#### Министерство науки и высшего образования Российской Федераци­­и

#### Федеральное государственное бюджетное образовательное учреждение науки и высшего образования

#### «Владимирский государственный университет

#### имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №5

по дисциплине

"Основы разработки веб приложений"

Тема: "Знакомство с фреймворком Laravel"

Выполнил:

ст. гр. ПРИ-117

В.Г. Хлызова

Принял:

Козаев А.Т.

Владимир, 2020 г.

## ЦЕЛЬ РАБОТЫ

Познакомиться со структурой приложений Laravel и разработать приложение “Корзина покупателя”.

ХОД РАБОТЫ

Подключаем БД к проекту, для этого редактируем переменные окружения в файле **.env** в корне проекта.

DB\_CONNECTION=mysql

DB\_HOST=127.0.0.1

DB\_PORT=3306

DB\_DATABASE=lab\_laravel

DB\_USERNAME=root

DB\_PASSWORD=

Генерируем модель Product и добавляем поля. Выполняем миграцию.

public function up()

{

Schema::create('products', function (Blueprint $table) {

$table->id();

$table->string("name", 255)->nullable();

$table->text("description")->nullable();

$table->string("photo", 255)->nullable();

$table->decimal("price", 6, 2);

$table->timestamps();

});

}

Заполняем БД начальными значениями.

<?php

use Illuminate\Database\Seeder;

class ProductsSeeder extends Seeder

{

/\*\*

\* Run the database seeds.

\*

\* @return void

\*/

public function run()

{

DB::table('products')->insert([

'name' => 'LG',

'description' => 'Protection Corning Gorilla Glass 4. MISC Colors Space Black, Luxe White, Modern Beige, Ocean Blue, Opal Blue.',

'photo' => 'https://i.ebayimg.com/00/s/NjQxWDQyNA==/z/VDoAAOSwgk1XF2oo/$\_35.JPG?set\_id=89040003C1',

'price' => 39999.01

]);

DB::table('products')->insert([

'name' => 'Samsung Galaxy S8',

'description' => 'Диагональ (дюйм): 5.8

Разрешение (пикс): 2960x1440

Встроенная память (Гб): 64

Фотокамера (Мп): 12

Процессор: Samsung Exynos 8895 Octa',

'photo' => 'https://cdn.svyaznoy.ru/upload/iblock/d1c/4165313\_10.jpg/resize/483x483/hq/',

'price' => 29999.71

]);

DB::table('products')->insert([

'name' => 'Honor 20 Pro',

'description' => 'Диагональ (дюйм): 6.26

Разрешение (пикс): 2340x1080

Встроенная память (Гб): 256

Фотокамера (Мп): 48 + 16 + 8 + 2 (четыре основные камеры)

Оптический зум: x3',

'photo' => 'https://cdn.svyaznoy.ru/upload/iblock/870ad14aa9255041eac3a9e7e7a271b0/1.jpg/resize/483x483/hq/',

'price' => 39999.91

]);

DB::table('products')->insert([

'name' => 'Apple iPhone 11',

'description' => 'Диагональ (дюйм): 6.1

Разрешение (пикс): 1792x828

Встроенная память (Гб): 256

Фотокамера (Мп): 12 + 12 (двойная)

Оптический зум: x2',

'photo' => 'https://cdn.svyaznoy.ru/upload/iblock/d47/iphone\_11\_g\_2.jpg/resize/483x483/hq/',

'price' => 99679.91

]);

DB::table('products')->insert([

'name' => 'Xiaomi Redmi',

'description' => 'Диагональ (дюйм): 6.22

Разрешение (пикс): 1520x720

Встроенная память (Гб): 32

Фотокамера (Мп): 12 + 2 (двойная)

Процессор: Qualcomm Snapdragon 439',

'photo' => 'https://cdn.svyaznoy.ru/upload/iblock/06b/86eee5cfc356ed4ab3462fdc841d7f98.jpg/resize/483x483/hq/',

'price' => 29888.88

]);

DB::table('products')->insert([

'name' => 'Nokia',

'description' => 'Диагональ (дюйм): 5.71

Разрешение (пикс): 1520x720

Встроенная память (Гб): 16

Фотокамера (Мп): 13

Процессор: MediaTek MT6761 Helio A22',

'photo' => 'https://cdn.svyaznoy.ru/upload/iblock/13e/1559822818\_nokia\_2\_2-spec-sheet-black.jpg/resize/483x483/hq/',

'price' => 99999.99

]);

}

}

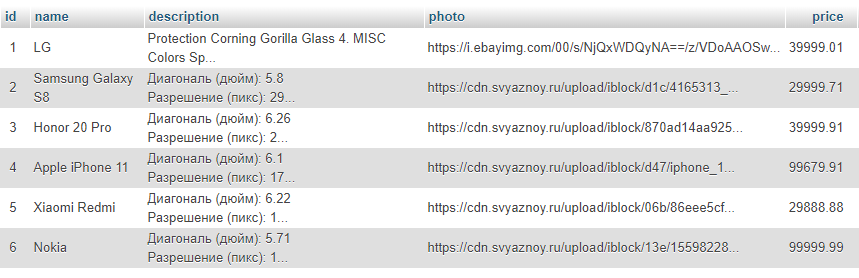


Рисунок 1. Заполнение БД записями.

Создаем контроллер для вывода, добавления и удаления всех продуктов и продуктов из корзины покупателя.

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Product;

use Session;

use App\Cart;

class ProductsController extends Controller

{

public function index()

{

$products = Product::all();

return view('products', compact('products'));

}

public function add\_products(Request $req)

{

$product = new Product();

$product->name = $req->input('name');

$product->description = $req->input('description');

$product->photo = $req->input('photo');

$product->price = (float)$req->input('price');

$product->save();

return redirect()->intended('/');

}

public function delete($id)

{

Product::find($id)->delete();

return redirect()->intended('/');

}

public function addToCart($id, Request $request)

{

$product = Product::find($id);

$oldCart = Session::has('cart') ? Session::get('cart') : null;

$cart = new Cart($oldCart);

$cart->add($product);

$request->session()->put('cart', $cart);

$request->session()->save();

//dd($request->session()->get('cart'));

return redirect()->intended('/');

}

public function showShoppingCart(Request $request){

//dd($request->session()->get('cart'));

if (Session::has('cart')){

$cart = $request->session()->get('cart');

return view('shoppingCart', ['products'=> $cart->items, 'totalPrice' => $cart->totalPrice]);

}

}

public function deleteFromCart($id, Request $request){

$oldCart = Session::get('cart');

$cart = new Cart($oldCart);

$product = Product::find($id);

$cart->delete($product);

$request->session()->put('cart', $cart);

$request->session()->save();

return redirect()->intended('showShoppingCart');

}

}

Класс содержащий данные для корзины: товары, общее кол-во продуктов и сумму всего заказа.

<?php

namespace App;

class Cart

{

public $items;

public $totalPrice=0;

public $quantity=0;

public function \_\_construct($oldCart){

if ($oldCart){

$this->items = $oldCart->items;

$this->totalPrice = $oldCart->totalPrice;

$this->quantity = $oldCart->quantity;

}

}

public function add($item){

$i = ['quantity'=> 0, 'item'=>$item ];

if ($this->items){

if(array\_key\_exists($item->id, $this->items)){

$i=$this->items[$item->id];

}

}

$i['quantity']++;

$this->items[$item->id] = $i;

$this->totalPrice+=$item->price;

$this->quantity++;

}

public function delete($item){

$array = $this->items;

$quantity = $array[$item->id]['quantity']-1;

$array[$item->id]['quantity']=$quantity;

$this->totalPrice-=$item->price;

if($array[$item->id]['quantity']==0){

unset($array[$item->id]);

}

$this->items = $array;

$this->quantity--;

}

}

Содержимое файла routes/web.ph

<?php

use Illuminate\Support\Facades\Route;

Route::get('/', 'ProductsController@index');

Route::post('/add\_products', 'ProductsController@add\_products');

Route::get('/delete/{id}', 'ProductsController@delete');

Route::get('/add', function(){

return view('add\_products');

});

Route::get('add-to-cart/{id}', 'ProductsController@addToCart');

Route::get('/showShoppingCart', 'ProductsController@showShoppingCart');

Route::get('/deleteFromCart/{id}', 'ProductsController@deleteFromCart');

Страница с выводом всех продуктов.

@extends('layout')

@section('title', 'Products')

@section('content')

<p class="btn-holder">

<a href="{{ url('/add') }}" class="btn btn-warning btn-block text-center" role="button">Add new product</a>

<a href="{{ url('/showShoppingCart')}}" class="btn btn-warning btn-block text-center" role="button">Shopping сart

<span class="badge">{{Session::has('cart') ? Session::get('cart')->quantity : ''}}</span></a>

</p>

<div class="container products">

<div class="row">

@foreach($products as $product)

<div class="col-xs-18 col-sm-6 col-md-3">

<div class="thumbnail">

<img src="{{ $product->photo }}" width="200" height="300">

<div class="caption">

<h4>{{ $product->name }}</h4>

<p>{{ \Illuminate\Support\Str::limit($product['description'], 50) }}</p>

<p><strong>Price: </strong> {{ $product->price }}$</p>

<p class="btn-holder"><a href="{{ url('add-to-cart/'.$product->id) }}" class="btn btn-warning btn-block text-center" role="button">Add to cart</a>

<p class="btn-holder"><a href="{{ url('delete/'.$product->id) }}" class="btn btn-warning btn-block text-center" role="button">Delete</a></p>

</div>

</div>

</div>

@endforeach

</div>

</div>

@endsection

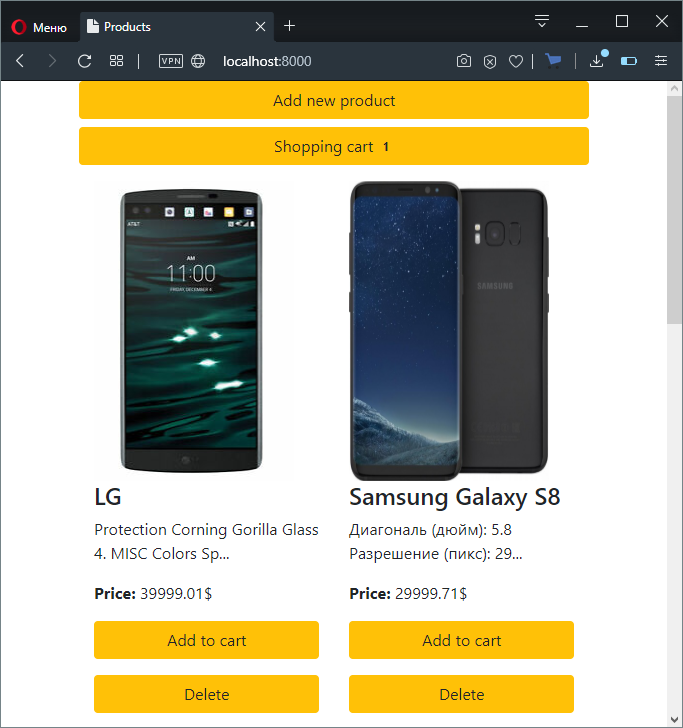


Рисунок 2. Страница с выводом всех продуктов.

Страница с добавлением нового продукта.

@extends('layout')

@section('title', 'addProducts')

@section('content')

<form action="/add\_products" method="post">

@csrf

<div>

<p><strong>Name:</strong>

<input name="name" type="text" placeholder="enter a name" id="name" class="form-control" required></p>

<p><strong>Description: </strong>

<textarea name="description" placeholder="enter a description" id="description" class="form-control" required></textarea></p>

<p><strong>Photo: </strong>

<input name="photo" type="text" placeholder="enter a photo url" id="photo" class="form-control" required></p>

<p><strong>Price: </strong>

<input name="price" type="text" placeholder="enter a price" id="price" class="form-control" required></p>

<button type="submit" class="btn btn-warning btn-block text-center">Add product</button>

</div>

</form>

@endsection

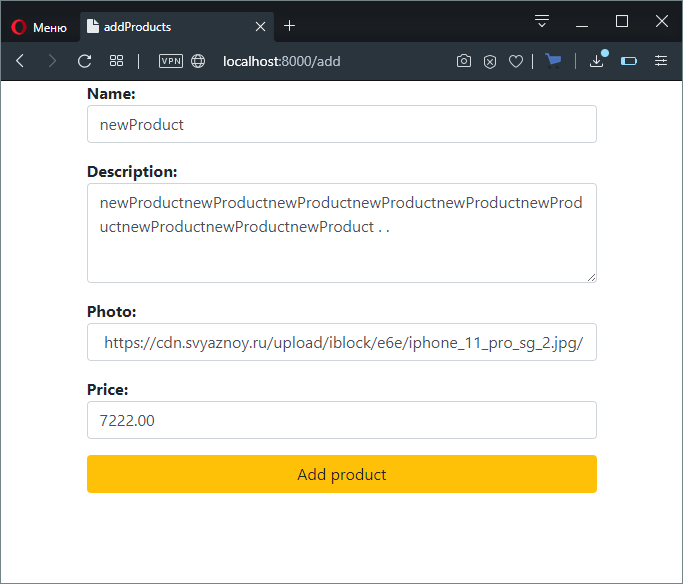


Рисунок 3. Страница добавления.

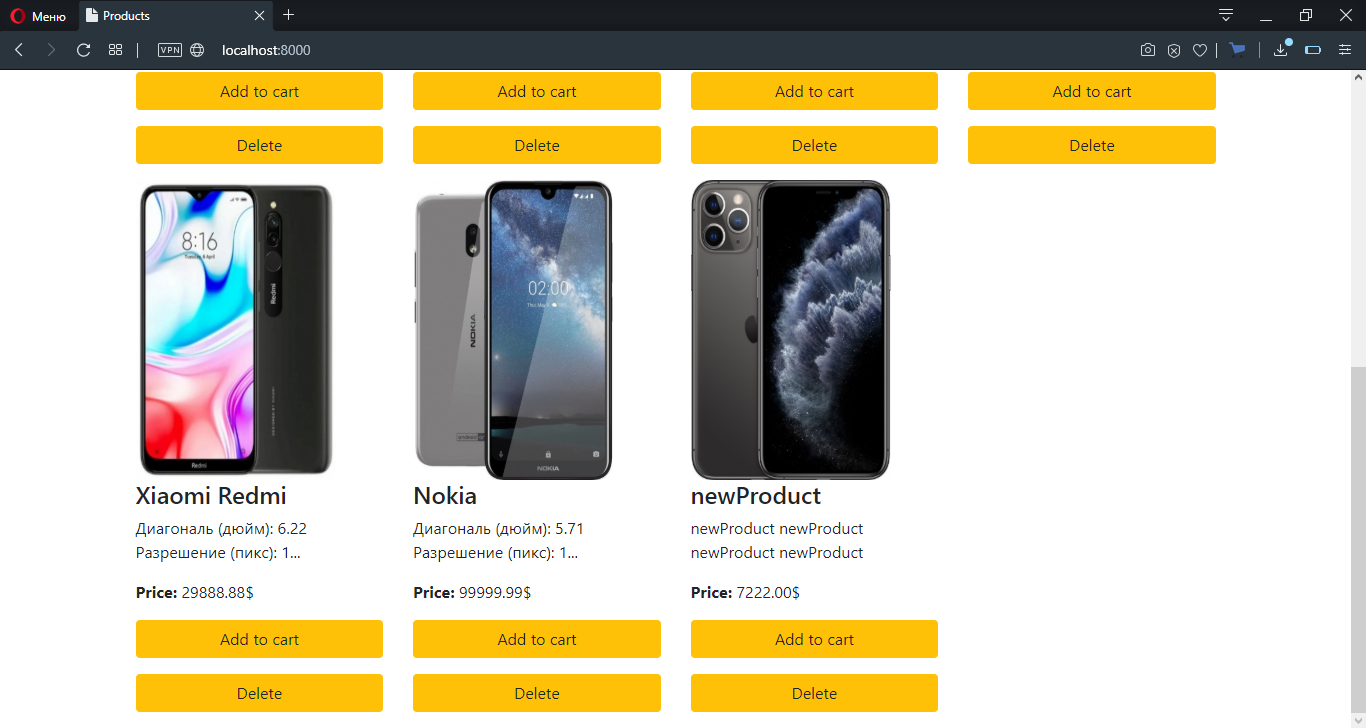


Рисунок 4. Новый продукт добавлен.

Корзина покупателя.

@extends('layout')

@section('title', 'Shopping cart')

@section('content')

<div class="container products">

<div class="colomn">

<div>

<h2>Shopping cart - {{ $totalPrice }}$</h2><br>

</div>

@foreach($products as $product)

<div>

<h4>{{ $product['item']['name'] }} - X{{$product['quantity']}} </h4>

<p>{{ \Illuminate\Support\Str::limit($product['item']['description'], 200) }}</p>

<p>Price: {{$product['item']['price']}}$</p>

<p class="btn-holder"><a href="{{ url('deleteFromCart/'.$product['item']['id'])}}" class="btn btn-warning btn-block text-center" role="button">Delete from cart</a></p>

</div>

@endforeach

</div>

</div>

@endsection

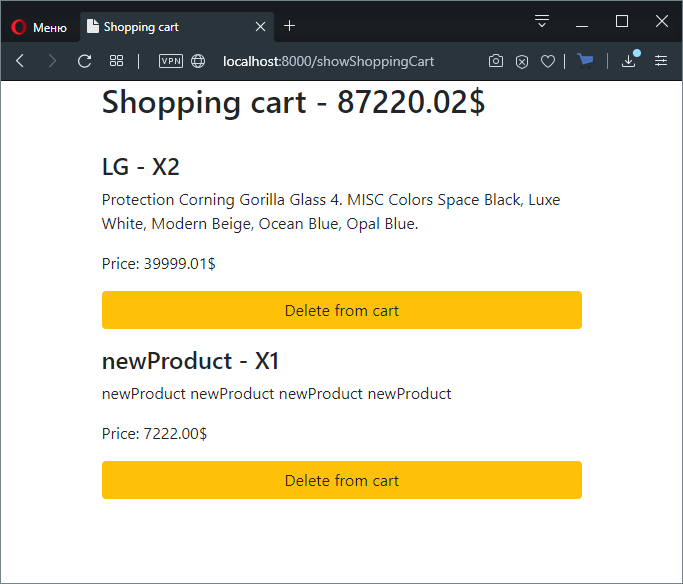


Рисунок 5. Корзина покупателя.

Ссылка на гит: <https://github.com/arranay/Basics-of-web-application-development>

ВЫВОД

В ходе лабораторной работы была изучена структура приложения Laravel и разработано приложение “Корзина покупателя”.