

Porovnávač vlastností kozmetických produktů

Bakalárska práca

Zuzana Lysová

Vedúci práce:
Ing. Pavel Turčíněk, Ph.D.

Brno 2022

NA MIESTE TOHTO LISTU
SA NACHÁDZA ORIGINAL
ZADANIA PRÁCE.

Chcela by som sa poďakovať všetkým, ktorí ma podporovali a stáli pri mne. Najviac ďakujem rodine, keďže bez jej podpory by bolo štúdium na vysokej škole oveľa náročnejšie. Taktiež ďakujem priateľovi, ktorý bol stále so mnou a kamarátom za podporu počas celého štúdia. V neposlednom rade veľmi pekne ďakujem vedúcemu práce, pánovi Pavlovi Turčínkovi, za pomoc s výberom témy a cenné rady ohľadom tvorby práce.

Čestné prehlásenie

Prehlasujem, že som prácu *Porovnávač vlastností kozmetických produktov* vypracovala samostatne a všetky použité zdroje a informácie uvádzam v zozname použitej literatúry. Súhlasím, aby moja práca bola zverejnená v súlade § 47b zákona č. 111/1998 Sb., o vysokých školách, v znení neskorších predpisov a v súlade s platnou Směrnici o zveřejňování vysokoškolských závěrečných prací. Prehlasujem, že tlačaná podoba záverečnej práce a elektronická podoba záverečnej práce zverejnená v aplikácii Závěrečné práce v Univerzitním informačním systému je identická.

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

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

V Brne dňa 6. mája 2022

.....
podpis

Abstract

LYSOVÁ, ZUZANA. *Properties comparator of cosmetic products*. Bachelor thesis. Brno : Mendel University in Brno, 2022.

The bachelor thesis deals with the design and implementation of an application that serves as a comparator of the properties of cosmetic products. It is able to download product information from several Slovak cosmetic e-shops and then present them to the user via a web application. The user can search for these cosmetic products either by name, category or brand of the product and then can be redirected to the product offer of a particular e-shop.

Key words

web scraping, web application, HTML, Python, BeautifulSoup, MongoDB, Flask

Abstrakt

LYSOVÁ, ZUZANA. *Porovnávač vlastností kozmetických produktov*. Bakalárska práca. Brno : Mendelova univerzita v Brně, 2022.

Bakalárska práca sa zaoberá návrhom a implementáciou aplikácie, ktorá slúži ako porovnávač vlastností kozmetických produktov. Je schopná stiahnuť informácie o produktoch z niekoľkých slovenských kozmetických e-shopov a následne ich prezentovať užívateľovi prostredníctvom webovej aplikácie. Užívateľ môže tieto kozmetické produkty vyhľadávať buď podľa názvu, kategórie, alebo značky produktu a následne môže byť presmerovaný na ponuku produktu konkrétneho e-shopu.

Kľúčové slová

web scraping, webová aplikácia, HTML, Python, BeautifulSoup, MongoDB, Flask

Obsah

1	Úvod a cieľ práce	8
1.1	Idea	8
1.2	Cieľ	9
2	Prehľad literatúry a prameňov	11
2.1	Porovnávače cien	11
2.2	Prehľad najznámejších porovnávačov	11
2.3	Web scraping	12
2.4	Etika a legálnosť web scrapingu	13
2.5	Programovací jazyk Python	14
2.5.1	Knižnica BeautifulSoup	14
2.5.2	Framework Flask	14
2.6	Databáza MongoDB	15
3	Metodika	16
4	Výsledky	18
4.1	Cieľová skupina	18
4.2	Funkčné a nefunkčné požiadavky	18
4.2.1	Funkčné požiadavky	18
4.2.2	Nefunkčné požiadavky	19
4.3	Diagram prípadov použitia	19
4.4	Entitne relačný diagram	19
4.5	Analýza zdrojových webov	22
4.6	Tvorba web crawleru	23
4.7	Problémy spojené s web scrapingom	24
4.8	Databáza	26
4.8.1	Výber databázového systému	26
4.8.2	Naplnenie databázy dátami	26
4.8.3	Nástroj MongoDB Compass	27
4.9	Návrh a tvorba webovej aplikácie	28
5	Diskusia a záver	33

5.1	Návrhy na vylepšenie	33
5.2	Záver	34
Literatúra		35
<hr/>		
Zoznam tabuliek		38
<hr/>		
Zoznam obrázkov		39
<hr/>		

1 Úvod a cieľ práce

V dnešnej dobe sa stávajú nákupy cez internet viac a viac populárne. Ľudia si na to zvykli aj v dôsledku pandémie Covid-19, keďže vládne opatrenia nedovoľovali otvorené prevádzky mnohých obchodov a služieb. Podľa údajov ŠSTATISTICKÉHO ÚRADU SLOVENSKEJ REPUBLIKY (2021) vzrástlo percento nákupov cez internet od roku 2019 do roku 2021 z 80,2 % na 86,5 % u žien vo veku 16 až 24 rokov. U žien vo veku 25 až 54 rokov bol pozorovaný nárast zo 76,6 % na 82,7 % (viď **obrázok 1.1**). Tieto čísla hovoria o percente žien, ktoré za posledných dvanásť mesiacov urobili nejaký nákup tovaru online v kategórii oblečenia (ktorá bola zo všetkých skúmaných kategórií kozmetike najbližšie). Tieto skupiny obyvateľstva (ženy vo veku 16–54 rokov) boli zvolené preto, že sú hlavné cieľové skupiny webovej aplikácie "porovnávač vlastností kozmetických produktov".

Nákup kozmetiky je možný buď v kamenných obchodoch ako Douglas, Sephora a pod., v drogériách, poprípade v lekárňach. Nie všetky produkty sú ale dostupné v obchodoch a častokrát sú zákazníci z menších miest a obcí, kde sa špecializované kozmetické predajne nenachádzajú, a tak im ostáva len objednať produkt online alebo vycestovať do mesta, kde takúto predajňu nájdú. To môže viesť k tomu, že si zákazníci radšej nájdú kozmetický produkt na nejakom eshope a nakúpia ho online. Je pravda, že slovenský trh neponúka nespočetné množstvo kozmetických e-shopov, no ak by existoval web, na ktorom sa dajú nájsť všetky ponuky e-shopov na jednom mieste, mohlo by to online nakupovanie kozmetiky výrazne zrýchliť a uľahčiť.

1.1 Idea

V tejto podkapitole sú zodpovedané základné otázky spojené s myšlienkou a podstatou webovej aplikácie "porovnávač vlastností kozmetických produktov". Dali by sa sem zaradiť odpovede na otázky čo má aplikácia robiť, aký bude jej prínos a v čom je iná ako ostatné.

1. Čo má aplikácia robiť?

Aplikácia bude slúžiť primárne na vyhľadávanie konkrétnych produktov dekoratívnej kozmetiky podľa názvu produktu alebo podľa iných kritérií, ako značka alebo kategória produktu. Výsledkom hľadania bude

produkt, resp. zoznam produktov zodpovedajúci hľadaným parametrom a informácie o tomto produkte (názov, značka, cena, obrázok produktu a odkaz na konkrétnu ponuku e-shopu).

2. Aký bude jej prínos?

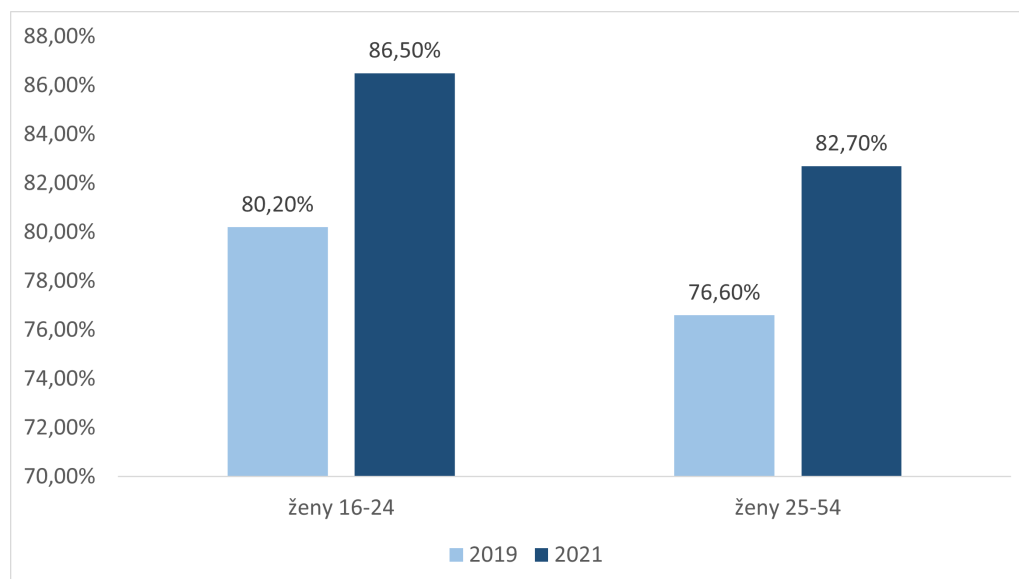
Prínosom tejto aplikácie bude to, že kupujúci budú môcť ušetriť čas pri prezeraní všetkých e-shopov a hľadaní najvýhodnejšej varianty a všetko nájdu na jednom mieste.

3. V čom je aplikácia iná ako ostatné?

Zákazníci tu nájdu niekoľko významnejších slovenských e-shopov s kozmetikou, pričom na už existujúcich konkurenčných porovnávačoch je týchto e-shopov len pár, a tak informácie, ktoré sú dohľadateľné napr. na portáli heureka nie sú stopercentne relevantné. Užívateľ je síce schopný nájsť si daný produkt s odkazom na pár e-shopov, no nemá informáciu o tom, aká je cena na iných, relatívne veľkých a známych, slovenských e-shopoch, ktoré sa na danom porovnávači nenachádzajú (napr. pinkpanda, douglas, makeup...).

1.2 Cieľ

Cieľom tejto bakalárskej práce je vytvoriť aplikáciu, ktorá bude schopná získať informácie (názov, značka, cena a pod.) o kozmetických produktoch z rôznych slovenských e-shopov, uloženie týchto informácií do databázy a vytvorenie webovej aplikácie, prostredníctvom ktorej budú tieto informácie poskytnuté užívateľom. Aplikácia by mala byť schopná uľahčiť a urýchliť nakupovanie kozmetiky online, keďže zákazník bude vedieť nájsť produkty dekoratívnej kozmetiky z takmer všetkých významnejších slovenských e-shopov a nebude musieť prezeráť ponuky týchto e-shopov jednotlivo.

**Obrázok 1.1**

Graf percenta internetových nákupov na Slovensku
(spracované podľa údajov Štatistického úradu SR (2021))

2 Prehľad literatúry a prameňov

2.1 Porovnávače cien

Porovnávače cien sú významným nástrojom elektronického obchodu. Umožňujú užívateľom nájsť ponuky rôznych e-shopov na rovnaký produkt a pomáhajú im vybrať si z viacerých variantov ceny produktu, ceny dopravy, doby doručenia a podobne. Väčšina porovnávačov cien neumožňuje priamo nákup prostredníctvom ich webovej stránky, ale len poskytuje rozličné informácie o produktoch, ako je napríklad ich cena, špecifikácia, vlastnosti, fotografie či hodnotenie zákazníkov na daný produkt, a zároveň tlačidlo na presmerovanie zákazníka na ponuku daného e-shopu. (BERÁNEK; REMEŠ, 2017)

Výhody porovnávačov cien môžeme rozdeliť na dve kategórie, ktorými sú výhody pre zákazníkov a výhody pre e-shopy. Hlavnou výhodou pre zákazníkov je šetrenie času, ktorý by museli venovať prezeraním desiatok e-shopov. Pri existencii porovnávača cien nájdú potrebné produkty za najnižšie ceny jednoducho, rýchlo a prehľadne. Pre e-shopy sú porovnávače produktov prínosné hlavne z marketingového hľadiska.

2.2 Prehľad najznámejších porovnávačov

Na československom trhu je aktuálne najväčším webom na porovnávanie cien a vlastností produktov heureka.sk respektíve heureka.cz. Heureka vznikla na českom trhu v roku 2007, na slovenskom trhu rok po tom, v roku 2008. Porovnávače heureka.sk a heureka.cz patria do skupiny Heureka Group a.s., pod ktorú spadajú aj porovnávače ďalších 7 európskych krajín (Bulharsko, Maďarsko, Slovinsko, Chorvátsko, Bosna a Hercegovina, Srbsko a Rumunsko). Všetky porovnávače Heureka Group a.s. zaznamenávajú 23 miliónov užívateľov mesačne a obsahujú produkty 55 tisíc e-shopov (HEUREKA, 2022).

Ďalšie, menej významné, porovnávače sú na slovenskom trhu napríklad nakup.sk s priemerným mesačným počtom užívateľov za rok 2021 cca 740 tisíc (NAJNAKUP, 2022) a pricmania.sk s 300 tisíc užívateľmi mesačne (PRICEMANIA, 2022). Na českom trhu je to napríklad zbozi.cz alebo hledej ceny.cz (u oboch portálov sú štatistiky nedostupné).

Existujú aj špecializované porovnávače, resp. vyhľadávače produktov, ktoré sa zaoberajú len istou kategóriou produktov, napríklad módou a obuvou a medzi ne patrí napr. glami.sk (glami.cz). Ďalej sú dostupné aj vyhľadávače nábytku favi.sk (favi.cz) alebo biano.sk (biano.cz).

V súčasnosti ale na našom trhu neexistuje žiaden porovnávač/vyhľadávač, ktorý by bol zameraný na kozmetiku, či už dekoratívnu, telovú alebo pleťovú. Na zahraničnom trhu takýchto webov tiež nie je veľa. Najznámejší porovnávač, ktorý je zameraný čisto na kozmetiku je web cosmetify.com, ktorý ale ponúka produkty z prevažne z britských e-shopov, čo nemusí byť pre českých a slovenských zákazníkov lákavé, vzhľadom na výšku poštovného, clo, dĺžku doby doručenia, jazyk a podobne.

2.3 Web scraping

Web scraping je takmer neoddeliteľnou súčasťou získavania dát z rôznych online zdrojov. Tieto dáta môžu byť štruktúrované, pološtruktúrované i neštruktúrované a zobrazujú sa vo forme webových stránok, databáz, emailov, tweetov, fotografií, videí a podobne (WATSON, 2014). Web scraping sa v obrovskej miere využíva aj pri tvorbe vyhľadávačov a porovnávačov produktov.

„Zvyšujúci sa počet online predajcov vedie k zvyšovaniu počtu online ponúk produktov. Zvládajúť porovnávanie takéhoto množstva produktov a zodpovedajúcich dát o cene je len ťažko zvládnuteľné manuálne“ (HORCH; KETT; WEISBECKER, 2015). Práve preto je web scraping stále dôležitejší v dnešnom svete „veľkých dát“, keďže slúži na automatizované získavanie dát a na premenu neštruktúrovaných dát do štruktúrovaného formátu.

Web scraping prebieha v troch fázach – analýza webu, web crawling a usporiadanie údajov.

Analýza webu sa zaoberá štruktúrou HTML stránky, súborov formátu CSS alebo XML, webových databáz a podobne. Cieľom je objaviť a pochopiť, akým spôsobom sú žiadané dáta uložené.

Na web crawling, ktorý slúži na indexovanie informácií, sa používajú najčastejšie programovacie jazyky Python alebo R a knižnice BeautifulSoup, rvest a podobne.

Po spracovaní potrebných údajov nastáva fáza organizácie dát, na ktorú sa taktiež používajú spomínané programovacie jazyky Python alebo R a NLP (Natural Language Processing) knižnice (KROTOV, 2018). Taktiež je potrebná aj práca s rôznymi databázovými systémami. Získané dáta sú najčastejšie ukladané do súborov formátu CSV, XLSX alebo JSON, poprípade sa ukládajú do databázy MySQL, PostgreSQL, MongoDB a podobne (HEYDT, 2018).

V dnešnej dobe existujú aj nástroje, ktoré fungujú tak, že užívateľ len „ukazuje a kliká“ na požadované objekty na webe, a tak potrebné dáta získava. Medzi takéto nástroje patrí napríklad import.io, Dexi.io, ParseHub, FMiner atď. Tieto nástroje nie vždy pracujú tak, ako užívateľ očakáva, keďže weby môžu mať rôzne podoby a tvorba a modifikácia webov prebieha veľmi rýchlo. Preto je často nevyhnutný zásah vlastného naprogramovaného kódu do web scrapingu (KROTOV, 2018).

Vzhľadom na to, že rôzne weby majú rôzne štruktúry stránok, s web scrapingom sú spojené aj viaceré problémy, ako napríklad dostupnosť, formát, veľkosť alebo prezentácia dát (KHAN, 2020). Preto je prvý krok pri získavaní informácií, napr. o produktoch z e-shopu, najprv identifikácia a extrakcia len jedného produktu.

Pri získavaní dát je dôležité to, aby získané informácie boli aktuálne, konzistentné a aby sa zmeny, ktoré sa udejú v zdrojových neštruktúrovaných dátach, preniesli aj do získaných, štruktúrovaných dát.

Na to, aby sa dal vytvoriť algoritmus na získanie a uloženie informácií z webových stránok je často potrebná analýza HTML štruktúry a jej elementov. Výsledkom takéhoto algoritmu sú HTML elementy spolu s konkrétnymi záznamami (HORCH; KETT; WEISBECKER, 2015).

2.4 Etika a legálnosť web scrapingu

Aj pri web scrapingu je nutné zaoberať sa stránkou legálnosti a etiky. Hlavné problémy spojené s web scrapingom na úrovni legálnosti sú protizákonný prístup a použitie dát, copyright, porušenie obchodného tajomstva, porušenie zmluvy a podobne. Na etickej úrovni môže ísť o narušenie súkromia osôb alebo organizácií, diskriminácia a zaujatosť, kvalita dát pri rozhodovaní atď. Tieto aspekty môžu viesť k rôznym problémom, ako zásah do súkromia, či zhoršenie reputácie firmy. Preto sa musia výskumní pracovníci riadiť aj množstvom etických a legálnych náležitostí, aj keď v súčasnosti neexistujú priamo zákony, ktoré sa zaoberajú priamo web scrapingom. Na zákonné obmedzenia slúžia v dnešnej dobe hlavne „Podmienky použitia“ webov. (KROTOV, 2020)

2.5 Programovací jazyk Python

Programovací jazyk Python vznikol v roku 1991. Využíva sa u mnohých platforiem, s ktorými sa ľudia stretávajú na každodennej báze, ako je napríklad YouTube, Google, Netflix a podobne. Dá sa použiť v rôznych oblastiach, ako veda, biznis alebo umenie, a v spojení s prácou s webmi, dátami a na administráciu systémov (LUBANOVIC, 2015).

Programovací jazyk Python dlhodobo patrí medzi najobľúbenejšie programovacie jazyky. Podľa údajov z Githubu (viď **tabuľka 2.1**) zaznamenal Python za posledných 5 rokov najväčší rast spomedzi všetkých programovacích jazykov, až 12,8 %. Kritérium obľúbenosti je na GitHubu merané vzhľadom na počet vyhľadávaní daného jazyka na Google (CARBONNELLE, 2022).

Podľa developerského prieskumu StackOverflow za rok 2021 (viď **tabuľka 2.2**) je Python taktiež medzi prvými piatimi najobľúbenejšími jazykmi. Prieskum mal vyše 83 000 respondentov. Otázka na respondentov sa pýtala na to, ktoré programovacie, skriptovacie a značkovacie jazyky ľudia použili za posledný rok, a či s nimi chcú pracovať aj v nasledujúcom roku, pričom mohli označiť viac možností.

2.5.1 Knižnica Beautiful Soup

Beautiful Soup je knižnica programovacieho jazyka Python, ktorej hlavnou funkciou je získavanie dát zo súborov vo formátoch HTML a XML. Táto knižnica umožňuje navigáciu, vyhľadávanie a úpravu údajov získaných z webu (RICHARDSON, 2018). Medzi hlavné výhody knižnice Beautiful Soup patrí to, že práca je veľmi rýchla a spracovanie stránok prebieha rovnako ako v prehliadači (MAITHANI, 2020).

2.5.2 Framework Flask

Flask je webový mikroframework napísaný v programovacom jazyku Python. Je vhodný na jednoduché webové aplikácie alebo ich časti. Flask sám o sebe neumožňuje veľa funkcionalít, ale podporuje rôzne rozšírenia, ktoré sú schopné do aplikácie ďalšie funkcie pridať. Tieto funkcie môžu byť napríklad overovanie formulárov, spracovávanie údajov a pod. Flask využíva šablónovací systém Jinja, ktorý je rýchly, pomerne dosť rozšírený a bezpečný. (PALLETS, ©2010)

Tabuľka 2.1 Popularita programovacích jazykov podľa GitHubu (CARBONNELLE, 2022)

Pozícia	Programovací jazyk	Popularita
1	Python	28,52 %
2	Java	18,12 %
3	JavaScript	8,9 %
4	C/C++	7,62 %
5	C#	7,39 %

Tabuľka 2.2 Popularita programovacích jazykov podľa StackOverflow (STACKOVERFLOW, 2022)

Pozícia	Programovací jazyk	Popularita
1	JavaScript	64,96 %
2	HTML/CSS	56,07 %
3	Python	48,24 %
4	SQL	47,08 %
5	Java	35,35 %

2.6 Databáza MongoDB

MongoDB patrí medzi NoSQL („Not Only SQL“) databázy, konkrétne do kategórie dokumentových databáz. Ďalšie kategórie NoSQL databáz (podľa toho, aké dáta sú ukladané a používané) sú grafová, key-value a stĺpcová. MongoDB je objektovo-orientovaná, čiže umožňuje ukladanie aj neštruktúrovaných dát. Dáta sú ukladané ako BSON dokumenty (binárny JSON), ktoré sú zhromažďované do kolekcí. Štruktúru dokumentov tvoria dvojice kľúč-hodnota. Databáza MongoDB je vhodná pri skladovaní veľkého množstva dát, ktoré sa často menia. (PARKER, 2013)

3 Metodika

V tejto kapitole sú popísané kroky, ktoré viedli k vytvoreniu tejto práce.

Existuje viacero modelov životného cyklu vývoja systému. Proces tvorby aplikácie, ktorá je produktom tejto bakalárskej práce, sa opiera o vodopádový model, ktorého schéma je znázornená na **obrázku 3.1**.

V rámci prvej fázy, fázy *špecifikácie a definície požiadaviek*, sú definované základné funkčné a nefunkčné požiadavky z pohľadu užívateľa a z pohľadu samotnej aplikácie. Definujú sa teda dve skupiny požiadaviek. Do prvej skupiny spadajú požiadavky popisujúce to, čo bude schopný robiť na webovej stránke porovnávača vlastností kozmetiky užívateľ. Druhá skupina pozostáva z požiadaviek na aplikáciu, a teda to, čo všetko musí viesť aplikácia robiť na to, aby mohla fungovať (od získania dát z e-shopov až po zobrazenie produktov na stránke).

Etapy *analýza, návrh a implementácia* sa v tomto projekte vzájomne prelínajú a dopĺňajú. Nie vždy je možné jednoznačne určiť, do ktorej konkrétnej fázy daný proces spadá, keďže nie je snaha vodopádový model úplne kopírovať, ale len sa ním inšpirovať.

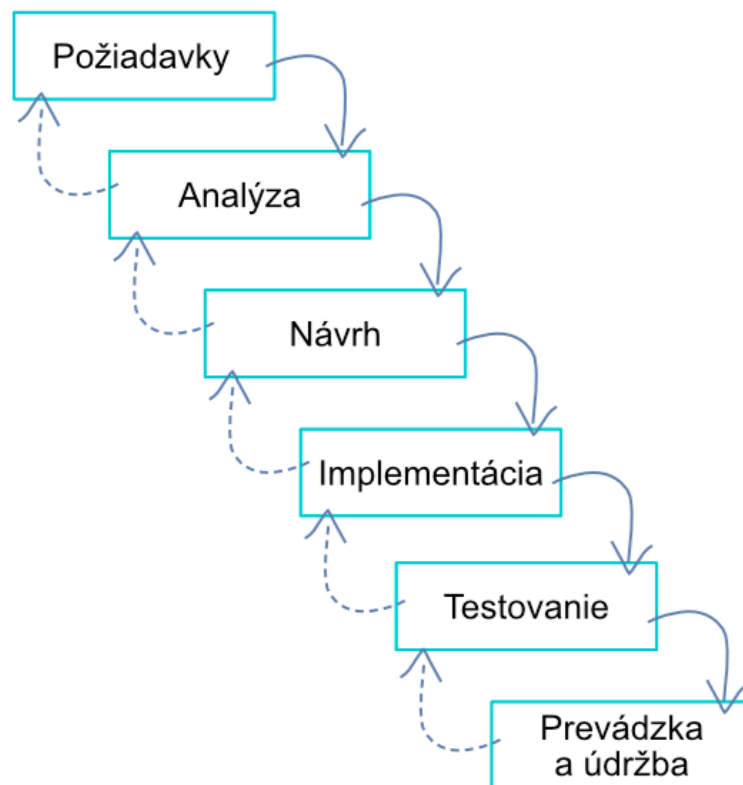
Do etapy *analýza* by sa dali zaradiť procesy ako objektovo orientovaná analýza a diagram prípadov použitia, štruktúrna analýza a entitne relačný diagram, a analýza e-shopov, z ktorých sa budú čerpať informácie o produktoch.

V *návrhovej* fáze ide o výber technológií, čiže programovacích jazykov, databázového systému a pod. Ako jazyk implementácie web scraperu je zvolený programovací jazyk Python a knižnica BeautifulSoup. Čo sa týka databázového systému, je vybratý databázový systém MongoDB a v rámci webovej aplikácie je zvolený taktiež programovací jazyk Python a jeho framework Flask. Ďalej do etapy návrhu spadajú aj grafické návrhy aplikácie, ako výber farieb, drôtený model, ktorý obsahuje štruktúru a umiestnenie elementov na stránke, a neskôr aj úplný grafický návrh. Drôtený model je tvorený pomocou online nástroja draw.io a kompletný grafický návrh v desktopovej aplikácii Affinity Designer.

Po návrhovej fáze prichádza na rad *implementácia*, čiže procesy ako tvorba web scraperu, ktorý získa požadované dáta z e-shopov, uloženie informácií o produktoch do databázy a vytvorenie webovej aplikácie, ktorá bude slúžiť ako porovnávač, respektíve vyhľadávač kozmetických produktov.

V časti *testovania* ide, v prípade tohto projektu, o overenie funkčnosti aplikácie a to tak, že sa vyberie testovacia skupina potenciálnych budúcich užívateľov, ktorá pozostáva z cca 10 ľudí. Overí sa tak, či je web ľahko použiteľný, užívateľsky prívetivý a či funguje tak, ako má fungovať.

Na záver sa aplikácia uvedie do *prevádzky* a bude sa vylepšovať, či už po stránke pamäťovej náročnosti, rýchlosti načítania, užívateľskej prívetivosti, vzhľadu, funkcií a podobne.



Obrázok 3.1
Vodopádový model (vlastné spracovanie)

4 Výsledky

4.1 Cieľová skupina

Cieľová skupina webu, ktorý slúži ako vyhľadávač/porovnávač vlastností (najmä ceny) kozmetiky, sú prevažne ženy vo veku od 15 do 50 rokov, ktoré rady nakupujú kozmetiku online a chcú nakupovať za čo najlepšiu cenu.

4.2 Funkčné a nefunkčné požiadavky

V tejto kapitole popíšem funkčné a nefunkčné požiadavky na systém, čiže to, čo má aplikácia vedieť robiť (funkčné požiadavky) a to, ako bude aplikácia fungovať (nefunkčné požiadavky).

4.2.1 Funkčné požiadavky

Aplikácia bude vedieť analyzovať vybrané e-shopy a získať požadované informácie o produktoch z html štruktúry, resp. z API, ktoré e-shop poskytuje. Ďalej bude aplikácia vedieť tieto získané dáta uložiť do databázy a taktiež pristupovať k týmto informáciám v databáze a prezentovať ich užívateľom.

Užívateľ aplikácie bude schopný vyhľadať produkty podľa názvu a zistiť, ktorý e-shop ponúka daný produkt za najnižšiu cenu. Ďalej bude užívateľ môcť prehľadávať produkty podľa kategórie alebo značky, ktoré si vyberie zo zoznamu. Kliknutím na tlačidlo s logom e-shopu, ktoré sa bude nachádzať pri každom produkte, bude užívateľ presmerovaný na konkrétnu ponuku e-shopu.

4.2.2 Nefunkčné požiadavky

Používanie webovej aplikácie bude možné v akomkoľvek prehliadači. Dáta budú na stránku načítavané z databázy MongoDB. Informácie o produktoch budú aktualizované raz denne. Jazyk webovej aplikácie bude slovenčina a ceny produktov budú v eurách.

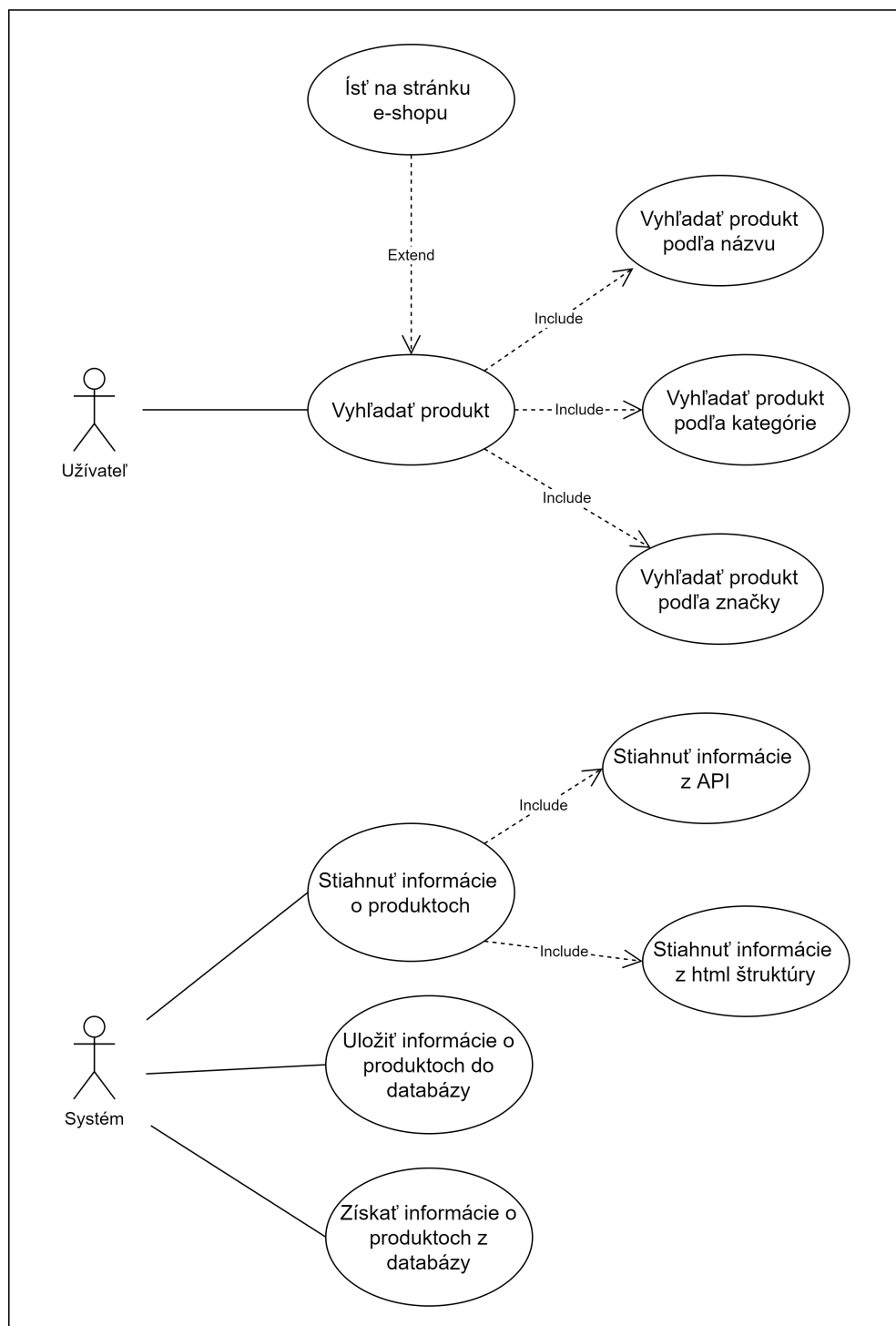
4.3 Diagram prípadov použitia

Diagram prípadov použitia, alebo use case diagram, je spojený s objektovo orientovaným prístupom k analýze systému. Patrí medzi UML (Unified Modeling Language) diagramy, ktoré predstavujú rôzne uhly pohľadov na navrhovaný systém a uľahčujú tak jeho návrh a analýzu. UML diagramy sa delia na diagramy štruktúry, správania a interakcie. Konkrétne príklady UML diagramov sú, okrem diagramu prípadov použitia, aj napríklad diagram tried, diagram aktivít, diagram nasadenia a ďalšie. (ČÁPKA, 2021)

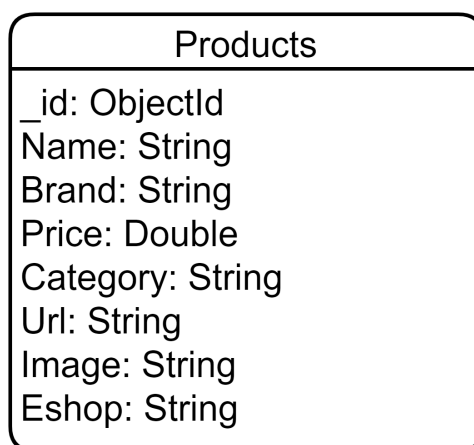
Use case (prípád použitia) je nejaká funkcionálna systém, ktorá prináša istý úžitok (Rábová, 2008). Hlavné zložky use case diagramu sú use case, aktori, hranice systému a väzby. V use case diagrame aplikácie porovnávača vlastností kozmetických produktov (viď **obrázok 4.1**) je možné vidieť dvoch aktórov – užívateľa a systém a ich prípady použitia.

4.4 Entitne relačný diagram

Entitne relačný diagram (ERD) patrí medzi základné nástroje štruktúrovanej analýzy. Slúži na zobrazenie statickej časti systému, na vyjadrenie dátovej štruktúry, vzťahov medzi dátami, pričom nemá informácie o funkciách, ktoré tieto dáta vytvárajú alebo používajú (RÁBOVÁ, 2008). Hlavné zložky ERD sú entity, atribúty a väzby. Vo vytvorenom entitne relačnom diagrame sa nachádza jedna entita – Products a jej atribúty, ktoré je možné vidieť na **obrázku 4.2**. Atribút "_id" je automaticky generovaný databázovým systémom. Ide o takzvané ObjectId, ktoré sa skladá z 12 bytov a pozostáva z časovej značky (4 bajty), náhodne generovanej hodnoty (5 bajtov) a inkrementačného počítadla, ktoré je inicializované taktiež na náhodnú hodnotu (3 bajty) (MONGODB, 2021).



Obrázok 4.1
Use case diagram (vlastné spracovanie)

**Obrázok 4.2**

ERD diagram (vlastné spracovanie)

```
{
  "_id": ObjectId,
  "Brand": String,
  "Name": String,
  "Price": Double,
  "Category": String,
  "Url": String,
  "Image": String,
  "Eshop": String
},
```

Obrázok 4.3Ukážka štruktúry dát v
databáze (vlastné spracovanie)

Vzhľadom na to, že sú dáta ukladané do databázy MongoDB, štruktúra databázy sa dá, okrem ERD diagramu, popísať aj pomocou súboru formátu JSON. Na **obrázku 4.3** sa nachádza ukážka štruktúry dát v MongoDB databáze, ktorá ukladá informácie o produktoch. Na **obrázku 4.4** sa nachádza konkrétna ukážka dát v databáze.

4.5 Analýza zdrojových webov

Prvým technickým krokom potrebným k vytvoreniu webovej aplikácie, ktorá slúži ako porovnávač/vyhľadávač kozmetiky zo slovenských kozmetických e-shopov je získanie informácií o produktoch, na čo je potrebná analýza webu vybraných internetových kozmetických obchodov. Zaujímá nás to, ako vyzerá HTML štruktúra, konkrétne jednotlivé tagy a atribúty týchto tagov, a v neposlednom rade ich obsah.

Na analýzu HTML štruktúry slúži nástroj "inspect" ("preskúmať"), ktorý poskytujú webové prehliadače. Umožňuje nám nielen sledovať jednotlivé HTML tagy v kóde, ale taktiež zvoliť kliknutím konkrétnu časť stránky, ktorú chceme analyzovať. Okrem analýzy HTML štruktúry (záložka "elements") je pomocou nástroja "preskúmať" taktiež možné sledovať to, čo sa deje v sieti pri načítaní stránky (záložka "network"), čo bude spomenuté nižšie.

Ako prvé potrebujeme zistiť, ako je zobrazený 1 produkt, a teda to, aký je to element, čo tento element obsahuje a aké má vlastnosti (class, id a pod.). Následne vieme takto získať informácie o každom jednom produkte na stránke.

Ďalší krok je určiť, ktoré konkrétne informácie o produktoch chceme získať. V prípade porovnávača vlastností kozmetiky je to značka, názov, cena, kategória a url daného produktu (aby sa užívateľ vedel na stránku produktu konkrétneho e-shopu jednoducho dostať), poprípade aj zdroj obrázku.

Pri niektorých e-shopoch bola analýza webu a jeho HTML štruktúry jednoduchá a bolo možné jednoznačne určiť atribúty jednotlivých HTML elementov, keďže vyzerajú napríklad takto: "class="sp-brand"". No napríklad pri e-shope Notino nastal problém, pretože triedy jednotlivých atribútov sa menili pri každom načítaní stránky a vyzerajú napríklad takto: "class = "sc-jeraig sc-kHOZWm kSc-BAw gGGBsj styled__StyledProductTile-sc-1yds6ou-0 czeUhw"". V tomto prípade teda nebolo možné, resp. by bolo veľmi zložité, vyparsovať túto stránku pomocou metód knižnice BeautifulSoup. Bolo by nutné použiť relatívne zložité regulárne výrazy a aj tak by sme nemali istotu, že sa mená tried o pár dní nezmenia na úplne iné. Preto je nutné zvážiť iný spôsob, ktorý môže viesť k získaniu informácií, ktoré potrebujeme parsovať, a to je zanalyzovať priebeh načítania stránky pomocou nástroja "network" webového prehliadača a zistiť, aké HTTP requesty bežia v sieti.

Zistilo sa, že pri načítaní stránky Notino je prevedený GET request, ktorého obsahom je štrukturovaný JSON dokument obsahujúci všetky informácie, ktoré potrebujeme k jednotlivým produktom.

Ďalším skúmaním sa zistilo to, že niektoré weby, ako aj napríklad vyššie spomenuté Notino, teda poskytujú API (Application Programming Interface), ktoré umožňuje prístup k požadovaným dátam, ktoré sú uložené vo formáte JSON alebo XML, a HTML tieto dáta len prezentuje užívateľom (BREUSS, 2021). Weby,

ktoré takéto API poskytujú sú práve weby, ktoré sa dajú nájsť aj na už existujúcich cenových porovnávačoch, ako je napríklad Heureka. Vzhľadom na to, že sú tieto dáta priamo poskytované e-shopmi na účely web scrapingu, je jednoduché prístupíť k jednotlivým informáciám, ktoré potrebujeme vyparsovať, keďže je dokument formátu JSON spracovaný prehľadne a je jednoducho čitateľný.

4.6 Tvorba web crawleru

V momente, keď máme analyzované štruktúry webov, využijeme získané informácie pri tvorbe web scraperu, resp. web crawleru. Na to je vhodný napr. programovací jazyk Python a knižnica BeautifulSoup, ktoré boli zvolené pri tomto projekte. Automatizovaný web scraper urýchli proces zhromažďovania informácií a je schopný zaznamenávať akékoľvek zmeny v dátach, ktoré časom nastanú (zmeny konkrétnych informácií, ako napr. cena produktu, nie zmeny v HTML štruktúre) (BREUSS, 2021). Na začiatku je nutné vytvoriť BeautifulSoup objekt, ktorý do seba ukladá obsah webovej stránky pomocou nasledujúcich príkazov:

```
import requests
from bs4 import BeautifulSoup
r=requests.get("http_adresa")
soup = BeautifulSoup(r.content, 'html.parser')
```

V prípade stránok, ktoré nemali jednoducho parsovateľnú HTML štruktúru, a získavali sa dáta z API, vyzerali príkazy na získanie informácií o produktoch nasledovne:

```
import requests
r=requests.get(url)
data = r.json()
```

Dôležitou súčasťou algoritmu sú metódy `find()` a `find_all()`. Tieto metódy umožňujú vyhľadávanie na základe názvu HTML tagu (napr. `<p>`, `<div>`, ``...), atribútov daného tagu (napr. `class`, `id`, `href`...), obsahu vybraného HTML elementu alebo podľa rôznych kombinácií spomenutých vlastností. Metóda `find_all()` vracia všetky výskyty, ktoré majú nami zadané vlastnosti. Metóda `find()` má len jeden takýto výstup. (RICHARDSON, 2020)

Ďalším spôsobom získavania dát podľa HTML štruktúry je okrem metód `find()` a `find_all()` aj metóda `get()`. Rozdiel je v tom, že metóda `get()` získa to, čo je obsahom vlastnosti HTML tagu. Napríklad výstupom HTML ele-

mentu `<li data-id="169321" data-brand="Estee Lauder">` po použití metody `get()`, je „Estee Lauder“. Detailnejšie je rozdiel medzi týmito metódami popísaný v **tabuľke 4.1**.

Tabuľka 4.1 Porovnanie metód
`find_all`, `find` a `get` (vlastné spracovanie)

<code>find_all()</code>	<pre>products = soup.find_all('li', class_='item')</pre>	získa všetky elementy <code></code> , ktoré majú triedu s názvom "item"
<code>find()</code>	<pre>price = soup.find('div', class_='price')</pre>	získa 1 element <code><div></code> , ktorý má triedu s názvom "pagination"
<code>get()</code>	<pre>brand = p.get("data-brand")</pre>	získa hodnotu vlastnosti "data-brand" elementu <code><p></code>

4.7 Problémy spojené s web scrapingom

Každý web je iný, a teda HTML štruktúra každého webu musí byť analyzovaná samostatne. V prípade, že HTML nie je ľahko parsovateľné, je nutné nájsť spôsob, ako požadované dáta z webovej stránky získať, pričom väčšinou je to pomocou API, ktoré takéto weby poskytujú. V takomto prípade je opäť nutné analyzovať a parsovať jednotlivé JSON dokumenty pre každý web samostatne, keďže atribúty môžu mať iné mená a celý JSON dokument môže mať úplne inú štruktúru. Spomenuté rozdiely HTML štruktúry a JSON dokumentov sa odzrkadľujú aj na zmenách celého web scrapera, keďže každá stránka bude vyžadovať iné metódy, iné atribúty a pod. Môže sa stať aj to, že weby sa budú web scrapingu brániť, a tak bude (takmer) nemožné získať z takýchto webov informácie, ktoré potrebujeme. (BREUSS, 2021)

Ďalším problémom, ktorý je spojený s web scrapingom je to, že webové stránky sa môžu meniť. Môžu začať používať nové technológie, zmeniť HTML štruktúru (len niektoré elementy alebo celkovo) alebo môžu webové stránky (e-shopy) úplne zaniknúť. Väčšina takýchto zmien je ľahko zaznamenateľná

a aktualizácia web scraperu nie je náročná, no treba myslieť na to, že internet je veľmi dynamický, a tak sa s niektorým z týchto problémov stretne každý web scraper. (BREUSS, 2021)

Pri sťahovaní dát o produktoch z rôznych e-shopov môžu nastať situácie, kedy sú rovnaké vlastnosti produktov pomenované rôznym spôsobom. Konkrétne môže ísť o to, že na jednom zdroji ide o kategóriu "Lícenka" a na druhom sa táto kategória volá "Lícenky". Tomu je potrebné sa vyhnúť, resp. to nejak vyriešiť. Najjednoduchší spôsob je použitie knižnice fuzzywuzzy. Táto knižnica slúži na párovanie reťazcov na základe Levensteinovej vzdialenosti a počíta rozdiely medzi dvomi reťazcami. Príklady použitia sú uvedené nižšie. V prípade web scraperu, ktorý je produktom tejto bakalárskej práce, sa logika knižnice fuzzywuzzy využívala tak, že sa určili "vzorové kategórie" (z e-shopu Notino) a cieľom bolo spárovať tieto "vzorové kategórie" s kategóriami ostatných e-shopov. Určila sa podmienka, ktorá porovnávala ratio ("podobnosť") dvoch názvov kategórií s nejakou hodnotou (po analýze rôznych hodnôt bola najvhodnejšia hodnota 77). Ak ku spárovaniu nedošlo, pridala sa táto kategória do kolekcie "referenčných kategórií". V liste "referenčných kategórií" sa nachádzali aj, už spomínané, "vzorové kategórie". Na záver sa aktualizovali kategórie všetkých produktov tak, aby napr. všetky produkty kategórie "Lícenka" mali kategóriu "Lícenky" a podobne. To viedlo k zúženiu počtu kategórií a k jednoduchšiemu, užívateľsky prívetivejšiemu vyhľadávaniu. Obdobným spôsobom boli vyriešené značky produktov, keďže tam nastali tiež situácie, že na jednom e-shope bola značka pomenovaná "Estee Lauder", na inom "Esté Lauder" a podobne.

```
Ratio1 = fuzz.ratio("Riasenka", "Riasenky")
Ratio2 = fuzz.ratio
    ("Paleta očných tieňov", "Paletky očných tieňov")
Ratio3 = fuzz.ratio("Lícenka", "Lícenky")
#Ratio1=88, Ratio2=93, Ratio3=86
```

4.8 Databáza

V nasledujúcich podkapitolách sú popísané procesy súvisiace s výberom databázového systému, skladovaním získaných dát, spôsobom uloženia a pod.

4.8.1 Výber databázového systému

Akonáhle je hotový algoritmus na získavanie vybraných informácií o produktoch z e-shopov, prichádza na rad vybrať spôsob ukladania týchto dát. Dáta je možné ukladať do súborov formátov ako napríklad csv alebo json, no oveľa lepší prístup je ukladanie týchto dát do databázy. Vzhľadom na to, že pri aplikácii, ako je porovnávač/vyhľadávač kozmetických produktov, ide o pomerne veľké množstvo dát, ktoré sa budú často meniť (najmä parameter cena), je veľmi vhodnou databázou databáza MongoDB (PARKER, 2013). Výhodou MongoDB je aj to, že dáta sú ukladané vo formáte JSON (respektíve BSON – binárny JSON), a tak sa s nimi jednoducho pracuje.

4.8.2 Naplnenie databázy dátami

Po výbere databázového systému sú informácie, ktoré boli získané pomocou web scraperu, uložené do databázy. Existuje na to príkaz `insert_many(list)`. Pri tomto príkaze nastáva problém, že dáta sú pri každom spustení programu plnené znova, a tak vznikajú duplicitné záznamy. Preto sa tento príkaz použije len pri prvom naplnení databázy. Pri každom ďalšom spustení programu je vhodnejšie použiť príkaz `update_one()`, ktorého konkrétna ukážka je v zdrojovom kóde nižšie. Dôležitý je parameter `"upsert=True"`. Tento parameter zabezpečí to, že v prípade, že dokument už v databáze existuje, len ho aktualizuje v prípade zmien, a ak tam daný dokument ešte nie je uložený, tak ho vytvorí. Postup pri prevolávaní funkcie `update_one()` je taký, že najprv sa získajú už existujúce produkty z databázy, ktoré sa uložia do listu produktov. Následne sú na základe atribútu `_id` postupne (pomocou for cyklu) aktualizované, resp. vznikajú nové dokumenty (vďaka parametru `upsert=True`).

Ukážku toho, ako vyzerajú produkty uložené v databáze je možné vidieť na **obrázku 4.4**.

```
import pymongo

conn_string=
'mongodb+srv://mongo:1234@cluster0.9xrn6.mongodb.net/test'
client=pymongo.MongoClient(conn_string)

mydb=client['Products']
data=mydb.products
...
#príkazy na vloženie do databázy
products=db.find({})
for p in products:
    db.update_one({"_id": ObjectId(p["_id"])},
                  { "$set":
                    {"Name": p["Name"],
                     "Brand": p["Brand"],
                     "Price": p["Price"],
                     "Category": p["Category"],
                     "Url": p["Url"],
                     "Image": p["Image"],
                     "Eshop": "pinkpanda"}}},
                  upsert=True)
```

Vzhľadom na to, že pri načítaní webovej stránky je nutné načítať všetky kategórie a značky (do rozbaľovacích zoznamov), bolo výhodné vytvoriť si kolekcie "categories" a "brands", ktoré budú obsahovať jedinečné (distinct) kategórie a značky produktov. To zabezpečí rýchlejšie načítanie stránky. Ak by sa mali tieto údaje načítavať z kolekcie "products", kde sú tisícky záznamov, načítavanie by trvalo viac ako tri sekundy, ktoré sa pokladajú za maximálnu prijateľnú dobu načítania (MEDIAGURU, 2018).

4.8.3 Nástroj MongoDB Compass

MongoDB Compass je voľne dostupné grafické užívateľské rozhranie (GUI), ktoré umožňuje analyzovať, agregovať a vyhľadávať údaje o jednotlivých dokumentoch databázy MongoDB. Ukazuje všetky dáta uložené do databázy prehľadne. Spôsob zobrazenia je buď ako štruktúrovaný JSON dokument, alebo sú tieto údaje naformátované do tabuľky. Medzi agregácie, ktoré umožňuje MongoDB Compass patrí napríklad \$sort, \$count, \$match, \$sample a podobne. (MONGODB, 2021) Tento nástroj bol často využívaný na sledovanie kolekcií, dokumentov a zmien v databáze.

```
{
  "_id": {
    "$oid": "6274dab2258ca3a8becae67d"
  },
  "Brand": "Revolution",
  "Name": "Revolution dvojdielna ceruzka - Felt and Kohl",
  "Price": 3.99,
  "Category": "Ceruzky na obočie",
  "Url": "https://www.pinkpanda.sk/produkt/revolution-dvojdielna-ceruzka-felt-and-kohl",
  "Image": "/licilasicdn.s3.amazonaws.com/public/product_images/5706/main/original.jpg?ver=202107071642",
  "Eshop": "pinkpanda"
},
{
  "_id": {
    "$oid": "6274daee43f78745632239e6"
  },
  "Brand": "Lancome",
  "Name": "Lash Idle",
  "Price": 24.99,
  "Category": "Riasenky",
  "Url": "https://www.douglas.sk/p/lancome-lash-idole-687121",
  "Image": "https://images.douglas.cz/687121/150/687121.jpg",
  "Eshop": "douglas"
},
{
  "_id": {
    "$oid": "6269827eb16e65cdd233a2d8"
  },
  "Brand": "Rimmel",
  "Name": "Stay Matte",
  "Price": 2.5,
  "Category": "Púdre",
  "Url": "https://www.notino.sk/rimmel/stay-matte-puder/",
  "Image": "https://cdn.notinoimg.com/detail_thumb/rimmel/3607345064505_01-o__24.jpg",
  "Eshop": "notino"
}
```

Obrázok 4.4

Ukážka dát v databáze (vlastné spracovanie)

4.9 Návrh a tvorba webovej aplikácie

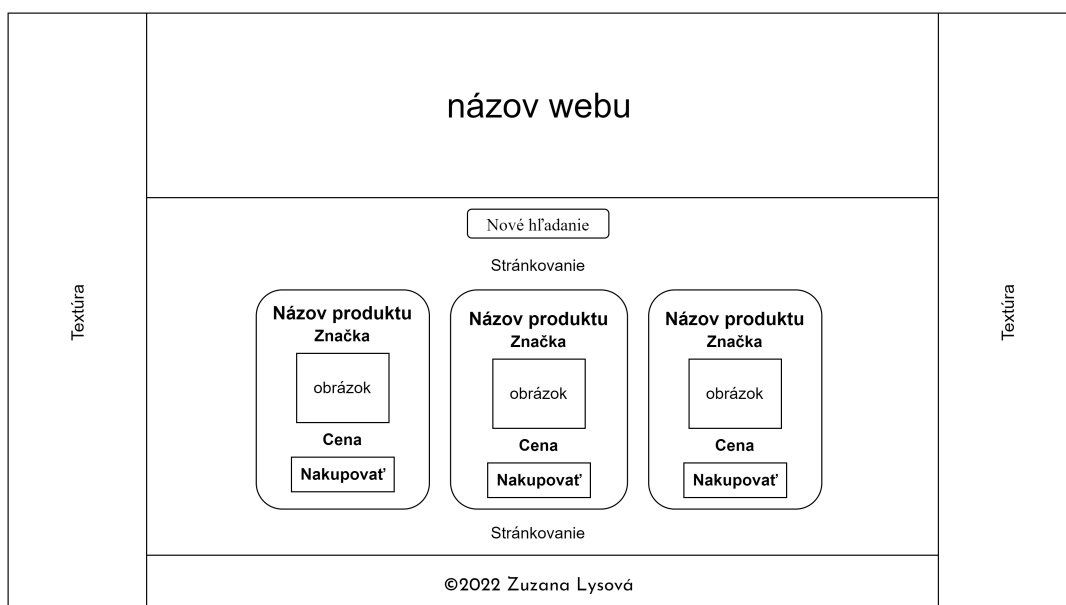
V momente, keď je hotový web scraper a dáta sú uložené v databáze, nastáva fáza tvorby webovej aplikácie. V rámci návrhu boli vytvorené drôtené modely pomocou nástroja draw.io (viď **obrázky 4.5, 4.6, 4.7 a 4.8**) a kompletný grafický návrh pomocou aplikácie Affinity Designer (viď **obrázky 4.9, 4.10, 4.11 a 4.12**).

Vzhľadom na to, že v dnešnej dobe množstvo ľudí surfuje na webe zo smartfónov, web je navrhovaný a implementovaný responzívne. Podľa ecommerce štatistík, ktoré uvádza SMITH (2022), až 79 % užívateľov smartfónov urobilo za posledných šesť mesiacov online nákup cez mobilný telefón.

Na tvorbu webu bol zvolený programovací jazyk Python, framework Flask a šablónovací systém Jinja. Web pozostáva z hlavnej stránky, kde je názov webu a vyhľadávací formulár, ktorý obsahuje vstupné pole (<input>) pre vyhľadávanie podľa názvu produktu a rozbaľovacie zoznamy (<select> a <option>) na vyhľadávanie podľa kategórie alebo značky. Po kliknutí na tlačidlo s lupou je užívateľ presmerovaný na stránku s produktmi.

**Obrázok 4.5**

Drôtený model – úvodná stránka – veľká obrazovka (vlastné spracovanie)

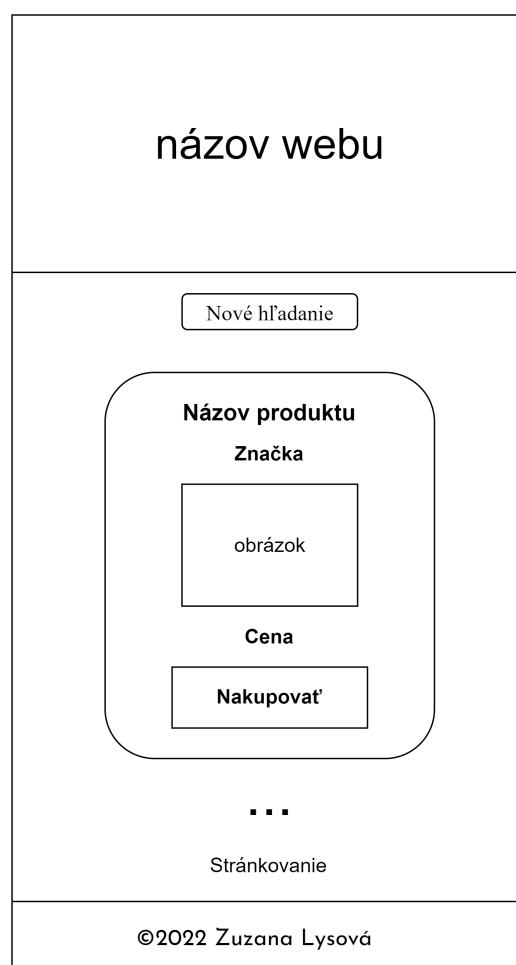
**Obrázok 4.6**

Drôtený model – podstránka – veľká obrazovka (vlastné spracovanie)

Na stránke s produktmi užívateľ vidí mriežku produktov, ktoré vyhovujú zvoleným kritériám. Obsahuje informácie o produktoch, konkrétne názov, značku, obrázok, cenu a tlačidlo s logom e-shopu, z ktorého daný produkt je. Tlačidlo slúži na presmerovanie na konkrétnu ponuku e-shopu. Produkty sú radené podľa ceny vzostupne.

**Obrázok 4.7**

Drôtený model – úvodná stránka – malá obrazovka (vlastné spracovanie)

**Obrázok 4.8**

Drôtený model - podstránka - malá obrazovka (vlastné spracovanie)

Čo sa týka súborovej štruktúry, webová aplikácia obsahuje niekoľko súborov formátu py, ktoré sa starajú o funkcionality stránky. Konkrétne ide o súbor `mongo.py`, ktorého úlohou je pripojenie ku kolekciám a dokumentom databázy pomocou connection stringu a získanie dát z databázy pomocou niekoľkých funkcií, ako sú napríklad funkcie na získanie produktov (`get_products`), na získanie všetkých značiek (`get_brands`) a pod. Druhý súbor formátu py je súbor `app.py`, ktorý je koreňom celej aplikácie, ktorá sa spúšťa na príkazovom riadku pomocou príkazu `"python app.py"`. Tento súbor rieši routing, queries a prakticky všetky funkcie, ktoré webová aplikácia obsahuje. Úzko súvisí aj s funkciami súboru `mongo.py`. Rieši sa tu celková logika vyhľadávania produktov, či už podľa značky, kategórie, alebo názvu produktu. Ďalšie súbory webovej aplikácie už súvisia s vykresľovaním v prehliadači a patria sem šablóny (súbory `html`) a statické súbory (obrázky a súbor `styles.css`).

**Obrázok 4.9**

Grafický návrh – úvodná stránka – veľká obrazovka (vlastné spracovanie)

**Obrázok 4.10**

Grafický návrh – podstránka – veľká obrazovka (vlastné spracovanie)

**Obrázok 4.11**

Grafický návrh – úvodná stránka – malá obrazovka (vlastné spracovanie)

**Obrázok 4.12**

Grafický návrh – podstránka – malá obrazovka (vlastné spracovanie)

Vyhľadávanie produktov podľa značky a kategórie prebieha za pomoci select boxov, ktorých zvolená možnosť vstupuje to url adresy ako parameter. Čo sa týka vyhľadávania podľa názvu produktu, to prebieha prostredníctvom inputu, z ktorého sa vytvorí regulárny výraz `regex = re.compile('.*' + request.args['name'] + '.*', re.IGNORECASE)`, ktorý sa porovná s názvami produktov v databáze. To zaisťuje to, že užívateľ nie je povinný napísať názov produktu doslovne, ale stačí, že použije aspoň časť názvu. V prípade, že užívateľ nezadá žiaden vstup do inputu a ani nevyberie žiadne položky zo selectu, kliknutie na tlačidlo s lupou vráti všetky produkty, ktoré sú uložené v databáze.

5 Diskusia a záver

Aplikácia, ktorá je predmetom tejto práce umožňuje užívateľom vyhľadávať kozmetické produkty podľa názvu, značky alebo kategórie produktu. Obsahuje databázu produktov z niekoľkých slovenských e-shopov, ktoré sú zobrazené užívateľom podľa zvolených kritérií. Vzhľadom na to, že na Slovensku ešte neexistuje porovnávač, ktorý by sa zaoberal kozmetickými e-shopmi a produktmi, dá sa povedať, že je to prvý porovnávač cien kozmetických produktov na Slovensku. V zahraničí síce takýto porovnávač existuje (cosmetify.com), no ten neobsahuje ponuku slovenských e-shopov.

Webová aplikácia je responzívna, takže umožňuje ľahkú prácu na väčších obrazovkách, ako notebook alebo monitor, ale aj na menších zariadeniach, ako mobil alebo tablet.

Aj napriek tomu, že je aplikácia funkčná a je schopná prevádzky, existujú možné rozšírenia, ktoré by aplikáciu mohli vylepšiť. Tieto návrhy na vylepšenia budú popísané v ďalšej podkapitole.

5.1 Návrhy na vylepšenie

Aplikácia aktuálne sťahuje a prezentuje dáta len zo slovenských kozmetických e-shopov a ceny sú v eurách, no keďže väčšina e-shopov má slovenskú aj českú verziu, dala by sa použiť pre obe krajiny a obe meny (eurá aj české koruny).

Momentálne sa aplikácia zaoberá len oblasťou dekoratívnej kozmetiky, čo by sa dalo rozšíriť aj na oblasť starostlivosti o pleť, a teda telovú a pleťovú kozmetiku. Taktiež by mohol byť porovnávač rozšírený o ďalšie e-shopy, ktoré zatiaľ nie sú zahrnuté.

Významným rozšírením by mohlo byť pridanie možnosti recenzovania produktov. Tým pádom by užívatelia vedeli na stránke nájsť aj to, ako produkt hodnotia ostatní ľudia, ktorí už daný produkt vyskúšali, a tak sa bude môcť rozhodovať o kúpe nie len na základe ceny, ale aj na základe hodnotenia.

Funkcia, ktorá by mohla výrazne uľahčiť vyhľadávanie je našepkávač pri vstupnom poli pre vyhľadávanie podľa názvu. Užívateľ by tak nemusel písať celý názov produktu, ale našepkávač by mu napovedal už pri zadaní napr. troch písmen, ktoré produkty by sa mohli zhodovať z jeho zámerom hľadania.

5.2 Záver

V práci boli postupne prevádzané kroky, ktoré viedli k splneniu cieľa a zadania práce. Medzi tieto kroky patrí analýza dostupných porovnávačov produktov na súčasnom slovenskom (českom) trhu a identifikovanie internetových obchodov, ktoré sa zaoberajú predajom kozmetických produktov. V ďalšom priebehu bol vytvorený web scraper na získanie informácií o produktoch, potrebných na vytvorenie porovnávača vlastností kozmetiky. Následne bol vytvorený dátový model a na základe návrhu bola vytvorená databáza produktov, ktorá bola postupne plnená relevantnými dátami. Potom nasledovalo vytvorenie návrhu a implementácia webovej aplikácie "porovnávač vlastností kozmetických produktov". Na záver bola overená funkčnosť tejto aplikácie potenciálnymi budúciimi užívateľmi.

Je teda možné skonštatovať, že sa podarilo splniť cieľ práce, keďže cieľom práce bolo vytvoriť aplikáciu, ktorá bude schopná získať informácie (názov, značka, cena a pod.) o kozmetických produktoch z rôznych slovenských kozmetických e-shopov, uložiť tieto informácie do databázy a vytvoriť webovú aplikáciu, kde budú tieto informácie poskytnuté užívateľom. Taktiež tým bolo splnené aj zadanie práce.

Webová aplikácia je, pomocou webhostingovej služby PythonAnywhere, dočasne dostupná na adrese: <http://sparkleweb.pythonanywhere.com/>.

Literatúra

- BERÁNEK, L., REMEŠ, R. Modelling E-Commerce Processes in the Presence of a Price Comparison Site [on-line]. In *The International Scientific Conference INPROFORUM 2017*. České Budějovice : University of South Bohemia, 2017, s. 327–332 [cit. 2022-01-14]. (ISBN 978-80-7394-667-8.) Dostupné na: <http://ocs.ef.jcu.cz/files/conferences/2/schedConfs/30/papers/972/public/972-1781-1-PB.pdf>.
- BREUSS, M. Beautiful Soup: Build a Web Scraper With Python [on-line]. In *Real Python*. 2021 [cit. 2022-04-10]. Dostupné na: <https://realpython.com/beautiful-soup-web-scraper-python/>.
- CARBONELLE, P.. *PYPL Popularity of Programming Language* [on-line]. 2022. [cit. 2022-02-26]. Dostupné na: <https://pypl.github.io/PYPL.html>.
- ČÁPKA, D. UML – Online kurz [on-line]. In *ITnetwork*. 2021 [cit. 2022-04-10]. Dostupné na: <https://www.itnetwork.sk/navrh/uml>.
- DOUGLAS. *Líčenie a dekoratívna kozmetika* [on-line]. ©2022. [cit. 2022-05-05]. Dostupné na: <https://www.douglas.sk/c/licenie-4003/>.
- HEUREKA. *Heureka v kostce* [on-line]. ©2007-2022. [cit. 2022-02-20]. Dostupné na: <https://www.heureka.cz/>.
- HEYDT, M. *Python Web Scraping Cookbook*. Birmingham : Packt Publishing, 2018. 366 s. ISBN 978-1-78728-521-7.
- HORCH, A., KETT, H., WEISBECKER, A. Web Information Systems and Technologies : Extracting Product Offers from e-Shop Websites [on-line]. In *11th International Conference on Web Information Systems and Technologies*. Cham : Springer, 2016 [cit. 2021-10-26]. (ISBN 978-3-319-30996-5.) Dostupné na: https://link.springer.com/chapter/10.1007/978-3-319-30996-5_12.
- KROTOV, V. A KOL. Tutorial : Legality and Ethics of Web Scraping. [on-line]. In *Communications of the Association for Information Systems*. Cham : Springer, 2020 [cit. 2021-10-26]. (ISBN 978-3-319-30995-8.) Dostupné na: <https://digitalcommons.murraystate.edu/cgi/viewcontent.cgi?article=1071&context=faculty>.
- LEMAN, M. Gentle white flowers in blossom in studio [on-line]. In *Pexels*. 2021 [cit. 2022-04-10]. Dostupné na: <https://www.pexels.com/photo/gentle-white-flowers-in-blossom-in-studio-6774558/>.
- LUBANOVIC, B. Introducing Python : Python in the Real World [on-line]. In *Modern Computing in Simple Packages*. Sebastopol : O’Riley Media, Inc, 2015 [cit. 2022-02-26]. (ISBN 978-1-449-35936-2.) Dostupné na: https://www.academia.edu/39376646/Introducing_Python.

- MAKEUP. *Dekoratívna kozmetika* [on-line]. ©2019-2022. [cit. 2022-05-05]. Dostupné na: <https://makeup.sk/categorys/2419/>.
- MAITHANI, M. Scrape Beautifully With Beautiful Soup In Python [on-line]. In *Developers Corner*. 2020 [cit. 2022-04-10]. Dostupné na: <https://analyticsindiamag.com/beautiful-soup-webscraping-python/>.
- MEDIAGURU Rychlost webu je pro úspěch na internetu stále důležitější [on-line]. In *MediaGuru*. 2018 [cit. 2022-05-05]. Dostupné na: <https://www.mediaguru.cz/clanky/2018/06/rychlost-webu-je-pro-uspech-na-internetu-stale-dulezitejsi/>.
- MONGODB, INC. What is MongoDB Compass? [on-line]. In *MongoDB*. 2021 [cit. 2022-04-09]. Dostupné na: <https://www.mongodb.com/docs/compass/current/>.
- NAJNAKUP. *Štatistiky* [on-line]. ©2007-2022. [cit. 2022-02-20]. Dostupné na: <https://www.najnakup.sk/statistiky>.
- NOTINO. *Makeup a dekoratívna kozmetika* [on-line]. ©2022. [cit. 2022-05-05]. Dostupné na: <https://www.notino.sk/kozmetika/dekorativna-kozmetika/>.
- PALLETS Flask Documentation [on-line]. In *Pallets Projects*. ©2010 [cit. -04-18]. Dostupné na: <https://flask.palletsprojects.com/en/2.1.x/>.
- PARKER, Z., SCOTT, P., VRBSKY, S. Comparing NoSQL MongoDB to an SQL DB [on-line]. In *ACMSE'13. ACMSE '13 : Proceedings of the 51st ACM Southeast Conference, 2013* [cit. 2022-04-01]. (ISBN 9781450319010.) Dostupné na: <https://dl.acm.org/doi/pdf/10.1145/2498328.2500047>.
- PINKPANDA. *Pink Panda* [on-line]. ©2022. [cit. 2022-05-05]. Dostupné na: <https://www.pinkpanda.sk/>.
- PRICEMANIA. *O nás* [on-line]. ©2007-2022. [cit. 2022-02-20]. Dostupné na: <https://www.pricemania.sk/o-nas/>.
- QUDUS KHAN, F. A KOL. Smart algorithmic based web crawling and scraping with template autoupdate capabilities [on-line]. In *Concurrency and Computation: Practice and Experience*. Cham : Springer, 2020 [cit. 2021-10-26]. (ISBN 978-3-319-30995-8.) Dostupné na: <https://onlinelibrary.wiley.com/doi/10.1002/cpe.6042>.
- RÁBOVÁ, I. *Podnikové informační systémy a technologie jejich vývoje*. Brno : Mendelova univerzita v Brně, 2008. 60 s. Učebný text.
- RICHARDSON, L. Beautiful Soup Documentation [on-line]. In *Crummy*. 2020 [cit. 2022-03-31]. Dostupné na: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/#beautiful-soup-documentation>.
- SAWYER, E Metodika tvorby webových aplikací [on-line]. In *robime.it*. 2016 [cit. 2022-04-11]. Dostupné na: <https://robime.it/metodika-tvorby-webovych-aplikacii/>.
- SHANTANU, S. FuzzyWuzzy Python library [on-line]. In *GeeksForGeeks*. 2017 [cit. 2022-05-05]. Dostupné na: <https://www.geeksforgeeks.org/fuzzywuzzy-python-library/>.

- SMITH, J. Mobile eCommerce Stats in 2022 and the Future Online Shopping Trends of mCommerce [on-line]. In *Outerbox*. 2022 [cit. 2022-04-27]. Dostupné na: <https://www.outerboxdesign.com/web-design-articles/mobile-e-commerce-statistics>.
- STACKOVERFLOW. *Developer Survey* [on-line]. 2021. [cit. 2022-02-26]. Dostupné na: <https://insights.stackoverflow.com/survey/2021#most-popular-technologies-language>.
- ŠSTATISTICKÝ ÚRAD SR Internet commerce for private use [on-line]. In *STATdat*. 2021 [cit. 2022-04-27]. Dostupné na: [http://statdat.statistics.sk/cognosext/cgi-bin/cognos.cgi?b_action=cognosViewer&ui.action=run&ui.object=storeID\(%22iDCA3C238F40D41D1AB4D5389123555AA%22\)&ui.name=Internet%20commerce%20for%20private%20use%20%5Bis1002rs%5D&run.outputFormat=&run.prompt=true&cv.header=false&ui.backURL=%2Fcognosext%2Fcps4%2Fportlets%2Fcommon%2Fclose.html](http://statdat.statistics.sk/cognosext/cgi-bin/cognos.cgi?b_action=cognosViewer&ui.action=run&ui.object=storeID(%22iDCA3C238F40D41D1AB4D5389123555AA%22)&ui.name=Internet%20commerce%20for%20private%20use%20%5Bis1002rs%5D&run.outputFormat=&run.prompt=true&cv.header=false&ui.backURL=%2Fcognosext%2Fcps4%2Fportlets%2Fcommon%2Fclose.html).
- WATSON, J. H. Tutorial: Big Data Analytics: Concepts, Technologies, and Applications. *Communications of the Association for Information Systems*, 2014, roč. 34, č. 65, s. 1248–1268.

Zoznam tabuliek

2.1	Popularita programovacích jazykov podľa GitHubu (CARBONNELLE, 2022)	15
2.2	Popularita programovacích jazykov podľa StackOverflow (STACKOVERFLOW, 2022)	15
4.1	Porovnanie metód find_all, find a get (vlastné spracovanie)	24

Zoznam obrázkov

1.1	Graf percenta internetových nákupov na Slovensku (spracované podľa údajov Štatistického úradu SR (2021))	10
3.1	Vodopádový model (vlastné spracovanie)	17
4.1	Use case diagram (vlastné spracovanie)	20
4.2	ERD diagram (vlastné spracovanie)	21
4.3	Ukážka štruktúry dát v databáze (vlastné spracovanie)	21
4.4	Ukážka dát v databáze (vlastné spracovanie)	28
4.5	Drôtený model – úvodná stránka – veľká obrazovka (vlastné spracovanie)	29
4.6	Drôtený model – podstránka – veľká obrazovka (vlastné spracovanie)	29
4.7	Drôtený model – úvodná stránka – malá obrazovka (vlastné spracovanie)	30
4.8	Drôtený model - podstránka - malá obrazovka (vlastné spracovanie)	30
4.9	Grafický návrh – úvodná stránka – veľká obrazovka (vlastné spracovanie)	31
4.10	Grafický návrh – podstránka – veľká obrazovka (vlastné spracovanie)	31
4.11	Grafický návrh – úvodná stránka – malá obrazovka (vlastné spracovanie)	32
4.12	Grafický návrh – podstránka – malá obrazovka (vlastné spracovanie)	32