**신주민 -**

**네이버 지도 API 를 이용한 공간**

**중,단기 임대 매칭 서비스**

**최종 보고서**

과목명: 종합설계 팀: 4조 CDT

팀장 : 이태규

팀원 :류권환 팀원 : 윤혜선 팀원 : 박혜림

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 구분 | 소속 | 성명 | 날짜 | 서명 |
| 작성자 | 한국외국어대학교 | 이태규 | 2019.06.02 |  |
| 한국외국어대학교 | 류권환 | 2019.06.02 |  |
| 한국외국어대학교 | 윤혜선 | 2019.06.02 |  |
| 한국외국어대학교 | 박혜림 | 2019.06.02 |  |
|  |  |  |  |
|  |  |  |  |
| 검토자 | 한국외국어대학교 | 이태규 | 2019.06.03 |  |
| 한국외국어대학교 | 이태규 | 2019.06.09 |  |
| 한국외국어대학교 | 류권환 | 2019.06.10 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| 사용자 |  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| 승인자 | 한국외국어대학교 | 홍진표 | 2019.06.11 |  |

머리말

본 문서는 네이버 지도 API를 활용한 공간 중,단기 임대 매칭 서비스개발을 위한

관련 기술과 시스템구성도 및 기능 설명, 그리고 이 시스템을 이용할 때의 설계를 기술한 것이다.

**목차**

**1. 개요**

1.1 목적

1.2용어, 약어정리

**2. 제품 소개**

2.1 제품 개요

2.2 제품 필요성

**3. 시스템 구성**

3.1 시스템 구성도

3.2 소프트웨어

3.3 개발 환경

**4. 제품 기능 및 기능동작**

4.1 App 기능

4.2 Firebase Realtime DB

**5. 시스템 시나리오**

5.1 순차 다이어그램

5.2 시나리오

**6 세부 추진 계획 및 일정**

**7. 팀원 담당 업무**

**8 참고 문헌**

## 1. 개요

본 장에서는 네이버 지도 API를 이용한 공간 중, 단기 임대 매칭 서비스에 대한

목적과 범위, 참고 자료, 그리고 본 문서의 개요를 소개한다.

### 1.1 목적

본 프로젝트의 목적은 네이버 지도 API를 이용한 공간 중, 단기 임대 매칭 서비스를 구축하는데 그 목적이 있다. 본 프로젝트는 기존 네이버 카페, 에브리타임 등에 임의로 임대, 임차 게시글을 올렸던 것을 공식적인 기관으로부터 제공받도록 하며, 카카오톡 로그인 API를 통해 간단하게 로그인을 하고 네이버 지도 API를 이용하여 사용자가 원하는 지역에 있는 중, 단기 임대에 관한 매칭 서비스를 제공받을 수 있도록 하는 것에 프로젝트의 목적을 둔다. 프로젝트를 진행하기 위해 아래의 사항을 구체적으로 명시하고 구현하도록 한다.

1. 네이버 지도 API의 원리와 사용법
2. 카카오 로그인 API의 원리와 사용법
3. Firebase를 이용한 서버, DB 구축 방법
4. 앱에서 서버와의 통신

#### 1.2용어, 약어 정리

본 요구사항 정의서에서 사용된 용어 및 그에 대한 설명은 다음과 같다.

* Server : 다른 프로그램에게 서비스를 제공하는 컴퓨터 프로그램으로, 서버 는 네트워크 전체를 감시 및 제어하거나 메인프레임이나 공중망을 통한 다른 네트워크와의 연결, 소프트웨어 자원(데이터, 프로그램, 파일)이나 하드웨어 자원(팩스, 모뎀, 프린터 공유)을 공유할 수 있도록 도와주는 역할 을 한다.
* Data Base : 특정 조직의 여러 사용자가 공유하여 사용할 수 있도록 통합

해서 저장한 운영 데이터의 집합이다.

본 요구사항 정의서에서 사용된 약어 및 풀이는 다음과 같다.

|  |  |  |
| --- | --- | --- |
| 용어 및 약어 | 풀이 | 비고 |
| DB | DataBase |  |
| JSP | Java Server Page |  |

[Table 1] 약어 및 용어

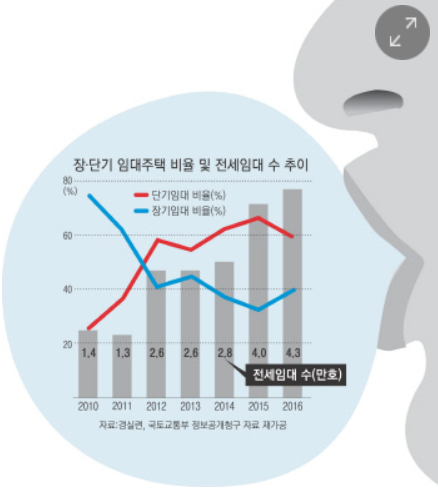
# 2. 제품 소개

본 장에서는 신주민 제품에 대한 간략한 개요 및 필요성을 소개한다.

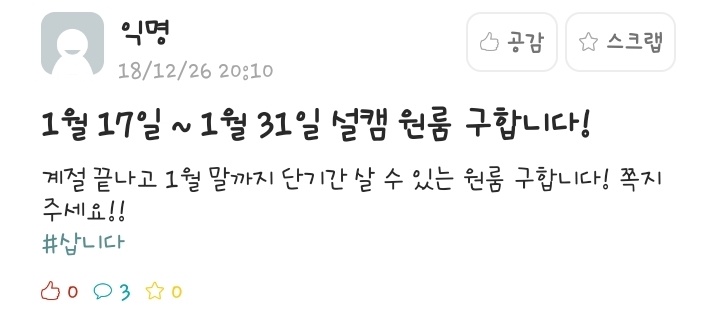
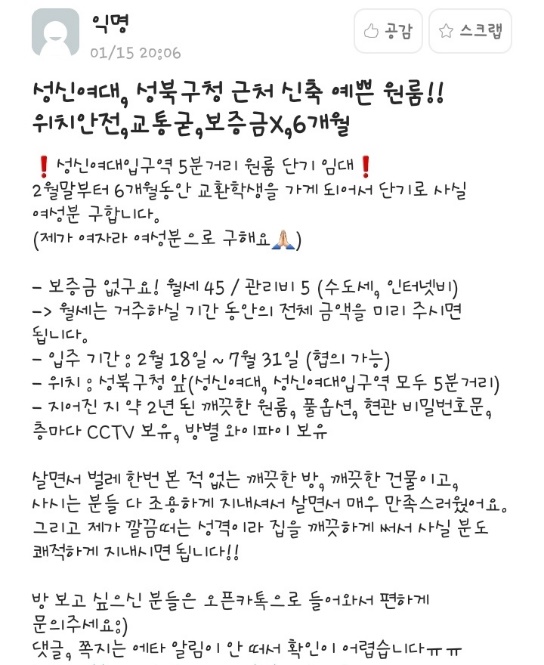
**2.1 제품 개요**

신주민 제품은 중단기 임대 서비스를 제공하여 새로운 주민을 받아들이겠다는 의미를 지니고 있다. 제품을 통해 방에 대한 정보를 제공받음으로써 임대인과 임차인이 보다 확실하고 편리하게 방을 거래할 수 있게 서비스를 제공한다. ‘Airbnb’, ‘여기어때’ 등 숙박업 어플리케이션과는 달리 신주민 제품은 중단기 임대업 어플리케이션으로 중단기 임대 관련 매칭 서비스를 제공한다.

**2.2 제품 필요성**



[그림 1 단기임대 비율]



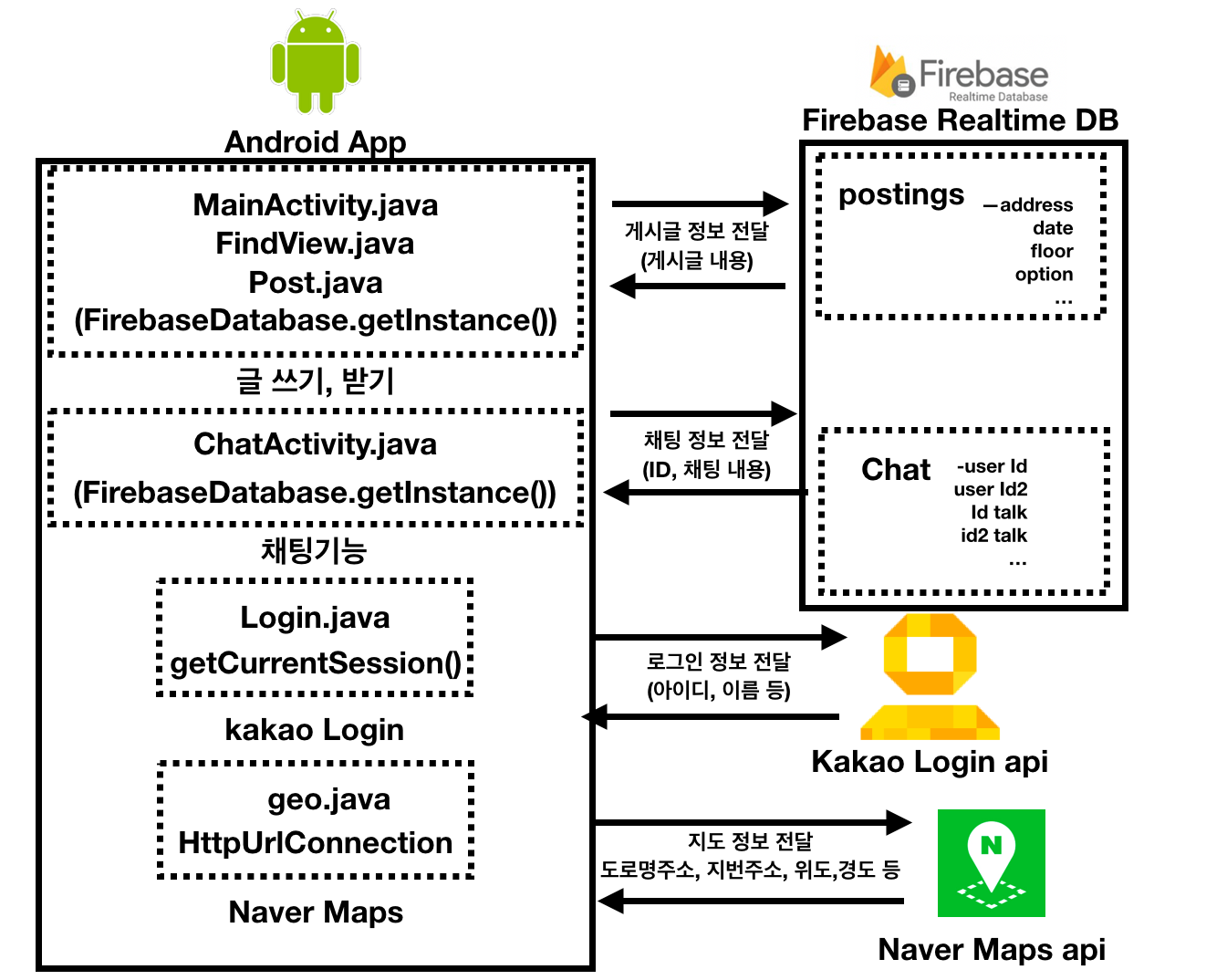
**[그림 2 중단기 임차의 수요]**

국토교통부 정보공개청구 통계에 따르면, 단기임대 비율이 장기임대 비율보다 높아지고 있는데, 대부분 한국에 머물면서 일해야 하는 외국인 노동자, 수능 기간에 공부하기 위해 서울로 올라와 공부해야 하는 수험생, 계절학기를 듣고 싶어하는 대학생, 지방으로 출장을 가야하는 직장인 등이 단기임대를 필요로 하고 있다. 또한 방학동안 고향에 내려가야 하는 자취생, 단기 유학을 떠나는 대학생 또는 직장인 등이 단기임대를 내놓기를 원하고 있다. 또한 현재 에브리타임이나 카페 등을 확인해보면 단기 임대를 원하는 사람들이 많은 것을 확인할 수 있다.

**3. 시스템 구성**

본 장에서는 네이버 지도 API를 활용한 모바일 안드로이드 어플리케이션을 개발하여 학생, 직장인, 외국인 등이 편리하게 집을 구할 수 있는 중,단기 임대 매칭 서비스 ‘신주민’의 시스템을 구성하기 위한 개별 소프트웨어를 기술한다.

3.1 시스템 구성도



[그림 3 시스템 구성도]

앱 내의 게시글, 채팅 기능 등은 Firebase의 RealtimeDB를 활용한다. 이는 별도의 서버 개발 없이 DB를 작성 가능하며 NoSQL의 형태를 띄고있어 관계형이 아니라 Table을 별도로 구성할 필요도 없고json타입의 형태로 저장된다. 또한 카카오 로그인, 네이버 Maps의 지도 정보 전달 기능은 json형태로 정보가 전달된다.

**3.2 소프트웨어**

**3.2.1 안드로이드 스튜디오**



[그림 4 안드로이드 스튜디오]

Android Studio는 Android 앱 개발을 위한 개발환경이다. gradle 빌드를 이용하며 Android Studio는 Android 앱을 개발할 때 효율성을 높여주는 기능들을 제공한다.

**3.2.2 Firebase**

asdf

[그림 5 Firebase]

Firebas는 저렴한 가격에 웹 개발과 모바일 개발에 직접적으로 도움이 되는 기능을 제공하는 백엔드 플랫폼이다. 백엔드 서비스를 이용하여 개발자들이 프로그램을 만들고자 할 때 제공되어 있는 다양한 기능들을 사용할 수 있으며 또한 관리할 수 있는 기능도 제공한다. Firebase RealtimeDB는 실시간으로 연동되며 별도의 서버 개발을 할 필요가 없다는 점이 장점이다.

**3.2.3 Naver Maps API**



[그림 6 네이버 Maps API]

Naver Maps API는 웹 서비스 또는 애플리케이션에 네이버 지도를 활용할 수 있도록 다양한 기능을 제공한다.

**3.2.4 KaKao Login API**



[그림 7 카카오 Login API]

사용자 관리는 카카오 플랫폼 서비스에서 제공하는 핵심 기능 중 하나이다. 사용자 관리는 쉽고 빠른 방법으로 사용자의 계정을 카카오 플랫폼과 연결하며, 로그인과 사용자 정보 요청 등 여러 기능을 제공한다.

**3.3 개발환경**

|  |  |  |
| --- | --- | --- |
| API | Android Application | Server & Database |
| Naver maps | Android studio | Firebase |
| Kakao login API | Java | NoSQL, JSON |
|  |  |  |

[Table 2 개발환경]

**4.제품 기능 및 기능동작**

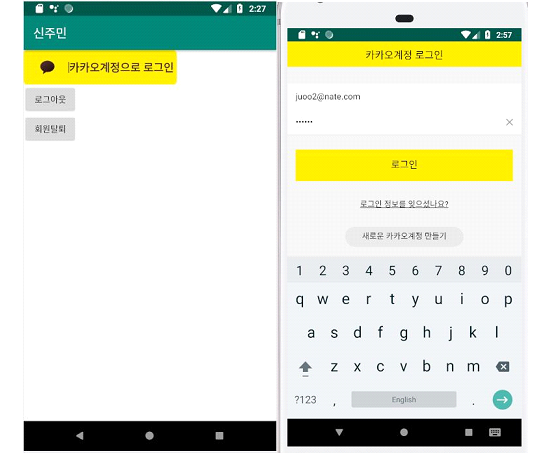
본 장에서는 프로젝트 진행을 위한 기술에 대하여 앱 사용에 대한 UX 순서를 기준으로 기술한다.

**4.1 APP 내 기능**

본 장에서는 APP내부에서의 기능에 대해 기술한다.

**4.1.1 회원가입, 로그인 기능**

로그인은 안전한 매칭을 위하여 가장 먼저 어플리케이션 내부에서 카카오톡 Login api를 통해 서비스를 제공한다. 로그인을 통해서 사용자의 이름과 이메일 등 사용자가 제공하는 것을 동의한 정보를 가져온다.

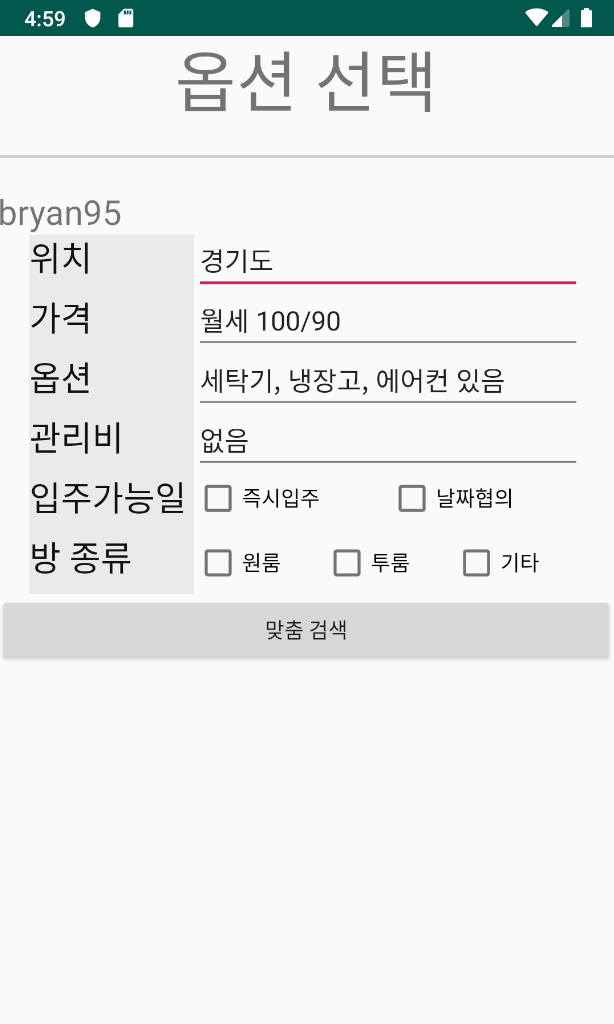
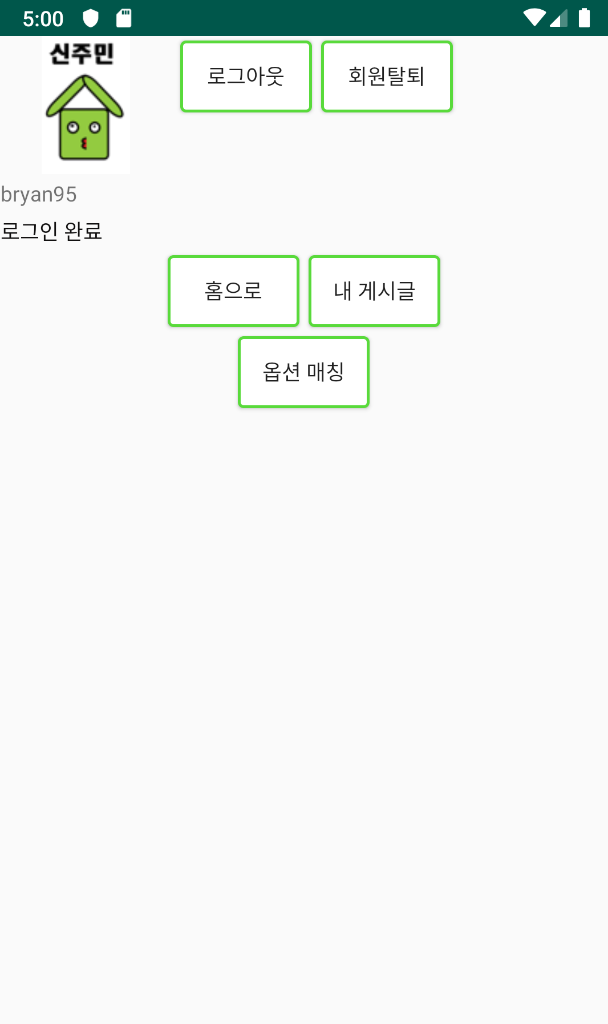


[그림 8 카카오 로그인]

위의 그림은 카카오톡 로그인 api를 애플리케이션에 사용한 그림이다. 로그인을 했을 경우 개인정보 제공 동의에 대해 새로운 창이 뜨고 동의를 해야 로그인이 진행된다. 동의하고 계속하기를 사용자가 누르면 로그인은 완료가 되고, 사용자가 제공에 동의한 항목을 데이터 베이스에 전송하여 저장한다.

4.1.2 option 설정 기능

로그인 후 option을 설정하여 원하는 옵션에 맞는 가구만을 MainActivity에 보여준다. 옵션은 로그인 후 보여지기도 하고, 수정을 원한다면 마이페이지에서 수정이 가능하다.

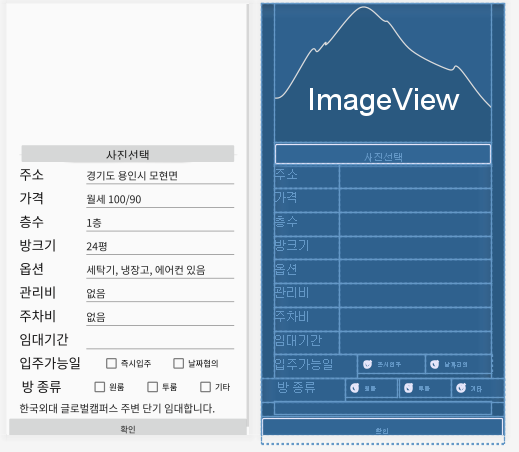
  

[그림 9 옵션 선택 가능]

로그인 시 옵션 선택 창으로 넘어가게 되는데 이때 그림 9에서처럼 bryan95라는 카카오톡 아이디가 나온다. 이때 위치, 가격, 옵션, 관리비, 입주가능일, 방 종류 등을 원하는 대로 선택하여 맞춤 검색을 한다면 MainActivity에 맞춤 검색한 내역이 나오게 된다. 그 후 추가 옵션 변경을 원한다면 마이페이지에서 옵션매칭을 다시 클릭하여 원하는 곳으로 나갈 수 있다.

**4.1.3 글쓰기, 사진 올리기 기능**

홈 화면에서 방 올리기 버튼을 누르면 글쓰기 화면으로 넘어간다. 이때 Activity\_post에서 글 포멧을 받아 사진, 주소, 가격, 층수, 방 크기, 옵션, 관리비, 주차비, 임대기간, 입주 가능일을 입력 받아 Firebase 데이터베이스에 저장한다. 사진 및 파일을 저장하기 위해서는 접근 권한이 필요하므로 권한 허용 동의 유무를 선택해야 사진을 올릴 수 있다. 체크박스에 있는 내용은 String으로 변환하여 데이터베이스에 전송한다.



[그림 10 post.xml]

권한에 동의 후 사진을 갤러리에서 불러오고, 주소, 가격, 층수, 옵션, 관리비, 주차비, 임대기간, 입주가능일, 방 종류에 대한 정보를 입력 후 밑에 간단한 설명을 추가한다. 확인 버튼을 누르면 “게시글을 올리시겠습니까?”가 뜨면서 게시글을 올릴지에 대한 확인 메시지가 Display된다.



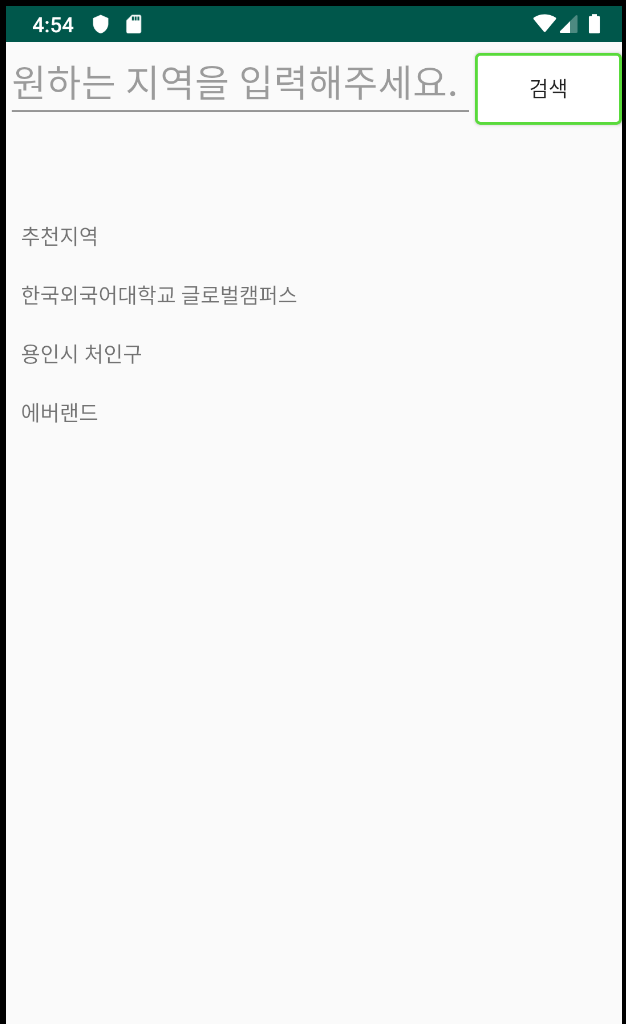
[그림 11 post event]

Activity\_post.xml에 게시글을 작성 후 확인버튼을 누르면 onclicklistener alertdialog 를 통해 게시글을 올릴지를 결정하게 된다. Yes 버튼을 누르면 게시글이 올라가면서 상세 글 보기로 넘어가고, No버튼을 누르면 홈 화면으로 돌아간다.

 [그림 12 사진 권한 요청]

사진을 갤러리에서 불러오기 위해서는 권한 API가 필요하다. Android Permission을 통해 카메라, 저장소 등 기본 앱에서 가지는 일부 기능에 대한 권한을 가져오는 것으로, 사용자가 허용하지 않으면 갤러리에서 사진을 불러올 수 없다.

**4.1.4 검색 기능**

[그림 13 검색]

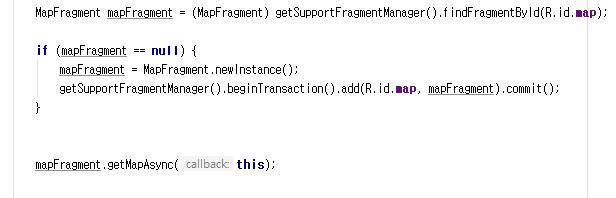
검색, 지도 기능은 FirebaseRealtimeDB안에 json 형식으로 저장된 값들을 기준으로 검색이 된다. FirebaseRealtimeDB안에는 지번 주소, 도로명 주소 등이 포함되어 있음으로 지명이나 도로명 등 어떤 것으로 검색해도 결과값이 나온다.

**4.1.5 Naver Maps 지도 기능**



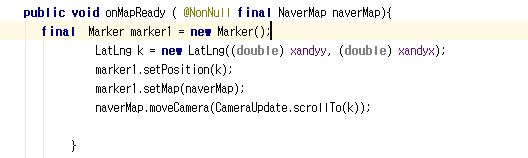
**[그림 14 지도]**

Maps 기능을 사용하기 위해서는 네이버 클라우드 플랫폼에 안드로이드 프로젝트를 등록해야 한다. 위의 그림과 같이 안드로이드 내에서 인터넷 사용을 허가해야 하고, 네이버 클라우드 플랫폼에서 발급받은 클라이언트 ID를 입력해야 한다.



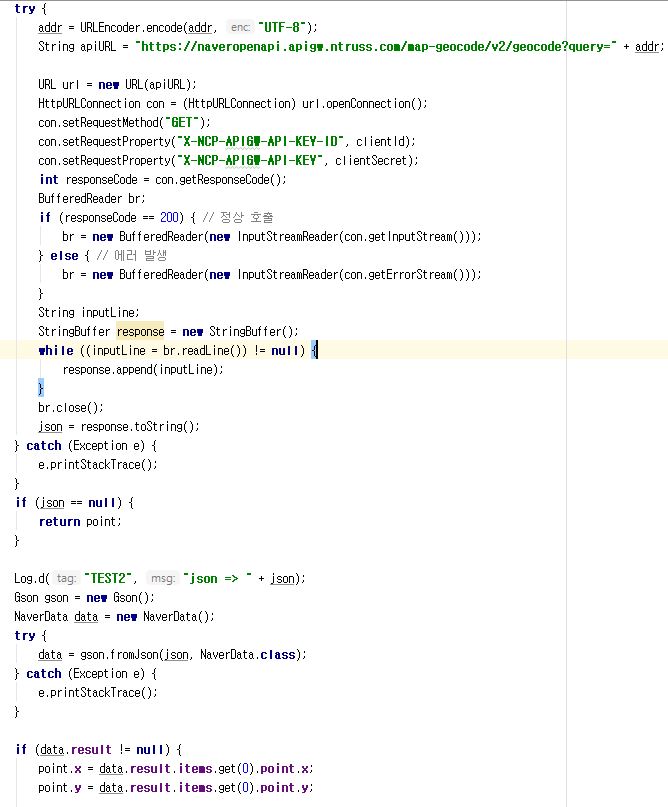
**[그림 15 지도 Pin 찍기 기능]**

앱 내에 지도를 생성하기위해 위의 코드와 같이 Maps API에 내장된 MapFragment를 호출하여 지도객체를 생성해야 한다. 지도 객체를 생성해야만 지도에 관련된 기능을 사용하여 지도에 표시 할 수가 있다.



[그림 16 Marker]

생성한 지도객체에 마커를 사용하여 사용자가 등록한 주소를 정확하게 표시할 수 있다. 마커를 생성하기 위해서는 다음과 같은 코드를 입력해야만 한다. 이 마커 기능은 Geocoding을 통해서 입력된 주소를 좌표로 변환하여 마커를 생성하고 지도에 표시한다.

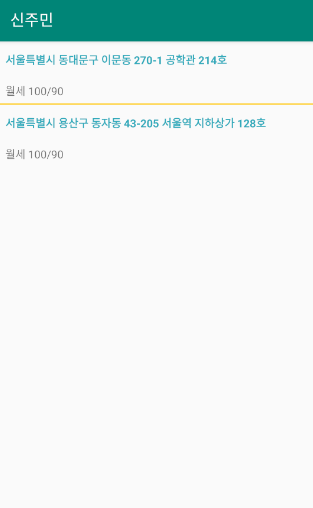


[그림 17 geocoding]

위의 코드는 네이버 Geocoding API를 사용한 코드이다. 지정된 URL에서 GET을 통해서 json으로 이루어진 결과를 얻어온다. 그 후 얻어온 결과를 data라는 함수에 저장한다. Geocoding 기능은 URL에 연결하여 텍스트로 이루어진 주소를 위도와 경도로 변환하는 것이다. 이를 통해서 사용자가 입력한 주소에 마커를 생성하고, 그곳으로 지도의 중심을 이동할 수 있다.

**4.1.6 ListView**

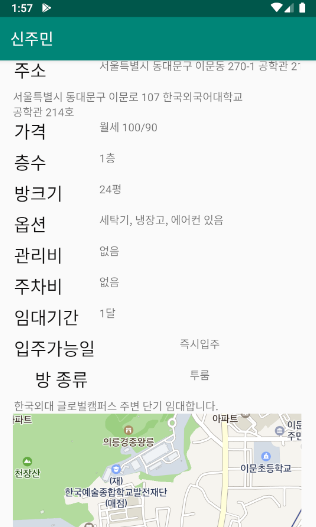
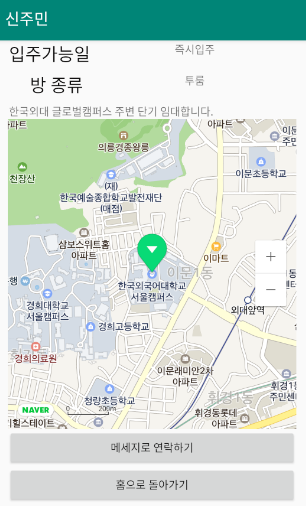
Android SDK의 ListView를 활용하여 검색결과로 나오는 Item들을 Item.xml의 포멧에 맞추어 Display한다. 홈 화면에서의 최근 조회한 방 내역, 상세 글 보기에서의 리뷰 내역을 Display한다. 또한, 검색과 마이페이지의 내가 쓴 글의 내역에서도 ListView가 포함된다.



**[그림 18 ListView]**

**4.1.7 상세 글 보기**

ListView의 글을 누르거나 혹은 글을 작성 한 이후 상세 글 보기를 할 수 있다. 이때 상세 글 보기의 내용들은 DB에서 제목(지번 주소 + 상세 주소)을 기준으로 검색된다. 내부에는 지번주소, 도로명주소, 임대 가능일, 층 수, 관리비, Key(글 생성시 마다 무작위 부여), 옵션, 주차비, 월세, 방 크기, 방 종류(원룸, 투룸 등), 위도, 경도, 작성자 ID, Naver Maps api를 활용한 상세 지도 등이 들어간다.

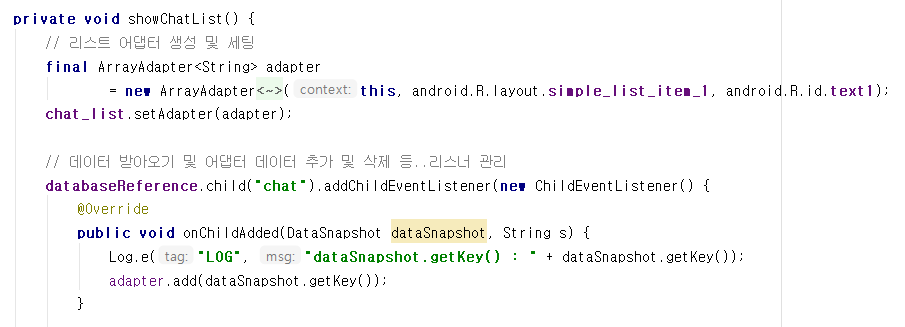
 

**[그림 19 상세 글 보기]**

그림 17을 참고하면 해당 글 작성자와 메시지로 채팅 할 수 있으며, 원하는 방이 아닐 경우 홈으로 돌아갈 수 있다.

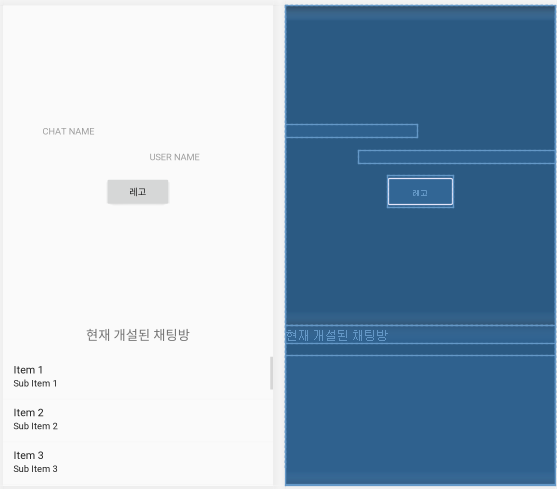
**4.1.8 Chatting 기능**

임대를 원하는 임대인이 게시글을 올린 후 임차인이 그 방을 원하면 연락하기를 통해 매칭을 시도한다. 매칭이 성공할 시 Chat 방이 생겨 실시간으로 대화를 주고받음으로써 정보 교환을 하게 된다. 주고받은 Chat은 firebase 데이터베이스에 실시간으로 저장된다. Chat은 카카오톡의 오픈 채팅방과 같이 여러 명이 대화를 주고받을 수 있도록 되어있고, Chat은 카카오톡 API로부터 받아온 아이디를 사용하여 매칭할 수 있다.



**[그림 20 Chat]**

리스트 어댑터를 생성하여 Chat 데이터를 받아오고, 어댑터 데이터를 추가, 삭제하는 등 리스너를 관리하는 역할을 한다.



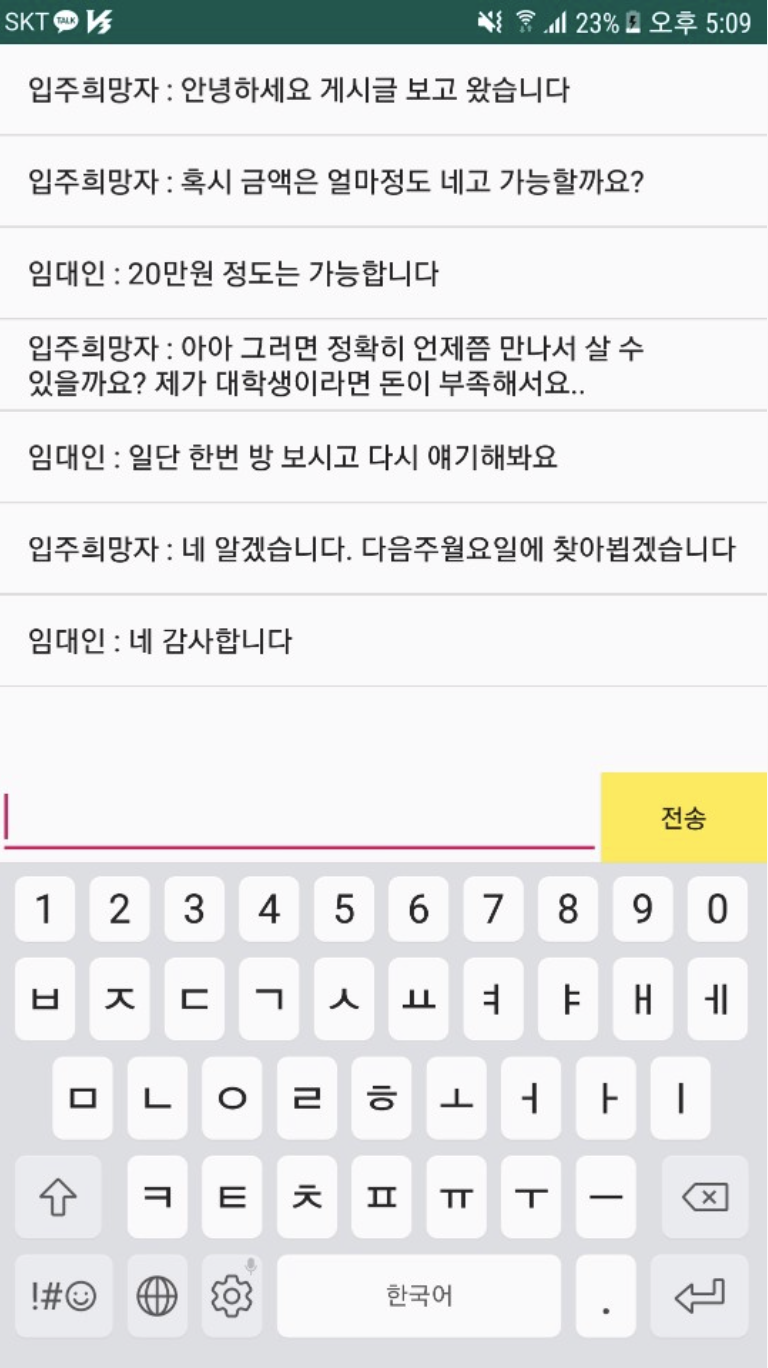
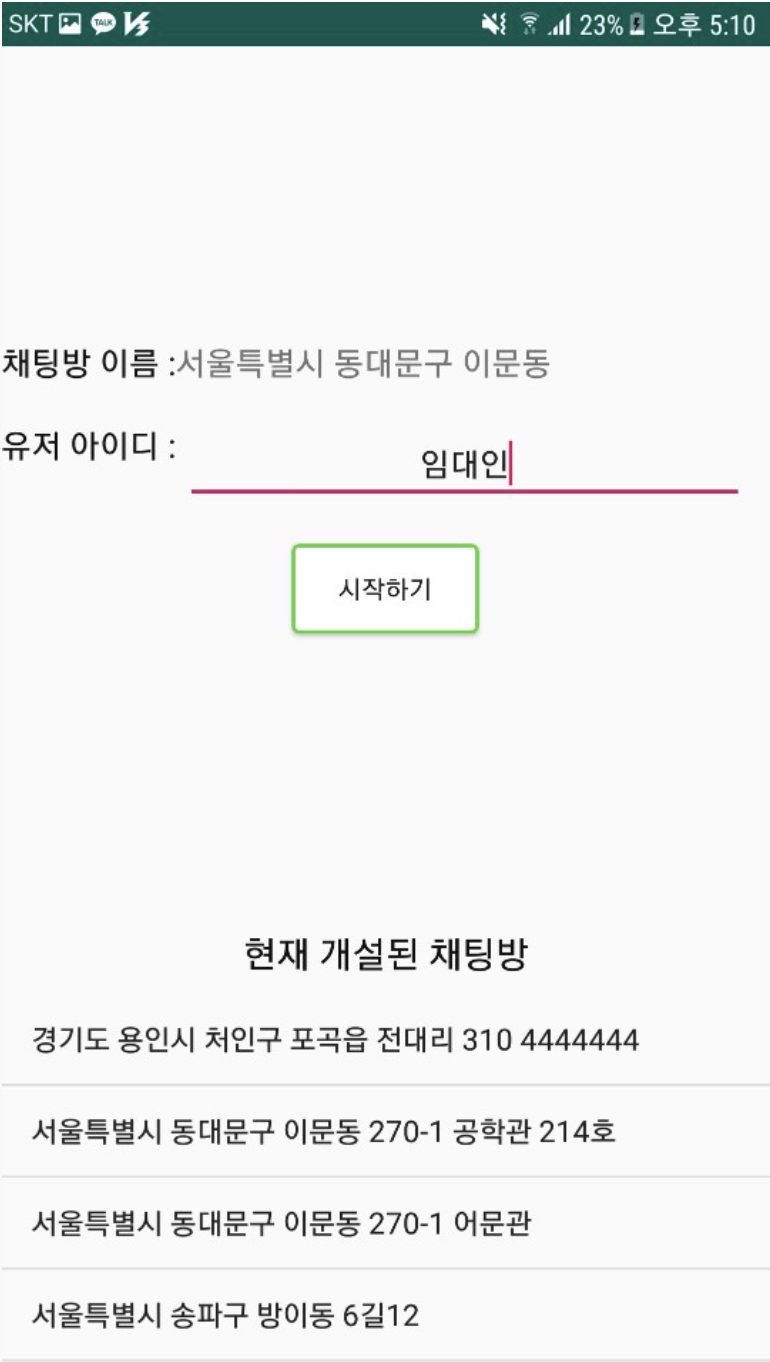
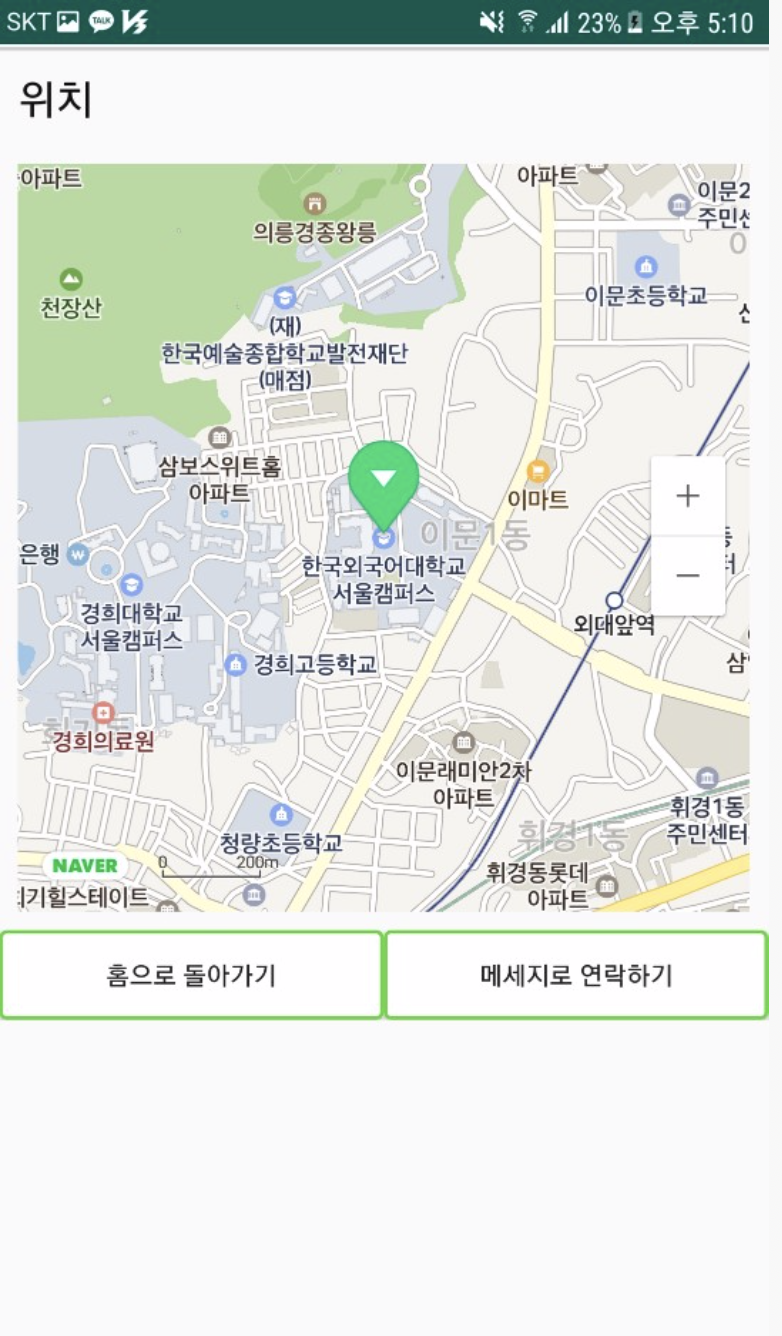
[그림 21 Chat 시작 화면]

Post.java에서 입력한 임대인의 아이디를 String으로 받아와서 채팅방 이름으로 지정하고 카카오 API로부터 받아온 아이디를 유저 이름으로 지정하여 채팅방에 입장한다. 이 때, 하단에는 현재 개설된 채팅방의 목록이 ListView를 통해 Display 된다.



[그림 22 Chat 전송]

Setonclicklistener를 통해 작성된 메시지를 string으로 가져와서 display 하고, 그 데이터는 firebase 데이터 베이스에 실시간으로 저장된다.



**[그림 23 Chat예시]**

다음 예시는 Chat예시를 보여준다. 위치, 조건등이 만족한 사용자는 메시지로연락하기 기능을 이용하여 각 게시글마다 생기는 체팅방에 입장한다. 이후 임대인, 임차인들은 체팅을 통해 가격등을 정하고 만나는 날짜를 정한다.

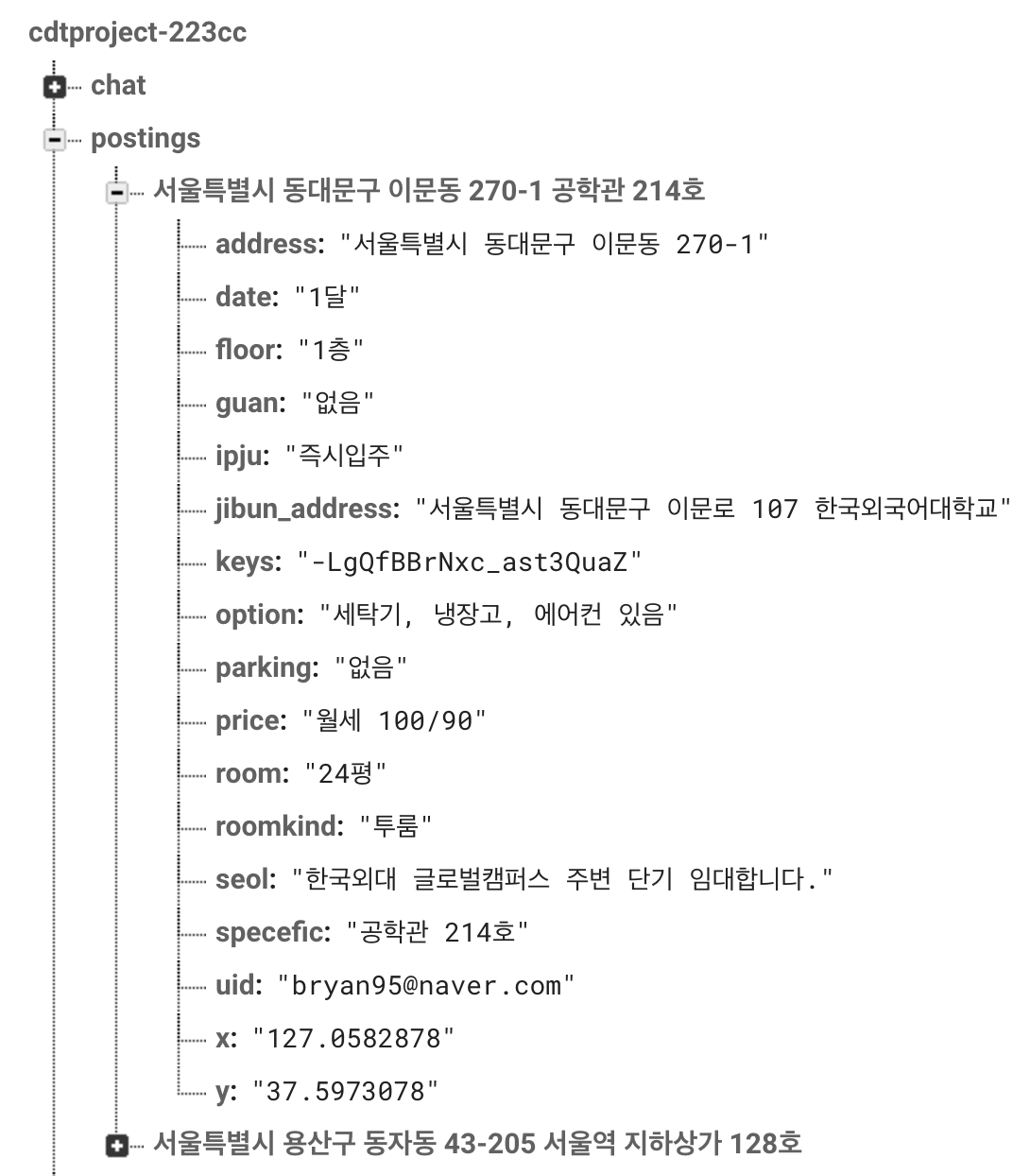
**4.2 Firebase Realtime DB**

**4.2.1 Firebase Realtime DB 데이터 저장 기능**

Firebase Realtime DB의 경우 관계형이 아닌 NoSQL DB를 활용함으로 사용에 따라 유용하게 기능을 넣을 수 있다. 예를 들어 기존의 관계형 테이블에서는 지정된 어트리뷰트에 따라 저장을 해야하고 필요에 따라 원하는 어트리뷰트를 넣지 못한다.

하지만 NoSQL은 유연한 스키마를 제공함으로 필요에 따라 특정 행을 넣을 수 있고, 확장성이 뛰어나며 수평적인 관계로 이루어져 있어 수정, 삭제가 단독으로 이루어져서 편하다.

본 앱에서는 Chat기능과 Post기능에서 NoSQL기반의 Firebase Realtime DB를 사용하고 있다.

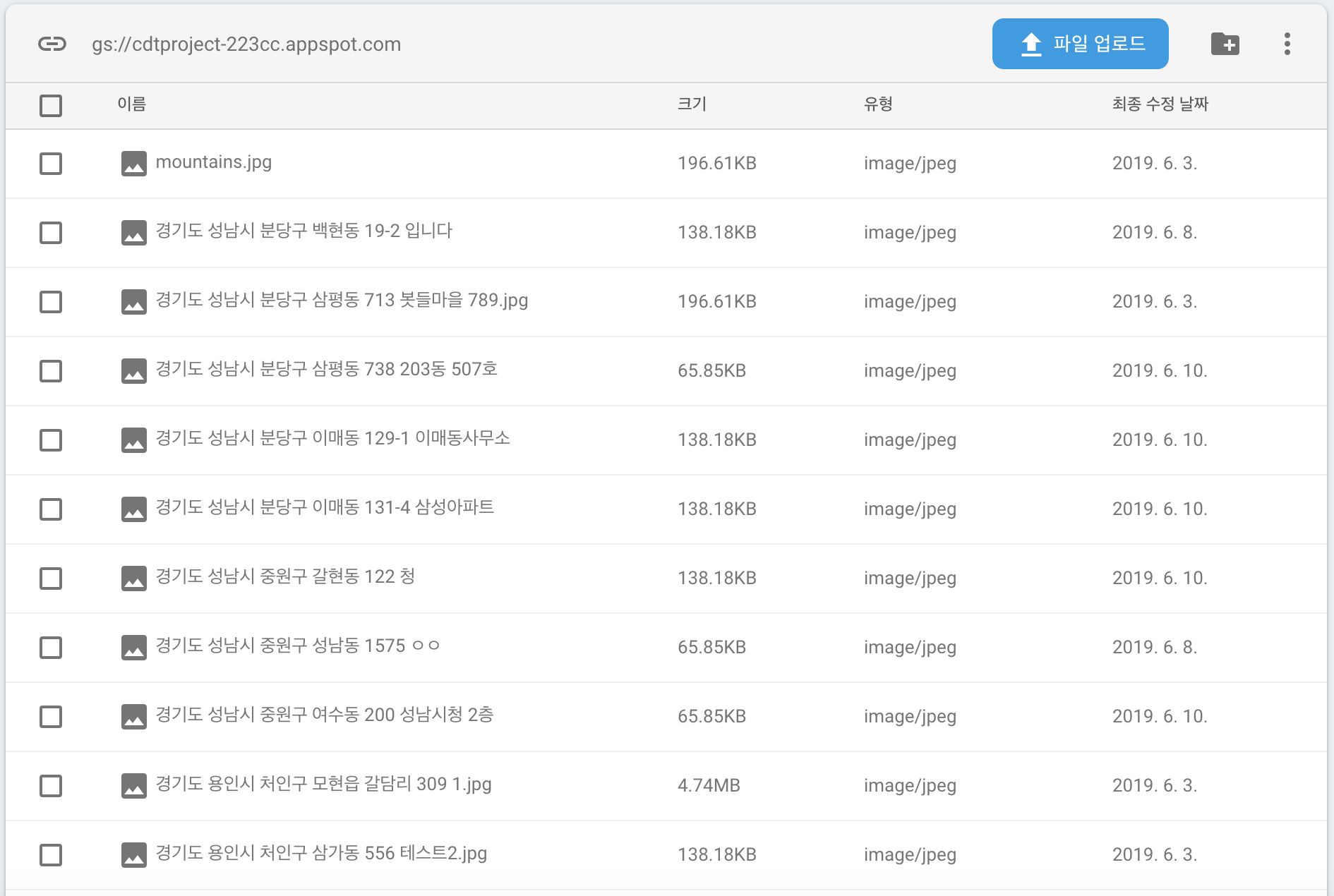


**[그림 24 Realtime DB에 사용된 NoSQL 테이블]**

그림 22에서와 같이 테이블이 비관계형, 수평적 json형태로 저장이 되어있음으로 수정,삭제가 용이하다.

**4.2.2 Firebase Storage**

Firebase Storage를 활용하여 게시글 이미지를 저장하고 이를 활용하여 사용한다. 이때 Post.java에서 포스팅한 데이터는 img\_url이라는 항목에 Firebase Storage에 업로드 된 이미지 주소를 저장한다. 이미지 주소는 지번주소+ 상세주소의 구조로 글 제목과 동일하다.

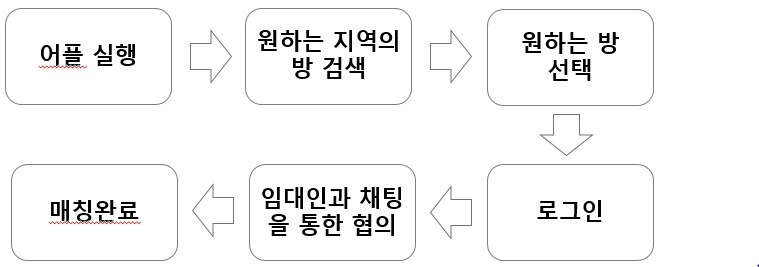


[그림 25 Firebase Storage]

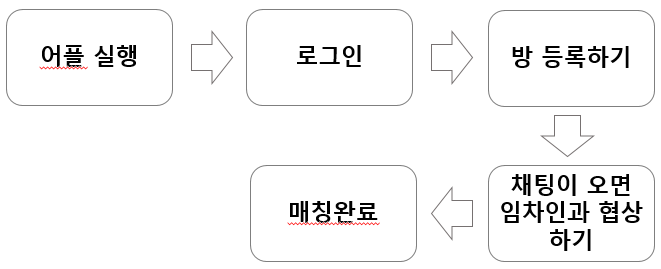
**5. 시스템 시나리오**

**5.1 순차 다이어그램**

<임차인>



[그림 26 임차인 순차 다이어그램]

<임대인>

[그림 27 임대인 순차 다이어그램]

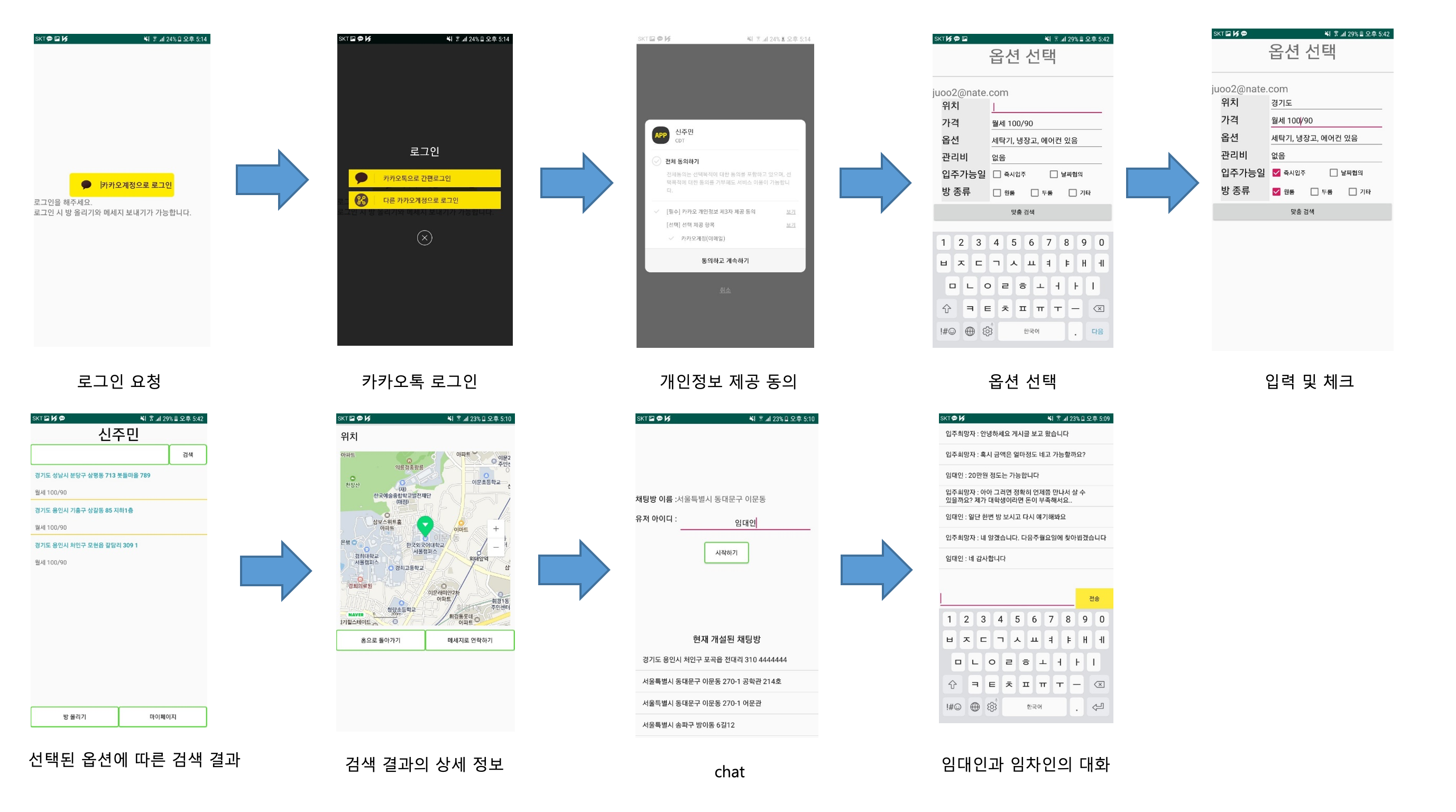
**5.2 시나리오**

<임대인 입장>

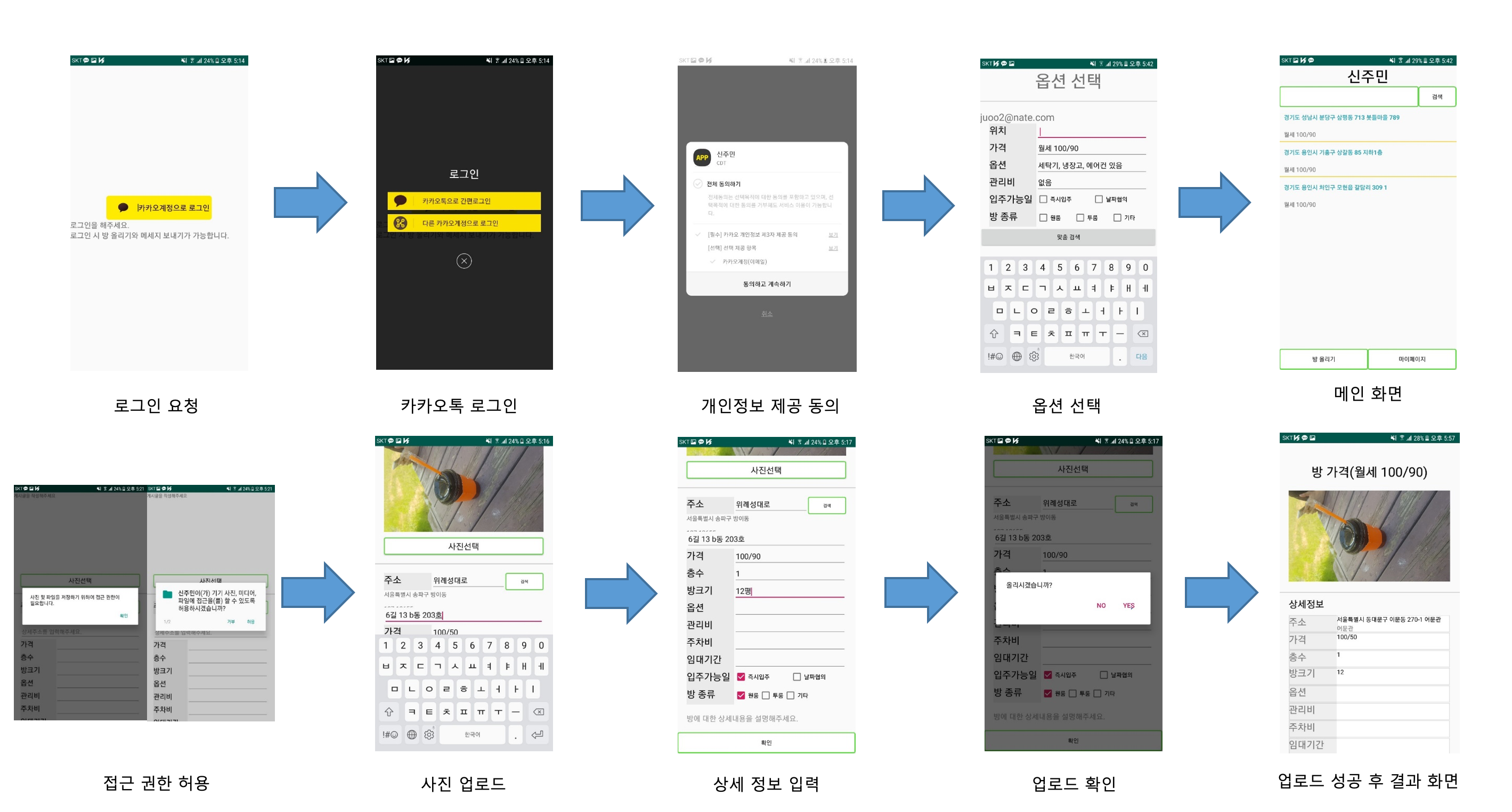
직장인 김씨가 올해 6월 한달 동안 해외로 출장을 가게 되었다. 하지만 현재 살고있는 방이 아직 계약기간이 남아있어서 마음대로 방을 빼지 못하는 상황이다. 이때 신주민 어플리케이션을 통해 출장 기간 동안 방을 임대한다는 글을 올렸다. 며칠 후 임대를 원하는 사람에게 연락이 와서 채팅을 통해 가격, 임대기간 등을 협상한 후 방을 임대해 주기로 하였다. 김씨는 출장 한달 동안의 월세를 낭비하지 않게 되었다.

<임차인 입장>

대학생 이씨가 올해 학기 계절 학기를 듣게 되어 3주간 학교 근처에서 거주할 수 있는 공간이 필요 하였다. 이때 신주민 어플리케이션을 통해 학교 근처의 방을 검색하여 임대인이 올린 글을 보고 임대인에게 연락을 하여 가격과 임대 기간 등을 협상 후 임대 받게 되었다.

****

**[그림 28 글 검색 시나리오]**



**[그림 29 글 작성 시나리오]**

**6. 세부 추진 계획 및 일정**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **구분** | **세부 구분** | **추진내용** | **추진일정(월)** | | | | |  |
| **4** | **5** | **6** | **7** | **8** | **9** |
| 계획 | 개발계획&환경 | 각 기능별 구현 일정&개발환경 |  |  |  |  |  |  |
| 분석 | 기존 방법과 비교 | 기존 공간 대여 앱과 비교 |  |  |  |  |  |  |
| 관련 시장 조사 | 기존 공간 대여 앱 조사 및 특징 조사 |  |  |  |  |  |  |
| 프로젝트 비용 | 프로젝트 개발 전, 서버/인건비 등 지출 비용 조사 |  |  |  |  |  |  |
| 설계 | 시스템 구성도 | 현실적 계획 수립과 분석 완료 후 기반으로 구성도 작성 |  |  |  |  |  |  |
| 개발 | 팀원1  앱 개발(App) | UI & UX, 해상도 최적화 작업 |  |  |  |  |  |  |
| 서버와의 통신 작업 시도 |  |  |  |  |  |  |
| 팀원2  API 활용 | 네이버 지도 API 를 활용하여 서버와의 통신 시도 |  |  |  |  |  |  |
| 서버와 최적화 작업 및 오류 수정 |  |  |  |  |  |  |
| 팀원3  서버 구축(Server) | 개발환경 구축 |  |  |  |  |  |  |
| db 연동을 위한 각종 설정 |  |  |  |  |  |  |
| 팀원 4  DB구축&활용(DB) | 데이터 저장할 DB스키마 구축 |  |  |  |  |  |  |
| 서버 담당과 함께 앱에 push 기능 구축 |  |  |  |  |  |  |
| 테스트 | App | App 내에서 잘 작동 되는지 여부 확인 |  |  |  |  |  |  |
| App서버 통신 | App에서 요청 된 내용&네이버 클라우드 서버에 전송 여부 확인 |  |  |  |  |  |  |
| DB확인 | 올바른 스키마에 들어가는지 확인. 결과 값 확인 |  |  |  |  |  |  |
| 종료 | 최적화&디버깅 | 오류 여부 지속적인 확인& 최적화 시도 |  |  |  |  |  |  |

[Table 3] 개발 계획

**7. 팀원 담당 업무**

|  |  |  |  |
| --- | --- | --- | --- |
| 번호 | 이름 | 역할 | 담당업무 |
| 1 | 이태규 | 팀장 | 전체 프로젝트 관리 및 FirebaseRealtimeDB, 안드로이드 클라이언트 통신(게시글 저장, 검색 등), 옵션 검색 기능 |
| 2 | 류권환 | 팀원 | Android 클라이언트 기능 및 UI 제작(카카오 API, 네이버 지도 API, 검색) |
| 3 | 박혜림 | 팀원 | Android 클라이언트 기능 및 UI 제작(FirebaseDB를 이용한 Chat 기능) |
| 4 | 윤혜선 | 팀원 | Android 클라이언트 기능 및 UI 제작(게시글, 권한 API를 이용한 사진 등록, ) |

[Table 4 담당 업무]

**8. 참고 문헌**

**8.1 github Repository**

<https://github.com/gpfla201/cdt>

**8.2 참고 사이트**

|  |  |
| --- | --- |
| Title | URL |
| Android Chat 기능 | <https://mailmail.tistory.com/44> |
| Firebase 연동 | <https://corikachu.github.io/articles/android/firebase/android-firebase-realtime-chatting-app> |
| Android 사진 불러오기 | <https://jeongchul.tistory.com/287> |
| KakaoDevelopers | https://developers.kakao.com/docs/android/user-management |

[Table 5 참고 사이트]