



국민대학교  
소프트웨어융합대학  
소프트웨어학부

# 캡스톤 디자인 I

## 종합설계 프로젝트

프로젝트 명	Silver Watch
팀 명	우리도 차겠지
문서 제목	결과보고서

Version	1.5
Date	2020-JUN-09

팀원	오 윤 재 (조장)
	박 현 서
	사 드
	송 지 영
	이 수 정
	정 지 현

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

#### CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 개설 교과목 캡스톤 디자인 수강 학생 중 프로젝트 "Silver Watch"를 수행하는 팀 "우리도 차겠지"의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 "우리도 차겠지"의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

## 문서 정보 / 수정 내역

<b>Filename</b>	SilverWatch_수행결과보고서.doc
<b>원안작성자</b>	오윤재, 박현서, 사드, 송지영, 이수정, 정지현
<b>수정작성자</b>	오윤재, 박현서, 사드, 송지영, 이수정, 정지현

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2020-05-31	전원	1.0	최초 작성	
2020-06-02	전원	1.1	내용 기입	수정된 연구내용 추가
2020-06-05	전원	1.2	내용 기입	연구결과 및 결과물 추가
2020-06-07	전원	1.3	내용 추가	시스템 구조 추가 설명 기입
2020-06-08	전원	1.4	내용 추가	메뉴얼 추가
2020-06-09	전원	1.5	내용 추가	자기평가 기입 및 마무리

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 목 차

1	개요 .....	4
1.1	프로젝트 개요 .....	4
1.2	추진 배경 및 필요성 .....	5
2	개발 내용 및 결과물 .....	7
2.1	목표 .....	7
2.2	연구/개발 내용 및 결과물 .....	8
2.2.1	연구/개발 내용 .....	8
2.2.2	시스템 기능 요구사항 .....	13
2.2.3	시스템 비기능(품질) 요구사항 .....	14
2.2.4	시스템 구조 및 설계도 .....	15
2.2.5	활용/개발된 기술 .....	20
2.2.6	현실적 제한 요소 및 그 해결 방안 .....	20
2.2.7	결과물 목록 .....	21
2.3	기대효과 및 활용방안 .....	22
3	자기평가 .....	23
4	참고 문헌 .....	26
5	부록 .....	28
5.1	사용자 매뉴얼 .....	28
5.2	운영자 매뉴얼 .....	30
5.3	배포 가이드 .....	30
5.4	테스트 케이스 .....	33

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

# 1 개요

## 1.1 프로젝트 개요

최근 소프트웨어는 하드웨어와 결합해 사람들의 일상에 확실하게 기여할 수 있는 분야에서 각광받고있다. 또한, 인공지능을 통한 빅데이터 학습을 기반으로 해서 많은 서비스들이 선보여지고 있다. 우리는 이러한 상황에서 헬스케어와 노인복지가 동시에 이루어질 수 있는 웨어러블용 어플리케이션과 해당 웨어러블 기기의 데이터를 살펴볼 수 있는 기능을 탑재한 관리자용 어플을 제작하려고 한다.

노년층에게 가장 쉽게 발생하면서도 조기 발견이 어려운 낙상사고에 대한 감지를 웨어러블 내장 3축 가속도센서 데이터를 학습시켜 가능하게 한다. 웨어러블 기기 등록을 통해서 해당 기기 착용자의 위치, 상태를 확인할 수 있는 모바일용 어플을 제작한다. 현재 낙상 감지는 애플워치5 이외에 다른 기기들에서 감지 및 예측이 가능한 사례가 발견되지 않았다. 낙상사고에 취약한 노년층과 그 보호자들을 위한 낙상감지 기능과 함께 치매노인을 위한 위치확인 서비스를 제공하는 노인 맞춤형 어플리케이션 제작에 초점을 맞췄다.

본 프로젝트에서 제공하는 기능으로는

1. 낙상 시 관리자 어플에 알림
2. 노인의 실시간 위치 감지
3. 착탈 감지를 통한 사용자의 활동 체크

등이 있다.

웨어러블 기기를 착용하고 사용함을 통해서 관리자, 보호자가 노인의 건강을 대략적으로 파악하고, 소재를 간단하게 파악하는 것을 가능하게 하여, 기술이 약자를 보호하는 것을 목표로 하는 시대에 발맞춘 프로젝트를 목표로 했다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 1.2 추진 배경 및 필요성

### 1.2.1 사회 현황

2016년부터 소비자위해감시 시스템에 수집된 65세 이상 고령자의 안전사고 2만 2천여건 중, 낙상사고가 56%에 달한다. 이 가운데 44%는 골절 증세를 겪어, 심각한 부상으로 이어진다. 또한 한국소비자원이 조사한 2013~2015년 고령자 낙상사고 주요 장소는 주택(72.1%)이다. 질병관리 본부 조사에서도 노인 10명 중 6명이 집에서 낙상을 입는 것으로 파악된다. 이처럼, 노인의 낙상사고는 익숙한 장소인 자택에서 발생한다는 것이 문제이다.

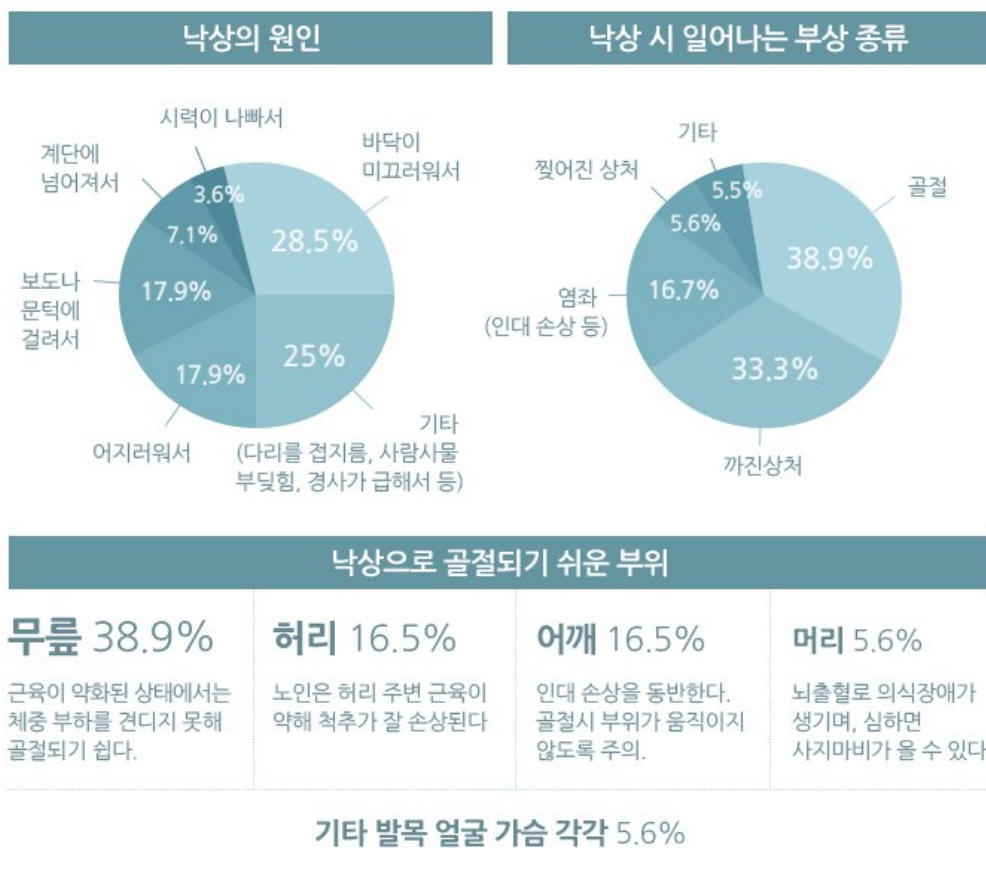


Figure 1 출처: 대한노인재활학회, 생명보험협회, 손해보험협회

낙상으로 인한 주요 손상은 고관절 골절로, 합병증을 유발 할 수 있어 2차적인 피해가 발생할 수 있다. 고관절골절을 입게 되면 엄청난 통증과 함께 허벅지쪽 출혈이 있기 때문에, 더더욱 주변 사람의 조속한 발견과 처치가 필요하다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

또한 노인 케어의 큰 문제로는 바로 치매노인의 실종이다. 치매노인 실종 신고는 2014년 8207건, 2015년 9046건, 2016년 9869건 그리고 2017년 10308건으로 매년 증가하고 있다. 실종 1명 당 평균 9명의 경찰관의 투입을 필요로 한다. 치매 노인은 일반인보다 인지 능력이 떨어져 구체적인 목적지가 없다. 그래서 치매 노인 발견에 걸리는 평균 시간은 11.8시간이다. 2018년 부터 치매 노인 약 6000명에게 배회 감지기를 지급했다. 이들 중 25명이 실종 되었으나 전원 발견되었고, 실종자 발견 평균 시간이 1.2시간으로 배회 감지기가 없는 케이스에 비해 평균 10.6시간을 단축하였다. 이는 웨어러블 기술 활용이 인력자원을 절약시키고 치매노인의 실종 시간을 현격히 감소시킬 수 있음을 보여준다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 2 개발 내용 및 결과물

### 2.1 목표

앞선 사례들을 통해 고령화 시대인 오늘날에 돌봄이 필요한 노인의 건강한 삶을 지원하는 노인 종합 케어시스템 구축을 목표로 Wear OS(스마트 워치)용 어플리케이션과 안드로이드 관리자 어플리케이션을 개발했다.

복합적인 노인의 건강 문제를 해결하기 위해 우리 조는 노인의 낙상사고 발생 시, 사고를 감지하고 가족이나 주변 보호자에게 알림을 보내는 기능을 이용하여 알림을 보내 즉각적인 대응을 할 수 있도록 지원하고자 한다.

기존의 유사 서비스와 차별화되는 핵심 요소들은 통합적인 노인 케어가 가능한 시스템을 워치 기기와 스마트 어플 한쌍에 담았다는 점이다. 기존의 유사 서비스들은 특정한 하나의 기능만을 제공했다. 이 프로젝트는 낙상과 스마트 워치 착용을 통한 활동 감지 및 위치 확인 기능을 gps, 3축 가속도 센서 등을 이용하여 하나의 종합 시스템을 구축하고자 목표했다. 또한 국내 점유율 1위인 안드로이드 OS를 기반으로 wear os와 안드로이드 관리자 어플리케이션을 사용하여 연령대가 높은 사용자도 친숙하게 사용이 가능할 수 있도록 고려하고자 한다. 또한 고연령자를 고려한 디자인을 제작하도록 노력한다. 디자인 측면에서는 사용자의 연령대를 고려한 디자인을 시도할 예정이다. (아이콘의 크기, 텍스트 정보의 크기 등)

스마트 워치의 작은 화면과 스마트 기기에 다루기 어려워 할 수 있는 노년층을 위해, 스마트 워치 앱에서 사용되는 기능은 최대한 자동화한다.

초고령 사회로 나아가는 시대에 맞게 유사 서비스와 달리 노인과 부양의 의무를 가진 보호자 모두의 부담을 덜고 도움이 될 수 있는 프로젝트를 목표로 한다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 2.2 연구/개발 내용 및 결과물

### 2.2.1 연구/개발 내용

#### 1. 머신러닝 :

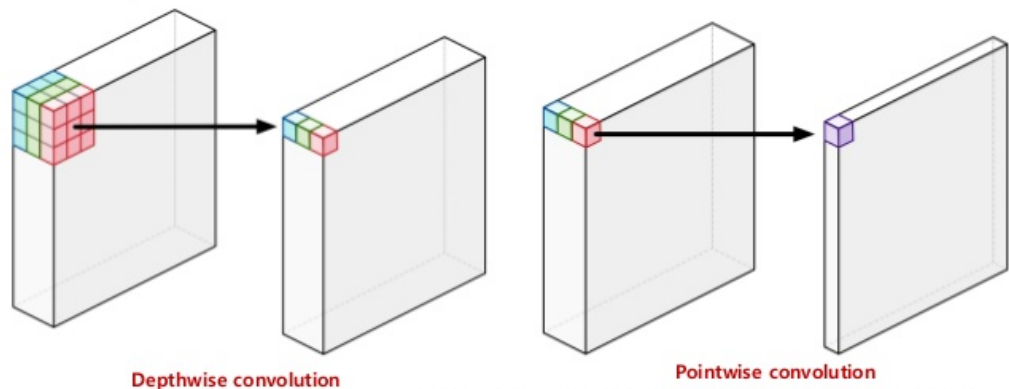
- 사용할 기술 : Pytorch, Flask
- 수집된 3 축 가속도계 센서 데이터를 갖고 사용자의 낙상 패턴을 분석해, 낙상을 감지한다.

#### 1.1머신러닝 연구/개발 내용

개발 초기에는 시계열 데이터를 처리하기 위해 RNN의 일종인 LSTM모형을 사용하려고 했으나 시계열 데이터를 분류하는 데에는 적합하지 않고 학습의 결과가 좋지 않아 모형을 Separable Convolution 사용하는 것으로 변경하게 되었다.

### Depthwise Separable Convolution

- Depthwise Convolution + Pointwise Convolution(1x1 convolution)



Figures from <http://machinethink.net/blog/googles-mobile-net-architecture-on-iphone/>

#### <Separable Convolution>

Separable Convolution은 기존의 convolution보다 parameter의 수와 연산량이 훨씬 적다. 해당 모형을 pytorch를 이용하여 예제를 참고해 구현하였다.

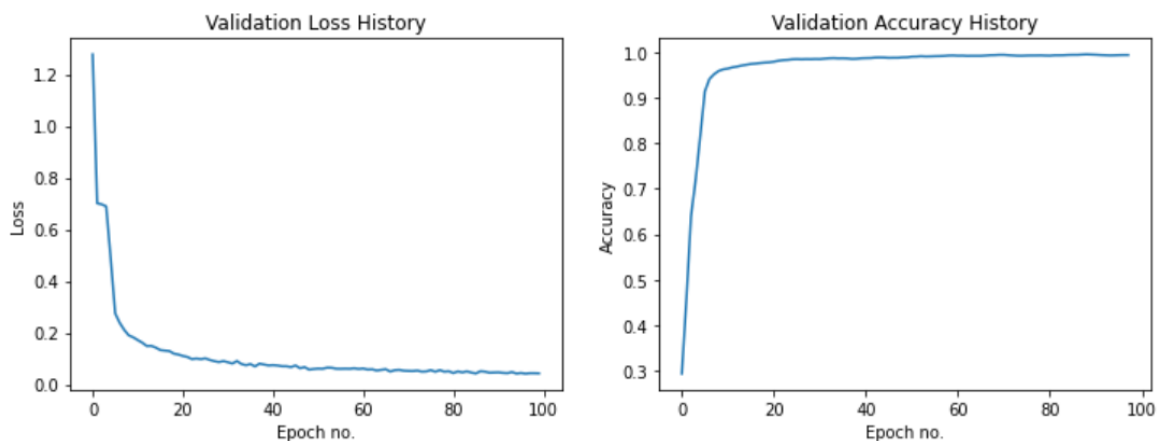
모델 학습에 들어가는 데이터셋의 경우 조원들 중 기기를 실제로 가지고 있는 5인이 참여하여 직접 수집했다. 결과적으로 낙상 데이터 500개와 걷기 데이터 700개를 수집하였다. 모델 학습 시 시리즈 길이를 정해 주어야 했다. 낙상 데이터 평균



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

에 맞추어 시리즈 길이를 30으로 맞추고, 걷기 데이터와 낙상 데이터의 데이터셋 확장을 위해서 길이 30을 넘는 자료의 경우 해당 자료를 잘라 새로운 데이터셋으로 추가하여 사용하였다. 그렇게 데이터셋 확장을 진행했고 걷기 데이터는 평균 길이가 매우 길어 데이터셋이 7천개 정도로 증대되었지만 낙상데이터는 평균 길이가 짧아 700개 수준이었다. 데이터의 불균형을 해결하기 위해서 낙상 데이터에 upsampling 기법을 적용해 낙상 데이터셋의 경우  $x, y, z$  가속도계 값에서 사람의 행동으로 차이가 날 수 없는 수준의 센서 값 변동을(수학적으로) 주고 데이터셋을 더 확장했다. 수치적 변화로 낙상 데이터를 걷기 데이터와 균형을 맞추어 7천개 수준에서 train, test 데이터셋을 제작해 모델을 학습시켰다. 초기에는 걷기와 낙상만을 분류했으나, 일상생활에서도 위치를 계속 착용할 것으로 예상하고 일상생활에서의 데이터도 수집(앉아있는 상태, 안정적으로 손목을 움직이지 않는 상태 등)을 추가 진행하여 학습시켰다.

## 1.2 머신러닝 결과물



< epoch에 따른 검증 데이터에 대한 정확도와 오차>

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

		예측	
		Positive	Negative
실제	Positive	1989	11
	Negative	21	1979

<테스트 데이터셋에 대한 모델 분류 성능>

## 2. 안드로이드 앱 개발

- 언어: Java, Kotlin
- 모바일 관리자 어플 :

### 1. 관리자가 보호 대상의 위험상황을 알 수 있도록 알림 기능을 사용한다.

#### A. 낙상 감지

#### B. 노약자의 웨어러블 기기 탈의 여부 확인

DB 로부터 user 의 고유한 watch\_id 를 받아와서 그에 해당하는 사용자가 위치를 착용하고 있는지를 확인하였다. 그 결과는 앱을 통해 recyclable view 로 리스트 형식으로 보여줬다.

#### C. 웨어러블 기기의 배터리 확인

DB 로부터 user 의 고유한 watch\_id 를 받아와서 그에 해당하는 기기의 배터리 잔량을 확인하였다. 그 결과는 앱을 통해 recyclable view 로 리스트 형식으로 보여줬다.

### 2. 노약자의 GPS 위치를 실시간으로 열람 가능

DB 서버로부터 user 들의 정보를 확인하고 그 user 들의 고유한 watch\_id 를 통해 사용자들의 위치를 DB 서버로부터 받아오는 방식을 선택하였다. 이 과정에서 서버팀과 논의하며 앱의 코드와 서버의 코드를 각자 수정하였다.

### 3. 다수의 노약자와 관리자 등록 연결 가능

스마트 워치의 Wear OS 로부터 생성된 QR 코드로 스마트 워치의 고유 watch\_id 를 받아오는 기능으로 위치 사용자를 등록하는 기능을 구현했다. 고유한 워치의 아이디를 어떻게 받을지 고민이었는데 Wear OS 에서 QR 코드를 생성하는 방향으로 해결하였다. 그리고 이 QR

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

코드를 관리자 앱에서 스캔하여 사용자의 이름과 전화번호를 받아서 등록하는 방식을 선택하였다.

- Wear OS 어플 :

1. 데이터 수집 앱 개발 - 센서에서 데이터를 수집하고, 누르는 버튼에 따라서 자동적으로 데이터에 라벨링을 하는 것을 목표로 한다.
  - A. 3 축 가속도 센서에 접근하여 데이터를 읽어오는 작업을 먼저 수행하였다.
  - B. 그 후 버튼을 사용하여 데이터 수집을 컨트롤할 수 있는 기능을 제작하였다. 버튼은 총 3 개로 나뉘며, 걷기(일상생활) 데이터 수집 시작, 넘어짐 데이터 수집 시작, 데이터 수집 종료로 나뉜다. 시작 버튼을 무엇을 눌렀냐에 따라서 데이터가 자동적으로 S3 의 다른 버킷에 저장된다.
  - C. 가속도 센서 데이터는 큐를 사용하여 순차적으로 저장하고, 데이터 수집을 종료하는 버튼이 눌리면 자동적으로 큐에 쌓인 데이터를 전송하고 큐를 비우도록 하였다.
2. 실 사용자용 앱 개발 - 최초 실행 시에는 등록 과정을 거친 후 실제 서비스를 실행하고, 그 후에는 등록 과정 없이 실제 서비스가 바로 실행되도록 제작한다. 실제 서비스는 수집된 데이터를 서버에 자동적으로 전송하는 작업을 의미한다.
  - A. 센서 데이터가 필수적인 앱이므로 위 데이터 수집 앱과 같이 센서 데이터를 읽어오는 작업을 먼저 수행하였다. 위와는 다르게 사용할 센서가 늘어나 작업을 분리하여 진행하였다. 사용한 센서로는 착용 감지 센서, 3 축 가속도 센서, GPS 등이 있으며 부가적으로 배터리 잔량도 불러올 수 있도록 작업하였다.
  - B. 그 후 관리자 앱과 연동할 수 있도록 고유한 번호를 생성하기 위해서 UUID 를 생성하였고, 연동 과정의 편리함을 위해서 UUID 를 데이터로 가지는 QR 코드를 만들어서 관리자 앱에서 촬영할 수 있도록 작업하였다.
  - C. 연동 작업이 완료된 것을 자동적으로 서버에 요청하는 방식으로

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

확인하고, 완료되었다면 각종 데이터를 수집하는 서비스를 실행한다.  
 서비스는 시계에서 수집한 각종 데이터를 설정된 주기에 맞춰서  
 서버에 자동적으로 전송한다.

D. 위 작업을 수동으로 하도록 하기에는 대상 연령층 및 시계 플랫폼  
 자체의 한계로 화면이 작아서 할 수 있는 작업이 한정되어 있기  
 때문에 실사용에 필요한 부분은 최대한 자동화 할 수 있도록 하였다.

### 3. Database 구축

- 기술 : MySQL
- 노약자의 GPS 값을 위도와 경도로 실시간 저장한다.
- 노약자의 기본 정보 저장(노인의 이름, 전화번호)한다.
- 스마트 워치의 unique key 값을 저장한다.
- 저장되는 데이터는 해당 시간(date time)과 함께 기록된다.

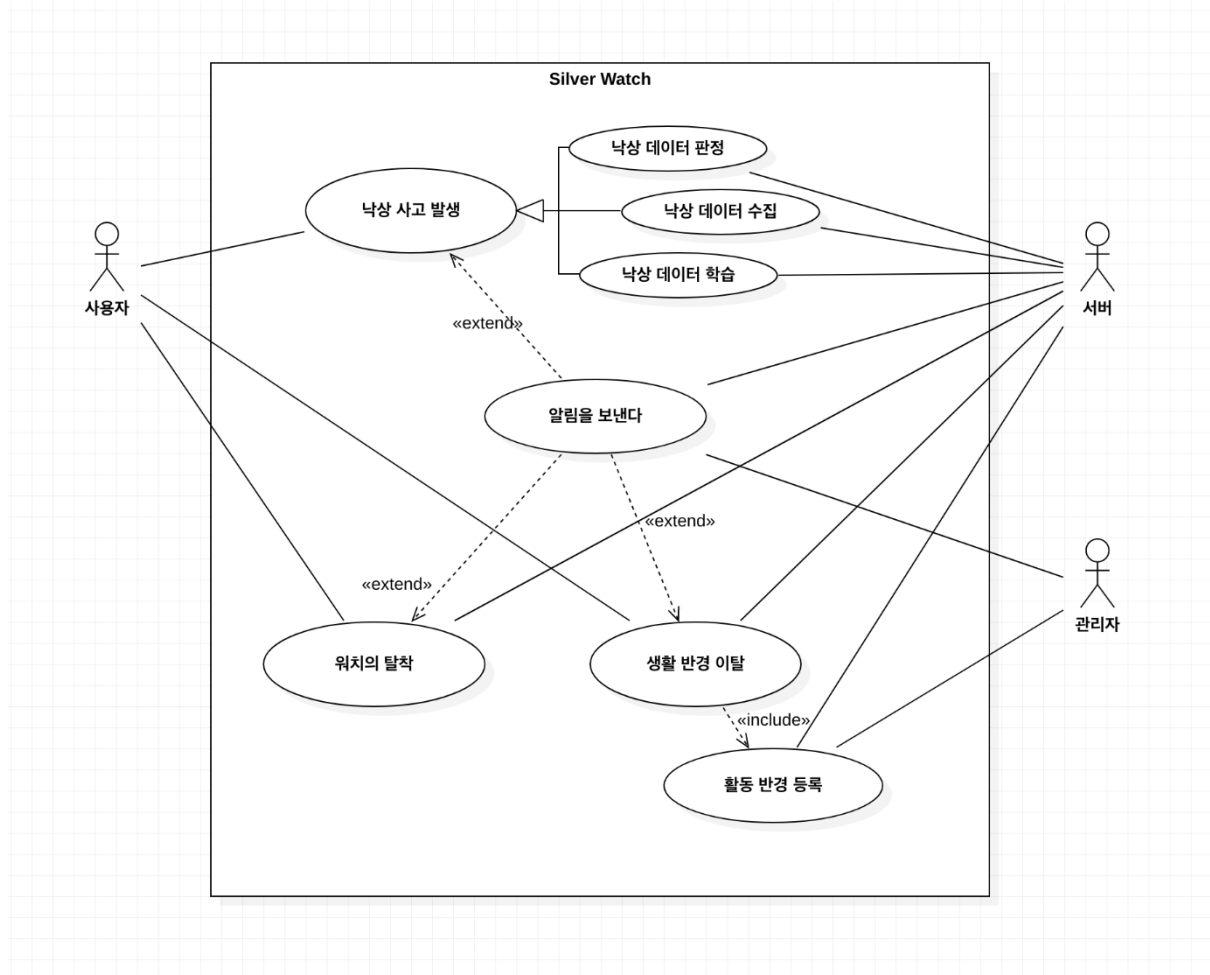
### 4. API 서버 구축

- 라이브러리 : Flask
- 스마트 워치에서 받은 데이터를 DB 에 저장하고 이를 분석한 Pytorch 로 낙상  
감지시 안드로이드 관리자 앱에 해당 이벤트를 전송한다.
- 안드로이드 관리자 앱에서 노인의 데이터 열람 요청시 DB 에 저장되어있는  
데이터를 관리자 앱에 전달한다.

### 5. 데이터 수집용 서버 구축 - 걷기 데이터와 넘어짐 데이터를 분류하여 저장하는 서버를 구축한다.

- AWS 의 S3 과 lambda, API Gateway 를 사용하였다.
- API Gateway 를 통해 lambda 에 요청을 보낸다.
- lambda 는 받은 데이터와 라벨을 이용하여 데이터를 담고 있는 파일을  
만들고, 받은 라벨을 이용하여 S3 의 다른 버킷에 저장한다.
- 파일명 겹침을 방지하기 위해서 UUID 를 사용하였고, 혹시 데이터가 섞였을  
때 구분을 하기 위해서 UUID 뒤에 walk / fall 을 붙여서 최종적인 파일을  
만들게 된다.

## 2.2.2 시스템 기능 요구사항



Use Case Diagram

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

기능적 요구사항	내용	상태
낙상	낙상 데이터 판정	완료
	낙상 데이터 수집	완료
	낙상 데이터 학습	완료
위치의 탈착	위치 탈착 유무 판정(알림 서비스에서 관리자 앱 열람 서비스로 변경)	변경
생활반경	생활 반경 등록 및 이탈 판정	미완료

### 2.2.3 시스템 비기능(품질) 요구사항

#### 1) 낙상 감지 시스템의 요구사항 (달성)

- 실시간으로 낙상을 판단해야 되는 경우는 낙상의 특성상 1 분 1 초가 아깝다. 따라서 낙상 여부를 판단하는데 3 초 이내에 판단할 수 있도록 한다. 3 초로 잡은 이유는 시계 - Lambda - S3 - Pytorch - S3 을 거치게 되는 설계 상 네트워크 지연을 감안하지 않을 수 없으므로 3 초를 허용치로 잡았다.

#### 2) 위치 정보 시스템의 요구사항 (달성)

- 위치 정보를 전송하는데 걸리는 시간은 1 초를 넘지 말아야 하며, 모든 과정은 자동적으로 진행되어야 한다.

#### 3) 기기 등록 시스템의 요구사항 (달성)

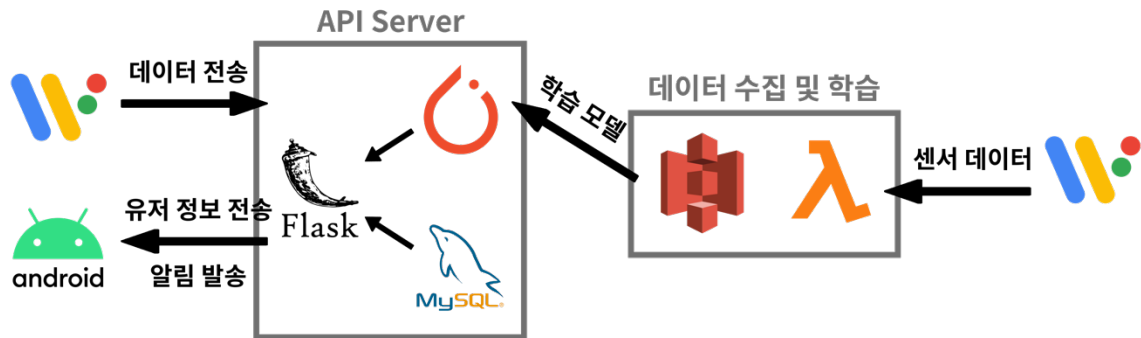
- 계정 별 ID 는 고유해야 한다.

#### 4) 배터리 및 착/탈 알림 시스템의 요구사항 (변경)

- 기존의 배터리 알림 시스템에서 스마트 위치의 배터리 열람 서비스로 변경하였다.
- 스마트 위치 착/탈 알림 시스템에서 스마트 위치 착/탈 여부 열람 서비스로 변경하였다.
- 웨어러블 기기의 핵심 기능은 위급상황에 대한 알림이며 이에 따라 낙상사고에 대한 정보를 최우선해야 한다. 보다 위급한 상황에서만 알림을 사용하는 것이 중요하다 판단하여 배터리와 탈착유무는 열람 서비스로 변경하였다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 2.2.4 시스템 구조 및 설계도



- 기존의 시스템 구조에서 변경된 사항은, 스마트 워치 사용자에게 보내지는 알림을 최소화하여 시스템을 자동화한 점이다. 이전에는 낙상사고 발생 시, 노인에게도 알림을 보내, 실제 낙상 사고가 발생한 것인지 확인하는 작업을 걸쳤지만, 이는 스마트 워치 착용자가 응급상황에 놓여있을 때, 보호자가 즉각적인 대응을 하기 어렵다는 판단을 하여 변경하게 되었다.

Use case name	낙상 알림
Scenario	노약자의 낙상사고 발생 시 관리자 어플에 알림
Triggering event	노약자의 낙상사고 발생
Brief description	낙상사고 발생 시, 웨어러블 기기가 감지하여 관리자 어플에 알림을 보낸다. 알림을 받은 보호자는 노약자의 상황을 파악하고 대처 할 수 있다.
Actors	노약자, 웨어러블 기기, 관리자 어플, 보호자
Related use cases	extend : {condition : 낙상인지 아닌지를 노약자가 선택하면 데이터를 반영해서 낙상 판단을 하는 알고리즘을 보완할 수 있게 한다.}
Stakeholders	<ul style="list-style-type: none"> <li>- 노약자는 본인의 낙상사고를 보호자에게 알려, 즉각적인 도움을 받고 싶어 한다.</li> <li>- 보호자는 노약자의 상황을 판단하고 대응을 하고 싶어 한다.</li> <li>- 어플리케이션 제공자는 노약자의 자이로센서 데이터를 분석하여 보다 정확한 알고리즘으로 개선하기를 바란다.</li> </ul>

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

Preconditions	낙상사고시 노약자가 웨어러블 기기를 바르게 착용하고 있다. 기기가 낙상사고인지를 정확하게 판단해서 관리자 어플에 전달할 수 있다.					
Post conditions	돌발적 상황에서 보호자가 노약자의 상황을 파악하고 도움을 요청할 수 있게 한다.					
Flow of activities	<table> <tr> <th>사용자</th> <th>서버</th> </tr> <tr> <td> 1. 사용자가 기기를 착용한 상태로 낙상사고를 경험한다.   2. 낙상사고임을 확인한다.   3. 관리자 어플의 알림을 통해 빠른 시간 내에 조치를 받는다. </td> <td> 1.1 낙상사고 데이터를 전송받는다.  1.2 낙상사고 데이터를 DB에 저장한다.   2.1 낙상사고임을 확인한 것을 전송받는다.  2.2 낙상사고 알림을 해당 기기의 관리자 어플에 전송한다.   3.1 서버는 추가된 데이터를 가지고 온라인 학습을 주기적으로 진행한다. </td> </tr> </table>		사용자	서버	1. 사용자가 기기를 착용한 상태로 낙상사고를 경험한다.  2. 낙상사고임을 확인한다.  3. 관리자 어플의 알림을 통해 빠른 시간 내에 조치를 받는다.	1.1 낙상사고 데이터를 전송받는다. 1.2 낙상사고 데이터를 DB에 저장한다.  2.1 낙상사고임을 확인한 것을 전송받는다. 2.2 낙상사고 알림을 해당 기기의 관리자 어플에 전송한다.  3.1 서버는 추가된 데이터를 가지고 온라인 학습을 주기적으로 진행한다.
사용자	서버					
1. 사용자가 기기를 착용한 상태로 낙상사고를 경험한다.  2. 낙상사고임을 확인한다.  3. 관리자 어플의 알림을 통해 빠른 시간 내에 조치를 받는다.	1.1 낙상사고 데이터를 전송받는다. 1.2 낙상사고 데이터를 DB에 저장한다.  2.1 낙상사고임을 확인한 것을 전송받는다. 2.2 낙상사고 알림을 해당 기기의 관리자 어플에 전송한다.  3.1 서버는 추가된 데이터를 가지고 온라인 학습을 주기적으로 진행한다.					
Exception conditions	낙상 사고로 확인되지 않은 경우(기기의 판단오류일 경우) -> 해당 데이터도 오류데이터로 서버가 저장해뒀다가 온라인 학습시 사용한다.					



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

Use case name	관리자 앱에서 기기 등록							
Scenario	보호자가 노약자의 웨어러블 기기를 휴대폰 어플에 등록							
Triggering event	보호자가 노약자의 건강상태 및 활동량 관리를 하고싶어 한다							
Brief description	노약자에게 웨어러블 기기(스마트 워치)를 착용하도록 하고, 보호자는 본인의 스마트 폰에 웨어러블 기기 관리 어플을 다운 받는다. 노약자의 기기 등록을 통해 해당 인물의 지속적인 건강관리를 할 수 있다.							
Actors	노약자, 보호자							
Related use cases	extends : {condition : 보호자가 여러 노약자를 관리함, 노약자가 여러 보호자에 등록될 수 있음.}							
Stakeholders	<ul style="list-style-type: none"><li>- 보호자는 관리자 어플에 노약자의 기기가 제공하는 센서 데이터가 정확하게 나타나길 바란다. (노약자는 노인 혹은 자녀에 해당)</li><li>- 노약자는 기기를 착용하는 동안 자신의 상황에 대한 정보가 관리자 어플을 통해 보호자에게 빠르게 전달되기를 바란다.</li></ul>							
Preconditions	노약자의 웨어러블 기기가 관리자 어플에 실시간 정보 제공을 할 수 있다.							
Post conditions	보호자는 관리자 어플을 통해 등록된 웨어러블 기기를 확인할 수 있다. 정상적으로 등록이 완료되고 난 후 해당 기기들에서 오는 데이터를 확인할 수 있다.							
Flow of activities	<table><tr><th>사용자</th><th>서버</th></tr><tr><td>1. 사용자는 보호자의 모바일 기기에 자신의 기기를 등록한다.</td><td>1.1 DB에 해당 관계정보를 저장한다.</td></tr><tr><td>2. 관리자 앱에서 관계된 기기 정보를 확인할 수 있다.</td><td>2.1 관리자 앱 로그인시 DB에서 관계 정보를 가지고와서 보여준다.</td></tr></table>		사용자	서버	1. 사용자는 보호자의 모바일 기기에 자신의 기기를 등록한다.	1.1 DB에 해당 관계정보를 저장한다.	2. 관리자 앱에서 관계된 기기 정보를 확인할 수 있다.	2.1 관리자 앱 로그인시 DB에서 관계 정보를 가지고와서 보여준다.
사용자	서버							
1. 사용자는 보호자의 모바일 기기에 자신의 기기를 등록한다.	1.1 DB에 해당 관계정보를 저장한다.							
2. 관리자 앱에서 관계된 기기 정보를 확인할 수 있다.	2.1 관리자 앱 로그인시 DB에서 관계 정보를 가지고와서 보여준다.							
Exception conditions	모바일 통신환경이 원활하지 않은 경우 서버와의 접속이 불안정해 등록이 제대로 되지 않을 수 있음 -> 서버 접속 실패시 관리자 어플리케이션에서의 대처가 필요							

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

Use case name	치매노인을 위한 GPS					
Scenario	보호자가 관리자 어플에서 치매노인의 웨어러블 기기 위치를 파악					
Triggering event	보호자는 치매 노인의 실종을 대비해 해당 환자의 위치를 실시간으로 제공받고 싶어한다.					
Brief description	- 보호자가 관리자 어플에 들어가, 관리 목록에 있는 해당 치매노인의 항목을 누른다. 관리 항목 중, GPS 추적을 눌러 노약자의 위치를 볼 수 있다.					
Actors	노약자, 보호자, 웨어러블 기기, 관리자 어플					
Related use cases	X					
Stakeholders	- 보호자는 노약자의 안전을 위해 해당 노약자의 위치를 실시간 추적하길 바란다.					
Preconditions	치매노인의 위치를 보호자가 실시간으로 확인할 수 있다.					
Post conditions	보호자가 비상상황 발생 시 노약자의 위치를 파악하고 상황을 실시간으로 대응할 수 있다.					
Flow of activities	<table><tr><th>사용자</th><th>서버</th></tr><tr><td>1. 사용자가 기기를 착용한채로 등록된 지역정보를 이탈한다.  2. 이탈 후의 위치정보를 통해서 빠른 조치가 가능하다.</td><td>1.1 기기는 gps정보를 일정한 간격으로 서버를 통해 DB에 전송한다. 1.2 gps정보가 등록된 지역정보에서 벗어나 있는지 확인한다.</td></tr></table>		사용자	서버	1. 사용자가 기기를 착용한채로 등록된 지역정보를 이탈한다.  2. 이탈 후의 위치정보를 통해서 빠른 조치가 가능하다.	1.1 기기는 gps정보를 일정한 간격으로 서버를 통해 DB에 전송한다. 1.2 gps정보가 등록된 지역정보에서 벗어나 있는지 확인한다.
사용자	서버					
1. 사용자가 기기를 착용한채로 등록된 지역정보를 이탈한다.  2. 이탈 후의 위치정보를 통해서 빠른 조치가 가능하다.	1.1 기기는 gps정보를 일정한 간격으로 서버를 통해 DB에 전송한다. 1.2 gps정보가 등록된 지역정보에서 벗어나 있는지 확인한다.					
Exception conditions	X					

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

### 1) 낙상 감지 시스템

- 낙상 감지 시스템은 크게 학습 시스템과 판단 시스템으로 나눌 수 있다. 학습 시스템은 수집한 센서 데이터와 라벨을 가지고 낙상 여부를 구별할 수 있도록 학습하고, 판단 시스템은 시계에서 넘어온 데이터를 기반으로 학습된 모델을 통해 현재 넘어온 센서 데이터가 낙상에 해당하는지 판별하여 시계와 관리자 앱인 스마트폰에 알림을 띄우게 된다.
- 시스템이 낙상이라고 판단한 경우 시계에 알림을 띄우고 사용자가 실제 낙상인지 아닌지 버튼으로 선택하게 되는데, 이를 기반으로 데이터에 라벨을 붙여서 추가적인 학습을 하는데 사용한다.
- 데이터 수집은 API Gateway - Lambda- S3 으로 이어지는 흐름을 통해 S3 으로 저장된다. 그 후 수집된 데이터는 Pytorch 를 통해 학습되고, 학습된 모델은 다시 S3 에 저장된다.

### 2) 위치 정보 시스템

- 시계에서 전송한 위치 데이터는 API 서버를 통해 DB 에 저장된다. API 서버는 주기적으로 DB 의 데이터를 분석하여 현재 사용자의 위치를 실시간으로 열람 할 수 있다.

### 3) 기기 등록 시스템

- 사용자 앱인 위치 앱 - 관리자 앱인 스마트폰 앱 간 연동을 지원하는 시스템으로, 각각의 계정 마다 UUID 를 이용하여 고유 번호를 부여하고, 부여된 고유 번호간 연결을 통해 관리자 앱에서 위치 앱에서 전송받은 데이터를 확인할 수 있도록 한다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 2.2.5 활용/개발된 기술

- 1) 낙상 감지 알고리즘 : 현재 데이터 셋의 크기가 작아 오버피팅 되었지만, 3 축 가속도계 센서의 값을 연속적으로 크게 맞춰 입력하는 경우, 해당 입력이 낙상인지 아닌지를 판단해주는 알고리즘을 학습시켜 개발했다.
- 2) Pytorch : 센서 데이터 학습에 활용했다.
- 3) Flask : 학습시킨 모델을 api 로 제공하기 위해 활용했고, 안드로이드 관리자앱에서 필요한 정보를 서버 api 를 통해 제공했다.
- 4) FCM : 스마트 워치에서 낙상이 판단되면, 이 상황을 안드로이드 관리자 어플리케이션에 알림으로 보낸다.

## 2.2.6 현실적 제한 요소 및 그 해결 방안

### 2.2.6.1 하드웨어

- 시계의 배터리 사용 문제 : GPS 사용 시 배터리의 사용 시간이 급격하게 짧아지는 문제가 있으므로 우선 GPS를 실시간으로 계속 사용하는 것이 아닌, 몇 분 단위로 샘플링하여 사용한다.

### 2.2.6.2 소프트웨어

- 동작 구별 문제 : 현재 조에서 보유하고 있는 스마트 워치의 종류와 개수가 한정되어 있어 기본적인 보행 동작과 실제로 넘어지는 동작의 구별이 어려울 수 있었다. 하지만 이는 조원들의 직접적인 일상 데이터와 낙상 데이터 수집을 통하여 스마트 워치 사용자의 동작을 분류할 수 있었다.

### 2.2.6.3 기타

- 통신 대역폭 문제 : 스마트 워치와 스마트 폰 간의 연결은 대역폭이 작아 대용량 데이터 전송에 제한이 있으나 이는 서버를 통한 송수신으로 가볍게 진행할 수 있었고, 불가피한 대용량 데이터 전송의 경우 Wi-Fi 네트워크에 직접 연결하면 해결 할 수 있다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 2.2.7 결과물 목록

대분류	소분류	기능	형식	비고
파일	낙상 데이터	자이로센서와 가속도센서의 낙상 감지 알고리즘에 사용되는 데이터	CSV	낙상 데이터 셋 : 약 500개 일상생활 데이터 셋 : 약 700개
	GPS 자료	사용자의 기존 생활 행동 반경을 저장하는 데이터	CSV	
알고리즘	낙상 감지	자이로센서와 가속도센서의 데이터를 이용해 사용자의 낙상사고 발생을 감지한다	모듈	
입력	스마트 워치	사용자의 건강 상태 데이터를 서버에 전송한다	HW	
출력	어플리케이션	서버로부터 전달받은 정보를 어플리케이션에서 확인한다	APK	

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 2.3 기대효과 및 활용방안

본 프로젝트는 노인 복지의 자동화, 낙상 데이터 셋의 오픈소스화, 그리고 위치 자체의 활용방안으로 기대효과를 볼 수 있다.

우선 노인 복지 자동화의 경우, 스마트 워치에 사용될 Wear OS 앱은 기본적으로 백그라운드로 실행되기 때문에 노인이 수동으로 조작할 필요가 없다. 그래서 번거로움을 덜어줄 수 있으며, 예기치 못한 사고가 발생하여 노인이 타인에게 연락이 불가능한 상황의 경우, 자동화된 시스템을 통해 비상상황을 감지하여 즉각 대응을 할 수 있다.

또한, 낙상 데이터 셋의 오픈소스로 공개하여 낙상 감지 알고리즘을 개발하는 경우에 사용할 수 있도록 하면 차후 관련 개발을 시도할 때 활용할 수 있다. 이를 통해 하나 프로젝트에서 끝나는 것이 아니라, 사회에 도움이 될 수 있는 기반을 만들 수 있다.

노인 복지의 자동화와 데이터의 오픈소스화는 모두 노인뿐만 아니라 다양한 연령대나 도움이 필요한 장애인에게도 제공 가능하여 서비스 사용 가능 범위를 넓힐 수 있다. 그러므로 가정과 함께 보호시설까지 서비스 확장 가능성이 있다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

### 3 자기평가

오윤재	<p>종합적인 노인 케어 시스템을 제작하는 Silver Watch 프로젝트에서 나는 프로젝트 매니저와 api 서버 개발 및 데이터베이스 설계를 담당했다. 우리 조는 이번 프로젝트를 통해 모든 조원이 새로운 분야에 도전하게 되었는데, 모두 성공적인 시도와 앞으로의 발전 가능성을 보여준다고 생각한다.</p> <p>API 서버 개발로 안드로이드 개발자 앱에서 필요로 하는 정보를 MySql을 통해 저장하거나 불러오는 역할을 수행했다. Restful api를 사용하여 보다 효과적으로 처리할 수 있었다.</p> <p>그리고 데이터베이스 설계를 맡아, 실질적으로 데이터를 저장하고 불러오는 값들을 정하여 각 테이블 별로 필수적인 key값과 unique 값을 구별하여 설계했다. 또한 FCM을 이용하여 안드로이드 관리자 앱에 푸시 알림 서비스를 개발하였다.</p> <p>이번 프로젝트를 통해서 나는 새로운 개발 분야에 한발짝 내딛을 수 있었고, 비록 한정된 하드웨어 종류와 개수로 인해 아쉬운 점도 있었지만, 조원들과 함께 직접 낙상데이터와 일상데이터를 수집하는 과정을 통해 연구 분야에서의 수집 데이터 형식을 새로이 알게 되었다. 3개월간의 짧은 시간동안 새로운 기술과 분야를 공부할 수 있게 되어 소중한 경험이라고 생각한다.</p>
박현서	<p>이번 프로젝트에서 나는 관리자를 위한 안드로이드 앱을 개발하였다.</p> <p>또한 발표 때에는 발표 영상과 시연 영상을 만들었다.</p> <p>처음으로 장기간에 걸쳐서 팀원들과 함께 하는 프로젝트여서 걱정이 많았으나 팀원들의 헌신적인 도움과 교수님들의 피드백을 통해 성공적으로 마무리할 수 있었다.</p> <p>본 프로젝트의 목적이었던 스마트 워치를 활용한 노인들을 위한 종합 복지 시스템을 만들 수 있었으며 낙상을 감지하고 사용자에게 알림을 보내는 실버워치의 핵심적인 기능과 위치 추적과 같은 기능들을 실제로 구현할 수 있었다.</p> <p>이러한 기능들을 사용자와 이어주는 안드로이드 어플리케이션을 만드는 과정에서 서버와 정보를 주고받는 것과 데이터베이스를 활용하는 경험을 할 수 있어서 아주 유익한 프로젝트였다.</p> <p>또한 안드로이드 스튜디오를 활용하여 앱을 만들고 나니 앞으로도 이 경험을 토대로 많은 활용점을 찾을 수 있어서 좋았다.</p> <p>또한 GitHub와 Slack을 활용하여 팀원들은 물론 교수님들과 소통하며 업무 효율을 높이는 방법을 배운 것도 큰 경험이었다.</p> <p>특히 GitHub를 실전에서 사용하는 방법을 배울 수 있어서 유익한 프로젝트였다.</p>



송지영	본 프로젝트에서 낙상 감지 모델의 구현 및 학습, api와 모델의 연결, 그리고 시각자료의 제작을 맡았다. 인공지능 분야를 잘 알지 못해서 초기에 학습을 진행할 때 모델 선정에 어려움을 겪었다. 그래서 프로젝트 초기에 구현했던 모델 이랑 현재 모델이 아예 달라졌지만 그 과정에서 학습 성능이 더 좋은 모델을 선정할 수 있었다. 현재 모델은 최종적으로 테스트 데이터셋에 대해 99%의 정확도를 보인다. 하지만 데이터의 양이 충분치 않아 오버피팅을 막지 못한 것이 아쉽다. 향후 데이터의 양을 더 늘린다면 모델 성능의 향상을 기대할 수 있다. 또한 현재의 모델은 낙상과 낙상이 아닌것을 구분하는 것에 최적화되어 있지만 모델을 고도화 시킨다면 사용자의 행위를 감지하는 모델로도 활용할 수 있을 것으로 기대된다.
이수정	이번 프로젝트에서 나는 수집된 학습 데이터를 전처리하고 관찰한 후, 적절하게 데이터셋 학습에 활용될 수 있도록 라벨링하고 csv파일로 제작하는 부분, 그리고 학습이 완료된 모델을 불러와 api로 제작하는 부분, 발표 영상을 편집 및 제작하는 부분을 담당하였다. 데이터 셋을 관찰하고 수집을 관리하면서 불균형적인 데이터에 대해서 알게 되었고, 프로젝트를 진행하고 있는 상황(코로나 바이러스)에서 이 부분을 해결할 수 없어 아쉬웠다. 미리 만들어진 데이터 셋이 존재했다면 더 좋았겠다는 생각이 많이 들었는데, 이 부분을 우리가 제작한 데이터 셋을 오픈소스화 해서 제공하면 이후 낙상 감지 알고리즘을 개발하거나 3축 가속도계 센서값을 행동별로 분류하여 사용하고 싶은 사람에게 도움이 될 것으로 예상된다. 이후 상황이 개선되고 데이터를 더 다양하게 수집할 수 있는 상황이 된다면 프로젝트 자체를 더 발전시키거나 활용할 수 있다고 판단된다.
정지현	현재 고령화사회로 나아가고 있는 상황에서 낙상 및 치매 노인의 실종 문제는 심각한 문제이며 이러한 경우에 빠른 대처가 상당히 큰 도움이 될 수 있는데, 그 부분에 대해서 도움을 줄 수 있는 프로젝트를 진행하게 되어 의미가 있었다고 생각한다. 이 프로젝트를 진행하면서 안드로이드 스마트 워치 용 앱 개발과 서버 인프라를 담당했는데, 인프라 관련은 작년에 자율주행경진대회의 랩타임 측정 플랫폼을 만들면서 경험한 부분을 바탕으로 쉽게 처리할 수 있었지만 스마트 워치용 앱 개발은 아무래도 처음 해보는 부분이다 보니 시행착오를 많이 겪어서 개발 진도를 맞추지 못한 점이 아쉬움으로 남는다. 세부적으로는 안드로이드 스마트 워치에서 센서에 접근하여 데이터를 가져오고, 가져온 데이터를 AWS를 통해서 수집할 수 있도록 하는 작업을 첫번째로 하였으며 두번째로는 워치에 설치한 앱을 통해서 낙상 감지용 가속도 센서 데이터를 비롯하여 착용 / 미착용 감지 결과, 배터리 상태 등을 서버로 자동적으로 전송하도록 하는 작업을 수행하였다. 그리고 서버 인프라 관리로는 머신러닝 / api 서버 구동 등에 활용할 수 있는



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

	<p>서버를 구축하고, AWS 환경을 구축하여 데이터 수집을 편하게 할 수 있도록 하였으며 api 서버 및 낙상 감지 서버의 배포 관련 작업을 했다.</p> <p>스마트 워치용 앱에 더 많은 기능을 담고 싶었지만 경험 부족으로 기능을 원하는 만큼 넣지 못한 것이 가장 아쉬운 점으로 남지만 새로운 플랫폼에 대한 경험과 팀원들 각자의 역할 분담을 통해서 효율적으로 개발하는 경험을 할 수 있었다는 점에서 좋은 기회였다고 생각한다.</p>
--	---

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 4 참고 문헌

번호	종류	제목	출처	발행년도	저자	기타
1	기사	실버 건강, 지켜야 산다	중앙일보	2017/04/14	배지영	
2	기사	노인 낙상 사고 72%, 집에서 발생	헬스조선	2017/01/11	황인태	
3	기사	겨울철 고령자 낙상 사고 ‘주의’	KBS	2019/11/29	손은혜	
4	기사	치매실종 1 만명 시대…몰랐던 지원과 대처책은 없나요?	디멘시아 뉴스	2019/09/18	조재민	
5	연구보고서	치매노인과 돌봄제공자를 위한 맞춤형 정책방안 모색	한국보건사회연구원	2018/12	유재언	
6	기사	길 잃는 치매노인 ‘하루 평균 27 명’	중앙일보	2017/06/11	박상욱	
7	기사	독거노인 2명 중 2명은 ‘사회활동 없어’	웰페어뉴스	2019/06/03	정두리	
8	기사	독거노인 절반 “경로당도 복지관도 안나가”	조선일보	2019/06/01	홍준기	
9	논문	Collection and Processing of Data from Wrist Wearable Devices in Heterogeneous and Multiple-User Scenario	NCBI	2016/09/16	Francisco de Arriba-Pérez	
10	기사	Latest trends in medical monitoring devices and wearable health technology	Business Insider	2020/02/01	Alicia Phaneuf	
11	서적	Do it! 안드로이드 앱 프로그래밍(전자책)	이지스퍼블리싱	2020/02/21	정재곤	
12	서적	깔끔한 파이썬 탄탄한 백엔드:지금까지 없었던 백엔드 개발자를 위한 파이썬	비제이퍼블릭	2019/03/15	송은우	
13	웹페이지	(pytorch) deep time series classification	Kaggle	2019/04/20	Devforfu	<a href="https://www.kaggle.com/purplejester/pytorch-deep-">https://www.kaggle.com/purplejester/pytorch-deep-</a>

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

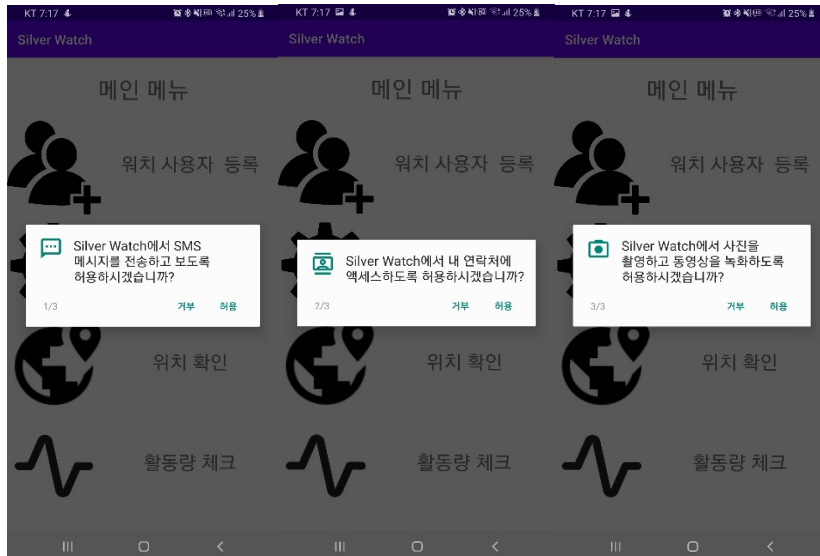
						<a href="#">time-series-classification/data</a>
14	웹 페이지	(linux) 백그라운드 작업 nohup 사용법	Github.io	2019/06/02	이성호	<a href="https://iseongho.github.io/posts/nohup/">https://iseongho.github.io/posts/nohup/</a>

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 5 부록

### 5.1 사용자 매뉴얼

#### 5.1.1 안드로이드 앱 권한 부여하기



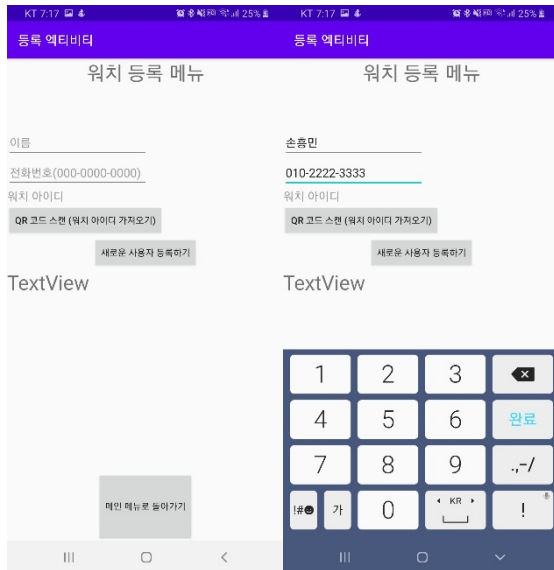
- 권한 부여 알림에 따라 '허용'을 누른다.
- 총 3개의 권한 부여에 대해 허용하면 완료
- 메시지 전송, 연락처 액세스, 사진 촬영에 대한 권한이 부여됨

#### 5.1.2 위치 사용자 등록하기

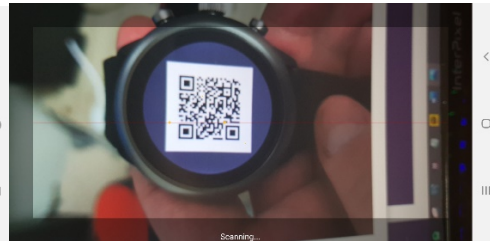
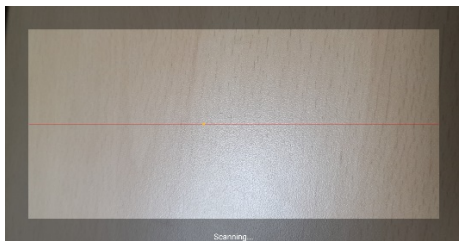


- 메인 메뉴에서 '위치 사용자 등록' 버튼 클릭

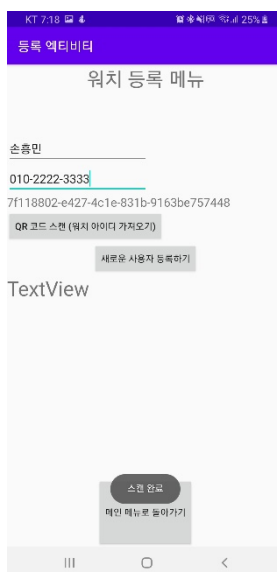
<div>  <div> <b>국민대학교</b>  <b>소프트웨어학부</b>  <b>캡스톤 디자인 I</b> </div> </div>	결과보고서		
	프로젝트 명	Silver Watch	
	팀 명	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09



- 워치 등록 메뉴에서 워치 사용자의 '이름', '전화번호'를 기입
- 작성 한 후에 'QR 코드 스캔' 버튼 누름



- QR 코드 스캔 모드가 실행되면 워치에서 생성되는 QR 코드를 스캔



- QR 코드를 스캔 한 후에 워치 ID가 화면에 표시됨
- '새로운 사용자 등록하기' 버튼을 누름
- 새로운 사용자가 서버에 등록됨

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 5.2 운영자 매뉴얼

### 5.2.1 H/W 구성

장치	모델명	비고
스마트 워치	해당 없음	Gps 센서, 3축 가속도 센서 탐재
스마트 폰	해당 없음	안드로이드 SDK 16 이상

### 5.2.2 S/W 구성

- 1) 낙상 데이터 학습시 필요한 개발 환경 :
  - Ubuntu 18.04
  - AWS Lambda
  - AWS S3
- 2) 안드로이드 개발 환경 :
  - 안드로이드 스튜디오 (minSDK 16)

## 5.3 배포 가이드

### 안드로이드 앱 & 웨어 앱

- 스토어에 배포하지는 못했으므로 adb (android debug bridge)를 이용하여 설치한다.
- 안드로이드 앱과 웨어 앱 모두 설정 - 정보에서 빌드 번호를 7 번 이상 연타하여 개발자 설정을 활성화 한 후, adb 를 활성화한다.
- 웨어 앱은 Wi-Fi 네트워크를 통해 설치해야하므로 컴퓨터와 같은 네트워크에 연결한다.
- 그 후 시계의 ip 주소를 알아낸 후,

```
adb connect {ip 주소}
```

- 명령어를 통해 시계와 연결한다.
- 그 후 앱을 설치한다. 명령어는 아래와 같다.

```
adb install -r {apkname}.apk
```

- adb 권한을 물어볼 수 있다. 허용한다.
- 완료되면 설치된 결과물을 볼 수 있다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

- 안드로이드 앱은 시계와 같은 방법으로 폰과 연결하거나, usb 를 통해 연결한다.
- 그 후 설치방법은 adb install 명령어를 위와 같이 사용한다.

#### api 서버

- pymysql, flask, flask\_restful, flask\_cors 모듈이 필요하다. 각각 pip 을 통해서 설치한다.
- 원래는 nginx 를 사용하려고 하였으나 도메인을 따로 만들기도 어려운 상황이고 간단히 uwsgi 서버만 따로 사용해도 된다고 판단하여서 nohup 을 통해서 실행할 수 있도록 했다.
- 프로젝트 폴더 안의 src/flask-server/silverwatch\_api 폴더로 들어간다.
- 다음 내용을 wsgi.py 로 저장한다.

```
from app import app

if __name__ == "__main__":
    app.run()
```

- 그 후 같은 폴더에서 다음 명령어를 입력한다.

```
nohup uwsgi --socket 0.0.0.0:5000 --protocol=http -w wsgi:app --enable-threads &
```

- 5000 번 포트로 서버가 실행된 것을 확인할 수 있다.

#### 낙상 감지 서버

- pytorch, numpy, pandas, flask 모듈이 필요하다. pip 을 통해서 설치한다.
- nohup 을 통해 model\_api.py 를 실행한다. 명령어는 아래와 같다.

```
nohup python3 model_api.py &
```

- 9000 번 포트를 통해서 서버가 실행된 것을 확인할 수 있다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

aws 푸시 서버

- src/aws/battery-alert 디렉토리의 py 파일과 zip 파일을 사용한다.
- zip 파일을 이용하여 lambda 의 layer 를 생성한다.
- py 파일은 전체 내용을 lambda 의 소스코드 부분에 붙여넣는다.
- 저장 버튼을 누르고 트리거 추가 버튼을 넣는다.
- EventBridge (CloudWatch Events)를 선택한다.
- 새 규칙을 누르고, 적절한 이름을 선택한다.
- 예약 표현식을 선택하고, cron(0 0-13 ? \* \* \*)를 입력한다.
  - o cron 의 시간은 UTC 기준이기 때문에 0-13 을 선택하여 한국 시간 9 시~22 시에 알림을 받을 수 있도록 한다. 매 시간마다 체크하여 알림을 발송하게 된다.
- 트리거를 추가한다.



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

## 5.4 테스트 케이스

대분류	소분류	기능	테스트 방법	기대 결과	테스트 결과
실행	자동 실행	스마트 워치 전용 앱의 서비스가 백그라운드에서 실행된다.	액티비티 상에 서비스 실행 결과를 출력하도록 한다.	Logcat 으로 실행 결과가 출력된다.	성공
정보 열람	등록 및 열람	스마트 워치 기기를 등록한다,	스마트 워치에서 QR 코드를 받아 회원가입을 받는다.	데이터베이스에 회원정보가 저장된다.	성공
낙상	감지	스마트 워치 사용자의 낙상을 감지한다.	스마트 워치를 착용한 상태로 넘어진다.	낙상 여부 결과가 나온다..	성공
센서	데이터 수집	가속도 센서에서 값을 가져와서 AWS S3 에 저장한다.	데이터 수집 앱을 작동시키고, 수집을 시작했다가 종료한다.	시작 버튼의 종류에 따라서 라벨링되고 분류되어 S3 에 저장된다.	성공
센서	데이터 전송	가속도 센서 및 GPS, 착용 감지 센서 등에서 값을 가져와서 API 서버에 전송한다.	시계를 착용하고 흔들어 보거나 위치를 이동하는 등의 변동을 만들어낸다.	각각 설정된 주기에 맞춰서 DB 에 값이 업데이트된다.	성공
정보 열람	위치 확인 및 열람	워치 사용자의 현재 위치를 확인한다	DB 서버로부터 위치 데이터를 watch_id 로 검색하여 가져온다.	워치 사용자의 가장 최근 위치를 가져와서 지도로 보여준다.	성공

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	Silver Watch	
	<b>팀 명</b>	우리도 차겠지	
	Confidential Restricted	Version 1.5	2020-JUN-09

정보 열람	배터리 잔량 확인	워치의 배터리 잔량을 확인한다.	DB 서버로부터 배터리 데이터를 watch_id 로 검색하여 가져온다.	해당 워치의 배터리 잔량을 보여준다.	성공
낙상	감지	스마트 워치 사용자의 낙상을 감지한다.	스마트 워치를 착용한 상태로 넘어진다.	낙상 유무 결과가 나온다.	성공