

Claude Automation

แนวทางใช้ Claude ทำ automation · งานวิศวกรรมโยธา ขนส่ง-จราจร และงานราชการด้าน
ก่อสร้าง

ฉบับปรับปรุง · มิถุนายน 2568 — Opus 4.8 · Sonnet 4.6 · Haiku 4.5
เรียบเรียงโดย ศิวา นาคอ้าย · วิศวกรโยธาชำนาญการ · สำนักสำรวจและออกแบบ กรมทางหลวง

Claude Automation · คู่มือฉบับภาษาไทย

แนวทางการใช้ Claude ทำ Automation สำหรับงานวิศวกรรมโยธา · วิศวกรรมขนส่งและจราจร · งานราชการด้านก่อสร้าง ฉบับปรับปรุง · มิถุนายน 2568 — ครอบคลุม Opus 4.8 · Sonnet 4.6 · Haiku 4.5

คู่มือนี้สำหรับใคร

วิศวกรและเจ้าหน้าที่ในหน่วยงานราชการที่เกี่ยวข้องกับงานก่อสร้างทุกประเภท — เช่น กรมทางหลวง, กรมทางหลวงชนบท, กรมโยธาธิการและผังเมือง, กรมชลประทาน, การทางพิเศษฯ, องค์กรปกครองส่วนท้องถิ่น และหน่วยงานที่ทำงานออกแบบ ประมาณราคา จัดซื้อจัดจ้าง และควบคุมงานก่อสร้าง ไม่ต้องมีพื้นฐานการเขียนโปรแกรมมาก่อน — เริ่มจากการใช้งานแบบ **ไม่เขียนโค้ด (no-code)** ได้ทันที แล้วค่อยต่อยอดสู่ **automation** จริงจัง ด้วย Claude Code

คู่มือนี้ให้อะไร

- กรอบคิดที่ถูกต้อง สำหรับใช้ AI ในงานที่มีความรับผิดชอบทางวิชาชีพและกฎหมาย
- **Use case** จริง แยกตามสายงาน: โยธา · ขนส่ง-จราจร · เอกสารราชการ/ก่อสร้าง
- ขั้นตอนลงมือทำ ทั้ง no-code (Claude.ai/Projects/Skills) และ automation (Claude Code/MCP)
- คลัง Prompt ภาษาไทย ที่คัดลอกไปใช้ได้เลย
- แผนเริ่มต้น 30/60/90 วัน และ Checklist ความปลอดภัย/การตรวจสอบ

วิธีอ่าน

ใช้เมนูด้านซ้ายเลือกบทได้ตามต้องการ · มีช่องค้นหาด้านบน (พิมพ์แล้วกด Enter เพื่อไฮไลต์) · สลับโหมดสว่าง/มืดได้ · ดาวน์โหลดฉบับ PDF ได้จากปุ่มด้านบนหรือการ์ดในหน้านี้

สารบัญ

ส่วนที่ 1 — กรอบคิดและการเตรียมตัว

1. ภาพรวมและปรัชญาการใช้งาน

2. เครื่องมือ Claude และการเลือกใช้ให้เหมาะกับงาน

3. หลักการสำคัญสำหรับงานราชการ (อ่านก่อนเริ่ม)

ส่วนที่ 2 — การประยุกต์ตามสายงาน 4. งานวิศวกรรมโยธา 5. งานวิศวกรรมขนส่งและจราจร 6. งานราชการและเอกสารงานก่อสร้าง

ส่วนที่ 3 — การลงมือทำ 7. ระดับ no-code: Claude.ai, Projects, Skills 8. ระดับ automation: Claude Code, Skills, MCP 9. คลัง Prompt และเทคนิคการสร้างงานภาษาไทย 10. ตัวอย่าง Workflow แบบ End-to-End

ส่วนที่ 4 — การนำไปใช้และความปลอดภัย 11. แผนเริ่มต้น 30/60/90 วัน 12. Checklist ความปลอดภัยและการตรวจสอบ 13. ข้อจำกัด ข้อควรระวัง และภาคผนวก

⚠️ ข้อความสำคัญ: คู่มือนี้เป็นแนวทางที่จัดทำโดยใช้ AI ช่วยเรียบเรียง โปรดปรับให้ตรงกับระเบียบ มาตรฐาน และบริบทของหน่วยงานท่าน และตรวจสอบข้อมูลอ้างอิงกับฉบับจริงก่อนนำไปใช้ การออกแบบ/คำนวณในงานวิศวกรรมควบคุม และเอกสารทางการทุกฉบับ ต้องผ่านการตรวจสอบและลงนามโดยผู้รับผิดชอบที่มีอำนาจตามกฎหมาย

1. ภาพรวมและปรัชญาการใช้งาน

Claude คืออะไรในบริบทงานของเรา

Claude คือ ผู้ช่วยทางวิศวกรรมและธุรกิจที่ทำงานกับ "ข้อความ ตาราง เอกสาร และโค้ด" ได้เร็วมาก แต่ไม่ใช่ ผู้รับผิดชอบทางวิชาชีพ ไม่ใช่ผู้ลงนามรับรอง และไม่ใช่ฐานข้อมูลมาตรฐานที่เชื่อถือได้ 100%

หลักคิดที่ถูกต้อง:

มอง Claude เป็น...	อย่ามอง Claude เป็น...
ผู้ช่วยถอดแบบ/ร่างเอกสาร/สรุปข้อมูล	วิศวกรผู้ออกแบบที่รับผิดชอบตามกฎหมาย
เครื่องมือเร่งความเร็ว "ร่างแรก" (first draft)	ผู้ผลิต "ฉบับสมบูรณ์" ที่ส่งได้ทันทีโดยไม่ตรวจ
ผู้ช่วยตรวจทาน/หาความผิดปกติ	ผู้ตัดสินใจขั้นสุดท้าย
ผู้เขียนสคริปต์คำนวณที่ "ตรวจสอบได้"	เครื่องคิดเลขในหัวที่เชื่อตัวเลขได้เลย

กฎเหล็ก 3 ข้อ (Golden Rules)


- Human-in-the-loop เสมอ** — ทุกผลลัพธ์ที่มีผลทางวิศวกรรม กฎหมาย หรือการเงิน ต้องมีวิศวกร/เจ้าหน้าที่ผู้รับผิดชอบตรวจสอบและลงนาม Claude สร้าง "ร่าง" คนสร้าง "ความรับผิดชอบ"
- ตัวเลขต้องพิสูจน์ได้ ไม่ใช่เชื่อ** — อย่าเชื่อการคำนวณที่ Claude "คิดในหัว" ให้สั่งเสมอว่า "เขียนเป็นสูตร/โค้ด Python/ตาราง Excel ที่ตรวจสอบทีละขั้นได้" แล้วรันจริง วิธีนี้เปลี่ยนจาก "เชื่อ AI" เป็น "ตรวจสอบเครื่องมือ"
- ข้อมูลลับ ออกเฉพาะที่จำเป็น** — ข้อมูลชั้นความลับของทางราชการ ข้อมูลส่วนบุคคล (PDPA) ราคาากก่อนเปิดช่อง ต้องพิจารณาช่องทางที่เหมาะสม (ดูบทที่ 3 และ 12)

Automation ที่ "คุ้ม" หน้าตาเป็นอย่างไร

งานที่เหมาะสมกับการ automate มีลักษณะ ทำซ้ำบ่อย + มีรูปแบบชัด + ใช้เวลามาก + ตรวจสอบผลได้ เช่น

- สรุปเอกสาร/ถอดข้อมูลจากไฟล์จำนวนมาก
- แปลงข้อมูลดิบ (CSV/Excel) เป็นตาราง/กราฟ/รายงานตามรูปแบบเดิม ๆ
- ร่างเอกสารตามเทมเพลตมาตรฐาน (TOR, หนังสือราชการ, รายงานความก้าวหน้า)
- ตรวจสอบความสอดคล้อง/ความครบถ้วนของชุดเอกสาร

งานที่ไม่ควร automate แบบปล่อยอัตโนมัติ: การตัดสินใจออกแบบ การอนุมัติ การลงนาม การตีความระเบียบที่กำกวม — ใช้ Claude "ช่วยคิด" ได้ แต่คนตัดสินใจ

 **สรุปบทนี้:** ใช้ Claude เป็น "ผู้ช่วยที่ขยันและไม่เหนื่อย" เพื่อเร่งงานร่างและงานทำซ้ำ — ความถูกต้องและความรับผิดชอบยังเป็นของวิศวกรเสมอ

2. เครื่องมือ Claude และการเลือกใช้ให้เหมาะกับงาน

เครื่องมือ	เหมาะกับ	ระดับ	หมายเหตุ
Claude.ai (เว็บ/แอป)	ถาม-ตอบ ร่างเอกสาร วิเคราะห์ไฟล์เดี่ยว ทำกราฟ/ตาราง	no-code	จุดเริ่มต้นของทุกคน
Projects (ใน Claude.ai)	งานที่มี "บริบทถาวร" เช่น มาตรฐานหน่วยงาน เทมเพลต ศัพท์เฉพาะ	no-code	อัปเดตเครื่องมือ/เทมเพลตไว้ครั้งเดียว ใช้ซ้ำได้
Artifacts	สร้างเอกสาร/ตาราง/แดชบอร์ด HTML ที่แก้ไขสด ๆ	no-code	เหมาะทำแดชบอร์ดสรุปโครงการ
Skills	ความสามารถเฉพาะ เช่น สร้าง Word/Excel/PDF/PowerPoint	no-code + code	มี skill สำเร็จ: docx , xlsx , pptx , pdf
Claude Desktop + MCP	เชื่อม Claude กับไฟล์ในเครื่อง/ระบบงาน	กึ่ง	ต่อ MCP server เช่น filesystem, Google Drive
Claude Code (CLI)	automation จริง: ประมวลผลไฟล์จำนวนมาก เขียนสคริปต์ ทำซ้ำเป็นระบบ	code	เครื่องมือหลักของเล่นนี้ฝั่ง automation

เส้นทางการเติบโต (แนะนำ)

- ขั้นที่ 1 Claude.ai ทำงานทีละชิ้น → เรียนรู้คำสั่งอย่างไรได้ผลดี
- ↓
- ขั้นที่ 2 Projects + Skills → ผังบริบท/เทมเพลตหน่วยงาน ลดการพิมพ์ซ้ำ
- ↓
- ขั้นที่ 3 Claude Code → เปลี่ยนงานทำมือซ้ำ ๆ เป็นสคริปต์/ไปป์ไลน์
- ↓
- ขั้นที่ 4 ตั้งเวลา/เชื่อมระบบ → รายงานอัตโนมัติ, เชื่อม MCP, subagents

💡 ไม่จำเป็นต้องเข้าไป Claude Code ตั้งแต่วันแรก — คนส่วนใหญ่ได้ประโยชน์มหาศาลแค่จากขั้นที่ 1-2 ก่อน แล้วจึงขยับเมื่อเจองานทำซ้ำที่ชัดเจน

3. หลักการสำคัญสำหรับงานราชการ

3.1 ความถูกต้องและการอ้างอิงมาตรฐาน

- LLM อาจอ้างเลขมาตรา/เลขข้อมาตรฐานผิด (เช่น มยผ., วสท., มอก., มาตรฐาน ทล./ทช., AASHTO, ACI) — ต้องเปิดมาตรฐานฉบับจริงตรวจสอบทุกครั้ง ใช้ Claude ช่วย "หาว่าควรดูมาตรฐานอะไร หัวข้อไหน" แล้วไปยืนยันกับเอกสารจริง
- สำหรับการคำนวณวิศวกรรม ให้ Claude แสดงสมมติฐาน สูตร และหน่วยทุกชั้น และทำเป็นสเปรดชีต/โค้ดที่ตรวจสอบซ้ำได้
- เก็บ "ที่มาของข้อมูล" (source) ไว้เสมอเพื่อให้ตรวจสอบย้อนหลังได้ (auditability)

3.2 ความลับและความปลอดภัยของข้อมูล

- **อย่าป้อน** ข้อมูลชั้นความลับของทางราชการ ราคาगतก่อนประกาศ ข้อมูลส่วนบุคคล (เลขบัตรประชาชน, ข้อมูลผู้ร้องเรียน) เข้าสู่บริการสาธารณะโดยไม่จำเป็น
- หากต้องใช้ข้อมูลจริง: **ปกปิด/ทำให้ไม่ระบุตัวตน (anonymize/redact)** ก่อน หรือใช้ช่องทางที่หน่วยงานอนุมัติ (เช่น บัญชีองค์กร/Enterprise ที่มีข้อตกลงไม่นำข้อมูลไปฝึกโมเดล) — ตรวจสอบนโยบายความเป็นส่วนตัวและสัญญาฉบับที่หน่วยงานใช้
- อ้างอิงกรอบกฎหมาย: **พ.ร.บ.คุ้มครองข้อมูลส่วนบุคคล พ.ศ. 2562 (PDPA)** และ **ระเบียบว่าด้วยการรักษาความลับของทางราชการ พ.ศ. 2544**
- การประมวลผลในเครื่อง (Claude Code อ่านไฟล์ในเครื่อง) ช่วยลดการอัปโหลด แต่ "เนื้อหาที่ส่งให้โมเดลคิด" ยังออกไปประมวลผล — ให้ส่งเฉพาะส่วนที่จำเป็น

3.3 ความรับผิดชอบทางวิชาชีพและกฎหมาย

- งานออกแบบ/คำนวณในสาขาวิศวกรรมควบคุม ต้องมี **วิศวกรผู้ได้รับใบอนุญาต (สภาวิศวกร/COE)** เป็นผู้รับรองและลงนาม — Claude **ไม่สามารถ** แทนได้ และผลลัพธ์จาก Claude ไม่ใช่ "การออกแบบที่รับรองแล้ว"
- เอกสารทางการ (TOR, สัญญา, หนังสือราชการ) ที่ Claude ร่าง ต้องผ่านการตรวจของเจ้าหน้าที่ผู้รับผิดชอบและผู้มีอำนาจตามสายงานก่อนใช้
- การจัดซื้อจัดจ้างต้องเป็นไปตาม **พ.ร.บ.การจัดซื้อจัดจ้างและการบริหารพัสดุภาครัฐ พ.ศ. 2560** และระเบียบ/หนังสือเวียนกรมบัญชีกลาง — Claude ช่วยร่าง/ตรวจสอบความครบถ้วนได้ แต่ไม่ตีความแทนคณะกรรมการ

3.4 ความเป็นกลางและการตรวจทาน

- ระวัง อคติจากข้อมูลนำเข้า — ถ้าป้อนข้อมูลด้านเดียว คำตอบก็เอนเอียง ให้ขอ "ข้อโต้แย้ง/ความเสี่ยง/ทางเลือกอื่น" ประกอบเสมอ
- กำหนดให้ Claude บอกเมื่อไม่มั่นใจ และ แยกแยะ "ข้อเท็จจริงจากเอกสารที่ให้" ออกจาก "การอนุมานของ AI"

ข้อความปฏิเสธความรับผิดชอบ (วางท้ายเอกสารที่ AI ช่วยร่างได้): "เอกสารฉบับนี้จัดทำร่างโดยใช้เครื่องมือ AI ช่วย และผ่านการตรวจสอบ/รับรองโดยเจ้าหน้าที่ผู้รับผิดชอบแล้ว ข้อมูลเชิงเทคนิคและการคำนวณได้รับการทวนสอบกับมาตรฐานที่เกี่ยวข้อง"

4. งานวิศวกรรมโยธา

แต่ละ use case กำกับระดับไว้: ● = no-code ทำได้เลย · ● = เหมาะกับ Claude Code (automation)

4.1 ถอดแบบและตรวจสอบ BOQ / ประมาณราคา

- **ตรวจ BOQ หาความผิดปกติ** — อัปโหลด BOQ (Excel) ให้ Claude หา: รายการที่ปริมาณ/ราคา ผิดปกติ, รายการตกหล่นเมื่อเทียบกับแบบ, หน่วยไม่สอดคล้อง, ผลรวมที่ไม่ตรง
- **เทียบ BOQ กับราคากลาง** — วางตารางราคากลาง (เช่น จากหลักเกณฑ์ของกรมบัญชีกลาง/ ฐานราคา) แล้วให้เทียบรายการต่อรายการ พร้อมคำนวณส่วนต่าง % และไฮไลต์รายการที่เกินเกณฑ์
- **รวม BOQ หลายสลิปไฟล์** — Claude Code อ่าน Excel หลายสัญญา/หลายหมวดงาน รวมเป็น ตารางเดียว สรุปยอดตามหมวด สร้างรายงานเปรียบเทียบอัตโนมัติ
- **คำนวณ Factor F / ปรับราคา** — เขียนสคริปต์คำนวณตามสูตร แสดงทุกขั้นให้ตรวจ

⚠ ราคากลาง/ตัวเลขประมาณการต้องผ่านเจ้าหน้าที่พัสดุและคณะกรรมการกำหนดราคากลาง ตามระเบียบ — Claude เป็นเครื่องมือช่วยตรวจ ไม่ใช่ผู้กำหนด

4.2 ตรวจสอบแบบและข้อกำหนด (Spec)

- **หาความขัดแย้งระหว่างแบบกับสเปก** — วางสเปกและรายการแบบ ให้หาจุดที่ขัดกัน (เช่น กำลังอัดคอนกรีตในแบบ ≠ ในสเปก, ชั้นทางในแบบตัด ≠ ในรายการประกอบแบบ)
- **สรุปข้อกำหนดเฉพาะ** — ย่อสเปกยาว ๆ เป็นเช็คลิสต์ที่ผู้ควบคุมงานใช้ตรวจหน้างานได้
- **ตรวจชุดแบบจำนวนมาก** — Claude Code + skill pdf อ่านแบบ PDF หลายแผ่น ดึงตาราง เหล็กเสริม/รายการวัสดุ จัดทำสรุบบัญแบบอัตโนมัติ

4.3 ช่วยตรวจการคำนวณเชิงวิศวกรรม

- **ทวนสอบการคำนวณ** — ให้ Claude ตรวจการคำนวณโครงสร้าง/ระบายน้ำ/ดินที่เราทำมาแล้ว โดย "คำนวณซ้ำอิสระ" แล้วเทียบผล หาจุดที่ต่าง
- **สคริปต์คำนวณซ้ำได้** — เช่น คำนวณปริมาณน้ำหลาก (rational method), ออกแบบท่อระบาย น้ำ, ตรวจ load combination — เขียนเป็น Python ที่เปลี่ยน input แล้วรันใหม่ได้ พร้อมหมายเหตุ อ้างมาตรฐาน

⚠️ **อย่าใช้ผลจาก Claude เป็นการออกแบบขั้นสุดท้าย** การออกแบบในงานวิศวกรรมควบคุม ต้องมีวิศวกรผู้ได้รับใบอนุญาตรับรอง ใช้ Claude เพื่อ "ตรวจซ้ำ/หาจุดพลาด/แรงงานคำนวณซ้ำ" เท่านั้น

4.4 การควบคุมงานและรายงานความก้าวหน้า

- **สรุปรายงานประจำวัน/สัปดาห์** — รวม site diary, สภาพอากาศ, จำนวนแรงงาน/เครื่องจักร เป็นรายงานรูปแบบมาตรฐาน
- **วิเคราะห์ความก้าวหน้าเทียบแผน (S-curve)** — วางแผนงาน vs ผลงานจริง ให้คำนวณ % ก้าวหน้า ความล่าช้า และร่างข้อเสนอเร่งรัด
- **แดชบอร์ดโครงการอัตโนมัติ** — Claude Code อ่านไฟล์ความก้าวหน้ารายงวด สร้างแดชบอร์ด HTML อัปเดตได้ทุกสัปดาห์

4.5 จัดการเอกสารโครงการ

- **ร่างหนังสือโต้ตอบ RFI/RFA, บันทึกขออนุมัติวัสดุ, รายงานปัญหาหน้างาน**
- **ค้นหา/จัดหมวดเอกสารสัญญาหลายร้อยไฟล์, ดึงเงื่อนไขสำคัญ (งวดงาน ค่าปรับ การรับประกัน) ออกเป็นตารางสรุป**

5. งานวิศวกรรมขนส่งและจราจร

● = no-code ทำได้เลย · ● = เหมาะกับ Claude Code (automation)

5.1 วิเคราะห์ข้อมูลปริมาณจราจร (Traffic Count)

- แปลงข้อมูลดิบเป็นรายงาน — อัปโหลด CSV/Excel ผลนับปริมาณจราจร ให้คำนวณ: ปริมาณรายชั่วโมง, ชั่วโมงเร่งด่วน (peak hour), ปริมาณต่อวัน, แปลงเป็น PCU, ทิศทางหลัก แล้วสร้างกราฟ
- ประมวลผลหลายสถานี/หลายวัน — Claude Code อ่านไฟล์นับจราจรจากหลายจุดสำรวจ คำนวณ AADT (ปรับค่าตามฤดูกาล/วันในสัปดาห์ถ้ามีปัจจัย) จัดทำตารางสรุปและกราฟต่อจุดอัตโนมัติ
- ทำความสะอาดข้อมูล — ตรวจสอบค่าผิดปกติ (เครื่องนับค้าง ค่ากระโดด ช่วงข้อมูลหาย) เติม/ตั้งธงข้อมูลที่น่าสงสัยอย่างมีเกณฑ์

5.2 ความจุถนนและระดับการให้บริการ (Capacity / LOS)

- จัดเรียงและคำนวณตามขั้นตอน — ช่วยจัดข้อมูลเข้าสูตร คำนวณ v/c ratio, ระดับการให้บริการ (LOS) ของช่วงถนน/ทางแยก ตามแนวทาง HCM โดยแสดงทุกขั้น
- วิเคราะห์ทางแยกสัญญาณไฟ — ช่วยตรวจ/เสนอ timing plan, คำนวณความล่าช้าเฉลี่ย, รอบสัญญาณ

⚠ ค่าพารามิเตอร์และวิธีตั้งอิงคู่มือ/มาตรฐานที่หน่วยงานใช้ (เช่น HCM ฉบับที่ใช้อ้างอิง, คู่มือของ ทล./สนข.) — ให้ตรวจค่ากับฉบับจริง

5.3 วิเคราะห์อุบัติเหตุและจุดเสี่ยง (Black Spot)

- จัดอันดับจุดเสี่ยง — วางข้อมูลอุบัติเหตุ (ตำแหน่ง ความถี่ ความรุนแรง) ให้จัดอันดับจุดอันตราย สรุปรูปแบบ/สาเหตุเด่น และเสนอมาตรการเบื้องต้น (วิศวกรรม/บังคับใช้/ให้ความรู้ — 3E)
- รวมข้อมูลหลายแหล่ง — Claude Code รวมข้อมูลอุบัติเหตุหลายปี/หลายระบบ จับคู่ตำแหน่งสรุปสถิติรายเส้นทาง สร้างตาราง/แผนภูมิจัดอันดับ

5.4 รายงานผลกระทบด้านการจราจร (Traffic Impact)

- ช่วยวางโครงสร้างรายงาน, สรุปข้อมูลปริมาณจราจรฐาน, ประมาณการจราจรที่เกิดจากโครงการ (trip generation), ร่างบทวิเคราะห์

- ● เตรียม input/แปลงผลลัพธ์จากแบบจำลองจราจร (เช่น O-D matrix, ผลจาก VISSIM/SUMO) เป็นตาราง/กราฟสำหรับรายงาน

5.5 งานนโยบาย/แผนและการมีส่วนร่วม

- ● สรุปผลการรับฟังความคิดเห็นประชาชน, จัดหมวดข้อกังวล, ร่างประเด็นตอบ
- ● ร่างสรุปผู้บริหาร (executive summary) จากรายงานการศึกษาความเหมาะสมที่ยาวมาก

6. งานราชการและเอกสารงานก่อสร้าง

ครอบคลุมงานของกรม/หน่วยงานที่เกี่ยวข้องกับการก่อสร้างทุกประเภท (เช่น กรมทางหลวง, กรมทางหลวงชนบท, กรมโยธาธิการและผังเมือง, กรมชลประทาน, การทางพิเศษฯ, อปท. ฯลฯ)

6.1 ร่างขอบเขตงาน (TOR) / ข้อกำหนดการจ้าง

- ร่าง TOR ตามโครงสร้างมาตรฐาน — ป้อนข้อมูลโครงการ (ขอบเขต งบประมาณ ระยะเวลา คุณสมบัติผู้ยื่น) ให้ร่าง TOR ตามหัวข้อมาตรฐาน พร้อมเว้นช่องสิ่งที่ต้องเติม
- ตรวจสอบ TOR หาช่องโหว่/ความเสี่ยง — หาเงื่อนไขที่อาจลืกล็อกสเปก ขัดระเบียบ กำกวม หรือ ตกหล่น (เกณฑ์ตรวจรับ การรับประกัน ค่าปรับ)

⚠ TOR ต้องผ่านคณะกรรมการจัดทำ TOR และเป็นไปตาม พ.ร.บ.จัดซื้อจัดจ้างฯ 2560 — Claude ช่วยร่าง/ตรวจ ไม่ใช่อนุมัติ

6.2 เอกสารประกวดราคา (e-bidding) และการพัสดุ

- ตรวจสอบสอดคล้องของชุดเอกสารประกวดราคา (ประกาศ-เอกสารเชิญชวน-TOR-แบบ-BQQ) ว่าวันที่ วงเงิน คุณสมบัติ ตรงกันทุกฉบับ
- จัดทำเช็คลิสต์เอกสารที่ผู้ยื่นข้อเสนอต้องส่ง และตรวจสอบความครบถ้วนของซองที่ยื่น (ช่วยกลั่นกรองเบื้องต้น)
- ตรวจสอบชุดเอกสารหลายโครงการพร้อมกัน หาเอกสารที่ขาด/วันที่ผิด/เลขที่ไม่ต่อเนื่อง

6.3 หนังสือราชการและงานสารบรรณ

- ร่างหนังสือภายนอก/ภายใน บันทึกข้อความ ตามรูปแบบระเบียบงานสารบรรณ
- ปรับระดับภาษา/น้ำเสียงให้เหมาะกับผู้รับ (เรียน/กราบเรียน คำลงท้าย)
- สร้างหนังสือเวียนจากเทมเพลต + ข้อมูลผู้รับหลายราย (mail merge แบบมีปรับ)

6.4 รายงานการประชุมและการติดตามงาน

- ถอดเทป/บันทึกการประชุม → สรุปวาระ มติ ผู้รับผิดชอบ กำหนดเสร็จ ในรูปแบบมาตรฐาน
- ติดตามมติที่ค้าง สรุปสถานะให้ที่ประชุมครั้งถัดไป
- รวบรวมงานหลายครั้งประชุม ติดตาม action item อัตโนมัติเป็นตารางเดียว

6.5 การบริหารสัญญาก่อสร้าง

- สรุปเงื่อนไขสัญญา (งวดงาน-งวดเงิน, ค่าปรับ, การรับประกัน, เงื่อนไขส่งมอบ, การแก้ไขสัญญา/งานเพิ่ม-ลด)
- ตรวจสอบการเบิกจ่ายงวดงานเทียบเงื่อนไข, คำนวณค่าปรับล่าช้า (แสดงวิธีคิด)
- ร่างหนังสือแจ้งเตือน/เร่งรัด/สงวนสิทธิ์ตามสัญญา

6.6 ตอบข้อหารือ ข้อร้องเรียน และเผยแพร่

- ร่างคำตอบข้อร้องเรียน/ข้อหารือประชาชน (ภาษาราชการสุภาพ ชัดเจน)
- แปลงรายงานเทคนิคเป็นข่าว/เนื้อหาเผยแพร่ที่ประชาชนเข้าใจง่าย
- ร่างคำของบประมาณ/เหตุผลความจำเป็นของโครงการ

7. ระดับ no-code: Claude.ai, Projects, Skills

7.1 เริ่มจาก Claude.ai

ขั้นพื้นฐานที่สุด: เปิด [claude.ai](#) → อัปโหลดไฟล์ → พิมพ์สั่งงานเป็นภาษาไทยได้เลย รองรับการอ่าน PDF, Word, Excel, รูปภาพ, CSV

เคล็ดลับ: สั่งให้แสดงผลเป็นตาราง/ดาวน์โหลดเป็นไฟล์ได้ เช่น "สรุปเป็นตาราง แล้วสร้างไฟล์ Excel ให้ดาวน์โหลด" (จะเรียก skill `xlsx` อัตโนมัติ)

7.2 Projects — ฝั่ง "บริบทหน่วยงาน" ไว้ใช้ซ้ำ

สร้าง Project แยกตามสายงาน แล้วอัปโหลด "ความรู้ถาวร" ไว้ครั้งเดียว เช่น:

ชื่อ Project	เอกสารที่ใส่ไว้ (Project Knowledge)
งานประมาณราคา	หลักเกณฑ์ราคากลาง, เทมเพลต BOQ, รายการมาตรฐานวัสดุ, Factor F
งาน TOR/พัสดุ	เทมเพลต TOR, checklist e-bidding, สรุประเบียบพัสดุที่ใช้บ่อย
งานจรรยา	คู่มือคำนวณ LOS ที่หน่วยงานใช้, รูปแบบรายงานมาตรฐาน, ค่าพารามิเตอร์
งานสารบรรณ	ระเบียบงานสารบรรณ, เทมเพลตหนังสือ, ตัวอย่างหนังสือที่ดี

จากนั้นทุกแชตใน Project จะ "รู้บริบท" โดยไม่ต้องอัปโหลดซ้ำ และตอบตามมาตรฐานของหน่วยงานเรา

ตั้ง Custom Instructions ของ Project (ตัวอย่างวางได้เลย):

คุณเป็นผู้ช่วยวิศวกรในหน่วยงานราชการไทยด้านงานก่อสร้าง

- ตอบเป็นภาษาไทยราชการที่สุภาพ กระชับ ตรวจสอบได้
- ทุกการคำนวณ ให้แสดงสมมติฐาน สูตร หน่วย และขั้นตอน อย่ายัดตัวเลขในหัว
- เมื่ออ้างมาตรฐาน (มยผ./วสท./ทล./ทช./AASHTO ฯลฯ) ให้ระบุว่าควรตรวจสอบฉบับจริง และอย่าเดาเลขข้อ/มาตราถ้าไม่แน่ใจ
- แยก "ข้อเท็จจริงจากเอกสารที่ให้" ออกจาก "ข้อสันนิษฐานของ AI" ให้ชัด
- ถ้าข้อมูลไม่พอ ให้ถามก่อน อย่าเดา

7.3 Skills สำเร็จรูปที่ใช้ได้ทันที

- `xlsx` — สร้าง/แก้ไขไฟล์ Excel มีสูตรและกราฟ
- `docx` — สร้างเอกสาร Word มีหัวกระดาษ สารบัญ เลขหน้า ตาราง (เหมาะหนังสือราชการ/รายงาน)
- `pptx` — สร้างสไลด์นำเสนอ
- `pdf` — อ่าน/รวม/แยก/กรอกฟอร์ม/ทำ OCR ไฟล์ PDF

เรียกใช้โดยพิมพ์งานเป็นภาษาไทยตรง ๆ เช่น "จากข้อมูลนี้ ทำรายงานความก้าวหน้าเป็นไฟล์ Word มี ตารางและหัวกระดาษหน่วยงาน"

8. ระดับ automation: Claude Code

Claude Code คือ CLI ที่ทำงานกับไฟล์ในเครื่องคุณได้โดยตรง — เหมาะกับงานที่ **ทำซ้ำกับไฟล์จำนวนมาก** หรือต้องการ **ไปป์ไลน์ที่รันใหม่ได้เรื่อย ๆ** (เว็บแอป/PDF ที่คุณกำลังอ่านอยู่นี้ ก็สร้างด้วย Claude Code)

8.1 จัดโครงสร้างโฟลเดอร์โครงการ

```
โครงการ/
├─ CLAUDE.md          ← บทบรรณาธิการ: มาตรฐาน คัดพท์ ที่ตั้งไฟล์ กฎการทำงาน
├─ ข้อมูลดิบ/        ← ไฟล์ต้นทาง (BOQ, traffic count, แบบ)
├─ เทมเพลต/         ← เทมเพลต TOR หนังสือราชการ รายงาน
├─ สคริปต์/          ← สคริปต์ Python ที่ Claude เขียน (ตรวจแล้วเก็บไว้ใช้ซ้ำ)
├─ ผลลัพธ์/          ← รายงาน/ตารางที่ผลิตออกมา
├─ .claude/
│   └─ skills/        ← skill เฉพาะหน่วยงาน (เช่น ตรวจ BOQ)
│       └─ settings.json ← สิทธิ์/การตั้งค่า
```

8.2 CLAUDE.md — หัวใจของ automation

ไฟล์นี้คือ "ความจำถาวร" ที่ Claude Code อ่านทุกครั้ง ตัวอย่างเริ่มต้น (ปรับให้ตรงหน่วยงาน):

```
# บทบรรณาธิการ
หน่วยงาน: [กรม/สำนัก...] งานหลัก: งานก่อสร้าง/ออกแบบ/ควบคุมงาน

## ภาษาและรูปแบบ
- ตอบและเขียนเอกสารเป็นภาษาไทยราชการ สุภาพ กระชับ
- หนังสือราชการให้อิงระเบียบงานสารบรรณ

## กฎการคำนวณ (สำคัญ)
- ห้ามคำนวณตัวเลขในหัว ให้เขียนเป็นสคริปต์ Python แล้วรันจริงเสมอ
- แสดงสมมติฐาน สูตร หน่วย ที่มาข้อมูล และผลทุกขั้น
- สร้างไฟล์ผลลัพธ์ที่เปิดตรวจได้ (Excel/CSV) ไม่ใช่แค่ตอบในแชต

## มาตรฐานที่อ้างอิงบ่อย
- โครงสร้าง/วัสดุ: มยผ., วสท., มอก. → ต้องให้คนตรวจกับฉบับจริง
- งานทาง: มาตรฐาน ทล. (กรมทางหลวง) / ทช. (กรมทางหลวงชนบท), AASHTO
- จราจร: HCM ฉบับ [...], คู่มือ [...]
- วัสดุ: พ.ร.บ.จัดซื้อจัดจ้างฯ 2560 + หนังสือเวียนกรมบัญชีกลาง

## ที่ตั้งไฟล์
- BOQ อยู่ใน ./ข้อมูลดิบ/boq/ · traffic count อยู่ใน ./ข้อมูลดิบ/traffic/
- เทมเพลตหนังสืออยู่ใน ./เทมเพลต/

## ความปลอดภัย
- อย่าใส่ข้อมูลส่วนบุคคล/ชั้นความลับลงในผลลัพธ์ที่จะเผยแพร่
- ทุกเอกสารทางการต้องมีหมายเหตุว่า "ร่างโดย AI ช่วย ต้องตรวจ/ลงนามโดยผู้รับผิดชอบ"
```

8.3 ตัวอย่างคำสั่ง automation จริง

เปิด Claude Code ในโฟลเดอร์โครงการ แล้วสั่งเป็นภาษาไทยได้เลย เช่น:

```
อ่านไฟล์ Excel BOQ ทุกไฟล์ในโฟลเดอร์ ./ข้อมูลดิบ/boq/
- รวมเป็นตารางเดียว แยกตามหมวดงาน
- คำนวณยอดรวมแต่ละหมวดและยอดรวมทั้งโครงการ
- ตั้งธงรายการที่ราคาต่อหน่วยสูงกว่าค่ามาตรฐานของหมวดนั้นเกิน 50%
- เขียนเป็นสคริปต์ Python ที่รันใหม่ได้ และส่งออกผลเป็น Excel ในโฟลเดอร์ ./ผลลัพธ์/
- อยุ่คำนวณในหัว ให้รันสคริปต์จริงและแสดงผลที่ได้
```

```
อ่านไฟล์นับปริมาณจราจรทุกไฟล์ใน ./ข้อมูลดิบ/traffic/ (CSV)
- คำนวณปริมาณรายชั่วโมง หา peak hour เข้า/เย็น และปริมาณรวมต่อวันของแต่ละสถานี
- แปลงเป็น PCU ตามค่าสัมประสิทธิ์ที่ระบุใน CLAUDE.md
- ทำกราฟปริมาณรายชั่วโมงต่อสถานี และตารางสรุปเปรียบเทียบทุกสถานี
- ตั้งธงช่วงข้อมูลที่น่าสงสัย (ค่าหาย/ค่ากระโดดผิดปกติ)
- ส่งออกเป็นรายงาน Excel + ไฟล์กราฟ
```

8.4 ความสามารถขั้นสูง

- **Subagents / Workflow** — แดกงานใหญ่เป็นหลายตัวช่วยทำขนาน เช่น ตรวจสอบเอกสาร 50 สัญญาพร้อมกัน
- **Skills เฉพาะหน่วยงาน** — สร้าง skill ของเราเอง เช่น "ตรวจ BOQ ตามเกณฑ์กรม"
- **MCP (Model Context Protocol)** — เชื่อม Claude กับระบบงาน เช่น filesystem, Google Drive, ฐานข้อมูล
- **งานตามเวลา (scheduled)** — ตั้งให้ผลิตรายงานสรุพอัตโนมัติเป็นรอบ (เช่น สรุปความก้าวหน้าโครงการทุกเช้าวันจันทร์)
- **Deploy เว็บแอป** — สร้างเครื่องมือ/แดชบอร์ดเป็นไฟล์ static แล้วนำขึ้น Cloudflare Pages / Netlify ได้ลิงก์สาธารณะใน 5-10 นาที (เว็บนี้คือตัวอย่าง)

⚠️ ก่อนเชื่อม MCP กับระบบ/ข้อมูลจริงของราชการ ต้องผ่านการพิจารณาด้านความมั่นคงปลอดภัยสารสนเทศและสิทธิการเข้าถึงตามระเบียบหน่วยงาน

9. คลัง Prompt และเทคนิคการสั่งงานภาษาไทย

9.1 โครงสร้าง prompt ที่ได้ผล (สูตร R-T-F-C)

1. R (Role/บทบาท) — "คุณเป็นวิศวกรประมาณราคาในกรม..."
2. T (Task/งาน) — สั่งให้ทำอะไร ชัด ๆ ทีละข้อ
3. F (Format/รูปแบบผล) — ตาราง / ไฟล์ Excel / หนังสือราชการ
4. C (Constraints/ข้อจำกัด) — อ้างมาตรฐานอะไร ห้ามเขา ให้แสดงวิธีคิด ถ้าข้อมูลไม่พอ

9.2 เทคนิคสำคัญ

- ขอให้ "ถามก่อนทำ" เมื่อข้อมูลไม่ครบ — ใส่ท้าย prompt: "ถ้าข้อมูลไม่พอให้ถามก่อน อย่าสมมติเอง"
- บังคับแสดงวิธีคิด — "แสดงสมมติฐาน สูตร และทุกขั้นตอนการคำนวณ"
- ขอความเห็นต่าง — "ชี้จุดเสี่ยง/ข้อโต้แย้ง/สิ่งที่อาจตกหล่นด้วย"
- แยกข้อเท็จจริงกับการเดา — "ทำเครื่องหมายส่วนที่เป็นการอนุมานของคุณ"
- ให้ตัวอย่าง — แปะตัวอย่างหนังสือ/รายงานที่ดี 1 ฉบับ คุณภาพผลลัพธ์จะดีขึ้นมาก
- ทำซ้ำเป็นรอบ — เริ่มจากโครงร่าง → ตรวจ → ค่อยให้ลงรายละเอียด

9.3 ตัวอย่าง Prompt พร้อมใช้ (คัดลอกไปปรับได้)

ตรวจ BOQ

คุณเป็นวิศวกรประมาณราคา ช่วยตรวจไฟล์ BOQ ที่แนบ:

- 1) หารายการที่ปริมาณหรือราคาต่อหน่วยผิดปกติ พร้อมเหตุผล
- 2) หารายการที่น่าจะตกหล่นเมื่อเทียบกับขอบเขตงานนี้: [อธิบายงาน]
- 3) ตรวจผลรวมแต่ละหมวดและยอดรวมว่าถูกต้องไหม

แสดงผลเป็นตาราง: รายการ | ประเด็น | ระดับความเสี่ยง | ข้อเสนอแนะ

ถ้าข้อมูลไม่พอให้ถามก่อน และอย่าเดาราคามาตรฐานเอง

ร่าง TOR

ช่วยร่าง TOR งานจ้าง[ก่อสร้าง/ออกแบบ/ที่ปรึกษา] ตามข้อมูลนี้:

- ชื่อโครงการ/ขอบเขต: ...
- งบประมาณ/ระยะเวลา: ...
- คุณสมบัติผู้ยื่นข้อเสนอ: ...

ใช้โครงสร้างหัวข้อ TOR มาตรฐานงานราชการ เว้นช่อง [ต้องเติม] ในส่วนที่ข้อมูลไม่พอ

ท้ายเอกสารให้สรุป "ประเด็นที่คณะกรรมการควรพิจารณาเพิ่ม" และ "จุดเสี่ยงที่อาจขัดระเบียบ/ลืกละเลย"

หมายเหตุ: นี่เป็นร่างที่ต้องผ่านคณะกรรมการและตรวจตาม พ.ร.บ.จัดซื้อจัดจ้างฯ 2560

สรุปรายงานการประชุม

จากบันทึก/ถอดเทปที่แนบ ช่วยสรุปรายงานการประชุมแบบราชการ:

- หัวข้อ: วาระ |สาระสำคัญ | มติที่ประชุม | ผู้รับผิดชอบ | กำหนดแล้วเสร็จ
 - แยกส่วน "เรื่องติดตามจากครั้งก่อน" ออกมา
 - ทำเป็นไฟล์ Word ตามรูปแบบรายงานการประชุม
- ถ้าตรงไหนในเทปไม่ชัด ให้ทำเครื่องหมาย [ตรวจสอบ] ไว้

วิเคราะห์ traffic count

จากไฟล์ผลนับปริมาณจราจรที่แนบ ช่วย:

- 1) คำนวณปริมาณรายชั่วโมง หาชั่วโมงเร่งด่วนเช้า/เย็น และปริมาณรวมต่อวัน
 - 2) แยกตามทิศทางและประเภทยานพาหนะ แปลงเป็น PCU (ใช้ค่า: รถยนต์=1, รถบรรทุก=..., จยย.=...)
 - 3) ทำกราฟและตารางสรุป และตั้งธงข้อมูลที่ผิดปกติ
- เขียนเป็นสคริปต์ Python รันจริง แล้วส่งออกเป็น Excel – อย่าคำนวณในหัว

ร่างหนังสือราชการ

ช่วยร่าง[หนังสือภายนอก/บันทึกข้อความ] เรื่อง: ...

- ผู้รับ: ... · ผู้ส่ง/หน่วยงาน: ...
- สาระที่ต้องสื่อ: ...
- โทณ: ทาการ สุภาพ กระชับ ตามระเบียบงานสารบรรณ

ร่างให้ครบรูปแบบ (เลขที่ วันที่ เรื่อง เรียง อ้างถึง สิ่งที่มาด้วย เนื้อหา ย่อหน้าสรุป)
เว้นช่อง [_____] ตรงที่ต้องเติมข้อมูลจริง

10. ตัวอย่าง Workflow แบบ End-to-End

● = no-code · ● = Claude Code (automation)

Workflow A — รายงานเปรียบเทียบ BOQ หลายสัญญา ●

1. รวบรวมไฟล์ BOQ ของทุกสัญญาไว้ใน ./ข้อมูลดิบ/boq/
2. สั่ง Claude Code: "อ่านทุกไฟล์ รวบรวมค่า คำนวณยอดตามหมวด เทียบราคาต่อหน่วย ข้ามสัญญา ตั้งธงรายการผิดปกติ เขียนสคริปต์ที่รันซ้ำได้ ส่งออก Excel + สรุป"
3. Claude เขียนสคริปต์ Python + รัน + ผลิตไฟล์ใน ./ผลลัพธ์/
4. วิศวกรเปิดไฟล์ตรวจ ปรับเกณฑ์ สั่งรันใหม่ได้ทันที
5. เก็บสคริปต์ไว้ใช้กับโครงการหน้า (แค่เปลี่ยนไฟล์ต้นทาง)

ได้อะไร: งานที่เคยทำมือเป็นวัน เหลือหลักนาที + ตรวจสอบย้อนได้เพราะมีสคริปต์

Workflow B — ชุดเอกสารประกวดราคา ● → ●

1. (no-code) ใน Project "งาน TOR/พัสดุ" ร่าง TOR จากข้อมูลโครงการ
2. (no-code) ให้ตรวจสอบสอดคล้อง ประกาศ-TOR-BOQ-แบบ (วันที่/วงเงิน/คุณสมบัติ)
3. (no-code) สร้าง checklist เอกสารที่ผู้ยื่นต้องส่ง เป็นไฟล์ Word
4. (Claude Code) เมื่อมีหลายโครงการ ให้ตรวจชุดเอกสารทั้งหมดทีเดียว หาเอกสารขาด/เลขผิด
5. เจ้าหน้าที่พัสดุตรวจ → คณะกรรมการพิจารณา → เผยแพร่ผ่าน e-GP

Workflow C — รายงานความก้าวหน้าโครงการรายสัปดาห์ ●

1. site diary + ผลงานรายงวด + แผนงาน → โฟลเดอร์ข้อมูลดิบ
2. Claude Code: คำนวณ % ก้าวหน้าเทียบแผน (S-curve) วิเคราะห์ความล่าช้า สร้างแดชบอร์ด HTML + สรุปผู้บริหาร
3. (ขั้นสูง) ตั้งให้รันอัตโนมัติทุกเช้าวันจันทร์
4. ผู้ควบคุมงานตรวจ ปรับ ส่งต่อผู้บริหาร

Workflow D — วิเคราะห์จุดเสี่ยงอุบัติเหตุประจำปี ●

1. ข้อมูลอุบัติเหตุหลายปี (หลายไฟล์/หลายรูปแบบ) → โฟลเดอร์เดียว
2. Claude Code: ทำความสะอาด รวม จับคู่ตำแหน่ง จัดอันดับ black spot สรุปรูปแบบ/สาเหตุ เสนอมาตรการ 3E เบื้องต้น
3. ส่งออกตารางจัดอันดับ + แผนภูมิ + ร่างรายงาน
4. วิศวกรจรรยาตรวจ ลงพื้นที่ยืนยัน จัดทำแผนปรับปรุง

💡 ทุก workflow จบที่ "คนตรวจและรับผิดชอบ" เสมอ — automation เร่งงานช่วงกลาง ไม่ใช่แทนการตัดสินใจช่วงท้าย

11. แผนเริ่มต้น 30/60/90 วัน

30 วันแรก — สร้างความคุ้นเคย (no-code)

- ทุกคนในทีมเริ่มใช้ Claude.ai กับงานเอกสารจริง 1 งาน/สัปดาห์
- เลือก "งานเจ็บปวดที่ทำซ้ำบ่อย" มา 2-3 งานเป็น pilot
- ตั้ง Project พร้อม Custom Instructions + อัปเดตเทมเพลต/มาตรฐานของหน่วยงาน
- ทำคลัง prompt ของทีม (เก็บ prompt ที่ได้ผลดีไว้ใช้ซ้ำ)

60 วัน — ขยายผลและฝังในกระบวนการ

- กำหนด "มาตรฐานการตรวจทาน" ผลลัพธ์จาก AI ก่อนใช้งานจริง
- เริ่มใช้ Skills (docx/xlsx/pdf) ในงานประจำ
- นำร่อง Claude Code กับ 1 workflow ที่ทำซ้ำกับไฟล์จำนวนมาก
- จัดทำ CLAUDE.md ของหน่วยงาน

90 วัน — เป็นระบบ

- มี 3-5 workflow ที่ใช้เป็นประจำ พร้อมสคริปต์ที่ตรวจแล้ว
- กำหนดนโยบายความปลอดภัยข้อมูล/ความเป็นส่วนตัวเป็นลายลักษณ์อักษร
- ประเมินผล: ประหยัดเวลาเท่าไร คุณภาพดีขึ้นไหม จุดเสี่ยงที่พบ
- ถ่ายทอด/อบรมขยายผลสู่ทีมอื่น

💡 เคล็ดลับการขยายผลในองค์กรราชการ: เริ่มจาก "งานที่เห็นผลเร็วและความเสี่ยงต่ำ" (เช่น สรุปเอกสาร ร่างหนังสือ) เพื่อสร้างความเชื่อมั่น ก่อนขยายไปงานที่ละเอียดอ่อนกว่า

12. Checklist ความปลอดภัยและการตรวจสอบ

ก่อนป้อนข้อมูล

- ข้อมูลนี้เป็นชั้นความลับหรือข้อมูลส่วนบุคคลหรือไม่? ถ้าใช่ → ปกปิด/ใช้ช่องทางที่อนุมัติ
- เป็นราคากลาง/ข้อมูลอ่อนไหวก่อนเปิดเผยหรือไม่?

ก่อนใช้ผลลัพธ์

- ตัวเลข/การคำนวณตรวจซ้ำด้วยเครื่องมือจริงแล้วหรือยัง?
- มาตรฐาน/เลขข้อที่อ้าง ตรวจสอบกับฉบับจริงแล้วหรือยัง?
- มีผู้รับผิดชอบ/ผู้มีอำนาจตรวจและลงนามหรือยัง?
- เอกสารทางการมีหมายเหตุการใช้ AI ช่วยร่าง + ผ่านการตรวจหรือยัง?
- งานออกแบบ/คำนวณวิศวกรรมควบคุม มีวิศวกรผู้ได้รับใบอนุญาตรับรองหรือยัง?

การเก็บร่องรอย (auditability)

- เก็บ prompt + ที่มาข้อมูล + เวอร์ชันผลลัพธ์ เพื่อให้ตรวจสอบย้อนหลังได้

ก่อนเผยแพร่เอกสารสาธารณะ (เช่น เว็บไซต์/เอกสารออนไลน์)

- เนื้อหาไม่มีข้อมูลลับ/ข้อมูลส่วนบุคคล/ข้อมูลภายในที่ยังไม่เปิดเผย
- เครื่องมือที่รับข้อมูลผู้ใช้ เป็นแบบ "ประมวลผลในเครื่องผู้ใช้" (ไม่อัปโหลดขึ้นเซิร์ฟเวอร์) หรือไม่
- การเผยแพร่ในนามหน่วยงาน ผ่านการขออนุญาต/ตรวจนโยบายแล้ว

13. ข้อจำกัด ข้อควรระวัง และภาคผนวก

13.1 ตารางข้อกำหนดและวิธีรับมือ

ข้อกำหนด	ผลกระทบ	วิธีรับมือ
ค่านวณเลขในหัวอาจผิด	ตัวเลขประมาณราคา/วิศวกรรมคลาดเคลื่อน	บังคับเขียนสคริปต์/สูตรแล้วรันจริง ตรวจสอบซ้ำ
อ้างมาตรฐานผิด (hallucination)	อ้างเลขข้อ/ค่าผิด	ตรวจกับมาตรฐานฉบับจริงเสมอ ห้ามใช้คำอ้างลอย ๆ
ไม่รู้ข้อมูลล่าสุด	ระเบียบ/ราคาที่เปลี่ยนใหม่อาจไม่รู้	ป้อนเอกสารฉบับปัจจุบันให้ + ตรวจสอบวันที่
ไม่รับผิดชอบทางกฎหมาย	ไม่ใช่ผู้ลงนาม/รับรอง	คนต้องตรวจและรับผิดชอบเสมอ
ความลับข้อมูล	เสี่ยงเปิดเผยข้อมูลอ่อนไหว	ปกปิดข้อมูล + ใช้ช่องทางที่หน่วยงานอนุมัติ
อาจเอนเอียงตาม input	วิเคราะห์ด้านเดียว	ขอความเห็นต่าง/ความเสี่ยงประกอบเสมอ

13.2 สรุปหลักคิดสุดท้าย

Claude เร่งความเร็ว "ร่างแรก" และงานทำซ้ำได้มหาศาล — แต่ความถูกต้อง ความรับผิดชอบ และการตัดสินใจ ยังเป็นของวิศวกรและเจ้าหน้าที่เสมอ

ใช้เป็น "ผู้ช่วยที่ขยันและไม่เหน็ดเหนื่อย" ไม่ใช่ "ผู้เชี่ยวชาญที่เชื่อได้โดยไม่ตรวจสอบ"

13.3 ภาคผนวก — แหล่งอ้างอิงและมาตรฐานที่เกี่ยวข้อง

ตรวจสอบฉบับล่าสุดเสมอ

- พ.ร.บ.การจัดซื้อจัดจ้างและการบริหารพัสดุภาครัฐ พ.ศ. 2560 + ระเบียบ/หนังสือเวียนกรมบัญชีกลาง
- หลักเกณฑ์การคำนวณราคากลางงานก่อสร้างของทางราชการ (กรมบัญชีกลาง)
- พ.ร.บ.คุ้มครองข้อมูลส่วนบุคคล พ.ศ. 2562 (PDPA)
- ระเบียบว่าด้วยการรักษาความลับของทางราชการ พ.ศ. 2544
- ระเบียบสำนักนายกรัฐมนตรีว่าด้วยงานสารบรรณ
- มาตรฐานวิชาชีพ/ใบอนุญาต: สภาวิศวกร (COE)
- มาตรฐานทางเทคนิค: มยผ. (กรมโยธาธิการและผังเมือง), วสท. (วิศวกรรมสถานแห่งประเทศไทย), มอก., มาตรฐาน ทล. (กรมทางหลวง), ทช. (กรมทางหลวงชนบท), AASHTO, ACI, HCM

— จบเอกสาร —

หมายเหตุ: เอกสารนี้เป็นแนวทาง/ร่างที่จัดทำโดยใช้ AI ช่วย โปรดปรับให้ตรงกับระเบียบ
มาตรฐาน และบริบทของหน่วยงานท่าน และตรวจสอบข้อมูลอ้างอิงกับฉบับจริงก่อนนำไปใช้