

SSAFY 7기 특화 프로젝트 최종 발표

청각 장애인을 위한 일상 소통 보조 어플

Team : D202

Contents

1. UCC
2. 기획 배경
3. 프로젝트 개요
4. 시연
5. 기술 소개
6. Q & A

01

UCC

02

기획 배경

지금부터, 상황을 하나 드리겠습니다.

청각장애인의 입장에서 생각해주세요.

청각 장애인의 소통 방법

수어, 필담, 구화



카페에 도착한 당신, 아메리카노에 얼음을 적게 넣고 싶습니다.

청각 장애인의 소통 방법

수어, 필담, 구화



“아메리카노에 얼음 적게” 어떻게 주문하실건가요?

청각 장애인이 소통하는 방법

수어, 필담, 구화

청각 장애인의 비장애인과의 소통 방법

수어, 필담, 구화

수어 : 몸짓이나 손짓으로 뜻을 표현

필담 : 글을 써서 서로 묻고 답함

구화 : 입술 모양 으로 뜻을 알아들음



손말 어플의 필요성

청각 장애인에 비해 부족한 통역사의 수



농인과 통역사의 수는 최대 200명당 1명 꼴

손말 어플의 필요성

청각장애인을 위한 서비스의 부실



이용량은 증가하고 있지만, 곧 서비스가 종료될 예정

손말 어플의 필요성

청각장애인을 위한 서비스의 부실

[뒤로](#)

 손말이음센터

 공지사항

공지 손말이음센터 앱 종료 안내

[목록](#)

2022-07-19

손말이음센터 앱서비스 시스템 노후화 및 일부 기능 인터넷익스플로어 (IE) 미지원 등에 따라 올해 하반기에 앱서비스가 종료 될 예정입니다.
기존 이용자분들은 PC, 스마트폰(크롬, 사파리)에서 신규 홈페이지 (107.kr)로 접속하여 이용 부탁드립니다.

이용량은 **증가**하고 있지만, 곧 **서비스**가 **종료**될 예정

청각 장애인의 비장애인과의 소통 방법

필담과 구화



“ 필담, 구화로 충분할까? ”



"소통불통" 이비인후과에... 청각장애인들 “도저히 못 믿겠다!”



“의사가 바쁘다는 표정을 지으면서
입으로 대충 말을 하는데,
무슨 말인지 도저히 알 수가 없었어요.
그래서 필담을 해달라고 부탁했는데
바쁘다고, 쓸 시간이 없다고,
보호자를 데리고 오라는 말만 했습니다..”

필담, 구화의 문제 사례

병원 사례 분석



구화 : 답변할 때 입 모양을 신경 써야하며, 반복해야한다.

필담, 구화의 문제 사례

병원 사례 분석

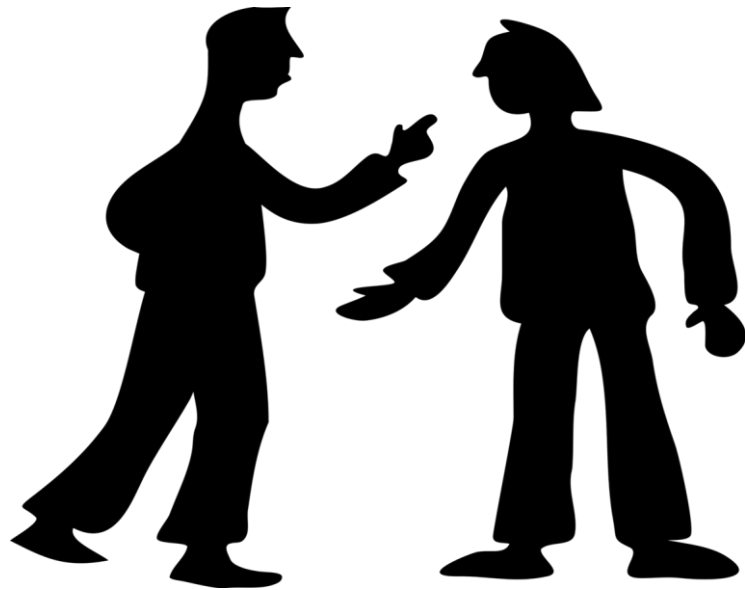


필담 : 작성 시간이 오래 걸리고, 먼 거리에서는 보기 어려움

상대가 주의를 기울여야 하는

소통 방식으로 인해

의사소통 갈등이 발생



청각 장애인은 소통의 제약을 받고 있다.

1. 필담의 소요 시간을 줄여 소통의 불편함 개선
2. 음성으로 제공되는 서비스의 이용 가능성 확대



03

프로젝트 개요

주요기능 소개

손말어플의 4가지 주요 기능

매크로



수어통역



음성 자막 변환



통화

매크로 기능

사전 등록 내용의 음성 출력



- 카테고리 별 매크로 조회
- 자주 쓰는 내용 매크로 등록
- 등록된 내용을 음성으로 출력

필담의 소요 시간과, 답하는 이의 불편함을 줄이기 위한 기능

텍스트 음성 번역 기능

실시간 음성 <-> 텍스트 번역

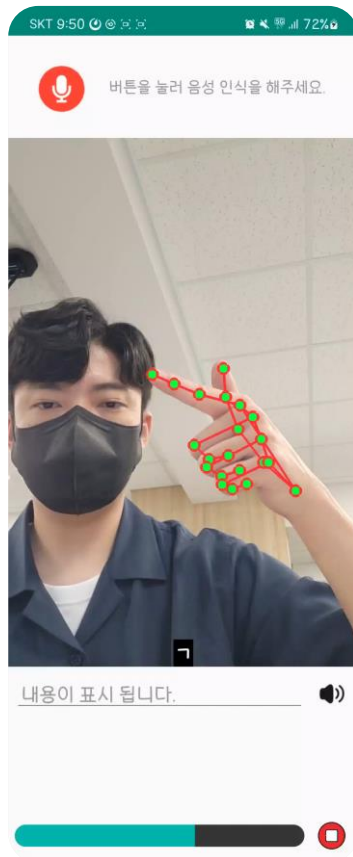


- STT를 이용한 실시간 음성 인식
- TTS를 통한 음성 출력
- 매크로를 이용한 빠른 음성 출력

필담의 불편함을 줄여 소통의 벽 완화하기 위한 기능

수어 음성 번역 기능

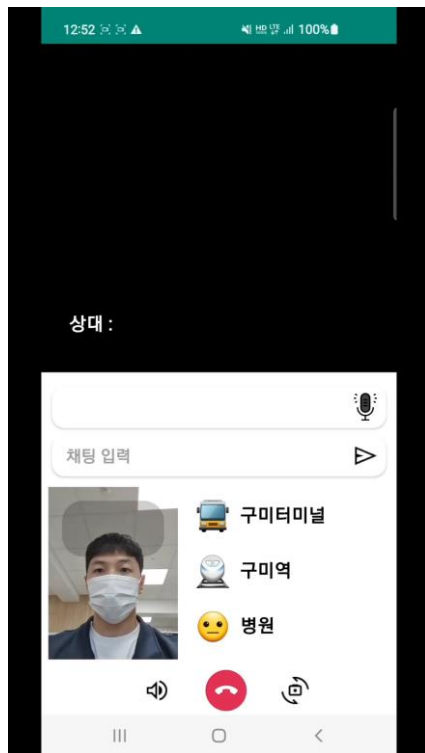
수어 사용자를 위한 수어 <-> 텍스트 번역



- 입력된 글로 등록된 매크로를 검색해 제공
- Tensorflow Lite로 카메라 수어 인식
- 인식된 글자를 한글 문법에 맞게 합쳐 입력

통화 번역 기능

통화 음성 <-> 텍스트 번역



- 텍스트로 입력 후 TTS로 전달
- 수어 인식 후 텍스트 표기
- 매크로 사용

음성으로 이루어진 서비스들을 이용할 수 있도록 보조하는 기능

04

시연

05

기술 소개

기술 설명

1: AI 모델

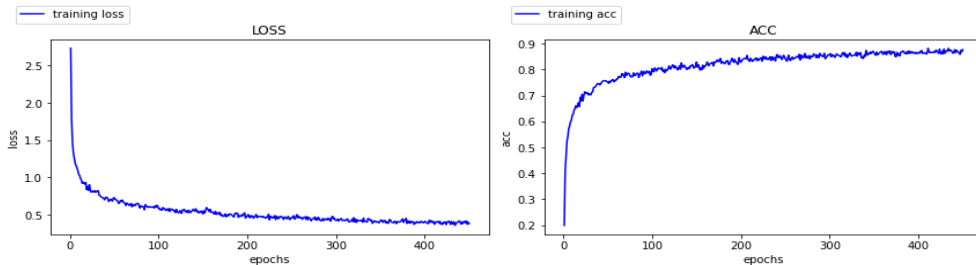
MediaPipe



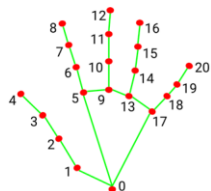
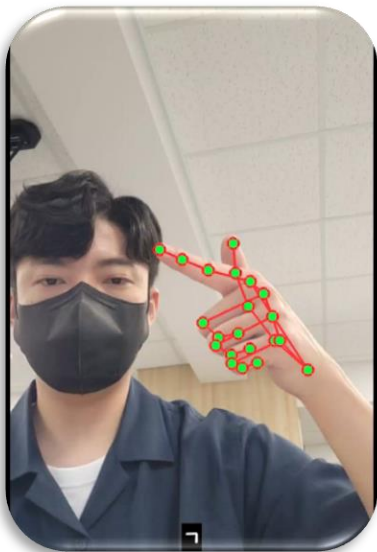
```
[ [ 1.52448729e-01 4.34540510e-01 5.47049979e-07 ]  
[ 1.82152718e-01 5.29042602e-01 -3.25236320e-02 ]  
[ 2.52020836e-01 5.83118498e-01 -3.80426273e-02 ]  
[ 3.12675834e-01 6.20321512e-01 -4.03913520e-02 ]  
[ 3.43068689e-01 6.64934456e-01 -4.20230851e-02 ]  
[ 3.42863023e-01 5.11993766e-01 -1.17076123e-02 ]  
[ 4.13777083e-01 5.34504712e-01 -2.26635486e-02 ]  
[ 4.58407104e-01 5.46908617e-01 -3.58532965e-02 ]  
[ 4.98599231e-01 5.54765880e-01 -4.73409407e-02 ]  
[ 3.51476908e-01 4.50715214e-01 -7.73185538e-03 ]  
[ 4.32063878e-01 4.54052866e-01 -1.70411747e-02 ]  
[ 4.84562993e-01 4.56474364e-01 -3.08668464e-02 ]  
[ 5.29164553e-01 4.55812305e-01 -4.15452123e-02 ]  
[ 3.42159778e-01 3.94686997e-01 -8.45576078e-03 ]  
[ 4.18537021e-01 3.84513080e-01 -2.20088419e-02 ]  
[ 4.67124701e-01 3.81350517e-01 -3.75027321e-02 ]  
[ 5.09073853e-01 3.77906173e-01 -4.81789634e-02 ]  
[ 3.17080408e-01 3.41267318e-01 -1.25723453e-02 ]  
[ 3.72829974e-01 3.19293797e-01 -2.90320795e-02 ]  
[ 4.11474228e-01 3.07055265e-01 -3.99099849e-02 ]  
[ 4.48853254e-01 2.99883723e-01 -4.66583222e-02 ] ]
```

```
#####data = np.array([angle], dtype=np.float32)#####  
[[13.078639 31.261017 19.123344 22.001425 1.9695394 5.226923  
18.842508 3.2440104 3.2073975 20.384327 5.086204 2.2255788  
12.87885 8.369481 5.2964168]]
```

TensorFlow



- MediaPipe를 적용하여 손의 **keypoint 값 추출**
- 각 keypoint들의 **각도를 계산**하여 모델 input으로 사용

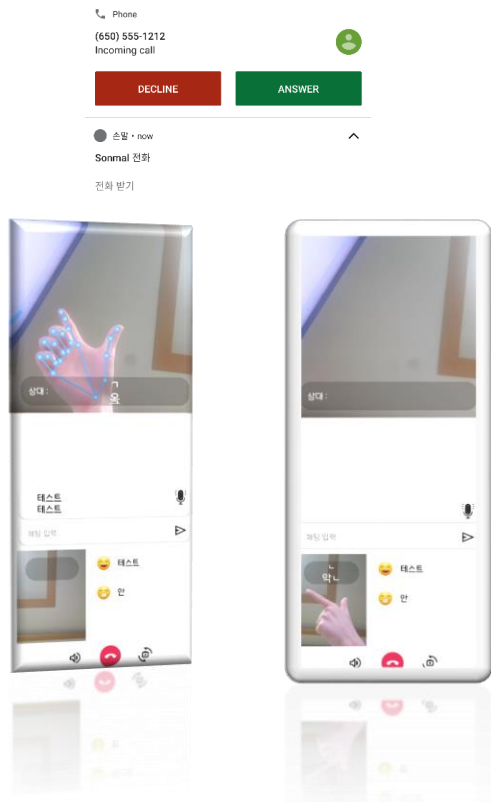


- | | |
|-----------------------|-----------------------|
| 0. WRIST | 11. MIDDLE_FINGER_DIP |
| 1. THUMB_CMC | 12. MIDDLE_FINGER_TIP |
| 2. THUMB_MCP | 13. RING_FINGER_MCP |
| 3. THUMB_IP | 14. RING_FINGER_PIP |
| 4. THUMB_TIP | 15. RING_FINGER_DIP |
| 5. INDEX_FINGER_MCP | 16. RING_FINGER_TIP |
| 6. INDEX_FINGER_PIP | 17. PINKY_MCP |
| 7. INDEX_FINGER_DIP | 18. PINKY_PIP |
| 8. INDEX_FINGER_TIP | 19. PINKY_DIP |
| 9. MIDDLE_FINGER_MCP | 20. PINKY_TIP |
| 10. MIDDLE_FINGER_PIP | |

- MediaPipe를 적용하여 손의 **keypoint 값 추출**
- 각 keypoint들의 **각도를 계산**하여 input에 삽입
- output으로 각 자음, 모음의 **일치율** 출력

기술 설명

3 : WebRTC를 이용한 통화 번역



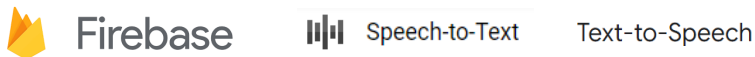
- 전화를 이용한 **WebRTC 연결** 및 화상통화



- **수어 번역** 및 출력

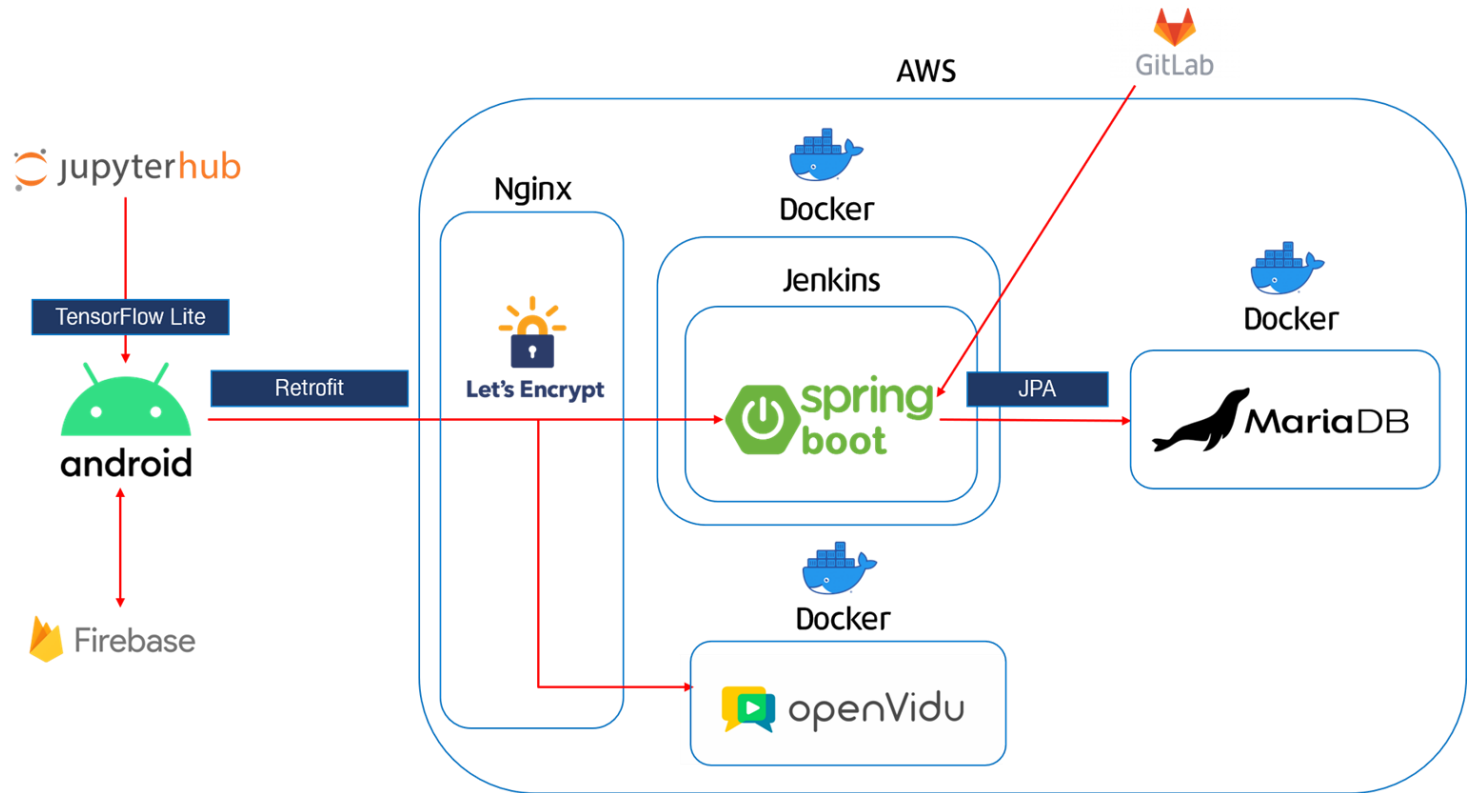


- 채팅, 매크로, STT의 텍스트 **데이터 전송**



프로젝트 구성도

손말 어플의 구성



프로젝트 관리 및 설계

- 이슈관리 : Jira
- 형상관리 : Gitlab
- ERD : ERD Cloud
- UI/UX : Figma

Server

- AWS EC2
- MariaDB 3.0.7
- HeidiSQL 12.1.0
- Nginx
- Docker
- Jenkins
- Certbot
- OpenVidu

Backend

- JDK 11
- SpringBoot 2.7.3
- Gradle 7.5
- Springfox
- Swagger UI 3.0.0
- Json Web Token 0.11.5
- Spring Security 5.7.3
- Jasypt 3.0.4
- Firebase 8.1.0
- JPA Hibernate
- Lombok
- Log4j

Frontend

- Android dolphin 2021.3.1
- Kotlin 1.7.20
- Gradle 7.5
- SDK Target&Compile SDK 33 / Min SDK 21
- WebRTC
- MediaPipe
- Firebase 20.0.6
- Paging 3.1.1
- Retrofit 2.9.0
- CameraX : 1.2.0-beta02

AI

- PyTorch 1.12
- CudaToolkit 11.6
- Tensorflow Lite 2.10.0
- Jupyter
- Anaconda3

배포 진행 과정

마켓 배포에 필요한 것 체크

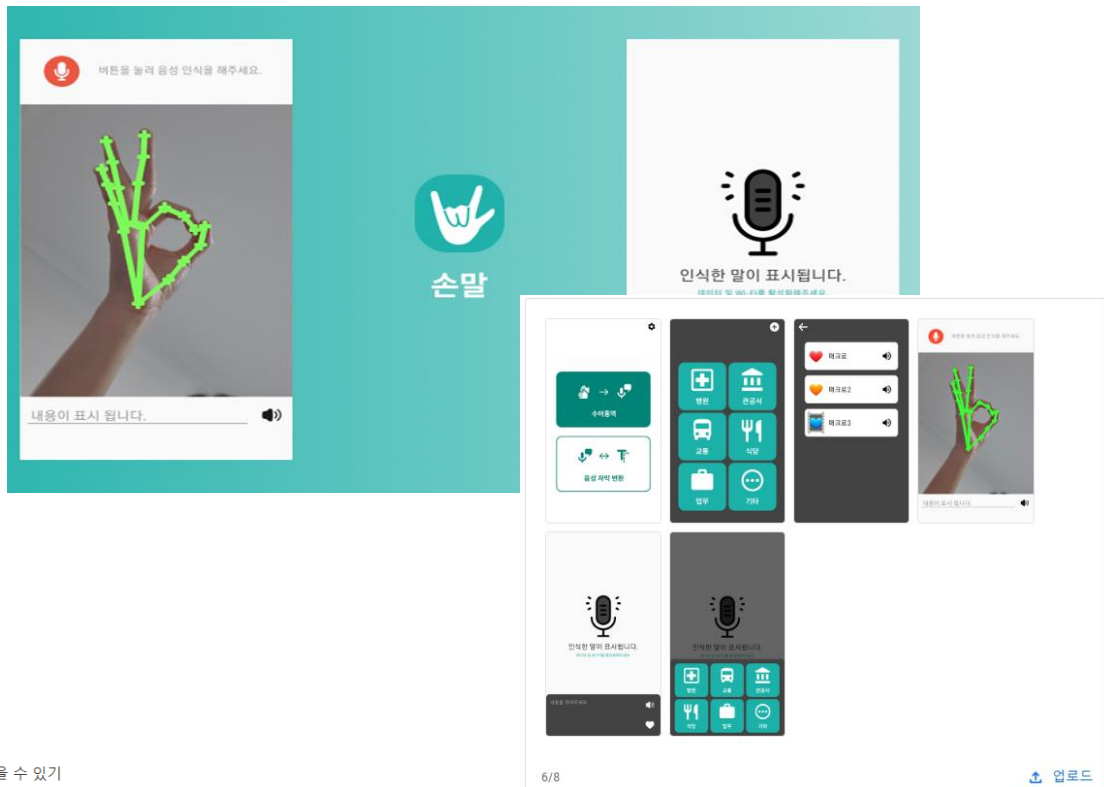


App 배포 check list

- ☐ 앱 권한 어디에 사용하는지 첫 화면에 명시할 것
 - 다 표기했는지 확인하자
- ☐ 개인정보처리방침 & 서비스 이용 동의 URL 필수 (= 약관 필수)
- ☐ 서버 SSL 미적용
 - 모든 http 허용 시, 리젝 사유
- ☐ 클릭 시 눌러지지 않는 버튼이 존재하면 안 된다.
 - ☐ 개발 중이라면, 반드시 개발 중이라고 명시할 것.
- ☐ 소셜 로그인 진행 시, 개발자 페이지에 aab 파일에 대한 key값 추가해두기
- ☐ 소셜 로그인으로 하는 경우, 출시할 때 테스트 계정 첨부
- ☐ 전-세계 출시일 경우, 소셜 로그인 해와 접속 풀어두기
- ☐ 다른 사람의 정보가 노출되는 순간 신고와 차단 가능 필수, 신고와 차단은 분리
 - SNS 기능 등이 있다면 필수라는 애가
 - 개발자 고려 사항 : 신고 및 차단 시, 상대방에게 그 정보가 보이지 않아야 함
 - (+ 차단 해제같은 기능도 있어야 할 듯 싶다고)

권장 사항

- ☐ 버튼 크기를 16, 18, 12dp 등 그대로 구현하면, 실제로 사용할 때 버튼이 잘 눌러지지 않을 수 있기 때문에 padding을 적용해 크기를 더 크게 잡아주는 게 좋음
 - 구글의 권장 사항은 48dp



후대전화 스크린샷을 2~8장 업로드하세요. 스크린샷은 PNG 또는 JPEG, 장당 최대 8MB, 가로세로 비율 16:9 또는 9:16, 가로세로 길이는 320px X 3840px 사이여야 합니다.

배포 현황

앱 스토어 배포 완료





김남희(FE)



서재형(FE)



배한용(FE)



정봉진(FE)



배시현(BE)



편예린(BE)

생활 통역권 확대 서비스 손말



Q & A