

惜奥动漫小程序

1. 说明

1.1 概述

本微信小程序直接观看动漫，搜索一下某个动漫下载自己喜欢的动漫选集，可以进行对某个动漫进行评价，说出自己的想法，可以进行收藏，对你看过的动漫会进行历史记录，可以发表你的意见，可以联系客服

1.2 前端和后端技术

前端技术： 微信小程序 CSS JAVASCRIPT

Wux-Weapp 组件库

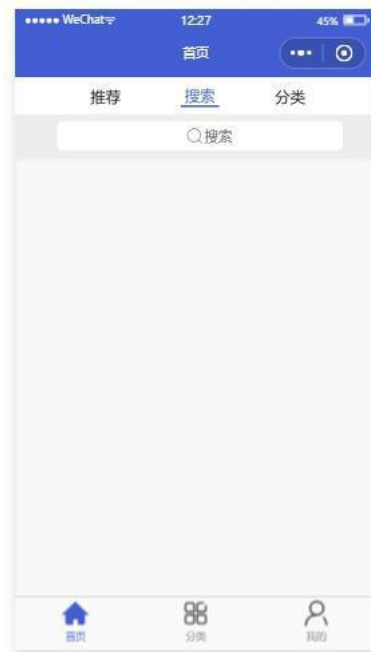
后端技术(云开发): 云数据库 云函数 云存储

2. 项目结构

页面名称	名称
首页	index
推荐	recomment
搜索	search
视频中心	video
分类	category
我的	user
历史记录	history
收藏	collect
云函数	addcommet addhistory changecount updateCommnet

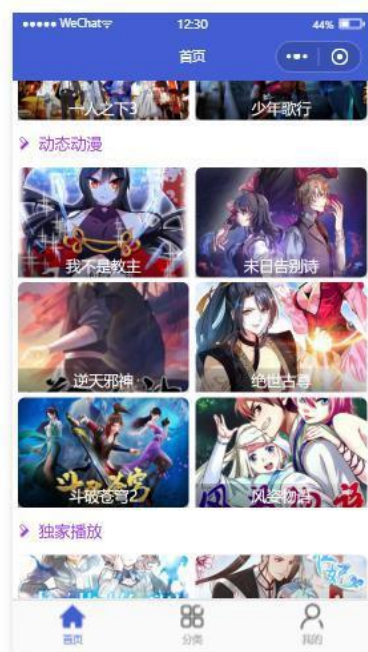
3. 功能

1. 首页：由推荐 搜索 分类 组成



推荐:

底部由轮番图和列表(热门推荐 动态动漫 独家播放)组成，
点击列表可以跳转到视频中心



搜索:

在搜索框输入你想要的动漫，可以出现相关动漫，这样更快找到你要的动漫

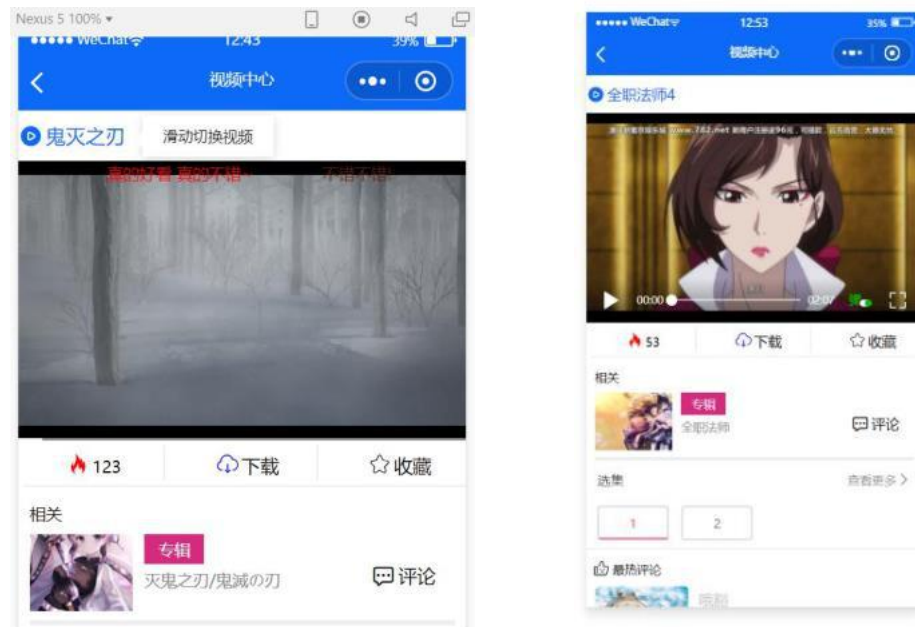


分类:

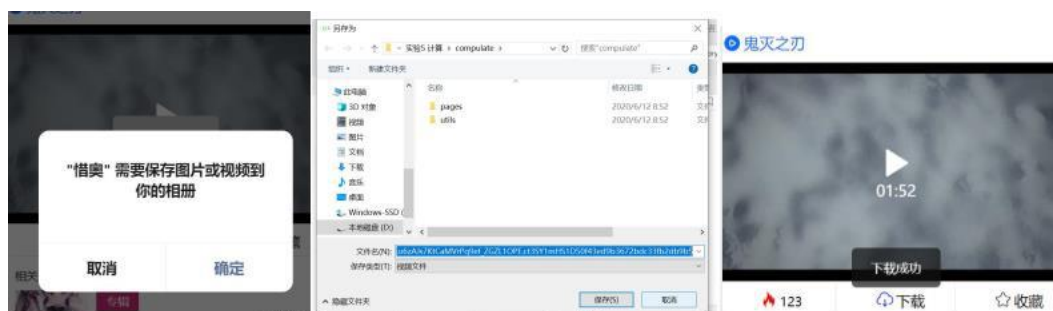
点击分类，会跳转分类界面，这样更方便的跳转

2.视频中心:

(1)点击视频图标的文字,会出现滑动视频的弹出框,在这区域可以向左或者向右进行切换视频内容,视频可以看到弹簧内容.



(2)点击下载可以下载视频



(3)点击收藏可以收藏此视频或者点击收藏可以取消视频



(4) 点击评论，用户被要求先登录，然后才能评论，在评论面板中，获取了用户的像和名称，点击评论框，输入评论内容，然后点击确定，可以看到自己的评论内容



(5) 选集下面是一个可以向左滑动的，这样可以出现更多的选集，点击查看更多会出现更多的选集内容，你可以选择其中选集内容

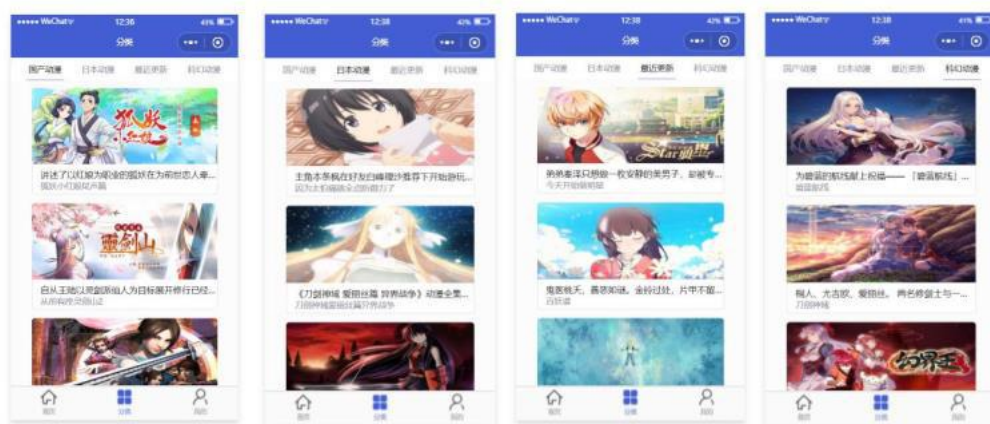


(6) 最热评论下的评论，你可以点赞，但只能点赞一次。



3.分类:

分类有国产动漫 日本动漫 最近更新 科幻动漫 每一个可以看到动漫图片 动漫描述 动漫名称



4. 我的:

登录 历史记录 我的收藏 意见反馈 联系客服



登录:

获取用户的信息 ， 如果授权就登录成功



历史记录

点击更多



我的收藏



意见反馈



联系客服



4. 云开发云函数

(1) addcomment

```
cloudfunctions | 当前环境: hyx07180913 > addcommet > JS index.js > m
1 // 云函数入口文件
2 const cloud = require('wx-server-sdk')
3
4 cloud.init({
5   env: 'hyx07180913-9kgac'
6 })
7 const db = cloud.database();
8 const _ = db.command;
9 const commetCollcetion = db.collection('Anime_Commet')
10 // 云函数入口函数
11 exports.main = async (event, context) => {
12   let {
13     id,
14     commet_image,
15     title,
16     navigateto_url,
17     commet_name,
18     commet,
19     hot
20   } = event;
21   let res;
22   let data = []
23   data = await commetCollcetion.where({
24     id:id
25   }).get();
26   if(data.data.length !== 0){
27     //进行删除
28     if(commet === false){
29       if(data.data.length !== 0){
30         //进行删除
31         if(commet === false){
32           res = await commetCollcetion.where({id:id}).remove();
33         }
34         return
35       }
36       res = await commetCollcetion.add({
37         data: {
38           id,
39           commet_image,
40           title,
41           commet:true,
42           navigateto_url,
43           commet_name,
44           hot
45         }
46       })
47     }
48     return {
49       res
50     }
51   }
```

(2) addhistory

```
cloudfunctions | 当前环境: hyx07180913 > addhistory > JS index.js > mai
1 // 云函数入口文件
2 const cloud = require('wx-server-sdk')
3 cloud.init({
4   env: 'hyx07180913-9kgac'
5 })
6 const db = cloud.database();
7 const _ = db.command;
8 const historyCollection = db.collection('Anime_history')
9 // 云函数入口函数
10 exports.main = async (event, context) => {
11   let {
12     id,
13     history_image,
14     title,
15     navigateto_url,
16     history_name,
17     time
18   } = event;
19   let res;
20   let data = [];
21   data = await historyCollection.where({
22     id:id
23   }).get();
24   if(data.data.length !== 0){
25     return
26   }
27   res = await historyCollection.add({
28     data: {
29       id,
30       history_image,
31       (property) navigateto_url: any
32       navigateto_url,
33       history_name,
34       time
35     }
36   })
37   return {
38     res
39   }
40 }
```

(3) changecount

```

1 // 云函数入口文件
2 const cloud = require('wx-server-sdk')
3 cloud.init()
4 // 云函数入口函数
5 exports.main = async(event, context) => {
6   const {
7     id
8   } = event;
9   const {
10     index
11   } = event;
12   const db = cloud.database()
13   const _ = db.command
14   const cnadidate = db.collection('Anime_audio_info')
15   let res = await cnadidate.where({
16     _id: id,
17     'audio_comment.comment_id': index
18   }).update({
19     data: {
20       'audio_comment.$.comment_mount': _.inc(1),
21       'audio_comment.$.iszhang': true
22     }
23   })
24   return {
25     res
26   }

```

(4) updateComment

```

// 云函数入口文件
const cloud = require('wx-server-sdk')
cloud.init({
  env: 'hyx07180913-9kgac'
})
const DB = cloud.database();
// 云函数入口函数
exports.main = async(event, context) => {
  const _ = DB.command;
  let { id, comment_content, comment_src, comment_title, time, comment_id } = event
  const rgb = []
  for (let i = 0; i < 3; ++i) {
    let color = Math.floor(Math.random() * 256).toString(16)
    color = color.length === 1 ? '0' + color : color
    rgb.push(color)
  }
  let color = '#' + rgb.join('')
  let res = await DB.collection('Anime_audio_info')

```

```
18 let res = await DB.collection('Anime_audio_info')
19   .where({
20     audio_id: id
21   }).update({
22     data: {
23       audio_comment: _.push({
24         comment_id: comment_id+1,
25         comment_title: comment_title,
26         comment_content: comment_content,
27         comment_src: comment_src,
28         comment_mount: 0,
29         color: color,
30         time: time+1,
31         iszhang: false
32       })
33     }
34   })
35   return {
36     res
```