电 子 科 技 大 学 成 都 学 院

CHENGDU COLLEGE OF

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

本科毕业综合考核项目报告



论文题目 基于J2EE的资源分享平台

学 号 1540610320

作 者 姓 名 周晔

指 导 教 师 王晓玲

# 摘要

网络中的资源既是多样化的，又具有唯一性。从而促使人们乐于分享资源，以供其他人获取该资源，如此便形成了共享资源。在如今，共享已经成为一种主体行为，为此我们组选题为基于J2EE的资源分享平台。

本系统与后台的数据交换的技术是jQuery和Ajax，后台界面的主要框架为frameset框架，对web端界面进行动态实现；在手机端主要用来java语言和xml知识，主要用的编辑器是Android studio，其中使用了Material Design设计语言作为应用界面设计准则，使用OkHttp向服务器发送请求，通过JSONObject解析JSON数据，对手机端的界面进行动态实现；在服务器端主要使用了jsp语言，主要用到的编辑器是Intellij IDEA，B/S架构的应用采用JavaWeb的技术来实现，对于具体的开发框架，我们选用的是应用比较广泛和成熟的SSH（SpringMVC、Spring、Hibernate整合框架来完成后台服务端的开发，SpringMVC用于控制请求处理和分发,Spring用于依赖注入,协助完成服务层开发。对web端和手机端的动态实现提供接口；主要的数据库是MySQL，对整个系统的数据看进行管理；主要的浏览器是谷歌的Chrome浏览器，为系统提供测试便利。

本系统主要功能有博客的上传、下载、浏览和评论；把用户的下载内容整合为找资源、我的资源、已下载和我的收藏；便签的分享主要是用户发布动态、图片，还可以看到其他用户的全部和热门的相关内容；个人中心是对当前用户的相关信息整合而成的页面，由此明确分类对系统进行管理，增强用户体验。这些都是在本文中做出了详细的说明，展示了本系统的功能和特色。

关键词：在线，资源，分享，博客，手机端，web端，服务器端

# ABSTRACT

The resources in the network are both diversified and unique. This encourages people to share resources so that others can get access to them, thus creating shared resources. In today's world, sharing has become a principal activity. For this purpose, our group chose the topic of online resource sharing system based on JSP.

This paper mainly introduces the process of requirements analysis, general design, detailed design, system implementation and system test of an online resource sharing system for various users.In the process of web front-end development, the system USES HTML+CSS+JavaScript technology for editing. The main editor is hbulider, in which jQuery and Ajax are used to realize data exchange with the background. Material Design Design language is used as the application interface Design guideline. OkHttp is used to send requests to the server, and JSONObject is used to analyze JSON data to realize the mobile phone interface dynamically. On the server side is mainly used JSP language, the main use of editor is Intellij IDEA, the application of B/S structure using JavaWeb technology, for a specific development framework, we choose is applied more widely and mature SSH (for SpringMVC, Spring, Hibernate) integration framework to complete the background server development, for SpringMVC is used to control the request processing and distribution.

The main functions of this system include blog upload, download, browse and comment. you can also see all relevant and popular content of other users. A personal center is a page that integrates relevant information about current users, so as to clearly classify and manage the system and enhance the user experience.

Keywords: Online, resource, share, blog, mobile, web, server

目录

[摘要 0](#_Toc524464775)

[ABSTRACT 1](#_Toc524464776)

[第一章 绪论 1](#_Toc524464777)

[1.1 选题的目的与意义 1](#_Toc524464778)

[1.2 国内外研究现状 1](#_Toc524464779)

[1.2.1 国内研究现状 1](#_Toc524464780)

[1.2.2 国外研究现状 2](#_Toc524464781)

[1.3 本报告主要工作 2](#_Toc524464782)

[1.4 复杂工程问题归纳 3](#_Toc524464783)

[1.5 报告章节安排 3](#_Toc524464784)

[第二章 相关技术研究 4](#_Toc524464785)

[2.1 相关技术介绍 4](#_Toc524464786)

[2.2 知识技能学习情况 5](#_Toc524464787)

[2.3 本章小结 5](#_Toc524464788)

[第三章 系统需求分析 6](#_Toc524464789)

[3.1 系统特点 6](#_Toc524464790)

[3.2 功能需求 6](#_Toc524464791)

[3.2.1 登录与注册 8](#_Toc524464792)

[3.2.2博客部分 13](#_Toc524464793)

[3.2.3评论部分 17](#_Toc524464794)

[3.2.4资源下载 20](#_Toc524464795)

[3.2.5说说功能 24](#_Toc524464796)

[3.2.6个人中心 26](#_Toc524464797)

[3.3 非功能需求 29](#_Toc524464798)

[3.3.1 性能需求 29](#_Toc524464799)

[3.3.2 易用性需求 29](#_Toc524464800)

[3.3.3 安全性需求 29](#_Toc524464801)

[3.4 可行性研究 29](#_Toc524464802)

[3.5 本章小结 29](#_Toc524464803)

[第四章 系统概要设计 30](#_Toc524464804)

[4.1 系统总体架构设计 30](#_Toc524464805)

[4.2 后台框架设计 31](#_Toc524464806)

[第五章 系统详细设计 32](#_Toc524464807)

[5.1 数据库设计 32](#_Toc524464808)

[5.2 模块设计 38](#_Toc524464809)

[5.2.1 用户账号模块设计 38](#_Toc524464810)

[5.3.2 个人中心模块设计 38](#_Toc524464811)

[5.3.3 我的博客模块 39](#_Toc524464812)

[5.3.4 数据资源分享 40](#_Toc524464813)

[5.3.5 说说圈 40](#_Toc524464814)

[5.3 本章小结 41](#_Toc524464815)

[第六章 系统实现 42](#_Toc524464816)

[6.1 用户账户模块 42](#_Toc524464817)

[6.2 个人中心模块实现 42](#_Toc524464818)

[6.3 我的博客模块 46](#_Toc524464819)

[6.4数据资源分享 54](#_Toc524464820)

[6.5说说圈 61](#_Toc524464821)

[6.4 本章小结 65](#_Toc524464822)

[第七章 系统测试 67](#_Toc524464823)

[7.1 测试环境 67](#_Toc524464824)

[7.2 功能测试 67](#_Toc524464825)

[7.2.1用户账号模块测试 67](#_Toc524464826)

[7.2.2个人中心模块测试 68](#_Toc524464827)

[7.2.4我的博客模块测试 69](#_Toc524464828)

[7.2.5数据资源模块测试 69](#_Toc524464829)

[7.2.6说说圈模块测试 69](#_Toc524464830)

[7.3 本章小结 70](#_Toc524464831)

[第八章 工程能力培养 70](#_Toc524464832)

[8.1 工程计划管控和执行情况 70](#_Toc524464833)

[8.2 对本工程领域发展的认识 72](#_Toc524464834)

[8.3 工程协作与交流 73](#_Toc524464835)

[第九章 结束语 74](#_Toc524464836)

[9.1 全文总结 74](#_Toc524464837)

[9.2 不足与下一步工作 74](#_Toc524464838)

[参考文献 75](#_Toc524464839)

[致谢 77](#_Toc524464840)

# 第一章 绪论

1.1 选题的目的与意义

21世纪是信息的时代，也是合作的时代，因为资讯太发达，而且对人对团队的要求也更高。只有知识、资源共享，优势互补，才能达成共赢。在线资源分享

可以促进人与人之间的交流，加强联系，为人们足不出户就能获取信息提供了一个信息的平台，资源共享是指通过共有或共用的方式使资源稀缺方获得所需资源。从源头上讲，资源是社会、经济和科技发展的基础物质，因此共享是社会发展中行为主体的一种权利。

实现人们在线信息的便捷分享，使人与人可以相互传递信息。通过本平台，人们可以方便分享资源和交流技术，使得各地人们可以方便的进行学习经验交流，资源分享和评论互动等。促进人们相互学习，形成一个庞大的资源共享平台，赋予网络领域的新特色。分享可以改变世界、可以改善人际关系，让人们多了一个交友平台，让“我”变成“我们”。在平台上传一份快乐的信息可以让其他人收获快乐，在线资源分享增强人们的联系，增加一份团结。

1.2 国内外研究现状

1.2.1 国内研究现状

在国内有一下几个类似的网站：

（1）中国科技资源共享网

中国科技资源共享网是[科技部](https://baike.baidu.com/item/%E7%A7%91%E6%8A%80%E9%83%A8)、财政部推动建设的国家科技基础条件平台门户网站，其宗旨是充分运用现代技术，推动科技资源共享，促进全社会科技资源优化配置和高效利用，提高我国科技创新能力。共享网由国家科技基础条件平台中心管理，建设、运行、维护单位为国家科技基础条件平台信息技术中心，该中心是科技部和北航共建的公益性技术支撑单位。共享网于2009年9月25日正式开通。

（2）博客网

博客网，原名博客中国，是IT分析家[方兴东](https://baike.baidu.com/item/%E6%96%B9%E5%85%B4%E4%B8%9C)先生于2002年8月发起成立的知识[门户网站](https://baike.baidu.com/item/%E9%97%A8%E6%88%B7%E7%BD%91%E7%AB%99)。作为[第二代互联网](https://baike.baidu.com/item/%E7%AC%AC%E4%BA%8C%E4%BB%A3%E4%BA%92%E8%81%94%E7%BD%91)门户，[博客](https://baike.baidu.com/item/%E5%8D%9A%E5%AE%A2)网是中立、开放和人性化的精选[信息资源共享](https://baike.baidu.com/item/%E4%BF%A1%E6%81%AF%E8%B5%84%E6%BA%90%E5%85%B1%E4%BA%AB)平台。2003年底，博客网已经成为全球中文第一[博客网站](https://baike.baidu.com/item/%E5%8D%9A%E5%AE%A2%E7%BD%91%E7%AB%99/9164864)。2005年7月，博客中国正式更名为“博客网”。 “越分享越多”，大概是最好的概括了。对于博客本人，对于阅读博客的人；对于自己，对于他人，对于社会；对于工作，对于生活，“越分享越多”的博客精髓注定了它的发展是势不可挡。

由此可见资源共享在国内是有很大的发展空间的，并且本系统网站对比相关网站有很多特色。

1.2.2 国外研究现状

以下是几个国外开发的相关网站：

（1）Instagram

Instagram是一个提供在线照片共享，视频共享和社交网络服务的应用程序，该程序允许用户拍摄照片和视频并把它们分享在各种社交网络，如：Facebook、微博、QQ空间分享和Flickr分享。该程序显着特点是它限制正方形的形状照片，并采用类似柯达傻瓜相机和宝丽来影像，而相比之下，16:9宽高是现在通常使用的移动设备相机比率模式。

（2）DropBox

Dropbox是由Dropbox公司运营的同步本地文件的网络存储在线应用，公司总部位于加州旧金山，该公司提供云存储，文件同步和客户端软件服务。该软件允许用户为自己的计算机创建专用的文件夹，然后同步到Dropbox上，其他用户可以查看该文件夹。文件夹里的文件也可以通过网站和手机应用程序上传。

（3）MediaFire

MediaFire是一个免费的文件和图像托管网站，它于2005年3月21日在美国得克萨斯州的雪兰创建的。根据Compete.com在2008年调查，它每年吸引将近60万人次的访问量。2012年7月，它提供多达50GB的存储空间。它被PC杂志评为“未被发现的百强网站”和“2008年热门网站”。它也受到CNET和Lifehacker好评。

（4）iCloud

iCloud是苹果公司针对iOS系统推出的一个云存储和云计算服务。用户可以访问iCloud里的照片和视频，Apple音乐，以及iPhone，iPad，Mac或PC上购买的任何电影。iCloud使用户能够在远程服务器上存储诸如文档，照片和音乐的数据，以便下载到ios或Windows设备，并向其他用户共享和发送数据，或在丢失或损坏的情况下管理其Apple设备。

国外也有着类似的网站，及其受广大人民群众的欢迎，所以前景良好。

1.3 本报告主要工作

本文主要介绍了在线资源共享平台系统的设计与实现过程，包括：

在线资源共享系统的需求分析

基于JSP在线资源共享需求分析包括功能性需求和非功能性需求两个方面，功能性需求主要包括：普通用户注册账号、登录系统、上传下载博客、资源下载、便签分享、个人管理等。非功能性需求包括性能需求

在线资源系统的设计

针对之前的需求分析，对本系统进行大体上的概要设计和全局上的系统设计，描述整个系统的架构和各个模块之间的关系，之后对各个模块进行详细设计。

1.4 复杂工程问题归纳

第一个重要问题是如何挖掘发明人之间的合作关系，如果使用传统的关系型数据库，那么就会涉及多表查询问题，首先我们需要查询出该发明人的所有专利，之后需要逐条专利查询所有的发明人，找到与该发明人有合作的其他发明人。这样虽然能够实现，但当数据量庞大时，查询效率会变得非常低下，那么如何保证在数据量庞大后仍能满足性能要求是亟待解决的一个问题。

第二个重要问题是如何通过关键词对专利查询。很多时候不仅仅是查询专利名称就可以满足查询需要的，很多时候，专利的关键词是隐藏在专利说明书中的，因此我们急需一种基于专利内容方案，但是专利的正文字数非常庞大，如果针对正文进行关键词提取，工作量会变得非常庞大，而摘要是专利全文的梗概，如果我们能从摘要中提取关键词，同样可以满足基于关键词查询的需求。

1.5 报告章节安排

本文第二章主要介绍了实现课题的关键技术和方法。

第三章主要描述系统的需求分析包括系统特点、功能性需求和性能需求。

第四章主要论述如何通过需求分析对该系统进行设计，包括系统架构设计和模块设计。

第五章主要介绍系统的实现过程以及各个模块使用到的方法定义、输入/输出参数、程序流程图和核心处理环节的关键代码。

第六章主要介绍系统的测试过程，包括测试环境、功能测试和性能测试。

第七章陈述了本人在毕业设计实现过程中的工程能力与职业素养培养。

第八章为结束语，总结以上的工作，指出不足之处和对下一步工作。

# 第二章 相关技术研究

2.1 相关技术介绍

针对系统的解决方案，从设计层面上来看，可以用Vsiso进行数据库设计，进行用例图、活动图时序图的设计。

对于web前端的开发，我们主要用hbulider编辑器来对面向用户的界面进行代码的编写。其中把HTML、CSS、JavaScript、JQuery等技术相互配合使用，从而达到用户的需求，在此处我参考了W3C（w3cschool）网站中对HTML、CSS、JavaScript、JQuery等技术的规范要求，主要使用的浏览器是谷歌的Chrome浏览器。

在静态界面中，使用技术HTML和CSS对页面进行布局排列、样式设置等内容。其中在对HTML标记语言的书写中，用了HTML5的标签属性等内容。在CSS中，我们通过使用各式各样的选择器、伪类等对HTML中的标签内容进行样式的设置。静态界面有了大致的网页内容，所以用JavaScript把静态界面的其中一些内容变成动态内容（例如：轮播图等）。

在后台管理网页中，依旧用到了技术HTML和CSS进行网页静态界面的编译，其中运用了frameset框架以便页面进行部分的换页，此外还用到了JavaScript的JQuery中的Ajax写了请求后台内容的数据，jQuery是一个快速、简洁的JavaScript框架，是继Prototype之后又一个优秀的JavaScript代码库，它封装JavaScript常用的功能代码，提供一种简便的JavaScript设计模式，优化HTML文档操作、事件处理、动画设计和Ajax交互从而使面向管理员的网页能完整的被操作，更好的对面向用户的网页进行管理。

B/S架构的应用采用JavaWeb的技术来实现，对于具体的开发框架，我们选用的是应用比较广泛和成熟的SSH（SpringMVC、Spring、Hibernate）整合框架来完成后台服务端的开发，SpringMVC用于控制请求处理和分发,Spring用于依赖注入,协助完成服务层开发,Spring框架就是在运行的时候,根据Spring的 applicationContext.xml配置文件来动态的，创建对象并调用对象的方法的，所以 使用Spring框架后不 需要用new方法创建而是通过框架自动产生一个对象，只需要在xml文件中对bean进行定义。Spring发展到3.0后,不需要在xml文件中逐个定义bean对象,只需在调用的文件中使用@autowired来声明所调用的对象即可，使开发更加简洁方便。调用Hibernate完成数据库的访问与修改, Hibernate 是 J2EE 应用开发中的一个持久层框架， 它通过配置的 XML 文件或者注解实现对象和存储过 程，SQL 语句的关联，而不是实现对象与数据库表的关联，并且可以实现动态生成 SQL 语句。Hibernate框架集合多种操作型关系数据的概念和方法，它是一个强大 的数据访问工具和解决的方法。

2.2 知识技能学习情况

在进行新项目的开发之前，我通过的菜鸟教程来看HTML、CSS、JavaScript、JQuery和ajax的规范标准，大概了解了下前端基础语法，又购买了相关J2EE书籍，学习了J2EE、Spring、Hibernate、等后端框架等基本内容，让我对后端框架有个基本的理解。

通过学习明白了整个前后端交互的过程，1.做前后端分离，前端通过http协议提交请求和相关数据，我们后端对请求做分发、业务处理、反馈结果、浏览器通过前端技术把、数据显示出来，2.后端可以写jsp脚本页面，把数据渲染部分交给服务器做。在做了一系列的学习工作之后，很快就掌握了这两种开发思路。

2.3 本章小结

本章介绍了毕业设计项目开始之前的知识学习情况，包括后台框架的选用调研，工作量，数据库的选择，并对后台框架和数据库进行了一定的了解和学习。

# 第三章 系统需求分析

3.1 系统特点

在线资源共享是面向任何种类的人群，具有极多的潜在用户。本系统结合了类似于博客网、网盘、空间动态等功能，涉及广泛，用户可以在家用web端和在外用手机端查看最新消息，方便快捷。

3.2 功能需求

在线资源分享平台是可以为用户提供获取资源的平台，用户获得资源的同时也可以上传自己的资源，这样就使发布上去的资源越来越多，用户能获取到的资源种类越来越多，从而使更多的潜在用户变成真正的用户，发布更多的资源。如此循环下去，在线资源共享会越来越有价值。而对于用户可以进行以下的操作：用例图如图3-1所示。

1.注册管理个人账号基本信息

2.管理个人博客信息：博客主页、写博客、发博客、管理博客分类

3.浏览主页：查看相关博文、过滤博文、搜索博文

4.管理个人数据资源：上传、修改个人数据资源、浏览他人数据源

对于管理员——整个系统的维护人员，就可以对用户的账号进行增删查改的管理，博文进行浏览删除管理，博文分类标签进行增删查改管理，资源数据进行浏览删除管理。这些操作是普通用户不能涉及操作的，所以对管理员有：

1.会员账号管理：停封、查看、浏览会员账号

2.博文管理：停封、查看、浏览博文

3.博客分类管理：新增、修改、删除博客分类标签

4.数据资源管理：停封、查看、浏览用户相应数据资源



图3-1 基于JSP在线资源共享

根据软件本身的系统结构，从功能的角度对系统的功能进行了初步的模块划分，主要将系统分为

1. 登录及相关模块，其包含注册、激活、登录、找回密码四个功能点。

2. 用户中心模块，其包含查询个人信息、修改密码、修改邮箱、收藏夹相关操作、管理员查看其他用户个人信息、管理员查看用户操作日志功能六个功能点。

用例图如图3-1。

用例图

根据用例图，可以进一步细化每个用例之间的业务关系，从业务层面针对本系统进行需求分析，目的是理清系统的操作逻辑和各个功能之间的耦合关系。系统业务流程图如图3-2。

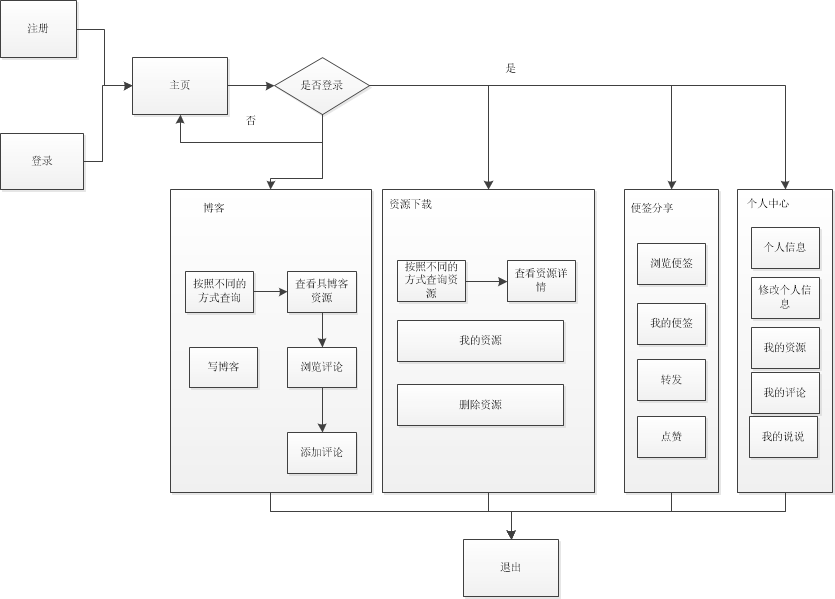


图3-2 系统业务流程图

3.2.1 登录与注册

(1)登录

角色：普通用户、管理员

用例描述：输入账号密码登录系统

前置条件：用户进入登录界面

后置条件：用户登录验证通过则跳转页面，否则继续停留在登录界面用户进入登录界面。

基本事件流：

1. 用户输入账号密码

2. 前端对输入信息进行格式校验

3. 后台判断用户名和密码是否匹配

4 返回登录结果

登录流程图如图3-3。

图3-3 登录流程图

(2)注册

角色：普通用户、管理员。

用例描述：填写基本信息注册账号以使用本系统。

前置条件：用户进入注册界面。

后置条件：用户注册成功则生成新的账号，否则提示用户注册失败继续注册。

基本事件流：

1. 用户输入注册所必需的信息

2. 前端对输入信息进行格式校验

3. 后台对用户数据进行存储，生成新账号

4. 返回注册结果

注册流程图如图3-5。



图3-5 注册流程图

3.2.2博客部分

(1)博客专栏

角色：普通用户

用例描述：用户在此专栏中可以查看到博客分类信息及最新的博文推送。

前置条件：用户登陆后，进入主页。

后置条件：点击博客专区，进入博客专栏页面。

基本事件流：

1. 用户在主页点击博客专栏区域

2. 进入博客专栏

3. 此区域显示最新的博客分类信息和最近的博文信息

4. 返回查询结果页面

博客专栏流程图如图3-6。



图3-6 博客专栏流程图

(2)按照博客分类查找博文

角色：普通用户

用例描述：在博客页面点击分类信息以查找该分类博文信息。

前置条件：用户进入主页后，点击博客按钮，进入博客页面。

后置条件：用户点击相关分类标签后，该页面显示出该分类下的所有博客文章。

基本事件流：

1. 用户点击分类区的分类标签

2. 前端对输入信息进行格式校验

3. 后端在数据库中查询该分类的博客

4. 返回查询结果页面

分类查找流程图如图3-7。



图3-7 分类查找流程图

(3)按照关键字搜索博文

角色：普通用户

用例描述：在博客页面搜索框输入搜索关键字，检索博文。

前置条件：用户进入主页后，点击博文标签，进入博文专栏。

后置条件：用户在搜索框中输入搜索关键字，然后跳转至相应页面显示搜索结果。

基本事件流：

1. 用户在搜索框中输入搜索关键字

2. 前端对输入信息进行格式校验

3. 后端在数据库中查询与该关键字相关的博客信息

4. 返回查询结果页面

关键字搜索流程图如图3-8。



图3-8 关键字搜索流程图

(4)浏览指定博文

角色：普通用户

用例描述：在博客页面搜索框输入搜索关键字，检索博文

前置条件：用户进入主页后，点击博文标签，进入博文专栏，通过不同检索方式检索出博文列表。

后置条件：用户点击列表中某一条博文，跳转至该博文详细界面

基本事件流：

1. 用户通过各种方式过滤出的博文列表

2. 用户点击某一博文信息

3. 跳转至相应页面显示出博文详细信息

浏览指定博文流程图如图3-9。



图3-9 浏览指定博文流程图

(5)写博文

角色：普通用户

用例描述：在用户可以把自己的资源技术归纳总结写成文章发表在博文区域专栏。

前置条件：用户登陆后，点击主页右侧写博客按钮。

后置条件：用户填入相应信息点击发布，在博客专栏中发布博文信息。

基本事件流：

1. 用户填写文章标题/文章分类/文章内容等信息

2. 用户点击发表博文，上传并发表博文信息

3. 跳转至博客专栏显示最近发表的博文

3.2.3评论部分

(1)评论博文

角色：普通用户

用例描述：用户在查看其他用户的博文后可以发表个人评论。

前置条件：用户进入他人博文详细界面，浏览评论区。

后置条件：用户在评论区输入评论信息，点击发表，更新评论区信息。

基本事件流：

1. 用户在评论区填写评论信息

2. 用户点击发表按钮发布评论信息

3. 刷新评论区显示最新评论信息

评论博文流程图如图3-10。



图3-10 评论博文流程图

(2)编辑删除我的博文评论

角色：普通用户

用例描述：用户在博文详情界面的评论区编辑自己的评论信息。

前置条件：用户在博文详情界面的评论区，看到自己的评论信息。

后置条件：用户点击删除评论按钮，刷新评论区内容。

基本事件流：

1. 用户在评论区查看自己的评论

2. 用户点击

3. 刷新评论区显示最新评论信息

编辑删除评论流程图如图3-11。



图3-11 编辑删除评论流程图

3.2.4资源下载

(1)资源下载专栏

角色：普通用户

用例描述：用户在此专栏中可以查看到资源分类信息及最新热门的资源推送。

前置条件：用户登陆后，进入主页。

后置条件：点击资源专区，进入资源专栏页面。

基本事件流：

1. 用户在主页点击资源标签

2. 进入到资源专栏

3. 此区域显示最新的资源分类信息和最近热门的资源信息

4. 返回查询结果页面

资源下载流程图如图3-12。

图3-12 资源下载流程图

(2)搜索数据资源

角色：普通用户

用例描述：在资源页面点击分类下拉菜单信息以查找该分类博文信息。

前置条件：用户进入主页后，点击资源标签，进入资源专栏页面。

后置条件：用户通过分类下拉菜单结合搜索关键字，检索出相关资源信息。

基本事件流：

1. 用户选择指定资源分类

2. 输入搜索信息

3. 后端在数据库中查询该分类下匹配关键字的数据资源

4. 返回查询结果页面

搜索数据流程图如图3-13。



图3-13 搜索数据流程图

(3)查看指定资源

角色：普通用户

用例描述：用户查看指定资源的详细信息。

前置条件：用户在资源专栏查看最近列表或通过搜索过滤出相关资源列表。

后置条件：用户点击某一指定资源项，跳转到该资源的详细界面。

基本事件流：

1. 用户选择指定资源分类

2. 输入搜索信息

3. 后端在数据库中查询该分类下匹配关键字的数据资源

4. 返回查询结果页面

(4)新增资源

角色：普通用户

用例描述：用户可以新增发布分享自己喜欢的资源。

前置条件：用户在我的资源页面点击新增资源按钮。

后置条件：用户填写资源基本信息并上传资源，点击发布按钮发布资源。

基本事件流：

1. 用户在我的资源截面，点击新增资源按钮。

2. 用户输入基本资源信息并且上传资源。

3. 后台存储资源文件及资源描述信息。

4. 跳转到资源专栏，显示最新发布的资源信息

新增资源流程图如图3-14。



图3-14 新增资源流程图

3.2.5说说功能

(1)说说圈

角色：普通用户

用例描述：用户在此区域可以浏览到其他用户的动态信息。

前置条件：用户登陆后，在主页点击说说标签。

后置条件：用户在此区域可以浏览其他用户的说说。

基本事件流：

1. 用户在主页点击说说标签

2. 跳转到说说圈页面，浏览其他用户最新说说

说说圈流程图如图3-15。



图3-15 说说圈流程图

(2)发布说说动态

角色：普通用户

用例描述：在说说圈页面发表自己最新动态，分享给其他用户。

前置条件：用户登陆后，在主页点击说说导航标签。

后置条件：用户填写动态内容点击发表说说。

基本事件流：

1. 用户在说说圈板块填写自己的说说内容，点击发表

2. 发表成功，显示用户发表的最新说说列表

(3)删除我的说说

角色：普通用户

用例描述：用户在说说圈板块可以删除自己的说说。

前置条件：用户在说说圈板块浏览到自己的说说。

后置条件：用户点击删除按钮，提示操作结果。

基本事件流：

1. 用户在说说圈板块查看到自己的说说信息

2. 用户点击删除按钮，删除该条动态

3. 刷新列表显示删除结果

删除说说流程图如图3-16。



图3-16 删除说说流程图

3.2.6个人中心

(1)我的信息

角色：普通用户

用例描述：用户在此截面可以查看到基本信息，并且管理个人信息及最新动态。

前置条件：用户登陆后在主页点击我的板块进入我的个人信息板块。

后置条件：用户在此区域点击其他小功能按钮进行操作。

基本事件流：

1. 用户在该板块可以看到自己的基本信息
2. 用户可以修改个人头像或者主题背景

3. 用户可以在此页面查看到自己的博客/说说/分享资源信息

我的信息流程图如图3-17。

图3-17 我的信息流程图

(2)编辑个人信息

角色：普通用户

用例描述：用户可以修改个人基本信息。

前置条件：用户在我的信息板块，点击修改个人信息按钮。

后置条件：用户在表单内填写预修改的信息，确认修改。

基本事件流：

1. 用户在修改页面可以看到当前用户的基本信息
2. 用户修改指定项基本信息
3. 用户提交修改
4. 反馈修改结果

编辑个人信息流程图如图3-18。



图3-18 编辑个人信息流程图

3.3 非功能需求

3.3.1 性能需求

用户对注册、登录、找回密码、查看个人信息、修改邮箱、修改密码、检索这几个接口的访问相应时间不超过1秒，其他接口响应时间可以略微延长，但应不超过1.5秒。

3.3.2 易用性需求

用户使用系统基本无障碍，完成各项功能的流程简单并符合用户的操作逻辑。

3.3.3 安全性需求

只有在本系统注册过的用户才能访问数据、只有获得授权的用户才能对某些敏感信息进行修改。

3.4 可行性研究

本节描述该系统在社会、经济、环境、法律和技术等方面的可行性论证。

3.5 本章小结

本章主要介绍了在线资源共享的需求分析部分，包括系统特点、功能需求、性能需求三个方面，比较详细地分析了资源分享平台的各个方面的内容，接下来就是根据希求分析进行系统总体设计和详细设计，这些内容将在第四章进行展开详细说明。

# 第四章 系统概要设计

4.1 系统总体架构设计

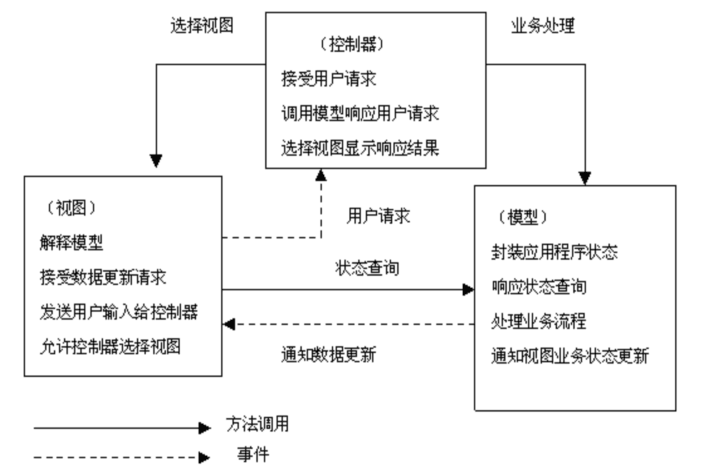


图4-1 系统架构图

系统基于MVC设计模式，整个系统可以分为：视图层，用于显示和渲染数据；控制层：业务模块流程的控制；服务层：业务模块的逻辑应用设计；持久层：与数据库进行交互。

数据库使用MySQL数据库。

4.2 后台框架设计

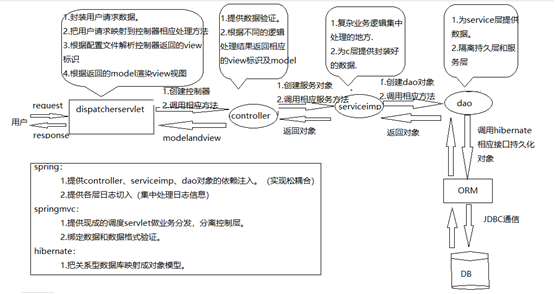


图4-3 系统结构图

按照之前的需求分析，整个系统前端分为八大模块，手机端分为六大模块，这几大模块将整个系统进行了功能上的划分，也明确了每一个模块中的功能点分布情况。

# 第五章 系统详细设计

5.1 数据库设计

对于在线资源共享数据存储采用的是MySQL关系型数据库，实体与实体之间关系更明确，更易于开发、事务管理、等后期维护。ER图设计如图5-1所示，数据字典如表5-1到5-8：

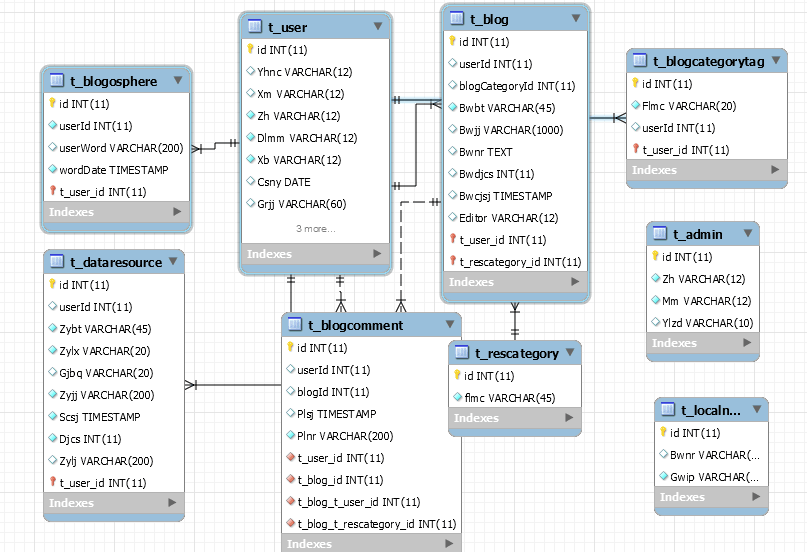


图5-1 ER图设计

表5-1 用户信息表设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 长度 | 是否可空 |
| id | 用户编号 | Int | 11 |  |
| Yhnc | 用户昵称 | Varchar | 12 | √ |
| Xm | 姓名 | Varchar | 12 | √ |
| Zh | 账户 | Varchar | 12 |  |
| Xb | 性别 | Varchar | 12 | √ |
| Cnsy | 出生年月 | Date |  |  |
| Grjj | 个人简介 | Varchar | 60 |  |
| Jybs | 禁用标识 | Char | 1 |  |
| userImg | 头像地址 | Varchar | 200 |  |
| userBgImg | 用户背景图片 | Varchar | 200 |  |

表5-2 数据资源表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 主 键 | 数据类型 | 长度 | 说 明 |
| Id | Id | √ | Integer | 11 |  |
| Zybt | 资源标题 |  | Varchar | 45 |  |
| Zylx | 资源类型 |  | Varchar | 20 | 资源分类表的主键 |
| Gjbq | 关键标签 |  | Varchar | 20 |  |
| Zyjj | 资源简介 |  | Varchar | 200 |  |
| Scsj | 上传时间 |  | Date |  |  |
| Djcs | 点击次数 |  | Integer |  |  |
| Zylj | 资源路径 |  | Varchar | 200 |  |
| userId | 上传用户 |  | Int | 11 | 用户表主键 |

表5-3 博客信息表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 数据类型 | 长度 | 主键 | 非空 |
| Id | 编号 | Int | 11 | √ | √ |
| Bwbt | 博文标题 | Varchar | 45 |  | √ |
| userId | 关联用户id | Int | 11 |  | √ |
| Bwjj | 博文简介 | Varchar | 1000 |  | √ |
| Bwnr | 博文内容 | Text |  |  |  |
| BlogCategoryId | 博文分类 | Int | 11 |  |  |
| Bwdjcs | 博文点击次数 | Int | 11 |  |  |
| Bwcjsj | 博文创建时间 | TIMESTAMP |  |  |  |
| Editor | 作者 | Varchar | 12 |  |  |

表5-4 博文评论表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 数据类型 | 长度 | 主键 | 非空 |
| Id | 主键 | Int | 11 | √ | √ |
| UserId | 用户 | Int | 11 |  | √ |
| BlogId | 博客id | Int | 11 |  | √ |
| Plsj | 评论时间 | TIMESTAMP |  |  | √ |
| Plnr | 评论内容 | Varchar | 200 |  | √ |

表5-5 博文分类表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 数据类型 | 长度 | 主键 | 非空 |
| Id | 主键 | Int | 11 | √ | √ |
| Flmc | 分类名称 | Varchar | 20 |  | √ |
| userId | 用户id | Int | 11 |  | √ |

表5-6说说圈信息表设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 长度 | 是否可空 |
| Id | 主键 | Int | 11 | 否 |
| userId | 用户id | Int | 11 | 否 |
| userWord | 用户说说内容 | Varchar | 200 | 否 |
| wordDate | 发表时间 | TIMESTAMP |  | 否 |

表5-7资源分类信息表设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 长度 | 是否可空 |
| Id | 主键 | Int | 11 | 否 |
| Flmc | 分类名称 | Varchar | 45 | 否 |

表5-8管理员表设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 长度 | 是否可空 |
| id | 编号 | Int | 11 |  |
| Zh | 管理员账号 | Varchar | 12 |  |
| Mm | 管理员密码 | Varchar | 12 |  |
| Ylzd | 预留字段 |  | 4 | √ |

5.2 模块设计

5.2.1 用户账号模块设计

此模块功能覆盖用户登录本系统及其相关内容，主要包括：用户注册和登录两个功能点。系统登录子模块组成图如图5-2：

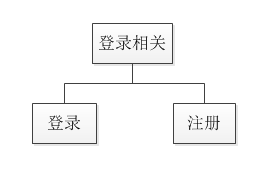


图5-2 登录相关子模块组成图

注册：注册需要用户输入用户名、昵称、密码、性别、出生日期。其中昵称、性别，用户名填写手机号，密码必须是6-10位，出生日期必须是过去的时间，其格式为”yyyy-MM-dd”。用户完成注册后可能返回提示两种提示信息：注册成功或注册失败的提示信息。

登录：用户输入用户名和密码进入系统，可能返回给用户的信息：登录成功、用户名或密码错误。当用户登录成功后跳转到博客主页。

5.2.2 个人中心模块设计

用户个人中心模块包含查询个人信息、修改密码、修改基本信息、修改头像及修改背景主题、我的博客列表、我的博客评论列表、我的说说列表。

查询个人信息:用户可以通过此接口查看到个人基本信息。

修改密码:用户可以修改个人密码。

修改基本信息：用户可以修改个人基本信息。

修改头像及修改主题：用户可以上传新的头像和修改默认主题图片。

我的博客列表、我的博客评论列表、我的说说列表：用户可以在个人中心看到与用户本人有关的博客、评论、说说信息。

个人中心个人中心模块组成图5-3



相关子模块组成图5-3

5.2.3 我的博客模块

当用户登录成功后，进入我的博客模块，会展示已下部分功能：写博客、博文搜索、编辑个人博文、查看博友指定博文、博文评论留言。

写博客：用户可以在此区域写自己想要发表的博文。

博文搜索：用户输入博文关键字，可以检索出想要的博文。

编辑个人博文：用户可以查看和删除自己的博文。

查看指定博友博文：用户可以在博文专栏等推送其他博友博文的地方查看其他博友博文内容。

博文评论：用户可以在每篇博文下方的评论区输入自己的留言评论等。

我的博客模块组成图5-4



相关子模块组成图5-4

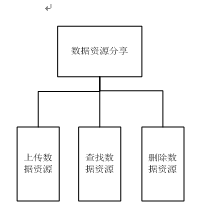
5.2.4 数据资源分享

此模块用户可以上传自己喜爱的数据资源，和他人分享，并且在资源专栏中浏览其他用户的数据资源。在此板块用户可以进行以下操作：上传数据资源、查找资源、浏览数据资源专栏。

上传数据资源：用户上传自己想分享的资源，并附上资源、分类、说明、等信息。

查找数据资源：用户可以通过输入关键资源描述信息或者通过分类标签查找匹配数据资源。

数据资源分享组成如图5-5



相关子模块组成图5-5

5.2.5 说说圈

用户在此可以发表自己最近的动态，查看其他人的动态。此模块有以下功能：发表说说、浏览说说、编辑说说。

发表说说：用户可以随时发表自己的心情。

浏览说说：用户在说说板块可以看到最近的一些说说信息。

编辑说说：用户可以删除自己的说说。

说说圈模块如图5-6



相关子模块组成图5-5

5.3 本章小结

本章根据之前的需求分析，先确定了系统的整体架构和系统结构，之后划分了系统模块，并对每一个模块进行了简要说明，并针对每个模块中涵盖的功能点进行了简单说明，明确了每个功能点的实现流程，接下来的第六章，我将基于其设计对每个功能点、每个模块的实现过程进行解释说明。

# 第六章 系统实现

6.1 用户账户模块

此模块含有注册、登录两个个功能点，用户必须先注册然后才能正常登录系统。由于这些功能是绝大多数系统都拥有的，在此省略实现过程。

6.2 个人中心模块实现

用户个人中心模块包含查询个人信息、修改密码、修改基本信息、修改头像及修改背景主题、我的博客列表、我的博客评论列表、我的说说列表。

我的个人主页会加载出用户的基本信息、头像、背景主题、该用户的说说、博客、及博客评论等列表

浏览个人中心页面流程：

* + - 1. 用户登录后
      2. 用户在主页点击“我的”导航标签
      3. 向服务器发送指定请求
      4. 服务器收到用户请求，调用相应业务逻辑方法从数据库中获取用户信息。返回指定页面。

输入参数：无

输出参数：“我的”板块jsp页面，携带用户基本信息、用户的评论、博客、说说等其他信息。

程序流程图如图5-1。



图5-1 新增收藏夹流程图

核心代码如图5-1

代码 5-1 新增收藏夹

|  |
| --- |
| public String showMyPsersonInf(HttpServletRequest request, Model model) {  TUser sessionUser=(TUser)request.getSession().getAttribute("mumber");  if (sessionUser == null) {  model.addAttribute("massage", "请先登录");  return "forward:/showHome"; }  JSONObject massage = new JSONObject();  massage.put("massage", "操作成功");  TUser user = (TUser) request.getSession().getAttribute("mumber");  List<TBlogcomment> bcList=null;//我的评论  List<TBlog> blogList=null;//我的博客  List<BlogoSphere> bpList=null;//説説  logger.info("user" + user);  if (user != null) {  logger.info("user" + user);  if (user.getUserImg() != null) { StringbasePath=request.getServletContext().getRealPath("/uploadfile").replaceAll("\\\\", "/")+"/";  user.setUserImg(user.getUserImg());  logger.info("userimguri"+user.getUserImg());  } bcList=blogCommentService.browseBlogCommentByUserId(user.getId());  blogList = blogService.loadBlogsByMember(user);  bpList=blogoSphereService.browseBlogSphereByUser(user.getId());  }  return "/webs/person"; } |

修改个人基本信息：

用户获取本人详细信息

用户修个某项信息

用户提交修改

服务器校验用户输入

向数据库保存新的个人信息

提示修改结果

程序流程图如图5-2。



图5-2 修改个人信息

核心代码如代码5-2

代码 5-2 修改个人信息

|  |
| --- |
| public String updateMember(HttpServletRequest request,@ModelAttribute("newUser") TUser newUser,BindingResult bindingResult,Model model) {  StringBuffer message = new StringBuffer();  String oldpwd = request.getParameter("oldpwd");  TUser user=newUser;  logger.info("nweUser"+newUser.toString());  //通过用户提供的id查询用户对象  TUser tempUser =(TUser) request.getSession().getAttribute("mumber");  //修改对象内部状态  if (user.getYhnc()!=null){  tempUser.setYhnc(user.getYhnc());  message.append("用户昵称,");  }  if(user.getDlmm()!=null&&user.getDlmm().trim().length()>0) {  if (oldpwd.equals(tempUser.getDlmm())) {  tempUser.setDlmm(user.getDlmm());  message.append("密码,");  } else {  message.append("密码修改失败");  model.addAttribute("massage",message);  return "/webs/editorPerson"; }  }  if(userService.saveOrUpdateMember(tempUser)){  logger.info("updateMember\_success");  } return "forward:/showMypersionInf"; |

6.3 我的博客模块

此功能模块处理所有关于博文的功能，如下

用户：

* 写博客
* 博文搜索
* 编辑个人博文
* 查看博友指定博文
* 博文评论留言

管理员：

* 查看所有用户发表的博文信息
* 创建并管理博文分类标签

1. 写博客：
2. 用户在写博客页面撰写博客分类、内容等信息
3. 用户可通过富文本编辑器修改字体、文章样式等、可以上传图片。
4. 用户提交修改
5. 服务器校验数据
6. 服务器存储数据到数据库
7. 返回处理结果

输入参数：1.博客标题2.博客内容3.博客分类

输出内容：新增成功

程序流程图如图5-3。



图5-3 写博客

核心代码如代码5-3

代码 5-3 写博客

|  |
| --- |
| TUser sessionUser=(TUser) request.getSession().getAttribute("mumber"); blog.setBwcjsj(new Date());//博客创建时间 blog.setBwdjcs(new Integer(0));//初始化博客点击次数 //blog.setEditor(userService.loadMember(blog.getUserId())); if (sessionUser!=null){  blog.setEditor(sessionUser.getYhnc());  blog.setUserId(sessionUser.getId());} blogService.saveOrUpdateBlog(blog); |

1. 博文搜索
2. 用户在博客专栏搜索框中输入搜索关键字并提交。
3. 服务器校验用户输入
4. 服务器在数据库中查询信息
5. 返回带结果数据的jsp模板

输入参数：搜索关键字keyword

输出参数jsp页面

程序流程图如图5-2。



图5-2 博文搜索

核心代码如代码5-2

代码 5-2 博文搜索

|  |
| --- |
| public String searchAllBlogs(HttpServletRequest request,Model model, @RequestParam()String searchContent,@RequestParam(required = false,defaultValue = "1") Integer pageNo) {  String uri = new String("/searchAllBlogs?pageNo=");  int count = blogService.searchBlogsCount(searchContent);  Page page = InitPage.getInstence(uri, pageNo, 4, count);  String searchContentbak = (String) request.getSession().getAttribute("rearchContent");  if (searchContent != null) {  request.getSession().setAttribute("rearchContent", searchContent); model.addAttribute("blogsList",blogService.searchBlogs(searchContent,pageNo,page.getPageSize()));  } else if (searchContentbak!=null){ model.addAttribute("blogsList",blogService.searchBlogs(searchContentbak,pageNo,page.getPageSize()));  }  model.addAttribute("catesList", categoryService.browseCategory());  model.addAttribute("page", page);  return "webs/boke"; } |

3.编辑个人博文

1.在带我的博文列表的页面点击指定博文对应的删除按钮，发送请求。

2.服务器更新数据库

输入参数：文章ID

输出参数：1.操作成功2.操作失败

程序流程图如图5-4。



图5-4 编辑个人博文

核心代码如代码5-4

代码 5-4 编辑个人博文

|  |
| --- |
| public String delBlog(Model model,@Validated(TBlog.BlogRuleD.class) TBlog blog, BindingResult bindingResult) {  StringBuffer message = new StringBuffer();  model.addAttribute("massage", message);  if (bindingResult.hasErrors()) {//遍历错误信息  List<ObjectError> errorList = bindingResult.getAllErrors();  for (ObjectError error : errorList) {  message.append(error.getDefaultMessage());  }  return "forward:/showMypersionInf";  }  TBlog tempBlog = blogService.loadBlogById(blog.getId());  if (tempBlog != null) {  List<TBlogcomment> bcList = blogCommentService.browseBlogCommentByBlogId(blog.getId());  for (TBlogcomment bc : bcList) {  blogCommentService.delComment(bc.getId());  }  blogService.delBlogById(tempBlog.getId());  message.append("删除成功");  } else {  message.append("没有此博文");  return "forward:/showMypersionInf";  }  return "forward:/showMypersionInf"; } |

5.查看博友指定博文

1. 用户点击指定博文列表的某一项
2. 浏览器发送请求
3. 服务器效验数据
4. 查询数据库
5. 返回带指定数据的jsp模板

输入参数：博文ID

输出参数：带博文结果的jsp页面

核心代码如代码5-5

代码5-5查看博友指定博文

|  |
| --- |
| if (id != null) {  TBlog blog = blogService.loadBlogById(id);  List <TBlogcomment> bcList=blogCommentService.browseBlogCommentByBlogId(id);  blog.setBwdjcs(blog.getBwdjcs()+1);//增加点击次数  blogService.saveOrUpdateBlog(blog);//保存修改  model.addAttribute("bcList", bcList);  model.addAttribute("user", currentuser);  model.addAttribute("blog", blog);  model.addAttribute("bc",new TBlogcomment());//绑定页面表格  logger.info("viewBlog after"+"bc size"+bcList.size()+"user"+currentuser); return "/webs/inner"; |

1. 查看博文评论
2. 用户提交博文ID
3. 服务器查找该博文ID下的所有评论及博文信息
4. 携带数据返回jsp模板

程序流程图如图5-6。



图5-6 查看博文评论

核心代码如代码5-6

代码 5-6 查看博文评论

|  |
| --- |
| public List<Object> browseBlogCommnetByBlogId(Model model, @RequestParam Integer blogId) {  List<Object> list=new ArrayList<>();  String massage="处理错误";  if (blogId != null) {  list.addAll(blogCommentService.browseBlogCommentByBlogId(blogId));  massage ="处理成功";  }  return list; } |

6.4数据资源分享

此模块用户可以上传自己喜爱的数据资源，和他人分享，并且在资源专栏中浏览其他用户的数据资源。

1. 上传数据资源
2. 用户写好数据资源基本信息
3. 用户上传数据文件
4. 服务器保存基本信息和文件

输入参数：基本资源信息 文件

输出结果：操作结果

程序流程图如图6-1。



图6-1 上传数据资源

核心代码如代码6-1

代码 6-1 上传数据资源

|  |
| --- |
| public String addDataResource(HttpServletRequest request, HttpServletResponse response,Model model, @ModelAttribute TDataresource data,@RequestParam(value = "upload",required = false) MultipartFile file) {  String massage=null;  TUser sessionUser=(TUser) request.getSession().getAttribute("mumber")；  Enumeration list = request.getParameterNames();  while(list.hasMoreElements()){  logger.info(list.nextElement().toString());  }  /\* \* 上传文件的属性\*/  String uploadFileName=file.getOriginalFilename();//上传文件的文件名  //创建一个StringBuffer对象存放返回给CKeditor的提示信息  StringBuffer sb = new StringBuffer();  String errMsg = "对不起，文件上传失败！";  try { response.setCharacterEncoding("UTF-8");  //取得UserFiles文件夹对应的物理路径  String basePath=request.getServletContext().getRealPath("/uploadfile").replaceAll("\\\\", "/");  //获取用户真实存储路径  userDataPath = basePath+"/"+sessionUser.getZh() + sessionUser.getId();  logger.info("basePath:"+basePath+"userDataPath"+userDataPath);  logger.info("uploadFileName"+uploadFileName);  File saveDir = new File(basePath);  if(!saveDir.exists())saveDir.mkdirs();  //当前登录用户的文件夹是否创建  File saveUserDataDir = new File(userDataPath);  if(!saveUserDataDir.exists())saveUserDataDir.mkdirs();  //处理上传文件  if (uploadFileName==null||uploadFileName.trim().length()<1){  //未选择上传文件时的错误提示信息  errMsg = "对不起，文件不能为空,请选择文件然后上传！";  errMsg = new String(errMsg.getBytes(),"iso8859-1");  // sb.append("window.parent.CKEDITOR.tools.callFunction(1,'','"+errMsg+"');\n");  }else{  String[] tmpNames = uploadFileName.split("\\.");  tempFilename = String.valueOf(System.currentTimeMillis());  String extName = null;  if (tmpNames!=null && tmpNames.length>1){  extName = tmpNames[tmpNames.length-1].toLowerCase();  tempFilename = tempFilename+"."+extName;  FileCopyUtils.copy(file.getBytes(), new File(userDataPath, tempFilename));  errMsg = "文件上传成功";  errMsg = new String(errMsg.getBytes(),"iso8859-1");   }else{ //保存文本数据  data.setScsj(new Date());//上传时间  data.setDjcs(new Integer(0));//点击次数  data.setZylj("/"+sessionUser.getZh() + sessionUser.getId()+"/"+tempFilename);  if (sessionUser!=null) data.setUserId(sessionUser);   if (dataResourceService.saveOrUpdateDataRes(data)) {  dataResourceService.saveOrUpdateDataRes(data);  massage = "上传资料成功";  }  model.addAttribute(massage);  return "forward:/showHome"; |

2.查找数据资源

1.用户选择资源分类和输入搜索关键字

2.服务器校验用户输入

3.服务器在数据库中查询数据

4.服务器返回查询结果

输入参数：资源呢分类ID 关键字

输出参数：操作结果

程序流程图如图6-2。



图6-2 查找数据资源

核心代码如代码6-2

代码 6-2 查找数据资源

|  |
| --- |
| List<TDataresource> dataList=null;  DataCategory dataCategory=null;  String uri =null;  Page page = null;  int count=0;//當前分類縂條數  String massage=null;//处理结果  String tempKeyWord = (String) request.getSession().getAttribute("keyWord");  List<DataCategory> cateList=dataCategoryService.browseAllDataCategory();//分类   if (cateId != null && (dataCategory = dataCategoryService.loadDataCategoryById(cateId)) != null) {  //分类不为空同时可查到此分类的信息  uri=new String("/searchDataByCateAndKey?cateId=" + cateId);/、  if(keyWord!=null){//如果关键字获取到  count = dataResourceService.loadDataResByDataCategoryCount(cateId.intValue(), keyWord);  page= InitPage.getInstence(uri,pageNo,4,count);  dataList=dataResourceService.loadDataResByDataCategory(dataCategory,keyWord,pageNo,page.getPageSize());  // count = dataResourceService.loadDataResByDataCategoryCount(cateId.intValue(), keyWord);  request.getSession().setAttribute("keyWord", keyWord);  }  else if (tempKeyWord!=null){//如果之前没有搜索关键字,从session中获取  count = dataResourceService.loadDataResByDataCategoryCount(cateId.intValue(), tempKeyWord);  page= InitPage.getInstence(uri,pageNo,4,count);  // count = dataResourceService.loadDataResByDataCategoryCount(cateId.intValue(), tempKeyWord); dataList=dataResourceService.loadDataResByDataCategory(dataCategory,tempKeyWord,pageNo,page.getPageSize());  }  page= InitPage.getInstence(uri,pageNo,4,count);  } else {//如果不能通过分类查找  massage = new String("非法分类参数");  }  model.addAttribute("url", "/searchDataByCateAndKey");  model.addAttribute("dataList", dataList);  model.addAttribute("page", page);  // model.addAttribute("uri", uri);  model.addAttribute("dataCategoryList",cateList );  model.addAttribute("cateId", cateId);  logger.info("dataList size"+dataList.size()+"page "+page.toString()); logger.info("after:cateId"+cateId+"keyWord"+keyWord+"pageNo"+pageNo+"pageSize"+page.getPageSize());  return "/webs/downloadSearchResult"; } |

1. 浏览数据资源专栏
2. 用户在主页点击数据资源专栏导航按钮
3. 用户在主页点击数据资源专栏导航按钮
4. 返回带数据的jsp模板

输出内容：带数据的jsp模板

程序流程图如图6-3。



图6-3 浏览数据资源

核心代码如代码6-3

代码 6-3 浏览数据资源

|  |
| --- |
| public String browseDataResourceByUser(HttpServletRequest request,Model model,@RequestParam(defaultValue ="1") Integer pageNo) {  String massage = null;  Page page = null;  int count=0;//总计  List<TDataresource> dataList=null;  String uri = "/browseDataResourceByUser?";  TUser sessionUser= (TUser) request.getSession().getAttribute("mumber");  if (sessionUser != null) {  count=dataResourceService.browseDataResourceByUserCount(sessionUser.getId());  page = InitPage.getInstence(uri, pageNo.intValue(), 5, count);  logger.info("page"+page.toString());  dataList = dataResourceService.browseDataResourceByUser(sessionUser.getId(), pageNo, page.getPageSize());  logger.info("dataList"+dataList.size());  }else{  page = InitPage.getInstence(uri, pageNo.intValue(), 5, count); }  model.addAttribute(massage);  model.addAttribute("page", page);  model.addAttribute("dataList",dataList);  return "/webs/download\_look\_res" } |

6.5说说圈

用户在此可以发表自己最近的动态，查看其他人的动态

1.浏览说说圈

1.用户在此可以查看到用户最新的说说

输入参数：无

输出内容：带返回数据的jsp模板

程序流程图如图7-1。



图7-1 浏览说说圈

核心代码如代码7-1

代码7-1 浏览说说圈

|  |
| --- |
| public String writeUserWordForm(Model model) {  logger.info("method"+CurrentLineInfo.getMethodName()+"line: "+CurrentLineInfo.getMethodName());  model.addAttribute("blogsSphere", new BlogoSphere());  return "redirect:/showUserBlogoSphere";  } |

2.发表说说

1. 用户填写动态内容，点击提交按钮提交
2. 服务器校验数据
3. 返回处理结果

输入参数：动态内容

输出内容：操作结果

程序流程图如图7-2。



图7-2 浏览数据资源

核心代码如代码7-2

代码 7-2 浏览数据资源

|  |
| --- |
| public String addUserWord(HttpServletRequest request,Model model, @ModelAttribute BlogoSphere blogoSphere) {  String massage =null;  blogoSphere.setWordDate(new Date());  TUser sessionUser=(TUser) request.getSession().getAttribute("mumber");  //logger.info("userId"+sessionUser.getZh());  // logger.info("userId"+sessionUser.toString());  //判断用户是否登录  if (sessionUser == null) {  model.addAttribute("massage", "请先登录");  return "forward:/showHome";  }  blogoSphere.setUser(sessionUser);  if (blogoSphereService.addUserWord(blogoSphere)) {   massage = "success";  } else {  massage = "error";  }  logger.info("Method"+CurrentLineInfo.getMethodName()+"line:"+CurrentLineInfo.getLineNumber()+"massage"+massage);  try {  massage=new String(massage.getBytes(),"iso8859-1");  } catch (UnsupportedEncodingException e) {  e.printStackTrace();  }  // model.addAttribute(massage);  return "redirect:/showUserBlogSphereByType"; } |

3.删除我的说说

1.用户在说说列表中点击相应删除按钮

2.服务器删除此信息

3.返回操作结果

前置参数：说说ID

输出那内容：操作及结果

程序流程图如图8-1。



图8-1 删除我的说说

核心代码如代码8-1

代码 8-1 删除我的说说

|  |
| --- |
| public String delUserWordByid(Model model, @RequestParam() Integer id) {  String massage = "操作成功";  if (!blogoSphereService.delUserWordById(id)) {  massage = "操作失败";  }  model.addAttribute("massage",massage);  return "redirect:/showMyBlogSphere"; } |

6.4 本章小结

本章讲述资源共享平台系统的实现过程，分别从登录、系统开启页、系统首页、系统便签分享、个人博文、个人中心六大模块并参照第五章的设计部分，对每个功能点的实现过程进行了比较详细的说明。

# 第七章 系统测试

7.1 测试环境

硬件： PC

操作系统：windows10

系统部署情况：部署在Tomcat远程服务器上

调试环境:IDEA

测试平台： browser(chrome)

7.2 功能测试

7.2.1用户账号模块测试

测试用户账号模块中、注册、登录 、修改密码、的流程是否有问题，同事设置了几条不同的用例，系统是否会返回错误信息。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 测试目的 | 测试用户从注册、登录、修改密码整个流程是否通过 | | | | | | |
| 编写人 | 周晔 | 时间 | 2018-8-15 | 状态 | | 检测结果为合格 | |
| 前置条件 | 用户未拥有账号 | | | | | | |
| 序号 | 测试步骤 | | | | 期待结果 | | 是否通过 |
| 1 | 用户输入用户名、密码、性别、出生年月、个人简介、验证码点击注册 | | | | 操作成功，用户收到注册成功提示弹窗 | | 通过，与预期结果一致。 |
| 2 | 用户输入已注册过的用户名 | | | | 操作失败，用户名重复 | | 不通过，与预期结果一致 |
| 3 | 两次输入密码不一致 | | | | 操作失败，用户两次输入密码不一致 | | 不通过，与预期结果一致 |
| 4 | 用户输入错误验证码 | | | | 操作失败，用户输入验证码有误 | | 不通过，与预期结果一致 |
| 5 | 用户输入刚才注册的账号和密码 | | | | 操作成功，用户登录成功 | | 通过，与预期结果一致 |
| 6 | 用户输入刚才注册的账号和错误的密码 | | | | 操作失败，用户账号或密码有误 | | 不通过，与预期结果一致 |
| 7. | 用户未输入账号或密码 | | | | 操作失败，用户账号或密码为空 | | 不通过，与预期结果一致 |
| 8. | 用户选择修改密码，用户输入原密码和新密码 | | | | 操作成功，用户修改密码成功 | | 通过，与预期结果一致 |
| 9 | 用户输入输入错误的原密码和新密码， | | | | 操作失败，用户原密码错误 | | 不通过， 预期结果一致 |
| 10 | 用户输入原密码和未输入新密码 | | | | 操作失败，新密码不能为空 | | 不通过，预期结果一致 |

7.2.2个人中心模块测试

测试个人中心模块中的我的基本信息、修改头像、修改背景主题照片、查看我的博文、我的博文评论、我的说说的流程是否有问题，同事设置了几条不同的用例，系统是否会返回错误信息。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 测试目的 | 我的基本信息、修改头像、修改背景主题照片、查看我的博文、我的博文评论、我的说说整个流程是否通过 | | | | | | |
| 编写人 | 周晔 | 时间 | 2018-8-15 | 状态 | | 检测结果为合格 | |
| 前置条件 | 用户已登录账号 | | | | | | |
| 序号 | 测试步骤 | | | | 期待结果 | | 是否通过 |
| 1 | 用户在个人中心界面，点击我的资料 | | | | 操作成功，显示个人基本信息 | | 通过，预期结果一致 |
| 2 | 用户在个人中心界面点击个人头像，从本地电脑中选择图片，点击上传 | | | | 操作成功，显示头像上传成功 | | 通过，预期结果一致 |
| 3 | 用户在个人中心界面点击个人头像，从本地电脑中选择其他非图片文件，点击上传 | | | | 操作失败，显示头像上传失败 | | 未通过，预期结果一致 |
| 4 | 用户在个人中心界面点击个人个人背景图片，从本地电脑中选择图片，点击上传 | | | | 操作成功，显示头像上传成功 | | 通过，预期结果一致 |
| 5. | 用户在个人中心界面点击个人背景图片，从本地电脑中选择其他非图片文件，点击上传 | | | | 操作失败，显示个人背景图片上传失败 | | 未通过，预期结果一致 |
| 6. | 用户在主页点击我的个人中心标签，进入我的个人中心 | | | | 操作成功 | | 通过，预期结果一致 |

7.2.4我的博客模块测试

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 测试目的 | 写博客、查看博客板块、评论博客、整个流程是否通过 | | | | | | |
| 编写人 | 周晔 | 时间 | 2018-8-15 | 状态 | | 检测结果为合格 | |
| 前置条件 | 用户已登录账号 | | | | | | |
| 序号 | 测试步骤 | | | | 期待结果 | | 是否通过 |
| 1 | 用户在写博客界面选择博客分类、输入博客内容，点击完成 | | | | 操作成功，博客发表成功 | | 通过，与预期结果一致 |
| 2. | 用户在写博客界面选择博客分类、未输入博客内容，点击完成 | | | | 操作失败，博客内容不能为空 | | 不通过，与预期结果不一致 |
| 3 | 用户在他人博客页面评论区输入评论内容，点击评论 | | | | 操作成功，评论成功。 | | 通过，与预期结果一致 |
| 4 | 用户在他人博客页面评论区，未输入评论，点击评论。 | | | | 操作失败，未输入评论内容 | | 不通过，与预期结果不一致 |

7.2.5数据资源模块测试

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 测试目的 | 数据资源板块、添加我的资源、下载资源、整个流程是否通过 | | | | | | |
| 编写人 | 周晔 | 时间 | 2018-8-15 | 状态 | | 检测结果为合格 | |
| 前置条件 | 用户已登录账号 | | | | | | |
| 序号 | 测试步骤 | | | | 期待结果 | | 是否通过 |
| 1 | 用户在主页点击下载标签，进入下载板块 | | | | 操作成功 | | 通过，与预期结果一致 |
| 2 | 用户在添加我的资源页面，输入资源名称、选择资源分类、输入资源简介，选择上传资源 | | | | 操作成功，新增资源成功 | | 通过，与预期结果一致 |
| 3 | 用户在添加我的资源页面，输入资源名称、选择资源分类、输入资源简介，选择上传资源，以上操作有未做的 | | | | 操作失败，新增资源失败 | | 不通过，与预期结果一致 |

7.2.6说说圈模块测试

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 测试目的 | 浏览说说圈、写说说、删除说说 | | | | | | |
| 编写人 | 周晔 | 时间 | 2018-8-15 | 状态 | | 检测结果为合格 | |
| 前置条件 | 用户已登录账号 | | | | | | |
| 序号 | 测试步骤 | | | | 期待结果 | | 是否通过 |
| 1 | 用户在主页，点击说说圈板块 | | | | 操作成功 | | 通过，与预期一致 |
| 2 | 用户在说说圈板块首部输入说说内容，点击发布 | | | | 操作成功，发表成功 | | 通过，与预期结果一致 |
| 3 | 用户在说说圈板块首部未输入说说内容，点击发布 | | | | 操作失败，发表内容不能为空 | | 不通过，与预期结果一致 |
| 4 | 用户在个人说说列表右侧点击删除按钮 | | | | 操作成功，删除成功 | | 通过，与预期结果一致 |

7.3 本章小结

本章介绍了根据第三章的需求分析对各个模块功能进行测试的过程，其中测试包括功能性测试，用于测试每项功能点是否满足之前的需求；

# 第八章 工程能力培养

8.1 工程计划管控和执行情况

基本管理原则：每位成员既是积极的建言者，又是负责的合作者，同时也是决策的制定者。决策应在充分的讨论基础上由大家共同做出，一旦决策做出就必须被及时有效的执行。禁止再有异议。

目标 1：按时按量完成项目的基本功能，按时发布产品及文档，这是本团队的最高目标。

目标 2：遵循规范化的项目运作标准，文档严谨完整，代码注释充分，便于后续维护，这是第二目标。

目标3：产品运行稳定，界面友好，用户易操作，尽量从用户的角度去看问题，并提出解决问题的方案。

目标4：注重团队建设，成员分工合理，团队成员合作默契，气氛融洽。每周的讨论会积极建言。在开发过程中积极协作。

目标5：项目设计和开发上尽量有创新，有亮点。

实现目标的方案：

1. 每周例会上各位组员积极对当前的开发工作进行积极的评审和建言，由组长做最后的作口头总结。

2. 小组成员都要密切监控风险状态，发现风险后提交风险报告。并由组长做出临时处理决定。然后在该周的例会上由小组成员共同讨论对风险的处理意见。并形成风险处理的日志做为以后的经验。

3. 在项目进行的过程当中，组员之间应该多进行各种形式的非正式沟通，以使沟通更加的方便、快捷。

评审机制：每周例会上小组讨论形成一致意见后并，并邀请团长和其他组长参加评议。对于重大的风险处即为通过，相关负责人针对改进意见开展下一周工作，严格执行例会上所制定的决策。小组会议持续评估其成效。每一项目阶段结束之前（里程碑前后），组织一次阶段评审会，评估整个阶段的工作效率和成果质量。尽量与项目例会合理意见，应该由团长及其他组长组成评审团对处理意见进行审议和评估。并以评审团的决议作为重要参考来制定决策。

8.2 对本工程领域发展的认识

Web前端开发是从网页制作演变而来，名称上有很明显的时代特征。在互联网的演化进程中，网页制作是Web1.0时代的产物，早期网站主要内容都是静态，以图片和文字为主，用户使用网站的行为也以浏览为主。随着互联网技术的发展和HTML5、CSS3的应用，现代网页更加美观，交互效果显著，功能更加强大。

1. web前端的发展历程：

2005年以后，互联网进入Web2.0时代，各种类似桌面软件的Web应用大量涌现，网站的前端由此发生了翻天覆地的变化。网页不再只是承载单一的文字和图片，各种富媒体让网页的内容更加生动，网页上软件化的交互形式为用户提供了更好的使用体验，这些都是基于前端技术实现的。以前会Photoshop和Dreamweaver就可以制作网页，现在只掌握这些已经远远不够了。无论是开发难度上，还是开发方式上，现在的网页制作都更接近传统的网站后台开发，所以现在不再叫网页制作，而是叫Web前端开发。Web前端开发在产品开发环节中的作用变得越来越重要，而且需要专业的前端工程师才能做好，这方面的专业人才近几年来备受青睐。Web前端开发是一项很特殊的工作，涵盖的知识面非常广，既有具体的技术，又有抽象的理念。简单地说，它的主要职能就是把网站的界面更好地呈现给用户。

2. web前端开发的核心技术：

掌握HTML是网页的核心，是一种制作万维网页面的标准语言，是万维网浏览器使用的一种语言，它消除了不同计算机之间信息交流的障碍。因此，它是目前网络上应用最为广泛的语言，也是构成网页文档的主要语言，学好HTML是成为Web开发人员的基本条件。

学好CSS是网页外观的重要一点，CSS可以帮助把网页外观做得更加美观。

学习JavaScript的基本语法，以及如何使用JavaScript编程将会提高开发人员的个人技能。

了解Unix和Linux的基本知识虽然这两点很基础，但是开发人员了解Unix和Linux的基本知识是有益无害的。

3.要熟练掌握的技术：

熟悉版本控制系统的用法熟悉使用一种优秀的版本控制系统，你在实际工作中就会发现精通一种版本控制系统是很幸福的事情。

学好Web框架当你掌握了HTML，服务器端脚本语言，CSS和JavaScript后，就应该找一个Web框架加快你的Web开发速度，使用框架可以节约你很多时间，如果你使用PHP，可选的框架有CakePHP，CodeIgniter，Zend等，Python程序员喜欢使用Django和 webpy，Ruby程序员喜欢使用RoR。

对于初学开发人员来说，以上知识是不是看起来很复杂，其实，当你一步步学习并为企业开发成功案例时，你会发现一切都是有意义的付出。

4.未来的发展趋势：

经过近几年的发展，现代前端已经发展到跨端、跨界面的革新阶段，目前主流以基于MVVM、Virtual DOM、移动端MNV\*思路和前后端同构技术进行开发的项目居多，实现的方向也多种多样，这些我们前面对应的章节也均有讲到。当然除了这些，关于未来，还有一些我们前端工程师需要了解的，那我们就一起来看下未来前端具体可能会发展成怎样的呢。

5.新标准的进化与稳定：

前端新标准和草案在不断更新，HTML、CSS、JavaScript标准也在渐渐完善，尽管这些新的规范最终会淘汰旧标准的使用，新的项目也会以最新的标准作为开发依据，但要完全停止旧标准的使用并完成企业级旧项目的升级，依然需要一段时间。例如原有CoffeeScript的项目不可能一次性的做出迁移重构，但我们的项目仍需要维护，我们不能脱离实际项目去谈技术，这就需要一段时间来慢慢修改；再如Web Component现在也不会马上作为唯一标准大力推广。但可以肯定的是，新的语言或技术标准一定会被推广使用，只是还需要时间。

同时基于标准也会出现一些衍生的脚本语法和规范来适应特定的应用场景，这些非标准的规范除了解决具体业务技术问题之外，极有可能进化成下个标准的一部分或被新的标准借鉴。例如CoffeeScript虽然最终没有形成JavaScript开发标准，但EcmaScript 6却借鉴了其中很多优秀的特性；或者目前生成Virtual DOM的衍生脚本语法，未来也是有可能被列入到JavaScript标准当中的。

经过大版本的更新稳定，目前前端三层结构实现已经形成了HTML5、CSS3、EcmaScript 6+标准规范结合的阶段，后面标准的新变化也会越来越小，至少迄今为止，我们无法预见HTML6的到来、CSS4的特性目前也令人担忧、EcmaScript 7的特性更新也并不明显，这都显示出，目前前端项目实践规范将会相对稳定一段较长的时间，后面的修改不会像之前一样具有颠覆性，这也是技术标准发展到一定成熟阶段会发生的事情。

8.3 工程协作与交流

开发过程中，小组的分工明确，每周会找出一个时间进行例会，每个人说出自己这一周里的出现的未解决的问题和对于功能的不理解问题，大家一起讨论解决。

# 第九章 结束语

9.1 全文总结

本文从选题理由、课题背景、前期知识技术调研、需求分析、设计再到实现、测试介绍了此课题以及其完整开发流程：

1. 首先我对整个项目进行了需求分析和设计，为后续的开发提供了明确的方向和蓝图。

2. 完成需求分析和设计后，我针对设计中的每一个模块进行开发，每一个模块中的功能点逐个实现。

3. 登录模块实现了注册、激活、登录、找回密码功能，且测试通过。

4. 针对博客、资源分享、说说 、我们都实现了基本功能。

5. 在博客专栏、用户可以自由发表博客，浏览文章、评论文章。

6. 在资源专栏，用户可以尽情搜索自己喜欢的资源，也可与他人分享资源。

7.在说说功能板块，用户可以实时同步自己的状态。

9.2 不足与下一步工作

本系统的一个不足之处就是虽然在线资源分享有着与博客、网盘、动态圈的功能，但是与相对应的博客网、百度网盘、QQ空间动态，内容和功能都不是很成熟和详细，但是在线资源分享的前景极好，在以后可以对此系统进行完善，争取减小与那些成熟的系统差距。

前期技术方向没有选择好，导致后来走了很多弯路，希望在以后的学习生活中，多吸取之前的教训，总结，多听，多看，多学。

# 参考文献

1. 未来科技,中国水利水电出版社《HTML5+CSS3+JavaScript从入门到精通》.
2. 郭霖,人民邮电出版社《第一行代码——Android》.
3. 李刚,电子工业出版社出版《疯狂Android讲义》.
4. 国家863中部软件孵化器,人民邮电出版社《Java从入门到精通》.
5. 巴萨姆(BryanBasham),中国电力出版社《Head first servlets and jsp》.
6. (美)梁,机械工业出版社《Java 程序设计基础篇》.
7. (美)梁,机械工业出版社《Java 程序设计进阶篇》
8. (美)Joshua Bloch,机械工业出版社《Effective java中文版》
9. (美)Paul Deck,人民邮电出版社《Spring.MVC-A.Tutorial-Spring.MVC学习指南》

致谢

论文和系统都是是在王晓玲老师的精心指导下完成的。她帮我完善了报告的内容，帮助我修正了许多错误，并且对于系统的设计中提出了许多我没有注意到的问题，使系统的设计开发避免了许多的出错的地方，给予了我很大帮助。在此，我向王晓玲老师表示衷心的感谢。

当然，在这个以团队的形式开发的系统中，一个好的团队是开发的基础，所以我要感谢团队中的其他两个成员，谢谢他们在开发过程中给予我的帮助和支持，让我明白团队协作的重要，为以后进入社会奠定了基础。同时，我还要感谢我家人的支持，衷心感谢所有关心和帮助过我的人们，谢谢他们为我所做的一切。