



สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

COURSE ID

WDC-122

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

WEB DEVELOPMENT / FULL STACK

Road to Full Stack Developer

สร้างระบบเว็บครบวงจรด้วย TypeScript ตั้งแต่ UI จนถึง Deploy

● 30 ชั่วโมง ● 5 วัน ● พื้นฐาน JavaScript, HTML/CSS และ SQL เบื้องต้น

● ระดับ Beginner to Intermediate



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

องค์กรจำนวนมากต้องการนักพัฒนาที่ส่งมอบฟีเจอร์ได้ครบวงจร ตั้งแต่หน้าจอผู้ใช้ API ฐานข้อมูล จนถึงการนำระบบขึ้นใช้งานจริง แต่นักพัฒนาส่วนใหญ่ถูกมัดเพียงฝั่งเดียว เมื่อต้องทำงานข้ามฝั่งก็มักเจอปัญหาโครงสร้างโปรเจกต์ไม่เป็นระบบ ชนิดข้อมูลระหว่าง Frontend กับ Backend ไม่ตรงกัน ระบบยืนยันตัวตนไม่ปลอดภัย และไม่มีขั้นตอนทดสอบหรือ Deploy ที่ทำซ้ำได้

หลักสูตรนี้พาผู้เรียนสร้างระบบจริงหนึ่งระบบด้วย TypeScript ทั้ง Stack ใช้ React 19, Vite 8 และ Tailwind CSS v4 ฝั่ง Frontend ใช้ Hono บน Node.js 24 LTS ฝั่ง Backend ใช้ PostgreSQL 18 กับ Prisma ORM 7 ฝั่งฐานข้อมูล แשר Schema ระหว่างสองฝั่งด้วย Zod ทดสอบด้วย Vitest และ Playwright แล้วปิดท้ายด้วยการแพ็คเกจระบบด้วย Docker พร้อม Pipeline สำหรับ Deploy ผ่านโปรเจกต์ระบบจองทรัพยากรภายในองค์กร BookingNest ที่ต่อยอดทุกวันจนรันได้ครบวงจร (อบรม 5 วัน วันละ 6 ชั่วโมง รวม 30 ชั่วโมง ระดับ Beginner to Intermediate)

เครื่องมือ

TS TypeScript

Re React

Hn Hono

pg PostgreSQL

Tw Tailwind CSS

ราคาค่าอบรม · PUBLIC / IN-HOUSE

10,500 บาท / ท่าน

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide รายวัน Source Code ของโปรเจกต์ครบทุก Workshop ไฟล์ Docker และ Pipeline ตัวอย่าง และ Certificate · รับจัด In-house ขอใบเสนอราคาได้

สมัครผ่านเว็บไซต์ ลด 10%



เพิ่มเพื่อน
LINE

☎ 02-570-8449

☎ 088-807-9770

LINE @itgenius

🌐 www.itgenius.co.th

01 วัตถุประสงค์ Objectives

- เข้าใจสถาปัตยกรรมของ Web Application สมัยใหม่และบทบาทของแต่ละชั้น ตั้งแต่ Client, API, Database จนถึง Infrastructure
- ใช้ TypeScript ได้อย่างมั่นใจทั้งฝั่ง Frontend และ Backend และแชร์ชนิดข้อมูลข้ามโปรเจกต์ได้
- ออกแบบและพัฒนา RESTful API ที่มีโครงสร้างดี ตรวจสอบข้อมูล และจัดการข้อผิดพลาดอย่างเป็นมาตรฐาน
- ออกแบบ Schema ฐานข้อมูล PostgreSQL และจัดการ Migration ด้วย Prisma ORM ได้
- พัฒนาระบบ Authentication และ Authorization ที่ปลอดภัยตามแนวทาง OWASP
- พัฒนา Frontend ด้วย React ที่จัดการ Server State, Form และสถานะ Loading/Error ได้ถูกต้อง
- เขียน Unit Test, Integration Test และ End-to-End Test ให้ระบบ Full Stack
- แพ้ระบบด้วย Docker และวางขั้นตอน Build และ Deploy ที่ทำซ้ำได้

02 กลุ่มเป้าหมาย Target Audience

- นักพัฒนา Frontend ที่ต้องการขยายทักษะไปสู่ Backend และฐานข้อมูล
- นักพัฒนา Backend ที่ต้องการพัฒนา Frontend สมัยใหม่ด้วย React ได้อย่างมั่นใจ
- นักพัฒนาที่ใช้ภาษาอื่น เช่น PHP, Java หรือ C# และต้องการย้ายมาใช้ TypeScript Stack
- Tech Lead ที่ต้องการวางมาตรฐานโครงสร้างโปรเจกต์ Full Stack ให้ทีม
- นักพัฒนาในองค์กรที่ต้องสร้างระบบภายใน (Internal Tools) ด้วยทีมขนาดเล็ก

03 จุดเด่นของหลักสูตร Highlights

- **TypeScript ทั้งระบบ** ภาษาเดียวและ Zod Schema ร่วมกันสองฝั่ง
- **Stack รุ่นปัจจุบัน** React 19, Vite 8, Hono, Prisma 7, PostgreSQL 18
- **ความปลอดภัยตั้งแต่ต้น** Refresh Token, httpOnly Cookie, RBAC และ OWASP
- **Test ครบทุกชั้น** Vitest สำหรับ API และ Playwright สำหรับ E2E
- **ปิดท้ายด้วยการ Deploy** Docker Compose และ CI/CD Pipeline

04 ความรู้พื้นฐาน Prerequisites

- เขียนโปรแกรมได้อย่างน้อย 1 ภาษา และเข้าใจตัวแปร ฟังก์ชัน เงื่อนไข ลูป และ Object
- มีพื้นฐาน HTML, CSS และ JavaScript เช่น Array Method, Promise และ async/await
- เข้าใจแนวคิด HTTP Request/Response และ SQL เบื้องต้น ใช้ Git ระดับ clone, commit, push และ pull ได้
- ไม่จำเป็นต้องเคยใช้ React, TypeScript, Prisma หรือ Docker มาก่อน ผู้สอนจะปูพื้นฐานส่วนที่ใช้จริงในโปรเจกต์ให้

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
ภาษาและโครงสร้าง	TypeScript 7, Node.js 24 LTS และ pnpm Workspaces (Monorepo) บบ VS Code พร้อม ESLint และ Prettier
Frontend	React 19, Vite 8, Tailwind CSS v4, React Router, TanStack Query, React Hook Form และ Zod
Backend และ Database	Hono 4 บบ Node.js, PostgreSQL 18, Prisma ORM 7, JWT, httpOnly Cookie และ Argon2
Testing และ Deploy	Vitest, Playwright, Docker, Docker Compose และ GitHub Actions หรือ GitLab CI
อุปกรณ์และบัญชี	Notebook RAM 16 GB พื้นที่ว่าง 30 GB ขึ้นไป ติดตั้ง Docker Desktop ได้ และบัญชี GitHub หรือ GitLab ของผู้เรียน

หมายเหตุ: หลักสูตรนี้เน้นแนวทาง SPA + REST API แยกส่วนด้วย React และ Hono จึงต่างจาก Full-Stack Web Application with Next.js 16 and Prisma ที่ใช้ Full Stack Framework และลงลึกกว่า React and NodeJS with Docker Workshop ในด้านความปลอดภัย การทดสอบ และ Pipeline เครื่องมือทั้งหมดเป็นโอเพนซอร์สและรันบนเครื่องของผู้เรียนด้วย Docker การ Deploy ในวันสุดท้ายใช้ได้กับ Server หรือ Cloud ที่ผู้เรียนหรือองค์กรใช้ หากต้องใช้บริการ Cloud ผู้เรียนใช้บัญชีของตนเองหรือแบบใช้ฟรี เวอร์ชันของเครื่องมือจะปรับตามรุ่นล่าสุด ณ วันอบรม

06 ภาพรวมการเรียนรู้ 5 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-5	สถาปัตยกรรม Full Stack, Monorepo ด้วย pnpm Workspaces, TypeScript 7 สำหรับทั้งสองฝั่ง, Zod Shared Schema และ Git Workflow สำหรับทีม
Day 2 Section 6-10	ออกแบบ RESTful API, Hono บบ Node.js 24, PostgreSQL 18 กับ Prisma ORM 7, Query ที่มีประสิทธิภาพ, Error Handling และ Logging
Day 3 Section 11-15	Session, JWT และ Refresh Token, Login Flow ที่ปลอดภัย, RBAC และสิทธิ์ระดับข้อมูล, ความปลอดภัยตาม OWASP และ Test ของ API ด้วย Vitest
Day 4 Section 16-20	React 19 และ Hook สมัยใหม่, Vite 8 และ Tailwind CSS v4, React Router และ TanStack Query, พอร์มด้วย React Hook Form และ E2E Test ด้วย Playwright
Day 5 Section 21-25	Dockerfile แบบ Multi-stage และ Docker Compose, เตรียมระบบสำหรับ Production, CI/CD Pipeline และ Master Workshop พัฒนาฟีเจอร์ใหม่ครบทุกชั้น

DAY
1

Modern Full Stack Foundations และ TypeScript

เนื้อหาการอบรม · Section 1-5

รวม
6 ชั่วโมง

1 ภาพรวม Full Stack Architecture

- บทบาทของ Client, API, Database และ Infrastructure ในระบบเว็บสมัยใหม่
- เปรียบเทียบ SPA + API แยกส่วน กับ Full Stack Framework อย่าง Next.js และเหตุผลที่หลักสูตรเลือก SPA + API
- การไหลของข้อมูลตั้งแต่ผู้ใช้กดปุ่มจนถึงฐานข้อมูล
- Lab: วาดแผนภาพสถาปัตยกรรมและ Flow การจองของ BookingNest

2 เตรียมเครื่องมือและโครงสร้างโปรเจกต์

- ติดตั้งและจัดการเวอร์ชัน Node.js 24 LTS และ pnpm
- สร้าง Monorepo ด้วย pnpm Workspaces แยก apps และ packages
- ตั้งค่า ESLint, Prettier และ EditorConfig ให้ทีมใช้มาตรฐานเดียวกัน
- Lab: สร้างโครง apps/web, apps/api และ packages/shared ที่ Build ได้ทั้งหมด

3 TypeScript สำหรับ Full Stack Developer

- Type, Interface, Union, Generic และ Utility Types ที่ใช้บ่อย
- ตั้งค่า tsconfig ให้เหมาะกับ Node.js และ Browser
- สิ่งที่เปลี่ยนใน TypeScript 6 และ 7 เช่น การเลือกรองรับ Target ES5 และ Native Compiler ที่ Build เร็วขึ้น
- Type Narrowing และการจัดการค่า null/undefined อย่างปลอดภัย
- Lab: กำหนด Type ของ Resource, Booking และ User แล้วใช้ร่วมกันทั้งสองฝั่ง

4 Zod และ Shared Schema

- นิยาม Schema ครึ่งเดียวแล้ว infer เป็น Type ให้ทั้งสองฝั่งใช้
- Validate ข้อมูลที่เข้าระบบและแปลง Error ให้อ่านง่าย
- จัด packages/shared ให้ import ได้จากทั้ง apps/web และ apps/api
- Lab: เขียน Zod Schema ของการจองพร้อมกฎวันเวลาและ Unit Test ด้วย Vitest

5 Workshop Day 1: Git Workflow และ Monorepo ของ BookingNest

- Branching Strategy แบบเรียบง่ายสำหรับทีมเล็ก
- Conventional Commits และการเขียน Pull/Merge Request ที่ดี
- Workshop: รวมเครื่องมือทั้งหมดเข้า Monorepo ของ BookingNest และส่ง Pull Request แรก
- Workshop: ตรวจสอบว่า Schema ใน packages/shared ผ่าน Test และ Lint ก่อน Merge



DAY
2

Backend API ด้วย Hono, PostgreSQL และ Prisma

เนื้อหาการอบรม · Section 6-10

รวม
6 ชั่วโมง

6 หลักการออกแบบ RESTful API

- ตั้งชื่อ Resource และเลือก HTTP Method และ Status Code ให้ถูกต้อง
- Pagination, Filtering, Sorting และรูปแบบ Error Response มาตรฐาน
- API Versioning และเอกสาร API ด้วย OpenAPI
- Lab: ออกแบบ Endpoint ทั้งหมดของ Resource และ Booking ก่อนเขียนโค้ด

7 Hono บน Node.js

- โครงสร้างแอป Routing, Middleware และ Context
- แยก Layer เป็น Route, Service และ Repository
- ใช้ Zod Validator กับ Request Body, Query และ Params
- Lab: สร้าง API ของ Resource พร้อม Validation จาก Shared Schema

8 PostgreSQL 18 และ Prisma ORM 7

- ออกแบบ Schema ความสัมพันธ์ One-to-Many และ Many-to-Many
- Prisma Schema, Migration และ Seed Data
- สิ่งที่เปลี่ยนใน Prisma 7 ที่ Prisma Client ไม่พึ่ง Rust Query Engine แล้ว
- Lab: รับ PostgreSQL ใน Docker ทำ Migration และ Seed ห้องประชุมและรถส่วนกลาง

9 การ Query อย่างมีประสิทธิภาพ

- Select เฉพาะ Field ที่ใช้, Include ความสัมพันธ์ และการหลีกเลี่ยง N+1 Query
- Transaction สำหรับงานที่ต้องสำเร็จพร้อมกัน
- สร้าง Index ให้ตรงกับรูปแบบการค้นหา
- Lab: วัดจำนวน Query ก่อนและหลังแก้ปัญหา N+1 ในหน้ารายการจอง

10 Workshop Day 2: Error Handling, Logging และ API ของ BookingNest

- Global Error Handler และการแยก Error ฝั่งผู้ใช้กับฝั่งระบบ
- Structured Logging และ Request ID
- Workshop: พัฒนา API จัดการ Resource และ Booking คสม CRUD ค้นหา และกรองตามวันที่
- Workshop: ทดสอบทุก Endpoint ด้วย HTTP Client และเปิดเอกสาร OpenAPI ให้ทีม Frontend



DAY
3

Authentication, Authorization และ API Quality

เนื้อหาการอบรม · Section 11-15

รวม
6 ชั่วโมง

11 แนวคิด Authentication สมัยใหม่

- เปรียบเทียบ Session, JWT และ Access/Refresh Token
- เก็บ Token ใน httpOnly Cookie เทียบกับ Local Storage และผลด้านความปลอดภัย
- Hash รหัสผ่านด้วย Argon2
- Lab: เลือกวิธียืนยันตัวตนของ BookingNest พร้อมเหตุผลด้านความปลอดภัย

12 พัฒนา Login Flow แบบครบวงจร

- Register, Login, Refresh Token และ Logout
- Token Rotation และการเพิกถอน Refresh Token
- ตั้งค่า CORS และ Cookie (SameSite, Secure) ให้ Frontend เรียกใช้ได้อย่างปลอดภัย
- Lab: สร้าง Endpoint สมัครงานและเข้าสู่ระบบพร้อม Refresh Token แบบ Rotation

13 Authorization แบบ Role-Based

- ออกแบบ Role และ Permission: User, Approver และ Admin
- Middleware ตรวจสอบสิทธิ์ตาม Role
- ตรวจสอบสิทธิ์ระดับข้อมูล เช่น แก้ไขได้เฉพาะการจองของตนเอง
- Lab: เพิ่ม API อนุมัติและปฏิเสธการจองสำหรับ Approver

14 ความปลอดภัยของ API

- ความเสี่ยงตาม OWASP ที่เกี่ยวกับ API: Broken Access Control, Injection และ Security Misconfiguration
- Rate Limiting และ Security Headers
- จัดการ Secret ผ่าน Environment Variables
- Lab: ลองเจาะสิทธิ์ข้ามผู้ใช้แล้วปิดช่องโหว่ที่พบ

15 Workshop Day 3: Testing สำหรับ API

- Unit Test ของ Service ด้วย Vitest
- Integration Test ของ Route โดยเรียกแอป Hono โดยตรงร่วมกับฐานข้อมูลทดสอบแยก
- วัด Code Coverage และเลือกตัวทดสอบอะไรก่อน
- Workshop: เขียนชุด Test ครอบคลุม Flow การจอง การอนุมัติ และสิทธิ์ทั้ง 3 ระดับ

DAY
4

Frontend ด้วย React 19, Vite 8 และ Tailwind CSS

เนื้อหาการอบรม · Section 16-20

รวม
6 ชั่วโมง

16 React 19 ที่นักพัฒนาต้องรู้

- Component, Props, State และการแยก Component ให้ดูแลง่าย
- Hook ที่ใช้บ่อย รวมถึง useActionState, useOptimistic และ useEffectEvent
- จัดโครงสร้างโฟลเดอร์แบบ Feature-based
- Lab: สร้าง Component รายการทรัพยากรและการจัดการจอง

17 Vite 8 และ Tailwind CSS v4

- โปรเจกต์ Vite 8 ที่ใช้ Rolldown เป็น Bundler และการตั้งค่า Proxy ไปยัง API
- Tailwind CSS v4 แนวคิด CSS-first Configuration และ Design Token ของโปรเจกต์
- สร้าง UI Component ที่ใช้ซ้ำได้และรองรับหน้าจอกทุกขนาด
- Lab: วาง Layout หลักและ Design Token ของ BookingNest

18 Routing และ Server State

- React Router สำหรับหน้า Public, หน้าที่ต้อง Login และหน้าตาม Role
- TanStack Query: Query, Mutation, Cache Invalidation และ Loading/Error State
- Optimistic Update เมื่อผู้ใช้สร้างหรือยกเลิกการจอง
- Lab: ดึงรายการทรัพยากรและการจองจาก API ด้วย TanStack Query

19 Form และ Authentication ฝั่ง Client

- React Hook Form ร่วมกับ Zod Schema จาก packages/shared
- จัดการสถานะ Login และ Refresh Token อัตโนมัติ
- Redirect เมื่อ Session หมดอายุหรือไม่มีสิทธิ์
- Lab: สร้างฟอร์มจองที่ใช้กฎ Validation ชุดเดียวกับ API

20 Workshop Day 4: หน้าจอ BookingNest และ E2E Test ด้วย Playwright

- เขียน Test จำลองผู้ใช้จริงตั้งแต่ Login จนจองสำเร็จ
- เตรียมข้อมูลทดสอบและรัน E2E แบบ Headless
- Workshop: พัฒนาหน้า Login, ค้นหาทรัพยากร, ปฏิทินการจอง, การจองของจีน และหน้าอนุมัติ
- Workshop: เขียน E2E Test ของ Flow การจองและการอนุมัติ



DAY
5

Containerize, Deploy และ Master Workshop

เนื้อหาการอบรม · Section 21-25

รวม
6 ชั่วโมง

21 Docker สำหรับ Full Stack Application

- Dockerfile แบบ Multi-stage สำหรับ API (Node.js) และ Web (Static Build ผ่าน Nginx)
- Docker Compose สำหรับ Web, API และ PostgreSQL พร้อม Health Check
- จัดการ Environment Variables และ Secret แยกตามสภาพแวดล้อม
- Lab: รับ BookingNest ทั้งระบบด้วยคำสั่งเดียว

22 เตรียมระบบสำหรับ Production

- รับ Migration อย่างปลอดภัยตอน Deploy
- Logging, Health Endpoint และ Graceful Shutdown
- Checklist ความปลอดภัยก่อนขึ้นระบบจริง
- Lab: ตรวจสอบระบบตาม Checklist และแก้จุดที่ยังไม่ผ่าน

23 CI/CD Pipeline เบื้องต้น

- Pipeline ขั้นตอน Lint, Test, Build และสร้าง Container Image
- Deploy ขึ้น Server หรือ Cloud ที่องค์กรใช้
- แนวทาง Rollback เมื่อ Deploy แล้วมีปัญหา
- Lab: ตั้ง Pipeline บน GitHub Actions หรือ GitLab CI ให้ Build และ Test อัตโนมัติ

24 ออกแบบฟีเจอร์ใหม่ตั้งแต่ Database ถึง UI

- วิเคราะห์ Requirement ของฟีเจอร์ตรวจสอบการจองซ้อนเวลาและแจ้งเตือนผู้จอง
- ออกแบบ Constraint ในฐานข้อมูล การเปลี่ยนแปลง API และหน้าจอที่เกี่ยวข้อง
- วางแผน Test ที่ต้องเพิ่มในแต่ละชั้น
- Lab: เขียนแผนงานและแบ่งงานเป็น Pull Request ย่อย

25 Master Workshop: BookingNest ครอบคลุม

- พัฒนาฟีเจอร์ตรวจสอบการจองซ้อนเวลาตั้งแต่ Constraint ในฐานข้อมูล, API จนถึง UI
- เพิ่ม Unit, API และ E2E Test ของฟีเจอร์ใหม่
- ส่งผ่าน Pipeline และ Deploy ระบบด้วย Docker Compose
- นำเสนอผลงานและทบทวนโค้ดร่วมกัน (Code Review)

07 เรียนจบแล้วได้อะไร Outcomes

- **มองระบบได้ครบวงจร** ตั้งแต่หน้าจอลงถึงฐานข้อมูลและการ Deploy
- **ใช้ TypeScript ได้ทั้ง Stack** Type-safe และแชร์ Schema สองฝั่ง
- **พัฒนา API ระดับใช้งานจริง** Hono, Prisma และ PostgreSQL
- **วางระบบความปลอดภัยเป็น** Authentication, RBAC และ OWASP
- **สร้าง Frontend สมัยใหม่** React 19, Tailwind CSS v4 และ TanStack Query
- **ส่งมอบอย่างมีวินัย** Test ทุกชั้น, Docker และ Pipeline

08 สิ่งที่คุณเรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมครบทั้ง 5 วัน และ Lab Guide รายวันพร้อมโค้ดตั้งต้นของแต่ละวันสำหรับตามให้ทัน
- Source Code ของ BookingNest แบบ Monorepo ครอบคลุม Workshop ใช้เป็นแม่แบบโปรเจกต์ Full Stack ของทีมได้
- ไฟล์ Dockerfile, Docker Compose และ Pipeline ตัวอย่างสำหรับ GitHub Actions และ GitLab CI
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สาธิต ไทยม ผู้สอนและนักพัฒนาที่สร้าง Web Application แบบ Full Stack ด้วย TypeScript, React, Node.js และ PostgreSQL ในงานจริง จุดเด่นคือการสอนผ่านระบบเดียวที่คุณเรียนต่อด้วยตัวเองทุกวัน อธิบายให้เห็นว่าแต่ละชั้นเชื่อมกันอย่างไร และให้ผู้เรียนลงมือแก้ปัญหาที่เจอจริงอย่าง Type ไม่ตรงกัน ลิงก์รั่ว และ Deploy ไม่ผ่าน เพื่อให้กลับไปส่งมอบฟีเจอร์ได้ครบทุกชั้นอย่างมั่นใจ

ราคาค่าอบรม 10,500 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

10,500 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

☎ 02-570-8449

☎ 088-807-9770

LINE @itgenius

✉ contact@itgenius.co.th



เพิ่มเพื่อน
LINE