

Le Hung Thien

Golang Backend Developer Intern

lethien.dev@gmail.com | lethien.dev | linkedin.com/in/lehungthien

Technologies

Languages: Go, JavaScript

Backend: Go (standard lib, httprouter), JWT, bcrypt, PostgreSQL, MySQL

Frontend: ReactJS, SCSS Modules

Tools: Linux (CLI), Git, Docker, Postman, Make

Architecture: RESTful APIs, Clean Architecture, Microservices (basic)

Education

FPT University, BS in Software Engineering

Sep 2023 – Apr 2027

- **GPA:** 3.5/4.0; awarded Honorable Student of Semester
- **Scholarship:** 50% tuition scholarship for the full duration of the degree
- **Certificates:** IELTS 6.0 Overall (2023), CS50x: CS50's Introduction to Computer Science

Projects

Greenlight – Go REST API for Movie Database

github.com/thienel/greenlight

Movie backend with authentication, access control, and production features.

- Built scalable Go REST API with **PostgreSQL** and **httprouter**, supporting full CRUD operations.
- Implemented token-based authentication with user registration, email activation, **bcrypt** password hashing and permission-based access control.
- Integrated production features: rate limiting, CORS, JSON logging, input validation, metrics, and SMTP email.
- Developed advanced querying with filtering, pagination, comprehensive tests, and optimized database pooling.

TikTok Clone – Full-Stack Social Media Platform

github.com/thienel/

Full-stack clone featuring .NET 9 backend, secure auth, and responsive React UI.

- Built **React** with SCSS modules, Context API state management, and custom hooks for API integration.
- Developed **.NET 9** backend API with **Clean Architecture**, Entity Framework Core, SQL Server, and **JWT authentication**.
- Implemented full authentication flow: user registration, email verification, password reset, and username validation with real-time availability checking.

Snippetbox – Go Web Application

github.com/thienel/snippetbox

A simple web app for managing code snippets with expiration and session-based auth.

- Developed in **Go** with **httprouter** and **alice** for routing and middleware chaining.
- Secured authentication using **bcrypt** password hashing and secure session cookies via **scs/mysqlstore**, with CSRF protection.
- Used prepared SQL statements to prevent injection and handled user input via **form/v4**.
- Created modular HTML templating system with embedded static assets and reusable layout components.

Personal Portfolio

www.lethien.dev/portfolio

Personal portfolio site built with React, TypeScript, and WebGL.

Relevant Readings

Selected readings that shaped my understanding of Go and scalable backend systems:

- **Let's Go** by Alex Edwards – Comprehensive guide to building web applications with Go
- **Let's Go Further** by Alex Edwards – Advanced Go web development techniques and best practices
- **Building Microservices** by Sam Newman – Design patterns and principles for microservices architecture