

내가쓴다 기획서

(포트폴리오 용도)

훈련 과정	자바(JAVA)를 활용한 안드로이드 앱(APP)
개발 기간	2018.9.7 ~ 9.20
APP 명칭	내가 쓴다

작성자 : 박 한 얼

App 설명 (이벤트 추천기)

이벤트, 복불복 게임등을 이용해 주변 사람들과 즐거운 시간을 즐길 수 있는 앱입니다.

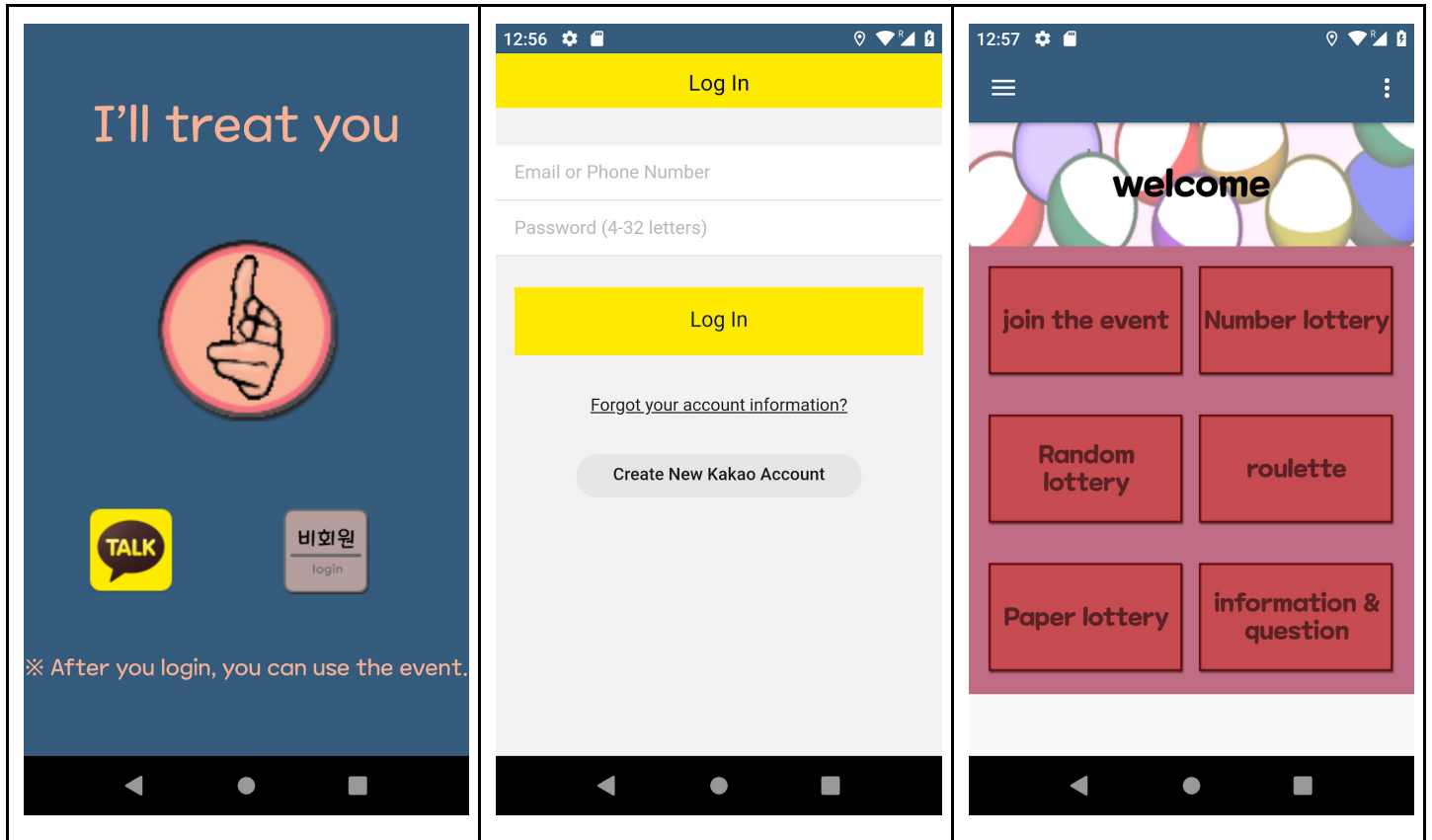
- 제작한 App을 구글 스토어에 등록하였습니다.



화면 구성

순서	화면명
1	메인 화면
2	각각의 추첨 화면
3	정보 및 문의 화면
4	종료된 리스트 확인 화면
5	관리자 확인 화면
6	개인 정보 동의 화면

스플레쉬 화면



스플레쉬 화면 설명

1. App을 실행하였을 때, 스플레쉬 화면에서 로그인을 할 수 있도록 설정하였습니다.
2. 카카오톡과 연동하여 로그인 할 수 있는 구조를 구축하였습니다.
3. 비회원으로 접속 시 메인 화면으로 바로 전환 됩니다.

핵심 코드

• 카카오톡 연동

```
private void selectImage() {
    kakaotalk.setOnClickListener((v) → {
        kakaoButton.performClick();
        //여기에 startActivity하면 performClick호출 안댐!
    });

    findViewById(R.id.imageButton_non).setOnClickListener((view) → {
        startActivity(new Intent( packageContext, SplashActivity.this, MainActivity.class));
        finish();
    });
}
```

```
private void requestMe() {

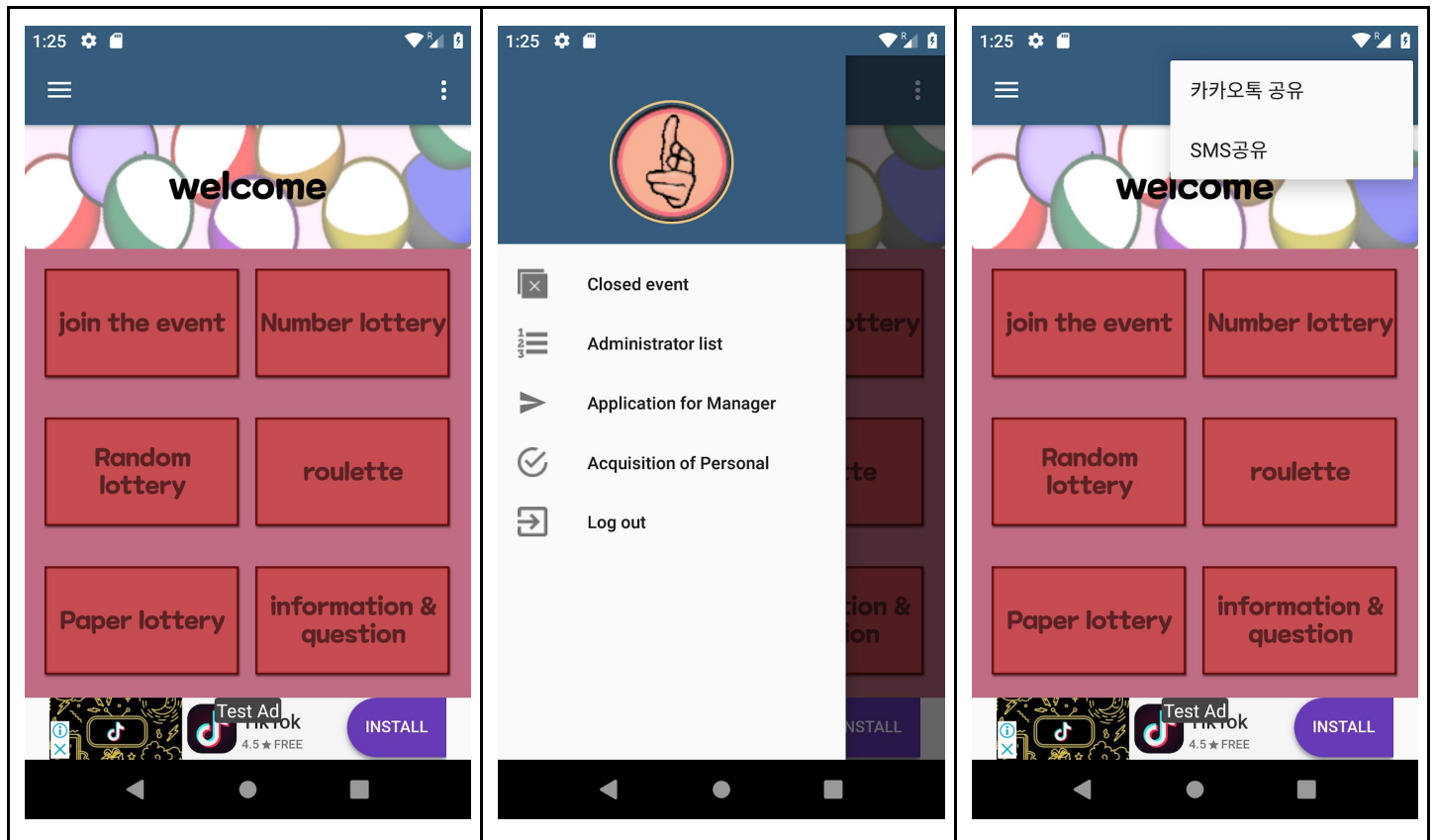
    UserManagement.getInstance().requestMe(new MeResponseCallback() {
        @Override
        public void onFailure(ErrorResult errorResult) { Log.e( tag: "onFailure", msg: errorResult + ""); }

        @Override
        public void onSessionClosed(ErrorResult errorResult) {
            Log.e( tag: "onSessionClosed", msg: errorResult + "");
        }

        @Override
        public void onSuccess(final UserProfile userProfile) {
            Log.e( tag: "onSuccess", msg: "연결 성공!");
            startActivity(new Intent( packageContext, SplashActivity.this, MainActivity.class));
        } //onSuccess

        @Override
        public void onNotSignedUp() { Log.e( tag: "onNotSignedUp", msg: "onNotSignedUp"); }
    });
} //requestme(Kakao)
```

메인 화면



메인 화면 설명

1. 6개의 기능을 한눈에 보여 줍니다.
2. 네비게이션 슬라이딩 메뉴를 통해서 구성을 세분화 하였습니다.
3. 카카오톡 과 sms 공유 기능을 제공합니다.

핵심 코드

• 메뉴 설정 및 공유 기능

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_kakao) {
        shareKakaoLink();
    } else if (id == R.id.action_SMS) {
```

```
void shareKakaoLink() { //어플공유
    try {

        FeedTemplate params = FeedTemplate
            .newBuilder(ContentObject.newBuilder( title: "내가 쓴다 어플을 지금 다운로드 받으세요!",
                imageUrl: "https://",
                LinkObject.newBuilder().setWebUrl("https://",
                    ) //마켓주소(아이콘클릭시)
                    .setMobileWebUrl("https://developers.kakao.com").build()) //마켓주소(클릭시)
                .build())
            .addButton(new ButtonObject( title: "앱에서 보기", LinkObject.newBuilder()
                .setMobileWebUrl("https://developers.kakao.com") //마켓주소(버튼클릭시)
                .setAndroidExecutionParams("key1=value1")
                .setIosExecutionParams("key1=value1")
                .build())
                .build());

        Map<String, String> serverCallbackArgs = new HashMap<>();
        serverCallbackArgs.put("user_id", "${current_user_id}");
        serverCallbackArgs.put("product_id", "${shared_product_id}");

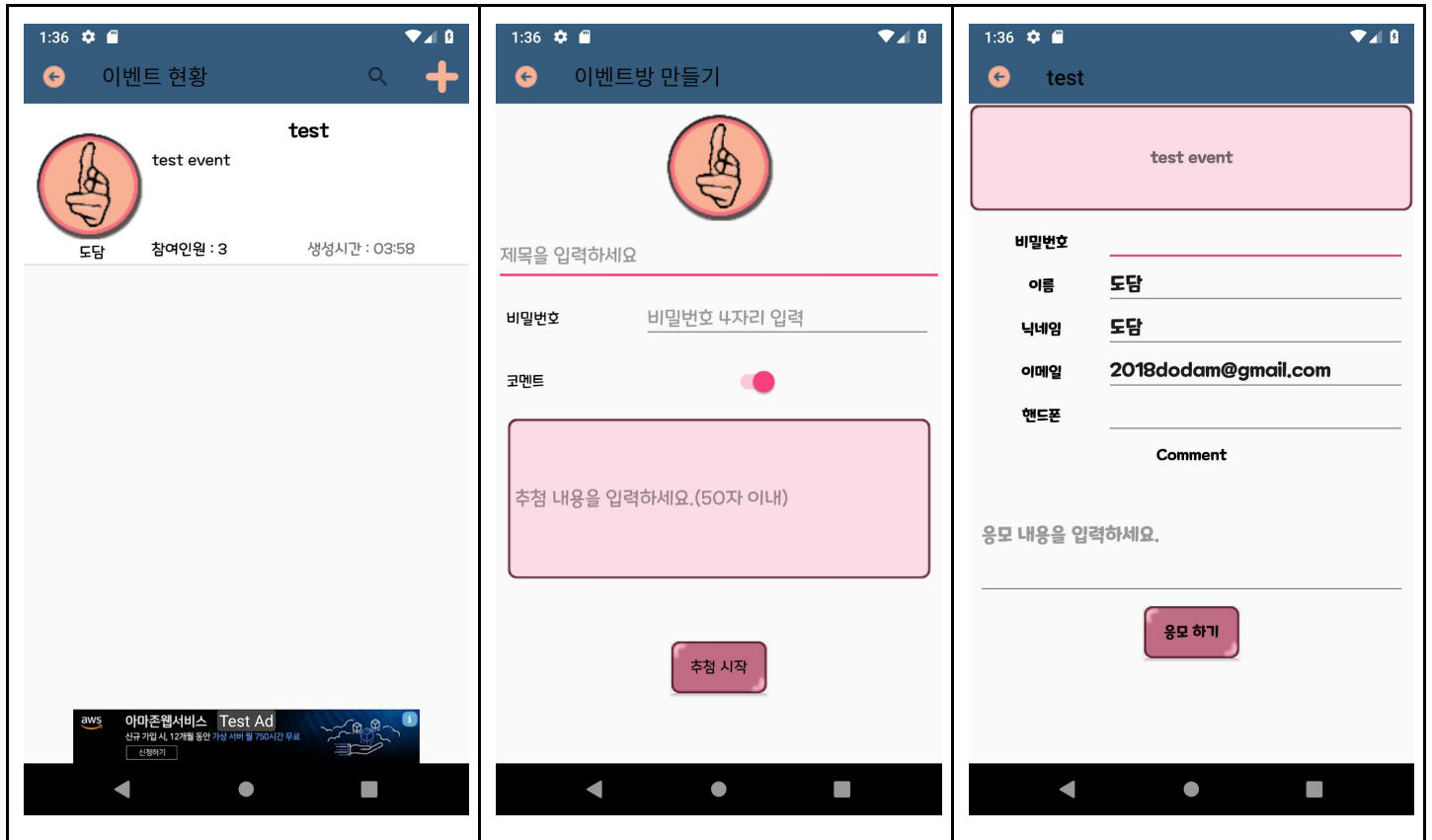
        KakaoLinkService.getInstance().sendDefault( context: this, params, serverCallbackArgs, new ResponseCallback<KakaoLinkResponse>() {
            @Override
            public void onFailure(ErrorResult errorResult) { Logger.e(errorResult.toString()); }

            @Override
            public void onSuccess(KakaoLinkResponse result) {

            }

        });
    } catch (Exception e) {
```

이벤트 기능 화면



이벤트 기능 화면 설명

1. 이벤트 개설을 위한 정보를 입력하면 실시간 서버에 저장되며, 관리자만 이벤트를 개설할 수 있도록 제한하였습니다.
2. 이벤트 개설 후 카카오톡으로 접속한 유저들이 이벤트에 참여 할 수 있습니다.
3. 이벤트에 개인정보를 입력하고 참여하였을 때 유저의 UID가 생성되어 중복 지원을 방지 하였습니다.
4. 관리자가 이벤트를 중지하고 추첨을 하게 되면 추첨된 사람의 목록을 확인 할 수 있습니다.

핵심 코드

- 서버에 생성 되어 있는 테이블에 항목을 저장합니다.

```
public void dataSave() { //데이터 저장
    myRef = FirebaseDatabase.getInstance().getReference("USER");//RealtimeDB
    myRef.addChildEventListener(new ChildEventListener() {
        @Override
        public void onChildAdded(@NonNull DataSnapshot dataSnapshot, @Nullable String s) {
            UserDTO userDTO = dataSnapshot.getValue(UserDTO.class);
            try {
                if (userUUID.equals(userDTO.getUserUUID())) {
                    if (userDTO.getManager() == null) {
                        Toast.makeText(context, "관리자 권한이 없습니다", Toast.LENGTH_SHORT).show();
                    } else {
                        String title = editTitle.getText().toString();
                        String substance = editSub.getText().toString();
                        String password = editPass.getText().toString();

                        myRef = FirebaseDatabase.getInstance().getReference("CURRENT_EVENT");//RealtimeDB

                        if (substance.length() == 0) { //코멘트추가 안하면
                            roomDTO = new RoomDTO(title, password, sub: "내용이 없습니다", userName, userUUID, eventCode: userName+time(), joinInUser: 0, time());
                            //타이틀, 비번, 내용, 관리자명, 관리자uid, 이벤트코드, 참여인원, 시간
                            myRef.push().setValue(roomDTO);
                        } else { //코멘트 추가 하면
                            roomDTO = new RoomDTO(title, password, substance, userName, userUUID, eventCode: userName+time(), joinInUser: 0, time()); //순서대로
                            //타이틀, 비번, 내용, 관리자명, 관리자uid, 이벤트코드, 참여인원, 시간
                            myRef.push().setValue(roomDTO);
                        }
                    }

                    Toast.makeText(context, "추첨을 시작 합니다.", Toast.LENGTH_SHORT).show();
                    finish();
                }
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}
```

- 데이터를 가져올때 딜레이를 주어 에러를 방지합니다.

```
new Handler().postDelayed(() -> {
    editTitle.setText(alTitle.get(LotteryActivity.position));
    editPass.setText(alPW.get(LotteryActivity.position));
    editSub.setText(alSub.get(LotteryActivity.position));
}, delayMillis: 500);
```

숫자 추첨 화면



숫자 추첨 화면 설명

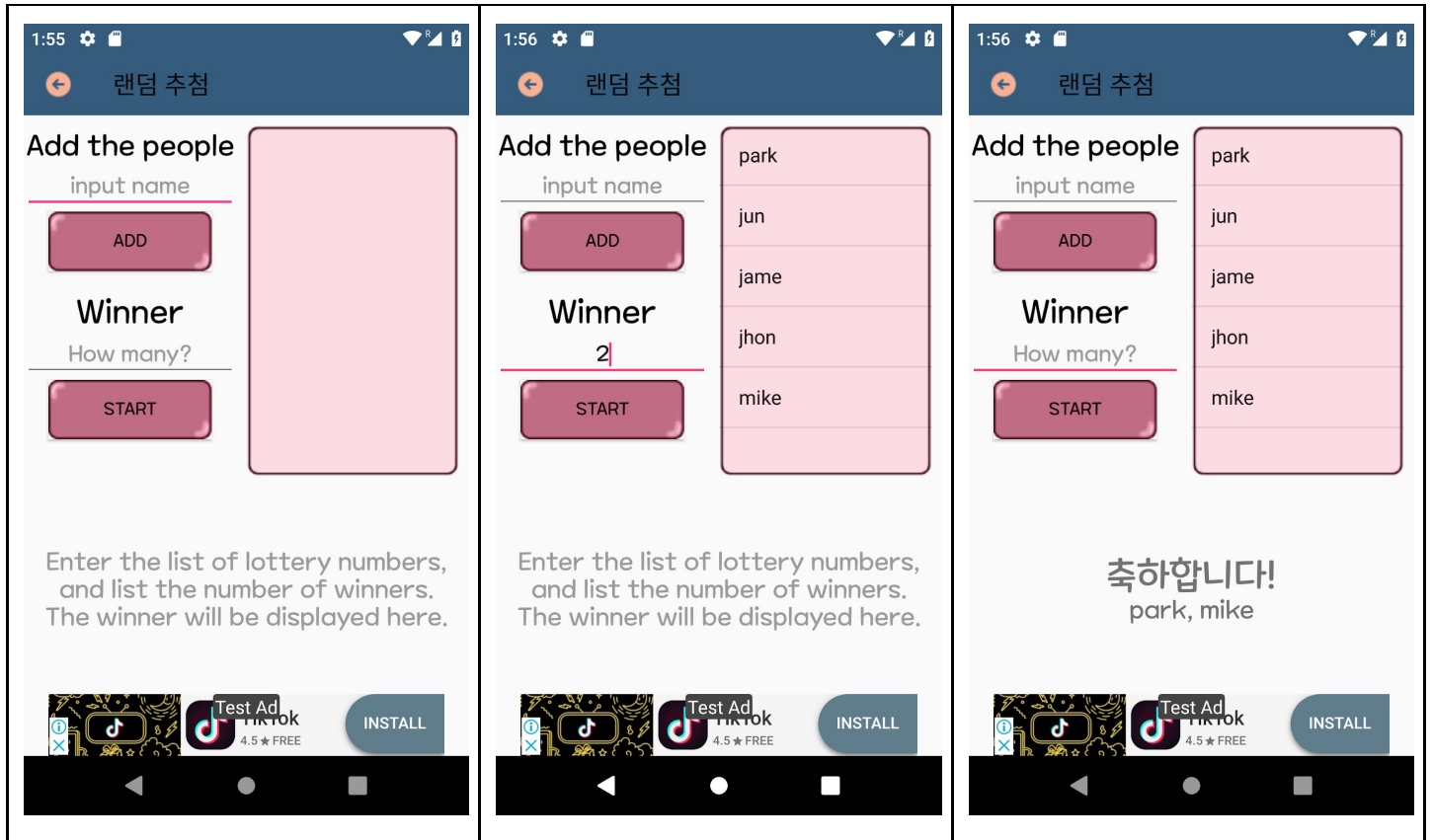
1. 최대 숫자와 추첨 갯수를 입력하여 결과를 리스트에 보여 줍니다.
2. 1부터 시작과 0부터 시작하는 스위치를 설정하였습니다.
3. 한번 입력된 값을 유지할지, 삭제 할 것인지를 설정하는 스위치가 있습니다.

핵심 코드

● 스위치를 이용한 설정 변경

```
switchMin.setOnCheckedChangeListener((compoundButton, isChecked) → {  
    if(isChecked) {  
        checkMin = true;  
        switchMin.setText("0부터 시작");  
        editMax.setHint("1~숫자입력");  
        startNumber = 0;  
    }else {  
        checkMin = false;  
        switchMin.setText("1부터 시작");  
        editMax.setHint("0~숫자입력");  
        startNumber = 1;  
    }  
}); //최소값 설정상태 확인  
switchEdit.setOnCheckedChangeListener((compoundButton, isChecked) → {  
    if(isChecked) {//on상태일때  
        check = true;  
        switchEdit.setText("입력값 초기화");  
    }else {  
        check = false;  
        switchEdit.setText("입력값 유지");  
    }  
}); //초기화 설정상태 확인
```

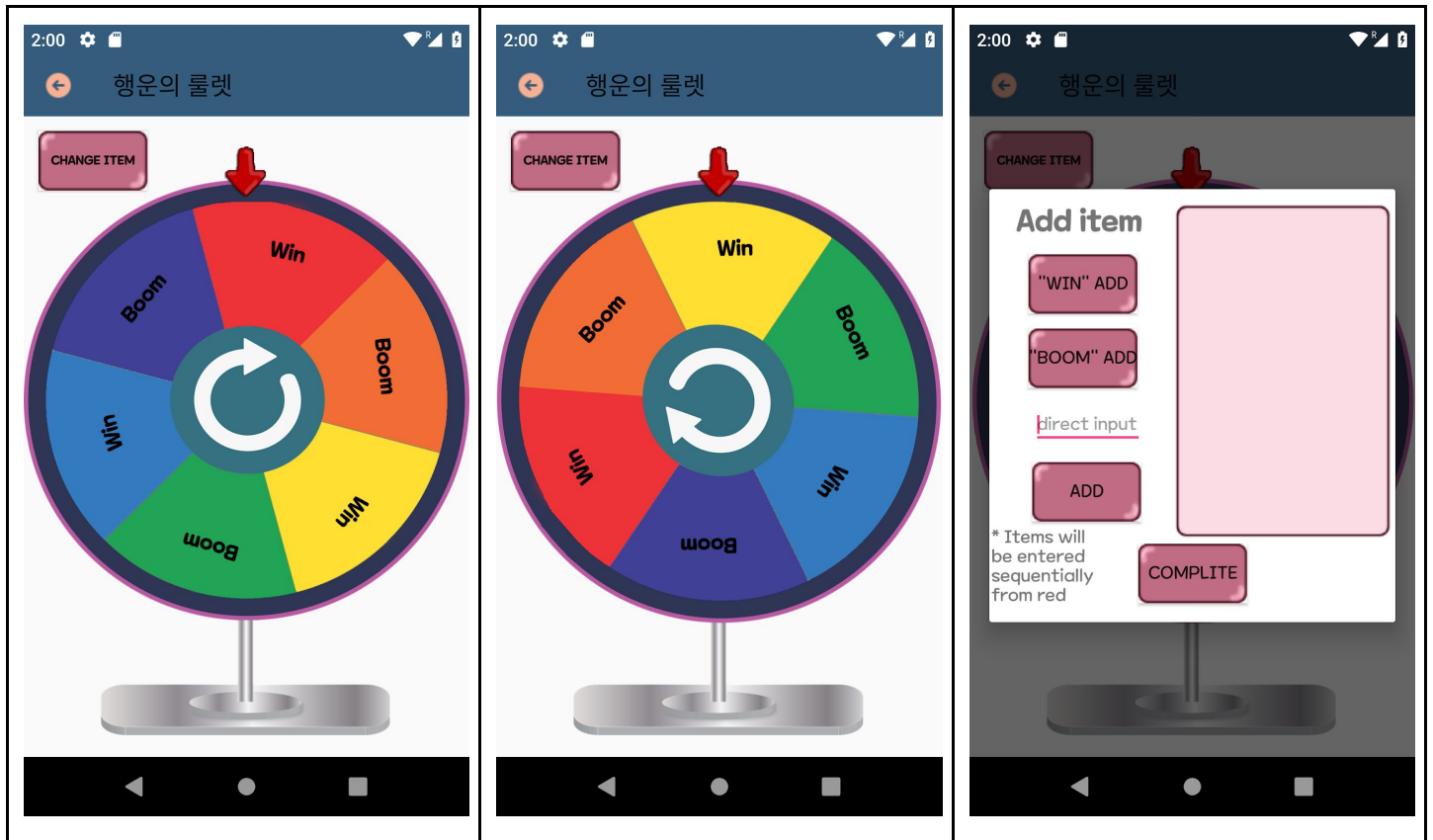
랜덤 추첨 화면



랜덤 추첨 화면 설명

1. 문자로 이를 입력하여 그 중 몇 개를 추첨하는 방식입니다.
2. 폰트 변경등을 이용하여 조금 더 아가자기한 UI를 구성하였습니다.

룰렛 화면

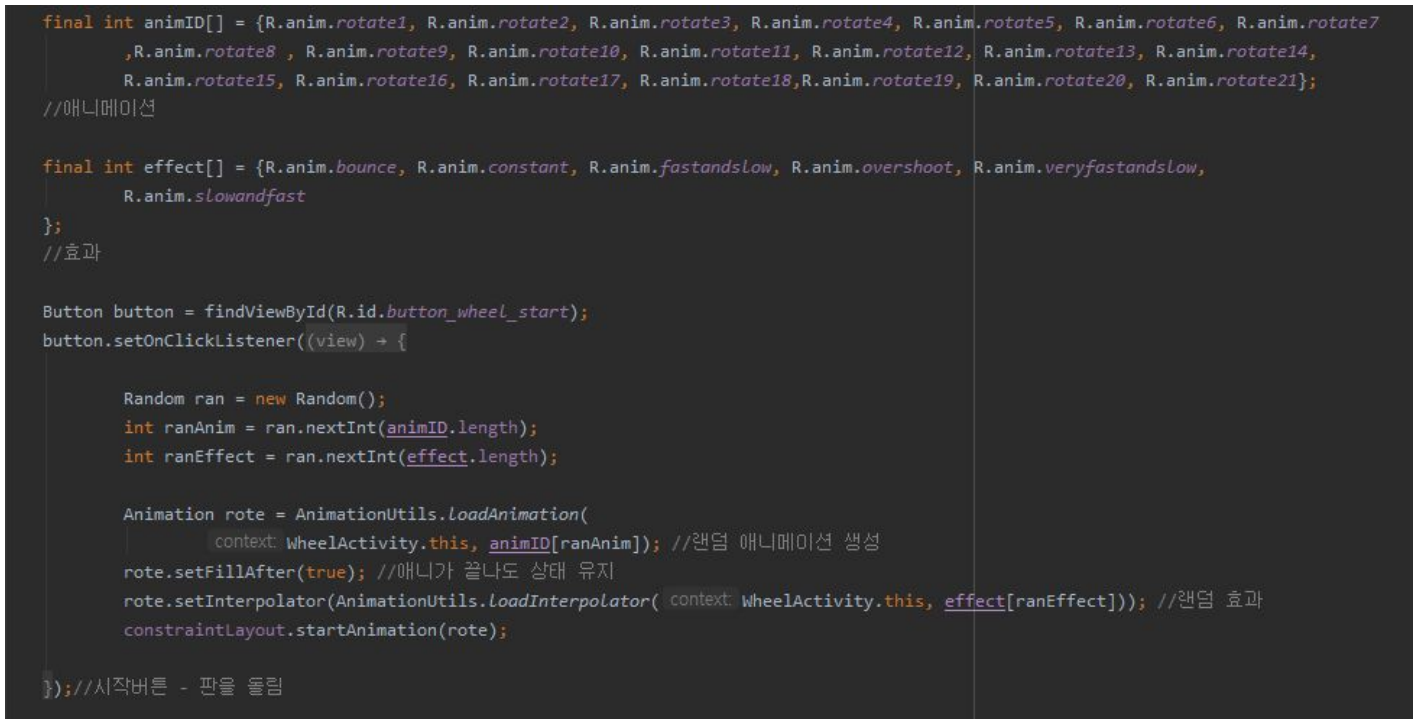
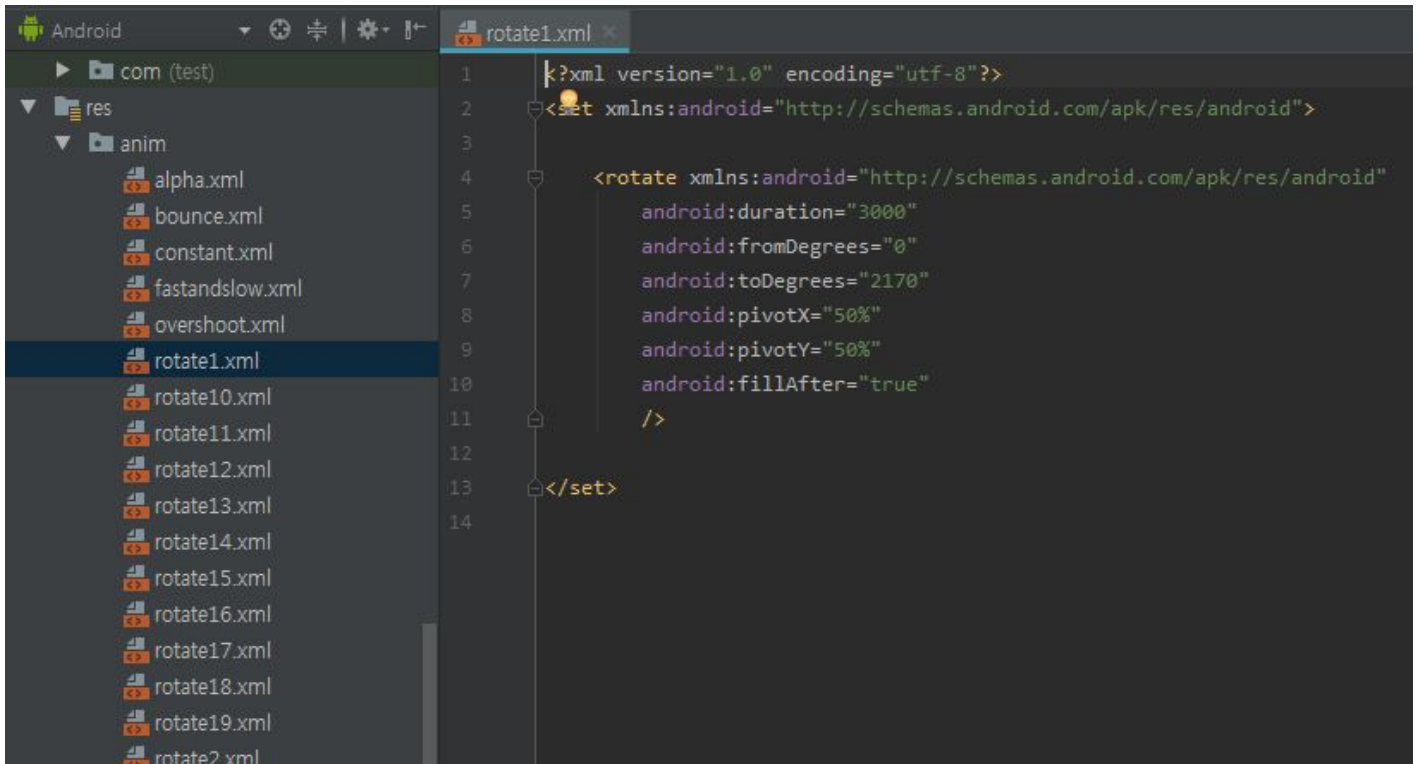


룰렛 화면 설명

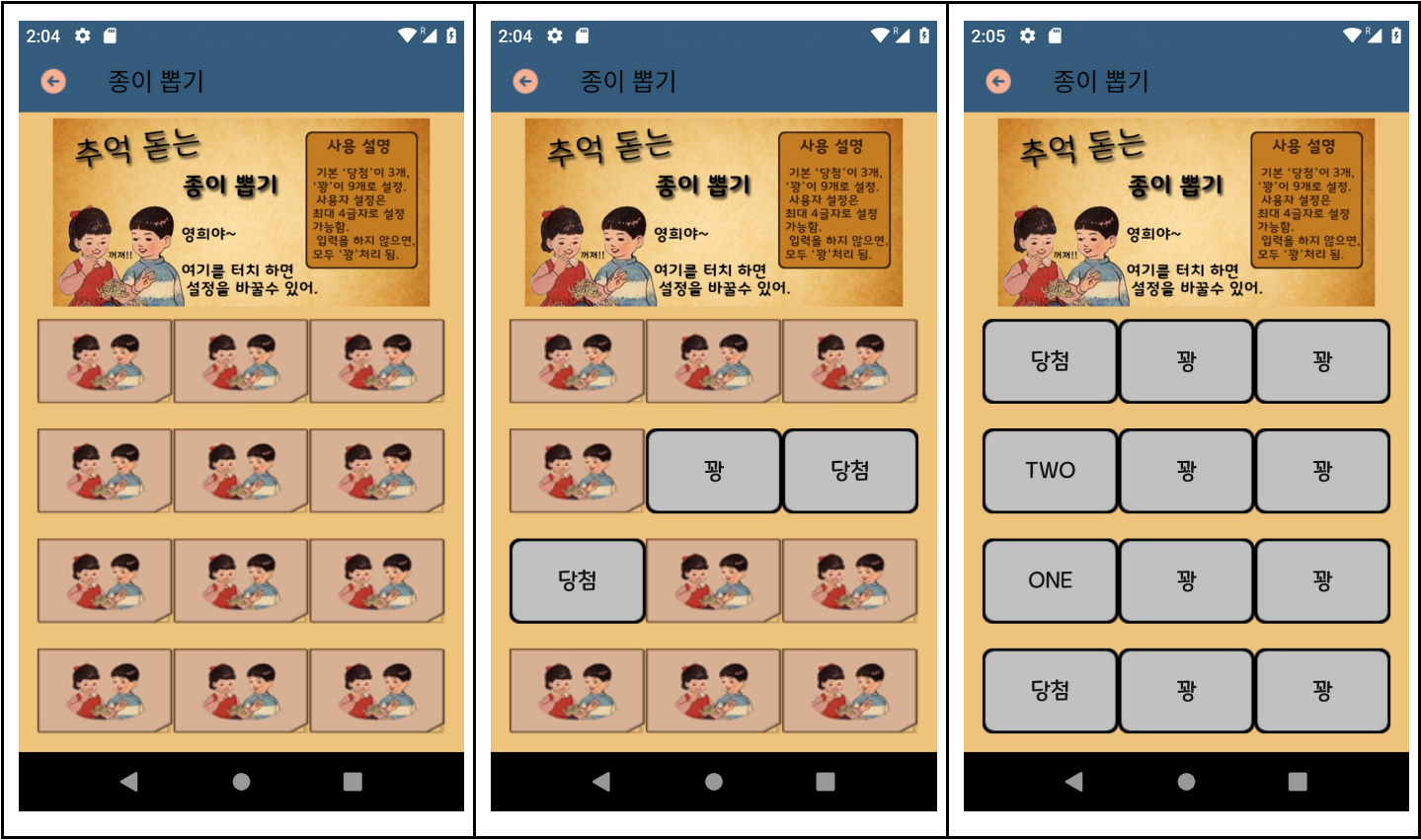
1. 애니메이션을 이용한 회전객체를 생성 회전을 통해 추첨하는 방식입니다.
2. 각각의 아이템을 설정할 수 있도록 제작하였습니다.
3. 글자와 이미지를 한 객체로 묶어서 자연스러운 회전을 보여줍니다.

핵심 코드

● 애니메이션을 이용한 랜덤 추첨



종이 추천 화면



종이 추천 화면 설명

- 1. 화면을 클릭하면 직접 추천 종이를 설정 할 수 있도록 하였습니다.
- 2. 한개씩 벗겨 결과를 확인 할 수 있습니다.

App 설정

build.gradle

```
dependencies {  
    implementation fileTree(include: ['*.jar'], dir: 'libs')  
    //noinspection GradleCompatible  
    implementation 'com.android.support:appcompat-v7:27.0.1'  
    implementation 'com.android.support:design:27.1.1'  
    implementation 'com.android.support.constraint:constraint-layout:1.1.3'  
    implementation 'com.android.support:support-v4:27.1.1'  
    implementation 'com.google.firebase:firebase-database:16.0.1'  
    implementation 'com.google.firebase:firebase-core:16.0.3'  
    testImplementation 'junit:junit:4.12'  
    androidTestImplementation 'com.android.support.test:runner:1.0.2'  
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'  
    implementation 'com.android.support:design:27.1.1'  
    implementation 'com.github.bumptech.glide:glide:3.7.0'  
    implementation group: 'com.kakao.sdk', name: 'kakaolink', version: project.KAKAO_SDK_VERSION  
    implementation group: project.KAKAO_SDK_GROUP, name: 'usermgmt', version: project.KAKAO_SDK_VERSION  
    compile group: 'com.kakao.sdk', name: 'kakaotalk', version: project.KAKAO_SDK_VERSION  
    // 원형 imageView 넣기  
    implementation 'de.hdodenhof:circleimageview:2.2.0'  
    implementation 'com.google.android.gms:play-services-ads:15.0.1'  
}  
  
apply plugin: 'com.google.gms.google-services'
```


소스 파일 정리

자바 파일 구성

▼

eventActivity_item

EventActivity

HLVAdapter

ItemClickListener

ListAdapter

▼

mDTOClass

RoomDTO

UserDTO

▼

nav_item_activity

InformationActivity

ListActivity

ManagerActivity

PrivateActivity

RequstActivity

CreateLotteryActivity

GlobalApplication

LotteryActivity

MainActivity

NumberActivity

PaperItemDialog

PaperLotActivity

RandomLotteryActivity

SplashActivity


WheelActivity


WheelItemDialog

WinUserDialog







파이어 베이스를 이용한 DB 관리

- 리얼 타임 DB를 이용하였습니다.

 **Firebase**

[Project Overview](#) 

개발

-  Authentication
-  **Database**
-  Storage
-  Hosting
-  Functions
-  ML Kit

품질
Crashlytics, Performance, Test Lab

애널리틱스
Dashboard, Events, Conversions, Au...

성장
Predictions, A/B Testing, Cloud Mes...

Spark
무료 \$0/월 [업그레이드](#)

MyTreat0914 ▾ Database

<https://mytreat0914.firebaseio.com/>

mytreat0914

CURRENT_EVENT

+

 -LMp7DCNkStMYXMsdfJi

-

 USER

+

x

+

 3-7c7N7t2ODU-Mn_zPTE8sDxyeXX5dLi2-uU

+

 3-7e69Pr2eje8sb1xfLH98_41ObU49Pq2lk

+

 3-7f59Pm0uDT_836wvrC98fw3O7c69vi0IE

+

 3-7f6Nro2eHZ7MDyyv3K-8z-zfXZ69nu3ufXWA

+

 3-bQ6d_s3O3V-cv4wflG9cbz xenb6d7u1-dv

+

 3-bR59Hk1OTX-8n9yP7H_8_4y-fV59Dg2elv

+

 3-fR4dLn0-LT_877zvzP98__x-vZ69zs1eV0

+

 3-fe5tLi1uDM_crzxvXH8MXp2-ne7tfnmw

+

 3-jc5NXl0PzO-sz5z_jN-tbk1uHR6Nio

+

 3-nQ6d3s2-jE9cL0zf7l_8n61uTW4dHo2KE

+

 3-nY7dvt2-PP_s35yPjM_8fr2evc7NXlZg

+

 3-ne6t3p3e3f88L3z_zE9sLywu7c7tnp00Bm

+

 3-rb69ng0OPP98f1w_TM_s3h0-HW5t_vkQ

+

 3-rd6Nvs3e7e8sPzw_bD9cH1xOja6N_v1uZh

+

 3-vd6dnh0-be8sP6zfvD9sPv3e_Y6NHhnw

+

 3-zU4tbl1efL_8r4y_N_dHj0ebW79-i

+

 3-zd7Njp2-vc8Mbzy_3F9sf020rY79_m1ll

통신을 위한 파이어베이스 파싱

- DB에서 데이터를 저장하고 불러오는 기능입니다.

```
public void dataSave() { //데이터 저장
    myRef = FirebaseDatabase.getInstance().getReference("USER");//RealtimeDB
    myRef.addChildEventListener(new ChildEventListener() {
        @Override
        public void onChildAdded(@NonNull DataSnapshot dataSnapshot, @Nullable String s) {
            UserDTO userDTO = dataSnapshot.getValue(UserDTO.class);
            try {
                if (userDTO.getUserUID().equals(userDTO.getUserUID())) {
                    if (userDTO.getManager() == null) {
                        Toast.makeText(CreateLotteryActivity.this, "관리자 권한이 없습니다", Toast.LENGTH_SHORT).show();
                    } else {
                        String title = editTitle.getText().toString();
                        String substance = editSub.getText().toString();
                        String password = editPass.getText().toString();

                        myRef = FirebaseDatabase.getInstance().getReference("CURRENT_EVENT");//RealtimeDB

                        if (substance.length() == 0) { //코멘트추가 안하면
                            roomDTO = new RoomDTO(title, password, "sub: " + "내용이 없습니다", userName, userUID, "eventCode: " + userName + time(), joinInUser: 0, time());
                            //타이틀, 비밀번호, 내용, 관리자명, 관리자UID, 이벤트코드, 참여인원, 시간
                            myRef.push().setValue(roomDTO);
                        } else { //코멘트 추가 하면
                            roomDTO = new RoomDTO(title, password, substance, userName, userUID, "eventCode: " + userName + time(), joinInUser: 0, time()); //순서대로 타이틀, 비밀번호, 내용, 관리자명, 관리자UID, 이벤트코드, 참여인원, 시간
                            myRef.push().setValue(roomDTO);
                        }
                    }

                    Toast.makeText(CreateLotteryActivity.this, "추첨을 시작 합니다.", Toast.LENGTH_SHORT).show();
                    finish();
                }
            }
        }
    });
}
```

```
void loadData () {
    DatabaseReference myRef = FirebaseDatabase.getInstance().getReference("CURRENT_EVENT");//RealtimeDB
    myRef.addChildEventListener(new ChildEventListener() {
        @Override
        public void onChildAdded(@NonNull DataSnapshot dataSnapshot, @Nullable String s) {
            RoomDTO roomDTO = dataSnapshot.getValue(RoomDTO.class);
            try {
                if (roomDTO.getEndTime() == null) {
                    String title = roomDTO.getTitle();
                    String sub = roomDTO.getSub();
                    String pw = roomDTO.getPw();
                    String dataKey = dataSnapshot.getKey();
                    int joinInUser = roomDTO.getJoinInUser();
                    String eventCode = roomDTO.getEventCode();

                    alTitle.add(title);
                    alSub.add(sub);
                    alPW.add(pw);
                    alDataKey.add(dataKey);
                    alJoinInUser.add(joinInUser);
                    alEventCode.add(eventCode);
                }
            } catch (NullPointerException e) {
                e.printStackTrace();
            }
        }
    });
}

@Override
public void onChildChanged(@NonNull DataSnapshot dataSnapshot, @Nullable String s) {
```

주요 사용 코드 정리

1. RecyclerView (horizontal) 사용
2. 이미지 및 각종 UI 구성은 listView로 사용
3. 이미지 라운드 처리 (CircleImageView) 사용
4. 파이어 베이스를 이용한 DB 관리
5. onClick/onLongClick 을 이용한 정보 삭제 및 화면 전환 기능 사용
6. searchView를 이용한 검색 기능 사용
7. 애드몹(AdMob) 광고 연동

어플리케이션 정보

추가 정보		
업데이트 날짜	크기	설치 수
2018년 9월 27일	5.2M	10+
현재 버전	필요한 Android 버전	콘텐츠 등급
1.7	4.2 이상	만 3세 이상 자세히 알아보기
권한	신고	제공
세부정보 보기	부적절한 앱으로 신고	dodam
개발자		
2018dodam@gmail.com		
개인정보처리방침		

개발 환경 : 안드로이드 스튜디오

App 용량 : 5.2MB

App 마켓 주소 :

<https://play.google.com/store/apps/details?id=com.ezen.haneor.mytreat>

필요한 Android 버전 : 4.2 이상