

Executive Guide

Why I Build AI for Usability, Not Scope

There is a quiet failure mode in AI projects that nobody talks about in case studies: the system that does everything the brief asked for, and that nobody actually uses on a Tuesday morning.

Prepared by

Vikram Katyani

Founder, IntelliMinds Digital

Executive Summary

Scope makes a product look impressive in a demo. Usability is what makes people actually use it on a Tuesday morning. The teams that win optimise for the second.

There is a quiet failure mode in AI projects that nobody talks about in case studies: the system that does everything the brief asked for, and that nobody actually uses on a Tuesday morning.

It is the most expensive failure in our industry. It also looks great in a steering-group slide deck.

Contents

1. Scope wins demos. Usability wins adoption.
2. How this changes what we build
3. What it costs you to optimise for scope
4. Key takeaways

There is a quiet failure mode in AI projects that nobody talks about in case studies: the system that does everything the brief asked for, and that nobody actually uses on a Tuesday morning.

It is the most expensive failure in our industry. It also looks great in a steering-group slide deck.

Scope wins demos. Usability wins adoption.

Scope is what you put in the proposal. Twelve features, four integrations, three personas, two languages. It is easy to defend, easy to price, and easy to demo.

Usability is harder to articulate. It is the difference between an answer in two clicks and an answer in seven. It is whether the AI's confidence is visible at a glance or buried in a tooltip. It is whether the user trusts the result enough to act on it without checking.

None of that shows up in a feature list. All of it determines whether the system is used.

How this changes what we build

We deliberately ship narrower than the brief on first release. Fewer features, executed so well that the user does not have to think about them. Then we expand — informed by how people actually use it, not by what they said in the workshop.

That is not a refusal to deliver scope. It is a refusal to deliver scope that does not survive contact with a real user.

A feature nobody uses is not a feature. It is a maintenance cost.

What it costs you to optimise for scope

Three things, every time. First, complexity that slows every later change. Second, a UI that is harder to learn, which depresses adoption. Third, a budget pattern where the second half of the project is spent removing things the first half added.

If you have ever sat in a steering group six months after launch listening to a debate about "why aren't people using it more?", you have seen the bill arrive.

Key takeaways

- Scope is easy to specify. Usability is what determines adoption.
- Ship narrower than the brief on release one. Expand from real usage.
- Features nobody uses are a maintenance cost, not an asset.
- Optimise for the user's Tuesday morning, not the steering group's quarterly review.

Continue the Conversation

IntelliMinds Digital helps organisations turn AI ambition into dependable business capability. If this guide has raised questions about your own AI programme, we'd welcome the conversation.

Next steps

- **Take the free AI Readiness Assessment** — a five-minute diagnostic with a personalised report.
- **Book a 30-minute discovery call** to discuss your priorities.
- **Explore the full library** of AI Executive Guides at intellimindsdigital.com/insights/



Read the latest version online

<https://intellimindsdigital.com/insights/why-build-for-usability/>

Scan to open the live article, which is kept up to date as our thinking evolves.

IntelliMinds Digital

Founder: Vikram Katyani

vikram@intellimindsdigital.com

www.intellimindsdigital.com