JON WAYNE CABUSBUSAN

Full-Stack Product Engineer | Web & Mobile | AI-Assisted Development

Iloilo City, Philippines | jonwayne.cabusbusan@gmail.com | +63 945 289 7584

linkedin.com/in/jnwync · jnwync.vercel.app · github.com/jnwync

SUMMARY

Product Engineer with full-stack and mobile experience delivering production software across React/Next.js web applications, Node.js/Express services, and React Native/Expo mobile apps. Strong in TypeScript, PostgreSQL, Prisma, Drizzle ORM, Firebase, and Supabase, with hands-on experience in authentication, authorization, business workflows, automated testing, accessibility, and release engineering. Uses structured AI-assisted and agentic development workflows for repository investigation, planning, implementation, debugging, testing, QA, code review, and documentation.

SKILLS

Languages: TypeScript, JavaScript, SQL, HTML5, CSS3

Frontend & Mobile: React, Next.js, React Native, Expo, Tailwind CSS, Radix UI, TanStack Query, SWR, i18next

Backend & APIs: Node.js, Express.js, Next.js APIs, REST, OpenAPI, Zod, webhooks, Server-Sent Events

Data & Storage: PostgreSQL, Prisma, Drizzle ORM, Firebase, Supabase, S3-compatible storage, Sanity CMS

Auth & Security: Auth.js, Firebase Auth, Supabase Auth, JWT, OAuth 2.0, RBAC

Testing & Quality: Jest, Vitest, Playwright, React Testing Library, Supertest, WCAG accessibility

Cloud & Mobile Release: Docker, GitHub Actions, Vercel, Sentry, Expo EAS, Xcode, GCP

AI-Assisted Development: Claude Code, OpenAI Codex, GitHub Copilot, Cursor, OpenCode

WORK EXPERIENCE

Inertia (Trackbill) | Full-Stack Product Engineer **Jul 2026 – Oct 2026**

Remote / Contract / Product Engineering across Web, Android and iOS

- Delivered product, reliability, and platform improvements across Node.js/Express/Prisma services, Next.js/React web, and Expo/React Native mobile apps spanning three production repositories.
- Hardened receipt ingestion and WhatsApp workflows by enforcing a single valid Split/No action per receipt with atomic state commits, stale/duplicate-action rejection, localized feedback, and regression coverage across Meta and Evolution webhook paths.
- Unified incoming-receipt routing across WhatsApp and native-app uploads with concurrency-safe folder resolution, idempotent folder reuse, explicit-folder preservation, optimistic rollback, and localized settings across 15 languages.
- Improved upload reliability by moving a 25–28 second OCR classification task off the synchronous request path, replacing a global 30-second cutoff with a 120-second default, and validating a 36.01-second multipart upload flow.
- Expanded reporting and administration with organization report history, OCR usage by channel, currency-aware MRR, cursor-paginated users, subscription-seat editing, and mobile export history, downloads, reruns, and CSV/XLSX/PDF support.
- Supported mobile release readiness through reproducible iOS IPA builds, Sign in with Apple and privacy configuration, Android/iOS App Links across two domains, and App Store asset and image-orientation fixes.
- Contributed 106 merged PRs across web (45), backend (29), and mobile (32) repositories, covering product workflows, reliability, testing, and release tooling.

Mashup Technology Ventures Inc. (OKRa) | Full-Stack Developer **May 2026 – Jul 2026**

Remote / Project-Based Development

- Built a department-first collaborative work-tracking platform spanning workspaces, workflow columns, tickets, cross-department sharing, comments, labels, activity history, invitations, attachments, and administrative oversight.
- Connected frontend, API, and database layers through reusable React workflows, shared TypeScript/Zod contracts, 32 current route handlers, PostgreSQL, Drizzle ORM, and 21 database migrations.
- Implemented security-critical authentication and authorization with Auth.js credential sessions, session versioning, route protection, role- and workspace-scoped permissions, password and invite flows, database-backed rate limiting, and administrative safeguards.
- Integrated resilient S3-compatible attachment and avatar storage using upload validation, staged and atomic persistence, per-kind size controls, protected preview/download behavior, missing-object handling, and recovery states.
- Authored 83 automated test files covering API routes, database behavior, shared schemas, authentication, responsive UI, accessibility, keyboard interaction, and end-to-end workflows.
- Established maintainable project boundaries through a pnpm monorepo separating the Next.js application, PostgreSQL/Drizzle database package, and shared TypeScript/Zod contracts.

Reisky Martial Arts — Surrey, BC, Canada | Freelance Full-Stack Developer **Jan 2026 – Apr 2026**

Freelance / International Client Project

- Designed and delivered a production website end-to-end across 23 routes using Next.js 16, React 19, strict TypeScript, responsive layouts, reusable components, and production deployment.
- Architected a flexible content and service layer with 12 modules switching at runtime between Sanity CMS/GROQ and static fallback data behind a single environment-controlled feature flag.
- Built an end-to-end type-safe content pipeline connecting 50 Sanity schema types to generated types, 12 tested anti-corruption mappers, and application domain models.
- Established 55 automated tests and measurable quality standards achieving Lighthouse Accessibility 100 across all 12 tested routes on mobile and desktop, with Best Practices 96 and SEO 92.

PackUp (SuppliPack) | Full-Stack Developer **Jun 2025 – Nov 2025**

Part-Time Startup Project

- Built responsive customer and administrator workflows (browsing, ordering, payments, profiles, notifications) with Next.js, React, TypeScript, Tailwind CSS, and Radix UI, backed by REST API routes and service-layer modules using Firebase Admin SDK, Firestore, Auth, and Storage.
- Enforced token verification, role-based authorization, and ownership checks at the API layer to protect customer records and administrator-only operations.
- Implemented concurrency-safe Firestore transactions for collision-free sequential product and order IDs and event-triggered notifications across five order lifecycle events.

DemiGAD (BayloCentral) | Full-Stack Developer **Jun 2025 – Nov 2025**

Part-Time Startup Project

- Built marketplace browsing and seller-facing experiences (listings, search, autocomplete, filters, cart, seller dashboard) using Next.js 15, React 19, SWR, Tailwind CSS, and Radix UI, on PostgreSQL-backed workflows with Prisma ORM, Supabase Auth, and Zod validation.
- Implemented a social feed supporting four post formats with likes, comments, shares, and real-time updates across relational data models and reusable client-side fetching hooks.
- Protected data integrity with database constraints preventing duplicate social actions while contributing reusable query fragments and domain-organized frontend and backend code.

Wisdomous Inc. (Apollo Medical Group) | Project-Based Full-Stack Developer **Jan 2025– May 2025**

Remote / Project-Based Development

- Led frontend delivery of a patient portal and hospital booking system producing responsive patient and administrator workflows from product requirements through implementation and end-to-end QA.
- Supported backend development and database improvements by refactoring queries and data structures to improve page responsiveness, booking reliability, and maintainability.
- Contributed across the complete delivery lifecycle including UI/UX decisions, API integration, admin features, sprint reviews, regression testing, and production release work.

DTI Iloilo MSME Industry Map | Full-Stack Developer **Aug 2024 – May 2025**

External client partnership through Central Philippine University

- Built an interactive geographic information system for browsing and managing Iloilo MSMEs, including map markers, filters, administrative forms, user management, and statistical dashboards.
- Developed REST API handlers for MSMEs, related addresses, users, signup requests, account approval, role lookup, and GeoJSON data using Next.js, TypeScript, Firestore, and Firebase Admin SDK.
- Created reusable Firestore data-access abstractions and implemented runtime validation, API-layer RBAC, one-to-many address lifecycle management, and manual cascade deletion for related records.
- Separated browser and privileged server operations while supporting an approval workflow that created Firebase Auth accounts, persisted user records, and removed completed signup requests.

EDUCATION

Central Philippine University

Bachelor of Science in Software Engineering

Relevant Coursework: Data Structures & Algorithms, Database Systems, Software Architecture, Software Testing & Quality Management, Machine Learning