




Le Huu Huy

 github.com/lhhuy02012005  [Portfolio](#)  hhuy.2005@gmail.com

EDUCATION

Sai Gon University

Step.2023 - June 2027

B.S. Computer Science

Current GPA: 3.16/4.0

SKILLS

Languages: Java, PHP, Python, JavaScript/TypeScript, C/C++, C#, HTML/CSS

Frameworks & Libraries: Spring Boot, Spring Data JPA, Hibernate, MapStruct, Laravel, React 19, FastAPI, LangChain

Security & IAM: Spring Security, FusionAuth, OAuth2 (Google Login), JWT

Databases & Caching: PostgreSQL, MySQL, Redis, ChromaDB, NoSQL

Architecture & Tools: Docker, Git/GitHub, Unix Shell, Maven, Postman, WebSocket, Cloudinary

PROJECTS

ShopZues — Scalable E-commerce Platform | *Spring boot, React, Firebase, MySQL, Redis*

Nov 2023

- Implemented a scalable Spring boot backend with JWT auth, RBAC (role/permission), and Redis (cache + queue + OTP).
- Integrated Firebase Realtime DB for live order status (State Pattern) & notifications, and FusionAuth for OTP email verification.
- Developed React frontend + Flutter mobile app that connect to backend APIs and cloud DB (Firebase + MySQL).
- Used Docker + docker-compose for consistent dev/test environments and easy deployment.
- Learned end-to-end full-stack architecture: auth flows, real-time sync, payment flow.

RAGAI (NexusRAG) — Hybrid Knowledge Base & Agentic Chat

2026

- **Tech Stack:** *FastAPI, React 19, TypeScript, PostgreSQL, ChromaDB, LightRAG, Ollama, Sentence-Transformers, Docker*
- Designed an end-to-end local-first RAG architecture: document upload → Docling parsing → chunking/dedup → embedding → vector + knowledge-graph indexing → hybrid retrieval → grounded generation.
- Implemented a hybrid retrieval pipeline combining ChromaDB vector search and LightRAG graph context, then improved precision with cross-encoder reranking (**BAAI/bge-reranker-v2-m3**).
- Built agentic chat with tool-calling (**search_documents**) and SSE streaming, delivering real-time events for thinking, tokens, sources, images, and final answer.
- Developed citation-first response format with source traceability (document/page/heading path), enabling explainable answers and quick navigation back to evidence.
- Added visual document intelligence: extracted images/tables from PDFs, generated captions, and injected media semantics into chunk embeddings for better retrieval quality.
- Engineered robust document lifecycle handling (PARSING/INDEXING/INDEXED/FAILED), timeout recovery, and async background processing for long-running ingestion tasks.
- Implemented workspace isolation across PostgreSQL metadata, vector collections, and knowledge-graph storage, supporting multi-tenant knowledge bases safely.
- Containerized infrastructure with Docker Compose (PostgreSQL + ChromaDB) and tuned runtime configs (chunk size, prefetch, rerank top-k, KG extraction limits) for both medium and high-spec machines.

CryptionChat (ShieldTalk) — Secure End-to-End Encrypted Messaging Platform

2026

- **Tech Stack:** *Spring Boot 3, Java 17, Spring Security 6, PostgreSQL, MongoDB, Redis, JWT (Nimbus), RSA/AES/BCrypt, Docker*
- Engineered a high-security messaging system featuring **End-to-End Encryption (E2EE)** using a hybrid RSA-2048 and AES-256-GCM architecture to ensure zero-knowledge data privacy.

- Implemented a robust Authentication & Authorization flow with **JWT (Nimbus JOSE)**, incorporating a **Token Versioning** strategy to enable instant remote logout and session invalidation upon security events.
- Developed a secure key management lifecycle: client-side RSA key pair generation, encrypted private key storage (AES-wrapped), and public key exchange via a centralized discovery service.
- Architected a scalable **Hybrid Database** system: PostgreSQL for relational metadata (User/Roles/Friendships) and MongoDB for high-volume, encrypted chat message persistence.
- Optimized discovery and social features with **PostgreSQL GIN Indexes** and Trigram search for sub-millisecond full-name lookups across large user bases.
- Built a high-performance friendship management system featuring pagination
- Integrated **Spring Security 6** with a custom Resource Server configuration, implementing Role-Based Access Control (RBAC) and Cross-Origin Resource Sharing (CORS) policies for secure API consumption.
- Containerized the entire ecosystem (App, Postgres, Mongo, Redis) using Docker, ensuring environment parity and streamlined deployment pipelines.

EXPERIENCE

Sai Gon University

Ho Chi Minh City, Viet Nam

Software Engineering Student

Sept. 2023 – Present

- Developed high-performance web systems using **Laravel** and **Spring Boot**, focusing on Service-Repository patterns and clean architecture.
- Architected secure authentication systems using **JWT**, **OAuth 2.0**, and **Spring Security**, managing stateless sessions and role-based access control (RBAC).
- Optimized application performance by implementing **Redis** for distributed caching and asynchronous background tasks (queues), reducing database load during high-traffic simulations.
- Integrated third-party ecosystems including **VNPAY** for payment processing, **GHN** for logistics automation, and **Firebase** for real-time data synchronization.
- Containerized applications with **Docker** and orchestrated development environments using Docker Compose to ensure cross-platform consistency.
- Collaborated on version control via **Git/GitHub**, managing complex branching workflows and maintaining codebase integrity through code reviews.

HOBBIES

Music & Arts: **Playing badminton** , Gym , Swimming.

Tech Exploration: Researching **Generative AI (RAG)** and local LLM deployment; building custom mechanical keyboards.