

VideoApp

介绍：一个可以看视频,收藏,点赞的简洁app。

内容目录：

VideoApp

文档内容

1:项目解决的现实问题分析；

(1.项目简介

(2.项目背景 (现实问题)

(3.项目设计

(4.项目价值 (解决的社会问题)

2:项目用到了哪些Android技术

3:总体设计

(1.功能结构图

(2.数据库设计

(3.类结构图

4:各个功能模块详细设计与实现、测试结果。

主模块设计

首页模块

 浏览视频

 视频信息

 点赞、收藏功能

 下载详情功能

 新增视频广播

收藏模块

个人模块

 登录/注册功能

 登录/注册实现

 个人信息

 投稿视频

ps:

操作逻辑：

文档内容

1:项目解决的现实问题分析；

(1.项目简介

该简洁无广告的视频应用程序旨在解决用户在观看视频时频繁遇到的广告打扰问题，提供一个干净、流畅的观看体验。

(2.项目背景 (现实问题)

在传统视频应用程序中，用户经常遇到繁琐的广告干扰，这些广告不仅中断了观看体验，还影响了用户的专注度和满意度。用户常常希望能够有一个没有广告干扰的视频平台，以便更好地享受视频内容。

(3.项目设计

1. 无广告：该应用程序摒除了任何形式的广告，提供一个完全无广告的环境。用户在观看视频时不会被打断，可以持续地欣赏视频内容，享受流畅的观看体验。无广告设计使用户能够专注于视频内容，提供了一个纯净、愉悦的观看环境。用户不必担心弹出广告、前置广告或视频中的插播广告，可以全情投入到视频欣赏中。
2. 精简用户界面：应用程序采用简洁明了的用户界面设计，旨在提供清爽的使用体验。界面布局简单，避免过多元素和复杂的视觉效果，使用户能够专注于视频内容。提供清晰的导航结构和标签，方便用户浏览和切换不同的功能模块，如视频列表、收藏夹和用户界面。
3. 点赞收藏功能：应用程序提供点赞和收藏功能，让用户对喜欢的视频进行点赞和收藏操作。点赞和收藏反映了用户的喜好和兴趣，同时也用于对视频进行排序和推荐。根据视频的点赞和收藏数量，应用程序会按一定比例折算进行排序，让用户优先浏览和观看受欢迎且优质的视频内容。这样，用户可以轻松发现和享受高质量的视频资源。
4. 上传视频功能：应用程序还提供上传视频的功能，允许用户将自己创作的视频上传到平台进行分享和展示。用户可以通过界面选择视频文件并填写相关信息（如标题、描述等），然后将视频上传到服务器进行存储和处理。这样，用户可以方便地展示自己的创作，与其他用户分享优质内容。

(4.项目价值 (解决的社会问题)

1. 提升用户体验：传统视频应用程序常常被冗长的广告打扰，影响用户的观看体验。该应用程序摒除了广告干扰，提供了一个无广告的环境，让用户能够专注、流畅地观看视频，提升了用户的观看体验。
2. 节约用户时间：广告的频繁插播导致用户观看视频时浪费了大量的时间。该应用程序去除了广告，用户不再被打断，能够更有效地利用时间观看视频内容，提高了用户的效率。
3. 推广优质内容：通过点赞和收藏功能，该应用程序能够根据用户的喜好和兴趣推荐和排序视频。这有助于发现和推广优质的视频内容，给那些创作者提供更多的曝光机会，激励他们创作更多优质的视频作品。
4. 提供创作平台：上传视频功能为用户提供了一个展示自己创作的平台。用户可以将自己的视频作品上传到应用程序上与其他用户分享，促进创作交流和创作者之间的互动。

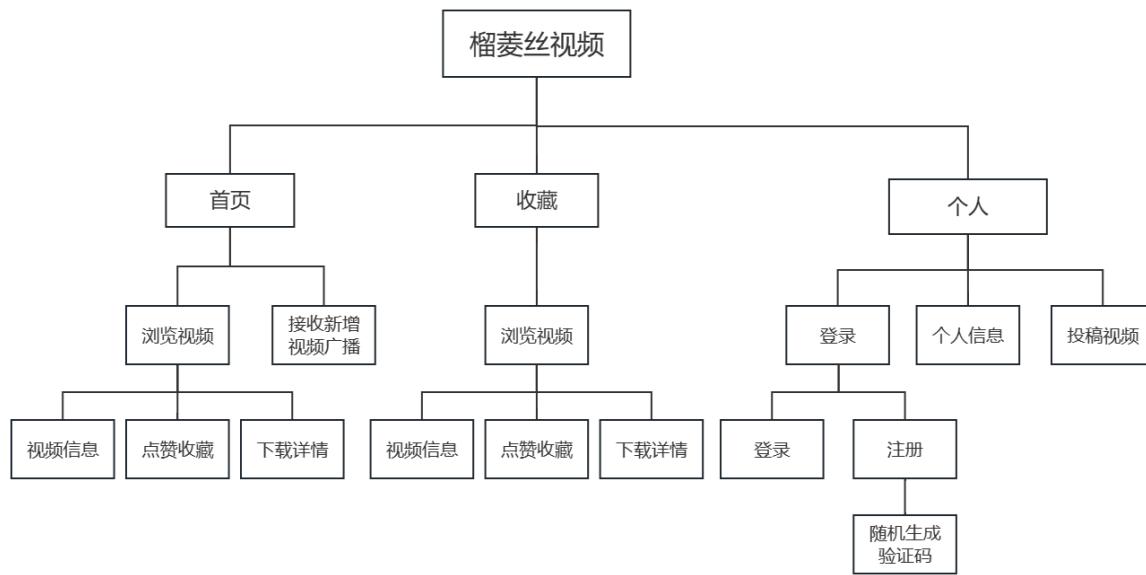
2:项目用到了哪些Android技术

序号	知识点	内容
1	UI布局及控件	常用控件的使用、基本的页面布局方法、Toast等消息提示等
2	Activity和Intent	Activity基本用法（界面元素获取、界面交互事件）、Intent的基本使用（显示Intent或隐式Intent、Intent启动其他活动和应用程序，界面传值）、Adapter（适配器）的基本使用将数据和视图进行绑定，将数据在用户界面中展示出来等
3	Fragment碎片	Fragment的创建、静态或动态加载、系统底部导航栏设计、Fragment与Activit的数据交互，等

序号	知识点	内容
4	数据持久化	Android程序数据的本地化存储：sharedPreferences、SQLite： sharePreferences存储已登录用户id（相当于一个token），存储已读视频，并变灰色。SQLite存储了Video, User, Star三张表
5	广播	接受系统广播、自定义广播，发送广播：登录/注册成功广播，综合通用广播
6	服务	后台完成的任务：下载视频详情到手机内存卡中

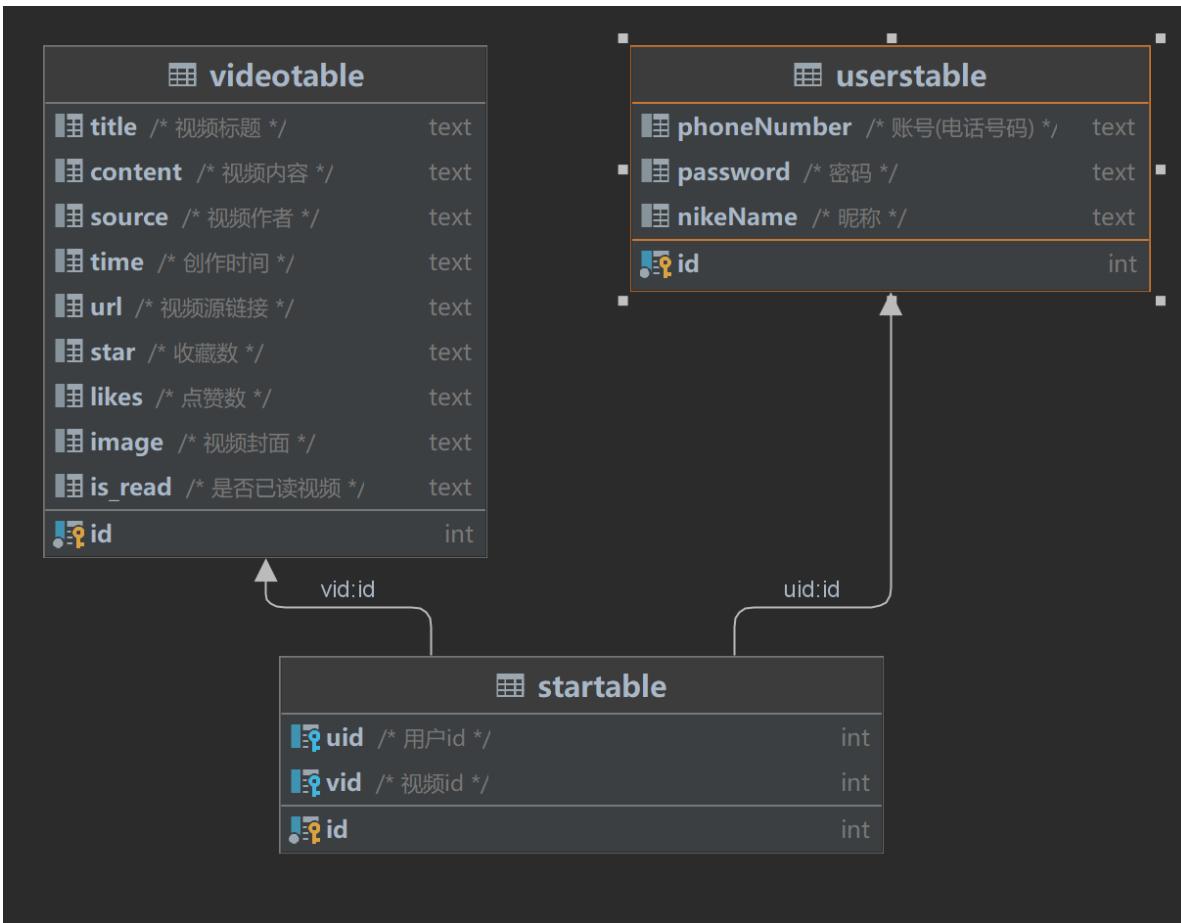
3:总体设计

(1.功能结构图)

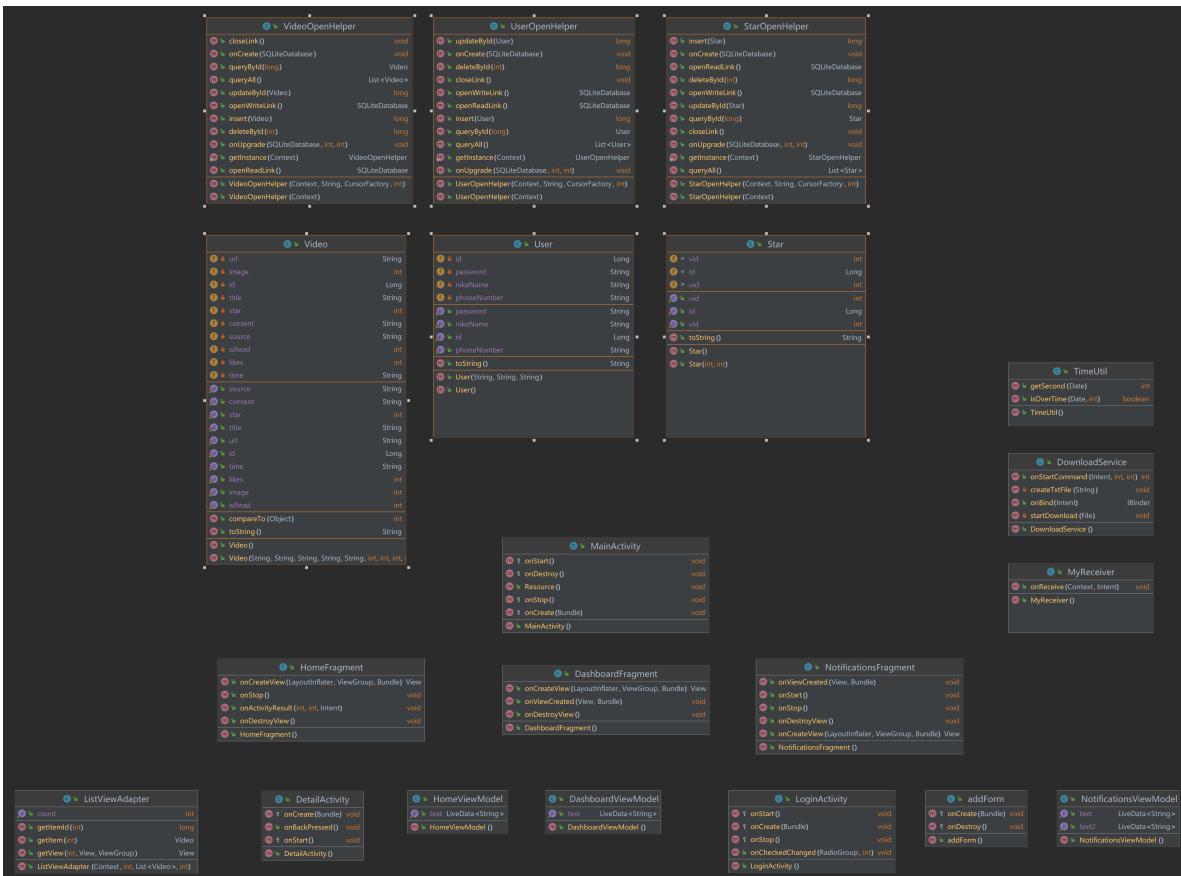


(2.数据库设计)

其中startTable为收藏表，用于存储用户id和他收藏的视频id，制作收藏夹功能；



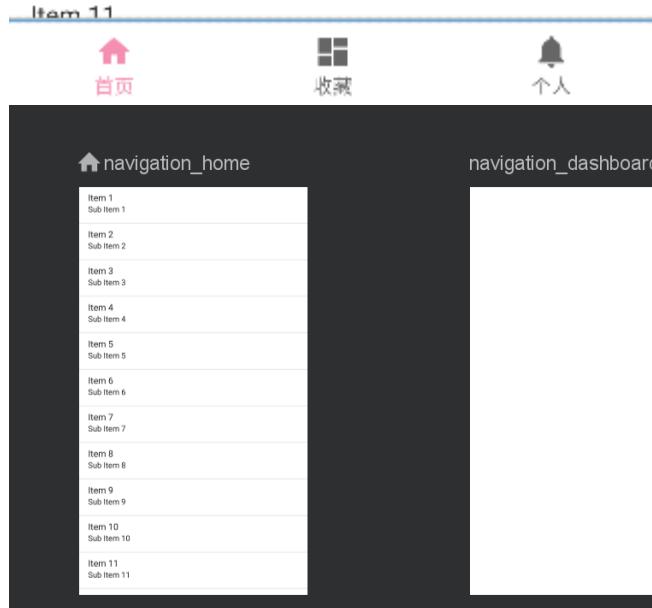
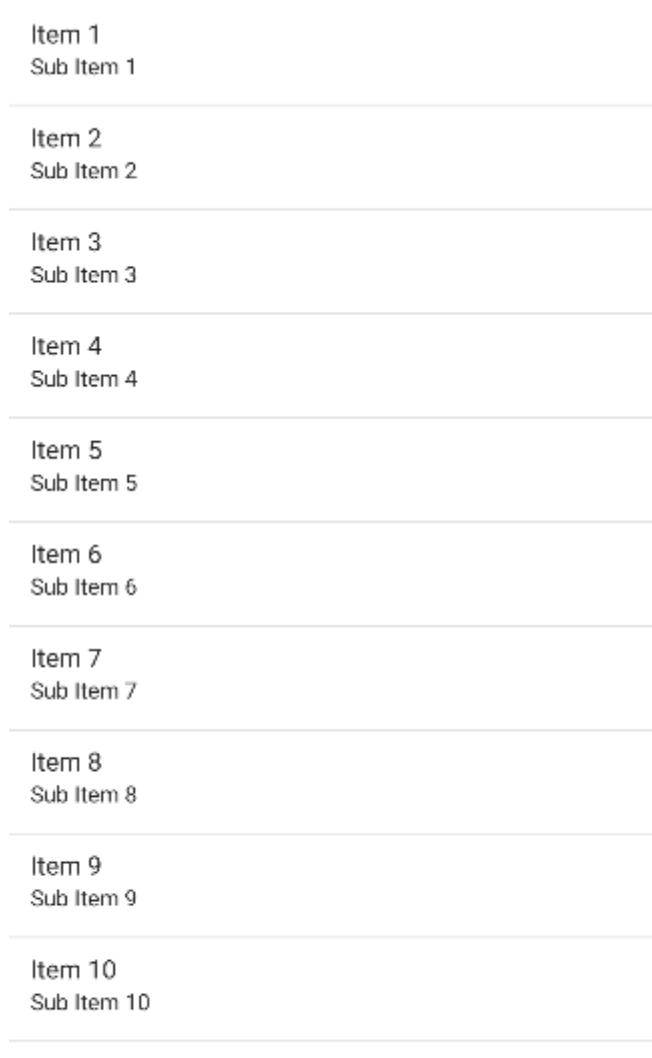
(3.类结构图



4:各个功能模块详细设计与实现、测试结果。

主模块设计

底部导航栏组件 和一个fragment组件



实现底部导航栏切换fragment功能



功能实现主要代码

```

BottomNavigationView navView = findViewById(R.id.nav_view);
AppBarConfiguration appBarConfiguration = new
AppBarConfiguration.Builder(R.id.navigation_home, R.id.navigation_dashboard,
R.id.navigation_notifications).build();
NavController navController = Navigation.findNavController(this,
R.id.nav_host_fragment_activity_main);

//往HomeFragment传值
navController.navigate(R.id.navigation_home, bundle);

NavigationUI.setupActionBarWithNavController(this, navController,
appBarConfiguration);
NavigationUI.setupWithNavController(binding.navview, navController);

```

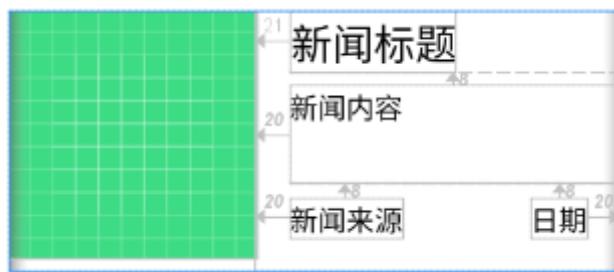
首页模块

浏览视频

在fragment中，使用ListView做容器

```
<!-- 装载详情的容器 -->
<ListView
    android:id="@+id/list_container"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"/>
```

存放多个LinearLayout视图



用ListViewAdapter（适配器）的基本使用将数据和视图进行绑定，并设置点击事件（将新闻数据从fragment包装发送到activity中）

```
//把信息加入到Adapter
ListViewAdapter adapter = new ListViewAdapter(getActivity(),
    R.layout.news_item, listItem, Configuration.ORIENTATION_LANDSCAPE);
listView.setAdapter(adapter);
listView.setOnItemClickListener((parent, view, position, id) -> {
    Intent intent = new Intent();
    intent.setClass(getActivity(), DetailActivity.class);
    intent.putExtra("news", listItem.get(position));
    startActivityForResult(intent, 1);
});
```

最终呈现效果：



视频信息

设计：标题+作者+日期+VideoView容器+点赞+收藏+ScrollView滑动窗口存放详情(里面存放文字说明和一个洗澡链接)



视频播放设置：(网上找的视频解码方法太难做了，要做感觉会做很久很久，所以我把视频存我的服务器的minio的桶里，播放视频的时候直接从服务器的url里获取视频就行，缺点就是加载视频比较慢，比较费我服务器的内存空间)

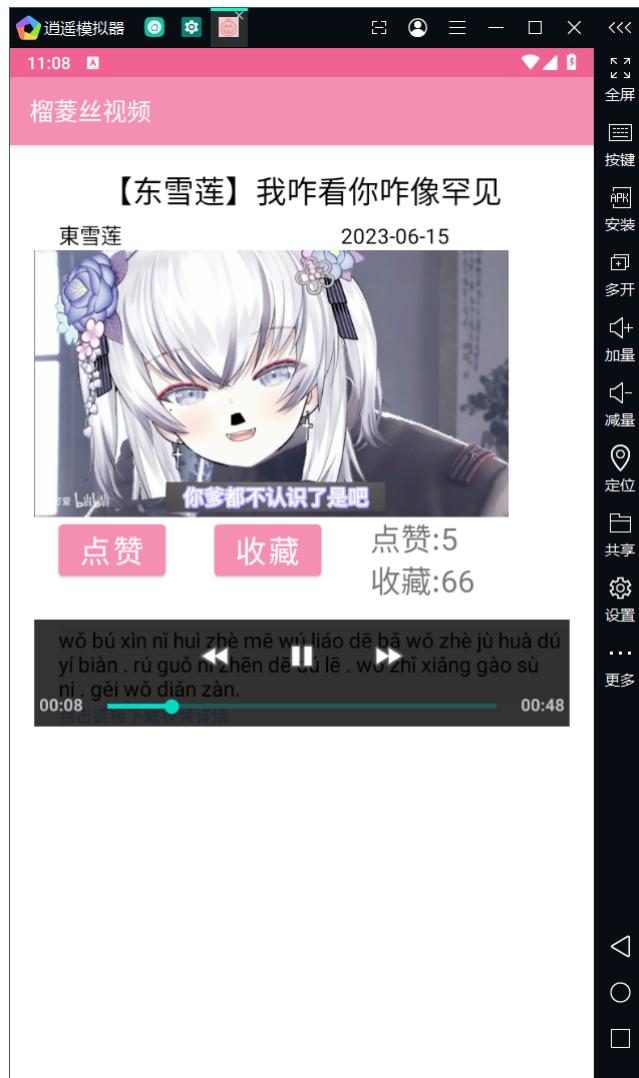
```

//数据来源
videoView videoView = findViewById(R.id.vv_detail_sp);
video = (Video) intent.getSerializableExtra("news");
//视频设置
MediaController mediaController = new MediaController(this);
mediaController.setAnchorView(videoView);
Uri uri = Uri.parse(video.getUrl());

videoView.setMediaController(mediaController);
videoView.setVideoURI(uri);
videoView.start();

```

最终呈现效果：



点赞、收藏功能

设置按钮监听，动态更新点赞数和收藏数，收藏还要在StarTable中添加一个数据存入用户id和视频id

```

likesButton.setOnClickListener(v -> {
    //点赞数+1
    int like = video.getLikes();
    video.setLikes(like + 1);
    videoHelper.updateById(video);
    likes.setText("点赞：" + video.getLikes());
});

```

```
starButton.setOnClickListener(v -> {
    //收藏数+1
    int stars = video.getStar();
    video.setStar(stars + 1);
    videoHelper.updateById(video);
    star.setText("收藏:" + video.getStar());
    //收藏关联关系
    Star star1 = new Star();
    star1.setUid(Integer.parseInt(uid));
    star1.setVid(Math.toIntExact(video.getId()));
    starHelper.insert(star1);

});
```

最终呈现效果：



下载详情功能

给文本[点击链接下载视频详情](#)设置点击监听，把要下载的数据封装传入service中

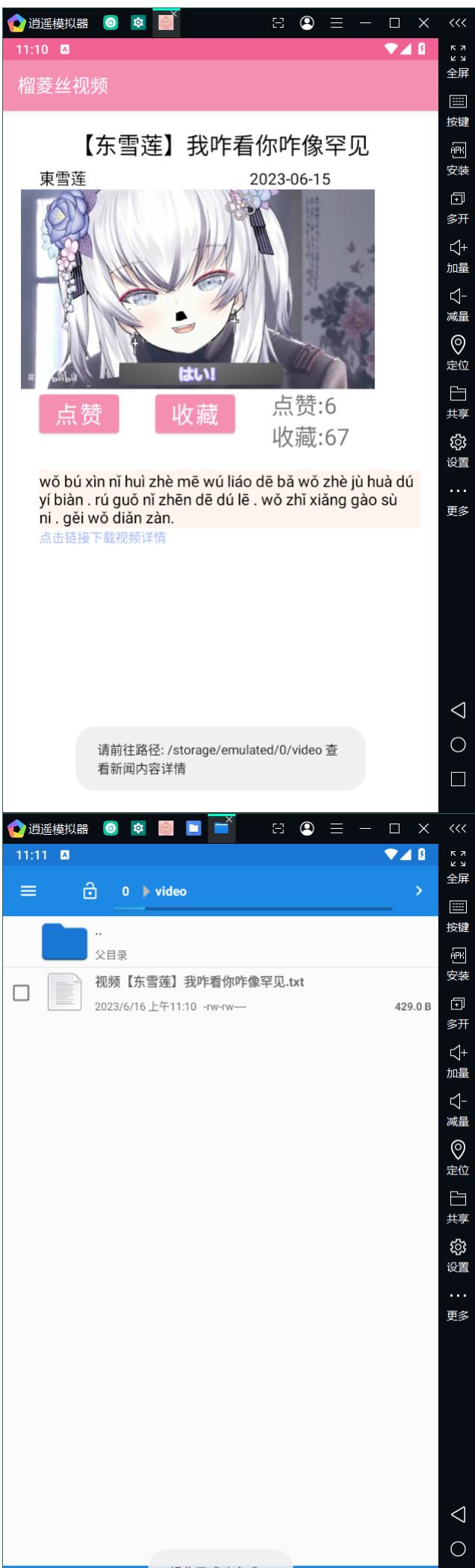
```
download = findViewById(R.id.btn_detail_download);
download.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String text = video.toString(); // 要下载的文本
        Intent serviceIntent = new Intent(getApplicationContext(),
DownloadService.class);
        serviceIntent.putExtra("text", text);
        serviceIntent.putExtra("title", "视频" + video.getTitle() +
".txt");
        startService(serviceIntent);
    }
});
```

在service中启动下载并发送广播告知下载保存的路径

```
@Override
public int onStartCommand(Intent intent, int flags, int startId) {
    // 在这里执行下载操作
    // 这可以是使用线程、异步任务等方式来下载内容title
    text = intent.getStringExtra("text");
    title = intent.getStringExtra("title");
    //下载文件
    String path = createTxtFile(text);
    //发送自定义广播
    IntentFilter intentFilter = new IntentFilter("broadcast");
    myReceiver = new MyReceiver();
    Intent intentReceiver = new Intent("broadcast");
    //直接发送广播
    intentReceiver.putExtra("broadcast", "请前往路径：" + path + " 查看新闻内容详情");
    registerReceiver(myReceiver, intentFilter);
    sendBroadcast(intentReceiver);

    return START_STICKY; // 指示系统在服务被终止后重新启动服务
}
```

最终呈现效果：





新增视频广播

收藏模块

加载Star表

浏览视频沿用首页模块的功能

最终呈现效果：

个人模块

分登录，注册，个人信息和投稿视频四个功能

给登录按钮设置了点击和长按事件（长按事件带一个参数在activity和activity之间传值）

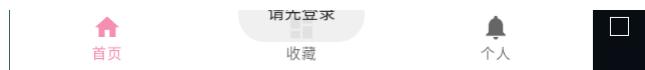
```
//为图片按钮设置点击事件
btnImage.setOnClickListener(view1 -> {
    Intent intent = new Intent();
    intent.setClass(getActivity(), LoginActivity.class);
    startActivity(intent);
});

//为图片按钮设置长按事件
btnImage.setOnLongClickListener(new View.OnLongClickListener() {
    @Override
    public boolean onLongClick(View view) {
        Intent intent = new Intent();
        intent.putExtra("long", R.id.rb_register);
        intent.setClass(getActivity(), LoginActivity.class);
        startActivity(intent);
        return false;
    }
});
```

点击事件进入登录，长按则在进入登录界面时自动切换至注册模块，在登录/注册模块中也可以点击RadioButton切换登录和注册功能

未登录的界面（这时候投稿视频会提示请先登录）：





登录/注册功能

遵从简洁风格，登录部分采用的User实体类只有手机号（账号），密码，昵称三个属性

界面设计（登录下面还藏了几个`android:visibility="gone"`的隐藏组件，在注册的时候显示出来）

最终呈现效果：



登录/注册实现

- 1.判断RadioButton按钮为：密码登录/账号注册
- 2.判断手机号格式是否正确（是否为11位手机号）
- 3.查数据库UserTable表找是否存在用户（登录则判断账号密码/注册则判断账号是否已经存在）
- 4.登录/注册成功后（登录则把用户id发送到MainActivity，在MainActivity中获取uid并存入sharedPreference中
注册则多一个把用户存入用户表的操作，再把用户id发送到MainActivity....同上）

登录按钮监听代码实现

```
btn_login.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (checkedId == R.id.rb_password) {
            log_phone = et_login_list1.getText().toString();
            log_password = et_login_list2.getText().toString();
            if (log_phone.length() < 11) {
                Toast.makeText(getApplicationContext(), "请输入正确的手机号码",
                Toast.LENGTH_SHORT).show();
            } else {
                boolean flag = true;
                boolean unLogin = true;
                for (User user : listitem) {
                    Log.d(TAG, "onClick: " + user);
                    if (Objects.equals(user.getPhoneNumber(),
                            log_phone)) {
                        flag = false;
                    }
                    if (Objects.equals(user.getPassword(),
                            log_password)) {
                        unLogin = false;
                    }
                    //登录成功通报表扬
                    Intent intent = new Intent("loginSuccess");
                    intent.putExtra("nickName", user.getNikeName());
                    Log.d(TAG, "onClick: 发送广播");
                    sendBroadcast(intent);

                    Log.d(TAG, "onClick: 跳转");
                    startActivity(new Intent(LoginActivity.this,
                            MainActivity.class).putExtra("uid", user.getId()));
                }
            }
            if (unLogin) {
                Toast.makeText(getApplicationContext(), "手机号和密码不匹配",
                Toast.LENGTH_SHORT).show();
            }
            if (flag) {
                Toast.makeText(getApplicationContext(), "用户名不存在",
                Toast.LENGTH_SHORT).show();
            }
        }
    }
})
```

```

        }
    }
} else if (checkedId == R.id.rb_register) {
    reg_phone = et_login_list1.getText().toString();
    reg_code = et_login_list2.getText().toString();
    reg_nikeName = et_login_list3.getText().toString();
    reg_password = et_login_list4.getText().toString();
    Log.d(TAG, "onClick: 注册");
    if (!Objects.equals(reg_code, verifycode) ||
reg_code.isEmpty()) {
        Toast.makeText(getApplicationContext(), "验证码错误",
Toast.LENGTH_SHORT).show();
    } else {
        boolean flag = true;
        for (User user : listitem) {
            if (Objects.equals(user.getNikeName(),
reg_nikeName)) {
                Toast.makeText(getApplicationContext(), "昵称已被注册",
Toast.LENGTH_SHORT).show();
                flag = false;
                break;
            }
        }
        if (flag) {
            //成功注册
            User user1 = new User(reg_phone, reg_password,
reg_nikeName);
            userHelper.openWriteLink();
            userHelper.insert(user1);
            Toast.makeText(getApplicationContext(), "注册：" +
reg_nikeName + "成功", Toast.LENGTH_SHORT).show();

            Intent intent = new Intent("registerSuccess");
            intent.putExtra("nickName", user1.getNikeName());
            Log.d(TAG, "onClick: 发送广播");
            sendBroadcast(intent);

            //查一下id值
            for (User user : userHelper.queryAll()) {
                if (Objects.equals(user.getNikeName(),
user1.getNikeName())) {
                    user1 = user;
                }
            }
            Log.d(TAG, "onClick: 跳转");
            startActivity(new Intent(LoginActivity.this,
MainActivity.class).putExtra("uid", user1.getId()));
        }
    }
}
);

```

把UserId发送到MainActivity后，在MainActivity中获取uid并存入sharedPreference中代码实现

```

Intent intent = getIntent();

if (intent != null) {
    Log.d(TAG, "onCreate: 获取uid");
    uid = intent.getLongExtra("uid", 0);
    //缓存到sharedPreferences中
    //加载sharedPreferences
    sharedPreferences = getSharedPreferences("config", MODE_PRIVATE);
    SharedPreferences.Editor edit = sharedPreferences.edit();
    edit.putString("uid", string.valueOf(uid));
    edit.commit();
}

```

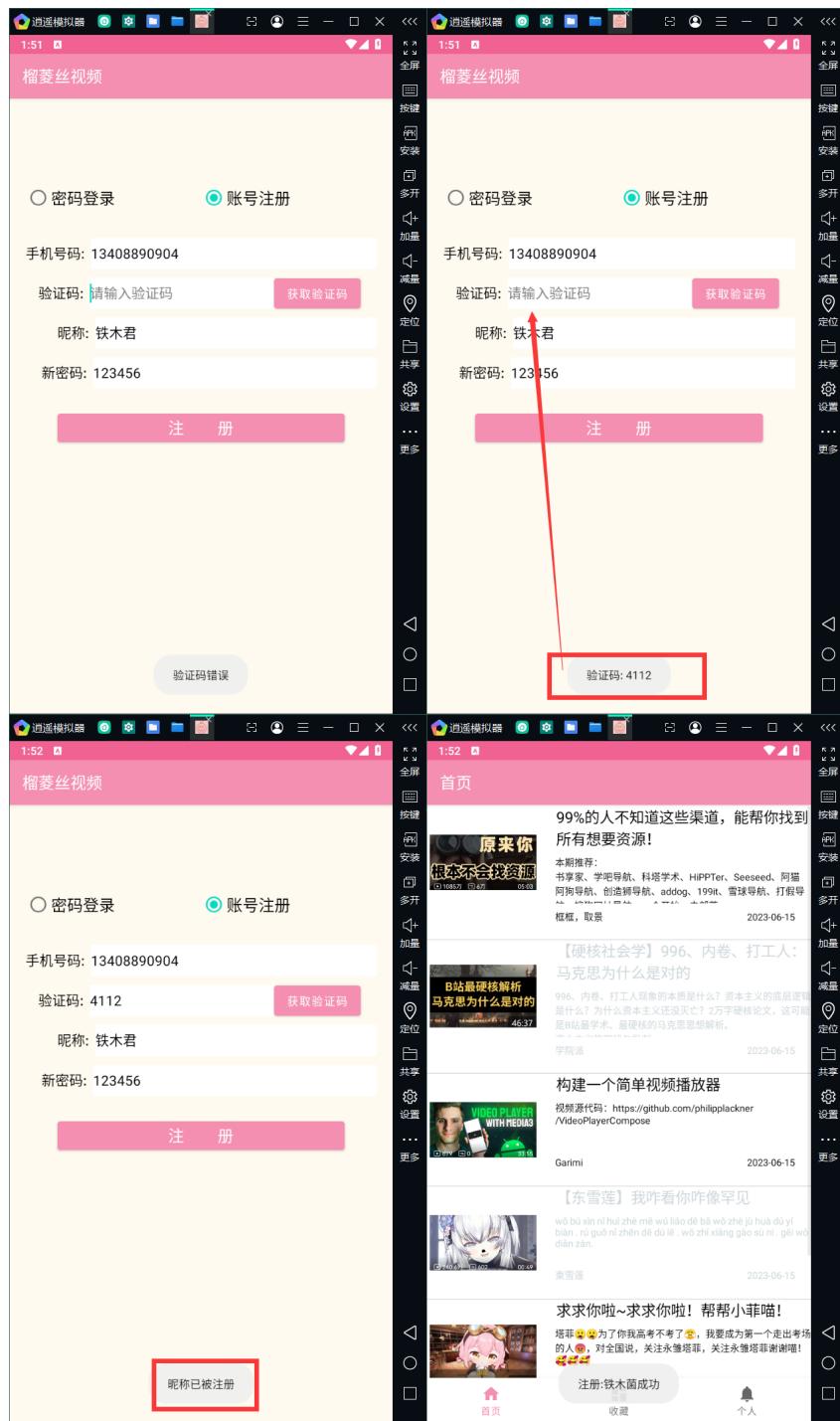
随机生成验证码代码实现

```

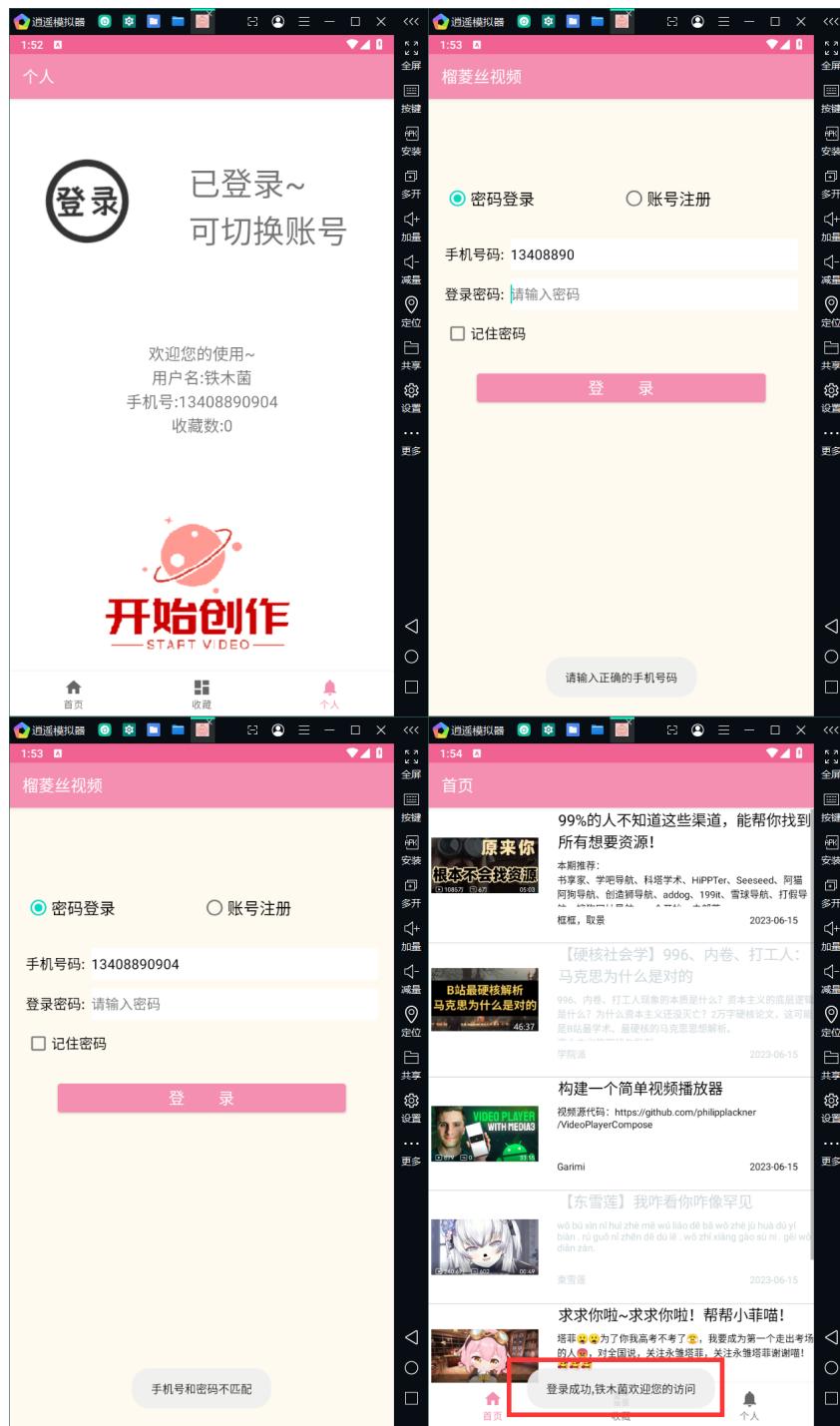
btn_get_verifycode.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (v.getId() == R.id.btn_get_verifycode) {
            if (et_login_list1.getText().toString().length() < 11) {
                Toast.makeText(getApplicationContext(), "请输入正确的手机号码",
                        Toast.LENGTH_SHORT).show();
            } else {
                boolean flag = true;
                for (User user : listitem) {
                    if (Objects.equals(user.getPhoneNumber(),
                            et_login_list1.getText().toString())) {
                        Toast.makeText(getApplicationContext(), "手机号已被注册",
                                Toast.LENGTH_SHORT).show();
                        flag = false;
                    }
                }
                if (flag) {
                    try {
                        SecureRandom secureRandom =
                        SecureRandom.getInstance("SHA1PRNG");
                        secureRandom.setSeed(10000L);
                        verifycode =
                        String.valueOf(secureRandom.nextInt(10000));
                        Toast.makeText(getApplicationContext(), "验证码: " +
                        verifycode, Toast.LENGTH_SHORT).show();
                    } catch (NoSuchAlgorithmException e) {
                        throw new RuntimeException(e);
                    }
                }
            }
        }
    });

```

最终呈现效果:



----->>



个人信息

登录后从SharedPreferences加载uid信息

```
//加载SharedPreferences
sharedPreferences = getActivity().getSharedPreferences("config",
MODE_PRIVATE);
String uid = sharedPreferences.getString("uid", "");
```

动态检测uid的值并查询数据库把登录用户的信息打印在个人信息栏上

```
new Handler().postDelayed(new Runnable() {
@Override
public void run() {
Log.d(TAG, "run: " + uid);
if (!uid.equals("") && uid != null && !uid.equals("0")) {
```

```
User user = userHelper.queryById(Long.parseLong(uid));
List<Star> stars = starHelper.queryAll();
Long userId = user.getId();
int starCount = 0;
for (Star star : stars) {
    if (star.getUid() == userId) {
        starCount++;
    }
}
textView = view.findViewById(R.id.text_notifications);
textView.setText("欢迎您的使用~" + "\n用户名:" +
user.getNikeName() + "\n手机号:" + user.getPhoneNumber() + "\n收藏数:" +
starCount);
loginText = view.findViewById(R.id.loginText);
loginText.setText("已登录~\n可切换账号");

}
},
}, 500);
```

最终呈现效果：



投稿视频

ps:

这部分代码获取了一大堆数据，调用了SQLite数据库和sharedPreferences，发送了广播，最后进行页面跳转。

写的比较乱就不展示了...

操作逻辑：

点击开始创作，跳转到addForm这个活动

1.往minio中添加视频，获取视频链接<http://mingyi.fun:9000/video/v6.mp4>

2.获取封面图片序号 2131165398

3.编写视频标题和简介

4.点击投稿视频，自动获取昵称和当前时间，把视频加入数据库，可以在首页看到

5.点击投稿视频后可以在主页看到广播提示“有人投稿了新视频”

6.点赞后视频点赞量增加，收藏后视频在收藏夹和个人信息中正常显示

最终完成结果展示：

MinIO Browser

Used: 533.64 MB

Name	Size	Last Modified	...
v6.mp4	43.87 MB	Jun 15, 2023 6:48 PM	...
v1.mp4	272.51 MB	Jun 10, 2023 9:25 PM	...
v3.mp4	18.48 MB	Jun 10, 2023 9:22 PM	...
v4.mp4	10.23 MB	Jun 10, 2023 9:22 PM	...
v5.mp4	2.85 MB	Jun 10, 2023 9:21 PM	...
v2.mp4	185.70 MB	Jun 10, 2023 5:24 PM	...

榴菱丝视频

视频标题 如何破解迷茫，找到生活方向？

视频简介 集合了年轻人最关注的难题，从克服自卑、消除内耗，到如何戒掉手机，爱上学习，应有尽有，曾帮助了一些朋友，今天整理出来给大家，方便按需查找，喜欢的话不妨三连起来，防止下次要看找不到啦！

大家好，先感谢你们的三连支持呀！

我把这一年做的干货视频基本都整理出来了，覆盖范围较广，希望能解决你们的烦恼，成为一个比较全面的工具检索~不过这只是我内容的冰山一角，我还会输出许多对大家有用的干货、观点解决等，这波关注不亏（捂嘴笑）

视频链接 <http://mingyi.fun:9000/video/v6.mp4>

全屏
按键
安装
多开
加量
减量
定位
共享
设置
更多

首页

99%的人不知道这些渠道，能帮你找到所有想要资源！

原来你根本不会找资源

本期推荐：
书享家、学吧导航、科塔学术、HiPPTer、Seeseed、阿猫阿狗导航、创造狮导航、addog、199it、雪球导航、打假导框框，取景

2023-06-15

【硬核社会学】996、内卷、打工人：

全屏
按键
安装
多开
加量

**B站最硬核解析
马克思为什么是对的**

46:37

马克思为什么是对的

996、内卷、打工仔现象的本质是什么？资本主义的底层逻辑是什么？为什么资本主义还没灭亡？2万字硬核论文，这可能是B站最学术、最硬核的马克思思想解析。

学院派 2023-06-15

构建一个简单视频播放器

VIDEO PLAYER WITH MEDIA3

Garimi 2023-06-15

【东雪莲】我咋看你咋像罕见

wō bù xìn nǐ hūi zhè mē wú liáo dē bā wō zhè jù huà dū yí biàn . rú guǒ nǐ zhēn dē dú lē . wǒ zhī xiāng gào sù nì . gēi wǒ diǎn zàn.

東雪莲 2023-06-15

求求你啦~求求你啦！帮帮小菲喵！

塔菲 😱为了更好了高考不考了，我要成为第一个走出考场的人，对全国说，关注永维塔菲，关注永维塔菲谢谢喵！

有人投稿了新视频，请查看

首页 收藏 个人

逍遥模拟器 2:00 全屏 按键 安装 多开 加量 减量 定位 共享 设置 ...

首页

构建一个简单视频播放器

VIDEO PLAYER WITH MEDIA3

Garimi 2023-06-15

【东雪莲】我咋看你咋像罕见

wō bù xìn nǐ hūi zhè mē wú liáo dē bā wō zhè jù huà dū yí biàn . rú guǒ nǐ zhēn dē dú lē . wǒ zhī xiāng gào sù nì . gēi wǒ diǎn zàn.

東雪莲 2023-06-15

求求你啦~求求你啦！帮帮小菲喵！

塔菲 😱为了更好了高考不考了，我要成为第一个走出考场的人，对全国说，关注永维塔菲，关注永维塔菲谢谢喵！

永维塔菲 2023-06-15

骂谁罕见啊？

骂谁罕见！！！

1 2023-06-16

如何破解迷茫，找到生活方向？

这个合集值得所有年轻人收藏

59:57

铁木菌 2023-06-16

首页 收藏 个人



逍遥模拟器

2:03 全屏

榴莲丝视频

如何破解迷茫，找到生活方向？

铁木菌 2023-06-16

但请相信我，花8分钟时间，认真看到最后

点赞 收藏 点赞:5 收藏:1

，希望能解决大家的困惑，成为一个人脉广泛的工具检索~不过这只是我内容的冰山一角，我还会输出许多对大家有用的干货、观点解决等，这波关注不亏（捂嘴笑）

对啦，也可以关注我的公众号「铁木君」，回复相应关键词，能获取许多免费又好用的资源哦~比如：

回复“读书”，我给你分享3000本精选书籍+铁木君私人高分书单。

回复“技能”，送你我这些年购买、收集的写作、软件基操、摄影等珍贵教程

[点击链接下载视频详情](#)

按键 安装 多开 加量 减量 定位 共享 设置 更多

回 回



