

**STŘEDNÍ PRŮMYSLOVÁ ŠKOLA BRNO,
PURKYŇOVA, PŘÍSPĚVKOVÁ ORGANIZACE**



WEB PRO AUTOBAZAR

Antonín Mrkvica

V4B

**PROFILOVÁ ČÁST MATURITNÍ ZKOUŠKY
MATURITNÍ PRÁCE**

BRNO 2019

Prohlášení

Prohlašuji, že jsem maturitní práci na téma Web pro autobazar vypracoval samostatně a použil jen zdroje uvedené v seznamu literatury.

Prohlašuji, že:

- Beru na vědomí, že zpráva o řešení maturitní práce a základní dokumentace k aplikaci bude uložena v elektronické podobě na intranetu Střední průmyslové školy Brno, Purkyňova, příspěvkové organizace.
- Beru na vědomí, že bude má maturitní práce včetně zdrojových kódů uložena v knihovně SPŠ Brno, Purkyňova, p. o., dostupná k prezenčnímu nahlédnutí. Škola zajistí, že nebude pro nikoho možné pořizovat kopie jakékoliv části práce.
- Beru na vědomí, že SPŠ Brno, Purkyňova, p. o. má právo celou moji práci použít k výukovým účelům a po mém souhlasu nevýdělečně moji práci užít ke své vnitřní potřebě.
- Beru na vědomí, že pokud je součástí mojí práce jakýkoliv softwarový produkt, považují se za součást práce i zdrojové kódy, které jsou předmětem maturitní práce, případně soubory, ze kterých se práce skládá. Součástí práce není cizí ani vlastní software, který je pouze využíván za přesně definovaných podmínek, a není podstatou maturitní práce.

Antonín Mrkvica

Dolní 340

691 63 Velké Němčice

V Brně dne 16. 4. 2019

.....

Vedoucí práce: RNDr. Lenka Hrušková

Anotace

Maturitní práce se zabývá vytvořením webové aplikace pro Autobazar. Aplikace je napsána v jazyce Python ve webovém frameworku Django. Design je řešen pomocí MaterializeCSS. Web umožňuje vystavování inzerátů s automobily, hodnocení prodeje automobilů prostřednictvím komentářů a správu (editaci a mazání) jednotlivých inzerátů, uživatelů a komentářů pomocí superuživatele s právy administrátora.

Obsah

Prohlášení	ii
Anotace	iv
Obsah.....	1
Seznam použitých zkratk	3
1 Teoretický úvod.....	4
2 Rozbor řešení	4
2.1 Use Case	4
2.2 Návrh databáze	6
2.3 Vývoj aplikace	6
2.4 Možnosti rozšíření aplikace	6
2.5 Design.....	6
3 Řešení.....	7
3.1 Django	7
3.2 Programovací jazyky	7
3.3 Databáze.....	7
3.4 Backend	8
3.5 Frontend.....	9
3.6 JQuery.....	11
3.7 Ajax	11
3.8 Verzovací nástroj GIT.....	11
4 Uživatelská příručka.....	12
4.1 První pohled	12
4.2 Filtrování vozidel – vyhledávání inzerátů	12
4.3 Prohlížení inzerátu, koupě vozidla	13
4.4 Registrace a Přihlášení	14
4.5 Prodávání automobilu	14
4.6 Mazání a Správa inzerátu – editace	15
4.7 Přidávání komentářů a profil uživatele	15
4.8 Správa uživatelů – administrátor	15

5	Závěr	17
6	Seznam obrázků	18
7	Seznam použitých zdrojů	19

Seznam použitých zkratk

HTML – HyperText Markup Language

CSS – Cascading Style Sheets

SQL – Structured Query Language

Django – Webový framework

MaterializeCSS – Frontend framework

jQuery – JavaScriptovská knihovna

JSON – JavaScript Object Notation

Ajax – Asynchronous JavaScript and XML

GIT – Distribuovaný systém správy verzí

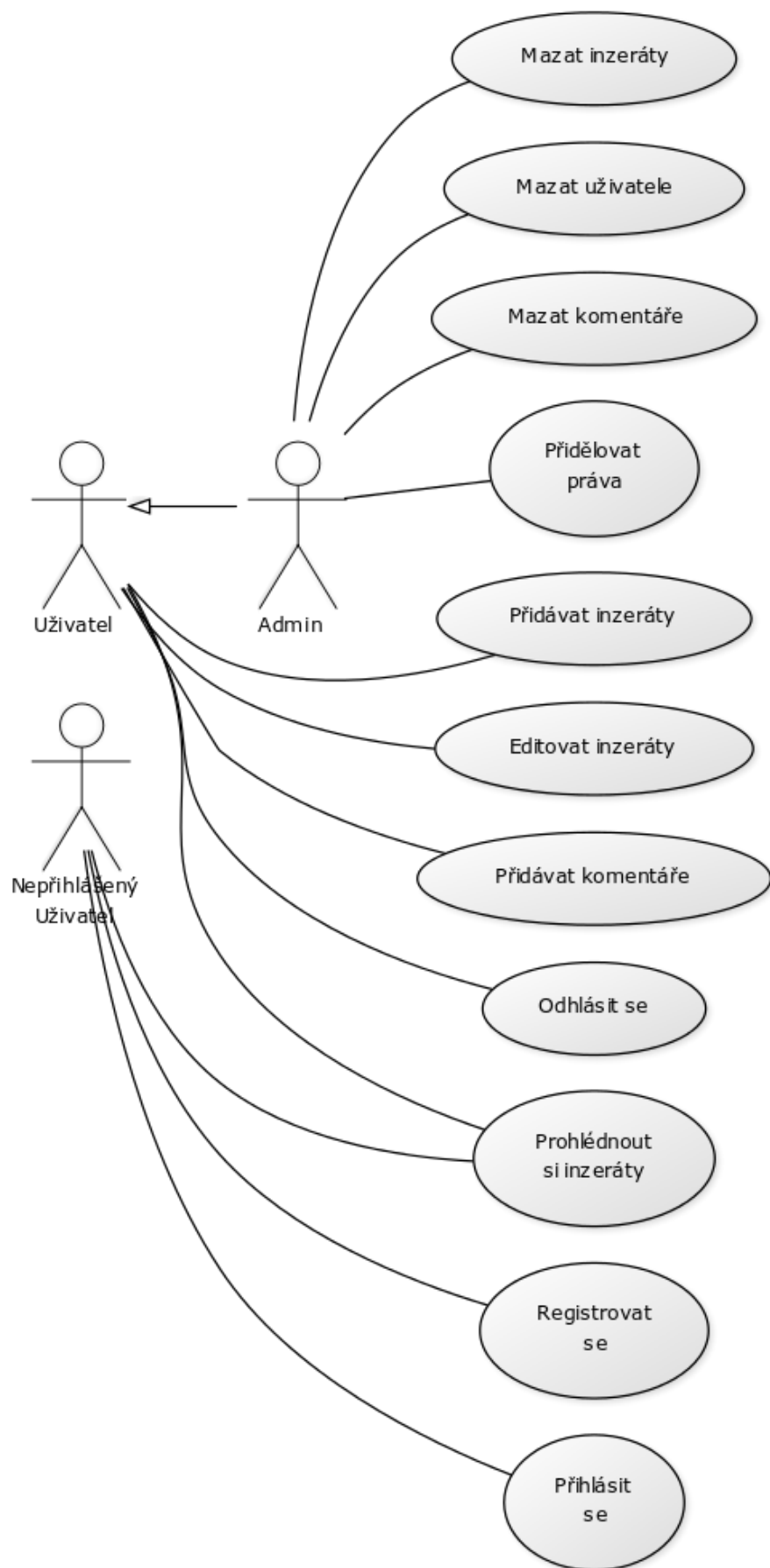
1 Teoretický úvod

Web pro Autobazar je webová aplikace, která umožňuje registrovaným uživatelům vystavovat inzeráty, které jsou formované pro prodej automobilů. Uživatel si může jednotlivé inzeráty prohlédnout a případně poslat majiteli zprávu, že má o dané vozidlo zájem.

2 Rozbor řešení

2.1 Use Case

Pro jednotlivé typy uživatelů jsem vytvořil diagram případů užití (USE CASE), které nutí člověka zamyslet se nad danou problematikou konkrétně možnosti uživatelů. Tento diagram mi pomohl při vytváření webové aplikace z hlediska práv uživatelů, ze kterého je hned jasné, který typ uživatele má jaká práva.



2.2 Návrh databáze

Databáze se skládá ze tří tabulek (Uživatel, Auto, Komentář). Auto má atribut majitel (ten, kdo inzerát vytvořil) a Komentář má atribut majitel (ten, kdo komentář vytvořil) a příjemce (ten, komu je komentář určen). Na profilu příjemce je poté komentář umístěn.

2.3 Vývoj aplikace

Na začátku jsem neměl jasnou představu, jak má aplikace vypadat, tak jsem několikrát celou aplikaci předělával. Postupně jsem projekt ladil, až do konečného stádia. A i v této fázi lze ještě projekt upravit a přidat další funkce.

2.4 Možnosti rozšíření aplikace

I když je aplikace již hotová, tak lze přidávat další vymoženosti například rozšířit možnost komentovat a odpovídat na komentář, vytvořit databázi již prodaných vozidel nebo přidat fórum pro automobilové nadšence.

2.5 Design

Tato maturitní práce je zaměřena spíše na funkčnosti celé webové aplikace a spolupráci mezi Django a HTML. MaterializeCSS jsem volil, protože nemám velké zkušenosti s grafikou ani se styly CSS. Design by šel zajisté předělat, aby celkový vzhled stránky vypadal lépe, ale k tomu bych potřeboval spolupráci grafika.

3 Řešení

3.1 Django

Django je webový framework, který ulehčuje a usnadňuje práci při vytváření projektu. Je určen pro rychlé zpracování celé problematiky. Django funguje na modelu MVC – Model View Controller. Využívá a propojuje soustavu souborů nejdůležitější z nich jsou:

- Manage.py – Spouštění server, vytváření databázi
- Settings.py – Nastavení serveru
- Models.py – Definování databáze
- Urls.py – Propojení url s view
- View.py – Zpracování dat, vytváření view

3.2 Programovací jazyky

Pro Backend celé webové aplikace jsem si vybral programovací jazyk Python a jeho webový framework Django, protože jsem s ním měl již zkušenosti. Tento framework velice usnadňuje a tím urychluje i celý vývoj aplikace. V základu pracuje s databází SQLite, takže není nutné žádné propojení s další databází. Zajímavé je právě modelování a přístup k databázi SQLite, protože Django má své metody jak pro vytváření databáze, tak i k získávání dat z ní. Pro Frontend aplikace jsem volil HTML, CSS, JavaScript a JQuery společně s frameworkem MaterializeCSS, se kterým se snadno a rychle tvoří konstrukce webové stránky. Tento framework pomáhá nejen zkušeným odborníkům, ale i jako nám, nezkušeným programátorům, kteří nemají úplný cit pro design.

3.3 Databáze

Jednotlivé tabulky a jejich atributy databáze jsou definovány v souboru „models.py“

```
class User(models.Model):
    is_admin = models.BooleanField(default=False)
    username = models.CharField(max_length=50)
    password = models.CharField(max_length=50)
    phone = models.CharField(max_length=50)
    email = models.CharField(max_length=50)
```

Obrázek 2: Definovaná tabulka pro uživatele

Tabulka uživatele se skládá z uživatelského jména, hesla, čísla telefonu, emailové adresy a práv pro administrátora stránky (viz. Obrázek 1 Definovaná tabulka pro uživatele). Řádky představují atributy tabulky.

```

class Car(models.Model):
    mark = models.CharField(max_length=50)
    model = models.CharField(max_length=50)
    images_paths = models.CharField(max_length=50, default=json.dumps([]))
    owner = models.ForeignKey(User, on_delete=models.CASCADE, null=True, related_name='owner')
    manufacture_date = models.CharField(max_length=50)
    kilometres = models.CharField(max_length=50)
    price = models.CharField(max_length=50)
    turbo = models.CharField(max_length=50)
    engine_capacity = models.CharField(max_length=50)
    performance_kw = models.CharField(max_length=50)
    performance_hp = models.CharField(max_length=50)
    add_date = models.DateField(max_length=50)
    fuel_type = models.CharField(max_length=50)
    repair = models.CharField(max_length=500)
    defects = models.CharField(max_length=500)
    description = models.CharField(max_length=500)
    buyer = models.ForeignKey(User, on_delete=models.CASCADE, null=True, related_name='buyer')

```

Obrázek 3: Tabulka Automobilu

Taktéž je zde definována tabulka pro automobil (viz Obrázek 2 Tabulka Automobilu).

```

class Comment(models.Model):
    author = models.ForeignKey(User, on_delete=models.CASCADE, null=True, related_name='author')
    receiver = models.ForeignKey(User, on_delete=models.CASCADE, null=True, related_name='reciever')
    text = models.CharField(max_length=50)
    date = models.CharField(max_length=50)

```

Obrázek 4: Tabulka Komentáře

A tabulku pro komentář (viz. Obrázek 3 Tabulka Komentáře).

3.4 Backend

Backend webové aplikace závisí na souboru „view.py“ a jeho propojení s „urls.py“. Ve view.py se nachází nejen odpovědi na dotazy webového prohlížeče, které vrací HTML stránku s přesně definovaným obsahem, ale také spojení s databází a práci s proměnnými, které obsahují data, jež daný dotaz požaduje nebo poskytuje.

```

path('', views.index, name='index'),
path('login', views.login, name='login'),
path('reg', views.reg, name='reg'),
path('logout', views.logout, name='logout'),
path('sell', views.sell, name='sell'),
path('search', views.search, name='search'),
path('viewmodels', views.viewmodels, name='viewmodels'),
path('view', views.view, name='view'),
path('edit_car', views.edit_car, name='edit_car'),
path('remove_image', views.remove_image, name='remove_image'),
path('delete_car', views.delete_car, name='delete_car'),
path('send_password', views.send_password, name='send_password'),
path('change_password', views.change_password, name='change_password'),
path('acc', views.acc, name='acc'),
path('profile', views.profile, name='profile'),
path('buy_car', views.buy_car, name='buy_car'),
path('add_comment', views.add_comment, name='add_comment'),
path('remove_comment', views.remove_comment, name='remove_comment'),
path('settings', views.settings, name='settings'),
path('remove_user', views.remove_user, name='remove_user'),

```

Obrázek 5: urls.py

V „urls.py“ je definováno, na které adrese se má provést dané view.

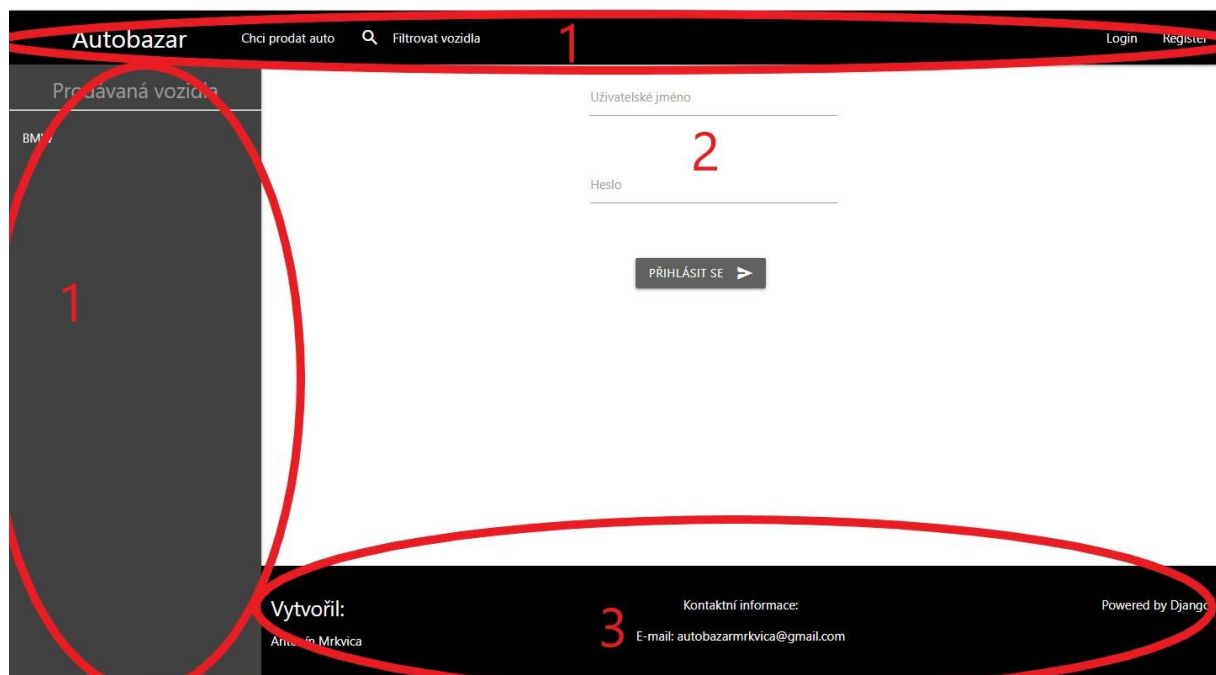
Path(„cesta v url adrese“, view.[konkretní název view ve view.py], name=„název“)

```
def login(request):
    if request.method == "POST":
        username = request.POST.get('username')
        passwd = request.POST.get('passwd')
        if username == "" or passwd == "":
            return render(request, "./login.html", {"msg": "Something is missing."})
        searched_user = User.objects.filter(username=username, password=passwd)
        if searched_user:
            request.session['user_id'] = searched_user[0].id
        else:
            return render(request, "./login.html",
                          {"user": current_user(request.session.get('user_id')), "car_list": sorted_dict,
                           "msg": "Uživatelské jméno neexistuje nebo bylo zadáno špatné heslo."})
        return redirect("index")
    else:
        return render(request, "./login.html",
                      {"user": current_user(request.session.get('user_id')), "car_list": sorted_dict})
```

Obrázek 6: Přihlášení

Po vytvoření dotazu na přihlášení se spustí script, který se vyhledá v „urls.py“ pod adresou „/login“, zde je připojeno view „login“. Nejprve se vygeneruje odpověď s HTML stránkou, která obsahuje formulář pro přihlášení. Až poté, co je formulář přihlášení potvrzen, se spustí druhá část kódu, která ověří údaje, a pokud vše souhlasí, uživatele přihlásí do systému pomocí cookies.

3.5 Frontend



Obrázek 7: Stránka pro přihlášení

Frontend je víceméně tvořen z prvků MaterializeCSS vložených do HTML stránek. Každá stránka je složena ze tří prvků (souborů). (Viz. Obrázek 6 Stránka pro přihlášení) První část (1) je Navigátor (horní i boční), ve druhé části (2) se nachází

samotný obsah stránky (přihlášení, registrace, inzeráty vozidel, ...) a nakonec ve třetí části (3) se nachází patička stránky (footer).

```
{% include "header_nav.html" %}
<main>
  <div class="row center">
    <form method="post">
      {% csrf_token %}
      <div class="col s12 m8 offset-m4 center-align">
        <div class="row">
          <div class="input-field col m5 s8 offset-s2 center">
            <input id="username" type="text" class="validate" name="username">
            <label class="active" for="username">Uživatelské jméno</label>
          </div>
        </div>
        <div class="row">
          <div class="input-field col m5 s8 offset-s2">
            <input id="passwd" type="password" class="validate" name="passwd">
            <label class="active" for="passwd">Heslo</label>
          </div>
        </div>
        <br>
      </div>
      <div class="col m3 offset-m4 center-align s8 offset-s2">
        <button class="btn waves-effect waves-light grey darken-2" type="submit" name="action">Přihlásit se
          <i class="material-icons right">send</i>
        </button>
      </div>
    </form>
    {% if msg %}
    <div class="row col s5 offset-s3">
      <p style="color: red">{{ msg }}</p>
      <a href="send_password">Chci poslat heslo na email.</a>
    </div>
    {% endif %}
  </div>
</main>

{% include "footer.html" %}
```

Obrázek 8: HTML stránka pro Přihlášení (zdrojový kód)

HTML stránka je tvořena tabulkou. Číslo s písmenem za „col“ (m8, s8, m5) označuje šířku sloupce, kdy řádek je tvořen z dvanácti hodnot. „Offset“ je posunutí doprava o určitý počet hodnot daného sloupce. Viz. Obrázek 7 HTML stránka pro Přihlášení (Zdrojový kód)

3.6 JQuery

```
$("#slider-range").slider({
    range: true,
    step: 10000,
    min: 0,
    max: parseInt("{{maxprice}}", 10),
    values: [parseInt("{{ominprice}}", 10), parseInt("{{omaxprice}}", 10)],
    slide: function (event, ui) {
        $("#price0").text(ui.values[0].toLocaleString());
        $("#price1").text(ui.values[1].toLocaleString());
    }
});
```

Obrázek 9: Ukázka JQuery

Některé HTML stránky jsou doplněny o JavaScript přesněji o JQuery. Obrázek 8 Ukázka JQuery obsahuje inicializaci a funkčnost sliderů (posuvníků) na stránce search.html (Filtrovat vozidla). Ve dvojitéch složených závorkách jsou načteny proměnné z JSONu, které posílá Django společně se stránkou.

3.7 Ajax

```
function buy_car() {
    $.ajax({
        url: window.location.origin + '/buy_car?car_id=' + "{{ car.id }}" + "&url=" + window.location.href,
        type: "GET",
        success: function (result) {
            M.toast({html: result, classes: 'rounded'});
        }
    });
}
```

Obrázek 10: Ukázka Ajax

Ajax se používá pro asynchronní zpracování požadavků. Já jsem Ajax využil pro spuštění pythonovského skriptu (buy_car ve view.py), který odesílá e-mail majiteli inzerátu s údaji uživatele, který chce dané vozidlo koupit.

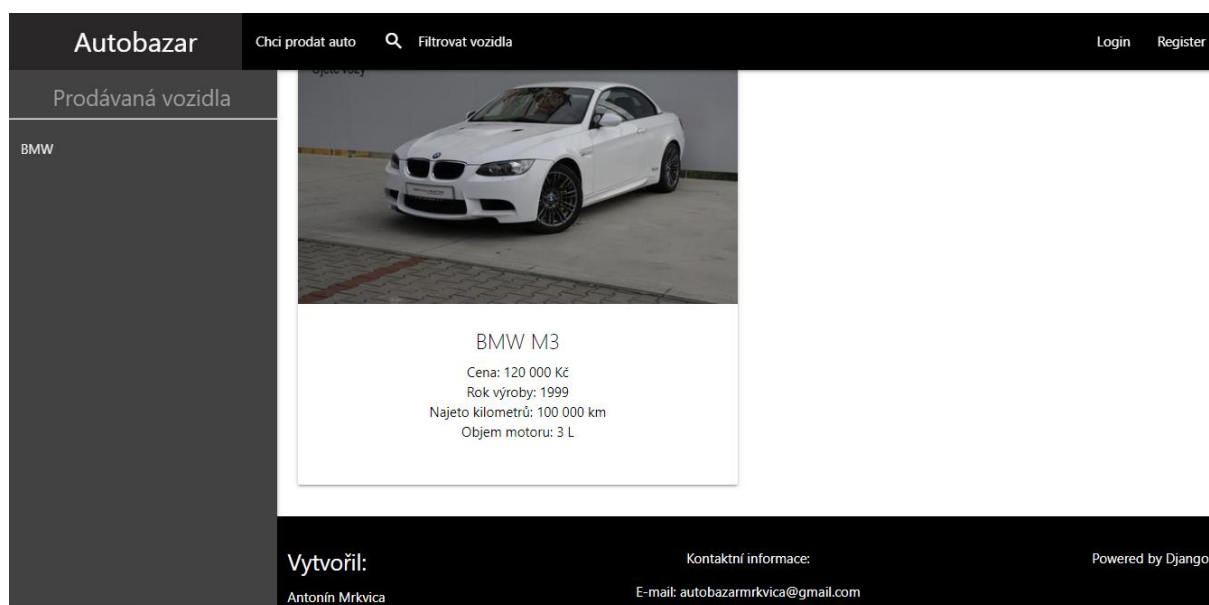
3.8 Verzovací nástroj GIT

Během celého projektu jsem používal verzovací nástroj GIT. Projekt se tímto nástrojem nahrává na stránky GITu [www.github.com](https://github.com/antoninmrkvica/Autobazar) konkrétně na adresu <https://github.com/antoninmrkvica/Autobazar>. Díky tomuto nástroji jsem mohl projekt vyvíjet a testovat na různých počítačích (Stolní PC, notebook) i odlišné platformy (Windows, Raspbian na Raspberry).

4 Uživatelská příručka

4.1 První pohled

Po prvním vstupu na webovou stránku Autobazaru uvidíme nahoře navigátor zleva Autobazar (odkaz na domovskou stránku autobazaru), Chci prodat auto (odkaz na stránku pro vyplnění inzerátu), Filtrovat vozidla. Na pravé straně navigátoru vidíme Login (přihlášení), Register (registrace) po přihlášení se tyto dvě položky změní na Logout (odhlášení) a Uživatelské jméno (odkaz na stránku s údaji o vašem profilu a možnostmi změny hesla). Navigátor vlevo tzv. Sidenav nám ukazuje značky všech aut právě prodáváných vozidel. Uprostřed stránky se nachází jednotlivé karty s inzeráty automobilů. Dole na stránce se skrývá footer s informacemi o webu a kontaktními údaji.



Obrázek 11: Domovská stránka Autobazaru

4.2 Filtrování vozidel – vyhledávání inzerátů

Nyní máme dvě možnosti. Jestliže si chceme inzeráty pouze prohlížet, nemusíme se přihlašovat a můžeme si vybrat inzerát buď na domovské stránce, nebo filtrovat inzeráty pomocí levého navigátoru nebo pomocí odkazu Filtrovat vozidlo.

Pokud klikneme na položku „BMW“ nacházející se v navigátoru vlevo, tak se položka rozbálí na poskytované modely dané značky, konkrétně modely značky BMW. Po vybrání modelu se inzeráty vyfiltrují pouze na modely dané značky.

Jestliže klikneme na Filtrovat vozidla, otevře se stránka se třemi posuvníky, které určují rozsah požadovaného filtru (Viz. Obrázek 12 Filtrování automobilů). Pokud tedy hledáme vozidla s rokem výroby nad 2000, jednoduše klikneme na levý posuvník (u čísla 1999 viz. Obrázek 12 Filtrování Automobilů) pod názvem „Rok výroby“

a přetáhneme jej doprava, dokud se číslo vlevo na posuvníku nezmění na požadovanou hodnotu v našem případě na 2000. Potvrdíme tlačítkem „POUŽIT FILTR“. Poté se zobrazí pouze inzeráty (náhledy) automobilů námi vybraným filtrem.



Filtrování vozidel

0Cena400 000

1999Rok výroby2005

0Najeto150 000

POUŽIT FILTER

Obrázek 12: Filtrování automobilů

4.3 Prohlížení inzerátu, koupě vozidla

Karta inzerátu obsahuje pouze ukázkové informace daného vozidla. Kliknutím na kartu se stránka přesměruje na inzerát daného vozidla.

BMW M5

Prodávající: [asd](#)

Značka: BMW	Turbo: Ano	Rok výroby: 2005
Model: M5	Typ Paliva: Benzín	Najeto: 150 000 km
Objem motoru: 2 L	Výkon: 100 kw (134 HP)	Cena: 400 000 Kč

Popis: Celkový stav automobilu zachovalý. Po prvním majiteli.

Závady: Žádné

Opravy: Žádné

CHCI KOUPIT TOHLE AUTO

Obrázek 13: Karta inzerátu BMW M5

V této kartě se nahoře nachází fotografie reálného vozidla. Pod obrázky nalezneme detailní informace o daném automobilu. Dole na stránce pak vidíme tlačítko „CHCI KOUPIT TOHLE AUTO“, které odesílá email majiteli inzerátu s vašimi kontaktními údaji, jestliže jste přihlášení.

4.4 Registrace a Přihlášení

Pro registraci klikneme vpravo nahoře na tlačítko „Register“. Stránka se přesměruje na registrační stránku. Po vyplnění všech údajů klikneme na tlačítko „REGISTROVAT“. Po registraci budete automaticky přihlášení.

Autobazar Chci prodat auto Filtrovat vozidla Login Register

Prodávaná vozidla

BMW

Uživatelské jméno Heslo Znovu heslo

E-mail Telefonní číslo

REGISTROVAT ➤

Vytvořil: Antonín Mrkvica

Kontaktní informace: E-mail: autobazarmrkvica@gmail.com

Powered by Django

Obrázek 14: Stránka registrace

Pro přihlášení klikneme na tlačítko „Login“. Vyplníme uživatelské jméno a heslo. Potvrdíme tlačítkem „PŘIHLÁSIT SE“. Poté budete přesměrováni na domovskou stránku.

4.5 Prodávání automobilu

Pro prodej vozidla musíte být přihlášení (viz. výše 4.4 Registrace a Přihlášení). Klikneme na tlačítko „Chci prodat auto“. Poté se nám otevře stránka s formulářem pro prodej automobilu. Vyplníme všechna pole, přidáme reálné fotografie automobilu (přidat obrázky) a potvrdíme vystavení inzerátu tlačítkem „ODESLAT“. Inzerát se ihned přidá mezi ostatní inzeráty.

Prodat auto

Značka

Zvol značku auta

Model

Objem motoru [L]

Turbo

Ne

Výkon [kW]

Tachometr [km]

Cena [Kč]

Rok výroby

Typ paliva

Popis

Poškození

Opravy

PŘIDAT OBRÁZKY

ODESLAT

Obrázek 15: Stránka pro prodej vozidla

4.6 Mazání a Správa inzerátu – editace

Mazat inzerát může pouze majitel inzerátu nebo administrátor stránky. V záložce daného inzerátu se vpravo dole nachází dvě kruhová tlačítka. První s názvem „EDIT“ pro úpravu inzerátu a druhé s ikonkou koše pro mazání inzerátu. Po kliknutí na ikonku koše se inzerát ihned smaže. Po kliknutí na „EDIT“ vás stránka přesměruje na formulář podobný při přidávání vozidla. Zde můžete cokoli změnit, doplnit nebo opravit, přidat či odebrat obrázky. Potvrdíme tlačítkem „ODESLAT“.

4.7 Přidávání komentářů a profil uživatele

Komentáře slouží ke zhodnocení uživatele po uskutečněním odkupu vozidla. Přes inzerát a kliknutím na odkaz prodávajícího, který se nachází vpravo, se dostaneme na profil prodávajícího, kde vidíme kontaktní údaje prodávajícího, dále komentáře ostatních uživatelů a dole pole pro vytvoření komentáře. Po vyplnění pole „Komentář“ potvrdíme tlačítkem „ODESLAT“. Vlastní komentáře lze mazat tlačítkem „Odstranit“.

4.8 Správa uživatelů – administrátor

Administrátor stránky je automaticky první registrovaný uživatel. Tento uživatel se speciálními právy může nejen to, co běžný uživatel jako nakupovat, přidávat, komentovat, ale především může mazat neslušné inzeráty (obrázky), komentáře nebo uživatele.

Uživatelé lze spravovat v sekci „Správa uživatelů“, kde se nachází všichni uživatelé. Zde lze rozšířit kolektiv administrátorů tím, že uživateli přidáme administrátorská práva zaškrtnutím políčka („checkboxu“) u příslušného uživatele ve sloupci „Práva“. Potvrdíme tlačítkem „ULOŽIT NASTAVENÍ PRÁV“. Pro odstranění uživatele klikneme na červené tlačítko „ODSTRANIT“ na řádce příslušného uživatele.

Správa uživatelů	
Profil	Práva
<u>UživatelA</u>	<input type="checkbox"/>
<u>admin</u>	<input checked="" type="checkbox"/>
<div>ULOŽIT NASTAVENÍ PRÁV ➤</div>	
<div>ODSTRANIT</div>	
<div>ODSTRANIT</div>	

Obrázek 16: Správa uživatelů

5 Závěr

Výsledkem práce je funkční webová aplikace pro autobazar. V průběhu vytváření aplikace jsem se potýkal s mnoha problémy. Věděl jsem, že se s nimi budu potýkat. Ale jelikož jsem framework Django už před touto zkušeností (projektem) ve velké míře znal, tak jsem čekal, že tvoření toho projektu nebude takový problém, ale i sám framework Django mi občas „hodil nějaký ten klacek pod nohy“. Tímto projektem jsem se hodně naučil například: rozvržení času pro delší práce, soustředění práce v jednu chvíli na danou problematiku. Došel jsem k názoru, že se člověk má stále co učit.

6 Seznam obrázků

Obrázek 1: Use Case.....	6
Obrázek 2: Definovaná tabulka pro uživatele.....	7
Obrázek 3: Tabulka Automobilu.....	8
Obrázek 4: Tabulka Komentáře.....	8
Obrázek 5: urls.py	8
Obrázek 6: Přihlášení	9
Obrázek 7: Stránka pro přihlášení	9
Obrázek 8: HTML stránka pro Přihlášení (zdrojový kód)	10
Obrázek 9: Ukázka JQuery.....	11
Obrázek 10: Ukázka Ajax	11
Obrázek 11: Domovská stránka Autobazaru	12
Obrázek 12: Filtrování automobilů	13
Obrázek 13: Karta inzerátu BMW M5	13
Obrázek 14: Stránka registrace	14
Obrázek 15: Stránka pro prodej vozidla	15
Obrázek 16: Správa uživatelů.....	16

7 Seznam použitých zdrojů

Django [online], poslední aktualizace 28. 3. 2019 00:15 [cit. 7. 4 2019], Wikipedie. Dostupné z WWW: <https://cs.wikipedia.org/wiki/Django>

Ajax [online], poslední aktualizace 21. 8. 2018 [cit. 7. 4 2019], Wikipedie. Dostupné z WWW: <https://cs.wikipedia.org/wiki/AJAX>

SQL [online], poslední aktualizace 12. 1. 2019 22:36 [cit. 7. 4 2019], Wikipedie. Dostupné z WWW: <https://cs.wikipedia.org/wiki/SQL>

CSS [online], poslední aktualizace 12. 1. 2019 22:34 [cit. 7. 4 2019], Wikipedie. Dostupné z WWW: https://cs.wikipedia.org/wiki/Kask%C3%A1dov%C3%A9_styl%C3%BD

JSON [online], poslední aktualizace 6. 5. 2018 22:35 [cit. 7. 4 2019], Wikipedie. Dostupné z WWW: https://cs.wikipedia.org/wiki/JavaScript_Object_Notation

GIT [online], poslední aktualizace 3. 12. 2018 00:51 [cit. 7. 4 2019], Wikipedie. Dostupné z WWW: <https://cs.wikipedia.org/wiki/Git>

HTML [online], poslední aktualizace 12. 1 2019 22:36 [cit. 7. 4 2019], Wikipedie. Dostupné z WWW: https://cs.wikipedia.org/wiki/Hypertext_Markup_Language

Yuml me[online], poslední aktualizace 7. 4. 2019 [cit. 7. 4 2019], Yuml me. Dostupné z WWW: <https://yuml.me>