# 极简抖音项目介绍

## 1．成员分工

熊家辉：负责基础接口：视频流、发布、登录注册、用户信息、发布列表的编写。

赵冬阳：负责点赞操作、用户点赞列表接口的编写。

唐杰辰：负责评论操作、用户评论列表接口的编写。

文超：负责关注操作、关注列表和粉丝列表接口的编写。

## 2. 项目**灵感**

参与字节青训营学习的同时，锻炼技术，深化理论知识，增强实践能力，实现一个极简版的抖音。在项目实践中，切实的实践了各位老师在课程中讲解的知识点，如语言基础、数据库、gin框架、编码规范等内容。

## 3. 项目亮点

为了提高团队开发效率，该项目分层清晰，分为DAO、DTO、CONTORLLER、ROUTER、COMMON、STATIC等6层。各层间通过方法进行调用，耦合度低，极大的提高了团队开发效率。实现了基础接口、扩展接口Ⅰ和扩展接口Ⅱ。性能测试良好。

## 4. 技术说明

本项目使用Gin框架+Gorm+MySQL+ffmpeg等技术。

### 4.1基本接口

**视频流接口：**

1．通过输入参数的时间戳来查询该时间戳之前所发布的最多30条视频，按投稿时间（created\_at）排序组成视频列表。并记录视频列表中最新的视频发布时间。

2. 若输入参数含有token则查询该token所属用户信息，通过信息中的followlist、FavoriteList等信息判断，登录用户是否点赞了该视频，是否关注了该用户。

3. 将dao中查询到的视频列表等信息，组装到dto的输出对象中并返回。

**用户注册：**

1. 首先对输入的用户名是否已经存在进行查询，若存在则返回用户已注册。

2. 对输入的密码与salt一起加密运算，生成加密字符串保存在数据库中

3. 登录生成token并返回

**用户登录：**

1. 首先检查是否存在该username

2. 对输入的密码加盐加密并与数据库中的加密字符串比较，相同则为成功

3. 将用户名、时间和私钥，三者加密拼接生成token，设置存活时间为3600s

4. 返回token

**用户信息：**

1．先对用户的token进行验权（是否有效、是否存在）

2．若存在并有效时间内，则取出用户信息，在dto中组装并返回。

**视频投稿：**

登录用户选择视频上传

1．处理文件名并视频文件存在本地的静态资源目录中。

2．使用ffmpeg对上传的视频抽取关键帧，生成封面图。

3．将以上信息整理并插入video数据表中，返回状态信息。

**发布列表：**

1．根据用户请求的token进行鉴权（是否存在，是否有效），并获取登录用户id。

2. 根据用户请求的userid查询该用户的所有video。

3. 根据video中的FavoriteList，判断登录用户是否点赞该视频。根据视频作者中的FollowerList判断登录用户是否关注该作者。

4. 遍历执行第三步组成video列表并返回。

### 4.2扩展接口Ⅰ

**点赞操作：**

mysql数据库video表的Favorite\_List字段存储所有的点赞用户id，中间用#分割开来，Favorite\_count存储该视频点赞的用户数量。

当用户点赞时，前端向后端传回token和视频id以及点赞或者取消点赞的标识信息，后端通过token验证用户并获得用户实体类，然后修改video表的当前对象的Favorite\_count+1或者-1，并对Favorite\_List的字符串进行修改。

**点赞列表：**

前端返回用户token，后端通过token验证用户并获得用户实体类，并查询video类的Favorite\_List字段中含有当前用户id的videolist并进行返回。

**评论操作：**  
 mysql数据库comments表的user\_id字段存储所有评论的用户id，comment\_text字段存储评论内容，video\_id字段存储评论对应的视频id。

当用户评论时，前端向后端传回token、视频id、评论内容、发布或删除评论的标识信息，若删除评论，则还需要评论的id。后端接收到以上数据后，先验证token合法性，再通过token得到用户id，最后在数据库的video表中检查视频id是否存在。验证检查工作完成无误后，根据标识信息选择添加操作或删除操作。添加操作包括储存用户的评论信息和对应的用户id、视频id等信息，填充comments表。删除操作包括查询评论id是否存在，不存在则返回错误信息，存在则删除。若评论操作成功响应，则向前端返回状态码0、状态描述、评论内容、用户信息、发布日期等。若响应失败，则状态码非0。  
**评论列表：**

当用户点开评论区时，前端向后端传回token和视频id。后端接收到以上数据后，先验证token合法性，再通过token得到用户id，最后检查视频id。验证检查工作完成无误后，根据视频id，查询数据库中的comments表吗，得到评论id、创建时间、用户id和评论内容。若评论列表响应成功，则向前端返回状态码0、状态描述和评论列表，其中评论列表包括评论id，用户id、用户名称、关注总数、粉丝总数、是否关注、评论内容、评论发布日期；若响应失败，则返回状态码非0。

### 4.3扩展接口Ⅱ

**关注操作：**

对前端获取的参数token进行检查，判断当前token是否过期或者错误；过滤关注自己或取消关注自己这种情况。数据库users表中followList字段表示关注列表；followerList字段表示粉丝列表；followCount表示关注人数；followerCount表示粉丝人数。利用当前对象id和参数to\_user\_id即可到users表中查询到两个对象的信息，看是否存在，不存在则返回相应的错误；存在则对其进行修改更新即可。

关注即是把to\_user\_id添到id用户的followList字段中，还要把id添加到to\_user\_id用户的followerList字段中；对应的followCount和followerCount都要进行+1操作。当然，在进行操作之前，我们就要对其进行查重，防止重复关注，导致关注和粉丝总人数出错。就是如果字符串to\_user\_id#在id的followList字符串中，我们即返回不进行操作，不存在则需要进行修改操作。我们进行数据保存时，储存的等长的字符加一个#为分隔符，方便进行查重；

取消关注即是把to\_user\_id从id用户的followList字段中删除，还要把id从to\_user\_id用户的followerList字段中删除；对应的followCount和followerCount都要进行-1操作。当然，在进行操作之前，我们就要对其进行查找是否存在，防止本没有关注，总人数

-1，导致关注和粉丝总人数出错。也就是如果字符串to\_user\_id#不在id的followList字符串中，我们即返回不进行操作，存在则需要进行修改操作。

**关注列表：**

从前端获取to\_user\_id（要查看id的关注列表）和token。对获取的参数token进行一个检查，判断当前token是否过期或者错误；查看数据库中users表的to\_user\_id对象是否存在，不存在则返回相应的错误；存在则获取其的followList字符串，对字符串通过#进行分割，获取每个用户的id，通过id获取users表中对应的信息。然后对获取信息进行组装输出。

**粉丝列表：**

从前端获取to\_user\_id（要查看id的关注列表）和token。对获取的参数token进行一个检查，判断当前token是否过期或者错误；查看数据库中users表的to\_user\_id对象是否存在，不存在则返回相应的错误；存在则获取其的followerList字符串，对字符串通过#进行分割，获取每个用户的id，通过id获取users表中对应的信息。然后对获取信息进行组装输出。

## 5.难点突破

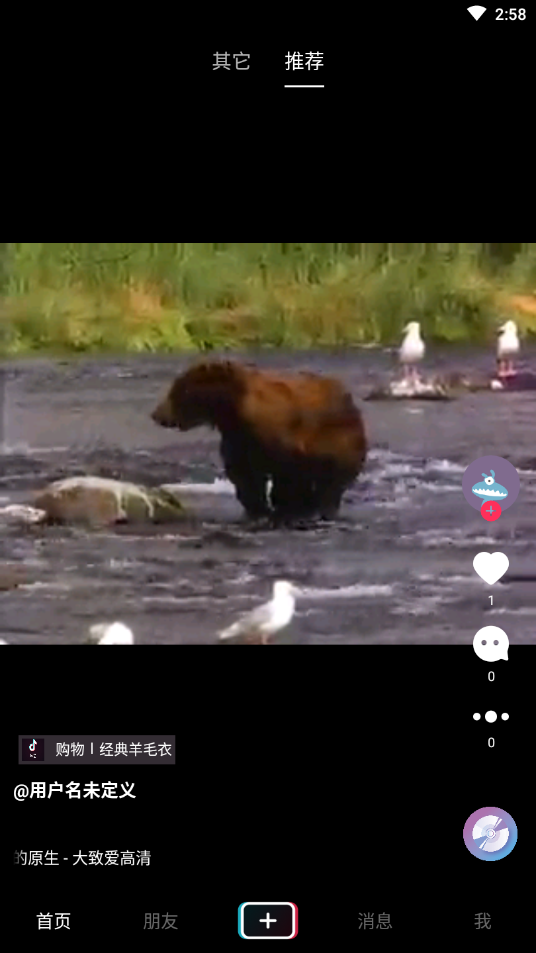
视频投稿中的封面图，采用ffmpeg对输入的视频进行抽取关键帧生成jpeg图片，并保存。

关注策略采用创建一个关注表，记录userA和userB之间的关系。

## 6.成果展示

### 基础接口：

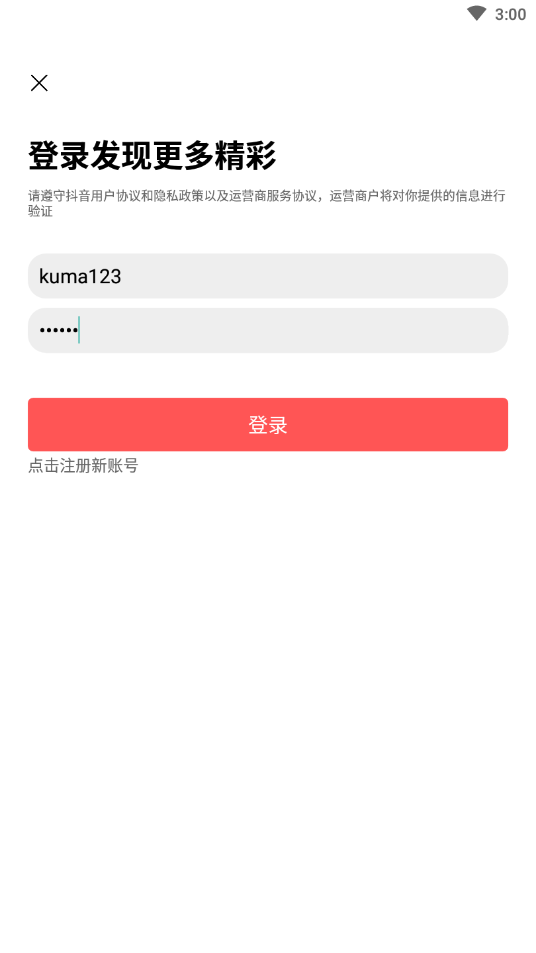
**视频流接口：**



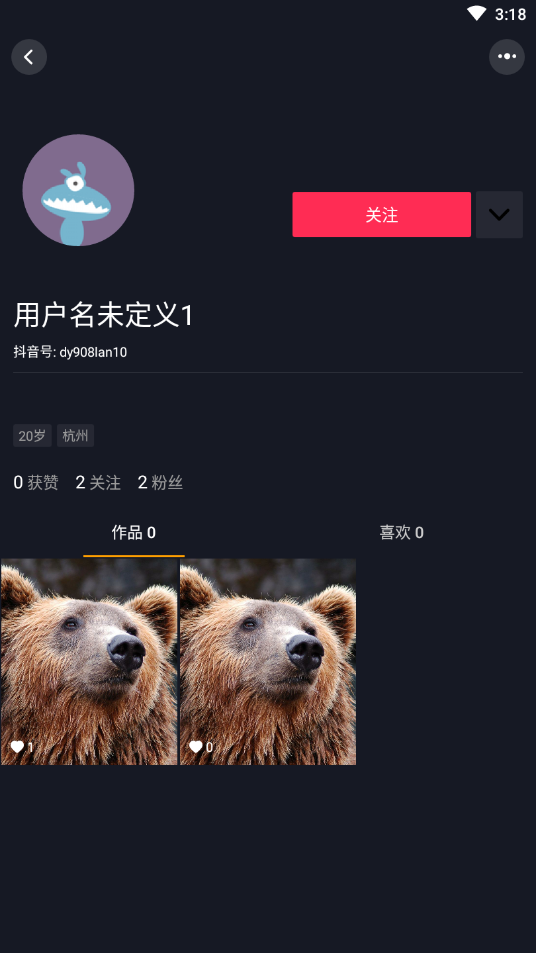
**用户注册：**



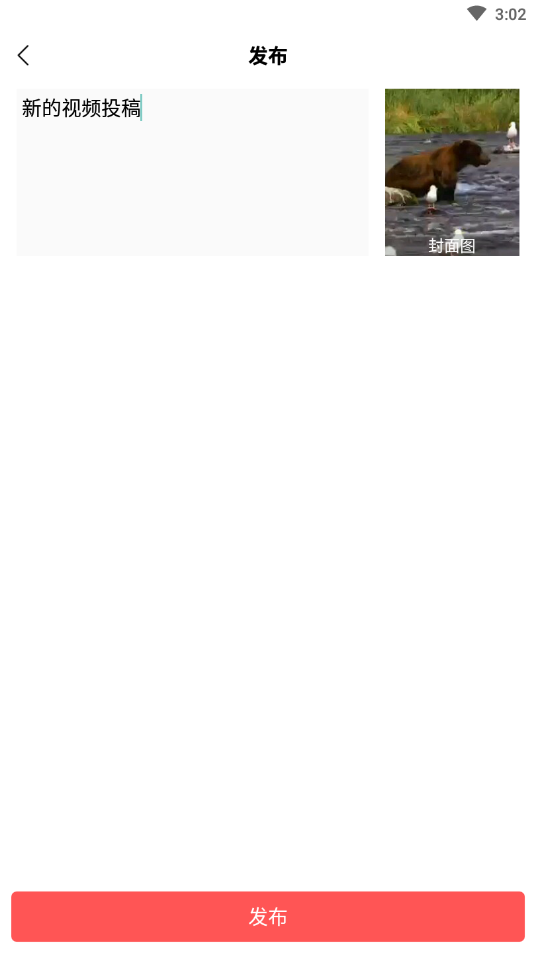
**用户登录：**



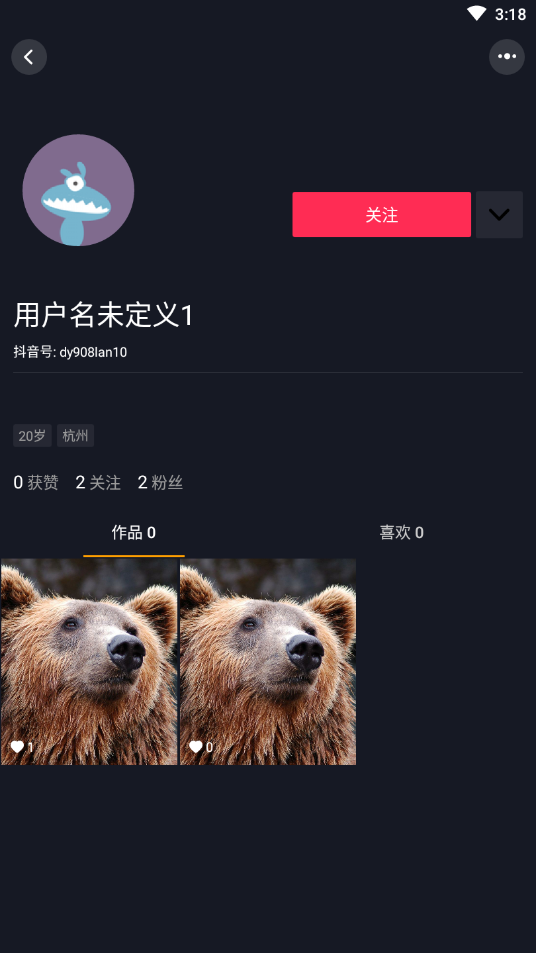
**用户信息：**



**视频投稿：**

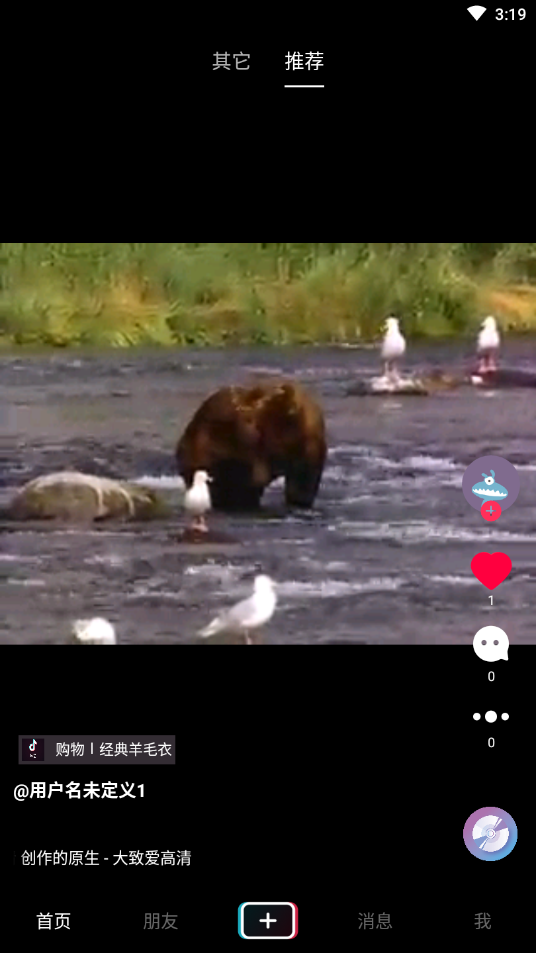


**发布列表：**



### 扩展接口Ⅰ

**点赞操作：**



**点赞列表：**



**评论操作：**



**评论列表：**



扩展接口Ⅱ

**关注操作：**



**关注列表：**



**粉丝列表：**

