

---

## 연구민

lourie@naver.com

606-8205

Japan Kyotofu Kyotoshi  
Sakyouku Tanaka  
Kamiyanagicho-39  
T-house 206

깃허브: [https://  
github.com/  
voidsatisfaction](https://github.com/voidsatisfaction)

### Profile

안녕하세요. 저는 프로그래밍을 이용해서 문제를 해결하는 것을 좋아하는 연구민 입니다. 저는 프로그래밍의 본질은 문제 해결이라고 생각합니다. 단순히 코드를 작성하는 것을 넘어서, 실제 생활에서의 문제를 해결하기 위한 무수한 방법 중 하나의 선택지라고 생각하며, 그 가능성은 무한하다고 생각합니다.

위와 같은 사상을 바탕으로 저는, 저 자신의 개인적인 문제들 뿐 아니라, 다양한 사람들의 문제들을 해결하기 위한 프로그래밍을 해왔습니다. 학교 사이트의 좋지 않은 UX 를 개선하고, 학생간의 과목별 평가를 공유하기 위한 School Share 애플리케이션 개발, 매일 자신이 공부한 내용을 복습하게 도와줄 수 있는 Repo-watcher의 개발, 보다 정확하고 감정에 의존하지 않은 트레이딩을 위한 자동 거래 시스템의 개발 등등.. 저의 모든 개발의 흔적은 모두 “문제 해결”에 초점이 맞추어져 있습니다.

저의 꿈은 이러한 문제해결을 통하여 세상의 많은 사람들을 행복하게 하는 것입니다. 그리고 항상 도전하고 싶습니다. 그리고 도전을 통해서 저 자신이 성장한다는 것을 언제나 느끼는 것을 좋아하고 앞으로도 그런 인생을 살아갈 것입니다.

### What I like

- 배움과 적용
- 함수형 프로그래밍 패러다임
- 좋은 아키텍처를 보고 배우고 적용하는 것
- 오픈소스
- 유닉스 사상
- 알고리즘 문제풀기(백준 온라인 저지 등)
- 아무런 생각 없이 주어진 미션을 하는 것이 아니라, 내가 주인의식을 갖고 미션을 찾아서 그것을 달성하는 것
- 고퍼

### Experience

[정규] 주식회사 하테나 — 2017.12 ~ 현재(프론트 / 백 웹 엔지니어)

주요 기술 키워드: Scala, TypeScript, Scss, Perl, MySQL, Docker, Domain Driven Development, Clean Architecture, Cake Pattern, Agile Scrum

현재 하테나 북마크 서비스(<http://b.hatena.ne.jp/>) 의 주 3일 풀타임 엔지니어로 근무하고 있습니다. 웹 프론트엔드(TypeScript), 웹 프론트 서버(Perl), 웹 코어 서버(Scala)를 넘나드는 개발을 하고 있습니다. SEO대책의 일환으로 관리자가, 어떠한 북마크가 적절하지 못하다고

---

---

판단되면 nofollow를 링크에 추가할 수 있도록 데이터베이스를 설계, 서버사이드, 클라이언트 사이드 구현을 하였습니다. 또한, 최대한 크롤러 봇이 많은 북마크 링크에 도달할 수 있도록, 북마크 사이트 하단에 이전 북마크와 이후 북마크를 배치하는 것도 설계부터 구현까지 담당하였습니다. 현재는 레거시 코드(Perl)에 있는 북마크 제목 수정 기능을 스칼라 코드로 옮기는 작업을 하고 있습니다. 모든 개발 프로세스는 애자일 프로세스로 관리되고 있습니다.

#### [개인] 가상화폐 자동거래 시스템 개발 — 2018.3 ~ 현재

주요 기술 키워드: TypeScript, Javascript, Nodejs, React, AWS MariaDB, AWS Lambda, AWS Lightsail, Serverless, Chat Ops

친구와 함께 가상화폐 자동 거래 시스템을 AWS플랫폼과 TypeScript를 이용해서 구현했습니다. 가상화폐 자동거래 시스템은, 외부의 서비스에 크게 의존하고, 금전적인 거래를 다루기 때문에 어느곳에서 에러가 발생했고, 왜 발생했는지 신속하게 알 수 있게, 또한, 에러 발생시 즉시 대응할 수 있도록, 도메인 로직을 구성하는 컴포넌트간의 책임을 가능한 분리를 해야 한다고 판단했습니다. 그래서 AWS-lambda를 이용해, Microservice 형태로 구성하였습니다. 거래소 별 가격을 그래프화 하기 위해서 react라이브러리(프레임워크)와 react-stockcharts 라이브러리를 이용해서 구현했습니다. 또한 에러는 개발자가 즉시 알 수 있도록, slack bot을 활용해서 바로 통지가 올 수 있도록 구성했습니다.

#### [인턴] CyberAgent 개발인턴(1달) - 2017.10 ~ 2017.11

주요 기술 키워드: Golang, Revel, Jenkins, Mysql, Agile Scrum, 이미지 처리, Jira, Confluence, DDD, TDD

AmebaOwnd(<https://www.amebaownd.com/>) 라는 서비스의 서버 개발을 담당했습니다. 모바일 애플리케이션 리뷰얼에 맞춰서 새로운 엔드포인트의 개발을 담당했습니다. 특히, TDD의 사상을 바탕으로 엄밀하게 테스트를 작성하고, JSON Schema역시 작성해서 작성한 프로그램의 무결성을 보장하기 위해서 노력했습니다. 또한, 이미지 데이터의 orientation을 바로잡아주는 기능을 구현하기도 했습니다.

#### [인턴] Indeed global tour(1주일) - 2017.9

Indeed본사에서 초청을 받아, 미국 텍사스를 일주일 동안 시찰하는 시간을 갖았습니다. 세계 최고의 잡 서치엔진 기업인 Indeed의 내부 구성과, 철학, 의사결정 시스템을 알게되어 매우 뜻 깊은 시간이었습니다. 특히, "data is king"이라는 말이 인상깊었습니다. 모든 의사 결정의 기반은 데이터라는 말인데, 나 자신도 단순히 직감 뿐 아니라, 데이터 라는 객관적인 evidence를 갖고 개발을 해나가는 개발자가 되어야 겠다고 생각했습니다.

#### [인턴] 주식회사 Pixiv 개발인턴(1주일) — 2017.9

주요 기술 키워드: React, Nodejs, Ruby On Rails, Open Source

Pixiv에서는 Pawoo(<https://pawoo.net/about>)라는 SNS 서비스를 운영하고 있는데, 그 모체가 Mastodon(<https://github.com/tootsuite/mastodon>)이라는 오픈소스로 구성되어 있

---

---

습니다. 인턴 기간중에 Mastodon의 이슈를 직접 해결하는 오픈소스 기여활동을 했습니다. 고정된 톱 이라는 새로운 기능을 추가했습니다. 이 인턴을 참가하면서, 오픈소스로 커 왔던 자신 역시, 오픈소스에 기여할 수 있다는 용기를 얻어 다양한 오픈소스 활동에 참가하게 된 계기가 되었습니다.

#### [인턴] Fringe81 개발인턴(2주일) - 2017.8

주요 기술 키워드: React-Native, DDD, 아키텍처

React-Native를 이용해서 애플리케이션을 만드는 한편, DDD의 아이디어를 가지고 “어떻게 하면 좋은 프론트엔드 애플리케이션을 만들 수 있는가?”에 대한 테마를 메인으로 한 인턴입니다. 결과물은 React-Native앱을 이용한 하이브리드 애플리케이션 입니다. 무엇보다, DDD를 바탕으로 앞으로 “단순 개발 뿐 아니라, 좋은 아키텍처를 설계할 수 있는 아키텍트가 되어야겠다” 라는 꿈을 심어준 인턴이었습니다.

#### [인턴] DeNA 개발인턴(3일) - 2017.8

주요 기술 키워드: Nodejs, 기획, 비즈니스 모델

사원1명과 개발자 학생3명이 다함께 3일동안 하나의 서비스를 완성하는 해커톤 형식의 인턴입니다. DeNA는 요코하마 DeNA라는 구단을 소유하는 기업인데, 해당 기업에서 활용 가능한 스포츠 서비스를 개발한다는 컨셉을 갖고 개발을 했습니다. 제가 담당하는 부분은 서버 사이드였으며, nodejs를 이용해서 도메인로직을 구현하였습니다. 아쉽게도 우승은 하지 못했으나, 단순히 프로그래밍을 공부하는 것 뿐 아니라, “서비스의 본질은 유저의 니즈다” 라는 생각을 갖게 해준 값진 경험이었습니다.

#### [인턴] Recruit 개발인턴(1달) - 2017.2~2017.3

주요 기술 키워드: React, Ruby On Rails, Scss, 애자일 스크럼

React와 Ruby On Rails를 활용해서 회사 내부에서 사용하고 있는 웹 사이트를 쉽게 만들어주는 웹 애플리케이션의 기능을 추가하고 이슈를 해결하는 일을 했습니다. 개인적으로 가장 큰 기여라고 생각하는 부분은, 원래 모바일 버전과 컴퓨터 버전을 따로 생성해야하는 기존의 버전에서 Media query를 도입함으로써, 하나의 웹 사이트를 만들어도 어떠한 디바이스에서도 적절한 디자인이 되도록 한 것입니다. 또한, 내부의 redux로직을 보다 알기 쉽게 변경하는 일도 담당했습니다.

#### [개인] School Share 애플리케이션 개발(6달) - 2016.6 ~ 2017.12

주요 기술 키워드: React-Native, Nodejs, 애자일 칸반, travis ci

저와 친구 둘이서 당시 최신 기술이었던 React-Native를 활용하여 School-Share라는 애플리케이션을 만들었습니다([https://play.google.com/store/apps/details?id=com.school\\_share](https://play.google.com/store/apps/details?id=com.school_share)) 이 프로젝트가 저의 첫 프로젝트였으므로 지식이 부족하여, 초반 아키텍처나 기반 코드는 친구가 작성하고, 저는 제가 할 수 있는 부분(디자인) 부터 활동 영역을 넓혀갔습니다. 처음에는 React, Redux와 같은 개념이 너무나도 생소하고 알기 힘들

---

---

었으나, 포기하지 않고 차근차근 이해하여 나중에는 제 자신이 State관리를 위한 코드 뿐 아니라, 테스트 코드, 테스트 서버 구성까지 담당하게 되었습니다. 포기하지 않으면 언젠가는 반드시 된다는 중요한 교훈을 알게 해준 뜻깊은 개발 경험이었습니니다.

## Open Source

오픈 소스에 관한 내용은

<https://github.com/voidsatisfaction/TIL/tree/master/Opensource>

이쪽에 정리해 두었습니다.

## Personal Project

TIL(Today I Learned)

저는 매일매일 공부한 내용을 TIL(<https://github.com/voidsatisfaction/TIL>) 에 저장해두고, 제가 직접만든 복습 웹 애플리케이션(<https://github.com/voidsatisfaction/Repo-watcher>)를 이용해서 매일마다 복습하고 있습니다. TIL은 1년 반 이상을 계속해서 유지/갱신하고 있습니다. 컨퍼런스에 다녀와서 얻은 교훈이나 참고가 되는 사항들(e.g [https://github.com/voidsatisfaction/TIL/blob/master/%40conference/2018/0404\\_cheol.md](https://github.com/voidsatisfaction/TIL/blob/master/%40conference/2018/0404_cheol.md)) 뿐 아니라, 자신이 공부했던것들(e.g <https://github.com/voidsatisfaction/TIL/blob/master/Architect/Architecture/README.md>)도 작성해두고 있습니다.

그 외에 다양한 개인 프로젝트들을 깃허브에 업로드 해 두었습니다.

## Education

일본 정부 국비유학생(2012 ~ 2017)

동경외국어대학 예비교육 졸업(2012 ~ 2013)

교토대학 경제학부 졸업(2013 ~ 2017)

교토대학 정보학연구과 사회정보학(2017 ~ 2019 졸업 예정)

프로그래밍은 대학교 3학년 후반기부터 시작했습니다(2018년 현재 3년차)

## Skills

언어(가장 잘 활용하는 순)

- JavaScript(Nodejs)
  - Scala
  - Golang
  - TypeScript
  - Ruby
  - Python
  - C++(알고리즘 풀이 전용)
  - 정규표현식
-

- 
- Perl
  - Scheme(취미)

#### 프론트엔드

- JavaScript(ECMA Script 2015) / TypeScript
- VanillaJs MVC
- React
- React-native
- Babel
- Webpack
- Ramdajs

#### 백엔드

- Scala
- Nodejs Express
- Ruby On Rails(사실 레일에 오르는 것을 그렇게 좋아하진 않습니다)
- SQL
- Docker

#### 아키텍처

- 도메인 주도 개발(DDD)
- 클린 아키텍처
- Cake pattern

#### 인프라

- AWS EC2
  - AWS Lightsail(개인 취미 공간)
  - AWS Lambda
  - AWS RDS
  - AWS VPC
  - AWS S3
-