7

YETTINCHI MA’RUZA

Korxona ilovalarini dasturlash · 7-mavzu

REST API loyihalash

Resurslar va HTTP · API Resource · versiya va sahifalash · xatolar formati · JWT · OpenAPI

REST | HTTP | JWT | OpenAPI | API Resource

ISINISH

Omborchining mobil ilovasi, 1C buxgalteriyasi va Telegram-bot OmborPro dan bir xil ma’lumotni qanday oladi?

Hozircha javob bermang. Brauzer uchun yozilgan Blade sahifalari bu mijozlarga yetarlimi — o‘ylab ko‘ring.

1

Mobil ilova HTML sahifani “o‘qib” undan mahsulot narxini ajratib olishi to‘g‘rimi?

2

1C tizimi “buyurtma yaratildimi yoki yo‘qmi” ekanini qanday biladi — matn bo‘yichami yoki boshqa belgi bormi?

3

Mobil ilova foydalanuvchisi kim ekanini server sessiya cookie siz qanday aniqlaydi?

O‘QUV MAQSADLARI

Bu ma’ruza oxirida siz quyidagilarni bajara olasiz…

1. **Tushuntirish** — REST tamoyillari, resurs va uning ko‘rinishi (representation) tushunchalarini
2. **Farqlash** — HTTP metodlarining xavfsiz va idempotent xossalarini hamda 2xx/4xx/5xx status kodlarini
3. **Loyihalash** — OmborPro uchun `/api/v1` resurslar daraxti, sahifalash va filtrlash parametrlarini
4. **Qo‘llash** — Laravel 12 da `Route::apiResource`, API Resource va Form Request ni
5. **Yozish** — RFC 7807 (Problem Details) formatidagi yagona xato javoblarini
6. **Tavsiflash** — JWT tuzilmasi, imzo va muddat (exp), Sanctum tokenlaridan farqini
7. **Yaratish** — Scribe yordamida OpenAPI hujjati va Postman kolleksiyasini
8. **Baholash** — API dizaynidagi xatolarni: noto‘g‘ri status kod, chegarasiz `per_page`, ma’lumot sizib chiqishi

01

REST resurslari va HTTP semantikasi

Resurs va URI · metodlar: xavfsiz va idempotent · status kodlar · sarlavhalar

REST | Resurs | Idempotentlik | Status kod

Resurs, URI va ko‘rinish

REST API — resurslar ustida standart HTTP metodlari bilan ishlash shartnomasi. URI — ot, metod — fe’l.

Resurs

Biznes obyekti yoki to‘plam: `products`, `orders/42`, `orders/42/items`.

Ko‘rinish

Resursning JSON tasviri — jadval ustunlari emas, mijoz uchun barqaror shartnoma.

Stateless

Har bir so‘rov o‘zi yetarli: token har safar `Authorization` sarlavhasida keladi.

Amal-endpoint

CRUD ga sig‘maydigan o‘tish: `POST /api/v1/orders/42/confirm`.

✗ `/api/getProducts`, `/api/order/delete/42` — URI da fe’l. ✓ `GET /api/v1/products`, `DELETE /api/v1/orders/42`.

Metod nimani anglatadi?

Metod	OmborPro misoli	Xavfsiz	Idempotent
GET	`GET /products?page=2` — ro‘yxat	ha	ha
POST	`POST /orders` — yangi buyurtma	yo‘q	yo‘q
PUT	`PUT /products/15` — to‘liq almashtirish	yo‘q	ha
PATCH	`PATCH /products/15` — narxni o‘zgartirish	yo‘q	kafolatlanmaydi
DELETE	`DELETE /products/15` — soft delete	yo‘q	ha
POST (amal)	`POST /orders/42/confirm` — holat o‘tishi	yo‘q	yo‘q

`POST` ni xavfsiz takrorlash uchun `Idempotency-Key` sarlavhasi: server shu kalit bilan birinchi javobni saqlab, takroriy so‘rovga qaytaradi.

STATUS KODLAR

Qaysi holatda qaysi kod?

Kod	Ma’nosi	OmborPro misoli
`200 OK`	muvaffaqiyat, tanada natija	mahsulotlar ro‘yxati
`201 Created`	yangi resurs + `Location`	buyurtma yaratildi
`204 No Content`	muvaffaqiyat, tana yo‘q	mahsulot o‘chirildi
`401 Unauthorized`	token yo‘q yoki muddati o‘tgan	`exp` o‘tib ketgan JWT
`403 Forbidden`	token bor, ruxsat yo‘q	omborchi hisob-fakturani so‘radi
`404 Not Found`	resurs topilmadi	`/products/9999`
`409 Conflict`	biznes-qoida to‘qnashuvi	rezerv uchun qoldiq yetarli emas
`422 Unprocessable`	validatsiya xatosi	`sku` band, `price` manfiy

Yana: `400` — buzuq JSON, `429` — limit oshdi (`Retry-After`), `500` — kutilmagan server xatosi.

▶ INTERAKTIV

Status kodni tanlang

Har bir vaziyat uchun eng to‘g‘ri HTTP javob kodini tanlang. (Interaktiv slaydlarda bosib bajariladi.)

1 `POST /api/v1/orders` **muvaffaqiyatli, yangi buyurtma yaratildi.**

A) 200
B) 201
C) 204
D) 202

2 **Rezerv qilish uchun** `UN-001` **dan 5 dona kerak, mavjud qoldiq 3.**

A) 400
B) 404
C) 500
D) 409

3 **Omborchi roli bilan** `GET /api/v1/invoices` **so‘raldi; token yaroqli.**

A) 403
B) 401
C) 404
D) 422

4 `price` **maydoni** `-500` **bo‘lib keldi.**

A) 409
B) 400
C) 422
D) 500

TUSHUNISHNI TEKSHIRISH

O‘ylang – Juftlikda – Bo‘lishing · 3 daqiqa

1-bo‘lim yakuni: REST resurslari va HTTP semantikasi

1

Nima uchun `/api/createOrder` yaxshi URI emas?

Savol 1

2

`POST` va `PUT` ning idempotentlik bo‘yicha farqi nimada?

Savol 2

3

401 va 403 status kodlari qachon qaytariladi?

Savol 3

MA’RUZADAGI MASHQ

Har kim o‘z qurilmasida · 10 daqiqa · Oson · 10 ball

1-topshiriq: HTTP metodining xossalari

API shlyuzi (gateway) takroriy yuborishga ruxsat berishdan oldin metod xossalarini bilishi kerak. Metod nomi berilgan (katta-kichik harf farqi yo‘q). `GET`, `HEAD`, `OPTIONS` — xavfsiz va idempotent; `PUT`, `DELETE` — xavfsiz emas, lekin idempotent; `POST`, `PATCH` — ikkalasi ham emas. Boshqa har qanday nom (bo‘sh satr ham) — `InvalidArgumentException`.

Imzo: `function metod_xossasi(string $method): array`

Qaytaradi: `['safe' => bool, 'idempotent' => bool]` — kalitlar aynan shu tartibda.

MISOL · CHAQIRUV	MISOL · NATIJA
`metod_xossasi('GET')`	`{"safe":true,"idempotent":true}`

Telefon kamerasi bilan skanerlang

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

02

Laravel’da API: marshrutlar, controller va API Resource

install:api · Route::apiResource · ingichka controller · JsonResource · 201 va Location

routes/api.php

apiResource

JsonResource

Form Request

ROUTES/API.PHP

Versiyalangan marshrutlar guruhi

```php
use App\Http\Controllers\Api\V1\AuthController;
use App\Http\Controllers\Api\V1\OrderController;
use App\Http\Controllers\Api\V1\ProductController;
use Illuminate\Support\Facades\Route;

Route::prefix('v1')->name('v1.')->group(function () {
    Route::post('auth/login', [AuthController::class, 'login'])
        ->middleware('throttle:5,1');

    Route::middleware(['auth:api', 'throttle:api'])->group(function () {
        Route::apiResource('products', ProductController::class);
        Route::apiResource('orders', OrderController::class)
            ->only(['index', 'store', 'show']);
        Route::post('orders/{order}/confirm',
            [OrderController::class, 'confirm']);
    });
});
```

▶ NATIJA

```
$ php artisan route:list --path=api/v1
POST      api/v1/auth/login ........ Api\V1\AuthController@login
GET|HEAD  api/v1/products ......... v1.products.index
POST      api/v1/products ......... v1.products.store
GET|HEAD  api/v1/products/{product} v1.products.show
POST      api/v1/orders/{order}/confirm
```

prefix('v1')

`/api` + `/v1` — URL da versiya; nomlar `v1.products.index`

throttle:5,1

login uchun daqiqasiga 5 urinish — parol tanlashdan himoya

auth:api

`api` guard (JWT) — tokensiz so‘rov 401 oladi

apiResource

index, store, show, update, destroy — `create/edit` yo‘q

orders/{order}/confirm

CRUD dan tashqari biznes-amal, `OrderService` ni chaqiradi

CONTROLLER

Ingichka API controller

```php
class ProductController extends Controller
{
    public function index(Request $request)
    {
        $perPage = min(max($request->integer('per_page', 20), 1), 100);
        $products = Product::with('category')->orderBy('name')
            ->paginate($perPage)->withQueryString();
        return ProductResource::collection($products);
    }

    public function store(StoreProductRequest $request)
    {
        $product = Product::create($request->validated());
        return (new ProductResource($product))
            ->response()   // model yangi — status 201
            ->header('Location', route('v1.products.show', $product));
    }
}
```

▶ NATIJA

```
GET  /api/v1/products?per_page=500  → 200, per_page 100 ga
cheklandi
POST /api/v1/products (sku band)    → 422 {"errors": {"sku":
[...]}}
POST /api/v1/products (to‘g‘ri)     → 201, Location:
.../products/16
```

`min(max(..., 1), 100)`

`per_page` chegarasi — 1 000 000 yozuvli javobdan himoya

`with('category')`

eager loading — Resource ichida N+1 bo‘lmaydi

`withQueryString()`

filtrlar `links.next` ichida saqlanadi

`$request->validated()`

faqat qoidalardan o‘tgan maydonlar

`->response()->header()`

201 avtomatik, `Location` ni biz qo‘shamiz

API RESOURCE

ProductResource — barqaror JSON shartnomasi

```php
class ProductResource extends JsonResource
{
    public function toArray(Request $request): array
    {
        return [
            'id' => $this->id,
            'sku' => $this->sku,
            'name' => $this->name,
            'unit' => $this->unit,
            'price' => $this->price,        // decimal:2 → "385000.00"
            'is_active' => $this->is_active,
            'category' => new CategoryResource(
                $this->whenLoaded('category')),
            'created_at' => $this->created_at?->toIso8601String(),
            'links' => ['self' => route('v1.products.show', $this->id)],
        ];
    }
}
```

▶ NATIJA

```
{"data": {"id": 15, "sku": "UN-001", "name": "Un oliy nav, 50 kg",
  "unit": "dona", "price": "385000.00", "is_active": true,
  "category": {"id": 3, "name": "Un mahsulotlari"},
  "created_at": "2026-10-20T09:15:00+05:00",
  "links": {"self": "https://omborpro.test/api/v1/products/15"}}}
```

`toArray()`

maydonlar qo‘lda tanlanadi — `deleted_at`, ichki ustunlar chiqmaydi

`'price' => $this->price`

`decimal:2` cast satr qaytaradi — float yo‘q

`whenLoaded('category')`

relation yuklanmagan bo‘lsa, kalit umuman chiqmaydi

`toIso8601String()`

sana vaqt zonasi bilan (Asia/Tashkent, +05:00)

`{"data": ...}`

Resource javobni standart holda `data` ichiga o‘raydi

Mijoz nima yuboradi va nima oladi

```
POST /api/v1/products HTTP/1.1
Host: omborpro.test
Accept: application/json
Content-Type: application/json
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

{"sku": "UN-002", "name": "Un 1-nav, 50 kg", "category_id": 3,
 "unit": "dona", "price": "355000", "min_stock": 20}
```

▶ NATIJA

```
HTTP/1.1 201 Created
Location: https://omborpro.test/api/v1/products/16
Content-Type: application/json

{"data": {"id": 16, "sku": "UN-002", "price": "355000.00", ...}}
```

`Accept: application/json`

xatolar ham JSON bo‘ladi (aks holda 302 redirect)

`Authorization: Bearer`

har bir so‘rovda token — sessiya yo‘q

`"price": "355000"`

pul satr sifatida — o‘nli aniqlik saqlanadi

`201 + Location`

mijoz yangi resurs manzilini darhol biladi

TUSHUNISHNI TEKSHIRISH

O‘ylang – Juftlikda – Bo‘lishing · 3 daqiqa

2-bo‘lim yakuni: Laravel’da API: marshrutlar, controller va API Resource

1

`Route::apiResource` qaysi metodlarni yaratadi va `Route::resource` dan farqi nimada?

Savol 1

2

Nima uchun modelni API dan to‘g‘ridan-to‘g‘ri qaytarmasdan API Resource ishlatamiz?

Savol 2

3

Yangi resurs yaratilganda Laravel 201 statusni qanday aniqlaydi?

Savol 3

03

Versiyalash, sahifalash va filtrlash

/api/v1 · buzuvchi o‘zgarishlar · offset va cursor · filter va sort · rate limit

Versiya | Sahifalash | Filtr | Rate limit

Qaysi o‘zgarish yangi versiya talab qiladi?

✗ Buzuvchi — faqat /api/v2 da

- maydonni o‘chirish yoki qayta nomlash: `name` → `title`
- turini o‘zgartirish: `"price": "385000.00"` → `385000`
- yangi majburiy parametr: `warehouse_id` endi shart
- status kod ma’nosini o‘zgartirish: 409 o‘rniga 422
- sahifalash formatini o‘zgartirish

✓ Buzmaydi — /api/v1 ga qo‘shiladi

- yangi ixtiyoriy maydon: `"barcode": null`
- yangi endpoint: `GET /api/v1/warehouses`
- yangi ixtiyoriy filtr: `?is_active=1`
- xato `detail` matnini aniqlashtirish
- unumdorlikni oshirish, indeks qo‘shish

Eskirgan versiya: hujjatda tugash sanasi va javobda `Sunset: Sat, 01 May 2027 00:00:00 GMT` sarlavhasi.

SAHIFALASH META

meta qanday hisoblanadi (sof PHP)

```php
<?php
function meta(int $total, int $perPage, int $page): array
{
    $empty = $total === 0;
    return [
        'current_page' => $page,
        'from' => $empty ? null : ($page - 1) * $perPage + 1,
        'last_page' => max(1, (int) ceil($total / $perPage)),
        'to' => $empty ? null : min($page * $perPage, $total),
        'total' => $total,
    ];
}

echo json_encode(meta(93, 20, 1)), PHP_EOL;
echo json_encode(meta(93, 20, 5)), PHP_EOL;
echo json_encode(meta(0, 20, 1)), PHP_EOL;
```

▶ NATIJA

```
{"current_page":1,"from":1,"last_page":5,"to":20,"total":93}
{"current_page":5,"from":81,"last_page":5,"to":93,"total":93}
{"current_page":1,"from":null,"last_page":1,"to":null,"total":0}
```

`ceil($total / $perPage)`

93 / 20 = 4.65 → 5 sahifa

`min($page * $perPage, $total)`

oxirgi sahifada `to` = 93, 100 emas

`$empty ? null`

bo‘sh ro‘yxatda `from` va `to` — `null` (Laravel ham shunday)

`max(1, ...)`

bo‘sh ro‘yxatda ham `last_page` = 1

JSON JAVOB

Sahifalangan Resource collection

```
{
  "data": [
    {"id": 15, "sku": "UN-001", "name": "Un oliy nav, 50 kg"}
  ],
  "links": {
    "first": "https://omborpro.test/api/v1/products?page=1",
    "last": "https://omborpro.test/api/v1/products?page=5",
    "prev": null,
    "next": "https://omborpro.test/api/v1/products?page=2"
  },
  "meta": {
    "current_page": 1, "from": 1, "last_page": 5,
    "path": "https://omborpro.test/api/v1/products",
    "per_page": 20, "to": 20, "total": 93
  }
}
```

▶ NATIJA

cursorPaginate() da meta: path, per_page, next_cursor, prev_cursor (total va last_page yo‘q — COUNT(*) bajarilmaydi)

`data`

joriy sahifa yozuvlari (`ProductResource` orqali)

`links.next`

mijoz keyingi sahifani URL ni o‘zi tuzmasdan oladi

`meta.total`

jami yozuvlar — offset sahifalashda `COUNT(*)` bilan

`meta.links`

Laravel bu yerga sahifa tugmalari ro‘yxatini ham qo‘shadi (qisqartirildi)

Ruxsat etilgan parametrlar ro‘yxati

```php
public function index(Request $request)
{
    $request->validate([
        'per_page' => ['integer', 'between:1,100'],
        'sort' => ['in:name,-name,price,-price,created_at,-created_at'],
        'category_id' => ['integer', 'exists:categories,id'],
    ]);
    $sort = $request->string('sort', 'name')->toString();
    $direction = str_starts_with($sort, '-') ? 'desc' : 'asc';

    return ProductResource::collection(
        Product::with('category')
            ->when($request->integer('category_id'),
                fn ($q, $id) => $q->where('category_id', $id))
            ->orderBy(ltrim($sort, '-'), $direction)
            ->paginate($request->integer('per_page', 20))
    );
}
```

▶ NATIJA

```
GET /api/v1/products?sort=-price&category_id=3  → 200, qimmatidan
arzoniga
GET /api/v1/products?sort=password              → 422 {"errors":
{"sort": [...]}}
GET /api/v1/products?per_page=1000              → 422
```

`in:name,-name,...`

saralash ustuni oq ro‘yxatdan — SQL injection va ichki ustunlar yopiq

`$request->string()`

Laravel `Stringable` qaytaradi; `toString()` — oddiy satr

`when(..., fn ...)`

filtr faqat parametr berilganda qo‘llanadi

`between:1,100`

katta `per_page` jim kesilmaydi, 422 bilan rad etiladi

TUSHUNISHNI TEKSHIRISH

O‘ylang – Juftlikda – Bo‘lishing · 3 daqiqa

3-bo‘lim yakuni: Versiyalash, sahifalash va filtrlash

1

`price` maydonini satrdan songa o‘zgartirish nima uchun yangi versiya talab qiladi?

Savol 1

2

Offset va cursor sahifalashning asosiy farqi nimada?

Savol 2

3

Nima uchun `sort` parametrini oq ro‘yxat bilan tekshirish kerak?

Savol 3

MA’RUZADAGI MASHQ

Har kim o‘z qurilmasida · 10 daqiqa · O‘rta · 20 ball

2-topshiriq: Sahifalash meta ma’lumoti

Laravel `paginate()` javobidagi `meta` blokini hisoblang. Jami yozuvlar `$total`, sahifa hajmi `$perPage` va joriy sahifa `$page` berilgan. `last_page` — kamida 1. `from` — sahifadagi birinchi yozuv tartib raqami (1 dan), `to` — oxirgisi; agar sahifada yozuv bo‘lmasa (bo‘sh ro‘yxat yoki `$page > last_page`), ikkalasi ham `null`. `$perPage` 1–100 oralig‘ida bo‘lmasa, `$page < 1` yoki `$total < 0` bo‘lsa — `InvalidArgumentException`.

Imzo: `function sahifa_meta(int $total, int $perPage, int $page): array`

Qaytaradi: `['current_page' => int, 'per_page' => int, 'total' => int, 'last_page' => int, 'from' => ?int, 'to' => ?int]` — shu tartibda.

MISOL · CHAQIRUV

```
sahifa_meta(93, 20, 1)
```

MISOL · NATIJA

```
{"current_page":1,"per_page":20,"total":93,"last_page":5,"from":1,"to":20}
```

Telefon kamerasi bilan skanerlang

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

04

Xatolar formati: Problem Details (RFC 7807)

application/problem+json · type, title, status, detail, instance · withExceptions · maxfiylik

RFC 7807 | problem+json | withExceptions | 409

RFC 7807 / 9457

Problem Details maydonlari

Har bir xato bir xil shaklda: `Content-Type: application/problem+json` va beshta standart maydon.

type va title

Xato turi URI si va uning o‘zgarmas nomi: `.../problems/insufficient-stock`, “Qoldiq yetarli emas”.

status

HTTP status kodi takrori: `409` — proxy kodni o‘zgartirsa ham asl qiymat saqlanadi.

detail va instance

Aynan shu holat izohi va so‘rov yo‘li: “UN-001: mavjud 3, so‘ralgan 5”, `/api/v1/orders/42/confirm`.

Kengaytmalar

Qo‘shimcha maydonlar: `errors` (validatsiya), `available`, `trace_id`.

Mijoz kodi `type` ga qarab qaror qiladi; `detail` matni o‘zgarishi mumkin, unga tayanmang.

Istisnodan problem+json gacha

```php
<?php
final class InsufficientStockException extends DomainException {}

function problem(Throwable $e, string $uri): array
{
    [$status, $title] = match (true) {
        $e instanceof DomainException => [409, 'Biznes-qoida buzildi'],
        $e instanceof InvalidArgumentException => [422, 'Validatsiya'],
        default => [500, 'Ichki xato'],
    };
    $detail = $status === 500 ? 'Kutilmagan xato' : $e->getMessage();
    return ['type' => 'about:blank', 'title' => $title,
        'status' => $status, 'detail' => $detail, 'instance' => $uri];
}
$e = new InsufficientStockException('UN-001: mavjud 3, so‘ralgan 5');
$flags = JSON_PRETTY_PRINT | JSON_UNESCAPED_UNICODE
    | JSON_UNESCAPED_SLASHES;
echo json_encode(problem($e, '/api/v1/orders/42/confirm'), $flags);
```

▶ NATIJA

```
{
    "type": "about:blank",
    "title": "Biznes-qoida buzildi",
    "status": 409,
    "detail": "UN-001: mavjud 3, so‘ralgan 5",
    "instance": "/api/v1/orders/42/confirm"
}
```

`match (true)`

istisno turiga qarab status va sarlavha tanlanadi

`$e instanceof DomainException`

`InsufficientStockException` ham shu shartga tushadi (meros)

`$status === 500`

kutilmagan xatoda ichki xabar mijozga chiqmaydi

`JSON_UNESCAPED_UNICODE`

o‘zbekcha harflar `\u2018` ga aylanmaydi

BOOTSTRAP/APP.PHP

Laravel 12 da xatolarni render qilish

```php
return Application::configure(basePath: dirname(__DIR__))
    // ->withRouting(...)->withMiddleware(...)
    ->withExceptions(function (Exceptions $exceptions) {
        $exceptions->shouldRenderJsonWhen(
            fn (Request $r, Throwable $e) => $r->is('api/*'));
        $exceptions->render(function (DomainException $e, Request $r) {
            if (! $r->is('api/*')) {
                return null; // web: odatiy xato sahifasi
            }
            return response()->json([
                'type' => 'https://omborpro.test/problems/conflict',
                'title' => 'Biznes-qoida buzildi',
                'status' => 409,
                'detail' => $e->getMessage(),
                'instance' => $r->getRequestUri(),
            ], 409, ['Content-Type' => 'application/problem+json']);
        });
    })->create();
```

▶ NATIJA

```
POST /api/v1/orders/42/confirm
HTTP/1.1 409 Conflict
Content-Type: application/problem+json

{"type": "https://omborpro.test/problems/conflict", "status": 409,
 "detail": "UN-001: mavjud 3, so‘ralgan 5", ...}
```

shouldRenderJsonWhen

`api/*` uchun har doim JSON — HTML xato sahifasi yo‘q

render(DomainException ...)

servis qatlamidagi barcha biznes-xatolar bitta joyda

return null

web marshrutlarida standart render saqlanadi

application/problem+json

RFC 7807 media turi sarlavhada

Yomon va yaxshi xato javobi

✗ Noto‘g‘ri

- `200 OK + {"success": false, "msg": "xato"}`
- har endpointda boshqa format: `error`, `message`, `msg`
- 500 javobida stack trace va SQL (`APP_DEBUG=true`)
- API so‘roviga HTML xato sahifasi yoki 302 redirect

✓ To‘g‘ri

- mos status: 404, 409, 422, 429
- yagona format: `type`, `title`, `status`, `detail`, `instance`
- 500 da umumiy matn + `trace_id`, tafsilot logda
- `api/*` uchun har doim JSON (`shouldRenderJsonWhen`)

TUSHUNISHNI TEKSHIRISH

O‘ylang – Juftlikda – Bo‘lishing · 3 daqiqa

4-bo‘lim yakuni: Xatolar formati: Problem Details (RFC 7807)

1

RFC 7807 bo‘yicha xato javobining standart maydonlarini sanang.

Savol 1

2

`InsufficientStockException` qanday qilib controllerda `try/catch` siz 409 javobga aylanadi?

Savol 2

3

Nima uchun productionda `APP_DEBUG=false` bo‘lishi shart?

Savol 3

MA’RUZADAGI MASHQ

Har kim o‘z qurilmasida · 10 daqiqa · O‘rta · 20 ball

3-topshiriq: Problem Details javobi

Ichki xato kodidan RFC 7807 javobini yasang. Ma’lum kodlar: `validation` (422, “Validatsiya xatosi”), `not_found` (404, “Resurs topilmadi”), `unauthenticated` (401, “Autentifikatsiya talab qilinadi”), `forbidden` (403, “Ruxsat yo‘q”), `insufficient_stock` (409, “Qoldiq yetarli emas”), `invalid_transition` (409, “Holat o‘tishi taqiqlangan”). Ular uchun `type` = `https://omborpro.test/problems/` + kod (pastki chiziq `-` ga almashtiriladi). `detail` bo‘sh satr bo‘lsa, o‘rniga `title` yoziladi. Noma’lum kod — 500, `type = about:blank`, `title` = “Ichki xato” va `detail` har doim “Kutilmagan xato” (ichki xabar mijozga chiqmaydi).

Imzo: `function problem(string $kod, string $detail, string $instance): array`

Qaytaradi: `['type' => string, 'title' => string, 'status' => int, 'detail' => string, 'instance' => string]` — shu tartibda.

MISOL · CHAQIRUV

```
problem('insufficient_stock', 'UN-001: mavjud 3, so‘ralgan 5', '/api/v1/orders/42/confirm')
```

MISOL · NATIJA

```
{"type":"https://omborpro.test/problems/insufficient-stock","title":"Qoldiq yetarli emas","status":409,"detail":"UN-001: mavjud 3, so‘ralgan 5","instance":"/api/v1/orders/42/confirm"}
```

Telefon kamerasi bilan skanerlang

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

05

Autentifikatsiya: token, JWT va Sanctum

Bearer token · JWT tuzilmasi va imzo · exp va refresh · jwt-auth · Sanctum bilan solishtirish

Bearer | JWT | HS256 | jwt-auth | Sanctur

header.payload.signature

JWT — imzolangan, lekin shifrlanmagan token. Uch qism base64url kodlangan va nuqta bilan birlashtirilgan.

Header

`{"alg":"HS256","typ":"JWT"}` → `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9`

Payload

Da’volar: `sub`, `iat`, `exp`, `jti` + `role`. Har kim o‘qiy oladi — maxfiy ma’lumot yozilmaydi.

Signature

HS256: `HMAC-SHA256(header.payload, JWT_SECRET)`. O‘zgartirilgan payload imzodan o‘tmaydi.

Tekshiruv tartibi: 3 qism bormi → `alg` kutilganmi → imzo `hash_equals` → `exp` > hozirgi vaqt.

SOF PHP

HS256 imzosini yasash va tekshirish

```php
<?php
function b64url(string $s): string
{
    return rtrim(strtr(base64_encode($s), '+/', '-_'), '=');
}
$key = 'faqat-test-uchun-sir-kalit-2026';
$sign = fn ($d) => b64url(hash_hmac('sha256', $d, $key, true));
$h = b64url(json_encode(['alg' => 'HS256', 'typ' => 'JWT']));
$p = b64url(json_encode(['sub' => 7, 'role' => 'menejer']));
$jwt = "$h.$p." . $sign("$h.$p");
echo $h, PHP_EOL;
[$h2, $p2, $s2] = explode('.', $jwt);
echo base64_decode(strtr($p2, '-_', '+/')), PHP_EOL;
$admin = b64url(json_encode(['sub' => 7, 'role' => 'admin']));
foreach ([$p2, $admin] as $payload) {
    $ok = hash_equals($sign("$h2.$payload"), $s2);
    echo $ok ? "imzo to‘g‘ri\n" : "soxta token\n";
}
```

▶ NATIJA

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9
{"sub":7,"role":"menejer"}
imzo to‘g‘ri
soxta token
```

`b64url()`

base64 → URL uchun xavfsiz: `+/` → `-_`, `=` yo‘q

`hash_hmac(..., true)`

xom (binary) HMAC-SHA256, keyin base64url

`base64_decode($p2)`

payload sirsiz o‘qiladi — u shifrlanmagan

`hash_equals()`

doimiy vaqtli solishtirish; `role: admin` ga almashtirilgan payload rad etiladi

Login: token berish

```php
public function login(Request $request)
{
    $credentials = $request->validate(
        ['email' => 'required|email', 'password' => 'required']);

    $token = auth('api')->attempt(
        $credentials + ['is_active' => true]);
    if (! $token) {
        return response()->json(
            ['message' => 'Email yoki parol noto‘g‘ri'], 401);
    }
    return response()->json([
        'access_token' => $token,
        'token_type' => 'bearer',
        'expires_in' => auth('api')->factory()->getTTL() * 60,
    ]);
}
```

▶ NATIJA

```
POST /api/v1/auth/login → 200
{"access_token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3Mi...",
 "token_type": "bearer", "expires_in": 3600}
```

`auth('api')`

`config/auth.php` dagi `'api' => ['driver' => 'jwt', ...]` guard

`attempt($credentials + ...)`

parolni xesh bilan tekshiradi; `is_active = true` sharti qo‘shiladi

`getTTL() * 60`

`jwt.ttl` daqiqada (standart 60) → sekund

401

qaysi maydon xato ekanini aytmaymiz — hisoblarni sanashdan himoya

`refresh()` / `logout()`

yangi token berish va joriysini qora ro‘yxatga yozish

TANLOV

Sanctum tokeni va JWT

Sanctum (opaque token)

- tasodifiy satr, bazada SHA-256 xesh
- har so‘rovda bazaga murojaat
- darhol bekor qilish: yozuvni o‘chirish
- huquqlar: `createToken('mobil', ['orders:read'])`
- SPA uchun cookie-sessiya rejimi ham bor

JWT (jwt-auth)

- imzolangan JSON, da’volar ichida
- bazasiz tekshiruv — servislar orasida qulay
- bekor qilish qiyin: qisqa `ttl` + refresh + blacklist
- payload o‘qiladi — maxfiy ma’lumot yo‘q
- kalitni himoyalash shart: `JWT_SECRET` faqat `.env` da

Ikkala holatda ham: faqat HTTPS, tokenni URL ga yozmaslik (loglarga tushadi), login uchun `throttle`.

TUSHUNISHNI TEKSHIRISH

O‘ylang – Juftlikda – Bo‘lishing · 3 daqiqa

5-bo‘lim yakuni: Autentifikatsiya: token, JWT va Sanctum

1

JWT payloadiga nima uchun parol yoki STIR yozib bo‘lmaydi?

Savol 1

2

JWT imzosini tekshirishda nima uchun `==` emas, `hash_equals()` ishlatiladi?

Savol 2

3

Sanctum tokenini JWT dan osonroq bekor qilish mumkinligi sababi nimada?

Savol 3

06

OpenAPI hujjatlari: Scribe va Postman

OpenAPI 3 · knuckleswtf/scribe · docblock teglari · /docs · Postman kolleksiyasi

OpenAPI | Scribe | Postman | Hujjat

OPENAPI 3

Spetsifikatsiya bo‘lagi

```yaml
openapi: 3.0.3
info: {title: OmborPro API, version: 1.0.0}
servers:
  - url: https://omborpro.test/api/v1
paths:
  /products/{id}:
    get:
      summary: Mahsulotni olish
      security: [{bearerAuth: []}]
      parameters:
        - {name: id, in: path, required: true, schema: {type: integer}}
      responses:
        '200': {description: Mahsulot (ProductResource)}
        '401': {description: Token yo‘q yoki muddati o‘tgan}
        '404': {description: Topilmadi (problem+json)}
components:
  securitySchemes:
    bearerAuth: {type: http, scheme: bearer, bearerFormat: JWT}
```

▶ NATIJA

Scribe → /docs (HTML hujjat), /docs.openapi (shu YAML), /docs.postman (kolleksiya)

servers

bazaviy URL — versiya shu yerda

paths

har bir endpoint: parametrlar va mumkin bo‘lgan javoblar

security

bu endpoint Bearer JWT talab qiladi

securitySchemes

token turi bir marta, `components` da tavsiflanadi

SCRIBE

Controller dagi docblock teglari

```php
/**
 * Mahsulotlar ro‘yxati
 *
 * Nomi bo‘yicha saralangan, sahifalangan katalog.
 *
 * @group Mahsulotlar
 * @authenticated
 * @queryParam category_id integer Kategoriya filtri. Example: 3
 * @queryParam sort string Saralash maydoni. Example: -price
 * @queryParam per_page integer 1–100 oralig‘ida. Example: 20
 * @apiResourceCollection App\Http\Resources\ProductResource
 * @apiResourceModel App\Models\Product with=category paginate=20
 */
public function index(Request $request)
{
    // ...
}
```

▶ NATIJA

```
$ php artisan scribe:generate
Processing route: [GET,HEAD] api/v1/products
Processed route: [GET,HEAD] api/v1/products
...
Checking for any pending upgrades to your config file...
```

`@group`

hujjatdagi bo‘lim: Mahsulotlar, Buyurtmalar, Autentifikatsiya

`@authenticated`

“token kerak” belgisi va Postman da `Authorization` sarlavhasi

`@queryParam ... Example:`

parametr turi, izohi va namunaviy qiymati

`@apiResourceModel ... paginate=20`

javob namunasi factory + Resource dan, sahifalash bilan

ISH OQIMI

Koddan Postman kolleksiyasigacha

1

O‘rnatish

`composer require --dev knuckleswtf/scribe`, `vendor:publish --tag=scribe-config`

2

Sozlash

`config/scribe.php`: `routes` → `api/*`, `auth.enabled` → Bearer, `postman` va `openapi` yoqilgan

3

Izohlash

controllerlarga `@group`, `@authenticated`, `@queryParam`, `@response` teglari

4

Yaratish

`php artisan scribe:generate` → `/docs`, `/docs.openapi`, `/docs.postman`

5

Postman

kolleksiyani import, `baseUrl` va `token` muhit o‘zgaruvchilari

6

Avtomatlash

`newman run` bilan kolleksiyani buyruq qatorida yugurtirish

Har bir API o‘zgarishidan keyin `scribe:generate` qayta ishga tushiriladi — hujjat koddan orqada qolmasin.

TUSHUNISHNI TEKSHIRISH

O‘ylang – Juftlikda – Bo‘lishing · 3 daqiqa

6-bo‘lim yakuni: OpenAPI hujjatlari: Scribe va Postman

1

OpenAPI hujjatining uchta asosiy bo‘limini ayting.

Savol 1

2

Scribe so‘rov tanasi parametrlarini qayerdan oladi?

Savol 2

3

Postman muhit o‘zgaruvchilari (`baseUrl`, `token`) nima uchun kerak?

Savol 3

XULOSA

Asosiy xulosalar

1 Resurs — ot, metod — fe’l

`GET /api/v1/products`, `POST /api/v1/orders/42/confirm`; URI da `get`, `create` yo‘q.

2 Status kod — signal

201 + `Location`, 204, 401/403, 404, 409 (biznes-qoida), 422 (validatsiya), 429.

3 API Resource — shartnoma

Maydonlar qo‘lda tanlanadi, pul satr, sana ISO 8601, relationlar `whenLoaded`.

4 Versiya va sahifalash

`/api/v1`, buzuvchi o‘zgarish faqat yangi versiyada; `per_page` 1–100, cursor — katta jurnallar uchun.

5 Yagona xato formati

RFC 7807: `type`, `title`, `status`, `detail`, `instance`; `withExceptions()` da markazlashgan.

6 JWT va hujjat

Imzolangan, shifrlanmagan token, qisqa `ttl`; Scribe → OpenAPI + Postman kolleksiyasi.

TAKRORLASH

Yakunlashdan oldin sakkizta savol

1. Idempotent metod nima? Misol keltiring.
2. Buyurtma yaratildi — qaysi status va sarlavha?
3. Qoldiq yetarli emas — qaysi status?
4. `whenLoaded('category')` nima qiladi?
5. Offset sahifalashning kamchiligi?
6. RFC 7807 media turi qanday?
7. JWT ning uch qismi?
8. Scribe hujjatini qanday yangilaymiz?

AMALIY MASHG‘ULOT

7-amaliy mashg‘ulot: 5 ta PHP topshiriq va loyiha bosqichi

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.

CHIQISH CHIPTASI

Darsni tark etishdan oldin

Javoblaringizni qog‘ozga yozing va o‘qituvchiga topshiring.

1

OmborPro dagi buyurtmani bekor qilish uchun endpoint, metod va muvaffaqiyatli/xato status kodlarini yozing.

2

JWT payloadiga nima yozish mumkin va nima mumkin emas — har biriga bitta misol.

3

Bugungi mavzudan sizning jamoa loyihangiz API sida birinchi navbatda nimani o‘zgartirasiz?