

本科毕业论文（设计）

概要设计说明书

|  |  |
| --- | --- |
| **学 生 姓 名** | **谢仁义** |
| **学号** | **2016081187** |
| **专业** | **软件工程** |
| **年级班级** | **2016级 5 班** |
| **指导教师** | **王嘉昀（讲师）** |
| **所在学院** | **软件工程学院** |
| **提交日期** | **2020年4月15日** |

2020 年 4 月

成都信息工程大学 软件工程学院

目录

[1 引言 - 1 -](#_Toc37711092)

[1.1 编写目的 - 1 -](#_Toc37711093)

[1.2 背景 - 1 -](#_Toc37711094)

[1.3 术语 - 1 -](#_Toc37711095)

[1.4 参考资料 - 2 -](#_Toc37711096)

[2 总体设计 - 3 -](#_Toc37711097)

[2.1 系统体系结构 - 3 -](#_Toc37711098)

[2.2 系统功能结构 - 3 -](#_Toc37711099)

[2.3 运行环境 - 4 -](#_Toc37711100)

[2.3.1 硬件环境 - 4 -](#_Toc37711101)

[2.3.2 软件环境 - 4 -](#_Toc37711102)

[2.4 系统的关键技术 - 5 -](#_Toc37711103)

[3 功能模块设计说明 - 5 -](#_Toc37711104)

[3.1 功能模块列表 - 5 -](#_Toc37711105)

[3.2 用户信息管理： - 5 -](#_Toc37711106)

[3.2.1 模块编号和功能描述 - 5 -](#_Toc37711107)

[3.2.2 操作者 - 6 -](#_Toc37711108)

[3.2.3 与本模块相关的码表和表 - 6 -](#_Toc37711109)

[3.2.4 界面设计与说明 - 6 -](#_Toc37711110)

[3.2.5 输入信息 - 6 -](#_Toc37711111)

[3.2.6 算法 - 7 -](#_Toc37711112)

[3.2.7 处理流程 - 7 -](#_Toc37711113)

[3.2.8 类设计 - 8 -](#_Toc37711114)

[3.3 公告信息管理： - 9 -](#_Toc37711115)

[3.3.1 模块编号和功能描述 - 9 -](#_Toc37711116)

[3.3.2 操作者 - 9 -](#_Toc37711117)

[3.3.3 与本模块相关的码表和表 - 9 -](#_Toc37711118)

[3.3.4 界面设计与说明 - 9 -](#_Toc37711119)

[3.3.5 输入信息 - 10 -](#_Toc37711120)

[3.3.6 算法 - 10 -](#_Toc37711121)

[3.3.7 处理流程 - 11 -](#_Toc37711122)

[3.3.8 类设计 - 12 -](#_Toc37711123)

[3.4 新闻信息管理： - 13 -](#_Toc37711124)

[3.4.1 模块编号和功能描述 - 13 -](#_Toc37711125)

[3.4.2 操作者 - 13 -](#_Toc37711126)

[3.4.3 与本模块相关的码表和表 - 13 -](#_Toc37711127)

[3.4.4 界面设计与说明 - 13 -](#_Toc37711128)

[3.4.5 输入信息 - 14 -](#_Toc37711129)

[3.4.6 算法 - 14 -](#_Toc37711130)

[3.4.7 处理流程 - 14 -](#_Toc37711131)

[3.4.8 类设计 - 15 -](#_Toc37711132)

[3.5 讨论区信息管理： - 16 -](#_Toc37711133)

[3.5.1 模块编号和功能描述 - 16 -](#_Toc37711134)

[3.5.2 操作者 - 16 -](#_Toc37711135)

[3.5.3 与本模块相关的码表和表 - 16 -](#_Toc37711136)

[3.5.4 界面设计与说明 - 16 -](#_Toc37711137)

[3.5.5 输入信息 - 17 -](#_Toc37711138)

[3.5.6 算法 - 17 -](#_Toc37711139)

[3.5.7 处理流程 - 18 -](#_Toc37711140)

[3.5.8 类设计 - 19 -](#_Toc37711141)

[3.6 评论管理： - 20 -](#_Toc37711142)

[3.6.1 模块编号和功能描述 - 20 -](#_Toc37711143)

[3.6.2 操作者 - 20 -](#_Toc37711144)

[3.6.3 与本模块相关的码表和表 - 20 -](#_Toc37711145)

[3.6.4 界面设计与说明 - 20 -](#_Toc37711146)

[3.6.5 输入信息 - 21 -](#_Toc37711147)

[3.6.6 算法 - 21 -](#_Toc37711148)

[3.6.7 处理流程 - 21 -](#_Toc37711149)

[3.6.8 类设计 - 22 -](#_Toc37711150)

[3.7 值班管理： - 23 -](#_Toc37711151)

[3.7.1 模块编号和功能描述 - 23 -](#_Toc37711152)

[3.7.2 操作者 - 23 -](#_Toc37711153)

[3.7.3 与本模块相关的码表和表 - 23 -](#_Toc37711154)

[3.7.4 界面设计与说明 - 23 -](#_Toc37711155)

[3.7.5 输入信息 - 23 -](#_Toc37711156)

[3.7.6 算法 - 24 -](#_Toc37711157)

[3.7.7 处理流程 - 24 -](#_Toc37711158)

[3.7.8 类设计 - 25 -](#_Toc37711159)

[3.8 反馈管理： - 26 -](#_Toc37711160)

[3.8.1 模块编号和功能描述 - 26 -](#_Toc37711161)

[3.8.2 操作者 - 26 -](#_Toc37711162)

[3.8.3 与本模块相关的码表和表 - 26 -](#_Toc37711163)

[3.8.4 界面设计与说明 - 26 -](#_Toc37711164)

[3.8.5 输入信息 - 26 -](#_Toc37711165)

[3.8.6 算法 - 27 -](#_Toc37711166)

[3.8.7 处理流程 - 27 -](#_Toc37711167)

[3.8.8 类设计 - 28 -](#_Toc37711168)

[3.9 友链管理： - 29 -](#_Toc37711169)

[3.9.1 模块编号和功能描述 - 29 -](#_Toc37711170)

[3.9.2 操作者 - 29 -](#_Toc37711171)

[3.9.3 与本模块相关的码表和表 - 29 -](#_Toc37711172)

[3.9.4 界面设计与说明 - 29 -](#_Toc37711173)

[3.9.5 输入信息 - 29 -](#_Toc37711174)

[3.9.6 算法 - 30 -](#_Toc37711175)

[3.9.7 处理流程 - 30 -](#_Toc37711176)

[3.9.8 类设计 - 31 -](#_Toc37711177)

[3.10 举报管理： - 32 -](#_Toc37711178)

[3.10.1 模块编号和功能描述 - 32 -](#_Toc37711179)

[3.10.2 操作者 - 32 -](#_Toc37711180)

[3.10.3 与本模块相关的码表和表 - 32 -](#_Toc37711181)

[3.10.4 界面设计与说明 - 32 -](#_Toc37711182)

[3.10.5 输入信息 - 32 -](#_Toc37711183)

[3.10.6 算法 - 33 -](#_Toc37711184)

[3.10.7 处理流程 - 33 -](#_Toc37711185)

[3.10.8 类设计 - 34 -](#_Toc37711186)

[3.11 报名管理： - 35 -](#_Toc37711187)

[3.11.1 模块编号和功能描述 - 35 -](#_Toc37711188)

[3.11.2 操作者 - 35 -](#_Toc37711189)

[3.11.3 与本模块相关的码表和表 - 35 -](#_Toc37711190)

[3.11.4 界面设计与说明 - 35 -](#_Toc37711191)

[3.11.5 输入信息 - 35 -](#_Toc37711192)

[3.11.6 算法 - 36 -](#_Toc37711193)

[3.11.7 处理流程 - 36 -](#_Toc37711194)

[3.11.8 类设计 - 37 -](#_Toc37711195)

[4 内部接口设计 - 38 -](#_Toc37711196)

[4.1 UserMapper接口 - 38 -](#_Toc37711197)

[4.2 AnnouncementMapper接口 - 39 -](#_Toc37711198)

[4.3 NewsMapper接口 - 40 -](#_Toc37711199)

[4.4 PostMapper接口 - 41 -](#_Toc37711200)

[4.5 CommentMapper接口 - 42 -](#_Toc37711201)

[4.6 OnDutyMapper接口 - 43 -](#_Toc37711202)

[4.7 FeedbackMapper接口 - 44 -](#_Toc37711203)

[4.8 FriendUrlMapper接口 - 45 -](#_Toc37711204)

[4.9 ReportMapper接口 - 46 -](#_Toc37711205)

[4.10 RegisterMapper接口 - 47 -](#_Toc37711206)

[5 系统出错处理设计 - 48 -](#_Toc37711207)

[5.1 出错信息 - 48 -](#_Toc37711208)

[5.2 补救措施 - 49 -](#_Toc37711209)

# 引言

## 编写目的

本文档的编写是为了完善CUIT-ACM微信公众号的详细开发设计包括系统总体设计、功能模块详细设计、接口详细设计。使开发人员对系统开发有一个详细的指导。以求在最短的时间高效的开发出CUIT-ACM微信公众号。

## 背景

随着微信小程序和公众号的快速发展，它们已经是日常生活中最常见的程序了，并且访问也非常方便只需要一个手机就行。目前微信小程序/公众号开发逐渐发展成一类标准。再结合目前高校中被越来越重视的 ACM竞赛，但是我校并没有一个用于来宣传该类竞赛的相关小程序/公众号，为了扩大这一类竞赛的影响力，有了设计CUIT-ACM小程序/公众号的想法。这一类软件目前很多高校都有，比如比较熟悉的川大和电子科大。再次结合我所掌握的技术，打算自己实现前后端的所有技术点。再次深入分析小程序和公众号的区别后，结合自己目前的状况，最终决定开发软件名字为CUIT-ACM微信公众号。用户群体为实验室学生和广大想参加该项竞赛的学生.

## 术语

表-1术语和缩略语

|  |  |
| --- | --- |
| **术语、缩略语** | **解 释** |
| CUIT | Chengdu University of Information Technology成都信息工程大学 |
| ACM | 全称为ACM国际大学生程序设计竞赛(也称ACM/ICPC), 之前是由国际计算机协会(ACM), 于2019年更换赞助商为JetBrains, 不过由于之前40多年都是ACM赞助, 所以现在圈内还是以ACM称谓. |
| MySQL | 属于甲骨文公司旗下子公司的一款关系数据库管理系统 |
| SSM | Spring, SpringMVC, Mybatis三大框架的简称 |
| HTML | 超文本标记语言 |
| JS | JavaScript的简称, 是解释型或即时编译型的编程语言 |
| React | 用于构建用户界面的JavaScript库 |
| ES6 | ECMAScript6是于2015年6月正式发布的JavaScript语言的标准 |
| CSS | Cascading Style Sheets 层叠样式表 |
| LESS | CSS 预处理语言 |
| MVC | 是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范 |
| Ant Design | 由蚂蚁金服公司开发的一套基于React框架的UI组件 |

。

## 参考资料

1. 马素霞. 计算机应用系统的设计与实现[M]. 北京: 清华大学出版社, 2009.
2. 曾探. JavaScript设计模式与开发实践. 人民邮电出版社, 2015.

# 总体设计

## 系统体系结构

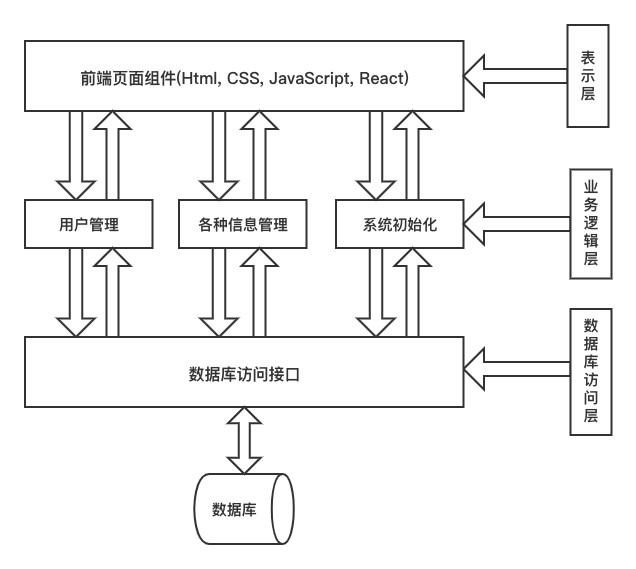


图2-1 系统体系结构图

## 系统功能结构

|  |
| --- |
| 图2-2系统功能结构图 |

## 运行环境

### 硬件环境

1. 服务器：CPU：双核及以上；内存：2G及以上；硬盘：120G及以上。
2. WebPC端：CPU：双核及以上；内存：2G及以上；硬盘：120G及以上。
3. 移动端: 能运行微信就行.

### 软件环境

1. 操作系统：

MacOS 10.11及以上。

1. 数据库系统：

Mysql数据库, redis数据库。

1. 开发平台及工具：

IntelliJ IDEA 2019.2。

Visual Studio Code。

1. 通信协议：

HTTP协议。

1. 其他软件：

iTerm, Office365。

## 系统的关键技术

本微信公众号是基于后端SSM框架, 前端React框架构建的一套完成的系统, 并且做到了前后端分离开发, 关键技术自然就是对SSM三大框架的掌握和对React的理解以及能编写相应的高效代码. 其中后端框架中最关键的就是要做到MVC的分层显示, 其中在M层中要编写高质量的SQL代码. 其次是如何利用React搭建一套完整的前端框架, 其中的ES6语言以及对CSS进行预处理的语言LESS的都是需要掌握的. 除此之外, 还需要优美的UI框架使界面更加美观, 选用的是Ant Design.

对于上述提到的关键技术, 首先需要一定的时间去学习掌握, 然后运用比较简单灵活性较高的方法来实现，尽量做到对浏览器的兼容，页面能完整的展示，用户层和数据层都能完整的衔接上，能正确的处理数据，用户也能得到真实的数据.

# 功能模块设计说明

## 功能模块列表

表3‑1 功能模块列表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块**  **编号** | **模块**  **名称** | **对应需求**  **功能编号** | **所对应**  **需求功能** | **实现**  **优先级** |
| XX\_YHXXGL01 | 用户信息管理 | MM\_YHXXGL01 | 用户信息管理 | 高 |
| XX\_GGXXGL02 | 公告信息管理 | MM\_GKXXGL02 | 公告信息管理 | 高 |
| XX\_XWXXGL03 | 新闻信息管理 | MM\_XWXXGL03 | 新闻信息管理 | 高 |
| XX\_TLQXXGL04 | 讨论区管理 | MM\_TLQXXGL04 | 讨论区管理 | 高 |
| XX\_PLGL05 | 评论管理 | MM\_PLGL05 | 评论管理 | 高 |
| XX\_ZBGL06 | 值班管理 | MM\_ZBGL06 | 值班管理 | 中 |
| XX\_FKGL07 | 反馈管理 | MM\_FKGL07 | 反馈管理 | 中 |
| XX\_YLGL08 | 友链管理 | MM\_YLGL08 | 友链管理 | 中 |
| XX\_JBGL09 | 举报管理 | MM\_JBGL09 | 举报管理 | 中 |
| XX\_BMGL10 | 报名管理 | MM\_BMGL10 | 报名管理 | 中 |

## 用户信息管理：

### 模块编号和功能描述

* 模块编号：MN\_YHXXGL01。
* 功能描述：对在公众号注册的人员进行管理, 审核, 以及提供一些基本的查询、修改等功能。

### 操作者

管理员可进行本功能模块所有操作。

### 与本模块相关的码表和表

表3‑2 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_USER | 用户信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供表信息进行查看用户的基本信息, 点击姓名后能查看某一位用户的具体信息. 可以在筛选框中填入信息进行对用户的筛选,提供基本的修改,删除功能键。

### 输入信息

|  |  |
| --- | --- |
| 表3-3用户信息输入输出信息表 | |
| **输入** | 查询、修改、删除时输入：用户编号（bigint） |
| 信息导入和修改时输入：用户名（varchar（50））、密码（varchar（50））、头像（varchar（255））、学号（varchar（30））、年级（int）、电话（varchar（20））、性别（int）、真实姓名(varchar(50)) |
| **输出** | 查询操作输出：用户名（varchar（50））、头像（varchar（255））、学号（varchar（30））、年级（int）、电话（varchar（20））、性别（int）、注册时间(datetime)、真实姓名(varchar(50))、用户身份(int) |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

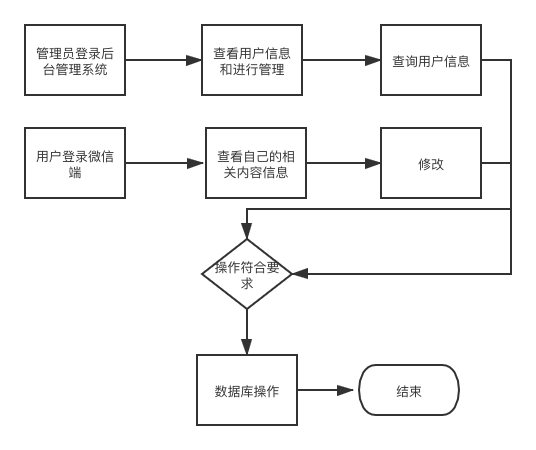


图3-1用户信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑2 用户信息管理类图

#### 类说明

（一）User类说明：

1. 功能：实现用户类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

（二）UserServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

1. 主要方法：

Add 方法实现用户信息的增加

Delete 方法实现用户信息的删除

Update 方法实现用户信息的修改

Select方法实现用户信息的查询

1. UserController类说明：
2. 功能：作为控制层负责接收前端发来的请求, 并获取参数
3. 主要方法：

Login方法实现用户的登录

GetLoginUser方法实现获取当前登录用户信息

GetMypPost方法实现获取当前登录用户发布的帖子

GetMyComment方法实现获取当前登录用户的评论

GetMyGameExperience方法实现获取当前登录用户的比赛经历

1. Servlet类说明：

功能：联系前台页面与后台逻辑代码。每一个servlet类对应一个UserServiceImpl业务功能

## 公告信息管理：

### 模块编号和功能描述

* 模块编号：MN\_GGXXGL02。
* 功能描述：对于在公众号中发布的公告进行管理。

### 操作者

管理员可进行本功能模块所有操作。

### 与本模块相关的码表和表

表3‑3 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_ANNOUNCEMENT | 公告信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供表信息进行查看公告基本信息, 点击公告标题后能查看该公告的具体信息, 展示各方面信息. 可以在筛选框中填入信息进行对公告的筛选,提供基本的修改,删除功能键, 除此之外, 还提供专门的添加公告页面.

### 输入信息

|  |  |
| --- | --- |
| 表3-4公告信息输入输出信息表 | |
| **输入** | 查询、修改、删除时输入：公告编号（bigint） |
| 信息导入和修改时输入：公告标题（varchar（200））、公告内容（text）、公告类别编号（bigint）、是否注册（int）、注册开始时间（datetime）、注册结束时间（datetime）、开始时间（datetime）、持续时间(varchar(50)) 、是否发布(int)、浏览量(view) |
| **输出** | 查询,详情操作输出：公告标题（varchar（200））、公告内容（text）、公告类别编号（bigint）、发布人姓名(varchar(50))、发布时间(datetime)、是否注册（int）、注册开始时间（datetime）、注册结束时间（datetime）、开始时间（datetime）、持续时间(varchar(50)) 、是否发布(int)、浏览量(view) |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

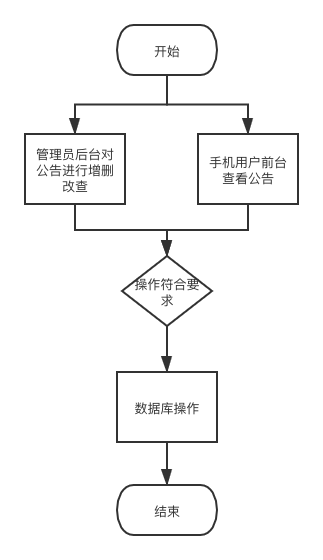


图3-3公告信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑4 公告信息管理类图

#### 类说明

（一）Announcement类说明：

1. 功能：实现公告类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) AnnouncementTag类说明：

1. 功能：实现公告类别类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

（三）AnnouncementServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addAnnouncement 方法实现用户信息的增加

deleteAnnouncement 方法实现用户信息的删除

updateAnnouncement 方法实现用户信息的修改

selectAnnouncement方法实现用户信息的查询

(四) AnnouncementController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于AnnouncementServiceImpl类说明

## 新闻信息管理：

### 模块编号和功能描述

1. 模块编号：MN\_XWXXGL03。
2. 功能描述：对在公众号发布的新闻进行管理。

### 操作者

管理员可进行本功能模块所有操作。

### 与本模块相关的码表和表

表3‑5 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_NEWS | 新闻信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供表信息进行查看新闻基本信息, 点击新闻标题后能查看该新闻的具体信息, 展示各方面信息. 可以在筛选框中填入信息进行对新闻的筛选,提供基本的修改,删除功能键, 除此之外, 还提供专门的添加新闻页面.

### 输入信息

|  |  |
| --- | --- |
| 表3-6新闻信息输入输出信息表 | |
| **输入** | 查询、修改、删除时输入：新闻编号（bigint） |
| 信息导入和修改时输入：新闻标题（varchar（200））、新闻内容（text）、新闻类别编号（bigint）、信息来源(varchar(50)) 、是否发布(int)、浏览量(view) |
| **输出** | 查询,详情操作输出：新闻标题（varchar（200））、新闻内容（text）、新闻类别编号（bigint）、发布人姓名(varchar(50))、发布时间(datetime)、信息来源 (varchar(50)) 、是否发布(int)、浏览量(view) |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

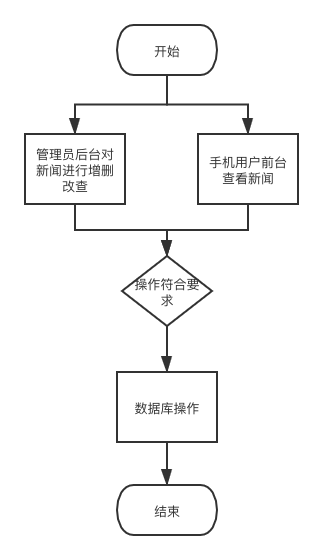


图3-5新闻信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑6 新闻信息管理类图

#### 类说明

(一) News类说明：

1. 功能：实现新闻类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) NewsTag类说明：

1. 功能：实现新闻类别类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(三) NewsServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addNews 方法实现新闻信息的增加

deleteNews 方法实现新闻信息的删除

updateNews 方法实现新闻信息的修改

selectNews方法实现新闻信息的查询

(四) NewsController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于NewsServiceImpl类说明

## 讨论区信息管理：

### 模块编号和功能描述

1. 模块编号：MN\_TLQXXGL04。
2. 功能描述：对在公众号讨论区发布的帖子的信息进行管理。

### 操作者

管理员可进行本功能模块所有操作, 手机端用户可进行增加自己的帖子,删除修改自己的帖子, 设置自己的帖子属性标签等操作。

### 与本模块相关的码表和表

表3‑7 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_POST | 帖子信息表 |  | √ | Input、output、update |

### 界面设计与说明

后台提供表信息展示帖子的一些基本属性, 点击帖子标题可查看帖子的详细情况, 在提供基本的删除, 修改功能键外, 还提供查看评论功能, 其中查看帖子的详情页面也要符合电脑端的展示, 做到尽可能的美观.

### 输入信息

|  |  |
| --- | --- |
| 表3-8帖子信息输入输出信息表 | |
| **输入** | 查询、修改、删除时输入：帖子编号（bigint） |
| 信息导入和修改时输入：帖子标题（varchar（50））、帖子内容（text）、帖子标签（bigint）、点赞数（int）、浏览量（int）、是否加精（int）、是否加热（int）、是否置顶（int） |
| **输出** | 查询操作输出：帖子标题（varchar（50））、帖子内容（text）、帖子标签（bigint）、点赞数（int）、浏览量（int）、是否加精（int）、是否加热（int）、是否置顶（int）、发布人（varchar(30)）、发布时间（datetime） |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

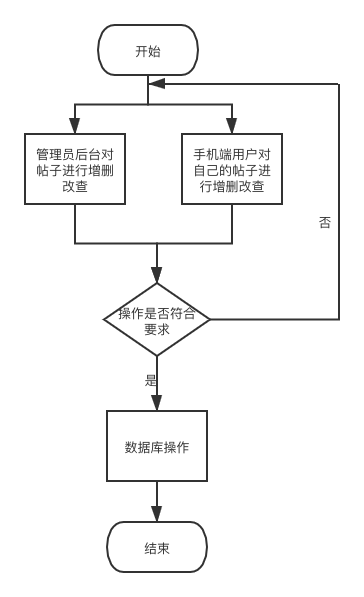


图3-7帖子信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑8 帖子信息管理类图

#### 类说明

(一) Post类说明：

1. 功能：实现帖子类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) Label类说明：

1. 功能：实现标签类(专为帖子的)的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(三) PostServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addPost 方法实现帖子信息的增加

deletePost 方法实现帖子信息的删除

updatePost 方法实现帖子信息的修改

selectPost方法实现帖子信息的查询

(四) PostController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于PostServiceImpl类说明

## 评论管理：

### 模块编号和功能描述

1. 模块编号：MN\_PLGL05。
2. 功能描述：对在公众号讨论区发布的帖子下面进行评论和回复的内容管理。

### 操作者

管理员可进行本功能模块所有操作, 手机端用户可自由的在任意一个帖子下面评论, 任意与评论进行回复, 删除自己的评论等。

### 与本模块相关的码表和表

表3‑9 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_COMMNET | 评论信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供表信息进行查看对应帖子下面的评论列表, 提供删除功能键。

### 输入信息

|  |  |
| --- | --- |
| 表3-10评论信息输入输出信息表 | |
| **输入** | 查询、删除时输入：评论编号（bigint） |
| 信息导入和修改时输入：评论内容（text）、回复帖子标号（bigint）、点赞数（int） |
| **输出** | 查询操作输出：评论内容（text）、回复帖子标号（bigint）、点赞数（int）、评论人（varchar（30））、评论时间（datetime） |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

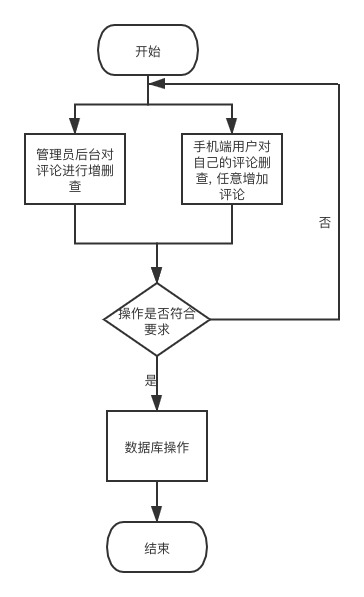


图3-9评论信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑10 评论信息管理类图

#### 类说明

(一) Comment类说明：

1. 功能：实现评论类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) Post类说明：

1. 功能：实现新闻类别类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(三) CommentServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addComment 方法实现评论信息的增加

deleteComment 方法实现评论信息的删除

updateComment 方法实现评论信息的修改

selectComment方法实现评论信息的查询

(四) CommentController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于CommentServiceImpl类说明

## 值班管理：

### 模块编号和功能描述

1. 模块编号：MN\_ZBGL06。
2. 功能描述：对在公众号上发布的实验室值班信息进行管理。

### 操作者

管理员可进行本功能模块所有操作。

### 与本模块相关的码表和表

表3‑11模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_ONDUTY | 值班信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供表信息进行查看值班的排班信息, 有添加值班信息的对话框, 填入对应的信息即可安排值班, 值班人员从用户表中选择, 可修改临时电话, 可以在筛选框中填入信息进行对值班信息进行筛选查看,提供基本的修改,删除功能键。

### 输入信息

|  |  |
| --- | --- |
| 表3-12值班信息输入输出信息表 | |
| **输入** | 查询、修改、删除时输入：用户编号（bigint） |
| 信息导入和修改时输入：值班人员用户编号(biging) 、值班人用户名（varchar（50））、值班人员电话（varchar（50））、值班周期开始时间（datetime）、值班周期结束时间（datetime） |
| **输出** | 查询操作输出：值班人员用户编号(biging) 、值班人用户名（varchar（50））、值班人员电话（varchar（50））、值班周期开始时间（datetime）、值班周期结束时间（datetime） |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

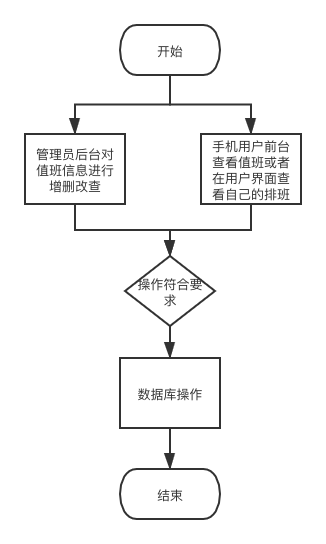


图3-11值班信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑12 值班信息管理类图

#### 类说明

(一) OnDuty类说明：

1. 功能：实现值班类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) OnDutyServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addOnDuty 方法实现值日信息的增加

deleteOnDuty 方法实现值日信息的删除

updateOnDuty 方法实现值日信息的修改

selectOnDuty方法实现值日信息的查询

(三) OnDutyController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于OnDutyServiceImpl类说明

## 反馈管理：

### 模块编号和功能描述

1. 模块编号：MN\_FKGL07。
2. 功能描述：对在公众号中用户个人界面中发起对本公众号的一些bug反馈和建议信息进行处理。

### 操作者

管理员可进行本功能模块所有操作。手机端用户可进行添加反馈内容。

### 与本模块相关的码表和表

表3‑13 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_FEEDBACK | 反馈信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供友好的反馈界面进行展示, 方便后台管理员查看, 这个操作只是为了吸收用户的反馈让公众号能更好的运行。

### 输入信息

|  |  |
| --- | --- |
| 表3-14反馈信息输入输出信息表 | |
| **输入** | 无 |
| 无 |
| **输出** | 反馈人编号（bigint）、反馈内容（text）、反馈时间（datetime） |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

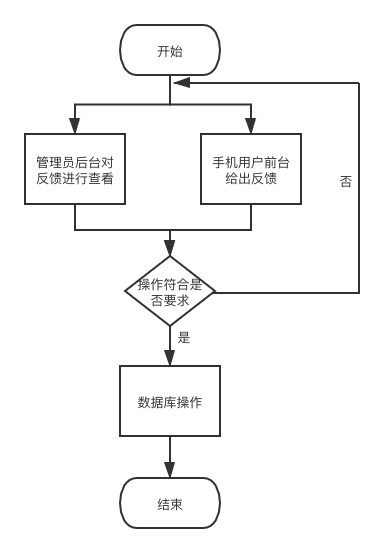


图3-13反馈信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑14 反馈信息管理类图

#### 类说明

(一) Feedback类说明：

1. 功能：实现反馈类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) FeedbackServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addFeedback 方法实现反馈信息的增加

selectFeedback方法实现反馈信息的查询

(三) FeedbackController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于FeedbackServiceImpl类说明

## 友链管理：

### 模块编号和功能描述

1. 模块编号：MN\_YLGL08。
2. 功能描述：对在公众号中添加的友情链接进行管理。

### 操作者

管理员可进行本功能模块所有操作。

### 与本模块相关的码表和表

表3‑15 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_FRIENDURL | 友链信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供表信息进行查看友链的基本信息, 点击友链名称后能直接跳转到相应的网页处, 可以在筛选框中填入信息进行对友链的筛选,提供对话框式添加友链, 提供基本的修改,删除功能键。

### 输入信息

|  |  |
| --- | --- |
| 表3-16友链信息输入输出信息表 | |
| **输入** | 查询、修改、删除时输入：友链编号（bigint） |
| 信息导入和修改时输入：友链名称（varchar（50））、友链地址（varchar（50））、友链类别（varchar（50）） |
| **输出** | 查询操作输出：友链名称（varchar（50））、友链地址（varchar（50））、友链类别（varchar（50）)、发布人姓名(varchar(50))、发布时间(datetime) |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

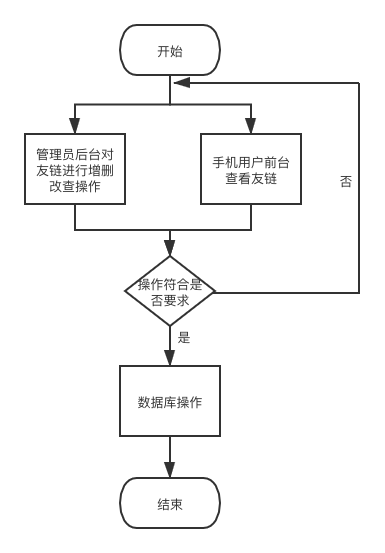


图3-15友链信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑16 友链信息管理类图

#### 类说明

(一) FriendUrl类说明：

1. 功能：实现友链类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) FriendUrlServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addFriendUrl 方法实现友链信息的增加

deleteFriendUrl方法实现友链信息的删除

updateFriendUrl 方法实现友链信息的修改

selectFriendUrl方法实现友链信息的查询

(三) FriendUrlController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于FriendUrlServiceImpl类说明

## 举报管理：

### 模块编号和功能描述

1. 模块编号：MN\_JBGL09。
2. 功能描述：对在公众号中帖子、评论和用户进行举报的管理。

### 操作者

管理员可进行本功能模块所有操作。手机端用户可举报公众号中的帖子、评论、用户进行举报后添加的数据.

### 与本模块相关的码表和表

表3‑17 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_REPORT | 举报信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供表信息进行查看举报的相关信息, 提供处理和删除功能键。

### 输入信息

|  |  |
| --- | --- |
| 表3-18举报信息输入输出信息表 | |
| **输入** | 无 |
| 无 |
| **输出** | 举报类型(varchar), 举报说明（varchar(50))、举报人姓名(varchar(50))、举报时间(datetime) 、举报理由(varchar(50))、处理人姓名(varchar(50))、处理时间(datetime)、处理结果(varchar(50))、是否处理(int) |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

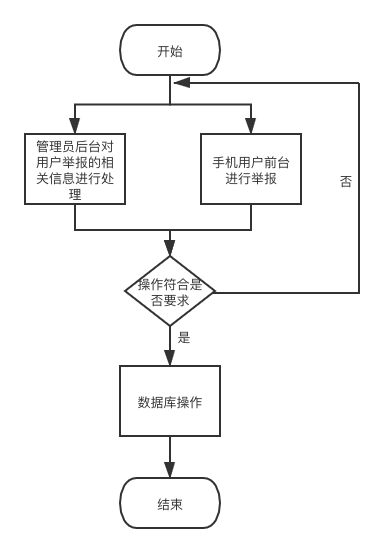


图3-17举报信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑18 举报信息管理类图

#### 类说明

(一) Report类说明：

1. 功能：实现友链类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) ReportServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addReport 方法实现举报信息的增加

deleteReport方法实现举报信息的删除

updateReport 方法实现举报信息的修改

selectReport方法实现举报信息的查询

(三) ReportController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于ReportServiceImpl类说明

## 报名管理：

### 模块编号和功能描述

1. 模块编号：MN\_BMGL010。
2. 功能描述：对在公众号中发布的公告中需要进行报名的公告对报名用户的信息处理。

### 操作者

管理员可进行本功能模块所有操作。手机端用户报名。

### 与本模块相关的码表和表

表3‑19 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| TB\_REGISTER | 报名信息表 |  | √ | Input、output、update |

### 界面设计与说明

提供表信息查看某一个公告报名人情况, 可以对报名人进行查询, 修改, ,删除以及可以导出报名人员表。

### 输入信息

|  |  |
| --- | --- |
| 表3-20报名信息输入输出信息表 | |
| **输入** | 查询、修改、删除时输入：用户学号（bigint） |
| 信息导入和修改时输入：对应的公告编号(bigint)、报名时间(datetime)、报名用户编号(bigint) |
| **输出** | 查询操作输出：对应的公告标题(varchar(50))、报名时间(datetime)、学号(bigint)、真实姓名(varchar(50) |
| 对应的界面展示 |
| **异常处理** | 友好的弹窗提示 |

### 算法

查询操作时涉及数据库相应的算法。

### 处理流程

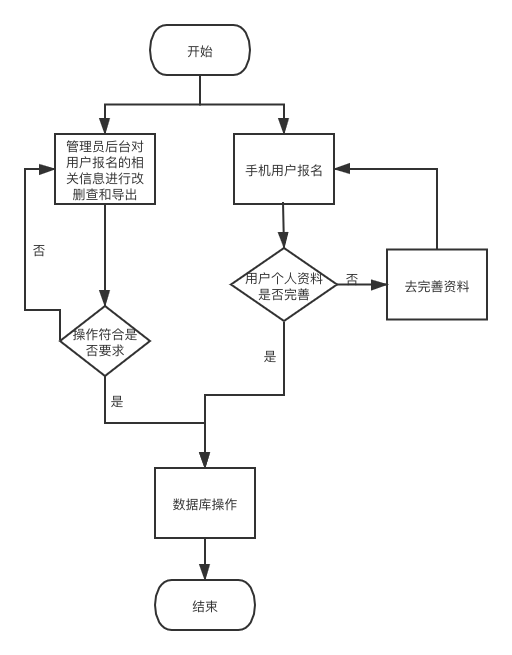


图3-19报名信息管理流程图

### 类设计

#### 类图

|  |
| --- |
|  |

图3‑20 报名信息管理类图

#### 类说明

(一) Register类说明：

1. 功能：实现友链类的属性定义和属性值的获取、设置。
2. 主要方法：主要方法为各属性的getter/setter方法。

(二) RegisterServiceImpl类说明：

1. 功能：作为Controller层和Mapper层的中间层, 用于在传递给Mapper时做一些数据上的处理和总结。

2. 主要方法：

addRegister 方法实现举报信息的增加

deleteRegister方法实现举报信息的删除

updateRegister 方法实现举报信息的修改

selectRegister方法实现举报信息的查询

(三) RegisterController类说明：

1. 功能：作为控制层负责接收前端发来的请求, 并获取参数
2. 主要方法：对应于RegisterServiceImpl类说明

# 内部接口设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表4-1 构件接口列表 | | | | |
| **模块名称** | **接口编号** | **接口名称** | **接口类型** | **说明** |
| 用户信息管理 | IN01 | UserMapper | 内部外 | 用户信息的增删查改功能 |
| 公告信息管理 | IN02 | AnnouncementMapper | 内部 | 公告信息的增删查改功能 |
| 新闻信息管理 | IN03 | NewsMapper | 内部 | 新闻信息的增删查改功能 |
| 讨论区信息管理 | IN04 | PostMapper | 内部外 | 讨论区的增删查改功能 |
| 评论管理 | IN05 | CommentMapper | 内部外 | 评论的增删查改功能 |
| 值班管理 | IN06 | OnDutyMapper | 内部 | 值班的增删查改功能 |
| 反馈管理 | IN07 | FeedbackMapper | 内部 | 反馈的增查功能 |
| 友链管理 | IN08 | FriendUrlMapper | 内部 | 友链的增删查改功能 |
| 举报管理 | IN09 | ReportMapper | 内部外 | 举报的增删查功能 |
| 报名管理 | IN10 | RegisterMapper | 内部外 | 报名的增删查改功能 |

## UserMapper接口

1. 接口属性设计

表4‑2 UserMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN01 |
| **接口名称** | UserMapper |
| **接口说明** | 用户信息的增删查改功能 |
| **数据来源** | 用户注册公众号时提供 |
| **调用者** | 后台管理员或者手机端用户 |
| **输入** | User类 |
| **输出** | User类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

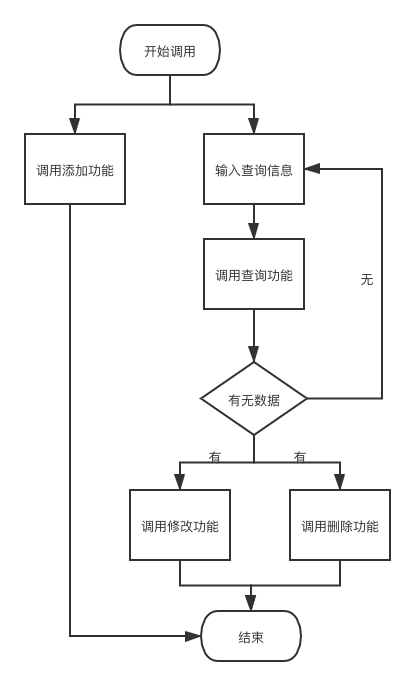


图4-1 UserMapper接口流程图

## AnnouncementMapper接口

1. 接口属性设计

表4‑2 AnnouncementMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN02 |
| **接口名称** | AnnouncementMapper |
| **接口说明** | 公告信息的增删查改功能 |
| **数据来源** | 后台管理员添加 |
| **调用者** | 后台管理员 |
| **输入** | Announcement类 |
| **输出** | Announcement类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

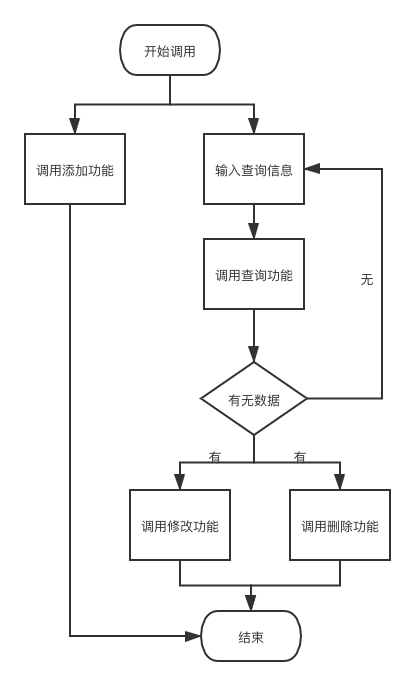


图4-2 AnnouncementMapper接口流程图

## NewsMapper接口

1. 接口属性设计

表4‑3 NewsMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN03 |
| **接口名称** | NewsMapper |
| **接口说明** | 新闻信息的增删查改功能 |
| **数据来源** | 后台管理员添加 |
| **调用者** | 后台管理员 |
| **输入** | News类 |
| **输出** | News类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

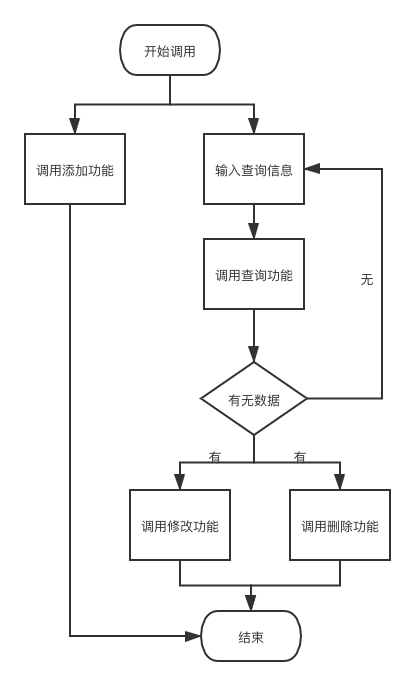


图4-3 NewsMapper接口流程图

## PostMapper接口

1. 接口属性设计

表4‑4 PostMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN04 |
| **接口名称** | PostMapper |
| **接口说明** | 讨论帖的增删查改功能 |
| **数据来源** | 公众号注册用户发表 |
| **调用者** | 公众号的注册用户 |
| **输入** | Post类 |
| **输出** | Post类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

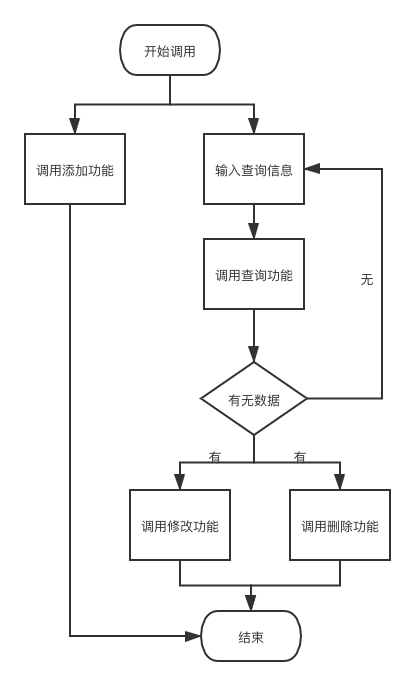


图4-4 PostMapper接口流程图

## CommentMapper接口

1. 接口属性设计

表4‑4 CommentMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN05 |
| **接口名称** | CommentMapper |
| **接口说明** | 评论的增删查改功能 |
| **数据来源** | 公众号注册用户发表 |
| **调用者** | 公众号的注册用户 |
| **输入** | Comment类 |
| **输出** | Comment类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

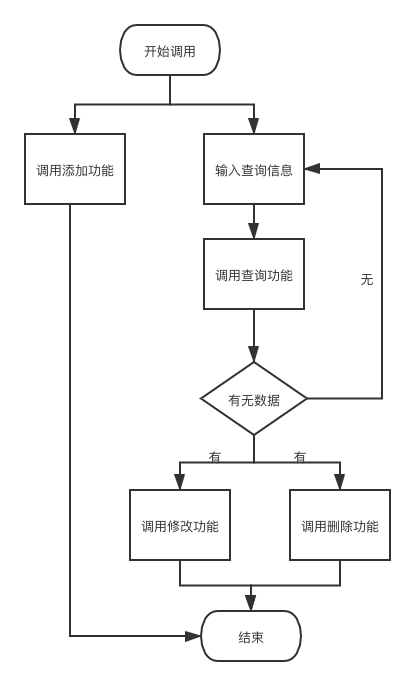


图4-5 CommentMapper接口流程图

## OnDutyMapper接口

1. 接口属性设计

表4‑6 OnDutyMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN06 |
| **接口名称** | OnDutyMapper |
| **接口说明** | 值班信息的增删查改功能 |
| **数据来源** | 后台管理员添加 |
| **调用者** | 后台管理员或者手机端用户 |
| **输入** | OnDuty类 |
| **输出** | OnDuty类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

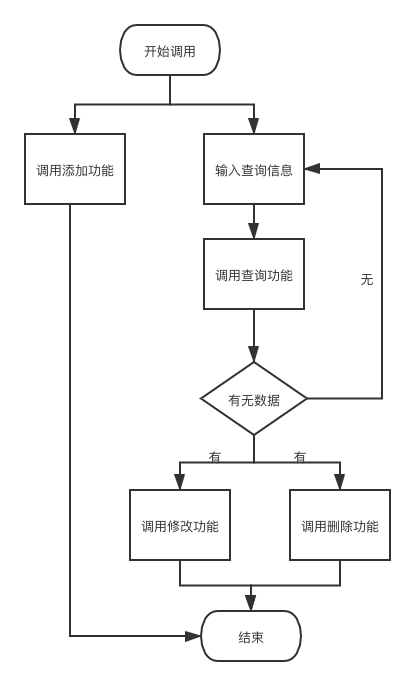


图4-6 OnDutyMapper接口流程图

## FeedbackMapper接口

1. 接口属性设计

表4‑7 FeedbackMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN07 |
| **接口名称** | FeedbackMapper |
| **接口说明** | 反馈信息的查和添加功能 |
| **数据来源** | 公众号手机端用户 |
| **调用者** | 管理员和公众号手机端用户 |
| **输入** | Feedback类 |
| **输出** | Feedback类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

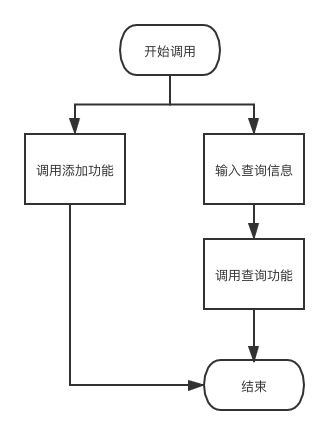


图4-7 FeedbackMapper接口流程图

## FriendUrlMapper接口

1. 接口属性设计

表4‑8 FriendUrlMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN08 |
| **接口名称** | FriendUrlMapper |
| **接口说明** | 友链信息的增删查改功能 |
| **数据来源** | 后台管理员添加 |
| **调用者** | 后台管理员 |
| **输入** | FriendUrl类 |
| **输出** | FriendUrl类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

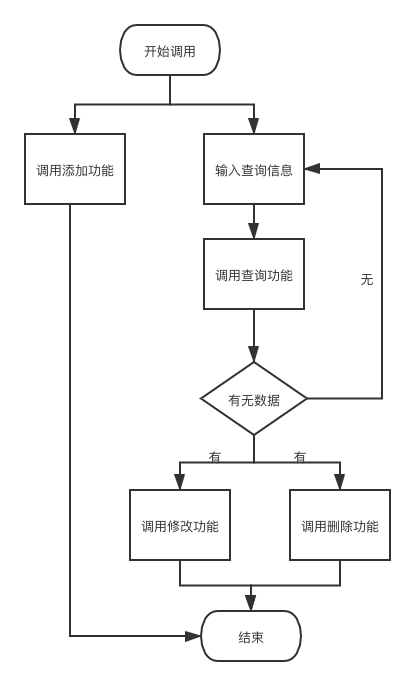


图4-8 FriendUrlMapper接口流程图

## ReportMapper接口

1. 接口属性设计

表4‑9 ReportMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN09 |
| **接口名称** | ReportMapper |
| **接口说明** | 举报信息的增删查功能 |
| **数据来源** | 公众号手机端用户 |
| **调用者** | 后台管理员或者手机端用户 |
| **输入** | Report类 |
| **输出** | Report类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

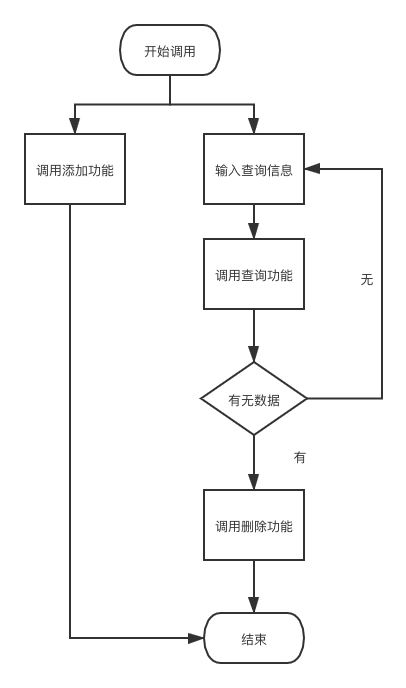


图4-9 ReportMapper接口流程图

## RegisterMapper接口

1. 接口属性设计

表4‑10 RegisterMapper接口说明

|  |  |
| --- | --- |
| **接口编号** | IN010 |
| **接口名称** | UserMapper |
| **接口说明** | 报名信息的增删查改功能 |
| **数据来源** | 公众号用户报名 |
| **调用者** | 后台管理员或者手机端用户 |
| **输入** | Register类 |
| **输出** | Register类 |
| **处理流程** | 按照流程图之上而下处理 |

1. 接口处理流程图

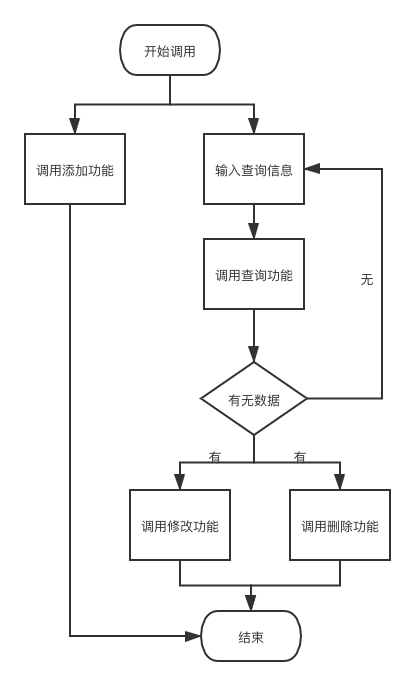


图4-10 RegisterMapper接口流程图

# 系统出错处理设计

## 出错信息

|  |  |  |
| --- | --- | --- |
| 序号 | 出错或故障情况 | 系统输出信息及处理 |
| 1 | 如果用户名和密码为空 | 客户端调用进行处理，系统弹出错误的提示，要求用户重新输入。 |
| 2 | 在需要输入信息的输入框时输入为空 | 客户端调用进行处理，系统弹出错误的提示，要求用户重新输入。 |
| 3 | 在插入的过程中，SQL语句有可能抛出异常  例如主键重复，不能为空的语句没有输入数值 | 系统会抛出异常信息，并给出提示信息，只要用户返回上一个页面，重新输入正确的数据，进行同样的操作即可得到正确的结果。 |
| 4 | 用户修改个人密码，输入的原密码与以前的不相同，或新密码与确认密码不相同时 | 调用语句提示用户重新输入 |

## 补救措施

系统通过不同的界面提示信息，提示用户更正操作，让后台屏蔽异常或抛出异常状态，使得管理员在调试的过程中发现问题。