

移动应用开发

“NWPU 四季”微信小程序说明文档

2019301552 黄士航

| | |
|---------------------|---|
| 一、小程序说明： | 1 |
| 二、解决的实际问题： | 1 |
| 三、应用场景： | 1 |
| 四、产品特点： | 2 |
| 五、用户和系统需求分析： | 2 |
| 5.1 用户需求..... | 2 |
| 5.1.1 学生需求 | 2 |
| 5.2 系统需求..... | 2 |
| 5.2.1 功能需求 | 2 |
| 5.2.2 非功能需求 | 3 |
| 六、交互及界面设计： | 3 |
| 6.1 交互设计..... | 3 |
| 6.2 页面设计..... | 4 |
| 6.2.1 首页（导航页） | 4 |
| 6.2.2 共享页..... | 4 |
| 6.2.3 图片页..... | 5 |
| 6.2.4 上传页..... | 6 |
| 6.2.5 其他页面 | 7 |
| 七、技术方案 | 7 |
| 7.1 数据库..... | 7 |
| 7.2 云函数..... | 7 |
| 八、系统的运营和维护 | 8 |

一、小程序说明：

本程序名为“NWPU 四季“，是一个校园景色摄影作品分享平台，旨在解决校园美景的分享面过小的问题，同时也可以提示校园的文艺气息，丰富学生的日常生活，提升同学们的艺术素养。

共有六个界面，依次为首页，共享页，收藏页，个人主页，上传页，图片页，下面依次介绍各个页面的功能及作用。

首页是用户进入小程序时所看到的界面。在这个界面给用户提供了四个按钮，分别是“上传”“收藏”“我的”“圈子”，点击四个按钮即可跳转。

共享页是所用用户分享图片的页面，其中包含各个用户上传的图片。

收藏页是用户收藏的图片存放的页面。

个人主页是对于每一个用户的，各个用户都有一个个人主页，上面包含了其上传的所有图片。

上传页是进行上传图片的界面。

图片页包含了点赞，收藏，下载，分享等功能，还可以全屏查看图片。

二、解决的实际问题：

（1）解决“美景只在少数人手里”的问题：

美景的拍摄是需要地点，时机和拍摄技术三者的加成，一张美丽的景色照是很难得的，所以美景往往只会掌握在少数人的手里。通过我的小程序，大家可以在圈子里实现共享，这样使用小程序的每一个同学都可以看到。

（2）发挥摄影社的最大作用

我校的摄影社没有太多的共享平台，大多数摄影社的同学都是拍了照片发到自己的QQ空间里，这样传播面太小。通过这一小程序，摄影社可以有更广泛的传播面。同时，也可以帮助提高全校同学的文艺素养，工大浓厚的工科氛围也可以添加一点艺术气息。

（3）给摄影爱好者们一种新的交友方式

通过分享校园美景，互相欣赏的人可以添加微信好友，这也是一种很好的交友方式。

三、应用场景：

（1）西工大学生看到校园美景，就可以随手拍摄，上传。工大有很多的美景，只存在片刻。在某天某刻的一位同学拍到了很好看的校园美景，就可以把这张图上传到小程序上，这样所有的人都可以在”圈子“里面看到，这里的每一个图片都可以被全校的同学看到。

(2) 西工大摄影社也可以参与进来，发布他们的作品，在这个小程序上进行交流，也可以帮助同学们提高摄影技术。

四、产品特点：

本产品主要有三大特色：

特色 1：简单，明了。程序的界面明晰，上手快，易使用

特色 2：交互友好。在程序中使用 Toast 和 loading 进行与用户之间的交互，界面清晰，经过测试，各项功能响应速度均在 2 秒之内。

特色 3：使用云数据库和云存储来存储数据，有较高的安全性。用户上传的图片保存在云存储中，并且使用云数据库中的数据建立用户和数据之间的联系。

五、用户和系统需求分析：

5.1 用户需求

5.1.1 学生需求

该程序面向的用户主要是西工大在校学生，学生主要有四大需求，分别是发布，浏览，点赞，收藏，下载，分享。这些都在小程序中实现了。

5.2 系统需求

5.2.1 功能需求

- 发布

用户可以将自己本地地图片上传至小程序端

- 浏览

用户浏览自己地收藏，上传以及其他人上传的图片

- 点赞

这是属于用户表达喜好的交互功能，提高用户体验度

- 收藏

用户对非常喜爱但又不想立即下载的图片可以进行收藏

- 下载

用户对一些图片可以进行下载

- 分享

用户会有分享的需求，可以将图片分享给微信好友。

5.2.2 非功能需求

- 1、性能:用户对于小程序性能的需求在于小程序的启动速度，界面的流畅性，网络请求的延时。而这些性能可能会影响用户体验。因此在系统设计时，要保证小程序的流畅程度。
- 2、可靠性:鉴于每天访问小程序的用户很多，因此服务器必须可靠，要严格避免当服务器遭遇停电、断网等特殊情况下用户无法完成正常的使用等问题。
- 3、图片清晰度:每个同学上传到数据库中的图片大小以及清晰度都不一样，这就需要我们尽可能保证每个图片的清晰度，并且使图片在小程序中有合适地呈现方式。

六、交互及界面设计:

6.1 交互设计

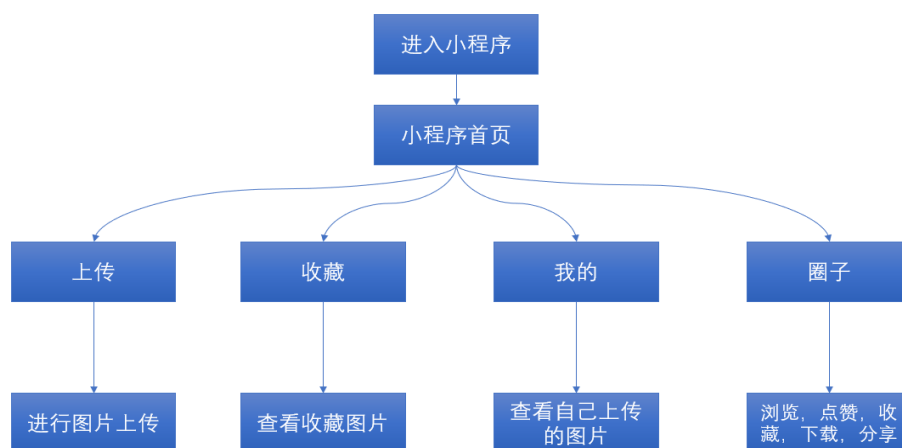


图 6.1 小程序整体逻辑图

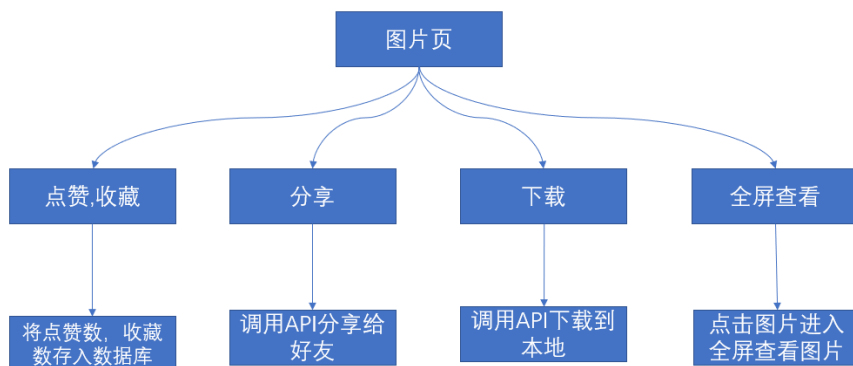


图 6.2 图片页（文物展示页）逻辑图

6.2 页面设计

使用了歼 20 战斗机的图片，同时顶部为工大蓝。

6.2.1 首页（导航页）

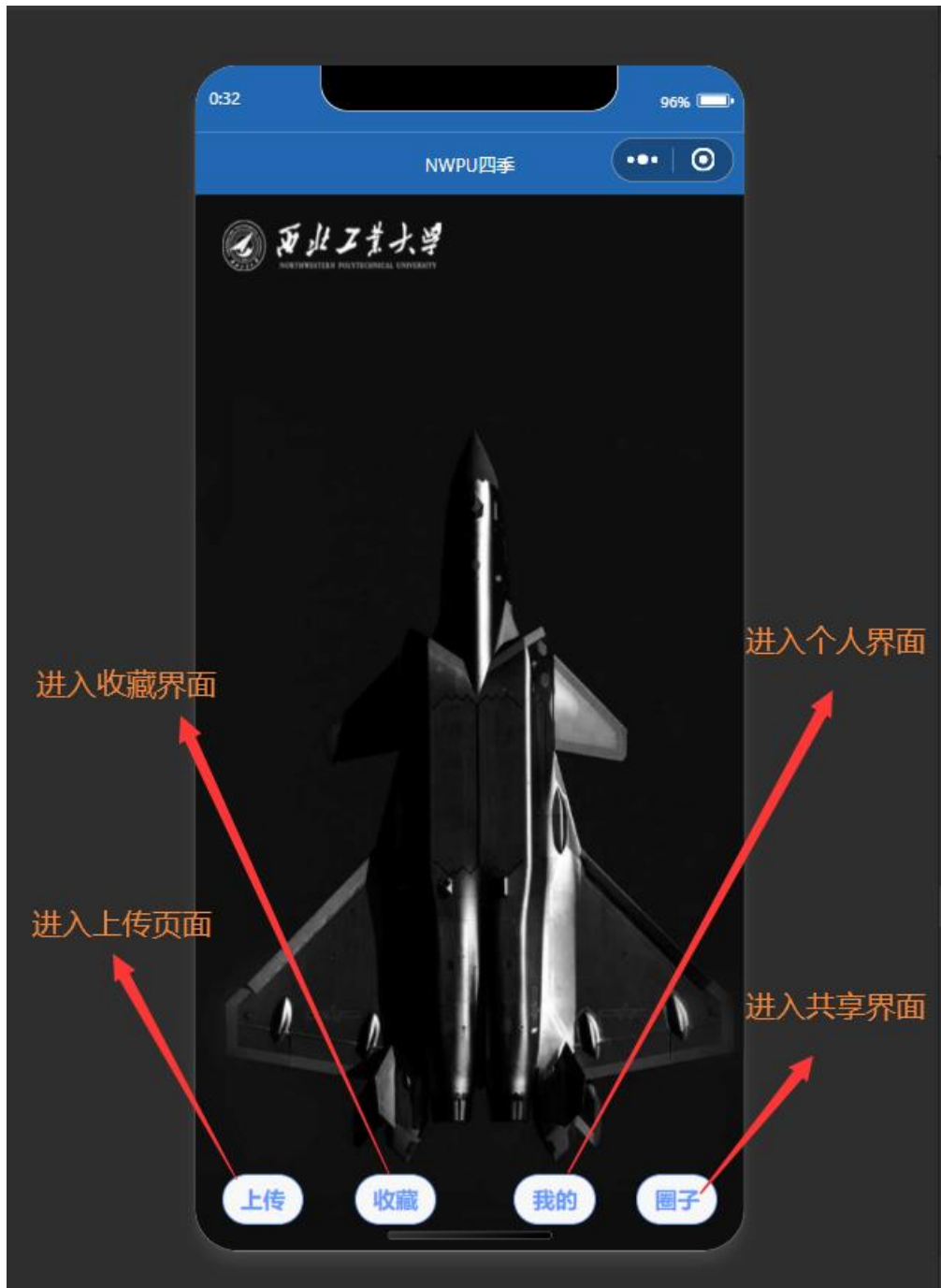


图 6.1 首页界面交互图

6.2.2 共享页

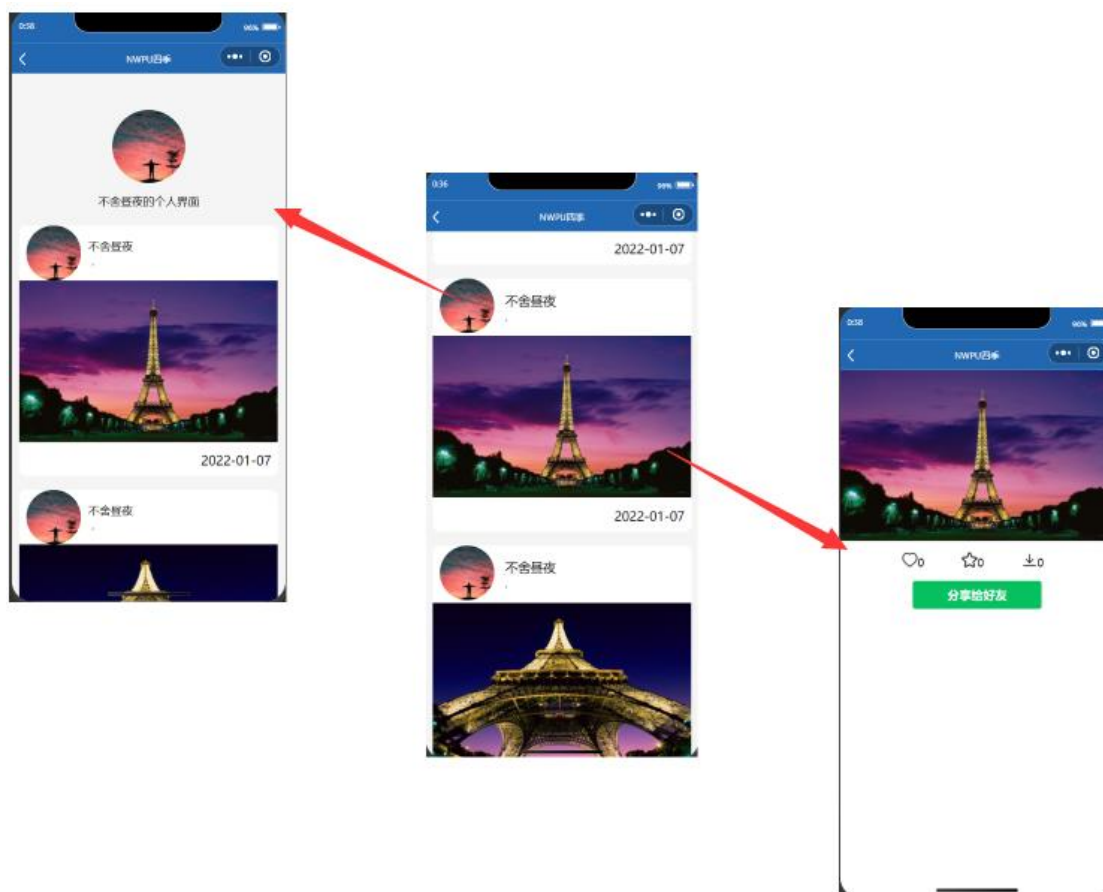


图 6.2 共享页交互图

6.2.3 图片页

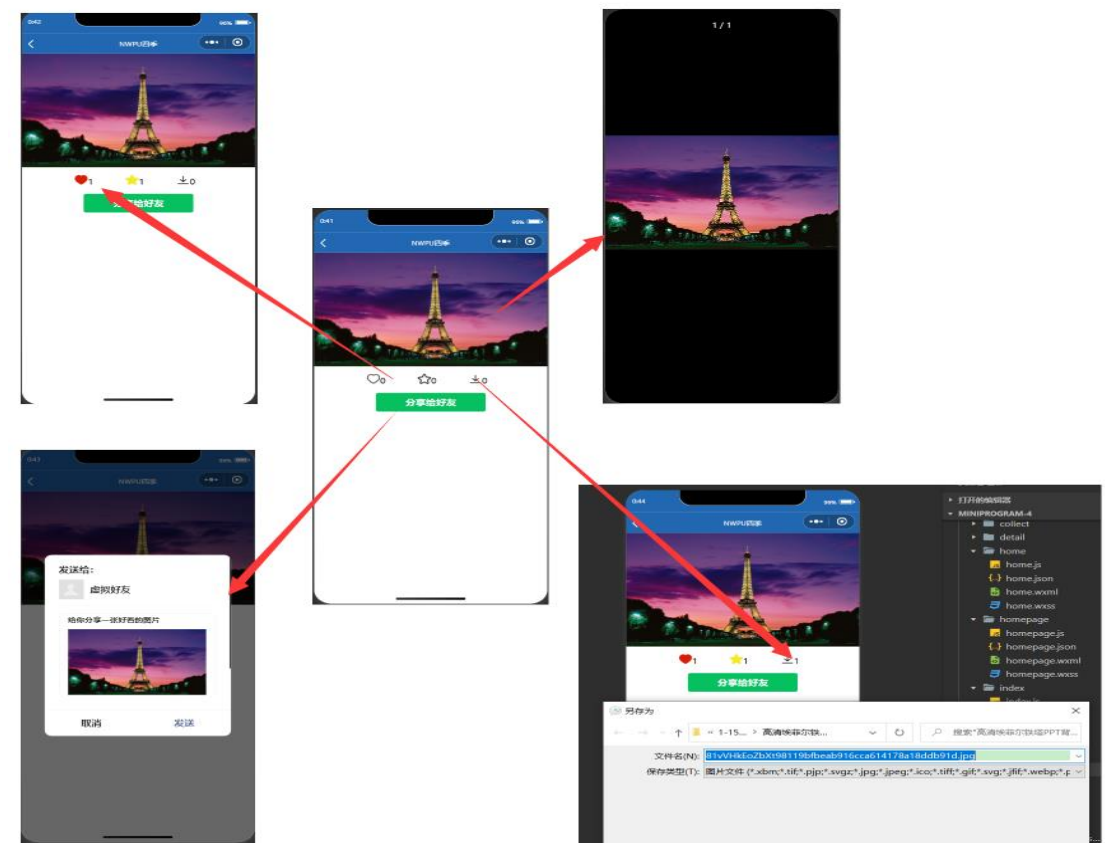


图 6.3 图片页交互图

6.2.4 上传页

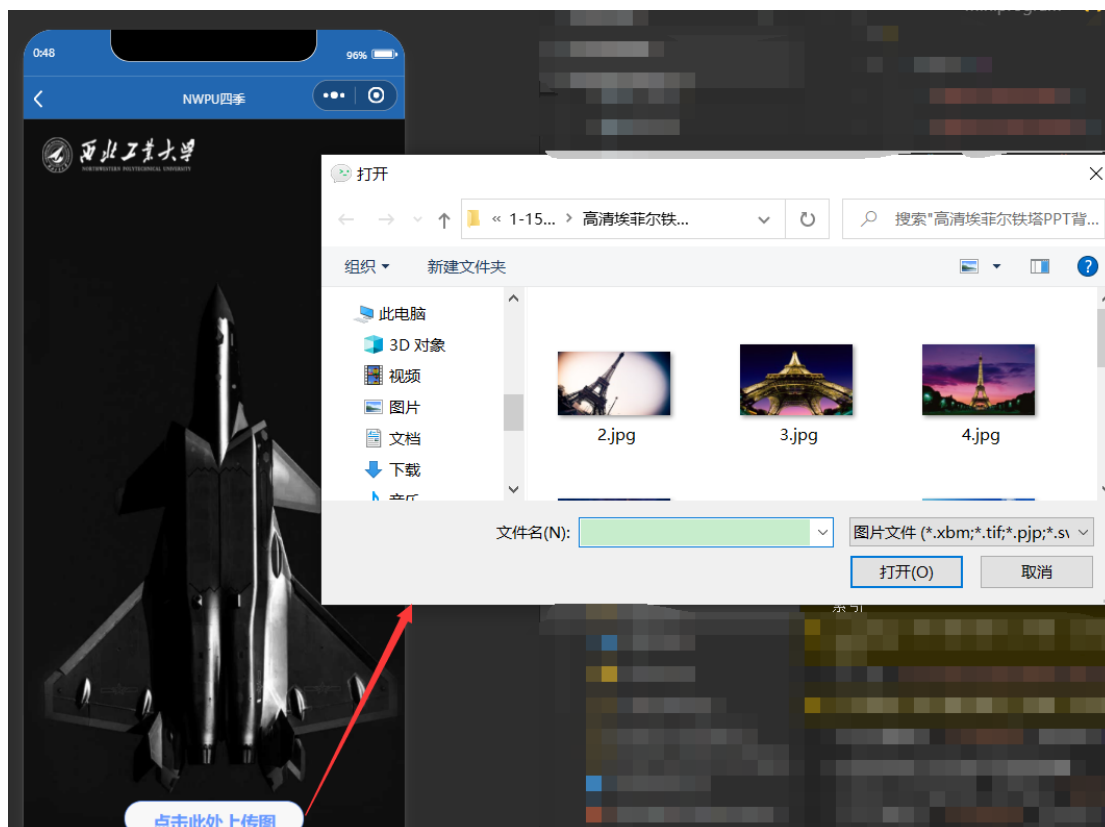


图 6.4 上传页交互图

6.2.5 其他页面

剩余页面和共享页类似，不做介绍。

七、技术方案

7.1 数据库

我总共设计了两个数据库，一个是 photos，一个是 collect
Photos 里面的字段如下图：

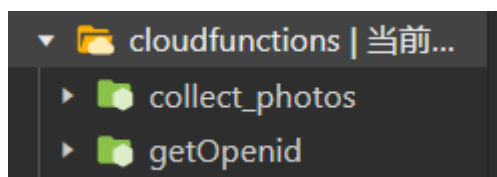
```
"_id": "17e3426e61d695540347313b58ea1f7e"  
"_openid": "  
"addDate": "2022-01-06"  
"avatarUrl": "https://thirdwx.qlogo.cn/mmopen/vi_32/JZewNRQdPurwEqJlFf..."  
"collectnum": 0  
"country": ""  
"downloadnum": 0  
"nickName": "不舍昼夜"  
"photoUrl": "cloud://mycloud-4gpztuype24a1268-6d79-myccloud-4gpztuype24..."  
"province": ""  
"upnum": 0
```

Collect 里面的字段如下图：

```
"_id": "5b049cc861d695fc02f548544fa359ea"  
"_openid": "  
"pictureid": "54ad1eea61d66e040391589d1b0ef665"
```

7.2 云函数

本程序一共使用了两个云函数，分别是 collect_photos 和 getOpenid。collect_photos 用于进行联表查询，getOpenid 用于获取用户的 openid



八、系统的运营和维护

我觉得并行性方面的效果可能会比较差。后台的表比较少,所以维护起来一个人就可以,主要是删除冗余的数据,以及很久没有被访问的数据。