



우리도 차겠지

오윤재 박현서 송지영 이수정 정지현 사드

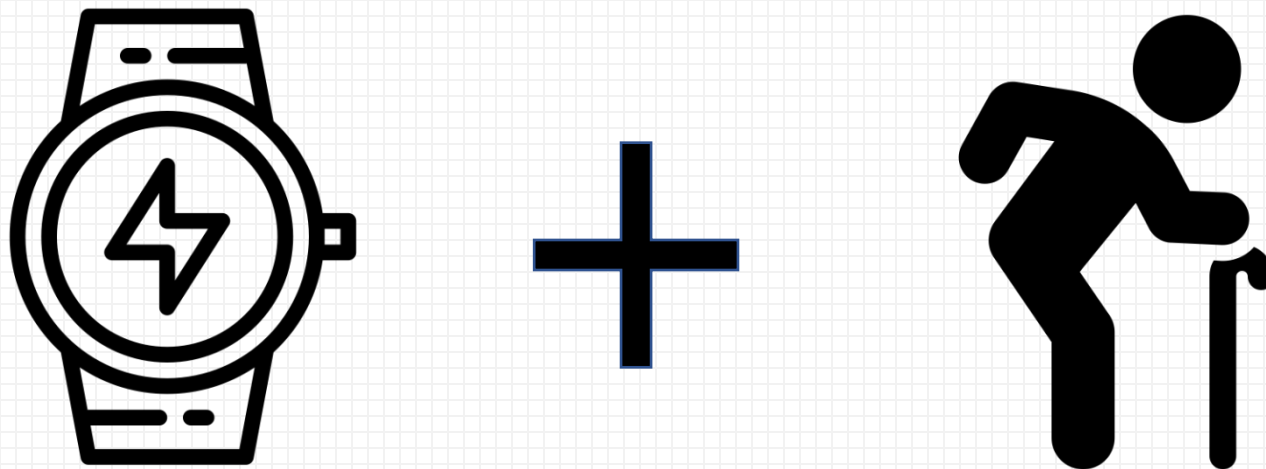
목차



- — 1. 프로젝트 목표
- — 2. 수행 내용 및 중간 결과
- — 3. 향후 추진 계획

프로젝트 목표

주제 선정 배경



스마트워치를 이용한 종합 노인 케어 시스템

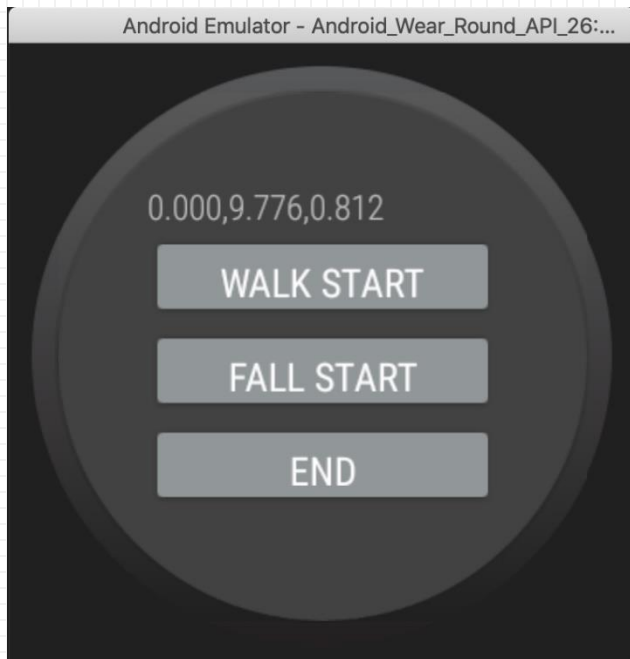
SILVER WATCH 기능



수행 내용 및 중간 결과

Wear 앱

1 데이터 수집을 위한 앱을 제작하여 팀원의 위치에 모두 설치 후 실제 수집 진행.



데이터 수집 앱

	x	y	z
0	-0.347922	1.412436	9.371981
1	0.330484	1.817755	9.883181
2	0.121596	2.068804	10.851442
3	-0.942006	0.498314	10.119856
4	-1.779952	0.655459	9.350901
5	0.318985	2.505264	6.790112
6	-3.047650	1.026282	7.275439
7	-2.260489	1.442141	8.356768
8	-3.355712	1.254334	6.930488
9	-7.808467	-3.641589	3.516421
10	-12.247329	-2.758608	4.710338

수집된 csv파일

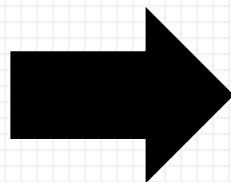
Wear 앱

- 2 Wear 앱에서 사용자를 구별하기 위한 방법으로 UUID생성.
- 3 사용자 등록 QR코드 생성.
- 4 필요한 기능을 테스트 할 수 있는 데모 앱 개발 완료.
- 5 센서 데이터 및 배터리 잔량을 전송하는 백그라운드 서비스 개발 중.

머신러닝

1 시계열 순서로 들어오는 데이터들을 일정 크기(series)로 잘라 데이터셋을 생성.

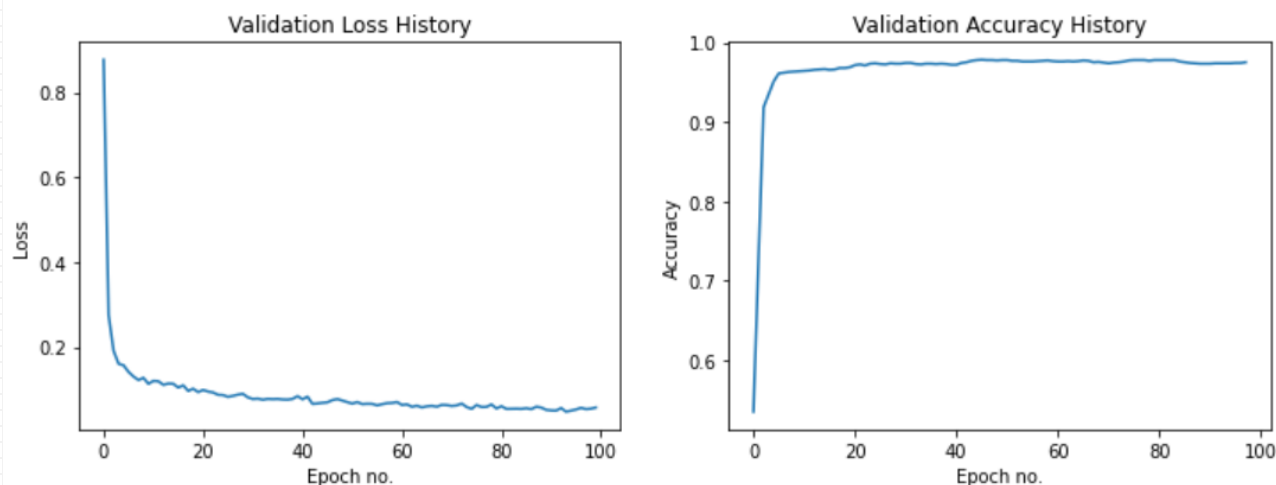
분류	데이터 개수
 낙상	543개
 걷기	713개



분류	데이터 개수
 낙상	613개
 걷기	5756개

머신러닝

2 Separable Convolution을 사용하여 분류 모델 구현



학습 그래프

```
answer = 0
for i in range(len(submit['activity'])):
    if submit['activity'][i] == y_tst['activity'][i]:
        answer = answer + 1

print("Test Accuracy : ", answer / len(submit['activity']) * 100, "%")
```

Test Accuracy : 97.61904761904762 %

테스트 데이터셋에 대한 정확도

3 학습된 Classifier를 Python Flask를 사용하여 API로 제공.

```
* Serving Flask app "__main__" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production d
ployment.
  Use a production WSGI server instead.
* Debug mode: off
```

```
Predicting on test dataset
[0]
```

```
Predicting on test dataset
[1]
```

0 : 낙상 1 : 걷기

안드로이드 앱

1 레이아웃 디자인 수정.

- 메인 메뉴 화면을 linear layout으로 변경.
- 메뉴 버튼을 설명하는 text view를 더욱 크게 설정.



안드로이드 앱

2 QR코드 스캔 기능 추가.

- QR코드 스캔 버튼 및 사용자 정보 입력 버튼 추가.
- 워치 등록 메뉴에서 사용자 정보를 전달 받는다.
- 사용자 정보를 aws 서버에 전달.
- 서버로 제대로 전달하지 못하는 경우를 대비하여 Exception Handling 구현.

워치 등록 메뉴

이름

전화번호(000-0000-0000)

워치 아이디

QR 코드 스캔 (워치 아이디 가져오기)

새로운 사용자 등록하기

메인 메뉴로 돌아가기

안드로이드 앱

3 접근 권한 부여 알림 설정

앱을 작동시키기 위한 권한들을 사용자로부터 받기 위하여 권한 부여 신청 알림을 만들어 놓았다.

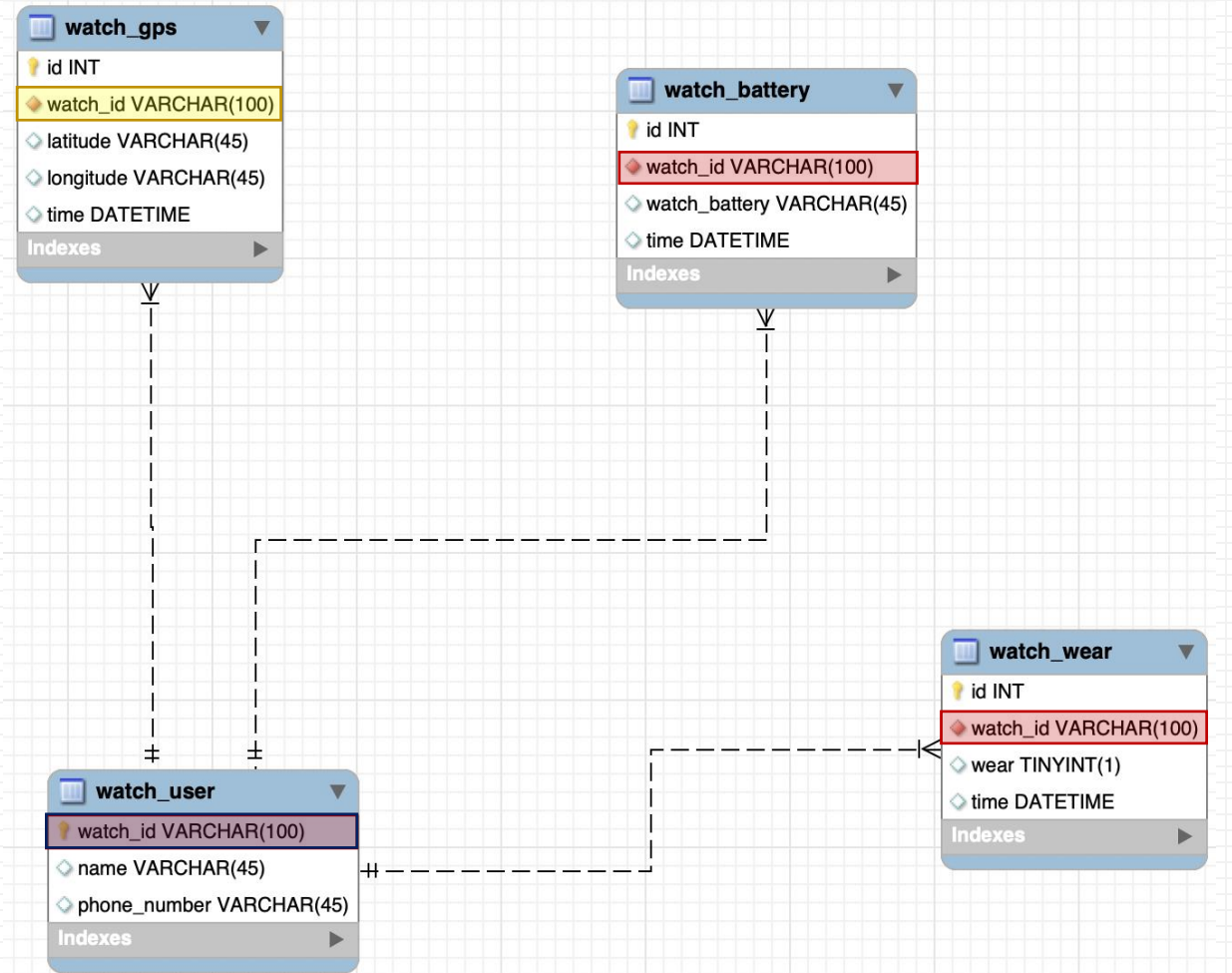
```
<uses-permission android:name="android.permission.RECEIVE_SMS" />  
<uses-permission android:name="android.permission.SEND_SMS" />  
<uses-permission android:name="android.permission.INTERNET"/>  
<uses-permission android:name="android.permission.READ_CONTACTS" />
```

접근 권한 : 문자 보내기, 인터넷 사용, 카메라 사용, 연락처 접근

데이터베이스

MySQL 설계

- Flask에 MySQL을 연동하고 restful api를 사용하여 정보를 받아온다.
- 기능별로 table을 제작.



서버

1 서버는 위치 사용자의 개인정보, 위치정보, 착탈여부 등을 timestamp와 함께 데이터베이스에 저장하거나 불러옴.

2 Firebase를 사용해 안드로이드 앱 푸시알림 제작.



향후 추진 계획

Wear 앱 & 머신러닝

Wear 앱

백그라운드 서비스 개발을 완료한다. 제작한 데모 앱을 바탕으로 서비스를 백그라운드에서 작동할 수 있게 변경하고, 모든 기능을 통합한다.

머신러닝

API 제공을 발전시킨다.

- 서버 API에 연동해 낙상여부에 따라 처리가 가능하도록 만들 예정이다.
- 모델의 추가 학습을 위해 다른 저장소에 업로드 하는 기능 구현.

안드로이드

- 1 사용자 등록과 동시에 작성 정보를 서버에 보내고 db 에 저장하는 기능 구현.
- 2 사용자 관리에서 등록된 기기의 사용자 인적사항을 list 형식으로 확인하는 기능 구현.
- 3 google map api를 통해 현재 위치 사용자의 위치를 관리자 앱에서 확인하는 기능 구현.
- 4 기기의 배터리가 충전이 필요하면 관리자 어플에 푸시 알림이 오는 기능 구현.
- 5 recyclable view를 활용하여 어떤 단말기에서도 레이아웃들이 동일하게 보이는 기능 구현.

서버 & 데이터베이스

- 1 스마트 워치의 사용자의 행동 반경을 MySQL에 저장할 수 있도록 구현.
- 2 현재 개발 중인 안드로이드 개발 앱에 푸시알림 기능을 추가하여 완성할 예정.



Q&A