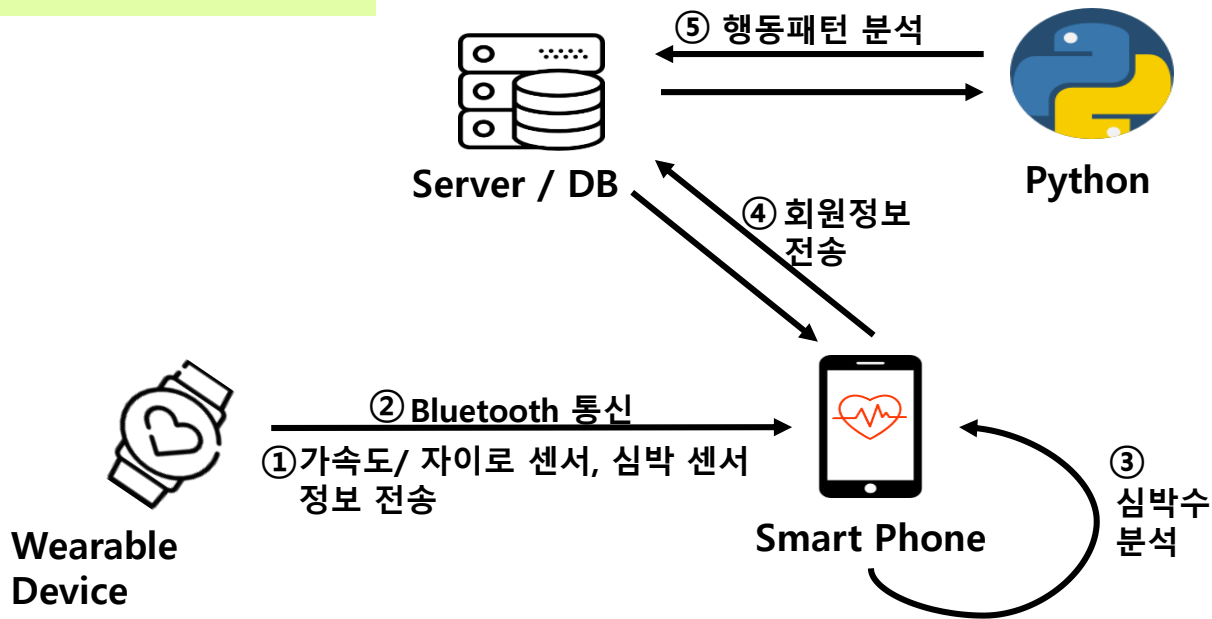


# Safety 서비스용 표준 IOT디바이스 상세 설계서

# Contents

1. 시스템 구성도
2. 시스템 흐름도
3. 메뉴 구성도(모바일)
4. 프로그램(기능) 목록
5. 기능 흐름도
6. 화면 설계서(모바일)
7. 모듈 설계서(HW)
8. 테이블 설계서(웹/모바일)
9. 프로그램 상세 로직
10. 개발 환경 (언어, Tool, 사용 시스템 등)

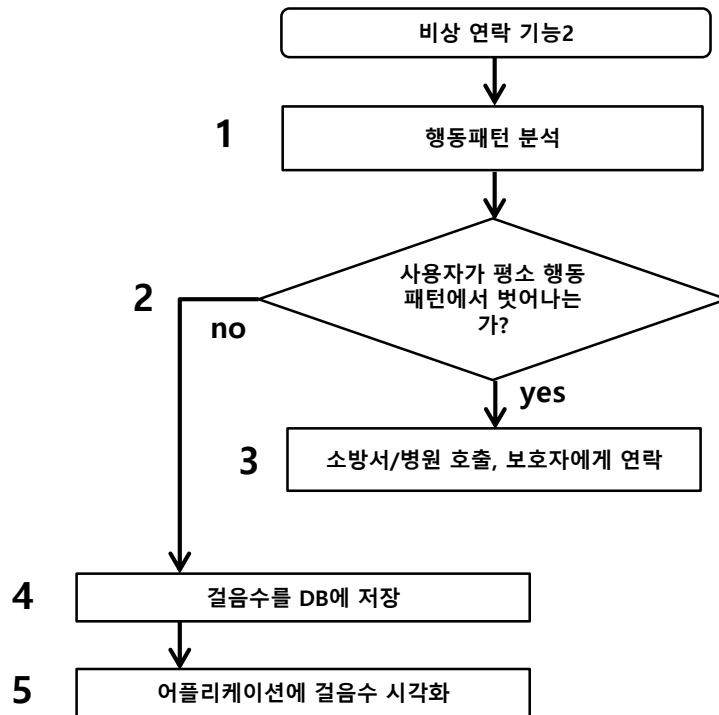
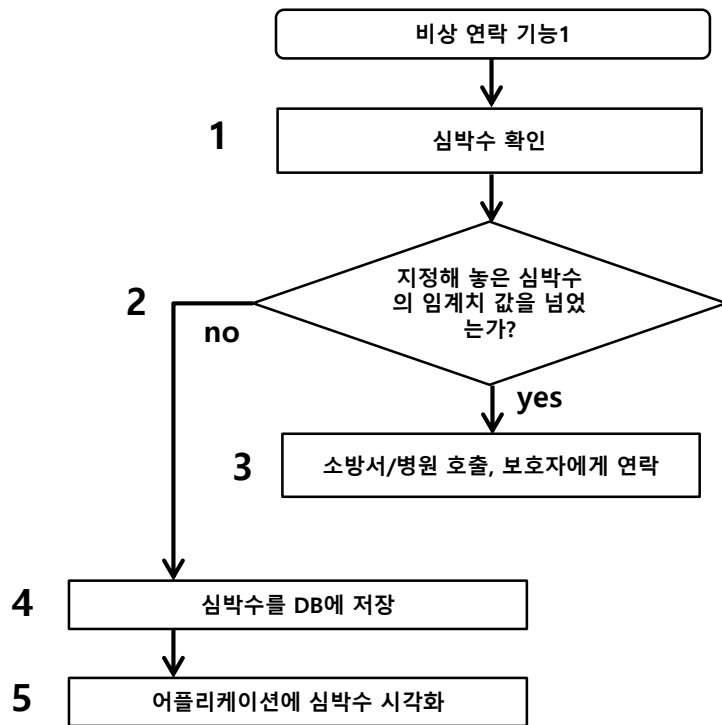
# 1. 시스템 구성도



## Key point

1. 가속도/자이로, 심박 센서만 부착된 저가의 Wearable Device
2. 심박수와 행동패턴 분석을 통해 위급상황 발생 시 보호자 및 기관에 연락을 취하는 어플리케이션. 사용자의 걸음 수와 심박수 정보를 전달하는 편리성을 위한 User Interface
3. 회원정보 및 가속도/자이로 센서로 부터 얻어진 데이터를 서버에 저장. 서버에 저장된 데이터를 통해 행동패턴 분석과 걸음 수를 측정 후 서버에 전송.

## 2. 시스템 흐름도\_비상연락기능



## 2. 시스템 흐름도\_비상연락기능

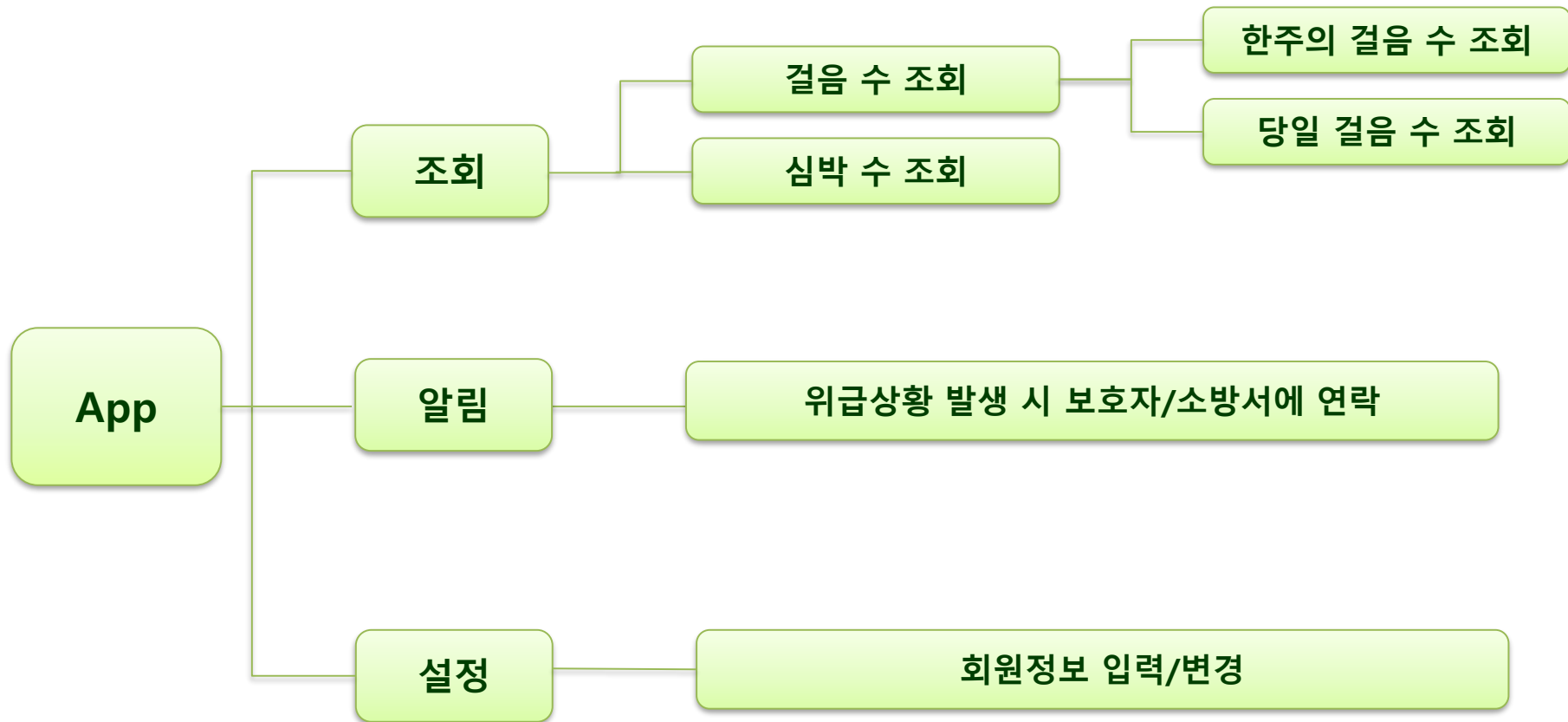
- 가정

- 홀로 생활하는 독거노인과 같은 1인 가구는 Wearable device와 스마트폰을 가지고 있다고 가정한다.
- 독거노인 A씨의 건강 악화로 심박수가 임계치 값을 넘은 것으로 가정한다.

### < 비상연락 기능 >

1. Wearable device의 센서를 통해 A씨의 심박수를 확인한다.
2. 지정해 놓은 심박수의 임계치 값을 넘었거나 저장되어 있는 평소 행동패턴에서 벗어나는지 확인한다.
3. 임계치 값을 넘었거나 행동패턴이 벗어난 경우 소방서/병원을 호출한다.  
앱 회원가입시 저장해 놓은 보호자 연락처 확인 후, 보호자에게 알린다.
4. Wearable device로 측정된 A씨의 심박수와 걸음수를 DB에 저장한다.
5. 휴대폰에 설치된 어플리케이션에 심박수와 걸음수를 시각화하여 A씨가 확인 가능하도록 한다.

### 3. 메뉴 구성도(모바일)



## 4. 프로그램 기능 목록

### SW

#### [회원가입]

: 주소, 생년월일, 성별, 보호자 연락처 등 개인정보 입력

#### [메뉴]

##### - 조회

- 걸음 수 조회
  1. 당일 걸음 수 조회
  2. 걸음 기록 조회 (1주일 동안)
- 심박 수 조회 (3시간동안)

##### - 설정

- 회원 정보
  - 회원 이름
  - 주소
  - 보호자 연락처 재입력
  - 생년월일, 재입력
  - 로그아웃
- 알림

### HW

#### [심박수 측정]

: 심박센서 이용해 1분단위로 심박수 측정

#### [위치 및 각도 측정]

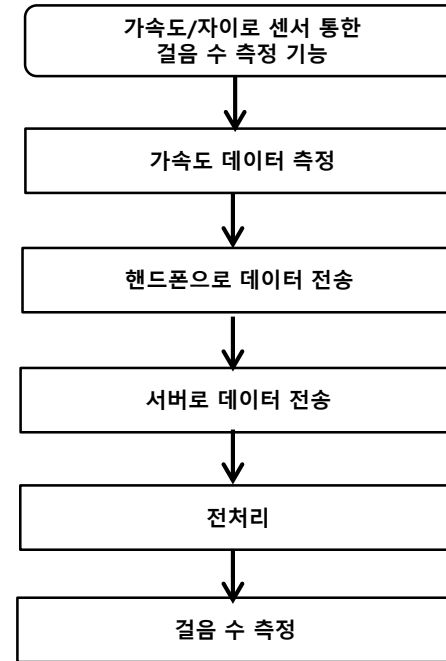
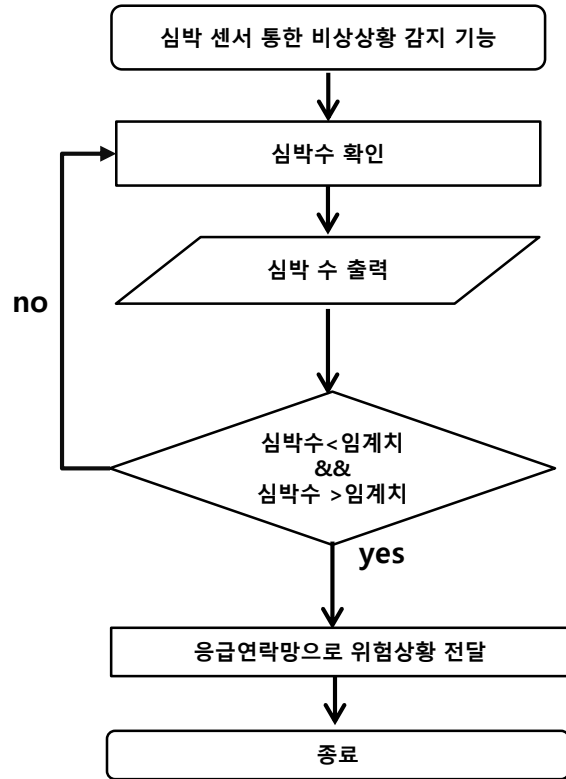
##### - 위치 측정

: 가속도 센서 이용한 사용자의 x,y,z축 위치 정보 측정

##### - 각도 측정

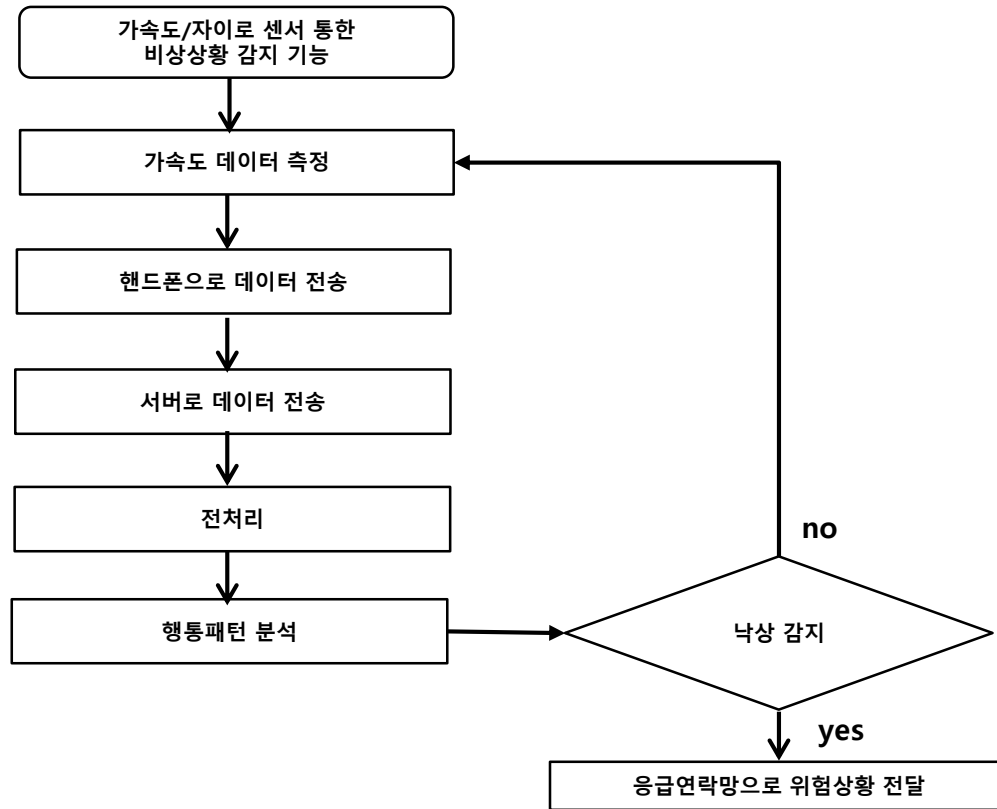
: 자이로 센서 이용한 사용자의 각도 정보 측정

## 5. 기능 흐름도






## 5. 기능 흐름도




## 6. 화면 설계서




### Health Ap

 Username

 ....

[회원가입 / ID PW찾기](#)



기능번호	요구사항번호
기능 명	사용자 - 로그인
기능설명	신규 회원은 회원가입시 입력한 아이디와 비밀번호로 로그인 할 수 있음. 기존 회원은 자동로그인 가능함
처리내용	<ul style="list-style-type: none"><li>✓ 신규 회원의 경우 회원 가입<ul style="list-style-type: none"><li>- 회원가입 시 입력한 아이디와 비밀번호를 입력하여 로그인</li></ul></li><li>✓ 기존 회원의 경우<ul style="list-style-type: none"><li>- 자동 로그인을 가능하게 함</li></ul></li></ul>
비고	✓ Database의 User_info 테이블과 연동
요구사항 명	로그인

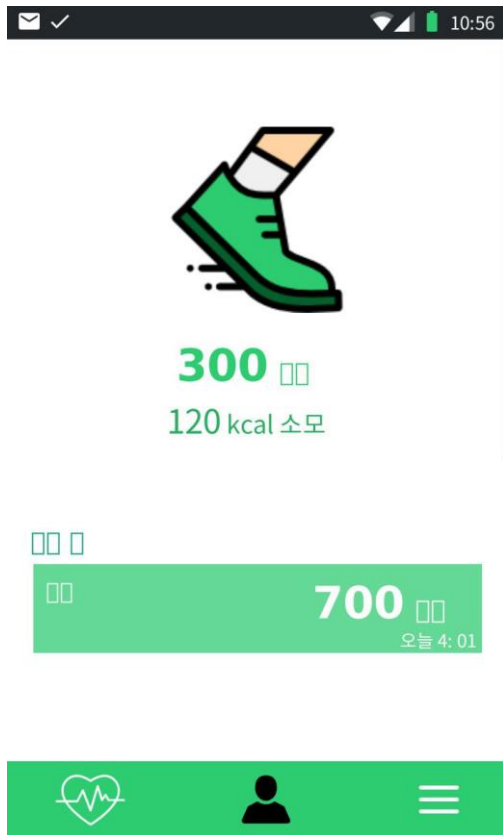
## 6. 화면 설계서



The image shows a mobile application interface for user registration. At the top, there is a status bar with a mail icon, a checkmark, signal strength, battery level, and the time 10:56. Below the status bar is a navigation bar with a back arrow and four empty square icons. The main content area contains five input fields, each with an icon and a label: an email icon for '아이디(이메일)', a lock icon for '비밀번호 (영문 숫자 6~15자 이내)', another lock icon for '비밀번호 확인', a phone icon for '보호자 혹은 병원 전화번호', and a house icon for '나의 집주소'. A red error message '사용가능한 id입니다' is displayed below the email field. At the bottom, there is a large green button with two empty square icons.

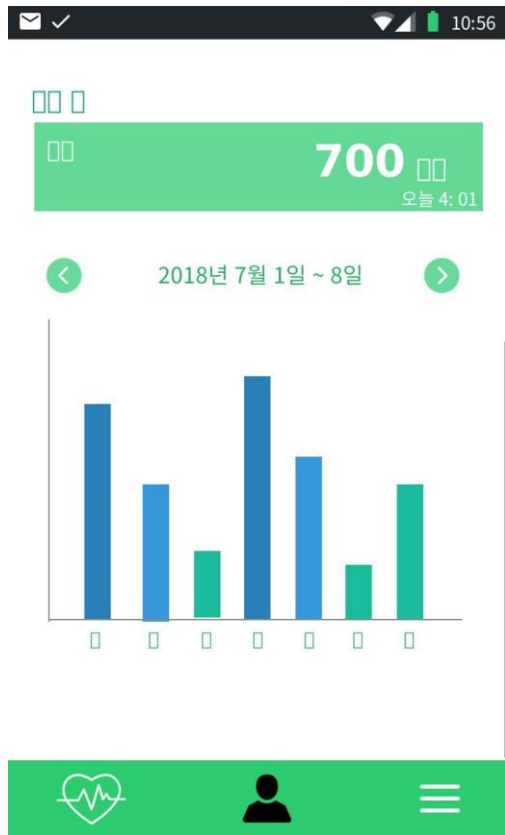
기능번호	요구사항번호
기능 명	사용자 - 회원가입
기능설명	신규 회원은 아이디와 비밀번호, 보호자 연락처, 집주소를 통해 회원가입 가능
처리내용	<ul style="list-style-type: none"> <li>✓ 개인 정보 통한 회원가입</li> <li>- 보호자 연락처, 집주소 입력을 통해 위급상황 발생 시 빠른 대처를 가능하도록 함</li> </ul>
비고	✓ Database의 User_info 테이블과 연동
요구사항 명	회원가입

## 6. 화면 설계서



기능번호	요구사항번호
기능 명	홈 - 걸음 수 확인①
기능설명	로그인 후 첫 화면 걸음 수 확인, 심박 수 확인 2가지 탭 중 하나를 선택할 수 있음
처리내용	<ul style="list-style-type: none"> <li>✓ 일일 걸음 수 조회 <ul style="list-style-type: none"> <li>- 하루 동안 움직인 걸음 수를 조회</li> <li>- 10걸음 이동시 어플리케이션에서 걸음 수 동기화</li> </ul> </li> <li>✓ 일일 소모 칼로리 조회 <ul style="list-style-type: none"> <li>- 하루 동안 소모한 칼로리 조회</li> <li>- 일일 걸음 수와 관련하여 소모 칼로리 량 계산</li> </ul> </li> </ul>
비고	✓ Database의 User_after 테이블과 연동
요구사항 명	일일 걸음 수 및 소모칼로리 조회

## 6. 화면 설계서



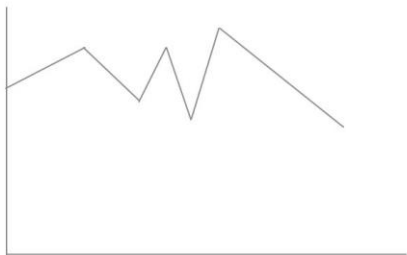
기능번호	요구사항번호
기능 명	홈 - 걸음 수 확인②
기능설명	로그인 후 첫 화면 걸음 수 확인, 심박 수 확인 2가지 탭 중 하나를 선택할 수 있음
처리내용	<ul style="list-style-type: none"> <li>✓ 일주일 동안의 걸음 수 시각화</li> <li>- 1주 동안 요일마다 걸음 수를 막대그래프로 시각화</li> <li>- 사용자의 운동량 파악을 통해 적정 운동량 권장</li> </ul>
비고	✓ Database의 User_after 테이블과 연동
요구사항 명	1주동안의 요일마다 걸음 수 시각화

## 6. 화면 설계서



오늘 4:01

□□ □□□



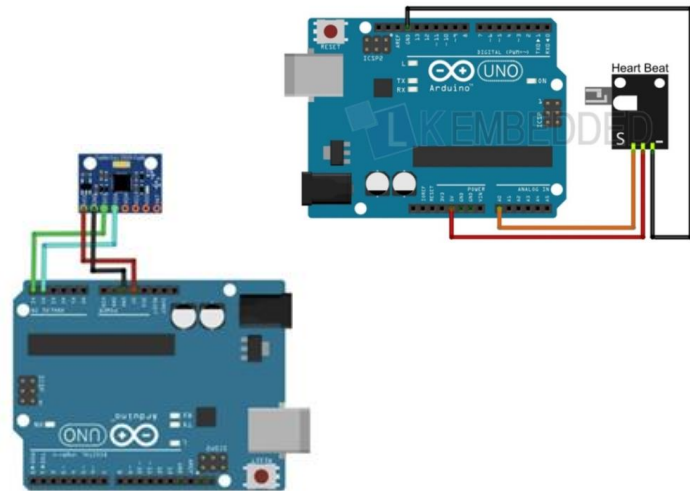
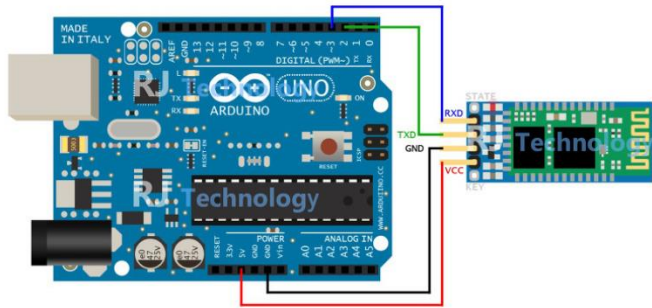
기능번호		요구사항번호	
기능 명	홈 - 심박수 조회		
기능설명	로그인 후 첫 화면 걸음 수 확인, 심박 수 확인 2가지 탭 중 하나를 선택할 수 있음		
처리내용	<ul style="list-style-type: none"> <li>✓ 현재 심박 수 조회 <ul style="list-style-type: none"> <li>- 현재 사용자의 심박 수 조회</li> <li>- 1분마다 심박 수 동기화</li> </ul> </li> <li>✓ 3시간 동안의 심박 수 시각화 <ul style="list-style-type: none"> <li>- 연속적으로 변하는 데이터를 꺾은선 그래프로 시각화</li> <li>- 사용자의 심박 수 변화의 인지를 통해 건강상태 확인 가능</li> </ul> </li> </ul>		
비고	✓ Database의 User_after 테이블과 연동		
요구사항 명	현재 심박수와 3시간동안의 심박 그래프		

## 6. 화면 설계서



기능번호	요구사항번호
기능 명	설정 - 회원정보
기능설명	회원정보에서 로그아웃을 통해 로그인 화면으로 돌아갈 수 있음
처리내용	<ul style="list-style-type: none"> <li>✓ 로그아웃 <ul style="list-style-type: none"> <li>- 로그아웃의 여부를 '로그아웃'버튼과 '취소'버튼을 통해 선택하는 대화상자가 뜸</li> <li>- '로그아웃'버튼의 경우 로그인 화면(첫 화면)으로 되돌아감</li> <li>- '취소'버튼의 경우 대화상자가 사라지며 다시 회원정보 화면으로 되돌아감</li> </ul> </li> <li>✓ 사용자 친화 UI <ul style="list-style-type: none"> <li>- 화면 상단의 '&lt;' 버튼 누르면 '설정' 화면으로 되돌아감</li> </ul> </li> </ul>
비고	✓ Database의 User_info 테이블과 연동
요구사항 명	로그아웃

## 7. 모듈 설계서



센서 종류	연결 핀	설명
HC-06	GND	아두이노의 GND에 연결
	TXD	아두이노의 2번핀에 연결
	RXD	아두이노의 3번핀에 연결
	VCC	아두이노의 5V에 연결
Pulse Sensor	S	아두이노의 A0에 연결
	+	아두이노의 5V에 연결
	-	아두이노의 GND에 연결
MPU-6050	VCC	아두이노의 5V에 연결
	GND	아두이노의 GND에 연결
	SCL	아두이노의 A5에 연결
	SDA	아두이노의 A4에 연결
	INT	아두이노의 A2에 연결



## 8. 테이블 설계서

프로그램 ID	PHC001	프로그램 명	Physical health care	작성일	2018.07.23	Page	1/2
---------	--------	--------	----------------------	-----	------------	------	-----

### <User\_Info >

항목명	Type	필수/선택	값 목록	활성여부	설명
member_id	primary_varchar(40)	필수		활성	▪ 최대 40자 이내, 숫자/문자/특수문자 혼용
member_protect_phone	varchar(20)	선택		활성	▪ 최대 20자 이내, -없이 숫자만 입력
user_info_date	timestamp	필수		활성	▪ 데이터가 입력된 시간을 표시
member_address	varchar(120)	필수		활성	▪ 최대 120자 이내

### <User\_Health >

항목명	Type	필수/선택	값 목록	활성여부	설명
user_hid	varchar(40)	필수		활성	▪ 최대 40자 이내, 숫자/문자/특수문자 혼용
user_heart	int(5)	필수		활성	▪ 최대 5자 이내
user_walk_x	double(8)	필수		비활성	▪ 소수점 포함 최대 8자 이내
user_walk_y	double(8)	필수		비활성	▪ 소수점 포함 최대 8자 이내
user_walk_z	double(8)	필수		비활성	▪ 소수점 포함 최대 8자 이내
user_move_angle	double(8)	필수		비활성	▪ 소수점 포함 최대 8자 이내
user_health_date	timestamp	필수		활성	▪ 데이터가 입력된 시간을 표시

## 8. 테이블 설계서

프로그램 ID	PHC002	프로그램 명	Physical health care	작성일	2018.07.23	Page	2/2
---------	--------	--------	----------------------	-----	------------	------	-----

<User\_After >

항목명	Type	필수/선택	값 목록	활성여부	설명
user_aid	primary_varchar(40)	필수		활성	▪ 최대 40자 이내, 숫자/문자/특수문자 혼용
walk_today	int(6)	필수		활성	▪ 7일 동안의 값 저장
alarm	int(3)	필수		비활성	▪ 초기값을 0으로 설정
heart_image	BLOB	필수		활성	-
walk_image	BLOB	필수		활성	-

## 9. 프로그램 상세 로직

### 개요

심장 박동 수 출력을 위한 코드

### 상세 로직

```
void setup(){
  pinMode(blinkPin,OUTPUT);    // 심장 박동에 깜빡임
  pinMode(fadePin,OUTPUT);     // 심장 박동에 맞춰 fade
  Serial.begin(115200);
  interruptSetup();            // sets up to read Pulse Sensor signal every 2mS
}

void loop(){
  serialOutput() ;
  if (QS == true){    // QS가 true라면 즉, 아두이노가 심박수를 감지하면 BPM과 IBI 결정
    fadeRate = 255;    // LED fade 효과 발생
                     // fadeRate 변수를 255로 설정하여 펄스로 LED를 fade
    serialOutputWhenBeatHappens(); // 비트 발생시 출력
    QS = false; }      // 다음 심박수를 위하여 다시 reset
  ledFadeToBeat();
  delay(1000);          // 측정속도 딜레이 시간을 1000으로 설정
}

void ledFadeToBeat(){
  fadeRate -= 15;        // LED fade value 설정
  fadeRate = constrain(fadeRate,0,255); // LED fade value가 음수값이 되지 않도록 함
  analogWrite(fadePin,fadeRate);    // fade LED
}
```

## 9. 프로그램 상세 로직

### 개요

심박 수 이상 감지 시 응급 상황 대처

### 상세 로직

```
private String PulseSensor(double min_temp, double max_temp, double now_temp,
                           String houseTemp, String housename)
{
    int Pulse = null;

    if (now_pulse < min_pulse &&now_pulse > max_pulse ) // 심박 수가 임계값을 벗어날 경우
    {
        //설정된 비상 응급망 및 보호자에게 전화
        mNum = mEditNumber.getText().toString();

        String tel = "tel:" + mNum;

        switch (v.getId()){
            case R.id.btnCall:
                startActivity(new Intent("android.intent.action.CALL", Uri.parse(tel)));
                break;

            case R.id.btnDialog:
                startActivity(new Intent("android.intent.action.DIAL", Uri.parse(tel)));
                break;
        }
    }
}
```

## 9. 프로그램 상세 로직

### 개요

자이로/가속도 센서를 이용한 행동패턴 분석을 위한 데이터 추출

### 상세 로직

```
void loop()
{
    get6050();//센서값 갱신
    //받아온 센서값을 출력합니다.
    Serial.print(AcX);
    Serial.print("");
    Serial.print(AcY);
    Serial.print("");
    Serial.print(AcZ);
    Serial.println();
    delay(15); }

void get6050(){
    Wire.beginTransmission(MPU);//MPU6050 호출
    Wire.write(0x3B);//AcX 레지스터 위치 요청
    Wire.endTransmission(false);
    Wire.requestFrom(MPU,14,true);//14byte의 데이터를 요청

    AcX=Wire.read()<<8|Wire.read();//두개의 나뉘어진 바이트를 하나로 이어붙입니다.
    AcY=Wire.read()<<8|Wire.read();
    AcZ=Wire.read()<<8|Wire.read();
    Tmp=Wire.read()<<8|Wire.read();
    GyX=Wire.read()<<8|Wire.read();
    GyY=Wire.read()<<8|Wire.read();
    GyZ=Wire.read()<<8|Wire.read(); }
```

## 9. 프로그램 상세 로직

### 개요

자이로/가속도 센서를 이용한 사용자의 걸음 수 측정

### 상세 로직

```
void loop()
{
    int ax, ay, az, gx, gy, gz;

    ax = GetData(ACCEL_XOUT_H);
    ay = GetData(ACCEL_YOUT_H);
    az = GetData(ACCEL_ZOUT_H);
    gx = GetData(GYRO_XOUT_H);
    gy = GetData(GYRO_YOUT_H);
    gz = GetData(GYRO_ZOUT_H);
    ax = ax / 1000;
    ay = ay / 1000;
    az = az / 1000;
    int val = ax * ax + ay * ay + az * az;

    if ( ( val > 500 ) && ( prev_val <= 500 ) )
    {
        // 0.3초 미만 간격으로 걸음이 추가되는 경우는 실제 걸음이 아니라 센서의 흔들림 및 노이즈일 가능성이 많음.
        if ( ( millis() - last_millis ) > 300 )
        {
            count++;
            last_millis = millis();
        }
    }
    prev_val = val;
    Serial.println(count/2);}
```

## 10. 개발 환경

구분		항목	적용 내역
SW 개발환경	OS	Windows 10	- 현 OS에서 작업
	개발환경 (IDE)	Android Studio3.0 아두이노 스케치 Pycharm	-안드로이드 어플리케이션 개발 도구 사용 -아두이노 코드 작성 및 개발 도구 사용 -데이터 마이닝을 통한 행동패턴 분석 도구
	개발도구	AWS EC2 phpmyadmin	-안드로이드 코드 작성 및 AWS 서버를 작성하기 유용한 개발도구 사용 -생체정보와 회원정보 저장을 위한 DataBase
	개발언어	PHP JAVA Python	-안드로이드와 서버의 연동을 위해 PHP 언어 사용 -안드로이드 어플리케이션을 만들기 위한 코드를 JAVA 기반으로 선택하여 작성 -Python을 기반으로 데이터 마이닝을 위한 코드 작성

## 10. 개발 환경

구분		항목	적용 내역
HW 개발환경	디바이스	안드로이드 기종(태블릿) Arduino nano 호환보드(R3)	<ul style="list-style-type: none"> <li>- 안드로이드 어플리케이션을 구동 및 실험</li> <li>- 각종 센서 및 장치들을 보드에 연결하여 데이터 수신</li> </ul>
	센서	심박센서 가속도/자이로 센서	<ul style="list-style-type: none"> <li>- 심박센서의 값을 읽어와 심박 수 출력</li> <li>- 심박 센서의 값을 읽어와 임계값을 벗어날 경우 응급 상황 전달</li> <li>- 가속도/자이로 센서의 값을 읽어와 사용자의 움직임을 감지하고 낙상 검출</li> <li>- 가속도/자이로 센서의 값을 읽어와 사용자의 걸음 수 출력</li> </ul>
	통신	블루투스 모듈 웹 서버(AWS)	<ul style="list-style-type: none"> <li>- 블루투스 모듈을 이용하여 아두이노와 안드로이드 통신</li> <li>- 아두이노에서 측정한 데이터를 핸드폰을 통해 웹 서버에 전송하여 데이터 저장 및 분석</li> </ul>
	개발언어	C/C++	<ul style="list-style-type: none"> <li>- 보드 및 개발 툴이 C언어 기반이므로 C언어로 작성</li> </ul>