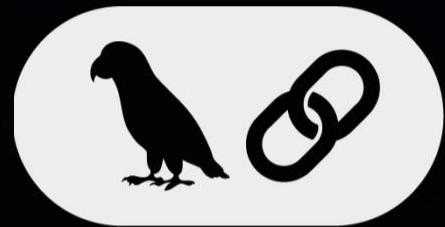


● LIVE

# สร้าง AI Chatbots สำหรับองค์กร



LangChain  
ร่วมกับ Next.JS  
และ supabase



มีวิดีโอบันทึกการอบรม  
ย้อนหลังให้ทุกวัน



สถาบันไอทีเนียส

4 วัน  
12 ชั่วโมงเต็ม



**Samit Koyom**  
สถาบันไอทีเนียส





ร่วมกับ  
และ

LangChain  
Next.JS   


## วิทยากร



อ.สา米ตร โภยม (ปาน)

ปริญญาโท คณะเทคโนโลยีและสารสนเทศ  
สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ



สถาบันไอทีเนี่ยส

### ▶ Frontend

Angular, React, Vue, Next, Nuxt, Bootstrap, Tailwind CSS

### ▶ Backend

PHP, Python, Java, Kotlin, Go, Rust, NodeJS, NestJS, .NET

### ▶ Database

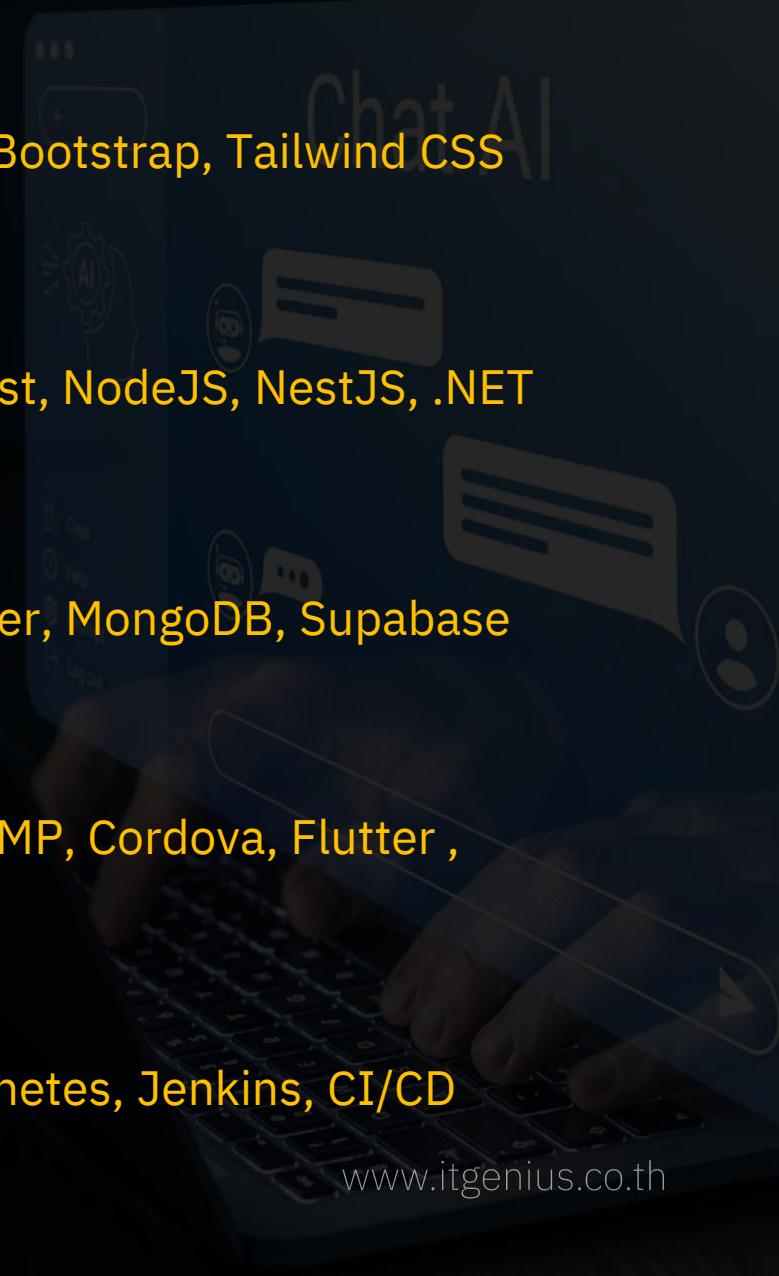
MySQL, PostgreSQL, MS SQL Server, MongoDB, Supabase

### ▶ Mobile

Java, Kotlin, Objective C, Swift, KMP, Cordova, Flutter ,  
React Native, Expo

### ▶ DevOps

Git, Github, Gitlab, Docker, Kubernetes, Jenkins, CI/CD



[www.itgenius.co.th](http://www.itgenius.co.th)

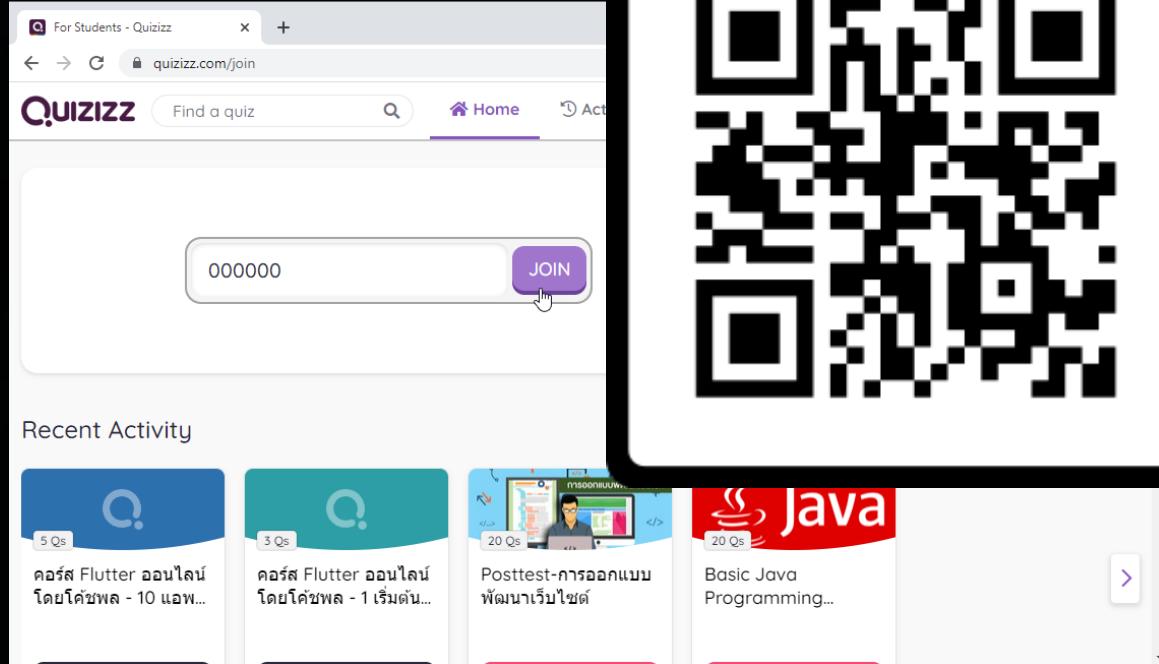
# แบบทดสอบก่อนอบรม



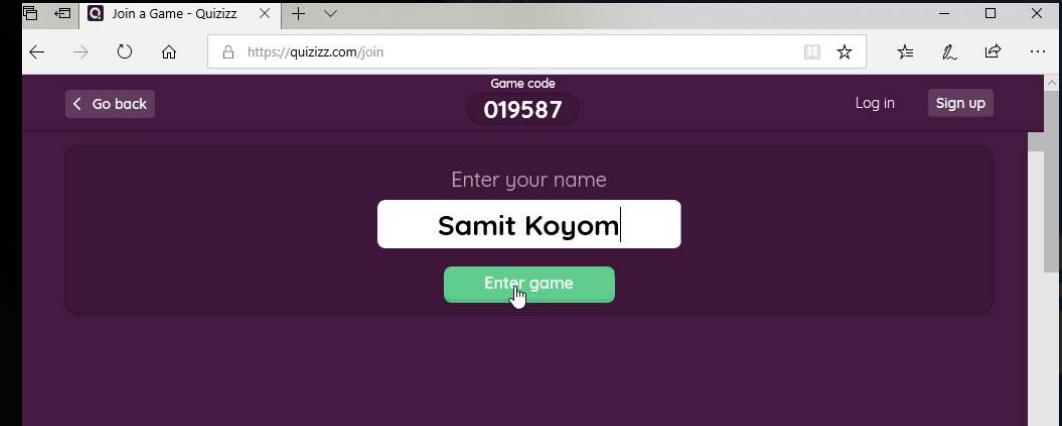
# Pretest ทำแบบทดสอบก่อนเรียน

**STEP 1:** เข้าทำแบบทดสอบกี่เลิ�ก์ ป้อนรหัสเข้าห้องสอบ

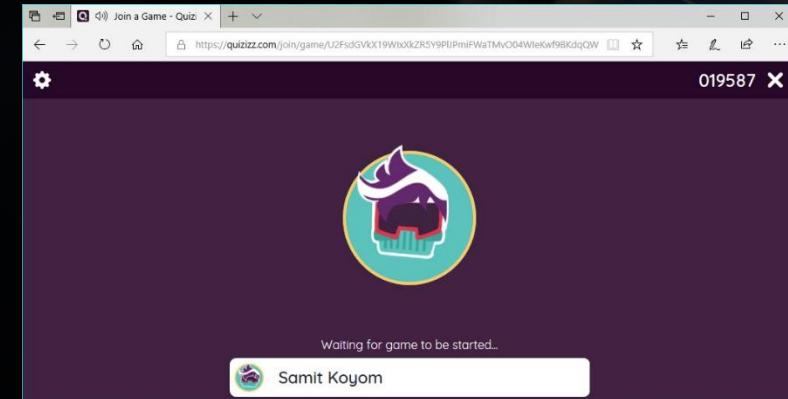
[quizizz.com/join](https://quizizz.com/join)



**STEP 2:** ป้อนชื่อ



**STEP 3:** รอผู้สอน Start ข้อสอบ



สถาบันไอทีเจเนียส

www.itgenius.co.th



**LangChain**  
ร่วมกับ **Next.JS**   
และ  **supabase**

ดาวน์โหลดเอกสารประกอบการอบรม

**[bit.ly/aichatbot-langchain](https://bit.ly/aichatbot-langchain)**



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)

# Course Outline



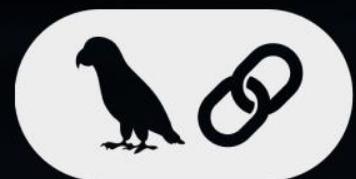
1. ภาพรวม AI Chatbot กับ Langchain.js
2. การพัฒนา Rest API ใน Next.js เพื่อใช้งานกับ Langchain.js
3. พื้นฐาน Langchain.js เชื่อมต่อกับ Gen AI
4. ระบบยืนยันตัวตนด้วย Supabase Auth
5. UI Chatbot ด้วย Prompt-kit-UI Shadcn/UI
6. AI Chatbot มีการเก็บประวัติ (Chat History)
7. เชื่อมต่อ AI กับเครื่องมือภายนอก (Tool Calling)
8. Document Loader, Embedding , Vector Store
9. พัฒนา RAG เพื่อให้ AI ตอบคำถามจากข้อมูลในเอกสารขององค์กร
10. การเผยแพร่ (Deployment) โปรเจ็คต์ไปใช้งานจริง

• LIVE

อบรมออนไลน์



## สร้าง AI Chatbots สำหรับองค์กร

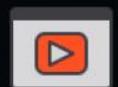


# LangChain

ร่วมกับ **Next.JS**



และ **supabase**



มีวิดีโอบันทึกการอบรม  
ย้อนหลังให้ทุกวัน



สอนสดผ่าน Zoom  
รับจำนำแนวจำนำด้วย

Chat AI วันที่ 1

The slide features a large blue speech bubble containing icons related to AI, such as a brain with 'AI', a message bubble, and a person icon. A hand is shown typing on a laptop keyboard. To the right, there is a large red circle with a white stylized 'I' inside.

Samit Koyom  
สถาบันไอทีจีเนียส

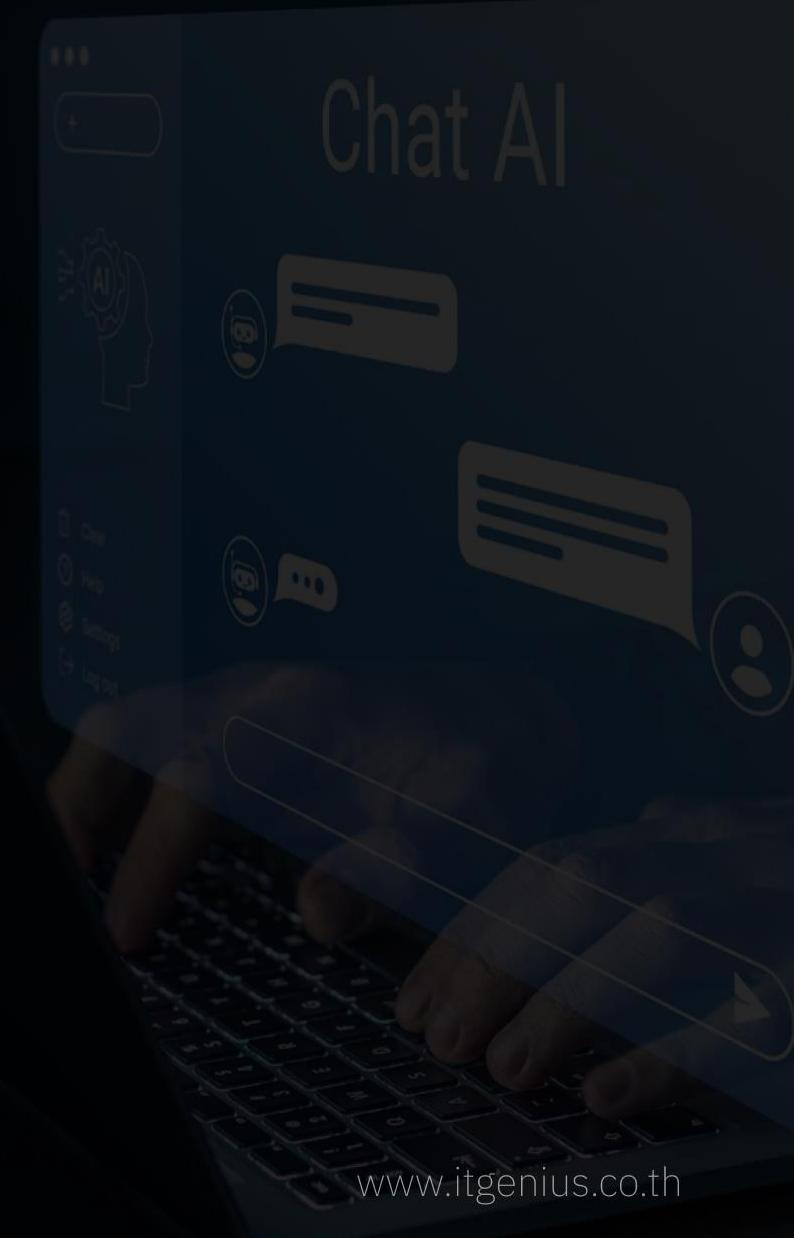


# Day 1

1. การรวม AI Chatbot กับ Langchain.js
2. การพัฒนา Rest API ใน Next.js เพื่อใช้งานกับ Langchain.js
3. พื้นฐาน Langchain.js เชื่อมต่อกับ Gen AI

# Workshop

## AI Chatbot with LangChain and NextJS

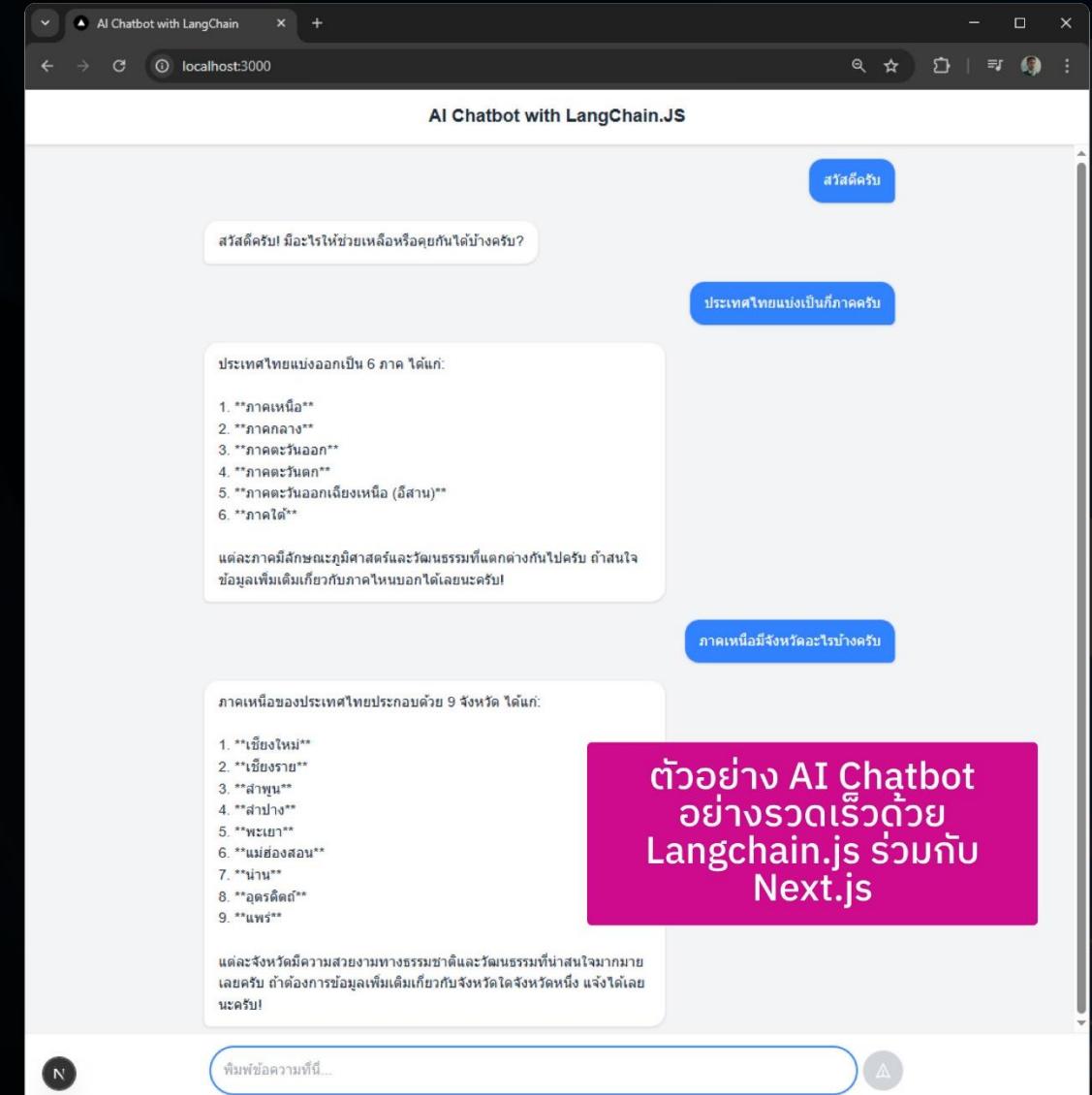
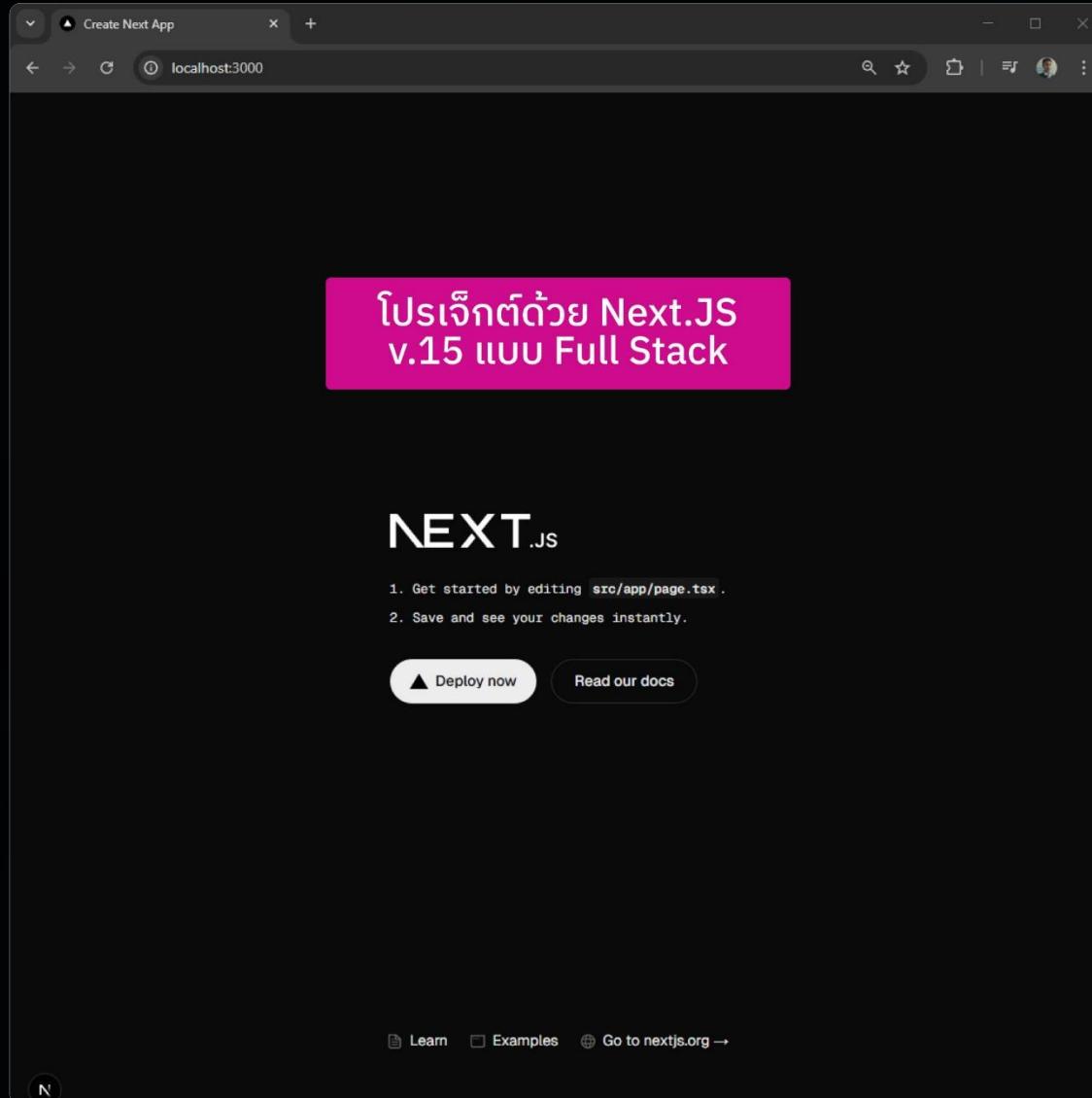


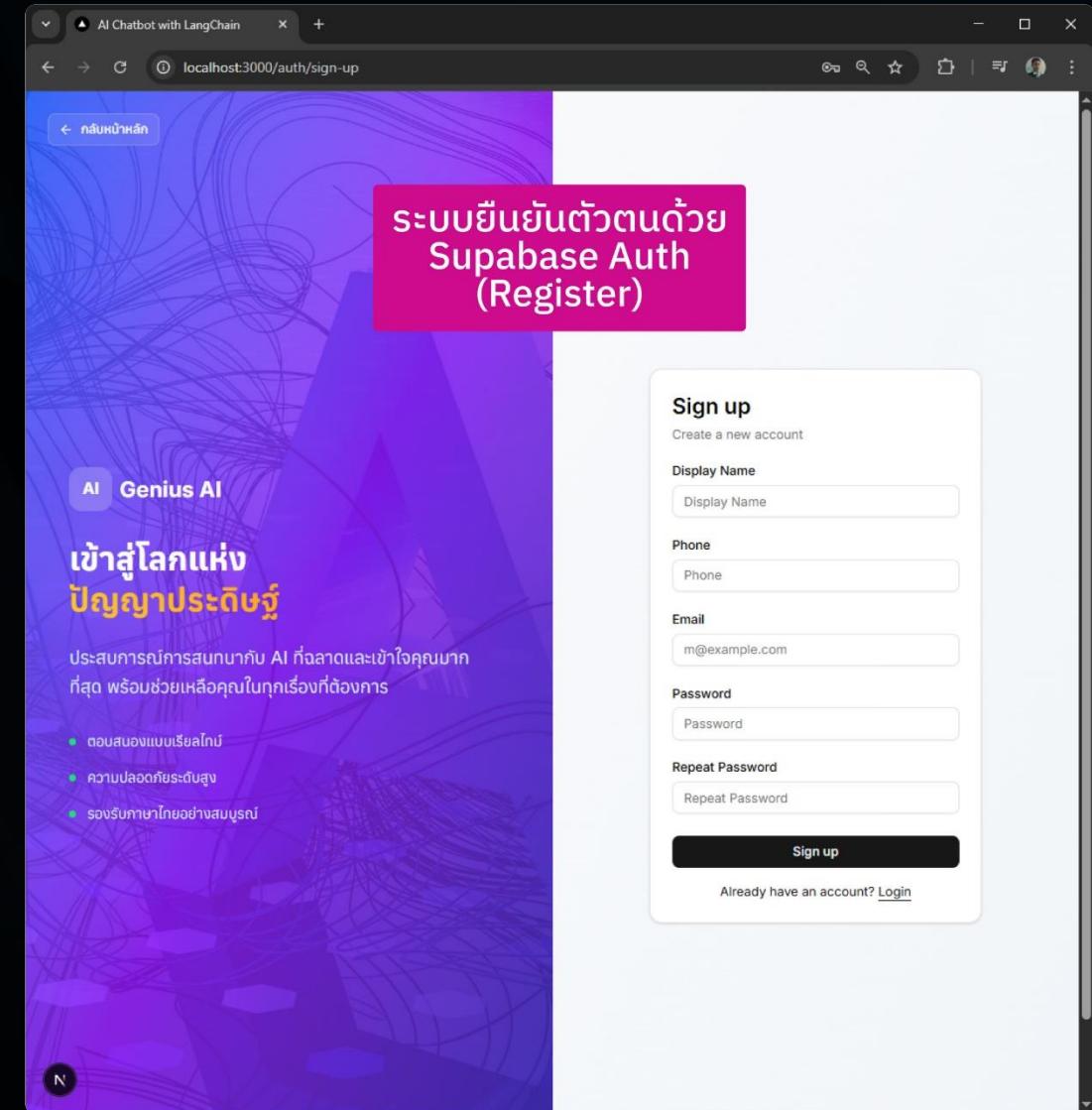
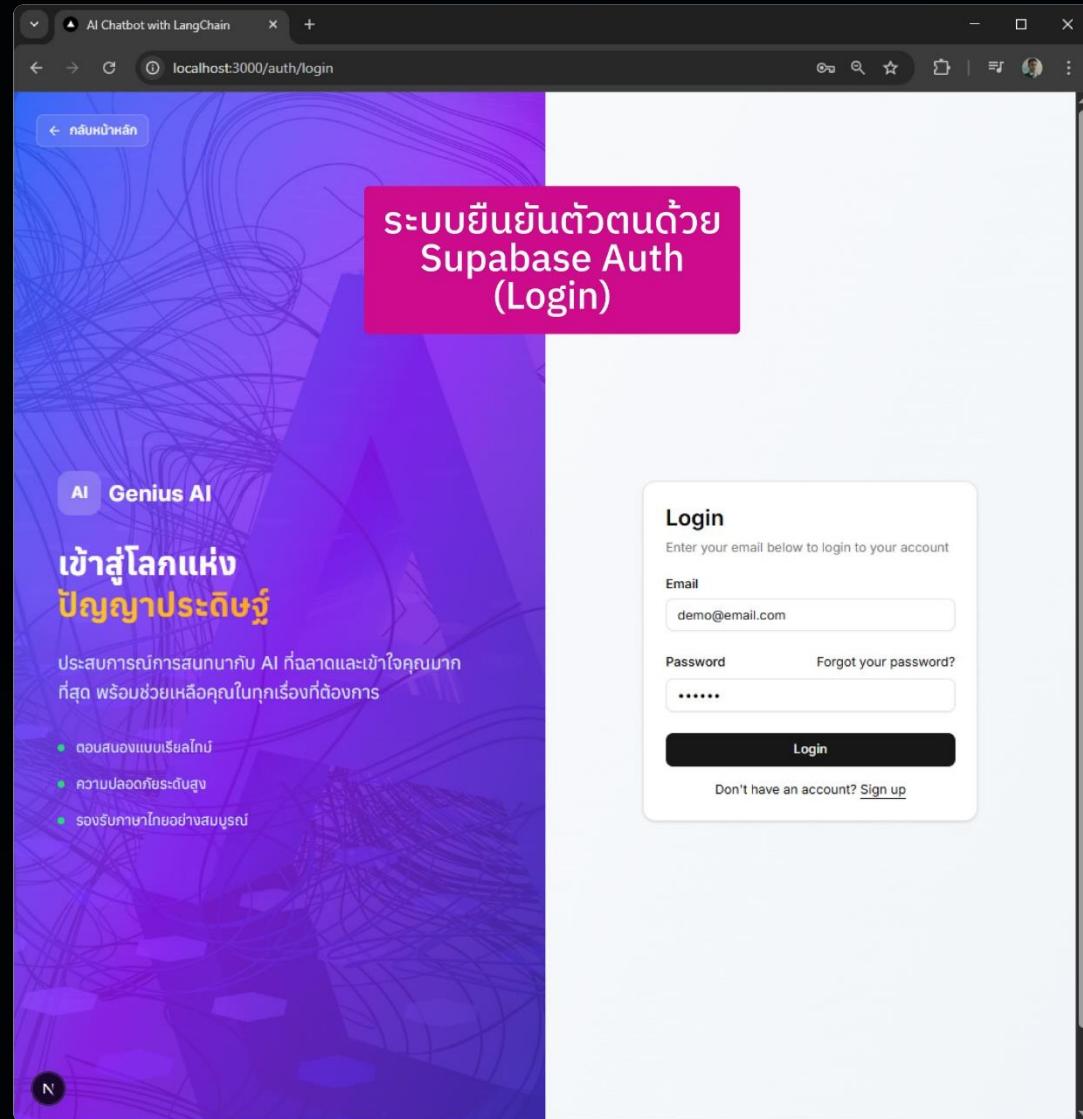
สถาบันไอทีจีเนียส

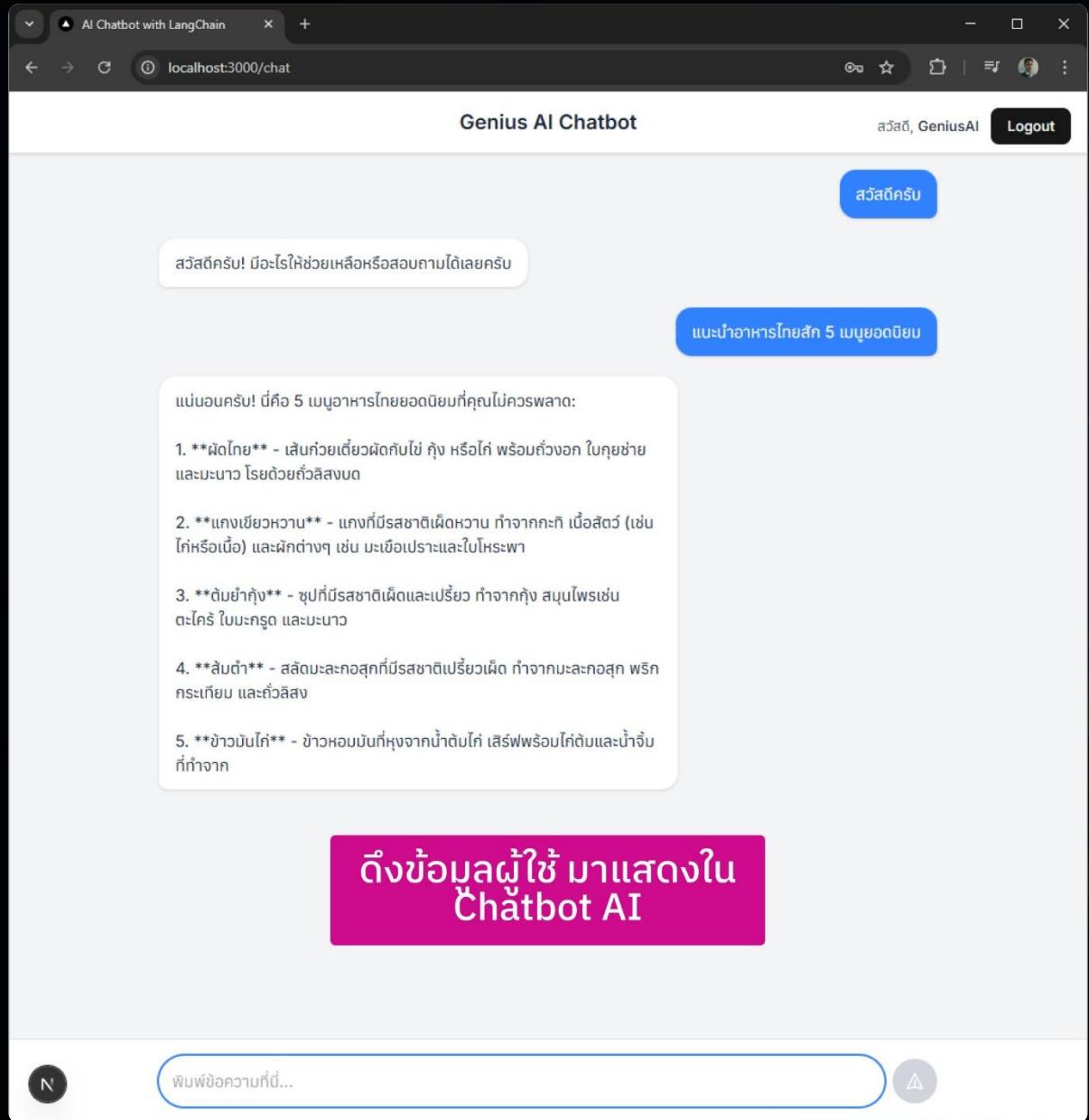
[www.itgenius.co.th](http://www.itgenius.co.th)

The screenshot shows the homepage of the Genius AI Chatbot. At the top, there's a purple header bar with the 'Genius' logo and a 'Logout' button. Below the header, a purple navigation bar contains links for 'Home', 'About', 'Features', 'Pricing', 'Contact', and 'Sign Up'. A search bar with the placeholder 'Search AI Chatbot' is positioned above a large central title. The main title 'Genius AI Chatbot' is displayed in a large, bold, black font for 'Genius AI' and a blue font for 'Chatbot'. Below the title is a detailed description of the product's features, mentioning AI, RAG, Document Loader, Vector Embeddings, and various tools like LangChain.js, Next.js, Supabase, and OpenAI GPT-4o-mini. Two buttons, 'Get Started' and 'Enterprise', are located below the description. The page is divided into several sections, each with an icon and a brief description: 'RAG & Document Search', 'Tool Calling & Smart Query', 'Security & Modern UI', 'Chat History System', 'Advanced Memory Management', and 'Modern UI & Responsive Design'. Each section includes a small icon and a brief explanatory text. At the bottom, a large blue call-to-action box features the text 'จุดเด่นที่ผู้ใช้ไว้วางใจ' (Features trusted by users), followed by three statistics: '10K+', '99.9%', and '5+'. Below these are the subtitles '易于使用', '高可用性', and '响应式设计'. The '99.9%' statistic is associated with 'Uptime & Reliability'. The '5+' statistic is associated with 'Number of integrations (RAG, Tool Calling, Document Loader, Security, UI)'. A footer bar at the bottom contains the 'Landing Page ....' button and the 'Genius AI Chatbot' logo.









prompt-kit.com/blocks

## Full chat app

Preview    Code

```
display: grid;
grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
gap: 1rem;
```

This creates a grid where:

- Columns automatically fit as many as possible
- Each column is at least 250px wide
- Columns expand to fill available space
- There's a 1rem gap between items

Would you like me to explain more about how this works?

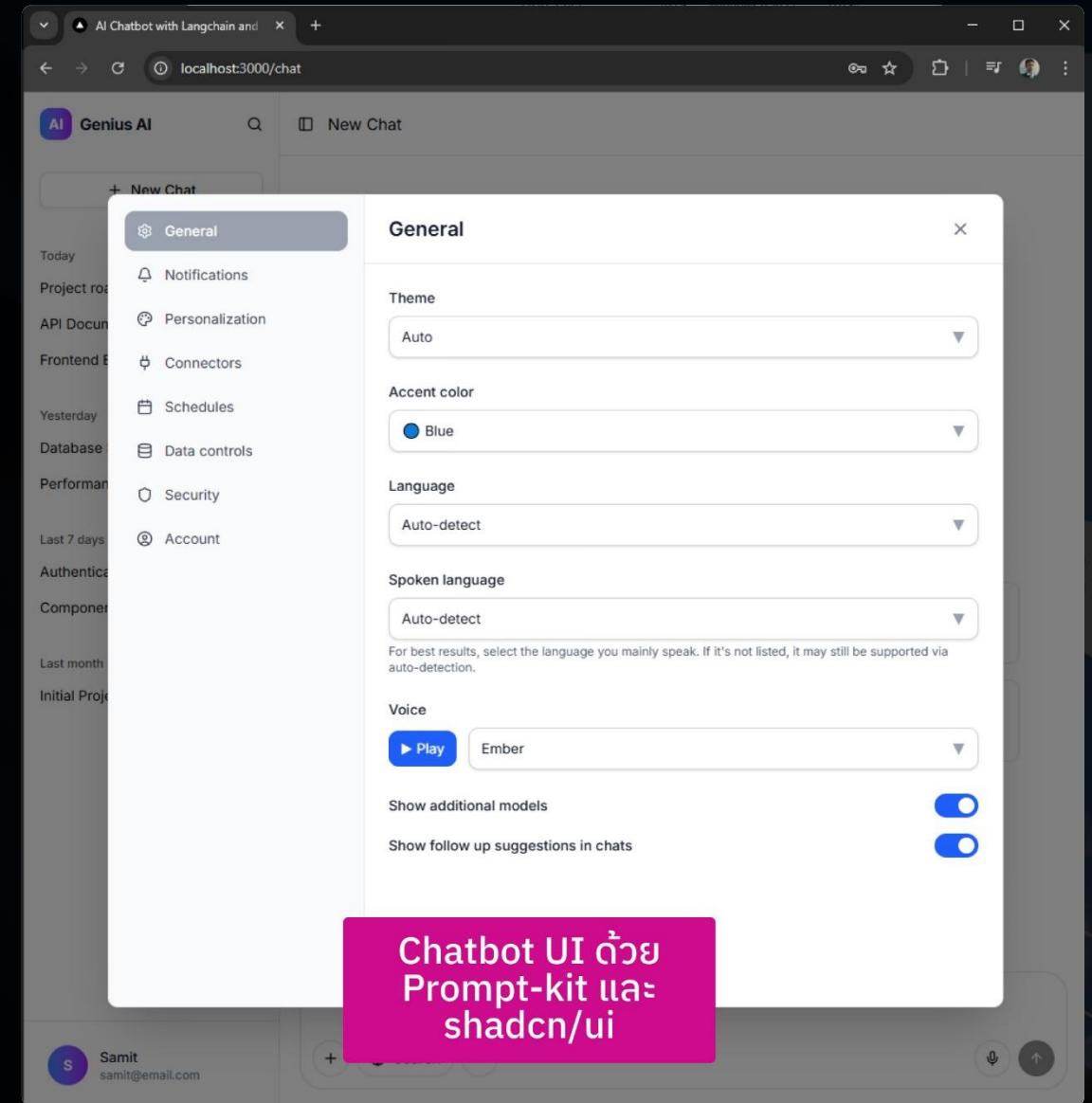
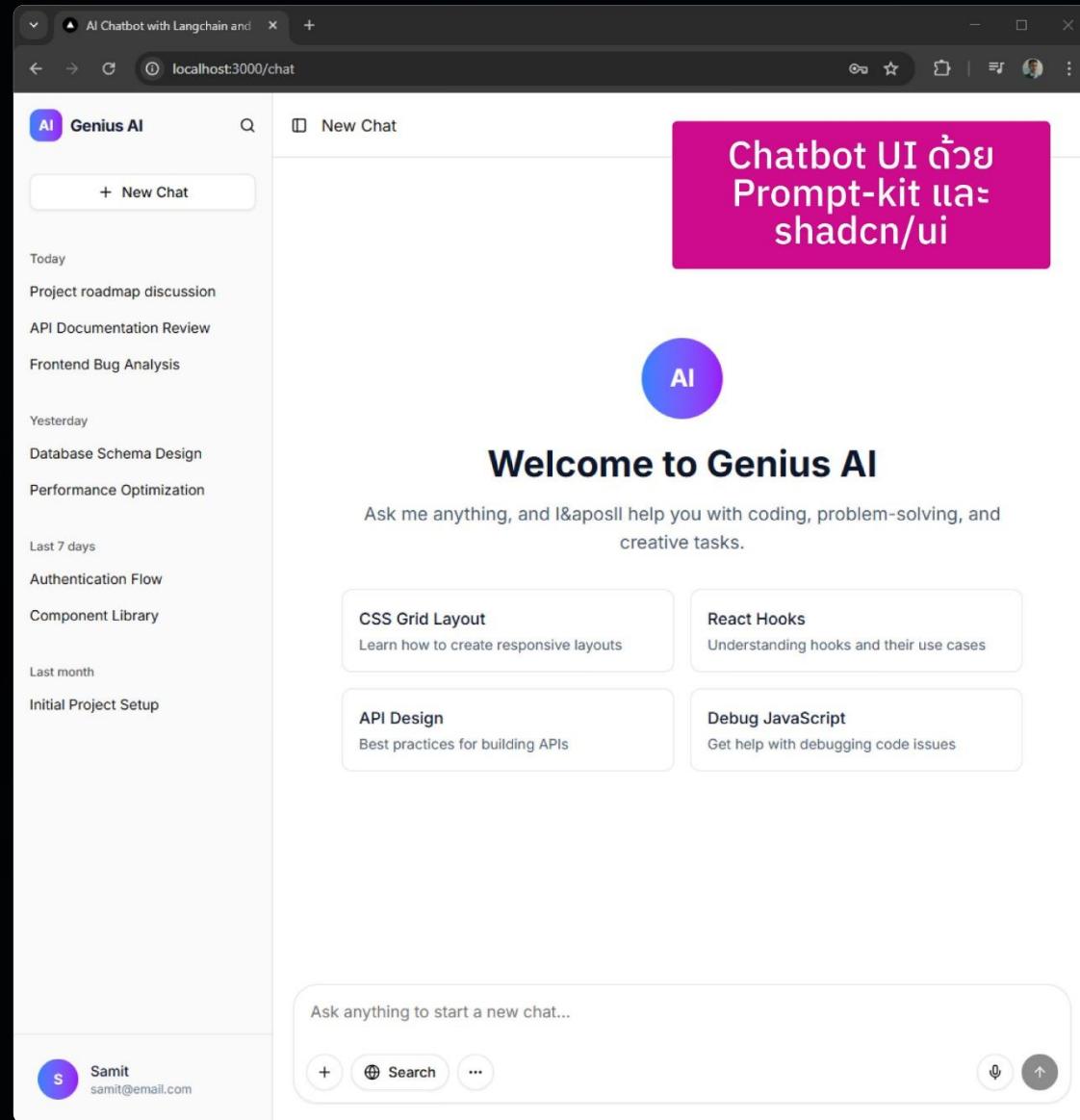
Ask anything

+    Search    ...

↑    ↓

# ปรับแต่ง Chatbot UI ด้วย prompt-kit





The screenshot shows a browser window with the URL `localhost:3000/chat/project-roadmap-discussion`. The interface has a sidebar on the left with a navigation menu:

- Today:
  - Project roadmap discussion (highlighted)
  - API Documentation Review
  - Frontend Bug Analysis
- Yesterday:
  - Database Schema Design
  - Performance Optimization
- Last 7 days:
  - Authentication Flow
  - Component Library
- Last month:
  - Initial Project Setup

A central panel displays a conversation:

Hello! Can you help me with a coding question?

Of course! I'd be happy to help with your coding question. What would you like to know?

How do I create a responsive layout with CSS Grid?

Creating a responsive layout with CSS Grid is straightforward. Here's a basic example:

```
// src/app/page.tsx
import { Button } from '@/components/ui/button'

export default function Page() {
  return (
    <main className='p-6'>
      <h1 className='text-2xl font-semibold'>Hello shadcn/ui</h1>
      <Button className='mt-4'>Press me</Button>
    </main>
  )
}
```

This creates a grid where:  
Columns automatically fit as many as possible  
Each column is at least 250px wide  
Columns expand to fill available space  
There is a gap between items

Ask anything

List top 5 frontend frameworks show in table

Chatbot UI ด้วย Prompt-kit และ shadcn/ui

Samit samit@email.com

localhost:3000/chat/project-roadmap-discussion



สถาบันไอทีเจเนียส

www.itgenius.co.th

New Chat

# Welcome to Genius AI Chatbot

Ask me anything, and I'll help you with coding, problem-solving, and creative tasks.

**CSS Grid Layout**  
Learn how to create responsive layouts

**React Hooks**  
Understanding hooks and their use cases

**API Design**  
Best practices for building APIs

Ask anything to start a new chat...

+ Search ...

AI Genius AI Q

+ New Chat

Today

- Project roadmap discussion
- API Documentation
- Frontend Bug Analysis

Yesterday

- Database Schema Design

Samit samit@email.com

- Upgrade plan
- Customize Genius AI
- Settings
- Log out

General Notifications Personalization

General

รอรับการแสดงผลบน Mobile Size

Blue

Language Auto-detect

Spoken language Auto-detect

Voice



AI Genius AI Q Project roadmap discussion

Light Dark System Today API Documentation Review Frontend Bug Analysis Yesterday Database Schema Design Performance Optimization Last 7 days Authentication Flow Component Library Last month Initial Project Setup

## เลือกเปลี่ยน Theme Light / Dark / System ได้

Can you help me with a coding question?

Of course! I'd be happy to help with your coding question. What would you like to know?

How do I create a responsive layout with CSS Grid?

Creating a responsive layout with CSS Grid is straightforward. Here's a basic example:

```
// src/app/page.tsx
import { Button } from '@/components/ui/button'

export default function Page() {
  return (
    <main className='p-6'>
      <h1 className='text-2xl font-semibold'>Hello shadcn/ui</h1>
      <Button className='mt-4'>Press me</Button>
    </main>
  )
}
```

This creates a grid where:  
Columns automatically fit as many as possible  
Each column is at least 250px wide  
Columns expand to fill available space  
There's a 1rem gap between items  
Would you like me to explain more about how this works?

List top 5 frontend frameworks show in table

Ask anything

+ Search ...

s Samit samit@email.com

AI Genius AI Q Project roadmap discussion

+ New Chat Today API Documentation Review Frontend Bug Analysis Yesterday Database Schema Design Performance Optimization Last 7 days Authentication Flow Component Library Last month Initial Project Setup

## การแสดงผลแบบ Dark mode

Can you help me with a coding question?

Of course! I'd be happy to help with your coding question. What would you like to know?

How do I create a responsive layout with CSS Grid?

Creating a responsive layout with CSS Grid is straightforward. Here's a basic example:

```
// src/app/page.tsx
import { Button } from '@/components/ui/button'

export default function Page() {
  return (
    <main className='p-6'>
      <h1 className='text-2xl font-semibold'>Hello shadcn/ui</h1>
      <Button className='mt-4'>Press me</Button>
    </main>
  )
}
```

This creates a grid where:  
Columns automatically fit as many as possible  
Each column is at least 250px wide  
Columns expand to fill available space  
There's a 1rem gap between items  
Would you like me to explain more about how this works?

List top 5 frontend frameworks show in table

Ask anything

+ Search ...

s Samit samit@email.com



The screenshot shows a web-based AI Chatbot interface. On the left, a sidebar lists previous conversations with titles like "Genius AI", "New Chat", "Today", "Yesterday", and "Last 7 days". The main chat area displays a series of AI-generated responses to various topics such as "My Little Pony", "Peppa Pig", "Avatar: The Last Airbender", "Sesame Street", "Kazoops!", "Tayo the Little Bus", and "Hello World". A prominent pink callout box highlights the AI's response to "Hello World" with the text: "AI Chatbot แบบมีการเก็บประวัติ (History) และจดจำเราได้". At the bottom, there is a text input field with placeholder "Continue the conversation..." and a blue button labeled "Send message".

my-langchain-chatbot

```
route.ts
p > api > chat_06_history_optimize > route.ts > ...
/** 
 * =====
 * API Route สำหรับ Chat ที่มีการเก็บประวัติและ Optimize
 * =====
 *
 * ฟีเจอร์หลัก:
 * - เก็บประวัติการสนทนาใน PostgreSQL
 * - ทำ Summary เพื่อประหยัด Token
 * - Trim Messages เพื่อไม่ให้เกิน Token Limit
 * - Streaming Response สำหรับ Real-time
 * - จัดการ Session ID อัตโนมัติ
 */
import { NextRequest } from 'next/server'
import { ChatOpenAI } from '@langchain/openai'
import { ChatPromptTemplate, MessagesPlaceholder } from '@langchain/core/prompts'
import { toUIMessageStream } from '@ai-sdk/langchain'
import { createUIMessageStreamResponse, UIMessage } from 'ai'
import { PostgresChatMessageHistory } from '@langchain/community/stores/messages'
import { Pool } from 'pg'

import { BaseMessage, AIMessage, HumanMessage, SystemMessage, MessageContent } from 'langchain/messages'
import { trimMessages } from '@langchain/core/messages'
import { StringOutputParser } from '@langchain/core/output_parsers'
import { encodingForModel } from '@langchain/core/utils/tiktoken'

export const dynamic = 'force-dynamic'
export const maxDuration = 30

// =====
// การตั้งค่า PostgreSQL Connection Pool
Analyze route.ts and its dependencies
In 8 Col 1 (34 selected) Spaces
```

AI Chatbot with Langchain and Streamlit

localhost:3000/chat/be480662-1f14-43a5-801f-8d59faf0b5b6

# ให้ AI เชื่อมต่อ กับ Tool ภายนอก เช่นฐานข้อมูลของเรา เพื่อสอบถามข้อมูลของคุณ

เก็บสืบเนื่องมาสู่ห้องครับ

Today

สวัสดีครับ ร้านดังอยู่ที่ไหนครับ

Yesterday

Gaming Mouse ขายไปได้กี่ชิ้นแล้วครับ

มีมาส์สำหรับเกมส์บีบหรือไม่ครับ?

MacBook Pro M3 ราคาเท่าไหร่...

ขอรายชื่อประเภทพิมพ์เมื่อหลัง...

การ Hello World ในภาษา Asse...

Explain React hooks and wh...

ขอ best practice ในการออกแบบ...

Last 7 days

เฉลี่ยราคาสต็อกแบบสัปดาห์ ให้ฟ...

ขอเนื้อเพลง hotel califonia ขอ...

How do I create a responsiv...

ขอรายชื่อบัตรเดบิตในไทยตอนโล...

Help me debug this JavaScr...

How do I create a responsiv...

แนะนำอาหารไทยยอดนิยม 5 รายการ...

สวัสดีครับ แผนที่ สถานที่ บาร์...

ประวัติการขายของสินค้า "Gaming Mouse" มีดังนี้:

วันที่ขาย	จำนวนที่ขาย (ชิ้น)	ยอดขาย (บาท)
21/8/2568	2	5,600
26/8/2568	5	14,000

**สรุป:**

ขายรวม: 7 ชิ้น

ยอดขายรวม: 19,600 บาท

หากต้องการข้อมูลเพิ่มเติมหรือสอบถามค่าใช้จ่ายอื่นๆ สามารถสอบถามได้ค่ะ!

ยอดขายระหว่าง MacBook กับ mouse ต่างกันเท่าไหร่ครับ

ยอดขายรวมของสินค้าแต่ละรายการมีดังนี้:

**MacBook Pro M3:** 159,800 บาท

**Gaming Mouse:** 19,600 บาท

**การคำนวณความแตกต่าง:**

159,800 - 19,600 = 140,200 บาท

ตั้งนี้ ยอดขายระหว่าง MacBook Pro M3 กับ Gaming Mouse ต่างกัน 140,200 บาทค่ะ หากมีคำาณเพิ่มเติม สามารถสอบถามได้ค่ะ!

Continue the conversation...

+ Search ...

S Samit samit@email.com

localhost:3000/chat/be480662-1f14-43a5-801f-8d59faf0b5b6

AI Genius AI

สวัสดีครับ ร้านดังอยู่ที่ไหนครับ

+ New Chat

Today

สวัสดีครับ ร้านดังอยู่ที่ไหนครับ

Yesterday

Gaming Mouse ขายไปได้กี่ชิ้นแล้วครับ

มีมาส์สำหรับเกมส์บีบหรือไม่ครับ ...

MacBook Pro M3 ราคาเท่าไหร่...

ขอรายชื่อประเภทพิมพ์เมื่อหลัง...

การ Hello World ในภาษา Asse...

Explain React hooks and wh...

ขอ best practice ในการออกแบบ...

Last 7 days

เฉลี่ยราคาสต็อกแบบสัปดาห์ ให้ฟ...

ขอเนื้อเพลง hotel califonia ขอ...

How do I create a responsiv...

ขอรายชื่อบัตรเดบิตในไทยตอนโล...

Help me debug this JavaScr...

How do I create a responsiv...

แนะนำอาหารไทยยอดนิยม 5 รายการ...

สวัสดีครับ แผนที่ สถานที่ บาร์...

ร้าน "เทคโนโลยี สมาร์ท สโตร์" ตั้งอยู่ที่:  
ที่อยู่: 123 ถนนเทคโนโลยี แขวงบึงกุ่ม เขตดินถัง กรุงเทพมหานคร 10400  
โทรศัพท์: 02-123-4567  
อีเมล: info@smarttechstore.com  
เว็บไซต์: www.smarttechstore.com

หากต้องการข้อมูลเพิ่มเติมสามารถสอบถามได้เลยครับ!

iPhone 15 Pro Max ราคาเท่าไหร่?

ขออภัยครับ ไม่พบข้อมูลเกี่ยวกับสินค้า "iPhone 15 Pro Max" ในฐานข้อมูล หากต้องการสอบถามเกี่ยวกับสินค้ารุ่นอื่น ๆ ห้ามข้อมูลเพิ่มเติม สามารถแจ้งได้เลยครับ!

ขอ Macbook Pro บี้กครับ ราคาเท่าไหร่

สามารถ AI ให้ไปค้น  
ข้อมูลจากเอกสารองค์กรได้

Continue the conversation...

+ Search ...

S Samit samit@email.com

www.itgenius.co.th



ITGenius Default project

Dashboard Docs API S

Settings Your profile Organization General API keys Admin keys People Projects Billing Limits Usage Data controls Project General API keys Webhooks People Limits

API keys

+ Create new secret key

You have permission to view and manage all API keys in this organization.

Do not share your API key with others or expose it in the browser or other client-side code. To protect your account's security, OpenAI may automatically disable any API key that has leaked publicly.

View usage per API key on the [Usage page](#).

NAME	SECRET KEY	PROJECT ACCESS	CREATED BY	PERMISSIONS
ai-chatbot-langchain	sk-...MJgA	Default project	Samit Koyom	All
n8n-api-key	sk-...Ve8A	SampleOpenAP...	Samit Koyom	All
n8n sample	sk-...blcA	Default project	Samit Koyom	All
n8n test	sk-...iIYA	Default project	Samit Koyom	All

ใช้ Open AI หรือ Model Opensource ฟรีก็ได้

<> Cookbook

坛坛 Forum

OpenRouter Search /

Models Chat Rankings Docs

Meta: Llama 3.3 70B Instruct (free)

meta-llama/llama-3.3-70b-instruct:free

Created Dec 6, 2024 | 65,536 context | \$0/M input tokens | \$0/M output tokens

The Meta Llama 3.3 multilingual large language model (LLM) is a pretrained and instruction tuned generative model in 70B (text in/text out). The Llama 3.3 instruction tuned text only model is optimized for multilingual dialogue use cases and outperforms many of the available open source and closed

Free Model weights

Overview Providers Apps Activity Uptime API

ใช้ Open AI หรือ Model Opensource ฟรีก็ได้

Providers for Llama 3.3 70B Instruct (free)

OpenRouter routes requests to the best providers that are able to handle your prompt size and parameters, with fallbacks to maximize uptime.

Sort by	
Venice	Latency 0.77s Throughput 121.1tps Uptime
Total Context 65.5K Max Output 65.5K Input Price \$0 Output Price \$0 Cache Read -- Cache Write -- Input Audio -- Input Audio Cache --	Latency 0.77s Throughput 121.1tps Uptime
Together	Latency 0.88s Throughput 146.2tps Uptime
Total Context 131.1K Max Output 2.0K Input Price \$0 Output Price \$0 Cache Read -- Cache Write -- Input Audio -- Input Audio Cache --	Latency 0.88s Throughput 146.2tps Uptime
Meta	Latency Throughput Uptime



A screenshot of a web browser window titled "Download Ollama on Windows". The URL in the address bar is "ollama.com/download/windows". The page content includes a navigation bar with links for "Models", "GitHub", "Discord", and "Turbo", and a search bar labeled "Search models". The main section is titled "Download Ollama" and features icons for "macOS", "Linux", and "Windows". The "Windows" icon is highlighted with a light gray background. Below the icons is a large black button with white text that reads "Download for Windows". A small note below the button states "Requires Windows 10 or later".

Download Ollama

macOS

Linux

Windows

Download for Windows

Requires Windows 10 or later



Qwen/Qwen2.5-7B-Instruct · [Hu](#)

huggingface.co/Qwen/Qwen2.5-7B-Instruct

Hugging Face Search models, datasets, users...

Models Datasets Spaces Docs Pricing

Qwen/Qwen2.5-7B-Instruct like 778 Follow Qwen 48.1k

Text Generation Transformers Safetensors English qwen2 chat conversational text-generation-inference

arxiv:2309.00071 arxiv:2407.10671 License: apache-2.0

Model card Files xet Community 22 Train Deploy Use this model

## Qwen2.5-7B-Instruct

[Qwen Chat](#)

### Introduction

Qwen2.5 is the latest series of Qwen large language models. For Qwen2.5, we release a number of base language models and instruction-tuned language models ranging from 0.5 to 72 billion parameters. Qwen2.5 brings the following improvements upon Qwen2:

- Significantly **more knowledge** and has greatly improved capabilities in **coding** and **mathematics**, thanks to our specialized expert models in these domains.

Downloads last month 11,201,081

Safetensors Model size 7.62B params Tensor type BF16 Chat template Files info

Inference Providers NEW Together AI Text Generation Examples

Run 15,000+ Models Instantly Inference Providers let you run inference on thousands of models served by our partners using a simple, unified, OpenAI-compatible serverless API ([Learn more](#)).

Samit Koyom's projects Hobby ai-chatbot-langchain

Find... Feedback

Overview Deployments Analytics Speed Insights Logs Observability Firewall Storage Flags Settings

## ai-chatbot-langchain

Repository Usage Domains Visit

### Production Deployment

Deployment  
ai-chatbot-langchain-6y2q9lbc-samit-koyoms-projects.vercel.app

Status Created  
Ready 4h ago by Ia

Domains  
ai-chatbot-langchain.vercel.app

Source  
main  
d740a66 initial commit

Deployment Configuration Fluid Compute Deployment Protection Skew Protection Cold Start Prevention

To update your Production Deployment, push to the main branch.

Deployments

Firewall 24h Enable Bot Protection Firewall is active

Observability 6h Edge Requests 161 Function Invocations 40 Error Rate 0%

Analytics Track visitors and page views Enable

Active Branches

ເພີ້ມແປ່ວ່າ (Deployment) ບໍລິຫານ Vercel Cloud ວິສ

ai-chatbot-langchain.vercel.app/chat/96da7afa-73c1-46eb-b129-fb6408232da3

Genius AI

+ New Chat

Today

ມີ macbook pro ລາຄາເກົ່າໄຮ້  
ສະເລັດຄົນ ສອບຄານເຮືອງແນວຕົດຄ...

ມີເມາສີເກີບນຶ່ງໄຣສາຍ ລາຄາ 2,800 ບາກ ຈຳນວນໃນສັດຖຸ 50 ຊົ່ວໂມງ ໂດຍບໍ່ມາລະເຊີຍດວ່າເປັນເມາສີທີ່ອຸບສົນຈັນໄວ້ຄວົນ

ຍອດຂາຍເມາສີເປັນອ່າງໂປ່ານ

ຂອງກົຍຄົນ ຂະນະນີ້ໄປສາມາຄເຫົ່າກົງບຸລຍອດຂາຍຂອງເມາສີເກີບນຶ່ງໄຣສາຍໄດ້ ກຽມບາລອງໃໝ່ອົກຄົ້ນໃນກາຍເລັ່ນ

ຍອດຂາຍຂອງ MacBook Pro M3 ມີປະວັດກາຍທີ່ໜັດ 2 ຮາຍຄາ ດັ່ງນີ້:

ວັນທີຂາຍ	ຈຳນວນທີ່ຂາຍ (ຊັ້ນ)	ຍອດຂາຍ (ບາກ)
20/8/2568	1	79,900
25/8/2568	1	79,900

ສຽບ:

ຂາຍຮວບ: 2 ຊົ່ວໂມງ  
ຍອດຂາຍຮວບ: 159,800 ບາກ ຄົນ

ເພີ້ມແປ່ວ່າ (Deployment) ບໍລິຫານ Vercel Cloud ວິສ

Continue the conversation...

+ Search ...



ສຕາບັນໄວ້ກົງເນື້ຍສ

www.itgenius.co.th



## Tech Stack



- **Frontend:** Next.js 15.5.0 (App Router)
- **UI Framework:** React 19.1.0
- **Styling:** Tailwind CSS 4
- **UI Components:**
  - shadcn/ui (New York style)
  - Prompt-kit UI Components
  - Radix UI (@radix-ui/react-\*)
  - Lucide React Icons
- **Authentication:**
  - Supabase Auth (@supabase/supabase-js, @supabase/ssr)
  - Password-based Authentication
- **Database:**
  - Supabase PostgreSQL with pgvector extension
  - Chat History Storage
  - User Session Management
  - Product & Sales Data (Tool Calling)
  - Vector Embeddings Storage (Document Search)
- **AI Integration:**
  - AI SDK v5 (@ai-sdk/react, @ai-sdk/langchain)
  - LangChain (@langchain/core, @langchain/openai, @langchain/community)
- **Tool Calling:** LangChain Agents with Supabase Tools
- **Document Processing:** Text & PDF loaders, Vector embeddings
- **Vector Search:** pgvector with cosine similarity
- **Language Model:** OpenAI GPT-4o-mini & text-embedding-3-small
- **Utilities:**
  - class-variance-authority (Component variants)
  - clsx & tailwind-merge (Conditional styling)
  - marked & react-markdown (Markdown rendering)
  - shiki (Syntax highlighting)
  - zod (Schema validation สำหรับ Tools)
  - remark-gfm & remark-breaks (Markdown extensions)
  - use-stick-to-bottom (Auto-scroll behavior)
  - d3-dsv (CSV parsing สำหรับ Document Loader)
- **TypeScript:** v5
- **Development:** ESLint 9



แก้ branch ให้เรียบตามได้อย่างง่าย

my-langchain-chatbot

EXPLORER: MY-LA... PROBLEMS PORTS OUTPUT DEBUG CONSOLE GITLENS TERMINAL

COMMIT GRAPH: MY-LANGCHAIN-CHATBOT

All Branches Search commits using natural language (11 for history), e.g. Show my commits from last month No results

BRANCH / TAG	GRAPH	COMMIT MESSAGE	AUTHOR	CHANGES	COMMIT DATE / TIME
08-document-loader-embedding-pgve...		update readme document loader	You	1	2 hours ago
09-rag		update welcome page	You	1	6 hours ago
10-deployment	✓	update front page	You	1	yesterday
07-tool-calling		10-deployment-finished	You	16	2 days ago
06-chat-history-optimize		09-rag-finished	You	8	2 days ago
05-dark-theme-ui		08-document-loader-embedding-pgvector-text-csv-pc	You	1	2 days ago
06-chat-history		08-document-loader-embedding-pgvector-text-csv-pc	You	4	2 days ago
04-prompt-kit-ui		08-document-loader-embedding-pgvector-text-csv	You	9	2 days ago
chat_01_start		07-tool-calling-finihed	You	9	2 days ago
chat_02_request		07-tool-calling-start	You	1	3 days ago
chat_03_template		06-chat-history-optimize-update-readme	You	1	3 days ago
chat_04_stream		06-chat-history-optimize-add-comment	You	6	3 days ago
chat_05_history		05-dark-theme-ui-complete update comment	You	3	3 days ago
chat_06_history_optimize		06-chat-history-update-comment	You	11	3 days ago
chat_07_tool_calling		06-chat-history-optimize-complete	You	11	3 days ago
chat_08_rag		06-chat-history-complete-update-linter	You	12	3 days ago
document_loader_embedding...		06-chat-history-complete-update-readme	You	1	4 days ago
health		06-chat-history-complete	You	20	4 days ago
test		06-integrate-api-to-ui-new-chat-complete	You	3	4 days ago
test-db		05-dark-theme-ui-complete #1	You	2	4 days ago
route.ts		05-dark-theme-ui-complete	You	7	4 days ago
auth		04-prompt-kit-ui-comple	You	38	4 days ago
chat					
test-history					
favicon.ico					
globals.css					
layout.tsx					
page.tsx					
components					
contexts					
hooks					
lib					



Home Workspaces API Network

Search Postman Ctrl K

Invite Upgrade No environment

Collections

Environments

Flows

History

Samit

Search collections

AIChatbotLangchain

- 01\_Next\_API
- 02\_Langchain\_Basic
  - POST 01\_/api/chat\_01\_start
  - POST 02\_/api/chat\_02\_request
  - POST 03\_/api/chat\_03\_template
  - POST 04\_/api/chat\_04\_stream
- 05\_Chat\_History
  - POST 01\_/api/chat\_05\_history?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9
  - GET 02\_/api/chat\_05\_history?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9 Copy
- 06\_Chat\_History\_Optimize
  - POST 01\_/api/chat\_06\_history\_optimize?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9
  - GET 02\_/api/chat\_06\_history\_optimize?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9 Copy
- 07\_Tool\_Calling
  - POST 01\_/api/chat\_07\_tool\_calling?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9
  - GET 02\_/api/chat\_07\_tool\_calling?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9 Copy
- Document\_Loader\_EMBEDDING\_pgVector
  - GET 01\_/api/document\_loader\_embedding\_pgvector/text\_csv
  - POST 02\_/api/document\_loader\_embedding\_pgvector/text\_csv
  - DEL 03\_/api/document\_loader\_embedding\_pgvector/text\_csv
  - PUT 04\_/api/document\_loader\_embedding\_pgvector/text\_csv
  - GET 05\_/api/document\_loader\_embedding\_pgvector/text\_csv\_pdf
- 08\_RAG
  - POST 01\_/api/chat\_08\_rag?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9
  - GET 02\_/api/chat\_08\_rag?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9 Copy
- DjangoWebSocket
- gofiber
- QR Menu App API

POST 04\_/api/chat\_04\_stream

Save Share

Send

Params Authorization Headers (9) Body Scripts Settings Cookies Beautify

Body Type: raw

```
1 {  
2   "messages": [  
3     {  
4       "id": "chat-id-001",  
5       "role": "user",  
6       "parts": [  
7         {  
8           "type": "text",  
9           "text": "สวัสดีครับ บริษัทของเรารักษาความปลอดภัยให้แผนกใหญ่มากที่สุดครับ"  
10        }  
11      ]  
12    }  
13  ]  
14 }
```

Response Hist Click Send to get a response

\_PAYLOAD\_

แยก API end point ไว้ให้กดสอบง่าย

Postbot Runner Start Proxy Cookies Vault Trash



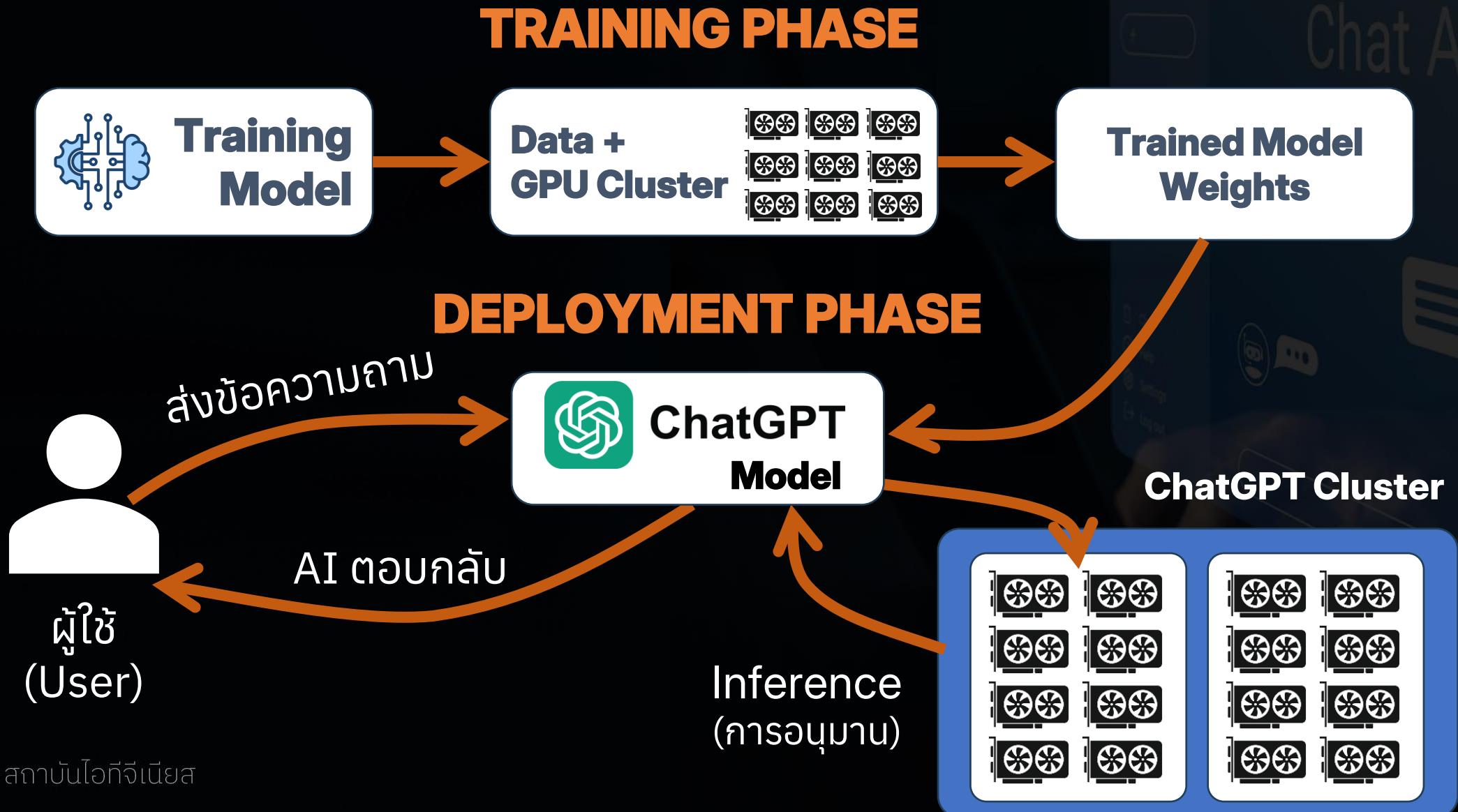
## 1. ภาพรวม AI Chatbot กับ Langchain.js



# การสร้างและการทำงานของ Gen AI Model



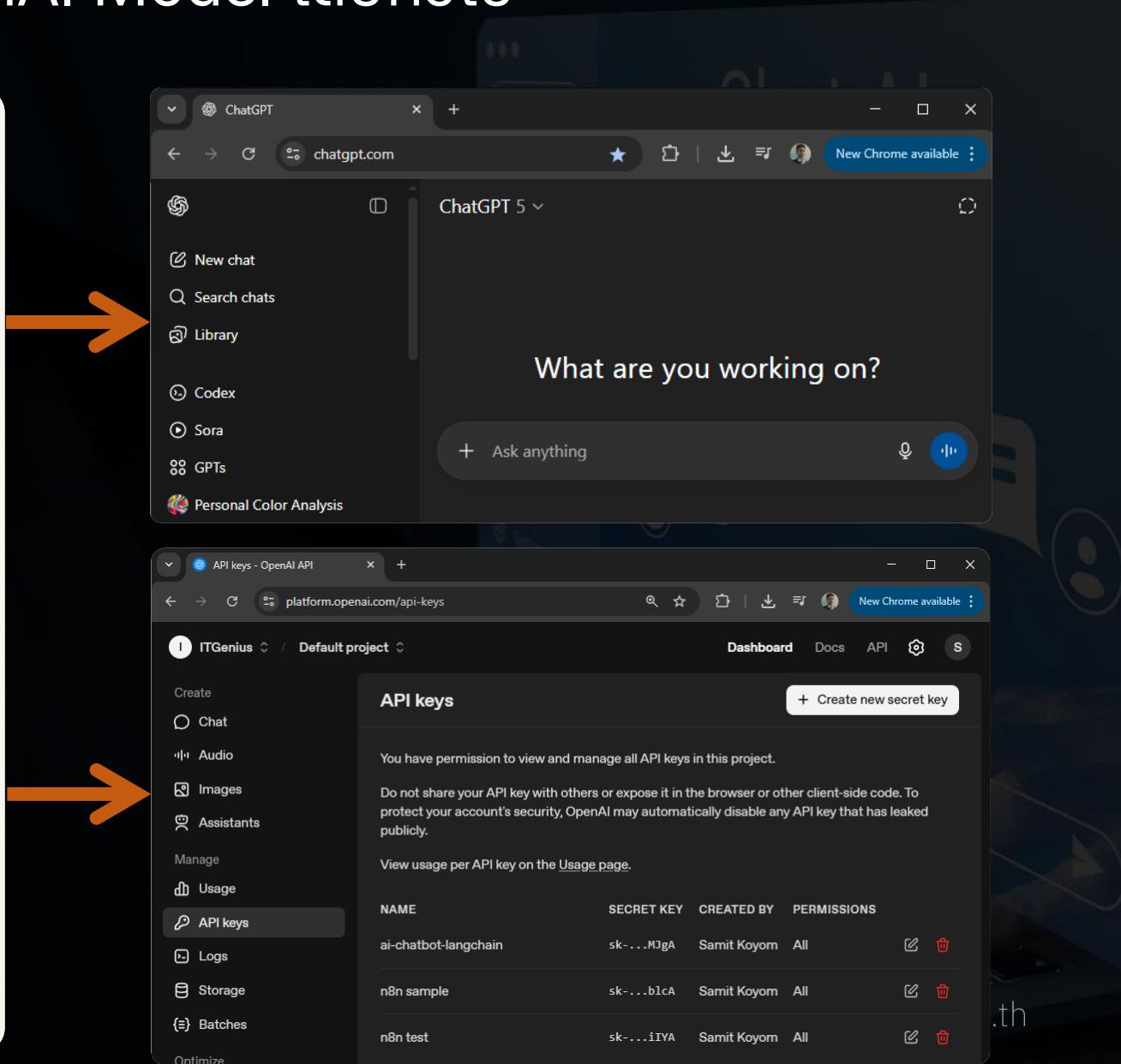
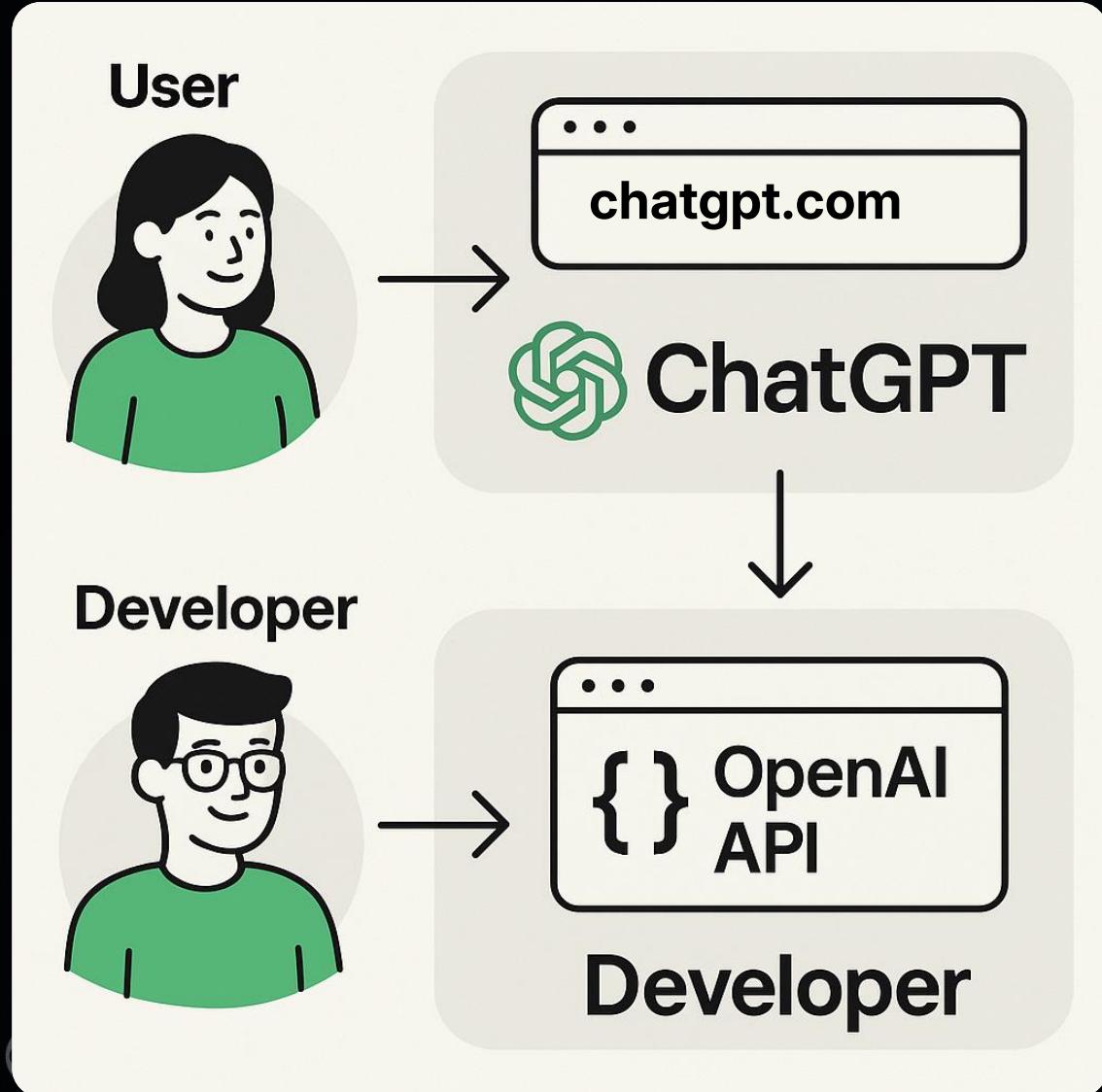
# การสร้างและการทำงานของ AI Model



# รูปแบบการเรียนรู้ใช้งาน Gen AI Model



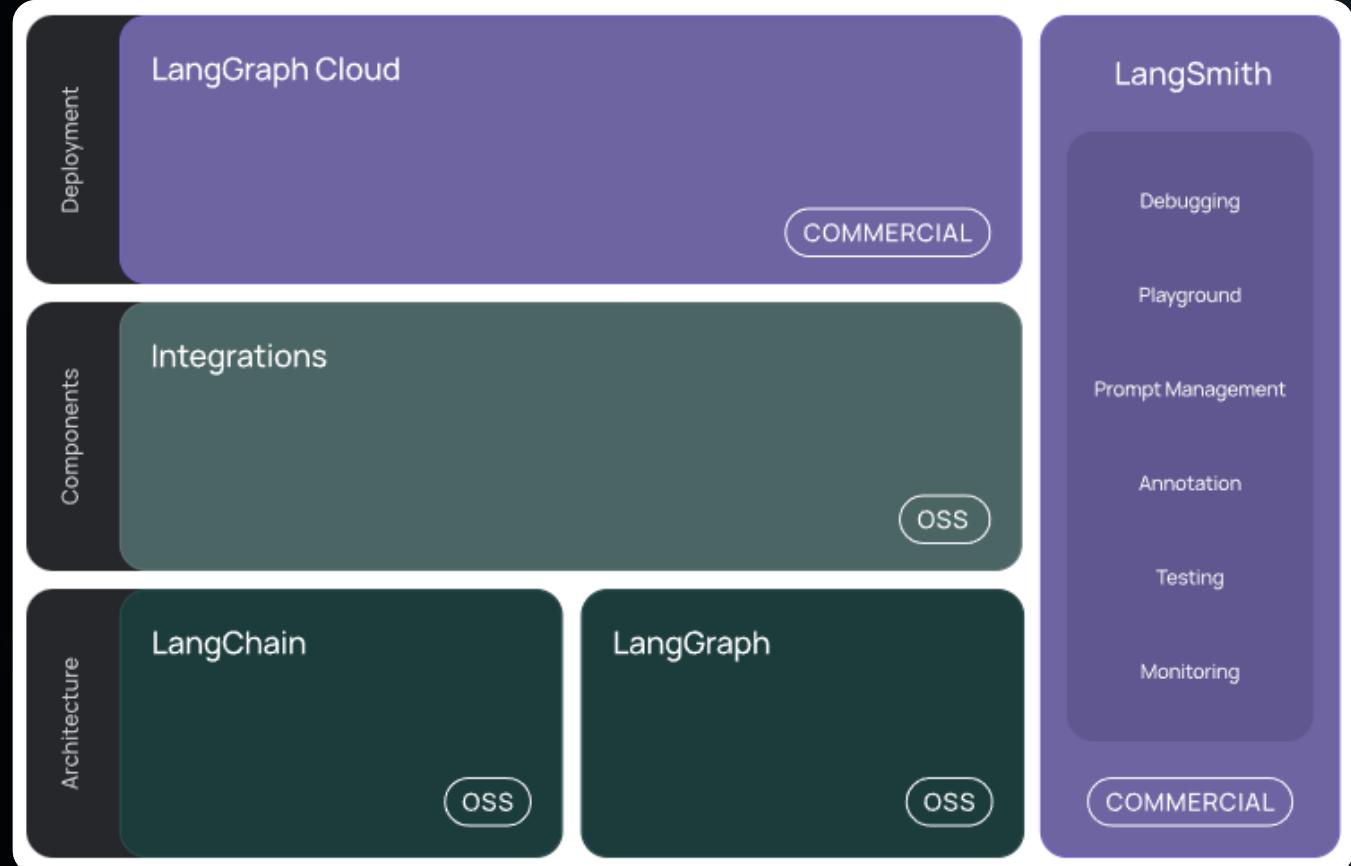
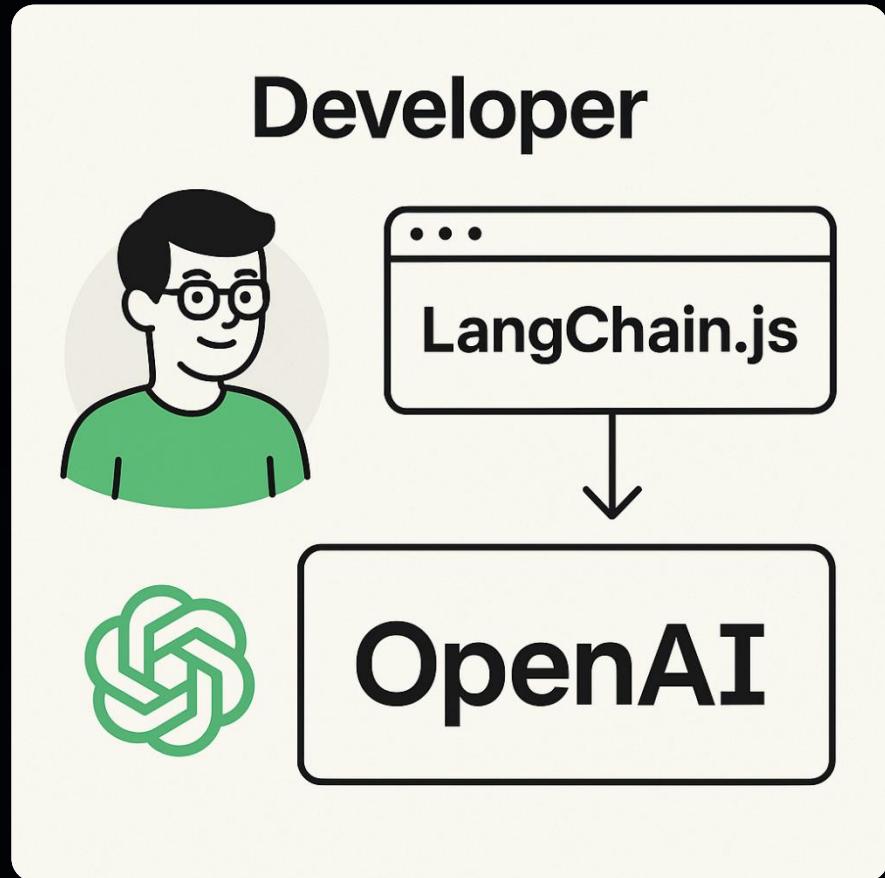
# การเข้าใช้งาน GenAI Model โดยก้าวไป

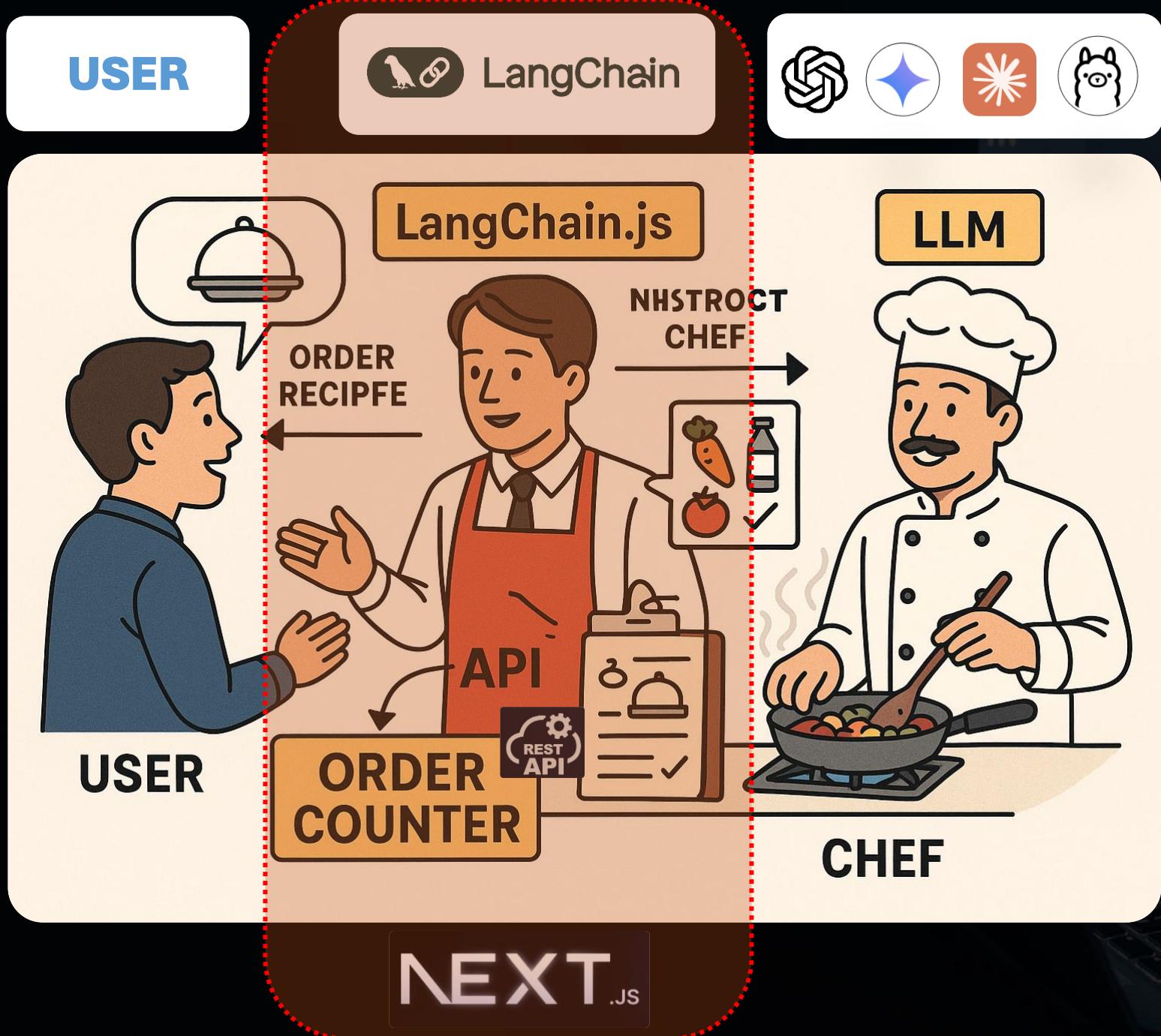


# การทำงานของ Langchain กับ AI Model



# นักพัฒนาเรียกทำงานกับ GenAI Model ผ่าน Langchain





AI SDK UI



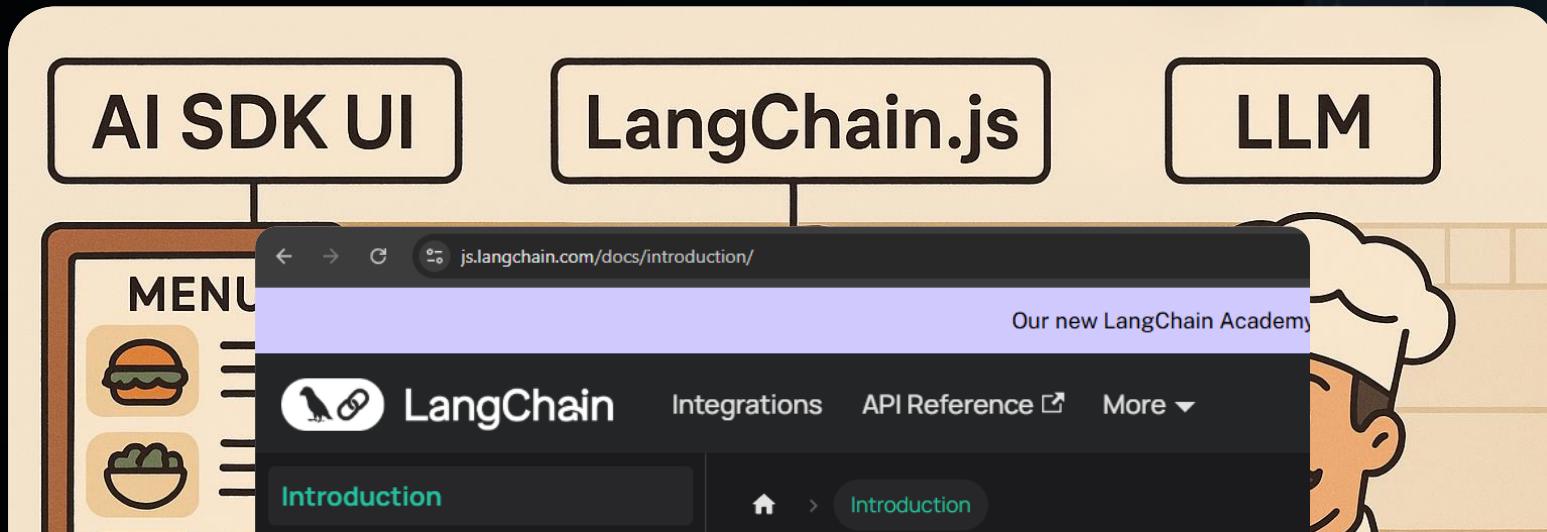
LangChain



AI SDK UI

LangChain.js

LLM



**The AI Toolkit for TypeScript**

From the creators of Next.js, the AI SDK is a free open-source library that gives you the tools you need to build AI-powered products.

Get Started    \$ npm i ai    Visit Playground

Trusted by builders at

This screenshot shows the AI SDK UI documentation page. It features a dark header with the title 'The AI Toolkit for TypeScript'. Below the header, there is a brief introduction and three calls-to-action: 'Get Started', '\$ npm i ai', and 'Visit Playground'. At the bottom, it says 'Trusted by builders at' followed by a list of logos for various companies.

**Introduction**

LangChain is a framework for building AI applications.

LangChain simplifies every stage of AI development:

- **Development:** Build your applications with simple, composable integrations. Use LangChain's powerful API to connect AI models to your application.
- **Productionization:** Use LangChain's built-in infrastructure to host your AI applications in the cloud.

Our new LangChain Academy is now available!

API keys - OpenAI API

ITGenius / Default project

Create

- Chat
- Audio
- Images
- Assistants

Manage

- Usage
- API keys
- Logs
- Storage
- Batches

API keys

You have permission to view and manage all API keys in this project.

Do not share your API key with others or expose it in the browser or other client-side code. To protect your account's security, OpenAI may automatically disable any API key that has leaked publicly.

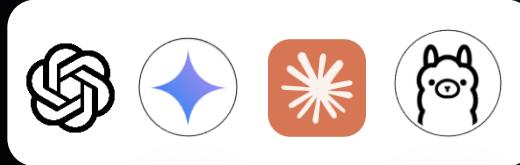
View usage per API key on the Usage page.

NAME	SECRET KEY	CREATED BY	PERMISSIONS
ai-chatbot-langchain	sk-...MjgA	Samit Koyom	All
n8n sample	sk-...blcA	Samit Koyom	All
n8n test	sk-...iiYA	Samit Koyom	All

Optimize

This screenshot shows the LangChain.js documentation page and the OpenAI API keys management interface. The documentation page features a large 'Introduction' section with a sub-section about LangChain's purpose and how it simplifies AI development. It also lists two bullet points: 'Development' and 'Productionization'. The OpenAI API keys management interface shows a sidebar with 'Create' and 'Manage' sections, and a main table listing three API keys: 'ai-chatbot-langchain', 'n8n sample', and 'n8n test'. Each row in the table includes columns for 'NAME', 'SECRET KEY', 'CREATED BY', and 'PERMISSIONS'. There is also an 'Optimize' button at the bottom right.

# Local LLM vs LLM providers



สถาบันไอทีเจเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)

# Local LLM



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)



Ollama

Thought for 1.8 seconds

New Chat

Settings

This week

ประเทศไทยมีกี่จังหวัด

สวัสดีครับ

จังหวัดของภาคกลาง (ประเทศไทย) – ตาราง 21 จังหวัด (รวมกรุงเทพมหานครเป็นเขตบริหารพิเศษ)

ลำดับ	ชื่อจังหวัด (อังกฤษ)	ชื่อจังหวัด (ไทย)
1	Nakhon Pathom	จังหวัดนนทบุรี
2	Nonthaburi	จังหวัดนนทบุรี
3	Pathum Thani	จังหวัดปทุมธานี
4	Samut Prakan	จังหวัดสมุทรปราการ
5	Samut Sakhon	จังหวัดสมุทรสาคร
6	Samut Songkhram	จังหวัดสมุทรสงคราม
7	Kamphaeng Phet	จังหวัดกำแพงเพชร
8	Nakhon Sawan	จังหวัดนนทบุรี
9	Phetchabun	จังหวัดเพชรบูรณ์
10	Phichit	จังหวัดพิจิตร
11	Phitsanulok	จังหวัดพิษณุโลก
12	Sukhothai	จังหวัดสุโขทัย

Send a message

⊕ Turbo gpt-oss:20b ↑



# Qwen: Qwen3 8B

# Local Host

Home Workspaces API Network

Ctrl K

Invite Settings Refresh Upgrade No environment

**POST 01/\_api/chat\_01\_start**

HTTP AIChatbotLangchain / 02\_Langchain\_Basic / 01/\_api/chat\_01\_start

POST {{baseURL}} /api/chat\_01\_start

Params Authorization Headers (16) Body Scripts Settings

Body: none form-data x-www-form-urlencoded raw binary GraphQL

This request does not have a body.

Send Cookies

POST 01/\_api/chat\_01\_start

POST 02/\_api/chat\_02\_request

POST 03/\_api/chat\_03\_template

POST 04/\_api/chat\_04\_stream

05\_Chat\_History

06\_Chat\_History\_Optimize

07\_Tool\_Calling

Document\_Loader\_EMBEDDING\_pgVector

08\_RAG

DjangoWebSocket

gofiber

QR Menu App API

Body Cookies (1) Headers (6) Test Results

200 OK 2 m 2.52 s 3.38 KB Save Response

{ } JSON Preview Visualize

```

1 "content": "<think>\nOkay, the user is asking about the budget for this year. I need to provide a clear and concise answer. Let me start by acknowledging their question. Then, I should outline the main aspects of the budget, like revenue targets, cost management, and key areas of investment. It's important to mention the overall direction, maybe something about growth and efficiency. Also, I should invite them to ask more questions if they need specific details. Keep the tone friendly and professional. Let me structure it step by step to make sure all key points are covered without being too technical.\n</think>\n\nสวัสดีครับ 😊\n\nปัจจุบันเราได้รับงบประมาณ 10-15% เพื่อพัฒนาธุรกิจให้เติบโตอย่างมีประสิทธิภาพครับ โดยนี่เป้าหมายหลักก็คือ: \n\n**การสร้างรายได้ที่ยั่งยืน** - คาดว่ารายได้จะเพิ่มขึ้น 10-15% เมื่อเทียบกับปีที่แล้ว โดยแบ่งการรายได้ตามแหล่งเพิ่มส่วนแบ่งตลาดในธุรกิจหลัก \n\n**การจัดการต้นทุนอย่างมีประสิทธิภาพ** - ควบคุมต้นทุนอย่างดีเพื่ออยู่ในระดับที่สมเหตุสมผล โดยมุ่งเน้นประสิทธิภาพของห้องงานและกระบวนการผลิต \n\n**การพัฒนาผลิตภัณฑ์และการตลาด** - คาดว่าจะมีการพัฒนาเทคโนโลยีและระบบงานใหม่เพื่อเพิ่มประสิทธิภาพการดำเนินงาน \n\n**ความยั่งยืนทางการเงิน** - รักษาระดับหนี้สินและกำไรอย่างดี เพื่อให้ธุรกิจสามารถรับมือกับความไม่แน่นอนในตลาดได้อย่างมั่นคง \n\nหากหากคุณมีคำถามเพิ่มเติมเกี่ยวกับแผนการเงินเฉพาะส่วนหรือต้องการข้อมูลเพิ่มเติม ยินดีร่วมตอบครับ 😊 \n\nคุณสนใจเรื่องใดเป็นพิเศษไหมครับ?", "usedModel": "qwen3:8b"
  
```



GPT-OSS

## Local Host

The screenshot shows the Postman interface with the following details:

- Header Bar:** Home, Workspaces, API Network, Search Postman, Ctrl + K, Invite, Settings, Notifications (1), Help, Upgrade.
- Left Sidebar:** Collections (Search collections), AIChatbotLangchain, 01\_Next\_API, 02\_Langchain\_Basic (selected), POST 01/\_api/chat\_01\_start, POST 02/\_api/chat\_02\_request, POST 03/\_api/chat\_03\_template, POST 04/\_api/chat\_04\_stream, 05\_Chat\_History, 06\_Chat\_History\_Optimize, 07\_Tool\_Calling, Document\_Loader\_EMBEDDING\_pgVector, 08\_RAG, DjangoWebSocket, gofiber, QR Menu App API.
- Request Details:** Method: POST, URL: {{baseURL}}/\_api/chat\_01\_start, Headers: 16, Body: none (selected), Params, Authorization, Scripts, Settings.
- Response Preview:** Status: 200 OK, Time: 3 m 35.82 s, Size: 3.42 KB, Save Response. The response body is a JSON object with "content" and "usedModel" fields.
- Content of Response Body:**

```
1 {  
2   "content": "สวัสดีครับ\ngานบริษัทเรา มีภาระรวมดังนี้ครับ\ng|\t รายการ | จำนวน (บาท) | เปอร์เซ็นต์ของงบรวม |\ng|-----|  
-----|\ng| **งบประมาณรวม** | 1 200 000 000 | 100 % |\ng| **รายได้** | 1 350 000 000 | 112 % |\ng| **ค่าใช้จ่าย** |  
1 200 000 000 | 100 % |\ng| **กำไรสุทธิ** | 150 000 000 | 12 % |\ng|\#\#\# การจัดสรรงบประมาณ\ng|\t แผนก | งาน (บาท) | เปอร์เซ็นต์ |\ng|-----|  
-----|\ng| การตลาด | 250 000 000 | 20 % |\ng| กำลังผลิต | 300 000 000 | 25 % |\ng| กำไรสุทธิ & พัฒนา | 150 000 000 | 12.5 % |\ng| ฝ่ายบริหาร &  
การเงิน | 100 000 000 | 8.3 % |\ng| ฝ่ายหัวหน้ากลุ่ม | 80 000 000 | 6.7 % |\ng| ค่าธรรมเนียม & ค่าบริการ | 120 000 000 | 10 % |\ng| ค่าตอบแทนพนักงาน  
(เงินส. ลา) | 100 000 000 | 8.3 % |\ng| **อื่น ๆ** | 100 000 000 | 8.3 % |\ng|\#\#\# จุดเด่นและความเปลี่ยนแปลง\ng1. **เพิ่มงบการตลาด 15 %**  
เพิ่อสนับสนุนความยั่งยืนด้วยก้าวแรกของการขยายตลาดใหม่ ๆ \ng2. **ลดค่าใช้จ่ายด้านการผลิต 5 %** ผ่านการปรับปรุงกระบวนการผลิตให้เกิดประโยชน์\ng3.  
**เพิ่มงบ R&D 10 %** เพื่อพัฒนาผลิตภัณฑ์ใหม่และขยายตัวผลิตภัณฑ์ที่มีอยู่现有的 \ng4. **กำไรสุทธิเพิ่มขึ้น 12 %** จากการควบคุมต้นทุนและเพิ่มยอดขาย\ng|\#\#\#  
ค่าแนะนำสำหรับพนักงาน\ng|- หากต้องการรายละเอียดเพิ่มเติมเกี่ยวกับงบประมาณของแผนกของคุณ หรือมีข้อเสนอแนะในการปรับใช้หัวหน้ากลุ่ม  
กรุณาติดต่อฝ่ายการเงินโดยตรง \ng|- โปรดตรวจสอบว่าการใช้จ่ายของคุณสอดคล้องกับงบประมาณที่กำหนดไว้  
และส่งรายงานการใช้จ่ายตามกำหนดเวลา\ng|\tหากมีคำขอเพิ่มเติมหรือยกเว้นลักษณะใดในส่วนใดเป็นพิเศษ ยินดีช่วยเสมอครับ!",  
3   "usedModel": "gpt-oss:latest"  
4 }
```

# Local LLM



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)



# Create GPU Droplet

Choose a datacenter region

 New York • Datacenter 2 • NYC2

Choose an image

OS

1-click Models

Custom Images

AI/ML Ready

Recommended: Linux bundled with [required GPU Drivers](#)

Inference Optimized

Deploy any model faster with production-grade-performance

Ubuntu

24.04 (LTS) x64

Fedora

42 x64

Debian

12 x64

CentOS

9 Stream x64

AlmaLinux

AlmaLinux 9

Rocky Linux

9 x64

## Summary

### GPU

Type: NVIDIA RTX4000 ADA

GPU: 1

VRAM: 20 GB

vCPU: 8

RAM: 32 GB

Boot Disk: 500 GB NVMe SSD

\$0.76/hr

### Total cost

\$0.76/hour

**Create GPU Droplet**

Want to maximize efficiency and cost?

Programmatically manage GPUs in a repeatable and re-usable way with our API. [Create via API](#)



< > 1 of 2 open incidents Droplet Connectivity To learn more, [check our status page](#).

Search by resource name or public IP (Ctrl+B)

Create ? My Team Estimated costs: \$11.11

PROJECTS

MANAGE

App Platform

Agent Platform New

Droplets

GPU Droplets New

Functions

Kubernetes

Volumes Block Storage

Databases

Spaces Object Storage

Container Registry

Backups & Snapshots

Networking

Monitoring

SaaS Add-Ons

By DigitalOcean

Billing

Support

Settings

ubuntu-gpu-4000adax

← Back to GPU Droplets

Web Console Actions

Active • Playground • TOR1 • default-tor1 • Ubuntu 24.04 (LTS) x64

Getting to know your GPU Droplet

Even if your GPU Droplet is powered off, you will still be billed as GPU, CPU, and associated resources are still reserved. To avoid charges, be sure to destroy the instance if it's no longer in use. Before destruction, you can take a snapshot to return back to.

Overview Insights Networking Volumes Backups & Snapshots Activity Settings

TOTAL GPU DROPLET COST

\$0.76 / hour

This is an estimation based on current configuration, you can see the [cost to date in billing](#).

Configuration Details

AUTOMATED BACKUPS

Reduce the stress of failure with automated droplet backups. They can be used for restoring lost or corrupt data and creating new Droplets.

Setup Automated Backups

Connection Details

Public IPv4 138.197.136.10

Private IP 10.118.0.2

Having trouble connecting? [Check out our troubleshooting guide](#).

Enhance Your Droplet's Network:

Improve security and ensure IP Address portability



pwsh in Samit x root@ubuntu-gpu-4000adax: +

(env) root@ubuntu-gpu-4000adax:/home/vllm\_project# nvidia-smi

Mon Sep 8 11:31:30 2025

NVIDIA-SMI 535.247.01			Driver Version: 535.247.01		CUDA Version: 12.2		
GPU	Name	Persistence-M	Bus-Id	Disp.A	Volatile	Uncorr.	ECC
Fan	Temp	Pwr:Usage/Cap		Memory-Usage	GPU-Util	Compute M.	MIG M.
0	NVIDIA RTX 4000 Ada Gene...	Off	00000000:01:00.0	0MiB / 20475MiB	0%	Default	Off
30%	42C	P0	31W / 130W				N/A

Processes:

GPU	GI	CI	PID	Type	Process name	GPU Memory Usage
ID	ID					
No running processes found						
(env) root@ubuntu-gpu-4000adax:/home/vllm_project#						



```
(env) root@ubuntu-gpu-4000adax:/home/vllm_project# python -m vllm.entrypoints.openai.api_server --model "Qwen/Qwen2.5-7B-Instruct"
INFO 09-08 16:20:51 [__init__.py:241] Automatically detected platform cuda.
(APIServer pid=9806) INFO 09-08 16:20:54 [api_server.py:1805] vLLM API server version 0.10.1.1
(APIServer pid=9806) INFO 09-08 16:20:54 [utils.py:326] non-default args: {'model': 'Qwen/Qwen2.5-7B-Instruct'}
(APIServer pid=9806) INFO 09-08 16:21:06 [__init__.py:711] Resolved architecture: Qwen2ForCausalLM
(APIServer pid=9806) `torch_dtype` is deprecated! Use `dtype` instead!
(APIServer pid=9806) INFO 09-08 16:21:07 [__init__.py:1750] Using max model len 32768
(APIServer pid=9806) INFO 09-08 16:21:12 [scheduler.py:222] Chunked prefill is enabled with max_num_batched_tokens=2048.
INFO 09-08 16:21:17 [__init__.py:241] Automatically detected platform cuda.
(EngineCore_0 pid=9886) INFO 09-08 16:21:21 [core.py:636] Waiting for init message from front-end.
(EngineCore_0 pid=9886) INFO 09-08 16:21:21 [core.py:74] Initializing a V1 LLM engine (v0.10.1.1) with config: model='Qwen/Qwen2.5-7B-Instruct', speculative_config=None, tokenizer='Qwen/Qwen2.5-7B-Instruct', skip_tokenizer_init=False, tokenizer_mode=auto, revision=None, override_neuron_config={}, tokenizer_revision=None, trust_remote_code=False, dtype=torch.bfloat16, max_seq_len=32768, download_dir=None, load_format=auto, tensor_parallel_size=1, pipeline_parallel_size=1, disable_custom_all_reduce=False, quantization=None, enforce_eager=False, kv_cache_dtype=auto, device_config=cuda, decoding_config=DecodingConfig(backend='auto', disable_fallback=False, disable_any whitespace=False, disable_additional_properties=False, reasoning_backend=''), observability_config=ObservabilityConfig(show_hidden_metrics_for_version=None, otlp_traces_endpoint=None, collect_detailed_traces=None), seed=0, served_model_name=Qwen/Qwen2.5-7B-Instruct, enable_prefix_caching=True, chunked_prefill_enabled=True, use_async_output_proc=True, pooler_config=None, compilation_config={"level":3,"debug_dump_path":","cache_dir":","backend":","custom_ops":[],"splitting_ops":["vllm.unified_attention","vllm.unified_attention_with_output","vllm.mamba_mixer2"],"use_inductor":true,"compile_sizes":[],"inductor_compile_config":{"enable_auto_functionalized_v2":false}, "inductor_passes":{},"cudagraph_mode":1,"use_cudagraph":true,"cudagraph_num_of_warmups":1,"cudagraph_capture_sizes":[512,504,496,488,480,472,464,456,448,440,432,424,416,408,400,392,384,376,368,360,352,344,336,328,320,312,304,296,288,280,272,264,256,248,240,232,224,216,208,200,192,184,176,168,160,152,144,136,128,120,112,104,96,88,80,72,64,56,48,40,32,24,16,8,4,2,1],"cudagraph_copy_inputs":false,"full_cuda_graph":false,"pass_config":{},"max_capture_size":512,"local_cache_dir":null}}
(EngineCore_0 pid=9886) INFO 09-08 16:21:23 [parallel_state.py:1134] rank 0 in world size 1 is assigned as DP rank 0, PP rank 0, TP rank 0, EP rank 0
(EngineCore_0 pid=9886) WARNING 09-08 16:21:23 [topk_topp_sampler.py:61] FlashInfer is not available. Falling back to the PyTorch-native implementation of top-p & top-k sampling. For the best performance, please install FlashInfer.
(EngineCore_0 pid=9886) INFO 09-08 16:21:23 [gpu_model_runner.py:1953] Starting to load model Qwen/Qwen2.5-7B-Instruct...
(EngineCore_0 pid=9886) INFO 09-08 16:21:23 [gpu_model_runner.py:1985] Loading model from scratch...
(EngineCore_0 pid=9886) INFO 09-08 16:21:23 [cuda.py:328] Using Flash Attention backend on V1 engine.
(EngineCore_0 pid=9886) INFO 09-08 16:21:24 [weight_utils.py:296] Using model weights format ['*.safetensors']
Loading safetensors checkpoint shards: 0% Completed | 0/4 [00:00<?, ?it/s]
Loading safetensors checkpoint shards: 25% Completed | 1/4 [00:00<00:01, 2.10it/s]
Loading safetensors checkpoint shards: 50% Completed | 2/4 [00:01<00:01, 1.98it/s]
Loading safetensors checkpoint shards: 75% Completed | 3/4 [00:01<00:00, 2.02it/s]
Loading safetensors checkpoint shards: 100% Completed | 4/4 [00:02<00:00, 1.98it/s]
Loading safetensors checkpoint shards: 100% Completed | 4/4 [00:02<00:00, 1.99it/s]
```



```
pwsh in Samit      x  root@ubuntu-gpu-4000adax: ~ + ^
```

(APIServer pid=9806) INFO 09-08 16:21:41 [launcher.py:44] Route: /v1/rerank, Methods: POST  
(APIServer pid=9806) INFO 09-08 16:21:41 [launcher.py:44] Route: /v2/rerank, Methods: POST  
(APIServer pid=9806) INFO 09-08 16:21:41 [launcher.py:44] Route: /scale\_elastic\_ep, Methods: POST  
(APIServer pid=9806) INFO 09-08 16:21:41 [launcher.py:44] Route: /is\_scaling\_elastic\_ep, Methods: POST  
(APIServer pid=9806) INFO 09-08 16:21:41 [launcher.py:44] Route: /invocations, Methods: POST  
(APIServer pid=9806) INFO 09-08 16:21:41 [launcher.py:44] Route: /metrics, Methods: GET  
(APIServer pid=9806) INFO: Started server process [9806]  
(APIServer pid=9806) INFO: Waiting for application startup.  
(APIServer pid=9806) INFO: Application startup complete.  
(APIServer pid=9806) WARNING: Invalid HTTP request received.  
(APIServer pid=9806) INFO: 27.145.115.105:2124 - "GET / HTTP/1.1" 404 Not Found  
(APIServer pid=9806) INFO: 27.145.115.105:2124 - "GET /favicon.ico HTTP/1.1" 404 Not Found  
(APIServer pid=9806) INFO: 27.145.115.105:2103 - "GET / HTTP/1.1" 404 Not Found  
(APIServer pid=9806) INFO: 27.145.115.105:2103 - "GET / HTTP/1.1" 404 Not Found  
(APIServer pid=9806) INFO: 27.145.115.105:2123 - "GET /v1/chat/completions HTTP/1.1" 405 Method Not Allowed  
(APIServer pid=9806) INFO: 27.145.115.105:2463 - "GET / HTTP/1.1" 404 Not Found  
(APIServer pid=9806) INFO 09-08 16:26:25 [chat\_utils.py:470] Detected the chat template content format to be 'string'. You can set `--chat-template-content-format` to override this.  
(APIServer pid=9806) INFO: 27.145.115.105:2464 - "POST /v1/chat/completions HTTP/1.1" 200 OK  
(APIServer pid=9806) INFO 09-08 16:26:31 [loggers.py:123] Engine 000: Avg prompt throughput: 4.2 tokens/s, Avg generation throughput: 5.8 tokens/s, Running: 0 reqs, Waiting: 0 reqs, GPU KV cache usage: 0.0%, Prefix cache hit rate: 0.0%  
(APIServer pid=9806) INFO 09-08 16:26:41 [loggers.py:123] Engine 000: Avg prompt throughput: 0.0 tokens/s, Avg generation throughput: 0.0 tokens/s, Running: 0 reqs, Waiting: 0 reqs, GPU KV cache usage: 0.0%, Prefix cache hit rate: 0.0%

POST http://138.197.136.10:8000/v1/chat/completions

POST http://138.197.136.10:8000/v1/chat/completions

Params Authorization Headers (9) Body Scripts Settings Cookies Beautify

Body

```
1 {  
2   "model": "Qwen/Qwen2.5-7B-Instruct",  
3   "messages": [  
4     {"role": "user", "content": "สวัสดี! คุณคือใคร?"}  
5   ]  
6 }
```

200 OK 3.87 s 990 B Save Response

Body Cookies Headers (4) Test Results

{ } JSON Preview Visualize

```
1 {  
2   "id": "chatmpl-47cb50135d85450087ecd80ee6ecbe1a",  
3   "object": "chat.completion",  
4   "created": 1757348785,  
5   "model": "Qwen/Qwen2.5-7B-Instruct",  
6   "choices": [  
7     {  
8       "index": 0,  
9       "message": {  
10         "role": "assistant",  
11         "content": "สวัสดี! ฉันเป็น Qwen เป็นโมเดล AI ที่สร้างโดย Alibaba Cloud ฉันสามารถช่วยตอบคำถาม สนทนา และให้ข้อมูลค่าๆ ได้ครับ/ค่ะ",  
12         "refusal": null,  
13         "annotations": null,  
14         "audio": null,  
15         "function_call": null,  
16         "tool_calls": [],  
17         "reasoning_content": null  
18       },  
19       "logprobs": null,  
20       "finish_reason": "stop",  
21       "stop_reason": null  
22     },  
23   ],  
24   "service_tier": null,  
25   "system_fingerprint": null,
```



# LLM providers

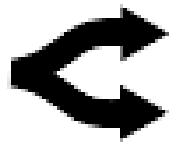


OpenRouter



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)



OpenRouter



Ollama



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)



# Qwen: Qwen3 8B

 OpenRouter

Home Workspaces API Network

Search Postman Ctrl K

Invite Settings Notifications Upgrade No environment

Collections Environments Flows History

Samit

POST 01/\_api/chat\_01\_start

HTTP AlChatbotLangchain / 02\_Langchain\_Basic / 01/\_api/chat\_01\_start

POST {{baseURL}}/\_api/chat\_01\_start

Params Authorization Headers (16) Body Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

This request does not have a body

Send

POST 01/\_api/chat\_01\_start

POST 02/\_api/chat\_02\_request

POST 03/\_api/chat\_03\_template

POST 04/\_api/chat\_04\_stream

05\_Chat\_History

06\_Chat\_History\_Optimize

07\_Tool\_Calling

Document\_Loader\_EMBEDDING\_pgVector

08\_RAG

DjangoWebSocket

gofiber

QR Menu App API

Body Cookies (1) Headers (6) Test Results

200 OK 20.49 s 3.64 KB Save Response

{ } JSON Preview Visualize

```

1 {
2   "content": "สวัสดีครับ! ขอบคุณที่ส่องความเกี่ยวกับงบประมาณปีนี้ครับ สำหรับปีนี้บริษัทได้จัดทำงบประมาณรวมทั้งหมด **1,200 ล้านบาท**  

  ซึ่งเพิ่มขึ้นจากปีที่แล้วประมาณ **10%** เพื่อรองรับเป้าหมายการเติบโตและโอกาสใหม่ๆ ที่เกิดขึ้นในตลาด。/\ก/\ก/## รายละเอียดงบประมาณหลัก:  

  **ค่าใช้จ่ายในการดำเนินงาน (Operational Costs):** \k - เพิ่มขึ้น **8%** เมื่อจากต้นทุนวัสดุคงเหลือและค่าแรงเพิ่มขึ้นจากภาวะเงินฟื้น  

  รวมถึงการลงทุนในระบบดิจิทัลเพิ่มเติมของบริษัทฯ ที่ต้องใช้เงินจำนวนมาก。/\n2. **การวิจัยและพัฒนา (R&D):** \k - ได้รับงบประมาณเพิ่มขึ้น **15%**  

  เพื่อสนับสนุนโครงการวิจัยและพัฒนาอย่างต่อเนื่อง และการพัฒนาผลิตภัณฑ์ที่ตอบโจทย์ลูกค้าในอนาคต。/\n3. **การตลาด (Marketing):** \k - เพิ่มขึ้น **12%**  

  เพื่อยกระดับการตลาดในช่องทางดิจิทัล และเพิ่มการรับรู้แบรนด์ในตลาดใหม่。/\n4. **การขยายตัว (Expansion):** \k - แผนขยายสาขาในกรุงเทพฯ  

  และการสร้างศูนย์เทคโนโลยีในจังหวัดเชียงใหม่ ได้รับงบประมาณ **200 ล้านบาท** สำหรับการลงทุนในโครงสร้างพื้นฐาน。/\n/\k/## จุดเด่นสำคัญ: \k- **ความยั่งยืน (Sustainability):** บริษัทมุ่งมั่นการลดต้นทุนพลังงานและเพิ่มประสิทธิภาพการใช้ทรัพยากร โดยจัดสรรงบประมาณ **10%** สำหรับโครงการค่าสิ่งแวดล้อม。/\n-  

  **พัฒนาทีมงาน:** งบประมาณส่วนการพัฒนาทีมงานและห้องปฏิบัติการเพิ่มขึ้น **15%** เพื่อเสริมสร้างศักยภาพทีม  

  /\k/\k/## ทางคุณสามารถเพิ่มเติมเกี่ยวกับแผนการใช้งบประมาณในแต่ละหัวข้อ หรือต้องการรายละเอียดเฉพาะเรื่อง สามารถสอบถามได้โดยตรง  

  มีดีข่าวเหลือทุกเรื่อง! 😊",
3 "usedModel": "qwen/qwen3-8b:free"
4

```

# LLM providers



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)

API keys - OpenAI API

platform.openai.com/api-keys

ITGenius / Default project

Dashboard Docs API S

Create Chat Audio Images Assistants

Usage API keys Logs Storage Batches Optimize

## API keys

+ Create new secret key

You have permission to view and manage all API keys in this project.

Do not share your API key with others or expose it in the browser or other client-side code. To protect your account's security, OpenAI may automatically disable any API key that has leaked publicly.

View usage per API key on the [Usage page](#).

NAME	SECRET KEY	CREATED BY	PERMISSIONS
ai-chatbot-langchain	sk-...MJgA	Samit Koyom	All
n8n sample	sk-...blcA	Samit Koyom	All
n8n test	sk-...iIYA	Samit Koyom	All



# GPT-4o mini



# OpenAI

paces ▾ API Network

Search Postman

Ctrl K

Invite



New Import

POST 01/\_api/chat\_01\_start



HTTP AIChatbotLangchain / 02\_Langchain\_Basic / 01/\_api/chat\_01\_start

POST

`{baseUrl} /api/chat_01_start`

Params

Authorization

Headers (16)

Body

Scripts

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

This request does not have a body

Body

Cookies (1)

Headers (6)

Test Results

200 OK

5.95 s

1.11 KB

{ } JSON

Preview

Visualize

```
1  {
2      "content": "สวัสดีครับ งบประมาณเป็นเรามีการจัดสรรงบประมาณตามแผนกลยุทธ์ของบริษัท โดยเน้นการลงทุนในด้านการพัฒนาผลิตภัณฑ์ใหม่ การตลาด และการฝึกอบรมพนักงาน เพื่อเพิ่มประสิทธิภาพในการทำงานและขยายตลาด/ก/หากคุณต้องการรายละเอียดเพิ่มเติมเกี่ยวกับงบประมาณในแต่ละแผนกหรือโครงการใด ๆ สามารถสอบถามได้เลยครับ",
3      "usedModel": "gpt-4o-mini-2024-07-18"
4 }
```

# LLM providers



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)



Google AI Studio

aistudio.google.com/apikey

## API Keys

+ Create API key

Quickly test the Gemini API

API quickstart guide

Code

```
curl "https://generativelanguage.googleapis.com/v1beta/models/gemini-2.0-flash:generateContent" \
-H 'Content-Type: application/json' \
-H 'X-goog-api-key: GEMINI_API_KEY' \
-X POST \
-d '{
  "contents": [
    {
      "parts": [
        {
          "text": "Explain how AI works in a few words"
        }
      ]
    }
]'
```

Your API keys are listed below. You can also view and manage your project and API keys in Google Cloud.

Look up API Key for project

Project number	Project name	API key	Created	Plan

Studio

Dashboard

API keys

Usage & Billing

Changelog

Documentation

Get API key

View status

Settings

Samitkoyom@gmail.com

AI

genius.co.th

# LLM providers



Azure AI Studio



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)

portal.azure.com/#view/Microsoft\_Azure\_CostManagement/Menu/~/subscriptions

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

New Chrome available

Home > Cost Management: Samit Koyom

## Cost Management: Samit Koyom | Azure subscriptions

Billing account

Search Add Refresh Export to CSV Troubleshoot Feedback

To make it easier to view all your subscriptions together we're removing this page and listing Azure subscriptions in the new All billing subscriptions page instead. To view your Azure subscriptions, select All billing subscriptions in the left pane, and then select Usage based / Azure subscriptions tab.

Overview Change scope Access control Diagnose and solve problems Reporting + analytics Monitoring Optimization Settings Billing Invoices Payment methods Azure subscriptions Help

View Azure subscriptions billed to your account. The charges shown below are estimated amounts based on your Azure usage and do not include tax. The amount excludes Azure reservations and marketplace transactions.

Text search Invoice section : All invoice sections Billing profile : All billing profiles Status : Any status

Showing 1 to 1 of 1 subscriptions.

Name ↑↓	ID ↑↓	Plan ↑↓	Invoice section ↑↓	Billing profile ↑↓	Status ↑↓	Service tenant ID	Month-to-date charges	Last month's charges
Azure subscription 1	9afdd173-1a1e-42d0-b...	Microsoft Azure Plan	Samit Koyom	Samit Koyom	Active	a0047fc6-f8fa-4f4c-96c...	0.00	0.00 ***

< Previous Page 1 of 1 Next >

Add or remove favorites by pressing Ctrl + Shift + F



สถาบันไอทีเจเนียส

www.itgenius.co.th

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

samit90daytalk@hotmail.com DEFAULT DIRECTORY

Home > SamitApp

## SamitApp | Keys and Endpoint

Azure OpenAI

Search Regenerate Key1 Regenerate Key2

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Resource visualizer

Resource Management

Keys and Endpoint (selected)

Encryption Pricing tier Networking Identity Cost analysis Properties

Security Monitoring Automation Help

**KEY 1**

These keys are used to access your Azure AI Foundry API. Do not share your keys. Store them securely—for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

Show Keys

KEY 1

KEY 2

Location/Region

eastus

Endpoint

https://samitapp.openai.azure.com/

<https://portal.azure.com/#@samit90daytalk@hotmail.onmicrosoft.com/resource/subscriptions/9afdd173-1a1e-42d0-b157-3c57455b4b30/resourceGroups/SampleApp/providers/Microsoft.CognitiveServices/accounts/SamitApp/cskeys>





ai.azure.com/resource/models/gpt-5-mini/version/2025-08-07/registry/azure-openai?wsid=/subscriptions/9afdd173-1a1e-42d0-b157-3c57455b4b30/resourceGroups/sampleapp/providers/Microsoft.CognitiveServices/accounts/samit...

Azure AI Foundry / samit-me4hlzb5-eastus2 / Models / gpt-5-mini

gpt-5-mini

Use this model Fine-tune

Overview Model catalog Playgrounds Build and customize Agents Templates Fine-tuning Observe and optimize Monitoring Protect and govern Azure OpenAI Evaluation Guardrails + controls Risks + alerts Governance Azure OpenAI Stored completions Batch jobs Assistant vector stores Data files My assets Models + endpoints Web apps More Management center

Details Existing deployments License

gpt-5-mini powers low cost and fast experiences such as real-time agents, orchestrating tool calls in response to customer support requests.

**Key Capabilities**

- A lightweight version for cost-sensitive applications.
- Now supporting minimal reasoning, a new verbosity setting, and the "customs" tool for raw text output.
- Supports new "allowed tools" tool choice that enables you to specify multiple tools in the tool choice instead of just one
- supports new "preamble" support, allowing the model to "think" before calling a tool. This is always enabled and controlled through prompting.
- gpt-5-mini supports multimodal inputs, real-time streaming and full tool support for smarter, more dynamic user experiences

See more

**Model Versions**

Learn more about region availability

East US 2

Model ID	Deployment type	Lifecycle	Max request	Retirement Date
2025-08-07	Global Standard, Data Zone Standard	Generally available	Input: N/A Output: N/A	Fri, Aug 7, 2026

**Quick facts**

gpt-5-mini

Chat completion Direct from Azure

Training data last updated May 2024

Pricing See direct from Azure pricing

**Model ID**

Reference this model ID when deploying the model in code

azureml://registries/azure-openai/models/gpt-5-mini/versions/2025-08-07



Azure AI Foundry / samit-me4hlzb5-eastus2 / Models / gpt-5-mini

Overview Model catalog Playgrounds Build and customize Agents Templates Fine-tuning Observe and optimize Monitoring Protect and govern Azure OpenAI Evaluation Guardrails + controls Risks + alerts PREVIEW Governance PREVIEW Azure OpenAI Stored completions Batch jobs Assistant vector stores Data files My assets Models + endpoints Web apps ... More Management center

gpt-5-mini

Use this model Fine-tune

Details Existing deployments License

Key Capabilities

- A lightweight version for cost-sensitive applications.
- Now supporting minimal reasoning, a new verbosity setting.
- Supports new "allowed tools" tool choice that enables you to...
- supports new "preamble" support, allowing the model to "th...
- gpt-5-mini supports multimodal inputs, real-time streaming

See more

Model Versions

Learn more about region availability

East US 2

Model ID	Deployment type
2025-08-07	Global Standard, Data Zone Standard

Data, media and languages

Property	Description
Supported data types	Inputs text, image
Outputs	text
Supported languages	en

Deploy gpt-5-mini

Deployment name \* gpt-5-mini-2

Deployment type Global Standard

Global Standard: Pay per API call with the highest rate limits. Learn more about Global deployment types. Data might be processed globally, outside of the resource's Azure geography, but data storage remains in the AI resource's Azure geography. Learn more about data residency.

Deployment details

Model version 2025-08-07

AI resource samit-me4hlzb5-eastus2

Capacity 100K tokens per minute (TPM)

Resource location East US 2

Content safety DefaultV2

Version upgrade policy Once a new default version is available

Customize Deploy Cancel

Quick facts

gpt-5-mini Chat completion Direct from Azure

Training data last updated May 2024

Pricing See direct from Azure pricing

Model ID Reference this model ID when deploying the model in code

azureml://registries/azure-openai/models/gpt-5-mini/versions/2025-08-07



Azure AI Foundry / samit-me4hlzb5-eastus2 / Deployments / gpt-5-mini-2

Overview Model catalog Playgrounds Build and customize Agents Templates Fine-tuning Observe and optimize Monitoring Protect and govern Azure OpenAI Evaluation Guardrails + controls Risks + alerts Governance Azure OpenAI Stored completions Batch jobs Assistant vector stores Data files My assets Models + endpoints Web apps ... More Management center

## ← gpt-5-mini-2

Details Metrics

[Open in playground](#) Request quota Edit Delete

**Endpoint**

Target URI  
https://samit-me4hlzb5-eastus2.cognitiveservices.azure.com/openai/responses...

**Deployment info**

Name	gpt-5-mini-2	Provisioning state	Succeeded
Deployment type	Global Standard	Created on	2025-09-09T09:35:38.393495Z
Created by	samit90daytalk@hotmail.com	Modified on	Sep 9, 2025 4:35 PM
Modified by	samit90daytalk@hotmail.com	Version upgrade policy	Once a new default version is available
Rate limit (Tokens per minute)	100,000	Rate limit (Requests per minute)	100
Model name	gpt-5-mini	Model version	2025-08-07
Life cycle status	GenerallyAvailable	Date created	Aug 7, 2025 7:00 AM
Date updated	Aug 7, 2025 7:00 AM	Model retirement date	Aug 8, 2026 7:00 AM

**Monitoring & safety**

Content filter DefaultV2

**Language** Javascript **SDK** OpenAI SDK **Authentication type** Key Authentication

## Get Started

Below are example code snippets for a few use cases. For additional information about Azure OpenAI SDK, see full [documentation](#) and [samples](#).

### 1. Authentication using API Key

For OpenAI API Endpoints, deploy the Model to generate the endpoint URL and an API key to authenticate against the service. In this sample endpoint and key are strings holding the endpoint URL and the API Key.

The API endpoint URL and API key can be found on the Deployments + Endpoint page once the model is deployed.

To create a client with the OpenAI SDK using an API key, initialize the client by passing your API key to the SDK's configuration. This allows you to authenticate and interact with OpenAI's services seamlessly.

```
const api_key = "<your-api-key>";
const endpoint = "https://samit-me4hlzb5-eastus2.cognitiveservices.azure.com/openai/v1";
const modelName = "gpt-5-mini";
const deployment_name = "gpt-5-mini-2";

const client = new OpenAI({
  baseURL: endpoint,
  apiKey: api_key
});
```

### 2. Install dependencies

- Install Node.js
- Copy the following lines of text and save them as a file package.json inside your folder.

```
{
  "type": "module"
```



Azure AI Foundry / samit-me4hlzb5-eastus2 / Deployments / gpt-5-mini-2

Docs All resources ⚙️ 😊 samit-me4hlzb5-eastus2 (eastus2, S0) SK

## ← gpt-5-mini-2

Overview Model catalog Playgrounds Build and customize Agents Templates Fine-tuning Observe and optimize Monitoring Protect and govern Azure OpenAI Evaluation Guardrails + controls Risks + alerts Governance Azure OpenAI Stored completions Batch jobs Assistant vector stores Data files My assets Models + endpoints Web apps More Management center

Details Metrics

Open in playground Request quota Edit Delete

### Endpoint

Target URI  
https://samit-me4hlzb5-eastus2.cognitiveservices.azure.com/openai/responses...

Key  
.....

### Deployment info

Name gpt-5-mini-2	Provisioning state Succeeded
Deployment type Global Standard	Created on 2025-09-09T09:35:38.393495Z
Created by samit90daytalk@hotmail.com	Modified on Sep 9, 2025 4:35 PM
Modified by samit90daytalk@hotmail.com	Version upgrade policy Once a new default version is available
Rate limit (Tokens per minute) 100,000	Rate limit (Requests per minute) 100
Model name gpt-5-mini	Model version 2025-08-07
Life cycle status GenerallyAvailable	Date created Aug 7, 2025 7:00 AM
Date updated Aug 7, 2025 7:00 AM	Model retirement date Aug 8, 2026 7:00 AM

### Language

REST curl Key Authentication

## Get Started

### 1. Authentication using API Key

For Serverless API Endpoints, deploy the Model to generate the endpoint URL and an API key to authenticate against the service. In this sample endpoint and key are strings holding the endpoint URL and the API Key. The API endpoint URL and API key can be found on the Deployments + Endpoint page once the model is deployed.

If you're using bash:

```
export AZURE_API_KEY=<your-api-key>
```

If you're in powershell:

```
$Env:AZURE_API_KEY = "<your-api-key>"
```

If you're using Windows command prompt:

```
set AZURE_API_KEY = <your-api-key>
```

### 2. Run a basic code sample

Paste the following into a shell

```
curl -X POST "https://samit-me4hlzb5-eastus2.cognitiveservices.azure.com/openai/responses?api-version=2025-04-01-preview" \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $AZURE_API_KEY" \
-d '{
  "messages": [
    {
      "role": "user",
      "content": "Hello, how are you?"
    }
  ]
}'
```

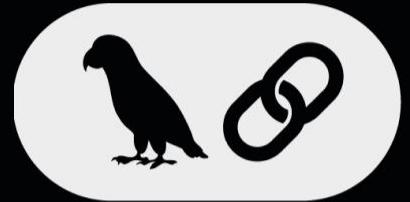


```
pwsh in Samit x + v
Samit $env:AZURE_API_KEY = "5KcA$GXATX"
Samit $uri = "https://samit-me4hlzb5-eastus2.cognitiveservices.azure.com/openai/responses?api-version=2025-04-01-preview"
Samit $headers = @{
    "Content-Type" = "application/json"
    "Authorization" = "Bearer $env:AZURE_API_KEY"
}
Samit $body = @{
    "input": "I am going to Thailand, what should I see?",
    "model": "gpt-5-mini-2"
}
Samit Invoke-RestMethod -Uri $uri -Method 'POST' -Headers $headers -Body $body

id : resp_68bfffba565a08190813c449c9d3d98f105209028245e369f
object : response
created_at : 1757412261
status : completed
background : False
content_filters :
error :
incomplete_details :
instructions :
max_output_tokens :
max_tool_calls :
model : gpt-5-mini-2
output : {@{id=rs_68bfffba5b9388190909f928e8620efa605209028245e369f; type=reasoning; summary=System.Object[]},@{id=msg_68bfffba1220819093fcdc6d65d801f405209028245e369f; type=message; status=completed; content=System.Object[]; role=assistant}}
parallel_tool_calls : True
previous_response_id :
prompt_cache_key :
reasoning : @{effort=medium; summary=}
safety_identifier :
service_tier : default
store : True
temperature : 1.0
text : @{format=}
tool_choice : auto
tools : {}
top_p : 1.0
```



# สร้าง AI Chatbots สำหรับองค์กร



LangChain  
ร่วมกับ Next.JS  
และ ➡ supabase



เมวิด์ไอบันทึกการอบรม  
ย้อนหลังให้ทุกวัน



สถาบันไอทีจีเนียส

4 วัน  
12 ชั่วโมงเต็ม



**Samit Koyom**  
สถาบันไอทีจีเนียส



# โปรแกรม (Tool and Editor) ที่ใช้บرم

แนะนำ หลักสูตรนี้ใช้ Node.js เวอร์ชัน 20 ขึ้นไป

1. Node.js 22.x
2. Visual Studio Code
3. Git
4. Ollama (Optional) - ไม่บังคับ

แนะนำ Ollama เป็นเครื่องมือรัน AI model แบบ Local เมน้ำสำหรับเครื่องที่ VGA แยก และคอมพิวเตอร์ควรมี Spec สูงพอควร ไม่จำเป็น และไม่บังคับให้ต้องติดตั้งหากเครื่องไม่พร้อม





# 1. ติดตั้ง Node JS



# Download Node.JS V.22.x

<https://nodejs.org/en/>

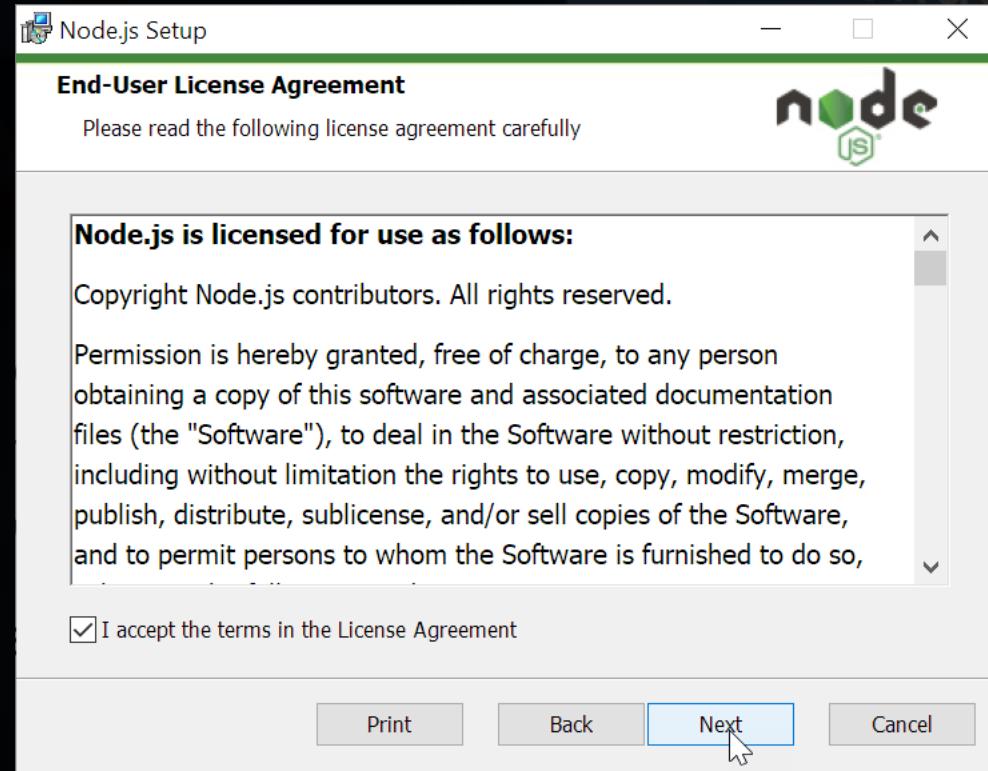
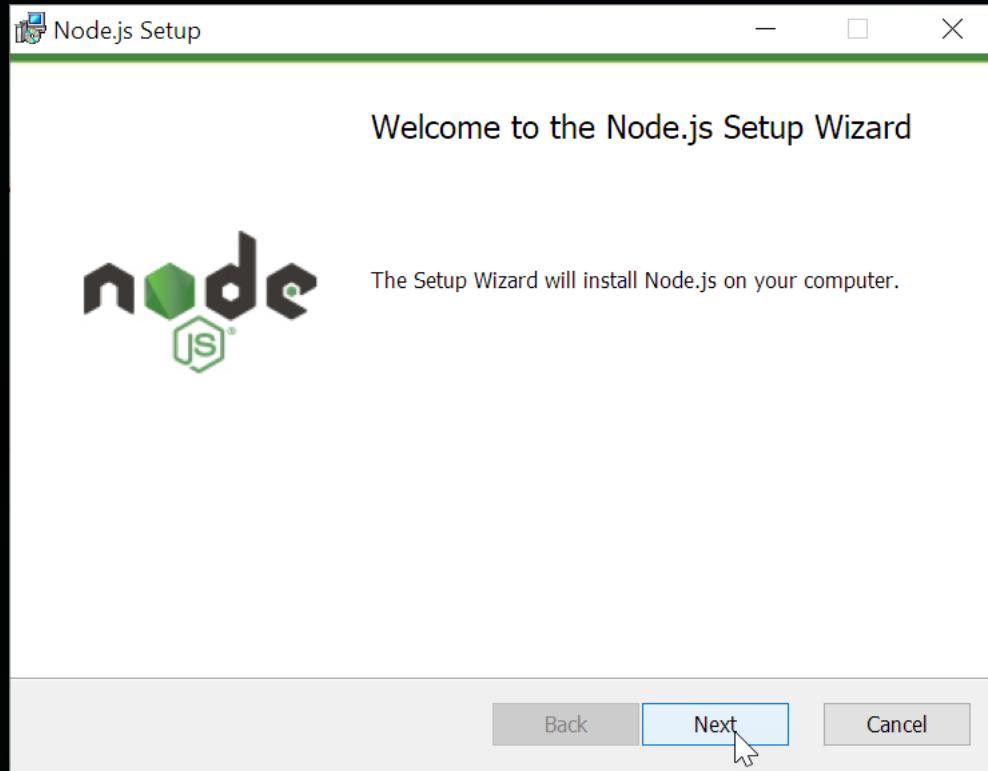
The screenshot shows the Node.js download page for version v22.15.0 (LTS). The page features a dark theme with orange highlights. At the top, there's a banner stating "New security releases to be made available Wednesday, May 14, 2025". The main heading is "Download Node.js®". Below it, there are dropdown menus for selecting the version ("v22.15.0 (LTS)"), platform ("Windows"), tool ("fnm"), and package manager ("npm"). A code block shows how to install Node.js using fpm and npm. A "Copy to clipboard" button is available for the PowerShell command. A note about fpm being a cross-platform version manager is present. Below this, there's a section for prebuilt Node.js packages, with dropdowns for Windows architecture ("x64") and a "Windows Installer (.msi)" button highlighted with a cursor. Other options include "Standalone Binary (.zip)". At the bottom, links to the changelog and blog post are provided, along with the URL <https://nodejs.org/dist/v22.15.0/node-v22.15.0-x64.msi>.



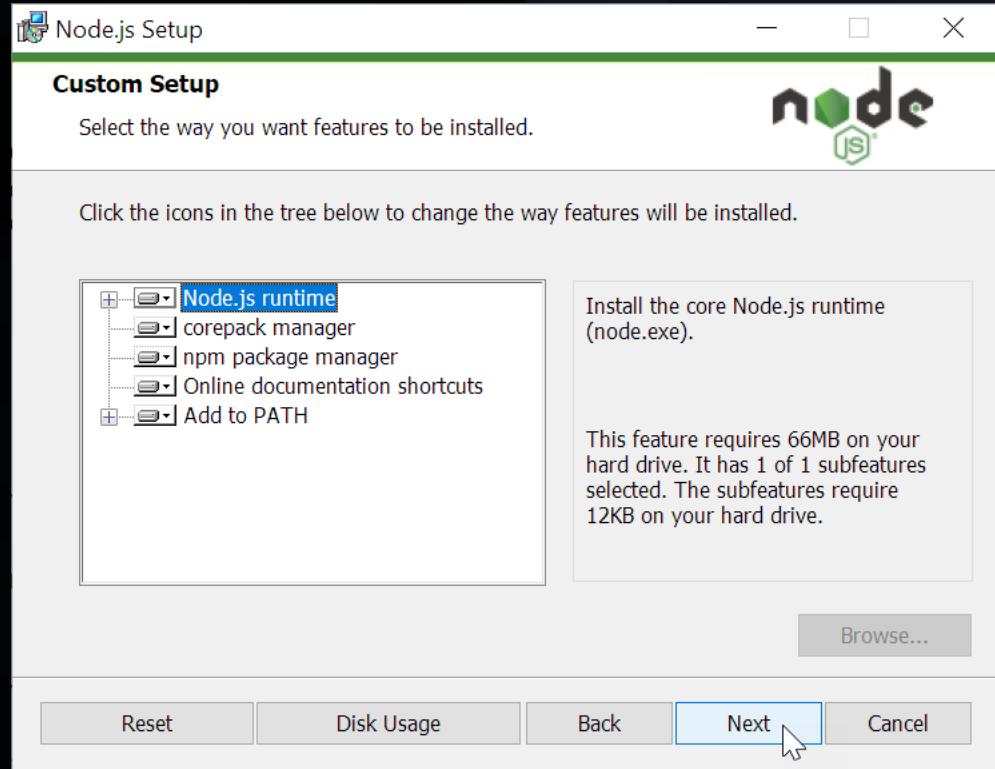
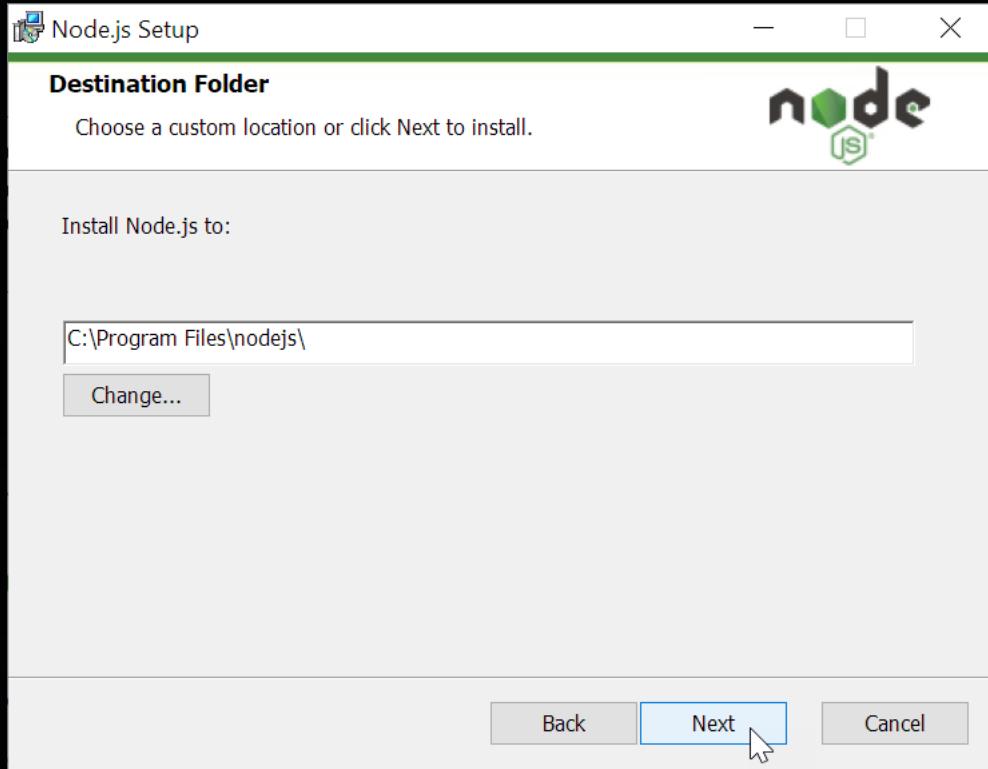
หมายเหตุ การอัปเดตของ Node.JS 20 ขึ้นไป สามารถใช้ Node.JS 21 , 22, 23 หรือ 24 ก็ได้  
สถาบันไอทีจีนีส

[www.itgenius.co.th](http://www.itgenius.co.th)

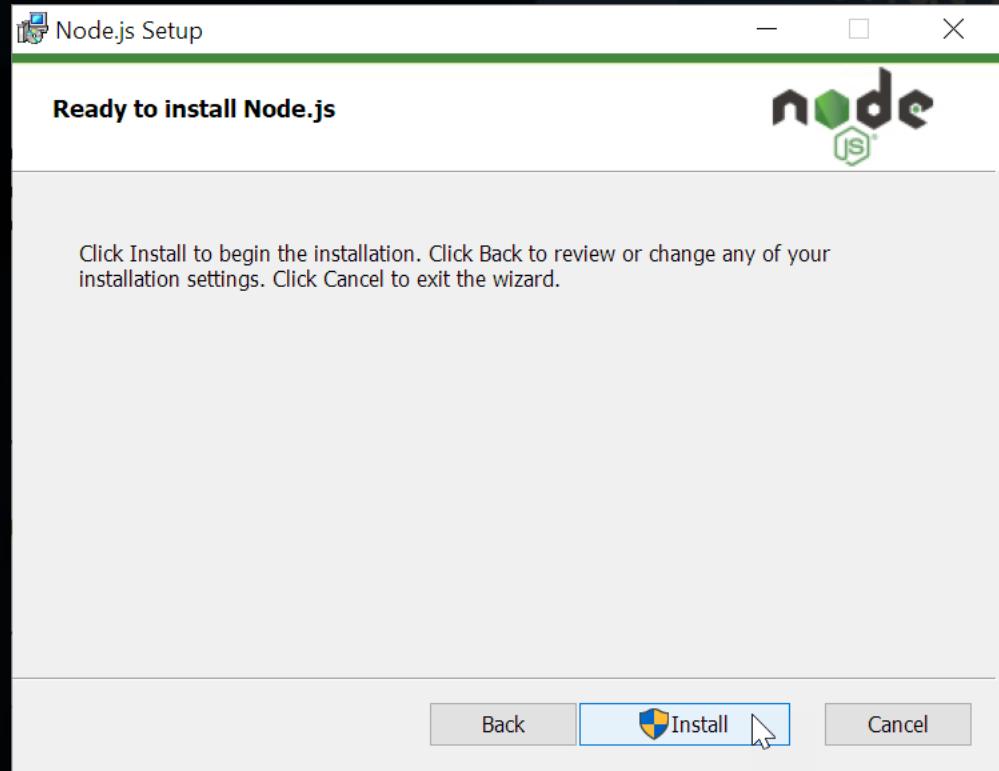
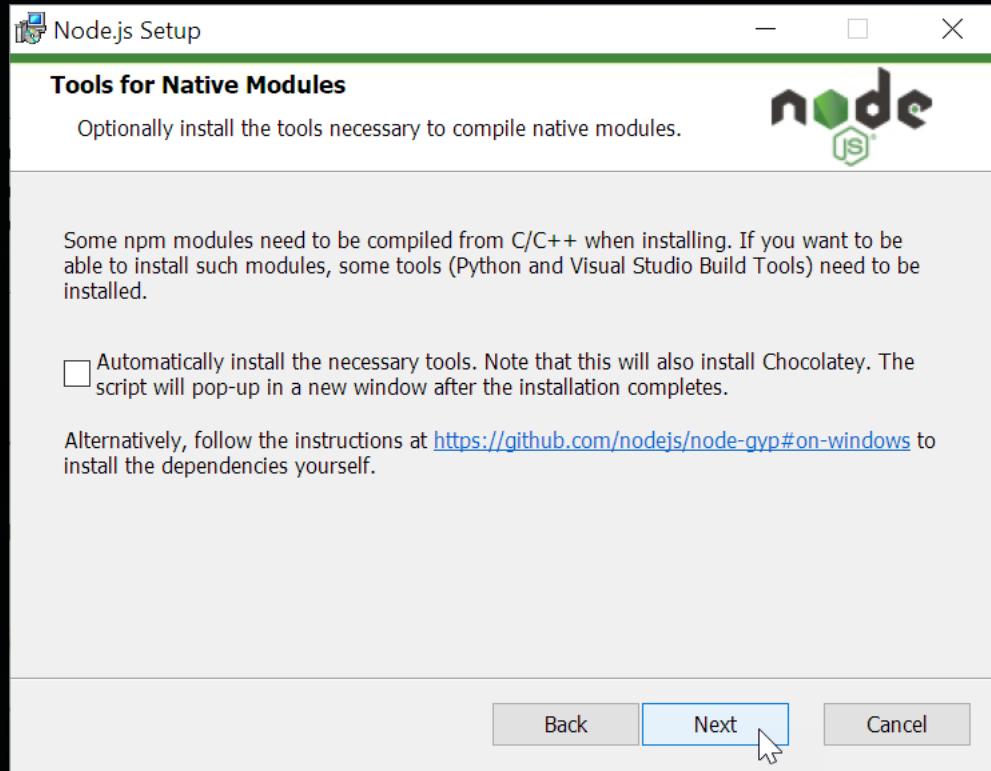
# ติดตั้ง Node.JS V.22.x



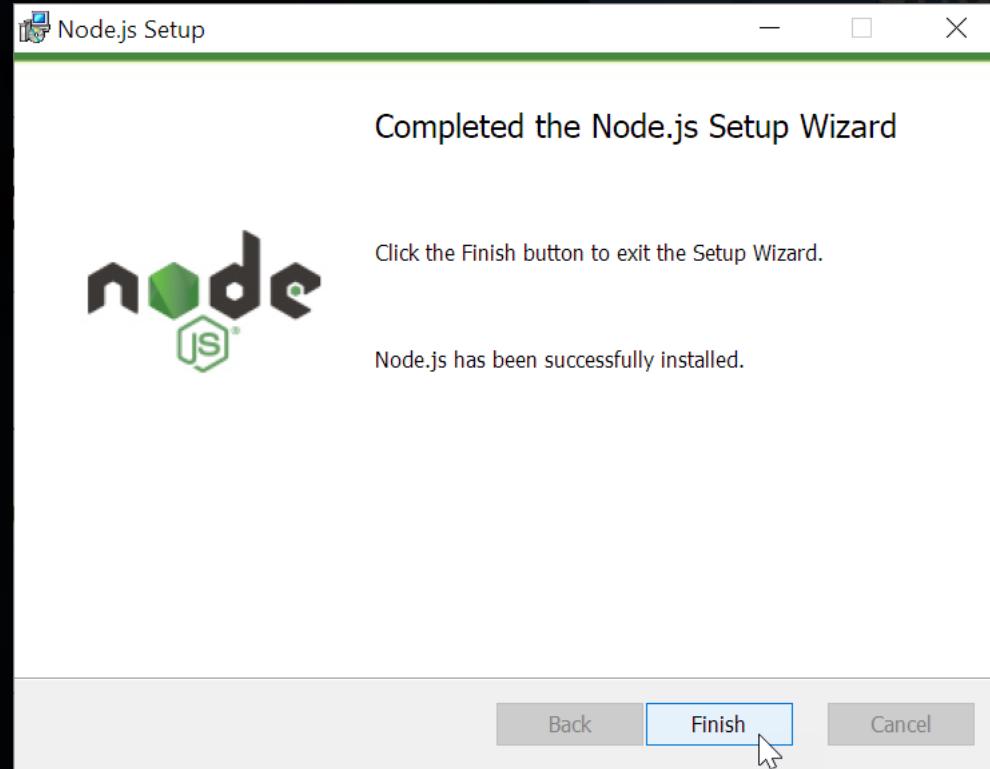
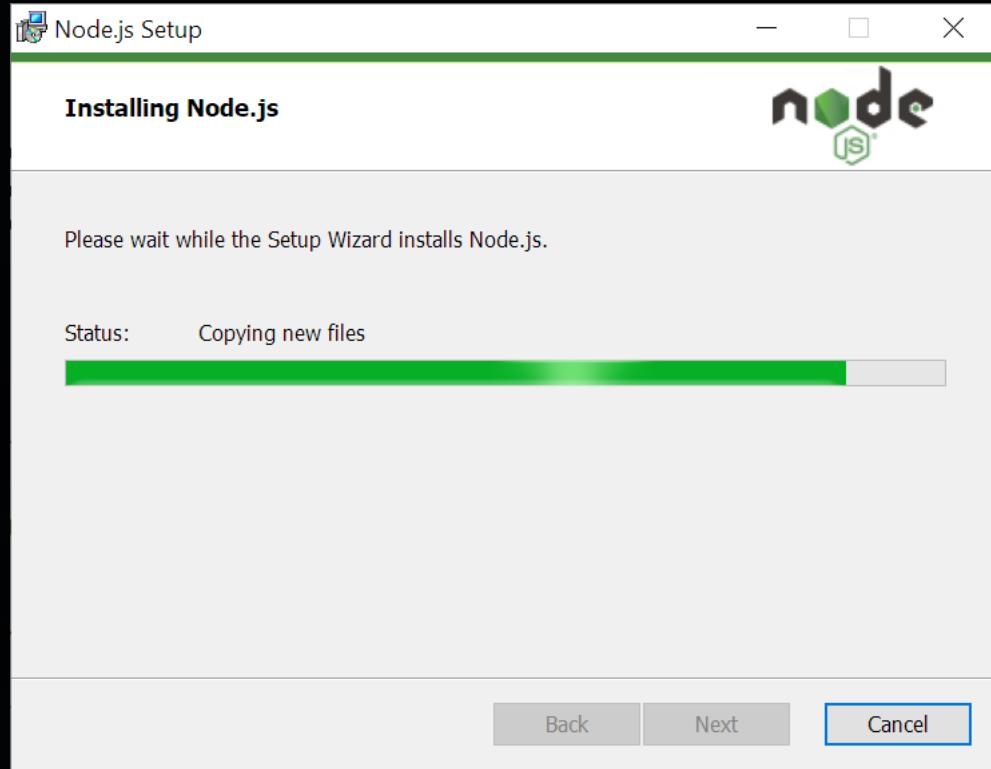
# ติดตั้ง Node.js V.22.x



# ติดตั้ง Node.js V.22.x



# ติดตั้ง Node.JS V.22.x



# ทดสอบหลังติดตั้งเสร็จ

```
node -v
```

```
C:\Users\Samit>node -v  
v22.14.0
```

```
C:\Users\Samit>
```

```
npx -v
```

```
C:\Users\Samit>npx -v  
10.9.2
```

```
C:\Users\Samit>
```

```
npm -v
```

```
C:\Users\Samit>npm -v  
10.9.2
```

```
C:\Users\Samit>
```

**หมายเหตุ** การอัปเดต Node.js 20 ขึ้นไป สามารถใช้ Node.js 21, 22, 23 หรือ 24 ก็ได้

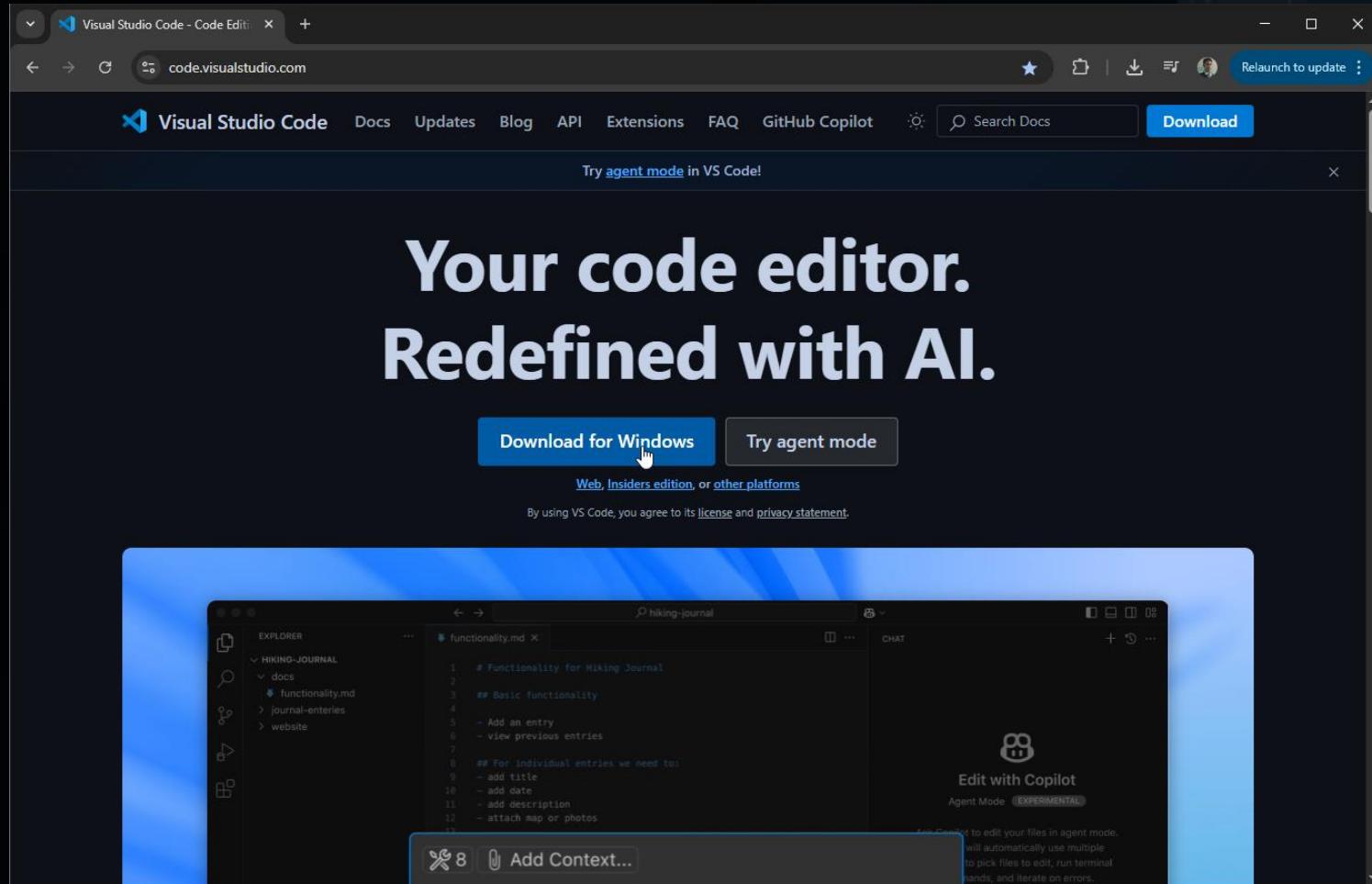




## 2. ติดตั้ง Visual Studio Code



# ຕັດຕັ້ງ Visual Studio Code ວຽກຄ້າສ່ວນເສຣິມທີ່ຈຳເປັນ



ເຂົ້າໄປລາວນີ້ໂລດ Visual Studio Code ໄດ້ທີ່ <https://code.visualstudio.com>

www.vtgenius.co.th

# การติดตั้งส่วนเสริม (Extension) ของ Visual Studio Code



# รายชื่อ Extensions ที่แนะนำสำหรับ VS Code

- 1. ES7+React/Redux/React-Native snippets** by dsznajder
- 2. Auto Import – ES6, TS, JSX, TSX** by Sergey Korenuk
- 3. Color Picker** by anseki
- 4. Material Icon Theme** by Philipp Kief
- 5. Tailwind CSS IntelliSense** by Tailwind Labs
- 6. Prettier – Code formatter** by Prettier
- 7. One Dark Pro** by binaryify

NEXT  
.JS





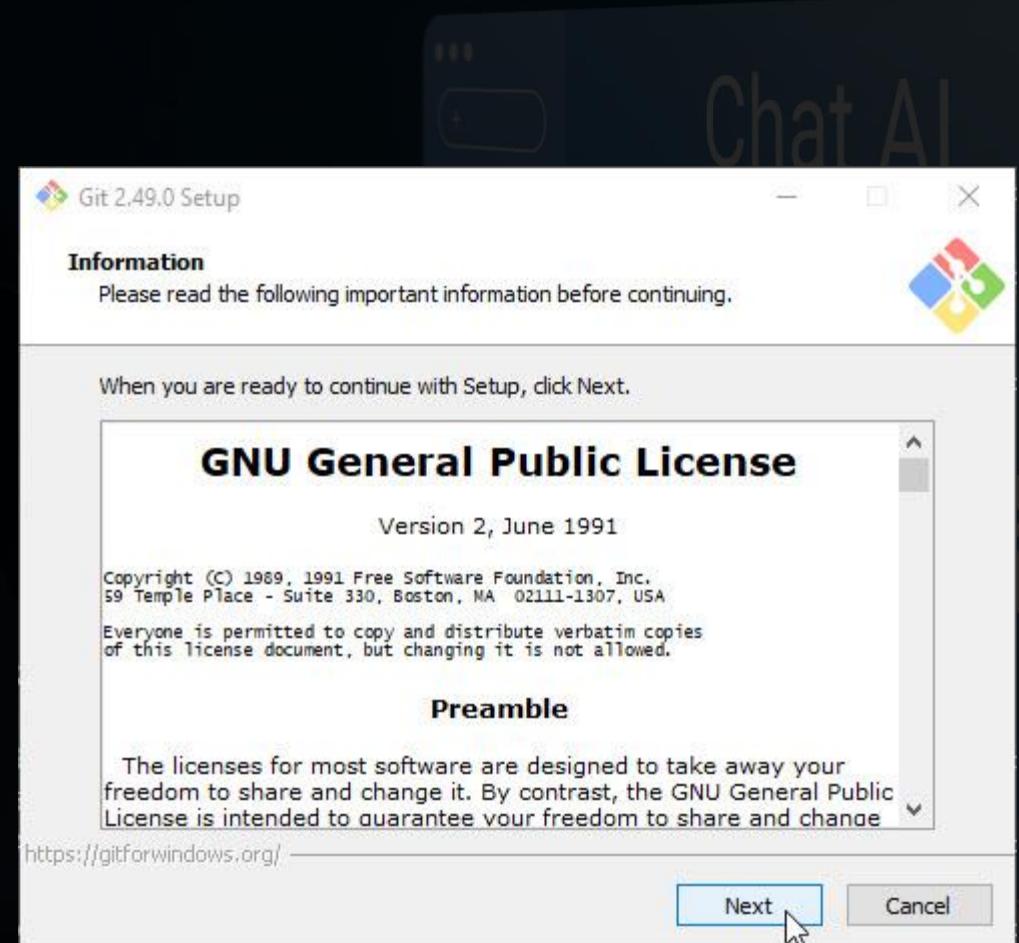
### 3. ติดตั้ง Git

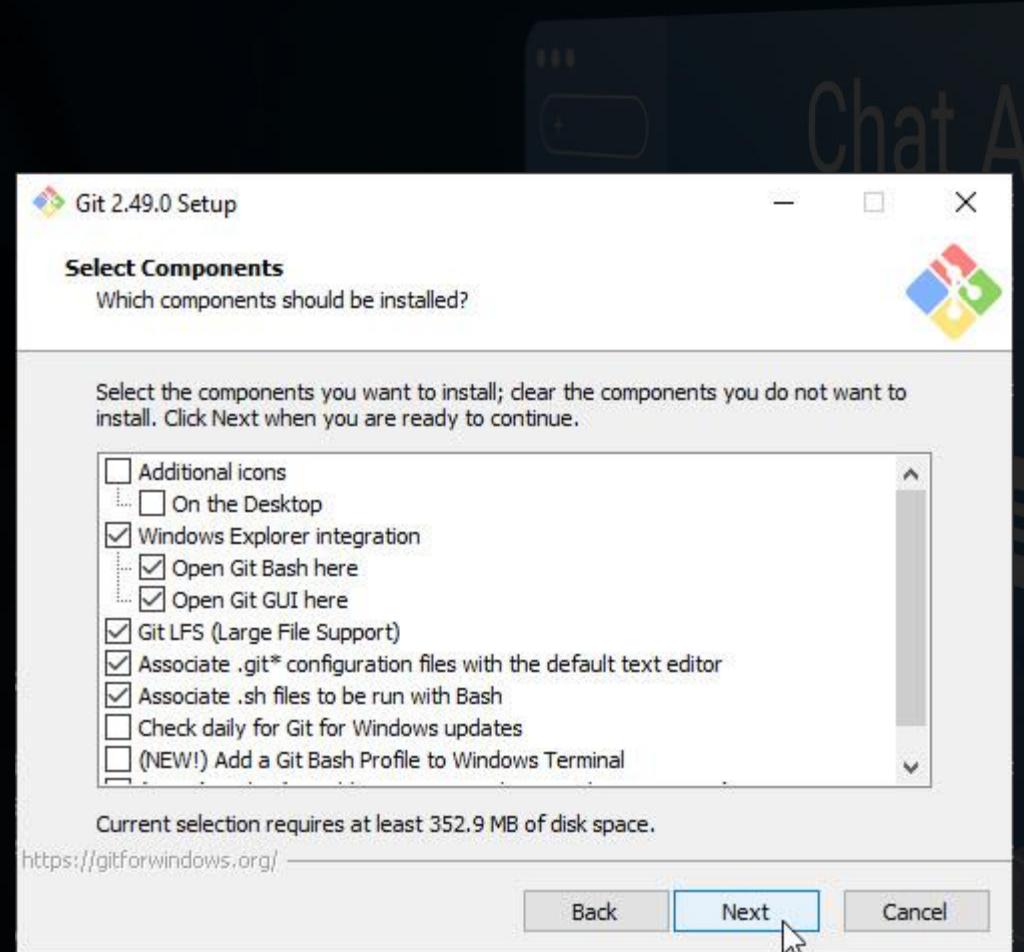
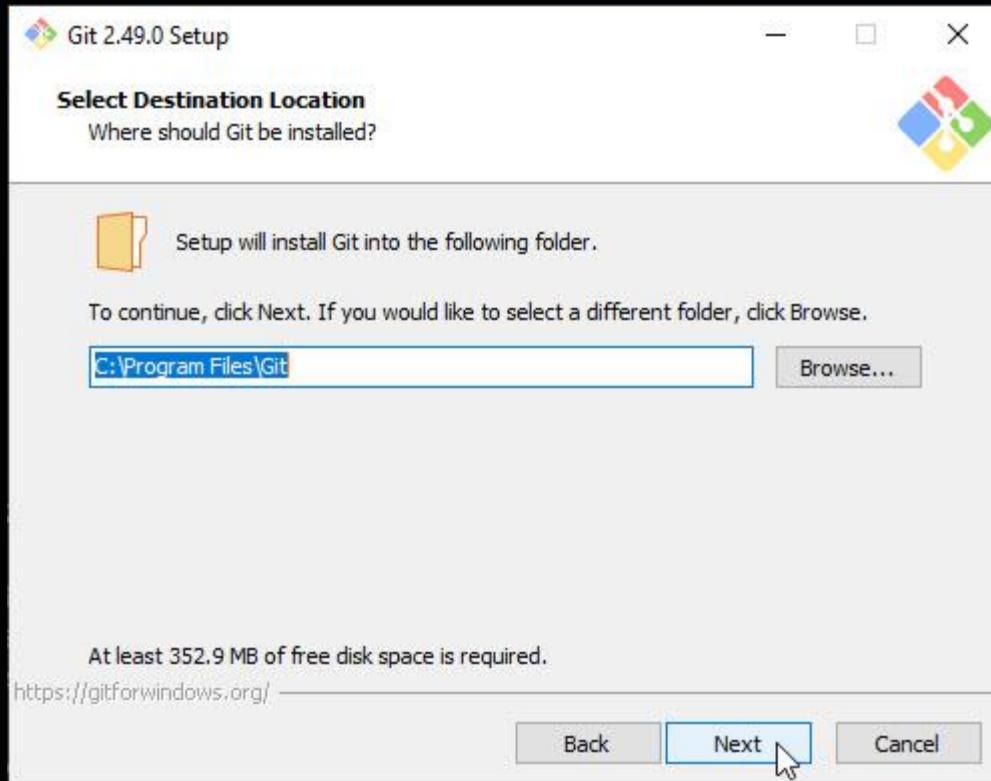


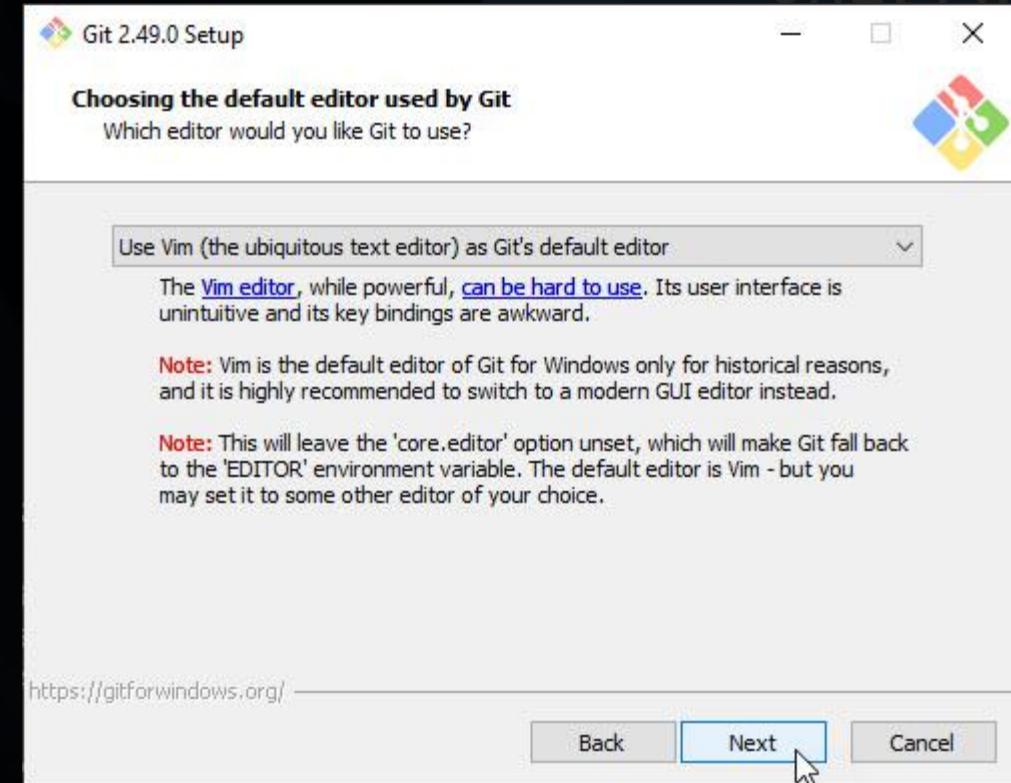
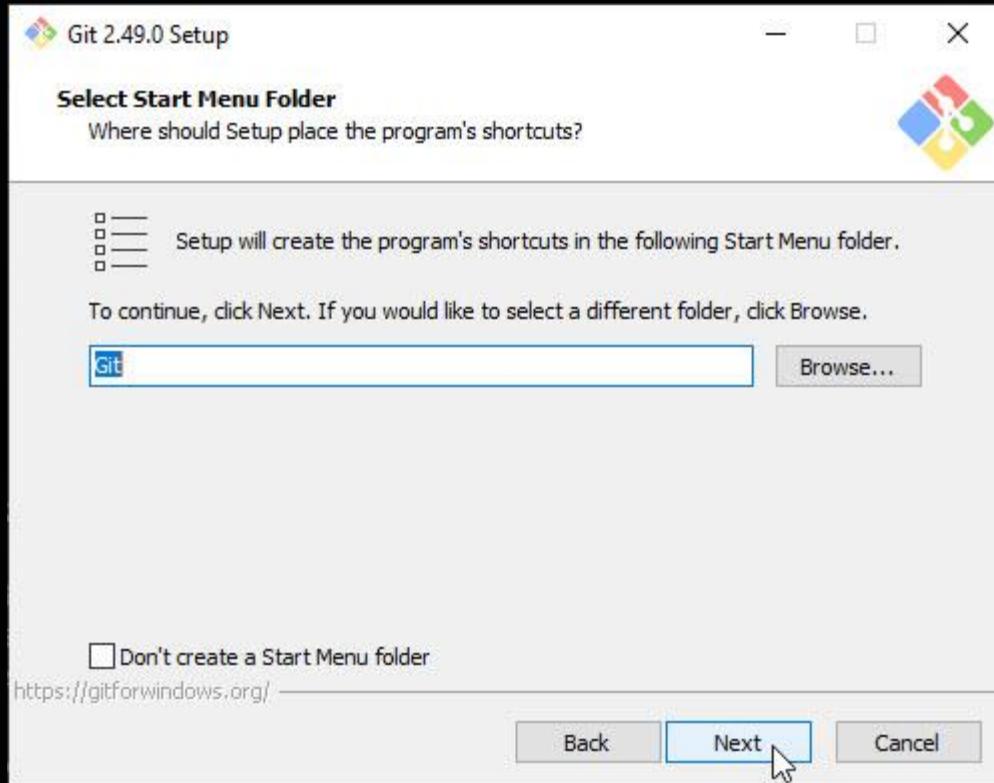
# ดาวน์โหลดไฟล์ติดตั้ง Git ได้ที่ <https://git-scm.com/>

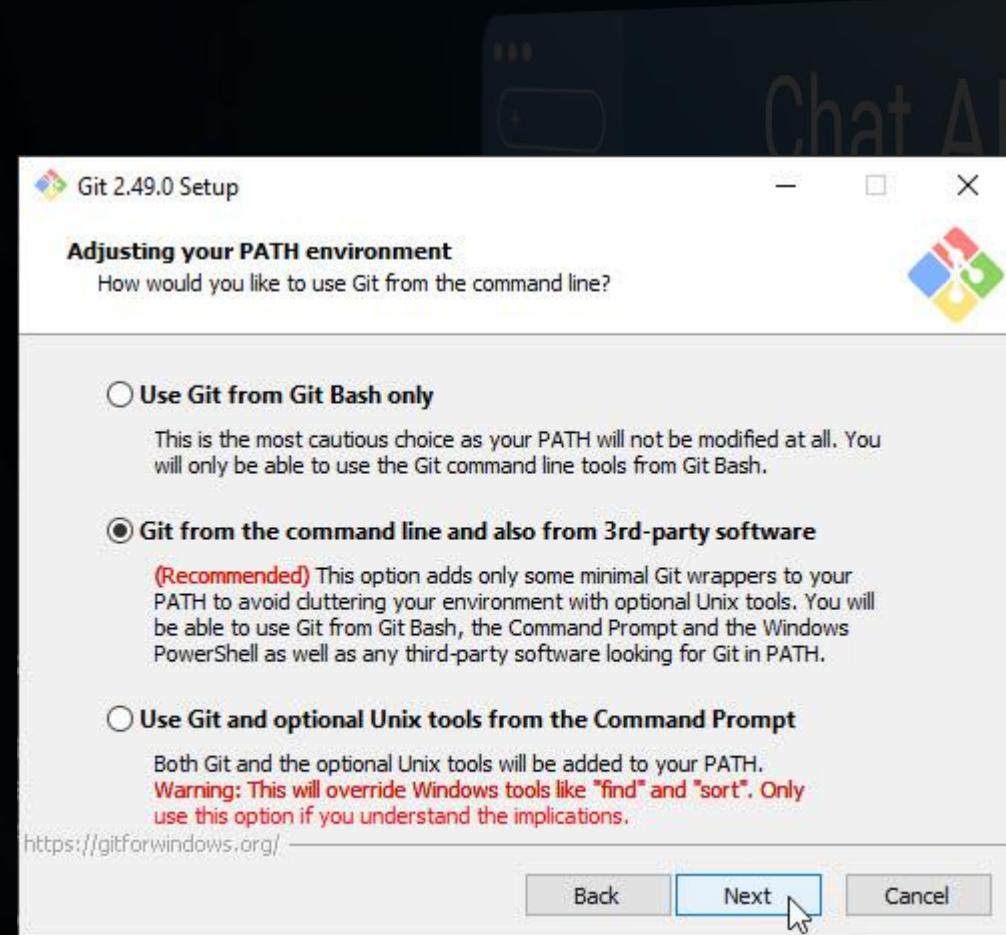
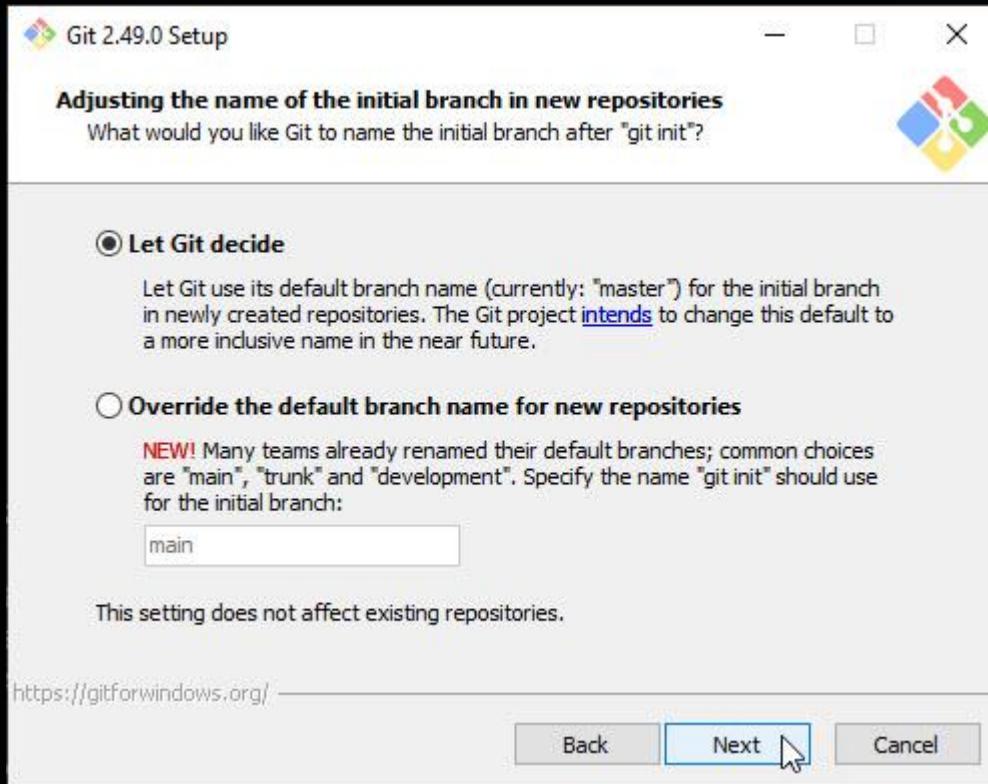
The screenshot shows the official website for Git (<https://git-scm.com/>). The page features a dark background with orange text highlights. At the top, there's a brief introduction to Git as a free and open-source distributed version control system. Below this, a diagram illustrates the concept of distributed branching between multiple repositories. The main navigation menu includes links for "About", "Documentation", "Downloads", and "Community". On the right side, there's a prominent section for the latest source release (2.49.0), with a "Download for Windows" button. Below this, there are links for "Windows GUIs", "Tarballs", "Mac Build", and "Source Code". At the bottom, there's a section for "Products providing Git hosting".

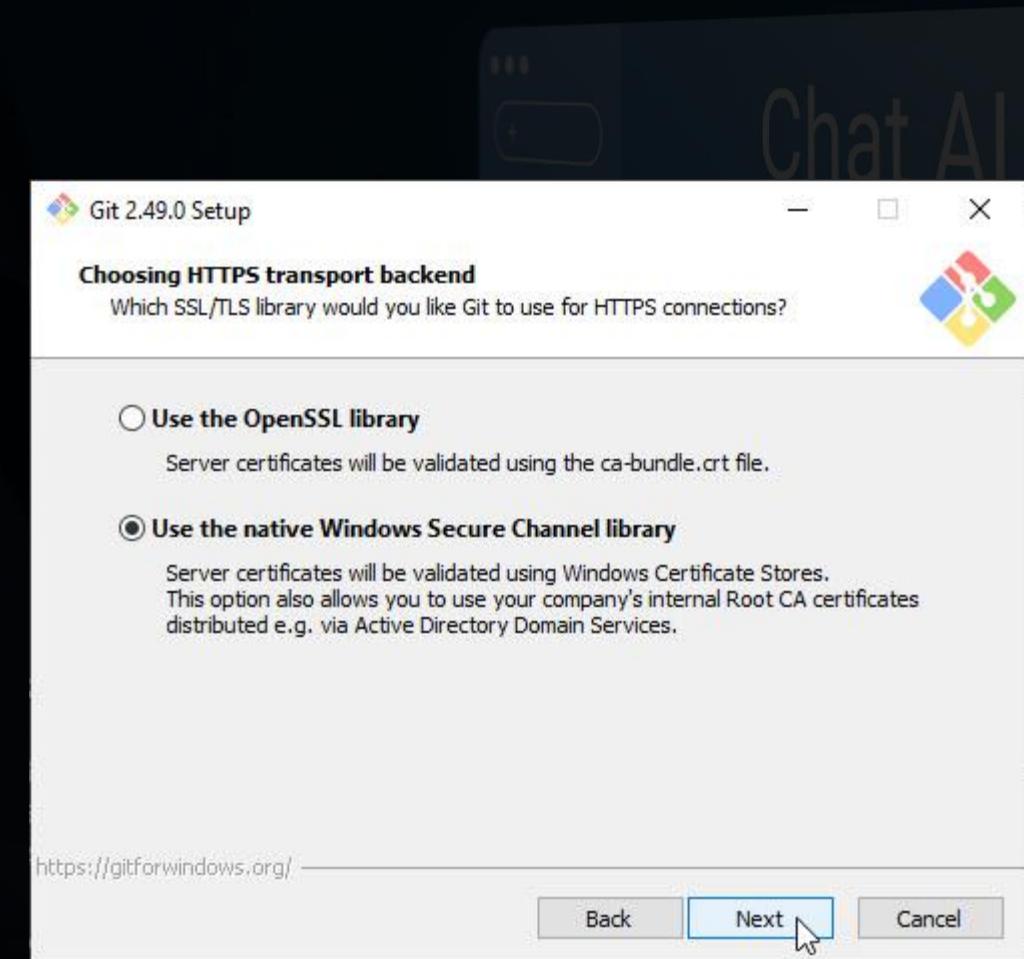
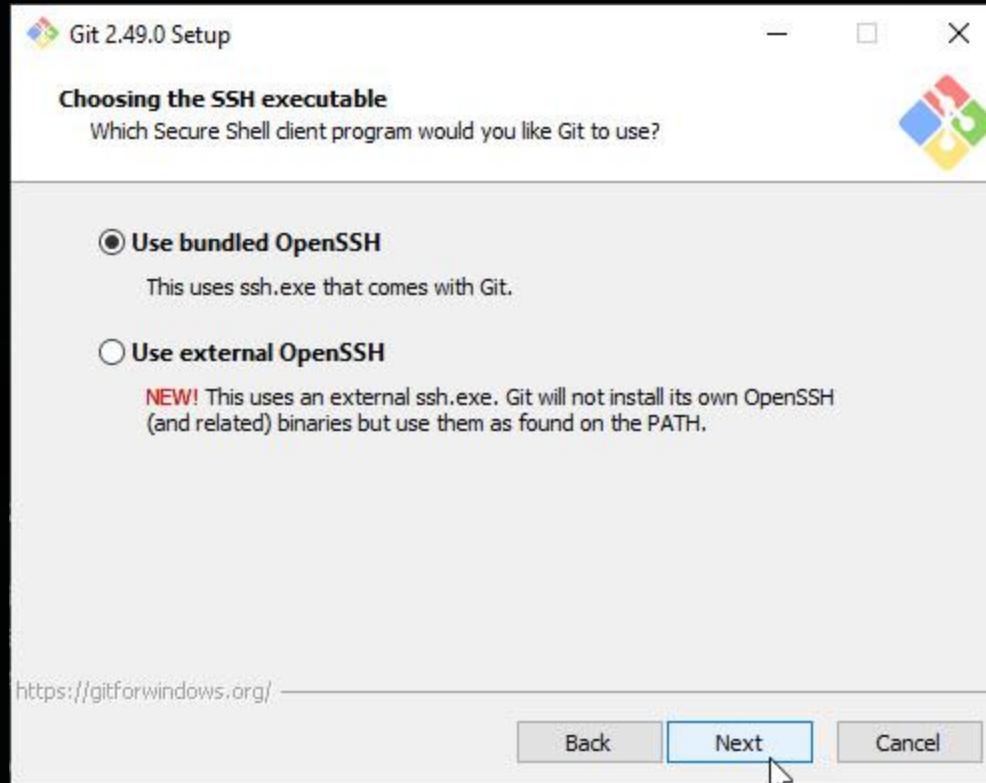


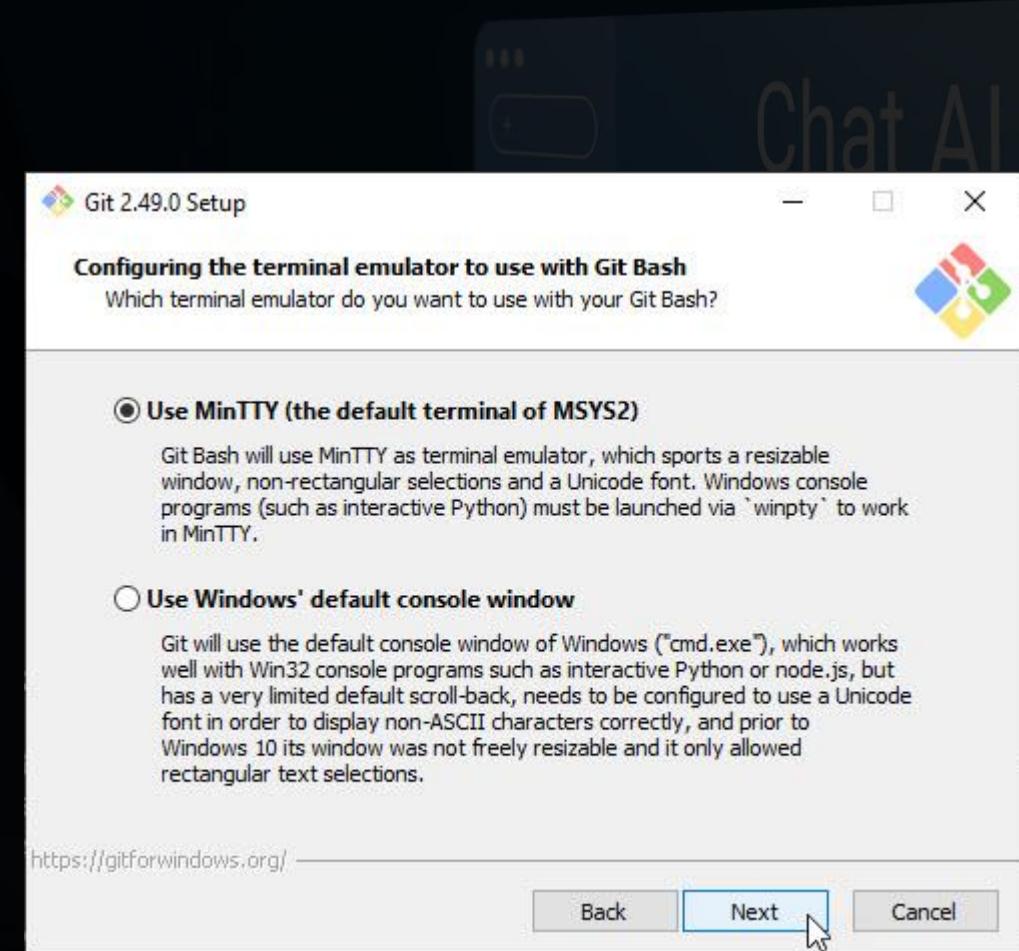
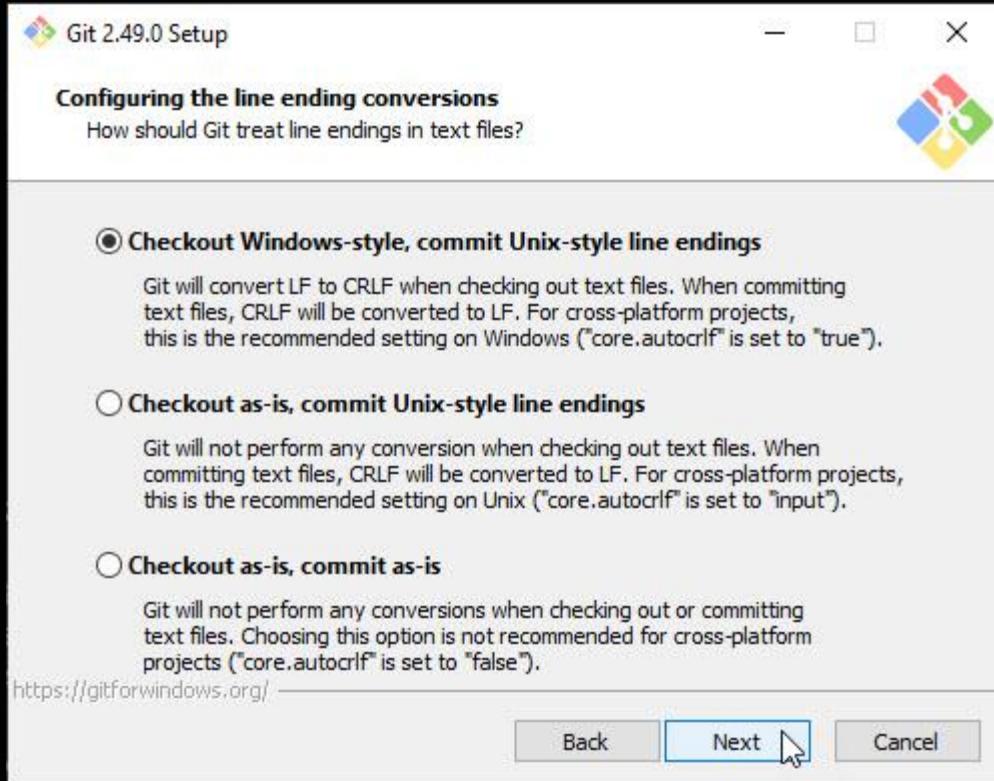


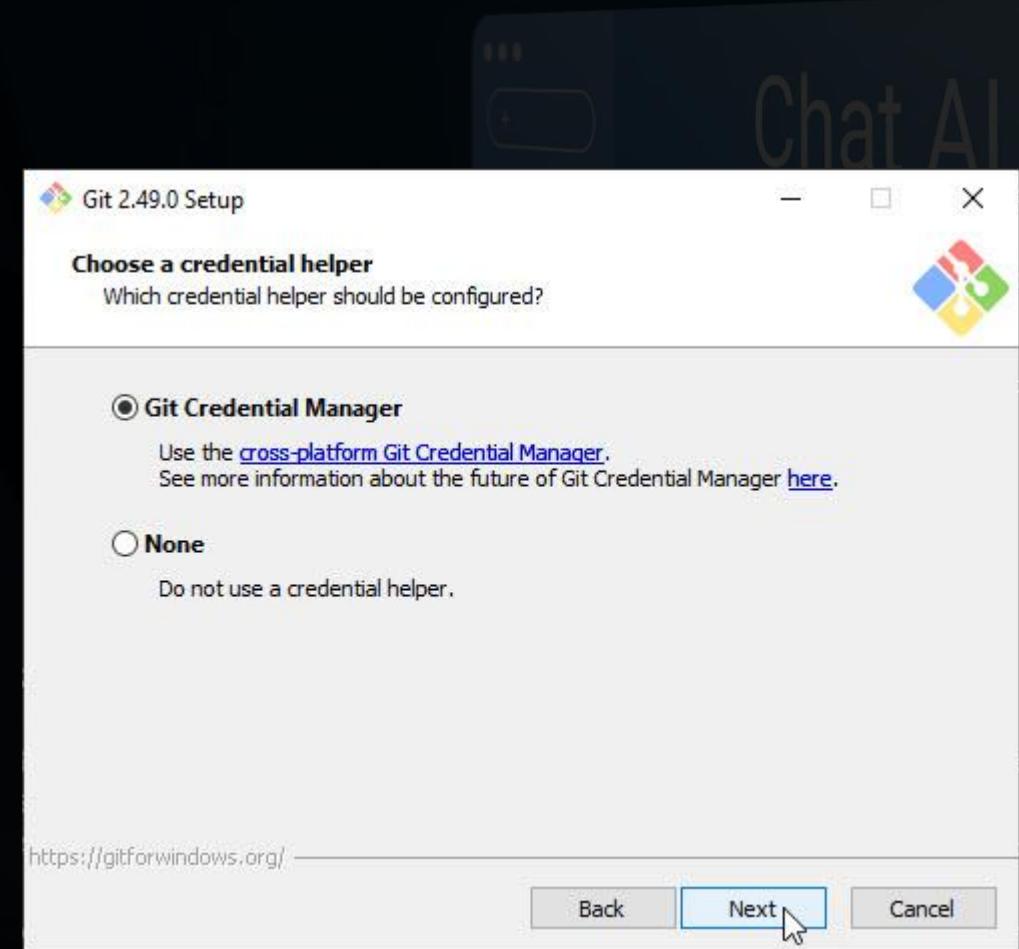
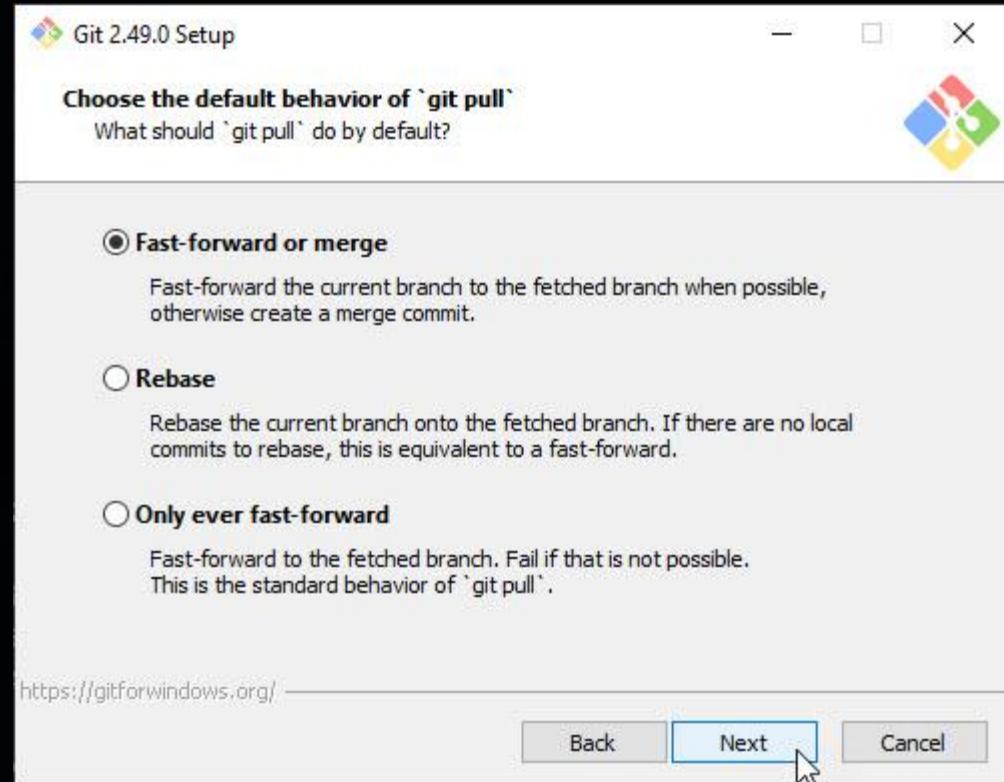


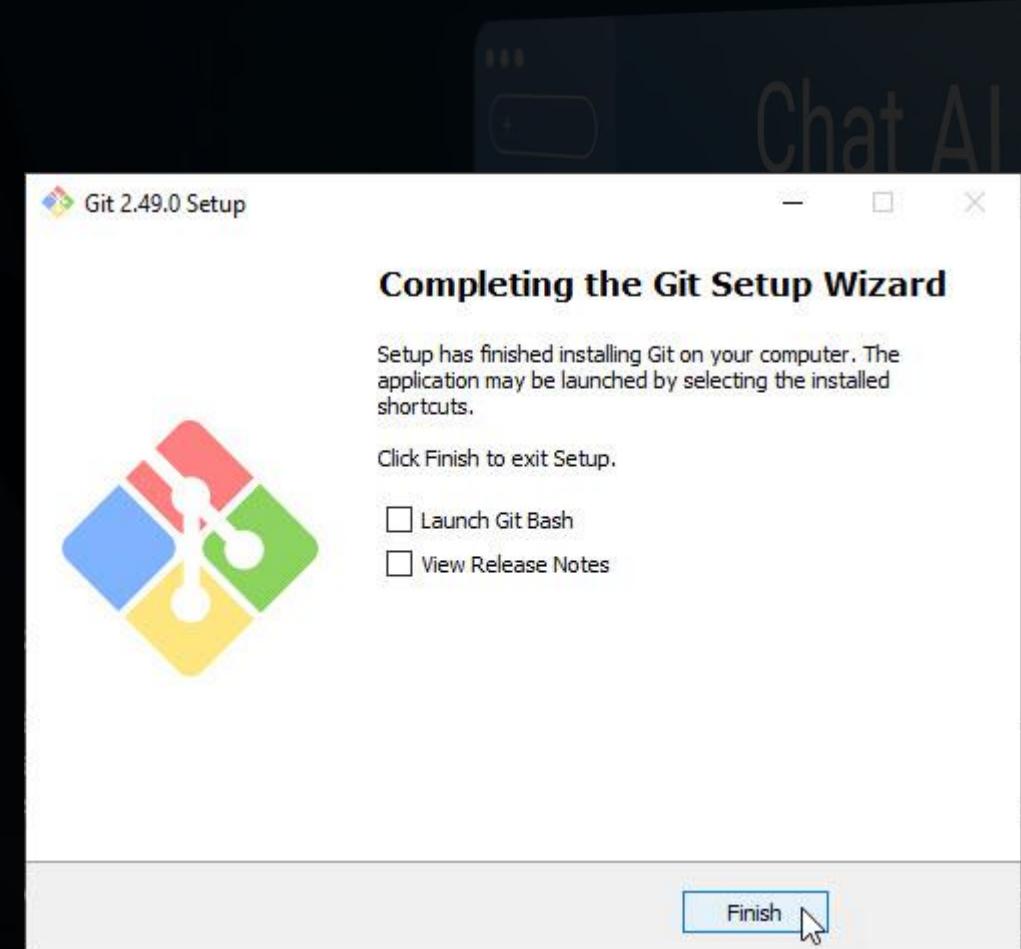
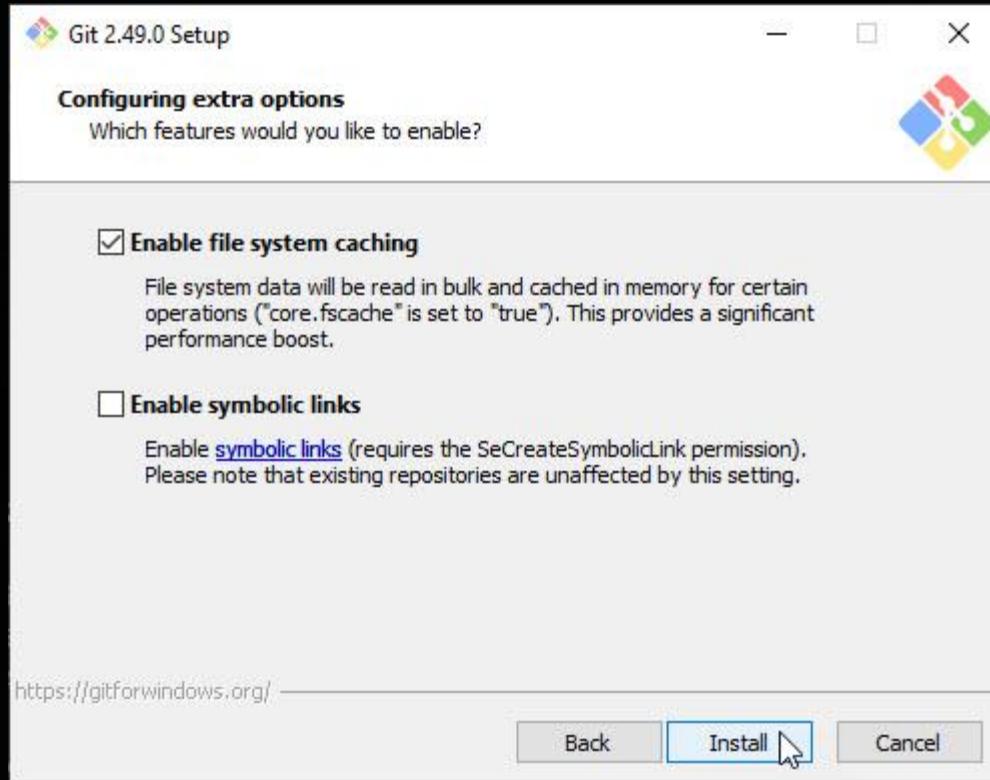








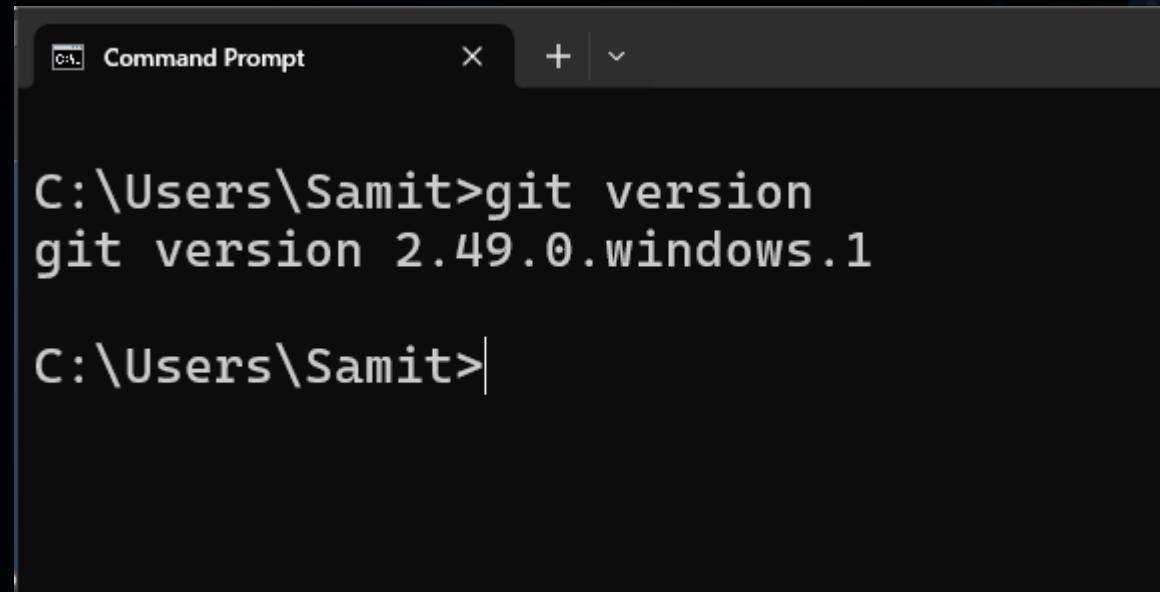




หลังติดตั้งสำเร็จกดสอบถามด้วยคำสั่ง

git version

หากพบเวอร์ชันดังภาพ ถือว่าติดตั้งเรียบร้อยพร้อมใช้งาน



```
Command Prompt
C:\Users\Samit>git version
git version 2.49.0.windows.1
C:\Users\Samit>
```





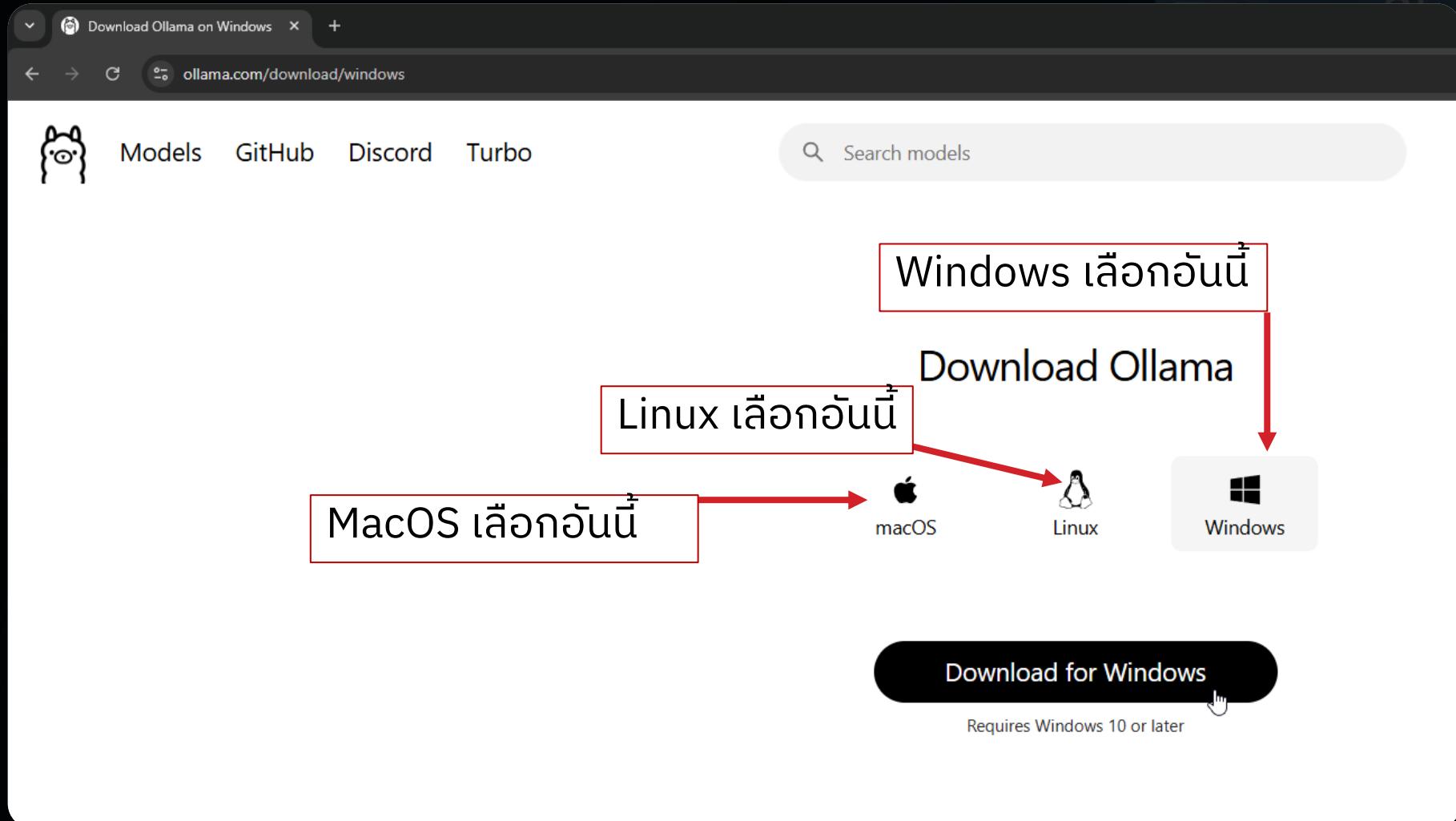
## 4. Ollama



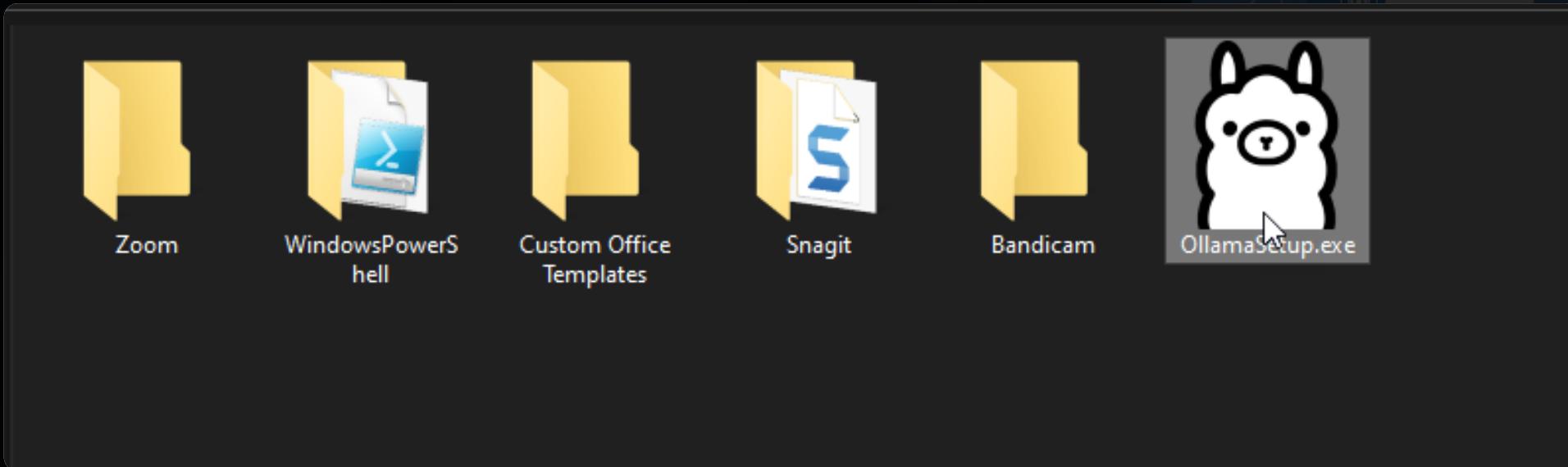
**แนะนำ** Ollama เป็นเครื่องมือรัน AI model แบบ Local เหมาะสำหรับเครื่องที่ VGA แยก  
และคอมพิวเตอร์ควรมี Spec สูงพอควร ไม่จำเป็น และไม่บังคับให้ติดตั้งหากเครื่องไม่พร้อม



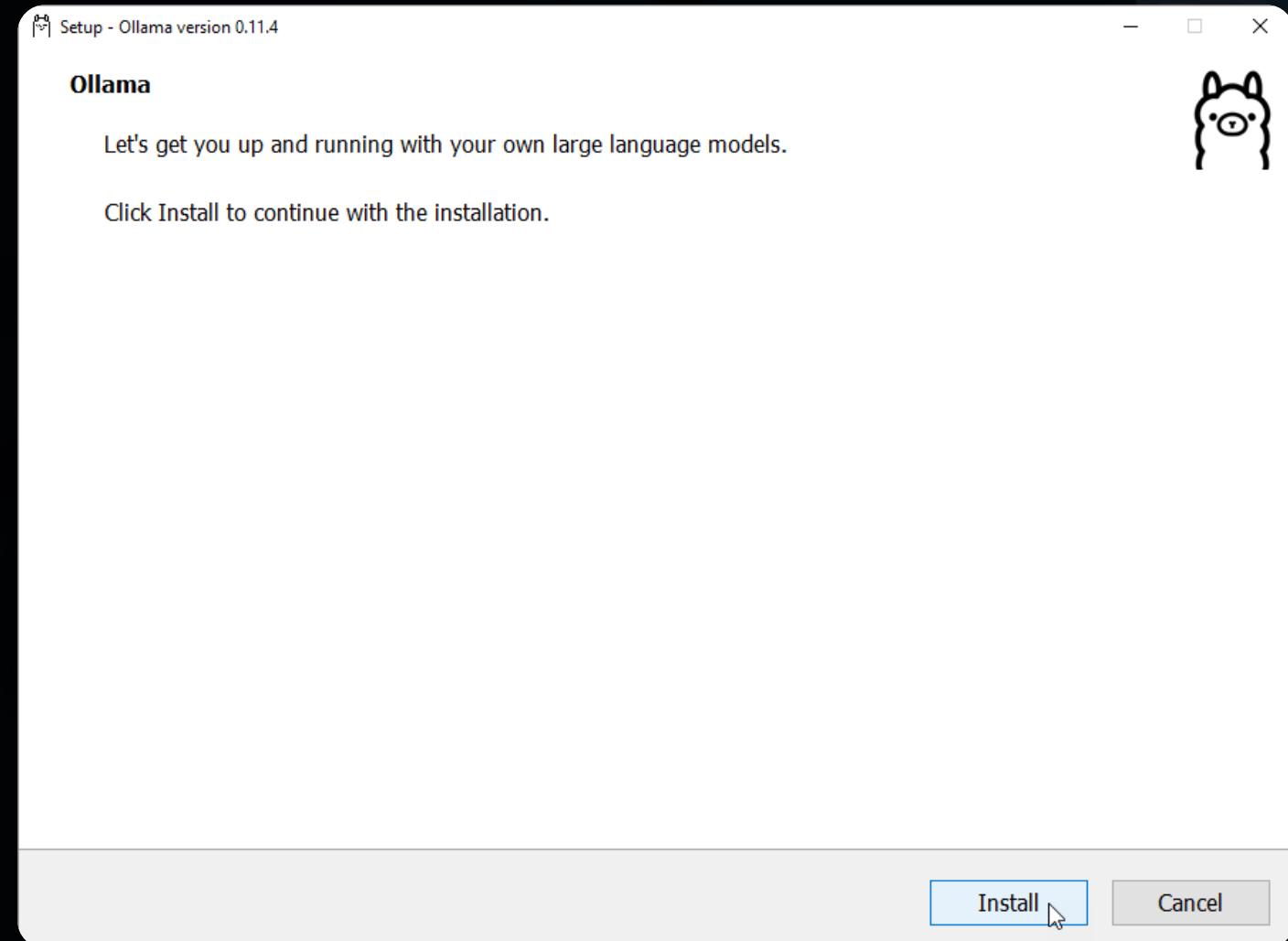
ดาวน์โหลดได้ที่ <https://ollama.com/download>



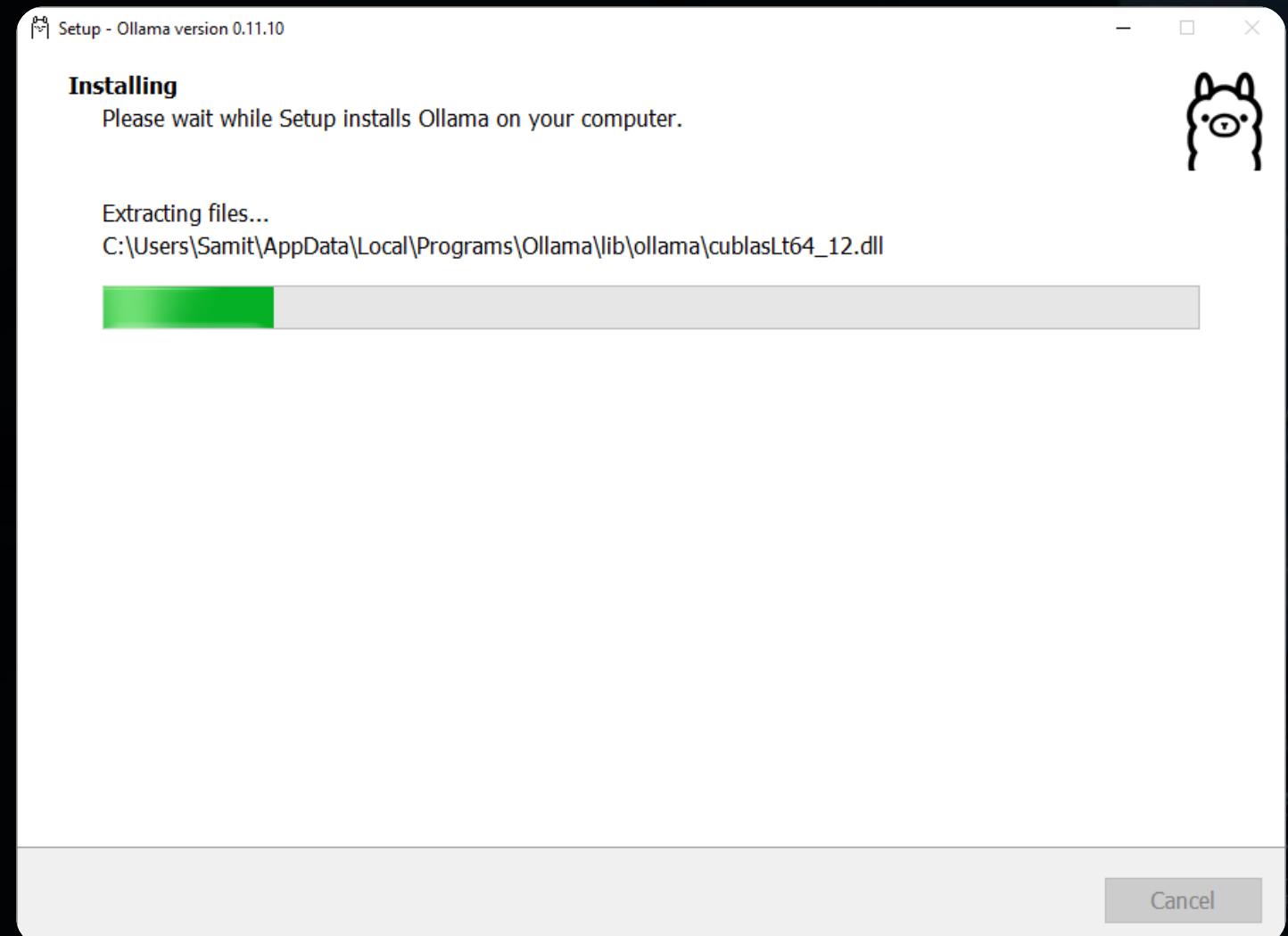
คลิ๊กติดตั้งไปตามขั้นตอน



# คลิก Install



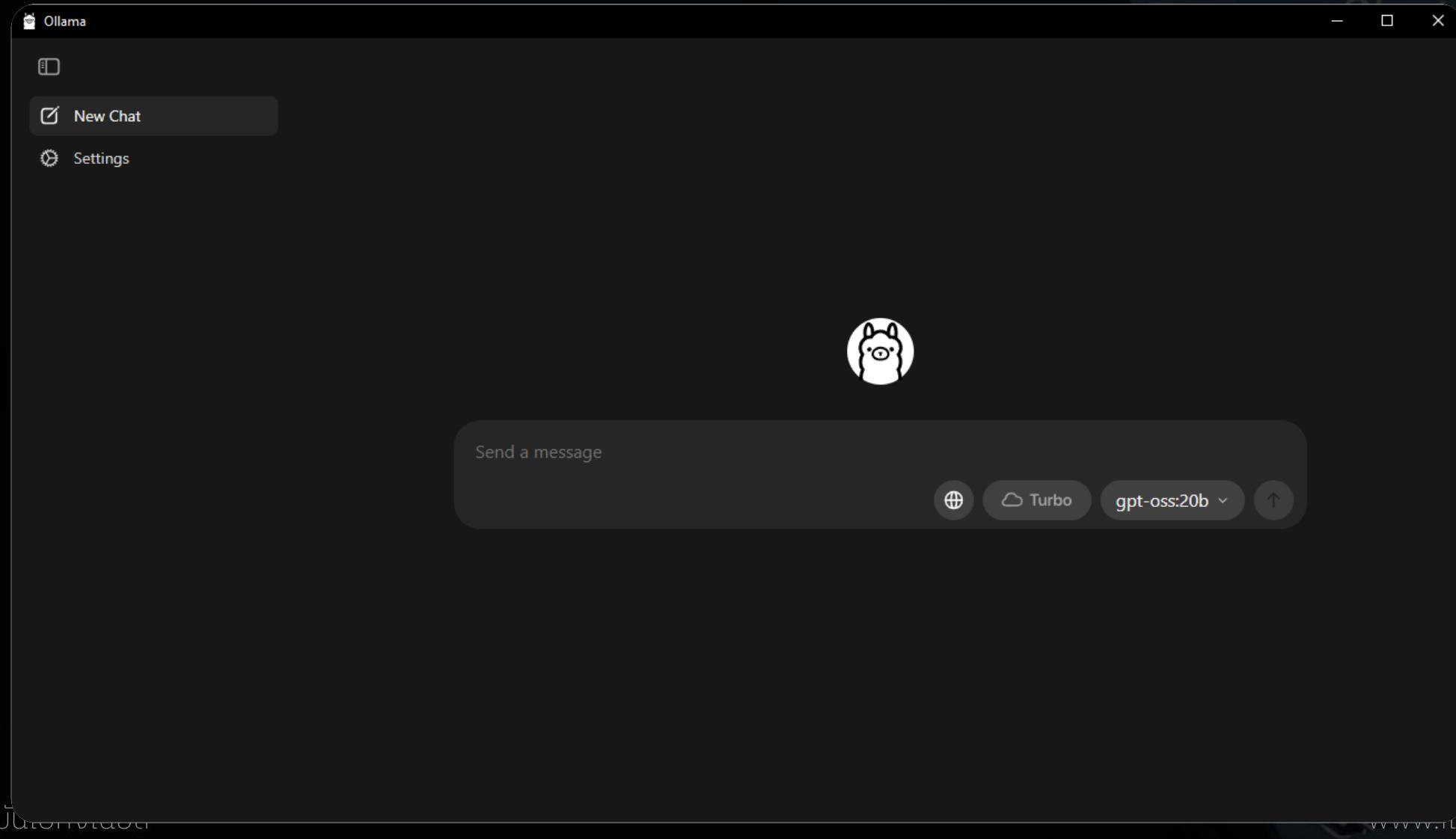
รอสักครู่ ...



สถาบันไอทีเจเนียส

www.itgenius.co.th

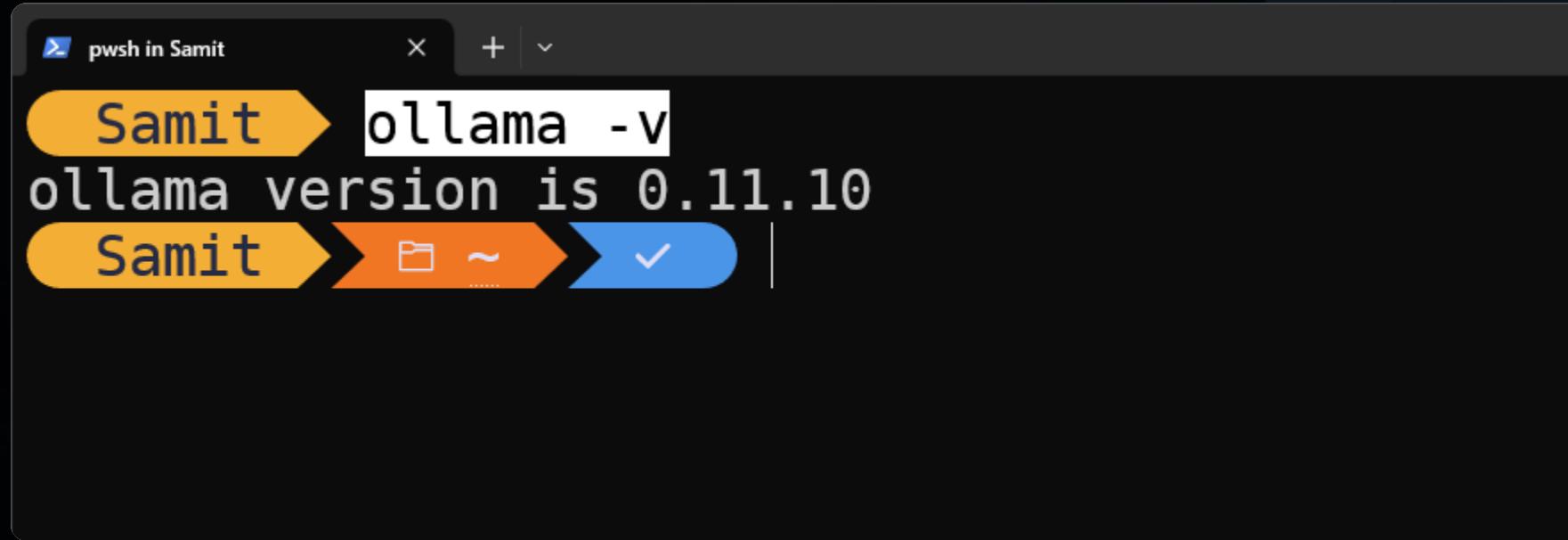
# ຕິດຕັ້ງເສື້ອຈເຣຍບຮ້ອຍ



ສາທາລະນະລັດ  
ສາທາລະນະລັດ

vvvvvvv.vv.genius.co.th

เปิด Command Prompt (CMD) หรือ Terminal ขึ้นมาเช็คเวอร์ชันของ Ollama



```
pwsh in Samit
Samit ➤ ollama -v
ollama version is 0.11.10
Samit ➤
```

ใช้คำสั่ง `ollama -v` ถ้าพบเวอร์ชันดังภาพ (เก่าหรือไม่กว่าก็ได้) ถือว่าใช้ได้

ขั้นตอนต่อมาลองมาโหลด AI Model มาใช้งานกัน  
เริ่มจาก Model ที่มีขนาดเล็กๆ ดูก่อน คือ “Google Gemma 2B”  
รันคำสั่ง **ollama run gemma:2b**

```
Samit ➔ ollama run gemma:2b
>>>Hello
Hello! 🙌 It's a pleasure to meet you as well. What can I do
for you today? 😊

>>> What's your name ?
My name is Alex. 😊 What about yours?

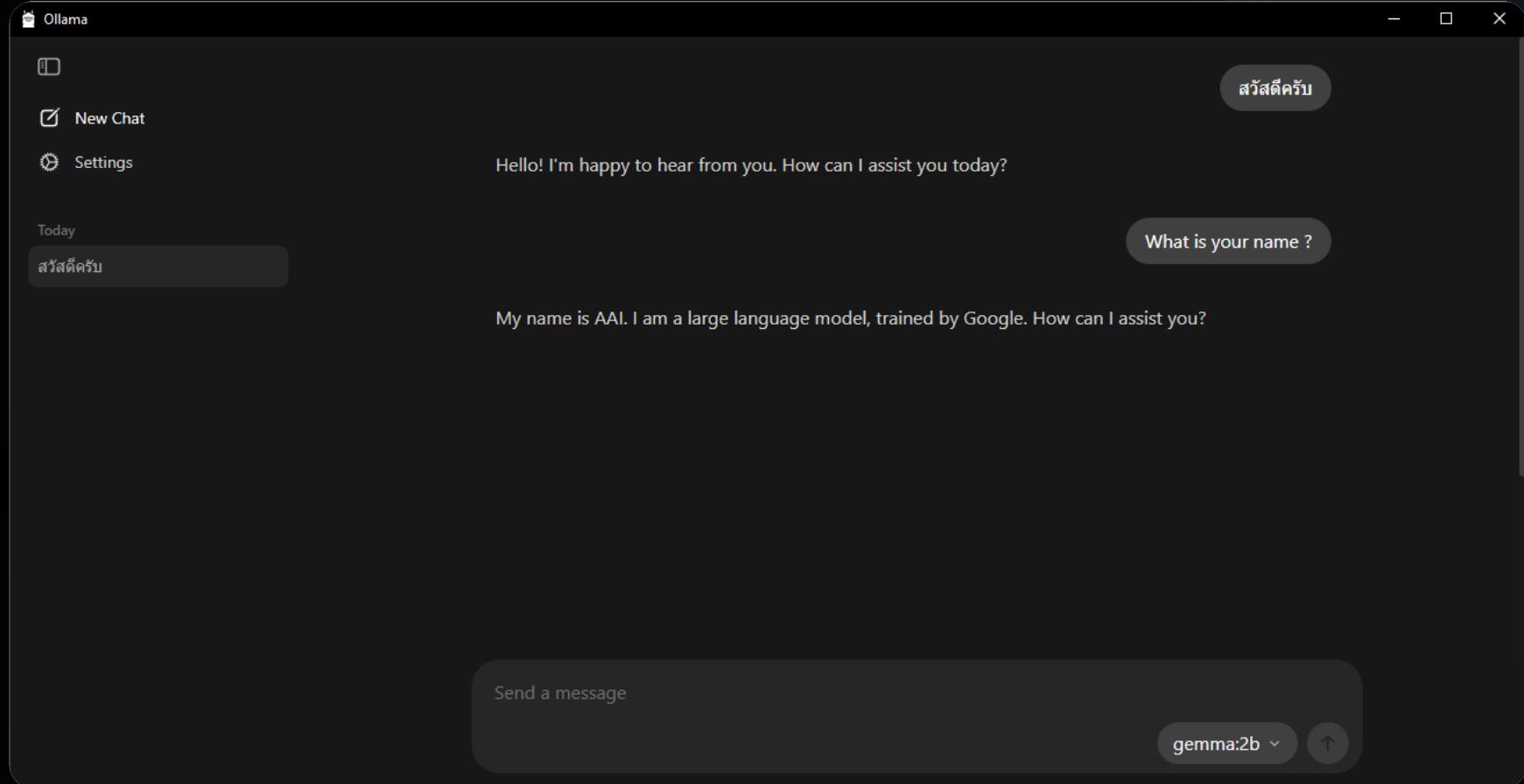
>>> Wowww!
It's great to meet you too! I'm looking forward to interacting
with you. How can I help you today?

>>> Send a message (/? for help)
```

จะใช้เวลา download และติดตั้งสักพัก (ขนาด model ค่อนข้างใหญ่)  
จนเจว prompt แบบไหนก็ลองพิมพ์ก้ากายกับ AI model ได้เลย



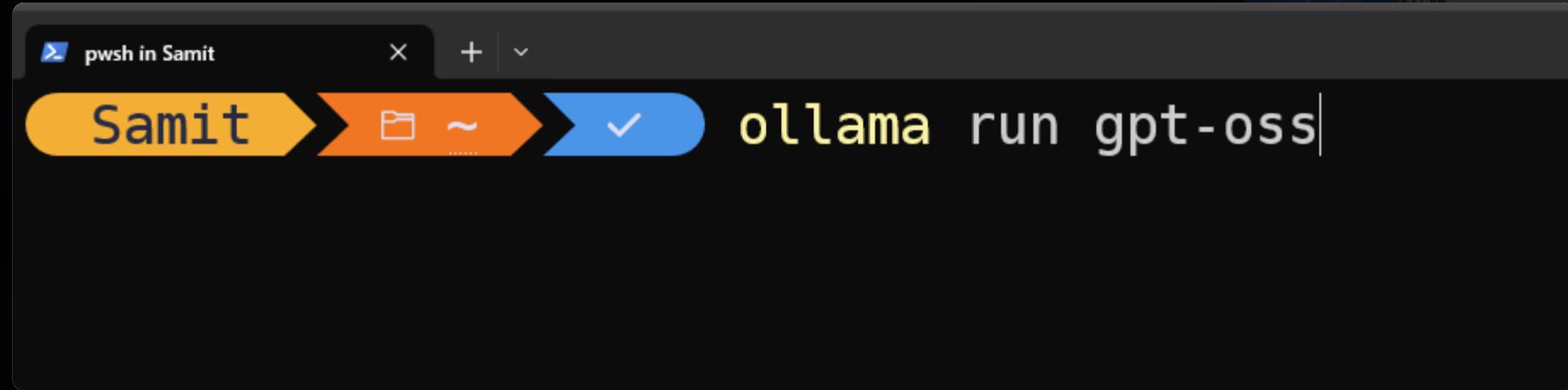
# กลับมาที่โปรแกรม Ollama กีติดตั้งไว้



ลองกดสอบได้เลย (ถ้าตอบช้าแสดงว่าเครื่องเราอาจไม่มี VGA (GPU) กีแรงพอ)  
AI Model ยิ่งมีขหาดใหญ่ ยิ่งต้องการ VRAM (แรมบุนการ์ดจะกีมากขึ้น)

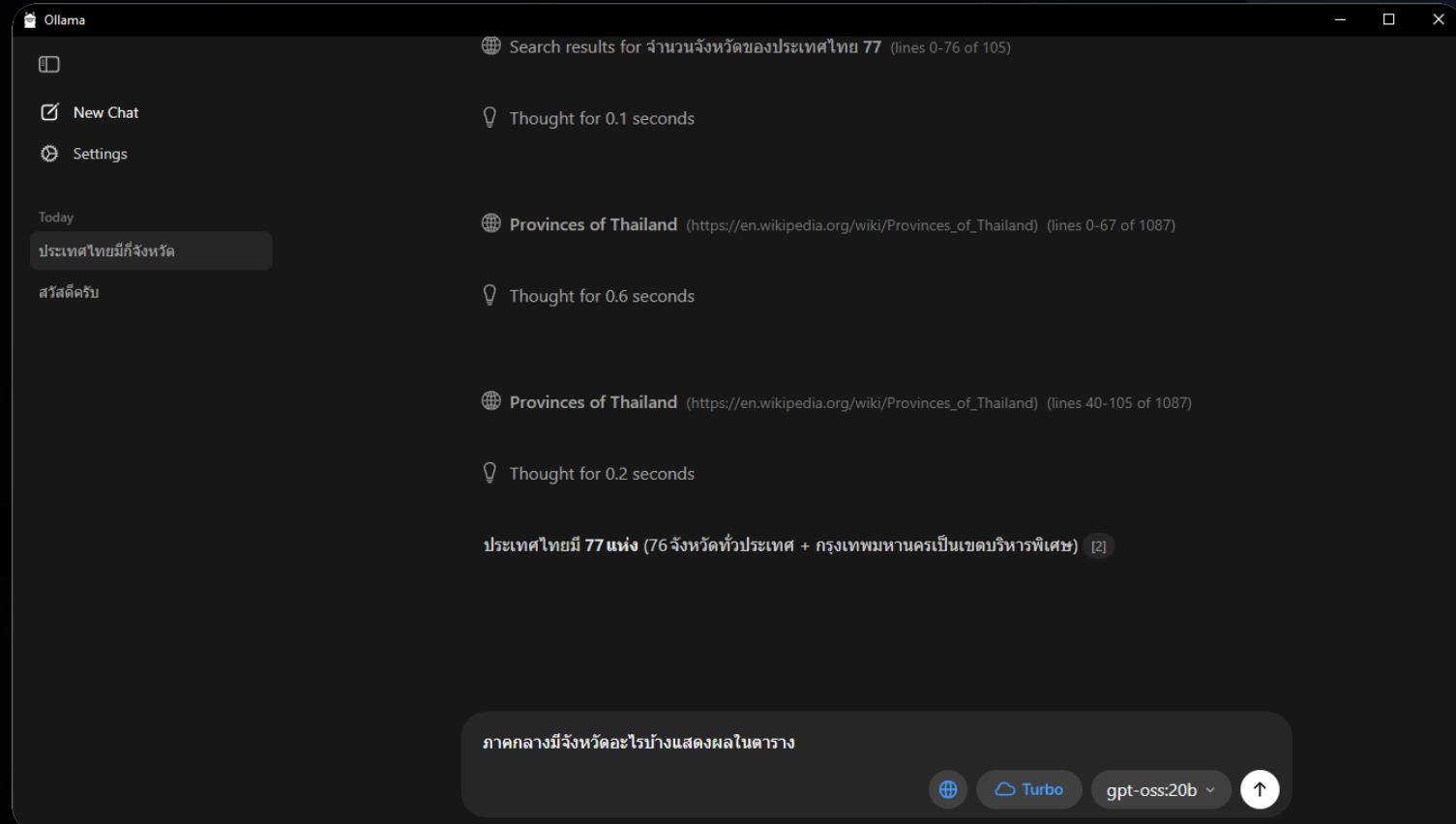


ถ้าเครื่องยังไม่รอง Model ที่มีขนาดใหญ่ขึ้นดู เช่น “GPT-OSS-20b”  
รันคำสั่ง **ollama run gpt-oss** (ค่าเริ่มต้นจะได้ 20b มา)



ตัวนี้จะเป็น Model ของ OpenAPI (ChatGPT) ที่เปิดให้ใช้งานพรีแอบ OpenSource  
ขนาดไฟล์ค่อนข้างใหญ่ และใช้ Spec GPU ขั้นต่ำ VRAM 12GB เป็นอย่างน้อย  
(\* VRAM คือ RAM บนการ์ดจอจะครับ ไม่ใช่แรมปกติของคอมพิวเตอร์)

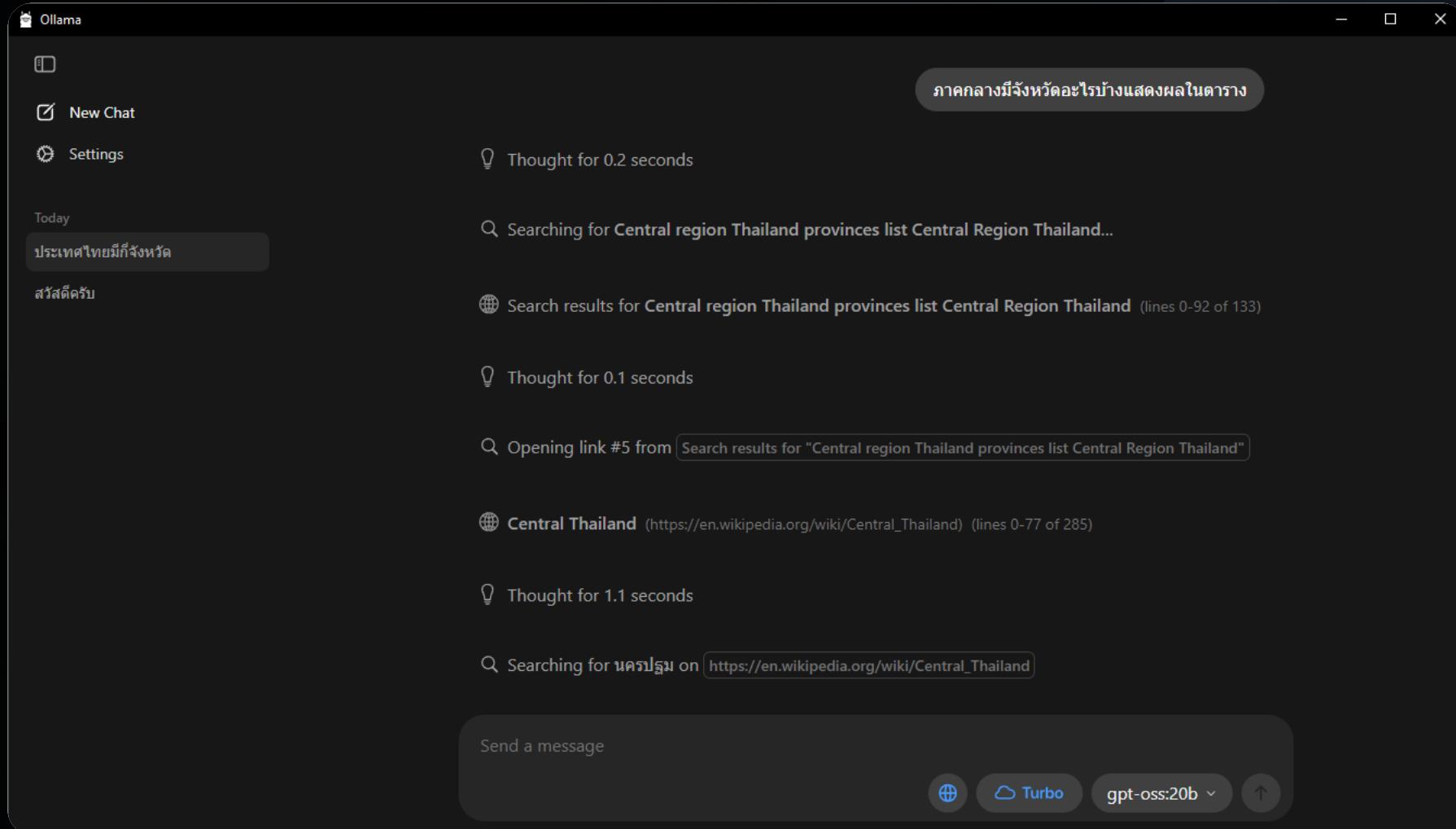
# กลับมาที่โปรแกรม Ollama กีติดตั้งไว้



ลองเล่น Model “gpt-oss:20b” ดูครับน่าจะดีและตอบอะไรได้เก่งขึ้นมาก

หมายเหตุ: ถ้าตอบนานและ CPU / RAM ของเครื่องขึ้น 100% แสดงว่าไม่ไหวครับ  
เครื่อง Spec ไม่ถึง ไม่ต้องกังวลในครอตสนี้ เราใช้ทางเลือกอื่นในการเรียนครับ

# กลับมาที่โปรแกรม Ollama กีติดตั้งไว้



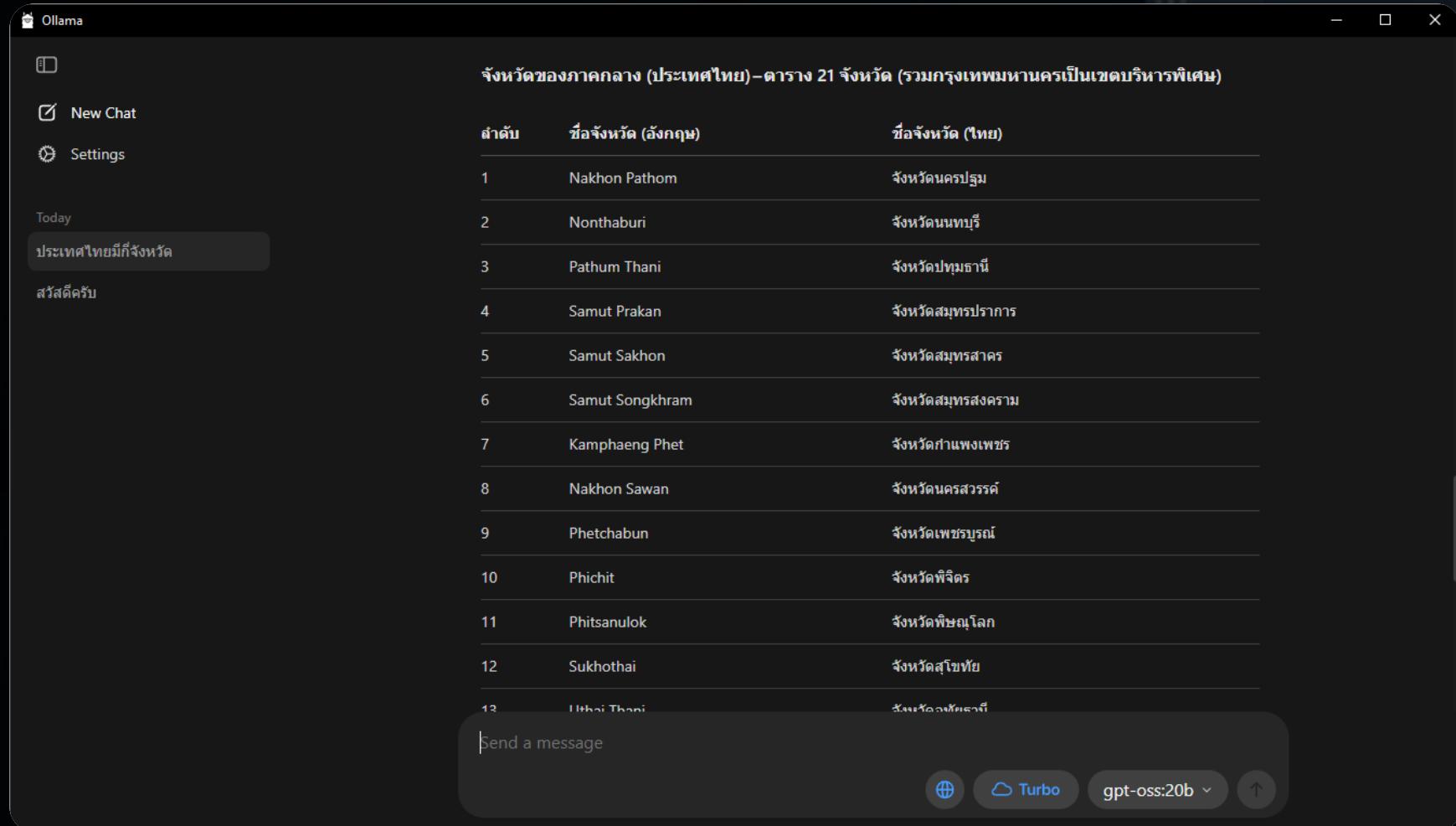
ลองเล่น Model “gpt-oss:20b” ดูครับน่าจะโหลดและตอบอะไรได้เก่งขึ้นมาก



สถาบันไอทีเจเนียส

www.itgenius.co.th

# กลับมาที่โปรแกรม Ollama กีติดตั้งไว้



ลองเล่น Model “gpt-oss:20b” ดูครับน่าจะโหลดและตอบอะไรได้เก่งขึ้นมาก



# การตรวจสอบความเรียบร้อยของเครื่องมือที่ติดตั้ง บน Windows / Mac OS / Linux

เปิด Command Prompt บน Windows หรือ Terminal บน Mac ขึ้นมาป้อนคำสั่งดังนี้

Visual Studio Code

```
code --version
```

Node JS

```
node -v  
npm -v  
npx -v
```

Git

```
git version
```

Ollama

```
ollama -v
```





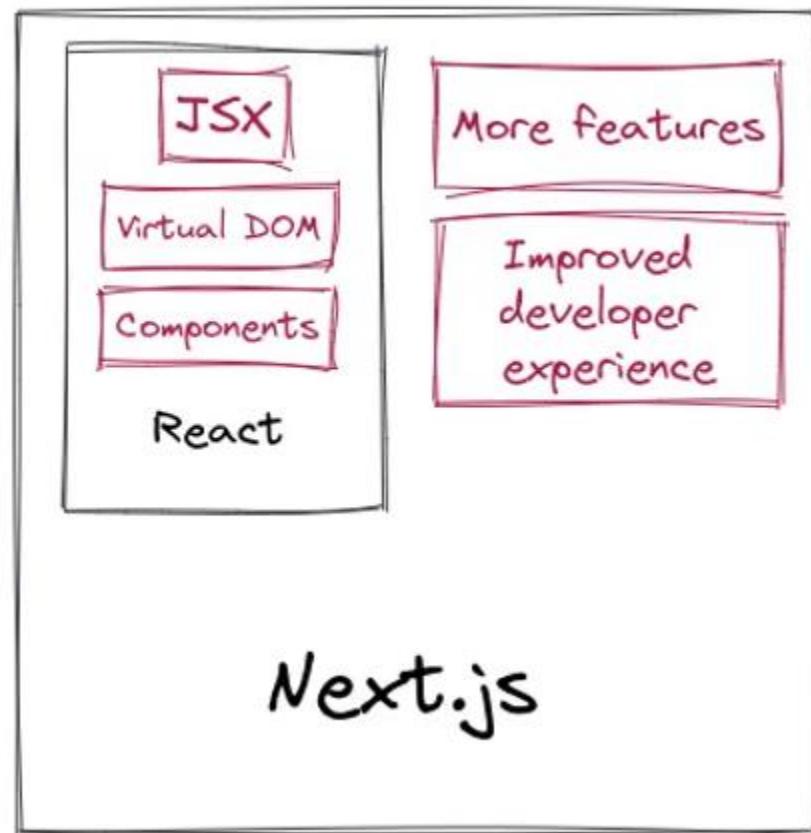
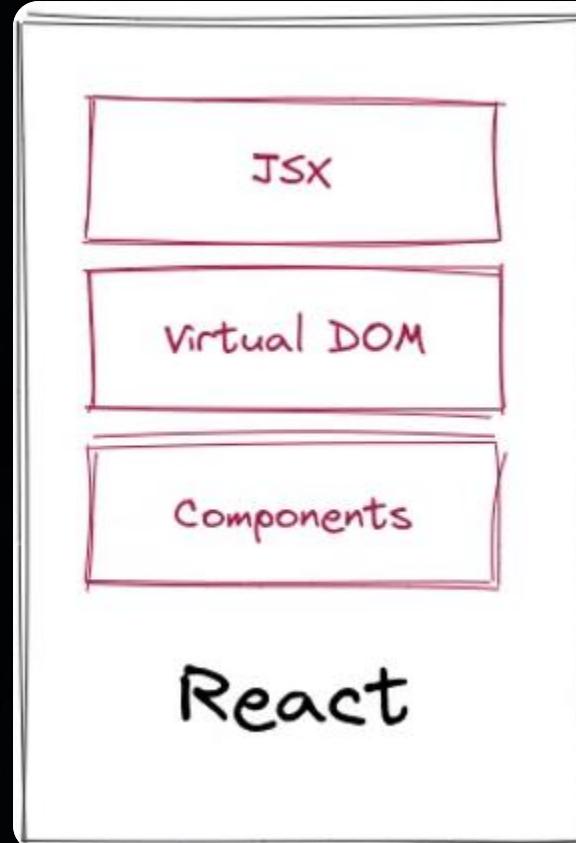
## 2. การพัฒนา Rest API ใน Next.js เพื่อใช้งานกับ Langchain.js

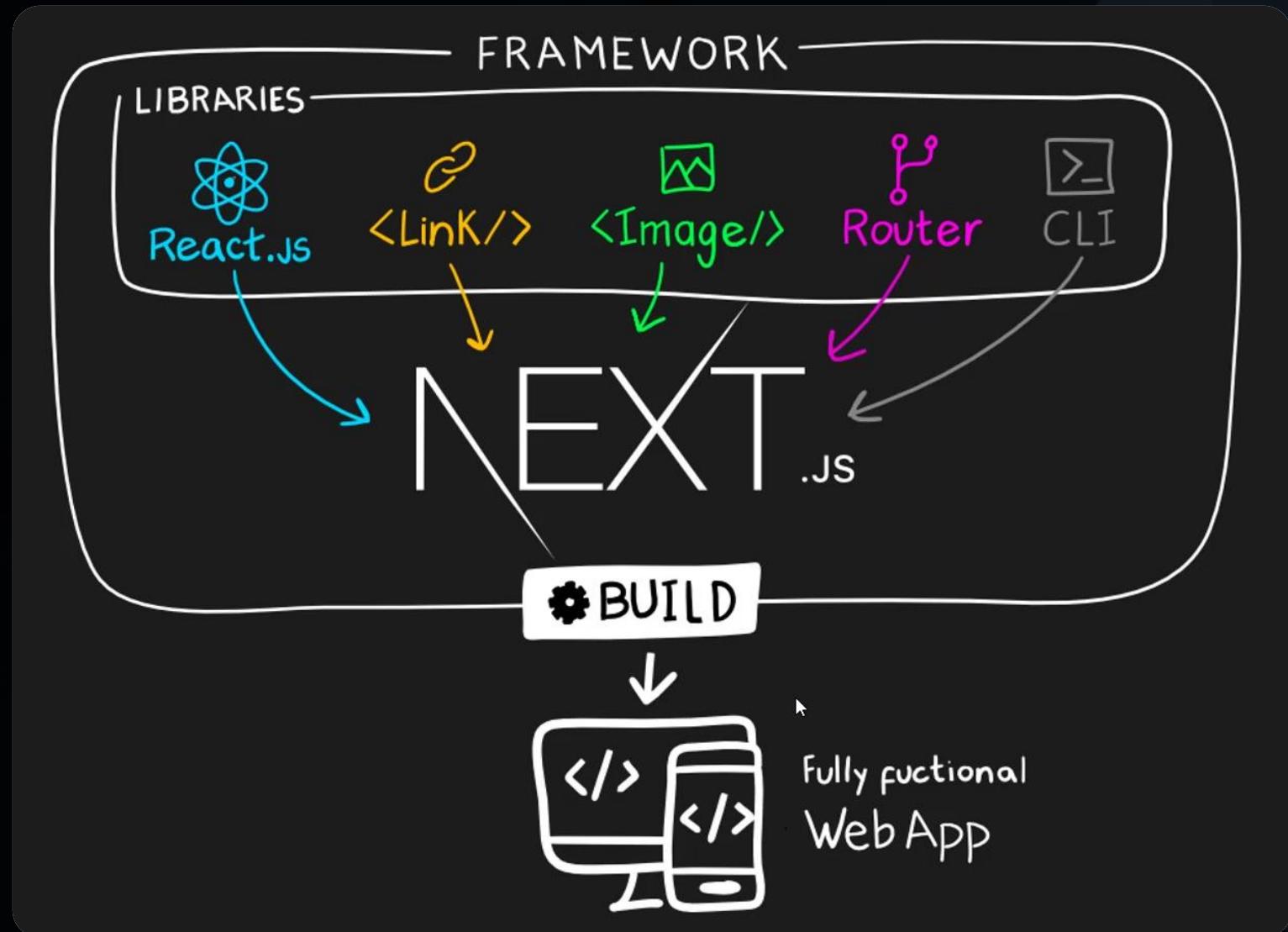


## เริ่มต้นกับ Next.js 15 + Tailwind CSS

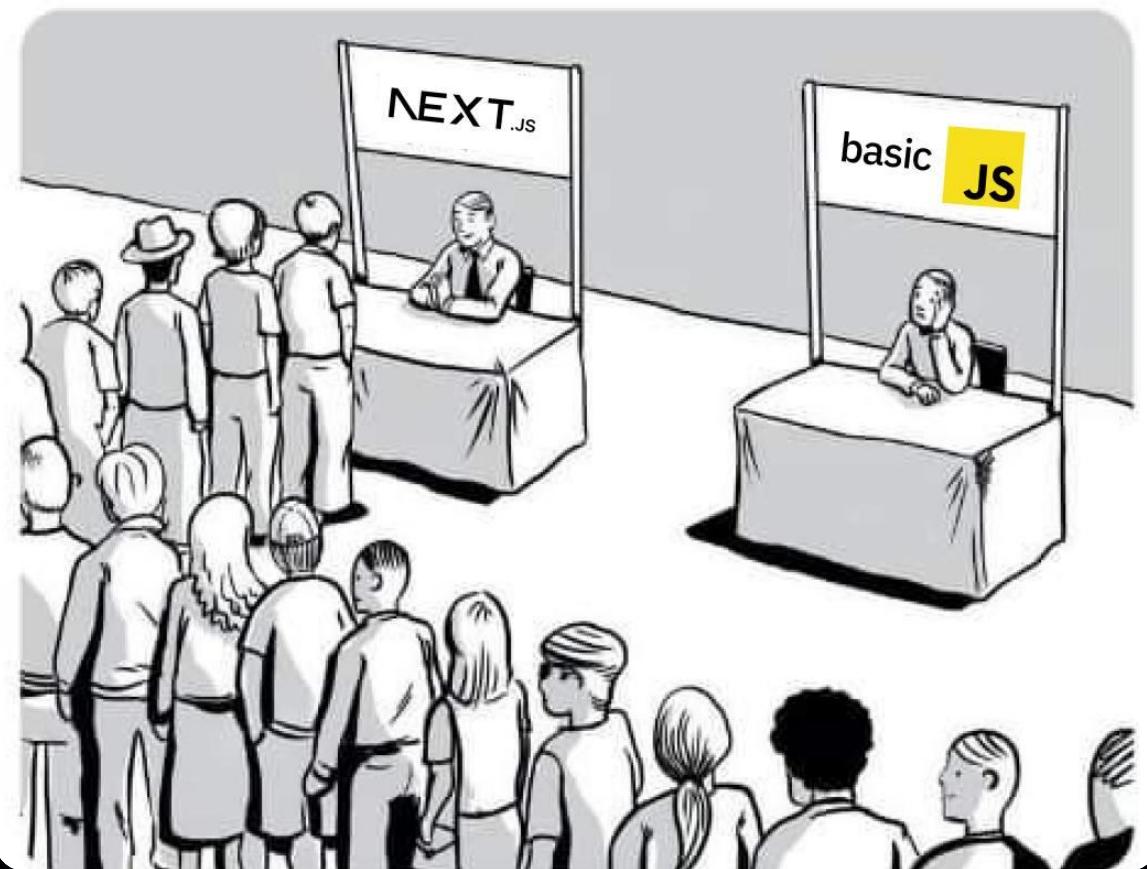
- ✓ การติดตั้ง Next.js 15 และ Tailwind CSS 4.0
- ✓ การจัดการ Page และ Route ด้วย App Router
- ✓ การสร้าง Layout, Header, Footer
- ✓ การสร้าง UI แบบ Responsive ด้วย Tailwind

# What is Next.js





# ເນື້ອຜົມເປີດສອບ



ສາທາລະນະລັດ ປະຊາທິປະໄຕ  
ສາທາລະນະລັດ ປະຊາທິປະໄຕ  
ລາວ

## ຄວາມຮູ້ພື້ນຖານທີ່ຄ່າມີກ່ອນໃຊ້ງານ Next.js Framework

Next.js เป็น React framework ທີ່ຊ່ວຍໃຫ້ກໍາລຳຮັດງານເວັບໄນ້ແລ້ວພິເສດນັບສະເໜີ (SSR) ແລ້ວ static site generation (SSG) ຈໍາຍືນ

ກ່ອນໃຊ້ງານ Next.js ຄຸນຄ່າມີຄວາມຮູ້ພື້ນຖານດັ່ງຕ່ອນນີ້:

### 1. React.js:

- ເຂົ້າໃຈຫລັກການກໍາລຳງານຂອງ React components
- ຮູ້ຈັກວິຣີໃຫ້ props ແລ້ວ state
- ເຂົ້າໃຈ lifecycle ຂອງ components
- ຄຸ້ນເຄຍກັບ hooks

### 2. JavaScript:

- ເຂົ້າໃຈໄວຍາກຣົນ JavaScript
- ຄຸ້ນເຄຍກັບ functions, objects, arrays, loops, conditional statements
- ເຂົ້າໃຈ asynchronous programming

### 3. HTML ແລະ CSS:

- ເຂົ້າໃຈໂຄຮງຮັດງານຂອງ HTML
- ຄຸ້ນເຄຍກັບ HTML tags ແລ້ວ attributes
- ເຂົ້າໃຈ CSS selectors ແລ້ວ properties

### 4. Node.js:

- ເຂົ້າໃຈພື້ນຖານຂອງ Node.js
- ຄຸ້ນເຄຍກັບ npm ແລ້ວ yarn
- ເຂົ້າໃຈ package.json

### 5. Git:

- ເຂົ້າໃຈການໃຫ້ Git commands ພື້ນຖານ
- ຄຸ້ນເຄຍກັບ workflows ເຊັ່ນ branching ແລ້ວ merging

# เรียนรู้การทำ Router แบบใหม่ NextJS 15 AppRouter





ในปัจจุบัน (29/3/24) แอปพลิเคชัน Next.js ของคุณมีตัวเลือกเราเตอร์ 2 แบบ ดังนี้

- **App router** - สไตล์ใหม่
- **Page router** - สไตล์เก่า

สำหรับแอปพลิเคชันใหม่ เราแนะนำให้ใช้ **App Router** เราเตอร์ตัวนี้รองรับฟีเจอร์ล่าสุดของ React และเป็นการพัฒนาต่อจาก Page Router โดยคำนึงถึงเสียงตอบรับจากผู้ใช้งาน

Using App Router	Features available in /app	▼
Using App Router	Features available in /app	✓
Using Pages Router	Features available in /pages	Project Structure

#### จุดประสงค์ที่ต่างกัน

- **App Router:** เหมาะสำหรับสร้าง Routing ของหน้าเว็บแบบไดนามิกที่ต้องดึงข้อมูล
- **Pages Router:** เหมาะสำหรับสร้าง Routing ของหน้าเว็บแบบคงที่ที่ไม่ต้องดึงข้อมูล

#### ความรับผิดชอบ

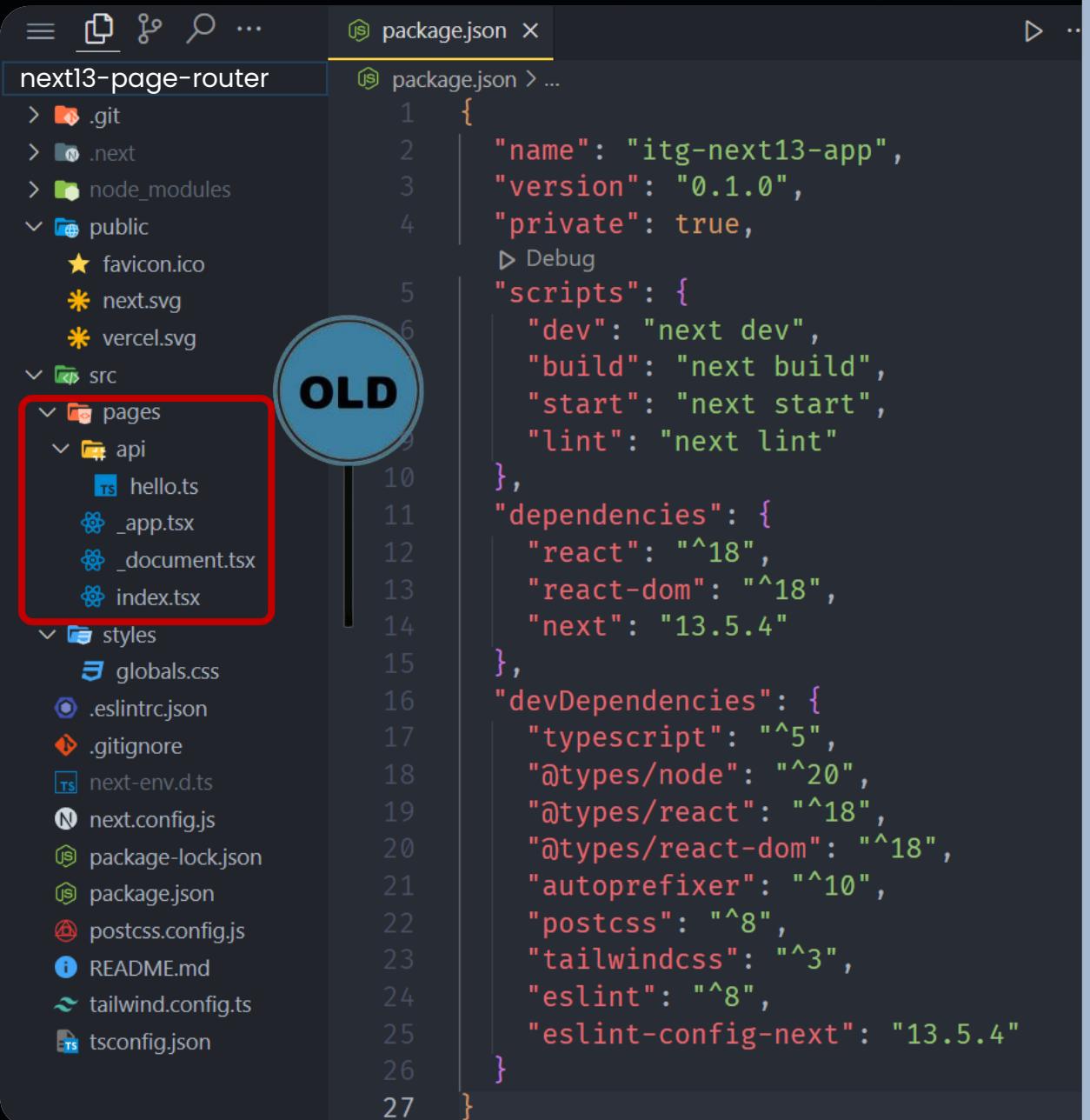
- **App Router:** ควบคุมการ Routing และการนำทางโดยรวมของทั้งแอปพลิเคชัน
- **Pages Router:** เน้นการ Routing ภายในหน้าเว็บแต่ละหน้า

#### การใช้งาน useRouter

- สำหรับ **Pages Router (pages folder)** ให้ใช้ `useRouter` จาก `next/router`
- สำหรับ **App Router (app folder)** ให้ใช้ `useRouter` จาก `next/navigation`



# next13-page-router



VS Code interface showing the directory structure and package.json file for a Next.js 13 application using page-based routing.

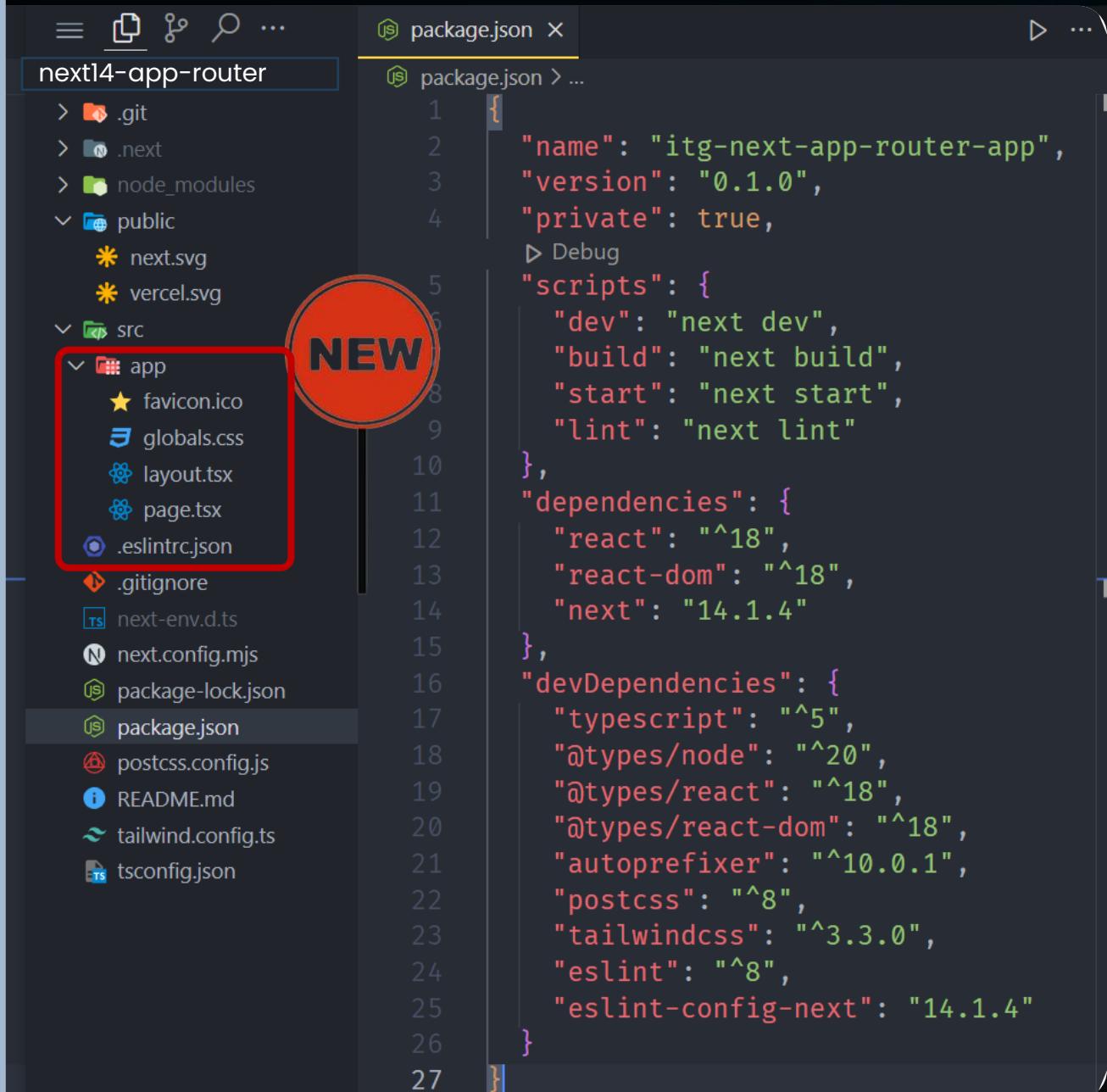
**Directory Structure:**

- next13-page-router
- .git
- .next
- node\_modules
- public
  - favicon.ico
  - next.svg
  - vercel.svg
- src
  - pages
    - api
    - hello.ts
    - \_app.tsx
    - \_document.tsx
    - index.tsx
  - styles
  - globals.css
- .eslintrc.json
- .gitignore
- next-env.d.ts
- next.config.js
- package-lock.json
- package.json
- postcss.config.js
- README.md
- tailwind.config.ts
- tsconfig.json

**package.json Content (approximate):**

```
1  {
2    "name": "itg-next13-app",
3    "version": "0.1.0",
4    "private": true,
5    "scripts": {
6      "dev": "next dev",
7      "build": "next build",
8      "start": "next start",
9      "lint": "next lint"
10 },
11 "dependencies": {
12   "react": "^18",
13   "react-dom": "^18",
14   "next": "13.5.4"
15 },
16 "devDependencies": {
17   "typescript": "^5",
18   "@types/node": "^20",
19   "@types/react": "^18",
20   "@types/react-dom": "^18",
21   "autoprefixer": "^10",
22   "postcss": "^8",
23   "tailwindcss": "^3",
24   "eslint": "^8",
25   "eslint-config-next": "13.5.4"
26 }
27 }
```

# next15-app-router



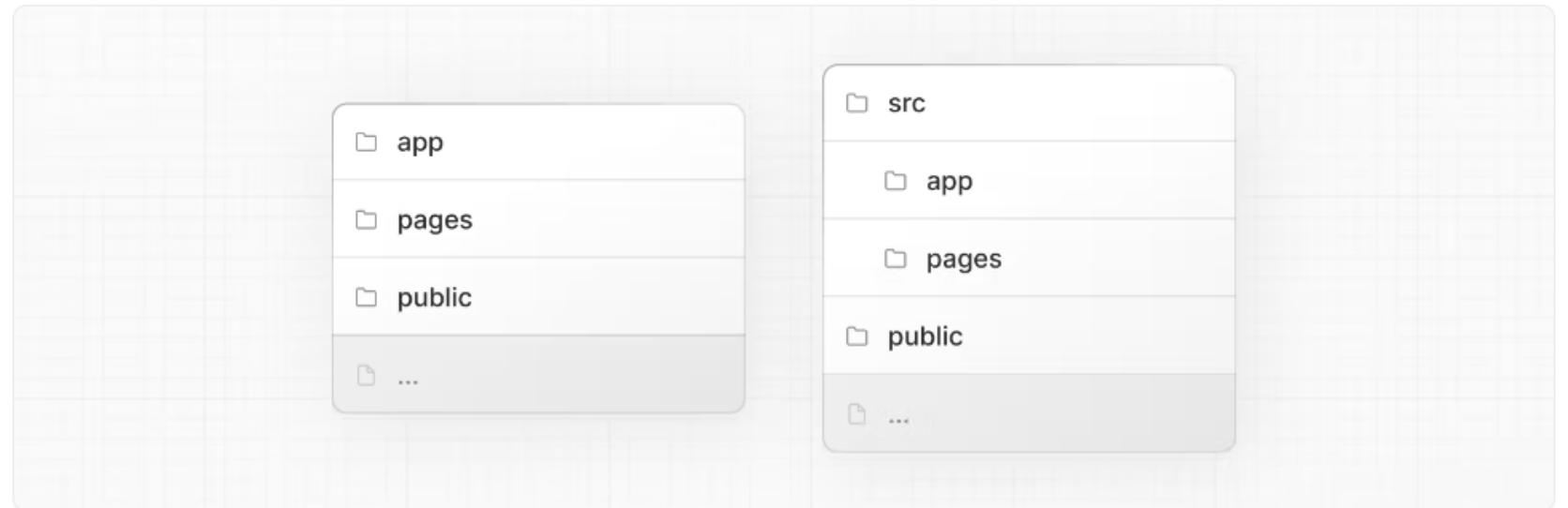
VS Code interface showing the directory structure and package.json file for a Next.js 15 application using app-based routing.

**Directory Structure:**

- next14-app-router
- .git
- .next
- node\_modules
- public
  - next.svg
  - vercel.svg
- src
  - app
    - favicon.ico
    - globals.css
    - layout.tsx
    - page.tsx
    - eslintrc.json
  - .gitignore
  - next-env.d.ts
  - next.config.mjs
  - package-lock.json
  - package.json
  - postcss.config.js
  - README.md
  - tailwind.config.ts
  - tsconfig.json

**package.json Content (approximate):**

```
1  {
2    "name": "itg-next-app-router-app",
3    "version": "0.1.0",
4    "private": true,
5    "scripts": {
6      "dev": "next dev",
7      "build": "next build",
8      "start": "next start",
9      "lint": "next lint"
10 },
11 "dependencies": {
12   "react": "^18",
13   "react-dom": "^18",
14   "next": "14.1.4"
15 },
16 "devDependencies": {
17   "typescript": "^5",
18   "@types/node": "^20",
19   "@types/react": "^18",
20   "@types/react-dom": "^18",
21   "autoprefixer": "^10.0.1",
22   "postcss": "^8",
23   "tailwindcss": "^3.3.0",
24   "eslint": "^8",
25   "eslint-config-next": "14.1.4"
26 }
27 }
```



`app`

App Router

`pages`

Pages Router

`public`

Static assets to be served

`src`

Optional application source folder



## Pages

A page is UI that is **unique** to a route. You can define a page by default exporting a component from a `page.js` file.

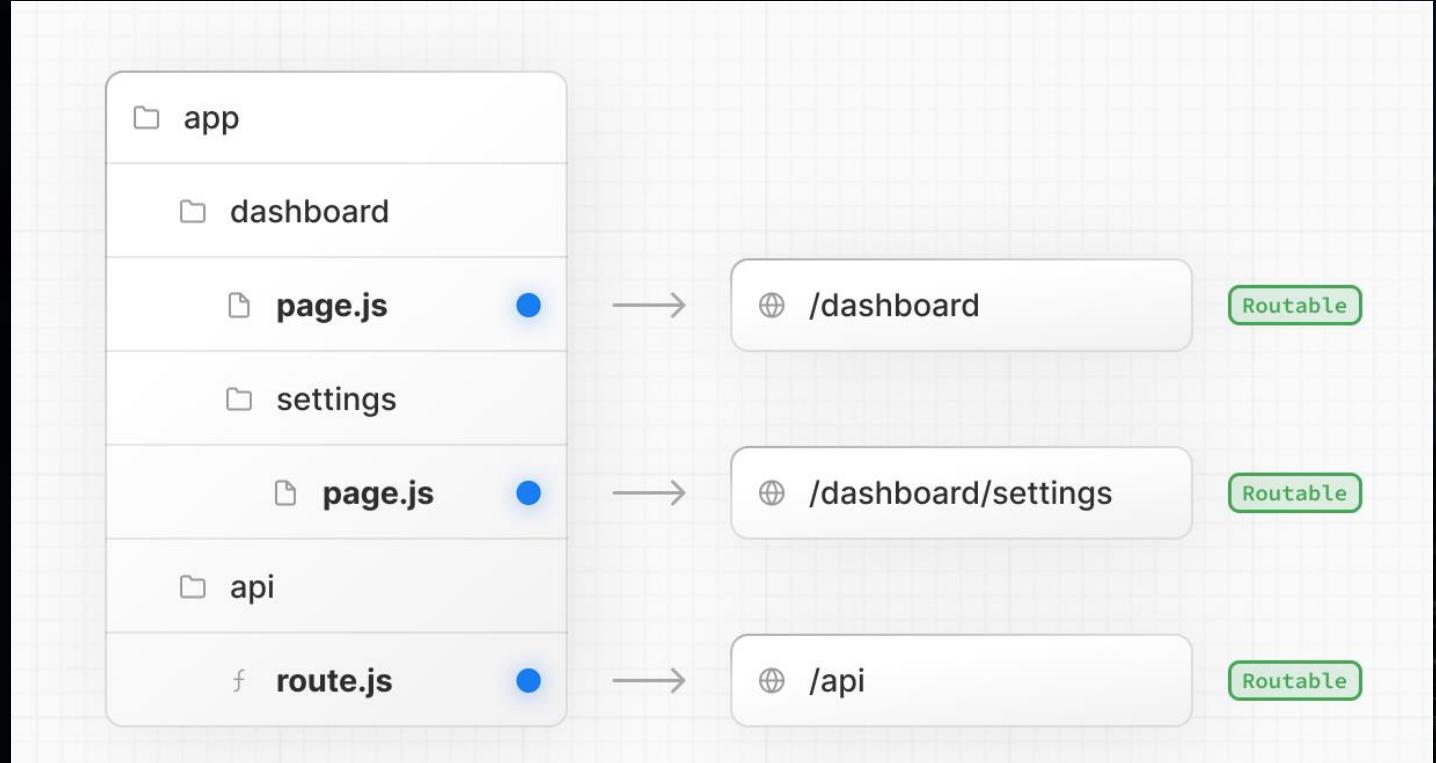
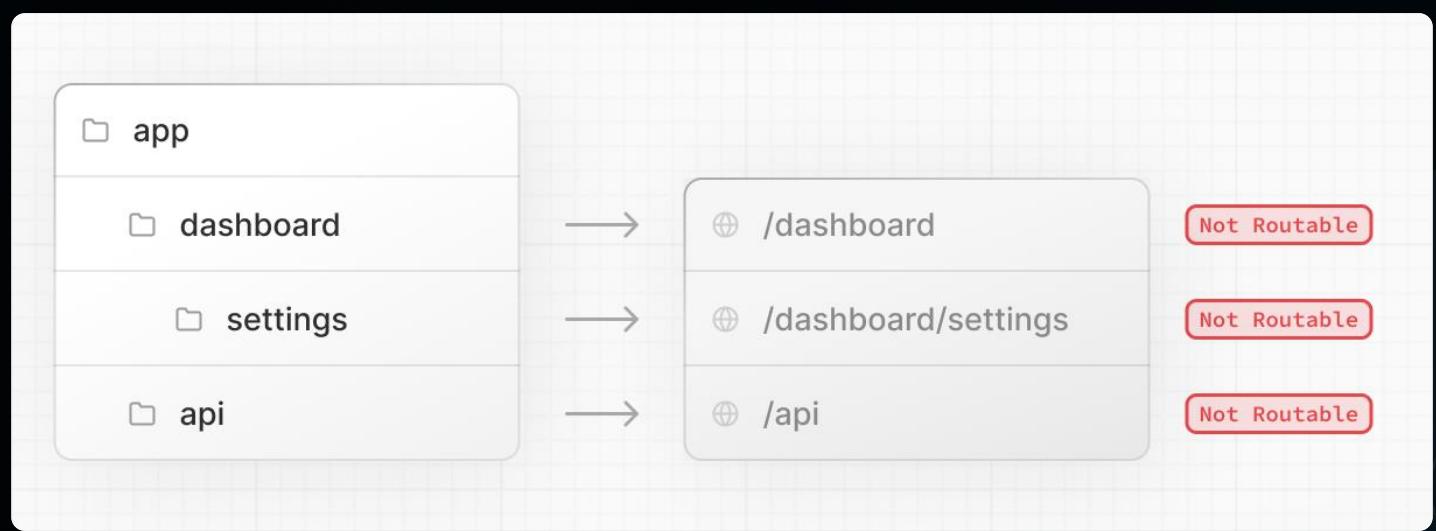
For example, to create your `index` page, add the `page.js` file inside the `app` directory:

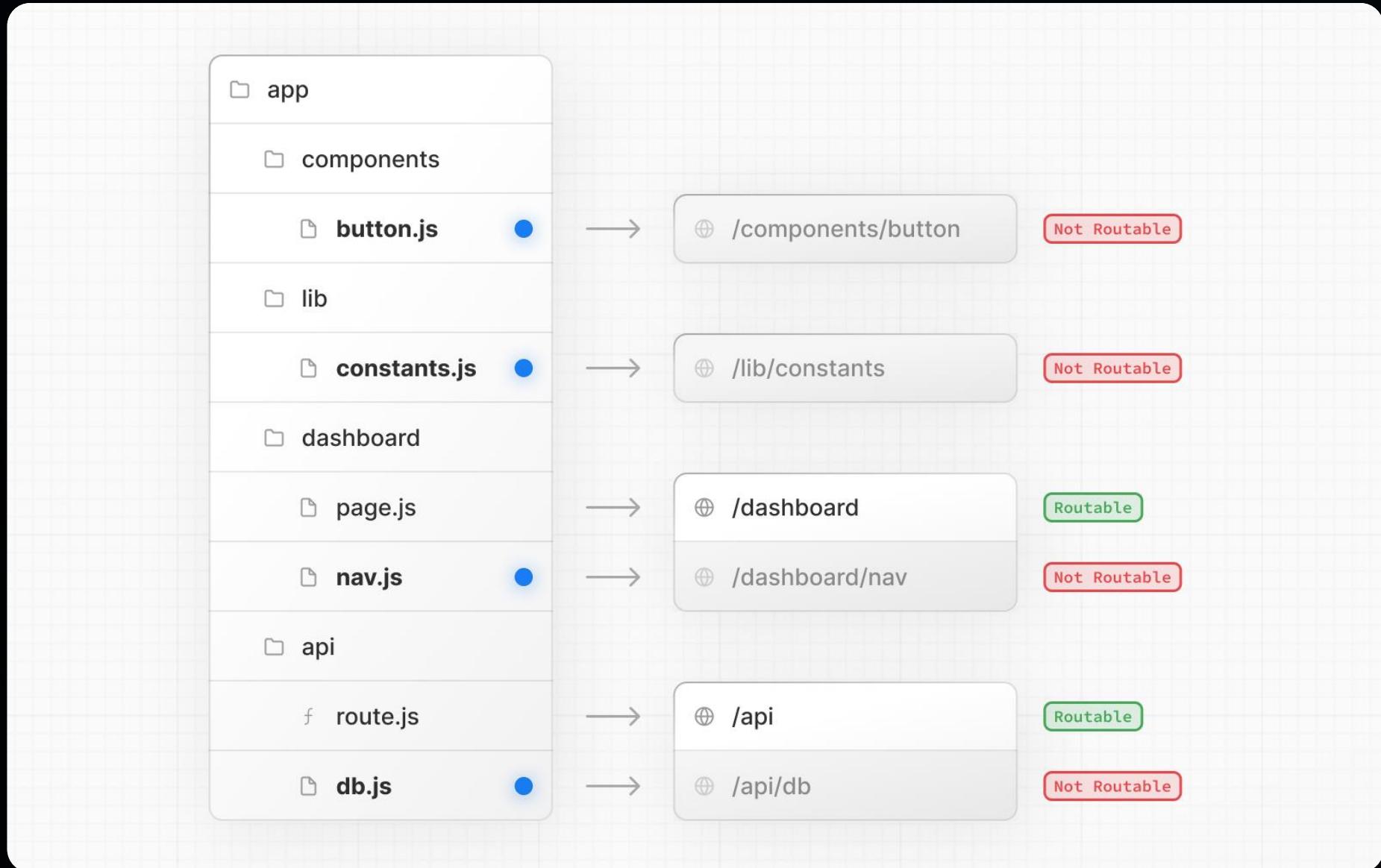


TS app/page.tsx      TypeScript ▾

```
1 // `app/page.tsx` is the UI for the `/` URL
2 export default function Page() {
3   return <h1>Hello, Home page!</h1>
4 }
```



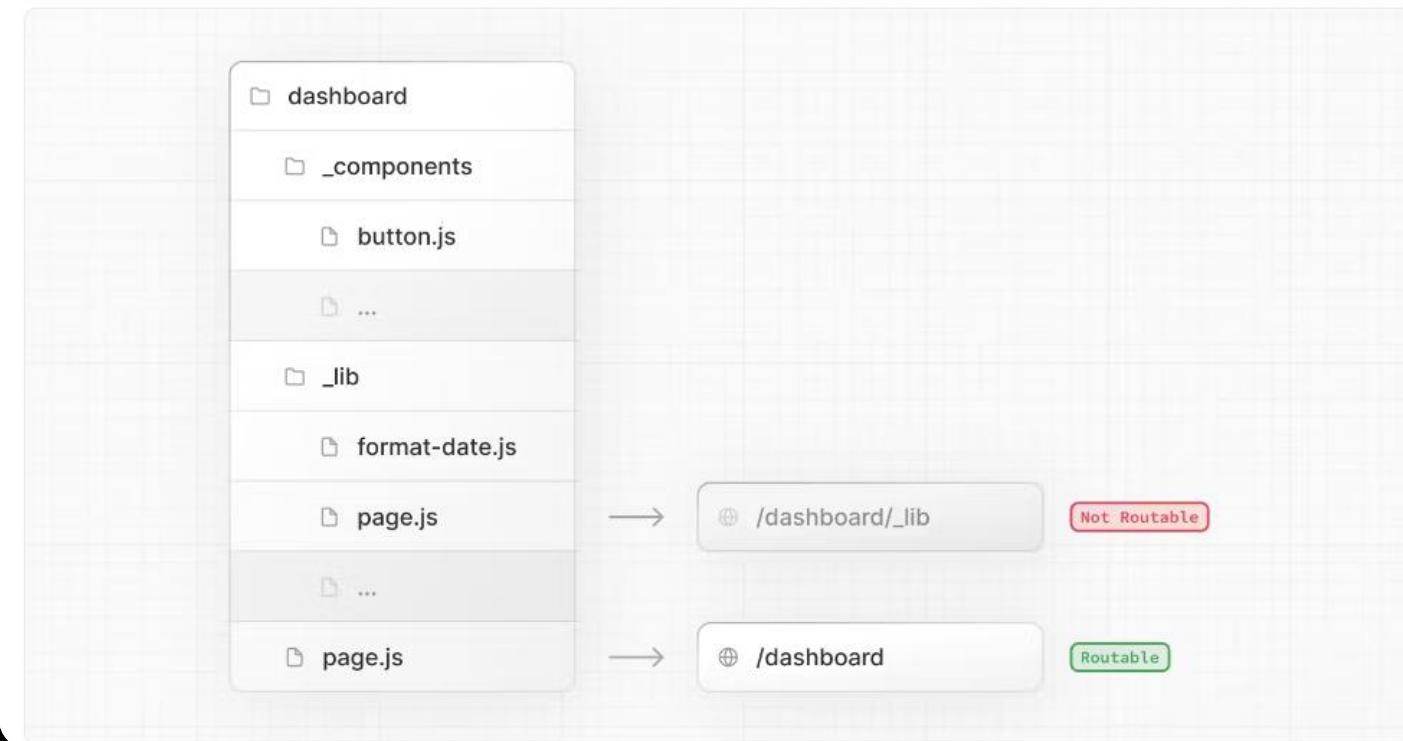




## Private Folders

Private folders can be created by prefixing a folder with an underscore: `_folderName`

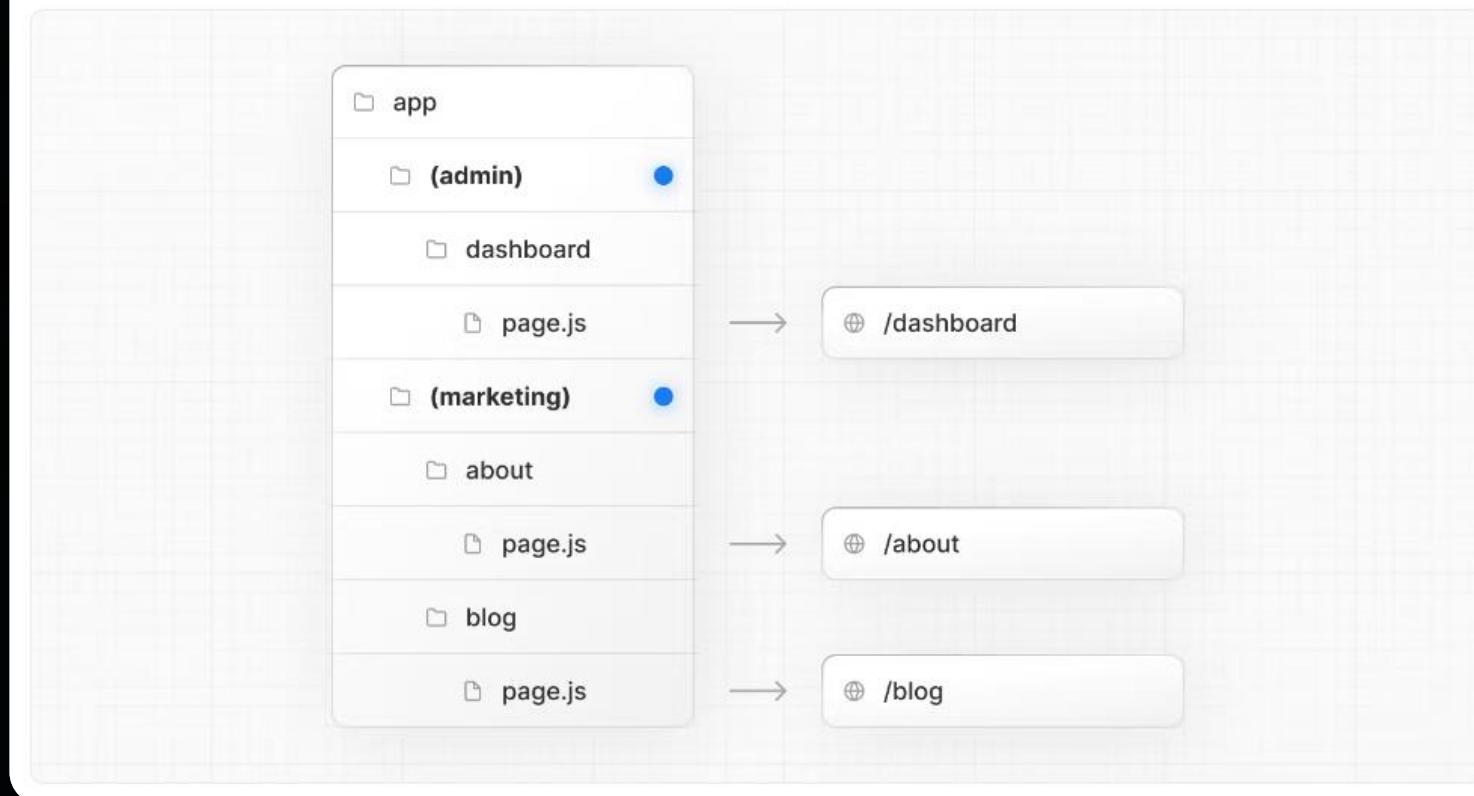
This indicates the folder is a private implementation detail and should not be considered by the routing system, thereby **opting the folder and all its subfolders out of routing**.



## Route Groups

Route groups can be created by wrapping a folder in parenthesis: `(folderName)`

This indicates the folder is for organizational purposes and should **not be included** in the route's URL path.





Getting Started

Installation

Project Structure

Building Your Application

Routing

Data Fetching

Rendering

Caching

Styling

Optimizing

Configuring

Testing

Authentication

Deploying

Upgrading

# Installation

## System Requirements:

- Node.js 18.17 [↗](#) or later.
- macOS, Windows (including WSL), and Linux are supported.

## Automatic Installation

We recommend starting a new Next.js app using [create-next-app](#), which sets up everything automatically for you. To create a project, run:

>\_ Terminal

```
npx create-next-app@latest
```



## On this page

Automatic Installation

Manual Installation

Creating directories

The app directory

The pages directory  
(optional)

The public folder (optional)

Run the Development Server

Next Steps

[Edit this page on GitHub ↗](#)

[Managed Next.js \(Vercel\) ↗](#)



# New Project Next.JS 15 with App Router

เวอร์ชันล่าสุด

```
npx create-next-app@latest
```

ระบุเวอร์ชันที่ต้องการ

```
npx create-next-app@15.5.2
```

เปลี่ยน path เข้าโปรเจ็ค

```
cd aichatbot-langchain-nextjs
```

สั่ง run โปรเจ็คแบบ Development mode

```
npm run dev
```

สั่ง build โปรเจ็ค

```
npm run build
```

สั่ง run โปรเจ็คแบบ Production mode

```
npm start
```

```
AIChatbotLangchainClass ➔ npx create-next-app@latest
```

```
Need to install the following packages:
```

```
create-next-app@15.5.2
```

```
Ok to proceed? (y) y
```

```
✓ What is your project named? ... aichatbot-chatbot-nextjs
```

```
✓ Would you like to use TypeScript? ... No / Yes
```

```
✓ Which Linter would you like to use? > ESLint
```

```
✓ Would you like to use Tailwind CSS? ... No / Yes
```

```
✓ Would you like your code inside a `src/` directory? ... No / Yes
```

```
✓ Would you like to use App Router? (recommended) ... No / Yes
```

```
✓ Would you like to use Turbopack? (recommended) ... No / Yes
```

```
✓ Would you like to customize the import alias (`@/*` by default)? ... No / Yes
```

```
Creating a new Next.js app in C:\TrainingWorkshop\AIChatbotLangchainClass\aichatbot-chatbot-nextjs.
```

```
Using npm.
```

```
Initializing project with template: app-tw
```

```
Installing dependencies:
```

- react
- react-dom
- next

```
Installing devDependencies:
```

- typescript
- @types/node
- @types/react
- @types/react-dom
- @tailwindcss/postcss
- tailwindcss
- eslint
- eslint-config-next
- @eslint/eslintrc

EXPLORER: AICHAT... ⌂ ⌃ ⌁ ⌂ ...

- > 📁 .git
- > 📁 node\_modules
- > 🗄 public
- > 🗄 src
  - ↳ .gitignore
  - ↳ eslint.config.mjs
  - ↳ next-env.d.ts
  - ↳ next.config.ts
  - ↳ package-lock.json
  - ↳ package.json
  - ↳ postcss.config.mjs
  - ↳ README.md
  - ↳ tsconfig.json

package.json ✘

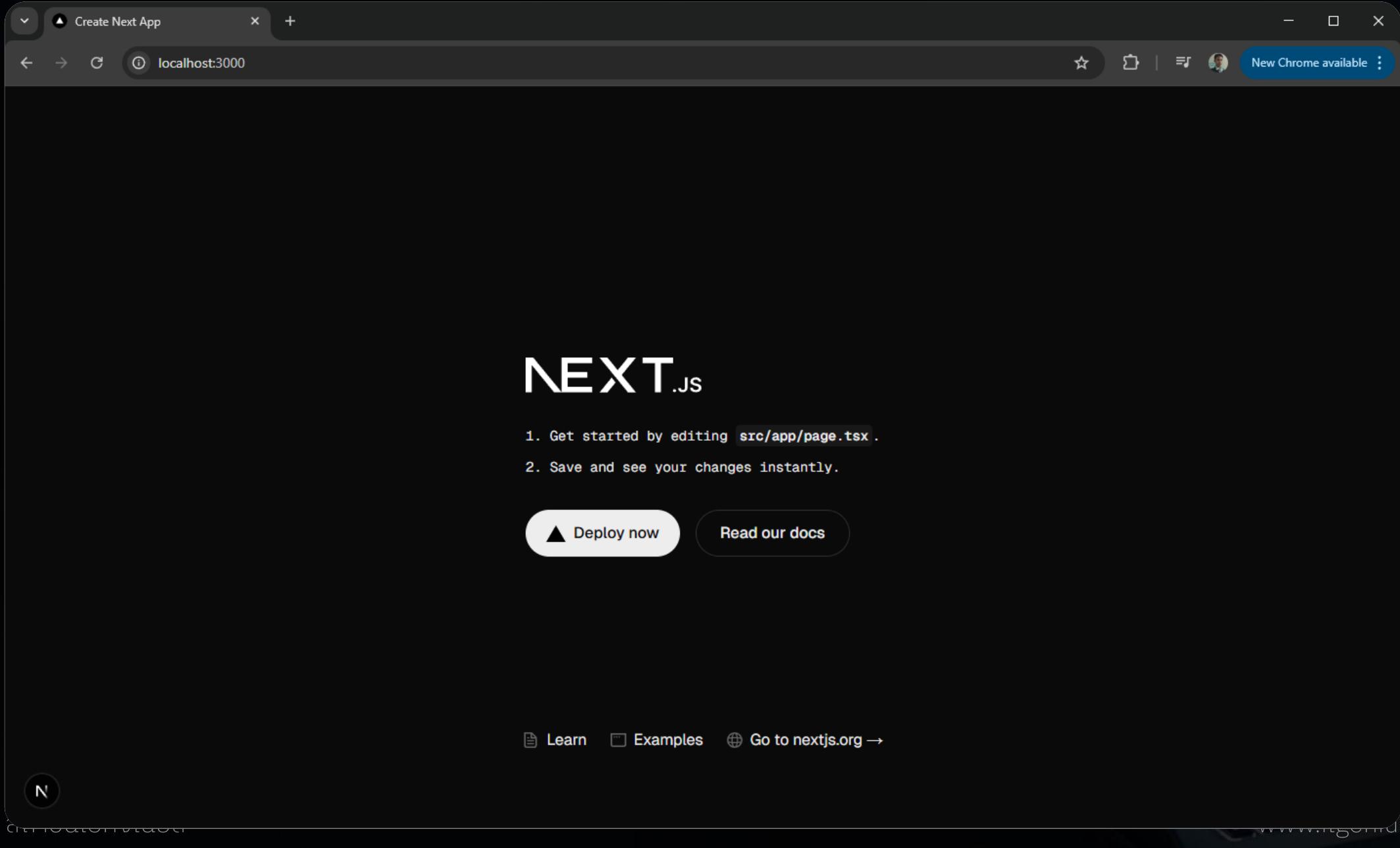
package.json > ...  
You, 1 minute ago | 1 author (You)

```
1  {
2      "name": "aichatbot-chatbot-nextjs",
3      "version": "0.1.0",
4      "private": true,
5      ▷ Debug
6      "scripts": {
7          "dev": "next dev",
8          "build": "next build",
9          "start": "next start",
10         "lint": "eslint"
11     },
12     "dependencies": {
13         "react": "19.1.0",
14         "react-dom": "19.1.0",
15         "next": "15.5.2"
16     },
17     "devDependencies": {
18         "typescript": "^5",
19         "@types/node": "^20",
20         "@types/react": "^19",
21         "@types/react-dom": "^19",
22         "@tailwindcss/postcss": "^4",
23         "tailwindcss": "^4",
24         "eslint": "^9",
25         "eslint-config-next": "15.5.2",
26         "@eslint/eslintrc": "^3"
27     }
28 }
```



สถาบันอาชญากรรม

www.itgenius.co.th



Create Next App

localhost:3000

Elements Console Sources Network Lighthouse >

5:39:42 PM - localhost:3000

http://localhost:3000/

# NEXT.js

1. Get started by editing `src/app/page.tsx`.

2. Save and see your changes instantly.

[Deploy now](#) [Read our docs](#)

Learn Examples Go to [nextjs.org](https://nextjs.org) →

Performance

Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)

▲ 0–49 ■ 50–89 ● 90–100



Search Postman Ctrl K

Invite Upgrade No environment

POST 04\_/api/chat\_04\_strear +

HTTP 02\_Langchain\_Basic / 04\_/api/chat\_04\_stream

Save Share Send

Params Authorization Headers (9) Body Scripts Settings Cookies Beautify

POST 04\_/api/chat\_04\_stream

Body raw

1 {  
2 "messages": [  
3 {  
4 "id": "chat-id-001",  
5 "role": "user",  
6 "parts": [  
7 {  
8 "type": "text",  
9 "text": "สวัสดีครับ บริษัทของเรารักษาความปลอดภัยมากที่สุดครับ"  
10 }  
11 ]  
12 }  
13 ]  
14 }

POST 01\_/api/chat\_05\_history?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9

GET 02\_/api/chat\_05\_history?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9 Copy

POST 01\_/api/chat\_06\_history\_optimize?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9

GET 02\_/api/chat\_06\_history\_optimize?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9 Copy

POST 01\_/api/chat\_07\_tool\_calling?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9

GET 02\_/api/chat\_07\_tool\_calling?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9 Copy

GET 01\_/api/document\_loader\_embedding\_pgvector/text\_csv

POST 02\_/api/document\_loader\_embedding\_pgvector/text\_csv

DEL 03\_/api/document\_loader\_embedding\_pgvector/text\_csv

PUT 04\_/api/document\_loader\_embedding\_pgvector/text\_csv

GET 05\_/api/document\_loader\_embedding\_pgvector/text\_csv\_pdf

POST 01\_/api/chat\_08\_rag?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9

GET 02\_/api/chat\_08\_rag?sessionId=4ebe30a5-0c01-4912-927d-78227102dee9 Copy

DjangoWebSocket

gofiber

QR Menu App API

Online Find and replace Console

Response Hist Click Send to get a response

Postbot Runner Start Proxy Cookies Vault Trash

แยก API end point ไว้ให้กดสอบง่าย





### 3. พื้นฐาน Langchain.js เชื่อมต่อกับ Gen AI

AI Chatbot with LangChain

localhost:3000

### AI Chatbot with LangChain.JS

สวัสดีครับ! มีอะไรให้ช่วยเหลือหรือคุยกันได้บ้างครับ?

ประเภทไทยแม่งออกเป็น 6 ภาค ได้แก่:

- \*\*ภาคเหนือ\*\*
- \*\*ภาคกลาง\*\*
- \*\*ภาคตะวันออก\*\*
- \*\*ภาคตะวันตก\*\*
- \*\*ภาคตะวันออกเฉียงเหนือ (อีสาน)\*\*
- \*\*ภาคใต้\*\*

แต่ละภาคมีลักษณะภูมิศาสตร์และวัฒนธรรมที่แตกต่างกันไปครับ ถ้าสนใจข้อมูลเพิ่มเติมเกี่ยวกับภาคไหนบอกได้เลยนะครับ!

ภาคเหนือมีซึ่งหน้าดูไวน้ำบ้างครับ

ภาคเหนือของประเทศไทยประกอบด้วย 9 จังหวัด ได้แก่:

- \*\*เชียงใหม่\*\*
- \*\*เชียงราย\*\*
- \*\*ลำปาง\*\*
- \*\*ลำพูน\*\*
- \*\*พะเยา\*\*
- \*\*แม่ฮ่องสอน\*\*
- \*\*น่าน\*\*
- \*\*อุตรดิตถ์\*\*
- \*\*แพร่\*\*

แต่ละจังหวัดมีความสวยงามทางธรรมชาติและวัฒนธรรมที่น่าสนใจมากมาย เลยครับ ถ้าต้องการข้อมูลเพิ่มเติมเกี่ยวกับจังหวัดใดจังหวัดหนึ่ง แจ้งได้เลย นะครับ!

พิมพ์ที่ข้อความที่... ▲

**ตัวอย่าง AI Chatbot  
อย่างรวดเร็วด้วย  
Langchain.js ร่วมกับ  
Next.js**



สถาบันไอทีเจเนียส

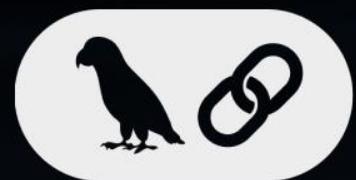
www.itgenius.co.th

• LIVE

อบรมออนไลน์



สร้าง AI Chatbots สำหรับองค์กร

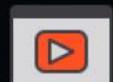


LangChain

ร่วมกับ Next.JS



และ supabase



มีวิดีโอบันทึกการอบรม  
ย้อนหลังให้ทุกวัน



สอนสดผ่าน Zoom  
รับจำนำเน็ตกัด

Chat AI วันที่ 2

Samit Koyom  
สถาบันไอทีจีเนียส



## Day 2

1. พื้นฐาน Langchain.js เชื่อมต่อกับ Gen AI
2. ระบบยืนยันตัวตนด้วย Supabase Auth

# 📁 โครงสร้างโปรเจกต์

```
my-langchain-chatbot/
├── src/
│   └── app/
│       ├── api/
│       │   ├── chat/
│       │   │   └── route.ts          # Chat API endpoint
│       │   ├── chat_01_start/
│       │   │   └── route.ts        # Step 1: Basic chat setup
│       │   ├── chat_02_request/
│       │   │   └── route.ts        # Step 2: Request handling
│       │   ├── chat_03_template/
│       │   │   └── route.ts        # Step 3: Prompt templates
│       │   ├── chat_04_stream/
│       │   │   └── route.ts        # Step 4: Streaming responses
│       │   └── test/
│           └── route.ts          # Test API endpoint
│       ├── globals.css          # Global styles
│       ├── layout.tsx           # Root layout
│       └── page.tsx              # Main chat interface
└── public/
    ├── eslint.config.mjs      # ESLint configuration
    ├── next.config.ts          # Next.js configuration
    ├── package.json             # Dependencies และ scripts
    ├── postcss.config.mjs      # PostCSS configuration
    ├── tailwind.config.ts      # Tailwind CSS configuration
    ├── tsconfig.json            # TypeScript configuration
    └── README.md                # Documentation
```



☰ ← → Home Workspaces API Network

Search Postman Ctrl K

+ Invite ⚙️ 🔔 Upgrade No environment

Samit

New Import

Collections

Environments

Flows

History

+

GET 01/\_api

HTTP AIChatbotLangchain / 01\_Next\_API / 01/\_api

GET {{baseURL}} /api

Send

Params Authorization Headers (7) Body Scripts Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

POST 01/\_api/test

PUT 02/\_api/test

DEL 03/\_api/test

GET 04/\_api/test?name=John

POST 05/\_api/test

PUT 06/\_api/test

DEL 07/\_api/test

POST 08/\_api/test

POST 09/\_api/chat

02\_Langchain\_Basic

POST 01/\_api/chat\_01\_start

POST 02/\_api/chat\_02\_request

POST 03/\_api/chat\_03\_template

POST 04/\_api/chat\_04\_stream

05\_Chat\_History

06\_Chat\_History\_Optimize

07\_Tool\_Calling

Document\_Loader\_EMBEDDING\_pgVector

08\_RAG

DjangoWebSocket

gofiber

QR Menu App API

vLLMSample

Response History

Click Send to get a response



Postbot Runner Start Proxy Cookies Vault Trash



API keys - OpenAI API

platform.openai.com/api-keys

ITGenius / Default project

Dashboard Docs API reference S

Create Chat Audio Images Assistants

Usage API keys Logs Storage Batches

Optimize Evaluations Fine-tuning

## API keys

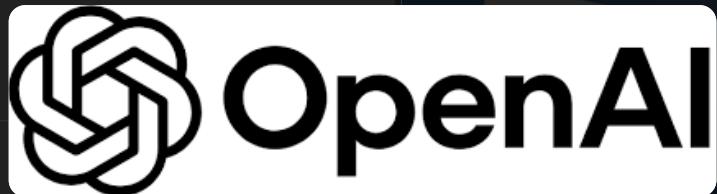
+ Create new secret key

You have permission to view and manage all API keys in this project.

Do not share your API key with others or expose it in the browser or other client-side code. To protect your account's security, OpenAI may automatically disable any API key that has leaked publicly.

View usage per API key on the [Usage page](#).

NAME	SECRET KEY	LAST USED	CREATED BY
chatbot-langchain-class	sk-...3CAA	Never	Samit Koyom
ai-chatbot-langchain	sk-...MJgA	Sep 8, 2025	Samit Koyom
n8n sample	sk-...blcA	Aug 6, 2025	Samit Koyom
n8n test	sk-...iiYA	Aug 2, 2025	Samit Koyom





```
package.json
```

```
"dependencies": {  
    "@langchain/core": "^0.3.75",  
    "@langchain/openai": "^0.6.11",  
    "langchain": "^0.3.33"  
}
```

```
.env
```

```
# === OPENAI (ChatGPT) ===  
OPENAI_API_KEY=your-openai-api-key  
OPENAI_MODEL_NAME="gpt-4o-mini"
```



```
route.ts
```

```
import { NextResponse } from "next/server"  
import { ChatOpenAI } from "@langchain/openai"  
  
export async function POST() {  
  
    // สร้าง instance ของ ChatOpenAI  
    const model = new ChatOpenAI({  
        model: process.env.OPENAI_MODEL_NAME || "gpt-4o-mini",  
        temperature: 0.7,  
        maxTokens: 300,  
    })  
  
    const input = `Translate "I love programming" into Thai.  
const response = await model.invoke(input)  
console.log(response) // ผลลัพธ์: ฉันรักการเขียนโปรแกรม  
  
    return NextResponse.json({ message: "Hello from Chat 01 - Start!" })  
}
```

Google AI Studio

API Keys

+ Create API key

Quickly test the Gemini API

API quickstart guide

```
curl "https://generativelanguage.googleapis.com/v1beta/models/gemini-2.0-flash:generateContent" \
-H 'Content-Type: application/json' \
-H 'X-goog-api-key: GEMINI_API_KEY' \
-X POST \
-d '{
  "contents": [
    {
      "parts": [
        {
          "text": "Explain how AI works in a few words"
        }
      ]
    }
  ]
}'
```

Your API keys are listed below. You can also view and manage your project and API keys in Google Cloud.

Look up API Key for project

Project number	Project name	API key	Created	Plan
...3547	n8n sample	...v_r4	Jul 30, 2025	Tier 1 Go to billing View usage data

Remember to use API keys securely. Don't share or embed them in public code. Use of Gemini API from a billing-enabled project is subject to pay-as-you-go pricing.

# Gemini



# Gemini



package.json

```
"dependencies": {  
  "@langchain/core": "^0.3.75",  
  "@langchain/google-genai": "^0.2.17",  
  "@langchain/openai": "^0.6.11",  
  "langchain": "^0.3.33",  
}
```



route.ts

```
import { NextResponse } from "next/server"  
import { ChatGoogleGenerativeAI } from "@langchain/google-genai"  
  
export async function POST() {  
  // สร้าง instance ของ GoogleGenerativeAI  
  const model = new ChatGoogleGenerativeAI({  
    model: process.env.GOOGLE_MODEL_NAME || "gemini-2.5-flash",  
    temperature: 0.7,  
    maxRetries: 2,  
    maxOutputTokens: 2048,  
  })  
  
  const input = `Translate "I love programming" into Thai.`  
  const response = await model.invoke(input)  
  console.log(response)  
  
  return NextResponse.json({ message: "Hello from Chat 01 - Start!" })  
}
```



.env

```
# === GOOGLE (Gemini) ===  
GOOGLE_API_KEY=your-google-api-key  
GOOGLE_MODEL_NAME="gemini-2.5-flash"
```



สถาบันไอทีจีเนียส

www.itgenius.co.th

ai.azure.com/resource/deployments/%2Fsubscriptions%2F9afdd173-1a1e-42d0-b157-3c57455b4b30%2FresourceGroups%2Fsampleapp%2Fproviders%2FMicrosoft.CognitiveServices%2Faccounts%2Fsamit-me4hlzb5-eastus2%2F...

Azure AI Foundry / samit-me4hlzb5-eastus2 / Deployments / gpt-5-mini-2

← gpt-5-mini-2

Overview Model catalog Playgrounds Build and customize Agents Templates Fine-tuning Observe and optimize Monitoring Protect and govern Azure OpenAI Evaluation Guardrails + controls Risks + alerts Governance Azure OpenAI Stored completions Batch jobs Assistant vector stores Data files My assets Models + endpoints

Details Metrics

Open in playground Request quota Edit Delete

Endpoint

Target URI  
https://samit-me4hlzb5-eastus2.cognitiveservices.azure.com/openai/r...

Key  
.....

Deployment info

Name gpt-5-mini-2	Provisioning state Succeeded
Deployment type Global Standard	Created on 2025-09-09T09:35:38.393495Z
Created by samit90daytalk@hotmail.com	Modified on Sep 9, 2025 4:35 PM
Modified by samit90daytalk@hotmail.com	Version upgrade policy Once a new default version is available
Rate limit (Tokens per minute) 100,000	Rate limit (Requests per minute) 100
Model name gpt-5-mini	Model version 2025-08-07
Life cycle status GenerallyAvailable	Date created Aug 7, 2025 7:00 AM

Language: Javascript | SDK: OpenAI SDK | Authentication type: Key Authentication

Get Started

Below are example code snippets for a few use cases. For additional information about Azure OpenAI SDK, see the documentation.

### 1. Authentication using API Key

For OpenAI API Endpoints, deploy the Model to generate the endpoint URL and an API key to authenticate a request. The endpoint and key are strings holding the endpoint URL and the API Key.

The API endpoint URL and API key can be found on the Deployments + Endpoint page once the model is deployed.

To create a client with the OpenAI SDK using an API key, initialize the client by passing your API key to the SDK's configuration. This allows you to authenticate and interact with OpenAI's services seamlessly:

```
const api_key = "<your-api-key>";
const endpoint = "https://samit-me4hlzb5-eastus2.cognitiveservices.azure.com/openai/v1/";
const modelName = "gpt-5-mini";
const deployment_name = "gpt-5-mini-2";

const client = new OpenAI({
  baseURL: endpoint,
  apiKey: api_key
});
```

### 2. Install dependencies



Azure AI Studio



สถาบันไอทีเจเนียส

www.itgenius.co.th



Azure AI Studio

package.json

```
"dependencies": {  
    "@langchain/core": "^0.3.75",  
    "@langchain/openai": "^0.6.11",  
    "langchain": "^0.3.33"  
}
```

.env

```
# === MS AZURE ===  
AZURE_OPENAI_API_KEY=your-azure-openai-api-key  
AZURE_OPENAI_API_INSTANCE_NAME=your-azure-openai-instance-name  
AZURE_OPENAI_API_DEPLOYMENT_NAME=gpt-5-mini-2  
AZURE_OPENAI_API_VERSION=2024-04-01-preview  
AZURE_OPENAI_API_MODEL_NAME="gpt-5-mini"
```

route.ts

```
import { NextResponse } from "next/server"  
import { AzureChatOpenAI } from "@langchain/openai"  
  
export async function POST() {  
    // สร้าง instance ของ AzureChatOpenAI  
    const model = new AzureChatOpenAI({  
        model: process.env.AZURE_OPENAI_API_MODEL_NAME || "gpt-5-mini",  
        maxTokens: 1024,  
        maxRetries: 2,  
        azureOpenAIapiKey: process.env.AZURE_OPENAI_API_KEY,  
        azureOpenAIapiInstanceName: process.env.AZURE_OPENAI_API_INSTANCE_NAME,  
        azureOpenAIapiDeploymentName: process.env.AZURE_OPENAI_API_DEPLOYMENT_NAME,  
        azureOpenAIapiVersion: process.env.AZURE_OPENAI_API_VERSION,  
    })  
  
    const input = `Translate "I love programming" into Thai.`  
    const response = await model.invoke(input)  
    console.log(response)  
  
    return NextResponse.json({ message: "Hello from Chat 01 - Start!" })  
}
```



สถาบันไอทีจีเนียส

www.itgenius.co.th

openrouter.ai/settings/keys

OpenRouter Search /

Models Chat Rankings Docs 

## Settings

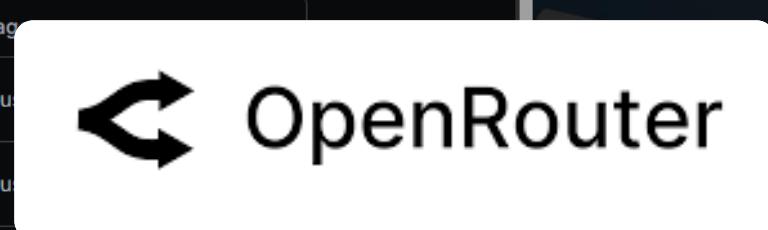
Account Credits Presets API Keys Provisioning Keys Integrations (BYOK) Training, Logging, & Privacy Organization Members

### API Keys

Create API Key

Manage your API keys to access all models from OpenRouter ⓘ

Key	Limit	Usage
ai-chatbot-langchain sk-or-v1-b55...2f2	Unlimited	\$0 used
n8n-sampe-api-key sk-or-v1-c0a...19f	Unlimited	\$0 used
n8n-app-key sk-or-v1-35c...7b1	Unlimited	\$0 used
n8n-sample sk-or-v1-3b3...8d9	Unlimited	\$0 used



The OpenRouter logo features a white rounded rectangle containing a black double-headed arrow icon pointing left and right, followed by the text "OpenRouter" in a bold, sans-serif font.



# OpenRouter



package.json

```
"dependencies": {  
    "@langchain/core": "^0.3.75",  
    "@langchain/openai": "^0.6.11",  
    "langchain": "^0.3.33"  
}
```



.env

```
# === OPENROUTER ===  
OPENROUTER_API_BASE="https://openrouter.ai/api/v1"  
OPENROUTER_API_KEY="your-openrouter-api-key"  
OPENROUTER_MODEL_NAME="qwen/qwen3-235b-a22b-2507"  
# OPENROUTER_MODEL_NAME="qwen/qwen3-8b:free"
```



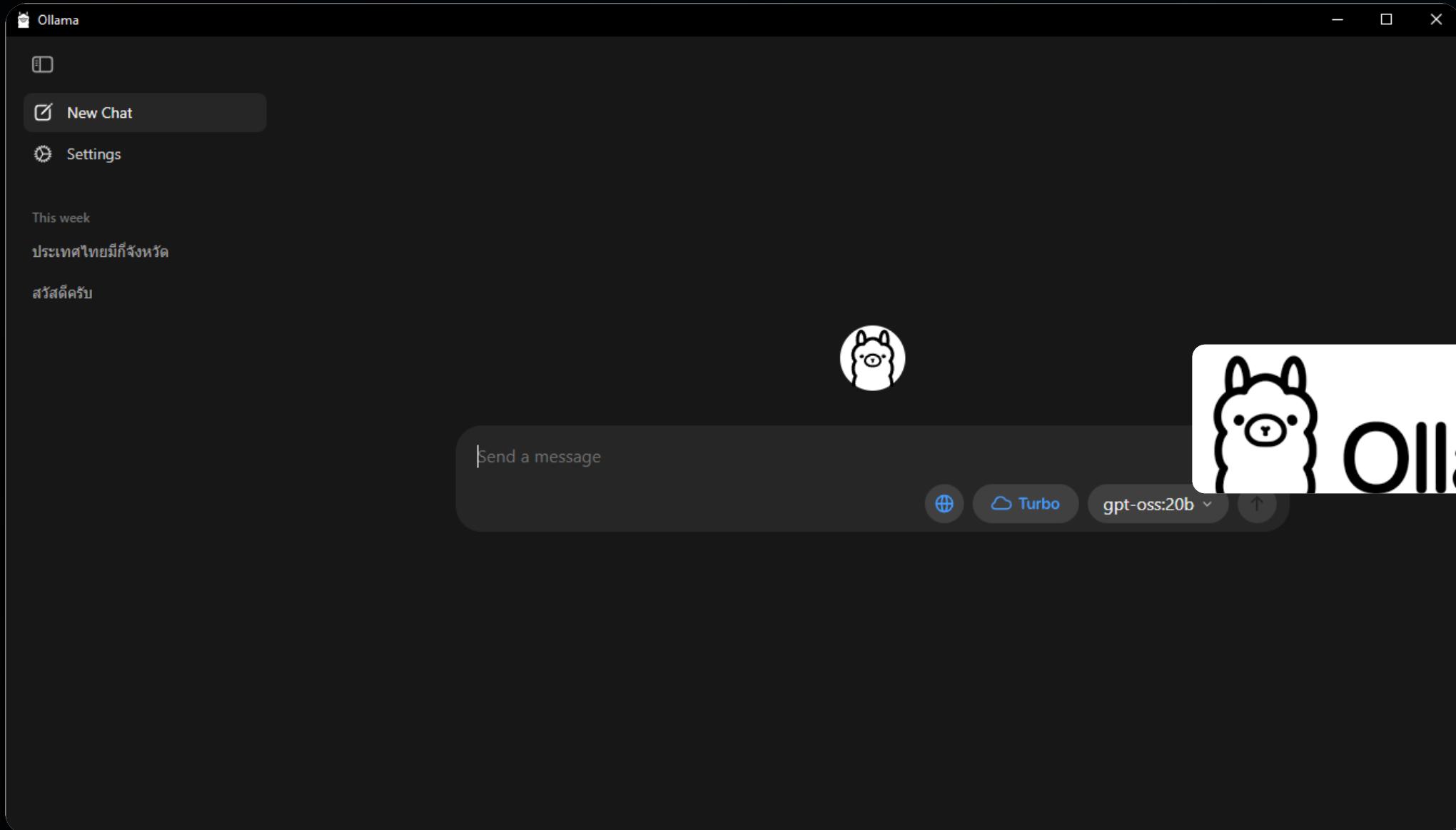
สถาบันไอทีจีเนียส



route.ts

```
import { NextResponse } from "next/server"  
import { ChatOpenAI } from "@langchain/openai"  
  
export async function POST() {  
    // สร้าง instance ของ ChatOpenAI (OpenRouter)  
    const model = new ChatOpenAI({  
        apiKey: process.env.OPENROUTER_API_KEY,  
        model: process.env.OPENROUTER_MODEL_NAME || "qwen/qwen3-235b-a22b-2507",  
        cache: false,  
        temperature: 0.7,  
        maxTokens: 300,  
        configuration: {  
            baseURL: process.env.OPENROUTER_API_BASE,  
        },  
        streamUsage: false  
    })  
  
    const input = `Translate "I love programming" into Thai.`  
    const response = await model.invoke(input)  
    console.log(response)  
  
    return NextResponse.json({ message: "Hello from Chat 01 - Start!" })  
}
```

www.itgenius.co.th



สถาบันไอทีเจเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)



package.json

```
"dependencies": {  
    "@langchain/core": "^0.3.75",  
    "@langchain/openai": "^0.6.11",  
    "langchain": "^0.3.33"  
}
```



.env

```
# === OLLAMA (Local) ===  
OLLAMA_API_BASE="http://localhost:11434/v1"  
OLLAMA_MODEL_NAME="gemma:2b"  
# OLLAMA_MODEL_NAME="gpt-oss:20b"  
# OLLAMA_MODEL_NAME="qwen3:8b"
```



สถาบันไอทีจีเนียส



route.ts

```
import { NextResponse } from "next/server"  
import { ChatOpenAI } from "@langchain/openai"  
  
export async function POST() {  
    // สร้าง instance ของ Ollama (Local)  
    const model = new ChatOpenAI({  
        model: process.env.OLLAMA_MODEL_NAME || "gemma:2b",  
        temperature: 0.7,  
        maxTokens: 1000,  
        configuration: {  
            baseURL: process.env.OLLAMA_API_BASE || "http://localhost:11434/v1",  
        },  
        apiKey: "ollama", // Ollama ไม่ต้องการ API key จริง แต่ต้องใส่ค่าอัลเรกีด์  
    })  
  
    const input = `Translate "I love programming" into Thai.`  
    const response = await model.invoke(input)  
    console.log(response)  
  
    return NextResponse.json({ message: "Hello from Chat 01 - Start!" })  
}
```



www.itgenius.co.th

```
(env) root@ubuntu-gpu-4000adax:/home/vllm_project# python -m vllm.entrypoints.openai.api server --model "Qwen/Qwen2.5-7B-Instruct"
INFO 09-08 16:20:51 [__init__.py:241] Automatically detected platform cuda.
(APIServer pid=9806) INFO 09-08 16:20:54 [api_server.py:1805] vLLM API server version 0.10.1.1
(APIServer pid=9806) INFO 09-08 16:20:54 [utils.py:326] non-default args: {'model': 'Qwen/Qwen2.5-7B-Instruct'}
(APIServer pid=9806) INFO 09-08 16:21:06 [__init__.py:711] Resolved architecture: Qwen2ForCausalLM
(APIServer pid=9806) `torch_dtype` is deprecated! Use `dtype` instead!
(APIServer pid=9806) INFO 09-08 16:21:07 [__init__.py:1750] Using max model len 32768
(APIServer pid=9806) INFO 09-08 16:21:12 [scheduler.py:222] Chunked prefill is enabled with max_num_batched_tokens=2048.
INFO 09-08 16:21:17 [__init__.py:241] Automatically detected platform cuda.
(EngineCore_0 pid=9886) INFO 09-08 16:21:21 [core.py:636] Waiting for init message from front-end.
(EngineCore_0 pid=9886) INFO 09-08 16:21:21 [core.py:74] Initializing a V1 LLM engine (v0.10.1.1) with config: model='Qwen/Qwen2.5-7B-Instruct', speculative_config=None, tokenizer='Qwen/Qwen2.5-7B-Instruct', skip_tokenizer_init=False, tokenizer_mode=auto, revision=None, override_neuron_config={}, tokenizer_revision=None, trust_remote_code=False, dtype=torch.bfloat16, max_seq_len=32768, download_dir=None, load_format=auto, tensor_parallel_size=1, pipeline_parallel_size=1, disable_custom_all_reduce=False, quantization=None, enforce_eager=False, kv_cache_dtype=auto, device_config=cuda, decoding_config=DecodingConfig(backend='auto', disable_fallback=False, disable_any whitespace=False, disable_additional_properties=False, reasoning_backend=''), observability_config=ObservabilityConfig(metrics_for_version=None, otlp_traces_endpoint=None, collect_detailed_traces=None), seed=0, served_model_name=None, enable_prefix_caching=True, chunked_prefill_enabled=True, use_async_output_proc=True, pooler_config=None, debug_dump_path=":3", cache_dir="", backend="", custom_ops=[], splitting_ops=[{"vllm.unified_attention": "with_output", "vllm.mamba_mixer2": ""}], use_inductor=true, compile_sizes=[], inductor_compile_config={"enable": true}, inductor_passes={}, cudagraph_mode:1, use_cudagraph=true, cudagraph_num_of_warmups:1, cudagraph_copy_inputs=false, full_cuda_graph=false, pass_config={}, max_capture_size:512, local_cache_dir:null}
(EngineCore_0 pid=9886) INFO 09-08 16:21:23 [parallel_state.py:1134] rank 0 in world size 1 is assigned as DP rank 0, PP rank 0, TP rank 0, EP rank 0
(EngineCore_0 pid=9886) WARNING 09-08 16:21:23 [topk_topp_sampler.py:61] FlashInfer is not available. Falling back to the PyTorch-native implementation of top-p & top-k sampling. For the best performance, please install FlashInfer.
(EngineCore_0 pid=9886) INFO 09-08 16:21:23 [gpu_model_runner.py:1953] Starting to load model Qwen/Qwen2.5-7B-Instruct...
(EngineCore_0 pid=9886) INFO 09-08 16:21:23 [gpu_model_runner.py:1985] Loading model from scratch...
(EngineCore_0 pid=9886) INFO 09-08 16:21:23 [cuda.py:328] Using Flash Attention backend on V1 engine.
(EngineCore_0 pid=9886) INFO 09-08 16:21:24 [weight_utils.py:296] Using model weights format ['*.safetensors']
Loading safetensors checkpoint shards:  0% Completed | 0/4 [00:00<?, ?it/s]
Loading safetensors checkpoint shards: 25% Completed | 1/4 [00:00<00:01, 2.10it/s]
Loading safetensors checkpoint shards: 50% Completed | 2/4 [00:01<00:01, 1.98it/s]
Loading safetensors checkpoint shards: 75% Completed | 3/4 [00:01<00:00, 2.02it/s]
Loading safetensors checkpoint shards: 100% Completed | 4/4 [00:02<00:00, 1.98it/s]
Loading safetensors checkpoint shards: 100% Completed | 4/4 [00:02<00:00, 1.99it/s]
```





```
package.json  
"dependencies": {  
    "@langchain/core": "^0.3.75",  
    "@langchain/openai": "^0.6.11",  
    "langchain": "^0.3.33"  
}
```



```
.env  
# === vLLM (Local) ===  
VLLM_API_BASE="http://localhost:8000/v1"  
VLLM_MODEL_NAME="meta-llama/llama-3.3-70b-instruct"
```



```
route.ts  
import { NextResponse } from "next/server"  
import { ChatOpenAI } from "@langchain/openai"  
  
export async function POST() {  
    // สร้าง instance ของ vLLM (Local) - ใช้ ChatOpenAI กับ baseURL ของ vLLM  
    const model = new ChatOpenAI({  
        model: process.env.VLLM_MODEL_NAME || "meta-llama/llama-3.3-70b-instruct",  
        temperature: 0.7,  
        maxTokens: 1000,  
        configuration: {  
            baseURL: process.env.VLLM_API_BASE || "http://localhost:8000/v1",  
        },  
        apiKey: "vllm", // vLLM ไม่ต้องการ API key จะริง แต่ต้องใส่ค่าอะไหล่ได้  
    })  
  
    const input = `Translate "I love programming" into Thai.`  
    const response = await model.invoke(input)  
    console.log(response)  
  
    return NextResponse.json({ message: "Hello from Chat 01 - Start!" })  
}
```



[Open incident](#)Networking in BLR1 and SGP1 Region To learn more, [check our status page](#).

X

PROJECTS

MANAGE

App Platform

Agent Platform New

Droplets

GPU Droplets New

Functions

Kubernetes

Volumes Block Storage

Databases

Spaces Object Storage

Container Registry

Backups & Snapshots

Networking

Monitoring

SaaS Add-Ons

By DigitalOcean

Billing

Support

Settings

API

 Search by resource name or public IP (Ctrl+B)[Create](#)My Team  
Estimated costs: \$11.11

# Gradient™ AI Platform

[Actions](#)[Get Started](#)[Agent workspaces](#)[Serverless inference](#)[Knowledge bases](#)[Guardrails](#)[Model provider keys](#)

## Model Endpoints

[About endpoint](#)

With a model endpoint you can use models from industry-leading providers without hosting or maintaining them yourself with token-based billing.

[Try and compare models in Model Playground](#)

### Usage

#### CHAT COMPLETIONS

Chat completions create a model response for the chat conversation supplied in the request. See a list of [supported models](#).

#### MODEL LIST

Returns a list of available models with their corresponding id values, which you use as the model parameter in your inference requests.

[Supported parameters](#)

## Model Access Keys

Grants you access to any model hosted by DigitalOcean through a model endpoint.

[Create model access key](#)

Name	Created at	...
SamitSampleKey	03:54 09/04/2025	...

### CHAT COMPLETIONS

Gradient SDK ▾ OpenAI GPT-oss-1...

```
import os
from gradient import Gradient

inference_client = Gradient(
    inference_key=os.environ.get(
        "MODEL_ACCESS_KEY"
    ),
)

inference_response = inference_client.chat.completions.create(
    messages=[
        {
            "role": "user",
            "content": "What is the capital of France?"
        }
    ],
    model="openai-gpt-oss-120b",
)

print(inference_response.choices[0].message.content)
```

[Copy](#)

### MODEL LIST

Gradient SDK ▾

```
from gradient import Gradient
```



สถาบันไอทีจีเนียส

www.itgenius.co.th

cloud.digitalocean.com/gen-ai/workspaces/11f08753-190b-a8ab-b074-4e013e2ddde4/agents/19f72c3b-8753-11f0-b074-4e013e2ddde4/settings?i=7c32b7

Open incident Networking in BLR1 and SGP1 Region To learn more, [check our status page](#).

Search by resource name or public IP (Ctrl+B)

Create ? ⚙️ My Team Estimated costs: \$11.11

PROJECTS

MANAGE

App Platform

Agent Platform **New**

Droplets

GPU Droplets **New**

Functions

Kubernetes

Volumes Block Storage

Databases

Spaces Object Storage

Container Registry

Backups & Snapshots

Networking

Monitoring

SaaS Add-Ons

By DigitalOcean

Billing

Support

Settings

API

← Back to agent-gpt-oss

## agent-gpt-oss

in [Playground](https://yhyp7q6tpuazlindrjg5ggxi.agents.do-ai.run)

Overview Playground Observability Activity Evaluations Resources **Settings**

Agent Info **agent-gpt-oss** **Edit**  
Playground

Workspace **agent-gpt-oss**

Region **Toronto • TOR1**

Tags Your agent currently has no tags. **Edit**

Agent Instructions คุณเชื่อ "Genius AI" เป็นผู้ช่วยตอบคำถามเรื่องทั่วไป แม่นๆ ตอบที่ชัดเจน และตอบเป็นภาษาไทยเมื่อคุณใช้คำสั่งเป็นไทย **Edit**

Retrieval Rules Retrieval Method: none  
Include citations? No  
K Value: 10 **Edit**

Endpoint Access Keys Use Agent Access Keys to allow outside applications access to your agent endpoint without setting your endpoint to public availability. **Create Key**

Name Created at

gradient



สถาบันไอทีจีเนียส

www.itgenius.co.th



```
● ● ●  
"dependencies": {  
  "@gradientai/nodejs-sdk": "^1.12.1",  
  "@langchain/community": "^0.3.55",  
  "@langchain/core": "^0.3.75",  
  "langchain": "^0.3.33",  
}
```

package.json

```
● ● ●
```

.env

```
# === Gradient AI (DigitalOcean) ===  
GRADIENT_API_BASE="https://api.gradient.ai/api/v1"  
GRADIENT_ACCESS_TOKEN=your-gradient-access-token  
GRADIENT_WORKSPACE_ID=agent-gpt-oss  
GRADIENT_MODEL_NAME="openai-gpt-oss-120b"
```

```
● ● ●  
import { NextResponse } from "next/server"  
import { Gradient } from "@gradientai/nodejs-sdk"  
import { GradientLLM } from "@langchain/community/llms/gradient_ai"  
  
export async function POST() {  
  // สร้าง instance ของ GradientLLM  
  const model = new GradientLLM({  
    gradientAccessKey: process.env.GRADIENT_ACCESS_TOKEN || "",  
    workspaceId: process.env.GRADIENT_WORKSPACE_ID || "agent-gpt-oss",  
    modelSlug: process.env.GRADIENT_MODEL_NAME || "openai-gpt-oss-120b",  
    inferenceParameters: {  
      maxGeneratedTokenCount: 2048,  
      temperature: 0.7,  
    },  
    gradientApiUrl: "https://apis.gradient.network/api/v1",  
  })  
  
  const input = `Translate "I love programming" into Thai.`  
  const response = await model.invoke(input)  
  console.log(response)  
  
  return NextResponse.json({ message: "Hello from Chat 01 - Start!" })  
}
```



สถาบันไอทีเนยส์

vvvvvvv.ngenys.com

# Chat Start with Langchain



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)

```
api/chat_01_start/route.ts

import { NextResponse } from "next/server"
import { ChatOpenAI } from "@langchain/openai"

export async function POST() {

    // สร้าง instance ของ ChatOpenAI (Model ChatGPT) - DigitalOcean
    const model = new ChatOpenAI({
        model: process.env.GPT_AGENT_MODEL_NAME, // ชื่อโมเดลที่ต้องการใช้
        apiKey: process.env.GPT_AGENT_API_KEY,
        temperature: 0.7,
        maxTokens: 300, // จำนวนคำตอบสูงสุดที่ต้องการ 300 token
        configuration: {
            baseURL: process.env.GPT_AGENT_API_BASE,
        },
    })

    // กำหนดข้อความที่ต้องการแปล
    const input = `Translate "I love programming" into Thai.`

    // Model จะทำการแปลข้อความ
    const response = await model.invoke(input)

    try {
        const response = await model.invoke([
            {
                role: "system",
                content:
                    "คุณเป็นจัดการฝ่ายการเงินของบริษัท คุณตอบคำถามให้พนักงานในบริษัทในเรื่องการเงิน",
            },
            {
                role: "human", // "human" เป็น alias ของ "user"
                content: "สวัสดีครับ งบประมาณปีนี้เป็นอย่างไรบ้าง?",
            },
        ])
    }

    const meta = response.response_metadata || {}
    const usedModel = meta.model || meta.model_name || "unknown"

    // ส่งกลับทั้งคำตอบและชื่อโมเดล (จะได้เห็นข้อว่า "ตอบจากโมเดลอะไร")
    return NextResponse.json({
        content: response.content,
        usedModel,
    })
}

} catch (error) {
    // Handle error
    console.error("Error:", error)
    return NextResponse.json({ error: "An error occurred" })
}
```



POST 01/\_api/chat\_01\_start

HTTP AIChatbotLangchain / 02\_Langchain\_Basic / 01/\_api/chat\_01\_start

POST {{baseUrl}} /api/chat\_01\_start

Params Authorization Headers (16) Body Scripts Settings Cookies

Query Params

	Key	Value	Description	Bulk Edit
	Key	Value	Description	

Body Cookies Headers (6) Test Results 200 OK • 4.01 s • 1.1 KB • Save Response

{ } JSON ▾ Preview Visualize

```
1 {  
2   "content": "สวัสดีครับ งบประมาณปีนี้เรามีการจัดสรรงบประมาณตามแผนกลยุทธ์ของบริษัท โดยเน้นการลงทุนในด้านการพัฒนาผลิตภัณฑ์ใหม่ การตลาด และการฝึกอบรมพนักงาน เพื่อเพิ่มประสิทธิภาพในการทำงานและขยายตลาด ก槐ากคุณต้องการรายละเอียดเพิ่มเติมเกี่ยวกับงบประมาณในแต่ละแผนกหรือโครงการ สามารถสอบถามได้เลยคับ",  
3   "usedModel": "gpt-4o-mini-2024-07-18"  
4 }
```



# Chat from Request



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)

```
api/chat_02_request/route.ts
import {NextRequest, NextResponse} from "next/server"
import { ChatOpenAI } from "@langchain/openai"

export async function POST(req: NextRequest) {

    // สร้างตัวแปรรับข้อมูลจาก client
    const body = await req.json()

    // ดึงข้อความจาก body
    const message: [] = body.message ?? []

    // สร้าง instance ของ ChatOpenAI (Model ChatGPT)
    const model = new ChatOpenAI({
        model: "gpt-4o-mini",
        temperature: 0.7,
        maxTokens: 300,
    })

    try {
        const response = await model.invoke(message)

        // ดึงชื่อโมเดลจริงจาก metadata (บาง provider ใช้ model หรือ model_name)
        const meta = response.response_metadata || {}
        const usedModel = meta.model || meta.model_name || "unknown"

        // ส่งกลับทั้งค่าตอบและชื่อโมเดล (จะได้เห็นชัดว่า "ตอบจากโมเดลอะไร")
        return NextResponse.json({
            content: response.content,
            usedModel,
        })
    } catch (error) {
        // Handle error
        console.error("Error:", error)
        return NextResponse.json({ error: "An error occurred" })
    }
}
```



POST 02./api/chat\_02\_request

HTTP AIChatbotLangchain / 02\_Langchain\_Basic / 02./api/chat\_02\_request

POST {{baseURL}} /api/chat\_02\_request

Params Authorization Headers (9) Body Scripts Settings Cookies Schema Beautify

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```

1 {
2   "message": [
3     {
4       "role": "system",
5       "content": "คุณเป็นจัดการฝ่ายการเงินของบริษัท คุณตอบคำถามให้พนักงานในบริษัทในเรื่องการเงิน"
6     },
7     {
8       "role": "user",
9       "content": "แล้วถ้าเป็นงบ R&D คิดว่าควรสักเท่าไหร่"
10    }
11  ]
12 }

```

Body Cookies Headers (6) Test Results

200 OK • 8.07 s • 2.33 KB • Save Response

JSON Preview Visualize

```

1 {
2   "content": "การกำหนดงบประมาณสำหรับการวิจัยและพัฒนา (R&D) ขึ้นอยู่กับหลายปัจจัย เช่น ขนาดของบริษัท, อุดหนุนรอมที่ดำเนินการ, เป้าหมายทางธุรกิจ, และกลยุทธ์การเติบโตของบริษัท/ภาคโดยทั่วไปแล้ว บริษัทที่มุ่งเน้นนักกรรมและภารกิจเพื่อความมั่นคงจะจัดสรรงบประมาณ R&D ในระดับที่สูงกว่า บริษัทที่เน้นการดำเนินงานที่มีอยู่แล้ว/ก1. **อุดหนุนรอมเทคโนโลยี**: อาจใช้ประมาณ 10-20% ของรายได้/ก2. **อุดหนุนรอมเกลือซ้อม**: อาจใช้ประมาณ 15-25% ของรายได้/ก3. **อุดหนุนรอมที่น้ำ**: อาจใช้ประมาณ 5-10% ของรายได้/ก4. ความต้องการผลิตภัณฑ์และบริการที่คาดว่าจะได้รับจากการลงทุนใน R&D ด้วย นอกเหนือจากนี้ยังควรคำนึงถึงความสามารถในการแข่งขันของบริษัทและผลตอบแทนที่คาดว่าจะได้รับจากการลงทุนใน R&D ด้วย ที่แตกต่างกันด้วย/ก/หากคุณมีข้อมูลเพิ่มเติมเกี่ยวกับบริษัทหรืออุดหนุนรอมที่คุณท่องานอยู่ ฉันสามารถให้คำแนะนำที่เฉพาะเจาะจง",
3   "usedModel": "gpt-4o-mini-2024-07-18"
4 }

```

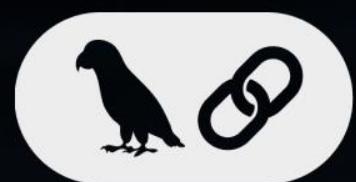


• LIVE

อบรมออนไลน์



## สร้าง AI Chatbots สำหรับองค์กร



# LangChain

ร่วมกับ **Next.JS**



และ

# supabase



มีวิดีโอบันทึกการอบรม  
ย้อนหลังให้ทุกวัน



สอนสดผ่าน Zoom  
รับจำนำเน็ตกัด

**Samit Koyom**  
สถาบันไอทีจีเนียส



## Day 3

1. พื้นฐาน Langchain.js เชื่อมต่อกับ Gen AI (ต่อ)
2. ระบบยืนยันตัวตนด้วย Supabase Auth

# Chat Prompt Template



## api/chat\_03\_template/route.ts

```
import {NextRequest, NextResponse} from "next/server"
import { ChatOpenAI } from "@langchain/openai"
import { ChatPromptTemplate } from "@langchain/core/prompts"
import { StringOutputParser } from "@langchain/core/output_parsers"

export async function POST(req: NextRequest) {

    // สร้างตัวแปรรับข้อมูลจาก client
    const body = await req.json()

    // ดึงข้อความจาก body - กำหนด type ให้ชัดเจน
    const messages: Array<{ role: string; content: string }> = body.message ?? []

    // กำหนดตัวแปร prompt template
    const prompt = ChatPromptTemplate.fromMessages([
        ['system', 'คุณเป็นจัดการฝ่ายการเงินของบริษัท คุณตอบคำถามให้พนักงานในบริษัทในเรื่องการเงิน'],
        ['user', '{question}']
    ])

    // สร้าง instance ของ ChatOpenAI (Model ChatGPT)
    const model = new ChatOpenAI({
        model: "gpt-4o-mini",
        temperature: 0.7,
        maxTokens: 300,
    })

    // สร้าง chain (prompt + model + output parser)
    const chain = prompt.pipe(model).pipe(new StringOutputParser())

    try {
        const response = await chain.invoke({
            question: messages[0].content ?? ""
        })

        return NextResponse.json({
            content: response,
        })
    } catch (error) {
        // Handle error
        console.error("Error:", error)
        return NextResponse.json({ error: "An error occurred" })
    }
}
```

```
import {NextRequest, NextResponse} from "next/server"
import { ChatOpenAI } from "@langchain/openai"
import { ChatPromptTemplate } from "@langchain/core/prompts"
import { StringOutputParser } from "@langchain/core/output_parsers"

export async function POST(req: NextRequest) {

    // สร้างตัวแปรรับข้อมูลจาก client
    const body = await req.json()

    // ดึงข้อความจาก body - กำหนด type ให้ชัดเจน
    const messages: Array<{ role: string; content: string }> = body.message ?? []

    // กำหนดตัวแปร prompt template
    const prompt = ChatPromptTemplate.fromMessages([
        ['system', 'คุณเป็นจัดการฝ่ายการเงินของบริษัท คุณตอบคำถามให้พนักงานในบริษัทในเรื่องการเงิน'],
        ['user', '{question}']
    ])
}
```



```

import {NextRequest, NextResponse} from "next/server"
import { ChatOpenAI } from "@langchain/openai"
import { ChatPromptTemplate } from "@langchain/core/prompts"
import { StringOutputParser } from "@langchain/core/output_parsers"

export async function POST(req: NextRequest) {

    // สร้างตัวแปลงข้อมูลจาก client
    const body = await req.json()

    // ดึงข้อความจาก body - กำหนด type ให้ชัดเจน
    const messages: Array<{ role: string; content: string }> = body.message ?? []

    // กำหนดตัวแปร prompt template
    const prompt = ChatPromptTemplate.fromMessages([
        ['system', 'คุณเป็นจัดการฝ่ายการเงินของบริษัท คุยกับค่าาคนให้พัฒนาในบริษัทในเรื่องการเงิน'],
        ['user', '{question}']
    ])

    // สร้าง instance ของ ChatOpenAI (Model ChatGPT)
    const model = new ChatOpenAI({
        model: "gpt-4o-mini",
        temperature: 0.7,
        maxTokens: 300,
    })

    // สร้าง chain (prompt + model + output parser)
    const chain = prompt.pipe(model).pipe(new StringOutputParser())

    try {
        const response = await chain.invoke({
            question: messages[0].content ?? ""
        })

        return NextResponse.json({
            content: response,
        })
    } catch (error) {
        // Handle error
        console.error("Error:", error)
        return NextResponse.json({ error: "An error occurred" })
    }
}

```

```

// สร้าง instance ของ ChatOpenAI (Model ChatGPT)
const model = new ChatOpenAI({
    model: "gpt-4o-mini",
    temperature: 0.7,
    maxTokens: 300,
})

// การสร้าง chain (prompt + model + output parser)
const chain = prompt.pipe(model).pipe(new StringOutputParser())

try {
    const response = await chain.invoke({
        question: messages[0].content ?? ""
    })

    return NextResponse.json({
        content: response,
    })
}

} catch (error) {
    // Handle error
    console.error("Error:", error)
    return NextResponse.json({ error: "An error occurred" })
}

```



New Import POST 03\_/api/chat\_03\_template + No environment

+ Search collections

AIChatbotLangchain

- 01\_Next\_API
- 02\_Langchain\_Basic
  - POST 01\_/api/chat\_01\_start
  - POST 02\_/api/chat\_02\_request
  - POST 03\_/api/chat\_03\_template
  - POST 04\_/api/chat\_04\_stream
- 05\_Chat\_History
- 06\_Chat\_History\_Optimize
- 07\_Tool\_Calling
- Document\_Loader\_EMBEDDING\_pg...
- 08\_RAG

DjangoWebSocket

gofiber

QR Menu App API

vLLMSample

HTTP AIChatbotLangchain / 02\_Langchain\_Basic / 03\_/api/chat\_03\_template Save Share ↗

POST {{baseURL}} /api/chat\_03\_template

Send

Params Authorization Headers (9) Body Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

```
1 {  
2   "message": [  
3     {  
4       "role": "user",  
5       "content": "สวัสดีครับ บริษัทเรามีงานด้านการวิจัย R & D หรือไม่ครับ"  
6     }  
7   ]  
8 }
```

Body Cookies Headers (6) Test Results 200 OK 12.55 s 1.13 KB Save Response

{} JSON Preview Visualize

```
1 {  
2   "content": "สวัสดีครับ บริษัทของเรามีงานประมวลผลเรื่องการวิจัยและพัฒนา (R&D)  
เพื่อสนับสนุนการสร้างสรรค์นวัตกรรมและพัฒนาผลิตภัณฑ์ใหม่ๆ ครับ  
หากคุณต้องการรายละเอียดเพิ่มเติมเกี่ยวกับงานประมวลผลหรือแผนการใช้จ่ายในด้านนี้  
สามารถติดต่อฝ่ายการเงินหรือฝ่ายวิจัยและพัฒนาได้ครับ  
โดยรายนี้จะให้ข้อมูลเพิ่มเติมตามที่คุณต้องการครับ"  
3 }
```

# Chat with Streaming



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)



```

import { NextRequest } from "next/server"
import { ChatOpenAI } from "@langchain/openai"
import { ChatPromptTemplate } from "@langchain/core/prompts"
import { toUIMessageStream } from "@ai-sdk/langchain"
import { createUIMessageStreamResponse, UIMessage, convertToModelMessages } from "ai"

// กำหนดให้ API ที่ทำงานแบบ Edge Runtime เพื่อประยุกต์กิจภาพที่ดีกว่า
export const runtime = "edge"

// กำหนดเวลาสูงสุดที่ API จะทำงานได้ ( เช่น 30 วินาที )
export const maxDuration = 30 // วินาที

export async function POST(req: NextRequest) {
  try {
    // ตั้งชื่อความลับ request body ที่ส่งมาจาก useChat hook
    const { messages }: { messages: UIMessage[] } = await req.json()

    // สร้าง Prompt Template เพื่อกำหนดแนวทางและรูปแบบการตอบของ AI
    const prompt = ChatPromptTemplate.fromMessages([
      ["system", "You are a helpful and friendly AI assistant."],
      // แปลง UIMessage ให้เป็นแบบที่ LangChain นิยม
      ...convertToModelMessages(messages),
    ])

    // เลือกรุ่นของโมเดล OpenAI ที่ต้องการใช้
    const model = new ChatOpenAI({
      model: "gpt-4o-mini", // ระบุรุ่น AI model ที่ใช้
      temperature: 0.7, // ค่านี้ส่งตรงคือค่าอ่อน (0 = เป็นระบบมาก, 1 = สร้างสรรค์มาก)
      maxTokens: 300, // จำนวน token สูงสุดที่สามารถตอบได้
      streaming: true, // เปิดใช้ streaming response
    })

    // สร้าง Chain โดยการเชื่อมต่อ Prompt กับ Model เข้าด้วยกัน
    const chain = prompt.pipe(model)

    // เชิญໃร่าน Chain พร้อมกับส่ง message ต่อต่อไป และรับผลลัพธ์แบบ stream
    const stream = await chain.stream({
      // LangChain ต้องการตัวแปลงเป็นภาษาไทยใน input ส่วนหัว prompt ที่ส่งจาก message history
    })

    // สร้าง Response กลับไปให้ Frontend
    const response = createUIMessageStreamResponse({
      stream: toUIMessageStream(stream),
    })

    return response
  } catch (error) {
    // จัด理 error และ log รายละเอียดเพิ่ม debug
    console.error("API Error:", error)
    // สร้าง error response กลับไปยัง client
    return new Response(
      JSON.stringify({
        error: "An error occurred while processing your request",
      }),
      {
        status: 500,
        headers: { "Content-Type": "application/json" }
      }
    )
  }
}

```

## package.json

```

"dependencies": {
  "@ai-sdk/langchain": "^1.0.23",
  "@ai-sdk/react": "^2.0.23",
  "ai": "^5.0.23",
  "@langchain/core": "^0.3.72",
  "@langchain/openai": "^0.6.9",
  "langchain": "^0.3.31"
}

```

## .env

```

# === OPENAI (ChatGPT) ===
OPENAI_API_KEY=your-openai-api-key
OPENAI_MODEL_NAME="gpt-4o-mini"

```



## api/chat\_04\_stream/route.ts

```
import { NextRequest } from "next/server"
import { ChatOpenAI } from "@langchain/openai"
import { ChatPromptTemplate } from "@langchain/core/prompts"
import { toUIMessageStream } from "@ai-sdk/langchain"
import { createUIMessageStreamResponse, UIMessage, convertToModelMessages } from "ai"

// กำหนดให้ API นี้ทำงานแบบ Edge Runtime เพื่อประสิทธิภาพที่ดีกว่า
export const runtime = "edge"

// กำหนดเวลาสูงสุดที่ API จะทำงานได้ ( เช่น 30 วินาที )
export const maxDuration = 30 // วินาที

export async function POST(req: NextRequest) {
  try {
    // ดึงข้อมูลจาก request body ที่ส่งมาจากการใช้ useChat hook
    const { messages }: { messages: UIMessage[] } = await req.json()

    // สร้าง Prompt Template เพื่อกำหนดบทบาทและรูปแบบการตอบของ AI
    const prompt = ChatPromptTemplate.fromMessages([
      ["system", "You are a helpful and friendly AI assistant."],
      // แปลง UIMessage ให้เป็นรูปแบบที่ LangChain เข้าใจ
      ...convertToModelMessages(messages),
    ])

    // เลือกรุ่นของโมเดล OpenAI ที่ต้องการใช้
    const model = new ChatOpenAI({
      model: "gpt-4o-mini", // ระบุรุ่น AI model ที่ใช้
      temperature: 0.7, // ความสร้างสรรค์ของคำตอบ (0 = เป็นระบบมาก, 1 = สร้างสรรค์มาก)
      maxTokens: 300, // จำนวน token สูงสุดที่สามารถตอบได้
      streaming: true, // เปิดใช้ streaming response
    })

    // สร้าง Chain โดยนำ Prompt กับ Model เข้าไปยัง
    const chain = prompt.pipe(model)

    // รี;y ใช้งาน Chain พร้อมกับ message ล่าสุดไป และรับผลลัพธ์แบบ stream
    const stream = await chain.stream({
      // LangChain ต้องการล่วงประเพณีๆ ใน input สำหรับ prompt ที่ส่งจาก message history
    })

    // ส่ง Response กลับไปยัง Frontend
    const response = createUIMessageStreamResponse({
      stream: toUIMessageStream(stream),
    })

    return response
  } catch (error) {
    // จัดการ error และ log ข้อความเพื่อ debug
    console.error("API Error:", error)
    // แปลง error response กลับไปยัง client
    return new Response(
      JSON.stringify({
        error: "An error occurred while processing your request",
      }),
      {
        status: 500,
        headers: { "Content-Type": "application/json" }
      }
    )
  }
}
```

```
import { NextRequest } from "next/server"
import { ChatOpenAI } from "@langchain/openai"
import { ChatPromptTemplate } from "@langchain/core/prompts"
import { toUIMessageStream } from "@ai-sdk/langchain"
import { createUIMessageStreamResponse, UIMessage, convertToModelMessages } from "ai"

// กำหนดให้ API นี้ทำงานแบบ Edge Runtime เพื่อประสิทธิภาพที่ดีกว่า
export const runtime = "edge"

// กำหนดเวลาสูงสุดที่ API จะทำงานได้ ( เช่น 30 วินาที )
export const maxDuration = 30 // วินาที

export async function POST(req: NextRequest) {
  try {
    // ดึงข้อมูลจาก request body ที่ส่งมาจากการใช้ useChat hook
    const { messages }: { messages: UIMessage[] } = await req.json()

    // สร้าง Prompt Template เพื่อกำหนดบทบาทและรูปแบบการตอบของ AI
    const prompt = ChatPromptTemplate.fromMessages([
      ["system", "You are a helpful and friendly AI assistant."],
      // แปลง UIMessage ให้เป็นรูปแบบที่ LangChain เข้าใจ
      ...convertToModelMessages(messages),
    ])

    // เลือกรุ่นของโมเดล OpenAI ที่ต้องการใช้
    const model = new ChatOpenAI({
      model: "gpt-4o-mini", // ระบุรุ่น AI model ที่ใช้
      temperature: 0.7, // ความสร้างสรรค์ของคำตอบ (0 = เป็นระบบมาก, 1 = สร้างสรรค์มาก)
      maxTokens: 300, // จำนวน token สูงสุดที่สามารถตอบได้
      streaming: true, // เปิดใช้ streaming response
    })
  }
}
```

## api/chat\_04\_stream/route.ts

```
●●●  
import { NextRequest } from "next/server"  
import { ChatOpenAI } from "@langchain/openai"  
import { ChatPromptTemplate } from "@langchain/core/prompts"  
import { toUIMessageStream } from "@ai-sdk/langchain"  
import { createUIMessageStreamResponse, UIMessage, convertToModelMessages } from "ai"  
  
// กำหนดไฟ API ที่ใช้งานแบบ Edge Runtime เพื่อประสิทธิภาพที่ดีกว่า  
export const runtime = "edge"  
  
// กำหนดเวลาสูงสุดที่ API จะทำงานได้ ( เช่น 30 วินาที )  
export const maxDuration = 30 // วินาที  
  
export async function POST(req: NextRequest) {  
  try {  
    // ตั้งชื่อค่าผ่าน request body ที่ส่งมาจาก useChat hook  
    const { messages }: { messages: UIMessage[] } = await req.json()  
  
    // สร้าง Prompt Template เพื่อกำหนดภาษาและรูปแบบการตอบของ AI  
    const prompt = ChatPromptTemplate.fromMessages([
      ["system", "You are a helpful and friendly AI assistant."],  
      // แปลง UIMessage ให้เข้ากับ LangChain เนื่องจาก  
      ... convertToModelMessages(messages),  
    ])  
  
    // เลือกชื่อรุ่นของ OpenAI ที่ต้องการใช้  
    const model = new ChatopenAI({  
      model: "gpt-4o-mini", // ระบุรุ่น AI model ที่ใช้  
      temperature: 0.7, // ความสร้างสรรค์ของคำตอบ ( 0 = เป็นระบบมาก, 1 = สร้างสรรค์มาก )  
      maxTokens: 300, // จำนวน token สูงสุดที่สามารถตอบได้  
      streaming: true, // เปิดไฟฟ์ streaming response  
    })  
  
    // สร้าง Chain โดยการเชื่อมต่อ Prompt กับ Model เนื้อต่อเนื่อง  
    const chain = prompt.pipe(model)  
  
    // เรียกใช้งาน Chain พร้อมกับส่ง message ล่าสุดไป และรับผลลัพธ์แบบ stream  
    const stream = await chain.stream({  
      // LangChain ต้องการตัวแปรเปล่าๆ ใน input สำหรับ prompt ที่สร้างจาก message history  
    })  
  
    // ส่ง Response กลับไปให้ Frontend  
    const response = createUIMessageStreamResponse({  
      stream: toUIMessageStream(stream),  
    })  
  
    return response  
  } catch (error) {  
    // จัดการ error และ log ข้อความเพื่อ debug  
    console.error("API Error:", error)  
    // ส่ง error response กลับไปยัง client  
    return new Response(  
      JSON.stringify({  
        error: "An error occurred while processing your request",  
      }),  
      {  
        status: 500,  
        headers: { "Content-Type": "application/json" }  
      }  
    )  
  }  
}
```

```
// สร้าง Chain โดยการเชื่อมต่อ Prompt กับ Model เนื้อต่อเนื่องกัน  
const chain = prompt.pipe(model)  
  
// เรียกใช้งาน Chain พร้อมกับส่ง message ล่าสุดไป และรับผลลัพธ์แบบ stream  
const stream = await chain.stream({  
  // LangChain ต้องการตัวแปรเปล่าๆ ใน input สำหรับ prompt ที่สร้างจาก message history  
})  
  
// ส่ง Response กลับไปให้ Frontend  
const response = createUIMessageStreamResponse({  
  stream: toUIMessageStream(stream),  
})  
  
return response  
} catch (error) {  
  // จัดการ error และ log ข้อความเพื่อ debug  
  console.error("API Error:", error)  
  // ส่ง error response กลับไปยัง client  
  return new Response(  
    JSON.stringify({  
      error: "An error occurred while processing your request",  
    }),  
    {  
      status: 500,  
      headers: { "Content-Type": "application/json" }  
    }  
  )  
}
```





# Supabase Authentication



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)

# Create Supabase Project



[supabase.com/dashboard/new/abfyguddkixdgmcedkus](https://supabase.com/dashboard/new/abfyguddkixdgmcedkus)

iamsamitdev's Org / New project

**Create a new project**

Your project will have its own dedicated instance and full Postgres database.  
An API will be set up so you can easily interact with your new database.

**Organization** iamsamitdev's Org Free

**Project name** ai-chatbot-langchain-nextjs

**Database password**   Copy  
Note: If using the Postgres connection string, you will need to [percent-encode](#) the password

This password is strong. [Generate a password](#).

**Region** Southeast Asia (Singapore)  
Select the region closest to your users for the best performance.

[SECURITY OPTIONS >](#)

[ADVANCED CONFIGURATION >](#)

[Cancel](#) [Create new project](#)



สถาบัน

genius.co.th

```
.env
```

```
# Environment variables for the AI Chatbot application
NODE_ENV=development

# Supabase config
NEXT_PUBLIC_SUPABASE_URL=your-supabase-url-here
NEXT_PUBLIC_SUPABASE_PUBLISHABLE_OR_ANON_KEY=your-supabase-publishable-or-anon-key-here
```



# Setup Shadcn/UI



The screenshot shows a dark-themed web browser window displaying the Next.js documentation. The URL in the address bar is <https://ui.shadcn.com/docs/installation/next>. The page has a navigation bar at the top with links to Docs, Components, Blocks, Charts, Themes, and Colors. A search bar is on the right.

**Sections:**

- Get Started
- Components
- Registry
- MCP Server
- Get Started
- Installation
- components.json
- Theming
- Dark Mode
- CLI
- Monorepo
- Open in v0
- JavaScript
- Blocks
- Figma
- Changelog
- Legacy Docs

**Next.js**

Install and configure shadcn/ui for Next.js.

**Create project**

Run the `init` command to create a new Next.js project or to setup an existing one:

```
pnpm npm yarn bun  
npx shadcn@latest init
```

Copy to Clipboard

Choose between a Next.js project or a Monorepo.

**Add Components**

You can now start adding components to your project.

```
pnpm npm yarn bun  
npx shadcn@latest add button
```

# Setup Supabase UI Library



The screenshot shows a dark-themed web browser window displaying the Supabase UI Library documentation. The URL in the address bar is <https://supabase.com/ui/docs/nextjs/password-based-auth>. The left sidebar contains navigation links for 'GETTING STARTED' (Introduction, Quick Start, FAQ), 'BLOCKS' (Client), and 'Password-Based Auth' (Social Auth, Dropzone, Realtime Cursor, Current User Avatar, Realtime Avatar Stack, Realtime Chat, Infinite Query Hook). The 'Password-Based Auth' section is currently selected. Below the sidebar, there's a search bar labeled 'Search UI Library...'. The main content area has a title 'Installation' and a code block showing the command to add the UI library via npm: 

```
$ npx shadcn@latest add https://supabase.com/ui/r/password-based-auth-nextjs.json
```

. A 'Copy' button is next to the command. Below this is a 'Folder structure' diagram and a code snippet for a 'route.ts' file.

## Installation

```
npm pnpm yarn bun
```

```
$ npx shadcn@latest add https://supabase.com/ui/r/password-based-auth-nextjs.json
```

Open in

## Folder structure

This block includes the **Supabase client**. If you already have one installed, you can skip overwriting it.

```
1 import { createClient } from '@lib/supabase/server'  
2 import { type EmailOtpType } from '@supabase/supabase-js'  
3 import { redirect } from 'next/navigation'  
4 import { type NextRequest } from 'next/server'  
5  
6 export async function GET(request: NextRequest) {  
7   const { searchParams } = new URL(request.url)  
8   const token_hash = searchParams.get('token_hash')  
9   const type = searchParams.get('type') as EmailOtpType | null  
10  const _next = searchParams.get('next')  
11  const next = _next?.startsWith('/') ? _next : '/'  
12  
13  if (token_hash && type) {  
14    const supabase = await createClient()  
15  
16    const { error } = await supabase.auth.verifyOtp({
```

# รายการ Library กี่เพิ่มเข้ามา

```
dependencies": {  
    "@ai-sdk/langchain": "^1.0.23",  
    "@ai-sdk/react": "^2.0.23",  
    "@langchain/core": "^0.3.72",  
    "@langchain/openai": "^0.6.9",  
    "@radix-ui/react-label": "^2.1.7",  
    "@radix-ui/react-slot": "^1.2.3",  
    "@supabase/ssr": "^0.7.0",  
    "@supabase/supabase-js": "^2.56.0",  
    "ai": "^5.0.23",  
    "class-variance-authority": "^0.7.1",  
    "clsx": "^2.1.1",  
    "langchain": "^0.3.31",  
    "lucide-react": "^0.541.0",  
    "next": "15.5.0",  
    "react": "19.1.0",  
    "react-dom": "19.1.0",  
    "tailwind-merge": "^3.3.1"  
}
```



# โครงสร้างโปรเจกต์ที่ได้



The image shows a screenshot of a code editor (VS Code) displaying a project structure and a package.json file. The project structure on the left includes folders for public, src (containing app, auth, chat, components, lib, .env, .env\_example, .env.production, .gitignore, components.json, eslint.config.mjs, and next-env.d.ts), and a package.json file. The package.json file content is as follows:

```
package.json
{
  "scripts": {
    "next": "next build",
    "start": "next start",
    "lint": "eslint"
  },
  "dependencies": {
    "@ai-sdk/langchain": "^1.0.23",
    "@ai-sdk/react": "^2.0.23",
    "@langchain/core": "^0.3.72",
    "@langchain/openai": "^0.6.9",
    "@radix-ui/react-label": "^2.1.7",
    "@radix-ui/react-slot": "^1.2.3",
    "@supabase/ssr": "^0.7.0",
    "@supabase/supabase-js": "^2.56.0",
    "ai": "^5.0.23",
    "class-variance-authority": "^0.7.1",
    "clsx": "^2.1.1",
    "langchain": "^0.3.31",
    "lucide-react": "^0.541.0",
    "next": "15.5.0",
    "react": "19.1.0",
    "react-dom": "19.1.0",
    "tailwind-merge": "^3.3.1"
  },
  "devDependencies": {
    "@eslint/eslintrc": "^3",
    "@tailwindcss/postcss": "^4",
    "@types/node": "^20",
    "@types/react": "^19",
    "@types/react-dom": "^19",
    "eslint": "^9",
    "eslint-config-next": "15.5.0",
    "tailwindcss": "^4",
    "tw-animate-css": "^1.3.7",
    "typescript": "^5"
  }
}
```

Samit Koyom, 2 weeks ago • Initial commit from





สถาบันไอทีจีนียส

localhost:3000

AI Genius

เข้าสู่ระบบ ลงทะเบียน

ปั๊บเก็บข้อมูล AI, RAG, และ Document Loader

# Genius AI Chatbot

สุดยอดแพลตฟอร์มชั้นนำที่รวมพลัง AI, RAG (Retrieval-Augmented Generation), Document Loader & Vector Embeddings และ Tool Calling คืนมาสู่ชุมชนอาชีวศึกษา, ฐานข้อมูล, และองค์กรในการให้คำแนะนำและแก้ไขปัญหาในเชิงลึก LangChain.js, Next.js, Supabase และ OpenAI GPT-4o-mini เพื่อประสิทธิภาพที่สูงสุด ปลอดภัย และรวดเร็ว

เข้าสู่ระบบ | เรียนรู้เพิ่มเติม

**RAG & Document Search**  
ความสามารถในการค้นหาข้อมูลจากเอกสาร, ฐานข้อมูล  
และ AI agent  
Vector Search + Structured Data

**Tool Calling & Smart Query**  
ค้นหาข้อมูลสืบค่า, ย้อนกลับ, และซึ่งกันและกัน  
จากฐานข้อมูล Supabase  
ของรัฐบาลไทย-ธุรกิจ, Partial Matching

**Security & Modern UI**  
มีเครื่องมือ เช่น Supabase Auth, RLS และ UI  
สวยงามที่สุด  
Responsive, TypeScript, Edge Runtime

**Chat History System**  
บันทึกและจัดการประวัติการสนับสนุนแบบ session  
Auto-title, Real-time, สนับสนุนภาษา

**Advanced Memory Management**  
มีเครื่องมือ เช่น Context Window, Token Counting,  
Smart Summarization

**Modern UI & Responsive Design**  
UI สวยงาม รับรองทุกอุปกรณ์  
shadcn/ui, Tailwind CSS, Mobile Friendly

จุดเด่นที่ผู้ใช้ไว้วางใจ

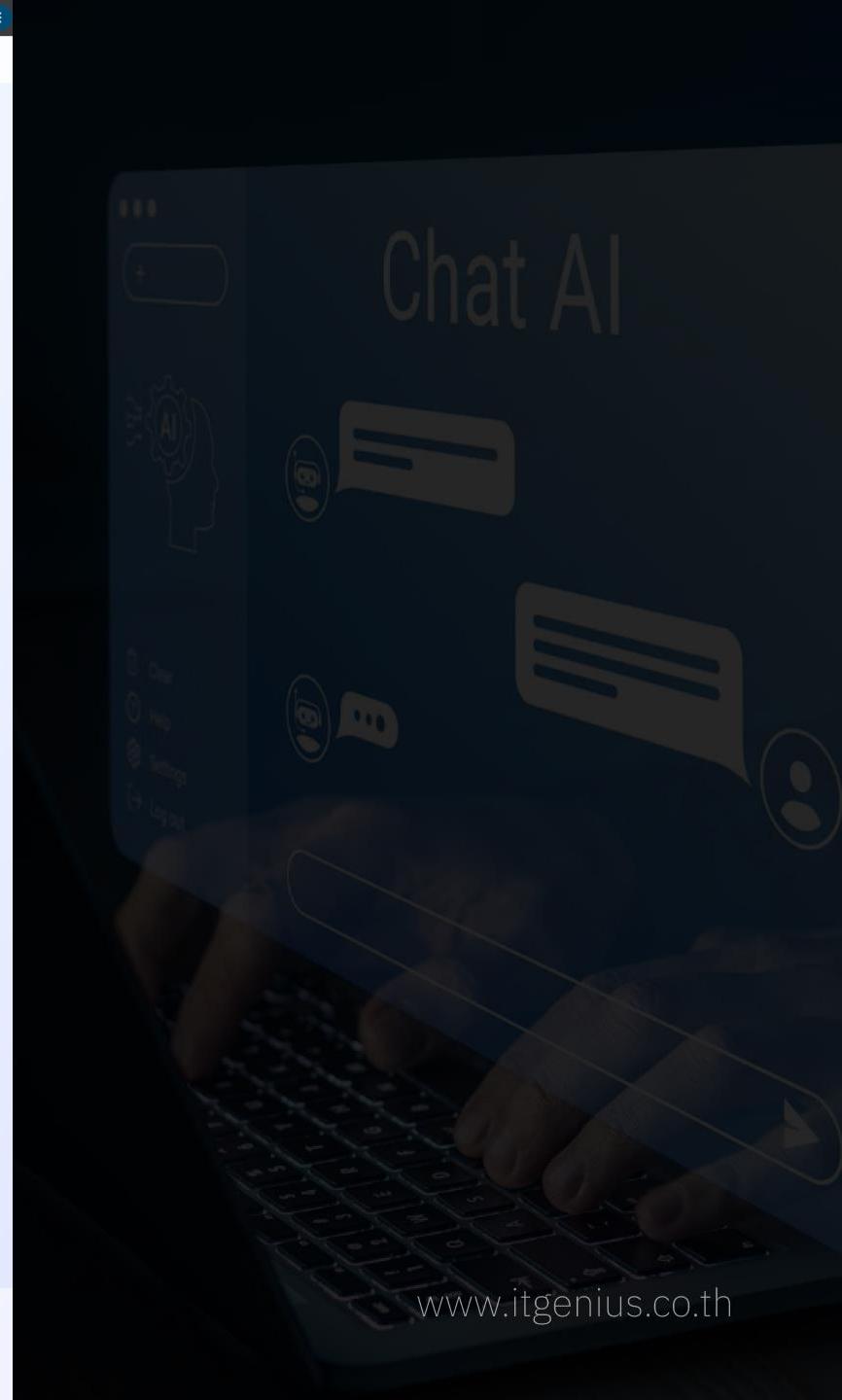
**10K+** มีผู้ใช้งานจริง

**99.9%** Uptime & Reliability

**5+** ระบบจัดการ: (RAG, Tool Calling, Document Loader, Security, UI)

พร้อมให้บริการ 24/7 | รองรับภาษาไทย-อังกฤษ | ปลอดภัยและทันสมัย

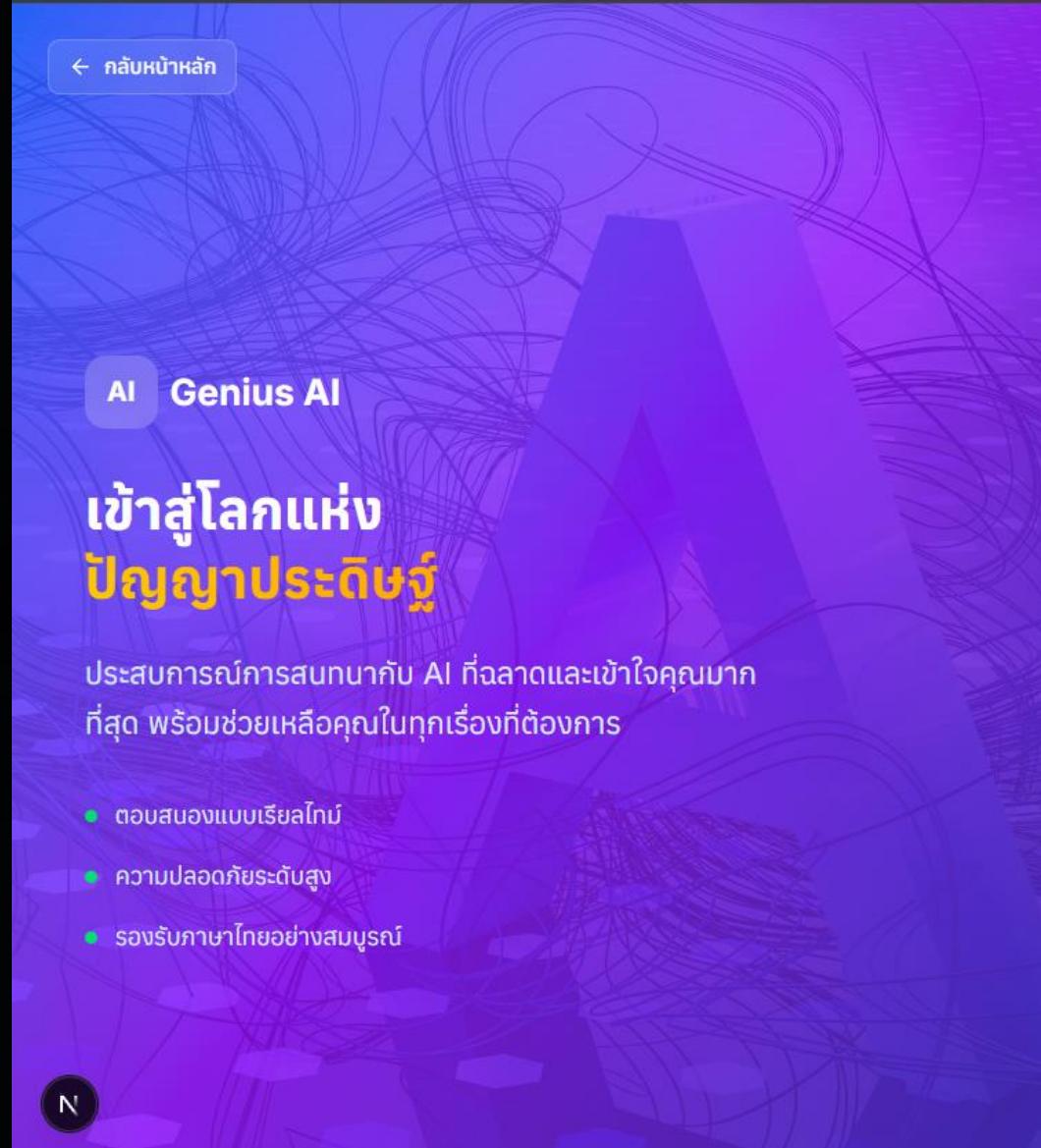
AI Genius AI Chatbot  
© 2025 Genius AI Chatbot. สร้างโดย ❤️ และ AI



www.itgenius.co.th

The image shows a web browser window with the following details:

- Title Bar:** "AI Chatbot with LangChain" and "localhost:3000/auth/login".
- Landing Page (Left Side):**
  - A blue header bar with a white "← กลับหน้าหลัก" button.
  - A white rounded rectangle containing the text "AI Genius AI".
  - A large yellow title "เข้าสู่โลกแห่งปัญญาประดิษฐ์".
  - A subtitle: "ประสบการณ์การสนทนากับ AI ที่จัดลาดและเข้าใจคุณมาก ที่สุด พร้อมช่วยเหลือคุณในทุกเรื่องที่ต้องการ".
  - A bulleted list:
    - ต่อสนองแบบเรียลไทม์
    - ความปลอดภัยระดับสูง
    - รองรับภาษาไทยอย่างสมบูรณ์
  - A small circular icon with the letters "ITC" and a "N" in the bottom right corner.- Login Form (Right Side):**
  - A white rounded rectangle with the heading "Login".
  - Text: "Enter your email below to login to your account".
  - An "Email" input field containing "me@email.com".
  - An "Password" input field containing "\*\*\*\*\*".
  - A "Forgot your password?" link.
  - A large black "Login" button.
  - A "Don't have an account? Sign up" link.



## Sign up

Create a new account

Display Name

Phone

Email

Password

Repeat Password

**Sign up**

Already have an account? [Login](#)

The screenshot shows a web browser window with two pages visible due to a transparency effect.

**Background Page (Left):**

- Header:** AI Chatbot with LangChain
- URL:** localhost:3000/auth/forgot-password
- Text:** กลับหน้าหลัก (Back to Home)
- Section:** Genius AI
- Title:** เข้าสู่โลกแห่งปัญญาประดิษฐ์
- Text:** ประสบการณ์การสนทนากับ AI ที่ฉลาดและเข้าใจคุณมากที่สุด พร้อมช่วยเหลือคุณในทุกเรื่องที่ต้องการ
- List:**
  - ตอบสนองแบบเรียลไทม์
  - ความปลอดภัยระดับสูง
  - รองรับภาษาไทยอย่างสบายน้ำ
- Logo:** N
- Page Footer:** สถาบันเทคโนโลยี  
www.ngenius.co.th

**Foreground Page (Right):**

- Title:** Reset Your Password
- Text:** Type in your email and we'll send you a link to reset your password
- Form:** Email input field containing m@example.com
- Button:** Send reset email
- Text:** Already have an account? [Login](#)



AI Chatbot with LangChain

localhost:3000/auth/forgot-password

← กลับหน้าหลัก

AI Genius AI

# เข้าสู่โลกแห่งปัญญาประดิษฐ์

ประสบการณ์การสนทนากับ AI ที่ฉลาดและเข้าใจคุณมากที่สุด พร้อมช่วยเหลือคุณในทุกเรื่องที่ต้องการ

- ตอบสนองแบบเรียลไทม์
- ความปลอดภัยระดับสูง
- รองรับภาษาไทยอย่างสมบูรณ์

N

สามารถติดตามเราได้ที่

Check Your Email

Password reset instructions sent

If you registered using your email and password, you will receive a password reset email.

www.ngenius.co.th

AI Chatbot with LangChain

localhost:3000/chat

# Genius AI Chatbot

สวัสดีค่ะ, Samit

Logout

สวัสดีครับ! มีอะไรให้ช่วยเหลือหรือสอบถามได้เลยนะครับ

คุณชื่ออะไร

ผมไม่มีชื่อเฉพาะ แต่คุณสามารถเรียกผมว่า "AI" หรือ "ผู้ช่วย" ก็ได้ครับ มีอะไรให้ช่วยไหมครับ?

งั้นตั้งชื่อให้ว่า อันดุล หลังกัน

ได้เลยครับ! คุณสามารถเรียกผมว่า "อันดุล" ได้เลย ถ้ามีอะไรอยากถามหรือ พูดคุยกับ tôiมาได้เลยนะครับ!

พิมพ์ข้อความที่นี่...



ร่วมกับ

และ

LangChain  
Next.JS   
supabase 

# Thank you



ว.สาเมตร โภยม  
สถาบันไอทีจีเนียส



สถาบันไอทีจีเนียส

[www.itgenius.co.th](http://www.itgenius.co.th)