

TEMS App

# Overview

- 방사능 측정 기기에 온도와 습도 측정 센서를 추가
- Esp32 칩을 사용해서 HTTP/HTTPS 프로토콜을 이용해서 네트워크를 통해 측정한 데이터를 서버로 전송 및 저장
- 블루투스나 스마트폰 이어폰 단자를 이용하던 기존의 모델과는 전혀 다른 모델로 어플리케이션 서버를 활용해서 더 다양한 서비스를 제공

# Welcome screen

처음 앱을 실행시켰을 때 나오는 화면.

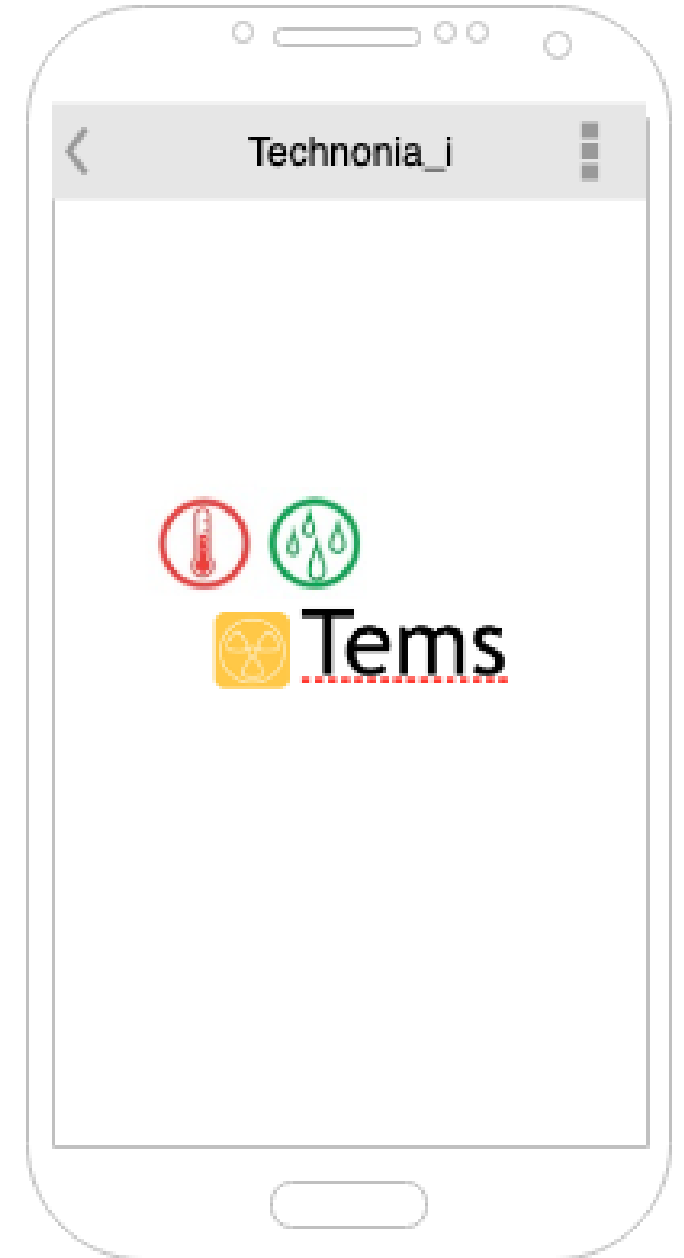
앱을 실행시켰을때 로딩 화면으로 나오게 된다.

로딩이 끝나면 로그인 화면 (Log in Screen) 혹은  
프로필 화면(Profile Screen)으로 이동하게 된다.

자동 로그인의 경우 프로필 화면으로 이동하게  
되고, 그렇지 않으면 로그인 화면으로 이동.

자동 로그인은 아이디 등을 로컬 스토리지에  
저장하고 확인하는 과정을 통해서 구현할 예정.

로딩 화면이기 때문에 뒤로 가기 버튼을 누르게  
되면 앱이 종료된다.



# Navigate from Welcome screen to ...

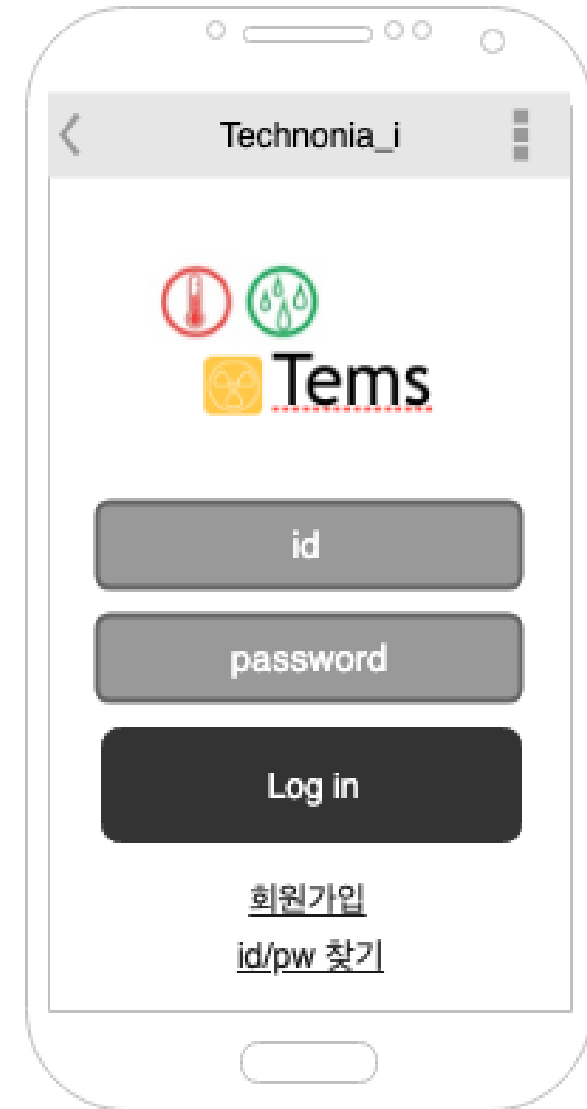


# Log in Screen

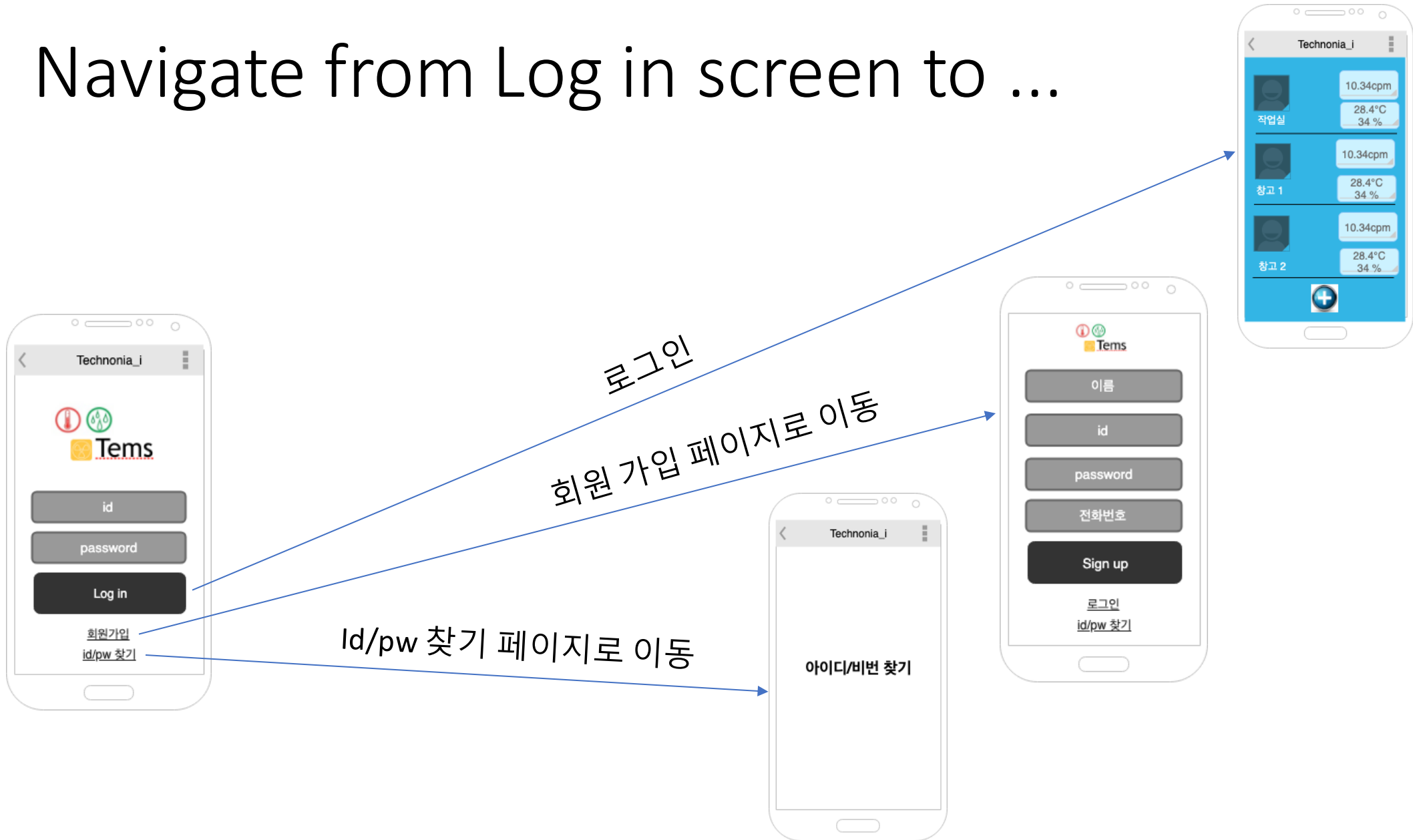
로그인 화면에서는 로그인을 할 수 있다.  
또한, 회원가입 화면이나  
아이디/비밀번호 찾기 화면으로의  
이동이 가능하다.

아이디 입력을 위한 텍스트 박스 위에는  
로고를 놔두는 것이 좋을 듯하다.

이것이 첫 화면이기 때문에, 뒤로 가기  
버튼을 누르면 앱이 종료될 것이다.



# Navigate from Log in screen to ...



# Server communication – Log in

- POST 요청을 보내서 로그인
- JSON 포맷 (사용자의 아이디와 비밀번호 등을 포함)
- HTTP/HTTPS를 사용해서 통신
- 서버에서 "OK"와 같은 답을 보내면 프로필 화면으로 이동
- 로그인 시 자동 로그인을 할지에 대한 여부를 묻기
- 자동 로그인의 경우 로컬 스토리지를 이용해서 사용자 정보를 저장
- 로그인 실패 시 알림 (팝업 메시지 등) - react native alert 기능을 이용

# Sign up Screen

회원 가입을 위한 화면.  
사용자의 이름, 전화번호와 사용할  
아이디 그리고 비밀번호를 입력해야  
한다.

회원 가입에 성공하면 로그인 페이지로  
돌아가게 된다.  
하단의 로그인 링크나 뒤로 가기를  
눌러도 로그인 화면으로 돌아가게 된다.

Id/pw 찾기 링크를 누르면  
아이디/비밀번호 찾기 화면으로  
이동하게 된다.

뒤로 가기 버튼을 누르면 로그인  
화면으로 돌아가게 된다.

Tems

이름

id

password

전화번호

Sign up

[로그인](#)

[id/pw 찾기](#)



# Navigate from Sign up screen to ...



# Server communication – Sign up

- JSON으로 통신
- POST 요청을 보내며, HTTP/HTTPS 프로토콜을 사용
- 입력한 사용자의 정보를 서버로 전송
- 사용자의 이름, 전화번호, 그리고 사용할 아이디와 비밀번호를 모두 입력해야 회원 가입 요청 가능 (전부 입력하지 않으면 alert)
- 회원 가입 완료라는 답이 서버로부터 오면 로그인 화면으로 이동
- 회원 가입 실패 혹은 에러 메시지가 서버로부터 오면 적절한 팝업 메시지를 띄우고 현재 화면에 계속 머문다

# ID/PW Finding Screen

아이디나 비밀번호를 찾는 서비스를 제공하는 화면.

로그인 화면이나 회원가입 화면에서 접근할 수 있으며, 뒤로 가기 버튼을 통해서 다시 이전 화면으로 돌아갈 수 있다.

이전에 아이디/비밀번호 찾기를 이용해 본 적이 없어서 어떤 식으로 디자인을 해야 할 지 모르겠어서 일단 우측과 같이 놔두었다. 후에 자세한 디자인이 필요할 것이다.



# Navigate from Finding Screen to ....

- 뒤로 가기 버튼을 통해서 이전 화면 (로그인 화면 혹은 회원 가입 화면)으로 돌아갈 수 있다
- 아이디/비밀번호 찾기 서비스를 완료하면 로그인 화면으로 이동

# Server Communication – Finding ID/PW

- 사용자가 등록한 전화번호를 사용한 인증
- 서버에서 랜덤하게 만든 문자열을 문자로 전송 후, 앱에 JSON 포맷으로 그 문자열을 전송. 만약 일정 시간 (대략 3분) 안에 주어진 문자열을 입력하면 서버와의 통신을 통해 기존 아이디/비밀번호를 앱으로 전송해주는 방식.
- 비밀번호의 경우, 임시 비밀번호를 발급해주는 통상적인 방법을 고려해 보는 것도 좋을 듯하다. 이 경우, 서버에 POST 요청을 보내 비밀번호를 업데이트해야 할 것이다.
- 아이디/비밀번호 찾기가 완료되면 통신을 완료하고 로그인 페이지로 이동하게 된다.

# Profile Screen

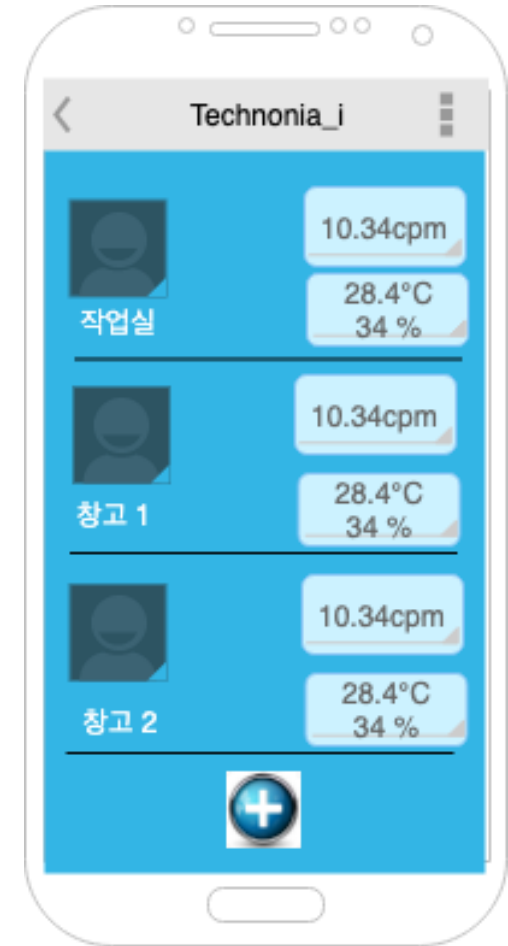
사용자가 등록한 기기들의 목록을 표시하는 화면.

기기 목록 아래의 + 버튼을 통해서 새로운 기기를 등록할 수 있는 화면으로 이동이 가능하다.

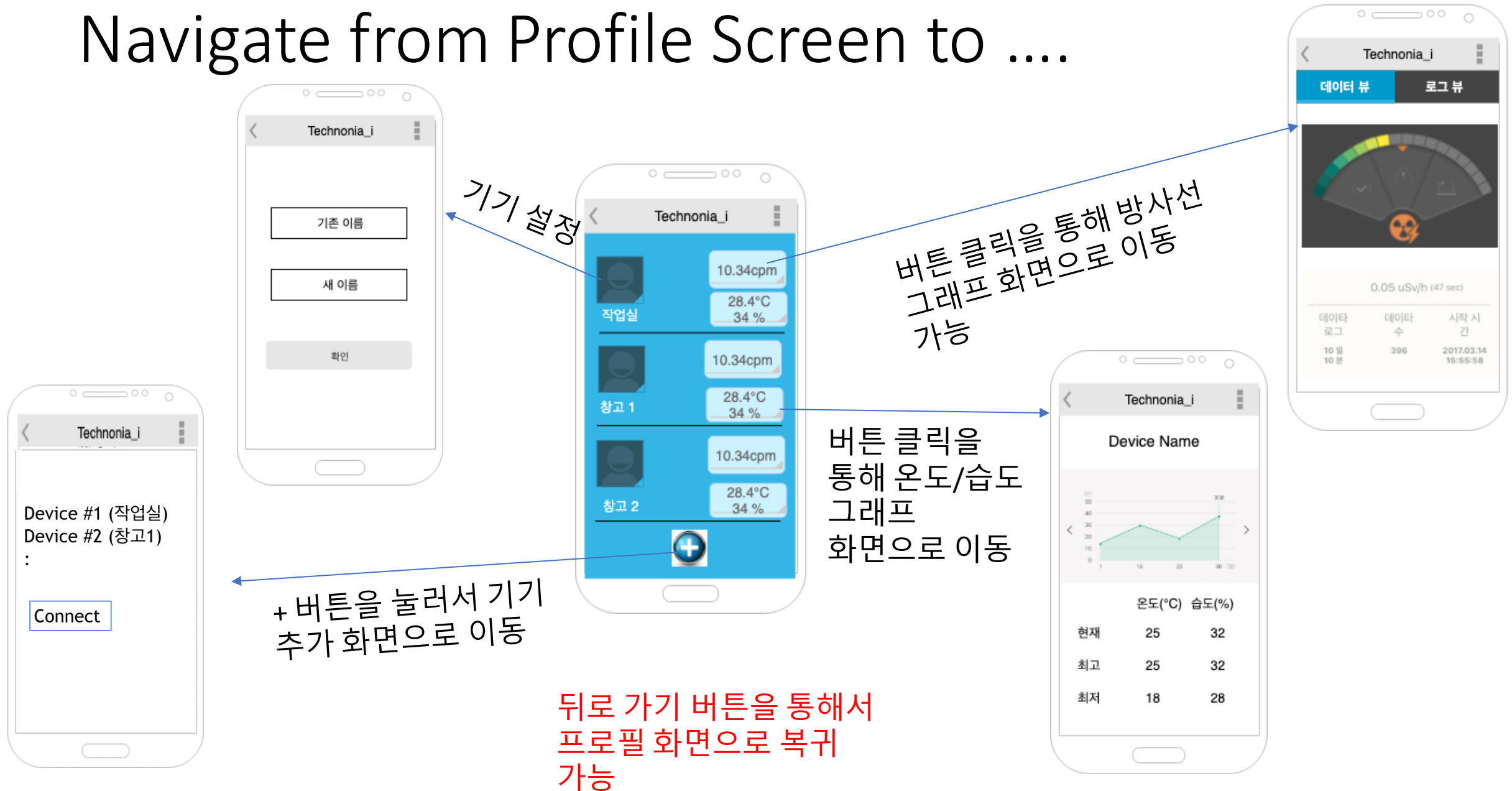
각 목록에서 좌측에는 적절한 이미지와 등록된 기기의 이름이, 우측에는 방사능 수치와 온도/습도를 나타내는 버튼들이 위치해 있다. 우측의 버튼들 중 위의 것을 누르면 방사능 수치 확인 화면으로 이동하며, 하단의 버튼을 누르게 되면 온도/습도 그래프 화면으로 이동하게 된다.

또한, 좌측의 이미지를 누르게 되면 기기 설정 관리 화면으로 이동하게 된다.

로그인이 된 상태에서 첫 화면이기 때문에, 뒤로 가기 버튼을 누르면 앱이 종료된다.



# Navigate from Profile Screen to ....



# Server Communication – Profile Screen

- 프로필 화면이 실행되기 전, 서버로 GET 요청을 보내 JSON 포맷의 데이터를 전송 받는다. 물론, 프로토콜은 HTTP/HTTPS
- 서버가 보낸 JSON 안에는 사용자가 등록한 기기의 목록과 각 기기가 측정한 현재 온도, 습도 그리고 방사능 수치가 있다.
- 기기의 상태 및 측정 값 등을 업데이트하기 위해서 일정 시간마다 요청을 보내야 한다.



# Device Registration

기기 등록을 위한 화면이다.

새로운 기기를 추가할 수 있게 해준다.  
기기 등록 방식은 아직 미정.

아마도 새 기기의 고유 번호 등을  
추가하는 서비스를 제공하게 될  
듯하다.

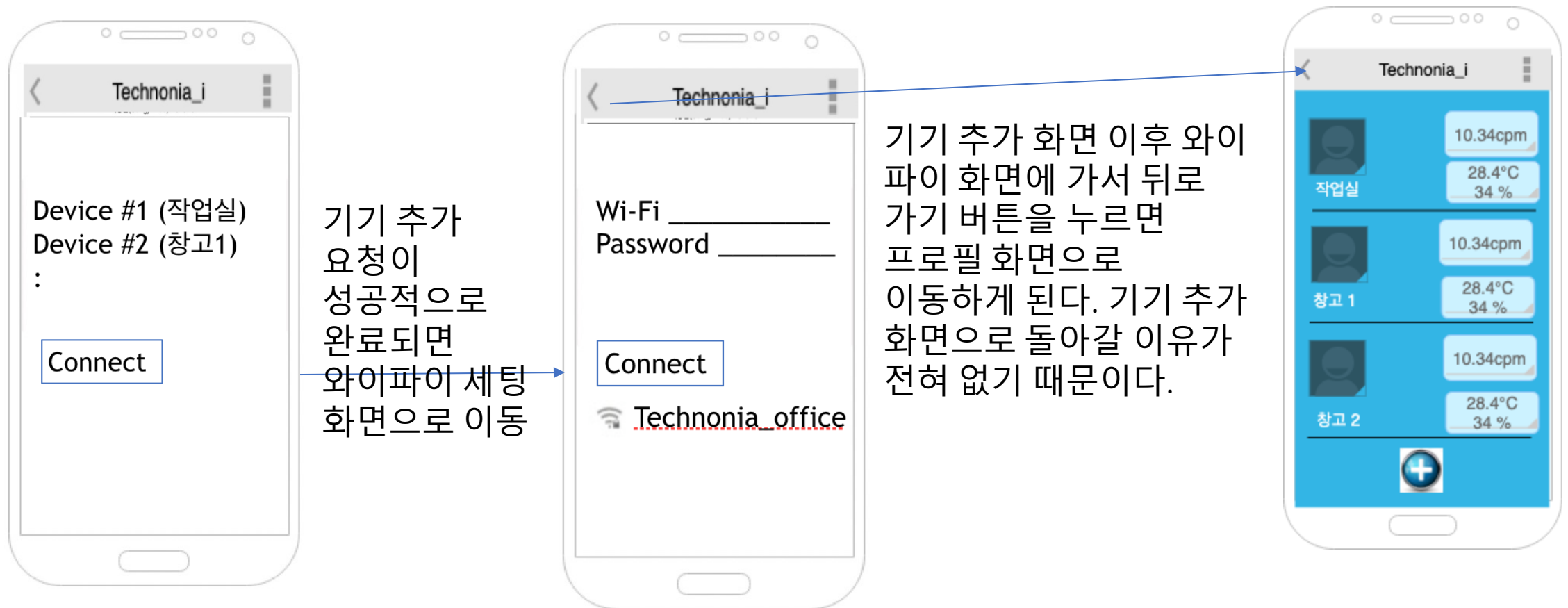
기기 등록에 성공하게 되면 프로필  
화면으로 이동하게 되며, 이때 새로  
등록한 기기가 목록에 추가될 것이다.



# Navigation – Device Registration Screen

- 뒤로 가기 버튼을 통해서 프로필 화면으로 이동 가능
- 기기 추가에 성공해도 프로필 화면으로 이동 가능
- 기기 추가를 완료하게 되면, 와이파이 세팅 화면으로 이동해서 기기에 와이파이를 연결한다.
- 기기 추가가 완료되면 자동적으로 프로필 화면의 목록에 새 기기가 추가되게 된다.
- 와이파이 세팅 화면에서 프로필 화면으로의 이동은 이후 와이파이 세팅 화면에 대한 설명을 포함하는 슬라이드에 추가하였다.
- 추가적인 내용은 다음 슬라이드에서 계속된다.

# Navigate – Device Registration Screen



# Server Communication – Device Registration

- JSON 포맷으로 데이터 전송
- HTTP/HTTPS POST 요청을 사용
- JSON에는 새 기기의 고유 번호(기기 번호)와 사용자의 아이디가 포함되어 있어야 한다.
- 서버가 등록 처리를 완료하면 새로운 기기 목록으로 프로필 화면을 렌더링
- 서버 내의 에러나 잘못된 기기 번호 등의 문제의 경우 사용자에게 알림

# Device Screen

기기 설정을 위한 화면.

기기 이름 변경이나 기기의 와이파이 재연결 및 재설정 그리고 기기 삭제 등을 할 수 있다.

기기 이름 변경이나 와이파이 재설정 버튼을 누르게 되면 해당 화면으로 이동하게 되지만, 기기 삭제 버튼을 누르게 되면 프로필 화면으로 돌아가게 된다. 이때, 프로필 화면의 목록에서 해당 기기가 사라지게 된다.

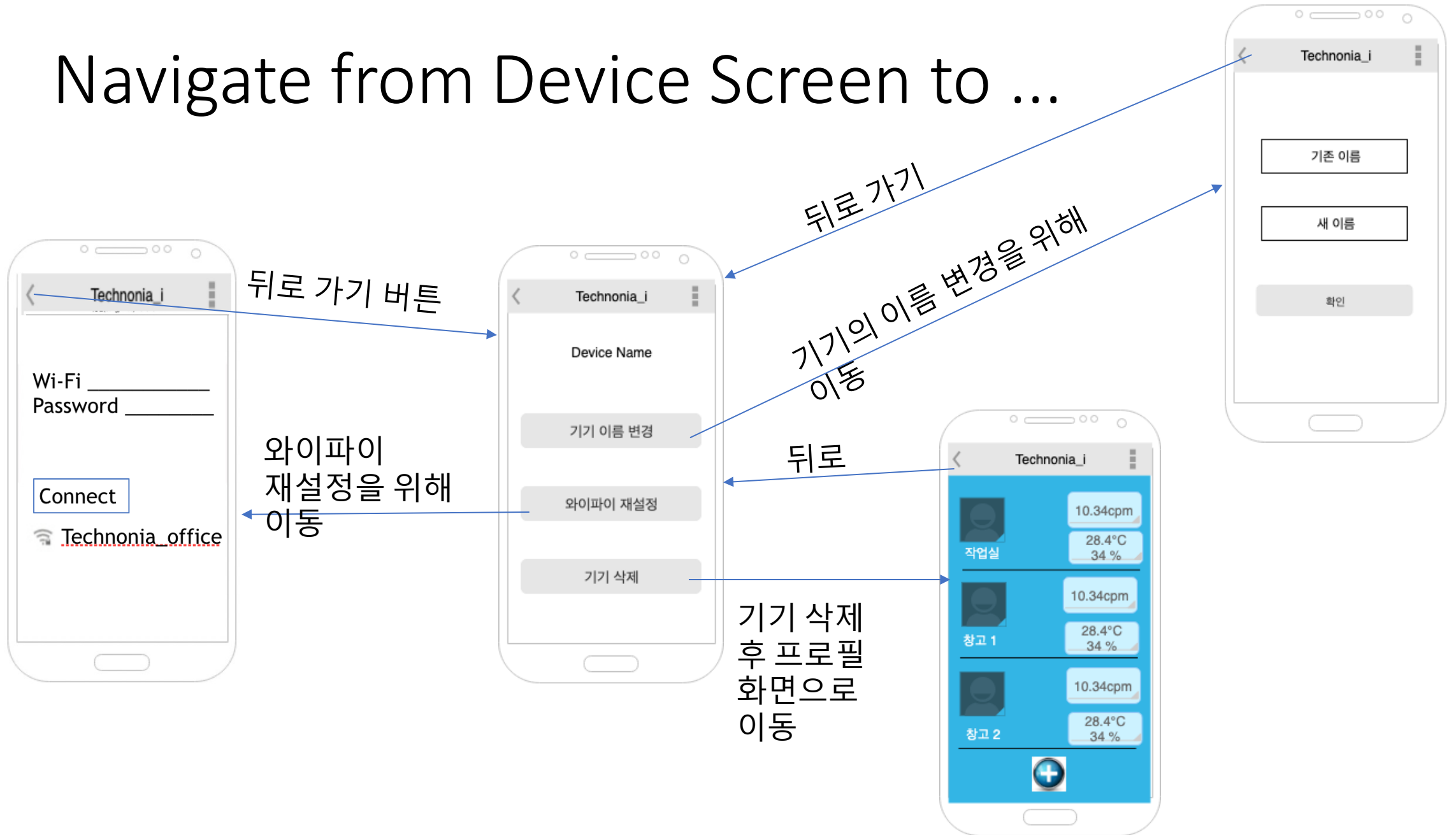
이후에 기기 설정 관련 기능을 추가하게 되면 이 화면에 새 버튼을 추가하는 것으로 쉽게 앱의 확장이 가능할 것이다.



# Removing Device

- 기기 삭제를 진행하기전, 계속 진행할 것인지에 대한 경고 메시지를 먼저 띄운다.
- 사용자가 기기 삭제를 진행하고자 한다면 HTTP/HTTPS POST 요청을 이용해서 기기 삭제에 대한 요청을 서버로 보낸다.
- 서버로 JSON 포맷의 데이터를 보내며, JSON 내에 삭제할 기기의 고유 번호와 사용자의 아이디를 포함시켜야 한다.
- 서버는 해당 기기를 데이터베이스에서 찾아서 삭제하고, 적절한 결과 문자열을 사용자에게 보낸다.
- 서버로부터 기기 삭제 성공 메시지를 받으면 프로필 화면으로 이동한다. 이때, 프로필 화면의 기기 목록에는 해당 기기가 사라져 있어야 한다.

# Navigate from Device Screen to ...



# Changing Device Name

등록된 기기의 이름을 변경할 수 있게 도와주는 화면.

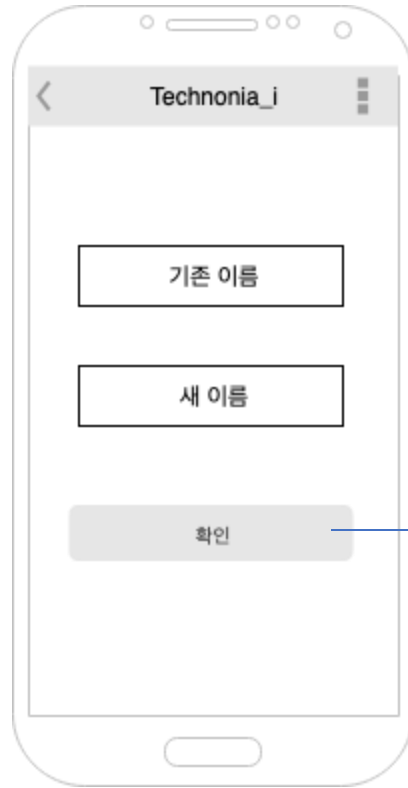
기기의 새로운 이름을 입력한 뒤 확인 버튼을 누르면 서버에 POST 요청을 보냄으로써 기기의 이름을 업데이트하게 된다.

기기 이름을 바꾼 뒤, 프로필 화면으로 돌아간다. 이때, 해당 기기의 이름이 업데이트 되어 있어야 한다.



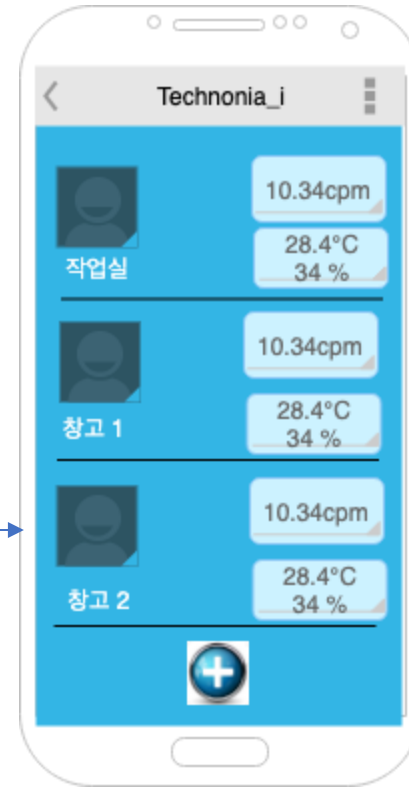


# Navigate from Device Name Screen to ...



이름 변경 완료 시 프로필 화면으로 이동.

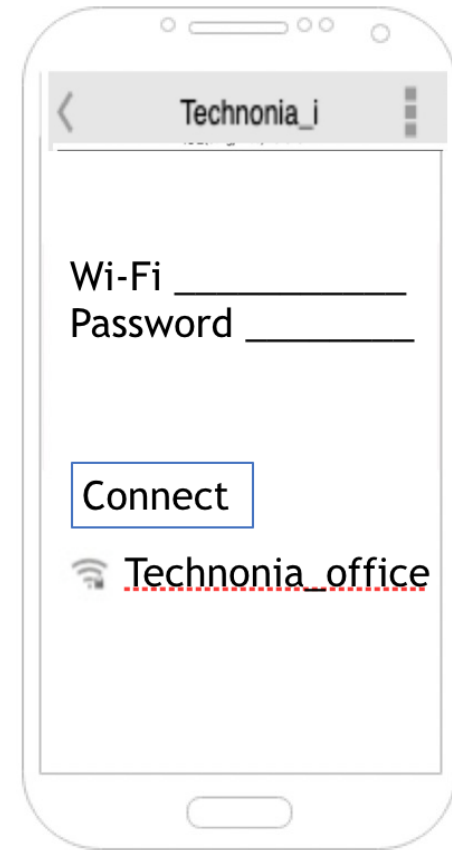
이때, 기기 목록에 해당 기기의 이름이 새 이름으로 업데이트 되어 있어야 한다.



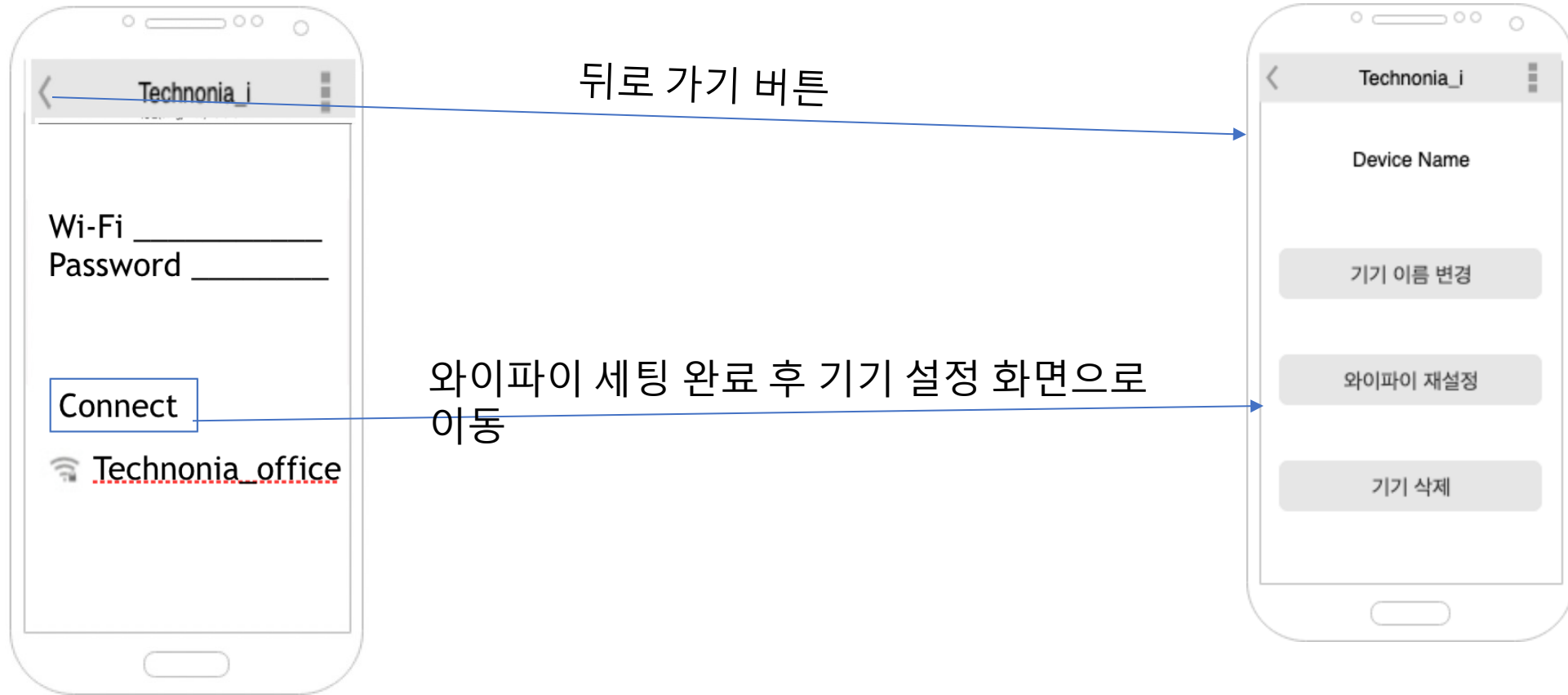
# Wi-Fi Setting Screen

Geiger 기기에 와이파이 등록 혹은 다른 와이파이 사용 등을 위한 설정 서비스를 제공하는 화면이다.

앱 화면에서 사용하려는 와이파이의 이름과 비밀번호를 입력하고 connect 버튼을 누르면 앱이 geiger 기기와 블루투스 통신을 통해 기기가 해당 와이파이를 사용할 수 있게 된다.



# Navigate from Wi-Fi Screen to ...



# Temperature & Humidity Graph

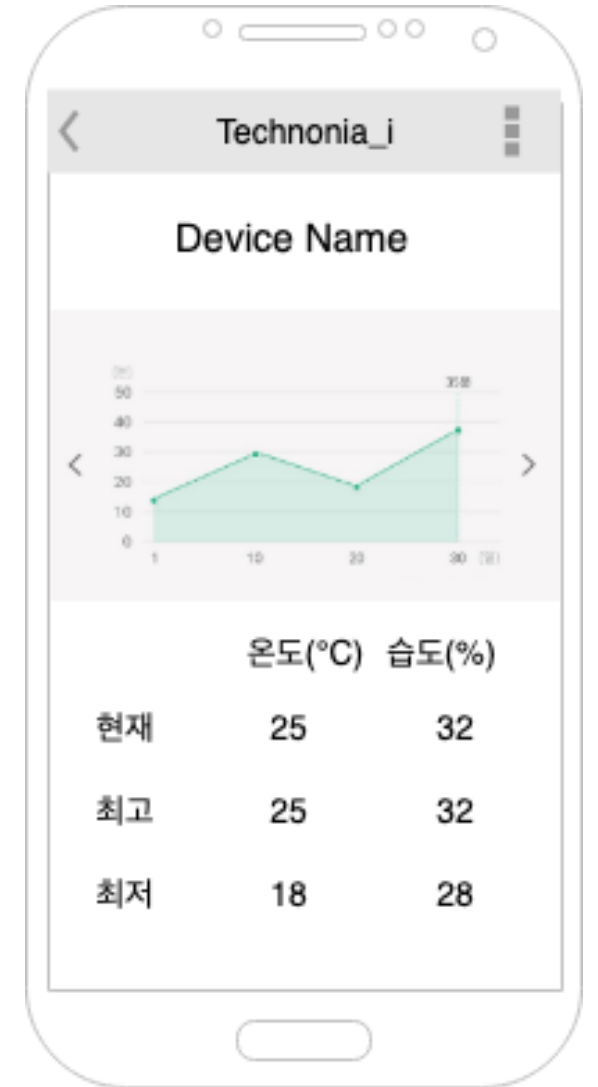
온도&습도 그래프를 그리는 화면.

사용자에게 온도 변화에 대한 그래프뿐만 아니라 현재 기온, 최고 기온과 최저 기온, 현재 습도와 최고 그리고 최저 습도를 제공한다.

현재 이미지 상으로는 단일 그래프이지만, 실제로는 온도 그래프와 습도 그래프를 동시에 그릴 예정이다.

추가로, 사용자가 그래프를 그리는 데이터의 범위를 직접 정할 수 있게 하는 버튼을 추가할 예정이다.

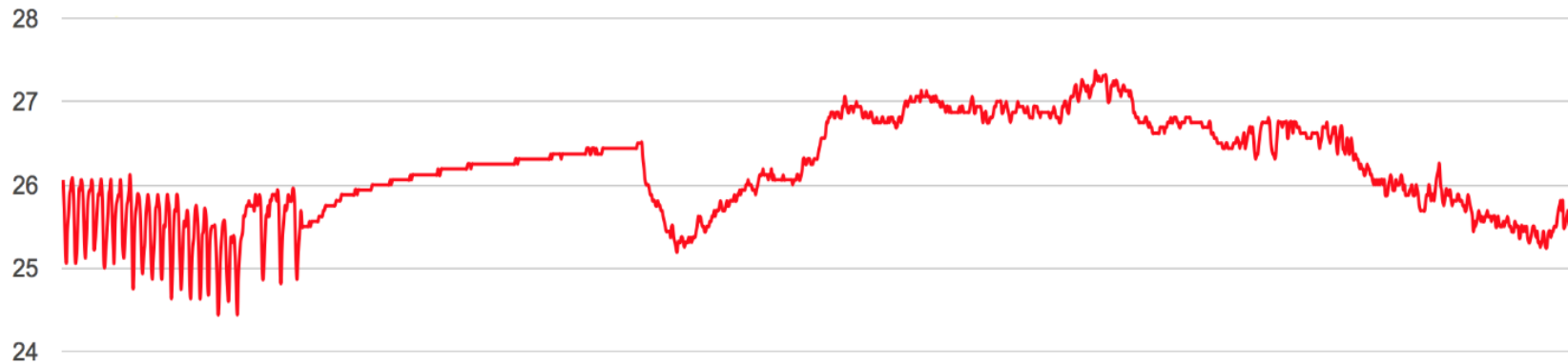
예를 들어, 3 개의 버튼이 있다고 하자. 각 버튼은 1일, 3일 그리고 1주일간의 데이터로 그래프를 그리도록 하는 역할을 한다고 하자. 그렇게 되면, 사용자는 각각의 버튼을 누름으로써 손쉽게 원하는 기간동안의 데이터 변화를 볼 수 있게 될 것이다.



# Graph range

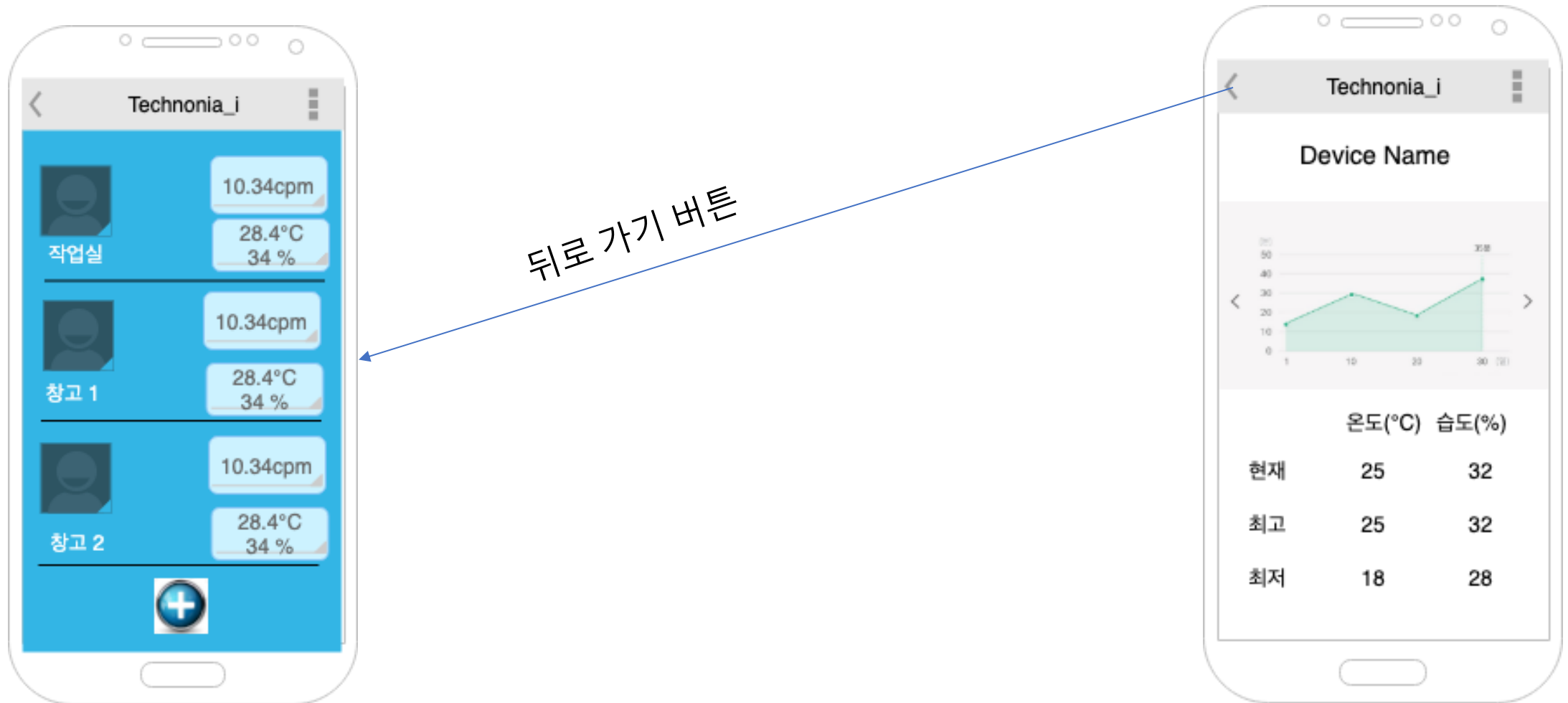
5시간 12시간 1일 1주일 1달 사용자 지정

temperature graph



바로 이전 슬라이드에서 언급했었던 그래프 기간을 위한 버튼은 대략 왼쪽의 이미지와 같은 형태가 될 것이다.

# Navigate from Graph Screen to ...



# Server Communication – Graph Screen

- HTTP/HTTPS GET 요청을 이용해서 온도와 습도 데이터를 포함한 JSON 포맷의 데이터를 서버로부터 전송받는다.
- 이 요청을 위해서 url 쿼리에 사용자의 아이디와 기기 고유 번호를 넣는다.
- 서버는 JSON 객체에 해당 기기가 지난 7일 (최대) 동안 수집한 데이터들을 온도와 습도 그리고 측정한 시간(혹은 날짜)로 나누어서 넣는다.

i.e.

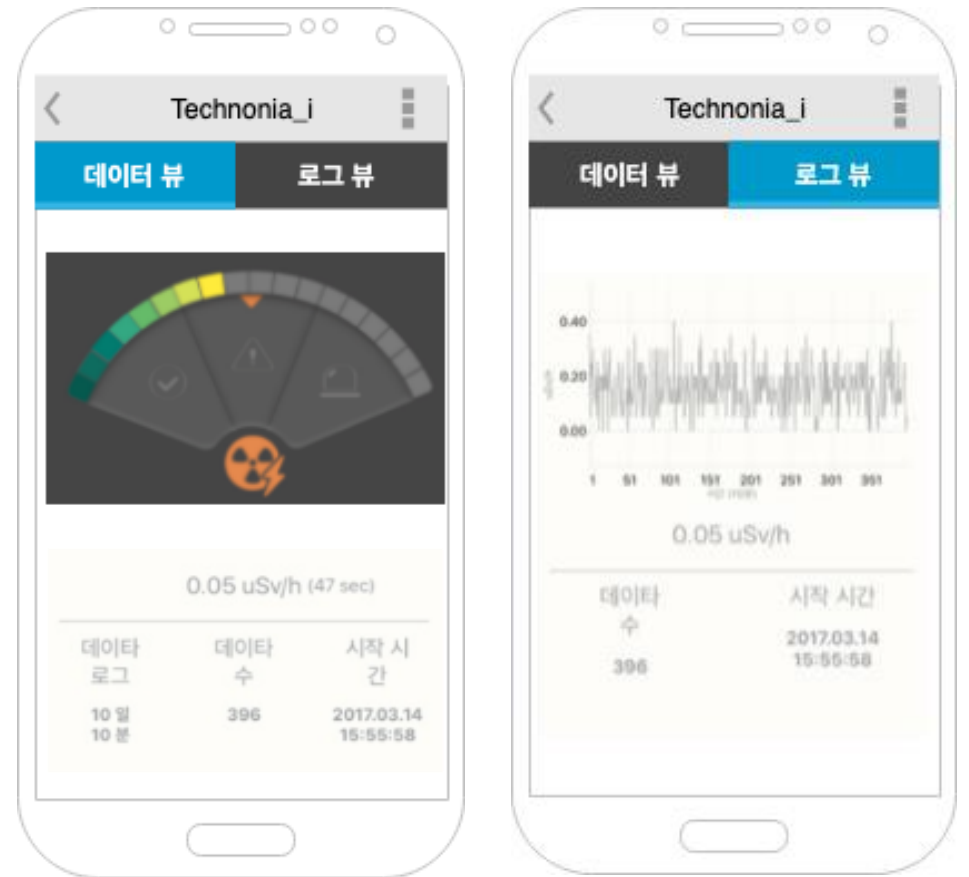
```
{data: [{temperature:27.5, humidity:43, datetime: "2019-06-01, 06:05"},  
{temperature:31.5, humidity:41, datetime: "2019-06-01, 06:10"}]}
```

# Radioactive Monitoring Screen

측정된 방사선 수치를 실시간으로  
모니터링하는 서비스를 제공하는 화면이다.

이 화면은 2 개의 뷰로 이루어져 있다 -  
하나는 데이터 뷰이고, 다른 하나는 로그  
뷰이다.

각 뷰에 대한 설명은 다음 슬라이드들에서  
계속된다.



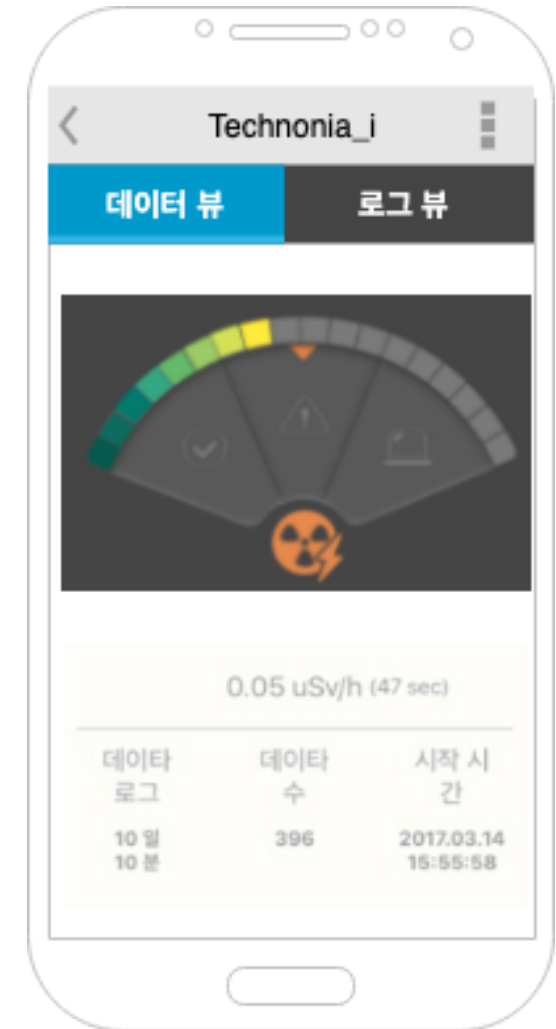


# Monitor Screen - DataView

이 데이터 뷰에서는 사용자가 현재 측정된 방사능 수치를 표시하고 현재 값이 어느 정도 위험도에 속하는지를 그래프를 통해서 나타낸다.

이전 ppt에서 기존의 Geiger MonitoringApp의 UI를 그대로 사용하는 것이 효율적이라는 의견이 있었다. 하지만, 아무래도 비즈니스 모델이 블루투스나 핸드폰의 이어폰 단자를 통해서 앱과 통신하던 기존의 모델과는 달리, 서버에 데이터를 전송하고 서버로부터 필요한 데이터를 받는 새로운 형식의 모델로 바뀌었기 때문에 기존의 데이터 뷰의 UI가 현재의 방식과 잘 맞지 않는다고 판단했기에 이렇게 기존과 다른 UI를 선택하였다.

새 UI의 구현을 위해서는 측정된 방사능 수치의 위험성을 보기 쉽게 하기 위해서 "안전", "약간 위험", "위험"과 "매우 위험"을 나타낼 수 있는 이미지들이 필요하다.



# Monitor Screen – Log View

로그뷰의 UI는 온도/습도 그래프 화면과 거의 동일하게 만들고자 한다.

측정한 방사능 수치를 그래프화 하고, 그래프화를 할 데이터의 날짜 범위를 사용자가 직접 고를 수 있게 만드는 것이다.



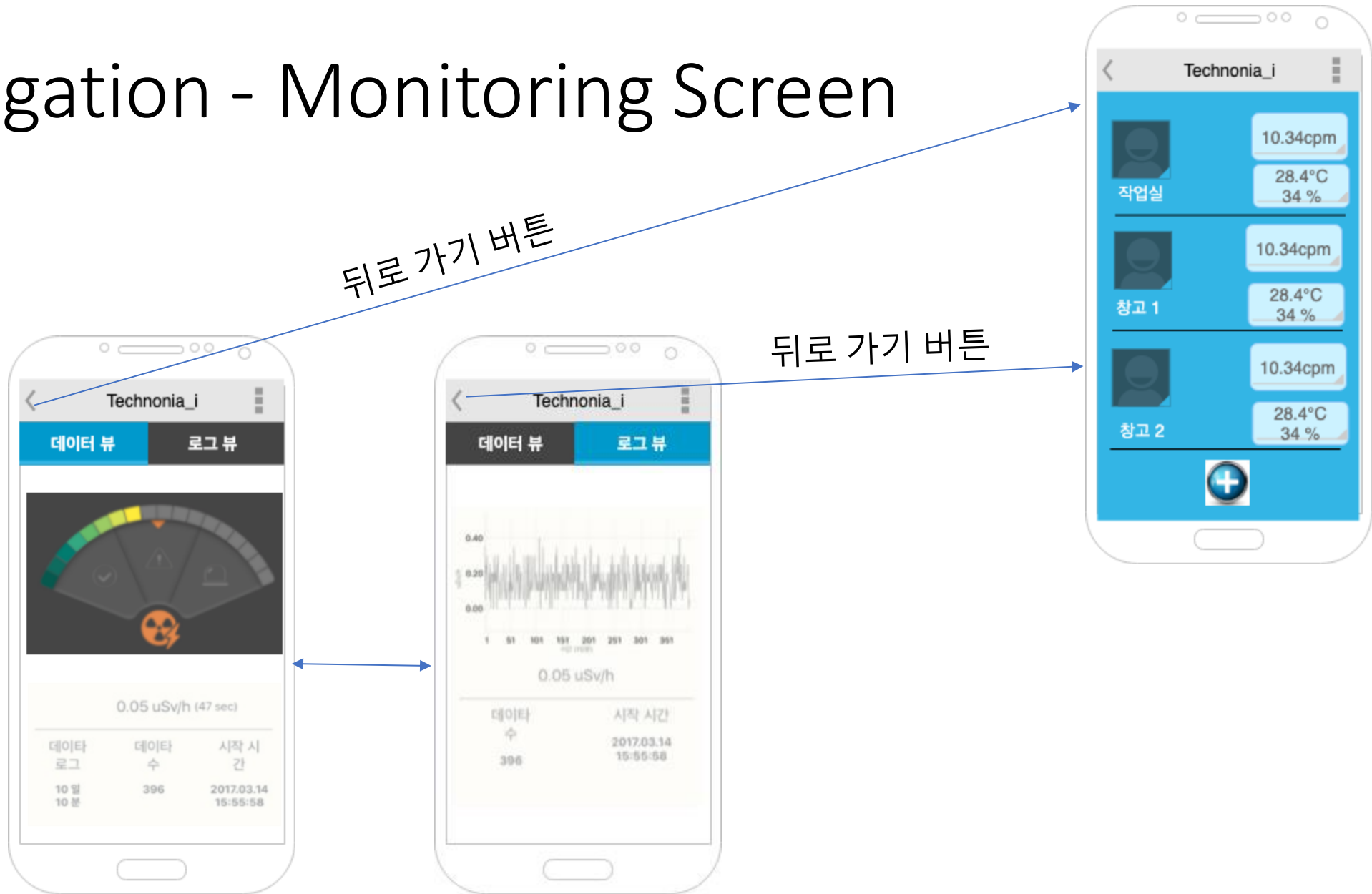
# Server Communication – Log View

- HTTP/HTTPS GET 요청을 이용해서 방사능 데이터를 포함한 JSON 포맷의 데이터를 서버로부터 전송받는다.
- 이 요청을 위해서 url 쿼리에 사용자의 아이디와 기기 고유 번호를 넣는다.
- 서버는 JSON 객체에 해당 기기가 지난 7일 (최대) 동안 수집한 데이터들을 온도와 습도로 나누어서 넣는다.

i.e.

```
{data: [{radioactive:0.02, datetime: "2019-06-01, 06:05"},  
{radioactive:0.03, datetime: "2019-06-01, 06:10"}]}
```

# Navigation - Monitoring Screen



# Application's Navigation Flow

