

30.01.26 · PERSPECTIVE · 7 MIN READ

Designing systems that scale with the business, not against it.

A system that works at ten orders a day and breaks at two hundred wasn't really finished the first time.

Scale is a design constraint, not a future problem

It's tempting to build for today's volume and "deal with scale later." In practice, the parts of a system that break under growth are usually structural, how data is modeled, how responsibilities are split, and those are exactly the parts that are hardest to change after the fact.

We don't over-engineer for hypothetical scale. We just make sure the early decisions don't quietly assume the business will stay small.

What that looks like in practice

For the restaurant ordering system, that meant designing the dashboard to handle multiple locations from day one, even for a client with a single location, because the data model is far more painful to change later than to design correctly once.

It rarely means building more. It means building the right shape, so that more volume is a configuration change, not a rebuild.

The parts of a system that break under growth are the parts that were hardest to see on day one.