



국민대학교
전자정보통신대학
컴퓨터공학부

캡스톤 디자인 I

종합설계 프로젝트

프로젝트 명	Smarlarm(스말람)
팀 명	<i>Sweetleep</i>
문서 제목	중간보고서

Version	1.5
Date	2020-APRIL-23

팀원	홍성표
	고민욱
	박대영
	박영민
	서민주
	양경식

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 개설 교과목 캡스톤 디자인I 수강 학생 중 프로젝트 " Smarlarm(스말람)"를 수행하는 팀 "SWEETLEAP"의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 "SWEETLEAP"의 팀원들의 서면 허락 없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역

Filename	중간보고서-
원안작성자	홍성표
수정작업자	박영민, 박대영, 서민주

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2020-04-19	홍성표	1.0	최초작성	
2020-04-22	박영민	1.1	수면패턴관련 수정	작성코드 기록
2020-04-22	박영민	1.2	프로젝트목표, 수면패턴	프로젝트목표(수면패턴) 수정 수면패턴 연구내용 추가
2020-04-23	홍성표	1.3	내용 수정 및 오타 수정	내용 수정 및 오타 수정
2020-04-23	박대영	1.4	오타 수정	오타 수정
2020-04-23	서민주	1.5	내용 수정 및 추가	프로젝트 목표(스마트알람서비스) 내용 수정 알람 추천 서비스 알고리즘, 프론트엔드 연구내용 및 수행내용 추가

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

내용

1	프로젝트 목표	4
2	수행 내용 및 중간결과	5
2.1	계획서 상의 연구내용	5
2.1.1	수면패턴 측정 알고리즘	5
2.1.2	알람 추천 서비스 알고리즘	8
2.1.3	프론트엔드	10
2.2	수행내용	13
2.2.1	수면패턴 측정 알고리즘	13
2.2.2	알람 추천 서비스	17
2.2.3	프론트엔드	18
3.	수정된 연구 내용 및 수정사항	21
3.1	수정된 내용	21
3.1.1	기존 시스템과 차별점	21
4.	향후 추진 내용	22
4.1	GPS연동 및 경로 파악	22
4.2	날씨 연동	23
5.	참고문헌	24

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

1 프로젝트 목표

본 프로젝트는 현대인들이 자신의 수면 패턴을 파악하고 그에 맞게 스스로 대응 할 수 있도록 도와주고자 한다. 또한 현대인들에게 필수가 된 알람 서비스는 실시간적으로 변화하는 교통상황 및 날씨를 파악하여 가장 최적의 알람 시간을 추천해 주는 데에 목표가 있다. 다음과 같은 3 가지 서비스를 중점적으로 제공한다.

- 수면 패턴 분석

스마트폰에 내장된 가속도계를 이용하여 자신의 수면을 기록한다. 사용자가 수면에 들기 전에 침대에 스마트폰을 놓고 잠에 든다. 수면 중 뒤척임을 감지하여, 수면 중 자신이 얼마나 움직이는지 그에 따른 수면 종류를 구분해 시각화 한다. 이를 통해 자신의 대략적인 수면주기를 파악할 수 있으며, 이는 개인의 대략적인 수면장애를 파악하는데 도움이 될 것이다.

- 스마트 알람 서비스

현대인에게 알람은 필수적인 기능이 되었다. 이에따라 기본적인 알람을 넘어 스마트한 알람 기능을 제공한다. Google 캘린더를 연동하고, 아침에 일어나 준비하는 시간과 교통수단을 입력 받으면 적당한 알람 시간을 추천한다. 만약 가는 길이 막힌다면, 혹은 기상상황이 좋지 않다면 조금 일찍 알람 시간을 제공한다.

- 정보제공

정보제공은 한 달 혹은 일주일 단위로 자신의 과거 수면 패턴 그래프를 모아 한 번에 보여주는 기능이다. 이런 기능을 통하여 어떤 상황에서 잠을 깊이 자는지 알 수 있으며 뒤척임, 소리를 통해 사용자의 건강 상태 또한 체크할 수 있다.

더불어 잠을 잘 때 듣기 좋은 노래 혹은 유용한 팁 등을 사용자에게 알려주어 깊은 숙면을 할 수 있도록 유도한다.

 <div> <p>국민대학교</p> <p>컴퓨터공학부</p> <p>캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

2 수행 내용 및 중간결과

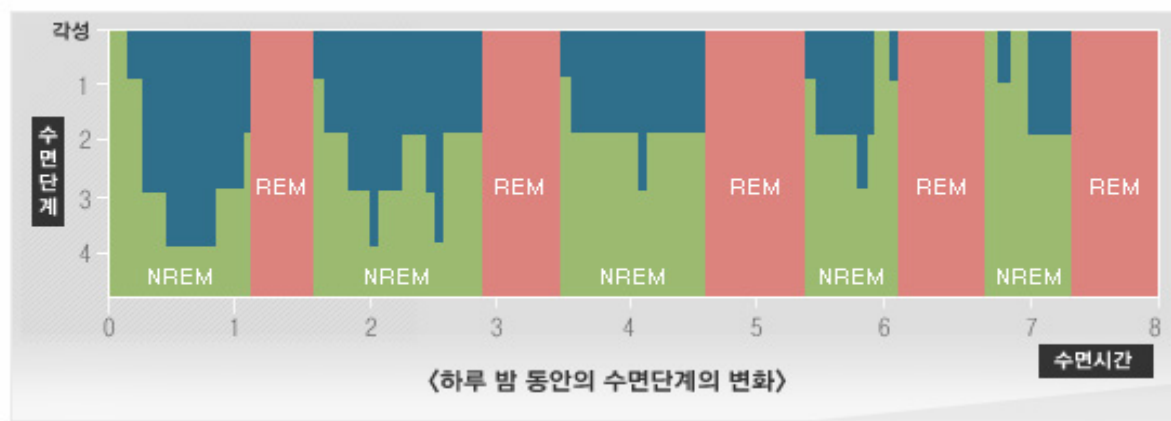
2.1 계획서 상의 연구내용

2.1.1 수면패턴 측정 알고리즘

인간의 수면은 다음과 같은 2가지 수면으로 이루어져 있다.

- **REM 수면** : 얇은 잠, 꿈을 꾸는 잠, 동공이 빠르게 움직임
- **N-REM 수면** : 깊은 잠, 1 ~ 4 단계로 이루어져 있음, 대뇌의 피로가 풀리는 시기

이러한 2 가지 수면종류는 다음과 같은 패턴으로 나타나게 된다.



잠든 직후 NREM수면으로 시작해 1단계(얇은 잠)부터 4단계(깊은 잠)까지 진행되며, 각 단계는 5~ 15분입니다. 그 후 REM수면에 진입하게 됩니다. REM수면은 깨기 직전의 얇은 잠이라고 볼 수 있다.

저희는 이러한 인간의 수면패턴에서 REM수면이 끝나고 다시 NREM수면으로 진입할 때, 신체의 상태변화를 주목하고자 한다. 각종 수면관련 연구결과에 따르면 이 시기에는 대뇌의 움직임이 활발한 시기이기 때문에, 그에 수반되는 각종 신체상태가 활발하다고 한다. 그 중 저희는 수면 중 뒤척임을 감지 및 시각화 하여 자신의 대략적인 수면주기를 파악할 수 있는 기능을 구현하려고 한다.

결국 가속도 센서 기반으로 REM수면에서 N-REM수면으로 넘어가는 그 지점을 파악하는 것이 핵심과제가 될 예정이다. 다음 그림은 가속도 기반으로 측정한 REM-NREM 수면 주기 관련 논문 내용 중 일부이다.

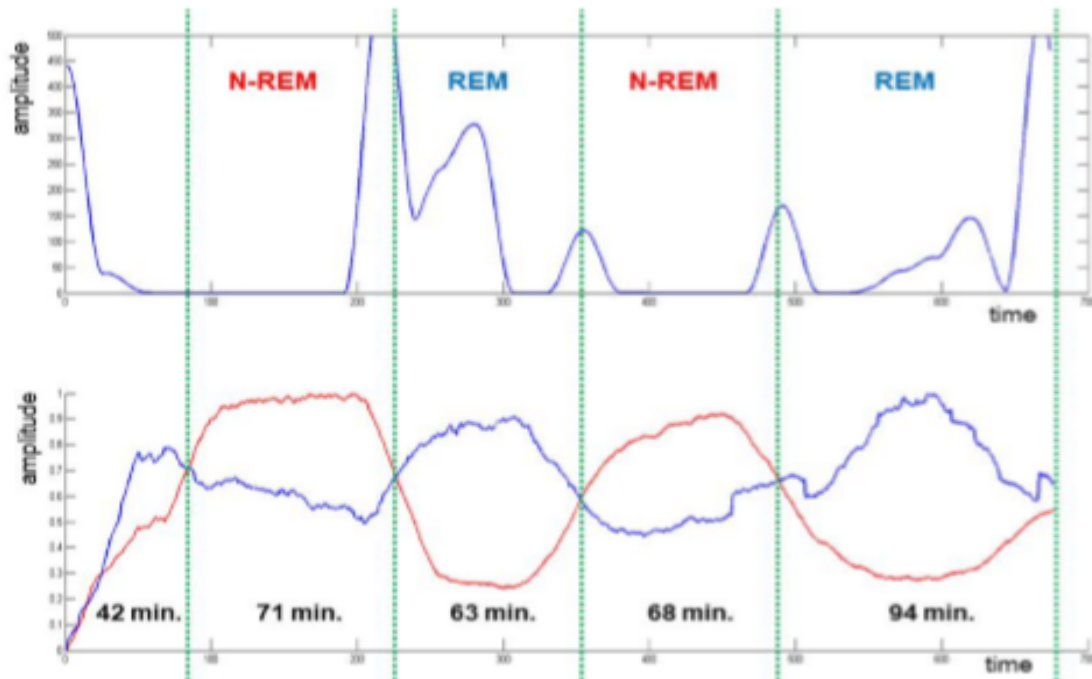


Fig. 5. Estimation of sleep periodic using moving information and EEG(upper : moving information, lower : power of δ -wave and Θ -wave).

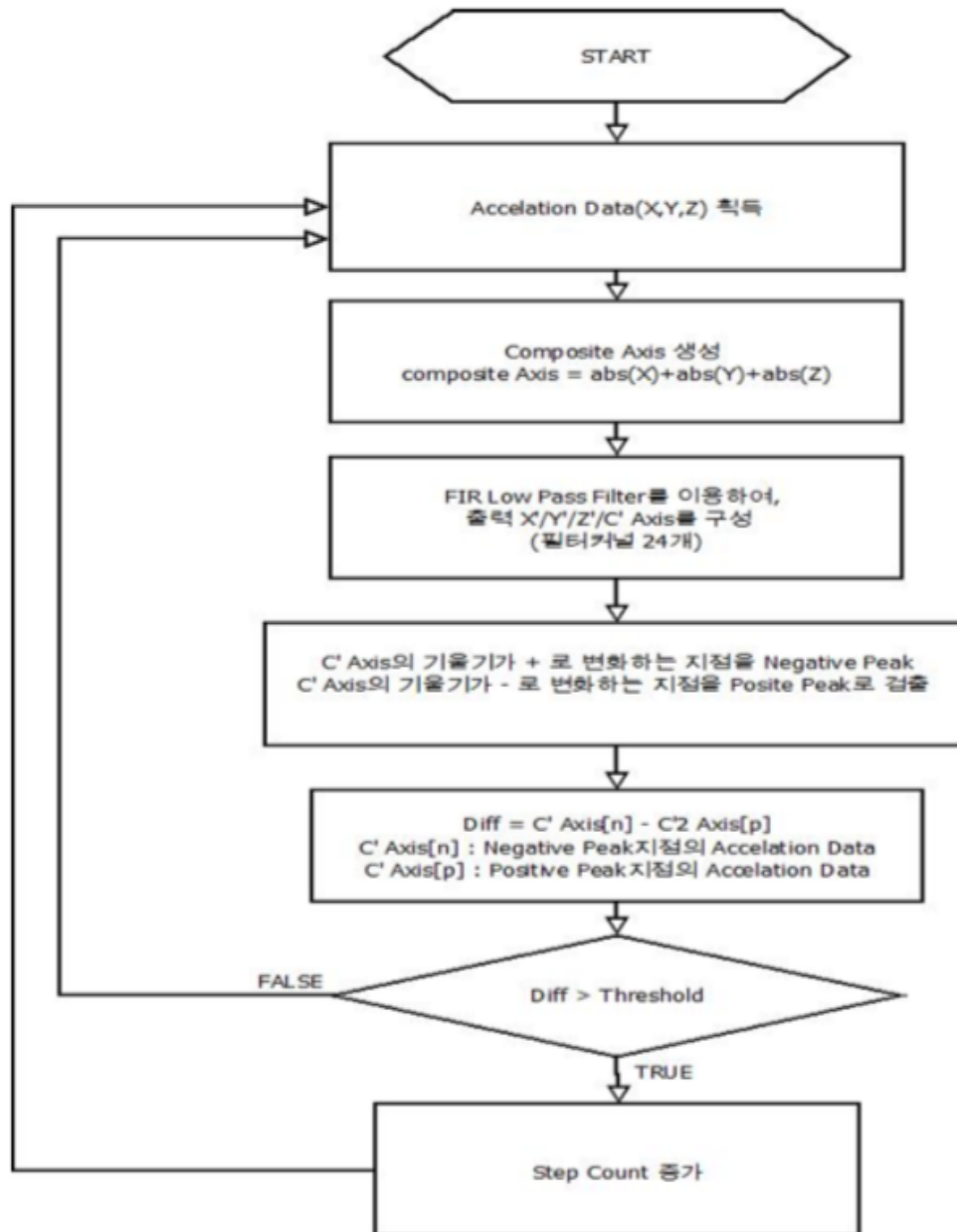
강경우, 김태선 (2019). 가속도 센서 데이터 기반 수면단계 예측 및 수면주기의 추정. 전기전자학회논문지, 23(4), 158-164.

그래프 상단은 가속도계 기반으로 한 움직임이며, 하단은 수면분석 DATA 중 하나인 EEG 생체 신호의 두가지 뇌파이다. 수면 중 움직임이 적은 N-REM 단계에서는 δ 뇌파가 θ 뇌파보다 파워가 우세한 것으로 나타나며, 반대로 REM 수면단계에서는 θ 뇌파가 δ 뇌파 보다 우세한 것으로 나타난다. 같은 구간에서 가속도계 기반 움직임은 REM 수면단계에서는 활발하고, N-REM 수면단계에서는 움직임이 거의 없는 경향을 보인다. 이러한 규칙성을 이용하여 저희는 수면다원검사(PSG) 보다는 정확도는 떨어지지만, 가속도 센서를 통해 수면단계를 구분해보고자 한다.

결국 가속도계 기반으로 측정한 움직임으로 수면종류를 구분하기 위해선, REM-NREM 수면구분을 위한 적절한 임계값(threshold)을 찾는 것이 중요한 과제가 될 것이다.

깊은 잠 시기(NREM수면 4단계)에서 기상을 하게 되면 기상 후에도 잠을 잔 것 같지 않은 그로기(groggy) 상태가 된다. 하지만 대뇌활성도가 높아지는 시기인 (REM수면-NREM수면1단계) 구간에서 기상을 하게 되면 상쾌하게 기상할 수 있다. 이러한 아이디어에 착안하여, 해당 시기에 일어날 수 있도록 도와주는 알람 서비스도 구현할 예정이다.

가속도계를 측정하는 방법은 다음과 같은 알고리즘 모델을 적용하였다.



(김영춘, 조문택, 이충식, 김갑수, 최만규, 주해중. (2015). 수면패턴 모니터링을 위한 가속도 센서 처리 알고리즘. 한국산학기술학회 학술대회논문집, (), 373-375.)

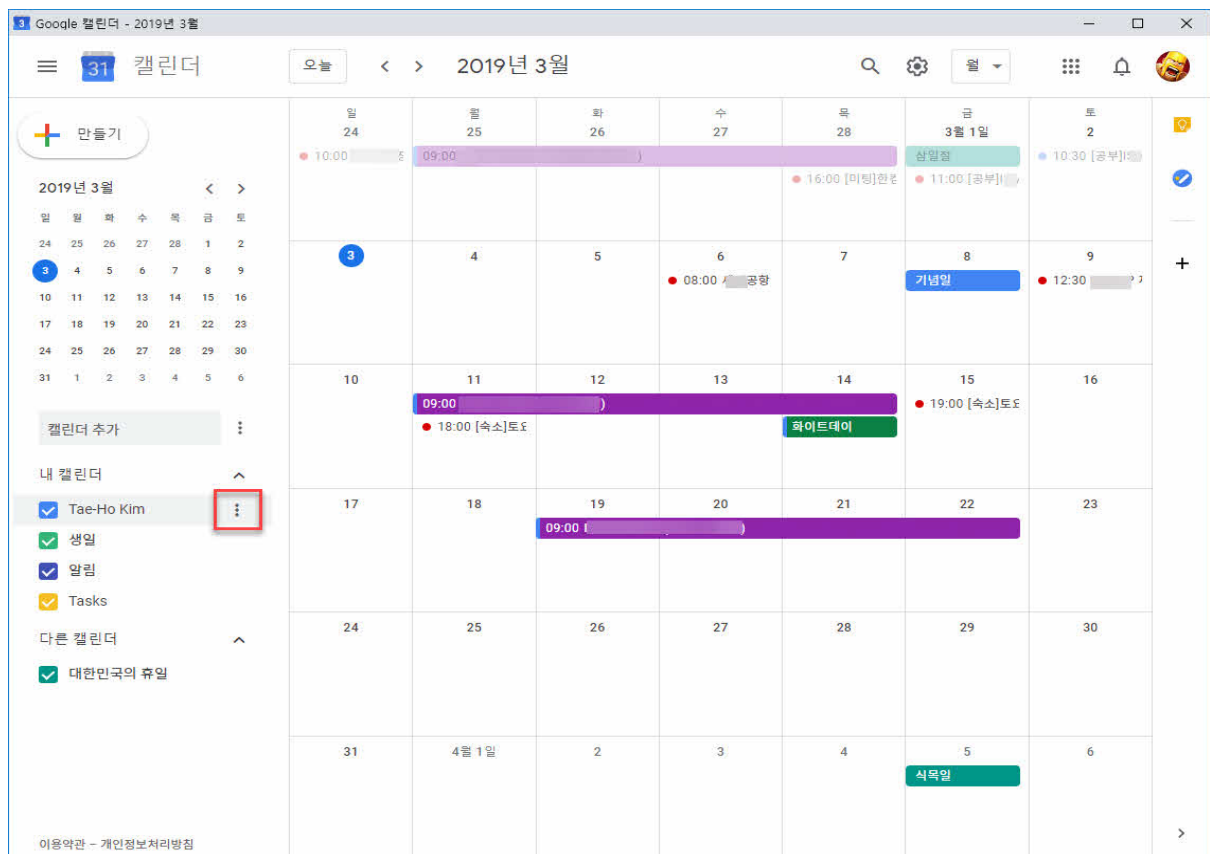
그리하여 저희는 현대인들이 자신의 수면상태를 별도의 장비가 없더라도, 스마트폰만으로도 자신의 수면주기를 파악할 수 있고 그에 기반한 스마트 알람 서비스를 구현하고자 한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

2.1.2 알람 추천 서비스 알고리즘

바쁜 현대인들에게 시간은 금과 같다. 어떻게 하면 더욱 시간을 효율적으로 사용할 것인지에 대한 고민은 끝없이 진행중에 있다. 요즘 중, 고등학교 학생들부터 대학생, 직장인들까지 일정이 많아지고 복잡해짐에 따라 일정관리가 필수가 되었고, 자연스럽게 일정관리 어플리케이션 사용자가 많아지고 있다. 대표적 일정관리 어플리케이션의 유형인 캘린더를 알람 어플리케이션과 연동시켜 사용자의 편의를 더욱 높이고자 다음과 같은 기능을 구현 할 예정이다.

우리 어플에서는 스마트폰에 연동되어 있는 Google 계정정보를 이용해 Google Calendar 와 연동한다. 그리고 사용자에게 준비시간/소요시간 및 주교통수단을 입력 받아 사용자가 언제 출발해야 할지 혹은 언제 일어나야 하는지 계산하여 제공한다.



이 기능의 핵심적인 부분은 일정 등록이다. Google Calendar API 를 통해 사용자가 기존에 등록해 놓은 일정을 불러오거나, 일정을 추가하도록 하여 알람을 설정하는데 도움을 준다. 일정 등록 시에 내용과 시작시간을 입력 받고 해당 데이터를 날짜 별로 불러온다. 알람 받을 일정을 선택한 후 반복 계획, 사운드 등을 설정한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

이렇게 등록된 일정은 전날 저녁 시간대에 FCM(Firebase Cloud Messaging) 서비스를 이용해 “내일 일정 알람을 등록 하시겠습니까?” 라는 메시지를 띄우고, 몇번의 터치로 간단히 알람을 설정할 수 있도록 도와준다.



또한 사용자가 준비시간, 목적지, 교통수단을 입력하게 되면 일정시작시간에 맞출 수 있게 알람 시간을 조정한다. 만약 목적지와 교통수단이 입력된다면, 교통상황을 체크하고 교통체증이 있는 경우 계획보다 조금 더 일찍 일어날 수 있는 알람을 울린다. 또는 알람 종료시간 기준으로 기상 상황(눈/비/태풍 등)이 좋지 않다면 역시 조금 더 일찍 알람을 울린다.

이 기능은 사용자가 원하지 않을 경우 설정에서 OFF할 수 있도록 한다.

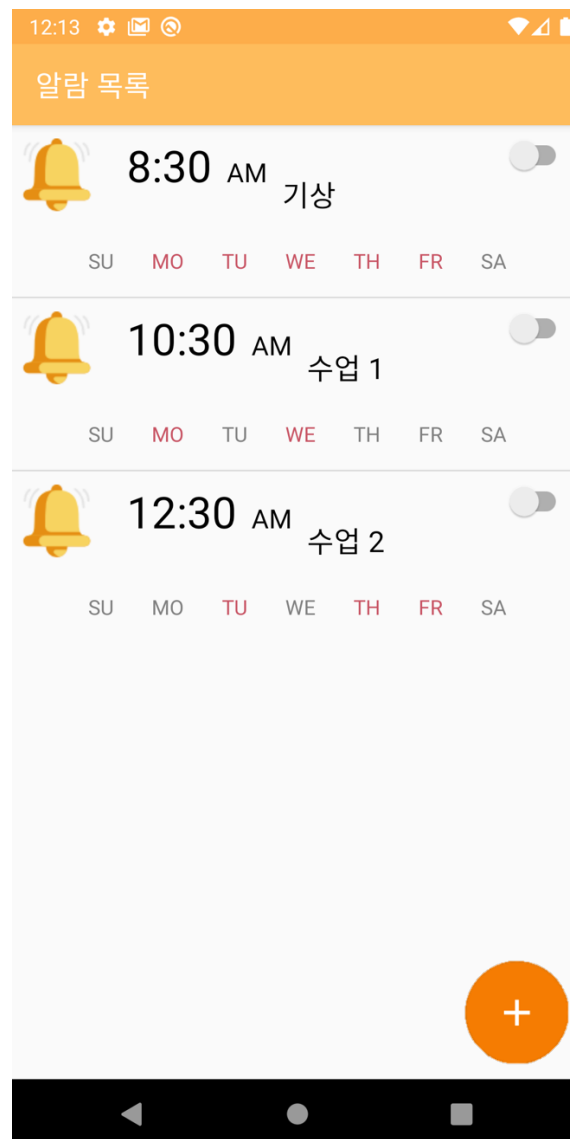
 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

2.1.3 프론트엔드

프론트엔드에서는 사용자가 수면패턴분석 기능 사용에 어려움을 느끼지 않도록 어플리케이션 첫 실행 시 사용방법을 알려주는 화면을 구성하여 어려움을 줄이고자 한다. 또한 최대한 직관적이고 복잡하지 않은 탭들로 구성할 예정이다.


1) 알람 탭

첫 화면은 알람이 가장 자주 사용하는 기능이기 때문에 홈 화면을 따로 만들지 않고 아래와 같은 방식의 UI를 사용한다. Switch 버튼을 toggle하여 ON/OFF가 가능하고, 설정된 알람시간, 일정 내용, 반복 요일 등을 확인할 수 있다. 클릭하면 세부적인 설정이 가능하도록 한다.



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

하단의 + 를 누르면 기본 알람을 설정할지, 일정과 연동해 알람을 설정할지 선택하도록 한다. 또한 수면 분석을 원하는 경우도 여기서 선택하도록 한다. 아래는 기본 알람 설정 화면이다.



The image shows a mobile app screen for setting an alarm. At the top, there's a back arrow and the title "< 알람". Below this is a large orange box containing a digital clock display. The time is set to 08:20, with "오전" (AM) on the left. Above the time, there are two columns of numbers: 06, 07, 08, 09, 10 on the left and 18, 19, 20, 21, 22 on the right. Below the time display, there are three rows of settings: "반복" (Repeat) set to "매일" (Every day), "사운드" (Sound) set to "상쾌한 아침" (Refreshing morning), and "다시 알림" (Repeat alarm) set to "켜기" (Turn on). At the bottom, there are two buttons: "알람 삭제" (Delete alarm) and "알람 저장" (Save alarm).

2) Bottom Navigation Bar

사용자들이 직관적으로 알 수 있도록 다음 세가지 탭으로 구성한다. 알람 탭을 누르게 되면 설정된 알람 리스트 및 알람을 설정할 수 있는 화면으로 이동한다. 그래프를 누를 시에는 지금까지 쌓았던 숙면그래프를 확인할 수 있도록 도와준다. 마지막 설정 탭을 통하여 개인정보 및 알람 소리 등 세부적인 내용을 설정할 수 있도록 도와준다.



3) 주의사항

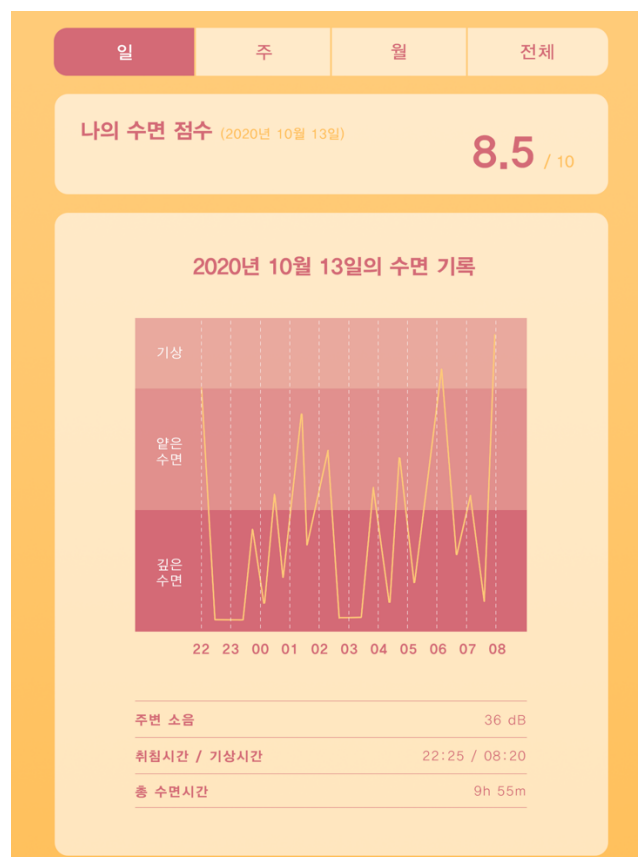
가속도 센싱을 통한 수면패턴그래프 이기 때문에 배터리 소모량이 많아 중간에 핸드폰이 꺼질 수 있는 경우도 생길 수 있다고 생각하여 구성한 UI 이다. 수면패턴분석에 들어가면 "핸드폰을 꼭 충전기에 연결해주세요" 라는 화면을 보여준다.

 <div> 국민대학교 컴퓨터공학부 캡스톤 디자인 I </div>	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23



4) 그래프 탭

그래프 탭을 이용하여 나의 수면 점수 및 수면 그래프를 확인할 수 있는 화면이다. 주변 소음 및 취침시간/기상시간 등을 쉽게 확인할 수 있으며, 일별 주별 월별을 선택하여 확인할 수도 있다.



 <div> <p>국민대학교</p> <p>컴퓨터공학부</p> <p>캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

2.2 수행내용

2.1.1 수면패턴 측정 알고리즘

실시간으로 가속도 좌표를 측정하여, 각 구간별(15분)로 움직임을 카운트하여 데이터로 표현해주는 식으로 구현하였다. 각 구간은 테스트를 통하여 정확도를 높이는 방향으로 설정할 예정이다. 아직 각 수면단계를 구분하는 부분은 정확히 구현하지 못했다.

해당 서비스는 안드로이드 백그라운드서비스에서 측정되어야 하므로 다음과 같은 method로 백그라운드 서비스를 실행시킨다. SDK 버전 26이상부터는 백그라운드 서비스가 막혀 있으므로, Notice를 통한 foreground 서비스를 호출한다.

```
public void startService() {
    Intent serviceIntent = new Intent(this, MyService.class);
    serviceIntent.putExtra("inputExtra", "수면 중 뒤척임 감지중입니다.");

    if (Build.VERSION.SDK_INT >= 26) {
        ContextCompat.startForegroundService(this, serviceIntent);
    }
    else {
        startService(serviceIntent);
    }
}
```

움직임이 감지된 경우 다음과 같은 onSensorChanged() 메소드에서 측정된 가속도계 좌표를 기반으로 속도(speed)를 구한 후 그 차이를 측정한다. 그 속도가 threshold(민감도) 보다 높다면, 의미 있는 움직임이라 보고 setMotioncounter() 메소드를 통해 해당 구간의 움직임 횟수를 증가시켜준다. 구간의 길이는 15분이며, 데이터 저장 형식은 hashmap을 사용하였다.

```
@Override
public void onSensorChanged(SensorEvent event) {
    if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {
        // 움직인 속도를 구하는 과정
        long currentTime = System.currentTimeMillis();
        long gabOfTime = (currentTime - lastTime);
        if (gabOfTime > 100) {
            lastTime = currentTime;
            x = event.values[SensorManager.DATA_X];
            y = event.values[SensorManager.DATA_Y];
            z = event.values[SensorManager.DATA_Z];

            speed = Math.abs(x + y + z - lastX - lastY - lastZ) / gabOfTime * 10000;

            // 속도가 설정값보다 크면 움직인 길로 조건에 걸림 ( 너무 큰 움직임은 측정 안함 )
        }
    }
}
```



```

if ( (speed > SHAKE_THRESHOLD) ){

    // 15 분 단위로 할때
    Calendar cal = Calendar.getInstance();
    int minute = cal.get(Calendar.MINUTE);
    int time = 0;

    if(0 <= minute && minute < 15){ // 0 ~ 14 분
        time = (cal.get(Calendar.HOUR_OF_DAY) * 60) + 15;
    }
    else if (15 <= minute && minute < 30){ // 15 ~ 29 분
        time = (cal.get(Calendar.HOUR_OF_DAY) * 60) + 30;
    }
    else if (30 <= minute && minute < 45){ // 30 분 ~ 44 분
        time = (cal.get(Calendar.HOUR_OF_DAY) * 60) + 45;
    }
    else if (45 <= minute && minute < 60){ // 45 분 ~ 59 분
        time = (cal.get(Calendar.HOUR_OF_DAY) * 60) + 60;
    }

    // 해쉬맵에 저장
    this.setMotionCounter(time);
}
lastX = event.values[DATA_X];
lastY = event.values[DATA_Y];
lastZ = event.values[DATA_Z];
}
}

// 해당 구간별 움직임을 담는 메소드
public void setMotionCounter(int time){
    if(countHash.get(time) == null){
        countHash.put(time, 1);
    }
    else {
        int temp = countHash.get(time);
        if(temp < 40) {
            countHash.put(time, temp + 1);
        }
    }
}
}
}

```

측정한 구간별 움직임 데이터를 보정작업 (빈값 메꾸기 등)을 거쳐서, 다음과 같은 ArrayList로 변환 후 각 println(), draw() 메소드를 통해 데이터를 표현해준다.

```

ArrayList<Integer> time = (ArrayList<Integer>) intent.getSerializableExtra("time");
ArrayList<Integer> motionCounter = (ArrayList<Integer>) intent.getSerializableExtra("motionCounter");
ArrayList<Integer> preTime = (ArrayList<Integer>) intent.getSerializableExtra("preTime");

```

 <div> <p>국민대학교</p> <p>컴퓨터공학부</p> <p>캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

```
printInfo(preTime, time, motionCounter); // 수면시간, 수면질, 구간별 움직임
draw(time, motionCounter); // 그래프
```

각 수면 종류를 판별 및 수면 질 계산 부분은 현재 개발단계에 있으며, 현재는 기상상태를 40으로 설정 후 (움직임이 없는 0은 **깊은 잠**, 30%에 해당하는 0~12까지를 **얕은 잠**, 70%에 해당하는 13~39까지를 **렘 수면**) 으로 설정하여 수면 퀄리티를 계산해보았다.

```
// 수면질 계산 메소드
private double setSQ(ArrayList<Integer> motionCounter){

    double SQ;
    for(int i=0; i<stepSleep.length; i++){
        stepSleep[i] = 0;
    }

    for(int i : motionCounter){

        if( i==0 ){ // 뒤척임 없음 stepSleep[0]
            stepSleep[0] += 15;
        }
        if( (i>0) && (i<=12) ){ // 뒤척임 적음 stepSleep[1]
            stepSleep[1] += 15;
        }
        if( (i>12) && (i<=39) ){ // 뒤척임 많음 stepSleep[2], max 값 40 은 기상상태
            stepSleep[2] += 15;
        }
    }
    SQ = (((stepSleep[2]*0.5) + (stepSleep[1]*0.75) + (stepSleep[0])*100)/(stepSleep[0] +
stepSleep[1] + stepSleep[2]));
    return SQ;
}
```

그래프는, 안드로이드 차트 API 중 가장 유명한
MPAndroidChart (<https://github.com/PhilJay/MPAndroidChart>) 를 이용하였다.

```
LineData data = new LineData(xaxes,lineDataSet1);
lineChart.setData(data);
lineChart.setVisibleXRangeMaximum(65f);
lineChart.getLegend().setEnabled(false);
lineChart.setDescription("뒤척임정도");
```

측정된 데이터를 표현한 결과는 다음과 같다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23



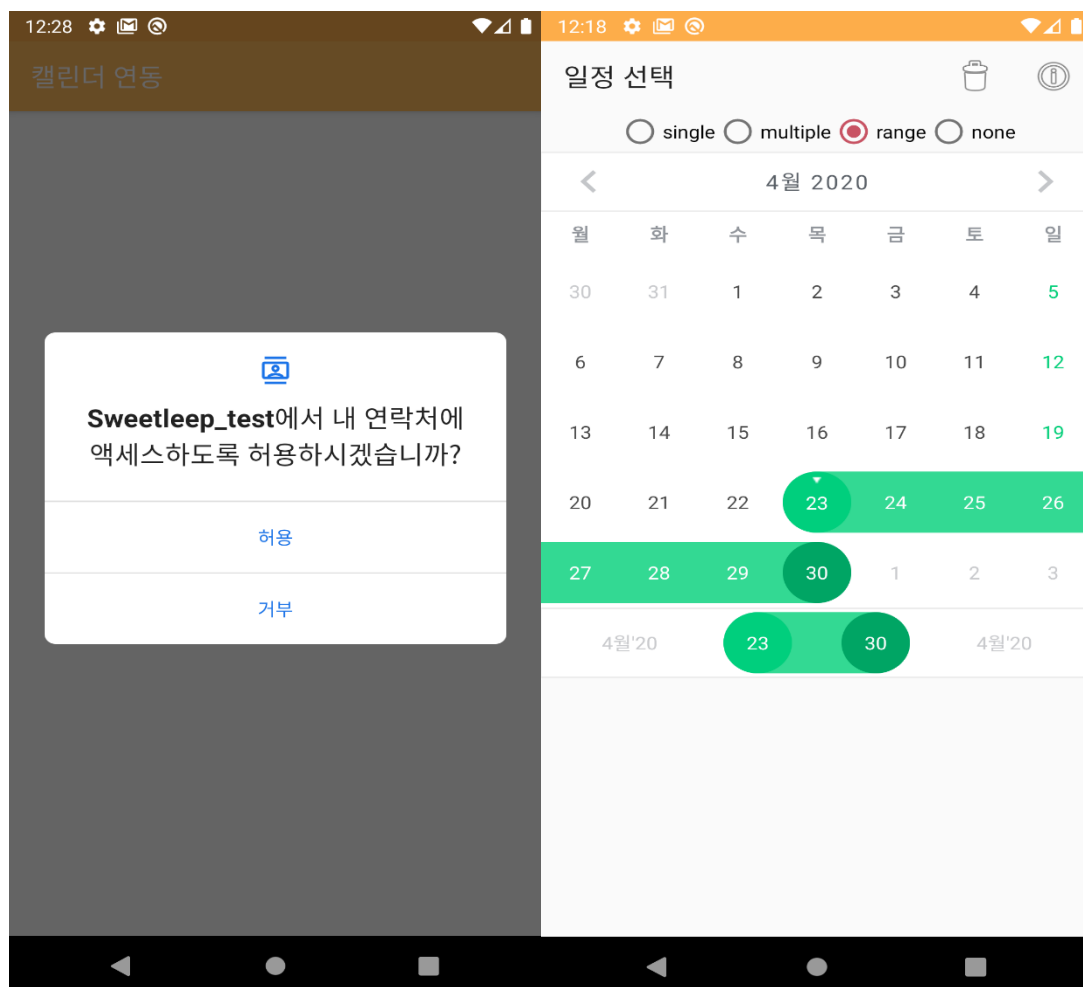
현재는 수면종류분류의 정확성을 보장할 수 없기 때문에, 뒤척임 정도로 표현을 하여 나타냈다.
 또한 수면 중 구간별 (15분) 움직임도 확인할 수 있다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

2.2.2 알람 추천 서비스

기존 서비스와 차별점을 두기 위한 서비스이다. 알람 설정을 위해 Google Calendar Api를 이용한 캘린더연동/생성, 일정 추가/확인 기능을 구현하였다.

1) 캘린더 연동



위와 같은 방법으로 지정된 캘린더에서 일정을 불러오고, 자신이 원하는 날짜를 선택하여 해당 일들의 일정 목록 및 일정 시작 시간 정보를 어플리케이션에 등록한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

2.2.3 프론트엔드

프론트 엔드에서는 사용자들이 직관적으로 알기 쉽도록 UI를 구성하였다. 처음 사용자들이 수면패턴 분석을 사용하는데 어려움이 없도록 앱 처음 실행에 사용방법을 알려주며 이 기능은 어플리케이션 첫 실행 시에만 보여주고, 이후 설정 탭을 통하여 다시 볼 수 있도록 만들었으며 그 외 부분은 탭 이름만 봐도 직관적으로 알 수 있도록 구성하였다. 또한 수면패턴분석시 주의사항도 직관적으로 알려준다.

1) 실행화면 Splash Activity

어플리케이션을 실행했을 경우의 화면이다. 앱이 구동되는 시간 동안 띄워준다.

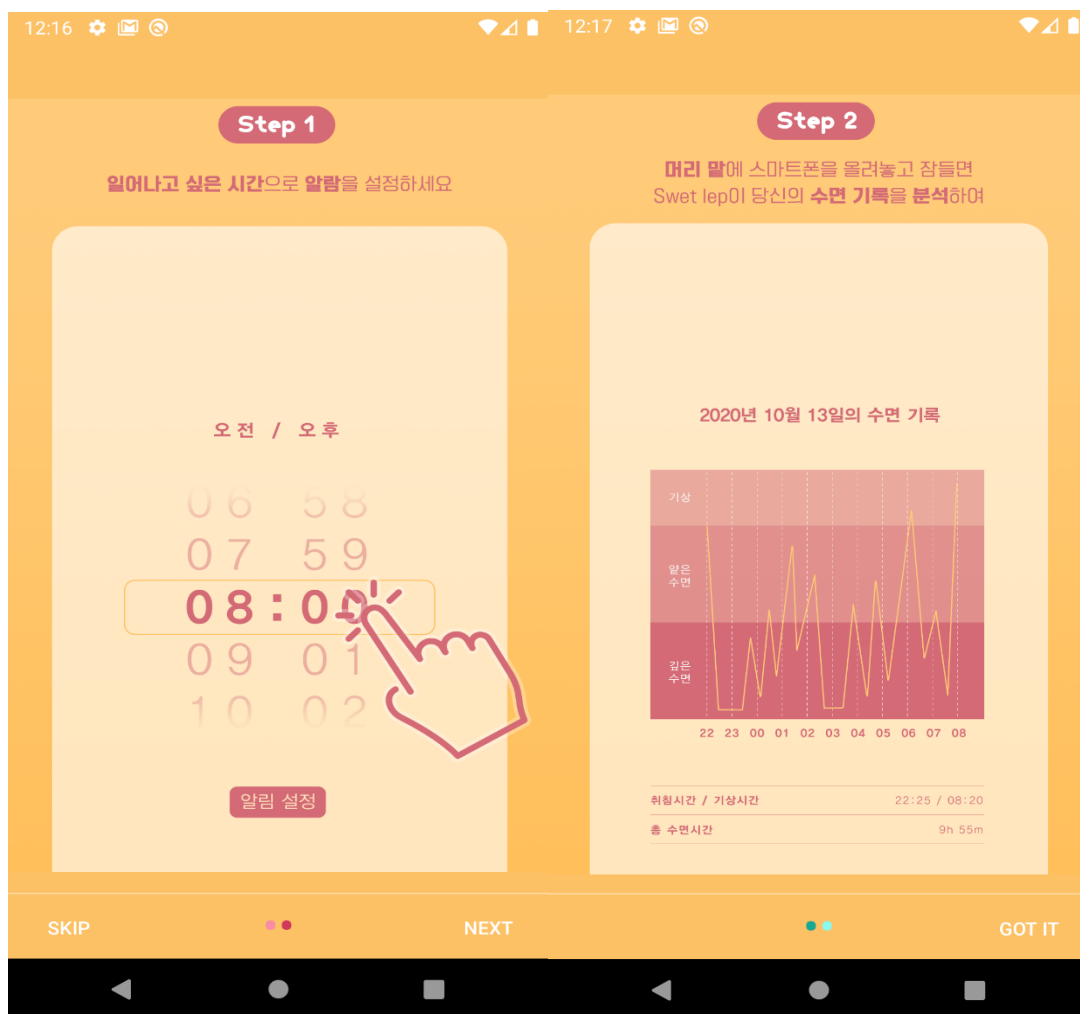


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

2) 튜토리얼 화면

Step1과 Step2를 통하여 사용자가 어떻게 수면패턴 및 알람 설정을 하는지에 대한 튜토리얼을 보여준다. 이 기능은 최초 실행 시 한번만 보여주며 그 이후에는 설정 탭을 통해서 다시 볼 수 있도록 하였다. 화면을 슬라이드하거나 NEXT를 누르면 다음 step으로 넘어가며, SKIP도 가능하다. 마지막 step에서는 GOT IT 버튼을 두어 튜토리얼 화면을 종료하도록 하였다.

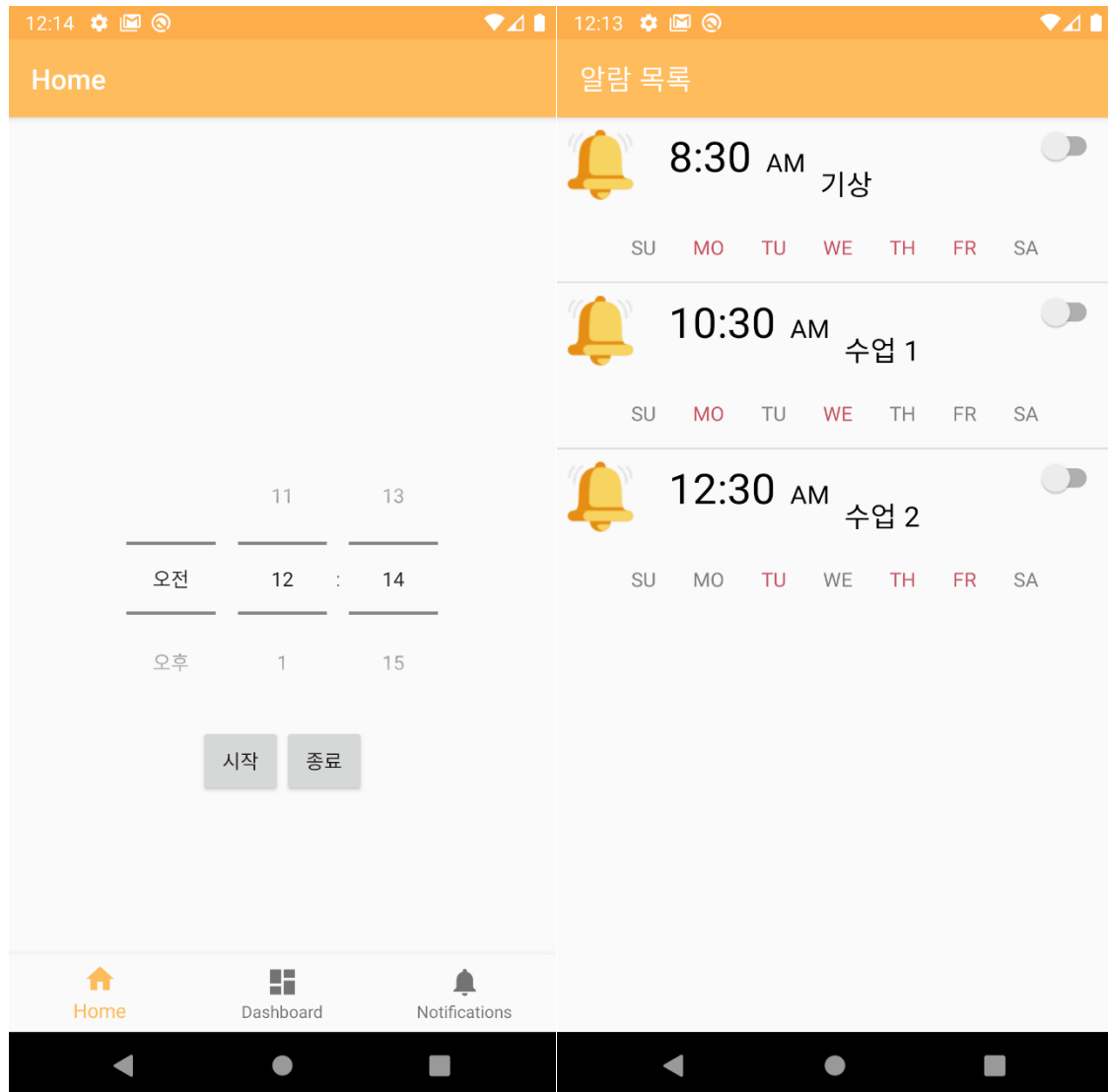
추후 Step을 추가할 예정이다.



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

3) 알람 설정화면

알람 설정화면이다. 내가 설정한 시간에 설정한 사운드로 알람을 설정한다. 설정한 알람은 목록에 저장되고, 목록에서 간단히 알람을 on/off 할 수 있다,



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

3. 수정된 연구 내용 및 수정사항

3.1 수정된 내용

3.1.1 기존 시스템과 차별점

기존 시스템과의 차별점이 가장 중요한 요소 중에 하나라고 볼 수 있다. 그러한 차별점은 우리는 수면방해요소에서 찾았다. 알람 맞추기 직전에 수면에 방해되는 요소를 가볍게 터치하여 사용자의 수면 방해 요소로 그래프를 클래스 분류하여 보여준다. 그렇게 되면 사용자마다 각각 어느 요소가 가장 방해되는지 알게 되고, 그 행동을 줄이게 유도한다.

수면패턴분석에 추가적으로 알람 서비스를 제공한다. 캘린더 연동을 통하여 알람 설정에 도움을 주며, 일정 시작 시간과 도착지 정보, 교통수단을 알게 되면 그날의 교통상황을 예상하여 조금 더 일찍 일어나야 할지 평소보다 여유를 가져도 될지에 따라 유동적으로 알려준다.

3.1.2 깊은 숙면 유도

사람의 5가지 감각을 통해서 깊은 숙면을 유도할 수 있다고 한다. 본 프로젝트는 핸드폰 어플리케이션이기에 오감 중 청각을 채택하여 깊은 숙면을 유도해준다. 통계적으로 사람들이 편안하게 느끼는 소리들을 담아 수면패턴분석시에 소리가 나을 수 있도록 한다. 잠이 들었고 판단되면 소리는 자연스럽게 종료된다. 빗소리, 바다소리, 산소리 등 여러가지 중에 사용자가 설정탭에서 선택할 수 있도록 한다. 혹시 원하는 소리가 있을경우 사용자 개인이 업로드하여 사용할 수 있도록 한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

4. 향후 추진 내용

4.1 GPS연동 및 경로 파악

스마트폰의 위치서비스를 사용하여 사용자 위치를 파악하여 목적지까지 걸리는 시간을 더욱 정확히 계산한다. 그렇게 되면 일정장소까지 걸리는 시간과 교통상황에 따른 알람시간 조정이 가능하다.



 <div> 국민대학교 컴퓨터공학부 캡스톤 디자인 I </div>	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleap	
	Confidential Restricted	Version 1.5	2020-04-23

4.2 날씨 연동

알람을 종료하는 화면에서 사용자가 날씨를 확인하고 우산을 챙겨야 할지 겉옷을 챙겨야 할지에 대한 인지를 할 수 있도록 종료화면에 오늘 날짜, 요일, 시간, 날씨를 표시해준다. 이렇게 되면 다시 확인하거나 잊어버리는 일을 줄여 준비시간 또한 절감할 수 있고 아침시간을 조금 더 효율적으로 사용할 수 있다.



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	Smarlarm(스말람)	
	팀 명	Sweetleep	
	Confidential Restricted	Version 1.5	2020-04-23

5. 참고문헌

- **가속도 센서 데이터 기반 수면단계 예측 및 수면주기의 추정.** 강경우, 김태선 (2019). 전
기전자학회논문지, 23(4), 158-164.
- **수면패턴 모니터링을 위한 가속도 센서처리 알고리즘.** 김영춘, 조문택, 이충식, 김갑수, 최
만규, 주해중. (2015). 한국산학기술학회 학술대회논문집, (), 373-375.