
사용자 인터페이스 및 실습

프로젝트 최종 보고서

<https://youtu.be/95UrH8PMOtU>

12 월 13 일

익담윤 팀

문준익, 이상윤, 신형담

목차

- 1) 최종계획안 Task 표
- 2) 12 주차 프로젝트 진도보고서
- 3) 13 주차 프로젝트 진도보고서
- 4) 14 주차 프로젝트 진도보고서
- 5) 최종보고서

1) 최종계획안 Task 표

어플 최초 기획 단계의 Task 표 입니다. 보고서를 읽어 가시면서 어떻게 발전했는지 보실 수 있게 구성 하였습니다

				난이도
DB (담당자 문준익)	로그인	로그인 데이터 처리	로그인 정보로 서버에서 유저 정보 가져오기	2
		로그인 화면 만들기	로그인 레이아웃 만들기	2
	회원가입	회원가입 데이터 처리	회원가입시 입력한 정보를 서버에 저장	3
		회원가입 화면 만들기	회원 가입 화면 만들기	3
	프로필	프로필 데이터 처리	서버에 저장되어있는 유저정보 가져와서 메인 화면에 보여주기	3
			유저가 작성,수정한 프로필 정보 서버에 저장	3
		프로필 작성 화면 만들기	프로필 레이아웃 구성	3
	오늘운동	루틴 데이터 가져오기	저장소에 선택된 내루틴 의 데이터를 가져오기	4
			그 루틴의 데이터를 읽고 안한 부분의 데이터를 파싱해서 화면에 띄워주기	4
			운동끝난후 오늘한 운동의 데이터 다시 업데이트	3
	게시판	게시판 루틴 데이터 처리	서버 루틴 목록에 있는 루틴들을 카테고리별 추천순으로 가져오기	3
			사용자가 루틴을 새로 만들어서 등록하면 종합해서 서버에 저장	5
			루틴 저장 눌렀을때 유저 저장소에 저장	2
게시판구현 (담당자 이상윤)		게시판	게시판 레이아웃 구성	4
			태그의 맞는 게시글 카테고리별 추천순으로 화면에 띄워주기	3
			사진,루틴설명 등의 세부내용 게시판에 전시하기	3
			운동 장소, 난이도 시간 대상등의 정보를 아이콘으로 표현	3
			게시글에 사진 첨부하기	4
		게시글 업로드	설정한 루틴 다이어리에 띄워주기	5
			운동 루틴 입력 받는 플랫폼 구현	5
			서버에 데이터 전송할수있게 가공	3

오늘의 운동 (담당자 신형담)		오늘의 운동화면	오늘의 운동 레이아웃 구현	3
			저장 버튼 눌렀을시 다이어리와 동기화	5
			DB에서 가져온 지금까지 해왔던 루틴 정보 화면에 출력	2
운동 다이어리		운동다이어리	다이어리 레이아웃 구성	4
			캘린더에 해당 날짜에 한 운동운동 조회	4
			DB에서 가져온 그전에 했던 운동들도 볼수있게 구성	4

추가구현 예상			가능성
	게시판 댓글기능		70%
	비포 애프터 - 포토후기	루틴을 수행해본 유저들의 포토후기	60%
	별점,만족도 평가, 좋아요 구현	사람들이 루틴 게시판을 보고 별정보고 루틴선택가능	80%
	작성자 좋아요 수 연동	좋아요 많이 받은 작성자->왕관(네임드)지정후 신뢰감 부여	70%
	운동 설명 기능	해당 운동이름을 꼭 누르면 작성자가 달아준 링크로 이동 할 수있음	20%

2) 12 주차 프로젝트 진도보고서

1. 팀원별 작업 내역

<문준익 작업내용>

(11 월 첫째주)

11/5~6

Test 프로젝트 따로 생성 하여 Firebase 와 연결파악, 데이터 전송 구조 파악

메인 작업용 프로젝트 YourHealth 생성

Firebase 와 YourHealth 프로젝트 연결

인트로 레이아웃 구성 11/6

로그인 레이아웃 구성 11/6

회원가입 레이아웃 구성 11/6

(11 월 둘째주)

비밀번호 찾기 레이아웃 구성 11/8

프로필 레이아웃 구성 11/9

어플 전체 적용할 메인 글꼴 적용 11/10

회원가입시 버튼 이벤트 구현

(모든 정보가 입력 되어야 버튼이 켜짐)

11/13~14

파이어 베이스(Authentication)이용

이메일,비밀번호를 이용하여 서버와 연결,

회원가입, 로그인 기능 구현완료

(11 월 셋째주)

프로필 레이아웃 구성,화면 이벤트 구현 11/15

갤러리 접근하여 사진 가져오기

유저 정보를 다룰 프로필 클래스 구성

파이어 베이스 (Cloude Firestore) 이용하여 프로필 정보 서버에 저장 11/16~17

파이어 베이스 (Storage) 이용하여 유저 프로필 사진정보 서버에 저장

프로필 레이아웃중 유저 프로필 사진 띄워주는 View 변경(Glide API 이용)11/18

지금까지 만든 레이아웃 간 화면 전환 다시 구성(Task 관리)

만든 레이아웃 디자인 세부사항 변경

팀원들간 코드 합치기 진행중 11/20~

<이상윤 작업내용>

11/05 : SNS 기능의 초기 화면 레이아웃 만들. 프로필 이미지를 동그랗게 표시하기 위해

CircleImageView 라는 라이브러리를 사용함. <https://github.com/hdodenhof/CircleImageView>

11/08 : SNS 기능의 초기 화면 레이아웃을 recyclerview 를 이용해 다시 만들. 주간 인기 루틴 게시판을 recyclerview 로 만들. 이 과정에서 게시글들을 자바 객체화 함. 처음 사용하는 recyclerview 의 난이도가 쉽지 않았음.

11/10 : SNS 기능의 게시글 내용을 보여주는 레이아웃을 scroll view 를 이용해 만들.

11/13 : 게시글을 업로드 하는 레이아웃 만들.

11/17 : 게시글 쓰기 레이아웃에서 checkBox 들을 라디오 버튼으로 바꿈. 게시글 내용에 해당하는 postContent 클래스를 구현함. 업로드 버튼을 눌렀을 때 제목, 내용 등의 정보를 postContent 객체에 담도록 이벤트 처리함. 그 외에도 게시글 목록에서 각 게시글에 리스너를 달아서 클릭에 반응할 수 있게 함.

11/20~ : Git 을 이용하여 코드를 합치는 작업 진행중임.

<신형담 작업내용>

11/05

다이어리 기능의 초기화면 레이아웃을 LinearLayout 에 CalendarView 를 배치하는 형태로 제작.

다이어리 기능의 운동일지 레이아웃을 LinearLayout 으로 기본 레이아웃만 구성함.

11/08

다이어리 기능에서 운동일지에 운동정보를 기록할 레이아웃을 LinearLayout 안에 EditText 와 Button 을 배치하는 형태로 제작함. (버튼은 각각의 추가된 LinearLayout 자신을 삭제하는 이벤트를 넣을 예정)

다이어리 운동일지를 LinearLayout 에서 ScrollView 로 수정함.

11/10

다이어리 운동일지에서 운동정보를 기록하는 레이아웃을 버튼 이벤트에 LayoutInflater 를 사용하여 구현함.

추가된 뷰를 각각 삭제하는 버튼 이벤트를 removeView 를 통해 구현하려 했으나 추가되었던 모든 뷰가 삭제되어 실패함.

11/13

운동일지에서 운동정보를 기록하는 LinearLayout 을 각각 삭제하는 버튼 이벤트를 해당 뷰를 추가하는 버튼 이벤트에서 등록하도록 하여 각각의 뷰를 삭제하는 이벤트를 구현하는데 성공함.

11/18

저장소기능의 기본 레이아웃을 ScrollView 로 제작함.

버튼 모양을 커스텀 소스로 형태로 적용하는 과정에서 실패함.

다이어리 초기화면에서 버튼 이벤트에 intent 를 통해 다이어리 운동일지 화면으로 넘어가도록 등록함.

11/20~ : Git 을 이용하여 코드를 합치는 작업 진행중임.

2.12 주차 진행 상황

현재 진행중

작업완료

DB (담당자 문준익)	로그인	로그인 데이터 처리 로그인 화면 만들기	로그인 정보로 서버에서 유저 정보 가져오기 로그인 레이아웃 만들기	2 2
	회원가입	회원가입 데이터 처리 회원가입 화면 만들기	회원가입시 입력한 정보를 서버에 저장 회원 가입 화면 만들기	3 3
	프로필	프로필 데이터 처리 프로필 작성 화면 만들기	서버에 저장되어있는 유저정보 가져와서 메인 화면에 보여주기 유저가 작성,수정한 프로필 정보 서버에 저장 프로필 레이아웃 구성	3 3 3
	오늘운동	루틴 데이터 가져오기	저장소에 선택된 내루틴 의 데이터를 가져오기 그 루틴의 데이터를 읽고 안전한 부분의 데이터를 파싱해서 화면에 띄워주기	4 4
			운동끝난후 오늘날의 운동의 데이터 다시 업데이트	3
	게시판	게시판 루틴 데이터 처리	서버 루틴 목록에 있는 루틴들을 카테고리별 추천순으로 가져오기 사용자가 루틴을 새로 만들어서 등록하면 종합해서 서버에 저장	3 5
			루틴 저장 눌렀을때 유저 저장소에 저장	2
게시판구현 (담당자 이상윤)		게시판	게시판 레이아웃 구성 태그의 맞는 게시글 카테고리별 추천순으로 화면에 띄워주기	4 3
			사진,루틴설명 등의 세부내용 게시판에 전시하기	3
			운동 장소, 난이도 시간 대상등의 정보를 아이콘으로 표현	3
			게시글에 사진 첨부하기	4
		게시글 업로드	운동 루틴 입력 받는 플랫폼 구현 설정한 루틴 다이어리에 띄워주기	5 5
			서버에 데이터 전송할수있게 가공	3
오늘의 운동 (담당자 신형담)		오늘의 운동화면	오늘의 운동 레이아웃 구현	3
			저장 버튼 눌렀을시 다이어리와 동기화	5
			DB에서 가져온 지금까지 해왔던 루틴 정보 화면에 출력	2
운동 다이어리		운동다이어리	다이어리 레이아웃 구성 캘린더에 해당 날짜에 한 운동운동 조회	4 4
			DB에서 가져온 그전에 했던 운동들도 볼수있게 구성	4
공통	코드 병합	깃허브 공부	첫 커밋 브랜치생성	2 2
			main kode 정하기, 각자 개발환경 맞추기	4
			pull, merge 숙련화	4

문준익 - 팀원들이 만든 레이아웃의 데이터를 저장,가져오기

이상윤 - 게시글 카테고리를 입력받는 방식 구상중

신형담 - 운동일지를 객체화하여 저장하는 기능 구현

3. 12 주차 특이사항

팀원 전부가 깃허브를 처음 써보는 환경 이여서 새로 배우고 합치는데 많은 어려움이 있음.

금방 할줄 알았으나 막상 해보니 개발보다 오래걸려 다른 방법으로 합칠지 깃을 좀더 공부해볼지 고민중

3) 13 주차 프로젝트 진도보고서

1. 팀원별 작업 내역

<문준익 작업내용>

11 월 4 켓주

11/24 : 게시판 데이터 서버에 저장 구현, 구조 파악

11/25 : 서버에 저장된 데이터 가져와서 sns_main 에 띄워 주려고 시도,
데이터 가져오는 정황 파악, 원하는 데이터만 뽑아오기 성공(로그) 하지만

어떤 방법을 써도 원하는 변수에 데이터가 저장안됨. 화면에 띄워줄 데이터가 오기도전에 화면이
실행되서 오류.

로그 를 통한 코드 순서 파악. (데이터 불러오기도 전에 화면을 실행 해버림)

11/26 : 13 주차 강의 듣고 스레드 개념 파악,위에 실패에 적용시키기 위한 스레드 구조 설계

11/27 : 우선 적으로 스레드 분리후 가져오는 것 성공, 하지만 handler 를 통한 화면에 띄워주기 실패

11/28 : 화면 요소 객체를 만들어 데이터 가져오면서 넣고 message 로 보내줌으로 성공

<이상윤 작업내용>

11/25

sns_weekly 에서 sns_routine 으로 가는 intent 구현. 객체를 FB 에서 받아서 넘기는 것 보다 게시글 id 만
넘기고 목적지 액티비티에 가서 받아오는 게 나을 것 같아 구조를 수정함

11/27

sns_upload 에서 게시글 카테고리 입력 구현, 문준익 팀원과 게시글을 식별 할 key 값을 어떻게 정할 지
토의함.

11/28

sns_upload 에서 게시글 카테고리 입력 구현 완료.

11/30

sns main 에서 sns_weekly 로 가는 인텐트 구현, 카테고리 정보를 넘겨서 sns_weekly 에 띄움.

sns_routine 의 heart 이미지 뷰에 온클릭 리스너를 달아서 이미지 변경(아이콘 drawable 에 추가함), 이미지 뷰를 클릭 시 좋아요 수 증가하는 부분 어느정도 구현.

<신형담 작업내용>

11/24

저장소에서 나만의 루틴에서 일 단위 운동을 담을 my_routine_box 레이아웃의 기본 뷰를 제작함. 그리고 my_routine_box 에 담길 운동 정보를 입력 받을 diary_box_my_routine 레이아웃의 기본 뷰를 제작함.(다이어리에서 운동 정보를 입력받는 diary_box 를 베이스로 함.)

11/26

저장소의 나만의 루틴 기능에서 버튼 이벤트에 LayoutInflater 를 등록하여 일 단위 루틴을 담을 my_routine_box 를 동적으로 추가, 삭제하고 그 안에 운동 정보를 입력 받을 diary_box_my_routine 를 동적으로 추가, 삭제하는 기능을 구현하는 과정에서 서로의 버튼 이벤트가 충돌함.

11/28

다이어리에서 운동 정보를 기록하는 뷰의 추가 디자인 작업을 함.

다이어리에서 운동 정보를 기록하는 뷰의 버튼의 형태를 xml 파일을 통해 원형으로 바꿈.

diary_main 액티비티에서 바깥 화면을 터치하면 키보드가 숨김 상태가 되도록 코드를 추가함.

2.13 주차 진행 상황

현재 진행중

작업완료

DB	로그인	로그인 데이터 처리	로그인 정보로 서버에서 유저 정보 가져오기	2
(담당자 문준익)		로그인 화면 만들기	로그인 레이아웃 만들기	2
	회원가입	회원가입 데이터 처리	회원가입시 입력한 정보를 서버에 저장	3
		회원가입 화면 만들기	회원 가입 화면 만들기	3
	프로필	프로필 데이터 처리	서버에 저장되어있는 유저정보 가져와서 메인 화면에 보여주기	3
		프로필 작성 화면 만들기	유저가 작성,수정한 프로필 정보 서버에 저장	3
			프로필 레이아웃 구성	3
	오늘운동	루틴 데이터 가져오기	저장소에 선택된 내루틴 의 데이터를 가져오기	4
			그 루틴의 데이터를 읽고 안한 부분의 데이터를 파싱해서 화면에 띄워주기	4
			운동끝난후 오늘한 운동의 데이터 다시 업데이트	3
	게시판	게시판 루틴 데이터 처리	서버 루틴 목록에 있는 루틴들을 카테고리별 추천순으로 가져오기	3
			사용자가 루틴을 새로 만들어서 등록하면 종합해서 서버에 저장	5
			루틴 저장 눌렀을때 유저 저장소에 저장	2
게시판구현		게시판	게시판 레이아웃 구성	4
(담당자 이상윤)			태그의 맞는 게시글 카테고리별 추천순으로 화면에 띄워주기	3
			사진,루틴설명 등의 세부내용 게시판에 전시하기	3
			운동 장소, 난이도 시간 대상등의 정보를 아이콘으로 표현	3
			게시글에 사진 첨부하기	4
		게시글 업로드	운동 루틴 입력 받는 플랫폼 구현	5
			설정한 루틴 다이어리에 띄워주기	5
			서버에 데이터 전송할수있게 가공	3
오늘의 운동		오늘의 운동화면	오늘의 운동 레이아웃 구현	3
(담당자 신형담)			저장 버튼 눌렀을시 다이어리와 동기화	5
			DB에서 가져온 지금까지 해왔던 루틴 정보 화면에 출력	2
운동 다이어리		운동다이어리	다이어리 레이아웃 구성	4
			캘린더에 해당 날짜에 한 운동운동 조회	4
			DB에서 가져온 그전에 했던 운동들도 볼수있게 구성	4
공통	코드 병합	깃허브 공부	첫 커밋	2
			브랜치생성	2
			main kode 정하기, 각자 개발환경 맞추기	4
			pull, merge 숙련화	4

문준익 - 현재 sns_main 부분만 성공한 스레드 분리후 데이터 가져오기를,
여러 화면에 적용시켜 화면 특성에 맞는 데이터 스레드 구상중.

이상윤 - 전체적으로 액티비티와 파이어 베이스의 연동을 매끄럽게 하기 위한 작업중임.

신형담 - 루틴 입력 포맷 제작중

3.13 주차 특이사항

문준익

11/30 : 다른 액티비티에선 실시간으로 서버에 업데이트 해줘야하는 부분이 있는데 어떻게 해야할지 잘 모르겠음, 이번주 쓰레드 구조 파악하기 에 너무 오래 걸림.

첨부사진 - 로그로 다 찍어보면서 데이터 순서구조 파악(보고서에 적합하지 않은 개인적인 메시지 블라인드 처리했습니다)

분홍 : 데이터 스레드

노랑 : 데이터 스레드 에서 호출한 함수들

마지막 분홍: 데이터 스레드 끝

그 다음 메시지 큐에 있는 메시지 실행됨

```

D/중복없음 넣음: => edf카테고리
=> edf카테고리
=> edfg카테고리
=> edfg카테고리
=> 상운카테고리
=> hipup카테고리
=> 형답카테고리
=> dddd카테고리
=> e o o 카테고리
D/중복됨 안넣음: => edfg카테고리
D/화면 띄워보자: => 0 띄워줘 제발발
D/데이터 가져오는중: => 시작카테고리
D/데이터 가져오는중: => 진짜시작 ==0카테고리
D/중복안되서 넣음이라는 메시지 받음: 상운형답마 넣었데
I/chatty: uid=10442(com.example.yourhealth) identical 7 lines
D/중복안되서 넣음이라는 메시지 받음: 상운형답마 넣었데
D/중복되서 안넣음이라는 메시지 받음: 상운형답마 안넣었데

```

이상윤

11/30 : sns_upload 에서 heart 버튼을 눌렀을 때, 이미 눌렀던 유저라면 좋아요를 취소하게 기능하고 누르지 않았던 유저는 좋아요가 증가하게 해야 함. 그러려면 그 유저가 버튼을 눌렀었다는 것을 저장해야 하는데, 그 방법을 고민 중임.

신형담

11/30 : 동적으로 추가해야 할 뷰의 단계가?(ex 추가된 뷰에서의 버튼이벤트로 또 뷰를 추가하는것)
높아질수록 버튼 이벤트의 처리가 까다로워서 원하는 방식으로 이벤트 처리가 어려움

4) 14 주차 프로젝트 진도보고서

1. 팀원별 작업 내역

<문준익 작업내용>

12 월 1 켓주

12/2 : 일단 어느정도 완성된 기능 레이아웃들을 연결하기위한 mainmenu 레이아웃 만듦, 로그인한 유저에 프로필 데이터를 가져와서 화면에 보여줌

화면전환 인텐트 정리

12/3 :

게시판(목록) -> 세부게시판->게시물 구조에서

postContent 라는 객체의 데이터를 필요한 부분만 불러와서 쓰고있었는데

게시판 - 모든 postContent 에서 category 만 불러와 다시 그 category 를 가지고있는 postContent 를 가져옴

세부 게시판 - 해당 category 에 있는 postContent 를 가져옴

게시물 - postContent 의 제목으로 가져옴

이런 구조에서

제목이 겹칠수 있는 상황을 고려하여 postContent 의 고유키를 제목+작성자 로 변경하여 고유 키 설정, 화면에 보여 줄수있게 객체로 다시 가져옴.

좋아요수, 조회수 구현 실패

12/4 : 게시물 임시저장 기능 구현 위해 서버 보단 서비스 컴포넌트를 이용하여 구현하려 시도 실패.

12/5 : dairy 해당 날짜에 그날 운동이 있다면 운동일지 보여주기, 없다면 운동일지 작성 이 될 수 있도록

유저 - 날짜 에 당하는 고유키를 구성하여 서버 저장, 불러오기 구현

유저가 일일 운동 일지를 작성한 후 저장하면서 생긴 데이터를 구성 팀원이 만든 데이터 객체를 서버에 저장, 불러옴

<이상윤 작업내용>

12/02

sns_routine 에서 FB 에서 받아온 postcontent 를 화면에 전시해줌.

12/03 작업 내용

sns_routine 에서 받아온 내용에 따라 아이콘 변경

sns_upload 에서 업로드, 임시저장 버튼 누르면 저장 후 sns_main 으로 이동

12/05

makeprofileActivity 에 직업 입력 받는 editText 만들고 profile 객체에 job 멤버 추가함

sns_routine 의 heart 이미지 뷰에 온클릭 리스너를 달아서 이미지 변경(아이콘 drawable 에 추가함), 이미지 뷰를 클릭 시 좋아요 수 증가하는 부분 어느정도 구현.

<신형담 작업내용>

12/01

diary_data 객체를 통해 운동일지의 정보를 저장 버튼을 누를 시 객체화 할 수 있도록 구현함.

12/03

diary_data_box 객체를 통해 diary_data 를 ArrayList 로 관리하여 하루 치 운동일지를 저장 버튼을 누를 시 객체화 할 수 있도록 수정함.

12/05

Intent 를 통해 diary_start 에서 선택한 날짜 정보를 diary_main 에서 사용할 수 있도록 구현함.

diary_start 에서 날짜를 선택하지 않았을 때 diary_main 으로 넘어가지 못하게 제어하고 Toast 메시지를 띄워주도록 수정함.

2.14 주차 진행 상황

마무리 단계				
현재 진행중				
작업완료				
DB (담당자 문준익)	로그인	로그인 데이터 처리 로그인 화면 만들기	로그인 정보로 서버에서 유저 정보 가져오기 로그인 레이아웃 만들기	2
	회원가입	회원가입 데이터 처리 회원가입 화면 만들기	회원가입시 입력한 정보를 서버에 저장 회원 가입 화면 만들기	3
	프로필	프로필 데이터 처리	서버에 저장되어있는 유저정보 가져와서 메인 화면에 보여주기	3
		유저가 작성,수정한 프로필 정보 서버에 저장		3
		프로필 작성 화면 만들기	프로필 레이아웃 구성	3
	오늘운동	루틴 데이터 가져오기	저장소에 선택된 내루틴 의 데이터를 가져오기 그 루틴의 데이터를 읽고 안한 부분의 데이터를 파싱해서 화면에 띄워주기	4
			운동끝난후 오늘한 운동의 데이터 다시 업데이트	3
	게시판	게시판 루틴 데이터 처리	서버 루틴 목록에 있는 루틴들을 카테고리별 추천순으로 가져오기	3
			사용자가 루틴을 새로 만들어서 등록하면 종합해서 서버에 저장	5
			루틴 저장 눌렀을때 유저 저장소에 저장	2
게시판구현 (담당자 이상윤)		게시판	게시판 레이아웃 구성 태그의 맞는 게시글 카테고리별 추천순으로 화면에 띄워주기	4
			사진,루틴설명 등의 세부내용 게시판에 전시하기	3
			운동 장소, 난이도 시간 대상등의 정보를 아이콘으로 표현	3
			게시글에 사진 첨부하기	4
		게시글 업로드	운동 루틴 입력 받는 플랫폼 구현 설정한 루틴 다이어리에 띄워주기	5
			서버에 데이터 전송할수있게 가공	3
	오늘의 운동 (담당자 신형담)	오늘의 운동화면	오늘의 운동 레이아웃 구현	3
			저장 버튼 눌렀을시 다이어리와 동기화	5
			DB에서 가져온 지금까지 해왔던 루틴 정보 화면에 출력	2
	운동 다이어리	운동다이어리	다이어리 레이아웃 구성 캘린더에 해당 날짜에 한 운동운동 조회	4
공통			DB에서 가져온 그전에 했던 운동들도 볼수있게 구성	4
	코드 병합	깃허브 공부	첫 커밋	2
			브랜치생성	2
			main kode 정하기, 각자 개발환경 맞추기 pull, merge 숙련화	4

문준익 - 팀원들이 만들어낸 루틴 객체를 여러 액티비티에서 필요한 부분만

가져오기 위해 고유키 고유키 이용 방식 구상 중

루틴 객체를 작성자 UID+제목 을 이용하여 고유키 할당

유저 프로필에 storage 변수 추가하여 유저들이 저장한 루틴들의 고유키만 저장.

Curroutine 변수 추가하여 현재 진행중인 루틴제목,day 저장

게시물 객체에 루틴 고유키 변수 추가

이상윤 - 루틴 객체를 입력 받는 방식과 저장하는 방식을 구체화 하는 중
신형담 - 루틴 객체를 입력 받는 방식과 저장하는 방식을 구체화 하는 중

3.14 주차사항

서비스 컴포넌트 수업을 듣고 (14 주차) 임시저장 기능을 구현 하려 했으나
서비스 객체에 필요한 데이터가 들어가지 않음. 인텐트나 내부 함수를 이용하여
시도 했으나 실패했음 . 하다가 안되면 서비스 컴포넌트 보단 데이터 객체의 변수를 설정하여 구현 예정

앱 실행시 sns_main 액티비티 에서 필요한 데이터를 불러 왔음. 원하는 데이터들이 잘 띄워져 있음, 하지만
자꾸 터치 시 튕기는 현상이 발생
똑같은 데이터들이 띄워져 있는데도 어떨 땐 잘 작동하고 어떨 땐 튕기는 상황
아직 이렇게 같은 환경에서 앱이 종료되는 문제를 해결하지 못함.

Makeprofile 액티비티에서 glide API 를 이용하여 image view 에 사진을 띄워주는데
최초 액티비티 실행 시 다른 layout 요소들이 안보이는 현상 발생
Image view 를 다시 눌러야만 화면이 정상적으로 보여짐
이유를 찾지 못함.

5) 최종보고서

1-1 로그인

-작업 내용 : FireBase 의 다양한 Authentication 방법 중 e-mail 을 이용한 방법을 채택 하였다.

1-2 회원가입

-작업 내용 : 회원 가입시에 입력 사항 중 하나라도 누락된다면 가입 버튼은 활성화 되지 않는다.

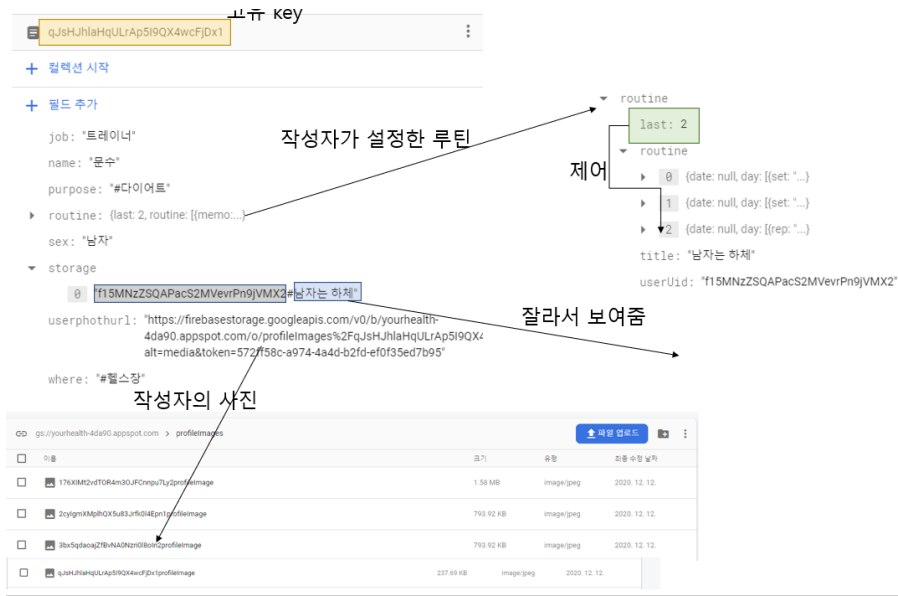
1-3 회원 정보 설정

-작업 내용 : 하나의 항목이라도 누락 될 경우 역시 저장 버튼은 활성화 되지 않는다. 단, 첫 설정 때 카테고리 버튼만 눌러도 켜지는 오류가 있다.



프로필 사진을 설정하는 기능도 구현하였으나, 데드라인을 앞두고 갑자기 유저 정보를 가져올 때 오류가 발생했다. 우선은 메인 화면에서만 프로필 사진을 변경 가능하게 해 두었다.

1-4 프로필 정보(main menu)



-작업 내용 : 레이아웃 생성 시 user의 uid를 이용하여 유저 정보를 서버 폴더로부터 가져온다. 가져온 정보는 위와 같이 각각 다른 기능에서 사용되는데,

루틴 - 오늘의 운동 ,다이어리

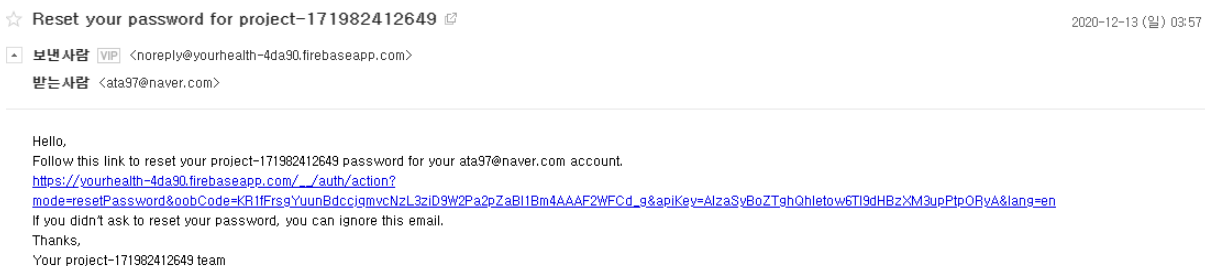
Storage 배열 - 저장소

이름, 카테고리 - 운동 마당

에서 사용된다. 데이터베이스는 FireBase CloudFireStore를 이용하였다. 해당 데이터베이스에서는 사진을 파일 형태로 업로드 할 수 없었기 때문에 방법을 생각해야 했다. 고민 끝에 떠올린 방식은, 사용자가 갤러리에서 사진 URI 정보를 가져오면, 사진을 FireBase Storage에 저장한 후 그 위치의 URL을 받아와서 String 형태로 서버 프로필 정보에 저장하는 것이다. 이렇게 하면 유저 정보와 프로필 사진을 동시에 다룰 수 있다.

1-5 비밀 번호 재설정

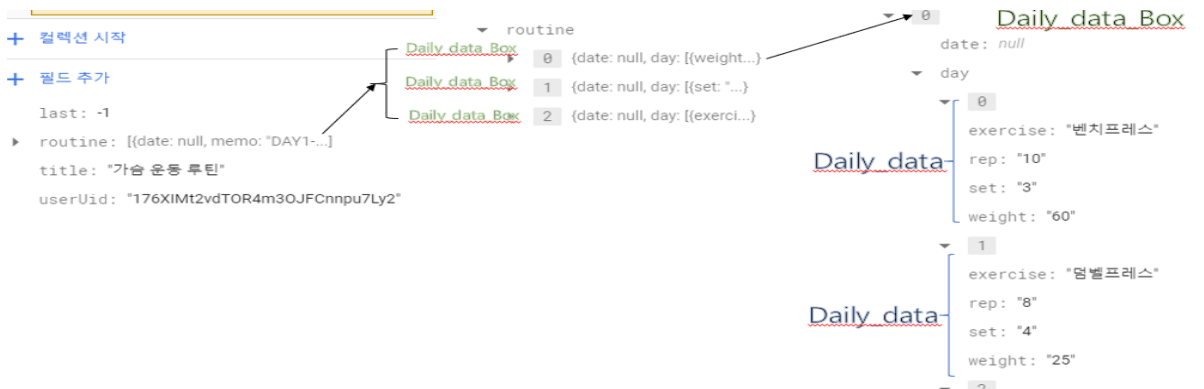
-작업 내용 : 데모 영상에서는 보이지 못했지만 구현 하였다.



사용자는 FireBase 로부터 이런 메일을 받아 링크를 통하여 비밀번호를 재설정 할 수 있다.

2-1 루틴 조회 - 문준익, 신형담

-작업 내용 : 레이아웃에 유저가 설정한 루틴을 보여주기 위하여 서버에 저장 되어있는 유저 정보에서 Routine 객체를 가지고 왔다. Routine 객체에 대해 설명해 보면



Diary_data : 하나의 운동 정보에 해당하는 객체

Diary_data_Box : 하루 분의 운동 일지를 표현하는 객체

Routine : 여러 일의 운동 일지 정보를 모아둔 객체

이다.

객체 안에 객체가 들어있는 복잡한 구조의 객체를 상황 별로 필요한 부분을 불러오는 것이 힘들었다.



Routine 객체는 유저가 어디까지 수행했는지를 나타내기 위해 last 변수를 만들었다,

이 변수를 이용 하여 그 날 유저가 수행해야 할 루틴을 가져올 수 있다.

우선 만약 routine.size()가 Routine 의 last+1 보다 작거나같다면 Routine의 last 는 -1 이 된다.

그후 routine 의 last+1 에 해당하는 daily_Box 데이터를 가져와 화면에 보여준다.

그럼 만약 끝까지 다 하게 되면 다시 배열0번째 인덱스를 보여주게 된다.

만약 루틴을 설정하지 않은 사용자라면 루틴 객체가 없다. 그럴 경우 바로 다이어리의 운동일지 작성으로 넘어 가게 된다. 원래는 루틴이 없을 경우 빈 화면을 보여주게 구현하였지만, 고민 끝에 다이어리를 보여 주기로 결정했다.

기여도 : 문준익(60%), 신형담(40%)

2-2 다이어리 연동 - 문준익

-작업 내용 : 2-1에서 가져온 routine 배열의 diary_data_box 데이터 하나를 날짜와 함께 서버에 저장해 준다. 자세한 설명은 다이어리 기능에서 하겠다.

2-3 루틴 수정

작업 내용 : 루틴은 여러 개의 day 에 해당하는 객체로 이루어져 있고, 각 객체의 참조는 list 로 가지고 있었다. 루틴을 전시할 때 각 day 들을 일렬의 버튼 형태로 보여주는데, 사용자가 원하는 날에 해당하는 버튼을 누르면 그 날짜의 운동들을 수정할 수 있게 하려고 하였다. 사용자가 어떤 버튼을 선택할 지 모르기 때문에 각 버튼은 자신이 몇 번째 버튼인지에 해당하는 index 를 리스너 객체의 멤버 변수로서 가지고 있게 하였다. 그래서 자신의 클릭 리스너가 호출이 되면, list 를 조회해서 해당하는 day 객체를 intent 로 넘기려고 하였다. 그 이후의 수정 기능은 다이어리에 이미 구현이 되어 있었기 때문에 구현이 가능할 것이라고 생각하였다. 하지만 루틴의 각각의 day 들은 임의로 삽입, 삭제가 가능하고 그 과정에서 각 버튼의 index 를 업데이트 하는 과정이 순탄치 않았다. 결국은 삭제 기능을 고려하지 않고 성공적으로 intent 를 넘겨 수정하는 기능을 구현하는 데에 거의 근접했지만, 그럴 경우 삭제 기능을 사용하지 못하게 되므로 최종 소스코드에서는 제외할 수 밖에 없었다. 시간이 조금만 더 있었다면 충분히 구현했을 것이다.

기여도 : 이상윤(50%), 신형담(50%)

3-1 운동 일지 작성

다이어리의 운동 일지 작성 기능은 사용자가 추가 버튼을 통해 하나의 운동 정보(운동 종류, 무게, 세트, 횟수)와 삭제 버튼을 담은 뷰를 동적으로 추가시켜주고 동시에 삭제 버튼을 통해 추가된 뷰를 삭제할 수 있도록 작동한다. 처음에는 삭제 버튼을 누르면 각각의 추가된 뷰가 삭제되지 않고 추가된 모든 뷰가 삭제되거나 앱이 중단 되었다. 프로젝트에서 가장 처음으로 구현을 시작한 기능이라 원인을 찾는데 거의 4~5 시간을 전전긍긍했다. 이는 삭제 버튼이 각각의 참조를 얻지 못한 것임을 알고 추가 버튼의 OnClick 에서 각각의 삭제 버튼의 참조를 얻어 setOnClickListener 를 구현하는 방식으로 해결했다.

3-2 일지 저장

작업 내용 :

운동 일지는 작성과정에서 뷰를 동적하고 추가하고 삭제 할 수 있기 때문에 이러한 이벤트들이 뷰를 담은 ArrayList 와 diary_data_box 객체의 ArrayList 에도 정확히 동기화를 시켜야 했기 때문에 구현에 상당히 어려움을 겪었고 시간도 많이 소요되었다.

다이어리에서 작성된 운동 일지를 저장하기 위해 유저가 작성한 각각의 운동 정보(운동종류, 무게, 세트, 횟수)를 담은 뷰를 ArrayList 에 담아 놓는다. 추가로 이 ArrayList 에 담긴 뷰의 EditText 정보를 String 으로 받아와 멤버로 저장할 객체(diary_data)와 이 객체(diary_data)를 담은 ArrayList 를 멤버로 갖는 객체(diary_data_box)를 구현한다. 저장 버튼을 통해 diary_data 객체를 생성하여 뷰의 EditText 정보로 멤버를 세팅해주고 diary_data_box 의 ArrayList 에 각각 diary_data 를 넣어주고 이를 뷰를 담은 ArrayList 의 사이즈만큼 for 문을 돌려 하루치의 운동 일지가 하나의 객체에 저장된다.

이 저장 된 객체를 서버에 UserId + Daily 라는 Collection 에 현재날짜 라는 document로 들어가게 된다.

저장된 모습은 3-2 에서 볼 수 있다.

기여도 : 문준익(50%) 신형담(50%)

3-3 일지 조회

작업 내용 : 3-2 에서 서버에 저장된 데이터이다

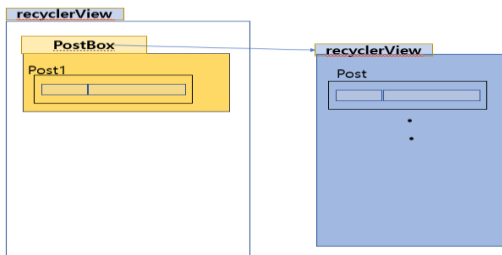


만약 저 data 라는 필드가 유저가 선택한 달력의 날짜와 같다면 버튼의 Text 가 "운동일지 보기"로 변경 된다, 매칭이 안되면 " 날짜, 운동일지 작성되기" 로 변경된다.

기여도 : 문준익(50%), 신형담(50%)

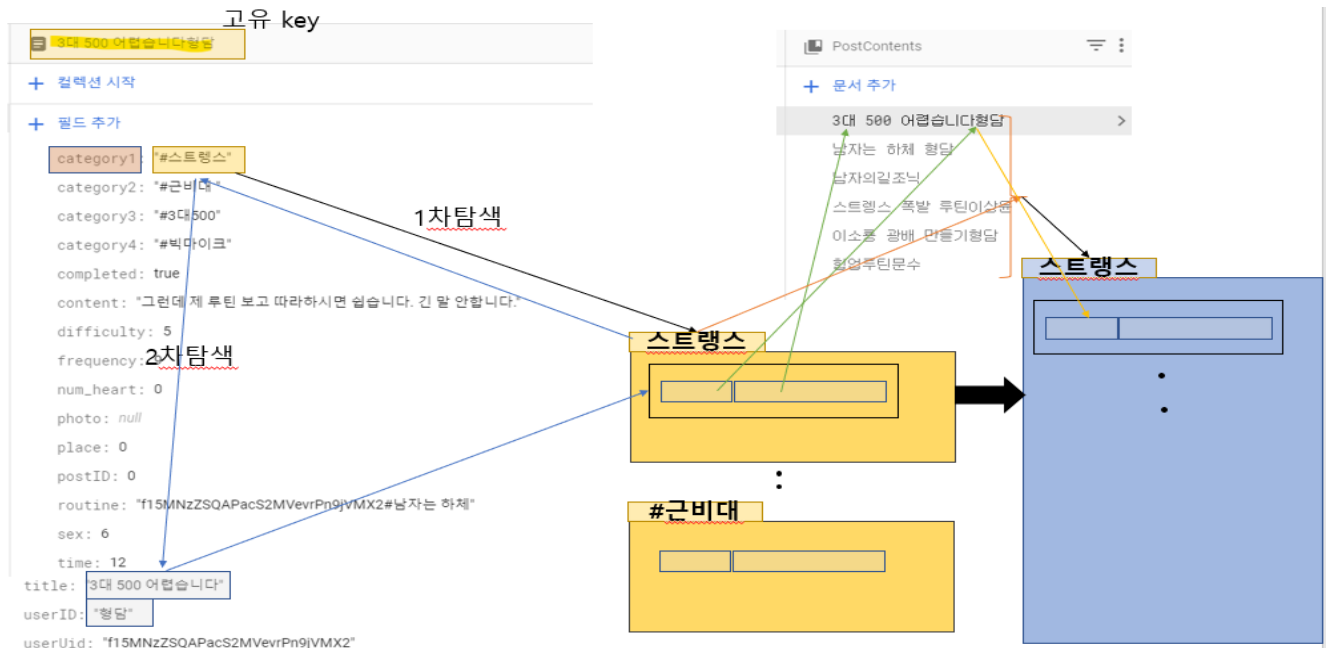
4-1 게시물 불러오기

작업 내용 : 운동 마당에서 게시글을 불러오는 것은 상당히 힘들었다.



간략히 이런 구조 였는데 1차로 PostBox 의 제목을 CateGory1 에 해당하는 데이터를 채우고 그 제목과 같은 CateGory1 field 값을 가지고있는 Post 들을 불러와 PostBox를 다시 채워 화면에 보여주는 구조였다.

서버와 연동해서 생각해 보면



서버에서 게시물 데이터를 순회하면서 Category1에 해당하는 String 데이터를 뽑아와 배열에 넣는다. 배열 내부에서 중복이 이루어지지 않는다면 모든 게시물 데이터 중, 이 String 데이터를 Category1의 값으로 가지고 있는 게시물 중에 title과 userID를 가져와서 post 객체로 만든 후 postBox에 넣는다. 이 과정을 모든 중복되지 않는 Category1값을 전부 가져올 때까지 한다.

그후 사용자가 원하는 게시물을 클릭하면 title+UserID를 이용하여 서버에서 해당 데이터를 PostContent 객체 형태로 가져와 준다.

처음에는 스레드 없이 시도했으나 아무리 로그를 출력하고, 구글링을 해봐도 실패했고, 스레드를 구현 해도 이해가 부족해서 핸들러를 다루지 못해 실패했다. 데이터를 가져오는 스레드에서 하나의 PostBox를 완성 시키고 그 PostBox 객체를 message를 통해 메인 스레드로 보냈다.

이렇게 받아온 데이터를 이용해서 게시글의 목록을 불러오기 위해 리사이클러 뷰를 사용해야 했는데, 어댑터를 사용하는 복잡한 구조 때문에 순탄치 않았다. 이 기능을 구현할 당시에는 기본적인 뷰와 뷰 그룹밖에 알지 못했는데, 이런 복잡한 뷰를 이용해서 게시글을 단순히 띄워주는 것 뿐 만 아니라, 여러 게시글을 카테고리에 따라 묶은 box 객체를 만들고 그 box를 여러 개 띄워줘야 했다. 객체 안에 객체가 있는 복잡한 구조와 리사이클러 뷰와의 연결이 쉽지 않았다.

최초 목적은 유저 카테고리 와 연동하여 모든 게시물 중 category1,2,3,4, 와 유저의 카테고리와 같은 값을 가지는 걸 찾으려 했으나, 스위치 변수 1,2,3,4를 사용하여 서칭 후 다시 서칭을 한다던지, 스레드를 분리해서 겹치는 category를 먼저 보여주고 다시 다른 스레드를 통해 그 외 데이터를 가져 오기, 등 여러 방법을 시도 해 보다 실패했다.

기여도 : 이상윤(50%), 문준익(50%)

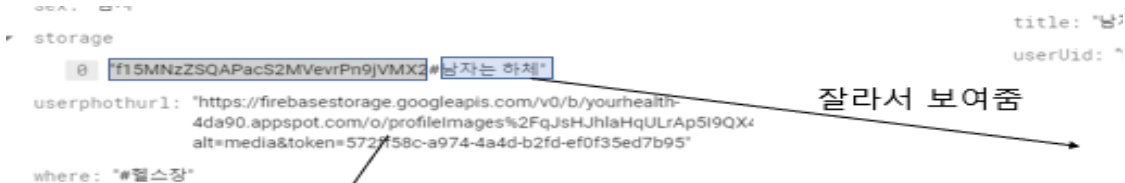
4-2 게시물 업로드

작업 내용 : 게시물 하나에도 제목, 내용, 글쓴이, 프로필 사진, 운동 사진, 난이도, 운동 시간 등 수 많은 멤버가 존재하기 때문에, 각각을 입력 받는 포맷에 해당하는 레이아웃을 만드는 데에 적잖은 시간이 걸렸다. 특히, 카테고리를 입력 받는 방식을 팀원들과 오래 고민하였다. 다이얼로그를 띄워서 선택하게 하는 방법, 일부분을 입력하면 자동완성 기능을 제공하는 방법 등을 생각해 보았지만 전자는 사용자 친화적이지 않아 제외했고, 후자는 구현 가능성이 낮아서 제외했다. 마지막에는 스트링을 입력 받는 방식을 선택하였다.

```
category1: "#스트레스"
category2: "#근비대"
category3: "#3대500"
category4: "#빅마이크"
completed: true
content: "그런데 제 루틴 보고 따라하시면 좋습니다. 긴 말 안합니다"
difficulty: 5
frequency: 9
num_heart: 0
photo: null
place: 0
postID: 0

routine: "f15MNzZSQAPacS2MVeVrPn9jVMX2#남자는 하체"
sex: 6
time: 12
title: "3대 500 여합니다"
userID: "형달"
userId: "f15MNzZSQAPacS2MVeVrPn9jVMX2"
```

루틴을 가져올 경우 엔 사용자의 storage배열이 있어야만 한다 있다면 서버에 있는 User 데이터의



Storage 배열의 정보들을 사용자가 볼때 파란 부분만 보이게 되고 해당 부분을 누르면 전체 데이터가 합쳐서 올라오게 된다. 또 postContent 에도 전체 부분이 같이 저장되어 있다.

전체 부분은 루틴의 고유 키 이다.

루틴 객체의 고유키를 설정 하기 위해 여러 아이디어를 생각해 보았다.

만든 사람의 Uid로만 할 시 뒤편에 쓰게 된다는 단점 이 있어 고유 하면서 다름을 부여하기 위해 Uid+제목 으로 설정 하였고 또한 필요 시 마다 쪼개어 쓰기 위해 중간에 #이라는 구분점 을 주어 5-2,5-3 에서 이용 하였다.

기여도 : 이상윤(50%), 문준익(50%)

4-3 게시물 조회

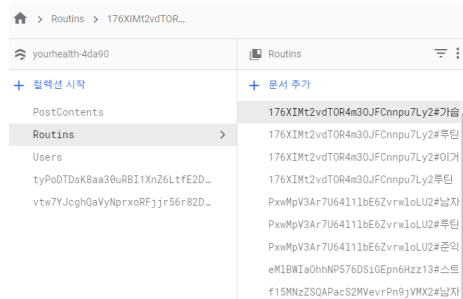
작업 내용 : 게시글을 조회하는 과정에서, 다양한 게시물 내의 속성에 따라 아이콘에 변화를 주었다. 가령, 운동 장소가 집, 헬스장, 야외일 경우에 각각 아이콘이 달라지게 하였다. 이런 방식이 유저로 하여금 한 눈에 루틴을 직관적으로 이해하게 해줄 거라고 생각했다. 또한, 추가 구현 사항으로 조회 수, 다운로드 수, 좋아요

수를 구현하려고 하였으나 실패하였다. 좋아요 버튼을 화면에 달고, 누를 때마다 아이콘을 변경하는 것까지는 구현하였으나, 사용자가 하나의 게시글에 여러 번 좋아요를 누를 수 없게 하는 방법을 찾아내지 못했다.

기여도 : 이상윤(50%), 문준익(50%)

4-4 루틴 다운로드

작업 내용 : 사용자가 무료 다운로드 버튼을 누르면 4-3 에서 저장된 루틴의 고유키를 통해 서버에서 탐색을



한다.

성공 한다면 해당 루틴의 고유 키가 User데이터의 Storage 배열에 add 된다.

5-1 루틴 작성

작업 내용 : 루틴을 입력 받는 포맷을 제공하는 것이 이 앱의 가장 큰 관문 중 하나였다. 하지만 루틴은 day의 집합이고, 각 day를 입력 받는 부분은 다이어리의 기능을 재사용하여 해결할 수 있을 거라고 생각하였다. 이런 발상이 이 기능의 구현 가능성을 높이는 데에 상당 부분 기여하였다. 이 기능을 위해서는 필연적으로 루틴 객체를 액티비티 간에 주고받아야 하는데, parcelable를 비롯한 여러 방법을 시도해보았지만 잘 되지 않았음. 루틴 객체는 멤버 변수로 day 객체로 이루어진 list를 가지고 있었고, 따라서 루틴 클래스 뿐 아니라 그 멤버로 들어가는 모든 클래스가 parcelable을 상속받아야 한다는 것을 깨닫는 데 까지 상당한 시간이 소요되었다.

기여도 : 이상윤(50%), 신형담(50%)

5-2 루틴 목록 조회

작업 내용 : 서버에 있는 User의 Storage 배열에서 가져와 저장해 둔다. 그 후 파란 부분만 잘라 또 다른 하나의 배열로 만들어 두면 만든 레이아웃 보여 준다.

기여도 : 문준익(60%), 이상윤(40%)

```

storage
  0 "f15MNzZSQAPacS2MVevrPn9jVMX2#"남자는 하체"
  userphotourl: "https://firebasestorage.googleapis.com/v0/b/yourhealth-4da90.appspot.com/o/profileimages%2FqJsHJhlaHqULrAp5i9QX%3Falt=media&token=572f58c-a974-4a4d-b2fd-ef0f35ed7b95"
  where: "#헬스장"
title: "남"
userId: "

```

잘라서 보여줌

5-3 루틴 선택

작업 내용 : 만약 선택을 한다면 해당 버튼의 Text를 가져와서 파란 부분만 잘린 배열과 비교하여 index 값을 얻어내고 원래 User 의 Storge 배열에다 방금 찾은 index값을 넣어 가져온후

Rotins 폴더에서 서칭하여 원하는 데이터를 가져온다.

이 가져온 Routine 객체는 다시 User 의 Routine 에 업데이트 된다.

기여도 : 문준익(80%), 이상윤(20%)

-마침말

저희 조는 사실 안드로이드를 처음 접하는 조원들로 구성되어서 프로젝트의 기본 요소를 제작할 때도 서로 협력한 부분이 많습니다. 따라서 3명 모두 같이 열심히 했고 동등하게 기여했다고 생각합니다.

-최종보고서 Task표

	세부 기능	who	symbol
1. 초기 설정	1-1 로그인	문준익	O
	1-2 회원 가입	문준익	O
	1-3 회원 정보 설정	문준익	△
	1-4 프로필 정보	문준익	O
	1-5 비밀번호 재설정	문준익	O
2. 오늘의 운동	2-1 루틴 조회	문준익, 신형담	O
	2-2 다이어리 연동	문준익	O
	2-3 루틴 수정	신형담, 이상윤	X
3. 다이어리	3-1 운동 일지 작성	신형담	O
	3-2 일지 저장	신형담, 문준익	O
	3-3 일지 조회	신형담, 문준익	O
4. 운동마당	4-1 게시글 불러오기	이상윤, 문준익	△
	4-2 게시글 업로드	이상윤, 문준익	O
	4-3 게시글 조회	이상윤, 문준익	O
	4-4 루틴 다운로드	문준익	O
5. 저장소	5-1 루틴 작성	신형담, 이상윤	O
	5-2 루틴 목록 조회	이상윤, 문준익	O
	5-3 루틴 선택	이상윤, 문준익	O