

Bespoke ERP Without the Development Project

Expert consulting

A proven automated platform

Fully managed hosting

Bespoke ERP has always made sense on paper and rarely in practice. Software shaped around how your business actually works beats bending your business to fit a package, but building it meant years of developers writing code by hand, at a price few smaller firms could justify. That has changed. AI can now generate the software itself, and with it goes most of the cost. What AI doesn't remove is the hard part: knowing what to build, and running it properly once it exists. That is what I provide. Each week I work with you to decide how your business should be modelled, drawing on twenty years of ERP experience. Your system is then generated on a platform I've engineered for exactly this: a deliberately chosen technology stack and an automated pipeline that turns agreed requirements into working, deployed software. Once it's live, I run it for you, overseen by someone who knows both the system and your business.

1 Working out what to build

This is the part of the service that matters most, and it is entirely human. An ERP system encodes decisions about your business: how stock is valued, when revenue is recognised, who approves what, and what happens when an order goes wrong. Get those right and the software follows. Get them wrong and no amount of code will save it. Making them well takes judgment, and that judgment comes from twenty years of designing ERP systems across accounting, inventory, ordering, logistics and compliance in the UK and Spain.

We meet weekly. You walk me through how your business really works, including the exceptions and workarounds that never make it into a requirements document, and I help you decide how the system should handle them. Often that means questioning how something is done today, not just copying it.

I've built my own tools to make these sessions count. Our meetings run on a custom platform where an AI assistant follows the conversation, already knows your system as it stands, and keeps a shared list of what we've agreed. Nothing goes on that list as agreed until I've confirmed it. After each session, our decisions become a precise written specification in your business's own language, with no loose ends, before anything is built. Over time, those specifications form a complete record of why your system works the way it does.

2 A platform built for this, not for everyone

Your system is generated on a platform I've built before your project starts. It has two parts: a deliberately chosen technology stack, and an automated pipeline that turns agreed specifications into working, deployed software.

A stack chosen for ERP, not for demos. Many AI-built apps run on whatever their tool defaults to. Those stacks are tuned for quick results and tied to a platform you rent indefinitely. Mine is chosen for what ERP needs to last:

Go	Fast, reliable and scalable. It compiles into a single self-contained program, so there's little to go wrong.
SQLite	Among the most thoroughly tested software ever written, running in billions of devices. Your whole database is one file you own, supported until at least 2050.
React and TypeScript	The industry's most widely used interface technology, so developers who can work on it are easy to find.
Open source throughout	No vendor lock-in, no subscriptions underneath, and nothing stopping another developer taking over.

I'll explain the reasoning behind each choice before we start.

An AI pipeline engineered, not improvised. Getting reliable software out of AI takes far more than asking it for code. My pipeline has been built over many months. It turns each specification into a structured build plan and holds the AI to automated quality gates it can't talk its way past. Independent AI reviewers then audit every piece of work without seeing how it was produced. Nothing reaches your system until it passes. AI is changing fast, and I maintain and upgrade the pipeline continuously, so your system benefits from every improvement. No process is perfect, though, and section 3 covers what happens when something slips through.

3 Run for you, properly

Software is only useful while it's running. Once your system is live, keeping it running is my job, so you don't need an IT department or a hosting contract of your own.

Hosting	Professional servers in European data centres, with encrypted connections as standard.
Backups	A tiered backup system: multiple independent copies, kept over different periods and in separate locations, including off-site storage. No single failure can take your records with it.
Monitoring	The system is watched continuously, errors are logged, and problems trigger alerts. I usually know about an issue before you do.
Updates	New features, fixes and security updates go live through the same automated pipeline, without disruptive upgrade projects.
Incidents	Handled by the person who designed your system and knows your business, not a support queue reading from a script.

The difference

Traditional custom ERP spends most of its budget and time on people writing code, and leaves you depending on whoever wrote it. My service puts that effort where it actually makes a difference. We decide together what your system should do. It's built on a platform engineered to last. And it's run properly once it's live. Changes we agree each week reach your system every week, not at the end of a long project. The result is software shaped around your business, on technology you can keep, looked after by someone who understands both.



Let's talk

Tell me what's broken, what you've tried, and what good would look like.

[**jon@pincas.eu**](mailto:jon@pincas.eu)