Porting Manual

기술 명세

프로젝트 관리

항목	사용 도구
이슈 관리	JIRA
형상 관리	Gitlab
커뮤니케이션	Confluence
	Mattermost
디자인	Figma
CI/CD	Jenkins
UCC 제작	Premiere pro

백엔드 서버

Skill	Version
AWS EC2 Ubuntu	20.04 LTS
AWS S3	
Docker	24.0.6
NGINX	1.25.2
django	4.2.5
MariaDB	11.1.2
certbot	2.7.0
uwsgi	2.0.22
Jenkins	2.141.1
PyCharm	2023.2.1

앱

Skill	Version
Node.js	`18.17.0`
React-Native	0.76.7
Typescript	21~

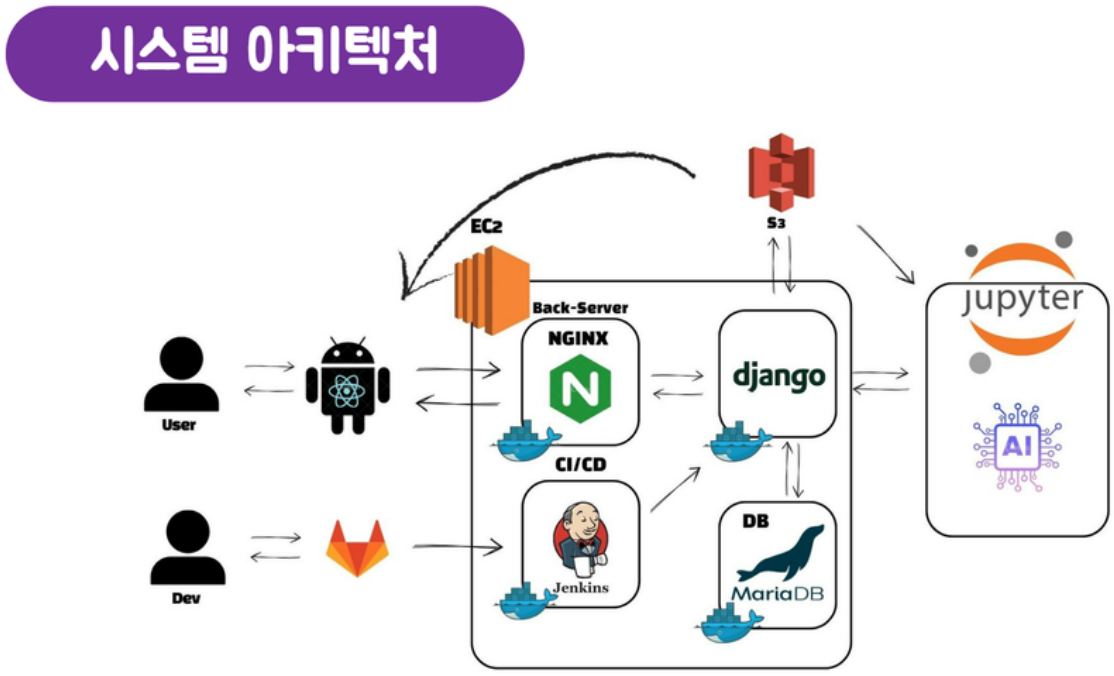
Android SDK	21~
Java	11.0
kotlin	1.9.0
Recoil	0.7.7
react-native-audio-recorder-player	3.5.4
react-native-firebase	18.4.0

AI 음성 학습 서버

Skill	Version
Ubuntu	20.04.4 LTS
Python	3.9.7
CUDA	12.1
pytorch	2.0.0+cu118
pytorch-lightning	2.0.9
Visual Studio Code	1.82.2
vim	8.1.3741

상세 라이브러리 정보: requirements.txt 참고

시스템 아키텍처



배포 가이드

서버

1. git clone

```
git clone https://lab.ssafy.com/s09-ai-speech-sub2/S09P22A404.git /home/ubuntu/moonheee/
```

2. Docker 설치

```
sudo apt-get update

sudo apt-get install -y \
    ca-certificates \
    curl \
    gnupg \
    lsb-release

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archi

echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://downl
  $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

apt-get update
apt-get install -y docker-ce docker-ce-cli containerd.io
```

3. docker compose 실행

```
cd /home/ubuntu/moonheee/S09P22A404

docker compose up
```

<details>
<summary>3-1. Jenkins 연동</summary>

docker compose를 직접 실행하지 않고 CI/CD를 구축하는 경우

jenkins pipeline script

```
pipeline {
    agent any
    stages {
        stage('Pull') {
            steps {
                echo "Branch : back"
                echo "Pull back branch"
                git branch: "back", url: "https://lab.ssafy.com/s09-ai-speech-sub2/S09P22A404", credentialsI

            }
        }
        stage('Compose Task') {
            steps {
                sh 'cd /var/jenkins_home/workspace/django/back/ && docker-compose down'
                sh 'cd /var/jenkins_home/workspace/django/back/nginx/ && docker build -t moonheee/nginx .'
                sh 'cd /var/jenkins_home/workspace/django/back/singchro/ && docker build -t moonheee/django
                sh 'cd /var/jenkins_home/workspace/django/back/ && docker-compose up -d'
            }
```
</details>

```
        }
    }
}
```

jenkins docker 실행

```
docker run -d \
  --name jenkins \
  -p 8093:8080 \
  --privileged \
  -v /usr/bin/docker:/usr/bin/docker \
  -v /var/run/docker.sock:/var/run/docker.sock \
  -v /var/jenkins_home:/var/jenkins_home \
  --user root \
  --restart unless-stopped \
  jenkins/jenkins:lts
```

4. django에서 DB migrate

```
docker exec -it django /bin/bash

python manage.py makemigrations
python manage.py migrate
exit
```

migrate는 최초 1회만 실행해주면 된다.

앱

1. bundle 파일 생성

```
npx react-native bundle --platform android --dev false --entry-file index.js --bundle-output android/app/src/mai
```

2. 프로젝트 run-android 실행

```
npx react-native run-android
```

3. 추출된 .apk파일 이용하기

- app\front\SingChro\android\app\build\outputs\apk\debug\app-debug.apk
- 휴대폰에 넣고 apk 파일을 통해 어플리케이션 설치

AI 학습 서버

1. git clone

```
git clone https://lab.ssafy.com/s09-ai-speech-sub2/S09P22A404.git
```

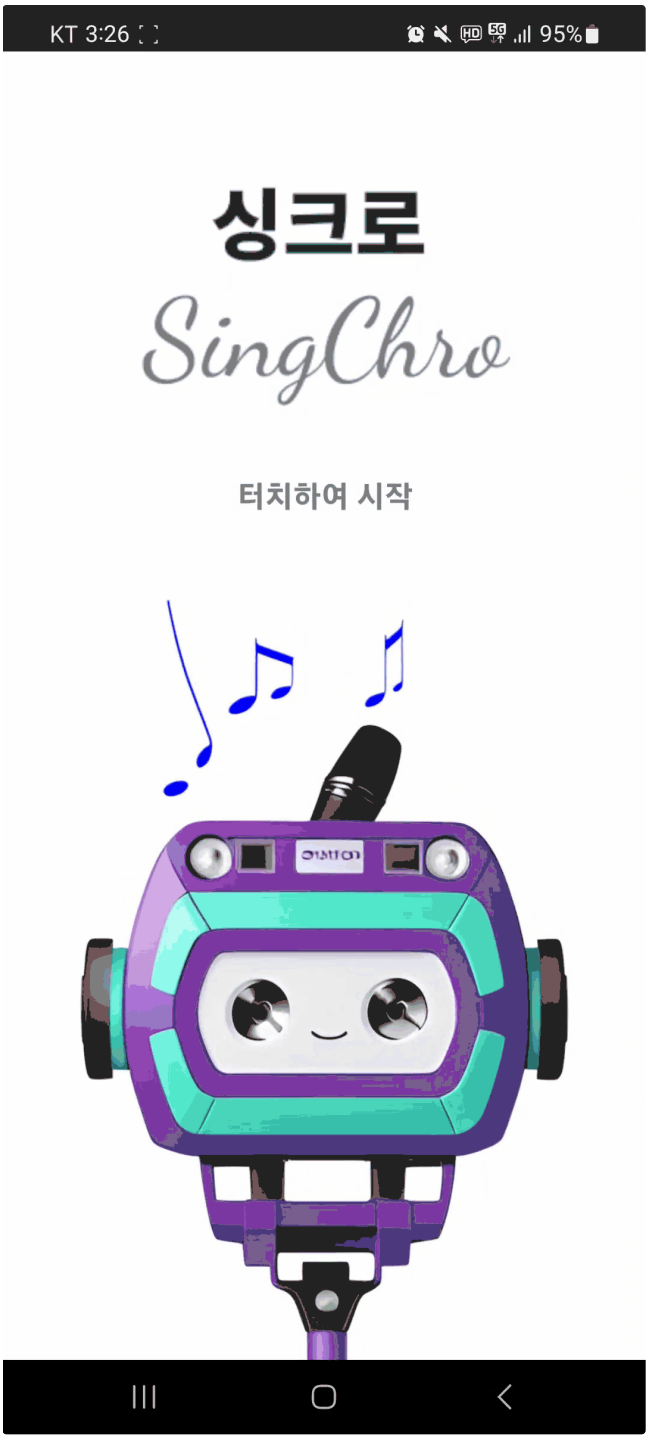
2. Pretrained model 다운로드

GitHub - PlayVoice/whisper-vits-svc: Core Engine of Singing Voice Conversion & Singing Voice Clone

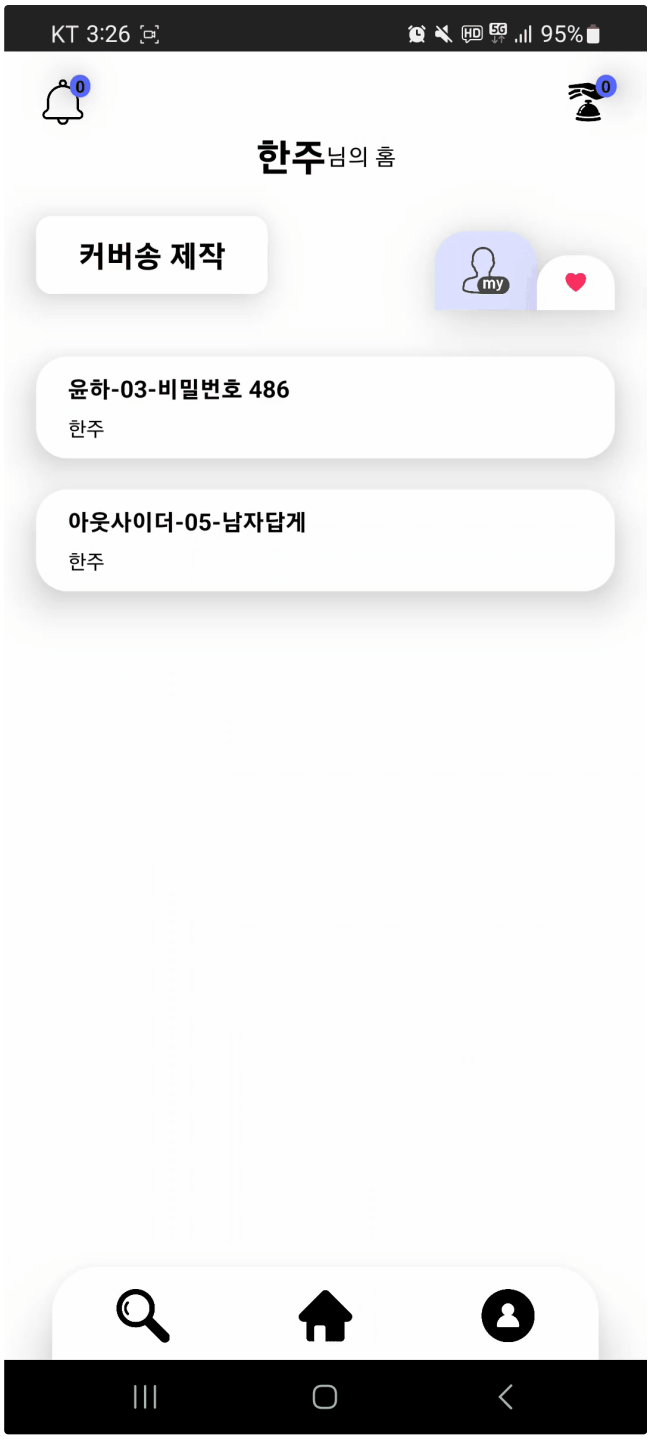
3. 제작 요청 탐색

```
python multiprocessing.py
```

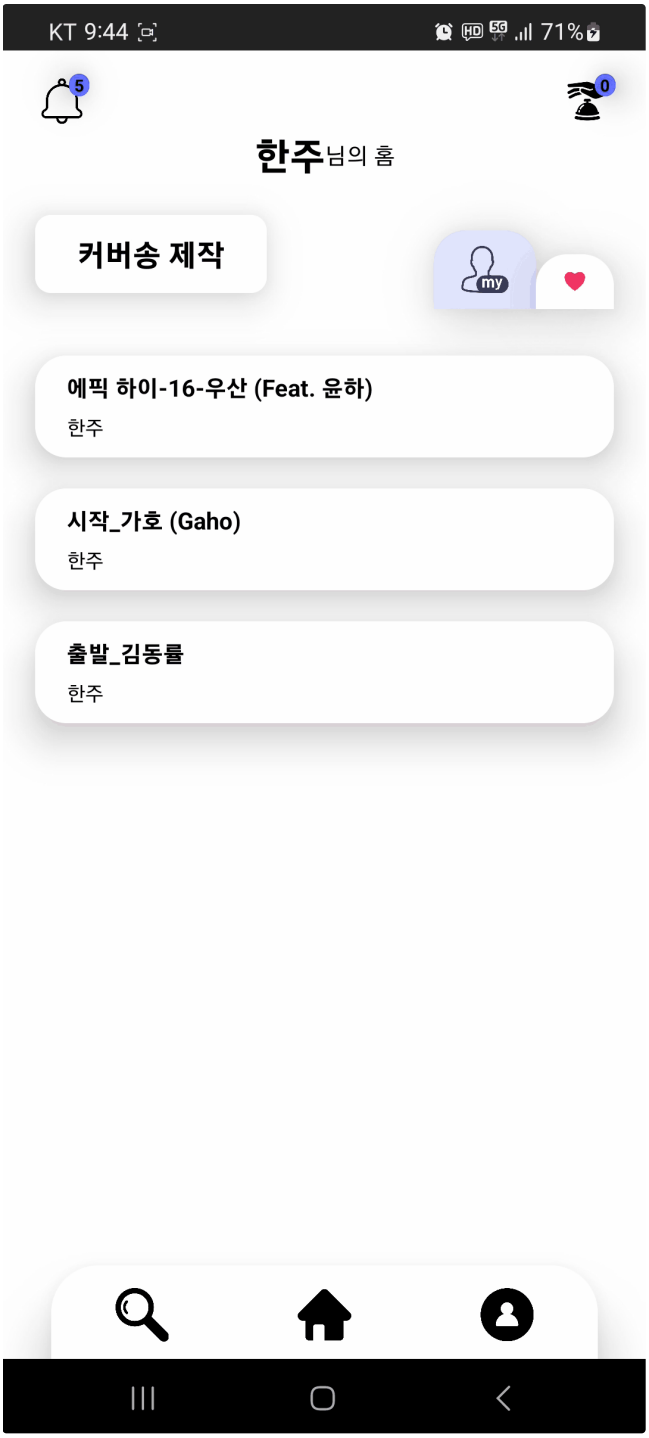
시연 시나리오



모델 생성



커버송 제작



커버송 제작 완료

노래 재생 및 좋아요