

**Fit Got U (홈 트레이닝 플랫폼)**

## 목차

1. 초록 .....	5
2. 개요 .....	5
2.1. 문제정의 .....	5
2.2. Stakeholders 설명 및 관계도 .....	6
2.3. 가치 .....	7
2.4. 핵심기술요소 .....	8
2.5. 목표 .....	8
3. 배경 .....	9
3.1. 유사제품 혹은 서비스 .....	9
3.2. 기술적 배경 .....	10
3.3. 제한요소분석 .....	10
3.3.1. 기술적 제약 .....	10
3.3.2. 데이터 수집 .....	10
3.3.3. 사회적 영향 .....	11
4. 시스템 구성 및 기능 .....	12
4.1. 활용 기술 스택 .....	12
4.3. 시스템 설명 .....	15
4.4. 최종 구현 환경 .....	16
5. 구현 결과 .....	17
5.1. 기능 설명 .....	17
5.1.1. 회원가입과 로그인 .....	17
5.1.2. 자세 인식 및 피드백 .....	17
5.1.3. 데이터분석 .....	17
5.1.4. 주변 트레이너 확인 .....	17

5.1.5. 트레이너 유튜브 확인 .....	17
5.1.6. 식단/루틴 신청.....	17
5.1.7. 식단/루틴 스케줄 확인 .....	18
5.1.8. 식단/루틴 제공.....	18
5.1.9. 유튜브 연동 .....	18
5.2. 동작 과정 설명 .....	18
5.2.1. 회원가입과 로그인 .....	18
5.2.2. 자세 인식 및 피드백.....	18
5.2.3. 데이터분석.....	19
5.2.4. 주변 트레이너 확인 .....	19
5.2.5. 트레이너 유튜브 확인 .....	19
5.2.6. 식단/루틴 신청.....	19
5.2.7. 식단/루틴 스케줄 확인 .....	19
5.2.8. 식단/루틴 제공.....	20
5.2.9. 유튜브 연동 .....	20
5.3. 테스트 시나리오 .....	21
5.3.1. 사용자(Trainee).....	21
5.3.2. 트레이너(Trainer).....	24
5.4. 테스트 결과 .....	27
5.5. 매뉴얼 .....	28
6. 과제 관리 .....	28
6.1. BD 차트 기반 진행 과정 설명 .....	28
6.1.1. product backlog .....	28
6.1.2. 백로그에 기반한 Iteration 계획 .....	30
6.2. 협업 방식/도구 설명 .....	35
6.2.1. Git.....	35

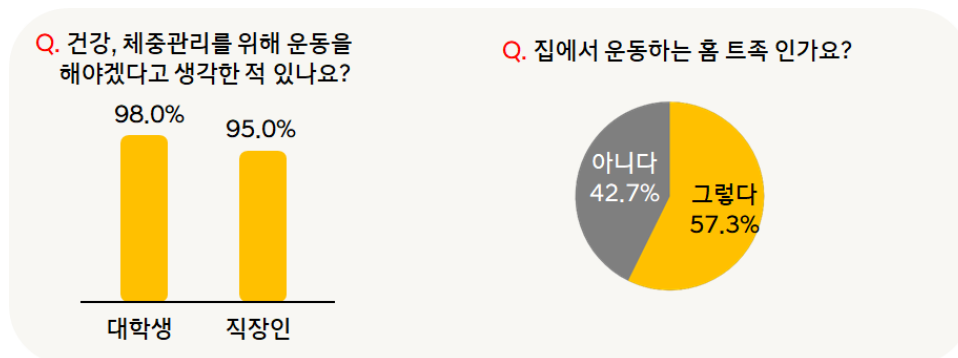
6.2.2. Slack .....	35
7. 성과 .....	35
7.1. 특허, 소프트웨어 등록 .....	35
7.2. 학술논문발표 등, 정량화 될 수 있는 기타 성과 .....	35
8. 결론 .....	36
8.1. 프로젝트 진행(실행) 과정에서 잘 된(한) 점 그 이유 .....	36
8.2. 프로젝트 진행 과정에서 아쉬운 점, 개선 아이디어 .....	36
8.3. 본 과제 결과물의 발전 방향 .....	37
8.3.1. 본 과제의 관련 분야에 대한 기여도 .....	37
8.3.2. 기술적 관점에서의 발전 방향 .....	38
8.3.3. 사업적 관점에서의 발전 방향: 향후 상용화 가능성 .....	38
9. 참고문헌 .....	38
10. 해당 프로젝트 github 링크 .....	39

## 1. 초록

최근 홈 트레이닝에 대한 수요가 증가하고 있다. 홈 트레이닝이란 운동은 해야 하지만, 시간이 여의치 않은 사람들이 각종 동영상(SNS, Youtube 등)이나 서적 등을 보면서 스스로 트레이닝을 하는 것이다. 홈 트레이닝은 시간뿐만 아니라 비용도 절약할 수 있다는 점에서 2030들에게 큰 인기를 끌고 있다. 이에 따라, 수 많은 홈 트레이닝 관련 어플리케이션이 존재한다. 하지만, 가장 큰 문제점은 운동에 대한 설명과 방법이 충분한 것에 비해, 운동을 하는 당사자가 운동이 정확한지 알기 힘들고 운동에 대한 피드백은 적다는 것이다. 또한, 일반적으로 운동을 하는 사람들은 잘못된 운동 자세로 신체에 무리를 줄 수 있다. 우리는 이 문제점을 Pose Estimation을 통해 사람의 움직임을 실시간으로 보여주는 것과 음성 피드백으로 해결하고자 했다. 이를 통해, 많은 사람들에게 좀 더 정확하고 편한 홈 트레이닝과 많은 금전적 비용을 들이지 않아도 누구나 언제 어디서든 운동을 할 수 있는 충분한 기회를 제공하고자 한다.

## 2. 개요

### 2.1. 문제정의



[그림 1] 홈 트레이닝 설문 조사<sup>1</sup>

현대의 사람들은 불규칙한 식습관, 부정확한 자세 및 신체 활동의 부족 등으로 척추측만증, 허리 디스크 등의 척추 질환 및 비만, 고혈압 등의 성인병과 같은 문제들에 고통받고 있다. 또한 먹고 살기 바쁜 과거의 생활수준과 달리 '웰빙 시대'로 접어들며 본인의 건강 및 운동에 대한 관심이 높아지고 있다.

---

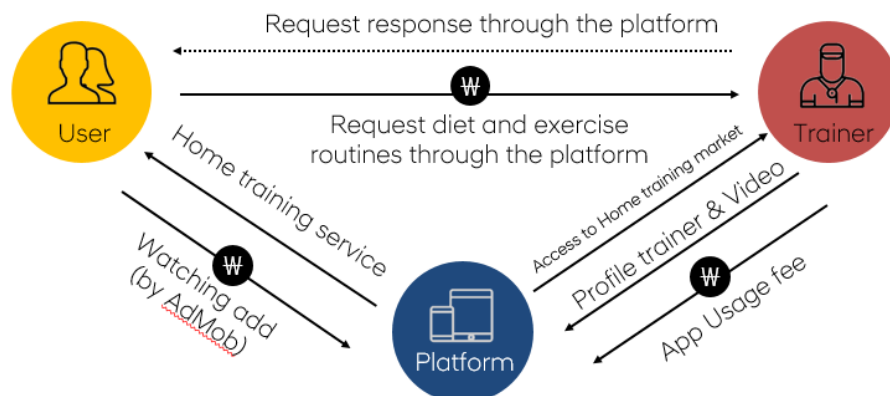
<sup>1</sup> 잡코리아 x 알바몬

[그림 1]은 성인남녀 831명을 대상으로 조사한 결과이다. 대학생의 98.0%, 직장인의 95.0%가 운동을 하겠다고 생각한 적이 있었다. 하지만 전체 인원 중 절반이 넘는 인원인 57.3%의 응답자가 집에서 운동을 한다고 응답할 만큼, 점점 더 많은 사람들이 개인적으로 운동을 하려는 추세이다.

이전의 트레이닝은 헬스장 또는 학원에 직접 찾아가 강사에게 지도를 받는 형식으로 바뀐 현대인들의 운동에 대한 작심(作心)을 꺾기에 충분하였다. 그에 따라 유튜브(Youtube) 등 동영상 플랫폼을 통한 운동 강의를 따라하는 방식의 홈 트레이닝이 유행하였으나, 이 트레이닝의 가장 큰 문제는 운동하는 당사자가 본인의 자세를 정확하게 파악하지 못한다는 것이었다. 대개 지도를 받기를 원하는 사람들은 운동 숙련자가 아니기 때문에 단순히 영상을 따라하는 것만으로는 정확한 부위에 제대로 단련하고 있는지를 판단하기 어려울 뿐만 아니라 운동 효과를 감소시키거나 더 나아가서 신체에 무리를 줄 수 있다. 따라서 단순히 운동에 대한 정보를 제공하는 것에서 그치지 않고 사용자에게 현재 자세에 대한 피드백을 줄 수 있다면 이러한 문제점을 극복할 수 있다.

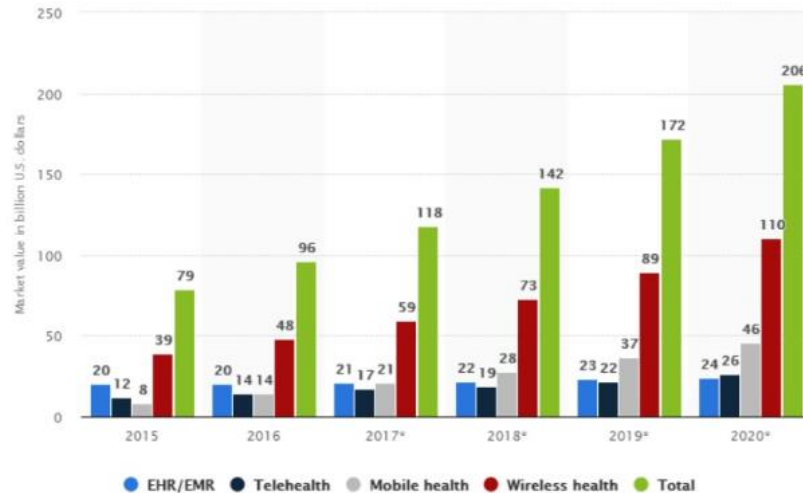
## 2.2. Stakeholders 설명 및 관계도

Stakeholders	Description
사용자(주 타겟)	집에서 운동을 하면서 자신의 운동 자세가 올바른 지를 확인하고 싶어하는 홈트족이다.
트레이너	요가나 헬스 등 각 분야의 전문가이다. 자신을 홍보할 수 있는 기회와 더불어 여러 트레이닝 활동을 통해 수입을 창출하기를 희망한다.



[그림 2] Stakeholder 관계도

## 2.3. 가치



[표 2] 글로벌 디지털 헬스 시장 규모<sup>2</sup>

[표 2]에서 볼 수 있는 것과 같이 글로벌 디지털 헬스 시장의 규모는 점점 증가하고 있다. 스마트폰, 통신 기술이 발전하고 5G와 같은 차세대 통신 기술이 상용화되면서 지금까지의 피트니스 시장과는 다른 새로운 블루오션이 펼쳐지게 될 것이다.

따라서 기존의 피트니스 어플리케이션과 동일하게 운동 자세에 대한 정보, 일정 관리와 같은 기본적인 피트니스 기능을 제공하면서 카메라를 통해 자신의 자세를 분석하여 피드백을 받을 수 있다면 사용자들의 흥미를 유발할 수 있을 것이다. 사용자는 자신의 운동 습관, 취약점과 같은 정보를 얻음과 동시에 운동에 대해 흥미를 느낄 수 있다. 여기에 그치지 않고 보다 전문적인 강습 또는 관리를 원하는 회원들이 실제 트레이너와 연락하며 상호 작용할 수 있다면, 사용자에게는 보다 나은 만족감을 안겨줄 것이고 트레이너에게는 기존의 소극적인 광고에서 벗어나 보다 공격적으로 회원들에게 자신을 홍보할 수 있는 새로운 홍보 수단으로써 역할을 제공할 수 있을 것으로 기대된다.

<sup>2</sup> Statista, <http://it.donga.com/27408/>

## 2.4. 핵심기술요소

기존의 피트니스 어플리케이션에서 한 걸음 더 나아가서 카메라를 통해 자신의 자세를 분석하여 올바른 자세인지에 대한 피드백을 제공하기 위해서는 카메라를 통해 촬영한 화면을 분석하여 사용자의 자세를 데이터화 할 수 있어야 한다. 이를 구현할 수 있는 방법에는 크게 두 가지 방법이 존재하는데 첫 번째는 중앙처리서버를 통해 영상, 이미지를 처리하고 그 결과값을 사용자에게 전송하는 것이다. 그러나 이러한 방식은 사용자가 많아질 수록 서버가 처리해야 하는 연산량이 많아지고 실시간 피드백이 중요한 어플리케이션의 특성상 네트워크 상태에 따라서 정확도가 떨어질 수 있다는 문제점을 가지고 있다.

두 번째 방법은 스마트폰 내에서 직접 이미지를 처리하는 것으로 점점 발전하는 스마트폰 성능 덕분에 가능해지게 되었다. 이러한 방식은 어플리케이션 자체에서 분석을 진행함으로써 이미지 처리를 위한 서버를 별도로 구축할 필요가 없는 이점이 있어서 개발 효율에 큰 이점을 가지게 된다. 또한 올바른 촬영 환경에서는 네트워크 없이도 운동에 대한 피드백을 줄 수 있다. 따라서 이번 프로젝트에서는 스마트폰 기기 내에서 촬영된 이미지를 처리하고 이를 바탕으로 피드백을 줄 수 있도록 하는 방향으로 개발하였다.

## 2.5. 목표

2.1. 문제정의에서 서술한 문제점을 해결하기 위해 사용자의 운동 자세를 인식하여 올바른 운동 자세와 현재 사용자의 운동 자세를 비교/분석하고 스피커를 통해 즉각적으로 사용자에게 피드백을 한다. 사용자는 지시에 따라 자세를 교정하고, 자세의 정확도를 향상시킨다. 이로써 사용자가 올바른 자세로 효율적인 홈 트레이닝을 할 수 있도록 할 뿐 아니라, 더 나아가 사용자로 하여금 운동에 흥미를 불러일으키고 목표한 바를 완수할 수 있도록 도움을 주는 것을 목적으로 한다.

이외에도 본 앱은 스스로 운동에 갈피를 잡지 못하는 사용자를 위해 전문 트레이너들이 업로드한 동영상 시청할 수 있도록 하였으며, 개인 트레이닝에 한계를 느끼는 사용자에게는 트레이너와 상담하고 운동과 식단을 추천/상담 받을 수 있도록 도와주는 기능을 제공한다. 실제로 트레이너와 상호 작용하고 싶은 사용자를 위해 등록된 트레이너를 자신이 사는 거주지에서 가까운 순서대로 표시해준다.

또한 사용자의 운동 정보에 대한 데이터를 분석하여 취약한 신체 부위를 파악하고 이를 바탕으로 운동을 추천해 줌으로써 장기적으로 보다 효율적인 운동 효과를 이끌어낼 수 있을 것이다. 이 데이터는 연락한 트레이너와 공유할 수 있으므로 보다 효과적인 운동 루틴 추천을 가능하게 한다.



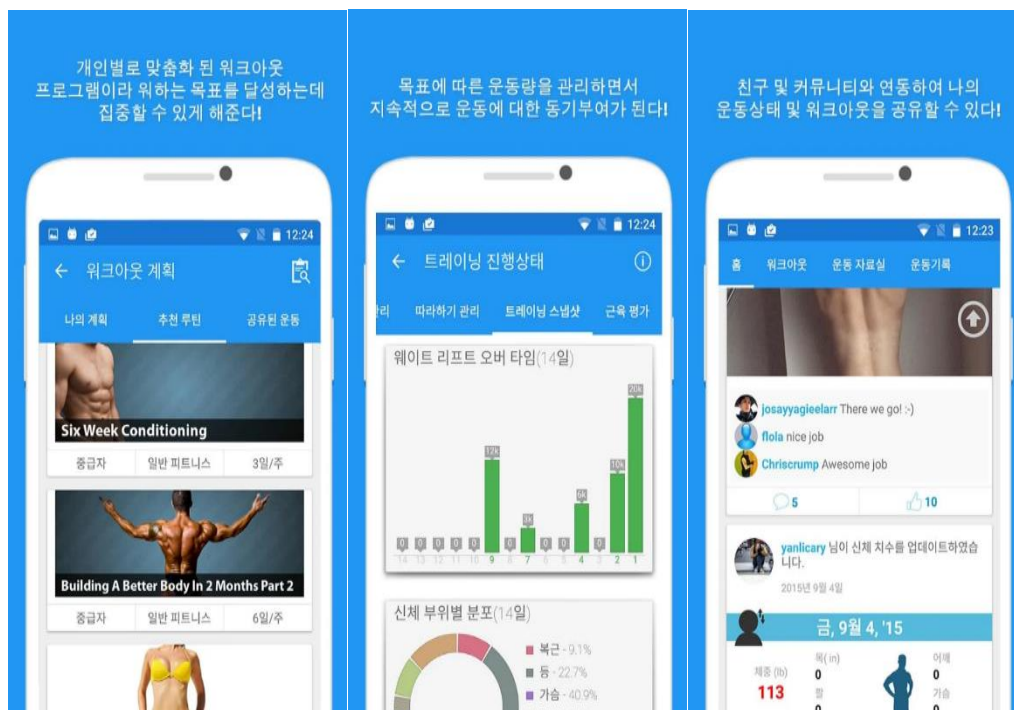
앞서 언급했듯이, 사용자는 본 앱을 통해 홈 트레이닝에서도 정확한 자세를 유지하며 운동할 수 있을 것이다. 또한 전문 트레이너와의 소통을 통해 사용자는 집에서도 헬스장에서 퍼스널 트레이닝을 받는 것과 같은 식단 추천이나 운동 루틴을 받을 수 있다. 본 앱과 사용자의 의지를 고취시키며 올바르게 효율적인 홈 트레이닝을 수행할 수 있을 것이다.

### 3. 배경

#### 3.1. 유사제품 혹은 서비스



[그림 3] Keep, JEFIT, 프로 피트니스 어플리케이션<sup>3</sup>



[그림 4] JEFIT의 설명 화면<sup>4</sup>

<sup>3</sup> Google Playstore

<sup>4</sup> Google Playstore, <https://play.google.com/store/apps/details?id=je.fit>

[그림 2]과 같이 구글 플레이스토어를 기준으로 등록된 운동 관련 어플리케이션들을 분석한 결과 많은 어플리케이션들이 [그림 3]와 같이 운동 자세에 대한 정보와 운동 일지 기록, 그리고 운동량 계산과 같은 단순한 역할을 하는데 치중되어 있다는 것을 확인할 수 있었다. 조사한 어플리케이션에는 사용자의 자세를 확인할 수 있는 기능이 없어 사용자는 자신의 자세가 올바른 자세인지 정확히 확인할 수 없기 때문에 자신이 생각한 것보다 운동의 실제 효과가 떨어질 수 있는 위험이 있다. 이는 장기적으로는 올바르지 못한 자세로 인한 신체 부상의 위험을 가지고 있으며 운동효과의 감소로 인한 운동 동기 감소와 같은 부작용을 초래할 수 있을 것으로 예상된다.

## 3.2. 기술적 배경

Fit-Got-You 어플리케이션에서는 스마트폰 기기 내에서 이미지를 처리하여 Pose Estimation을 사용했다. 이와 관련된 학과목으로는 "기계학습 및 데이터마이닝", "오픈소스SW", "컴퓨터 비전"이 있다. 기계학습 및 데이터마이닝 강의에서는 기본적인 딥러닝의 원리와 그 방법에 대해서 알 수 있고 컴퓨터 비전에서는 이미지나 영상에서 어떻게 원하는 오브젝트만을 추출할 수 있는지에 대해 배울 수 있다. 오픈소스SW에서는 딥러닝, Pose Estimation에 필요한 오픈소스들을 어떻게 다루어야 하는지에 대해서 배울 수 있다. 이는 스마트폰 어플리케이션 기반으로 만들어진 프로그램이기 때문에 "모바일 네트워크" 학과목의 지식과 데이터 서버 구축을 위한 "데이터베이스" 과목의 학습이 선행되었다.

## 3.3. 제한요소분석

### 3.3.1. 기술적 제약

스마트폰 기기 자체에서 Pose Estimation 이미지 처리를 할 수 있을 정도로 스마트폰의 성능은 가파르게 좋아지고 있으나 Pose Estimation을 정확하게 하면서 실시간 피드백을 가능하게 하기 위해서는 인식할 수 있는 사람의 수를 1명으로 제한할 필요가 있었다. 사람의 수가 늘어날 수록, 그리고 많은 사람을 정확히 구분하고 자세를 인식하는 것은 굉장히 많은 연산량을 필요로 하기 때문에 홈트레이닝이 이루어지는 환경을 고려하여 1명의 사람만을 인식할 수 있도록 하였다.

### 3.3.2. 데이터 수집

사용자의 운동자세를 Pose Estimation을 활용해 바로잡아주고 사용자의 운동 기록을 분석해 다음 더 나은 운동을 수행할 수 있게 도와주는 것이 본 프로젝트의 목적이다. 이를 위해서는 운동에 대한 정확한 표준 데이터가 필요하다. 하지만 트레이너마다 또는 각 운동에 관한 교본 및 논문마다 정확한 자세에 대한 정의가 다르다. 심지어 현재까지 논쟁이 되는 부분도 있다. 따라서 운동에 대한 바른 정의에 대한 제약이 있으며, 본 프로젝트를 진행하기 위해 유튜브에 있는 다수의 트레이너의 자세를 Pose Estimation으로 다시 분석하고 최적 값을 적용하였다.

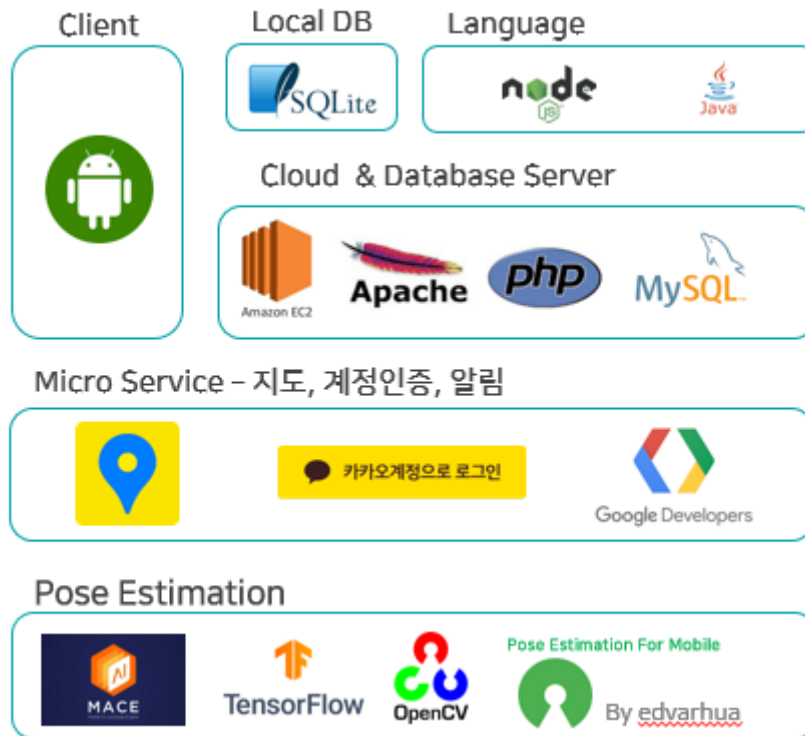
### 3.3.3. 사회적 영향

본 프로젝트를 진행하면서 홈 트레이닝이 활성화되면 헬스장 시장에 악영향을 미치지 않을지 걱정했었다. 운동 자세는 Pose Estimation으로 피드백을 제공하지만, 식단이나 루틴은 트레이너가 제공하는 부분이기 때문에 트레이너 역시 우리의 프로젝트에 호의적이어야 한다.

Nielsen Buzzword의 조사에 따르면 매년 SNS 상에서 홈 트레이닝에 관한 게시글이 증가함과 동시에 헬스장에 관한 게시글도 증가했다. 그리고 실제 홈 트레이닝으로 얻은 운동에 대한 관심이 실제 헬스장으로 연장되는 경우도 상당했다. 따라서 홈 트레이닝의 활성화가 헬스장에게 악영향을 미치는 것이 아닌 초심자에게 운동에 대한 관심의 시작임으로 다가가야 한다. 그리고 트레이너에게 식단이나 루틴 추천 활동을 통한 수입의 새로운 경로를 개척해줌과 동시에 홍보의 수단으로도 활용할 수 있음을 강조하여 홈 트레이닝에 대한 헬스장 시장의 시선을 긍정적으로 바꿔 트레이너와 상생을 도모해야 한다.

## 4. 시스템 구성 및 기능

### 4.1. 활용 기술 스택



[그림 6] 어플리케이션 기술 스택

- 어플리케이션은 [그림 6]와 같은 구조를 가지고 있다. 안드로이드 운영체제에서 동작하기 때문에 개발 환경은 안드로이드 스튜디오와 자바 프로그래밍 언어를 사용한다.
- 로그인, 위치정보와 관련된 기능은 카카오 API 중에서도 지도와 로그인 API를 통해 구현된다.
- 내부 데이터베이스로는 SQLite를 사용한다. SQLite는 사용자의 운동 데이터를 기록하고 이를 분석하는 데에 사용된다.
- 동작 인식 기능은 오픈 소스인 PoseEstimationForMobile<sup>5</sup> 을 사용한다.
- PEFM은 OpenCV와 TensorFlow, 그리고 Mace를 활용하여 안드로이드 스마트폰의 GPU를 계산에 사용하며 사람의 동작을 인식한다.

<sup>5</sup> <https://github.com/edvardHua/PoseEstimationForMobile>

- OpenCV와 TensorFlow로 학습시킨 동작인식 모델을 mace를 통해 스마트폰의 GPU를 활용하여 계산할 수 있도록 최적화해준다.
- 초기엔 PEFM과 더불어 PoseNet<sup>6</sup>, OpenPose<sup>7</sup>와 같은 오픈소스를 활용하여 서버에서 이미지를 처리하고 결과값을 어플리케이션에 전달하는 방식을 후보로 고려하기도 하였다. 하지만 Openpose는 한 장의 이미지를 처리하는 것보다 연속된 비디오 파일을 분석하는데 최적화되어 있기 때문에 부적합하다고 판단했으며 PoseNet은 시간 여건상 불가능할 것이라고 판단하게 되었다.
- 서버 및 데이터베이스는 AWS 서버에 APM(Apache + PHP + MYSQL)을 설치하고 이 웹 서버를 통해 어플리케이션과 데이터를 주고받는다. 알림 기능의 경우, Node.js 서버를 통해 구글 클라우드 플랫폼에서 제공하는 FCM(Firebase Cloud Message) 서비스를 사용한다.

---

<sup>6</sup> <https://github.com/tensorflow/tfjs-models/tree/master/posenet>

<sup>7</sup> <https://github.com/CMU-Perceptual-Computing-Lab/openpose>

## 4.2. 핵심 알고리즘

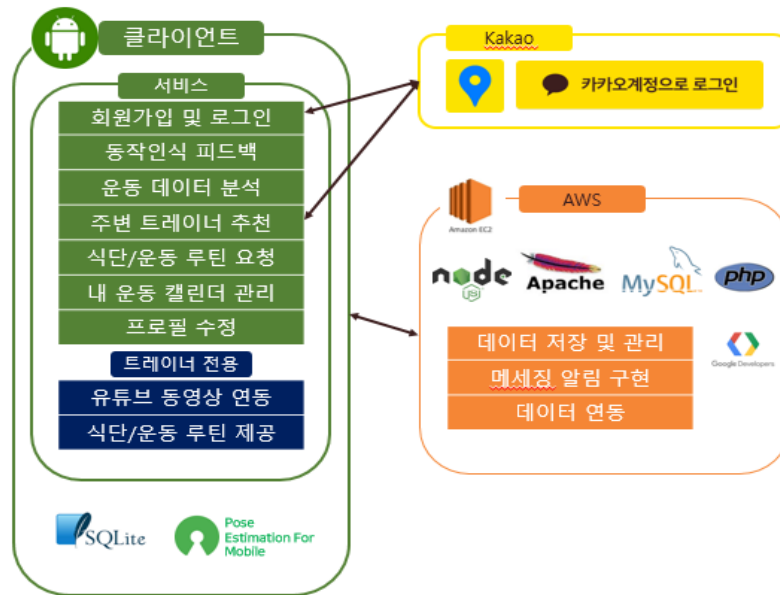
PoseEstimation의 핵심 알고리즘은 크게 2가지로 나눌 수 있다. 먼저 이미지를 처리하는 부분과 이미지를 바탕으로 운동을 판단하는 부분이다. 첫 번째로, 이미지를 처리하는 부분은 카메라를 통해 촬영되는 이미지를 먼저 학습시킨 데이터의 사이즈와 일치시켜야 한다. 이제 해당 이미지를 통해 PoseEstimation을 해야 한다.

```
PoseEstimation(Activity activity) throws IOException {  
  
    int lengthValues = getImageSizeY() * getImageSizeX() * 3;  
  
    float[] floatValues = new float[lengthValues];  
  
    floatBuffer = FloatBuffer.wrap(floatValues, 0, lengthValues);  
  
    kernelPath = Environment.getExternalStorageDirectory().getAbsolutePath() + File.separator  
        + "mace";  
  
    File = new File(kernelPath);  
  
    if (!file.exists()) {  
        file.mkdir();  
    }  
  
  
  
  
    int result = JniMaceUtils.maceMobilenetSetAttrs(  
        2, 0, 3, 3,  
        kernelPath);  
  
    Log.i("APPModel", "maceMobilenetSetAttrs result = " + result);  
  
    JniMaceUtils.maceMobilenetCreateEngine("cpm_v1", "GPU");  
  
}
```

위 부분은 PoseEstimation의 핵심이 되는 부분이다. 학습된 결과 값을 Mace를 통해 현재 촬영되는 이미지를 계산하는 것이다. 여기서 이미지 계산이란 학습 시에 지정해둔 점(이하 포인트) 즉, 머리, 목, 어깨, 팔꿈치, 손목, 엉덩이, 무릎, 발목을 찾는 것이다. 이후 bitmap을 byte로 변환해 이용한다.

두 번째는 운동을 판단하는 부분이다. 첫 번째 결과로 나온 포인트들을 이용해 신체 부분의 각도를 계산하는 방법으로 운동을 판단했다. 예를 들어 스쿼트의 경우, 엉덩이와 무릎을 잇는 직선과 무릎과 발목을 잇는 직선 사이의 각도를 이용했다.

### 4.3. 시스템 설명



[그림 10] 주요 컴포넌트 구조도

본 프로젝트는 사용자에게 안드로이드 모바일 어플리케이션을 제공하며 각 구성에 대한 설명은 아래와 같다.

- 안드로이드 스튜디오에서 제공하는 기능을 그대로 사용하였으며, java와 xml를 통해 화면을 구성한다.
- 일반 사용자는 회원가입, 로그인, 프로필 수정, 식단/운동 루틴 요청, 트레이너 유튜브 열람, 동작 인식 피드백, 데이터 분석 기능을 사용할 수 있다.
- 트레이너는 일반 사용자의 기능과 자신의 유튜브를 연동하는 기능, 식단/운동 루틴 제공 기능을 사용할 수 있다.
- 회원가입 및 로그인 서비스는 Kakao Login API를 이용해서 인증한다.
- 계정 관련 데이터, 식단/루틴 데이터, 유튜브 데이터, 분석된 운동 데이터는 서버에 저장된다.
- 트레이너 추천 서비스는 사용자와 트레이너들의 주소 정보를 Kakao Map API를 이용해 좌표로 바꿔 가까운 거리 순으로 나타낸다.
- 주요 데이터는 AWS 서버의 APM을 통해 교류한다.
- 알림은 Node.js로 FCM 서비스를 이용한다.
- 동작 인식 피드백 서비스는 어플리케이션 내부에 구현된 PoseEstimation 기능을 사용한다.
- 어플리케이션 내에 저장되어야 하는 일반적인 환경설정과 운동 데이터는 내부 DB인 SQLite를 통해 저장된다.

#### 4.4. 최종 구현 환경

- Android Studio 3.4
- Android SDK 28
- Android NDK 16
- Python3
- TensorFlow(1.4)
- MACE(0.9.0, mac os 10.x이상 필요)
- OpenCV 3.4.1
- 삼성 갤럭시 s9+(SM-G965x) - Android Version 8(Oreo)
- NodeJS 8
- JAVA 8
- AWS EC2
- Apache 2.4
- MYSQL 5.5
- PHP 7.2
- Firebase SDK 17
- KAKAO SDK 1.17.0



## **5. 구현 결과**

### **5.1. 기능 설명**

#### **5.1.1. 회원가입과 로그인**

- 사용자는 카카오톡 계정으로 회원가입과 로그인할 수 있다.

#### **5.1.2. 자세 인식 및 피드백**

- 사용자는 자신이 원하는 운동을 선택할 수 있다.
- 사용자는 선택한 운동에 대한 자세에 어떤 문제가 있는지 음성으로 피드백 받는다.
- 사용자는 자신의 운동 횟수를 확인할 수 있다.

#### **5.1.3. 데이터분석**

- 사용자는 시각화된 그림을 통해 분석된 데이터를 한 눈에 확인할 수 있다.
- 사용자는 일주일 내의 자신의 운동 데이터를 확인할 수 있다.

#### **5.1.4. 주변 트레이너 확인**

- 사용자는 자신의 거주지에게 가까운 트레이너들의 체육관 위치를 확인할 수 있다.
- 사용자는 트레이너 목록을 선택하여 트레이너의 상세 정보, 유튜브 목록을 볼 수 있다.

#### **5.1.5. 트레이너 유튜브 확인**

- 사용자는 트레이너들의 유튜브 영상을 확인해 운동과 관련된 정보를 얻을 수 있다.

#### **5.1.6. 식단/루틴 신청**

- 사용자는 원하는 트레이너에게 식단과 루틴을 신청할 수 있다.
- 트레이너의 수락 및 거부 여부를 알림을 통해 사용자에게 전송한다.
- 트레이너가 식단과 루틴을 보내면 사용자는 알림을 받아 이를 확인할 수 있다.

#### 5.1.7. 식단/루틴 스케줄 확인

- 사용자는 달력을 통해 식단 및 루틴을 제공받은 일자를 한 눈에 확인할 수 있다.
- 사용자는 트레이너가 보낸 식단과 루틴을 일자를 선택하여 확인할 수 있다.

#### 5.1.8. 식단/루틴 제공

- 트레이너는 사용자가 식단과 루틴을 신청하면 알람을 받는다.
- 트레이너는 사용자의 운동 데이터를 확인하여 사용자 맞춤형 루틴을 제공할 수 있다.
- 트레이너는 사용자에게 식단과 루틴을 보낼 수 있다.

#### 5.1.9. 유튜브 연동

- 트레이너는 자신의 유튜브 채널 주소를 통해 유튜브를 연동할 수 있다.

### 5.2. 동작 과정 설명

#### 5.2.1. 회원가입과 로그인

- 로그인 화면에서 로그인 버튼을 누르면 카카오톡 계정으로 로그인 버튼이 나타나고 로그인에 성공할 경우 메인 화면으로 넘어간다. 첫 로그인이라면 추가로 몸무게, 키, 주소의 정보를 받는다.

#### 5.2.2. 자세 인식 및 피드백

- 원하는 운동을 선택하고 나타나는 화면의 사람 모양의 테두리에 자신의 몸을 맞춘다. 테두리가 없어지면 그 자리에서 운동을 진행한다.
- 운동을 할 때마다 '잘했어요', '너무 내려갔어요', '너무 올라갔어요' 등의 음성 피드백을 들을 수 있다.
- 운동을 끝내면 전체 운동 횟수와 맞는 자세를 취한 운동 횟수를 보여준다.

### 5.2.3. 데이터분석

- 운동 데이터를 날짜별로 확인할 수 있다. 분석한 결과에 따라 어떤 부분을 보완해야 하는지 알려준다.
- 모든 운동 데이터를 종합한 결과를 시각화한 신체 모형 사진으로 보여준다.

### 5.2.4. 주변 트레이너 확인

- 자신의 주소를 기반으로 주변의 트레이너 PT룸 위치를 맵 상에서 보여준다.
- 리스트는 가까운 순으로 정렬되어 나타난다.

### 5.2.5. 트레이너 유튜브 확인

- 트레이너들의 유튜브를 확인하며 유용한 정보를 얻을 수 있다.

### 5.2.6. 식단/루틴 신청

- 식단/루틴 신청 버튼을 누르면 트레이너에게 알림이 간다. 트레이너가 신청을 수락하면 알림을 받을 수 있다.

### 5.2.7. 식단/루틴 스케줄 확인

- 트레이너가 식단/루틴을 보내면 알림이 온다.
- 알림을 터치하면 어플이 실행되며, 내 정보-내 캘린더 메뉴를 통해 지금까지 받은 식단/루틴을 달력으로 확인할 수 있다.
- 특정 날짜를 터치하면 상세한 정보를 확인할 수 있다.

#### **5.2.8. 식단/루틴 제공**

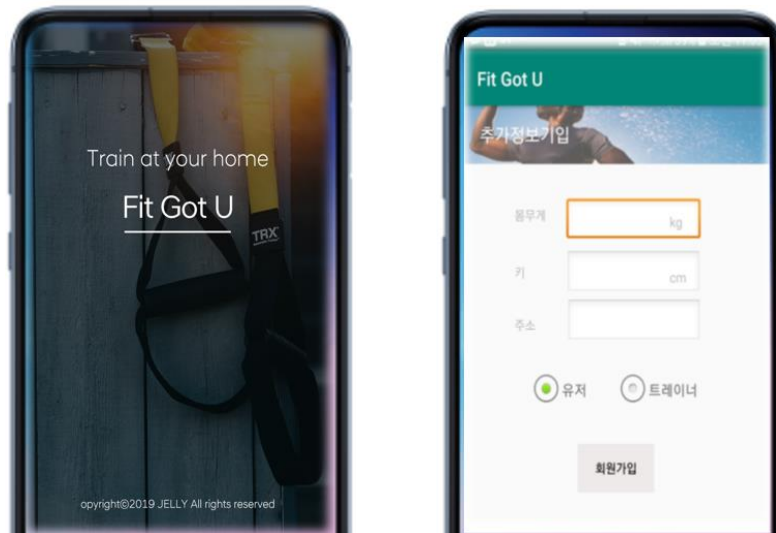
- 유저가 식단/루틴을 신청하면 알림이 온다.
- 알림을 터치하면 어플이 실행되며, 내 정보-유저 관리 메뉴를 통해 자신에게 식단/루틴을 신청한 유저들을 관리할 수 있다.
- 유저의 신청은 수락/거절을 할 수 있으며, 버튼을 누르면 유저에게 알림이 간다.
- 거절할 경우 관리 메뉴에서 해당 유저가 사라지고, 수락할 경우 유저의 분석된 운동 데이터를 확인하거나 식단/루틴을 제공할 수 있다.
- 식단/루틴 제공 버튼을 누르면 새로운 창을 통해 유저에게 식단/루틴을 제공할 수 있다.

#### **5.2.9. 유튜브 연동**

- 트레이너는 자신의 유튜브 채널 주소를 입력하는 것으로 자신의 유튜브 채널을 Fit-Got-U에 연동시킬 수 있다.

### 5.3. 테스트 시나리오

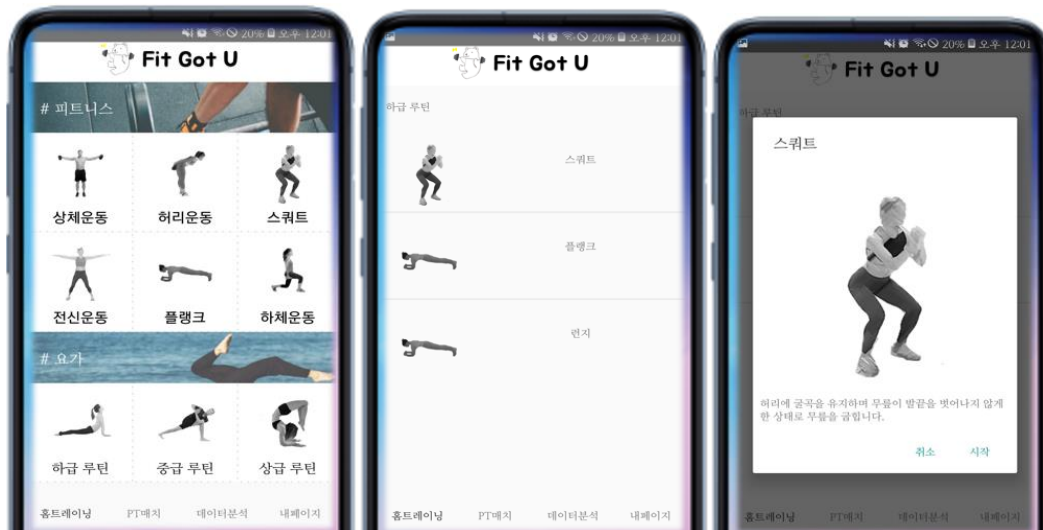
#### 5.3.1. 사용자(Trainee)



a. 카카오 계정 연동을 통한 회원가입을 진행하고 추가 정보를 입력한다.

a.1. 몸무게, 키, 주소 입력

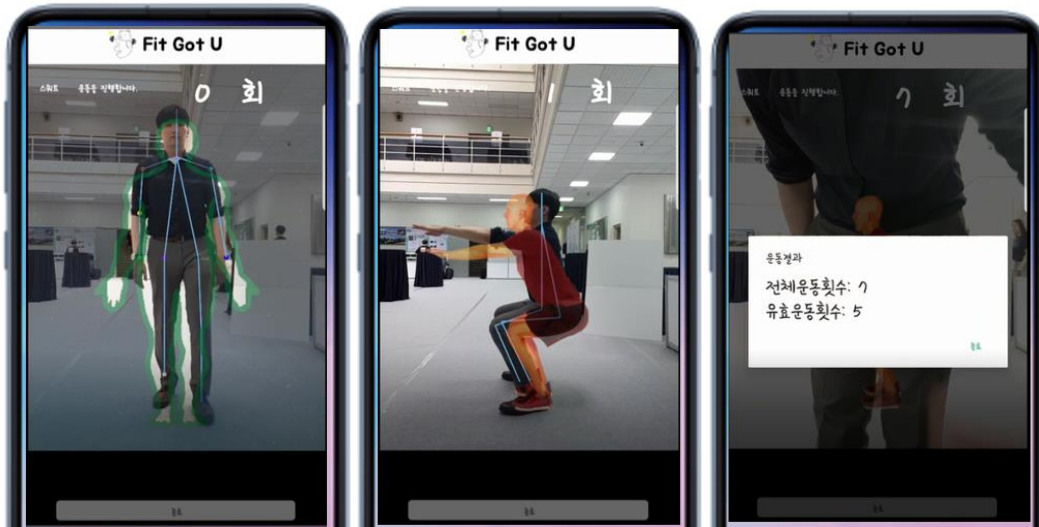
a.2. 유저 또는 트레이너 여부 확인



b. 원하는 운동을 선택한다.

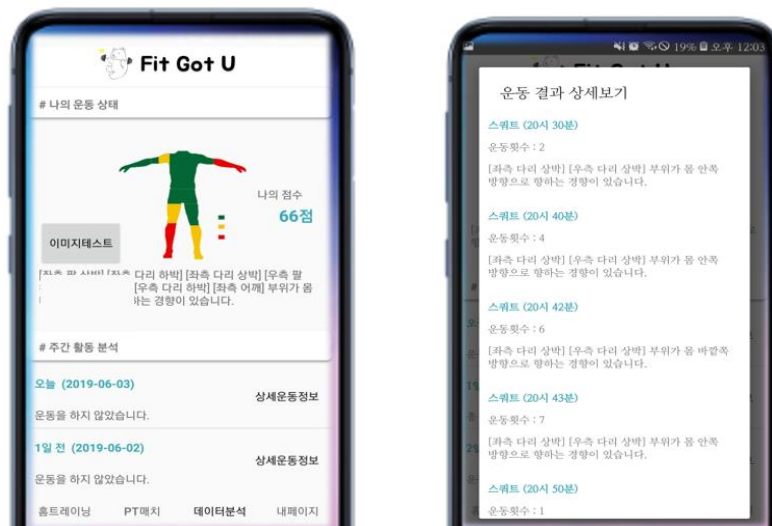
b.1. 하단 메뉴바에서 홈 트레이닝 선택 후 원하는 운동 카테고리 선택

b.2. 원하는 난이도 및 운동 선택



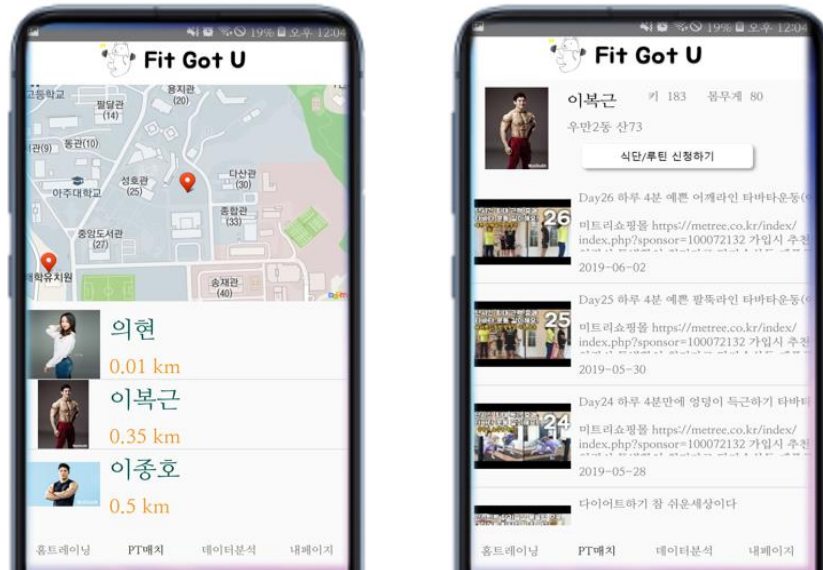
c. 운동을 진행한다.

- c.1. 운동자세에 대한 가이드 확인 후 운동 시작.
- c.2. 자신의 운동에 대한 음성 피드백을 확인하며 운동을 진행
- c.3. 운동 종료 희망 시 종료 클릭 후 운동 기록 확인.



d. 운동 데이터를 확인한다.

- d.1. 하단 메뉴 바에서 데이터 분석 선택
- d.2. 전체 운동 데이터를 시각화한 이미지 확인 및 내 점수 확인
- d.3. 최근 1주일간 운동 기록 확인 및 상세분석기록 확인

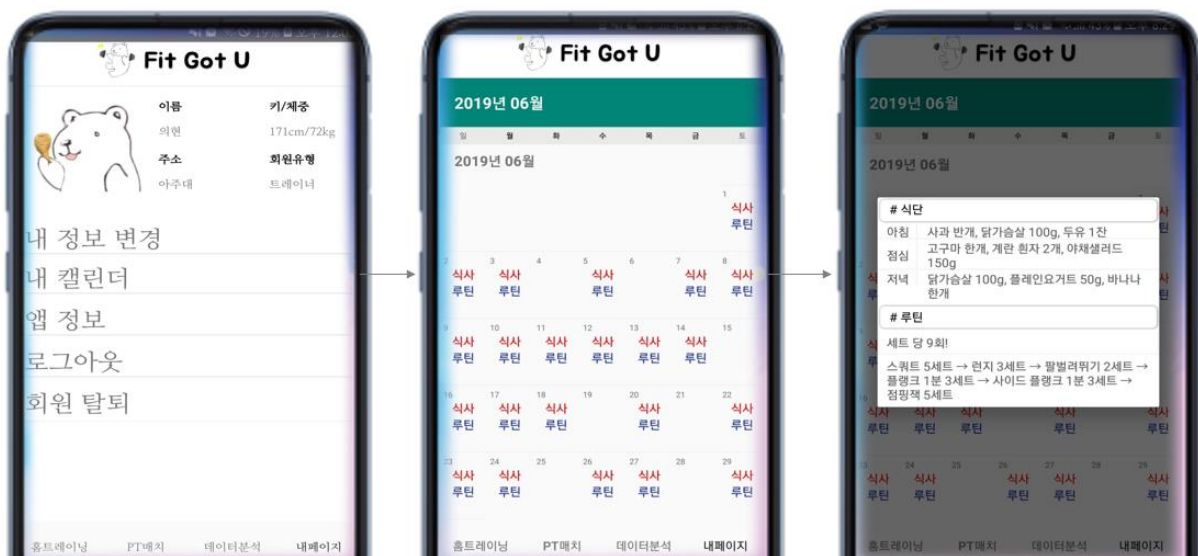


e. PT목록을 확인 후 희망 시 식단/운동 루틴 추천을 요청

e.1. 하단 메뉴 바에서 PT매칭을 클릭

e.2. 내 주변 PT를 지도상으로 확인 후 원하는 PT를 클릭

e.3. PT의 유튜브 영상을 보고 신뢰가 생기면 식단/운동 루틴 신청



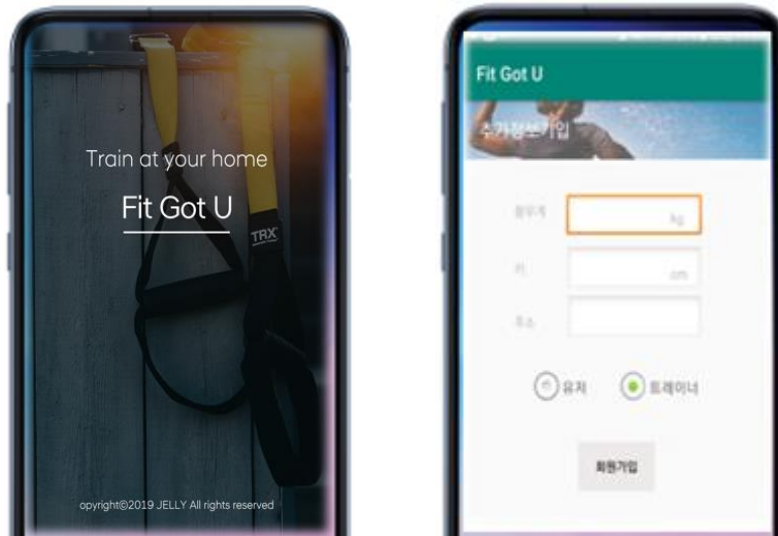
f. 내 정보 관리

f.1. 내 페이지의 기본 정보 확인 및 변경

f.2. 내 캘린더 선택 시 달력 및 날짜 별 식사 및 루틴 유무 확인 가능

f.3. 달력 내 날짜 선택 시 트레이너에게 받은 식단 및 루틴 확인 가능

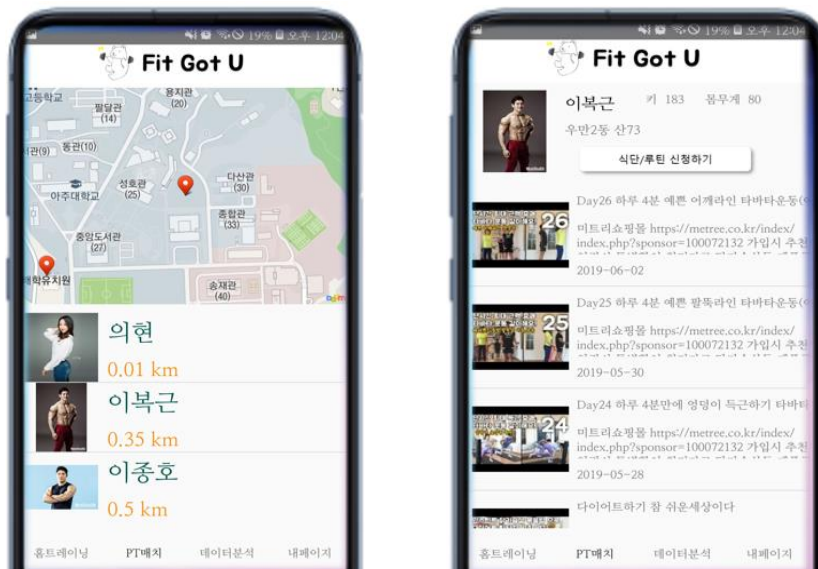
### 5.3.2. 트레이너(Trainer)



a. 카카오 계정 연동을 통한 회원가입을 진행하고 추가 정보를 입력한다.

a.1. 체형에 관한 정보

a.2. 트레이너에 체크



b. 유튜브 영상 관리를 진행한다.

b.1. 하단 메뉴 바에서 내 페이지 클릭

b.2. 유튜브 관리에서 트레이너의 유튜브 홈 주소를 입력



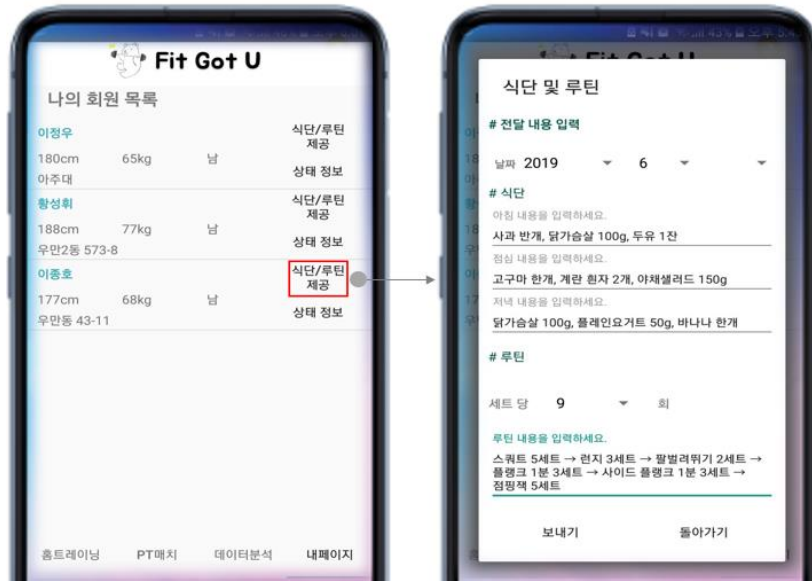


c. 식단/운동 루틴에 대한 요청을 승낙 및 관리를 진행한다.

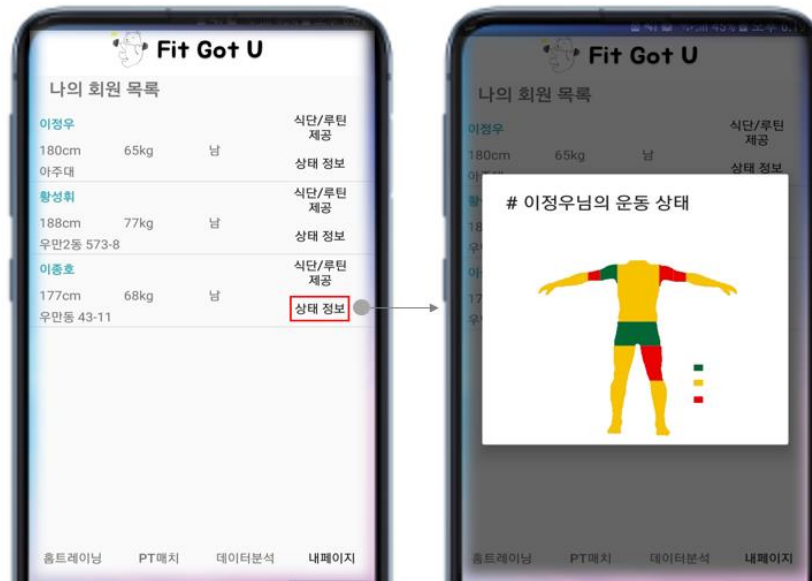
c.1. 하단 메뉴 바에서 내 페이지 클릭

c.2. 내 회원 관리에서 요청 목록을 확인 및 승낙/거부 선택

c.2.1. 승낙 시, 회원의 식단/루틴 제공 및 상태 정보 버튼으로 변경



C.2.1.1. 식단/루틴 제공 버튼 선택 시 식단 및 루틴 제공 가능



c.2.1.2. 상태 정보 버튼 선택 시 회원의 운동데이터 분석 정보 확인 가능

c.2.2. 거절 시, 나의 회원 목록에서 자동 삭제

## 5.4. 테스트 결과

정량적 KPI	성공 여부
로그인, 회원가입, 일정, 영상 등록 등 정보의 추가, 변경, 삭제와 관련된 기능들은 3초 이내에 모든 작업이 완료된다	2초 내에 완료 성공
운동의 피드백을 받는데 있어서 올바르게 앉은 자세를 취할 경우 2초 이내에 음성 피드백을 통해 이를 알려준다	77% 정확도로 1초 이내에 피드백 성공
결제를 진행하는 과정이 사용자의 입력 등의 부분을 제외하고 5초 이내에 이루어진다	프로젝트 일정 변경으로 기능 삭제 실패
데이터 분석에 사용될 시각화한 데이터는 3초 이내에 그려진다	0.5초 이내에 시각화 성공
용자가 식단/운동 루틴을 요청하고 3초 내에 트레이너에게 알려진다	2초 내에 알림 성공
트레이너가 작성한 식단/운동 루틴 데이터가 3초 내에 사용자에게 전달된다	2초 내에 알림 성공

정성적 KPI	성공 여부
로그인, 회원가입, 일정, 영상 등록 등 정보의 추가, 변경, 삭제와 관련된 기능들이 올바르게 작동한다	성공
운동 자세를 측정하는데 있어서 준비 동작을 오인하지 않는다	성공
매일 운동 데이터를 분석하여 사용자가 알기 쉽게 보여준다	성공

## 5.5. 매뉴얼

5.2 동작 과정을 참고.

## 6. 과제 관리

### 6.1. BD 차트 기반 진행 과정 설명

#### 6.1.1. product backlog

Story ID	User Stories	Priority	Task ID	Task	Task Description	Time Estimate (days)	Assigned To
1	프로젝트 환경 구축(공통)	10	1.1	변다운 자트 및 프로젝트 준비	변다운 자트, 깃허브, Slack	4.5	이종호
1	프로젝트 환경 구축(공통)	30	1.2	기술 스택 선택	사용할 기술 스택 체계화	3.0	이종호
1	프로젝트 환경 구축(공통)	10	1.3	안드로이드 스튜디오 설치	안드로이드 스튜디오 설치 및 JDK, API 설치	1.5	하민중
1	프로젝트 환경 구축(공통)	10	1.4	구글 클라우드 플랫폼 확보	구글 클라우드 플랫폼으로 서버 확보	1.0	이정우
2	시나리오 구성(공통)	10	2.1	시나리오를 구성	전체적인 시나리오를 작성	2.0	이종호
2	시나리오 구성(공통)	30	2.2	세부적인 시나리오 구성	시나리오들의 자세한 내용 작성	7.0	정의현
3	동작 인식 방식 선택(공통)	20	3.1	동작 인식 관련 자료 조사	동작 인식과 관련된 소스의 종류 조사	1.0	이정우
3	동작 인식 방식 선택(공통)	20	3.2	동작 인식 소스 분석 및 선택	관련 소스를 분석해 프로젝트와 가장 잘 맞는 동작 인식 소스 선택	5.0	하민중
4	로그인 화면(공통)	10	4.1	로그인 Control Flow	로그인 화면 개요/Control Flow 작성 (회원 트레이너 공통)	1.0	황성휘
4	로그인 화면(공통)	15	4.2	로그인 화면 Design	로그인 화면 UI 설계, 로그인 정보 결정	1.5	황성휘
4	로그인 화면(공통)	20	4.3	로그인 화면 개발	UI 상호작용 구현	2.5	이종호
5	회원가입 화면(공통)	10	5.1	회원가입 체계화의 및 Flow 제작	회원가입 Control Flow / 개요 작성 (회원 트레이너 공통)	1.0	정의현
5	회원가입 화면(공통)	15	5.2	회원가입 화면 Design	회원가입 화면 UI 설계, 회원가입 정보 결정 (회원 트레이너 선택으로 구분)	1.0	정의현
5	회원가입 화면(공통)	20	5.3	회원가입 화면 개발	UI 상호작용 구현	2.0	이종호
6	회원 메인 화면 - 프로필(공통)	10	6.1	사용자 프로필 화면 디자인	사용자 프로필 화면 UI 설계 / Control Flow 작성 (로그인 이후 접근 가능)	1.0	황성휘
6	회원 메인 화면 - 프로필(공통)	20	6.2	사용자 프로필 화면 구현	UI 상호작용 구현	2.0	황성휘
6	회원 메인 화면 - 프로필(공통)	15	6.3	프로필 화면 Test	프로필 확인/수정 기능 Test	1.0	황성휘
7	메인 서버 구축(공통)	20	7.1	필요 시스템 분석	서버 환경 정합 및 API 분석	2.0	정의현
7	메인 서버 구축(공통)	20	7.2	DB 설계	서버에서 저장하고 관리해야 할 데이터 스키마 및 기초 데이터 설계	2.0	정의현
7	메인 서버 구축(공통)	20	7.3	메인 서버 구축	체계화한 정보를 토대로 실 서버 구현	2.0	정의현
8	카카오로그인 인증(공통)	10	8.1	카카오로그인 API 분석	회원가입과 로그인 등에 사용될 API 분석	1.0	이종호
8	카카오로그인 인증(공통)	10	8.2	카카오로그인 인증 구현	카카오로그인 API를 이용한 계정 인증 구현	1.0	이종호
8	카카오로그인 인증(공통)	10	8.3	회원가입 및 로그인 Test	구현한 기능 Test	1.0	이종호
9	회원 메인 화면 - 운동선택(회원)	10	9.1	운동선택 Flow 설계	운동선택 Control Flow / 개요 작성	1.0	정의현
9	회원 메인 화면 - 운동선택(회원)	15	9.2	운동선택 UI 작성	운동선택 UI/상호작용 설계	2.0	정의현
9	회원 메인 화면 - 운동선택(회원)	20	9.3	운동선택 UI 구현	운동선택 UI/상호작용 구현	2.0	하민중
9	회원 메인 화면 - 운동선택(회원)	15	9.4	동작 인식 피드백과 연결	동작 인식 피드백 기능과 연결되도록 merge	2.0	하민중
10	동작 인식 피드백 (회원)	15	10.1	기본 UI 설계	동작인식 UI / Control Flow 설계	2.0	이정우
10	동작 인식 피드백 (회원)	35	10.2	PoseEstimationForMobile 적용	카메라를 이용해 이미지를 캡처해서 스마트폰 내부에서 동작 인식	2.0	하민중
10	동작 인식 피드백 (회원)	35	10.3	이미지 처리 후 데이터 후처리	이미지 분석 결과를 이용해서 신체의 각도, 위치 등 필요한 정보로 가공	4.0	이정우
10	동작 인식 피드백 (회원)	20	10.4	처리받은 값으로 피드백	후처리한 값을 이용해서 자세 인식, 피드백 제공을 위한 부도 클래스 작성	2.0	하민중

Story ID	User Stories	Priority	Task ID	Task	Task Description	Time Estimate (days)	Assigned To
10	동작 인식 피드백 (회원)	20	10.5	스쿼트에 대한 피드백	스쿼트 운동 자세에 대한 피드백 알고리즘 작성	1.0	이정우 ▼
10	동작 인식 피드백 (회원)	20	10.6	플랭크에 대한 피드백	플랭크 운동 자세에 대한 피드백 알고리즘 작성	1.0	하민종 ▼
10	동작 인식 피드백 (회원)	20	10.7	요가자세1에 대한 피드백	요가자세1에 대한 피드백 알고리즘 작성	1.0	이정우 ▼
10	동작 인식 피드백 (회원)	20	10.8	요가자세2에 대한 피드백	요가자세2에 대한 피드백 알고리즘 작성	1.0	하민종 ▼
10	동작 인식 피드백 (회원)	20	10.9	동작인식 피드백 Test	동작인식 기능 / 사용자 피드백 Test	1.0	이정우 ▼
							▼
11	트레이너 추천 탭 (회원)	10	11.1	트레이너 추천 탭 UI 디자인	사용자 프로필 화면 UI / Control Flow 설계	1.5	이종호 ▼
11	트레이너 추천 탭 (회원)	20	11.2	트레이너 추천 탭 UI 구현	트레이너 추천 탭 UI 구현	2.5	이종호 ▼
11	트레이너 추천 탭 (회원)	10	11.3	트레이너 추천 탭 Test	트레이너 추천 탭 기능 및 UI 확인/수정 기능 Tests	1.5	이종호 ▼
							▼
12	거리에 따른 트레이너 정렬(회원)	10	12.1	카카오맵 API 분석	주소를 좌표로 변환하기 위한 API 분석	2.0	이종호 ▼
12	거리에 따른 트레이너 정렬(회원)	10	12.2	거리좌표에 따른 트레이너 정렬	얻은 값으로 트레이너 정렬	1.0	이종호 ▼
12	거리에 따른 트레이너 정렬(회원)	10	12.3	트레이너 추천 탭에 병합	기능 병합	1.0	이종호 ▼
							▼
13	유튜브 동영상 연동 (트레이너)	10	13.1	유튜브 API 분석	유튜브 동영상 처리 관련 API 분석	2.5	황성희 ▼
13	유튜브 동영상 연동 (트레이너)	10	13.2	유튜브 계정 연동	트레이너 계정과 유튜브 계정 연동	2.5	황성희 ▼
13	유튜브 동영상 연동 (트레이너)	10	13.3	유튜브 관련 UI 구현	트레이너 추천탭과 연동된 유튜브 동영상 UI 구현	1.0	황성희 ▼
							▼
14	맞춤형 식단/운동 루틴 탭 (회원)	10	14.1	맞춤형 식단/운동 루틴 탭 UI 디자인	맞춤형 식단/운동 루틴 탭 UI / Control Flow 설계	2.0	이종호 ▼
14	맞춤형 식단/운동 루틴 탭 (회원)	15	14.2	맞춤 식단 수신 기능 구현	트레이너가 전송한 맞춤 식단 수신 기능 구현	2.5	황성희 ▼
14	맞춤형 식단/운동 루틴 탭 (회원)	15	14.3	맞춤 운동 루틴 수신 기능 구현	트레이너가 전송한 맞춤 운동 루틴 수신 기능 구현	2.5	정의현 ▼
14	맞춤형 식단/운동 루틴 탭 (회원)	15	14.4	맞춤형 식단/운동 루틴 탭 Test	트레이너 추천 탭 기능 및 UI 확인/수정 기능 테스트	1.5	이종호 ▼
							▼
15	데이터 분석(회원)	10	15.1	데이터 분석 Flow	데이터 분석 Control Flow 설계	2.5	정의현 ▼
15	데이터 분석(회원)	15	15.2	데이터 분석환경 구축	Anaconda와 데이터 분석 라이브러리 등 환경구축 및 서버 연동	3.0	이종호 ▼
15	데이터 분석(회원)	20	15.3	데이터 전처리	사용자 데이터 전처리 (데이터 분류 및 정규화)	2.0	이종호 ▼
15	데이터 분석(회원)	20	15.4	데이터 분석	사용자 데이터 분석 및 결과 도출 코드 작성	2.5	정의현 ▼
15	데이터 분석(회원)	15	15.5	데이터 결과 화면 개발	결과 도출 화면 구축 등 데이터 UI 구성	2.5	황성희 ▼
15	데이터 분석(회원)	10	15.6	DB 및 서버 연동 작업	구축된 서버와 안드로이드에서 필요한 데이터를 연동	3.5	정의현 ▼
15	데이터 분석(회원)	20	15.7	데이터 분석 Test	분석기능 테스트 및 UI Test	2.5	황성희 ▼
							▼
16	트레이너 메인 화면 - 회원 관리	10	16.1	트레이너 회원 관리 화면 디자인	회원 관리 화면 UI 설계 / Control Flow 작성 (로그인 이후 접근 가능)	2.0	하민종 ▼
16	트레이너 메인 화면 - 회원 관리	25	16.2	트레이너 회원 관리 화면 구현	UI 상호작용 구현	2.5	황성희 ▼
16	트레이너 메인 화면 - 회원 관리	15	16.3	회원 관리 화면 Test	회원 관리 프로필 화면 UI 확인 및 Test	1.5	정의현 ▼
							▼
17	맞춤형 식단/운동 루틴 탭 (트레이너)	10	17.1	트레이너 식단/루틴 탭 디자인	맞춤형 식단/운동 루틴 탭 UI / Control Flow 설계	1.5	황성희 ▼
17	맞춤형 식단/운동 루틴 탭 (트레이너)	15	17.2	회원 요청 알림 기능 구현	회원의 식단/운동 루틴 요청 알림 구현	2.5	정의현 ▼
17	맞춤형 식단/운동 루틴 탭 (트레이너)	15	17.3	스케줄러 기능 구현	식단/운동 루틴 스케줄러 기능 구현	2.5	이종호 ▼
17	맞춤형 식단/운동 루틴 탭 (트레이너)	10	17.4	식단/운동 루틴 탭 Test	식단/운동 루틴 탭 알림 및 스케줄러 기능 Test	1.5	황성희 ▼
							▼

Story ID	User Stories	Priority	Task ID	Task	Task Description	Time Estimate (days)	Assigned To
19	발표 및 피드백	10	19.2	발표 자료 작성	진행사항 정리 및 발표 자료 작성	3.0	이종호 ▼
19	발표 및 피드백	15	19.3	발표 및 피드백	최종 발표 및 피드백 수용	1.5	정의현 ▼
							▼
							▼



### 6.1.2. 백로그에 기반한 Iteration 계획

Iteration 1	
Start Date	3/25/2019
End Date	4/5/2019
# of Developers	5
Efficiency Factor	0.6
Work Days	10
Man Days	50
Effective Man Days	30
m - Ideal Remaining Effort	-3

Total Tasks Assigned	30.0
Available Time - Assigned time	0.0

[illegible]

Iteration 2	
Start Date	4/8/2019
End Date	4/19/2019
# of Developers	5
Efficiency Factor	0.6
Work Days	10
Man Days	50
Effective Man Days	30
m - Ideal Remaining Effort	-3

Total Tasks Assigned	21.0
Available Time - Assigned time	9.0

Task Id	Title	Estimate	Status	Date of Completion	Time Spent
5.1	회원가입 체계회의 및 Flow 제작	1.0	Complete ▼	4/8/2019	1.0
5.2	회원가입 화면 Design	1.0	Complete ▼	4/9/2019	2.0
5.3	회원가입 화면 개발	2.0	Complete ▼	5/9/2019	2.0
6.1	사용자 프로필 화면 디자인	1.0	Complete ▼	4/15/2019	2.5
6.2	사용자 프로필 화면 구현	2.0	Complete ▼	5/6/2019	2.0
6.3	프로필 화면 Test	1.0	Complete ▼	5/7/2019	1.0
7.1	필요 시스템 분석	2.0	Complete ▼	4/12/2019	3.5
7.2	DB 설계	2.0	Complete ▼	4/18/2019	3.0
7.3	메인 서버 구축	2.0	Complete ▼	4/18/2019	2.0
8.1	카카오톡 로그인 API 분석	1.0	Complete ▼	4/8/2019	1.5
8.2	카카오톡 로그인 인증 구현	1.0	Complete ▼	4/9/2019	2.0
8.3	회원가입 및 로그인 Test	1.0	Complete ▼	5/9/2019	1.0
10.1	기본 UI 설계	2.0	Complete ▼	4/18/2019	2.0
10.2	PoseEstimationForMobile 적용	2.0	Complete ▼	5/3/2019	3.0
			▼		
			▼		
			▼		
			▼		
			▼		

Iteration 3	
Start Date	4/29/2019
End Date	5/10/2019
# of Developers	5
Efficiency Factor	0.6
Work Days	8
Man Days	40
Effective Man Days	24
m - Ideal Remaining Effort	-3

Total Tasks Assigned		25.5			
Available Time - Assigned time		-1.5			
Task Id	Title	Estimate	Status	Date of Completion	Time Spent
4.3	로그인 화면 개발	2.5	Complete ▼	5/1/2019	2.0
5.3	회원가입 화면 개발	2.0	Complete ▼	5/8/2019	2.0
6.2	사용자 프로필 화면 구현	2.0	Complete ▼	5/6/2019	2.0
6.3	프로필 화면 Test	1.0	Complete ▼	5/7/2019	1.0
8.3	회원가입 및 로그인 Test	1.0	Complete ▼	5/9/2019	1.0
10.2	PoseEstimationForMobile 적용	2.0	Complete ▼	5/3/2019	3.0
10.3	이미지 처리 후 데이터 후처리	4.0	Complete ▼	5/13/2019	3.0
10.4	처리받은 값으로 피드백	2.0	Complete ▼	5/16/2019	2.0
12.1	카카오맵 API 분석	2.0	Complete ▼	5/13/2019	2.0
12.2	거리좌표에 따른 트레이너 정렬	1.0	Complete ▼	5/20/2019	1.0
12.3	트레이너 추천 탭에 병합	1.0	Complete ▼	5/21/2019	1.0
9.1	운동선택 Flow 설계	1.0	Complete ▼	5/1/2019	1.0
9.2	운동선택 UI 작성	2.0	Complete ▼	5/1/2019	1.0
9.3	운동선택 UI 구현	2.0	Complete ▼	5/10/2019	2.0
			▼		
			▼		
			▼		
			▼		
			▼		

- Iteration3부터는 본래 계획대로 진행되지 못하고 연기되었다.



Iteration 4	
Start Date	5/13/2019
End Date	5/24/2019
# of Developers	5
Efficiency Factor	0.6
Work Days	10
Man Days	50
Effective Man Days	30
m - Ideal Remaining Effort	-3

Total Tasks Assigned		40.5			
Available Time - Assigned time		-10.5			
Task Id	Title	Estimate	Status	Date of Completion	Time Spent
11.1	트레이너 추천 탭 UI 디자인	1.5	Complete ▼	5/13/2019	1.0
11.2	트레이너 추천 탭 UI 구현	2.5	Complete ▼	5/14/2019	2.0
11.3	트레이너 추천 탭 Test	1.5	Complete ▼	5/20/2019	2.0
13.1	유튜브 API 분석	2.5	Complete ▼	5/15/2019	3.0
13.2	유튜브 계정 연동	2.5	Complete ▼	5/21/2019	2.0
13.3	유튜브 관련 UI 구현	1.0	Complete ▼	5/18/2019	2.0
14.1	맞춤형 식단/운동 루틴 탭 UI	2.0	Complete ▼	5/22/2019	2.0
14.2	맞춤 식단 수신 기능 구현	2.5	Complete ▼	5/25/2019	1.5
14.3	맞춤 운동 루틴 수신 기능 구현	2.5	Complete ▼	5/26/2019	1.5
14.4	맞춤형 식단/운동 루틴 탭 Test	1.5	Complete ▼	5/27/2019	1.0
15.1	데이터 분석 Flow	2.5	Complete ▼	5/14/2019	1.0
15.2	데이터 분석환경 구축	3.0	Complete ▼	5/16/2019	2.5
15.3	데이터 전처리	2.0	Complete ▼	5/18/2019	2.0
15.4	데이터 분석	2.5	Complete ▼	5/21/2019	2.5
10.3	이미지 처리 후 데이터 후처리	4.0	Complete ▼	5/13/2019	3.5
10.4	처리받은 값으로 피드백	2.0	Complete ▼	5/16/2019	2.0
12.2	거리좌표에 따른 트레이너 정렬	1.0	Complete ▼	5/20/2019	1.0
12.3	트레이너 추천 탭에 병합	1.0	Complete ▼	5/21/2019	1.0
15.5	데이터 결과 화면 개발	2.5	Complete ▼	5/23/2019	3.0

- Iteration 4부터는 주말 없이 개발을 진행하였다.

Iteration 5	
Start Date	5/25/2019
End Date	6/13/2019
# of Developers	5
Efficiency Factor	0.6
Work Days	13
Man Days	65
Effective Man Days	39
m - Ideal Remaining Effort	-3

Total Tasks Assigned	46.0
Available Time - Assigned time	-7.0

Task Id	Title	Estimate	Status	Date of Completion	Time Spent
15.5	데이터 결과 화면 개발	2.5	Complete ▼	5/23/2019	3.0
15.6	DB 및 서버 연동 작업	3.5	Complete ▼	5/26/2019	5.0
15.7	데이터 분석 Test	2.5	Complete ▼	5/28/2019	2.0
16.1	트레이너 회원 관리 화면 디자인	2.0	Complete ▼	5/26/2019	1.0
16.2	트레이너 회원 관리 화면 구현	2.5	Complete ▼	5/27/2019	1.0
16.3	회원 관리 화면 Test	1.5	Complete ▼	5/28/2019	1.0
17.1	트레이너 식단/루틴 탭 디자인	1.5	Complete ▼	5/26/2019	1.0
17.2	회원 요청 알림 기능 구현	2.5	Complete ▼	5/27/2019	1.0
17.3	스케줄러 기능 구현	2.5	Complete ▼	5/30/2019	3.0
17.4	식단/운동 루틴 탭 Test	1.5	Complete ▼	5/31/2019	1.0
18.1	전체 기능 점검	2.0	Complete ▼	6/3/2019	2.0
18.2	오류 수정	4.0	Complete ▼	6/7/2019	4.0
18.3	최종 시연	1.0	Complete ▼	6/11/2019	1.0
19.1	최종 진행사항 파악	1.5	Complete ▼	6/6/2019	1.5
19.2	발표 자료 작성	3.0	Complete ▼	6/10/2019	2.0
19.3	발표 및 피드백	1.5	Complete ▼	6/13/2019	1.5
14.1	맞춤형 식단/운동 루틴 탭 UI	2.0	Complete ▼	5/22/2019	2.0
14.2	맞춤 식단 수신 기능 구현	2.5	Complete ▼	5/25/2019	1.5
14.3	맞춤 운동 루틴 수신 기능 구현	2.5	Complete ▼	5/26/2019	1.5
14.4	맞춤형 식단/운동 루틴 탭 Test	1.5	Complete ▼	5/27/2019	1.0
10.4	처리받은 값으로 피드백	2.0	Complete ▼	5/16/2019	2.0

- 최종 시연에는 모든 작업을 완료할 수 있었다.

## 6.2. 협업 방식/도구 설명

### Cooperative tools



#### 6.2.1. Git

- branch를 통한 소스 관리와 개발 상태 공유에 좋은 git를 사용했다.
- 편리한 GUI 방식을 제공해주는 SourceTree와 Github Desktop을 사용했다.

#### 6.2.2. Slack

- 본 팀은 크게 PoseEstimation을 담당하는 pose팀과 그 외의 모든 것을 담당하는 팀으로 나뉜다.
- 그 외를 담당하는 팀은 다시 UI를 담당하는 Front End팀과 서버 및 데이터를 담당하는 Back End팀으로 나뉜다.
- 한 개인이 한 팀에만 소속되어 있는 게 아닌, 한 개인이 여러 복수의 팀에 소속되어 있어 각 팀마다 고유한 채널이 필요하다.
- Slack으로 각 팀별로 고유의 채널을 만들어 역할 수행에 혼선이 없이 각 팀의 목표를 달성했다.

## 7. 성과

### 7.1. 특허, 소프트웨어 등록

- 해당사항 없음.

### 7.2. 학술논문발표 등, 정량화 될 수 있는 기타 성과

- '아주 Greative 소프트웨어 콘서트' 장려상 수상

## 8. 결론

### 8.1. 프로젝트 진행(실행) 과정에서 잘 된(한) 점 그 이유

본 프로젝트를 진행하며 큰 목표를 3개 가지고 시작을 했다. 1) 사용자의 운동자세 실시간 피드백을 통한 자세 교정, 2) 사용자 운동 데이터 자동 관리 및 분석 제공, 3) 트레이너에게 식단/루틴 요청 기능을 통한 운동의 전문화. 이를 진행함에 있어서 초기에 기술적 장벽이 크게 다가왔다. 핵심 기술 중 하나인 Pose Estimation을 전문적으로 활용해본 팀원은 없었고 각기 몇몇 특정 분야에 것들을 수행할 수 있는 역량만을 가지고 있었다.

따라서 초기에는 자료를 보고 학습하는데 급급하여 일정에 차질이 발생하기도 했다. 하지만 전 팀원이 모두 본인이 맡은 바에 충실히 하였고, Iteration 계획을 맞추기 위해 부족한 부분은 서로 도우며 이끌어 나갔다. 그 결과 끝에는 Iteration 계획을 맞추며 프로젝트를 마칠 수 있었다.

또한 기능적으로 개선할 사항도 있지만, 1) 운동자의 동작을 인식하고 음성으로 피드백을 주는 기능, 2) 동작 인식을 하며 나온 데이터를 바탕으로 사용자의 운동기록을 자동으로 관리하고 데이터 분석을 제공하는 기능, 3) 사용자와 트레이너를 매칭 시켜주며 식단/운동 루틴을 메시지를 통해 주고받을 수 있는 기능을 구현하며 초기 목표를 달성할 수 있었다. 또한 부가적인 기능을 더하며 여러 사용자 편의 기능을 구현한 점이 잘 된 점이라고 할 수 있다.

### 8.2. 프로젝트 진행 과정에서 아쉬운 점, 개선 아이디어

프로젝트의 큰 목표는 달성을 했지만 그 완성도에 있어서 아쉬움이 남는다. 우선 Pose Estimation의 경우 정확도가 떨어지는 경향이 있다. 주변 사물이 사용자의 의복 색과 비슷하면 그 사물을 사용자의 인체 일부분으로 인식하여 값의 오류가 생기는 문제가 있었다. 이를 개선하기 위해 학습 모델을 개선을 해보았으나 학습 데이터의 부족으로 한계가 있었다. 또한 사용자의 체형에 따른 모든 경우의 수를 고려하기에 한계가 있어 여러 체형의 평균치로 적용하여 진행하였는데 이 진행 방식으로 오차가 발생하는 것이 아쉬웠다. 데이터 분석에서도 많은 데이터가 누적이 되어 유의미한 분석이 더욱 진행되었으면 좋았겠지만 개발 일정에 맞추어 몇 사람이 테스트를 진행을 하다 보니 많은 데이터를 확인하지 못해 분석 부분에서의 완성도도 개선이 필요하다. 이 점 역시 아쉬운 점이다.

그리고 아래 항목들은 프로젝트를 진행하면서 구현의 필요성은 느끼나, 시간이 부족하고 우선순위에 밀려 미처 구현하지 못했던 점이다.

- 1) 직접 각 운동에 맞는 정보를 입력해야 하므로 조사에 따라서 우선 순위가 높은 운동을 우선으로 구현했다. 우선 순위는 Nielsen Buzzword의 홈 트레이닝 시 가장 많이 하는 운동을 참조했다. 추후에 추가로 각 운동을 분석하고 입력을 진행해야 한다.

- 2) 앞으로 수익창출을 위해 식단/루틴 추천 부분에 결제 기능을 추가하여 실제로는 사용자와 트레이너 간의 금전적인 교환이 추가되어야 한다.

추가로 아래 항목은 중간 발표 중 다른 팀의 좋은 의견을 받았으나, 이 역시 시간이 부족하여 미처 구현하지 못한 아이디어이다.

- 1) 운동 기록을 소모한 열량(kcal)와 연관을 지어 사용자에게 제공을 해주면 다이어트를 진행하는 사용자들에게 도움이 될 것 같다.

### 8.3. 본 과제 결과물의 발전 방향

#### 8.3.1. 본 과제의 관련 분야에 대한 기여도

초기 아이디어를 바탕으로 유사 제품들을 확인했을 때, 홈 트레이닝 어플을 활용함에 있어서 많은 불편함이 존재했다. 홈 트레이닝 어플의 대부분은 사용자가 스스로 운동을 하고 운동기록을 관리하는 어플이 많았다. 직접 입력을 하고 관리를 하는 부분으로 인해 사용자의 목표 달성율이 낮다는 평이 많았다. 그리고 최근에 들어서는 운동 자세를 인식하고 카운팅을 해주는 어플이 가장 인기가 많았다. 하지만 올바른 자세를 애니메이션으로 반복해서 보여줄 뿐 피드백은 주지 않았다.

그러므로 본 프로젝트는 홈 트레이닝 서비스 시장에서 각 관절의 각도 계산을 통해 피드백을 준다는 의미를 제공했다. 기술개선이 필요한 부분이 있지만 특정 운동자세에 가장 주요한 관절의 각도를 분석하고 이에 맞춰 운동에 대한 피드백을 준다는 점은 앞으로 홈 트레이닝 시장에서 주요한 기술로 발전할 것이라 생각한다.

또한 사용자가 트레이너를 직접 선택하고 식단 및 운동 루틴을 요청하고 트레이너는 이를 응답하는 방식으로, 트레이너에게 홈 트레이닝 시장이라는 새로운 수입 경로를 제공하는 시도를 통해 IT발전에 발 맞춰 홈 트레이닝 IT업체와 트레이너의 새로운 상생 모델을 제안했다고 생각한다.

### 8.3.2. 기술적 관점에서의 발전 방향

- 운동 인식 부분에서 더 많은 표본을 학습시킨다면 인식의 정확도가 높아질 것이다.
- 운동 시 각도를 계산하는 부분에서 좀 더 좋은 알고리즘이 필요해 보인다. 실시간으로 입력되는 데이터를 좀 더 빠르고 정확하게 처리할 필요가 있다.
- 현재 인식이 가능한 운동이 2가지인데, 운동의 가짓수를 늘린다.
- 트레이너의 위치를 알려주는 카카오 다음 맵 API가 불러오는데 시간이 꽤 걸린다(약 3초). 이 시간을 AsyncTask 등을 통해 줄일 수 있으면 좋을 것이다.
- 트레이너들의 유튜브 영상 목록을 불러오는데 걸리는 시간을 줄인다.
- 트레이너가 이미 수락 버튼을 누른 유저에 대해서 수락 취소 버튼을 만든다.
- 데이터 분석의 표본을 더 학습시켜 정확도를 높인다.
- 트레이너가 유저의 데이터를 확인할 때, 그 유저의 상세 정보와 매일 기록된 운동 데이터를 확인할 수 있게 한다.
- 회원가입 부분에서 주소를 입력할 때 스피너 방식을 추가한다.
- 이미지 처리 성능을 높이기 위해 Picasso 등의 오픈소스를 활용한다.

### 8.3.3. 사업적 관점에서의 발전 방향: 향후 상용화 가능성

현재까지 각 운동 관절을 분석해주는 어플이 없는 것으로 조사됨에 따라 충분히 향후 상용화 가능성이 존재한다. 다만 이를 위해서는 앞서 언급한 기술이 개선됨을 전제로 한다. 본 서비스의 수익 모델은 크게 2가지로 나뉘 볼 수 있다. 우선 많은 사용자 유치를 하고, 운동관련 용품 등을 광고할 수 있는 플랫폼으로서의 기능을 통한 수익 창출을 생각해 보았다. 그리고 식단/운동 루트를 사용자가 요청할 때에는 그에 따른 합리적인 금액을 지불하며, 이때 발생하는 수수료를 통한 수익 모델을 생각했다. 이 두 수익 모델을 통한 기대 수익으로 운영을 하며 이윤을 추구하는 방향으로 발전한다면 향후 상용화 가능성이 존재한다고 생각한다.

## 9. 참고문헌

- Pose Estimation 관련

OpenPose: <https://github.com/CMU-Perceptual-Computing-Lab/openpose>

PoseNet: <https://github.com/tensorflow/tfjs-models/tree/master/posenet>

PoseEstimationForMobile: <https://github.com/edvardHua/PoseEstimationForMobile>

Mace: <https://github.com/XiaoMi/mace>

TensorFlow: <https://www.tensorflow.org/tutorials/keras?hl=ko>

## 10. 해당 프로젝트 github 링크

<https://github.com/loewe0804/Fit-Got-U>

서버 소스는 nodejs 파일 1개와 php 파일 십여 개가 전부이며 찾은 수정이 없어 따로 github 등을 사용하지 않았다.