阳江职业技术学院

毕业论文（设计）

题目 Blog 博客网站设计与实现

系别 信息工程系 专业 计算机网络技术

姓名 邹玉潇 学号 201630622153

指导老师 职称

2018 年 12 月 7 日

##### 摘要

最近几年博客在国内发展的非常迅猛,它可以作为其使用者个人或者工作中的文章、日记等分享的平台,其渐渐成为新兴的网路媒介,并且扩展至营销、商业推广等商业应用,越来越多的人们建立了属于自己的博客来展示自己的风采。

该博客网站设计的目的旨在建立以spring、springmvc、mybatis、springboot和MySql数据库为工具,功能简洁、结构灵活且精致、轻巧的个人博客网站系统。其中网络日志的管理作为本系统的主要目标,且同时可以满足用户对于发表博客,评论他人博客，个人信息等的管理和展示。

该设计的开发环境为IntelliJ IDEA 2017,使用java编程语言进行编码,采用的是MySql数据库。

该项目着重于后台功能实现。

关键字：java，spring，springmvc，mybatis，springboot，thymeleaf；个人博客

**abstract**

**In recent years, blog has developed rapidly in China. It can be used as a platform for sharing articles, diaries and so on. It has gradually become a new network media, and expanded to commercial applications such as marketing and business promotion. More and more people have established their own blogs to show their style. 。**

**The purpose of this blog website design is to build a simple, flexible and delicate personal blog website system with spring, spring mvc, mybatis ，springbootand MySql database as tools. Among them, the management of network log is the main goal of this system, and it can also satisfy users'management and display of blog publishing, commenting on others' blogs, personal information and so on.**

**The development environment of this design is IntelliJ IDEA 2017. It uses Java programming language to code, and uses MySql database.**

**Key words: java, spring, spring mvc, mybatis, spring boot，thymeleaf; personal blog**

##### 目录

目录

**摘要2**

**目录3**

**第一章 绪论4**

1.1博客的概念4

1.1博客的历史4

1.1博客带来了什么5

1.1博客的崛起6

1.1博客今后的发展7

**第二章 开发工具及相关技术8**

1.1开发工具8

2.1IntelliJ IDEA8

2.1MySql8

1.1开发技术8

2.2.1Spring8

2.2.2Springmvc13

2.2.3Mybatis13

2.2.4Springboot14

**第三章 功能实现15**

3.1.1需求分析16

3.1.2数据库设计16

3.1.3功能概况17

3.1.4各模块实现及核心技术18

3.1.4.1用户19

3.1.4.1.1注册19

3.1.4.1.1登录23

3.1.4.1.1用户信息管理24

3.1.4.1.1管理员24

3.1.4.2博文25

3.1.4.3类别26

3.1.4.4评论27

**结论28**

**参考文献29**

##### 第一章 绪论

###### 1.1博客的概念

blog的全名应该是Web log，中文意思是“网络日志”，后来缩写为Blog，而博客(Blogger)就是写Blog的人。从理解上讲，博客是“一种表达个人思想、网络链接、内容，按照时间顺序排列，并且不断更新的出版方式”。简单的说博客是一类人，这类人习惯于在网上写日记。

Blog是继Email、BBS、ICQ之后出现的第四种网络交流方式，是网络时代的个人“读者文摘”，是以超级链接为武器的网络日记，是代表着新的生活方式和新的工作方式，更代表着新的学习方式。具体说来，博客(Blogger)这个概念解释为使用特定的软件，在网络上出版、发表和张贴个人文章的人。

一个Blog其实就是一个网页，它通常是由简短且经常更新的帖子所构成，这些张贴的文章都按照年份和日期倒序排列。Blog的内容和目的有很大的不同，从对其他网站的超级链接和评论，有关公司、个人构想到日记、照片、诗歌、散文，甚至科幻小说的发表或张贴都有。许多Blogs是个人心中所想之事情的发表，其它Blogs则是一群人基于某个特定主题或共同利益领域的集体创作。

随着Blogging快速扩张，它的目的与最初的浏览网页心得已相去甚远。目前网络上数以千计的 Bloggers发表和张贴Blog的目的有很大的差异。不过，由于沟通方式比电子邮件、讨论群组更简单和容易，Blog已成为家庭、公司、部门和团队之间越来越盛行的沟通工具，因为它也逐渐被应用在企业内部网络(Intranet)中。

###### 1.2博客的历史

最早，是由Jorn Barger在1997年12月提出博客这个名称。但是在1998年，互联网上的博客网站却屈指可数。那时，Infosift的编辑Jesse J.Garrett想列举一个博客类似站点的名单，便在互联网上开始了艰难的搜索。

终于在1998年的12月，他的搜集好了部分网站的名单。他把这份名单发给了Cameron Barrett，Cameron觉得这份名单非常有用，就将它在Camworld网站上公布于众。其它的博客站点维护者发现此举后，也纷纷把自己的网址和网站名称、主要特色都发了过来，这个名单也就日渐丰富。到了1999年初，Jesse的“完全博客站点”名单所列的站点已达23个。

由于Cameron与Jesse共同维护的博客站点列表既有趣又易于阅读，吸引了很多人的眼球。在这种情况下，Peter Merholz宣称：“这个新鲜事物必将引起大多数人的注意。作为未来的一个常用词语，web-blog将不可避免地被简称为blog，而那些编写网络日志的人，也就顺理成章地成为blogger——博客”。这代表着博客被正式命名。

随着博客数量的增多，每个博客网站上编写的网络日志的内容也混杂起来，以至把每一个新出的站点主要内容和特色都不可能搞清楚。Cameron后来就只在网站上登载熟悉的博客站点了。时隔不久，Brigitte Eaton也搜集出了一个名叫“Eaton网络门户”的博客站点名单，并且提出应该以日期为基础组织内容。这也建立了blog分类排列的一大标准。

1999年7月，一个专门制作博客站点的“Pitas”免费工具软件发布了，这对于博客站点的快速搭建起着很关键的作用。随后，上百个同类工具也如雨后春笋般制作出来。这种工具对于加速建立博客站点的数量，是意义重大的。同年的8月份，Pyra发布了Blogger网站，Groksoup也投入运营，使用这些企业所提供的简单的基于互联网的工具，博客站点的数量终于出现了一种爆炸性增长。1999年末，软件研发商Dave Winer向大家推荐Edit This Page网站，Jeff A. Campbell发布了Velocinews网站。所有的这些服务都是免费的，他们的目的也很明确：让更多的人成为博客，来网上发表意见和见解。

###### 1.3博客带来了什么

现在，全世界每天传播的媒体内容，有一半是由6大媒体巨头所控制。其利益驱动、意识形态以及传统的审查制度，使得这些经过严重加工处理的内容已经越来越不适应人们的需求。媒体的工业化，内容出口的工厂化，都在严重影响其发展。

比如，以美联社为例，有近4000人专业记者，每天“制造并出厂”2000万字的内容，每天发布在8500多种报纸、杂志和广播中，把读者当作“信息动物”一样。这种大教堂式的模式主导了整个媒体世界。这时，以个人为中心的博客潮流却开始有力冲击传统媒体，尤其是对新闻界多年形成的传统观念和道德规范。

博客是一种满足“五零”条件(零编辑、零技术、零体制、零成本、零形式)而实现的“零进入壁垒”的网上个人出版方式，从媒体价值链最重要的三个环节：作者、内容和读者三大层次，实现了“源代码的开放”。并同时在道德规范、运作机制和经济规律等层次，将逐步完成体制层面的真正开放，使未来媒体世界完成从大教堂模式到集市模式的根本转变。

博客的出现集中体现了互联网时代媒体界所体现的商业化垄断与非商业化自由，大众化传播与个性化(分众化，小众化)表达，单向传播与双向传播3个基本矛盾、方向和互动。这几个矛盾因为博客引发的开放源代码运动，至少在技术层面上得到了根本的解决。

###### 1.4博客的崛起

这几年，对于所有新闻媒体来说，都品尝到了技术变革的滋味。如今，再没有任何人会否认互联网对媒体带来的革命，但是，好像也没有多少人感知到互联网的神奇：颠覆性的力量似乎并没有来到人间。

所有的核心在于时间。对于性急的人来说，时间如同缓慢的河流，对于从容的人来说，时间又是急流。互联网的力量的确还没有充分施展，因为互联网的商业化起始，到今天仅仅才10年；互联网作为一种新的媒体方式，从尝试到今天，也刚刚跨过10年。

对于一种全新的媒体形式来说，10年实在过于短暂。但是，10年也足以让人们感受到势不可挡的力量，以及依然静静潜伏着的冲击力。而今，随着博客的崭露头角，网络媒体异常的力量开始展现了，声势逐渐发大。虽然，博客依然在大多数人的视野之外，但是，他们改变历史的征程已经启动。

1998年，个人博客网站“德拉吉报道”率先捅出克林顿莱温斯基绯闻案；

2001年，911事件使得博客成为重要的新闻之源，而步入主流；

2002年12月，多数党领袖洛特的不慎之言被博客网站盯住，而丢掉了乌纱帽；

2003年，围绕新闻报道的传统媒体和互联网上的伊拉克战争也同时开打，美国传统媒体公信力遭遇空前质疑，博客大获全胜；

2003年6月，《纽约时报》执行主编和总编辑也被“博客”揭开的真相而下台，引爆了新闻媒体史上最大的丑闻之一；

2004年4月，轰动一时的Gmail测试者大部分从bloggers中产生；

……

这一系列发源于博客世界的颠覆性力量，不但塑造着博客自身全新的形象，而且，也在深刻地改变着媒体的传统和未来走向。

###### 1.5博客今后的发展预测

博客秉承了个人网站的自由精神，但是综合了激发创造的新模式，使其更具开放和建设性。要在网络世界体现个人的存在，张扬个人的社会价值，拓展个人的知识视野，建立属于自己的交流沟通的群体。从这个意义上说，博客将会变得越来越普及，越来越为更多的人接受。2002年8月“博客中国”网站的开通，标志着“博客”(Blog)现象在中国互联网界的显露；11月18日，新闻传播学术网站“紫金网”在改版之际，推出“博客擂台”新栏目。从信息传播的角度看，博客网站、频道的出现对于博客个体而言，意味着将信息采集与发布的通道最大程度的简单化与快捷化。也许你现在还不相信，但是想一想，为什么在互联网的最低潮时期，在个人网站大萧条时刻，在商业网站最凄惨之际，没有一分投资，没有大公司炒作，博客却开始如火如荼？“当然，一切还都需要时间。博客刚刚开始发展，尤其是中国，目前仅仅是博客概念普及和萌芽阶段。不能完全依据现在的状况判断未来，而要看1～2年后，3～5年后的发展步伐。当然，第一步，你自己应该首先去实践一下。”

博客作为一种新表达的方式，它的传播不仅情绪，包括大量的智慧、意见和思想。某种意义上说，它也是一种新的文化现象，博客的出现和繁荣，真正凸现网络的知识价值，标志着互联网发展开始步入更高的阶段。

##### 第二章 开发工具及相关技术介绍

###### 2.1开发工具

2.1.1IntelliJ IDEA

IDEA 全称IntelliJ IDEA，是用于java语言开发的集成环境（也可用于其他语言），IntelliJ在业界被公认为最好的java开发工具之一，尤其在智能代码助手、代码自动提示、重构、J2EE支持、Ant、JUnit、CVS整合、代码审查、 创新的GUI设计等方面的功能可以说是超常的。IDEA是JetBrains公司的产品，这家公司总部位于捷克共和国的首都布拉格，开发人员以严谨著称的东欧程序员为主。

2.1.2MySql

MySQL是一个关系型数据库管理系统，由瑞典MySQL AB 公司开发，目前属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一，在 WEB 应用方面，MySQL是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件。

MySQL是一种关系数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

MySQL所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策，分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库。

###### 2.2开发技术

2.2.1Spring

Spring是一个开放源代码的设计层面框架，他解决的是业务逻辑层和其他各层的松耦合问题，因此它将面向接口的编程思想贯穿整个系统应用。Spring是于2003 年兴起的一个轻量级的Java 开发框架，由Rod Johnson创建。简单来说，Spring是一个分层的JavaSE/EE full-stack(一站式) 轻量级开源框架。

Spring是全面的和模块化的。Spring有分层的体系结构，这意味着你能选择使用它孤立的任何部分，它的架构仍然是内在稳定的。因此从你的学习中，你可得到最大的价值。例如，你可能选择仅仅使用Spring来简单化JDBC的使用，或用来管理所有的业务对象。

它的设计从底部帮助你编写易于测试的代码。Spring是用于测试驱动工程的理想的framework。

Spring对你的工程来说，它不需要一个以上的framework。Spring是潜在地一站式解决方案，定位于与典型应用相关的大部分基础结构。它也涉及到其他framework没有考虑到的内容。

轻量——从大小与开销两方面而言Spring都是轻量的。完整的Spring框架可以在一个大小只有1MB多的JAR文件里发布。并且Spring所需的处理开销也是微不足道的。此外，Spring是非侵入式的：典型地，Spring应用中的对象不依赖于Spring的特定类。

控制反转——Spring通过一种称作控制反转（IoC）的技术促进了低耦合。当应用了IoC，一个对象依赖的其它对象会通过被动的方式传递进来，而不是这个对象自己创建或者查找依赖对象。你可以认为IoC与JNDI相反——不是对象从容器中查找依赖，而是容器在对象初始化时不等对象请求就主动将依赖传递给它。

面向切面——Spring提供了面向切面编程的丰富支持，允许通过分离应用的业务逻辑与系统级服务（例如审计（auditing）和事务（transaction）管理）进行内聚性的开发。应用对象只实现它们应该做的——完成业务逻辑——仅此而已。它们并不负责（甚至是意识）其它的系统级关注点，例如日志或事务支持。

容器——Spring包含并管理应用对象的配置和生命周期，在这个意义上它是一种容器，你可以配置你的每个bean如何被创建——基于一个可配置原型（prototype），你的bean可以创建一个单独的实例或者每次需要时都生成一个新的实例——以及它们是如何相互关联的。然而，Spring不应该被混同于传统的重量级的EJB容器，它们经常是庞大与笨重的，难以使用。

框架——Spring可以将简单的组件配置、组合成为复杂的应用。在Spring中，应用对象被声明式地组合，典型地是在一个XML文件里。Spring也提供了很多基础功能（事务管理、持久化框架集成等等），将应用逻辑的开发留给了你。

MVC——Spring的作用是整合，但不仅仅限于整合，Spring 框架可以被看做是一个企业解决方案级别的框架。客户端发送请求，服务器控制器（由DispatcherServlet实现的)完成请求的转发，控制器调用一个用于映射的类HandlerMapping，该类用于将请求映射到对应的处理器来处理请求。HandlerMapping 将请求映射到对应的处理器Controller（相当于Action）在Spring 当中如果写一些处理器组件，一般实现Controller 接口，在Controller 中就可以调用一些Service 或DAO 来进行数据操作 ModelAndView 用于存放从DAO 中取出的数据，还可以存放响应视图的一些数据。 如果想将处理结果返回给用户，那么在Spring 框架中还提供一个视图组件ViewResolver，该组件根据Controller 返回的标示，找到对应的视图，将响应response 返回给用户。

所有Spring的这些特征使你能够编写更干净、更可管理、并且更易于测试的代码。它们也为Spring中的各种模块提供了基础支持。

轻量——从大小与开销两方面而言Spring都是轻量的。完整的Spring框架可以在一个大小只有1MB多的JAR文件里发布。并且Spring所需的处理开销也是微不足道的。此外，Spring是非侵入式的：典型地，Spring应用中的对象不依赖于Spring的特定类。

控制反转——Spring通过一种称作控制反转（IoC）的技术促进了低耦合。当应用了IoC，一个对象依赖的其它对象会通过被动的方式传递进来，而不是这个对象自己创建或者查找依赖对象。你可以认为IoC与JNDI相反——不是对象从容器中查找依赖，而是容器在对象初始化时不等对象请求就主动将依赖传递给它。

面向切面——Spring提供了面向切面编程的丰富支持，允许通过分离应用的业务逻辑与系统级服务（例如审计（auditing）和事务（transaction）管理）进行内聚性的开发。应用对象只实现它们应该做的——完成业务逻辑——仅此而已。它们并不负责（甚至是意识）其它的系统级关注点，例如日志或事务支持。

容器——Spring包含并管理应用对象的配置和生命周期，在这个意义上它是一种容器，你可以配置你的每个bean如何被创建——基于一个可配置原型（prototype），你的bean可以创建一个单独的实例或者每次需要时都生成一个新的实例——以及它们是如何相互关联的。然而，Spring不应该被混同于传统的重量级的EJB容器，它们经常是庞大与笨重的，难以使用。

框架——Spring可以将简单的组件配置、组合成为复杂的应用。在Spring中，应用对象被声明式地组合，典型地是在一个XML文件里。Spring也提供了很多基础功能（事务管理、持久化框架集成等等），将应用逻辑的开发留给了你。

MVC——Spring的作用是整合，但不仅仅限于整合，Spring 框架可以被看做是一个企业解决方案级别的框架。客户端发送请求，服务器控制器（由DispatcherServlet实现的)完成请求的转发，控制器调用一个用于映射的类HandlerMapping，该类用于将请求映射到对应的处理器来处理请求。HandlerMapping 将请求映射到对应的处理器Controller（相当于Action）在Spring 当中如果写一些处理器组件，一般实现Controller 接口，在Controller 中就可以调用一些Service 或DAO 来进行数据操作 ModelAndView 用于存放从DAO 中取出的数据，还可以存放响应视图的一些数据。 如果想将处理结果返回给用户，那么在Spring 框架中还提供一个视图组件ViewResolver，该组件根据Controller 返回的标示，找到对应的视图，将响应response 返回给用户。

所有Spring的这些特征使你能够编写更干净、更可管理、并且更易于测试的代码。它们也为Spring中的各种模块提供了基础支持。

1.方便解耦，简化开发

通过Spring提供的IoC容器，我们可以将对象之间的依赖关系交由Spring进行控制，避免硬编码所造成的过度程序耦合。有了Spring，用户不必再为单实例模式类、属性文件解析等这些很底层的需求编写代码，可以更专注于上层的应用。

2.AOP编程的支持

通过Spring提供的AOP功能，方便进行面向切面的编程，许多不容易用传统OOP实现的功能可以通过AOP轻松应付。

3.声明式事务的支持

在Spring中，我们可以从单调烦闷的事务管理代码中解脱出来，通过声明式方式灵活地进行事务的管理，提高开发效率和质量。

4.方便程序的测试

可以用非容器依赖的编程方式进行几乎所有的测试工作，在Spring里，测试不再是昂贵的操作，而是随手可做的事情。例如：Spring对Junit4支持，可以通过注解方便的测试Spring程序。

5.方便集成各种优秀框架

Spring不排斥各种优秀的开源框架，相反，Spring可以降低各种框架的使用难度，Spring提供了对各种优秀框架（如Struts,Hibernate、Hessian、Quartz）等的直接支持。

6.降低Java EE API的使用难度

Spring对很多难用的Java EE API（如JDBC，JavaMail，远程调用等）提供了一个薄薄的封装层，通过Spring的简易封装，这些Java EE API的使用难度大为降低。

7.Java 源码是经典学习范例

Spring的源码设计精妙、结构清晰、匠心独运，处处体现着大师对Java设计模式灵活运用以及对Java技术的高深造诣。Spring框架源码无疑是Java技术的最佳实践范例。如果想在短时间内迅速提高自己的Java技术水平和应用开发水平，学习和研究Spring源码将会使你收到意想不到的效果。

Spring提供许多功能，在此我将快速地依次展示其各个主要方面。

spring

spring

首先，让我们明确Spring范围。尽管Spring覆盖了许多方面，但我们已经有清楚的概念，它什么应该涉及和什么不应该涉及。

Spring的主要目的是使JavaEE易用和促进好编程习惯。

Spring不重新开发已有的东西。因此，在Spring中你将发现没有日志记录的包，没有连接池，没有分布事务调度。这些均有开源项目提供（例如Commons Logging 用来做所有的日志输出，或Commons DBCP用来作数据连接池），或由你的应用程序服务器提供。因为同样的的原因，我们没有提供O/R mapping层，对此，已有友好的解决办法如Hibernate和JDO。Spring的目标是使已存在的技术更加易用。

例如，尽管我们没有底层事务协调处理，但我们提供了一个抽象层覆盖了JTA或任何其他的事务策略。

Spring没有直接和其他的开源项目竞争，除非我们感到我们能提供新的一些东西。例如，像许多开发人员，我们从来没有为Struts高兴过，并且感到在MVC web framework中还有改进的余地。在某些领域，例如轻量级的IoC容器和AOP框架，Spring有直接的竞争，但是在这些领域还没有已经较为流行的解决方案。（Spring在这些区域是开路先锋。）

Spring也得益于内在的一致性。

所有的开发者都在唱同样的的赞歌，基础想法依然是Expert One-on-One J2EE设计与开发的那些。

并且我们已经能够使用一些主要的概念，例如倒置控制，来处理多个领域。

Spring在应用服务器之间是可移植的。

当然保证可移植性总是一次挑战，但是我们避免任何特定平台或非标准化，并且支持在WebLogic，Tomcat，Resin，JBoss，WebSphere和其他的应用服务器上的用户。

2.2.2Springmvc

Spring MVC属于SpringFrameWork的后续产品，已经融合在Spring Web Flow里面。Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。使用 Spring 可插入的 MVC 架构，从而在使用Spring进行WEB开发时，可以选择使用Spring的SpringMVC框架或集成其他MVC开发框架。

2.2.3Mybatis

MyBatis 本是apache的一个开源项目iBatis, 2010年这个项目由apache software foundation 迁移到了google code，并且改名为MyBatis 。2013年11月迁移到Github。

iBATIS一词来源于“internet”和“abatis”的组合，是一个基于Java的持久层框架。iBATIS提供的持久层框架包括SQL Maps和Data Access Objects（DAOs）。

MyBatis [2] 是支持普通 SQL查询，存储过程和高级映射的优秀持久层框架。MyBatis 消除了几乎所有的JDBC代码和参数的手工设置以及结果集的检索。MyBatis 使用简单的 XML或注解用于配置和原始映射，将接口和 Java 的POJOs（Plain Ordinary Java Objects，普通的 Java对象）映射成数据库中的记录。

每个MyBatis应用程序主要都是使用SqlSessionFactory实例的，一个SqlSessionFactory实例可以通过SqlSessionFactoryBuilder获得。SqlSessionFactoryBuilder可以从一个xml配置文件或者一个预定义的配置类的实例获得。

用xml文件构建SqlSessionFactory实例是非常简单的事情。推荐在这个配置中使用类路径资源（classpath resource)，但你可以使用任何Reader实例，包括用文件路径或file://开头的url创建的实例。MyBatis有一个实用类----Resources，它有很多方法，可以方便地从类路径及其它位置加载资源。

简单易学：本身就很小且简单。没有任何第三方依赖，最简单安装只要两个jar文件+配置几个sql映射文件易于学习，易于使用，通过文档和源代码，可以比较完全的掌握它的设计思路和实现。

灵活：mybatis不会对应用程序或者数据库的现有设计强加任何影响。 sql写在xml里，便于统一管理和优化。通过sql语句可以满足操作数据库的所有需求。

解除sql与程序代码的耦合：通过提供DAO层，将业务逻辑和数据访问逻辑分离，使系统的设计更清晰，更易维护，更易单元测试。sql和代码的分离，提高了可维护性。

提供映射标签，支持对象与数据库的orm字段关系映射

提供对象关系映射标签，支持对象关系组建维护

提供xml标签，支持编写动态sql。

2.2.4Springboot

Spring Boot是由Pivotal团队提供的全新框架，其设计目的是用来简化新Spring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。

##### 第三章 功能实现

###### 3.1.1需求分析

Blog网站主要实现前台，后台的管理功能。前台实现用户注册、登录、管理个人信息、管理博文，回复他人评论；以及查看他人发表的博文、评论他人博文。后台管理实现管理员对用户、博文、评论和博文的分类管理，。

###### 3.1.2 数据库设计

**用户表**

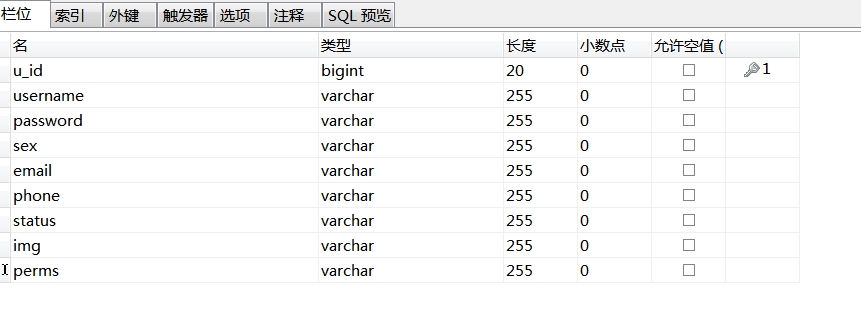


图3-1

**博文类别**

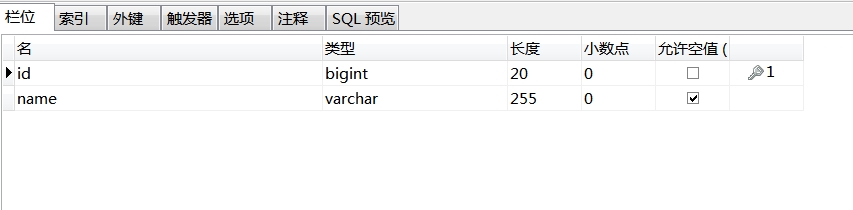
****

图3-2

**评论表**

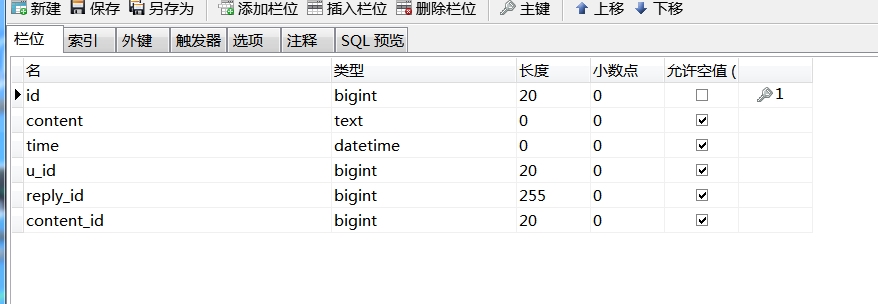
****

图3-3

**评论回复**

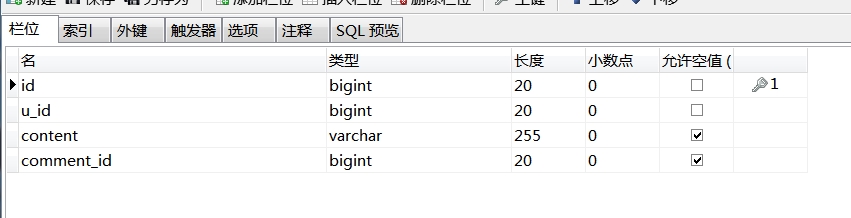
****

图3-4

**博文**

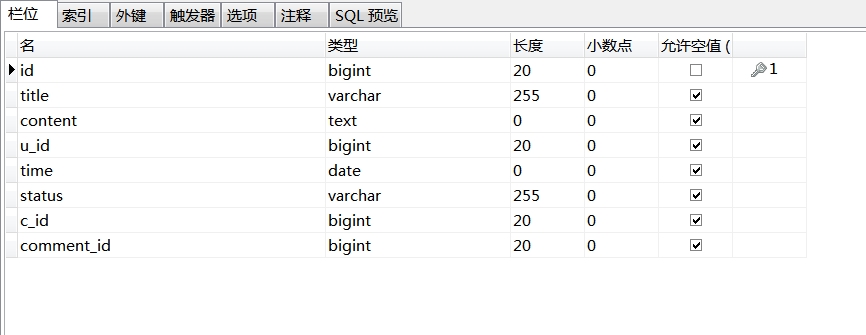
****

图3-5

###### 3.1.3功能概况

###### 3.1.4 各模块的实现及核心技术

3.1.4.1用户模块

3.1.4.1.1 注册

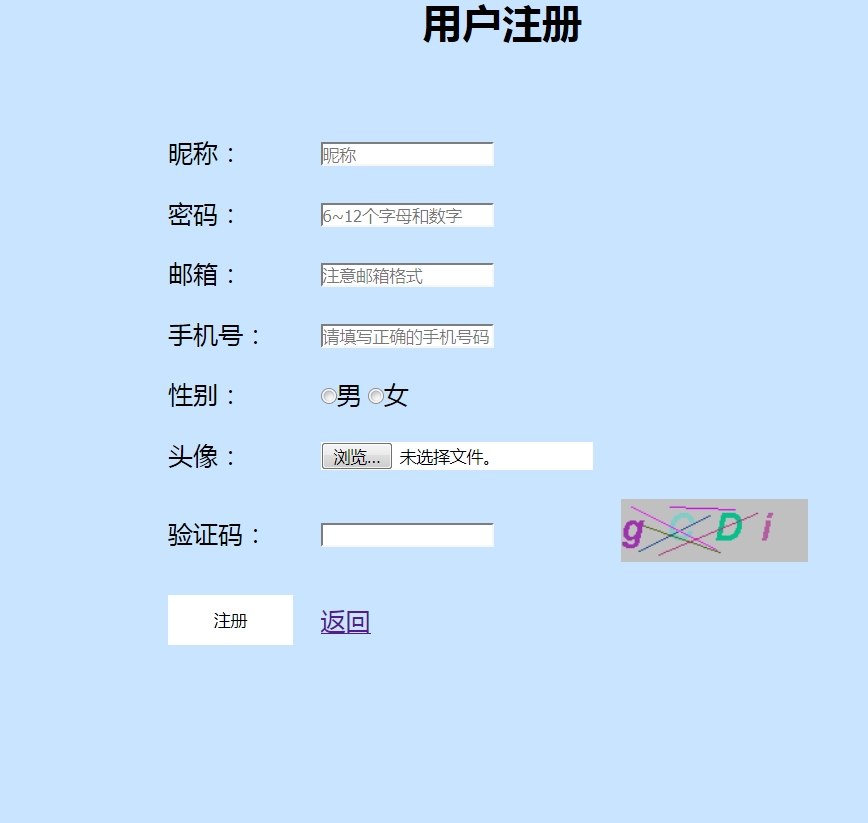


图3-6

如图3-6所示，是注册界面，用户必须把信息填写完整才能完成注册功能，其中头像不是必填项。用户注册成功后，用户信息会存储到数据库中，并跳转到首页。下次访问直接登录即可。为了用户的信息安全，每个手机号、邮箱只可以注册一次；在用户注册时必须正确填写验证码。控制层代码是：

public String reg(@RequestParam String username, @RequestParam String password, @RequestParam String sex,

@RequestParam String email, @RequestParam String phone, MultipartFile file,

@RequestParam String code, HttpSession session, Model model)throws IOException {

System.out.println( file );

if (username == null || username.trim().length() == 0){

model.addAttribute( "username","用户名不能为空" );

return "register";

}

if (password == null || password.trim().length() == 0){

model.addAttribute( "password","密码不能为空" );

return "register";

}

if (sex == null || sex.trim().length() == 0)

{

model.addAttribute( "sex","性别不能为空" );

return "register";

}

if (email == null || email.trim().length() == 0){

model.addAttribute( "email","邮箱不能为空" );

return "register";

}

if (phone == null || phone.trim().length() == 0){

model.addAttribute( "phone","手机号不能为空" );

return "register";

}

if (code == null || code.trim().length() == 0) {

model.addAttribute( "code","验证码不能为空" );

return "register";

}

User eistUsername = userService.selectByUsername(username);

User eistEmail = userService.selectByEmail( email );

User eistPhone = userService.selectByPhone( phone );

String imageCode = String.valueOf( session.getAttribute( "imageCode" ) );

String status = "1";

if (eistUsername != null) {

model.addAttribute( "username","用户名已存在" );

return "register";

}

if (eistEmail != null){

model.addAttribute( "email","该邮箱已被注册" );

return "register";

}

if (eistPhone != null) {

model.addAttribute( "phone","该手机号已注册" );

return "register";

}

if (!email.matches("^[a-z0-9A-Z]+[- | a-z0-9A-Z . \_]+@([a-z0-9A-Z]+(-[a-z0-9A-Z]+)?\\.)+[a-z]{2,}$")){

model.addAttribute( "email","邮箱格式错误" );

return "register";

}

if (!phone.matches( "1[356789]\\d{9}" )) {

model.addAttribute( "phone","手机号格式错误" );

return "register";

}

if (!imageCode.equalsIgnoreCase( code )){

model.addAttribute( "code" ,"验证码错误");

return "register";

}

// 获取用户提交的文件名12345

String oringinalFilename = file.getOriginalFilename();

// 获取Input控件中name属性

String name = file.getName();

// 文件保存路径

// 动态生成文件名--》UUID（生成唯一标识符，网卡号+时间)

String randomUUID = UUID.randomUUID().toString();

// 通过oringinalFilename 获取后缀名

int index = oringinalFilename.lastIndexOf( "." );

String exet = oringinalFilename.substring( index );

System.out.println( exet );

// 根据系统时间生成对应的文件夹，格式为yyyyMMdd

Date date = new Date();

SimpleDateFormat sdf = new SimpleDateFormat( "yyyyMMdd\\HH\\mm\\ss" );

String datestr = sdf.format( date );

String fliePath = "D:\\File\\" + datestr;

File file1 = new File( fliePath );

if (!file1.exists()) {

file1.mkdirs();

}

fliePath += "\\" + randomUUID + exet;

System.out.println( fliePath );

file.transferTo( new File( fliePath ) );

String img = "\\images\\user\\01.jpg " /\*+ randomUUID + exet \*/;

User user = new User();

user.setUsername( username );

user.setPassword( MD5Util.encode( password ) );

user.setSex( sex );

user.setEmail( email );

user.setPhone( phone );

user.setStatus( status );

user.setImg( img );

userService.reg( user );

session.setAttribute( "user",user );

return "index.html";

}

3.1.4.1.2 登录

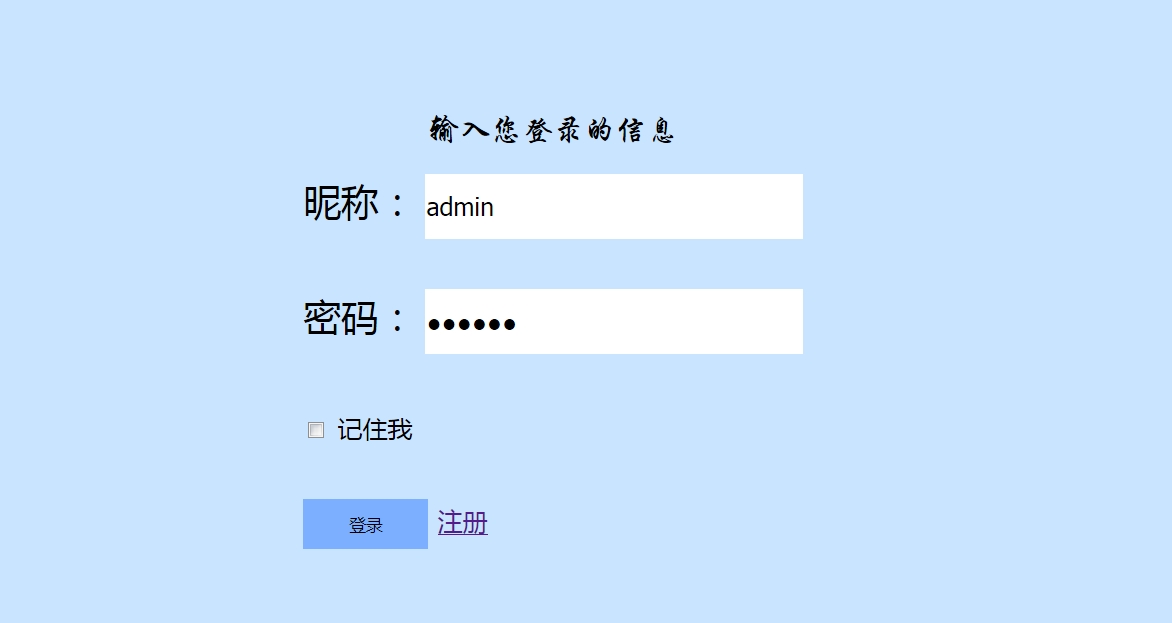


图3-7

如图3-7所示，是登陆页面，在本界面上设计了两个输入文本框，用于输入用户的用户帐号和密码，一个登陆按键，用于登陆操作。还有一个记住密码的选择项，给予用户便捷的操作体验；登陆按键旁是注册链接，如果用户还没有注册本系统的帐号，可以点击此按钮进去注册。登录校验主要采取了shiro框架去实现，代码如下：

public Map<String, Object> login(@RequestParam String username, @RequestParam String password, HttpSession session, Model model){

// 使用shiro 编写认证操作

Subject subject = SecurityUtils.getSubject();

// 封装用户数据

UsernamePasswordToken token = new UsernamePasswordToken(username,MD5Util.encode( password ));

User user = userService.findByUsername( token.getUsername() );

// 执行登录方法

try {

subject.login( token );

}catch (UnknownAccountException e){

return onBadResp( "用户名错误" );

}catch (IncorrectCredentialsException e){

return onBadResp( "密码错误" );

}

session.setAttribute( "user",user );

return onSuccessRep( "登录成功" );

}

3.1.4.1.3 用户信息管理

用户成功登录后，可以进入个人主页，如图3-8所示，对个人信息编辑。

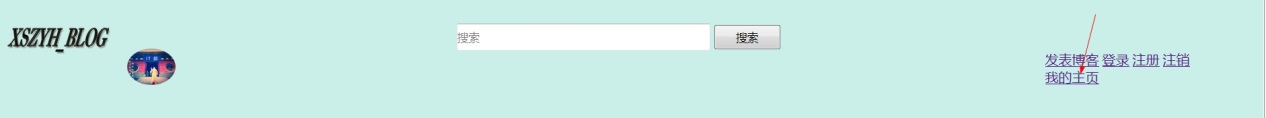


图3-8

3.1.4.1.3 管理员

管理员登录后，可以对博文类别进行管理如添加博文类别、删除博文列表或者修改博文类别，以及其他信息进行管理，如对一些不健康的博文进行管理，或者发表不健康信息博文的博主进行管理。这样可以确保不良信息出现在本站。维护本站的良好风气。如图3-9所示。

图3-9

3.1.4.2博文

博文只有用户登录成功才具有权限操作，登录成功后可以在本站首页查看本站内所有的博文，并可以通过博文标题进行搜索博文，发表博文，在个人主页修改自己发表的博文。其中在编辑博文时可以把未完成的博文保存为草稿，下次可以继续编辑，保存为草稿的博文，他人是无法查看的。如下图3-10~3-12所示：

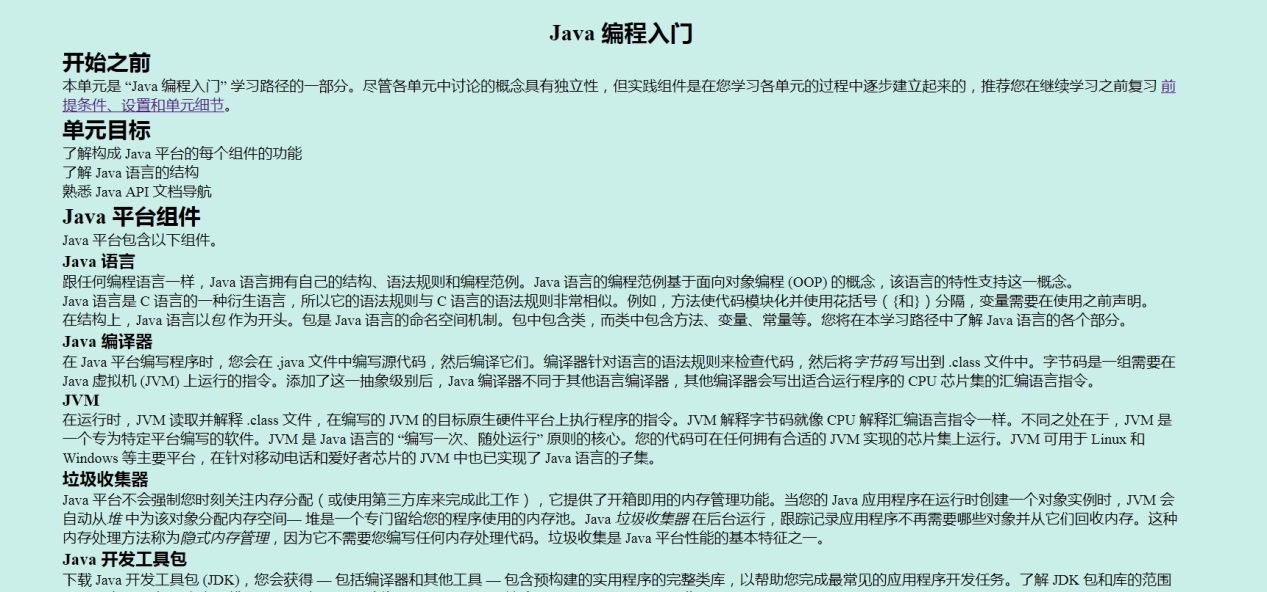


图3-10

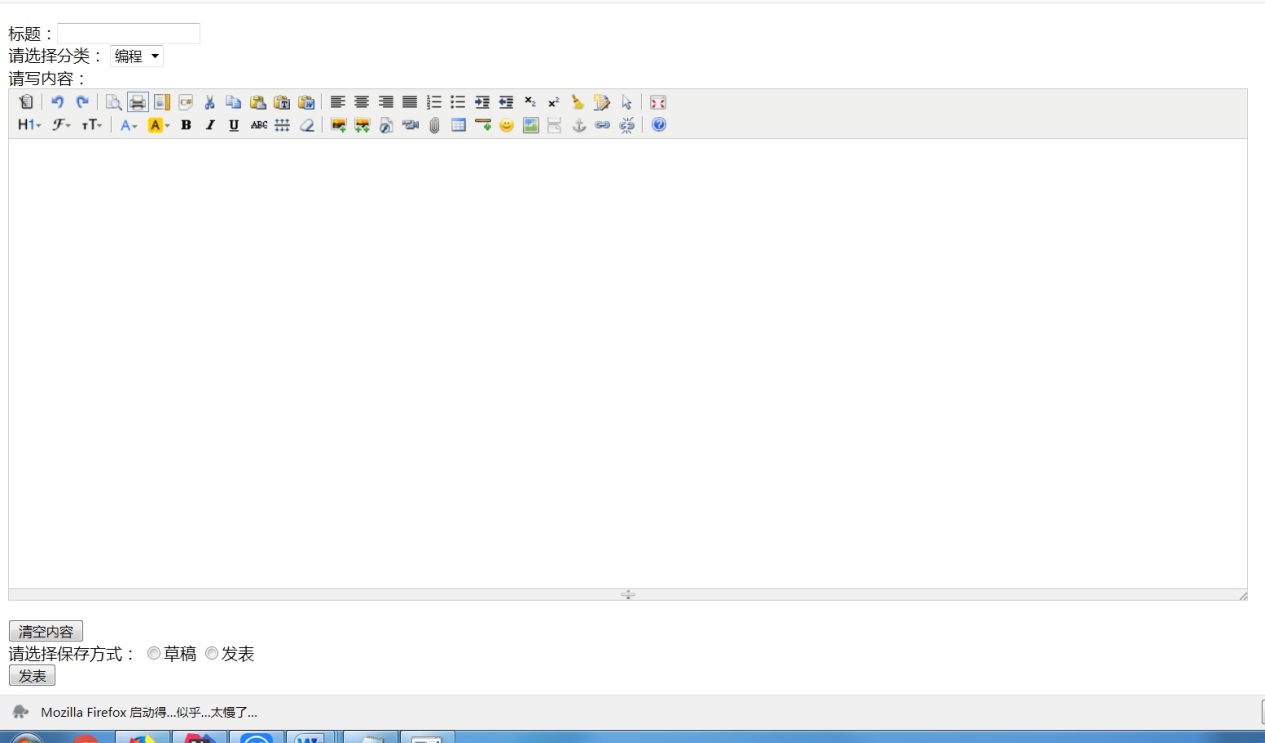


图3-11

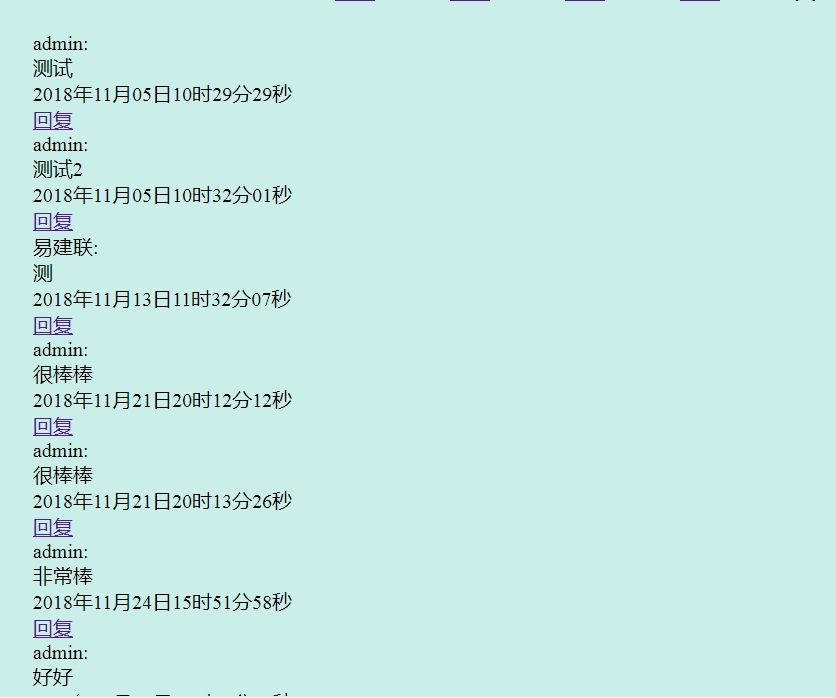
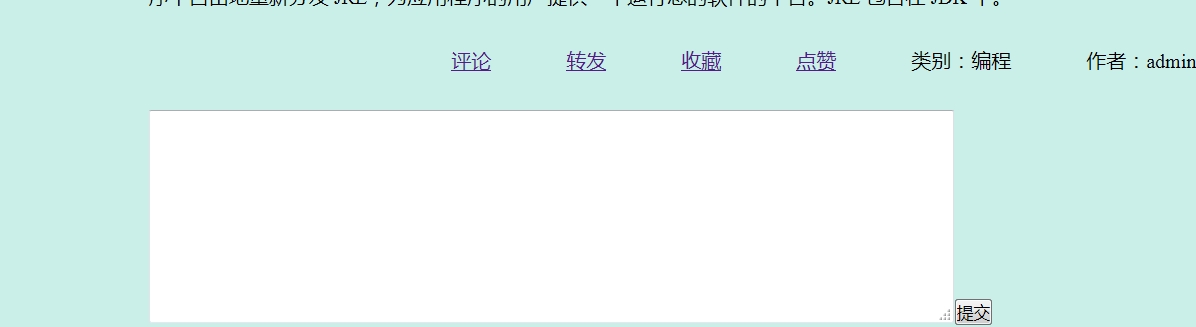
图3-12

3.1.4.3博文类别

博文列表只有管理员才有权限操作，管理员登录在后台管理里进行操作。

3.1.4.4评论

只有用户登录后才具有查看博文的评论和参与评论。



##### 结论

论文研究了博客的需求分析，并对博客的使用和功能做了详细的讲解，并且对本次开发所用到的知识进行详细的介绍。博客可以方便人们（学生）校园生活、提供学习交流机会。

经过了不断的努力，该程序设计与实现基本完成。Blog的开发是基于IntelliJ IDEA 2017平台上的。这个设计和开发是一个非常宝贵的实践经验，因为这个开发的很多内容都是新接触，此次开发对我个人的知识储备提高了很多。

##### 参考文献

1. GQ君. SpringBoot+Mybatis 完整实例 增删查改. 2017年07月21日 17:36:44 https://blog.csdn.net/baidu\_16757561/article/details/75669589
2. 雨降 session的取值与传值 2016年08月13日 19:10:51 https://blog.csdn.net/u011848397/article/details/52200936
3. 夏子藤 javaweb项目中的文件上传下载功能的实现 2016-07-22 14:49 https://www.cnblogs.com/xiaziteng/p/5695158.html