

김현우 | Web Developer

Spring, React 기반의 웹 개발을 좋아합니다.

꼼꼼한 **테스트**를 통해 안정성을 추구하고자 합니다.

문제 해결에 필요한 기술이라면, **빠르게 학습하고 적용**합니다.

개발자는 코드로 이야기하지만, 결국 **사람**과 함께 일합니다.

모든 코드는 사람이 **읽기 좋아야** 합니다.

페어 프로그래밍과 코드리뷰같은 **공유, 협업 문화**에 익숙합니다.

우아한테크코스 백엔드 과정 중 프론트엔드 분야에 흥미를 가지고 공부를 시작했습니다.

최근에는 TDD 기반의 React 개발에 관심을 가지고 공부하고 있습니다.

SKILLS

BackEnd Java8, JUnit5, Spring Boot, Spring Data JPA

FrontEnd JavaScript(ES6), React, Vue2, Jest

Infra MySQL, AWS EC2, Jenkins, Nginx, Ubuntu

Phone 010-3020-9754

E-mail gusdn9754@gmail.com

Github github.com/hwookim

Education

동국대학교(서울)

2016.03 ~ (휴학)

멀티미디어공학과 2학년 수료

한국디지털미디어고등학교

2013.03 ~ 2016.02

웹프로그래밍학과 졸업

About Me

변경에 유연하고 안정성 있는 코드를 작성하기 위해 노력합니다.

TDD 사이클의 개발을 자연스럽게 익혀왔습니다.

웹 개발을 처음 시작할 때부터 테스트가 있는 코드를 접했습니다.
테스트를 통해 하나의 기술 명세를 제공하고, 리팩토링 시 더욱 안전하게 사이드 이펙트를 관리할 수 있도록 합니다.

통일성 있고, 직관성 있는 코드를 구현하고자 노력합니다.

컴포넌트의 역할과 책임을 고민하고 적절하게 나누려고 노력합니다.

역할이 한 눈에 보이는 네이밍을 위해 늘 고민합니다.

프로젝트 진행 시 일관된 코드를 작성하기 위해 회의를 통해 **팀 컨벤션**을 정했습니다.
팀 전원이 코드리뷰에 참여하면서 쉽게 이해되지 않는 코드는 더 직관성 있게 바꿀 수 있도록 논의하는 과정을 매번 거쳤습니다.

관련 PR(<https://bit.ly/3kGk08m>)에서 Vue 코드에 대한 팀 컨벤션을 확인하실 수 있습니다.

다양한 방법으로 의견을 공유하고, 토론하는 것을 즐깁니다.

자신의 생각과 경험을 글로 공유합니다.

우아한테크코스에서 매 레벨이 마무리될 때마다 **회고의 글쓰기**를 진행했습니다.
그 중 팀프로젝트 회고에 대한 글은 **우수 글**로 선정, 우아한형제들 기술 블로그에도 기고되었습니다.

- 우아한형제들 기술 블로그 기고 글 : <https://bit.ly/3s6h61A>

우아한테크코스 미션 코드리뷰에서 학습한 내용에 대해 글로 작성하고 공유합니다.
그 외에 여러 의견이 나올 수 있는 상황에 대한 자신의 생각이나 직접 경험했던 트러블 슈팅, 개발 시 도움이 되었던 방법론 등에 대해서도 정리해 공유했습니다.

- 의견이 갈리는 분야에 대한 자신의 생각 공유
Custom Exception, 언제 써야할까? : <https://bit.ly/3np2BTk>
상태코드, 뭘 줘야할까? : <https://bit.ly/38MGD8J>
- 프로젝트 도중 경험한 문제 해결 상황을 공유
프로퍼티 파일을 활용해 쉽게 실행 환경 분리하기 : <https://bit.ly/36Huj78>

PROJECT

도란 도란 (2020.07 ~ 2020.11)

지도 위에서 주위 사람들과 소통할 수 있는 위치 기반 커뮤니티 서비스



도란도란은 지도 위 특정 위치에 글을 써 주위 사람과 소통할 수 있도록 하는 커뮤니티 서비스입니다.

글을 작성함으로 자신의 이야기를 전하거나, 특정 장소에 대한 질문을 할 수 있는 소통 창구를 제공합니다.

댓글 작성자와 지도 상 글의 위치 간의 거리를 표시해 소통 시 신뢰도와 친밀감을 형성할 수 있도록 했습니다.

 github.com/woowacourse-teams/2020-doran-doran

기여

- Vue.js 기반 프론트 페이지 구현
 - Vue Style Guide를 참고한 팀 컨벤션 정리
 - KakaoMap API 사용을 위한 Vue.js 플러그인 구현
 - vue-router를 통한 모달창 관리
- Git Flow, Github Kanban을 통한 프로젝트 관리
- 코드리뷰를 통한 코드 병합 관리
- Spring Boot 기반 API 설계 및 구현
 - 댓글 CRD 구현
- 인프라 환경 구축
 - Nginx를 통한 정적 자원 제공 및 캐싱
 - Jenkins를 통한 무중단 CI/CD 환경 구축

기술 스택

BackEnd : Java8, JUnit5, Spring Boot 2.3, Spring Data JPA, QueryDSL

FrontEnd : JavaScript (ES6), Vue.js 2, Vuetify, Axios

Infra : Maria DB, AWS EC2, Jenkins, Nginx, Git

PROJECT

도란 도란 (2020.07 ~ 2020.11)

 github.com/woowacourse-teams/2020-doran-doran

KakaoMap API를 위한 Vue.js 플러그인 구현

API 사용을 위해 script헤더를 동적으로 생성하고, 생성이 확인된 이후에 객체를 호출해야하는 문제가 있었습니다. 더불어 한 컴포넌트에서만 사용하는 것이 아니라 전역적인 API 메소드 사용이 예상되기도 했습니다.

전역적 사용을 위해 vuex를 통한 상태관리도 고려했으나, script헤더의 동적 생성 시점이 문제되었습니다.

Vue 플러그인은 Vue Application이 생성될 때 로드하기 때문에 컴포넌트가 생성되기 전에 script를 생성할 수 있는 동시에 전역적으로 사용이 가능했기 때문에 해당 문제에 적합한 해결 방법이었습니다.

더 자세한 문제 해결 과정은 팀 위키(<https://bit.ly/3nJEmzx>)에서 보실 수 있습니다.

코드리뷰를 통한 코드 병합 관리

팀 전원이 코드리뷰에 참여했습니다.

다양한 관점에서의 리뷰로 코드 품질 향상과 동시에 전체 코드에 대한 이해도를 높였습니다.

Jenkins를 통한 CI/CD 환경 구축

Jenkins의 Publish over SSH 기능과 Script를 통해 **Blue-Green 방식의 무중단 배포 환경**을 구축했습니다.

Githook을 통해 push가 발생하는 순간 Jenkins에서 알림을 받고 저장소를 패치합니다. health-checking을 통해 현재 실행되고 있는 서버 포트를 파악하고 저장합니다.

저장소의 코드를 빌드합니다.

프론트 파일의 경우 실행 중인 포트에 따라 api 호스트를 변경합니다.

SSH를 통해 배포 서버로 빌드된 파일과 함께 현재 실행할 .sh파일을 전달합니다. jar 파일을 실행하고 정상 작동하는지 health-checking 합니다.

이전 포트에 실행되고 있는 서비스를 종료합니다. 이로써 무중단 배포가 무사히 진행되었습니다.

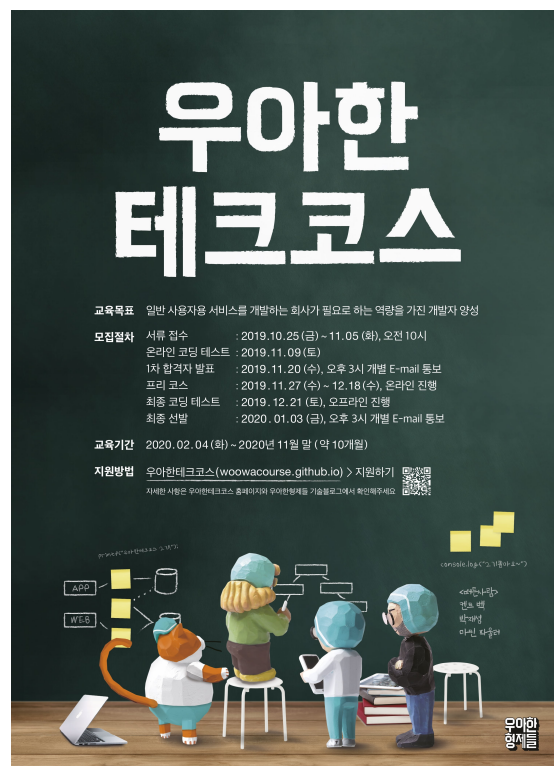
관련 PR(<https://bit.ly/3kEK5EA>)에서 구현 코드를 확인하실 수 있습니다.

우아한테크코스 - 백엔드 과정 (2020.02 ~ 2020.11)

일반 사용자용 서비스를 개발하는 회사가 필요로 하는 역량을 가진 개발자 양성

교육 방식 : 페어 프로그래밍, 미션 기반의 코드리뷰

(미션 구현 후, 경력 개발자에게 코드리뷰 형식의 피드백을 받음)



LEVEL1 : 프로그래밍 기본

자바 프로그래밍 언어에 대한 기본 문법 학습

읽기 좋은 코드와 테스트코드에 대한 중요성 파악

LEVEL2: 웹 프로그래밍

Spring 프레임워크 기반의 웹 애플리케이션 개발

TDD 사이클의 개발 및 리팩토링 경험

LEVEL3: 팀 프로젝트

하나의 서비스를 밑바닥부터 만드는 경험

LEVEL4: 웹 백엔드 심화, 프로젝트 유지보수

웹 서버를 직접 구현, HTTP에 대한 이해도 향상

성능을 고려한 시스템 설계 및 구축 경험

레거시 코드 리팩토링 경험

백엔드 역량과 소프트 스킬 역량 향상

현업에서의 개발 프로세스와 스킬을 간접적으로 느껴보고자 참여했습니다.

Java를 기반으로 한 Spring 웹 애플리케이션 개발 능력,

미션 기반의 페어프로그래밍과 팀 프로젝트를 통한 협업 역량,

TDD 사이클의 개발과 레거시 코드에 대한 리팩토링 경험,

현직자의 코드리뷰를 통한 클린 코드 역량 등을 얻을 수 있었습니다.