

编号 201602222016022232

南京航空航天大学金城学院

# 毕 业 设 计

题 目 “丢丢妹”校园失物招领平台  
的设计与实现

学生姓名 陈家盛

学 号 2016022232

学 院 信息工程学院

专 业 计算机科学与技术

班 级 20160222

指导教师 郭慧敏 副教授

二〇二〇年五月



## 本科毕业设计（论文）诚信承诺书

本人郑重声明：所呈交的毕业设计（论文）（题目：“丢丢妹”校园失物招领平台的设计与实现）是本人在导师的指导下独立进行研究所取得的成果。尽本人所知，除了毕业设计（论文）中特别加以标注引用的内容外，本毕业设计（论文）不包含任何其他个人或集体已经发表或撰写的成果作品。

作者签名：陈家盛

2020 年 5 月 20 日

（学号）： 2016022232



# “丢丢妹”校园失物招领平台的设计与实现

## 摘 要

本次毕设，设计并实现了基于微信小程序平台、使用云开发的校园失物招领的微信小程序。小程序主要实现了用户注册并登录，发布丢失物品、分类展示物品、提供发布者联系信息等功能。此小程序是基于 JS、WXSS、JSON、WXML 等技术，还利用到了云开发模块的云函数、云数据库、云存储等功能。

本文主要分为四个部分。第一部分对系统的需求分析，主要从功能需求、可行性以及非功能需求等展开论述。第二部分大纲概要设计主要从开发环境及工具、总体结构、功能架构、数据库设计展开论述。第三部分详细设计小程序登录模块、首页展示模块、发布模块的设计与实现展开。最后是对测试途中测试到的疑难问题的解决做了总结。

**关键词：**失物招领平台，微信小程序，开发思维，模块设计



# “Diudiumei”Design and realization of campus lost and found platform

## Abstract

This graduation design, design and realize based on WeChat mini program platform, the use of cloud development of campus lost and found weChat mini program. The mini program mainly realizes the functions of registering and logging in, publishing lost items, classifying display items, providing contact information for publishers, etc. This mini program is based on JS, WXSS, JSON, WXML and other technologies, but also the use of cloud development module cloud functions, cloud database, cloud storage and other functions.

This paper is divided into four parts. The first part of the system demand analysis, mainly from the functional requirements, feasibility and non-functional requirements. The outline design of the second part mainly discusses from the development environment and tools, the overall structure, the functional architecture, and the database design. The third part of the detailed design of mini program login module, home display module, release module design and implementation. Finally, it summarizes the solution of the difficult problem stoisted during the test.

**Keywords:** lost and found platform; WeChat mini program; development thinking; module design



## 目录

|                   |    |
|-------------------|----|
| 摘 要               | i  |
| Abstract          | ii |
| 第一章 引言            | 1  |
| 1.1 开发背景及其意义      | 1  |
| 1.2 小程序优点及其缺点     | 2  |
| 1.2.1 优点如下        | 2  |
| 1.2.2 缺点如下        | 2  |
| 1.3 小程序与其他开发系统的区别 | 2  |
| 第二章 小程序基本技术简介     | 4  |
| 2.1 JS (页面逻辑)     | 4  |
| 2.2 JSON (页面配置)   | 4  |
| 2.3 WXML (页面结构)   | 4  |
| 2.4 WXSS (页面样式表)  | 5  |
| 第三章 系统各需求分析       | 6  |
| 3.1 可行性分析         | 6  |
| 3.2 功能需求分析        | 6  |
| 3.3 非功能需求分析       | 6  |
| 3.4 本章小结          | 7  |
| 第四章 失物招领小程序概要设计   | 9  |
| 4.1 开发环境及工具       | 9  |
| 4.2 总体架构设计        | 9  |
| 4.2.1 TabBar 设计   | 9  |
| 4.2.2 TaBar 详情设计  | 10 |
| 4.3 功能架构设计页面      | 11 |
| 4.4 数据库设计         | 13 |
| 4.4.1 数据库表说明      | 13 |



|                               |    |
|-------------------------------|----|
| 4.4.2 数据详细设计说明.....           | 13 |
| 4.5 本章小结.....                 | 15 |
| 第五章 失物招领小程序详细设计.....          | 16 |
| 5.1 登陆模块设计与实现.....            | 16 |
| 5.1.1 页面设计.....               | 16 |
| 5.1.2 页面实现.....               | 17 |
| 5.2 首页展示及招领模块设计与实现.....       | 21 |
| 5.2.1 页面设计.....               | 21 |
| 5.2.2 页面实现.....               | 21 |
| 5.3 发布模块设计与实现.....            | 23 |
| 5.3.1 页面设计.....               | 23 |
| 5.3.2 页面实现.....               | 25 |
| 5.4 本章小结.....                 | 26 |
| 第六章 小程序维护与演化.....             | 27 |
| 6.1 swiper 高度自适应, 改变默认高度..... | 27 |
| 6.1.1 问题分析.....               | 27 |
| 6.1.2 解决方案.....               | 27 |
| 6.2 返回上个页面并刷新.....            | 28 |
| 6.2.1 问题分析.....               | 28 |
| 6.2.2 解决方案.....               | 28 |
| 6.3 用户无法在程序内退出.....           | 29 |
| 6.3.1 问题分析.....               | 29 |
| 6.3.2 解决方案.....               | 29 |
| 6.4 用户没有登陆就可以使用发布模块.....      | 29 |
| 6.4.1 问题分析.....               | 29 |
| 6.4.2 解决方案.....               | 30 |
| 6.5 无法多账号调试.....              | 30 |
| 6.5.1 问题分析.....               | 30 |



|                 |    |
|-----------------|----|
| 6.5.2 解决方案..... | 31 |
| 第七章 总结与展望.....  | 32 |
| 7.1 总结.....     | 32 |
| 7.2 展望.....     | 32 |
| 参考文献.....       | 33 |
| 致 谢.....        | 34 |



## 第一章 引言

### 1.1 开发背景及其意义

在移动数据时代日新月异的今天,web 端已经慢慢离开公众的视野,移动端已经攻占城池,各式各样的手机,眼花缭乱的 App 给予了公众用户很大的选择。在初级开发者的世界有三门旗帜 BAT (Baidu、Alibaba、Tencent),其中腾讯开发的各大游戏以及通讯 App 已经成为大众不可或缺的存在,在移动支付中 Alibaba 的支付宝, Tencent 的微信支付都已经走在世界前端。其中微信支付的平台,一个已经走向世界的通讯服务平台,也带出了很多便利大众的服务以及新文化。例如扫一扫、朋友圈、微信支付、公众号、小程序、摇一摇等等等一系列引领中国的新理念。其中小程序平台的搭建给很多前端开发者以及用户带来崭新的文化交流。

Tencent 官方给小程序定义为:一种新的开放能力,可以在微信内被便捷地获取和传播,同时具有出色地使用体验。

基于微信小程序技术的失物招领平台,前端采用 WXML+JS+WXSS+JSON,程序内部搭载自定义组件以及云开发中的云数据库+云存储+云函数。系统角色即为用户,可读写自己发布的物品信息,可读他人发布的物品信息。

微信小程序有着出色的开发文档,是我认为最为出色的中文开发文档之一。对于开发者来说,这是一个基于国语的一个查找手册,在开发者使用过程中开发者会不断地在这本手册中查找到有适合需求地 API,以及可以实现开发者需求的框架,官方更是设计有很多基于中国人自己使用特色的组件,只有你想不到,没有官方做不到的服务。(如图 1.1 所示)



图 1.1 微信小程序开发文档





## 1.2 小程序优点及其缺点

微信现今依旧把小程序作为整体生态搭建的重中之重。小程序的优缺点也是逐渐被大家褒奖或修正。

### 1.2.1 优点如下

(1) 便捷——即用即走是微信小程序上线就搭建的地基，也正是这个原因使得小程序可以代替众多的 App，或是作为阉割版本的承载体，或是作为 App 的整体嫁接。为手机节省了众多的存储空间，无需担心存储空间或是内存空间的不足，更是为你的手机节省了电量。

(2) 无需推广——由于微信二维码的普及，扫一扫成为大众口口相传的便捷功能，微信小程序依靠微信流量，它已经完全嵌入了微信的聊天，公众号体系，完美地融合在微信之中，这使得微信小程序在未曾推广便已经获得顶级移动流量，这也在另一方面巩固了微信的成熟，更是壮大了微信团队。

(3) 相对于前端开发者来说，小程序更是一个带给前端的天降利器，微信小程序的云开发功能使得前端工作者无需了解太多负责的后端知识，开发者即能体验到全栈开发的乐趣，刨除性能不说，微信小程序无疑给开发者们带来了全新的开发体验。

(4) 前途无量——小程序总会给人无限的憧憬，相信未来小程序会给开发者和用户更多的惊喜。

### 1.2.2 缺点如下

(1) 受控——相比于 App，小程序有太多来自微信的限制，从功能接口到类别内容都受到微信的监管，部分稍微敏感一点的内容甚至会收到封禁管制。

(2) 留住用户——虽然很多数不胜数的小程序已经杀出重围，但是普遍来说，主打“即用即走”的小程序在留住用户的能力上还是和 App 有着很大的差距，或许大众都会有这样的感觉，你有可能在 App 上花上一整个下午的时光如：微博、抖音、头条等等，但是你无法在小程序内停留 10 分钟，或许正是这样的便捷使得小程序生来就有这样的缺点。

## 1.3 小程序与其他开发系统的区别

对于开发者，熟悉 web 开发的程序员们，对于微信小程序的入门显得有些轻松，小程序的四大页面 wxss、js、json、wxml 对应于 css、js、json、xml，这使得开发者几乎无需学习其他的标签语言就可以玩转小程序的开发，其中较为特别的只有 wxml，这是微信生态体系独有



的标签语言，他是结合 xml 以及 html 的组合能力，结合基础组件、事件系统即可以构建出页面的结构。

所以从整体而言开发语言层面和传统的页面差别不大，是使用 JavaScript 脚本编写逻辑代码，使用类似于 XML 的 WXML 来描述页面的结构，使用类似于 CSS 的 WXSS 来描述页面的样式。由于页面渲染以及逻辑分离的运行机制于传统网页存在差异，微信官方更是给出了小程序的开发生态一体式 IDE——微信开发者工具。开发者可以直接借助微信开发者工具完成程序的开发、编译、运行、界面、逻辑测试、真机调试、提交发布等一系列功能。

从开发的前期准备看，网页开发者只需要用到浏览器以及一些辅助的开发 IDE 即可，小程序则需要申请小程序账号，安装微信开发者工具等过程。



## 第二章 小程序基本技术简介

### 2.1 JS（页面逻辑）

小程序的主要逻辑开发语言是 js，指定页面的数据，页面生命周期、页面事件处理函数等等。所以通常小程序的开发和 web 开发时有很大的相似性的，不同点在于，网页开发渲染线程和脚本线程时互斥的，这也是为什么长时间的脚本运行可能会导致页面失去响应，而小程序中二者分开，运行在不同的线程中，开发者可以使用各种浏览器暴露出来的 DOM API，进行 DOM 的选中和操作。所以前端开发的一些库如 JQuery 等是无法在小程序中运行的并且网页开发需要面对各式各样的浏览器，PC 端的 IE、Chrome、FireFox，移动端需要面对 Safari、Chrome 以及 IOS、Android 的各式 WebView。对于复杂一些的页面 Page 构造器已经不能满足，此时的 Component 构造器应运而生，将方法放在 methods: {} 中，这种创建方式类似于自定义组件，可以同样使用 behaviors 等复杂特性。

### 2.2 JSON（页面配置）

小程序中一般有两种 JSON 文件，一种是根目录下的 APP.JSON，一种是每个页面下的“页面名”.JSON。前者用来对微信小程序进行全局配置，决定整个程序的页面路径、窗口表现情况、网络延时时间以及设置多 Tab 等等。后者是对单个页面进行配置，页面中的配置项会覆盖根目录下的 APP.JSON 的配置情况。

### 2.3 WXML（页面结构）

WXML 是小程序框架设计的一套标签语言，结合了基础组件和事件系统构建出页面结构。WXML 具有许多强大的能力，例如：进行数据绑定、操作列表和条件渲染、模板以及引用等等。

数据绑定即绑定来自对应 PAGE 的 DATA，遵循 Mustache 语法将变量包起来，可作用于内容、组件属性、控制属性、关键字、运算、组合等等。

列表渲染即使用 wx:for、block wx:for、wx:key 等控制属性绑定数组，然后即可使用各项数据重复渲染。

条件渲染即使用 wx:if、block wx:if 来判断是否需要对某程序段进行渲染。

模板即为 WXML 提供的 template，可以在其中定义程序段，而后在不同地方调用。



引用即为 WXML 提供的两种方式 import 和 include。Import 可在某文件中使用其定义的 template，其中 import 的作用域为 import 文件中已经定义的 template。而 include 相当于是将整个代码拷贝到 include 位置。

## 2.4 WXSS（页面样式表）

WXSS 和 CSS 相比，类似于 JAVA 中的继承关系，WXSS 具有 CSS 的大部分特性，并且进行了扩充和整改（如：扩展了尺寸单位、样式导入等等），以便于适应微信小程序的开发。

（如图 2.1 所示）WXSS 目前支持的选择器有 id 选择器、标签选择器、类与伪类选择器、data 属性选择器等等常用 CSS3 的选择器。

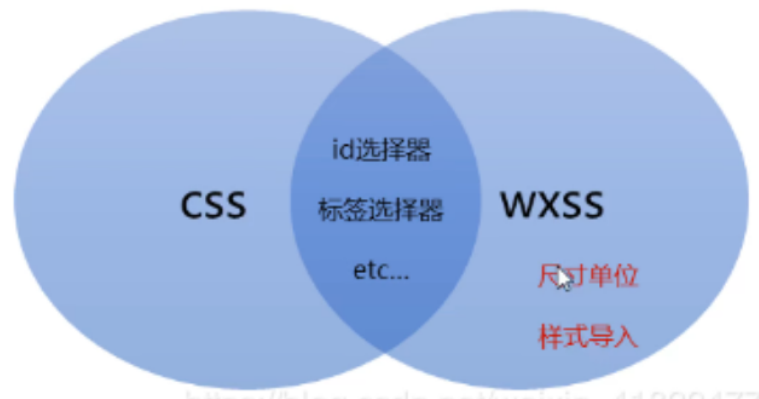


图 2.1 WXSS 与 CSS 的关系



## 第三章 系统各需求分析

### 3.1 可行性分析

基于微信小程序的发展，几乎所有成熟的 APP 都已经或多或少在微信小程序上做了完整或者阉割的嫁接工作，所以现在在开发基于微信小程序的失物招领平台一定是可行的。再者微信对于学生或者一个区域的社区工作者一定是不可或缺的，在微信平台上基于“即用即走”理念开发的小程序更是会收到大众的使用。

拿笔者在 IKEA 宜家、迪卡侬或者 H&M 购物的体验来看，在需要会员支付的时候扫一下二维码即可打开小程序用户页面，甚至可以直接扣款，与微信支付的功能融为一体。从此方面考虑，失物招领平台在校园的普及也几乎是可以看见未来的。甚至可以在小程序中发布赏金，以及添加与失主对话等等直接性的操作。

### 3.2 功能需求分析

功能需求是软件需求中最重要、最主要、最常见的，也是最为复杂的需求，他是一个软件产品能够解决用户问题和长生价值的基础。此软件的功能需求有：

- R1：系统应允许用户查看寻找丢失的物品。
- R2：系统应将所有物品集中并分类展示。
- R3：系统应允许发布者招领其发布的物品，且其他用户只用可读的权限。
- R4：系统应允许用户发布丢失的物品信息。
- R5：系统应允许用户更改用户信息，做到用户自由原则。
- R6：系统应允许已登陆的用户查看发布者信息，以便招领失物。
- R7：系统应帮助用户快速注册并登陆
- R8：系统应允许用户实现退出登陆操作

### 3.3 非功能需求分析

软件需求除了功能需求外，其他皆被划分为非功能需求，包括性能需求、质量属性、对外接口、约束.....

软件需求中质量属性对系统成败的影响极大，因此在某些情况下，非功能需求又被用来特指质量属性。



## 1. 性能需求

- (1) Speed（速度）：每个用户的操作必须都在 5 秒内完成。
- (2) Capacity（容量）：系统应该能够存储至少 1 万条物品信息
- (3) Throughput（吞吐量）：系统至少完成 100 条/分钟正确的解析
- (4) Load（负载）：系统应允许 50 个用户同时从服务器中进行数据的上传下载
- (5) time-critical（实时性）：在数据更新后系统必须帮助用户进行数据的实时更改

## 2. 质量属性

- (1) Reliability（可靠性）：在客户端与服务器前端通信时，若网络故障，系统不能出现故障。
- (2) Availability（可用性）：系统的可用性要达到 95%。
- (3) Security（安全性）：用户能看到所有发布者发布的信息，但是只有发布者可以更改或删除该信息。
- (4) Maintainability（可维护性）：若系统要增加新的模块，要能够在一周内完成。
- (5) Portability（可移植性）：系统能够在不同的操作系统中运行或移植。
- (6) Usability（易用性）：使用系统一日后，所有用户对程序的使用效率必须达到 5 分钟内解决完成一个需求。

## 3. 约束

- (1) 系统要使用 JS 语言开发
- (2) 系统要在 IOS、和 Android 同时可以使用

## 4. 数据需求

- (1) 系统需要存储所有用户发布的物品信息
- (2) 系统存储的数据必须带有图片以及分类

## 3.4 本章小结

需求是用户的一种期望，软件系统通过满足用户的期望来解决用户的问题，IEEE 对需求的定义为[IEEE610.12-1990]：

- 1) 用户为了解决问题或达到某些目标所需要的条件或能力；
- 2) 系统或系统部件为了满足合同、标准、规范或其他正式文档所规定的要求而需要具备的条件或能力；



3) 对 1)或 2)中的一个或一种能力的一种文档化的表述

在系统需求分析准备完毕之后便可以开始着手系统的概要设计，数据库设计等。在所有需求中最重要的是功能需求。





## 第四章 失物招领小程序概要设计

### 4.1 开发环境及工具

使用微信小程序自带微信开发者工具，原生开发一体式完备搭建。开发者可以直接借助微信开发者工具完成程序的开发、编译、运行、界面、逻辑测试、真机调试、提交发布等一系列功能。

IDE 为开发者提供了强大的模拟器，编辑器，调试器以及强大的利器云开发。开发者可以在此 IDE 中直接进行多账户调试，以及各种屏幕的适应度测试。

### 4.2 总体架构设计

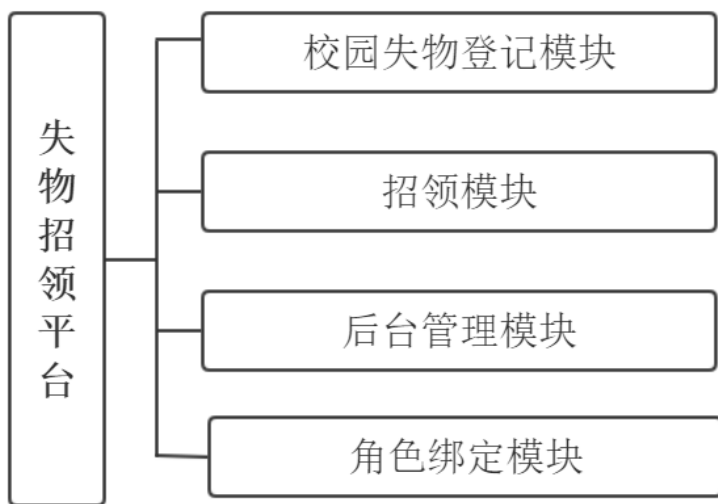


图 4.1 失物招领平台总体架构

#### 4.2.1 TabBar 设计

全局配置设置三个 TabBar 页面：“发现”、“发布”、“我的”。（如图 4.2 所示）所有的 TabBar 都需要在全局配置 App.json 文件下布置好，其中配置的具体含义有：“Pagepath”：TabBar 页面路径、“Text”：页面名、“IconPath”：页面未选中时系统显示的图片、“SelectedIconPath”：页面选中时系统显示的图片、“Color”：未选中时系统显示的字体颜色、“SelectedColor”：选中后





系统显示的字体颜色.....



图 4.2 TaBar 设计



图 4.3 发现页面

#### 4.2.2 TaBar 详情设计

“发现”页面中设置四个 Swiper-tab 列表：“卡片证件”、“电子产品”、“包”、“其他”。其中每个 Swiper-tab 中皆列出该分类中的总数据，其所有数据存在微信小程序的云开发数据库中。（如图 4.3 所示）



图 4.4 发布页面



图 4.5 我的页面

“发布”页面，使用者可以在此页面发布丢失物品，修改物品信息，以及后续根据需求继续开发的操作。（如图 4.4 所示）

“我的”页面，可以显示使用者信息，修改用户信息，查看帮助，查看关于以及退出登陆的操作。（如图 4.5 所示）

### 4.3 功能架构设计页面

云开发是腾讯官方专门为了小程序开发而设，开发者使用云开发可以无需搭建服务器，使用完整的原生云端支持和微信服务支持，弱化后端的功能以及运维的概念，平台提供的 API 进行核心业务开发，即可实现快速上线和迭代需求。

云开发一总共提供了几大技术支持：

表 4.6 小程序云开发

| 技术支持 | 优点      | 解释                              |
|------|---------|---------------------------------|
| 云函数  | 代替服务器搭建 | 云函数是在云端运行的代码，开发人员只需要编写本程序业务逻辑代码 |



表 4.6 小程序云开发（续）

| 技术支持 | 优点            | 解释  |
|------|---------------|---|
| 数据库  | 代替数据库         | 一个既可在小程序前端操作，也能在云函数中读写的 JSON 数据库，免费 2G 容量   |
| 存储   | 代替自建存储功能和 CDN | 在小程序前端直接上传/下载云端文件，在云开发控制台可视化管<br>理，免费 5G 容量 |
| 云调用  | 小程序原生微信服务集成   | 基于云函数免鉴权使用小程序开放接口的能力，包括服务端调用、<br>获取开放数据等能力  |

云开发（如图 4.6 所示）也是前端开发者的一个福利，可以免去后端庞大的技术难关。开发者使用 Appid 新建云开发模板确认后即可看见开发者工具中的小程序工程目录中多了名为 cloudfunctions 的文件夹，这个就是用来存放云函数的文件夹。

点开顶部云函数，设置环境名称（env）后，系统会自动分配给开发者一个环境，此时的云开发控制台都是空的，然后对开发环境中的 env 名称进行更改搭建。Env 参数会决定接下来开发者发起的云开发的调用（wx.cloud.xxxx）会请求到哪一个云环境中，所以要看清楚使用的是默认环境还是需要自己搭建的新环境。

其中 TraceUser 为 true 则在云开发的用户访问中可以看到调用资源的用户或者开发者。

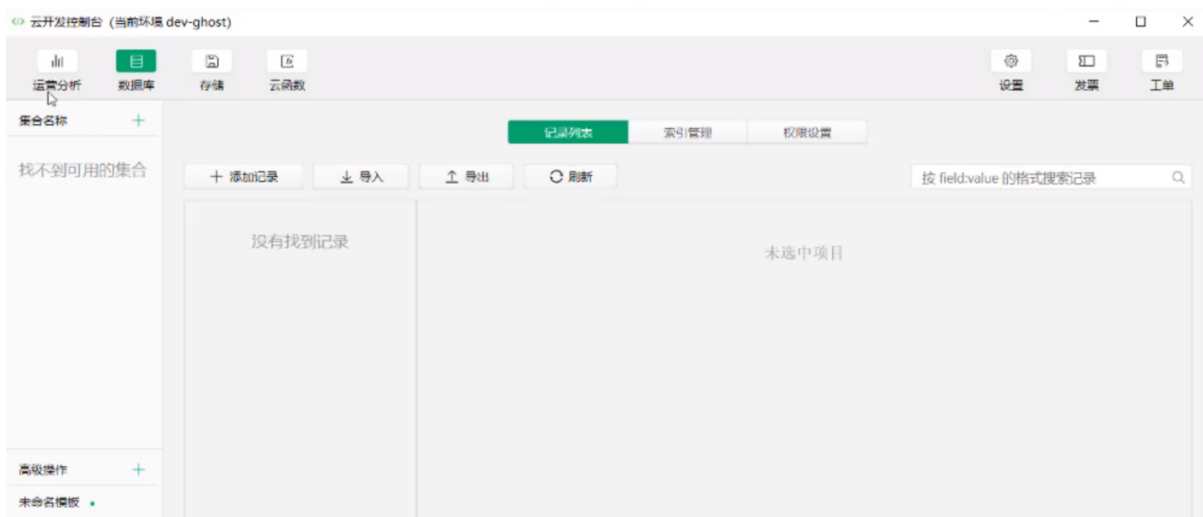


图 4.6 云开发界面



## 4.4 数据库设计

### 4.4.1 数据库表说明

数据库设计是使用云开发控制台中的云数据库功能。云数据库设计比普通 web 开发有着令人羡慕的便捷感受。本程序主要建立 5 个数据表，分别存放用户数据、卡片证件的失物、包的失物以及其他的失物数据。

用户数据：用户 id、登陆者 openid、热度、用户姓名、手机号码、用户头像、注册时间、微信账号号码。

卡片证件列表：物品 id、发布者 openid、详细描述、照片信息、物品名称、发布者手机号码、发布者姓名、发布者微信号码。

电子产品列表：物品 id、发布者 openid、详细描述、照片信息、物品名称、发布者手机号码、发布者姓名、发布者微信号码。

包列表：物品 id、发布者 openid、详细描述、照片信息、物品名称、发布者手机号码、发布者姓名、发布者微信号码。

其他物品列表：物品 id、发布者 openid、详细描述、照片信息、物品名称、发布者手机号码、发布者姓名、发布者微信账号号码。

### 4.4.2 数据详细设计说明

小程序主要使用五个表，分别是：users（用户数据表）、0（卡片证件列表）、1（电子产品列表）、2（包列表）、3（其他物品列表）。（如表 4.1 所示）

表 4.2 系统架构表

| 表名    | 说明     | 作用               |
|-------|--------|------------------|
| users | 用户数据表  | 存放首次登陆的用户信息      |
| 0     | 卡片证件列表 | 存放用户发布的卡片证件列表的信息 |
| 1     | 电子产品列表 | 存放用户发布的电子产品列表的信息 |
| 2     | 包列表    | 存放用户发布的包列表的信息    |
| 3     | 其他物品列表 | 存放用户发布的其他物品列表的信息 |

#### (1) Users（用户数据表）

用于记录此失物招领系统注册用户的信息，包括系统编号、发布者 openid、发布者热度、



发布者姓名、发布者手机号码、发布者头像链接、发布者微信号码。（如表 4.3 所示）

表 4.3 users——用户数据表

| 字段名称         | 说明         | 类型     | 是否为主键 | 是否默认空值 |
|--------------|------------|--------|-------|--------|
| _id          | 系统编号       | String | 是     | 否      |
| _openid      | 发布者 openid | String | 否     | 否      |
| links        | 发布者热度      | String | 否     | 是      |
| name         | 发布者姓名      | String | 否     | 是      |
| phoneNumber  | 发布者手机号码    | String | 否     | 是      |
| src          | 发布者头像链接    | String | 否     | 是      |
| time         | 发布者注册时间    | String | 否     | 是      |
| wechatNumber | 发布者微信号码    | String | 否     | 是      |

## (2) 0（卡片证件列表）

用于记录此失物招领系统用户发布的有关卡片证件分类的物品信息，包括系统编号、发布者 openid、物品详细描述、图片链接、物品名称、发布者手机号码、发布者微信号码、发布者姓名。（如表 4.4 所示）

表 4.4 物品通用数据表

| 字段名称         | 说明         | 类型     | 是否为主键 | 是否默认空值 |
|--------------|------------|--------|-------|--------|
| _id          | 系统编号       | String | 是     | 否      |
| _openid      | 发布者 openid | String | 否     | 否      |
| describe     | 物品详细描述     | String | 否     | 是      |
| imgurl       | 图片链接       | String | 否     | 是      |
| name         | 物品名称       | String | 否     | 是      |
| phoneNumber  | 发布者手机号码    | String | 否     | 是      |
| wechatNumber | 发布者微信号码    | String | 否     | 是      |
| username     | 发布者姓名      | String | 否     | 是      |

## (3) 1（电子产品列表）

用于记录此失物招领系统用户发布的有关卡片证件分类的物品信息，包括系统编号、发



布者 openid、物品详细描述、图片链接、物品名称、发布者手机号码、发布者微信号码、发布者姓名。（如表 4.4 所示）

(4) 2（包列表）

用于记录此失物招领系统用户发布的有关卡片证件分类的物品信息，包括系统编号、发布者 openid、物品详细描述、图片链接、物品名称、发布者手机号码、发布者微信号码、发布者姓名。（如表 4.4 所示）

(5) 3（其他物品列表）

用于记录此失物招领系统用户发布的有关卡片证件分类的物品信息，包括系统编号、发布者 openid、物品详细描述、图片链接、物品名称、发布者手机号码、发布者微信号码、发布者姓名。（如表 4.4 所示）

## 4.5 本章小结

小程序由于省去了大量的后端搭建，整体架构的搭建显得十分轻松便捷，在搭建的过程中开发者可以实时地在 IDE 中查看系统地搭建状况，这是小程序十分便捷的一个优点。但是失去了后端强大的技术支持，在开发者测试过程中，大量的数据滚动时，会有明显的不适感，所以官方也是允许开发者使用一些优质的后端技术来改善使用。



## 第五章 失物招领小程序详细设计

### 5.1 登陆模块设计与实现

#### 5.1.1 页面设计

小程序的登陆有两种情况：

- (1) 从来没有登陆过此程序，系统中没有该用户的记录（）
- (2) 登陆过此程序，系统中含有该用户信息

当用户处于第一中情况的时候，页面只能显示登陆的按钮，且头像自动设置为默认，点击登陆按钮后系统提交请求，请求用户使用登陆权限，获得用户的基本信息并存储在数据库中。



图 5.1 未登录的情况



图 5.2 已登录过的情况

当用户处于第二种情况的时候，页面可以显示该用户信息，且有修改用户信息，查看用户物品以及退出登陆的权限，由于系统中已经含有用户信息，用户无需点击登陆，设计接口直接调用 users 数据进行登陆。



### 5.1.2 页面实现

登陆模块通过小程序自带的 userInfo 存储登陆信息，与以往手机或者个人注册流程优化很多。

UserInfo 中有大量从微信获取的个人信息，十分方便快捷：如 AvatarUrl、City（用户所在城市）、Country（用户所在国家）、NickName（用户微信名）、Province（用户所在省份）等可以直接调取无需注册，也是十分的便捷。在设置用户信息的时候可以额外添加需要的属性，开发者在这里调取了用户\_id、\_openid（用户唯一标识）、name（用户昵称）、src（用户头像），并手动设置 links、phonename（用户手机号码）、time（用户登记时间）、wechatNumber（用户微信号码）。这一系列信息为后面获取整体的信息提供便利，在用户登陆后也可以实现更改信息等等操作。

在 index.wxml 的登陆模块中设置 bindgetuserinfo 控件，在 index.js 文件中获取 getMyInfo 信息，存入数据库对应表中。并把数据权限设置为所有用户可读，仅创建者可读写。（如图 5.3 所示）



图 5.3 数据库权限设置

使用 node.js 下的后端程序，实现已经登陆过的用户自动登陆的需求，由于云开发，小程序可以无需搭建一个后端的环境，调用官方已经提供的 login 函数，部署在 cloudFunction 上，适配开发者搭建好的环境 ID。（如图 5.4 所示）





```
1 // 云函数模板
2 // 部署: 在 cloud-functions/login 文件夹右击选择 “上传并部署”
3
4 const cloud = require('wx-server-sdk')
5
6 // 初始化 cloud
7 cloud.init({
8   // API 调用都保持和云函数当前所在环境一致
9   env: cloud.DYNAMIC_CURRENT_ENV
10 })
11
12 /**
13  * 这个示例将经自动鉴权过的小程序用户 openid 返回给小程序端
14  *
15  * event 参数包含小程序端调用传入的 data
16  *
17  */
18 exports.main = (event, context) => {
19   console.log(event)
20   console.log(context)
21
22   // 可执行其他自定义逻辑
23   // console.log 的内容可以在云开发云函数调用日志查看
24
25   // 获取 WX Context (微信调用上下文), 包括 OPENID、APPID、及 UNIONID (需满足 UNIONID 获取条件) 等信息
26   const wxContext = cloud.getWXContext()
27
28   return {
29     event,
30     openid: wxContext.OPENID,
31     appid: wxContext.APPID,
32     unionid: wxContext.UNIONID,
33     env: wxContext.ENV,
34   }
35 }
36
```

图 5.4 云开发 login 函数

在首页 index.js 中的生命周期函数——监听页面 onReady: function 调用 wx.cloud.callFunction, name 为 login。在数据库中查询 users 的 \_openid 与所使用的用户的 openid 相比较, 若存在相匹配的信息则自动登陆。使用 if-else 实现若 \_openid 不存在或者用户尚未登陆过则设置 disabled 为 false, 禁用自动登陆权限。



```
20  * 生命周期函数--监听页面初次渲染完成
21  */
22  onReady: function () {
23    wx.cloud.callFunction({
24      name: 'login',
25      data: {}
26    }).then((res) => {
27      // console.log(res.result.openid);
28      db.collection('users').where({
29        _openid : res.result.openid
30      }).get().then((res)=>{
31        //如果没有登陆过则设置disabled为false
32        if(res.data.length){
33          // console.log(res.data[0]);
34          // app.globalData.userinfo = Object.assign(app.globalData.userinfo , res.data[0]);
35          app.globalData.userinfo = res.data[0];
36          // console.log(app.globalData.userinfo);
37
38          this.setData({
39            src: app.globalData.userinfo.src,
40            name: app.globalData.userinfo.name,
41            logged: true
42          })
43        }else{
44          this.setData({
45            disabled: false
46          });
47          console.log("亲先登录")
48        }
49      });
50    });
51  });
52  },
```

图 5.5 自动登陆代码

修改个人信息部分提供了修改昵称，手机号，微信号的操作。页面设置简单，三个 input 控件以及一个 button 控件，此处与发布物品信息不同，这里使用 bindinput 控件，可以直接在输入的时候读取到 input 控件中的信息，方便系统存取。Button 控件调用 updateinformation() 函数，此功能及时更新数据库‘users’中对应的用户信息。（如图 5.6 所示）



图 5.6 完善信息界面

此处使用三个 Handle（name、phoneNumber、wechatNmaber）获取各个 input 中信息，获取事件的 value 值后，使用 setData（）方法设置参数，最后直接传到 handleall 中便于数据库的更新。（如图 5.7 所示）

```
99   handleall(){
100     this.updateinformation()
101   },
102
103   updateinformation(){
104
105     wx.showLoading({
106       title: '更新中',
107     })
108     db.collection('users').doc(app.globalData.userinfo._id).update({
109       data : {
110         name: this.data.name,
111         phoneNumber: this.data.phoneNumber,
112         wechatNumber: this.data.wechatNumber,
113       }
114     }).then((res)=>{
115       wx.hideLoading();
116       wx.showToast({
117         title: '更新成功! ',
118       })
119     })
120   }
121 }
```

图 5.7 handleall 函数



## 5.2 首页展示及招领模块设计与实现

### 5.2.1 页面设计

首页展示时本程序最大的工作量所在，首先在 TabBar 页面的发现页中搭建四个 switch-tab，在每一个 TAB 中展示该类数据（包括图片、名称、描述以及发布者），并且点击该数据的时候需要读取其详细信息（包括物品名称、描述、发布者、发布者手机号码、发布者微信号码）方便失主联系发布者。

系统应允许用户在登陆状态时触发招领模块，可以招领自己发布的物品信息。

### 5.2.2 页面实现

首页展示页面主要可以分为两个部分——switch-tab 模块、展示模块。

switch-tab 模块：在 WXML 页面（如图 5.8 所示）直接调用 view，其中的 class 需要设置为（swiper-tab-list {{currentTab==0?'on':''}}），并且需设置对应的 data-current。

展示模块：在逻辑页面（如图 5.9 所示）中调用 wx.cloud.database().collection()调取对应数据库的数据，开发者可以选择 orderBy()函数，实现数据的排序，这里实现按照发布时间的排序，获取到数据后传入到对应的局部变量中去。在 wxml 页面中获取到 js 获取的数据，并且设置 key 值为\_id。

```
62 <swiper-item>
63
64 <view wx:for="{{bag}}" wx:key="_id">
65 <navigator url="../detailrelease/detail2?id={{item._id}}">
66 <image src="{{item.imgurl}}" mode="aspectFill" class="image1"></image>
67 <label >{{item.name}}</label>
68 <label >{{item.describe}}</label>
69 <label >发布者: {{item.username}}</label>
70 </navigator>
71 </view>
72 </swiper-item>
```

图 5.8 WXML 页面



```
69 db.collection("3").orderBy(this.data.current, 'desc').get().then(res => {
70   this.setData({
71     product: res.data
72   })
73 });
74
75 db.collection("2").orderBy(this.data.current, 'desc').get().then(res => {
76   this.setData({
77     bag: res.data
78   })
79 });
80
81 db.collection("0").orderBy(this.data.current, 'desc').get().then(res => {
82   this.setData({
83     card: res.data
84   });
85
86   db.collection("1").orderBy(this.data.current, 'desc').get().then(res => {
87     this.setData({
88       ele: res.data
89     })
90   });
91 });
```

图 5.9 JS 逻辑页面

页面详情采用的是跳转页面，此时的跳转时需要附带该数据所在表中的 id 信息，有序每张表中的 id 信息都是唯一的，所以新页面的调取。新页面中同样设置了 getApp()以及 wx.cloud.database()操作，在读取云开发数据库对应表数据后，传值给局部变量 product[]中，并且在 wxml 页面中读取展示。此数据传来的是该表中所有的信息数据，在此处开发者只抽取部分信息展示，重要数据进行隐藏。此处的信息有品名称、描述、发布者、发布者手机号码、发布者微信号码。其中手机号码设置了自定义拨打组件（如图 5.10 所示），微信号码同样设置了复制功能的组件。（如图 5.11 所示）



图 5.10 自定义拨打电话组件

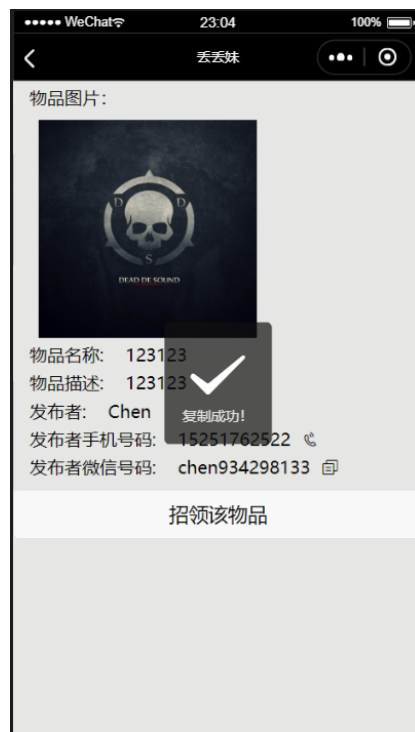


图 5.11 自定义复制数据组件

招领模块是隐藏在展示的页面之中的，这里效仿微信朋友圈以及微博的发送方式，在展示的页面实现招领，在招领之后，云开发数据库里会自动删除该发布者发布的物品信息，并且这个操作只有发布者可以点击，若非发布者系统则自动隐藏起来。

由于招领功能只能由发布者点击，非发布者无法发现，所以在发布信息的时候，开发者同时上传了发布者的 openid，openid 是每一个登陆的用户的唯一标识，所以在检验是否为发布者来说是最适合的检验方式。此处在此 button 组件中添加 wx:if 操作，设置局部变量——own，js 页面中的 own 默认值是为 false，代表此条信息非登录者所发布，在 js 页面 ONLOAD()方法中，调用 if 函数将全局变量 app.globalData.userinfo.\_openid 与局部变量的 data.openid 作比较，若唯一标识的 openid 相同则设置局部变量 own 为 true，此时详情页面则可以展示出招领该物品的 button。

点击招领后系统会将页面直接跳转到 tabbar 的发现页面，此时若信息没有进行更新，不用担心，用户可以下拉刷新，则系统会重新获取云开发数据库的信息，也是十分的快捷。

## 5.3 发布模块设计与实现

### 5.3.1 页面设计

首先发布页面是只有已经登录，且由发布权限的用户才可以看见，（如图 5.13 所示）若



用户没有登陆或者用户退出登陆，则发布模块会显示——请先登陆！（如图 5.12 所示）



图 5.12 未登录的情况



图 5.13 已登录的情况

若用户已经登陆则可以显示全部模块，（如图 5.14 所示）点击发布物品模块后，系统则跳到发布页面。发布页面主要设置三个控件获取用户输入——picker（获取分类）、input（获取物品名称）、textarea（获取物品详细信息）。



图 5.14 发布物品获取页面



5.15 获取物品信息

### 5.3.2 页面实现

使用三个控件获取分类、物品信息、以及联系方式的信息，并使用 image 组件在上传页面展示出来，上传完成后通过 navigator 使用 switchTab 回到 tabBar 首页。

首先在 wxml 页面设置好控件获取类别信息，js 页面会获得相应的云开发数据库的表中，其次在将获取的文字信息传入局部信息对应的载体中，如物品名称传入 name 中、详细信息传入 describe 中，图片信息传入 imgurl 中。值得一提的是云开发提供免费的存储空间，开发者可以减图片上传到云存储中，将图片的 url 地址传如数据库中，读取信息时读取数据库中的 url 地址，系统则会以图片形式展出。（如图 5.15 所示）

上传完毕后发布页面可以预览发布信息，由于发布信息不可更改，所以提醒发布者仔细审查，后期如若有错只能当作招领处理，重新发布。

最后，值得一提的是发布者会在发布页面将发布者信息发布出去，此处仅显示发布者姓名、发布者手机号码、发布者微信号码，而发布者的 openid 将被隐藏起来，用户无需担心信息泄漏问题。此处为了展示页面的招领模块做准备数据，所以云数据库中的数据量还是很大的。





## 5.4 本章小结

小程序的基本需求及框架已经完成，通过微信开发者工具直观的编译效果，小程序在前端样式设计（CSS）、逻辑界面设计（JS）、页面结构（WXML）、页面配置（JSON）、小程序逻辑结构（APP.JS）、小程序公共配置（APP.JSON）、小程序公共样式表（APP.WXSS）的搭建与测试效果都十分的快捷。小程序的云开发功能更是给开发者带来了许多便利之处，在腾讯官方的带领下，云开发也在不断地进行优化，希望可以带给我们更多的惊喜！



## 第六章 小程序维护与演化

### 6.1 swiper 高度自适应，改变默认高度

#### 6.1.1 问题分析

由于 swiper 组件的宽高不能限定死，写成固定值会导致其他数据占式不完全，不写或者设置成 100% 或者设置成 auto 皆非妥当处理，无法实现滚动效果。还有就是用高度乘以数量，也就是所说的获取数据数组长度，根据数据长度来动态改变每页的长度。而每个设备显示的字号或者分辨率的不同会导致显示出现 bug。（如图 6.1 所示）



图 6.1 小程序无法翻页图



6.2 小程序实现翻页效果

#### 6.1.2 解决方案

使用 `getSystemInfo` 获取系统信息，得到屏幕宽高，设置 `style` 为 `height: {{winHeight-31}}px`。即设置成自适应，即获取系统的屏幕大小，同时在整个 `view` 外面设置 `scroll-view` 控件，方向为 `scroll-y`。（实现效果如图 6.2 所示）



## 6.2 返回上个页面并刷新

### 6.2.1 问题分析

在返回上一个页面，并且刷新页面，此需求很常见，但是不解决会导致页面信息滞后，影响用户与程序的交互体验。用户使用小程序的时候难免会有提交数据后需要更新页面的需求，信息滞后会致页面停留造成用户使用体验极差的后果，并且存在非常大的性能隐患，如果用户在操作的是一个已经被删除或者是从来都没有的数据，如果开发者没有处理好则会是系统整个处于崩溃的状态。

### 6.2.2 解决方案

微信小程序的开发中，每个页面的 js 页面官方都会自动生成 `onLoad: Function(option)`、`onReady: Function()`，这些事小程序的生命周期函数——`onLoad` 是监听页面加载功能，`onReady` 是监听页面初次渲染完成功能。一般的加载信息或是配置数据都会在这两个页面中完成，此时开发者需要做的是调用 `onpulldownrefresh: function()`，这是官方给出的下拉刷新的组件，开发者需要在 js 页面以及 json 页面（“enablePullDownRefreash”: true）中配置好，并且在此控件中再次调用 `onLoad: Function(option)`、`onReady: Function()` 即可。（如图 6.3 所示）

```
117  /**
118   * 页面下拉加载
119   *
120   */
121   onPullDownRefresh: function () {
122
123     wx.showNavigationBarLoading() //在标题栏中显示加载
124
125     setTimeout(function () {
126
127       wx.hideNavigationBarLoading() //完成停止加载
128
129       wx.stopPullDownRefresh() //停止下拉刷新
130
131     }, 1500);
132
133     this.onLoad();
134     this.onReady();
135
136     this.setData({
137       logged: getApp().globalData.logged,
138     });
139
140   },
141
142 })
```

图 6.3 刷新实现代码



## 6.3 用户无法在程序内退出

### 6.3.1 问题分析

小程序用户在使用时如若担心信息泄露以及处于保守起见需退出登陆，在程序内部无法进行退出操作。

### 6.3.2 解决方案

Wxml 页面增加退出控件，并配置 bindtap，在 js 页面将全局变量 app.globalData.userInfo、app.globalData.name、app.globalData.src、app.globalData.logged 接设置为 NULL，同时页面参数 name、src、logged 重新定义。

若用户需要选择自动登陆可以在我的页面直接下拉刷新，开发者在 onpulldownrefresh: function()调用了自动登陆的操作。（如图 6.4 所示）



图 6.4 退出登陆效果图

## 6.4 用户没有登陆就可以使用发布模块

### 6.4.1 问题分析

用户在直接进入小程序时若没有登陆直接使用发布物品模块也可发布成功，这回导致程序数据控制混乱，页面展示信息不对称等错误。



## 6.4.2 解决方案

在 tabBar 我的登陆模块中设置 logged 的全局变量（默认为 false），存储在 app.js 中，若用户已登录，则获取到当前用户的 openid，且设置 logged 为 true。

在 tabBar 发布中的 js 文件中调用全局变量 logged，wxml 页面在 navigator 组件中调用 wx:id('{{! logged}}') 若 logged 为 false, 代表没有用户登陆, 则不显示发布物品, 显示 text 控件, 内容为——请先登陆！（如图 6.5 所示）



图 6.5 用户未登录情况图



6.6 用户已登陆情况

在用户至我的页面登陆之后返回该页面需要下来刷新，更新数据后则显示发布模块。（如图 6.6 所示）

## 6.5 无法多账号调试

### 6.5.1 问题分析

加入账号调试时，系统报错（如图 6.7 所示）登入账号无法完成调试，原因是该账户不是此小程序，小程序开发者。

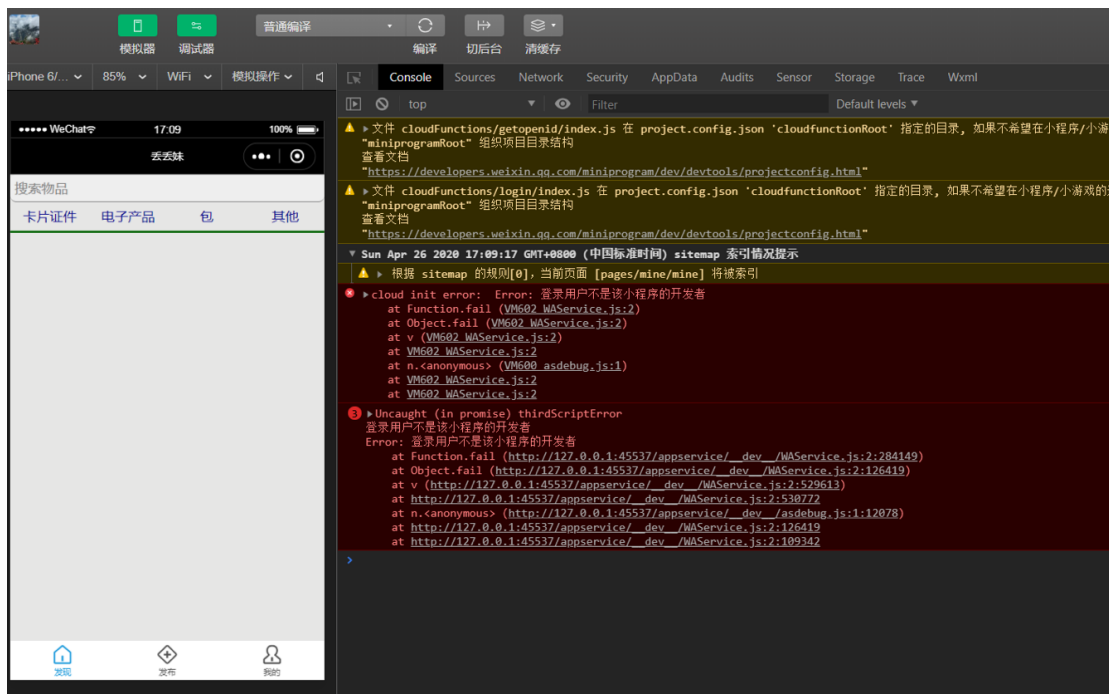


图 6.7 系统多账户调试报错

## 6.5.2 解决方案

在微信小程序后台将该微信账号添加进项目成员名单，并且添加运营者权限、开发者权限、数据分析者（基础分析）权限。

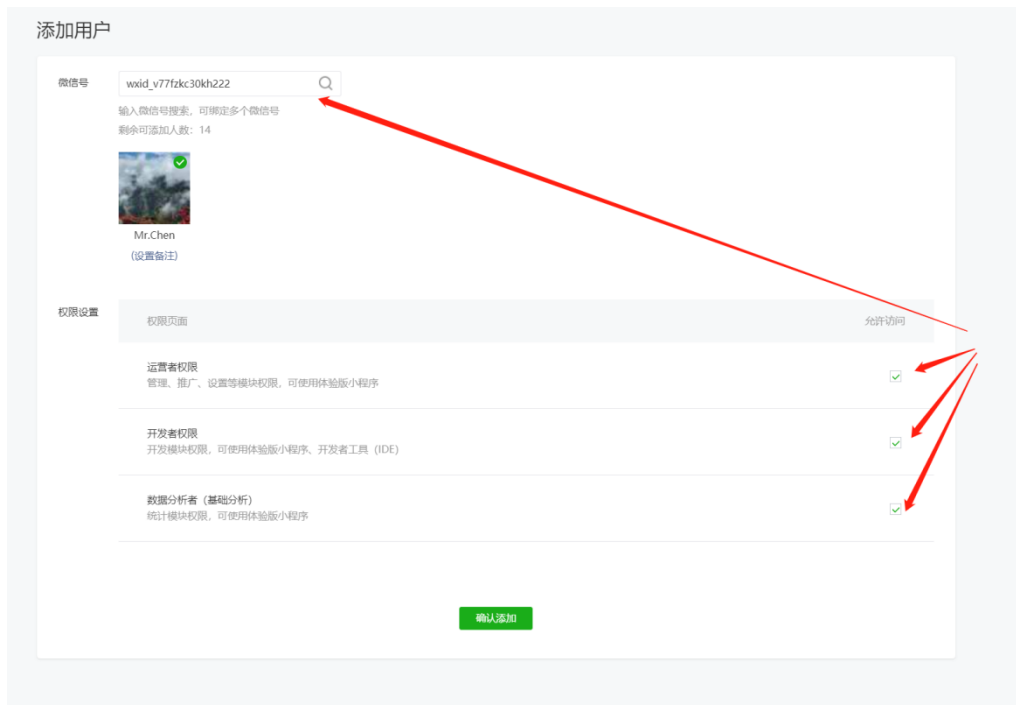


图 6.8 多账户调试解决方案



## 第七章 总结与展望

### 7.1 总结

在微信官方的带领下，小程序已经逐渐成熟起来。从前端样式的设计，到逻辑页面的处理，再到云存储云数据库的搭建，小程序给予了开发者和用户极大的快乐。本文主要完成了以下任务：

- (1) 介绍了小程序开发与传统 WEB 开发的区别及优缺点
- (2) 根据官方文档开发出便捷校园局部使用的失物招领平台
- (3) 对系统的各个模块的功能以及实现做了概述以及详情描述
- (4) 对维护及测试过程的疑难问题等做了简单的汇总

通过此次毕业设计，我也认识到很多需求都有无数种实现方式，好的开发者能够迅速在这些实现方式中找到最好的，或者说较好的解决方式。所以在程序员的成长之路中，最基础的数据结构以及算法是程序员们必不可少的必修课，这会帮助我们开发出时间复杂度、控件复杂度最小的，最易维护的代码。编程的语言是优美的，同时我再次回顾了 JAVA 基础、HTML、CSS、JS 以及数据库等等的常用知识，这些知识在开发中反复的使用，帮助我在工作以及面试的简历上也添上了浓重的一笔。

### 7.2 展望

在开发完本系统后，我认为为了健壮小程序整体网站的流畅运转，强大的后端服务器以及数据库也是可以在云开发内部通过云函数直接完成的。对于自己开发的小程序，我认为还有许多地方可以优化，能够增删一些功能或代码。比如用户可以在首页搜索对应的丢失物品，直接找到自己丢失的物品；比如可以添加地图的 API，用户可以具体到某个地点直接查询物品；比如用户可以直接在小程序中加发布者为好友，直接在小程序内沟通，而不是拨打电话或者添加微信后再进行沟通等等。再或许，用上更好的设计模式，遵循更好的设计原则等等，都会带来更加完善的系统。

所以在软件开发日渐成熟的领域，小程序的不断完善一定会给开发者和用户更多的惊喜。十分感激这些不断改变生活运行方式的程序员们，是他们使得人机交互日益完备、日益便捷，这给予了用户取之不尽用之不竭的快乐。





## 参考文献

- [1] 骆斌, 丁二玉, 刘钦, 软件工程的技术基础[M]. 北京: 机械工业出版社, 2012
- [2] Baron Schwartz, Peter Zaitsev ,Vadim Tkachenko High Performance MySQL [M]. Third Edition, New York:Housee of Electronics Industry, 2013
- [3] Eric A.Meyer, Estelle Weyl , 安道译, CSS 权威指南[M]. 北京: 中国电力出版社, 2019
- [4] Mark Allen weiss 著 冯舜玺, 数据结构与算法分析 (Java 语言描述) [M]. 北京: 机械工业出版社, 2014
- [5] 张同光, 陈明, 李跃恩, 刘艳君, 沈林, 张家平, Linux 操作系统[M]. 北京: 清华大学出版社, 2014
- [6] 郭卡, 戴亮, Python 数据爬取技术与实战手册, 北京: 中国铁道出版社, 2018
- [7] 陈惠贞, 陈俊荣, PHP7&MySQL 跨设备网站开发[M].第二版, 北京: 清华大学出版社 2017
- [8] 刘萍, 李学峰, 谢旻旻, 赵颖, 潘春花, ASP.NET 动态网站设计教程[M]. 第二版, 北京: 清华大学出版社, 2016
- [9] 汤小丹, 梁红兵, 哲凤屏, 汤子瀛, 计算机操作系统[M]. 第四版, 西安: 西安电子科技大学出版社, 2018
- [10] Douglas Crockford, 赵泽欣, JavaScript 语言精辟[M]. 电子工业出版社, 2012
- [11] 张帆, VUE.JS 项目开发实战, 机械工业出版社, 2018
- [12] ElisabethRobson,徐阳, 丁小峰等译, Head First HTML 与 CSS[M]. 第二版, 北京: 中国电力出版社, 2013
- [13] Freeman, OReilly Taiwan 公司译, Head First 设计模式[M]. 北京: 中国电力出版社, 2007
- [14] Erich Eamma, 刘建中 译, 设计模式——可复用对象软件的基础[M]. 北京: 机械工业出版社, 2019
- [15] 程杰, 大话设计模式[M]. 北京: 清华大学出版社, 2007
- [16] 率辉, 数据结构高分笔记[M]. 北京: 机械工业出版社, 2018
- [17] 张帆, 微信小程序项目开发实践——用 WePY、mpvue、Taro 打造高效的小程序[M]. 北京: 电子工业出版社, 2019
- [18] 高洪涛, 从零开始学微信小程序开发[M]. 北京: 电子工业出版社, 2017
- [19] 苏震巍, 微信开发升读解析[M]. 北京: 电子工业出版社, 2017
- [20] 李刚, 疯狂 XML 讲义[M]. 北京: 电子工业出版社, 2019
- [21] 龚小勇, 邱跃鹏, 微信小程序应用开发[M]. 北京: 高等教育出版社, 2018





## 致 谢

经过几个月的努力，本次毕业设计已经接近尾声。首先要感谢郭慧敏导师对我的教导，郭老师对学生认真负责的态度我从中学习到很多。老师们以及网络上各位技术大牛严谨的治学精神和深厚的理论水平也都使我受益匪浅。同时，要感谢 Bilibili 的 up 主无私的视频奉献，还有微信小程序官方文档的默默支持，特别是在小程序的入门以及云开发的搭建给了我太多太多的灵感，正因为如此我才能顺利的完成设计。当遇见无从下手的需求，那些层出不穷的 BUG 让我慢慢产生了焦虑的心态，这时他们悉心制作的文档以及视频真的给了我无穷无尽的力量。同样十分感谢腾讯微信团队各位技术大牛对小程序平台的搭建，这使得开发者在开发过程中，无论是使用 IDE 还是查阅开发文档皆是如鱼得水，腾讯团队坚持的即用即走的便捷概念，已经从用户渗透到了开发者。最后感谢《软件开发的技术基础》一书的编撰者骆斌老师以及丁二玉老师，在开发过程中觉得思路混乱的时候，此书一遍遍的提醒我软件开发那六个阶段，让我真的收益匪浅。在此，我再说一次谢谢，谢谢大家。