



เลขที่ 2569/001

บริษัท โค딩 คอนซัลแตนท์ (ไทยแลนด์) จำกัด
(codingthailand.com)
77/269 หมู่ 20 ตำบลขามใหญ่ อำเภอเมือง
อุบลราชธานี จังหวัดอุบลราชธานี 34000

11 พฤษภาคม 2569

เรื่อง ขอเชิญบุคลากรเข้าร่วมอบรมเชิงปฏิบัติการ

เรียน

สิ่งที่ส่งมาด้วย 1. รายละเอียดหลักสูตร "พัฒนา AI/LLMs App แบบ On-Premise สำหรับใช้ในองค์กร รุ่นที่ 2"

ตามที่ทาง บริษัท โค딩 คอนซัลแตนท์ (ไทยแลนด์) จำกัด ได้จัดทำหลักสูตร "พัฒนา AI/LLMs App แบบ On-Premise สำหรับใช้ในองค์กร รุ่นที่ 2" โดยเน้นการเรียนการสอนแบบปฏิบัติจริง เพื่อให้ผู้เรียนสามารถนำความรู้เกี่ยวกับการพัฒนา AI/LLMs Application มาปรับใช้กับองค์กรได้ โดยเรียนรู้ตั้งแต่พื้นฐานจนถึงขั้นสูง นอกจากนี้ทางบริษัทยินดีให้คำแนะนำและเป็นที่ปรึกษาให้กับบุคลากรอย่างใกล้ชิดจนสามารถนำความรู้เหล่านี้ไปปรับใช้กับองค์กรอีกด้วย

ในการนี้ บริษัท โค딩 คอนซัลแตนท์ (ไทยแลนด์) จึงใคร่ขอเรียนเชิญผู้ที่มีความสนใจลงทะเบียนอบรมหลักสูตรของเรา โดยมีรายละเอียดตามเอกสารที่แนบมานี้

จึงเรียนมาเพื่อโปรดพิจารณาอนุมัติ จักขอบพระคุณยิ่ง

ขอแสดงความนับถือ

(นายเอกนรินทร์ คำคุณ)
กรรมการผู้จัดการ

บริษัท โค딩 คอนซัลแตนท์ (ไทยแลนด์) จำกัด

โทร. 085-4952624 อีเมล: codingthailand@gmail.com

ข้อมูลหลักสูตรและลงทะเบียน: <https://www.codingthailand.com/course/on-premise-ai-llm-app-for-enterprise-2>

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

วันที่ 29-30 มิ.ย. และ 1-3 ก.ค. 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
- เพื่อให้ผู้เรียนใช้งาน Tool Calling และสร้าง MCP (Model Context Protocol) Server เพื่อเชื่อมต่อระบบภายในองค์กร (RESTful API, Database) ได้อย่างปลอดภัยและมีประสิทธิภาพ
- เพื่อให้ผู้เรียนสร้างระบบ RAG และ Agentic RAG ที่รองรับเอกสารหลายรูปแบบ (Text, Docx, PDF, Image) พร้อมจัดการ Vector Database สำหรับการค้นหาข้อมูลภายในองค์กร
- เพื่อให้ผู้เรียนได้เรียนรู้หลักการ Prompt Engineering, Context Engineering, Harness Engineering เพื่อควบคุม LLM ให้เหมาะสมกับบริบทองค์กร
- เพื่อให้ผู้เรียนได้เรียนรู้หลักการของ Agent Skills และนำมาใช้กับ LLM Application ขององค์กรได้
- เพื่อให้ผู้เรียนสามารถสร้าง 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 เป็นต้น)
- เพื่อให้ผู้เรียนเรียนรู้การใช้เครื่องมือสำหรับจัดการค่าใช้จ่าย และเพิ่มประสิทธิภาพระบบด้วยเทคนิค Cost Optimization, Semantic Caching, Fallback Models และ Prompt Compression เป็นต้น
- เพื่อให้ผู้เรียนจัดการ Error Handling, Retry Logic 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

- เคยเขียนโปรแกรมด้วยภาษาใดภาษาหนึ่งมาก่อน เช่น JavaScript เป็นต้น

เนื้อหาการเรียนและตารางเรียน 5 วัน

เวลา	เนื้อหา
วันที่ 1	
09.00 – 10.30 น.	LLMs & On-Premise AI Strategy <ul style="list-style-type: none"> ● ติดตั้ง และตั้งค่าโปรแกรมต่างๆ ที่เกี่ยวข้อง ● แนวคิดของ AI แบบ Cloud ● แนวคิดและประโยชน์ของการติดตั้ง AI/LLM แบบ On-Premise ● LLM คืออะไร ● สถาปัตยกรรม LLM และประเภทของโมเดล ● หลักการ และแนวคิดการเลือกโมเดล ● 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 <ul style="list-style-type: none"> ● การติดตั้ง Ollama และจัดการโมเดลต่างๆ ● การใช้งาน Ollama ● การตั้งค่าโมเดลและปรับแต่งประสิทธิภาพ ● โมเดลที่ทำงานกับภาษาไทยได้ดี ● ติดตั้งและทดสอบโมเดลหลายตัว และการใช้งาน Chat UI สำเร็จรูป การติดตั้ง Next.js เพื่อเตรียมสร้าง AI Chatbot <ul style="list-style-type: none"> ● การติดตั้ง Next.js ● การสร้างโปรเจกต์ใหม่ ● พื้นฐานการเขียน Next.js ● การใช้งาน UI สำเร็จรูป ● การติดตั้งฐานข้อมูล และการเชื่อมต่อ

เวลา	เนื้อหา
	<ul style="list-style-type: none"> ● ทดลองสร้างหน้า dashboard และ ระบบ Authentication
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	AI Agent Fundamentals <ul style="list-style-type: none"> ● การใช้งาน LMM ประเภทต่างๆ และแนวปฏิบัติที่ดี ● สถาปัตยกรรม Agent ● ประเภทของ Agent ● แนะนำ Tool Calling และ Function Calling (การเรียกใช้ฟังก์ชัน) ● ทำอย่างไรให้ LLM ตอบคำถามขององค์กร
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	LangChain.js Core Concepts <ul style="list-style-type: none"> ● แนะนำ และติดตั้ง LangChain.js ● สถาปัตยกรรม และ Core component ● ทำความรู้จักกับ Agents, Models, Messages, Tools, Short-term memory ● การเปรียบเทียบ Chat Models กับ LLMs ทั่วไป ● ทดลองสร้าง AI Agent ง่ายๆ
วันที่ 2	
09.00 – 10.30 น.	Next.js Chat UI Development <ul style="list-style-type: none"> ● แนะนำการสร้าง UI ● Real-time Streaming (การส่งข้อมูลแบบเรียลไทม์) ด้วย AI SDK ● สร้างคอมโพเนนต์หน้า Chat ● แนะนำการจัดการ Message History และการจัดการ Session ● ทดลองเชื่อมต่อกับ Langchain
10.30 – 10.45 น.	พักเบรก
10.45 -12.00 น.	Prompt Engineering & Context Engineering <ul style="list-style-type: none"> ● รูปแบบการออกแบบและเขียน Prompt แบบต่างๆ เช่น (Zero-shot, Few-shot, Chain-of-Thought) ● Context Engineering สำหรับองค์กร ● Context Engineering Best Practices ● เทคนิคการจัดการ Context อย่างมีประสิทธิภาพ ● แนะนำ Harness Engineering ● การสร้าง dynamic system prompt ใน Langchain ● การเขียน Tools ด้วย Langchain ● ทดลองสร้าง AI Chatbot ที่สามารถเรียกใช้ Tools ได้
12.00 – 13.00 น.	พักรับประทานอาหาร

เวลา	เนื้อหา
13.00 – 14.30 น.	แนวคิดของ MCP <ul style="list-style-type: none"> ภาพรวม MCP (Model Context Protocol) ความสำคัญของ MCP แนะนำการสร้าง MCP Server เพื่อเชื่อมต่อ API ภายในองค์กร การเชื่อมต่อฐานข้อมูล (PostgreSQL) การเชื่อมต่อ Authentication การเรียก Tool จาก MCP ทดลองสร้าง MCP Server สำหรับระบบองค์กร ทดลองให้ AI Chatbot เรียก Tool จาก MCP
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	AI Security & Guardrails <ul style="list-style-type: none"> หลักการของ Middleware ใน Langchain การใช้งาน Middleware ใน Langchain การป้องกัน Prompt Injection (การแทรก Prompt ที่เป็นอันตราย) PII Detection (ตรวจจับและปิดบังข้อมูลส่วนบุคคล) แนะนำหลักการ และการใช้งาน Guardrails ทดลองสร้างระบบ Guardrails ใน Langchain
วันที่ 3	
09.00 – 10.30 น.	RAG Architecture & Vector Databases <ul style="list-style-type: none"> แนวคิด RAG และการป้องกันการสร้างข้อมูลเท็จ การเลือก LLM ประเภท Embeddings Embeddings (การแปลงข้อความเป็นตัวเลข) และความคล้ายคลึงของ Vector ตัวเลือก Vector Database: Qdrant, Chroma, PGVector, Supabase Vector การติดตั้ง Qdrant (Docker) และตั้งค่า แนะนำ Retrieval Pipeline
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	Production RAG Implementation <ul style="list-style-type: none"> Document Loaders (การโหลดเอกสาร) - PDF, DOCX, TXT, CSV กลยุทธ์การแบ่งข้อความ (Chunking Strategies) Metadata Filtering และ Hybrid Search (การค้นหาแบบผสม) การเพิ่มประสิทธิภาพการค้นหา (MMR, Threshold, Reranking) ทดลองสร้าง จัดเก็บเอกสารและค้นหาข้อมูลใน Vector Database ทดลองให้ AI Chatbot สามารถตอบคำถามจากระบบ RAG
12.00 – 13.00 น.	พักรับประทานอาหาร

เวลา	เนื้อหา
13.00 – 14.30 น.	Agentic RAG แบบ Multimodal <ul style="list-style-type: none"> ● สถาปัตยกรรม Agentic RAG ● การทำความเข้าใจภาพ (OCR, Vision Models) ● การเลือก Vision Model มาใช้งานอย่างเหมาะสม ● การดึงข้อมูลจากภาพ แปลงเป็นข้อความ ● Pipeline การประมวลผลเอกสารที่ซับซ้อน ● ทดลองสร้าง AI Agent Bot ที่รองรับรูปภาพ
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	Short-term memory & Long-term memory ใน LangChain <ul style="list-style-type: none"> ● short-term memory คืออะไร ● ทดลองใช้งาน short-term memory ● ทดลองใช้ Runtime ใน LangChain ● การใช้งาน Long-term memory
วันที่ 4	
09.00 – 10.30 น.	Workshop: Short-term memory & Long-term memory <ul style="list-style-type: none"> ● ทดลองสร้าง Chat History เพื่อให้ AI จำจดบทสนทนา ● ทดลองทำ Workshop สำหรับการตั้งค่า user preference เพื่อให้ระบบจดจำสไตล์และตัวตนของผู้ใช้งาน
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	ทำความรู้จักกับ Multi-Agent Systems <ul style="list-style-type: none"> ● แนวคิด Multi-Agent และ LangChain.js ● ทำไมและเมื่อไหร่ต้องใช้ Muti Agent ● การเลือก Pattern แบบต่างๆ ● การสื่อสารระหว่าง Agent และจัดการสถานะ ● ทดลองให้ AI Agent มี Agent เพิ่ม ด้วย Subagents ● ทดลองสร้าง และใช้ Skill กับ AI Agent
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	Workshop: Multi-Agent Systems <ul style="list-style-type: none"> ● ทดลองให้ AI Agent มี Agent เพิ่ม ด้วย Subagents ● ทดลองสร้าง และใช้ Skill กับ AI Agent
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	แนะนำ vLLM <ul style="list-style-type: none"> ● แนะนำ vLLM ● พื้นฐานของ Inference Server

เวลา	เนื้อหา
	<ul style="list-style-type: none"> ● กลไกการทำงานของ vLLM ● คุณสมบัติและข้อได้เปรียบของ vLLM ● ข้อแตกต่างระหว่าง vLLM กับ Ollama ● ทดลองใช้ vLLM Playground
วันที่ 5	
09.00 – 10.30 น.	แนะนำ LiteLLM <ul style="list-style-type: none"> ● แนะนำ LiteLLM ● การติดตั้ง LiteLLM ด้วย Docker ● LiteLLM แก้ปัญหาอะไรบ้าง ● LiteLLM Proxy Server คืออะไร ● พีเจเออร์เสริมและการใช้งาน ● ทดลองติดตั้ง และสร้าง Proxy Server
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	Production Deployment Strategy <ul style="list-style-type: none"> ● การตั้งค่า Docker และ Docker Compose ● Deploy vLLM สำหรับ High-Performance Inference (ประมวลผลเร็ว) ● AI Gateway ด้วย LiteLLM Proxy ● Load Balancing และ Rate Limiting (การกระจายโหลดและจำกัดอัตรา)
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	Observability & Monitoring <ul style="list-style-type: none"> ● แนะนำ Langfuse ● การเชื่อมต่อ Langfuse กับ LiteLLM สำหรับ Tracing (ติดตามการทำงาน) ● กลยุทธ์การทำ Logging (บันทึกข้อมูล) ● Metrics และ Dashboards (แดชบอร์ดแสดงผล) ● Latency, Token, Error Rate
14.30 – 14.45 น.	พักเบรก
14.45 – 16.00 น.	แนะนำการนำ CI/CD เข้ามาใช้งาน <ul style="list-style-type: none"> ● แนะนำ CI/CD Pipeline สำหรับแอป AI ● Code Review และแนวปฏิบัติที่ดี ● ถาม-ตอบ และปิดหลักสูตร