

Laravel — pierwsza aplikacja

Czas: 2 × 45 min

praca indywidualna

Cel: Utworzenie pierwszej aplikacji w frameworku Laravel: trasy, kontroler, szablony Blade z układem, migracja, model Eloquent i lista rekordów z bazy.

Po zajęciach potrafisz:

- tworzy projekt Laravel i uruchamia serwer deweloperski
- definiuje trasy z nazwami i parametrami, w tym route model binding
- buduje widoki Blade dziedziczące wspólny układ (`@extends`, `@section`, `@yield`)
- tworzy tabelę migracją, wypełnia ją seederem i pobiera dane modelem Eloquent
- posługuje się poleceniami `php artisan` i rozpoznaje strukturę katalogów projektu

Sprzęt i materiały

- komputer z Windows 11, dostęp do internetu (pobranie pakietów Composera)
- PHP ≥ 8.3 z Composerem (Laravel Herd lub php.new) albo XAMPP (PHP 8.2 → Laravel 12)

Oprogramowanie

- **Laravel Herd dla Windows (zalecane, bezpłatny)** — zawiera PHP, Composer i instalator Laravel
herd.laravel.com/windows
- **Dokumentacja Laravel — instalacja** — alternatywa: skrypt `php.new` (PowerShell jako administrator)
laravel.com/docs/13.x/installation
- **Composer** — przy pracy na PHP z XAMPP
getcomposer.org/download
- **Visual Studio Code** — rozszerzenie Laravel (oficjalne) podświetla Blade
code.visualstudio.com

Przygotowanie przed zajęciami

- Nauczyciel instaluje wcześniej Herd lub PHP 8.3+ z Composerem. Sprawdź: `php -v`, `composer -V`, `laravel --version`.

Bezpieczeństwo

- Plik `.env` zawiera klucz aplikacji `APP_KEY` i dane bazy — nie udostępniaj go i nie wysyłaj do repozytorium (jest w `.gitignore`).
- Tryb `APP_DEBUG=true` jest tylko do pracy lokalnej; na serwerze produkcyjnym musi być `false`.

Przebieg zadania

1 Nowy projekt

Z Herd: `cd ~\Herd`, potem `laravel new filmoteka` (instalator pyta m.in. o zestaw startowy — wybierz **None**, o bazę — wybierz **SQLite**; pytania mogą się różnić między wersjami). Bez instalatora: `composer create-project laravel/laravel filmoteka` — przy PHP 8.2 z XAMPP Composer zainstaluje Laravel 12; kod zadania działa w obu wersjach.

1. Utwórz projekt i uruchom go: Herd — `http://filmoteka.test`; w innym przypadku `php artisan serve` i `http://127.0.0.1:8000`. Zanotuj wersję: `php artisan --version`.
2. Opisz w sprawozdaniu rolę katalogów i plików: `app/Models`, `app/Http/Controllers`, `routes/web.php`, `resources/views`, `database/migrations`, `public`, `.env`, `vendor`. Obejrzyj listę tras: `php artisan route:list`.

2 Trasy i widoki Blade

Szablony zapisuj w `resources/views` z rozszerzeniem `.blade.php`. `{{ $x }}` wypisuje wartość z automatycznym escapowaniem.

1. Utwórz układ `resources/views/layouts/app.blade.php`: nagłówek „Filmoteka”, menu (Start, Filmy, O nas — adresy przez `route('nazwa')`, aktywna pozycja wyróżniona dyrektywą `@class` i `request()->routeIs(...)`, `@yield('content')`, stopka z wersją Laravel (`app()->version()`). Styl: `public/css/style.css` dołączony przez `asset('css/style.css')`.
2. W `routes/web.php` zamień trasę `/` tak, by zwracała widok `start` z przekazaną zmienną `dzisiaj` (bieżąca data `now()->format('d.m.Y')`); nadaj trasie nazwę `start`. Dodaj trasę `/o-nas` (nazwa `o-nas`) przekazującą tablicę 3 autorów, wypisaną w widoku pętlą `@foreach`.
3. Oba widoki dziedziczą układ (`@extends('layouts.app')`, `@section('title', ...)`, `@section('content') ... @endsection`). Sprawdź, co pokaże strona, gdy jeden z autorów będzie miał w nazwie znaki `<b>`.

3 Migracja, model i dane

1. Wykonaj `php artisan make:model Film -m`. Zmień nazwę tabeli w migracji na `filmy` i dodaj kolumny: `tytuł` (string 120), `rezyser` (string 80), `rok` (unsignedSmallInteger), `gatunek` (string 20), `ocena` (decimal 3,1, nullable) oraz `timestamps()`. W modelu ustaw `protected $table = 'filmy'`, `$fillable` i stałą `GATUNKI`.
2. Uruchom `php artisan migrate` i obejrzyj tabelę (SQLite: rozszerzenie bazy w VS Code lub `php artisan db:table filmy`). Dlaczego bez `$table` Laravel szukałby tabeli `films`?
3. Utwórz seeder `php artisan make:seeder FilmSeeder` dodający 8 filmów przez `Film::create([...])` (np. z zadania 17), wywołaj go w `DatabaseSeeder` i wykonaj `php artisan db:seed`. Sprawdź liczbę rekordów w `php artisan tinker: App\Models\Film::count()`.

4 Kontroler i lista z bazy

1. Wykonaj `php artisan make:controller FilmController`. Metoda `index(Request $request)` pobiera filmy posortowane po tytule, opcjonalnie filtrowane gatunkiem z `?gatunek=` (`when(...)`), i przekazuje je do widoku `filmy.index` razem z listą gatunków. Metoda `show(Film $film)` zwraca widok `filmy.show`.
2. Dodaj trasy `GET /filmy` (nazwa `filmy.index`) i `GET /filmy/{film}` (nazwa `filmy.show`). Sprawdź adres nieistniejącego filmu, np. `/filmy/999` — jaki kod zwraca Laravel i dlaczego?
3. Widok `filmy/index.blade.php`: formularz GET z listą gatunków (`@selected`), liczba filmów, tabela z tytułem jako odnośnikiem `route('filmy.show', $film)`, `@if ($filmy->isEmpty())` z komunikatem. Widok `filmy/show.blade.php`: szczegóły i data dodania `created_at->format('d.m.Y H:i')`.
4. Zrób zrzuty: start, o nas, lista, lista z filtrem, szczegóły, wynik `route:list`. Spakuj projekt **bez** `vendor/` i `node_modules/` jako `nazwisko_imie_zad20.zip` — będzie potrzebny w zadaniu 21.

W sprawozdaniu umieść

- ☐ tabela wykonanych poleceń (krok.punkt) z poleceniami artisan i fragmentami kodu
- ☐ opis katalogów projektu i wersja Laravel/PHP
- ☐ zrzuty: strony aplikacji, `route:list`, wynik `Film::count()`
- ☐ archiwum `nazwisko_imie_zad20.zip` i odpowiedzi na pytania kontrolne

Pytania kontrolne

1. Co robi migracja i dlaczego jest lepsza od ręcznego tworzenia tabeli w phpMyAdmin przy pracy zespołowej?
2. Czym jest route model binding i ile linii kodu z zadania 17 zastępuje?
3. Dlaczego `{{ $autor }}` jest bezpieczne, a `{!! $autor !!}` wymaga ostrożności?
4. Do czego służą nazwy tras (`->name(...)`) i funkcja `route()` ?

Kryteria oceny

Kryterium	Punkty
Projekt uruchomiony, opis struktury katalogów, <code>route:list</code>	3
Układ Blade z menu (aktywna pozycja) i 2 widoki dziedziczące z przekazanymi danymi	4
Migracja tabeli <code>filmy</code> , model z <code>\$table</code> i <code>\$fillable</code> , seeder z 8 rekordami	5
Kontroler: lista z filtrem gatunku i szczegóły (route model binding, 404)	5
Zrzuty, archiwum, odpowiedzi na pytania	3
Razem	20

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).