

안녕하세요! 백엔드 개발자 곽지혁입니다.

저는 코드의 확장성과 안정성을 깊이 고려하며, 모든 코드 작성 시 명확한 근거를 바탕으로 작업합니다. 구조적이고 근거 있는 코드를 작성하는 습관은 장기 프로젝트의 성공뿐만 아니라 개인의 성장에도 중요한 요소라고 믿습니다.

백엔드 개발자로서, 대규모 트래픽 처리와 자원 관리에 대한 깊은 이해를 가지고 있습니다. 복잡한 시스템에서 성능을 극대화하고 자원을 효율적으로 활용하며, 안정적이고 신뢰할 수 있는 서비스를 제공하기 위해 지속적으로 연구하고 개선해 나가고 있습니다. 이를 통해 고성능 애플리케이션의 구현과 안정적인 서비스 운영을 목표로 하고 있습니다.

- ✓ 팀 전체의 목표를 이해하고, 협업을 통해 그 목표를 달성하는 데 적극 기여합니다.
- ✓ 변화하는 요구사항에 유연하게 대응할 수 있도록, 확장성을 염두에 두고 코드를 설계합니다.
- ✓ 코드의 효과와 목적을 명확히 이해하며, 근거 있는 코드를 작성하려고 항상 노력합니다.
- ✓ 최적화와 자원 관리를 통해 사용자의 경험을 개선할 때 큰 보람을 느낍니다.

Profile

Contacts

- Email: kwlgur0416@gmail.com
- GitHub: [Hyukji](#)
- Blog: [늦었지만 빠른때](#)

Education

- Samsung Software Academy For Youth 11th (SSAFY)
 - 2024.01 ~
 - SW 필수 지식과 알고리즘 중심의 몰입형 코딩 교육
- 한국과학기술원 (KAIST) 몰입캠프
 - 2019년 6월 ~ 2019년 7월
 - 협업을 통해 집중개발을 경험하는 프로그래밍 캠프
 - 안드로이드 앱 및 Unity 3D 게임 개발
- 대구경북과학기술원 (DGIST)
 - 기초학부 공학 학사, 컴퓨터 트랙
 - 2018.02 - 2024.02
 - GPA: 3.65/4.3

Stacks

- BE : [SpringBoot](#), [JPA](#), [QueryDsl](#), [Mysql](#), [MongoDB](#), [node.js](#),
- Infra : [AWS\(EC2, S3\)](#), [Nginx](#), [Jenkins](#), [Docker-compose](#), [Docker](#)
- FE : [Vue.js](#)
- Data : [Python](#), [PyTorch](#), [NumPy](#), [pandas](#), [Matplotlib](#)

Language Skills

- OPIC IM1 (2024.03)

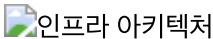


Projects

- ▶ 온라인 알고리즘 스터디를 위한 웹사이트 개발

2024.7 - 2023.8

First Project of Samsung Software Academy For Youth

프로젝트 개요

- 문제 제기
 - 스터디의 일정 및 문서 관리의 비효율성.
 - 코드 공동 작성의 부재로 인한 피드백 한계.
 - 구두 설명으로 인한 이해의 어려움.
- 역할
 - **Infra** 구축 및 DB 설계
 - **Nginx**를 이용한 ssl 인증, 프록시 설정
 - **Jenkins**, **WebHook**를 활용한 **ci/cd** 구축
 - **Docker**를 활용한 코드 실행을 위한 환경 구축
 - **redis**의 캐시 처리 최적화를 위한 환경 구축인프라 아키텍처
 - 폴더 구조 및 스터디 템플릿
 - **재귀 구조**를 이용한 폴더 기능 구현
 - 템플릿 구현을 통한 코드 자동 업로드 기능 구현
 - 폴더 구조 API 최적화
 - **N+1** 문제 해결
 - **queryDsl** 및 **fetchJoin**을 이용해 **N+1** 문제 해결
 - **Redis**를 이용한 캐시 활용.
 - 스터디의 전체 폴더 구조의 경우 추가, 수정, 삭제 보다 조회의 호출빈도가 더 높음.
 - **nGrinder**를 이용한 성능 테스트를 통해 **95.6%**의 성능 향상(23초 -> 1초)성능향상 전(23초) 성능향상 후(1초)
 - Java, Python 코드 실행
 - 실행 언어의 확장성을 고려해 **팩토리 패턴**을 활용.

CodeExecutorFactory

```
@Component
public class CodeExecutorFactory {
```

```

        private final Map<Language, CodeExecutor> executors;

        public CodeExecutorFactory(List<CodeExecutor>
executorList) {
            executors = executorList.stream()

.collect(Collectors.toMap(CodeExecutor::getLanguage,
Function.identity()));
        }

        public CodeExecutor getExecutor(Language language) {
            return executors.get(language);
        }
    }

```

CodeExecutor

```

public interface CodeExecutor {

    String executeCode(String code, String input);

    File createCodeFile(String code, String hostPath) throws
IOException;

    Language getLanguage();

    default String readProcessOutput(InputStream
inputStream) throws IOException { ... }
}

@Component
public class JavaCodeExecutor implements CodeExecutor { ...
}

@Component
public class PythonCodeExecutor implements CodeExecutor {
    ... }

```

- Container를 활용한 코드 실행 구현
 - 코드 실행 API 요청 시 컨테이너 내부에서 코드를 실행하고 결과를 반환한다.
 - 가상화 기술의 활용 이유 : 검증되지 않은 코드를 서버에서 직접 실행하는 것은 보안적인 취약점이 존재할 수 있다고 판단함.
 - 코드 실행 API의 요청이 많아진다면 자원 관리에 문제가 생길 수 있음
 - Container에 할당할 메모리, cpu를 설정
 - 무한 루프와 같은 경우를 대비해 컨테이너 시간이 10초를 넘어가지 않도록 설정
 - 세마포어를 이용한 컨테이너 개수 제한

기술 및 라이브러리

- BE & DB: [SpringBoot](#), [JPA](#), [QueryDsl](#), [Mysql](#), [Redis](#)
- Infra: [Nginx](#), [ec2](#), [Jenkins](#), [Docker](#), [Dokcer-compose](#)
- Etc : [WebRTC](#)

관련 자료

깃허브 주소

▶ RealWorld

2023.8 - 2023.10

SpringBoot Project

프로젝트 개요

- SpringSecurity를 이용한 인증/인가
 - [AuthenticationFilter](#), [AuthenticationManager](#), [AuthenticationProvider](#) 등 단순한 인증, 인가 구현보다는 [SpringSecurity](#)에서 지향하는 아키텍처들을 이해하고 구현하고자 함.
 - [Security Context](#)의 [TheadLocal](#)에 대해 공부하는 등 내부 구조를 이해하고자 노력함
- 구체적인 Test 작성
 - [단위테스트](#), [통합테스트](#), [인수테스트](#)를 꼼꼼하게 작성하면서 프로젝트를 진행함.
 - [Fixtures](#) 와 [Stub](#) 객체들을 활용해 테스트를 진행
- [CQRS](#) 패턴 적용
 - 게시글의 경우 조회 빈도가 더 높을 것으로 예상되어 읽기를 위한 간소화한 데이터 모델을 별도로 구성함

기술 및 라이브러리

- BE : [SpringBoot](#), [Mysql](#), [JPA](#), [QueryDsl](#)

관련 자료

[GitHub](#)

▶ 어절 단위 청취 기능을 탑재한 iOS 어플리케이션 개발


2023.1 - 2023.4

1인 개발 프로젝트

프로젝트 개요

- 문제 제기 및 솔루션
 - 어학 공부 중 부족한 부분을 이해하기 위해 여러 번 반복 청취가 필요함.

- 쉼도잉 학습 시 기존 플레이어의 초 단위 이동 및 AB 구간 반복 기능에 불편함 존재.
- **어절 단위로 오디오를 구분**해 보다 편리하고 정확한 위치 이동을 제공하는 어플리케이션 개발
- 결과
 - 어절 단위 청취 기능
 - 오디오 파일 생성 시, 오디오 파장 분석 및 어절 단위 구분
 - 어절 단위로 이동 및 반복 기능

 Onpeat_audio

- 오디오 파일 관리 기능
 - wifi & usb 를 이용한 오디오 파일 업로드 기능
 - 폴더 구조를 이용한 파일 관리

 upload

 upload

- 프로젝트 과정을 블로그에 기록하여 공유.

관련 자료

[GitHub](#), [Blog](#)

- ▶ 소스코드 유사도 분석 알고리즘 및 이를 이용한 교육 보조 프로그램

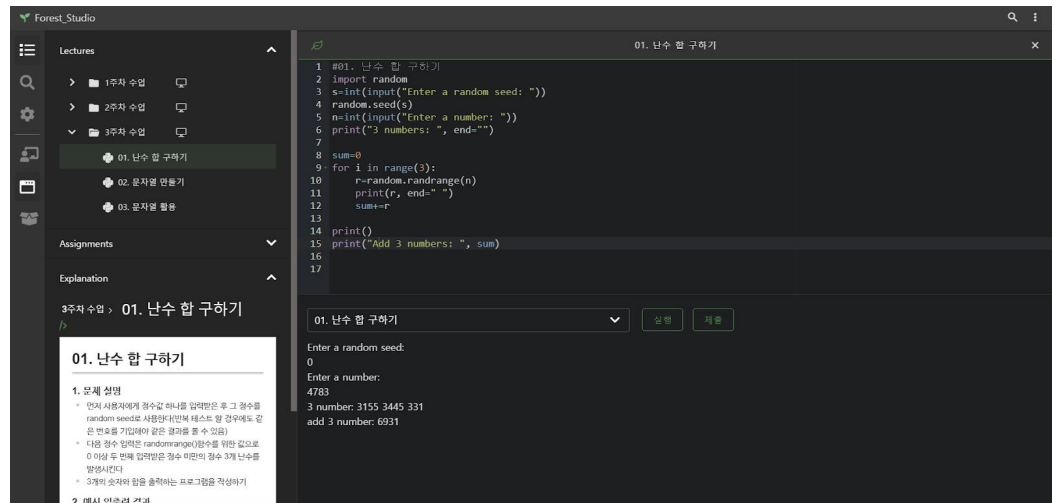
2020.03 - 2020.12

Undergraduate Group Research Program (UGRP) - DGIST

프로젝트 개요

- 문제 제기 및 솔루션
 - 코딩을 처음 배우는 학생들의 코드는 길이가 짧아 구조적 유사성은 적합한 표절 기준이 아님
 - 코딩 스타일을 기준으로 표절도를 검사하되 학습 과정에 따른 코딩 스타일의 가변성을 고려함
 - 이 알고리즘을 적용한 코딩 교육 사이트를 만들고자 함
- 결과
 - 코딩 스타일 기반의 소스코드 유사도 계산 알고리즘 연구 (Python, tokenizer)
 - 변수명, 클래스명, 연산자 주위 공백 등을 기준으로 코딩 스타일 vector를 생성
 - 지금까지 작성해온 코딩 스타일과 현재의 코딩 스타일을 비교하여 표절도를 계산함
 - 코딩 스타일에 급격한 변화가 있을 경우 높은 표절도를 가짐
 - 코드 표절 검사 기능을 활용한 코딩 교육 사이트 (MEVN stack)
 - 로그인 및 회원가입
 - 강좌 및 과제 등록
 - 개인 학습현황
 - 실습 과제를 위한 실시간 Python interpreter
 - 샌드박스 및 에디터 화면 분할 기능

■ 화면



역할

- 팀 구성: 4인
- 알고리즘 연구
 - 연산자 및 괄호 주위 공백, 평균 함수 길이 등의 코딩 스타일 vector 추출
 - 표절도 계산식 정립
 - K-means Clustering을 이용한 군집 내 편차 계산
- 사이트 개발
 - 유저 및 강의 관리
 - 강의 캘린더 및 일정 조율 기능
 - 에디터 부가 기능(분할, 확장, 다중)
 - AWS EC2 서버 및 MongoDB Atlas 관리

기술 및 라이브러리

- Data Analysis: **Python**, **tokenizer**, **pandas**, **NumPy**
- Web Development: **Vue.js**, **Node.js**, **AWS EC2(Ubuntu)**, **MongoDB**, **GitHub**, **Vuetify**
- Extra: **Docker**, **Socket.io**

관련 자료

[GitHub](#)

Work Experiences

▶ 올인원주식회사 인턴십

2021.01 - 2021.03

사용자의 영어 수준을 평가하는 알고리즘 개선

역할

- 사용자의 영어 수준을 평가하는 알고리즘 개선

- 기존 알고리즘

- 사용자의 임시 수준을 설정하고, 그에 맞는 문제를 제공.
- 제공된 문제의 정답 여부에 맞춰 임시 수준이 변동.

- 문제 제기 및 솔루션

- 문제 난이도의 부적확성
 - 사용자의 수준에 맞춘 정답률 계산 시, 같은 난이도의 문제라도 정답률 분포에 큰 차이가 발생.
 - 사용자의 수준별 정답률을 바탕으로 문제 난이도를 재배치하여 알고리즘의 정확도를 향상시킴.
- 후반부 문제에 대한 과도한 의존성
 - 기존 알고리즘이 후반부 문제의 정답 여부에 지나치게 의존함.
 - 임시 수준 변동에 가중치를 부여하여, 후반부 문제로 갈수록 가중치를 줄임으로써 문제를 해결함.

성과

- 특허 출원
 - 빅데이터 기반 학습자 맞춤형 외국어 레벨 측정 시스템 (출원번호: 1020210035794)
 - 맞춤형 외국어 문제선정 시스템 (출원 번호: 1020210035829)



Awards

- 삼성 청년 SW아카데미 프로젝트 우수상 - 삼성전자 주식회사 (2024)
- 삼성 청년 SW아카데미 성적우수상 - 삼성전자 주식회사 (2024)
- 2023 Dean's List - DGIST (2023)



Certificates

- SW Certificate Professional - 삼성전자 주식회사 (2024)