

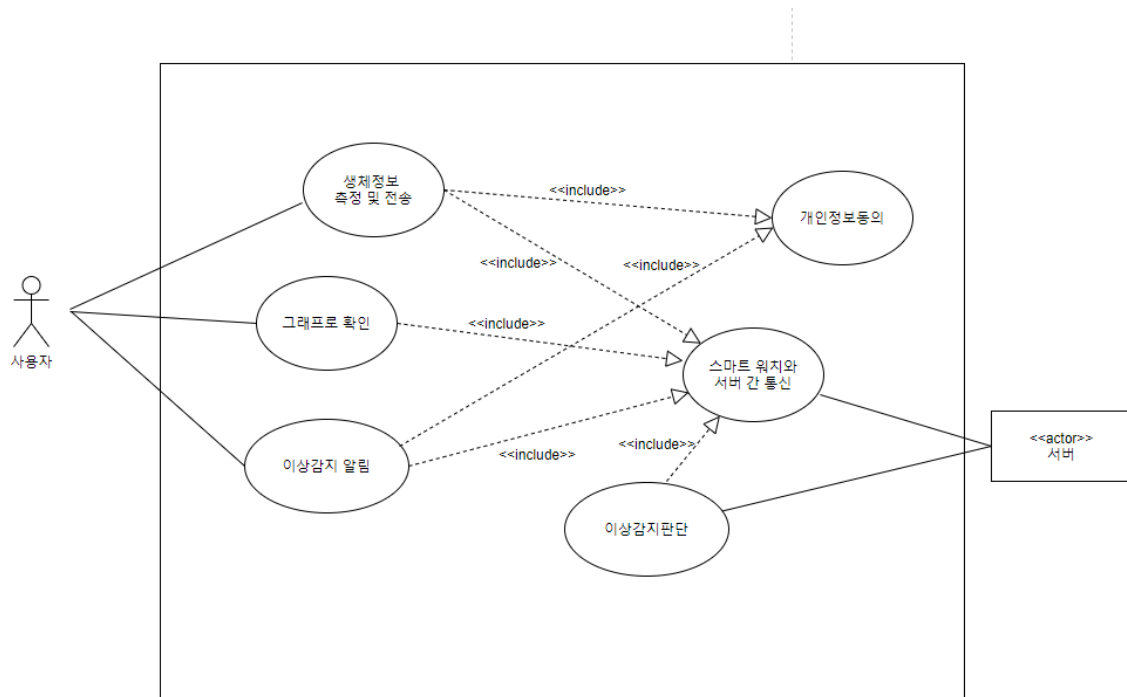
MIT B

(조형래, 연재민, 유성원, 김 건)

1. 개발 시스템/SW 요구사항 - 자연어로 된 개략적인 요구사항

스마트 워치에서 측정된 심장박동수와 걸음수 데이터를 사용하여 사용자의 심박수에 대한 정상/비정상을 판단하여 알림 메시지를 통해 알려주고 수치를 그래프로 나타내어 사용자가 한눈에 확인할 수 있도록 한다.

2. UC 다이어그램 및 UD 기술서



유스케이스명	개인정보동의
액터명	주 액터 : 사용자
개요	사용자는 생체정보 측정과 알림메시지에 대한 동의를 한다.
사전조건	· 사용자는 해당 어플리케이션을 사용하기 위해 설치한다.
사후조건	· 생체정보를 실시간으로 전송할 수 있는 서버환경이 구축되어야 한다. · 알림메시지를 나타낼 수 있어야 한다.
기본흐름	1. 개인정보 및 알림에 대한 동의 의사 화면을 출력한다. 2. 사용자는 개인정보 동의를 한다. 3. 사용자의 생체정보가 실시간으로 측정된다. 4. 상황에 따라 사용자는 알림메시지를 받을 수 있다.
대체흐름	2.a 사용자가 동의를 하지 않는다. 2.a.1 사용자는 해당 어플리케이션을 강제로 종료한다.

유스케이스명	생체정보 측정 및 전송
액터명	주 액터 : 사용자 / 부 액터 : 서버
개요	스마트 워치를 이용해 사용자의 생체정보를 측정한다.
사전조건	<ul style="list-style-type: none"> · 사용자의 생체정보 측정에 대한 동의가 필요하다. · 사용자가 스마트 워치를 바르게 착용해야한다. · 스마트 워치의 전원이 꺼지면 안된다.
사후조건	<ul style="list-style-type: none"> · 측정된 생체정보가 전달 될 수 있는 통신환경이어야한다.
기본흐름	<ol style="list-style-type: none"> 1. 「개인정보동의」유스케이스를 실행한다. 2. 사용자의 생체정보를 측정하고 서버로 전송한다.
대체흐름	<ol style="list-style-type: none"> 2.a. 생체정보가 일정시간 측정되지 않는다. 2.a.1. 올바르게 착용해달라는 경고메시지를 출력한다. 2.a.2. 일정시간이상 측정되지 않으면 수면상태나 착용하지 않는경우로 판단한다.

유스케이스명	스마트워치와 서버간 통신
액터명	주 액터 : 사용자 / 부 액터 : 서버
개요	스마트 워치에서 측정한 신체정보를 서버에 받고 이상감지판단 결과를 스마트 워치에 전송한다.
사전조건	<ul style="list-style-type: none"> · 애플리케이션이 실행되어 있어야 한다. · 이상감지 판단결과가 존재해야한다. · 어플리케이션이 정상적으로 전송 될 수 있는 통신환경을 갖춰야한다.
사후조건	<ul style="list-style-type: none"> · 통신환경(와이파이 또는 블루투스)에 따라 자동으로 선택해 전송함 · 서버에서 스마트 워치의 생체정보를 받을 수 있게 허용함
기본흐름	<ol style="list-style-type: none"> 1. 스마트워치와 데이터 전송을 위해 통신을 한다. 2. 「생체정보 측정 및 전송」유스케이스를 실행한다. 3. 「이상감지판단」유스케이스를 실행한다. 4. 「이상감지알림」유스케이스를 실행한다. 5. 사용자는 상황에따라「그래프로 확인」유스케이스를 실행할 수 있다.
대체흐름 1	<ol style="list-style-type: none"> 1.a 사용자의 환경이 네트워크가 없는 경우 1.a.1. 네트워크가 없다는 알림 메시지 출력 1.a.2. 네트워크와 연결될 때까지 생체정보 측정하지 않음
대체흐름 2	<ol style="list-style-type: none"> 5.a 이상감지 판단 결과를 스마트 워치가 받지 못한 경우 5.a.1. 그 결과 값은 사라지고 최신 탐지값을 전송한다.

유스케이스명	이상감지판단
액터명	주 액터 : 사용자 / 부 액터 : 서버
개요	실시간으로 측정된 생체정보를 이용해 서버에서 이상감지판단을 한다.
사전조건	<ul style="list-style-type: none"> · 생체정보가 측정되어야하고 서버를 통해 받아져야한다. · 이상판단을 할 수 있는 알고리즘이 개발되어있어야 한다.
사후조건	· 판단결과를 사용자가 알 수 있도록 서버를 통해 전달할 수 있어야 한다.
기본흐름	<ol style="list-style-type: none"> 1. 「생체정보 측정 및 전송」유스케이스를 실행한다. 2. 서버는 사용자의 생체정보를 이용해 이상감지판단을 한다. 3. 이상감지판단 시 비정상되면 「이상감지 알림」유스케이스를 실행한다.
대체흐름	<p>3.a 이상감지판단 시 이상이 아닌 정상으로 판별된다.</p> <p>3.a.1 정상인 경우에는 아무 알림도 사용자에게 전달하지 않고 계속해서 측정을 한다.</p>

유스케이스명	그래프 확인
액터명	주 액터 : 사용자 / 부 액터 : 서버
개요	사용자가 자신의 현재와 과거의 생체정보를 그래프로 확인 할 수 있다.
사전조건	<ul style="list-style-type: none"> · 실시간으로 생체정보가 측정이 되어야 한다. · 어플리케이션이 생체데이터를 시각화 해야한다.
사후조건	· 사용자가 그래프화된 생체정보를 볼 수 있다.
기본흐름	<ol style="list-style-type: none"> 1. 「생체정보 측정 및 전송」유스케이스를 실행한다. 2. 사용자가 그래프 확인을 누른다. 3. 서버는 일정주기의 그래프를 만들어서 스마트워치로 전송한다. 4. 사용자가 그래프를 한 눈에 확인 할 수 있다.
대체흐름	<p>3.a 스마트워치에서 그래프를 받지못한다.</p> <p>3.a.1 다시 서버에 요청한다. 계속해서 안받아지는경우 네트워크 상태를 확인해달라는 메시지를 출력한다.</p>

유스케이스명	이상감지 알림
액터 명	주 액터 : 사용자 / 부 액터 : 서버
개요	사용자가 자신의 이상상황을 서버를 통해 알림을 받을 수 있다.
사전조건	<ul style="list-style-type: none"> · 알림메시지를 받는다는 사용자의 동의가 필요하다. · 사용자의 생체정보가 실시간으로 서버로 전송되어야한다. · 서버도 스마트 워치 간의 통신이 원활해야한다.
사후조건	· 사용자가 알림메시지를 확인하고 상태를 알 수 있다.

기본흐름	1. 「개인정보동의」유스케이스를 실행한다. 2. 「생체정보 측정 및 전송」유스케이스를 실행한다. 3. 「이상감지판단」유스케이스를 실행한다. 4. 이상감지 판단 시 사용자에게 알림메시지를 출력한다. 5. 사용자는 메시지를 확인한다.
대체흐름 1	1a. 사용자가 동의하지 않는다. 1a.1. 사용자는 알림메시지에 대한 기능을 사용할 수 없다는 사실을 알리고 다시 되물어본다. 1a.2. 다시 되물어도 동의를 하지않는경우 해당기능을 사용할 수 없다.
대체흐름 2	5a. 사용자가 5분동안 메시지를 확인하지않았다. 5a.1. 메시지를 다시 보낸다.

3. 개인 업무/연구/개발 범위 정의, 개인 별 학습 및 개발 일정

- 개인 업무 및 개발 범위

조형래 : 팀 리더 / 머신러닝 / 팀원학습지도 / 테스트

연재민 : 팀 리더와 함께 문서, 과제 분석 / 데이터 관리 및 시각화 / 어플 - 그래프

유성원 : 메시지 브로커 구축 / 머신러닝 / 어플 - 서버와의 통신

김 건 : 팀의 회의 내용 기록 및 관리 / 어플 - 디자인 , 알림 메시지 및 상태 표현

- 공통 업무 : 어플 구현

- 학습 일정

공통 학습 (4월 ~ 5월중순)

- 개발환경 구축 및 기본개념
- 각종 클래스의 개념 및 문법
- 다양한 컬렉션
- android 개발 튜토리얼

연재민

> postgresql (03.20~03.26)

- 개발환경 구축 및 기본개념
- SELECT, INSERT, UPDATE, DELETE
- Order by, Distinct, Where, Limit
- Offset & Fetch, IN, Between
- Group By, NULLIF, Upset
- Inner Joins, Left Joins

> timescaleDB (03.27~04.04)

- 개발환경 구축 및 기본개념
- Hypertables, Distributed Hypertables
- Schema management, Writing data / Reading data
- Actions, Data Retention
- timescaleDB에서 제공하는 예시 및 실습

유성원

> KAFKA (03.20 ~ 03.31)

- 기본 개념 및 생태계 숙지 , AWS에 설치 및 실행
- 테스트 프로듀서 앱을 만들어 실행 및 확인
- 테스트 컨슈머 프로듀서 구현
- 파이프라인을 통해 테스트 컨슈머 프로듀서 앱을 사용해 csv파일 저장 및 확인

> 머신 러닝 (04.01 ~ 04.30)

- 유클리드 거리와 마할라노비스
- 가우시안 분포와 분별함수
- 공분산행렬
- LDA(Linear Discriminant Analysis)
- scikit-learn을 통해 마할라노비스를 이용한 이상탐지 알고리즘 구현

● 개발 일정

> 1차 개발 : 서버 (things board) (04.23~25)

> 2차 개발 : 이상감지 (Machine Learning) (04.25~04.30)

> 3차 개발 : 실사용을 위한 어플 구현 (kotlin) (05.15~05.30)