# 投稿接口、发布列表模块实现

## 接口实现

接口一：登录用户选择视频上传（/douyin/public/action/）

1. **从POST请求中获取用户信息（username）和视频信息（title，data）**
2. **若当前用户不存在，返回statuscode为1，结束视频上传；若用户存在，为了防止文件名冲突，采用如下的命名方式：查询数据库获得该视频发布者的用户ID及该视频在数据库中即将对应的ID，使用ffmpeg获得视频第一帧作为视频方面并保存在本地，将视频标题一起作为视频文件和视频封面文件在本地的文件名保存在/public目录下。**
3. **获取当前服务器域名并拼接视频和封面静态数据URL，将视频相关信息保存数据库字段对应的结构体中,并以此添加到video数据库中；**
4. **投稿完成后返回响应。**

* **Controlle层——publish.go**

1. **Publish函数：视频发布的处理**

* **Service层——video.go**

1. **VideoListResponse结构体：表示视频切片及相关信息的响应结构体**

**type VideoListResponse struct {**

**repository.Response**

**VideoList []Video `json:"video\_list"`**

**Token string `json:"token"`**

**}**

1. **GetServerDomain函数：获取服务器的域名**
2. **Publish函数：发布并保存上传文件到公共目录**

* **repository层——video.go**

1. **TableName函数：Video 结构体的一个方法，它返回字符串值 "videos"。这个方法用于指定 Video 结构体在数据库中对应的表名**
2. **VideoDao结构体：封装视频数据的访问和操作逻辑（注：无具体的属性）**

**type VideoDao struct {**

**}**

1. **NewVideoDaoInstance函数：获取 VideoDao 的单例实例，确保只有一个实例存在**
2. **QueryVideoByAuthor函数：按照作者的用户名查询视频列表**
3. **QueryVideoFeed函数：按照 Feed 的要求返回视频列表**
4. **QueryVideoLatest函数：查询下一个主键 ID**
5. **CreateVideo函数：在数据库中创建视频记录**
6. **UpdateVideoUrl函数：更新视频的 URL**
7. **UpdateUserFavcount函数：更新用户的点赞总数和作品总数**

接口二：用户的视频发布列表，直接列出用户所有投稿过的视频（/douyin/public/list/）

1. **由于每次重启代码空间，域名会改变，所以需要根据当前域名修改数据库中的URL；**
2. **获取当前页面的用户，并在video数据库中查询该用户发布的所有视频信息，将其信息保存在videos列表中；**
3. **videos中保存的数据是与数据库字段对应的信息，将其转换为前端需要的json列表；**
4. **返回响应和json数据**

* **Controlle层——publish.go**

1. **PublishList函数：获取发布视频列表**

* **Service层——videoService.go**

1. **Video结构体：表示视频对象的数据结构体**

**type Video struct {**

**Id int64 `json:"id"`**

**Author repository.User `json:"author"`**

**PlayUrl string `json:"play\_url"`**

**CoverUrl string `json:"cover\_url"`**

**FavoriteCount int64 `json:"favorite\_count"`**

**CommentCount int64 `json:"comment\_count"`**

**IsFavorite bool `json:"is\_favorite"`**

**Title string `json:"title"`**

**}**

1. **ConvertVideoDBToJSON函数：将数据库中的视频对象转换为 JSON 格式返回**

* **Service层——video.go**

1. **PublishList函数：获取发布视频的列表，并对用户的令牌进行验证和处理**

接口三：视频Feed流：支持所有用户刷抖音，视频按投稿时间倒序推出（/douyin/feed）

1. **获取当前登录用户的token，若用户信息不存在，则返回错误信息；**
2. **若用户存在或用户未登录，调用服务层的函数完成视频流查询；**
3. **服务层的Feed函数先根据当前服务器域名更新数据库中的URL，然后按照时间顺序倒序查询video数据库中的视频，并返回一个结构体切片，该切片表中包含视频的基本信息；**
4. **然后查询当前用户的点赞信息，并返回当前用户点赞的视频ID数组，若之前返回的包含视频信息的切片中与当前用户点赞视频存在交集，则将视频信息的is\_favourte改为true，并返回视频切片和响应。**

* **Controlle层——feed.go**

1. **Feed函数：查询当前登录用户，返回视频流信息**

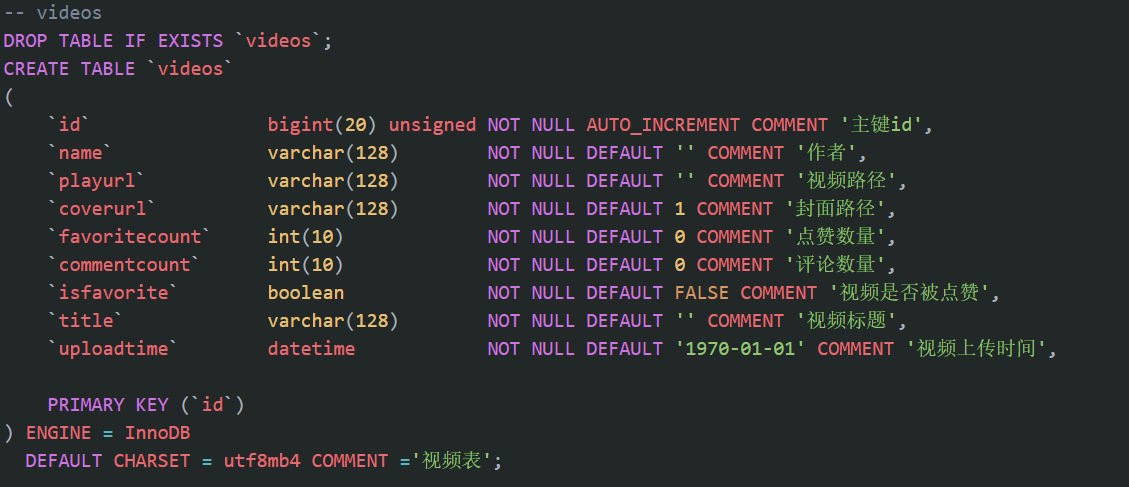
* **Service层——feed.go**

1. **Feed函数：从数据库中查询视频流信息和当前用户点赞信息，将其合并返回给APP；**

* **Repository层——video.go**

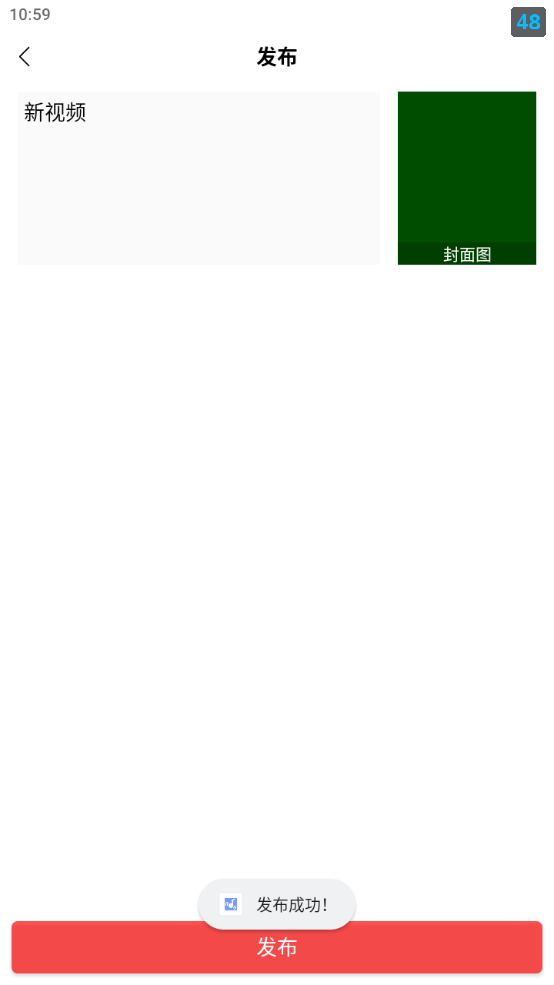
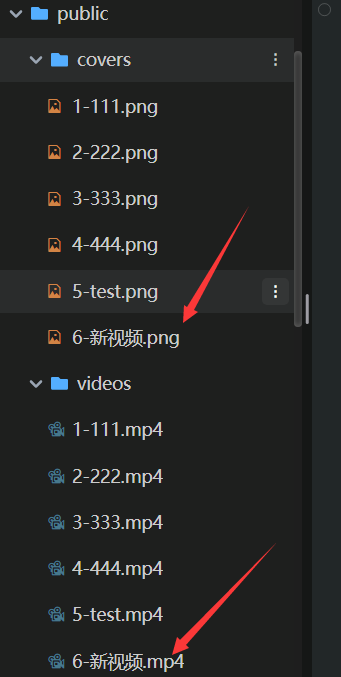
1. **QueryFavoriteVideoIdbyUsername：获取当前用户的点赞视频ID**
2. **数据库设计**

**1、视频表**



1. **测试结果**

**视频发布测试：**



**视频列表测试：**



**视频流测试：**

