

요구사항 명세서 ver2

목차

1. 개요
 - 1-1. 결과물에 대한 간단한 설명 (목차 2번 용어 정의 참고)
 - 1-2. 운영 환경
2. 용어 정의
3. 기능적 요구사항 (목차 2번 용어 정의 참고)
4. 비기능적 요구사항 (목차 2번 용어 정의 참고)
5. 제약사항 (목차 2번 용어 정의 참고)
6. firebase 구조
 - 6-1. firebase storage
 - 6-2. firebase database
7. 요구 사항 그룹화
 - 7-1. 기능 상 완전히 똑같은 기능의 id 조합
 - 7-2. 기능 상 똑같지는 않지만 유사한 id 조합
8. 역할 분담

1. 개요

1-1. 결과물에 대한 간단한 설명 (목차 2번 용어 정의 참고)

관심사에 따라서 사람들을 연결해 주는 서비스이다.(식사를 같이 하고 싶은 사람끼리 연결하는 기능으로 나중에 확장할 수도 있다.)

사용자의 위치에 따라서 모임을 추천할 수 있다. (gps나 지도 관련 api 사용 예정)

게시물 기능, 친구 추가, 채팅 방 기능이 있다.

1-2. 운영 환경

운영체제: android

서버(백엔드): firebase

사용 api : 지도 관련 api (구체적으로 어떤 지도 api인지는 아직 정하지 않음)

2. 용어 정의

회원: 회원 가입이 완료된 사람

모임: 2명 이상의 회원들이 모여서 소통하기 위한 방

친구: 친구 추가를 통해서 연결된 회원. 친구로 추가된 회원은 친구 목록을 통해 관리할 수 있다.

대화방: 1:1채팅, 모임에서 대화하기 위한 방

3. 기능적 요구사항 (목차 2번 용어 정의 참고)

***참고사항:**

굵은 글씨 with 기울임 꼴-좀 더 자세히 정의 해야 함

밑줄 - 다른 화면과 일관성을 맞춰야 할 부분

(밑줄 with 분홍색: 서로의 기능에 없는 기능 ,

밑줄 with 나머지 색: 서로의 기능에 있지만 작동 과정이 일관성이 맞지 않는 부분, 단 화면에 표시되는 데이터가 다른 것은 설계가 그런 것이기 때문에 괜찮음)

같은 색의 기능 이름: 똑같거나 유사한 기능

로그인 회원가입 관련

| ID | 구분 | 기능 이름 | 기능 설명 | 우선순위 | 비고 |
|----|----|--------|--|------|--|
| A1 | | 회원가입 | 이메일, 비밀번호, 프로필 사진, 활동지역(보류), 생일,관심사를 사용자에게 입력받아 설정하고 회원가입한다.(지역을 입력 받을 때는 글씨를 입력하면 자동완성 된다.) | 상 | firebase,기본 ui,B5의 solution2 or solution 3를 하기 전까지는 지역정보는 보류 |
| A2 | | 관심사 설정 | 회원가입(A1)에서 관심사를 선택할 수 있는 화면을 제공하여 사용자에게 관심사를 미리 입력받아 설정한다. | 중 | firebase,기본 ui(A1-A2) |
| A3 | | 로그인 | 회원가입(A1)에서 등록된 이메일, 비밀번호를 입력하고 로그인 을 한다. 로그인에 성공하면 해당 계정으로 접속하고 홈화면(B1)을 실행한다. 로그인에 실패하면 토스트 메시지로 표시한다. {페이스북, 구글 로그인, 카카오톡} | 상 | firebase,기본 ui(A3-A4) |
| A4 | | 로그아웃 | 현재 접속한 회원이 로그아웃 버튼을 클릭하면 해당 계정의 접속을 종료한다. 로그아웃에 성공하면 로그인화면(A3)으로 이동한다. (프로필 화면에 배치) | 상 | firebase,기본 ui |
| | | | | | |

홈 화면 관련

| ID | 구분 | 기능 이름 | 기능 설명 | 우선순위 | 비고 |
|----|----|---------|--|------|--------------------------------|
| B1 | | 관심 카테고리 | navigationbar fragment : HOME 홈 화면에 본인이 설정한 관심있는 카테고리 아이콘을 확인할 수 있다. (사용자가 원하는 카테고리만 배치) 아이콘 클릭 시 세부 카테고리 화면(C1)으로 넘어가게 된다. | | firebase, 기본 ui, navigationbar |
| | | | | | |

| | | | | |
|----|--|----------------|--|--|
| B2 | | 카테고리 설정 | 기본 카테고리 중 에 관심 카테고리를 B1에 추가 및 삭제를 할 수 있다. | 수량제한 없음 전체 카테고리 수 만큼 _____firebase, 기본 ui |
| B3 | | <u>게시물 리스트</u> | navigationbar fragment : HOME 관심사 설정에서 적용된 카테고리들과 관련된 게시물(모임)을 보여준다. 게시물 클릭시 B4의 activity가 시작되어서 게시물의 내용을 볼 수 있다. 게시물 생성 버튼을 누르면 B12가 실행된다. 각 게시물의 모임 참여 버튼을 누르면 그 모임은 D2에서 확인할 수 있다.(이미 참여한 회원은 버튼이 비활성화된다.) | firebase, recyclerview, intent, navigationbar(B1-B2-B3) |
| B4 | | <u>게시물</u> | 모임의 구체적인 정보(모임 관련 이미지, 인원, 게시물 내용(모임시간, 회비, 가입요건, 안내사항))를 확인할 수 있다. 참여멤버를 recyclerview로 확인할 수 있다. _recyclerview의 요소에 있는 버튼을 추가하면 그 사람을 친구 추가할 수 있다. 관리자(게시물을 처음에 만든 사람)와 대화할 수 있는 대화버튼, 모임에 가입 신청을 할 수 있는 가입버튼이 있다. 게시물에 대한 소개글이 있고 navigationbar가 존재하여서 그 모임에 대한 게시판(채팅)도 볼 수 있다. 게시판에서는 좋아요 및 댓글을 작성할 수 있다. 모임 정보, 게시판, 채팅 기능 있음 | 모임 참여 버튼을 누르면 그 게시물은 D2에서 확인할 수 있다.(이미 참여한 회원은 버튼이 모임탈퇴 버튼으로 바뀐다.) 관리자는 다른거 만든 후에 여유 있으면 실시하는 것으로 친구 추가 기능은 우선순위가 중이므로 나중에 한다. _____ firebase, recyclerview, intent, navigationbar |

| | | | | | |
|----|--|--------------------|--|---|--|
| B5 | | <u>내 주변 소모임 추천</u> | <p>navigationbar fragment : MAP</p> <p>지도에 내 주변 소모임(내 관심사)들을 아이콘(소모임 프로필사진)으로 확인할 수 있다. 아이콘 아래에는 카테고리 이름을 볼 수 있다. 아이콘을 클릭할 시 B4로 이동한다.</p> <p>navigationbar의 요소를 클릭하면 현재 activity에서 fragment만 바뀌면서 본인 위치와 가까운 순으로 B12에서 업로드한 모임의 리스트를 표시하고 지도 상으로 마커를 통해서도 표시한다. (현재 카테고리에 관련된 모임만 표시한다.) (문제 1: 모임에 여러 명이 있는데 그 모임의 위치는 어떻게 특정할 것인가? 제안 1: B12에서 업로드한 위치를 참고) solution1: 현재 위치와 가까운 순으로 소모임 추천(위도, 경도) solution2: 시 군 구 정보로 검색해서 소모임 추천 solution3: 관심 지역(위도, 경도)을 입력받은 것과 가까운 순으로 추천</p> | | firebase, 지도 api, navigationbar, recyclerview solution2부터 하기 |
| B6 | | 대화방 | <p>navigationbar fragment :</p> <p>CHAT 기능 D2</p> | | D2참고 |
| B7 | | 프로필 | <p>navigationbar fragment :</p> <p>PROFILE 내 프로필을 재설정(사진, 자기소개, 개인정보 등)할 수 있다, 로그아웃, 내 모임 관리 기능 친구목록 확인 : D1</p> | | firebase, 기본 ui |
| B8 | | 검색기능 | 소모임 이름이나 | 중 | |

| | | | | | |
|-----|--|--------|---|---|--|
| | | | 카테고리 이름을 활용하여 모임 검색을 할 수 있다. 검색된 모임 클릭 시 B4이동 | | |
| B9 | | 알림기능 | 알림 아이콘을 클릭하여 내 모임의 최근 소식이나 공지사항을 확인할 수 있다. | 중 | |
| B11 | | 친구 목록 | navigationbar의 요소를 클릭하면 D1이 작동한다. | 중 | 다른 거 개발한 후에 여유되면 실시 |
| B12 | | 게시물 생성 | 모임과 관련된 게시물을 생성한다. 글 제목과 이미지, 글 본문을 업로드 할 수 있다. 또한 모임의 장소도 업로드 가능하다. 모임의 장소가 업로드 되지 않았을 경우에는 업로드가 되지 않는다. (모임의 장소는 지도 상에서 클릭 또는 텍스트 검색으로 위도와 경도를 특정한다.) | | firebase, 기본 ui, 지도 api, B5의 solution2 or solution 3를 하기 전까지는 지역정보는 보류 |

세부 카테고리 화면 관련

| ID | 구분 | 기능 이름 | 기능 설명 | 우선순위 | 비고 |
|----|----|----------------|--|------|--|
| C1 | | <u>게시물 리스트</u> | navigationbar의 요소를 클릭하면 현재 activity에서 fragment만 바뀌면서 현재 세부 카테고리에 관한 게시물의 리스트를 보여준다. 각 게시물은 해당 카테고리와 관련된 모임이다. 각 게시물을 클릭했을 때 C2가 실행된다. 게시물 생성 버튼을 누르면 C3이 실행된다. 각 게시물의 모임 참여 버튼을 누르면 그 모임은 D2에서 확인할 수 있다. (이미 참여한 회원은 버튼이 비활성화 된다.) | | B3참고 |
| C2 | | <u>게시물</u> | 모임의 구체적인 정보(모임 관련 이미지, 인원, 게시물 내용(모임시간, 회비, 가입요건, 안내사항))를 확인할 수 있다. 참여 | | 모임 참여 버튼을 누르면 그 게시물은 D2에서 확인할 수 있다.(이미 참여한 회원은 버튼이 모임탈퇴 버튼으로 바뀐다.) 관리자 |

| | | | | | |
|----|--|-------------------------|--|---|--|
| | | | <p>멤버를 recyclerview로 확인할 수 있다.</p> <p>recyclerview의 요소에 있는 버튼을 추가하면 그 사람을 친구 추가 할 수 있다. 관리자(게시물을 처음에 만든사람)와 대화 할 수 있는 대화버튼, 모임에 가입신청을 할 수 있는 가입버튼이 있다. 게시물에 대한 소개글이 있고 navigationbar가 존재하여서 그 모임에 대한 게시판(채팅)도 볼 수 있다. 게시판에서는 좋아요 및 댓글을 작성할 수 있다.</p> | | <p>는 다른거 만든 후에 여유 있으면 실시하는 것으로 친구 추가 기능은 우선순위가 중이므로 나중에 한다._____B4참고</p> |
| C3 | | 게시물 생성 | <p>모임과 관련된 게시물을 생성한다. 글 제목과 이미지, 글 본문을 업로드 할 수 있다. 또한 모임의 장소도 업로드 가능하다. 모임의 장소가 업로드 되지 않았을 경우에는 업로드가 되지 않는다.(모임의 장소는 지도상에서 클릭 또는 텍스트 검색으로 위도와 경도를 특정한다.)</p> | | B12참고 |
| C4 | | <u>본인 위치와 가까운 모임 추천</u> | <p>navigationbar의 요소를 클릭하면 현재 activity에서 fragment만 바뀌면서 본인 위치와 가까운 순으로 C3에서 업로드한 모임의 리스트를 표시하고 지도상으로 마커를 통해서도 표시한다.(현재 카테고리에 관련된 모임만 표시한다.) (문제 1: 모임에 여러 명이 있는데 그 모임의 위치는 어떻게 특정할 것인가? 제안 1: C3에서 업로드한 위치를 참고)</p> | | B5참고 |
| C5 | | 친구 목록 | <p>navigationbar의 요소를 클릭하면 D1이 작동한다.</p> | 중 | 다른 거 개발한 후에 여유 되면 실시 |
| C6 | | 대화방 | <p>navigationbar의 요소를 클릭하면 D2가 작동한다.</p> | | B6참고 |
| C7 | | 프로필 | <p>navigationbar fragment : PROFILE</p> <p>내 프로필을 재설정(사진, 자기소개, 개인정보 등)할 수 있다, 로그아웃 친구목록 확인 : D1</p> | | B7참고 |
| C8 | | 검색기능 | <p>소모임 이름이나 카테고리</p> | 중 | |

| | | | | | |
|----|--|------|--|---|--|
| | | | 리이름을 활용하여 모임 검색을 할 수 있다. 검색된 모임 클릭 시 B4이동 | | |
| C9 | | 알림기능 | 알림 아이콘을 클릭하여 내 모임의 최근 소식이나 공지사항을 확인할 수 있다. | 중 | |

공통 요소

| ID | 구분 | 기능 이름 | 기능 설명 | 우선순위 | 비고 |
|----|----|-------|--|------|-----------------------|
| D1 | | 친구목록 | 친구 관계가 맺어진 사람들의 리스트를 보여준다. 친구 리스트의 요소를 클릭하면 D3기능이 작동한다. | 중 | |
| D2 | | 대화방 | 모든 대화방을 리스트로 볼 수 있다. 모임과 관련된 대화방을 클릭하면 C2로 이동하고 1대1 대화와 관련된 대화방을 클릭하면 D3으로 이동한다. | | firebase,recyclerview |
| D3 | | 채팅 | 1대1로 채팅을 할 수 있다. | 중 | |
| D4 | | | | | |

4. 비기능적 요구사항 (목차 2번 용어 정의 참고)

5. 제약사항 (목차 2번 용어 정의 참고)

6. firebase 구조

6-1. firebase storage

| 폴더 명 | 사진 이름 |
|--|-------------------|
| 프로필 사진{ user_profile_image } | 회원의 uid.jpg |
| 번개 모임 베너 이미지{ lighting_meeting_room } | 번개 모임의 문서 id.jpg |
| 게시판 이미지{ posting_image } | 게시판의 문서 id.jpg |
| 주기적 모임 베너 이미지 { periodic_meeting_room } | 주기적 모임의 문서 id.jpg |
| 대회 베너 이미지{ competition_room } | 대회의 문서id.jpg |
| 장소 대여 베너 이미지{ place_rental_room } | 장소 대여의 문서 id.jpg |

6-2. firebase database

(필드 이름은 작성하거나 변경하면 노선에 정리, 종괄호 빨간색으로 표시함)

(기본키는 파랑색, 외래키는 초록색임)

(필드의 값의 유형은 디폴트는 string이고 혹시 string이 아니어야 할 경우에는 별도로 작성 바랍니다. ex>댓글 문서 id 리스트(array){comment_id_list} → string이 아니고 array여서 괄호 안에 보라색으로 표시함)

공통 필드(periodic_meeting_room,lighting_meeting_room,competition_room,place_rental_room){

```
var title:String,var max:String,var info_text:String,
var writer_uid:String,var address:String,
var category: String,var upload_time:Long,var posting_id_list:ArrayList<String>,
var chatting_id_list:ArrayList<String>
```

}

| 컬렉션 id | 문서 id | 필드(기본키-파란색, 외래키-초록색) |
|-----------------------------------|--------------------|--|
| 회원(user) | firebase auth의 uid | 회원 id(firebase auth의 uid){uid}, 닉네임{nickname}, 관심사{interest_array}(여러개 허용 array), 이메일(email), 속해있는 모임의 리스트(array){meeting_room_id_list}, 내가 만든 모임(관리자 기능 관련), 활동 지역{location}, 활동지역위도(number){positionx}, 활동지역경도(number){positiony}, 생일{birthday},프로필 메시지{profile_message}, 회원 정보 수정 시간(number){edit_time},등록한 장소의 리스트(array){place_id_list},예약을 신청한 리스트(array){reservation_list}, 속해있는 번개모임의 리스트(array){lightmeeting_room_id_list},신청한 대회의 리스트(array){competition_id_list},등록한 대회의 리스트(array){competition_create_id_list} |
| 주기적 모임 {periodic_meeting_room} | 자동 설정 | 모임 이름{title},정원{max}, 표지 게시물 본문(info_text),작성자 uid(관리자 기능 관련){writer_uid}, 모임 참여자 uid 리스트(array){member_list}, 모임 장소{address}, 지역, 소속 카테고리{category}, 작성 시간(number){upload_time}, 게시판의 문서 id 리스트(array){posting_id_list}, 채팅 리스트(array){chatting_id_list},모임시작시간(number){start_time}, 모임종료시간(number){end_time},위도(number){positionx}, 경도(number){positiony} |
| 번개 모임 {lighting_meeting_room} | 자동 설정 | 모임 이름{title},정원{max}, 표지 게시물 본문(info_text),작성자 uid(관리자 기능 관련){writer_uid}, 모임 참여자 uid 리스트(array){member_list}, 모임 장소{address}, 지역, 소속 카테고리{category}, 작성 시간(number){upload_time}, 게시판의 문서 id 리스트(array){posting_id_list}, 채팅 리스트(array){chatting_id_list}, 모임시작시간(number){start_time}, 모임종료시간(number){end_time},위도(number){positionx}, 경도(number){positiony},날짜(String){date} |
| 대회{competition_room} | 자동 설정 | 공정리뷰 수(number){num_of_positive},대회 이름{title}, 대회 참가 정원{max}, 게시물 본문(info_text), 작성자 uid(관리자 기능 관련){writer_uid}, 대회 참여자 uid 리스트(array){member_list}, 대회 장소{address}, 지역, 소속 카테고리{category}, 작성 시간(number){upload_time}, 게시판의 문서 id 리스트(array){posting_id_list}, 채팅 리스트(array){chatting_id_list}, 대회시작시간{start_time}, 대회종료시간{end_time},공식비공식분류{type},대회장소{location},위도(number){positionx},경도(number){positiony},날짜(String){date} |
| 장소 대여 {place_rental_room} | 자동 설정 | 공정리뷰 수(number){num_of_positive},대여장소명{title},정원{max}, 본문(info_text),작성자 uid(관리자 기능 관련){writer_uid}, 모임 참여자 uid 리스트(array){member_list}, 업체 주소{address}, 지역, 소속 카테고리{category}, 작성 시간(number){upload_time}, 게시판의 문서 id 리스트(array){posting_id_list}, 채팅 리스트(array){chatting_id_list}, 모임시작시간(number){start_time}, 모임종료시간(number){end_time},위도(number){positionx},경도(number){positiony}, 장소 예약 리스트(array){reservation_id_list}, 오시는길{route}, 장소 상세주소(String){addressdetail}, 문서id{document_id} |
| 장소 예약{reservation} | 자동 설정 | 장소 uid{place_uid},대여장소명{place_name},예약자uid(예약하는사람의uid){uid}, 예약 시작시간{reser_start}, 예약 종료시간{reser_end}, 인원수(number){num_of_people}, 승인 완료 여부(boolean){ok_check},예약 신청시간(number){reser_time} |
| 지역 {location} | 시 id → 구 id | 구이름이 필드 이름, 그 필드의 데이터타입이 array(모임리스트){모임 리스트가 너무 방대해 지면 모임 컬렉션에 지역 필드로 대체한다. } |
| 카테고리 | 운동 | 세부종목이름= 필드 이름, 필드의 데이터 타입은 array(해당 카테고리에 속하는 모임 리스트)-탐색 성능 이점을 위해서 |
| 게시판{posting} | 자동 설정 | 작성자uid{writer_uid}, 제목{title}, 내용(사진){image},글 본문(text), 작성시간(number){upload_time}, 댓글 문서 id 리스트(array){comment_id_list} |

| | | |
|-------------|-------|---|
| 댓글{comment} | 자동 설정 | 작성자 uid(눌렀을때 프로필 보이게 하는 기능으로 확장가능){writer_uid}, 작성 시간(number){upload_time}, 댓글 내용(comment_text) |
|-------------|-------|---|

수정사항

- 회원 collection에 프로필 메시지 필드 추가(모임에 가입한 사람들을 표시할 때 사용할 목적임)
- 필드 이름 정의(중괄호 빨간색)
- 모임 ,게시판 , 댓글 카테고리 구조 일부 변경

코딩 참고 사항

- posting 컬렉션의 작성시간은 number 타입으로 해야 함(현재시간을 milli second로 올리면 될듯) 그래야 업로드 순서에 따라 정렬할 수 있음
- 만약 firebase storage의 user_profile_image만 변경될 때는 user 컬렉션의 edit_time을 현재 시간으로 변경해 주면 변경된 프로필 이미지를 B4의 게시판에서 반영할 수 있게 함

8. 역할 분담

A1~A3 로그인 회원가입 난이도: 하(재웅)

B1,B3(fragment), c1 홈화면 관심사 및 게시물 리스트 난이도: 중 (동경)

B2(activity) 카테고리 설정 난이도: 하(재웅)

B4(activity) 게시판 난이도: 중 (현준)

B7(fragment)프로필 난이도: 중 (범학)

=====

B12(기타 게시물 정보) 난이도 : 난이도: 중

//B6(D2){fragment}, 대화방 난이도: 중

B5(지도 관련){난이도: 상},B12(지도 관련){난이도: 상}