



# Résumé

You can also download this page as PDF instead (Experimental). [Download PDF](#)

## Summary

Worked as a full-stack developer and DevOps/InfraOps Engineer at many companies. Have led growth at development departments in many start-up teams as lead engineer.

Love to contribute to open sources and tech communities by sharing knowledge and experience.

Interested in devising better problem-solving methods for challenging tasks, learning new technologies and tools, and working to create a better world.

## Work Experience

### **NGuard Security Project (Extension bot to improve safety in Discord) (Dec. 2021 ~ Present)**

- As lead engineer, Worked on DevOps, InfraOps and Core developer for this project.
- Designed and provisioned the entire infrastructure on the Oracle cloud infrastructure to meet security compliance and make it easier to manage/scale.
  - Continuously improved the infrastructure architecture since launching the service.
- Established a core architecture for regulating outbound DNS traffic utilizing Cloudflare's DNS Firewall(WAF) product.
  - In addition, 'tunnelling' technology ensures that the service can be operated stably and with high security even under strong regulations regarding the ingress firewall of VM instances.
- Introduced Zero-trust member identity solution to the team, established security policies, and configured SSO integration with over 20 internal systems.
- Rewrote the core API server/infrastructure to ensure the security and fast speed of processing requests.

- Using the Clean Architecture structure, redundant code was reduced, and the code was written to run more efficiently.
- In the rewriting process, the API library was changed from Koa.js to Nest.js, which supports high/strict typing and faster processing. This resolved potential security issues and improved responses.

## **Kyunggi High School - Co-curricular(aka club) activities online selection system (Feb. 2024 ~ Mar. 2024)**

- As lead engineer, Worked on DevOps, InfraOps and core developer for this project.
- Designed and provisioned the entire infrastructure on the AWS Cloud (Korea) to make a stable web service.
  - Continuously improved the infrastructure architecture since first launching the service.
  - Used EC2 for the database(MongoDB) and API server and Amplify for the frontend, which was made with Next.js SSR.
  - Used ALB(Application Elastic-load balancer) to balance load level for API for a first-come-first-served selection period.
- Rewrote core API and frontend infrastructure to minimize operation costs and improve user experience by processing requests quickly.
  - Using the Clean Architecture structure, redundant code was reduced, and the code was written to run more efficiently.
- The development period was a very short work period of two weeks, but worked hard to run this project stably and smoothly, making this project a success.

## **Netpy Cloud Project (Infrastructure/server hosting service) (Jan. 2024 ~ Present)**

- As a full-stack engineer, Worked as a core developer for this project.
- Made some key API services with Nest.js, Go, and Kotlin to improve user experience by processing requests quickly with an efficient way to run.
  - Core API — built with Nest.js.
    - To make the code as clean and efficient as possible, the code written with Next.js API Router was rewritten using clean architecture and domain-driven development. As a result, redundant code was reduced, and the code was written to run more efficiently.
  - File Server API — built with Go (Gin framework).
    - When uploading and downloading files, it is very important to check whether there is any malicious code or virus inside the file, so we have configured it to detect viruses and operate a

safe server using a virtualization environment.

- Due to the system structure, which has a Micro-service architecture and other APIs, gRPC and HTTP/2 communication (described below) were actively used for high speed and high security.
- gRPC API — built with Kotlin.
- Due to the API system structure with a micro-service architecture, an API for gRPC and HTTP/2 communication was developed for high speed and high security. This API is designed to access and read/write common database tables used by several other APIs and operates on the server where the database is installed.
- This made it possible to perform business logic more safely by opening only the internal network firewall for gRPC's HTTP/2 communication, without opening the firewall of the database and other servers.

## **Schooler.kr Project (Korean-style school-management information system, aka MIS)**

**(Dec. 2023 ~ Present)**

- This project is still in progress. It will be released later.

## **VenturOX Project (Wheelchair with extended features for the disabled)**

**(Dec. 2023 ~ Present)**

- This project is still in progress. It will be released later.

*Please note that projects/teams not on the list here may not be disclosed due to contractual issues or may have been worked on for a long time and are inappropriate for posting here.*

## **Honours & Awards**

*This section is still updating and will be released later.*

## **Education**

- *Sep. 2022 ~ Present: British International School, Hanoi (BIS Hanoi) — Hanoi, Vietnam.*
- *Mar. 2021 ~ Sep. 2022: Suwon-dasan Middle School — Suwon-si, Gyunggi-do, Korea.*