7

YETTINCHI AMALIY MASHG‘ULOT

Korxona ilovalarini dasturlash · 2 soat · 7-mavzu

REST API: JWT va OpenAPI hujjatlari

Jamoa loyihasi bosqichi · 5 ta PHP topshiriq · o‘qituvchi bahosi

REST | HTTP | JWT | OpenAPI | API Resource

MASHG‘ULOT MAQSADI

Amaliy mashg‘ulot oxirida siz quyidagilarni bajara olasiz...

1. **Yaratish** — mahsulotlar va buyurtmalar uchun `/api/v1` REST endpointlarini (API Resource, Form Request)
2. **Sozlash** — `php-open-source-saver/jwt-auth` orqali JWT autentifikatsiya: login, refresh, logout
3. **Qo‘llash** — sahifalash, filtrlash va RFC 7807 formatidagi yagona xato javoblarini
4. **Hujjatlash** — API ni Scribe yordamida OpenAPI spetsifikatsiyasi va `/docs` sahifasi bilan
5. **Tekshirish** — endpointlarni Postman kolleksiyasi va `newman` orqali

Qanday bajariladi va qanday baholanadi?

Ish tartibi

1. Saytga login va parolingiz bilan kiring: **emro.uz/korxona/kirish.html**
2. **7-amaliy** sahifasini oching (oxirgi slayddagi QR kod).
3. Yo‘riqnoma bo‘yicha jamoa loyihasida bosqichni bajaring, **commit** va **push** qiling.
4. PHP topshiriqlarni yeching: ▶ **Sinab ko‘rish** → ✓ **Topshirish** (yashirin testlar).
5. Loyiha havolasini (GitHub repo, PR yoki tag) topshiring — o‘qituvchi **baho** qo‘yadi.

Baholash mezonlari

- **Natija** — loyiha bosqichi mezonlar ro‘yxati bo‘yicha bajarilgan.
- **Topshiriqlar** — PHP topshiriqlar testlardan o‘tgan.
- **Kod sifati** — tushunarli nomlar, PSR-12, kichik va mazmunli commitlar.
- **Jamoaviy hissa** — har bir a’zo o‘z qismini tushuntira oladi.
- **Muddat** — muddatdan keyin topshirilgan ish “kechikkan” deb belgilanadi.

01

1-qadam. API skeleti: marshrut, Resource, validatsiya

install:api · Api/V1 controllerlar · ProductResource · OrderResource · 201/422/409

install:api

make:resource

apiResource

TERMINAL

Skelet buyruqlari

```
php artisan install:api
php artisan make:controller Api/V1/ProductController --api
php artisan make:controller Api/V1/OrderController --api
php artisan make:resource ProductResource
php artisan make:resource OrderResource
php artisan make:resource OrderItemResource
php artisan route:list --path=api/v1
```

▶ NATIJA

```
INFO  API scaffolding installed. Please add the
[Laravel\Sanctum\HasApiTokens] trait to your User model.
INFO  Controller
[app/Http/Controllers/Api/V1/ProductController.php] created
successfully.
```

`install:api`

`routes/api.php` + Sanctum; `/api` prefiksi avtomatik

`--api`

5 ta API metodi; `--model=Product` bilan `Product $product` binding

`make:resource`

`app/Http/Resources` ichida `JsonResource` sinfi

ORDERRESOURCE

Buyurtma ko‘rinishi

```php
class OrderResource extends JsonResource
{
    public function toArray(Request $request): array
    {
        return [
            'id' => $this->id,
            'number' => $this->number,          // BY-2026-000123
            'status' => $this->status->value,   // OrderStatus enum
            'counterparty' => new CounterpartyResource(
                $this->whenLoaded('counterparty')),
            'total' => $this->total,            // "1155000.00"
            'currency' => $this->currency,
            'items' => OrderItemResource::collection(
                $this->whenLoaded('items')),
            'created_at' => $this->created_at->toIso8601String(),
        ];
    }
}
```

▶ NATIJA

```
{"data": {"id": 42, "number": "BY-2026-000042", "status": "draft",
  "total": "1155000.00", "currency": "UZS",
  "items": [{"product_id": 15, "quantity": 3, "price":
"385000.00"}]}}
```

`$this->status->value`

enum o‘rniga uning satr qiymati: `draft`, `confirmed`…

`whenLoaded('items')`

ro‘yxatda qatorlarsiz, `show` da qatorlar bilan

`OrderItemResource::collection`

ichki to‘plam ham Resource orqali

CONTROLLER

Servisni chaqiruvchi store va confirm

```php
public function store(StoreOrderRequest $request, OrderService $orders)
{
    $data = OrderData::fromArray($request->validated());
    $order = $orders->create($data);

    return (new OrderResource($order->load('items')))
        ->response()
        ->header('Location', route('v1.orders.show', $order));
}

public function confirm(Order $order, OrderService $orders)
{
    $this->authorize('confirm', $order);         // OrderPolicy
    $order = $orders->confirm($order);           // 409 bo‘lishi mumkin

    return new OrderResource($order->load('items'));
}
```

▶ NATIJA

```
POST /api/v1/orders            → 201 Created
POST /api/v1/orders/42/confirm → 200 {"data": {"status":
"confirmed"}}
POST /api/v1/orders/42/confirm → 409 problem+json (holat o‘tishi
taqiqlangan)
```

`OrderData::fromArray`

2-mavzudagi DTO — servis massiv emas, aniq tur oladi

`$this->authorize()`

5-mavzudagi `OrderPolicy`; ruxsat yo‘q — 403

`$orders->confirm()`

tranzaksiya, invariantlar va holat o‘tishi — servisda

02

2-qadam. JWT autentifikatsiya

jwt-auth o‘rnatish · api guard · JWTSubject · login, refresh, logout

jwt-auth | auth:api | JWTSubject

TERMINAL

jwt-auth ni o‘rnatish

```
composer require php-open-source-saver/jwt-auth
php artisan vendor:publish \
--provider="PHPOpenSourceSaver\JWTAuth\Providers\LaravelServiceProvider"
php artisan jwt:secret
```

▶ NATIJA

```
jwt-auth secret [...] set successfully.

# .env
JWT_SECRET=<tasodifiy 64 belgi — repozitoriyga tushmaydi>
```

`vendor:publish`

`config/jwt.php` — ttl, refresh_ttl, algo, blacklist

`jwt:secret`

HS256 uchun sir kalit `.env` ga yoziladi

`.env.example`

faqat `JWT_SECRET=` (qiymatsiz) qatori

USER MODELI

JWTSubject va api guard

```
// config/auth.php
'guards' => [
    'web' => ['driver' => 'session', 'provider' => 'users'],
    'api' => ['driver' => 'jwt', 'provider' => 'users'],
],

// app/Models/User.php
class User extends Authenticatable implements JWTSubject
{
    public function getJWTIdentifier()
    {
        return $this->getKey();
    }
    public function getJWTCustomClaims(): array
    {
        return ['roles' => $this->getRoleNames()];
    }
}
```

▶ NATIJA

```
Token payload: {"iss": ".../api/v1/auth/login", "iat": ..., "exp":
...,
 "sub": "7", "jti": "...", "roles": ["menejer"]}
```

`'driver' => 'jwt'`

`auth:api` middleware shu guard bilan ishlaydi

`getJWTIdentifier()`

`sub` da’vosi — foydalanuvchi ID si

`getJWTCustomClaims()`

qo‘shimcha da’vo: spatie rollari (maxfiy emas)

TEKSHIRUV

Autentifikatsiya ssenariysi

1

Tokensiz

`GET /api/v1/products` → 401 problem+json

2

Login

`POST /api/v1/auth/login` → 200, `access_token`, `expires_in`

3

Token bilan

`Authorization: Bearer ...` → 200 va mahsulotlar ro‘yxati

4

Rol tekshiruvi

omborchi tokeni bilan `POST /orders/42/confirm` → 403

5

Logout

`POST /api/v1/auth/logout` → 204; eski token bilan → 401

03

3-qadam. OpenAPI hujjati va Postman kolleksiyasi

Scribe · docblock teglari · /docs · collection.json · newman

Scribe

Postman

newman

POSTMAN TESTS

Login javobidan tokenni olish

```
// Login so‘rovi → Tests (Post-response) bo‘limi
pm.test("Login muvaffaqiyatli", function () {
    pm.response.to.have.status(200);
});
const body = pm.response.json();
pm.environment.set("token", body.access_token);

// Kolleksiya → Authorization: Bearer Token = {{token}}
// Mahsulot yaratish so‘rovi → Tests
pm.test("201 va Location", function () {
    pm.response.to.have.status(201);
    const loc = pm.response.headers.get("Location");
    pm.expect(loc).to.include("/products/");
});
```

▶ NATIJA

```
POST Login            ✓ Login muvaffaqiyatli
GET  Mahsulotlar      ✓ 200
POST Mahsulot yaratish ✓ 201 va Location
```

pm.environment.set

token muhit o‘zgaruvchisiga yoziladi

{{token}}

kolleksiyadagi barcha so‘rovlar shu tokenni ishlatadi

pm.test

har bir so‘rov uchun kutilgan status va sarlavhalar

TERMINAL

Hujjat yaratish va kolleksiyani yugurtirish

```
composer require --dev knuckleswtf/scribe
php artisan vendor:publish --tag=scribe-config
php artisan scribe:generate

npx newman run docs/postman/omborpro.postman_collection.json \
  -e docs/postman/local.postman_environment.json
```

▶ NATIJA

```
┌──────────────────────────┬──────────┬──────────┐
│                          │ executed │   failed │
├──────────────────────────┼──────────┼──────────┤
│                 requests │       12 │        0 │
│               assertions │       18 │        0 │
└──────────────────────────┴──────────┴──────────┘
```

scribe:generate

`/docs`, `/docs.openapi`, `/docs.postman` yangilanadi

newman run

kolleksiyani buyruq qatorida yugurtiradi

-e ...environment.json

muhit: `baseUrl`, `token` (repozitoriyda tokensiz)

Topshirishdagi endpointlar ro‘yxati

Endpoint	Metod	Kutilgan status
`/api/v1/auth/login`	POST	200 / 401 / 422 / 429
`/api/v1/auth/refresh`, `/logout`	POST	200 / 204
`/api/v1/products`	GET, POST	200 (sahifalangan) / 201 / 422
`/api/v1/products/{product}`	GET, PUT, DELETE	200 / 204 / 404
`/api/v1/orders`	GET, POST	200 / 201 / 422
`/api/v1/orders/{order}/confirm`	POST	200 / 403 / 409

Har bir qator Postman da kamida bitta muvaffaqiyatli va bitta xato so‘rov bilan qoplansin.

LOYIHA BOSQICHI

Natija va topshirish

Natija: mahsulotlar va buyurtmalar uchun JWT bilan himoyalangan `/api/v1` REST API, yagona problem+json xato formati, Scribe yaratgan OpenAPI hujjati (`/docs`) va `newman` bilan muvaffaqiyatli o‘tadigan Postman kolleksiyasi.

Baholash mezonlari

✓ `routes/api.php` da `v1` guruhi, `Route::apiResource` (products, orders) va `orders/{order}/confirm` amal-endpointi bor.

✓ Barcha javoblar API Resource orqali; parol, `deleted_at` kabi maydonlar chiqmaydi, pul satr ko‘rinishida.

✓ Ro‘yxatlar sahifalangan (`data`, `links`, `meta`), `per_page` 1–100 bilan cheklangan, filtr/saralash oq ro‘yxat bilan.

✓ Yaratishda 201 + `Location`, validatsiyada 422, biznes-xatoda 409 problem+json, ruxsat yo‘qligida 403.

✓ JWT: login, refresh, logout ishlaydi; `JWT_SECRET` repozitoriyda yo‘q; login `throttle` bilan himoyalangan.

✓ `/docs` sahifasi va OpenAPI fayli Scribe bilan yaratilgan, endpointlar guruhlangan va izohlangan.

✓ Postman kolleksiyasi va muhit fayli repozitoriyda; `newman run` 0 ta xato bilan o‘tadi.

📤 GitHub repozitoriyingizdagi `feature/api-v1` Pull Request havolasini yuboring. PR tavsifida `route:list --path=api/v1` chiqishi, `/docs` sahifasi skrinshoti va `newman run` natijasi bo‘lsin.

Telefon kamerasi bilan skanerlang

Loyiha havolasini amaliy sahifasidagi «Loyiha bosqichi» bo‘limida topshiring.

AMALIY TOPSHIRIQ

Kompyuterda mustaqil · Oson · 10 ball

4-topshiriq: Authorization sarlavhasidan token

Middleware `Authorization` sarlavhasidan Bearer tokenni ajratib oladi. Sarlavha `null` bo‘lishi mumkin. Chetdagi bo‘sh joylar e’tiborga olinmaydi. To‘g‘ri format: `Bearer` so‘zi (katta-kichik harf farqi yo‘q), kamida bitta bo‘sh joy, so‘ng bo‘sh joysiz token. Boshqa sxema (`Basic ...`), tokensiz `Bearer` yoki ichida bo‘sh joy bor token — `null`.

Imzo: `function bearer_token(?string $header): ?string`

Qaytaradi: Token satri yoki `null`.

MISOL · CHAQIRUV	MISOL · NATIJA
`bearer_token('Bearer abc.def.ghi')`	`"abc.def.ghi"`

Telefon kamerasi bilan skanerlang

1. Saytga kiring (login va parol)
2. PHP kodni yozing va sinab ko‘ring
3. ✓ Topshirish — natija o‘qituvchiga

AMALIY TOPSHIRIQ

Kompyuterda mustaqil · O‘rta · 20 ball

5-topshiriq: HS256 JWT yasash

JWT yasang: header `['alg' => 'HS256', 'typ' => 'JWT']`, payload — berilgan massiv. Har ikkalasi `json_encode` (qo‘shimcha bayroqsiz) qilinadi va base64url kodlanadi (`+` → `-`, `/` → `_`, oxiridagi `=` olib tashlanadi). Imzo — `header.payload` satrining `hash_hmac('sha256', ..., $secret, true)` xom natijasi, u ham base64url. Natija: `header.payload.signature`. Sir kalit 8 belgidan qisqa bo‘lsa — `InvalidArgumentException`.

Imzo: `function jwt_yasash(array $payload, string $secret): string`

Qaytaradi: Uch qismli token satri.

MISOL · CHAQIRUV

```
jwt_yasash(['sub' => 7, 'role' => 'menejer', 'exp' =>
1800000000], 'test-sir-2026')
```

MISOL · NATIJA

```
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOjcsInJvbGUiOiJtZW
5lamVyIiwiZXhwIjoxODAwMDAwMDAwfQ.FiXXdFWcvGc7-qxOPNCSGVBnK3GzC-
akzbrSKob3QlA"
```

Telefon kamerasi bilan skanerlang

1. Saytga kiring (login va parol)
2. PHP kodni yozing va sinab ko‘ring
3. ✓ Topshirish — natija o‘qituvchiga

AMALIY TOPSHIRIQ

Kompyuterda mustaqil · O‘rta · 20 ball

6-topshiriq: Ro‘yxat so‘rovi parametrlari

`GET /api/v1/products` query parametrlarini (satrlar massivi) tekshirib, tayyor so‘rov rejasini qaytaring. `page` — standart 1, faqat raqamlardan iborat va ≥ 1; `per_page` — standart 20, 1–100. `sort` — vergul bilan ajratilgan maydonlar, standart `name`; ruxsat etilganlar: `name`, `sku`, `price`, `created_at`; `-` prefiksi kamayish (`desc`), aks holda `asc`. `filter` — ixtiyoriy massiv: `category_id` (raqamlar → int) va `is_active` (`'1'`/`'true'` → true, `'0'`/`'false'` → false). Har qanday noto‘g‘ri qiymat yoki noma’lum maydon/filtr — `InvalidArgumentException`.

Imzo: `function api_sorov(array $q): array`

Qaytaradi: `['page' => int, 'per_page' => int, 'sort' => [[maydon, 'asc'|'desc'], ...], 'filter' => [kalit => qiymat]]` — `filter` kalitlari kirishdagi tartibda.

MISOL · CHAQIRUV

```
api_sorov([])
```

MISOL · NATIJA

```
{"page":1,"per_page":20,"sort":[["name","asc"]],"filter":[]}
```

Telefon kamerasi bilan skanerlang

1. Saytga kiring (login va parol)
2. PHP kodni yozing va sinab ko‘ring
3. ✓ Topshirish — natija o‘qituvchiga

AMALIY TOPSHIRIQ

Kompyuterda mustaqil · Qiyin · 30 ball

7-topshiriq: JWT ni tekshirish

Kelgan tokenni HS256 bo‘yicha tekshiring va payloadni qaytaring. Tartib muhim: (1) token aynan 3 ta bo‘sh bo‘lmagan qismdan iborat bo‘lmasa — `InvalidArgumentException`; (2) header dagi `alg HS256` bo‘lmasa (masalan, `none`) yoki imzo mos kelmasa — `UnexpectedValueException` (imzoni `hash_equals` bilan solishtiring); (3) payload JSON obyekt bo‘lmasa yoki unda butun sonli `exp` bo‘lmasa — `InvalidArgumentException`; (4) `exp <= $now` yoki `nbf` bor va `nbf > $now` bo‘lsa — `DomainException`.

Imzo: `function jwt_tekshir(string $token, string $secret, int $now): array`

Qaytaradi: Payload — assotsiativ massiv (`json_decode(..., true)`).



Telefon kamerasi bilan skanerlang

1. Saytga kiring (login va parol)
2. PHP kodni yozing va sinab ko‘ring
3. ✓ Topshirish — natija o‘qituvchiga

MISOL · CHAQIRUV

```
jwt_tekshir('eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOjcsInJvbGUiOiJtZW5lamVyIiwiZXhwIjoxODAwMDAwMDAwfQ.FiXXdFWcvGc7-qxOPNCSGVBnK3GzC-akzbrSKob3QlA', 'test-sir-2026', 1790000000)
```

MISOL · NATIJA

```
{"sub":7,"role":"menejer","exp":1800000000}
```

AMALIY TOPSHIRIQ

Kompyuterda mustaqil · Qiyin · 30 ball

8-topshiriq: So‘rovlar limiti (rate limiter)

Laravel `RateLimiter` ga o‘xshash hisoblagich yozing. Konstruktor: `$max` — oynadagi ruxsat etilgan so‘rovlar, `$window` — oyna uzunligi (sekund); ikkalasi ham ≥ 1, aks holda `InvalidArgumentException`. Har bir kalit uchun oyna **birinchi so‘rov vaqtidan** boshlanadi va `start + window` gacha davom etadi; `$now >= start + window` bo‘lsa, yangi oyna ochiladi. `hit()` so‘rovni hisobga oladi: limit tugamagan bo‘lsa — ruxsat, aks holda rad etadi (rad etilgan so‘rov hisoblagichni oshirmaydi). Kalitlar bir-biridan mustaqil.

Imzo: `final class RateLimiter`

Qaytaradi: `hit(): ['allowed' => bool, 'remaining' => int, 'retry_after' => int]`. Ruxsatda `retry_after` = 0; rad etilganda `remaining` = 0, `retry_after = start + window - now`.

MISOL · CHAQIRUV

```
(function () { $r = new RateLimiter(3, 60); $o = []; foreach ([0, 10, 20, 30] as $t) { $o[] = $r->hit('u7', $t); } return $o; })()
```

MISOL · NATIJA

```
[{"allowed":true,"remaining":2,"retry_after":0},{"allowed":true,"remaining":1,"retry_after":0},{"allowed":true,"remaining":0,"retry_after":0},{"allowed":false,"remaining":0,"retry_after":30}]
```

Telefon kamerasi bilan skanerlang

1. Saytga kiring (login va parol)
2. PHP kodni yozing va sinab ko‘ring
3. ✓ Topshirish — natija o‘qituvchiga

TOPSHIRIQLAR VA TOPSHIRISH

7-amaliy: 8 ta PHP topshiriq

1 **HTTP metodining xossalari**
Oson · 10 ball · ma’ruzada

2 **Sahifalash meta ma’lumoti**
O‘rta · 20 ball · ma’ruzada

3 **Problem Details javobi**
O‘rta · 20 ball · ma’ruzada

4 **Authorization sarlavhasidan token**
Oson · 10 ball

5 **HS256 JWT yasash**
O‘rta · 20 ball

6 **Ro‘yxat so‘rovi parametrlari**
O‘rta · 20 ball

7 **JWT ni tekshirish**
Qiyin · 30 ball

8 **So‘rovlar limiti (rate limiter)**
Qiyin · 30 ball

Telefon kamerasi bilan skanerlang

PHP kod saytda yoziladi va server tomonidan avtomatik tekshiriladi.

Loyiha havolasini ham shu sahifada topshirasiz; o‘qituvchi baho qo‘yadi.