



RESTful API with Laravel 5.7

Representational state transfer หรือ REST คือ การสร้าง Webservice ชนิดหนึ่งที่ใช้สื่อสารกันบน Internet ใช้หลักการแบบ stateless คือไม่มี session ซึ่งต่างจาก webservice แบบอื่นเช่น WSDL และ SOAP การทำงานของ RESTful Webservice จะอาศัย URI/URL ของ request เพื่อค้นหาและประมวลผล แล้วตอบกลับไปในรูป XML, HTML, JSON โดย response ที่ตอบกลับจะเป็นการยืนยันผลของคำสั่งที่ส่งมา และสามารถพัฒนาด้วยภาษา programming ได้หลากหลาย คำสั่งก็จะมีตาม HTTP verbs ซึ่งก็คือ GET, POST, PUT, DELETE

ซึ่งการเขียนและออกแบบ Restful API ค่อนข้างมีรายละเอียดและซับซ้อนอยู่พอสมควร การเลือกใช้ Framework อย่าง Laravel จะช่วยเราได้มาก เพราะ Laravel เตรียมความพร้อมเรื่องการออกแบบและสร้าง API มาเป็นอย่างดี โดยเฉพาะใน Laravel 5.7 นี้จะมีความสามารถใหม่ ๆ ที่เกี่ยวข้องกับการทำ API มากมาย ดังนั้นผู้เข้าอบรมหลักสูตรนี้จะได้แนวทางที่ถูกต้อง พร้อมตัวอย่างการเขียน Restful API ในโปรเจกต์จริง ในการทำงานจริง ที่ผู้สอนมีประสบการณ์ตรงมาถ่ายทอดให้นำไปฝึกฝนและต่อยอดในงานของตนเองได้เป็นอย่างดี

วัตถุประสงค์:

- เรียนรู้การใช้งานคำสั่ง PHP Artisan ใน Laravel เพื่อสร้าง Code ส่วนของ Controller , Model และ View
- เรียนรู้การทำงานกับ Middleware ใน Laravel 5.7
- เรียนรู้การสร้าง Route แบบ Resource ใน Laravel สำหรับทำ API
- เรียนรู้การจัดการกับไฟล์ JSON สำหรับ RESTful API
- เรียนรู้การ Validation
- เรียนรู้การเขียน Restful API ทั้ง 4 process คือ CRUD (Creat-Read-Update-Delete)
- เรียนรู้การใช้งาน Sessions และ Access Token



- เรียนรู้การใช้งาน Laravel Passport เสริมความปลอดภัยให้กับ RESTful API แบบ OAuth 2
- เข้าใจแนวทางการแบ่งหน้าเนื้อหา
- เรียนรู้คำสั่งการ Sorting , Filtering และ Caching
- เรียนรู้การทำงานกับ CORS เมื่อต้องเรียกใช้งานผ่าน Javascripts
- พร้อมฝึกฝนทำ Workshop ในงานจริงอีกมากมาย

กลุ่มเป้าหมาย:

- นักเรียนนักศึกษา
- ครู อาจารย์ วิทยากรที่สนใจ
- นักวิชาการ นักไอที หรือผู้ดูแลระบบ
- ตลอดจนผู้สนใจทั่วไปที่สนใจออกแบบและพัฒนา RESTful API ด้วย Laravel 5.7

ความรู้พื้นฐาน:

- สามารถใช้งานระบบคอมพิวเตอร์และอินเทอร์เน็ตได้
- ความรู้พื้นฐานในการเขียน PHP & MySQL มาบ้าง
- มีความรู้พื้นฐานการเขียนและทำงานกับ Laravel framework มาบ้าง
- ความรู้พื้นฐานในการออกแบบเว็บไซต์มาบ้าง
- ความรู้พื้นฐานในการทำงานกับฐานข้อมูลมาบ้าง

ระยะเวลาในการอบรม:

- 24 ชั่วโมง

ราคาคอร์สอบรม:

- 8,000 บาท (ราคานี้ยังไม่ได้รวมภาษีมูลค่าเพิ่ม)



วิทยากรผู้สอน:

- อาจารย์สามมิตร โกยม

คอร์สที่ควรอบรมก่อนหน้า:

- Basic Laravel 5.7 สำหรับผู้เริ่มต้น

คอร์สต่อเนื่องที่แนะนำ:

- Laravel 5.7 Workshop สร้างร้านค้าออนไลน์ (E-Commerce Shop Website) ใน 3 วัน

เนื้อหาการอบรม:

Module 1: ดาวน์โหลดและติดตั้งเครื่องมือสำหรับการพัฒนา RESTful API ด้วย Laravel

- ติดตั้ง XAMPP สำหรับใช้งาน PHP & MySQL
- ดาวน์โหลดและติดตั้ง Composer
- ดาวน์โหลดและติดตั้ง Virtual Box
- ดาวน์โหลดและติดตั้ง Vagrant
- ดาวน์โหลดและติดตั้ง Visual Studio Code พร้อมส่วนเสริมที่จำเป็น
- ดาวน์โหลดและติดตั้ง Git
- ดาวน์โหลดและติดตั้ง NodeJS และ NPM
- ดาวน์โหลดและติดตั้ง Postman สำหรับไว้ทดสอบ API

Module 2: Creating and Setting the Initial Structure of Laravel for the RESTful API

- Obtaining Laravel Using Composer
- Adding Laravel Homestead as a Dependency for The Laravel Project
- Preparing and Configuring Laravel Homestead



- Adding the Virtual Domain on The System
- Using Vagrant to Manage the Laravel Homestead Virtual Machine
- Exercise Find out how to go out from the Virtual Machine
- Keep The Laravel Project Updated

Module 3: Configuring and Using Sublime Text 3 to Develop the RESTful API with Laravel

- Installing the Package Control
- Creating and Using Projects
- Installing and Using SublimeCodeIntel to Resolve Laravel and RESTful API classes
- Installing and Using SideBarEnhancement
- Installing and Using SublimeLinter for PHP
- Installing and Using PHPCompanion to Auto-import Laravel Definitions

Module 4: Discovering and Configuring the Laravel Structure for the RESTful API

- Discovering the Laravel Structure
- Discovering the Laravel PHP Artisan Commands
- Configuring the Laravel Environment Variables
- Understanding the Laravel Routes System for RESTful APIs

Module 5: Creating the Initial Laravel Components for the RESTful API

- How to Solve Common Errors During the Development of the RESTful API
- Creating Initial Laravel Structures for The Models and Its Migrations
- Creating an Initial Laravel Structure for The Controllers
- Creating Some End-Point for the RESTful API Using Laravel Resource Routes



Module 6: Implementing the RESTful API Models and its Relationships using Laravel Eloquent

- Implementing The Properties for Category
- Implementing The Properties for Product
- Implementing The Properties for Transaction
- Implementing The Properties for User
- Implementing The Relationships Between Models

Module 7: Creating the Database Structure Using Migrations from Laravel

- Solving a Common Issue with the Laravel Migrations
- Implementing The Migration for Users
- Implementing The Migration for Categories
- Implementing The Migration for Products
- Implementing The Migration for Transactions
- Creating the Migration for The Pivot Table

Module 8: Creating the Laravel Factories for Database Seeding

- Creating The Laravel Factory for User
- Creating The Factory for Category
- Creating The Factory for Product
- Creating The Factory for Transaction
- Using The Factories from The DatabaseSeeder of Laravel
- Executing The Migrations and Seeder using Artisan from Laravel



Module 9: Implementing the Operations for UserController

- Implementing The Index Method for UserController
- Implementing The Show Method for UserController
- Implementing The Store Method for UserController
- Implementing The Update Method for UserController
- Implementing The Destroy Method for UserController

Module 10: Implementing the operations for Buyer

- Implementing The Index Method for BuyerController
- Implementing The Show Method for BuyerController

Module 11: Implementing the operations for Seller

- Implementing The Index Method for SellerController
- Implementing The Show Method for SellerController

Module 12: Improving the current RESTful API operations

- Defining Mutators and Accessor for Models
- Generalizing The Response Methods
- Using The Generalized Methods
- Using The Generalized Methods for Error Responses

Module 13: Handling Errors and Exceptions with the Laravel Handler

- Returning Validation Errors as a JSON Response
- Returning Model Not Found Errors as a JSON Response
- Handling AuthenticationException



- Handling AuthorizationException
- Handling NotFoundHttpException
- Handling MethodNotAllowedHttpException
- Handling General HttpException
- Catching Exceptions When Removing Related Resources
- Handling Unexpected Exceptions

Module 14: Implicit Model Binding with Routes and Methods

- Using The Laravel Implicit Model Binding for Some Methods
- Resolving Buyer Using Laravel Global Scopes
- Resolving Seller Using Laravel Global Scopes

Module 15: Implementing Soft Deleting for All the Models

- Modifying the Migrations for Soft Deleting
- Using soft Deleting for All the Models

Module 16: CORS for Laravel

- About CORS
- Installing the CORS Package for Laravel
- Configuring the CORS Package
- Allowing CORS Only for the API
- Allowing CORS on Error Responses Too

Module 17: Implementing the Operations for Category

Module 18: Implementing the Operations for Product



Module 19: Implementing Complex Operations within Transaction

Module 20: Implementing Complex Operations for Buyer

Module 21: Implementing Complex Operations for Category

Module 23 Implementing Complex Operations for Seller

Module 24: Implementing Complex Operations for Product

Module 25: Adding an Image for Products

Module 26: Sending Email for Users Accounts Verification

Module 27: The Middleware and the Rate Limiting

Module 28: Transforming Responses with PHP Fractal for Security and Compatibility

Module 29: Sorting and Filtering Results Based on Query Parameters

Module 30: Using Laravel Passport to Implement the Initial Security Layer with OAuth2

- About OAuth2 and Security on APIs
- What is Laravel Passport
- Preparing the Laravel Project to Use Laravel Passport
- Using the Migrations of Laravel Passport
- Installing Laravel Passport Definitively on the API
- Making Users Authenticatable Through API Tokens
- Registering the Required Routes
- Configuring the Authentication System
- Defining the Expiration Time for the Tokens



- Using client_credentials Grant to Protect Some Routes
- How to Obtain and Use Tokens Using Client Credentials
- Protecting All the Routes
- Using the Password Grant Type
- Preparing the API to Use the Passport Components
- Creating the Views to Manage the Personal Tokens
- About the Personal Tokens and How to Use It
- Allowing the Management of API Clients
- About the Authorization Code Grant Type
- Using the Implicit Grant Type
- Refreshing Tokens