สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

COURSE ID
AIC-113

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

AI / FORWARD DEPLOYED ENGINEER

Forward Deployed Engineer (FDE) 201

จาก AI Pilot สู่ระบบ Production ที่องค์กรวางใจได้

- 18 ชั่วโมง
- 3 วัน
- พื้นฐาน FDE 101, Python และ Docker
- ระดับ Advanced



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

Pilot ด้าน AI ที่ผ่านการสาธิตแล้วยังห่างจากระบบที่องค์กรยอมเปิดให้พนักงานใช้จริงอีกหลายขั้น คำตอบต้องวัดคุณภาพได้ทุกครั้งที่แก้โค้ด Agent ต้องเห็นเฉพาะข้อมูลและเครื่องมือที่ผู้ใช้คนนั้นมีสิทธิ์ ข้อมูลส่วนบุคคลต้องไม่หลุดไปกับ Prompt หรือ Log และฝ่ายบริหารต้องเห็นต้นทุนกับผลตอบแทนเป็นตัวเลข งานช่วงนี้คือหัวใจของ Forward Deployed Engineer (FDE) ที่ต้องพาระบบจาก Pilot ไปถึงวัน Go-live ด้วยโค้ดที่ทีมลูกค้ารับไปดูแลต่อได้

หลักสูตรนี้เป็นเส้นทางสาย Code-first ต่อจาก FDE 101 โดยทบทวนสั้น ๆ แล้วลงมือทันที วันแรกสร้าง RAG Service ด้วย Python และ FastAPI ตั้งแต่ Ingestion, Hybrid Search, Reranking และ Structured Output ไปจนถึง Evaluation Harness ที่รันใน CI วันที่สองยกระดับ Agent สู่มาตรฐานองค์กรด้วย MCP Server, A2A, Workflow ที่มีคนอนุมัติ, Guardrails, การปกปิด PII, SSO, RBAC และ Audit Log วันที่สามลดต้นทุนและ Latency วาง Tracing, CI/CD และ Container แล้วปิดงานแบบที่ปรึกษาด้วยแผน UAT, Business Case และการส่งมอบผู้บริหาร พร้อม Capstone ที่เก็บเป็นผลงานได้ (อบรม 3 วัน วันละ 6 ชั่วโมง รวม 18 ชั่วโมง ระดับ Advanced)

เครื่องมือ

- FDE 201
- Py Python & FastAPI
- RAG Production RAG & Evals
- MCP MCP & A2A Agents
- Sec Guardrails & RBAC

ราคาค่าอบรม · PUBLIC / IN-HOUSE

6,900 บาท / ท่าน

สมัครผ่านเว็บไซต์ ลด 10%

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide โค้ดตัวอย่างครบทุก Lab เทมเพลต UAT และ Business Case และ Certificate · รับจัด In-house ขอใบเสนอราคาได้

เพิ่มเพื่อน LINE

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

01 วัตถุประสงค์ Objectives

- วางโครง AI Service ด้วย Python และ FastAPI ที่ทดสอบ แพ็ก และส่งต่อให้ทีมลูกค้าได้
- สร้าง RAG ระดับ Production ด้วย Ingestion ที่ดี Hybrid Search, Reranking และ Structured Output
- สร้าง Evaluation Harness พร้อม Golden Set และใช้เป็นด่านตรวจ Regression ใน CI
- เชื่อม Agent กับระบบภายในผ่าน Tool Calling, MCP Server และ A2A อย่างมีขอบเขต
- ออกแบบ Workflow ที่ทำงานนาน ทำต่อได้เมื่อสะดุด และมีจุดให้คนอนุมัติ
- ป้องกัน Prompt Injection ปกปิด PII และคุมสิทธิ์ด้วย SSO, RBAC และ Audit Log
- ลดต้นทุนและ Latency และติดตามระบบด้วย Tracing ตามแนวทาง OpenTelemetry
- วางแผน UAT ทำ Business Case และส่งมอบระบบให้ผู้บริหารพร้อม Production Package

02 กลุ่มเป้าหมาย Target Audience

- ผู้ที่ผ่าน FDE 101 หรือทำ AI Pilot มาแล้ว และต้องพาระบบขึ้น Production ในองค์กร
- AI Engineer และ Backend Developer ที่สร้าง RAG หรือ Agent ด้วย Python
- Solution Engineer และที่ปรึกษาด้านเทคนิคที่ต้องส่งมอบระบบ AI ให้ลูกค้าองค์กร
- Tech Lead และ Platform Engineer ที่ดูแลความปลอดภัย การวัดผล และต้นทุนของระบบ AI
- ทีม IT ภายในที่ต้องรับระบบ AI ต่อจากผู้พัฒนาและดูแลหลัง Go-live

03 จุดเด่นของหลักสูตร Highlights

- **Code-first ทั้ง 3 วัน** Python, FastAPI, Docker และ CI
- **RAG ที่วัดผลได้** Hybrid Search, Reranking และ Eval
- **Agent ระดับองค์กร** MCP, A2A และคนอนุมัติในขั้นสำคัญ
- **ปลอดภัยตั้งแต่ออกแบบ** Guardrails, PII, SSO, RBAC และ Audit
- **ปิดงานแบบที่ปรึกษา** UAT, ROI และส่งมอบผู้บริหาร

04 ความรู้พื้นฐาน Prerequisites

- ผ่านหลักสูตร Forward Deployed Engineer (FDE) 101 หรือมีประสบการณ์สร้าง AI Pilot ที่เทียบเท่า
- เขียน Python ได้คล่อง เข้าใจ REST API, JSON และการใช้ Virtual Environment
- ใช้ Git, Terminal และ Docker ขั้นพื้นฐานได้
- มี API Key ของผู้ให้บริการโมเดลของตนเองหรือของบริษัท หรือรันโมเดลบนเครื่องด้วย Ollama ได้

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook (macOS, Windows หรือ Linux) แนะนำ RAM 16 GB ขึ้นไป พร้อมสิทธิ์ติดตั้งโปรแกรม
ภาษาและ Framework	Python, FastAPI, Pydantic และ pytest ใน VS Code หรือ Editor ที่ถนัด
โมเดล AI	API Key ของผู้ให้บริการโมเดลที่ผู้เรียนเลือกเอง หรือ Open Model ขนาดเล็กผ่าน Ollama
ข้อมูลและการค้นหา	Vector Database แบบ Open Source เช่น PostgreSQL กับ pgvector หรือ Qdrant บน Docker
Production	Docker Desktop, Git, CI แบบฟรีของผู้เรียน เช่น GitHub Actions และ OpenTelemetry

หมายเหตุ: สถาบันไม่ได้จัดเตรียมบัญชี API Key, Cloud Credit, GPU หรือ Server ให้ ผู้เรียนใช้ API Key ของผู้ให้บริการโมเดลของตนเองหรือของบริษัท (มีค่าใช้จ่ายตามการใช้งานจริง ซึ่งใน Lab มักไม่สูง) หรือรัน Open Model ขนาดเล็กบน Notebook ด้วย Ollama แทนได้ ส่วน CI และ Cloud ใช้ Free Tier ของผู้เรียนเอง หลักสูตรนี้ต่อยอดจาก Forward Deployed Engineer (FDE) 101 ที่เน้น Discovery, Prototype ด้วย Claude Code, Multi-Agent Architecture, Local AI และ Handover โดยทบทวนสั้น ๆ แล้วเน้นงาน Production แบบเขียนโค้ด เนื้อหา RAG ต่อยอดจาก RAG & Knowledge Base และเรื่อง Gateway, Routing และ Guardrail ระดับแพลตฟอร์มมีรายละเอียดเต็มใน Enterprise GenAI Gateway and Private AI Platform ส่วน GraphRAG, Fine-tuning และ Red Teaming เจาะลึกใน GraphRAG and Agent Memory, LLM Fine-Tuning with LoRA และ AI Security and LLM Red Teaming ชื่อเครื่องมือ เวอร์ชันของ Protocol และฟีเจอร์เปลี่ยนบ่อย เนื้อหาจะอิงเวอร์ชันล่าสุดในวันอบรม

06 ภาพรวมการเรียน 3 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-5	วางโครง Service ด้วย Python และ FastAPI, ทำ Ingestion และ Chunking กับเอกสารจริง, Hybrid Search กับ Reranking, Structured Output และ Context Engineering แล้วสร้าง Evaluation Harness ที่รันใน CI
Day 2 Section 6-10	Tool Calling ที่เชื่อถือได้, MCP Server สำหรับระบบภายใน, A2A และ Workflow ที่ทำงานนานพร้อมจุดอนุมัติ, Guardrails กับ Prompt Injection และ PII, SSO, RBAC และ Audit Log
Day 3 Section 11-15	ลดต้นทุนและ Latency ด้วย Caching และ Model Routing, Tracing ด้วย OpenTelemetry, CI/CD และ Container, แผน UAT, ROI และการส่งมอบผู้บริหาร ปิดท้ายด้วย Capstone ยกระดับ Pilot สู่ Production

DAY 1

Production RAG พร้อมระบบวัดผล

เนื้อหาการอบรม · Section 1-5

รวม
6 ชั่วโมง

1 จาก Pilot สู่ Production และโครง Service ด้วย FastAPI

- ทบทวนสั้น ๆ จาก FDE 101 และช่องว่างระหว่าง Pilot กับระบบที่องค์กรยอมเปิดใช้
- โครงสร้างโปรเจกต์ Python ที่ทีมลูกค้ารับไปดูแลต่อได้: Layer, Config และการเก็บ Secret
- FastAPI กับ Pydantic: Endpoint, Validation, Async และ Streaming Response
- Health Check, Error Handling มาตรฐาน และ Test ด้วย pytest ตั้งแต่วันแรก
- Lab: วางโครง AI Service พร้อม Dockerfile และ Test ชุดแรก

2 Lab: Ingestion และ Chunking กับเอกสารจริง

- แยกข้อความจาก PDF, Office, ตาราง และหน้าเว็บ พร้อมรับมือภาษาไทยและเอกสารสแกน
- เลือกวิธี Chunking: ตามขนาด ตามโครงสร้างเอกสาร หรือตามความหมาย และผลต่อคุณภาพคำตอบ
- เก็บ Metadata ที่ใช้ต่อได้: แหล่งที่มา วันที่ หน่วยงาน และระดับสิทธิ์ของเอกสาร
- Re-index แบบเพิ่มเฉพาะส่วนที่เปลี่ยน และลบเอกสารที่ถูกยกเลิกออกจาก Index
- Lab: สร้าง Ingestion Pipeline ที่รันซ้ำได้จากชุดเอกสารตัวอย่าง

3 Lab: Hybrid Search และ Reranking

- จุดอ่อนของ Vector Search อย่างเดียวกับรหัสสินค้า ชื่อเฉพาะ และคำย่อ
- รวม Keyword Search (BM25) กับ Vector Search และรวมผลด้วย Reciprocal Rank Fusion
- Reranker ช่วยเรียงผลใหม่ให้แม่นขึ้น และราคาที่ต้องจ่ายด้าน Latency
- กรองผลด้วย Metadata และสิทธิ์ของผู้ใช้ก่อนส่งเข้าโมเดล
- Lab: เปรียบเทียบ Vector, Hybrid และ Hybrid กับ Reranker บนคำถามชุดเดียวกัน

4 Structured Output และ Context Engineering

- บังคับคำตอบเป็น JSON ตาม Schema ตรวจด้วย Pydantic และจัดการเมื่อโมเดลตอบผิดรูปแบบ
- จัด Context ให้คุ้ม: ลำดับข้อมูล งบ Token การอ้างอิงแหล่งที่มา และการตอบว่าไม่ทราบ
- เก็บ Prompt เป็นไฟล์มีเวอร์ชัน และทดสอบก่อนเปลี่ยนทุกครั้ง
- จุดตัดสินใจ: เมื่อไรควรใช้ Multimodal RAG, GraphRAG หรือ Fine-tuning แทนการปรับ RAG
- Lab: เพิ่ม Endpoint ที่ตอบเป็น JSON พร้อมแหล่งอ้างอิงที่ตรวจสอบได้

5 Lab: Evaluation Harness และ Regression ใน CI

- สร้าง Golden Set จากคำถามจริงของผู้ใช้ร่วมกับผู้เชี่ยวชาญฝั่งลูกค้า
- วัดฝั่ง Retrieval (Recall@k, MRR) และฝั่งคำตอบ (ความถูกต้อง ความสอดคล้องกับแหล่งข้อมูล)
- LLM-as-a-Judge: ใช้อย่างไรให้เชื่อถือได้ และเทียบผลกับคนตรวจเป็นระยะ
- ใช้ Library อย่าง Ragas หรือเขียน Metric เองด้วย pytest แล้วตั้งเกณฑ์ผ่าน
- Lab: ต่อ Eval เข้า CI ให้ Pull Request ที่ทำคะแนนตกถูกตีกลับ

DAY 2

Agent ในองค์กร: เชื่อมระบบอย่างปลอดภัย

เนื้อหาการอบรม · Section 6-10

รวม 6 ชั่วโมง

6 Lab: Tool Calling ที่เชื่อถือได้

- ออกแบบ Tool ให้โมเดลเลือกใช้ถูก: ชื่อ คำอธิบาย Schema และตัวอย่าง
- ตรวจ Argument ก่อนเรียกจริง ตั้ง Timeout, Retry และทำ Tool ให้เรียกซ้ำได้อย่างปลอดภัย
- แยก Tool อ่านข้อมูลออกจาก Tool ที่เปลี่ยนแปลงข้อมูล และจำกัดจำนวนรอบของ Agent
- ส่ง Error กลับให้โมเดลแก้ตัวเอง แทนการหยุดทั้งงาน
- Lab: สร้าง Agent ที่เรียก API ภายในหลายตัวพร้อม Test ของแต่ละ Tool

7 Lab: MCP Server สำหรับระบบภายในองค์กร

- ต่อยอดจาก MCP ใน FDE 101 สู่ Server ที่ใช้ร่วมกันหลายทีมผ่าน HTTP
- ออกแบบ Tools, Resources และ Prompts ให้เหมาะกับระบบ เช่น Ticket, HR หรือ ERP
- Authorization ของ MCP แบบ OAuth และส่งตัวตนผู้ใช้ไปถึงระบบปลายทาง
- Versioning, Logging และการทดสอบ MCP Server ก่อนเปิดให้ Agent ใช้
- Lab: สร้าง MCP Server ห่อระบบภายในจำลองที่เคารพสิทธิ์รายผู้ใช้

8 A2A และ Workflow ที่ทำงานนานพร้อมคนอนุมัติ

- Agent2Agent (A2A) Protocol: Agent Card, Task และเมื่อไรควรใช้คู่กับ MCP
- Workflow ที่ใช้เวลาเป็นชั่วโมงหรือเป็นวัน: Checkpoint, ทำต่อเมื่อสะดุด และกันงานซ้ำ
- Human-in-the-loop: หยุดรอการอนุมัติก่อนขั้นที่มีผลจริง เช่น จ่ายเงินหรือแก้ข้อมูลลูกค้า
- แจ้งสถานะความคืบหน้าให้ผู้ใช้และระบบอื่นติดตามได้
- Lab: Workflow อนุมัติคำขอที่หยุดรอคนตัดสินใจแล้วทำต่อได้

9 Lab: Guardrails, Prompt Injection และ PII

- Guardrails ฝั่ง Input และ Output: หัวข้อต้องห้าม รูปแบบคำตอบ และการตรวจข้อเท็จจริง
- Prompt Injection แบบอ้อมที่แฝงมากับเอกสาร หน้าเว็บ และผลลัพธ์ของ Tool
- ตรวจและปกปิด PII ก่อนส่งเข้าโมเดลและก่อนเขียน Log ตามหลัก PDPA
- Sandbox สำหรับรันโค้ดที่ Agent สร้าง: Container แยก จำกัดเครือข่ายและไฟล์
- Lab: ทดสอบโจมตีระบบของตัวเองแล้วเพิ่มชั้นป้องกันจนผ่าน

10 Lab: SSO, RBAC และ Audit Log

- เชื่อม SSO ขององค์กรผ่าน OpenID Connect และส่งตัวตนผู้ใช้ไปทุกชั้นของระบบ
- RBAC กำหนดว่าบทบาทใดเรียก Tool ใดได้ และเห็นเอกสารระดับใด
- Retrieval ที่เคารพสิทธิ์: กรองเอกสารตามสิทธิ์ของผู้ถามก่อนเข้าถึงโมเดล
- Audit Log บันทึกว่าใคร ถามอะไร Agent เรียก Tool ใด และผลเป็นอย่างไร
- Lab: ทำให้ผู้ใช้สองบทบาทได้คำตอบและสิทธิ์ต่างกันจากระบบเดียวกัน

DAY 3

Go-live และการปิดงานแบบที่ปรึกษา

เนื้อหาการอบรม · Section 11-15

รวม 6 ชั่วโมง

11 Lab: ลดต้นทุนและ Latency

- แยกต้นทุนต่อคำขอ: Token เข้า Token ออก Embedding Reranker และ Tool
- Prompt Caching ของผู้ให้บริการ และ Response Cache ที่ต้องระวังเรื่องสิทธิ์ผู้ใช้
- Model Routing: งานง่ายใช้โมเดลเล็ก งานยากใช้โมเดลใหญ่ พร้อม Fallback
- Batch สำหรับงานที่ไม่ต้องตอบทันที และ Streaming เพื่อลดเวลารอที่ผู้ใช้รู้สึก
- Lab: วัดต้นทุนและ Latency ก่อนและหลังปรับ โดยคุมคะแนน Eval ไม่ให้ตก

12 Lab: Observability และ Tracing

- Trace ทั้งเส้นทางของคำขอ: Retrieval, การเรียกโมเดล, Tool และ Agent แต่ละขั้น
- OpenTelemetry และ GenAI Semantic Conventions สำหรับตั้งชื่อ Span และ Attribute
- Metric ที่ต้องมี: Latency, Token, ต้นทุน, อัตรา Error และ Feedback จากผู้ใช้
- Online Evaluation จาก Traffic จริง และ Alert เมื่อคุณภาพหรือต้นทุนผิดปกติ
- Lab: ส่ง Trace ไปยัง Backend แบบ Open Source แล้วไล่หาคำขอที่ช้าที่สุด

13 Lab: CI/CD และ Container สำหรับระบบ AI

- Multi-stage Docker Image และ Docker Compose สำหรับ Service, Vector Database และ Worker
- Pipeline: Test, Eval Gate, Security Scan แล้วจึง Build และ Deploy
- ทำเวอร์ชันให้ครบ: โค้ด Prompt โมเดล และ Index ย้อนกลับได้พร้อมกัน
- ปล่อยแบบค่อยเป็นค่อยไปให้ผู้ใช้บางกลุ่มก่อน พร้อมเกณฑ์ Rollback
- Lab: สร้าง Pipeline ที่ Build, ทดสอบ และปล่อย Service เวอร์ชันใหม่

14 UAT, ROI และ Executive Handover

- แผน UAT ร่วมกับผู้ใช้ฝั่งธุรกิจ: Test Scenario, เกณฑ์ผ่าน และการเก็บ Feedback
- Business Case: ต้นทุนต่อเดือน เวลาที่ประหยัดได้ และสมมติฐานที่ต้องวัดซ้ำหลังใช้งาน
- Executive Readout: เล่าผล Pilot ความเสี่ยงที่เหลือ และข้อเสนอขยายผลให้ผู้บริหารตัดสินใจ
- Operating Model หลัง Go-live: ใครดูแลข้อมูล Prompt Eval และค่าใช้จ่าย
- Workshop: เขียน UAT Plan และ Business Case หนึ่งหน้าของโจทย์ในคลาส

15 Capstone: ยกระดับ Pilot สู่ Production-ready Package

- รับ Pilot ตั้งต้นที่ยังไม่พร้อมใช้ แล้วตรวจหาช่องโหว่ด้านคุณภาพ ความปลอดภัย และต้นทุน
- ปรับปรุงด้วยงานทั้ง 3 วัน: Eval ใน CI, สิทธิ์รายผู้ใช้, Guardrails, Tracing และ Container
- รวมเป็นชุดส่งมอบ: โค้ด ผล Eval, Architecture, Runbook, UAT Plan และ Business Case
- นำเสนอต่อผู้บริหารจำลอง ตอบคำถามเชิงเทคนิค และรับ Feedback จากวิทยากร
- เก็บชุดส่งมอบไว้เป็นผลงาน (Portfolio) สำหรับบทบาท FDE

07 เรียนจบแล้วได้อะไร Outcomes

- **สร้าง RAG Service ได้** FastAPI, Hybrid Search และ Reranking
- **วัดคุณภาพได้ทุกครั้ง** Golden Set และ Eval Gate ใน CI
- **เชื่อม Agent กับองค์กรได้** Tool Calling, MCP และ A2A
- **คุมความปลอดภัยได้** Guardrails, PII, SSO และ RBAC
- **คุมต้นทุนและติดตามได้** Caching, Routing และ Tracing
- **ปิดงานกับลูกค้าได้** UAT, Business Case และ Handover

08 สิ่งที่ผู้เรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ครบทั้ง 3 วัน และ Lab Guide ทีละขั้นตอน
- โค้ดตัวอย่าง FastAPI Service, Ingestion Pipeline, Eval Harness, MCP Server และไฟล์ CI/CD ครบทุก Lab
- เทมเพลต Golden Set, Security Checklist, UAT Plan, Business Case และ Executive Readout
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สามิตร โกยม ผู้สอนและวิศวกรที่พัฒนาและดูแลระบบ AI ที่ใช้งานจริงในองค์กร ทั้ง RAG, Agent ที่เชื่อมกับระบบภายใน และงานด้านความปลอดภัยกับการวัดผล จุดเด่นคือการสอนแบบเขียนโค้ดจริงบน Pilot ตั้งต้นชุดเดียวที่ค่อย ๆ ยกระดับตลอด 3 วัน ทุกการปรับต้องพิสูจน์ด้วยตัวเลขจาก Eval, Trace และต้นทุน และฝึกเล่าผลให้ผู้บริหารฟังในภาษาธุรกิจ เพื่อให้ผู้เรียนกลับไปพาโปรเจกต์ AI ขององค์กรข้ามจาก Pilot ไปถึง Go-live ได้อย่างมั่นใจ

ราคาค่าอบรม 6,900 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

6,900 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

02-570-8449 · 088-807-9770

@itgenius · contact@itgenius.co.th

เพิ่มเพื่อน LINE