Howie Wang

AI-Native Full-Stack Developer | Claude Code · Cursor | Node.js · WeChat Mini Programs · Cloud

Belconnen, ACT · howiewang122@gmail.com · github.com/how1ewang

PROFILE

I build and run production software on my own by directing AI coding agents. Over the past year I took a biotech company's commerce platform from an empty repository to 900+ members as the only person on the project — Mini Program, staff console, API, registration tool, cloud infrastructure and operations — with Cursor and then Claude Code writing the code. I own the requirements, the product and design decisions, the verification (760+ automated tests, scripted production audits) and every production change. Final-semester Master of IT at UNSW; looking for AI-native full-stack developer, AI product engineer or forward deployed engineer roles.

TECHNICAL SKILLS

AI-Native Development	Claude Code (agentic coding, parallel sessions, approval-gated production work), Cursor (agent mode); spec-first prompting, agent context (AGENTS.md, Cursor rules, persistent memory), reviewing and testing AI-written code, guardrails for agent-driven deploys
Back-End	Node.js 22, Express, PostgreSQL 18, SQLite, JWT / RBAC authorisation, RESTful API design
Front-End	HTML / CSS / JavaScript, WeChat Native Mini Program (WXML / WXSS / WXS)
Payments & Integrations	WeChat Pay v3 (JSAPI / H5), Alipay (PC / JSAPI), AdaPay, Alibaba Cloud SMS, WeCom (Enterprise WeChat) SSO
Cloud & DevOps	Alibaba Cloud ECS / OSS (direct and multipart uploads, transfer acceleration, custom-domain media), Docker Compose, Nginx reverse proxy & HTTPS, PM2 process management, Let's Encrypt auto-renewal
CI/CD & Tooling	GitHub Actions (automated testing + Docker build), ESLint, Jest unit testing, miniprogram-ci automated deployment
Databases	PostgreSQL incremental migration framework (49 migrations), dual-track PG/JSON storage with fallback, SQLite embedded storage, automated backups with verified restores

PROJECT EXPERIENCE

YuanQiPai (元气湃) — Integrated Business Platform | Sole Developer, built with AI agents

Oct 2025 – Present | Node.js · Express · PostgreSQL · WeChat Mini Program · Alibaba Cloud

An end-to-end digital commerce platform for a biotech company, serving both a dealer network and consumer members. Built and run single-handedly — system design, full-stack development, testing, production deployment and operations — with AI coding agents writing the code under my direction. 240+ commits; live since the 10 October 2026 launch with 900+ members.

How It Was Built

- One developer, AI coding agents as the implementation team: Cursor (agent mode) from the first commit, Claude Code since October 2026
- Shared agent context — a prompt template, Cursor rules, an AGENTS.md (services, ports, database caveats, conventions) and persistent Claude Code memory (deploy rules, lessons from past incidents) — so every session starts from the same ground rules
- Spec-first loop: written requirements and acceptance criteria → agent implementation → automated tests → my review and on-device testing → hash-verified deploy
- Parallel agent sessions for independent work — a Node.js / PostgreSQL major-version upgrade, a secret rotation and a media-domain migration ran in the same week — with every production step gated on my approval
- Tests as the contract for AI-written code: 760+ Jest cases, ESLint at zero errors and scripted production audits after each deploy (details below)

System Architecture

Four interconnected modules built on a unified back-end with multi-channel front-ends:

- **Admin Console** (45 pages, 100+ screens) — Four staff roles with two-person approval for money movements, page builder / design editor, product & order management, membership tiers & levels, coupons, content library, online academy, dealer network management, anti-counterfeit traceability, financial reconciliation & report exports, customer registration sync
- **Node.js API Service** (18 route modules, 43 service modules) — Express REST architecture, JWT + RBAC multi-role access control, layered middleware (auth / rate-limiting / error handling)
- **WeChat Mini Program** (60 pages) — Dynamic home-page rendering engine, product catalogue & shopping cart, online payment, user profile, team & agent hierarchy, content browsing & upload, anti-counterfeit scan verification, loyalty points wallet & withdrawals
- **Customer Registration Sub-system** (independently deployed) — Public-facing mobile form + admin back-office, SQLite storage, ECharts data dashboard with three-level China map drill-down, real-time data sync with the main platform

Key Technical Highlights

Payment Pipeline & Financial Integrity

- Integrated three payment channels — WeChat Pay v3 API (JSAPI / H5), Alipay (PC / JSAPI) and AdaPay — behind a unified payment abstraction layer
- Built a complete after-sales state machine: refund request → review → original-channel refund, including orphan refund detection (auto-clawback of delayed payments arriving after order cancellation)
- Connected to WeChat Order Centre so users can track orders directly from WeChat's native “My Orders” interface
- Commission and monthly reward settlement: configurable rules and thresholds, rule snapshots on every order, draft → recalculate → lock → payout, with reversal
- Two-person (maker-checker) approval for every money movement — receipts, refunds, dealer orders and settlements — plus time-limited permission grants
- WeChat Pay public-key verification of payment and refund notifications for merchants without platform certificates

Database Evolution & Dual-Track Resilience

- Designed a PostgreSQL incremental migration framework (000 → 047, 49 migrations) with automatic detection and application on startup
- Upgraded production off end-of-life Node.js 18 / PostgreSQL 13 to Node.js 22 / PostgreSQL 18 with a pre-change snapshot and rollback plan — about 30 seconds of downtime
- Daily automated backups of both databases (PostgreSQL and SQLite), checked by test restores
- Implemented a PG-primary / JSON-file-fallback dual-track read/write strategy, enabling zero-downtime database transitions in production
- Automatic empty-table back-fill from JSON seed files, ensuring no data loss on fresh deployments

Deep WeChat Ecosystem Integration

- WeChat login: silent session login + one-tap phone number authorisation, with automatic new/existing user detection and account merging
- WeCom SSO: admin panel supports Enterprise WeChat QR-code login, OAuth callback + contact directory sync
- miniprogram-ci automation: CI pipeline directly uploads Mini Program preview and production builds
- Full HTTP → HTTPS migration: automated URL normalisation in database records, cross-origin resource headers injection, CDN/OSS dual-egress strategy

Security Hardening

- Helmet security headers + fine-grained CORS configuration + tiered rate limiting (login / submission / general API)
- Timing-safe token comparison (`crypto.timingSafeEqual`) to prevent timing side-channel attacks
- Comprehensive audit logging of sensitive operations, recording IP, operator and timestamp
- RBAC multi-role access control with endpoint-level permission checks
- Secret rotation, key-only SSH and least-privilege storage access; login-protected access to personal-document uploads; configuration-drift checks between servers and version control

Engineering Quality

- Jest test suite: 61 test files, 760+ test cases covering API contracts, after-sales refund flows, payment simulation, security and data consistency
- GitHub Actions CI: automated test execution + Docker image build on push, branch protection on `main`
- ESLint full-repo linting, maintained at 0 errors
- Docker Compose orchestration: PostgreSQL + Redis (reserved) + Back-end + Nginx, single-command startup

Cross-System Data Synchronisation

- Designed an asynchronous sync architecture from the registration sub-system to the main platform: immediate enqueue on submission, exponential back-off retry on failure (up to 30 attempts)
- Intelligent member matching: prioritises country + ID document number, falls back to phone number
- Placeholder member claim mechanism: registration creates a placeholder record; when the customer later logs into the Mini Program, the account is automatically claimed by phone number match — no duplicate records
- End-to-end referral tracking: from registration link parameters through to Mini Program `referrer_id`

Deployment & Operations

- Production on Alibaba Cloud ECS: PM2 process management + Nginx reverse proxy + Let's Encrypt auto-renewal
- Alibaba Cloud OSS: browser-direct, chunked and resumable uploads (4 parallel parts, automatic retry, transfer acceleration — about 21× faster from overseas); videos served from a custom OSS domain with automated certificate renewal, offloading ~80% of server egress
- Hash-verified deployment script: diffs against the live commit, refuses stale checkouts and server-side drift, backs up, verifies every file and rolls back automatically; 67 Mini Program versions shipped via miniprogram-ci
- Domain architecture: API at api.yuanqipai.com.cn + registration at djb.yuanqipai.com.cn
- Health checks, production verification checklists and automated audits covering every admin screen, Mini Program page, API endpoint and media asset

Customer Registration System (YuanQiPai Sub-project) | Sole Developer, built with AI agents

May 2026 – Jul 2026 | Node.js · Express · SQLite · ECharts

A standalone customer information collection and management system for offline business scenarios, supporting customers from 20 countries.

- **Mobile form:** 20-country selection with cascading fields, China province/city/district three-level selector (3,423 districts), ID document photo upload, product feedback and satisfaction survey
- **Admin data dashboard:** ECharts China map visualisation with national → province → city → district three-level drill-down, automatic colour gradient based on volume
- **Security:** IP rate limiting (30 submissions/min), login failure lockout, secure session configuration, trusted reverse proxy chain
- **Data export:** CSV full export, region and time-range filtering, search by name / phone / WeChat ID / ID number
- **Cross-system sync:** Webhook real-time push + async queue retry, achieving eventual consistency between two independently deployed systems

SUMMARY OF STRENGTHS

- **AI-native delivery** — Directing AI coding agents (Claude Code, Cursor) to ship production software alone: clear specs, small verified steps, tests as the contract, and human approval on anything irreversible
- **End-to-end solo delivery** — Experienced across the full project lifecycle: requirements analysis, system design, front-end & back-end development, database architecture, testing and production deployment
- **Payment systems** — Hands-on integration with WeChat Pay v3, Alipay and third-party aggregated payment providers; strong awareness of financial data integrity
- **WeChat ecosystem** — Deep understanding of Mini Program lifecycle, login flows, payment processes and Order Centre integration
- **Cloud-native deployment** — Proficient with Docker, Nginx, PM2 and CI/CD pipelines; real-world production operations experience on Alibaba Cloud
- **Database design** — Skilled in incremental migration frameworks, dual-track storage strategies and data consistency guarantees at enterprise scale
- **Production operations** — Upgrades, backups, security reviews and incident response on a live system with real customers
- **Engineering discipline** — Committed to code quality, automated testing, security hardening and CI/CD best practices

EDUCATION

Master of Information Technology — University of New South Wales (UNSW Sydney) · graduating December 2026