**Amazing Blog个人博客空间**

**概要设计说明书**

东软实训第13组

组长：王思哲

组员：张海涛

陈鹏旭

郑小妹

目录

[1 引言(郑小妹撰写) 2](#_Toc517895511)

[1.1 编写目的 2](#_Toc517895512)

[1.2 背景 2](#_Toc517895513)

[1.3 定义 2](#_Toc517895514)

[1.4 参考资料 3](#_Toc517895515)

[2 总体设计 3](#_Toc517895516)

[2.1 需求规定 (陈鹏旭撰写) 3](#_Toc517895517)

[2.2 运行环境 (张海涛撰写) 6](#_Toc517895518)

[2.3 基本设计概念和处理流程 (郑小妹撰写) 7](#_Toc517895519)

[2.4 系统结构 (王思哲撰写) 10](#_Toc517895520)

[2.5 功能需求与程序的关系（陈鹏旭撰写） 11](#_Toc517895521)

[2.6 人工处理过程（张海涛撰写） 12](#_Toc517895522)

[2.7 类图 13](#_Toc517895523)

[2.8 尚未问决的问题 （王思哲撰写） 15](#_Toc517895524)

[3 接口设计 16](#_Toc517895525)

[3.1 用户接口（郑小妹撰写） 16](#_Toc517895526)

[3.2 外部接口（张海涛撰写） 16](#_Toc517895527)

[3.3 内部接口 (陈鹏旭撰写) 16](#_Toc517895528)

[4 运行设计（张海涛、陈鹏旭撰写) 22](#_Toc517895529)

[4.1 运行模块组合 22](#_Toc517895530)

[4.2 运行控制 23](#_Toc517895531)

[4.3 运行时间 40](#_Toc517895532)

[5 系统数据结构 41](#_Toc517895533)

[5.1 逻辑结构设计要点 41](#_Toc517895534)

[5.2物理结构设计要点 42](#_Toc517895535)

[5.3数据与结构 44](#_Toc517895536)

[6系统出错处理设计(前端，后台，数据库负责人都写) 47](#_Toc517895537)

[6.1出错信息 47](#_Toc517895538)

[6.2补救措施 48](#_Toc517895539)

[6.3系统维护设计 49](#_Toc517895540)

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 作者 | 版本描述 | 日期 |
| 1.0 | 王思哲 | 内容整合，初稿 | 2018/6/27 |
| 2.0 | 王思哲 | 修订完善 | 2018/7/7 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1 引言(郑小妹撰写)

1.1 编写目的

a.     该文档列出《AmazingBlog》博客空间项目开发的总体要求，安排规划及进度要求。

b.     该文档对项目的功能、性能等进行具体描述，作为软件开发人员进行项目开发的指导。

c.     作为验收标准，项目后续测试的依据。

1.2 背景

a.   项目名称：Amazing Blog

b.   提出者及开发人员：JavaWeb第13组成员

c.   用户：想拥有可存放、展示文字与个人介绍的博客空间的广大网民

1.3 定义

个人空间：提供日志管理，留言板，个人动态等功能的网站。

日志：带标题的文章，可接收评论。

1.4 参考资料

a． 东软提供的《概要说明书》模板。

2 总体设计

2.1 需求规定 (陈鹏旭撰写)

功能需求：

|  |  |
| --- | --- |
| 功能名称 | 备注 |
| 注册，登录及找回密码 func\_Login | 注册信息，并能够登录到个人博客。 |
| 主页 func\_Page | 用于显示个人基本资料及头像，最新留言，个人心情列表。 |
| 日志 func\_Blog | 日志分类，及日志管理。 |
| 留言板 func\_Board | 留言信息及留言回复。 |
| 个人档 func\_Intro | 个人信息维护及个人照片上传。 |
| 个人中心 func\_Center | 用于好友信息，朋友动态信息，发表心情，空间互访，好友推荐。 |

注册登录以及找回密码：

|  |  |  |
| --- | --- | --- |
| 输入流 | 处理 | 输出 |
| 用户注册页，输入的用户账号密码邮箱或者安全问题答案 | 判断合法性：查询数据库是否已存在，特殊字符等 | 合法账号密码存入数据库，进入注册成功界面，跳转到主页 |
| 登录页，用户账号密码 | 判断合法性：查询数据库判断账号密码正确性 | 登录成功界面，跳转到主页 |
| 找回密码页，账号或者安全问题的回答 | 判断安全问题的回答 | 向邮箱发送密码，跳转到登录界面 |

主页：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 子功能 | 描述 | 输入 | 处理 | 输出 |
| 查询个人主页基本资料信息 | 主要用于查询当前的个人主页信息 | 登录或者注册界面携带用户的信息进入主页，用户账号，id | 连接数据库查询个人信息 | 当前页面显示个人信息 |
| 新建个人主页信息 | 主要用于新建一个个人主页 | 当前页面点击新建,用户账号，id | 输入信息 | 新建成功并返回首页显示个人信息 |
| 修改个人主页信息 | 主要用于修改已有的个人主页信息 | 当前页面点击修改,用户账号，id | 连接数据库查询个人信息，并修改 | 修改成功并返回首页显示个人信息 |
| 删除个人主页信息 | 主要用于删除一个个人主页及相关信息 | 用户账号，id | 数据库里面删除 | 返回首页显示个人信息 |

日志：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 子功能 | 描述 | 输入 | 处理 | 输出 |
| 查询个人日志信息 | 主要用于查询当前的个人日志信息 | 点击查询 | 数据库查询 | 显示日志列表 |
| 新建个人日志信息 | 主要用于新建一个个人日志 | 新建 | 提交保存数据库 | 弹框成功提交，并重新进入个人日志界面（刷新） |
| 修改个人日志信息 | 主要用于修改已有的个人日志信息 | 修改 | 修改完毕，重新插入数据库 | 弹框成功提交，并重新进入个人日志界面（刷新） |
| 删除个人日志信息 | 主要用于删除一个个人日志及相关信息 | 删除 | 删除选中的个人日志 | 弹框成功提交，并重新进入个人日志界面（刷新） |

留言：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 子功能 | 描述 | 输入 | 处理 | 输出 |
| 查询留言板信息 | 主要用于查询当前的留言板信息 | 点击查询 | 数据库查询 | 显示留言板信息 |
| 新建留言板信息 | 主要用于新建一个留言板 | 新建 | 插入数据库 | 重新显示留言板信息界面 |
| 修改留言板信息 | 主要用于修改已有的留言板信息 | 修改 | 修改完毕，重新插入数据库 | 显示更新后的留言板信息界面 |
| 删除留言板信息 | 主要用于删除一个留言板及相关信息 | 删除 | 删除数据库中对应的留言信息 | 显示更新后的留言板信息界面 |

个人档：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 子功能 | 描述 | 输入 | 处理 | 输出 |
| 查询个人信息 | 主要用于查询当前的个人信息 | 点击查询 | 数据库查询 | 更新界面 |
| 新建个人信息 | 主要用于新建一个个人信息（新建个人信息及个人照片上传） | 新建 | 信息，以及图片存入数据库 | 更新界面 |
| 修改个人信息 | 主要用于修改已有的个人信息（个人信息维护及个人照片上传） | 修改 | 数据库信息更新 | 更新界面 |
| 删除个人信息 | 主要用于删除一个个人信息 | 删除 | 数据库信息更新 | 更新界面 |

个人中心：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 子功能 | 描述 | 输入 | 处理 | 输出 |
| 查询好友列表信息 | 主要用于查询当前的好友列表信息 | 点击查询 | 数据库查询 | 显示好友列表 |
| 删除个人好友信息 | 主要用于删除当前的好友列表中的好友 | 删除 | 数据库里面删除 | 更新好友列表 |
| 空间互访 | 主要用于访问好友的空间 | 访问 | 界面跳转 | 进入好友的主页 |
|  |  |  |  |  |
| 查询朋友动态信息 | 主要用于查询当前的朋友动态信息 | 查询 | 数据库查询，并显示信息列表 | 进入朋友动态界面 |
| 移除朋友动态信息 | 主要用于移除当前的朋友动态中的动态 | 移除 | 数据库删除 |  |
|  |  |  |  |  |
| 查询当前个人心情 | 主要用于查询当前的个人心情信息 | 查询 | 数据库查询 | 显示心情 |
| 新建个人心情 | 主要用于新建当前的个人信息 | 新建 | 新建心情添加进数据库 | 新建成功刷新心情列表 |
| 删除个人心情 | 主要用于删除个人心情信息记录 | 删除 | 数据库删除对应的心情 | 删除成功，刷新心情列表 |
|  |  |  |  |  |
| 查询好友推荐列表 | 主要用于查看当前的好友推荐信息 | 查询 | 数据库查询好友的好友 | 显示推荐列表 |
| 查看好友空间 | 主要用于查看推荐好友的空间信息 | 点击查看 | 链接进入好友空间 | 好友的空间主页 |
| 添加好友 | 主要用于添加好友 | 添加 | 数据库添加用户好友 | 返回好友推荐界面 |

2.2 运行环境 (张海涛撰写)

硬件环境：

CPU：3GHz及以上

内存：20G以上

软件环境：

操作系统：Microsoft Windows XP或更高版本

数据库: MySQL 5.x InnoDB

Web浏览器：Tomcat 6.0以上

支持浏览器：Internet Explorer 7.0及其以上版本

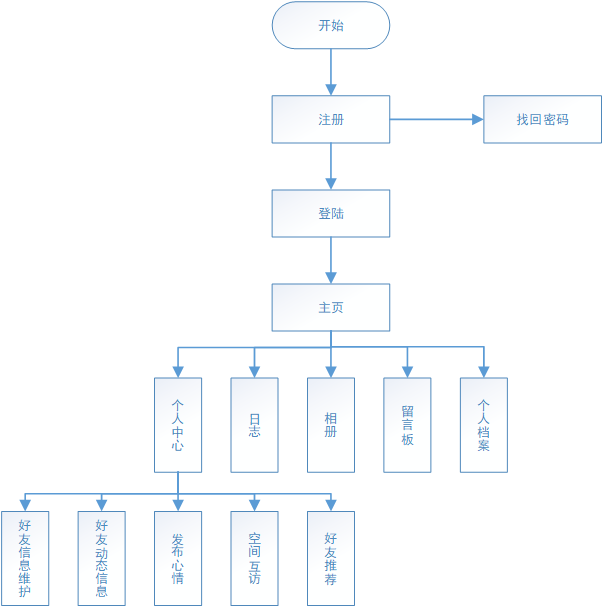
开发环境：MyEclipse 2014(jdk1.7)，Tomcat 8.x, MySQL 5.x InnoDB

2.3 基本设计概念和处理流程 (郑小妹撰写)

**基本概念：**

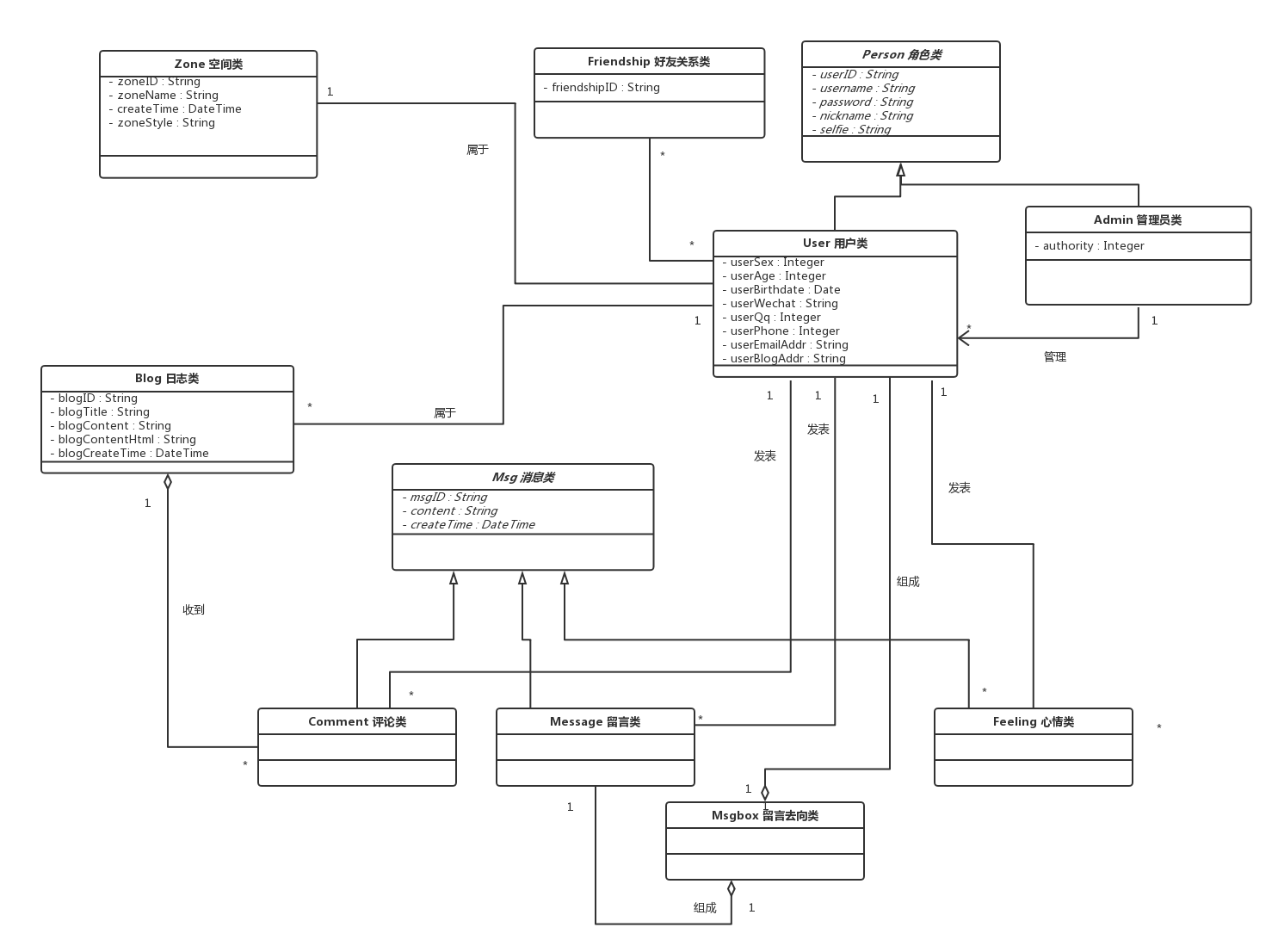
项目系统采用流行的B/S结构模式。系统的分析设计采用面向对象的技术，应用PowerDesigner和Processon等工具进行辅助设计。

**处理流程：**



|  |  |
| --- | --- |
| **模块** | **处理流程** |
| 用户注册登陆 | 用户根据个人情况即是否有账号来选择注册或是登陆；注册时需填写系统给出的必填个人信息；登陆时需要填写正确的用户名和密码 |
| 个人主页 | 用户可以在个人主页界面通过点击相关入口进行主页的查询、新建、修改、删除操作 |
| 个人日志 | 用户可以在个人日志界面通过点击相关入口进行个人日志的查询、新建、修改、删除操作 |
| 留言板 | 用户可以在好友留言表入口处查看留言；可以在好友日志或心情等地方创建留言；可以打开自己的留言记录进行修改已发布的留言；用户可以删除已发布的留言 |
| 个人档案管理 | 用户可以在个人信息界面通过点击相关入口进行个人资料的查询、修改、删除操作 |
| 好友列表 | 用户可以在个人主页处点击查看好友列表；在好友列表中删除某个好友；在好友列表中选择某个好友访问其空间 |
| 朋友动态 | 可以在动态界面点击某条动态进行查看；点击某条动态下的删除按钮删除好友动态信息 |
| 个人心情 | 在个人主页可显示当前心情；点击心情按钮新建个人心情；在个人心情列表删除个人心情 |
| 好友推荐 | 在好友列表界面显示好友推荐列表；点击好友列表中的某好友可查看其空间；点击添加按钮可以添加新好友 |
| 空间装扮 | 个人空间中显示当前空间装扮；主页有空间装扮入口，点击进入可以对空间装扮进行更换 |

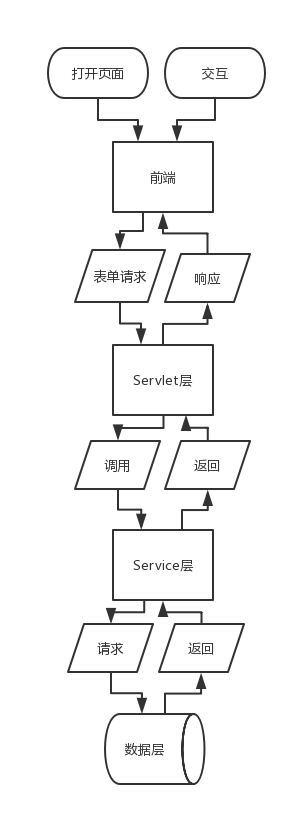
整体类图： (王思哲撰写)



2.4 系统结构 (王思哲撰写)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 标识符 | 模块名 | 模块功能描述 | 调用模块并接收 | 返回模块 |
| 01 | 前端（View） | 显示用户数据，并提供给用户可交互的组件 | 02 | 无 |
| 02 | Servlet（View） | 接收来自前端的表单或操作请求，直接返回页面或者调用Service层，得到数据后再返回到前端页面 | 03 | 01 |
| 03 | Service层  （Control） | 接收来自Servlet层请求，执行业务逻辑，向数据层获取数据，返回给Servlet层 | 04 | 02 |
| 04 | 数据层（Model） | 接收来自Service层的请求，并从数据库取出对应数据，返回ResultSet集合给Service层 | 无 | 03 |

各系统调用流程图：



2.5 功能需求与程序的关系（陈鹏旭撰写）

本条用一张如下的矩阵图说明各项功能需求的实现同各块程序的分配关系：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 用户管理 | 主页管理 | 日志管理 | 留言板管理 | 个人管理 | 个人中心 | 数据库 | 控制器 |
| 用户登录注册找回密码 | √ |  |  |  |  |  | √ | √ |
| 个人主页 |  | √ |  |  |  |  | √ | √ |
| 日志 |  |  | √ |  |  |  | √ | √ |
| 留言板 |  |  |  | √ |  |  | √ | √ |
| 个人档 |  |  |  |  | √ |  | √ | √ |
| 个人中心 |  |  |  |  |  | √ | √ | √ |

2.6 人工处理过程（张海涛撰写）

用户人工处理过程：

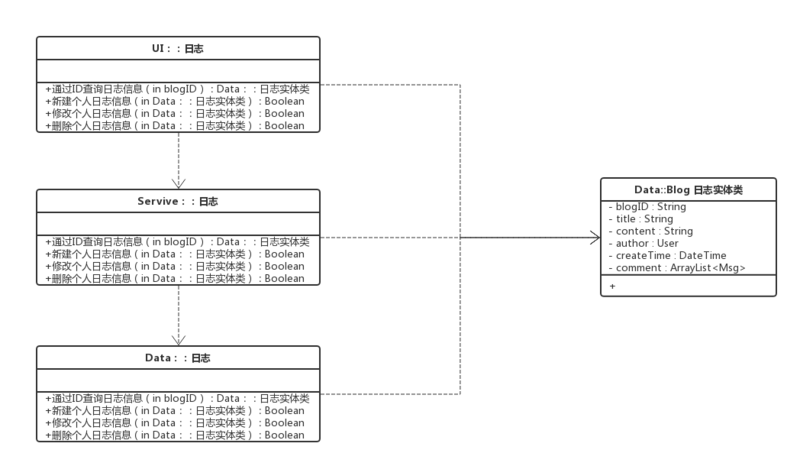
1. 用户登录输入用户名和密码
2. 用户注册输入详细个人信息
3. 用户注销自身登陆
4. 用户修改/删除已有个人信息
5. 用户编写/删除日志
6. 用户给某博客编写/删除评论
7. 用户给另一用户编写/删除留言
8. 用户添加及删除好友
9. 用户修改空间装扮

管理员人工处理过程：

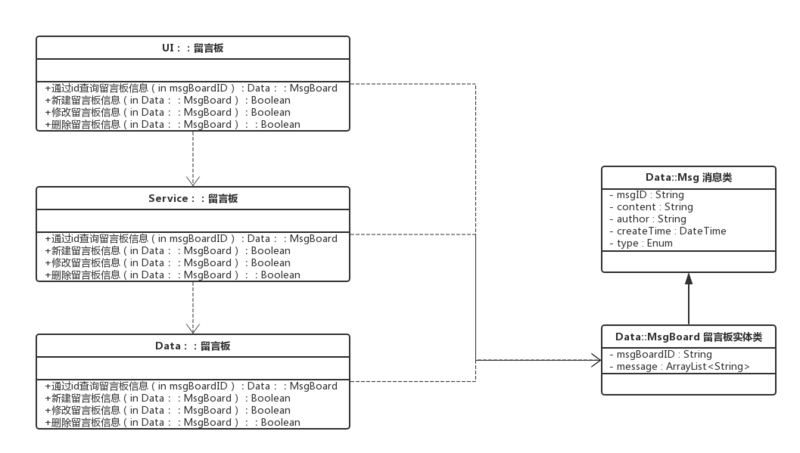
1. 管理员登录时用户名和密码输入
2. 管理员对用户各种信息权限的修改
3. 管理员选择服务时的请求的提交
4. 管理员修改页面时的输入

2.7 类图

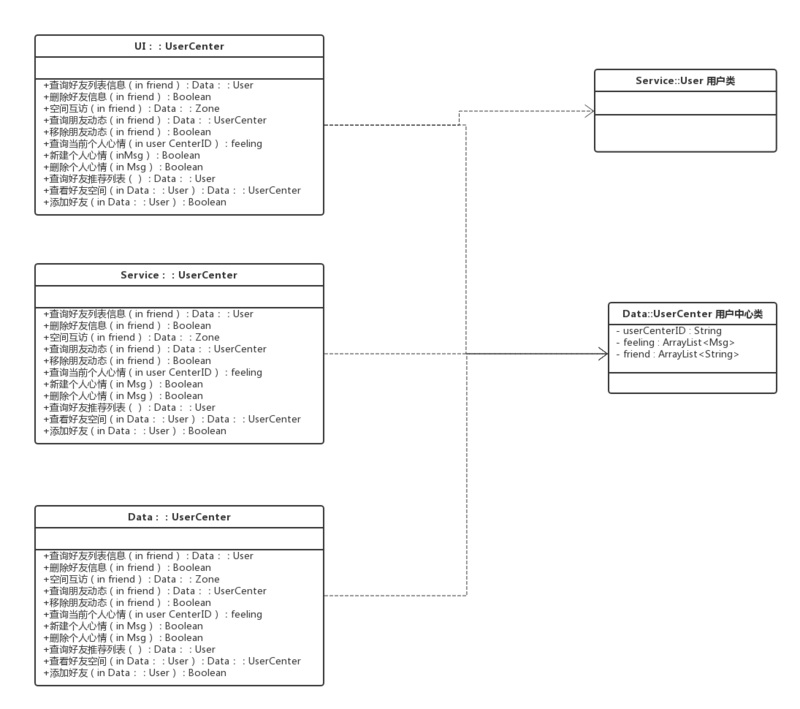
1. 日志模块



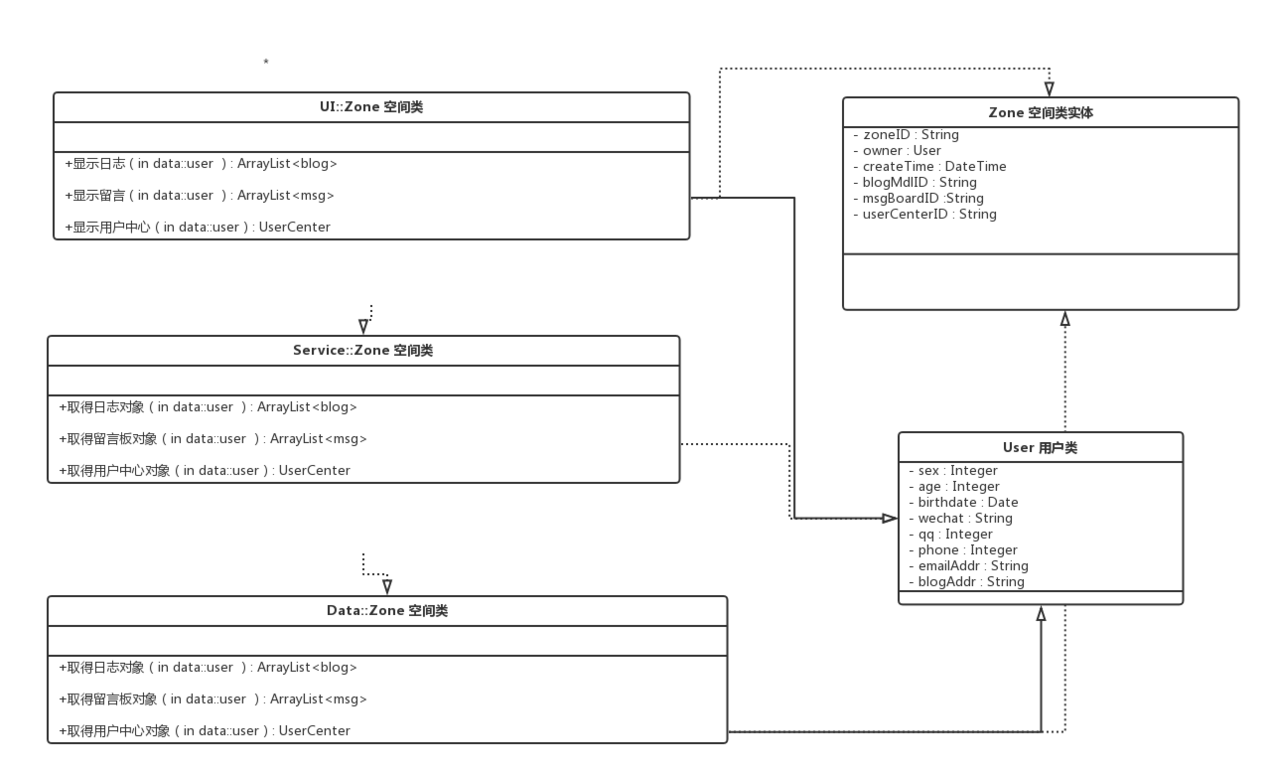
1. 留言板模块



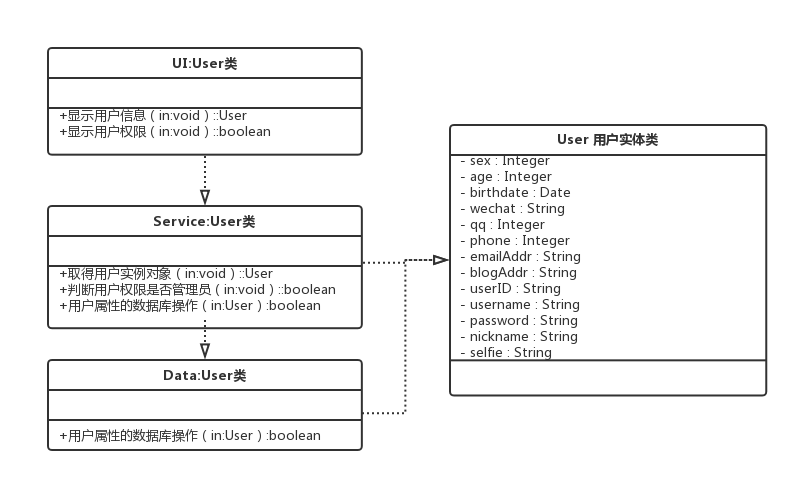
1. 用户个人中心模块



1. 4.空间模块



1. 5.用户模块



2.8 尚未问决的问题 （王思哲撰写）

在概要设计阶段，限于能力与经验，对系统所需要的类定义尚不全面，对数据库设计的合理性也尚不清楚，只能等具体开发阶段不断修正！

2018/7/7：

开发已进入尾声，修订了一些刚开始设计不周全的地方。

3 接口设计

3.1 用户接口（郑小妹撰写）

用户主要是通过操作浏览器来完成个人空间的管理。

登陆注册方面需要填写一些表单信息，点击一些功能按钮；

留言板、心情、动态编辑等需要在文本框控件进行文字输入；

其他入口通过对应界面的图标或文字链接即可进入，当鼠标图标变为小手形状时表示此处为链接入口。

3.2 外部接口（张海涛撰写）

* 1. 软件与硬件的接口：通过鼠标，键盘，图形显示界面等输入输出进行交互
  2. 系统与各个支持软件之间的接口：本系统通过jdbc-odbc桥接口操作MySQL 5.0数据库，通过Tomcat 8.0配置服务器，开发工具使用MyEclipse 2014（JDK8 JRE8）。
  3. 用户通过浏览器访问博客网站。

3.3 内部接口 (陈鹏旭撰写)

Webroot/page：

前端页面

Com.amzblog.servlet：

后台servlet层接口：接收前端表单，请求service处理

Com.amzblog.service：

后台service层接口：处理核心业务逻辑，接收servlet请求，请求数据库数据。

Com.amzblog.dao：

数据库操作层接口：被service调用，接收service层数据，进行相关数据库操作，返回给service层正确的数据，错误时会抛出错误。

Com.amzblog.db：

数据库配置层：含hibernate

4 运行设计（张海涛、陈鹏旭撰写)

4.1 运行模块组合

|  |  |  |
| --- | --- | --- |
| 外界控制 | 运行模块组合 | 支持软件 |
| 登录 | 用户管理/主页管理 | MySql/Tomcat |
| 注册 | 用户管理 | MySql/Tomcat |
| 新建个人主页信息 | 主页管理 | MySql/Tomcat |
| 修改个人主页信息 | 主页管理 | MySql/Tomcat |
| 删除个人主页信息 | 主页管理 | MySql/Tomcat |
| 查询个人日志信息 | 日志管理/主页管理 | MySql/Tomcat |
| 新建日志信息 | 日志管理/主页管理 | MySql/Tomcat |
| 修改个人日志信息 | 日志管理 | MySql/Tomcat |
| 删除个人日志信息 | 日志管理 | MySql/Tomcat |
| 查询留言板信息 | 留言板管理/主页管理 | MySql/Tomcat |
| 新建留言板信息 | 留言板管理/主页管理 | MySql/Tomcat |
| 修改留言板信息 | 留言板管理 | MySql/Tomcat |
| 删除留言板信息 | 留言板管理 | MySql/Tomcat |
| 查询个人信息 | 个人管理/主页管理 | MySql/Tomcat |
| 修改个人信息 | 个人管理 | MySql/Tomcat |
| 删除个人信息 | 个人管理 | MySql/Tomcat |
| 查询好友列表 | 个人中心/主页管理 | MySql/Tomcat |
| 删除个人好友信息 | 个人中心 | MySql/Tomcat |
| 空间互访 | 个人中心 | MySql/Tomcat |
| 查询朋友动态信息 | 个人中心/主页管理 | MySql/Tomcat |
| 删除朋友动态信息 | 个人中心 | MySql/Tomcat |
| 查询个人心情 | 个人中心/主页管理 | MySql/Tomcat |
| 新建个人心情 | 个人中心/主页管理 | MySql/Tomcat |
| 删除个人心情 | 个人中心 | MySql/Tomcat |
| 查询好友推荐列表 | 个人中心 | MySql/Tomcat |
| 添加好友 | 个人中心/主页管理 | MySql/Tomcat |
| 查看空间装扮 | 主页管理 | MySql/Tomcat |
| 更换当前空间装扮 | 主页管理 | MySql/Tomcat |

4.2 运行控制

登陆：



注册：



新建个人主页信息：



修改个人主页信息：



删除个人主页信息：



查询个人日志信息：



新建日志信息：



修改个人日志信息：



删除个人主页信息：



查询留言板信息



新建留言板信息：



修改留言板信息：



删除留言板信息：



新建相册：



修改相册信息：



删除相册：



查询个人信息：



修改个人信息：



删除个人信息：



查询好友列表：



删除个人好友信息：



空间互访：



查询朋友动态信息：



删除朋友动态信息：



查询个人心情：



新建个人心情：



删除个人心情：



查询好友推荐列表：



添加好友：



查询现有访问权限：



修改访问权限：



查看空间装扮：



更换当前空间装扮：



4.3 运行时间

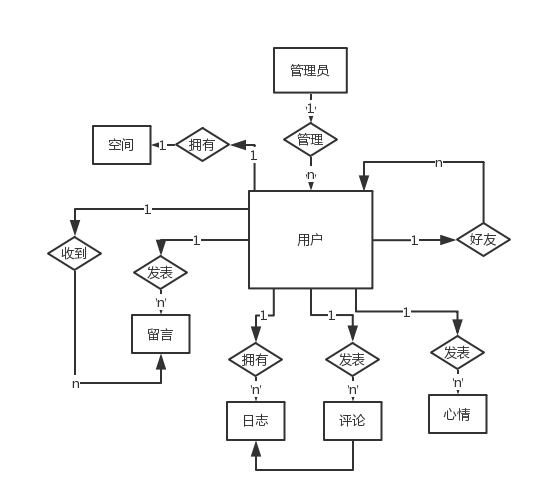
各模块运行时间应控制在1-2秒内（其中大部分为响应用户的动作）。

函数和实践的运行时间要看所在计算机的配置，系统的运行时间基本上可以达到用户提出的要求，系统响应时间视情况而定，一般正常情况下应该很快。

5 系统数据结构

5.1 逻辑结构设计要点

E-R图：



数据库概念设计图：



5.2物理结构设计要点

注：表和表字段命名规范：

1、表名的定义：

规范格式：系统名\_表名（字母全部大写）

比如，本系统中的用户表定义为：AMZBLOG\_USER;

2、表字段的定义：

规范格式：表名+字段名称（首字母大写），以避免字段重名出现在不同表中。

比如：表AMZBLOG\_USER中的字段标识ID定义为：UserId;

3、表字段长度的定义：

   规范格式：长度一般为8的整数倍；

   比如，VARCHAR的长度可为8或16或24或32或64等，最长为255。

4、表字段类型为日期型的定义：

   规范格式：一般日期型的字段设为date。

5、公共表名的定义：

   规范格式：PUBLIC\_表名（字母全部大写）

   比如，各个子系统都会用到的公共表，如角色或操作日志表可定义为：

   PUBLIC\_ROLE, PUBLIC\_OPERATELOG

存储要求：mysql 5.x InnoDB。

访问方法：只能通过com.amzblog.dao层访问数据库。

存取的物理关系：暂无描述。

设计考虑和保密条件：所有密码采用md5加密存储。

数据库物理结构设计图：



5.3数据与结构

数据结构：

个人空间表



用户表



管理员表



日志表



评论表



留言去向关系表



留言表



心情表



好友关系表



访问方法：通过数据Dao层的类调用数据库来访问数据。

6系统出错处理设计(前端，后台，数据库负责人都写)

6.1出错信息

用一览表的方式说朗每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。

|  |
| --- |
| 用户登录时，如果输入的用户名和密码不和数据库中的相匹配，则系统自动弹出提示窗口，输出错误信息 |
| 用户注册时，如果注册使用的用户名已经被使用过，则不能注册，系统自动弹出提示窗口，输出错误信息 |
| 管理员登录时，如果输入的用户名和密码不和数据库中的匹配，则不能进入管理界面 |
| 需要操作数据库时，如果操作失败，或者发生异常，则输出相应错误信息 |
| 如果用户在表单中输入的数据不符合预定的要求，则弹出错误信息 |
| 如果用户上传的图片不符合要求，则弹出错误信息 |
| 如果数据库损坏，不能正常操作，则弹出出错信息 |
| 如果网络异常不能正常连接，则弹出异常信息 |

6.2补救措施

说明故障出现后可能采取的变通措施，包括：

1. **后备技术：**

定期备份数据库文件，一旦数据库发生异常，立即用备份的数据库覆盖原来的数据库文件。

1. **恢复及再启动技术：**

定期对数据库和服务器进行更新和检查，并定期备份重要文件

1. **数据库连接错误：**

这类错误主要是数据库设置不正确引起的，回滚本次操作，生成操作日志，提醒维护人员自己检查数据库问题即可。

1. **输入错误：**

主要是用户输入不规范造成的，我们在尽量减少用户出错的条件下对用户进行提醒，然后再次操作。

1. **乱码：**

主要发生在提交汉字信息的情况下，这时我们只需要对提交的汉字重新进行UTF-8编码

1. **其他不可预知的错误：**

程序也会有一些我们无法预知或没有考虑，我们对此不可能走出完全的异常处理，为了保证数据的安全，要经常对数据库进行备份，然后返回错误信息，以逐步完善程序。

1. **安全技术：**

为保护注册用户的合法权益，对用户的一些信息进行安全处理，对一些信息进行加密处理，使用session以及application禁止各类用户的非法访问未授权网页。

6.3系统维护设计

|  |  |  |
| --- | --- | --- |
| 标识符 | 模块名 | 模块检查点 |
| 01 | 前端（View） | Webroot/page |
| 02 | Servlet（View） | com.amzblog.servlet |
| 03 | Service层  （Control） | com.amzblog.service |
| 04 | 数据层（Model） | Mysql设置，com.amzblog.model, com.amzblog.dao, com.azmblog.daoImpl, hibernate.cfg.xm，每个类的hbm.xml |