

Revolution!

Speed Work for the AI

Era

Break the Old Rules:

Build First, Learn Faster, and Get Real Work
Moving

Daisuke Yasuda

Revolution! Speed Work for the AI Era

Copyright © 2026 Daisuke Yasuda

All rights reserved.

No part of this book may be reproduced or distributed without permission, except as allowed by law.

Contents

Introduction

Chapter 1: The Old Rules of Work Have Broken Down

Chapter 2: Build Before You Write the Proposal

Chapter 3: Get customers to say “no” quickly

Chapter 4: Align the Timelines of Managers and Developers

Chapter 5: Build First, Then Document

Chapter 6: A demo is your strongest sales tool

Chapter 7: Move from Market Research to In-Market Testing

Chapter 8: Make mistakes while they're cheap

Chapter 9: AI Is a Tool, Not the Player

Chapter 10: Knowledge Alone Is No Longer Enough

Chapter 11: Engineers Should Bring Three Options

Chapter 12: Developing team members with AI

Chapter 13: Treat Unpaid Development as a Business Investment

Chapter 14: Increase revenue engines instead of people

Chapter 15: Redesign the Business, Not Just the AI Layer

Final Chapter: Before You Overthink It, Build One Thing

Summary: 15 principles for taking this book home to your practice

Conclusion

Introduction

AI Alone Does Not Make the Whole Job Faster



OUTPUT

Documents, plans, and screens multiply

MEETINGS

More to read and explain

DECISIONS

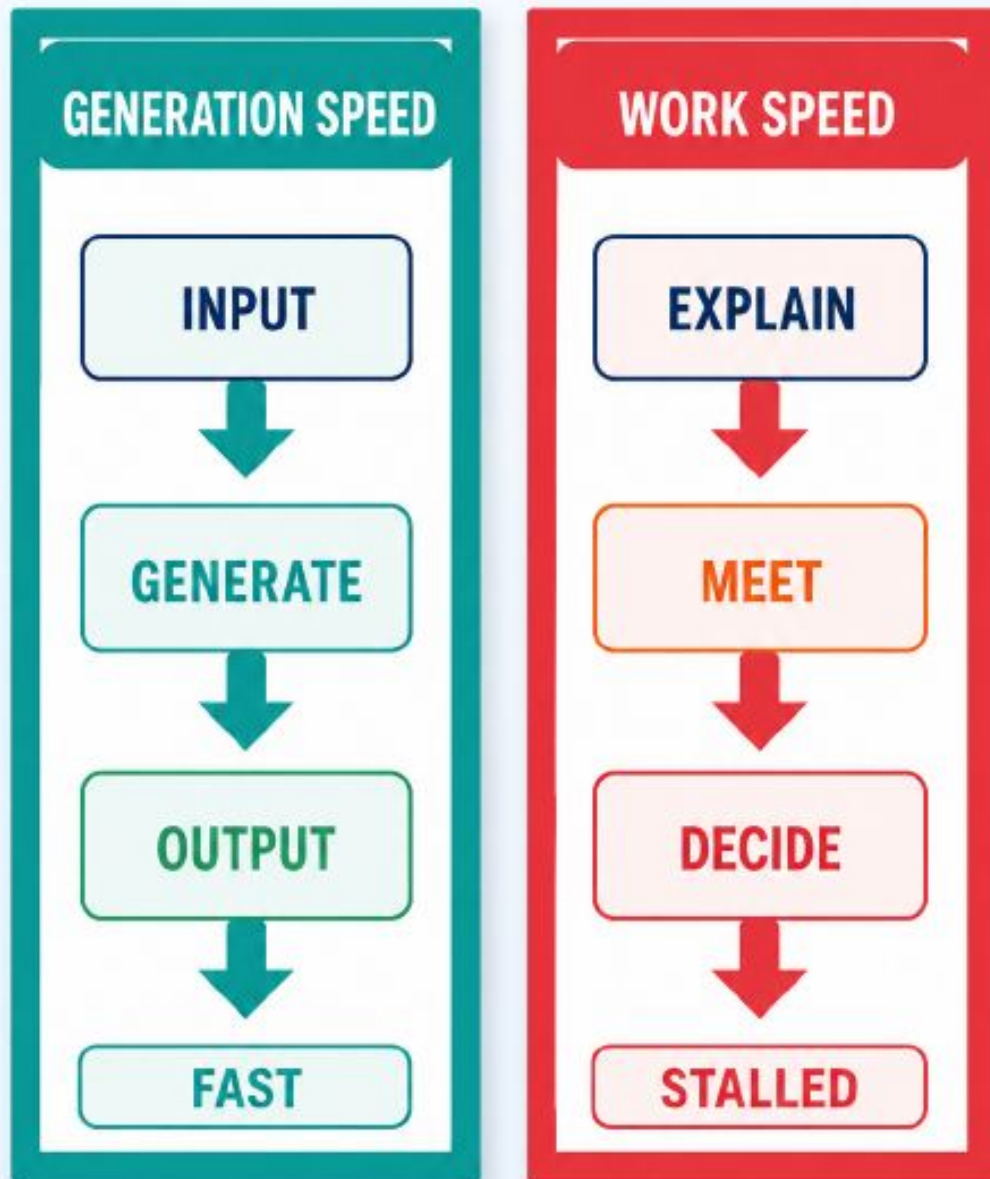
Decision-makers have no more time

REWORK

The same process keeps the backlog alive

Change the order of work, not just the tools

Generation Speed Is Not Work Speed

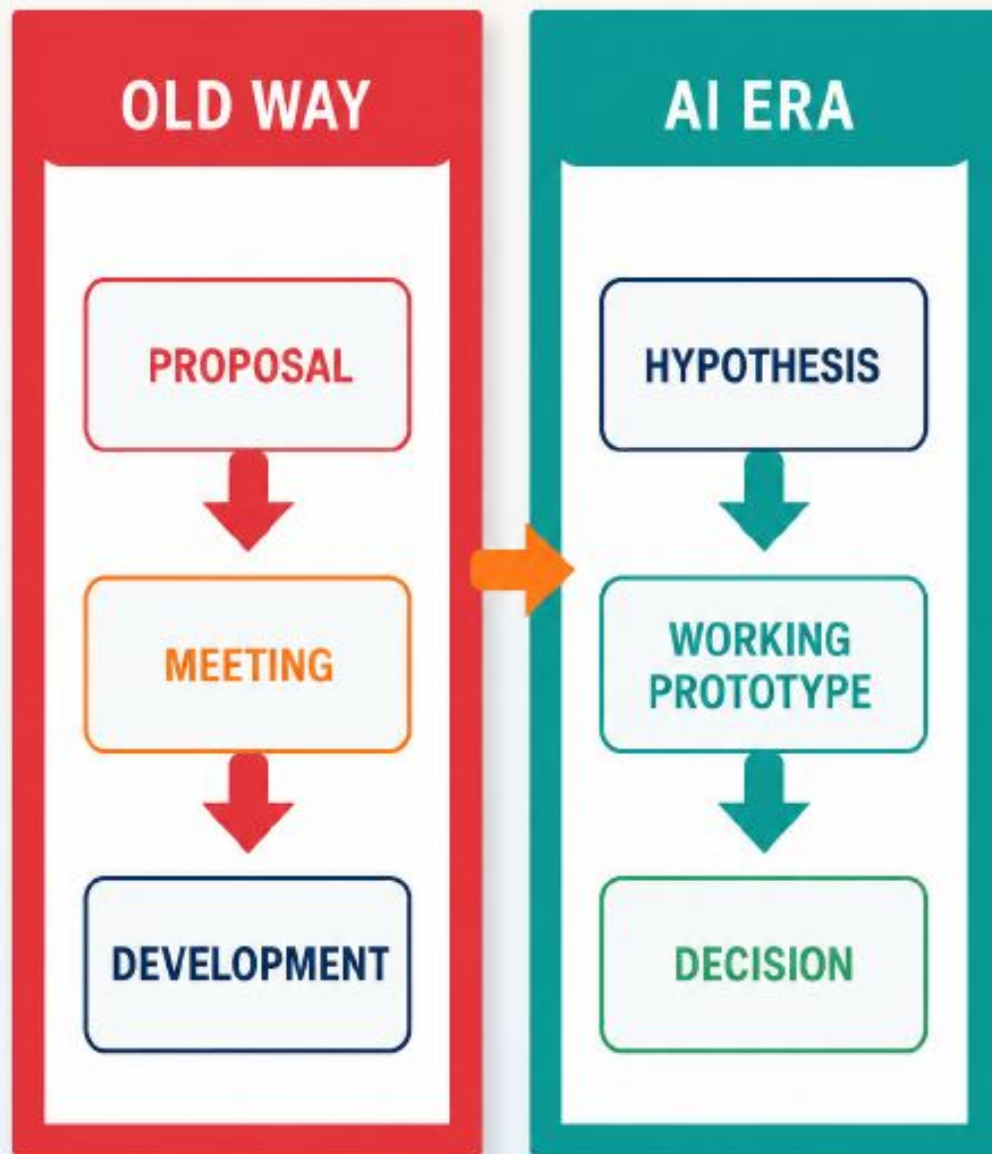


Faster output alone does not accelerate decisions

I misunderstood this for almost a year

Since I started using AI, I've created a variety of things.

Reverse the Order of Work

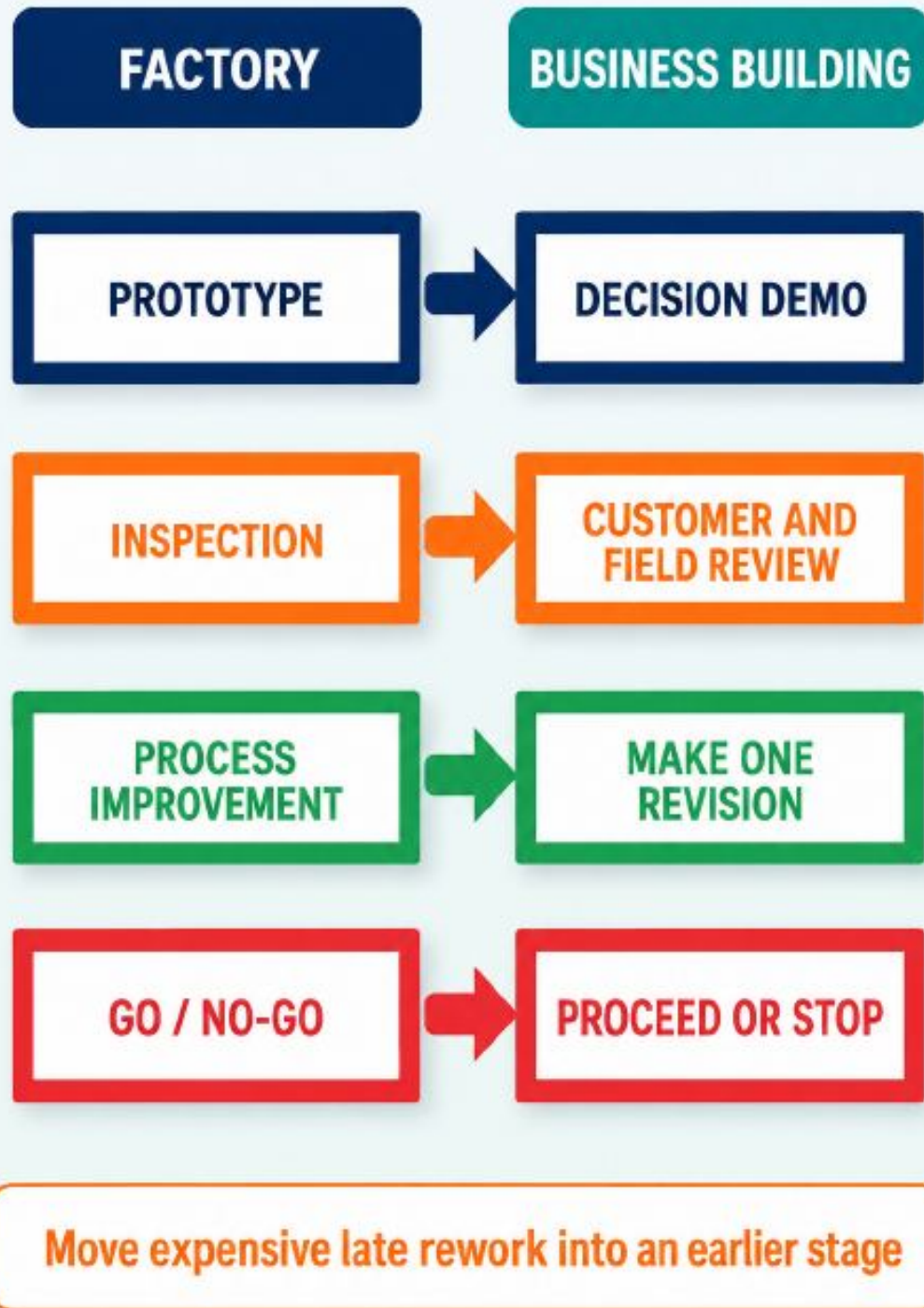


Build first, then decide—instead of explaining first

Build a decision-ready artifact before the proposal

There is only one thing I want to convey most in this book.

Bring Factory Process Improvement into Business Building



The origin is in the factory

My origins are not in the AI industry.

One Cycle for Using This Book



First, create something that can be judged.

You don't need to have memorized a list of AI tools by the time you finish reading this book.

Chapter 1: The Old Rules of Work Have Broken Down



High Prototyping Costs Supported the Old Rules

Prototyping incurs major costs

People



Specialized knowledge is required

- Designers
- Analysis specialists
- Evaluation specialists
- And more

Time



Engineering takes a lot of time

- Moves step by step
- Waiting time occurs
- Many reworks

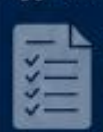
Equipment



Large equipment and tooling are necessary

- Machining equipment
- Measuring equipment
- Dedicated jigs and fixtures
- Dedicated tools
- Specialized software
- And more

Approval



The approval process takes a long time

- Prepare application
- Confirm with stakeholders
- Build up approvals

Think Ahead

Because prototyping is expensive, thoroughly review before you build was the right thing to do.



Thorough upfront planning

- ✓ Organize requirements
- ✓ Define specifications
- ✓ Review structure
- ✓ Analyze risks
- ✓ Plan validation
- ✓ Confirm feasibility



Refine the plan



- Reduce rework
- Gain stakeholder alignment
- Build a plan that truly moves forward

Conclusion

Because high prototyping costs demanded long, careful deliberation, it became sound practice to **“Think Ahead.”**

Common sense comes from cost.

How you proceed with your work is not determined solely by your thoughts.

There was a reason for this approach.

They could not afford to repeatedly destroy what they had created.

Three Prototyping Costs Reduced by AI



Three Hours on Screen Beat Three Weeks of Meetings

THREE WEEKS OF MEETINGS



VERBAL AGREEMENT



THREE HOURS WITH THE SCREEN



REVIEW THE REAL THING

THIS IS DIFFERENT

DECISION

Find the Difference Instead of Inventing the Right Answer

THINKING FROM A BLANK PAGE



REVIEW THE REAL THING



NEXT REVISION



Choose the Hypothesis Before the Specification

Before building, test the axis. Start small and move forward with confidence.



The first thing to decide is the hypothesis you want to test, not the specifications.

So, do we need to think nothing before making it?

AI Changed the Order of Deliberation

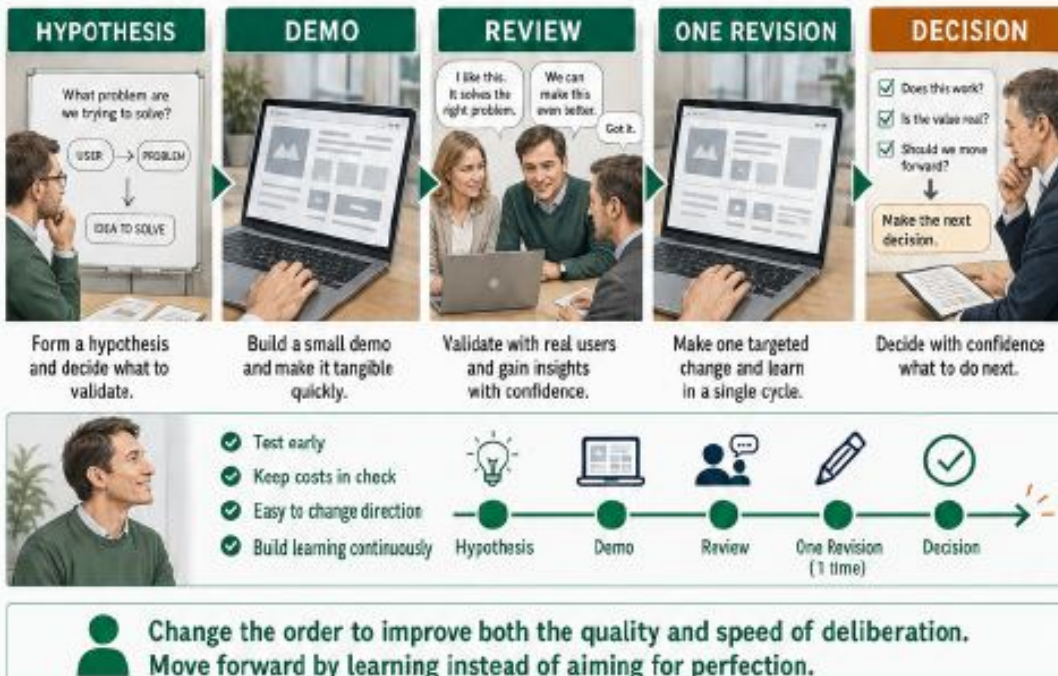
OLD WAY

Invest heavily up front, validate late.



AI ERA

Start small to learn fast. Iterate forward with clarity.



What AI broke was the order of consideration.

AI does not completely automate planning, sales, and development.

Chapter 2: Build Before You Write the Proposal

Build the Real Thing Before the Proposal



Three Illusions Created by a Proposal



Documents Do Not Lie, but People Read Too Much into Them

It's not that the proposal itself is bad.

The Boundary Between a Production Product and a Decision Demo

PRODUCTION PRODUCT — BUILT TO OPERATE

Built to Run



Built for continuous use with a focus on data, security, and quality.



SECURITY

- Authentication & authorization
- Data encryption in transit
- Audit logging
- Vulnerability management



ACCESS CONTROL

- Role-based access control (RBAC)
- Principle of least privilege
- Approval workflows



MAINTENANCE

- Ongoing improvements & feature expansion
- Monitoring & alerting
- Backups & recovery



TESTING

- Test planning & test cases
- System, integration, and acceptance testing
- Automated tests & regression testing



PRODUCTION DATA

Real customer data processed and stored securely for reliable operations.

DECISION DEMO — BUILT TO TEST DIRECTION

Built to Test Direction



Built to validate direction quickly with the minimal screen and sample data.



SMALL BUILD

- Minimal screens & functionality
- Sample data is sufficient
- Test quickly in a short time



SCOPE IS LIMITED

- Define scope boundaries
- Detailed work isn't necessary
- Easy to change



DISPOSABLE

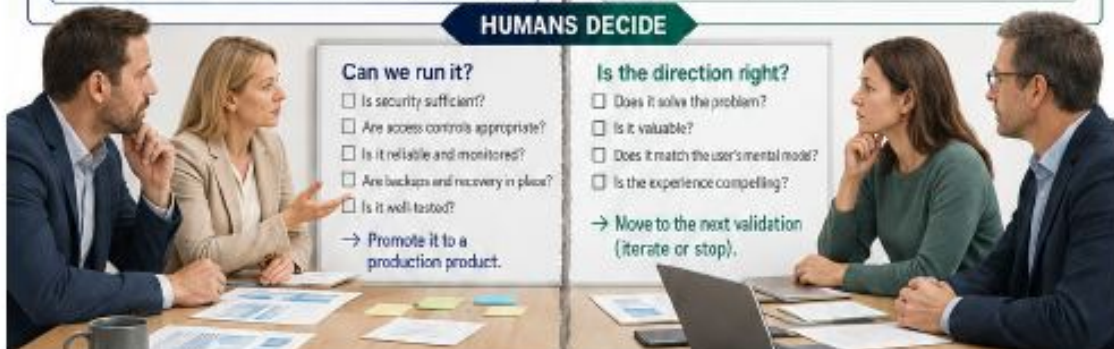
- Build it to explore and discard
- Easy to throw away afterward
- No need to preserve assets



SAMPLE DATA

Use fabricated or anonymized data to reduce preparation effort and risk.

HUMANS DECIDE



Can we run it?

- ☐ Is security sufficient?
- ☐ Are access controls appropriate?
- ☐ Is it reliable and monitored?
- ☐ Are backups and recovery in place?
- ☐ Is it well-tested?

→ Promote it to a production product.

Is the direction right?

- ☐ Does it solve the problem?
- ☐ Is it valuable?
- ☐ Does it match the user's mental model?
- ☐ Is the experience compelling?

→ Move to the next validation (iterate or stop).

Abstract Questions Become Concrete Questions



Building First Improves the Questions

In meetings without a working artifact, questions become abstract.

Three Questions That Define the Minimum Deliverable



How do you decide on the smallest working artifact?

When you don't know what to make, use three questions.

How to Build a One-Day Demo

MORNING

CHOOSE
THE
HYPOTHESIS



NOON

BUILD
SMALL



Build the smallest possible version that lets us test.

AFTERNOON

SHOW AND
REVISE



Highlight
Status colors are hard to read.

Fix
Adjust the color tones and make labels clearer.



FINAL

DECIDE
WHETHER TO
PROCEED



How to make a one-day demo

If you want to create a demo in one day, don't start writing code in the morning.

A Proposal Built After the Real Thing Is Stronger



Success cannot be promised. Move forward from facts.

Chapter 3: Get customers to say “no” quickly

≡ An Early ‘No’ Reduces Rework ≡



Fear of customer rejection slows down confirmation.

Separate Praise from Decision Evidence

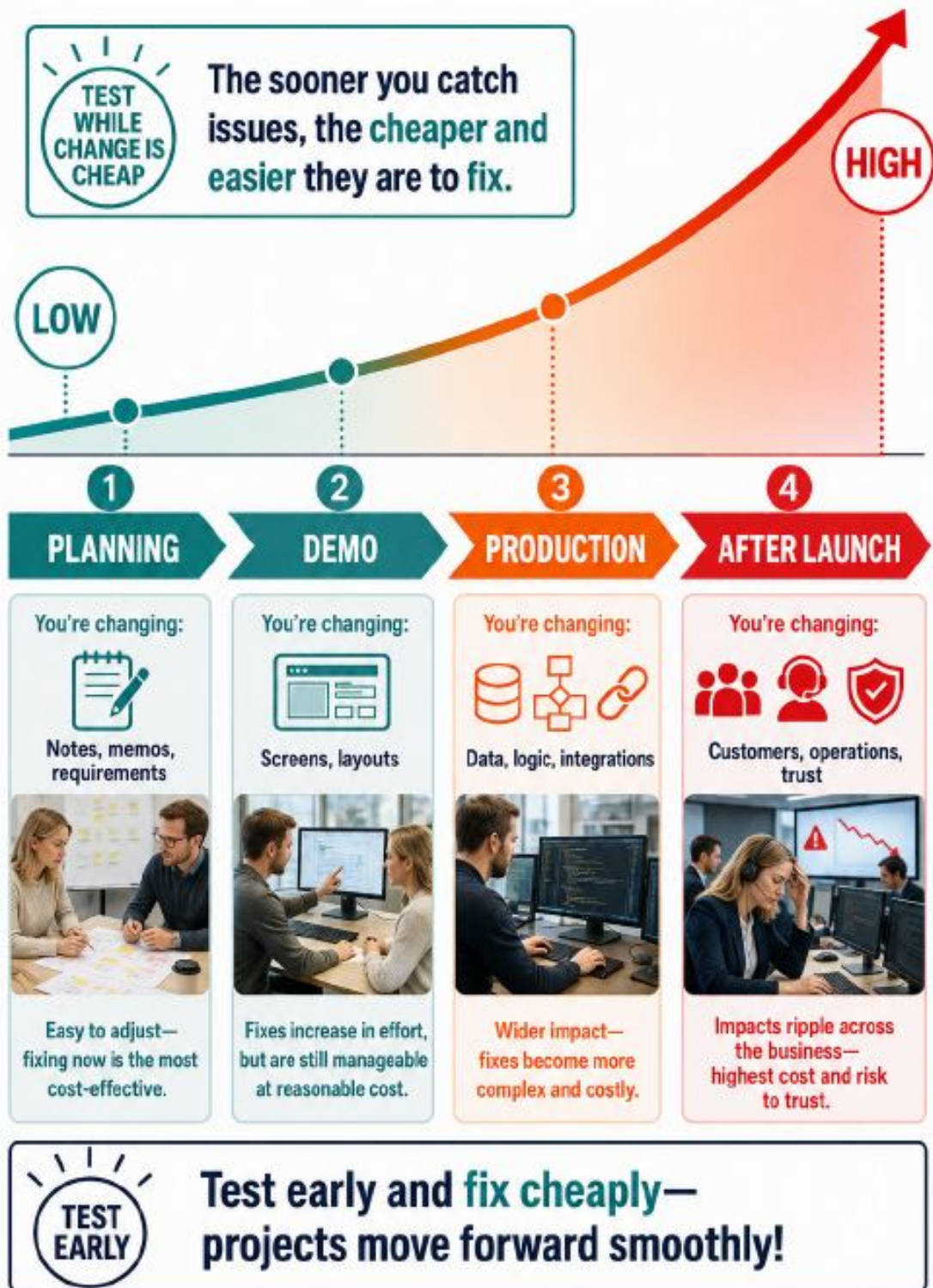


I don't know anything about "That's good" yet.

What do creators most want to hear when they show a demo?

"Sounds good."

Rework Costs Rise in Later Stages



“Different” after completion is expensive

Let's say you created a business system over several months.

I asked about the requirements.

Five Questions That Change the Answer

Do you have any thoughts?

Hmm, not really...

Polite but vague

- Sounds good
- Seems convenient
- Looks helpful

1 Do you like it?  

2 What is different?  

3 What is missing?  

4 Where did you hesitate?  

5 What would you do next?  

  **Specific insight, actionable answer** 

Don't ask, "Do you have any opinions?"

After showing the demo, some people ask me this.

"Do you have any opinions?"

Declare It Unfinished to Get Honest Feedback

FINISHED PRESENTATION



UNFINISHED DEMO



DEFENSIVE REACTION



HONEST FEEDBACK



Understand the work behind the request rather than just making it into a specification.

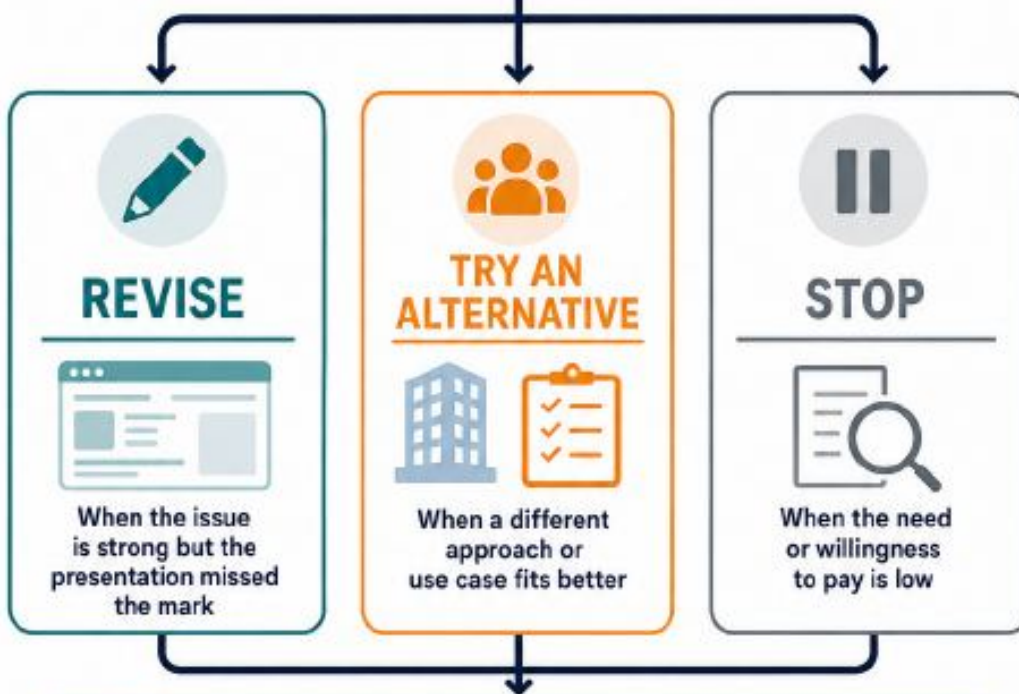
Human judgment is required here.



A Rejected Demo Still Becomes a Decision Asset



REJECTED DEMO



DECISION ASSET
Rejection is useful when it produces evidence for the next decision.

Chapter Takeaway

It is not a failure if a customer tells you that it is different.

Failure is not told until after completion.

Chapter 4: Align the Timelines of Managers and Developers

