广西科技大学

**普通本科毕业设计（论文）说明书**

**课题名称**  基于微信小程序和云开发的

扫码点餐的设计与开发

学 院 启迪数字学院

专 业 软件工程

班 级 软件Q177班

学 号 201709601295

姓 名 周静茹

指导教师 李 明

二○二一 年 二 月 二十 日

# 摘 要

随着时代的发展、软件技术的成熟，人们也在不断地提升精神需求的层次，人们对个性化的追求越来越强烈，人们希望有一个展示自我、表达自我的个性化平台。由此，个人博客应时而生。个人博客又称为网络个人日记，在博客上，我们不仅可以发表文章，既可以是表达自己所见所想，还可以分享学习资源，别人通过网址就能浏览你的博客，可以评论你的博客内容，易于结识志趣相投之人。所以综合来讲，博客具有三大作用，个人展示、自由表达；学习交流，资源分享；深度交流沟通。

本文在查阅相关论文资料的基础上，分析了个人博客系统的功能需求，以此设计了个人博客系统的功能结构，功能版块由普通用户版块、管理员版块组成。普通用户版块包括浏览文章、评论文章、分享文章、浏览博主信息、搜索文章、查看友情链接；管理员版块包括登录版块、文章管理版块、文章类别管理版块、评论管理版块、个人信息管理版块、系统管理版块。

本课题采用SSM框架搭建项目，使用mysql建立数据库，使用Bootstrap框架搭建前台界面、javascript验证博主登录信息、jsp动态生成网页来完成开发。

**关键词：个性化；个人博客；SSM；javascript；jsp**

# Abstract

With the development of the times and the maturity of software technology, people are also constantly improving the level of spiritual needs. People are more and more strong in the pursuit of individuality. People want to have a personalized platform to show themselves and express themselves. As a result, personal blogs come into being from time to time. Personal blog is also known as online personal diary. On the blog, we can not only publish articles, but also express what we see and think, and share learning resources. Others can browse your blog through the website, comment on your blog content, and easily meet people with similar interests. So generally speaking, blog has three functions: personal display, free expression; learning communication, resource sharing; in-depth communication.

On the basis of consulting relevant papers, this paper analyzes the functional requirements of the personal blog system, and designs the functional structure of the personal blog system. The functional block is composed of ordinary user block and administrator block. The common user module includes browsing articles, commenting articles, sharing articles, browsing blogger information, searching articles and viewing friendship links; the administrator module includes login module, article management module, article category management module, comment management module, personal information management module and system management module.

This project uses SSM framework to build the project, Mysql to build the database, bootstrap framework to build the foreground interface, JavaScript to verify the blogger's login information, JSP to dynamically generate the web page to complete the development.

**Keywords: personalization; personal blog; SSM; JavaScript; JSP**

目录

[摘 要 I](#_Toc2991)

[Abstract II](#_Toc377)

[1. 绪论 1](#_Toc4546)

[1.1 课题研究背景及意义 1](#_Toc2792)

[1.1.1 课题研究背景 1](#_Toc20705)

[1.1.2 课题研究意义 1](#_Toc2147)

[1.2 国内外研究现状 2](#_Toc11399)

[2. 相关技术和开发工具介绍 3](#_Toc27472)

[2.1 B/S简介 3](#_Toc19400)

[2.2 MVC模式、SSM框架、JSP 3](#_Toc14490)

[2.3 Eclipse、MySQL、Tomcat 4](#_Toc22881)

[3. 系统需求分析 5](#_Toc19335)

[3.1 功能性需求分析 5](#_Toc6422)

[3.1.1 总体功能分析 5](#_Toc8209)

[3.1.2 普通用户需求分析 6](#_Toc26908)

[3.1.3 管理员需求分析 6](#_Toc11575)

[3.2 非功能性需求分析 10](#_Toc6055)

[3.2.1 技术可行性 10](#_Toc14020)

[3.2.2 经济可行性 10](#_Toc30228)

[3.3.3 易使用性 10](#_Toc31324)

[3.3.4 安全性 10](#_Toc7757)

[3.3.5 运行环境约束 10](#_Toc31178)

[4. 系统设计 11](#_Toc29923)

[4.1 系统总体设计 11](#_Toc32025)

[4.2 系统前台设计 12](#_Toc30599)

[4.2.1 首页界面 12](#_Toc27905)

[4.2.2 文章信息界面 12](#_Toc17527)

[4.2.3 关于博主界面 14](#_Toc9174)

[4.3 数据库设计 15](#_Toc27682)

[4.3.1 数据库需求分析 15](#_Toc22770)

[4.3.2 数据库概念结构设计 15](#_Toc5784)

[4.3.3 数据库逻辑结构设计 17](#_Toc4567)

[4.4 系统后台设计 19](#_Toc569)

[4.4.1 管理员登录 19](#_Toc20129)

[4.4.2 文章管理 21](#_Toc1261)

[4.4.3 文章类别管理 25](#_Toc29427)

[4.4.4 评论管理 27](#_Toc3028)

[4.4.5 个人信息管理 30](#_Toc5958)

[5. 系统实现 32](#_Toc13787)

[5.1 系统前台实现 32](#_Toc3533)

[5.1.1 首页界面 32](#_Toc18328)

[5.1.2 文章信息界面 33](#_Toc32577)

[5.1.3 关于博主界面 33](#_Toc27478)

[5.2 系统后台实现 34](#_Toc26457)

[5.2.1 管理员登录 34](#_Toc1598)

[5.2.2 文章管理 35](#_Toc13753)

[5.2.3 文章类别管理 38](#_Toc29687)

[5.2.4 评论管理 39](#_Toc16283)

[5.2.5 个人信息管理 41](#_Toc17475)

[6. 系统测试 42](#_Toc20355)

[6.1 测试前台 42](#_Toc12879)

[6.1.1 前台发表评论 42](#_Toc19378)

[6.1.2 查询文章 43](#_Toc3040)

[6.2 测试后台 43](#_Toc28997)

[6.2.1 后台登录 43](#_Toc11900)

[6.2.2 发表文章 47](#_Toc20758)

[6.2.3 后台审核评论 48](#_Toc13116)

[7. 总结与展望 51](#_Toc25819)

[7.1 总结 51](#_Toc26105)

[7.2 展望 52](#_Toc1566)

[致谢 53](#_Toc19169)

[参考文献 54](#_Toc7673)

# 

# 1. 绪论

## 1.1 课题研究背景及意义

### 1.1.1 课题研究背景

互联网发展迅速，各种信息、资源也越来越网络化，人与人之间的互动在网络平台上更加频繁，人们只需在手机、电脑或平板等电子产品上连接网络，就能轻松、快捷获取自己所需的信息、资源，通过QQ、微信、微博等社交软件就能便捷地与人交流。同时，我们也想在网络上有专属的个人展示平台，通过这个平台，我们能够将自己的思想心得、学习资源、生活日常等分享给别人，别人可以评论、浏览。由此，个人博客应运而生。个人博客是一个面向全网的广泛的交流平台，热度居高不下，越来越多人在博客上发表自己的所思所想，以文章、图片、视频等方式呈现，不仅方便快捷，而且能够直抒己见，让人们能够充分表达自我[1]。

再者，往长远处看，互联网不断深入人们的生活，web3.0将是彻底改变人们生活的互联网形式。web3.0将以网络化和个性化为特征，提供更多人工智能服务，而个人博客技术在个性化方面极具代表性。个人博客的内容是由博主自行编辑的，不拘一格；个人博客面向全网，别人通过网址或者搜索关键字就能看到你的博客；访客可以查看你的博客内容，对你的博客发表观点，博主也可以作出回应，因此便形成了交流；访客还能将博客分享到第三方[2]，在web3.0的时代，个性化会是主导的需求，围绕网民互动及个性体验的互联网应用技术会不断的被完善和发展，所以毋庸置疑，个人博客的功能会不断地被丰富和拓展，个人博客使用量也会越来越高。

### 1.1.2 课题研究意义

在这个信息化的时代，我们不仅希望利用资源增强自身技能，丰富自己的学识，我们也更希望展现自己，表达自己。而个人博客就是专属的个人空间，我们可以发表文章、图片、视频、加好友等，可以充分展示自己；个人博客的风格由自己建设管理，充分地保证了个性化；操作简单，用户体验好；普及度高，别人通过浏览器就能访问你的博客；我们还可以在博客上做专题栏目，比如摄影、文学、影视等，因此我们可以通过博客结识更多志同道合的朋友，提供了学习交流的平台，而且提升了交流的高度。个人博客无异于是为我们搭建了一个广阔的个性化平台，而人们的个性化需求也日渐增强，所以深入研究个人博客的开发，是具有长远意义的。

基于以上所述，本文将通过研究如何设计界面简洁、体验友好、交互效果好的个人博客系统以构建一个保障个性化的网络平台，建设一个让人们尽情地抒发自己心中所想，吐露心声的网络场所。

## 1.2 国内外研究现状

博客的起源地是美国和欧洲的一些工业国家，源于英文单词Blog，Blog是Weblog的简称。Web指互联网，log原本专指航海日志，后来泛指任何类型的流水记录，所以Weblog的整体意思就是网络流水记录，也就是“网络日志”[3]。Blog的最初版本只是作为网络资源过滤器来使用的，先是筛选特定的网站，并撰写这些网站的简介，然后把这些网站的相关文字信息放在网页上展示，供用户浏览。随着时代的发展，人们的需求层次在不断提升，人们不再满足于成为信息被动方，也希望自己能够“做主人”、发布信息，所以Blog不再仅仅是是作为过滤器，而是每个人都可以使用Blog将自己的想法发布到网络上，别人不仅可以看到，而且还能对此进行评论，因此Blog又发展成为了一个互动性强的网络交流平台。

2000年Blog传入中国，国人将其音译，博客一词由此而来。博客有三层含义，第一层含义属于名词范畴，就是信息，包括文字、图片、视频、音频等；第二层含义仍旧属于名词范畴，就是作者，也就是发布博客的人；第三层含义属于动词范畴，就是行为，博客是一种动作。随着博客技术的成熟和人们需求的发展，博客的用途也越来越广泛，对于个人来说，博客可以用于发表文章、见解、用于学习交流等，还可以是一种谋生手段，博客写手也是一种职业，他们通过博客访问量来赚取费用；对于企业来说，博客可以用于商业宣传、企业文化传播、产品推广等。

现在，人们越来越注重个性化的发展，重视自我表达、自我展示，扩宽自己的视野，实现自我价值，提高交流深度，而个人博客又正是以自主性、互动性、开放性、共享性为特点，所以个人博客的使用量正在不断增加，深受各界欢迎。个人博客的出现，给我们的学习、生活、工作带来了很大的影响，我们的生活拥有了新的展现形式，我们的学习交流不再局限于自己现实中的朋友圈，当工作遇到难题时，我们还能通过博客找到同行与之交流。

如今，个人博客不但能满足我们的个性化需求，而且已经深入我们的学习、生活、工作中，由此，专题博客也衍生而来。专题博客的应用越来越广泛，人们在博客上发布专题资源、答疑解惑等。当我们碰到问题在浏览器上搜索关键词寻找解答时，常常会看到来自某个人博客提供的信息、资料，能够针对性、有效地帮我们解决问题。在国外，专题博客的运用比较成熟。出现了环保人士发起的个人绿色博客，专门宣传绿色环保知识、呼吁人们保护环境、提倡绿色生活，以促进人类的可持续发展[4]。还有个人理财博客，用于宣讲金融理财知识，面向比较缺乏理财概念或是不善于理财、金融知识掌握较少的人群[5]。可见，个人博客已经能够应用到很多领域中，比如教育、环保、金融等，它的拓展性很强，应用范围广泛，有很大的发展空间[6]。

# 2. 相关技术和开发工具介绍

## 2.1 原生微信小程序

B/S是Brower/Server的缩写，是一种软件结构，客户端只要安装一个浏览器（Browser），通过浏览器与Web服务器进行数据交互。

优缺点：B/S架构最大的优点：总体拥有成本低、维护方便、 分布性强、开发简单，系统的扩展非常容易，只要有一台能上网的电脑就能使用。B/S架构最大的优点：最大的缺点就是通信开销大、系统和数据的安全性较难保障。

## 2.2 css、JavaScript、小程序云开发

MVC模式：

MVC是Model View Controller的缩写，是一种常用的软件开发模式。

model：模型，包含实体层（entity）、dao层、service层。主要特点是可以实现业务层和数据层分离，前端界面以及中间控制器分离、连接后台数据库，降低了整体代码的耦合度[7]。

实体层（entity），用于存放实体类，数据库中有多少个表，在Java中就要定义多少个类，每个表中有多少个数据列，其对应的类中就要定义多少个成员属性，并且数据类型要和数据库表数据列的数据类型相对应。在实体类中除了要定义成员属性，还要定义无参构造方法和有参构造方法、get方法和set方法，无参构造器和有参构造器都是用来实例化对象的，但有参构造器在实例化对象的同时，能够给对象成员属性自定义赋值，set方法用于给对象成员属性自定义赋值，get方法用于获取对象成员属性的值。实体类的作用是用来封装从数据库或者前端页面传过来的数据。

dao层，称为数据访问层或数据持久层，包括定义操作数据库的规范的接口，在该接口中定义方法，对软件基本功能进行说明；还包括实现增、删、查、改功能的类，该类要实现定义规范的接口，然后在该类中重写父接口的方法，在重写方法中写实现增、删、查、改功能的具体代码。dao的作用是用来表述软件要具备的基本功能以及实现这些基本功能。

service层，业务逻辑层，包括定义业务的接口，在该接口中定义方法，说明每个方法实现的业务；还包括实现业务的类，该类要继承定义业务的接口，然后在该类中重写父接口的方法，在重写方法中写实现业务的的代码，其实就是在重写方法中调用dao层的方法。业务有简单的、有复杂的，简单的业务通常是调用dao层中的一两个方法就能完成，复杂的业务需要调用dao层中的多个方法才能完成。

View：视图，是用户看到并与之交互的界面。用户在界面上向服务器发送请求，比如用户进行信息注册或信息查询等操作。

Controller：控制器，接收用户的请求，并予以相应处理，再将处理结果响应给用户，满足用户的需求。Controller实质就是Servlet，当用户通过浏览器向服务器发送请求后，Controller会接收请求，接着调用service层的方法处理请求，然后服务器从Controller获取处理结果，并将处理结果响应给浏览器。

SSM框架：

SSM（Spring+SpringMVC+MyBatis）框架是强化版的MVC。

Spring：是一个开源框架，Spring是一个轻量级的Java 开发框架。Spring相当于是一个大型的类容器，它装载了项目中的所有类，然后程序员只需通过调用相应类的构造方法就能够实例化对象，所以Spring减轻了程序员的代码工作量，不需要程序员写实例化对象的代码[8]。

SpringMVC：负责接收用户请求，它的核心Servlet即分发器，它会解析路径以寻找处理用户请求的Controller，Controller再调用服务层的方法来实现用户的请求然后分发器会通过视图解析器找到相应的视图，由视图将处理结果反馈给用户[8]。

Mybatis：是一个基于Java的持久层框架，封装了jdbc，让数据库底层操作变得透明[8]，让Java连接数据库的操作变得更简便、降低了代码的冗余，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。

JSP：

JSP是Java Server Pages的缩写，它和servlet技术一样，都是一种用于开发动态web资源的技术。

JSP技术的特点：JSP是一种服务器端脚本语言，其出现降低了Servlet编写页面的难度。JSP本质上就是Servlet，因为JSP是首先被翻译成Servlet后才编译运行的，因此JSP能够实现Servlet所能够实现的所有功能，相比html而言，html只能为用户提供静态数据，而Jsp技术允许在页面中嵌套java代码，为用户提供动态数据[9]。

## 2.3 微信开发者工具、MySQL、Tomcat

Eclipse：是一个开放源代码的、基于Java 的可扩展开发平台，集成了Java虚拟机jvm，集编译、运行功能于一体，它会自动帮我们编译src目录下的.java文件，在代码中写入main方法，点击Run后，虚拟机就会运行起来帮我们运行程序。

MySQL：是一个开源的[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511" \t "https://baike.baidu.com/item/MySQL/_blank)，可以处理拥有上千万条记录的大型数据库，为多种[编程语言](https://baike.baidu.com/item/%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80" \t "https://baike.baidu.com/item/MySQL/_blank)提供了 [API](https://baike.baidu.com/item/API" \t "https://baike.baidu.com/item/MySQL/_blank)，其所使用的 SQL 语言是用于访问[数据库](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93/103728" \t "https://baike.baidu.com/item/MySQL/_blank)的最常用标准化语言，优化了 [SQL](https://baike.baidu.com/item/SQL" \t "https://baike.baidu.com/item/MySQL/_blank)查询算法，有效地提高了查询速度[10]。

Tomcat：Tomcat服务器是Apache软件基金Jakarta项目中的一个核心项目，是开源的、免费的。Tomcat实际上就是一个用Java开发的程序。Tomcat是一个Web容器，用来部署web项目，也就是存放web程序。Tomcat的下载、安装、卸载简单，运行时占用的系统资源小，非常适用于开发和调试JavaWeb程序。

# 3. 系统需求分析

## 3.1 功能性需求分析

### 3.1.1 总体功能分析

本文的扫码点餐系统的功能模块分为：普通用户模块、管理员模块。



图3-1个人博客系统功能结构模块图

### 3.1.2 普通用户需求分析

普通用户模块：普通用户在浏览器输入相应网址就可以直接进入博客系统，普通用户能够查询文章、浏览文章内容、评论文章、分享文章到第三方、浏览博主信息、查看友情链接。



图3-2普通用户用例图

### 3.1.3 管理员需求分析

管理员模块：管理员模块包括登录模块、文章管理模块、文章类别管理模块、评论管理模块、个人信息管理模块、系统管理模块。

（1）登录模块：管理员通过用户名、密码登录后台。

（2）文章管理模块：文章管理模块包括发表文章、修改文章、删除文章模块，管理员发表文章时要输入文章标题、内容、定义文章的类别，关键字可有可无。

（3）文章类别管理模块：文章类别管理模块包括增加文章类别、删除文章类别模块，管理员可以添加文章类别，删除文章中没有定义的类别。

（4）评论管理模块：评论管理模块包括审核评论、删除评论模块，管理员可以审核评论，如果审核通过，该评论可以在前台显示，否则不能显示；管理员还可以删除评论。

（5）个人信息管理模块：管理员可以修改个人简介、昵称、个性签名。

（6）系统管理模块：管理员可以添加，修改，删除友情链接路径、修改管理员的后台登录密码、刷新系统缓存和安全退出系统。

图3-3管理员后台管理用例图

图3-4管理员后台文章管理用例图

图3-5管理员后台文章类别管理用例图

图3-6管理员后台评论管理用例图

图3-7管理员后台个人信息管理用例图

图3-8管理员后台系统管理用例图

## 3.2 非功能性需求分析

### 3.2.1 技术可行性

本博客系统使用maven管理项目、mybatis连接数据库，springmvc接收并处理浏览器的请求，前端的布局使用了Bootstrap框架，使用jsp动态生成网页。这些框架和技术功能强大、稳定，而且已经普遍应用了，得到了市场的认可，足够成熟。

### 3.2.2 经济可行性

该博客系统用的开发工具有Eclipse、MySQL、Tomcat，Eclipse是开源的Java程序开发工具；MySQL是一个开源的[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511" \t "https://baike.baidu.com/item/MySQL/_blank)；Tomcat是Apache基金组织的开源的免费的中小型的javaEE服务器。所以，整个开发过程并没有什么经济开销。

### 3.3.3 易使用性

该博客系统无论是前台还是后台，其界面简洁，结构清晰，功能一目了然，易于操作和使用；访问便捷，普通用户在浏览器输入网址即可进入系统，无需注册、登录，管理员的帐号和密码已经分配好，输入帐号和密码即可登录后台。

### 3.3.4 安全性

该博客系统只为博主分配了帐号和密码，可直接登录后台，只有博主才能对系统的信息进行增删查改操作，而且博主登录时对密码进行了加密。其他用户无需注册，只需输入正确的网址就能登录系统，且只能在前台进行浏览、查询文章、评论文章操作。所以，保证了普通用户和博主的个人信息安全。

### 3.3.5 运行环境约束

该博客系统采用的是B/S架构，所以用户端只需要下载一个浏览器，在浏览器上输入对应网址就能进入系统前台。

# 4. 系统设计

## 4.1 系统总体设计

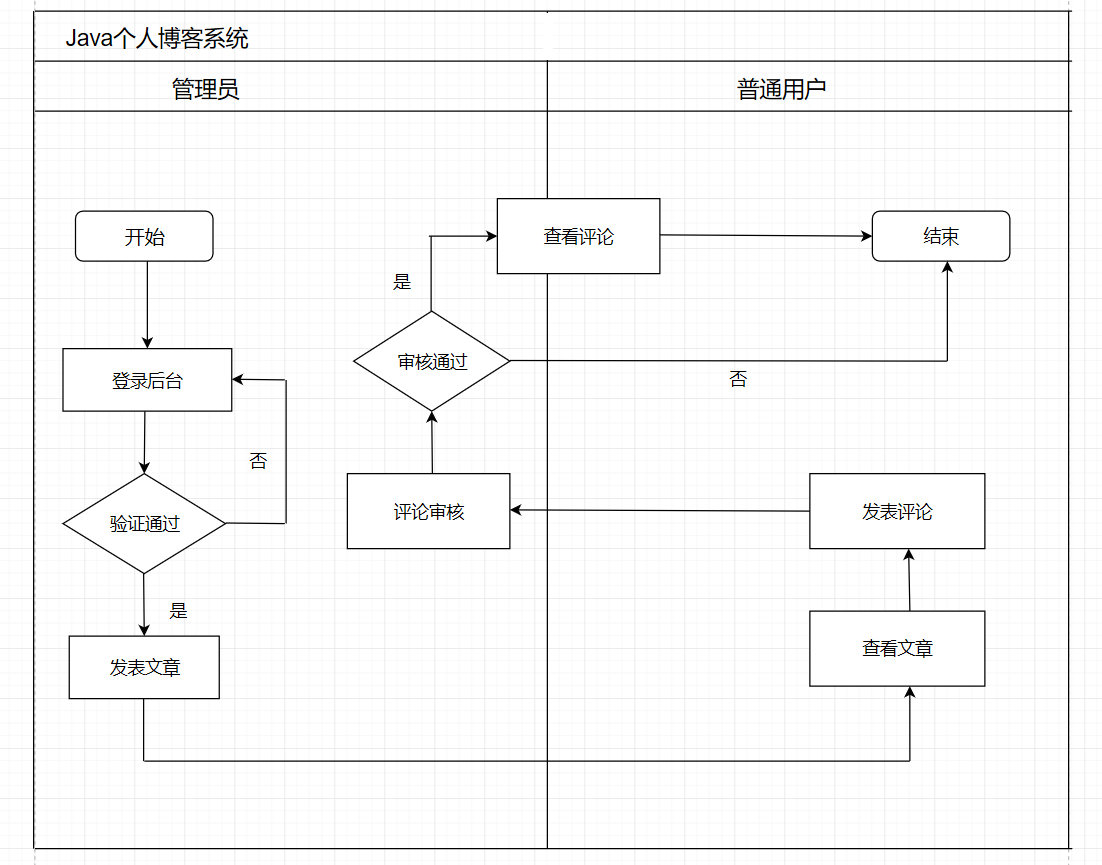


图4-1系统总体流程图

## 4.2 系统前台设计

### 4.2.1 首页界面

用户都无需注册、登录，通过正确的网址就可以直接进入系统前台。



图4-2用户进入系统前台时序图

### 4.2.2 文章信息界面

在文章信息界面可以看到文章信息、评论信息，可以评论文章和分享文章到第三方。

（1）查看文章信息和评论信息：

用户在首页可以直接浏览文章信息；点击文章的日期或标题进入文章信息界面；通过输入关键字查询到符合条件的文章后，点击文章的标题或链接进入文章信息界面；点击“按日志类别”或者“按日志日期”后会将符合条件的文章筛选出来，再点击文章的日期或标题进入文章信息界面。

1. 评论文章：

用户进入文章信息界面后，输入评论内容，再输入验证码无误之后点击发表评论按钮，就能发表评论。

1. 分享文章：

用户进入文章信息界面后，点击第三方链接的图标后就可以将文章分享到第三方。



图4-3用户查询文章时序图



图4-4用户发表评论时序图



图4-5用户分享文章时序图

### 4.2.3 关于博主界面

用户在首页点击关于博主，就可以查看博主个人简介信息。



图4-6用户查看博主信息时序图

## 4.3 数据库设计

### 4.3.1 数据库需求分析

根据功能性需求分析，设计数据库的数据项和数据结构如下：

1. 文章信息，包含的数据项有：文章ID、标题、摘要、发表时间、阅读量、内容、类别ID
2. 管理员信息，包含的数据项有：用户名、密码、昵称、个人简介、个性签名、头像
3. 文章类别，包含的数据项有：类别ID、名称、序号
4. 评论信息，包含的数据项有：评论人IP、文章ID、评论内容、评论日期、评论审核状态
5. 友情链接信息，包含的数据项有：友情链接ID、名称、链接路径、链接序号

### 4.3.2 数据库概念结构设计

根据如上数据库需求分析，用E-R图描述各个实体如下：

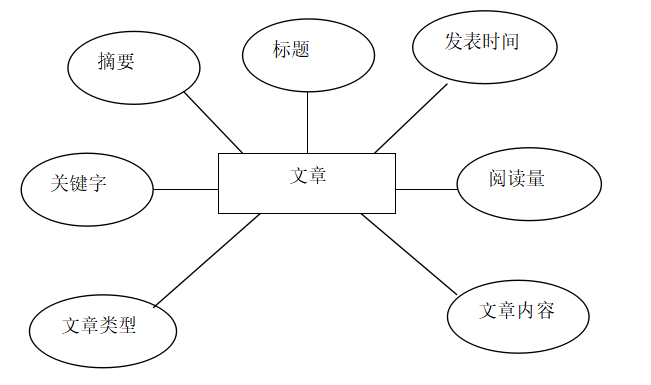


图4-7文章E-R图

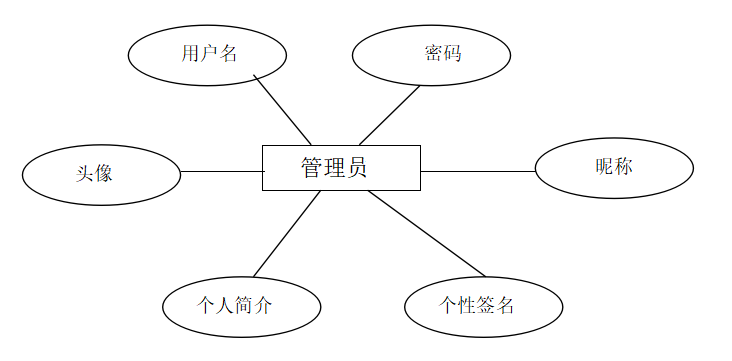


图4-8管理员E-R图

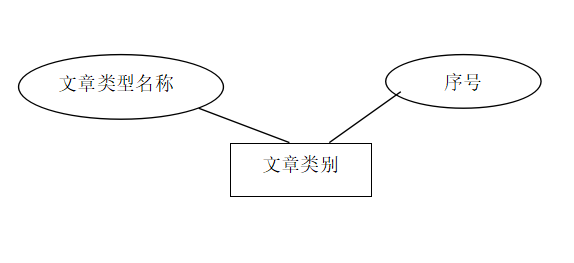


图4-9文章类别E-R图

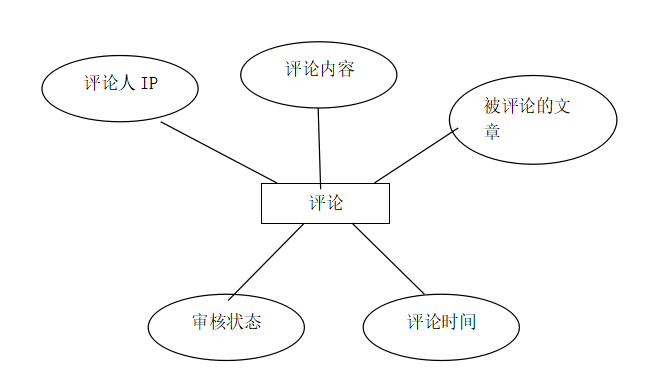


图4-10评论E-R图

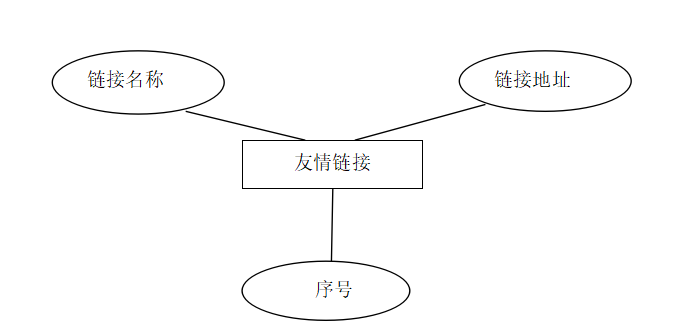


图4-11友情链接E-R图

### 4.3.3 数据库逻辑结构设计

1. 表4-1是数据库表t\_blog，用来存储文章信息
2. 表4-2是数据库表t\_blogger，用来存储博主信息
3. 表4-3是数据库表t\_blogtype，用来存储文章类别信息
4. 表4-4是数据库表t\_comment，用来存储评论信息
5. 表4-5是数据库表t\_link，用来存储友情链接信息

表4-1 文章信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文章信息表 | | | | |
| 字段名 | 数据类型 | 长度 | 说明 | 描述 |
| article\_id | int | 11 | 不能是null，  添加了主键约束 | 文章ID |
| article\_title | varchar | 20 | 可以是null值 | 文章题目 |
| article\_summary | varchar | 400 | 可以是null值 | 文章摘要 |
| article\_date | date |  | 可以是null值 | 发表文章日期 |
| article\_replyhit | int | 11 | 可以是null值 | 文章阅读次数 |
| article\_content | text |  | 可以是null值 | 文章内容 |
| article\_typeid | int | 11 | 可以是null值 | 文章类别ID |

表4-2 管理员信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 管理员信息表 | | | | |
| 字段名 | 数据类型 | 长度 | 说明 | 描述 |
| blogger\_id | int | 11 | 不能是null，  添加了主键约束 | 管理员ID |
| blogger\_name | varchar | 50 | 可以是null值 | 管理员用户名 |
| blogger\_password | varchar | 100 | 可以是null值 | 管理员密码 |
| blogger\_profile | text |  | 可以是null值 | 管理员个人简介 |
| blogger\_sign | varchar | 100 | 可以是null值 | 管理员个性签名 |
| blogger\_image | varchar | 100 | 可以是null值 | 管理员头像 |

表4-3 文章类别信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文章类别信息表 | | | | |
| 字段名 | 数据类型 | 长度 | 说明 | 描述 |
| article\_typeid | int | 11 | 不能是null，  添加了主键约束 | 文章类别ID |
| article\_typename | varchar | 30 | 可以是null值 | 文章类别名称 |
| article\_orderno | int | 11 | 可以是null值 | 文章类别序号 |

表4-4 评论信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 评论信息表 | | | | |
| 字段名 | 数据类型 | 长度 | 说明 | 描述 |
| review\_id | int | 11 | 不能是null，  添加了主键约束 | 评论人ID |
| reviewer\_ip | varchar | 50 | 可以是null值 | 评论人IP |
| article\_id | int | 11 | 可以是null值 | 所评文章ID |
| review\_content | varchar | 1000 | 可以是null值 | 评论内容 |
| review\_date | date |  | 可以是null值 | 评论时间 |
| review\_state | int | 11 | 可以是null值 | 审核评论 |

表4-5 友情链接信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 友情链接信息表 | | | | |
| 字段名 | 数据类型 | 长度 | 说明 | 描述 |
| link\_id | int | 11 | 不能是null，  添加了主键约束 | 链接ID |
| link\_name | varchar | 100 | 可以是null值 | 链接名称 |
| link\_url | varchar | 200 | 可以是null值 | 链接地址 |
| link\_orderno | int | 11 | 可以是null值 | 链接序号 |

## 4.4 系统后台设计

### 4.4.1 管理员登录

管理员在前台点击登录后台，进入后台登录界面，输入正确的用户名和密码后再点击登录就能进入系统后台。

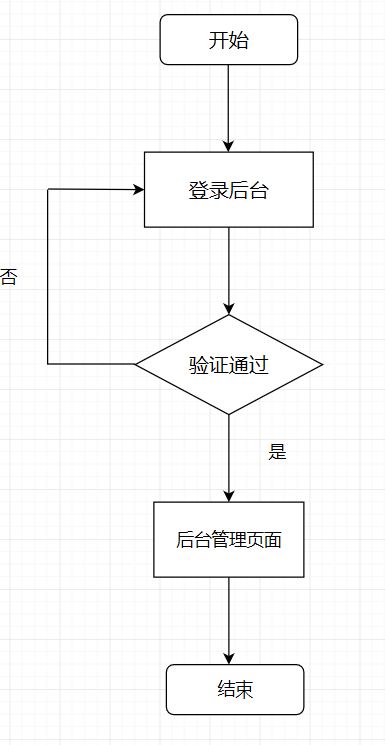


图4-12后台登录模块流程图



图4-13管理员登录后台时序图

### 4.4.2 文章管理

文章管理包括添加文章、查询文章、修改文章、删除文章功能。

添加文章：点击写博客后，进入写文章页面，输入标题、类别、内容，关键字可写可不写，再点击发布博客按钮后。在前台刷新一下，就可以看到添加的文章了。

查询文章：在搜索框中输入文章标题关键字或者完整的标题，再点击搜索，就可以查询到对应文章了。

修改文章：点击博客信息管理，进入博客信息管理页面，选择要修改的文章，再点击修改，然后输入相应文章信息，点击发布博客按钮，就可以成功修改文章了。

删除文章：点击博客信息管理，进入博客信息管理页面，选择要删除的文章，再点击删除，然后点击确定就能成功删除文章了。

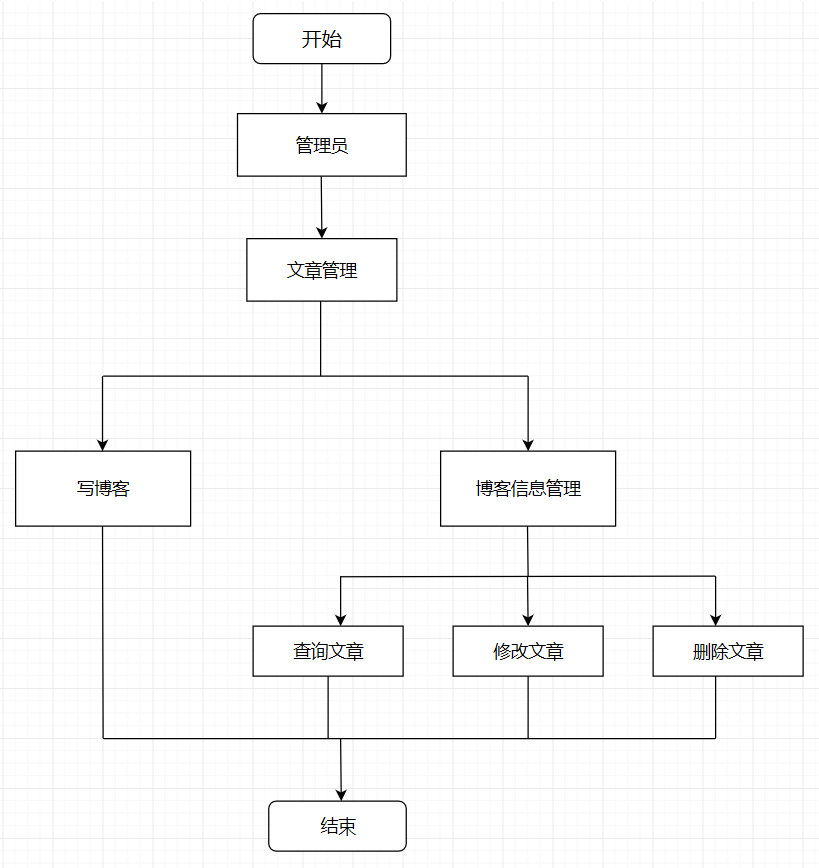


图4-14后台文章管理流程图



图4-15添加文章时序图



图4-16查询文章时序图



图4-17修改文章时序图



图4-18删除文章时序图

### 4.4.3 文章类别管理

文章类别管理包括添加、修改、删除文章类别功能。

添加文章类别：点击博客类别信息管理，进入博客类别信息管理页面，点击添加，再输入类别名称、序号，然后点击确定，就可以成功添加类别了。

修改文章类别：点击博客类别信息管理，进入博客类别信息管理页面，点击修改，再输入类别名称、序号，然后点击确定，就可以成功修改类别了。

删除文章类别：点击博客类别信息管理，进入博客类别信息管理页面，选择要删除的文章类别，接着点击删除，再点击确定，就可以成功删除类别了。但是注意，如果已发表的文章中定义了类别，该类别不能被删除。

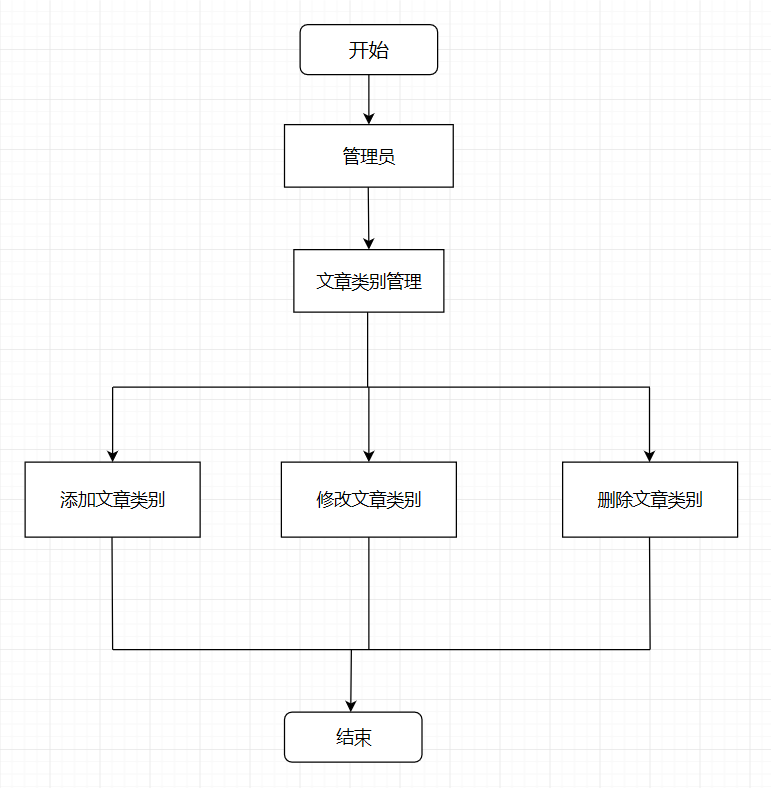


图4-19文章类别管理流程图



图4-20添加文章类别时序图



图4-21修改文章类别时序图



图4-22删除文章类别时序图

### 4.4.4 评论管理

评论管理包括审核评论、删除评论功能。

审核评论：点击评论审核，进入评论审核页面，选择要审核的评论，然后如果点击审核通过，则该评论就能在前台的文章信息界面显示，如果点击了审核不通过，则该评论不能在在前台的文章信息界面显示。

删除评论：点击评论信息管理，进入评论信息管理页面，选择要删除的评论，再点击删除，然后点击确定，就能成功删除评论了。

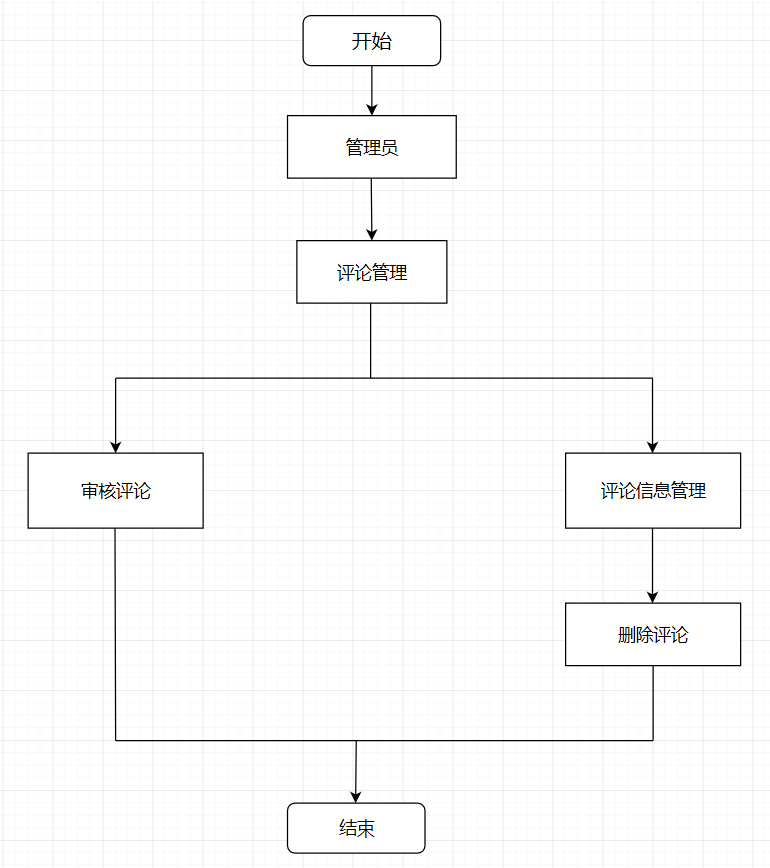


图4-23评论管理流程图



图4-24审核评论时序图



图4-25删除评论时序图

### 4.4.5 个人信息管理

个人信息管理模块包括修改个人信息功能。

修改个人信息：点击修改个人信息，输入用户名、昵称、个性签名、个人简介，再点击提交就成功修改个人信息了。

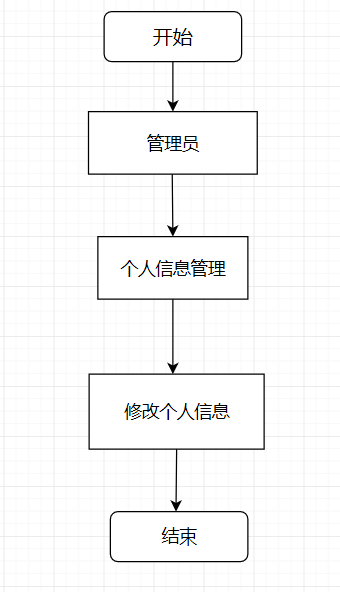


图4-26个人信息管理流程图



图4-27修改个人信息时序图

# 5. 系统实现

## 5.1 系统前台实现

### 5.1.1 首页界面

前台的首页界面，采用了Bootstrap框架的栅格搭建页面布局，采用媒体查询设置响应式布局，用jsp动态生成网页内容。



图5-1首页界面

### 5.1.2 文章信息界面

前台的文章信息界面，采用了Bootstrap框架的栅格搭建页面布局，采用媒体查询设置响应式布局，用jsp动态生成网页内容。

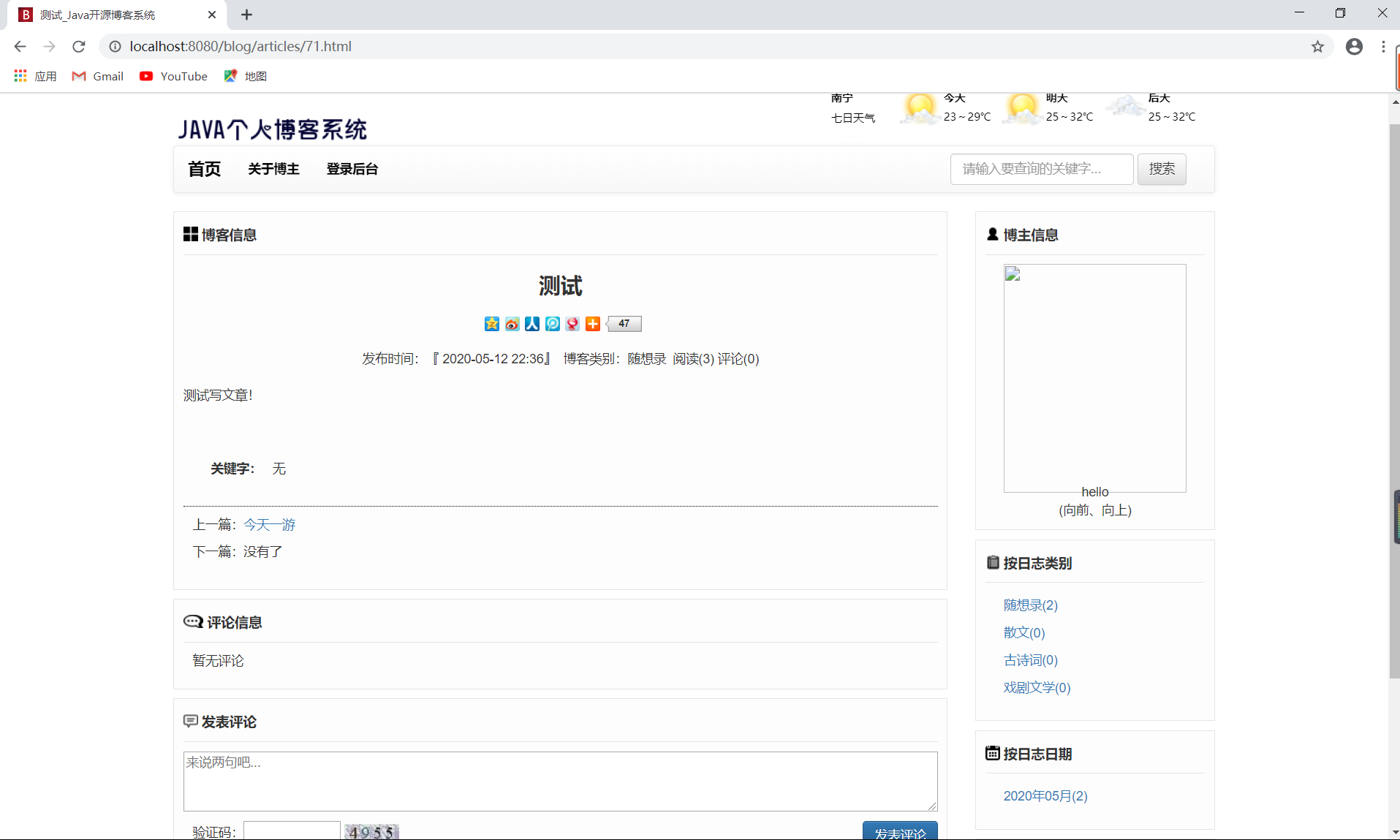


图5-2文章信息界面

### 5.1.3 关于博主界面

前台的关于博主界面，采用了Bootstrap框架的栅格搭建页面布局，采用媒体查询设置响应式布局，用jsp动态生成网页内容。

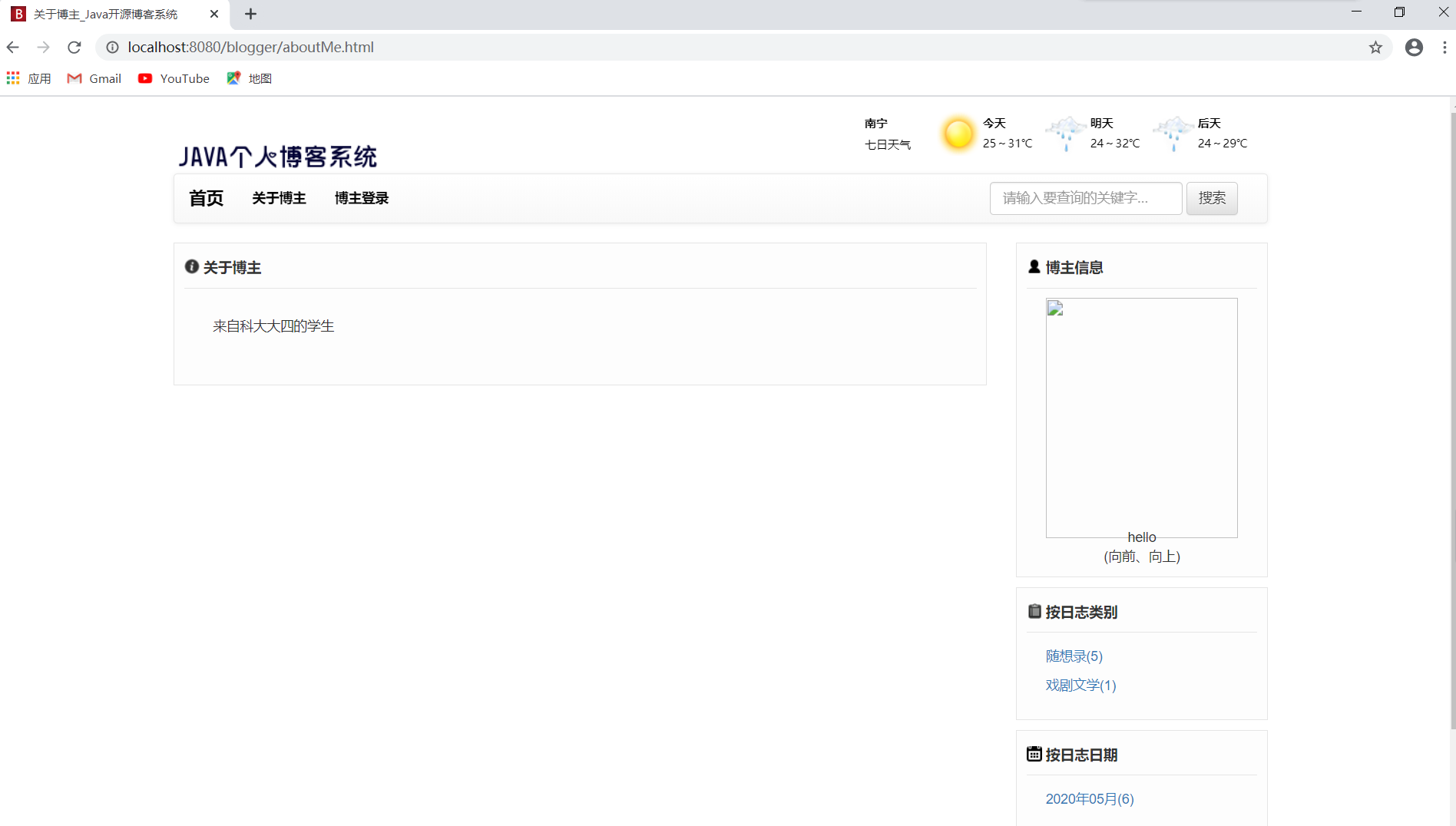


图5-3关于博主界面

## 5.2 系统后台实现

### 5.2.1 管理员登录

管理员在前台点击登录后台，进入登录后台界面，输入用户名、密码。点击登录后，先是通过JavaScript验证输入是否为空，如果用户名为空或两者都为空，会提示用户名不能为空，如果密码为空，会提示密码不能为空；如果输入不为空，jsp表单向服务器提交请求，服务器会接收登录请求，再找到Controller层的相应类，并调用其“登录”方法获取用户名和密码的值，然后检验输入的值是否正确，如果正确，管理员进入后台管理页面，如果不正确，会提示“用户名或密码错误”。

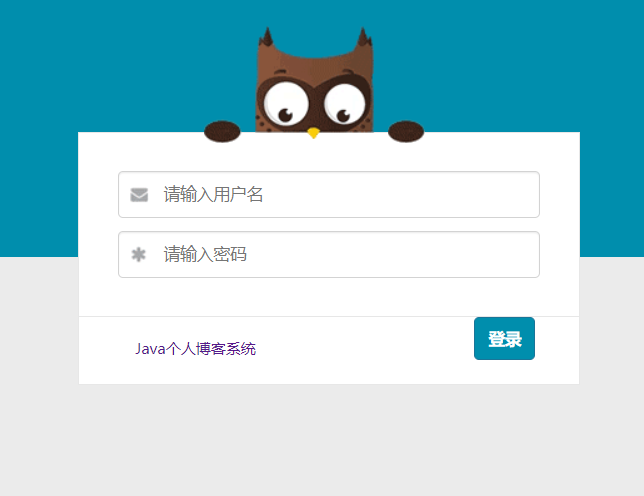


图5-4后台登录界面

### 5.2.2 文章管理

写文章：点击写博客后进入写文章页面，输入标题、类别、内容。点击发布博客后，先是通过JavaScript验证输入是否为空，如果标题、类别、内容都为空，会出现请输入标题的提示信息，如果标题、类别、内容任意一个为空，会出现请输入对应项的提示信息；如果文章信息不为空，jsp表单向服务器提交请求，服务器会接收发布博客请求，再找到Controller层的相应类，并调用其save方法，在该方法中判断文章ID是否存在，如果不存在，调用service层的“添加”方法将文章信息持久化到数据库中并返回添加结果，然后通过对象response获取结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。实现代码如下：

@RequestMapping("/save")

public String save(Blog blog,HttpServletResponse response)throws Exception{

int resultTotal=0;

if(blog.getId()==null){

resultTotal=blogService.add(blog);

blogIndex.addIndex(blog);

}else{

resultTotal=blogService.update(blog);

blogIndex.updateIndex(blog);

}

JSONObject result=new JSONObject();

if(resultTotal>0){

result.put("success", true);

}else{

result.put("success", false);

}

ResponseUtil.write(response, result);

return null;}

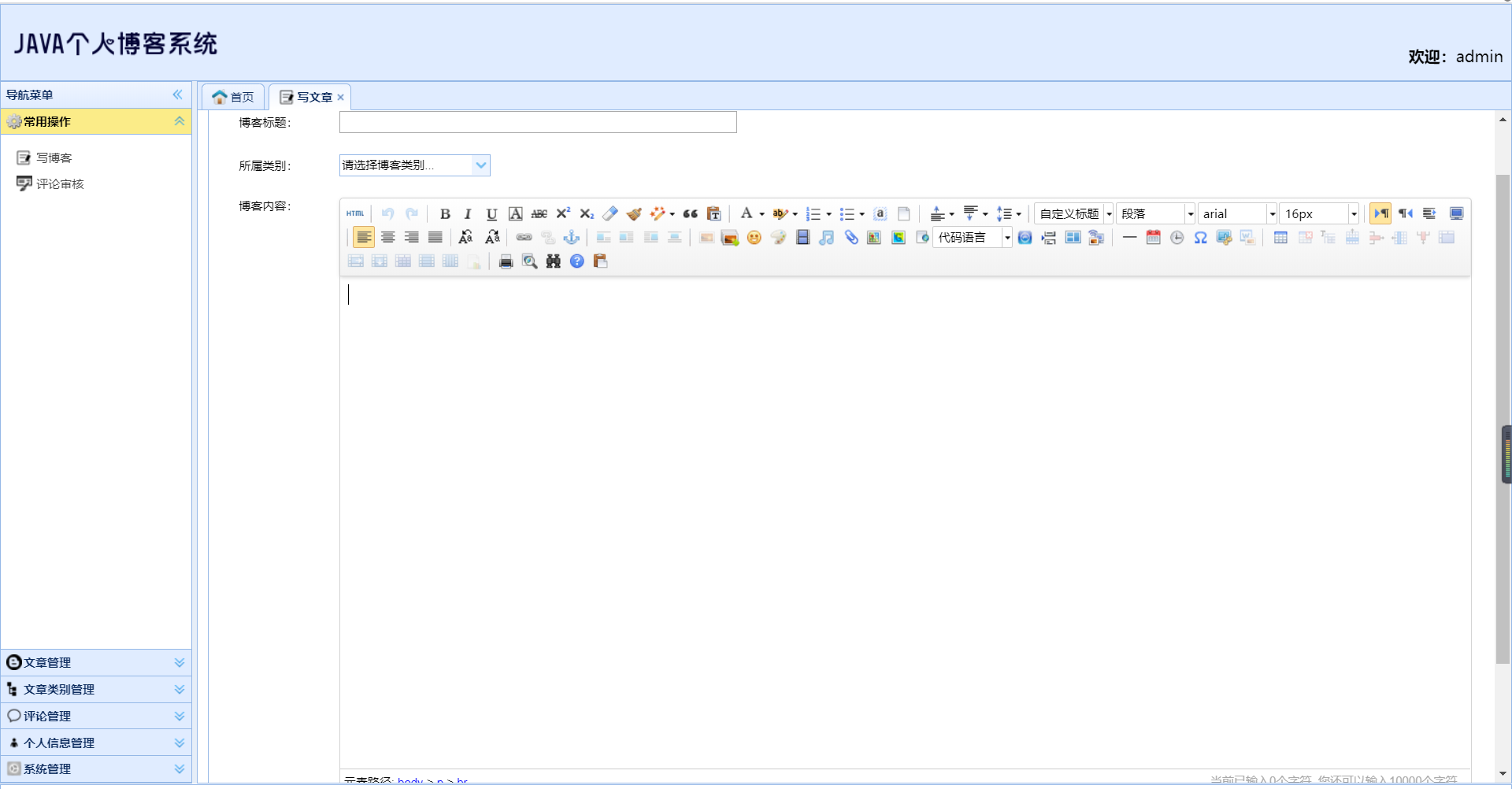


图5-5写文章界面

点击博客信息管理，进入博客信息管理页面：

1. 查询文章：在检索框中输入标题关键字或完整的文章题目。点击搜索后，jsp表单向服务器提交请求，服务器会接收搜索请求，再找到Controller层的相应类，并调用其list方法，该方法返回一个Blog型的list集合，该集合存放了查询的结果，创建一个对应数据类型的集合去接收结果集，然后通过对象response获取结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。
2. 修改文章：勾选要修改的文章，点击修改后，jsp表单向服务器提交请求，服务器会接收修改请求，再找到Controller层的相应类，并调用其save方法，在该方法中判断文章ID是否存在，如果存在，调用service层的“修改”方法将修改后的文章信息持久化到数据库中并返回修改结果，通过对象response获取结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。
3. 删除文章：勾选要删除的文章，既可以单篇删除也可以批量删除，点击删除后，jsp表单向服务器提交请求，服务器会接收删除请求，再找到Controller层的相应类，并调用其“删除”方法，在方法中又调用service层的“删除”方法将结果持久到数据库中并返回删除结果，通过对象response获取结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。

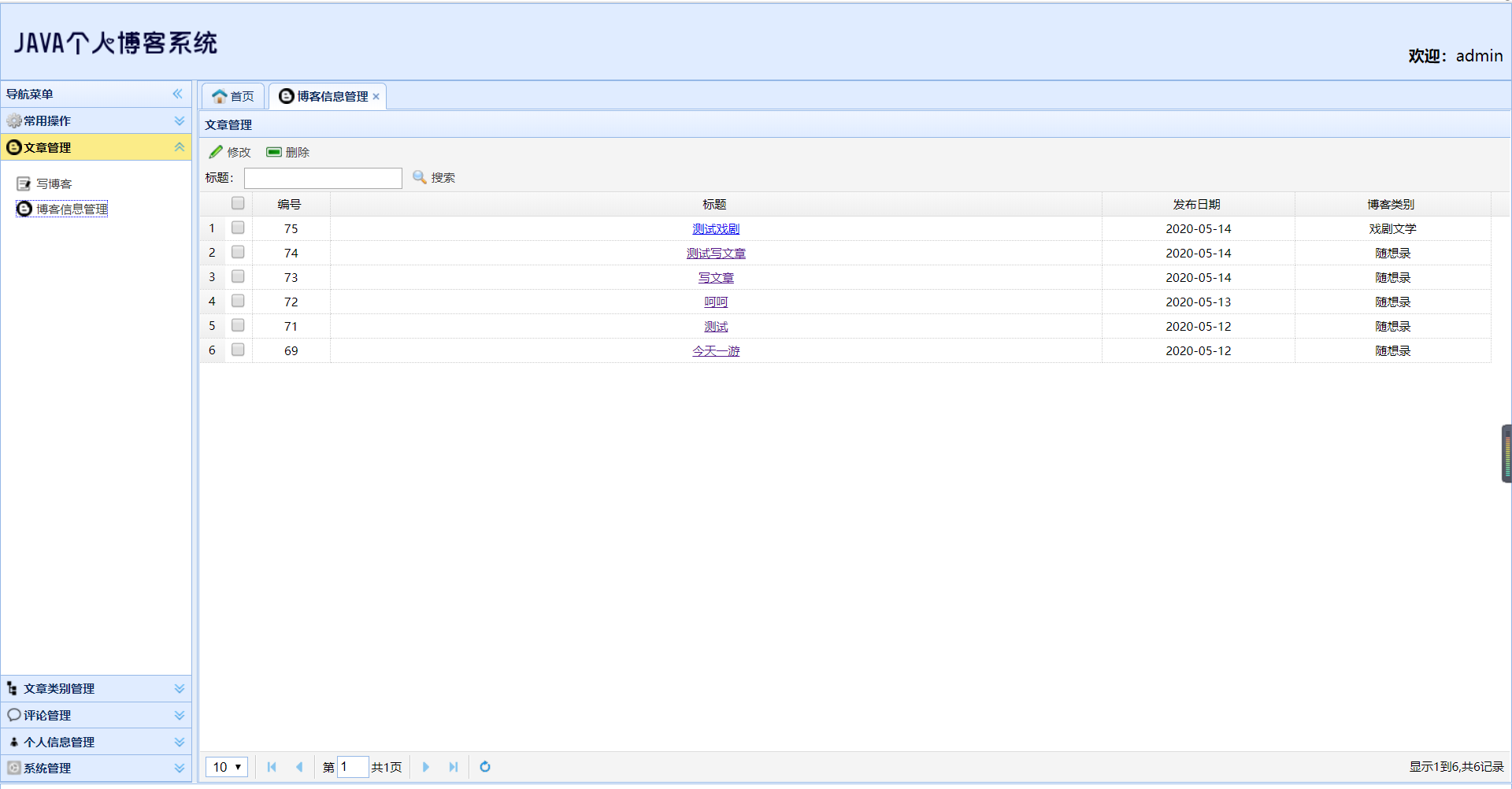


图5-6博客信息管理界面

### 5.2.3 文章类别管理

点击博客类别信息管理，进入博客类别信息管理页面：

1. 添加博客类别：点击添加后，jsp表单向服务器提交请求，服务器会接收添加请求，再找到Controller层的相应类，并调用其“保存”方法，在该方法中判断文章类别ID是否存在，如果不存在，调用service层的“添加”方法将文章类别信息持久化到数据库中并返回添加结果，然后通过对象response获取结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。
2. 修改博客类别：点击修改后，jsp表单向服务器提交请求，服务器会接收修改请求，再找到Controller层的相应类，并调用其“保存”方法，在该方法中判断文章ID是否存在，如果存在，就调用service层的“修改”方法将修改后的文章信息持久到数据库中并返回修改结果，通过对象response获取结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。
3. 删除博客类别：勾选要删除的博客类别，既可以单个删除也可以批量删除，点击删除后，jsp表单向服务器提交请求，服务器会接收删除请求，再找到Controller层的相应类，并调用其“删除”方法，在方法中先判断已发表的文章是否定义了博客类别，如果定义了，则该类别不能删除，否则会调用service层的“删除”方法将结果持久到数据库中并返回删除的结果，通过对象response获取结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。

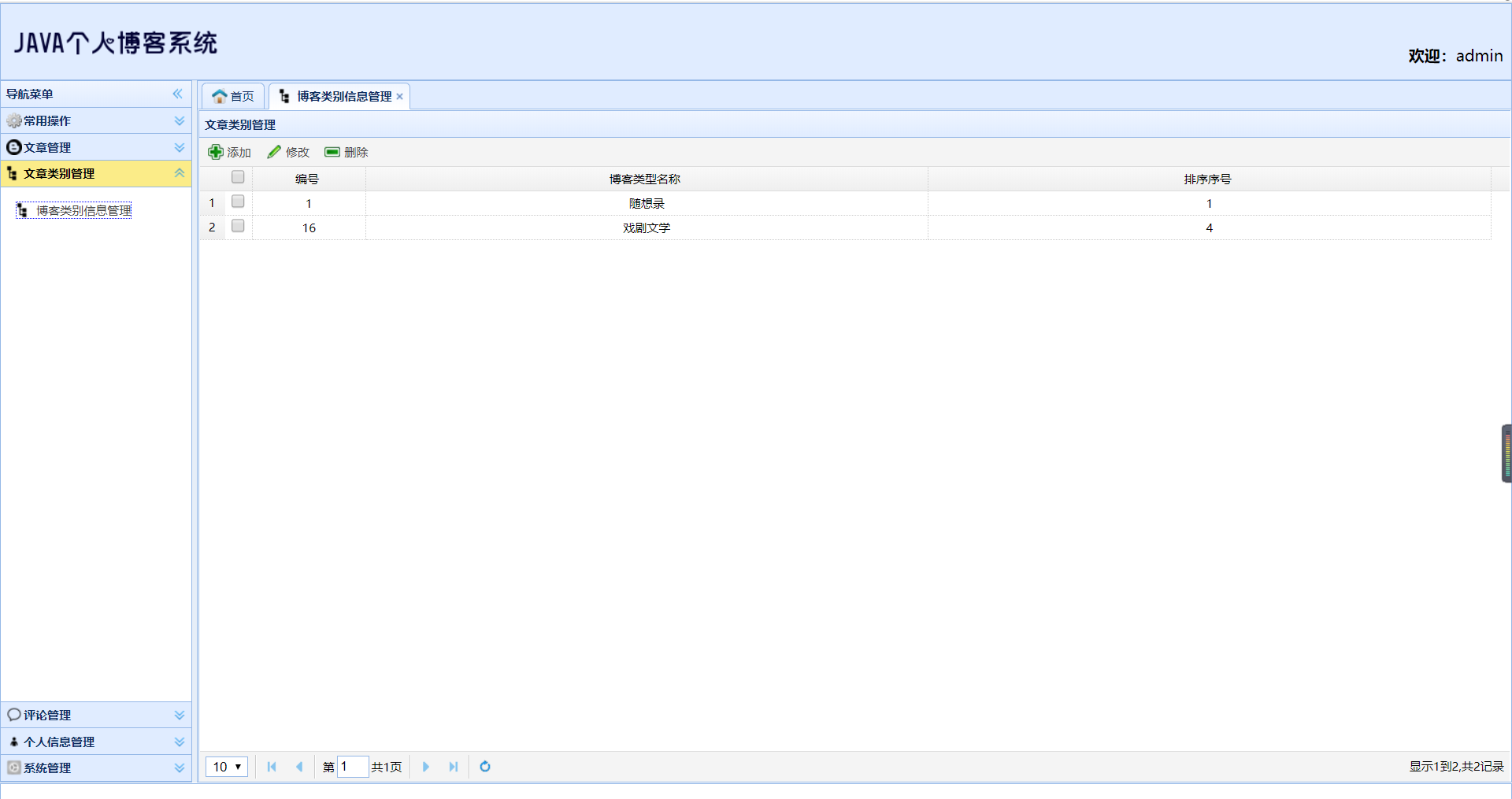


图5-7博客类别信息管理界面

### 5.2.4 评论管理

点击评论审核，进入评论审核页面：

勾选要审核的评论，既可以单条审核也可以批量审核。点击审核按钮后，jsp表单向服务器提交请求，服务器会接收审核通过请求，再找到Controller层的相应类，并调用其“审核”方法，该方法会获取按钮的值，如果按钮的值为“审核通过”，则调用service层的“修改”方法返回评论信息；如果按钮值为“审核不通过”，则调用dao层的“修改”方法持久到数据库。通过对象response获取评论结果，最后服务器会将评论结果响应给浏览器。实现代码如下：

@RequestMapping("/review")

public String review(@RequestParam(value="ids")String ids,@RequestParam(value="state")Integer state,HttpServletResponse response)throws Exception{

String []idsStr=ids.split(",");

for(int i=0;i<idsStr.length;i++){

Comment comment=new Comment();

comment.setState(state);

comment.setId(Integer.parseInt(idsStr[i]));

commentService.update(comment);

}

JSONObject result=new JSONObject();

result.put("success", true);

ResponseUtil.write(response, result);

return null;

}

点击评论信息管理，进入评论信息管理页面：

删除评论：勾选要删除的评论，既可以单条删除也可以批量删除。点击删除后，jsp表单向服务器提交请求，服务器会接收删除请求，再找到Controller层的相应类，并调用其“删除”方法持久到数据库。通过对象response获取删除结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。

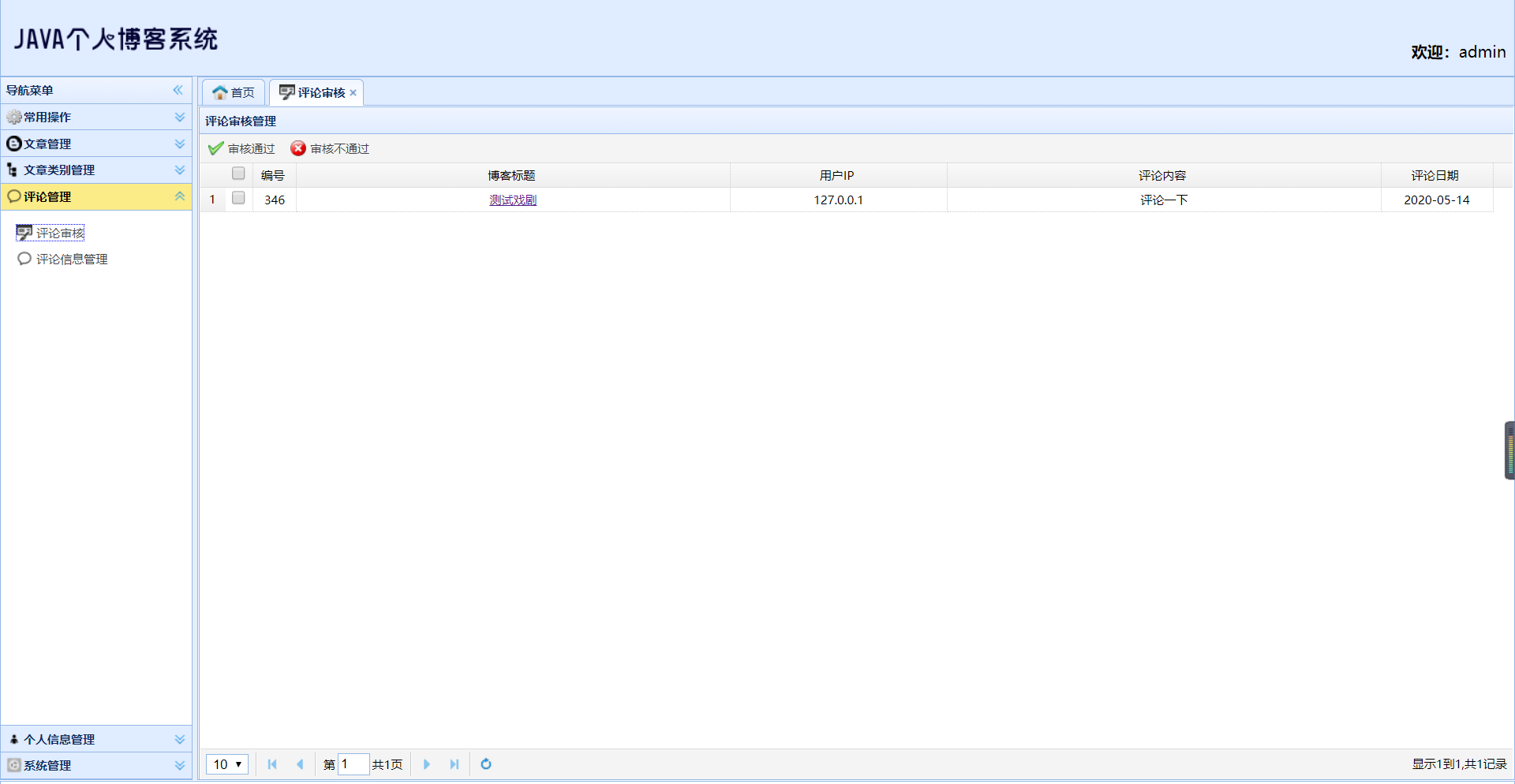


图5-8评论审核界面

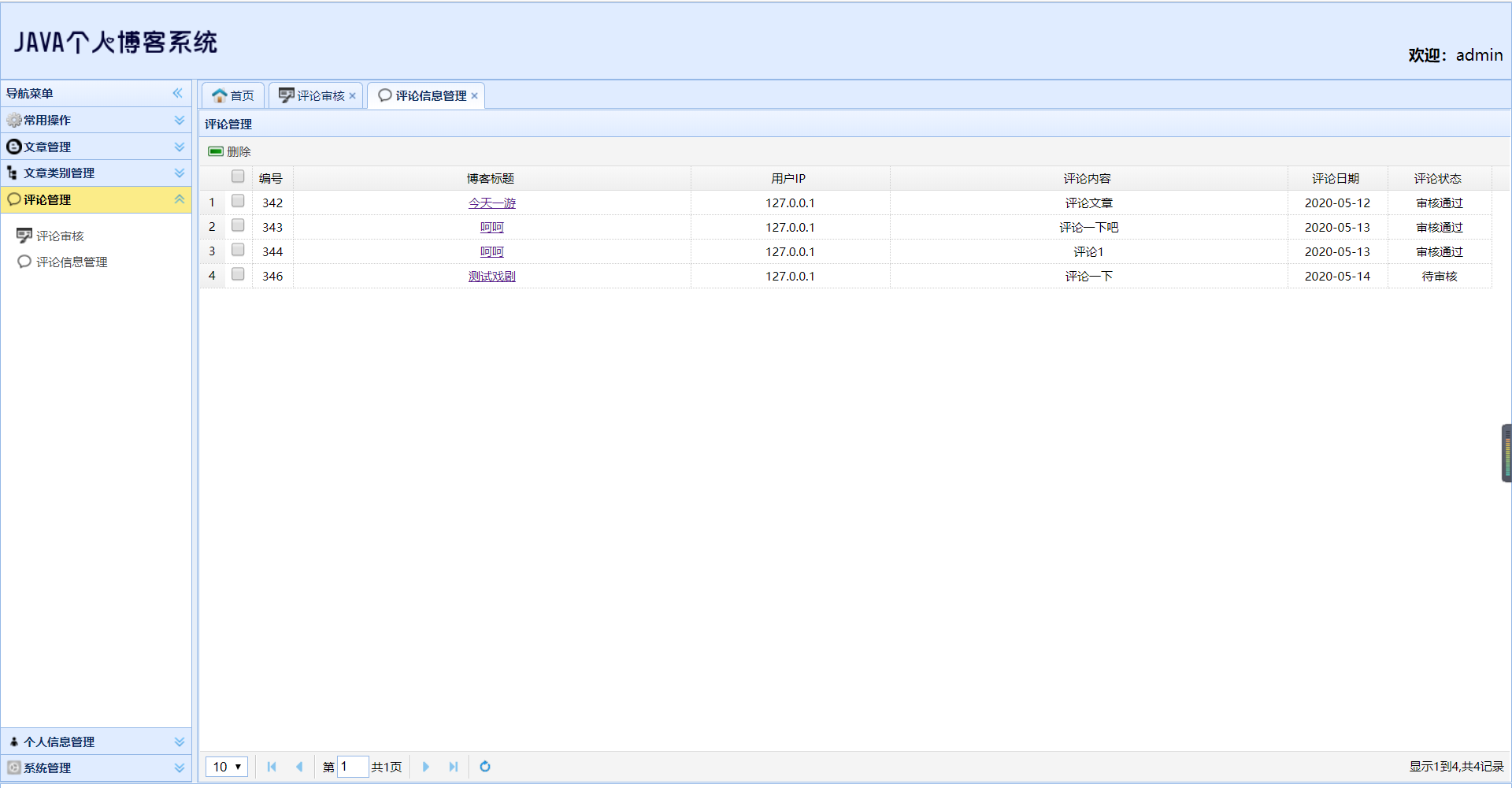


图5-9评论信息管理界面

### 5.2.5 个人信息管理

修改个人信息：点击修改个人信息后，进入修改个人信息页面，输入用户名、昵称、个性签名、个人简介信息。先通过JavaScript判断输入是否为空，如果为空，点击提交后，会提示请输入对应项的提示信息。如果输入都不为空，点击提交后，jsp表单向服务器提交请求，服务器会接收修改请求，再找到Controller层的相应类，并调用其“保存”方法将修改后的个人信息持久到数据库中并返回修改结果，通过对象response获取结果，最后服务器会找到相应的视图，由该视图负责将结果响应给浏览器。

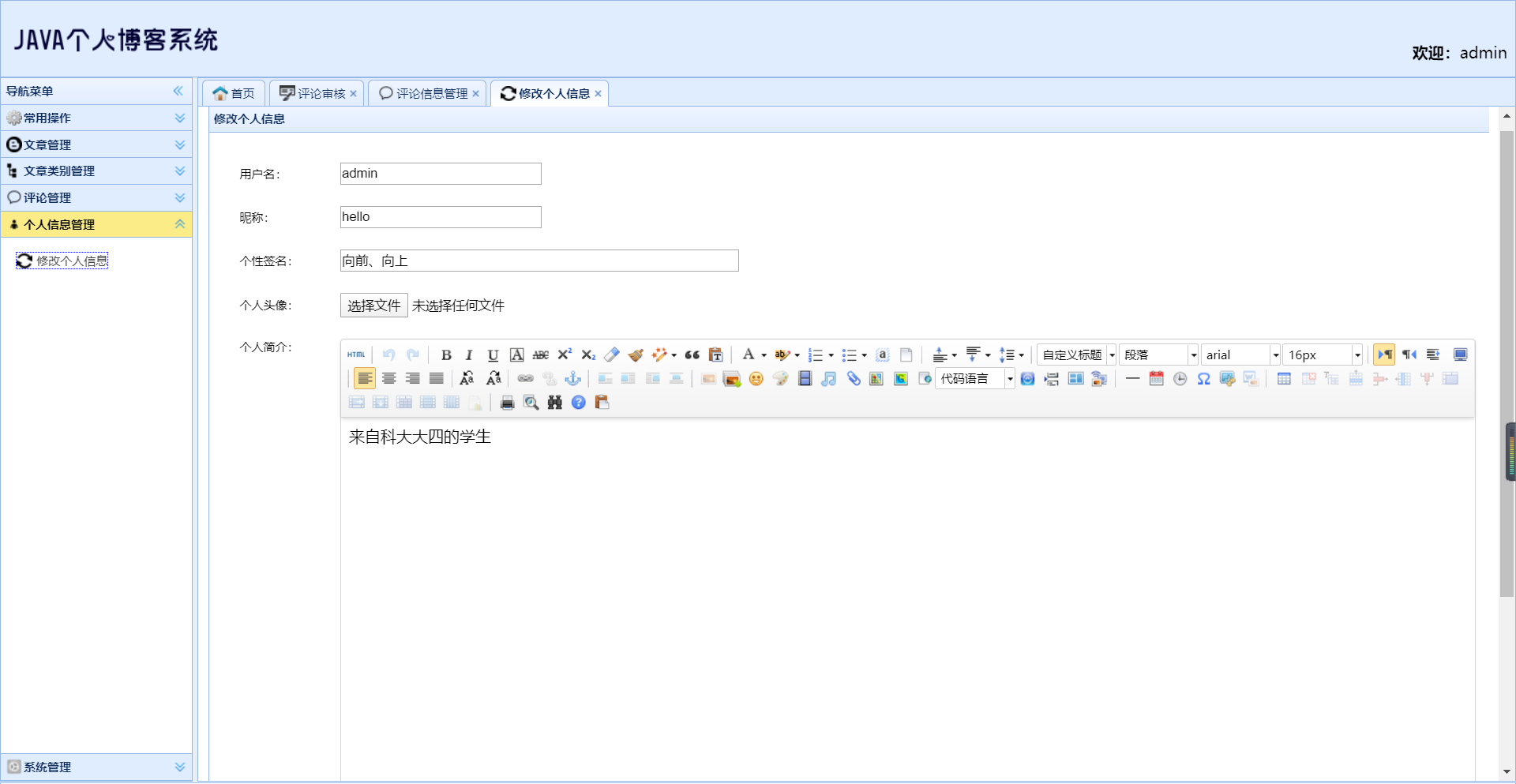


图5-10修改个人信息界面

# 系统测试

## 6.1 测试前台

### 6.1.1 前台发表评论

用户点击点击文章题目或日期进入文章信息界面，在文章信息界面中可以发表评论，输入评论内容后，再输入正确的验证码，然后点击发表评论按钮就能成功提交评论了。待管理员审核通过后才能在文章信息界面中显示。

表6-1前台发表评论测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 登录前台测试用例表 | | | | |
| 测试方法 | 测试内容 | 测试目的 | 期望结果 | 测试结果 |
| 点击文章题目或日期进入文章信息界面 | 用户在前台发表评论 | 用户是否能评论文章 | 用户能进行评论 | 用户能进行评论 |



图6-1前台发表评论测试图

### 6.1.2 查询文章

用户在搜索框中输入相应关键字，就能查询到相应文章

表6-2前台查询文章测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 前台查询文章测试用例表 | | | | |
| 测试方法 | 测试内容 | 测试目的 | 期望结果 | 测试结果 |
| 在搜索框中输入  “文章” | 用户在前台查询博客 | 按条件查询文章功能是否实现 | 显示含有“文章”的博客 | 能查询到符合条件的博客 |



图6-2前台查询文章测试图

## 6.2 测试后台

### 6.2.1 后台登录

在前台首页可以点击登录后台，然后输入用户名：admin，输入密码: 123456就能够进入后台系统，用户名和密码都要输入以及输入正确，否则系统会提示错误信息。

表6-3登录后台测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 登录后台测试用例表 | | | | |
| 测试方法 | 测试内容 | 测试目的 | 期望结果 | 测试结果 |
| 用户名、密码都不输入或者只输入密码，点击登录 | 用户登录后台 | 是否能验证后台登录信息 | 提示用户名不能为空 | 提示用户名不能为空 |
| 只输入用户名 | 提示密码  不能为空 | 提示密码  不能为空 |
| 输入除admin以外的用户名或除123456以外的密码 | 提示用户名或密码错误 | 提示用户名或密码错误 |
| 输入用户名admin、密码123456 | 进入后台  页面 | 进入后台  页面 |

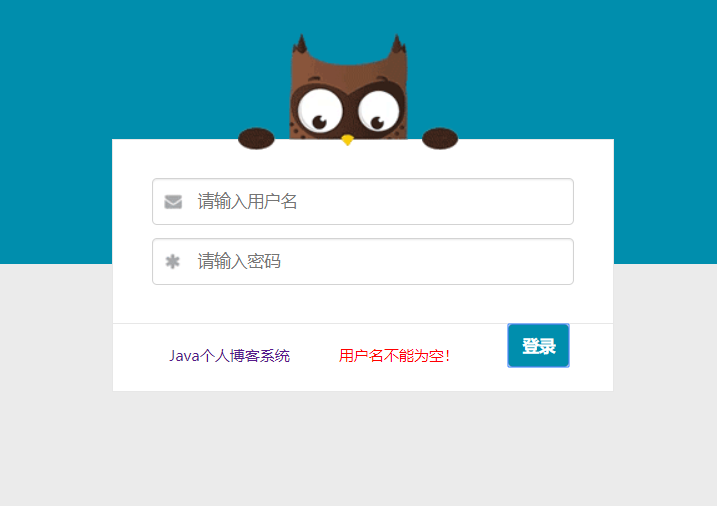


图6-3后台登录测试图1

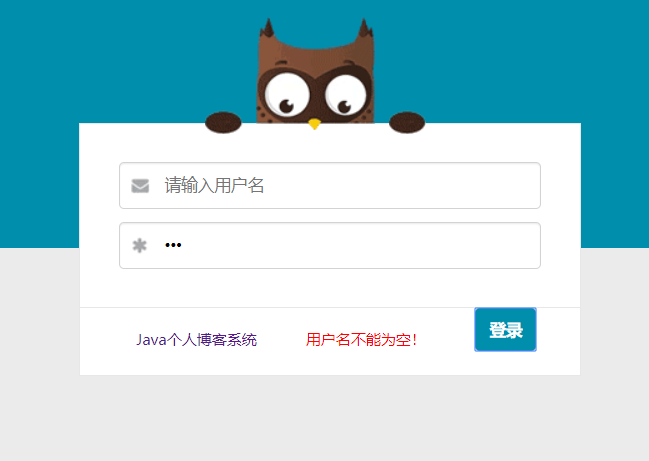


图6-4后台登录测试图2

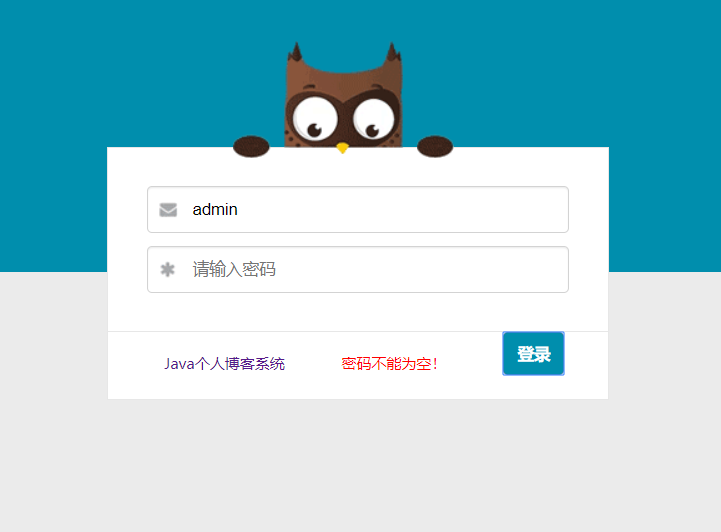


图6-5后台登录测试图3

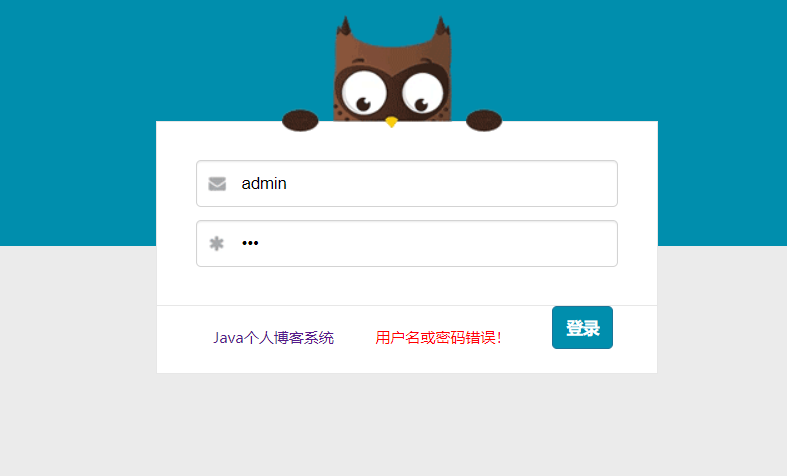


图6-6后台登录测试图4

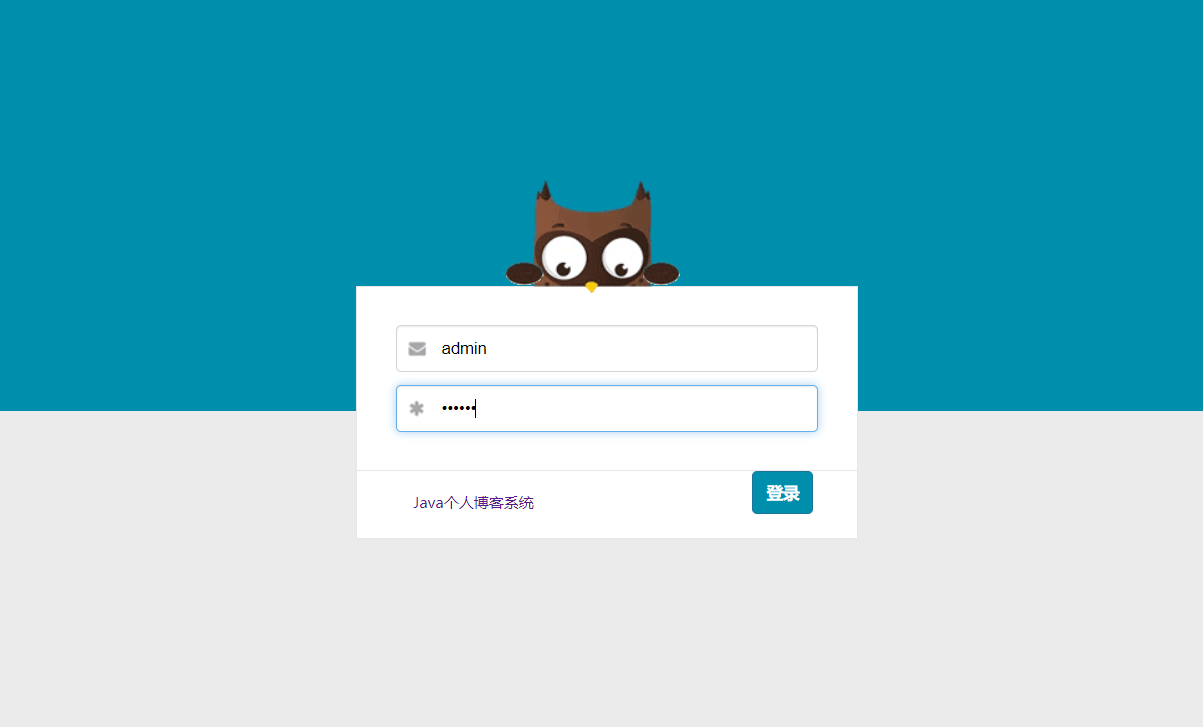


图6-7后台登录测试图5

### 6.2.2 发表文章

点击文章管理下的写博客，进入写博客页面，输入文章信息，再点击发布博客就能发表文章了，在前台的首页会显示新文章。

表6-4后台发表文章测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 后台发表文章测试用例表 | | | | |
| 测试方法 | 测试内容 | 测试目的 | 期望结果 | 测试结果 |
| 在写博客页面输入相应内容之后点击发布博客按钮 | 博主写博客 | 发布博客功能是否实现 | 前台可以看到新添加的文章 | 前台可以看到新添加的文章 |

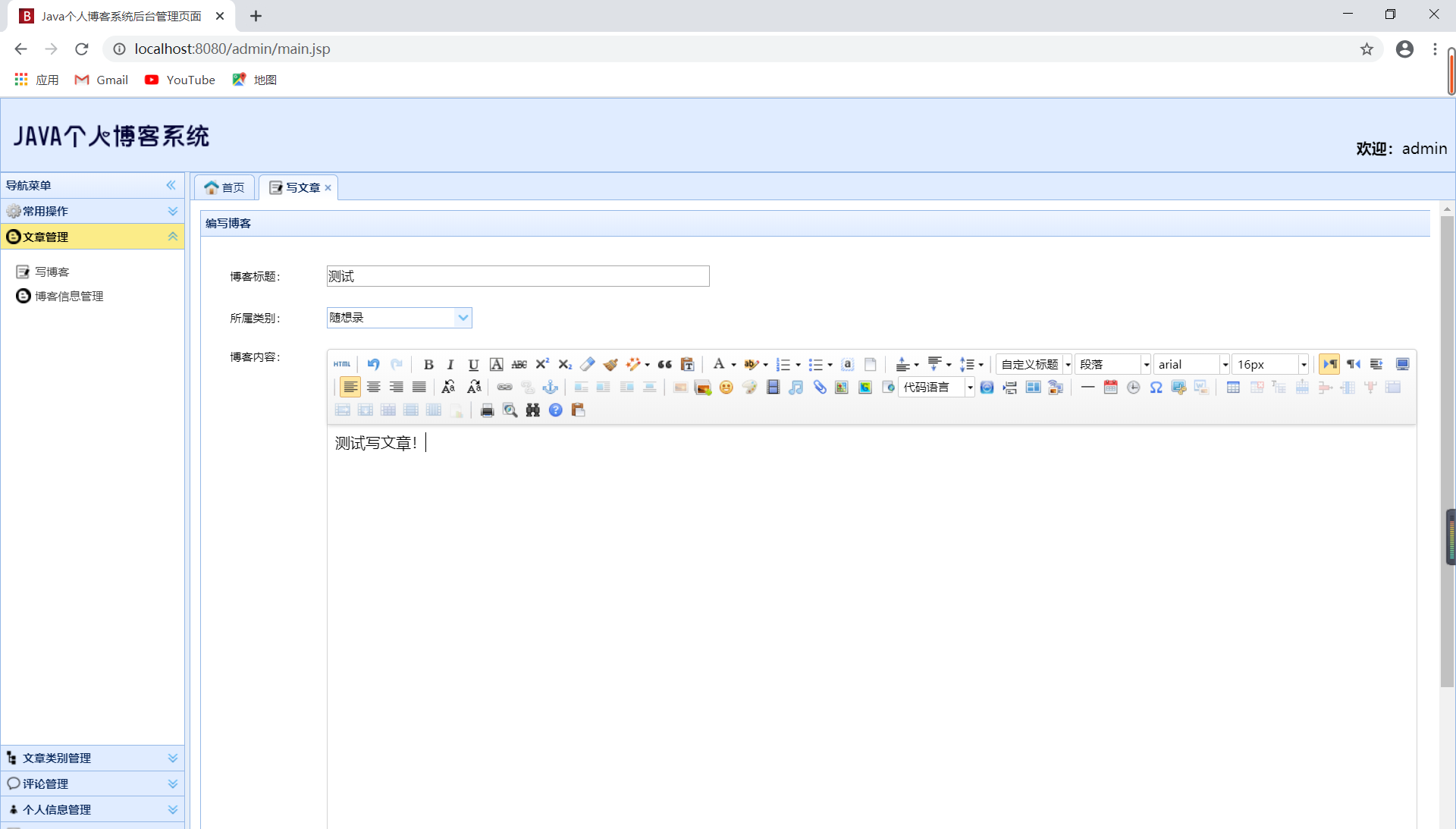


图6-8后台写文章测试图1

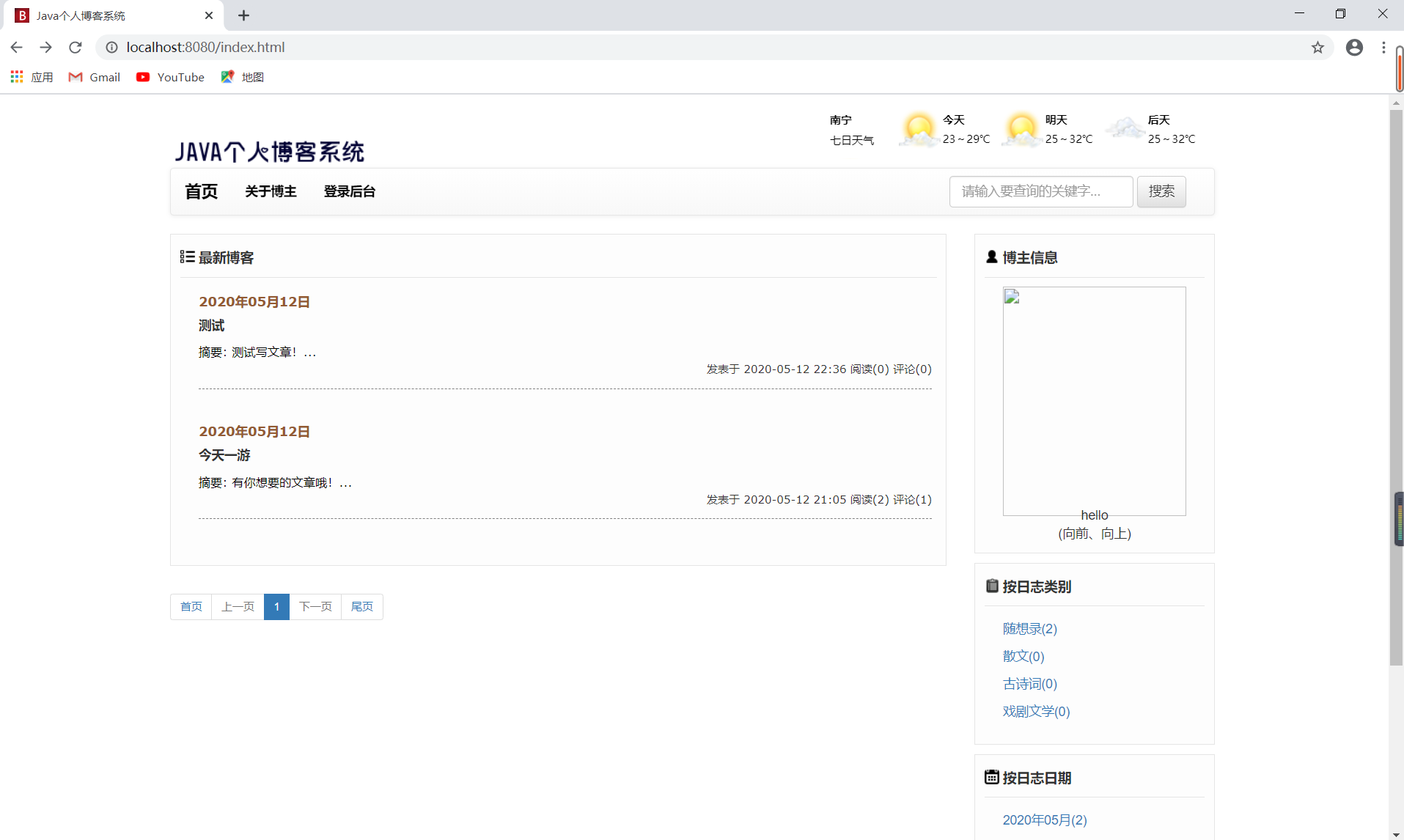


图6-9后台写文章测试图2

### 6.2.3 后台审核评论

点击评论管理下的评论审核，先是选择要审核的评论，再点击审核通过，则该评论在前台的文章信息界面能显示，否则不能显示。

表6-5后台审核评论测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 后台审核评论测试用例表 | | | | |
| 测试方法 | 测试内容 | 测试目的 | 期望结果 | 测试结果 |
| 勾选要审核的评论，然后点击审核通过 | 管理员后台审核评论 | 管理员是否能审核评论 | 文章信息界面可以看到审核通过的评论内容 | 文章信息界面可以看到审核通过的评论内容 |
| 勾选要审核的评论，然后点击审核不通过 | 文章信息界面没有显示审核不通过的评论内容 | 文章信息界面没有显示审核不通过的评论内容 |

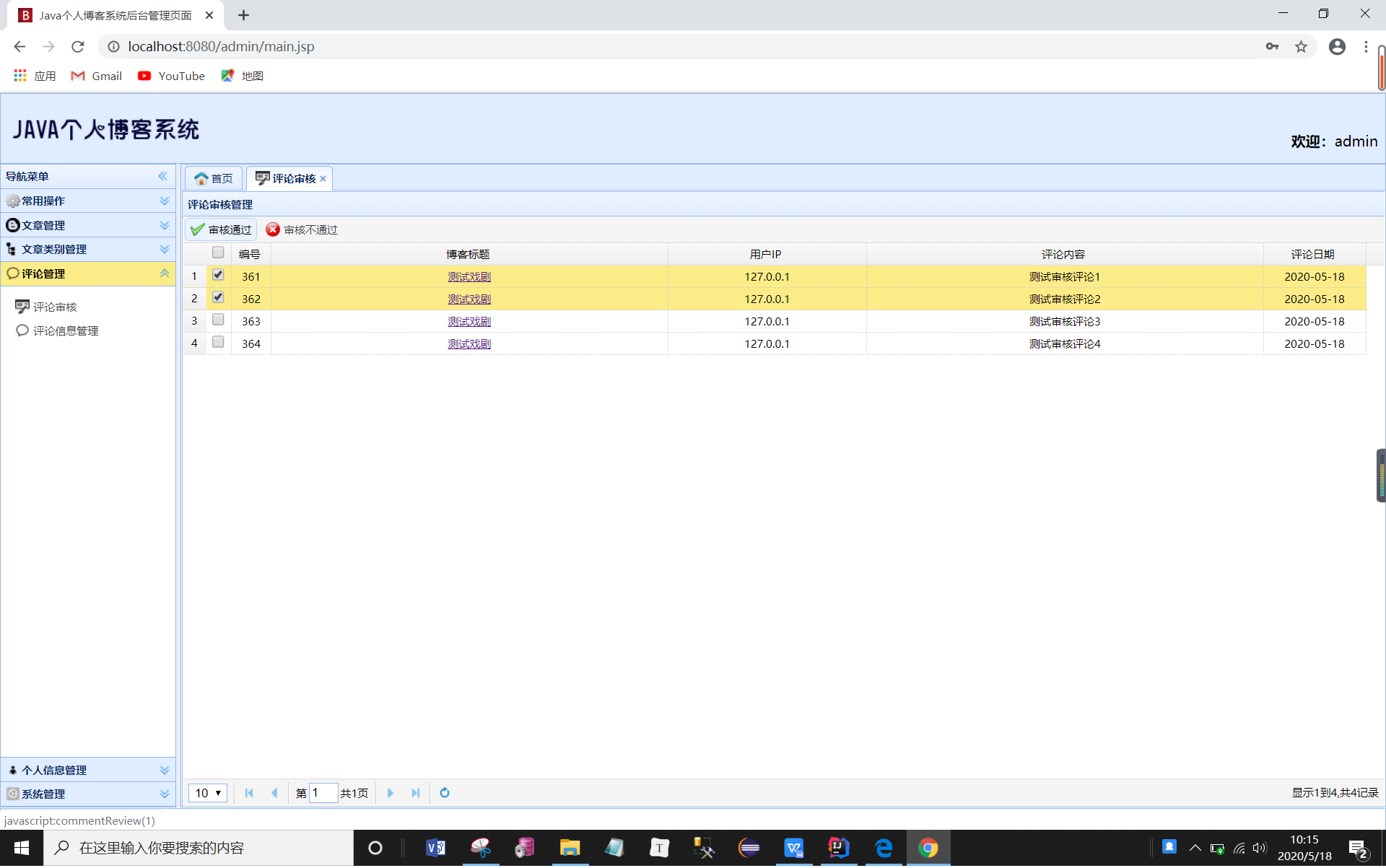


图6-10后台审核评论测试图1

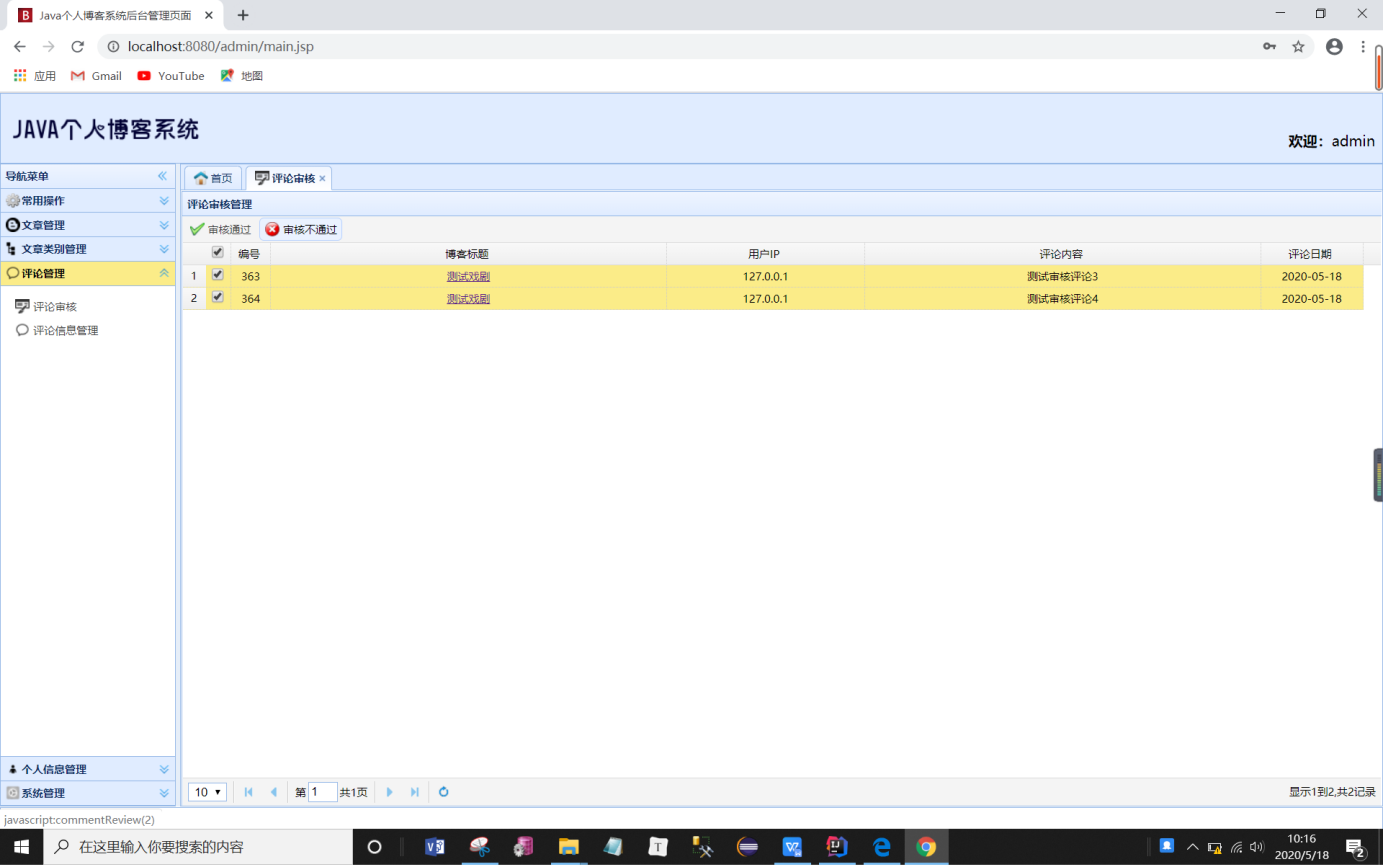


图6-11后台审核评论测试图2



图6-12后台审核评论测试图3

# 总结与展望

## 7.1 总结

本文主要论述了一个基于SSM的个人博客系统的设计与开发过程。通过设计改博客系统，我深入了解并掌握了SSM框架的原理、强大之处和应用。

从选题之初，老师就给予了很多指导，给我们提供了很多文档资料，让我们的选题有了明确的方向。在选题方面，我先是调查了当今时代人们在哪一方面需求量是比较多的，再结合自身的技术能力和学习能力，最终确定了设计和实现一个个人博客系统。我查到使用量居高不下的平台中包括个人博客系统，于是对个人博客系统的需求做了仔细分析，将个人博客系统的功能做了汇总，然后根据自己的能力筛选功能。个人博客系统最大的优点是满足了人们展示自我、表达自我的个性化需求，而我自身也有对自我表达平台的渴望，所以对于完成此课题的设计和实现，我是很有动力的，接着就积极查询相关论文资料，对个人博客技术的实现做了总结，从而确定了个人博客系统的功能以及实现系统的技术、开发工具。

本博客系统采用了SSM框架搭建项目，其实一开始，我对这个框架的应用掌握得并不是很深。虽然在之前的小组项目了解过，但是只做项目的一部分而已，对SSM整体并没有太深的了解，所以要把整个SSM应用起来，不免有些吃力。不过好在用MVC模式做过雇员系统，也有一定的项目基础。但是做这个个人博客系统时，也总是会出现小细节的错误，比如字母遗漏，或是字母大小写写错，所以通过此次大实操，编写代码的速度和正确率有效提高了。写一些业务逻辑的代码时，逻辑思维不够全面，以致于不能实现某个功能，就去请教了老师和同学，然后对比自己和老师、同学思路的差异，才明白自己思路出错在哪，所以编程能力有了一定的提升，这也为我以后的工作奠定了一定的技术基础。此个人博客系统的完成，并不只是完全靠自己，还有老师和同学的助力，至此，我很感激老师和同学的指导和帮助。

这次毕业设计，从选题、相关文档的编写、系统的代码实现、论文的撰写至完成，整整经历了5个月，虽然付出了很多的精力，但是也有了不少的收获。通过这次毕业设计，我重新认识了软件开发这个行业，它对程序员的逻辑思维能力的要求很高，而逻辑思维能力的加强又依赖于你的代码实操能力。以前总是听学长学姐说，代码要多敲才能学好软件工程这门课，但前提是自己应该要能完全理解代码的逻辑，再自己敲出代码，然后要反复敲代码，这样才能真正有效提高软件开发能力，对此，我现在真的深有体会。除此之外，还提高了文档写作的能力，也更加深入巩固了专业知识、加强了专业技术能力，理论和实操能力兼得，对专业知识理解更透彻了，对SSM开发模式的理解更深入了，逻辑思维能力也有了一定程度的提升，编写代码的速度也有了一定的提高。

## 7.2 展望

虽然实现了个人博客系统，在个性化方面有一定的体现，但是也只是实现了基本的功能，相对于市场上的个人博客系统，界面过于简单、功能也不够完善齐全、不能够完全满足用户的需求。但是也正是因为这样，我更加要深入学习专业知识，培养专业技术，不仅仅是为设计功能更多、界面体验更友好的软件，更是想要全面充实自己的知识和技能，增强自信心，实现自我价值。通过这次毕业设计，全面激起了我对软件专业探索和学习的兴趣，确定了自己的技术所长和发展方向。

在软件这一行业，技术迭代和更新速度很快，所以我要更加自觉学习更多新知识、新技术，以适应时代的需要，为自己的长远发展建立扎实稳固的基础，为人们设计出更有效提高工作效率，丰富人们娱乐性活动和有助于学习、充实生活的软件。软件开发技术有一定的门道，想要熟练掌握需要一个循序渐进的过程，因为这门行业不仅技术更新快，而且逻辑性强，要学习的知识会比较综合，所以基础方面的掌握是非常重要的，最基础的无非是编写代码的速度、熟练程度。所以，在答辩之后在毕业之前，在这段空档中，要将自己的实操能力提高上一个层次，同时注重思维逻辑能力的培养。

今后，我会将重心放在前端开发方面，因为用户对一个软件的印象和评价通常都是先从界面开始的。如果一个软件的界面整洁、美观、版块结构清晰，用户的观感舒服，便会激发用户的使用欲望。当然，也不会完全忽视后台的重要性，对于后台方面，目标是掌握主流的软件开发模式，虽然界面是软件的门面，但是如果没有实用、丰富的功能，软件的实用性价值也还是会大打折扣。界面体验友好、功能齐全是一个软件的基本优点，也应该是每个程序员要努力的方向，也更是我自己的目标。

# 致谢

四年的大学时光一晃而过，在这四年里，时常经历迷茫的时段，不知道自己未来会如何，不确定自己将要往哪方面发展。此次的毕业设计，在很大程度上解答了我四年的迷惑，让我进一步了解了我的学习能力和实操能力，因此确定了自己的发展方向。

首先，要感谢我的指导老师，在选题和论文撰写、技术方面给予了我很多建设性的意见。老师非常有责任心、有耐心，对我们很上心，及时督促我们的毕业设计进度，认真察看我们提交的文档资料，认真检查每个细节点，从格式到内容，细心指导我修改论文的方向，在内容方面如何拓展方面予以重要意见，及时帮助我们更正错误。老师平时工作很忙，但是每向其提出疑问时，老师都会在当天解答疑问，并给出建议，这很大程度地提高了我做毕业设计的进度。

其次，要感谢我的朋友，他们有的已经工作了，自己每天已经有不小的工作量了，但是每当我碰到技术问题时，还是不耐其烦地给我讲解、演示。有的朋友在前端技术方面给了我一些宝贵的意见，有的朋友还找了一些她们公司的已经上线了的类似我毕业设计的项目给我做参考。所以，朋友的技术帮助和信心鼓励、倾心支持也是我能实现个人博客系统功能的一大原因。

以上，就是我要感谢的人，没有他们答疑解惑，我的毕业设计之路必然非常艰辛，有了他们的帮助，我才能按时完成个人博客系统的设计与实现。

# 参考文献

1. 储雷,张健.基于JavaEE的个人博客系统的设计与实现[J].电脑知识与技术，2016(12):117-120
2. 季家健，刘琳岚.基于 Java EE 的个人博客系统[J].信息通信，2017(8):114-116
3. 刘磊.基于Web框架的博客管理系统设计与实现[J].计算机时代，2017(2):22-23
4. [Joosse, Sofie，](https://www.zhangqiaokeyan.com/search.html?doctypes=1_2_3_4_5_6_7_1-0_9_4-0&sertext=Joosse, Sofie&option=202)[Brydges, Taylor.](https://www.zhangqiaokeyan.com/search.html?doctypes=1_2_3_4_5_6_7_1-0_9_4-0&sertext=Brydges, Taylor&option=202)Blogging for Sustainability: The Intermediary Role of Personal Green Blogs in Promoting Sustainability[J].[environmental communication](https://www.zhangqiaokeyan.com/academic-journal-foreign_environmental-communication_info_3180_1/)，2018(12):686
5. [Hoffmann, Arvid,](https://www.zhangqiaokeyan.com/search.html?doctypes=1_2_3_4_5_6_7_1-0_9_4-0&sertext=Hoffmann, Arvid&option=202)[Otteby, Katarina.](https://www.zhangqiaokeyan.com/search.html?doctypes=1_2_3_4_5_6_7_1-0_9_4-0&sertext=Otteby, Katarina&option=202)Personal finance blogs: Helpful tool for consumers with low financial literacy or preaching to the choir[J].[International Journal of Consumer Studies](https://www.zhangqiaokeyan.com/academic-journal-foreign_international-journal-consumer-studies_info_543_1/)，2018(2):241
6. 苏海燕.互联网背景下视频博客发展对策研究[J].新闻研究导刊，2019(9):188-189
7. 陈沛捷.个人博客系统设计与实现[D].天津大学软件学院，2017
8. 李阳.SSM框架在Web应用开发中的设计与实现[J].计算机技术与发展，2016(12):191
9. 杨帆,林勇,胡秀兵.基于MVC模式个人博客系统的设计与实现[J].计算机时代, 2014(1):23-26
10. 余思源,张伟.基于JAVA的个人博客系统的设计与实现[J].电脑知识与技术，2018(14):129