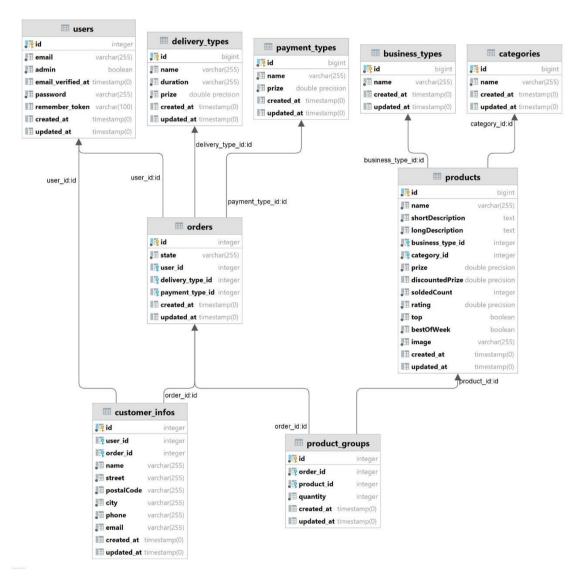
Zadanie

Vytvorte webovú aplikáciu - eshop, ktorá komplexne rieši nižšie definované prípady použitia vo vami zvolenej doméne. Presný rozsah a konkretizáciu prípadov použitia si dohodnete s Vašim cvičiacim na cvičení. Ako tému pre náš e-shop sme si vybrali horolezectvo. Od pôvodnej verzie, ktorú sme si načrtli vo figme sme sa odklonili len minimálne ako aj návrh našej. Webstránka nášho e-shopu bude plne responzívne to znamená, že používatelia na mobilných zariadeniach, tabletoch, notebookoch vidia cely obsah ktorý je pre nich príslušne zmenšený, zväčšený alebo premiestnený na inú pozíciu. Čo sa týka prípadov použitia, tie sme splnili všetky a niektoré z nich sú aj bližšie opísané nižšie.

Fyzický model



lde o fyzický dátový model, ktorý používame už aj vo finálnej verzií. Medzi dve hlavné tabuľky môžeme považovať *products* a *orders*. Každý produkt môže byť uložený do tzv. skupiny, ktorá je priradená k objednávke a obsahuje aj množstvo daného produktu. Obsahuje taktiež značku a kategóriu. Na druhej strane tabuľka *orders* predstavuje všetky objednávky v systéme. Ku každej objednávke sú priradené informácie o nakupujúcom. Objednávka ma rôzne typy dopravy a platby. Tabuľka *users* obsahuje základne prihlasovacie údaje a zvyšné informácie sú uložené v tabuľke *customer_infos*. Rozdiel od návrhu je v odstránení tabuľky s obrázkami, ktoré sme nahradili jedným stĺpcom v tabuľke *products* s cestou k obrázku.

Programovacie prostredie

Projekt bol naprogramovaný pomocou HTML, CSS, PHP a JS. Pri vyváraní obrazoviek boli použité knižnice ako Bootstrap, Laravel. Na ukladanie dát sme použili databázu PostgreSQL na administráciu tejto databázy sme použili pgAdmin. Na písanie programu sme využívali prostredie od spoločnosti JetBrains PHPStorm. Na verziovanie a zdieľanie projektu sme použili GitHub.

Ako externú knižnicu sme použili **Image intervention**. Ide o knižnicu, ktorá slúži na zmenu veĹkosti obrázkov. A zároveň sme použili **Breeze** pre manažement používateľov.

Opis riešenia

Tvorenie projektu prebiehalo v troch hlavných fázach a v jednom kontrolnom bode.

V prvej fáze sme si vytvorili šablóny v programe figma ako bude náš e-shop vyzerať.

V druhej fáze sme si nami vytvorené šablóny spravili responzívne. Implementovali sme databázu vytvorili modely a controllery. Naša webstránka je postavená na volaniach cez Routes ktoré ak bolo potrebne prenášali aj argumenty. Pomocou crud funkcii sme data na našej stránke zobrazovali upravovali a ďalej spracovávali. V controlleroch boli tieto crud požiadavky presmerovane do funkcii a následne volali databázu.

V tretej časti sme implementovali admin rozhranie. Toto admin rozhranie slúži na pridávanie, upravovanie a odstraňovanie produktov z nášho e-shopu.

Stručný opis implementácie vybraných prípadov použitia Základné filtrovanie

Filtrovať produkty podľa zadaných parametrov je možné na obrazovke zobrazujúcej všetky produkty. Filtrovať sa da podľa ceny, značky, druhu výstroje. V controlleri sa použijú hľadané informácie z requestu a následne sa z databázy sa vyfiltrujú len požadované objekty. Takto vyfiltrované produkty sa vratia na stránku spolu s filtrami. Týmto sa zabezpečili, že používateľovi ostanú zaškrtnuté filtre. Ak ich chce však vymazať, stačí v hlavičke sa presmerovať na produkty a začne sa bez filtrov.



Stránkovanie

Stránkovanie sa nachádza na obrazovke, ktorá zobrazuje všetky produkty alebo vyfiltrovanú časť týchto produktov. Na jednu stránku sa zmesti 6 produktov. Stránkovanie je implementovane pomocou funkcie pagination, ktorú ponuka Laravel pri výberoch z databázy. Medzi jednotlivými stránkami je možné sa presúvať pomocou tlačidiel predošli, ďalší alebo kliknutím na číslo stránky. Avšak tieto tlačidla sú viditeľne len ak nasledujúca/predošlá stránka existuje. Pri tvorbe pagination sme príkazom vytvorili príslušne blade súbory a tieto sme si upravili podľa nás.

Pridanie do košíka

Do košíka je možné pridať ľubovoľný počet produktov ich počet sa upravuje buď manuálne alebo pomocnými šípkami. A odošle sa na back-end pomocou tlačidla. Ak už košík obsahuje daný produkt, tak kvantita tohto produktu bude zmenená na novú hodnotu. To do akého košíka je vložený produkt určujeme pomocou čísla objednávky uloženej v Session. Ak ide o quest používateľa, vytvorí sa nová objednávka a toto id sa uloží a teda neskôr použije. Ak ide o prihláseného používateľa, najskôr sa pozrieme do databázy, či už náhodou používateľ nemá vytvorenú objednávku. Ak áno tak ju použijeme, inak postupujeme rovnako ako pri quest používateľovi. Krajný prípad keď používateľ dva krát pridá ten istý produkt do košíku sme vyriešili tak, že sa použije vždy len to množstvo, ktoré vložil neskôr.

Zmena množstva pre daný produkt

V pravom hornom rohu každého produktu sa nachádza tlačidlo X, ktoré nám odstráni položku z košíka. Alebo ak chceme zmeniť kvantitu produktu, používame rovnakú logiku ako pri pridávaní produktu. Používateľ nastaví požadovanú kvantitu a pomocou tlačidla "OK" ho zmení. A na stránke sa aktualizujú ceny.

Vyhľadávanie

Searchbar sa nachádza v hlavičke stránku. Jeho funkcionalita spočíva vo vyhľadávaní rovnakých stringov v názvoch produktov. Veľkosť písmen sa neberie do úvahy. Ide teda o fulltextové vyhľadávanie nad tabuľkou *products*. Toto vyhľadávanie nepoužíva žiadne predtým definované filtrovanie.

Prihlásenie a registrácia

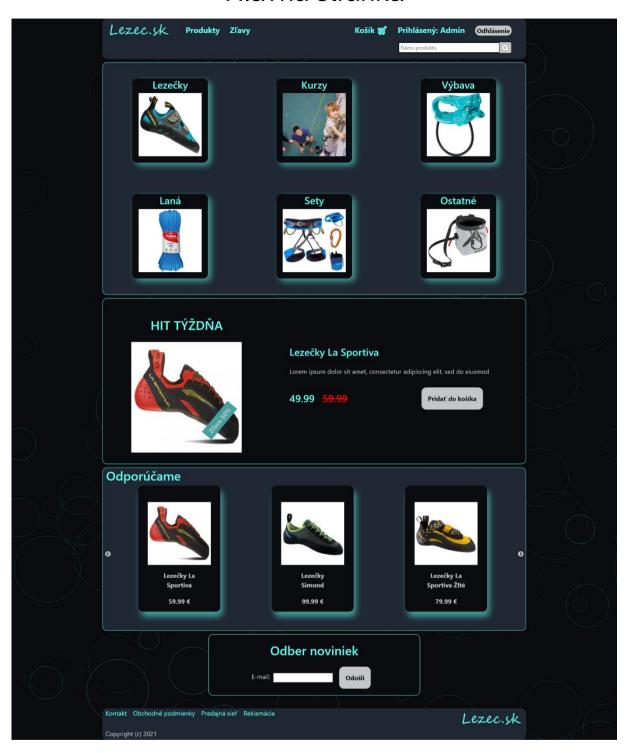
V hlavičke stránky sa nachádza odkaz s názvom prihlásenie. Po kliknutí na prihlásenie sa zobrazí obrazovka s prihlásením, kde sa od užívateľa vyžaduje email a heslo. Tieto údaje sú skontrolované s databázou registrovaných zákazníkov, ak sa zadané údaje zhodujú s údajmi registrovaného užívateľa, po kliknutí na prihlásenie je presmerovaný na hlavnú stránku, kde môže pokračovať v nákupe. Pokiaľ ešte nie je registrovaný môže kliknúť na text pod tlačidlom prihlásenie a registrovať sa.

Ak je užívateľ prihlásený zmení sa aj header stránky. Namiesto odkazu prihlásenie sa zobrazí meno užívateľa, ktorý je prihlásený a tlačidlo na odhlásenie. Ak by sa prihlásil admin, tento odkaz bude aktívny a dá sa s ním dostať na admin stránku. To že je používateľ admin, máme v databáze uložené ako flag admin.

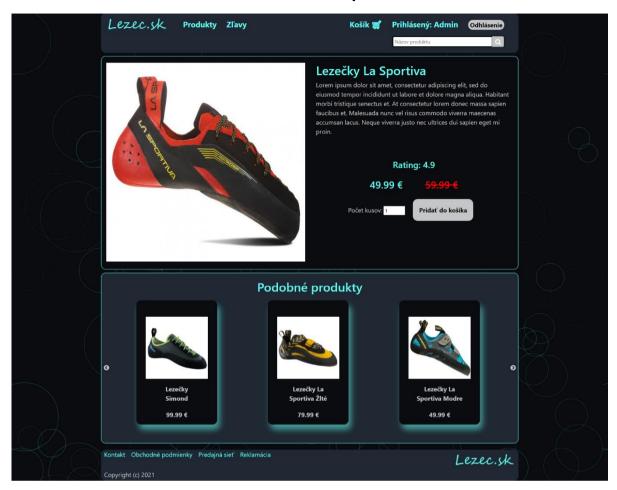
Editácia produktu adminom

Admin dokáže zeditovať každý produkt v databáze. Po kliknutí na edit v stĺpci akcie, sa mu zobrazí popup okno, v ktorom môže editovať atribúty aj obrázky. Môže nahrať 1 alebo až 4 obrázky pre produkt. Ak chce ale nejaký obrázok odstrániť, musí nahrať všetky obrázky nanovo, ale už bez nechceného obrázka. Takto sme vymysleli odstránenie obrázku pre produkty z dôvodu toho, že sme nepoužili zvlášť stránku, ale len popup okno. Usúdili sme, že ak by sme zrobili aj odstránenie obrázkov, vzniklo by veľa redundantného kódu.

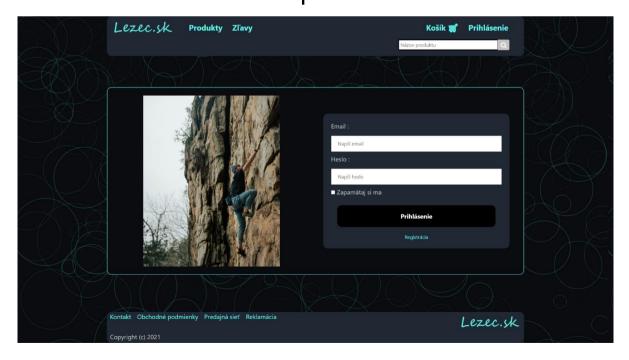
Konečná podoba webstránky Hlavná stránka



Stránka detailu produktu



Stránka prihlásenia



Stránka košíka

