Faysal Aziz

Laravel Evolution

From Foundation to Advanced Intelligent Production in 5 days

FA-0283 · 5 days

DevOps, Cloud & Infrastructure · Data & Analytics

Introduction

This five-day program begins by reinforcing key Laravel fundamentals—essential for ensuring solid foundations—then progresses into production-grade deployment practices on Linux, significant enhancements in Laravel 11, and advanced topics like RESTful APIs, JWT authentication, event-driven architecture, and building a recommendation engine using machine learning. The instructor's 30+ years of experience promise immediate, practical value—no unnecessary theory.

Delivered by a seasoned expert with **30+ years** of industry experience. The program emphasizes outcome-driven methods, production-ready practices, and real-world solutions over academic theory.

Learning outcomes

Participants will be able to:

- Build a Laravel application from ground up, quickly reinforcing core features.
- Deploy Laravel securely and optimally on Linux environments.
- Understand major changes and leverage new capabilities introduced in Laravel 11.
- Construct robust RESTful APIs with JWT-based authentication.
- Employ events, queues, and real-time features using Laravel Reverb.
- Integrate ML to create a recommendation engine.
- Apply patterns and operational best practices for scalable production systems.

Prerequisites

- Basic familiarity with Laravel (version 10+).
- General understanding of Linux, Git, and Composer.
- Some experience with databases and PHP (8.x).
- Eagerness to extend Laravel into production and advanced backend systems.

Training outline

1. 1. Refreshing Laravel Fundamentals

- Installing Laravel via Composer; environment setup with .env.
- MVC architecture, routing, controllers, views, Blade templating.
- Basic Eloquent ORM: relations, querying, migration/seeding.
- Form requests and validation; intro to middleware.

2. 2. Linux Production-Ready Deployment

- Server prerequisites: PHP 8.2+, required extensions, Composer, Git.
- Configure Nginx/Apache, SSL (Certbot), and document root security.
- File and folder permissions, SSH key setup, environment hardening.
- PHP-FPM tuning, supervisor for queue workers.
- CI/CD integration (GitHub Actions/Jenkins); using Laravel Forge or custom scripts.
- Containerization (Docker) and serverless options.

3. 3. Transitioning from Laravel 10 to 11 (and vice versa)

- Official release timeline, support schedule.
- Slimmer skeleton: consolidated routes, middleware, and exceptions.
- Optional API and broadcasting scaffolding.
- New defaults: Pest testing, SQLite, min PHP 8.2 requirement.
- New make: commands (enum, interface, class).
- Introduced features: health route (/up), rate limiting per-second, Dumpable trait, model casts() method, once() helper, automatic policy discovery.
- Laravel Reverb (WebSocket) overview.

4. 4. Advanced Application Architecture

- Event-driven design: events, listeners, dispatching, and testing.
- Queue usage: Redis/Beanstalkd, backoff, retries.
- Introduce Laravel Reverb for real-time features.
- Service-repository patterns; injecting dependencies; action classes; debugging using Dumpable.

5. 5. RESTful APIs & JWT Authentication

- Designing API controllers, resources, and resource collections.
- Implement JWT authentication using Tymon/JWT-Auth; manage tokens, refresh logic, and scopes.
- Compare Sanctum vs JWT; secure endpoints; validation and error handling.
- API versioning, throttling/rate limiting strategies.

6. 6. Machine Learning Integration: Recommendation Engine

- Overview of ML techniques: collaborative filtering, feature pipelines.
- Using PHP-ML libraries or external Python microservices.
- Data preparation and integration through Eloquent models.
- Build recommendation logic (similarity matrices, model training).
- Expose recommendation API, implement retraining schedule.

7. 7. Performance, Monitoring & Scaling

- Caching plan: config, route, view cache strategies.
- DB optimization: indexing, query profiling, eager loading limits.
- Application logging, metrics, and external monitoring tools (ELK, Honeybadger).
- Health route usage, uptime checks in Kubernetes or load-balanced environments.
- Queue scaling and horizontal scaling patterns.

8. 8. Migration Roadmap & Wrap-Up

- Discuss upgrade process 10 → 11; common pitfalls and dependency issues.
- Rollback strategies and deployment safety nets.
- Q\&A, recap, and next steps for participants' own projects.

© 2026 Faysal Aziz · Kuala Lumpur