#### 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 №6

по дисциплине

"Основы разработки веб приложений"

Тема: "Использование Docker для запуска приложений на Laravel"

Выполнил:

ст. гр. ПРИ-117

В.Г. Хлызова

Принял:

Козаев А.Т.

Владимир, 2020 г.

## ЦЕЛЬ РАБОТЫ

Познакомиться со средством контейнеризации Docker и использовать его для запуска приложения на Laravel.

ХОД РАБОТЫ

В папке с проектом из прошлой лабораторной работы создаем Dockerfile со следующим содержимым:

FROM php:7.4.5-fpm

RUN apt-get update \

&& docker-php-ext-install pdo mysqli pdo\_mysql && \

pecl install -o -f redis \

&& rm -rf /tmp/pear \

&& docker-php-ext-enable redis

WORKDIR /var/www

COPY . /var/www

EXPOSE 9000

CMD ["php-fpm"]

В этой же директории создаем файл docker-compose.yml. В конце файла добавляем ещё два сервиса: db – для взаимодействия приложения с бд и redis- для хэширования:

version: "3.2"

services:

 nginx:

   image: nginx:alpine

   volumes:

     - .:/var/www

     - ./nginx/conf.d/:/etc/nginx/conf.d/

   ports:

     - 81:81

     - 443:443

   depends\_on:

     - php

 php:

   build:

     context: .

   volumes:

    - .:/var/www

   ports:

     - 9000:9000

   depends\_on:

     - db

     - redis

 db:

   image: mysql:5.7.22

   ports:

    - 3307:3306

   environment:

     MYSQL\_DATABASE: lab\_laravel

     MYSQL\_ROOT\_PASSWORD: root

 redis:

  image: redis:4.0

  ports:

    - 6379:6379

Создаем файл по пути nginx/conf.d/default.conf, его содержимое:

server {

   listen 81;

   index index.php product.plade.php;

   error\_log  /var/log/nginx/error.log;

   access\_log /var/log/nginx/access.log;

   root /var/www/public;

   location ~ \.php$ {

       try\_files $uri =404;

       fastcgi\_split\_path\_info ^(.+\.php)(/.+)$;

       fastcgi\_pass php:9000;

       fastcgi\_index index.php;

       include fastcgi\_params;

       fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

       fastcgi\_param PATH\_INFO $fastcgi\_path\_info;

   }

   location / {

       try\_files $uri $uri/ /index.php?$query\_string;

       gzip\_static on;

   }

}

В файле .env изменяем подключение к бд и CACHE\_DRIVER, меняем REDIS\_HOST:

DB\_CONNECTION=mysql

DB\_HOST=db

DB\_PORT=3306

DB\_DATABASE=lab\_laravel

DB\_USERNAME=root

DB\_PASSWORD=root

BROADCAST\_DRIVER=log

CACHE\_DRIVER=redis

QUEUE\_CONNECTION=sync

SESSION\_DRIVER=file

SESSION\_LIFETIME=120

REDIS\_HOST=redis

REDIS\_PASSWORD=null

REDIS\_PORT=6379

Выполняем миграцию и процесс сидирования для добавления записей в бд. В файл ProductsController.php добавляем вывод данных из кэша (при удалении или добавлении новых товаров кэш сбрасывается):

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Product;

use Session;

use App\Cart;

class ProductsController extends Controller

{

  public function index()

  {

    if(\Cache::get('products')==null)

      \Cache::put('products', Product::all());

    $products = \Cache::get('products');

    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();

      \Cache::put('products', null);

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

  }

  public function delete($id)

  {

      Product::find($id)->delete();

      \Cache::put('products', null);

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

  }

}

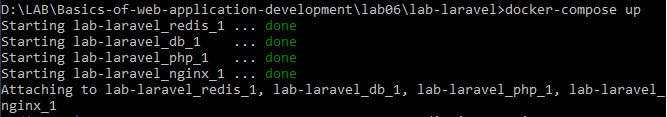


Рисунок 1. Запуск описанных сервисов.

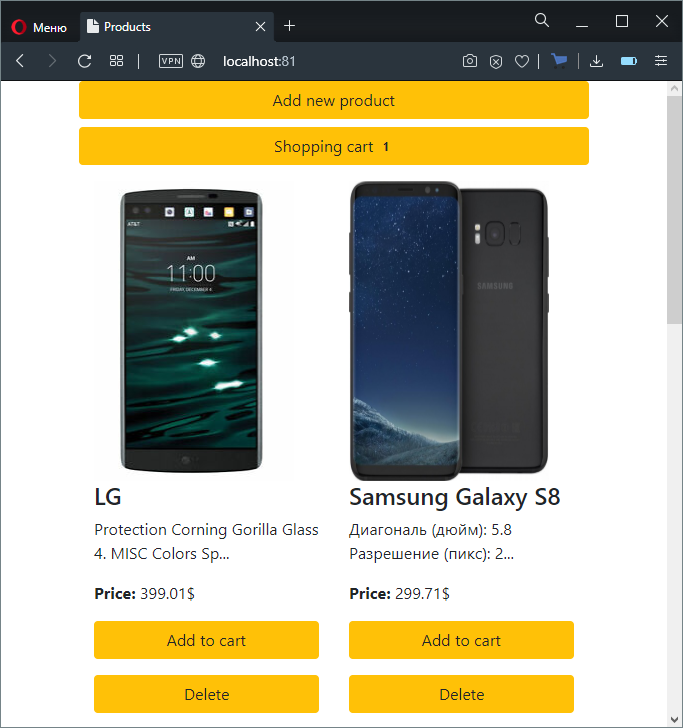


Рисунок 2. Страница приложения с выводом всех продуктов (из кэша).

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

ВЫВОД

В ходе лабораторной работы было изучено средство контейнеризации Docker и использование его для запуска приложения на Laravel.