrazení. Tie budú slúžiť primárne na ekonomické zhodnotenie jednotlivých rokov a rozho-

dovanie manažéra o vývoji spoločnosti, či sa už bude jednať o prijatí nových zamestnancov, ale aj o prípadných budúcich investíciách spoločnosti. Sekundárne budú prístupné osobné štatistiky každého zamestnanca, ktoré budú slúžiť iba na informatívne účely.

### 1.3 Metodika práce

Ešte pred vyvíjaním informačného systému a následnou implementáciou vo forme webovej aplikácie je za potreby oboznámenie sa so základnými pojmami, ktoré súvisia s touto problematikou. Taktiež bude za potreby vytvoriť analýzu dostupných nástrojov podľa vybraných kritérií, ktoré budú nie len zamerané na základné a technické parametre ako je licencia, výkon, bezpečnosť a pod., ale aj odporúčenia, týkajúce sa potrebných minimálnych znalostí pre využívanie týchto nástrojov.

Postup tvorby webovej aplikácie bude korešpondovať so životným cyklom vývoja informačných systémov. To znamená, že už na začiatku bude za potreby detailne preskúmanie aktuálnej situácie v spoločnosti, ich podnikovými pravidlami a taktiež novými požiadavkami. Následne bude vytvorený návrh informačného systému v podobe diagramu prípadov užitia (Use-case diagram), ktorý bude reprezentovať všetky doposiaľ zistené požiadavky od spoločnosti. Pre jednotlivé prípady užitia sa vytvoria diagramy aktivít (Activity diagram) a nakoniec entitne-relačný diagram (Entity-relationship diagram), ktorý bude reprezentovať štruktúru databázy a uloženie dát.

Pre implementáciu návrhu bude použitý nástroj, ktorý obстоjí v analýze s najlepším hodnotením. Pomocou tohto nástroju sa implementujú všetky potrebné časti aplikácie. Nevyhnutnou súčasťou implementácie bude skriptovací jazyk PHP a taktiež jazyk SQL, ktorý je potrebný pre komunikáciu s databázou. Pre vizuálne spracovanie webovej aplikácie bude použitý značkovací jazyk HTML a pre grafickú úpravu kaskádové štýly CSS. Taktiež bude použitý skriptovací jazyk JavaScript, a to hlavne na kontrolu integrity dát zadávaných v HTML formulároch.

Ďalším krokom pri vývoji informačného systému je testovanie implementovaného návrhu. Otestované budú technické parametre aplikácie, ako je napríklad čas potrebný pre zobrazenie jednotlivých častí aplikácie. Taktiež bude testovanie prebiehať z pohľadu užívateľa, čím sa odstránia funkčné nedostatky aplikácie. Na záver vývoja webovej aplikácie bude potrebné nájsť vhodné obdobie pre zavedenie systému do spoločnosti za účelom minimalizovania možných problémov. Následne bude trvať skúšobné obdobie, kde sa rozhodne, či je systém definitívne spoľahlivý.

## 2 Literárna rešerš

Tak ako výroba automobilu musí prejsť istými fázami, tak isto musí mať aj samotný informačný systém svoj životný cyklus vývoja. Či už vo forme jednoduchej webovej aplikácie, alebo vo forme zložitých informačných systémov. Pri tvorbe webovej aplikácie je veľmi dôležité dbať na všetky aspekty už pri analyzovaní a tvorbe návrhu informačného systému a webovej aplikácie. Preto je výhodné už na začiatku ozrejmiť, čo je to informačný systém, životný cyklus informačného systému a aké diagramy sú potrebné pri návrhu a implementácii informačného systému.

### 2.1 Informačný systém

Podľa internetovej encyklopédie WhatIs je informačný systém definovaný ako „zbierka technických a ľudských zdrojov, ktoré poskytujú úložisko, výpočet, distribúciu a komunikáciu pre informácie požadované od všetkých alebo od istých častí podniku“ (WhatIs, 2016).

Rábová (2008, str.8) vo svojej publikácii uvádza, že informačný systém je „komplex informácií, ľudí, použitých informačných technológií, organizácie práce, riadenia chodu systému a konečne technických prostriedkov a metód slúžiacich k zberu, prenosu, uchovaniu a ďalšiemu spracovaniu dát za účelom tvorby a prezentácii informácií.“

Inými slovami môžeme povedať, že informačný systém je konkrétna množina prvkov, ktorá sa môže skladať z ľudí, hardwaru, softwaru, dátovej základne, ale aj zo samotného riadenia systému. Táto množina je však spojená rôznymi vzájomnými väzbami, ktoré znázorňujú určité správanie a vzťahy medzi nimi (Koch a Ondrák, 2004, str. 4).

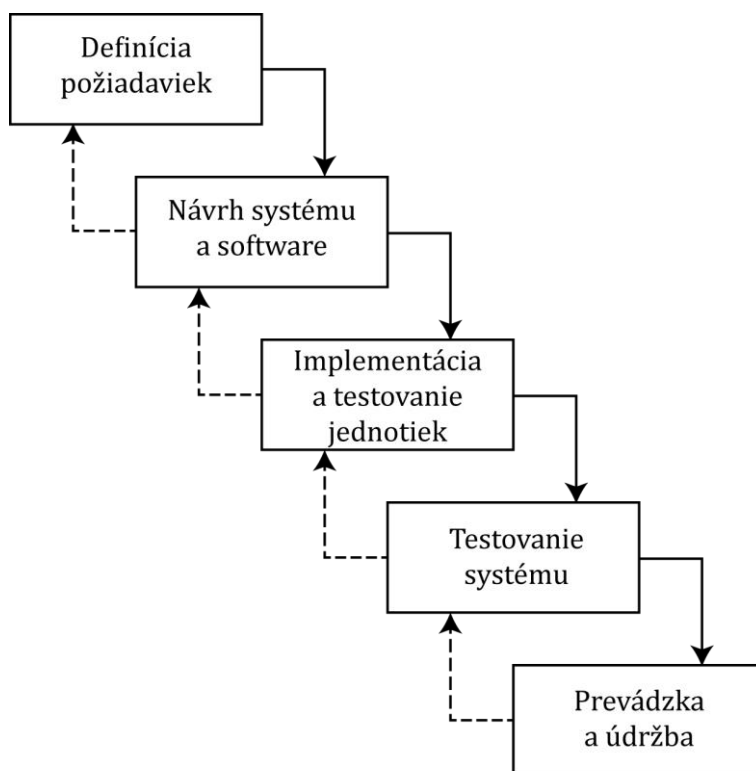
#### Životný cyklus vývoja informačného systému

Pod pojmom životný cyklus vývoja informačného systému môžeme rozumieť dynamický proces tvorby softwarového produktu. Takýto životný cyklus začína najskôr nápadom, ideou riešiť daný problém a končí likvidáciou, alebo výmenou za iný softwarový produkt. Medzi etapy životného cyklu patria:

- **Špecifikácia problému.** Cieľom je už od začiatku určiť, či je vhodné daný koncept systému realizovať. Výsledkom tejto etapy je podrobná špecifikácia daného konceptu, taktiež primárne určenie požiadaviek spoločnosti.
- **Analýza a návrh systému.** V tejto fáze je už vytvorená štruktúra systému. Požiadavky z predchádzajúcej etapy sú popísané podrobnejšie a sú rozdelené na požiadavky funkčné a dátové. Výsledkom tejto etapy je logický model systému, kde sú analyzované všetky systémové funkcie a rozdelenie systému na subsystémy.
- **Implementácia.** Cieľom tejto etapy je realizácia návrhu v implementačnom prostredí. Ide hlavne o tvorbu a vyladenie programov, ktoré sú použité v sys-

téme, ako aj o úpravy kúpených produktov. Neoddeliteľnou súčasťou tejto fázy je aj konverzia dát, čo vlastne značí, že z akých zdrojov sa budú dáta do databáz ukladať. Taktiež sa tu vytvára programová dokumentácia a príručka pre užívateľov.

- **Zavedenie.** Výsledkom tejto etapy je hladký a rýchly priebeh na nový systém. Patrí sem inštalácia technického a programového vybavenia, ale aj konverzia dát, ktoré sú v papierovej forme, na elektronické dáta a zaškolenie užívateľov. Ako náhle sa systém zavedie, prebieha skúšobná doba, ktorej výhodou je, že sa môže systém doladiť, čím nevzniknú žiadne straty v podniku.
- **Prevádzka a údržba.** Vytvoriť informačný systém alebo webovú aplikáciu neznamená len jeho vývoj, ale taktiež jeho prevádzka a údržba po plnom nasadení do podniku. Cieľom tejto fázy je priebežne aktualizovať systém, dostatočne zaistiť bezpečnosť a bezproblémový chod systému (Rábová, 2008, str. 38).



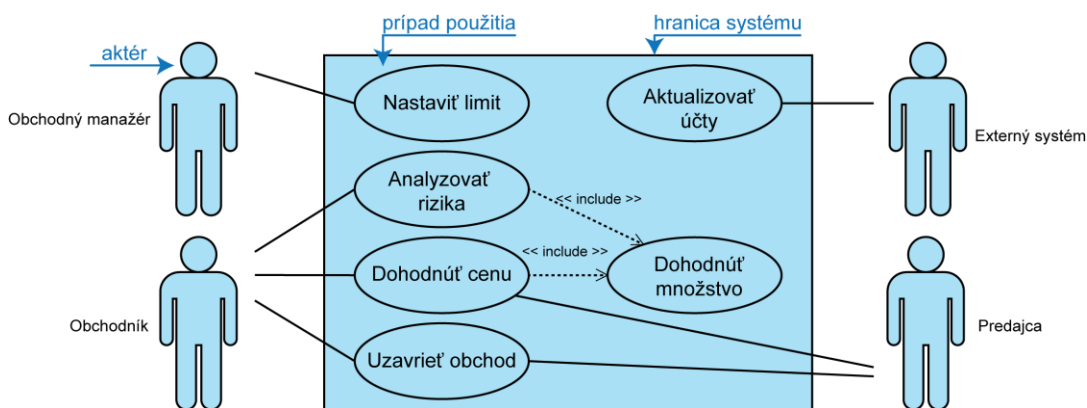
Obr. 1 Schéma vývoja softwarového produktu, model vodopád  
Zdroj Rábová, 2008, str. 40

## 2.2 Diagram prípadov užitia (Use-case diagram)

Pri modelovaní systému je najdôležitejším aspektom zachytiť dynamické správanie sa systému. Pod pojmom dynamické správanie sa systému rozumieme stav, kedy je daný systém v prevádzke. Takýto stav systému môžeme zachytiť pomocou Unified Modeling Language (UML). Jedným z UML nástrojov je aj use-case diagram, v preklade diagram prípadov užitia, pomocou ktorého môžeme modelovať tento dynamický charakter systému (Tutorialspoint.com, 2016).

Jan Kettenis (2007, str.4) uvádza, že „diagram prípadov užitia schematicky znázorňuje, ako by mal systém reagovať pri rôznych žiadostiach od jednej zo zúčastnených strán, nazývaných tiež ako hlavní aktéri.“ Na obrázku č. 2 je znázornený jednoduchý príklad diagramu prípadov užitia, ktorý sa skladá z:

- **aktérov.** Môžu predstavovať užívateľa, alebo externý spolupracujúci systém. Jedná sa vlastne o abstraktného užívateľa systému. V diagrame sa schematicky vyjadruje ako silueta postavy.
- **prípadov užitia.** Predstavujú službu, ktorú musí daný systém poskytovať a ktorá je vždy iniciovaná aktérom, čo znamená, že podnet musí vzniknúť už v okolí systému. V diagrame sa schematicky zakresľuje ako elipsa, ktorej popis odpovedá danej službe systému.
- **hranice systému.** Vyjadruje vymedzenie hranice medzi systémom a jeho okolím. V okolí systému môžu vznikať rôzne udalosti, na niektoré z nich môže systém reagovať a na iné nemusí. Ak vzniknutá udalosť predstavuje jeden z prípadov užitia, systém už na túto udalosť musí reagovať. V diagrame sa schematicky vyjadruje ako ohraničený obdĺžnik (Šimonová, Myšková a Jirava, 2006, str.35).



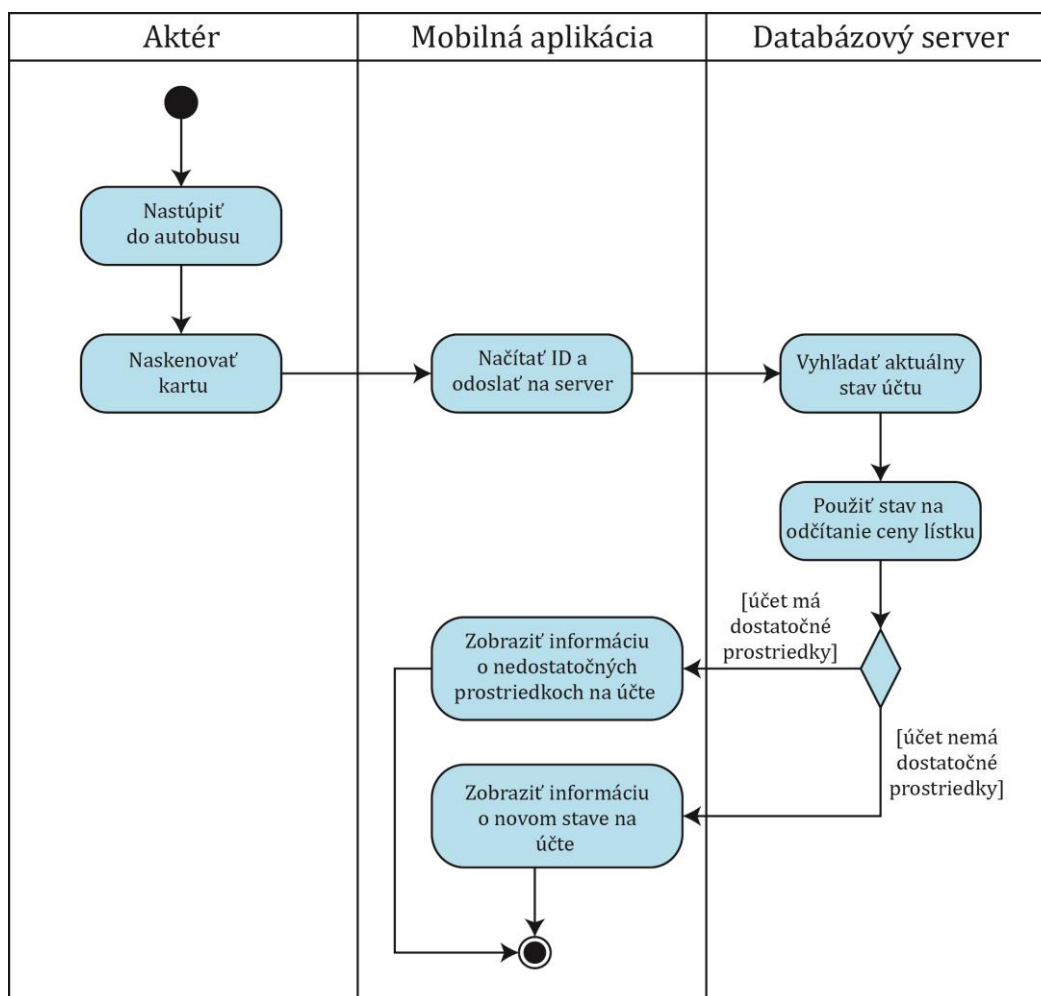
Obr. 2 Diagram prípadov užitia (Use-case diagram)  
Zdroj Fowler, 2004, str.103

## 2.3 Diagram aktivít (Activity diagram)

Druhým z nástrojov UML je diagram aktivít. V zásade sa v ňom znázorňuje prechod z počiatočného stavu na stav iný, prípadne koncový. Takýto prechod môže byť tvorený z jednotlivých krokov, ktoré môžu byť vykonávané súbežne alebo sekvenčne

(Škutová, 2011, str. 49). Tým sa odlišuje diagram aktivít od ostatných diagramov UML. Tie zobrazujú tok správy z jedného objektu na objekt iný, ale diagram aktivity znázorňuje tok správy z jednej aktivity na aktivitu inú (Palus, 2006). Taktiež je možné ho priradiť k ľubovoľnému prvku, či sa jedná už o prípad užitia, rozhranie, entitu alebo aj komponentu (Škutová, 2011, str. 49).

Obrázok č. 3 ilustratívne znázorňuje proces platby autobusového lístku pomocou mobilnej aplikácie.



Obr. 3 Diagram aktivity, proces zaplataenia autobusového lístku pomocou mobilnej aplikácie  
Zdroj Jin, 2015, str. 51

Tento proces, činnosť, začína u aktéra a znázorňuje sa ikonou čiernej bodky. Následne sa proces presúva pomocou riadiaceho alebo informačného toku, ktorý sa znázorňuje ikonou šípky, do jednotlivých stavov výpočtu. Tie sa schematicky zakresľujú ako obdĺžniky s popisom daného stavu, ktorý má byť vykonaný. Aktivita sa môže vetviť, kde podľa splnenia správnosti výsledku sa vyberie vetva, ktorá bude spracovaná. Vetvenie aktivity sa značí ikonou diamantu. Samotná aktivita končí v bode, ktorý je obklopený kružnicou (Sparx Systems, 2016).



## 2.4 Entitne-relačný diagram (ERD)

Tento diagram predstavuje koncept databázy z pohľadu koncového užívateľa. V tomto prípade databáza obsahuje základné zložky, a to entity, atribúty a vzťahy. Je to z toho dôvodu, že entity predstavujú reálne objekty, ktoré sú zainteresované s koncovým užívateľom (Coronel, Morris a Rob, 2010, str.100). Každá z týchto entít obsahuje svoje atribúty, ktoré sú jej dátovými prvkami. Tie môžu byť:

- **Totálne** – musia mať hodnotu pre každý výskyt entity
- **Parciálne** – nemusia mať hodnotu pre každý výskyt entity
- **Základné** – nedajú sa odvodiť z iných atribútov
- **Odvožené** – sú vypočítané alebo odvodené z iných atribútov (Šimonová a Panuš, 2007, str.28)

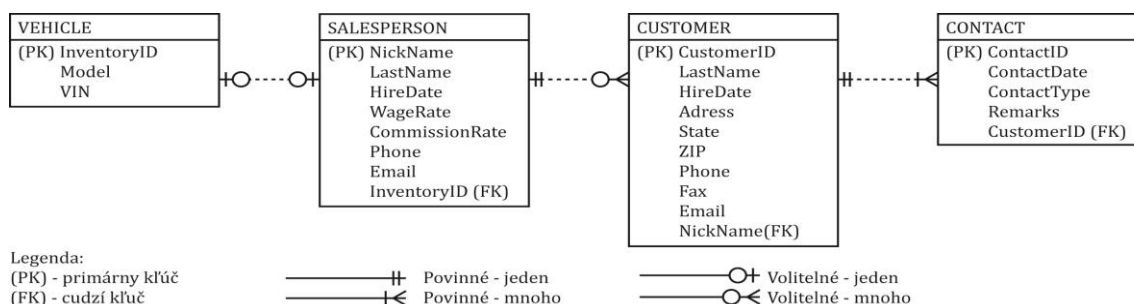
Nedeliteľnou a dôležitou súčasťou každej entity je identifikačný kľúč. Kľúčom môže byť atribút, alebo skupina atribútov, pomocou ktorých je možné jednotlivé entity identifikovať alebo sa odkazovať na entity iné. Najzákladnejšími dvomi sú:

- **Primárny kľúč** – slúži na primárne identifikovanie entity a je unikátny
- **Cudzí kľúč** – reprezentuje atribút v entite, ktorý sa odkazuje na iný primárny kľúč inej entity (Watt, 2012, str. 30).

Pôvodný entitne-relačný model sprístupňoval možnosť pridelenia vzťahov aj atribútom. V dnešnej praxi však môžu mať vzájomné vzťahy iba entity. Tie sa delia podľa kardinality na:

- **1:1** – prvej entite odpovedá maximálne jedna druhá entita
- **1:N** – prvej entite odpovedá n druhých entít, ale naopak druhej entite odpovedá maximálne jedna prvá entita
- **N:M** – n prvým entitám odpovedá m druhých entít a naopak

Na obrázku č. 4 je príklad entitne-relačného diagramu. Vzťahy medzi entitami sú znázornené úsečkou, kde podľa značenia sa jedná buď o povinné alebo voliteľné inštancie entity (Kroenke a Auer, 2015, str. 244).



Obr. 4 Ukážka entitne relačného diagramu s legendou, verzia IE  
Zdroj Kroenke a Auer, str. 243, str. 259

## 2.5 Jazyk PHP a dynamické webové aplikácie

Keď v roku 1994 Rasmus Lerdorf chcel sledovať počet návštevníkov na svojej webovej stránke, ešte netušil, že jeho skriptovací jazyk sa bude spolu so značkovacím jazykom HTML<sup>1</sup> používať dodnes. Jeho originálny názov bol „Personal Home Page“, ktorý bol neskôr nahradený názvom PHP: Hypertext Preprocessor (Ullman, 2008, Introduction XI). PHP bolo vyvíjané desiatky rokov, kde od verzie sú dostupné objektovo-orientované funkcie. Tie boli podstatne vylepšené až od verzie PHP 5 (Vasiliev, 2007, str.6). Najaktuálnejšou verziou je PHP 7 (php.net, 2016).

Webový dizajnéri oceňujú jeho dostupnosť a pohodlnosť, zatiaľ čo programátori oceňujú flexibilitu, silu, rozmanitosť a rýchlosť. (Lerford, Kevin, MacIntyre, 2006, Preface XII). Práve kvôli týmto vlastnostiam je jedným z najpoužívanejších skriptovacích jazykov pre podporu vývoja dynamických webových aplikácií, ktoré by mali spĺňať nasledujúce vlastnosti:

- Schopnosť reagovať na rôzne parametre (napr. čas v dni alebo verziu užívateľovho webového prehliadača)
- Schopnosť uchovávať dáta (napr. uchovávať prihlasovacie údaje po registrácii užívateľa)
- Obsahovať HTML formuláre
- Musia obsahovať rozhranie pre administrátora, kvôli prípadnej zmene obsahu
- Musia byť vybudované na statických webových stránkach (Ullman, 2008, Introduction X).

## 2.6 PHP Framework a architektúra MVC

Pod pojmom PHP framework môžeme rozumieť akúsi základnú platformu, pomocou ktorej môžeme vyvíjať webové aplikácie. Táto platforma zvyčajne poskytuje istú štruktúru aplikácie, knižnice a funkcie, ktoré sú preddefinované a použiteľné kdekoľvek v aplikácii (oneextrapixel.com, 2010).

Väčšina PHP frameworkov funguje na architektúre MVC, ktorá je často silne zameraná na objektovo-orientované programovanie. MVC je vlastne softwarový návrhový vzor, postavený na interakciu medzi tromi základnými komponentami:

- **Model.** Obsahuje všetky analytické časti aplikácie. Môžu to byť napríklad funkcie, ktoré určujú ako budú dáta uložené. Táto časť využíva aj služby tretích strán, za účelom splnenia požiadaviek. Model rieši problematiku interakcie medzi databázou a aplikáciou.
- **View.** Jednotlivé pohľady zobrazuje časti užívateľského rozhrania aplikácie. Tie môžu zahŕňať značkovací jazyk HTML, štýly CSS<sup>2</sup> ale aj JavaScript<sup>3</sup>. Inými

---

<sup>1</sup> Jazyk, ktorý je určený na vytváranie a formátovanie webových stránok

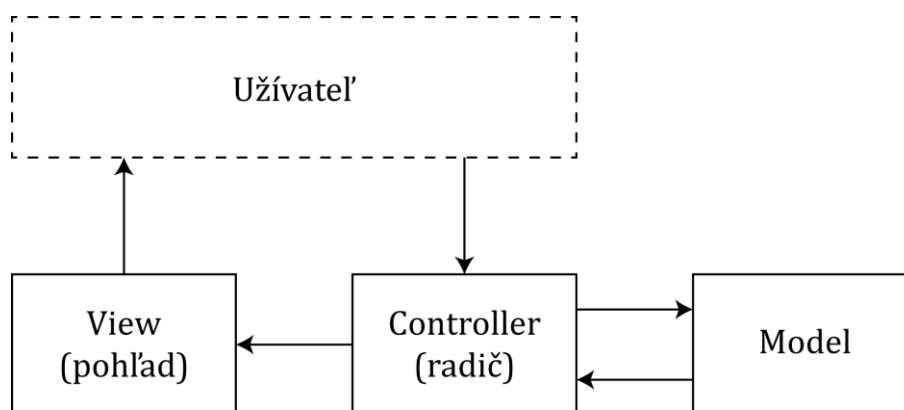
<sup>2</sup> Označuje sa ako jednoduchý mechanizmus, pomocou ktorého je možné vizuálne formátovať webové stránky

<sup>3</sup> Vysvetlené v kapitole 2.8

slovami sa v pohľadoch uchováva všetko čo užívateľ vidí vo svojom webovom prehliadači.

- **Controller.** V preklade radič alebo prepínač, je časť aplikácie, ktorá spája model a pohľad do jedného celku. Hlavnou funkciou je izolovanie logických požiadaviek z modelu, od časti rozhrania v pohľadoch, a taktiež rozhodovať, ako bude aplikácia reagovať na jednotlivé požiadavky od užívateľa.

Pokiaľ užívateľ vytvorí požiadavku na aplikáciu, tak je primárne odoslaná na radič, ktorý rozhoduje ako má byť požiadavka spracovaná. Následne je odoslaná na vybraný model. Tam sa spracuje a odošle sa výsledok z požiadavky späť do radiča, ktorý vyberie v ktorom pohľade sa výsledok požiadavky má zobrazit' užívateľovi (Pitt, 2012, str. 1).



Obr. 5 Architektúra MVC  
Zdroj Pitt, 2012, str.1

## 2.7 Jazyk SQL, MySQL a phpMyAdmin

Structured Query Language (SQL) vznikol v spoločnosti IBM Corporation a je používaný na definovanie a spracovávanie databázy. SQL nie je celkom programovací jazyk, ale jedná sa skôr o dátový podjazyk. To znamená, že obsahuje iba konštrukcie, pomocou ktorých je možné komunikovať a spracovávať databázu.

Tieto konštrukcie sa delia na niekoľko kategórii, z ktorých dve najdôležitejšie sú príkazy jazyka DDL a príkazy jazyka DML (Kroenke a Auer, 2015, str.128).

- **DDL (Data Definition Language).** Reprezentuje podmnožinu príkazov, pomocou ktorých je možné vytvárať, definovať, editovať a rušiť rôzne štruktúry a objekty v databázach. Taktiež obsahuje príkazy, pomocou ktorých je možné vytvárať a odoberať privilégia rôznym užívateľom.
- **DML (Data Manipulation Language).** Obsahuje podmnožinu príkazov určených na manipulácie s údajmi uložených v databáze. Patrí sem najpoužívanejší príkaz SELECT, ktorý slúži pre výber údajov z databázy. Taktiež sú tu definované príkazy pre vkladanie, aktualizáciu a zmazanie údajov (Lacko, 2011, str. 64).

Jedným z najpopulárnejších databázových platforiem, ktorý je dostupný s licenciou Open Source, je aj platforma MySQL. Jej výhodami je nie len Open Source licencia, ale aj jednoduchosť inštalácie (Stephens, Plew a Jones, 2010, str.39). MySQL platforma patrí spoločnosti Sun Microsystems, ktorá je vo vlastníctve spoločnosti Oracle (Lacko, 2011, str.36).

Pre správu databázového systému MySQL sa označuje za najoblúbenejší nástroj phpMyAdmin. Pomocou tohto nástroja sa dajú vytvárať tabuľky, editovať ich, rušiť, vytvárať záznamy, ale tiež importovať alebo exportovať databázy bez nutnosti písania SQL príkazov. Jeho alternatívami môžu byť napríklad mysql (textový interpret príkazov), MySQL Administrator (užívateľské rozhranie pre správu) ale aj MySQL Query Browser (užívateľské rozhranie pre testovanie príkazov SQL). Nevýhodou týchto alternatív je, že fungujú iba v prípadoch, kedy je možné k databáze pristupovať. Pokiaľ sú spustené na lokálnych zariadeniach, nie je možné pristupovať k MySQL serveru poskytovateľa internetových služieb, a to z bezpečnostných dôvodov (Kofler a Öggl, 2007, str. 233).

## 2.8 Javascript

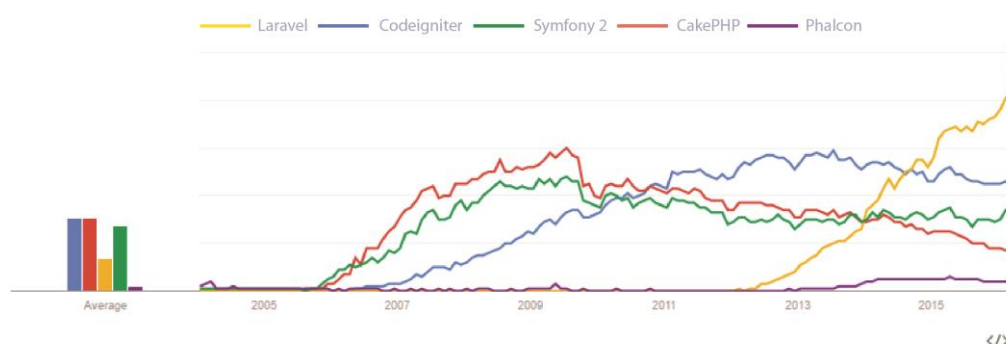
Javascript sa označuje za mocný a kompaktný skriptovací jazyk, a slúži na interakciu s webovými stránkami. Je z časti založený na programovacom jazyku Java, ktorý je objektovo orientovaný. Aj keď používa jeho syntax a sémantiku, jedná sa o samostatný dynamický jazyk (Zakas, 2006, str. 29). Medzi tie najpoužívanejšie dôvody používania JavaScriptu patrí:

- **Oživenie webových stránok** – pomocou JavaScriptu je možné oživiť statické webové stránky, či už pomocou interaktívnej galérie alebo sprístupnením nejakej služby.
- **Overovanie vstupných dát** – webové stránky, ktoré nepoužívajú JavaScript, sú vytvorené tak, že overovanie vstupných dát prebieha na strane serveru a v prípade nesprávnosti je odoslaná validačná chyba späť na stranu klienta. Tento problém rieši JavaScript, ktorý podporuje validáciu dát už na strane klienta a nemusí tak dochádzať k opakovaným požiadavkám na server.
- **Odl'ahčenie serverov** – JavaScript umožňuje preniesť radu výpočtov zo strany serveru, na stranu klienta, konkrétne do webového prehliadača. Či už sa jedná o jednoduchú kalkulačku, alebo zložitý systém na výpočet úverov, JavaScript značne prispieva k nižšej vyťažnosti serverov (Holzner, 2003, str. 35).

## 3 Analýza a návrh

### 3.1 Porovnanie PHP frameworkov

V roku 2015 vytvoril internetový blog webhostface.com v spolupráci so spoločnosťou Google<sup>4</sup> a s trendom GitHubu<sup>5</sup>, prieskum ohľadom najpopulárnejších PHP frameworkov (Kitipova, webhostface.com, 2015). Keďže spoločnosť Google patri medzi tie popredné spoločnosti sveta a GitHub má značnú komunitu webových vývojárov, ktorá presiahla už 14 miliónov užívateľov (GitHub.com, 2016), bolo vybraných na vlastnú analýzu päť PHP frameworkov, ktoré sú výsledkom tohto prieskumu a v nasledujúcom texte sú stručne popísané.



Obr. 6 Vývoj popularity PHP frameworkov podľa prieskumu Google Trends  
Zdroj: Google Trends

- **Laravel** je podľa prieskumu najpopulárnejším PHP frameworkom. Bol vydaný v roku 2012, preto sa radí medzi tie relatívne mladé frameworky. Jeho prednosťami sú jednoducho použiteľné funkcie, či už sa jedná o autentifikáciu, smerovanie a ukladanie premenných do vyrovnávacej pamäte (webrevison.com, 2016). Jeho ďalšími výhodami sú veľmi podrobná dokumentácia, jadro frameworku obsahuje rozsiahlu funkcionálnosť, ktorá sa dá samozrejme ešte rozšíriť o prídavné balíčky (Jakoubě, 2013, zdroj.cz). Radí sa medzi stredne rýchle PHP frameworky, ale taktiež poskytuje veľmi silné zabezpečenie. Aktuálne je dostupný pod MIT licenciou<sup>6</sup> a je vhodný pre značne pokročilých vývojárov (Kitipova, 2015, webhostface.com).

<sup>4</sup> Spoločnosť založená Larry Pagom a Sergeyom Brinom v roku 1998, ktorej prednosťou je hlavne inteligentný vyhľadávací nástroj Google (Google.com, 2016).

<sup>5</sup> Označuje za webovú službu podporujúcu vývoj softwaru pomocou bezplatného hostingu open-source projektov, viditeľných pre ostatných vývojárov (GitHub.com, 2016).

<sup>6</sup> Licencia slobodného softwaru, ktorá vznikla v Massachusettskom technologickom inštitúte (Haskell, 2014, opensource.com).

- **Codeigniter** bol vytvorený spoločnosťou Ellislab v roku 2006 a v súčasnosti je k dispozícii pod MIT licenciou. Je postavený na MVC architektúre. Jeho najväčšou výhodou je, že je jednoduchý na učenie pre začínajúcich vývojárov, ktorí ešte nepracovali so žiadnym PHP frameworkom (Arora, 2015, noeticforce.com). Jeho ďalšími výhodami sú jednoduchá inštalácia a konfigurácia, jeho kompatibilita na mnohých zdieľaných a špecializovaných platformách, čím minimalizuje konflikty vo verziách jazyku PHP (Monus, 2015, hongkiat.com). Taktiež pri vytváraní aplikácii umožňuje vývojárom voľnosť pri písaní kódu tým, že poskytuje iba veľmi základnú štruktúru (Kitipova, 2015, webhostface.com).
- **Symfony 2** sa radí medzi vysoko výkonné objektovo-orientované PHP frameworky a je k dispozícii pod MIT licenciou. Pri vývoji aplikácie si je možné vybrať zo sady PHP komponentov, ktoré má k dispozícii, ako sú napríklad vytváranie formulárov, objektovej konfigurácie, smerovania a verifikácie (Monus, 2015, hongkiat.com). Pýši sa značne rozsiahlou dokumentáciou a jeho nevýhodou je, že začínajúci vývojár potrebuje relatívne veľa času na naučenie sa tohto PHP frameworku. Je to spôsobené tým, že má zložitý adresárový systém a používa množstvo komponentov tretích strán (Kitipova, 2015, webhostface.com). Symfony 2 je určený prevažne na zložité projekty a do poručuje sa pre pokročilejších vývojárov, čo ale nemusí byť pravidlom. (Arora, 2015, noeticforce.com)
- **CakePHP** bol vydaný v roku 2005 poľským vývojárom Michalom Tatorynowiczom, ktorý sa značne nechal inšpirovať od Ruby on Rails<sup>7</sup>. Je založený na architektúre MVC a aktuálne je dostupný pod MIT licenciou (Porebski, Przystalski a Nowak, 2011, str.9). CakePHP poskytuje celkom dobrú bezpečnosť a radí sa medzi priemerne rýchle frameworky, ktoré nepotrebujú náročnú inštaláciu a konfiguráciu. Dôvodom nižšej rýchlosti oproti iným PHP frameworkom je, že niektoré operácie, ktoré používa obsahujú nezvyčajne veľa kódu. Na druhej strane jeho základná štruktúra zahrňuje už množstvo doplnkov, ktoré sú nevyhnutné pre fungovanie aplikácie (Kitipova, 2015, webhostface.com).
- **Phalcon** sa radí spolu s frameworkom Codeigniter medzi tie najvýkonnejšie PHP frameworky. Je na rozdiel od iných frameworkov napísaný v programovacom jazyku C a C++. Je relatívne jednoduchý, ale na druhej strane, musí webový vývojár ovládať základnú problematiku programovania v jazyku C. V porovnaní s ostatnými frameworkmi, Phalcon poskytuje značne menej funkcií, čo je dôvodom, že potrebuje náročnú konfiguráciu a inštaláciu (Kitipova, 2015, webhostface.com). Pracuje na MVC architektúre a poskytuje funkcie ako sú ORM<sup>8</sup>, žiadosti objektových knižníc, šablónovací vzor, ale taktiež stránkovanie (Cogneau, 2013, sitepoints.com). Od roku 2012 je dostupný

---

<sup>7</sup> Framework pre podporu vývoja webových aplikácií v programovacom jazyku Ruby.

<sup>8</sup> Objektové relačné mapovanie – zabezpečuje automatickú konverziu dát medzi relačnou databázou a objektovo orientovaným programovacím jazykom.

pod New BSD licenciou. To znamená, že na rozdiel od frameworkov, ktoré majú MIT licenciou, musí tiež re-distribúovaná binárna forma kódu obsahovať prehlásenie o autorských právach (Phalcon.com, 2016).

V nasledujúcom texte sú vybrané a odôvodnené kritéria pre porovnanie PHP frameworkov. Toto porovnanie je primárne určené pre webového vývojára, ktorý ešte nepracoval so žiadnym PHP frameworkom. Preto bolo vybraných nasledujúcich šesť kritérií. Tie sú rozdelené na tri kategórie a to základné parametre (hierarchia, jazyk, licencia), výkonnostné a bezpečnostné parametre, a nakoniec posledným kritériom, ktoré je zároveň oporným bodom pri analýze, je určenie užívateľa podľa jeho znalostí s vývojom webových aplikácií pomocou frameworkov.

Prvou z nich je architektúra, na ktorej je postavený samotný framework. Keďže všetky vybrané frameworky majú MVC architektúru, a jedna sa o najpopulárnejšiu a najpoužívanejšiu architektúru v oblasti webových aplikácií, je každý framework ohodnotený rovnako. Ďalším kritériom bol programovací, respektíve skriptovací jazyk, na ktorom sú frameworky postavené. Keďže všetky fungujú na báze skriptovacieho jazyka PHP, ktorý je robustný a rýchly, každý framework dostal päť bodov. Nakoniec, keďže sa jedná o analýzu pre začínajúcich vývojárov, bol strhnutý jeden bod frameworku Phalcon, a to z toho dôvodu, že pri kombinácii skriptovacieho jazyka PHP a programovacieho jazyka C++, môžu začínajúcim vývojárom vzniknúť komplikácie pri interakcii týchto dvoch jazykov. Tretím zo základných parametrov je licencia, kde všetky frameworky sú ohodnotených piatimi bodmi, okrem frameworku Phalcon, pretože pri tomto frameworku je nutné v každom binárnom súbore uvádzať licenčné práva, čo môže začiatočník často opomenúť.

Následne boli vybrané dva technické parametre. Prvým z nich je rýchlosť daného frameworku. Po preštudovaní vybraných frameworkov, je možné vydedukovať záver, že framework, ktorý je relatívne rýchly obsahuje spravidla efektívnejšie a jednoduchšie napísané funkcie, čo môže byť pre začiatočníka veľmi výhodné. Samozrejme tento záver nemusí súvisieť s bezpečnosťou daných frameworkov. Z nich najrýchlejší je Phalcon. Codeigniter sa radí medzi nadpriemerne rýchle a ostatné tri frameworky sú priemerne rýchle. Laravel, Codeigniter a Symfony 2 sa radia medzi jedny z najbezpečnejších frameworkov.

Posledným a najdôležitejším kritériom bolo rozhodnutie, aké znalosti musí mať webový vývojár, ktorý používa jednotlivé frameworky. Začínajúci vývojár sa môže najrýchlejšie naučiť frameworky Codeigniter a Phalcon. Frameworky Symfony 2 a CakePHP by sa učil relatívne dlho, a najdlhšie by sa učil framework Laravel. Samozrejme tieto závery sú relatívne a môžu sa odchyľovať od tejto analýzy, a to z dôvodu individuálneho zmýšľania a schopnosti sa učiť jednotlivých začínajúcich vývojárov.

Na obrázku č. 7 je graficky znázornené porovnanie vybraných frameworkov podľa jednotlivých kritérií. Každý framework mohol dosiahnuť maximálne 30 bodov, za každé porovnávacie kritérium 5 bodov. Najviac bodov dosiahol framework Codeigniter, a to 29 bodov. Je to hlavne z dôvodu, že sa jedná o nadpriemerne rýchly a vysoko bezpečný framework, a vďaka voľnosti a jednoduchej štruktúre najviac

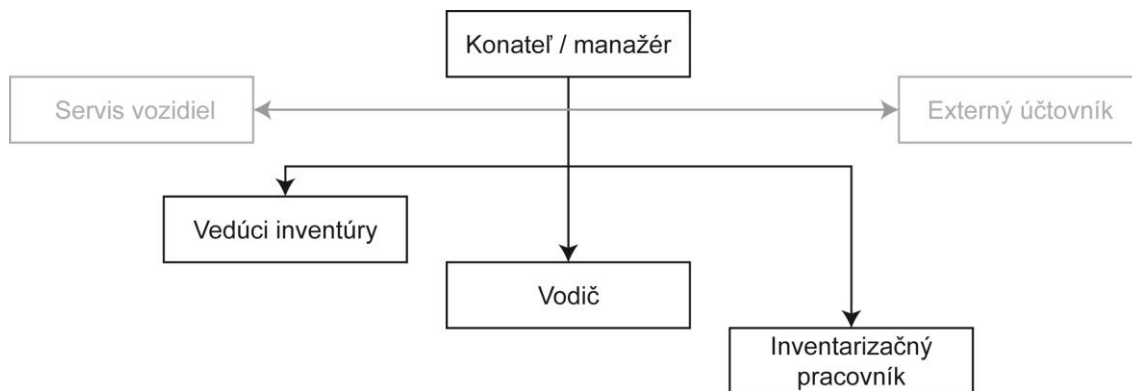
vyhovujúci pre začínajúcich vývojárov webových aplikácií. Z týchto dôvodov bol vybraný pre implementáciu informačného systému.



Obr. 7 Grafické porovnanie vybraných PHP frameworkov  
Zdroj Upravené z [webhostface.com/blog/popular-php-frameworks-2015/](http://webhostface.com/blog/popular-php-frameworks-2015/)

### 3.2 Analýza spoločnosti

Organizačná štruktúra spoločnosti, pre ktorú bol informačný systém v podobe webovej aplikácie vytvorený, má podobu klasickej spoločnosti s ručením obmedzeným. Táto spoločnosť je zameraná na vykonávanie inventúr. Na čele spoločnosti je konateľ, ktorý je tiež manažérom. Má na starosti najdôležitejšie aktivity a komunikuje s ostatnými subjektami výhradne pomocou e-mailovej alebo hlasovej komunikácie.



Obr. 8 Organizačná štruktúra spoločnosti s externými systémami (vyznačené sivou farbou)



Táto spoločnosť má aktuálne statickú webovú stránku, kde sú základné informácie týkajúce sa spoločnosti, prijímania nových zamestnancov a tiež obsahuje kontakt pre zákazníkov. Keďže tento informačný systém sa netýka webovej prezentácie spoločnosti, ale je zameraný na organizáciu práce a zlepšenie interného fungovania spoločnosti, bude vytvorený nezávisle na tejto webovej stránke.

**Manažér** má väčšinou na starosti evidenciu a vytváranie nových subjektov. V prvom rade eviduje, prepúšťa, pokutuje zamestnancov. Taktiež eviduje zákazníkov, ktorí si objednali inventúru. Ako jediný v spoločnosti komunikuje so servisom vozidiel, externým účtovníkom a s externými poskytovateľmi prepravy osôb. Vytvára, upravuje a ruší inventúry, ktoré boli dohodnuté so zákazníkom a eviduje prihlásených zamestnancov na tieto inventúry. Určuje výšku hodinových miezd a rozosiela výplatné pásky všetkým aktívnym zamestnancom, ktoré vytvoril externý účtovník. Vyhlasuje konkurzy na prijímacie konanie nových zamestnancov, čím vlastne rozhoduje o prijatí nových zamestnancov. Rozhoduje o investíciách spoločnosti, či sa už jedná o investície do zariadení potrebných pre vykonávanie inventúry, alebo investície do vozidiel, a pod.

**Vedúci inventúry** je osoba, ktorá má na starosti priebeh inventúry. Kontroluje, či sa na inventúru dostavili všetci zamestnanci v stanovený čas. Následne kontaktuje manažéra a informuje ho o priebehu inventúry.

**Vodič** sa môže prihlásiť na inventúru ako vodič, ale aj ako inventarizačný pracovník, a to v prípadoch, kedy už majú vozidlá obsadenú pozíciu vodiča.

**Inventarizačný pracovník** je značne najpočetnejším typom zamestnanca v spoločnosti. Ten sa na inventúru prihlasuje iba ako inventarizačný pracovník. Taktiež nahlasuje sťažnosti manažérovi, napríklad nahlásenie neprimerane rýchlej jazdy vodiča.

Všetci zamestnanci musia spĺňať pravidlá pre aktívnych zamestnancov. V prvom rade, pokiaľ zamestnanec dostal pokuty týkajúce sa absencie, a jeho vyplatená mzda je vypočítaná na hodnotu nižšiu ako 0, stáva sa automaticky prepusteným zamestnancom. Druhým pravidlom je, že každý zamestnanec musí splniť stanovený počet inventúr v mesiaci, inak sa stáva automaticky neaktívnym zamestnancom.

Priebeh inventúry začína v prvotnom impulze od zákazníka, ktorý pomocou zverejnených kontaktov na webovej stránke kontaktuje manažéra. Po vzájomnej komunikácii a dohode kedy má byť inventúra vykonaná, manažér informuje zamestnancov o novej inventúre. Tí sa môžu rozhodnúť, či chcú túto inventúru absolvovať. Aby nedochádzalo k nezrovnalostiam, manažér informuje zamestnancov či boli vybraní na danú inventúru. Všetci prihlásení zamestnanci sa dostavia v stanovený čas na miesto, kde sídli spoločnosť a nasleduje presun na inventúru.

Po skončení inventúry vyplní vedúci inventúry report zamestnancov, kde sú hlavne zaznamenané časy, kedy sa reálne dostavili na miesto určenia, kedy začali a skončili zamestnanci pracovať a akú dlhu mali povinnú prestávku (obrázok č. 9). Je to z dôvodov, ako je nedostavenie sa zamestnanca na inventúru, čím sa vlastne inventúra predĺži. Taktiež môže byť dôvodom státie v kolóne pri preprave na dané miesto, čím sa posunie čas začiatku inventúry.

	A	B	C	D	E	F	G
1	ID inventúry:			Mesto:			
2	Zákazník:			Ulica:			
3	Dátum:			Číslo:			
4	Zodpovedný vedúci:			PSČ:			
5	Počet zamestnancov:						
6	Počet vodičov:						
7	Vzdialenosť:						
8	Čas cesty:						
9							
10	Poradové. č.	ID	Príezvisko	Meno	Nástup	Koniec	Prestávka
11							

Obr. 9 Ukážka vzoru reportu zamestnancov

Aktuálne má spoločnosť vo vlastníctve šesť vozidiel, ktoré sa používajú výhradne na prevoz zamestnancov tykajúcich sa inventúry. K týmto vozidlám má prístup iba manažér, vodiči a vedúci inventúr. Podľa pravidiel spoločnosti sa nesmú používať na osobné účely. Keďže vodiči, ktorí sú zamestnaní v spoločnosti, majú vodičský preukaz typu B<sup>9</sup>, sú vozidlá päť, sedem a deväť miestne. Pokiaľ je potrebné na inventúru dopraviť viac ako cca 20 zamestnancov, manažér kontaktuje externú spoločnosť, ktorá poskytuje prenájom autobusovej dopravy. V prípade, kedy vozidlo nie je z nejakého dôvodu spôsobilé na jazdu, vodič nahlási manažérovi, že vozidlo je nepojazdné. Ten kontaktuje servis vozidiel, ktorý vozidlo opraví. Samozrejme v túto dobu nemôže byť vozidlo priradené na žiadnu inventúru.

Na konci mesiaca manažér vytvorí dokument so všetkými inventúrami. Tento dokument obsahuje všetky informácie o odpracovaných hodinách, najazdených kilometroch a dĺžkach prestávok zamestnancov pre jednotlivé inventúry a odošle ho externému účtovníkovi, ktorý vypočíta celkový počet odpracovaných hodín za daný mesiac pre jednotlivého zamestnanca, jeho odvody sociálnej a zdravotnej poisťovni, a následne celkovú vyplatenú mzdu. Z tejto mzdy sa odčítajú pokuty, ktoré dostal daný zamestnanec. Následne vytvorí výplatné pásky, ktoré odošle manažérovi a ten príslušnému zamestnancovi.

Na obrázku č. 10 je zobrazený vzor výplatnej pásky. Tá obsahuje hlavičku, kde sú základné informácie. Následné jednotlivé časti obsahujú podrobné informácie, z ktorých sa v pravej dolnej časti vypočíta celková mzda. V spoločnosti sa mzda vypláca výhradne na bankový účet zamestnanca. V ľavej spodnej časti je odstavec (v ukážke prázdny), v ktorom sa vypisujú jednotlivé inventúry, ktoré zamestnanec absolvoval.

<sup>9</sup> Oprávnenie viesť vozidlo, ktorého maximálna prípustná hmotnosť je 3500 kilogramov a nemá okrem miesta vodiča viac ako osem miest na sedenie (pravda.sk, 2016)

Výplatná páska 20150500080	Názov spoločnosti <b>(00080) Meno Priezvisko</b>		Máj 2015 (24) DÓVERA zdravotná poisťovňa, a.s.				
Odpracované	0,00	Prekážky	0,00	Základná mzda	0,00	Odmena štatutára	0,00
Dovolenka	0,00	Plat.volno	0,00	Mzda za nadčas	0,00	Súbežný pr. pom	0,00
Sviatky	0,00	Neplat.volno	0,00	Nadčasový prípl.	0,00	Práca v zahraničí	0,00
Nadčas	0,00	Náhr.volno	0,00	Ostatné príplatky	0,00	Zo soc. fondu	0,00
Nočné	0,00	Absencia	0,00	Náhrady	0,00	Nat príjem zo SF	0,00
Poh.na prac.	0,00	Vyňatie	0,00	Prémie a odmeny	0,00	Naturálie	0,00
Pod.doma	0,00	Iné	0,00	Jednoráz. odmeny	0,00	Fiktívna mzda	0,00
Nemoc	0,00			Ostatné odmeny	0,00	Storno	0,00
				Hrubá mzda	0 €		
				Starobné poistenie	0 €		
				Invalidné poistenie	0 €		
				Sociálne poistenie spolu	0 €		
				Zdravotné poistenie	0 €		
				Iné príjmy	0 €		
				Daň zo mzdy	0 €		
				Čistá mzda	0 €		
				K výplate v hotovosti	0 €		
				K výplate bankovým prevodom na účet <b>101242424/0300</b>	<b>0 €</b>		

Obr. 10 Ukážka výplatnej pásky pri nulovej mzde

### 3.3 Špecifikácia požiadaviek

Pri návrhu diagramu užitia je potrebné rozhodnúť aké funkcie, prípady užitia, má informačný systém vo forme webovej aplikácie poskytovať. Po vzájomnej konzultácii so spoločnosťou a z výsledkov analýzy z podkapitoly 5.1, bolo možné vytvoriť špecifikáciu požiadaviek. Tieto požiadavky boli následne prevedené do diagramu prípadov užitia, ktorý je predmetom prílohy C.

Hlavnou úlohou systému je organizácia inventúr, prihlasovanie a odhlasovanie na inventúry zo strany inventarizačných pracovníkov, vodičov a vedúcich inventúr. Taktiež má poskytovať prácu s inventúrami, čo znamená vytváranie inventúr, ich úpravu a zrušenie zo strany manažéra. Pri úprave a zrušení inventúry je tiež potrebné informovať prihlásených zamestnancov o týchto zmenách, respektíve o oficiálnom zrušení danej inventúry. S inventúrami je tiež nutné evidovať rezervácie vozidiel, prihlasovanie a odhlasovanie zamestnancov na inventúry. Zo strany vedúceho inventúry je za potreby ukladať reporty zamestnancov k danej inventúre.

Ďalšími požiadavkami na webovú aplikáciu je zobrazenie výplatných pásoch jednotlivému zamestnancovi, ktorých obsah sa vyplní automaticky systémom. Je to z dôvodu, aby ich manažér nemusel odosielať každému aktívnemu zamestnancovi zvlášť. Taktiež je v požiadavkách spoločnosti vytvoriť systém štatistík, ktoré by pomohli pri budúcom rozhodovaní. Napríklad štatistiky ohľadom najaktívnejších typoch vozidiel, z ktorých bude zrejmé, že sa oplatí investovať do typu vozidla, ktoré bolo v danom roku najpoužívanejšie. Ale aj štatistiky o aktivite zamestnancov, ktoré môžu pomôcť pri rozhodovaní o povýšení inventarizačných pracovníkov na vedúcich inventúr.

## Funkčné požiadavky

- Užívateľ systému sa vždy musí prihlásiť a odhlásiť zo systému.
- Užívateľ systému sa prihlasuje s užívateľským menom, ktorý je totožný s jeho e-mailovou adresou.
- Užívateľ systému si môže zmeniť svoje prihlasovacie heslo.
- Užívateľ systému môže požiadať o vygenerovanie nového hesla.
- Užívateľ systému si môže zobrazit' svoje osobné informácie.
- Užívateľ systému si spravuje svoje osobné informácie individuálne.
- Zamestnanec si môže zobrazit' ohlásené inventúry a inventúry na ktorých je prihlásený.
- Zamestnanec<sup>10</sup> si môže zobrazit' informácie o inventúre.
- Zamestnanec si môže zobrazit' archív inventúr, ktoré absolvoval.
- Zamestnanec si môže zobrazit' svoje výplatné pásky.
- Zamestnanec môže vytvorit' hlásenie pre manažéra.
- Zamestnanec si môže zobrazit' svoje osobné štatistiky.
- Zamestnanec sa môže odhlásiť z inventúry.
- Inventarizačný pracovník sa môže prihlásiť na inventúru v určenej lehote iba ako inventarizačný pracovník.
- Vodič sa môže prihlásiť na inventúru ako inventarizačný pracovník alebo ako vodič.
- Vodič si môže zobrazit' informácie o vozidlách.
- Vedúci inventúry sa môže prihlásiť na inventúru iba ako vedúci inventúry.
- Vedúci inventúry si môže zobrazit' zoznam zamestnancov.
- Vedúci inventúry si môže zobrazit' detailne informácie zamestnanca.
- Vedúci inventúry musí vložit' report po skončení inventúry.
- Manažér môže vytvorit' inventúru pre nového zákazníka.
- Manažér môže vytvorit' inventúru pre evidovaného zákazníka.
- Manažér si môže zobrazit' aktívne, neaktívne a ukončené inventúry.
- Manažér môže zrušiť inventúru.
- Manažér môže upraviť inventúru.
- Systém musí informovať zamestnancov o zmenách alebo zrušení inventúry.
- Manažér môže vytvorit' nového zamestnanca.
- Manažér môže upraviť informácie o zamestnancovi.
- Manažér môže zmeniť aktivitu zamestnanca.
- Manažér môže prepustiť zamestnanca.
- Manažér si môže zobrazit' všetkých zamestnancov.
- Manažér si môže zobrazit' detail zamestnanca a jeho osobné štatistiky.
- Manažér môže pokutovať zamestnanca.
- Manažér môže vytvorit' alebo upraviť zákazníka.
- Manažér si môže zobrazit' podrobné informácie o zákazníkovi.

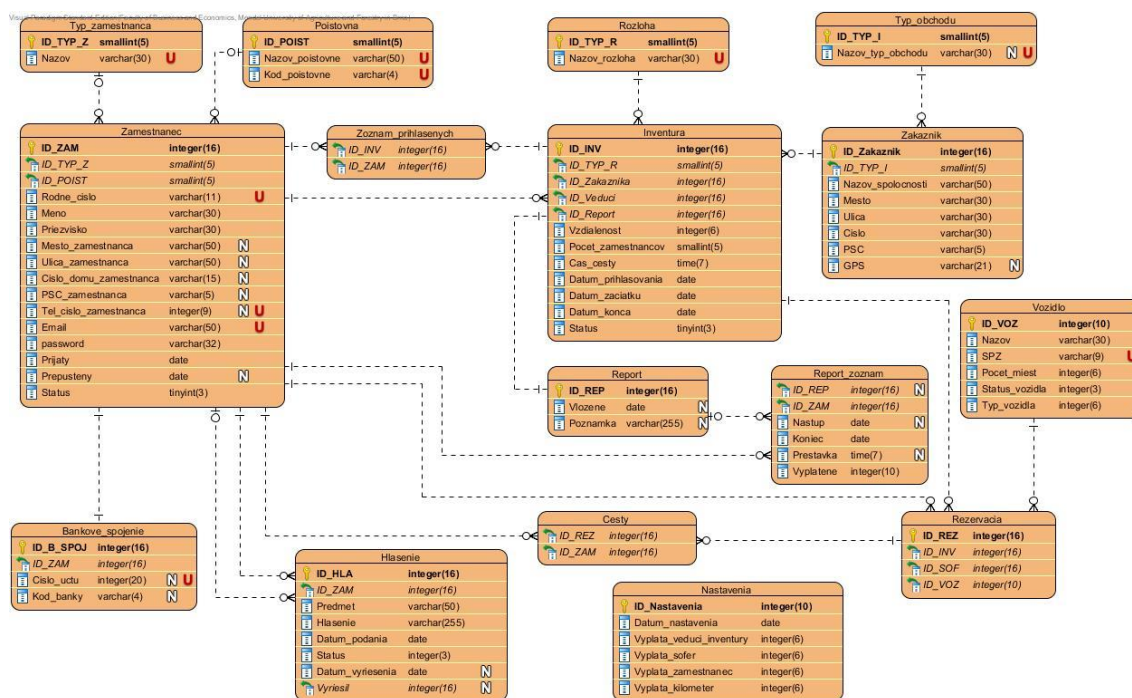
---

<sup>10</sup> Generalizácia - inventarizačný pracovník, vodič, vedúci inventúry

- Manažér si môže zobrazit' rezervácie vozidiel.
- Manažér si môže zobrazit' vozidlá.
- Manažér môže spravovať vozidlá.
- Manažér si môže zobrazit' celkové štatistiky, štatistiky zamestnancov a štatistiky vozidiel.
- Manažér si môže zobrazit' a vyriešiť hlásenia od zamestnancov.
- Manažér môže nastaviť systém.
- Systém musí kontrolovať aktivity zamestnancov.
- Inventúra musí trvať minimálne 3 hodiny.

### 3.4 Návrh entitne-relačného diagramu

Po vytvorení diagramu prípadov užitia bolo jednoznačne, aké dáta je nutné v databáze uchovávať. Na toto znázornenie bol použitý entitne-relačný diagram, ktorý sa skladá zo 16-tich entít:



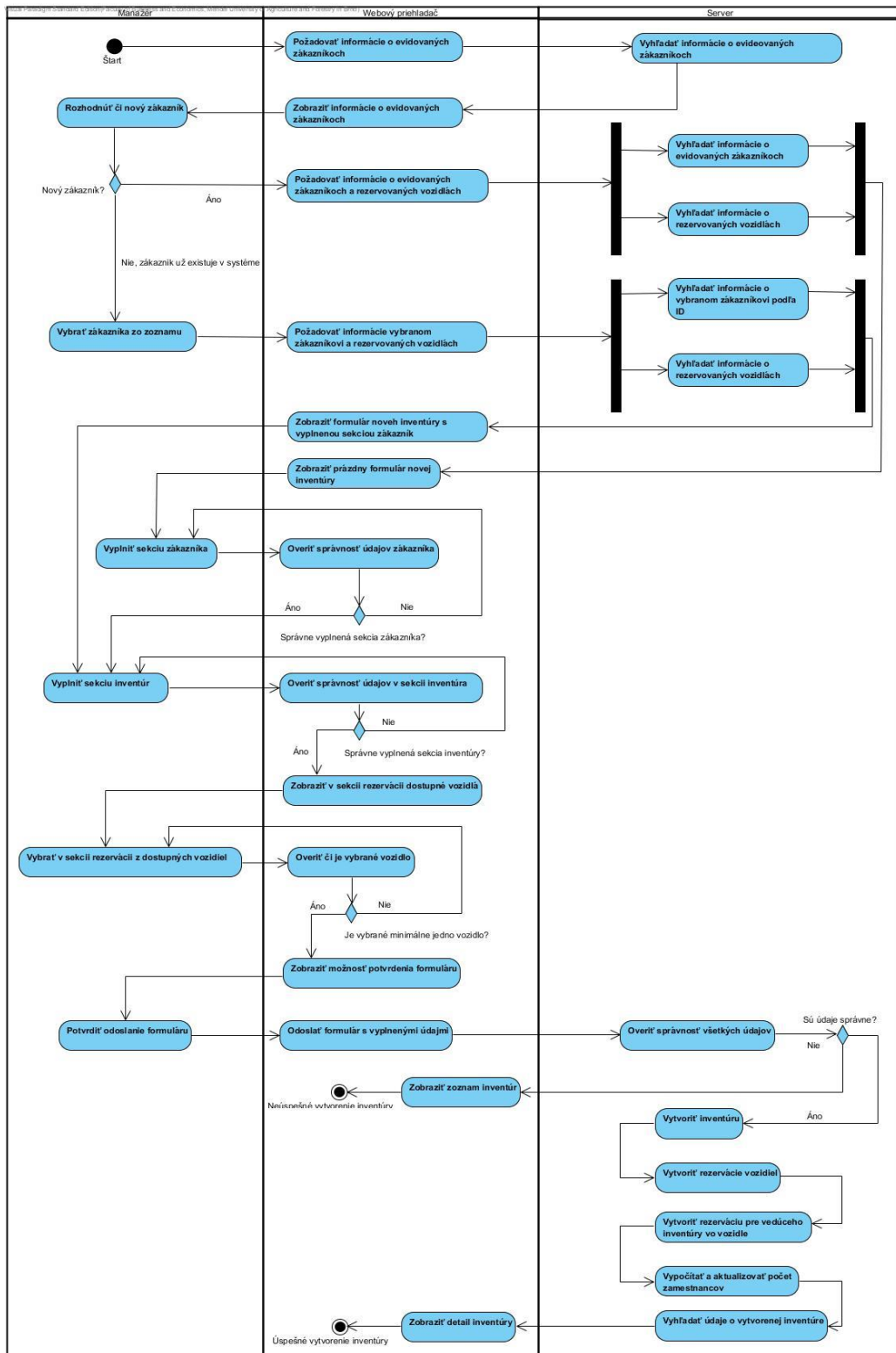
Obr. 11 Entitne relačný diagram

Medzi základné entity patria Zamestnanec, Inventúra, Zákazník, Vozidlo, Hlásenie, Report a Report zoznam a sú pripojené k jednotlivým pomocným entitami príslušnými väzbami. Zamestnanec má vzťah s entitami Typ zamestnanca, Poistovňa a Bankové spojenie. Inventúra má pomocnú entitu Rozloha a Zákazník má Typ obchodu. Tieto pomocné entity boli vytvorené za účelom zjednodušenia základných entít, alebo z dôvodu zabránenia zbytočným duplicitám v uložených dátach. Diagram obsahuje tiež dve spojovacie entity a to Zoznam prihlásených a Cesty, ktoré

reprezentujú vzťah N:M medzi danými dvomi entitami. V žiadnej relácii nie je entita s názvom Nastavenia, pretože obsahuje hodinové mzdy jednotlivých zamestnancov. Je to z toho dôvodu aby bolo možné systém prispôsobiť reálnej inflácii, ktorá zahŕňa aj zvyšovanie miezd.

### 3.5 Návrh diagramov aktivít

Po vytvorení diagramov užitia, ktorý reprezentuje aké funkcie má systém poskytovať a po vytvorení entitne-relačného diagramu, ktorý reprezentuje aké dáta majú byť uložené v databáze, bolo možné vytvoriť diagram aktivít pre jednotlivé prípady užitia. Všetky diagramy aktivít, pre jednotlivé prípady užitia, sú obsahom priloženého DVD (bližšie v prílohe A) a v nasledujúcom texte je vysvetlený a stručne popísaný diagram aktivít pre vytvorenie inventúry manažérom, ktorý je zobrazený na obrázku č. 12. Ten začína v počiatočnom uzle s názvom Štart. Následne sa vytvorí požiadavka pre načítanie informácií o zákazníkoch a ich zobrazení v zozname vo webovom prehliadači. Následne má manažér možnosť si vybrať, či vytváraná inventúra je pre nového zákazníka alebo pre zákazníka, ktorý už je evidovaný v systéme. Tu sa aktivita delí na dve línie. Rozdielom je príprava dát pre zobrazenie formuláru. Pokiaľ je inventúra vytváraná pre nového užívateľa, je potrebné vyplniť sekciu zákazníka, pokiaľ pre vybraného existujúceho zákazníka, systém túto sekciu vyplní automaticky sám. Jednotlivé sekcie formuláru sú kontrolované v reálnom čase pomocou webového prehliadača, JavaScriptu. Po správnom vyplnení všetkých sekcií sa zobrazí manažérovi možnosť odoslania formuláru, kde sa následne dáta skontrolujú druhýkrát na strane serveru. Je to z toho dôvodu, aby nevznikli problémy v integrite dát ukladaných do databázy, v prípadoch kedy zlyhalo overenie pomocou JavaScriptu alebo užívateľovi sa podarilo túto kontrolu dát obísť. Po nesprávnej verifikácii dát inventúry sa nevkladajú žiadne dáta, manažérovi sa zobrazí zoznam všetkých inventúr. Po správnej verifikácii sa dáta novej inventúry vytvorí a manažérovi sa zobrazí detail vytvorenej inventúry.



Obr. 12 Diagram aktivity – vytvorenie inventúry

### 3.6 Návrh webového rozhrania

Posledným krokom, pred začatím implementácie, je potrebné rozhodnúť ako budú vyzerat' jednotlivé pohľady užívateľov. Pre demonštráciu bol vybraný pohľad manažera spoločnosti, konkrétne zoznam zákazníkov (obrázok č. 13). Ten obsahuje hlavičku, ktorá zabezpečuje jednotlivé prístupy k aplikácii a pre každý typ užívateľa je rozdielna. Následne telo pohľadu sa vždy mení podľa toho, v ktorej sekcii sa aktuálne užívateľ nachádza. Pokiaľ sa jedná o zoznam, pod telom sa automaticky zobrazí stránkovanie.

Profil	Inventúry	Zamestnanci	Zákazníci	Rezervácie	Vozidlá	Štatistiky	Odhlásiť	✉	⚙	?
Nový zákazník		Hľadať								
ID zákazníka	Názov spoločnosti	Mesto	Ulica	Akcia						
1.	Spoločnosť Prvá	Poprad	Hlavná 1	Upraviť   Detail						
2.	Spoločnosť Druhá	Poprad	Hlavná 2	Upraviť   Detail						
3.	Spoločnosť Tretie	Poprad	Hlavná 3	Upraviť   Detail						
4.	Spoločnosť Štvrtá	Poprad	Hlavná 4	Upraviť   Detail						
5.	Spoločnosť Piata	Poprad	Hlavná 5	Upraviť   Detail						
6.	Spoločnosť Šiesta	Poprad	Hlavná 6	Upraviť   Detail						
7.	Spoločnosť Siedma	Poprad	Hlavná 7	Upraviť   Detail						
8.	Spoločnosť Ôsma	Poprad	Hlavná 8	Upraviť   Detail						
9.	Spoločnosť Deväta	Poprad	Hlavná 9	Upraviť   Detail						
10.	Spoločnosť Desiata	Poprad	Hlavná 10	Upraviť   Detail						
Stránkovanie (pagination)										

Obr. 13 Drôtený model zoznamu zákazníkov

Celá webová aplikácia je na bielom pozadí (#FFFFFF). Avšak pre hlavičku a telo pohľadov bola vybraná sivá farba (#F9F9F9), s jemným vnútorným tieňom. Je to z toho dôvodu, že v tejto kombinácii sa jedná o minimálny kontrast medzi pozadím a komponentami, ktorý je na druhej strane dostačujúci pre viditeľnosť hraníc komponent.

Obr. 14 Ukážka kontrastu medzi pozadím a telom pohľadu

V aplikácii sú tiež použité tlačidlá, ktorých autorom je Nicolas Gallagher a sú voľne dostupné na GitHubu. Boli vybrané, pretože vo vzťahu so sivými telami pohľadov sú výrazne, čím sa dáva najavo užívateľovi jasne a zreteľne, ktoré akcie môže vykonávať.



Obr. 15 Ukážka tlačidiel

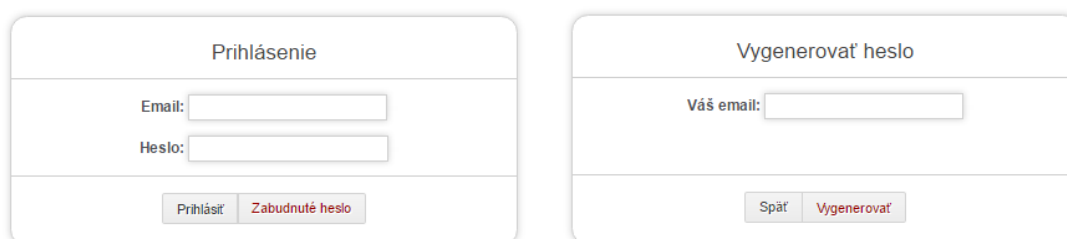


## 4 Implementácia a výsledky

Samotná implementácia navrhnutého informačného systému je vo forme webovej aplikácie. Tá je vytvorená v PHP frameworku Codeigniter, ktorý je výsledkom analýzy z kapitoly 5.2. Aplikácia tiež využíva funkcie JavaScriptu a to hlavne v prípadoch overovania formulárov na strane klienta. Framework bolo potrebné nastaviť a pripojiť k databázovému systému MySQL, ktorý bol primárne na lokálnom serveri XAMPP. Následne bola vyexportovaná databáza a importovaná na server poskytovateľa spolu so samotnou aplikáciou. URL adresa domény a zoznam prihlasovacích údajov sú v prílohe B, kde sa nachádza kópia webovej aplikácie, ktorá je určená na prezentovanie tejto práce.

### 4.1 Prihlasovanie a odhlasovanie užívateľov

Pri zadaní URL adresy do webového prehliadača sa zobrazí užívateľovi prihlasovací formulár, ktorý je znázornený na obrázku č. 16 (vľavo). Pokiaľ užívateľ zabudol heslo, môže sa dostať odkazom „zabudnuté heslo“ na formulár (obrázok č. 16, vpravo), kde po správnom zadaní e-mailu si môže nechať vygenerovať nové 6-miestné prihlasovacie heslo. Toto heslo je automaticky odoslané na zadanú e-mailovú adresu a užívateľ si ho môže po úspešnom prihlásení v systéme zmeniť.



The image shows two side-by-side web forms. The left form, titled "Prihlásenie", contains two input fields: "Email:" and "Heslo:". Below these fields are two buttons: "Prihlásiť" and "Zabudnuté heslo". The right form, titled "Vygenerovať heslo", contains one input field: "Váš email:". Below this field are two buttons: "Späť" and "Vygenerovať".

Obr. 16 Formulár prihlásenia a formulár zabudnutého hesla

Codeigniter poskytuje pre autentifikáciu knižnicu Form validation. Po úspešnej autentifikácii Codeigniter sám vytvorí session (v preklade reláciu), ktorá je nastavená tak, aby uložila údaje o prihlasovanom užívateľovi a to jeho e-mail, status prihlásenia, hodnota privilégii, jeho osobné ID, meno a priezvisko. Tieto údaje sa používajú v rôznych častiach aplikácie a sú potrebné pre jej správne fungovanie. Po odhlásení sa samozrejme táto relácia zmaže.

### 4.2 Privilégia užívateľov

Keďže navrhnutý diagram prípadov užitia obsahuje štyroch aktérov, aplikácia obsahuje štyri hlavné sekcie, ktoré umožňujú rôzne sprístupnenie funkcií aplikácie pre jednotlivých užívateľov. To znamená, že každý typ užívateľa má svoje privilégia.

gia, ktoré sa v mnohých častiach aplikácie kontrolujú pomocou jednoduchej podmienky:

```
if ($this->session->userdata('permission')==1) {
    //... kód ...
} else {
    redirect (base_url('main/restricted'));
}
```

Pokiaľ sa pokúsi spustiť funkciu neautorizovaný užívateľ (užívateľ ktorý má vo svojej relácii uloženú inú hodnotu privilégii), automaticky sa jeho relácia zmaže. Tým sa vlastne odhlási zo systému a je presmerovaný na pohľad, ktorý ho informuje o neautorizovanom vstupe. Samozrejme, nemá zmysel kontrolovať privilégia pre funkcie systému uložené v radiči s názvom Main, keďže túto časť majú sprístupnenú všetci užívatelia.

### 4.3 Zoznamy a vyhľadávanie v nich

Všetky zoznamy v aplikácii obsahujú stránkovanie (pagination), ktoré je dostupné už v samotnom frameworku. Pre znázornenie implementácie bol vybraný zoznam zamestnancov, konkrétne zoznam všetkých zamestnancov a je ho možné vidieť nasledujúcim obrázku.

ID zamest.	Meno a priezvisko	Mesto	Kontakt	E-mail	
00011	Tomáš Gotch	-nezadané-	+421 917841254	gotch.tomas@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00012	Marcel Lipták	-nezadané-	+421 908149658	liptak.marcel@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00013	Dominika Benková	Poprad	+421 917077897	benkova.dominika@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00014	Kamila Frisiková	-nezadané-	+421 903698745	frisikova.kamila@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00017	Alžbeta Bendiková	Poprad	+421 917558958	bendikova.alzbeta@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00018	Alexandra Dobrotová	-nezadané-	+421 904587412	dobrotova.alexandra@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00019	Ján Čongva	Kežmarok	+421 905548251	congva.jan@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00020	Ján Dobrota	-nezadané-	+421 958368254	dobrota.jan@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00021	Luboš Dobrota	-nezadané-	+421 903125789	dobrota.lubos@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>
00022	Filip Smolec	-nezadané-	+421 587458962	smolec.filip@gmail.com	<a href="#">Upraviť</a> <a href="#">Detail</a>

Obr. 17 Zoznam všetkých zamestnancov

Tento zoznam sa nachádza v sekcii zamestnanci a obsahuje hlavičku, kde je možnosť presmerovania na formulár vytvorenia nového užívateľa. Taktiež obsahuje sériu tlačidiel, ktoré slúžia na selektovanie zoznamu podľa typu užívateľa. Násled-

ne je zobrazený zoznam, ktorý obsahuje 10 záznamov, kde každý obsahuje základné informácie a akcie, ktoré môžu byť vykonané s daným záznamom.

Hlavička tabuľky záznamov reprezentuje taktiež automatické vyhľadávacie kritéria. V tomto prípade to znamená, že je možné vyhľadať zamestnancov podľa ich mena, priezviska, identifikačného čísla, kontaktu, emailu ale taktiež podľa mesta. Pre vyhľadanie zadaného reťazca slúži nasledujúci SQL príkaz:

```
SELECT ID_ZAM, Meno, Priezvisko, Mesto_zamestnanca, Tel_cislo_zamestnanca, Email
FROM zamestnanec
WHERE (Meno LIKE "%'. $searchStr.'" OR Priezvisko LIKE "%'. $searchStr.'" OR
ID_ZAM LIKE "%'. $searchStr.'" OR Mesto_zamestnanca LIKE "%'. $searchStr.'" OR
Ulica_zamestnanca LIKE "%'. $searchStr.'" OR Email LIKE "%'. $searchStr.'");
```

## 4.4 Vytváranie inventúr

Ako už bolo znázornené na diagrame aktivity vytvárania inventúry (kapitola 3.5), je možné vytvoriť inventúru pre nového alebo už pre evidovaného zákazníka. Na obrázku č. 18 je zobrazený formulár pre nového zákazníka, ktorý sa skladá z troch sekcií: sekcia zákazník, sekcia inventúra a sekcia rezervácie (vozidlá).

Obr. 18 Formulár novej inventúry pre nového zákazníka

Vo všetkých formulároch je overenie pomocou JavaScriptu. V tomto prípade to znamená, že sa kontroluje napríklad PSČ zákazníka, ktoré sa musí skladať iba z piatich čísiel. Taktiež sa v tomto formulári overuje či jednotlivé časy inventúry sú v časovom poradí za sebou. Po správnom vyplnení sekcie inventúry sa zobrazí sekcia rezervácie, ktorá obsahuje dostupné vozidlá na zvolený časový úsek trvania inventúry. Samozrejme sa povolí odoslanie formuláru vytvorenia novej inventúry iba v prípade, kedy sú správne vyplnené všetky tri sekcie.

## 4.5 Detail zákazníka

Pre každého vytvoreného zákazníka je možné zobrazit' jeho podrobný detail, ktorý sa skladá z dvoch častí. Prvou sú všeobecné informácie, ktoré obsahujú základne identifikačné prvky, adresu kde zákazník sídli a taktiež informácie o počet uskutočnených inventúr vybraného zákazníka spolu s dátumom poslednej uskutočnenej inventúry. V tejto časti detailu je taktiež mapa, ktorá slúži pre informatívne účely. Používa knižnice tretej strany, a to Google Maps API. Druhou časťou tohto pohľadu je zoznam inventúr, ktoré boli vykonané pre daného zákazníka.

**Všeobecné informácie** Inventúry zákazníka

Identifikačné číslo: 00053  
 Názov spoločnosti: ThermGas  
 Typ spoločnosti: Iné  
 Kontaktná adresa: Hviezdoslavova 363/38  
 05801, Poprad  
 Počet inventúr: 2  
 Posledná inventúra: 25.03.2016

**Inventúry zákazníka**

ID inventúry	Zákazník	Mesto	Dátum a čas začiatku	Vedúci inventúry
00115	ThermGas	Poprad	25.03.2016 3:00	Janka Bendiková
00111	ThermGas	Poprad	31.01.2016 10:00	Janka Bendiková

Obr. 19 Detail zákazníka

V interakcii s Google Maps API, vzniká jeden problém. Keďže Google je známy tým, že dokáže dohľadať potrebné informácie. Pri zadaní náhodných znakov zvyčajne vráti nulový výsledok, ale v niektorých prípadoch, vráti výsledok GPS súradníc zadanej adresy z úplne iného kontinentu. Toto by bolo možné vyriešiť podmnožinou, ktorá by obsahovala všetky obce a mestá na území Slovenskej republiky. Toto východisko je taktiež predmetom možného rozšírenia aplikácie.

## 4.6 Prihlasovanie na inventúry

Už z nadpisu práce je možné vydedukovať, že aplikácia musí obsahovať aj prihlasovanie na vytvorené inventúry. To je možné uskutočniť v sekcii inventúry a následne vybraním zo zoznamu inventúr, pokiaľ to danému užívateľovi je povolené, respektíve nie je v túto dobu prihlásený na inú inventúru. Prihlasovať sa môžu inventarizační pracovníci a vedúci inventúry. Vodiči majú možnosť sa prihlásiť aj ako inventarizační pracovníci. Rozdielom v týchto prihlasovaniach je iba miesto vloženia identifikačného čísla zamestnanca do databázy, ktorej diagram je možné vidieť na obrázku č. 11.

Pre prihlásenie **inventarizačného pracovníka** je potrebné identifikačné číslo inventúry a identifikačné číslo rezervácie (vozidla), ktorá má ešte voľné miesto. Preto sa identifikačné číslo zamestnanca vkladá do tabuliek s názvom *zoznam\_prihlasenych* a s názvom *cesty*.

```
$sql='INSERT INTO zoznam_prihlasenych (ID_INV,ID_ZAM) VALUES('.$id_inv.','.$id_zam.);'  
$sql='INSERT INTO cesty (ID_REZ,ID_ZAM) VALUES ('.$id_rez.','.$id_zam.);'
```

**Vodič**, ktorý sa prihlasuje na inventúru ako vodič, potrebuje pre prihlásenie iba identifikačné číslo zvolenej inventúry. Atribút *ID\_SOF* záznamu vybranej rezervácie je aktualizovaný na hodnotu reprezentujúcu identifikačné číslo prihlasovaného vodiča.

```
$sql='UPDATE rezervacia SET ID_SOF='.$id_zam.' WHERE rezervacia.ID_REZ='.$id_rez;
```

Prihlásenie **vedúceho inventúry** je obdobné ako u vodiča, s tým rozdielom, že jeho identifikačné číslo sa vkladá do atribútu *ID\_Veduci* v tabuľke *inventura*. Keďže pri vytvorení inventúry sa zároveň vytvorí aj rezervácia miesta vo vozidle (vytvorí sa záznam v tabuľke *cesty* s nulovou hodnotou atribútu *ID\_ZAM*), je potrebné tento atribút záznamu aktualizovať na identifikačné číslo vedúceho inventúry.

```
$sql='UPDATE inventura SET ID_Veduci='.$id_zam.' WHERE inventura.ID_INV='.$id_inv;  
$sql='UPDATE cesty SET ID_ZAM='.$id_zam.' WHERE ID_REZ='.$id_rez.' AND ID_ZAM IS NULL';
```

## 4.7 Vytváranie a vkladanie reportov zamestnancov

Keďže jedným z predmetov analýzy spoločnosti bol aj dokument obsahujúci report zamestnancov, ktorý je možné vidieť na obrázku č. 9. Ten býval vypisovaný vedúcim inventúry a často dochádzalo narušeniu integrity v dátach. Pre vytváranie, generovanie vyplnených reportov je použitá knižnica tretej strany PHP Excel, ktorej autorom je Muhammad Ikhsanudin.

Po vytvorení a nastavení objektu *\$ExcelFile*, sa vkladajú potrebné údaje do jednotlivých buniek a to pomocou príkazu:

```
$ExcelFile->getActiveSheet()->setCellValue('A1', 'ID inventory:');  
$ExcelFile->getActiveSheet()->setCellValue('C1', $inventoryInfo->ID_INV);
```

Na záver sa objekt *\$ExcelFile* prevedie do fyzického súboru typu .xls, určia sa potrebné parametre pre nastavenie súboru a následne začína sťahovanie dokumentu.

Vedúci inventúry môže informácie z tohto dokumentu vložiť automaticky načítaním dokumentu do formuláru, alebo ručne vyplniť jednotlivé polia formuláru. Po vybraní súboru sa zavolá funkcia s názvom *isCorrectFile(a,b)*, kde parametrami funkcie sú identifikačné číslo inventúry a názov spoločnosti zákazníka. Tieto parametre slúžia na kontrolu názvu vybraného súboru, či sa jedná o súbor vygenerovaný aplikáciou. Následne po vybraní a odoslaní súboru sa zavolá funkcia s názvom *createNewReportAuto(\$id\_inv)*, kde parametrom je identifikačné číslo inventúry. Vybraný súbor sa otvorí a skontroluje sa identifikačné číslo zadané v bunke C1, ktorá má vytvorenú ochranu proti editácii. Na záver sa načítajú potrebné údaje z tabuľky do premennej *\$data*, z ktorej sa po presmerovaní vyplní formulár vloženia reportu zamestnancov.

Kamila Frisková
Inventúry
Zamestnanci
Výplatné pásky
Štatistiky
Odhlásiť
✉ ?

### Editácia reportu

Identifikačné číslo inventúry: 00170

Zákazník: Hypernova Garbiarska

Adresa: Garbiarska , Liptovský Mikuláš

Vedúci inventúry: Kamila Frisková

Dátum a čas príchodu: 01.05.2016 01 : 55

Dátum a čas odchodu: 01.05.2016 12 : 05

Por.č.	ID zam.	Priezvisko	Meno	Nástup			Koniec			Prestávka			
S	00073	Lenčes	Matej	01.05.2016	01	55	01.05.2016	10	00				✓
S	00046	Koblížna	Veronika	01.05.2016	02	00	01.05.2016	10	00				✓
1	00072	Adásek	Matej	01.05.2016	01	55	01.05.2016	00	00	00	30		✗ Odišiel pred tým ako prísieť?
2	00053	Baláž	Tomáš	01.05.2016	01	55	01.05.2016	11	55	00	35		✓
3	00067	Bendíková	Monika	01.05.2016	02	00	01.05.2016	12	00	00	30		✓
4	00062	Dobrota	Luboš	01.05.2016	02	00	01.05.2016	12	00	00	30		✓
5	00065	Hudáková	Lenka	01.05.2016	02	00	01.05.2016	11	55	00	40		✓
6	00049	Kecskéš	Matej	01.05.2016	01	55	01.05.2016	12	00	00	35		✓
7	00074	Lenčes	Michal	01.05.2016	01	55	01.05.2016	12	05	00	40		✓
8	00070	Mrkvička	Jožko	01.05.2016	01	55	01.05.2016	12	05	00	40		✓
9	00071	Petrovič	Peter	01.05.2016	02	00	01.05.2016	12	00	00	30		✓
10	00064	Ružisková	Alena	01.05.2016	02	00	01.05.2016	11	55	00	30		✓
11	00068	Sloboda	Ján	01.05.2016	01	55	01.05.2016	01	55	00	00		⚠ Absencia

Na inventúru sa nedostavil zamestnanec Ján Sloboda (00068).

Obr. 20 Formulár vloženia reportu zamestnancov

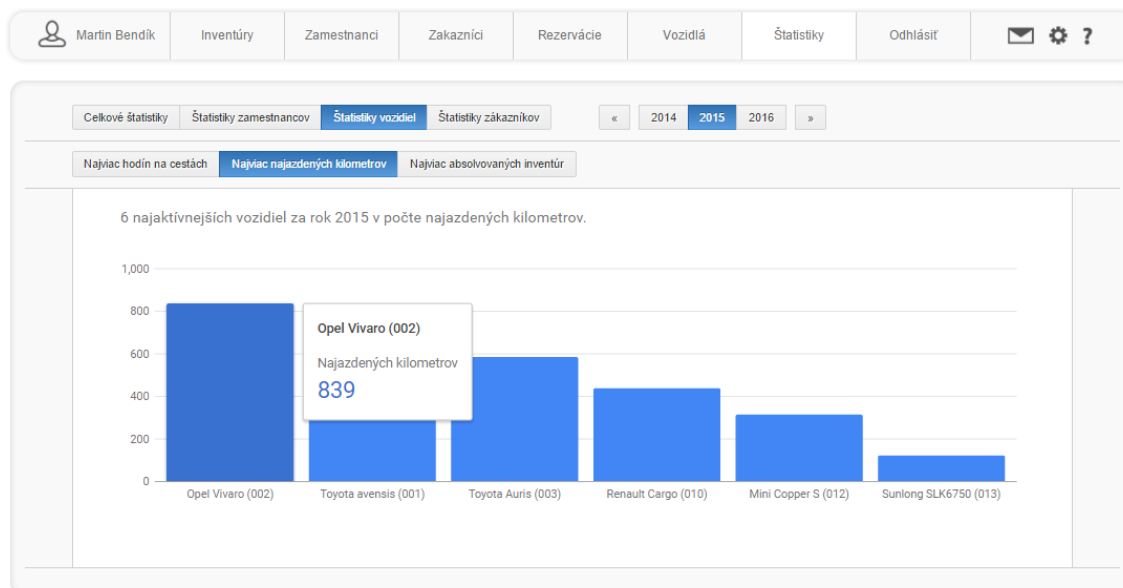
V prípade, kedy vedúci inventúry odošle report a zamestnanec sa nedostavil na inventúru, je automaticky vytvorené hlásenie pre manažéra, ktoré ho informuje o jeho absencii. Ten môže kontaktovať pracovníka a požiadať ho o vysvetlenie alebo ho rovno pokutovať v sekcii zamestnancov, konkrétne v detaile daného zamestnanca. Pokutovať zamestnanca je samozrejme možné iba do konca aktuálneho mesiaca, a to z dôvodu, aby údaje vo výplatných páskach boli totožné s reálnymi vyplatenými mzdami zamestnancom, ktoré sa uzatvárajú posledný deň v mesiaci.

ID inventúry:	Status:	Zakazník:	Absencia dňa:	
00170	Bez pokuty	Hypernova Garbiarska-Liptovský Mikuláš	01.05.2016	<input type="button" value="Pokutovať"/>
00167	Pokutovaný: 0 €	Kenvelo-Košice	07.04.2016	<small>Vyplatené pásky boli vytvorené, už nie je možné pokutovať zamestnanca.</small>
00157	Pokutovaný: -20 €	Adidas-Košice	18.01.2016	

Obr. 21 Ukážka zoznamu pokút pre vybraného zamestnanca

## 4.8 Štatistiky a grafy

Táto časť aplikácie používa knižnice tretej strany a to Google Charts API. Tá ponúka rôzne druhy grafov, z ktorých bol vybraný stĺpcový graf, ktorý sa dá považovať za jeden z najpriehľadnejších grafov. Pre znázornenie výsledku a implementácie je vybraný graf reprezentujúci najaktívnejšie vozidlá v počte najazdených kilometrov, ktorý je možné vidieť v nasledujúcom obrázku.



Obr. 22 Grafické znázornenie aktivity vozidiel podľa počtu najazdených kilometrov

Pre zobrazenie tohoto typu grafu slúži funkcia s názvom *drawTopVehiclesManager(\$modeDet, \$year, \$limit)*, kde parametrami funkcie sú typ štatistík vozidiel, rok zobrazenia a počet zobrazených vozidiel. V tomto prípade ide o druhý typ štatistík vozidiel za rok 2015. Na začiatku tejto funkcie sa získajú potrebné dáta pre prvých šesť vozidiel. Keďže sa jedná o dynamicky vyplňované grafy, je potrebné získané dáta z databázy upraviť na akceptovateľný formát a to pomocou jednodu-

chého cyklu *foreach*. Následne sa nastavujú názvy pre jednotlivé stĺpce grafu a názov celého grafu.

## 4.9 Kontrola aktivity zamestnancov

Manažér strácal množstvo času len zisťovaním, ktorí zamestnanci absolvovali minimálny počet inventúr za mesiac. Preto bolo priam nutné vymyslieť kontrolu pre aktivitu zamestnancov. Implementácia bola vytvorená na strane serveru, v MySQL databáze, a to pomocou udalosti (event) a procedúry (procedure). Hlavička udalosti je vygenerovaná a nastavená tak, aby udalosť začala 1.1.2017 a opakovala sa každý mesiac.

Telo udalosti obsahuje na začiatku potrebné deklarácie premenných, kde najdôležitejšou časťou je kurzor, ktorý spolu s podmienkou typu *repeat* slúži na prechod všetkými vybranými záznamami. Pre každý vybraný záznam sa zavolá procedúra s názvom *skontrolujAktivitu(id\_zam)*, kde parametrom je identifikačné číslo zamestnanca, na ktorý ukazuje kurzor. V prípade že sa v tejto procedúre zistí, že zamestnanec nemá dostatočný počet inventúr, automaticky sa zmení jeho aktivita na hodnotu 0.

```
CREATE DEFINER=`root`@`localhost` EVENT `skontroluj_aktivitu` ON SCHEDULE EVERY 1 MONTH
STARTS '2017-01-01 00:00:00' ON COMPLETION NOT PRESERVE ENABLE DO
```

```
BEGIN
```

```
DECLARE id_zam INT DEFAULT 0;
```

```
DECLARE koniec INT DEFAULT 0;
```

```
DECLARE kurzor CURSOR FOR SELECT zamestnanec.ID_ZAM FROM zamestnanec WHERE zamestnanec.ID_TYP_Z>1 AND zamestnanec.Status=1;
```

```
DECLARE CONTINUE HANDLER FOR NOT FOUND SET koniec = 1;
```

```
OPEN kurzor;
```

```
REPEAT
```

```
    FETCH kurzor INTO id_zam;
```

```
    IF NOT koniec THEN
```

```
        CALL skontrolujAktivitu(id_zam);
```

```
    END IF;
```

```
UNTIL koniec END REPEAT;
```

```
CLOSE kurzor;
```

```
END
```



## 5 Testovanie a zavedenie

### 5.1 Testovanie

Zrejme najdôležitejšou súčasťou nie je návrh a implementácia, ale testovanie výsledného riešenia. V tejto časti vývoja informačných systémov je možné odhaliť nedostatky súvisiace či už s integritou dát, ale aj v samotných funkciách systému a v dobe vykonania týchto funkcií. Webová aplikácia bola testovaná dvomi spôsobmi.

Prvý z nich bol pomocou open-source aplikácie Apache JMeter, kde boli vytvorené HTTP požiadavky na server, ktoré je možné vidieť v tabuľke č. 1. Tento test sa skladá z požiadaviek typu GET, kde sa hlavne vyhľadávajú dáta v databáze a následne odosielajú do webového prehliadaču užívateľa.

Tab. 1 Výsledok testu aplikácie pre požiadavky typu GET

Požiadavka na:	A [pož.]	B [ms]	C [%]	D [pož./s]	E [Kb/s]	F [B]
Aktívne inventúry	1000	229	0.00	3.148	21.28	6921
Archív inventúr	1000	300	0.00	3.148	23.86	7516
Neaktívne inventúry	1000	205	0.00	3.153	12.27	3985
Zoznam zamestn.	1000	232	0.00	3.152	32.45	10542
Zoznam manažérov	1000	242	0.00	3.154	16.45	5341
Zoznam vedúcich	1000	238	0.00	3.155	19.58	6353
Zoznam pracovníkov	1000	238	0.00	3.154	32.49	10550
Zoznam vodičov	1000	234	0.00	3.162	22.91	7419
Zoznam zákazníkov	1000	228	0.00	3.163	29.30	9396
Rezervácie	1000	342	0.00	3.161	33.49	10848
Zoznam vozidiel	1000	284	0.00	3.163	52.85	12646
Štatistiky	1000	235	0.00	3.167	19.71	6372
<b>TOTAL</b>	<b>12000</b>	<b>251</b>	<b>0.00</b>	<b>37.526</b>	<b>343.129</b>	<b>9363</b>

Legenda: Stĺpec A: počet požiadaviek  
 Stĺpec B: priemerná doba požiadavky v milisekundách  
 Stĺpec C: percentuálne vyjadrenie nesplnených požiadaviek  
 Stĺpec D: počet požiadaviek za jednu sekundu  
 Stĺpec E: rýchlosť prenášaných dát v kilobajtoch za sekundu  
 Stĺpec F: priemerné celkové prenesené dáta v bajtoch

Hlavnou úlohou tohto testu bolo zistenie, či sú jednotlivé požiadavky splnené do nastaveného času 1000 ms. Pre presnosť výsledkov bolo v krátkom časovom úseku odoslaných pre každú akciu 1000 požiadaviek v náhodnom poradí, spolu 12000. Stĺpec D reprezentuje percentuálne nesplnenie stanoveného času pri odoslaní, čo

znamená že všetky požiadavky majú hodnotu 0,00 a boli splnené v stanovenej dobe. Stĺpec B reprezentuje priemernú dobu na splnenie požiadavky, kde priemerná doba je 251 ms. Druhou dôležitou časťou bolo zistenie počtu prenesených dát, ktoré sú znázornené v stĺpci G. Najväčšiu hodnotu má požiadavka s názvom Zoznam vozidiel. Je to z toho dôvodu, že sa jedna o zoznam všetkých vozidiel a je potrebné zistiť potrebné informácie pre každé vozidlo. Na záver je možné skonštatovať, že testy dopadli dobre. To znamená, že server požiadavky splňuje relatívne rýchlo a je táto aplikácia vhodná aj pre mobilné pripojenia, kde priemerná veľkosť načítanej stránky je približne 9,4 Kb. Taktiež vie server odpovedať na priemerne 37 rôznych požiadaviek za sekundu (stĺpec D), čo je na rozsah aplikácie úplne dostačujúce.

Druhé testovanie aplikácie bolo z pohľadu užívateľa, kde bolo vybraných niekoľko osôb, ktoré neboli zamestnancami spoločnosti. A to z toho dôvodu, že keby testoval systém zamestnanec spoločnosti, vedel by o všetkých pravidlách spoločnosti a zadával by údaje relatívne správne. Tým by sa odhalil minimálny počet chýb systému. Na druhej strane, keďže testovali systém nezainteresované osoby, zadávali do formulárov množstvo nesprávnych údajov, čím sa zvýšila pravdepodobnosť nájdenia chyby v systéme.

V aplikácii bolo zistených niekoľko chýb, z ktorých najzávažnejšou je nespĺnenie podnikového pravidla a je znázornené na obrázku č. 23. V tomto prípade pri vkladaní reportu zamestnancov bolo možné vložiť hodnotu odpracovanej doby nižšiu ako 3 hodiny, pričom podnikové pravidlo hovorí jednoznačne a to, že minimálna doba inventúry je 3 hodiny, aj v prípade, že reálne trvala menej. Následne boli všetky zistené nedostatky aplikácie odstránené a máme ešte dostatok času na testy, ktoré už budú vykonávané zamestnancami spoločnosti a budú simulovať chod systému, kde sa už finálne zistí spoľahlivosť aplikácie.

Por. č.	ID zam.	Priezvisko	Meno	Nástup	Koniec	Prestávka
S	00038	Bendik	Peter	20.04.2016 05 : 05	20.04.2016 06 : 05	🟢
S	00035	Lenčeš	Michal	20.04.2016 05 : 05	20.04.2016 12 : 05	🟢
S	00034	Lenčeš	Matej	20.04.2016 05 : 05	20.04.2016 12 : 05	🟢
1	00033	Adásek	Matej	20.04.2016 05 : 10	20.04.2016 15 : 45	🟢
2	00017	Bendiková	Alžbeta	20.04.2016 00 : 00	20.04.2016 00 : 00	🟡

Obr. 23 Nesplnenie podnikového pravidla spoločnosti

## 5.2 Zavedenie

Posledným, ale taktiež nevyhnutným krokom je zavedenie nového systému do praxe. Keďže spoločnosť aktuálne vykonáva inventúry a bolo by rizikové zavádzať novinku v podobe informačného systému, bolo určené predpokladané zavedenie v prvom kvartáli nadchádzajúceho roku. Toto obdobie bolo vybrané na základe predlžovania pracovných pomerov medzi zamestnancami a spoločnosťou. Tí musia

v tomto období prísť do spoločnosti a podpísať potrebné dokumenty, čím sa vlastne nevyradia z kolektívu zamestnancov. Pri tomto podpisovaní budú oboznámení o zavedení informačného systému a taktiež poskytnú najaktuálnejšie informácie potrebné pre ich registráciu do systému. Obdobne sa budú evidovať zákazníci, ktorí budú ukladaní do systému, v prípade dohodnutia vykonania inventúry. Výhodou je taktiež redukovanie zákazníkov, ktorí už nemajú aktuálny záujem o vykonávanie inventúr. V tomto období bude systém pod dohľadom, a to z toho dôvodu, aby bolo možné v prípadných problémoch včas eliminovať chyby tak, aby nevznikli straty spoločnosti.



## 6 Diskusia

### 6.1 Ekonomické zhodnotenie a prínosy

Stávajúce riešenie informačného systému, na rozdiel od aktuálnej situácie v spoločnosti, dokáže ušetriť mnoho času všetkým zamestnancom. To znamená, že sa podarilo vytvoriť informačný systém v podobe webovej aplikácie, ktorý dokáže uchovávať potrebné dáta na jednom mieste, centralizovať ich a rýchlo v nich vyhľadávať. Táto webová aplikácia má overovanie integrity dát aj na strane klienta, aj na strane serveru, čím sa maximálne eliminujú chyby, ktoré vznikali jednoduchým vyplňovaním dokumentov tabuľkového programu Excel.

Značným prínosom pre spoločnosť je tiež organizácia práce a kontrola aktivity inventarizačných pracovníkov, vodičov a vedúcich inventúr. To znamená, že systém automaticky kontroluje prihlasovanie a odhlasovanie na inventúry, čím sa značné bremeno prenáša z manažéra na systém. Taktiež už nebude musieť manažér na konci mesiaca pre aktívnych zamestnancov overovať, či splnili požadovaný počet inventúr za mesiac.

Keďže informácie o plánovaných inventúrach sú rozhádzané v dokumentoch v jednotlivých priečinkoch, trvá relatívne dlho kým manažér zistí dostupnosť vozidiel na konkrétny dátum. Preto systém poskytuje sekciu rezervácie, kde si môže manažér značne rýchlejšie vyhľadať dostupnosť vozidiel na konkrétny dátum. Taktiež systém poskytuje pre zamestnancov zobrazenie výplatných pásov, čím sa značne šetrí čas manažéra a externého účtovníka, ktorí museli tieto výplatné pásy odosielať zamestnancom jednotlivo. Systém taktiež poskytuje možnosť zobrazenia štatistík v podobe grafov. Tie slúžia na informatívne účely a v prípade manažéra môžu ovplyvniť budúci vývoj spoločnosti, či už v prijatí nových zamestnancov, alebo rôznych investíciách do zariadení a pod.

Predpokladané náklady spoločnosti pre výsledné riešenie systému v podobe webovej aplikácie sú znázornené v tabuľke č. 2. Keďže je tento systém tvorený ako bakalárska práca za účelom zdokonalenia sa v danej problematike vývoja informačných systémov a webových aplikácii, je cena vývoja daného systému stanovená na 0 €.

Tab. 2 Predpokladané náklady spoločnosti

Typ nákladu	Jednorazové náklady	Náklady ročne
Vývoj informačného systému	0 €	0 €
Registrácia domény pre IS	12,80 €	0 €
Prenájom serveru	0 €	65,52 €

### 6.2 Možné rozšírenia

Aplikácia poskytuje funkcie, ktoré požadovala spoločnosť, ale taktiež každý fungujúci systém je možné zdokonaľiť alebo rozšíriť. Nie len v období kým sa zavedie

informačný systém v podobe webovej aplikácie do spoločnosti ale aj v nasledujúcom období je možné vybrať niektoré rozšírenia a implementovať ich do systému. Tie môžu byť napríklad:

- Evidovanie naskenovaných (spočítaných) produktov pre jednotlivé inventúry.
- Rozšírenie internej komunikácie v systéme pomocou firemných e-mailových schránok.
- Keďže vozidlá spoločnosti je možné sledovať pomocou GPS, tak je možné rozšírenie o sledovanie vozidiel v reálnom čase vo webovej aplikácii.
- Radenie v zoznamoch podľa vybraných kritérií.
- Kontrola existencie zadaného e-mailu.
- Kontrola platnosti zadanej e-mailovej adresy.
- Kontrola existencie mesta vo formulároch.

### 6.3 Iné využitie aplikácie

Aplikácia je primárne vytvorená podľa požiadaviek spoločnosti. Na druhej strane sa podarilo vytvoriť štruktúru informačného systému v podobe webovej aplikácie, ktorý je možné aplikovať aj do iných spoločnosti. Tie musia splňovať základné kritéria flexibilného času zamestnancov, práca musí byť vykonávaná mimo sídla spoločnosti a k tomuto miestu sa musí dopravovať pomocou vozidiel spoločnosti. Samozrejme je za potreby urobiť pár dekoračných zmien, ktoré sa zvyčajne týkajú iba nadpisov jednotlivých pohľadov aplikácie.

Pre príklad bude uvedená zahraničná spoločnosť, ktorá poskytuje služby v podobe predaja, inštalácie a servisu bowlingových dráh. Táto spoločnosť splňuje všetky kritéria, zamestnanci sú vyplácaní od počtu výjazdov, ktoré absolvovali. Taktiež sa jedná o miesta kde spoločnosť nesídlí a používa vlastné prostriedky prepravy. Pracuje tu niekoľko desiatok zamestnancov, manažér im rozvrhuje jednotlivé výjazdy, kde vždy je zodpovedná jedná osoba za daný výjazd. Konkrétne, v tomto využití by boli sprístupnené iba sekcie manažéra, vedúceho a pracovníka. V systéme by sa nastavila hodnota rovnaká hodinová mzda pre vedúceho a pracovníka, čím by sa zabezpečilo rovnaké ohodnotenie pre všetkých prihlásených zamestnancov na daný výjazd.

Výsledkom by bola menšia verzia informačného systému (bez sekcie vodiča), ktorý taktiež poskytuje evidovaných zákazníkov, zamestnancov, detaily, štatistiky a vkladanie reportov, ktoré by reprezentovali časové informácie ohľadom daného výjazdu. Samozrejme sa jedná iba o teoretické východisko, keďže je efektívnejšie prerobiť pohľady a radiče od základov, s využitím už vytvorených funkcií a modelov.

## 7 Záver

Primárnym cieľom tejto práce bolo vytvoriť informačný systém v podobe webovej aplikácie, ktorý má zjednodušovať organizáciu práce v spoločnosti. Súčasťou bolo vytvoriť systém, ktorý dokáže odhaľovať chyby v integrite dát a následne ich správne prezentovať. Sekundárnym, rozširujúcim cieľom bolo vytvoriť štatistiky, ktoré majú pomáhať v budúcom rozvoji spoločnosti. V prvej časti práce sú vysvetlené základné pojmy, ktoré bolo potrebné naštudovať ešte pred samotným návrhom informačného systému.

Následne mohla začať realizačná časť. V nej bolo za potreby najprv analyzovať súčasný stav spoločnosti, z čoho bolo následne možné zistiť požiadavky spoločnosti na informačný systém. Táto časť práce taktiež obsahuje analýzu a porovnanie vybraných nástrojov na podporu tvorby webových aplikácií z pohľadu začínajúceho webového vývojára. Následne boli vytvorené diagramy potrebné pre návrh a modelovanie informačného systému a štruktúry databázy, ktoré sa opierajú o poznatky z teoretickej časti práce.

Práca stručne popisuje implementáciu základných funkcií webovej aplikácie, či už sa jedná o webové rozhranie, ktoré určuje prístupy jednotlivých zamestnancov k aplikácii, ale aj samotnú implementáciu vybraných funkcií, ako sú zobrazovanie zoznamov, vyhľadávanie, detaily subjektov a pod.

V závere práce je znázornený priebeh testovania stávajúceho riešenia aplikácie, ktoré prebiehalo v dvoch krokoch, a to testovanie technických parametrov a následne testovanie z pohľadu užívateľov. Taktiež bol vybraný dátum zavedenia systému do spoločnosti v prvom kvartáli nadchádzajúceho roku. Pred zavedením sú plánované ešte testovania a simulácie, ktoré budú presne znázorňovať chod systému v praxi. V tomto období je možné doplniť systém o možné rozšírenia, ktoré boli uverejnené v kapitole diskusia a neboli predmetom tejto práce.

Môžeme skonštatovať, že výsledkom práce je plnohodnotný informačný systém, ktorý odpovedá požiadavkám spoločnosti.





## 8 Literatúra

- ARORA, SUNIL. *PHP Frameworks: The Best 10 for Modern Web Development* [on-line]. [cit. 2016-04-15]. Dostupné z: <http://noeticforce.com/best-php-frameworks-for-modern-web-development>
- Best PHP Framework for 2016* [online]. [cit. 2016-04-15]. Dostupné z: <http://webrevisions.com/tutorials/php-framework-the-best-php-framework-for-2013/#.VxE-rjCLTIV>
- CARLOS CORONEL, STEVEN MORRIS. *Database systems: design, implementation, and management*. 9. vyd. Australia: Course Technology Cengage Learning, 2011. ISBN 9780538469685.
- FOWLER, MARTIN. *UML distilled: a brief guide to the standard object modeling language*. 3. vyd. Boston: Addison-Wesley, 2004. ISBN 0321193687.
- GITHUB. *About GitHub*. [online]. [cit. 2016-05-07]. Dostupné z: <https://github.com/about>
- GOOGLE. *Společnost – Google*. [online]. [cit. 2016-05-07]. Dostupné z: <http://www.google.cz/about/company/>
- HANWELL, MARCUS. *Should I use a permissive license? Copyleft? Or something in the middle?* [online]. [cit. 2016-04-15]. Dostupné z: <https://opensource.com/business/14/1/what-license-shoul--i-use-open-source-project>
- HOLZNER, STEVEN. *JavaScript profesionálně: [kompletní referenční příručka]*. Praha: Mobil Media, 2003. ISBN 8086593401.
- JAKOUBĚ, JAROSLAV. *ORM test PHP frameworků – Kohana, Laravel* [online]. [cit. 2016-04-15]. Dostupné z: <https://www.zdrojak.cz/clanky/orm-test-php-frameworku-kohana-laravel/>
- JIN, LEE EU. *Bus Tracking and Ticket Payment System for UTAR*. Perak Campus, 2015. Universiti Tunku Abdul Rahman.
- KETTENIS, JAN. *Getting Started With Use Case Modeling* [online]. San Francisco: Oracle corporation, 2007. [cit. 2016-04-15]. Dostupné z: <http://www.oracle.com/technetwork/testcontent/gettingstartedwithusecase-modeling-133857.pdf>
- KITPOVA, IVA. *Most popular PHP Frameworks 2015 [INFOGRAPHIC]* [online]. [cit. 2016-04-15]. Dostupné z: <https://www.webhostface.com/blog/popular-php-frameworks-2015/>
- KOFLER, MICHAEL A BERND ÖGGL. *PHP 5 a MySQL 5: průvodce webového programátora*. Vyd. 1. Brno: Computer Press, 2007. ISBN 9788025118139.
- KOCH, MILOŠ A VIKTOR ONDRÁK. *Informační systémy a technologie*. Vyd. 1. Brno: Akademické nakladatelství CERM, 2004. ISBN 8021427256.

- KOVÁČ, PETER. *Skupiny a podskupiny vodičských oprávnění* [online]. [cit. 2016-04-15]. Dostupné z: <http://auto.pravda.sk/poradna/clanok/76658-skupiny-a-podskupiny-vodicky-opraveni/>
- KROENKE, DAVID A DAVID J. AUER. *Databáze*. 1. vyd. Brno: Computer Press, 2015. ISBN 9788025143520.
- LACKO, LUBOSLAV. *1001 tipů a triků pro SQL*. Vyd. 1. Brno: Computer Press, 2011. ISBN 9788025130100.
- LERDORF, RASMUS, KEVIN TATROE A PETER MACINTYRE. *Programming PHP*. Vyd. 2. Beijing Sebastopol [Calif.]: O'Reilly, 2006. ISBN 0596006810.
- MONUS, ANNA. *10 PHP Frameworks For Developers – Best Of* [online]. [cit. 2016-04-15]. Dostupné z: <http://www.hongkiat.com/blog/best-php-frameworks/>
- PAVUS. *UML: activity diagram - diagram aktivit. OO, UML, analýza, metodologie* [online]. 2005 [cit. 2016-0- -15]. Dostupné z: <http://mpavus.wz.cz/uml/uml-b-activity-3-2-3.php>
- PITT, CHRIS. *Pro PHP MVC. New York: Distributed to the book trade worldwide by Springer Science+Business Media*, 2012. ISBN 9781430241652.
- RÁBOVÁ, IVANA. *Podnikové informační systémy a technologie jejich vývoje*. Vyd. 1. Tribun EU. Brno: Tribun EU, 2008. ISBN 9788073995997.
- SPARX SYSTEMS. *UML 2 Activity Diagram*. [cit. 2016-05-07]. Dostupné z: [http://www.sparxsystems.com.au/resources/uml2\\_tutorial/uml2\\_activitydiagram.html](http://www.sparxsystems.com.au/resources/uml2_tutorial/uml2_activitydiagram.html)
- STEPHENS, RYAN K, RONALD R. PLEW A ARIE JONES. *Naučte se SQL za 28 dní*. Vyd. 1. Překlad Lukáš Krejčí. Brno: Computer Press, 2010. ISBN 9788025127001.
- ŠIMONOVÁ, STANISLAVA, RENÁTA MYŠKOVÁ A PAVEL JIRAVA. *Projektování informačních systémů - UML, procesní řízení [CD-ROM]*. 2. vyd. Pardubice: Univerzita Pardubice, Ekonomicko-správní fakulta, 2008. ISBN 9788073951313.
- ŠIMONOVÁ, STANISLAVA A JAN PANUŠ. *Databázové systémy I: pro kombinovanou formu studia*. Vyd. 1. Pardubice: Univerzita Pardubice, 2007. ISBN 9788071949886.
- ŠKUTOVÁ, JOLANA. *Projektování informačních systémů: učební text předmětu Informační systémy [CD-ROM]*. Vyd. 1. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, Fakulta strojní, 2011. ISBN 9788024827667.
- TERRANCE. *An Overview of PHP Framework Guides for Developers* [online]. [cit. 2016-04-15]. Dostupné z: <http://www.onextrapixel.com/2010/05/24/an-overview-of-php-framework-guides-for-developers/>
- TUTORIALSPPOINT. *UML - Use Case Diagrams* [online]. 2016 [cit. 2016-04-15]. Dostupné z: [http://www.tutorialspoint.com/uml/uml\\_use\\_case\\_diagram.htm](http://www.tutorialspoint.com/uml/uml_use_case_diagram.htm)
- ULLMAN, LARRY E. *PHP 6 and MySQL 5 for dynamic Web sites*. Berkeley, Calif.: Peachpit Press, 2008. ISBN 032152599X.
- VASILIEV, YULI. *PHP Oracle Web development: data processing, security, caching, XML, Web services, and AJAX; a practical guide to combining the power, performance, scalability, and reliability of Oracle Database with the ease of use, short deve-*

- 
- lopment time, and high performance of PHP*. Birmingham [u.a.]: Packt Publ: Packt Pub, 2007. ISBN 9781847193636.
- WATT, ADRIENNE A NELSON ENG. *Database design*. Vyd. 2. [online]. 2. 2012 [cit. 2016-04-15]. Dostupné z: <https://opentextbc.ca/dbdesign01/>
- ZAKAS, NICHOLAS C. *Javascript pro webové vývojáře: programujeme profesionálně*. Vyd. 1. Brno: Computer Press, 2009. ISBN 9788025125090.



## 9 Zoznam obrázkov

Obr. 1	Schéma vývoja softwarového produktu, model vodopád Zdroj Rábová, 2008, str. 40	14
Obr. 2	Diagram prípadov užitia (Use-case diagram) Zdroj Fowler, 2004 , str.103	15
Obr. 3	Diagram aktivity, proces zaplatenia autobusového lístku pomocou mobilnej aplikácie Zdroj Jin, 2015, str. 51	16
Obr. 4	Ukážka entitne relačného diagramu s legendou, verzia IE Zdroj Kroenke a Auer, str. 243, str. 259	17
Obr. 5	Architektúra MVC Zdroj Pitt, 2012, str.1	19
Obr. 6	Vývoj popularity PHP frameworkov podľa prieskumu Google Trends Zdroj: Google Trends	21
Obr. 7	Grafické porovnanie vybraných PHP frameworkov Zdroj Upravené z <a href="http://webhostface.com/blog/popular-php-frameworks-2015/">webhostface.com/blog/popular-php-frameworks-2015/</a>	24
Obr. 8	Organizačná štruktúra spoločnosti s externými systémami (vyznačené sivou farbou)	24
Obr. 9	Ukážka vzoru reportu zamestnancov	26
Obr. 10	Ukážka výplatnej pásky pri nulovej mzde	27
Obr. 11	Entitne relačný diagram	29
Obr. 12	Diagram aktivity - vytvorenie inventúry	31
Obr. 13	Drôtený model zoznamu zákazníkov	32
Obr. 14	Ukážka kontrastu medzi pozadím a telom pohľadu	32
Obr. 15	Ukážka tlačidiel	32
Obr. 16	Formulár prihlásenia a formulár zabudnutého hesla	33
Obr. 17	Zoznam všetkých zamestnancov	34
Obr. 18	Formulár novej inventúry pre nového zákazníka	35

---

<b>Obr. 19</b>	<b>Detail zákazníka</b>	<b>36</b>
<b>Obr. 20</b>	<b>Formulár vloženia reportu zamestnancov</b>	<b>38</b>
<b>Obr. 21</b>	<b>Ukážka zoznamu pokút pre vybraného zamestnanca</b>	<b>39</b>
<b>Obr. 22</b>	<b>Grafické znázornenie aktivity vozidiel podľa počtu najazdených kilometrov</b>	<b>39</b>
<b>Obr. 23</b>	<b>Nesplnenie podnikového pravidla spoločnosti</b>	<b>42</b>
<b>Obr. 24</b>	<b>Diagram prípadov užitia</b>	<b>60</b>
<b>Obr. 25</b>	<b>Ukážka rezervácií</b>	<b>61</b>
<b>Obr. 26</b>	<b>Detail vozidla</b>	<b>61</b>
<b>Obr. 27</b>	<b>Profil zamestnanca</b>	<b>62</b>
<b>Obr. 28</b>	<b>Formulár pre editáciu profilu</b>	<b>62</b>
<b>Obr. 29</b>	<b>Detail inventúry - všeobecné informácie</b>	<b>63</b>
<b>Obr. 30</b>	<b>Detail inventúry - zoznam zamestnancov</b>	<b>63</b>
<b>Obr. 31</b>	<b>Zoznam výplatných pásov</b>	<b>64</b>
<b>Obr. 32</b>	<b>Výplatná páska vo formáte PDF</b>	<b>64</b>
<b>Obr. 33</b>	<b>Výber zobrazenia mesačných uzávierok</b>	<b>65</b>
<b>Obr. 34</b>	<b>Detail inventúry vo formáte PDF</b>	<b>65</b>

## **10 Zoznam tabuliek**

<b>Tab. 1</b>	<b>Výsledok testu aplikácie pre požiadavky typu GET</b>	<b>41</b>
<b>Tab. 2</b>	<b>Predpokladané náklady spoločnosti</b>	<b>45</b>





# Prílohy

## A Obsah priloženého DVD

Priložené DVD obsahuje nasledujúce priečinky:

- **Aplikácia** - obsahuje všetky zdrojové kódy aplikácie a taktiež knižnice tretích strán, ktoré boli použité v práci (priečínok application/third\_party).
- **Databáza** - obsahuje vyexportovanú štruktúru MySQL databázy.
- **Diagramy** - obsahuje všetky diagramy, ktoré boli vytvorené pri návrhu informačného systému.

## B URL adresa a prihlasovacie údaje

Aktuálne riešenie aplikácie je optimalizované pre webový prehliadač Google Chrome a je dostupné na adrese:

- <http://martinbendik.studenthosting.sk>

Prihlasovacie údaje:


- Manažér:
  - Login: bendik.martin@gmail.com
  - Heslo: 123456789
- Vedúci inventúry:
  - Login: bendikova.janka@gmail.com
  - Heslo: 123456789
- Vodič:
  - Login: bendik.peter@gmail.com
  - Heslo: 123456789
- Inventarizačný pracovník:
  - Login: bendikova.alzbeta@gmail.com
  - Heslo: 123456789



## D Rezervácie a vozidlá

Zobrazíť dnešný deň								03.01.2016 - 10.01.2016		Prejsť na	Vyberte
	03.01.2016	04.01.2016	05.01.2016	06.01.2016	07.01.2016	08.01.2016	09.01.2016	10.01.2016			
Toyota Avensis Miest: 5		Liptovský Mikuláš				Prešov	Prešov				
Toyota Auris Miest: 5				Prešov							
Toyota Prius Miest: 7		Košice									
Opel Vivaro Miest: 9	Košice										
Opel Vivaro Miest: 9											
Minibus Miest: 25										Košice	

Obr. 25 Ukážka rezervácií

Detail vozidla:					
ID vozidla:	00014				
Názov vozidla:	Toyota Avensis				
ŠPZ vozidla:	PP-843-BJ				
Počet miest:	5				
Typ vozidla:	Sedan				
Status vozidla:	Aktívne				
<input type="button" value="Späť"/>					

Obr. 26 Detail vozidla

## E Profil zamestnanca

Martin Bendík	Inventúry	Zamestnanci	Zakazníci	Rezervácie	Vozidlá	Štatistiky	Odhlásiť	
---------------	-----------	-------------	-----------	------------	---------	------------	----------	--

Upraviť profil Zmeniť heslo

**Základné informácie:**

Identifikačné číslo:	00039
Meno a priezvisko:	Martin Bendík
Rodné číslo:	935896/9334

**Kontaktná adresa:**

Ulica:	Prvá 363/38
PSČ:	05801
Mesto:	Poprad

**Kontaktné údaje:**

Telefónne číslo:	+421 907555666
Email:	bendik.martin@gmail.com

**Poisťovňa:**

Názov poisťovne:	Union zdravotná poisťovňa, a.s.
Kód poisťovne	27

**Bankové spojenie:**

Číslo účtu:	25515222
Kód banky:	0300

Obr. 27 Profil zamestnanca

Martin Bendík	Inventúry	Zamestnanci	Zakazníci	Rezervácie	Vozidlá	Štatistiky	Odhlásiť	
---------------	-----------	-------------	-----------	------------	---------	------------	----------	--

**Manažér**

Identifikačné číslo:	00039
Meno a priezvisko:	Martin Bendík
Rodné číslo:	935896/9334

**Kontaktná adresa:**

Ulica:	<input type="text" value="Prvá Ulica"/>	
Číslo domu:	<input type="text" value="123/45"/>	
PSČ:	<input type="text" value="12345"/>	
Mesto:	<input type="text" value="Poprad"/>	
Telefónne číslo:	<input type="text" value="903123456"/>	

**Poisťovňa a bankové spojenie:**

Názov poisťovne:	<input type="text"/>	
Číslo účtu:	<input type="text" value="25515222"/>	
Kód banky:	<input type="text" value="0300"/>	

Späť

Obr. 28 Formulár pre editáciu profilu

## F Detail inventúry

👤 Martin Bendík
Inventúry
Zamestnanci
Zakazníci
Rezervácie
Vozidlá
Štatistiky
Odhlásiť
📧 ⚙️ ?


Všeobecné informácie
Zoznam zamestnancov
Report

Identifikačné číslo: 00155  
 Názov spoločnosti: ThermGas  
 Typ spoločnosti: Iné  
 Rozloha spoločnosti: Malá predajňa

Kontaktná adresa: Hviezdoslavová 363/38  
 05801, Poprad

Uzávierka prihlasovania: 08.01.2016 0:50  
 Dátum a čas začiatku: 08.01.2016 9:00  
 Dátum a čas konca: 08.01.2016 20:00


Čas cesty: 0:15  
 Vzdialenosť: 10 Km




Obr. 29 Detail inventúry – všeobecné informácie

👤 Janka Bendiková
Inventúry
Zamestnanci
Výplatné pásky
Štatistiky
Odhlásiť
📧 ?

Všeobecné informácie
Zoznam zamestnancov
Report


 Názov vozidla: Toyota Auris  
 ŠPZ vozidla: KE-621-AA  
 Počet miest: 5

Por. č.	Typ	ID	Priezvisko	Meno	Tel. číslo	Email
1.	Šofér	00046	Koblížna	Veronika	0903665254	koblizna.veronika@zoznam.sk
2.	Vedúci inventúry	00045	Bendiková	Janka	0903789635	bendikova.janka@gmail.com
3.	Zamestnanec	00047	Valluš	Matej	0903996336	vallus.matej@gmail.com
4.	Zamestnanec	00058	Bendiková	Alžbeta	0917558958	bendikova.alzbeta@gmail.com
5.	Zamestnanec	00042	Bendík	Matej	-nevyplnené-	bebewar007@azet.sk


 Názov vozidla: Opel Vivaro  
 ŠPZ vozidla: KE-588-AX  
 Počet miest: 9

Por. č.	Typ	ID	Priezvisko	Meno	Tel. číslo	Email
6.	Šofér	00073	Lenčeš	Matej	-nevyplnené-	lences.michal@gmail.com
7.	Zamestnanec	00074	Lenčeš	Michal	-nevyplnené-	lences.matej@gmail.com
8.	Zamestnanec	00071	Petrovič	Peter	-nevyplnené-	petrovic.peter@gmail.com
9.	Zamestnanec	00061	Dobrota	Ján	0958368254	dobrota.jan@gmail.com
10.	Zamestnanec	00052	Balážová	Monika	0903547852	balazova.monika@gmail.com
11.	Zamestnanec	00055	Lipták	Marcel	0908149658	liptak.marcel@gmail.com
12.	Zamestnanec	00056	Benková	Dominika	0917077897	benkova.dominika@gmail.com
13.	Zamestnanec	00049	Kecskéš	Matej	0905698521	kecskes.matej@gmail.com
14.	Zamestnanec	00068	Sloboda	Ján	0908554855	sloboda.jan@gmail.com

Obr. 30 Detail inventúry – zoznam zamestnancov

## G Výplatné pásky

Janka Bendiková	Inventúry	Zamestnanci	Výplatné pásky	Štatistiky	Odhlásit	?
-----------------	-----------	-------------	----------------	------------	----------	---

Výplatné pásky						«	2014	2015	2016	2017	2018	»
	Rok a mesiac: Máj - 2016	Počet inventúr: 0	Počet pokút: 0	Mzda v hrubom: 0 €	Pre tento mesiac ešte nie je možné zobraziť výplatné pásky.							
	Rok a mesiac: Apríl - 2016	Počet inventúr: 1	Počet pokút: 0	Mzda v hrubom: 66 €	Zobrazíť							
	Rok a mesiac: Marec - 2016	Počet inventúr: 0	Počet pokút: 0	Mzda v hrubom: 0 €	Zobrazíť							
	Rok a mesiac: Február - 2016	Počet inventúr: 5	Počet pokút: 0	Mzda v hrubom: 362 €	Zobrazíť							
	Rok a mesiac: Január - 2016	Počet inventúr: 10	Počet pokút: 0	Mzda v hrubom: 618.5 €	Zobrazíť							

Obr. 31 Zoznam výplatných pásek


Výplatná páska	Názov spoločnosti	Január 2016
20160100045	(00045) Janka Bendiková	(25) Všeobecná zdravotná poisťovňa, a.s.
Odpracované	83,00 Prerážky	0,00 Základná mzda
Dovolenka	0,00 Plat.volno	0,00 Mzda za nadčas
Sviatky	0,00 Neplat. voľno	0,00 Nadčasový prípl.
Nadčas	0,00 Náhr. voľno	0,00 Ostatné príplatky
Nočné	0,00 Absencia	0,00 Náhrady
Poh.na prac.	0,00 Vyňatie	0,00 Prémie a odmeny
Pod. doma	0,00 Iné	0,00 Jednoráz. odmeny
Nemoc	0,00	0,00 Ostatné odmeny
		618,50 Odmena štatutára
		0,00 Súbežný pr. pom
		0,00 Práca v zahraničí
		0,00 Zo soc. fondu
		0,00 Nat príjem zo SF
		0,00 Naturálie
		0,00 Fiktívna mzda
		0,00 Storno
00145   01.01.2016   A3Sport - Košice Vyplatené: 78 €		Hrubá mzda
00163   29.01.2016   Club 24 - Poprad Vyplatené: 48.5 €		
00155   08.01.2016   ThermGas - Poprad Vyplatené: 66 €		Starobné poistenie
00147   03.01.2016   Adidas - Košice Vyplatené: 42 €		Invalidné poistenie
00161   26.01.2016   EXIsport - Banská Bystrica Vyplatené: 60.5 €		Sociálne poistenie spolu
00149   04.01.2016   Hypernova Garbiarska - Liptovský Mikuláš Vyplatené: 66 €		Zdravotné poistenie
00160   22.01.2016   Alpine Pro - Prešov Vyplatené: 71.5 €		
00150   06.01.2016   Alpine Pro - Prešov Vyplatené: 48 €		Iné príjmy
00154   10.01.2016   Columbia - Košice Vyplatené: 54 €		Daň zo mzdy
00157   18.01.2016   Adidas - Košice Vyplatené: 84 €		Čistá mzda
		K výplate v hotovosti
		K výplate bankovým prevodom
		na účet <b>45364563/0300</b>
		<b>445.87 €</b>




V prípade akýchkoľvek nezhôd a pripomienok kontaktujte prosím manažéra spoločnosti. Kontaktné údaje najdete v informačnom systéme.

Obr. 32 Výplatná páska vo formáte PDF



## H Výstup pre archiváciu

 Martin Bendík

[Inventúry](#)
[Zamestnanci](#)
[Zakazníci](#)
[Rezervácie](#)
[Vozidlá](#)
[Štatistiky](#)
[Odhliásit](#)
  

[Vyplátné pásky](#)
[Mesačné uzávierky](#)
[Nastavenie miezd](#)

[«](#)
[<](#)
Marec 2016
Apríl 2016
Máj 2016
[>](#)
[»](#)
[Vynulovať](#)

[Zobrazit PDF](#)

[Zrušiť](#)

Obr. 33 Výber zobrazenia mesačných uzávierok

Kenvelo - Košice	2016040700167
<b>Identifikačné číslo inventúry:</b>	00167
<b>Typ inventúry:</b>	Sklad
<b>Typ spoločnosti:</b>	Knihkupectvo
<b>Názov spoločnosti:</b>	Kenvelo
<b>Adresa spoločnosti:</b>	Moldavská 32, 04011, Košice
<b>Zodpovedný vedúci:</b>	Bendiková Janka (00045) bendikova.janka@gmail.com 00421 903 789 635
<b>Začiatok:</b>	07.04.2016 11:15
<b>Koniec:</b>	07.04.2016 19:45
<b>Čas jednosmernej cesty:</b>	01:15:00
<b>Vzdialenosť jednosmernej cesty:</b>	140 Km
<b>Počet interných vozidiel:</b>	2
<b>Počet interných vodičov:</b>	2
<b>Počet externých vozidiel:</b>	0
<b>Počet externých vodičov:</b>	0
<b>Počet zamestnancov:</b>	7
<b>Vozidlá:</b>	PP-843-BJ   Toyota Avensis (interné vozidlo) KE-621-AA   Toyota Auris (interné vozidlo)
<b>Inventúru absolvovali:</b>	
VEI 00045	Bendiková Janka 66 €
SOF 00048	Južanský Peter 38 €
SOF 00046	Kobližná Veronika 38 €
ZAM 00072	Adásek Matej 27.76 €
ZAM 00067	Bendiková Monika 27.76 €
ZAM 00062	Dobrota Ľuboš 27.76 €
ZAM 00074	Lenčeš Michal 27.76 €
ZAM 00070	Mrkvička Jožko 27.76 €
ZAM 00051	Sedláčková Nikola 27.76 €
ZAM 00068	Sloboda Ján 0 € <span style="float: right;">Poznámka: Absencia</span>

Obr. 34 Detail inventúry vo formáte PDF