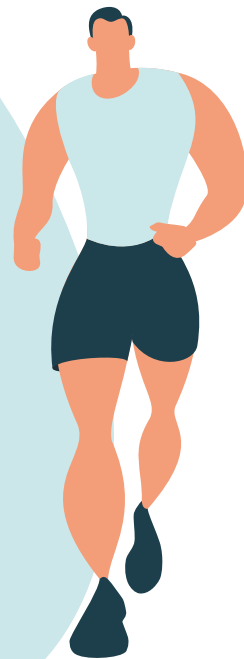




확찐자 길들이기

안드로이드 스튜디오 프로젝트



팀원 소개

MKT

음식, 프로필,
공지사항,
FAQ 구현 및 테스트
만보기 기능 구현
DB 설계



전한웅

메인, 운동, 목표
구현 및 테스트
스프링 서버 설정
레트로핏 연동

목차

01

개요

02

프로젝트 설계

03

기술 아키텍처

04

프로젝트 모델링

05

화면 구성 및 시연

06

추가 설명

이 개요



주제 선정 배경

COVID-19로 인한 거리두기가 장기화 되면서
'확찐자'라는 신조어가 생길 정도로 많은
사람들이 건강관리(체중관리)에 소홀 해짐.



주제 설명



운동

- 날짜별 운동 기록 관리
- 총 칼로리 소모량



목표

- 도전 목표 기록



식사

- 날짜별 식사 기록 관리
- 식사별 칼로리 섭취량

프로젝트 절차

01

자료 수집과 문제 분석

- 기존 서비스 중인 어플리케이션 분석
- 분석을 통해 차별화된 전략 제시

02

디자인 분석

- 사용자에게 필요한 디자인 분석
- 디자인 벤치마크

03

구축 및 개발

- 간편한 사용성을 기반으로 개발 진행
- 목표 설정을 통한 사용자의 도전 정보 관리 기능 제공

04

어플리케이션 테스트

- 어플리케이션 안정성 테스트

프로젝트 수행 일정

TIMELINE												
기간	4/26	4/30	5/3	5/7	5/10	5/14	5/17	5/21	5/24	5/28	5/31	6/4
수행시간	3HR		4HR		4HR		4HR		4HR		4HR	
주제 선정 및 자료 수집	착수											
디자인 설계 및 와이어프레임 작성												
프론트엔드 작업			구축									
백엔드 작업												
테스트 수행												

02

프로젝트 설계



기능 설계

어떤 기능을 가진 서비스 인가?



음식



운동



목표

정보 관리와 조회를 간편하게 사용

기능 설계

사용자가 만족을 느낄 수 있는 추가 서비스



만보기

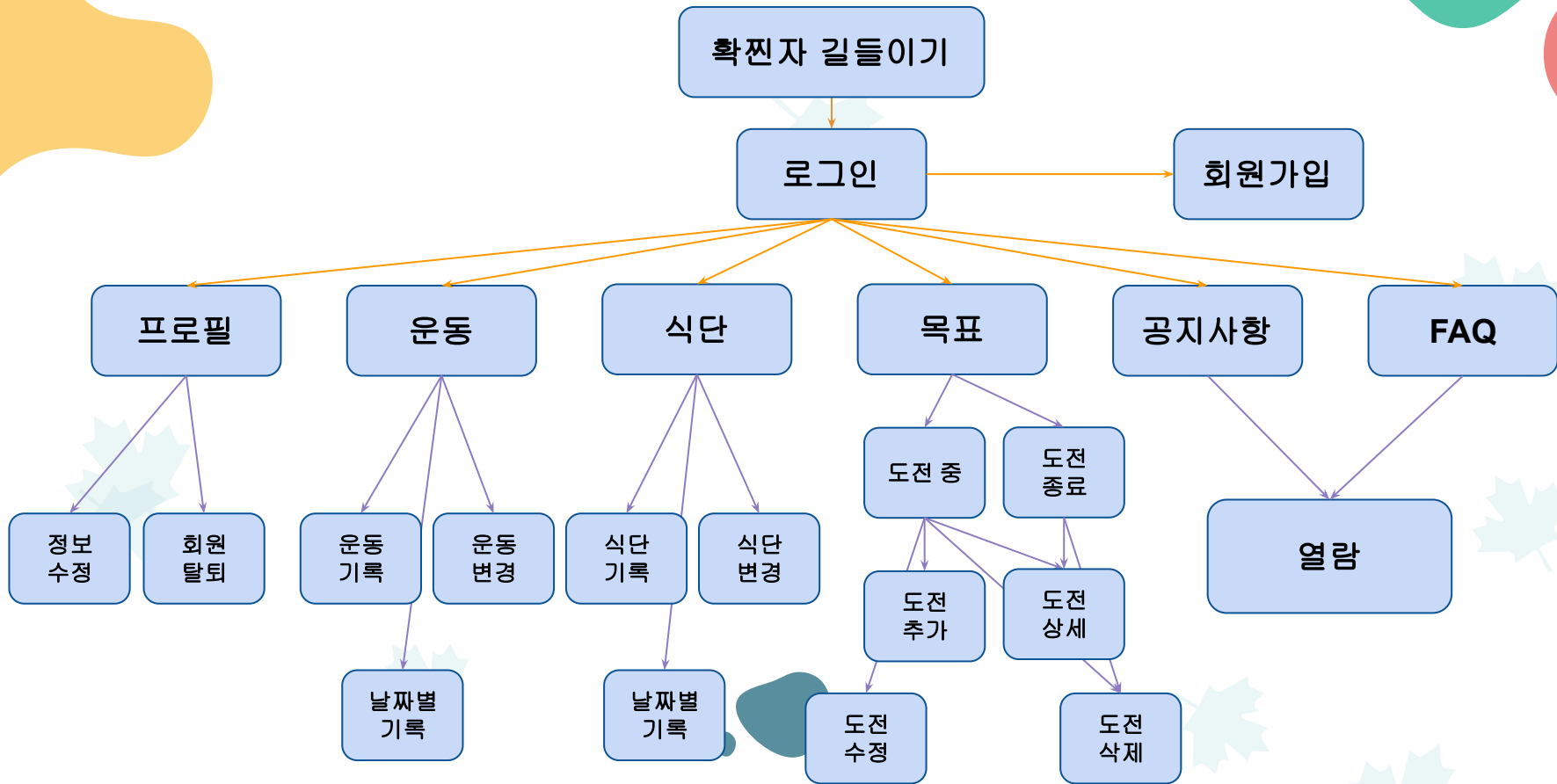
하루 간의 자신의 걸음 측정 가능
(X / 10000으로 진척도 확인)



물 섭취량

하루 간의 물 섭취량 확인 가능
(성인 기준 2L, 한 잔당 200ml 10잔 권장)

정보 설계



디자인 설계



사용자 시각적 피로도 고려



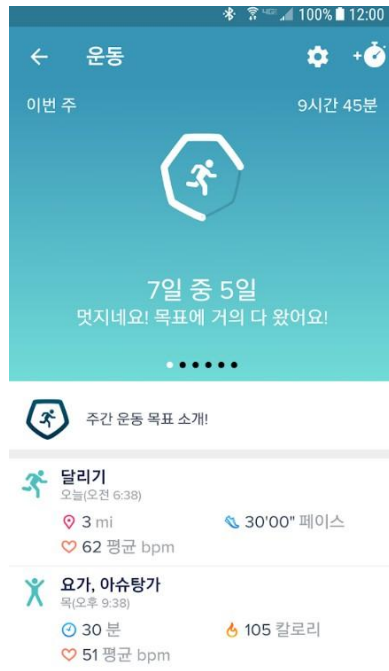
원색을 지양
파스텔 톤 색상 선택

직관적 UI 구성



사용자 연령층 고려
직관적 문구 사용

벤치마킹 앱 - Fitbit



차별화 전략

저체중
확찐자

정상
안찐자

과체중
좀찐자

비만
확찐자

고도비만
확!찐자

재미를 위해 BMI수치별 척도를 다르게 표현



03

기술 아키텍처

개발 환경

Android Studio
Eclipse java EE ide for Web
Developers

통합 개발 도구

JAVA
Spring 5.0.7
Maven MVC

사용 언어 및 프레임워크

Window 10

운영체제

MySQL 8.0.21
workbench

데이터베이스

CircleProgressBar
Retrofit
Gson

라이브러리

Github
Sourcetree

버전 관리

시스템 구성도

로그인

회원가입

SERVICE

운동

운동 기록 및 관리
총 칼로리 소모 표시

음식

식사 기록 및 관리
시간대 별 칼로리
섭취 내역 확인

목표

목표 설정 및 Dday
확인
종료된 목표의 성공
여부 기록

데일리

오늘 하루의 운동,
식사, 걸음 수, 물
섭취량 확인
진행중인 목표 확인

DATABASE

Exercise

Meal

Goal

Daily

Member

메뉴 구조도

메뉴구조도						
Depth1	Depth2	Depth3	Depth4	Depth5	Depth6	비고
메인	메뉴	목표리스트	추가			메인 진행중 목표 표시 텍스트 클릭시 이동가능
			진행중	상세	변경 삭제	
			종료	상세	성공 여부 선택 삭제	
		프로필	정보수정			메인 체중 및 BMI 표시 텍스트 클릭시 이동가능
		공지사항	상세			
		FAQ				
	운동	추가				날짜별 조회 가능
		상세	수정 삭제			
	음식	추가				날짜별 조회 가능 / 4가지 분류로 나눠서 관리
		수정				

04

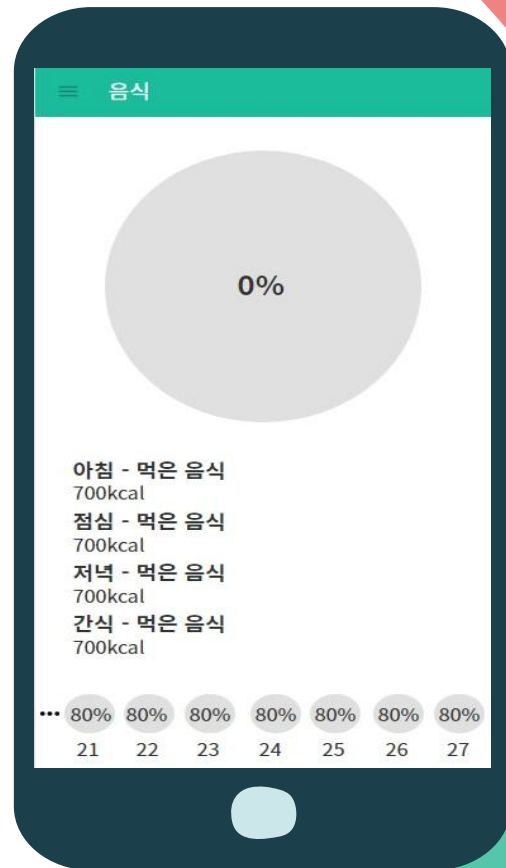
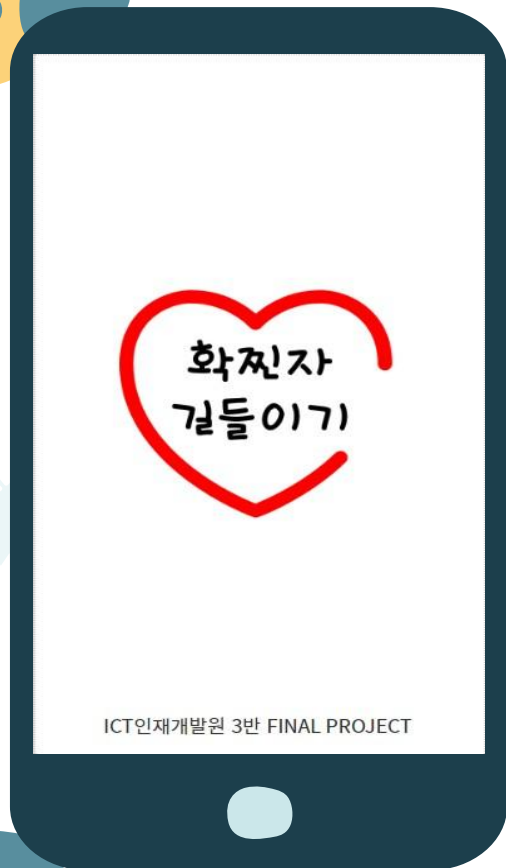
프로젝트 모델링



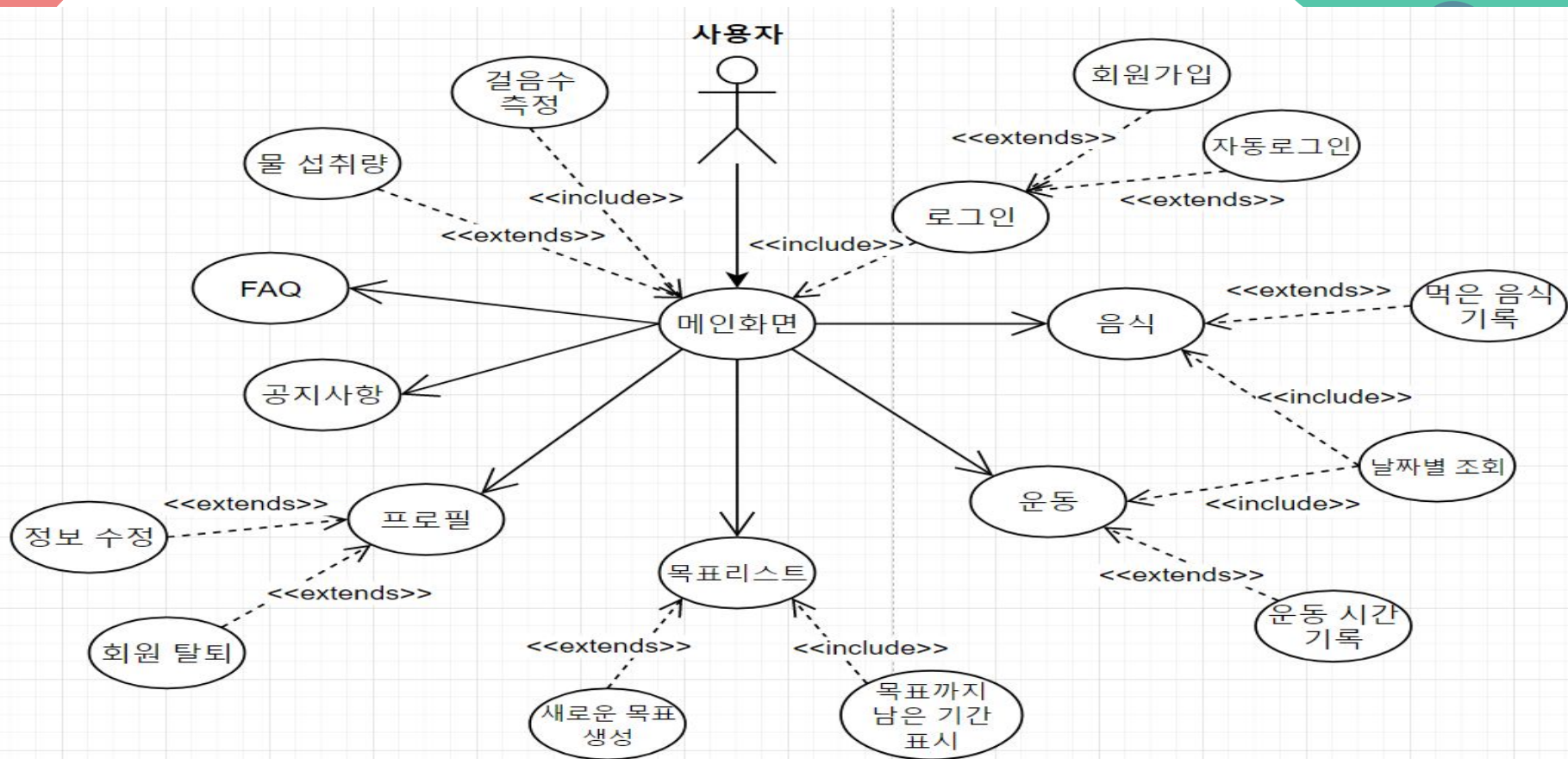
요구사항 정의

구분	특징	비고
데일리	만보기, 물 섭취량, 하루 권장 섭취량 표시, 오늘 소모한 운동 칼로리	
운동	운동 추가, 운동 삭제, 운동 변경, 날짜별 기록 열람, 총 소모 칼로리 표시	하단 버튼으로 최근 일주일 간의 기록 간편 클릭 캘린더로 날짜 선택 가능
음식	음식 추가, 음식 삭제, 음식 변경, 날짜별 기록 열람, 총 섭취 칼로리 표시	하단 버튼으로 최근 일주일 간의 기록 간편 클릭 캘린더로 날짜 선택 가능 사용자별 몸무게에 맞는 권장 칼로리 섭취량 제시
목표	도전 중, 도전 종료로 분할 도전 추가, 도전 삭제, 도전 변경, Dday 표시	도전 종료된 목표는 성공 여부 선택 가능 성공 여부 선택시 색상 변경 D-7부터 날짜 강조

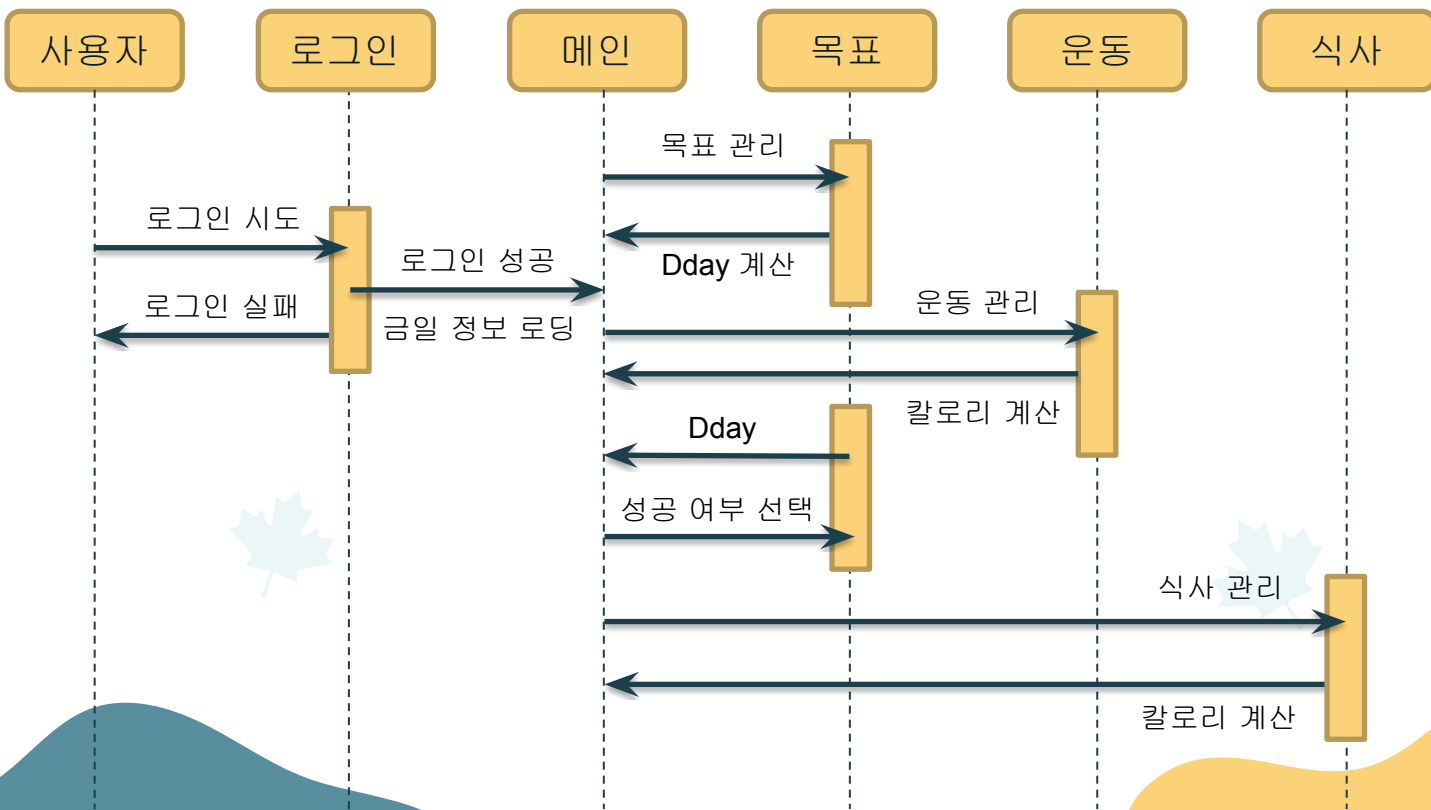
와이어프레임



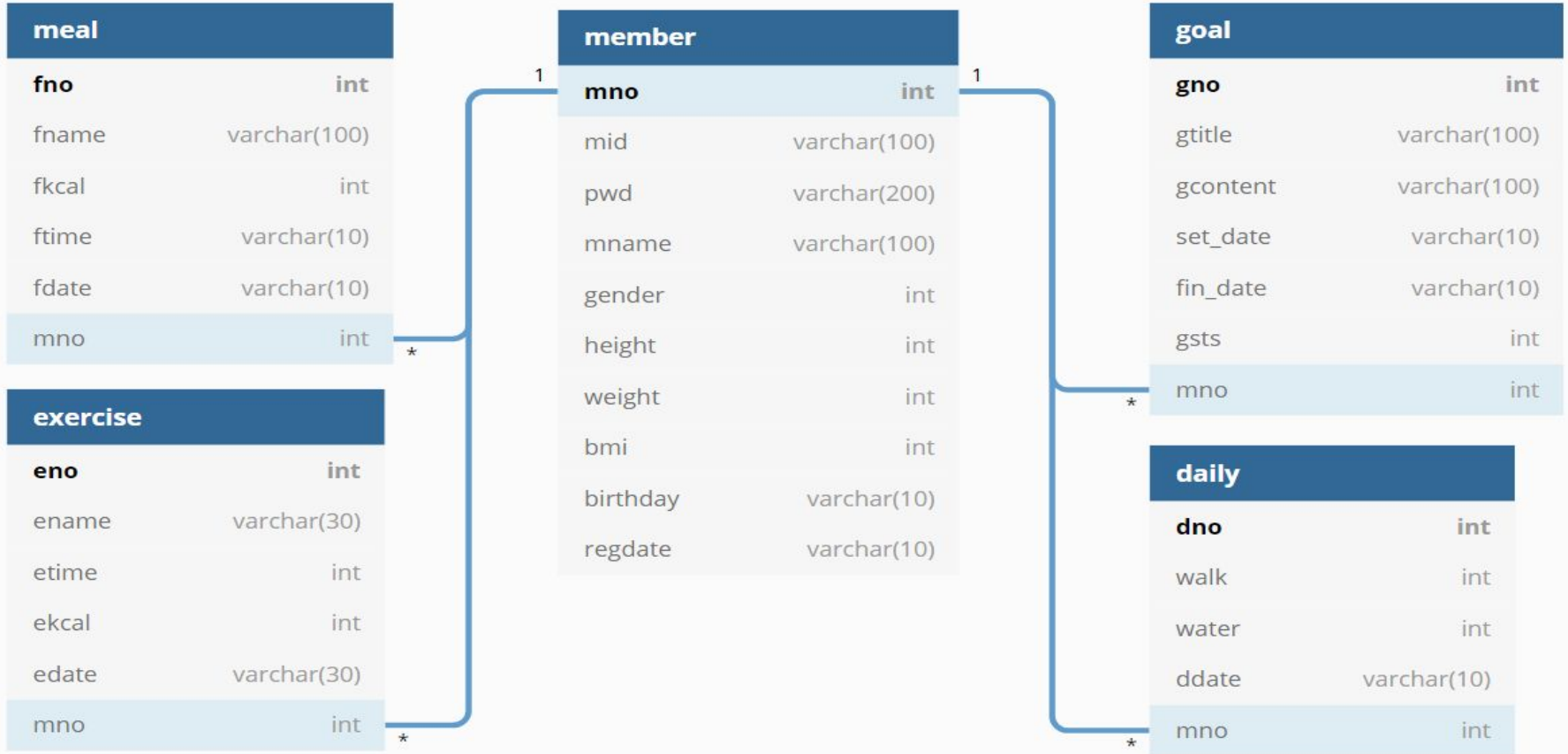
유스케이스 다이어그램



시퀀스 다이어그램

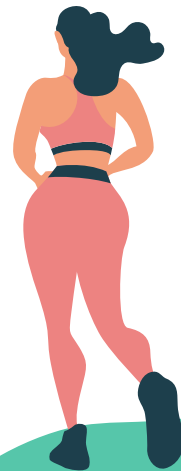
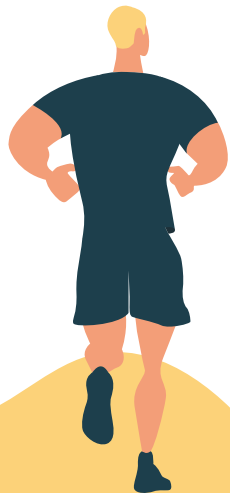


ERD



05

화면 구성 및 시연



로그인

7:53

화짜자
길들이기

아이디

비밀번호

☐ 자동 로그인

로그인 회원가입

7:55

← 회원가입

성별 : ☒ 남성 ☐ 여성

지수

jisoo97 중복확인

사용 가능한 아이디입니다.

....

....

158 46

1996 6 02

생년월일 : 1997 7 03

1998 8 04

취소 회원가입

8:00

☰ 확찐자 길들이기

진행중인 목표

스쿼트 하기 D-4

걸음 수
0 / 10000

음식 0 / 1479kcal

운동 0 kcal

46kg 확찐자

물 섭취량 5 / 10

1잔당 200ml 기준 현재 약 : 1000 ml

메인

8:02

☰ 확찐자 길들이기

지수

1997-08-29

158cm

46kg

확찐자 (18)

프로필

목표 리스트

공지사항

FAQ

로그아웃

D-4

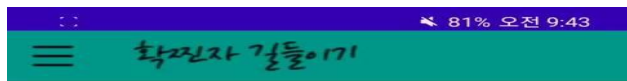
479kcal

0 kcal

찐자

메인

(만보기)



진행중인 목표

목표 이름 Dday

걸음 수
1 / 10000

음식 0 / 0kcal

운동 0 kcal

0kg 확찐자

물 섭취량 0 / 10

1잔당 200ml 기준

+ -

현재 약 : 0ml



프로필

8:11

← 프로필

지수

♀

여성

↕

158cm

KG

46kg

🎂

1997-08-29

📅

2021-06-03

☰

5개의 목표 중
0개 성공

회원탈퇴

내정보 수정

1:15

← 프로필

지수

○

여성

회원탈퇴

회원탈퇴를 원하신다면
비밀번호를 입력해주세요.

.....

취소

확인

회원탈퇴

내정보 수정

8:03

← 내정보 수정

☐ 남성

☒ 여성

지수

비밀번호

비밀번호확인

158

46

2020

5

02

생년월일 :

2021

6

03

2022

7

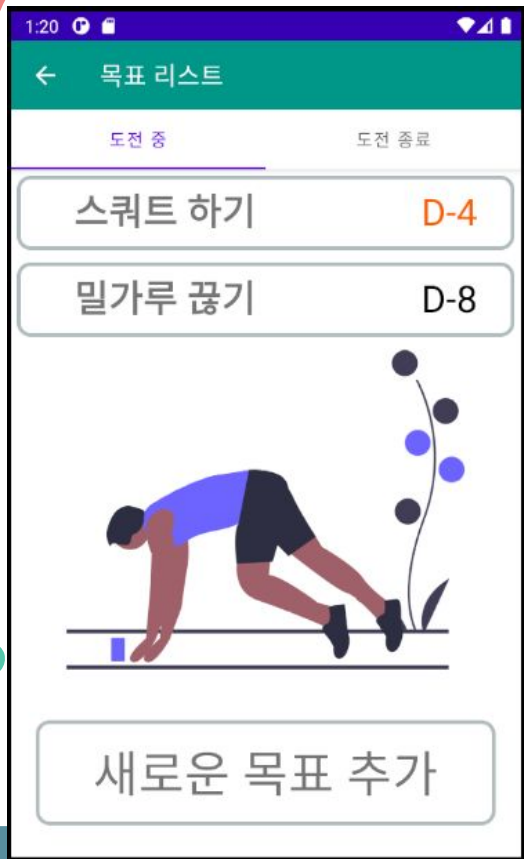
04

취소

수정하기

목표

(도전 중)



목표

(도전 종료)



식사

8:17

← 식단

197%

일일 권장량 1479kcal 중 2922kcal 섭취

아침 - 음식을 선택해주세요.
총 0kcal

점심 - 레드콤보, 삼겹살, 밥
총 2922kcal

저녁 - 음식을 선택해주세요.
총 0kcal

간식 - 음식을 선택해주세요.
총 0kcal

... 05/28 05/29 05/30 05/31 06/01 06/02 06/03

8:17

← 식단 음식선택

선택된 음식

직접 추가하기

선택되지 않은 음식

밥

삼겹살

레드콤보

배추김치

취소 완료

8:17

← 식단

날짜 선택

2021년
6월 3일 (목)

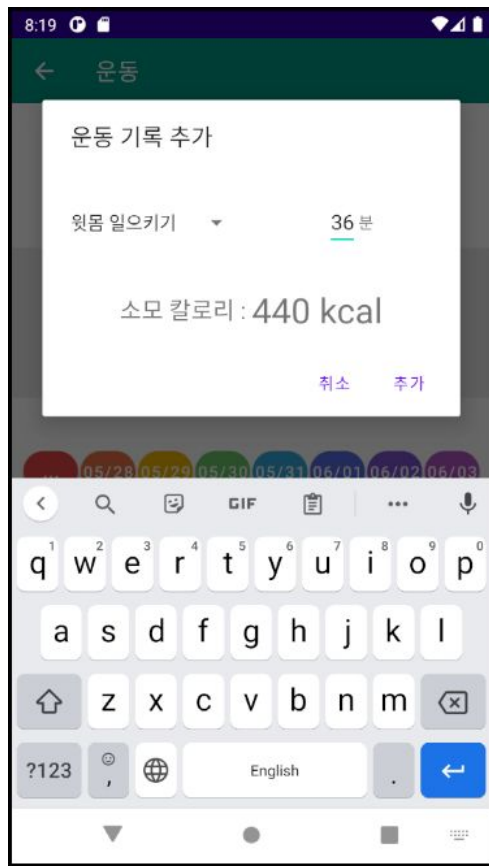
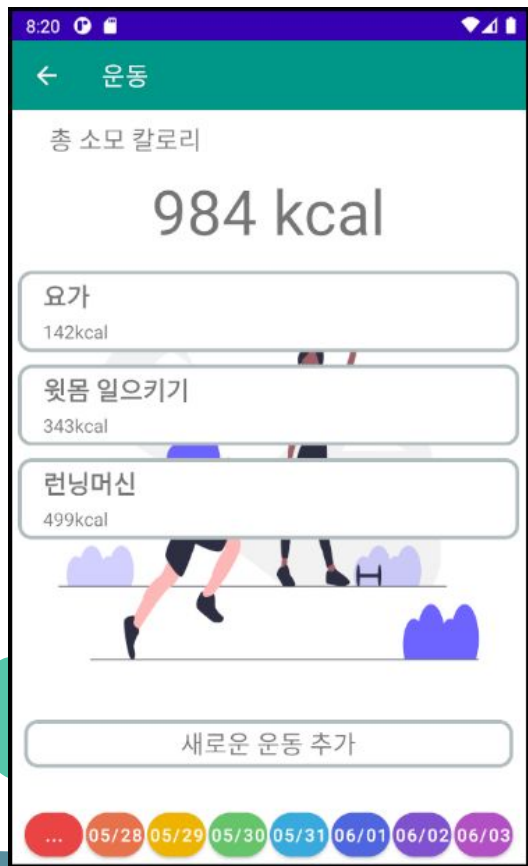
< 2021년 6월 >

일	월	화	수	목	금	토
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

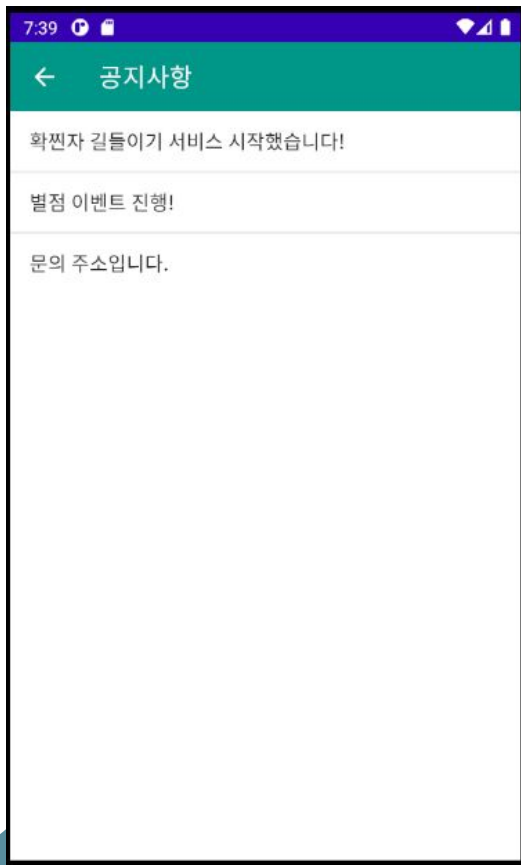
취소 확인

... 05/28 05/29 05/30 05/31 06/01 06/02 06/03

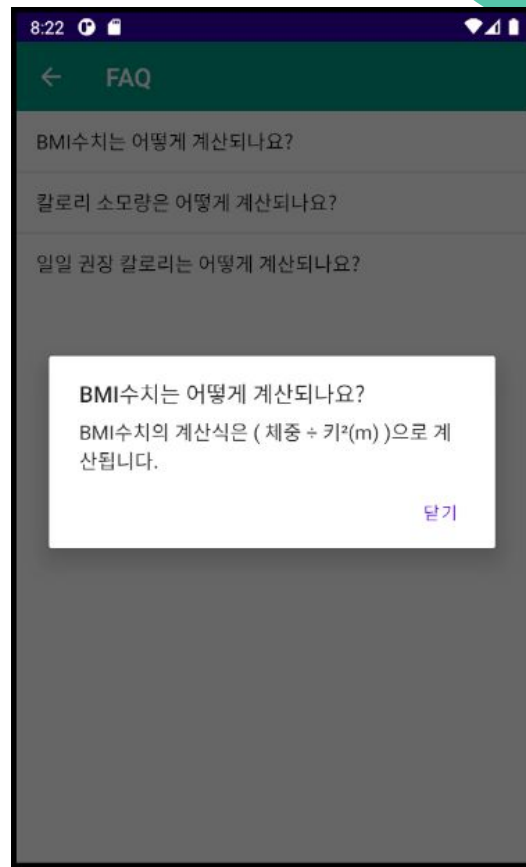
운동



공지사항



FAQ



어플리케이션 시연



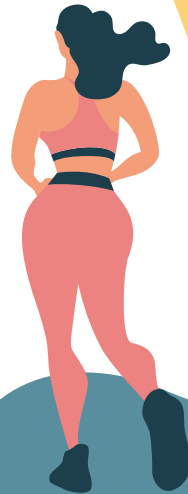
06

추가 설명



(

is



주요 소스 코드

(Transactional 활용)

물 섭취량 0 / 10



1잔당 200ml 기준

현재 약: 0 ml

```
31 @Transactional
32 @Override
33 public Integer plusWater(String ddate, int mno) {
34     mapper.updateWater(+1, ddate, mno);
35     return mapper.getDailyWater(ddate, mno);
36 }
37
38 @Transactional
39 @Override
40 public Integer minusWater(String ddate, int mno) {
41     mapper.updateWater(-1, ddate, mno);
42     return mapper.getDailyWater(ddate, mno);
43 }
```

주요 소스 코드

(Retrofit 사용)

```
@GET("exer/{edate}/{mno}")
Call<List<ExerciseVO>> getDailyExer(@Path("edate") String edate, @Path("mno") int mno);
```

```
11 private static RetrofitClient instance = null;
12 private static RetrofitInterface retrofitInterface;
13 private static String baseUrl = "http://10.0.2.2:8181/";
14 private static Gson gson = new GsonBuilder().setLenient().create();
15
16 private RetrofitClient() {
17     Retrofit retrofit = new Retrofit.Builder()
18         .baseUrl(baseUrl)
19         .addConverterFactory(GsonConverterFactory.create(gson))
20         .build();
21     retrofitInterface = retrofit.create(RetrofitInterface.class);
22 }
23
24 public static RetrofitClient getInstance() {
25     if(instance == null) {
26         instance = new RetrofitClient();
27     }
28     return instance;
29 }
30
31 public static RetrofitInterface getRetrofitInterface() { return retrofitInterface; }
32 }
```

```
26 @GetMapping(value="/{edate}/{mno}", produces = {MediaType.APPLICATION_JSON_UTF8_VALUE})
27 public ResponseEntity<List<ExerciseVO>> getExerList(@PathVariable("edate") String edate,
28                                                     @PathVariable("mno") int mno){
29
30     ResponseEntity<List<ExerciseVO>> entity = null;
31
32     try {
33         entity = new ResponseEntity<List<ExerciseVO>>(service.getAll(edate, mno), HttpStatus.OK);
34     } catch (Exception e) {
35         e.printStackTrace();
36         entity = new ResponseEntity<List<ExerciseVO>>(HttpStatus.BAD_REQUEST);
37     }
38     return entity;
39 }
```

```
retrofitClient = RetrofitClient.getInstance();
retrofitInterface = RetrofitClient.getRetrofitInterface();
retrofitInterface.getDailyExer(date, mno).enqueue(new Callback<List<ExerciseVO>>() {
    @Override
    public void onResponse(Call<List<ExerciseVO>> call, Response<List<ExerciseVO>> response) {
        if (response.isSuccessful()) {
            exerList = response.body();
            adapter = new RecyclerViewAdapter(exerList);
            recyclerView3.setAdapter(adapter);
            adapter.notifyDataSetChanged();
        } else {
            Toast.makeText(context: ExerciseActivity.this,
                text: "시스템 에러가 발생했습니다. 잠시 후 다시 시도해 주세요.",
                Toast.LENGTH_SHORT).show();
        }
    }
    @Override
    public void onFailure(Call<List<ExerciseVO>> call, Throwable t) {
        Toast.makeText(context: ExerciseActivity.this,
            text: "시스템 에러가 발생했습니다. 잠시 후 다시 시도해 주세요.",
            Toast.LENGTH_SHORT).show();
    }
});
```


주요 소스 코드

(Background Service - 만보기)

```
public class BackgroundWalking extends Service implements SensorEventListener {
```

```
RetrofitClient retrofitClient;  
RetrofitInterface retrofitInterface;
```

```
SensorManager sensorManager;  
Sensor stepDetectorSensor;
```

```
private static int mno;
```

```
@Override
```

```
public int onStartCommand(Intent intent, int flags, int startId) {
```

```
sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
stepDetectorSensor = sensorManager.getDefaultSensor(Sensor.TYPE_STEP_DETECTOR);
sensorManager.registerListener(listener: this, stepDetectorSensor, SensorManager.SENSOR_DELAY_FASTEST);
```

```
if(intent != null) {  
    mno = intent.getIntExtra( name: "mno", defaultValue: 0);  
} //if
```

```
return super.onStartCommand(intent, flags, startId);
```

```
@Override
```

```
public void onSensorChanged(SensorEvent sensorEvent) {
    if(sensorEvent.values[0] == 1.0f) {
        addWalk(mno);
    } //if
}
```

```
private void addWalk(int mno) {
    retrofitInterface.addWalk(mno).enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {

        } //onResponse

        @Override
        public void onFailure(Call<String> call, Throwable t) {
            Log.d( tag: "addWalk", msg: "addWalk Fail");
            t.printStackTrace();
            addDaily(mno);
        } //onFailure
    }); //countWalk
} //getWalk

} //Service
```

주요 소스 코드 (MealActivity)

```
public class MealActivity extends AppCompatActivity {

    RetrofitClient retrofitClient;
    RetrofitInterface retrofitInterface;

    LocalDateTime today = LocalDateTime.now();
    DateTimeFormatter dbFormat = DateTimeFormatter.ofPattern("yyyy-MM-dd");
    DateTimeFormatter btnFormat = DateTimeFormatter.ofPattern("MM/dd");

    String date = dbFormat.format(today);

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.meal);
        getRDI(mno);
        try { TimeUnit.MILLISECONDS.sleep( timeout: 100); } catch (InterruptedException e) { e.printStackTrace(); }
        getMealList(date, mno);

        //----- onClick 이벤트 -----
        bfl.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(getApplicationContext(), MealSelectActivity.class);
                intent.putExtra( name: "time", value: "breakfast");
                intent.putExtra( name: "date", date);
                intent.putExtra( name: "mno", mno);
                startActivityForResult(intent, requestCode: 200);
            }
        });

        moreDaysAgoBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AlertDialog.Builder dlg = new AlertDialog.Builder( context: MealActivity.this);
                dlg.setTitle("날짜 선택");
                if(meal_calendar.getParent() != null) {
                    ((ViewGroup) meal_calendar.getParent()).removeView(meal_calendar);
                }
                dlg.setView(meal_calendar);

                dlg.setPositiveButton( text: "확인", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        mealDatePick = (DatePicker) meal_calendar.findViewById(R.id.mealDatePick);
                        date = mealDatePick.getYear()+"-"+String.format("02d", mealDatePick.getMonth()+1)+"-"+String.format("02d", mealDatePick.getDayOfMonth());
                        getMealList(date, mno);
                    }
                });

                dlg.setNegativeButton( text: "취소", listener: null);
                dlg.show();
            }
        });
    }

    //onCreate
```

```
private void getMealList(String date, int mno) {
    retrofitInterface.getDailyMeal(date, mno).enqueue(new Callback<List<MealVO>>() {
        String breakfast = null, lunch = null, dinner = null, desert = null;
        int breakfastKcal = 0, lunchKcal = 0, dinnerKcal = 0, desertKcal = 0;

        @Override
        public void onResponse(Call<List<MealVO>> call, Response<List<MealVO>> response) {
            progress = 0;
            Log.d( tag: "load", msg: "불러오는중");
            response.body().forEach(mealVO -> {
                switch(mealVO.getTime()) {
                    case "breakfast":
                        if(breakfast != null) {breakfast += " ";}
                        breakfast += mealVO.getName();
                        breakfastKcal += mealVO.getFkcal();
                        break;

                }
                progress += mealVO.getFkcal();
            });

            getRDI(mno);
            int per = (int)(progress/rdi*100);
            circlebar.setProgress(per);
            all.setText("일일 권장량 " + (int)rdi + "kcal 중 " + progress + "kcal 섭취");
        }

        @Override
        public void onFailure(Call<List<MealVO>> call, Throwable t) {
            Toast.makeText(getApplicationContext(), text: "시스템 에러가 발생했습니다. 잠시 후 다시 시도해 주세요.", Toast.LENGTH_SHORT).show();
            t.printStackTrace();
        }
    });
}

private void getRDI(int mno) {
    retrofitInterface.getInfo(mno).enqueue(new Callback<MemberVO>() {
        @Override
        public void onResponse(Call<MemberVO> call, Response<MemberVO> response) {
            MemberVO info = response.body();
            do{
                rdi = ((info.getHeight()-100) * (info.getGender() == 1 ? 0.9f : 0.85f) * 30);
            }while(rdi == 0);

            @Override
            public void onFailure(Call<MemberVO> call, Throwable t) {
                Toast.makeText(getApplicationContext(), text: "시스템 에러가 발생했습니다. 잠시 후 다시 시도해 주세요.", Toast.LENGTH_SHORT).show();
                t.printStackTrace();
            }
        }
    });
}
```

버전 관리

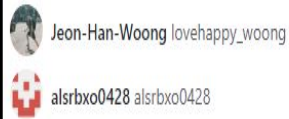
Android Studio

Jeon-Han-Woong / BodyChecker_Android

Jeon-Han-Woong 2021-06-04 BodyChecker README e2a2590 yesterday 143 commits

.idea	2021/05/08 AM 12:41 BodyChecker	29 days ago
app	공지사항 글 수정	yesterday
gradle/wrapper	2021/04/29 BodyChecker	last month
.gitignore	2021/04/29 BodyChecker	last month
README.md	2021-06-04 BodyChecker README	yesterday
build.gradle	2021-05-18 민규태	18 days ago
gradle.properties	2021/04/29 BodyChecker	
gradlew	2021/04/29 BodyChecker	
gradlew.bat	2021/04/29 BodyChecker	
settings.gradle	2021/04/29 BodyChecker	

Contributors 2



README.md

Android Studio

건강 관리 앱 프로젝트 - 확편자 길들이기

진행내용 2021/06/04 - Merge & Collaborator

전한웅 & 민규태

- 일부 오류 수정
- 레이아웃 디자인 개선
- PPT 작업


NEXT









테스트, 발표

버전 관리

Rest Server

Jeon-Han-Woong / BodyChecker_Retrofit

 Jeon-Han-Woong Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker_Retrofit 275d74c yesterday 76 commits

	.settings	프로젝트 이름 변경	16 days ago
	src	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker_Retrofit	yesterday
	target	삭제된 walk 메소드 기능 복구	yesterday
	.classpath	Retrofit Ready	20 days ago
	.gitignore	최근 Eno를 받아오기 위한 getEno 생성	
	.project	프로젝트 이름 변경	
	.springBeans	Retrofit Ready	
	pom.xml	2021-05-25 민규태	

Contributors 2



Jeon-Han-Woong lovehappy_woong



alsrbxo0428 alsrbxo0428

버전 관리

Github / Sourcetree

The screenshot shows the Sourcetree application window. The top menu bar includes File (F), Edit (E), View (V), Settings (S), Tools (T), and Help (H). The toolbar contains icons for Cut, Pull, Push, Patch, Branch, Merge, Stash, Queue, and Tag. The main workspace is divided into three panes: a file explorer on the left, a commit history graph in the center, and a commit details pane on the right. The commit history graph shows a series of commits on the master branch, with the most recent commit being '2021-06-04 BodyChecker REA' by Jeon-Han-Woong. The commit details pane shows the commit message, author, and a list of files changed.

Commit Hash	Message	Date	Author	SHA-1
e2a2590	공지사항 글 수정	4 6 2021 15:10	Jeon-Han-Woong	80a4be5
4d3df82	MainActivity	4 6 2021 14:28	alsrbxo0428	4d3df82
2208511	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	4 6 2021 12:49	alsrbxo0428	2208511
49e1860	2021-06-04 민규태 회원가입 아이디 중복체크 문제 해결 회원탈퇴 기능	4 6 2021 12:49	alsrbxo0428	49e1860
276af4d	운동 / 목표 코드 리팩토링 - Woong	4 6 2021 12:46	Jeon-Han-Woong	276af4d
440daed	2021-06-03 BodyChecker README	3 6 2021 18:08	Jeon-Han-Woong	440daed
e00f7b5	2021-06-03 민규태 MainActivity -> Service 정보 보내기	3 6 2021 18:02	alsrbxo0428	e00f7b5
15686db	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	3 6 2021 17:12	alsrbxo0428	15686db
d50b0b8	버튼 색상 변경	3 6 2021 17:12	Jeon-Han-Woong	d50b0b8
9659568	2021-06-03 민규태 만보기 기능 구현	3 6 2021 17:11	alsrbxo0428	9659568
19e8f16	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	3 6 2021 12:31	Jeon-Han-Woong	19e8f16
577a796	액티비티 패키지 정리	3 6 2021 12:27	Jeon-Han-Woong	577a796
2304709	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	3 6 2021 12:26	alsrbxo0428	2304709
6d8f7b1	2021-06-03 민규태 만보기기을 위한 서비스클래스 생성 클래스파일 주:	3 6 2021 12:26	alsrbxo0428	6d8f7b1
b6a0d4c	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	2 6 2021 18:14	Jeon-Han-Woong	b6a0d4c

The screenshot shows the Sourcetree application window. The top menu bar includes File (F), Edit (E), View (V), Settings (S), Tools (T), and Help (H). The toolbar contains icons for Cut, Pull, Push, Patch, Branch, Merge, Stash, Queue, and Tag. The main workspace is divided into three panes: a file explorer on the left, a commit history graph in the center, and a commit details pane on the right. The commit history graph shows a series of commits on the master branch, with the most recent commit being '오류 수정' by Jeon-Han-Woong. The commit details pane shows the commit message, author, and a list of files changed.

Commit Hash	Message	Date	Author	SHA-1
fd67ec7	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	4 6 2021 14:30	Jeon-Han-Woong	275d74c
6ee77be	중간 커밋	4 6 2021 14:28	Jeon-Han-Woong	6ee77be
1bb40bd	삭제된 walk 메소드 기능 복구	4 6 2021 14:27	alsrbxo0428	1bb40bd
d45f186	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	4 6 2021 12:48	alsrbxo0428	d45f186
f23e822	2021-06-04 민규태 회원탈퇴를 위한 조회 기능 추가	4 6 2021 12:47	alsrbxo0428	f23e822
910a8f0	DailyMapper 수정	4 6 2021 12:46	Jeon-Han-Woong	910a8f0
3871107	2021-06-03 민규태 daily - getWalk, addWalk 구현	3 6 2021 17:10	alsrbxo0428	3871107
62770b3	Post로 변경	3 6 2021 12:26	Jeon-Han-Woong	62770b3
e5b33f7	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	1 6 2021 18:13	alsrbxo0428	e5b33f7
226834c	pom	1 6 2021 18:13	alsrbxo0428	226834c
6988467	메인 화면에 표시할 목표 구현 - Woong	1 6 2021 18:06	Jeon-Han-Woong	6988467
d773897	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	1 6 2021 15:14	Jeon-Han-Woong	d773897
d7d73aa	pom	1 6 2021 15:14	Jeon-Han-Woong	d7d73aa
8d167ef	중간커밋	1 6 2021 15:11	alsrbxo0428	8d167ef
6e2268a	Merge branch 'master' of https://github.com/Jeon-Han-Woong/BodyChecker	31 5 2021 18:07	alsrbxo0428	6e2268a

개선점

만보기, 물 섭취량
날짜별 열람 기능

운동 및 식사 추천 시스템

01

02

03

04

관리자 기능 활성화

웹 페이지 개설 및 연동

FINISH

Thank You for Watching!

Question / Answer

