

아동/여성의 안전을 지키기 위한 통합 지원 서비스

# 지켜 Dream

<프로젝트 발표>

팀명: Lucy(루시)

구성원: 나창대, 김다연, 김준호, 박성민



# Content

- 1 서비스 제작 배경 및 필요성
- 2 화면 설계
- 3 개발 과정 및 방법
- 4 코드 설명
- 5 서비스 시연
- 6 Q&A



# 1. 제작 배경 및 필요성



## ▶ 제작 배경

※ 아동, 여성을 대상으로 한 강력 범죄 사건이 끊이지 않고 발생하고 있음

### 강남 아파트 여성 납치 현장 가보니... '대rob이지만 인적 드물어'

주민 "한적한 동네인데...황당하고 믿어지지 않아" SNS "강남 한복판에서" "피의자들 신상 공개해야"

뉴스1 제공 | 입력 : 2023.03.31 21:36



강남 아파트 여성 납치 현장 모습(2023.03.31) 뉴스1 김동규 기자 © news1

(서울=뉴스1) 김동규 기자 = "이곳은 평일 밤 11시 넘으면 인적이 드물어요."

강남에서 심야에 여성이 납치돼 살해됐다는 소식이 전해진 31일 저녁, 서울 강남구 역삼동의 납치 현장 인근에서 경사를 하는 한 상인의 말이다.

그는 "가게를 밤 10시까지 여는데 그 시간대에도 사람이 많이 다니는 길은 아니기에 11시 이후에는 인적이 더 드물었을 것"이라고 덧붙였다.

### 아이 삼촌? "할머니 이름 아느냐" 따져 물어 9세 여아 유괴 막은 주민

입력 2023.04.05 11:20 수정 2023.04.05 22:23

남보라 기자

304 0

공원서 놀던 아이에게 간식 주며 유인  
"인형 줄게" 집으로 데려가  
현관에서 있던 아이 주민이 구해

아홉 살 여자 어린이를 유인하던 40대 남성을 집요하게 따라가 범죄를 막아낸 주민이 화제다. 아이를 자신의 집 현관까지 유인했던 이 남성은 주민의 기지로 현행범으로 체포됐다.



40대 남성이 9세 여아를 데리고 가자(오른쪽 사진) 이를 수상히 여긴 주민 이씨(왼쪽 사진)가 아이들을 쫓아가는 모습이 CCTV에 담겼다. MBC 보도 캡처

### '안양토막살인' 정성현 "내가 죽이는 거 본 사람 있냐"

뉴스1 입력 : 2022.05.18 10:33 수정 : 2022.05.18 10:33



[서울=뉴스1]정성현, 2022.05.18. (사진=채널A '블랙: 악마를 보았다' 캡처) photo@newsis.com\*재판매 및 DB 금지 \*재판매 및 DB 금지

[서울=뉴스1]유다연 인턴 기자 = '안양 초등생 토막 살인사건'의 범인 정성현이 변변하게 무고하다고 주장한다.

해당 사건으로 아동안전 지킴이집 정책이 시행됨



### \* '아동 안전지킴이집'이란?

위급한 상황에 처한 어린이들을 임시로 보호해 주는 아동 긴급보호소로, 아이들이 쉽게 찾을 수 있도록 편의점·약국·문구점 등이 안전지킴이 집이 되어 아이들의 안전을 지켜주는 역할을 함

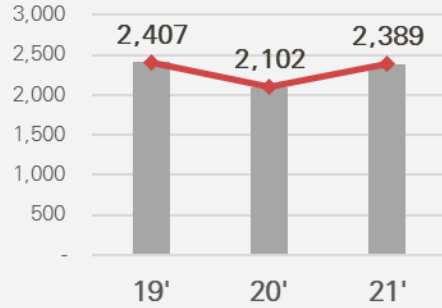
### \* '여성 안심지킴이집'이란?

위기 상황이 발생했을 시 안전하게 대피하도록 하여 보호해주고, 비상벨을 누를 경우 경찰이 즉시 현장 출동하여 안심 귀가를 돕는 서비스

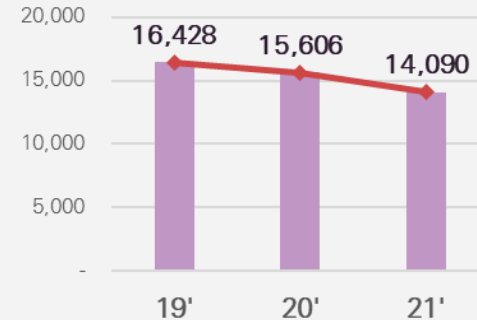
# 1. 제작 배경 및 필요성

## 지킴이집 인식 향상 필요

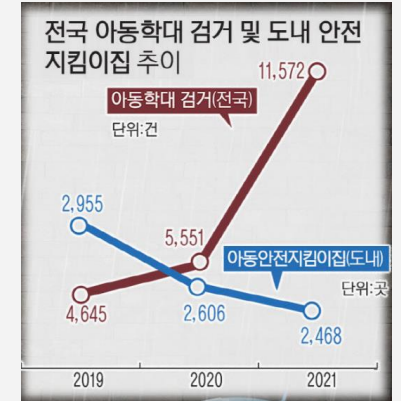
- 여성 및 아동 대상 강력 범죄가 지속적으로 발생하고 있음에도, 범죄예방 등의 중요한 역할을 하는 안전지킴이집 관련 정보를 평소 알지 못하는 경우가 많음.  
따라서, 충분한 정보 전달이 필요한 상황
- 지킴이집 정책에 대한 인식 부족으로 지킴이집 수가 감소하고 있다.



<2019~2021년 아동강력범죄 피해 건수>



<2019~2021년 여성강력범죄 피해 건수>

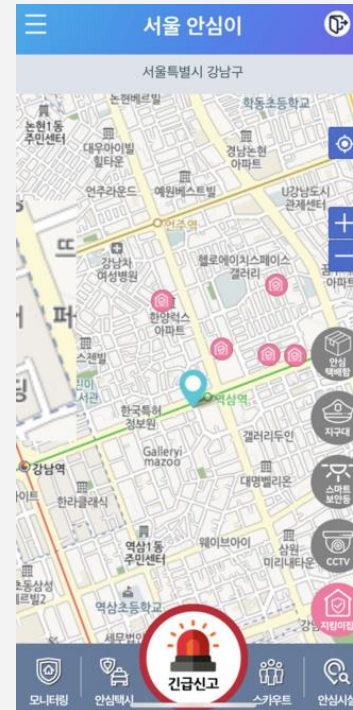


## 통합 정보 제공 필요

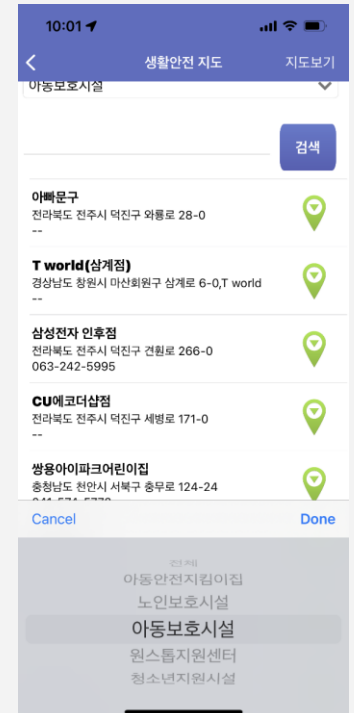
- 아동 안전지킴이집 및 여성 안심지킴이집의 정보를 얻기 위해서는 각각 다른 앱을 다운받아서 사용해야 함

### ▶ 필요성

- 분산되어있는 아동 지킴이집과 여성 지킴이집 정보를 한 곳에서 제공하여 빠르게 찾고, 이용자들끼리 정보를 공유할 수 있는 기회를 제공함
- 아동 및 여성 안전 지킴이집 정책과 지킴이집 활용 방법에 대한 인식을 높일 수 있음



<여성안심지킴이집 정보>



<아동안전지킴이집 정보>

## 서비스 소개

- **적용 대상 고객**

**여성과 아동이 있는 가족 구성원** 또는 **지킴이집 정보 및 긴급신고 서비스**가 필요한 고객

- **차별성**

1) **여성 지킴이집과 아동 지킴이집에 대한 정보**를 제공하는 기존 어플리케이션은 두 정보를 통합하여 제공하지 않으며, 본 서비스는 **두 정보를 모두 볼 수 있다는 점에 차별성이 있음**

2) 사용자가 **긴급신고 대기시간을 필요시점에 스스로 조절하여, 수동적으로 사용할 수 있다.**  
(유사 앱에선 목적지 근처에 도착하면 신고대기 기능이 자동으로 종료되는 문제점을 보완)

- **핵심기능**

1) 여성 및 아동 지킴이집의 위치 및 정보를 **지도 검색**을 통해 찾아볼 수 있음

2) 쉽고 **빠르게 신고 및 도움을 요청할 수 있는 긴급신고대기 기능**을 통해 위급 상황에 신속하게 대처할 수 있음

## 서비스 활용방안 및 기대효과

### [ 활용방안 ]

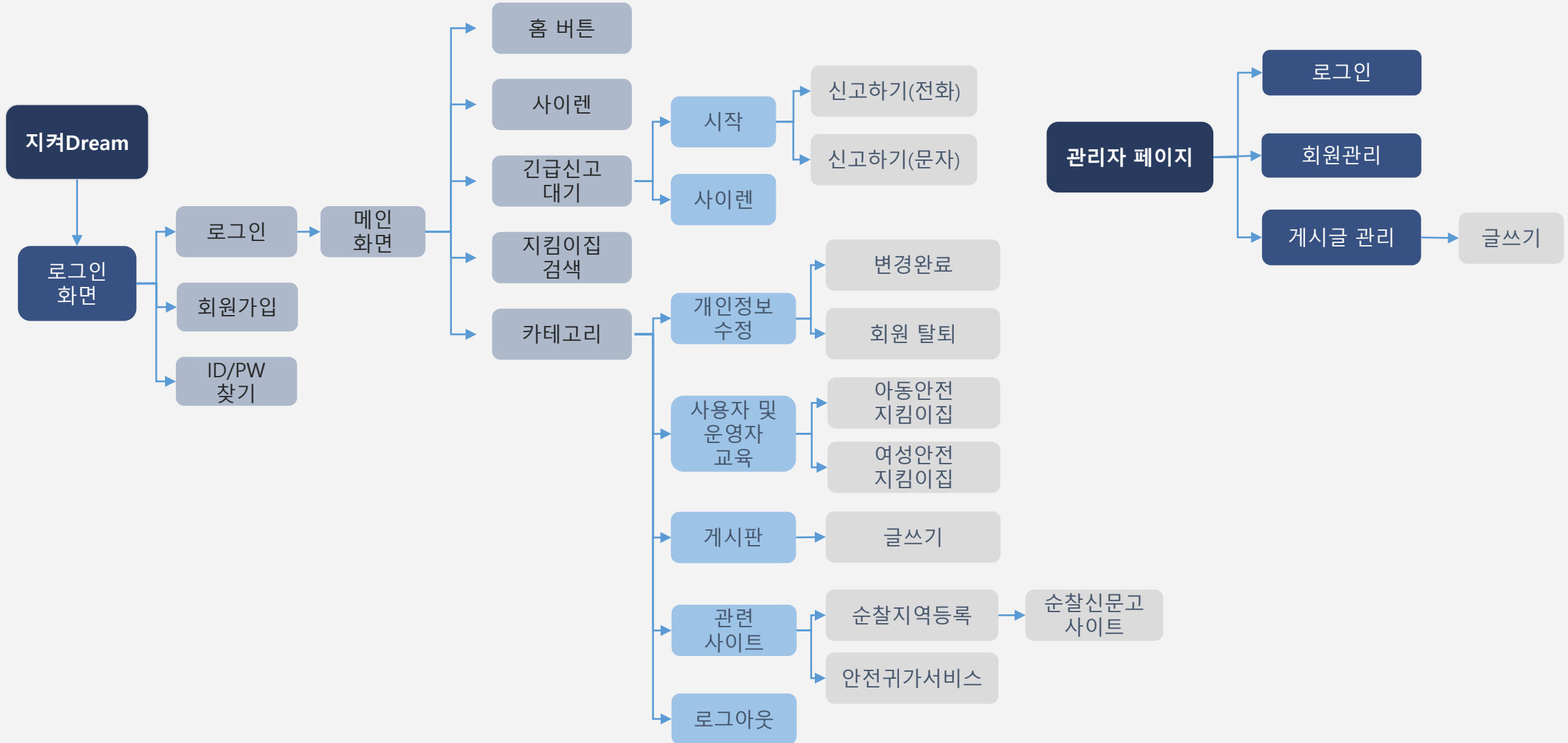
- ✓ 아동 및 여성에게 위협으로부터 안전한 공간에 대한 정보 제공
- ✓ 우리 가족 생활반경에 있는 안전지킴이집 정보를 미리 확인하여 범죄 상황에서 빠르게 이용할 수 있음
- ✓ 지킴이집의 사업주 또는 직원에게 지킴이집의 역할에 대한 교육을 제공할 수 있음
- ✓ 늦은 밤 귀가하는 여성에게 경호자가 동행하여 안심 귀가를 돕는 안심 귀가 서비스 제도를 신청하기 위한 정보를 제공
- ✓ 순찰 신문고 링크로 이동하여 취약지 순찰 범위 등록을 신청할 수 있음

### [ 기대효과 ]

- ✓ 아동과 여성 지킴이집 정보를 하나의 서비스로 찾을 수 있고, 관련된 다양한 정보를 제공하여 사용자 편의성 증대
- ✓ 지킴이집에 대한 인식과 이용자가 증가됨에 따라 범죄 예방에 도움이 되어, 지역사회 안전과 복지를 증진시킬 수 있음
- ✓ 본 서비스를 통해 안전 지킴이집 정책에 대한 시민들의 인지도와 인식이 증가되어, 지킴이집 선정을 희망하는 사업장 확대

## 2. 화면 설계

### ▶ Use Flow\_사용자 / 관리자





## 2. 화면 설계\_로그인/ 회원가입

화면	설명
 <p>The image displays three mobile application screens for the '지켜 Dream' app, illustrating the membership registration process. Red dashed boxes and numbers highlight specific UI elements and their functions:</p> <ul style="list-style-type: none"><li><b>Screen 1 (Main App Interface):</b><ul style="list-style-type: none"><li>1-1: ID, PW input fields.</li><li>1-2: '회원가입' (Sign Up) button.</li><li>1-3: '로그인' (Login) button.</li><li>1-4: '아이디/비밀번호 찾기' (Find ID/Password) button.</li></ul></li><li><b>Screen 2 (Terms and Conditions):</b><ul style="list-style-type: none"><li>2-1: '약관에 동의합니다.' (I agree to the terms) checkbox.</li></ul></li><li><b>Screen 3 (Sign Up Form):</b><ul style="list-style-type: none"><li>3-1: The entire sign-up form area.</li><li>3-2: '중복확인' (Check for duplicates) button.</li><li>3-3: 'submit' button.</li></ul></li></ul>	<p>1-1: ID, PW 입력 받는 input 창</p> <p>1-2: 회원가입 화면으로 이동시켜주는 버튼</p> <p>1-3:로그인 버튼을 클릭하여 로그인이 성공 시, 메인 화면으로 이동</p> <p>1-4: 아이디, 비밀번호를 찾을 수 있는 페이지로 이동</p> <p>2-1: 회원가입 버튼을 클릭하면 회원약관 동의 화면으로 이동하고, 약관 동의 체크 및 동의 버튼을 눌러야 회원가입 페이지로 넘어갈 수 있음</p> <p>3-1: 회원가입 정보를 입력할 수 있는 화면</p> <p>3-2: "중복확인" 버튼 클릭 시, 사용하고자 하는 아이디가 중복되는지 확인가능</p> <p>3-3: "가입 완료" 버튼을 누르면 회원가입이 완료됨</p>



## 2. 화면 설계\_아이디/비밀번호 찾기

화면	설명
	<p>4-1: 이름과 휴대폰 번호를 입력하여, 회원가입 시 설정한 아이디를 찾을 수 있음</p> <p>4-4: 비밀번호를 찾아야 하는 경우, 제공된 관리자 이메일로 문의하여야 한다.</p>
	<p>4-2: 입력한 이름 및 핸드폰번호가 가입된 정보와 일치할 경우, 아이디를 화면에 출력</p>
	<p>4-3: 가입된 정보와 일치 하지 않을 경우 확인 요청 메시지가 뜨며, 다시 시도 버튼을 클릭하여 다시 입력할 수 있음</p>

## 2. 화면 설계\_앱 홈화면

화면	설명
	<p>5-1: 사이렌 버튼 터치 시 사이렌 소리가 울리며, 소리가 울리는 동안 음파 이미지가 사이렌 주변으로 생성됨</p> <p>5-2: 버튼 터치 시 긴급 신고 대기 페이지로 이동</p> <p>5-3: 버튼 터치 시 "지킴이집 검색" 화면으로 이동</p> <p>5-4: 버튼 터치 시 카테고리 생성</p> <p>5-5: 버튼 터치 시 앱 홈 화면으로 이동</p>
	<p>6-1: 개인정보를 수정 할 수 있는 페이지로 이동</p> <p>6-2: 사용자/운영자 교육 페이지로 이동</p> <p>6-3: 게시판 페이지로 이동</p> <p>6-4: 관련 사이트 페이지로 이동</p> <p>6-5: 로그아웃 되어 로그인 페이지로 이동</p>

## 2. 화면 설계\_긴급신고대기

화면	설명
 <p>The image displays two mobile application screens for the '지켜 Dream' app, illustrating the emergency reporting process. The left screen shows the initial state with buttons for '신고하기 (전화)' (7-1), '신고하기 (문자)' (7-2), '시작' (7-3), and '사이렌' (7-4). The right screen shows the state after clicking '시작', where the buttons are highlighted in blue and labeled 7-5, 7-6, 7-7, and 7-8 respectively.</p>	<p>7-1: 버튼이 비활성화 되어있는 상태로, 긴급신고 대기 기능을 시작하기전에 실수로 누르는 것을 방지함</p> <p>7-2: 긴급신고대기 기능을 시작하는 버튼으로, 버튼을 누를 경우 위의 신고하기(전화) 및 신고하기(문자) 버튼이 활성화 됨</p> <p>7-3: 버튼을 누르면 사이렌 소리가 울림</p> <p>7-4: 네비게이션 바에서 메인 기능인 긴급신고 대기과 지킴이집검색 기능을 선택하여 이동할 수 있음</p> <p>7-5: 시작 버튼 클릭 후 버튼이 활성화된 상태이며, 버튼 클릭 시 112로 전화 연결하기위한 화면으로 이동</p> <p>7-6: 시작 버튼 클릭 후 버튼이 활성화된 상태이며, 버튼 클릭 시 보호자에게 긴급상황을 알릴 수 있는 문자를 전송할 수 있음</p> <p>7-7: 시작 버튼 클릭 후 기능을 종료할 수 있는 버튼으로 바뀌며, 종료를 원할 시(집 도착 시), 버튼을 눌러 버튼을 비활성화함</p>


## 2. 화면 설계\_긴급신고대기

화면	설명
<div data-bbox="275 354 710 1315"></div> <div data-bbox="710 1062 749 1093">8-1</div>	<p>8-1: 신고하기(전화) 버튼을 누를 시, 112로 전화 연결할 수 있는 화면으로 전환됨</p> <p>8-2: 회원가입 시 저장한 보호자번호로 문자를 보내줌</p> <div data-bbox="1019 354 1454 1315"></div> <div data-bbox="1416 444 1454 475">8-3</div>

## 2. 화면 설계\_지킴이집 검색

화면	설명
	<p>9-1: "지킴이집 검색" 화면</p> <p>9-2: 찾고자 하는 지역을 키워드로 검색하여, 인근 지킴이집의 위치를 지도 위에서 찾아볼 수 있음</p> <p>9-3: 입력한 키워드와 관련된 장소 목록들이 하단에 출력되며, 마커가 지도 위에 찍힌다.</p> <p>9-4: 지킴이집 마커 선택 시, 해당 마커의 상세위치 정보를 확인할 수 있다.</p> <p>&lt;마커의 범례&gt; 하늘색(숫자): 검색 결과로 나온 위치 파란색: 아동지킴이집 빨간색: 여성지킴이집</p>

## 2. 화면 설계\_개인정보 수정

화면	설명
 <p>10-1: 개인정보를 수정할 수 있는 화면</p> <p>10-2: 작성 취소 버튼을 누르면 앱 홈화면으로 이동</p> <p>10-3: 수정할 정보를 모두 입력한 후 변경 완료 버튼을 누르면 변경된 정보로 저장되며 앱 홈 화면으로 이동</p> <p>10-4: 회원 탈퇴 버튼을 누르면 탈퇴 여부를 확인하며, "네"를 누를 시 로그인 한 회원의 정보가 삭제되며 로그인 페이지로 이동</p> <p>10-5: 회원탈퇴 여부를 재차 확인하는 메세지</p>	

## 2. 화면 설계\_사용자 및 운영자 교육, 관련 사이트

화면	설명
	11-1: 아동 안전지킴이집 클릭 시 교육 내용이 출력
	11-2: 여성 안전지킴이집 클릭 시 교육 내용이 출력
	12-1: 순찰지역 등록을 원하는 사용자들에게, 순찰 신문고 이용 방법을 안내
	12-2: 버튼 클릭 시 경찰청 순찰지역 등록 신청 사이트가 브라우저에서 열림  12-3: 여성 안심귀가 스카우트 서비스에 대해 소개하고, 신청 방법을 제공



## 2. 화면 설계\_게시판, 게시판 글쓰기

화면	설명
<div><p>13-1: 전체 카테고리 목록</p><p>13-2: 글쓰기 버튼</p><p>13-3: 글 목록</p><p>13-4: 위험지역 공유</p><p>13-5: 제목을 입력하세요.</p><p>13-6: 내용을 입력하세요.</p><p>13-7: &lt;작성취소 &gt;완료&gt;</p></div>	<p>13-1: 콤보박스 터치 시 게시판 카테고리가 생성되며, 원하는 카테고리를 누르면 해당되는 게시글만 목록에 불러온다.</p> <p>13-2: 글쓰기 버튼을 누르면 게시글을 작성할 수 있는 화면으로 이동</p> <p>13-3: 글 제목 터치 시 게시글 상세 정보를 열람할 수 있음</p> <p>13-4: 콤보박스를 눌러서 작성할 글의 카테고리를 선택</p> <p>13-5: 작성하고자 하는 제목과 내용을 작성</p> <p>13-6: "작성 취소" 버튼을 누르면 게시글 리스트 화면으로 이동</p> <p>13-9: "완료" 버튼을 누르면 작성한 게시글이 저장되며, 게시글 리스트 화면으로 이동되어 작성한 글을 목록에서 볼 수 있음</p>

## 2. 화면 설계\_관리자페이지\_로그인 화면, 관리자페이지\_메인 화면

화면	설명
<div><div><div>Admin Page</div><div><div><div>14-1</div><div><div>Login To Admin Page</div><div><div>userid</div><div>password</div><div>로그인</div></div></div></div></div></div></div>	<p>14-1: 관리자 아이디 및 비밀번호를 입력하면 관리자 메인 페이지로 이동</p> <p>15-1: 관리자 메인화면</p> <p>15-2: 관리자 메인화면에서 회원 및 게시판 등록 현황을 볼 수 있음</p> <p>15-3: 로그아웃 버튼 클릭 시 로그아웃 되어, 로그인 페이지로 이동</p> <p>15-4: 다른 페이지에 있다가 Home 카테고리를 클릭하면 관리자 메인 화면으로 이동</p> <p>15-5: 회원관리 카테고리를 클릭하면 회원 데이터를 볼 수 있는 화면으로 이동</p> <p>15-6: 게시판 관리 카테고리를 클릭하면 게시판 데이터를 볼 수 있는 화면으로 이동</p>

## 2. 화면 설계\_관리자페이지\_회원 관리

화면

설명

Admin Page

로그아웃

Home

회원관리

게시판 관리

16-1

회원 관리

사용자 회원 관리

16-2

추가

ID	Name	보호자 연락처	휴대폰 번호	수정 / 삭제
user1	User1	123456789	111222333	<div><div>16-3</div><div>수정</div><div>16-4</div><div>삭제</div></div>
user10	User10	123456789	111222342	<div><div>수정</div><div>삭제</div><div>16-5</div><div>확인</div></div>
user2	User2	123456789	111222334	<div><div>수정</div><div>삭제</div></div>
user3	User3	123456789	111222335	<div><div>수정</div><div>삭제</div></div>
user4	User4	123456789	111222336	<div><div>수정</div><div>삭제</div></div>

16-1: 가입한 회원 데이터를 모두 볼 수 있는 테이블

16-2: 신규회원을 추가하여 데이터 테이블에 업로드 할 수 있음

16-3: 버튼을 클릭하여 회원 정보를 수정할 수 있음

16-4: 버튼을 클릭하여 회원 정보를 삭제할 수 있음

16-5: 회원 삭제 시 삭제 확인 버튼이 생성되고, 확인 버튼을 눌러야 해당 계정이 삭제됨

## 2. 화면 설계\_관리자페이지\_회원 관리

화면	설명
<div><div>Admin Page</div><div><div>Home</div><div>회원관리</div><div>게시판 관리</div></div><div><div>회원 관리</div><div>사용자 회원 관리</div><div><div>Data Table</div><div>회원 정보 수정17-1</div><div><div>이름:</div><div>r감사과</div><div>비밀번호:</div><div>undefined</div><div>보호자 휴대폰 번호:</div><div>010-1111-1111</div><div>휴대폰 번호:</div><div>010-2222-2222</div><div>수정취소</div></div><div><div>ID</div><div>13</div><div>Name</div><div>13</div></div></div></div></div> <div><div>Admin Page</div><div><div>Home</div><div>회원관리</div><div>게시판 관리</div></div><div><div>회원 관리</div><div>사용자 회원 관리</div><div><div>Data Table</div><div>회원 추가17-2</div><div><div>ID:</div><div></div><div>이름:</div><div></div><div>비밀번호:</div><div></div><div>보호자 휴대폰 번호:</div><div></div><div>휴대폰 번호:</div><div></div><div>추가취소</div></div><div><div>ID</div><div>Name</div><div>보호자 연락처</div><div>휴대폰 번호</div><div>수정 / 삭제</div><div><div>user1</div><div>User1</div><div>null</div><div>123456789</div><div>수정</div><div>삭제</div></div></div></div></div></div>	<p>17-1: 회원 정보를 수정할 경우, 정보를 입력하고 "수정 " 버튼으로 수정을 완료하거나 "취소"버튼으로 수정을 취소할 수 있음</p> <p>17-2: 회원을 신규로 생성할 수 있으며, 정보를 모두 입력하고 "추가 " 버튼으로 신규 회원 생성을 완료하거나 "취소"버튼으로 추가를 취소할 수 있음</p>

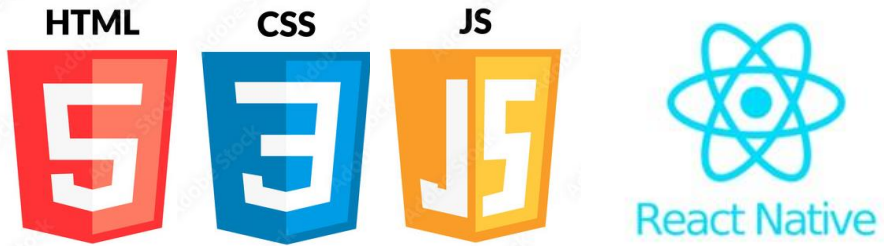
## 2. 화면 설계\_관리자페이지\_게시판 관리

화면	설명															
<div><div><div>Admin Page로그아웃</div><div><div>Home</div><div>회원관리</div><div>게시판 관리</div></div><div><div>게시판 관리</div><div>18-1게시글 목록<div>18-4글쓰기18-3삭제</div><div>18-2<div><div><div>카테고리</div><div>공지</div></div><table><tr><th></th><th>카테고리</th><th>작성자 ID</th><th>글 제목</th><th>작성 일자</th></tr><tr><td><input type="checkbox"/></td><td>공지</td><td>admin</td><td>지켜 Dream 서비스 확대 안내</td><td>2023-06-07 13:54:44</td></tr><tr><td><input type="checkbox"/></td><td>공지</td><td>admin</td><td>서비스 점검 안내</td><td>2023-06-07 10:06:40</td></tr></table></div><div>18-5</div></div></div></div><div><div><div>Admin Page로그아웃</div><div><div>Home</div><div>회원관리</div><div>게시판 관리</div></div><div><div>글쓰기</div><div>게시판 관리 &gt; 글쓰기19-1</div><div><div>카테고리: 위험지역</div><div>제목: 제목을 입력하세요.</div><div>Enter text ...</div><div>19-2&gt; 저장하기</div></div></div></div></div><div><div><div>Admin Page로그아웃</div><div><div>Home</div><div>회원관리</div><div>게시판 관리</div></div><div><div>게시판 관리</div><div>20-1공지강남구 경찰청 지킴이집 이벤트 소개admin 2023-05-30 11:09:18</div><div>20-2강남구 경찰청에서 진행하는 지킴이집 이벤트를 소개합니다. 많은 관심 부탁드립니다. (URL: ...)</div><div>삭제하기</div></div></div></div></div></div>		카테고리	작성자 ID	글 제목	작성 일자	<input type="checkbox"/>	공지	admin	지켜 Dream 서비스 확대 안내	2023-06-07 13:54:44	<input type="checkbox"/>	공지	admin	서비스 점검 안내	2023-06-07 10:06:40	<p>18-1: 콤보박스를 누르면 카테고리를 선택하여 해당되는 게시글 리스트를 볼 수 있음</p> <p>18-2: 각각의 게시글을 체크박스로 선택할 수 있음</p> <p>18-3: 삭제버튼을 누르면 체크박스로 선택된 게시글을 한번에 삭제할 수 있음</p> <p>18-4: 글쓰기 버튼을 누르면 게시물을 작성할 수 있음</p> <p>19-1: 작성하는 게시글의 카테고리를 선택하고, 게시글의 제목 및 내용을 작성할 수 있음</p> <p>19-2: 버튼을 클릭하면 작성한 게시글이 저장된 후 게시글 리스트 화면으로 이동</p> <p>20-1: 선택한 게시물의 상세 내용을 보여줌</p> <p>20-2: 삭제하기 버튼 누를 시 열람 중인 게시글을 바로 삭제할 수 있음</p>
	카테고리	작성자 ID	글 제목	작성 일자												
<input type="checkbox"/>	공지	admin	지켜 Dream 서비스 확대 안내	2023-06-07 13:54:44												
<input type="checkbox"/>	공지	admin	서비스 점검 안내	2023-06-07 10:06:40												

### 3. 개발 과정 및 방법

#### ▶ 개발 Tool

##### Front-end



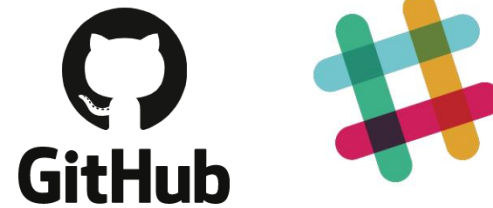
##### Database



##### Back-end



##### DevOps



### 3. 개발 과정 및 방법

#### ▶ 활용 공공데이터

데이터 명	출처	데이터 유형
경찰청_아동안전지킴이집 정보 서비스	공공 데이터 포털	Open API
서울시 안심귀갓길 서비스	서울시 열린 데이터 광장	Open API

#### ▶ 개발 과정

기간	내용
4.10~4.21	주제선정 및 기획 / UI&UX 디자인(앱의 외관, 직관적인 사용자 경험을 설계)
4.22~5.12	프론트엔드 개발(HTML, CSS, JavaScript를 이용하여 앱의 사용자 인터페이스 개발)
5.13~5.26	백엔드 개발(데이터 처리, 데이터베이스 연동, API 개발 등)
5.27~5.28	클라이언트 및 서버배포
5.28~5.31	테스트 및 디버깅 / 구글 플레이스토어 배포
6.1~	유지보수와 업데이트



## 4. 코드 설명

## ▶ 사이렌소리

```
// 사이렌 소리 !  
let switch_siren = false; // 사이렌소리 스위치  
const audio = new Audio("./audio/siren.mp3");  
  
function playMusic(){  
    if(switch_siren == false){  
        audio.play()  
        switch_siren = true;  
    }else {  
        audio.pause()  
        switch_siren = false;  
    }  
}
```

### 1. Audio 객체 제어

- 음원 파일을 제어할 수 있는 JavaScript에서 제공하는 내장 객체인 Audio를 이용하여 play와 pause 메소드를 사용.
- Play() : 음원파일을 재생
- Pause() : 재생중인 음원파일을 일시정지

## ▶ 긴급신고(Client)

```
// singo_btn 버튼 클릭 이벤트 핸들러
function singo_btn(guardianHp) {
  const userAgent = navigator.userAgent.toLowerCase();
  let singo = document.querySelectorAll('.singo_btn');
  const start_btn = document.querySelector('#start_btn');
  const telButton = document.getElementById('tel');
  const smsButton = document.getElementById('sms');
  const tel = "document.location.href = 'tel:112'"; // 경찰서로 고정 될거기때문에 const
  let sms = "document.location.href = "; // 내용물을 받아와야 해서 let으로 변수 선언

  // singo_btn 버튼 토글 및 클릭 이벤트 처리
  singo.forEach((data) => {
    data.classList.toggle('active');
    if (data.classList.contains('active')) {
      telButton.setAttribute('onclick', `${tel}`);
      smsButton.setAttribute('onclick', `${sms}`);
    } else {
      telButton.setAttribute('onclick', "#");
      smsButton.setAttribute('onclick', "#");
    }
  });
}

// 안드로이드랑 IOS를 구별 해내는 SMS 처리 코드 추가
if (/iphone|ipad|ipod/.test(userAgent)) {
  // ios 처리 코드 추가
  console.log("ios에서 실행 중입니다.");
  sms += `sms:${guardianHp}&body=고객님의 소중한 가족이 위험합니다`;
} else if (/android/.test(userAgent)) {
  // Android 처리 코드 추가
  console.log("Android에서 실행 중입니다.");
  sms += `sms:${guardianHp}?body=고객님의 소중한 가족이 위험합니다`;
}
```

### 1. Location.href 사용

- location은 JavaScript의 내장 객체를 이용하여 새로운 URL을 설정 할 수 있는 href 매소드를 사용.

### 2. URI스킴 사용

- URI 스킴이란 HTTP, HTTPS, FTP, tel, sms등 프로토콜을 말함.
- Tel과 sms 프로토콜을 사용 하여 전화걸기와 문자메세지 전송 하기를 구현.

## ▶ 긴급신고(Server)

```
const token = authHeader.split(' ')[1];
jwt.verify(
  token,
  config.jwt.secretKey,
  async (error, decoded) => {
    if (error) {
      return res.status(401).json(AUTH_ERROR);
    }
    const user = await userRepository.findById(decoded.userid);
    const admin = await userRepository.findById(decoded.userid);

    if (admin) {
      req.userid = admin.userid;
    } else if (user) {
      req.userid = user.userid;
    } else {
      return res.status(401).json(AUTH_ERROR);
    }
    next();
  }
);
```

### 1. 토큰 값을 이용하여 UserID 찾기

- 저희 서버에 접근을 하게 되면 토큰 값으로 ID를 찾아옴.
- Jwt.verif 함수는 JWT를 검증하는 함수.
- DB에서 User ID를 검색 후 찾지 못하면 401에러 코드를 반환하고, User 또는 Admin의 아이디를 찾았으면 조건만에 맞는 ID로 반환 함.

## ▶ 긴급신고(Server)

```
// 보호자 전화번호 받아오기
export async function guardianHp(req, res, next){

  const userid = await adminRepository.UserfindById(req.userid);

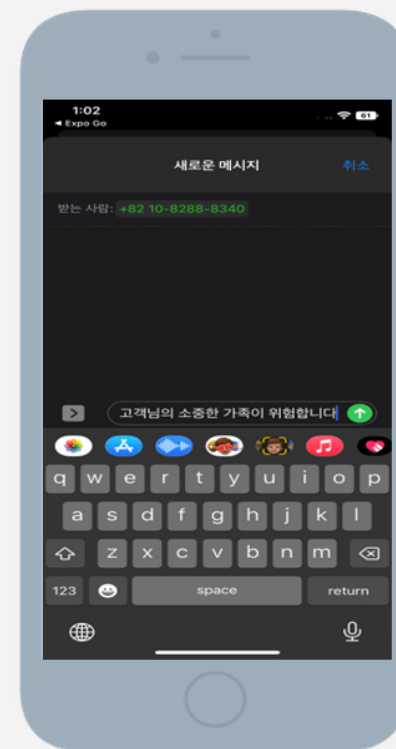
  try {
    // DB에서 해당 사용자를 삭제
    const sql = 'SELECT guardianHp FROM user WHERE userid = ?';
    const [rows] = await pool.query(sql, [userid.userid]);
    const guardianHp = rows[0].guardianHp;

    // 성공시 보호자 번호
    res.status(200).json({ guardianHp });

  } catch (error) {
    console.error(error);
    res.status(500).json({ message: '서버 오류' });
  }
}
```

## 2. 보호자 연락처 받아오기

- Client의 ID를 토큰 값으로 찾아와, 그 ID를 갖고 DB에 접근하여 보호자 연락처(guardianHp)의 정보를 받아 Client에게 반환.



## ▶ 카카오 지도 API에서 제공하는 HTML + Javascript 코드 내에 지킴이집 API에서 가져온 정보가 저장된 json 파일에 접근

```
async function func2() {
  var result = await fetch('safekid_fin.json')
  .then(response => response.json())

  var positions = []
  for (let i = 0; i < result.length; i++) {
    // 경도
    let LONGITUDE = parseFloat(result[i].x)
    // 위도
    let LATITUDE = parseFloat(result[i].y)
    let REMARK = result[i].name
    let ADDRESS = result[i].address
    let HP = result[i].tel

    var temp2 = {
      title: REMARK,
      latlng: new kakao.maps.LatLng(LONGITUDE, LATITUDE)
    }
    let temp3 = JSON.stringify(temp2)
    positions.push(temp3)
  }
}
```

### 1. Json 형식의 파일에서 데이터 가져오기

- fetch('safekid\_fin.json')을 통해 safekid\_fin.json 파일을 가져오고, response.json()을 사용하여 가져온 데이터를 JSON 형식으로 변환.

### 2. 데이터 추출 및 저장

- 가져온 JSON 데이터를 순회하면서 각 항목의 경도(LONGITUDE), 위도(LATITUDE), 이름(REMARK), 주소(ADDRESS), 전화번호(HP)를 추출
- 추출한 정보를 이용하여 새로운 객체(temp2)를 생성
- Kakao Maps API의 클래스를 사용하여 위도와 경도로 좌표를 생성
- 생성한 객체와 좌표를 문자열로 변환한 후, positions 배열에 추가
- Positions에 추가된 위치 정보는 kakao에서 제공한 마커 표시 코드에 의하여 지도에 표시됨

## ▶ 표시된 마커를 클릭하면 open API에서 가져온 정보를 하단 컨테이너에 띄워줌

```
kakao.maps.event.addListener(marker, 'click', function () {  
  
    let content =  
    `<h2 style="font-size:18px">${REMARK}</h2>  
<h2 style="display:block; color:gray; font-weight:500">${ADDRESS}</  
h2><br>  
  
<h2 style="display:inline; float: left;">${HP} (관리기관 전화번호)</  
h2>`  
  
    const infoBox = document.getElementById('content');  
    infoBox.style.display = 'block'  
    infoBox.innerHTML = content;  
})  
  
kakao.maps.event.addListener(map, 'click', () => {  
    const infoBox = document.getElementById('content');  
    infoBox.style.display = 'none'  
})
```

### 1. 마커 클릭 이벤트

- kakao.maps.event.addListener 함수를 사용하여 마커의 클릭 이벤트를 등록
- 클릭 이벤트 함수 내에서는 클릭한 마커에 해당되는 지킴이집의 시설명, 주소, 관리기관 전화번호 등의 정보를 포함한 HTML을 생성하도록 함

### 2. 정보 표시 및 숨기기

- 클릭 이벤트 함수 내에서 정보를 표시하기 위한 컨테이너 요소를 가져옴
- 안보이도록 설정해놓은 컨테이너를 보이도록 하고, 위에 작성한 HTML 컨테이너를 내용으로 추가
- 정보 표시 컨테이너는 클릭한 마커의 위치에 따라 동적으로 변경
- 지도를 클릭했을 때는 정보 표시 컨테이너를 숨김



## ▶ 게시판 데이터를 가져와서 화면에 리스트로 출력하고, 스크롤 이벤트를 처리하여 페이지를 추가로 로드

```
let postIds = []
const token = localStorage.getItem('token'); // 토큰을 로컬 스토리지에서 가져옴
let page = 0

const server = 'https://port-0-safedream-backend-otjl2cli33x5tw.sel4.cloudtype.app';

// 무한 스크롤 적용 코드
function fetchData(page) {
  let selectedCategory = categorySelect.value;
  fetch(`${server}/safedream/board_user?category=${selectedCategory}&page=${page}`, {
    method: 'GET',
    headers: {
      'Authorization': 'Bearer ' + token // 가져온 토큰을 헤더에 추가
    }
  }).then(response => response.json())
    .then(data => {
      for (let i in data.result) {
        let listul = document.createElement('ul');
        let listli0 = document.createElement('li');
        let listli1 = document.createElement('li');
        listul.classList.add('content');
      }
    });
}

function handleScroll() {
  console.log(page)
  const scrollHeight = boardList.scrollHeight;
  const scrollTop = boardList.scrollTop;
  const clientHeight = boardList.clientHeight;

  if (scrollTop + clientHeight >= scrollHeight) {
    page = page + 15;
    fetchData(page)
    return page
  }
}

boardList.addEventListener('scroll', handleScroll)
```

### 1. fetchData 함수

- 선택된 카테고리 and 페이지 번호를 기반으로 서버에서 데이터를 가져옴.
- fetch 함수를 사용하여 GET 요청을 보내고, 가져온 데이터를 JSON 형식으로 변환
- 가져온 데이터를 반복문을 통해 화면에 출력
- 리스트 항목을 클릭했을 때, 해당 게시물 정보를 출력하는 새로운 페이지로 이동

### 2. 스크롤 이벤트 처리

- boardList 요소(리스트 출력 부분)의 스크롤 이벤트를 감지하여 처리
- 스크롤 위치와 요소의 크기를 비교하여 스크롤이 끝에 도달하면 DB에서 가져올 레코드 번호를 증가시키고 fetchData 함수를 호출하여 추가 데이터를 가져옴

▶ 선택한 카테고리 정보에 해당되는 게시판 데이터를 불러와서 전달하고, 스크롤 이벤트를 처리하여 페이지를 추가로 로드

```
export async function createlist(req, res) {  
  const category = req.query.category;  
  let page = req.query.page;  
  // MySQL에서 게시판 정보를 조회합니다.  
  const result = await Repository.list(category, page);  
  
  if(!result){  
    return res.status(401).json({ error: 'Invalid userid' });  
  }  
  
  return res.status(200).json({result})  
}
```

```
export async function list(category, page) {  
  console.log(page)  
  if (category === '전체') {  
    return db.execute('SELECT post_id, title, DATE_FORMAT  
      (created_At, \'%Y-%m-%d\') as created_At FROM board order by  
      post_id desc limit ?, 15 ', [page])  
      .then((result) => result[0]);  
  } else {  
    return db.execute('SELECT post_id, title, DATE_FORMAT(created_At,  
      \'%Y-%m-%d\') as created_At FROM board WHERE category=? order by  
      post_id desc limit ?, 15', [category, page])  
      .then((result) => result[0])  
  }  
}
```

1. Client에서 선택한 카테고리에 해당되는 게시판을 불러옴

- Client 요청으로부터 카테고리 and page 정보를 받아옴
- list 메서드를 호출하여 해당 카테고리와 페이지에 맞는 게시판 데이터를 조회
- 조회 결과가 없을 경우, 401 상태 코드와 에러 메시지를 반환
- 조회 결과를 200 상태 코드와 함께 JSON 형식으로 client에 전달

2. 카테고리와 page 정보를 토대로 DB에서 게시판 정보를 가져옴

- 카테고리가 '전체'인 경우, 모든 게시판 데이터를 조회하여 페이지에 따라 15개씩 데이터를 최근에 생성된 순서대로 가져옴
- 카테고리가 '전체'가 아닌 경우, 해당 카테고리에 해당하는 게시판 데이터를 조회하여 데이터를 가져옴

## ▶ Troubleshooting

### ✓ 아동안전지킴이집 공공데이터 open API 사용 중 발생

#### \* 문제점

공공데이터 포털에서 제공하는 데이터는 전국의 지킴이집 시설정보를 가지고 있어서 불필요한 데이터를 분류하는 작업이 필요했음.  
Open API를 호출할 때 검색하려는 지역의 범위를 위도, 경도의 최대, 최소 값을 기준으로 설정할 수 있는데, 서울시 범위의 위도 및 경도 최대, 최소값을 대입하여 불러오는 경우 서울시 외의 다른 지역 데이터가 함께 포함됨

#### \* 원인

위도 경도의 최대, 최소값으로는 서울시 범위를 정확하게 구분할 수 없음

#### \* 해결방안

전국의 아동안전지킴이집 API에서 fetch를 사용하여 원하는 데이터를 가져와서 DB에 저장하고, sql문으로 주소 컬럼에서 서울시에 해당되는 데이터만 select한 뒤 저장하여, 서울시 지킴이집 정보만 json 파일로 저장하여 사용함.

open api 규칙상 한 페이지에 100개까지만 정보를 출력할 수 있기 때문에 for문을 사용해야 했음.

```
for(let i=0; i <=121; i++){
  const url = `https://www.safe182.go.kr/api/lcm/
  safeMap.do?esntlId=10000550&authKey=f93f11232504459b&
  detailDate1=09&pageIndex=${i}&pageUnit=100`;
  fetch(url, {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
  })
    .then((response) => response.json())
    .then((data) => {
      for(let j=0; j < 13; j++){
        let name = data.list[j].bsshNm;
        let address = data.list[j].adres;
        let tel = data.list[j].telno;
        let x = data.list[j].lcinfoLa;
        let y = data.list[j].lcinfoLo;
        db.execute('insert into safekid (name,
        address, tel, x, y) values (?, ?, ?, ?, ?)
        ', [name, address, tel, x, y]).catch(err
        => console.log(err))
      }
    })
    .catch((error) => console.error(error));
}
```

```
-- sql 쿼리
insert into safekid_fin (name, address, tel, x, y) select
name, address, tel, x, y from safekid where address like
concat('%', '서울', '%');
```

## 5. 서비스 시연

## \* 프로젝트 후기



김다연: 처음으로 개발 프로젝트를 경험해보면서 매일 높은 산을 만나는 것 같았지만 팀원들과 함께 해결하면서 좋은 경험과 기억이 되었습니다. 초반에 갑작스럽게 팀원이 줄었고, 촉박한 시간 내에 프로젝트 결과물을 완성해야 했기 때문에 우리에게 가장 합리적인 목표를 찾아나가야 했습니다. 어떤 것을 포기하고 어떤 것을 끝까지 가져가야 할 지, 일정을 맞추기 위해 무엇이 우선시 되어야 하는지 고민하는 시간이 많았습니다. 그리고 그 결과를 모든 팀원들이 존중해주고 든든하게 함께해줘서 목표를 이룰 수 있었습니다.

나창대: 프로젝트를 진행하면서 많은 문제들이 있었으나 조원들이 합심하여 그 문제에 대해 생각해보고 해결해가면서 많은걸 배우게 되었고 그렇게 하나씩 해결해 가며 성장하는 저희의 모습을 볼때 보람을 느꼈습니다.

김준호: 프로젝트를 하면서 많은 것을 처음 해보았고, 처음에는 어려움이 많았 습니다. 하지만 팀원들과 함께 해결하면서 다양한 경험을 쌓을 수 있었고, 그 경험은 매우 감사하게 느껴졌습니다. 이러한 경험으로 성장하고 더욱 더 많은 것을 이루는데 도움이 되었습니다

박성민: 수업시간에 배우는 것보다 프로젝트 하면서 얻어가는 것들이 더 많았습니다. 첫 프로젝트이기에 아쉬운 점도 많았으나, 그렇기에 앞으로 남은 프로젝트에는 시간 안배를 잘 할 수 있을 것 같습니다. 팀원분들 모두 너무 고생하셨고, 끝까지 잘 완수한 것만으로도 너무 다행이고 감사했습니다.



## 6. Q&A

# Thank you~

