PMBOK 8-ALIGNED · PROJECT DELIVERY

PM Delivery Guide (Ver.2)

Project Delivery Playbook + PM Masterclass — ฉบับผู้เรียน (Learner) (PMBOK 8-aligned)

Learner Edition

อ้างอิง PMBOK 8th Edition (Principles / Domains / Focus Areas) ประกอบ Playbook V2

เอกสารนี้ไม่ใช่เอกสารทางการของ PMI — This is not official PMI material.

สารบัญ / Table of Contents

Ch.00 — PMBOK Primer — กรอบคิดสำหรับเล่มนี้

Ch.01 — Pre-sales — จาก Opportunity ถึง Proposal/SOW

Ch.02 — Initiation — Charter และ Stakeholder

Ch.03 — Requirements, Scope และ WBS

Ch.04 — Schedule, Cost และ Resource

Ch.05 — Risk, Procurement, Communications และ Quality Plan

Ch.06 — Execution — ทีม, คุณภาพ และการควบคุมประจำวัน

Ch.07 — Monitoring & Change — EVM และ Integrated Change Control

Ch.08 — Verification / UAT — Control Quality และ Release Readiness

Ch.09 — Go-Live & Hypercare

Ch.10 — Transition & Closure

Appendix-A-Artifact-Catalogue

Appendix-B-Golden-Rules-Poster

Appendix-C-PM-Glossary-Extended

Appendix-D-Master-Answer-Key

Appendix-E-SDLC-Role-Output-Matrix

Appendix-F-Problem-to-Chapter-Index

Chapter 00 — PMBOK Primer: ทำไมเล่มนี้เดินตาม A→H

1. Opening Scenario Hook

[Teaching Scenario] คุณเป็น PM คนใหม่ที่เพิ่งได้รับมอบหมายให้ดูแลโครงการ Direct Booking Platform ของ SHG — เพื่อนร่วมงานแนะนำให้คุณ "อ่าน PMBOK หน่อย" คุณเปิดหาแล้วเจอหลายฉบับ บางฉบับเป็น process 49 ข้อ บางฉบับเป็น principles 12 ข้อ บางฉบับเป็น domains — คำถามคือ: **ต้องรู้อะไรจริง ๆ ก่อนเริ่มงานหน้าแรก**

[PMBOK 8] บทนี้จะตั้งกรอบ: PMBOK คือ Body of Knowledge ไม่ใช่ methodology — และเล่มนี้จัดเรียงตาม **workflow จริงที่โครงการเดินผ่าน (A→H: Pre-sales → Closure)** แทนการเรียงตาม Knowledge Area หรือ Performance Domain เพราะเป้าหมายของเล่มนี้คือ "เปิดดูตอนมีปัญหาหน้างาน แล้วรู้ว่าต้องทำอะไร"

2. Why It Matters

[PMBOK 8] PMBOK เปลี่ยนจาก "ชุดกระบวนการที่ต้องทำครบ" เป็น "กรอบที่ช่วยให้ PM ตัดสินใจตามบริบท" — แต่การจะ tailor ได้ ต้องรู้จักส่วนประกอบก่อน: Principles (หลักคิด), Performance Domains (ด้านที่ต้องบริหาร), Focus Areas (จุดเน้น) และ Process Groups (วงจรบริหาร) — บทนี้ให้แผนที่ของส่วนประกอบเหล่านี้ เพื่อให้บทต่อ ๆ ไป (A→H) มีภาษากลาง

[Best Practice] ประสบการณ์สำคัญ แต่ประสบการณ์อย่างเดียวทำให้ PM มองเฉพาะสิ่งที่เคยเจอ — PMBOK คือแผนที่ของคำถามสำคัญ: Business Need ชัดไหม, Stakeholder ใคร, Scope/Change ควบคุมได้ไหม, Risk มี owner ไหม, และหลังส่งมอบใครวัด benefit

3. Mental Model

```
Business Need
  -> Value Delivery (ทำไมต้องทำ)
  -> Project Management (ทำอย่างไร)
       Principles — หลักคิดที่ยึดตลอด
       Performance Domains — ด้านที่ต้องบริหารให้ครบ
       Process Groups — Initiating -> Planning -> Executing -> M&C -> Closing
       Knowledge Areas — องค์ความรู้เฉพาะด้าน (เดิม)
  -> Workflow A→H ของเล่มนี้: Pre-sales -> Initiation -> Planning -> Execution
     -> Monitoring/Change -> Verification/UAT -> Go-Live/Hypercare -> Closure
```

[PMBOK 8] เล่มนี้เรียงตามแถวล่าง (A→H) เพราะเป็นลำดับเวลาที่งานเกิดจริง — แต่ทุกบทแฝง Principles, Domains และ Process Groups ไว้ข้างใน

4. Main Lesson

4.1 PMBOK คืออะไร — ไม่ใช่ methodology

[PMBOK 8] PMBOK (Project Management Body of Knowledge) คือมาตรฐานและองค์ความรู้ของวิชาชีพ — ไม่ใช่ methodology สำเร็จรูปที่ทุกโครงการต้องเดินเหมือนกัน เน้น: การสร้าง Value, การบริหารเป็นระบบ, การ Tailor, การ

ตัดสินใจบนข้อมูล, การบริหาร Stakeholder, การจัดการความไม่แน่นอน, การวัดผลและปรับตัว, และการส่งมอบ Outcome ไม่ใช่แค่ Output

4.2 องค์ประกอบของกรอบ PMBOK 8 (ตามที่เล่มนี้อ้างอิง)

[PMBOK 8] ภาพรวมระดับสูงที่เล่มนี้ใช้:

องค์ประกอบ	จำนวน	บทบาท
Principles	6	หลักคิดที่ยึดตลอดโครงการ (เช่น value focus, stakeholder stewardship, systems thinking)
Performance Domains	7	ด้านที่ต้องบริหารให้ครบ (เช่น stakeholders, team, development approach, delivery, planning, uncertainty, performance)
Focus Areas	5	จุดเน้นการปฏิบัติ (เช่น tailoring, governance, quality, risk, value)
Process Groups	5	Initiating → Planning → Executing → Monitoring & Controlling → Closing (จาก Process Groups: A Practice Guide)

> **หมายเหตุ:** รายละเอียดระดับหัวข้อของ PMBOK 8 เป็นไปตาม reference basis ของเล่มนี้ (Playbook V2 §1) — ตัวเลขข้างต้นเป็นกรอบที่เล่มนี้ใช้; เนื้อหาหลักของเล่มเน้น "วิธีใช้" มากกว่า "จำจำนวน"

4.3 Process Groups ไม่ใช่ Project Phases

[PMBOK 8] Planning, Executing และ Monitoring & Controlling เกิดซ้ำและทำงานร่วมกันตลอดโครงการ — ไม่ใช่ขั้นตอนที่เกิดครั้งเดียวแล้วจบ — เล่มนี้แยกเป็นช่วง A–H เพื่อให้เห็นงานจริง แต่ Process Group ทั้ง 5 ยังทำงานอยู่ข้างในทุกช่วง

4.4 ทำไมเล่มนี้เรียง A→H ไม่ใช่เรียงตาม Domain/KA

[Best Practice] Knowledge Area (เดิม) เรียงตามหัวข้อวิชา — เหมาะกับเรียน แต่เวลาเจอปัญหาหน้างาน ผู้อ่านไม่รู้ว่า "ต้องเปิดบทไหน" — Workflow A→H เรียงตามเวลาที่งานเกิดจริง: "พรุ่งนี้จะขึ้น Production → เปิด Ch.9", "ลูกค้าขอเพิ่ม scope → เปิด Ch.7" — เหมาะกับ field manual (ดู Appendix F สำหรับ Problem → Chapter index)

4.5 วิธีใช้เล่มนี้ — 2 โหมด

[Best Practice]

- **Study Mode:** อ่านเรียงบท Ch.0 → Ch.10 เพื่อเรียนจบเป็น PM
- **Field Mode:** ใช้ Appendix F (Problem → Chapter) + Quick Reference Card ท้ายบท + Appendix A (Artifact Catalogue) + Appendix E (Role/Output Matrix) เพื่อ lookup ตอนติดปัญหา

5. PM Decision Thinking

```
Decision: เมื่อเริ่มงานใหม่ ควรอ่าน/ใช้เล่มนี้แบบ Study Mode หรือ Field Mode (และจะ tailor ระดับไหน)
Owner: PM เอง (กับ Sponsor/PO สำหรับ governance ระดับโครงการ)
Inputs: ประสบการณ์, บริบทโครงการ, ความเสี่ยง, ความเร่งด่วน
Options:
  A: Study Mode เต็ม (อ่านเรียงบท) — เหมาะกับ PM ใหม่/โครงการแรก
  B: Field Mode (เปิดเฉพาะบทที่เจอปัญหา) — เหมาะกับ PM มีประสบการณ์/งานเร่ง
  C: ผสม (เรียนบทที่สำคัญ + lookup ตอนติด) — ปกติ (แนะนำ)
Trade-offs: ความลึก vs ความเร็ว, ความครบถ้วน vs เวลา
Risk: อ่านแต่ไม่ใช้ (ทฤษฎีลอย) หรือใช้แต่ไม่เข้าใจกรอบ (ปฏิบัติมืดบอด)
Evidence: สมุดบันทึกบทเรียนส่วนตัว, decision log ของโครงการ
Next Action: เลือกโหมด -> เริ่ม Ch.1 (Pre-sales) หรือเปิดบทที่ตรงกับปัญหา -> ใช้ Appendix เป็นเครื่องมือ
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario] โครงการ SHG ถูกใช้เป็น Scenario เดียวตลอดเล่ม — ตัวเลขล็อกที่ `scenarios/HOTEL-BOOKING-PLATFORM-CASE.md` v1.0: เครือโรงแรม 12 แห่ง, งบ Phase 1 = 12 ล้านบาท, เป้าหมาย Direct Booking 10% → 35% ใน 18 เดือน, launch ก่อน High Season (1 พ.ย.) — บทนี้แนะนำให้รู้จักตัวละครหลัก (คุณจิรา CEO/Sponsor, คุณสุทธิ PM, คุณนภา PO, คุณภัทร VP Marketing, คุณวีระ CTO) ก่อนเดินเรื่อง Ch.1–Ch.10

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. อ่าน PMBOK แล้วคิดว่าเป็น checklist ที่ต้องทำครบทุกข้อ — ผล: เอกสารเยอะแต่ไม่เกิดการตัดสินใจ
2. เลือก methodology จากกระแส ("ทุกโครงการต้อง Agile") — ผล: ไม่ตรงบริบท
3. ใช้เล่มนี้เป็นตำรา แต่ไม่เปิดตอนหน้างาน — ผล: ไม่ได้ใช้คุณค่า field manual
4. สับสน Process Group กับ Project Phase — ผล: วางแผนผิดจังหวะ

Common Misconceptions:

ความเข้าใจผิด	ความจริง
PMBOK = Waterfall	PMBOK คือ body of knowledge; approach ต้อง tailor
PMBOK ฉบับใหม่ทำให้ฉบับเก่าใช้ไม่ได้	ฉบับเก่ายังมีคุณค่าเชิงปฏิบัติ (e-Book เดิมยังอยู่)
PM = คนตามงาน	PM เชื่อม goal, people, decision, risk, change, value
Go-live = success	Go-live เป็น output; value ดูที่ outcome/benefit
รู้ชื่อ framework = เก่ง	ต้องรู้ว่าใช้กับบริบทไหนและควบคุมอะไร

8. Interview Questions

Foundational

- **Q:** PMBOK คืออะไร และไม่ใช่อะไร?
- **Answer Direction:** คือ Body of Knowledge/มาตรฐานของวิชาชีพ — ไม่ใช่ methodology, ไม่ใช่ checklist, ไม่รับประกันความสำเร็จ
- **Warning Signs:** ตอบว่า "คือคู่มือทำตามขั้นตอน"

Scenario

- **Q:** ทำไมเล่มนี้ถึงเรียง A→H แทนการเรียงตาม Knowledge Area?
- **Answer Direction:** เรียงตามเวลางานเกิดจริง → เปิดหาได้ตอนเจอปัญหา (field use) — KA เรียงตามหัวข้อวิชา เรียนง่ายแต่ lookup ยาก
- **Warning Signs:** ตอบว่า "ไม่มีเหตุผล แค่เปลี่ยนสไตล์"

Senior PM

- **Q:** คุณจะ tailor การใช้ PMBOK สำหรับโครงการขนาดเล็กอย่างไร?
- **Answer Direction:** ใช้ Minimum Practical Pack (Playbook §6): charter, stakeholder list, scope/backlog, timeline, RAID, acceptance, status, UAT, go-live checklist, handover — ตัดเอกสารที่ไม่สร้าง decision ออก แต่ไม่ตัด control
- **Warning Signs:** ตัดขั้นตอนที่ไม่อยากทำโดยไม่มีเหตุผล (Tailoring ≠ ตัดทิ้ง)

Executive

- **Q:** ทำไมองค์กรต้องมีภาษากลางเรื่อง PM แม้ใช้หลาย methodology?
- **Answer Direction:** ภาษากลาง (Scope, Risk, Change, Acceptance) ทำให้ทีม/ผู้บริหารสื่อสารกันรู้เรื่องและตัดสินใจบนข้อมูลเดียวกัน — ลดความเข้าใจผิดข้ามทีม
- **Warning Signs:** ตอบว่า "ไม่จำเป็น ถ้าจ้าง PM เก่ง ๆ"

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
PMBOK	องค์ความรู้การบริหารโครงการ	แผนที่และภาษากลางของวิชาชีพ	คิดว่าเป็น methodology	Framework
Framework	กรอบคิด	โครงช่วยถามให้ครบ ไม่ตัดสินแทนเรา	สับสนกับ Checklist	Checklist
Tailoring	การปรับให้เหมาะบริบท	เลือกวิธี/ระดับ control ตาม risk	คิดว่า = ตัดขั้นตอน	Predictive, Agile, Hybrid
Process Group	กลุ่มกระบวนการบริหาร	5 กลุ่ม เกิดซ้ำได้ตลอดโครงการ	สับสนกับ Phase	Phase

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Principle	หลักคิด	ยึดตลอดโครงการ	จดจำแต่ไม่ใช้	Domain
Performance Domain	ด้านการบริหาร	มิติที่ต้องดูแลให้ครบ	สับสนกับ Process	Principle
Output	ผลส่งมอบ	สิ่งที่ทีมสร้าง/ส่งให้ตรวจ	สับสนกับ Outcome	Outcome
Outcome	ผลการเปลี่ยนแปลง	สิ่งที่เกิดเมื่อ output ถูกใช้จริง	สับสนกับ Output	Benefit
Benefit	ประโยชน์ที่วัดได้	ตัวเลขที่ business ได้รับ	สับสนกับ Output	Business Value
Workflow A–H	ลำดับงานจริง	Pre-sales → Closure	คิดว่าเป็น Process Group	Process Group

10. Workshop

บทปฐมบทไม่มี Workshop — ตาม `Ver.2/master_plan.md` §5 (Ch.0 ยกเว้น Workshop/Assessment) — เริ่มฝึกจริงที่ Ch.1 เป็นต้นไป

11. Checklist ใช้งานจริง (เริ่มต้นอ่านเล่มนี้)

- ☐ เข้าใจว่า PMBOK = body of knowledge ไม่ใช่ methodology
- ☐ รู้จัก 2 โหมดอ่าน: Study Mode (เรียงบท) / Field Mode (lookup)
- ☐ รู้โครง A→H: Pre-sales, Initiation, Planning, Execution, Monitoring/Change, Verification/UAT, Go-Live/Hypercare, Closure
- ☐ รู้จัก Scenario หลัก (SHG): 12 โรงแรม, 12M, 35% ใน 18 เดือน, launch ก่อน 1 พ.ย.
- ☐ รู้จักเครื่องมือ Appendix: A (Artifact), E (Role/Output), F (Problem→Chapter)
- ☐ เตรียมสมุดบันทึกบทเรียน + decision log ของตัวเอง

12. Assessment

บทปฐมบทไม่มี Assessment — ตาม `Ver.2/master_plan.md` §5 — ประเมินความเข้าใจด้วยคำถามท้ายบทที่ 13 และ Checkpoint ในบทถัดไป

13. Executive Summary

มิติ	สรุป
Why	PMBOK คือแผนที่ของคำถามสำคัญ — ไม่ใช่สูตรสำเร็จ
Decision Improved	การเลือกใช้กรอบ/เครื่องมือถูกต้องตามบริบทและโหมดการใช้งาน

มิติ	สรุป
Failure Prevented	ป้องกันการใช้ PMBOK แบบ checklist, methodology ตามกระแส และมองข้าม outcome/benefit
What to Monitor	ความครบของมิติการบริหาร (value, stakeholders, risk, quality, delivery) ตลอดโครงการ

14. เชื่อมไปบทถัดไป

Bridge: เมื่อมีกรอบคิดแล้ว เริ่มเดิน workflow — **Ch.1 (Pre-sales)** จะพาไปจาก "ได้โจทย์ลูกค้า" ถึง "Proposal/SOW" ซึ่งเป็นจุดเริ่มของ Value Delivery (บทนี้เป็นบทเดียวที่ไม่มี Workshop/Assessment — ตั้งแต่นี้ทุกบทมีครบ 15 หัวข้อ)

15. Quick Reference Card

```
PMBOK PRIMER (Ch.0)
------------------------------------------------------------
PMBOK = Body of Knowledge (ไม่ใช่ methodology/checklist)
องค์ประกอบ: Principles (6) + Performance Domains (7) + Focus Areas (5)
            + Process Groups (5): Initiating->Planning->Executing->M&C->Closing
เล่มนี้เรียง A→H ตามเวลางานจริง:
  A Pre-sales -> B Initiation -> C Planning -> D Execution
  -> E Monitoring/Change -> F Verification/UAT -> G Go-Live/Hypercare -> H Closure
ใช้ 2 โหมด: Study (เรียงบท) / Field (lookup ผ่าน Appendix F + Quick Reference Card)
Scenario: SHG — 12 โรงแรม, 12M, 35% direct ใน 18 เดือน, launch ก่อน 1 พ.ย.

กฎ 3 ข้อห้าม:
- ห้ามใช้ PMBOK เป็น checklist
- ห้ามเลือก methodology จากกระแส
- ห้ามมองข้าม Outcome/Benefit (Go-live ≠ จบ)
```

Chapter 01 — Pre-sales: จาก Opportunity ถึง Proposal/SOW

1. Opening Scenario Hook

[Teaching Scenario Extension] วันจันทร์เช้า Sales ของบริษัท Booking Tech Solutions (BTS) ซึ่งเป็นบริษัทรับพัฒนาระบบดิจิทัล ส่ง RFP (Request for Proposal) ฉบับหนึ่งมาให้คุณซึ่งเป็น Pre-sales PM:

> Siri Hospitality Group (SHG) เครือโรงแรม 12 แห่ง มีห้องพักรวม ~2,400 ห้อง ขายห้องผ่าน OTA (Agoda, Booking.com) ถึง 70% เสีย Commission 15–25% ไม่มีฐานข้อมูลลูกค้าของตัวเอง ต้องการสร้าง Direct Booking Platform (Website + Mobile App) ให้สัดส่วน Direct Booking เพิ่มจาก 10% เป็น 35% ภายใน 18 เดือนหลัง Launch — งบ Phase 1 อยู่ที่ 12 ล้านบาท ต้องการ Launch ก่อน High Season (1 พฤศจิกายน) และขอใบเสนอราคาภายใน 3 สัปดาห์

ภายในห้องประชุมแรก Sales พูดว่า "โจทย์ชัด 12 ล้านบาท เราน่าจะเสนอ 9 ล้าน รับงานมาเถอะ" ส่วนคุณในฐานะ PM รู้สึกว่ายังมีคำถามอีกมาก: โรงแรม 12 แห่งใช้ PMS ต่างยี่ห้อกันหมดหรือไม่ ข้อมูลห้องว่างซิงก์แบบไหน OTA commission ที่ว่าคือตัวเลขจริงหรือไม่ และทีมเรามี capacity พอส่งมอบภายใน 8 เดือนหรือไม่

คำถามของบทนี้คือ: **"รับงาน" ควรถูกตัดสินใจจากอะไร — ราคาเร็ว ๆ หรือหลักฐานที่พอจะประเมิน risk และ value ได้จริง**

2. Why It Matters

[PMBOK 8] ช่วง Pre-sales ยังไม่ใช่โครงการที่ได้รับ Authorization แต่เป็นจุดที่โครงการถูกกำหนดชะตากรรมไว้ล่วงหน้า ถ้าตรงนี้พลาด โครงการที่ตามมาจะแบกสัญญาที่ผิดพลาดไปทั้งอายุขัย

[Best Practice] ถ้า Proposal ไม่มี Out of Scope, ไม่ระบุ Assumption, ประเมินจากคำบอกเล่าครั้งแรก หรือ Sales รับปาก feature นอกเอกสาร ความขัดแย้งจะย้ายไปเกิดที่กลางโครงการ ตอนที่ทีมส่งมอบแล้ว แต่ลูกค้ายืนยันว่างานไม่ตรงกับที่ "ตกลงกัน" ไว้

[PMBOK 8] แนวคิดที่ควรยึดคือ Value Delivery: ก่อนเขียน Proposal ทีมต้องเข้าใจ Business Need, Desired Outcome และ Success Measures ของลูกค้าให้ได้ก่อน เพราะ Proposal ที่ดีคือเอกสารที่แสดงว่าบริษัทเข้าใจปัญหา ไม่ใช่แค่รายการฟีเจอร์พร้อมราคา

3. Mental Model

จาก Playbook V2 Phase A flow:

```
Opportunity Intake
  -> Initial Discovery (ปัญหาคืออะไร เกิดกับใคร ผลกระทบเท่าไร)
  -> As-Is / Root Cause (ทำไมถึงเป็นแบบนี้)
  -> Desired Outcome + Success Measures (วัดอะไรว่าแก้สำเร็จ)
  -> Solution Shaping (ทางเลือก ไม่ใช่ทางเดียว)
  -> ROM Estimate (ขอบเขต + สมมติฐาน + ช่วงราคา + confidence)
  -> Internal Review (feasibility + margin + risk)
  -> Proposal (มี In/Out of Scope ชัดเจน)
  -> Negotiation (Impact analysis ก่อนยอมเปลี่ยน)
  -> SOW / Contract (ผูกพันตามเอกสารที่ approve แล้ว)
  -> Bid / No-Bid Decision
```

[PMBOK 8] จุดเน้นของบทนี้: อย่าข้าม Discovery แล้วตรงไปเขียน Proposal เพราะ Proposal ที่ไม่มี basis คือการพนันด้วยเงินของบริษัท

4. Main Lesson

4.1 Opportunity Intake — อย่าถามแค่ว่า "งบเท่าไร"

[PMBOK 8] ทุก Opportunity ต้องถูกบันทึกเป็น Opportunity Brief อย่างน้อยต้องตอบ: ลูกค้าคือใคร ใครเป็นผู้ขอ ปัญหาที่เล่ามาเบื้องต้น Deadline และงบประมาณที่เปิดเผย Decision Maker และเหตุผลที่ต้องทำตอนนี้

[Teaching Scenario Extension] ในกรณี SHG: Opportunity Brief ของ BTS ควรบันทึกว่า SHG มี 12 โรงแรม, OTA 70%, commission 15–25%, งบ Phase 1 = 12 ล้านบาท, deadline proposal 3 สัปดาห์, decision maker คือ คุณจิรา (CEO) และ Business Owner คือ คุณภัทร (VP Marketing) — ตัวเลขทั้งหมดตรงกับ Scenario Master

4.2 Initial Discovery — คำถาม 10 ข้อขั้นต่ำ

[PMBOK 8] Discovery คือการเก็บ fact ก่อนออกความเห็น ใช้คำถามขั้นต่ำ 10 ข้อจาก Playbook V2 §A2: ปัญหาปัจจุบันคืออะไร เกิดกับใคร เกิดบ่อยแค่ไหน ผลกระทบคืออะไร ทำไมต้องแก้ตอนนี้ ถ้าไม่ทำจะเกิดอะไร Outcome ที่ต้องการคืออะไร Success วัดอย่างไร มีข้อจำกัดอะไร และใครมีอำนาจตัดสินใจ

[Best Practice] ระหว่าง Discovery ให้จับ As-Is Process (trigger, actor, step, data, decision, system, waiting, error, rework, pain, impact) แล้วทำ Root Cause Analysis แยก symptom / immediate cause / root cause — อย่ารีบแปลง pain point เป็น software feature ทันที

4.3 Desired Outcome และ Success Measures

[PMBOK 8] Outcome ต้องวัดได้มากกว่า "มีระบบจอง" เช่น ลด Double Booking, ลด Average Response Time, เพิ่ม Booking Completion Rate หรือลด Manual Reconciliation

[Teaching Scenario Extension] สำหรับ SHG Outcome ที่ล็อกใน Scenario Master คือ: Direct Booking จาก 10% → 35% ภายใน 18 เดือน, ลด OTA Commission ≥ 3 ล้านบาท/ปี, Customer Database ≥ 10,000 profiles ใน 12 เดือน, Booking NPS ≥ 40 — Proposal ของ BTS ต้องอ้างอิงตัวเลขเหล่านี้ ไม่ใช่ตั้ง KPI ขึ้นใหม่

4.4 Solution Options — อย่าสมมติว่า Solution ต้องเป็น Custom Software

[Best Practice] ทางเลือกอาจเป็น Process Improvement, Training, SaaS, Customize Existing System, Integration, Custom Development, Phased Hybrid หรือ Paid Discovery/POC — แต่ละ Option ต้องระบุ Business Value, Included Capabilities, Exclusions, Assumptions, Dependencies, Risks, Estimated Timeline, Cost Range และ Confidence Level

4.5 ROM Estimate — ประมาณการเพื่อตัดสินใจ ไม่ใช่ Commitment

[Best Practice] ROM (Rough Order of Magnitude) ต้องมี Scope Basis, Assumptions, Team Model, Duration Range, Cost Range, Confidence, Major Unknowns, Contingency Logic และ Expiration/Validity — อย่าปล่อยให้ตัวเลขเดียวถูกส่งให้ลูกค้าโดยไม่มีช่วงและสมมติฐาน

4.6 Internal Review ก่อนส่ง Proposal

[Best Practice] ตรวจ Business fit, Technical feasibility, Delivery feasibility, Operational feasibility, Commercial feasibility, Security/Compliance, Contract risk, Resource availability และ Margin — ถ้าตรวจพบว่า Delivery Team ไม่มี capacity หรือ PMS integration มีความเสี่ยงสูง ต้องกลับไปปรับ Proposal หรือตัดสินใจ No-Bid

4.7 Proposal — 19 องค์ประกอบ

[Best Practice] Proposal ที่ดีควรมี Executive Summary, Business Understanding, Problem and Desired Outcome, Proposed Solution, Scope and Deliverables, In Scope/Out of Scope, Delivery Approach, High-Level Timeline, Team Structure, Pricing, Assumptions, Dependencies, Client Responsibilities, Acceptance Approach, Payment Milestones, Change Approach, Warranty/Support, Proposal Validity และ Next Steps

[PMBOK 6] ที่นี่เองที่ Business Case และ Value ส่วนต้นของ Integration ถูกใช้: Proposal คือสะพานจาก Business Need ไปสู่ Project Charter ใน Ch.2

4.8 Negotiation — ทุกการลดราคา/เร่งเวลา/เพิ่ม scope ต้องมี Impact Analysis

[Best Practice] หลัก Trade-off:

```
เพิ่ม Scope -> ต้องเพิ่มเวลา งบ คน หรือลด Scope อื่น
ลดเวลา   -> ต้องเพิ่มทรัพยากร ลด Scope หรือเพิ่ม Risk
ลดงบ     -> ต้องลด Scope เปลี่ยน Approach หรือยอมรับ Constraint
```

4.9 SOW / Contract — ขอบเขตที่ผูกพัน

[Best Practice] SOW ควรมี Deliverables, Scope, Milestones, Acceptance Criteria/Process, Responsibilities, Dependencies, Payment Terms, Change Control, Warranty, Support, IP, Confidentiality, Liability, Termination, Assumptions และ Exclusions

[PMBOK 8] จำไว้ว่า Proposal กับ SOW ต้องไม่ขัดกัน — ถ้า Proposal บอกว่ามี Out of Scope แต่ SOW กลับรวมงานนั้นไว้ ความขัดแย้งจะกลับมาหา PM ในภายหลัง

4.10 Bid / No-Bid — การตัดสินใจที่คนส่วนใหญ่ข้าม

[Best Practice] ควรประเมิน: ปัญหาของลูกค้ามี Business Value เพียงพอหรือไม่ บริษัทมี Capability/Capacity ส่งมอบหรือไม่ Scope/Timeline/Budget ระดับ Proposal มี Basis พอหรือไม่ มี Commercial/Delivery Risk ใดที่ต้องรู้ก่อนเสนอราคา และควร Bid, No-Bid, Partner หรือเสนอ Discovery Phase แบบเสียเงินก่อน

5. PM Decision Thinking

```
Decision: BTS ควร Bid หรือ No-Bid ใน RFP ของ SHG (และถ้า Bid จะเสนอราคา/ขอบเขตแบบใด)
Owner: Opportunity Owner (CEO/Sales Director) ตัดสินใจสุดท้าย; Pre-sales PM จัดเตรียม evidence
Inputs: Opportunity Brief, Discovery Summary, Solution Options, ROM, capacity check,
margin, risk register เริ่มต้น
Options:
  A: Bid แบบ Fixed Price เต็ม Phase 1 (dev + QA) — margin ขัดแต่รับ delivery risk เอง
  B: Bid แบบ Hybrid (Fixed core + T&M สำหรับ PMS integration ที่ไม่ชัด) — ลด risk ฝั่ง vendor
  C: No-Bid / เสนอ Paid Discovery Phase ก่อน — ปลอดภัยสุด แต่เสี่ยงเสีย opportunity
Trade-offs: margin vs delivery risk, ความเร็วในการตอบ vs ความแม่นยำของ Proposal, โอกาสทางธุรกิจ
vs ชื่อเสียงเมื่อส่งมอบไม่ได้
Risk of wrong decision: รับงานแล้วเจอ PMS integration ต่างยี่ห้อ 12 แบบ -> scope ระเบิด -> โดนหนี
margin
Evidence: bid/no-bid decision record, ROM basis + assumptions, capacity confirmation,
initial risk register
Next Action: ถ้า Bid -> เขียน Proposal ครบ 19 องค์ประกอบ -> Internal review -> ส่งลูกค้า -> เตรียม
negotiation position
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG x BTS)

[Teaching Scenario Extension] สมมติว่าหลัง Discovery ทีม BTS พบ:

- PMS ของ 12 โรงแรมมียี่ห้อต่างกัน 3 ยี่ห้อหลัก มี API ไม่ครบทุกโรงแรม ต้องทำ Adapter แยก (ตรงกับ Risk #1 ใน Scenario Master: "PMS Integration ล่าช้าเพราะ API ไม่พร้อม")
- งบ 12 ล้านบาทของ SHG แบ่งเป็น Dev Team 6.0, Cloud 1.0, Design 0.8, PMS Integration 0.5, Payment Setup 0.2, QA/Security 0.5, Contingency 1.5, Management Reserve 1.5
- BTS จึงเสนอ **Option B (Hybrid)**: Fixed Price 6.5 ล้านบาทสำหรับ Development + QA/Security (ตรงกับสองแถวแรก) และ T&M แบบมี cap 0.5 ล้านบาทสำหรับ PMS Adapter ที่ scope ยังไม่ชัด — โดย Cloud, Design Agency และ Payment ยังคงเป็นค่าใช้จ่ายที่ SHG จัดการเอง เพื่อให้รวมแล้วไม่เกินกรอบ 12 ล้านบาท

[Best Practice] ตัวอย่างนี้แสดงว่า Proposal ที่ดีไม่ใช่แค่เลขเดียว แต่คือการจัดสรรความเสี่ยง (risk allocation) ระหว่าง vendor กับลูกค้าอย่างโปร่งใส — ตรงกับหลัก "เลือก contract type จาก scope maturity และ risk" ที่จะเรียนลึกใน Ch.5

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes (จาก Playbook V2 §A.11):

1. รับปากราคาและเวลาจากคำบอกเล่าครั้งแรก — ผล: ขาด basis ในการ defend เมื่อ scope เปลี่ยน

2. Estimate โดย PM คนเดียว — ผล: ไม่เห็น skill/capacity gap จริงของทีม
3. ใช้ Prototype เป็น Commitment — ผล: ลูกค้าเข้าใจว่า prototype = ฟีเจอร์ที่รับประกัน
4. Proposal ไม่มี Out of Scope — ผล: งานทุกอย่างกลายเป็น "ใน scope" โดยปริยาย
5. Timeline ไม่รวมเวลารอลูกค้า — ผล: แผนสวยแต่ใช้จริงไม่ได้
6. Proposal กับ SOW ขัดกัน — ผล: dispute ตอนส่งมอบและจ่ายเงิน
7. Sales รับปาก Requirement นอกเอกสาร — ผล: PM ต้องแบกสัญญาที่ตัวเองไม่เคยเห็น
8. ไม่ระบุ Confidence ของ Estimate — ผล: ROM ถูกอ่านเป็นราคาผูกพัน
9. ไม่ตรวจ Team Capacity — ผล: รับงานมาแล้วไม่มีคนทำ

Common Misconceptions:

ความเข้าใจผิด	ความจริง
Pre-sales ไม่ใช่งาน PM	PM ที่เข้ามาช่วงนี้ช่วยล็อก scope, risk และ basis ของสัญญา
ROM คือราคาที่ลูกค้าจะจ่าย	ROM คือช่วงประมาณการเพื่อตัดสินใจ ไม่ใช่ commitment
ลูกค้ารู้ว่าตัวเองต้องการอะไร	Discovery ต้องพิสูจน์ — หลายครั้ง pain ≠ solution
Bid ทุกใบเพื่อไม่เสียโอกาส	No-Bid ที่มีเหตุผลช่วยปกป้อง margin และชื่อเสียง
Proposal ยิ่งหน้ายิ่งดี	Proposal ต้องทำให้ decision maker เห็น value + risk ได้ในหน้าครึ่ง

8. Interview Questions

Foundational

- **Q:** อะไรคือความแตกต่างระหว่าง Proposal กับ SOW?
- **Answer Direction:** Proposal คือข้อเสนอ (มี executive summary, scope, timeline, price, assumptions) ส่วน SOW คือขอบเขตงานที่ผูกพันตามสัญญา ต้องไม่ขัดกัน
- **Warning Signs:** ตอบว่าเป็นเอกสารเดียวกัน หรือสลับบทบาทกัน

Scenario

- **Q:** ลูกค้าบอกว่ามีงบ 12 ล้านบาทและอยากได้ใบเสนอราคาภายในสัปดาห์หน้า คุณจะทำอย่างไร?
- **Answer Direction:** ไม่รีบตั้งราคา — เก็บ Opportunity Brief, Discovery ขั้นต่ำ, ตรวจ capacity, หา major unknowns แล้วเสนอ ROM พร้อม confidence และข้อเสนอเรื่อง Paid Discovery ถ้าข้อมูลไม่พอ
- **Warning Signs:** เสนอราคาทันทีโดยไม่มีคำถาม หรือไม่มี In/Out of Scope

Senior PM

- **Q:** Sales รับปาก feature กับลูกค้าแล้วโดยไม่ผ่านคุณ คุณจะจัดการอย่างไร?
- **Answer Direction:** นำ feature เข้า impact analysis อย่างเป็นทางการ แจ้งผลต่อ scope/timeline/price ให้ Sales และลูกค้าเห็น trade-off ก่อนเซ็น — บันทึกทุกอย่างลง proposal/decision record
- **Warning Signs:** เงียบแล้วแบกงานเพิ่ม หรือทะเลาะกับ Sales โดยไม่สร้างทางเลือก

Executive

- **Q:** ทำไมเราถึงควร No-Bid บน RFP ที่ดูมีโอกาสดี?
- **Answer Direction:** อ้าง evidence: delivery risk สูง (เช่น PMS API ไม่พร้อม), margin ไม่พอ, capacity ไม่มี, หรือ compliance risk — และเสนอทางเลือก (partner / paid discovery / รอรอบหน้า)
- **Warning Signs:** ตอบด้วยความรู้สึก "น่าเสียดาย" โดยไม่มีตัวเลข

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Opportunity	โอกาสทางธุรกิจ	lead/RFP/request ที่ยังไม่ใช่โครงการที่ได้รับอนุมัติ	สับสนกับ Project	Bid/No-Bid, Proposal
Bid / No-Bid	การตัดสินใจเสนอราคาหรือไม่	ตัดสินด้วย value, capability, capacity, risk	คิดว่า No-Bid คือการแพ้	Opportunity, Proposal
Discovery	การสำรวจทำความเข้าใจปัญหา	เก็บ fact, pain, root cause, outcome ก่อนออก solution	รีบข้ามไปเป็น feature	Business Case
ROM	Rough Order of Magnitude	ประมาณการระดับสูงเพื่อตัดสินใจ ไม่ใช่ commitment	คิดว่าเป็นราคาผูกพัน	Estimate, Proposal
Proposal	ข้อเสนอโครงการ	executive summary, scope, timeline, price, assumptions	สับสนกับ SOW	SOW, Bid/No-Bid
SOW	Statement of Work	ขอบเขตงานที่ผูกพันตามสัญญา	สับสนกับ Proposal	Contract, Proposal
Assumption	สิ่งที่ถือว่าจริงแต่ยังไม่ได้พิสูจน์	ต้องบันทึกและติดตาม เพราะถ้าผิดกลายเป็น risk	ไม่ถูก track จนเป็น surprise	Risk, Constraint
Desired Outcome	ผลลัพธ์ที่ต้องการ	วัดได้ เช่น 35% direct booking ใน 18 เดือน	สับสนกับ Output	Success Measures, Benefit
Success Measures	ตัวชี้วัดความสำเร็จ	KPI ที่ยืนยันว่า outcome เกิดจริง	ตั้งตัวเลขลอย ๆ	Desired Outcome
Risk Allocation	การจัดสรรความเสี่ยง	ใครรับ risk ไหน ผ่าน contract/approach	คิดว่า vendor ต้องรับทุกอย่าง	Contract, T&M
T&M	Time and Material	จ่ายตามเวลา/วัสดุ เหมาะกับ scope ไม่ชัด	ไม่มี cap/rate/evidence แล้วบานปลาย	Fixed Price
Fixed Price	ราคาคงที่	เหมาะกับ scope ชัด vendor รับ cost risk	ใช้กับ scope ไม่ชัดแล้วเจอ dispute	T&M, Contract

10. Workshop

Scenario

[Teaching Scenario Extension] ทีมคุณ (Pre-sales PM ที่ BTS) ได้รับ RFP จาก SHG เหมือนบทนำ แต่ CEO ขอให้ตอบภายใน 2 สัปดาห์แทน 3 สัปดาห์ และลูกค้าเปิดเผยข้อมูลเพิ่มเติมว่า "PMS บางโรงแรมยังไม่มี API ครบ แต่ฝ่าย IT ของเขาจะช่วย" — ฝ่าย Sales อยากได้ราคา Fixed ตัวเดียวเพราะ "ลูกค้าชอบแบบนั้น"

Your Role

คุณคือ Pre-sales PM ที่ต้องเตรียม Bid/No-Bid recommendation + โครง Proposal ให้ CEO ตัดสินใจ

Available Information

- Scenario Master v1.0 (ตัวเลข SHG ทั้งหมด), Discovery ครบ 10 คำถาม, ROM ของ Functional Leads มีช่วง 6.0–8.5 ล้านบาท, ทีม dev ฟรี capacity เริ่มเดือนถัดไป 80%

Missing Information

- ความชัดเจนของ PMS API ทั้ง 12 โรงแรม (มีข้อมูลแค่ "ไม่ครบ"), เงื่อนไข payment ของ SHG, ระดับการมีส่วนร่วมของฝ่าย IT ลูกค้า, ผลการตรวจ compliance/security ของ platform ที่ต้องทำ

Decision Required

- Bid หรือ No-Bid? ถ้า Bid: Fixed, Hybrid หรือ T&M? ขอบเขต Phase 1 ควรตัดอะไรออกจาก MVP?

Constraints

- งบ Phase 1 ของลูกค้า 12 ล้านบาท (ห้ามเกิน), launch ก่อน High Season, Proposal ต้องมี In/Out of Scope, ต้องไม่รับปากสิ่งที่ไม่มี basis

Expected Output

- Bid/No-Bid decision record หนึ่งหน้า (พร้อม 3 options + trade-off), โครง Proposal 1 หน้า (In/Out of Scope + pricing approach + assumptions), รายการ missing information ที่ต้องขอเพิ่ม

Debrief Questions

1. อะไรคือสัญญาณที่บอกว่าควร No-Bid ในกรณีนี้
2. ทำไม Hybrid pricing ถึงเหมาะกับงานที่มี major unknown อย่าง PMS integration
3. ถ้า CEO บังคับ Fixed Price คุณจะปกป้อง margin ด้วยกลไกอะไรใน SOW

Evaluation Criteria

- มี evidence-based options ไม่ใช่คำตอบเดียว, missing information ถูกระบุเป็นรายการ, pricing ถูกจัดสรรความเสี่ยงอย่างสมเหตุผล, decision record มี owner + next action

11. Checklist ใช้งานจริง (Pre-sales / Phase A)

- ☐ Opportunity Brief บันทึกแล้ว (ลูกค้า, ผู้ขอ, ปัญหา, deadline, งบ, decision maker)
- ☐ Discovery ครบ 10 คำถามขั้นต่ำ และมี As-Is/Root Cause
- ☐ Desired Outcome + Success Measures ตรงกับตัวเลข Scenario Master (35% direct, 18 เดือน, NPS ≥ 40)
- ☐ Solution Options ≥ 2 (ไม่ใช่แค่ custom development)
- ☐ ROM มีช่วง + assumptions + confidence + expiration
- ☐ Internal review ผ่าน (feasibility, margin, capacity, security, contract risk)
- ☐ Proposal มีครบ 19 องค์ประกอบ รวม In/Out of Scope และ Client Responsibilities
- ☐ Negotiation ทุกข้อผ่าน Impact Analysis ก่อนตอบรับ
- ☐ SOW ไม่ขัดกับ Proposal และมี acceptance/payment/change control
- ☐ Bid/No-Bid decision บันทึกเป็นลายลักษณ์อักษร
- ☐ ส่งมอบ Handover Record ไป Ch.2 (Proposal version, SOW version, สิ่งที่ Sales รับปาก, assumptions, risks)

12. Assessment

1. **[Decision Case]** ลูกค้าขอเพิ่ม "Mobile App" เข้า Proposal ทั้งที่ RFP ระบุแค่ Website — งบเท่าเดิม เวลาเท่าเดิม คุณจะทำอะไรก่อนตอบรับ? (ระบุ evidence ที่ต้องขอ และทางเลือกอย่างน้อย 3 ทาง)
2. **[Decision Case]** Sales เสนอราคา 9 ล้านบาทแบบ Fixed โดยอ้างว่าคู่แข่งเสนอ 8.5 ล้าน แต่ ROM ของทีมคือ 9.5–12 ล้าน — คุณจะแนะนำ CEO อย่างไร?
3. **[Trade-off Case]** ระหว่าง Fixed Price กับ T&M สำหรับงาน PMS integration ที่ API ยังไม่ชัด — อธิบาย trade-off ด้าน risk allocation และ margin
4. **[Artifact Review]** ตรวจ Proposal ที่ไม่มีหัวข้อ Out of Scope และ Assumptions — อธิบายว่าเมื่อเซ็นไปแล้วจะเกิดปัญหาอะไรตอนส่งมอบ
5. **[Executive Communication]** เขียนสรุป 1 ย่อหน้าให้ CEO ว่า ทำไมต้องเพิ่ม Paid Discovery Phase ก่อนเซ็นสัญญาเต็ม
6. **[Cross-Knowledge Analysis]** ถ้า SHG ยืนยัน launch ก่อน High Season (1 พ.ย.) แต่ Discovery พบว่า PMS integration น่าจะใช้เวลา 6 เดือน — scope/timeline/price ต้องถูกปรับอย่างไร และใครควรเป็นคนตัดสินใจ
7. **[Recall]** อะไรคือความแตกต่างระหว่าง ROM กับราคาผูกพันใน Proposal

13. Executive Summary

มิติ	สรุป
Why	คุณภาพของ Proposal/SOW กำหนดขอบเขตของความเสี่ยงที่ PM จะต้องแบกไปทั้งโครงการ
Decision Improved	Bid/No-Bid, contract type และ In/Out of Scope ถูกตัดสินด้วย evidence แทนความรู้สึก
Failure Prevented	ป้องกัน scope dispute, margin หาย, และสัญญาที่ส่งมอบไม่ได้
What to Monitor	จำนวน/คุณภาพของ assumption, confidence ของ ROM, และความสอดคล้อง Proposal ↔ SOW

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: เมื่อ SOW เซ็นแล้ว งานยังไม่ใช่ "โครงการ" จนกว่าจะมี Project Charter ที่ Sponsor อนุมัติ — Ch.2 (Initiation) จะเปลี่ยนสัญญาให้กลายเป็นอำนาจการบริหาร: Charter, Stakeholder Register, Governance และ Kickoff

Field	Handoff
Input	Opportunity Brief, Discovery Summary, Solution Options, ROM, Initial Risk Register
Output	Approved Proposal, Signed SOW/Contract, Bid/No-Bid Decision Record
Creator	Sales + Pre-sales PM / Legal / Commercial
Artifact owner	Opportunity Owner -> PM (หลัง authorize)
Reviewer	Legal, Finance, Delivery Owner
Approval authority	CEO / Authorized Signatory
Minimum acceptance	Usable (Proposal กับ SOW ต้องไม่ขัดกัน)
Next chapter use	Ch.2 ใช้ SOW, assumptions และ initial risks มาสร้าง Project Charter + Stakeholder Register

15. Quick Reference Card

```
PRE-SALES (A) — เป้าหมาย: ตัดสินใจ Bid/No-Bid ด้วย evidence
----------------------------------------------------------------
1. Opportunity Brief  -> ใครขอ ทำไมตอนนี้ งบ/เวลา/decision maker
2. Discovery 10 คำถาม -> ปัญหา ใครเจอ ผลกระทบ outcome วัดอะไร
3. As-Is + Root Cause -> อย่ารีบแปลง pain เป็น feature
4. Solution Options   -> ไม่ใช่แค่ custom dev (SaaS/process/partner/POC)
5. ROM                -> ช่วง + assumptions + confidence (ไม่ใช่ราคา)
6. Internal Review    -> feasibility, margin, capacity, security
7. Proposal 19 ข้อ    -> ต้องมี In/Out of Scope + Client Responsibilities
8. Negotiation        -> ทุกยอมลด/เพิ่มต้องมี Impact Analysis
9. SOW                -> ไม่ขัด Proposal + acceptance/payment/change
10. Bid/No-Bid        -> บันทึกเป็น decision record

กฎ 3 ข้อห้าม:
- ห้ามรับปากราคา/เวลาจากคำบอกเล่า
- ห้าม Proposal ไม่มี Out of Scope
- ห้าม Sales รับปาก feature นอกเอกสาร

มือขวาของ PM ในบทนี้: Assumption Log + Initial Risk Register
```

Chapter 02 — Initiation: จากสัญญา สู่ "โครงการที่ได้รับอนุญาต"

1. Opening Scenario Hook

[Teaching Scenario Extension] SOW เซ็นแล้ว SHG อนุมัติงบ Phase 1 จำนวน 12 ล้านบาท และคุณ (PM ของ BTS) ได้รับอีเมลจากคุณจิรา (CEO ของ SHG) ว่า "ฝ่ายเราตั้งคุณสุทธิเป็น PM ประสานงานฝั่งเรา และคุณนภาเป็น Product Owner เริ่มงานได้เลย"

ภายในทีมของคุณ ทุกคนคิดว่างานเริ่มได้ทันทีเพราะ "เซ็นสัญญาแล้ว" แต่คุณรู้ว่ายังขาดอะไรอยู่: ยังไม่มีเอกสารที่ระบุอย่างเป็นทางการว่าใครเป็น Sponsor, PM มีอำนาจแค่ไหน, ใครอนุมัติอะไร, และใครเป็น Stakeholder ที่ต้องดูแล

[PMBOK 8] ช่วง Initiation คือการเปลี่ยน "ข้อตกลงทางการค้า" ให้เป็น "โครงการที่ได้รับอนุญาตและมี governance" — ถ้าข้ามขั้นนี้ งานอาจเริ่มเร็ว แต่ทุกการตัดสินใจใหญ่จะกลายเป็นการทะเลาะว่า "ใครมีสิทธิ์ตัดสินใจ"

2. Why It Matters

[PMBOK 6] Integration Management เริ่มต้นที่นี่ด้วยกระบวนการ **Develop Project Charter** — Charter คือเอกสารที่ Sponsor ออกให้เพื่อให้สิทธิ์ PM อย่างเป็นทางการและกำหนดทิศทางธุรกิจของโครงการ

[Best Practice] โครงการที่ไม่มี Charter มักพบปัญหา: PM ไม่มีอำนาจเรียกทรัพยากร, ฝ่ายต่าง ๆ ไม่รู้ว่าโครงการนี้ทำเพื่ออะไร, และทุก escalation ต้องไล่ถามว่า "ใครเป็นคนตัดสินใจ" — นี่คือต้นทุนที่สูงกว่าเวลาที่เสียไปกับการเขียน Charter หนึ่งหน้า

[PMBOK 6] Stakeholder Management ก็เริ่มที่ Initiating เช่นกัน: ถ้าระบุ Stakeholder ผิดหรือช้า งาน Ch.3 (Requirements) จะเก็บ requirement ไม่ครบ เพราะไม่รู้ว่าใครเป็นเจ้าของความต้องการจริง

3. Mental Model

จาก Playbook V2 Phase B flow:

```
Internal Handover (B1) — Sales -> Delivery: เอาสิ่งที่รับปากไว้ออกมาทั้งหมด
  -> Project Charter (B2) — Sponsor อนุมัติ: ทำไม ใคร งบเท่าไร อำนาจแค่ไหน
  -> Identify Stakeholders (B3) — ใครมีผล/ได้รับผล
  -> Analyze Stakeholders (B4) — power, interest, influence, impact, attitude
  -> Governance + Decision Rights (B5) — ใครตัดสินใจอะไร ตาม threshold ใด
  -> Initial RAID (B6) — Risks, Assumptions, Issues, Dependencies
  -> Kickoff (B7) — ทุกฝ่ายเห็น objective, scope, roles, decision rights พร้อมกัน
  -> ส่งต่อ Planning (Ch.3)
```

[PMBOK 8] หัวใจของบทนี้: Initiation ตอบคำถาม "เรามีสิทธิ์เริ่มและรู้ว่าใครเป็นใครหรือยัง" ก่อนจะลงลึกวางแผน

4. Main Lesson

4.1 Internal Sales-to-Delivery Handover (B1)

[Best Practice] ก่อนเขียน Charter ให้ทำ handover ระหว่าง Sales กับ Delivery: ลูกค้าซื้ออะไร Proposal/SOW เวอร์ชันไหน Sales รับปากอะไรเพิ่ม Estimate ใครทำ Assumptions อะไร Margin เท่าไร Payment milestone ไหน Risks อะไร — ผลลัพธ์คือ Handover Record ที่ทำให้ PM ไม่ต้องไล่ตามความจำของ Sales

4.2 Develop Project Charter (B2) — หัวใจของ Integration

[PMBOK 6] Charter ควรมี: Project Purpose, Business Need, Measurable Objectives, Success Criteria, High-Level Scope, High-Level Requirements, Major Deliverables, Summary Milestones, Initial Budget, High-Level Risks, Assumptions, Constraints, Sponsor, PM, PM Authority, Governance, Approval Requirements, Exit Criteria และ Charter Approval

[PMBOK 8] จำไว้ว่า **Charter ไม่ใช่ Detailed Plan** — เป็นเอกสารระดับสูงที่ให้อำนาจและทิศทาง ไม่ใช่ที่เก็บ schedule รายละเอียด (งานนั้นคือ Ch.4)

[Teaching Scenario] Charter ของ SHG ควรระบุ: Purpose = เพิ่ม Direct Booking จาก 10% เป็น 35% ใน 18 เดือน, Budget = 12 ล้านบาท (Phase 1), Milestone = Launch ก่อน 1 พฤศจิกายน, Sponsor = คุณจิรา (CEO), PM = คุณสุทธิ, PO = คุณนภา, High-Level Risks = PMS API ไม่พร้อม, Payment Security, Conversion ต่ำ

4.3 Identify + Analyze Stakeholders (B3–B4) — จาก lesson-06

[PMBOK 6] Stakeholder ไม่ได้มีเฉพาะคนที่เข้าประชุม ต้องพิจารณา Sponsor, Customer PO, Decision Maker, End User, Operations, IT, Security, Legal, Finance, Vendor, Regulator, Support, Management และผู้ได้รับผลกระทบทางลบ

Stakeholder Register ที่ดีไม่ใช่ contact list — ต้องมี Role, Interest, Influence, Impact, Attitude, Information Need, Engagement Need, Decision Right, Acceptance Role และ Risk of Resistance

Power–Interest Grid:

Power	Interest	Approach
High	High	Manage closely
High	Low	Keep satisfied
Low	High	Keep informed
Low	Low	Monitor

[Teaching Scenario] สำหรับ SHG: คุณจิรา (CEO) = High/High → Manage closely, คุณสมศรี (VP Ops) = High/Medium → Keep satisfied, Front Desk Staff = Low/High → Keep informed, ลูกค้า (Guests) = Low/High → Monitor + UX research

4.4 Engagement Strategy — current vs desired

[PMBOK 6] ระดับ engagement: unaware, resistant, neutral, supportive, leading — ใช้ Engagement Assessment Matrix เทียบ current กับ desired แล้วออกแบบ action ที่มี owner, channel, trigger และ success signal

[PMBOK 8] Communication คือการส่งข้อมูลให้เข้าใจตรงกัน ส่วน Engagement คือการออกแบบปฏิสัมพันธ์ให้ decision, adoption และ benefit เกิดขึ้นจริง — สองเรื่องนี้เชื่อมกันแต่ไม่เหมือนกัน (รายละเอียดการสื่อสารอยู่ Ch.5)

4.5 Governance และ Decision Rights (B5)

[Best Practice] กำหนด Sponsor Decision, Steering Committee, Project Board, PM Authority, PO Authority, Technical Authority, Change Authority, Acceptance Authority, Escalation Path, Decision SLA และ Meeting Cadence — เอกสารนี้คือ "รัฐธรรมนูญ" ของโครงการ ตอนมี conflict จะได้เปิดดูว่าใครตัดสินใจ ไม่ใช่ใครเสียงดังกว่า

4.6 Initial RAID (B6)

[Best Practice] เปิด RAID Log ตั้งแต่ตอนนี้: Risks, Assumptions, Issues, Dependencies — ทุก record ต้องมี ID, Description, Owner, Date, Impact, Response, Due Date, Status, Escalation

4.7 Kickoff (B7)

[Best Practice] Kickoff ต้องยืนยัน: Business Objective, Scope/Deliverables, Out of Scope, Timeline/Milestones, Roles, Decision Rights, Communication, Working Approach, Dependencies, Risks, Change Process และ Immediate Actions — จบด้วย MoM + Action Register ที่มี owner และ due date

4.8 หมายเหตุ Integration — ถูกผ่าเป็น 3 ท่อน (ไม่ใช่ความผิดพลาด)

[PMBOK 6] Integration Management ทำงานตลอดทั้งโครงการ: **Charter → บทนี้ (Ch.2)**, **Perform Integrated Change Control → Ch.7**, **Close Project or Phase → Ch.10** — นี่คือ pattern ปกติของ KA ที่ครอบคลุมทั้ง lifecycle ไม่ใช่การตัดเนื้อหา

5. PM Decision Thinking

```
Decision: ขอบเขตอำนาจของ PM ใน Charter ควรอยู่ระดับใด (และ Stakeholder ใครต้องถูก engage ก่อน
Kickoff)
Owner: Sponsor ออก Charter; PM เสนอ draft; Steering Committee เห็นชอบ governance
Inputs: SOW, Handover Record, business need, success criteria, stakeholder signals, org
context
Options:
  A: PM Authority กว้าง (ตัดสินใจได้ถึง threshold 1M + เปลี่ยน scope minor ได้) — เร็ว แต่เสี่ยง
governance หลุด
  B: PM Authority ปานกลาง (threshold 500K, scope change ต้อง CCB) — สมดุล (default ที่แนะนำ)
  C: PM Authority แคบ (ทุกอย่างต้อง Steering) — ปลอดภัย แต่ช้าและ PM กลายเป็นคนเดินเอกสาร
Trade-offs: decision speed vs control, PM empowerment vs sponsor oversight
Risk of wrong decision: PM ไม่มีอำนาจ -> escalation ท่วม; PM มีอำนาจเกิน -> ตัดสินใจผิดโดยไม่มี check
Evidence: approved charter, decision rights matrix, escalation log, charter approval record
Next Action: เขียน Stakeholder Register + Governance Map -> เรียก Kickoff -> ส่งต่อ Ch.3
(Requirements/Scope)
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario] หลัง Kickoff ระหว่าง BTS กับ SHG ทีมตกลงกันดังนี้:

- **Charter (SHG ฝั่ง):** Sponsor = คุณจิรา, PM = คุณสุทธิ, PO = คุณนภา, งบ 12 ล้านบาท, Launch ก่อน 1 พ.ย., Objective = 35% direct booking ใน 18 เดือน
- **Stakeholder Register:** คุณภัทร (VP Marketing, Business Owner) = High/High, คุณสมศรี (VP Ops) = High/Medium, คุณกาญจนา (Revenue Mgr) = Medium/High, คุณวีระ (CTO) = High/High, 2C2P + PMS Vendors = Medium/Medium, Front Desk = Low/High
- **Governance:** Steering Committee = คุณจิรา + คุณภัทร + คุณวีระ + PM; Change Authority = CCB (คุณจิรา อนุมัติ baseline change, PM อนุมัติ minor ≤ 500K); Acceptance Authority = คุณนภา (PO) + คุณภัทร สำหรับ UAT sign-off
- **Initial RAID:** Risk #1 = PMS API ไม่พร้อม (High/High), Risk = Payment Security (Low/Critical), Assumption = ทีม recruit ครบใน 2 สัปดาห์, Dependency = 2C2P contract, PMS vendor API docs

[PMBOK 8] สังเกตว่า Ch.2 จบที่ทุกฝ่าย "เห็นภาพเดียวกัน" — ยังไม่มีการวางแผนรายละเอียด นั่นคืองาน Ch.3–5

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. เริ่มงานทันทีที่เซ็นสัญญา โดยไม่มี Charter — ผล: PM ไม่มีอำนาจเรียกทรัพยากร
2. ให้ PM เขียน Charter เองแล้ว Sponsor แค่เซ็น — ผล: Sponsor ไม่ได้ ownership ต่อทิศทางโครงการ
3. Stakeholder Register = รายชื่อคนในอีเมล — ผล: พลาดคนที่มี influence จริง
4. ใส่ Accountable หลายคนใน decision — ผล: ไม่มีใครรับผิดชอบสุดท้าย
5. Governance ตั้งแล้วไม่เปิดใช้ — ผล: conflict กลับไปตัดสินด้วยเสียงดัง

6. Kickoff เป็นแค่ presentation ที่ไม่มี action register — ผล: ทุกคนเห็นแต่ไม่มีใครรับผิดชอบ

Common Misconceptions:

ความเข้าใจผิด	ความจริง
Charter กับ Project Plan คือสิ่งเดียวกัน	Charter ให้อำนาจและทิศทาง; Plan มีรายละเอียดการบริหาร
Stakeholder = คนที่เกี่ยวข้องกับการประชุม	คือใครก็ตามที่มีผลหรือได้รับผลจากโครงการ
PM อนุมัติทุกอย่างได้	Authority ต้องเป็นไปตาม Governance/Decision Rights
Sponsor แค่เซ็นเอกสาร	Sponsor ต้องให้ direction, escalation และ visible support
Kickoff เสร็จ = Initiation เสร็จ	Initiation จบเมื่อ governance + stakeholder strategy พร้อม

8. Interview Questions

Foundational

- **Q:** Project Charter ต่างจาก SOW อย่างไร?
- **Answer Direction:** SOW คือขอบเขตงานตามสัญญาจากมุม commercial; Charter คือ authorization ภายในที่ Sponsor ออกให้ PM — ทั้งคู่เป็น input ของกันและกัน
- **Warning Signs:** ตอบว่าเหมือนกัน หรือสลับบทบาท

Scenario

- **Q:** CFO ของลูกค้าไม่พอใจที่ไม่ได้อยู่ใน Steering Committee และขู่จะชะลอการอนุมัติ คุณจะทำอย่างไร?
- **Answer Direction:** กลับไปที่ Stakeholder Register — ประเมิน power/interest ของ CFO (น่าจะ High/High ด้าน finance) เสนอ Sponsor เพิ่ม CFO ใน governance หรือให้บทบาท Finance representative + decision right ด้านงบ — ไม่ทะเลาะ แต่ปรับโครงสร้าง
- **Warning Signs:** แก้ด้วยการ "เชิญมาประชุมเฉย ๆ" หรือไม่ทำอะไรแล้วรอให้เดือดร้อน

Senior PM

- **Q:** Sponsor ไม่อยากออก Charter เพราะ "เสียเวลา" คุณจะจัดการอย่างไร?
- **Answer Direction:** อธิบายผลของ decision latency ที่จะเกิด (ทุก escalation ไม่มีเจ้าภาพ) เสนอ Charter ฉบับสั้น 1–2 หน้า และขอ approval ผ่าน session สั้น ๆ — แสดงว่า Charter คือเครื่องมือลดความเสี่ยง ไม่ใช่เอกสารเพิ่มภาระ
- **Warning Signs:** ยอมทำงานโดยไม่มี Charter แล้วมารับความเสี่ยงเองคนเดียว

Executive

- **Q:** ทำไม PM ต้องมี "อำนาจ" ที่เขียนไว้ใน Charter ไม่ใช่แค่ทำงานตามที่ได้รับมอบหมาย?
- **Answer Direction:** อำนาจที่ชัดเจน = decision speed + accountability; ไม่มีอำนาจ = escalation ทุกรายละเอียด, PM กลายเป็นคนเดินเอกสาร และ decision ถูกเลื่อนจนค่าเสียหายสูง
- **Warning Signs:** ตอบเรื่อง title/ตำแหน่ง แทน decision authority

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Project Charter	เอกสารอนุมัติโครงการ	Sponsor ให้สิทธิ์ PM + กำหนดทิศทาง	สับสนกับ Project Plan	Sponsor, PM Authority
Sponsor	ผู้สนับสนุนโครงการ	ให้ทรัพยากร อนุมัติ ตัดสินใจระดับสูง	สับสนกับ Product Owner	Charter, Governance
PM Authority	อำนาจของ PM	ขอบเขตตัดสินใจตาม threshold	คิดว่า PM อนุมัติทุกอย่าง	Governance, Decision Rights
Governance	กลไกตัดสินใจ/อนุมัติ	ใครตัดสินใจอะไร ตาม threshold ใด	สับสนกับ Meeting	CCB, RACI
Decision Rights	สิทธิ์ตัดสินใจ	อำนาจ approve/reject/defer	สับสนกับ Responsibility	Governance
Stakeholder Register	ทะเบียน stakeholder	role, interest, influence, needs, strategy	สับสนกับ Contact list	Power-Interest Grid
Power-Interest Grid	ตารางอำนาจ/ความสนใจ	เลือกวิธี engage แต่ละ stakeholder	สับสนกับ Org chart	Stakeholder Register
Engagement	การมีส่วนร่วม	ระดับการรับรู้/สนับสนุน/มีบทบาท	สับสนกับ Communication	Current vs Desired
RAID	Risks, Assumptions, Issues, Dependencies	log กลางที่ต้องมี owner + review	ทำครั้งเดียวแล้วเก็บ	Risk Register, Issue
Kickoff	ประชุมเริ่มโครงการ	ทุกฝ่ายเห็น objective/roles/decision rights	เป็นแค่ presentation	Charter, Governance
CCB	Change Control Board	อนุมัติ change ที่กระทบ baseline	คิดว่า PM อนุมัติเองได้ทุกเรื่อง	Change Request, Governance
Stakeholder Engagement Assessment	การประเมินการมีส่วนร่วม	เทียบ current vs desired engagement	ไม่ทำ และเดาเอา	Engagement, Register

10. Workshop

Scenario

[Teaching Scenario] คุณเพิ่งได้รับแต่งตั้งเป็น PM ฝั่ง SHG (คุณสุทธิ) หลัง Kickoff กับ BTS — Sponsor (คุณจิรา) ต้องการให้เริ่ม Sprint 0 ภายในสัปดาห์หน้า แต่คุณพบว่า Stakeholder Register ยังมีแค่ชื่อคน 4 คน และไม่มี Governance Map

Your Role

PM ฝั่งลูกค้าที่ต้องทำให้ Initiation สมบูรณ์ก่อน Sprint 0

Available Information

- Charter ที่ Sponsor เซ็นแล้ว (งบ 12M, launch ก่อน 1 พ.ย.), SOW จาก BTS, รายชื่อทีมใน Scenario Master (PO, dev 12 คน, QA 2, DevOps 1, Hotel Coordinator 1)

Missing Information

- ความเห็น/attitude ของคุณสมศรี (VP Ops) และ GM 12 โรงแรม, ความต้องการข้อมูลของ 2C2P/PMS vendors, decision threshold ที่ Sponsor อนุมัติ, escalation path สำหรับกรณี PMS API ไม่พร้อม

Decision Required

- ใครอยู่ใน Steering Committee? threshold ของ PM authority เท่าไร? Stakeholder กลุ่มใดต้องถูก engage ก่อน Sprint 0 เริ่ม?

Constraints

- ต้องไม่ขัดกับ roles ที่ล็อกใน Scenario Master, ต้องไม่เริ่ม Sprint 0 โดยไม่มี governance, ห้ามแก้ตัวเลขงบ/เวลาที่ล็อกไว้

Expected Output

- Stakeholder Register + Engagement Strategy (อย่างน้อย 8 stakeholder), Governance Map หนึ่งหน้า (decision rights + escalation), RAID เริ่มต้น 5 รายการ, Kickoff agenda ที่เหลือ

Debrief Questions

1. Stakeholder กลุ่มไหนที่ถูกมองข้ามง่ายที่สุดในโครงการนี้ และผลคืออะไร
2. ถ้า PM ไม่มี threshold ชัดเจน จะเกิดอะไรตอน Ch.7 (Change Control)
3. Governance ที่ "ตั้งแล้วไม่ใช้" ต่างจากที่ "ใช้จริง" อย่างไร

Evaluation Criteria

- Register มี influence/engagement/owner ไม่ใช่แค่ชื่อ, Governance มี decision rights + escalation, RAID มี owner ทุกรายการ, การตัดสินใจไม่ขัดกับ Scenario Master

11. Checklist ใช้งานจริง (Initiation / Phase B)

- ☐ Internal Handover Record ทำแล้ว (Proposal/SOW version, สิ่งที่ Sales รับปาก, assumptions)
- ☐ Charter ผ่าน Sponsor อนุมัติแล้ว (purpose, objectives, budget, milestones, PM authority, exit criteria)
- ☐ Stakeholder Register มี ≥ 8 รายการ พร้อม power/interest/current/desired/owner

- ☐ Governance Map อนุมัติแล้ว (Steering, CCB, PM/PO/Technical/Acceptance authority, escalation path)
- ☐ Initial RAID เปิดแล้ว ทุก record มี owner + due date
- ☐ Kickoff จบด้วย MoM + Action Register (owner + due date ทุก action)
- ☐ ส่งต่อ Ch.3: Stakeholder needs, decision rights, acceptance roles พร้อมใช้เก็บ requirements

12. Assessment

1. **[Decision Case]** Sponsor ขอให้ PM "จัดการทุกอย่างเอง" โดยไม่กำหนด threshold ใน Charter — คุณจะแนะนำอย่างไร และจะเกิดความเสี่ยงอะไรถ้า PM มีอำนาจไม่จำกัด?
2. **[Decision Case]** Stakeholder Register มีแค่ 4 คน (CEO, CTO, PO, PM) ก่อน Kickoff — ระบุกลุ่มที่ขาดและเหตุผลที่ต้อง engage ก่อน Sprint 0
3. **[Trade-off Case]** ระหว่าง PM authority กว้างกับแคบ — อธิบาย trade-off ด้าน decision speed vs control พร้อมข้อเสนอ threshold สำหรับโครงการ 12 ล้านบาทนี้
4. **[Artifact Review]** ตรวจ Charter ที่มี "PM เขียนเอง Sponsor แค่เซ็น" และไม่มี exit criteria — อธิบายว่า governance จะพังตรงไหนเมื่อเกิด conflict
5. **[Executive Communication]** เขียน 1 ย่อหน้าให้คุณจิรา (CEO) ว่า ทำไมต้องมี Steering Committee และ PM threshold ที่ชัดเจน ก่อน Sprint 0
6. **[Cross-Knowledge Analysis]** ถ้า CFO ของ SHG ต้องการเป็นผู้ตัดสินใจเรื่องงบทุกครั้ง (แม้ minor) — สิ่งนี้จะกระทบ Ch.4 (Cost) และ Ch.7 (Change) อย่างไร และคุณจะปรับ governance อย่างไร
7. **[Recall]** ความแตกต่างระหว่าง Stakeholder Register กับ Power-Interest Grid คืออะไร

13. Executive Summary

มิติ	สรุป
Why	Initiation ล็อก "ใครมีอำนาจทำอะไร" ก่อนงานเริ่ม — ความชัดเจนตรงนี้ลดความขัดแย้งตลอดทั้งโครงการ
Decision Improved	การอนุมัติ, escalation และ change decision มีเจ้าภาพชัดเจนตาม Governance
Failure Prevented	ป้องกัน PM ไร้อำนาจ, Stakeholder พลาด, และ governance ที่ไม่มีใครใช้
What to Monitor	Stakeholder engagement shift, RAID ที่มี owner, และการใช้ decision rights จริง

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: Initiation จบเมื่อมี authorization + governance + stakeholder strategy — Ch.3 จะใช้ Stakeholder needs และ decision rights เหล่านี้ไปเก็บ Requirements, กำหนด Scope และสร้าง WBS

Field	Handoff
Input	SOW, Handover Record, Business Need (จาก Ch.1)
Output	Approved Charter, Stakeholder Register + Engagement Strategy, Governance Map, Initial RAID

Field	Handoff
Creator	PM + BA (Stakeholder), Sponsor (Charter/Governance)
Artifact owner	Sponsor (Charter/Governance), PM (Register/RAID)
Reviewer	PMO, Business Owner, Change Lead, Functional Leads
Approval authority	Sponsor / Steering Committee
Minimum acceptance	Usable (ต้องมี decision rights + engagement owner)
Next chapter use	Ch.3 ใช้ stakeholder needs, decision rights และ acceptance roles เพื่อเก็บ requirements และ approve scope baseline

15. Quick Reference Card

```
INITIATION (B) — เป้าหมาย: Authorization + Governance + Stakeholder ชัด
---------------------------------------------------------------
1. Internal Handover  -> สิ่งที่ Sales รับปากทั้งหมด ออกมาจากความจำ
2. Charter            -> Sponsor อนุมัติ: ทำไม ใคร งบ เป้า อำนาจ PM
3. Stakeholder Reg.   -> role, interest, influence, impact, attitude
4. Power-Interest     -> Manage closely / Keep satisfied / Keep informed / Monitor
5. Engagement         -> current vs desired + owner + trigger + success signal
6. Governance         -> Steering, CCB, PM/PO/Tech/Acceptance authority, escalation
7. Initial RAID       -> ทุก record มี ID, Owner, Due date, Escalation
8. Kickoff            -> objective, scope, roles, decision rights + MoM/Actions

กฎ 3 ข้อห้าม:
- ห้ามเริ่มงานโดยไม่มี Charter (PM ไร้อำนาจ)
- ห้าม Stakeholder Register = contact list
- ห้าม Governance ตั้งแล้วไม่ใช้

Integration note: Charter = บทนี้, Change Control = Ch.7, Closure = Ch.10
```

Chapter 03 — Requirements, Scope และ WBS: ทำให้ "โจทย์" กลายเป็น "ขอบเขตที่ควบคุมได้"

1. Opening Scenario Hook

[Teaching Scenario] คุณคือคุณสุทธิ (PM ฝั่ง SHG) ระหว่าง workshop requirements คุณภัทร (VP Marketing) บอกว่า "ผมต้องการ dashboard ที่ดู conversion ทุกอย่าง และต้องมี loyalty program ด้วยนะ เดี๋ยวเราค่อยขายเพิ่มทีหลัง" ขณะที่คุณวีระ (CTO) ตอบว่า "ทีมเราทำได้ แต่ต้องรู้ขอบเขตก่อน" และคุณกาญจนา (Revenue Manager) ถามว่า "แล้วข้อมูลราคา/ห้องว่างของ 12 โรงแรมจะมาครบเมื่อไร"

ทุกคนพูดเป็นภาษา "ความต้องการ" แต่ยังไม่มีใครพูดเป็นภาษา "requirement ที่ตรวจรับได้" — และถ้าเก็บคำว่า "ทุกอย่าง" เข้าไปใน scope วันนี้ พรุ่งนี้ทั้งทีมจะต้องเดาว่าจริง ๆ แล้วต้องสร้างอะไร

[PMBOK 8] บทนี้คือจุดที่ Stakeholder needs จาก Ch.2 ถูกแปลงเป็น requirement, scope boundary และ WBS ที่ Ch.4 (Schedule/Cost/Resource) จะเอาไปวางแผนต่อได้จริง

2. Why It Matters

[PMBOK 6] Scope Management ตอบคำถามสามข้อ: จะส่งมอบอะไร จะไม่ส่งมอบอะไร และจะรู้ได้อย่างไรว่าส่งมอบแล้วผ่าน — ถ้าขอบเขตไม่ชัด Schedule จะเป็นการเดาวัน, Cost เป็นตัวเลขไม่มีฐาน, Quality ไม่มีเกณฑ์ตรวจรับ และ Change Control กลายเป็นการเถียงว่า "เรื่องนี้เคยอยู่ใน scope ไหม"

[Best Practice] Scope ที่ดีไม่ใช่เอกสารยาว แต่เป็นเส้นแบ่งที่ทีม ผู้บริหาร และผู้ใช้เข้าใจตรงกันพอที่จะตัดสินใจได้

3. Mental Model

จาก Playbook V2:

```
Stakeholder Need (Ch.2)
  -> Tailor Delivery Approach (C1) — Predictive/Agile/Hybrid ต่อ workstream
  -> Plan Requirements Work (C2) — จะเก็บ/วิเคราะห์/อนุมัติ/trace อย่างไร
  -> Collect & Analyze Requirements (C3) — types + techniques + quality
  -> Document Set Decision (C4) — SRS/FSD หรือสิ่งทดแทน ตาม complexity
  -> Define Scope (C5) — Product vs Project, In/Out, acceptance criteria
  -> WBS + Dictionary (C6) — แตกตาม deliverable จนถึง work package
  -> Scope Baseline (approved) -> Ch.4 (Schedule/Cost/Resource)
```

> **Cross-reference (Quality):** Acceptance Criteria ถูกกำหนดในบทนี้ต่อ requirement; **Test Strategy / Quality Plan ฉบับเต็มอยู่ Ch.5** (วางแผนคุณภาพ) และ **ผลการทดสอบ/UAT Evidence อยู่ Ch.8** — อย่ามองข้ามเส้นนี้

4. Main Lesson

4.1 Tailor Delivery Approach (C1) — เลือกวิธี ไม่ใช่เลือกตามใจ

[PMBOK 8] เลือก Predictive/Agile/Hybrid จาก: Requirement Stability, Technical Uncertainty, Contract Type, Regulatory, Release Frequency, Customer Availability, Team Maturity, Dependency, Architecture, Risk และ Need for Formal Approval

[Teaching Scenario] SHG ใช้ Hybrid: การวางแผน scope/schedule/cost ทำแบบ predictive (ต้องควบคุม 12M และ launch 1 พ.ย.) แต่การพัฒนาเป็น sprint-based (Scrum) ตาม Scenario Master — รายละเอียด Tailoring อยู่ Ch.4/Ch.6 (กล่อง "ถ้าโครงการเป็น Agile")

4.2 Plan Requirements Work (C2)

[Best Practice] กำหนด: แหล่ง requirement, stakeholder ที่ต้องร่วม, เทคนิค elicitation, ประเภท requirement, format, review/approval, traceability (RTM), change control และ versioning — ผลลัพธ์คือ Requirements Management Plan

4.3 Collect and Analyze Requirements (C3)

[PMBOK 6] ประเภท requirement: Business, Stakeholder, Solution (Functional/Non-functional), Transition, Data, Integration, Reporting, Security/Compliance, Operational และ Acceptance

[Best Practice] เทคนิค: Interview, Workshop, Observation, Questionnaire, Document Analysis, Interface Analysis, Process Modeling, Prototyping, Benchmarking, Focus Group, Story Mapping, Use Case, User Story, Data Modeling

Requirement ต้องมีคุณภาพ 9 ข้อ: Clear, Complete, Consistent, Feasible, Testable, Traceable, Prioritized, Unambiguous, Necessary และ Owned

4.4 SRS/FSD — เอกสารหรือสิ่งทดแทน (C4)

[PMBOK 8] PMBOK ไม่บังคับชื่อเอกสาร — บังคับว่า Requirement ถูก Elicit, Analyze, Document, Prioritize, Approve, Trace, Validate และ Control

[Best Practice] SRS (Software Requirements Specification) ควรมีเมื่อ contract formal/ระบบซับซ้อน/หลายทีม/ต้อง sign-off/compliance; FSD (Functional Specification) ครอบคลุม behavior, business rules, screen behavior, validation, error handling — **ห้ามตัด FSD โดยไม่มีสิ่งทดแทน** เมื่อ logic ซับซ้อน/หลายระบบ/offshore/audit/ผูก payment (ดู Document Substitution Matrix ใน Playbook V2 §C4.4)

4.5 Define Scope (C5)

[PMBOK 6] แยก **Product Scope** (คุณสมบัติของสิ่งที่ส่งมอบ) ออกจาก **Project Scope** (งานที่ต้องทำเพื่อให้ได้สิ่งนั้น) — Scope Statement ควรมี Product Scope Description, Project Deliverables, Acceptance Criteria, Exclusions, Constraints และ Assumptions

[Teaching Scenario] SHG Phase 1 (MVP) In Scope: Customer Web App, Mobile App (iOS+Android), Landing Page, Back Office Web App, Booking Engine, PMS Sync 12 โรงแรม, Payment (2C2P), Notification — Out of Scope: Loyalty Program, Digital Key, Chatbot, Multi-language (นอกจาก TH/EN), Restaurant/Spa Booking, Channel Manager (ทั้งหมดตรง Scenario Master §4)

4.6 WBS และ WBS Dictionary (C6)

[PMBOK 6] WBS แตก Total Project Scope เป็น Deliverable และ Work Package ที่ estimate/assign/schedule/cost/control ได้ — ใช้ Decomposition + 100% Rule (ผลรวมของ child = parent = approved scope เท่านั้น ไม่รวมทุกคำขอ)

ขั้นตอน: ระบุ final product → ระบุ major deliverables → เลือก decomposition logic → แตก deliverables → ตรวจ 100% Rule → หยุดที่ work package → ระบุ owner → สร้าง WBS Dictionary → ตรวจ scope coverage → approve scope baseline

[Best Practice] WBS Dictionary ต้องมี: WBS ID, Work Package, Description, Deliverable, Included/Excluded Work, Owner, Requirements, Acceptance Criteria, Quality Criteria, Assumptions, Constraints, Dependencies, Milestone, Resource, Estimate, Cost Account, Risk, Approval

[PMBOK 6] อย่าแตก WBS ตาม department — แตกตาม deliverable (ไม่ใช่ "งานของแผนก IT" แต่เป็น "PMS Sync Adapter" หรือ "Payment Integration")

4.7 Scope Baseline และ Agile alternative

[PMBOK 6] Scope Baseline = Scope Statement + WBS + WBS Dictionary

[PMBOK 8] ถ้า Agile: ใช้ Product Goal, Product Backlog, Epic/Feature, User Story + Acceptance Criteria, Release Boundary, Definition of Done — แต่ยังต้องมี Scope/Value Boundary และ Traceability

5. PM Decision Thinking

```
Decision: คำขอ "loyalty program" ของคุณภัทร อยู่ใน scope baseline หรือเป็น change (และควร
include/defer/replace)
Owner: PM วิเคราะห์ impact; PO คุณภา จัด priority; Sponsor/CCB อนุมัติเมื่อกระทบ baseline
Inputs: requirement, scope statement, WBS, WBS dictionary, acceptance criteria, impact
analysis, value
Options:
  A: Include ใน Phase 1 — value สูงแต่เสี่ยง payment/launch readiness
  B: Defer ไป Phase 2 (ตาม Scenario Master: Loyalty อยู่ Out of Scope) — รักษา MVP + high
season launch
  C: Replace ฟีเจอร์ value ต่ำกว่าใน MVP — ต้องมี decision evidence ว่า replace อะไร
Trade-offs: value เพิ่ม vs schedule/cost/quality risk, scope completeness vs launch readiness
Risk: รับ "ทุกอย่าง" เข้า MVP -> scope ระเบิด -> 12M/8 เดือนไม่พอ
Evidence: updated scope baseline, decision record, stakeholder communication
Next Action: แปลงคำขอเป็น requirement ที่ตรวจรับได้ -> ถ้าเกิน MVP ให้เข้า change process (Ch.7)
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario] ทีม SHG เก็บ requirements สำหรับ Direct Booking Platform:

- **Business Requirement:** เพิ่ม Direct Booking จาก 10% เป็น 35% ใน 18 เดือน, ลด OTA commission ≥ 3 ล้านบาท/ปี
- **Functional (ตัวอย่าง):** ผู้ใช้ค้นหาห้องพักตามโรงแรม/วันที่/จำนวนผู้เข้าพัก, เปรียบเทียบราคา, จองและชำระเงินผ่าน 2C2P, รับ confirmation email/SMS
- **Non-functional:** Page load ≤ 3 วินาที, Booking confirmation ≤ 5 วินาที, Uptime ≥ 99.5%, PCI-DSS + PDPA compliance
- **WBS ตัวอย่าง (Payment Integration work package):** Payment Gateway Contract, API Design, Payment Flow Implementation, Callback Handling, Security Review, Payment Testing, Reconciliation Setup — แต่ละ item มี owner + acceptance criteria ใน WBS Dictionary

[PMBOK 8] WBS fragment นี้จะถูกใช้ต่อใน Ch.4 เพื่อนิยาม activities, dependencies และ duration

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. รับคำว่า "ทั้งหมด/ทุกอย่าง" เป็น requirement — ผล: ขอบเขตไร้ขีดจำกัด
2. เก็บ requirement เป็น "ฟีเจอร์" ทันทีโดยไม่ถาม decision need — ผล: สร้างของที่ไม่มีใครใช้
3. WBS แตกตามแผนก/activity — ผล: ไม่เห็น deliverable และ 100% Rule พัง
4. ไม่มี acceptance criteria ต่อ requirement — ผล: UAT (Ch.8) เถียงกันว่า "แบบไหนถึงจะผ่าน"
5. ลืม Out of Scope — ผล: ทุกอย่างกลายเป็น "ใน scope"
6. ไม่มี RTM — ผล: ตรวจไม่ได้ว่า requirement ไหนถูก build/test/accept

Common Misconceptions:

ความเข้าใจผิด	ความจริง
WBS คือ task list	WBS คือ deliverable/work-package structure
Requirement ละเอียด = ดี	ต้อง Testable + Traceable + Prioritized
100% rule = รวมทุกคำขอ	รวมเฉพาะ approved scope
SRS/FSD เป็นชื่อเอกสารที่บังคับ	บังคับที่ coverage ไม่ใช่ชื่อไฟล์
Scope ชัด = เขียนยาว	Scope ชัด = มี boundary + acceptance

8. Interview Questions

Foundational

- **Q:** Product Scope ต่างจาก Project Scope อย่างไร?

- **Answer Direction:** Product Scope = คุณสมบัติของสิ่งที่ส่งมอบ (features); Project Scope = งานทั้งหมดที่ต้องทำเพื่อส่งมอบ (blueprint, migration, test, training)
- **Warning Signs:** ตอบว่าเหมือนกัน หรือสลับกัน

Scenario

- **Q:** ลูกค้าขอ "รายงานทุกอย่าง" ระหว่างเก็บ requirement คุณจะจัดการอย่างไร?
- **Answer Direction:** ถามกลับ: รายงานใช้ตัดสินใจอะไร ใครเป็นผู้ใช้ ข้อมูลจากระบบไหน ต้องแม่นแค่ไหน ตรวจรับอย่างไร → แปลงเป็น requirement slices ที่ testable + traceable
- **Warning Signs:** รับคำแล้วใส่ใน scope หรือปฏิเสธโดยไม่มีทางเลือก

Senior PM

- **Q:** WBS ควรแตกลึกแค่ไหน?
- **Answer Direction:** หยุดที่ Work Package ที่ estimate/assign/วัด progress/accept/control ได้ — ลึกเกิน = กลายเป็น activity list (งานของ Ch.4); ไม่ลึกพอ = ควบคุมไม่ได้
- **Warning Signs:** ตอบ "แตกให้ละเอียดที่สุด" หรือ "ตามความรู้สึก"

Executive

- **Q:** ทำไม Scope Baseline ต้องได้รับการอนุมัติก่อนเริ่ม build?
- **Answer Direction:** Baseline = ฐานเทียบวัดทุกอย่าง (schedule/cost/change) — ถ้าไม่มี baseline การรายงานสถานะและ change control จะกลายเป็นความเห็น ไม่ใช่การวัดผล
- **Warning Signs:** ตอบว่า "เป็นกระบวนการของ PM" โดยไม่เชื่อมกับ governance

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Requirement	ข้อกำหนดความต้องการ	ต้อง Testable + Traceable	คิดว่า "ฟีเจอร์" คือ requirement	Scope, RTM
Product Scope	ขอบเขตผลิตภัณฑ์	คุณสมบัติของสิ่งที่จะส่งมอบ	ไม่แยกจาก Project Scope	Project Scope
Project Scope	ขอบเขตโครงการ	งานที่ต้องทำเพื่อส่งมอบ product	ไม่แยกจาก Product Scope	WBS
Scope Statement	คำนิยามขอบเขต	In/Out, deliverable, acceptance, constraint	เขียนยาวแต่ไม่มี boundary	Scope Baseline
WBS	Work Breakdown Structure	แตก approved scope เป็น work packages	แตกตาม department	Work Package
WBS Dictionary	พจนานุกรม WBS	owner, boundary, acceptance, dependency	คิดว่า WBS diagram พอแล้ว	WBS

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
100% Rule	กฎ 100%	WBS ครอบคลุม approved scope เท่านั้น	รวมทุกคำขอ	Scope Baseline
Scope Baseline	ฐานควบคุมขอบเขต	Scope Statement + WBS + Dictionary	สับสนกับ Project Plan	Validate/Control Scope
RTM	Requirements Traceability Matrix	เชื่อม requirement → design → test → acceptance	ทำเพื่อให้ครบ template	Requirement, Test Case
Validate Scope	ตรวจรับขอบเขต	ยืนยัน deliverable กับผู้มี authority	สับสนกับ Control Scope	Control Scope
Control Scope	ควบคุมขอบเขต	จัดการ change ต่อ baseline	สับสนกับ Validate Scope	Change Request
Scope Creep	ขอบเขตบาน	เพิ่มงานโดยไม่ผ่าน change control	สับสนกับ Progressive Elaboration	Change Request

10. Workshop

Scenario

[Teaching Scenario] คุณได้รับมอบหมายให้กำหนด Scope Phase 1 ของ SHG — คุณภัทร (VP Marketing) ต้องการ "dashboard ที่ดู conversion ทุกอย่าง" และ "loyalty program" ส่วนคุณวีระ (CTO) ต้องการให้ล็อก PMS sync เป็น workstream แยก

Your Role

PM (คุณสุทธิ) ที่ต้องสร้าง Scope Artifact Pack

Available Information

- Scenario Master v1.0 (In/Out of Scope MVP, 8 objectives, stakeholders, constraints), Stakeholder Register จาก Ch.2

Missing Information

- นิยาม "conversion ทุกอย่าง" ของคุณภัทร (KPI อะไร, ความถี่, ผู้ใช้), ข้อมูลราคา/ห้องว่างที่โรงแรมจะให้, ระดับ API ของ PMS แต่ละยี่ห้อ, งบ/เวลาที่แน่นอนของงาน dashboard

Decision Required

- ฟีเจอร์ใดอยู่ใน MVP, ฟีเจอร์ใด defer ไป Phase 2, acceptance criteria ของ dashboard คืออะไร

Constraints

- งบ 12M / launch ก่อน 1 พ.ย. / loyalty อยู่ใน Out of Scope ตาม Scenario Master

Expected Output

- Requirements List (≥ 8 รายการ มี priority + owner), Scope Statement (In/Out/Assumptions/Constraints), WBS fragment สำหรับ "Payment Integration" + WBS Dictionary 2 work packages

Debrief Questions

1. ทำไม "conversion ทุกอย่าง" ยังไม่ใช่ requirement
2. Acceptance criteria ที่ดีสำหรับ dashboard คืออะไร
3. ถ้าไม่มี RTM จะเกิดอะไรตอน Ch.8

Evaluation Criteria

- Requirement testable + traceable, Scope มี In/Out ชัด, WBS แตกตาม deliverable, decision record มี owner + evidence

11. Checklist ใช้งานจริง (Planning — Scope / Phase C1–C6)

- ☐ Delivery approach ถูก tailor ต่อ workstream (ไม่ใช่ "Agile ทั้งหมด")
- ☐ Requirements Management Plan กำหนดแล้ว (sources, techniques, approval, RTM, change)
- ☐ Requirements ครบทุกประเภทที่เกี่ยวข้อง (functional, non-functional, data, security, acceptance)
- ☐ Requirement ทุกข้อ Testable + Traceable + มี owner + priority
- ☐ SRS/FSD หรือสิ่งทดแทน ครบ coverage (ดู Substitution Matrix) — ไม่ตัดโดยไม่มีสิ่งทดแทน
- ☐ Scope Statement มี In/Out of Scope + Assumptions + Constraints + Acceptance Criteria
- ☐ WBS แตกตาม deliverable ผ่าน 100% Rule
- ☐ WBS Dictionary มี owner, acceptance, dependency ต่อ work package
- ☐ Scope Baseline ผ่าน approval (Sponsor/CCB)
- ☐ Cross-ref: Test Strategy/Quality Plan = Ch.5, Test Results/UAT = Ch.8

12. Assessment

1. **[Decision Case]** คุณภัทรขอเพิ่ม loyalty program ใน MVP — ตาม Scenario Master ข้อนี้อยู่ใน Out of Scope — คุณจะเสนออะไร (include/defer/replace) พร้อม evidence และ trade-off
2. **[Decision Case]** "รายงานทุกอย่าง" ถูกขอเพิ่ม — เขียนคำถาม 5 ข้อที่ต้องถามก่อนแปลงเป็น requirement
3. **[Artifact Review]** ตรวจ WBS ที่แตกเป็น "งานของ IT", "งานของ Marketing", "งานของ Finance" — อธิบายว่าทำไมไม่ใช่ WBS ที่ดี และแก้อย่างไร
4. **[Trade-off Case]** ระหว่าง SRS + FSD ฉบับเต็ม กับ User Story + Acceptance Criteria — อธิบาย trade-off สำหรับงานที่มี compliance สูง (payment/PCI-DSS)

5. **[Artifact Construction]** สร้าง WBS fragment สำหรับ "Payment Integration" ของ SHG (อย่างน้อย 5 work packages พร้อม owner + acceptance criteria)
6. **[Cross-Knowledge Analysis]** ถ้า PMS API ของ 3 โรงแรมไม่รองรับ real-time sync — กระทบ scope (Ch.3), schedule (Ch.4) และ test (Ch.8) อย่างไร
7. **[Recall]** 100% Rule ของ WBS คืออะไร และต่างจาก "รวมทุกคำขอ" อย่างไร

13. Executive Summary

มิติ	สรุป
Why	ขอบเขตที่ชัดคือฐานของ schedule, cost, quality และ change control ทั้งหมด
Decision Improved	คำขอใหม่ถูกเทียบกับ baseline + impact แทนการรับด้วยความเกรงใจ
Failure Prevented	ป้องกัน scope creep, requirement ที่ทดสอบไม่ได้ และ WBS ที่ควบคุมไม่ได้
What to Monitor	จำนวน change request ต่อ baseline, requirement coverage, acceptance criteria readiness

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: Scope Baseline ผ่าน approval แล้ว — Ch.4 จะใช้ work packages + owners + acceptance criteria ไปนิยาม activities, dependencies, duration, cost และ resource (และใส่กล่อง Agile สำหรับ sprint-based delivery)

Field	Handoff
Input	Stakeholder Register, Decision Rights (Ch.2), SOW (Ch.1)
Output	Requirements List, Scope Statement, WBS, WBS Dictionary, RTM
Creator	BA + PM (requirements), PM + Leads (WBS)
Artifact owner	PO (requirements), PM (scope/WBS)
Reviewer	Stakeholder owners, QA Lead, Functional Leads
Approval authority	Sponsor / CCB (scope baseline)
Minimum acceptance	Usable (ต้องมี In/Out + acceptance criteria + RTM)
Next chapter use	Ch.4 ใช้ work packages, owners, acceptance และ constraints เพื่อสร้าง Activity List, Schedule, Cost และ Resource Plan

15. Quick Reference Card

```
SCOPE (C1–C6) — เป้าหมาย: Scope Baseline ที่ควบคุมได้
------------------------------------------------------------
1. Tailor approach      -> Predictive/Agile/Hybrid ต่อ workstream
2. Plan requirements    -> sources, techniques, approval, RTM, change
3. Collect/analyze      -> 9 คุณภาพ: Clear, Testable, Traceable, Owned...
4. SRS/FSD              -> ดู coverage ไม่ใช่ชื่อไฟล์; ห้ามตัดโดยไม่มีตัวแทน
5. Define scope         -> Product vs Project, In/Out, acceptance criteria
6. WBS                  -> แตกตาม deliverable ผ่าน 100% Rule
7. WBS Dictionary       -> owner, acceptance, dependency ต่อ work package
8. Scope Baseline       -> Sponsor/CCB อนุมัติก่อนเริ่ม build

กฎ 3 ข้อห้าม:
- ห้ามรับ "ทั้งหมด/ทุกอย่าง" เป็น requirement
- ห้าม WBS แตกตามแผนก
- ห้ามลืม Out of Scope

Cross-ref: Test Strategy/Quality = Ch.5 | Test Results/UAT = Ch.8
```

Chapter 04 — Schedule, Cost และ Resource: แปลง Scope เป็นแผนที่วัดได้

1. Opening Scenario Hook

[Teaching Scenario] คุณสุทธิ (PM SHG) นั่งดู Gantt chart ที่ทีมวาดให้: มีแถบสีสวยงาม 12 Sprint ตาม Scenario Master แต่เมื่อคุณถามว่า "ถ้า Payment Integration ช้า 1 สัปดาห์ กระทบ launch วันที่ 1 พ.ย. หรือไม่" ไม่มีใครตอบได้ เพราะไม่มี dependency network, ไม่มี float, ไม่มี critical path

คุณวีระ (CTO) บอกว่า "ใช้เงิน 12 ล้านก็จบ" แต่คุณกาญจนา (Revenue Manager) ถามว่า "เงิน 12 ล้านนี่ใช้ตอนไหน และถ้าเร่งงานต้องเพิ่มงบเท่าไร" และคุณสมศรี (VP Ops) กังวลว่า "key users 80 คนมีงานประจำ จะมาทดสอบ UAT ได้จริงหรือ"

[PMBOK 8] บทนี้รวม 3 Knowledge Areas ที่ทำงานคู่กัน: Schedule (งานจะเสร็จเมื่อไร), Cost (ต้องใช้เงินเท่าไร ตอนไหน) และ Resource (ใครทำ มีเวลาจริงหรือไม่) — แยกดูทีละตัวก็เห็น แต่จะเห็นภาพจริงก็ต่อเมื่ออ่านพร้อมกัน

2. Why It Matters

[PMBOK 6] Schedule ที่ดีต้องตอบได้ว่า: ถ้างานหนึ่งช้า งานใดช้าตาม งานใดมีเวลาสำรอง และ recovery option ใดคุ้มค่า — Cost ที่ดีต้องอ่านเงินคู่กับงาน (Earned Value) ไม่ใช่ดูว่า "ใช้ไปเท่าไร" — Resource ที่ดีต้องแยก "มีชื่อในแผน" ออกจาก "มี capacity จริง"

[Best Practice] โครงการส่วนใหญ่พังไม่ใช่เพราะขาดแผน แต่เพราะแผนเป็นแค่ภาพ ไม่ใช่ระบบที่ตอบคำถาม "ถ้า...แล้ว...": ถ้า Payment ช้า แล้ว launch ล่าช้าไหม ถ้า key users ไม่ว่าง แล้ว UAT เลื่อนไหม ถ้าใช้เงินเกิน แล้ว baseline ไหนต้องเปลี่ยน

3. Mental Model

```
WBS Work Packages (Ch.3)
  -> Activity List (C8) — งานย่อยที่ลงมือทำได้
  -> Dependency Network (C9) — อะไรมาก่อน/หลัง
  -> Duration Estimate (C10) — effort vs duration ต้องแยก
  -> CPM + Float (C11) — critical path คือเส้นที่กำหนดโครงการ
  -> Schedule Baseline (C11) — ฐานวัด variance
  -> Cost Estimate + Time-phased Baseline (C12) — เงินตามเวลา
  -> EVM: PV / EV / AC (C12) — วัด performance ระหว่างทาง
  -> Resource Plan + RACI (C13) — ใครทำ มี capacity จริงหรือไม่
  -> Ch.5 (Risk/Procurement/Comms) + Ch.6 (Execution)
```

4. Main Lesson

4.1 Schedule — จาก Work Package เป็นกิจกรรม (C8–C11)

[PMBOK 6] เปลี่ยน Work Package (Ch.3) เป็น Activity List + Activity Attributes + Milestone List — WBS ไม่ใช่ Activity List

Dependency 4 แบบ (C9): Finish-to-Start (ทั่วไป), Finish-to-Finish, Start-to-Start, Start-to-Finish — แหล่งที่มา: Mandatory, Discretionary, External, Internal — อย่าลืม Lead (เริ่มก่อนจบ) และ Lag (เวลารอ)

CPM (C11):

```
Forward Pass: EF = ES + Duration
Backward Pass: LS = LF - Duration
Total Float: TF = LS - ES (หรือ LF - EF)
Critical Path = เส้นทางที่กำหนด Project Duration (โดยทั่วไป TF ต่ำสุด/เป็นศูนย์)
```

[Teaching Scenario] SHG timeline 8 เดือน (Sprint 0–12): ถ้า Payment Integration (Sprint 3–4) ช้า กระทบ Mobile App (Sprint 5–6) และ Back Office (Sprint 7–8) ตาม dependency — แต่ถ้าล่าช้าอยู่บนเส้นที่มี float ก็อาจไม่กระทบ launch

Schedule Compression (C11):

- **Crashing** = เพิ่มทรัพยากรเพื่อย่นเวลา → เพิ่มต้นทุน + coordination risk
- **Fast Tracking** = ทำงานขนาน → เพิ่ม rework + quality risk
- **Split Release / Phased** = แบ่งส่งมอบเป็น wave
- **Rebaseline** = ขออนุมัติเปลี่ยน baseline เมื่อ delay เกิน tolerance

4.2 Cost — Estimate, Baseline และ EVM (C12)

[PMBOK 6] แยกให้ชัด: Estimate (ประมาณการ) → Budget (วงเงินอนุมัติ) → Cost Baseline (budget แบ่งตามเวลาเพื่อควบคุม) → Contingency Reserve (known-unknowns) → Management Reserve (unknown-unknowns, ต้องขอ Sponsor ก่อนใช้)

EVM:

```
PV = Planned Value (งานที่ควรเสร็จตามแผน)
EV = Earned Value (งานที่เสร็จจริง ตาม evidence)
AC = Actual Cost (เงินที่ใช้จริง)
SV = EV - PV | CV = EV - AC
SPI = EV / PV | CPI = EV / AC
EAC = BAC / CPI (ถ้า trend เดิมต่อ) | VAC = BAC - EAC
```

[Best Practice] ตัวเลขต่ำกว่า 1 ไม่ใช่คำตอบสุดท้าย แต่เป็นประตูเข้าสู่คำถาม: variance มาจาก cost/schedule/completion? EV มี acceptance evidence จริงหรือ? เป็น one-time หรือ trend?

4.3 Resource — คน ทักษะ capacity และอำนาจ (C13)

[PMBOK 6] แยก Role (บทบาท) / Person (คนจริง) / Assignment (การมอบหมาย) / Capacity (เวลาจริง) — RACI: Responsible (ลงมือ), Accountable (รับผิดชอบผลสุดท้าย — ต้อง 1 คน), Consulted (ให้ input), Informed (รับรู้)

[Best Practice] Resource Calendar แสดง availability; Resource Histogram แสดง demand vs capacity — เมื่อเห็น peak/overload เริ่มคุย backfill, vendor support, resequence หรือ scope trade-off ด้วยหลักฐาน

[Teaching Scenario] SHG ต้องการ 80 key users (จาก 12 โรงแรม) สำหรับ UAT แต่ finance month-end close ซ้อนกับ UAT wave → ต้อง phased UAT + backfill หรือเลื่อน wave

4.4 กล่อง Agile: ถ้าโครงการเป็น Agile (แทรกจาก lesson-15/16)

[PMBOK 8] ถ้า delivery เป็น Agile (เช่น ส่วน UX/Booking Journey ของ SHG):

- **Scrum** สำหรับ product increment: Product Backlog (PO จัด priority), Sprint 2–4 สัปดาห์, Sprint Planning, Daily Scrum, Sprint Review, Retrospective — Sprint Goal ปกป้อง focus
- **Kanban** สำหรับ flow/support/incident: Flow Board (To Do/In Progress/Review/Done), WIP Limit, explicit policy — ใช้กับ defect triage, payment incident, support
- **Predictive vs Agile vs Hybrid:** ดู Requirement Stability, Change Cost, Compliance, Feedback Need — **Hybrid** = budget/milestones คุมแบบ predictive แต่ build เป็น sprint (ตรงกับ SHG: 12M + launch 1 พ.ย. คุมแน่น, development เป็น sprint)
- **Agile ไม่ได้แปลว่าไม่มีแผน** — เปลี่ยนจังหวะ planning ให้สั้นลง และต้องมี Definition of Done ที่รวม quality/security/acceptance

> **หมายเหตุ:** Tailoring ลึก ๆ กลับมาอีกครั้งใน Ch.6 (กล่อง Agile) — บทนี้แนะนำหลักการให้เห็นภาพการวางแผน

5. PM Decision Thinking

```
Decision: Schedule delay ควร recover within baseline, เปลี่ยน baseline หรือปรับ scope/release
Owner: PM วิเคราะห์; Sponsor/Steering ตัดสินเมื่อกระทบ milestone หรือ baseline
Inputs: WBS, activity list, dependency, duration, critical path, float, resource
availability, cost baseline, risk
Options:
  A: Use float + resequence — ราคาถูกสุด แต่ใช้ได้เมื่อ float จริง
  B: Crashing (เพิ่มคน/โอที) — ย่นเวลาแต่เพิ่ม cost + coordination risk
  C: Fast Tracking (ทำงานขนาน) — เร็วแต่เพิ่ม rework/quality risk
  D: Split Release (wave) — ลด waiting แต่เปลี่ยน user experience
  E: Rebaseline — ยอมรับ delay อย่างเป็นทางการ ต้อง Sponsor อนุมัติ
Trade-offs: speed vs cost, speed vs quality, milestone pressure vs readiness
Risk: recovery ที่ย่นวันแต่เพิ่ม defect -> go-live ยังไม่พร้อมอยู่ดี
Evidence: updated network, recovery brief, approved baseline change (ถ้า rebaseline)
Next Action: เลือก option -> ประเมินผลต่อ cost baseline + resource -> สื่อสาร decision -> Ch.5
(risk/procurement)
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario]

- **Schedule:** Sprint 0 (Setup, 2 สัปดาห์) → Sprint 1–10 (feature) → Sprint 11 (UAT/Performance/Security, 2 สัปดาห์) → Sprint 12 (Bug Fix + Soft Launch 3 โรงแรม) → Full Launch 12 โรงแรม → Post-Launch

Support 3 เดือน — critical path ต้องผ่าน Payment Integration → Mobile App → Back Office → UAT → Launch
- **Cost (12 ล้านบาท):** Dev Team 6.0, Cloud 1.0, Design 0.8, PMS Integration 0.5, Payment Setup 0.2, QA/Security 0.5, Contingency 1.5, Management Reserve 1.5 — Contingency = PM ใช้ได้ตาม governance, Management Reserve = Sponsor เท่านั้น
- **Resource:** ทีม Projectized (PO, PM/SM, UX 2, Frontend 4, Backend 3, QA 2, DevOps 1, Business 3) + External (PMS Specialist, 2C2P) — 80 key users ต้องมี capacity commitment จาก functional managers

[PMBOK 8] ตัวอย่าง EVM ระหว่างทาง: ถ้า ณ เดือนที่ 4 PV = 6M, EV = 5M, AC = 6.5M → SPI = 0.83, CPI = 0.77 → ต้องถามต่อ: งานที่ "เสร็จ" มี evidence หรือไม่ สาเหตุคืออะไร และ EAC จะเป็นเท่าไร (รายละเอียดการควบคุมอยู่ Ch.7)

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. Gantt chart ที่ไม่มี dependency/float/owner — ผล: ดูมั่นใจแต่ตอบคำถาม "ถ้า...แล้ว..." ไม่ได้
2. Effort กับ Duration ปนกัน (8 person-days ≠ 8 วัน) — ผล: schedule ผิดพลาด
3. Estimate โดย PM คนเดียว — ผล: ไม่เห็น skill/capacity gap
4. ใช้ Funding Envelope เป็น Cost Baseline — ผล: ควบคุมเงินผิด (ERP 45M vs 60M บทเรียนเดิม)
5. Resource plan มีชื่อคนครบ แต่ไม่มี capacity sign-off — ผล: UAT ว่างจริงแค่บางช่วง
6. RACI มี Accountable หลายคน — ผล: ไม่มีใครรับผิดชอบจริง
7. ใช้ Contingency/Management Reserve โดยไม่ผ่าน governance — ผล: สำรองหมดก่อนเจอของจริง

Common Misconceptions:

ความเข้าใจผิด	ความจริง
ใช้เงินน้อย = ดี	ต้องเทียบกับ EV และ schedule (อาจแค่ทำงานช้า)
เพิ่มคนแก้ delay ได้เสมอ	เพิ่ม ramp-up + coordination cost
Fast tracking ฟรี	เพิ่ม rework + quality risk
ใส่ชื่อคนครบ = มี resource	ต้องมี capacity, skill และ authority
Agile ไม่ต้องวางแผน	Agile วางแผนบ่อยและปรับจาก feedback

8. Interview Questions

Foundational

- **Q:** Critical Path คืออะไร และต่างจาก Milestone อย่างไร?
- **Answer Direction:** CP คือเส้นทางกิจกรรมที่กำหนดระยะเวลาโครงการ (float ต่ำสุด) ส่วน Milestone คือจุดสำคัญที่ไม่มี duration — งานบน CP ช้า = โครงการช้า
- **Warning Signs:** ตอบว่า "เส้นทางที่ยาวที่สุดในตาราง" โดยไม่อธิบาย float/dependency

Scenario

- **Q:** Payment Integration ช้า 1 สัปดาห์ ก่อน launch 1 พ.ย. คุณจะทำอะไร?
- **Answer Direction:** ไม่สรุปทันที — ดูว่า delay อยู่บน critical path หรือไม่, float ที่เหลือ, downstream dependency (Mobile App, Back Office), แล้วเสนอ options (float, crashing, fast track, split release) พร้อม trade-off ต่อ cost/resource/quality
- **Warning Signs:** ตอบ "เร่งทีม" โดยไม่มี mechanism หรือเลื่อน launch ทันทีโดยไม่วิเคราะห์

Senior PM

- **Q:** 80 key users ไม่ว่าง UAT เพราะงานประจำ คุณจะจัดการ capacity อย่างไร?
- **Answer Direction:** ดู resource histogram → เสนอ phased UAT ตาม module, backfill, vendor support สำหรับ test prep, หรือเลื่อน wave ที่ชน month-end — capacity commitment ต้องมาจาก functional managers ไม่ใช่ PM สั่งเอง
- **Warning Signs:** ตอบว่า "บังคับให้มา" หรือขอ Steering โดยไม่มีข้อมูล workload

Executive

- **Q:** CPI = 0.77 หมายความว่าอะไร และคุณจะรายงาน Sponsor อย่างไร?
- **Answer Direction:** อธิบายว่าใช้เงินไปแล้วได้งานกลับมาน้อยกว่าแผน (EV < AC) — ไม่ใช่แค่ "เงินหมด" แต่เป็น performance issue; เสนอ cause analysis + EAC forecast + options; รายงานแยก funding (12M) ออกจาก baseline/reserve
- **Warning Signs:** ปกป้องตัวเลข หรือพูดเฉพาะ AC โดยไม่เชื่อม EV

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Activity List	รายการกิจกรรม	งานจาก work packages ที่ใช้จัดตาราง	สับสนกับ WBS	WBS, Work Package
Dependency	ความสัมพันธ์ก่อนหลัง	FS/FF/SS/SF + lead/lag	มองข้าม external dependency	Network Diagram
Critical Path	สายงานวิกฤต	เส้นที่กำหนดระยะเวลา project	คิดว่าทุกงานช้า = ล่าช้าเท่ากัน	Float
Float / Slack	เวลาสำรอง	เลื่อนได้โดยไม่กระทบ milestone	สับสนกับ Buffer	Critical Path
Schedule Baseline	ฐานควบคุมเวลา	ตารางอนุมัติไว้ใช้วัด variance	สับสนกับ Gantt	Control Schedule
Crashing	เร่งด้วยทรัพยากร	เพิ่มคน/เงินเพื่อลดเวลา	คิดว่าฟรี	Fast Tracking
Fast Tracking	ทำงานขนาน	ซ้อนงาน เพิ่ม rework risk	คิดว่าฟรี	Crashing
Cost Baseline	ฐานควบคุมต้นทุน	งบแบ่งตามเวลาเพื่อควบคุม	สับสนกับ Funding	BAC, Budget

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Contingency Reserve	สำรอง risk ที่รู้	PM ใช้ตาม governance	สับสนกับ Management Reserve	Management Reserve
EVM	Earned Value Management	วัดงานที่เสร็จจริงเป็นเงิน	อ่านแค่ AC	PV, EV, AC, CPI, SPI
EAC	Estimate at Completion	forecast ต้นทุนจบงาน = BAC/CPI	คิดว่าเท่ากับ BAC เสมอ	VAC
RACI	ตารางบทบาท	Responsible/Accountable/Consulted/Informed	ใส่ A หลายคน	Decision Rights
Capacity	กำลังทำงานจริง	เวลาที่ใช้ได้จริง ไม่ใช่ชื่อในแผน	มีชื่อ = มี capacity	Resource Calendar
Sprint	รอบส่งมอบ Agile	2–4 สัปดาห์ มี Sprint Goal	คิดว่า Sprint = deadline ตามใจ	Product Backlog
WIP Limit	จำกัดงานค้าง	ป้องกันรับงานเกิน capacity	ยิ่งทำหลายงานยิ่งเร็ว	Kanban

10. Workshop

Scenario

[Teaching Scenario] ทีม SHG เจอว่า PMS Integration (Sprint 7–8) มีความเสี่ยงล่าช้า เพราะ PMS vendor หนึ่งในสามยี่ห้อตอบ API spec ช้า และ finance key users แจ้งว่าไม่ว่าง UAT ช่วงปลายเดือน — พร้อมกันนี้คุณกาญจนา (Revenue) ขอ dashboard เพิ่ม 2 ฟีเจอร์

Your Role

PM (คุณสุทธิ) ต้องสร้าง recovery + resource decision

Available Information

- Schedule baseline (Sprint 0–12), งบ 12M (Contingency 1.5M), Resource plan (ทีม 12 + business), Scenario Master risks

Missing Information

- API spec ของ PMS vendor ยี่ห้อที่ 3 จะมาถึงเมื่อไร, ความเห็นของ QA ว่า fast-track กระทบ test มากแค่ไหน, capacity sign-off จาก functional managers, ราคา/เวลาที่แน่นอนของ dashboard 2 ฟีเจอร์

Decision Required

- recovery option สำหรับ PMS delay, การจัดการ UAT capacity (phased/backfill), และ dashboard ควรไปไหน (ใน/นอก scope)

Constraints

- Launch ก่อน 1 พ.ย., งบ 12M (ใช้ contingency ต้องมี evidence + governance), key users มี BAU duties

Expected Output

- Schedule recovery brief (dependency + critical path + 3 options + trade-off), resource decision (UAT wave plan + owner + authority), decision record สำหรับ dashboard

Debrief Questions

1. ทำไมต้องรู้ว่า delay อยู่บน critical path ก่อนตัดสินใจ
2. Contingency reserve ใช้ตอนไหน และใครอนุมัติ
3. Resource decision ที่ดีต้องมี evidence อะไรบ้าง

Evaluation Criteria

- มี mechanism ต่อ recovery (ไม่ใช่ "เร่งทีม"), แยก EVM ถูกต้อง, RACI/capacity มี owner + authority, ตัวเลขไม่ขัด Scenario Master

11. Checklist ใช้งานจริง (Planning — Schedule/Cost/Resource / Phase C8–C13)

- ☐ Activity List สร้างจาก Work Packages (ไม่ใช่ WBS ซ้ำ)
- ☐ Dependency Network มี FS/FF/SS/SF + external dependencies
- ☐ Duration estimate แยก effort vs duration และมี basis
- ☐ CPM + Float วิเคราะห์แล้ว — critical path ชัดเจน
- ☐ Schedule Baseline ผ่าน approval
- ☐ Cost Estimate มี basis; Cost Baseline time-phased; reserve แยกชัด
- ☐ EVM setup (PV/EV/AC measurement rules + evidence definition) พร้อม
- ☐ Resource Plan มี capacity sign-off จาก functional managers
- ☐ RACI มี Accountable หนึ่งคนต่อ decision สำคัญ
- ☐ Resource histogram ไม่มี overload ที่ไม่มีการจัดการ
- ☐ Agile inserts: ถ้าใช้ Scrum/Kanban มี backlog owner + DoD + WIP policy
- ☐ ส่งต่อ Ch.5: risks, procurement constraints, communication plan

12. Assessment

1. **[Decision Case]** Payment Integration ช้า 1 สัปดาห์ — ให้ข้อมูลที่ต้องรู้ก่อนตัดสินใจ + 3 recovery options พร้อม trade-off ต่อ cost/resource/quality
2. **[Decision Case]** 80 key users ไม่ว่าง UAT ช่วงปลายเดือน — เสนอ resource options และ authority ที่ถูกต้อง
3. **[EVM Case]** ณ เดือนที่ 4: PV = 6M, EV = 5M, AC = 6.5M — คำนวณ SPI/CPI/EAC/VAC และระบุคำถามที่ต้องถามต่อ
4. **[Trade-off Case]** Crashing vs Fast Tracking สำหรับ PMS delay — อธิบาย trade-off และเลือกพร้อมเหตุผล
5. **[Artifact Review]** ตรวจ Gantt chart ที่ไม่มี dependency/owner/estimate basis — ระบุ decision risk
6. **[Artifact Construction]** สร้าง Resource Plan + RACI fragment สำหรับ UAT (SHG) — ระบุ capacity, skill, authority
7. **[Recall]** Effort กับ Duration ต่างกันอย่างไร และทำไม 8 person-days ไม่เท่ากับ 8 วัน

13. Executive Summary

มิติ	สรุป
Why	Schedule/Cost/Resource คือสามเหลี่ยมที่ทำให้ scope กลายเป็นแผนที่วัดและควบคุมได้
Decision Improved	Recovery, cost และ capacity ตัดสินด้วย critical path/EVM/data แทนความรู้สึก
Failure Prevented	ป้องกัน Gantt หลอก, เงินเกินโดยไม่รู้, และ resource ที่มีชื่อแต่ไม่มี capacity
What to Monitor	SPI/CPI trend, critical path health, resource histogram, reserve usage

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: แผนเวลา/เงิน/คนผ่าน baseline แล้ว — Ch.5 จะเพิ่ม Risk, Procurement, Communications, Environment และรวมเป็น Integrated Plan ก่อน Planning Gate (C14–C19)

Field	Handoff
Input	WBS, Scope Baseline, Acceptance Criteria (Ch.3)
Output	Activity List, Dependency Network, Schedule Baseline, Cost Baseline, Resource Plan + RACI
Creator	PM + Planner / PM + Finance / PM + Functional Leads
Artifact owner	PM (schedule), PM + Finance (cost), PM (resource plan)
Reviewer	Functional Leads, Commercial, QA
Approval authority	Sponsor / Steering (baseline), Functional Managers (capacity)
Minimum acceptance	Usable (ต้องมี baseline + capacity sign-off + EVM rules)
Next chapter use	Ch.5 ใช้ baselines + resource ไปวาง Risk, Procurement, Communications และ Environment

15. Quick Reference Card

```
SCHEDULE/COST/RESOURCE (C8–C13)
------------------------------------------------------------
Schedule:
  WP -> Activity List -> Dependency -> Duration -> CPM/Float -> Baseline
  Delay? ดู critical path + float ก่อน -> use float / crash / fast-track / split / rebaseline
Cost:
  Estimate -> Budget -> Time-phased Baseline -> Contingency/Management Reserve
  EVM: PV/EV/AC -> SV/CV/SPI/CPI -> EAC = BAC/CPI -> VAC = BAC-EAC
Resource:
  Role vs Person vs Assignment vs Capacity
  RACI: A หนึ่งคนต่อ decision | Histogram เจอ overload -> backfill/vendor/resequence
Agile insert:
  Scrum = increment (Sprint Goal + DoD) | Kanban = flow (WIP limit + policy)
  Hybrid = predictive control + sprint build (ตรงกับ SHG)

กฎ 3 ข้อห้าม:
- ห้าม Gantt ที่ไม่มี logic
- ห้ามอ่านเงินโดยไม่ดู EV
- ห้ามมีชื่อคนโดยไม่มี capacity sign-off
```

Chapter 05 — Risk, Procurement, Communications, Environment IIa: Planning Gate

1. Opening Scenario Hook

[Teaching Scenario] ใกล้จบช่วงวางแผน คุณสุทธิ (PM SHG) นั่งรวมแผนย่อยเข้าด้วยกัน: Schedule + Cost + Resource ผ่านแล้ว แต่ยังมีคำถามค้าง: PMS integration มีความเสี่ยงสูง (Scenario Master Risk #1) จะจัดการอย่างไร 2C2P contract เงื่อนไขอะไร ใครต้องรู้เรื่องไหนเป็นรายสัปดาห์ และที่สำคัญ — **Test Strategy ยังไม่ถูกล็อก** ทั้งที่ Ch.8 จะต้องใช้ตรวจรับผลงาน

คุณวีระ (CTO) ถามว่า "เราเตรียม Test Plan ไว้หรือยัง" คุณกาญจนา (Revenue) ถามว่า "รายงานสถานะใครเห็นบ้าง" และคุณสมศรี (VP Ops) ถามว่า "ถ้าระบบล่ม ใครรับผิดชอบ"

[PMBOK 8] บทนี้คือ "ปิดจบการวางแผน": วาง Risk, Procurement, Communications, Environment และ Quality/Test Strategy แล้วรวมเป็น Integrated Plan ผ่าน **Planning Gate (C19)** — ผ่านตรงนี้แล้วทีมถึงจะเริ่ม Execution (Ch.6)

2. Why It Matters

[PMBOK 6] Risk ที่ไม่วางแผน = เจอ issue ตอนช้าเกินไป Procurement ที่ไม่วางแผน = สัญญาที่ไม่รองรับการส่งมอบจริง Communications ที่ไม่วางแผน = ข้อมูลไปผิดคน/ผิดเวลา และ Test Strategy ที่ไม่วางแผน = UAT (Ch.8) เถียงกันว่า "แบบไหนถึงจะผ่าน"

[Best Practice] แผนย่อยแต่ละอันดูดีได้คนเดียว แต่ Integrated Plan คือการทำให้เห็นว่า Risk กระทบ Schedule อย่างไร Procurement กระทบ Cost อย่างไร Test Strategy กระทบ Resource อย่างไร — นี่คือหัวใจของ Integration ที่ PM ต้องถือภาพรวม (ต่อจาก Ch.2, กลับมาอีกครั้งใน Ch.7 และจบที่ Ch.10)

3. Mental Model

```
Ch.4 Baselines (schedule/cost/resource)
  -> Risk Planning (C15) — register, response, owner, trigger
  -> Procurement Planning (C16) — make-or-buy, contract type, vendor
  -> Communications Planning (C14) — ใครรู้อะไร เมื่อไร ผ่านช่องทางใด
  -> Quality/Test Strategy (C7) — วางแผนไว้ตรงนี้ ผลจริงอยู่ Ch.8
  -> Environment/Security/Release/Transition (C17) — env, security, cutover เบื้องต้น
  -> Integrated PM Plan (C18) — รวมทุกแผนเป็นหนึ่งเดียว
  -> Planning Gate (C19) — Sponsor/CCB อนุมัติ -> Ch.6 Execution
```

> **Cross-reference (Quality): Test Strategy / Quality Plan วางแผนที่นี่ (Ch.5) — ผลการทดสอบจริง/UAT Evidence อยู่ Ch.8** อย่าสับสนสองจุดนี้

4. Main Lesson

4.1 Risk Planning (C15) — จาก lesson-13

[PMBOK 6] กระบวนการ Risk: Plan → Identify → Qualitative Analysis → Quantitative (เมื่อจำเป็น) → Plan Responses → Implement → Monitor

Risk statement ต้องมี Cause + Event + Impact:

```
"หาก data cleansing ของ legacy systems ไม่ผ่าน threshold ก่อน migration rehearsal
UAT จะใช้ข้อมูลที่ไม่น่าเชื่อถือ ทำให้ defect เพิ่มและ go-live เสี่ยง"
```

Risk fields: ID, Cause, Risk Event, Impact, Probability, Impact Rating, Exposure, Trigger, Owner, Response, Contingency, Residual Risk, Status

Response strategies:

- Threat: Avoid > Mitigate > Transfer > Accept (+ Escalate)
- Opportunity: Exploit, Enhance, Share, Accept (+ Escalate)

[Best Practice] Accept ≠ ไม่ทำอะไร — Active acceptance มี contingency reserve, trigger และ fallback plan — Risk Owner ควรเป็นคนที่รู้/มีอำนาจต่อ risk นั้น ไม่ใช่ PM เสมอไป

4.2 Procurement and Vendor Planning (C16) — จาก lesson-14

[Best Practice] เริ่มจาก Need และ Risk ไม่ใช่จาก contract type — Make-or-Buy ต้องมี rationale

Contract Type	เหมาะกับ	ต้องมี control
Fixed Price	Scope ชัด, vendor รับ cost risk	Scope ที่ชัดพอ + change control
Time and Material	Scope ไม่ชัด, ต้องเรียนรู้	Rate, Cap, Approval Cadence, Acceptance
Hybrid	บางส่วนชัด บางส่วนไม่ชัด	Fixed core + T&M cap สำหรับส่วน discovery

[Teaching Scenario] SHG: 2C2P (transaction fee 2.5% + 7 บาท/txn) — ต้องดูไม่แค่ fee แต่ SLA, security obligation, settlement, incident response, refund process และ integration support; PMS Vendors (T&M ~500K รวม) — ต้องมี cap + acceptance; AWS (pay-as-you-go ~120K/เดือน) — ต้องมี cost forecast + monitoring

4.3 Communications Planning (C14) — จาก lesson-12

[PMBOK 6] Communication Plan (5W1H): Audience, Information, Purpose, Format, Frequency, Owner, Channel, Escalation, Confidentiality

3 รูปแบบ: Push (email/report), Pull (dashboard/wiki), Interactive (meeting/workshop)

[Best Practice] แยก **Status Report** (บอกสถานะตามรอบ) ออกจาก **Decision Brief** (ขอ decision พร้อม evidence + deadline) — Dashboard สีแดงไม่ใช่ escalation ถ้าไม่มี decision ask

[Teaching Scenario] SHG cadence: Daily Team Sync (dev), Weekly Status (Steering + Stakeholders), Sprint Review (PO/คุณนภา + business), Risk Review (รายสัปดาห์), UAT Defect Review (ช่วง Ch.8), Go-live Command Center (Ch.9)

4.4 Quality / Test Strategy Planning (C7) — วางแผนไว้ที่นี่

[PMBOK 6] Plan Quality Management: กำหนด Quality Standard, Acceptance Criteria, Test Strategy, Review Approach, Verification vs Validation, Defect Process, Acceptance Authority, Evidence

- **Verification** = สร้างถูกตาม spec (build the product right)
- **Validation** = สร้างสิ่งที่ตอบ need (build the right product)
- **Cost of Quality** = Prevention + Appraisal (ต้นทุนทำให้ถูก) vs Failure (rework, incident, lost trust)
- **Test levels (F.2):** Unit → Component → Integration → System/SIT → Performance → Security → Regression → UAT → Production Verification

[Teaching Scenario] SHG Test Strategy ควรระบุ: SIT สำหรับ booking flow + PMS sync, Performance Test (3 วินาที load, 5 วินาที confirmation, uptime 99.5%), Security/Penetration Test (PCI-DSS, PDPA), UAT ด้วย key users 80 คน, Regression ก่อน release — **ผลจริงของทั้งหมดนี้ถูกตรวจและรายงานใน Ch.8**

4.5 Environment, Security, Release, Transition Planning (C17)

[Best Practice] Environment Plan: Dev/Test/SIT/UAT/Staging/Production/DR — Security: access, auth, encryption, logging, vulnerability, privacy, backup, incident — Transition: data migration, training, cutover, support, rollback (แผนเต็มของ rollout อยู่ Ch.9)

4.6 Integrated PM Plan (C18) + Planning Gate (C19)

[PMBOK 6] Integrated Plan แสดง Scope/Release, Timeline, Cost, Resource, Quality, Risk, Communication, Procurement, Change, Transition, Governance — ไม่ใช่มองเป็นแผนแยก

[Best Practice] Gate C (Ready for Execution) ตรวจ: Requirements คุณภาพ, Scope/Release boundary, WBS/Backlog, Acceptance Criteria, Schedule, Resource Commit, Cost Baseline, Risks มี Owner, Client Dependencies, Environment, Change Process, Governance, ทีมเห็นแผนเดียวกัน — ผ่านแล้วถึงเริ่ม Execution

5. PM Decision Thinking

```
Decision: ผ่าน Planning Gate (C19) หรือต้องปิด gap ใดก่อนเริ่ม Execution
Owner: PM นำเสนอ; Sponsor/CCB อนุมัติ baselines + gate
Inputs: integrated plan (ทุก subsidiary plan), risk register, procurement, test strategy,
readiness checklist (Playbook §7)
Options:
  A: ผ่าน gate ทันที — ทุกอย่างครบ
  B: ผ่านแบบมีเงื่อนไข (conditional) — ระบุ gap + owner + due date เช่น Test Strategy ยังต้องล็อก UAT
data
  C: ไม่ผ่าน กลับไปแก้ — เหมาะเมื่อ baseline อ่อนจน Execution จะสร้างของเสีย
Trade-offs: momentum vs readiness, schedule pressure vs rework cost
Risk: ผ่าน gate ทั้งที่ baseline ไม่พร้อม -> Ch.6–8 เจอ defect/dispute มาก
Evidence: gate review record, approved baselines, open-gap log
Next Action: ถ้าผ่าน -> Kickoff Execution (Ch.6); ถ้าไม่ -> ปิด gap ตาม due date แล้ว review ใหม่
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario]

- **Risk Register (ตัวอย่าง 2 รายการ):**
- R-01: PMS API ไม่พร้อม (High/High) — Mitigate: PoC integration ก่อน Sprint 1, daily data review, wave migration rehearsal — Owner: คุณวีระ (CTO)
- R-02: Payment Security Breach (Low/Critical) — Avoid: PCI-DSS audit ก่อน launch — Owner: คุณวีระ
- **Procurement:** 2C2P (transaction fee + SLA + incident response), PMS Vendors (T&M cap 500K), AWS (pay-as-you-go + forecast), Design Agency (Fixed 800K)
- **Communications:** Weekly Status ถึง Steering (คุณจิรา, คุณภัทร, คุณวีระ, PM), Sprint Review กับ PO + business, Daily Team Sync ภายในทีม, Risk Review รายสัปดาห์
- **Test Strategy:** SIT + Performance (3s/5s/99.5%) + Security (PCI-DSS/PDPA) + UAT (key users 80) + Regression — ผลจริง → Ch.8
- **Planning Gate:** Sponsor อนุมัติ baselines + เปิดรายการ gap (เช่น test data ยังไม่ครบ) พร้อม due date — ผ่านแบบ conditional

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. Risk register เขียน "ทีมไม่พร้อม" แบบกว้าง ไม่มี trigger/owner — ผล: monitor ไม่ได้
2. Risk ทำครั้งเดียวตอนวางแผนแล้วเก็บ — ผล: risk กลายเป็น issue โดยไม่รู้ตัว
3. เลือก contract type จากความเคยชิน ไม่ใช่ scope maturity — ผล: dispute/บานปลาย
4. Payment release โดยไม่มี acceptance evidence — ผล: เสียทั้งเงินและอำนาจควบคุม vendor
5. ส่ง report เดียวกันให้ทุกคน — ผล: ไม่มีใครได้สิ่งที่ต้องใช้ตัดสินใจ
6. Test Strategy ไม่ถูกล็อกตอนวางแผน — ผล: Ch.8 เถียงว่า "แบบไหนถึงจะผ่าน"

7. ผ่าน Planning Gate ทั้งที่ baseline ยังอ่อน — ผล: ของเสียถ่ายไป Execution

Common Misconceptions:

ความเข้าใจผิด	ความจริง
Risk คือสิ่งไม่ดีเท่านั้น	รวม opportunities (exploit/enhance/share)
Risk Owner = PM เสมอ	ควรเป็นคนที่มีความรู้/อำนาจต่อ risk นั้น
Accept = ไม่ทำอะไร	Active accept มี reserve + trigger + fallback
Procurement Plan = Contract	Plan คือ strategy; contract คือ binding agreement
Vendor ทำงานเสร็จ = accept	ต้องมี acceptance evidence
Dashboard สีแดง = escalation	Escalation ต้องมี decision ask + response
QA รับผิดชอบคุณภาพคนเดียว	คุณภาพเริ่มจาก requirement (Ch.3) และวางแผนที่นี่

8. Interview Questions

Foundational

- **Q:** Risk ต่างจาก Issue อย่างไร?
- **Answer Direction:** Risk = เหตุการณ์ยังไม่เกิด (มี trigger, owner, response); Issue = เกิดแล้ว (ต้องแก้/escalate ตอนนี้)
- **Warning Signs:** ใช้คำสลับกัน หรือเรียกทุกอย่างว่า risk เพื่อเลี่ยง escalation

Scenario

- **Q:** PMS integration risk สูง ก่อนเริ่ม Sprint 1 — คุณจะทำอะไร?
- **Answer Direction:** ระบุ trigger ที่วัดได้ (เช่น API doc ล่าช้าเกิน X วัน, PoC ไม่ผ่าน) → เลือก response (mitigate: PoC ก่อน, vendor support เพิ่ม) → ตั้ง owner (คุณวีระ) → กำหนด review cadence และ escalation path
- **Warning Signs:** เขียน "เราจะระวัง" โดยไม่มี trigger/owner/response

Senior PM

- **Q:** 2C2P ค่า fee ถูกที่สุดแต่ SLA เรื่อง incident response อ่อนแอ — คุณจะเลือกอย่างไร?
- **Answer Direction:** ดู Total Cost of Ownership + risk: fee ต่ำแต่ถ้า incident กลาง campaign ทำ revenue เสียหาย — นำ SLA/incident requirement ไป negotiation หรือเลือก provider ที่ trade-off ดีกว่า; บันทึก commercial decision record
- **Warning Signs:** เลือกจาก fee อย่างเดียว

Executive

- **Q:** ทำไมต้องมี Planning Gate ก่อนเริ่ม build?

- **Answer Direction:** Gate = จุดที่ Sponsor ยืนยันว่า baselines + แผนย่อยพร้อมและเห็นตรงกัน — เปลี่ยน "ความตั้งใจ" เป็น "คำมั่น" ที่วัดผลได้; ช่วยหยุดของเสียก่อนเริ่มลงทุน Execution
- **Warning Signs:** ตอบว่า "เป็นขั้นตอน PM" โดยไม่เชื่อมกับ control/risk

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Risk	ความเสี่ยง	เหตุการณ์ไม่แน่นอนที่กระทบ objective	สับสนกับ Issue	Issue, Trigger
Issue	ปัญหาที่เกิดแล้ว	ต้องมี owner, action, escalation	สับสนกับ Risk	Risk
Trigger	สัญญาณเตือน	เงื่อนไขที่บอกว่า risk ใกล้เกิด/เกิดแล้ว	ไม่กำหนด → monitor ไม่ได้	Risk Owner
Mitigate	ลดความเสี่ยง	ลด probability หรือ impact	คิดว่า = กำจัดได้หมด	Avoid, Transfer
Residual Risk	ความเสี่ยงคงเหลือ	risk หลัง response — ต้องมี owner	ลืมว่ายังเหลืออยู่	Risk
Make-or-Buy	ทำเองหรือซื้อ	ตัดสินจาก core, cost, risk, capacity	เลือกจากความเคยชิน	Procurement
SOW	ขอบเขตงานตามสัญญา	ผูกพันตาม contract	สับสนกับ Proposal	Contract
Fixed Price	ราคาคงที่	scope ชัด vendor รับ cost risk	ใช้กับ scope ไม่ชัด	T&M
T&M	Time and Material	จ่ายตามเวลา/วัสดุ	ไม่มี cap/evidence	Fixed Price
Communication Plan	แผนสื่อสาร	ผู้รับ ข้อมูล ความถี่ ช่องทาง owner	ทำครั้งเดียวแล้วเก็บ	Stakeholder
Status Report	รายงานสถานะ	บอกสถานะตามรอบ	สับสนกับ Decision Brief	Decision Brief
Decision Brief	บันทึกขอตัดสินใจ	decision ask + evidence + deadline	สับสนกับ Status Report	Escalation
Test Strategy	แนวทางการทดสอบ	test levels, env, data, entry/exit, defect	คิดว่า = Test Plan รายละเอียด	RTM, UAT
Verification	ตรวจตาม spec	build the product right	สับสนกับ Validation	Validation
Validation	ตรวจว่าตอบ need	build the right product	สับสนกับ Verification	Verification
Planning Gate	ประตูอนุมัติแผน	Sponsor ยืนยันพร้อมเริ่ม Execution	ผ่านทั้งที่ baseline อ่อน	Baseline

10. Workshop

Scenario

[Teaching Scenario] ก่อน Planning Gate ทีม SHG พบว่า: (1) Test Strategy ยังไม่ระบุ UAT data source และ exit criteria, (2) 2C2P ยังไม่ยืนยัน SLA ด้าน incident response, (3) Risk "key users ไม่ว่าง UAT" ยังไม่มี trigger/owner

Your Role

PM (คุณสุทธิ) ที่ต้องปิด gap ก่อนเสนอ Gate ต่อ Sponsor

Available Information

- Baselines (schedule/cost/resource จาก Ch.4), Scenario Master risks, draft communication plan

Missing Information

- UAT data ใครเตรียม (real data จาก PMS?), exit criteria ที่ PO/คุณภัทร จะ accept, SLA ของ 2C2P เรื่อง incident, ตัวเลข trigger สำหรับ key-user availability

Decision Required

- ผ่าน gate แบบเต็ม, ผ่านแบบ conditional (ระบุ gap + due date), หรือเลื่อน gate — และใครเป็น owner ของแต่ละ gap

Constraints

- Launch ก่อน 1 พ.ย. (เลื่อน gate = กินเวลา), งบ 12M, ห้ามเริ่ม Execution โดยที่ risk หลักไม่มี owner

Expected Output

- Gate recommendation (A/B/C + เหตุผล), Gap Log (gap, owner, due date, evidence), Risk Register อัปเดต 3 รายการ, Test Strategy outline หน้าเดียว

Debrief Questions

1. Conditional approval ต่างจาก "ผ่านเลย" อย่างไร และเมื่อไรถึงสมเหตุสมผล
2. ทำไม Test Strategy ต้องล็อกก่อน Execution ทั้งที่ผลอยู่ Ch.8
3. ถ้า Sponsor เร่งให้ผ่าน gate จะมีหลักฐานอะไรช่วย defend

Evaluation Criteria

- Gate decision มี evidence + owner + due date, risk มี trigger/owner, test strategy ครบองค์ประกอบ (scope, env, data, entry/exit, authority)

11. Checklist ใช้งานจริง (Planning — Risk/Procurement/Comms/Quality/Env / Phase C7, C14–C19)

- ☐ Risk Register ครบ: ทุก risk มี cause/event/impact, trigger, owner, response, residual, review cadence
- ☐ Risk ที่กระทบ baseline มี escalation/change path ชัด
- ☐ Procurement Plan: make-or-buy rationale, contract type ตาม scope maturity, payment ผูก acceptance evidence
- ☐ Vendor evaluation ดู SLA/security/incident ไม่ใช่ราคาอย่างเดียว
- ☐ Communication Plan: audience, info, format, frequency, owner, channel, escalation
- ☐ Status Report ต่างจาก Decision Brief ถูกแยกใช้
- ☐ Quality/Test Strategy วางแผนแล้ว: test levels, env, data, entry/exit, acceptance authority
- ☐ Environment/Security/Release/Transition Plan (C17) มีแล้ว (รายละเอียด rollout → Ch.9)
- ☐ Integrated PM Plan เชื่อมทุก subsidiary plan
- ☐ Planning Gate (C19) ผ่านพร้อม evidence + gap log
- ☐ Cross-ref: ผลทดสอบจริง/acceptance evidence = Ch.8

12. Assessment

1. **[Decision Case]** Risk "key users ไม่ว่าง UAT" — เขียน risk statement, trigger, owner, response, residual risk และ escalation path
2. **[Decision Case]** 2C2P เสนอ fee ต่ำแต่ SLA อ่อน — ควรเลือก/negotiate อย่างไร พร้อม commercial decision record
3. **[Trade-off Case]** ระหว่างผ่าน Planning Gate แบบ conditional กับเลื่อน gate 2 สัปดาห์ — อธิบาย trade-off ต่อ launch 1 พ.ย. และให้คำแนะนำ
4. **[Artifact Review]** ตรวจ Risk Register ที่เขียน "ทีมไม่พร้อม", "data อาจมีปัญหา" — แก้เป็น actionable risks
5. **[Executive Communication]** เขียน 1 ย่อหน้าให้ Sponsor ว่า ทำไม Test Strategy ต้องถูกล็อกก่อนเริ่ม build (แม้ผลจะอยู่ Ch.8)
6. **[Cross-Knowledge Analysis]** ถ้า 2C2P ยืนยันว่า settlement ใช้เวลา 2 วันทำการ (ต่างจากที่คาด 1 วัน) — กระทบ requirement (Ch.3), test (Ch.5/Ch.8) และ operating model (Ch.9) อย่างไร
7. **[Recall]** ความต่างระหว่าง Verification กับ Validation (พร้อมตัวอย่างจาก SHG)

13. Executive Summary

มิติ	สรุป
Why	การวางแผนที่ดีจบที่ Integrated Plan ที่ผ่าน Gate — เปลี่ยนความตั้งใจเป็นคำมั่นที่วัดได้
Decision Improved	Risk/procurement/comms/test ตัดสินด้วย evidence + owner + trigger
Failure Prevented	ป้องกัน risk ไร้เจ้าของ, สัญญาไม่รองรับงานจริง, ข้อมูลไปผิดคน, และ UAT ที่เถียงกันตอนท้าย

มิติ	สรุป
What to Monitor	Gap log ของ conditional approval, risk trigger, vendor SLA compliance, test readiness

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: Planning Gate ผ่าน — ทีมมี baselines + integrated plan พร้อม — Ch.6 (Execution) จะลงมือสร้าง deliverables ตามแผน พร้อมบริหารทีม คุณภาพ (ฝั่ง Manage Quality) และจัดการ blocker รายวัน

Field	Handoff
Input	Baselines จาก Ch.4, Scope Baseline (Ch.3), Charter/Governance (Ch.2)
Output	Risk Register, Procurement Plan, Communication Plan, Test Strategy, Environment Plan, Integrated PM Plan, Gate Approval
Creator	PM + Owners (risk/comms), Procurement Manager (procurement), QA Lead (test strategy)
Artifact owner	ตามแผนย่อย (Risk Owners, Procurement Mgr, QA Lead) / PM สำหรับ integrated plan
Reviewer	Sponsor, Leads, Legal, Finance
Approval authority	Sponsor / CCB (gate + baselines)
Minimum acceptance	Usable (ทุกแผนย่อยมี owner + evidence)
Next chapter use	Ch.6 ใช้ integrated plan + test strategy + communication plan ไปบริหาร Execution

15. Quick Reference Card

```
RISK/PROC/COMMS/QUALITY/ENV + GATE (C7, C14–C19)
----------------------------------------------------------------
Risk:     cause/event/impact + trigger + owner + response + residual
          Threat: Avoid > Mitigate > Transfer > Accept | Opp: Exploit/Enhance/Share
Procure:  Make-or-Buy -> contract type ตาม scope maturity -> payment ผูก acceptance
Comms:    5W1H + Push/Pull/Interactive | Status Report ≠ Decision Brief
Quality:  Test Strategy วางแผนที่นี่ -> ผลจริง Ch.8 (Ver = ตาม spec, Val = ตาม need)
Env:      Dev/Test/SIT/UAT/Staging/Prod/DR + Security + Transition (rollout = Ch.9)
Gate:     C19 — Sponsor อนุมัติ baselines + gap log ก่อน Execution

กฎ 3 ข้อห้าม:
- ห้าม Risk register ไม่มี trigger/owner
- ห้ามเลือก contract จากความเคยชิน
- ห้ามผ่าน Gate โดย Test Strategy ยังไม่ล็อก
```

Chapter 06 — Execution: ลงมือสร้าง พร้อมรักษาคุณภาพและทีม

1. Opening Scenario Hook

[Teaching Scenario] Planning Gate ผ่านแล้ว ทีม SHG เริ่ม Sprint 1 — คุณสุทธิ (PM) เห็น board เต็มไปด้วยงาน แต่เมื่อคุณถาม QA ว่า "payment flow จะเริ่ม test เมื่อไร" QA ตอบว่า "เดี๋ยวค่อยดูตอนใกล้เสร็จ" และเมื่อคุณถาม Developer คนหนึ่งว่า "ทำไม story นี้ยังไม่ Done" เขาตอบว่า "code เขียนเสร็จแล้วไง"

[PMBOK 8] นี่คือสองกับดักของ Execution: (1) คุณภาพที่คิดว่า "มาทำตอนท้าย" แทนที่จะฝังไว้ในกระบวนการ (Manage Quality / QA) และ (2) นิยาม "เสร็จ" ที่ไม่ตรงกันระหว่างคนเขียน คนทดสอบ และคนรับงาน (Definition of Done) — Execution ไม่ใช่แค่ "เขียน code ให้เสร็จ" แต่คือการสร้าง deliverables พร้อม evidence ผ่านกระบวนการที่ป้องกัน defect

2. Why It Matters

[PMBOK 6] Executing คือที่ที่แผนกลายเป็นของจริง — Direct and Manage Project Work, Manage Quality, Develop/Manage Team, Manage Communications, Implement Risk Responses, Conduct Procurements, Manage Stakeholder Engagement

[Best Practice] ถ้า Execution ไม่มีวินัย (ไม่มี DoR/DoD, QA มาทีหลัง, blocker ไม่ถูก escalate) ปัญหาจะสะสมไประเบิดที่ Ch.8 (UAT) และ Ch.9 (Go-live) — ต้นทุนการแก้ตอนนั้นแพงกว่าการป้องกันตอนนี้หลายเท่า

3. Mental Model

```
Integrated Plan (Ch.5) + Baselines (Ch.4)
  -> Confirm Ready Work (DoR) — งานพร้อมเริ่มหรือยัง
  -> Build/Configure/Integrate
  -> Peer Review -> Test (ระหว่างทาง)
  -> Definition of Done — ครบทุกเกณฑ์ ไม่ใช่ "code เสร็จ"
  -> Work Performance Data -> Ch.7 (Monitoring)
  -> Deliverable -> Ch.8 (Verification/UAT)
ควบคู่กัน:
  Manage Quality (QA) — กระบวนการป้องกัน defect
  Develop/Manage Team — Tuckman, conflict, motivation
  Manage Comms/Risk/Stakeholder — ตามแผน Ch.5
```

> **Cross-reference (Quality):** แผนคุณภาพ/Test Strategy = Ch.5, **Manage Quality (QA ฝั่งกระบวนการ) = บทนี้**, **ผลการทดสอบจริง/QC = Ch.8**

4. Main Lesson

4.1 Execution Flow (D.4)

[Best Practice]

- **Predictive:** Analyze → Design → Build → Test → UAT → Deploy
- **Agile:** Refine → Plan Sprint → Build/Test → Review → Retro → Release
- **Hybrid:** Fixed Milestones + Iterative Development + Formal Gates (ตรงกับ SHG)

4.2 Definition of Ready (DoR) และ Definition of Done (DoD) (D.6–D.7)

[Best Practice]

- **DoR (พร้อมเริ่ม):** Purpose, Scope, Requirement, Acceptance Criteria, Dependencies, Design/Decision, Test Data, Owner, Estimate, Priority
- **DoD (เสร็จจริง):** Build Complete, Code Review, Unit Test, Integration Test, Security Check, Documentation, Acceptance Criteria Pass, Defect Threshold Met, Deployed to Required Environment, Evidence Captured

[Teaching Scenario] ที่ SHG: story "Payment Flow" ยังไม่ Done แม้ code เสร็จ เพราะยังขาด unit test, security check และ deployment ไป test environment — DoD ช่วยให้ทุกคนพูดคำว่า "เสร็จ" หมายถึงสิ่งเดียวกัน

4.3 Manage Quality / QA — ฝั่งกระบวนการ (จาก lesson-10)

[PMBOK 6] Manage Quality (QA) = ดูว่ากระบวนการช่วยป้องกันปัญหาหรือไม่: design review, data migration checklist, test readiness gate, root-cause review, process improvement — **ไม่ใช่** การตรวจผลลัพธ์ (นั่นคือ Control Quality → Ch.8)

[Best Practice] Cost of Quality: Prevention/Appraisal (จ่ายเพื่อทำให้ถูก) ถูกกว่า Failure (rework, incident, lost trust) เสมอ — QA คือการลงทุนฝั่ง prevention

[Teaching Scenario] SHG QA activities ระหว่าง Sprint: test readiness gate ก่อนรับ story เข้า sprint, design review สำหรับ payment flow, data migration checklist ก่อน Sprint 7–8, regression policy ก่อน release

4.4 Develop Team และ Manage Team (จาก lesson-11)

[PMBOK 6]

- **Develop Team:** Tuckman (Forming → Storming → Norming → Performing), สร้าง trust, skill development, ให้อำนาจ (empowerment)
- **Manage Team:** จัดการ conflict ด้วย 5 เทคนิค — Collaborate/Problem Solve, Compromise, Withdraw/Avoid, Smooth/Accommodate, Force/Direct — ต้องเลือกตามสถานการณ์ ไม่ใช่ใช้วิธีเดียวทุกครั้ง
- **Manage Team inputs:** team performance, issue log, work performance data

[Best Practice] PM ไม่ใช่ "คนตามงาน" แต่เป็นผู้ที่ทำให้ทีมมีเป้าหมายร่วม, ขจัด blocker และสร้าง environment ที่ทีมทำงานได้ — ถ้าเห็น Storming (ขัดแย้ง) อย่าหนี ต้องจัดการเป็น process

4.5 Manage Communications / Risk / Stakeholder / Procurement ระหว่าง Execution (D.2)

[Best Practice] ตามแผนจาก Ch.5: Manage Communications (ส่งสารตาม cadence, จับ feedback), Implement Risk Responses (ทำตาม response plan, monitor trigger), Manage Stakeholder Engagement (ติดตาม engagement

จริง vs แผน), Conduct Procurements (vendor ตาม contract, เก็บ evidence)

4.6 กล่อง Agile: Sprint Execution (จาก lesson-15/16)

[PMBOK 8] ถ้าใช้ Scrum (SHG ใช้สำหรับ UX/Booking Journey):

- **Sprint Planning:** เลือก backlog ตาม Sprint Goal — ปกป้อง Sprint Goal จากงานใหม่ที่แทรก
- **Daily Scrum (15 นาที):** เมื่อวานทำอะไร / วันนี้จะทำอะไร / มี blocker อะไร
- **Sprint Review:** โชว์ increment ให้ stakeholder ตรวจ — feedback เปลี่ยน backlog
- **Retrospective:** START / CONTINUE / STOP — ปรับปรุงกระบวนการ
- **Kanban** สำหรับ flow/support/incident: จำกัด WIP, เห็น bottleneck

[Best Practice] Agile ไม่ใช่การไม่มีวินัย — Ceremony ที่ไม่มี feedback ที่เปลี่ยน decision คือ Agile theater

5. PM Decision Thinking

```
Decision: story "Payment Flow" ที่ code เสร็จแต่ test/security ยังไม่ครบ — ควรนับเป็น Done หรือไม่
Owner: Team + QA Lead (DoD check); PM ดูแล process; PO กำหนด business acceptance (Ch.8)
Inputs: DoD, test evidence, review records, security check, deployment status, sprint goal
Options:
  A: นับเป็น Done — เร็ว แต่ส่งของไม่ครบ DoD -> defect ไปตก Ch.8/Ch.9
  B: ไม่นับเป็น Done จนกว่าครบ DoD (แนะนำ) — ช้าชั่วคราว แต่คุณภาพอยู่ในกระบวนการ
  C: ปรับ DoD ให้เบาลง — ต้องผ่านทีม/PO และมีเหตุผล (เช่น ย้าย security check ไปขั้น release)
Trade-offs: velocity ระยะสั้น vs quality/trust ระยะยาว, sprint commitment vs reality
Risk: DoD ที่ยืดหยุ่นเกิน -> "เสร็จ" กลายเป็นคำไร้ความหมาย
Evidence: DoD checklist, sprint review minutes, quality metrics, blocker log
Next Action: นับตาม DoD -> update progress data -> Ch.7 (monitoring) / Ch.8 (verification)
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario]

- **Sprint 1–2 (Core Booking Flow):** ทีมทำ search → view → book บน web — QA วาง test cases คู่ขนานตั้งแต่ design (QA ใน Sprint ไม่ใช่หลัง Sprint)
- **Sprint 3–4 (Payment):** DoR ต้องมี test data (card test จาก 2C2P sandbox) และ design decision — DoD รวม PCI-DSS checklist
- **Sprint 5–6 (Mobile App):** ทีม mobile กับ backend ใช้ API contract review ก่อน integration เพื่อลด defect
- **Sprint 7–8 (Back Office + Inventory Sync):** data migration checklist + PMS adapter test ก่อน merge
- **Retro หลังทุก Sprint:** ปรับ WIP, แก้ bottleneck, ปรับปรุง test data readiness
- **Blockers:** PMS API doc ล่าช้า → escalate ตาม RAID (Risk R-01, owner คุณวีระ) → vendor support เพิ่ม

[PMBOK 8] สังเกตว่าการทดสอบไม่ได้ "รอตอนท้าย" — มันถูกฝังในทุก sprint ผ่าน DoD และ test readiness gate ส่วนการตรวจรับอย่างเป็นทางการยังเป็นหน้าที่ของ Ch.8

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. ไม่มี DoR/DoD — ผล: รับงานไม่พร้อมเข้าสู่ sprint แล้ว "เสร็จ" ไม่ตรงกัน
2. QA มาทีหลัง (test หลัง build เสร็จหมด) — ผล: defect กองและแก้ช้า
3. Sprint Goal ถูกแทรกงานตลอด (marketing ขอ feature กลาง sprint) — ผล: สิ่งสำคัญไม่เสร็จ
4. Blocker ไม่ถูก escalate — ผล: ทีมติดค้างหลายวันโดยไม่มีใครรู้
5. Conflict ถูกหลีกเลี่ยงแทนจัดการ — ผล: พิษสะสม กระทบทีม
6. Manage Quality ถูกเข้าใจว่าเป็น QC — ผล: ไม่มีใครดูกระบวนการ มีแต่ตรวจผลปลายทาง
7. Retro ไม่มี action — ผล: วนทำผิดซ้ำเดิม

Common Misconceptions:

ความเข้าใจผิด	ความจริง
Execution = เขียน code	คือสร้าง deliverables + evidence ผ่านกระบวนการ
QA คือตรวจผลตอนท้าย	QA = ดูกระบวนการ (prevention); QC = ตรวจผล (Ch.8)
"เสร็จ" = code เสร็จ	ต้องครบ DoD (test, review, security, deploy, evidence)
Agile ไม่ต้องมีวินัย	Agile มีวินัยที่สั้น เร็ว และเรียนรู้จาก evidence
PM ต้องสั่งงานทีมทุกอย่าง	PM สร้าง environment + ขจัด blocker; ทีม self-organize (Agile)
Retro เป็นแค่การประชุม	Retro ต้องจบด้วย action ที่มี owner

8. Interview Questions

Foundational

- **Q:** Definition of Done ต่างจาก Definition of Ready อย่างไร?
- **Answer Direction:** DoR = เกณฑ์ก่อนเริ่มงาน (scope, AC, test data, owner); DoD = เกณฑ์ว่าเสร็จจริง (build, review, test, security, deploy, evidence)
- **Warning Signs:** ตอบว่าเหมือนกัน หรือสลับกัน

Scenario

- **Q:** Marketing ขอเพิ่ม feature กลาง Sprint 5 (Mobile App) — คุณจะจัดการอย่างไร?
- **Answer Direction:** ไม่รับเข้าสู่ sprint โดยตรง — ปกป้อง Sprint Goal; เอา feature เข้า backlog, ให้ PO (คุณนภา) จัด priority, ถ้าจำเป็นจริงต้องผ่าน change process (Ch.7) และยอมรับผลต่อ sprint commitment
- **Warning Signs:** รับทุกงานเข้า sprint เพราะ "Agile เปลี่ยนได้" หรือปฏิเสธโดยไม่มีทางเลือก

Senior PM

- **Q:** ทีมมี conflict ระหว่าง dev กับ QA เรื่อง test data — คุณจะจัดการอย่างไร?

- **Answer Direction:** ใช้ conflict resolution ตามสถานการณ์: เริ่มจาก collaborate (หาต้นตอ — test data ไม่พร้อม?), ถ้าจำเป็น compromise/escortate ตาม governance — บันทึกเป็น process improvement ไม่ใช่โทษบุคคล
- **Warning Signs:** เลือกข้าง หรือหลีกเลี่ยงจนปัญหาโต

Executive

- **Q:** ทำไมต้องลงทุนกับ QA ระหว่าง sprint ทั้งที่ "test จริงอยู่ Ch.8"?
- **Answer Direction:** QA (prevention) ถูกกว่า failure cost หลายเท่า — defect ที่พบใน sprint แก้ภายใน 2 ชั่วโมง; ที่พบใน UAT/Production แก้เป็นวัน/สัปดาห์ และกระทบ trust — QA ระหว่างทางลดงานและความเสี่ยงของ Ch.8/Ch.9
- **Warning Signs:** ตอบว่า "QA คือค่าใช้จ่าย" โดยไม่เห็น cost of quality

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Definition of Ready	เกณฑ์พร้อมเริ่ม	scope, AC, test data, owner, estimate ชัด	สับสนกับ DoD	Definition of Done
Definition of Done	เกณฑ์เสร็จจริง	build + review + test + security + deploy + evidence	คิดว่า "code เสร็จ" = Done	Definition of Ready
Manage Quality (QA)	การประกันคุณภาพ	ดูกระบวนการป้องกัน defect	สับสนกับ QC	Control Quality
Control Quality (QC)	การควบคุมคุณภาพ	ตรวจผลลัพธ์จริง (Ch.8)	สับสนกับ QA	Manage Quality
Cost of Quality	ต้นทุนคุณภาพ	Prevention + Appraisal vs Failure	คิดว่า Quality = ค่าใช้จ่ายเพิ่ม	QA, QC
Sprint Goal	เป้าหมาย sprint	focus ของรอบส่งมอบ	ให้งานแทรกตลอด	Sprint
Daily Scrum	ประชุมประจำวัน	เมื่อวาน/วันนี้/blocker (15 นาที)	เป็น status meeting ยาว	Sprint
Retrospective	ทบทวนรอบ	START/CONTINUE/STOP + action	ไม่มี action ออกมา	Sprint Review
WIP Limit	จำกัดงานค้าง	ป้องกันรับงานเกิน capacity	ยิ่งทำหลายงานยิ่งเร็ว	Kanban
Tuckman	ขั้นพัฒนาทีม	Forming→Storming→Norming→Performing	คิดว่าทีมข้ามขั้นได้โดยไม่มี cost	Develop Team
Conflict Resolution	แก้ความขัดแย้ง	collaborate/compromise/withdraw/smooth/force	ใช้วิธีเดียวทุกครั้ง	Manage Team

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Blocker	อุปสรรค	สิ่งที่ขัดงาน — ต้อง escalate	ปล่อยค้างไม่รายงาน	RAID, Escalation

10. Workshop

Scenario

[Teaching Scenario] กลาง Sprint 5 (Mobile App) QA พบว่า: (1) test data ของ payment sandbox ยังไม่พร้อม (2C2P ช้า), (2) Developer หนึ่งคนนับ story เป็น Done โดยไม่มี unit test, (3) Marketing ขอเพิ่ม "flash sale" หน้าจอหลัก — ทีมเริ่มมีเสียงบ่นกัน

Your Role

PM (คุณสุทธิ) ที่ต้องรักษา Sprint Goal + quality process

Available Information

- DoR/DoD ที่ทีมตกลงกัน, Sprint plan, RAID (R-01 PMS, 2C2P dependency), Scenario Master

Missing Information

- 2C2P จะให้ sandbox เมื่อไร, QA เห็น defect trend อย่างไร, Marketing ต้องการ flash sale ในรอบนี้จริงหรือไม่ (impact ต่อ sprint commitment)

Decision Required

- story ที่ไม่มี unit test จะจัดการอย่างไร, จะ escalate 2C2P อย่างไร, flash sale ควรเข้าสู่ sprint หรือ backlog

Constraints

- DoD ต้องไม่ถูกลดโดยไม่มีเหตุผล, Sprint Goal (Mobile App core booking) ต้องไม่หลุด, ต้องไม่ขัดกับ Scenario Master (loyalty/feature ใหม่ = Phase 2+)

Expected Output

- Decision record 3 ข้อ (DoD enforcement, 2C2P escalation, flash sale placement), blocker log update, sprint review agenda ที่รวม quality metrics

Debrief Questions

1. ทำไมการบังคับ DoD ใน sprint ถึงช่วย Ch.8
2. การ escalate 2C2P ที่ดีต้องมีข้อมูลอะไร
3. Retro ที่ดีควรจบด้วยอะไร

Evaluation Criteria

- DoD ไม่ถูกลดเงียบ ๆ, escalation มี evidence + owner, feature ใหม่เข้าผ่าน priority/change process, บันทึก decision ครบ

11. Checklist ใช้งานจริง (Execution / Phase D)

- ☐ DoR ใช้ก่อนรับงานเข้าสู่ sprint/work package
- ☐ DoD รวม test/review/security/deploy/evidence และถูกใช้จริง
- ☐ QA ทำงานคู่ขนาน (test cases ตั้งแต่ design; test readiness gate)
- ☐ Design/data review ก่อน merge (โดยเฉพาะ payment/PMS sync)
- ☐ Daily blocker management — blocker มี owner + escalate path
- ☐ Risk responses ถูก implement ตามแผน (Ch.5) + trigger monitor
- ☐ Stakeholder engagement ตามแผน (Ch.2) — ติดตามจริง vs desired
- ☐ Communication ตาม cadence (Ch.5) — มี feedback loop
- ☐ Team management: conflict จัดการเป็น process, performance feedback
- ☐ Work Performance Data ถูกเก็บ (ส่งให้ Ch.7)
- ☐ Retro (ถ้า Agile) จบด้วย action + owner
- ☐ Cross-ref: QC ผลจริง = Ch.8, monitoring/change = Ch.7

12. Assessment

1. **[Decision Case]** story code เสร็จแต่ไม่มี unit test + security check — จะจัดการอย่างไรโดยไม่ทำลาย velocity ระยะยาว?
2. **[Decision Case]** Marketing ขอ flash sale กลาง sprint — ตัดสินใจอย่างไร (เข้า sprint/backlog/change) พร้อมเหตุผล + กระบวนการ
3. **[Artifact Review]** ตรวจ sprint board ที่งานทุกชิ้น "Done" แต่ไม่มี evidence/test result — อธิบายว่าทำไม board นี้หลอก และแก้อย่างไร
4. **[Trade-off Case]** ระหว่างบังคับ DoD (ช้า 2 วัน) กับปล่อยผ่านแล้วแก้ใน Ch.8 — อธิบาย trade-off ด้าน cost of quality
5. **[Scenario Case]** QA กับ dev ขัดแย้งเรื่อง test data — เลือก conflict resolution technique พร้อมเหตุผลและขั้นตอน
6. **[Cross-Knowledge Analysis]** ถ้า 2C2P sandbox ช้า 1 สัปดาห์ — กระทบ sprint 3–4 (Ch.6), monitoring (Ch.7) และ UAT (Ch.8) อย่างไร
7. **[Recall]** ความต่างระหว่าง Manage Quality (QA) กับ Control Quality (QC) พร้อมตัวอย่างจาก SHG

13. Executive Summary

มิติ	สรุป
Why	Execution คือจุดที่แผนกลายเป็น deliverables — วินัยตรงนี้กำหนดคุณภาพของ Ch.8/Ch.9
Decision Improved	"เสร็จ" ถูกตัดสินด้วย DoD + evidence ไม่ใช่ความรู้สึก
Failure Prevented	ป้องกัน defect สะสม, sprint goal หลุด, conflict เน่า และ QA ที่มาทีหลัง
What to Monitor	DoD compliance, defect trend, blocker ageing, sprint velocity vs commitment

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: Execution กำลังสร้าง deliverables พร้อม Work Performance Data — Ch.7 (Monitoring & Change) จะวัดผลจริงเทียบ baseline จัดการ variance และทำ Perform Integrated Change Control (Integration ท่อนที่ 2)

Field	Handoff
Input	Integrated Plan, Baselines (Ch.4/Ch.5), Sprint/Work plan
Output	Deliverables, Work Performance Data, Quality/QA Evidence, Team/Blocker Records
Creator	Delivery Team + Leads; QA Lead (QA evidence); PM (team/process)
Artifact owner	Workstream Leads / QA Lead / PM
Reviewer	PM, QA, PO (business)
Approval authority	ตาม Governance; acceptance อย่างเป็นทางการ = Ch.8
Minimum acceptance	Usable (evidence ต่อ deliverable + DoD check)
Next chapter use	Ch.7 ใช้ Work Performance Data + baselines ไปวัด variance, forecast และ change control

15. Quick Reference Card

```
EXECUTION (D) — เป้าหมาย: สร้าง deliverables + evidence ด้วยกระบวนการที่มีวินัย
---------------------------------------------------------------
1. DoR ก่อนเริ่ม   -> scope, AC, test data, owner, estimate
2. Build/Integrate -> ตาม sprint/work package
3. Peer Review     -> ก่อน QA
4. Test ระหว่างทาง -> QA คู่ขนาน (ไม่ใช่รอท้าย)
5. DoD             -> build + review + test + security + deploy + evidence
6. Manage Quality  -> QA ดูกระบวนการ (prevention) | QC ตรวจผล = Ch.8
7. Team            -> Tuckman + conflict resolution + blocker escalation
8. Comms/Risk/Stakeholder -> ตามแผน Ch.5
9. Work Performance Data -> ส่ง Ch.7

Agile insert:
  Sprint Planning -> Daily Scrum -> Review -> Retro (START/CONTINUE/STOP)
  ปกป้อง Sprint Goal | Kanban: WIP limit + bottleneck

กฎ 3 ข้อห้าม:
- ห้าม "เสร็จ" โดยไม่ครบ DoD
- ห้าม QA มาทีหลัง
- ห้าม Retro ไม่มี action
```

Chapter 07 — Monitoring, Controlling และ Change: วัดจริง ควบคุม และอนุมัติการเปลี่ยนแปลง

1. Opening Scenario Hook

[Teaching Scenario] เดือนที่ 5 ของ SHG: Sprint 9–10 กำลังจะจบ แต่รายงานพบว่า CPI = 0.90 และ Sprint 7–8 (Back Office + Inventory Sync) ล่าช้า 1 สัปดาห์ พร้อมกันนี้คุณภัทร (VP Marketing) ขอเพิ่ม "flash sale campaign" และคุณวีระ (CTO) แจ้งว่า PMS vendor ขอค่าใช้จ่ายเพิ่มสำหรับ adapter พิเศษ

[PMBOK 8] ผู้บริหารหลายคนจะถาม "แล้วจะทัน launch ไหม" ทีมอาจตอบ "เดี๋ยวเร่งให้" แต่ PM ต้องทำมากกว่านั้น: วัด variance เทียบ baseline, forecast ผลลัพธ์, หาสาเหตุ และพาทุกคำขอเปลี่ยนแปลงเข้าสู่ **Perform Integrated Change Control** — เพราะ Integration ทำงานตลอดโครงการ (ท่อนที่ 2 ของ 3 ท่อน: Charter = Ch.2, Change = บทนี้, Closure = Ch.10)

2. Why It Matters

[PMBOK 6] Monitoring & Controlling ไม่ได้เริ่มหลัง Execution เสร็จ — เกิดควบคู่กัน: Collect Actuals → Compare with Baseline → Analyze Variance → Forecast → Identify Cause → Develop Options → Decide → Implement → Verify

[Best Practice] ถ้าไม่มีระบบนี้: สีเขียวบนรายงานคือความเห็น, change ถูกอนุมัติด้วยวาจา, baseline ค่อย ๆ เลื่อนโดยไม่มีใครสังเกต และ sponsor ตัดสินใจจากข้อมูลที่คลาดเคลื่อน — ระบบ monitoring + change control คือสิ่งที่ทำให้ "การบริหาร" ต่างจาก "การตามงาน"

3. Mental Model

```
Work Performance Data (Ch.6)
  -> Collect Actuals (daily/weekly)
  -> Compare with Baseline (schedule/cost/scope)
  -> Variance + Forecast (SPI/CPI/EAC, critical path)
  -> Identify Cause (ทำไม — ไม่ใช่แค่เท่าไร)
  -> Develop Options + Decide (มี governance)
  -> Implement + Verify
ควบคู่กัน:
  Change Intake -> Impact Analysis -> CCB Decision -> Baseline Update
  RAID Review (ต่อเนื่อง) + Escalation (เมื่อเกินอำนาจ)
  -> สถานะพร้อมส่ง Ch.8 (Verification/UAT)
```

4. Main Lesson

4.1 Monitoring Cycle (E.3) + Status Reporting (E.4)

[Best Practice] รายงานต้องตอบ: ตอนนี้อยู่ที่ไหน / เทียบแผนเป็นอย่างไร / อะไรเปลี่ยน / ทำไม / กระทบอะไร / ต้องตัดสินใจอะไร / ใครต้องทำอะไร / ภายในเมื่อไร

[Teaching Scenario] Status รายสัปดาห์ของ SHG: CPI/SPI, critical path health, defect trend, RAID update, decision ask (พร้อม deadline)

4.2 Schedule Control (E.6)

[PMBOK 6] ดู: Actual Start/Finish, Remaining Duration, Critical Path, Float, Milestone Variance, Forecast Finish, Recovery Plan — delay ต้องถูกเทียบกับ critical path (บทเรียน Ch.4) ก่อนสรุป

4.3 Cost Control และ EVM (E.7) — จาก lesson-09

[PMBOK 6] ใช้เมื่อข้อมูลและ governance เหมาะสม:

```
SV = EV - PV | CV = EV - AC | SPI = EV/PV | CPI = EV/AC
EAC = BAC/CPI (trend เดิม) | VAC = BAC - EAC
```

[Best Practice] ตัวเลข < 1 เป็นประตูคำถาม ไม่ใช่คำตอบ — ถาม: variance มาจาก cost/schedule/completion? EV มี evidence จริง? one-time หรือ trend? reserve ใดใช้ได้ ใครอนุมัติ?

[Teaching Scenario] SHG ณ เดือนที่ 5: ถ้า PV = 7.5M, EV = 6.75M, AC = 7.5M → SPI = 0.90, CPI = 0.90, EAC ≈ 12/0.90 = 13.3M, VAC ≈ -1.3M → ยังพอรับได้ใน Contingency 1.5M แต่ต้องถามสาเหตุและตั้ง forecast ใหม่ — ห้ามเอา Management Reserve 1.5M มาใช้โดยไม่ผ่าน Sponsor

4.4 Validate Scope vs Control Scope (E.8)

[PMBOK 6]

- **Validate Scope:** ลูกค้า/Sponsor ตรวจรับ deliverable (acceptance evidence) — กระบวนการตรวจรับจริงเต็มรูปแบบอยู่ Ch.8
- **Control Scope:** ควบคุมการเปลี่ยน Scope Baseline — ทุก change ผ่านกระบวนการ

4.5 Perform Integrated Change Control (E.5) — หัวใจของบท

[PMBOK 6] Change Flow:

```
Submit Request -> Log -> Check Baseline -> Clarify -> Impact Analysis
-> Develop Options -> Recommend -> Change Authority Decision
-> Update Plan/Baseline -> Communicate -> Implement -> Verify -> Close
```

Impact Areas ต้องดูครบ: Scope, Schedule, Cost, Quality, Resource, Risk, Contract, Operation, Benefits

Change Authority: PM อนุมัติ minor ตาม threshold; PO จัด priority; Sponsor/CCB อนุมัติ baseline change; Commercial/Legal อนุมัติ contract change

[Best Practice] Integrated Change Control ไม่ได้มีไว้เพื่อปฏิเสธ change — มีไว้เพื่อให้ change ที่ควรทำถูกอนุมัติด้วยความเข้าใจผลกระทบ และ change ที่ยังไม่ควรทำถูกเลื่อน/แบ่งเฟสอย่างมีเหตุผล — แยก Issue (เกิดแล้ว) ออกจาก Change Request (ขอปรับ baseline)

4.6 RAID Review และ Escalation (E.9–E.10)

[Best Practice] Risk/Assumption/Issue/Dependency ต้อง review ต่อเนื่อง — Escalate เมื่อ: เกินอำนาจ PM, กระทบ baseline, ต้องการ Sponsor decision, cross-project conflict, contract risk, security/legal, critical milestone, benefit at risk

5. PM Decision Thinking

```
Decision: flash sale campaign ของคุณภัทร + ค่าใช้จ่าย PMS adapter เพิ่ม —
approve/reject/defer/split อย่างไร
Owner: PM ทำ impact analysis; CCB/Sponsor ตัดสินตาม governance; Commercial/Legal สำหรับ
contract impact
Inputs: charter, baselines, status reports, change request, risk register, quality
evidence, contract
Options:
  A: Approve ทั้งคู่ — ต้องมีงบ/เวลา/scope ใหม่ + baseline update
  B: Approve แบบมีเงื่อนไข (flash sale defer ไปหลัง stabilization; adapter ใช้ contingency ตาม
governance)
  C: Reject/split — flash sale ไป Phase 2 (Scenario Master: feature ใหม่ = Phase 2+),
adapter เจรจา contract
Trade-offs: business value vs delivery risk, speed vs stability, local request vs
enterprise outcome
Risk: อนุมัติโดยไม่เห็นผลกระทบ -> baseline ใช้ควบคุมไม่ได้
Evidence: integrated impact pack, decision record, updated baselines, communication
Next Action: ตัดสินใจ -> update plan/baseline/log -> สื่อสาร -> verify หลัง implement
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario]

- **สถานะเดือนที่ 5:** SPI/CPI = 0.90, Sprint 7–8 ล่าช้า 1 สัปดาห์ (อยู่บน critical path — กระทบ launch) → เสนอ recovery: fast-track บางส่วนของ Back Office + เพิ่ม QA คู่ขนาน; monitor ใหม่ทุกสัปดาห์
- **Change 1 — Flash Sale:** คุณภัทรต้องการ campaign ก่อน launch — Impact: scope + sprint 9–10 + payment load test → เสนอ defer ไป Phase 2 (ตาม Scenario Master §4) หรือทำเป็น pilot เล็กหลัง stabilization → CCB อนุมัติ defer
- **Change 2 — PMS Adapter เพิ่ม:** vendor ขอ T&M เพิ่ม 300K เกิน cap — Impact: cost baseline + contingency → เสนอใช้ Contingency 1.5M ตาม governance + ขอ vendor แบ่ง milestone → Sponsor อนุมัติแบบมีเงื่อนไข
- **RAID:** R-01 (PMS API) trigger ใกล้เกิดขึ้น → เพิ่ม daily data review + vendor SLA enforcement; 2C2P dependency update

[PMBOK 8] สังเกต: ทุกการตัดสินใจมี decision record, baseline update และ communication — ไม่มีอะไรผ่าน "ด้วยวาจา"

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. ปล่อย change ผ่านคำพูด — ผล: ไม่มีใครจำได้ว่าตกลงอะไร baseline ไหนเปลี่ยน
2. Issue กับ Change Request ปนกัน — ผล: แก้ issue โดยไม่รู้ว่า baseline เปลี่ยน
3. ดูแค่ % complete โดยไม่ดู EV/evidence — ผล: รายงานเขียวแต่ของจริงไม่เสร็จ
4. ใช้ Funding Envelope (12M) เป็นข้ออ้างเลี่ยง governance — ผล: reserve หมดโดยไม่มีเหตุผล
5. รอ "ถึงเวลาจริง" ค่อย monitor — ผล: ปัญหาสะสมจน recovery แพง
6. Escalate ทุกอย่าง (หรือไม่ escalate เลย) — ผล: เสียงดังกลบ decision หรือ decision ช้าเกินไป
7. Baseline เปลี่ยนแต่ไม่ update ทุกฝ่าย — ผล: ทีมวางแผนจากแผนเก่า

Common Misconceptions:

ความเข้าใจผิด	ความจริง
Monitoring คือการทำรายงาน	คือการวัด + วิเคราะห์ + ตัดสินใจ + verify
Change Control ทำให้ช้าเสมอ	ลด rework และทำให้ trade-off โปร่งใส
PM อนุมัติทุก change	Authority ตาม Governance (minor = PM, baseline = CCB/Sponsor)
CPI/SPI ตอบทุกอย่าง	ต้องดู cause, forecast, quality, risk
ใช้เงินน้อย = ดี	ต้องเทียบกับ EV และ schedule
Escalation = แพ้	Escalate ที่ถูกเวลา = ใช้ governance ให้เกิดประโยชน์

8. Interview Questions

Foundational

- **Q:** Issue ต่างจาก Change Request อย่างไร?
- **Answer Direction:** Issue = ปัญหาที่เกิดแล้ว (ต้องแก้/escalate); Change Request = คำขอปรับ baseline อย่างเป็นทางการ (ต้อง impact analysis + approval)
- **Warning Signs:** ใช้สลับกัน หรือเรียกทุกอย่างว่า change

Scenario

- **Q:** CPI = 0.90 และ Sprint 7–8 ล่าช้าบน critical path — คุณจะรายงานและตัดสินใจอย่างไร?
- **Answer Direction:** แยก variance ออกจาก cause; คำนวณ EAC/VAC; ดู critical path/float; เสนอ recovery options พร้อม trade-off; ระบุ authority ถ้าต้อง rebaseline; สื่อสารเป็น decision brief (ไม่ใช่แค่ตัวเลข)
- **Warning Signs:** ตอบ "เร่งทีม" หรือรายงานแค่ "CPI ต่ำ"

Senior PM

- **Q:** Sponsor อนุมัติ change ด้วยวาจาในที่ประชุม แล้วบอกว่า "จดไว้ก็แล้วกัน" — คุณจะทำอย่างไร?

- **Answer Direction:** ยืนยันผ่านระบบ: บันทึกเป็น change request + impact pack, ขอ decision record/email confirm ก่อน update baseline — อธิบายว่า baseline ที่ไม่มี evidence ใช้ defend ไม่ได้ตอนปิดโครงการ
- **Warning Signs:** เงียบแล้วทำตาม หรือทะเลาะโดยไม่มีทางเลือกที่สะดวกทั้งสองฝ่าย

Executive

- **Q:** ทำไม change ต้องผ่าน CCB ทั้งที่ Sponsor เห็นด้วยแล้ว?
- **Answer Direction:** CCB ทำให้เห็นผลกระทบครบทุกด้าน (scope/schedule/cost/quality/risk/contract) ก่อนตัดสินใจ และสร้าง audit trail — ลดความเสี่ยงที่ decision รีบร้อนจะพังทีหลัง; decision ใหญ่ยังจบที่ Sponsor/CCB เหมือนเดิม แค่มีข้อมูลครบ
- **Warning Signs:** ตอบว่า "เป็นขั้นตอนราชการ"

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Monitoring	การติดตาม	วัดจริงเทียบ baseline ต่อเนื่อง	คิดว่า = รายงานตามรอบ	Controlling
Controlling	การควบคุม	วิเคราะห์ variance + แก้ + verify	สับสนกับ Monitoring	Monitoring
Variance	ส่วนเบี่ยงเบน	SV/CV — ต่างจากแผนเท่าไร	ดูแค่ตัวเลขไม่ดู cause	Forecast
Forecast	การคาดการณ์	EAC/VAC — จบงานจะจบแบบไหน	คิดว่าเท่ากับ BAC เสมอ	EAC, VAC
Change Request	คำขอเปลี่ยนแปลง	ขอปรับ baseline อย่างเป็นทางการ	สับสนกับ Issue	Issue, Impact Analysis
Impact Analysis	วิเคราะห์ผลกระทบ	ดูครบ scope/schedule/cost/quality/risk/contract	ทำผิวเผินเป็น opinion	CCB
CCB	Change Control Board	อนุมัติ change ที่กระทบ baseline	คิดว่า PM อนุมัติเองได้ทุกเรื่อง	Governance
Baseline Update	ปรับฐานควบคุม	update หลังอนุมัติ change	สับสนกับ Status Update	Change Request
Issue	ปัญหาที่เกิดแล้ว	ต้องมี owner, action, escalation	สับสนกับ Risk/Change	Risk, Change Request
Escalation	การยกระดับ	ส่งเรื่องให้ authority ตัดสินใจ	ไม่ escalate / escalate ทุกอย่าง	Governance
Decision Brief	บันทึกขอตัดสินใจ	decision ask + evidence + deadline	สับสนกับ Status Report	Status Report
Validate Scope	ตรวจรับขอบเขต	ยืนยัน deliverable กับผู้มี authority (Ch.8)	สับสนกับ Control Scope	Control Scope

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Control Scope	ควบคุมขอบเขต	จัดการ change ต่อ scope baseline	สับสนกับ Validate Scope	Change Request

10. Workshop

Scenario

[Teaching Scenario] เดือนที่ 5 ของ SHG: CPI/SPI = 0.90, Sprint 7–8 ล่าช้า 1 สัปดาห์ (critical path), คุณภัทรขอ flash sale ก่อน launch, PMS vendor ขอ T&M เพิ่ม 300K — คุณต้องจัดทำ decision pack สำหรับ CCB รอบถัดไป

Your Role

PM (คุณสุทธิ) ที่ต้องสร้าง integrated impact pack + recommendation

Available Information

- Baselines (12M, Sprint 0–12), EVM ตัวเลข, Risk Register (R-01 PMS), Scenario Master (feature ใหม่ = Phase 2+)

Missing Information

- รายละเอียด flash sale (scope/ระยะเวลา/impact ต่อ payment load), เหตุผล vendor ขอเพิ่ม (work เกิน SOW จริงหรือ?), ความเห็น QA ต่อ fast-track Back Office, การอนุมัติ contingency ของ Sponsor

Decision Required

- flash sale: approve/defer/reject? PMS adapter: approve ใช้ contingency / เจรจา / ปฏิเสธ? recovery สำหรับ critical path delay: option ใด?

Constraints

- Launch ก่อน 1 พ.ย., งบ 12M + reserve ต้องใช้ตาม governance, feature ใหม่ตาม Scenario Master ต้องไป Phase 2+, ทุก decision ต้องมี decision record + baseline update

Expected Output

- Impact Pack (impact analysis ครบทุกด้าน ต่อ 2 change), CCB recommendation (options + trade-off + authority), recovery brief สำหรับ schedule delay, updated RAID

Debrief Questions

1. ทำไม impact analysis ต้องดู "benefits" ด้วย ไม่ใช่แค่ scope/cost
2. การใช้ Contingency Reserve ต่างจาก Management Reserve อย่างไร
3. ถ้า CCB อนุมัติต่างจากคำแนะนำ คุณจะทำอะไร

Evaluation Criteria

- Impact ครบทุกด้าน, options + trade-off ชัด, authority/evidence ถูกต้อง, ไม่ขัด Scenario Master, baseline update ระบุครบ

11. Checklist ใช้งานจริง (Monitoring & Change / Phase E)

- ☐ Actuals ถูกเก็บตาม cadence (daily/weekly) พร้อม evidence
- ☐ Variance + forecast คำนวณ (SV/CV/SPI/CPI/EAC/VAC เมื่อเหมาะสม)
- ☐ Critical path health ถูกติดตามทุกสัปดาห์
- ☐ Status report ตอบ 8 คำถาม (อยู่ไหน/เทียบแผน/เปลี่ยนอะไร/ทำไม/กระทบ/decision/ใคร/เมื่อไร)
- ☐ Change intake: ทุกคำขอถูกล็อก (Change Log)
- ☐ Impact analysis ครบทุกด้าน (scope/schedule/cost/quality/resource/risk/contract/operation/benefits)
- ☐ Decision ผ่าน authority ที่ถูกต้อง (PM minor / CCB baseline / Sponsor reserve)
- ☐ Baseline ถูก update หลังอนุมัติ + สื่อสารทุกฝ่าย
- ☐ RAID review ต่อเนื่อง — risk trigger ถูก monitor
- ☐ Escalation ใช้เมื่อเข้าเงื่อนไข (E.10)
- ☐ Decision record + communication ครบ
- ☐ ส่งต่อ Ch.8: deliverables + evidence พร้อมตรวจรับ

12. Assessment

1. **[EVM Case]** เดือนที่ 5: PV = 7.5M, EV = 6.75M, AC = 7.5M — คำนวณ SPI/CPI/EAC/VAC และอธิบายว่า reserve ตัวไหนใช้ได้ ใครอนุมัติ
2. **[Decision Case]** flash sale ก่อน launch — ทำ impact analysis (ครบทุกด้าน) + แนะนำ approve/defer/reject พร้อม authority
3. **[Decision Case]** PMS vendor ขอ T&M เพิ่ม 300K เกิน cap — เสนอทางเลือก (ใช้ contingency/เจรจา/ปฏิเสธ) พร้อม evidence และ decision path
4. **[Artifact Review]** ตรวจ Change Log ที่มี "เพิ่มฟีเจอร์" โดยไม่มี impact analysis/baseline update/approval — ระบุสิ่งที่ขาดและผลที่จะเกิด
5. **[Scenario Case]** Sponsor อนุมัติ change ด้วยวาจา — อธิบายขั้นตอนที่คุณจะทำให้เป็นทางการ
6. **[Cross-Knowledge Analysis]** ถ้า fast-track Back Office ทำให้ defect เพิ่ม — กระทบ Ch.8 (UAT) และ Ch.9 (Go-live) อย่างไร และคุณจะ monitor อะไร
7. **[Recall]** เงื่อนไขที่ควร escalate (จาก E.10) มีอะไรบ้าง

13. Executive Summary

มิติ	สรุป
Why	Monitoring + Change Control คือระบบที่ทำให้การบริหารต่างจากการตามงาน
Decision Improved	ทุก change ผ่าน impact analysis + authority + evidence
Failure Prevented	ป้องกัน baseline เลื่อนเงียบ ๆ, change ด้วยวาจา, และรายงานที่หลอก
What to Monitor	SPI/CPI trend, critical path, reserve usage, change queue, escalation ageing

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: งานหลักเสร็จ + ถูกควบคุมแล้ว — Ch.8 (Verification/UAT) จะพิสูจน์ว่า solution ตรง requirement และพร้อมขึ้น Production (Quality ฝั่ง QC — cross-ref กับ Test Strategy ที่วางใน Ch.5)

Field	Handoff
Input	Work Performance Data (Ch.6), Baselines (Ch.4/Ch.5), Change Log
Output	Status Reports, Variance/Forecast, Approved Change Log + Updated Baselines, Updated RAID
Creator	PM + Impact Owners
Artifact owner	PM
Reviewer	CCB/Sponsor (decision), Stakeholders (report)
Approval authority	PM (minor) / CCB-Sponsor (baseline) / Commercial-Legal (contract)
Minimum acceptance	Usable (decision record + baseline update ครบ)
Next chapter use	Ch.8 ใช้ deliverables + evidence + acceptance criteria เพื่อ Verification/UAT/Release Readiness

15. Quick Reference Card

```
MONITORING & CHANGE (E)
------------------------------------------------------------
Cycle: Collect -> Compare baseline -> Variance/Forecast -> Cause
       -> Options -> Decide -> Implement -> Verify
EVM:   SV/SPI/CV/CPI -> EAC = BAC/CPI -> VAC = BAC - EAC
Change: Intake -> Log -> Impact (ทุกด้าน) -> Options -> CCB/Authority
        -> Baseline Update -> Communicate -> Implement -> Verify
RAID:  Review ต่อเนื่อง + Escalate เมื่อเข้าเงื่อนไข (E.10)

กฎ 3 ข้อห้าม:
- ห้าม change ผ่านวาจา
- ห้ามแยก Issue กับ Change Request ปนกัน
- ห้ามรายงาน % โดยไม่มี EV/evidence

Integration note: Charter = Ch.2 | Change Control = บทนี้ | Closure = Ch.10
```

Chapter 08 — Verification, UAT และ Release Readiness: พิสูจน์ว่า "พร้อมขึ้น Production"

1. Opening Scenario Hook

[Teaching Scenario] Sprint 11 ของ SHG เริ่มแล้ว UAT dashboard เป็นสีเขียว แต่เมื่อคุณสุทธิ (PM) ถามลึกลงไป กลับพบว่า: ไม่มี requirement coverage, ไม่มี defect severity, ไม่มี retest result, key users จาก 3 โรงแรมยังไม่ได้ทดสอบ payment scenario จริง และ UAT sign-off ยังไม่เซ็น

[PMBOK 8] สีเขียวบน dashboard ไม่ใช่หลักฐาน — หลักฐานคือ connection ระหว่าง requirement → test case → test result → acceptance (RTM) ที่มี evidence ครบ — บทนี้คือจุดที่ **Test Strategy จาก Ch.5** กลายเป็น **ผลการทดสอบจริง** และเป็นจุดตัดสินใจ Go/No-Go ก่อนส่งต่อ Ch.9

2. Why It Matters

[PMBOK 6] Verification (สร้างถูกตาม spec) + Validation (สร้างสิ่งที่ตอบ need) + Release Readiness คือการพิสูจน์ว่า solution ทำงานตาม requirement, มีคุณภาพ, ปลอดภัย, ลูกค้ายอมรับ และ operations พร้อมรับช่วง — ถ้าข้ามหรือทำเป็นพิธี ต้นทุนจะย้ายไป Ch.9 (Go-live ผิดพลาด) และ Ch.10 (closure dispute)

[Best Practice] UAT ไม่ควรใช้แทน System Test — QA/SIT พิสูจน์ technical/functional correctness ส่วน UAT พิสูจน์ว่า business user ยอมรับว่า solution รองรับ business process จริง

3. Mental Model

```
Test Strategy (Ch.5) + Deliverables (Ch.6) + Baselines
  -> Test Planning (env, data, entry/exit)
  -> SIT / System Test (technical correctness)
  -> UAT Prepare (pack: scope, participants, scenarios)
  -> UAT Execute (business users + evidence)
  -> Defect Triage (severity ≠ priority; fix/defer/waive)
  -> RTM check (requirement -> test -> result -> acceptance)
  -> Release Readiness Review (F.7 checklist)
  -> Go / No-Go Decision (Sponsor/Business Owner)
  -> Ch.9 (Go-Live & Hypercare)
```

> **Cross-reference (Quality):** แผน Test/Test Strategy = Ch.5 | QA ฝั่งกระบวนการ = Ch.6 | **ผลการทดสอบจริง/QC/UAT = บทนี้**

4. Main Lesson

4.1 Test Levels (F.2)

[Best Practice] Unit → Component → Integration → System Test → SIT → Performance → Security → Regression → UAT → Production Verification — แต่ละ level มี purpose + owner + exit criteria

4.2 QA vs UAT (F.3)

[PMBOK 6] QA/SIT พิสูจน์ technical correctness; UAT พิสูจน์ business acceptance — UAT ไม่ใช้แทน system test; ถ้า SIT ยังไม่ผ่าน UAT จะกลายเป็นการทดสอบซ้ำที่แพงและไร้ความหมาย

4.3 UAT Planning (F.4)

[Best Practice] ต้องมี: UAT Scope, Participants, Environment, Data, Scenarios, Expected Result, Entry Criteria, Exit Criteria, Defect Handling, Sign-off Authority, Schedule, Evidence

[Teaching Scenario] SHG UAT: scope = booking flow + payment + back office + PMS sync, participants = key users 80 คน (จาก 12 โรงแรม), environment = UAT env กับ test data (ต้องไม่ใช้ production data), scenarios = ตาม acceptance criteria จาก Ch.3, sign-off = คุณนภา (PO) + คุณภัทร (VP Marketing) ตาม Governance

4.4 Requirements Traceability (F.5)

[Best Practice] RTM เชื่อม: Business Need → Requirement → Design → Build → Test Case → Test Result → Acceptance — ถ้า RTM ไม่ครบ แปลว่ามี requirement ที่ไม่ถูกทดสอบ (coverage gap)

4.5 Defect Management (F.6)

[Best Practice] Defect ต้องมี: ID, Description, Environment, Steps, Expected, Actual, Severity, Priority, Owner, Fix Version, Retest, Status

[PMBOK 6] Severity ≠ Priority: Tester/QA กำหนด Severity (ผลกระทบ: Critical/Major/Minor/Low); PO กำหนด Priority (ความเร่งด่วน: Immediate/High/Medium/Low) — defect 20 รายการที่เป็น cosmetic อาจน่ากังวลน้อยกว่า defect เดียวที่ทำให้ payment fail

4.6 Release Readiness (F.7) + Go/No-Go (F.8)

[Best Practice] ตรวจ: Scope Complete, Tests Pass, Critical Defects Closed, Security Accepted, Performance Accepted, Data Migration Tested, Monitoring Ready, Backup Ready, Rollback Ready, Training Complete, Support Ready, Communication Ready, Approval Ready

Go/No-Go inputs: Business Readiness, Technical Readiness, Operational Readiness, Security, Data, Defects, Rollback, Support, Risk, Stakeholder Approval

[Best Practice] Go/No-Go ไม่ใช่การเชียร์ให้ผ่าน — ถ้า defect critical ค้าง หรือ rollback ยังไม่ทดสอบ คำตอบที่ถูกอาจเป็น No-Go หรือ Go แบบมีเงื่อนไข (conditional) — Sponsor/Business Owner เป็น authority

5. PM Decision Thinking

```
Decision: UAT พบ payment defect 8% failure rate + key users 3 โรงแรมยังไม่ทดสอบ — Go, No-Go
หรือ Conditional?
Owner: PM + QA Lead วิเคราะห์; PO/คุณภัทร (business acceptance); Sponsor (go-live authority)
Inputs: RTM coverage, test results, severity, retest, regression, data readiness, rollback,
support, Scenario Master
Options:
  A: Go — ทุก exit criteria ผ่าน
  B: Conditional Go — เปิดใช้งานบางส่วน (pilot 3 โรงแรม) + ปิด defect ตาม due date (ตรงกับ Sprint
12 soft launch)
  C: No-Go — defect critical ค้าง / rollback ยังไม่พร้อม -> เลื่อน launch (กระทบ high season)
Trade-offs: launch timing vs failure cost, momentum vs trust, pilot vs full launch
Risk: Go ทั้งที่ payment ไม่เสถียร -> campaign กลาง launch ล่ม -> revenue + trust เสียหาย
Evidence: release readiness report, defect triage, retest results, rollback test, support
plan, signed acceptance
Next Action: ตัดสินใจ -> ดำเนินการ (launch/cutover = Ch.9) หรือปิด gap -> review ใหม่
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario]

- **SIT:** booking flow + PMS sync ผ่าน แต่ payment sandbox พบ defect 2 รายการ (callback timeout, duplicate confirmation)
- **UAT:** key users ทดสอบตาม scenarios — พบ payment failure 8% ใน beta (Risk: Low Conversion — Scenario Master §10) → triage: severity Critical (payment) / priority Immediate (PO)
- **Defect ตัดสินใจ:** fix critical ก่อน, defer cosmetic, regression retest หลัง fix
- **RTM:** พบ coverage gap — 3 โรงแรมยังไม่ได้ทดสอบ cancellation scenario → เพิ่ม wave
- **Release Readiness:** security (PCI-DSS) ผ่าน, performance ผ่าน (3s/5s/99.5%), rollback plan ทดสอบแล้ว, support team พร้อม (Ch.9 hypercare)
- **Go/No-Go:** เสนอ **Conditional Go — Soft Launch 3 โรงแรม** (ตาม Sprint 12 ของ Scenario Master) + ปิด defect critical ภายใน 1 สัปดาห์ + monitor payment failure rate < threshold — Sponsor (คุณจิรา) อนุมัติ

[PMBOK 8] สังเกตว่า decision มี evidence และเป็น conditional ที่มีเงื่อนไขชัด ไม่ใช่ "ผ่านไปก่อน"

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. ใช้ UAT แทน System Test — ผล: ทดสอบซ้ำแพง หรือ business เจอ bug ที่ QA ควรเจอ
2. นับ defect ทั้งหมดเป็น "เยอะ" โดยไม่ดู severity — ผล: ตัดสินใจผิด (cosmetic 20 ตัว vs critical 1 ตัว)
3. UAT sign-off เป็นพิธี — ผล: ผู้ใช้ไม่ได้ทดสอบจริง แต่เซ็นเพราะเกรงใจ
4. ไม่มี RTM — ผล: ไม่รู้ว่า requirement ไหนยังไม่ถูกทดสอบ
5. Go/No-Go ตัดสินจากแรงกดดัน campaign — ผล: ขึ้น production ทั้งที่ rollback ยังไม่พร้อม

6. Test data ไม่ตรงของจริง — ผล: ผล test ไม่สะท้อน production
7. ไม่มี release readiness checklist — ผล: ลืม training/support/communication ตอนท้าย

Common Misconceptions:

ความเข้าใจผิด	ความจริง
จำนวน defect น้อย = พร้อม	ต้องดู severity, coverage, retest, data readiness
สีเขียวบน dashboard = ผ่าน	ต้องมี evidence เชื่อม requirement → test → acceptance
UAT = การทดสอบรอบสุดท้ายทั่วไป	UAT = business acceptance โดยเฉพาะ ไม่แทน system test
Severity = Priority	Tester ตั้ง severity; PO ตั้ง priority
Go/No-Go ตัดสินโดย PM	Authority = Sponsor/Business Owner
Sign-off = จบ	ต้องมี acceptance evidence + residual risk decision

8. Interview Questions

Foundational

- **Q:** Verification ต่างจาก Validation อย่างไร?
- **Answer Direction:** Verification = สร้างถูกตาม spec (build the product right); Validation = สร้างสิ่งที่ตอบ need (build the right product) — SHG: verify ว่า booking flow ทำงานตาม SRS; validate ว่าผู้ใช้จองสำเร็จ และ conversion ดี
- **Warning Signs:** สลับกันหรือตอบว่าเหมือนกัน

Scenario

- **Q:** UAT พบ payment failure 8% ก่อน launch campaign — คุณจะทำอย่างไร?
- **Answer Direction:** ไม่สรุปทันที — triage severity (critical?), หาสาเหตุ, fix + regression retest, ประเมิน rollback/support readiness, นำเสนอ Go/No-Go แบบมี evidence ต่อ Sponsor (อาจ conditional: pilot ก่อน)
- **Warning Signs:** ตอบ "เลื่อน launch" ทันที หรือ "go เลย campaign รอไม่ได้" โดยไม่มี evidence

Senior PM

- **Q:** PO อยากเซ็น UAT sign-off ทั้งที่ key users ยังทดสอบไม่ครบทุก scenario — คุณจะจัดการอย่างไร?
- **Answer Direction:** อธิบายความเสี่ยงของการ accept โดยไม่ครบ coverage; เสนอทางเลือก: ขยาย UAT wave, ลด scope ของ sign-off (waiver ระบุ scenario ที่เลื่อน), หรือ conditional sign-off พร้อม due date — บันทึก residual risk อย่างเป็นทางการ
- **Warning Signs:** ปล่อยให้เซ็นเพื่อความสะดวก หรือยึดตายตัวโดยไม่มีทางเลือก

Executive

- **Q:** ทำไมต้องมี Release Readiness Report ก่อน Go/No-Go?

- **Answer Direction:** รายงานรวม evidence จากทุกมุม (scope, test, security, performance, data, rollback, training, support, approval) ให้ decision maker เห็นภาพเดียว — ลดความเสี่ยงที่ Sponsor ตัดสินใจจากข้อมูลไม่ครบ และสร้าง audit trail สำหรับ go-live
- **Warning Signs:** ตอบว่า "เป็นกระบวนการ QA"

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Verification	ตรวจตาม spec	build the product right	สับสนกับ Validation	Validation
Validation	ตรวจว่าตอบ need	build the right product	สับสนกับ Verification	Verification
SIT	System Integration Test	ทดสอบรวมระบบ/การเชื่อมต่อ	ใช้แทน UAT	UAT
UAT	User Acceptance Test	ผู้ใช้ธุรกิจตรวจรับ business process	ใช้แทน system test	SIT, Acceptance
RTM	Requirements Traceability Matrix	requirement → test → result → acceptance	ทำเพื่อให้ครบ template	Coverage
Defect Severity	ความรุนแรง	ผลกระทบต่อ value/user/risk	สับสนกับ Priority	Defect Priority
Defect Priority	ความเร่งด่วน	ต้องแก้เมื่อไร (PO ตั้ง)	สับสนกับ Severity	Defect Severity
Release Readiness	ความพร้อมขึ้น Production	scope/test/security/data/rollback/support/approval	เช็คเฉพาะ test ผ่าน	Go/No-Go
Go/No-Go	การตัดสินใจขึ้น Production	ตัดสินจาก readiness + risk + rollback + support	ตัดสินจากแรงกดดัน	Release Readiness
Residual Risk	ความเสี่ยงคงเหลือ	risk หลัง mitigation — ต้องมี owner	ลืมว่ายังเหลืออยู่	Risk
Acceptance Evidence	หลักฐานตรวจรับ	พิสูจน์ว่า deliverable ผ่าน criteria	คิดว่า sign-off พอ	UAT Sign-off
Test Strategy	แนวทางการทดสอบ	วางแผนไว้ Ch.5; ผลจริง = บทนี้	คิดว่าเท่ากับ Test Plan	Test Plan

10. Workshop

Scenario

[Teaching Scenario] ก่อน Planning Go/No-Go ของ SHG: UAT พบ payment defect (8% failure), 3 โรงแรมยังไม่ได้ทดสอบ cancellation scenario, RTM แสดง coverage 85%, rollback plan ยังไม่เคยทดสอบจริง, support team ยังไม่ได้ฝึก hypercare runbook

Your Role

PM (คุณสุทธิ) ที่ต้องเสนอ Go/No-Go recommendation พร้อม evidence

Available Information

- Test results, defect log (severity/priority), RTM, release readiness checklist (F.7), Scenario Master (Sprint 12 = soft launch 3 โรงแรม)

Missing Information

- สาเหตุ payment defect (gateway? flow? data?), ผล retest หลัง fix, ความพร้อม rollback test, training status ของ support team, ความเห็น Sponsor ต่อ conditional approach

Decision Required

- Go / No-Go / Conditional Go? ถ้า conditional: เงื่อนไขอะไร owner ใคร due date เท่าไร?

Constraints

- Launch ก่อน High Season (1 พ.ย.), ต้องไม่ขึ้น production โดยไม่มี rollback, Go/No-Go authority = Sponsor

Expected Output

- Release Readiness Report (ตาม F.7 checklist), Go/No-Go recommendation + options + trade-off, gap log (owner + due date), residual risk statement

Debrief Questions

1. ทำไม "defect น้อย" ยังไม่พอต่อการตัดสินใจ Go
2. Conditional Go ต่างจาก Go เต็มอย่างไร และเมื่อไรถึงสมเหตุสมผล
3. ถ้า Sponsor เร่งให้ Go ทั้งที่ rollback ยังไม่พร้อม คุณจะ defend อย่างไร

Evaluation Criteria

- Decision มี evidence (RTM, severity, retest, rollback, support), conditional มีเงื่อนไข + owner + due date, authority ถูกต้อง, ไม่ขัด Scenario Master

11. Checklist ใช้งานจริง (Verification/UAT / Phase F)

- ☐ Test Plan ตาม Test Strategy (Ch.5): scope, env, data, entry/exit
- ☐ SIT/System Test เสร็จก่อน UAT (ไม่ใช้ UAT แทน)
- ☐ UAT pack พร้อม: participants, scenarios, expected results, sign-off authority
- ☐ RTM ครอบคลุม 100% (requirement → test → result → acceptance)
- ☐ Defect log ครบ: severity (QA) + priority (PO) + owner + retest
- ☐ Retest + regression หลัง fix มี evidence
- ☐ Security/Performance accepted (PCI-DSS, 3s/5s/99.5%)
- ☐ Data migration tested; monitoring/backup/rollback ready
- ☐ Training + support พร้อม (Ch.9 hypercare)
- ☐ UAT sign-off เซ็นโดย acceptance authority (PO/คุณภัทร) พร้อม evidence
- ☐ Release Readiness Report (F.7) สมบูรณ์
- ☐ Go/No-Go decision บันทึกโดย Sponsor + residual risk statement
- ☐ Cross-ref: แผน Test = Ch.5, QA = Ch.6, ผลจริง = บทนี้

12. Assessment

1. **[Decision Case]** payment defect 8% + 3 โรงแรมยังไม่ทดสอบ — เสนอ Go/No-Go/Conditional พร้อม evidence + trade-off + authority
2. **[Decision Case]** PO อยากเซ็น UAT ทั้งที่ coverage 85% — คุณจะทำอย่างไร (options + risk + residual decision)
3. **[Artifact Review]** ตรวจ UAT dashboard สีเขียวที่ไม่มี coverage/severity/retest/acceptance — ระบุสิ่งที่ขาดและผล
4. **[Scenario Case]** defect 20 รายการ cosmetic กับ 1 รายการ payment critical — อธิบาย severity vs priority และการตัดสินใจ
5. **[Trade-off Case]** Soft Launch 3 โรงแรม (conditional) vs Full Launch ทั้ง 12 โรงแรม — trade-off ต่อ campaign, support และ trust
6. **[Cross-Knowledge Analysis]** ถ้า RTM พบ coverage gap 15% — กระทบ UAT schedule (Ch.8) และ Go-live (Ch.9) อย่างไร และทางเลือกคืออะไร
7. **[Recall]** องค์ประกอบของ Release Readiness (F.7) มีอะไรบ้าง (ระบุ ≥ 7)

13. Executive Summary

มิติ	สรุป
Why	Verification/UAT คือหลักฐานว่า "พร้อมขึ้น Production" — ไม่ใช่แค่สีเขียวบน dashboard
Decision Improved	Go/No-Go ตัดสินจาก evidence + risk + rollback + support

มิติ	สรุป
Failure Prevented	ป้องกัน UAT พิธี, defect แฝง, coverage gap และ go-live ที่ไม่มีหลักประกัน
What to Monitor	RTM coverage, defect severity ageing, retest rate, readiness checklist status

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: Go/No-Go ผ่าน (conditional: soft launch) — Ch.9 (Go-Live & Hypercare) จะบริหาร Cutover, Rollback, Monitoring และ Stabilization ตามแผนที่วางใน Ch.5 (C17)

Field	Handoff
Input	Deliverables + evidence (Ch.6), Test Strategy (Ch.5), Change Log (Ch.7)
Output	Test Results, UAT Sign-off, Release Readiness Report, Go/No-Go Decision
Creator	QA Lead (test), Business Users (UAT), PM + Release Mgr (readiness)
Artifact owner	QA Lead / Acceptance Authority / PM
Reviewer	Tech Lead, PO, Security, Ops
Approval authority	Sponsor / Business Owner (Go/No-Go)
Minimum acceptance	Usable (ต้องมี RTM + severity + readiness + sign-off)
Next chapter use	Ch.9 ใช้ Go decision + cutover/rollback plan + support readiness เพื่อ Go-Live & Hypercare

15. Quick Reference Card

```
VERIFICATION/UAT (F) — เป้าหมาย: พิสูจน์พร้อม Production
------------------------------------------------------------
Test levels: Unit -> SIT -> Performance -> Security -> Regression -> UAT
UAT: scope/participants/env/data/scenarios/entry-exit/sign-off authority
RTM: requirement -> design -> build -> test -> result -> acceptance
Defect: Severity (QA) ≠ Priority (PO); retest + regression มี evidence
Readiness (F.7): scope/test/security/perf/data/backup/rollback/training/support/approval
Go/No-Go: Sponsor ตัดสินจาก evidence + risk; conditional ต้องมีเงื่อนไข + owner + due date

กฎ 3 ข้อห้าม:
- ห้ามใช้ UAT แทน system test
- ห้าม Go โดยไม่มี rollback ที่ทดสอบแล้ว
- ห้าม sign-off แบบพิธี โดยไม่มี coverage/evidence

Cross-ref: แผน Test = Ch.5 | QA = Ch.6 | ผลจริง = บทนี้
```

Chapter 09 — Go-Live และ Hypercare: ขึ้น Production อย่างควบคุมได้

1. Opening Scenario Hook

[Teaching Scenario] Go/No-Go ผ่านแบบ conditional — SHG เตรียม Soft Launch 3 โรงแรมในคืนวันศุกร์ (ก่อน High Season 1 พ.ย.) คุณสุทธิ (PM) ตรวจ checklist พบว่า: cutover plan มีขั้นตอนครบ แต่ Rollback Plan ยังไม่ถูกทดสอบจริง, support team ยังไม่ได้ซ้อม hypercare runbook, และคุณภัทร (VP Marketing) ถามว่า "แล้ว campaign เปิดวันไหน"

[PMBOK 8] บทนี้เป็นเนื้อหาใหม่ที่ e-Book เดิมไม่มี: Go-live ไม่ใช่ "กดปุ่ม deploy แล้วจบ" แต่คือการบริหาร **Cutover, Rollback, Monitoring และ Stabilization** อย่างเป็นระบบ — และ Go-live ไม่เท่ากับ Project Closure (นั่นคือ Ch.10)

2. Why It Matters

[Best Practice] โครงการจำนวนมากไปถึง go-live แต่พังเพราะ: ไม่มี cutover sequence ชัด, rollback ไม่เคยทดสอบ, ไม่มี command center, hypercare ไม่มี exit criteria หรือ communication ไปผิดคน — ต้นทุนของความผิดพลาดตอนนี้สูงที่สุดในโครงการ (ลูกค้าใช้จริง เงินจริง trust จริง)

[PMBOK 8] Go-Live คือจุดที่ output เริ่มกลายเป็น outcome — แต่ benefit (35% direct booking ใน 18 เดือน) จะพิสูจน์ได้หลัง stabilization และถูกส่งต่อเป็นความรับผิดชอบของ business (Ch.10)

3. Mental Model

```
Go/No-Go Decision (Ch.8) + Cutover/Rollback Plan (Ch.5 C17)
  -> Entry Criteria Check (G.2)
  -> Cutover Window: Freeze -> Backup -> Deploy -> Configure
     -> Migrate Data -> Reconcile -> Smoke Test -> Business Verify
  -> Enable Users -> Monitor
  -> Rollback? (trigger -> decide ภายในเวลาจำกัด -> restore)
  -> Hypercare Command Center (triage, SLA, daily review)
  -> Stabilization Criteria (G.8) ผ่าน
  -> Operations Acceptance -> Ch.10 (Transition & Closure)
```

4. Main Lesson

4.1 Entry Criteria (G.2)

[Best Practice] ก่อนเริ่ม cutover ต้องมี: Go Decision, Approved Cutover Plan, Rollback Plan (ทดสอบแล้ว), Production Access, Data Ready, Support Team Ready, Monitoring Ready, Communication Ready, Owners Available

4.2 Cutover Plan (G.3)

[Best Practice] ต้องระบุ: Task, Sequence, Start/End, Owner, Dependency, Validation, Decision Point, Rollback Trigger, Communication, Evidence — cutover เป็น sequence ที่ทุกขั้นตอนมี owner และจุดตรวจ

4.3 Deployment Steps (G.4)

```
Freeze/Change Window -> Backup -> Deploy -> Configure -> Migrate Data
-> Reconcile Data -> Smoke Test -> Business Verification -> Enable Users
-> Monitor -> Communicate -> Confirm Production Status
```

4.4 Rollback Plan (G.5) — ห้ามขึ้น Production โดยไม่มี

[Best Practice] ต้องตอบ: Rollback Trigger (อะไร), Who Decides, Maximum Decision Time (กี่นาที), Backup Location, Restore Steps, Data Reconciliation, Communication, Business Continuity

[Teaching Scenario] SHG: trigger = payment failure rate > threshold หรือ critical booking flow ล่ม ภายใน 30 นาทีหลัง enable — ผู้ตัดสินใจ = คุณวีระ (CTO) + PM ภายใน 15 นาที — restore จาก backup + message ถึงลูกค้า/โรงแรม

4.5 Hypercare (G.6)

[Best Practice] Hypercare = ช่วงสนับสนุนเข้มหลัง go-live: Command Center, Support Hours, Incident Priority, Triage, Technical/Business Owners, Daily Review, Metrics, Exit Criteria

4.6 Hypercare Metrics (G.7) + Stabilization Criteria (G.8)

[Best Practice]

- **Metrics:** Incident Count, Severity, Response/Resolution Time, Transaction Success, Error Rate, Performance, User Adoption, Data Reconciliation, Support Volume
- **Stabilization:** ไม่มี Critical Incident ค้าง, Error Rate ใน threshold, Performance Stable, Business Transaction ถูกต้อง, Support รับช่วงได้, Known Issues มี Plan, Operations ยอมรับ Handover

[Teaching Scenario] SHG soft launch 3 โรงแรม: monitor payment success ≥ 99%, booking confirmation ≤ 5 วินาที, incident response ตาม SLA กับ 2C2P, daily review กับ hotel partners — stabilizaztion ผ่านเมื่อ 7 วันไม่มี critical incident + payment error < threshold

5. PM Decision Thinking

```
Decision: หลัง soft launch พบ payment failure rate สูงเกิน threshold — Rollback, Fix-in-place
หรือ Continue with mitigation?
Owner: Go-Live Authority (Sponsor/คุณจิรา) ตัดสิน; PM + CTO นำเสนอ options ภายในเวลาจำกัด
Inputs: rollback plan, monitoring data, defect triage, business impact, support capacity,
cutover evidence
Options:
  A: Rollback (restore ระบบเดิม) — ปลอดภัยสุด แต่ลูกค้าเสีย trust + campaign กระทบ
  B: Fix-in-place (hotfix + monitor เข้ม) — เร็ว แต่ต้องมั่นใจว่า root cause ชัด
  C: Continue with mitigation (จำกัดฟีเจอร์, workaround) — กลาง ๆ แต่เสี่ยงสะสม
Trade-offs: speed to restore vs stability, trust vs momentum, local vs full impact
Risk: ตัดสินใจช้า -> ผลกระทบกระจาย; ตัดสินใจเร็วเกิน -> rollback เองก็เสียหาย
Evidence: incident log, rollback trigger check, decision record ภายในเวลาจำกัด,
communication
Next Action: ตาม decision -> stabilize -> monitor -> ขยาย/ถอนตาม criteria -> Ch.10
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario]

- **Cutover (Soft Launch 3 โรงแรม):** คืนวันศุกร์ 02:00–06:00 — freeze, backup, deploy, migrate data 3 โรงแรม, reconcile (ตรงกับ Scenario Master Sprint 12)
- **Smoke test:** booking flow + payment + PMS sync ผ่าน
- **Enable users:** front desk + ลูกค้าเริ่มจอง — campaign ของคุณภัทรเปิดพร้อมกันแบบ pilot
- **Monitor:** พบ payment failure 4% (ต่ำกว่า trigger) + 2 incident minor (confirmation email ช้า) — fix-in-place + monitor ต่อ
- **Hypercare 2 สัปดาห์:** command center (PM, CTO, QA, support, 2C2P hotline), daily review, metrics tracking
- **ขยาย:** หลัง 7 วันเสถียร → ขยายไปอีก 3 โรงแรม → ครบ 12 ตามแผน full launch
- **Stabilization:** ผ่านเมื่อ error rate < threshold + support รับช่วงได้ → ส่งต่อ Ch.10

[PMBOK 8] สังเกตว่า decision แต่ละจุดมี trigger, เวลาจำกัด และ evidence — ไม่มี "เดี๋ยวค่อยว่ากัน"

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. ขึ้น Production โดยไม่มี Rollback Plan ที่ทดสอบจริง — ผล: ผิดพลาดแล้วฟื้นไม่ได้
2. Cutover ไม่มี sequence/owner ต่อขั้น — ผล: วุ่นวาย ไม่รู้ใครทำอะไร
3. ไม่มี Command Center / escalation path — ผล: incident วิ่งหาคนไม่มีเจ้าภาพ
4. Hypercare ไม่มี exit criteria — ผล: อยู่แบบไม่มีที่สิ้นสุด หรือออกเร็วเกิน
5. ใช้ Production data ผิด (test กับ data จริง) — ผล: ข้อมูลเสียหาย
6. Communication ไม่ถึงลูกค้า/โรงแรม — ผล: โรงแรมไม่รู้ว่าต้องเปลี่ยน workflow

7. คิดว่า Go-live = จบโครงการ — ผล: benefit/adoption ไม่มีเจ้าของ (Ch.10)

Common Misconceptions:

ความเข้าใจผิด	ความจริง
Go-live = Project Closure	Closure คือ Ch.10 (handover, benefit owner, lessons)
Rollback แพงเกินไป ต้องทำ	ไม่มี rollback = การพนันกับ production
Hypercare = ทีม support ธรรมดา	เป็นช่วงเข้มมี command center + metrics + exit criteria
Deploy ผ่าน = สำเร็จ	ต้อง monitor + stabilize + business verify
ระบบใหม่ต้องไม่มี incident	ต้องมี triage + known issues + plan
Cutover = ขั้นตอนเดียว	เป็น sequence ที่มี validation + decision point ต่อขั้น

8. Interview Questions

Foundational

- **Q:** Rollback Plan ต้องตอบคำถามอะไรบ้าง?
- **Answer Direction:** Trigger, ผู้ตัดสินใจ, เวลาจำกัด, backup location, restore steps, data reconciliation, communication, business continuity
- **Warning Signs:** ตอบว่า "กู้ backup กลับ" โดยไม่มี trigger/time/owner

Scenario

- **Q:** หลัง soft launch พบ payment failure สูงกว่า threshold — คุณจะทำอย่างไร?
- **Answer Direction:** เปิด trigger → นำเสนอ options (rollback/fix-in-place/mitigation) ภายในเวลาจำกัด → ตัดสินใจตาม evidence (root cause, business impact) → บันทึก decision → สื่อสาร (ลูกค้า/โรงแรม/support) → monitor หลัง action
- **Warning Signs:** รอให้ทีม "ดูอีกที" โดยไม่มีเวลาจำกัด หรือ rollback ทันทีโดยไม่ดู severity

Senior PM

- **Q:** คุณจะออกแบบ Hypercare อย่างไรให้มี exit ที่ชัดเจน?
- **Answer Direction:** กำหนด stabilization criteria (metrics + threshold + ระยะเวลา), command center + roles, daily review, incident SLA, known-issues plan — exit เมื่อ criteria ผ่าน + operations ยอมรับ handover; ถ้าไม่ผ่าน ขยาย hypercare พร้อมเหตุผล
- **Warning Signs:** ตอบว่า "คุม 2 สัปดาห์แล้วจบ" โดยไม่มี criteria

Executive

- **Q:** ทำไมต้องยอมเสียเวลา/เงินกับ cutover rehearsal และ rollback test?

- **Answer Direction:** rehearsal เผยปัญหาในสภาพปลอดภัย (ราคาถูก); rollback test ทำให้มั่นใจว่าถ้าผิดพลาดฟื้นได้ — ต้นทุนเล็กน้อยเทียบกับ downtime/campaign ล่ม/trust เสียหาย — เหมือนประกัน
- **Warning Signs:** ตอบว่า "เสียเวลาทำไม แค่ deploy"

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Cutover	การเปลี่ยนผ่านเข้าสู่ระบบใหม่	สลับจากระบบเดิมไประบบใหม่ใน Production	สับสนกับ Go-Live	Go/No-Go
Rollback	การย้อนกลับระบบ	แผน restore เมื่อ go-live ผิดพลาด	ไม่ทดสอบแล้วขึ้น Production	Cutover
Go-Live	การขึ้น Production	เริ่มให้ผู้ใช้ใช้งานจริง	คิดว่า = Closure	Hypercare
Hypercare	สนับสนุนเข้มหลัง Go-live	command center + triage + metrics + exit	คิดว่า = support ธรรมดา	Stabilization
Command Center	ศูนย์บัญชาการเหตุการณ์	จุดรวม incident/triage/decision หลัง go-live	ไม่มี escalation path	Incident
Stabilization	การเข้าสู่สภาวะเสถียร	incident ลด, performance คงที่, support รับช่วงได้	คิดว่า 2 วันจบ	Hypercare
Smoke Test	ทดสอบขั้นพื้นฐาน	ตรวจว่าดีploy ผ่าน/ระบบลุกก่อน enable users	ใช้แทน full test	Deployment
Data Reconciliation	ตรวจสอบข้อมูล	เทียบ data หลัง migration ให้ตรงกัน	ข้ามไป	Migration
Incident	เหตุการณ์ผิดปกติ	ต้องมี triage, severity, SLA, owner	สับสนกับ Defect (ช่วง dev)	Defect
Rollback Trigger	เงื่อนไขย้อนกลับ	ตัวเลข/สัญญาณที่บอกว่า "ถอย"	ไม่กำหนด	Rollback
Production Verification	ตรวจหลังขึ้นจริง	ยืนยันว่าระบบทำงานใน Production	ข้าม	Smoke Test

10. Workshop

Scenario

[Teaching Scenario] SHG เตรียม cutover soft launch 3 โรงแรมคืนวันศุกร์ — พบว่า: rollback ยังไม่เคยทดสอบจริง, support ยังไม่ได้ซ้อม runbook, campaign ของคุณภัทรจะเปิดวันจันทร์ถัดไป, และ GM โรงแรม 1 แห่งยังไม่ยืนยันความพร้อมของ front desk

Your Role

PM (คุณสุทธิ) ที่ต้องตัดสินใจ: เลื่อน cutover, ทำแบบมีเงื่อนไข, หรือเดินหน้าตามแผน

Available Information

- Go decision (conditional จาก Ch.8), cutover plan draft, monitoring/alert setup, Scenario Master (soft launch 3 โรงแรม = Sprint 12)

Missing Information

- ผล rollback rehearsal (ยังไม่ทำ), training status ของ front desk โรงแรมที่ 3, ความพร้อม 2C2P hotline, communication draft ถึงลูกค้า

Decision Required

- เดินหน้าคืนนี้ / เลื่อน 48 ชม. เพื่อปิด gap / ขึ้น 2 โรงแรมก่อน แล้วโรงแรมที่ 3 ตามหลัง — พร้อมเหตุผลและ owner ของ gap

Constraints

- Launch ก่อน High Season (1 พ.ย.), ห้ามขึ้น production โดยไม่มี rollback ที่ทดสอบแล้ว, campaign ต้องไม่เปิดก่อนระบบเสถียร

Expected Output

- Go/Defer decision record (options + trade-off + owner + due date), gap closure plan (rollback rehearsal, training, communication), hypercare plan หน้าเดียว (command center, roles, metrics, exit criteria)

Debrief Questions

1. ทำไม "เลื่อน 48 ชม." อาจถูกกว่าการขึ้นทั้งที่ gap ค้าง
2. Hypercare exit criteria ที่ดีควรเป็นอย่างไร
3. ถ้าเกิด incident กลาง cutover ใครตัดสินใจ rollback ภายในกี่นาที

Evaluation Criteria

- Decision มี evidence + trade-off, gap มี owner + due date, rollback trigger/time/owner ชัด, hypercare มี exit criteria, ไม่ขัด Scenario Master

11. Checklist ใช้งานจริง (Go-Live & Hypercare / Phase G)

☐ Entry criteria ครบ (Go decision, cutover plan, rollback tested, access, data, support, monitoring, communication, owners)

☐ Cutover plan มี sequence + owner + validation + decision point ต่อขั้น

- ☐ Rollback plan ทดสอบจริง: trigger, ผู้ตัดสินใจ, เวลาจำกัด, restore steps, reconciliation, communication
- ☐ Deployment steps ครบ (freeze → backup → deploy → configure → migrate → reconcile → smoke → business verify → enable)
- ☐ Monitoring + alert ตั้งก่อน enable
- ☐ Communication พร้อม (ลูกค้า, โรงแรม, support, partners)
- ☐ Hypercare: command center + roles + support hours + triage + SLA + daily review
- ☐ Metrics ติดตาม (incident, error rate, performance, transaction success, reconciliation)
- ☐ Stabilization criteria กำหนดไว้ล่วงหน้า
- ☐ Known issues มี plan + owner
- ☐ Operations ยอมรับ handover ก่อนจบ hypercare
- ☐ ส่งต่อ Ch.10: stabilization acceptance + handover pack + benefit baseline

12. Assessment

1. **[Decision Case]** หลัง soft launch พบ payment failure 6% (threshold 4%) — เสนอ rollback/fix-in-place/mitigation พร้อม trigger, เวลาจำกัด, decision owner
2. **[Decision Case]** คืนก่อน cutover พบว่า rollback ยังไม่ทดสอบ — คุณจะเดินหน้า/เลื่อน/แบ่งเฟสอย่างไร พร้อมเหตุผล
3. **[Artifact Review]** ตรวจ cutover plan ที่มี task list แต่ไม่มี owner/sequence/validation/rollback trigger ต่อขั้น — ระบุความเสี่ยงและวิธีแก้
4. **[Scenario Case]** โรงแรมที่ 3 ไม่พร้อม (front desk ยังไม่ซ้อม) — ตัวเลือกในการจัดการพร้อม trade-off
5. **[Trade-off Case]** Rollback (ลูกค้าเสีย trust, campaign กระทบ) vs Fix-in-place (เสี่ยง root cause ยังไม่ชัด) — อธิบายและให้คำแนะนำ
6. **[Cross-Knowledge Analysis]** ถ้า hypercare พบว่า PMS sync error 5% — กระทบ booking accuracy (Ch.9) และ benefit 35% direct booking (Ch.10) อย่างไร และทางเลือกคืออะไร
7. **[Recall]** Stabilization criteria (G.8) มีอะไรบ้าง (ระบุ ≥ 5)

13. Executive Summary

มิติ	สรุป
Why	Go-Live คือจุดที่ระบบถูกใช้จริง — ความผิดพลาดที่นี่แพงที่สุดในโครงการ
Decision Improved	rollback/fix/mitigation ตัดสินด้วย trigger + เวลาจำกัด + evidence
Failure Prevented	ป้องกันขึ้น production ไร้ rollback, hypercare ไร้ exit, และ communication ไร้เจ้าภาพ
What to Monitor	incident count/severity, error rate, performance, reconciliation, stabilization criteria

14. เชื่อมไปบทถัดไป + Artifact Handoff

Bridge: Stabilization ผ่าน + Operations ยอมรับ — โครงการเข้าสู่ช่วงปิด: Ch.10 (Transition & Closure) จะทำ Operational Handover, Final Acceptance, Contract/Financial Closure, Lessons Learned และ Benefit Handover (Integration ท่อนที่ 3: Close Project or Phase)

Field	Handoff
Input	Go decision (Ch.8), Environment/Release Plan (Ch.5), Incident/Hypercare Records
Output	Cutover/Rollback Evidence, Incident Log, Hypercare Report, Stabilization Acceptance
Creator	Release Manager + DevOps (cutover), Support + PM (hypercare)
Artifact owner	Release Manager / Hypercare Owner / Operations Owner
Reviewer	Tech Lead, Ops, Sponsor
Approval authority	Sponsor (go-live), Operations Owner (stabilization)
Minimum acceptance	Usable (ต้องมี rollback test + stabilization evidence)
Next chapter use	Ch.10 ใช้ stabilization acceptance + hypercare data ไปทำ Handover, Closure และ Benefit Handover

15. Quick Reference Card

```
GO-LIVE & HYPERCARE (G)
------------------------------------------------------------
Entry: Go decision + cutover plan + rollback tested + data/support/monitoring ready
Cutover: freeze -> backup -> deploy -> configure -> migrate -> reconcile
         -> smoke -> business verify -> enable -> monitor -> communicate
Rollback: trigger? ใครตัดสินใจ? กี่นาที? backup ที่ไหน? restore ยังไง?
          reconcile? สื่อสารใคร? business continuity?
Hypercare: command center + triage + SLA + daily review + metrics + exit criteria
Stabilization: ไม่มี critical incident + error rate ในเกณฑ์ + support รับช่วงได้

กฎ 3 ข้อห้าม:
- ห้ามขึ้น Production โดยไม่มี rollback ที่ทดสอบแล้ว
- ห้าม hypercare ไร้ exit criteria
- ห้ามคิดว่า Go-live = จบโครงการ (จบจริง = Ch.10)
```

Chapter 10 — Transition, Closure และ Benefit Handover: ปิดโครงการอย่างมืออาชีพ

1. Opening Scenario Hook

[Teaching Scenario] SHG ผ่าน stabilization 2 เดือนหลัง soft launch — ทีมลดเหลือ 6 คนตาม Operating Model (Scenario Master §16) และคุณสุทธิ (PM) เตรียม "ปิดโครงการ" แต่ยังมีคำถาม: ใครเป็น owner ของระบบหลังส่งมอบ, contract กับ 2C2P/PMS vendors ปิดยังไง, lessons learned เก็บไว้ที่ไหน, และที่สำคัญ — **ใครรับผิดชอบว่า direct booking จะถึง 35% ใน 18 เดือน**

[PMBOK 8] บทสุดท้ายของ workflow A→H: **Close Project or Phase** (Integration ท่อนที่ 3 ของ 3) — ปิดภาระผูกพันทั้งหมด, ส่งมอบให้ Operations, ปิด contract/การเงิน, เก็บบทเรียน และกำหนด Benefit Owner — **Go-live ไม่ใช่จุดจบ; Closure คือจุดที่ Output ถูกส่งต่อให้กลายเป็น Outcome และ Benefit**

2. Why It Matters

[Best Practice] โครงการที่ปิดไม่ดี: ระบบถูกส่งต่อแบบไม่มี owner → หลัง 3 เดือนไม่มีใครดูแล, benefit ไม่มีคนวัด → 35% direct booking ไม่รู้ว่าเกิดหรือไม่, lessons ไม่ถูกเก็บ → องค์กรวนทำผิดซ้ำ, และ contract ยังค้าง → ภาระผูกพันทางการเงิน/กฎหมาย

[PMBOK 6] Closing ไม่ใช่แค่ "ส่ง deliverable" — ครอบคลุม acceptance, knowledge transfer, contract/admin closure, open-item ownership และ transition to operation — นี่คือจุดที่บทเรียน lesson-01 กลับมา: Output ต่างจาก Outcome และ Benefit ต้องมี owner หลัง handover

3. Mental Model

```
Stabilization Acceptance (Ch.9)
  -> Operational Handover (H.3) — owner, support model, SLA, runbook
  -> Final Acceptance (H.4) — deliverables ครบ + sign-off
  -> Contract/Financial Closure (H.5) — invoice, payment, vendor, PO
  -> Lessons Learned (H.6) — what worked/didn't + action owner
  -> Benefit Handover (H.7) — benefit owner, measure, baseline, review date
  -> Closure Report (H.8) + Sponsor Approval (H.9 exit criteria)
```

4. Main Lesson

4.1 Operational Handover (H.3)

[Best Practice] ต้องมี: System Ownership, Support Model, SLA, Runbook, Monitoring, Incident Process, Backup, DR, Access, Vendor Contact, Known Issues, Maintenance

[Teaching Scenario] SHG: ทีมเหลือ 6 คน (2 Frontend, 2 Backend, 1 QA, 1 DevOps) — PM ส่งมอบ runbook + monitoring + incident process ให้ Operations; hypercare (Ch.9) เปลี่ยนเป็น support ตาม SLA; known issues (เช่น PMS sync error 5%) มี plan + owner

4.2 Final Acceptance (H.4)

[Best Practice] ตรวจ: Contract Deliverables, Acceptance Evidence, Outstanding Items, Waiver, Warranty, Payment, Sign-off — acceptance ต้องอ้าง evidence จาก Ch.8 (UAT sign-off) ไม่ใช่แค่ "ดูแล้วโอเค"

4.3 Contract และ Financial Closure (H.5)

[Best Practice] Final Invoice, Payment, Vendor Closure, Asset Transfer, License, PO Closure, Budget Reconciliation — ปิดกับ 2C2P, PMS vendors, AWS, Design Agency ตามสัญญา (Ch.1–2 ตั้งต้น vendor relationship, Ch.5 วาง procurement)

4.4 Lessons Learned (H.6)

[Best Practice] เก็บ: What Worked, What Did Not, Root Causes, Decisions, Risk Outcomes, Estimate Accuracy, Stakeholder Lessons, Technical Lessons, Recommended Actions, Owner for Improvement — lessons ที่ไม่มี action owner = แค่บันทึก

[Teaching Scenario] SHG lessons: PMS PoC ก่อน Sprint 1 ช่วยลด risk จริง, fast-track Back Office เพิ่ม defect (ควรประเมิน quality impact ให้ดี), test data readiness ต้องเข้า DoR ตั้งแต่แรก

4.5 Benefit Handover (H.7) — Output ≠ Benefit

[PMBOK 6] ต้องระบุ: Benefit Owner, Benefit Measure, Baseline, Target, Measurement Date, Data Source, Review Cadence

[Teaching Scenario] SHG: Benefit = Direct Booking 10% → 35% ภายใน 18 เดือน — Owner = คุณภัทร (VP Marketing) + คุณนภา (PO) ตาม Operating Model — Measure = % direct booking จากระบบ — Baseline = 10% — Review = รายเดือน post-launch (ตรง Scenario Master §14–15) — **PM ไม่ใช่ benefit owner หลังปิดโครงการ**

4.6 Closure Report (H.8) + Exit Criteria (H.9)

[Best Practice] Closure Report: Objectives, Scope Delivered, Acceptance, Schedule, Cost, Quality, Risks, Changes, Benefits, Outstanding, Lessons, Handover, Final Approval

Exit Criteria: Deliverables Accepted, Operations Handover Complete, Support Ready, Financial Closure, Contract Closure, Lessons Captured, Resources Released, Benefit Owner Assigned, Sponsor Approves Closure

5. PM Decision Thinking

```
Decision: โครงการ SHG พร้อมปิดหรือยัง — หรือต้องปิด open items ใดก่อน (เช่น PMS known issue, contract
2C2P)
Owner: PM เสนอ closure; Sponsor อนุมัติ; Operations Owner รับ handover; Benefit Owner (คุณภัทร)
รับ benefit
Inputs: closure report draft, handover pack, acceptance evidence, contract status, lessons,
benefit plan
Options:
  A: ปิดเลย — ทุก exit criteria ผ่าน
  B: ปิดแบบมีเงื่อนไข (open items มี owner + due date) — เช่น known issue ฝาก Operations ตาม SLA
  C: ยังไม่ปิด — ขาด acceptance/contract/benefit owner ที่จำเป็น
Trade-offs: release resources early vs completeness, momentum vs governance
Risk: ปิดเร็ว -> ภาระผูกพันค้าง; ปิดช้า -> ทรัพยากรผูกเปล่า
Evidence: signed acceptance, handover acceptance, closure report, sponsor approval
Next Action: หลังปิด -> PIR/benefit review ตาม cadence (คุณภัทร) + lessons นำไปใช้
```

6. ตัวอย่างจริงจาก Case ต่อเนื่อง (SHG)

[Teaching Scenario]

- **Handover:** runbook + monitoring + SLA ส่งให้ Operations; ทีมเหลือ 6 คน; known issues (PMS sync 5% → กำลังแก้ไปสู่ ≥98% ตาม Scenario Master §15)
- **Final Acceptance:** คุณภัทร + คุณนภา เซ็น acceptance อ้าง UAT evidence (Ch.8) + waiver สำหรับ cosmetic defect
- **Financial/Contract:** ปิด 2C2P (settlement reconciliation), PMS vendors (final T&M ตาม cap), AWS (transfer ownership ของ account), Design Agency (final payment ตาม milestone)
- **Lessons:** 3 บทเรียนหลัก + action owners
- **Benefit Handover:** Owner = คุณภัทร, Target = 35% ภายใน 18 เดือน, Baseline = 10%, Review = รายเดือน, Data source = booking analytics
- **Closure:** Sponsor (คุณจิรา) อนุมัติ — PIR หลัง 6 เดือนตรวจ benefit

[PMBOK 8] สังเกตว่า "ความสำเร็จของโครงการ" ณ จุดนี้คือ "ส่งต่อได้อย่างเป็นระเบียบ" — benefit จริง (35%) จะวัดกันทีหลัง โดยคนที่รับผิดชอบคือ business ไม่ใช่ทีม project

7. จุดตัดสินใจและกับดักที่พบบ่อย

Common Mistakes:

1. คิดว่า Go-live (Ch.9) = ปิดโครงการ — ผล: ไม่มี handover/benefit owner
2. ส่งระบบให้ Ops โดยไม่มี runbook/SLA/known-issue plan — ผล: support งง หลัง 3 เดือนพัง
3. Acceptance จากการ "ดูแล้วโอเค" ไม่มี evidence — ผล: dispute ตอนจ่ายเงิน/ปิด
4. ไม่ปิด contract/finance — ผล: ภาระค้าง ใบแจ้งหนี้เกิน
5. Lessons learned เป็นพิธี ไม่มี action owner — ผล: องค์กรวนทำผิดซ้ำ

6. Benefit ไม่มี owner หลังปิด — ผล: 35% ไม่มีใครวัด (lesson-01 บทเรียนเดิม)
7. ปล่อย resource โดยไม่ release อย่างเป็นทางการ — ผล: ทีมถูกดึงไปงานอื่นยังไม่ปิด

Common Misconceptions:

ความเข้าใจผิด	ความจริง
Project จบเมื่อ system ส่งมอบ	จบเมื่อ exit criteria ครบ (H.9)
PM เป็น benefit owner หลังปิด	Benefit Owner ต้องเป็น business (คุณภัทร)
Lessons learned = ประชุมเล่ากันฟัง	ต้องมี register + action owner
Acceptance = เซ็นรับของ	ต้องอ้าง evidence (Ch.8 UAT sign-off)
Closure = เอกสารเยอะ	Closure = ปิดภาระผูกพัน + เก็บบทเรียน + ตั้ง benefit
Handover = ส่งไฟล์	Handover = owner + support model + SLA + known issues

8. Interview Questions

Foundational

- **Q:** องค์ประกอบของ Operational Handover (H.3) มีอะไรบ้าง?
- **Answer Direction:** System Ownership, Support Model, SLA, Runbook, Monitoring, Incident Process, Backup, DR, Access, Vendor Contact, Known Issues, Maintenance
- **Warning Signs:** ตอบ "ส่งเอกสารให้ Ops"

Scenario

- **Q:** หลัง stabilization คุณพบว่า PMS known issue ยังค้าง (sync 5% error) — คุณจะปิดโครงการอย่างไร?
- **Answer Direction:** ไม่ปิดแบบไร้เงื่อนไข — ฝากเป็น known issue ใน handover (มี owner + SLA + due date), บันทึกใน closure report + residual risk, และเชื่อมกับ benefit plan (Inventory Sync Accuracy ≥ 98%) — closure แบบมีเงื่อนไขที่ชัดเจน
- **Warning Signs:** ปิดเลยโดยไม่พูดถึง known issue หรือเลื่อนปิดโดยไม่มีเหตุผล

Senior PM

- **Q:** Benefit (35% direct booking) จะถูกวัดหลังปิดโครงการ — คุณจะมั่นใจได้อย่างไรว่ามันจะเกิด?
- **Answer Direction:** ตั้ง Benefit Owner (คุณภัทร) + measure + baseline + target + review cadence + data source ก่อนปิด; ทำ benefit handover เป็นส่วนหนึ่งของ closure; จัด PIR ตามรอบ — PM ไม่ต้องเป็น owner แต่ต้องทำให้ ownership ชัดเจนก่อนจาก
- **Warning Signs:** ตอบว่า "ไม่ใช่เรื่องเราหลังปิด" โดยไม่จัด ownership

Executive

- **Q:** ทำไมต้องเสียเวลากับ Lessons Learned ในเมื่องานเสร็จแล้ว?

- **Answer Direction:** lessons ที่มี action owner ช่วยให้องค์กรไม่จ่ายค่า tuition ซ้ำ (estimate, vendor, test readiness) — ต้นทุนเล็กน้อย vs วนทำผิดซ้ำในโครงการถัดไป; เป็นส่วนหนึ่งของ organizational learning
- **Warning Signs:** ตอบว่า "ไม่มีเวลา" หรือทำเป็นพิธี

9. PM Dictionary ประจำบท

Term (EN)	คำอธิบาย (TH)	ความหมายเชิงปฏิบัติ	การใช้ผิดที่พบบ่อย	Term ที่เกี่ยวข้อง
Handover	การส่งมอบผลงาน	owner + support model + SLA + runbook + known issues	คิดว่า Go-live = handover สำเร็จ	Transition to Operation
Transition to Operation	ส่งต่อให้ฝ่ายปฏิบัติการ	วางแผนตั้งแต่ Planning (Ch.5 C17)	คิดว่า Go-live = ส่งมอบเสร็จ	Handover
Final Acceptance	ตรวจรับสุดท้าย	deliverables ครบ + evidence + sign-off	เซ็นแบบพิธี	Acceptance
Contract Closure	ปิดสัญญา	invoice, payment, vendor, PO, license	ค้างไว้ไม่มีใครปิด	Financial Closure
Lessons Learned	บทเรียน	what worked/didn't + action owner	ประชุมเล่ากันฟัง	PIR
Benefit Owner	เจ้าของ benefit	business คนที่วัดผล หลังปิดโครงการ	PM เป็น owner	Benefit
Benefit Measure	ตัววัด benefit	% direct booking, commission saved	ตั้งโดยไม่ผูก baseline	Baseline, Target
Closure Report	รายงานปิดโครงการ	objectives, scope, acceptance, cost, lessons, handover	เขียนสั้น ๆ ไม่มีหลักฐาน	Exit Criteria
PIR	Post-Implementation Review	ทบทวนหลัง implement เทียบ target	ทำช้าเกินไปจนไม่มีใครจำ	Lessons Learned
Exit Criteria	เกณฑ์ปิด	H.9 — ครบแล้ว Sponsor อนุมัติปิด	เปิดใช้เฉพาะตอนปิด	Closure Report
Open Items	รายการค้าง	ต้องมี owner + due date ก่อนปิด	ปิดโดยไม่พูดถึง	Waiver
Waiver	การสละสิทธิ์/เลื่อน	ยอมรับ item ที่เลื่อน อย่างเป็นทางการ	ใช้ปิดบังปัญหา	Open Items

10. Workshop

Scenario

[Teaching Scenario] SHG ใกล้ปิดโครงการ: ทีมลดเหลือ 6 คน, hypercare จบ, แต่ (1) PMS known issue (sync 5%) ยังค้าง, (2) 2C2P ยังมี final invoice ไม่ชัดเจน, (3) lessons learned ยังไม่ถูกเก็บ, (4) คุณภัทร ยังไม่ได้ยืนยันว่าเป็น benefit owner

Your Role

PM (คุณสุทธิ) ที่ต้องทำให้ Closure สมบูรณ์

Available Information

- Stabilization acceptance (Ch.9), UAT evidence (Ch.8), contract list (2C2P, PMS vendors, AWS, Design Agency), Scenario Master §14–16 (outcomes, benefits, operating model)

Missing Information

- ความชัดเจน final invoice 2C2P, SLA/owner ของ PMS known issue ที่ Ops ยอมรับ, กำหนดการ lessons workshop, confirmation ของคุณภัทรเรื่อง benefit review cadence

Decision Required

- ปิดเลย / ปิดแบบมีเงื่อนไข (open items + owner + due date) / เลื่อนปิด — และใครเป็น owner ของแต่ละ item

Constraints

- Exit criteria (H.9) ต้องครบก่อน Sponsor อนุมัติ, benefit owner ต้องเป็น business ไม่ใช่ PM, ห้ามปิดโดยไม่มี handover

Expected Output

- Closure recommendation + open-items log (owner + due date), handover pack checklist, lessons register (3 items + action owners), benefit handover plan (owner, measure, baseline, target, cadence)

Debrief Questions

1. Open items แบบไหนที่ "ปิดแบบมีเงื่อนไขได้" และแบบไหนที่ต้องปิดก่อน
2. ทำไม benefit owner ต้องชัดเจนก่อน Sponsor อนุมัติ closure
3. Lessons ที่ดีควรจบด้วยอะไร

Evaluation Criteria

- Exit criteria ถูกใช้จริง, open items มี owner + due date, benefit plan ครบ (owner/measure/baseline/target/cadence), lessons มี action owner, ไม่ขัด Scenario Master

11. Checklist ใช้งานจริง (Transition & Closure / Phase H)

☐ Operational Handover ครบ: owner, support model, SLA, runbook, monitoring, incident, backup/DR, access, vendor contact, known issues

☐ Final Acceptance อ้าง evidence (Ch.8) + waiver อย่างเป็นทางการ

☐ Contract/Financial Closure: invoice, payment, vendor, PO, license, budget reconciliation

- ☐ Lessons Learned register + action owners
- ☐ Benefit Handover: owner (คุณภัทร), measure, baseline (10%), target (35%/18 เดือน), data source, review cadence
- ☐ Closure Report ครบ (H.8)
- ☐ Exit criteria (H.9) ตรวจแล้ว: deliverables accepted, handover complete, support ready, finance/contract closed, lessons captured, resources released, benefit owner assigned
- ☐ Sponsor อนุมัติ closure
- ☐ PIR/benefit review กำหนดไว้ (เช่น 6 เดือน)
- ☐ ส่งต่อ: benefit measurement เป็นของ business; lessons กลับสู่องค์กร

12. Assessment

1. **[Decision Case]** PMS known issue ยังค้างก่อน closure — ปิดเลย/ปิดมีเงื่อนไข/เลื่อน พร้อมเหตุผล + owner + due date
2. **[Decision Case]** คุณภัทรยังไม่ยืนยันบทบาท benefit owner — คุณจะทำอย่างไรก่อน Sponsor อนุมัติ closure
3. **[Artifact Review]** ตรวจ Closure Report ที่ไม่มี handover/lessons/benefit owner — ระบุสิ่งที่ขาดและผล
4. **[Scenario Case]** 2C2P final invoice ไม่ตรงกับที่ตกลง — ขั้นตอนการจัดการ (H.5) พร้อม evidence
5. **[Trade-off Case]** ปิดเร็ว (ปล่อยทีม) vs ปิดช้า (เก็บทีมไว้) — trade-off ต่อ open items และต้นทุน
6. **[Cross-Knowledge Analysis]** ถ้า 6 เดือนหลัง launch direct booking ได้ 20% (ไม่ถึง 35%) — ใครทำอะไร (benefit owner, PM? PIR) และระบบที่ปิดไปแล้วจะจัดการอย่างไร
7. **[Recall]** Exit Criteria (H.9) มีอะไรบ้าง (ระบุ ≥ 6)

13. Executive Summary

มิติ	สรุป
Why	Closure คือจุดที่โครงการส่งต่อ output ให้ business สร้าง outcome/benefit ต่อ
Decision Improved	ปิดด้วย exit criteria + open-items owner แทน "ดูว่าเสร็จแล้ว"
Failure Prevented	ป้องกัน handover ไร้เจ้าภาพ, contract ค้าง, lessons ที่ไม่ถูกใช้ และ benefit ไร้ owner
What to Monitor	Benefit trend (35% ใน 18 เดือน), known issues, PIR schedule

14. จบเล่ม + Artifact Handoff

จบ workflow A→H: จาก Opportunity (Ch.1) ถึง Closure (บทนี้) — ผู้อ่านที่ทำครบทั้ง 11 บทจะเห็นวงจรครบ: Value (Ch.1) → Authorization (Ch.2) → Scope (Ch.3) → Plan (Ch.4–5) → Execute (Ch.6) → Control (Ch.7) → Verify (Ch.8) → Go-Live (Ch.9) → Close (Ch.10) — Appendix A–F เป็นเครื่องมือ lookup สำหรับ field use

Field	Handoff
Input	Stabilization acceptance (Ch.9), UAT evidence (Ch.8), contract list (Ch.1/Ch.5)
Output	Handover Pack, Final Acceptance, Contract/Financial Closure, Lessons Learned, Benefit Handover Plan, Closure Report
Creator	PM + Leads (handover), PM (closure report), PM + Business Owner (benefit)
Artifact owner	Operations Owner (handover), Business Owner (benefit), PM (closure)
Reviewer	Sponsor, Finance, Legal, Operations
Approval authority	Sponsor / Acceptance Authority
Minimum acceptance	Usable (exit criteria ครบ + open items มี owner)
Next use	Benefit review/PIR หลังปิด (business); lessons ใช้ในโครงการถัดไป

15. Quick Reference Card

```
TRANSITION & CLOSURE (H) — เป้าหมาย: ปิดภาระผูกพัน + ตั้ง benefit owner
------------------------------------------------------------
1. Handover   -> owner, support model, SLA, runbook, known issues
2. Acceptance -> deliverables + evidence + sign-off (อ้าง Ch.8)
3. Contract   -> invoice, payment, vendor, PO, license, reconciliation
4. Lessons    -> what worked/didn't + action owner
5. Benefit    -> owner (business), measure, baseline, target, cadence
6. Report     -> objectives, scope, acceptance, cost, lessons, handover
7. Exit       -> H.9 ครบ -> Sponsor อนุมัติ -> PIR ตามรอบ

กฎ 3 ข้อห้าม:
- ห้ามคิดว่า Go-live = จบ (จบจริงที่นี่)
- ห้ามปิดโดยไม่มี benefit owner
- ห้าม lessons ไม่มี action owner

Integration note: Charter = Ch.2 | Change = Ch.7 | Closure = บทนี้ — ครบ 3 ท่อน
```

Appendix A — Artifact Catalogue (Lookup Table)

ใช้เป็นตาราง lookup: ต้องการหลักฐานอะไร → ดูว่าอยู่ช่วงไหน (A–H) ใครเป็นเจ้าของ และจำเป็นระดับไหน (M = Mandatory, C = Conditional, R = Recommended, O = Optional) — "Mandatory" หมายถึงเนื้อหา/หลักฐานต้องมี ไม่จำเป็นต้องเป็นไฟล์ชื่อนั้น (Playbook §2.1)

A.1 Artifact ตามช่วง A–H

Artifact	Phase	Level	Owner หลัก	ใช้ทำอะไรต่อ
Opportunity Brief	A	R	Sales/Opportunity Owner	ตั้งต้น Discovery
Discovery Summary	A	M	PM + BA	basis ของ Proposal
Problem Statement	A	M	PM + BA	ผูก Business Need
Desired Outcomes / Success Measures	A	M	Customer Sponsor	วัด value (35% direct ใน 18 เดือน)
Assumption Log	A–H	M	PM	ฐานของ risk/change
Initial Stakeholder List	A	M	PM	ต่อยอด Register (Ch.2)
Solution Options	A	R	Architect + BA	เลือก approach
High-Level Scope	A	M	PM	Proposal In/Out of Scope
ROM Estimate	A	M	Functional Leads	ตั้งราคา/ตัดสินใจ Bid
Initial Risk Register	A	M	PM	ต่อยอด Risk (Ch.5)
Proposal	A	M (External)	PM + Sales	สัญญา/ลูกค้า review
SOW / Contract	A	M (External)	Legal + PM	Authorization (Ch.2)
Bid / No-Bid Decision	A	R	CEO/Commercial	decision record
Project Charter	B	M	Sponsor	อำนาจ PM + ทิศทาง
Stakeholder Register + Engagement Strategy	B	M	PM + BA	engagement + requirement (Ch.3)
Governance Model / Decision Rights	B	M	PM + Sponsor	ใครตัดสินใจอะไร
RACI	B/C	R	PM + Leads	clarity บทบาท

Artifact	Phase	Level	Owner หลัก	ใช้ทำอะไรต่อ
RAID Log	B–H	M	PM	ต่อเนื่องทุกช่วง
Kickoff Deck + MoM	B	R/M	PM	alignment + actions
Requirements Management Plan	C	M (as content)	BA + PM	วิธีเก็บ/อนุมัติ/trace
SRS / FSD (หรือสิ่งทดแทน)	C	C	BA	coverage ของ requirement
RTM	C–F	M/C	BA + QA	requirement→test→acceptance
Project Scope Statement	C	M (Predictive)	PM + PO	boundary
WBS + WBS Dictionary	C	M (Predictive)	PM + Leads	schedule/cost/resource (Ch.4)
Product Backlog	C/D	M (Agile)	PO	sprint delivery
Activity List + Dependency Network	C	M (Predictive)	PM + Planner	CPM/baseline
Schedule Baseline	C	M (Predictive)	PM	วัด variance (Ch.7)
Cost Baseline	C	M (Predictive)	PM + Finance	EVM/control (Ch.7)
Quality Plan / Test Strategy	C/F	M	QA Lead + PM	**วางแผนไว้ Ch.5, ผลจริง Ch.8**
Risk Register + Response Plan	B–H	M	Risk Owners	monitor/implement (Ch.6–7)
Communication Plan	C	M (as content)	PM	สื่อสารทุกช่วง
Resource Plan	C	M	PM + Functional Mgrs	capacity/sign-off
Procurement Plan	C	C	Procurement Mgr	vendor/contract
Release / Environment Plan	C	M/C	DevOps + PM	rollout (Ch.9)
Change Log + Decision Record	E	M	PM/CCB	baseline update
Status Report	E	M	PM	สถานะ/decision ask
UAT Plan + UAT Sign-off	F	M	QA + PO	acceptance evidence
Release Readiness Report	F	M	PM + Release Mgr	Go/No-Go
Cutover Plan + Rollback Plan	F/G	M (Production)	Release Mgr + DevOps	Go-Live (Ch.9)
Hypercare Plan + Report	G	C/M	Support + PM	stabilization
Handover Pack	H	M	Leads + Support	Ops รับช่วง
Closure Report	H	M	PM	Sponsor อนุมัติปิด
Lessons Learned	H	M	PM + Team	องค์กรเรียนรู้

Artifact	Phase	Level	Owner หลัก	ใช้ทำอะไรต่อ
Benefit Handover Plan	H	M	PM + Business Owner	วัด 35% หลังปิด

A.2 Index ตามเรื่อง (Process-to-Artifact สรุปจาก Playbook §5)

เรื่อง	Artifact หลัก
Scope	Requirements, Scope Statement, WBS + Dictionary, Scope Baseline, Backlog, RTM, Acceptance Criteria, Change Log
Schedule	Activity List, Dependency Network, CPM, Float, Schedule Baseline, Recovery Plan
Cost	Cost Estimate, Basis of Estimate, Budget, Cost Baseline, EVM Report
Quality	Quality Plan, Test Strategy, Checklist, Test Cases, Defect Log, UAT, Acceptance Evidence
Resource	Team Structure, RACI, Resource Calendar, Capacity Plan, Training Plan
Risk	Risk Approach, Risk Register, Response Plan, Issue Log, Contingency

A.3 Minimum Practical Pack ตามขนาดโครงการ (Playbook §6)

ขนาด	ต้องมี
Small	Charter, Stakeholder List, Scope/Backlog, Timeline, RAID, Acceptance Criteria, Status, UAT/Acceptance, Go-live Checklist, Handover
Medium	+ Requirements Repository, WBS/Release Plan, Cost, Resource Plan, Communication, Quality/Test Strategy, Change Control, Cutover, Hypercare
Large/Regulated	+ Formal Business Case, Detailed SRS/FSD, RTM, Architecture, Security/Privacy, Procurement, Quantitative Risk, Formal Baselines, Steering Governance, Data Migration Rehearsal, DR, Audit Evidence

> หลักการ: ทุก Artifact ต้องมี Purpose และ Owner — ห้ามสร้างเพื่อให้ครบ Template (Golden Rule #18)

Appendix B — Golden Rules for Project Managers (พิมพ์เป็น 1 หน้า)

มาจาก Playbook V2 §9 — ใช้เป็น poster ติดโต๊ะ/ห้องประชุม — แต่ละข้อ map ไปบทที่เกี่ยวข้อง

#	Golden Rule	บทที่เกี่ยวข้อง
1	อย่ารับปากก่อนมี Basis	Ch.1 (Pre-sales)
2	อย่า Estimate คนเดียว	Ch.1, Ch.4
3	อย่าเรียก Feature List ว่า Scope ที่ชัด	Ch.3
4	อย่าสร้าง WBS เป็น Timeline	Ch.3
5	อย่าทำ CPM ก่อน Dependency และ Duration พร้อม	Ch.4
6	อย่าคิดว่า SRS/FSD สำคัญเพราะชื่อเอกสาร ให้ตรวจ Coverage	Ch.3
7	อย่าปล่อย Change ผ่านคำพูด	Ch.7
8	อย่ารอวันสุดท้ายจึง Validate Scope	Ch.7, Ch.8
9	อย่าใช้ UAT แทน QA	Ch.8
10	อย่าขึ้น Production โดยไม่มี Rollback	Ch.9
11	อย่าคิดว่า Go-live คือ Project Closure	Ch.9, Ch.10
12	อย่าปิด Project โดยไม่มี Operational Owner และ Benefit Owner	Ch.10
13	ทุก Assumption ต้องมี Validation Owner	Ch.1, Ch.5
14	ทุก Risk ต้องมี Owner	Ch.5, Ch.7
15	ทุก Action ต้องมี Due Date	Ch.2, Ch.6
16	ทุก Decision สำคัญต้องมี Evidence	Ch.7, Ch.10
17	ทุก Baseline Change ต้องผ่าน Authority	Ch.7
18	ทุก Artifact ต้องมี Purpose ไม่ใช่สร้างเพื่อให้ครบ Template	ทั้งเล่ม
19	เมื่อข้อมูลใหม่เกิดขึ้น ให้ Replan อย่างมี Governance	Ch.7
20	PM มีหน้าที่ Integrate ไม่ใช่ทำแทนผู้เชี่ยวชาญทุกคน	ทั้งเล่ม

เวอร์ชันย่อ (หัวใจ 5 ข้อ)

```
1. อย่ารับปากก่อนมี Basis        (Ch.1)
2. อย่าใช้ UAT แทน QA            (Ch.8)
3. อย่าขึ้น Production โดยไม่มี Rollback (Ch.9)
4. อย่าปิด Project โดยไม่มี Benefit Owner (Ch.10)
5. ทุก Decision สำคัญต้องมี Evidence (Ch.7)
```

Appendix C — PM Glossary (ฉบับรวมสำหรับเล่ม)

สำคัญ: คำศัพท์นี้เป็นฉบับรวมของเล่ม — คำทั้งหมดถูกเพิ่มเข้า `governance/PM_GLOSSARY.md` (ส่วน "Ver.2 — field-guide additions") แล้ว ไม่ใช่ glossary ที่ขัดกัน (ตาม master_plan §5 ข้อ 9) — ดูไฟล์ต้นทางเป็น authoritative

C.1 คำศัพท์ใหม่ของเล่ม (PMBOK 8 / workflow A–H)

Term (EN)	คำอธิบาย (TH)	ใช้ในบท
Bid / No-Bid	การตัดสินใจเสนอราคาหรือไม่ — ตัดสินด้วย value, capability, capacity, risk	Ch.1
Business Case	เหตุผลทางธุรกิจของโครงการ — ผูก value, cost, timeline, risk, approval	Ch.1
CCB	Change Control Board — อนุมัติ change ที่กระทบ baseline ตาม threshold	Ch.2, Ch.7
Cutover	การสลับจากระบบเดิมไประบบใหม่ใน Production	Ch.9
Definition of Done	เกณฑ์ว่างานเสร็จจริง (build + review + test + security + deploy + evidence)	Ch.6
Definition of Ready	เกณฑ์ว่างานพร้อมเริ่ม (scope, AC, test data, owner, estimate)	Ch.6
Discovery	การสำรวจทำความเข้าใจปัญหาก่อนออก solution	Ch.1
Fixed Price	ราคาคงที่ — เหมาะกับ scope ชัด	Ch.1, Ch.5
Go/No-Go	การตัดสินใจขึ้น Production — ตัดสินจาก readiness + risk + rollback + support	Ch.8, Ch.9
Governance	กลไกตัดสินใจและอนุมัติ — ใครตัดสินใจอะไร ตาม threshold ใด	Ch.2
Hypercare	ช่วงสนับสนุนเข้มหลัง Go-live — จนเข้าสู่สภาวะเสถียร	Ch.9
Impact Analysis	การวิเคราะห์ผลกระทบครบทุกด้านก่อน change	Ch.7
Issue	ปัญหาที่เกิดแล้ว — ต้องมี owner, action, escalation	Ch.5, Ch.7
Make-or-Buy	ทำเองหรือซื้อ — ตัดสินจาก core, cost, risk, capacity	Ch.5
MVP	Minimum Viable Product — scope ต่ำสุดที่พิสูจน์ value	Ch.3, Ch.4
Opportunity	โอกาสทางธุรกิจ — lead/RFP ที่ยังไม่ใช่โครงการ	Ch.1
PIR	Post-Implementation Review — ทบทวนหลัง implement เทียบ target	Ch.10
Proposal	ข้อเสนอโครงการ — ยังไม่ผูกพันเท่า SOW	Ch.1
RAID	Risks, Assumptions, Issues, Dependencies — log กลาง	Ch.2
Release Readiness	ความพร้อมขึ้น Production (F.7)	Ch.8

Term (EN)	คำอธิบาย (TH)	ใช้ในบท
Residual Risk	ความเสี่ยงคงเหลือหลัง response — ต้องมี owner	Ch.5
Rollback	การย้อนกลับระบบ — แผน restore เมื่อ go-live ผิดพลาด	Ch.9
ROM	Rough Order of Magnitude — ประมาณการระดับสูง ไม่ใช่ commitment	Ch.1
RTM	Requirements Traceability Matrix — requirement→test→acceptance	Ch.3, Ch.8
SIT	System Integration Test — ทดสอบรวมระบบ	Ch.8
SOW	Statement of Work — ขอบเขตงานที่ผูกพันตามสัญญา	Ch.1
Stabilization	การเข้าสู่สภาวะเสถียรหลัง go-live	Ch.9
T&M	Time and Material — จ่ายตามเวลา/วัสดุ ต้องมี cap + evidence	Ch.1, Ch.5
Test Strategy	แนวทางการทดสอบ — วางแผน Ch.5, ผลจริง Ch.8	Ch.5, Ch.8
UAT	User Acceptance Test — ผู้ใช้ธุรกิจตรวจรับ	Ch.8
Workstream	สายงานย่อยของโครงการ — แยกตาม deliverable	ทั่วเล่ม
Vendor	ผู้ขาย/คู่สัญญา — จัดการผ่าน contract/SLA/acceptance	Ch.1, Ch.5

C.2 คำศัพท์เดิม (จาก PM_GLOSSARY.md) ที่เล่มใช้อ้างอิง

Term (EN)	คำอธิบายสั้น
Acceptance	การยอมรับผลงานอย่างเป็นทางการ (≠ Approval)
Contingency Reserve	สำรองสำหรับ known-unknowns — PM ใช้ตาม governance
Critical Path	เส้นทางที่กำหนด Project Duration
Earned Value (EV)	มูลค่างานที่เสร็จจริง (วัดเป็นเงิน)
Iron Triangle	Scope–Schedule–Cost กระทบกัน
Management Reserve	สำรองสำหรับ unknown-unknowns — Sponsor เท่านั้น
Project Charter	เอกสารเริ่มต้นโครงการ — Sponsor ออก
RACI	Responsible, Accountable, Consulted, Informed
Scope Creep	ขอบเขตบานโดยไม่ผ่าน change control
Sponsor	ผู้สนับสนุนโครงการ — ทรัพยากร + อนุมัติ
Tailoring	การปรับวิธีบริหารให้เหมาะบริบท
Verification / Validation	ตาม spec / ตาม need
WBS	โครงสร้างจำแนกงานตาม deliverable

Term (EN)	คำอธิบายสั้น
WIP Limit	จำกัดงานค้าง (Kanban)

> ดูคำเต็ม + การใช้ผิดที่พบบ่อยได้ที่ `../../governance/PM_GLOSSARY.md`

Appendix D — Master Answer Key (รวมทุกบท)

Instructor-facing only. ไฟล์นี้คือดัชนีรวมคำตอบ Assessment ของทุกบท — คำตอบเต็มอยู่ใน answer-key ของแต่ละบท (ลิงก์ในตาราง) หลักการประเมินร่วมทุกบท: **decision-based ≥ 90% ของน้ำหนัก, ใช้ rubric เกณฑ์ผ่าน ไม่ใช่คำตอบเดียวที่ถูก** (ตาม `COURSE_STANDARD.md`)

ภาพรวม Assessment ต่อบท

บท	จำนวนข้อ	ลักษณะเด่น	Answer Key เต็ม
Ch.0	—	ไม่มี Assessment (บทปฐมบท) — มี Discussion Prompts	ch-00-answer-key.md
Ch.1	7	Pre-sales: Decision + Trade-off (Fixed vs T&M)	ch-01-answer-key.md
Ch.2	7	Initiation: Charter, Authority, Stakeholder	ch-02-answer-key.md
Ch.3	7	Scope/WBS: MVP, WBS, 100% Rule	ch-03-answer-key.md
Ch.4	7	Schedule/Cost/Resource + EVM Case	ch-04-answer-key.md
Ch.5	7	Risk/Procurement/Comms/Quality-Plan	ch-05-answer-key.md
Ch.6	7	Execution: DoD, QA/QC, Sprint กลางทาง	ch-06-answer-key.md
Ch.7	7	Monitoring: EVM, Change Control, Escalation	ch-07-answer-key.md
Ch.8	7	Verification/UAT: Release Readiness, UAT gate	ch-08-answer-key.md
Ch.9	7	Go-Live/Hypercare: Cutover, Rollback, Stabilization	ch-09-answer-key.md
Ch.10	7	Closure: Handover, Benefit Owner, Exit Criteria	ch-10-answer-key.md

เกณฑ์ผ่านร่วมทุกบท (rubric กลาง)

1. **Decision Case** — ผ่านเมื่อ: ระบุ missing information ได้, เสนอ options ≥ 3 พร้อมเกณฑ์เลือก, ไม่ตอบ "รับเลย/ทำเลย" โดยไม่มี evidence
2. **Trade-off Case** — ผ่านเมื่อ: ชี้ trade-off ทั้ง 2 ด้าน + แนะนำทางเลือกพร้อมเหตุผลที่ผูกกับ business objective ของ SHG (12M budget, 35% direct booking, launch ก่อน 1 พ.ย.)
3. **Artifact Review** — ผ่านเมื่อ: เจอจุดบกพร่องของ artifact ครบตามเกณฑ์ของบท (ไม่ใช่แค่ "ชื่อเอกสารมี")
4. **Executive Communication** — ผ่านเมื่อ: พูดกับผู้บริหารด้วย risk + option + recommendation ไม่ใช่ technical detail
5. **Cross-Knowledge Analysis** — ผ่านเมื่อ: เชื่อมผลกระทบข้ามบท (เช่น fast-track → defect → UAT scope) ครบสายเหตุผล
6. **Recall** — ตรวจว่าแม่นยำตรงกับ Playbook section ที่ระบุ (เช่น G.8, H.9, E.10)

สรุปคำตอบสำคัญรายบท (แนวตอบย่อ — ดูไฟล์เต็มเพื่อ rubric รายข้อ)

Ch.1 — Pre-sales

1. ลูกค้าขอเพิ่ม Mobile App → ไม่ตอบรับทันที, ขอ evidence + Impact Analysis, เสนอ Base/Option หรือ Phase 2
2. Sales เสนอ 9M แต่ ROM 9.5–12M → อย่าต่ำกว่า ROM โดยไม่มีเหตุผล; ลด scope / เพิ่ม budget / No-Bid
3. Fixed vs T&M เมื่อ PMS API ไม่ชัด → Hybrid: fixed core + T&M มี cap สำหรับ adapter
4. Proposal ไม่มี Out of Scope/Assumptions → ปฏิเสธส่ง; อ้างว่าผูก contract ภายหลัง
5. Paid Discovery Phase → อธิบาย CEO ด้วย value: ลด delivery risk > ค่าใช้จ่าย
6. launch ก่อน High Season แต่ PMS ใช้ 6 เดือน → ตั้งคำถาม feasibility ตั้งแต่ต้น, option รวม cutover windows
7. ROM ≠ ราคาผูกพัน → ROM คือช่วง estimate มี basis; ราคาผูกพันหลัง SOW

Ch.2 — Initiation

1. Sponsor อยากให้ PM จัดการเอง ไม่มี threshold → ต้องล็อก PM authority + escalation threshold ใน Charter
2. Stakeholder Register 4 คน → ขาดทีม 2C2P/PMS/โรงแรม; ใช้ RACI + power-interest สแกนต่อ
3. PM authority กว้าง vs แคบ → กว้าง = คล่องตัว, แคบ = ควบคุม; เลือกตามความเสี่ยงโครงการ
4. Charter PM เขียนเอง Sponsor เซ็น ไม่มี exit criteria → Charter ต้องมี goal/budget/threshold/exit criteria
5. Steering Committee + threshold → สร้าง escalation path ชัด, กัน PM รับความเสี่ยงคนเดียว
6. CFO ตัดสินใจงบทุกครั้ง → จัด role: CFO = sponsor finance, PM = day-to-day; threshold กำหนดไว้
7. Stakeholder Register vs Power-Interest Grid → Register = ข้อมูล, Grid = เครื่องมือจัดลำดับ engagement

Ch.3 — Scope/WBS

1. เพิ่ม loyalty program ใน MVP → ทดสอบกับ objective: ไม่ช่วย 35% direct booking ใน 18 เดือน → ไม่เข้า MVP
2. "รายงานทุกอย่าง" → ขอ criteria: ใครใช้, ตัดสินใจอะไร, ความถี่; ตัดให้เหลือที่ใช้งานจริง
3. WBS แตกตามแผนก → ผิด: ต้องแตกตาม deliverable (100% Rule)
4. SRS/FSD เต็ม vs User Story + AC → เลือกตามความเสี่ยง: fixed-price ใช้ SRS, ภายในใช้ story + AC
5. WBS "Payment Integration" → แตกถึง work package + criteria เสร็จ
6. PMS API ไม่รองรับ real-time → ตรวจ constraint ตั้งแต่ requirement, มี fallback design
7. 100% Rule → ลูกหลานรวมกัน = พ่อแม่พอดี ไม่เกินไม่ขาด

Ch.4 — Schedule/Cost/Resource

1. Payment Integration ช้า 1 สัปดาห์ → ดูก่อนว่าอยู่ critical path หรือไม่ → แล้วเลือก crashing/fast-track/re-baseline
2. 80 key users ไม่ว่าง UAT → มี mitigation ตั้งแต่วางแผน resource (ช่วงเวลา, สำรองผู้ใช้)
3. EVM: PV=6M, EV=5M, AC=6.5M → SPI 0.83, CPI 0.77 → behind + over budget → corrective action
4. Crashing vs Fast Tracking → crashing เพิ่ม cost ลด duration; fast-track ทำงานคู่ขนาน เพิ่ม rework risk
5. Gantt ไม่มี dependency/owner/basis → ปฏิเสธ; CPM ต้องพึ่ง dependency + duration ที่มี basis
6. Resource Plan + RACI สำหรับ UAT → ระบุใครทำอะไร + conflict (UAT users ทำงานประจำ) มีสำรอง

7. Effort ≠ Duration → effort คือคน-ชั่วโมง, duration คือเวลาจริง (คนเพิ่ม ≠ duration ลดเสมอ)

Ch.5 — Risk/Procurement/Comms/Quality-Plan

1. Risk "key users ไม่ว่าง UAT" → มี owner, มี trigger, มี response (สำรอง, ช่วงเวลา, incentive)
2. 2C2P fee ต่ำแต่ SLA อ่อน → ประเมิน cost of poor SLA (settlement delay) มากกว่า fee
3. Conditional gate vs เลื่อน 2 สัปดาห์ → gate แบบมีเงื่อนไข (risk-based) เมื่อ evidence พอ; เลื่อนเมื่อไม่
4. Risk register "ทีมไม่พร้อม" → ต้องมี: owner, likelihood/impact, response, trigger, วันที่ทบทวน
5. Test Strategy ล็อกก่อน build → กำหนด QA scope/เกณฑ์ผ่าน/รอบ ก่อนงานเริ่ม; เปลี่ยนทีหลังแพง
6. settlement 2 วันทำการ → กระทบ booking flow + hypercare; ต้องแจ้ง stakeholders ตั้งแต่ plan
7. Verification vs Validation → Verification = ตรง spec ไหม, Validation = ตรง need ไหม

Ch.6 — Execution

1. story เสร็จแต่ไม่มี unit test/security check → ยังไม่ Done; DoD เป็นสัญญาทีม ไม่ยืดหยุ่นตามแรงกด
2. flash sale กลาง sprint → เข้า change management: impact + decision โดยไม่ทำลาย sprint goal
3. sprint board ทุกชิ้น Done แต่ไม่มี evidence → Done ≠ done; ตรวจ artifact/definition
4. บังคับ DoD (ช้า 2 วัน) vs ปล่อยแล้วแก้ Ch.8 → บังคับ DoD; defect ช้าแพงกว่า
5. QA กับ dev ขัดแย้ง test data → PM แก้ root cause (test data plan) ไม่ใช่จับคนกลาง
6. 2C2P sandbox ช้า 1 สัปดาห์ → vendor dependency: escalate + fallback + กันผลกระทบต่อ sprint
7. QA vs QC → QA = กระบวนการ (ป้องกัน), QC = ตรวจผล (ตรวจจับ)

Ch.7 — Monitoring & Change

1. EVM: PV=7.5M, EV=6.75M, AC=7.5M → SPI 0.9, CPI 0.9 → behind + over → วิเคราะห์ variance ก่อน action
2. flash sale ก่อน launch → เข้า PICC: impact ต่อ scope/schedule/cost/risk → decision โดย CCB/sponsor
3. PMS vendor ขอ T&M เพิ่ม 300K → ตรวจ contract + change control; ไม่จ่ายนอก process
4. Change Log ไม่มี impact/baseline/approval → ปฏิเสธ; ทุก change ต้องมี 3 อย่างนี้
5. Sponsor อนุมัติด้วยวาจา → ขอ confirmation เป็นลายลักษณ์อักษร; oral ≠ change
6. fast-track ทำให้ defect เพิ่ม → ยอมรับ trade-off หรือชะลอ; ติดตาม quality metric ควบคู่
7. escalation (E.10) → เงื่อนไขชัด: budget/time threshold, sponsor unavailable, vendor breach

Ch.8 — Verification/UAT

1. payment defect 8% (threshold 4%) + 3 โรงแรมยังไม่ทดสอบ → ยังไม่ release-ready; เกณฑ์ผ่านไม่ยืดหยุ่น
2. PO อยากเซ็น UAT ที่ coverage 85% → ไม่; ขอ evidence + แผนปิด gap หรือ conditional sign-off ระบุความเสี่ยง
3. dashboard สีเขียวไม่มี evidence → ตรวจ evidence (log, จำนวนเคส, บันทึก) ไม่ใช่สี
4. cosmetic 20 ตัว vs payment critical 1 ตัว → priority ตาม impact; critical ก่อน, cosmetic เข้า backlog
5. Soft Launch 3 โรงแรม vs Full Launch → soft launch ลด risk, เก็บ learning; เหมาะเมื่อ readiness ไม่เต็ม
6. RTM coverage gap 15% → ไม่ผ่าน gate; ระบุ gap + แผนปิด ก่อน UAT สรุปผล
7. Release Readiness (F.7) → องค์ประกอบ: ผล QA/UAT, RTM ครบ, เกณฑ์ผ่าน, rollback, owner, sign-off

Ch.9 — Go-Live/Hypercare

1. payment failure 6% (threshold 4%) → อยู่เหนือเกณฑ์ → escalate; ระงับ/ชะลอ cutover ตามแผน
2. คืนก่อน cutover พบ rollback ยังไม่ทดสอบ → หยุด; rollback ไม่ทดสอบ = ยังไม่ go/no-go ได้
3. cutover plan ไม่มี owner/sequence/validation/trigger → ปฏิเสธ; 4 องค์ประกอบนี้ขาดไม่ได้
4. โรงแรมที่ 3 ไม่พร้อม → ลดขอบเขต cutover (เลื่อนโรงแรมที่ 3) หรือเลื่อนทั้ง cutover — ตาม impact
5. Rollback vs Fix-in-place → rollback เมื่อ risk สูง/ไม่รู้ root cause; fix-in-place เมื่อเล็กและรู้สาเหตุ
6. PMS sync error 5% หลัง hypercare → ตัดสินใจ: rollback บางส่วน / fix-in-place / ขยาย hypercare ตาม SLA
7. Stabilization criteria (G.8) → เกณฑ์ที่ทำให้ hypercare จบได้: metrics อยู่ในเกณฑ์ x วันติด

Ch.10 — Closure

1. PMS known issue ค้างก่อน closure → ไม่ออกจาก hypercare ยัง; ย้ายเป็น operational backlog + owner
2. คุณภัทรไม่ยืนยัน benefit owner → closure ไม่จบ; ต้องมี operational + benefit owner ระบุ
3. Closure Report ไม่มี handover/lessons/benefit → ไม่ผ่าน; 3 ส่วนนี้บังคับ
4. 2C2P final invoice ไม่ตรง → ตรวจ contract + deliverable + sign-off ก่อนจ่าย; dispute ผ่าน process
5. ปิดเร็ว (ปล่อยทีม) vs ปิดช้า → ปิดเร็วเสี่ยง knowledge loss; ปิดช้าเสีย cost — ชั่งกับ readiness จริง
6. 6 เดือนหลัง launch direct booking 20% (ไม่ถึง 35%) → ติดตาม benefit; วิเคราะห์ gap + แผนปรับ (ไม่ใช่ปิดแล้วลืม)
7. Exit Criteria (H.9) → ครบ: scope ส่งมอบ, acceptance, handover, lessons, benefit baseline, financial close

วิธีใช้

- **Instructor:** ใช้ rubric รายข้อในไฟล์ answer-key เต็มของแต่ละบท (ลิงก์ตารางบน) — ไฟล์นี้เป็น index สำหรับรวมผลสอบหลายบท
- **การให้คะแนนรวม:** ใช้ Assessment Composition ต่อบท (ใน answer-key เต็ม) — application/judgement ≥ 90% ของน้ำหนักตาม `COURSE_STANDARD.md`

Appendix E — SDLC Role & Output Matrix

สรุปบทบาทหน้าที่ตลอดกระบวนการส่งมอบโครงการ (A–H) แบ่งเป็น 2 ส่วน: (1) PM ทำ vs ทีมทำ (2) Output ที่ได้ → ใช้ทำอะไรต่อ
ใช้เป็น **Field Mode lookup** — เปิดหน้านี้เมื่ออยากรู้เร็วๆ ว่า Phase ไหนใครต้องทำอะไร และงานที่ได้ออกมาต้องส่งต่อให้ใคร

Phase	PM ทำ	ทีมทำ (ไม่ใช่ PM)	Output → ใช้ทำอะไรต่อ
A. Pre-sales	ประสาน Discovery, คุม Estimate, ทำ Proposal, วิเคราะห์ Impact ตอน Negotiate	Sales (รับ Opportunity, เจรจา), BA (Discovery/Root Cause), Architect (ออกแบบ Solution Options), Finance (ตั้งราคา), Legal (ทำ SOW)	Opportunity Brief, ROM Estimate, Proposal, Signed SOW → ใช้ตัดสิน Bid/No-Bid และเป็นฐานทำ Charter ใน Phase B
B. Initiation	ทำ Charter, วิเคราะห์ Stakeholder, ตั้ง Governance, เปิด RAID, จัด Kickoff	Sponsor (อนุมัติ Charter), BA (ช่วยวิเคราะห์ Stakeholder)	Project Charter, Stakeholder Register, Governance Model, RAID Log → เป็น "ใบอนุญาต" ให้เริ่มวางแผนละเอียดใน Phase C
C. Detailed Planning	เลือก Delivery Approach, รวม Plan ทุกด้านเป็น Integrated Plan, นำเสนอ Baseline Gate	BA (Requirements/SRS/FSD), Architect (Technical Design/TDD-SDD), QA (Test Strategy/Test Plan), Planner (CPM/Schedule), Finance (Cost), DevOps (Environment Plan)	WBS, **SRS/FSD (ตามเงื่อนไข), Technical Design (TDD/SDD), Test Strategy/Test Plan, RTM**, Schedule/Cost Baseline, Risk Register, Integrated PM Plan → เป็น "พิมพ์เขียว" ที่ทีมใช้ลงมือทำจริงใน Phase D และใช้วัด Variance ใน Phase E
D. Execution	Confirm งานพร้อมทำ (Definition of Ready), จัดการ Blocker, ติดตามความคืบหน้า	Delivery Team (Build ตาม Tech Spec), Tech Lead (Peer Review), QA (รัน Test ตาม Test Plan), Workstream Lead (ส่งมอบงาน)	Deliverables, Work Performance Data, Test Execution Evidence → ส่งเข้า Monitoring (E) เพื่อเทียบกับ Baseline และเตรียมเข้า UAT (F)
E. Monitoring & Change	เก็บ Actual เทียบ Baseline, วิเคราะห์ Variance, คุม Change Control	Workstream Leads (รายงานงานจริง), CCB/Sponsor (ตัดสิน Change ที่กระทบ Baseline), Customer (Validate Scope)	Variance/Forecast Report, Change Decision, Updated RAID → ทำให้ Baseline ทันสถานะจริง และเป็นหลักฐานประกอบ Go/No-Go ทีหลัง
F. Verification/UAT	ประสาน Release Readiness, สรุปเสนอ Go/No-Go	QA (Test/SIT ตาม Test Plan จาก C), Business Users (ทำ UAT จริง), Product Owner (อนุมัติด้าน Business), Sponsor (ตัดสิน Go/No-Go)	**Test Results (SIT/System Test)**, UAT Sign-off, Release Readiness Report, Go/No-Go Decision → เป็นเงื่อนไขเปิดทางเข้า Go-Live (G)

Phase	PM ทำ	ทีมทำ (ไม่ใช่ PM)	Output → ใช้ทำอะไรต่อ
G. Go-Live & Hypercare	คุม Cutover, ตัดสินใจร่วมเรื่อง Rollback ถ้าจำเป็น	DevOps (Deploy/Migrate ข้อมูล), QA+Business (Smoke Test/ตรวจธุรกิจ), Support (Monitor/Triage หลังขึ้นระบบ)	Deployment Evidence, Hypercare Report, Stabilization Acceptance → เป็นหลักฐานว่าระบบนิ่งพอจะส่งต่อให้ Operations ใน Phase H
H. Transition & Closure	ประสานปิดโครงการ, จัด Lessons Learned	Operations (รับ Handover ไปดูแลต่อ), Finance/Legal (ปิดสัญญา/การเงิน), Business Owner (รับผิดชอบ Benefit ต่อ)	Handover Pack, Closure Report, Benefits Handover Plan → โครงการปิดจบ ระบบเข้าสู่การดูแลปกติ และเริ่มวัดผล Benefit จริงหลังจบโปรเจกต์

Pattern ที่เห็นซ้ำทุก Phase

PM แทนไม่ได้ "ลงมือทำเนื้องาน" เองเลย บทบาทหลักของ PM คือ **ประสาน → ตัดสินใจในกรอบอำนาจ → ผลักดันให้ได้ Output → ส่งต่อ** ส่วนเนื้องานจริง (วิเคราะห์ ออกแบบ เขียน ทดสอบ อนุมัติเชิงธุรกิจ) เป็นของ Role เฉพาะทางเสมอ และ Output ของแต่ละ Phase จะกลายเป็น **Input บังคับ** ของ Phase ถัดไปเป็นทอดๆ ตลอดสาย A→H

เอกสารที่มักหาไม่เจอ: SRS / FSD / Tech Spec / Test Plan อยู่ตรงไหน

คำถามที่พบบ่อยที่สุดคือเอกสารเหล่านี้ "หายไปไหน" — คำตอบคือมันไม่ได้หาย แต่เป็น Output ของ **Phase C (Detailed Planning)** ไม่ใช่ Phase D:

- **SRS / FSD** — ผลผลิตของ BA ใน Phase C (Conditional: ต้องมีเมื่อ Contract Formal, ระบบซับซ้อน, ต้อง Sign-off — ดู Playbook V2 §C4) ถ้าตัดออกต้องมีเอกสารทดแทน เช่น User Story + Acceptance Criteria ที่ Coverage เทียบเท่า
- **Technical Design (TDD/SDD)** — ผลผลิตของ Architect/Tech Lead ใน Phase C เช่นกัน (หรือ ADR/Architecture Docs เป็นเอกสารทดแทน)
- **Test Strategy / Test Plan** — วางแผนใน Phase C (Plan Quality and Acceptance) แล้วนำไป **รันจริง** ใน Phase D และ F — ตัว Test Plan (แผน) กับ Test Results (ผลการรัน) จึงอยู่คนละ Phase กัน
- **RTM (Requirements Traceability Matrix)** — เริ่มสร้างใน Phase C แล้วอัปเดตต่อเนื่องถึง Phase F เพื่อโยง Requirement → Design → Test → Acceptance

PMBOK ไม่บังคับชื่อไฟล์เป๊ะ — สิ่งที่บังคับคือ "เนื้อหา" ต้องถูก Elicit, Analyze, Document, Approve, Trace ให้ครบ ไม่ว่าจะเก็บเป็นไฟล์แยกหรือรวมอยู่ใน Backlog/Wiki ก็ได้ (ดู Playbook V2 §C4.1)

Appendix F — Problem → Chapter Index

> หัวใจของ field manual: เจอปัญหาหน้างาน → เปิดตารางนี้ → ไปบทที่ตรง + ใช้ Checklist ท้ายบท และ Quick Reference Card — ไม่ต้องอ่านทั้งเล่ม

วิธีใช้

1. ค้นหาประโยคที่ใกล้เคียงกับปัญหาที่เจอในตารางด้านล่าง
2. เปิดบทที่ชี้ไว้ (เนื้อหาหลัก) — ถ้าเร่งด่วนให้เปิด **Checklist ท้ายบท** ก่อน
3. ถ้าปัญหาเกิดจากงานในบทก่อนหน้า ให้ดูคอลัมน์ "อาจต้องย้อนไป" เพื่อตรวจ root cause
4. Appendix เสริม: A — Artifact Catalogue, B — Golden Rules, E — Role/Output Matrix

ตาราง Problem → Chapter

#	ปัญหาที่เจอหน้างาน	บทหลัก	Checklist ท้ายบท	อาจต้องย้อนไป
1	ยังไม่มี Proposal/SOW ต้องประเมินราคาหรือตอบ RFP	Ch.1	Ch.1 §11	—
2	ลูกค้า/ฝ่ายขายกดดันให้รับปาก scope หรือราคา	Ch.1	Ch.1 §11	Ch.5 (risk)
3	Scope ยังคลุมเครือ ต้องแตกเป็นชิ้นงานให้ทีมทำได้	Ch.3	Ch.3 §11	Ch.2 (charter)
4	Feature List ถูกเรียกเป็น "Scope ที่ชัด" แต่ยังไม่ครบ	Ch.3	Ch.3 §11	Ch.1 (basis)
5	ลูกค้าขอเพิ่ม/ลดขอบเขตกะทันหัน	Ch.7	Ch.7 §11	Ch.3 (baseline)
6	WBS แตกไม่ถูก (เช่น แตกตามแผนก ไม่ใช่ deliverable)	Ch.3	Ch.3 §11	—
7	ทำ Schedule ไม่ได้ เพราะ dependency/duration ไม่พร้อม	Ch.4	Ch.4 §11	Ch.3 (WBS)
8	โครงการล่าช้ากว่าแผน / critical path มีปัญหา	Ch.4, Ch.7	Ch.4 §11, Ch.7 §11	Ch.3 (scope)
9	งบประมาณบานปลาย / ใช้เงินเกิน plan	Ch.7	Ch.7 §11	Ch.4 (cost baseline)
10	ต้องการรู้ว่าโครงการ "จริงๆ แล้ว" ไปได้ดีแค่ไหน	Ch.7 (EVM)	Ch.7 §11	Ch.4 (baseline)

#	ปัญหาที่เจอหน้างาน	บทหลัก	Checklist ท้ายบท	อาจต้องย้อนไป
11	ทีมไม่พอ / คนเก่งถูกดึงไปงานอื่น / ไฟล์ทีมขาด	Ch.4 (Resource)	Ch.4 §11	Ch.2 (RACI)
12	ทีมทำงานไม่ตรงตามที่ตกลง / Done ไม่มี evidence	Ch.6	Ch.6 §11	Ch.4 (plan), Ch.5 (QA plan)
13	QA กับ Dev ขัดแย้ง / test data ไม่พร้อม	Ch.6	Ch.6 §11	Ch.5 (test strategy)
14	อยากรู้ว่า "คุณภาพจะพอไหม" ก่อนส่งงาน	Ch.5 (Test Strategy), Ch.8 (QC)	Ch.5 §11, Ch.8 §11	—
15	ผลทดสอบยังไม่ครบ/defect เยอะ แต่โดนกดให้ผ่าน	Ch.8	Ch.8 §11	Ch.5 (เกณฑ์ผ่าน)
16	ลูกค้า/PO อยากเซ็น UAT ทั้งที่ coverage ไม่ครบ	Ch.8	Ch.8 §11	Ch.5 (UAT plan)
17	เจอความเสี่ยงใหม่/ความเสี่ยงที่เคยล็อกเกิดจริง	Ch.5	Ch.5 §11	Ch.1 (assumption)
18	Vendor (PMS/2C2P) ไม่ส่งงานตามสัญญา/แก้ช้า	Ch.5 (Procurement)	Ch.5 §11	Ch.1 (SOW)
19	สื่อสารกับ stakeholders ไม่ตรง / คนสำคัญไม่รู้เรื่อง	Ch.5 (Comms)	Ch.5 §11	Ch.2 (register)
20	Stakeholder สำคัญไม่โผล่ / ไม่ให้ความร่วมมือ	Ch.2	Ch.2 §11	—
21	Sponsor อนุมัติด้วยวาจา / ไม่มี threshold ชัดเจน	Ch.2, Ch.7	Ch.2 §11, Ch.7 §11	—
22	พรุ่งนี้จะขึ้น Production ต้องตรวจความพร้อม	Ch.9	Ch.9 §11	Ch.8 (release readiness)
23	ต้องวางแผน cutover / กลัวขึ้นแล้วพัง	Ch.9	Ch.9 §11	Ch.8 (UAT)
24	ระบบขึ้นแล้ว error/performance มีปัญหา	Ch.9 (Hypercare)	Ch.9 §11	Ch.8 (QC)
25	ต้องตัดสินใจ rollback หรือ fix-in-place	Ch.9	Ch.9 §11	—
26	งานเสร็จแล้วแต่ไม่รู้จะปิดโครงการยังไง	Ch.10	Ch.10 §11	Ch.9 (handover)
27	ไม่รู้ว่าใครรับงานต่อหลัง project จบ (ops/benefit owner)	Ch.10	Ch.10 §11	Ch.2 (charter roles)
28	ต้องสรุป lessons learned / closure report	Ch.10	Ch.10 §11	—
29	อยากรู้ภาพรวมว่า PM ต้องทำอะไรบ้างตลอดโครงการ	Ch.0	Ch.0 §11	Appendix E (role matrix)

#	ปัญหาที่เจอหน้างาน	บทหลัก	Checklist ท้ายบท	อาจต้องย้อนไป
30	ไม่รู้ว่าต้องสร้าง/ส่งเอกสารอะไรตอนไหน	Appendix A	—	Appendix E
31	ต้องการข้อควรระวังสรุป 1 หน้า	Appendix B	—	—
32	ไม่รู้คำศัพท์ PM ที่ใช้ในเล่มนี้	Appendix C	—	governance/PM_GLOSSARY.md

ดัชนีย้อนกลับ: บท → ปัญหาที่บทนั้นตอบ

บท	ปัญหาประเภทหลักที่บทนี้ตอบ
Ch.0	ภาพรวมกรอบคิด PMBOK 8, วิธีใช้เล่ม (Study/Field Mode)
Ch.1	Opportunity, Proposal, ROM/SOW, Fixed vs T&M, No-Bid
Ch.2	Charter, PM authority, Stakeholder, Steering Committee
Ch.3	Requirements, Scope Baseline, WBS, 100% Rule, MVP
Ch.4	Schedule (CPM), Cost (EVM พื้นฐาน), Resource/RACI, Agile
Ch.5	Risk register, Procurement/vendor, Comms plan, Test Strategy, Environment
Ch.6	Execution, DoD, Manage Quality/Team, Daily control, Agile sprint
Ch.7	Monitor, EVM analysis, Perform Integrated Change Control, Escalation
Ch.8	QC/Control Quality, UAT, RTM, Release Readiness gate
Ch.9	Cutover, Rollback, Go/No-Go, Hypercare, Stabilization
Ch.10	Handover, Closure Report, Benefit Owner, Lessons Learned, Exit Criteria

หมายเหตุ (สำหรับผู้ทำเล่ม)

- ตารางนี้เป็น "ดัชนีเดียวของเล่ม" ที่ต้องอัปเดตทุกครั้งที่มีการย้าย/เพิ่มหัวข้อในบท — ตรวจใน Phase 4 (รวมเล่ม) ว่าหมายเลขหัวข้อ Checklist ยังตรง
- กรณีปัญหาแบบผสม (เช่น งบเกิน + ลูกค้าโกรธ) ให้เปิดบทที่ใกล้เคียงที่สุดก่อน แล้วใช้คอลัมน์ "อาจต้องย้อนไป" ไล่ root cause