



สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

COURSE ID

WDC-90

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

WEB DEVELOPMENT / NODE.JS

Advanced NestJS

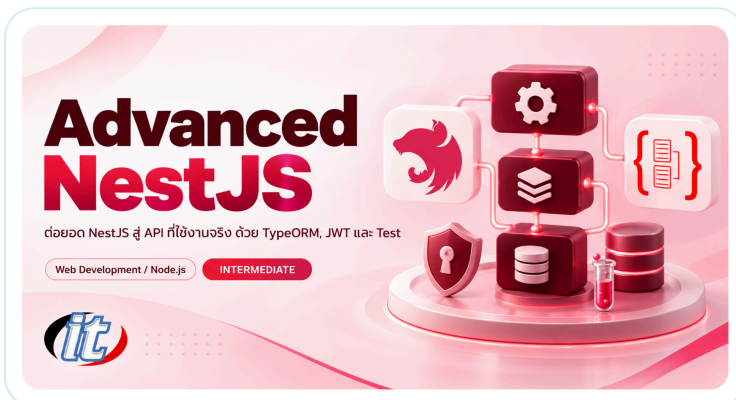
ต่อยอด NestJS สู่ API ที่ใช้งานจริง ด้วย TypeORM, JWT และ Test

● 18 ชั่วโมง

● 3 วัน

● พื้นฐาน NestJS และ TypeScript

● ระดับ Intermediate



ระยะเวลา: 3 วัน

18 ชั่วโมง (3 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

NestJS เป็นเฟรมเวิร์กฝั่ง Server ของ Node.js ที่ทีมพัฒนามักนิยมใช้สร้าง API ระดับองค์กร เพราะมีโครงสร้าง Module, Dependency Injection และ TypeScript ที่ชัดเจน แต่เมื่อผ่านขั้นพื้นฐานไปแล้ว นักพัฒนามักติดอยู่กับเรื่องที่ทำให้ระบบจริงยาก เช่น ออกแบบ Relation ของฐานข้อมูลไม่ดีขึ้นเกิด Query ซ้ำ เปิด synchronize ไว้บน Production ไม่มี Transaction ระบบล็อกอินที่ไม่ปลอดภัย และไม่มี Test รองรับเมื่อต้องแก้ไขโค้ด ขณะที่ NestJS 12 และ TypeORM 1.x ก็เปลี่ยนแนวทางหลายอย่างจากรุ่นที่ตำราส่วนใหญ่ใช้

หลักสูตรนี้ต่อยอดจากพื้นฐาน NestJS สู่การสร้าง API ที่พร้อมใช้งานจริงด้วย NestJS 12 บน Node.js 24 LTS เริ่มจากเชื่อม MySQL 8.4 ด้วย TypeORM 1.x ออกแบบ Entity และ Relation เขียน Query ขั้นสูงด้วย QueryBuilder จัดการ Migration และ Transaction จากนั้นออกแบบ RESTful API ด้วย DTO และ Validation ทั้งแบบ class-validator และ Standard Schema ใช้ Interceptor, Exception Filter และเอกสาร OpenAPI แล้วทำระบบล็อกอินด้วย JWT พร้อม Guard และ Role เพิ่มความปลอดภัย Cache และ Rate Limit เขียน Unit และ E2E Test ด้วย Vitest ปิดท้ายด้วยการนำแอปขึ้น Docker และ Capstone สร้าง API ร้านค้าออนไลน์ครบวงจร (อบรม 3 วัน วันละ 6 ชั่วโมง รวม 18 ชั่วโมง ระดับ Intermediate)

เครื่องมือ

N

NestJS 12

TS

TypeScript

TO

TypeORM

My

MySQL 8.4

JWT

JWT Auth

ราคาค่าอบรม · PUBLIC / IN-HOUSE

9,500 บาท / ท่าน

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide ซอร์สโค้ดตัวอย่างครบทุก Lab ไฟล์ docker-compose สำหรับ MySQL และ Certificate · รับจัด In-house ขอใบเสนอราคาได้

สมัครผ่านเว็บไซต์ ลด 10%



เพิ่มเพื่อน
LINE

☎ 02-570-8449

☎ 088-807-9770

💬 @itgenius

🌐 www.itgenius.co.th

01 วัตถุประสงค์ Objectives

- ตั้งโปรเจกต์ NestJS 12 แบบปัจจุบัน พร้อม Configuration และเครื่องมือทดสอบ
- เชื่อม MySQL ด้วย TypeORM 1.x และออกแบบ Entity กับ Relation ได้ถูกต้อง
- เขียน Query ขึ้นสูง จัดการ Migration และ Transaction ได้อย่างปลอดภัย
- ออกแบบ RESTful API ด้วย DTO, Validation, Interceptor และ Exception Filter
- สร้างเอกสาร API ตามมาตรฐาน OpenAPI ให้ทีมใช้งานร่วมกัน
- ทำระบบล็อกอินด้วย JWT และกำหนดสิทธิ์ด้วย Guard และ Role
- เขียน Unit Test และ E2E Test ให้กับ API ด้วย Vitest
- นำแอป NestJS ขึ้น Docker พร้อม Health Check

02 กลุ่มเป้าหมาย Target Audience

- นักพัฒนาที่ผ่านพื้นฐาน NestJS แล้วและต้องการสร้าง API สำหรับงานจริง
- นักพัฒนา Node.js และ Express ที่ต้องการโครงสร้างที่เป็นระบบมากขึ้น
- นักพัฒนา Full-stack ที่ดูแลทั้ง Frontend และ API ด้วย TypeScript
- ทีมพัฒนาที่ต้องการอัปเดตโปรเจกต์ NestJS เดิมเป็นรุ่น 12
- ครู อาจารย์ และนักศึกษาสายคอมพิวเตอร์ที่ต้องการแนวทางพัฒนา Backend แบบปัจจุบัน

03 จุดเด่นของหลักสูตร Highlights

- อัปเดตเป็น **NestJS 12** Node.js 24 LTS, TypeORM 1.x และ Vitest
- ฐานข้อมูลอย่างมืออาชีพ Relation, QueryBuilder, Migration และ Transaction
- **API ที่ออกแบบถูกหลัก** DTO, Validation, Interceptor และ OpenAPI
- **ปลอดภัยตั้งแต่ต้น** JWT, Guard, Role, Helmet และ Rate Limit
- **มี Test สองระดับ** Unit Test และ E2E Test พร้อม Docker

04 ความรู้พื้นฐาน Prerequisites

- เคยสร้างโปรเจกต์ NestJS และเข้าใจ Module, Controller และ Provider
- เขียน TypeScript ได้ เช่น Class, Interface, Decorator และ async/await
- เข้าใจ HTTP, RESTful API และ JSON
- รู้จักฐานข้อมูลเชิงสัมพันธ์และคำสั่ง SQL พื้นฐาน
- โน้ตบุ๊กที่ติดตั้ง Node.js, Docker Desktop และ VS Code ได้ พร้อมสิทธิ์ติดตั้งโปรแกรม

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook Windows, macOS หรือ Linux เชื่อมต่ออินเทอร์เน็ต ติดตั้ง Docker Desktop ได้
แพลตฟอร์ม	Node.js 24 LTS, TypeScript, NestJS 12 และ Nest CLI
ฐานข้อมูล	MySQL 8.4 LTS บน Docker, TypeORM 1.x กับไดรเวอร์ mysql2 และ DBeaver
ไลบรารีเสริม	@nestjs/jwt, @nestjs/swagger, @nestjs/throttler, @nestjs/terminus, class-validator และ Zod
ทดสอบและ Deploy	Vitest, Supertest, Postman หรือ Bruno และ Docker Compose

หมายเหตุ: หลักสูตรนี้ปรับปรุงใหม่ปี 2026 จากฉบับเดิมที่ครอบคลุม TypeORM กับ MySQL, Web API และ JWT เนื้อหาเปลี่ยนเป็น NestJS 12 ที่เป็น ESM และใช้ Vitest เป็นคำเริ่มต้นของโปรเจกต์ใหม่ ย้าย TypeORM เป็นรุ่น 1.x ที่ใช้ DataSource และไดรเวอร์ mysql2 เปลี่ยนจาก synchronize เป็น Migration เพิ่ม Standard Schema Validation, OpenAPI, Guard แบบ Global, Refresh Token, Cache, Rate Limit, Test และ Docker หลักสูตรนี้ต่อจาก Basic NestJS และไม่ครอบคลุม Microservices ซึ่งอยู่ในหลักสูตร Building Microservices with NestJS เครื่องมือทั้งหมดเป็นโอเพนซอร์สหรือใช้ฟรี

06 ภาพรวมการเรียนรู้ 3 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-5	ตั้งโปรเจกต์ NestJS 12, เชื่อม MySQL 8.4 ด้วย TypeORM 1.x, ออกแบบ Entity และ Relation, เขียน Query ด้วย QueryBuilder แล้วจัดการ Migration, Seed และ Transaction
Day 2 Section 6-10	ออกแบบ Controller และ Route ขั้นสูง, ตรวจสอบข้อมูลด้วย ValidationPipe หรือ Standard Schema, ใช้ Interceptor และ Exception Filter, ทำเอกสาร OpenAPI และ Workshop สร้าง API ร้านค้า
Day 3 Section 11-15	ทำระบบล็อกอินด้วย JWT และ Guard, กำหนดสิทธิ์ด้วย Role, เพิ่มความปลอดภัยและ Cache, เขียน Unit และ E2E Test ด้วย Vitest แล้วปิดท้ายด้วย Docker และ Capstone

DAY
1

TypeORM และฐานข้อมูล MySQL

เนื้อหาการอบรม · Section 1-5

รวม
6 ชั่วโมง

1 Lab: เตรียมโปรเจกต์ NestJS 12

- ขงใหม่ใน NestJS 12: แพ็กเกจแบบ ESM, Standard Schema และ Vitest
- สร้างโปรเจกต์ด้วย Nest CLI บน Node.js 24 LTS และเลือก ESM หรือ CommonJS
- จัดการ Configuration ด้วย @nestjs/config พร้อมตรวจค่า Environment
- รับ MySQL 8.4 ด้วย Docker Compose และเชื่อมต่อกับ DBeaver
- Lab: โครงโปรเจกต์ร้านค้าออนไลน์ที่ใช้ตลอดหลักสูตร

2 TypeORM 1.x กับ NestJS

- แนวคิด ORM, DataSource และรูปแบบ Data Mapper
- สิ่งที่เปลี่ยนจาก TypeORM 0.3 เป็น 1.x เช่น Node.js 20 ขึ้นไปและไดรเวอร์ mysql2
- ตั้งค่า TypeOrmModule.forRootAsync ร่วมกับ ConfigService
- สร้าง Entity และ Inject Repository เข้า Service
- เหตุผลที่ไม่ควรเปิด synchronize บน Production

3 Lab: Entity และ Relation

- ชนิดคอลัมน์, Primary Key แบบตัวเลขหรือ UUID และ Index
- One to One, Many to One และ One to Many
- Many to Many พร้อมตารางเชื่อมที่มีข้อมูลเพิ่ม
- Cascade, Eager และ Lazy Relation และผลกระทบต่อประสิทธิภาพ
- Lab: ออกแบบ Entity สินค้า หมวดหมู่ ลูกค้า และคำสั่งซื้อ

4 Query ขั้นสูง

- Find Options: where, relations, order และ select
- QueryBuilder สำหรับ Join, Group By และรายงานสรุป
- Pagination และการค้นหาแบบหลายเงื่อนไข
- ตรวจสอบและแก้ปัญหา N+1 Query จาก Log ของ SQL
- รับ Raw SQL แบบ Parameterized อย่างปลอดภัย

5 Lab: Migration, Seed และ Transaction

- สร้างและรัน Migration ด้วย TypeORM CLI
- เขียนสคริปต์ Seed ข้อมูลตัวอย่างที่รันซ้ำได้
- Transaction ด้วย DataSource.transaction และ QueryRunner
- Optimistic Locking ด้วยคอลัมน์ Version
- Lab: บันทึกคำสั่งซื้อและตัดสต็อกสินค้าภายใน Transaction



DAY
2

ออกแบบ RESTful API ขั้นสูง

เนื้อหาการอบรม · Section 6-10

รวม
6 ชั่วโมง

6 Controller และ Routing ขั้นสูง

- ออกแบบ Resource, Status Code และ Header ให้ถูกหลัก REST
- Route Parameter, Query String และ Pipe แปลงชนิดข้อมูล
- รูปแบบ Route แบบใหม่ที่ใช้ชื่อพารามิเตอร์สำหรับ Wildcard
- API Versioning ผ่าน URI หรือ Header
- Custom Decorator สำหรับดึงข้อมูลจาก Request

7 Lab: DTO และ Validation

- ValidationPipe กับ class-validator และ class-transformer
- whitelist, transform และ Mapped Types เช่น PartialType
- StandardSchemaValidationPipe กับ Zod ที่มาใหม่ใน NestJS 12
- เลือกแนวทาง Validation ให้เหมาะกับโปรเจกต์และทีม
- Lab: ตรวจสอบข้อมูลสินค้าและคำสั่งซื้อครบทุก Endpoint

8 Interceptor, Exception Filter และ Serialization

- Interceptor สำหรับ Logging, วัดเวลา และจัดรูปแบบ Response
- Exception Filter กลางสำหรับแปลง Error ของฐานข้อมูล
- ซ่อนฟิลด์ที่ไม่ควรส่งออกด้วย Serialization
- ลำดับการทำงานของ Middleware, Guard, Interceptor และ Pipe

9 เอกสาร API ด้วย OpenAPI

- ติดตั้ง @nestjs/swagger และเปิดหน้าเอกสาร Swagger UI
- ใช้ CLI Plugin ลด Decorator ที่ต้องเขียนซ้ำ
- อธิบาย DTO, Response และการยืนยันตัวตนแบบ Bearer ในเอกสาร
- ส่งออกไฟล์ OpenAPI ให้ทีม Frontend สร้าง Client ต่อได้

10 Workshop: RESTful API ร้านค้าออนไลน์

- API สินค้าและหมวดหมู่พร้อม Pagination, Filter และ Sorting
- API คำสั่งซื้อที่ทำงานภายใน Transaction
- Upload รูปสินค้าพร้อมตรวจสอบขนาดและชนิดไฟล์ด้วย ParseFilePipe
- ทดสอบทุก Endpoint ผ่าน Swagger UI, Postman หรือ Bruno

DAY
3

Authentication, Testing และ Deploy

เนื้อหาการอบรม · Section 11-15

รวม
6 ชั่วโมง

11 Lab: ระบบล็อกอินด้วย JWT

- Hashing รหัสผ่านด้วย bcrypt หรือ Argon2 เทียบกับการเข้ารหัสข้อมูล
- ออก Access Token ด้วย @nestjs/jwt และตรวจด้วย AuthGuard
- Guard แบบ Global พร้อม Decorator @Public() สำหรับ Route สาธารณะ
- Passport Strategy เป็นทางเลือกเมื่อต้องรองรับการล็อกอินหลายแบบ
- Lab: API สมัครงานเข้าสู่ระบบ และดูข้อมูลของตนเอง

12 Authorization และ Refresh Token

- Role Decorator และ RolesGuard สำหรับ Admin, Staff และ Customer
- ตรวจสอบความเป็นเจ้าของข้อมูล เช่น ลูกค้านี้ได้เฉพาะคำสั่งซื้อของตนเอง
- Refresh Token แบบหมุนเวียนและการเพิกถอน Token
- Lab: กำหนดสิทธิ์ให้ API ร้านค้าครบทุก Endpoint

13 ความปลอดภัยและประสิทธิภาพ

- ตั้งค่า CORS และ Helmet ให้เหมาะกับการใช้งานจริง
- จำกัดจำนวนคำขอด้วย @nestjs/throttler
- Cache ผลลัพธ์ที่เรียกบ่อยด้วย CacheModule
- ช่องโหว่ที่พบบ่อยตาม OWASP API Security Top 10
- Logging และ Health Check ด้วย @nestjs/terminus

14 Lab: Unit Test และ E2E Test

- เขียน Unit Test ด้วย Vitest และ Test.createTestingModule
- Mock Repository และ Service ที่ต้องพึ่งพา
- E2E Test ด้วย Supertest กับฐานข้อมูลทดสอบบน Docker
- วัด Test Coverage และรัน Test อัตโนมัติก่อน Merge

15 Workshop: Docker และ Capstone

- เขียน Dockerfile แบบ Multi-stage สำหรับแอป NestJS
- Docker Compose สำหรับ API และ MySQL พร้อมรัน Migration อัตโนมัติ
- Capstone: เติมระบบรีวิวสินค้าหรือคู่มือส่วนลดพร้อมสิทธิ์และ Test
- นำเสนอผลงาน รีวิวโค้ดร่วมกัน และสรุป Checklist ก่อนขึ้น Production

07 เรียนจบแล้วได้อะไร Outcomes

- ใช้ TypeORM อย่างมืออาชีพ Relation, QueryBuilder และ Migration
- จัดการข้อมูลได้ถูกต้อง Transaction และ Locking
- ออกแบบ API ได้ดี DTO, Validation และ OpenAPI
- ทำระบบสิทธิ์ได้ JWT, Guard, Role และ Refresh Token
- เขียน Test ได้เอง Vitest, Unit Test และ E2E Test
- พร้อมขึ้นระบบจริง Security, Cache และ Docker

08 สิ่งที่คุณเรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ครบทั้ง 3 วัน และ Lab Guide ทีละขั้นตอน
- ซอร์สโค้ดตัวอย่างของ API ร้านค้าออนไลน์และ Capstone ครอบคลุม Lab
- ไฟล์ docker-compose สำหรับ MySQL 8.4 และ Dockerfile แบบ Multi-stage
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สนิทวงศ์ กมลภากรณ์ ผู้สอนและนักพัฒนาระบบที่ใช้ NestJS และ TypeScript สร้าง API และระบบหลังบ้านในงานจริง จุดเด่นคือการสอนผ่านโปรเจกต์ร้านค้าออนไลน์ชุดเดียวที่ต่อเติมตลอดหลักสูตร อธิบายว่าเบื้องหลังแต่ละ Decorator และแต่ละคำสั่งของ ORM ทำงานอย่างไร และพาผู้เรียนลงมือทำจนได้ API ที่มีสิทธิ์ผู้ใช้ มี Test และพร้อมขึ้นระบบด้วยตนเอง

ราคาค่าอบรม 9,500 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

9,500 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

☎ 02-570-8449

📞 @itgenius

☎ 088-807-9770

✉ contact@itgenius.co.th



เพิ่มเพื่อน
LINE