

Semestrálny projekt - Elektronický obchod

Webové technológie

Autori: Jaroslav Ertl, Leon Radó

Slovenská Technická Univerzita v Bratislave

Fakulta informatiky a informačných technológií

Obsah

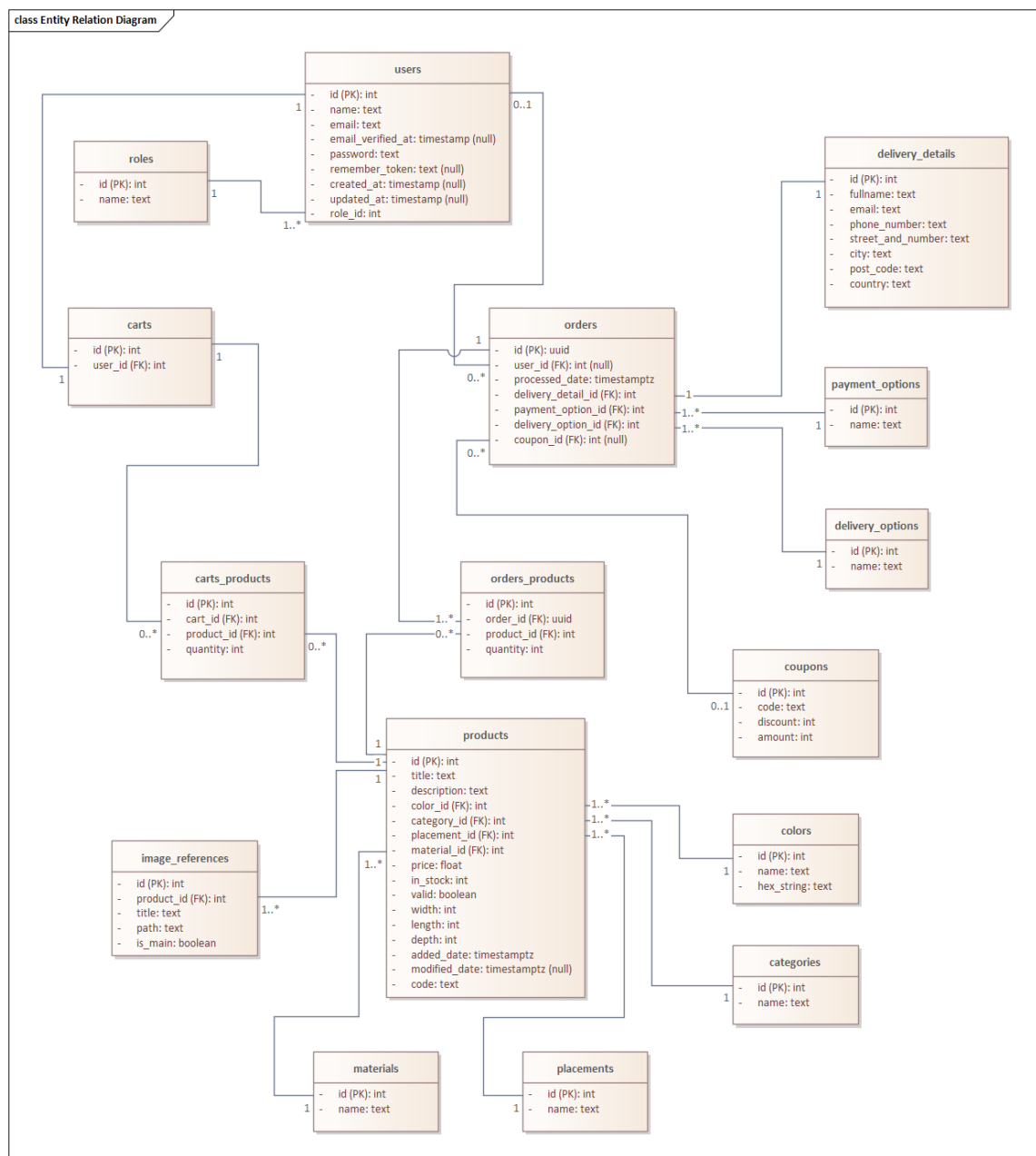
Obsah	2
Zadanie	2
Diagram fyzického dátového modelu	3
Návrhové rozhodnutia	4
Programovacie prostredie	4
Opis implementácie	4
Zmena množstva pre daný produkt	4
Prihlásenie	5
Vyhľadávanie, filtrovanie a usporiadanie	5
Stránkovanie	6
Pridanie produktu do košíka	6
Snímky obrazovky	6
Domovská stránka	6
Detail produktu	7
Košík s produktami	8
Prihlásenie	8
Návod na spustenie	9

Zadanie

Zadaním semestrálneho projektu bolo navrhnuť a vytvoriť webovú aplikáciu - eshop v konkrétnej doméne. Našou zvolenou doménou bol nábytok. Bolo treba implementovať 2 časti tejto aplikácie a to - klientskú a administrátorskú. Klientská časť umožňuje zákazníkovi prihlásiť, zaregistrovať a odhlásiť sa, zobrazíť produkty, vyhľadávať a filtrovať produkty, pridávať produkty do košíka a možnosť "objednať" produkty v košíku. Administrátorská časť umožňuje adminovi prihlásiť sa, odhlásiť sa, zobrazíť produkty, pridať produkt, odobrať produkt a upraviť produkt.

Zadaním bolo taktiež aj vytvoriť skice pre jednotlivé obrazovky, ktoré sa v našom eshope nachádzajú. Taktiež bolo potrebné vytvoriť fyzický dátový model pomocou UML class diagramu, ktorý reprezentuje schému databázy nášho eshopu.

Diagram fyzického dátového modelu



Vo fyzickom dátovom modeli došlo k niekoľkým malým zmenám, hlavne z dôvodu použitia knižnice Laravel Breeze na autentifikáciu. Rozhodli sme sa použiť pôvodnú tabuľku users, tak ako sa používa v knižnici Laravel Breeze, jediné, čo sme v nej upravili je pridanie stĺpca role_id.

Nakoľko táto tabuľka má id typu int, museli sme upraviť aj príslušné cudzie kľúče v tabuľkách carts a orders.

Posledná zmena bola v tabuľke products, zmena atribútu modified_date na nullable.

Návrhové rozhodnutia

Jediná pridaná knižnica je Laravel Breeze, ktorá bola použitá na autentifikáciu. Rolu sme riešili ako atribút používateľa, teda každý zaregistrovaný používateľ dostal rolu customer, ktorá obmedzovala jeho prístup. Zaregistrovať sa s rolou admin v aplikácii nie je možné. Takýto účet sa dá vytvoriť iba na pozadí. Pre administrátora je vytvorený účet s emailom 'admin@admin.com' a heslom 'admin123'.

Programovacie prostredie

Používané technológie v tomto projekte sú:

- PHP Laravel
- PostgreSQL
- Bootstrap CSS

Opis implementácie

Zmena množstva pre daný produkt

Zmena množstva produktu v košíku e-shopu je implementovaná v CartController pomocou metódy update a je zobrazená v cart.blade.php. Funkcionalita sa líši podľa toho či je zákazník prihlásený alebo nie. Ak je prihlásený tak:

- Načíta sa košík z modelu Cart podľa user_id, ktoré je pridelené danému zákazníkovi
- Následne sa na základe akcie od zákazníka zvýši alebo zníži množstvo produktu v tabuľke
- Po skončení sa aktualizuje množstvo produktu a aktualizuje sa množstvo pre daný produkt v košíku aj v databáze

Ak používateľ prihlásený nie je tak:

- Košík sa nenačítava z modelu, ale načíta sa zo session 'cart'
- Následne sa aktualizuje množstvo v session na základe akcie od zákazníka
- V tomto prípade sa nezmení nič v databáze, lebo sa s ňou nepracuje

V košíku sa zobrazí informácia s úspešnou alebo chybovou správou.

Tento proces je umožnený zákazníkovi pomocou cart.blade.php, ktorý zobrazí formulár s tlačítkami + a -, ktoré zákazníkovi umožnia meniť množstvo produktu v košíku. Tento formulár odošle POST požiadavku do CartControllera pre metódu update. Pole pre množstvo produktu v košíku je len readonly, takže sa nedá prepísať a dá sa meniť iba

pomocou + a - tlačidiel. Cena je zobrazená podľa ceny produktu a jeho množstva v košíku podľa $price * quantity$.

Prihlásenie

Prihlásenie je implementované v `AuthenticatedSessionController` (registrácia v `RegistredUserController`). Na prihlasovací formulár sa dá dostať odkazom v navigácii, alebo v prípade, že sa používateľ chce prihlásiť ako admin, tak zadaním `/admin-login` v URL. Používateľ musí zadať svoj email a heslo, na základe ktorého sa používateľ prihlási s konkrétnou rolou. Všetky cesty v aplikácii sú zabezpečené cez middleware, buď cez `AdminMiddleware`, alebo `CustomerMiddleware`, takže na základe role účtu sa určuje kam sa prihlásený používateľ vie dostať, a čo môže vykonávať.

Vyhľadávanie, filtrovanie a usporiadanie

Vyhľadávanie produktov je implementované v `ProductController` v metóde `index` a je zobrazené v `products.blade.php`. V metóde `index` sa skontroluje či zákazník niečo zadal do vyhľadávania. Pokiaľ nič nezadal, tak sa produkty zobrazia bez vyhľadávania. Ak by zadal nejakú frázu, tak to vyhľadá produkty podľa názvu produktu. Produkty sú vyhľadávané podľa `ilike` funkcie a sú `case insensitive`. Napríklad, ak zákazník chce vyhľadať stolík a napíše `st`, tak mu nájde všetky produkty, ktoré majú v názve `st`.

Filtrovanie produktov je implementované v `ProductController` v metóde `index` a je zobrazené v `products.blade.php`. Filtrovať môže zákazník podľa kategórie, farby, materiálu, umiestnenia produktu (spálňa, kuchyňa...) alebo ceny, kde môže zadať cenový rozsah. V metóde `index` sa skontroluje, ktoré filtre sú zákazníkom zadané a následne sa zobrazia iba tie produkty, ktoré spĺňajú tieto filtre.

Zoradenie produktov je implementované v `ProductController` v metóde `index` a je zobrazené v `products.blade.php`. Zákazník môže produkty zoradiť podľa ceny (zostupne alebo vzostupne) alebo abecedne. V metóde `index` sa skontroluje, ktoré zoradenie je aplikované. Pokiaľ nie je žiadne, tak je považované za základné zoradenie a produkty sa zobrazia bez nejakého zoradenia. Ak zákazník zadal zoradenie, tak sa skontroluje či sa jedna o cenové zostupné, cenové zostupné alebo abecedné zoradenie. Na základne zadaného zoradenie sa zoradia príslušné produkty a zobrazia sa v správnom poradí.

Tieto procesy sú zákazníkovi umožnené pomocou `products.blade.php`. Pre vyhľadávanie produktov je poskytnuté vyhľadávacie pole, pre filtre je poskytnutý postranný panel, ktorý obsahuje možné filtre a pre zoradenie je poskytnutý dropdown s možnými možnosťami zoradenia. Na základe vybratej akcie sa nasledovne odošle požiadavka GET s parametrami `search`, `category`, `color`, `sort` ... pre vyhľadávanie, zoradenie a všetky filtre. Dopyt, ktorý vracia produkty sa na základe týchto parametrov upraví a zobrazí len požadované produkty.

Stránkovanie

Stránkovanie je implementované v ProductController v metóde index a je zobrazené v products.blade.php s použitím šablóny Bootstrap 5 paginácie, ktorý je zadaný v súbore bootstrap-5.blade.php. Pri zobrazovaní požadovaných produktov sa vykoná dopyt, ktorý je modifikovaný *paginate(10)*, ktorý definuje koľko produktov sa má zobraziť na stránku. V našej aplikácii je to 10 produktov na stránku. Následne je k tomuto dopytu pridaná požiadavka aj s filtrami, zoradením a vyhľadávaním pomocou *append* čo zaručí, že sa filtre, zoradenie a vyhľadávanie nestratia a aplikujú sa. Stránkovanie je zobrazené na spodu stránky, pričom je poskytnutých len toľko stránok koľko pre produkty treba. Dá sa medzi nimi navigovať pomocou preklikávania na jednotlivé čísla alebo šípky na bokoch.

Pridanie produktu do košíka

Pridanie produktu do košíka je implementované v CartController pomocou metódy add a je zobrazené v product-detail.blade.php. Najprv sa overí či id produktu existuje v databáze a či je quantity celé kladné číslo (minimálne 1). Následne sa produkt získa pomocou jeho id. Následne, ak je zákazník prihlásený tak:

- Sa načíta košík pre konkrétneho zákazníka
- Skontroluje sa či už neobsahuje produkt v košíku, ak už obsahuje, tak sa mu len zvýši quantity, ak nie tak sa tam pridá produkt na novo
- Následne aktualizuje košík tabuľku v databáze

Ak zákazník nie je prihlásený tak:

- Jeho košík sa nachádza v session, takže sa získa z tadiaľ
- Skontroluje sa či už neobsahuje produkt v košíku, ak už obsahuje, tak sa mu len zvýši quantity, ak nie tak sa tam pridá produkt na novo
- V tomto prípade sa nezmení nič v databáze, lebo sa s ňou nepracuje

Po úspešnom pridaní produktu do košíka je zákazník presmerovaný na stránku košíka s úspešnou správou, ktorá mu oznamuje, že produkt bol úspešne pridaný do košíka.

Tento proces je umožnený zákazníkovi pomocou product-detail.blade.php, kde si môže vybrať quantity daného produktu. Následne klikne na tlačidlo "Pridať do košíka" a odošle POST požiadavku na CartController pre metódu add. Táto požiadavka už je neskôr spracovaná pomocou CartController.

Snímky obrazovky

Domovská stránka

Hľadať

Hľadať

Domov

Produkty

Prihlásiť sa

Košík

Spálňa




Kuchyňa

Kúpeľňa

Obývačka

Kancelária

Exteriér



Najpredávanejšie

Detail produktu

Hľadať

Hľadať

Domov

Produkty

Prihlásiť sa

Košík

Spálňa





Kuchyňa

Kúpeľňa

Obývačka

Kancelária

Exteriér



Moderná posteľ

649.99€

Typ produktu

Farba

Materiál

Umiestnenie

Rozmery Š-D-H (cm)

Postele

Sivá

Textília

Spálňa

180-210-50

1


Pridať do košíka

Popis

Tento kus nábytku prináša dokonalú kombináciu štýlu, komfortu a funkčnosti. Navrhnutý s dôrazom na kvalitu a trvanlivosť, je ideálnou voľbou pre každý interier. Jeho nadčasový dizajn a kvalitné materiály zaručujú, že sa hodi do rôznych miestností a štýlov zariadenia. Či už v obývacej izbe, spálni, kancelárii alebo jedálni, tento nábytok prinesie do vášho domova nielen pohodlie, ale aj estetický dojem. Jeho praktické využitie a moderný vzhľad ho robia ideálnym doplnkom každého priestoru.

Súvisiace produkty

Košík s produktami



Hľadať

Hľadať

Domov

Produkty

Prihlásiť sa

Košík

Spálňa

Kuchyňa

Kúpeľňa

Obyvacia

Kancelária


Exteriér

1 Košík


2 Doprava a platba

3 Dodacie údaje

Produkt pridaný do košíka!



Moderná posteľ


Skladom 

-


1

+

649.99€



Klasická stolička

Skladom 

-

2

+

59.98€

Zlavový kód


Vložiť

← Späť k nákupu


Celkom: 709.97€

Pokračovať →

Domov Podmienky Produkty Kontakt



Prihlásenie

 Nestora

← Spät

Prihlásenie

Email

Heslo

☐ Zapamätat si ma

Prihlásiť sa

Nemáte ešte účet? [Zaregistruvať sa](#)

Návod na spustenie

1. Inštalácia PHP dependencies

```
composer install
```

2. Skopírovanie .env súboru

```
copy .env.example .env
```

3. Konfigurácia .env súboru

DB_CONNECTION=pgsql

DB_HOST=127.0.0.1

DB_PORT=5432

DB_DATABASE=_____

DB_USERNAME=_____

DB_PASSWORD=_____

SESSION_DRIVER=database

SESSION_LIFETIME=120

4. Migrácia databázy (vytvorenie tabuľiek)

```
php artisan migrate:fresh
```

5. Seed databázy (načítanie hodnôt)

```
php artisan db:seed
```

6. Vytvorenie symbolickej linky pre úložisko

```
php artisan storage:link
```

7. Inštalácia NPM packages

```
npm install
```

8. Spustenie aplikácie

```
php artisan serve
```