3

UCHINCHI AMALIY MASHG‘ULOT

Korxona ilovalarini dasturlash · 2 soat · 3-mavzu

Domen modellari, migratsiyalar va admin panel

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

Ma’lumotnoma | Hujjat | Normalizatsiya | Eloquent | Migratsiya

MASHG‘ULOT MAQSADI

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

1. **Yaratish** — OmborPro ning asosiy jadvallari uchun migratsiyalar: tashqi kalitlar, noyob indekslar, soft delete
2. **Yozish** — Eloquent modellari: `$fillable`, `casts()`, bog‘lanishlar, `OrderStatus` enum cast
3. **To‘ldirish** — fabrikalar va seederlar orqali boshlang‘ich va test ma’lumotlarini
4. **Sozlash** — ma’lumotnomalar uchun admin panel (Filament 3)
5. **Tekshirish** — `migrate:fresh --seed` bilan sxema va ma’lumotlar to‘liq qayta yaratilishini

Qanday bajariladi va qanday baholanadi?

Ish tartibi

1. Saytga login va parolingiz bilan kiring: **emro.uz/korxona/kirish.html**
2. **3-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

Domen modellari va migratsiyalar

Yaratish tartibi · make:model -mfs · tashqi kalitlar · bog‘lanishlar

Migratsiya | Model | Tartib

TARTIB

Migratsiyalar ketma-ketligi va asosiy cheklovlar

#	Jadval	Tashqi kalitlar	Cheklovlar
1	`categories`	`parent_id` → categories (nullable)	—
2	`products`	`category_id` → categories (restrict)	`sku` unique, soft delete
3	`warehouses`, `counterparties`	—	`code` unique; `inn` char(9), soft delete
4	`stocks`, `stock_movements`	warehouse, product, user	unique(warehouse_id, product_id)
5	`orders`, `order_items`	counterparty, warehouse, created_by; order (cascade)	`number` unique
6	`invoices`, `payments`	order, counterparty; invoice	`number` unique, `external_id` unique nullable
7	`exchange_rates`, `audit_logs`, `import_batches`	user (nullable)	unique(date, currency)

2-QADAM

Generatorlar ketma-ketligi

```
php artisan make:model Category -mfs
php artisan make:model Product -mfs
php artisan make:model Warehouse -mfs
php artisan make:model Counterparty -mfs
php artisan make:model Stock -m
php artisan make:model StockMovement -m
php artisan make:model Order -mf
php artisan make:model OrderItem -mf
php artisan make:model Invoice -m
php artisan make:model Payment -m
php artisan make:migration add_is_active_to_users_table --table=users
```

▶ NATIJA

```
INFO  Model [app/Models/Category.php] created successfully.
INFO  Migration
[database/migrations/2026_09_21_100000_create_categories_table.php
] created successfully.
INFO  Factory [database/factories/CategoryFactory.php] created
successfully.
INFO  Seeder [database/seeders/CategorySeeder.php] created
successfully.
```

-mfs

ma’lumotnomalar: migratsiya + fabrika + seeder

-m

registrlar: ma’lumot servis orqali yoziladi

--table=users

mavjud jadvalni o‘zgartirish

MIGRATSIYA

orders va order_items

```php
Schema::create('orders', function (Blueprint $table) {
    $table->id();
    $table->string('number', 20)->unique();     // BY-2026-000123
    $table->foreignId('counterparty_id')->constrained();
    $table->foreignId('warehouse_id')->constrained();
    $table->string('status', 16)->default('draft')->index();
    $table->decimal('total', 14, 2)->default(0);
    $table->foreignId('created_by')->constrained('users');
    $table->timestamps();
});
Schema::create('order_items', function (Blueprint $table) {
    $table->id();
    $table->foreignId('order_id')->constrained()->cascadeOnDelete();
    $table->foreignId('product_id')->constrained();
    $table->integer('quantity');
    $table->decimal('price', 14, 2);
    $table->decimal('total', 14, 2);
});
```

▶ NATIJA

```
orders.created_by → users.id
order_items.order_id → orders.id ON DELETE CASCADE
```

`constrained('users')`

ustun nomi konventsiyaga mos emas — jadvalni aniq ko‘rsatamiz

`->index()`

holat bo‘yicha filtrlash tez bo‘lsin

`cascadeOnDelete()`

qoralama buyurtma o‘chirilsa — qatorlari ham

`currency`

slaydda qisqartirilgan: `char('currency', 3)->default('UZS')` — ISO 4217

MODEL

Stock — bog‘lanishlar va accessor

```php
class Stock extends Model
{
    public $timestamps = false;
    protected $fillable = ['warehouse_id', 'product_id',
        'quantity', 'reserved'];

    public function product(): BelongsTo
    {
        return $this->belongsTo(Product::class)->withTrashed();
    }
    protected function available(): Attribute
    {
        return Attribute::make(
            get: fn () => $this->quantity - $this->reserved,
        );
    }
}
```

▶ NATIJA

```
$stock->quantity  → 120
$stock->reserved  → 15
$stock->available → 105
```

`$timestamps = false`

jadvalda `created_at`/`updated_at` yo‘q

`withTrashed()`

o‘chirilgan mahsulot qoldig‘i ham ko‘rinsin

`Attribute::make(get:)`

hisoblanadigan `available` atributi

02

Fabrikalar va seederlar

definition() · holatlar · DatabaseSeeder · boshlang‘ich va test ma’lumotlari

Factory

Seeder

Faker

4-QADAM

ProductFactory va holat

```php
class ProductFactory extends Factory
{
    public function definition(): array
    {
        return [
            'sku' => fake()->unique()->numerify('UN-###'),
            'name' => fake()->randomElement(['Un', 'Guruch', 'Shakar'])
                . ' ' . fake()->numberBetween(1, 50) . ' kg',
            'category_id' => Category::factory(),
            'unit' => fake()->randomElement(['dona', 'kg', 'quti']),
            'price' => fake()->numberBetween(1000, 500000),
        ];
    }
    public function inactive(): static
    {
        return $this->state(['is_active' => false]);
    }
}
```

▶ NATIJA

```
Product::factory()->count(3)->make()->pluck('sku')
= ["UN-417", "UN-052", "UN-938"]
Product::factory()->inactive()->create()->is_active  → false
```

`unique()->numerify`

`#` — tasodifiy raqam; `unique()` takrorlamaydi

`Category::factory()`

kategoriya berilmasa, avtomatik yaratiladi

`inactive()`

holat — testlarda o‘qilishi oson

5-QADAM

DatabaseSeeder

```php
public function run(): void
{
    $ombor = ['TSH-1' => 'Toshkent markaziy', 'SAM-1' => 'Samarqand'];
    foreach ($ombor as $code => $name) {
        Warehouse::updateOrCreate(['code' => $code], ['name' => $name]);
    }
    User::updateOrCreate(['email' => 'admin@omborpro.test'], [
        'name' => 'Administrator',
        'password' => Hash::make('password'),
    ]);
    if (app()->isLocal()) {
        $cats = Category::factory()->count(5)->create();
        Product::factory()->count(50)
            ->state(fn () => ['category_id' => $cats->random()->id])
            ->create();
        Counterparty::factory()->count(20)->create();
    }
}
```

▶ NATIJA

```
INFO  Seeding database.
Category::count()      → 5
Product::count()       → 50
Warehouse::pluck('code') → ["TSH-1", "SAM-1"]
```

`updateOrCreate`

boshlang‘ich ma’lumot: qayta ishga tushirilsa takrorlanmaydi

`isLocal()`

demo ma’lumot faqat lokal muhitda

`$cats->random()`

mahsulotlar mavjud 5 kategoriyaga taqsimlanadi

`'password'`

faqat lokal test paroli — productionda boshqa

6-QADAM

Bazani qayta yaratish va tekshirish

```
php artisan migrate:fresh --seed
php artisan db:seed --class=WarehouseSeeder
php artisan model:show Product
php artisan tinker --execute='dump(App\Models\Product::count());'
```

▶ NATIJA

```
INFO  Dropping all tables ..... DONE
INFO  Running migrations. ... 14 DONE
INFO  Seeding database.
App\Models\Product ... Table: products, Relations: category
BelongsTo
50
```

`migrate:fresh --seed`

noldan sxema + ma’lumot (faqat lokal)

`--class`

bitta seederni ishga tushirish

`model:show`

model ustunlari, turlari va bog‘lanishlari

03

Admin panel (Filament 3)

O‘rnatish · resurslar · forma va jadval · kirishni cheklash

Filament

Admin panel

CRUD

7–8-QADAM

Filament o‘rnatish va resurs yaratish

```
composer require filament/filament:"^3.2" -W
php artisan filament:install --panels
php artisan make:filament-user
php artisan make:filament-resource Category --generate
php artisan make:filament-resource Product --generate --soft-deletes
php artisan make:filament-resource Counterparty --generate \
    --soft-deletes
php artisan make:filament-resource Warehouse --generate
```

▶ NATIJA

```
What is the ID? › admin
INFO  Filament panels successfully installed.
INFO  Success! admin@omborpro.test may now log in at
http://localhost:8080/admin/login.
INFO  Filament resource [App\Filament\Resources\ProductResource]
created successfully.
```

`-W`

bog‘liq paketlarni ham yangilashga ruxsat

`--panels`

`AdminPanelProvider` va `/admin` paneli

`--generate`

jadval ustunlaridan forma va jadval

`--soft-deletes`

o‘chirilganlar filtri, tiklash va butunlay o‘chirish

FORMA

ProductResource::form()

```php
public static function form(Form $form): Form
{
    return $form->schema([
        TextInput::make('sku')->required()->maxLength(32)
            ->unique(ignoreRecord: true),
        TextInput::make('name')->required()->maxLength(255),
        Select::make('category_id')
            ->relationship('category', 'name')->required(),
        Select::make('unit')->required()->options([
            'dona' => 'dona', 'kg' => 'kg',
            'litr' => 'litr', 'quti' => 'quti',
        ]),
        TextInput::make('price')->numeric()->minValue(0)->required(),
        TextInput::make('min_stock')->integer()->default(0),
        Toggle::make('is_active')->default(true),
    ]);
}
```

▶ NATIJA

```
/admin/products/create
  SKU [UN-001]  Nomi [...]  Kategoriya [▼]  Birlik [▼]
  Narx [15000]  Min. qoldiq [0]  Faol [✓]   [Saqlash]
```

`unique(ignoreRecord: true)`

tahrirlashda yozuvning o‘zi hisobga olinmaydi

`relationship('category', 'name')`

kategoriyalar ro‘yxati modeldagi bog‘lanishdan

`numeric()->minValue(0)`

manfiy narx kiritib bo‘lmaydi

ProductResource::table() va canAccessPanel

```php
public static function table(Table $table): Table
{
    return $table
        ->columns([
            TextColumn::make('sku')->searchable()->sortable(),
            TextColumn::make('name')->searchable(),
            TextColumn::make('category.name'),
            TextColumn::make('price')->money('UZS')->sortable(),
            IconColumn::make('is_active')->boolean(),
        ])
        ->filters([TrashedFilter::make()]);
}

// app/Models/User.php  (implements FilamentUser)
public function canAccessPanel(Panel $panel): bool
{
    return $this->is_active && $this->email === 'admin@omborpro.test';
}
```

▶ NATIJA

SKU ▲	Nomi	Kategoriya	Narx	Faol
UN-001	Un 1-nav 50 kg	Un mahsulotlari	UZS 15,000.00	✓
SH-002	Shakar 1 kg	Shirinliklar	UZS 9,000.00	✓

`category.name`

bog‘lanish ustuni — nuqta bilan

`money('UZS')`

pul formatida ko‘rsatish

TrashedFilter

soft delete qilinganlarni ko‘rsatish / yashirish

`canAccessPanel`

kim kira olishini model hal qiladi; 5-mavzuda — `admin` roli

LOYIHA BOSQICHI

Natija va topshirish

Natija: OmborPro ning asosiy jadvallari migratsiyalar bilan yaratilgan, Eloquent modellari bog‘lanishlar va turlar bilan yozilgan, fabrikalar va seederlar boshlang‘ich hamda demo ma’lumotlarni yaratadi, ma’lumotnomalar admin panel orqali boshqariladi.

Baholash mezonlari

- ✓ `php artisan migrate:fresh --seed` toza bazada xatosiz bajariladi
- ✓ Migratsiyalarda pul `decimal(14,2)`, STIR `char(9)`, `sku`/`number` unique, `stocks` da unique(warehouse_id, product_id)
- ✓ `products` va `counterparties` soft delete qilinadi; hujjat jadvallarida soft delete yo‘q
- ✓ Modellarda `$fillable`, `casts()` (jumladan `OrderStatus::class`) va kamida 8 ta bog‘lanish metodi bor
- ✓ Fabrikalar `unique()` bilan, seederlar boshlang‘ich ma’lumotni `updateOrCreate` bilan yaratadi; demo ma’lumot faqat lokal
- ✓ Admin panelda kategoriyalar, mahsulotlar, omborlar va kontragentlar CRUD i ishlaydi, o‘chirilganlarni tiklash mumkin
- ✓ Panelga kirish `canAccessPanel` bilan cheklangan
- ✓ O‘zgarishlar Pull Request orqali, migratsiya tartibi PR tavsifida yozilgan

Telefon kamerasi bilan skanerlang

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

📤 Pull Request havolasini topshiring. PR tavsifiga `php artisan migrate:fresh --seed` chiqishi, `php artisan model:show Order` natijasi va admin paneldagi mahsulotlar ro‘yxati skrinshotini qo‘shing.

AMALIY TOPSHIRIQ

Kompyuterda mustaqil · Oson · 10 ball

4-topshiriq: Model nomidan jadval nomi

Eloquent konventsiyasini soddalashtirilgan holda takrorlang: model nomi (`StockMovement`) snake_case ga o‘tkaziladi (`stock_movement`) va oxirgi so‘z ko‘plik shakliga keltiriladi. Ko‘plik qoidalari: undosh + `y` bilan tugasa — `y → ies` (`category → categories`); `s`, `x`, `z`, `ch`, `sh` bilan tugasa — `+es`; qolganlari — `+s`. Model nomi `^[A-Z][A-Za-z0-9]*$` ga mos bo‘lmasa — `InvalidArgumentException`.

Imzo: `function jadval_nomi(string $model): string`

Qaytaradi: Jadval nomi satri, masalan `'stock_movements'`.

MISOL · CHAQIRUV

```
jadval_nomi('StockMovement')
```

MISOL · NATIJA

```
"stock_movements"
```

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: Kategoriyalar daraxti yo‘llari

Kategoriyalar `['id' => int, 'name' => string, 'parent_id' => ?int]` ro‘yxati sifatida berilgan (`parent_id` — o‘ziga bog‘lanish). Seeder va admin panel uchun har bir kategoriyaning to‘liq yo‘lini yasang: ildizdan boshlab nomlar `/` bilan. Takroriy `id` yoki mavjud bo‘lmagan `parent_id` — `InvalidArgumentException`; aylana (kategoriya o‘zining avlodiga bog‘langan) — `DomainException`.

Imzo: `function kategoriya_yollari(array $kategoriyalar): array`

Qaytaradi: `[id => 'Ildiz / ... / Nom', ...]` — `id` bo‘yicha o‘sish tartibida.

MISOL · CHAQIRUV

```
kategoriya_yollari([['id' => 1, 'name' => 'Oziq-ovqat', 'parent_id' => null], ['id' => 2, 'name' => 'Un mahsulotlari', 'parent_id' => 1], ['id' => 3, 'name' => 'Un 1-nav', 'parent_id' => 2]])
```

MISOL · NATIJA

```
{"1":"Oziq-ovqat","2":"Oziq-ovqat / Un mahsulotlari","3":"Oziq-ovqat / Un mahsulotlari / Un 1-nav"}
```

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: Ommaviy to‘ldirish ($fillable / $guarded)

Eloquent ning `fill()` qoidasini sodda ko‘rinishda takrorlang. Har bir kirish kaliti uchun: `$fillable` da bo‘lsa — olinadi; aks holda `$guarded` da bo‘lsa yoki `$guarded === ['*']` bo‘lsa — tashlanadi; aks holda `$fillable` bo‘sh bo‘lsa va kalit `_` bilan boshlanmasa hamda `.` saqlamasa — olinadi; qolgan holatda tashlanadi. Model **to‘liq himoyalangan** (`$fillable` bo‘sh va `$guarded === ['*']`) bo‘lsa va kirish bo‘sh bo‘lmasa — `DomainException` (Laravel dagi `MassAssignmentException` kabi).

Imzo: `function ommaviy_toldirish(array $fillable, array $guarded, array $input): array`

Qaytaradi: Yoziladigan atributlar massivi — kirishdagi tartibda.

MISOL · CHAQIRUV

```
ommaviy_toldirish(['sku', 'name', 'price'], ['*'], ['sku' => 'UN-001', 'is_active' => false, 'price' => 100])
```

MISOL · NATIJA

```
{"sku":"UN-001","price":100}
```

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: Migratsiya rejasi: migrate, rollback, status

Migratsiya fayllari nomlari (`2026_09_14_100000_create_products_table`) va `migrations` jadvali (`['nom' => batch]`) berilgan. Buyruqni simulyatsiya qiling: `migrate` — bajarilmagan fayllar nom bo‘yicha tartibda va yangi batch raqami (bajariladigan narsa bo‘lmasa batch `null`); `rollback` — `$qadam = 0` bo‘lsa oxirgi batch dagi migratsiyalar, `$qadam > 0` bo‘lsa oxirgi `$qadam` ta migratsiya, bekor qilish tartibida (batch, so‘ng nom bo‘yicha kamayish); `status` — har bir fayl uchun `Ran [batch]` yoki `Pending`. Fayl nomi formati noto‘g‘ri yoki noma’lum buyruq — `InvalidArgumentException`; jadvalda bor, lekin fayli yo‘q migratsiya — `DomainException`.

Imzo: `function migratsiya_reja(array $fayllar, array $bajarilgan, string $buyruq, int $qadam = 0): array`

Qaytaradi: `migrate`: `['run' => list, 'batch' => ?int]`; `rollback`: `['revert' => list]`; `status`: `[nom => 'Ran [n]' | 'Pending']` nom bo‘yicha tartibda.

MISOL · CHAQIRUV

```
migratsiya_reja(['2026_09_14_100100_products',
'2026_09_14_100000_categories', '2026_09_20_090000_barcode'],
['2026_09_14_100000_categories' => 1,
'2026_09_14_100100_products' => 1], 'migrate')
```

MISOL · NATIJA

```
{"run":["2026_09_20_090000_barcode"],"batch":2}
```

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

8-topshiriq: Qoldiq registrini jurnaldan tiklash

`stock_movements` jurnali berilgan: har bir yozuv `[ombor, sku, tur, miqdor]`. Qoldiq registrini (`stocks`) tartib bilan qayta hisoblang: `in` — `quantity += q`; `out` (rezervdan jo‘natish) — `quantity -= q`, `reserved -= q`; `reserve` — `reserved += q`; `release` — `reserved -= q`; `adjust` — `quantity += q` (`q` manfiy bo‘lishi mumkin, lekin 0 emas). Boshqa turlarda `q > 0`. Invariantlar: `reserve` faqat `mavjud ≥ q` bo‘lsa; `out` va `release` faqat `reserved ≥ q` bo‘lsa; `adjust` dan keyin `quantity ≥ reserved`. Buzilsa — `DomainException`; noma’lum tur yoki noto‘g‘ri miqdor — `InvalidArgumentException`.

Imzo: `function qoldiq_registri(array $harakatlar): array`

Qaytaradi: `['OMBOR:SKU' => ['quantity' => int, 'reserved' => int, 'available' => int], ...]` — kalit bo‘yicha alifbo tartibida.

MISOL · CHAQIRUV

```
qoldiq_registri([['TSH-1', 'UN-001', 'in', 100], ['TSH-1', 'UN-001', 'reserve', 30], ['TSH-1', 'UN-001', 'out', 20]])
```

MISOL · NATIJA

```
{"TSH-1:UN-001":{"quantity":80,"reserved":10,"available":70}}
```

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

3-amaliy: 8 ta PHP topshiriq

1 **Keyingi hujjat raqami**
O‘rta · 20 ball · ma’ruzada

2 **Excel jadvalini normallashtirish**
O‘rta · 20 ball · ma’ruzada

3 **Audit uchun o‘zgarishlar farqi**
Oson · 10 ball · ma’ruzada

4 **Model nomidan jadval nomi**
Oson · 10 ball

5 **Kategoriyalar daraxti yo‘llari**
O‘rta · 20 ball

6 **Ommaviy to‘ldirish ($fillable / $guarded)**
O‘rta · 20 ball

7 **Migratsiya rejasi: migrate, rollback, status**
Qiyin · 30 ball

8 **Qoldiq registrini jurnaldan tiklash**
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.