**软件工程**

**系统设计说明书**

**团队名称： 软件工程四人小团队**

**班 级： 19级卓越班**

**指导教师： 代祖华**

**成 员： 刘温元 闫雨馨 刘转弟 李瑞婷**

1．引言

1.1编写目的

从该阶段考试正式进入软件的实际开发阶段，本阶段完成系统的大致设计并明确系统的数据库与软件结构。在系统设计阶段主要是把一个软件需求转化为软件表示的过程，这种表示只是描述软件的总的概貌。本概要设计说明书的目的就是进一步细化软件设计阶段得到的软件总体概貌，把它加工成在程序细节上非常接近于源程序的软件表示。

在完成了“背包知识社区论坛”的可行性研究和需求分析的基础上，为了明确软件需求、安排项目规划与进度，组织软件开发和测试，项目小组在考虑了几种可能的解决方案，并对每种方案进行深入探讨后，提出了这份概要设计说明书。

此概要设计说明书对管理系统软件的软件架构、模块细化、程序的总体结构、接口设计以及数据库设计规划等方面做了全面的概括性说明，为“背包知识社区论坛”系统的详细设计奠定了基础。

1.2预期读者和阅读建议

该文档的预期读者为“背包知识社区论坛”程序开发员、系统设计师以及测试工程师等。本文档主要针对“背包知识社区论坛”设计概述、软件架构、子系统的结构和模块功能以及系统接口、数据库设计规划等方面进行了详细的说明，笔者建议，在阅读本文档之前，细细阅读“术语与缩略语”部分，这将有利于您对本文档的理解。对于程序开发员，笔者建议仔细阅读系统的软件架构及子系统的结构和功能模块部分，了解每个功能模块之后，这样对系统的功能及运行模式将有整体的把握，然后阅读后面几部分内容，有利于“背包知识社区论坛”系统的实现。而测试工程师应把重点放在系统接口、数据库设计规划以及数据库设计上，在系统测试过程中，最容易出错的就是这两方面。总之，笔者建议读者根据自己阅读本文档的目的选择着重点，这样在满足需求的同时提高效率。

1.3参考资料

[1] 孙卫琴,李洪成编著.Tomcat与Java Web开发技术详解[M].电子工业出版社，2004

[2] 飞思科技产品研发中心编著.JSP应用开发详解[M].北京：电子工业出版社，2004

[3] 张海藩编著.软件工程导论[M].北京：清华大学出版社,2003

[4] 萨师煊，王珊著.数据库系统概论[M].北京：高等教育出版社,2003

1.4术语与缩略语

无

1. 设计概述

2.1 业务流程

业务流程描述：用业务流程图描述现系统各项业务处理过程，并结合业务流程图详细准确地说明业务处理过程中数据的收集、输入、传递、存储、加工的方法、算法、输出的数量及形式等。

现行系统的业务流程图是分析和描述现行系统业务的重要工具之一。其主要作用是：描述现行系统的业务情况以便于与管理人员交流，它是系统设计的前提和保证。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 符号 |  |  |  |  |  |  |
| 名称 | 实体 | 输入/输出的报表 | 处 理 | 业务流向 | 文档 | 存储 |

表2-1 业务流程图使用的符号：

2.1.1 管理员管理系统的业务流程图

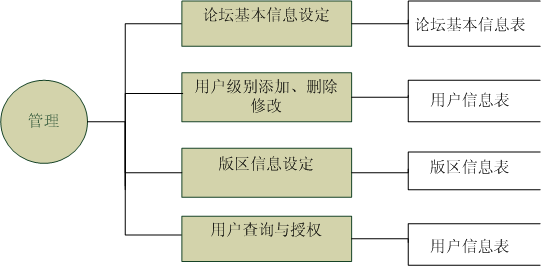


图2-1 管理员管理本系统业务流程图

系统管理员在后台界面可以对论坛的一些基本信息进行设定，可以对用户进行删除和修改，可以对讨论区的版面区域进行设置，可以查询系统中的用户。操作的相应结果都会存储在对应的数据库表中。

2.1.2 用户使用系统的用例图

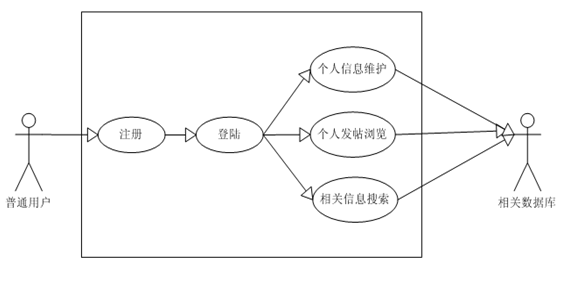
用户使用本网站的信息，包括用户注册，登录，发贴，修改个人信息。用户系统的业务用例图如下：

图2-2 普通用户使用系统的用例图

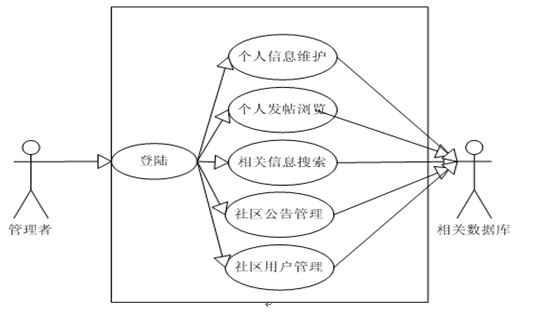
一般的用户在注册用户名之后可以登陆到本网站，可以执行的功能有个人信息的维护，发帖子，浏览帖子，还可以根据帖子的主题或者作者名字来进行相应的搜索。相应的操作结果都将存储于对应的数据库表中。

图2-3 管理员使用系统的用例图

与一般的用户不同，具有管理权限的用户在登陆前台页面后除了可以执行一般用户的功能例如个人信息维护，发帖浏览帖子之外，还可以对社区的一些公用信息，例如社区公告、社区最新活动消息进行添加和删除，还可以对一些用户进行管理。相应的操作结果都将存储于数据库中对应的表里。

1. 系统总体结构图

3.1系统运行总体流程



图3-1 后台管理系统运行流程图

后台管理员打开后台管理界面后，输入用户名：admin，输入密码:admin，如果输入错误则需要返回登录页面继续输入。如果输入正确，则可以进入后台管理的操作界面。这时，管理员可以执行后台管理功能。可执行的功能包括：论坛基本信息的设定、论坛社区信息的管理、用户的查询以及用户的删除。管理人员在完成后台的维护工作之后，只需要点击退出，管理员便可以退出后台管理系统。

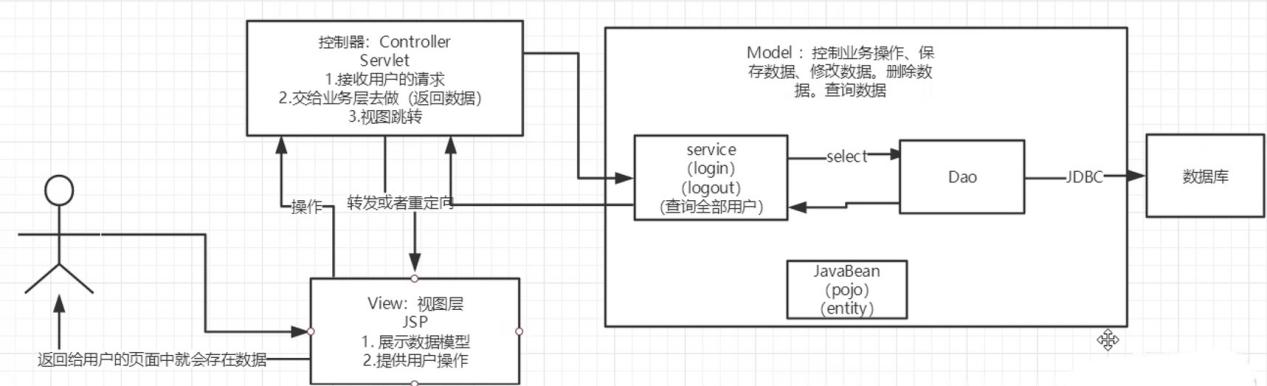


图3-2 用户前台系统运行流程图

首先，用户需要注册一个账号和密码，然后在登录页面输入已注册的账号密码。如果输入错误，则要重新回到登录页面输入正确的账号密码。如果输入正确，则可以进入用户操作页面进行相应的操作。然后点击退出，用户退出本网站。需要注意的是，用户在注册账号时需要受到一定的限制，一些特殊的字符例如：fuck，不可以用于注册用户账号。这个信息是后台管理人员在后台维护界面设定的。同样也可以在用户注册页面给用户以人性化的小提示。

3.2.采用适合的软件设计模式设计软件系统总体结构：

采用MVC设计模式，设计软件系统总体结构，如下图及说明：



说明：（1）在背包问题知识社区系统采用MVC设计模式，结合网络数据库开发技术来设计本系统。开发语言采用JSP，数据库使用SQL Server数据库,完成以下基本功能：用户管理、背包问题相关资源上传、删除、修改和获取、论坛发布等功能。

（2）servlet专注于处理请求以及控制视图跳转，JSP专注于显示数据。

3.3系统体系结构

3.3.1 系统软件结构图

管理员后台管理功能结构图如下所示：



图3-3 管理员后台管理功能结构图

本图详细的解释了后台管理界面所包含的内容：管理人员的登录界面、注册信息管理、论坛基本信息的管理、用户信息的管理以及密码更改。

管理员前台管理功能结构图如下图说示范：



图3-4 管理员前台管理功能结构图

本图详细的解释了前台用户界面所包含的内容：用户注册、用户登录、个人信息管理、论坛讨论区、社区管理（针对具有管理权限的用户）。

3.3.2 系统软件模型设计

把论坛分为五个模块

1.社区模块：用户在这一模块可以自己浏览自己喜欢的文章，然后收藏。

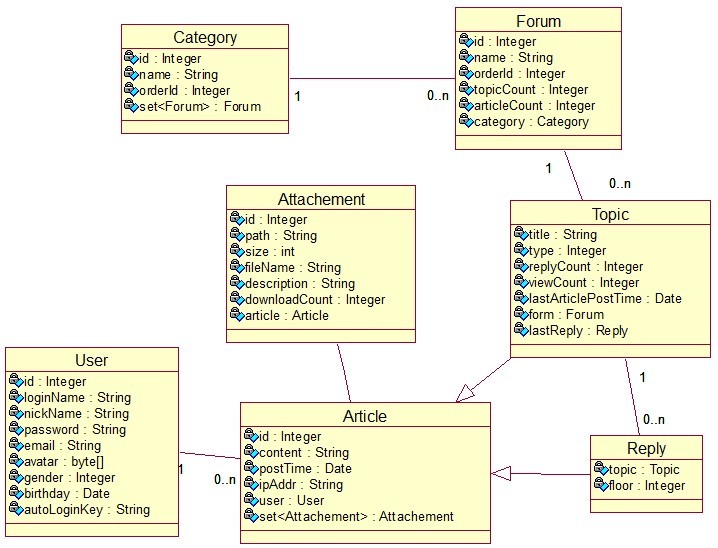
2.用户中心模块：用户在这一模块可以修改自己的信息。

3.信息模块：用户在这一模块可以看到自己在别的用户帖子下评论的内容，用户之间的信息交流也可以通过这个模块显示。

4.资源模块：用户可以在这个模块看到自己收藏的资源，还可以在这里上传资源到社区，供别的用户使用。

5.管理员模块：管理员可以在在个模块，增加、删除、修改、查询其他用户的权限，还要对用户上传的资源内容加以审核。

论坛有分类（Category），一个分类下有多个版面(Forum)，一个版面可以发表多个主题(Topic)，一个主题对应多个回复(Reply)。由于主题和回复都有内容、发表时间、对应的用户等，所以把他们的共有的属性抽像成一个文章类(Article)。一篇文章对应着多个附件（Attachement）,还有一篇文章对应一个用户（User）。



1. 数据库、系统接口设计规划

4.1数据库逻辑设计

数据库的逻辑设计即是表结构和表关系的设计，本系统的E-R图如下：

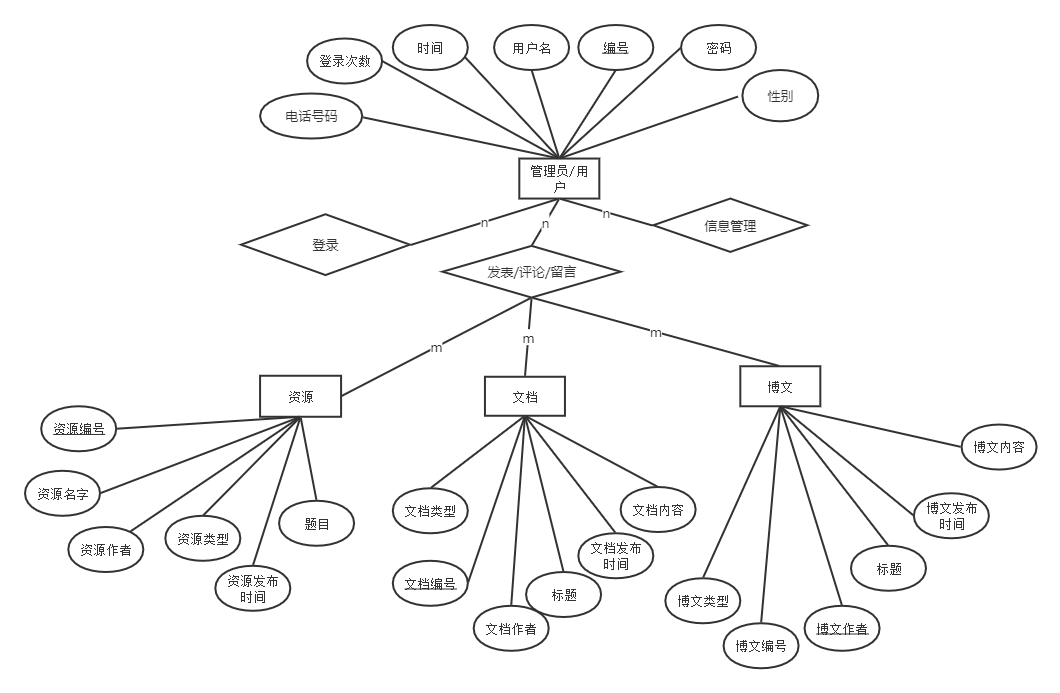
****

图4-1 数据库E-R图

**4.2数据字典**

数据字典是数据信息的集合。是数据流图中所有要素严格定义的场所，这些要素包括数据流、数据流的组成、 文件、加工小说明及其他应进入字典的一切数据，其中每个要素对应数据字典中的一项条目。

表4-1用户及管理员表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否约束 | 备注 |
| 性别 | VARCHAR |  | 是 | 男，女 |
| 密码 | INT | 0 | 否 |  |
| 编号 | INT |  | 是 |  |
| 用户名 | VARCHAR |  | 否 |  |
| 时间 | VARCHAR |  | 否 |  |
| 登陆次数 | INT |  | 是 |  |
| 电话号码 | INT |  | 否 | 绑定 |

4-2 资源表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否约束 | 备注 |
| 资源编号 | INT |  | 是 |  |
| 资源作者 | VARCHAR |  | 否 |  |
| 资源类型 | VARCHAR |  | 否 | 文档，代码，  博文 |
| 资源题目 | VARCHAR |  | 否 |  |
| 资源发布时间 | VARCHAR |  | 否 |  |

表4-3博文表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否约束 | 备注 |
| 博文编号 | INT |  | 是 |  |
| 博文作者 | VARCHAR |  | 否 |  |
| 博文标题 | VARCHAR |  | 否 | 背包问题 |
| 博文类型 | VARCHAR |  | 否 |  |
| 博文发布时间 | VARCHAR |  | 否 |  |

表4-4 帖子信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段大小 | 主键 | 备注 |
| Id | Numeric | 9 | Y | 序号 |
| Title | Varchar | 200 | N | 标题 |
| Content | Varchar | 3000 | N | 内容 |
| Addusername | Varchar | 20 | N | 发表人名字 |
| Starttime | Numeric | 9 | N | 开始时间 |
| endtime | Numeric | 9 | N | 结束时间 |

表4-5帖子回复表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段大小 | 主键 | 备注 |
| Replyid | Numeric | 9 | Y | 回复人序号 |
| Replywriter | Varchar | 20 | N | 回复人名字 |
| Replytime | Numeric | 9 | N | 回复时间 |
| Replycontent | Varchar | 3000 | N | 回复内容 |

4.3系统接口设计

外部接口：

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。在界面设计上，应做到简单明了，易于操作，并且要注意到界面的布局，应突出重要以及出错信息。外观上也要做到合理化，考虑到用户多对WINDOS风格较熟悉，应尽量向这一方向靠拢。总的来说，系统的用户界面应做到可靠性、简单性、易学性和实用性。

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.在界面设计上，应做到简单明了，易于操作，并且要注意到界面的布局，应突出重要以及出错信息.外观上也要做到合理化，考虑到用户多对windos风格较熟悉，应尽量向这一方向靠拢.总的来说，系统的用户界面应做到可靠性、简单性、易学性和实用性。

5.软件重用方案

5.1什么是软件重用

软件重用，是指在两次或多次不同的软件开发过程中重复使用相同或相似软件元素的过程。软件元素包括程序代码、[测试用例](https://baike.baidu.com/item/%E6%B5%8B%E8%AF%95%E7%94%A8%E4%BE%8B/1928697" \t "_blank)、设计文档、设计过程、需求分析文档甚至领域知识。通常，可重用的元素也称作[软构件](https://baike.baidu.com/item/%E8%BD%AF%E6%9E%84%E4%BB%B6" \t "_blank)，可重用的软构件越大，重用的粒度越大。

5.2软件重用层次

（1）知识重用（例如，[软件工程](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E5%B7%A5%E7%A8%8B" \t "_blank)知识的重用）。

（2）方法和标准的重用（例如，面向对象方法或国家制定的软件开发规范的重用）。

（3）软件成分的重用。

5.3重用方案

（1）思想的重用：说明，设计，开发方法学和技术等。

（2）领域知识的重用：文档、技术手册、计划参加人员、分析数据、测试和维护信息数据库等。

（3）个别部件的重用：源代码，子例序，模块，操作系统，软件包，编程语言和工具等。

5.4软件重用的好处

使用软件重用技术可以减少软件开发活动中大量的重复性工作，这样就能提高软件生产率，降低开发成本，缩短开发周期。同时，由于[软构件](https://baike.baidu.com/item/%E8%BD%AF%E6%9E%84%E4%BB%B6" \t "_blank)大都经过严格的质量认证，并在实际运行环境中得到校验，因此，重用软构件有助于改善[软件质量](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E8%B4%A8%E9%87%8F" \t "_blank)。此外，大量使用软构件，软件的灵活性和标准化程度也可望得到提高。

6.关键类的重点服务

1：管理类  
分别为用户和管理员，用户主要是管理个人信息和论文，资源等的发表，管理员主要是用来约束用户行为，管理用户文档，资源等的管理。

2：控制类  
主要是对资源，文档进行访问，修改，查询等操作。  
3：数据类  
该类主要进行文档、资源的上传或删除等，对资源，文档，博文等进行具体的细节操作。

7.详细设计

7.1命名规则

1名字不能用保留字和关键字，但如果保留字或关键字中有字母大写就不算保留字或关键字了。

2 开头字符能用字母、$或者\_。

3 名字中不能用+、- 空格等字符。

4 部分易混的保留字关键字和非保留字关键字：

 null, native, goto，const，instanceof，default是关键字保留字

 then , sizeof , main，Boolean，unsigned,java, Integer不是关键字保留字

7.1.1 JSP基本数据类型

int长度数据类型有：byte(8bits)、short(16bits)、int(32bits)、long(64bits) float长度数据类型有：单精度（32bits float）、双精度（64bits double）

boolean类型变量的取值有：ture、false

char数据类型有：unicode字符,16位

对应的类类型：Integer、Float、Boolean、Character、Double、Short、Byte、Long

7.1.2 常量命名

常量的名字应该都使用大写字母，并且指出该常量完整含义。如果一个常量名称由多个单词组成，则应该用下划线来分割这些单词。

例如： MAX\_VALUE

7.1.3 其它部分命名规则

1.包的命名

JSP包的名字都是由小写单词组成。但是由于Java面向对象编程的特性，每一名Java程序员都可以编写属于自己的Java包，为了保障每个Java包命名的唯一性，在最新的Java编程规范中，要求程序员在自己定义的包的名称之前加上唯一的前缀。由于互联网上的域名称是不会重复的，所以程序员一般采用自己在互联网上的域名称作为自己程序包的唯一前缀。

2.类的命名

类的名字必须由大写字母开头而单词中的其他字母均为小写；如果类名称由多个单词组成，则每个单词的首字母均应为大写例如TestPage；如果类名称中包含单词缩写，则这个缩写词的每个字母均应大写，如：XMLExample,还有一点命名技巧就是由于类是设计用来代表对象的，所以在命名类时应尽量选择名词。

3.方法的命名

方法的名字的第一个单词应以小写字母作为开头，后面的单词则用大写字母开头。

例如： sendMessge