วารสารแม่โจ้เทคโนโลยีสารสนเทศและนวัตกรรม ปีที่ 12 ฉบับที่ 3 กันยายน – ธันวาคม 2569

Received: 28 ก.พ. 2569 **Revised:** 7 พ.ค. 2569 **Accepted:** 11 พ.ค. 2569

การออกแบบเวิร์กโฟลว์เอ็นเอทเอ็นร่วมกับเอไอเอเจนท์ สำหรับระบบถาม-ตอบงานขายอัตโนมัติ

Design of n8n Workflow with AI Agents for Automated Sales Question-Answering

ธนพนธ์ อินท์ขะ[1], อลงกต กองมณี[1], สมนึก สินธุปวน[1] และ พาสน์ ปราโมกข์ชน[1]*

[1]สาขาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้

Thanapon Inkha[1], Alongkot Gongmanee[1], Somnuek Sinthupuan[1] and Part Pramokchon[1]*

[1]Computer Science Department, Faculty of Science, Maejo University

*Corresponding author: part@mju.ac.th

Abstract

The objectives of this research were to develop and design an automated workflow using n8n combined with an AI Agent for a sales Q&A system, and to evaluate system efficiency and user satisfaction. The developed system applies Large Language Model (LLM) technology working with a Knowledge Base through the Retrieval-Augmented Generation (RAG) process. n8n is utilized as a workflow orchestrator to connect the LINE application with the AI processing unit. The results indicated that the n8n workflow design effectively enabled the system to transmit messages and retrieve product information to answer customer inquiries accurately, rapidly, and naturally, replacing repetitive human tasks. The user satisfaction assessment showed a high level of satisfaction, demonstrating that the system can be effectively applied to real-world sales operations.

Keywords: *n8n; AI Agent; Automated Q&A System; LINE Chatbot*

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาและออกแบบเวิร์กโฟลว์อัตโนมัติด้วยแพลตฟอร์ม n8n ร่วมกับเอไอเอเจนท์ (AI Agent) สำหรับระบบถาม-ตอบงานขาย และประเมินประสิทธิภาพและความพึงพอใจของผู้ใช้งานระบบ โดยระบบที่พัฒนาขึ้นเป็นการประยุกต์ใช้เทคโนโลยี Large Language Model (LLM) ทำงานร่วมกับฐานข้อมูลความรู้ (Knowledge Base) ผ่านกระบวนการ Retrieval-Augmented Generation (RAG) โดยใช้แพลตฟอร์ม n8n เป็นตัวกลางในการจัดการเวิร์กโฟลว์เชื่อมต่อระหว่างแอปพลิเคชัน LINE และระบบประมวลผลปัญญาประดิษฐ์ ในการประเมินประสิทธิภาพได้ดำเนินการทดสอบด้วยชุดคำถามจำนวน 50 กรณีศึกษา และประเมินความพึงพอใจของผู้ใช้งานด้วยแบบสอบถาม ผลการดำเนินงานพบว่า การออกแบบเวิร์กโฟลว์ด้วยแพลตฟอร์ม n8n ช่วยให้ระบบสามารถรับส่งข้อความและดึงข้อมูลสินค้าเพื่อตอบคำถามลูกค้าได้อย่างถูกต้อง รวดเร็ว และเป็นธรรมชาติ ช่วยลดภาระการตอบคำถามซ้ำ ๆ ของมนุษย์ได้ ผลการประเมินความพึงพอใจจากกลุ่มตัวอย่างจำนวน 30 คน พบว่ามีความพึงพอใจในภาพรวมอยู่ในระดับมากที่สุด (ค่าเฉลี่ย 4.46) โดยเฉพาะอย่างยิ่งในด้านความเหมาะสมของรูปแบบและความสะดวกในการใช้งาน ซึ่งแสดงให้เห็นว่าระบบสามารถนำไปประยุกต์ใช้ในงานขายจริงได้อย่างมีประสิทธิภาพ

คำสำคัญ: *เอ็นเอทเอ็น; เอไอเอเจนท์; ระบบถาม-ตอบอัตโนมัติ; ไลน์แชทบอท*

1. บทนำ

ในยุคดิจิทัลปัจจุบัน ธุรกิจพาณิชย์อิเล็กทรอนิกส์ (E-commerce) มีอัตราการเติบโตอย่างก้าวกระโดด ส่งผลให้พฤติกรรมของผู้บริโภคเปลี่ยนแปลงไปโดยมักคาดหวังการบริการที่รวดเร็วทันใจ ความเร็วในการให้ข้อมูลแก่ลูกค้าจึงกลายเป็นปัจจัยชี้วัดที่สำคัญยิ่งต่อการตัดสินใจซื้อสินค้า (จักรินทร สันติรัตนภักดี, 2561) อย่างไรก็ตาม ปัญหาหลักที่ผู้ประกอบการส่วนใหญ่ต้องเผชิญคือ ข้อจำกัดด้านทรัพยากรบุคคลและเวลา ทำให้ไม่สามารถตอบกลับข้อความลูกค้าได้ตลอด 24 ชั่วโมง หรือเกิดความล่าช้าเมื่อมีปริมาณคำถามหลั่งไหลเข้ามาเป็นจำนวนมาก (วันชัย แซ่ลิ่ม, 2564; มนต์ทิชา รัตนพันธ์ และฉัตรวดี สายใยทอง, 2566) ซึ่งคำถามเหล่านั้นมักเป็นคำถามซ้ำๆ (Repetitive questions) ที่เกี่ยวข้องกับรายละเอียดและราคาสินค้า

ที่ผ่านมา เพื่อรับมือกับปัญหาดังกล่าว หลายองค์กรได้มีการประยุกต์ใช้ระบบตอบกลับอัตโนมัติหรือแชทบอท (Chatbot) ผ่านแพลตฟอร์มยอดนิยมอย่างแอปพลิเคชัน LINE เพื่อสนับสนุนงานบริการและการขายอย่างแพร่หลาย แต่ระบบแชทบอทส่วนใหญ่ยังคงพึ่งพาสถาปัตยกรรมแบบดั้งเดิม (Rule-based) ซึ่งทำงานภายใต้เงื่อนไขการตั้งค่าชุดคำถาม-คำตอบ (Keyword matching)

ไว้ล่วงหน้า (ณภัทร ไชยพราหมณ์ และคณะ, 2563) ข้อจำกัดสำคัญของระบบนี้คือขาดความยืดหยุ่นในการทำความเข้าใจบริบท หากลูกค้าพิมพ์คำถามด้วยรูปแบบประโยคที่ซับซ้อน สะกดผิด หรือใช้คำที่อยู่นอกเหนือจากคีย์เวิร์ดที่ตั้งไว้ แชทบอทจะไม่สามารถตอบสนองได้อย่างถูกต้องและเป็นธรรมชาติ ส่งผลให้ประสบการณ์การใช้งานของลูกค้าลดลง

เพื่อก้าวข้ามข้อจำกัดของแชทบอทแบบเดิม เทคโนโลยีปัญญาประดิษฐ์แบบสร้างสรรค์ (Generative AI) หรือโมเดลภาษาขนาดใหญ่ (Large Language Model: LLM) ได้เข้ามามีบทบาทสำคัญในการยกระดับการโต้ตอบด้วยภาษาธรรมชาติ (OpenAI, 2023) อย่างไรก็ตาม แม้ LLM จะมีความสามารถในการวิเคราะห์เจตนาของผู้ถามได้อย่างยอดเยี่ยม แต่ก็พบจุดอ่อนสำคัญคือ "อาการหลอนข้อมูล" (Hallucination) ซึ่งเกิดจากการที่ AI สร้างคำตอบขึ้นมาเองโดยไม่มีข้อมูลจริงรองรับ สิ่งนี้อาจสร้างความเสียหายต่อธุรกิจหาก AI แจ้งราคาสินค้าหรือโปรโมชันผิดพลาด ดังนั้น การนำเทคนิค Retrieval-Augmented Generation (RAG) มาทำงานร่วมกับฐานข้อมูลเฉพาะขององค์กร (Knowledge Base) จึงเป็นกระบวนการที่จำเป็นอย่างยิ่ง (Wei et al., 2022) เทคนิคนี้จะควบคุมให้ AI ทำหน้าที่เป็น "เอไอเอเจนท์ (AI Agent)" ที่ต้องสืบค้นข้อมูลจริงในฐานข้อมูลมาประมวลผลเป็นคำตอบเท่านั้น จึงมั่นใจได้ว่าข้อมูลที่สื่อสารไปยังลูกค้าจะถูกต้องและแม่นยำเสมอ ทั้งนี้ การพัฒนาระบบแชทบอทในปัจจุบันมีการใช้ AI Agent หลายรูปแบบ เช่น รูปแบบอิสระที่ตัดสินใจทั่วไป (Autonomous Agent) และรูปแบบที่ทำงานร่วมกับเครื่องมือเฉพาะทาง งานวิจัยนี้ใช้หลักการเลือก AI Agent ในรูปแบบที่เน้นการสืบค้นข้อมูล (Retrieval Agent) เนื่องจากต้องการความแม่นยำสูงสุดในการให้ข้อมูลสินค้า และป้องกันการหลอนข้อมูลให้ได้มากที่สุด

อย่างไรก็ตาม การพัฒนาระบบ AI Agent ที่มีกระบวนการ RAG ผสานเข้ากับแอปพลิเคชัน LINE มักมีความซับซ้อนทางเทคนิคและต้องอาศัยความเชี่ยวชาญด้านการเขียนโปรแกรมคอมพิวเตอร์ (Hard coding) ทำให้ผู้ประกอบการเข้าถึงเทคโนโลยีเหล่านี้ได้ยาก งานวิจัยนี้จึงเล็งเห็นช่องว่างดังกล่าว และนำเสนอการใช้เครื่องมือสร้างเวิร์กโฟลว์อัตโนมัติแบบ Low-code platform อย่าง n8n มาเป็นศูนย์กลาง (Orchestrator) แม้ว่าในปัจจุบันจะมีแพลตฟอร์มการสร้างเวิร์กโฟลว์รูปแบบอื่น เช่น Make หรือ Zapier แต่ผู้วิจัยเลือกใช้แพลตฟอร์ม n8n เนื่องจากเป็น Low-code platform ที่รองรับการทำงานแบบติดตั้งบนเซิร์ฟเวอร์ส่วนตัว (Self-hosted) ซึ่งช่วยเพิ่มความปลอดภัยของข้อมูลองค์กร ลดต้นทุนค่าใช้จ่าย และสามารถปรับแต่งการเชื่อมต่อกับโมเดลภาษาและฐานข้อมูลได้อย่างอิสระ (n8n, 2024) ระบบนี้จะทำหน้าที่เชื่อมต่อและควบคุมการแลกเปลี่ยนข้อมูลระหว่างแอปพลิเคชัน LINE, โมเดลภาษา LLM, และฐานข้อมูลสินค้าเข้าด้วยกันอย่างเป็นระบบโดยไม่ต้องเขียนโค้ดที่ซับซ้อน

ดังนั้น งานวิจัยนี้จึงมีจุดมุ่งหมายเพื่อศึกษาและออกแบบเวิร์กโฟลว์อัตโนมัติด้วย n8n ร่วมกับเอไอเอเจนท์ สำหรับประยุกต์ใช้เป็นระบบถาม-ตอบงานขายผ่านแอปพลิเคชัน LINE โดยมี

เป้าหมายเพื่อสร้างระบบที่สามารถตอบคำถามลูกค้าได้อย่างถูกต้อง เป็นธรรมชาติ ตลอด 24 ชั่วโมง เพื่อยกระดับประสิทธิภาพงานขาย ลดความซับซ้อนในการพัฒนาระบบ และลดภาระงานของผู้ประกอบการได้อย่างเป็นรูปธรรม

2. วัตถุประสงค์

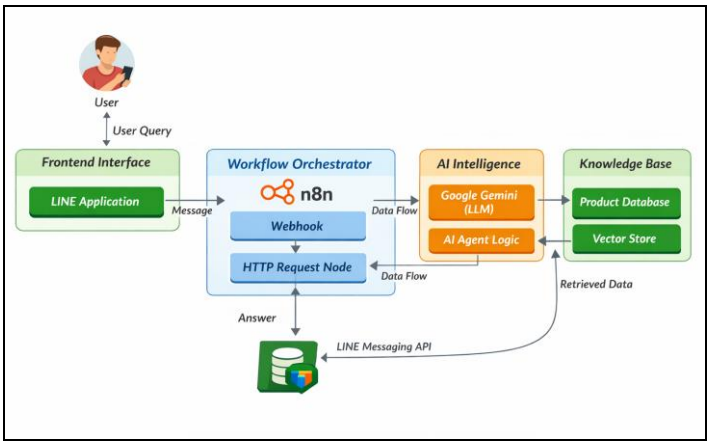
1. เพื่อพัฒนาและออกแบบเวิร์กโฟลว์อัตโนมัติด้วย n8n ร่วมกับ AI Agent สำหรับประยุกต์ใช้เป็นระบบถาม-ตอบงานขายผ่าน LINE Application
2. เพื่อประเมินประสิทธิภาพการทำงานของระบบ และความพึงพอใจของผู้ทดลองใช้งาน

3. วิธีดำเนินการ

ผู้วิจัยได้ดำเนินการออกแบบและพัฒนาระบบโดยแบ่งออกเป็นส่วนสถาปัตยกรรมและส่วนการออกแบบเวิร์กโฟลว์ นอกจากนี้ได้กำหนดระเบียบวิธีการวิจัยสำหรับการประเมินประสิทธิภาพการทำงานของระบบ โดยใช้วิธีการทดสอบด้วยชุดคำถาม (Test cases) และประเมินความพึงพอใจของผู้ใช้งานด้วยแบบสอบถาม รวมถึงมีการเปรียบเทียบประสิทธิภาพของระบบที่พัฒนาขึ้น (AI Agent และ RAG) กับระบบแชทบอทพื้นฐานเดิม (Rule-based) และเปรียบเทียบเวลาที่ใช้ในการตอบคำถามของระบบกับการทำงานด้วยตนเองของมนุษย์

3.1 สถาปัตยกรรมระบบ (System Architecture) ระบบทำงานโดยการผสาน 4 องค์ประกอบหลัก

1. Frontend Interface: ใช้ LINE Application เป็นช่องทางติดต่อสื่อสารระหว่างลูกค้ากับระบบ
2. Workflow Orchestrator: ใช้ n8n (Self-hosted/Cloud) เป็นศูนย์กลางควบคุมการไหลของข้อมูล (Data Flow)
3. AI Intelligence: ใช้ Large Language Model (LLM) ทำหน้าที่วิเคราะห์เจตนาและสร้างคำตอบ โดยงานวิจัยนี้เลือกใช้ AI Agent ในรูปแบบ Retrieval Agent เนื่องจากมีคุณสมบัติในการดึงข้อมูลเฉพาะเจาะจงจากฐานข้อมูลมาตอบคำถามได้อย่างแม่นยำ ซึ่งเหมาะกับบริบทของงานขายที่ต้องการข้อมูลสินค้าที่ถูกต้องและเป็นปัจจุบัน
4. Knowledge Base: ฐานข้อมูลสินค้า (เช่น Google Sheets หรือ Vector Store) โดยใช้โมเดล Text Embedding ของ Google Gemini ในการแปลงข้อความเป็นเวกเตอร์ สำหรับให้ AI Agent ค้นหาข้อมูลจำเพาะ

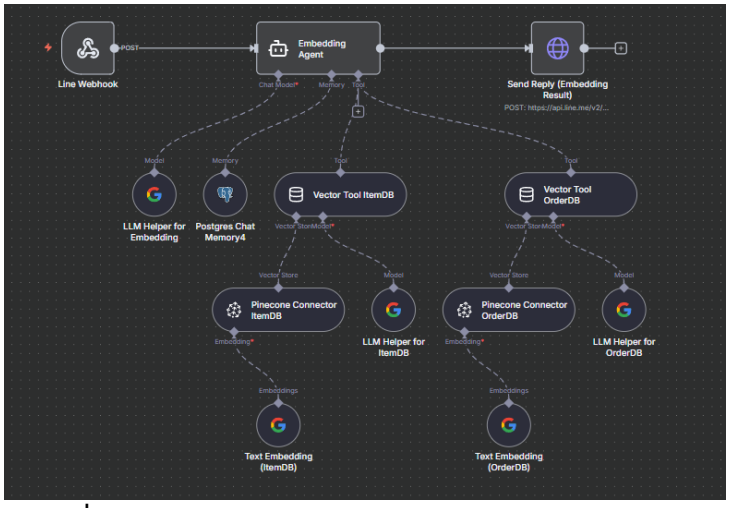


ภาพที่ 1 สถาปัตยกรรมภาพรวมของระบบถาม-ตอบอัตโนมัติด้วย n8n ร่วมกับเอไอเอเจนท์

ดังแสดงในภาพที่ 1 สถาปัตยกรรมระบบได้รับการออกแบบโดยผสานการทำงานของ 4 องค์ประกอบหลักเข้าด้วยกันอย่างเป็นระบบ เริ่มต้นจากผู้ใช้งานส่งข้อความสอบถามผ่าน Frontend Interface (แอปพลิเคชัน LINE) ข้อมูลจะถูกส่งมายัง Workflow Orchestrator ซึ่งทำหน้าที่เป็นศูนย์กลางควบคุมการไหลของข้อมูล โดยใช้แพลตฟอร์ม n8n ในการรับสัญญาณผ่าน Webhook จากนั้นระบบจะส่งข้อมูลข้อความไปยัง AI Intelligence ซึ่งใช้เทคโนโลยี Large Language Model (LLM) อย่าง Google Gemini ในการวิเคราะห์เจตนาของคำถาม ในกรณีที่คำถามต้องการข้อมูลเชิงลึกของสินค้า AI Agentจะทำการดึงข้อมูลจำเพาะจาก Knowledge Base (ฐานข้อมูลสินค้าจำพวก Vector Store) ผ่านกระบวนการ Retrieval-Augmented Generation (RAG) เมื่อ AI Agent ประมวลผลและสังเคราะห์คำตอบที่ถูกต้องเรียบร้อยแล้ว n8n จะทำหน้าที่ส่งข้อความตอบกลับไปยังผู้ใช้งานผ่าน LINE Messaging API ทันที ทำให้กระบวนการทั้งหมดเกิดขึ้นแบบอัตโนมัติ (Automated Workflow) และตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติ

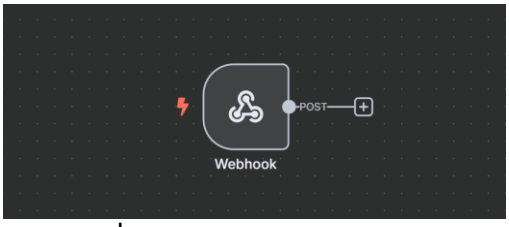
3.2 การออกแบบเวิร์กโฟลว์บน n8n (n8n Workflow Design)

ในการออกแบบเวิร์กโฟลว์บนแพลตฟอร์ม n8n ผู้วิจัยได้กำหนดโครงสร้างการทำงานอัตโนมัติ (Automated Workflow) เพื่อทำหน้าที่เป็นศูนย์กลางในการรับส่งและประมวลผลข้อมูลระหว่างแอปพลิเคชัน LINE, AI Agent, และฐานข้อมูลความรู้ โดยระบบถูกขับเคลื่อนผ่านการเชื่อมต่อโหนด (Nodes) คำสั่งหลักที่ทำงานผสานกันตามลำดับขั้นตอนดังแสดงในภาพที่ 2 ดังต่อไปนี้



ภาพที่ 2 แผนผังเวิร์กโฟลว์การทำงานรวมบนแพลตฟอร์ม n8n

1. Webhook Node (Start) ทำหน้าที่รอรับสัญญาณ Event แบบ HTTP POST จาก LINE Messaging API เมื่อมีผู้ใช้งานพิมพ์ข้อความเข้ามา ดังแสดงในภาพที่ 3



ภาพที่ 3 ตัวอย่าง Node Webhook

เมื่อ LINE Messaging API ส่งข้อมูลมายัง Webhook Node ข้อมูลจะอยู่ในรูปแบบ JSON โดยมีโครงสร้างหลักดังนี้

```
{
  "events": [
    {
      "replyToken": "xxxxxxxxxx",
      "type": "message",
      "source": {
        "userId": "Uxxxxxxxx"
      },
      "message": {
        "type": "text",
        "id": "1234567890",
```

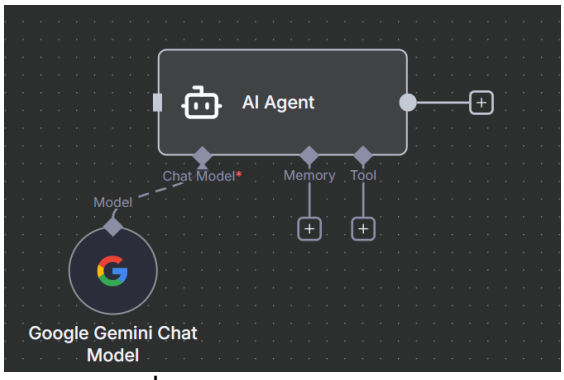
```
        "text": "สินค้าตัวนี้ราคาเท่าไหร่"
      }
    }
  ]
}
```


จากโครงสร้าง JSON ดังกล่าว ระบบจะดึงข้อมูลฟิลด์สำคัญมาใช้งาน ได้แก่

- replyToken ใช้สำหรับส่งข้อความตอบกลับไปยังผู้ใช้งานใน session เดียวกัน
- message.text ข้อความที่ผู้ใช้งานพิมพ์เข้ามา
- source.userId รหัสประจำตัวผู้ใช้งาน

ข้อมูลเหล่านี้จะถูก Extract ภายใน n8n และส่งต่อไปยัง AI Agent Node เพื่อดำเนินการวิเคราะห์เจตนาและสร้างคำตอบในขั้นตอนถัดไป

2. AI Agent Node (Chain) เชื่อมต่อกับ API ของโมเดลภาษา (เช่น Gemini) โดยมีการกำหนด System Prompt ให้ AI สวมบทบาทเป็น "พนักงานขาย" ที่มีความสุภาพและเชี่ยวชาญข้อมูลสินค้า ดังแสดงในภาพที่ 4



ภาพที่ 4 ตัวอย่าง AI Agent Node

ผู้วิจัยได้ออกแบบ System Prompt เพื่อควบคุมพฤติกรรมของ AI Agent ให้ทำงานในลักษณะ Retrieval-Augmented Generation (RAG) อย่างเคร่งครัด โดยกำหนดให้เอไอเรียกใช้เครื่องมือเวกเตอร์ทั้งสองฐานข้อมูลทุกครั้งก่อนสร้างคำตอบ และห้ามสร้างข้อมูลขึ้นเอง (Hallucination Prevention) นอกจากนี้ยังออกแบบให้รองรับการใช้งานหน่วยความจำ (Memory) ตาม userId เพื่อรองรับการสนทนาแบบต่อเนื่อง

ตัวอย่าง System Prompt ที่ใช้ควบคุม AI Agent

ระบบ: คุณคือ Embedding Agent สำหรับ LINE Chatbot

หน้าที่: รับคำถามผู้ใช้ ⟶ เรียกใช้ Vector Tool ทั้งสองตัวทุกครั้ง ⟶ รวมผลและสรุปคำตอบภาษาไทยให้อ่านง่าย

เครื่องมือที่ต้องเรียกใช้ทุกครั้ง:

1) Vector Tool ItemDB (namespace: itemdb)

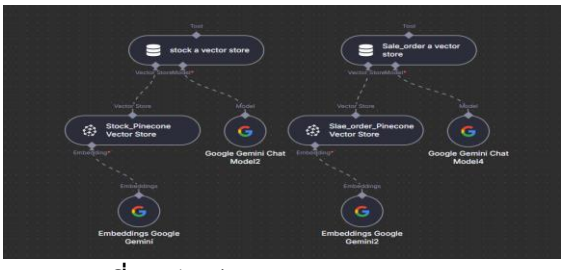
2) Vector Tool OrderDB (namespace: order_db)

กติกา:

- ต้องเรียกใช้ทั้งสองเครื่องมือทุกครั้ง
- ห้ามแต่งข้อมูลเอง
- หากไม่พบข้อมูลจากทั้งสองฐาน ⟶ ตอบว่า "ไม่พบข้อมูล"
- สรุปคำตอบเป็นภาษาไทยล้วน สั้น กระชับ

จากโครงสร้าง Prompt ข้างต้น จะเห็นได้ว่าผู้วิจัยกำหนดข้อบังคับเชิงตรรกะ (Instruction Constraints) เพื่อควบคุมลำดับการทำงานของเอไอเอเจนท์ โดยกำหนดให้เรียกใช้ Vector Tool ทั้งสองฐานข้อมูลเสมอ ก่อนสร้างคำตอบ ซึ่งเป็นการบังคับใช้กระบวนการ RAG อย่างชัดเจน นอกจากนี้ยังมีการกำหนดกฎห้ามแต่งข้อมูลเอง และกำหนดรูปแบบการตอบกลับให้กระชับ เพื่อให้เหมาะสมกับบริบทการสนทนาผ่าน LINE Application

3. Data Retrieval Tool ในกรณีที่คำถามเกี่ยวข้องกับรายละเอียดสินค้า AI จะเรียกใช้เครื่องมือเพื่อดึงข้อมูลล่าสุดจากฐานข้อมูล Knowledge Base (RAG) เพื่อให้คำตอบมีความถูกต้องแม่นยำ ดังแสดงในภาพที่ 5



ภาพที่ 5 ตัวอย่าง Data Retrieval Tool

ในขั้นตอน Data Retrieval Tool AI Agent จะสร้างคำค้น (vector_query) จากคำถามของผู้ใช้งาน แล้วส่งไปยังฐานข้อมูลเวกเตอร์ (Vector Store) เพื่อค้นหาเอกสารหรือรายการข้อมูลที่มีความใกล้เคียงทางความหมาย (Semantic Similarity) มากที่สุด

โครงสร้างข้อมูลสินค้าในฐานข้อมูล (ItemDB) ที่เอไอสามารถเข้าถึงได้ ประกอบด้วยฟิลด์หลัก ดังนี้

- product_id : รหัสสินค้า
- product_name : ชื่อสินค้า
- description : รายละเอียดสินค้า
- specification : คุณสมบัติหรือสเปก
- price : ราคาจำหน่าย
- stock_quantity : จำนวนคงเหลือ
- promotion : ข้อมูลโปรโมชั่น (ถ้ามี)
- last_update : วันที่อัปเดตข้อมูลล่าสุด

สำหรับฐานข้อมูลคำสั่งซื้อ (OrderDB) โครงสร้างข้อมูลประกอบด้วย

- order_id : รหัสคำสั่งซื้อ
- customer_id : รหัสลูกค้า
- product_id : รหัสสินค้า
- order_date : วันที่สั่งซื้อ
- quantity : จำนวนที่สั่ง
- order_status : สถานะคำสั่งซื้อ
- sales_staff : ผู้ดูแลงานขาย

ตัวอย่างรูปแบบข้อมูลที่ได้กลับมา


```
{
  "matches": [
    {
      "product_id": "ITM001",
      "product_name": "Drone X200",
      "price": 15900,
      "stock_quantity": 12,
      "score": 0.92
    }
  ]
}
```

ผลลัพธ์ที่ได้จาก Vector Store จะถูกส่งกลับไปยัง AI Agent ในรูปแบบ context เพื่อใช้ประกอบการสร้างคำตอบตามกระบวนการ Retrieval-Augmented Generation (RAG) โดยเอไอจะสรุปเฉพาะข้อมูลที่อยู่ใน context ที่ได้รับ และไม่สร้างข้อมูลเพิ่มเติมนอกเหนือจากผลการค้นหา

4. HTTP Request Node (Reply) ส่งข้อความตอบกลับไปยังผู้ใช้งานผ่าน LINE Reply Token ดังแสดงในภาพที่ 6

หลังจาก AI Agent สังเคราะห์คำตอบเรียบร้อยแล้ว ระบบจะเข้าสู่ขั้นตอนการส่งข้อความตอบกลับ (Reply) ผ่าน HTTP Request Node โดย n8n จะทำหน้าที่จัดรูปแบบข้อมูลให้อยู่ในรูปแบบที่ LINE Messaging API กำหนด และส่งคำขอแบบ HTTP POST ไปยัง Endpoint สำหรับการตอบกลับข้อความ

Endpoint ที่ใช้ในการตอบกลับข้อความคือ https://api.line.me/v2/bot/message/reply ซึ่งเป็น API สำหรับตอบกลับข้อความแบบ Reply ภายใน session เดียวกัน โดยต้องใช้ค่า replyToken ที่ได้รับจาก Webhook ในขั้นตอนแรก

ตัวอย่าง Data Payload

```
{
  "replyToken": "xxxxxxxxxxxxxxxx",
  "messages": [
    {
      "type": "text",
      "text": "สินค้ารุ่น X200 ราคา 15,900 บาท และมีสินค้าในสต๊อก
จำนวน 12 ชิ้น"
    }
  ]
}
```

โดยมีฟิลด์สำคัญดังนี้

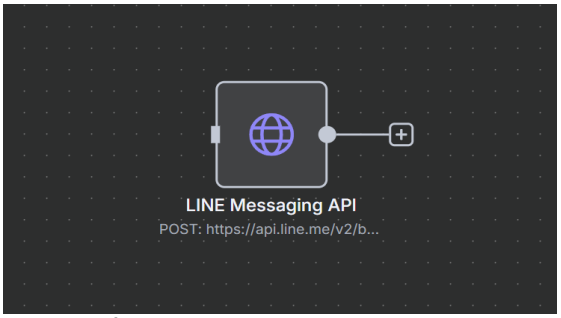
- replyToken : ใช้ระบุ session ของข้อความที่ต้องการตอบกลับ
- messages : โครงสร้างข้อความที่ต้องการส่งกลับ ซึ่งสามารถเป็น text, image, flex message หรือรูปแบบอื่นตามที่ LINE รองรับ

ในส่วนของ Header ระบบจะต้องแนบ Channel Access Token เพื่อยืนยันตัวตนของ LINE Official Account ดังตัวอย่าง

Authorization: Bearer {Channel Access Token}

Content-Type: application/json

กระบวนการทั้งหมดตั้งแต่รับข้อความจนถึงส่งคำตอบกลับเกิดขึ้นภายในรอบการทำงานเดียว (Single Workflow Execution) ทำให้สามารถตอบสนองแบบ Real-time โดยจากการทดสอบพบว่าค่าเฉลี่ยระยะเวลาการตอบกลับอยู่ที่ 2–3 วินาทีต่อข้อความ ซึ่งอยู่ในระดับที่เหมาะสมต่อการใช้งานเชิงพาณิชย์



ภาพที่ 6 ตัวอย่าง HTTP Request Node

4. ผลการศึกษา

จากการพัฒนาและทดสอบระบบถาม-ตอบงานขายอัตโนมัติ ผู้วิจัยขอเสนอผลการศึกษาแบ่งเป็น 2 ส่วน ดังนี้

4.1 ผลการทดสอบการทำงานของระบบ (Functional Testing) จากการทดสอบเวิร์กโฟลว์ที่ออกแบบด้วย n8n พบว่าระบบสามารถทำงานได้ตามเงื่อนไขที่กำหนดอย่างสมบูรณ์ เมื่อเปรียบเทียบประสิทธิภาพของระบบที่พัฒนาขึ้น (AI Agent และ RAG) กับระบบพื้นฐานเดิม (Rule-based chatbot) พบว่าระบบใหม่สามารถวิเคราะห์เจตนาจากข้อความที่ซับซ้อนและพิมพ์สะกดผิดได้ดีกว่าอย่างชัดเจน นอกจากนี้ เมื่อเปรียบเทียบระยะเวลาการทำงาน พบว่าการตอบคำถามด้วยตนเอง (Manual) จำนวน 50 ครั้ง ใช้เวลาเฉลี่ย 45 วินาทีต่อข้อความ ในขณะที่การทำงานด้วยระบบจำนวน 50 ครั้ง ใช้เวลาตอบกลับเฉลี่ยเพียง 2.31 วินาทีต่อข้อความ การเชื่อมต่อ Webhook สามารถรับข้อความจาก LINE ได้แบบ Real-time โดยมีความหน่วง (Latency) เฉลี่ยอยู่ในเกณฑ์ที่ยอมรับได้

- ความถูกต้อง AI Agent สามารถแยกแยะเจตนาของคำถามและดึงข้อมูลจาก Knowledge Base มาตอบได้ถูกต้อง เช่น การบอกราคาสินค้า การแจ้งโปรโมชั่น โดยไม่เกิดอาการหลอนข้อมูล (Hallucination) ในหัวข้อที่ทดสอบ

จากการทดสอบระบบถาม-ตอบงานขายอัตโนมัติ ผู้วิจัยได้ดำเนินการทดสอบจำนวน 50 กรณีคำถาม (Test Cases) ซึ่งสร้างขึ้นจากการรวบรวมประวัติการสนทนาจริงของลูกค้า (Chat

history) โดยมีผู้ดูแลระบบงานขายของบริษัทเป็นผู้คัดเลือกชุดคำถาม เพื่อให้ครอบคลุม 5 กลุ่มเจตนา (Intents) หลัก เกี่ยวกับ

- ราคาสินค้า
- รายละเอียดและสเปกสินค้า
- สถานะสต๊อกสินค้า
- โปรโมชั่น
- ข้อมูลคำสั่งซื้อ

1. ความแม่นยำของคำตอบ

   ผลการทดสอบดังแสดงในตารางที่ 1 และตารางที่ 2 มีรายละเอียดดังนี้

   - ตอบถูกต้อง 48 คำถาม
   - ตอบคลาดเคลื่อน 2 คำถาม

   โดยคำตอบที่คลาดเคลื่อนเกิดจากการดึงข้อมูลที่มีความใกล้เคียงทางความหมายแต่ไม่ตรงรุ่นสินค้า

2. ระยะเวลาการตอบกลับ

   วัดระยะเวลาตั้งแต่ Webhook รับข้อความจนถึง HTTP Reply ส่งกลับผู้ใช้งาน โดยใช้ Timestamp ใน n8n

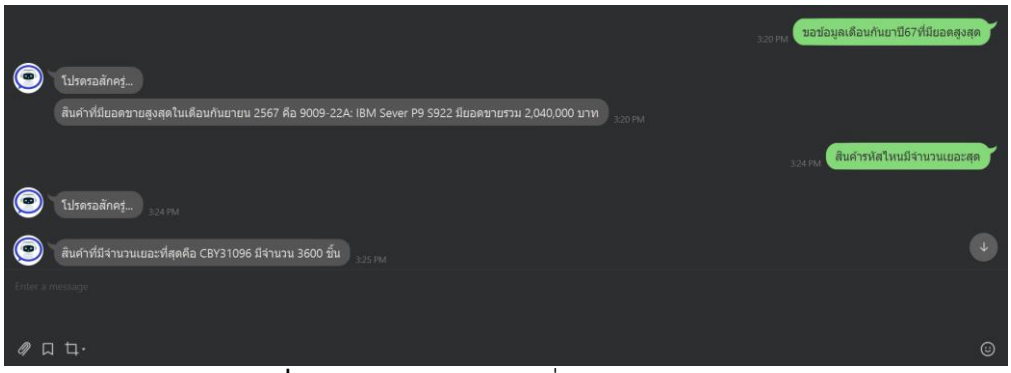
   - ค่าเฉลี่ย (Mean) = 2.31 วินาทีต่อข้อความ
   - ค่าสูงสุด = 3.84 วินาที
   - ค่าต่ำสุด = 1.72 วินาที
   - ส่วนเบี่ยงเบนมาตรฐาน (S.D.) = 0.54

   ซึ่งอยู่ในระดับที่เหมาะสมต่อการใช้งานเชิงพาณิชย์ผ่าน LINE Application

3. อัตราการเกิดอาการหลอนข้อมูล (Hallucination Rate) จากการทดสอบทั้ง 50 คำถาม ไม่พบกรณีที่ระบบสร้างข้อมูลราคาหรือโปรโมชั่นที่ไม่มีอยู่ในฐานข้อมูล คิดเป็นอัตราการเกิด Hallucination เท่ากับ 0% โดยมีวิธีการวัดผลจากการตรวจสอบและเทียบเคียงคำตอบของ AI Agent กับชุดข้อมูลจริงในฐานข้อมูล Knowledge Base หากพบว่า AI สร้างข้อมูลใดที่ไม่ปรากฏในฐานข้อมูล จะนับเป็นการเกิดข้อผิดพลาด 1 ครั้ง ทั้งนี้เนื่องจากมีการบังคับให้ AI Agent ใช้ข้อมูลผ่านกระบวนการ RAG เท่านั้น

ตารางที่ 1 สรุปผลการทดสอบประสิทธิภาพระบบ

ตัวชี้วัด	ค่า
จำนวนคำถามทดสอบ	50
ตอบถูกต้อง	48
Accuracy	96.00%
ค่าเฉลี่ย Response Time	2.31 วินาที
Hallucination Rate	0%



ภาพที่ 7 ตัวอย่างแชท LINE ที่แสดงการตอบคำถาม

ตารางที่ 2 ผลการทดสอบตามสถานการณ์

ลำดับ	สถานการณ์ทดสอบ	ตัวอย่างคำถาม	ผลลัพธ์ของระบบ	ความถูกต้อง
1	ถามราคาสินค้าที่มีในสต๊อก	“Drone X200 ราคาเท่าไหร่”	แจ้งราคาถูกต้องและจำนวนคงเหลือ	✓
2	พิมพ์สะกดผิด	“โดรน x200 ราคาไหร”	AI เข้าใจบริบทและตอบได้ถูกต้อง	✓
3	ถามสินค้าที่หมดสต๊อก	“Drone Z500 มีของไหม”	แจ้งว่าสินค้าหมดและเสนอรุ่นใกล้เคียง	✓

4.2 ผลการประเมินความพึงพอใจของผู้ใช้งาน

สำหรับการประเมินความพึงพอใจของผู้ใช้งานที่มีต่อระบบถาม-ตอบอัตโนมัตินี้ ผู้วิจัยได้ใช้แบบสอบถาม (Questionnaire) เป็นเครื่องมือในการเก็บรวบรวมข้อมูล ซึ่งได้รับการทดสอบความ

เที่ยง (Reliability) ของแบบสอบถามก่อนนำไปใช้งานจริง โดยดำเนินการทดสอบการยอมรับระบบ (User Acceptance Testing: UAT) กับกลุ่มตัวอย่างแบบเจาะจง จำนวน 30 คน เพื่อให้เป็นไปตามหลักการวิจัยที่เหมาะสม ผลการวิเคราะห์ความพึงพอใจของผู้ใช้งาน โดยเปรียบเทียบกับค่าเกณฑ์ 3.50 พบว่า ค่าเฉลี่ยความพึงพอใจในภาพรวมเท่ากับ 4.46 ซึ่งมีค่าสูงกว่าเกณฑ์ที่กำหนด อันสะท้อนถึงระดับความพึงพอใจของผู้ใช้งานที่อยู่ในระดับมากที่สุด ผลการวิเคราะห์ข้อมูลแสดง ดังตารางที่ 3

เทคนิค Questionnaire

การประเมินความพึงพอใจของผู้ใช้งานระบบ เป็นอีกวิธีหนึ่งที่ใช้ในการทดสอบประสิทธิภาพของระบบที่พัฒนาขึ้น โดยใช้แบบสอบถาม (Questionnaire) เป็นเครื่องมือในการเก็บรวบรวมความคิดเห็นจากผู้ใช้งานจริง ซึ่งเป็นพนักงานของบริษัทที่ได้ทดลองใช้งานระบบผ่าน LINE Official Account แบบสอบถามนี้มีวัตถุประสงค์เพื่อประเมินความพึงพอใจด้านการใช้งาน ความสะดวก และความเหมาะสมของระบบ โดยผลที่ได้จะนำไปใช้ประกอบการวิเคราะห์ผลการทดสอบระบบ

รูปแบบของแบบสอบถาม

แบบสอบถามที่ใช้ในการประเมินความพึงพอใจเป็นแบบมาตราส่วนประมาณค่า 5 ระดับ (Likert Scale) โดยมีรายละเอียดดังนี้

ระดับความคิดเห็น

1 = น้อยที่สุด

2 = น้อย

3 = ปานกลาง

4 = มาก

5 = มากที่สุด

N/A = ไม่ระบุ

$\bar{X}$ = ค่าเฉลี่ย

S.D. = ส่วนเบี่ยงเบนมาตรฐาน

เกณฑ์การแปลผลค่าคะแนนเฉลี่ยความพึงพอใจ

คะแนนเฉลี่ย 1.00 – 1.80 หมายถึง ระดับน้อยที่สุด

คะแนนเฉลี่ย 1.81 – 2.60 หมายถึง ระดับน้อย

คะแนนเฉลี่ย 2.61 – 3.40 หมายถึง ระดับปานกลาง

คะแนนเฉลี่ย 3.41 – 4.20 หมายถึง ระดับมาก

คะแนนเฉลี่ย 4.21 – 5.00 หมายถึง ระดับมากที่สุด

ตารางที่ 3 แสดงผลการประเมินความพึงพอใจของผู้ใช้งานระบบ

รายการ	**ระดับความพึงพอใจ จำนวน ร้อยละ**									
	5	**4**	**3**	**2**	**1**	**N/A**	**รวม**	$\overline{X}$	**SD**	**แปลผล**
1. ความเหมาะสมของการออกแบบ	18	6	6	0	0	0	30	4.40	0.81	มากที่สุด
2. ความเหมาะสมของรูปแบบ	21	6	3	0	0	0	30	4.60	0.67	มากที่สุด
3. ความเหมาะสมการใช้สี	18	6	6	0	0	0	30	4.40	0.81	มากที่สุด
4. ความน่าใช้งาน	15	9	6	0	0	0	30	4.30	0.79	มากที่สุด
5. ความสะดวก	21	6	3	0	0	0	30	4.60	0.67	มากที่สุด
ภาพรวมความพึงพอใจ								**4.46**	**มากที่สุด**	

จากตารางที่ 3 พบว่าภาพรวมอยู่ในระดับมากที่สุด ($\overline{X}$ = 4.46) ปัจจัยที่ได้รับคะแนนสูงสุด คือ ด้านความเหมาะสมของรูปแบบและด้านความสะดวกในการใช้งาน ($\overline{X}$ = 4.60) โดยผู้ใช้งานมีความเห็นว่าระบบมีการจัดเรียงรูปแบบคำตอบข้อความที่อ่านง่าย และมีความสะดวกอย่างมาก เนื่องจากระบบสามารถดึงข้อมูลที่ต้องการมาตอบกลับได้อย่างรวดเร็ว รองลงมาคือความเหมาะสมของการออกแบบและการใช้สี ($\overline{X}$ = 4.40) ซึ่งผู้ใช้งานรู้สึกว่าลักษณะการแสดงผลและโทนสีมีความสบายตา และกลมกลืนกับอินเทอร์เฟซของแอปพลิเคชัน LINE ได้อย่างเป็นธรรมชาติ

5. สรุปและอภิปรายผล

งานวิจัยนี้ประสบความสำเร็จในการออกแบบและพัฒนาเวิร์กโฟลว์อัตโนมัติด้วยแพลตฟอร์ม n8n ร่วมกับเทคโนโลยีเอไอเอเจนท์ (AI Agent) เพื่อประยุกต์ใช้เป็นระบบถาม-ตอบงานขายผ่านแอปพลิเคชัน LINE จากผลการทดสอบประสิทธิภาพของระบบ(Functional Testing) จำนวน 50 กรณีศึกษา (Test cases) พบว่าสถาปัตยกรรมที่ผสานการทำงานของโมเดลภาษาขนาดใหญ่ (LLM) ร่วมกับฐานข้อมูลความรู้ผ่านเทคนิค RAG (Retrieval-Augmented Generation) สามารถประมวลผลเจตนาของลูกค้าและตอบคำถามได้อย่างถูกต้อง แม่นยำถึง ร้อยละ 96 โดยมีระยะเวลาความหน่วง (Latency) ในการตอบกลับ เฉลี่ยเพียง 2.31 วินาที ต่อข้อความ นอกจากนี้ กระบวนการ RAG ยังสามารถป้องกันปัญหาเอไอหลอนข้อมูล (Hallucination) ได้อย่างสมบูรณ์ ซึ่งผลลัพธ์ด้านความถูกต้องแม่นยำและการป้องกันการหลอนข้อมูลนี้ สอดคล้องกับงานวิจัยของ Wei et al. (2022) ที่พบว่าเทคนิค RAG ช่วยควบคุมและยกระดับความน่าเชื่อถือของการสร้างคำตอบในโมเดลภาษาขนาดใหญ่ได้อย่างมีประสิทธิภาพ

ในด้านผลการประเมินการยอมรับระบบ (User Acceptance Testing: UAT) จากกลุ่มพนักงานผู้ทดลองใช้งาน พบว่ามีความพึงพอใจในภาพรวมอยู่ในระดับมากที่สุด ($\bar{X}$ = 4.46) ผลลัพธ์ดังกล่าวสามารถอภิปรายได้ว่า การออกแบบเวิร์กโฟลว์ด้วย n8n ซึ่งเป็นเครื่องมือแบบ Low-code ช่วยแก้ปัญหาความซับซ้อนในการพัฒนาระบบได้อย่างมีนัยสำคัญ โครงสร้างที่ออกแบบไว้มีความยืดหยุ่นสูง ช่วยให้ผู้ดูแลระบบสามารถอัปเดตข้อมูลสินค้าผ่านฐานข้อมูลได้โดยตรงโดยไม่ต้องแก้ไขโค้ดโปรแกรม สอดคล้องกับสมมติฐานที่ว่าการนำ AI Agent มาใช้สามารถลดข้อจำกัดของแชทบอทแบบเดิม (Rule-based) ด้วยการสื่อสารที่เป็นธรรมชาติและยืดหยุ่นกว่า รวมถึงสอดคล้องกับงานวิจัยก่อนหน้าที่พบว่าการประยุกต์ใช้เทคโนโลยี AI และแชทบอทในงานบริการลูกค้าช่วยเพิ่มความรวดเร็วและประสิทธิภาพในการตอบสนองความต้องการของผู้บริโภคได้อย่างชัดเจน (จักรินทร สันติรัตนภักดี, 2561) ท้ายที่สุด ระบบนี้ช่วยลดภาระงานของบุคลากรในการตอบคำถามซ้ำซากได้อย่างเป็นรูปธรรม และยกระดับประสบการณ์การบริการลูกค้าให้ได้รับข้อมูลที่ถูกต้องตลอด 24 ชั่วโมง

งานที่สืบเนื่องในอนาคต

ควรขยายขีดความสามารถของสถาปัตยกรรมระบบให้รองรับการประมวลผลข้อมูลในรูปแบบมัลติมีเดีย เช่น การวิเคราะห์ภาพถ่ายสลิปการโอนเงิน หรือการรับส่งพิกัดตำแหน่ง เพื่อสนับสนุนธุรกรรมการขายให้มีความครอบคลุมและครบวงจรมากยิ่งขึ้น พร้อมทั้งควรประยุกต์ใช้เทคนิคการจัดเก็บข้อมูลชั่วคราวสำหรับชุดคำถามที่มีการใช้งานบ่อย เพื่อช่วยลดระยะเวลาความหน่วงในการประมวลผล และลดต้นทุนจากการเรียกใช้งาน API ของโมเดลภาษาขนาดใหญ่อย่างมีประสิทธิภาพ นอกจากนี้ ควรมีการศึกษาและทบทวนเพิ่มเติมเกี่ยวกับเทคโนโลยีและ

กระบวนการ RAG (Retrieval-Augmented Generation) ในรูปแบบขั้นสูง (Advanced RAG) เพื่อยกระดับความแม่นยำในการดึงข้อมูลที่มีความซับซ้อนมากขึ้น รวมถึงควรมีการพัฒนาความเข้มแข็งเชิงระเบียบวิธีและเชิงวิชาการในการวิจัยระยะต่อไป เช่น การขยายขอบเขตการประเมินผลการใช้งานในระยะยาว หรือการทดสอบกับกลุ่มลูกค้าเป้าหมายจริงในสเกลที่ใหญ่ขึ้น เพื่อยืนยันประสิทธิภาพของระบบได้อย่างครอบคลุม

6. กิตติกรรมประกาศ

ขอขอบคุณภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้ ที่ให้การสนับสนุนในการทำโครงงานครั้งนี้ และขอขอบคุณบริษัท บลู ซิส เต็ม เทคโนโลยี จำกัด และทีมงานได้แก่ คุณโสภณ เพชรนิล และ คุณนภัสนันท์ จิตรนันทิวัฒน์ ที่ให้คำแนะนำและการสนับสนุนตลอดระยะเวลาการดำเนินงาน

7. เอกสารอ้างอิง

จตุรงค์ จิตติยพล, & พงษ์ศักดิ์ ดรพินิจ. (2565). **การพัฒนาระบบสารสนเทศเพื่อการเรียนรู้คำศัพท์เครือข่ายคอมพิวเตอร์ผ่านแอปพลิเคชัน Line Chatbot**. *วารสารสหวิทยาการวิจัยและวิชาการ, 2*(4), 607-618. https://so03.tci-thaijo.org/index.php/IARJ/article/view/263016/174812

จักรินทร สันติรัตนภักดี. (2561). **การตลาดออนไลน์และบริการลูกค้าด้วยแชทบอท กรณีศึกษา : การใช้ Chatfuel ปฏิสัมพันธ์กับลูกค้าผ่านเมสเซนเจอร์.** *วารสารศรีปทุมปริทัศน์ ฉบับวิทยาศาสตร์และเทคโนโลยี, 10*, 71-87. https://ph02.tci-thaijo.org/index.php/spurst/article/download/162381/117162

ณภัทร ไชยพราหมณ์, ณัฐวุฒิ ทุมนัต, & ชูพันธุ์ รัตนโภคา. (2563). **ระบบตอบกลับและแจ้งข้อมูลทางการศึกษาผ่านไลน์บอท.** *Journal of Information Science and Technology, 10*(2), 59-70.บุญชม ศรีสะอาด. (2556). *การวิจัยเบื้องต้น*. สุวีริยาสาส์น. https://ph02.tci-thaijo.org/index.php/JIST/article/download/241894/164548

มนต์ทิชา รัตนพันธ์, & ฉัตรวดี สายใยทอง. (2566). **การพัฒนา Line Chatbot สำหรับงานบริการด้านงานวิจัย กรณีศึกษาสถาบันวิจัยและพัฒนา มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี.** *Journal of Applied Research on Science and Technology (JARST), 22*(1), 78-89. https://ph01.tci-thaijo.org/index.php/rmutt-journal/article/download/251602/170619 /919924

วันชัย แซ่ลิ่ม. (2564). **การพัฒนาระบบโต้ตอบอัตโนมัติผ่านแอปพลิเคชันไลน์ เพื่อสนับสนุนการให้บริการแก้ปัญหาคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์.** *วารสารวิชาการ ปชมท, 10*(2), 56-65. https://www.council-uast.com/journal/dl-file.php?id=162

LINE Developers. (2023). ***Messaging API Reference*.** Retrieved from https://developers.line.biz/en/reference/messaging-api/

n8n. (2024). ***Workflow Automation with n8n*. Retrieved from https://docs.n8n.io/**

OpenAI. (2023). *GPT-4 Technical Report*. Retrieved from https://arxiv.org/abs/2303.08774

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., & Zhou, D. (2022). **Chain-of-thought prompting elicits reasoning in large language models.** *Advances in Neural Information Processing Systems, 35*, 24824-24837. https://arxiv.org/abs/2201.11903