МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»

(БГТУ им В. Г. Шухова)

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №3

Дисциплина: Технологии WEB-программирования

“Серверное программирование. Laravel”

Выполнил: ст. группы ПВ-41

Медведев Е. В.

Проверил: Картамышев С. В.

Белгород, 2021

**Цель:** познакомиться с основами backend разработки web-приложений. Научится писать скрипты на языке PHP. Познакомиться с основами работы docker. Познакомиться с фреймворком Laravel Lumen и научиться разворачивать проект, производить его настройку. Научится работать с API в приложении Postman.

**Задание к лабораторной работе:**

1. Развернуть базовое приложение Lumen.

2. Настроить конфигурацию работы приложения с docker.

3. Добавить модуль для работы с API.

4. Добавить несколько контроллеров со статическими данными.

5. Продемонстрировать работу API в Postman.

**Код программы:**

Dockerfile

FROM php:7.4-fpm  
  
# Copy composer.lock and composer.json  
COPY composer.json /var/www/  
  
# Set working directory  
WORKDIR /var/www  
  
# Install dependencies  
RUN apt-get update && apt-get install -y \  
 build-essential \  
 libpng-dev \  
 libjpeg62-turbo-dev \  
 libfreetype6-dev \  
 locales \  
 libzip-dev \  
 zip \  
 jpegoptim optipng pngquant gifsicle \  
 vim \  
 unzip \  
 git \  
 curl  
  
# Clear cache  
RUN apt-get clean && rm -rf /var/lib/apt/lists/\*  
  
# Install extensions  
RUN docker-php-ext-install pdo\_mysql zip exif pcntl  
RUN docker-php-ext-configure gd --with-freetype=/usr/include/ --with-jpeg=/usr/include/  
RUN docker-php-ext-install gd  
  
# Install composer  
RUN curl -sS https://getcomposer.org/installer | php -- --install-dir=/usr/local/bin --filename=composer  
  
# Add user for laravel application  
RUN groupadd -g 1000 www  
RUN useradd -u 1000 -ms /bin/bash -g www www  
  
# Copy existing application directory contents  
COPY . /var/www  
  
# Copy existing application directory permissions  
COPY --chown=www:www . /var/www  
  
# Change current user to www  
USER www  
  
# Expose port 9000 and start php-fpm server  
EXPOSE 9000  
CMD ["php-fpm"]

.env

APP\_NAME=Laravel  
APP\_ENV=local  
APP\_KEY=base64:UKcUYcIot11sGAbyTFR+jivcJTiVPrh+zkHNLmIf60g=  
APP\_DEBUG=true  
APP\_URL=http://localhost  
  
LOG\_CHANNEL=stack  
LOG\_DEPRECATIONS\_CHANNEL=null  
LOG\_LEVEL=debug  
  
DB\_CONNECTION=mysql  
DB\_HOST=localhost  
DB\_PORT=3306  
DB\_DATABASE=web  
DB\_USERNAME=root  
DB\_PASSWORD=super123  
  
BROADCAST\_DRIVER=log  
CACHE\_DRIVER=file  
FILESYSTEM\_DRIVER=local  
QUEUE\_CONNECTION=sync  
SESSION\_DRIVER=file  
SESSION\_LIFETIME=120  
  
MEMCACHED\_HOST=127.0.0.1  
  
REDIS\_HOST=127.0.0.1  
REDIS\_PASSWORD=null  
REDIS\_PORT=6379  
  
MAIL\_MAILER=smtp  
MAIL\_HOST=mailhog  
MAIL\_PORT=1025  
MAIL\_USERNAME=null  
MAIL\_PASSWORD=null  
MAIL\_ENCRYPTION=null  
MAIL\_FROM\_ADDRESS=null  
MAIL\_FROM\_NAME="${APP\_NAME}"  
  
AWS\_ACCESS\_KEY\_ID=  
AWS\_SECRET\_ACCESS\_KEY=  
AWS\_DEFAULT\_REGION=us-east-1  
AWS\_BUCKET=  
AWS\_USE\_PATH\_STYLE\_ENDPOINT=false  
  
PUSHER\_APP\_ID=  
PUSHER\_APP\_KEY=  
PUSHER\_APP\_SECRET=  
PUSHER\_APP\_CLUSTER=mt1  
  
MIX\_PUSHER\_APP\_KEY="${PUSHER\_APP\_KEY}"  
MIX\_PUSHER\_APP\_CLUSTER="${PUSHER\_APP\_CLUSTER}"

Docker-compose

version: '3'  
services:  
  
 app:  
 build:  
 context: .  
 dockerfile: ./docker/Dockerfile  
 image: digitalocean.com/php  
 restart: unless-stopped  
 tty: true  
 environment:  
 SERVICE\_NAME: app  
 SERVICE\_TAGS: dev  
 working\_dir: /var/www  
 volumes:  
 - ./:/var/www  
 - ./php/local.ini:/usr/local/etc/php/conf.d/local.ini  
 networks:  
 - app-network  
  
 webserver:  
 image: nginx:alpine  
 restart: unless-stopped  
 tty: true  
 ports:  
 - "80:80"  
 - "443:443"  
 volumes:  
 - ./:/var/www  
 - ./nginx/conf.d/:/etc/nginx/conf.d/  
 networks:  
 - app-network  
  
 db:  
 image: mysql:5.7  
 restart: unless-stopped  
 tty: true  
 ports:  
 - "3306:3306"  
 environment:  
 MYSQL\_ROOT\_PASSWORD: 'super123'  
 MYSQL\_DATABASE: 'web'  
 MYSQL\_USER: 'qwerty'  
 MYSQL\_PASSWORD: '123'  
 networks:  
 - app-network  
  
networks:  
 app-network:  
 driver: bridge

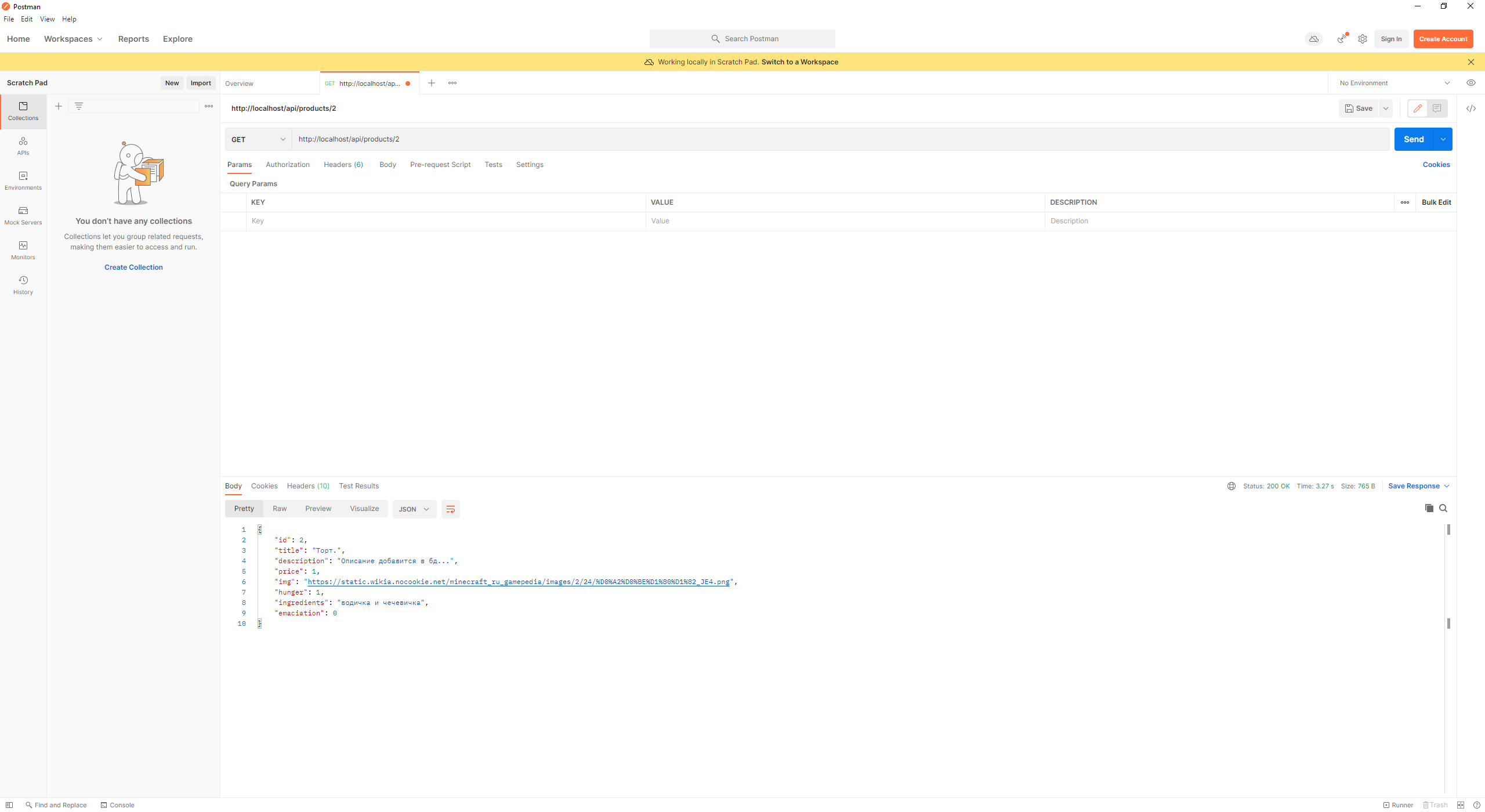
UserContoller

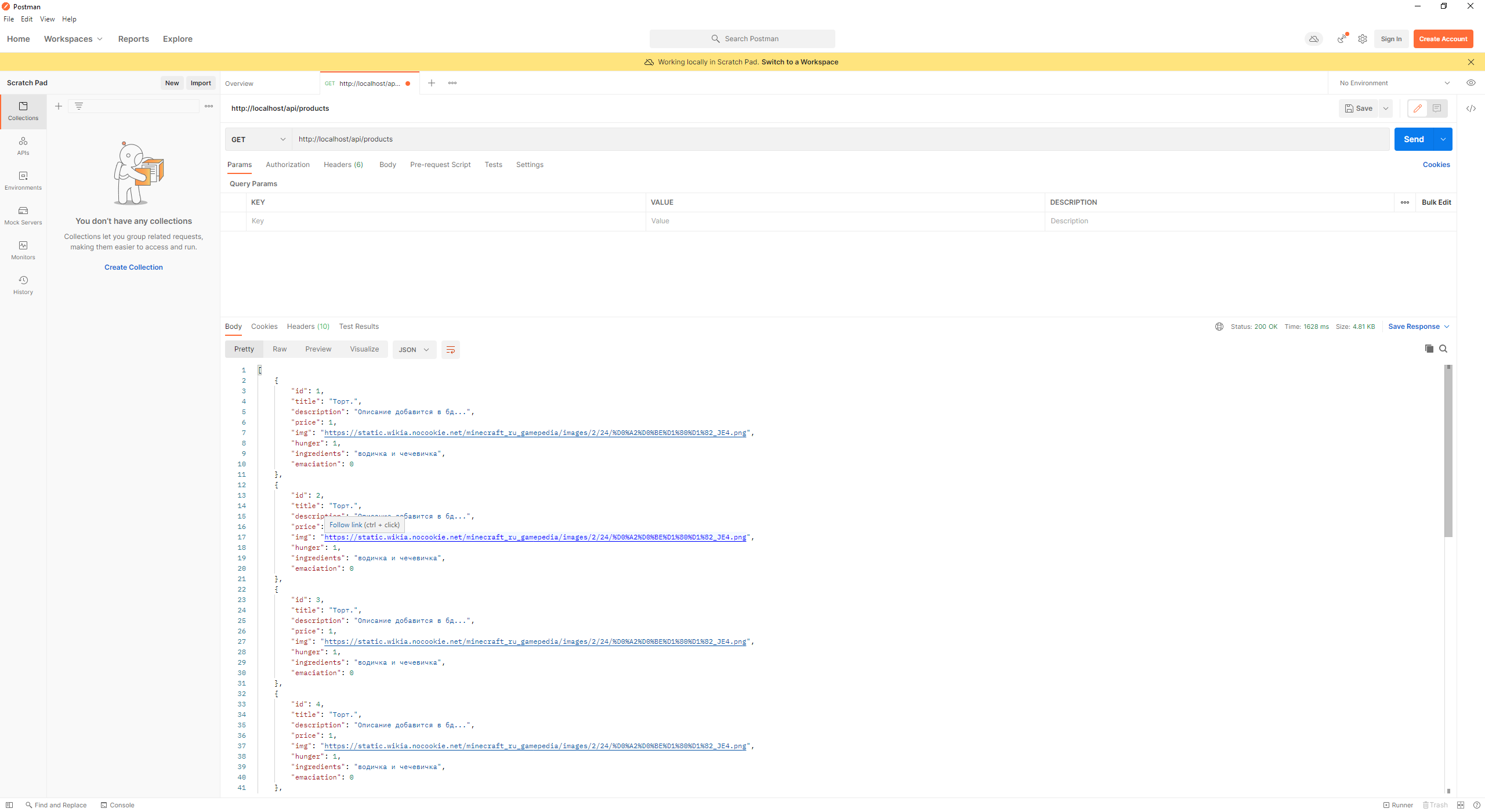
<?php  
  
namespace App\Http\Controllers;  
  
use Illuminate\Routing\Controller;  
use Illuminate\Http\Request;  
  
class UserController extends Controller  
{  
 public $arr = array(  
 [  
 'id' => *1*,  
 'login' => 'admin',  
 'password' => 'password'  
 ],  
 [  
 'id' => *2*,  
 'login' => 'biba',  
 'password' => '12345'  
 ],  
 [  
 'id' => *3*,  
 'login' => 'boba',  
 'password' => 'qwerty'  
 ]  
 );  
  
 /\*\*  
 \* Список пользователей.  
 \* @return array[]  
 \*/  
 public function list()  
 {  
 return $this->arr;  
 }  
  
 /\*\*  
 \* Информация о пользователе  
 \* @param $id  
 \* @return array  
 \*/  
 public function info($id)  
 {  
 return $this->arr[$id-*1*];  
 }  
  
 public function authorization(Request $request)  
 {  
 foreach ($this->arr as $value) {  
 if ($value['login'] == $request->get('login') && $value['password']== $request->get('password')){  
 return [  
 'userId'=> $value['id']  
 ];  
 }  
 }  
 return [  
 'userId'=> -*1* ];  
 }  
}

ProductController

<?php  
  
namespace App\Http\Controllers;  
  
use Illuminate\Routing\Controller;  
  
/\*\*  
\* Контроллер для управления товарами  
\*/  
class ProductController extends Controller  
{  
 public $arr = array(  
 [  
 'id' => *1*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *2*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *3*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *4*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *5*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *6*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *7*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *8*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *9*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ],  
 [  
 'id' => *10*,  
 'title' => 'Торт.',  
 'description' => 'Описание добавится в бд...',  
 'price' => *1*,  
 'img' => 'https://static.wikia.nocookie.net/minecraft\_ru\_gamepedia/images/2/24/%D0%A2%D0%BE%D1%80%D1%82\_JE4.png',  
 'hunger' => *1*,  
 'ingredients' => 'водичка и чечевичка',  
 'emaciation' => *0.0* ]  
 );  
 /\*\*  
 \* Список товаров.  
 \* @return array[]  
 \*/  
 public function list()  
 {  
 return $this->arr;  
 }  
  
 /\*\*  
 \* Информация о товаре  
 \* @param $id  
 \* @return array  
 \*/  
 public function info($id)  
 {  
 foreach($this->arr as $item){  
 if ($item['id'] == $id)  
 return $item;  
 }  
 }  
}

Работа с postman





Cсылка на репозиторий: https://github.com/Evgeniy400/web2/tree/web3