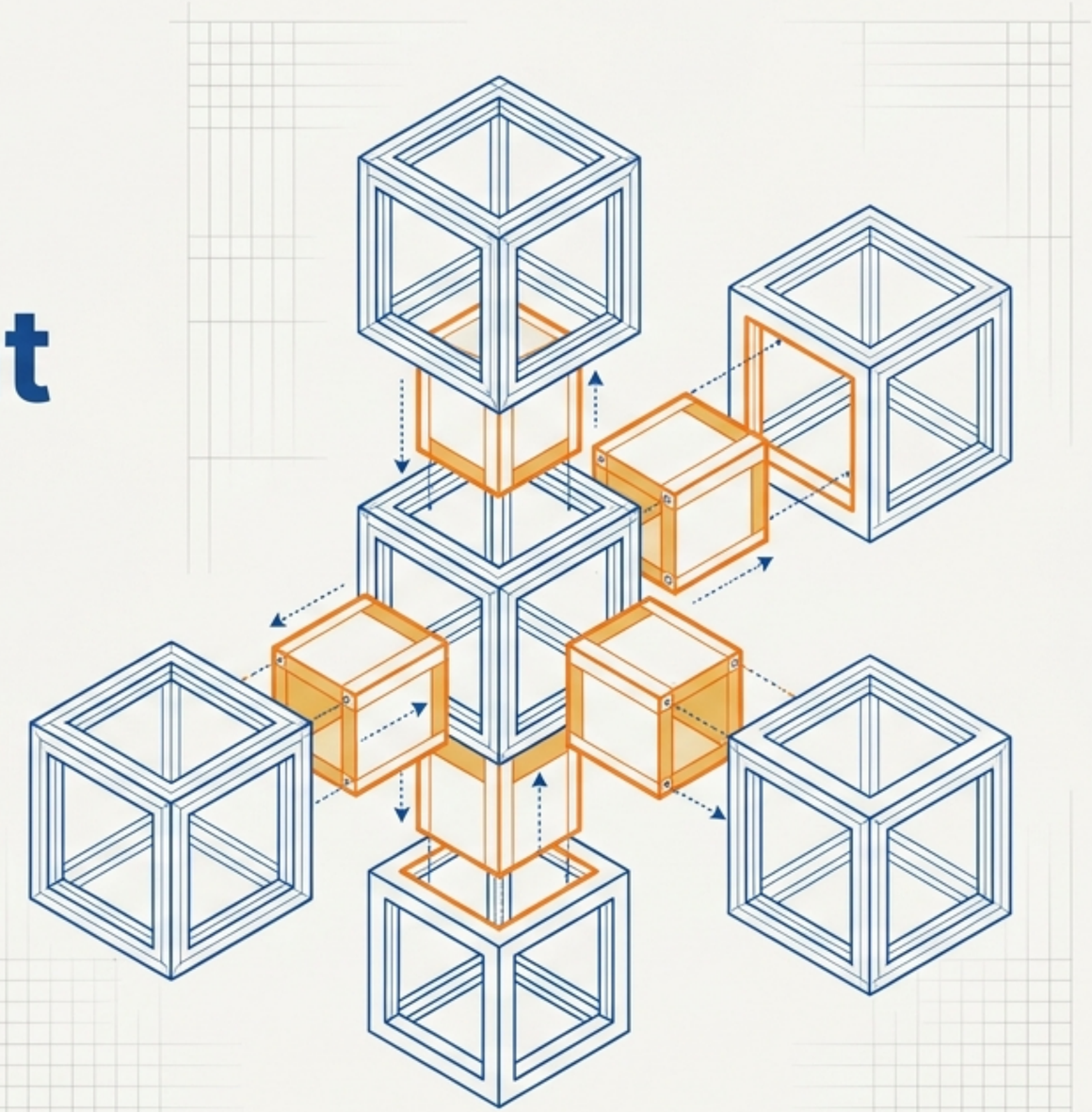
AppForge 6GL: The Final Blueprint

กรอบการทำงาน UX/UI & Agent-Native Foundation (ฉบับสังเคราะห์เพื่อการลงมือปฏิบัติ)



DATE: 23 ตุลาคม 2026
REVISION: 1.0 (Codex × Muse Code × Antigravity)
TARGET: 10-Month Agent-Led Operations

Gemini Notebook



เป้าหมายสูงสุด: พร้อมรองรับ Agent-led Operations แบบเต็มรูปแบบ ภายใน 23 สิงหาคม 2027 (กรอบการลงทุน 10 เดือน)

1
Re-baseline
ตรวจสอบฐานปัจจุบันบน Main/Deployed อย่างละเอียดก่อนสร้างใหม่

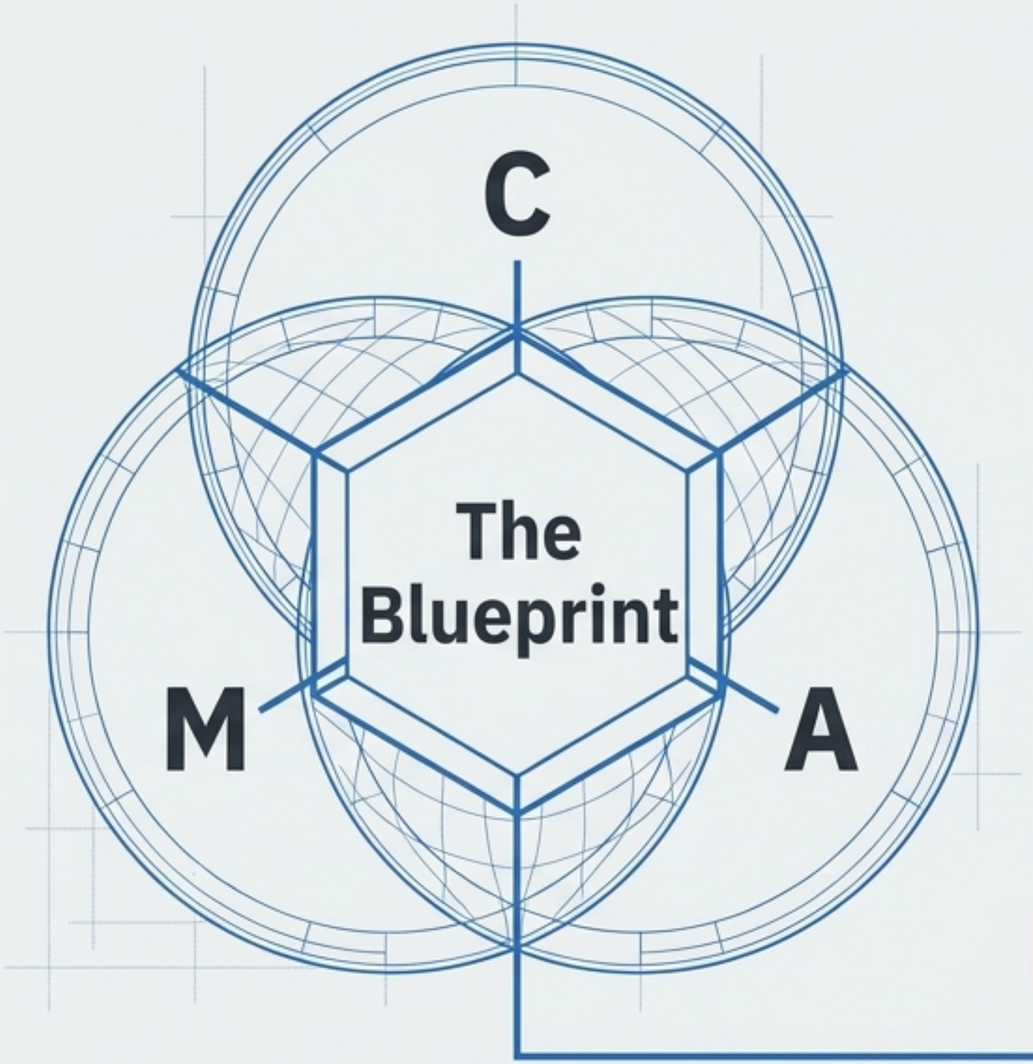
2
Reuse
นำโครงสร้างและ Binding เดิมที่ใช้งานได้มาใช้ซ้ำ (AppSpec)

3
Prove
พิสูจน์ความสำเร็จด้วย 1 Workflow ผ่าน Human, Jarvis และ Agent

4
Scale
ขยายผลไปยังส่วนอื่นๆ เมื่อมีหลักฐานการทำงานที่สมบูรณ์

จุดยืนสำคัญ: ระบบต้องทำงาน*ธุรกิจได้จริง*ทั้งระบบคนและ AI
การประหยัด Token เป็นเพียงผลพลอยได้ *ไม่ใช่เป้าหมายหลัก*ที่ต้องแลกกับคุณภาพ

Gemini Notebook



Codex: Quality & Contract

- กำหนดกรอบคุณภาพ Output และ Visual standards
- รักษา Intent ของ Requirement
- กลยุทธ์เส้นทาง Agent-native

Muse Code: Migration & Reuse

- เน้นวินัยในการ Re-baseline และรันระบบเดิม
- ใช้ Capability sidecar ลดการเขียนโค้ดซ้ำ
- ป้องกัน Untrusted content ตั้งแต่ฐานราก

Antigravity: Execution & Concurrency

- โครงสร้างแกนกลาง (Capability Substrate)
- บังคับใช้ Optimistic Concurrency Control
- สถาปัตยกรรม Tri-surface บนฐานเดียวกัน

บทสรุป: ใช้ C เป็นมาตรฐานคุณภาพ, M เป็นวินัยย้ายระบบ และ A เป็นบททดสอบความทนทานของ **Execution**

Gemini Notebook

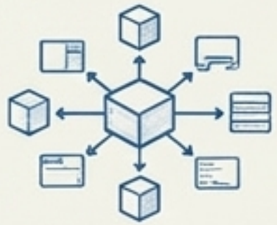
The 8 Core Directives

8 ข้อเสนอชี้ขาดเพื่อเริ่มงาน (The Constitution)



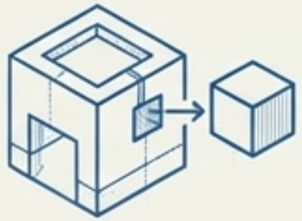
1. Check Before Build

ตรวจ Main และ Deployed build ก่อนสร้างของใหม่เสมอ



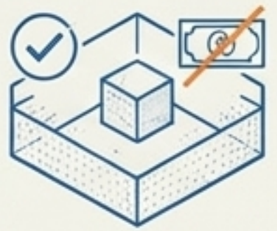
2. Single Source of Logic

Business logic ต้องอยู่ที่ Domain Services ชุดเดียว



3. Preserve AppSpec

รักษาโครงสร้างเดิม เพิ่ม Capability Sidecar เท่าที่จำเป็น



4. Pilot Scope

เริ่มจาก Approval Workflow ใน Sandbox (ไม่ผูกการจ่ายเงิน)



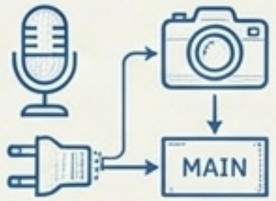
5. Visual Proof First

พิสูจน์ความสวยงามด้วย Output จริง ไม่ต้องรอ Backend เสร็จ



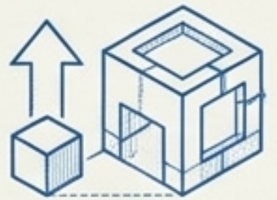
6. Security & Execution First

Authority ต้องมาก่อนเปิดให้ Agent ทำการแก้ไขข้อมูล



7. Adapters Later

Voice/Vision เป็น Input Adapters ทำทีหลังเมื่อระบบนิ่งแล้ว



8. Scale on Readiness

ขยายผลเมื่อมีหลักฐานความพร้อม ไม่บังคับสร้างทุกอย่างพร้อมกัน

Gemini Notebook

The Decision Register

รับ (Adopt)

- Re-baseline ถอยกลับไปประเมินระบบปัจจุบัน
- รวมศูนย์ Business Implementation & Projection Bindings
- สร้าง Untrusted-content controls ตั้งแต่แรก

ปรับ (Adapt)

- Capability Sidecar: อ้างอิง Domain ที่มีอยู่ (ไม่สร้าง Rules DSL ใหม่)
- Reuse patterns: ตรวจสอบ Topology ก่อนใช้ซ้ำ
- Legacy Compatibility: ใช้ฐานเดิมก่อนเพิ่ม Flag เวอร์ชั่นใหม่

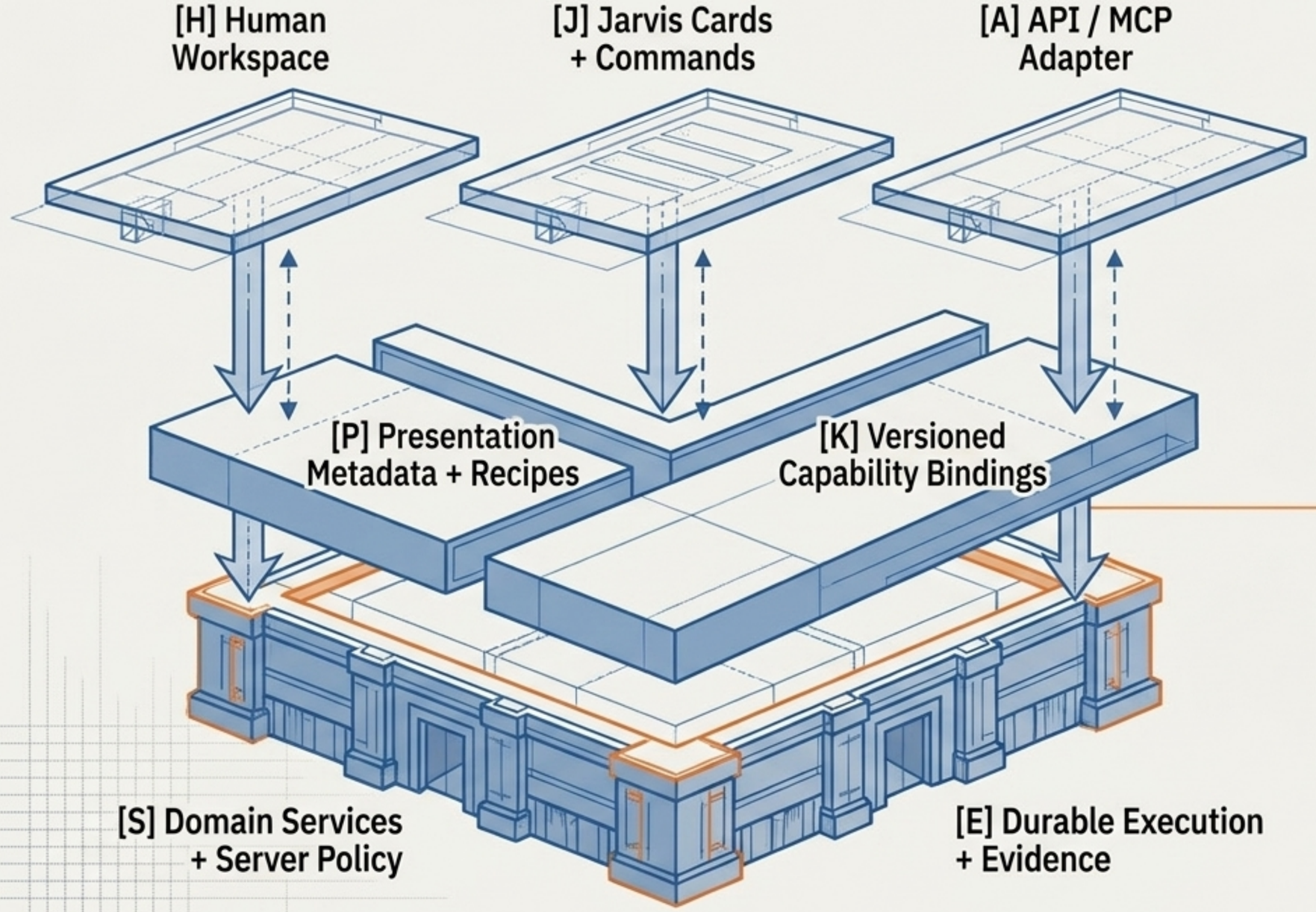
ชะลอ (Delay)

- Bespoke UI สำหรับทุก Entity (รอจนกว่า Task demand ชัดเจน)
- Voice/Vision features (รอจนกว่า Approval Contract ผ่าน)

ไม่รับ (Reject - Hard Stops)

- ❌ บังคับ Byte-equal ทุก PR (ต้องยอมให้แก้ Security ได้)
- ❌ ใช้ Redis แทน Ledger ถาวร
- ❌ ส่ง Raw Internal IDs ให้ End-user โดยไม่มีระบบ Auth

Gemini Notebook



Single Truth, Multiple Surfaces

ทุกเส้นทาง Mutation ไม่ว่าจะเป็นคนหรือ AI ต้องวิ่งผ่าน Authorization และ Execution Record ของ Server ที่ชั้นล่างสุดเสมอ

Gemini Notebook

Ownership Boundaries (เส้นแบ่งความรับผิดชอบ)

	Owns (รับผิดชอบ)	Must Not Become (ห้ามกลายเป็น)
Domain Services & Policy	กฎเกณฑ์ธุรกิจ, สิทธิ์, Invariants	ห้ามสร้างชุด Business Logic แยกต่างหากสำหรับ Agent
Capability Contract	Identity, IO refs, กฎ Concurrency	ห้ามเป็น Interpreter รัน Rules/SQL แบบอิสระ
Presentation Metadata	การจัดกลุ่ม, UI Roles, Visual states	ห้ามเก็บสำเนา Permission ที่ Server ไม่ได้ตรวจสอบ
Execution Record	หลักฐานถาวร (Evidence), การ Recovery	ห้ามทำตัวเป็นแค่ Response Cache ที่อ้าง exactly-once

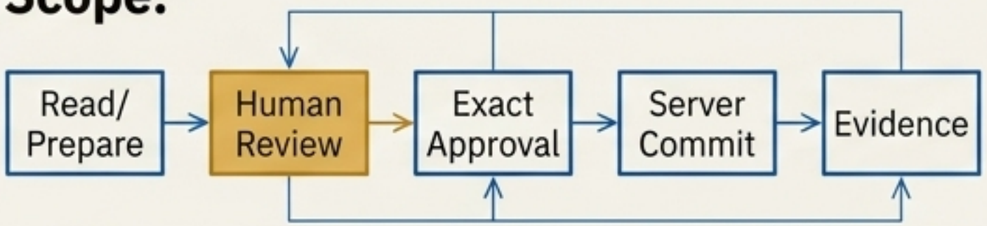
Gemini Notebook

The Proving Grounds

Pilot A: Approval Workflow (Agent/Hybrid Foundation)

Focus:
ตัดสินเคลมพร้อมบันทึกหลักฐาน (รันใน Sandbox)

Scope:



Pilot B: POS Product Editor (Visual Proof)

Focus:
สร้างมาตรฐาน UX ใหม่ (Grouping, Readable Labels, Money States)

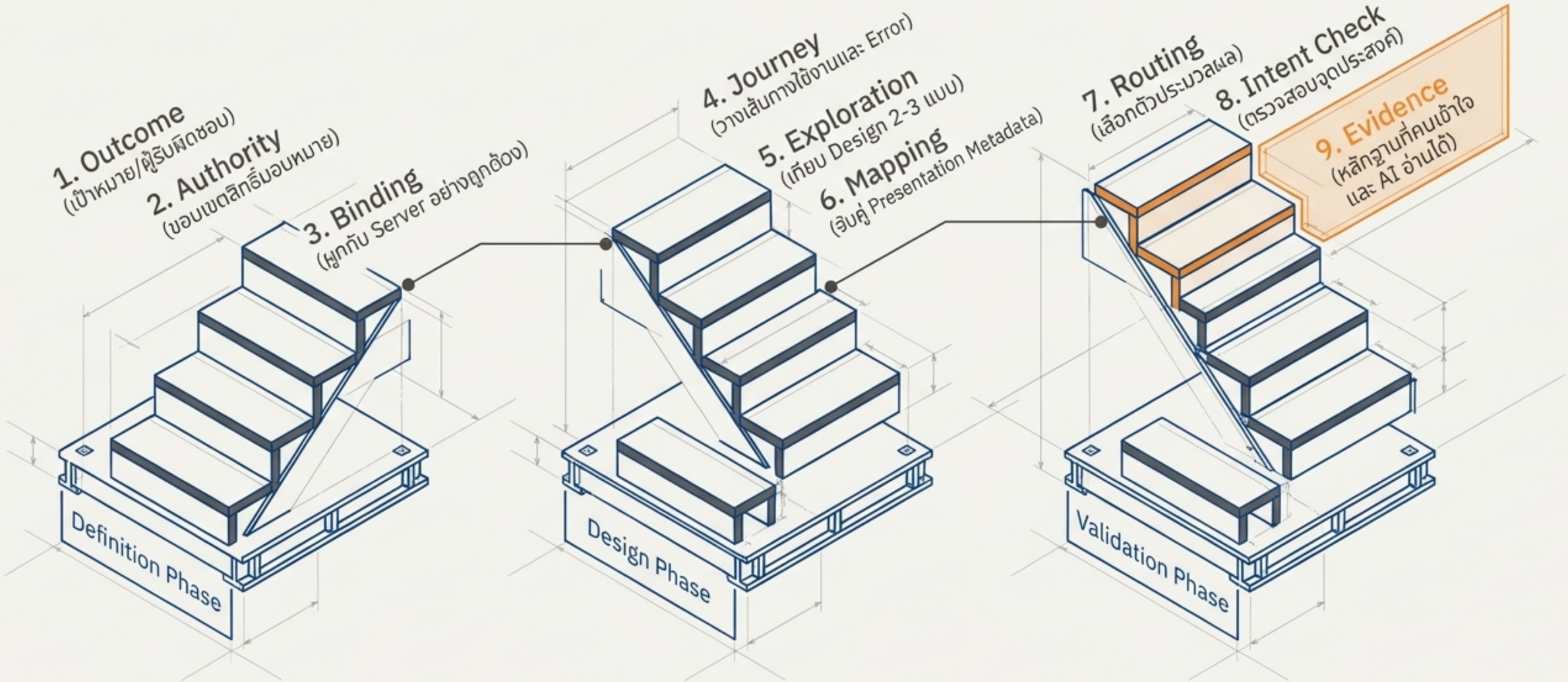
Scope:
สร้าง Before/After แบบ Standalone เพื่อพิสูจน์การทำงานของ AI

Next Phase (หลัง 90 วัน): Pilot C (Booking) - ขยายผลเพื่อทดสอบ Resource-level conflicts

EXCLUSION ZONE: ทั้งสอง Pilot **ห้ามยุ่งเกี่ยวกับ** Checkout, Payment Settlement, **หรือระบบภาษี** (Fiscal/Tax) **โดยเด็ดขาดจนกว่าจะสมบูรณ์**

Gemini Notebook

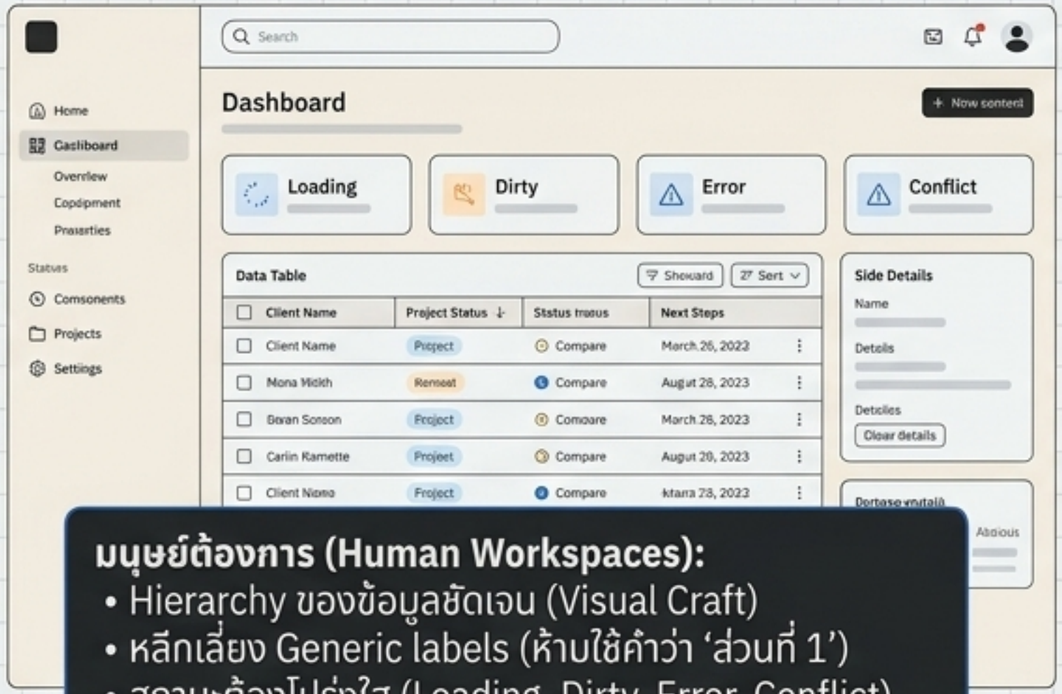
The 9-Step Journey for Every Capability



Gemini Notebook

Human vs. Agent: What Quality Means

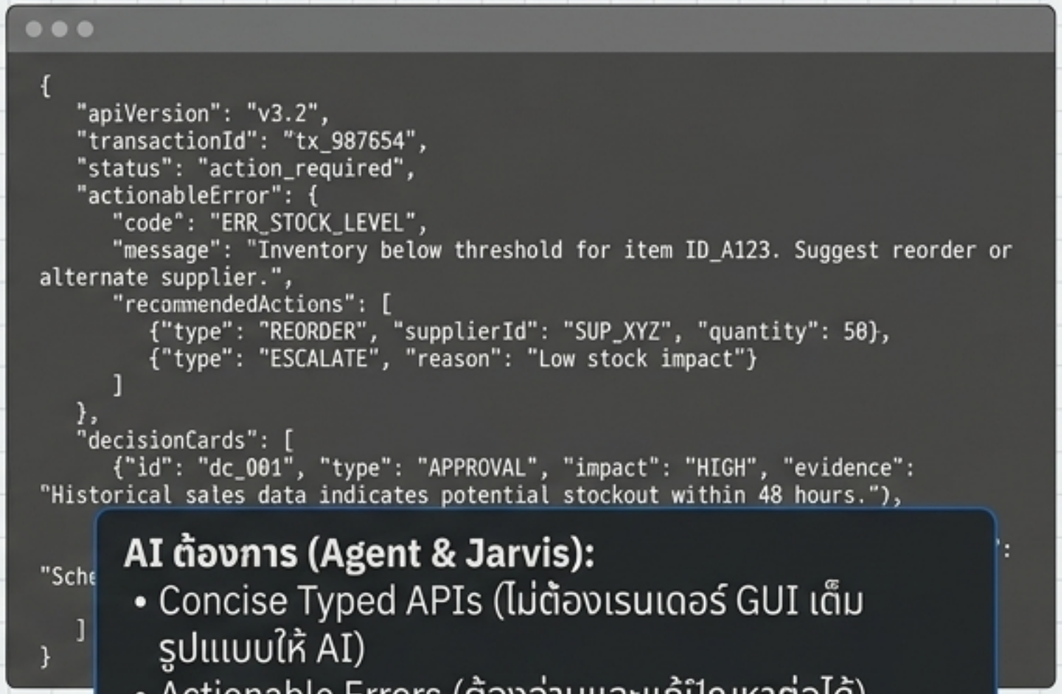
Human Workspace



มนุษย์ต้องการ (Human Workspaces):

- Hierarchy ของข้อมูลชัดเจน (Visual Craft)
- หลีกเลี่ยง Generic labels (ห้ามใช้คำว่า ‘ส่วนที่ 1’)
- สถานะต้องโปร่งใส (Loading, Dirty, Error, Conflict)
- Navigation นำทางง่าย หาทางกลับ/แก้ไขได้เสมอ

Agent / Jarvis



```
{
   "apiVersion": "v3.2",
   "transactionId": "tx_987654",
   "status": "action_required",
   "actionableError": {
      "code": "ERR_STOCK_LEVEL",
      "message": "Inventory below threshold for item ID_A123. Suggest reorder or
alternate supplier.",
      "recommendedActions": [
         {"type": "REORDER", "supplierId": "SUP_XYZ", "quantity": 50},
         {"type": "ESCALATE", "reason": "Low stock impact"}
      ]
   },
   "decisionCards": [
      {"id": "dc_001", "type": "APPROVAL", "impact": "HIGH", "evidence":
"Historical sales data indicates potential stockout within 48 hours."),
"Sche
   ]
}
```

AI ต้องการ (Agent & Jarvis):

- Concise Typed APIs (ไม่ต้องเรนเดอร์ GUI เต็มรูปแบบให้ AI)
- Actionable Errors (ต้องอ่านและแก้ปัญหาต่อได้)
- Decision Cards ต้องมี Impact และ Evidence ชัดเจน

Strategic Note: Machine-only ไม่ต้องมี GUI ครบทุกส่วน, สร้าง Human Surface เฉพาะจุดที่ต้องมีการอนุมัติหรือตรวจสอบ (Approval)

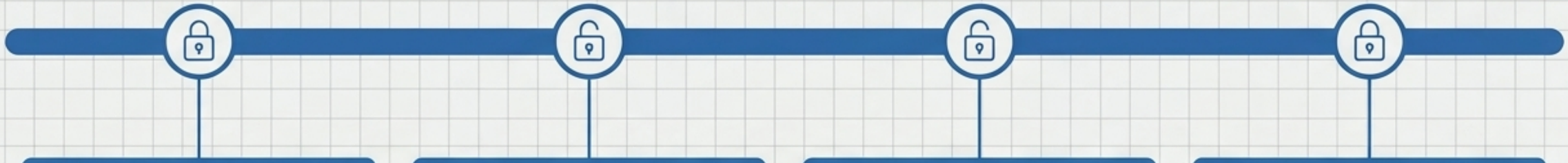
Gemini Notebook

The Action Backlog (FR00 - FR12)

Phase 0 (Foundation - P0)	Phase 1 (Proof - P1)	Phase 2 (Scale - P2)
ต้องทำก่อนเริ่มเขียนโค้ดบรรทัดแรก	สร้างเพื่อพิสูจน์ระบบ	ขยายผลหลังฐานรากมั่นคง
• FR00: ตรวจสอบ Baseline (Main/Deployed/Artifacts) • FR01: ล็อกขอบเขต Pilot • FR02: Audit เรื่อง Version และ Compatibility • FR05: กำหนด Capability Contract • FR06: วาง Security Boundaries • FR07: สร้าง Durable Execution	• FR03/04: UX/UI Metadata Slice • FR08: วางระบบ Jarvis Gateway • FR09: กำหนด Quality Gates และเกณฑ์ประเมิน • FR10: ทดสอบพิสูจน์ใน Domain ที่ 2 (Booking)	• FR11: ติดตั้ง Voice/Vision Adapters • FR12: ขยาย Capability Packs เต็มรูปแบบ

Gemini Notebook

The Critical Path (Exit Gates)



Gate 1: 22 พ.ย. 2026

- ผ่าน FR01-07 ขอบเขต Sandbox
- Agent ทำงานผ่าน API ได้ (ไม่ต้องใช้ Browser)
- เห็น Before-After ของ POS Editor

Gate 2: 22 ธ.ค. 2026

- ผ่าน FR08-09 ครบ Hybrid Pilot
- ระบบ Handoff / Takeover ทำงานสมบูรณ์
- ระบบ Recovery ทำงานเมื่อ Crash

Gate 3: 21 ม.ค. 2027

- ผ่าน FR10 (Cross-Domain Proof)
- ระบบจัดการ Conflict ข้าม Domain ผ่านเกณฑ์

Gate 4: 23 ส.ค. 2027

- รองรับ Multistep, Voice/Vision
- ขยาย Capability Packs พร้อมรัน Production จริง

Gemini Notebook

The Hard Blockers

คะแนนความสวยงาม ไม่สามารถชดเชยสิ่งเหล่านี้ได้ (ไม่อนุโลมให้ผ่านเด็ดขาด)



❌ Wrong-target / Unauthorized

ทำงานผิดเป้าหมาย หรือข้ามเส้นสิทธิ์การทำงาน



❌ Tenant Leak

ข้อมูลทะลุข้าม Tenant หรือทำข้อมูลสำคัญสูญหาย



❌ Duplicate Financial Effect

เกิด Side effect ซ้ำซ้อนทางการเงิน หรือย้อนกลับไม่ได้



❌ Misleading Success

ทำงานผิดพลาดแต่ Backend หรือ UI แจ้งว่า สำเร็จ



❌ Silent Downgrade

เปลี่ยนพฤติกรรมหลักของระบบเดิมโดยไม่แจ้งเตือนล่วงหน้า

Gemini Notebook

10 Critical Anti-Patterns

1. อย่ารวม Contract กับ Implementation ไว้ที่เดียวกัน

2. อย่าก๊อปปี้ Sample Schemas มาใช้บน Production ตรงๆ

3. อย่าใช้ Row Revision แทน Invariant ของธุรกิจ

4. อย่าใช้ Redis แทน Execution Ledger ถาวร

5. อย่าซ่อนสถานะ (States) จนความจริง/ปัญหาหายไป

6. อย่า Reuse Database Table ข้าม Product แบบมักง่าย

7. อย่ารีบ Merge Jarvis Gateway โดยไม่ดูระบบเดิมที่มีอยู่

8. อย่าแก้ Emitter ผิด Revision

9. อย่าคิดว่า Source เหมือนเดิมแล้ว Runtime จะเหมือนเดิม

10. อย่าวัด UX แก้สำเร็จจากจำนวนฟิลด์ ต้องเทสต์ใช้งานจริง

Gemini Notebook

Immediate Next Steps (สิ่งที่ทีมต้องทำวันนี้)

- [ ] **1. Execute FR00 (Baseline Matrix)**
  รัน Read-only Inventory ของระบบปัจจุบัน (Main/Deployed/Artifacts) ห้าม Checkout ทับงานค้างเด็ดขาด

- [ ] **2. Draft the Slice (FR01/03)**
  เสนอแผน Vertical Slice สำหรับ Approval Workflow และ POS Editor พร้อมระบุไฟล์, เจ้าของ, และแผน Rollback

- [ ] **3. Define Baseline ADR**
  Tech Lead ตัดสินใจขั้นสุดท้ายว่า Revision / Artifact ใดคือ Baseline เริ่มต้นที่แท้จริงของโครงการ

DISCLAIMER: การรับทราบเอกสารฉบับนี้ ไม่ได้แปลว่าอนุมัติให้ Deploy อัตโนมัติ งานแรกคือทำแผนตาม FR00 ส่งให้พิจารณาเริ่มทันที

Gemini Notebook