

**Wink详细设计规约**

软件工程课程设计



**指导教师：**

**杜庆峰**

**小组成员：**

**1752026 李航**

**1652977 王星洲**

**1650350 乔宇**

**1753499 潘小逸**

目录

[修订历史 3](#_Toc29339639)

[1. 引言 4](#_Toc29339640)

[1.1 编写目的 4](#_Toc29339641)

[1.2 背景与依据 4](#_Toc29339642)

[1.3 参考资料 4](#_Toc29339643)

[2. 系统软件结构 4](#_Toc29339644)

[3. 浏览模块 4](#_Toc29339645)

[3.1 模块描述 4](#_Toc29339646)

[3.2 功能 4](#_Toc29339647)

[3.3 模块详细设计 5](#_Toc29339648)

[3.3.1 搜索新闻的详细设计 5](#_Toc29339649)

[3.3.2 搜索动态的详细设计 7](#_Toc29339650)

[3.3.3 查看所有动态的详细设计 8](#_Toc29339651)

[3.3.4 频道管理的详细设计 9](#_Toc29339652)

[3.3.5 切换频道的详细设计 10](#_Toc29339653)

[3.3.6增加/删除频道的详细设计 10](#_Toc29339654)

[3.3.7 查看新闻的详细设计 12](#_Toc29339655)

[3.3.8 查看新闻评论的详细设计 12](#_Toc29339656)

[4. 社交 模块 13](#_Toc29339657)

[4.1 模块描述 13](#_Toc29339658)

[4.2 功能 14](#_Toc29339659)

[4.3 模块详细设计 14](#_Toc29339660)

[4.3.1 查看关注/我的动态的详细设计 14](#_Toc29339661)

[4.3.2 评论动态的详细设计 16](#_Toc29339662)

[4.3.3 关注用户的详细设计 18](#_Toc29339663)

[4.3.4 取消关注的详细设计 20](#_Toc29339664)

[4.3.5 拉黑粉丝的详细设计 22](#_Toc29339665)

[4.3.6 评论新闻的详细设计 23](#_Toc29339666)

[4.3.7 发布动态的详细设计 25](#_Toc29339667)

[5. 个人收藏模块 26](#_Toc29339668)

[5.1 模块描述 26](#_Toc29339669)

[5.2 功能 26](#_Toc29339670)

[5.3 模块详细设计 26](#_Toc29339671)

[5.3.1 收藏新闻的详细设计 26](#_Toc29339672)

[5.3.2 查看收藏的详细设计 28](#_Toc29339673)

[5.3.3 取消收藏的详细设计 29](#_Toc29339674)

[6.账户管理模块 31](#_Toc29339675)

[6.1 模块描述 31](#_Toc29339676)

[6.2 功能 31](#_Toc29339677)

[6.3 模块详细设计 31](#_Toc29339678)

[6.3.1 注册的详细设计 31](#_Toc29339679)

[6.3.2 登录的详细设计 33](#_Toc29339680)

[6.3.3 修改昵称的详细设计 35](#_Toc29339681)

[6.3.4 修改密码的详细设计 37](#_Toc29339682)

# 修订历史

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编写日期 | SEPG | 版本 | 说明 | 作者 | 评审时间 | 评审人员 | 批准日期 | 签字人员 |
| 2019/11/12 |  | 1.0 | 初稿 | 李航 |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

# 1. 引言

## 1.1 编写目的

《详细设计书》是项目设计的第二个阶段，这个阶段的主要任务是在《概要设计书》基础上，对概要设计中产生的功能模块进行过程描述，设计功能模块的内部细节，包括类的详细逻辑和接口及类之间的调用关系，为编写源代码提供必要的说明。

本文档的目的旨在推动软件工程的规范化，使设计人员遵循统一的详细设计书写规范，节省制 作文档的时间，降低系统实现的风险，做到系统设计资料的规范性与全面性，以利于系统的实现、测试、维护、版本升级等。

概要设计解决了软件系统总体结构设计的问题，包括整个软件系统的结构、模块划分、模块功 能和模块间的联系等。详细设计则要解决如何实现各个模块的内部功能，即模块设计。具体的说，模块设计就是要详细描述各模块中包含的类的处理逻辑及其提供接口的参数格式、请求方式。但这并不等同于系统实现阶段用具体的语言编码，它只是对实现细节作精确的描述，这样编码阶段就可以将详细设计中对功能实现的描述，直接翻译、转化为用某种程序设计语言书写的程序。详细设计的详细程度应达到可以编写程序的水平。

## 1.2 背景与依据

本项目是一款新闻APP，用户可以查看新闻、视频，对新闻发表一些自己的评论。同时，本项目也有发表、查看动态功能，找到意向相投的好友，共同商讨自己的看法与生活态度。详细设计的依据来自先前的3个文档。

## 1.3 参考资料

[1] IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

[2] Software H. Example Use Case Specification[R/OL]. [2013-11-15]. http://www.hippo- software.co.uk/downloads/Example%20Use%20Case%20Specification.pdf.

[3]wink项目组需求规约.pdf

[4]wink项目组需求分析规约.pdf

[5]wink项目组概要设计规约.pdf

# 2. 系统软件结构

# 3. 浏览模块

## 3.1 模块描述

浏览模块包括了用户所有可以浏览的内容，包括新闻、动态、评论等。浏览模块的所有功能不需要登录即可使用。在浏览新闻时，可以选择不同频道的新闻进行查看，也可以切换、添加或删除频道。点击新闻标题可以查看新闻的详情和新闻评论。未登录用户的浏览动态只能查看所有人的动态。

## 3.2 功能

1）搜索新闻：用户可以通过关键词、时间、类型中的一个或多个条件搜索对应的新闻进行浏览。

2）搜索动态：用户可以通过关键词、时间、关注列表中的一个或多个条件搜索对应的动态进行浏览。

3）查看所有动态：用户可以在动态页面的名为“所有“的子页面查看到所有用户发布的动态。未登录的用户也可以查看到所有动态。

4）频道管理：用户可以切换/增加/删除频道来进行频道管理。

5）切换频道：用户可以通过点击新闻页面上方频道栏中的不同的频道名称来切换频道，并浏览相关新闻。

6）增加/删除频道：用户可以通过点击新闻页面上方的加号进入频道管理页面，app提供大量不同的频道分类供用户选择，用户可以将自己感兴趣的频道加入到频道栏中并对它们进行自由排序，也可以将不感兴趣的频道从频道栏中删除。

7）查看新闻：用户可以在新闻页面找到自己感兴趣的新闻，并点击进入新闻详情页面，下拉可以浏览新闻全部内容。

8）查看新闻评论：用户可以在新闻内容下方看到所有用户对于该新闻的评论。

## 3.3 模块详细设计

### 3.3.1 搜索新闻的详细设计

3.3.1.1 输入项

1）搜索新闻类别

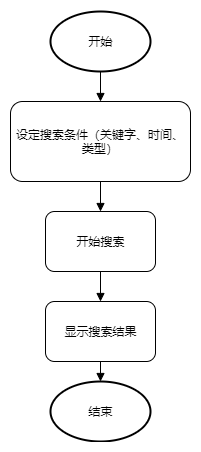
2）搜索新闻关键词

3）搜索新闻时间

3.3.1.2 输出项

1）新闻搜索结果的页面

3.3.1.3 逻辑流程



3.3.1.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 完成搜索新闻功能，支持关键词搜索，类型搜索，发布时间搜索 |
| URL | /searchNews |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| type | false | string | 支持类型：头条/社会/国内/国际/娱乐/体育/军事/科技/财经/时尚。 |
| keyword | false | string | 搜索关键词 |
| time | false | string | 新闻发布时间，满足”yyyy-MM-dd hh:mm:ss"的子字符串格式 |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success/error |
| data | list | 包含多个新闻的信息，每条新闻用一个json对象表示，每条新闻包含author,time,title,type,url,picture字段 |

3.3.1.5 限制条件与出错处理

1）限制所有搜索条件都为模式匹配的子字符串。

2）当搜索结果为空时，搜索结果页面显示空白。

### 3.3.2 搜索动态的详细设计

3.3.2.1 输入项

1）搜索动态所属好友

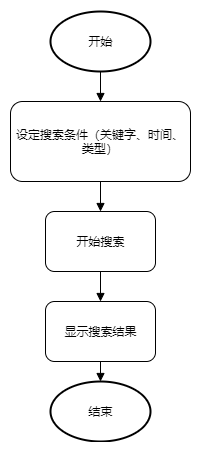
2）搜索动态关键词

3）搜索动态时间

3.3.2.2 输出项

1）动态搜索结果的页面

3.3.2.3 逻辑流程



3.3.2.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 完成搜索动态功能，支持作者昵称搜索，关键字搜索，发布时间搜索 |
| URL | /searchTrend |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| friend | false | string | 好友昵称 |
| keyword | false | string | 搜索关键字 |
| time | false | string | 时间 满足”yyyy-MM-dd hh:mm:ss"的子字符串格式 |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success，搜索成功。error，搜索失败，没有搜索结果 |
| data | list | 动态的list列表。每条动态包括id，content，time，comment\_num，news\_id字段 |

3.3.2.5 限制条件与出错处理

1）限制所有搜索条件都为模式匹配的子字符串。

2）当搜索结果为空时，搜索结果页面显示空白。

### 3.3.3 查看所有动态的详细设计

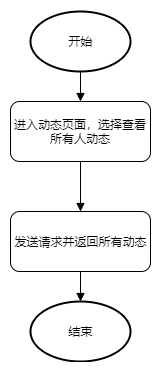
3.3.3.1 输入项

无

3.3.3.2 输出项

1）显示所有动态的页面

3.3.3.3 逻辑流程



3.3.3.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 从数据库中获取所有的动态记录 |
| URL | /getAllTrend |
| 支持格式 | JSON |
| HTTP请求方式 | GET |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| 无 |  |  |  |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success/error |
| data | list | 包含所有动态的具体信息，每条动态信息用一个json对象表示，每条信息包含动态id(String ID)，作者名(String nick\_name)，作者id(String author\_id)，发布时间(String relaese\_time)，动态内容(String content)，关联新闻ID(String news\_id)，新闻title(String title)，新闻url(String url)字段，按照release\_time升序排列 |

### 3.3.4 频道管理的详细设计

频道管理功能具体包括切换频道与增加/删除频道，详见3.3.5和3.3.6

### 3.3.5 切换频道的详细设计

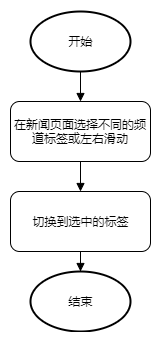
3.3.5.1 输入项

1)用户选择的新闻频道

3.3.5.2 输出项

2）所选频道的新闻列表

3.3.5.3 逻辑流程



### 3.3.6增加/删除频道的详细设计

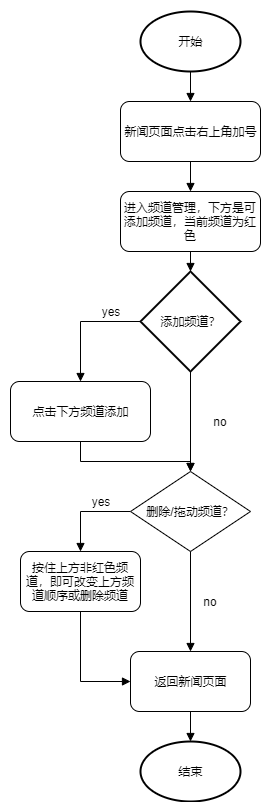
3.3.6.1 输入项

1）用户进行的频道增删等改动操作

3.3.6.2 输出项

1）修改后的频道列表

3.3.6.3 逻辑流程



### 3.3.7 查看新闻的详细设计

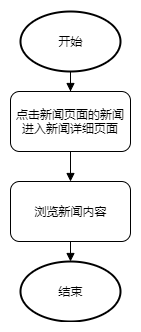
3.3.7.1 输入项

1）用户所选择的新闻ID

3.3.7.2 输出项

1）显示新闻详情的标题，作者，时间和详细内容的页面

3.3.7.3 逻辑流程



### 3.3.8 查看新闻评论的详细设计

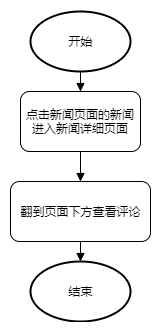
3.3.8.1 输入项

1）用户所选择的新闻ID

3.3.8.2 输出项

1）显示新闻的评论列表，包括评论作者，时间，评论内容

3.3.8.3 逻辑流程



3.3.8.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 获取数据库中新闻的所有评论 |
| URL | /getNewsAllComment |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| news\_id | true | string | 需要获取所有评论的新闻ID |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success(有评论)/error(没有评论) |
| data | list | 包含该新闻的所有评论，每条评论的信息包含评论作者名(String nick\_name)，作者id(String user\_id)，评论内容(String content)，评论时间(String release\_time) |

3.3.8.5 限制条件与出错处理

1）某条新闻无评论时，评论列表为空。

# 4. 社交 模块

## 4.1 模块描述

社交模块包括了用户和其他用户互动的所有功能，例如查看自己或好友的动态，评论好友动态，查看自己的粉丝和关注的用户，关注或取消关注用户，拉黑粉丝，评论新闻，发表动态等。尽已登录的用户可以使用社交模块的功能。

## 4.2 功能

 1）查看关注/我的动态：已登录的用户可以在动态页面顶部的动态栏中选择查看自己的动态以及关注的用户发布的动态，若用户未登录则需先进行登录。

 2）评论动态：若用户关注了动态的发布者，则用户可以在该动态的详细页面中评论该动态。

 3）关注用户：用户可以在动态大厅以及新闻评论中选择自己感兴趣的用户进行关注，关注之后用户成为被关注用户的粉丝。

 4）取消关注：如果用户对关注的用户不再感兴趣，或者不想看到该用户发布的动态，可以选择在关注界面取消关注。

 5）拉黑粉丝：用户被粉丝恶意评论后，可以选择解除该粉丝的关注。

 6）评论新闻：用户可以在新闻详情页面评论新闻，用户评论某条新闻时，系统将自动将该条评论作为动态发布。该动态还包括该新闻的链接。

 7）发布动态：用户可以选择发布一条纯文字内容的动态。

## 4.3 模块详细设计

### 4.3.1 查看关注/我的动态的详细设计

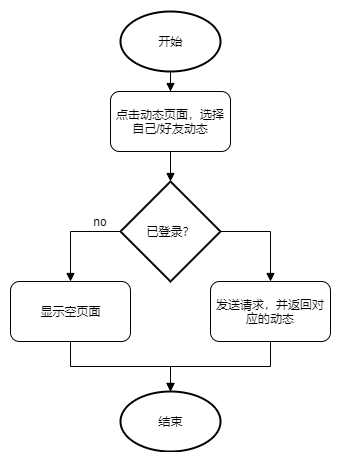
4.3.1.1 输入项

1）用户ID

4.3.1.2 输出项

1）包括该用户自己和所有其关注用户的动态页面

4.3.1.3 逻辑流程



4.3.1.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 从数据库中获取某个用户关注的所有用户发布的动态 |
| URL | /getFriendTrend |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| ID | true | string | 需要获取关注用户发布的动态的用户的ID |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success/error |
| data | list | 包含所有关注用户的动态的具体信息，每条动态信息用一个json对象表示，每条信息包含动态id(String ID)，作者名(String nick\_name)，作者id(String author\_id)，发布时间(String relaese\_time)，动态内容(String content)，关联新闻ID(String news\_id)，新闻title(String title)，新闻url(String url)字段，按照release\_time升序排列 |

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 从数据库中获取某一用户自己发布的所有的动态 |
| URL | /getMyTrend |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| ID | true | string | 需要获取自己动态的用户的ID |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success/error |
| data | list | 包含自己的动态的具体信息，每条动态信息用一个json对象表示，每条信息包含动态id(String ID)，作者名(String nick\_name)，作者id(String author\_id)，发布时间(String relaese\_time)，动态内容(String content)，关联新闻ID(String news\_id)，新闻title(String title)，新闻url(String url)字段，按照release\_time升序排列 |

4.3.1.5 限制条件与出错处理

1）限制用户在查看动态前应登录账号

2）如果用户未登录，则动态页面显示为空

### 4.3.2 评论动态的详细设计

4.3.2.1 输入项

1）用户ID

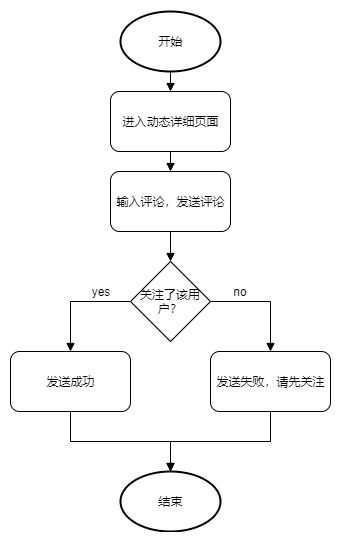
2）评论的动态ID

3）评论的详细内容

4.3.2.2 输出项

1）评论成功的示信息

4.3.2.3 逻辑流程



4.3.2.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 向数据库中插入评论动态的相关数据 |
| URL | /insertTrendComment |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| user\_id | true | string | 需要发布动态的user\_id |
| comment | true | string | 需要发布的动态内容 |
| trend\_id | true | string | 要评论的动态id |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success(发布动态评论成功)/insert error(动态评论发布失败) |

4.3.2.5 限制条件与出错处理

1）用户应当在登录后打开动态并对动态进行评论

2）如果用户未登录，则显示请先登录的提示信息

### 4.3.3 关注用户的详细设计

4.3.3.1 输入项

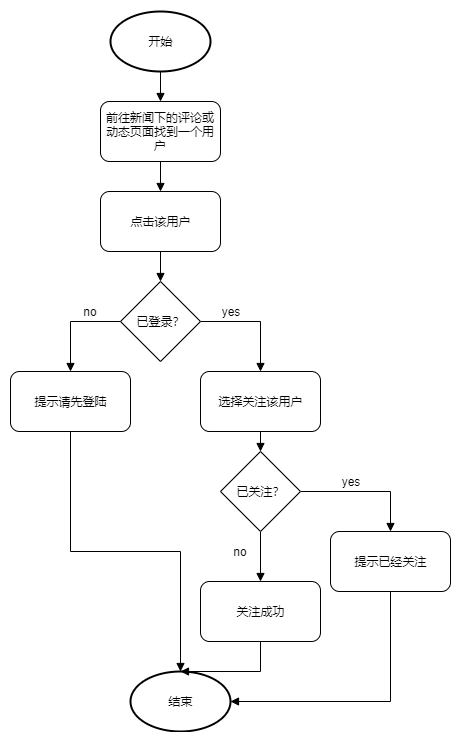
1）用户ID

2）关注用户ID

4.3.3.2 输出项

1）关注成功的提示信息

4.3.3.3 逻辑流程



4.3.3.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 向数据库中插入一个关注关系元组 |
| URL | /insertInterest |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| user\_id | true | string | 需要添加关注的用户ID |
| friend\_id | true | string | 被关注的用户ID |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success代表添加成功/error代表添加失败 |

4.3.3.5 限制条件与出错处理

1）用户应当在登录后进行关注操作

2）用户不能重复关注同一个用户

3）如果用户未登录，显示请先登录的提示信息

4）如果用户尝试关注一个已关注用户，显示已关注，不要重复点击的提示信息

### 4.3.4 取消关注的详细设计

4.3.4.1 输入项

1）用户ID

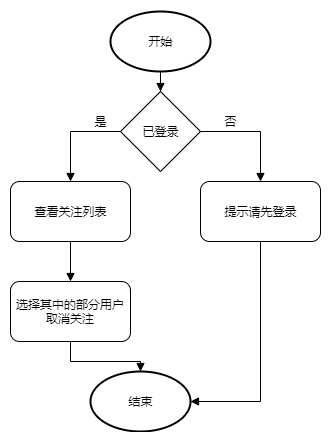
2）取消关注用户ID

4.3.4.2 输出项

1）取消关注成功的提示信息

2）刷新后的关注列表页面

4.3.4.3 逻辑流程



4.3.4.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 从数据库中删除一条关注关系元组 |
| URL | /deleteFriend |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| user\_id | true | string | 需要取关的用户ID |
| friend\_id | true | string | 被取关的好友ID |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success代表取关成功/error代表取关失败无好友关系 |

4.3.4.5 限制条件与出错处理

1）用户应当在登录后在关注列表中取消关注

2）如果用户未登录，无法打开关注列表

### 4.3.5 拉黑粉丝的详细设计

4.3.5.1 输入项

1）用户ID

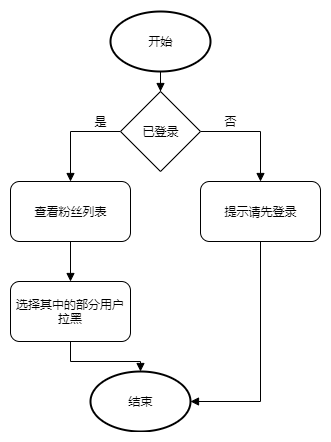
2）拉黑粉丝ID

4.3.5.2 输出项

1）拉黑成功的提示信息

2）刷新后的粉丝列表页面

4.3.5.3 逻辑流程



4.3.5.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 从数据库中删除一条关系元组 |
| URL | /cancelInterest |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| user\_id | true | string | 之前关注发起“拉黑粉丝”动作的用户ID |
| friend\_id | true | string | 被关注的用户ID，发起“拉黑粉丝”动作的用户 |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success代表取消被关注成功/error代表取消被关注失败，无好友关系 |

4.3.5.5 限制条件与出错处理

1）用户应当在登录后在粉丝列表中拉黑粉丝

2）如果用户未登录，无法打开粉丝列表

### 4.3.6 评论新闻的详细设计

4.3.6.1 输入项

1）用户ID

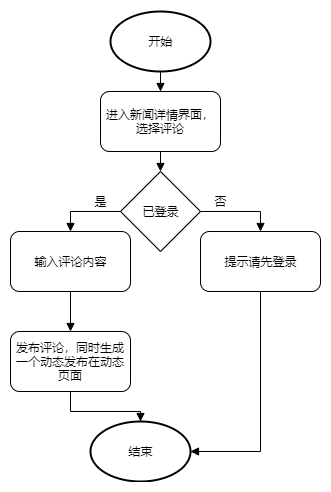
2）评论新闻ID

3）评论的详细内容

4.3.6.2 输出项

1）评论成功的提示信息

4.3.6.3 逻辑流程



4.3.6.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 向数据库中插入一个动态发布元组 |
| URL | /releaseTrend |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| user\_id | true | string | 需要发布动态的user\_id |
| content | true | string | 需要发布的动态内容 |
| news\_id | true | string | 相关的新闻id |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success(发布动态成功)/insert trend error(插入动态表失败)/release trend error(插入发布动态失败) |

4.3.6.5 限制条件与出错处理

1）用户应当在登录后打开新闻并对新闻进行评论

2）如果用户未登录，则显示请先登录的提示信息

### 4.3.7 发布动态的详细设计

4.3.7.1 输入项

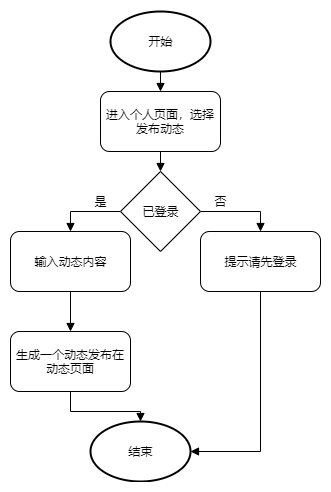
1）用户ID

2）动态的详细内容

4.3.7.2 输出项

1）发表成功的提示信息

4.3.7.3 逻辑流程



4.3.7.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 向数据库中插入一个动态发布元组 |
| URL | /releaseTrend |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| user\_id | true | string | 需要发布动态的user\_id |
| content | true | string | 需要发布的动态内容 |
| news\_id | true | string | 相关的新闻id |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success(发布动态成功)/insert trend error(插入动态表失败)/release trend error(插入发布动态失败) |

4.3.7.5 限制条件与出错处理

1）用户应在登录后发表动态

2）如果用户未登录，显示请先登录的的提示信息

# 5. 个人收藏模块

## 5.1 模块描述

个人收藏模块包括用户对自己感兴趣的内容进行收藏以便后续可以便捷的查看已收藏内容的功能。用户可以在浏览一条新闻时对该新闻进行收藏，可以浏览自己收藏的新闻的列表，并对其中某条新闻取消收藏。

## 5.2 功能

 1）收藏新闻：用户可以在新闻详情页面收藏自己感兴趣的新闻，以便下次更快的找到这条新闻。

 2）查看收藏：用户可以查看自己的新闻收藏列表。

 3）取消收藏：用户可以将已收藏的新闻取消收藏。

## 5.3 模块详细设计

### 5.3.1 收藏新闻的详细设计

5.3.1.1 输入项

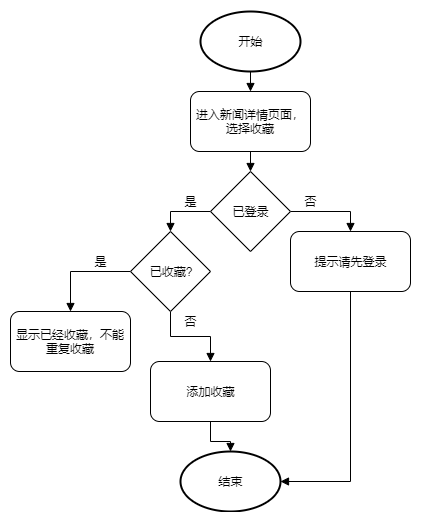
1）用户ID

2）收藏新闻ID

5.3.1.2 输出项

1）收藏成功的提示信息

5.3.1.3 逻辑流程



5.3.1.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 添加、查询、取消收藏 |
| URL | /collect |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| key | true | string | add:添加收藏，query：查询是否收藏，delete：取消收藏 |
| user | true | string | 用户id |
| news | true | string | 新闻id |

**返回字段：**

返回一个String字符串，对于add和delete结果始终为True，对于query，True表示已收藏，False表示未收藏。对于其他key请求，返回BadRequest。

5.3.1.5 限制条件与出错处理

1）用户应在登录后收藏新闻

2）用户不可以重复收藏同一条新闻

3）如果用户未登录，显示请先登录的提示信息

4）如果用户尝试收藏同一条新闻，显示已收藏，请不要重复点击的提示信息

### 5.3.2 查看收藏的详细设计

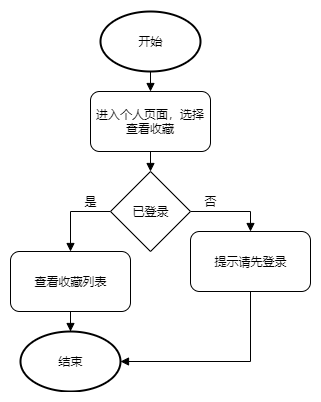
5.3.2.1 输入项

1）用户ID

5.3.2.2 输出项

1）包括用户收藏列表的页面

5.3.2.3 逻辑流程



5.3.2.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 添加、查询、取消收藏 |
| URL | /collect |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| key | true | string | add:添加收藏，query：查询是否收藏，delete：取消收藏 |
| user | true | string | 用户id |
| news | true | string | 新闻id |

**返回字段：**

返回一个String字符串，对于add和delete结果始终为True，对于query，True表示已收藏，False表示未收藏。对于其他key请求，返回BadRequest。

5.3.2.5 限制条件与出错处理

1）用户应在登录后查看收藏

2）如果用户未登录，显示请先登录的提示信息

### 5.3.3 取消收藏的详细设计

5.3.3.1 输入项

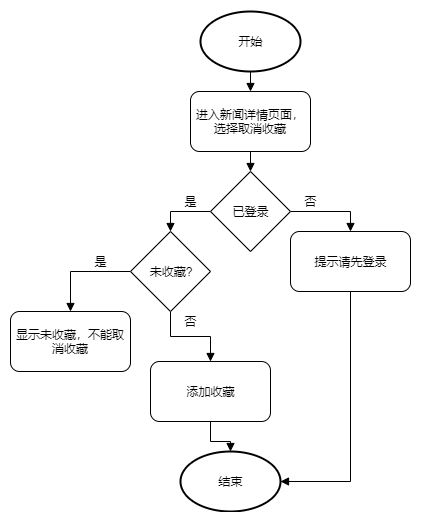
1）用户ID

2）取消收藏的新闻ID

5.3.3.2 输出项

1）取消收藏成功的提示信息

5.3.3.3 逻辑流程



5.3.3.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 添加、查询、取消收藏 |
| URL | /collect |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| key | true | string | add:添加收藏，query：查询是否收藏，delete：取消收藏 |
| user | true | string | 用户id |
| news | true | string | 新闻id |

**返回字段：**

返回一个String字符串，对于add和delete结果始终为True，对于query，True表示已收藏，False表示未收藏。对于其他key请求，返回BadRequest。

5.3.3.5 限制条件与出错处理

1）用户应在登录后取消收藏

2）用户不能取消收藏未收藏的新闻

3）如果用户未登录，显示请先登录的提示信息

4）如果用户尝试取消收藏一条未收藏的新闻，显示未收藏，不可取消收藏的提示信息

# 6.账户管理模块

## 6.1 模块描述

账户管理模块包括用户权限的管理功能，如基础的注册登录功能，以及修改个人昵称和密码的功能。

## 6.2 功能

 1）注册：用户可以选择注册一个个人账号。

2）登陆后在app上与其他用户进行互动。

 3）修改昵称：登录之后用户可以修改昵称，此昵称将作为登录账号凭证，并且是唯一的。

 4）修改密码：用户可以选择修改密码，首先输入当前密码再将新密码输入两次，两次密码相同则可以进行密码修改。

## 6.3 模块详细设计

### 6.3.1 注册的详细设计

6.3.1.1 输入项

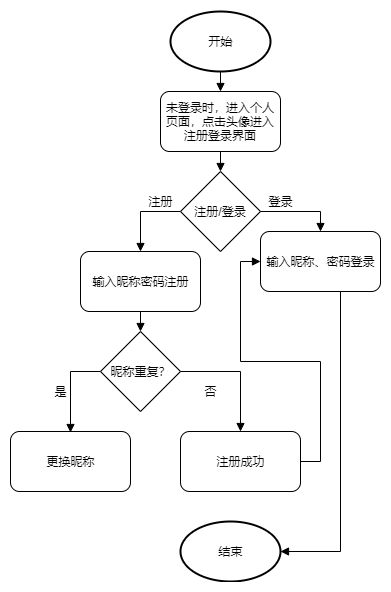
1）注册用户昵称

2）注册用户密码

6.3.1.2 输出项

1）注册成功，请登录的提示信息

6.3.1.3 逻辑流程



6.3.1.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 完成注册功能 |
| URL | /register |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| id | true | string | 用户名 |
| passowrd | true | string | 密码 |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success/error， 错误的唯一情况是注册用户名重复 |

6.3.1.5 限制条件与出错处理

1）注册用户昵称不能在数据库中有重复昵称

2）如果用户尝试使用重复的注册昵称，显示昵称重复的提示信息

### 6.3.2 登录的详细设计

6.3.2.1 输入项

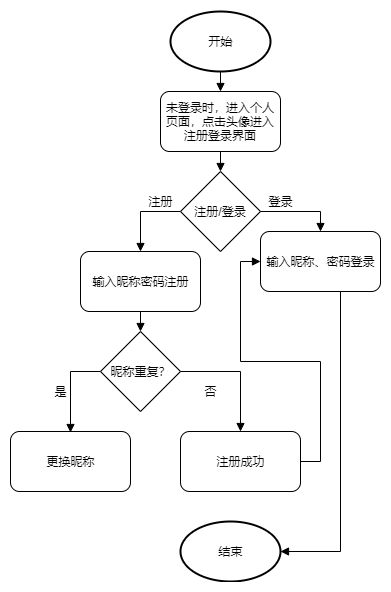
1）登录用户昵称

2）登录用户密码

6.3.2.2 输出项

1）新闻中心的显示页面

6.3.2.3 逻辑流程



6.3.2.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 进行用户登录 |
| URL | /login |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| id | true | string | 用户名 |
| password | true | string | 密码 |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success/error |
| id | string | 用户在数据库中的唯一标识id |

6.3.2.5 限制条件与出错处理

1）登录用户的昵称和密码应当匹配

2）如果用户输入的昵称和密码不匹配，显示用户名或密码错误的提示信息

### 6.3.3 修改昵称的详细设计

6.3.3.1 输入项

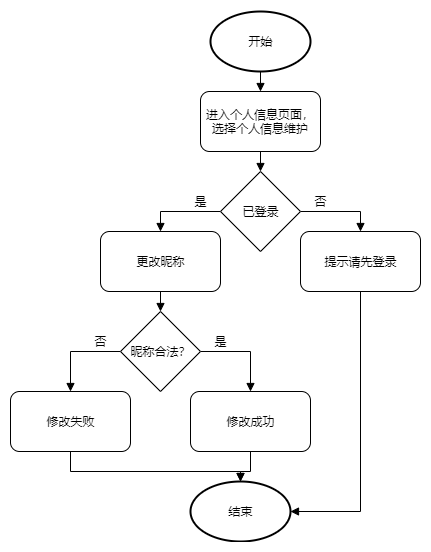
1）用户ID

2）修改的用户昵称

6.3.3.2 输出项

1）修改成功的提示信息

6.3.3.3 逻辑流程



6.3.3.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 完成用户修改昵称功能 |
| URL | /modifyName |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| user | true | string | 用户id |
| nick\_name | true | string | 新昵称 |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success。当前没有执行规则检查。应当放在前端验证。 |

6.3.3.5 限制条件与出错处理

1）修改的用户昵称不能在数据库中有重复昵称

2）如果用户尝试使用一个重复的昵称，显示昵称重复的提示信息

### 6.3.4 修改密码的详细设计

6.3.4.1 输入项

1）用户ID

2）用户的原始密码

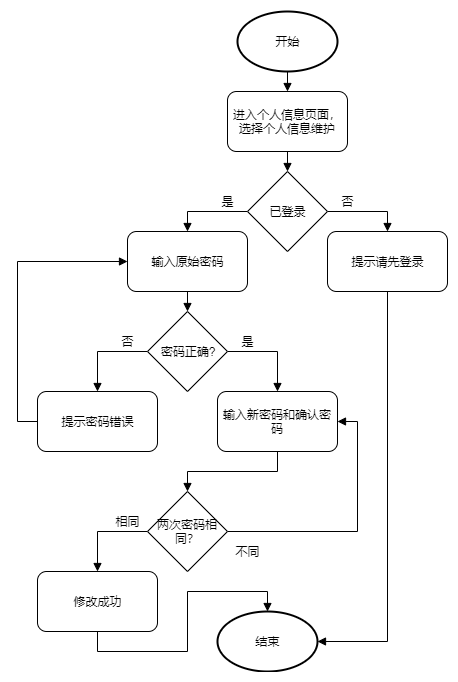
3）修改的用户密码

4）二次输入的修改的密码

6.3.4.2 输出项

1）显示修改成功的提示信息，并自动退出登陆状态，返回新闻中心

6.3.4.3 逻辑流程



6.3.4.4 接口

**基本信息：**

|  |  |
| --- | --- |
| 功能描述 | 完成修改密码功能 |
| URL | /modifyPassword |
| 支持格式 | JSON |
| HTTP请求方式 | POST |

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| user | true | string | 用户id |
| password | true | string | 新密码 |

**返回字段：**

|  |  |  |
| --- | --- | --- |
| 返回字段 | 字段类型 | 说明 |
| status | string | 返回结果状态。success。当前没有执行规则检查。应当放在前端验证。 |

6.3.4.5 限制条件与出错处理

1）用户应当先正确输入原始密码后修改密码

2）用户应当保证两次输入的密码相同

3）如果用户输入了错误的原始密码，显示密码错误的提示信息

4）如果用户两次输入的修改密码不一致，显示两次输入密码不一致的提示信息