สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

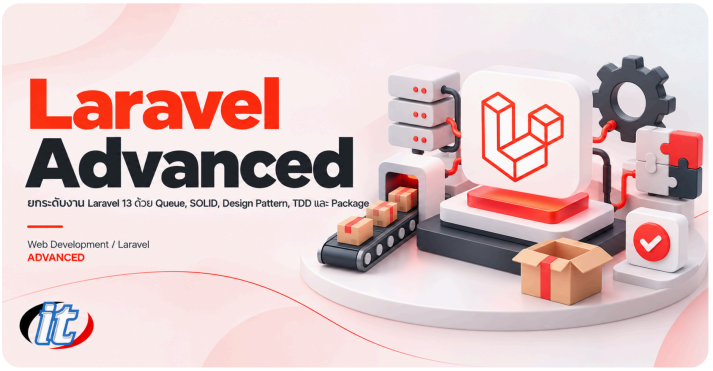
COURSE ID
WDC-01-36
WEB DEVELOPMENT / LARAVEL

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

Laravel Advanced

ยกระดับงาน Laravel 13 ด้วย Queue, SOLID, Design Pattern, TDD และ Package

- 30 ชั่วโมง
- 5 วัน
- พื้นฐาน ผ่าน Laravel Basic มาแล้ว
- ระดับ Advanced



ระยะเวลา: 5 วัน
30 ชั่วโมง (5 วัน วันละ 6 ชม.) · 09:00-16:00 น.

รูปแบบ: Classroom / Hybrid
Onsite หรือ Online ผ่าน Microsoft Teams · Public / In-house

รายละเอียดหลักสูตร
www.itgenius.co.th

เมื่อสร้างเว็บด้วย Laravel ได้แล้ว โจทย์ถัดไปคือทำให้ระบบรองรับงานจริงที่ใหญ่ขึ้น งานที่ใช้เวลานานต้องย้ายไปทำเบื้องหลังด้วย Queue โค้ดใน Controller ที่ยาวขึ้นเรื่อย ๆ ต้องแยกเป็นส่วนที่ทดสอบได้ และทีมต้องมี Test กับ Pipeline ที่ช่วยให้แก้ไขระบบได้โดยไม่กลัวพัง ความเข้าใจเรื่อง Service Container หลัก SOLID และ Design Pattern จึงเป็นสิ่งที่แยกนักพัฒนาที่ใช้ Laravel เป็น ออกจากผู้ที่ดูแลระบบ Laravel ขนาดใหญ่ได้อย่างมั่นใจ

หลักสูตรนี้ต่อยอดจาก Laravel พื้นฐานบน Laravel 13 และ PHP 8.5 เริ่มจากเบื้องหลังการทำงานของ Service Container, Routing และ Eloquent ขั้นสูง ระบบ Queue ด้วย Redis และ Horizon, Event, Notification และ Scheduler จากนั้นจัดโครงสร้างโค้ดด้วยหลัก SOLID, Repository และ Design Pattern ฝึกเขียน Test ด้วย Pest และ TDD ตั้งแต่ Unit Test ถึง Browser Test พัฒนา Package ของตัวเอง ทำ CI/CD ด้วย GitHub Actions แล้วปิดท้ายด้วย Capstone ที่รวมทุกเรื่องเข้าด้วยกัน (อบรม 5 วัน วันละ 6 ชั่วโมง รวม 30 ชั่วโมง ระดับ Advanced)

เครื่องมือ: Laravel 13 · PHP 8.5 · Redis และ Horizon · Pest · GitHub Actions

ราคาค่าอบรม · PUBLIC / IN-HOUSE

10,900 บาท / ท่าน

สมัครผ่านเว็บไซต์ ลด 10%

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide ซอร์สโค้ดครบทุก Lab ตัวอย่าง Package และ Workflow ของ GitHub Actions และ Certificate · รับจัด In-house ขอใบเสนอราคาได้

เพิ่มเพื่อน LINE

02-570-8449 · 088-807-9770 · @itgenius · www.itgenius.co.th

01 วัตถุประสงค์ Objectives

- เข้าใจการทำงานของ Service Container, Service Provider และ Facade และนำมาใช้ออกแบบโค้ดได้
- ใช้ Routing, Form Request และ Eloquent ขั้นสูง เช่น Polymorphic Relationship และ Custom Cast ได้
- ออกแบบงานเบื้องหลังด้วย Queue, Redis, Horizon และดูแล Worker บนเซิร์ฟเวอร์ด้วย Supervisor ได้
- ใช้ Event, Listener, Notification, Artisan Command และ Task Scheduling ได้เหมาะกับงาน
- จัดโครงสร้างโค้ดตามหลัก SOLID ด้วย Repository, Service, Action Class และ Design Pattern ได้
- เขียน Test ด้วย Pest ทั้ง Unit, Feature, Browser และ Architecture Test และทำงานแบบ TDD ได้
- แยกโค้ดเป็น Package ทดสอบ และนำไปใช้ซ้ำในหลายโปรเจกต์ได้
- ตั้ง CI/CD ด้วย GitHub Actions ให้ตรวจโค้ดและรัน Test ทุกครั้งที่ส่งงานได้

02 กลุ่มเป้าหมาย Target Audience

- นักพัฒนาเว็บที่ใช้ Laravel ในงานจริงและต้องการยกระดับไปอีกขั้น
- โปรแกรมเมอร์ที่ต้องดูแลระบบ Laravel ขนาดใหญ่และต้องการโครงสร้างที่ดูแลง่าย
- ฝ่ายไอทีและทีมพัฒนาระบบขององค์กรที่ต้องการวางมาตรฐานโค้ดและการทดสอบ
- Tech Lead ที่ต้องรีวิวโค้ดและวาง Pipeline ให้ทีม Laravel
- ผู้ที่ผ่านหลักสูตร Basic Laravel for Beginners มาแล้ว

03 จุดเด่นของหลักสูตร Highlights

- **อัปเดตเป็น Laravel 13** บน PHP 8.5 พร้อม PHP Attributes ใหม่
- **Queue ระดับใช้งานจริง** Redis, Horizon และ Supervisor
- **โค้ดที่ทีมดูแลได้** SOLID, Repository และ Design Pattern
- **Test ครบทุกชั้น** Pest, TDD, Browser และ Architecture Test
- **Package และ CI/CD** ใช้โค้ดซ้ำและตรวจอัตโนมัติด้วย GitHub Actions

04 ความรู้พื้นฐาน Prerequisites

- ผ่านหลักสูตร Laravel พื้นฐาน หรือสร้างเว็บด้วย Laravel ได้ทั้ง Route, Controller, Blade และ Eloquent
- เข้าใจหลัก OOP ใน PHP เช่น Class, Interface และ Namespace
- ใช้ Git, Composer และ Command Line ได้คล่อง
- โน้ตบุ๊กที่ติดตั้ง PHP 8.3 ขึ้นไป, Composer, Node.js, Redis และ VS Code ได้ หรือใช้ Laravel Herd หรือ Sail พร้อมสิทธิ์ติดตั้งโปรแกรม

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook Windows, macOS หรือ Linux RAM 8 GB ขึ้นไป เชื่อมต่ออินเทอร์เน็ต พร้อมสิทธิ์ติดตั้งโปรแกรม
Framework	Laravel 13 บน PHP 8.5 (ใช้ 8.3 ขึ้นไปได้) ผ่าน Laravel Herd หรือ Sail
Queue และฐานข้อมูล	Redis, Laravel Horizon และ MySQL หรือ SQLite
ทดสอบและตรวจโค้ด	Pest พร้อม Browser Plugin, Laravel Pint และ Larastan
CI/CD	Git และ GitHub Actions ด้วยบัญชี GitHub ของผู้เรียน

หมายเหตุ: เครื่องมือทั้งหมดเป็นโอเพนซอร์สหรือมีแบบใช้ฟรี ผู้เรียนใช้บัญชี GitHub ของตนเองสำหรับ Socialite และ GitHub Actions หลักสูตรนี้ปรับปรุงใหม่ปี 2026 เปลี่ยนจาก Laravel 5 เป็น Laravel 13 ใช้ PHP Attributes สำหรับ Job และ Controller, Queue::route และ Pest แทน PHPUnit แบบเดิม เพิ่ม Browser Test ด้วย Pest Browser Plugin ควบคู่กับ Laravel Dusk, Architecture Test, Larastan และ CI ด้วย GitHub Actions ตัดหัวข้อสร้างแอปด้วย Vue.js ออกเพราะมีในหลักสูตร Laravel with Vue.js Workshop แล้ว หลักสูตรนี้ต่อยอดจาก Basic Laravel for Beginners และไม่ทวนพื้นฐาน

06 ภาพรวมการเรียน 5 วัน Big Picture

วัน	หัวข้อหลัก
Day 1-2 Section 1-8	เตรียมโปรเจกต์ระดับทีมด้วย Pint และ Larastan, Service Container และ Facade, Routing และ Controller ขั้นสูง, Eloquent ขั้นสูง, Queue ด้วย Redis, Horizon กับ Supervisor, Event และ Notification, Artisan Command และ Scheduler
Day 3 Section 9-12	หลัก SOLID ในโปรเจกต์ Laravel, Repository, Service และ Action Class, Design Pattern อย่าง Decorator, Pipeline และ Strategy, ค้นหาด้วย Scout และ Login ด้วย Socialite
Day 4-5 Section 13-20	เขียน Test ด้วย Pest, Mock และ Fake, ฝึก TDD และ Refactor, Browser Test และ Architecture Test, พัฒนา Package ของตัวเอง, CI/CD ด้วย GitHub Actions แล้วปิดท้ายด้วย Capstone

DAY 1

เบื้องหลัง Laravel 13 และ Eloquent ขั้นสูง

เนื้อหาการอบรม · Section 1-4

รวม
6 ชั่วโมง

1 Lab: เตรียมโปรเจกต์ Laravel 13 ระดับทีม

- ของใหม่ใน Laravel 13 และการ Upgrade จากรุ่นก่อน
- สภาพแวดล้อมด้วย Laravel Herd หรือ Sail (Docker)
- จัดรูปแบบโค้ดด้วย Pint และตรวจด้วย Larastan
- ปรับแต่ง Stub และสร้างคำสั่ง make ของตัวเอง
- Lab: โปรเจกต์ตั้งต้นที่ใช้ต่อทั้งหลักสูตร

2 Service Container, Provider และ Facade

- Dependency Injection และ Autowiring ใน Laravel
- Binding, Singleton และ Contextual Binding
- เมื่อไรควรใช้ app()→make() และเมื่อไรไม่ควร
- Service Provider และการสร้าง Facade ของตัวเอง
- Lab: สลับผู้ให้บริการส่ง SMS ด้วย Interface

3 Routing และ Controller ขั้นสูง

- Route Model Binding แบบกำหนด Key และ Scoped Binding
- Controller Attributes อย่าง #[Middleware] และ #[Authorize]
- Single Action Controller และการจัดระเบียบ Controller
- Form Request สำหรับ Validation และ Authorization ขั้นสูง
- Lab: Refactor Controller ที่ยาวเกินให้กระชับ

4 Lab: Eloquent ขั้นสูง

- Polymorphic Relationship สำหรับความคิดเห็นและรายการโปรด
- Value Object ร่วมกับ Eloquent ด้วย Custom Cast
- Query Scope และ Eager Loading ป้องกันปัญหา N+1
- Model Factory สำหรับข้อมูลทดสอบที่สมจริง
- Lab: ระบบรายการโปรด (Favorites)

DAY 2

Queue, Event และงานเบื้องหลัง

เนื้อหาการอบรม · Section 5-8

รวม
6 ชั่วโมง

5 Lab: Queue ด้วย Redis

- Job, Dispatch และการเลือก Connection
- กำหนดการทำงานด้วย #[Tries], #[Backoff] และ #[Timeout]
- จัดเส้นทาง Job ไปยังคิวด้วย Queue::route
- Job Chaining, Batching และ Unique Job
- Lab: ส่งการแจ้งเตือนจำนวนมากผ่าน Queue

6 Laravel Horizon และ Supervisor

- ติดตั้ง Horizon และอ่าน Dashboard
- กำหนดจำนวน Worker และการกระจายงานใน config/horizon.php
- จัดการ Failed Job การ Retry และการแจ้งเตือนเมื่อคิวค้าง
- ดูแลโปรเซสของ Horizon บนเซิร์ฟเวอร์ด้วย Supervisor
- Lab: ปรับคิวให้รองรับงานเร่งด่วนและงานหนัก

7 Event, Listener และ Notification

- Event และ Listener แบบทำงานทันทีและแบบเข้าคิว
- Domain Event สำหรับแยกผลข้างเคียงออกจาก Business Logic
- Observer เทียบกับ Event และข้อควรระวัง
- Notification หลายช่องทาง ทั้งอีเมลและฐานข้อมูล
- Lab: แจ้งเตือนเมื่อชำระเงินสำเร็จ

8 Lab: Artisan Command และ Task Scheduling

- สร้าง Artisan Command พร้อม Argument และ Option
- ตั้งเวลางานด้วย Scheduler และป้องกันการทำงานซ้อน
- งานตามรอบ เช่น สรุปยอดรายวันและล้างข้อมูลเก่า
- Lab: รายงานสรุปยอดขายส่งอีเมลทุกเช้า

DAY 3

SOLID, Design Pattern และบริการเสริม

เนื้อหาการอบรม · Section 9-12

รวม
6 ชั่วโมง

9 หลัก SOLID ในโปรเจกต์ Laravel

- SRP และ OCP กับโค้ดที่เติบโตขึ้นเรื่อย ๆ
- LSP และ ISP ผ่าน Interface ที่เล็กและชัดเจน
- DIP ร่วมกับ Service Container
- Object ขนาดเล็กช่วยให้ทดสอบและเปลี่ยนแปลงได้ง่าย
- Lab: Refactor โค้ดคำนวณราคาตามหลัก SOLID

10 Lab: Repository, Service และ Action Class

- Repository แบบเรียบง่ายและเมื่อไรไม่จำเป็นต้องใช้
- นำ Method ที่ใช้ร่วมกันใน Repository กลับมาใช้ซ้ำ
- Service และ Action Class สำหรับ Business Logic
- DTO สำหรับส่งข้อมูลระหว่างชั้นอย่างมี Type
- Lab: หน้าประวัติการชำระเงินแบบแยกชั้นชัดเจน

11 Design Pattern ที่ใช้บ่อยใน Laravel

- Decorator สำหรับเพิ่ม Cache ให้ Repository
- Chain of Responsibility ผ่าน Pipeline และ Middleware
- Strategy สำหรับสลับวิธีคำนวณหรือช่องทางชำระเงิน
- View Presenter สำหรับจัดรูปข้อมูลก่อนแสดงผล
- Lab: ระบบส่วนลดหลายเงื่อนไขด้วย Pipeline

12 Lab: ค้นหาด้วย Scout และ Login ด้วย Socialite

- ค้นหาข้อมูลด้วย Laravel Scout แบบ Database Engine
- ภาพรวม Engine ภายนอก เช่น Meilisearch และ Typesense
- Login ด้วยบัญชี GitHub ผ่าน Laravel Socialite
- Lab: หน้าค้นหาสินค้าและ Login ด้วย GitHub

DAY 4

Testing และ TDD

เนื้อหาการอบรม · Section 13-16

รวม
6 ชั่วโมง

13 Lab: พื้นฐานการทดสอบด้วย Pest

- Unit Test และ Feature Test ใน Laravel
- RefreshDatabase และ Model Factory
- Dataset และ Expectation ของ Pest
- Lab: Test ระบบรายการโปรดที่สร้างไว้

14 Mock, Fake และ Integration Test

- ทดสอบการส่งอีเมลด้วย Mail::fake()
- Queue::fake(), Event::fake() และ Notification::fake()
- จำลอง API ภายนอกด้วย Http::fake()
- การทำ Mock และข้อควรระวังเมื่อ Mock มากเกินไป
- Lab: Integration Test ของขั้นตอนชำระเงิน

15 Lab: TDD จากตัวอย่างจริง

- วงจร Red, Green และ Refactor
- เขียน Test ก่อนแล้วจึงสร้างฟีเจอร์ทีละขั้น
- Refactor โค้ด Laravel โดยมี Test รองรับ
- Lab: สร้างหน้าโปรไฟล์ผู้ใช้ด้วย TDD

16 Browser Test และ Architecture Test

- Browser Test ด้วย Pest Browser Plugin ที่ทำงานบน Playwright
- Laravel Dusk สำหรับโปรเจกต์ที่ใช้อยู่เดิม
- Architecture Test บังคับกติกาของโครงสร้างโค้ด
- Lab: Browser Test ขั้นตอนสมัครสมาชิกและ Login

DAY 5

Package, CI/CD และ Capstone

เนื้อหาการอบรม · Section 17-20

รวม
6 ชั่วโมง

17 Lab: พัฒนา Package ของตัวเอง

- แยกโค้ดที่ใช้ซ้ำออกมาเป็น Package
- โครงสร้าง Package, Service Provider และ Package Discovery
- Publish Config, Migration และ View ของ Package
- Lab: Package บันทึกประวัติการแก้ไขข้อมูล

18 แนวปฏิบัติที่ดีในการพัฒนา Package

- ทดสอบ Package ด้วย Orchestra Testbench
- Semantic Versioning และการดูแลความเข้ากันได้
- เผยแพร่บน Packagist หรือใช้เป็น Private Repository
- เขียน README และ Changelog ให้ทีมใช้งานต่อได้

19 Lab: CI/CD ด้วย GitHub Actions

- Workflow ที่รัน Pint, Larastan และ Pest ทุก Pull Request
- จัดการ Secret และไฟล์ .env ของแต่ละสภาพแวดล้อม
- ขั้นตอน Deploy และรีสตาร์ต Worker ด้วย horizon:terminate
- Lab: Pipeline ที่ไม่ให้ Merge เมื่อ Test ไม่ผ่าน

20 Workshop: Capstone ระบบสมาชิกและการชำระเงิน

- ออกแบบฟีเจอร์ด้วย Action Class, Event และ Queue
- เขียน Test แบบ TDD และ Browser Test ของขั้นตอนหลัก
- ให้ AI Assistant ช่วยรีวิวโค้ดแล้วตรวจทานเองทุกจุด
- นำเสนอผลงานและรีวิวการออกแบบร่วมกัน

07 เรียนจบแล้วได้อะไร Outcomes

- **เข้าใจเบื้องหลัง Laravel** Service Container และ Facade
- **งานเบื้องหลังที่เสถียร** Queue, Horizon และ Scheduler
- **โค้ดดูแลง่าย** SOLID และ Design Pattern
- **มั่นใจด้วย Test** Pest, TDD และ Browser Test
- **ใช้โค้ดซ้ำได้** Package ของทีมเอง
- **ตรวจอัตโนมัติ** CI/CD ด้วย GitHub Actions

08 สิ่งที่ผู้เรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ครบทั้ง 5 วัน และ Lab Guide ทีละขั้นตอน
- ซอร์สโค้ดครบทุก Lab พร้อม Test และโปรเจกต์ Capstone
- ตัวอย่าง Package และไฟล์ Workflow ของ GitHub Actions ที่นำไปปรับใช้ได้
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สามิตร โกยม ผู้สอนและนักพัฒนาที่ใช้ Laravel สร้างและดูแลระบบงานจริงมาหลายรุ่น เน้นสอนจากปัญหาที่ทีมเจอเมื่อระบบโตขึ้นแล้วค่อย ๆ Refactor ให้เห็นว่า Queue, SOLID, Design Pattern และ Test ช่วยอะไรได้จริง ผู้เรียนลงมือเขียนโค้ดและ Test ในทุกหัวข้อ เพื่อกลับไปยกระดับโปรเจกต์ Laravel ของตนเองได้ทันที

ราคาค่าอบรม 10,900 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

10,900 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

02-570-8449 · 088-807-9770

@itgenius · contact@itgenius.co.th

เพิ่มเพื่อน LINE