WHY ENGINEERS WHO WRITE PRODUCT SPECS CLOSE MORE CROSS-TEAM GAPS

THE SPEC NOBODY READS IS NOT THE PROBLEM. THE SPEC NOBODY WRITES IS.

A PRODUCT SPEC IS A DECISION, NOT A DOCUMENT

SHOULD WE BUILD THIS?

WHO IS IT FOR?

WHAT PROBLEM DOES IT SOLVE?

WHAT IS THE EXPECTED IMPACT?

VALUE

EFFORT

RISK

PRODUCT SPEC

- PROBLEM
- GOAL
- SOLUTION
- SUCCESS METRICS
- SCOPE
- CONSTRAINTS
- RISKS & ASSUMPTIONS

HOW WILL WE MEASURE SUCCESS?

WHAT ARE THE OPTIONS?

BUILD ✓

DON'T BUILD ✗

WHY THIS SOLUTION?

TRADE-OFFS?

ALTERNATIVES CONSIDERED?

CONFIDENCE LEVEL? ★★★★☆

A GOOD SPEC:

- ✓ FORCES CLARITY
- ✓ ALIGNS STAKEHOLDERS
- ✓ REDUCES WASTE
- ✓ ENABLES CONFIDENCE

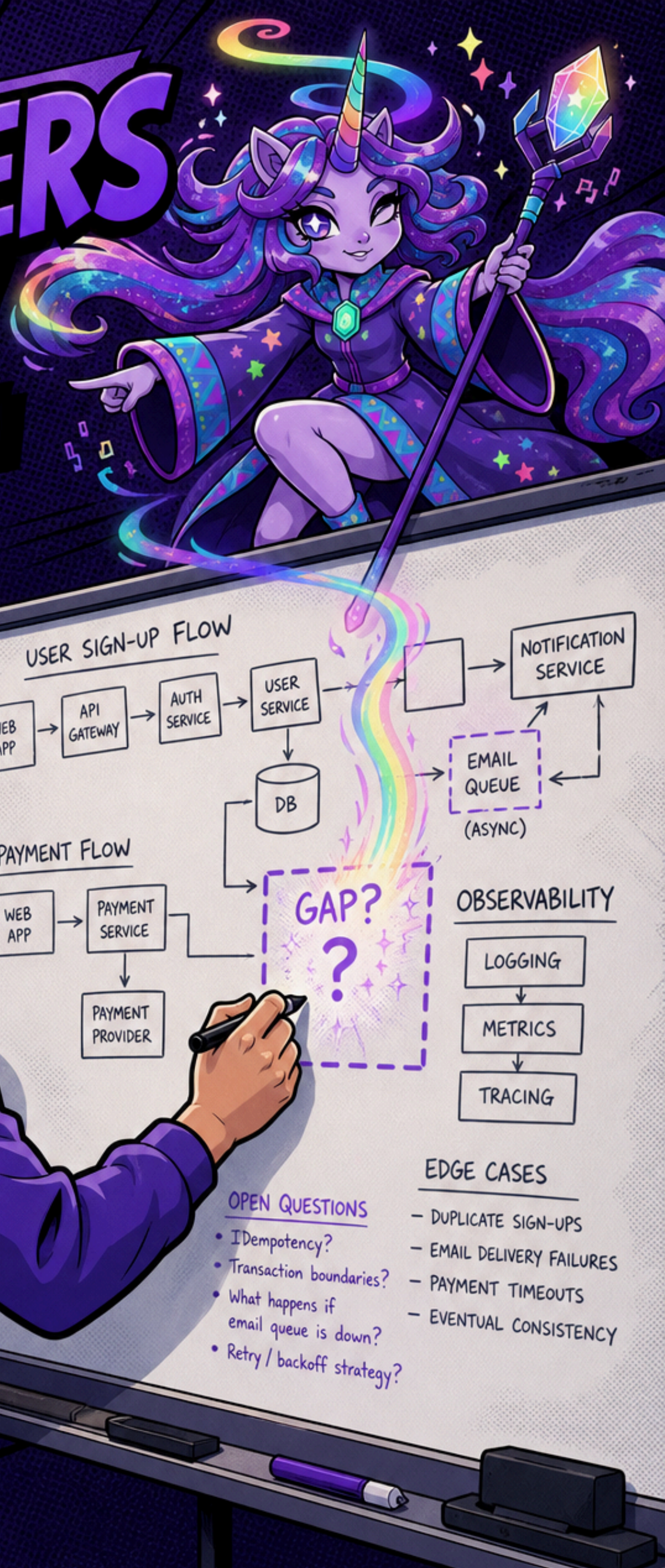
WHY ENGINEERS SPECIFICALLY SHOULD BE WRITING THEM

The traditional **sequential model,** where product defines requirements and hands them off to engineering, is increasingly challenged in fast-moving product environments. Modern systems tend to require deeper collaboration between product discovery and technical feasibility.

Engineers aren't passive consumers of decisions made upstream. They're the ones who know the **system best.** They know what the database can and can't do. They know which "simple" feature request requires rebuilding a core abstraction. They know where the **edge cases** live.

When an engineer engages with spec writing, they bring a perspective that no product manager, no matter how skilled, can fully replicate. **By writing a technical spec, engineers are forced to examine a problem before going straight into code, where they may overlook some aspect of the solution.** That same forcing function applies to product specs. The act of writing surfaces the gaps that a verbal conversation buries.

Having the engineer write a product requirements document ensures the features can be delivered and that they understand them. **That's not a nice-to-have. It's the difference between discovering a feasibility constraint in week one versus week eight.**



TECHNICAL DESIGN DOCS AND PRODUCT SPECS ARE NOT THE SAME THING

TECHNICAL DESIGN DOC

- ARCHITECTURE & SYSTEM DESIGN
- TECHNICAL APPROACH
- DATA MODELS & SCHEMAS
- APIS & INTEGRATIONS
- IMPLEMENTATION DETAILS

FOCUS: HOW
FOR ENGINEERS WHO BUILD

PRODUCT SPEC (PRD)

- USER PROBLEM & OPPORTUNITY
- FEATURES & SOLUTIONS
- GOALS & SUCCESS METRICS
- AUDIENCES & STAKEHOLDERS
- PRIORITIES & TRADE-OFFS

FOCUS: WHAT
FOR EVERYONE INVESTED IN OUTCOMES

ENGINEERING TEAM

- TECHNICAL FEASIBILITY
- SCALABILITY
- CODE QUALITY
- SYSTEM RELIABILITY

PRODUCT TEAM

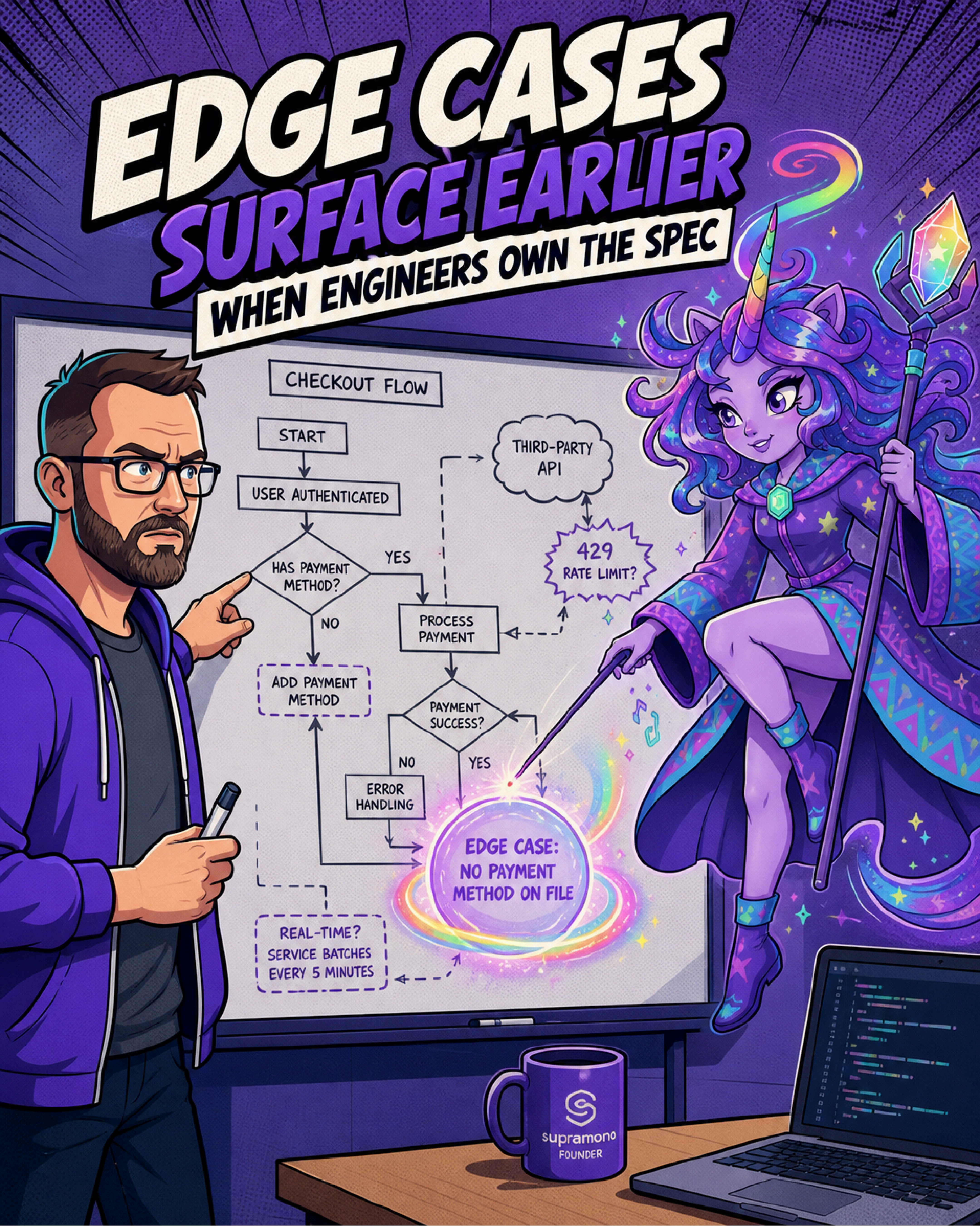
- USER NEEDS
- COMPETITIVE DIFFERENTIATION
- MARKET TRENDS
- PRODUCT EXPERIENCE

BUSINESS TEAM

- REVENUE IMPACT
- CUSTOMER EXPERIENCE
- OPERATIONAL EFFICIENCY
- STRATEGIC ALIGNMENT

UNDERSTANDING THIS DISTINCTION HELPS ENGINEERS WRITE OUTCOME-FIRST, NOT IMPLEMENTATION-FIRST.

THAT SHIFT CHANGES THE CONVERSATION.


EDGE CASES
SURFACE EARLIER
WHEN ENGINEERS OWN THE SPEC
CHECKOUT FLOW
START
USER AUTHENTICATED
HAS PAYMENT METHOD?
YES
NO
ADD PAYMENT METHOD
PROCESS PAYMENT
THIRD-PARTY API
429 RATE LIMIT?
PAYMENT SUCCESS?
NO
YES
ERROR HANDLING
EDGE CASE: NO PAYMENT METHOD ON FILE
REAL-TIME? SERVICE BATCHES EVERY 5 MINUTES
supramono
FOUNDER

WRITING IN THE LANGUAGE OF TRADEOFFS


The Discipline of Writing What we are not building
In Scope
✓ Core user authentication
✓ Project creation & management
✓ Task assignment & tracking
✓ Team collaboration & comments
✓ Real-time notifications
✓ Basic reporting & dashboards
Out of Scope
(NOT BUILDING)
✗ User-level frequency controls for notifications
✗ Advanced analytics & ML insights
✗ White-label / custom branding
✗ Offline mode
✗ Third-party marketplace integrations
✗ Custom workflows (beyond defaults)
✗ Multi-language (beyond English)

WHAT CHANGES WHEN ENGINEERS WRITE SPECS WELL

The practical effects are visible fairly quickly on teams where engineers take spec writing seriously.

MORE FEATURES
CHANGE SCOPE
USER FEEDBACK
ADD THIS
NEW PRIORITY

SPECIFICATION
1. PROBLEM
2. USER GOALS
3. FUNCTIONAL REQUIREMENTS
4. CONSTRAINTS & TRADEOFFS
5. NON-FUNCTIONAL REQUIREMENTS
6. OUT OF SCOPE
7. OPEN QUESTIONS

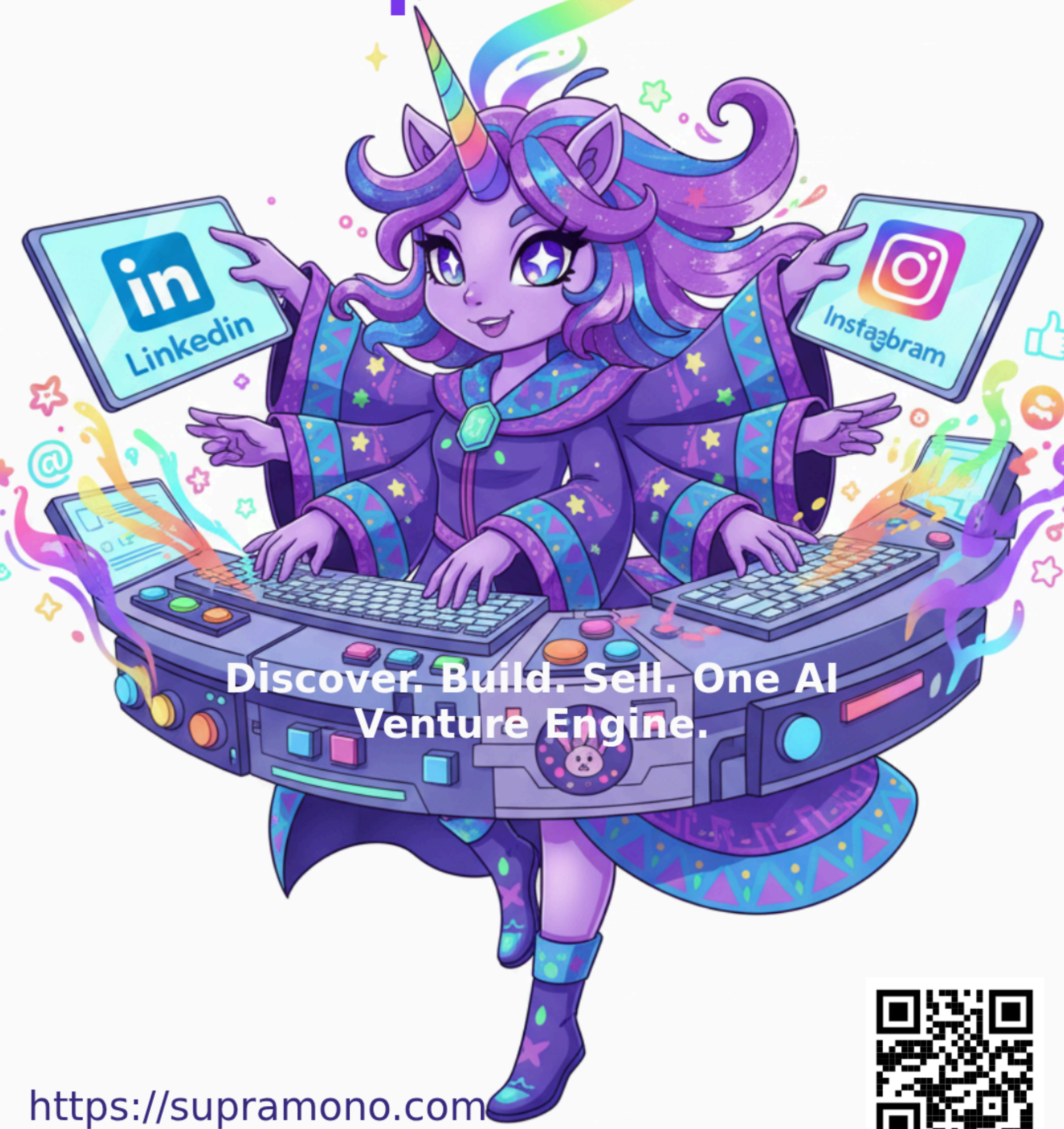
REFINE FLOW
TECH RISK
EDGE CASES
PERF IMPACT
DATA MODEL
SECURITY REVIEW

Where to Start

You don't need to write a perfect spec on the first try. **You need to write one.**


A NOTE TRADE-OFFS
STRENGTHS / UPSIDES
ENGINEERS BRING TECHNICAL DEPTH + EXECUTION CONTEXT
FASTER ALIGNMENT IN SMALL, SCRAPPY TEAMS
SHARPER QUESTIONS LEAD TO BETTER DECISIONS
COMPOUNDS SKILL + IMPACT ACROSS TEAMS
LIMITATIONS / TRADE-OFFS
SPEC WRITING TAKES TIME TIME THAT CAN CONFLICT WITH DELIVERY PRESSURE, ESPECIALLY ON SMALL TEAMS
ENGINEERS MAY LACK USER RESEARCH BACKGROUND → RISK OF UNDERWEIGHTING USER BEHAVIOR
ON TEAMS WITH DEDICATED PMS OR TPMS, A CLEAR DIVISION OF OWNERSHIP MAY BE MORE EFFICIENT
OVER-SPECIFICATION IS A FAILURE MODE: SLOWS TEAMS, CREATES FALSE CERTAINTY
STRONGEST IN CROSS-FUNCTIONAL, FAST-MOVING CONTEXTS — NOT UNIVERSAL ACROSS TEAM STRUCTURES OR DOMAINS

supramono



Discover. Build. Sell. One AI Venture Engine.

https://supramono.com