**摘 要**

随着社会进步,校园论坛也越受人们的喜欢。校园论坛是经过社区论坛，提供给人们一个虚构的空间减少压力。现代生活中，不论成年人或者未成年人都面临强大的压力，在网络这个毫无压力的空间，人们能够更好地放开自己的世界，以便更好地投入现实中去。给我们创造一个崭新的表现自我的舞台。校园论坛为这些有共同爱好的年轻人打造了另一片交流的天地，深受大学生和老师推崇。

校园之家系统分为后台和前台，后台主要包括管理员管理、用户管理、帖子管理、资源信息管理四个模块，前台主要包括用户的登陆、用户的注册、发布资源链接、主题管理几个功能。

本文介绍了开发中的开发模式如何以及在本系统中采用的技术方面。本次设计应用的是Windows下的Apache、Java、MySQL开源[软件](http://baike.baidu.com/view/37.htm)，并采用MVC三层架构思想构建网站整体，并使用SpringMVC、Spring、MyBatis框架，Apache服务器和MySQL数据库进行开发。

**关键词**：校园之家系统，管理，框架，服务

**ABSTRACT**

Along with the social development, the campus forum is getting more and more popular. The campus forum is a community forum that provides people with a fictional space to reduce the pressure. In modern life, whether adults or minors are facing strong pressure, in this network without the pressure of space, people can better open a world of his own, to better put into reality. Create a new stage for us to show ourselves. , the campus forum for these common love of young people to create a different world of exchange, by college students and teachers respected.

A system of the campus is divided into background and foreground background, including the administrator management, user management, post management, information resource management module four, the mainly includes user login, registration, release, delete posts, posting links several functions. The detailed analysis of the demand, considering a variety of business operation, the development of the late strictly follow the software development process, unified plann, stop-by-stop implementation.

This discusses the system development mode and technology used the system. The application of this design is under the Windows Apache+MySQL+Java open source software, and build a website using MVC three layer architecture, and use the SpringMVC+Spring+MyBatis framework, Apache server and MySQL database development.

**KEY WORDS**: the campus system, Administration，Frame,Servic

**第1章 绪 论**

**1.1 课题的提出**

在21世纪,地球因为互联网的存在,已经成为一个村庄。人们导航在互联网的世界里,享受着互联网带来无限乐趣。我们成了习惯阅读在线新闻,BBS KanTie,在线网购,写自己的平时的生活的内容,等等。还有各种各样的搜索引擎,帮助我们找到你想要查找的信息和资源。总之,互联网已经成为一种生活的方式,一个强大的知识地。这种设计,围绕着如何构建在线论坛系统的研究。在线论坛系统(BBS)是一个各种各样的信息交流和传播,它有别于传统媒体如报纸、电视,任何人都可以随便BBS上的他的言论,不受任何限制。除了可以表达短信可以上传和发布图片,音乐,Flash,可以选择字体大小,颜色,也可以选择有趣的表达模式。只要花一个注册论坛(甚至不需要)你可以游泳,可以得到无穷无尽的快乐。这个系统是基于系统特点,目的是为用户提供平常交流的平台,在这里发表自己的看法,观点,为生活添加乐趣交朋友。

随着社会推进,校园论坛逐步受人们的喜欢。校园论坛是经过社区论坛，提供给人们一个虚构的空间减少压力。现代生活中，不论成年人或者未成年人都面临强大的压力，在网络这个毫无压力的空间，人们能够更好地放开自己的世界，以便更好地投入现实中去。给我们创造一个崭新的展示自我的舞台。在这里人们可以碰触到许多平时难以碰触的资源。当代大学生习惯结交新朋友，寻找有共同兴趣的人交流和讨论自己的想法，校园论坛为这些有共同爱好的年轻人打造了另一片交流的天地，深受大学生和老师推崇。

**1.2 国内现状研究**

网络的普及成为趋势,论坛的内容越来越多,BBS,深受广大网民的喜爱,因此,商业网站BBS迅速上升,和互联网提供一个交流的平台,还有BBS在线技术支持。

目前国内的论坛存在很多个,也可以说是无数了,BBS可以可以区分为五个类型:校园BBS成立以来,校园BBS发展迅速。大部分的论坛建立了网络中心,也有私人的论坛;商业的BBS站,这里主要是商业的推广、产品介绍、等等,手机方面、电脑业务业务方面,房地产业务站方面无处不在;专业的BBS站是公司和别的大型企业所建立的自己的论坛系统主要用于建立区域的文件的传输和信息发布系统,论坛主要用于通信,是许多娱乐的首选网站,个人的BBS论坛是一些生产商的资料,你的个人主页建设用于采纳别人的想法,更有利于和朋友之间的交流。

**1.3 系统特点**

校园之家系统就是基于校园论坛的的系统，为校园的学子提供学习资料，并且可以提供互相之间相互交流的平台，可以互相交流的彼此之间的学习经验，建立彼此之间的交流群体，互相进步。同时可以展现自己的平时被隐藏的才能，找到志同道合的人一起交流。同样，也可以帮助学校进一步了解学生的平时的爱好和学习的氛围，也为下一步开展活动的指导方向。本系统就是旨在为学子提供家的感觉，可以畅所欲言，可以得到需要的信息。

本文共分七章，各章的主要内容如下：

第一章“绪论”部分主要总结了系统的开发的背景、国内研究现状、系统特点和组织结构。

第二章“理论及技术基础”部分对论文研究中所用到的理论、技术简单的介绍。

第三章“系统需求分析”部分主要描述了系统需求分析过程,包括系统的描述,系统功能的基本情况的分析,可行性的分析,需求的硬件和软件环境。

第四章“总体设计”部分对系统的总体设计过程做了描述,主要包括系统的总体功能模块的分析,功能模块的设计。

第五章“详细设计”是系 统的详细设计，包括数据库设计和界面设计。

第六章“编码实现”部分主要对该系统的核心代码的解释,并列举了相应的核心代码。

第七章“系统测试”部分主要对该系统测试,主要包括测试目标、测试标准和根据测试目标和测试黑盒测试和白盒测试的标准。

**第2章 理论及技术基础**

**2.1 Spring+SpringMVC+MyBatis框架**

Spring是一个开源框架,Spring崛起于2003年,是属于Java开发框架的轻量级的,是一个J2EE开发和设计概念和原型。它的创建是为了解决复杂的应用开发。Spring的应用是基于JavaBean的原则，但在这之前只能通过EJB。Spring不仅仅是用于服务器的方面的建设应用和运用应用的。从简单的方面、可测试的方面和松耦合的方面,所有的Java的应用程序都能在Spring中获得受益。

每一个模块的Spring的框架都可以能够单独的运行,或与其他的模块共同去实现功能。每个模块的功能如下:

1. 核心容器:核心的容器可以为Spring的框架开发了基本的功能。
2. Spring的上下文。
3. SpringAOP:经过配置管理的功能,面向方面的功能,关系到Spring框架受SpringAOP模块集成。
4. SpringDAO:对于抽象层可以是一个有价值的层次的结构,并且可以使用这个结构来处理异常和不一样的数据库的厂商的信息。这个结构可以能够很好的简化对于错误信息的处理,减少需要对编纂异常的代码的行数。用于JDBC异常遵守一般的异常层次结构。
5. SpringORM: 对于加多个ORM的框架是可以通过Spring的框架来完成的,它可以提供很好的ORM的工具。
6. Spring Web模块: 对程序模块中的应用是基于模块的Web环境下，Web应用程序可以给出一个上下文。Web的模块还精简一部分相应的处理的请求,使请求的参数能够定到对应的域对象。

Spring MVC框架:是为了于MVC实现来建造一个功能繁多的Web的应用程序。经过策略接口, 可配置的MVC框架是可以的,MVC可以用大量的视图的技术,例如iText、JSP和POI。Javabean可以将模型存储在映射, 接口由视图实现,负责显示模型;逻辑控制器说,控制器的实现。在J2EE的服务器Spring的框架的功能是可以去使用,并且在不受管理的环境下大部分的功能也可以能适用。Spring的关键点是: 提供一个独特的J2EE服务的服务对象可以用来访问业务和数据绑定到支持。

MyBatis:首先apache的开源之一是iBatis，它是以java的持久层的框架为基础。MyBatis去除大多的JDBC代码和参数手动设置和检索结果集是因为IBATIS持久层框架包括SQL映射和数据访问对象(DAO)。注释配置或单一的XML和最初的映射受MyBatis的运行, 数据库中的记录受该接口和Java pojo的映射。

其实，创造SqlSessionFactory单个实例作用于用XML的文件来完成是一个普通的事情。推荐在这个配置类路径中使用的资源(资源类路径),但是您可以使用任何读者实例,包括使用文件路径或文件。MyBatis很多资源和方法,它可以很易地加载资源的相应的路径和其他位置。

**2.2 MVC架构**

构建用户界面是MVC最初是目的。M代表模型模型,V代表视图的视图,C代表控制器控制器。MVC的目标是在于减低数据的表达、对数据的描述，提高质量、降低应用程序的操作的耦合度。同时也让得软件具备可维护的,可修复的,可伸缩的、灵活的、和封装增加。  
 从各种各样的数据通过数据模型可以访问和控制分离可以提高分布式系统的设计。MVC设计模式分为三个部分。该模型属于应用对象，没有用户界面。它在屏幕上显示,代表用户的数据流。控制器用于定义用户界面对于用户的输入条件的响应,负责操作的用户的行动为根据的模型。模型通过更新数据,以反映变化的观点。

**2.3 MySQL数据库**

MySQL能够用于最近大部分的系统，如流行的关系数据库相关的操作系统,它存在一个分布式的管理系统。MySQL完全可以用于网络方面,创建的数据库可以在有网络的地方访问互联网,因此,可以在任何有网络的地方与任何有网络的人共享网络上的数据库信息。MySQL和强大的功能,使用简单,易于管理,操作快,可靠性高,安全、隐私、MySQL等。MySQL是由C和c++组成的, 许多平台都可以能够来支持它的工作,为不同编程的语言提供了相应的API的函数,而且使用的核心线程能够实现多线程,可以很好的支持多个CPU;并且提供的非事务性的存储机制和事务性的存储机制; 也提供了基于线程的内存方面的分配系统;MySQL的使用具有双重许可,用户可以自由软件在GNU许可条款或使用MySQL的软件开源软件的形式,也能从MySQL AB公司拿到营业许可。  
 除了上述特点之外,MySQL和最大的一个特点,如UNIX操作系统上,它是免费的,可以从互联网上下载服务器和客户端软件。也可以从互联网上得到很多匹配的第三方软件或工具。在Windows的系统中,客户端的程序和客户端的程序库是完全免费的。它的主要的优势包括:

1.支持的查询语言。MySQL也可以能够去支持SQL,而且SQL是技术的语言,是所有数据库系统必备选择。还可以使用支持ODBC应用程序(开放数据库连接),ODBC是微软开发的一个数据库通信协议。

2.优越的性能。没有限制的MySQL数据库的用户,多个客户端可以运用相同的数据库资源在同一瞬间。可以使用几个输入查询和视图界面访问MySQL互动的结果。接口是:命令行客户端程序,Web浏览器或X Window系统客户端程序。此外,还有由语言的接口,编写。所以,你可以选择使用客户端程序,程序或编写自己的客户机应用程序。

3.连通性好和安全。MySQL是完全网络化和全球化,可以在任意的地方在互联网上访问数据库的相应信息,因此,所有人都可以在随意的一个地方可以共享数据库信息。并且MySQL也可以访问控制,可以控制谁看不到你的数据。

4.的可移植性。MySQL也能够在别的版本中运行的UNIX的系统和其他UNIX的系统,也可以从自己家中的电脑运行服务器。

5.支持开放的分布。对MySql来说，他本是是容易发现获取的,如使用Web浏览器就可以获得。如果你不能理解的东西是如何工作的,或者好奇某种算法,它可以是源,分析源代码。如果你不喜欢某处,你可以替换它。

**2.4 Velocity**

Velocity，它是以java为基础。它可以让任何人只使简单易懂的模板的语言来取代了对象得定义的Java的代码。

Web在用到Velocity,界面的开发者和Java开发者应该都坚持MVC的原则,即页面开发者可以用来关注页面的显示的效果,并通过Java应用程序开发人员专注于业务逻辑的代码。速度将从一个web页面单独的Java代码,比如网站的长期维护方便,同时也为我们在JSP、PHP和提供了一个可选择的解决方案。

Velocity远远超出网站在这个领域发展的能力,例如,它可以从模板生成SQL和PostScript(模板),XML,它也可以被视为一个独立的工具来生成源代码和报告,或用作其他系统集成组件。

**2.5 JavaScript技术**

JavaScript是属于网络的脚本的语言,在Web应用的程序开发过程中,可以为页面提供动态效果,为用户提供更好的景色。通常加入在HTML JavaScript来实现自己的功能。是一种解释脚本语言(预编译的代码不会)。HTML页面可以增加互动的行为。可以直接嵌入到HTML页面,但是作为一个单独的js文件有利于分离的结构和行为。跨平台的特性,可以能让多数的浏览器的支持,也可以在不同的程序的平台中运行。Javascript提供了四个基本数据类型,和两个特殊数据类型用来处理数据和文本。变量提供一个存储信息,表达可以完成更复杂的信息处理。

**第3章 系统需求分析**

**3.1 可行性分析**

在当前系统可以提出一个新系统的基础上初步调查目标,即在新系统的建设需要,这是系统开发和评价的基础。新系统的目标应充分体现了铁路的战略目标,发展方向和直接为社会服务的基本特征,与此同时,新系统目标和当前应密切相关,系统的基本功能,并且可以实现分期付款。应该指出,新系统目标不可能在总体规划阶段是非常具体的,它也会清晰的一步一步的过程中发展和量化。

确定新系统的目的,可以从以下三个方面可以实现新系统的目标可行性分析:

(1)技术可行性。根据新系统目标来衡量所需的技术是否具备,一般从硬件、软件的性能要求、环境条件、技术人员水平和数量等方面去考虑和分析,开发人员应该首先考虑的能力和水平,技术力量和可以考虑技术人员的培养和发展。

(2)经济可行性。估算新系统的开发成本和未来的运营和维护成本,估计将得到新系统的好处,和比较成本和效益,看看是否好。

系统的好处可以从两个方面考虑的经济和社会效益。对车站票务管理系统应该分析其社会效益。系统投入运行后,例如,它可以提供不能提供及时的信息,方便用户查询和使用这些信息来提高多少,多少速度增加,为管理者提供太多的帮助做出决定,等等。

(3)操作可行性。在手术后新系统当前系统的影响,包括组织、管理方式、工作环境等)和后果估计和评价。现有的管理培训也应该被认为是与此同时,补充,分析在给定时间可以完成系统开发任务。

根据上述三个方面可行性分析之后,我认为这个项目是可行的。

**3.2 系统基本情况描述**

建立校园之家系统，校园之家系统中有管理员、会员及用户的各种功能如帖子管理、用户群管理、资源信息管理、修改个人资料功能。

**3.2.1系统功能分布图**

整个系统实现的功能如3.1图所示：

管

理

员

用

户

校园之家

系统

管理用户

管理版块

管理主题

管理用户

登录

发送信息

发表回复

发表主题

注册

图3.1 系统功能分布图

注销登录

管理用户权限

管理版块权限

增加好友

删除好友

增加资源

删除资源

**3.3 系统功能分析**

**3.3.1 登录**

用户登录在每一个系统中都占有着很大的比重，登录设计的好坏直接决定了系统是否能够健全的使用的重要标志。针对此问题我的方法是，前后台的登陆是分开的，用户和管理员面向不同页面；在登录的时候，只要选择要登录的系统，然后将正确的用户名，密码即可。然后系统会跟输入的账号和密码对其进行校验，其中比较关键的部分也是登录校验部分，这个部分影响到系统的安全的方面，所以在系统中登录时很重要的。登录的具体设计方式如图所示：

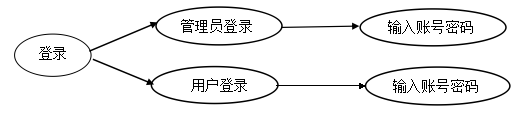


图3.2 登录设计图

**3.3.2 管理员管理**

管理员管理模块主要包括管理员的添加，修改，查看，删除等功能。

管理员可实现的操作有：

（1）管理员的添加：增加管理员的信息，按照页面给出的表单进行填写，填写完毕后点击确定即可。

（2）管理员的修改：修改管理员的相应信息，在管理员列表中直接点击修改链接，按照页面给出的表单进行填写信息，填写完毕后点击确定即可。

**3.3.3 用户管理**

对于管理员和用户，拥有对用户或者用户本身的添加，修改，查看，删除等操作。

（1）对用户修改：修改用户的信息，在用户列表中直接点击修改，按照页面给出的表单进行填充，填写完毕后点击确定便可。

（2）对用户删除：删除用户的信息，在用户列表中直接点击相应的用户删除。

（3）查看：查看用户的信息，在用户列表中可以直接点击相应的用户查看。

用户的添加：增加用户信息，按照页面给出的字段进行填写，填写完毕后点击确定。

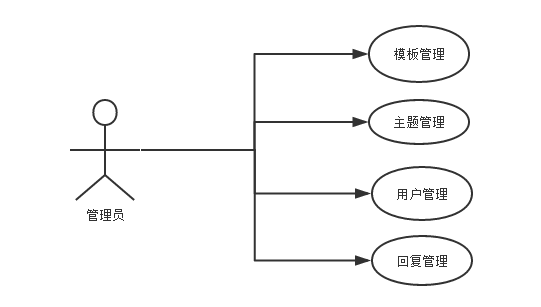
**3.3.4 主题管理**

用户可以在对应的模板下发布自己想要的主题，只要进入添加主题的页面中，便可以去添加内容。当然还有别的限制,如文字内容不能超出400字，字体类型、字号大小、字体的颜色都可以自己去设置,从而丰富了文章的内容。

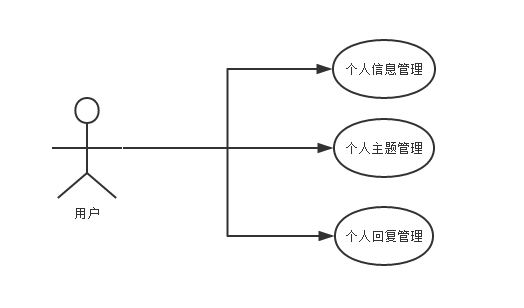
**3.4 系统用例图**

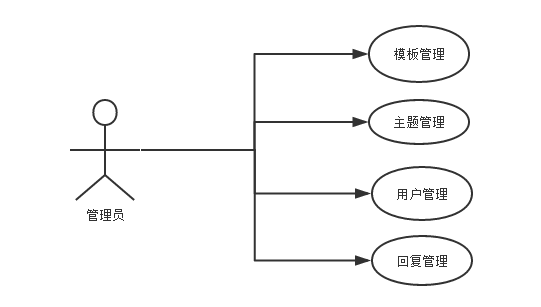
**3.4.1 管理员用例**

管理员的管理的功能比较多，例如，模块管理，主题管理，回复管理，用户管理等。也体现出管理员的权限级别。其用例图如3.3图所示：

图3.3 管理员用例图

**3.4.2 管理员用例**

****用户的功能也有许多，例如个人主题管理，个人信息管理，个人回复管理等。可以看出用户的权限在个人。其用例图如3.4图所示：

图3.4 用户用例图

**3.5 软硬件环境要求**

软件要求：Windows XP或WIN7操作系统。数据库要求Mysql5.0或以上的数据库。服务器是Apache2.2或者以上。

硬件要求：内存是1G以上，intel586 / 50MHz以上的CPU。

**第4章 系统总体设计**

在总体设计阶段的基本任务是回答并描述如何实现目标体系的内容。

**4.1 整体功能模块分析**

根据需求分析的说明，该系统应具有基本功能可以总结为以下几个模块，如图4.1所示：

校园之家系统

发表主题

发表回复

删除信息

用户注销

浏览主题

用户登录

用户注册

修改个人信息

增加好友

删除好友

发送信息

查看信息

图4.1 系统组织图

对象包含了：游客、管理员、用户，功能有注册用户、用户的登录、浏览相应主题、回复主题、发表主题、用户的注销、修改个人信息。

**4.2 功能模块设计**

在本节中,我们将进一步划分子模块,每个子系统模块的子模块,每个模块功能说明如下。

**4.2.1 管理员管理模块**

管理员的管理有对管理员的增、删、改、查功能。如图4.2所示：

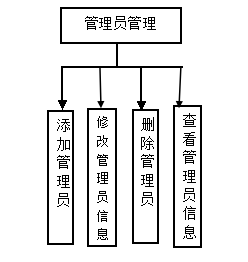


图 4.2 管理员的管理功能图

添加管理员是对管理员的进行新的管理员的添加。

修改管理员信息是对管理员的详细信息进行修改。

删除管理员是对部分需要删除和不使用的管理员用户进行删除。

查看管理员信息是管理员的详细信息进行查看。

**4.2.2 用户管理模块**

用户管理主要包括对用户的增、删、查功能。如图4.3所示：

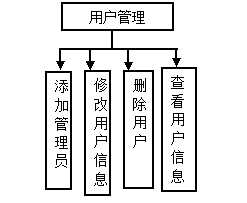


图 4.3 用户管理功能图

添加管理员是对管理员的进行新的用户的添加。

修改用户信息是对用户的详细信息进行修改。

删除用户是对部分需要删除和不使用的用户用户进行删除。

查看用户信息是用户的详细信息进行查看。

**4.2.3 主题管理模块**

主题管理主要包括对主题和回复的增删改查功能。如图4.4所示：

回复模块

主题模块

主题管理

删除回复

查看回复

修改回复

增加回复

查看主题

修改主题

删除主题

增加主题

图4.4 主题管理图

主题模块是对主题进行增删改查的操作。

回复模块是对回复进行增删改查的操作。

**4.3 关系模式设计**

本系统中的设计中，有实体的设计，例如用户，模块，主题，回复等实体设计，那么对于这些实体彼此之间存在关系。对于一个系统的了解，最主要的是对于系统中的各个关系的了解。

那么这写实体之间的关系到底是如何的，如图4.5所示：

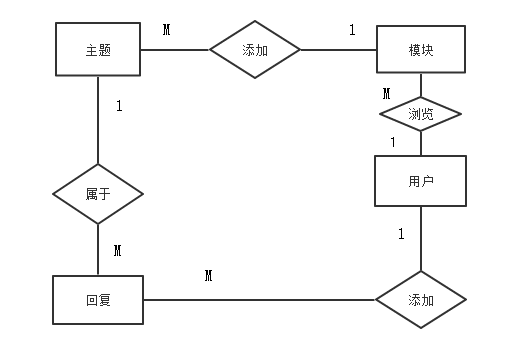


图4.4 主题管理图

**4.4 本章小结**

这部分主要进行了整个系统的总体框架设计,还包括系统的功能,以及系统中使用的字段。

系统总体框架的设计开发项目将系统作为一个整体的概念,使用这个系统,可以了解系统的模块尽快,以便我们可以尽快发现问题,解决问题。

有利于系统的详细设计,每一部分将用于详细设计,尽快明确系统设计思想。

**第5章 详细设计**

详细的设计阶段的任务是在总体的设计的阶段提出了稍微抽象的概括的方法去解决具体的问题,这是“应该如何具体的实现系统”。在此阶段将包括各个模块的数据库的设计和详细的设计,确定如何实现模块所要的算法和数据结构。

**5.1模块设计**

**5.1.1 用户模块设计**

用户的注册的流程，先输入用户的信息，然后对用户输入的信息的正确性和合理性进行校验，校验失败则定位到信息有误的位置,校验成功则跳转到主页。

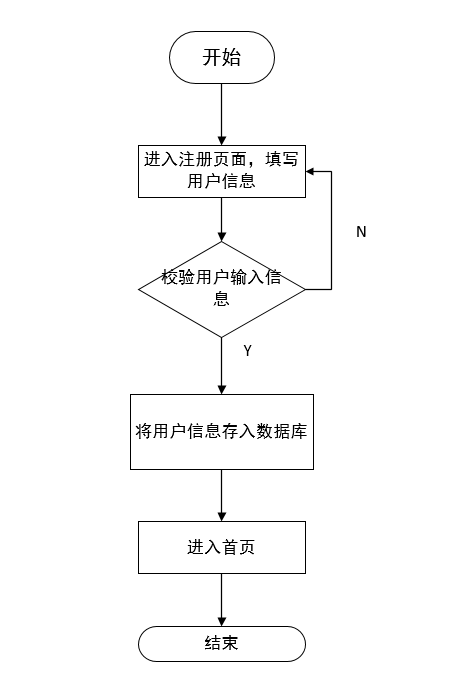


图5.1用户注册流程图

**5.1.2 主题模块设计**

主题模块的流程是浏览主题，如果想要添加主题，首先校验用户是否登录，成功后，需要填写相应的主题信息，通过对信息的校验，校验成功后，跳转到主页面。

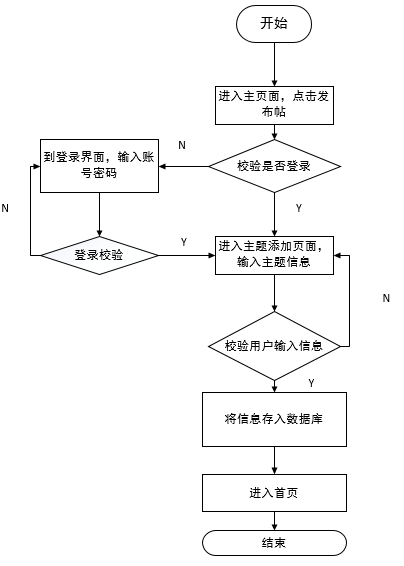


图5.2 主题流程图

**5.1.3 回复模块设计**

回复模块的流程是在主题的详情页面，想要回复主题，写完要回复信息后，点击提交后，首先验证是否登录，然后将回复添加到数据库，跳转到主题详情界面。

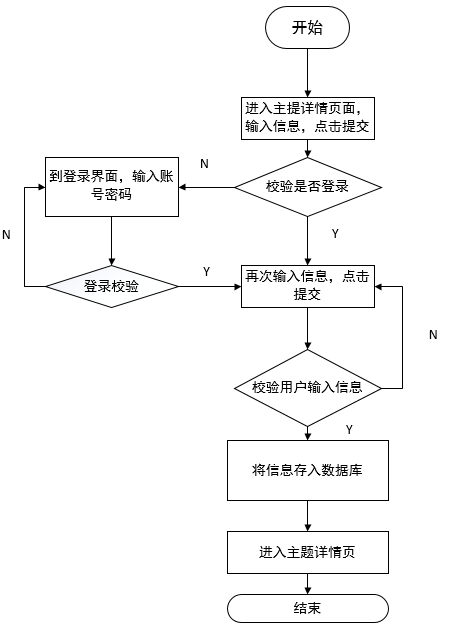


图5.3 回复模块流程图

**5.2 界面设计**

**5.2.1 首页界面设计**

页面设计如图5.4所示：

图5.4 首页部分界面



图5.5 首页部分界面

**5.2.2 主题界面设计**

页面设计如图：

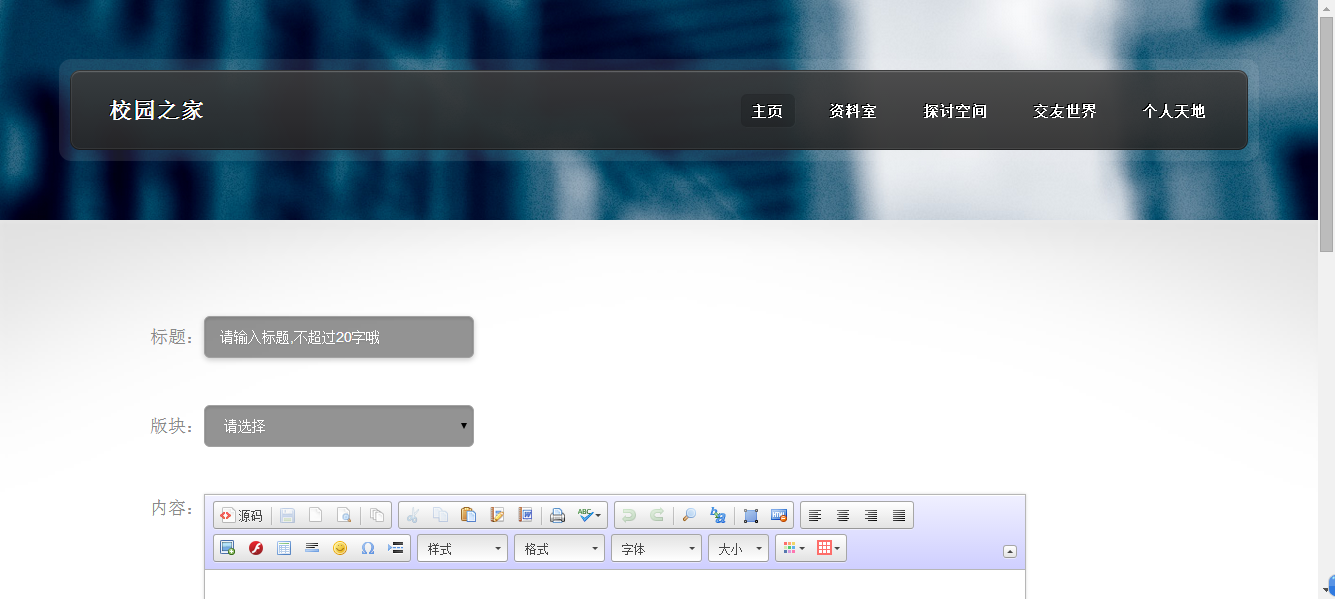
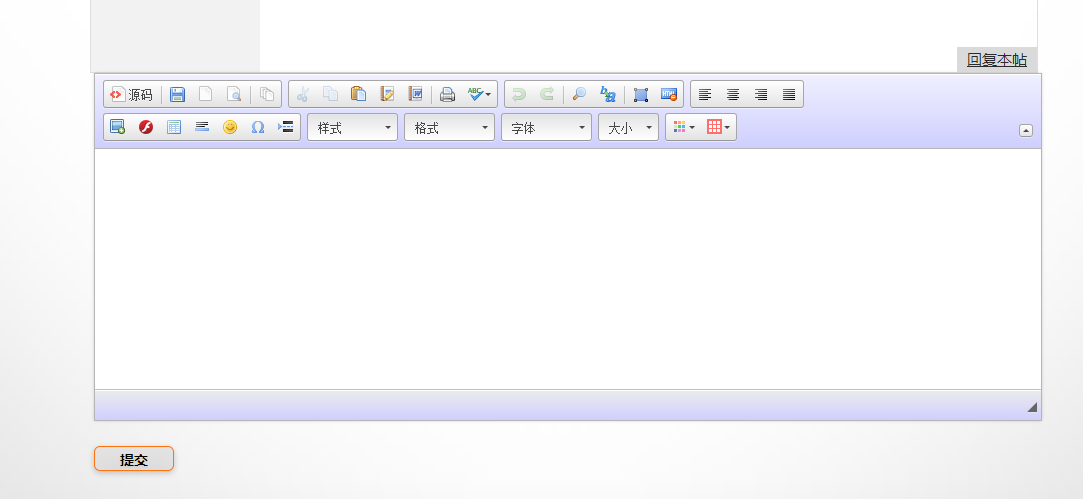
图5.6 主题部分界面

图5.7 主题部分界面

**5.2.3 主题详情界面设计**

 图5.8 主题详情部分页面

主题详情页面还有回复的功能，如图5.9所示。

图5.9 主题详情部分页面

**5.2.4 登录界面设计**

页面设计如图5.10所示：



图5.10 登录用户界面

**5.2.4 注册界面设计**

页面设计如图：



图5.11 注册用户界面

**5.3 数据库设计**

**5.3.1 数据结构设计**

根据描述，本系统中抽象的数据模型为：主题、回复、用户、模块。根据需求分析，可设计如下表：

1. 用户的信息表：id、账号、密码、姓名、出生日期、性别、E-mail地址、联系电话、用户头像、QQ、注册时间。
2. 论坛的模块表：id、名称、上级模块Id、创建时间
3. 论坛的主题表：id、名称、内容、发布者、所属模块、是否为精华帖子、创建时间.
4. 回复表：Id、标题、回复Id、内容、回复者Id、回复时间.

对于数据库的字段详细的信息请看附录。如图所示。

模板的详细的信息表：

表5.1 模板数据设计表

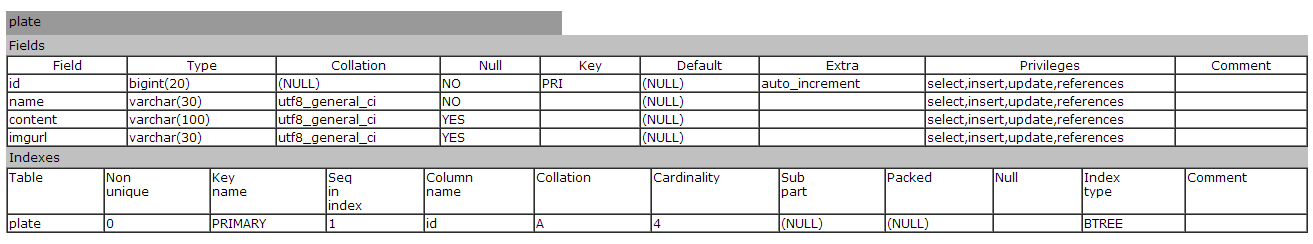
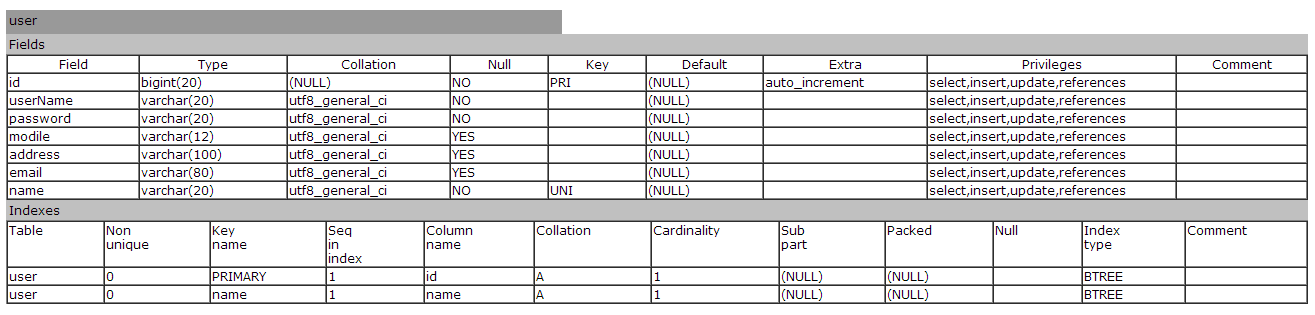
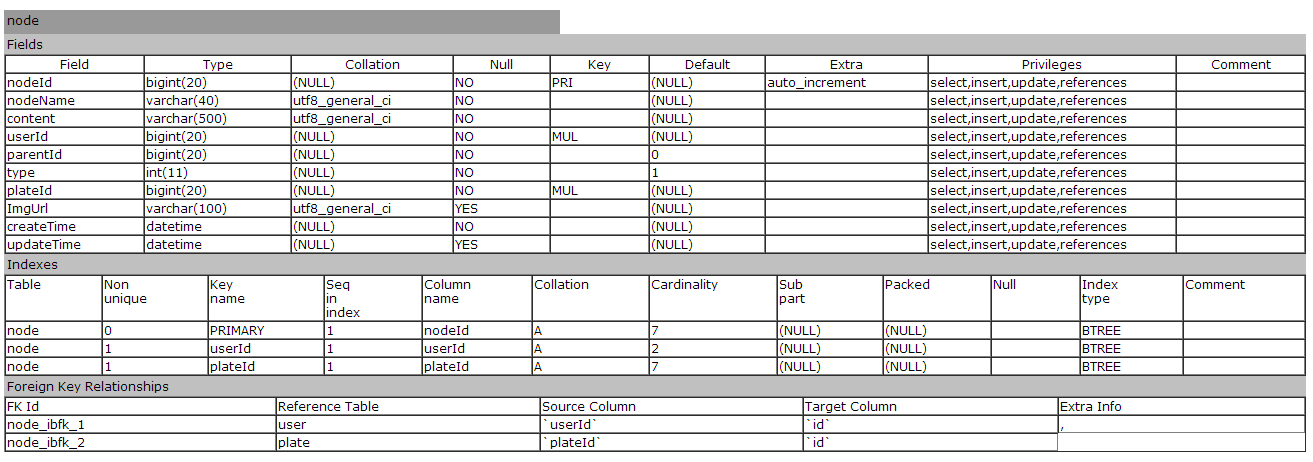
用户的详细的信息表：

表5.2 用户数据设计表



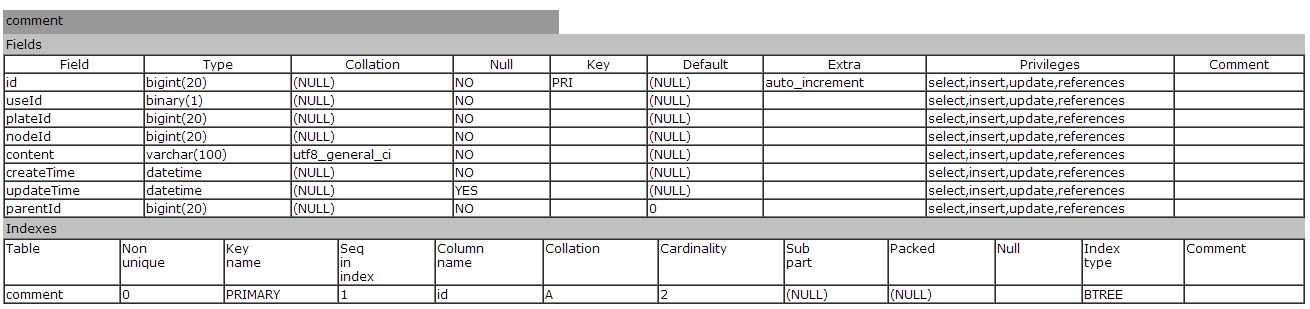
主题的详细信息表：

表5.3 主题数据设计表



回复的详细的信息表：

表5.4 回复数据设计表



**第6章 编码实现**

在完成需求分析、概要设计、详细设计的步骤后，接下来就是编码。

**6.1 数据库链接**

系统采用Spring、SpirngMVC、MyBatis架构，数据库采用MySQL，使用Spring的配置文件spring-config-datasource.xml连接MySQL，其连接的MySQL的详细配置的信息如下：

<bean id="demotionSysDataSource" class="com.jolbox.bonecp.BoneCPDataSource"

destroy-method="close">

<!--驱动配置-->

<property name="driverClass" value="${jdbc.driverClass}" />

<!--相应驱动的jdbcUrl-->

<property name="jdbcUrl" value="${demotion.jdbc.url}" />

<property name="username" value="${demotion.jdbc.username}" />

<property name="password" value="${demotion.jdbc.password}" />

<!-- 分区数 ，默认值2，最小1，推荐3-4，视应用而定-->

<property name="partitionCount" value="${demotion.jdbc.partitionCount}" />

<!-- 每次去拿数据库连接的时候一次性要拿几个,默认值：2 -->

<property name="acquireIncrement" value="${demotion.jdbc.acquireIncrement}" />

<!-- 每个分区最小的连接数 -->

<property name="minConnectionsPerPartition" value="${demotion.jdbc.minConnectionsPerPartition}" />

<!-- 每个分区最大的连接数 -->

<property name="maxConnectionsPerPartition" value="${demotion.jdbc.maxConnectionsPerPartition}" />

<!-- 缓存prepared statements的大小，默认值：0 -->

<property name="statementsCacheSize" value="${demotion.jdbc.statementsCacheSize}" />

<!-- 每个分区释放链接助理进程的数量-->

<property name="releaseHelperThreads" value="${demotion.jdbc.releaseHelperThreads}" />

<!-- 检查数据库连接池中空闲连接的间隔时间，单位是分 -->

<property name="idleConnectionTestPeriodInMinutes" value="${demotion.jdbc.idleConnectionTestPeriodInMinutes}" />

<property name="idleMaxAgeInMinutes" value="${demotion.jdbc.idleMaxAgeInMinutes}" />

<property name="connectionTimeoutInMs" value="${connectionTimeoutInMs}" />

<property name="closeConnectionWatchTimeoutInMs" value="${closeConnectionWatchTimeoutInMs}" />

<property name="defaultAutoCommit" value="true" />

<property name="lazyInit" value="false" />

</bean>

使用配置 Dao的形式进行数据库操作，DAO对数据表的增加,删除,查询,修改,DAO没有其他逻辑操作。由前台页面请求到相应的行动逻辑运算,这是界面编程,模块之间的低耦合,有利于程序的可伸缩性。

**6.2虚拟主机的搭建**

1.打开apache目录下httpd文件，找到如下模块

 NameVirtualHost \*:80

    # Virtual hosts

    #Include conf/extra/hosts.conf

去掉#，这样就能打开了hosts虚拟主机的文件。要在hosts.conf配置好。

2.打开hosts的文件后，要配置好虚拟主机的localhost，参照hosts的文件中的实例，修改成如下：

<VirtualHost \*:80>

    ServerAdmin www.schoolhome.com

    DocumentRoot "f:\wamp\www"

</VirtualHost>

修改配置如下：

目录应该修改局部WAMP环境列表

3.在重新启动Apache，localhost可以知道本地主机可以打开，配置应该是很容易的。

4.打开host的路径为C:\WINDOWS\system32\drivers\etc\hosts，加一句代码

  127.0.0.1       www.schoolhome.com

**6.3 功能实现**

模块包括用户的一些功能，具体的可分为用户的注册、用户的登录、用户的注销、修改个人的信息、查看版面、查看主题、查看回复、发表主题和发表回复等功能。

由于主题是在模板中存在的，所以在版块是可以发表主题，但是回复只能是针对主题的信息进行回复，所以必须在相应的主题下才能发表回复信息，也就是在主题的页面下才能够点击发表回复信息。

现将部分功能说明如下：

首页：游客或者用户来到我的网站，需要先进入首页进行浏览。

部分代码展示：

@RequestMapping("/index.html")

public ModelAndView indexhtml(HttpServletRequest request, HttpServletResponse response){

List<Plate> plateList= plateService.selectPlateAll();//获取所有模板

List<Plate> plateslist=new ArrayList<Plate>();

if(plateList.size()>3){

plateslist=plateList.subList(0,3);//取前三个模板

}

Plate plateRight=new Plate();

if(plateList.size()==4){

plateRight=plateList.get(3);

}

ModelAndView model=new ModelAndView("index");//定义模板类

model.addObject("plateList",plateslist);

model.addObject("plateRight",plateRight);

model.addObject("plateListAll",plateList);

return model;

}

用户登录:用户在具有帐号的前提，能够在登录页login.html或者主页index.html中输入账号和密码进行登录，哪呢将用户名和密码提交到LoginAction.java中的对应的login()方法，可以让用户提交的对应的用户名和密码传递给后台的Dao进行查询操作，存在用户信息则登陆成功，否则将返回到登录页面，页面提示为登陆失败。

部分代码展示

ModelAndView model=null;

String name=request.getParameter("username");//获取输入账号

String password=request.getParameter("password");//获取输入密码

logger.info("name:{},password:{}",name,password);

String msg=null;

if(!(stringBlank(name) && stringBlank(password))){

if(stringBlank(name)){

msg="001";

}

if(stringBlank(password)){

msg="002";

}

}

if(stringBlank(msg) && !stringBlank(name) && !stringBlank(password)){

User user=new User();

user.setUserName(name);

user.setPassword(password);

List<User> users= userService.selectUser(user);//查询用户

if(users!=null && users.size()>0){

user=users.get(0);

Cookie cookie=new Cookie("userName",user.getName());//存cookie

cookie.setMaxAge(TIME);

cookie.setPath("/");

response.addCookie(cookie);

ShardedJedis jedis= jedisSentinelPool.getResource();

try{

String key="pin\_"+user.getName();

jedis.setex(key,TIME,user.getName());//存入缓存

}catch (Exception e){

logger.error("",e);

}

model=new ModelAndView("redirect:http://www,schoolhome.com/index.html");//定义模板类

return model;

}else{

msg="003";

}

}

发布主题： 用户可以发布主题，对自己感兴趣的话题进行讨论，点击发贴可以进入发帖界面。

部分代码展示

ModelAndView model=new ModelAndView("node/insert");//定义模板

String msg=null;

if(node!=null){

if(node.getNodeName()==null){

msg="标题不能为空,请确定后提交";

}

if(node.getContent()==null){

msg="内容不能为空，请确定后提交";

}

if(node.getParentId()==null){

node.setParentId(0);

}

String pin=getCookieValue(request, "userName");//获取cookie

ShardedJedis jedis= jedisSentinelPool.getResource();

User user=new User();

if(stringBlank(pin)){

msg="请先登录，再发布信息！";

user.setName(pin);

}

}

查看版面：在首页能够点击相应的模板到相应的版面，并能进行浏览。

部分代码：

@RequestMapping("/nodelist.html")

public ModelAndView nodelistbyId(HttpServletRequest request, HttpServletResponse response){

ModelAndView model=new ModelAndView("nodelist");//定义模板

String id=request.getParameter("id");

Node node=new Node();

if(id==null){

id="1";

}

node.setPlateId(Integer.valueOf(id));

List<Node> nodeList=nodeService.selectNode(node);//查询主题

List<NodeVo> nodeVos= changeVoList(nodeList);

model.addObject("nodeList",nodeVos);

model.addObject("plateListAll",plateService.selectPlateAll());

return model;

}

private List<NodeVo> changeVoList( List<Node> nodeList){

List<NodeVo> list=new ArrayList<NodeVo>();

if(nodeList!=null){

for(Node node:nodeList){

list.add(changVo(node));

}

}

return list;

}

主题详情页:点击模块中的任意主题进入主题详情页。

部分代码：

@RequestMapping("/nodeDetails.html")

public ModelAndView nodeDetails(HttpServletRequest request, HttpServletResponse response){

String nodeId=request.getParameter("nodeId");//获取主题id

ModelAndView model=null;

Integer id=null;

String msg=null;

try{

id=Integer.valueOf(nodeId);

}catch (Exception e){

logger.error("",e);

msg="001";

}

if(stringBlank(msg)){

Node node=new Node();

node.setNodeId(id);

List<Node> nodeList=nodeService.selectNode(node);//查询主题

if(nodeList!=null && nodeList.size()>0){

Comment comment=new Comment();

comment.setNodeId(id);

List<Comment> list=commentService.selectComment(comment);//获取回复列表

model=new ModelAndView("node/nodeDetails");//定义模板

if(list!=null){

model.addObject("commentList",changeComment(list));

}

model.addObject("node",changVo(nodeList.get(0)));

}

}

return model;

}