

หลักสูตร "พัฒนา AI/LLMs App แบบ On-Premise สำหรับใช้ในองค์กร"

วันที่ 26-30 มกราคม 2569 เวลา 09.00-16.00 น.

(5 วันเต็ม)

วัตถุประสงค์ของหลักสูตร

- เพื่อให้ผู้เรียนเข้าใจแนวคิด ประโยชน์ และการวางแผนพัฒนา AI/LLMs Application แบบ On-Premise สำหรับองค์กร รวมถึงการเลือกใช้ LLMs Open Source และเครื่องมือแบบ Local เช่น Ollama ให้เหมาะสมกับงานและทรัพยากร
- เพื่อให้ผู้เรียนพัฒนา AI Chat Bot, AI Agent, Multi-Agent System และ Workflow ด้วย LangChain.js ได้อย่างมืออาชีพ
- เพื่อให้ผู้เรียนสร้างระบบ RAG และ Agentic RAG ที่รองรับเอกสารหลายรูปแบบ (Text, Table, Image) พร้อมจัดการ Vector Database สำหรับการค้นหาข้อมูลภายในองค์กร
- เพื่อให้ผู้เรียนใช้งาน Tool Calling และสร้าง MCP (Model Context Protocol) Server เพื่อเชื่อมต่อระบบภายในองค์กร (API, Database, LDAP) ได้อย่างปลอดภัยและมีประสิทธิภาพ
- เพื่อให้ผู้เรียนออกแบบ Prompt Engineering และ Context Engineering ที่แม่นยำ เหมาะสมกับบริบทองค์กร และรองรับการประมวลผลแบบ Streaming
- เพื่อให้ผู้เรียนสร้าง Chat UI ด้วย Next.js พร้อมรองรับ Real-time Streaming, Authentication, และ Role-Based Access Control (RBAC)
- เพื่อให้ผู้เรียนเข้าใจและป้องกันความเสี่ยงด้าน AI Security รวมถึงการสร้าง Guardrails (PII Detection, Prompt Injection Prevention, Output Moderation เป็นต้น)
- เพื่อให้ผู้เรียนจัดการค่าใช้จ่ายและเพิ่มประสิทธิภาพระบบด้วยเทคนิค Cost Optimization, Semantic Caching, Fallback Models และ Prompt Compression
- เพื่อให้ผู้เรียนสร้าง CI/CD Pipeline สำหรับการพัฒนา ทดสอบ และอัปเดต AI App อย่างต่อเนื่องและปลอดภัย
- เพื่อให้ผู้เรียนจัดการ Error Handling, Retry Logic, Circuit Breaker Pattern และ Fallback Strategy เพื่อรักษาความเสถียรของระบบ AI ใน Production
- เพื่อให้ผู้เรียน Deploy AI Application แบบ Production-grade ด้วย Docker, vLLM, AI Gateway (LiteLLM Proxy) และเซิร์ฟเวอร์ในองค์กร
- เพื่อให้ผู้เรียนติดตั้ง Observability Stack (Langfuse, Monitoring, Logging) สำหรับติดตามผลประเมินประสิทธิภาพ และทำ A/B Testing
- เพื่อให้ผู้เรียนสามารถประยุกต์ใช้ ถ่ายทอดองค์ความรู้ และสร้าง Use Case ในองค์กรตนเอง เช่น HR Bot, IT Support Bot, Policy QA Bot ได้อย่างเป็นรูปธรรม

พื้นฐานผู้เรียน

- เคยเขียนโปรแกรมด้วยภาษาใดภาษานึ่งมาก่อน เช่น JavaScript เป็นต้น

เนื้อหาการเรียนและตารางเรียน 5 วัน

เวลา	เนื้อหา
วันที่ 1	
09.00 – 10.30 น.	LLMs & On-Premise AI Strategy - ติดตั้ง และตั้งค่าโปรแกรมต่างๆ ที่เกี่ยวข้อง - แนวคิดและประโยชน์ของการติดตั้ง AI แบบ On-Premise - สถาปัตยกรรม LLM และประเภทของโมเดล - การเลือกขนาดโมเดล: 7B, 13B, 70B – ข้อดีข้อเสียของแต่ละขนาด - Quantization (การบีบอัดโมเดล) - GGUF, GPTQ, AWQ - ความต้องการฮาร์ดแวร์และการวางแผนค่าใช้จ่าย - ข้อดี ข้อเสีย ของ Cloud vs. On-Premise
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	Setup Local LLM Environment - การติดตั้ง Ollama และจัดการโมเดลต่างๆ - การใช้งาน Ollama - การตั้งค่าโมเดลและปรับแต่งประสิทธิภาพ - Lab: ติดตั้งและทดสอบโมเดลหลายตัว
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	LangChain.js Core Concepts - สถาปัตยกรรม: Agents, Models, Prompts , Messages, Memory, Tools - การเปรียบเทียบ Chat Models กับ LLMs ทั่วไป - Output Parsers (ตัวแปลงผลลัพธ์) - JSON, Structured Output เป็นต้น - Conversation Memory และการจัดการ Context Window - Lab: สร้างแชทบอทพลิเคชันอย่างง่าย พร้อมระบบจดจำบทสนทนา
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	Prompt Engineering & Context Engineering - รูปแบบการออกแบบ Prompt (Zero-shot, Few-shot, Chain-of-Thought) - Context Engineering สำหรับองค์กร - การจัดการ Token และควบคุมค่าใช้จ่าย

เวลา	เนื้อหา
	- Prompt Templates และการใช้ตัวแปร - Workshop
วันที่ 2	
09.00 – 10.30 น.	AI Agent Fundamentals - สถาปัตยกรรม Agent และ ReAct Pattern (Reasoning + Acting) - Tool Calling และ Function Calling (การเรียกใช้ฟังก์ชัน) - ประเภทของ Agent (ReAct, OpenAI Functions, Structured Chat) - การสร้าง Custom Tools (เครื่องมือที่กำหนดเอง) - Lab: สร้าง Agent ที่มีเครื่องมือและค้นหาข้อมูลจาก API
10.30 – 10.45 น.	พักเบรก
10.45 -12.00 น.	Advanced Tool Calling & MCP Server - ภาพรวม MCP (Model Context Protocol) - สร้าง MCP Server เพื่อเชื่อมต่อ API ภายในองค์กร - การเชื่อมต่อฐานข้อมูล (PostgreSQL, MySQL) - การเชื่อมต่อ Authentication - การแสดงผล UI แบบโต้ตอบได้ - Lab: สร้าง MCP Server สำหรับระบบองค์กร
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	Multi-Agent Systems - แนวคิด Multi-Agent และ LangChain.js - การเลือก Pattern - การใช้งาน และ flow ของ Tool calling - Lab: สร้าง Workflow ที่ Agent ทำงานเป็นลำดับ
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	Complex Multi-Agent Workflow - รูปแบบ Hierarchical Agent (Manager-Worker) - การประมวลผลแบบขนาน และรวมผลลัพธ์ - การสื่อสารระหว่าง Agent และจัดการสถานะ - Workshop: สร้าง Bot ที่มีผู้เชี่ยวชาญหลายด้าน
วันที่ 3	
09.00 – 10.30 น.	RAG Architecture & Vector Databases แนวคิด RAG และการป้องกันการสร้างข้อมูลเท็จ

เวลา	เนื้อหา
	• การเลือก LLM ประเภท Embeddings • Embeddings (การแปลงข้อความเป็นตัวเลข) และความคล้ายคลึงของ Vector • ตัวเลือก Vector Database: Qdrant, Chroma, PGVector, Supabase Vector • การติดตั้ง Qdrant (Docker) และตั้งค่า • Lab: การจัดเก็บเอกสารและค้นหาข้อมูล
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	Production RAG Implementation • Document Loaders (การโหลดเอกสาร) - PDF, DOCX, TXT, CSV • กลยุทธ์การแบ่งข้อความ (Chunking Strategies) • Metadata Filtering และ Hybrid Search (การค้นหาแบบผสม) • การเพิ่มประสิทธิภาพการค้นหา (MMR, Threshold, Reranking) • Lab: สร้าง AI Bot ตอบคำถามเอกสาร
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	Agentic RAG – Multimodal • สถาปัตยกรรม Agentic RAG • การดึงข้อมูลจากตารางและประมวลผล • การทำความเข้าใจภาพ (OCR, Vision Models) • Pipeline การประมวลผลเอกสารที่ซับซ้อน • Lab: สร้าง Bot รองรับ ข้อความ + ตาราง + รูปภาพ
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	RAG Optimization & Evaluation • การใช้ Semantic Caching (แคชเชิงความหมาย) • Query Transformation และ Expansion (การแปลงและขยายคำค้นหา) • เมตริกการประเมิน RAG (Faithfulness, Relevance) • Workshop: ปรับแต่งประสิทธิภาพ Bot
วันที่ 4	
09.00 – 10.30 น.	Next.js Chat UI Development • สถาปัตยกรรม Next.js และการใช้งานเบื้องต้น • Real-time Streaming (การส่งข้อมูลแบบเรียลไทม์) ด้วย AI SDK • คอมโพเนนต์หน้า Chat • Message History และการจัดการ Session • Lab: สร้างหน้า Chat สำหรับองค์กร

เวลา	เนื้อหา
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	Authentication & RBAC - กลยุทธ์การยืนยันตัวตน (JWT, Session) - Role-Based Access Control - RBAC (การควบคุมสิทธิ์ตามบทบาท) - การจัดการผู้ใช้และสิทธิ์การเข้าถึง - การเชื่อมต่อกับระบบ Authentication ขององค์กร - Lab: สร้างระบบ Authentication และกำหนดบทบาทผู้ใช้
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	AI Security & Guardrails - การป้องกัน Prompt Injection (การแทรก Prompt ที่เป็นอันตราย) - PII Detection และ Redaction (ตรวจจับและปิดบังข้อมูลส่วนบุคคล) - Content Moderation และ Output Filtering (กรองเนื้อหาและผลลัพธ์) - Input Validation และ Sanitization (ตรวจสอบและทำความสะอาดข้อมูลนำเข้า) - Lab: สร้างชั้นความปลอดภัยสำหรับระบบ
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	Error Handling & Resilience - กลยุทธ์การจัดการข้อผิดพลาดอย่างครอบคลุม - Retry Logic และ Exponential Backoff (ลองใหม่และเพิ่มเวลารอ) - Circuit Breaker Pattern (รูปแบบตัดวงจร) - Fallback Models และ Graceful Degradation (โมเดลสำรอง) - Workshop: สร้างระบบจัดการข้อผิดพลาดระดับ Production
วันที่ 5	
09.00 – 10.30 น.	Production Deployment Strategy - การตั้งค่า Docker และ Docker Compose - Deploy vLLM สำหรับ High-Performance Inference (ประมวลผลเร็ว) - AI Gateway ด้วย LiteLLM Proxy - Load Balancing และ Rate Limiting (การกระจายโหลดและจำกัดอัตรา) - Lab: การติดตั้งระบบทั้งหมดบนเซิร์ฟเวอร์ On-Premise
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	Cost Optimization & Performance - การติดตามและวิเคราะห์การใช้ Token - การใช้ Semantic Caching (แคชเชิงความหมาย) - เทคนิค Prompt Compression (บีบอัด Prompt)

เวลา	เนื้อหา
	- กลยุทธ์การเลือกโมเดล (Fallback Models) - Lab: สร้าง Layer การลดต้นทุน
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	Observability & Monitoring - การติดตั้ง Langfuse สำหรับ Tracing (ติดตามการทำงาน) - กลยุทธ์การทำ Logging (บันทึกข้อมูล) - Metrics และ Dashboards (แดชบอร์ดแสดงผล) - Latency, Token, Error Rate - Framework สำหรับทำ A/B Testing - Lab: ติดตั้งระบบ Observability
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	การสร้าง CI/CD และ ปิดอบรม - CI/CD Pipeline สำหรับแอป AI - การทดสอบและติดตั้งอัตโนมัติ - Code Review และแนวปฏิบัติที่ดี - ถาม-ตอบ และปิดหลักสูตร

- ติดต่อ: codingthailand@gmail.com โทร: 085-4952624 อ.เอก