ไอทีจีเนียส เอ็นจิเนียริ่ง
IT GENIUS ENGINEERING CO., LTD.

COURSE ID
GENAI-5D
AI / GENERATIVE AI

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

Generative AI for Software Developer

- 30 ชั่วโมง
- 5 วัน
- ระดับ Intermediate–Advanced

ระยะเวลา: 5 วัน
30 ชั่วโมง (วันละ 6 ชม.)

รูปแบบ: In-house
ขอใบเสนอราคาได้

รายละเอียดหลักสูตร
www.itgenius.co.th

ปี 2026 คือจุดเปลี่ยนสำคัญของการพัฒนาซอฟต์แวร์ จากยุค “AI ช่วยเติมโค้ดอัตโนมัติ” สู่ยุค **Agentic Software Engineering** ที่ AI สามารถวางแผน เขียน แก้ไข ทดสอบ และส่งมอบฟีเจอร์ได้แบบกึ่งอัตโนมัติภายใต้การกำกับของนักพัฒนา นักพัฒนายุคใหม่จึงไม่ได้แข่งกันที่ “พิมพ์โค้ดเร็วแค่ไหน” อีกต่อไป แต่แข่งกันที่ความสามารถในการสื่อสารเจตนา (Intent) การออกแบบบริบท (Context Engineering) และการกำกับ AI Agent ให้ทำงานได้ถูกต้อง ปลอดภัย และมีคุณภาพ

หลักสูตรฉบับ 5 วันนี้ต่อยอดจากฉบับ 3 วัน โดยเพิ่ม **สองวันเชิงลึกด้าน AI Engineering และการนำระบบ AI ขึ้น Production (LLMOps)** เหมาะกับผู้ที่ต้องการก้าวจาก “ใช้ AI ช่วยพัฒนา” ไปสู่ “สร้างแอปพลิเคชันที่ขับเคลื่อนด้วย AI ได้จริง” ครอบคลุมตั้งแต่พื้นฐาน LLM, Prompt/Context Engineering, การใช้ AI Coding Agent, การเชื่อมต่อผ่าน MCP, การสร้างแอป AI (RAG, Tool Calling, AI Agent) ไปจนถึงการประเมินผล วาง Guardrails/Security และ Deploy ขึ้น Production ปิดท้ายด้วย **Capstone Project** ที่ผู้เรียนสร้างระบบจริงตั้งแต่ต้นจนจบ

TECH STACK

- Claude · Claude Code
- ChatGPT
- Gemini
- Copilot Agent
- Cursor
- MCP
- Python
- LangChain
- Vector DB

ราคาคอร์สอบรม · IN-HOUSE

ขอใบเสนอราคา

ราคาเหมาจ่ายสำหรับการอบรมภายในองค์กร · กำหนดวันและจำนวนผู้เรียนได้ตามตกลง · ขอใบเสนอราคาได้

เพิ่มเพื่อน LINE

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

01 วัตถุประสงค์ Objectives

- เข้าใจพื้นฐาน Generative AI, LLM, Transformer และ Reasoning Model พร้อมขีดความสามารถและข้อจำกัดของโมเดลยุค 2026
- เปรียบเทียบและเลือกใช้โมเดลชั้นนำให้เหมาะกับงาน (Claude, GPT, Gemini และโมเดล Open-source)
- ก้าวจาก Prompt Engineering สู่ Context Engineering เพื่อสั่งให้ AI ออกแบบ เขียน วิเคราะห์ และแก้ไขโค้ดได้อย่างแม่นยำ
- ใช้ AI Coding Agent ใน Workflow จริง ตั้งแต่วางแผน เขียนโค้ด Refactor เขียนเทสต์ ไปจนถึง Code Review
- เชื่อม AI เข้ากับเครื่องมือและแหล่งข้อมูลขององค์กรผ่าน MCP และสร้างแอป AI ด้วย RAG, Tool Calling และ AI Agent
- ประเมินผล ทดสอบ และนำระบบ AI ขึ้น Production ด้วย LLMOps, Evaluation, Observability และ Guardrails

02 กลุ่มเป้าหมาย Target Audience

- Software Developer ทุกสาย ทั้ง Full Stack / Backend / Frontend ที่ต้องการพัฒนาในยุค AI-first
- DevOps Engineer และ Software Architect ที่ต้องการนำ AI มาเสริมกระบวนการพัฒนา
- ผู้ที่ต้องการเปลี่ยนสายสู่ AI Engineer / AI Application Developer
- AI Enthusiast, Technical Lead และผู้ที่กำลังเปลี่ยนผ่านสู่ยุค AI
- นักพัฒนาแอปหรือ Freelancer ที่ต้องการลดเวลาและต้นทุนการพัฒนา รวมถึงผู้สอน/วิทยากร/Mentor ด้านการพัฒนาซอฟต์แวร์

03 ความรู้พื้นฐาน Prerequisites

- มีประสบการณ์เขียนโปรแกรมภาษาใดภาษาหนึ่ง (Python, JavaScript, TypeScript, Java ฯลฯ) — สำหรับวันที่ 4–5 แนะนำให้คุ้นเคย Python หรือ Node.js
- เข้าใจหลักการซอฟต์แวร์พื้นฐาน เช่น MVC, API, Database และการใช้งาน Git
- เคยใช้ VS Code หรือ IDE อื่นในการพัฒนา และใช้งานระบบปฏิบัติการ Windows / macOS / Linux ได้
- อ่าน-เขียนภาษาอังกฤษระดับพื้นฐานได้ (ไม่จำเป็นต้องคล่อง)

04 ระยะเวลาอบรม Duration

- **30 ชั่วโมง** · แบ่งเป็น 5 วัน (วันละ 6 ชั่วโมง) · เน้นลงมือทำจริงทุกวัน (Hands-on) ปิดท้ายด้วย Capstone Project

05 เนื้อหาการอบรม Training Topics

- **DAY 1** · Foundations: Generative AI, LLM & Context Engineering (Section 1–3)
- **DAY 2** · Agentic Coding, MCP & Software Design (Section 4–7)
- **DAY 3** · AI Prototyping, Automation, Testing & Best Practices (Section 8–11)
- **DAY 4** · AI Engineering: Building AI-Powered Applications (Section 12–15)
- **DAY 5** · Production, LLMOps, Evaluation & Capstone (Section 16–19)

DAY 1

Foundations: Generative AI, LLM & Context Engineering

เนื้อหาการอบรม · Section 1–3

รวม
6 ชั่วโมง

1 Generative AI & LLM Landscape ในปี 2026

- แนวคิด Generative AI และบทบาทใน Software Development Life Cycle (SDLC)
- ภาพรวมตลาดโมเดลปี 2026: Claude (Opus 4.8 / Sonnet 4.6 / Haiku 4.5), GPT-5.5, Gemini 3.5 และโมเดล Open-source
- ความต่างของ Rule-based, LLM-based และ Reasoning/Agentic Model
- เลือกโมเดลให้เหมาะกับงาน: ความสามารถ ต้นทุน (cost/token) ความเร็ว Context Window และความเป็นส่วนตัวของข้อมูล

2 เข้าใจ LLM และ Transformer เชิงลึก

- สถาปัตยกรรมเบื้องหลัง LLM: Transformer, Tokenization และ Context Window
- Reasoning Model และ “Thinking” คืออะไร ต่างจากรุ่นก่อนอย่างไร
- RLHF และ Post-training ทำให้ AI ทำตามคำสั่งและปลอดภัยขึ้นได้อย่างไร
- เข้าใจ Hallucination และเหตุผลเชิงเทคนิคที่ AI ตอบผิด

3 จาก Prompt Engineering สู่ Context Engineering สำหรับนักพัฒนา

- หลักการ Prompt ที่ดี: Clear, Specific, Role-based, Few-shot และโครงสร้าง Prompt สำหรับงาน Dev (Coding, Debugging, Explain, Refactor, Unit Test)
- Context Engineering: จัดการบริบท Knowledge File และ System Prompt และเขียนไฟล์มาตรฐาน AGENTS.md / CLAUDE.md
- Workshop วันที่ 1: ติดตั้งและตั้งค่า ChatGPT, Claude, Gemini + API Key, ลอง Generate/Refactor/Explain เปรียบเทียบหลายโมเดล และเขียนไฟล์ AGENTS.md แรกของคุณ

DAY 2

Agentic Coding, MCP & Software Design

เนื้อหาการอบรม · Section 4–7

รวม
6 ชั่วโมง

4 AI Coding Assistant & Agentic Coding

- สามรูปแบบของเครื่องมือ AI Coding: Inline Suggestion → Chat Assistant → Autonomous Agent
- เครื่องมือ Agentic Coding ปี 2026: Claude Code, Cursor, GitHub Copilot Agent Mode, Windsurf, OpenAI Codex
- ใช้ Agent วางแผนและสร้างฟีเจอร์แบบ end-to-end (Plan → Edit → Run → Verify) พร้อมเทคนิคกำกับ Agent และรีวิว Diff

5 MCP — เชื่อม AI เข้ากับเครื่องมือและข้อมูลจริง

- Model Context Protocol (MCP) คืออะไร ทำไมถูกเรียกว่า “USB-C ของโลก AI”
- สถาปัตยกรรม MCP: Host, Client, Server และการสื่อสารผ่าน JSON-RPC
- เชื่อม AI เข้ากับ GitHub, Database, File System และเครื่องมือภายในองค์กร พร้อมแนวทาง Security / Access Control

6 AI สำหรับ Software Design & System Architecture

- Generate UML, ERD, Sequence และ Architecture Diagram จาก Prompt (Mermaid / PlantUML)
- ใช้ AI ออกแบบ API แบบ RESTful / GraphQL และร่าง OpenAPI Spec รวมถึงประเมินทางเลือกเชิงสถาปัตยกรรม (Trade-off)

7 Legacy Code, Refactoring & AI Code Review

- ให้ AI อธิบายโค้ดเก่า (Explain Code), ทำ Reverse Engineering, Refactor และเพิ่ม Comment/Documentation
- AI Code Review: ตรวจหา Bug, ช่องโหว่ความปลอดภัย และปัญหาคุณภาพโค้ดก่อน Merge
- Workshop วันที่ 2: กำกับ Agent สร้างระบบ API + Database Schema + Diagram, ติดตั้ง MCP Server (GitHub/Filesystem) และให้ Agent อ่าน Legacy Code แล้ว Refactor + Review

DAY 3

AI Prototyping, Automation, Testing & Best Practices

เนื้อหาการอบรม · Section 8–11

รวม
6 ชั่วโมง

8 AI Prototyping & UI Generation

- ใช้ AI สร้าง UI ด้วย HTML, Tailwind CSS และ Component สำเร็จรูป
- เครื่องมือ Generate เว็บ/UI จาก Prompt ปี 2026: v0, Lovable, Bolt, Framer AI
- Generate เว็บ Static/Full-stack และ Deploy บน GitHub Pages / Vercel / Netlify พร้อมปรับแต่งโค้ดให้ใช้งานจริง

9 AI-powered Chatbot, RAG & Assistant (เกริ่นนำ)

- แนวคิด RAG (Retrieval-Augmented Generation) และเหตุผลที่จำเป็นต่อ Chatbot ระดับองค์กร
- สร้าง Chatbot ตอบคำถามจากเอกสาร/Knowledge Base และ FAQ Bot / Coding Assistant ด้วย Claude / ChatGPT
- ภาพรวมการต่อยอดด้วย Vector Database และ Embeddings (ปูทางสู่วันที่ 4)

10 AI Testing, Automation & Spec-driven Development

- ใช้ AI เขียน Unit Test, Integration Test และสร้าง Test Data
- Spec-driven Development: เขียน Spec/Requirement ชัดเจน แล้วให้ AI สร้างและตรวจงานตาม Spec พร้อมเชื่อม CI/CD

11 Best Practices, Security & Governance (สำหรับนักพัฒนา)

- ผสาน AI เข้ากับ Dev Lifecycle: Plan → Build → Test → Review → Deploy พร้อมข้อควรระวังด้าน Security, Hallucination, Privacy, Copyright
- Workshop วันที่ 3: สร้างเว็บแอป/UI จาก Prompt แล้ว Deploy ขึ้น Hosting จริง, ทำ RAG Chatbot เบื้องต้น และรวบรวม Prompt Library ส่วนตัว

DAY 4

AI Engineering: Building AI-Powered Applications

เนื้อหาการอบรม · Section 12–15

รวม
6 ชั่วโมง

12 LLM API & Application Integration

- พื้นฐานการเรียก LLM ผ่าน API: Anthropic, OpenAI และ Gemini
- พารามิเตอร์สำคัญ: Temperature, Max Tokens, System Prompt, Structured Output (JSON Mode)
- ออกแบบ Prompt Template, จัดการ Context Window, Streaming, Error Handling, Rate Limit และต้นทุน Token

13 Production-grade RAG

- สถาปัตยกรรม RAG เต็มรูปแบบ: Ingest → Chunk → Embed → Store → Retrieve → Generate
- Embeddings และ Vector Database (pgvector, Qdrant, Pinecone, Chroma)
- ยกระดับคุณภาพการค้นคืน: Chunking Strategy, Hybrid Search, Re-ranking และลด Hallucination ด้วย Citation/Grounding

14 Tool/Function Calling & Building AI Agents

- Tool / Function Calling: ให้ LLM เรียก Function, API และ Database
- แนวคิด AI Agent (Planning, Tool Use, Looping) และ Pattern: single-agent, multi-agent, workflow orchestration
- ภาพรวม Framework สำหรับสร้าง Agent: LangChain, LlamaIndex, LangGraph

15 Building Your Own MCP Server

- ก้าวจาก “ผู้ใช้” สู่ “ผู้สร้าง” MCP Server เพื่อเปิดเครื่องมือ/ข้อมูลองค์กรให้ AI Agent พร้อมออกแบบ Tools, Resources, Prompts และ Security
- Workshop วันที่ 4: เขียนโปรแกรมเรียก LLM API แบบ Structured Output, สร้างระบบ RAG ตอบจากเอกสารของผู้เรียน, สร้าง AI Agent ที่เรียก Tool และพัฒนา MCP Server อย่างง่าย

DAY 5

Production, LLMOps, Evaluation & Capstone

เนื้อหาการอบรม · Section 16–19

รวม
6 ชั่วโมง

16 Evaluation & Testing of AI Systems

- ทำไมระบบ AI ทดสอบยากกว่าโปรแกรมทั่วไป (Non-deterministic)
- ออกแบบ Eval Set และ Metric สำหรับ LLM/RAG, ใช้ LLM-as-a-Judge และ Regression Testing เมื่อเปลี่ยนโมเดล/Prompt

17 LLMOps & Observability

- LLMOps Lifecycle: Prompt/Model Versioning, Deployment และ Monitoring
- ติดตาม Cost, Latency และคุณภาพคำตอบใน Production พร้อม Logging, Tracing, Observability
- เทคนิคลดต้นทุนและเพิ่มประสิทธิภาพ: Caching, Routing, Model Selection

18 AI Security, Guardrails & Responsible AI

- ภัยเฉพาะ LLM: Prompt Injection, Jailbreak, Data Leakage (อ้างอิง OWASP Top 10 for LLM)
- วาง Guardrails: Input/Output Validation, Content Filtering, Policy และการจัดการ PII
- หลัก Responsible AI: Bias, Transparency และ Accountability

19 Deployment & Capstone Project

- แนวทาง Deploy ระบบ AI: Serverless, Container, Managed Platform และการออกแบบให้ Scalable/Reliable
- Capstone Project: สร้างแอป AI แบบ end-to-end (ออกแบบ → สร้าง RAG/Agent → ประเมินผล → เพิ่ม Guardrails → Deploy) แล้วนำเสนอผลงาน

สนใจจัดอบรม IN-HOUSE · ขอใบเสนอราคา

ขอใบเสนอราคา

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

เพิ่มเพื่อน LINE