6

OLTINCHI MA’RUZA

Korxona ilovalarini dasturlash · 6-mavzu

Korxona ilovalari interfeysi

SSR va SPA · ma’lumotlarga boy ekranlar · filtrlash va saralash · sahifalash · formalar · dashboardlar

Blade | Livewire 3 | Sahifalash | Formalar | Dashboard

ISINISH

Omborchi 12 000 ta mahsulot ichidan “UN-” bilan boshlanadigan va qoldig‘i kam bo‘lganlarini qanday topadi?

Excel dagidek barcha qatorni bitta sahifaga chiqarib qo‘ysak-chi? Avval taxmin qiling.

1

12 000 qatorli HTML jadval brauzerda qancha vaqtda ochiladi va uni kim oxirigacha varaqlaydi?

2

Filtrlangan ro‘yxat havolasini hamkasbingizga yuborsangiz, u aynan shu ro‘yxatni ko‘rishi kerakmi?

3

Qidiruv maydoniga har bir harf yozilganda serverga so‘rov ketishi yaxshimi yoki yomonmi?

O‘QUV MAQSADLARI

Bu ma’ruza oxirida siz quyidagilarni bajara olasiz…

1. **Farqlash** — server tomonida render (Blade, Livewire), Inertia.js va SPA yondashuvlarini hamda ularni tanlash mezonlarini
2. **Loyihalash** — ma’lumotlarga boy ro‘yxat ekranini: filtrlar paneli, jadval, sahifalash, bo‘sh va yuklanish holatlari
3. **Yozish** — `when()`, oq ro‘yxatli saralash va `paginate()->withQueryString()` bilan xavfsiz ro‘yxat so‘rovini
4. **Tanlash** — `paginate`, `simplePaginate` va `cursorPaginate` dan vaziyatga mosini
5. **Qo‘llash** — FormRequest, `old()`, `@error` va Livewire validatsiyasini hujjat formalarida
6. **Qurish** — Livewire 3 komponentini `#[Url]`, `WithPagination` va `wire:model.live.debounce` bilan
7. **Baholash** — dashboard ko‘rsatkichlarini amaliy foydasi, rolga mosligi va hisoblash narxi bo‘yicha

01

Server tomonida render va SPA

SSR · SPA · Inertia.js · Livewire 3 · tanlov mezonlari

SSR | SPA | Livewire | Inertia.js

IKKI YONDASHUV

Server render (SSR) va bir sahifali ilova (SPA)

Server tomonida render (Blade, Livewire)

- HTML serverda yasaladi, brauzer tayyor sahifani oladi
- Bitta kod bazasi, bitta til — PHP
- Sessiya va `@can` bilan ruxsatlar (5-mavzu)
- Validatsiya bir joyda — FormRequest
- Kamchilik: klassik Blade da har harakat — to‘liq qayta yuklash

Bir sahifali ilova (Vue, React)

- Brauzer JSON oladi va sahifani o‘zi chizadi
- Juda yuqori interaktivlik, offline rejim mumkin
- Mobil ilova bilan umumiy API
- Narxi: frontend + API, ikki joyda validatsiya
- Token autentifikatsiya, build jarayoni, JS mutaxassis

Back-office uchun SEO ahamiyatsiz — hal qiluvchi omillar: jamoa ko‘nikmasi, interaktivlik va muddat.

SPEKTR

To‘rt yondashuv: qanday ishlaydi va qachon mos

Yondashuv	Qanday ishlaydi	Qachon mos
Blade (klassik SSR)	Controller → Blade → HTML; har harakat — yangi sahifa	Oddiy CRUD, hisobotlar, ichki portallar
Blade + Livewire 3	Komponent holati serverda; AJAX bilan faqat o‘zgargan HTML	Jadval, filtr, dinamik formalar — PHP jamoa
Inertia.js + Vue/React	Laravel controller sahifaga props beradi; API shart emas	Boy UI kerak, lekin bitta Laravel ilova
SPA + REST API	Alohida JS ilova, server faqat JSON qaytaradi	Mobil/offline, bir nechta klient, katta frontend jamoa

OmborPro: Blade + Livewire 3. REST API 7-mavzuda — mobil klient yoki tashqi tizimlar uchun qo‘shiladi.

Bitta Livewire so‘rovining aylanishi

1

Birinchi render

Server komponentni HTML qilib beradi; ichida holat surati (snapshot) bor

2

Harakat

Foydalanuvchi qidiruvga yozadi: `wire:model.live.debounce.300ms="search"`

3

AJAX so‘rov

Livewire JS `POST /livewire/update` ga snapshot va o‘zgarishlarni yuboradi

4

Serverda

Komponent tiklanadi, `$search` yangilanadi, `updatedSearch()` hook, `render()`

5

DOM yangilanishi

Javobdagi HTML bilan faqat o‘zgargan qismlar almashtiriladi (morph)

Har bir harakat — server so‘rovi. Shuning uchun `debounce`, `wire:model.blur` va kerakli joyda `.live` muhim.

▶ INTERAKTIV

Qaysi yondashuv yetarli?

Har bir ekranni mos qutiga joylashtiring. (Interaktiv slaydlarda bosib bajariladi.)

Buxgalterning hisob-fakturalar ro‘yxati va filtrlari

Omborchining kirim-chiqim formasi

Kuryerlar uchun internetsiz ishlaydigan mobil ilova

Admin uchun foydalanuvchilar va rollar sahifasi

Ombor xaritasida javonlarni sudrab joylashtiruvchi interaktiv sxema

Rahbar uchun kunlik savdo dashboardi

Mijozlar o‘zi buyurtma beradigan iOS/Android ilova

Buyurtma hujjati: qator qo‘shish va jami summani darhol hisoblash

Blade + Livewire yetarli

SPA / mobil klient kerak

TUSHUNISHNI TEKSHIRISH

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

1-bo‘lim yakuni: Server tomonida render va SPA

1

SSR va SPA ning korxona back-office ilovasi uchun asosiy farqi nimada?

Savol 1

2

Livewire 3 da komponent holati qayerda saqlanadi va brauzer nima oladi?

Savol 2

3

Qaysi talab paydo bo‘lsa, Livewire o‘rniga alohida SPA/mobil klient haqida o‘ylash kerak?

Savol 3

02

Ma’lumotlarga boy ekranlar

Ro‘yxat ekrani anatomiyasi · jadval dizayni · formatlash · eager loading

Ro‘yxat ekrani | Jadval | Formatlash | N+1

ANATOMIYA

Ro‘yxat ekrani nimalardan iborat

Mahsulotlar, buyurtmalar, hisob-fakturalar — barcha ro‘yxat ekranlari bir xil tuzilishda bo‘lsa, foydalanuvchi tez o‘rganadi.

1 Sarlavha va asosiy amal

“Mahsulotlar” + “Yangi mahsulot” tugmasi (faqat `products.manage` ruxsati borlarga)

2 Filtrlar paneli

Qidiruv, kategoriya, holat, sana oralig‘i; “Tozalash” tugmasi

3 Jadval

Saralanadigan ustunlar, o‘ngga tekislangan raqamlar, holat belgilari

4 Sahifalash

“1–20 / 1 248” va sahifa havolalari; sahifadagi qatorlar soni tanlovi

5 Bo‘sh holat

“Hech narsa topilmadi” + filtrni tozalash havolasi

6 Yuklanish holati

`wire:loading` — “Qidirilmoqda…”, jadval xiralashadi

BLADE JADVAL

Mahsulotlar jadvali: bo‘sh holat bilan

```blade
<table class="table table-sm table-hover">
    <thead class="sticky-top">
        <tr><th>SKU</th><th>Nomi</th><th>Kategoriya</th>
            <th class="text-end">Narx, so‘m</th></tr>
    </thead>
    <tbody>
    @forelse ($products as $p)
        <tr>
            <td><code>{{ $p->sku }}</code></td>
            <td>{{ $p->name }}</td>
            <td>{{ $p->category->name }}</td>
            <td class="text-end">{{ som($p->price) }}</td>
        </tr>
    @empty
        <tr><td colspan="4">Hech narsa topilmadi</td></tr>
    @endforelse
    </tbody>
</table>
```

▶ NATIJA

```
SKU      Nomi                Kategoriya     Narx, so‘m
UN-001   Un oliy nav 50 kg   Un mahsulotlari    385 000
GR-007   Guruch lazer 25 kg  Yorma               41 000
```

`@forelse … @empty`

ro‘yxat bo‘sh bo‘lsa — bo‘sh holat qatori

`{{ … }}`

qiymat `htmlspecialchars` bilan ekranlanadi — XSS dan himoya

`$p->category->name`

controllerda `with('category')` bo‘lmasa — N+1

`som()`

o‘zimiz yozgan formatlash yordamchisi (keyingi slayd)

`text-end`

pul va raqamlar o‘ngga tekislanadi

FORMATLASH

Pul va qoldiq belgisi — sof PHP

```php
function som(int $x): string
{
    return number_format($x, 0, '.', ' ') . ' so‘m';
}

function badge(int $qty, int $res, int $min): string
{
    $free = $qty - $res;
    return match (true) {
        $free <= 0 => 'tugagan',
        $free < $min => 'kam',
        default => 'yetarli',
    };
}

echo som(1250000), "\n";
echo badge(120, 20, 50), ' | ', badge(30, 0, 50), "\n";
echo badge(5, 5, 10), ' | ', som(-45000);
```

▶ NATIJA

```
1 250 000 so‘m
yetarli | kam
tugagan | -45 000 so‘m
```

`number_format($x, 0, '.', ' ')`

ming ajratgich — bo‘sh joy, kasr qismi yo‘q

`$qty - $res`

mavjud qoldiq = quantity − reserved (3-mavzudagi `stocks`)

`match (true)`

shartlar tartib bilan tekshiriladi — birinchi mos kelgani

Blade da

`<span class="badge">{{ badge(...) }}</span>` — rang + matn

TUSHUNISHNI TEKSHIRISH

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

2-bo‘lim yakuni: Ma’lumotlarga boy ekranlar

1

Ro‘yxat ekranida “bo‘sh holat” nima uchun kerak?

Savol 1

2

Jadvalda pul ustuni qanday tekislanadi va formatlanadi?

Savol 2

3

Jadvalda kategoriya nomi chiqarilganda N+1 muammosining oldini qanday olasiz?

Savol 3

MA’RUZADAGI MASHQ

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

1-topshiriq: Qoldiq belgisi (badge)

Mahsulotlar jadvalida har bir qator yonida qoldiq holati ko‘rsatiladi. Mavjud qoldiq = `quantity - reserved`. Mavjud qoldiq 0 bo‘lsa — `tugagan`, `minStock` dan kichik bo‘lsa — `kam`, aks holda — `yetarli`. Agar `quantity` yoki `reserved` manfiy bo‘lsa yoki `reserved > quantity` bo‘lsa, bu invariant buzilganini bildiradi — `DomainException` tashlang.

Imzo: `function qoldiq_badge(int $quantity, int $reserved, int $minStock): string`

Qaytaradi: Satr: `'tugagan'`, `'kam'` yoki `'yetarli'`.

MISOL · CHAQIRUV	MISOL · NATIJA
`qoldiq_badge(120, 20, 50)`	`"yetarli"`



Telefon kamerasi bilan skanerlang

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

03

Filtrlash va saralash

Query string — holat manbai · when() · ILIKE qidiruv · oq ro‘yxatli saralash

Filtr | when() | orderBy | Whitelist

CONTROLLER

ProductController@index: filtr, saralash, sahifa

```php
public function index(Request $request): View
{
    $search = trim((string) $request->query('search'));
    $sort = in_array($request->query('sort'), self::SORTS, true)
        ? $request->query('sort') : 'name';
    $dir = $request->query('dir') === 'desc' ? 'desc' : 'asc';
    $products = Product::query()
        ->with('category:id,name')
        ->when($search !== '', fn ($q) => $q->where(
            fn ($w) => $w->where('name', 'ilike', "%{$search}%")
                ->orWhere('sku', 'ilike', "%{$search}%")))
        ->when($request->integer('category_id'),
            fn ($q, $id) => $q->where('category_id', $id))
        ->orderBy($sort, $dir)->orderBy('id')
        ->paginate(20)
        ->withQueryString();
    return view('products.index', compact('products'));
}
```

▶ NATIJA

```
GET /products?search=un&category_id=3&sort=price&dir=desc
SQL (soddalashtirilgan):
select * from products where (name ilike ? or sku ilike ?)
  and category_id = ? and deleted_at is null
  order by price desc, id asc limit 20 offset 0
bindings: ['%un%', '%un%', 3]
```

`self::SORTS`

`['sku', 'name', 'price', 'created_at']` — oq ro‘yxat

`when($cond, fn)`

qiymat bo‘sh bo‘lsa — shart qo‘shilmaydi

`where(fn ($w) => …)`

OR shartlar qavs ichida — boshqa filtrlarni buzmaydi

`->orderBy('id')`

barqaror tartib: sahifalar orasida takror yo‘q

`withQueryString()`

sahifa havolalarida filtrlar saqlanadi

Saralash parametrini xavfsiz tahlil qilish

```php
const SORTABLE = ['sku', 'name', 'price', 'created_at'];

function parseSort(?string $p): array
{
    $p = $p ?: 'name';
    $dir = str_starts_with($p, '-') ? 'desc' : 'asc';
    $col = ltrim($p, '-');
    if (!in_array($col, SORTABLE, true)) {
        return ['name', 'asc'];   // ruxsat etilmagan → standart
    }
    return [$col, $dir];
}

foreach (['-price', 'sku', null, 'password; DROP'] as $p) {
    echo implode(' ', parseSort($p)), "\n";
}
```

▶ NATIJA

```
price desc
sku asc
name asc
name asc
```

`?sort=-price`

JSON:API uslubi: `-` — kamayish tartibi

`in_array(..., true)`

qat’iy taqqoslash — tur ham tekshiriladi

`'password; DROP'`

oq ro‘yxatda yo‘q — SQL ga umuman yetib bormaydi

Muqobil

noma’lum qiymatda 400 xato qaytarish (API uchun qulay)

XAVFSIZ SARALASH

Foydalanuvchi ustun nomini tanlamasin

Noto‘g‘ri: kirish to‘g‘ridan-to‘g‘ri SQL ga

- `->orderBy($request->sort)` — istalgan ustun nomi
- `?sort=password` — maxfiy ustun bo‘yicha saralash
- `->orderByRaw($request->sort)` — SQL injection
- Noma’lum ustun — 500 xato va stack trace

To‘g‘ri: oq ro‘yxat va standart qiymat

- `SORTS = ['sku', 'name', 'price']`
- Yo‘nalish faqat `asc` yoki `desc`
- Noma’lum qiymat → standart saralash
- `->orderBy('id')` — barqaror tartib

Qiymatlar binding orqali himoyalanadi, ustun va yo‘nalish nomlari esa yo‘q — ularni faqat oq ro‘yxat himoyalaydi.

TUSHUNISHNI TEKSHIRISH

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

3-bo‘lim yakuni: Filtrlash va saralash

1

Filtr holatini nima uchun URL query string da saqlash kerak?

Savol 1

2

`->when($request->integer('category_id'), ...)` kategoriya berilmaganda nima qiladi?

Savol 2

3

Nima uchun saralashga `->orderBy('id')` qo‘shiladi?

Savol 3

MA’RUZADAGI MASHQ

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

2-topshiriq: Saralash parametrini tahlil qilish

URL dagi `sort` parametri vergul bilan ajratilgan ustunlar ro‘yxati: `-price,name` — avval narx bo‘yicha kamayish, keyin nom bo‘yicha o‘sish. Har bir qism `trim` qilinadi, bo‘sh qismlar e’tiborsiz qoldiriladi; bitta boshidagi `-` — `desc`, aks holda `asc`. Ustun `$allowed` oq ro‘yxatida bo‘lmasa, bitta ustun ikki marta uchrasa yoki ustunlar 3 tadan ko‘p bo‘lsa — `InvalidArgumentException`. Parametr `null` yoki faqat bo‘sh qismlardan iborat bo‘lsa — `[[$default, 'asc']]`.

Imzo: `function sort_parse(?string $param, array $allowed, string $default): array`

Qaytaradi: Juftliklar ro‘yxati: `[['price', 'desc'], ['name', 'asc']]` — URL dagi tartibda.

MISOL · CHAQIRUV

```
sort_parse('-price', ['sku', 'name', 'price'], 'name')
```

MISOL · NATIJA

```
[["price","desc"]]
```

Telefon kamerasi bilan skanerlang

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

04

Sahifalash

paginate · simplePaginate · cursorPaginate · links() · Livewire WithPagination

paginate()

cursorPaginate()

WithPagination

#[Url]

UCHTA USUL

paginate, simplePaginate, cursorPaginate

Usul	SQL	Nima beradi	Qachon
`paginate(20)`	`count(*) + limit 20 offset N`	Sahifa raqamlari, jami soni, `lastPage()`	Ma’lumotnomalar: mahsulotlar, kontragentlar
`simplePaginate(20)`	`limit 21 offset N` (count yo‘q)	Faqat “Oldingi / Keyingi”	Katta ro‘yxat, jami son kerak emas
`cursorPaginate(20)`	`where id > ? order by id limit 21`	Kursor; sahifa raqami yo‘q	Millionlab qatorli jurnal, cheksiz aylantirish

Har uchala usul ham `->withQueryString()` va `{{ $x->links() }}` bilan ishlaydi.

Sahifalash meta-ma’lumotini hisoblash

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

echo json_encode(meta(137, 20, 1)), "\n";
echo json_encode(meta(137, 20, 7)), "\n";
echo json_encode(meta(0, 20, 1));
```

▶ NATIJA

```
{"current_page":1,"last_page":7,"from":1,"to":20}
{"current_page":7,"last_page":7,"from":121,"to":137}
{"current_page":1,"last_page":1,"from":null,"to":null}
```

`ceil(137 / 20)`

7 sahifa; oxirgisida 17 ta qator

`max(1, …)`

bo‘sh ro‘yxatda ham `last_page = 1`

`from / to`

“121–137 / 137” yozuvi uchun; bo‘sh bo‘lsa `null`

Laravel da

`firstItem()`, `lastItem()`, `total()`, `lastPage()`

ProductTable: qidiruv URL da, filtr → 1-sahifa

```php
class ProductTable extends Component
{
    use WithPagination;
    #[Url(as: 'q', except: '')]
    public string $search = '';
    public function updatedSearch(): void
    {
        $this->resetPage();
    }
    public function render()
    {
        return view('livewire.product-table', [
            'products' => Product::with('category')
                ->search($this->search)->orderBy('name')
                ->paginate(20),
        ]);
    }
}
```

▶ NATIJA

```
/products?q=un&page=3  → qidiruv “gur” ga o‘zgardi
/products?q=gur         → 1-sahifa, jadval AJAX bilan yangilandi
```

`use WithPagination;`

`Livewire\WithPagination` — `page` URL da saqlanadi

`#[Url(as: 'q', except: '')]`

`Livewire\Attributes\Url`: `$search` ↔ `?q=`; bo‘sh bo‘lsa URL da yo‘q

`updatedSearch()`

`$search` o‘zgargandan keyin chaqiriladigan hook

`->search(...)`

Product modelidagi local scope (`scopeSearch`) — amaliyotda yozamiz

product-table.blade.php

```
<div>
    <input type="search" class="form-control"
           placeholder="SKU yoki nom bo‘yicha qidirish"
           wire:model.live.debounce.300ms="search">

    <div wire:loading wire:target="search" class="small">
        Qidirilmoqda…
    </div>

    <table class="table table-sm">
        @foreach ($products as $p)
            <tr wire:key="product-{{ $p->id }}"> … </tr>
        @endforeach
    </table>

    {{ $products->links() }}
</div>
```

▶ NATIJA

Foydalanuvchi “un” yozdi → 300 ms kutiladi → bitta so‘rov “Qidirilmoqda…” → jadval va sahifalash yangilanadi

`<div> … </div>`

Livewire komponenti — bitta ildiz element

`.live.debounce.300ms`

yozish to‘xtagach 300 ms dan keyin so‘rov; `.live` siz — keyingi harakatgacha kutadi

`wire:loading`

so‘rov davomida ko‘rinadi; `wire:target` — qaysi xossa uchun

`wire:key`

ro‘yxat qatorlari uchun noyob kalit — DOM to‘g‘ri yangilanadi

TUSHUNISHNI TEKSHIRISH

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

4-bo‘lim yakuni: Sahifalash

1

`paginate` va `simplePaginate` ning SQL darajasidagi farqi nima?

Savol 1

2

`stock_movements` jurnali uchun qaysi sahifalash usuli mos va nima uchun?

Savol 2

3

Livewire da qidiruv o‘zgarganda nima uchun `resetPage()` chaqiriladi?

Savol 3

MA’RUZADAGI MASHQ

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

3-topshiriq: Sahifalash havolalari oynasi

Sahifalash panelida hamma sahifa raqami ko‘rsatilmaydi. Har doim 1-sahifa, oxirgi sahifa va joriy sahifaning ikki tomonidagi `$window` tadan sahifa ko‘rsatiladi. Ko‘rsatiladigan qo‘shni raqamlar orasida faqat bitta sahifa tushib qolsa — o‘sha sahifa raqami qo‘yiladi, bittadan ko‘p bo‘lsa — `'...'`. `$last < 1`, `$current` 1..`$last` oralig‘ida bo‘lmasa yoki `$window < 0` bo‘lsa — `InvalidArgumentException`.

Imzo: `function sahifa_havolalari(int $current, int $last, int $window): array`

Qaytaradi: O‘sish tartibidagi ro‘yxat: butun sonlar va `'...'` satrlari, masalan `[1, '...', 8, 9, 10, 11, 12, '...', 20]`.

MISOL · CHAQIRUV

```
sahifa_havolalari(10, 20, 2)
```

MISOL · NATIJA

```
[1,"...",8,9,10,11,12,"...",20]
```

Telefon kamerasi bilan skanerlang

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

05

Formalar va validatsiya

FormRequest · old() va @error · PRG · Livewire real vaqt validatsiyasi · hujjat formasi

FormRequest | @error | wire:model.blur | OrderForm

FORMREQUEST

StoreProductRequest: ruxsat va qoidalar

```php
class StoreProductRequest extends FormRequest
{
    public function authorize(): bool
    {
        return $this->user()->can('products.manage');
    }
    public function rules(): array
    {
        $product = $this->route('product');   // tahrirda — model
        return [
            'sku' => ['required', 'regex:/^[A-Z]{2}-\d{3}$/',
                Rule::unique('products')->ignore($product)],
            'name' => ['required', 'string', 'max:150'],
            'unit' => ['required', Rule::in(Product::UNITS)],
            'price' => ['required', 'numeric', 'decimal:0,2', 'gt:0'],
        ];
    }
}
```

▶ NATIJA

```
POST /products  sku=un-1, price=0
302 → oldingi sahifa, sessiyada xatolar:
sku:   The sku field format is invalid.
price: The price field must be greater than 0.
(lang/uz/validation.php bilan — o‘zbekcha)
```

`authorize()`

`false` bo‘lsa — 403; ruxsat nomi 5-mavzudan

`Rule::unique(...)->ignore($product)`

tahrirda o‘z yozuvi “band” hisoblanmaydi

`Product::UNITS`

modeldagi konstanta: `['dona', 'kg', 'litr', 'quti']`

`decimal:0,2`

pul: ko‘pi bilan 2 kasr xona; `min_stock` uchun — `integer|min:0`

BLADE FORMA

old(), @error va @class bilan maydonlar

```blade
<form method="POST" action="{{ route('products.update', $product) }}">
    @csrf
    @method('PUT')
    <label for="sku" class="form-label">SKU *</label>
    <input id="sku" name="sku" value="{{ old('sku', $product->sku) }}"
        @class(['form-control', 'is-invalid' => $errors->has('sku')])>
    @error('sku')
        <div class="invalid-feedback">{{ $message }}</div>
    @enderror
    <label for="price" class="form-label">Narx, so‘m *</label>
    <input id="price" name="price" inputmode="decimal"
        value="{{ old('price', $product->price) }}"
        @class(['form-control', 'is-invalid' => $errors->has('price')])>
    @error('price')
        <div class="invalid-feedback">{{ $message }}</div>
    @enderror
    <button class="btn btn-primary">Saqlash</button>
</form>
```

▶ NATIJA

Xato bo‘lsa: maydon qizil ramkada, ostida xabar,
kiritilgan qiymat saqlanib qoladi (old).
Muvaffaqiyat: 302 → /products, “Mahsulot UN-001 saqlandi”

@csrf

CSRF tokeni — usiz 419 xato (5-mavzu)

@method('PUT')

HTML forma faqat GET/POST biladi — yashirin `_method`

old('sku', $product->sku)

avval kiritilgan qiymat, bo‘lmasa — bazadagi

@class([...])

shartli CSS sinflari: xato bo‘lsa `is-invalid`

HUJJAT FORMASI

OrderForm: massiv qoidalari va servis

```php
class OrderForm extends Component
{
    public ?int $counterpartyId = null;
    public array $lines = [['product_id' => null, 'quantity' => 1]];
    protected function rules(): array
    {
        return [
            'counterpartyId' => 'required|exists:counterparties,id',
            'lines.*.quantity' => 'required|integer|min:1',
        ];
    }
    public function save(OrderService $orders)
    {
        $this->validate();
        $order = $orders->create(OrderData::fromArray($this->all()));
        return $this->redirectRoute('orders.show', $order);
    }
}
```

▶ NATIJA

```
Xato: lines.0.quantity — The lines.0.quantity field must be at
least 1.
To‘g‘ri: draft buyurtma BY-2026-000124 → /orders/124
```

`lines.*.quantity`

har bir qator uchun qoida; xuddi shunday `lines.*.product_id — exists:products,id`

`$this->validate()`

`rules()` bo‘yicha; xatolar `@error` ga tushadi

OrderService

biznes-logika va tranzaksiya — 4-mavzudagi servisda

`redirectRoute()`

Livewire 3: saqlangandan keyin hujjat sahifasiga

▶ INTERAKTIV

Formalar bo‘yicha tezkor savollar

To‘g‘ri javobni tanlang. (Interaktiv slaydlarda bosib bajariladi.)

1 **Blade formada validatsiya muvaffaqiyatsiz bo‘lsa, Laravel nima qaytaradi?**

A) 200 va bo‘sh sahifa

B) 302 — oldingi sahifaga, xatolar va kiritilgan qiymatlar sessiyada

C) 500 xato

D) 404 xato

2 **Brauzerda `required` atributi bor. Serverda validatsiya kerakmi?**

A) Ha — brauzer tekshiruvini osongina chetlab o‘tish mumkin

B) Yo‘q, brauzer yetarli

C) Faqat adminlar uchun

D) Faqat API uchun

3 **Livewire da maydonni u fokusdan chiqqanda tekshirish uchun?**

A) `wire:click`

B) `wire:poll`

C) `wire:model.blur` + `#[Validate]`

D) `wire:key`

TUSHUNISHNI TEKSHIRISH

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

5-bo‘lim yakuni: Formalar va validatsiya

1

Nima uchun brauzer validatsiyasi yetarli emas?

Savol 1

2

Tahrirlash formasida SKU unikal qoidasi qanday yoziladi?

Savol 2

3

PRG (Post/Redirect/Get) qanday muammoni hal qiladi?

Savol 3

06

Dashboardlar

Amaliy KPI · rolga mos koʻrinish · agregat soʻrovlar · diagramma · drill-down

KPI | Agregatsiya | Chart.js | @can

KPI

OmborPro dashboardi: kim nimani ko‘radi

Har bir kartochka — bitta savolga javob va bosilganda filtrlangan ro‘yxatga o‘tadi.

1 Bugungi savdo

`invoiced` + `shipped` buyurtmalar summasi; kechagi kunga nisbatan % — menejer, admin

2 Tasdiqlash kutmoqda

`confirmed` holatidagi buyurtmalar soni → `/orders?status=confirmed` — menejer

3 Debitorlik

`unpaid` va `partial` hisob-fakturalar: `total - paid_amount` — buxgalter

4 Kam qoldiq

`quantity - reserved < min_stock` → `/products?low=1` — omborchi

5 7 kunlik savdo

chiziqli diagramma: kunlar bo‘yicha summa

REPORTSERVICE

KPI lar: SQL agregatlari va qisqa kesh

```php
public function kpi(): array
{
    return Cache::remember('dashboard:kpi', now()->addMinutes(5),
        fn () => [
            'sales_today' => Order::whereDate('created_at', today())
                ->whereIn('status', ['invoiced', 'shipped'])
                ->sum('total'),
            'pending' => Order::where('status', 'confirmed')->count(),
            'debt' => Invoice::whereIn('status', ['unpaid', 'partial'])
                ->sum(DB::raw('total - paid_amount')),
            'low_stock' => Stock::join('products', 'products.id',
                    '=', 'stocks.product_id')
                ->whereRaw('quantity - reserved < products.min_stock')
                ->count(),
        ]);
}
```

▶ NATIJA

```
['sales_today' => '48750000.00', 'pending' => 7,
 'debt' => '126400000.00', 'low_stock' => 12]
4 ta agregat so‘rov; keyingi 5 daqiqa — keshdan
```

`whereDate(..., today())`

“bugun” — `Asia/Tashkent` mintaqasida

`sum(DB::raw(...))`

qarz qoldig‘i ifoda bo‘yicha yig‘iladi

`whereRaw('quantity - reserved < …')`

mavjud qoldiq = quantity − reserved

`Cache::remember`

5 daqiqalik kesh — dashboard bazani “qiynamaydi”

BLADE + CHART.JS

Rolga mos kartochka va 7 kunlik diagramma

```blade
<div class="row g-3">
    @can('invoices.view')
        <x-kpi-card title="Debitorlik" :value="som($kpi['debt'])"
            :href="route('invoices.index', ['status' => 'unpaid'])" />
    @endcan
</div>
<canvas id="sales7" height="120"></canvas>
<script src="https://cdn.jsdelivr.net/npm/chart.js@4.4.1"></script>
<script>
    new Chart(document.getElementById('sales7'), {
        type: 'line',
        data: {
            labels: @json($series->keys()),
            datasets: [{ label: 'Savdo, so‘m',
                data: @json($series->values()) }],
        },
    });
</script>
```

▶ NATIJA

Buxgalter: “Debitorlik 126 400 000 so‘m” kartochkasi + diagramma
Omborchi: kartochka ko‘rinmaydi (invoices.view yo‘q)

@can('invoices.view')

ko‘rinishni yashiradi; sahifa ham serverda tekshiriladi

<x-kpi-card>

anonim Blade komponenti: `resources/views/components/kpi-card.blade.php`

@json(...)

PHP qiymatini xavfsiz JSON ga aylantirib JS ga beradi

:href

drill-down — kartochkadan filtrlangan ro‘yxatga

TUSHUNISHNI TEKSHIRISH

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

6-bo‘lim yakuni: Dashboardlar

1

“Harakatga undaydigan” KPI ga OmborPro dan bitta misol keltiring.

Savol 1

2

Nima uchun KPI ni Blade ichida `Order::all()` bilan hisoblash noto‘g‘ri?

Savol 2

3

`@can` bilan kartochkani yashirish xavfsizlik uchun yetarlimi?

Savol 3

XULOSA

Asosiy xulosalar

1 Interfeys — ish joyi

Back-office da SEO emas, ma’lumot zichligi, tezlik va xatosiz kiritish muhim; OmborPro — Blade + Livewire 3.

2 Holat URL da

Filtr, saralash va sahifa query string da: ulashish, “Orqaga” va yangilash ishlaydi.

3 Saralash — faqat oq ro‘yxat

Ustun nomi binding bo‘lmaydi; `in_array(..., true)` va `->orderBy('id')` bilan barqaror tartib.

4 To‘g‘ri sahifalash

`paginate` — ma’lumotnoma, `cursorPaginate` — katta jurnal; `withQueryString()` va filtrda `resetPage()`.

5 Validatsiya serverda

FormRequest, `old()`, `@error`, PRG; Livewire da `wire:model.blur` va massiv qoidalari.

6 Dashboard — harakat uchun

SQL agregatlari servisda, qisqa kesh, rolga mos kartochkalar va drill-down.

TAKRORLASH

Yakunlashdan oldin sakkizta savol

1. SSR va SPA ning asosiy farqi?
2. Livewire 3 da `wire:model` modifikatorsiz qachon yuboriladi?
3. `?sort=password` ga to‘g‘ri munosabat?
4. `simplePaginate` nima uchun tezroq?
5. `withQueryString()` nima qiladi?
6. Tahrirda SKU unikal qoidasi?
7. `{!! $p->name !!}` xavfi?
8. KPI qayerda hisoblanadi?

AMALIY MASHG‘ULOT

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

4 **Filtrlarni saqlovchi query string**
Oson · 10 ball

5 **Filtrdan xavfsiz WHERE**
O‘rta · 20 ball

6 **Mahsulot formasini validatsiya qilish**
O‘rta · 20 ball

7 **Dashboard ko‘rsatkichlari**
Qiyin · 30 ball

8 **Ma’lumotlar jadvali: filtr + saralash + sahifa**
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

Buyurtmalar ro‘yxati uchun qaysi filtrlar va qaysi sahifalash usulini tanlaysiz? Nima uchun?

2

`?sort=` parametri bilan bog‘liq ikki xavfni va ularning yechimini yozing.

3

Bugungi mavzudan loyihangizdagi qaysi ekranni birinchi bo‘lib qayta ishlaysiz?