



สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

COURSE ID
AIC-06

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

AI / LLM ENGINEERING

AI Chatbots with LangChain.js, Next.js and Supabase

สร้าง AI Chatbot ที่ตอบจากเอกสารองค์กร
ด้วย TypeScript ตั้งแต่ต้นจน Deploy

● 24 ชั่วโมง

● 4 วัน

● พื้นฐาน TypeScript และ Next.js

● ระดับ Intermediate



ระยะเวลา: 4 วัน

24 ชั่วโมง (4 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

องค์กรจำนวนมากมีความรู้กระจายอยู่ในคู่มือ นโยบาย สเปกสินค้า และเอกสารภายในบนร้อยไฟล์ พนักงานเสียเวลาค้นหาและถามคำถามเดิมซ้ำ ๆ AI Chatbot ที่ตอบจากเอกสารเหล่านี้ได้อย่างถูกต้อง อ้างอิงแหล่งที่มาได้ และรู้ว่าผู้ใช้แต่ละคนมีสิทธิ์เห็นข้อมูลใด จึงกลายเป็นระบบที่หลายองค์กรอยากมี แต่การทำให้ใช้งานได้จริงต้องอาศัยทั้งงาน AI งานเว็บ และงานฐานข้อมูลไปพร้อมกัน

หลักสูตรนี้พาผู้เรียนสร้าง AI Chatbot สำหรับองค์กรด้วย TypeScript ตั้งแต่ต้นจนขึ้นระบบจริง ใช้ Next.js 16 สร้างทั้งหน้าแชทและ API ในโพรเจกต์เดียว ใช้ LangChain.js 1.x เชื่อมโมเดลจากหลายผู้ให้บริการพร้อมตอบแบบ Streaming และสร้าง Agent ด้วย createAgent ใช้ Supabase ทำระบบสมาชิก เก็บประวัติแชทที่ป้องกันด้วย Row Level Security และเก็บเอกสารเป็นเวกเตอร์ด้วย pgvector จากนั้นทำ RAG ที่แสดงแหล่งอ้างอิง ปรับการค้นหาให้แม่นยำขึ้น วัดคุณภาพคำตอบ เพิ่ม Tool Calling และ MCP วาง Guardrail แล้ว Deploy บน Vercel หรือ Docker ปิดท้ายด้วย Capstone สร้าง Chatbot ขององค์กรครบทั้งระบบ (อบรม 4 วัน วันละ 6 ชั่วโมง รวม 24 ชั่วโมง ระดับ Intermediate)

เครื่องมือ

LC LangChain.js 1.x

N Next.js 16

SB Supabase

TS TypeScript

pg pgvector

ราคาค่าอบรม · PUBLIC / IN-HOUSE

12,500 บาท / ท่าน

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide ซอร์สโค้ดโปรเจกต์ครบทุก Lab สคริปต์ SQL สำหรับ Supabase 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

- สร้างแอป Full-stack ด้วย Next.js 16 ที่มีทั้งหน้าแชทและ AI API ในโปรเจกต์เดียว
- ใช้ LangChain.js 1.x เชื่อมโมเดลจากหลายผู้ให้บริการ พร้อมตอบแบบ Streaming
- ทำระบบสมาชิกด้วย Supabase Auth และเก็บประวัติแชทที่ป้องกันด้วย Row Level Security
- สร้าง Pipeline นำเอกสารองค์กรเข้า Supabase ในรูปเวกเตอร์ด้วย pgvector
- พัฒนา Agentic RAG ที่ตอบจากเอกสารพร้อมแสดงแหล่งอ้างอิง และปรับการค้นหาให้แม่นยำขึ้น
- เพิ่ม Tool Calling และเชื่อม MCP Server ให้ Agent ทำงานกับระบบอื่นได้
- วาง Guardrail ด้านความปลอดภัยและข้อมูลส่วนบุคคล และวัดคุณภาพคำตอบก่อนใช้งานจริง
- Deploy ระบบขึ้น Vercel หรือ Docker พร้อมตั้งค่า Environment อย่างปลอดภัย

02 กลุ่มเป้าหมาย Target Audience

- Web Developer สาย Frontend และ Full-stack ที่ต้องการเพิ่มทักษะด้าน AI และ LLM
- นักพัฒนา JavaScript และ TypeScript ที่ต้องการสร้างแอป AI ที่ใช้งานได้จริง
- Solution Architect และ Tech Lead ที่ต้องวางระบบ AI Chatbot ให้องค์กร
- ทีมในองค์กรที่อยากมีบอทถามตอบจากคู่มือ สเปก หรือนโยบาย
- อาจารย์ วิทยากร และที่ปรึกษาที่ต้องการต้นแบบ RAG ไปสอนหรือปรับใช้ต่อ

03 จุดเด่นของหลักสูตร Highlights

- **ปรับปรุงใหม่ปี 2026** LangChain.js 1.x, Next.js 16 และ Supabase
- **TypeScript ทั้งระบบ** หน้าแชท, API และ Agent ในโปรเจกต์เดียว
- **RAG ที่อ้างอิงได้** pgvector, Hybrid Search และแสดงแหล่งที่มา
- **ปลอดภัยตั้งแต่ต้น** Supabase Auth, RLS และ Guardrail
- **พร้อมขึ้นระบบจริง** Deploy บน Vercel หรือ Docker

04 ความรู้พื้นฐาน Prerequisites

- เขียน JavaScript หรือ TypeScript ได้ และเคยใช้ React หรือ Next.js เบื้องต้น
- เข้าใจ REST, HTTP, ไฟล์ .env และใช้ Git ขึ้นพื้นฐานได้
- รู้ SQL เบื้องต้น เช่น ตาราง คีย์ และความสัมพันธ์ จะช่วยได้มาก
- โน้ตบุ๊กที่ติดตั้ง Node.js, VS Code และ Docker Desktop ได้ พร้อมสิทธิ์ติดตั้งโปรแกรม

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook Windows, macOS หรือ Linux เชื่อมต่ออินเทอร์เน็ต ติดตั้ง Node.js และ Docker Desktop ได้
Framework	Next.js 16, TypeScript, Tailwind CSS, shadcn/ui และ LangChain.js 1.x
Backend	Supabase (Auth, Postgres, pgvector, Storage) และ Supabase CLI สำหรับพัฒนาในเครื่อง
โมเดล AI	API ของผู้ให้บริการ LLM ที่ผู้เรียนเลือก เช่น OpenAI, Anthropic, Google หรือ Groq
ข้อมูลตัวอย่าง	ชุดคู่มือและนโยบายของคณาจารย์มหาวิทยาลัยไทยสำหรับทุก Lab

หมายเหตุ: ปรับปรุงใหม่ปี 2026 เทียบกับรุ่นเดิม: เปลี่ยนจาก LLMChain และ Agent Executor ที่เลิกใช้แล้วมาใช้ LangChain.js 1.x กับ createAgent และ Middleware อัปเดตเป็น Next.js 16 ก็ใช้ proxy.ts แทน middleware.ts ใช้ @supabase/ssr กับคีย์แบบ Publishable และ Secret รุ่นใหม่ของ Supabase เพิ่ม Agentic RAG, Hybrid Search, การวัดคุณภาพคำตอบ, MCP และ Guardrail Framework เป็นโอเพนซอร์ส Supabase และ Vercel มีแพ็คเกจใช้ฟรี แต่การเรียก API ของผู้ให้บริการ LLM ใช้บัญชีของผู้เรียนเองซึ่งอาจมีค่าใช้จ่ายตามการใช้งาน ชื่อโมเดลและฟีเจอร์เปลี่ยนเร็ว เนื้อหาอาจอิงเวอร์ชันล่าสุด ณ วันอบรม

06 ภาพรวมการเรียนรู้ 4 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-5	เริ่มโปรเจกต์ Next.js 16 ด้วย TypeScript, สร้าง AI API ด้วย Route Handler, ใช้ LangChain.js 1.x เชื่อมโมเดลหลายค่าย, ตอบแบบ Streaming และสร้างหน้าแชทด้วย shadcn/ui
Day 2 Section 6-9	ตั้งค่า Supabase และ Supabase CLI, ทำระบบสมาชิกด้วย Supabase Auth และ proxy.ts, ออกแบบตารางประวัติแชทพร้อม Row Level Security และให้ AI จำบทสนทนาได้
Day 3-4 Section 10-18	ทำ RAG จากเอกสารองค์กรด้วย pgvector, Agentic RAG พร้อมแหล่งอ้างอิง, ค้นหาแบบ Hybrid, วัดคุณภาพคำตอบ, Tool Calling และ MCP, วาง Guardrail แล้ว Deploy บน Vercel หรือ Docker



DAY
1

Next.js และ LangChain.js พื้นฐาน

เนื้อหาการอบรม · Section 1-5

รวม
6 ชั่วโมง

1 Lab: ภาพรวม AI Chatbot และเตรียมโปรเจกต์

- Generative AI, LLM และสิ่งที่ Chatbot องค์กรต้องมีมากกว่าการถามตอบ
- สถาปัตยกรรมของระบบในหลักสูตร: Next.js, LangChain.js, Supabase และ Vercel
- เลือกผู้ให้บริการโมเดลและประเมินค่าใช้จ่ายต่อการใช้งาน
- ติดตั้ง Node.js, pnpm และสร้างโปรเจกต์ด้วย create-next-app แบบ TypeScript
- Lab: จัดโครงสร้างโปรเจกต์ ตั้งค่า ESLint และเก็บโค้ดบน Git

2 Lab: AI API ด้วย Route Handler ใน Next.js 16

- App Router, Server Component และ Client Component ในมุมมองของแอป AI
- สร้าง Route Handler ใน app/api/chat/route.ts รับ Request และส่ง Response
- ตรวจสอบข้อมูลขาเข้าด้วย Zod และจัดการ Error อย่างเป็นระบบ
- เก็บ API Key ใน .env.local และไม่ให้หลุดไปฝั่ง Client
- Lab: ทดสอบ Endpoint ด้วย Thunder Client หรือ Postman

3 Lab: LangChain.js 1.x เชื่อมโมเดลภาษา

- โครงสร้างแพ็คเกจของ LangChain.js 1.x และสิ่งที่ย้ายไป @langchain/classic
- เชื่อม Chat Model จากหลายผู้ให้บริการ และสลับโมเดลโดยไม่แก้ไขโค้ดส่วนอื่น
- Message, System Prompt และ Prompt Template สำหรับควบคุมคำตอบ
- Structured Output ด้วย Zod เพื่อให้ได้ข้อมูลที่โค้ดนำไปใช้ได้
- Lab: ต่อโมเดลเข้ากับ Route Handler ที่สร้างไว้

4 Lab: ตอบแบบ Streaming

- ทำไม Chatbot ต้องตอบแบบ Streaming และผลกระทบต่อประสบการณ์ผู้ใช้
- Stream ผลจากโมเดลและ Agent ใน LangChain.js
- ส่ง ReadableStream ออกจาก Route Handler และยกเลิกคำขอลากลางทางได้
- จัดการ Timeout, Rate Limit และ Error ระหว่าง Stream
- Lab: API ที่ทยอยส่งคำตอบให้หน้าเว็บแบบ Real-time

5 Lab: หน้าแชทด้วย shadcn/ui

- ติดตั้ง shadcn/ui กับ Tailwind CSS และเลือก Component ที่จำเป็น
- ประกอบหน้าแชทจาก Card, Input, Button, Avatar และ Scroll Area
- จัดการ State ของข้อความฝั่ง Client และรับข้อความแบบ Streaming
- แสดงผล Markdown และโค้ดในคำตอบ พร้อมสถานะ Loading และ Error
- Lab: หน้าแชทที่คุยกับ AI API ได้ครบวงจร



DAY
2

ระบบสมาชิก ประวัติแชท และความจำ

เนื้อหาการอบรม · Section 6-9

รวม
6 ชั่วโมง

6 Lab: เริ่มต้น Supabase

- Supabase คืออะไร: Postgres, Auth, Storage และ API ในบริการเดียว
- สร้างโปรเจกต์บน Dashboard และรู้จักกับ Publishable กับ Secret
- ใช้ Supabase CLI รันระบบในเครื่องด้วย Docker และจัดการ Migration
- สร้าง TypeScript Types จาก Schema เพื่อเรียกฐานข้อมูลแบบมี Type
- Lab: เชื่อม Next.js กับ Supabase ทั้งฝั่ง Browser และ Server

7 Lab: ระบบสมาชิกด้วย Supabase Auth

- ใช้ @supabase/ssr สร้าง Client สำหรับ Server Component และ Route Handler
- รีเฟรช Session ใน proxy.ts ของ Next.js 16 และป้องกันหน้าที่ต้องล็อกอิน
- สมัครสมาชิกและล็อกอินด้วยอีเมล และด้วยบัญชี Google ผ่าน OAuth
- ตรวจสอบตัวตนผู้ใช้ฝั่ง Server ทุกครั้งก่อนเรียก AI API
- Lab: หน้า Login, Register และ Logout ที่ใช้งานได้จริง

8 Lab: ประวัติแชทและ Row Level Security

- ออกแบบตาราง conversations และ messages ที่ผูกกับผู้ใช้
- เปิด Row Level Security และเขียน Policy ให้เห็นเฉพาะข้อมูลของตัวเอง
- บันทึกคำถามและคำตอบทุกครั้ง พร้อมข้อมูลโมเดลและจำนวน Token
- ดึงรายการบทสนทนาเก่ามาแสดง และตั้งชื่อบทสนทนาอัตโนมัติด้วย AI
- Lab: ทดสอบว่าผู้ใช้คนหนึ่งเข้าถึงแชทของอีกคนไม่ได้

9 Lab: ให้ AI จำบทสนทนา

- ส่งประวัติการสนทนาเข้า Prompt เพื่อให้คุยต่อเนื่องได้
- จำกัดขนาด Context ด้วยการตัดข้อความเก่าและสรุปบทสนทนา
- Short-term Memory ด้วย Checkpointer และ thread_id ของ LangChain
- เลือกระหว่างเก็บความจำในตารางของแอปหรือใน Checkpointer บน Postgres
- Lab: Chatbot ที่จำบริบทข้ามหลายคำถามและเปิดแชทเก่าคุยต่อได้



DAY
3

RAG จากเอกสารองค์กร

เนื้อหาการอบรม · Section 10-14

รวม
6 ชั่วโมง

10 แนวคิด RAG และ Embedding

- RAG ทำงานอย่างไร และต่างจากการ Fine-tune โมเดลตรงไหน
- Embedding คืออะไร และการเลือก Embedding Model ที่รองรับภาษาไทย
- Chunking: ขนาด การซ้อนทับ และผลต่อความแม่นยำของคำตอบ
- pgvector บน Supabase: คอลัมน์ vector, Distance Operator และ HNSW Index
- ต้นทุนและเวลาในการประมวลผลเอกสารจำนวนมาก

11 Lab: Pipeline นำเอกสารเข้าระบบ

- อัปโหลดไฟล์ PDF, DOCX และข้อความขึ้น Supabase Storage
- Document Loader และ RecursiveCharacterTextSplitter ของ LangChain.js
- สร้างตาราง documents, ฟังก์ชัน match_documents และ Index
- บันทึก Chunk พร้อม Metadata เช่น ชื่อไฟล์ หน้า และหน่วยงาน ผ่าน SupabaseVectorStore
- Lab: สคริปต์ Ingestion ที่รันซ้ำได้และอัปเดตเอกสารที่แก้ไข

12 Lab: Agentic RAG ด้วย createAgent

- เปลี่ยน Retriever เป็น Tool ให้ Agent ตัดสินใจเองว่าเมื่อไรต้องค้นเอกสาร
- สร้าง Agent ด้วย createAgent พร้อม System Prompt ที่บังคับให้ตอบจากเอกสาร
- ให้ AI ตอบว่าไม่ทราบเมื่อเอกสารไม่มีข้อมูล แทนการเดาคำตอบ
- แสดงแหล่งอ้างอิง ชื่อไฟล์ และข้อความที่ใช้ตอบบนหน้าแชท
- Lab: Chatbot ถามตอบจากคู่มือองค์กรพร้อมลิงก์ไปยังเอกสารต้นฉบับ

13 Lab: ค้นหาให้แม่นยำขึ้น

- กรองผลด้วย Metadata เช่น หน่วยงาน ประเภทเอกสาร และวันที่
- Hybrid Search รวม Full-Text Search ของ Postgres กับ Vector Search
- เขียนคำถามใหม่ก่อนค้นหา และจัดอันดับผลลัพธ์ใหม่ (Reranking)
- จำกัดสิทธิ์เห็นเอกสารตามผู้ใช้หรือทีมด้วย Row Level Security
- Lab: เปรียบเทียบผลค้นหาก่อนและหลังปรับจูนด้วยชุดคำถามเดียวกัน

14 Lab: วัดคุณภาพคำตอบ

- สร้างชุดคำถามทดสอบพร้อมคำตอบที่คาดหวังจากเอกสารจริง
- วัดความถูกต้อง การอ้างอิงตรงแหล่ง และการตอบนอกเอกสาร
- ใช้ LLM ช่วยให้คะแนนคำตอบ และข้อควรระวังของวิธีนี้
- ติดตาม Trace การทำงานของ Agent ด้วย LangSmith หรือ Log ของระบบ
- Lab: รับชุดทดสอบทุกครั้งที่เปลี่ยน Prompt หรือวิธีค้นหา

DAY
4

Tool, ความปลอดภัย และการ Deploy

เนื้อหาการอบรม · Section 15-18

รวม
6 ชั่วโมง

15 Lab: Tool Calling และ MCP

- Tool Calling คืออะไร และงานที่ Chatbot องค์กรควรทำผ่าน Tool
- นิยาม Tool ด้วยฟังก์ชัน tool() พร้อม Schema แบบ Zod
- ใช้ Middleware จัดการ Error ของ Tool และขออนุมัติก่อนทำงานสำคัญ
- เชื่อม MCP Server เข้ากับ Agent ผ่าน @langchain/mcp-adapters
- Lab: Agent ที่ค้นเอกสาร เรียก API ภายใน และสร้างคำขอแทนผู้ใช้

16 ความปลอดภัยและ Guardrail

- Prompt Injection จากผู้ใช้และจากเนื้อหาในเอกสาร พร้อมแนวทางป้องกัน
- ซ่อนข้อมูลส่วนบุคคลด้วย Middleware และขอควรคำนึงตาม PDPA
- จำกัดจำนวนคำขอต่อผู้ใช้ และควบคุมค่าใช้จ่ายของโมเดล
- บริหาร Secret, สิทธิ์ของคีย์ Supabase และ Log ที่ไม่เก็บข้อมูลอ่อนไหว
- Lab: ทดสอบโจมตีระบบของตัวเองแล้วอุดช่องโหว่

17 Lab: Deploy ขึ้นใช้งานจริง

- เตรียมโปรเจกต์ Supabase สำหรับ Production และย้าย Migration ขึ้นไป
- Deploy บน Vercel พร้อมตั้ง Environment Variables และเชื่อม GitHub
- ทางเลือก Deploy ด้วย Docker โดยใช้ output แบบ standalone
- ตรวจสอบ Log, แก้ปัญหาที่พบบ่อย และติดตามการใช้งานหลังเปิดระบบ
- Lab: ทดสอบ Chatbot บน URL จริง

18 Workshop: Capstone Chatbot องค์กร

- เลือกโจทย์ เช่น บอกรายงาน HR บอกละเอียดสินค้า หรือบอก IT Helpdesk
- นำเอกสารเข้าระบบ กำหนดสิทธิ์ตามทีม และเพิ่ม Tool ที่จำเป็น
- วัดคุณภาพคำตอบด้วยชุดทดสอบ แล้วปรับจนผ่านเกณฑ์
- นำเสนอผลงาน รีวิวร่วมกัน และสรุป Checklist ก่อนเปิดใช้ในองค์กร

07 เรียนจบแล้วได้อะไร Outcomes

- สร้างแอป AI ด้วย Next.js หน้าแรก, API และ Streaming
- ใช้ LangChain.js 1.x คล่อง โมเดลหลายค่าย, createAgent และ Tool
- ระบบสมาชิกที่ปลอดภัย Supabase Auth และ Row Level Security
- RAG จากเอกสารองค์กร pgvector, Hybrid Search และแหล่งอ้างอิง
- ควบคุมคุณภาพได้ ชุดทดสอบ, Trace และ Guardrail
- ขึ้นระบบจริงได้ Vercel, Docker และ Supabase Production

08 สิ่งที่คุณเรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ครบทั้ง 4 วัน และ Lab Guide ทีละขั้นตอน
- ซอร์สโค้ดโปรเจกต์ Next.js และ LangChain.js ครบทุก Lab พร้อมโปรเจกต์ Capstone
- สคริปต์ SQL และ Migration สำหรับ Supabase ทั้งตารางแรก, RLS Policy และ pgvector
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สามิตร โทยม ผู้สอนและนักพัฒนาที่สร้างเว็บแอปด้วย Next.js และระบบ AI Chatbot สำหรับใช้งานในองค์กร ทั้งงานถามตอบจากเอกสาร งานบริการภายใน และระบบที่ต้องควบคุมสิทธิ์การเข้าถึงข้อมูล จุดเด่นคือการสอนจากโปรเจกต์เดียวที่ต่อยอดทีละขั้นจนพร้อมใช้งาน อธิบายเหตุผลของการออกแบบแต่ละชั้น และพาผู้เรียนวัดผลทุกครั้งที่ปรับระบบ เพื่อให้กลับไปสร้างและดูแล Chatbot ขององค์กรได้อย่างมั่นใจ

ราคาค่าอบรม 12,500 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

12,500 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

☎ 02-570-8449

📞 @itgenius

☎ 088-807-9770

✉ contact@itgenius.co.th



เพิ่มเพื่อน
LINE