

TONG DUC THANH NAM

AI Developer Intern | AI Product Builder | Full-stack TypeScript Developer

Ho Chi Minh City, Vietnam | 0869887363 | tongducthanhnam@gmail.com
github.com/TongDucThanhNam | tongducthanhnam.id.vn | English: Aptis B2

PROFESSIONAL SUMMARY

Software Engineering graduate focused on AI application products, browser extensions, automation tools, web apps, and mobile utilities. Built and shipped projects across AI workspaces, published Chrome/Firefox extensions, Discord operations bots, knowledge inbox tools, e-commerce prototypes, and Android apps. Strongest value: turning vague workflows into working products across UI, backend, persistence, external APIs, automation, deployment, and AI-assisted prototyping.

TECHNICAL SKILLS

Languages: TypeScript, JavaScript, Kotlin, Python	AI Apps: Vercel AI SDK, streaming responses, model/tool routing, BYOM workflows
Frontend: React, Next.js, TanStack, Tailwind CSS, Zustand, Shadcn/UI	Backend: Hono, Express.js, Node.js, tRPC, REST APIs, Bun
Extensions: WXT, Chrome Extension, Firefox Extension, content scripts, service workers, tab messaging	Data/Infra: Cloudflare Workers/D1/R2, Drizzle, SQLite, MongoDB, Google Sheets API
Automation: Discord bots, slash commands, webhooks, RBAC flows, OCR fallback	Mobile: Android IME, Jetpack Compose, Kotlin

EXPERIENCE

Backend Developer - Amazing Technology Solutions Co., Ltd. <i>Ho Chi Minh City, Vietnam Node.js, Express.js</i>	May 2024 - Jul 2024
- Contributed to backend development for an enterprise interview workflow management project. - Worked on Node.js/Express.js backend tasks and API-related delivery in a short project timeline.	

SELECTED PROJECTS AND PRODUCTS

Eragear Web Copilot - Published Browser Extension <i>Repository: github.com/TongDucThanhNam/eragear-web-extension WXT, React, Tailwind CSS, browser APIs</i>	Chrome + Firefox
- Built and published a browser extension that connects browsing context with an AI workspace through a client-side bridge and BYOM workflow. - Implemented Google Search overlay, tab discovery/opening, tab-to-tab provider messaging, AI-site settings, auto-submit controls, copy helpers, and streaming response UI. - Built a research workflow where the user verifies search results and pages before clipped content is converted into clean Markdown and sent back into the AI context. - Designed around privacy-first constraints: local browser execution, direct provider-session interaction, and no extra middleman server for prompt/response handling.	
Eragear AI Workspace - AI-native Web App <i>Live: eragear-web-terasumi.terasumi.workers.dev React/TanStack, Hono, Cloudflare Workers/D1, Drizzle, Better Auth, Bun, AI SDK</i>	Full-stack AI Product
- Built an AI workspace for long-running conversations, artifacts, model/tool selection, research flows, and human-in-the-loop interaction. - Implemented core AI product flows: streaming chat, selectable model/tool/search controls, usage metadata, persistence, credit-accounting architecture, and Cloudflare deployment. - Explored context engineering patterns by separating chat UX, artifact sidecar, research input, local knowledge, and browser-assisted workflows instead of treating AI as a plain chatbot box. - Built the product as a small SaaS-style system with authentication, database schema, backend routes, frontend workspace state, and deployable Cloudflare infrastructure.	
Private Operations Discord Bot - Internal Company Automation <i>Bun, TypeScript, discord.js, sunar, Google Sheets API, webhooks, RBAC, testable domain modules</i>	Automation Tool
- Built a Discord bot to support internal operations for a company, without exposing the company's private operational data in public materials. - Implemented HR workflows for leave, overtime, late/early requests, personal statistics, admin registration/editing, and Google Sheets persistence. - Added operations modules for banking notifications, vehicle route/fuel records, fuel invoice XML parsing, Gemini OCR fallback for image invoices, and PO receipt drafts. - Separated Discord handlers from business logic using module boundaries, slash commands, modal/button interactions, role-based permissions, environment validation, tests, and deployment runbooks.	

Link Saver - Knowledge Inbox Web App + Extension

Product Prototype

Demo: link-saver-web-terasumi.terasumi.workers.dev | TanStack Start, React, WXT, tRPC, Better Auth, Drizzle, Cloudflare D1/R2, Polar, Bun/Turborepo

- Built and deployed a knowledge inbox around the product loop: save URL/current tab, extract readable content, archive snapshots, read, search/recall, and export.
- Designed a monorepo with web app, browser extension, API package, auth package, database schema/migrations, shared UI, typed environment access, and Cloudflare deployment package.
- Modeled storage boundaries with relational metadata in D1/SQLite and large archived snapshots in R2, with AI summaries grounded by citations and export targets such as Markdown, JSON, and llms.md.

Eragear Code Copilot - Desktop-first AI Coding Assistant

In Progress

Electron, Agent Client Protocol, TypeScript, runtime packages, tRPC, Bun/Turborepo

- Designed a desktop-first AI coding assistant around Electron, Agent Client Protocol, shared runtime packages, and strict app/runtime package boundaries.
- Kept Electron main/preload thin and moved business behavior into runtime packages: session lifecycle, tool-call handling, permission boundaries, persistence, settings, background tasks, and tRPC procedures.
- Modeled a private local execution path from renderer to preload bridge, Electron IPC, DesktopRuntimeHost, runtime service over stdio NDJSON, runtime use cases, and ACP agent process.

AI Chatbot Tree - Branching Conversation and Context Control

AI UX Prototype

Demo: ai-chatbot-tree.vercel.app | Next.js, AI SDK, Zustand, D3.js, React Flow, Tailwind CSS

- Built and deployed a tree-based chat UI where the Conversation Tree is the single source of truth and the model context is projected from an explicit recipe.
- Implemented branching, regeneration, manual message insertion, context preview/export/import/reset, D3/React Flow visualization, and local persistence with compression/checksum.

Seller Keyboard - Published Android Utility App

Google Play

Google Play: [com.terasumi.sellerkeyboard](https://play.google.com/store/apps/details?id=com.terasumi.sellerkeyboard) | Kotlin, Android IME, Jetpack Compose

- Published a custom Android keyboard with 100+ downloads for online sellers to insert reusable text/image snippets while chatting with customers.
- Focused on a narrow workflow: reducing repeated typing for product introductions, prices, promotions, customer replies, and seller chat operations.

rem-viet - Full-stack E-commerce Website

Public Repository

Demo: rem-viet-web-terasumi.terasumi.workers.dev | React/Next.js, Node.js, Express.js, MongoDB, Docker, Bun/Turborepo

- Built and deployed a full-stack e-commerce website for selling curtains and blinds with separate frontend/backend services and Dockerized local execution.
- Implemented learning-oriented backend/frontend architecture: Express routing/middleware, MongoDB/Mongoose CRUD schema, controller/service/repository separation, Swagger API docs, Next.js routing/fetching/SSR, and dashboard/admin-oriented roadmap.

FashionAI - Personalized E-commerce with AI Recommendation

Academic/Product Prototype

Next.js, Shadcn/UI, Tailwind CSS, Node.js, Express.js, MongoDB, Python, FastAPI, Pandas, NumPy, Scikit-learn, TensorFlow

- Built a personalized e-commerce website prototype using an AI service layer for recommendation-related workflows.
- Separated the system into frontend, backend, and AI service setup, with Docker Compose deployment path and local development instructions for each service.

EDUCATION

Ton Duc Thang University - Software Engineering

Sep 2020 - Apr 2025