Fakulta informatiky a informačných technológií STU v Bratislave

**Semestrálny projekt - elektronický obchod**

**Adam Pašteka, René Revay**

2025

**Obsah**

[1. Zadanie 4](#_Toc196905486)

[2. Diagram fyzického dátového modelu 5](#_Toc196905487)

[2.1. Starý 5](#_Toc196905488)

[2.2. Nový 6](#_Toc196905489)

[2.3. Odovodnenie a zmeny 6](#_Toc196905490)

[2.3.1. Tabuľka user\_items 6](#_Toc196905491)

[3. Návrhové rozhodnutia 7](#_Toc196905492)

[3.1. Riešenie rolí používateľov 7](#_Toc196905493)

[3.1.1. Guest 7](#_Toc196905494)

[3.1.2. User 7](#_Toc196905495)

[3.1.3. Admin 7](#_Toc196905496)

[3.2. .... 7](#_Toc196905497)

[4. Opis implementácie 8](#_Toc196905498)

[4.1. Prihlásenie user a admin 8](#_Toc196905499)

[4.2. Vyhľadávanie 8](#_Toc196905500)

[4.3. Pridanie produktu do košíka 8](#_Toc196905501)

[4.4. Stránkovanie (pagination) 8](#_Toc196905502)

[4.5. Základné filtrovanie 8](#_Toc196905503)

[4.6. Zmena množstva produktu 8](#_Toc196905504)

[4.7. Pridávanie produktov do košíka pre prihláseného používateľa a hosťa 8](#_Toc196905505)

[4.8. Pridávanie nových produktov cez AdminPage 9](#_Toc196905506)

[5. Screenshots 10](#_Toc196905507)

[5.1. HomePage 10](#_Toc196905508)

[5.2. Search-filter 10](#_Toc196905509)

[5.3. Products 11](#_Toc196905510)

[5.4. Login 11](#_Toc196905511)

[5.5. Register 12](#_Toc196905512)

[5.6. Shopping-cart 12](#_Toc196905513)

[5.7. Order 13](#_Toc196905514)

[5.8. Admin-page 13](#_Toc196905515)

[5.8.1. Add new product 14](#_Toc196905516)

[5.8.2. Delete product 14](#_Toc196905517)

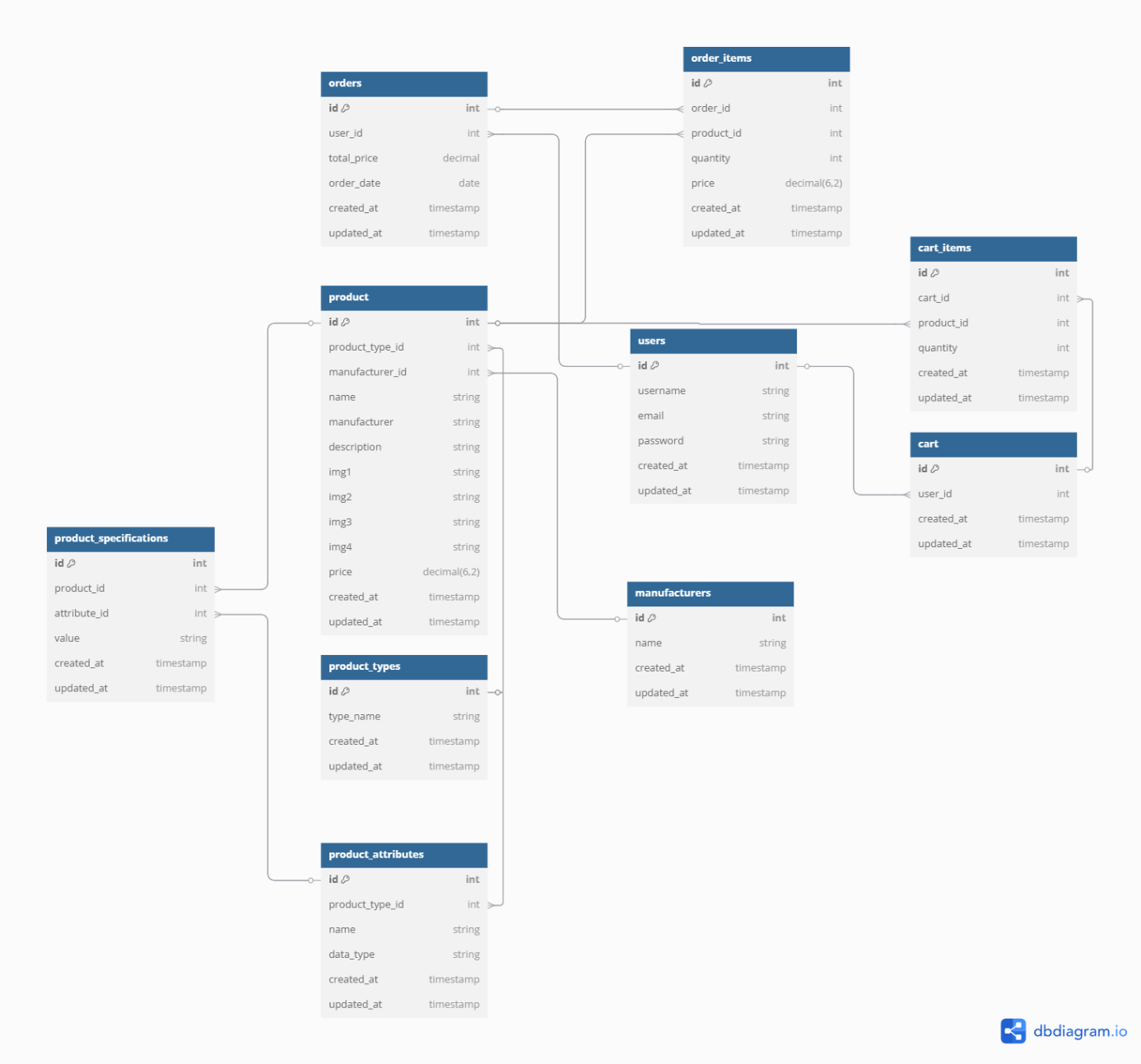
[5.8.3. Add product 15](#_Toc196905518)

[5.8.4. Edit product 16](#_Toc196905519)

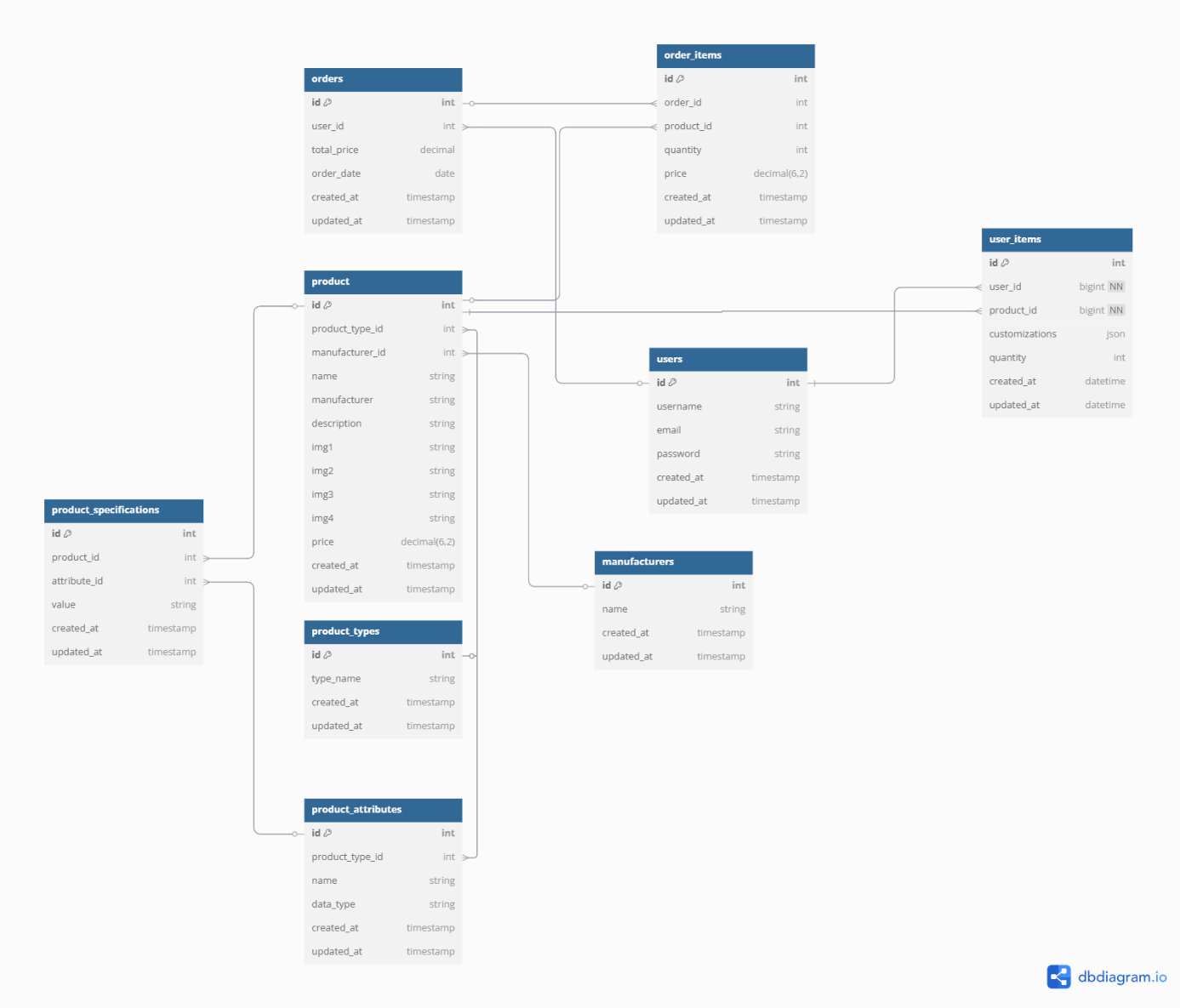
1. Zadanie

Vytvorte webovú aplikáciu - eshop, ktorá komplexne rieši nižšie definované prípady použitia vo vami zvolenej doméne (napr. elektro, oblečenie, obuv, nábytok). Presný rozsah a konkretizáciu prípadov použitia si dohodnete s vašim vyučujúcim.

1. Diagram fyzického dátového modelu
   1. Starý



* 1. Nový



* 1. Odovodnenie a zmeny

V novej verzii databázy boli tabuľky cart a cart\_items odstránené a nahradené jednou novou tabuľkou user\_items. Táto zmena prináša jednoduchší a flexibilnejší spôsob ukladania produktov priradených používateľom, napríklad pri ukladaní nakonfigurovaných produktov.

* + 1. Tabuľka user\_items
* priamo prepája používateľa s produktom (user\_id, product\_id),
* umožňuje ukladať rôzne prispôsobenia cez JSON stĺpec customizations,
* sleduje počet kusov (quantity),
* nahrádza potrebu zvlášť držať cart\_id.

1. Návrhové rozhodnutia
   1. Riešenie rolí používateľov

Rozlíšenie administrátora od bežného používateľa a hosta (guest) sa rieši kombináciou **pri prihlásení** a **na úrovni routes cez middleware**.

**Pri prihlásení** (LoginController):

* Overíme prihlasovacie údaje (email alebo username + password).
* Ak is\_admin = true, presmerujeme na admin panel (adminPage), inak na index.

**Middleware** (Kernel.php + IsAdmin):

* auth zabezpečí, že prístup k chráneným routám majú len prihlásení.
* is.admin (v app/Http/Middleware/IsAdmin.php) kontroluje, či má používateľ flag is\_admin == true.
  + 1. Guest
* Nepřihlásený používateľ vidí všetky verejné stránky.
* Položky pridané do košíka sa ukladajú do session('cart') na serverovej session.
* Pokus o prístup k routám, ktoré majú auth middleware (napr. objednávka, admin), vedie:
  + **pri objednávke** redirect na login,
  + **pri admin routách** redirect na login
    1. User
* Po prihlásení (auth middleware) má prístup k správe košíka a objednávke.
* Položky v košíku sa ukladá do tabuľky user\_items
  + 1. Admin
* Pri registrácii a seedovaní má is\_admin = true.
* Po prihlásení s is\_admin = true redirect na /admin.
* Plná CRUD funkcionalita pre produkty:
  + **GET** /admin/products (index),
  + **POST** /admin/products (store),
  + **PUT** /admin/products/{id} (update),
  + **DELETE** /admin/products/{id} (destroy + fyzické vymazanie obrázkov)

1. Prog. Prostredie
   1. Databáza

Použili sme databázový systém MySQL spravovaný cez phpMyAdmin a celý stack beží na XAMPP. Tento prístup nám umožnil minimalizovať konfiguráciu servera, využiť dobre známe rozhranie phpMyAdmin na správu databázy a okamžite pracovať s databázou bez potreby zložitého nastavovania prístupu či oprávnení pre PostgreSQL.

1. Opis implementácie
   1. Prihlásenie user a admin

Formulár POSTuje na routu /login, ktorú obsluhuje LoginController::login(). Tá overí prihlasovacie údaje cez Laravel Auth, pri úspechu presmeruje na predvolenú stránku a pri neúspechu vráti späť s chybovou flash správou.

* 1. Vyhľadávanie

GET na /search-filter/{type?} volá ProductController::searchFilter(). Získa voliteľný type param (ID kategórie) a query parameter name, zostaví Eloquent dotaz s ->where('type\_id', $type) a/nebo ->where('name', 'like', "%{$name}%"), výsledok stránkuje (napr. ->paginate(10)) a posiela do Blade view.

* 1. Pridanie produktu do košíka

V detaili produktu je tlačidlo „Pridať do košíka“, ktoré odosiela POST na /shopping-cart/add (môže byť aj inline JS). V CartController::add() sa položka vloží alebo navýši v session-ovom „cart“ poli a používateľ dostane flash správu „Produkt pridaný do košíka“.

* 1. Stránkovanie (pagination)

V oboch prípadoch (zoznam produktov aj výsledky vyhľadávania) sa používa Laravel paginator. V controllere sa volá ->paginate(n), v Blade šablóne sa potom vykreslia odkazy metodou {{ $products->links() }}.

* 1. Základné filtrovanie

Vedľa vyhľadávacieho poľa sú tlačidlá pre jednotlivé typy produktov (napr. Luky, Kuše, Praky…), ktoré odkazujú na /search-filter/{type}. Pri kliknutí sa do controllera pošle type a výsledky sa obmedzia pomocou ->where('type\_id', $type).

* 1. Zmena množstva produktu

Na stránke košíka sa pri zmene hodnoty <input type="number" name="amounts[...]"> spustí JS funkcia updateTotal(), ktorá aktualizuje cenu, a následne sendQuantityUpdate(productId, quantity) posiela AJAX POST na /shopping-cart/update-quantity. V CartController::updateQuantity() sa upraví položka v session-ovom „cart“ poli a vráti sa JSON s úspechom.

* 1. Pridávanie produktov do košíka pre prihláseného používateľa a hosťa

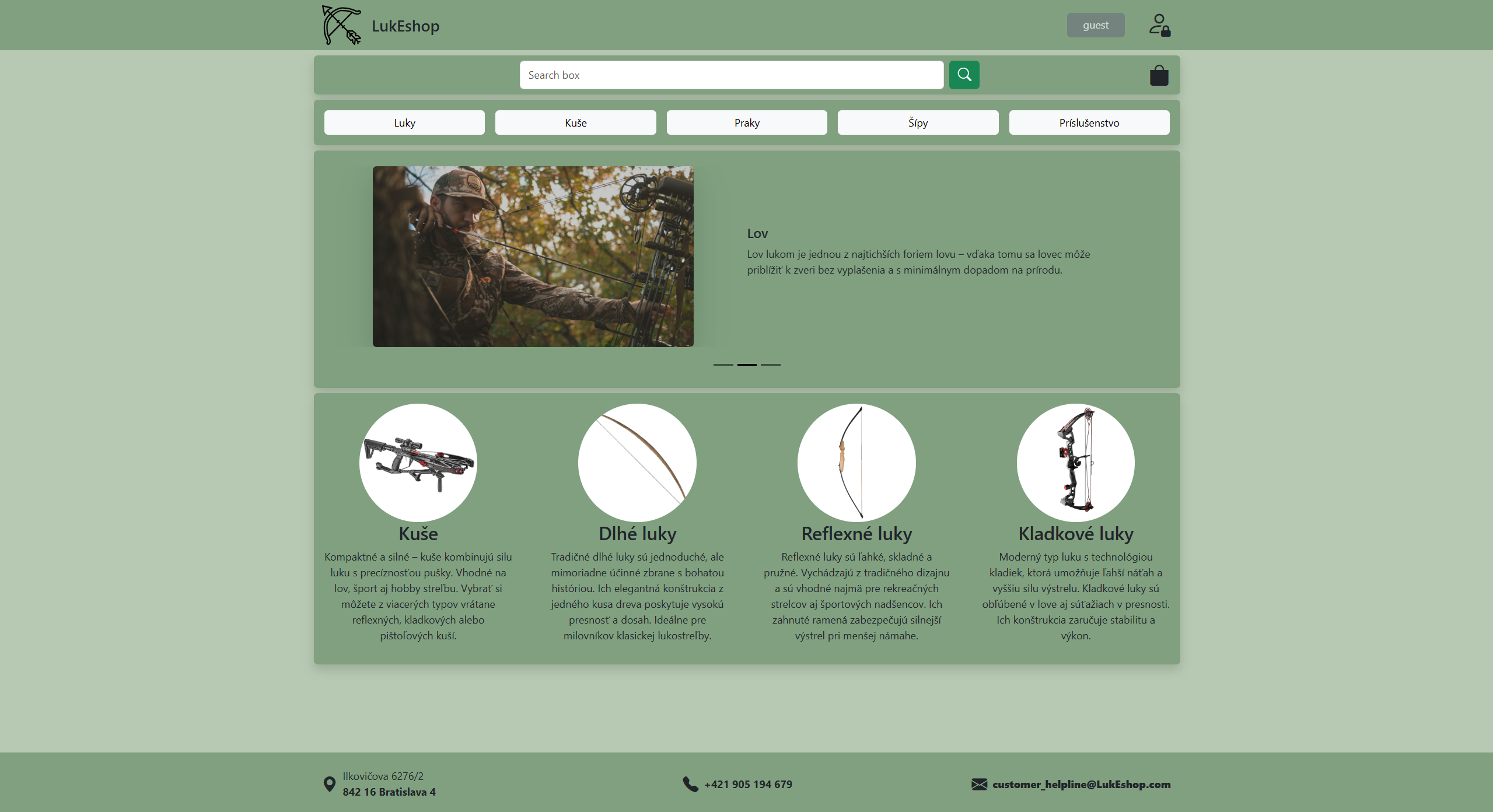
Keď používateľ, ktorý nie je prihlásený (guest), klikne na tlačidlo „Pridať do košíka“, aplikácia odošle POST požiadavku na /shopping-cart/add. Backend v CartController::add() rozpozná, že používateľ nie je autentifikovaný, a pridanú položku (ID produktu a jeho množstvo) uloží do session, konkrétne do poľa session('cart'), kde sa každé ID produktu mapuje na aktuálne množstvo. Po úspešnom pridaní sa session košík aktualizuje a používateľovi sa zobrazí flash správa „Produkt pridaný do košíka“. Tento hosťovský košík existuje iba počas trvania session prehliadača; jej vyprázdnením alebo zatvorením sa košík odstráni.

V prípade prihláseného používateľa prebieha pridanie veľmi podobne – klik na „Pridať do košíka“ spustí rovnakú POST požiadavku na /shopping-cart/add. V CartController::add() sa najskôr overí stav autentifikácie pomocou Auth::check(). Ak je používateľ prihlásený, položka sa namiesto session uloží do databázovej tabuľky cart\_items volaním metódy Cart::updateOrCreate(), ktorá buď vytvorí nový záznam pre daného používateľa a produkt, alebo zvýši množstvo existujúcej položky. Navyše, vždy keď sa používateľ prihlási a v session je ešte hosťovský košík, aplikácia v rámci CartService::mergeGuestCart() iteruje cez všetky položky zo session a presunie ich do databázy pre tohto používateľa, pričom množstvá sa pri zhodných ID produktov spočítajú. Po úspešnom zlúčení sa hosťovský košík v session vyprázdni. Vďaka tomu má prihlásený používateľ k dispozícii svoj košík uložený trvalo v databáze, ktorý sa zároveň plynulo doplní o produkty, ktoré si pred prihlásením vybral ako hosť.

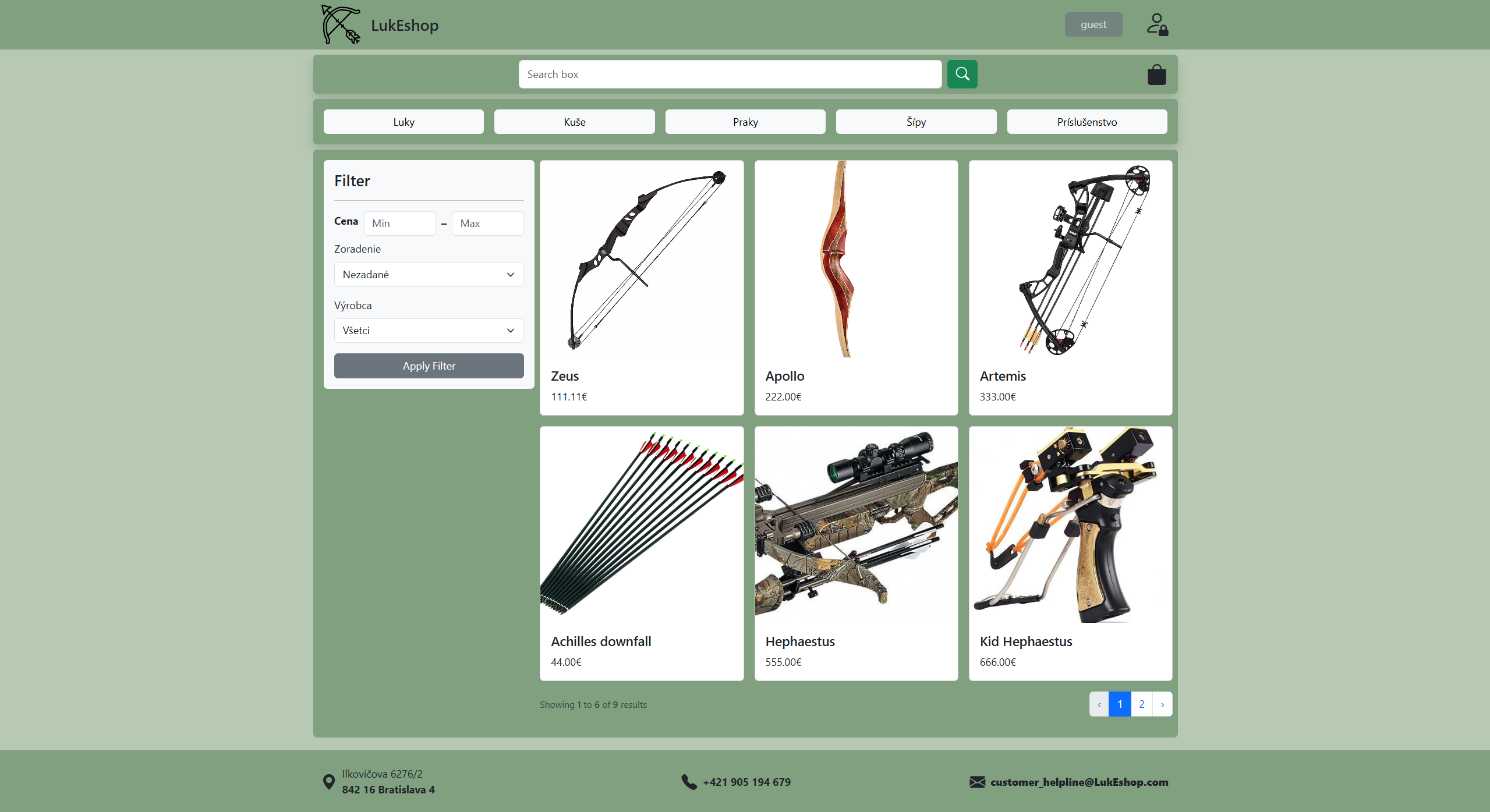
* 1. Pridávanie nových produktov cez AdminPage

Keď je administrátor prihlásený, zobrazí sa mu špeciálne administrátorské rozhranie, v ktorom vidí už existujúce produkty a môže zvoliť možnosť „Add Item“. Po kliknutí na „Add Item“ sa najskôr zobrazí výber preddefinovaných typov – luk, kuša, prak a pod. – kde administrátor vyberie kategóriu nového tovaru, a až potom klikne na „Add Product“. Aplikácia mu následne zobrazí formulár, v ktorom vyplní názov, popis, cenu,  dostupné parametre pre daný produkt a obrázky pričom minimum sú 2 obrázky pre daný produkt.

1. Screenshots
   1. HomePage



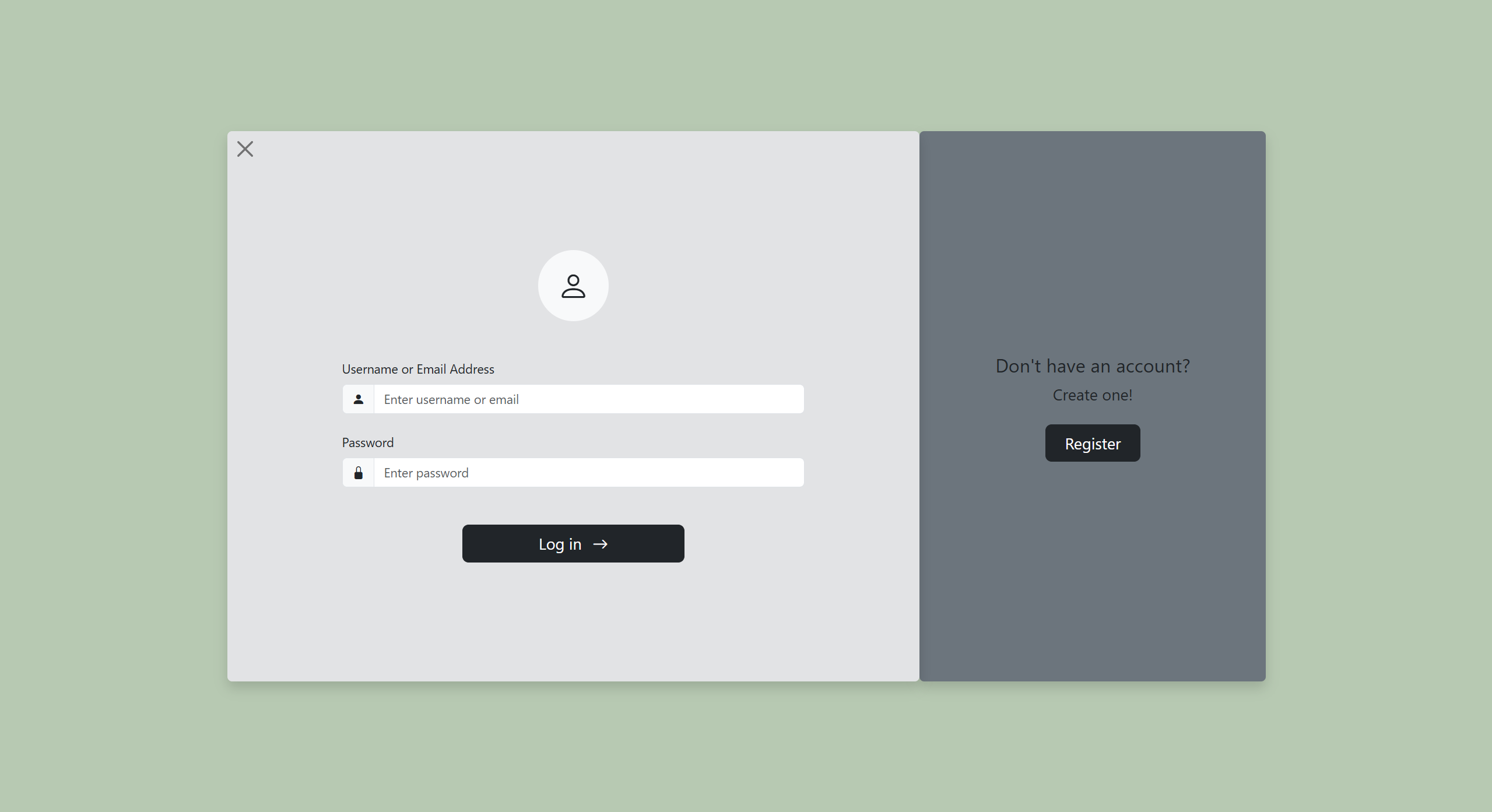
* 1. Search-filter



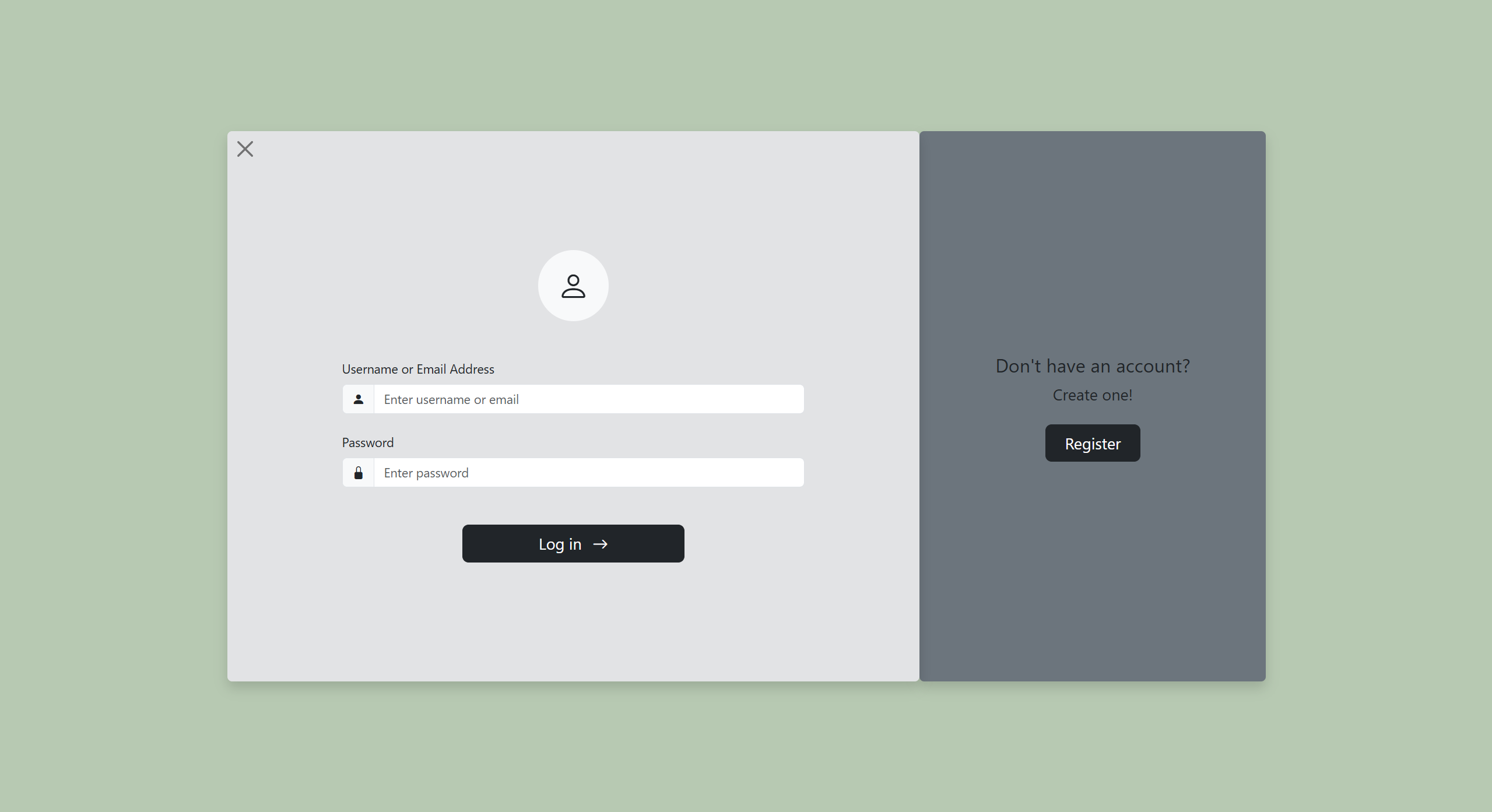
* 1. Products



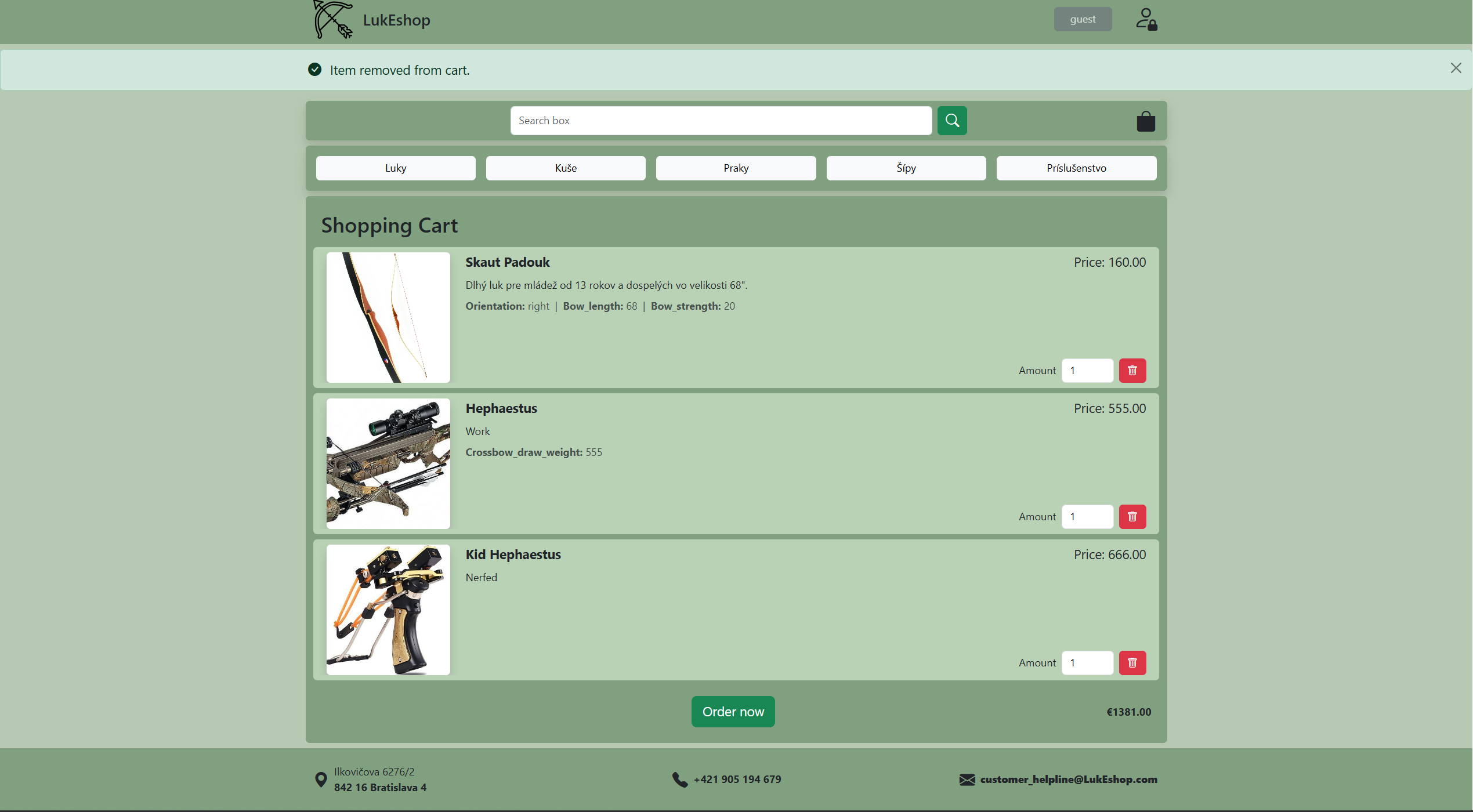
* 1. Login



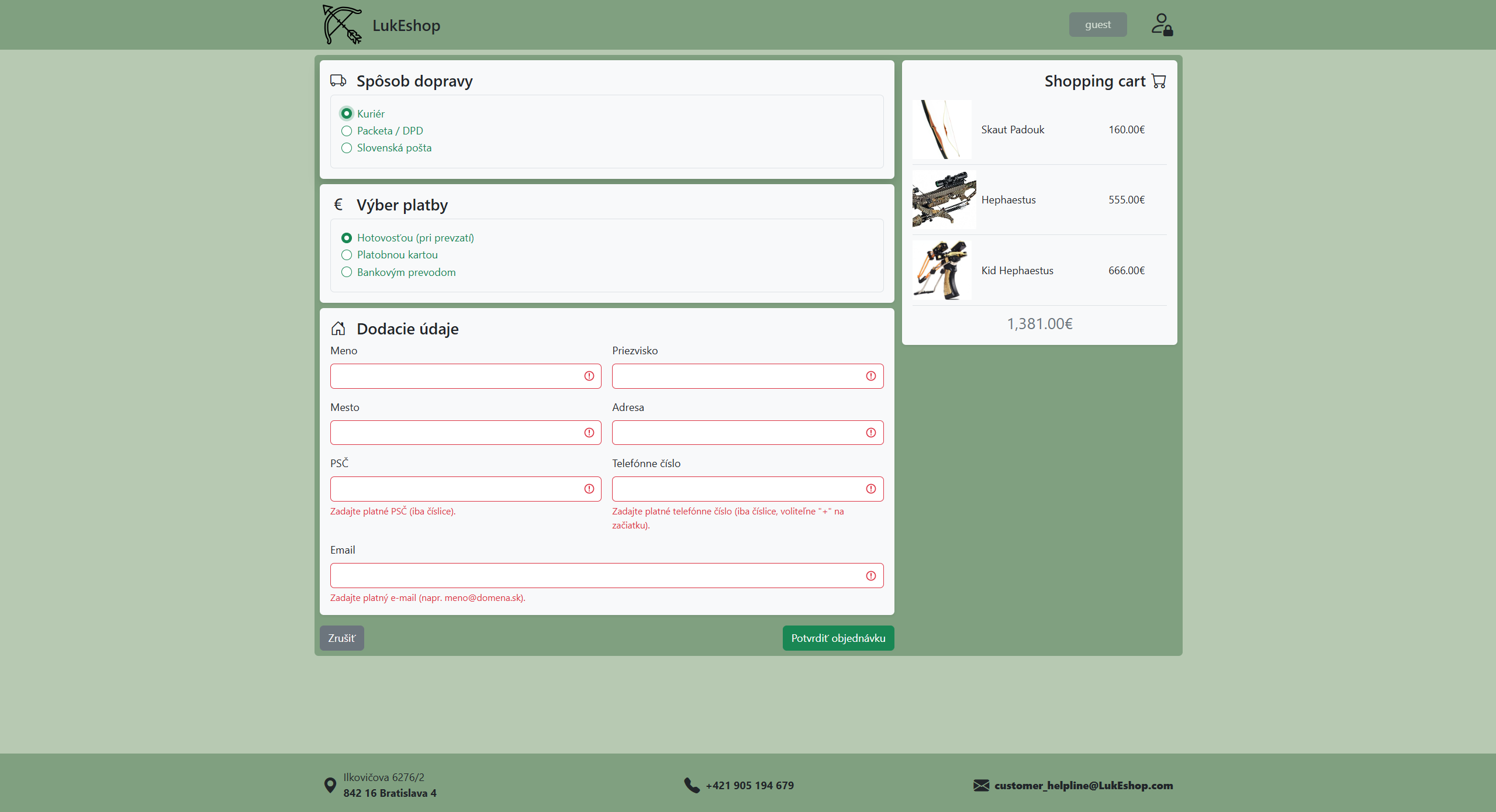
* 1. Register



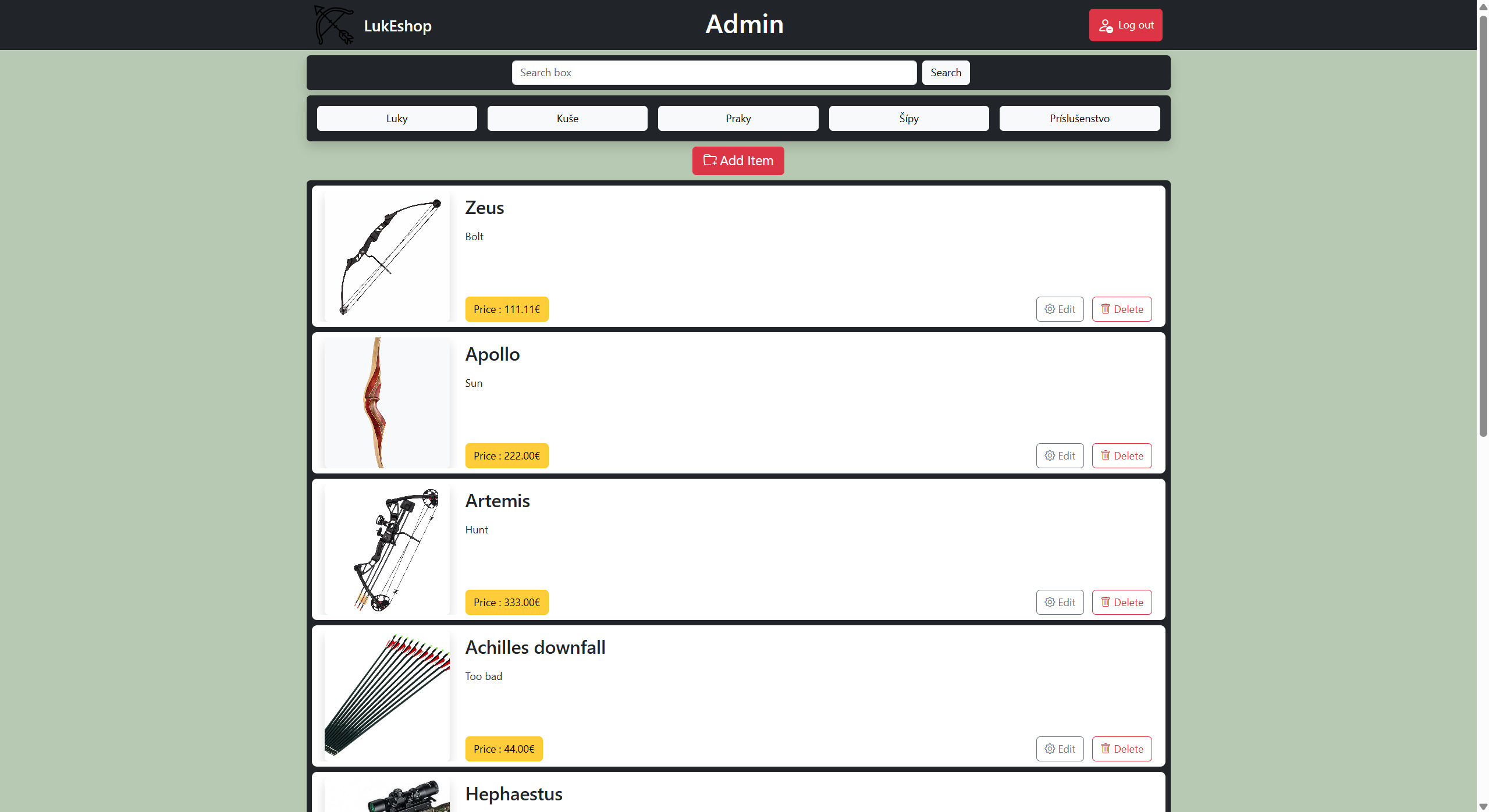
* 1. Shopping-cart



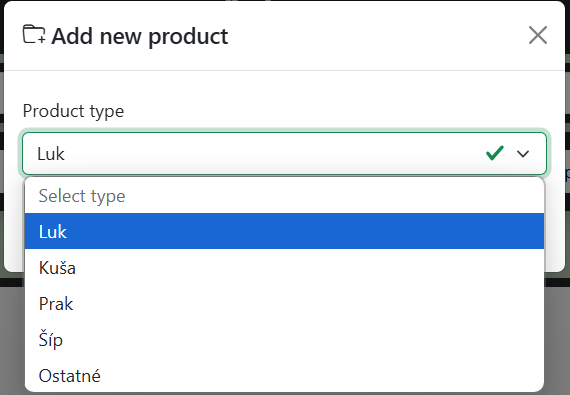
* 1. Order



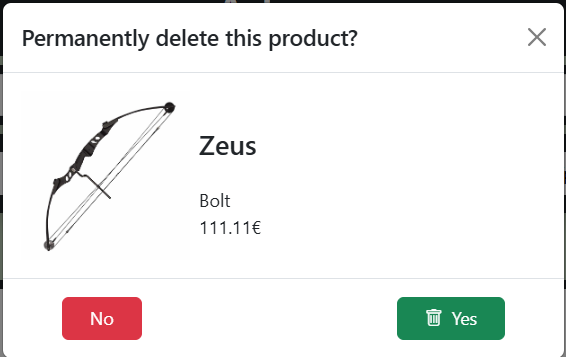
* 1. Admin-page



* + 1. Add new product

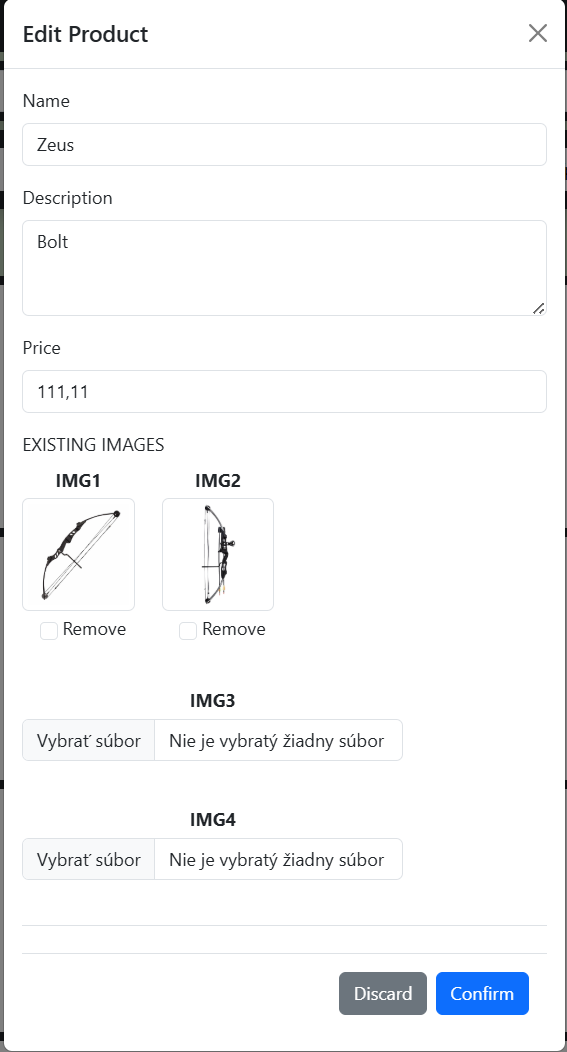


* + 1. Delete product



* + 1. Add productObrázok, na ktorom je text, snímka obrazovky, číslo, softvér

       Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.
    2. Edit product



1. Používateľská príručka(win11)
   1. Inštalácia Xampp
2. <https://www.apachefriends.org/download.html>
3. stiahnite verziu kde je php verzia 8.2 + napr.



1. v dowload files alebo v download files browsera otvorte inštalačný balík
2. Inštalácia Xampp

Obrázok, na ktorom je text, snímka obrazovky, softvér, počítačová ikona

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Obrázok, na ktorom je text, snímka obrazovky, softvér, počítačová ikona

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Obrázok, na ktorom je text, snímka obrazovky, softvér, počítačová ikona

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Obrázok, na ktorom je text, snímka obrazovky, softvér, displej

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Obrázok, na ktorom je text, snímka obrazovky, softvér, počítačová ikona

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Obrázok, na ktorom je text, snímka obrazovky, softvér, operačný systém

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

1. Zapnite moduly Apache a MySQL

Obrázok, na ktorom je text, snímka obrazovky, softvér, počítačová ikona

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

1. Overte či sa viete dostať na <http://localhost/phpmyadmin>

Obrázok, na ktorom je text, snímka obrazovky, softvér, webová stránka

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

1. Overte verziu php cez cmd príkazom php -v

Obrázok, na ktorom je text, snímka obrazovky, písmo

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

**Je potrebné mať verziu minimálne 8.2**

* 1. Vytvorenie Priečinku pre projekt a Composer
     + 1. Na ploche alebo niekde kde chcete vytvorte priečinok na projekt
       2. Do path poľa zadajte path cmd

Obrázok, na ktorom je snímka obrazovky, softvér, text, multimediálny softvér

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

* + - 1. Naklonujte git repozitár do tohto priečinku použite príkaz

git clone <https://github.com/Tencobijemuchy/Hubacka2022.git>

Obrázok, na ktorom je text, snímka obrazovky, písmo, čierny

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

* + - 1. Do priečinka kde sa nachádza súbor **artisan** nainštalujte Composer použite príkazy

cd Hubacka2022

cd projekt

Obrázok, na ktorom je text, písmo, snímka obrazovky, čierny

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Nainštalujte Composer príkazom

composer install

počkáme kým sa dokončí inštalácia

* 1. Databáza a app key
     1. Vygenerujte kľúč pre laravel app použite príkaz

php artisan key:generate

Obrázok, na ktorom je text, snímka obrazovky, písmo

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

* + 1. Vytvorte databázu použitím príkazu

php artisan migrate

Na otázku či chcete vytvoriť databázu napíšte **yes**

Obrázok, na ktorom je text, snímka obrazovky, softvér, písmo

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

* + 1. Naplňte databázu pomocou seederov použite príkaz

php artisan db:seed

Obrázok, na ktorom je text, snímka obrazovky, písmo, softvér

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

* + 1. Overte existenciu databázy v <http://localhost/phpmyadmin>

Obrázok, na ktorom je text, snímka obrazovky, softvér, počítačová ikona

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Obrázok, na ktorom je text, snímka obrazovky, písmo, číslo

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

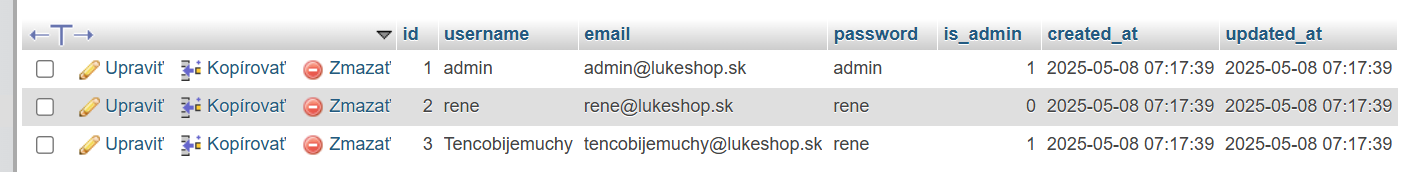
Dvojklikom otvorte databázu lukeshop

* + 1. Overte jednotlivé tabuľky či sú naplnené

Obrázok, na ktorom je text, snímka obrazovky, písmo, číslo

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Users



Product types

Obrázok, na ktorom je text, písmo, snímka obrazovky, číslo

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

**Mali by byť naplnené tabuľky users, product\_types,product\_specifications,product\_atributes, products, migrations, manufacturers**

* 1. Zapnite projekt
     1. Zapnite projekt použitím príkazu

php artisan serve

Obrázok, na ktorom je text, snímka obrazovky, písmo

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

* + 1. Otvorte Link <http://127.0.0.1:8000/>

Obrázok, na ktorom je text, snímka obrazovky, softvér, webová stránka

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.