

**项目总结报告**

关于资讯整合的

 微信小程序Collector

G20小组·Grateful dead

浙江大学城市学院

计算机与科学技术学院软件工程专业

Release date : 2019年6月16日

Version : V0.1

# 文档说明

文档信息

文档信息表

|  |  |
| --- | --- |
| 文档原作者 | 许涛 |
| 创建日期 | 2019年6月16日 |
| 当前版本 | V0.1 |
| 上次版本 |  |

**变更记录**

变更记录表

|  |  |  |  |
| --- | --- | --- | --- |
| **变更日期** | **变更人** | **版本** | **备注** |
| 2019.6.16 | 许涛 | V0.1 | 创建文档 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[文档说明 2](#_Toc11683262)

[前言 6](#_Toc11683263)

[因为现在的新闻、资讯APP多得如春笋一般，虽然各个公司都有自己的不同侧重点的推荐阅读内容，但是要是想在碎片化时间内广泛阅读大量的不同领域的热点资讯，就不得不打开不同的APP。这降低了使用者的阅读效率和积极程度。毕竟现在用户很多都是怀有“我必须要不断学习，那我就装些不同的应用来充实自己”的想法，而实际行动起来因为个人的“懒惰”和外界的影响最终不了了之。此款APP能够从不同的平台上将各类热点咨询搜集起来，统一展示用户。因此我们将其命名为“Collector”。 6](#_Toc11683264)

[因为小程序内主打的内容是热点和分类资讯，所以我们的主要面向群体是所有人群。总体来讲，我们的小程序面对的人群是很广泛的。 6](#_Toc11683265)

[项目开发计划 6](#_Toc11683266)

[甘特图截图 6](#_Toc11683267)

[可行性分析 7](#_Toc11683268)

[关键技术分析 7](#_Toc11683269)

[需求分析： 8](#_Toc11683270)

[用户类别与代表： 8](#_Toc11683271)

[界面原型： 8](#_Toc11683272)

[功能需求： 9](#_Toc11683273)

[性能需求： 10](#_Toc11683274)

[静态数量化需求： 10](#_Toc11683275)

[界面原型的用户确认： 10](#_Toc11683276)

[访谈一 10](#_Toc11683277)

[访谈二 11](#_Toc11683278)

[数据字典： 12](#_Toc11683279)

[E-R图： 12](#_Toc11683280)

[总体设计： 13](#_Toc11683281)

[HIPO图： 13](#_Toc11683282)

[业务流图： 13](#_Toc11683283)

[详细设计： 14](#_Toc11683284)

[数据库设计： 14](#_Toc11683285)

[**标识** 14](#_Toc11683286)

[**数据库概述** 14](#_Toc11683287)

[**文档概述** 14](#_Toc11683288)

[**参考文件** 14](#_Toc11683289)

[**数据库级设计决策** 15](#_Toc11683290)

[界面原型设计： 15](#_Toc11683291)

[PDL伪码： 16](#_Toc11683292)

[项目实现与测试： 20](#_Toc11683293)

[代码规范： 20](#_Toc11683294)

[用户手册： 23](#_Toc11683295)

[单元测试 33](#_Toc11683296)

[**1接口测试** 33](#_Toc11683297)

[**测试工具：Postman** 33](#_Toc11683298)

[**2. 小程序官方测试工具** 36](#_Toc11683299)

[**3兼容性测试** 37](#_Toc11683300)

[**4易用性测试** 41](#_Toc11683301)

[**5功能测试** 42](#_Toc11683302)

[**6 UI界面测试：** 52](#_Toc11683303)

[子系统及系统测试 53](#_Toc11683304)

[**3.2.1显示** 53](#_Toc11683305)

[**3.2.2 喜欢/收藏** 54](#_Toc11683306)

[**3.2.3 用户信息和资讯存储** 54](#_Toc11683307)

[**4 集成测试** 55](#_Toc11683308)

[**5 测试用例** 55](#_Toc11683309)

[集成测试 56](#_Toc11683310)

[测试用例 57](#_Toc11683311)

[**资讯页面** 57](#_Toc11683312)

[**我的页面** 57](#_Toc11683313)

[**资讯页面** 57](#_Toc11683314)

[**收藏页面** 57](#_Toc11683315)

[**历史页面** 57](#_Toc11683316)

[项目总结 58](#_Toc11683317)

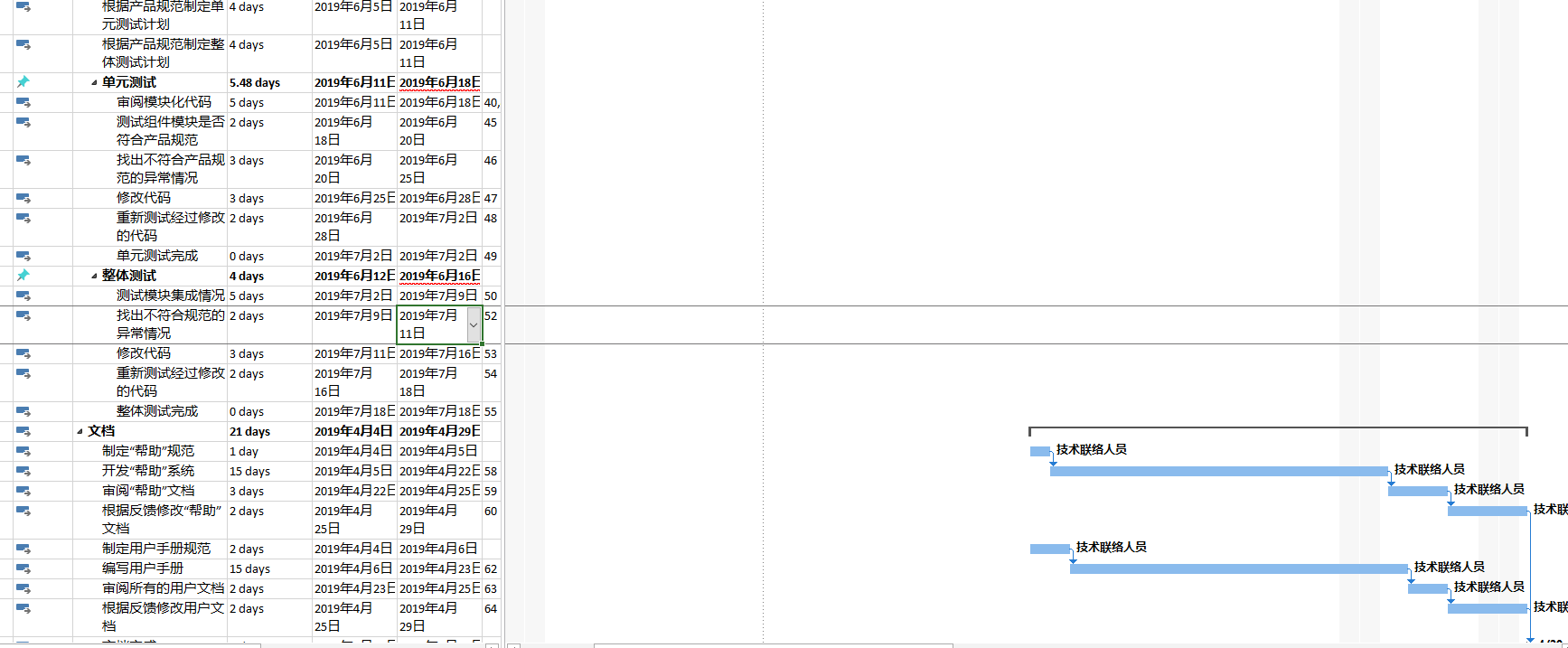
# 前言

# 因为现在的新闻、资讯APP多得如春笋一般，虽然各个公司都有自己的不同侧重点的推荐阅读内容，但是要是想在碎片化时间内广泛阅读大量的不同领域的热点资讯，就不得不打开不同的APP。这降低了使用者的阅读效率和积极程度。毕竟现在用户很多都是怀有“我必须要不断学习，那我就装些不同的应用来充实自己”的想法，而实际行动起来因为个人的“懒惰”和外界的影响最终不了了之。此款APP能够从不同的平台上将各类热点咨询搜集起来，统一展示用户。因此我们将其命名为“Collector”。

# 因为小程序内主打的内容是热点和分类资讯，所以我们的主要面向群体是所有人群。总体来讲，我们的小程序面对的人群是很广泛的。

# 项目开发计划

## 甘特图截图



# 可行性分析

## 关键技术分析

1. 这个系统没有经济效益，而且网络上并没有类似于该项目的完整源码，所以不同的模块所需的技术要分开学，而且相应负责的同学要保持密切的共同，才能实现项目的基本功能，开发这个小程序，我们主要是想让用户在休息和放松时使用。

2. 系统分成前后端，前端是微信小程序的开发，小程序基于HTML5，同时有微信自带的框架MINA，微信的开发者文档将是我们学习的一大利器，同时其拥有成熟的开发者社区，我们对此还是充满信心。

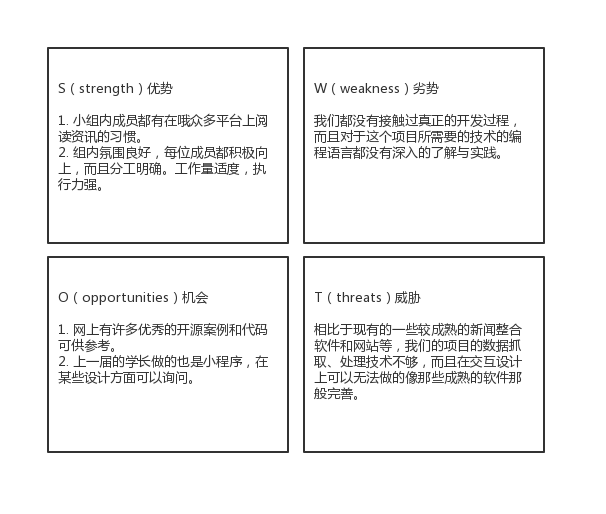
3. 系统的后端上运行NODE.JS，NGINX和MONGODB，使用阿里云。微信小程序也提供了高度封装好的api，减少了网络问题的发生。网上能够参考的优秀教程和源码丰富。我们可以进行大量的学习

4. 最后是微信服务号的使用，微信服务号的接口也较为完善。

5、我们可以通过高强度的学习在三个月内完成开发

# 需求分析：

## 用户类别与代表：



## 界面原型：





## 功能需求：

**资讯阅读者使用的微信小程序：**

3.2.1.1授权

输入：

阅读者添加该微信小程序，添加后进入主界面，点击授权微信账号登入，小程序会自动跳出授权登陆界面信息，点击确认即可

输出：

确认个人信息后，点击提交，小程序就授权成功。

异常处理：

* + - 1. 若用户点击不同意，无法选择个人的个性标签和将用户阅读记录和喜欢导入数库。

3.2.1.2收藏

输入：

点击用户当前阅读新闻下方的心形按钮

输出：

小程序界面提示手藏成功

异常处理：

如果上传失败会显示错误提示

3.2.1.3删除分组：

输入：

点击通知按钮，选择通知分组，在屏幕下方有两个按钮，一个是创建分组一个是删除分组，选中你要删除的组，点击删除分组，确认后，点击删除。

输出：

小程序提示删除成功。

异常处理：

没有选中分组时就点击删除，小程序弹窗提示请至少选择一组删除。

## 性能需求：

### 静态数量化需求：

1.支持的终端数量；

测试阶段支持100台左右的终端。

2.支持同时运行的用户数量；

测试阶段支持100位左右的用户。

3.要处理的信息量和类型。

* + - 1. 文本

## 界面原型的用户确认：

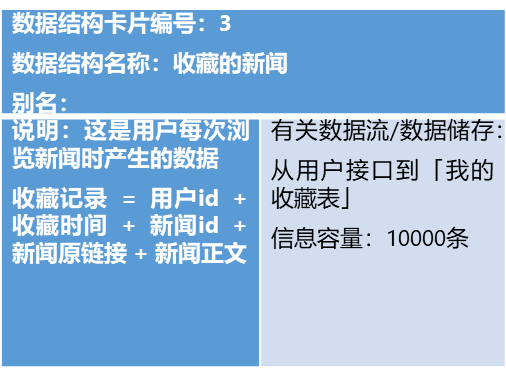
### 访谈一

|  |  |  |
| --- | --- | --- |
| **19.5.8用户需求访谈二** | | |
| 访谈对象：郑同学（计算1703） | 时间： 19.4.24 下午11点30分到11点45分 | 地点：线上交流 |
| 访谈主题纲要：给用户介绍最新的界面原型和功能的设计说明，用户就此内容提出了一些建议。 | | |
| “界面还好，总体风格挺干净的“  “觉得收藏上面的框不太好看，看起来太硬朗了” | | |
| 改善方法 | | |
| 与小组成员先对用户提出的问题进行讨论，探讨改进的可行性和改进的最佳方案。  提出方案后再和用户沟通确认。 | | |

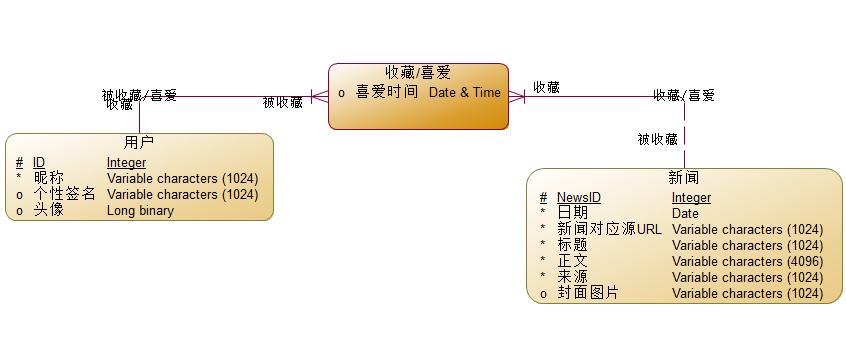
### 访谈二

|  |  |  |  |
| --- | --- | --- | --- |
| **19.5.8用户需求访谈一** | | | |
| 访谈对象：周XX（软工1703） | | 时间：19.5.7 晚23:00-23：10 | 地点：明德一6XX寝室 |
| 主要的纲要：重审并介绍了原型界面以及操作的详细流程，概括了准备提供的功能。用户提出了一些想完善的要点，我们做了记录并准备修改 | | | |
| 序号 | 主要问题/不足 | | 改善方法 |
| 1 | 在“时光机”功能上，每条新闻附着的不应该是新闻的采集时间，应该改为用户收藏该咨询的时间 | | **修改数据库的用户字段的设置** |
| 2 | 页面设置的再精美一些，卡片式的风格就很不错 | | **和设计前端的成员交流** |
| 3 | 出错信息中的当删掉所有的标签时的提示可以去掉，如果没有选择标签的话就按系统自动推荐来（默认推荐收藏和喜爱多的新闻） | | **和小组成员交流** |

## 数据字典：



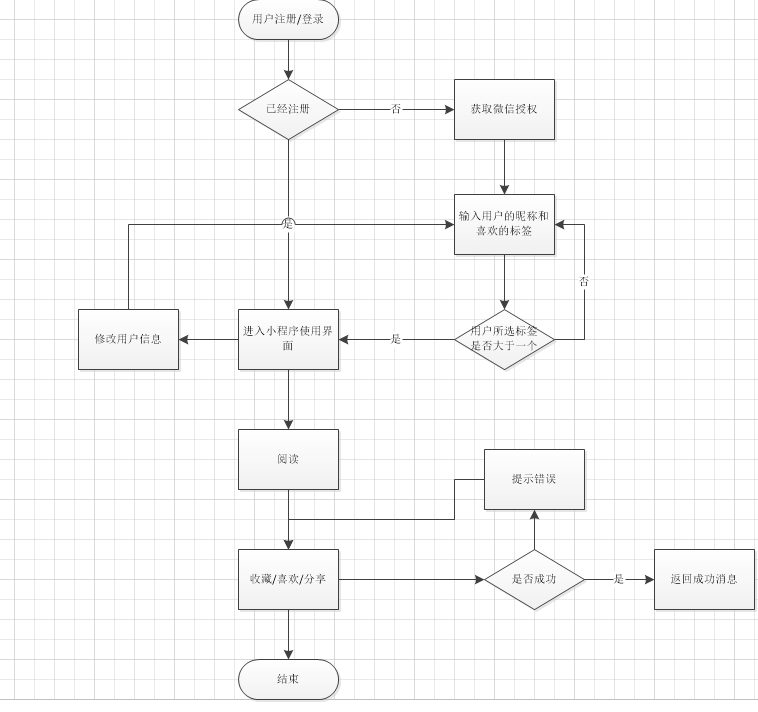
## E-R图：



# 总体设计：

## HIPO图：

## 业务流图：



# 详细设计：

## 数据库设计：

**标识**

DBDD，Collect小程序数据库顶层说明，作者周磊，版本号v0.1，发行号v0.1

**数据库概述**

本数据库主要用途是为了给爬虫程序提供不同资讯的存放的一个容器。同时允许用户在使用小程序的时候通过接口来调用数据库中的资讯，以在界面展示。

本数据库采用的是非关系型数据库，采用此类型的数据库的原因是考虑到我们具体的需求。因为网页的爬取的信息并不是有规律的，不同的网页，图片和文字的布局都不一样。所以不能按照同一模板来存放不同字段的信息。所以我们采用了非关系型数据库，先把整个网页的文档以JSON格式存下来。在通过分类的程序，把关键字和页面的布局存下来。这样比Mysql等关系型数据库的存放更为灵活。

维护方法我们的开发人员会每隔两天查看检查数据库是否有异常（字段的内容是否合理，有没有怪异的参数信息）还有服务器的运行的压力（暂时未讨论出明确的解决方案）。

数据库搭建在Nginx + Node.js + Express的坏境下。选择的是非关系型数据库的热门数据库MongoDB。

数据库部署在阿里云主机上，24小时不间断运行，同时我们会定期的创建快照，保证资料的安全和稳定。

**文档概述**

本文档用于向开发人员和客户介绍本项目采用的数据库的数据结构、定义的接口和交互的逻辑，并尽量简洁明了的介绍数据库所起到的作用和数据的流向以及处理过程，相信这在对后期的测试和数据库的开发是一个指南。

**参考文件**

* **GB/T 8567-2006 《中华人民共和国国家标准-计算机软件文档编制规范》 之** *7.14数据库(顶层)设计说明(DBDD)*
* **《软件工程导论（第六版）》 张海藩 牟永敏 编著 ISBN 978-7-302-33098-1**
* **《PMBOK》项目管理计划书**

**数据库级设计决策**

* 关于该数据库应接受的查询或其他输入和它应产生的输出(显示、报告、消息、响应等)的设计决策————通过Node.js结合HTTP的GET和POST，调用接口
* 有关响应每次输入或查询的数据库行为的设计决策，包括动作、响应时间和其他性能特性、所选择的方程式/算法/规则、配置和对不允许的输入的处理。——————每次输入，【1】如果是请求搜索相关ID和标签的资讯的集合，将通过数据库语言在数据库中搜索（提前会建立好索引并定期更新），搜索并返回对应的信息。【2】如果是爬虫请求插入信息，先检测字段有无异常，无异常则插入，并返回成功结果。响应时间每条不应超过5秒
* 有关数据库/数据文件如何呈现给用户的设计决策————通过前端的设计以一个整洁的布局将资讯展示给用户
* 有关数据库要提供的可用性、保密性、私密性和运行连续性的层次与类型的设计决策——————数据库的登录信息只有组内开发人员知道，并且使用前需在小组内申明理由并获得全员同意。同时给服务器增加一些安全的保障（购买额外的服务），同时每天给服务器的数据库记录进行备份，选择的阿里云的主机运行较为稳定且广受好评

## 界面原型设计：

（见上方）

## PDL伪码：

1.1检测用户是否已经注册：

procedure 用户登录界面

if 没有注册过 then

跳转至注册界面

else

跳转至用户首界面

end if

end 用户登录界面

1.2注册页面：

procedure 用户注册界面

从分类栏选择推送标签

读入标签id

选择一个标签，则 标签数++

if <标签数=0> then

弹出对话框：请选择一个以上的标签

else if<标签数>0>then

弹出对话框：操作成功

end if

end if

end 用户注册界面

1.3登陆页面

Procedure 用户登陆页面

If <小程序已登录> then

Procedure 首页

Else if <小程序未登录> then

Procedure 登陆弹窗

If click <确定> then

Procedure 主页

Else if click <取消> then

Procedure 登陆弹窗

输出：请重新登录

end if

2.用户主界面

2.1下拉刷新页面：

Procedure 主页

onPullDownRefresh()

出现小点

and打印：玩命加载中

与服务器段连接wx.request{}

wx.hideLoading()

wx.stopPullDownRefresh()

if <加载成功> then

输出：刷新成功

输出新载入的资讯栏内容

else if<加载失败> then

输出：刷新失败，请重试

end if

2.2上拉加载页面：

Procedure 主页

获取当前页和共几页

onReachBottom()

打印：玩命加载中

加载成功，页数++

与服务器端连接wx.request

wx.hideLoading()

if <加载成功> then

输出：加载成功

and输出新载入的资讯栏内容

else if<加载失败> then

输出：加载失败，请重试

end if

2.3跳转资讯内容页面

Procedure 主页

if <获取内容成功> then

跳转该条资讯页面

输出资讯内容

else if <获取内容失败> then

弹出：获取资讯失败，请重试

end if

2.4切换分类

Procedure 主页

If click 某一分类 then

输出该分类的资讯

End if

If click 标签图标 then

Procedure 标签页面

End if

3.我的页面

3.1跳转收藏页面

Procedure 我的

If click 收藏 then

Procedure 收藏页面

if c <收藏数=0>then

打印：暂无收藏资讯

else if c <收藏数>0>then

Procedure 一天内

If c <收藏时间<1天>then

输出资讯栏内容

Procedure 一周内

Else if c <收藏时间<1周>then

输出资讯栏内容

Procedure 十五天内

Else if c <收藏时间<15天>then

输出资讯栏内容

End if

End if

3.2删除收藏

Procedure 收藏页面

If click 删除 then

选择，选择数++

if<选择数=0> then

弹出对话框：你还没有选择任何资讯

Else

弹出确定框

If click 确定 then

将这些收藏的资讯从数据库中删除

弹出对话框：删除成功

Else if click 取消 then

End if

end if

end if

3.3跳转历史页面

Procedure 我的

If click 历史 then

Procedure 历史页面

if<历史数=0>then

打印：暂无收藏资讯

else if<历史数>0>then

Procedure 一天内

If c <历史时间<1天>then

输出资讯栏内容

If click 一周内 then

Procedure 一周内

Else if c <历史时间<1周>then

输出资讯栏内容

End id

If click 一周内 then

Procedure 十五天内

Else if c <历史时间<15天>then

输出资讯栏内容

End if

End if

End if

3.4跳转标签页面

Procedure 我的

If click 标签then

Procedure 标签页面

If click 添加 then

弹出添加符号

If click 加号 then

订阅标签数++

发送标签id至后端

End if

End if

If click 删除 then

弹出删除符号

If click 减号 then

订阅标签数--

发送删除请求至后端

End if

If <标签数<0> then

弹出对话框：请留下一个标签

End if

End id

4.资讯页面

4.1点赞

Procedure 资讯内容

If click 点赞 then

点赞数++

End if

If click 点赞 then

点赞数--

End if

4.2评论

Procedure 资讯内容

If click 评论 then

输入：评论内容

If click 发送 then

上传至数据库

输出：评论内容

End if

If click 删除 then

评论数--

End if

If click 回复 then

输入：回复内容

If click 发送 then

上传至数据库

输出：回复内容

End if

# 项目实现与测试：

## 代码规范：

空格与运算符

通常运算符 ( = + - \* / ) 前后需要添加空格:

实例:

var x = y + z;  
var values = ["Volvo", "Saab", "Fiat"];

代码缩进

使用 Tab来缩进代码块：

语句规则

简单语句的通用规则:

* 一条语句通常以分号作为结束符。

实例:

var values = ["Volvo", "Saab", "Fiat"];  
  
var person = {  
    firstName: "John",  
    lastName: "Doe",  
    age: 50,  
    eyeColor: "blue"  
};

复杂语句的通用规则:

* 将左花括号放在第一行的结尾。
* 左花括号前添加一空格。
* 将右花括号独立放在一行。
* 不要以分号结束一个复杂的声明。

函数:

function toCelsius(fahrenheit) {  
    return (5 / 9) \* (fahrenheit - 32);  
}

循环:

for (i = 0; i < 5; i++) {  
    x += i;  
}

条件语句:

if (time < 20) {  
    greeting = "Good day";  
}

else {  
    greeting = "Good evening";  
}

对象规则

对象定义的规则:

* 将左花括号与类名放在同一行。
* 冒号与属性值间有个空格。
* 字符串使用双引号，数字不需要。
* 最后一个属性-值对后面不要添加逗号。
* 将右花括号独立放在一行，并以分号作为结束符号。

实例:

var person = {  
    firstName: "John",  
    lastName: "Doe",  
    age: 50,  
    eyeColor: "blue"  
};

短的对象代码可以直接写成一行:

实例:

var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};

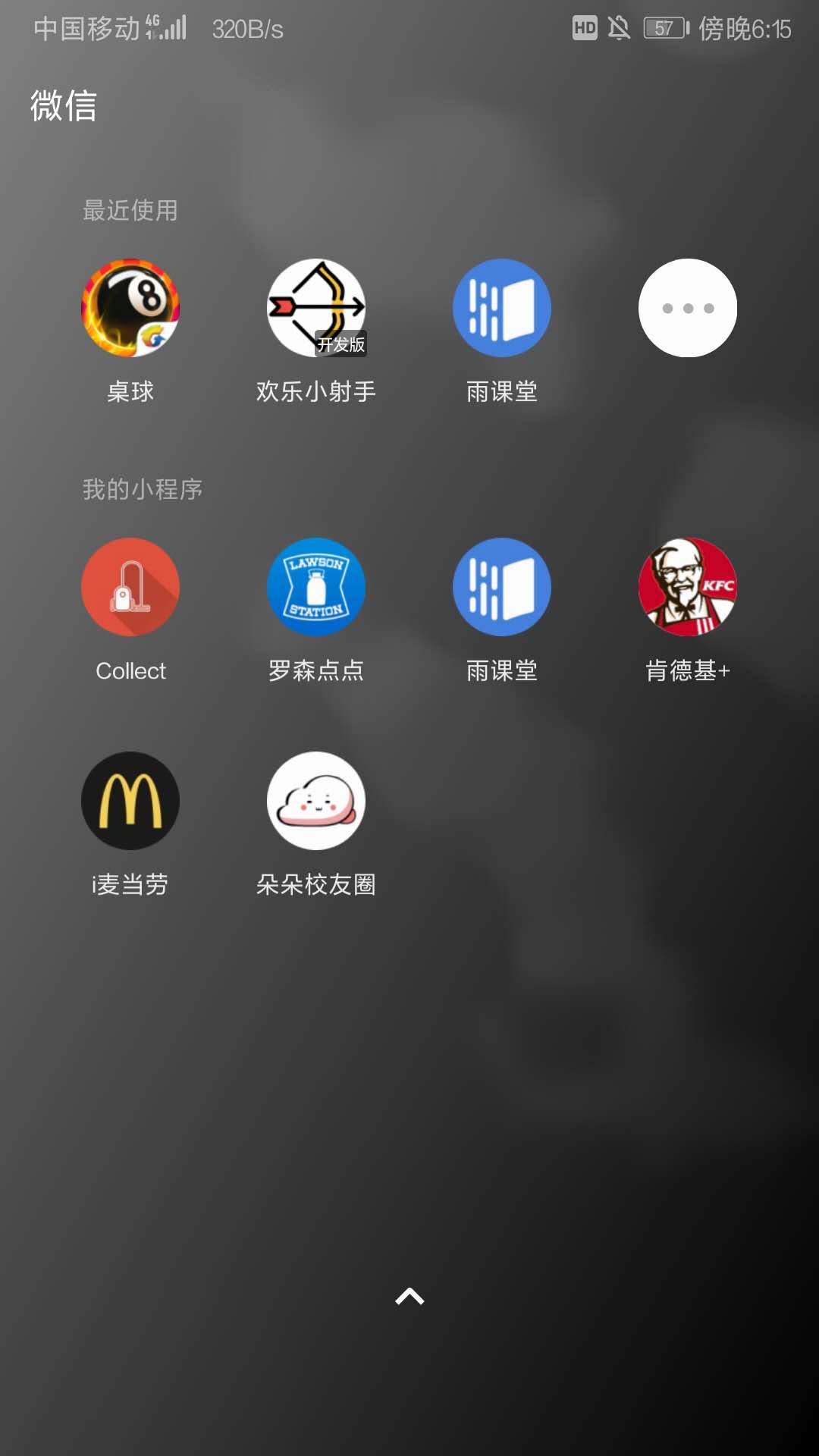
变量名

驼峰

使用小写文件名

你必须保持统一的风格，我们建议统一使用小写的文件名。

## 用户手册：



登陆微信后，可以下滑页面进入小程序栏目，如第一幅图所示；也可以点击省略号进入详细栏目，如第二幅图所示。在两个页面，都可以通过点击Collector的图标进入小程序。点击后，将会跳出一下界面：



等待几秒钟，即可进入小程序主页面，如下图所示：

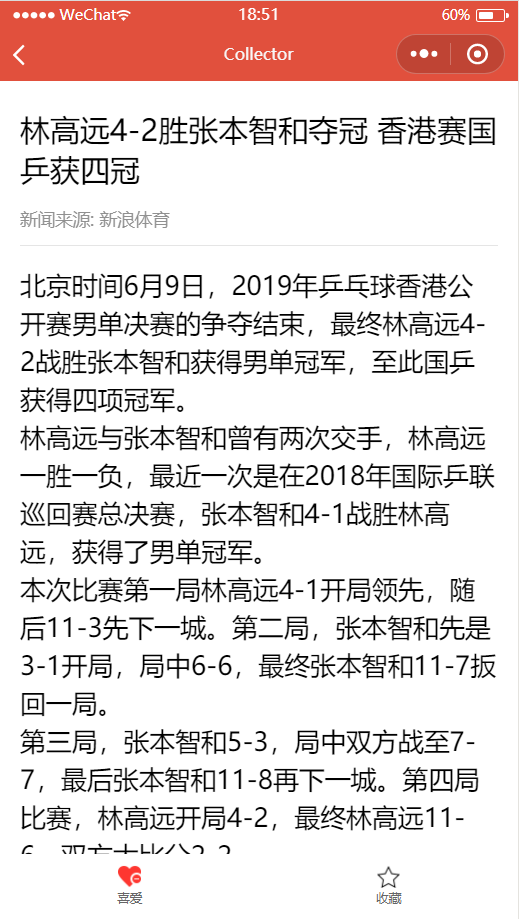


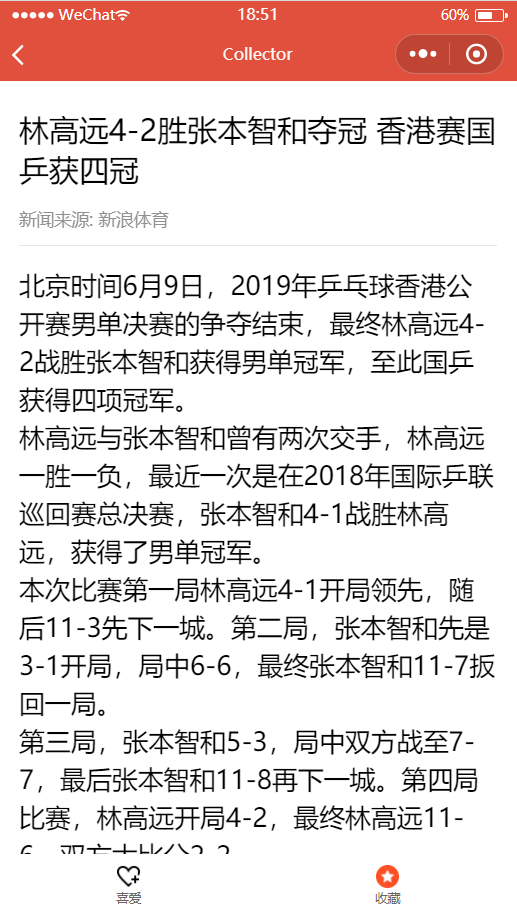
在此界面我们可左右滑动标签栏，来显示更多分类。点击某一个标签后，可进入该资讯的推送界面。若再次点击标签，即可刷新资讯栏目的推送资讯。如下图所示：



资讯栏目中，我们可以上下滑动页面来显示更多资讯，每一个资讯条都包含有资讯的标题，时间和来源网站名。点击资讯条后，即可进入该资讯的详细页面进行阅读。如下所示：





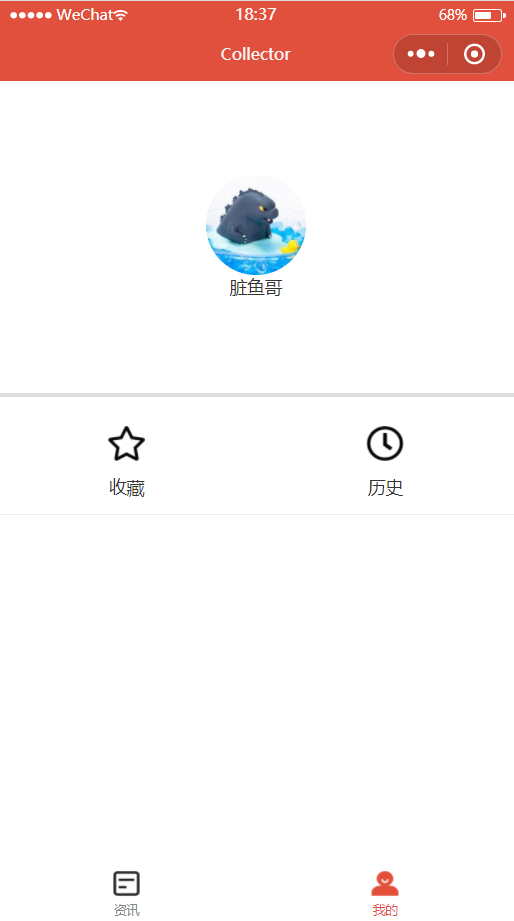


从上到下依次为标题，来源，内容。下滑页面时即可显示更多内容。

底部为固定的栏目，点击后可实现喜爱和收藏的功能。其中收藏功能将在之后详细介绍。点击图标后，该图标便会变成红色，再次点击后则变回原先的颜色。

点击顶部左上角“<”图标后，可返回之前的页面。

回到主页，点击“我的”图标后，可转到“我的”页面，如下图所示：



顶上为用户微信头像有微信名称，这个需要获取用户个人信息。

中间是“收藏”与“历史”图标。点击后可进入对应的界面。

以下为“收藏”界面：



该页面对用户收藏资讯的时间做了一个分类，将不同时间收藏的资讯放在资讯栏中。

以下为“历史”页面：



和“收藏”页面一样，按照查看过的时间进行分类。

## 单元测试

**1接口测试**

**测试工具：Postman**

1.1测试URL：[https://api.jisuapi.com/news/get?channel=a&start=b&num=c&appkey=ff5c61c75de843af](https://api.jisuapi.com/news/get?channel=a&start=0&num=b&appkey=ff5c61c75de843af)

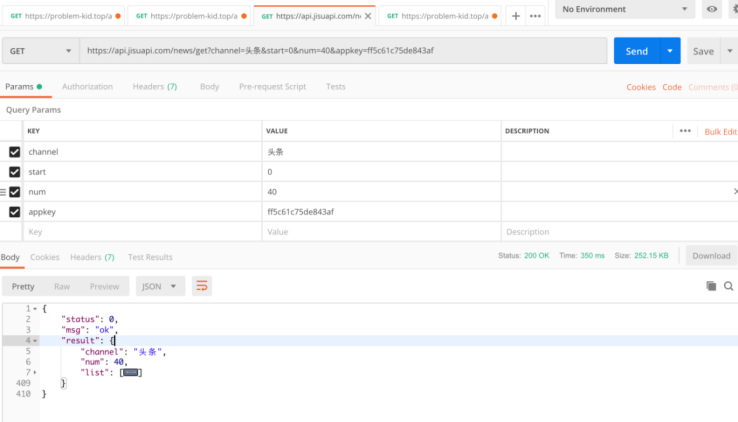
1.1.1测试参数：

1.1.1.1

.a=头条，b=0，c=40

预期结果：返回一个json文件，包含40条头条新闻以数组的形式。

测试结果：

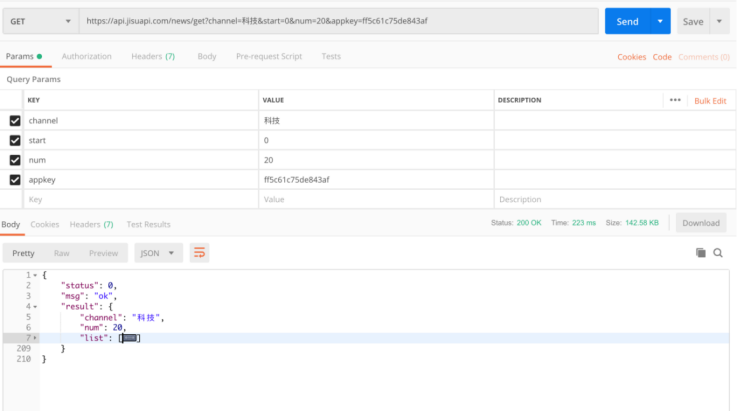


1.1.1.2

a=科技b=0 c=20

预期结果：返回一个json文件，包含20条科技新闻以数组的形式。

测试结果

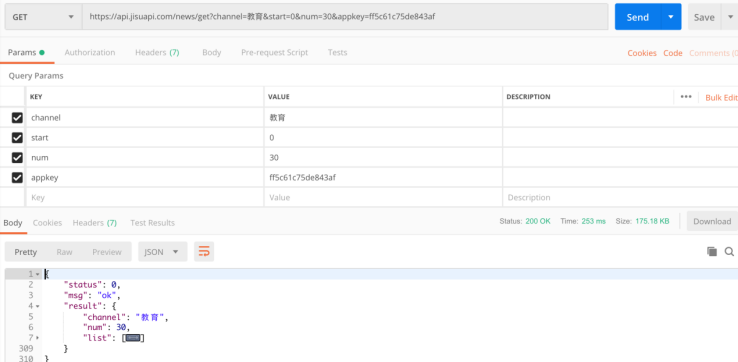


1.1.1.3

a=教育b=0 c=30

预期结果：返回一个json文件，包含20条科技新闻以数组的形式。

测试结果



1.2 测试URL

https://problem-kid.top/api/v1/user/TimeMachine?openid=a&time=b&typ=c

1.2.1测试参数：

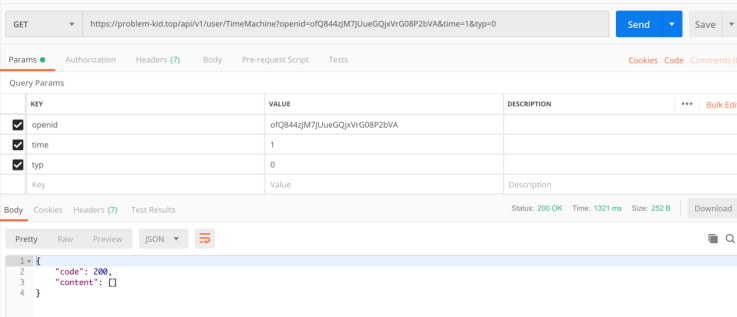
1.2.1.1

a=ofQ844zJM7JUueGQjxVrG08P2bVA b=1 c=0

参数介绍：a表示用户的唯一openid ，b=1，2，3分别对应着 一天内和15天内和一个月内的用户喜爱数据 c=0，1分别对应着用户喜爱和收藏

预期结果：没有返回新闻内容

测试结果：

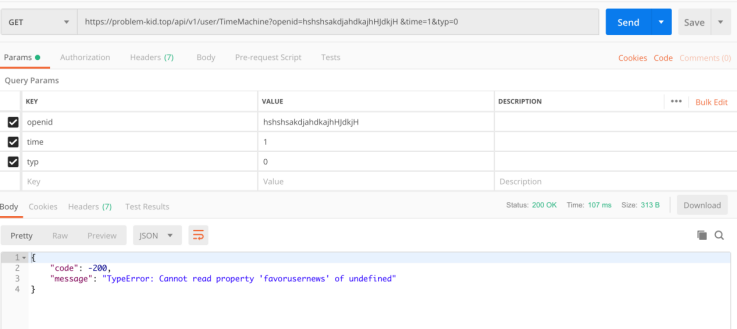


1.2.1.2

a=hshshsakdjahdkajhHJdkjH b=1 c=0

预期结果：返回 错误的状态码

测试结果：

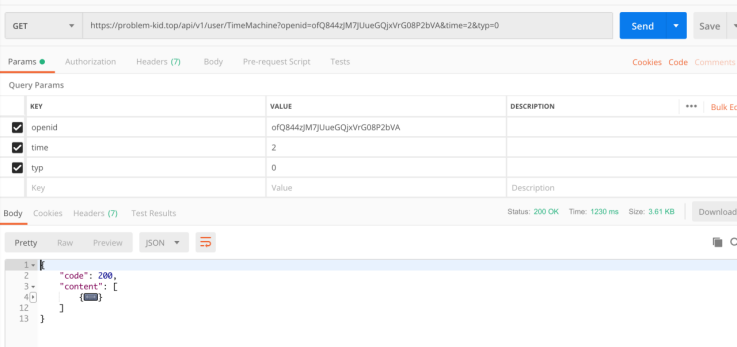


1.2.1.2

a=ofQ844zJM7JUueGQjxVrG08P2bVA b=2 c=0

预期结果：返回15天内用户喜爱的数据 以json的形式

测试结果：

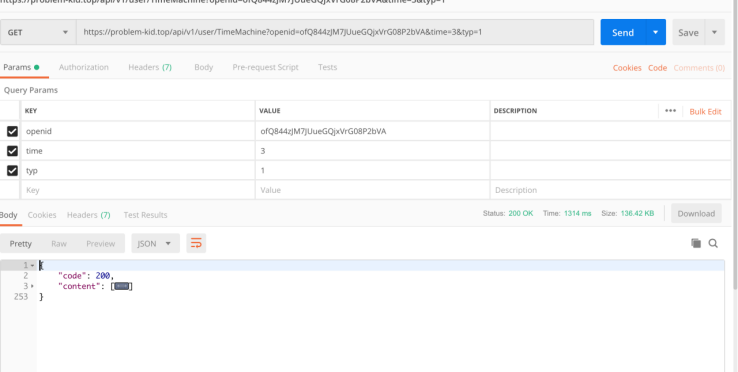


1.2.1.3

a= ofQ844zJM7JUueGQjxVrG08P2bVA b=3 c=0

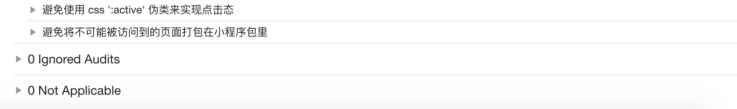
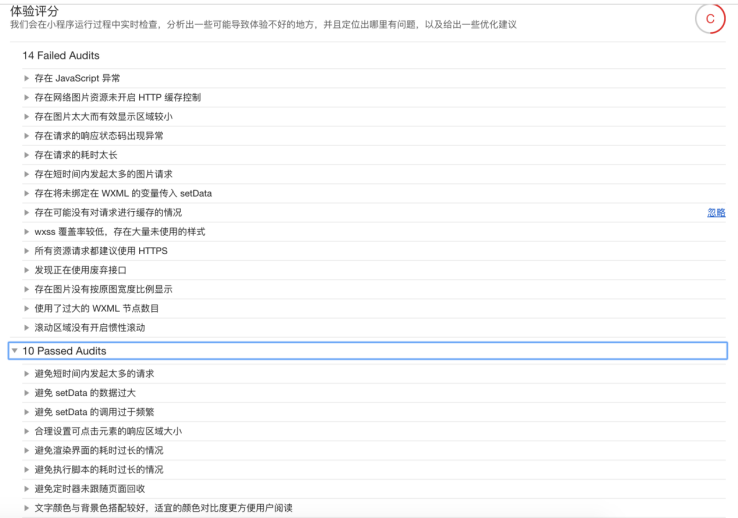
预期结果：返回30天内用户喜爱的数据 以json的形式

测试结果：



1. **小程序官方测试工具**

人工操作得出的测试结果



**3兼容性测试**

**手机操作系统**：

**安卓系统**：

三星S8（SM-G9500、Android 9）

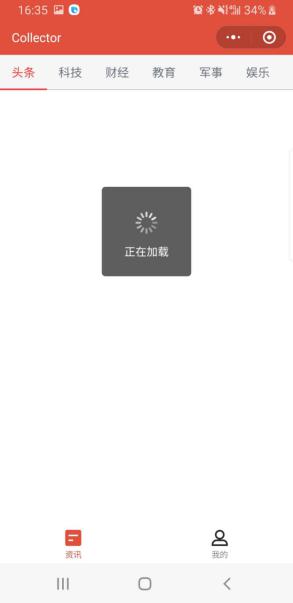
小程序界面



**屏幕显示**：正常

**微信版本**：7.0.4

**断网下的显示情况**：





**4易用性测试**

**页面加载的进度条或动画特效：**



**加载成功提醒：**

无

**交互的对话框提醒**



**新闻列表展示，需要点击详情进行下一步的查看**

****

**异常状态的显示**

断网下无法加载图片

**5功能测试**

**用户流畅度测试**

在网络情况较好的时候，用户运行较为流畅

其中查看「喜爱」和「收藏」的时候会有至少2秒的延迟产生

**用户导航情况**

入口和出口都有十分明确指向，便于用户操作



图 1



图 2



图 3

**与小程序官方的出入口是否有冲突**

与小程序官方的「结束小程序按钮」没有冲突

**异常情况下的处理**

在阅读新闻的详情的时候的异常下用户可以直接点击左上角的返回键退出详情界面

如图4



图 4

当用户在「喜爱」、「收藏」界面遇到异常时可以点击左上角的返回键或是直接点击上方的标签页面进行重新加载、刷新



图 5

**用户的输入**

本程序没有任何需要用户输入的接口。

**用户的可操作空间**

用户的可点击区域空间充足，没有很密集的操作区域

**是否有无法添加、无法删除、无法返回、无法退出的现象**

无

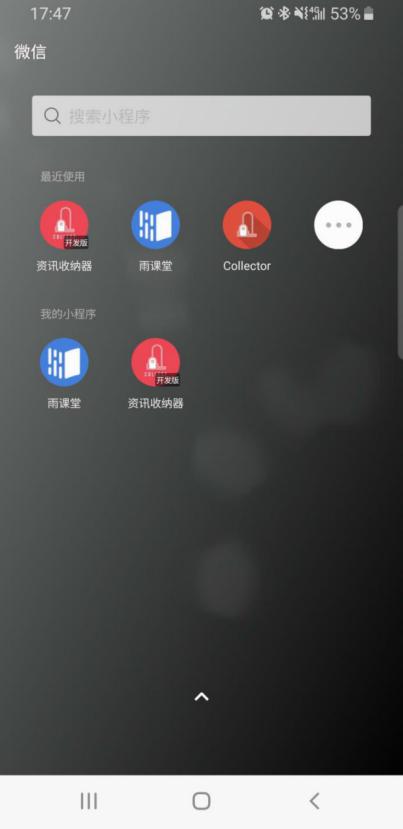
**页面跳转的提示**

无

**与微信功能的交互**

**入口：**

微信下拉进入小程序窗口

****

点击资讯收纳器就可以进入了

注：因为微信政策问题，本程序在小程序商城无法搜索到

**小程序的缓存情况**

清空缓存信息的话，小程序要求重新授权登陆信息

当同一账号在不同的设备上登入时，小程序会保存用户喜爱的新闻和收藏的新闻

**权限测试：**

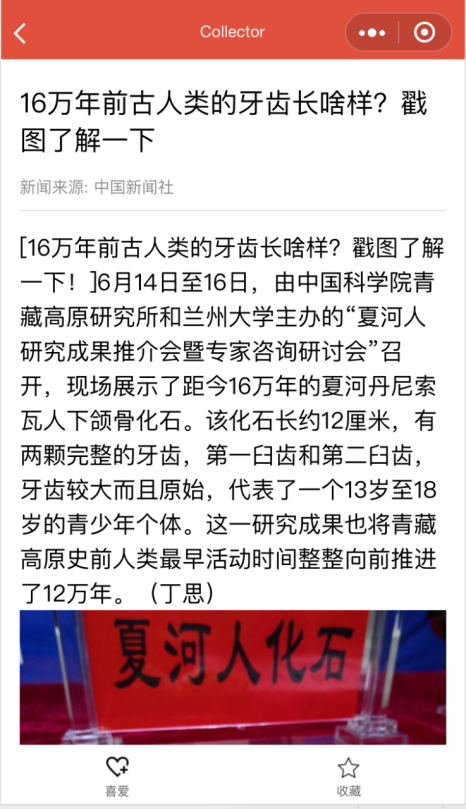
未授权微信小程序的情况下，全部功能会受到限制

在授权微信小程序的情况下，是以微信的身份操作业务

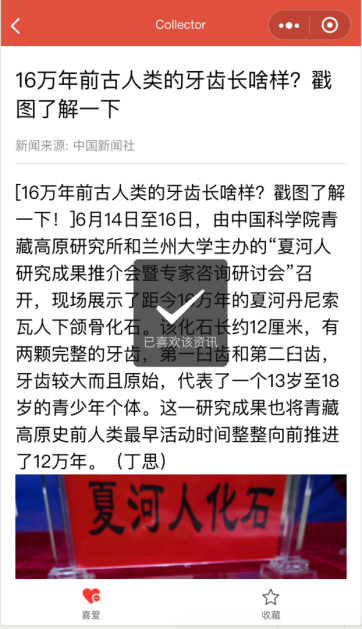
同一账号可以在不同的手机端查看个人的数据

**阅读新闻**

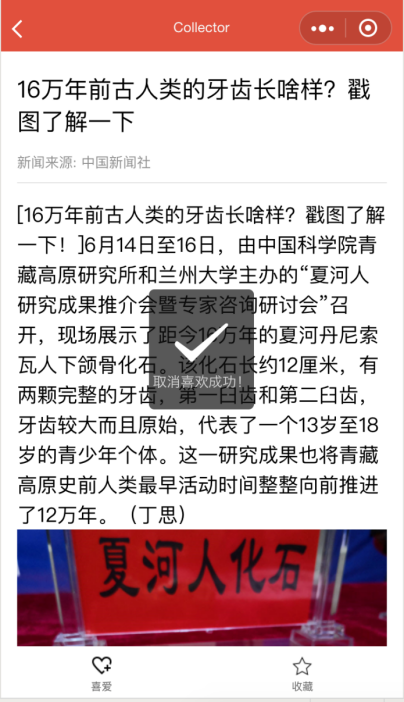
点击列表中的新闻，成功的跳转到新闻的详情界面



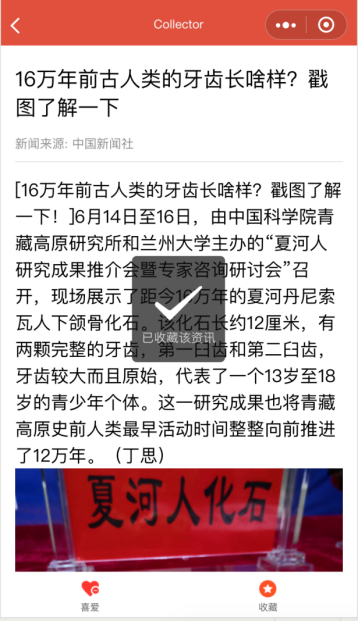
**喜爱新闻**



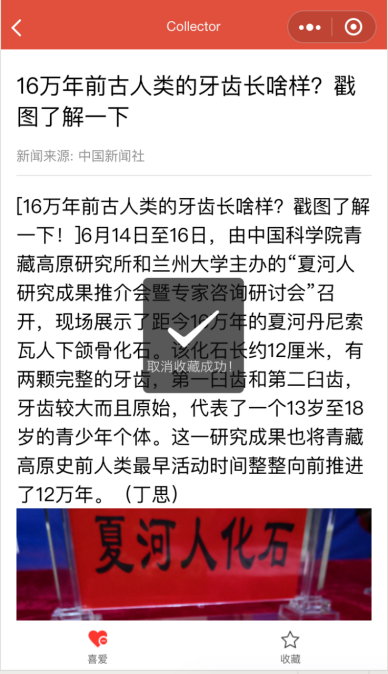
**取消喜爱**

****

**收藏：**

****

**取消收藏**

****

**查看喜爱记录**

****

**查看收藏记录**

****

**6 UI界面测试：**

**标签栏目分别为「头条」「科技」「财经」「教育」「军事」「娱乐」「体育」**

**底部的栏目分别为「资讯」「我的」**

**在不同的页面我们都使用了同一种交互方式：点击进入和点击返回**

## 子系统及系统测试

**3.2.1显示**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试模块：显示 | | | | |
| 标识符：1001 | | | | |
| 测试项：用于向用户显示最新资讯功能 | | | | |
| 测试环境：iPhone(iOS11+12) + Android 8.0 | | | | |
| 前置条件：用户手机网络情况正常 | | | | |
| 操作步骤：   1. 点击小程序按钮 2. 等待加载 3. 查看列表显示 4. 点击某一列表栏 5. 等待详细页面显示 6. 点击返回按钮 7. 等待返回列表页 8. 上拉刷新 9. 点击“我的”按钮 | | | | |
| 输入数据 | 期望输出 | | | 实际结果 |
| 手势动作——点击小程序按钮 | 正确进入小程序 | | | 正确进入 |
| 等待显示 | 显示列表 | | | 正确显示 |
| 点击某一列表栏 | 跳转到指定页面 | | | 正确跳转 |
| 点击返回按钮 | 返回到列表页 | | | 正确返回 |
| 上拉刷新 | 正确显示列表页，有新的新闻 | | | 正确重新加载，但是由于API的新闻刷新时间长，故并无新的新闻显示 |
| 点击“我的”按钮 | 切换到“我的”页面 | | | 正确显示 |
| 结论：通过 | | 测试人：周磊 | 测试时间：2018.5.25 | |

**3.2.2 喜欢/收藏**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试模块：喜欢/收藏 | | | | |
| 标识符：1002 | | | | |
| 测试项：用户可操作的功能，喜欢和收藏功能 | | | | |
| 测试环境：iPhone(iOS11+12) + Android 8.0 | | | | |
| 前置条件：已经点击了列表页的某一界面 | | | | |
| 操作步骤：   1. 点击喜欢 2. 再点击喜欢 3. 点击收藏 4. 再点击收藏 | | | | |
| 输入数据 | 期望输出 | | | 实际结果 |
| 点击喜欢 | 弹窗提示成功 | | | 提示成功 |
| 再点击喜欢 | 弹窗提示成功 | | | 提示成功 |
| 点击收藏 | 弹窗提示成功 | | | 提示成功 |
| 再点击收藏 | 弹窗提示成功 | | | 提示成功 |
| 结论：通过 | | 测试人：周磊 | 测试时间：2018.5.26 | |

**3.2.3 用户信息和资讯存储**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试模块：用户信息和资讯存储 | | | | |
| 标识符：1003 | | | | |
| 测试项：用户操作时后端数据库的交互功能 | | | | |
| 测试环境：Windows10 | | | | |
| 前置条件：用户进行了登录和操作 | | | | |
| 操作步骤：  用户进入小程序  用户进入某一资讯  用户点击喜欢/收藏，再点击 | | | | |
| 输入数据 | 期望输出 | | | 实际结果 |
| 用户进入小程序 | 若新用户数据库User表增加字段，否则不变 | | | 增加成功 |
| 点击喜爱 | 若新用户数据库User表的喜爱字段增加对应资讯ID，否则不变 | | | 增加成功 |
| 再点击喜爱 | User表的喜爱字段对应资讯的ID删除 | | | 删除成功 |
| 结论：通过 | | 测试人：周磊 | 测试时间：2018.5.26 | |

**4 集成测试**

采用自底向上的的集成测试

把每个单元组合成实现这些子系统的族

为每个子系统编写测试用例

对有单元组成的子系统进行测试

重复上述步骤，把小的子系统组合成大的子系统再进行综合测试

**5 测试用例**

喜欢/收藏

|  |  |  |
| --- | --- | --- |
| 输入数据 | 期望输出 | 实际结果 |
| 点击喜欢 | 弹窗提示成功 | 提示成功 |
| 再点击喜欢 | 弹窗提示成功 | 提示成功 |
| 点击收藏 | 弹窗提示成功 | 提示成功 |
| 再点击收藏 | 弹窗提示成功 | 提示成功 |

用户信息和资讯存储

|  |  |  |
| --- | --- | --- |
| 用户进入小程序 | 若新用户数据库User表增加字段，否则不变 | 增加成功 |
| 点击喜爱 | 若新用户数据库User表的喜爱字段增加对应资讯ID，否则不变 | 增加成功 |
| 再点击喜爱 | User表的喜爱字段对应资讯的ID删除 | 删除成功 |

显示

|  |  |  |
| --- | --- | --- |
| 手势动作——点击小程序按钮 | 正确进入小程序 | 正确进入 |
| 等待显示 | 显示列表 | 正确显示 |
| 点击某一列表栏 | 跳转到指定页面 | 正确跳转 |
| 点击返回按钮 | 返回到列表页 | 正确返回 |
| 上拉刷新 | 正确显示列表页，有新的新闻 | 正确重新加载，但是由于API的新闻刷新时间长，故并无新的新闻显示 |
| 点击“我的”按钮 | 切换到“我的”页面 | 正确显示 |

## 集成测试

采用自底向上的的集成测试

把低层子系统组合成实现这些子系统的族

为每个子系统编写测试用例

对有单元组成的子系统进行测试

重复上述步骤，把小的子系统组合成大的子系统再进行综合测试

## 测试用例

**资讯页面**

|  |  |  |
| --- | --- | --- |
| 测试功能 | 预期结果 | 评价 |
| 点击分类 | 显示对应页面 |  |
| 点击资讯 | 跳转至对应资讯页面 |  |
| 左右滑动标签 | 显示更多分类标签 |  |
| 上下滑动页面 | 显示更多资讯条目 |  |
| 点击我的 | 跳转至我的页面 |  |

**我的页面**

|  |  |  |
| --- | --- | --- |
| 测试功能 | 预期结果 | 评价 |
| 点击资讯 | 跳转至资讯页面 |  |
| 点击收藏 | 跳转至收藏页面并显示收藏的资讯 |  |
| 点击历史 | 跳转至历史页面并显示浏览历史 |  |

**资讯页面**

|  |  |  |
| --- | --- | --- |
| 测试功能 | 预期结果 | 评价 |
| 点击喜爱 | 图标变红/白并存入数据库 |  |
| 点击收藏 | 图标变红/白并存入数据库 |  |
| 上下滑动页面 | 显示更多资讯内容 |  |

**收藏页面**

|  |  |  |
| --- | --- | --- |
| 测试功能 | 预期结果 | 评价 |
| 上下滑动页面 | 显示更多资讯条目 |  |
| 点击一周内/十五天内/一天内 | 显示对应天数的资讯记录 |  |

**历史页面**

|  |  |  |
| --- | --- | --- |
| 测试功能 | 预期结果 | 评价 |
| 上下滑动页面 | 显示更多资讯条目 |  |
| 点击一周内/十五天内/一天内 | 显示对应天数的资讯记录 |  |

# 项目总结

在一个学期的锻炼中，我们体会到，软件开发中的确有许多问题会相互纠缠和累积在一起，尤其是瀑布模型，当前一环节出现问题时，下一环节就会很难进行下去，让人捉襟见肘。虽然软件开发过程中的确没有“银子弹”，但这不代表我们就应该在“焦油坑”里面越陷越深，前人给我们总结了许多有用的软件开发过程的经验和建议可供我们借鉴和参考，我们可以站在前人的肩膀上继续进步。

结合我们自身的项目进行分析，我们曾在可行性分析阶段出现一个大的问题。我们进行需求分析前，隐隐觉得我们的可行性分析没有做到位。虽然我们的项目在技术上是可行性，但是由于我们做的项目是微信小程序，平台是腾讯公司的，我们做的软件必须符合腾讯公司对于小程序的要求。我们去找了许多的腾讯方面关于微信小程序、微信号的使用和服务号的使用规范。但我们还是不敢确定我们是否该进行下去，因为我们知道我们的软件开发模式是瀑布模型，一旦前面一个环节出现不到位的情况，后面做的一切工作都会化为泡沫。于是我们求助了杨枨老师，杨枨老师给了我们许多建议，建议我们找腾讯方面的人员确认，给我们提出以服务号的形式可能可以解决我们的问题。我们打了许多客服电话，发现有一些客服并不是技术客服，基本上没有什么用处，根本联系不到什么腾讯方面的人。后来我们找到了微信小程序大赛中的腾讯方面的人员，我们最终确认了我们原来的方案是不可行的，是违反腾讯方面的规定的，但是杨老师所说的服务号的方案是可行的，于是我们更新了我们的可行性方案，决定用以服务号的方式继续我们的项目，幸好我们的组员可以借到一个公司注册的服务号供我们小组使用，最终我们小组化险为夷，避免了一错再错，也避免了过去所做的工作化为泡影。由此我们体会到软件开发过程中的每个环节都有它的道理，我们应该仔细地去落实，不懂的地方就应该主动寻求帮助，而不是任由问题存在而不去解决。

关于团队

我觉得我们的小组比较团结，大家都认真负责的完成自己的工作，也能共同为整个项目而奋斗，小组成员也能积极沟通，我想这和我们经常在一起工作有着密不可分的关系。积极沟通是整个团队最重要的一部分，我们积极交流关于项目的一些看法和问题，这使得我们的努力都是往一个方向的，不会南辕北辙。虽然我们没有出去team building过，但是我们经常一起吃食堂应该也算是一种team building吧。

希望下学期再一起共事。