Nguyen Van Nhat Huy

Middle Full-stack Developer | Node.js | Next.js/React | PostgreSQL | AWS/Docker

(+84) 364 392 404 | nguyenvannhathuy.dev@gmail.com | linkedin.com/in/nhath240401 | HCM, Vietnam

Core Technologies

Languages: TypeScript, JavaScript, SQL

Backend: Node.js, Express.js, NestJS, RESTful APIs, WebSocket, Authentication, Authorization, RBAC

Frontend: React.js, Next.js, Vue.js, Tailwind CSS, HTML5/CSS3, State Management, Responsive Design

Database & Storage: PostgreSQL, Redis, MinIO, Schema Design, Indexing, Query Optimization, Transactions

Architecture & DevOps: Modular/Clean Architecture, Event-driven Processing, Caching, Docker, AWS, GitHub Actions, CI/CD, Linux

Testing & Tools: Jest, Postman, Git, Agile/Scrum, AI-Assisted Development (Antigravity, Codex)

Professional Experience

QKIT Software — **Ho Chi Minh City, Vietnam**

Full-stack Developer — *Feb 2024 – Apr 2026*

- Delivered end-to-end full-stack features across frontend, backend, database, and deployment layers using Node.js, Express.js/NestJS, Next.js, PostgreSQL, Docker, and AWS.
- Led a 5-member delivery group, including requirement breakdown, API contract design, task planning, code review, junior support, QA coordination, and release troubleshooting.
- Designed RESTful APIs, RBAC permissions, background jobs, real-time communication flows, and internal admin dashboards for healthcare, reporting, and business operation systems.
- Improved backend performance through PostgreSQL schema redesign, indexing, transaction optimization, Redis caching, and RabbitMQ-based asynchronous processing.
- Supported CI/CD deployment and production operations across AWS-based environments, including Dockerized services, environment configuration, release debugging, and production issue investigation.
- Leveraged AI-assisted development tools (Antigravity, Codex) to accelerate backend code generation, optimize complex PostgreSQL queries, and streamline production debugging.

QKIT Software — **Ho Chi Minh City, Vietnam**

Frontend Developer — *Feb 2023 – Jan 2024*

- Built and maintained production-facing UI modules and administrative dashboards using Vue.js, React.js, Tailwind CSS, and component-based architecture.
- Integrated complex frontend screens with REST APIs, including authentication flows, multi-step forms, searchable data tables, pagination, loading states, and error handling.
- Improved frontend maintainability by extracting reusable UI components, shared layout utilities, and generic API integration layers.
- Collaborated with backend developers and QA to identify API schema mismatches, reproduce UI bugs, validate acceptance criteria, and ship stable CRUD-heavy business features.

Projects

Electronic Medical Record System – Clinic & Patient Management Platform

Tech Stack: Node.js, Express.js, PostgreSQL, Redis, RabbitMQ, MinIO, Docker, AWS, WebSocket, RBAC

- Developed backend services for patient profiles, medical records, appointments, clinic workflows, user roles, document management, and internal administration.
- Designed multi-tenant RBAC for doctors, nurses, clinic admins, and staff users, enforcing strict tenant-level access control for sensitive medical data.
- Redesigned PostgreSQL schemas, indexes, and query patterns, improving medical record search and retrieval performance by approximately **40%**.
- Implemented RabbitMQ background workers for long-running tasks, reducing blocking operations in request-response workflows.
- Integrated MinIO object storage with secure upload and access flows for medical documents and attachments.

MarketLens – Automated Market Data Processing & Reporting Platform

Tech Stack: Node.js, PostgreSQL, Redis, RabbitMQ, Docker, GitHub Actions, AWS ECS, Next.js

- Developed backend services for an automated reporting platform that processes market analytics data from multiple external sources and generates business-ready reports.
- Improved reporting performance and stability by optimizing PostgreSQL batch operations, transaction boundaries, indexing strategies, and query execution patterns.
- Designed asynchronous processing flows with RabbitMQ workers to handle long-running data aggregation, report generation, retries, and failed-task recovery outside request-response cycles.
- Improved operational visibility by exposing processing status, failed-job reasons, retry history, and export status through internal admin APIs and dashboard views.
- Automated Docker-based deployments to AWS ECS using GitHub Actions, improving deployment consistency across development, staging, and production environments.

CaseFlow – Support Ticket & SLA Tracking Platform
Tech Stack: Node.js, Express.js, PostgreSQL, Redis, RabbitMQ, MinIO, Docker, WebSocket, RBAC

- Centralized customer issues, internal service requests, ticket ownership, SLA deadlines, comments, attachments, and resolution history into a custom operation platform for a client.
- Replaced spreadsheet/chat-based tracking workflows with structured ticket lifecycle management, including priority levels, status transitions, assignee changes, escalation rules, and audit-friendly activity history.
- Implemented transactional updates in PostgreSQL to keep ticket status, assignment history, SLA records, audit logs, attachment metadata, and notification events consistent during concurrent operations.
- Optimized ticket queue performance with PostgreSQL indexes and Redis caching for advanced filtering, keyword search, unread counters, SLA views, and dashboard summary data.
- Used RabbitMQ workers for SLA reminders, escalation notifications, email delivery, delayed jobs, attachment processing, and retry handling for failed background tasks.
- Built internal admin APIs and dashboard features for support queues, SLA monitoring, agent workload tracking, overdue ticket follow-up, and audit trail review.

Education

VNUHCM – University of Information Technology (UIT) **Ho Chi Minh City, Vietnam**
Bachelor of Information Technology *2019 – 2023*

Certifications

- TOEIC 700+