**ВВЕДЕНИЕ**

Была задача – создать интернет – магазин для компании True Games, в котором есть часть для неавторизованного пользователя (каталог, детальная карточка товара, страница «о нас» и страница «где нас найти?»), часть для авторизованного пользователя (добавление товара в коризну и оформление заказа) и админка (добавление, удаление и рекдактирование товара, управление заказами). Также необходимо было сделать регистрацию и авторизацию пользователя

Использованные технологии: PHP фреймворк Laravel – для написания логики приложения, CSS и JavaScript для оформления внешней части приложения и клиентской логики (отправление запросов на сервер). Вся работа велась через Github

**СОЗДАНИЕ ИНТЕРНЕТ МАГАЗИНА ДЛЯ КОМПАНИИ TRUE GAMES**

Весь проект находится в публичном репозитории на Github (ссылка - <https://github.com/IgorLiovin105/true-games>), где видно, когда происходили изменения и что было добавлено в проект

10.02.2023 (Пятница) была сделана инициализация проекта на Laravel и репозитория на Github, а также коммит для проверки работы проекта

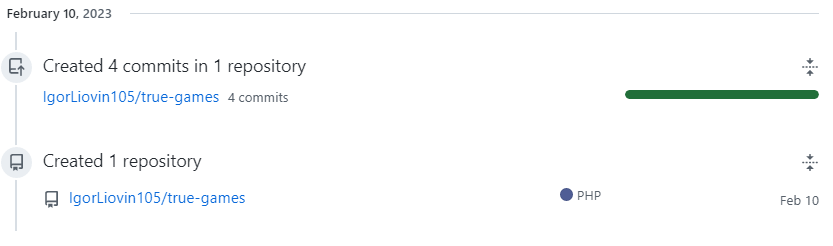


Рисунок 1 – Коммиты 10.02.2023

11.02.2023 (Суббота) был сделан шаблон для страниц, прописана логика регистрации, где производится проверка данных пользователя, создание самого пользователя, хеширование пароля, логика авторизации пользователя, где проверяются данные пользователя и выполняется вход в аккаунт, а также выход пользователя из аккаунта

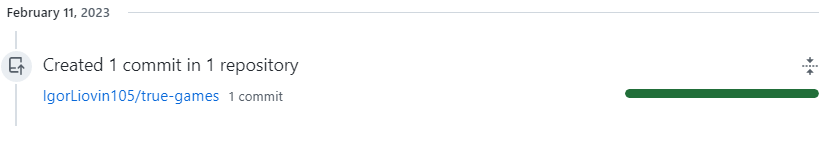


Рисунок 2 - Коммиты 11.02.2023

13.02.2023 (Понедельник) была сделана страница, в которой выводятся все имеющиеся на сайте товары, которые есть в наличии. Выводится картинка товара, название и цена

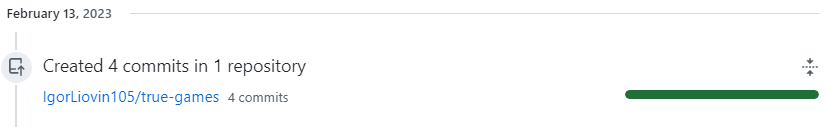


Рисунок 3 – Коммиты 13.02.2023

14.02.2023 (Вторник) был сделан вывод детальной страницы товара. В ней выводится картинка товара, название, цена, количество данного товара в магазине, описание, модель, страну и производства. Также было добавлено условие, чтобы при нулевом количестве выводилось, что товара нет в наличии и его нельзя добавить в коризну

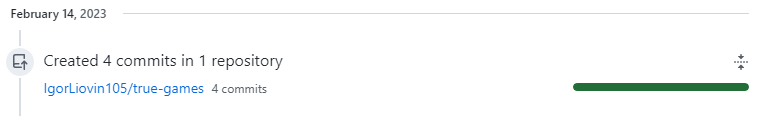


Рисунок 4 – Коммиты 14.02.2023

15.02.2023 (Среда) было сделано добавление в корзину. Данная функция работает без перезагрузки страницы, так как написан Fetch запрос к серверу на JavaScript. В навигационном меню у авторизованного пользователя есть иконка корзины с количеством товаров в ней и при добавлении товара оно меняется, было прописано условие, по которому нельзя добавить товар в корзину неавторизованному пользователю. Также в этот день была добавлена первая функция для админа – публикация товара в магазине. В первую очередь была добавлена форма добавления товара, где можно было вбить все данные о товаре. Затем была реализована логика добавления с выгрузкой картинки на сервер. После отправки формы и создания товара, админа перебрасывает на страницу каталога

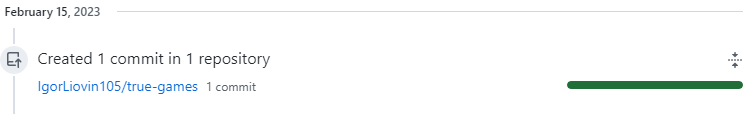


Рисунок 5 – Коммиты 15.02.2023

16.02.2023 (Четверг) были усовершенствованы каталог и корзина. Изначально в коризну можно было добавить только одну штуку любого товара, но была добавлена возможность изменять количество товара в корзине и добавлять один товар сколько угодно раз, пока он есть в наличии. Также в самой корзине был сделан вывод цены всех товаров, добавленных в корзину, с учётом их количества

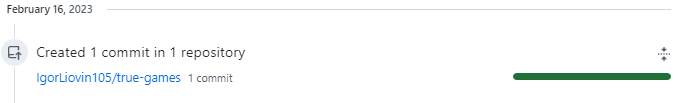


Рисунок 6 – Коммиты 16.02.2023

17.02.2023 (Пятница) была добавлена возможность оформления заказа. На странице корзины была добавлена кнопка для оформления заказа, после чего нужно было подтвердить логин и пароль пользователя. После подтверждения, пользователя кидало на страницу профиля, где отображались все его активные заказы

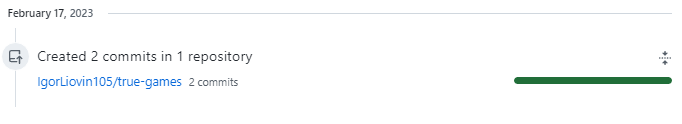


Рисунок 7 – Коммиты 17.02.2023

18.02.2023 (Суббота) была сделана детальная страница для каждого заказа, где был написан номер заказа и все товары с количеством и ссылкой на детальную страницу товара

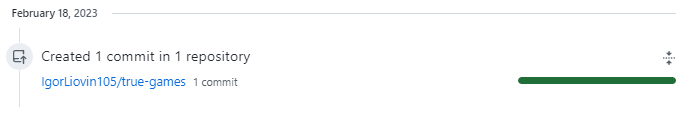


Рисунок 8 – Коммиты 18.02.2023

20.02.2023 (Понедельник) была сделана функция отмены заказа. Для этого на странице заказа, либо в профиле необходимо нажать на кнопку «Отменить заказ». После чего админ должен зайти в данный заказ, указать причину его отмены и подтвердить. Далее пользователь более не сможет посмотреть данный заказ. Также для админа была добавлена функция удаления товара в каталоге и страница, на которой он может просматривать все заказы и видеть заказы, помеченные на отмену. В последнюю очередь была добавлена возможность фильтрации товаров по категориям

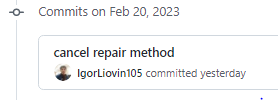


Рисунок 9 – Коммиты 20.02.2023

**ПРИЛОЖЕНИЕ А**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<meta name="token" content="{{ csrf\_token() }}">

<title>@yield('title') | True Games</title>

<link rel="stylesheet" href="{{ asset('css/app.css') }}">

</head>

<body>

<div class="wrapper">

@include('layouts.header')

<main class="main">

@yield('content')

</main>

</div>

<script src="{{ asset('js/app.js') }}"></script>

</body>

</html>

**ПРИЛОЖЕНИЕ Б**

<header class="header">

<div class="header\_\_container">

<a href="{{ route('main') }}" class="header\_\_logo">Logo</a>

<button id="menu-icon" class="header\_\_icon"><span></span><span></span><span></span></button>

<nav id="menu" class="header\_\_menu menu">

<ul class="menu\_\_list">

@guest

<li><a href="{{ route('register') }}" class="menu\_\_link">Регистрация</a></li>

<li><a href="{{ route('login') }}" class="menu\_\_link">Вход</a></li>

@endguest

@auth

@if (auth()->user()->login == 'admin')

<li><a href="{{ route('repairsPage') }}" class="menu\_\_link">Посмотреть заказы</a></li>

<li><a href="{{ route('createProduct') }}" class="menu\_\_link">Создать товар</a></li>

@endif

<li><a href="{{ route('cart') }}" class="menu\_\_link">Коризна <span style="color: #fff"

id="cart-quantity">{{ App\Models\Cart::where('user\_id', auth()->user()->id)->where('status', 'added')->count() }}</span></a>

</li>

<li><a href="{{ route('profile') }}" class="menu\_\_link">{{ auth()->user()->name }}</a></li>

<li>

<form action="{{ route('logout') }}" method="POST">

@csrf

<button type="submit" class="menu\_\_link">Выход</button>

</form>

</li>

@endauth

</ul>

</nav>

</div>

</header>

**ПРИЛОЖЕНИЕ В**

// Register method

public function register(RegisterRequest $request)

{

$data = $request->validated();

$user = User::create([

'name' => $data['name'],

'surname' => $data['surname'],

'patronymic' => $data['patronymic'],

'login' => $data['login'],

'email' => $data['email'],

'password' => bcrypt($data['password']),

]);

if($user) {

auth('web')->login($user);

return redirect()->route('profile');

}

return back();

}

// Login method

public function login(LoginRequest $request)

{

$data = $request->validated();

if(auth('web')->attempt($data)) {

return redirect()->route('profile');

}

return back();

}

// Logout method

public function logout()

{

auth('web')->logout();

return redirect()->route('main');

}

**ПРИЛОЖЕНИЕ Г**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Главная')

{{-- Sedtion content --}}

@section('content')

<section class="catalog">

<ul class="categories\_\_list">

@foreach($categories as $category)

<li><a href="{{ route('category', $category->id) }}" class="category\_\_link">{{ $category->name }}</a></li>

@endforeach

</ul>

<ul class="catalog\_\_list">

@foreach ($products as $product)

<li>

<a href="{{ route('detail', $product->id) }}" class="catalog\_\_link">

<img class="catalog\_\_img" src="{{ asset('img/' . $product->img) }}" alt="">

<h3 class="catalog\_\_title">{{ $product->name }}</h3>

<p class="catalog\_\_price">{{ $product->price }}₽</p>

@auth

@if(auth()->user()->login == 'admin')

<div class="catalog\_\_delete"><a href="{{ route('deleteProduct', $product->id) }}">Удалить товар</a></div>

@endif

@endauth

</a>

</li>

@endforeach

</ul>

</section>

@endsection

**ПРИЛОЖЕНИЕ Д**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', $product->name)

{{-- Sedtion content --}}

@section('content')

<section class="detail">

<div class="detail\_\_content">

<img src="{{ asset('img/' . $product->img) }}" alt="" class="detail\_\_img">

<div class="detail\_\_info">

<h2 class="detail\_\_title">{{ $product->name }}</h2>

<p class="detail\_\_description">{{ $product->description }}</p>

<p class="detail\_\_setting">Модель - {{ $product->model }}</p>

<p class="detail\_\_setting">Страна производства - {{ $product->country }}</p>

<p class="detail\_\_setting">Год производства - {{ $product->year }}</p>

<p class="detail\_\_setting">Категория - <a href="{{ route('category', $product->category->id) }}">{{ $product->category->name }}</a></p>

<p class="detail\_\_setting">Цена - {{ $product->price }}₽</p>

<p class="detail\_\_setting">Количество - <span id="product-quantity">{{ $product->quantity }}</span></p>

<p id="response" style="display: none"></p>

@guest

<a href="{{ route('login') }}" class="detail\_\_login">Чтобы добавить товар в коризну необходимо войти в профиль</a>

@endguest

@auth

<form id="add-to-cart" action="">

@if($product->quantity < 1)

<p>Данного товара нет в наличии</p>

@else

<button id="add-to-cart-button" data-id="{{ $product->id }}" class="detail\_\_cart" type="submit">Добавить в корзину</button>

@endif

</form>

@endauth

</div>

</div>

</section>

<script src="{{ asset('/js/cart.js') }}"></script>

@endsection

**ПРИЛОЖЕНИЕ Е**

const addToCart = document.getElementById('add-to-cart')

const token = document.head.querySelector('[name="token"][content]').content

const id = document.getElementById('add-to-cart-button').dataset.id

let cartQuantity = document.getElementById('cart-quantity')

let productQuantity = document.getElementById('product-quantity')

addToCart.addEventListener('submit', function(e) {

e.preventDefault()

fetch('/add-to-cart', {

method: 'POST',

headers: {

'Content-type': 'application/json',

'X-CSRF-TOKEN': token,

},

body: JSON.stringify({id}),

}).then(res => res.json()).then(res => {

if(res.status == 405) {

alert(res.message)

return console.log(res)

}

productQuantity.innerHTML = res.productQuantity,

cartQuantity.innerHTML = res.cartQuantity,

alert(res.message)

return console.log(res)

})

})

**ПРИЛОЖЕНИЕ Ё**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Коризна')

{{-- Sedtion content --}}

@section('content')

<section class="cart">

<div class="cart\_\_content">

@if($price > 0)

<a href="{{ route('userCheck') }}" class="cart\_\_repair" type="submit">Оформить заказ на {{ $price }}₽</a>

@endif

@forelse ($cartItems as $item)

<div class="cart\_\_item">

<img src="{{ asset('img/' . $item->product->img) }}" alt="">

<h2>{{ $item->product->name }}</h2>

<div class="cart\_\_quantity">

<a href="{{ route('changeQuantity', ['id' => $item->id, 'method' => 'incr']) }}">+</a>

<p>{{ $item->quantity }}</p>

<a href="{{ route('changeQuantity', ['id' => $item->id, 'method' => 'decr']) }}">-</a>

</div>

<p>{{ $item->product->price }}₽</p>

<p>total: {{ $item->summary\_price }}₽</p>

<a href="{{ route('deleteFromCart', $item->id) }}" class="cart\_\_delete">Удалить товар</a>

</div>

@empty

<h1 class="cart\_\_empty">В корзине нет товаров</h1>

@endforelse

</div>

</section>

@endsection

**ПРИЛОЖЕНИЕ Ж**

public function index()

{

$products = Product::where('quantity', '>', 0)->get();

$categories = Category::all();

return view('pages.index', compact('products', 'categories'));

}

public function category($id)

{

$products = Product::where('quantity', '>', 0)->where('category\_id', $id)->get();

$categories = Category::all();

return view('pages.index', compact('products', 'categories'));

}

public function detail($id)

{

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

return view('pages.detail', compact('product'));

}

public function cart()

{

$cartItems = Cart::where('user\_id', auth()->user()->id)->where('status', 'added')->get();

$price = $cartItems->sum('summary\_price');

return view('pages.cart', compact(['cartItems', 'price']));

}

public function addToCart(Request $request)

{

$product = Product::find($request->id);

if($product->quantity < 1) {

return response()->json(['message' => 'Вы не можете добавить данный товар в коризну, так как его нет в наличии'], 200);

}

if($cart = Cart::where('user\_id', auth()->user()->id)->where('product\_id', $product->id)->where('status', 'added')->first()) {

$cart->quantity++;

$cart->summary\_price = $cart->quantity \* $product->price;

$cart->save();

}

else {

Cart::create([

'user\_id' => auth()->user()->id,

'product\_id' => $request->id,

'summary\_price' => Product::find($request->id)->price,

'quantity' => 1

]);

}

$product->quantity--;

$product->save();

$cartCount = Cart::where('user\_id', auth()->user()->id)->where('status', 'added')->count();

return response()->json(['message' => 'Товар добавлен в коризну', 'productQuantity' => $product->quantity, 'cartQuantity' => $cartCount], 200);

}

public function changeQuantity($id, $method)

{

$cart = Cart::find($id);

if($method == 'incr' && $cart->product->quantity > 0) {

$cart->quantity++;

$cart->product->quantity--;

$cart->summary\_price = $cart->product->price \* $cart->quantity;

$cart->product->save();

$cart->save();

}

if($method == 'decr' && $cart->quantity > 1) {

$cart->quantity--;

$cart->product->quantity++;

$cart->summary\_price = $cart->product->price \* $cart->quantity;

$cart->product->save();

$cart->save();

}

return back();

}

public function deleteFromCart($id)

{

$cart = Cart::find($id);

$cart->product->quantity += $cart->quantity;

$cart->product->save();

Cart::destroy($id);

return back();

}

}

**ПРИЛОЖЕНИЕ З**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Проверка пользователя')

{{-- Sedtion content --}}

@section('content')

<form class="form" method="POST" action="{{ route('makeRepair') }}">

<input class="form\_\_text" name="login" type="text" placeholder="Введите логин">

@error('login')<p class="error">Логин набран неправильно</p>@enderror

<input class="form\_\_text" name="password" type="password" placeholder="Введите пароль">

@error('password')<p class="error">Пароль набран неправильно</p>@enderror

@csrf

<button type="submit">Войти</button>

</form>

@endsection

**ПРИЛОЖЕНИЕ И**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Профиль')

{{-- Sedtion content --}}

@section('content')

<section class="profile">

<h1 class="profile\_\_title">Привет {{ auth()->user()->name }}</h1>

<div class="profile\_\_list">

@foreach($repairs as $repair)

<div class="profile\_\_item">

<h2>Ваш заказ под номером {{ $repair->id }} находится в статусе: {{ $repair->status }}</h2>

<a href="{{ route('repair', $repair->id) }}">Посмотреть заказ детально</a>

</div>

@endforeach

</div>

</section>

@endsection

**ПРИЛОЖЕНИЕ К**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Заказ детально')

{{-- Sedtion content --}}

@section('content')

<section class="repair">

<h1 class="repair\_\_title">Заказ номер {{ $repairNumber->id }}</h1>

<h2 class="repair\_\_subtitle">Финальная цена: {{ $repairNumber->price }}₽</h2>

<h2 class="repair\_\_subtitle">Статус данного заказа: {{ $repairNumber->status }}</h2>

<ul class="repair\_\_list">

@foreach($repairItems as $item)

<li>

<a href="{{ route('detail', $item->product->id) }}" class="repair\_\_item">

<img src="{{ asset('img/' . $item->product->img) }}" alt="">

<h2>{{ $item->product->name }}</h2>

<p>Количество: {{ $item->quantity }}</p>

<p>Цена: {{ $item->summary\_price }}₽</p>

</a>

</li>

@endforeach

</ul>

<a href="{{ route('cancelRepair', $repairNumber->id) }}">Отменить заказ</a>

</section>

@endsection

**ПРИЛОЖЕНИЕ Л**

public function makeRepair(LoginRequest $request)

{

$data = $request->validated();

if(auth('web')->attempt($data)) {

$cartItems = Cart::where('user\_id', auth()->user()->id)->where('status', 'added')->get();

$repair = Repair::create([

'user\_id' => auth()->user()->id,

'price' => $cartItems->sum('summary\_price')

]);

Cart::where('user\_id', auth()->user()->id)->where('status', 'added')->update(['status' => 'formed', 'repair\_id' => $repair->id]);

return redirect()->route('profile');

}

return back();

}

public function repair($id)

{

$repairItems = Cart::where('repair\_id', $id)->get();

$repairNumber = Repair::find($id);

return view('pages.repair', compact('repairItems', 'repairNumber'));

}

public function cancelRepair(Request $request)

{

Repair::where('id', $request->id)->update(['status' => 'Ждёт отмены']);

return redirect()->route('profile');

}

}

**ПРИЛОЖЕНИЕ М**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Заказы')

{{-- Sedtion content --}}

@section('content')

<section class="canceled-repair">

<h1 class="canceled-repair\_\_title">Заказы</h1>

<div class="canceled-repair\_\_list">

@foreach($repairs as $repair)

<div class="canceled-repair\_\_item">

<h2>Заказ номер {{ $repair->id }} находится в статусе: {{ $repair->status }}</h2>

<p>Дата-время создания: {{ $repair->created\_at }}</p>

<p>Дата-время изменения: {{ $repair->updated\_at }}</p>

<p>Имя пользователя: {{ $repair->user->name }}</p>

@if($repair->status == 'Ждёт отмены')

<a href="{{ route('cancelRepairPage', $repair->id) }}">Отменить заказ</a>

@endif

</div>

@endforeach

</div>

</section>

@endsection

**ПРИЛОЖЕНИЕ Н**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Создать товар')

{{-- Sedtion content --}}

@section('content')

<section class="create">

<form class="create\_\_form" method="POST" action="{{ route('createProduct') }}" enctype="multipart/form-data">

@csrf

<input class="create\_\_form-text" type="text" name="name" placeholder="Enter name">

<input type="file" name="img" placeholder="Load image">

<input class="create\_\_form-text" type="text" name="description" placeholder="Enter description">

<input class="create\_\_form-text" type="text" name="model" placeholder="Enter model">

<input class="create\_\_form-text" type="text" name="country" placeholder="Enter country">

<input class="create\_\_form-text" type="number" name="year" placeholder="Enter year">

<input class="create\_\_form-text" type="number" name="price" placeholder="Enter price">

<input class="create\_\_form-text" type="number" name="quantity" placeholder="Enter quantity">

<select name="category\_id" id="">

<option value="">Выбрать категорию</option>

@foreach($categories as $category)

<option value="{{ $category->id }}">{{ $category->name }}</option>

@endforeach

</select>

<button type="submit">Опубликовать товар</button>

</form>

</section>

@endsection

**ПРИЛОЖЕНИЕ О**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Заказы')

{{-- Sedtion content --}}

@section('content')

<section class="canceled-repair">

<h1 class="canceled-repair\_\_title">Заказы</h1>

<div class="canceled-repair\_\_list">

@foreach($repairs as $repair)

<div class="canceled-repair\_\_item">

<h2>Заказ номер {{ $repair->id }} находится в статусе: {{ $repair->status }}</h2>

<p>Дата-время создания: {{ $repair->created\_at }}</p>

<p>Дата-время изменения: {{ $repair->updated\_at }}</p>

<p>Имя пользователя: {{ $repair->user->name }}</p>

@if($repair->status == 'Ждёт отмены')

<a href="{{ route('cancelRepairPage', $repair->id) }}">Отменить заказ</a>

@endif

</div>

@endforeach

</div>

</section>

@endsection

**ПРИЛОЖЕНИЕ П**

{{-- Extending layouts app --}}

@extends('layouts.app')

{{-- Section title --}}

@section('title', 'Отмена заказа')

{{-- Sedtion content --}}

@section('content')

<section class="cancel">

<div class="canter\_\_content">

<h1>Удаление заказа {{ $repair->id }}</h1>

<form method="POST" action="{{ route('cancelRepairProcess') }}" class="reason">

@csrf

<input type="text" name="reason">

<input type="hidden" name="id" value="{{ $repair->id }}">

<button type="submit">Отменить заказ</button>

</form>

</div>

</section>

@endsection

**ПРИЛОЖЕНИЕ Р**

public function createProductPage()

{

$categories = Category::all();

return view('admin.create-product', compact('categories'));

}

public function createProduct(Request $request)

{

$data = $request->validate([

'name' => 'required|string',

'img' => 'required|image|mimes:png,jpeg,jpg,svg',

'description' => 'required|string',

'model' => 'required',

'country' => 'required',

'year' => 'required',

'price' => 'required',

'quantity' => 'required',

'category\_id' => 'required',

]);

$imgName = time() . '-' . $request->img->getClientOriginalName();

$request->img->move(public\_path('img'), $imgName);

Product::create([

'name' => $data['name'],

'img' => $imgName,

'description' => $data['description'],

'model' => $data['model'],

'country' => $data['country'],

'year' => $data['year'],

'price' => $data['price'],

'quantity' => $data['quantity'],

'category\_id' => $data['category\_id'],

]);

return redirect()->route('main');

}

public function deleteProduct($id)

{

Product::destroy($id);

return redirect()->route('main');

}

public function repairsPage()

{

$repairs = Repair::all();

return view('admin.repairs', compact('repairs'));

}

public function cancelRepairPage($id)

{

$repair = Repair::find($id);

return view('admin.cancel-repair', compact('repair'));

}

public function cancelRepair(Request $request)

{

Repair::where('id', $request->id)->update(['status' => 'Отменён', 'reason' => $request->reason, 'is\_canceled' => true]);

return redirect()->route('repairsPage');

}

}

**ПРИЛОЖЕНИЕ С**

Route::get('/', [CatalogController::class, 'index'])->name('main');

Route::get('/catalog/{id}', [CatalogController::class, 'category'])->name('category');

Route::get('/detail/{id}', [CatalogController::class, 'detail'])->name('detail');

Route::middleware(['guest'])->group(function () {

Route::view('/login', 'pages.login')->name('login');

Route::view('/register', 'pages.register')->name('register');

Route::post('/register-process', [AuthController::class, 'register'])->name('registerProcess');

Route::post('/login-process', [AuthController::class, 'login'])->name('loginProcess');

});

Route::middleware(['auth'])->group(function () {

Route::get('/profile', [AuthController::class, 'profilePage'])->name('profile');

Route::post('/logout', [AuthController::class, 'logout'])->name('logout');

Route::get('/cart', [CatalogController::class, 'cart'])->name('cart');

Route::post('/add-to-cart', [CatalogController::class, 'addToCart'])->name('addToCart');

Route::get('/change-quantity/{id}/{method}', [CatalogController::class, 'changeQuantity'])->name('changeQuantity');

Route::get('/delete-from-cart/{id}', [CatalogController::class, 'deleteFromCart'])->name('deleteFromCart');

Route::view('/user-check', 'pages.user-check')->name('userCheck');

Route::post('/make-repair', [RepairController::class, 'makeRepair'])->name('makeRepair');

Route::get('/repair/{id}', [RepairController::class, 'repair'])->name('repair');

Route::get('/cancel-repair/{id}', [RepairController::class, 'cancelRepair'])->name('cancelRepair');

});

Route::middleware(['admin'])->group(function () {

Route::get('/admin/create-product', [AdminController::class, 'createProductPage'])->name('createProductPage');

Route::post('/admin/create-product', [AdminController::class, 'createProduct'])->name('createProduct');

Route::get('/admin/delete-product/{id}', [AdminController::class, 'deleteProduct'])->name('deleteProduct');

Route::get('/admin/repairs', [AdminController::class, 'repairsPage'])->name('repairsPage');

Route::get('/admin/cancel-repair-form/{id}', [AdminController::class, 'cancelRepairPage'])->name('cancelRepairPage');

Route::post('/admin/cancel-repair-process', [AdminController::class, 'cancelRepair'])->name('cancelRepairProcess');

});