|  |
| --- |
| 四川师范大学 |
| 软件设计说明书 |
| “HRAM-SNU”系统软件设计说明书 |

|  |
| --- |
| 团队名称：HR Hunters  2016-4-7 |

目录

[文档变更记录 0](#_Toc447471676)

[引言 0](#_Toc447471677)

[1.1 项目介绍 0](#_Toc447471678)

[1.2 开发团队 0](#_Toc447471679)

[1.3 使用的文字处理和绘图工具 0](#_Toc447471680)

[静态体系设计 0](#_Toc447471681)

[2.1 全局数据结构 0](#_Toc447471682)

[2.2 模块设计 1](#_Toc447471683)

[2.2.1 模块划分 1](#_Toc447471684)

[2.2.2 系统类图及说明 1](#_Toc447471685)

[2.3 界面设计 1](#_Toc447471686)

[2.4 数据库设计 1](#_Toc447471687)

[动态结构 1](#_Toc447471688)

[3.1 用例顺序图 1](#_Toc447471689)

[3.2 需说明的对象状态图 1](#_Toc447471690)

[3.3 内外部接口 1](#_Toc447471691)

[其他设计 1](#_Toc447471692)

[4.1 安全性设计 1](#_Toc447471693)

[4.1.1 说明 1](#_Toc447471694)

[4.1.2 数据传输设计 2](#_Toc447471695)

[4.1.3 身份验证设计 2](#_Toc447471696)

[4.2 系统错误处理 2](#_Toc447471697)

[4.3 系统性能设计 2](#_Toc447471698)

[附录 2](#_Toc447471699)

[5.1 词汇表 2](#_Toc447471700)

[5.2 参考文献 3](#_Toc447471701)

# 文档变更记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **变更（+/-）说明** | **作者** | **版本号** | **日期** | **批准** |
| 1 | 初稿 | 徐志坚 | 1.0 | 2016/4/5 | 其他成员 |
| 2 |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# 引言

## 1.1 项目介绍

人力资源保障协会后台管理系统（简称：HRAM-SNU）是一套以网站服务为基础，手机移动客户端（APP软件）、微信公众号等多模式为补充的在线综合性服务系统。向用人单位及广大人民群众提供高效便捷的人力资源和社会保障服务，最终实现全在线、全时空的“互联网+人力资源和社会保障服务”模式。

## 1.2 开发团队

老板：廖雪花

项目经理：徐志坚

系统架构师：徐志坚

软件工程师：徐志坚、黄足英

测试工程师：徐志坚、黄足英

需求分析师：徐志坚、黄足英

## 1.3 使用的文字处理和绘图工具

文字处理软件：Microsoft Office Word 2013

绘图工具：Microsoft Office Visio 2013

# 静态体系设计

## 2.1 全局数据结构

public static final int POSITION\_TOP\_LEFT = 1;

public static final int POSITION\_TOP\_RIGHT = 2;

public static final int POSITION\_BOTTOM\_LEFT = 3;

public static final int POSITION\_BOTTOM\_RIGHT = 4;

public static final int POSITION\_CENTER = 5;

private static final int DEFAULT\_MARGIN\_DIP = 5;

private static final int DEFAULT\_LR\_PADDING\_DIP = 5;

private static final int DEFAULT\_CORNER\_RADIUS\_DIP = 8;

private static final int DEFAULT\_POSITION = POSITION\_TOP\_RIGHT;

private static final int DEFAULT\_BADGE\_COLOR = Color.parseColor("#CCFF0000"); //Color.RED;

private static final int DEFAULT\_TEXT\_COLOR = Color.WHITE;

全局变量：

1.用application来保存全局变量

2.使用普通的类Data.class来保存全局变量

常用数据结构：

1.SimpleArrayMap<K,V>与ArrayMap<K,V>

实质上ArrayMap继承自SimpleArrayMap,主要是为了实现像HashMap一样的api方法，让习惯使用HashMap的开发者感觉不到差异，本质上是SimpleArrayMap+Map的再封装。

2.SparseArray<T>与SparseArrayCompat<T>和LongSparseArray<T>

这3个类中，前2个基本上是同一类，只不过第二个类有removeAt方法，第三个是Long类型的。

这3个类也是用来代替HashMap,只不过他们的键(key)的类型是整型Integer或者Long类型，在实际开发中，如月份缩写的映射，或者进行文件缓存映射，viewHolder都特别适用

3.AtomicFile

AtomicFile首先不是用来代替File的，而是作为File的辅助类从在， AtomicFile的作用是实现事务性原子操作，即文件读写必须完整，适合多线程中的文件读写操作。

用来实现多线程中的文件读写的安全操作

4. list(列表)

5.array(数组)

6.枚举结构

## 2.2 模块设计

### 2.2.1 模块划分

1.个人信息操作：

1.登录注册模块

2.个人信息设置模块

3.系统设置模块

2.多篇显示操作：

4.首页显示模块(包含搜索功能)

5.“蚊香圈”显示模块

3.单篇文章操作：

6.提问模块

7.经验分享模块

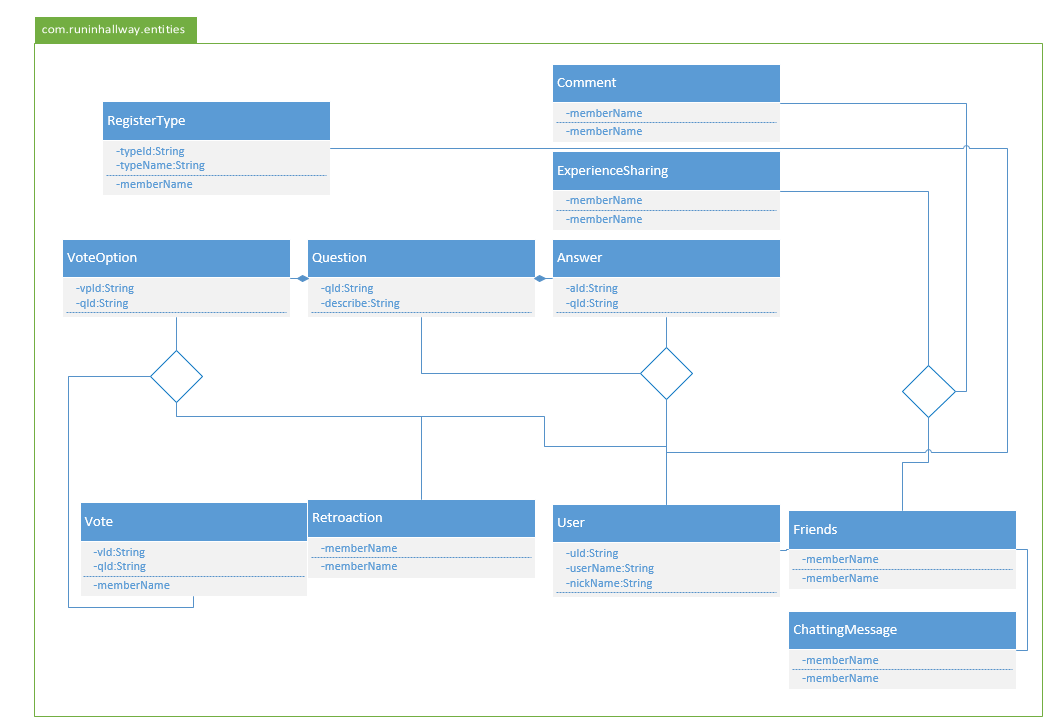
8.发起投票模块

9.回答问题模块

10.参与投票模块

11.查看经验分享模块

### 2.2.2 系统类图及说明



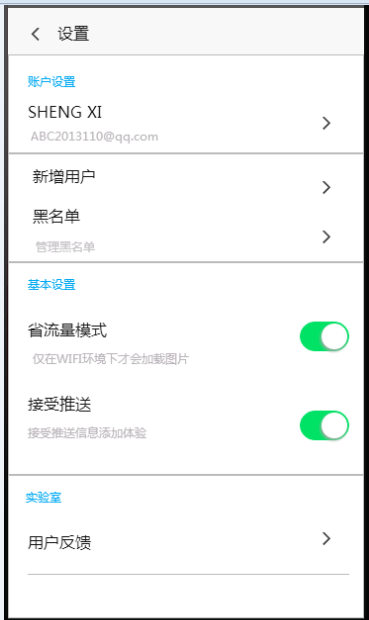
## 2.3 界面设计

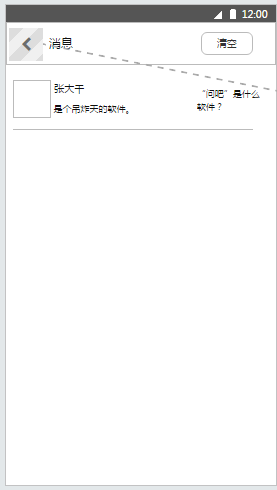
所有界面：

**

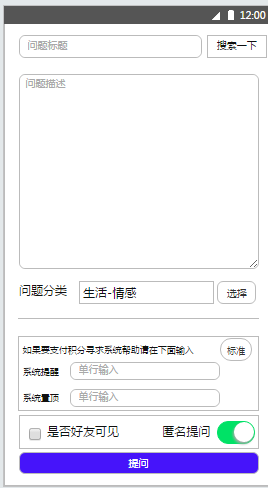
个人信息界面：

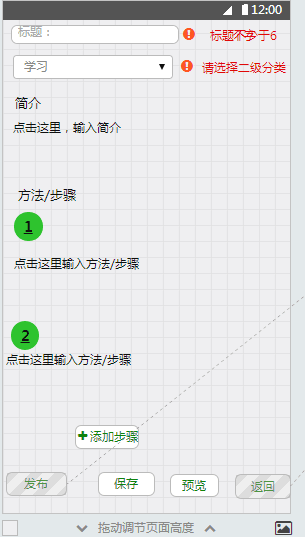
* *

**

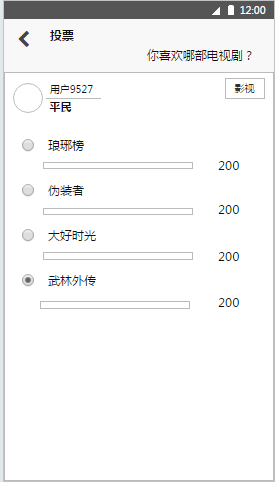
**

发起界面：

**

**

参与界面:

**

**

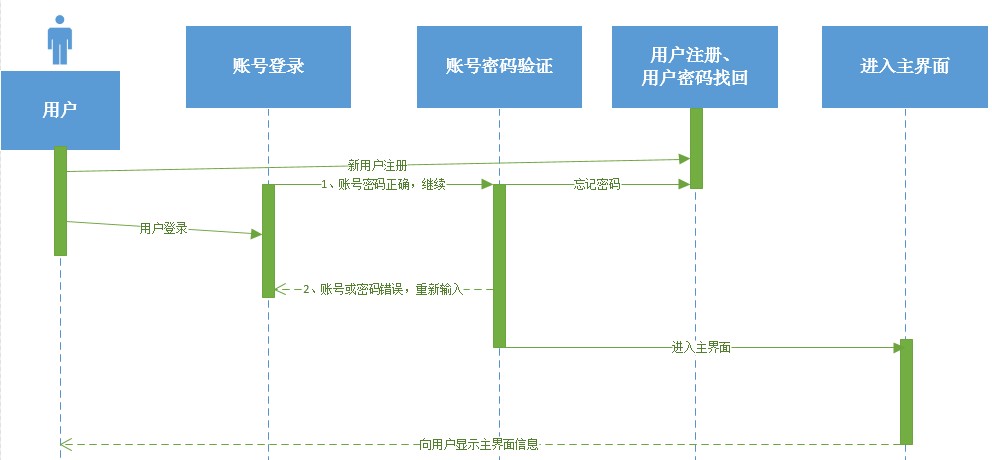
## 2.4 数据库设计

|  |  |  |
| --- | --- | --- |
| **1.系统模块表(T\_SysModule)** | | |
| 变量名称 | 变量类型 | 变量需求 |
| pk\_id | nvarchar(50) | 用户注册时系统自动产生的用户编号 |
| U\_LoginTypeID | nvarchar(50) | 用户登录方式（手机/邮箱） |
| U\_LoginTypeValue | nvarchar(50) | 用户手机注册的手机号（11位） |
| **2.用户信息表/账户信息表(T\_User)** | | |
| 变量名称 | 变量类型 | 变量需求 |
| pk\_id | nvarchar(50) | 用户注册时系统自动产生的用户编号（自增） |
| U\_LoginName | nvarchar(50) | 用户注册时设置的昵称 |
| U\_LoginTypeID | nvarchar(50) | 用户设置的头像地址，没有设置则为默认头像 |
| U\_Password | nvarchar(50) | 用户的性别 |
| U\_UserName | nvarchar(50) | 用户的生日 |
| U\_Phone | nvarchar(50) | 用户设置的职业 |
| U\_MaiBox | nvarchar(50) | 用户的爱好 |
| **3.用户类型表表(T\_UserType)** | | |
| 变量名称 | 变量类型 | 变量需求 |
| pk\_id | nvarchar(50) | 用户注册时系统自动产生的用户编号 |
| U\_LoginTypeID | nvarchar(50) | 用户登录方式（手机/邮箱） |
| U\_LoginTypeValue | nvarchar(50) | 用户手机注册的手机号（11位） |
| **3．问题表(T\_Question)** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | Int | 用户的编号 |
| Q\_Anonymous\_IS | Boolean | 是否匿名提问 |
| Q\_ID | Int（主码） | 系统自动为问题生成的编号(时间+随机数) |
| Q\_Category | Int | 涉及问题的分类 |
| Q\_Title | Char（） | 提问时为问题写的标题 |
| Q\_Text | Text | 提问的内容 |
| Q\_Up\_Num | Int | 点赞的个数 |
| Q\_Down\_Num | Int | 反对的个数 |
| Q\_Collection\_Num | Int | 收藏的数量 |
| Q\_Share\_Num | Int | 分享的数量 |
| Q\_Browse\_Sum | Int | 所有的浏览量 |
| Q\_Browse\_Day | Int | 当日浏览量 |
| Q\_Browse\_Month | Int | 最近一月浏览量 |
| Q\_Browse\_Year | Int | 最近一年浏览量 |
| Q\_Time\_First | Date | 问题什么时候提出的 |
| Q\_Time\_Near | Date | 最近一次修改的时间 |
| **4．经验分享表(T\_** **Experience\_Sharing)** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | Int | 用户的编号 |
| ES\_ID | int（主码） | 系统自动为经验分享生成的编号(时间+随机数) |
| ES\_Category | Int | 涉及经验分享的类别 |
| ES\_Title | Char（） | 分享经验时为问题写的标题 |
| ES\_Text | Text | 分享经验的内容（能否在里面插入图片等其他特殊类型） |
| ES\_Up\_Num | Int | 点赞的个数 |
| ES\_Down\_Num | Int | 反对的个数 |
| ES\_Collection\_Num | Int | 收藏的数量 |
| ES\_Share\_Num | Int | 分享的数量 |
| ES\_Browse\_Sum | Int | 所有的浏览量 |
| ES\_Browse\_Day | Int | 当日浏览量 |
| ES\_Browse\_Month | Int | 最近一月浏览量 |
| ES\_Browse\_Year | Int | 最近一年浏览量 |
| ES\_Time\_First | Date | 问题什么时候提出的 |
| ES\_Time\_Near | Date | 最近一次修改的时间 |
| **5．投票表（T\_Vote）** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | int | 用户的编号 |
| V\_Anonymous\_IS | boolean | 是否匿名 |
| V\_ID | int（主码） | 系统自动为投票生成的编号 |
| V\_Title | Char() | 发起投票的标题 |
| V\_Text | Char() | 发起投票的问题（详细、说明问题,可为空） |
| V\_ Category | int | 投票所属的类型 |
| V\_Time\_Start | Date | 投票发起的时间 |
| V\_Time\_End | Date | 到了截止时间，他人不能参与 |
| **6.投票选项表(T\_Vote\_Option)** | | |
| 变量名称 | 变量类型 | 变量需求 |
| V\_ID | int（主码） | 系统自动为投票生成的编号 |
| VO\_Option | Int | 选项号 |
| VO\_Option\_Text | Char | 选项内容 |
| VO\_Option\_Num | Int | 该选项的票数 |
| **7.分类表（T\_Category）该表是开发者先行填入,三级分类组合即为类别号** | | |
| 变量名称 | 变量类型 | 变量需求 |
| Category\_First\_ID | int(3) | 一级分类ID |
| Category\_First\_Name | char | 一级分类名 |
| Category\_Second\_ID | int(3) | 二级分类ID |
| Category\_Second\_Name | char | 二级分类名 |
| Category\_Third\_ID | int(3) | 三级分类ID |
| Category\_Third\_Name | char | 三级分类名 |
| **8.参与回答表（T\_Jion\_Question）** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | Int | 参与用户的编号 |
| JQ\_ID | int（主码） | 系统自动为问题生成的编号 |
| JQ\_Anonymous\_IS | boolean | 是否匿名 |
| JQ\_TH | Int | 第几个回答 |
| JQ\_Text | Text | 内容 |
| JQ\_Time | Date | 参与的时间（精确到天） |
| JQ\_Great | boolean | 是否被采纳 |
| JQ\_IS | Boolean | 是否参与（限制一次回答） |
| **9.参与经验分享表(T\_Jion\_ES)** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | Int | 参与用户的编号 |
| ES\_ID | int（主码） | 系统自动为经验分享生成的编号 |
| JES\_Anonymous\_IS | boolean | 是否匿名 |
| JES\_TH | Int | 第几个评论 |
| JES\_Text | Text | 内容 |
| JES\_Time | Date | 参与的时间（精确到天） |
| JES\_IS | Boolean | 是否参与（限制一次评论） |
| **10.参与投票表(T\_Jion\_Vote)** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | Int | 参与者用户的编号 |
| V\_ID | int（主码） | 系统自动为问题生成的编号 |
| VO\_Option | Int | 选择的选项 |
| JVO\_IS | Boolean | 是否参与（限制一次投票） |
| **11.追问追答表（T\_In\_Ask）** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | Int | 发起者用户的编号 |
| Q\_ID | int（主码） | 系统自动为问题生成的编号 |
| IA\_Anonymous\_IS | boolean | 是否匿名 |
| IA\_IS | boolean | 提问方还是回答方 |
| IA\_TH | Int | 位于该回答第几层 |
| IA\_TH\_Count | Int | 第几次（限制追问的次数） |
| IA\_Text | Text | 追问的内容 |
| IA\_Time | Date | 参与的时间（精确到天） |
| IA\_FromUserID | Int | （发送者ID）指向用户表 |
| IA\_ToUserID | Int | (接受者ID)指向用户表 |
| **12.回复表（T\_In\_ES）** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | Int | 发起者用户的编号 |
| ES\_ID | int（主码） | 系统自动为问题生成的编号 |
| IES\_Anonymous\_IS | boolean | 是否匿名 |
| IES\_IS | boolean | 提问方还是回答方 |
| IES\_TH | Int | 第几层，用于显示次序 |
| IES\_Text | Text | 回复的内容 |
| IES\_Time | Date | 参与的时间（精确到天） |
| IES\_FromUserID | Int | （发送者ID）指向用户表 |
| IES\_ToUserID | Int | (接受者ID)指向用户表 |
| **13．未读消息列表（T\_Wait\_Message）** | | |
| 变量名称 | 变量类型 | 变量需求 |
| User\_ID | Int | 用户编号 |
| WM\_ID | Int | 未读信息标号（自增，查看后减去） |
| WM\_FromUserID | Int | （发来信息的ID,系统默认为0）指向用户表 |
| WM\_ToUserID | Int | (接收信息ID)指向用户表 |
| WM\_Kind | char | 提醒消息的类型（系统/投票/提问/经验分享/好友消息） |

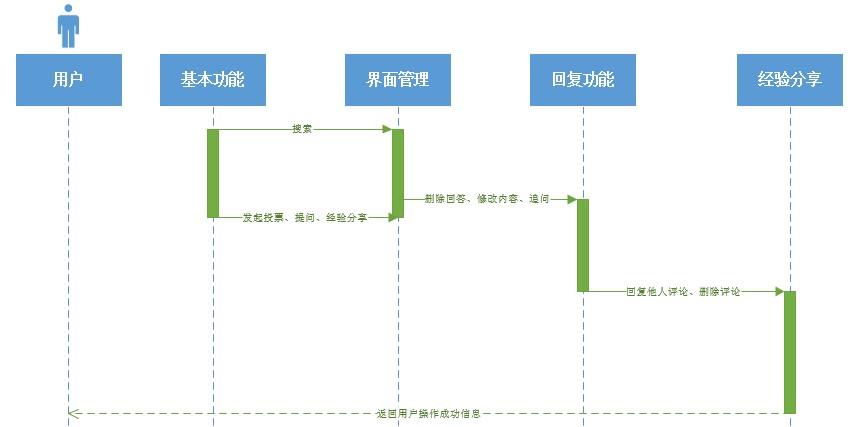
# 动态结构

## 3.1 用例顺序图

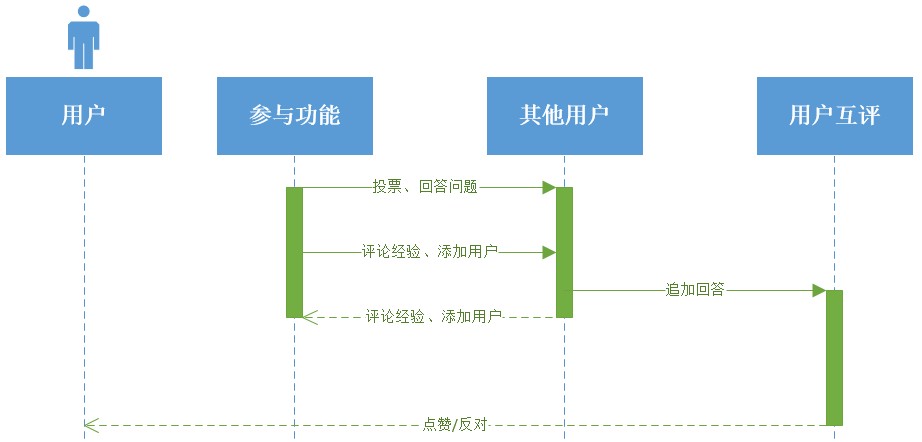
登陆注册用例顺序图：



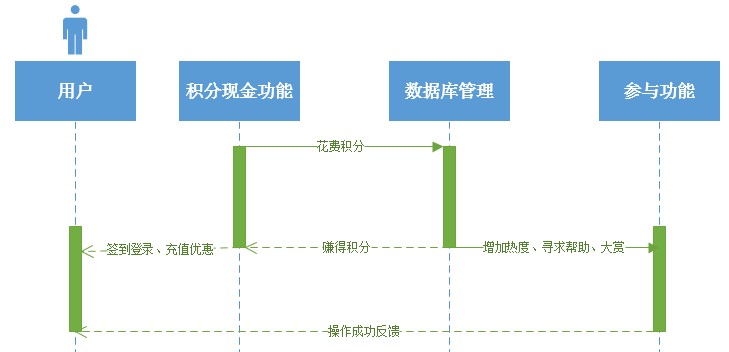
基本功能用例顺序图：



参与功能用例顺序图：

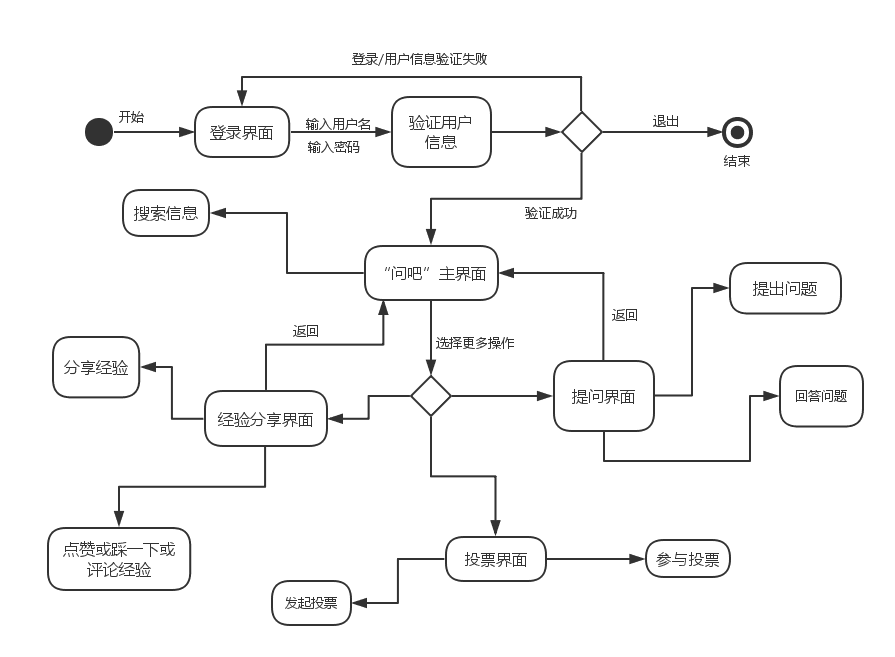


积分现金功能用例顺序图：



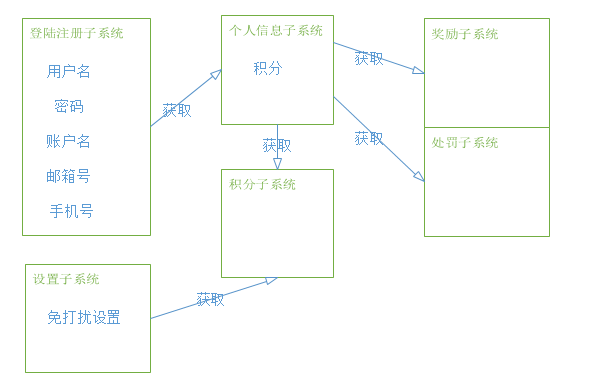
## 3.2 需说明的对象状态图

对象状态图：



## 3.3 内外部接口

以下图表示：



# 其他设计

## 4.1 安全性设计

### 4.1.1 说明

该部分介绍了数据传输用的方法，身份验证的方法，以及在系统出错时的处理。

在奔溃后在编辑的内容应存储到草稿，并且实时备份。

### 4.1.2 数据传输设计

采用Web Sever: 通过tomcat发布Web Sever,然后在Android端通过ksoap2-android包访问Web Sever数据.

### 4.1.3 身份验证设计

我们项目采用用户名/密码方式进行用户验证和QQ、微信授权登陆。

用户名/密码方式：用户登录时，将用户的用户名和密码以加密的方式送到认证区域，与认证区域保存的用户信息进行比对。如果验证通过，读取用户的分类，大神，专家，平民，以便之后的操作，允许用户进行随后的访问操作，反之则拒绝用户下一步的操作。

QQ、微信授权登陆：自动获取本地已存在的QQ、微信账号，通过用户自主授权登陆游戏。用户游戏相关信息存储在用户授权的QQ或微信账号对应的账号下。

## 4.2 系统错误处理

为了在系统出现异常情况下给用户以明确的提示，可采用两种方式予以提示：

1．使用Android的Toast.makeText()函数直接提示，这主要在输入或修改的情况下使用；

2．使用统一的错误界面提示，该界面对应于error.java页面。错误界面样式如下图所示：

错误提示界面



## 4.3 系统性能设计

搜索到搜索结果出现，总用时不超过2秒钟。

其他各个界面的转换相关点击，总用时不超过1秒钟。

# 附录

## 5.1 词汇表

定义系统或产品中涉及的重要术语，为读者在阅读文档时提供必要的参考信息。

|  |  |  |
| --- | --- | --- |
| **序号** | **术语或缩略语** | **说明性定义** |
| 1 | PM | Project Manager,项目经理 |
| 2 | GUI | 图形用户界面 |
| 3 | Android | 安卓应用程序 |
| 4 | UI | （用户界面）的简称。泛指用户的操作界面，UI设计主要指界面的样式，美观程度。 |
| 5 | Android | 安卓系统。 |
| 6 | SDK | 软件开发工具包。 |
| 7 | Java | 一种可以撰写跨平台应用程序的面向对象的程序设计语言。 |
| 8 | MySQL | MySQL是一个关系型数据库管理系统。 |
| 9 | *Toast* | Toast是Android中用来显示显示信息的一种机制 |
| 10 | Web Sever | 。WEB服务器也称为WWW(WORLD WIDE WEB)服务器，主要功能是提供网上信息浏览服务 |
| 11 | tomcat | Tomcat是Apache 软件基金会（Apache Software Foundation）的Jakarta 项目中的一个核心项目，由Apache、Sun 和其他一些公司及个人共同开发而成。 |
| 12 | ksoap2-android | KSOAP2的步骤来编写调用WebService的客户端程序 |

## 5.2 参考文献

《软件测试》（机械工业出版社）

《软件工程》（机械工业出版社）

《软件工程与管理》（机械工业出版社）

《Android应用程序设计》（清华大学出版社）

等。