

CA CatalystAgile

# The AI-Assisted Execution Guide

From Problem to Product

Ignatious P. · Product Strategy & Execution Lead · MBA | CSM | CSPO · Fintech delivery

ignatious@catalystagile.com · catalystagile.com

**I direct AI-assisted execution with product-grade acceptance criteria, testing, and delivery judgment.**

A practical step-by-step process for building useful BA/PO tools with AI assistance — proven across 50 shipped tools and enterprise delivery work.

**50**

Free Tools

**\$50M+**

Shipped

**15**

Case Studies

**5**

Step Process

v3.0 · April 2026 · catalystagile.com

# 00 What's Inside

A compact operating guide for turning BA/PO judgment into useful, testable tools.

<b>01</b>	<b>The Process</b>	Five steps from problem to product
<b>02</b>	<b>Problem Identification</b>	Get the problem right before building
<b>03</b>	<b>How to Describe What You Want</b>	The prompt formula
<b>04</b>	<b>Testing Checklist</b>	Observation-driven validation
<b>05</b>	<b>Ship / Kill Criteria</b>	Decision framework
<b>06</b>	<b>15 Real Examples</b>	Shipped and killed projects
<b>07</b>	<b>Common Pitfalls</b>	8 mistakes to avoid
<b>08</b>	<b>Tools &amp; Costs</b>	The AI-assisted execution stack
<b>09</b>	<b>Your Turn</b>	The one-week experiment

All tools referenced in this guide are free and available at [catalystagile.com/tools.php](https://catalystagile.com/tools.php).

# 01 The Process

Five steps from problem to product: repeatable, testable, honest.

<b>Identify the Problem</b>	Not "we need a feature." A real pain point. If you cannot explain it in one sentence, you do not understand it yet.
<b>Define Success</b>	Make success specific and testable. What would you recognize immediately as solved?
<b>Prototype with AI Assistance</b>	Describe inputs, logic, outputs, edge cases, and acceptance criteria. Iterate against the result.
<b>Test with Real Users</b>	Show it to 3-5 colleagues. Do not explain it. Watch where they hesitate.
<b>Ship or Kill</b>	Attach to outcomes, not ideas. If it works, share it. If it fails, learn quickly and move on.

## Operating principle

AI does not replace product judgment. It removes delay between clear thinking and a working prototype.

## 02 Problem Identification

The most important step. Get this wrong and everything else is wasted effort.

Can you explain the problem to a colleague in one sentence? If not, stop. You do not understand it yet.

<b>Who has this problem?</b>	Specificity creates relevance.
<b>How often does it happen?</b>	Frequency drives value.
<b>What do they do today?</b>	The workaround exposes the real gap.
<b>What does solved look like?</b>	Success must be observable.
<b>Would you use this yourself?</b>	A direct gut check on real value.

### Problem statement template

[Role] spends [time/effort] on [task] because [root cause]. A tool that [solution] would save [measurable outcome].

## 03 How to Describe What You Want

You do not need to start with code. Start with a clear product brief.

<b>1. What it does</b>	Plain-language purpose of the tool.
<b>2. Inputs</b>	Fields, choices, numbers, constraints, and defaults.
<b>3. Processing</b>	Rules, calculations, scoring, categorization, and edge cases.
<b>4. Outputs</b>	Text, table, chart, checklist, export, or recommendation.
<b>5. Format</b>	How it should look and behave on desktop and mobile.

### Example prompt

Build a tool where users enter stakeholders, rate power and interest, then generate a Mendelow grid, communication cadence, escalation path, and PDF-ready output. Use clear mobile layout, validation, and empty-state handling.

# 04 Testing Checklist

Show it to 3-5 colleagues. Do not explain it. Just watch.

- Do they understand the purpose without explanation?
- Can they complete the primary task without getting stuck?
- Does the output match what they expected?
- Where do they hesitate?
- What do they try to click that is not clickable?
- Do they try to export or share results?

<b>Mobile</b>	Works on phone first.
<b>Empty input</b>	No crashes; clear errors.
<b>Export</b>	PDF, CSV, copy, or email as needed.
<b>Speed</b>	Results under 2 seconds.
<b>Accessible</b>	Keyboard and contrast checked.

# 05 Ship / Kill Criteria

No attachment to ideas. Attachment to outcomes.

<b>SHIP IF</b>	Solves the core problem; people use it; feedback is specific; maintenance is manageable; you would use it yourself.
<b>KILL IF</b>	Does not solve the core pain; too complex for the value; better tools already exist; requires maintenance you cannot support; users need heavy explanation.

Killed examples: Backlog Health Analyzer, Automated Meeting Scheduler, Requirements Quality Scorer. Each failed for a different product reason. Fast prototyping made the learning affordable.

# 06 15 Real Examples from My Work

12 shipped, 3 killed. Useful proof beats abstract claims.

1	<b>Mission Control PWA</b>	Cognitive support for daily tasks	40% task completion improvement; 9 weeks vs 6-month estimate.
2	<b>User Story Slicer</b>	Story splitting debates	Free community tool.
3	<b>Stakeholder Map Generator</b>	50+ stakeholder program	Used on \$50M+ project with 73 stakeholders.
4	<b>CatalystAgile.com</b>	Portfolio needed proof, not claims	16+ pages, 50 tools, Mission Control mockups, analytics-ready funnel.
5	<b>RICE Prioritization Calculator</b>	Opinion-led prioritization	Compare up to 10 features with data-driven scoring.
6	<b>Acceptance Criteria Generator</b>	Vague requirements	Turns ambiguity into Given-When-Then criteria.
7	<b>Diplomatic Email Rewriter</b>	High-friction stakeholder emails	Multiple tone options and faster drafting.
8	<b>Sprint Velocity Calculator</b>	No easy velocity trend view	Forecasting and trend visibility.
9	<b>Definition of Done Checker</b>	Inconsistent quality standards	Shareable DoD checklists.
10	<b>Risk Radar</b>	Static risk documents	Dynamic risk visualization.
11	<b>Kano Model Prioritizer</b>	Confusing must-have and delight features	Data-driven classification.
12	<b>BA/PO Skills Assessment</b>	No structured capability gap view	Radar chart with recommendations.

Killed: Backlog Health Analyzer, Automated Meeting Scheduler, Requirements Quality Scorer. Fast failure protected time and focus.

# 07 Common Pitfalls

8 mistakes I see constantly and how to avoid them.

- Building before understanding the problem.
- Trying to build enterprise software as a first version.
- Not testing with real users.
- Over-engineering before validation.
- Getting attached to the idea instead of the outcome.
- Writing vague prompts.
- Skipping the kill decision.
- Forgetting that 50 tools means 50 tools to maintain.

## **Maintenance rule**

Static client-side tools with no database, no API key dependency, and no signup are intentionally low-maintenance. That is part of the product strategy.

## 08 Tools & Costs

The AI-assisted execution stack is modest; the value comes from judgment.

<b>Perplexity Pro</b>	<b>\$20/mo</b>	Research, framework discovery, fact-checking.
<b>Gemini AI Pro</b>	<b>\$20/mo</b>	Alternative reasoning and prototyping.
<b>Claude Code</b>	<b>\$20/mo</b>	Primary implementation assistant.
<b>Claude CLI</b>	<b>Free</b>	Command-line iteration.
<b>Notion</b>	<b>Free tier</b>	Task tracking and documentation.

### Division of labor

AI assists with implementation drafts and changes. I direct the workflow, acceptance criteria, prioritization, testing, and product decisions.

# 09 Your Turn

The one-week experiment: 4 focused hours spread across 5 days.

<b>Day 1</b>	Identify one repetitive BA/PO problem and write it in one sentence.
<b>Day 2</b>	Build a small prototype with clear inputs, outputs, and processing rules.
<b>Day 3</b>	Test with one colleague. Watch behavior; do not defend the design.
<b>Day 4</b>	Fix the biggest confusion point. One fix only.
<b>Day 5</b>	Ship or kill. Either outcome creates learning.

**I am not a traditional software engineer. I am a product and business analysis leader who turns workflows, decisions, acceptance criteria, and validation loops into usable prototypes.**

You need to know what problem to solve, describe it clearly, test it honestly, and make strategic decisions.

READY TO START?

# Try one problem this week.

One problem. One tool. Five days.

**I direct AI-assisted execution with product-grade acceptance criteria, testing, and delivery judgment.**

<b>Try the 50 free tools</b>	<b><a href="https://catalystagile.com/tools.php">catalystagile.com/tools.php</a></b>	All tools are free. No signup. No data collection.
<b>See the complete build process</b>	<b><a href="https://catalystagile.com/portfolio.php">catalystagile.com/portfolio.php</a></b>	How CatalystAgile was built, step by step.
<b>Connect directly</b>	<b><a href="mailto:ignatious@catalystagile.com">ignatious@catalystagile.com</a></b>	Questions, feedback, or BA/PO strategy conversations.

**50**  
Free Tools

**\$50M+**  
Projects Shipped

**15**  
Case Studies

**~\$60/mo**  
AI Stack Cost