

Проектна задача по

Имплементација на системи со отворен код

и

Напреден веб дизајн

Изработил:

Владимир Стојчески 171248

Професор:

Иван Чорбев

Асистент:

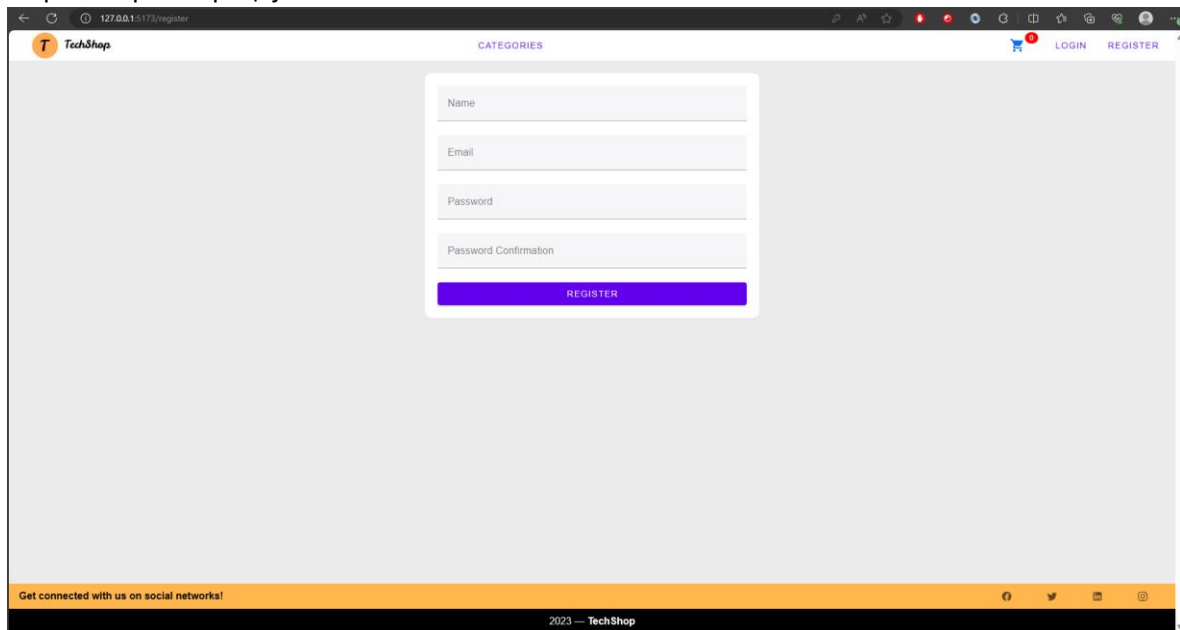
Бобан Јоксимоски

1. Вовед

Проектот е е-продавница која се состои од 2 независни апликации (backend и frontend). Бекендот е направел во Ларавел 9 додека фронтендот во Vue3. Апликацијата ги содржи основните функционалности за една е-продавница: Креирање на сметка, најава, листање на продукти, филтрирање по категорија и производител, додавање на продукти во кошничка, правење на нарачка, преглед и можност за ажурирање на податоци во профилот, историја на нарачки и додавање на нов продукт (за админ корисници).

2. Автентикација и авторизација

Форма за регистрација:

A screenshot of a web browser showing the registration form for 'TechShop'. The browser's address bar shows '127.0.0.1:5173/register'. The page has a light gray background. At the top left is the 'TechShop' logo. At the top center is the word 'CATEGORIES'. At the top right are links for 'LOGIN' and 'REGISTER'. The registration form is a white box in the center with four input fields: 'Name', 'Email', 'Password', and 'Password Confirmation'. Below these fields is a purple 'REGISTER' button. At the bottom of the page is a dark orange footer with the text 'Get connected with us on social networks!' and social media icons. The footer also shows '2023 — TechShop'.

Автентикацијата и авторизацијата работат со JWT.

Повик кон бекенд:

{host}/api/register (POST)

Body: {

```
"name" : "Vladimir Stojcheski",  
"email" : "stojcheskivladimir@gmail.com",  
"password" : "Password123!",  
"password_confirmation" : "Password123!"
```

}

Одговор: {

```
"message": "User successfully registered",
```

```

"user": {
  "name": "Vladimir Stojcheski",
  "email": "stojcheskivladimir@gmail.com",
  "updated_at": "2023-09-17T16:29:15.000000Z",
  "created_at": "2023-09-17T16:29:15.000000Z",
  "id": 6
},
"token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwOi8vbG9jYXRob3N0OjgwMDAvYXBpL3JlZ2lzdGVyIiwiaWF0IjoxNjk0OTY4MTU1LCJleHAiOiJlOTQ5NzE3NTUsIm5iZil6MTY5NDk2ODE1NSwianRpIjoibDIFYjRzZ3NwZGQwRG4yciIsInN1Yil6IjYiLCJwcnYiOiIyM2JkNW40OTQ5ZjYwMGFkYjM5ZTcwMWM0MDA4NzJkYjdhNTk3NmY3In0.nQPhR4P_1yO-GH0gU6Rj0q-NgDvvFgmAaz7U8og5Lhs"
}

```

Бекенд код за регистрација:

```

public function register(Request $request)
{
    $validator = Validator::make($request->all(), [
        'name' => 'required|string|between:2,100',
        'email' => 'required|string|email|max:100|unique:users',
        'password' => 'required|string|confirmed|min:6',
    ]);
    if($validator->fails()){
        return response()->json($validator->errors()->toJson(), status: 400);
    }

    $user = User::create([
        'name' => $request->get( key: 'name'),
        'email' => $request->get( key: 'email'),
        'password' => Hash::make($request->get( key: 'password')),
    ]);

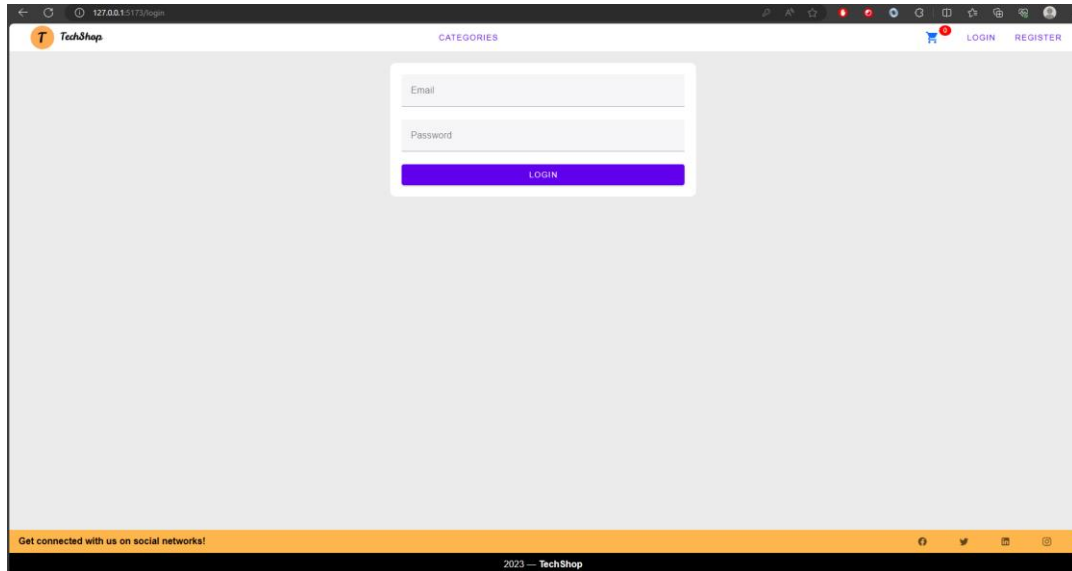
    $token = JWTAuth::fromUser($user);

    return response()->json([
        'message' => 'User successfully registered',
        'user' => $user,
        'token' => $token,
    ], status: 200);
}

```

За генерирање на токен се користи библиотеката: "tymon/jwt-auth": "^2.0"

По успешна регистрација, се појавува соодветна порака
Форма за најава:



Повик кон бекенд:

{host}/api/login (POST)

Body: {

"email" : "stojcheskivladimir@gmail.com",
"password" : "Password123!"

}

Одговор: {

"access_token":

"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwOi8vbG9jYWxob3N0OjgwMDAvYXBpL2xvZ2luliwiaWF0IjoxNjk0OTY4NDEzLCJleHAiOiJlOTQ5NzlwMTMslm5iZiI6MTY5NDk2ODQxMywianRpljoieXQ4S1lIZUxOOFBTcFJaVYlsInN1YiI6IjUuLCJwcnYiOiIyM2JkNWM4OTQ5ZjYwMGFkYjM5ZTcwMWM0MDA4NzJkYjdhNTk3NmY3In0.X2o-2666m4lldG6mkeXkHm9RBTCRDj9iyND_K6HgP-o",

"token_type": "bearer",

"expires_in": 3600

}

Бекенд код:

```

public function login(Request $request)
{
    $validator = Validator::make($request->all(), [
        'email' => 'required',
        'password' => 'required',
    ]);
    if ($validator->fails()) {
        return response()->json($validator->errors(), status: 422);
    }

    if (! $token = auth()->attempt($validator->validated())) {
        return response()->json(['error' => 'Unauthorized'], status: 401);
    }
    $user = User::where("email", $validator->getData()['email']->get();

    $token = JWTAuth::fromUser($user[0]);
    return $this->respondWithToken($token);
}

```

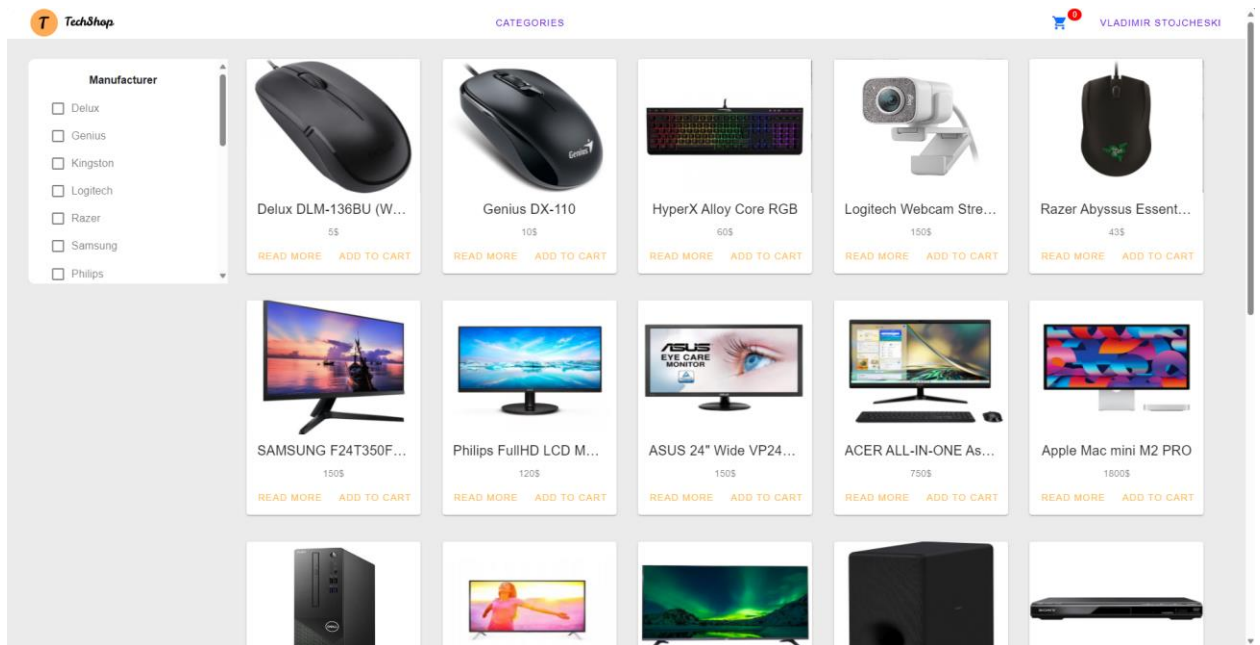
При успешна најава се појавува соодветна порака и се прави редирекција кон почетна страна.



Токенот што го добива фронтендот се чува во localStorage и за потребните барања кон бекенд се зема од таму.

3. Продавница

Почетна:



Тука се листаат сите продукти од база. Една картичка претставува еден продукт. Од левата страна има филтер за производители од типот на multiple choice select. Во хедерот има и

CATEGORIES

Computers and IT equipment

Computer accessories

TV/Audio/Video

Monitors

Appliances

Configurations

филтер за категории.

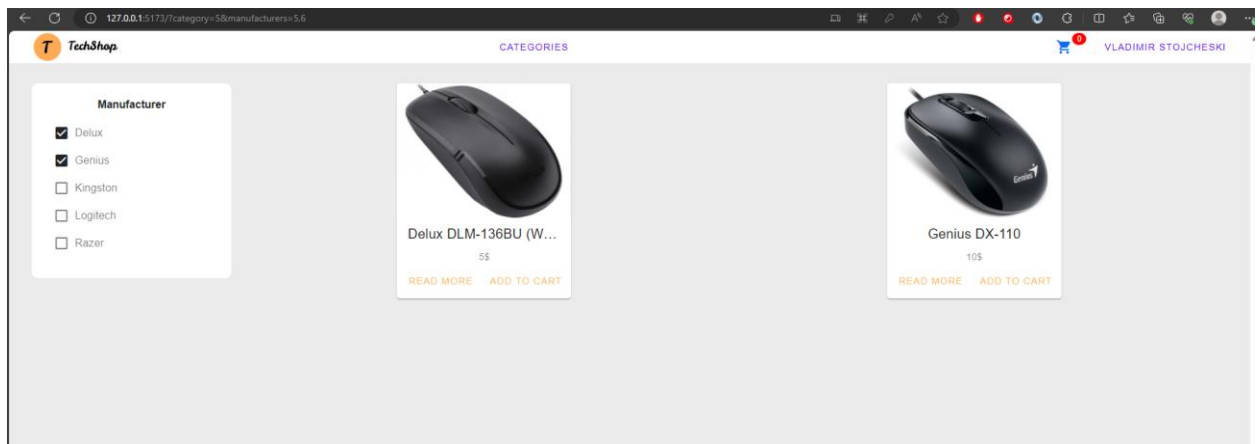
Категориите се состојат од:

Главни категории (Computers and IT equipment, TV/Audio/Video, Appliances) и секоја главна категорија си има свои подкатегории. Продуктите се поврзани со подкатегории, а главните категории служат само за полесна навигација.

Филтерите се запишуваат како query string во url и во store (Vuex).

Пример за филтер:

{host}/?category=5&manufacturers=5,6



За сите филтери се користи еден повик кон бекенд.

`{host}/products/all (POST)`

Body: search: {

```
"manufacturers": [
  6,
  5
],
"category": 5,
"manufacturersToFilter": [],
"query": ""
}
```

Одговор: [

```
{
  "id": 16,
  "created_at": "2023-09-17T02:22:15.000000Z",
  "updated_at": "2023-09-17T02:22:15.000000Z",
  "title": "Delux DLM-136BU (W&G) Wired Optical Mouse",
  "img":
"http://localhost:8000/storage/products/Wluy4caMo9ul00KR27CQWW15Kt1I46YPHzuqYpVp.png",
  "description": "<p><strong><span style=\"font-family: Roboto;\"><span style=\"font-size:
14px;\">Delux DLM-136BU (W&amp;G) Wired Optical Mouse</span></span></strong><br
/><strong><span style=\"font-family: Roboto;\"><span style=\"font-size:
14px;\">&nbsp;&nbsp;&nbsp;White&amp;Green</span></span></strong><br
/><strong><span style=\"font-family: Roboto;\"><span style=\"font-size:
14px;\">&nbsp;&nbsp;&nbsp;USB</span></span></strong><br
/><strong><span style=\"font-family: Roboto;\"><span style=\"font-size:
14px;\">&nbsp;&nbsp;&nbsp;Fashional outlook design</span></span></strong><br
/><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;Eudipleural ergonomic
design</span></span></strong><br
/><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;Resolution: 1.000 DPI</span></span></strong><br
/><strong><span style=\"font-family: Roboto;\"><span style=\"font-size:
14px;\">&nbsp;&nbsp;&nbsp;Keystroke Lifetime: 3</span></span></strong><br
/><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">000</span></span></strong><br
/></strong></p>"
```

```

/><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">000
cycles</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span
style=\"font-size: 14px;\">&nbsp;USB cable 160cm</span></span></strong></p>\",
  "price": 5,
  "category_id": 5,
  "manufacturer_id": 5
},
{
  "id": 17,
  "created_at": "2023-09-17T02:25:09.000000Z",
  "updated_at": "2023-09-17T02:25:09.000000Z",
  "title": "Genius DX-110",
  "img":
"http://localhost:8000/storage/products/ph7rbVIXHx7sANnLR7UCRzb1fhqiQoO0ePM6g6WR.pn
g",
  "description": "<p>Genius DX-110</p>\r\n<p>&nbsp;Wired
Mouse</p>\r\n<p>&nbsp;PS2</p>\r\n<p>&nbsp;Color: Black</p>\r\n<p>&nbsp;Resolution:
1200 DPI</p>\r\n<p>&nbsp;Buttons: Three buttons (left</p>\r\n<p>&nbsp;middle button with
scroll wheel</p>\r\n<p>&nbsp;right)</p>\r\n<p>&nbsp;Either hand</p>\",
  "price": 10,
  "category_id": 5,
  "manufacturer_id": 6
}
]

```

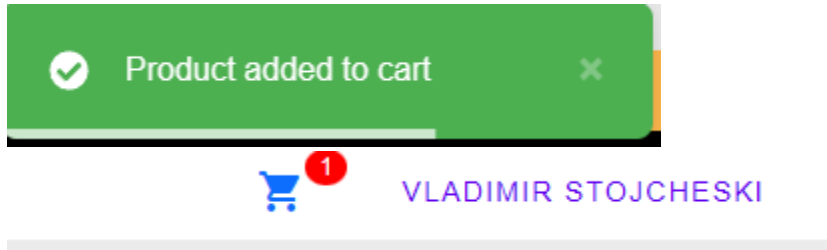
Бекенд код:

```

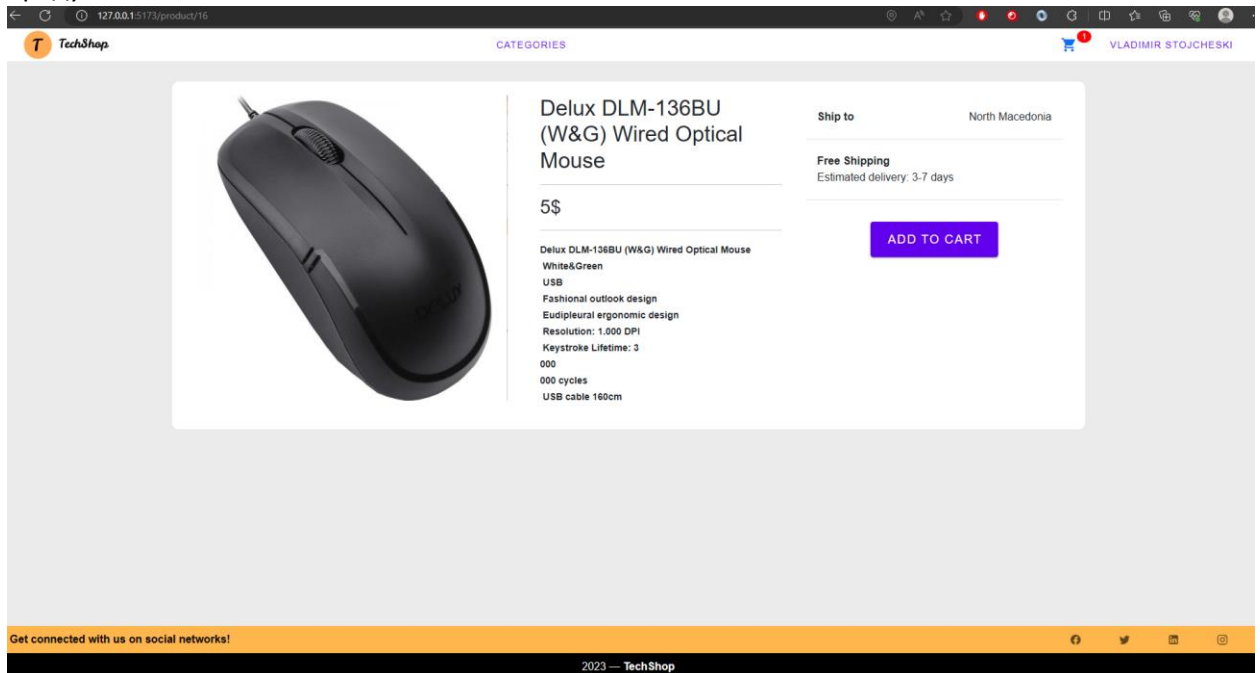
public function getAllFiltered(GetProductRequest $request)
{
    $search = $request->search;
    $listMan = $search['manufacturers'];
    $cat = $search['category'];
    if (isset($cat) && $cat != 0) {
        if (sizeof($listMan) > 0) {
            $allProducts = Product::where("category_id", $cat)->whereIn("manufacturer_id", $listMan)->get();
        } else {
            $allProducts = Product::where("category_id", $cat)->get();
        }
    } else {
        $allProducts = Product::whereIn("manufacturer_id", $listMan)->get();
    }
    foreach ($allProducts as $product) {
        $product->img = asset( path: 'storage/' . $product->img);
    }
    return $allProducts;
}

```


Со клик на копчето Add to cart на некој продукт, тој продукт се додава во кошничката и се зголемува бројот кај кошничката во хедерот.



Со клик на копчето Read more, не носи на нов поглед кој што ни се прикажуваат детали за продуктот:



Повик кон бекенд:

`{host}/products/{id} (GET)`

Одговор: [

```
{
  "id": 16,
  "created_at": "2023-09-17T02:22:15.000000Z",
  "updated_at": "2023-09-17T02:22:15.000000Z",
  "title": "Delux DLM-136BU (W&G) Wired Optical Mouse",
  "img":
"http://localhost:8000/storage/products/Wluy4caMo9ul00KR27CQWW15Kt1I46YPHzuqYpVp.png",
```

```

    "description": "<p><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">Delux DLM-136BU (W&G) Wired Optical Mouse</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;White&Green</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;USB</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;Fashional outlook design</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;Eudipleural ergonomic design</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;Resolution: 1.000 DPI</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;Keystroke Lifetime: 3</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;000</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;000 cycles</span></span></strong><br /><strong><span style=\"font-family: Roboto;\"><span style=\"font-size: 14px;\">&nbsp;&nbsp;&nbsp;USB cable 160cm</span></span></strong></p>",

    "price": 5,

    "category_id": 5,

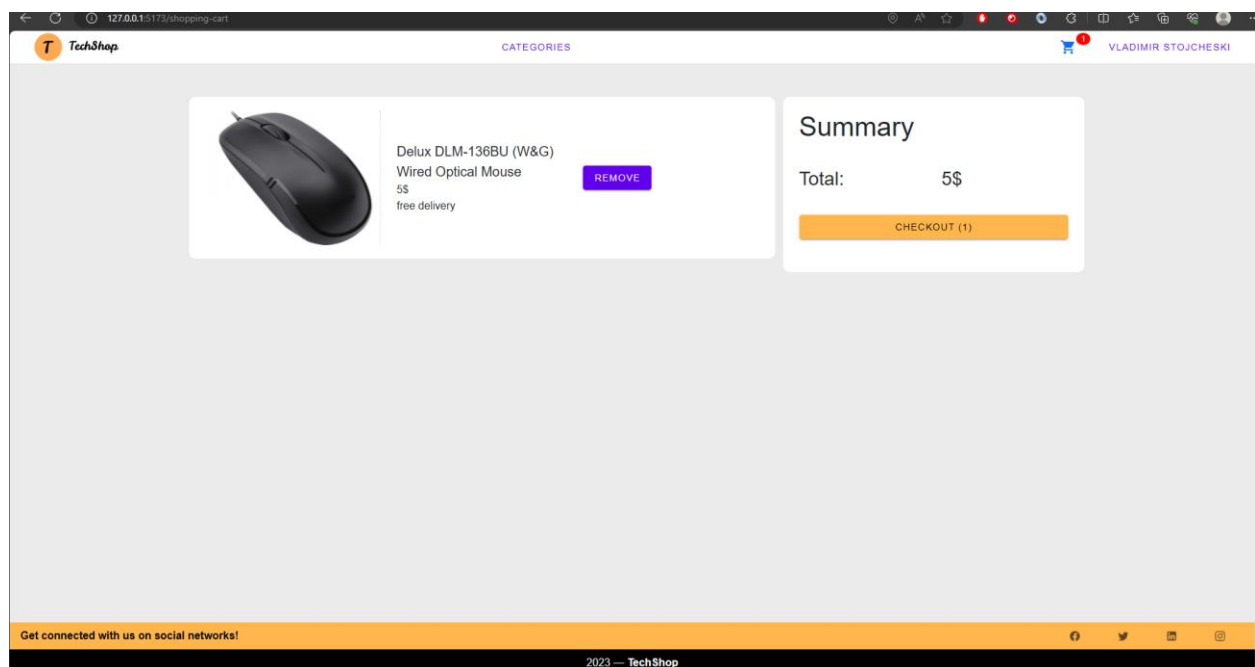
    "manufacturer_id": 5

  }

]

```

Со клик на кошничката во хедерот апликацијата не носи на следниот поглед:



Тука се листаат сите продукти во кошничката како и вкупната сума. Со клик на копчето remove, се бриши продуктот од кошничката.

Доколку сте најавени на апликацијата, со клик на името и потоа Profile, не носи на поглед за преглед на профил.

VLADIMIR STOJCHESKI

Profile

TechShop CATEGORIES VLADIMIR STOJCHESKI

Profile

First name Vladimir	Last name Stojcheski	Email stojcheski.vladimir@gmail.com
Country Macedonia	City Prilep	Phone number 3897100000
Street name Ulica Ulica 12	Zip 7500	SAVE

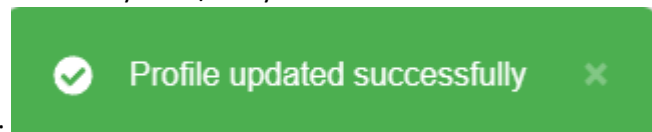
Order history
You have not made any orders yet!

Get connected with us on social networks!

2023 — TechShop

Тука може да се ажурираат личните податоци со пополнување/менување на полињата и клик на

копчето Save. Ке се прикаже соодветна порака:



Повик кон бекенд:

{host}/api/profile/edit (POST)

Headers: {

“Authorization” : “Bearer {token}”

}

Body: {

```
"phone_number": "3897100000",  
"country": "Macedonia",  
"city": "Prilep",  
"street_name": "Ulica Ulica 12",  
"zip": "7500"  
}  
Одговор: {  
  "message": "User successfully updated"  
}
```

Бекенд код:

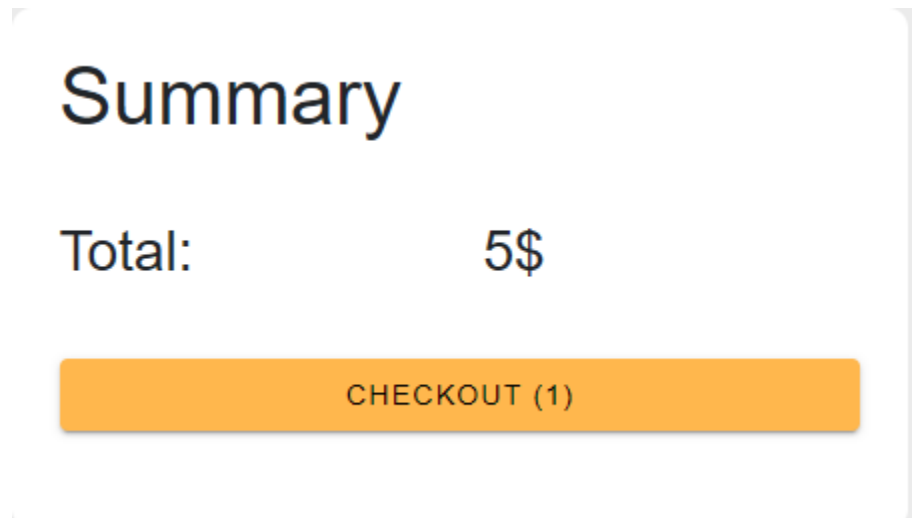
```
public function update(Request $request)  
{  
    $userAuth = auth()->user();  
    $user = User::find($userAuth['id']);  
    $validatedData = $request->validate([  
        'country' => 'nullable|string',  
        'city' => 'nullable|string',  
        'street_name' => 'nullable|string',  
        'zip' => 'nullable|string',  
        'phone_number' => 'nullable|string',  
    ]);  
  
    $user->update($validatedData);  
  
    return response()->json([  
        'message' => 'User successfully updated'  
    ], status: 200);  
}
```

Order history

You have not made any orders yet!

Тука се прикажуваат сите нарачки што корисникот ги направил.

Со клик на копчето Checkout во кошничката:



ни се отвора формата за правење на нарачка.

A screenshot of the TechShop checkout page. The page has a light gray background. At the top left is the TechShop logo. At the top center is the word "CATEGORIES" in purple. At the top right is a shopping cart icon with a red notification bubble and the name "VLADIMIR STOJCHESKI". The main content area is divided into two columns. The left column contains a form with sections: "Personal info" (First name: Vladimir, Last Name: Stojcheski), "Delivery address" (Country: Macedonia, City: Prilep, Street name: Ulica Ulica 12, Zip: 7500), "Contact" (Email: stojcheskivladimir@gmail.com, Phone Number: 3897100000), and "Payment" (Card number, CCV, and an ORDER button). The right column displays a list of items: "Delux DLM-136BU (W&G) Wired Optical Mouse" for 5\$, "Logitech Webcam StreamCam White" for 150\$, and "Razer Abyssus Essential Ambidextrous 7" for 43\$. At the bottom right of this list is "Total: 198\$".

Доколку се ажурирани податоците во профилот, тука автоматски се популираат во формата. Од десната страна се листаат продуктите за кои се прави нарачката. Со клик на копчето Order, се прави нова нарачка.

Повик кон бекенд:

{host}/order/create (POST)

Headers: {

“Authorization” : “Bearer {token}”

}

Body: {

"first_name": "Vladimir",

"last_name": "Stojcheski",

"email": "stojcheskivladimir@gmail.com",

"phone": "38971111111",

"country": "Macedonia",

"city": "Prilep",

"streetName": "Ulica ulica 12",

"zip_code": "7500",

"total_amount": "198",

"products": [

"16",

"19",

"20"

]

}



Order created successfully



Бекенд код:

```

public function create(Request $request)
{
    $user = auth()->user();
    if ($user == null) {
        response()->json(['error' => 'Unauthorized'], status: 401);
    }
    $frontendHost = config(key: 'app.frontend_host');
    $uniqueOrderNumber = Order::generateUniqueOrderNumber();
    // Validate the request data
    $validatedData = $request->validate([
        'first_name' => 'required|string',
        'last_name' => 'required|string',
        'email' => 'required|email',
        'phone' => 'required|string',
        'country' => 'required|string',
        'city' => 'required|string',
        'streetName' => 'required|string',
        'zip_code' => 'required|string',
        'total_amount' => 'required|string',
        'products' => 'required|array',
    ]);

    // Create a new order including the delivery address
    $order = Order::create([
        'first_name' => $validatedData['first_name'],
        'last_name' => $validatedData['last_name'],
        'email' => $validatedData['email'],
        'phone_number' => $validatedData['phone'],
        'country' => $validatedData['country'],
        'city' => $validatedData['city'],
        'street_name' => $validatedData['streetName'],
        'zip_code' => $validatedData['zip_code'],
        'total_amount' => $validatedData['total_amount'],
        'order_number' => $uniqueOrderNumber,
        'user_id' => $user['id']
    ]);

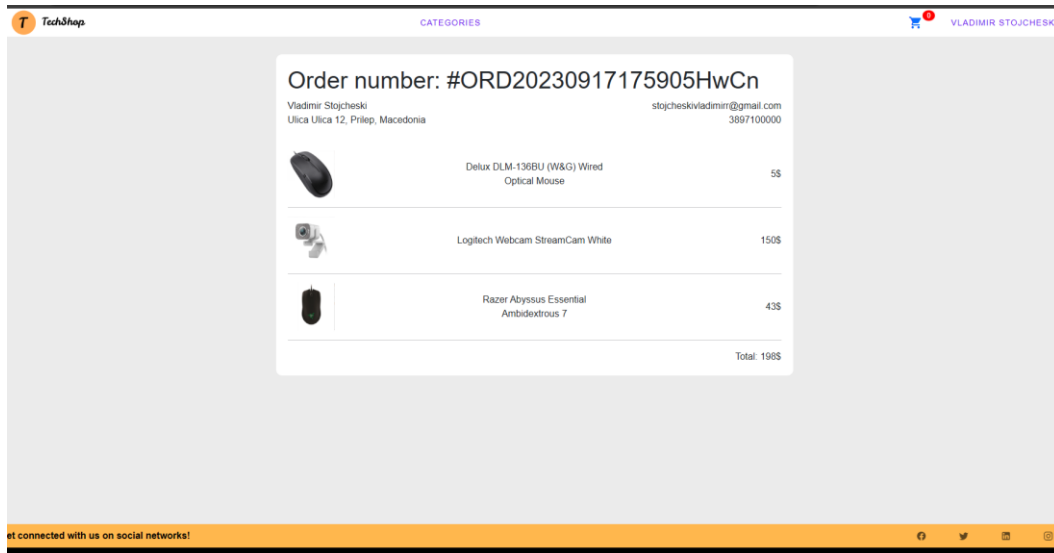
    $quantity = 5;
    $unitPrice = 10.99;
    // Attach products to the order (assuming you have an array of product IDs)
    $order->products()->attach($validatedData['products'], ['quantity' => $quantity,
        'unit_price' => $unitPrice,]);

    Mail::to($order->email)->send(new OrderDetails($order, $frontendHost));

    return response()->json(['message' => 'Order created successfully', 'orderId' => $order->id], status: 201);
}

```

Потоа не реди ректира кон преглед на нарачка и статусу мејл за нарачката:



TechShop <soanotificaiton@gmail.com>
to me ▾

Order Details

Thank you for your order. Your order details are as follows:

- Order Number: ORD20230917001431higv
- Total price: 1700\$
- Number of products: 2



You can view your order by clicking on the following link:

[View Order Details](#)

Повик кон бекенд:

{host}/api/order/{id} (GET)

Headers: {

“Authorization” : “Bearer {token}”

}

Одговор: {


```
"order": {
  "id": 5,
  "order_number": "ORD20230917175905HwCn",
  "total_amount": 198,
  "first_name": "Vladimir",
  "last_name": "Stojcheski",
  "country": "Macedonia",
  "city": "Prilep",
  "street_name": "Ulica Ulica 12",
  "zip_code": "7500",
  "email": "stojcheskivladimir@gmail.com",
  "phone_number": "3897100000",
  "user_id": 6,
  "created_at": "2023-09-17T17:59:05.000000Z",
  "updated_at": "2023-09-17T17:59:05.000000Z",
  "products": [
    {
      "id": 16,
      "created_at": "2023-09-17T02:22:15.000000Z",
      "updated_at": "2023-09-17T02:22:15.000000Z",
      "title": "Delux DLM-136BU (W&G) Wired Optical Mouse",
      "img":
"http://localhost:8000/storage/products/Wluy4caMo9ul00KR27CQWW15Kt1I46YPHzuqYpVp.png",
      "description": "<p><strong><span style='font-family: Roboto;'><span style='font-size: 14px;'>Delux DLM-136BU (W&G) Wired Optical Mouse</span></span></strong><br /><strong><span style='font-family: Roboto;'><span style='font-size: 14px;'>&nbsp;&nbsp;&nbsp;White&Green</span></span></strong><br /><strong><span style='font-family: Roboto;'><span style='font-size: 14px;'>&nbsp;&nbsp;&nbsp;USB</span></span></strong><br /><strong><span style='font-family: Roboto;'><span style='font-size: 14px;'>&nbsp;&nbsp;&nbsp;Fashional outlook design</span></span></strong><br /><strong><span style='font-family: Roboto;'><span style='font-size: 14px;'>&nbsp;&nbsp;&nbsp;Eudipleural ergonomic design</span></span></strong><br /><strong><span style='font-family: Roboto;'><span style='font-size: 14px;'>&nbsp;&nbsp;&nbsp;Resolution:
```

1.000 DPI
 Keystroke Lifetime: 3
000
000 cycles
 USB cable 160cm</p>",</div>

"price": 5,</p></div>

"category_id": 5,</p></div>

"manufacturer_id": 5,</p></div>

"pivot": {</p></div>

"order_id": 5,</p></div>

"product_id": 16,</p></div>

"quantity": 5,</p></div>

"unit_price": 10.99</p></div>

}</p></div>

{</p></div>

"id": 19,</p></div>

"created_at": "2023-09-17T02:28:10.000000Z",</p></div>

"updated_at": "2023-09-17T02:28:10.000000Z",</p></div>

"title": "Logitech Webcam StreamCam White",</p></div>

"img":</p></div>

"http://localhost:8000/storage/products/Yien8vpN8dKCOPwpDoW0LjxrOOKYHqr0eb623dGc.png",</p></div>

"description": "<p>Logitech Webcam StreamCam White</p>\r\n<p> PN: 960-001299</p>",</p></div>

"price": 150,</p></div>

"category_id": 5,</p></div>

"manufacturer_id": 8,</p></div>

"pivot": {</p></div>

"order_id": 5,</p></div>

"product_id": 19,</p></div>

```

        "quantity": 5,
        "unit_price": 10.99
    }
},
{
    "id": 20,
    "created_at": "2023-09-17T02:29:49.000000Z",
    "updated_at": "2023-09-17T02:29:49.000000Z",
    "title": "Razer Abyssus Essential Ambidextrous 7",
    "img":
"http://localhost:8000/storage/products/7uLE69qgDsntOf3JXSImm7FlfyvF8DkvGpTf5bL.png",
    "description": "<p><strong>Razer Abyssus Essential Ambidextrous
7</strong></p>\r\n<p><strong>200 DPI Optical Gaming Mouse
Black</strong></p>\r\n<p><strong>&nbsp;True 7</strong></p>\r\n<p><strong>200 DPI optical
sensor</strong></p>\r\n<p><strong>&nbsp;Underglow lighting powered by Razer
Chroma</strong></p>\r\n<p><strong>&nbsp;Inter-Device
Synchronization</strong></p>\r\n<p><strong>&nbsp;3 Hyperresponse
buttons</strong></p>\r\n<p><strong>&nbsp;Up to 10 million keystroke
durability</strong></p>\r\n<p><strong>&nbsp;RZ01-02160300-R3M1</strong></p>",
    "price": 43,
    "category_id": 5,
    "manufacturer_id": 9,
    "pivot": {
        "order_id": 5,
        "product_id": 20,
        "quantity": 5,
        "unit_price": 10.99
    }
}
]
}
}

```

И се прикажува на профил:

Profile

First name	Last name	Email
Vladimir	Stojcheski	stojcheskivladimir@gmail.com
Country	City	Phone number
Macedonia	Prilep	3897100000
Street name	Zip	SAVE
Ulica Ulica 12	7500	

Order history

#ORD20230917175905HwCn	198\$	Sun Sep 17 2023 20:00:06 GMT+0200 (Central European Summer Time)	VIEW
------------------------	-------	---	------

Со клик на копчето View се прикажуваат деталите за нарачката

Доколку сте најавени како админ можете да додавате нов продукт. Во хедерот ќе се појави копче

CATEGORIES ADD PRODUCT

Add product

И се прикажува следната форма:

CATEGORIES ADD PRODUCT

Choose File No file chosen

UPLOAD

Product title

Price

Category

Subcategory

Manufacturer

Description

File Edit View Format

↶ ↷

Paragraph

B *I*

≡ ≡ ≡ ≡

≡ ≡

P POWERED BY TINY

ADD PRODUCT

Повик кон бекенд: (POST – curl)

```
curl --location --request POST '{host}/api/products/create' \
--header 'Content-Type: multipart/form-data' \
--header 'Authorization: Bearer {token}' \
--form 'title="Test"' \
--form 'description="test"' \
--form 'price="200"' \
--form 'category_id="2"' \
--form 'manufacturer_id="1"' \
```

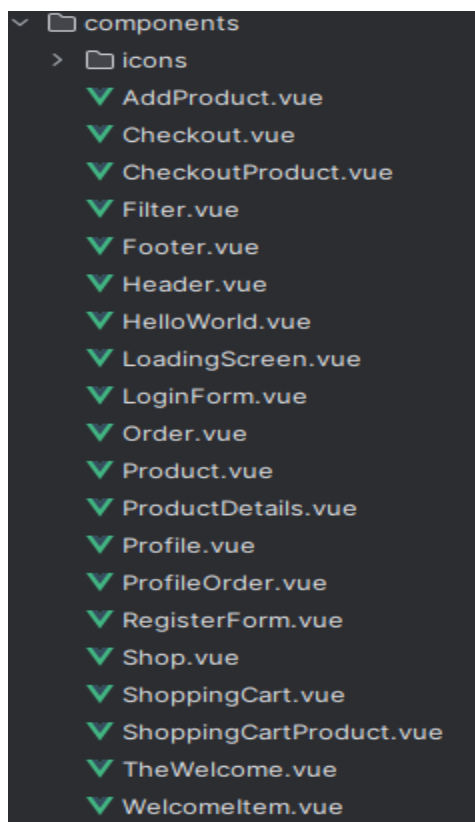
--form 'img=@"path_to_your_image.jpg"'

4. Фронтенд

Технологија

- Vue3
- Vuex
- Vuetify
- Axios
- Vue-router
- Vue-toastification

Компоненти:



Vuex store:

```
3 export default new Vuex.Store({ options: {
4   state: {
5     manufacturers: [],
6     category: null,
7     query: {},
8     cart: [],
9   },
10  mutations: {
11    setManufacturers(state, manufacturers) :void {
12      state.manufacturers = manufacturers;
13    },
14    setPrice(state, category) :void {
15      state.category = category;
16    },
17    setQuery(state, query) :void {
18      state.query = query
19    },
20    addToCart(state, product) :void {
21      state.cart.push(product);
22    },
23    removeFromCart(state, productId) :void {
24      state.cart = state.cart.filter(item => item.id !== productId);
25    },
26    emptyCart(state) :void {
27      state.cart = []
28    }
29  },
30  // You can add actions to update filters and commit mutations here
31  actions: {
32    updateQuery({ commit }, query) :void {
33      // You can perform asynchronous operations here if needed
34      // For now, directly commit the mutation
35      commit('setQuery', query);
36    },
37    addProductToCart({ commit, state }, product) :void {
38      if (!state.cart.find(item => item.id === product.id)) {
39        commit('addToCart', product);
40      }
41    },
42    removeProductFromCart({ commit }, productId) :void {
43      commit('removeFromCart', productId);
44    },
45    emptyProductsFromCart({commit}) :void {
46      commit('emptyCart')
47    }
48  },
49  getters: {
50    cartItems: state => state.cart,
51  },
52 })
```

Main.js:

```
const toastOptions :{position: string} = {
  position: 'bottom-right'
}

const routes :[{path: string, component: co...} = [
  { path: '/', component: Shop, name: "Home" },
  { path: '/product/:id', component: () :Promise<...> => import('@/components/ProductDetails.vue'), name: "ProductDetails"},
  { path: '/product/add', component: AddProduct, name: "AddProduct"},
  { path: '/shopping-cart', component: ShoppingCart, name: "ShoppingCart"},
  { path: '/checkout', component: Checkout, name: "Checkout"},
  { path: '/order/:id', component: Order, name: "Order"},
  { path: '/login', component: LoginForm, name: "LoginForm"},
  { path: '/register', component: RegisterForm, name: "RegisterForm"},
  { path: '/profile', component: Profile, name: "Profile"},
]

const router :Router = createRouter( options: {
  // 4. Provide the history implementation to use. We are using the hash history for simplicity here.
  history: createWebHistory(),
  routes// short for 'routes: routes'
})

const vuetify : = createVuetify( vuetify: {
  components,
  directives,
  ssr: true,
})

const emitter :Emitter<Record<...>> = mitt()

const app :App<Element> = createApp(App)
app.use(router).use(store).use(vuetify).use(Toast, toastOptions)
app.config.globalProperties.emitter = emitter
app.mount( rootContainer: '#app')
```

5. Бекенд

Технологија:

- Php 8.0.2
- Laravel/framework 9.19
- Tymon/jwt-auth 2.0
- Laravel/tinker


```

Route::get( uri: "/products", [ProductController::class, "getAll"]);
Route::get( uri: "/products/{id}/all", [ProductController::class, "getAllByCategory"]);
Route::get( uri: "/products/{id}", [ProductController::class, "getById"]);
Route::get( uri: "/order/{id}", [OrderController::class, "getOrderWithProducts"]);

Route::get( uri: "/categories/sub/all", [SubCategoryController::class, "getAll"]);
Route::get( uri: "/categories/all", [CategoryController::class, "getAll"]);
Route::get( uri: "/manufacturers/{id}", [ManufacturerController::class, "getAllByCategory"]);
Route::get( uri: "/manufacturers", [ManufacturerController::class, "getAll"]);

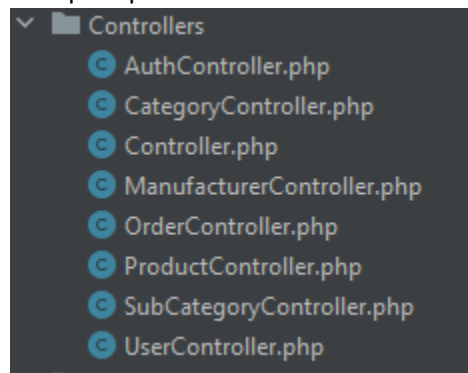
Route::post( uri: "/products/all", [ProductController::class, "getAllFiltered"]);
Route::post( uri: "/manufacturers/all", [ManufacturerController::class, "getAllByProducts"]);
Route::post( uri: "/products/create", [ProductController::class, "createProduct"]);
Route::post( uri: "/order/create", [OrderController::class, "create"]);

Route::middleware(['api'])->group(function() {
    Route::post( uri: '/login', [AuthController::class, 'login']);
    Route::get( uri: '/logout', [AuthController::class, 'logout']);
    Route::post( uri: '/register', [AuthController::class, 'register']);
    Route::get( uri: '/getaccount', [AuthController::class, 'getaccount']);
    Route::get( uri: '/profile', [UserController::class, 'getUserProfile']);
    Route::put( uri: '/profile/edit', [UserController::class, 'update']);
});

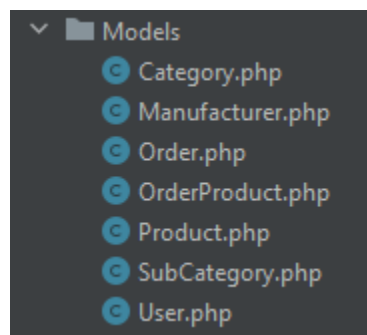
```

Апи:

Контролери:



Модел:



6. База на податоци

