

Curriculum Vitae

HyungRyul Kim

updated: @Apr 23, 2020

HYPERCONNECT

Full-time, Seoul, Republic of Korea – 2019.11 ~ Present

AZAR API Team Backend Software Engineer 2019.11 ~

AZAR Match Squad Team Backend Software Engineer 2020.03 ~

- **Changed to 3 tabs & enhanced match** 2020.03 ~ 2020.04

A project to enhance the user's card match experience and provide an improved user UX as one of the AZAR Match Squad tasks.

Technology used : Kotlin, Java, Spring Boot, JPA, MySQL, Docker, DynamoDB, Redis etc

- ✓ Major policy consultation and decision-making to strengthen matches
- ✓ Additional API implementation by UI design reorganization
- ✓ Data management and push message implementation when matching users
- ✓ Micro service development and maintenance work
- ✓ Data paging processing using DynamoDB global secondary index
- ✓ Match profile cards can be reused for a certain period of time through redis caching

- **Config Store microservice & annotation client library** 2019.12 ~ 2020.02

Development of annotation library and micro service in charge of DB access that can inject and change field values of running JVM-based applications without restarting the server

Technology used : Kotlin, Java, Spring Boot, JPA, MySQL, Docker, Reflection, Redis etc

- ✓ Development and distribution of annotation library that can be injected and changed when the values of private fields of the application that is

running using reflection are desired.

- ✓ For security, the micro service in charge of DB access is operated and the library receives the latest value of the DB through an api call.
- ✓ Improve performance by applying Coroutines
- ✓ Development of a library that periodically updates values in the DB after registering a lambda expression that extracts fields declared annotation using BeanPostProcessor and updates them in the field using reflection.
- ✓ Event triggering with redis pub / sub when immediate update is required
- ✓ Respond to all types through type inference
- ✓ Support 3 types of data structure: Map, List, Set

- **Sign up with Apple** *2020.03 ~ 2020.04*

Support Apple login according to Apple's new SNS login policy

Technology used : Java, Spring Boot, JPA, MySQL, Docker, Redis etc

- ✓ Apple login added to iOS app according to Apple policy
- ✓ After real server distribution, 28% of all iOS users use Apple login

- **Match Publisher microservice** *2020.03 ~ 2020.04*

Production of micro service that sends matching user data from Discover Server to WebRTC Broker Server

Technology used : Kotlin, Spring Boot, JPA, MySQL, Docker, Redis etc

- ✓ Analyzing existing legacy codes, considering compatibility with old versions, and producing with Kotlin
- ✓ Asynchronous transmission implementation using WebClient

NHN

Intern, Pangyo, Republic of Korea – 2019.01 ~ 2019.06

Dooray R&D Team Backend Software Developer 2019.01 ~ 2019.06

- **Dooray Docs Co-Editor prototype** *2019.03 ~ 2019.06*

Co-editing editor module development

Technology used : Java, JavaScript, Spring Boot, JPA, WebSocket, MySQL, Event Sourcing Pattern, Redis etc

- ✓ Development of co-editing module applicable to NHN Toast Editor
- ✓ Editor cursor related processing with JavaScript input event
- ✓ Real-time change applied by using WebSocket
- ✓ Concurrency processing using Event Sourcing Pattern
- ✓ Event Stream data management using Redis Stream

- **Dooray messenger Built-in slash commands** 2019.01 ~ 2019.02

Built-in voting, scrum, first-come-first-served slash command built into dooray messenger

Technology used : Java, Spring Boot, MySQL, NGINX, SonarQube, Gradle, Git, Junit etc

- ✓ MSA style development using Java / Spring Boot
- ✓ Using JPA and MySQL
- ✓ Real service distribution operation and testing

CNTEXTILE

Freelancer, Seoul, Republic of Korea – 2018.10 ~ 2018.12

- **Internal ERP System** 2018.10 ~ 2018.12

Development of a homepage that issues goods management and invoices in-house

Technology used : Python, HTML5, JavaScript, Django, AWS, Linux, NGINX, Docker, Jenkins, MySQL etc

- ✓ Build the company's internal ERP homepage and system using the Django template engine.
- ✓ Deploy automation pipeline with Docker and Jenkins
- ✓ Daily document printing function
- ✓ Distribution using NGINX, Uwsgi