

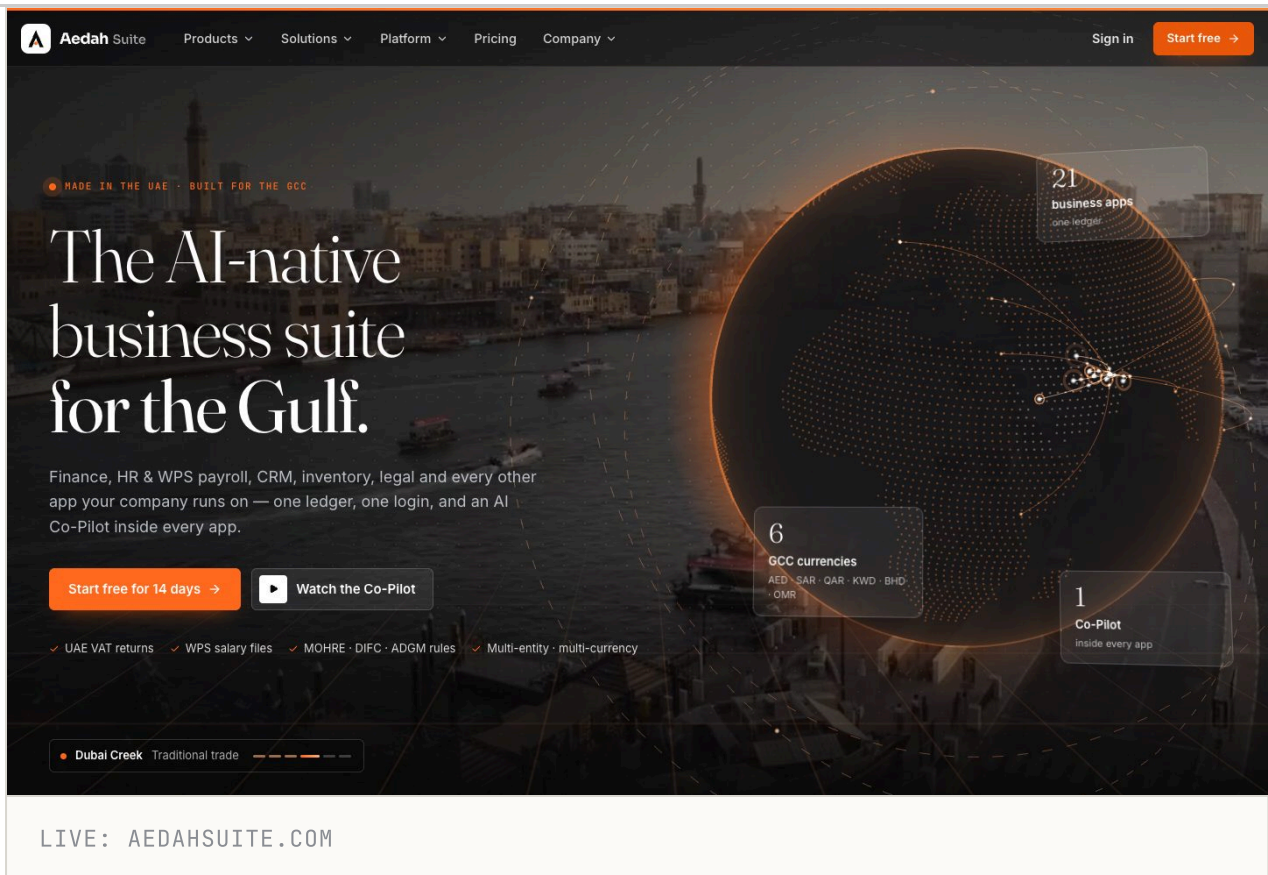
AedahSuite: twenty-two business apps for UAE and GCC companies on one platform

Small and mid-sized companies in the UAE usually run on a patchwork: one tool for the books, a payroll bureau for WPS, a CRM that does not know about invoices, inventory in a spreadsheet, approvals on WhatsApp. Every month...

Role: Founder, product owner and architect; built with AI-assisted engineering under my direction and review

Stack: Next.js 16, NestJS 10, Prisma 5 (multi-schema), PostgreSQL 18, Claude tool-use for the Co-Pilot, PM2 and nginx on Linux, Cloudflare

Live: aedahsuite.com



Context

Small and mid-sized companies in the UAE usually run on a patchwork: one tool for the books, a payroll bureau for WPS, a CRM that does not know about invoices, inventory in a spreadsheet, approvals on WhatsApp. Every month somebody re-keys the numbers before the VAT return. AedahSuite is my answer to that: separate apps for finance, people, sales, operations, documents and compliance that share one login, one set of companies and permissions, one ledger and one AI Co-Pilot.

It is my own product, not a client build, and it is pre-launch. It runs as a demo and staging environment with five industry demo companies loaded, and production hosting in the UAE is planned for when the first customer contract closes. I am stating that plainly because the scale of the engineering is real, while the customer base is not yet.

What I built

Twenty-two apps, each a bounded context

Finance and Accounting, HR and Payroll, CRM and Sales, Marketing, Project Management, Inventory and WMS, Procurement, Legal and Compliance, Customer Support, Fleet, Real Estate, Trade and Logistics, Analytics and BI, Retail POS and E-commerce, Events and Ticketing, e-Signature, Expenses and Cards, Business Mail, Drive and Docs, Password Vault, Learning and Training, and the AI Co-Pilot. Each app owns its own PostgreSQL schema with no foreign keys into another app, so an app can be licensed, removed or deployed on its own without dragging the others with it.

- ◆ **One ledger.** CRM invoices, POS shift takings, payroll runs, expense claims, supplier bills and ticket sales post to Finance as they happen, so the VAT return, cash position and dashboards stay current without exports.
- ◆ **UAE rules built in.** FTA-format VAT returns from the ledger, WPS salary files, gratuity and end of service under MOHRE, DIFC and ADGM rules, emirate and free-zone obligations in Legal, and Peppol PINT-AE e-invoicing in Finance.
- ◆ **GCC entities.** Each company in a group carries its own currency, VAT profile, weekend and public holidays, with reporting rolled up in AED or any base currency.

A shared platform layer

Approvals, audit trail, idempotency keys on money-moving and sending routes, an outbox for events between apps, notifications, background jobs, custom fields and per-company app licences are built once and used by every app. Tenant isolation is enforced at the data layer: every request sees exactly the company it is authorised for.

An AI Co-Pilot that cannot act alone

The Co-Pilot reads across every app a company licenses, within the permissions of the person asking. It analyses, drafts and proposes actions through tool calls, and every create, approve or update waits for a human to confirm. Salaries and vault secrets are never readable to it, each step is audited, and a customer can switch it off, point it at a local model or use an approved provider.

Apps that would be products on their own

- ◆ **Business Mail** hosts company email on the customer's own domain: shared inboxes with SLAs, rules and templates, DNS records generated per installation, DKIM signing, inbound SMTP and direct delivery.
- ◆ **e-Signature** routes documents for signature with access codes and one-time passwords and seals the final PDF with a SHA-256 hash and a public verification page.
- ◆ **Password Vault** is zero-knowledge: items are encrypted in the browser, unlocked with passkeys, and shared through groups and access requests.

Decisions worth noting

- ◆ **Separable by design.** Dedicated schemas per app make a modular product possible: a customer can start with one app and add others without a migration of what they already have.
- ◆ **Three ways to run it.** A shared cloud, a dedicated instance on the customer's own PostgreSQL database, or on the customer's own servers. The same code serves all three, because every domain and host is configuration, never hard-coded.
- ◆ **No source on servers.** Builds happen on my machine and only build artifacts are shipped; database migrations run over an SSH tunnel after a fresh backup.
- ◆ **Honest claims.** The product does not promise UAE data residency or Arabic interfaces until they exist, and the website says so.

Why it is in the portfolio

It shows the full breadth of what I do in one place: product definition across twenty-two business domains, a multi-tenant architecture that stays separable, regional compliance turned into software, AI that is useful without being trusted blindly, and the operations to run it. It is also the platform I implement for clients who want one system instead of a patchwork.