“一族一谱”管理系统

概要设计说明书

小组：软件工程第六组

组员：王聪 曾德龙 蔡楷欣 曾欣

程钰涵 陈冠旭 郑湘萍

版次： v1.9

日期：2021年5月22日

文档修改历史记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本号 | 修改人 | 日期 | 修改内容 | 备注 |
| v1.0 | 郑湘萍 | 5.17 | 文档的1、2、4、6、7的编写 |  |
| v1.1 | 曾欣 | 5.17 | 文档3、5、8、9的编写 |  |
| v1.2 | 曾欣 | 5.23 | 修改层次方框图 |  |
| v1.3 | 郑湘萍 | 5.23 | 添加3.3内容，整合内容与格式 |  |
| v1.4 | 郑湘萍 | 6.12 | 补充表格、程序流图，文字说明 |  |
| v1.5 | 曾欣 | 6.13 | 初步审核以及修改 |  |
| v1.6 | 曾欣 | 6.28 | 补充程序处理流程和接口设计 |  |
| v1.7 | 郑湘萍 | 6.30 | 补充出错处理表，调整页面格式 |  |
| V1.8 | 曾欣 | 7.1 | 优化层次结构图，补充接口使用 |  |
| V1.9 | 陈冠旭 | 7.3 | 更新了内部接口、出错处理对策、安全保密设计、维护设计 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1 引言 4](#_Toc76223839)

[1.1 编写目的 4](#_Toc76223840)

[1.2 背景 4](#_Toc76223841)

[1.3 定义 4](#_Toc76223842)

[1.4 参考资料 5](#_Toc76223843)

[2 任务概述 5](#_Toc76223844)

[2.1 目标 5](#_Toc76223845)

[2.2 运行环境 6](#_Toc76223846)

[2.3 需求概述 6](#_Toc76223847)

[2.4 条件与限制 6](#_Toc76223848)

[3 总体设计 6](#_Toc76223849)

[3.1 处理流程 6](#_Toc76223850)

[3.1.1 微信小程序端上的程序流程 6](#_Toc76223851)

[3.1.2 网页端上的程序流程 12](#_Toc76223852)

[3.2 总体结构和模块外部设计 15](#_Toc76223853)

[3.3 功能分配 15](#_Toc76223854)

[4 接口设计 16](#_Toc76223855)

[4.1 外部接口 16](#_Toc76223856)

[4.1.1 用户界面 16](#_Toc76223857)

[4.1.2 软件接口 18](#_Toc76223858)

[4.2 内部接口 19](#_Toc76223861)

[5 系统数据结构设计 19](#_Toc76223862)

[5.1 逻辑结构设计 19](#_Toc76223863)

[5.2 物理结构设计 20](#_Toc76223864)

[5.3 数据结构与程序的关系 21](#_Toc76223865)

[6 运行设计 21](#_Toc76223866)

[6.1 运行模块的组合 21](#_Toc76223867)

[6.2 运行控制 22](#_Toc76223868)

[6.3 运行时间 22](#_Toc76223869)

[7 出错处理设计 22](#_Toc76223870)

[7.1 出错输出信息 22](#_Toc76223871)

[7.2 出错处理对策 23](#_Toc76223872)

[8 安全保密设计 24](#_Toc76223873)

[8.1 数据传输层面 24](#_Toc76223874)

[8.2 应用层级 24](#_Toc76223875)

[8.3 发布阶段层级 25](#_Toc76223876)

[9 维护设计 25](#_Toc76223877)

1 引言

* 1. 编写目的

本阶段已在系统的需求分析的基础上，对”一族一谱”管理系统做概要设计。主要解决了实现该系统需求的程序模块设计问题。包括如何把系统划分成若干个模块、决定各个模块之间的接口、模块之间传递的信息，以及数据结构、模块结构的设计等。在以下的概要设计报告中将对在本阶段中对系统所做的所有概要设计进行详细的说明。

* 1. 背景

古释：“族者，属也，与其子孙共相联属，其旁支别属。”中国自古以来就很注重宗族文化传承，而族谱就是其宗族文化传承的重要载体，属于珍贵的人文资料，有其不可替代的独特功能。而目前，现有族谱管理方式主要采用有纸化存储与记录，这种方式效率低下，且耗费一定的人力，故提出开发一个线上管理修改族谱的管理系统，即“一族一谱”管理系统

a. 开发的软件系统的名称：“一族一谱”管理系统  
　　b. 本项目的任务提出者：广东省云浮市云安区富林镇民主村委结二村村委会

本项目的开发者：软件工程第六小组

用户: 广东省云浮市云安区富林镇民主村委结二村  
　　c. 该软件系统同其他系统或其他机构的基本的相互来往关系：本项目分为两个端：微信小程序和网页。其中，微信小程序主要使用Windows 10系统上的微信开发者工具进行开发；网页端主要基于Java，Spring Boot，Maven，MongoDB技术栈进行开发。两个段共同使用一个Bmob后端云，进行数据的交互。

1.3 定义

JavaScript：JavaScript（简称“JS”） 是一种具有函数优先的轻量级，解释型或即时编译型的[编程语言](https://baike.baidu.com/item/%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/9845131" \t "_blank)。虽然它是作为开发[Web](https://baike.baidu.com/item/Web/150564" \t "_blank)页面的[脚本语言](https://baike.baidu.com/item/%E8%84%9A%E6%9C%AC%E8%AF%AD%E8%A8%80/1379708" \t "_blank)而出名，但是它也被用到了很多非[浏览器](https://baike.baidu.com/item/%E6%B5%8F%E8%A7%88%E5%99%A8/213911" \t "_blank)环境中，JavaScript 基于原型编程、多范式的动态脚本语言，并且支持面向对象、命令式和声明式（如[函数](https://baike.baidu.com/item/%E5%87%BD%E6%95%B0/301912" \t "_blank)式编程）风格。

Spring Boot：Spring Boot是由Pivotal团队提供的全新[框架](https://baike.baidu.com/item/%E6%A1%86%E6%9E%B6/1212667" \t "_blank)，其设计目的是用来[简化](https://baike.baidu.com/item/%E7%AE%80%E5%8C%96/3374416" \t "_blank)新[Spring](https://baike.baidu.com/item/Spring/85061" \t "_blank)应用的初始搭建以及开发过程。该框架使用了特定的方式来进行配置，从而使开发人员不再需要定义样板化的配置。通过这种方式，Spring Boot致力于在蓬勃发展的快速应用开发领域(rapid application development)成为领导者。

Maven：Maven项目对象模型(POM)，可以通过一小段描述信息来管理项目的构建，报告和[文档](https://baike.baidu.com/item/%E6%96%87%E6%A1%A3/1009768" \t "_blank)的[项目管理工具](https://baike.baidu.com/item/%E9%A1%B9%E7%9B%AE%E7%AE%A1%E7%90%86%E5%B7%A5%E5%85%B7/6854630" \t "_blank)软件。

MongoDB ：MongoDB 是一个基于分布式文件存储的数据库。

1.4 参考资料

《软件工程导论》，张海藩，清华大学出版社

可行性分析报告，软件工程第六小组，2021/04/11

需求说明书，软件工程第六小组，2021/05/10

软件工程概要设计说明书（ISO 标准）：

https://wenku.baidu.com/view/7b43698626d3240c844769eae009581b6ad9bd4b.html

2 任务概述

2.1 目标

本系统主要应用于各宗族之间的电子族谱信息管理，主要目的是方便族谱管理人员进行族谱修建、族谱编辑，提高族谱修建的效率，其主要应用对象是广东省云浮市云安区富林镇民主村委结二村村民，后期可逐渐扩充范围到各个宗族对应的宗族群体。

2.2 运行环境

本系统分为网页端与微信小程序端，网页端运行平台为Windows XP 或以上的系统，所需浏览器版本为Internet Explorer6.0 或以上版本（建议采用 Internet Explorer8.0），微信小程序端运行平台为安卓6.0.0以上的手机。所有的应用数据库运行在Bmob后端云、MongoDB数据库上。

2.3 需求概述

一族一谱管理系统旨在实现族谱管理的现代化和提高族谱修建的效率。其主要内容应该包括网页端的族谱信息表导入，族谱信息编辑（添加、修改、删除），族谱册的导出（包括样式和格式的选择），发布通知模块，微信小程序端的匹配族谱，申请修改，查看族谱模块。（详见《一族一谱需求说明书》）

2.4 条件与限制

1. 一个用户在系统中只能有一个账号。

2. 用户想要修改族谱信息需进行申请

3 总体设计

3.1 处理流程

3.1.1 微信小程序端上的程序流程

登陆模块程序流程图如图3-1所示：

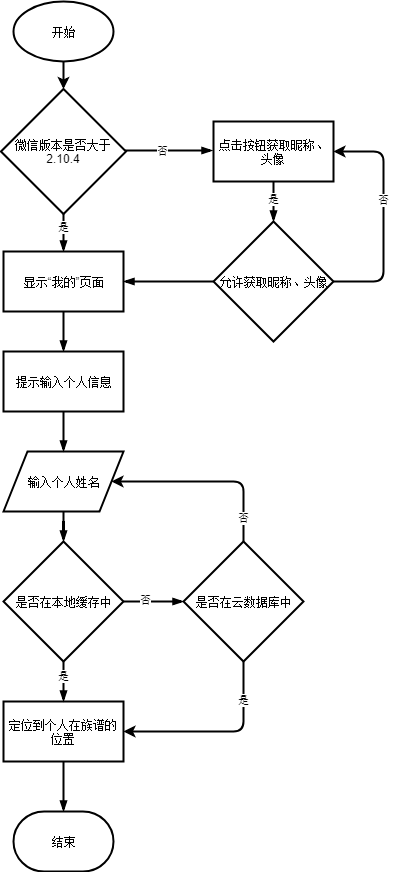


图3-1 登陆模块程序流程图

申请修改模块程序流程图如图3-2所示：

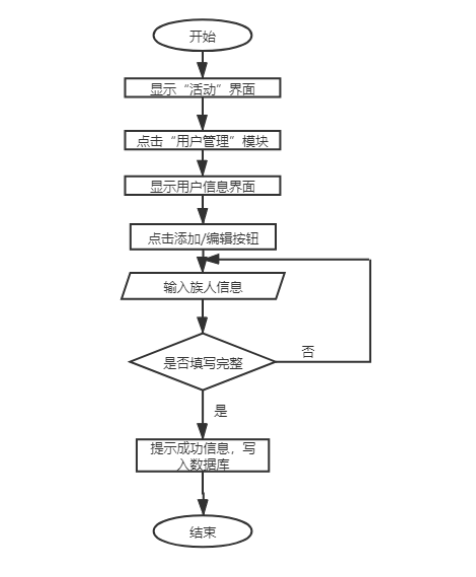


图3-2申请修改模块程序流程图

查看族谱模块程序流程图如图3-3所示：

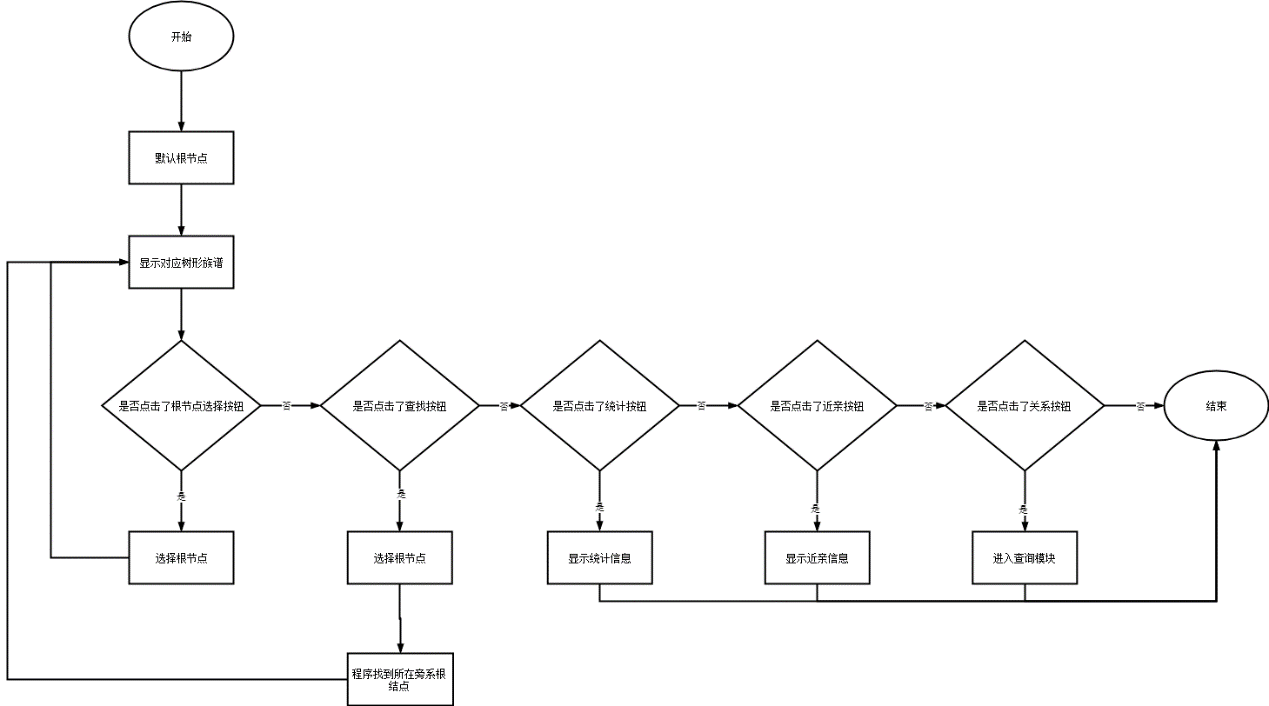


图3-3查看族谱模块程序流程图

打印成册模块程序流程图如图3-4所示：

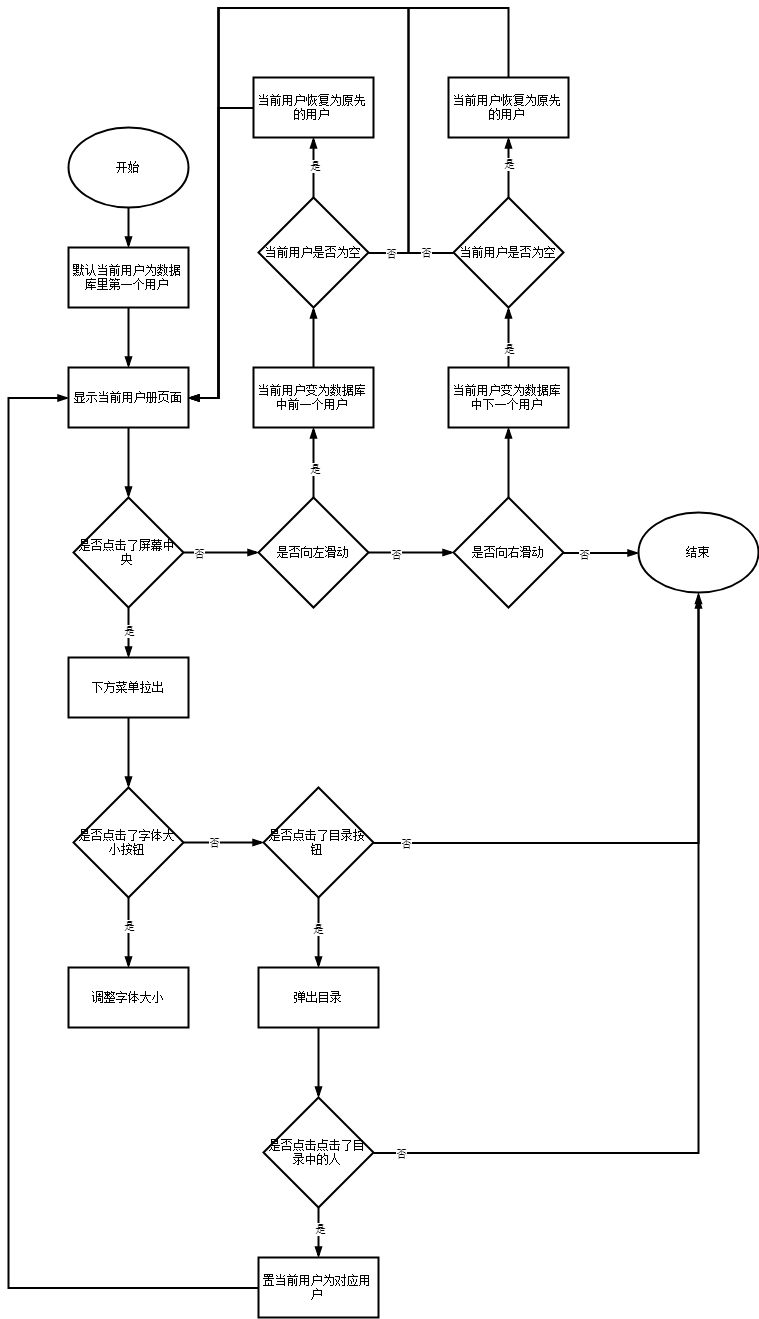


图3-4打印成册模块程序流程图

历史信息查询模块程序流程图如图3-5所示：

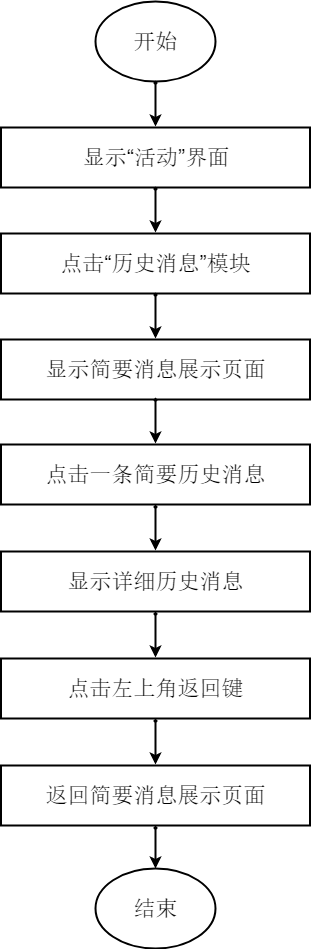


图3-5历史信息查询模块程序流程图

除了在用户需求上要求的基本功能之外，还添加了统计功能和关系查询功能。统计模块的程序流程图如图3-6所示，关系查询程序流程图如图3-7所示：

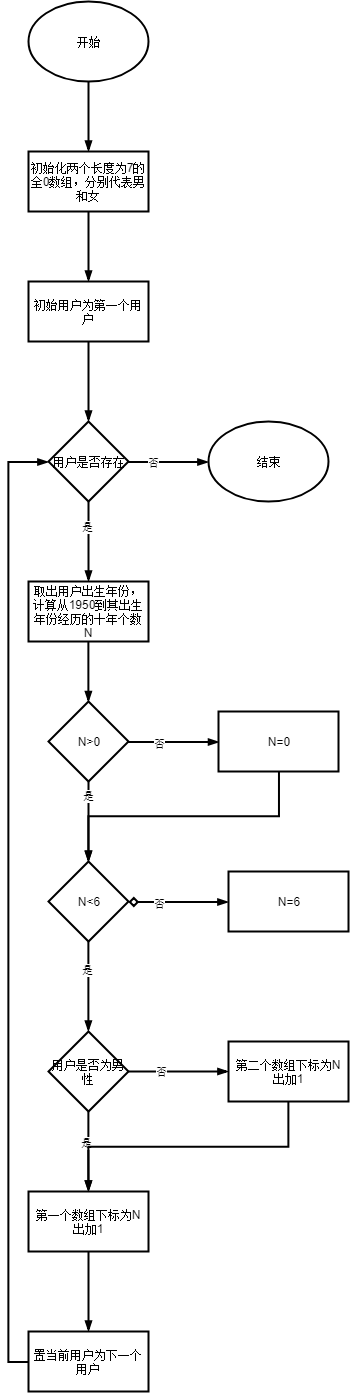
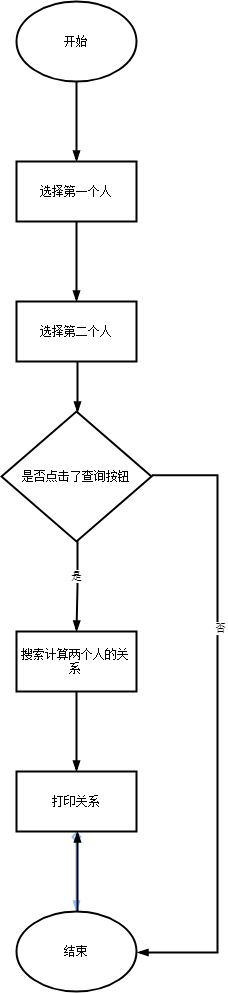
 

图3-6统计模块程序流程图 图3-7 关系查询模块程序流程图

3.1.2 网页端上的程序流程

用户信息添加修改模块程序流程图如图3-8所示：

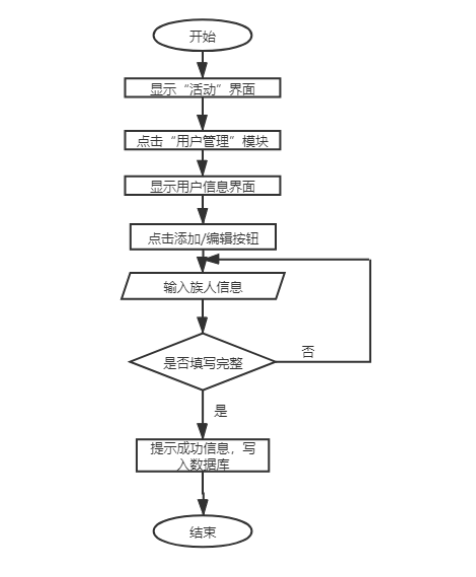


图3-8 用户信息添加修改模块程序流程图

发布以及查看通知模块程序流程图如图3-9、3-10所示：

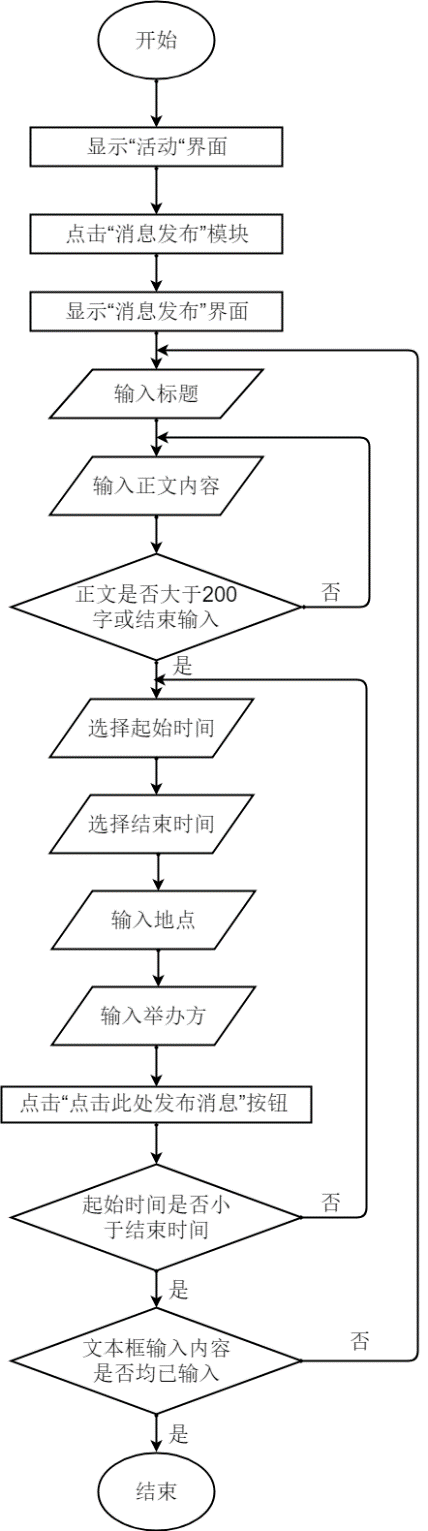
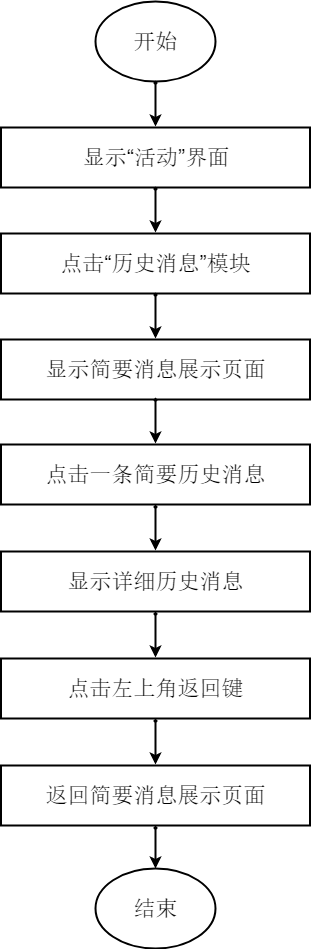
 

图3-9 发布通知模块程序流程图 图3-10 查看消息通知模块程序流程图

审批模块程序流程图如图3-11所示：

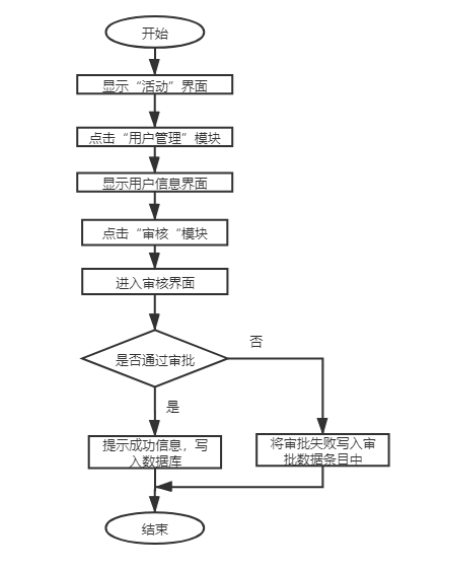


图3-11 审批模块程序流程图

3.2 总体结构和模块外部设计

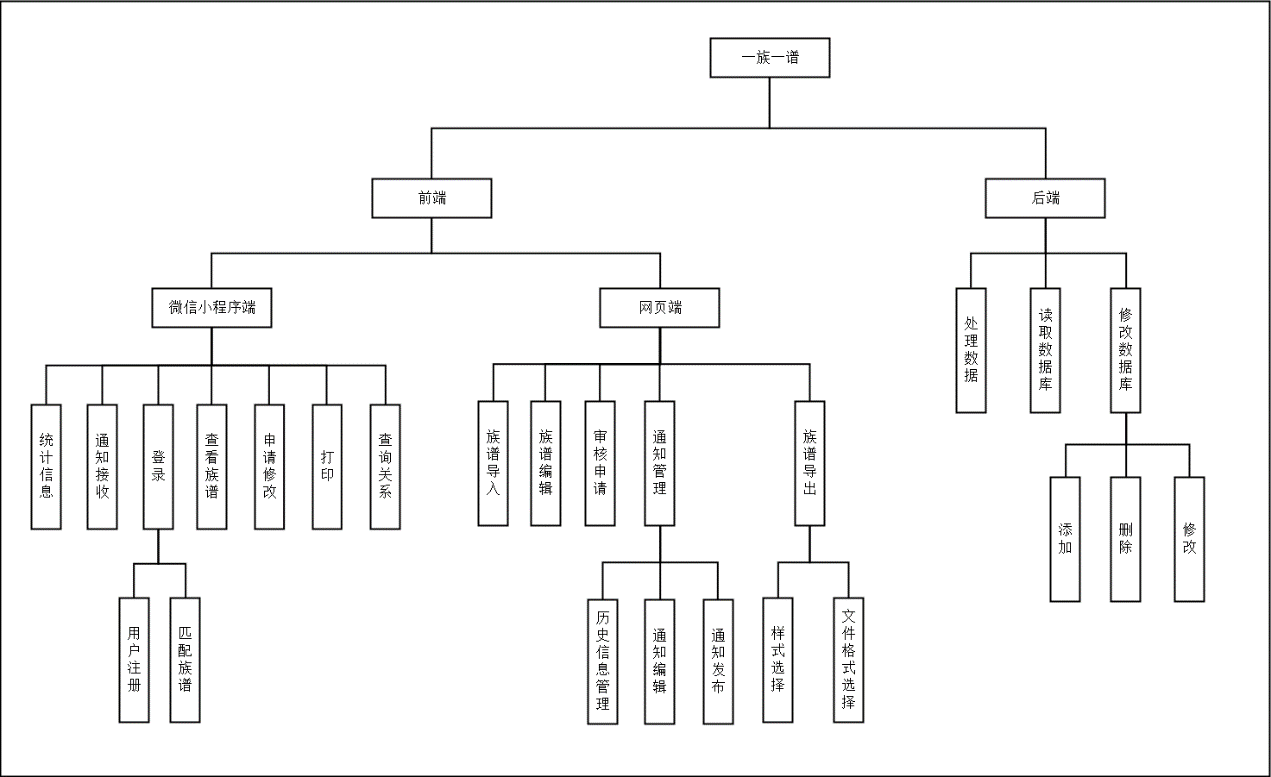


图3-12系统层次方框图

3.3 功能分配

在用户界面部分各项功能需求的实现同各模块的分配关系如表3-1所示。

表3-1 功能需求与系统模块的关系表

|  |  |  |
| --- | --- | --- |
|  | 微信小程序端模块 | 网页端管理员模块 |
| 查询功能 | √ | √ |
| 数据录入功能 |  | √ |
| 修改族谱功能 |  | √ |
| 删除信息功能 |  | √ |
| 打印功能 | √ | √ |

4 接口设计

4.1 外部接口

4.1.1 用户界面

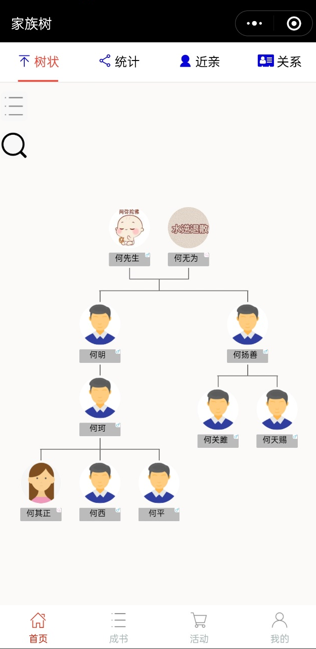
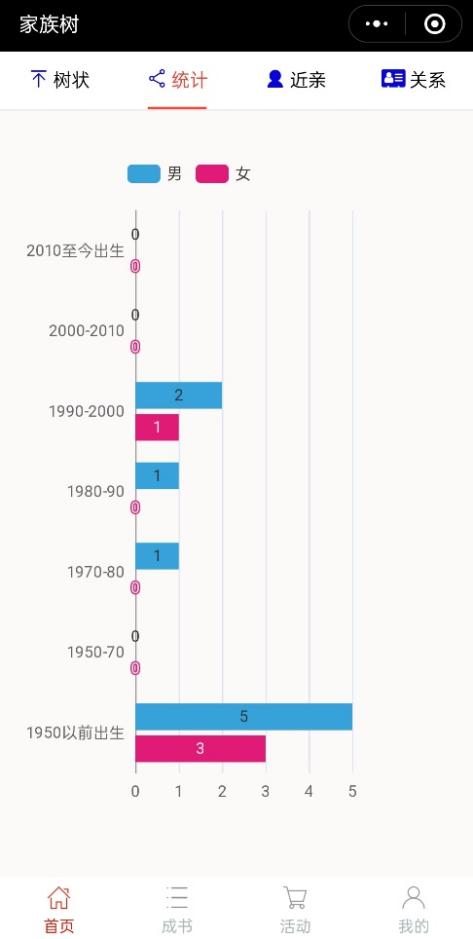
在用户界面部分，根据需求分析的结果，用户需要一个用户友善界面。在界面设置部分，应做到简单明了，易于操作，并且要注意到界面的布局，应突出显示重要信息，外观也要做到合理化。总的来说，系统的用户界面应做到可靠性、简单性、易学习和使用。

在微信小程序端，运行主界面首先弹出个人信息补充表，如图4-1:



图4-1登录界面

在注册或登陆后然后显示出族谱树型图、我的主页、关系查询页面等。如图4-2：

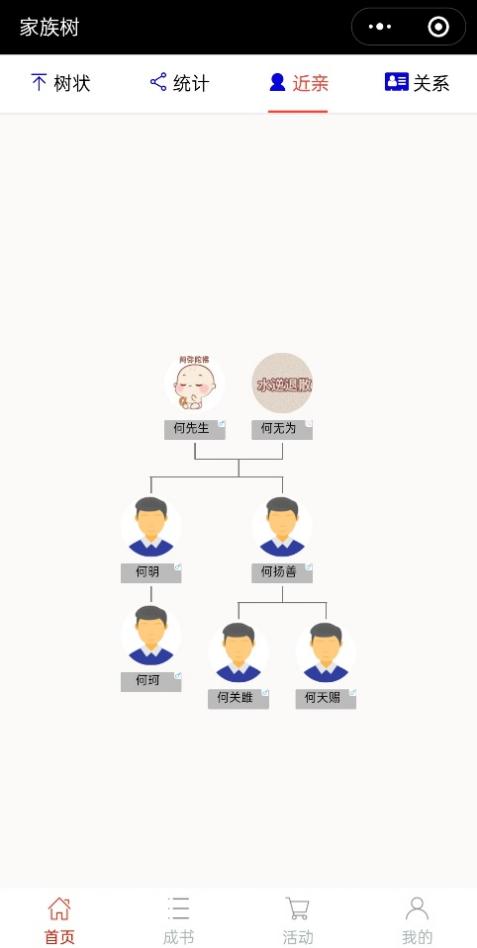
 

图4-2首页功能界面

还有成书和消息发布功能如图4-3,4-4所示：

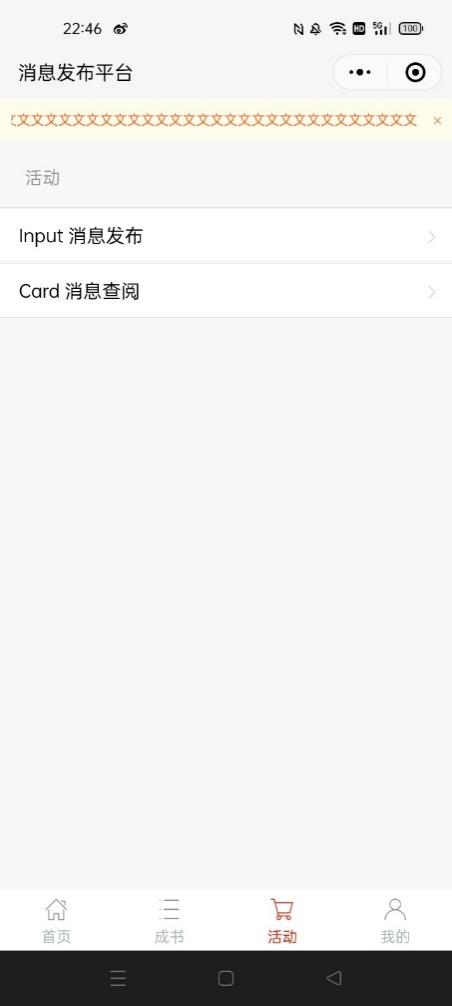
**** 

图4-3 成书功能界面 图4-4 消息发布界面

在网页端，点击链接进入后，首先弹出账号注册或登陆界面，如图4-5；登陆之后主界面显示族谱目录界面，如图4-6；总体族谱人员显示信息界面如图4-7.

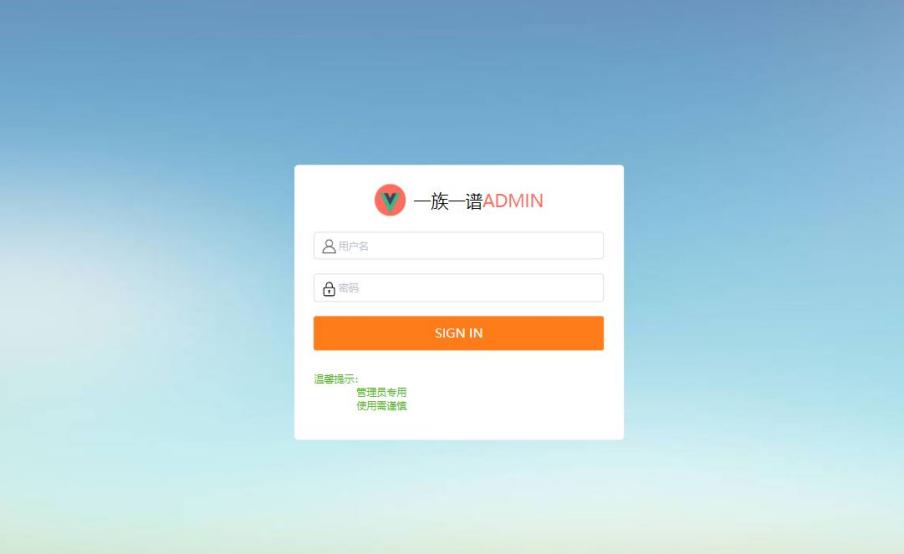


图4-5 账号注册或登录界面



图4-6 族谱目录显示界面

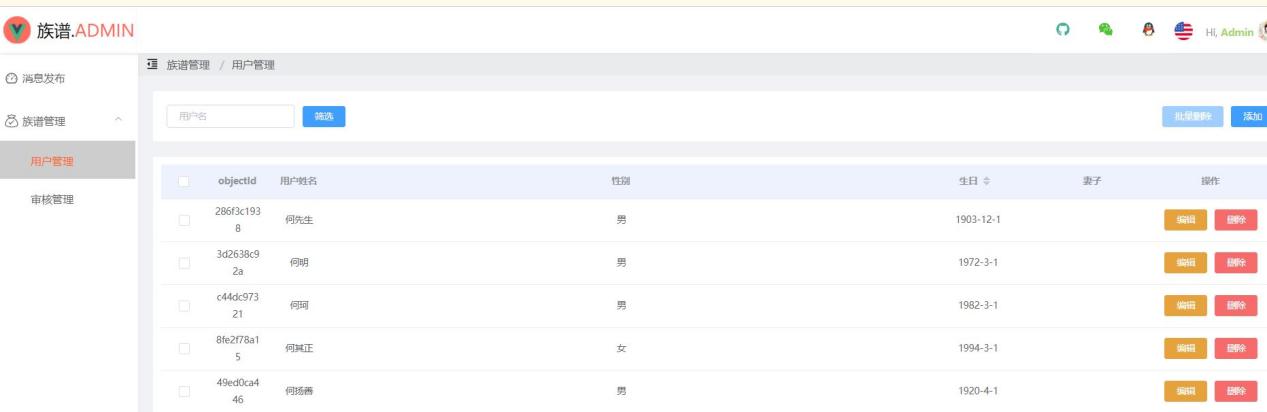


图4-7 族谱信息显示界面

4.1.2 软件接口

微信小程序端：

1. 微信小程序端与服务器之间通过https协议使用wx.request接口进行数据上传和数据下载。

2. 微信端用户通过wx.login接口登录小程序，通过微信内置函数wx.getUserInfo去登陆注册。

3. 微信端小程序通过wx.getStorage/wx.getStorageSync读取本地缓存，通过wx.setStorage/wx.setStorageSync写数据到缓存。

4. 若需查询族谱数据库，则需通过Bmob.Query(tableName).find()函数去进行读取。

5. 在网络软件接口方面，使用一种无差错的传输协议，采用滑动窗口方式对数据库进行网络传输以及接收。

网页端接口：

使用Bmob.Object.extend方法来创建新的Bmob.Object子类

1. 管理员网页端通过Bmob.Query()实现提取数据库的数据的功能，实现数据的阅览，读取。
2. 管理员网页端通过query.save()实现对数据库进行增加与修改。任何Bmob.Object子类都有save方法可以用于将当前对象的内容保存到服务端
3. 管理员网页端通过query.destroy()对数据库进行删除。

4.2 内部接口

内部接口方面，系统内部与数据库接口均为SQL链接，客户端在产生请求时，可以借由代码的接口逻辑与服务器建立链接。其次各模块之间采用参数传递、返回值的方式进行信息传递。具体的参数结构将在下面数据结构设计的内容中说明。接口传递的信息将是以数据结构封装了的数据，以参数传递或返回值的形式在各模块间传输。

5 系统数据结构设计

5.1 逻辑结构设计

系统数据库所维护的数据表样式，这里的逻辑结构是在数据库中真实存储的结构，没有数据类型说明，如下表5-1所示为数据库根节点存储逻辑结构。

表5-1 数据库根节点逻辑结构

|  |  |
| --- | --- |
| ObjectId | RelationId |

5.2 物理结构设计

物理数据结构设计主要是设计数据在模块中的表示形式，以结构化的方式表示，如下表5-2，5-3，5-4，5-5所示。

表5-2 信息录入表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 注释 |
| ObjectId | String | 无 | 唯一识别id（自动生成） |
| wife | String | “” | 妻子的id |
| level | Number | 无 | 辈分 |
| username | String | “” | 名字 |
| is\_wife | Boolean | False | 是否是别人的妻子 |
| children | Array | [] | 孩子的id数组 |
| gender | String | 无 | 性别 |
| birthday | String | 无 | 生日,YYYY-MM-DD格式 |

表5-3 修改信息审核表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 注释 |
| ObjectId | String | 无 | 唯一识别id（自动生成） |
| operation | Number | 无 | 0表示插入，把Node1插入到Node2的relation关系处，1表示删除，删除Node1，2表示更新 |
| Node1 | String | 无 | 第一个Person的ObjectId |
| Node2 | String | 无 | 第二个Person的ObjectId |
| UpdateMsg | Object | 无 | 以js的object类型，携带需要更新的信息，如{“username”:“何先生”,“birthday”:“2001-01-01”} |
| Relation | String | 无 | 参数，表述第一个节点跟第二个节点的关系，子女/配偶 |

表5-4 族谱文件根节点信息

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 注释 |
| ObjectId | String | 无 | 唯一识别id（自动生成） |
| RelationId | String | 无 | 对应在Person表的ObjectId |

表5-5 通知活动信息

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 注释 |
| ObjectId | String | 无 | 唯一识别id（自动生成） |
| title | String | 无 | 活动标题 |
| mainBody | String | 无 | 活动正文 |
| startTime | String | 当前北京时间 | 起始时间 |
| endTime | String | 当前北京时间 | 结束时间 |
| place | String | 无 | 地点 |
| undertaker | String | 无 | 举办方 |

5.3 数据结构与程序的关系

物理数据结构主要用于各模块之间函数的信息传递，接口传递信息将是以数据结构封装的数据以参数传递或者返回值的形式在各个模块中传输。

服务器程序在对录入族谱数据信息时的表（逻辑结构设计中的表）进行修改，对查询关系时的表进行联表查询等。同时将查询返回的结果送入显示模块或者打印模块中准备打印等。

6 运行设计

6.1 运行模块的组合

1. 普通族人用户：主要涉及查看族谱信息等模块，涉及到数据库的调用。

2. 管理员：主要涉及基本数据的设定、编辑族谱信息和导入等模块，涉及到了数据库的查询、修改和更新。

6.2 运行控制

1. 用户的登陆需要进行姓名、性别、地区等相关信息进行匹配族谱信息。

2. 用户申请修改族谱信息操作需要管理员核对审批后执行。

3. 管理员登陆后可以对族谱信息进行设置和修改。

6.3 运行时间

从占用时间角度来看，网页端系统打开链接之后就处于占用系统时间状态，一般运行次数较多，占用数据库使用时间的2/3。微信小程序端打开后即占用系统时间，退出时即结束进程，一般运行次数较少，占用数据库使用时间的1/3。其次，在需求分析中，对运行时间的要求较为严格的必须对做出的操作有较快的反应。

7 出错处理设计

7.1 出错输出信息

在出错处理方面，采用弹窗提醒进行输入确认的方式，争取在输入时及时解决出错问题，用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式以及含意。如下表7-1所示：

表7-1 出错信息表

|  |  |  |
| --- | --- | --- |
| 错误类型 | | 错误提示 |
| 输入信息错误 | 输入为空，姓名输入错误 | 输入框旁有红色感叹号作为提示，点击感叹号会显示详细出错原因，此时无法进行提交。 |
| 关系、性别输入错误 | 若存在婚姻关系输入时，若输入均为同种性别，此时会输出“暂不支持同性结婚”。 |
| 出生日期输入错误 | 结合生物学知识对年龄进行判断，若不符合逻辑则会输出“男性最小剩余年龄在12-14岁，请认真填写”。 |
| 查询模块出错 | 未完整输入两个待查询节点信息 | 输出显示”请填写节点信息“。 |
| 活动消息模块出错 | 活动输入时间错误 | 输出”起始时间大于等于结束时间，输入有误“。 |
| 活动信息输入不完整 | 提示未填写的信息。 |

7.2 出错处理对策

（1）由于我们数据库服务器由Bomb官方提供，意味着我们不需要关心服务器是否会受到攻击的问题，即服务器安全问题已经默认转交给提供服务器的官方进行解决。为了保护好数据，我们只需要定期对数据库进行备份和维护，将一些可能的异常保存的信息进行删除。

（2）由于出错的可能性基本在输入阶段通过弹窗提示修改和输入出错时禁止提交的方式在前期得到有效的阻止，因而对于数据格式等可能出错的地方只需定期进行适当的数据库内容检查即可。

（3）对于程序中可能出错的地方，使用try、catch语句，即例外处理。通过运用例外处理技术，我们可以实现用结构化的方式来响应错误事件的发生，让例外处理代码与正常脚本代码科学分离。代码使用格式如下：

try {  
　　 // 此处是可能产生例外的语句  
 } catch(error) {  
　　 // 此处是负责例外处理的语句  
 }

特别的，针对于微信小程序开发而言，可以在代码中可能出错的地方加入官方提供的回调函数：利用success处理正确运行后应该执行的语句，利用fail来处理出错之后应该运行的语句，也可以使用complete来让语句强制运行。使用示例如下：

wx.cloud.callFunction({

name:"test",

data:{

a:10,

b:20

},

success(res){

console.log("success",res);

},

fail(res){

console.log("fail",res)

},

complete(res){

console.log("complete",res)

}

})

8 安全保密设计

数据库我们采用bomb后端云进行存储，其官方提供了传输层面、应用层次、发布层次等不同粒度的权限控制的方式，确保用户数据的安全。

8.1 数据传输层面

Bomb后端云官方为了进一步保证数据在传输层面的安全性，对数据进行了加密封装，采用了requestId + timestamp + Application Key的一次性对称加密算法和服务端主动防御的技术，确保数据在传输过程中能够不被Fiddler、Wireshark等抓包工具恶意抓取进行分析破坏。

8.2 应用层级

在应用层级上，创建应用时Bmob会自动给每个应用分配三个Key，分别是Application Key, 、REST API Key 、Master Key：

* Application Key 应用程序Key：主要是iOS/Android/Unity/Cocos2d-x等SDK开发测试和发布时，用于请求、数据的凭证。
* REST API Key：主要是REST API接口开发测试和发布时，用于请求数据的凭证。
* Master Key：超级权限Key，开发者内部使用，使用此Key请求数据，表、对象、ACL等其他层次的权限不起作用。

8.3 发布阶段层级

当程序进入发布阶段，只需要进入到这个程序的Bmob后台，依次点击“应用设置->安全认证”，将程序的签名/Bundle ID信息进行填写，就可以确保数据的安全。

并且，为了避免普通用户随意修改增删它人的信息，系统管理保证了只有授权的用户（管理员）才能进入系统对它人数据进行操作，而只对普通用户仅能显示可读取的信息。因此整体的系统安全保密性较高。

9 维护设计

数据维护方面主要对服务器上的数据库数据进行维护，由于Bomb官方提供了客户端与服务器之间的一个长连接的维护，使得消息更新时由服务端可以主动推送消息，所以数据库内外数据的一致性可以得到足够的保证。

但是，由于Bomb官方没有提供乐观锁等功能，且在数据库层面只有Number类型数据在使用原子计算器时才可以实现用户输入时锁定表，输入完成后解锁表的功能，所以我们需要自行实现上锁操作。因此，我们引入了申请表，对于数据我们需要审批通过之后才可以更新，实际上相当于实现了锁的功能。而对于消息发布功能，由于不同消息的发布时不相关的，实际上只会对数据条目进行添加，删除等功能只有管理员才可进行操作，故而也无需在代码层级上锁。

最后，让专门的数据库管理人员进行定期的数据维护，进行错误的排查即可。