

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky

Informační systém realitní kanceláře  
Petr Váňa

Bakalářská práce  
2019

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Akademický rok: 2018/2019

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE (PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Petr Váňa**  
Osobní číslo: **I16156**  
Studijní program: **B2646 Informační technologie**  
Studijní obor: **Informační technologie**  
Název tématu: **Informační systém realitní kanceláře**  
Zadávající katedra: **Katedra informačních technologií**

### Z á s a d y p r o v y p r a c o v á n í :

Cílem této práce je vytvoření funkční aplikace, která bude umožňovat evidenci dat a řízení obchodní činnosti realitní kanceláře. Aplikace bude umožňovat tyto funkcionality: přístup uživatelů do systému podle jejich práv, evidenci nabízených nemovitostí k prodeji podle kategorií a parametrů, evidenci zákazníků realitní kanceláře včetně adresy, telefonu, e-mailu a dalších nutných údajů, zobrazení nabízených nemovitostí dle parametrů, které si v aplikaci zvolí zájemce o koupi nebo prodej nemovitosti a generování sestav dle volitelně zadaných kritérií. V teoretické části bude řešerše systémů, které se zabývají touto problematikou. Praktická část bude obsahovat popis použitých technologií, návrh databáze, ER diagram s využitím "Crow's Foot" notace entity-relationship a aplikační řešení. Pro vytvoření aplikace bude využit skriptovací jazyk PHP nebo JAVA a databáze MySQL nebo Oracle.

Rozsah grafických prací:

Rozsah pracovní zprávy: min. 30 stran

Forma zpracování bakalářské práce: tištěná

Seznam odborné literatury:

LACKO, Luboslav. Oracle - Správa, programování a použití databázového systému. Brno: Computer Press a.s., 2007. 573 s. ISBN 978-80-251-1490-2  
GROFF, James R. a Paul N. WEINBERG. SQL kompletní průvodce. Brno: Computer Press a.s., 2005. 936 s. ISBN 80-251-0369-2  
NARAMORE, Elizabeth, Jason GERNER, Scouarnec YANN LE and Timothy BORONCZYK. PHP 6, MySQL, Apache: Vytváříme webové aplikace. Brno: Computer Press a.s., 2009. 816 s. EAN:9788025127674  
DRUSKA, P. CSS a XHTML - tvorba dokonalých webových stránek krok za krokem, Grada, 2006. 200 s. ISBN: 80-247-1382-9

Vedoucí bakalářské práce:

Ing. Miloslav Macháček, Ph.D.

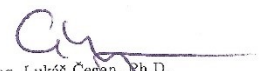
Katedra informačních technologií

Datum zadání bakalářské práce: 31. října 2018

Termín odevzdání bakalářské práce: 12. května 2019



Ing. Zdeněk Němec, Ph.D.  
děkan



Ing. Lukáš Cegan, Ph.D.  
pověřený vedením katedry

V Pardubicích dne 20. března 2018

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 3. 5. 2019

Petr Váňa

## **PODĚKOVÁNÍ**

Tímto bych rád poděkoval panu Ing. Miloslavu Macháčkovi, Ph.D. za vedení práce a rady při vypracovávání práce. Dále děkuji také svým rodičům za umožnění studia a podporu.

## **ANOTACE**

Bakalářská práce se zabývá problematikou vytvoření webové aplikace pro realitní kancelář. Teoretická část je zaměřena na historii, vývoj, technologie webových aplikací a řešerši systémů se stejným zaměřením. Praktická část obsahuje popis vytvářené aplikace. Použití jazyků HTML a CSS pro celkový vzhled webové stránky, PHP pro implementaci logiky na straně serveru, JavaScript nebo jQuery pro realizaci aplikační logiky na straně klienta, dále systém řízení báze dat MySQL pro využití databáze s příloženým ER diagramem obsahujícím návrh databáze.

## **KLÍČOVÁ SLOVA**

PHP, MySQL, CSS, HTML, JavaScript, jQuery, webová aplikace, informační systém, realitní kancelář

## **TITLE**

Information system of real estate agency.

## **ANNOTATION**

Bachelor thesis is focused on creating web application for real estate agency. Theoretical part is focused on history, evolution and technologies of web applications and there is a research of systems with same problematics. Practical part contains of description of created application, usage of HTML and CSS languages which are used for general look of the web page, PHP usage for application logic on server side, JavaScript or jQuery usage for application logic on client side. Furthermore, database management system MySQL usage for storing data in database with attached ER diagram of database design.

## **KEYWORDS**

PHP, MySQL, CSS, HTML, JavaScript, jQuery, web application, information system, real estate agency

# OBSAH

Seznam obrázků.....	9
Seznam tabulek .....	10
Seznam zdrojových kódů .....	11
Seznam zkratk .....	12
Úvod .....	13
<b>1 Teoretická část .....</b>	<b>14</b>
1.1 Historické zákonitosti .....	14
1.1.1 Statické webové stránky .....	14
1.1.2 Dynamické webové stránky.....	14
1.2 Technologie webu.....	15
1.2.1 HTML .....	15
1.2.2 CSS .....	15
1.2.3 PHP .....	16
1.2.4 JavaScript.....	17
1.2.5 jQuery .....	17
1.2.6 MySQL .....	18
1.3 Problematika systémů .....	19
1.3.1 Zákazník.....	20
1.3.2 Zaměstnanec .....	20
1.4 Rešerše .....	20
1.5 Dostupná řešení na trhu .....	21
1.5.1 M&M Reality.....	21
1.5.2 RE/MAX.....	22
1.5.3 Realitní kancelář EVROPA .....	23
1.5.4 Porovnání konkurenčních řešení.....	25
1.5.5 Porovnání s vytvářenou aplikací.....	26
<b>2 Praktická část.....</b>	<b>28</b>
2.1 Návrh systému, pro koho určeno .....	28
2.1.1 Storyboard.....	28
2.1.2 Určení a využití.....	29

2.2	Role a oprávnění .....	29
2.3	Funkcionalita .....	30
2.3.1	Zákaznická část .....	30
2.3.2	Administrační část .....	32
2.4	Rich picture, aktivity diagram, architektura a struktura .....	34
2.4.1	Rich picture .....	34
2.4.2	Aktivity diagram .....	35
2.4.3	Architektura a struktura .....	36
2.5	Použité technologie, databáze, SW .....	38
2.5.1	Technologie .....	38
2.5.2	Databáze .....	38
2.5.3	SW .....	38
2.6	Databáze .....	39
2.6.1	Tabulky .....	39
2.6.2	ER diagramy .....	45
2.7	Ukázka zdrojových kódů .....	46
2.8	Uživatelská příručka .....	48
2.8.1	Práce s vlastnostmi, tvoření filtrů .....	48
2.8.2	Změna stavu aktivity .....	51
2.8.3	Přidat nemovitost – vyplnění vlastností .....	51
2.9	Instalační příručka .....	52
	<b>Závěr .....</b>	<b>53</b>
	<b>Použitá literatura .....</b>	<b>54</b>
	<b>Přílohy .....</b>	<b>55</b>



## SEZNAM OBRÁZKŮ

Obrázek 1: Storyboard .....	28
Obrázek 2: Rich picture .....	34
Obrázek 3: Aktivitý diagram .....	35
Obrázek 4: MVC.....	36
Obrázek 5: ER diagram.....	45
Obrázek 6: ER diagram 2.....	46
Obrázek 7: ER diagram 3.....	46
Obrázek 8: Druh nabídky.....	48
Obrázek 9: Hlavní filtr .....	49
Obrázek 10: Klíčová slova.....	50
Obrázek 11: Přidat nemovitost – vlastnosti .....	51

## **SEZNAM TABULEK**

Tabulka 1: Porovnání řešení podle nabízených služeb .....	25
Tabulka 2: Subjektivní porovnání aspektů .....	26

## **SEZNAM ZDROJOVÝCH KÓDŮ**

Ukázka kódu 1: Validace .....	46
Ukázka kódu 2: Vytvoření PDO .....	47
Ukázka kódu 3: Vložení adresy .....	47

## SEZNAM ZKRATEK

AJAX	Asynchronous JavaScript And XML
CD	Compact Disc
CSS	Cascading Style Sheets
DOM	Document Object Model
ER	Entity-relationship
FAQ	Frequently asked questions
HTML	Hyper Text Markup Language
JSON	JavaScript Object Notation
MVC	Model, View and Controller
PDF	Portable Document Format
PHP	Hypertext Preprocessor
SQL	Structured Query Language
SW	Software
XML	Extensible Markup Language

# ÚVOD

Bakalářská práce je zaměřena na vytvoření webové aplikace, která bude umožňovat evidenci dat a řízení obchodní činnosti realitní kanceláře. Základní funkcionalita zahrnuje přístup uživatelů do systému podle jejich práv, evidenci nemovitostí podle kategorií a parametrů, evidenci uživatelů a možnost správy dat.

Návrh celé aplikace se odvíjí od řešerše vybraných systémů a je zaměřen především na jednoduchost, snahu o vylepšení a nápravu nedostatků stávajících systémů.

Aplikace je naprogramována v programovacím jazyce PHP s doplněním o JavaScript, potažmo jQuery. Vzhled webu je napsán v jazyce HTML s využitím CSS. Data jsou ukládána v relační databázi MySQL.

V první, teoretické, části se nachází popis historických zákonitostí webových stránek, výčet a seznámení s technologiemi pro vývoj. Dále je zde zahrnuto rozebrání problematiky systémů realitních kanceláří z pohledu zákazníků a zaměstnanců a nakonec řešerše vybraných systémů a jejich vzájemné porovnání.

V druhé, praktické, části je podrobně popsána vytvářená aplikace. Nejprve je text zaměřen na celkový návrh systému. Poté jsou detailně vysvětleny zavedené role, oprávnění, funkcionalita a návrh databáze, kde se nachází model a jeho důkladný popis. Na závěr jsou přiloženy některé ukázky kódu a uživatelská příručka, ve které jsou podrobně vysvětleny vybrané části aplikace a instalační příručka.

# 1 TEORETICKÁ ČÁST

## 1.1 Historické zákonitosti

Následující kapitola se zabývá historií webových stránek. Zaměřuje se na rozdíly mezi statickými a dynamickými weby a na jejich aplikování na systémy realitních kanceláří.

### 1.1.1 Statické webové stránky

Statické webové stránky jsou charakteristické tím, že jejich obsah, který je později zobrazen uživateli v prohlížeči, není nijak dynamicky generován na serveru. Označení statický ale neznamená, že se na stránce nemohou provádět žádné akce nebo změny, mezi které nepatří přesměrování na jinou stránku.

Řešení pro realitní kancelář:

- Vytvoření stránek pouze s informacemi o realitní kanceláři a kontaktními údaji.
  - Velmi jednoduché řešení.
  - Bez inzerce nemovitostí velká nevýhoda oproti konkurenčním systémům.
- Stejný web jako v předchozím bodě navíc s výpisem nemovitostí.
  - Nepřehledná správa.
  - Přidávání nemovitostí buď přímo do kódu, nebo do externího souboru (například XML, JSON), který by JavaScript zpracoval a vytvořil z něho obsah.
  - Žádná administrace.
  - Web by se mohl podobat dnešním řešením.

Z předchozích bodů je zřejmé, že pro realitní kanceláře statické webové stránky nejsou vhodným řešením, to ale neznamená, že jsou v dnešní době nepoužitelné. Jejich uplatnění může být efektivní v případě jednoduchých webů. Ty mají stále k dispozici JavaScript, který nabízí velkou škálu možností, jak web obohatit. Některé současné weby jsou implementovány celé v JavaScriptu za použití frameworku.

### 1.1.2 Dynamické webové stránky

Dynamické webové stránky jsou takové, kde obsah stránky může být dynamicky generován na serveru. Poskytují v podstatě vše, co statické stránky, a kromě toho nabízejí svoji dynamiku, ale jak již bylo zmíněno výše, dynamika není vždy nutná nebo vyžadovaná.

Řešení pro realitní kancelář:

- Vytvoření komplexních webových stránek.
  - Přehledná webová administrace.
  - Dynamické načítání obsahu.
  - Využití databáze.
  - Dnešní moderní řešení podobných systémů.

Z předchozích bodů vyplývá, že pro realitní kanceláře jsou dynamické webové stránky rozhodně vhodnější než statické. V dnešní době již není problém s dynamickým webem, jelikož poskytovatelé webhostingu obecně nabízejí podporu PHP, databáze apod.

## 1.2 Technologie webu

Následující kapitola se zabývá popisem technologií pro vytváření webových aplikací.

### 1.2.1 HTML

Zkratka HTML zastupuje označení Hyper Text Markup Language. Tento značkovací jazyk je používán pro vytváření webových stránek. Byl vytvořen v roce 1991 a od té doby prošel řadou verzí. Od roku 2014 se ovšem používá ustálená verze HTML5. Soubory HTML mají koncovku „.html“.

HTML pro vytváření struktury a jednotlivých prvků webu využívá značky. Tyto značky jsou nazývány tagy a jsou základním stavebním kamenem HTML. Jednotlivé tagy jsou zapisovány mezi špičaté závorky, mohou být buď párové, nebo nepárové. Párové mají už podle svého názvu tagy dva, počáteční a koncový, přičemž koncový se rozpozná tak, že má před názvem lomítko. Nepárové mají tag pouze počáteční.

[8]

### 1.2.2 CSS

Zkratka CSS zastupuje označení Cascading Style Sheets. Byl vytvořen společností World Wide Web Consortium. Nynější používaná verze je CSS3. Soubory CSS mají koncovku „.css“. Jak bylo zmíněno výše, HTML je použito pro vytvoření struktury a jednotlivých prvků, a CSS popisuje, jak by se prvky měly zobrazovat. Od HTML 3.2 bylo možné například přidávat barvu písma přímo do atributu tagu, ale toto řešení se ukázalo jako velmi nevýhodné, protože jeho použití bylo poměrně nepříjemné a neumožňovalo znovuvyužití stylu.

S příchodem CSS však přišla velká změna, tento jazyk umožňoval ušetření práce vývojářům hlavně tím, že jednotlivé styly mohly být použity na více tagů na více stránkách. Styly se ve

většinou případů přemístily pryč z HTML souborů do CSS souborů, které se dají jednoduše připojit k HTML stránce. Přesto styly mohou být zapsány do HTML stránek pomocí tagu `style` nebo přímo do tagu přes atribut `style`, ale nejvíce vhodná a používaná forma je ukládání do externích CSS souborů.

[8]

### 1.2.3 PHP

PHP je velice rozšířený skriptovací programovací jazyk pro tvorbu dynamických webových stránek. Zkratka PHP zastupuje označení Hypertext Preprocessor. Nejnovější vydaná verze je PHP 7.3.3 ze 7. března 2019. PHP je vydáván pod open source licenci. Dokáže fungovat na velké škále platform a je kompatibilní s mnoha používanými servery. Podporuje velké množství databází, se kterými komunikuje pomocí předpřipravených funkcí.

Soubory PHP mají koncovku „.php“, mohou obsahovat HTML, CSS, JavaScript, text a samotný PHP kód. Nejdříve je odeslán požadavek klientem na server, na něm je vykonána serverová část, PHP kód, a výsledek, HTML stránka, je následně zaslán klientovi, kterému se zobrazí v prohlížeči. V souborech se zapisuje mezi „<?php“ a „?>“, druhá část není povinná v případě, že se za ní nenachází nic dalšího například HTML kód. Tento kód se běžně v PHP části nemůže vyskytovat, je možné ho použít v příkazu `echo`, který ho zpracuje jako text a v nezměněné formě je předán interpretu.

Používané datové typy:

- String,
- Integer,
- Float nebo Double,
- Boolean,
- Array,
- Object,
- NULL,
- Resource.

Velkým rozdílem oproti většině programovacích jazyků je používaná typová dynamičnost. PHP automaticky přetypovává proměnné na základě přiřazované hodnoty. Také zde není žádný příkaz pro deklaraci proměnných, ty jsou vytvořené až v momentě, kdy se do nich načte hodnota.

[8], [6]



### 1.2.4 JavaScript

JavaScript je programovací jazyk používaný zejména pro vytváření chování webových stránek, ale používají ho i některé desktopové a serverové aplikace. V roce 1995 ho vytvořil Brendan Eich. Jeho název může budít dojem, že má něco společného s jazykem JAVA, ale není tomu tak.

Podobně jako u výše zmíněného CSS, může být napsán přímo v HTML, případně PHP souboru nebo také může být načtený z externího souboru, který má příponu „.js“. Podmínkou pro funkčnost je, že musí být zapsaný vně tagu „head“ nebo „body“. Čistý kód se pak zapisuje do párového tagu „script“, nebo se externě nahrává ze souborů s využitím atributu „src“, v těchto souborech se nachází už jen čistý kód. Nejvíce vhodná a používaná možnost je přes nahrávání ze souboru, rozděluje totiž HTML a JavaScript kód, což umožňuje lepší čtení a správu. JavaScript soubory mohou být dokonce ukládány do cache paměti, a tím zvyšují rychlost načítání stránky.

JavaScript má podobně jako PHP dynamické datové typy, které se deklarují přes klíčové slovo „var“. Využívá datové typy:

- string,
- number,
- boolean,
- undefined,
- null,
- array,
- object.

[8]

### 1.2.5 jQuery

jQuery je framework pro JavaScript. Jedná se o nejrozšířenější a nejpoblárnější framework. Má velkou podporu a funguje stejně v hlavních webových prohlížečích. Ulehčuje programování a je jednoduchý na naučení. Zapouzdřuje funkce, pro které by JavaScript potřeboval více řádků kódu. Nabízí tato rozšíření:

- **HTML/DOM manipulace** – Manipulace s HTML prvky pomocí objektového modelu HTML.
- **CSS manipulace** – Metody pro manipulaci s CSS.
- **HTML událostní metody** – Spustí nebo přidělí obslužnou metodu dané události pro vybrané elementy.
- **Efekty a animace** – Metody pro vytváření efektů a animací.

- **AJAX metody** – Umožňuje pomocí metod asynchronně komunikovat mezi klientem a serverem.
- **Další** – Například metody pro procházení seznamů apod.

## AJAX

Jak bylo zmíněno výše, pro načtení dynamického kontextu nebo dat z databáze apod. slouží PHP, tento kód se pouze spouští při načítání webové stránky. V některých případech je potřeba načíst taková data i za běhu a je žádoucí, aby nedošlo k obnově (znovunačení) stránky. K tomu slouží komunikace AJAX, která byla již možná před vytvořením jQuery frameworku, avšak ten ji velice zjednodušuje. Zkratka AJAX zastupuje Asynchronous JavaScript And XML, což může být zavádějící název, jelikož se pro přenos dat nepoužívá jen XML, ale i čistý text, nebo JSON.

Průběh v sedmi krocích:

- Nastane událost.
- XMLHttpRequest objekt je vytvořen.
- XMLHttpRequest objekt posílá požadavek na server.
- Server zpracuje požadavek.
- Server pošle zpátky odpověď.
- Odpověď se přečte.
- Jsou uplatněny změny.

[8], [2]

### 1.2.6 MySQL

MySQL je nejpopulárnější relační databázový systém. Je vydáván pod licencí open source. V současné době je vyvíjen a distribuován společností Oracle Corporation. Slouží pro přístup, přidávání, měnění a mazání dat uložených v počítačové databázi. Uživatel pouze zadává dotazy, umístění a přesnost uložení si obstará systém sám.

Relační databáze ukládá data strukturovaně do více tabulek. Tabulka obsahuje sloupce a každý řádek představuje jeden záznam dat. Mezi jednotlivými tabulkami pak vznikají různé vztahy, například „1:1“, což znamená, že jednomu konkrétnímu záznamu odpovídá právě a pouze jeden záznam z tabulky druhé. Různé rozdělení těchto vztahů pak určují:

- **Kardinalita** – Určuje, kolik možných relací lze vytvořit pro danou entitu. Jsou tři možné druhy:
  - **1:1** – Například jeden člověk má právě jedno rodné číslo.
  - **1:N** – Například jeden člověk může vlastnit více než jeden automobil.

- **M:N** – Například vztah mezi zaměstnanci a zaměstnavateli, zaměstnanec může mít více než jednoho zaměstnavatele a zároveň zaměstnavatel může mít více než jednoho zaměstnance. Pro vyjádření tohoto vztahu v databázi se vytváří nová propojovací tabulka.
- **Parcialita** – Určuje povinnost, nebo nepovinnost vztahu, například je-li člověk povinen mít mobilní telefon. Není, tudíž vztah mezi tabulkami člověk a mobilní telefon by mohl být „0..1:0..N“, v tomto konkrétním případě se předpokládá, že jeden telefon nemohou vlastnit dva lidé a telefon nemusí mít žádného vlastníka.

[8], [5]

## SQL

SQL je standardizovaný strukturovaný dotazovací jazyk pro přístup a manipulaci s databázemi.

Zkratka SQL zastupuje Structured Query Language. Základní klíčová slova jazyka:

- **SELECT** – Výběr dat.
- **INSERT** – Vložení dat.
- **UPDATE** – Změna dat.
- **DELETE** – Mazání dat.
- **WHERE** – Vytváření restrikce.
- **ORDER BY** – Seřazení výsledků.

## SQL omezení

SQL omezeními se specifikují pravidla pro data v tabulce, dají se buď aplikovat na úrovni tabulky, nebo sloupců. Příklady často používaných omezení:

- **NOT NULL** – Zajišťuje, že data v tomto sloupci nemohou nabývat hodnoty NULL.
- **UNIQUE** – Zajišťuje, že data v tomto sloupci se nemohou opakovat.
- **PRIMARY KEY** – V sloupci s tímto označením je unikátní primární identifikátor jednotlivých řádků. Sám o sobě má vlastnosti NOT NULL a UNIQUE.
- **FOREIGN KEY** – V sloupci s tímto označením je unikátní identifikátor jednotlivých řádků z cizí tabulky.
- **CHECK** – Kontrola, že data v daném sloupci nabývají specifických hodnot.
- **DEFAULT** – Nastavení defaultní hodnoty pro daný sloupec, použitá v případě, že není zadána žádná hodnota.
- **INDEX** – Index slouží pro rychlejší získání dat z databáze.

[8]

## 1.3 Problematika systémů

Následující kapitola se zabývá problematikou systémů realitních kanceláří. Nejdříve jsou zde probrány základní požadavky a očekávání zákazníka, následně je popsána daná problematika z pohledu zaměstnance.

### **1.3.1 Zákazník**

Nejdůležitějšími vlastnostmi aplikace jsou vzhled, atraktivnost, rychlost, přívětivost, jednoduchost a funkčnost. Konkurence na trhu velkou mírou ovlivňuje úspěšnost, proto by neměl být zanedbán nebo podceněn ani jeden z výše uvedených bodů, a to ani při nízké nebo žádné konkurenci.

Kvalita prvního dojmu, který si jakýkoliv zákazník (při návštěvě webu) utvoří, se do značné míry odvíjí od vzhledu dané aplikace. Pokud pro zákazníka není dostatečně atraktivní, pravděpodobnost, že ji přestane využívat, se zvyšuje. Atraktivita je samozřejmě subjektivní a nelze vyhovět každému uživateli, ale je vhodné se držet módních trendů. Udržování a aktualizování vzhledu bývá nedílnou součástí činnosti vývojáře.

Další aspekt, který by mohl zákazníka odradit od používání aplikace, je složitost nabízených funkcí. Například při vyplňování formuláře by měl být zákazník schopen bez problému vyplnit všechny povinné položky.

Nefunkčnost uživatel nemusí vůbec rozpoznat, např. pokud se mu nezobrazí všechny nemovitosti, které by podle zadaného filtru měly, avšak takovýto nález může znamenat ztrátu potenciálního zákazníka již nadobro.

### **1.3.2 Zaměstnanec**

Nejdůležitější vlastností je funkčnost, nejméně důležitou je naopak vzhled. Zde není potřeba zaměstnance nalákat na používání aplikace, a jelikož ji pravděpodobně využívá poměrně často, je seznámen s jejím prostředím a snadno se v ní vyzná, oproti zákazníkovi, u kterého platí pravý opak.

Nejčastěji systém bude využívat makléř dané realitní kanceláře s umožněným přístupem do administrační části aplikace. Měl by mít možnost zde jednoduše přidávat nové nemovitosti nebo měnit stávající, tudíž je pro něj výše zmíněná funkčnost klíčová.

## **1.4 Rešerše**

Rešerše je zaměřená na dostupná SW řešení zabývající se problematikou systémů realitních kanceláří. Všechna nalezená řešení jsou realizována jako webové aplikace.

### **Obecné vlastnosti**

- Vyhledávání nemovitostí podle různých filtrů.
- Informace o nemovitosti na oddělené stránce s podrobnými informacemi, galerií s popisem fotografií a kontaktem na makléře.

- Možnost vytvoření poptávky.
- Možnost vložit nemovitost k prodeji přes formulář.
- Výpis stávajících poboček a zaměstnanců.
- Nabízení dalších služeb.

[3], [7], [1]

## 1.5 Dostupná řešení na trhu

Následující kapitola se zabývá analýzou dostupných řešení systému realitních kanceláří na českém trhu. Jsou zde popsány základní informace, nabízené služby, vyhledávací schopnosti a další. Rovněž jsou popsány výhody a nevýhody vybraných systémů, které jsou následně porovnány i s vytvářeným systémem.

### 1.5.1 M&M Reality

M&M reality je česká společnost, zabývající se prodejem nemovitostí už od roku 2005. Za své dlouholeté působení na českém trhu si dokázala vybudovat vynikající pověst a jejich služeb využívá ročně přes 30 000 zákazníků. Tato společnost patří jednoznačně mezi špičku tuzemských realitních kanceláří.

#### Služby

- **Prodat nebo pronajmout nemovitost** – Vyplnění jednoduchého formuláře obsahující pole pro základní kontaktní údaje, popis, lokalitu a zprávu. Následně proběhne navázání kontaktu se zákazníkem.
- **Poptávka** – Vyplnění formuláře se zadáním jednotlivých parametrů, které má hledaná nemovitost splňovat, a kontaktních údajů, na které po dobu tří měsíců budou zákazníkovi posílány informace o nových nemovitostech dle daného výběru.
- **Hypotéka a pojištění** – Poskytování poradenství ohledně hypoték a pojištění na nemovitosti a jejich zřízení.
- **Výkup nemovitostí** – Možnost rychlého odkupu nemovitostí až s 85% tržní hodnotou.
- **Online poradna** – Výčet FAQ s možností konkrétního dotazu.

#### Vyhledávání a filtrování

Webová stránka nabízí možnost přímo vyhledat nemovitost podle identifikačního čísla nebo použít filtr, který se v základní podobě skládá ze čtyř položek:

- druh nabídky,
- typ nemovitosti,
- lokalita,
- cena.

V položce lokalita je možnost vybrat jeden nebo více krajů s případným upřesněním podle okresu, města, městské části nebo ulice. Při výběru typu nemovitosti se filtr rozšíří o několik dalších položek, které souvisí s vybraným typem. S každou změnou se provádí přepočítání celkového počtu nemovitostí, které odpovídá daným parametrům.

### **Pobočky a zaměstnanci**

Pobočky se dají nalézt na položce kontakty, jsou vyfiltrované podle krajů a informace o nich poskytuje jejich detailní stránka, na které se nachází i seznam všech zaměstnanců.

[3]

## **1.5.2 RE/MAX**

Společnost RE/MAX byla založena v roce 1973 v USA. Díky své revoluční strategii postupně dokázala expandovat téměř do celého světa. Do České republiky se dostala v roce 2005. V Česku se nachází přes 100 jejích poboček, po celém světě jich je dokonce 7 900.

### **Služby**

- **Prodat nebo pronajmout nemovitost** – Vyplnění jednoduchého formuláře obsahující pole pro základní kontaktní údaje, typ nemovitosti, lokalitu a krátký popis. Následně proběhne navázání kontaktu se zákazníkem.
- **Poptávka** – Vyplnění formuláře se zadáním jednotlivých parametrů, které má hledaná nemovitost splňovat, a kontaktních údajů, na které bude zákazník v případě nálezů kontaktován.
- **RX FINANCE** – Poskytnutí hypotéky s mnoha výhodami.
- **RE/MAX club** – Věrnostní program pro zákazníky, díky kterému je možné získat různé výhody u partnerů RE/MAX.
- **Aukce** – V případě dvou a více zájemců je možnost prodat nemovitost pomocí aukce, a získat tak nejvyšší možnou cenu.
- **Online poradna** – Formulář na kontakt pro jakýkoliv dotaz.

### **Vyhledávání a filtrování**

Hlavní filtr:

- kategorie,
- druh nabídky,
- kraj,
- upřesnit lokalitu.

Toto filtrování je umožněno vždy. V kategoriích je možnost vybrat mezi:

- projekty,
- zahraniční,
- oblíbené,
- typy nemovitostí.

Pokud je vybrána jedna z prvních tří možností, můžeme ještě zvolit typ nemovitosti, v opačném případě se zobrazí rozšiřující filtr:

- velikost,
- druh objektu,
- stav objektu,
- vlastnictví,
- podlaží,
- velikost pozemku,
- další vlastnosti.

### **Pobočky a zaměstnanci**

Jednotlivé pobočky jsou vyfiltrovány podle krajů v položce kanceláře. Informace o nich poskytuje jejich detailní stránka, na které se nachází i seznam všech zaměstnanců. Konkrétní makléři se dají vyhledat stejným způsobem na položce makléři.

[7]

### **1.5.3 Realitní kancelář EVROPA**

Společnost Realitní kancelář EVROPA byla založena v roce 1997. Působí zejména v České republice a na Slovensku, kde má přes 40 poboček (v ČR 31).

#### **Služby**

- **Prodat nebo pronajmout nemovitost** – Vyplnění jednoduchého formuláře obsahující pole pro typ nemovitosti, adresu, krátký popis, pobočku a základní kontaktní údaje. Následně proběhne navázání kontaktu se zákazníkem.
- **Poptávka** – Vyplnění formuláře s možností výběru typu nemovitosti a parametrem upřesňujícím daný typ, zadání maximální ceny, lokality, pobočky a kontaktních údajů. Každá poptávka je poté vystavena makléřem na stránce se všemi poptávkami.
- **Trezor 48** – Funkce Trezor 48 slouží pro jedinečnou příležitost získat informace o nově přidaných nemovitostech, které se v běžné nabídce zobrazují až po 48 hodinách. Stačí vyplnit formulář s téměř totožnými možnostmi jako u výše zmíněných poptávek. Tímto zákazník může získat výhodu, která mu pomůže k vytoužené koupi nemovitosti.

- **Online odhad ceny** – Vyplnění formuláře popisující nemovitost. Následné zaslání odhadované ceny do 24 hodin. Pro přesný odhad jsou využívány specializovaná databáze a cenové mapy.
- **Výkup nemovitostí** – Možnost rychlého odkupu nemovitostí například v případě finanční tísně.
- **Expats relocation** – Nabídka pro zahraniční klienty, která přináší komplexní řešení při vyhledání a zajištění přechodného bydlení. Tým EXPAT podává i další nutné informace o bydlení v dané lokalitě.
- **Online poradna** – Formulář na kontakt pro jakýkoliv dotaz.
- **Hypotéka** – Poskytování poradenství ohledně hypoték. U některých nemovitostí přímo možnost kontaktovat hypotečního poradce.

### Vyhledávání a filtrování

Hlavní filtr:

- kategorie,
- druh nabídky,
- kraj,
- upřesnit lokalitu,
- cena.

Toto filtrování je umožněno vždy. V kategoriích jsou možnosti:

- typy nemovitostí,
- novinky,
- zlevněné.

Přičemž lze vybrat více druhů kategorií s možností výběru, zda se jedná o novinku nebo slevu, což z daného zobrazení na stránce nemusí být na první dojem zřejmé. Dále také možnost rozšířeného hledání:

- specifický parametr pro vybraný typ nemovitosti,
- stav,
- stavba,
- plocha,
- makléř,
- pobočka,
- číslo zakázky.

Nutno konstatovat, že položky makléř, pobočka a číslo zakázky vytvářejí průnik množiny. Zvolením některé pobočky se nijak nezmění možnost výběru makléře.



## Pobočky a zaměstnanci

Jednotlivé pobočky je možno vyfiltrovat podle krajů. Zobrazený seznam poskytuje o každé pobočce základní informace, jejich bližší charakteristika je uvedena na detailní stránce každé z nich, kde se mimo jiné nachází i výpis zaměstnanců. Konkrétní makléře je možné vyhledat přes jejich specializaci na daný typ nemovitosti a kraj.

[1]

### 1.5.4 Porovnání konkurenčních řešení

V následující kapitole budou porovnány analyzované systémy podle nabízených služeb a podle osobního hodnocení daných aspektů. Všechna řešení nabízejí velmi podobné služby.

Tabulka 1: Porovnání řešení podle nabízených služeb

	M&M	RE/MAX	Realitní kancelář EVROPA
Prodej a pronájem	Ano	Ano	Ano
Poptávka	Ano	Ano	Ano + trezor
Výpis poboček	Ano	Ano	Ano
Výpis makléřů	Ano	Ano	Ano
Funkční filtrování	Ano	Ne vždy	Ano
Podrobné filtrování	Ne	Ano	Ano
Hypotéka	Ano	Ano	Ano
Výkup nemovitosti	Ano	Ne	Ano
Online odhad ceny	Ne	Ne	Ano
Aukce	Ne	Ano	Ne
Online poradna	Ano + FAQ	Ano	Ano
Validní web	Ne	Ne	Ne
Tisk nemovitosti	Ano	Ano	Ano
Nemovitost do PDF	Ano	Ne	Ne
Přeposlat na e-mail	Ano	Ano	Ano
Hlídaní změn u nemovitosti	Ne	Ne	Ano
Expats relocation (přechodný pobyt)	Ne	Ne	Ano

[3], [7], [1]

Následující tabulka zachycuje subjektivní názor autora, daná hodnocení vycházejí z webových stránek jednotlivých realitních kanceláří. Je využit hodnotící systém 1-5, přičemž známka 1 je nejlepší. Každé hodnocení může obsahovat dodatečný komentář.

Tabulka 2: Subjektivní porovnání aspektů

	M&M	RE/MAX	Realitní kancelář EVROPA
Celkový vzhled	2 – působí méně moderně	1	1
Filtrování nemovitostí	1 – jednoduchý filtr	3 – komplexnější filtr obsahuje nesmyslné vyhledávání	2 – komplexnější filtr není vždy jasný
Formuláře	1 – přehledné a jednoduché	2	1 – přehledné a velmi hezké provedení
Nabízené služby	1	1	1
Celková přehlednost	1	1	1
Popularita na trhu	1	1	2

[3], [7], [1]

### 1.5.5 Porovnání s vytvářenou aplikací

Vytvářený systém se zaměřuje na jednoduchost a efektivitu. Nejčastější požadavek přicházejícího zákazníka bude vyhledávání nemovitostí ke koupi, nebo vytváření požadavků na prodej, proto se navrhovaná aplikace soustředí na tato vylepšení oproti analyzovaným systémům.

Filtr nemovitostí se skládá ze dvou částí, hlavní filtr:

- druh nabídky,
- typ nemovitosti,
- vybrané filtry k danému typu nemovitosti,
- region,
- cena.

Kromě ceny, která je staticky umístěná na stránce, se ostatní filtry nahrávají podle dat z databáze, které je možno editovat v administraci, takže systém není omezen v případě rozšiřování. Při výběru regionu se ještě zobrazí možnost upřesnit lokalitu. Při zvolení typu

nemovitosti se načtou vybrané filtry k danému typu nemovitosti jak do hlavního filtru, tak i do rozšiřujícího, který je možno libovolně zobrazit a skrýt, a tak nepřekáží, není-li využíván.

Výhody proti stávajícím systémům:

- **M&M** – Vyhledávání nabízí i rozšiřující filtr. Jednodušší formulář na požadavek prodeje. Webové stránky nejsou validní.
- **RE/MAX** – Rozšiřující filtr neobsahuje nesmyslné položky, ale pouze relevantní k danému typu nemovitosti. Jednodušší formulář na požadavek prodeje. Webové stránky nejsou validní.
- **Realitní kancelář EVROPA** – Rozšiřující filtr je přehlednější. Webové stránky nejsou validní.

### **Pobočky a zaměstnanci**

Jednotlivé pobočky je možno vyfiltrovat podle krajů. Zobrazený seznam poskytuje potřebné informace o každé z nich. Na výpis makléřů se lze dostat prostřednictvím jednotlivých odkazů, odkud je pak dále možné otevřít detailní stránku každého makléře.

[3], [7], [1]

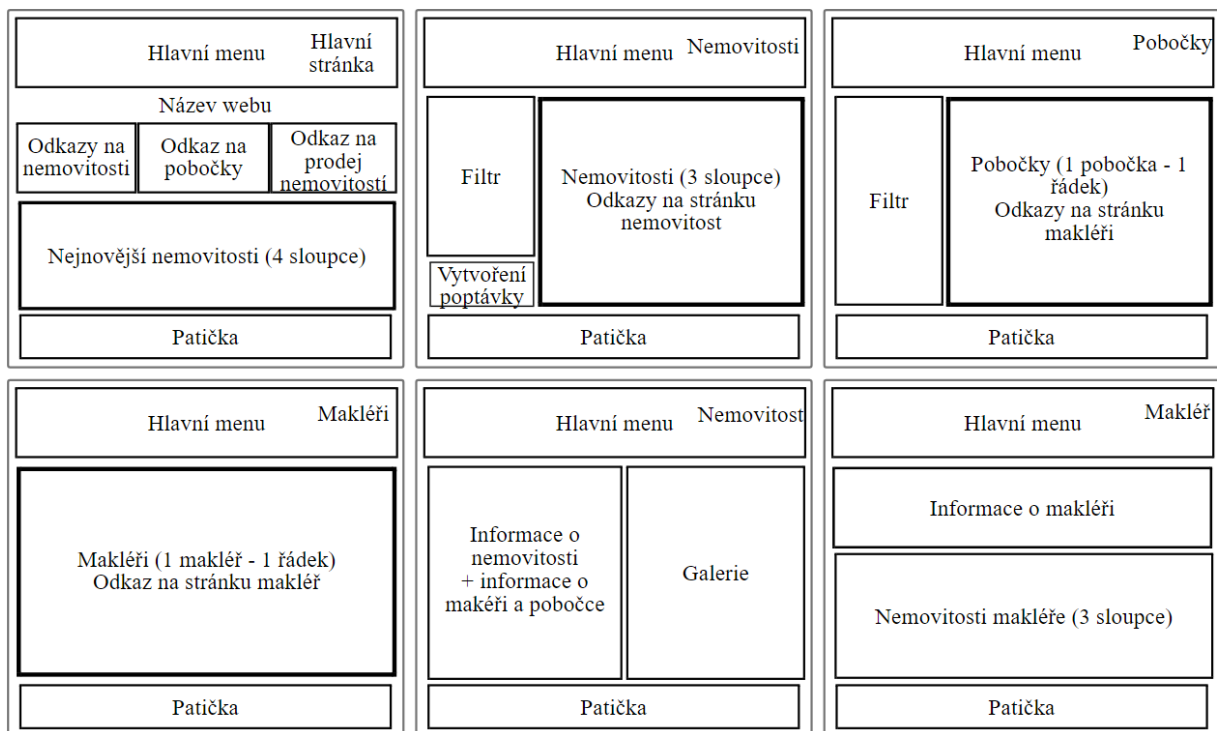
## 2 PRAKTICKÁ ČÁST

V následující kapitole bude podrobně představena vytvářená aplikace pro systém realitní kanceláře. Je zde popsán návrh vzhledu aplikace, cílové zaměření, použité role a oprávnění, funkcionalita, databáze, použitá architektura a technologie. Součástí jsou také diagramy popisující chod aplikace a obecně celého systému. Na závěr přiloženy ukázky kódů, uživatelská a instalační příručka.

### 2.1 Návrh systému, pro koho určeno

#### 2.1.1 Storyboard

Storyboard uvádí rozložení jednotlivých stránek v aplikaci. Jsou v něm naznačeny přechody mezi stránkami a další různé akce. Název dané stránky se nachází v pravém rohu v jednotlivých oknech.



Obrázek 1: Storyboard

Všechny stránky jsou rozděleny do tří částí.

- **Hlavička** – Nachází se v ní menu.
- **Tělo** – Nachází se v něm obsah dané stránky.
- **Patička** – Nachází se v ní dodatečné informace.

### 2.1.2 Určení a využití

V následující kapitole bude popsáno určení a využití vytvářené aplikace.

#### Realitní kancelář

Vytvářená aplikace je určena pro společnosti (realitní kanceláře), které chtějí inzerovat pohodlně své nemovitosti. Aplikace umožňuje veškerou nutnou správu v administrační části, také dovoluje neomezený růst kategorií, ve kterých jsou nemovitosti inzerovány, a společně s tím vytvoření celého filtru pro danou kategorii.

#### Zákazníci

Tato aplikace je určena pro zákazníky, kteří preferují přehledné nabídky a inzeráty. Umožňuje jim požadovanou nemovitost rychle najít a jednoduše kontaktovat makléře. Stejně tak ji mohou využít v opačném případě, chtějí-li svoji nemovitost snadno a rychle prodat.

## 2.2 Role a oprávnění

V této kapitole je popsáno rozdělení rolí, jejich význam v systému a možnost přidělení oprávnění.

#### Základní role

- administrátor,
- zaměstnanec,
- zákazník.

Různých typů zaměstnanců může být libovolný počet, v systému jsou zavedeny zaměstnanecké role makléř a ředitel pobočky.

#### Zavedená oprávnění systému

- **Administrátor** – Speciální oprávnění, které má pouze administrátor.
- **Přístup do administrace** – Základní oprávnění pro přístup do administrace.
- **Přidávání pobočky** – Oprávnění na přidání nové pobočky.
- **Přidávání zákazníka** – Oprávnění na přidání nového zákazníka.
- **Přidávání zaměstnance** – Oprávnění na přidání nového zaměstnance. Toto oprávnění není možné přidělit vytvářenému uživateli, ten dostane pouze oprávnění Přístup do administrace.
- **Správa nemovitosti** – Oprávnění na správu nemovitostí, které přihlášený zaměstnanec přidal.
- **Správa pobočky** – Oprávnění na správu pobočky, pod kterou přihlášený zaměstnanec spadá.
- **Správa zaměstnanců** – Oprávnění na správu zaměstnanců na pobočce, pod kterou přihlášený zaměstnanec spadá, neznamená možnost úpravy oprávnění zaměstnanců.

- **Správa míst** – Oprávnění na správu krajů, okresů a obcí.
- **Správa vlastností** – Oprávnění na správu vlastností, které jsou přiřazovány nemovitostem (stavy, typy nemovitostí, vlastnosti typů nemovitostí a klíčová slova spadající pod vlastnosti typů nemovitostí).
- **Správa nových zakázek** – Oprávnění na přidělování nových zakázek makléřům.
- **Správa zákazníků** – Oprávnění na správu zákazníků na pobočce, pod kterou přihlášený zaměstnanec spadá.
- **Správa oprávnění** – Oprávnění na možnost přidělení nebo odebrání oprávnění zaměstnanci (není možné upravovat administrátora). Úzce souvisí s oprávněními přidávání zaměstnance a správa zaměstnanců, bez kterých není možné zaměstnancům oprávnění přidávat nebo odebrat.
- **Přístup do transakcí** – Oprávnění na možnost přístupu k provedeným transakcím.

#### **Přidělená oprávnění podle role**

- **Zákazník** – Uživatel zákazník nemá a nemůže dostat žádná z výše uvedených oprávnění.
- **Administrátor** – Tento uživatel je jedinečný tím, že má všechna výše uvedená oprávnění. Pro správnou funkčnost je nezbytné, aby měl přidělena všechna oprávnění, jelikož samotné právo administrátor je samo o sobě k ničemu a nedovoluje ani přihlášení do systému.
- **Zaměstnanec** – Výjimka u tohoto uživatele je, že nemůže získat oprávnění „Administrátor“. Role spadající do této skupiny (makléř, ředitel pobočky) nemají žádná defaultní práva. Proto je vhodné přidělení minimálních nutných oprávnění pro schopnost vykonávat svoji práci.

## **2.3 Funkcionalita**

Tato kapitola popisuje funkcionalitu webové aplikace, je rozdělena na zákaznickou a administrační část. Podrobné ovládání vybraných funkcí se nachází v kapitole 2.8.2.8 Uživatelská příručka.

### **2.3.1 Zákaznická část**

Základní možnosti zákazníků:

- filtrování a prohlížení nemovitostí,
- ukládání filtrů (vytvoření poptávky),
- procházení poboček a makléřů,
- podání nemovitosti na prodej, či k pronájmu.

#### **Filtrování a prohlížení nemovitostí**

Vyhledávání nemovitostí se provádí na stránce nemovitosti, pro upřesnění hledání slouží filtr, který se nachází v levé části obrazovky. Hlavní část zákazník vidí neustále, podrobnější

vyhledávání si může zobrazit v případě nutnosti, to se mu odkryje při kliknutí na tlačítko „Zobrazit podrobnější vyhledávání“, které lze opětovně skrýt.

Jednotlivé kategorie filtru jsou označeny nadpisem tučně. Filtrování je velmi intuitivní, pokud zákazník v dané kategorii vybere jednu nebo více položek, jsou mu zobrazeny právě nemovitosti, které splňují dané podmínky, to samé platí u hodnot „od ... do“, kde je možné vyplnit jen jednu část. Pokud položka není vyplněna, není na ni brán zřetel.

### **Ukládání filtrů (vytvoření poptávky)**

Tato možnost se nachází přímo pod výše zmíněným vyhledáváním. Slouží pro jednoduché znovupoužití vyplněného filtru nebo pro vytvoření poptávky. Zákazník si může nechat vygenerovat odkaz, který mu usnadní návrat do aplikace v případě, že chce prohledávat nemovitosti se stejnými parametry.

### **Procházení poboček a makléřů**

Pro získání informací o pobočce nebo makléřích slouží stránky pobočky a „makléři“. Stránka pobočky nabízí výpis všech poboček realitní kanceláře s možností filtrování podle regionu. Přes odkaz u jednotlivých poboček se uživatel dostane na stránku makléři, kde se nachází výpis všech makléřů na dané pobočce. Pro více informací stačí rozkliknout odkaz u makléřů.

### **Podání nemovitosti na prodej, či k pronájmu**

V tomto případě zákazník vyplní jednoduchý formulář, nacházející se na stránce prodat nemovitost, ve kterém zadá kontaktní údaje, krátce popíše nemovitost a vybere pobočku pro zprostředkování prodeje.

### 2.3.2 Administrační část

V této podkapitole jsou popsány funkce administrace. Jsou zahrnuty pohledy administrátora, který má jedinečná oprávnění na vše, a zaměstnance s oprávněními, která mu umožňují provádět popisované operace (zaměstnanec s maximálním možným oprávněním).

V následujícím výčtu stránek jsou popsány funkce, které se nachází na jednotlivých stránkách.

- **Login** – Umožňuje přihlášení podle přihlašovacího jména a hesla s následným přístupem do administrace.
- **Logout** – Zajišťuje odhlášení uživatele.
- **Zákazníci** – Obsahuje seznam zákazníků z pobočky přihlášeného zaměstnance (administrátor má možnost výběru pobočky). Nabízí možnost aktivace/deaktivace uživatele přes zaškrtačací políčko s popisem „Aktivovat“ a odkaz na úpravu údajů uživatele.
- **Zaměstnanci** – Jedná se o seznam zaměstnanců z pobočky přihlášeného zaměstnance (administrátor má možnost výběru pobočky). Obsahuje možnost aktivace/deaktivace uživatele přes zaškrtačací políčko s popisem „Aktivovat“ a odkaz na úpravu údajů uživatele.
- **Upravit uživatele** – Umožňuje změnu údajů uživatele, v případě, že se jedná o zaměstnance dále i změnu přihlašovacího jména, hesla, fotografie a oprávnění.
- **Nemovitosti** – Obsahuje seznam nemovitostí přihlášeného zaměstnance (administrátorovi se zobrazí seznam nemovitostí na vybrané pobočce). Nabízí možnost aktivace/deaktivace nemovitosti přes zaškrtačací políčko s popisem „Aktivovat“ a odkaz na editaci nemovitosti.
- **Upravit nemovitost** – Zprostředkovává úpravu dané nemovitosti pomocí editace galerie, přidávání nové fotografie, úpravy textů k jednotlivým fotografiím, mazání. Dále je zde možnost provést transakci nemovitosti.
- **Přidat nemovitosti** – Umožňuje přidání nové nemovitosti přihlášenému makléři (administrátor volí makléře). Nabízí přímé nahrání fotografií do galerie, ale úprava textů k jednotlivým fotografiím se nachází až na stránce „Upravit nemovitost“. Nově přidaná nemovitost je defaultně neaktivní.
- **Pobočky** – Obsahuje seznam poboček, jenž je možný filtrovat podle kraje. Přihlášený zaměstnanec má možnost aktivovat/deaktivovat pouze pobočku, pod kterou spadá. Administrátor může aktivovat/deaktivovat všechny pobočky a také vidí odkazy na jejich editaci.
- **Upravit pobočku** – Umožňuje úpravu údajů pobočky, pod kterou přihlášený zaměstnanec spadá (administrátor má možnost editace všech poboček). Dále je také možno nahrát novou fotografii místo stávající.
- **Přidat pobočku** – Umožňuje přidání nové pobočky.
- **Vlastnosti nemovitostí**
  - **Druh nabídky** – Obsahuje přidávání, editaci a mazání druhu nabídky.
  - **Typ nemovitosti** – Umožňuje přidávání, editaci a mazání typů nemovitostí.

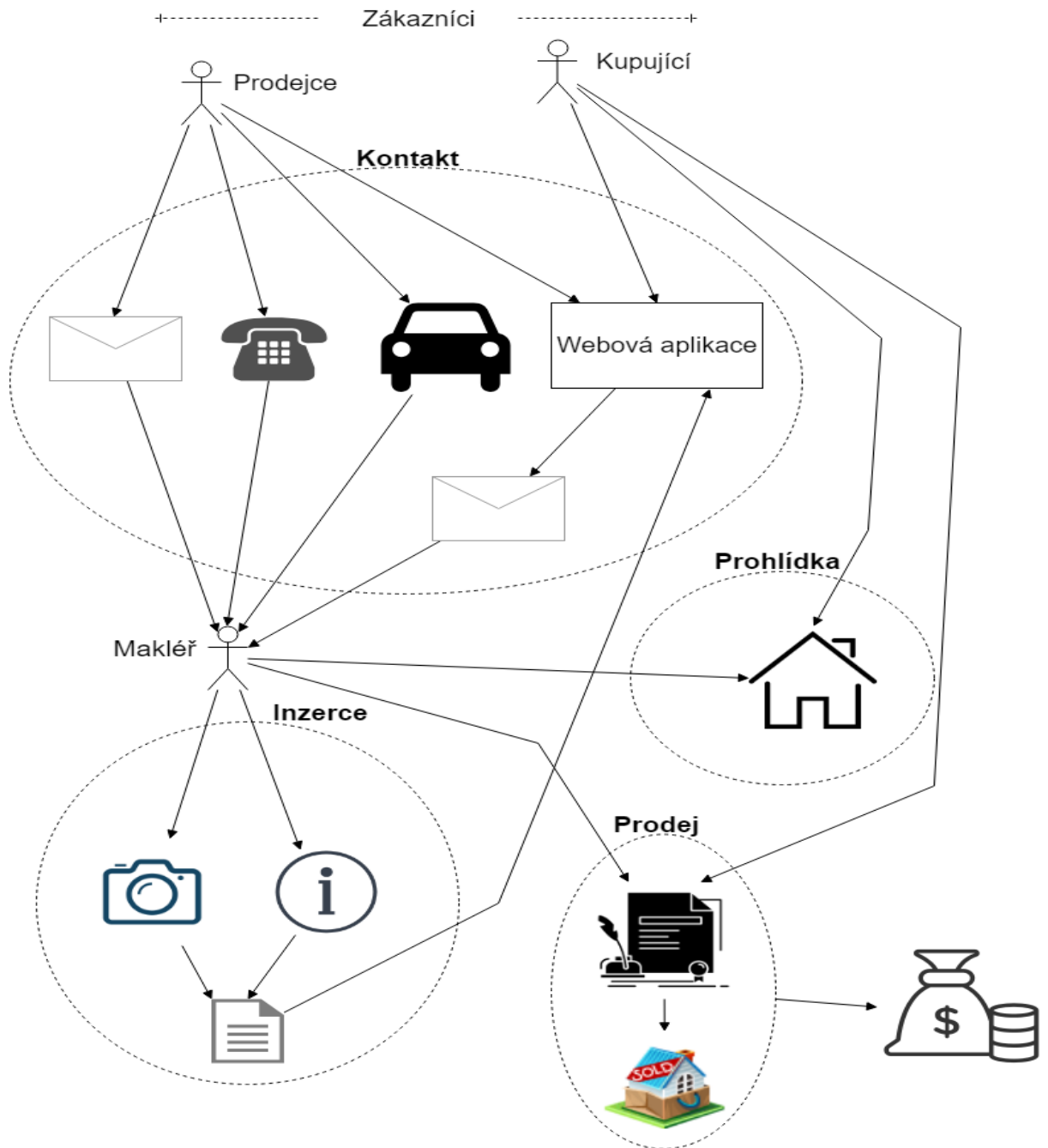


- **Vlastnosti typu nemovitosti** – Obsahuje přidávání, editaci a mazání vlastností, které spadají pod daný typ nemovitosti.
- **Klíčová slova vlastností** – Umožňuje přidávání, editaci a mazání klíčových slov, která patří pod danou vlastnost typu nemovitosti.
- **Zakázky** – Obsahuje seznam nově přichozích požadavků na prodej nemovitostí na pobočce přihlášeného zaměstnance (administrátor může vybírat mezi pobočkami). Také umožňuje přiřadit zakázku makléři, nebo zrušit požadavek.
- **Transakce** – Obsahuje seznam provedených transakcí na pobočce přihlášeného zaměstnance (administrátor může vybírat mezi pobočkami). Je možno filtrovat podle makléře nebo data a řadit podle stáří nebo ceny. Také nabízí možnost zobrazit počet provedených transakcí makléřů.
- **Kraje** – Umožňuje přidávání nového kraje a obsahuje seznam již stávajících krajů s možností úpravy jejich názvu a kódu.
- **Okresy** – Umožňuje přidávání nového okresu, který spadá pod vybraný kraj. Výběr tohoto kraje filtruje seznam stávajících okresů s možností úpravy jejich názvu a kódu.
- **Obce** – Umožňuje přidávání nové obce, která spadá pod vybraný kraj a okres. Výběr tohoto kraje a okresu filtruje seznam stávajících obcí s možností úpravy jejich názvu a kódu.

## 2.4 Rich picture, aktivity diagram, architektura a struktura

V následující kapitole se nachází rich picture, který popisuje obecně chod realitní kanceláře s využitím vytvářené aplikace. Dále je zde uveden aktivity diagram, který detailně popisuje danou činnost. Nakonec je popsána použitá architektura pro vytvoření aplikace doplněna o ukázkou její struktury.

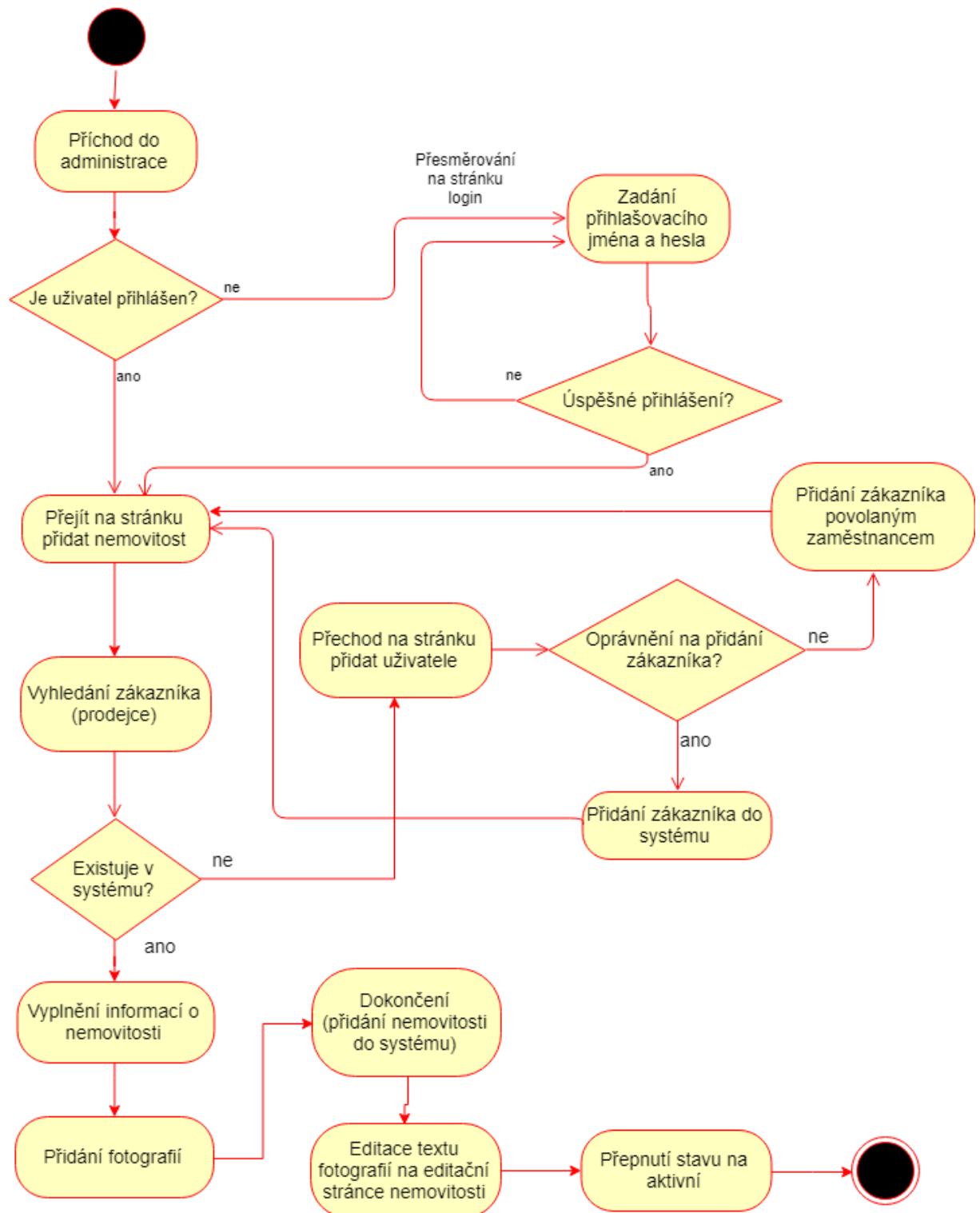
### 2.4.1 Rich picture



Obrázek 2: Rich picture

## 2.4.2 Aktivita diagram

Aktivita diagram popisuje aktivitu přidání nemovitosti zaměstnancem.



Obrázek 3: Aktivita diagram

### 2.4.3 Architektura a struktura

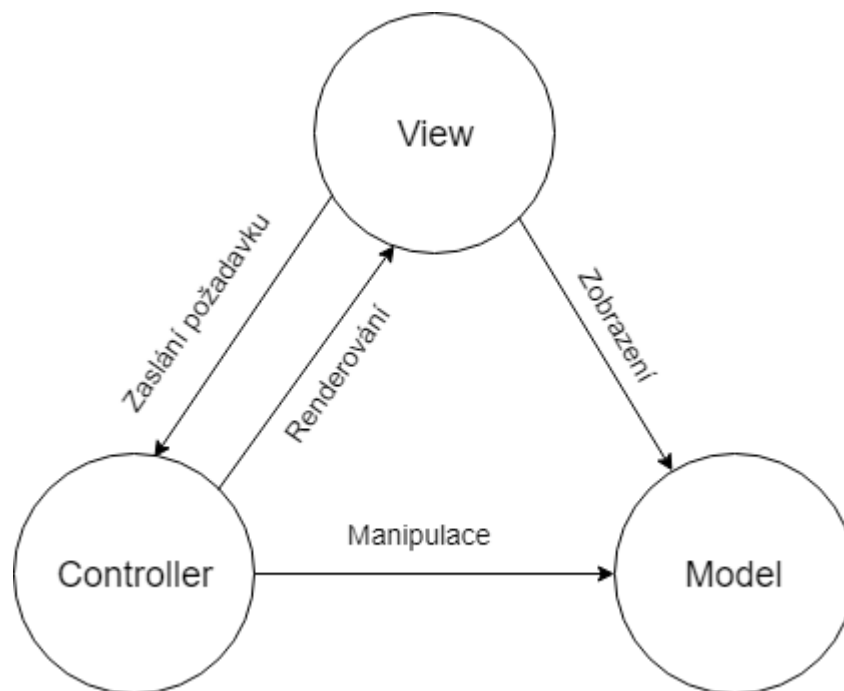
Následující kapitola obecně popisuje MVC architekturu a také způsob, jak byla aplikovaná na vyvíjenou aplikaci.

#### MVC

MVC je zkratkou pro model, view, controller. Je to architektonický vzor používaný při vývoji aplikací. Většinou se používá jeho základní myšlenka rozvržení s různými modifikacemi. MVC architektura rozděluje aplikaci na tři části:

- **Model** – Model znázorňuje data. Je možné si ho představit jako článek, který se stará o data aplikace. Získává data do modelu z databáze a opětovně ukládá jejich stav.
- **View** – View je uživatelské rozhraní. Zobrazuje data z modelu a také je dovoluje uživateli měnit.
- **Controller** – Controller zpracovává požadavky uživatele. Přípravuje potřebný view s daty získanými z modelu.

Následující obrázek představuje vztahy mezi MVC.



Obrázek 4: MVC

[4]

## Použití MVC ve vyvíjené aplikaci

Všechny modely jsou uloženy ve složce „model“. Konkrétním modelem může být například nemovitost, ta je v aplikaci definována jako třída „Nemovitost“. Tato třída v podstatě vytváří šablonu pro tabulku nemovitosti z databáze. Vytvořený objekt ve skutečnosti odpovídá řádku z tabulky. Každá modelová třída má svoji knihovni třídu, která slouží pro manipulaci s databází, tyto třídy jsou uloženy ve složce „model-controllers“, tento název je odvozen od modelových tříd s přidáním „C“ na začátek. Statické metody vně slouží pro výběr, vkládání, změnu a mazání dat.

Controllers neboli ovladače jsou uloženy ve složce „controllers“. Zpracovávají požadavky od uživatele. Příklady požadavků na nemovitosti:

- **Vložení nové** – Controller nejdříve zpracuje a otestuje vyplněná data uživatelem, následně vytvoří objekt typu „Nemovitost“ a z knihovni třídy zavolá metodu, která vkládá nemovitost do tabulky.
- **Vytvoření view nemovitosti** – Controller zpracuje data zadaná uživatelem (filtr), zavolá z knihovni třídy metodu, která mu vrátí pole objektů typu „Nemovitost“. Vracená data zpracuje do příslušného view neboli pohledu. Ten je pak odeslán uživateli.

View ve vytvářené aplikaci je možné rozdělit do tří skupin:

- **Index** – Základní stránka sloužící jako šablona, do které jsou postupně vnořovány další pohledy. Součástí této šablony je základní struktura webové stránky, menu a patička.
- **Stránky** – Každá další stránka může být chápána jako další šablona, které se dynamicky načítá ze stránky index pomocí funkce „GET“ přes parametr „page“. V podstatě je načítána šablona do šablony.
- **View přes controller** – Tyto pohledy jsou následně načítány do stránky přes výše zmíněné ovladače, které vytvářejí potřebné pohledy. V případech, kdy není nutné získávat pohledy s použitím JavaScriptu přes AJAX, mohou být pohledy uloženy přímo na daných stránkách.

Spojením výše zmíněných tří kroků vzniká výsledná webová stránka, která se zobrazuje uživateli.

## 2.5 Použité technologie, databáze, SW

### 2.5.1 Technologie

V této podkapitole jsou uvedeny technologie, které byly použity při vytváření aplikace.

- **PHP** – Vhodný pro programování dynamických webových stránek. Výhodou může být dynamické typování, škála funkcí pro komunikaci s databázemi, rozsáhlé manuály a komunita.
- **HTML** – Vytváření struktury webových stránek. Výhodou je jednoduchost, rozšířenost a dokumentace.
- **CSS** – Úprava vzhledu webových stránek. Výhodou je jednoduchost, rozšířenost a dokumentace.
- **JavaScript** – Rozšiřuje webové stránky a programování na straně klienta.
- **jQuery** – JavaScript framework. Přináší zjednodušení kódu a rozšíření JavaScriptu. Výhodou je jeho popularita a dokumentace.

[8], [6], [2]

### 2.5.2 Databáze

Pro vytváření této aplikace byl použit databázový relační systém MySQL. Je to vhodné řešení pro uložení podobné struktury dat. Zajišťuje integritu dat a vysokou bezpečnost. Dalšími výhodami jsou jeho rozšířenost a dokumentace.

[5]

### 2.5.3 SW

V této podkapitole je popsán SW, který byl použit při vytváření aplikace.

- **NetBeans** – Vývojové prostředí pro psaní kódu v PHP, HTML, CSS, JavaScript, jQuery. Vhodný pro svoji přehlednost. Velká výhoda je, že podporuje všechny nutné jazyky potřebné pro vytvářenou aplikaci.
- **phpMyAdmin** – Webový klient pro jednoduchou správu databáze. Výhodou je jeho přehlednost a rychlá správa databáze. Bez nutnosti instalace. Nevýhodou je jeho režim návrháře.
- **MySQL Workbench** – Aplikace pro návrh a správu databáze. Výhodou je možnost vytvoření přehledného ER diagramu v návrháři.

## 2.6 Databáze

Následující kapitola je věnována popisu jednotlivých tabulek a databázovému modelu v podobě ER diagramů.

### 2.6.1 Tabulky

Pro realizaci aplikace bylo vytvořeno a použito 19 tabulek. Názvy tabulek nebo sloupců tabulek jsou v níže uvedených výčtech psány bez diakritiky a kurzívou.

#### Uživatelé

Slouží pro ukládání uživatelů. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *prihlasovací\_jmeno* – Textové pole, které může nabývat hodnoty NULL a které slouží pro uložení přihlašovacího jména.
- *heslo* – Je textové pole, může nabývat hodnoty NULL a slouží pro uložení hesla.
- *jmeno* – Textové pole, které slouží pro ukládání jména.
- *prijmeni* – Je textové pole a slouží k uložení příjmení.
- *foto* – Textové pole sloužící pro ukládání odkazu na fotografii.
- *aktivni* – Je boolean záznam, ukládá, zda je uživatel v systému aktivní, pokud ne, ztrácí možnost se přihlásit.
- *opraveni* – Je textové pole, může nabývat hodnoty NULL (zákazník) a ukládá řetězec oprávnění.

Tvoří relace na tabulky:

- *pobocka* – Relace „0..N:1“. Obsahuje cizí klíč *id\_pobocka*, který tvoří odkaz na pobočku, pod kterou uživatel spadá.
- *role* – Relace „0..N:1“. Obsahuje cizí klíč *id\_role*, který tvoří odkaz na roli, jež byla uživateli přiřazena.
- *adresa* – Relace „1:1“. Obsahuje cizí klíč *id\_adresa*, který tvoří odkaz na adresu, jež patří uživateli.
- *kontakt* – Relace „1:1“. Obsahuje cizí klíč *id\_kontakt*, který tvoří odkaz na kontakt, jež patří uživateli.
- *transakce* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

#### Role

Číselníková tabulka, která slouží pro ukládání rolí. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *role* – Je textové pole a ukládá název konkrétní role.

Tvoří relace na tabulky:

- *uzivatele* – Relace „1:1“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

## **Kontakt**

Tabulka uchovávající kontaktní údaje. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *telefon* – Textové pole, které ukládá telefonní číslo.
- *e-mail* – Je textové pole a ukládá e-mail.

Tvoří relace na tabulky:

- *uzivatele* – Relace „1:1“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.
- *pobočka* – Relace „1:1“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

## **Pobočka**

Tabulka pro ukládání poboček. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *popis* – Textové pole, které může nabývat hodnoty NULL, ukládá popis pobočky.
- *foto* – Je textové pole, může nabývat hodnoty NULL a ukládá odkaz na fotografii.
- *aktivni* – Je boolean záznam, ukládá, zda je pobočka v systému aktivní, pokud ne, tak nebude zobrazena ve výpisu pro zákazníky.

Tvoří relace na tabulky:

- *adresa* – Relace „1:1“. Obsahuje cizí klíč *id\_adresa*, který tvoří odkaz na adresu, jež patří pobočce.
- *kontakt* – Relace „1:1“. Obsahuje cizí klíč *id\_kontakt*, který tvoří odkaz na kontakt, jenž patří pobočce.
- *novy\_prodej* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.
- *uzivatele* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

## **Nový prodej**

Tabulka uchovávající nově příchozí zakázky. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *jmeno* – Textové pole, které slouží pro uložení jména.
- *prijmeni* – Textové pole, jež ukládá příjmení.
- *telefon* – Je textové pole, do něhož se ukládá telefonní číslo.
- *e-mail* – Je textové pole, které ukládá e-mail.
- *popis* – Textové pole, jež ukládá popis o zakázce a nemovitosti.

Tvoří relace na tabulky:

- *pobočka* – Relace „0..N:1“. Obsahuje cizí klíč *id\_pobočka*, jenž tvoří odkaz na pobočku, na kterou byla zakázka poslána.

## **Adresa**

Tabulka uchovávající adresy. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *ulice* – Textové pole, které slouží pro uložení ulice.
- *cp* – Textové pole, jež ukládá číslo popisné.



Tvoří relace na tabulky:

- *pobocka* – Relace „1:1“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.
- *uzivatele* – Relace „1:1“. Předává svůj primární klíč, který bude sloužit jako cizí klíč.
- *nemovitost* – Relace „1:1“. Předává svůj primární klíč, který bude sloužit jako cizí klíč.
- *obec* – Relace „0..N:1“. Obsahuje cizí klíč *id\_obec*, jenž tvoří odkaz na obec, ve které se adresa nachází.

## **Obec**

Tabulka uchovávající obce. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *nazev* – Textové pole, které slouží pro uložení názvu obce.
- *kod* – Textové pole, jež ukládá kód obce.

Tvoří relace na tabulky:

- *adresa* – Relace „1:1“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.
- *okres* – Relace „0..N:1“. Obsahuje cizí klíč *okres\_id*, jenž tvoří odkaz na okres, ve kterém se obec nachází.

## **Okres**

Tabulka uchovávající okresy. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *nazev* – Textové pole, které slouží pro uložení názvu okresu.
- *kod* – Textové pole, jež ukládá kód okresu.

Tvoří relace na tabulky:

- *obec* – Relace „1:1“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.
- *kraj* – Relace „0..N:1“. Obsahuje cizí klíč *kraj\_id*, jenž tvoří odkaz na kraj, ve kterém se okres nachází.

## **Kraj**

Tabulka uchovávající kraje. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *nazev* – Textové pole, které slouží pro uložení názvu okresu.
- *kod* – Textové pole, jež ukládá kód okresu.

Tvoří relace na tabulky:

- *okres* – Relace „1:1“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

## **Nemovitost**

Tabulka uchovávající nemovitosti. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *popis* – Textové pole, které ukládá popis nemovitosti.
- *cena* – Je celé číslo, jež ukládá cenu nemovitosti.
- *poznamka\_cena* – Textové pole, které ukládá poznámku k ceně a které může nabývat hodnoty NULL.
- *datum\_vlozeni* – Časové razítko, ukládá, kdy byla nemovitost přidána.
- *aktivni* – Boolean záznam, ukládá, zda je nemovitost v systému aktivní, pokud ne, tak nebude zobrazena ve výpisu pro zákazníky.
- *provedena\_transakce* – Boolean záznam, ukládá, zda nemovitost byla prodaná, pokud ano, bude vidět pouze ve výpisu transakcí.

Tvoří relace na tabulky:

- *uzivatele* (vlastník) – Relace „0..N:1“. Obsahuje cizí klíč *id\_vlastnik*, jenž tvoří odkaz na vlastníka nemovitosti.
- *uzivatele* (makléř) – Relace „0..N:1“. Obsahuje cizí klíč *id\_makler*, jenž tvoří odkaz na makléře, který nemovitost prodává.
- *adresa* – Relace „1:1“. Obsahuje cizí klíč *id\_adresa*, který tvoří odkaz na adresu nemovitosti.
- *transakce* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.
- *fotky* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.
- *stav* – Relace „0..N:1“. Obsahuje cizí klíč *id\_stav*, jenž tvoří odkaz na stav nemovitosti.
- *nemovitost\_vlastnost* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč

### **Stav**

Číselníková tabulka, slouží pro ukládání stavu nemovitostí. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *stav* – Textové pole, které ukládá název konkrétního stavu.

Tvoří relace na tabulky:

- *uzivatele* – Relace „1:1“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

### **Transakce**

Tabulka uchovávací transakce, jedná se o propojovací tabulku, která zařídí „M:N“ relaci mezi tabulkami *nemovitost* a *uzivatele*. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *datum* – Časové razítko, ukládá, kdy vznikla transakce.

Tvoří relace na tabulky:

- *uzivatele* – Relace „0..N:1“. Obsahuje cizí klíč *id\_zajemce*, jenž tvoří odkaz na zájemce o nemovitost.
- *nemovitost* – Relace „0..N:1“. Obsahuje cizí klíč *id\_nemovitost*, jenž tvoří odkaz na nemovitost.

### **Typ nemovitost**

Číselníková tabulka uchovávací typy nemovitostí. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *typ\_nemovitost* – Textové pole, které slouží pro uložení názvu typu.

Tvoří relace na tabulky:

- *nemovitost* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.
- *vlastnost* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

## Vlastnost

Tabulka uchovávající vlastnosti. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *vlastnost* – Textové pole, které slouží pro uložení názvu vlastnosti.
- *zobrazeno* – Boolean záznam, ukládá, zda se daná vlastnost bude zobrazovat v hlavním filtru daného typu nemovitosti.

Tvoří relace na tabulky:

- *typ\_nemovitost* – Relace „0..N:1“. Obsahuje cizí klíč *id\_typ\_nemovitost*, jenž tvoří odkaz na typ nemovitosti, pod který vlastnost spadá.
- *vlastnosti* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

## Vlastnosti

Tabulka uchovávající klíčová slova k vlastnostem. Obsahuje položky:

- *id* – Primární klíč, celé číslo, slouží pro jedinečnou identifikaci.
- *text* – Textové pole, uložení názvu klíčového slova.

Tvoří relace na tabulky:

- *vlastnost* – Relace „0..N:1“. Obsahuje cizí klíč *id\_vlastnost*, jenž tvoří odkaz na vlastnost, pod kterou toto klíčové slovo spadá.
- *nemovitost\_vlastnost* – Relace „1:0..N“. Předává svůj primární klíč, který poté bude sloužit jako cizí klíč.

## Nemovitost vlastnost

Jedná se o propojovací tabulku, která zařídí „M:N“ relaci mezi tabulkami *nemovitost* a *vlastnosti*. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *popis* – Textové pole, slouží pro uložení popisu konkrétního klíčového slova vlastnosti k dané nemovitosti.

Tvoří relace na tabulky:

- *vlastnosti* – Relace „0..N:1“. Obsahuje cizí klíč *id\_vlastnost*, jenž tvoří odkaz na klíčové slovo.
- *nemovitost* – Relace „0..N:1“. Obsahuje cizí klíč *id\_nemovitost*, jenž tvoří odkaz na nemovitost.

## Uložené filtry

Tabulka uchovávající uložené filtry nemovitostí. Záznamy v této tabulce jsou uloženy na 90 dní. Netvoří žádné relace s jinými tabulkami. Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *id\_url* – Textové pole, jež ukládá pseudonáhodně generovaný řetězec, který slouží jako odkaz adresa na daný filtr.
- *select\_string* – Je textové pole, které ukládá dotaz se zadanými filtry.
- *e-mail* – Textové pole, které ukládá e-mail zákazníka a může nabývat hodnoty NULL.
- *zasilani\_e-mailu* – Boolean záznam, ukládá, zda se zákazníkovi budou odesílat nově přidané nemovitosti, které odpovídají uloženému dotazu.
- *vytvoreno* – Časové razítko, jež ukládá, kdy byl řádek vytvořen.

## **Oprávnění**

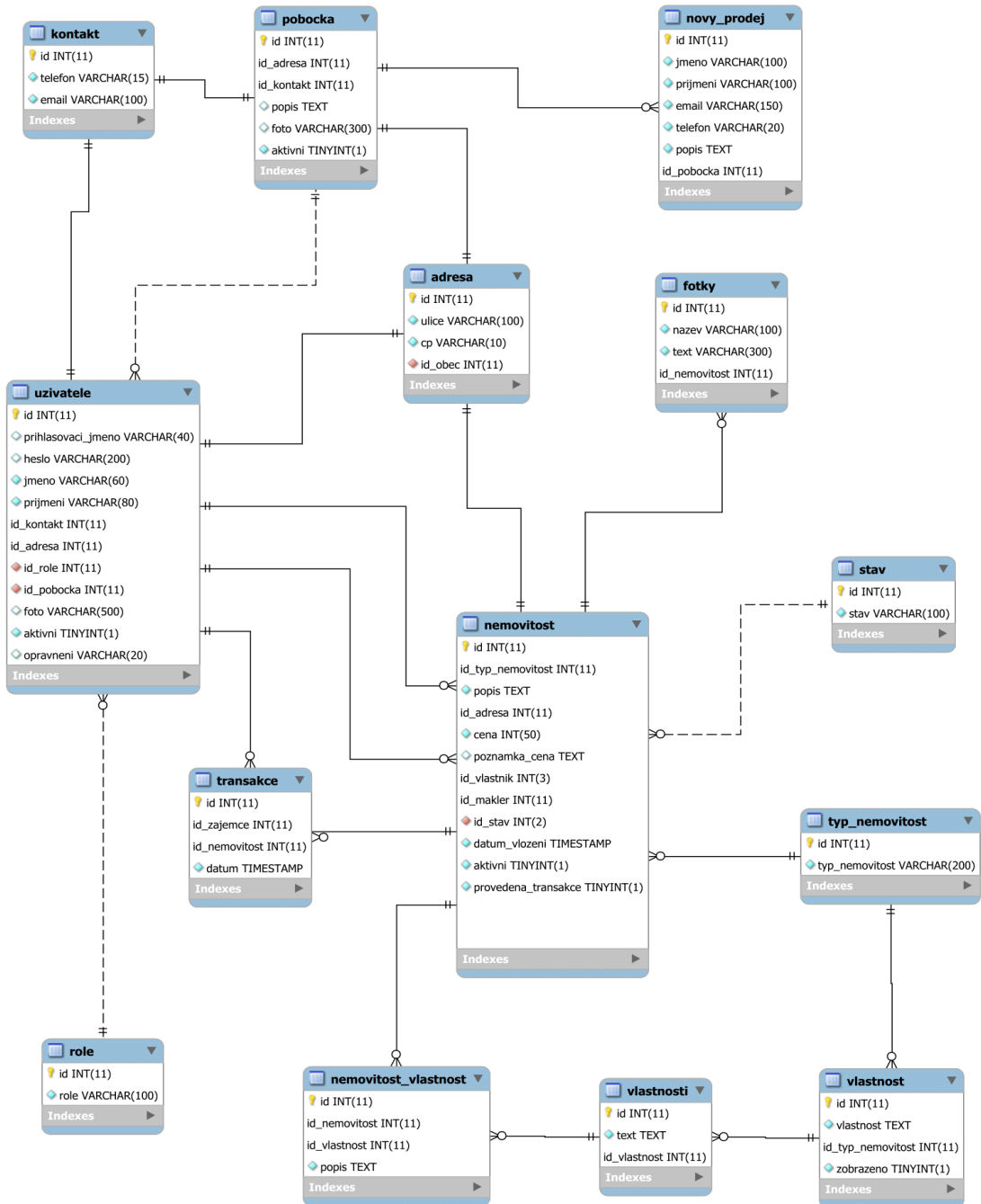
Tabulka, která identifikuje jednotlivá oprávnění. Netvoří žádné relace s jinými tabulkami.

Obsahuje položky:

- *id* – Je primární klíč, celé číslo a slouží pro jedinečnou identifikaci.
- *pozice* – Celé číslo, slouží pro uložení pozice. Označuje význam jednotlivých čísel v řetězci oprávnění, která mají jednotliví zaměstnanci.
- *nazev* – Textové pole, které ukládá název daného oprávnění.

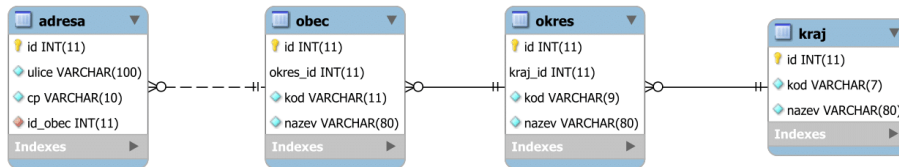
## 2.6.2 ER diagramy

První diagram obsahuje většinu tabulek, pro větší přehlednost byly některé tabulky rozděleny do více obrázků.



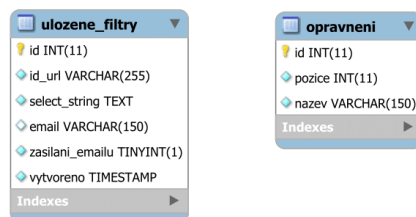
Obrázek 5: ER diagram

Druhý diagram znázorňuje další vztah tabulky adresa.



Obrázek 6: ER diagram 2

Třetí diagram obsahuje tabulky, které nemají žádnou vazbu.



Obrázek 7: ER diagram 3

## 2.7 Ukázka zdrojových kódů

V následující kapitole jsou zobrazeny a popsány ukázky zdrojových kódů.

### Validace dat

```
function zvaliduj($text) {  
    $text = trim($text);  
    $text = stripslashes($text);  
    $text = htmlspecialchars($text);  
    return $text;  
}
```

Ukázka kódu 1: Validace

Při vkládání dat uživatelem do databáze je vhodné zadané hodnoty prověřit následujícím kódem, který zajišťuje ochranu proti nechtěným vstupům. Nejdříve funkce *trim* zajistí odstranění přebytečných znaků, funkce *strslashes* odstraní znak „\“ a nakonec funkce *htmlspecialchars* zajistí, že pokud by například vstup byl spustitelný skript, stane se z něho bezcenný text.

### Připojení k databázi

```
function get_pdo($type, $servername, $user, $pass, $database) {
    switch ($type) {
        case 'mysql':
            $dsn = "{$type}:dbname={$database};host={$servername}";
            break;
    }
    return new PDO($dsn, $user, $pass);
}
```

Ukázka kódu 2: Vytvoření PDO

Pro zajištění spojení PHP s databází je využívána třída PDO. Tento objekt vytváří třída *get\_pdo*, která přejímá pět parametrů. Typ databázového systému, adresu serveru, přihlašovací jméno, heslo a připojovanou databázi. Při vytváření PDO objektu konstruktor třídy požaduje připojovací řetězec *dsn* a dále má volitelné parametry jméno a heslo.

### Vkládání dat do databáze, ochrana proti SQL injection

```
public static function insertAdresa(Adresa $adresa) {
    $db = DbInfo::getPDOConnection();
    $datainsert =
[NULL, $adresa->getUlice(), $adresa->getCp(), $adresa->getObec()->getId()];
    $insert =
$db->prepare("INSERT INTO `adresa`(`id`, `ulice`, `cp`, `id_obec`) VALUES (?,?,?,?)");
    $insert->execute($datainsert);
    $select = $db->prepare("SELECT LAST_INSERT_ID()");
    $select->execute();
    $row = $select->fetch();
    return $row["LAST_INSERT_ID()"];
}
```

Ukázka kódu 3: Vložení adresy

Výše uvedený kód představuje metodu, která vkládá adresu do databáze. Nejdříve je vytvořeno spojení s databází přes PDO pomocí metody *getPDOConnection*. Následně jsou data z objektu typu *adresa* vložena do pole. Dále se připraví SQL příkaz pro vložení nové adresy pomocí příkazu *prepare*. Zde je vytvořena ochrana proti SQL injection tím, že místo přímého vkládání hodnot jsou využity zástupné znaky v podobě otazníků. Pomocí funkce *execute*, jsou pak jednotlivé hodnoty z vytvořeného pole navázány na jednotlivé otazníky.

Funkce *insertAdresa* následně vrací id, které se při každém vložení generuje databází, pakliže by vložení adresy proběhlo neúspěšně, vrací -1. Zjištění hodnoty id zajistí SQL dotaz "SELECT LAST\_INSERT\_ID()".

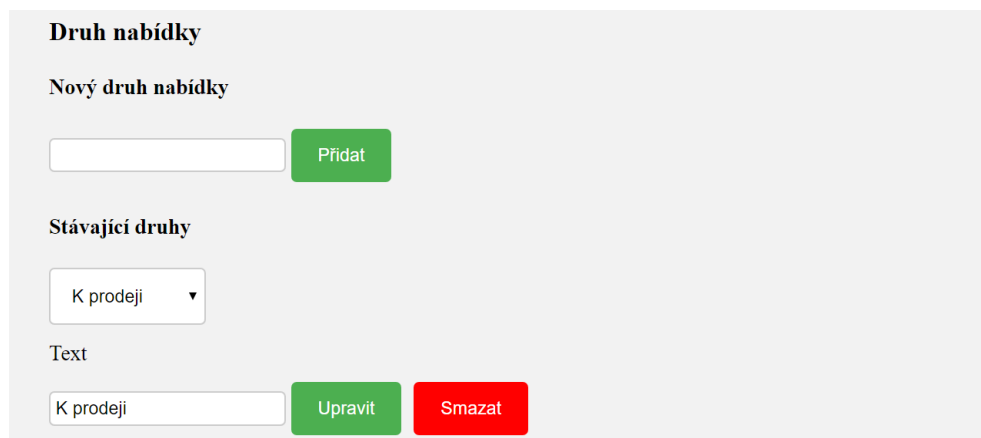
## 2.8 Uživatelská příručka

V následující kapitole jsou popsány vybrané části, které zásadním způsobem mění a ovládají celý systém aplikace.

### 2.8.1 Práce s vlastnostmi, tvoření filtrů

V této části je podrobně vysvětleno chování filtrů, jakožto vlastností nemovitostí.

#### Druh nabídky



**Druh nabídky**

Nový druh nabídky

Stávající druhy

▼

Text

Obrázek 8: Druh nabídky

Druh nabídky zobrazuje stavy nemovitostí. Je napojen na tabulku *stav* a nabývá například hodnot „K prodeji“ a „K pronájmu“. Data z databáze jsou následně zobrazena v hlavní části filtru nemovitostí.

V první části formuláře je možnost přidat nový druh vyplněním náležitího inputu a přidáním přes tlačítko „přidat“. Ve zbytku je následně správa stávajících druhů. V selektoru je vybrána položka, která bude zpracovávána. Její text se zobrazí v input boxu a lze změnit přes tlačítko



„upravit“. Jednotlivý druh je možno také smazat přes tlačítko „smazat“. Provádění změn na již zavedené druhy v systému se velmi nedoporučuje.

### Typ nemovitosti

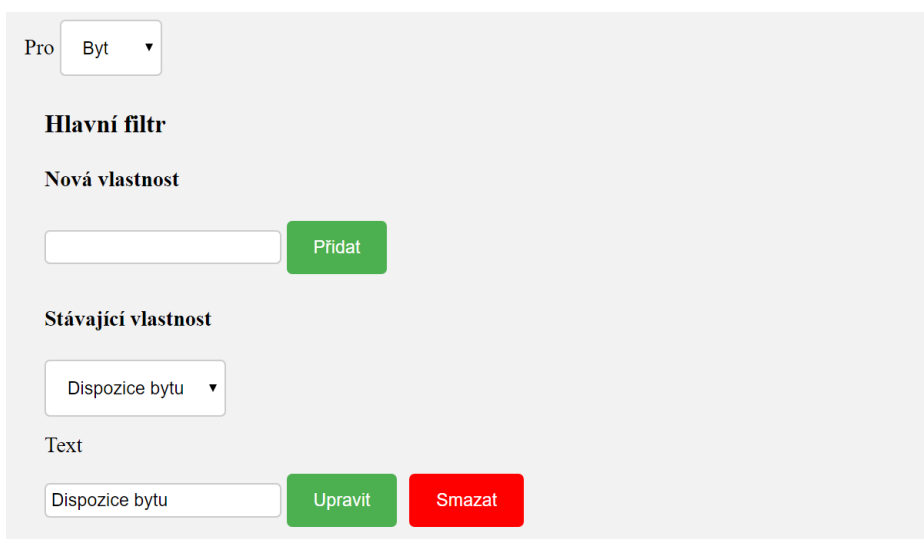
Typ nemovitosti zobrazuje typy nemovitostí. Je napojen na tabulku *typ\_nemovitosti* a nabývá například hodnot „Dům“ a „Byt“. Data z databáze jsou následně zobrazena v hlavní části filtru nemovitostí.

Funkčnost a vzhled tohoto formuláře jsou stejné jako u výše zmíněného druhu nabídky, umožněno je přidání nového typu, editace a mazání stávajícího. Provádění změn na již zavedené typy v systému se velmi nedoporučuje.

### Vlastnosti typů nemovitostí

Tyto vlastnosti se rozdělují na ty, které spadají do hlavního filtru a ty, které jsou pouze v rozšiřujícím filtru.

### Hlavní filtr



Obrázek 9: Hlavní filtr

Hlavní filtr zobrazuje vybrané vlastnosti typů nemovitostí. Je napojen na tabulku *vlastnost* a nabývá například hodnot „Dispozice bytu“, každý řádek z tabulky pak spadá pod nějaký typ nemovitosti. Data z databáze jsou následně zobrazena v hlavním filtru nemovitostí.

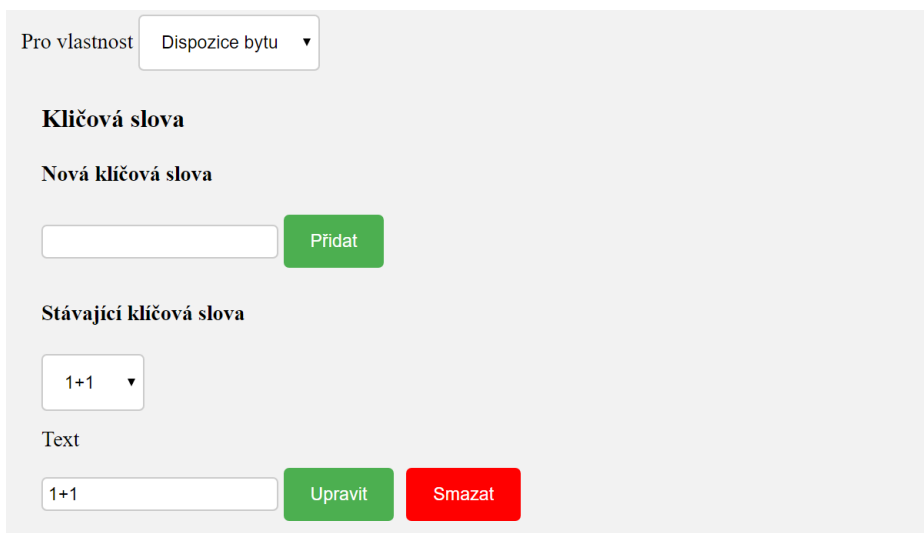
V první části formuláře je selektor pro vybrání, se kterým typem nemovitosti se bude pracovat. Funkčnost tohoto formuláře je jinak stejná jako u výše zmíněného druhu nabídky, umožněno je přidání nové vlastnosti, editace a mazání stávající. Provádění změn na zavedené vlastnosti v systému se velmi nedoporučuje.

## Rozšiřující filtr

Funkčnost je totožná jako u výše zmíněného hlavního filtru. Pro výběr typu nemovitosti slouží stejný selektor.

### Klíčová slova

Každé klíčové slovo je napojeno na jednotlivou vlastnost a nějakým způsobem ji konkrétně označuje.



Obrázek 10: Klíčová slova

Klíčová slova jsou pak ve filtru zobrazena ke své vlastnosti. Jsou napojena na tabulku *vlastnosti* a nabývají například hodnot „1+kk“, „1+1“, které spadají pod vlastnost „Dispozice bytu“.

Selektor s vlastnostmi je napojen na selektor typů nemovitostí, který je umístěn nad hlavním filtrem. Možnosti naplnění klíčových slov:

- ano – Použití v případě, kdy vlastnost dostatečně definuje svůj význam a není třeba ji rozdělovat do více skupin (vlastnost obsahuje pouze jedno klíčové slovo). Například vlastnost balkón, kde vlastnost již jednoznačně určuje, že daná nemovitost má balkón. Ve filtru se pak zobrazí pouze vlastnost se zaškrtačím políčkem.
- rozsah – Použití v případě, pokud se vlastnost bude filtrovat pomocí číselných hodnot (vlastnost obsahuje pouze jedno klíčové slovo). Například vlastnost velikost bytu. Ve filtru se pak zobrazí inputy od a do.
- výčet – Použití v případě, kdy vlastnost dostatečně nedefinuje svůj význam a je třeba ji rozdělovat do více skupin (vlastnost více klíčových slov). Například vlastnost dispozice bytu, kde hodnoty budou „1+kk“, „1+1“ apod. Ve filtru se pak zobrazí vlastnost a pod ní zaškrtačící políčka s klíčovými hodnotami.

Funkčnost tohoto formuláře je jinak stejná jako u výše zmíněného druhu nabídky, umožněno je přidání nového klíčového slova, editace a mazání stávajícího. Provádění změn na již zavedená klíčová slova v systému se velmi nedoporučuje.

## 2.8.2 Změna stavu aktivity

V této části jsou podrobně vysvětleny následky změn aktivity.

- **Pobočka** – Při deaktivování pobočky se automaticky deaktivují všichni uživatelé, kteří spadají pod pobočku a také jejich nemovitosti.
- **Zaměstnanec** – Deaktivuje se zaměstnanec a s ním se automaticky deaktivují všechny nemovitosti, které mu náleží. Daný zaměstnanec ztrácí možnost přihlášení. Neaktivní makléř se přestane vypisovat na stránce makléři.
- **Zákazník** – Deaktivuje se zákazník a s ním se automaticky deaktivují všechny nemovitosti, které mu náleží. Danému zákazníkovi nelze přidat nová nemovitost.
- **Nemovitost** – Deaktivuje se pouze nemovitost. Není viditelná pro zákazníky.

## 2.8.3 Přidat nemovitost – vyplnění vlastností

V této části je podrobně vysvětleno vyplnění vlastností při přidávání nebo úpravě nemovitostí.

**Hlavní filtr**  
☰ Přidat novou vlasnost  
Cena:  Poznámka k ceně:   
 Dispozice bytu: 1+1 ▼ Popis:  ☰ Přidat klíčové slovo  
 Velikost bytu: rozsah ▼ Popis:  ☰ Přidat klíčové slovo

**Rozšířený filtr**  
☰ Přidat novou vlasnost  
 Konstrukce: Cihla ▼ Popis:  ☰ Přidat klíčové slovo  
 Vlastnictví: Družstevní ▼ Popis:  ☰ Přidat klíčové slovo  
 Výtah: Ano ▼ Popis:  ☰ Přidat klíčové slovo

Obrázek 11: Přidat nemovitost – vlastnosti

Možnosti zobrazené na obrázku se objeví až po zvolení typu nemovitosti. Zaškrtnutá tlačítka slouží pro přidání dané položky k nemovitosti, která zahrnuje vybranou vlastnost, klíčové slovo a nepovinný popis.

Při vyplnění rozsahu je pro správnou funkčnost nezbytné v popisu zaprvé uvést hodnotu, která se bude později filtrovat, například „80 m<sup>2</sup>“.

## **2.9 Instalační příručka**

Z příloženého CD je nejprve nutné nahrát soubory ze složky „realitnikancelar“ na server. V dalším kroku je potřeba vytvořit novou databázi a importovat soubor s názvem „db.sql“ ze složky „sql“. Poté je nutné přepsat konfigurační soubor databáze s názvem „dbinfo.php“, který se nachází ve složce „config“. Přístup do administrace jako admin je umožněn pomocí přihlašovacího jména „admin“ a hesla „test“.

## ZÁVĚR

V rámci této bakalářské práce byla vytvořena funkční webová aplikace pro systém realitní kanceláře, která splňuje všechny požadavky, jež jsou uvedeny v zadání.

Aplikace byla vypracována jako obecný systém pro realitní kanceláře, a tudíž obsahuje důležité dynamické prvky, které by mohly být potenciálním zákazníkem požadovány. Tyto prvky je možné spravovat v administraci, takže zákazník si velkou část výsledné aplikace může navrhnout a udělat podle sebe bez zásahu vývojáře. Všechny statické prvky mohou být dodělány na míru.

Náprava nedostatků, které byly u stávajících systémů nalezeny, a následná snaha o jejich vylepšení byly provedeny úspěšně. Web je validní. Vylepšení bylo zaměřeno především na filtr nemovitostí. Výsledkem je přehledný filtr, který je rozdělený na základní a rozšiřující část.

Bakalářská práce přináší v první řadě vytvoření kvalitního návrhu, databáze a backendu aplikace. Její přínos lze rovněž spatřovat v tom, že základ tohoto systému by následně mohl sloužit jiným, podobně zaměřeným. Za hodnotné lze také považovat aplikování dostatečné ochrany proti SQL injection. Slabinou může být osobní nedostatek citu autora pro design webových stránek, který se může projevit absencí profesionálního vzhledu frontendu.

## POUŽITÁ LITERATURA

- [1] *EVROPA realitní kancelář* [online]. c2016 [cit. 2019-03-05]. Dostupné z:  
<https://www.rkevropa.cz/>
- [2] *jQuery* [online]. c2019 [cit. 2019-03-18]. Dostupné z: <https://jquery.com/>
- [3] *M&M reality* [online]. [2016] [cit. 2019-03-05]. Dostupné z:  
<http://www.mmreality.cz/cs/>
- [4] MVC Architecture. *TutorialsTeacher* [online]. c2019 [cit. 2019-03-25]. Dostupné z:  
<https://www.tutorialsteacher.com/mvc/mvc-architecture>
- [5] *MySQL* [online]. c2019 [cit. 2019-03-18]. Dostupné z: <https://www.mysql.com/>
- [6] *PHP* [online]. c2001-2019 [cit. 2019-03-18]. Dostupné z: <http://php.net/>
- [7] *Realitní kanceláře RE/MAX* [online]. [2019] [cit. 2019-03-05]. Dostupné z:  
<https://www.remax-czech.cz/>
- [8] *W3Schools Online Web Tutorials* [online]. [2018] [cit. 2018-10-09]. Dostupné z:  
<https://www.w3schools.com/>

## **PŘÍLOHY**

Příloha A – Instalační CD.....	56
--------------------------------	----

## **PŘÍLOHA A – INSTALAČNÍ CD**

Instalační CD obsahuje zdrojové soubory aplikace a SQL soubor pro vytvoření databáze.