

FadeOut

실시간 시한성 메시지 교환 채팅 App



Developed by 명평윤(13)

Contents

01. Fade Out

02. App UI

03. App 기능 설명

04. 애로 사항

01

FadeOut



‘실시간 시한성 메시지 교환 채팅 App’



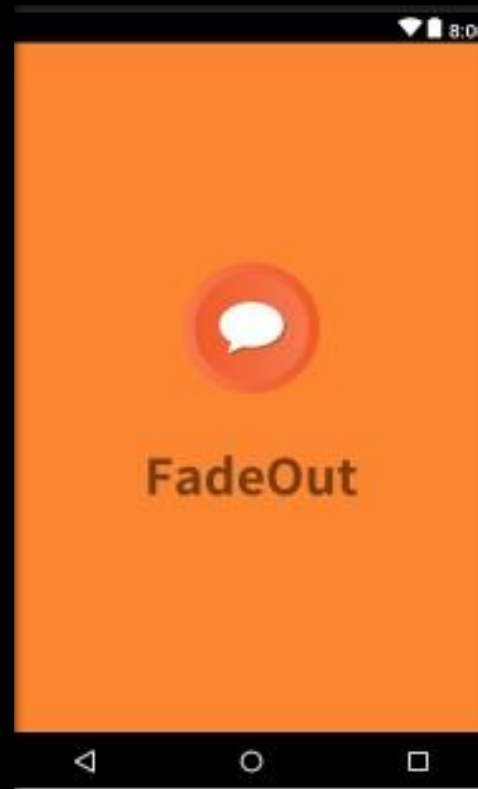
Firebase



Android
Studio

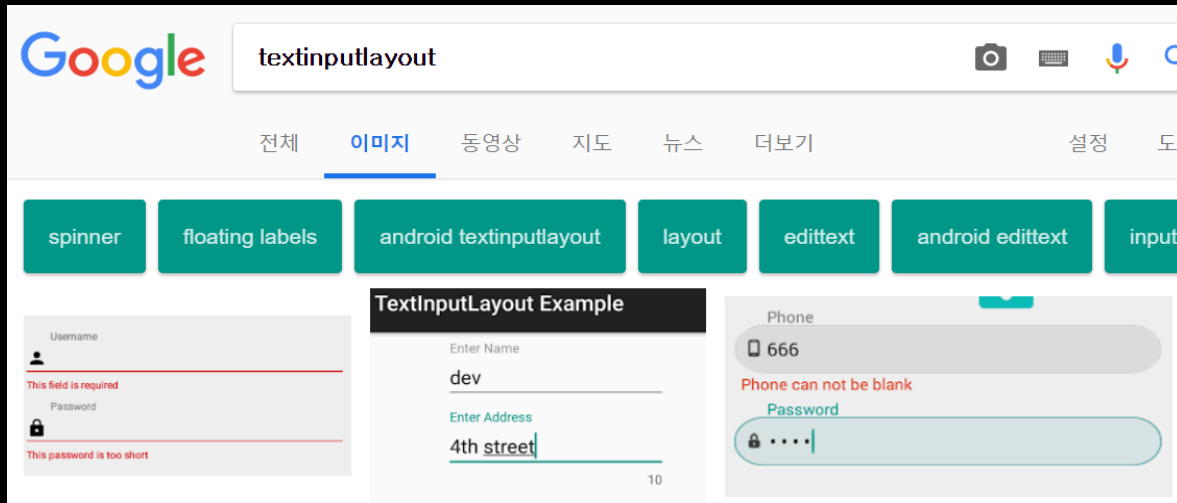
02 App UI

```
//상태바 안보이게 전체화면으로(회원가입 및 로그인 화면에서)  
getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,  
    WindowManager.LayoutParams.FLAG_FULLSCREEN);
```



메인 화면

02 App UI



```
<android.support.design.widget.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
```



로그인 화면

02 App UI

```
<!--프로필 작성 -->
<ImageView
    android:src="@drawable/add_to_photos_black_24"
    android:id="@+id/signupActivity_imageView_profile"
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:layout_margin="50dp"/>

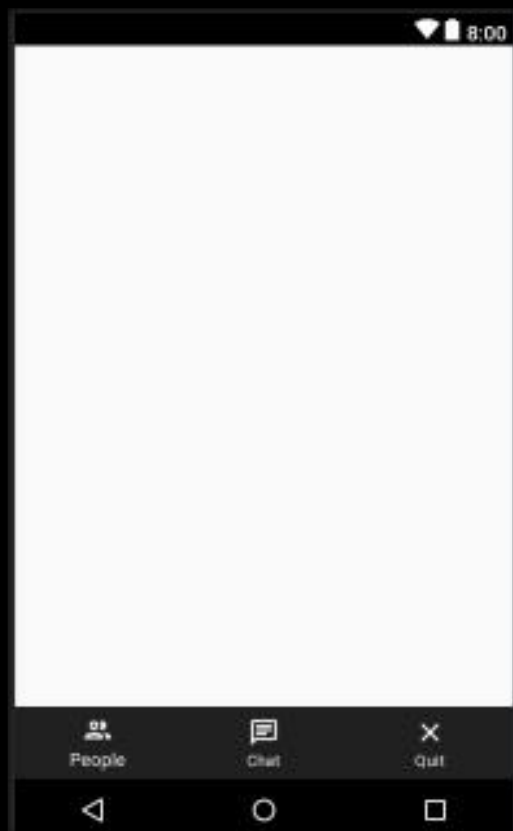
<android.support.design.widget.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
```



회원가입 화면

02 App UI

기본 메뉴 화면



```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">

    <item
        android:id="@+id/action_people"
        android:icon="@drawable/ic_people_black_24"
        android:title="People"
        android:fontFamily="@font/notosans_bold"/>

    <item
        android:id="@+id/action_chat"
        android:icon="@drawable/ic_chat_black_24"
        android:title="Chat"
        android:fontFamily="@font/notosans_bold"/>

    <item
        android:id="@+id/action_quit"
        android:icon="@drawable/ic_quit_black_24"
        android:title="Quit"
        android:fontFamily="@font/notosans_bold"/>

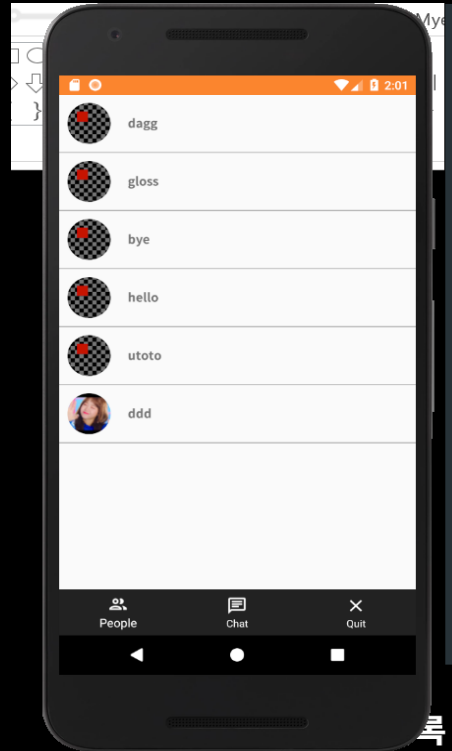
</menu>
```

```
<android.support.design.widget.BottomNavigationView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/mainActivity_bottomNavigationView"
    android:layout_alignParentBottom="true"
    app:itemBackground="@color/common_google_signin_btn_text_light_pressed"
    app:itemTextColor="@android:color/white"
    app:itemIconTint="@android:color/white"
    app:menu="@menu/bottom_navigation_main"/>
```

02 App UI

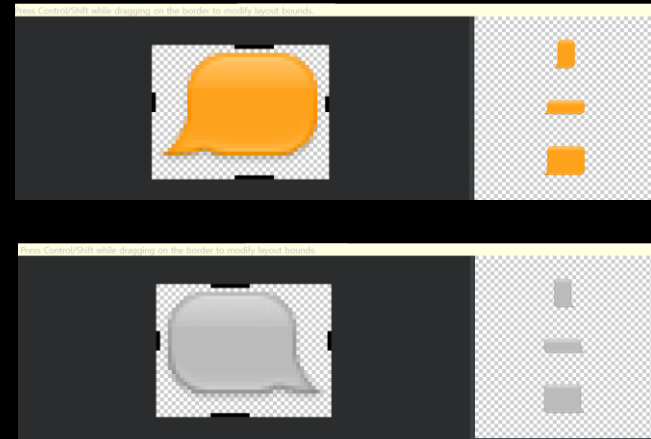


채팅 메시지
출력 화면



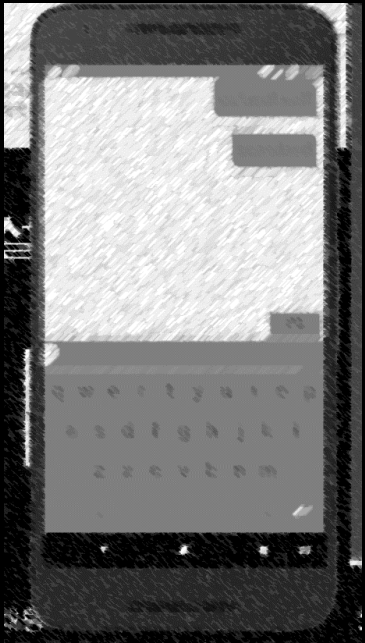
친구 목록
화면

```
<android.support.v7.widget.RecyclerView  
    android:id="@+id/messageActivity_recyclerView"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_above="@+id/messageActivity_button"  
    android:layout_alignParentEnd="true"  
    android:layout_alignParentRight="true"  
    android:layout_alignParentTop="true"  
    android:layout_alignParentLeft="true"  
    android:layout_alignParentStart="true">  
  
</android.support.v7.widget.RecyclerView>
```



Resizable
Bitmaps
(9-Patch files)

03 App 기능 설명



사라지는 메시지
채팅 기능

`private final int OUT_TIME = 5000; //5초 후에 메시지 사라짐`

```
// 2초간 멈추게 하고싶다면
Handler handler = new Handler();
handler.postDelayed(() -> {
    FirebaseDatabase.getInstance().getReference()
        .child("chatrooms").child(chatRoomUid).child("comments").setValue(null); //방을 나가면 DB의 채팅 내용 사라짐
}, OUT_TIME); // 1000은 1초를 의미합니다.
```



03 App 기능 설명



기본적인 채팅 기능

```
//내가 보낸 메시지 출력
if (comments.get(position).uid.equals(uid)) //현재 계정과 DB의 메시지를 상대방에게 보낸 계정이 동일하면
{
    messageViewHolder.textView_message.setText(comments.get(position).message); //메시지 내용
    messageViewHolder.textView_message.setBackgroundResource(R.drawable.rightbubble); //메시지 버블
    messageViewHolder.linearlayout_destination.setVisibility(View.INVISIBLE); //내가 특할때는 상대방 메시지 버블 감추기
    messageViewHolder.linearlayout_main.setGravity(Gravity.RIGHT); //채팅 오른쪽 정렬(default는 왼쪽)
```



```
//상대방이 보낸 메시지
else {

    Glide.with(holder.itemView.getContext())
        .load(destinationUserModel.profileImageUrl)
        .apply(new RequestOptions().circleCrop())
        .into(messageViewHolder.imageView_profile); //채팅창에 상대방의 프로필 사진 가져 오기
    messageViewHolder.textView_name.setText(destinationUserModel.userName); //상대 계정 내용
    messageViewHolder.linearlayout_destination.setVisibility(View.VISIBLE); //상대방 채팅쪽 보이도록
    messageViewHolder.linearlayout_main.setGravity(Gravity.LEFT); //채팅 왼쪽 정렬(default지만 확실히 하기 위해 gravity 지정)
    messageViewHolder.textView_message.setBackgroundResource(R.drawable.leftbubble); //메시지 버블
    messageViewHolder.textView_message.setText(comments.get(position).message); //메시지 내용
    messageViewHolder.textView_message.setTextSize(25); //글씨 크기
```

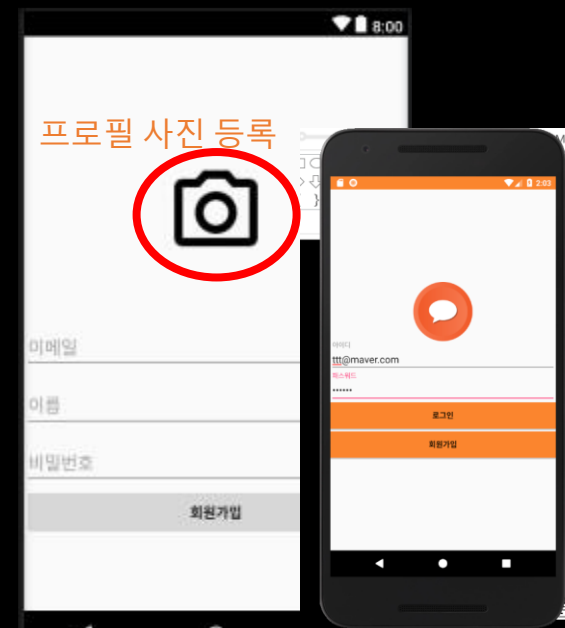


03 App 기능 설명

```
FirebaseAuth.getInstance()
    .createUserWithEmailAndPassword(email.getText().toString(), password.getText().toString())
    .addOnCompleteListener(activity: SignUpActivity.this,
        //회원가입이 완료되면
        (task) -> {
            final String uid = task.getResult().getUser().getUid(); //uid값(계정의 구분 값)
            FirebaseStorage.getInstance().getReference().child("userImages") //프로필 이미지
                .child(uid).putFile(imageUri)
                .addOnCompleteListener((task) -> {
                    //VisibleForTests/ //에러 없애기
                    String imageUrl = task.getResult().getDownloadUrl().toString();

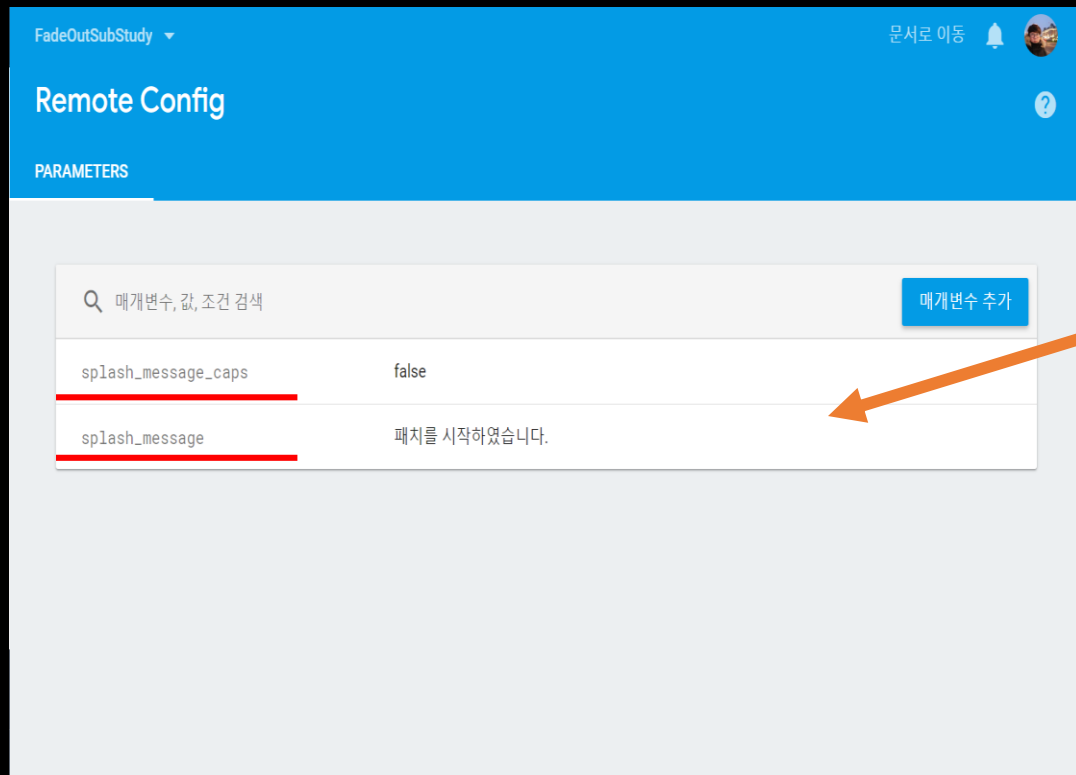
                    UserModel userModel = new UserModel();
                    userModel.userName = name.getText().toString();
                    userModel.profileImageUrl = imageUrl;
                    userModel.uid = FirebaseAuth.getInstance().getCurrentUser().getUid();

                    //데이터베이스에 Username 집어 넣기
                    FirebaseDatabase.getInstance().getReference().child("users")
                        .child(uid).setValue(userModel);
                    SignUpActivity.this.finish();
                });
        });
```



회원 가입
및 로그인 기능

03 App 기능 설명



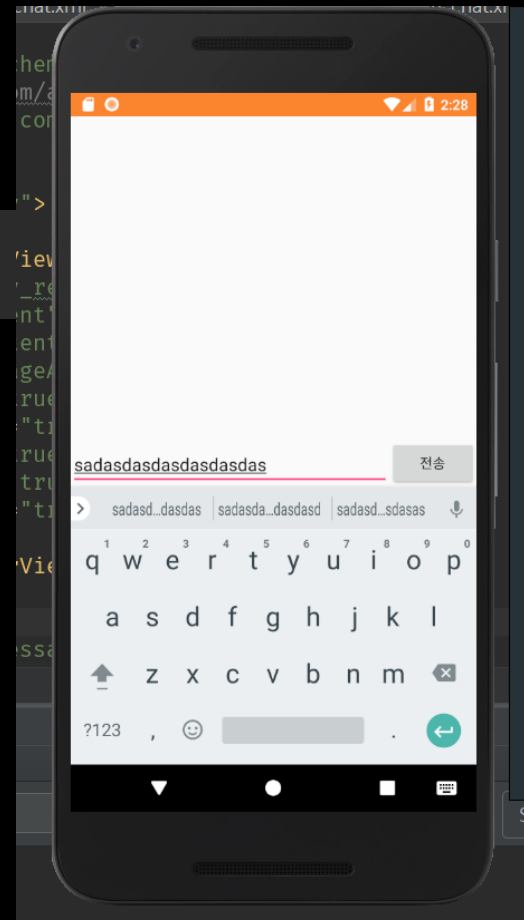
```
<?xml version="1.0" encoding="utf-8"?>
<!-- START xml_defaults -->
<defaultsMap> <!-- Firebase Console의 원격을 이용해 변경할 파라미터 관리-->
  <entry>
    <key>splash_background</key> <!--splash 배경색 지정-->
    <value>#FFFB8530</value>
  </entry>
  <entry>
    <key>splash_message_caps</key> <!-- 메시지 띄우기-->
    <value>>false</value>
  </entry>
  <entry>
    <key>splash_message</key> <!--메시지 내용 관리-->
    <value>서버 점검 중입니다.</value>
  </entry>
</defaultsMap>
<!-- END xml_defaults -->
```

원격 제어 기능

03 App 기능 설명

```
private final int EDITTEXT_LENGTH = 20; //입력 길이 제한(실시간으로 사라지는 채팅들의 가독성을 위해)
```

```
editText.setFilters(new InputFilter[]{  
    new InputFilter.LengthFilter(EDITTEXT_LENGTH)//입력 길이 수 제한  
});
```



채팅 입력 글자 수 제한

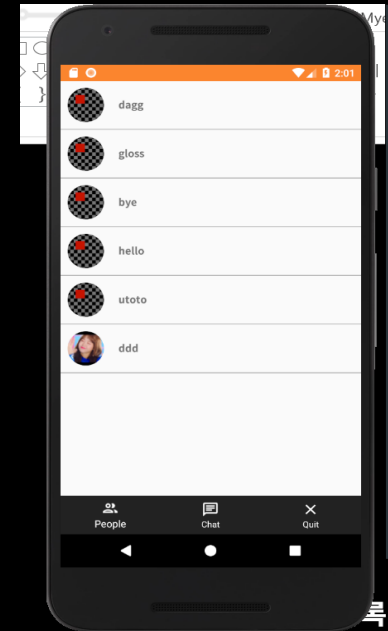
03 App 기능 설명

```
//서버에서 넘어온 데이터
@Override
public void onDataChange(DataSnapshot dataSnapshot) {
    //데이터 쌓기
    userModels.clear(); //누적되는 데이터 삭제

    for(DataSnapshot snapshot : dataSnapshot.getChildren())
    {
        //본인의 계정을 친구목록에서 지우기
        UserModel userModel = snapshot.getValue(UserModel.class);

        if(userModel.uid.equals(myUid))
            continue;
        userModels.add(userModel);
    }
    //새로고침(데이터 쌓은 걸로 친구 목록 띄우기)
    notifyDataSetChanged();
}
```

```
//친구 프로필 누르면 채팅창 열기
holder.itemView.setOnClickListener((view) -> {
    Intent intent = new Intent(view.getContext(), MessageActivity.class);
    intent.putExtra( name: "destinationUid", userModels.get(position).uid);
    //화면 전환 애니메이션 효과 삽입
    ActivityOptions activityOptions = ActivityOptions.makeCustomAnimation(view.getContext(),
        R.anim.fromright, R.anim.toleft);
    startActivity(intent, activityOptions.toBundle());
});
```

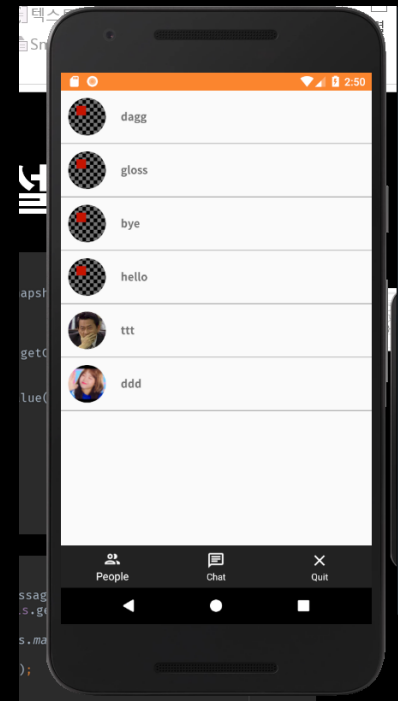


친구 목록화 기능

03 App 기능 설명

```
<set xmlns:android="http://schemas.android.com/apk/res/android">
  <!-- 왼쪽에 있는 액티비티 화면이 오른쪽 현재 화면으로 밀려오기 -->
  <translate
    android:duration="300"
    android:fromXDelta="-100%" android:toXDelta="0"
    android:interpolator="@android:anim/accelerate_decelerate_interpolator"/>
</set>
```

```
<set xmlns:android="http://schemas.android.com/apk/res/android">
  <!-- 오른쪽의 액티비티 화면이 현재의 왼쪽 화면으로 밀려 들어옴 -->
  <translate
    android:duration="300"
    android:fromXDelta="0%" android:toXDelta="100%"
    android:interpolator="@android:anim/accelerate_decelerate_interpolator"/>
</set>
```



화면 전환

03 App 기능 설명

```
bottomNavigationView.setOnNavigationItemSelectedListener((item) → {  
    switch (item.getItemId()){  
        case R.id.action_people: //친구 목록 보기  
            getFragmentManager().beginTransaction().replace(R.id.ma  
            return true;  
        case R.id.action_chat: //채팅 목록 보기  
            getFragmentManager().beginTransaction().replace(R.id.ma  
            return true;  
        case R.id.action_quit: //앱 종료  
            moveTaskToBack( nonRoot: true);  
            finish();  
    }  
})
```



앱 종료(직관화)

04 애로사항

```
dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:animated-vector-drawable:27.1.0'
    implementation 'com.android.support:support-v4:27.1.0'
    implementation 'com.android.support:appcompat-v7:27.1.0'
    implementation 'com.android.support:support-media-compat:27.1.0'
    implementation 'com.android.support.constraint:constraint-layout:1.1.0'
    implementation 'com.android.support:recyclerview-v7:27.1.0'
    implementation 'com.android.support:design:27.1.0'
    implementation 'com.github.bumptech.glide:glide:4.3.0'

    implementation 'com.firebaseui:firebase-ui-auth:3.3.0'
    ⚠ implementation 'com.firebaseui:firebase-ui-database:3.1.0'

    // Auth
    implementation 'com.google.firebase:firebase-auth:12.0.1'
    implementation 'com.google.android.gms:play-services-auth:12.0.1'
    implementation 'com.google.android.gms:play-services-appinvite:12.0.1'

    // firebase
    implementation 'com.google.firebase:firebase-database:12.0.1'
    implementation 'com.google.firebase:firebase-messaging:12.0.1'
    implementation 'com.google.firebase:firebase-config:12.0.1'
```

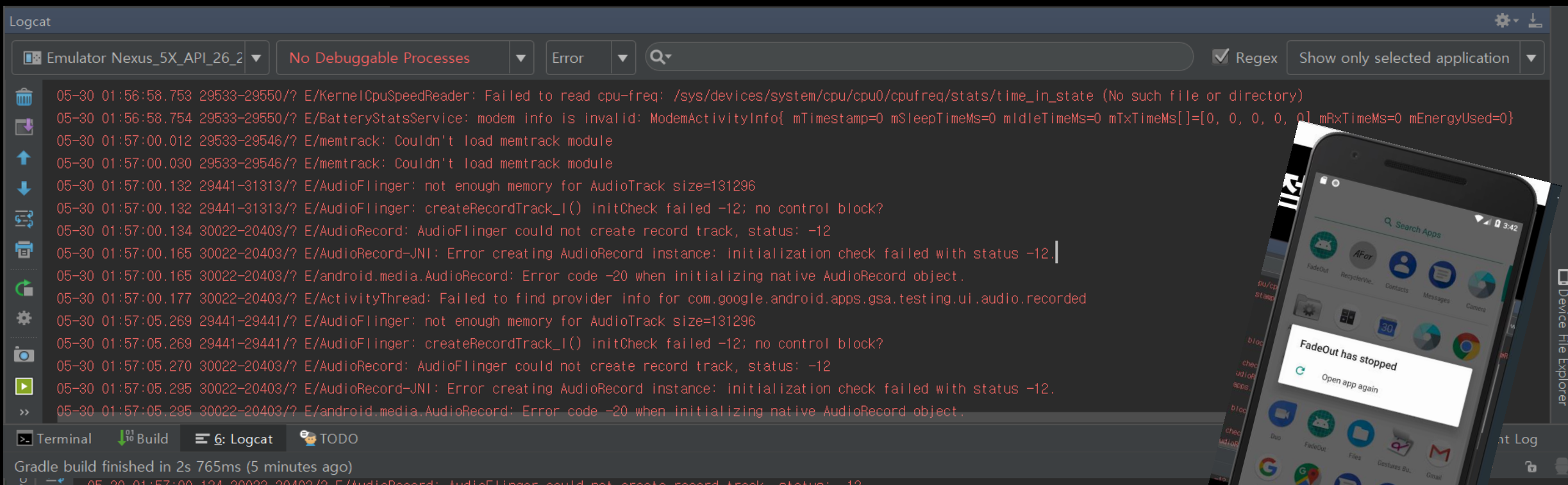
Dependency 호환성 문제

04 애로사항



코드 내 permission 및
Firebase 권한(DB, Storage)

04 애로사항



원인을 알 수 없는 Logcat error와 그리고 강제 종료되는 현상

FadeOut

실시간 시한성 메시지 교환 채팅 App



감사합니다.
이어서 시연 진행하겠습니다.