Hung Phi Vo

Software Engineer
Ho Chi Minh City, Vietnam | hungphiivo@gmail.com | linkedin.com/in/hungphiivo

Software Engineer with nearly five years building web platforms and back-end systems in TypeScript and Python, from greenfield architecture to production. Recent work centres on regulated multi-tenant SaaS (identity verification, KYC/AML, audit trails) and AI agents built on large language models (RAG, MCP).

SKILLS

- **Languages:** TypeScript, JavaScript, Python, SQL
- **Back-end:** Node.js, NestJS, Express.js, Django, PostgreSQL, MongoDB
- **Front-end:** React, Next.js
- **Architecture:** DDD + CQRS, Hexagonal (ports and adapters), multi-tenancy, RBAC
- **AI:** AI agents, RAG, MCP, AWS Bedrock, OpenAI, Gemini, LangChain, LangGraph, Langfuse
- **Security:** OIDC, KYC/AML integration (Sumsub), OWASP ZAP (DAST)
- **Cloud, DevOps & Testing:** AWS, Docker, Nginx, Playwright, Testcontainers, Selenium, Behave, Cucumber, Pytest

EXPERIENCE

TMA SOLUTIONS, HCMC Jan 2022 – Oct 2026
Software Engineer

- Owned features end to end: requirements analysis, architecture, implementation, testing, review and release
- Integrated third-party and regulated external services: identity, KYC/AML, payments and LLM providers
- Set up project foundations: repository structure, environment configuration and test infrastructure
- Named Star Performer for Q4 2024, chosen across two development centres

MEU SOLUTIONS, HCMC Aug 2021 – Dec 2021
Information Security Intern

- Performed basic security testing and vulnerability assessment
- Supported back-end development tasks using Express.js and PostgreSQL

PROJECTS

FINANCIAL ADVISORY PLATFORM Apr 2026 – Oct 2026
One of two main developers, owning identity, compliance and AI, in a team of 5
A multi-tenant SaaS for financial advisory firms: onboarding with KYC/AML checks, financial plans and advice documents, e-signing, compliance review and an AI assistant. TypeScript monorepo (NestJS + React) on AWS ECS.

- Integrated national digital identity sign-in (OIDC + PAR + JWKS) for adviser-led and self-serve onboarding
- Moved Sumsub AML screening from enterprise-only webhooks to a background job, extending screening to every onboarding path
- Built the Enhanced Due Diligence module: a five-step review wizard with an immutable audit trail
- Rewrote a slow approval flow with set-based writes, cutting database operations from ~8,400 to ~1,100 and median time from 5.1 s to 1.4 s, with all 11 affected tables matching the previous implementation
- Built the AI advisory agent on AWS Bedrock (Claude) behind a model gateway, with model fallback, per-request cost tracking and web search with cited sources
- Built a per-customer knowledge base (RAG) that mixes PostgreSQL full-text search with pgvector
- Set up the Turborepo monorepo and a DDD + CQRS back-end, with a port/adapter boundary around every external dependency
- Implemented RBAC with the same checks in navigation, front-end routes and the API, plus per-firm data isolation
- Built the test and security foundation: unit, end-to-end, real-PostgreSQL (Testcontainers) and Playwright test layers, plus OWASP ZAP DAST scans from local to production with a finding-triage process

  Tech: TypeScript, NestJS, React, PostgreSQL (pgvector), AWS Bedrock, AWS ECS, Docker, Playwright

CRM PLATFORM Apr 2025 – Apr 2026

Main developer of the AI agent and its MCP server, in a team of 8

A CRM (contacts, enquiries, leads, opportunities, forms) with two AI agents: one scores and suggests leads, the other lets users run CRM actions by chatting.

- Built the chat agent from scratch in Python with LangChain and LangGraph, turning users' requests into CRM actions through tool calls
- Built an MCP server from scratch with seven tools (create, update and bulk-update contacts and leads, convert contacts to enquiries), so the agent acts through the CRM's own API
- Made the agent provider-agnostic (OpenAI, Gemini, Ollama), switchable by configuration
- Added pgvector semantic search over contact notes, and Langfuse for prompt management and tracing

  Tech: TypeScript, Python, LangChain, LangGraph, MCP, NestJS, React, PostgreSQL (pgvector)

REAL ESTATE BROKERAGE WEBSITE Sep 2025 – Dec 2025

Front-end and Strapi developer from project kickoff, in a team of 5

A website for a real estate brokerage: buy, sell and lease services, plus sale and lease listings managed in Strapi CMS.

- Built the home, listings and case studies pages in Next.js (App Router) and TypeScript from the designer's mockups, rendered server-side for SEO
- Connected those pages to their Strapi APIs and saved enquiry form submissions to Strapi, so the client manages listings, content and enquiries in the admin panel without a developer

  Tech: TypeScript, Next.js, Tailwind CSS, Strapi, PostgreSQL

BLOCKCHAIN DATA API PLATFORM Oct 2024 – Mar 2025

Full-stack developer of the client dashboard and the load balancer from kickoff, in a team of 4

A subscription API through which clients read and write data on the IOTA blockchain, built from three parts: a client dashboard, a load balancer and an API gateway.

- Built the client dashboard, front-end and back-end, where clients manage the API gateway's endpoints, their API keys and subscription plans
- Integrated Stripe checkout and webhooks for two subscription plans, issuing the client's API key once payment cleared
- Built the load balancer and API proxy (Nginx and an Express.js service) that validates each API key and enforces the plan's limits on requests per minute and payload size before forwarding to the API gateway

  Tech: TypeScript, Next.js, Node.js (Express.js), MongoDB, Stripe, Nginx

DATA MANAGEMENT PLATFORM Jan 2022 – Sep 2024

Full-stack developer and main author of the end-to-end test suite, in a team of 10

A research data platform for public-health field programs in more than 20 countries, on one multi-tenant codebase: a Django core app, Plotly Dash reports and an ODK-X field app for field workers.

- Built features in Python and JavaScript on the Django core app and the ODK-X field app, much of it on the field app workers use to get tasks and submit data
- Made flaky end-to-end runs resume from the failed step instead of restarting from zero, cutting full run time by about 30% and shortening the team's preparation for each production release
- Wrote Selenium and Behave end-to-end tests covering the whole platform, one full business flow per feature file
- Wrote unit and BDD tests every sprint: Jasmine and Pytest on the core app, Cucumber on the field app

  Tech: Python, JavaScript, Django, Plotly Dash, ODK-X, PostgreSQL, Selenium, Behave

EDUCATION

TAN TAO UNIVERSITY, TAY NINH Mar 2026 – expected 2027

M.Sc. Computer Science

POSTS AND TELECOMMUNICATIONS INSTITUTE OF TECHNOLOGY, HCMC Sep 2017 – Jun 2022

Engineer's Degree (Bachelor of Engineering equivalent) in Information Security

CERTIFICATIONS

- TOEIC Listening & Reading, CEFR B1 (Intermediate), ETS, 2026