

# 西安邮电大学

## 毕业设计（论文）

题目： 基于 SSH 整合框架的校内网系统的设计与实现

学院： 自动化学院

专业： 自动化

班级： 自动 1403 班

学生姓名： 王震

学号： 06141095

导师姓名： 侯雪梅 职称： 副教授

起止时间： 2017 年 12 月 5 日 至 2018 年 6 月 10 日

## 毕业设计（论文）声明书

本毕业论文《基于SSH整合框架的校内网平台体系的设计与实现》是本人在指导老师指点下自力钻研、写作的功效，论文中所援用别人的文献、数据、图件、资料均已明白标注；对本文的钻研做出重要贡献的小我和团体，均已在文中以明白文字注明并表示感谢。

本人彻底明白《西安邮电大学本科毕业设计（论文）管理办法》的各项规定并志愿遵照。

本人深知本声明书的法律责任，违规后果由本人承担。

论文作者签名：

日期：      年   月   日

## 西安邮电大学本科毕业设计(论文)选题审批表

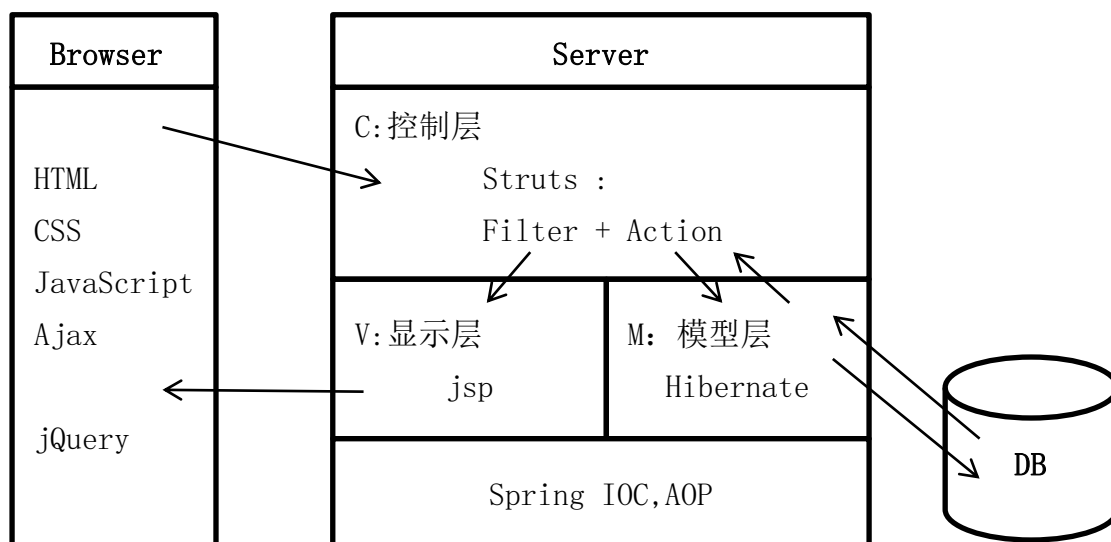
申报人	侯雪梅	职 称	副教授		学 院	自动化		
题目名称	基于 SSH 整合框架的校内网系统的设计与实现							
题目来源	科研				教学		其它	是
题目类型	硬件设计		软件设计	是	论文		艺术作品	
题目性质	应用研究		是		理论研究			
题目简述	随着互联网的兴起,打破了传统人类社交的地理限制条件,使得人们可以通过网络认识外面更加丰富多彩的世界。本课题意在通过使用 SSH 整合框架提供后端支持来向用户提供一个好玩的社交系统。							
对学生知识与能力要求	1. 熟练掌握 Spring, Struts, Hibernate 三者的原理及其整合。 2. 熟练掌握前端界面的大概设计过程及 HTML, CSS 等技术。 3. 熟悉 mybatis 数据库设计。 4. 熟悉 MyEclipse 开发工具的基本使用。							
具体任务以及预期目标	1. 掌握 SSH 整合框架的原理及其步骤。 2. 熟悉前端界面的设计过程。 3. 熟悉数据库的设计过程。 4. 基于 SSH 框架的校内网系统开发成果形式: 毕业论文、源程序清单。							
时间进度	2017. 12. 5 - 2018. 1. 15 查阅相关文献; 2018. 1. 16 - 2018. 2. 28 完成开题报告; 2018. 3. 1 - 2018. 4. 30 完成总体设计和各模块代码编写; 2018. 5. 1 - 2018. 5. 31 完成毕业论文的撰写; 2018. 6. 1 - 2018. 6. 7 准备毕业论文答辩 PPT; 2018. 6. 8 - 2018. 6. 9 准备毕业论文答辩。							
系(教研室)主任 签字	2017 年 12 月 9 日			主管院长 签字		2017 年 12 月 9 日		

## 西安邮电大学本科毕业设计（论文）开题报告

学号	06141095	姓名	王震	导师	侯雪梅
题目	基于 SSH 整合框架校内网系统的设计与实现				
<p>选题目的</p> <p>大学是人生一个比较美好的阶段，这个阶段每个人都希望在别人面前展示自己最好的一面，并且希望可以认识更多志同道合的同学，让自己变得更加优秀，但是传统的社交方式有着诸多的限制和不便。21 世纪进入互联网时代，随着网络的兴起，打破了人类社交的地理限制，人们可以在互联网上结交更多外地的小伙伴，可以在网上发现一些有趣的人和事，来丰富自己的生活阅历。</p>					
<p>前期基础</p> <p>已学课程：《高级程序语言设计》、《Java 语言程序设计》、《计算机网络》 《现代操作系统》、《数据结构与算法》</p> <p>掌握工具：Eclipse、IntelliJ IDEA、Tomcat、Oracle、Mysql、Maven、Redhat</p> <p>资料累积：《Java 程序设计》、《数据库设计》、《操作系统基础》</p> <p>软硬件条件：软件：IntelliJ IDEA、myeclipse、MySQL 数据库 硬件：一台内存 4G 并安装 jdk 环境的电脑</p>					
<p>要研究和解决的问题</p> <p>1. 开发框架整合问题</p> <p>Spring, Struts, Herbanate 三个框架都属于大型 Web 项目开发的必备框架，首先要明白这三个框架的基本原理以及其整合的要点。</p> <p>2. 搭建一个比较完整的开发环境的问题</p> <p>一个完整的开发环境影响着项目的开发效率以及代码的性能，学会部署 JDK 环境和 Maven 环境。</p> <p>3. 具体的开发流程问题</p> <p>需要自己事写好开发的流程工作，先做登录和注册功能，再做具体的社交模块，学会分模块去完成项目。</p>					

工作思路和方案

校内网系统结构如图所示：



校内网系统的核心部分在于服务器端的开发，服务器端采用 MVC 模式开发，其中使用 Struts 组件中的 action 作为核心控制器用于接收并处理浏览器端发送的各种请求。

第 1 周--第 2 周：在电脑上搭建软件开发环境以及所有的开发工具；

第 3 周--第 4 周：在编辑器上编辑测试环境的代码，对所搭建的环境进行测试；

第 5 周--第 6 周：初步完成用户的注册和登录两个模块的功能；

第 7 周--第 8 周：接着完成用户可以发布信息功能；

第 9 周--第 10 周：完成各用户之间的一个沟通交流的功能；

第 11 周--第 12 周：对整个项目的一个测试发布阶段；

第 13 周--第 14 周：撰写毕业论文，准备答辩。

指导教师意见

签字：

年 月 日

西安邮电大学毕业设计（论文）成绩评定表

学生姓名	王震	性别	男	学号	06141095	专业班级	自动 1403 班
课题名称	基于 SSH 整合框架校内网系统的设计与实现						
指导教师意见	评分（百分制）：                      指导教师(签字)： _____                      年    月    日						
评阅教师意见	评分（百分制）：                      评阅教师(签字)： _____                      年    月    日						
验收小组意见	评分（百分制）：                      验收教师(签字)： _____                      年    月    日						
答辩小组意见	评分（百分制）：                      答辩小组组长(签字)：                      年    月    日						
评分比例	指导教师评分    (%) 评阅教师评分    (%) 验收小组评分    (%) 答辩小组评分    (%)						
学生总评成绩	百分制成绩			等级制成绩			
答辩委员会意见	毕业论文(设计)最终成绩(等级)：  学院答辩委员会主任(签字、学院盖章)：                      年    月    日						

# 目 录

摘 要 .....	I
ABSTRACT .....	II
第一章 绪言.....	1
1.1 课题背景.....	1
1.2 课题研究现状.....	1
1.3 课题任务.....	1
1.4 论文结构.....	2
第二章 软件技术介绍.....	3
2.1 前端界面技术介绍.....	3
2.2 后端框架技术介绍.....	3
2.3 数据库技术介绍.....	4
第三章 软件设计概要.....	5
3.1 需求分析说明.....	5
3.1.1 完成的功能 .....	5
3.1.2 使用的技术要点 .....	5
3.2 uml 用例图 .....	5
第四章 功能实现.....	6
4.1 前端页面设计.....	6
4.1.1 首页面设计 .....	6
4.1.2 注册页面设计 .....	6
4.1.3 注册成功页面设计 .....	7
4.2 数据表设计.....	7
4.3 后端框架设计.....	10
4.3.1 创建 web 项目 .....	10
4.3.2 界面设计步骤 .....	10

4.3.3 建立数据库数据表 .....	12
4.3.4 整合 SSH .....	13
4.3.5 用户注册功能 .....	17
4.3.6 上传头像功能 .....	20
4.3.7 用户登录功能 .....	21
第五章 结束语 .....	22
5.1 论文总结 .....	22
5.2 未来工作 .....	22
致 谢 .....	23
参考文献 .....	24
附录 A：系统原理图 .....	25
附录 B：显示结果图 .....	25
附录 C：部分程序代码 .....	26



## 摘 要

大学阶段每个人都盼望在他人眼前展现自己最佳的一面，希望可以结交更多同窗好友，让自己变得更加的优异，可是传统的社交模式有着诸多的限制和不便，随着网络的兴起，打破了人类社交的地理限制，人们可以在互联网上结识更多的地理社交圈以外的小伙伴，所以本平台正是基于让更多的大学生可以认识更多的志同道合的优秀学生的目的，这样可以在网上发现一些比较有趣的人和事，来丰富自己的人生阅历。

针对上述现象，本课题设计出了一种可以让用户在网上发布自己的生活照片、自己的兴趣爱好以及自己对一些热点事件的看法等，本平台上的其他人可以关注并看见自己感兴趣的人的生活，相互观看对方对于自己的人生感悟以及对待日常生活的态度，以及这个世界上美好的人或事，让每一个人的生活都不再局限于自己的地理范围内，而是充分的可以结识自己身边以外的人。

本平台主要使用 Java 语言去完成，整个系统的后端体系采用 SSH 整合框架进行搭建，前端界面主要使用 HTML5、CSS 以及 jQuery 完成。

**关键词：**Spring; Java; Struts; Hibernate; 社交

## ABSTRACT

Everyone in the college stage is looking forward to showing their best in front of others, hoping to make more classmates and make themselves more outstanding, but the traditional social model has many limitations and inconveniences. With the rise of the Internet, Breaking the limits of human social geography, people can get to know more friends outside the geographical circle on the Internet, so this platform is based on the goal of allowing more students to know more like-minded and outstanding students. Find more interesting people and things on the Internet to enrich your life experience.

In view of the above phenomenon, this project has designed a way for users to publish their own photos of life on the Internet, their own interests and hobbies, and their opinions on some hot events. Others on this platform can pay attention and see what they are interested in. People's lives, mutual viewing of each other's attitude towards their own life and attitude towards daily life, and the beautiful people or things in this world, make each person's life no longer confined to their own geographical scope, but full of You can meet people outside of your side.

This platform is mainly implemented in Java language. The backend system of the entire system is built using the SSH integration framework. The front-end interface is mainly implemented using HTML5, CSS, and jQuery.

**Keywords:** Spring;Java; Struts;Hibernate;social

## 第一章 绪言

### 1.1 课题背景

进入 21 世纪，反观世界的每一次工业革命都会将世界带入一个全新的领域和全新的生活方式。第三次工业革命是以计算机信息技术为革命的，我们现在正处于第三次革命的浪潮之上，看着我们身边的世界在不断的更新换代之中，感叹技术带给我们生活上的便捷。

作为一个中国人，我们理应为了中国而发奋图强，反观每一次的工业革命都是从欧美日等国家开始的，并且每一次的发起国最后都能在新一轮的世界力量中占得头筹，并站在世界力量的巅峰，而中国则每次都是被接受国，所以我们更应该抓住信息技术的发展机遇奋力而上。21 世纪以来，互联网在中国大地发展迅猛，腾讯的社交应用、百度的搜索引擎技术以及阿里巴巴的电子商务为首。之前一直对社交网站特别感兴趣，于是本课题做一个社交网站为主题，旨在为我们提供更加便捷好用，可以分享自己的社会经历的社交网站。

### 1.2 课题研究现状

沟通与交流可能是人与人之间最不可缺少的一项活动了，人生而就有沟通交流的欲望，可是最早期的人们之间的社交仅限于一定地理范围以内的人，是地理限制了人类交流的更多可能性，随着互联网的蓬勃发展，各类社交软件层出不穷，最常见的有以下两种：

1. 以工作为重点的社交软件，比如各种的电子邮箱，可以从很远对的地方发送自己的工作邮件到指定联系人的邮箱里，这大大提高了工作效率。

2. 以生活娱乐交友为重心的社交软件可谓是百家争鸣，其中以微信和 QQ 最为出名，还有各类的交友网站，本着人与人之间的沟通交流，微信能时刻与自己的朋友发送消息，及时聊天，还能向好友分享自己的美好生活。

### 1.3 课题任务

本次课题的任务旨在建立一个简单的社交平台以及可以上传自己的日常生活照片并与好友分享自己的生活经验的平台。本次课题的后端采用的是 SSH (Spring, Struts, Hibernate) 整合框架，Spring 框架是一个已经趋于成熟的后端框架，它可以使后端做起来非常的方便简洁，主要起控制反转的作用，即将创建 bean 对象的

权利由人去创建改为了 Spring 框架自动的去创建 bean 对象;Hibernate 的主要作用是用于连接数据库并操作数据库来获取系统所需的数据。

本次课题是由前端界面，后端系统，数据库三者组成，前端界面负责向用户展示数据以及接收用户的数据，然后向后端系统反馈用户信息数据，后端处理之后经过从数据库的操作，然后向用户返回处理的结果页面并进行展示。

#### 1.4 论文结构

本文各章节的安排如下：

第一章：绪言：本次课题的课题背景以及研究现状。

第二章：软件技术介绍：本次毕设中所使用到的各种技术性的理论知识。

第三章：软件设计概要：简单描述本次毕设的软件设计概要。

第四章：功能实现：扼要描述重点功能的实现。

第五章：结束语：对毕设以及论文中的问题进行总结并得出心得体会。

## 第二章 软件技术介绍

### 2.1 前端界面技术介绍

1. HTML 技术: HTML 原本是部署于 WEB 服务器上, 经过前端浏览器的请求, 通过 HTTP 网络协议经过网络层下载到浏览器中, 并经由浏览器解释并显示执行。HTML 就是固定版的 XML, 它用各类标签、属性来展现网页内容的一种手段。

2. CSS 技术: CSS 是将 HTML 文本文件中的文字内容与其表现样式分隔开, 实现了代码的可重复性。网页设计上与页面显示效果有关的都是用 CSS 里面的属性来定义的, 比如: 页面上文段的高、宽等属性以及图片、表格等显示的效果。

3. JS 技术: JS 是一种网页语言。主要用于使 HTML 网页变成动态的效果。本来处于静止态的 HTML 网页就可以变成一种运动起来的效果, 提升了用户体验。

### 2.2 后端框架技术介绍

1. Spring 技术: 实质是用配置 Spring 文件来帮助管理 Java Bean 对象, 减少对象之间的耦合度。它提供了控制器端的 DispatcherServlet 和 Controller 两个核心的组件, 可以替代之前控制器端的 Servlet 的作用, 用起来会更加的简洁。Spring 中的核心技术就是 IOC 技术, 主要使用的是反射技术, 即根据在 Spring 的 xml 文件中的 <bean> 标签的配置, 可以按照 <bean> 标签设置的类的字符串名字去生成这个类的一个实例化对象并存入到 Spring 容器中去<sup>[3]</sup>。

2. Struts 技术: Struts 是一个轻量级的 MVC 框架, 即自己写的代码与这个框架的耦合度不高, 可以随时更换其他的框架去完成任务, 与 SpringMVC 相媲美。提供的最核心的组件就是 Action 组件, 此组件用于处理浏览器发送来的各种请求并进行一定的处理并向用户返回数据<sup>[4]</sup>。

3. Hibernate 技术: Hibernate 的产生是为了解决用 Jdbc 来写数据访问层代码的缺点。用 Jdbc 写代码, 是需要我们写大量的 SQL 语句的, 当我们使用 Jdbc 得时候, 就得要一会儿面向 Java 写代码, 一会儿又面向数据库写 SQL 语句的, 这样很麻烦, 而 Hibernate 恰好解决了这个问题, 可以自动的生成 SQL 语句, 是一种纯面向对象的数据访问层的技术并且还是一种可以跨数据库的技术<sup>[5]</sup>。

4. jsp 技术: jsp 是一种运行在 Server 端的并且可以实现动态化网页页面的技术。jsp 用于揭示数据成果, 数据成果揭示在浏览器的页面上。Hibernate 就是由访问数据库的代码所组成, 当 Hibernate 返回数据的时候, 然后会把这些数据交给 jsp 去显示。

5. Ajax 技术:用于改善用户体验的技术,其实质是向服务器发送异步的请求,服务器返回部分数据,整个过程,页面不会被刷新,更不会打断用户的操作,以此提升用户体验<sup>[9]</sup>。

## 2.3 数据库技术介绍

1. MySQL 数据库介绍: MySQL 数据库由于是开源免费的数据库软件,因此受到了大多数企业以及个人的使用,主要用于存放系统当中的所有数据的任务<sup>[2]</sup>。

2. SQL 语言: SQL 语句又分为标准版和方言版。SQL 语句里面有各种和数据库进行打交道的功能语言。当与数据库进行交流的时候,电脑相当于客户端,数据库相当于 Server 端<sup>[2]</sup>。

## 第三章 软件设计概要

### 3.1 需求分析说明

#### 3.1.1 完成的功能

1. 注册功能。
2. 登录功能。
3. 注销功能。
4. 更换头像功能。
5. 创建相册功能。
6. 上传图片功能。
7. 上传文件功能。

#### 3.1.2 使用的技术要点

SSH(Spring + Struts + Hibernate)整合框架 、HTML、CSS、MySQL。

### 3.2 uml 用例图

uml 用例图是一种用于描述体系功效的图，可以说明体系有什么样的角色。用户用例图如下图 3.1 所示：一个普通的用户具有注册，登录系统的功能，以及可以进行管理账号、管理自己的相册的功能。

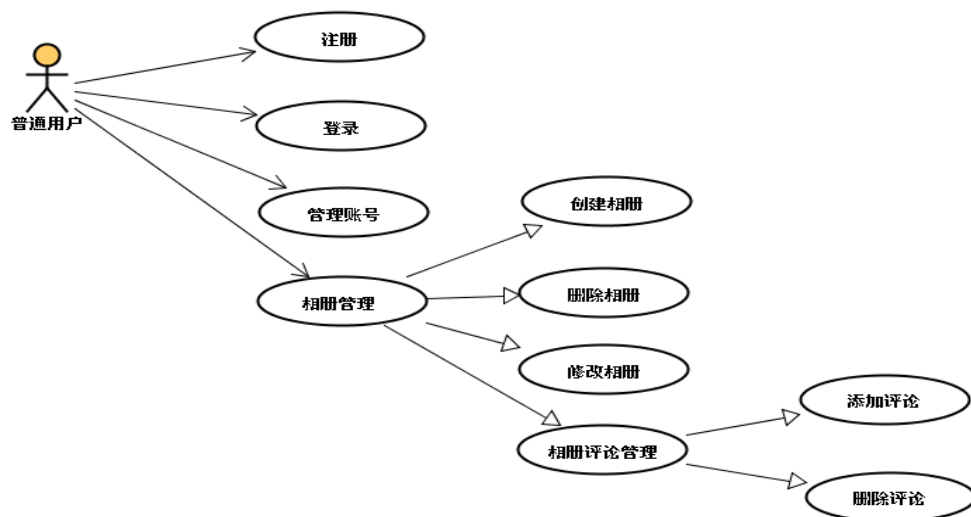


图 3.1 用户用例图

## 第四章 功能实现

### 4.1 前端页面设计

#### 4.1.1 首页面设计



图 4.1 首页面图

#### 4.1.2 注册页面设计

**简单一步，快速注册**

真实姓名:

性别: ☒ 男 ☐ 女

身份:

类型:

大学:

电子邮箱:

设置密码:

确认密码:

验证码: 请输入以下验证码，看不清楚一张? [1234]

中国 | 美国 | 加拿大 | 英国 | 澳大利亚 | 法国 | 德国 | 新西兰 | 新加坡 | 韩国 | 俄罗斯 | 日本 | 意大利 | 爱尔兰 | 荷兰 | 马来西亚 | 瑞士 | 泰国 | 乌克兰 | 南非 | 芬兰 | 瑞典 | 西班牙 | 比利时 | 挪威 | 丹麦 | 菲律宾 | 波兰 | 印度 | 奥地利 |

北京 | 上海 | 天津 | 重庆 | 黑龙江 | 吉林 | 辽宁 | 山东 | 山西 | 陕西 | 河北 | 河南 | 湖北 | 湖南 | 海南 | 江苏 | 江西 | 广东 | 广西 | 云南 | 贵州 | 四川 | 内蒙古 | 宁夏 | 甘肃 | 青海 | 西藏 | 新疆 | 安徽 | 浙江 | 福建 | 香港 | 澳门 | 台湾

• 西安交大	• 长安大学	• 西北工大
• 西北大学	• 陕西师大	• 西安电子科大
• 西安理工	• 西安科大	• 西安工大
• 西外	• 西邮	• 西安医学院
• 西安财经	• 西北政法	• 西安体育学院
• 西安美院	• 西安音乐学院	• 西安文理学院
• 西京学院	• 西安翻译学院	• 培华学院
• 欧亚学院	• 西安外事	• 西安石油
• 西安建筑科大	• 第四军医大学	• 西安电子科技大学高等职业技术学院
• 西北工业大学明德学院	• 陕西师范大学高等职业技术学院	• 长安大学兴华学院
• 延安大学	• 陕西中医学院	• 咸阳师范学院
• 陕西科大	• 宝鸡文理学院	• 渭南师范

图 4.2 用户注册页面图



### 4.1.3 注册成功页面设计



图 4.3 用户注册成功页面图

### 4.2 数据表设计

表 4.1 user 表

字段名	类型	约束	说明
u_id	bigint	primary key identity	用户 ID
u_email	varchar		邮箱
u_name	varchar		真实姓名
u_pwd	varchar		密码
u_photo	varchar	default 'default.gif'	头像
u_sex	char		性别(男, 女)
u_homePro	int	foreign key reference province(pro_id)	家乡省 ID
u_homeCity	int	foreign key reference city(ci_id)	家乡市 ID
u_birth	datetime		生日
u_tel	varchar		固定电话

u_mobil	varchar		手机号码
u_QQ	int		QQ 号
u_msn	varchar		MSN
u_website	varchar		个人网站
u_artistry	int	foreign key reference artistry(ar_id)	技校 ID
u_artistry Year	int		技校入学年份
u_juniorSc h	int	foreign key reference junior(ju_id)	初中 ID
u_juniorYe ar	int		初中入学年份
u_primaryS ch	varchar		小学
u_primaryY ear	int		小学入学年份
u_interest	varchar		兴趣爱好
u_music	varchar		喜欢的音乐
u_movie	varchar		喜欢的电影
u_game	varchar		喜欢的游戏
u_cartoon	varchar		喜欢的动漫
u_sport	varchar		喜欢的运动
u_book	varchar		喜欢的书籍
u_assn	varchar		加入的社团
u_visited	int	default 0	访问数
u_doing	varchar		状态
u_online	int	default 1	在线状态 (1- 在线, 2-离线)

表 4.2 country 表

字段名称	类型	约束	说明
co_id	int	Primary key identity	国家 ID
co_name	varchar		国家名

表 4.3 province 表

字段名称	类型	约束	说明
pro_id	int	Primary key identity	省份 ID
pro_name	varchar		省份名
pro_cout ryId	int	foreign key reference coutry(co_id)	国家 ID

表 4.4 city 表

字段名称	类型	约束	说明
Ci_ty	int	Primary key identity	城市 ID
Ci_name	varchar		城市名
Ci_proId	int	foreign key reference coutry(co_id)	省份 ID

表 4.5 city 表

字段名	类型	约束	说明
to_id	int	primary key identity	乡村 ID
to_name	varchar		乡村名
to_cityI d	int	foreign key reference city(ci_id)	所属城市 ID

## 4.3 后端框架设计

### 4.3.1 创建 web 项目

在 Myeclipse 中创建一个新的 JavaWeb 项目。

### 4.3.2 界面设计步骤

1. 在 WebRoot 下导入静态图片包：images 包。
2. 在 WEB-INF 下新建一个 public 包，用于存放一些整个项目中一些公共的 jsp 页面文件。
3. 首页面的头部分的设计如下图 4.4:

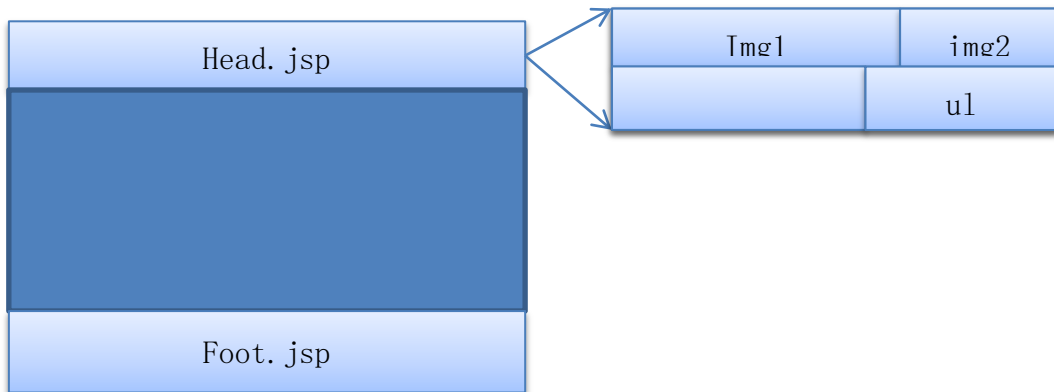


图 4.4 head 部分设计图

- (1) 针对于首页面,那么我们可以一步步的分析整个页面的布局,由 head. jsp、中间空白内容页面、foot. jsp 这三者构成。
- (2) 对于 head. jsp, 那么我们可以看出,它上面由两个图片构成,下左面也由一个图片构成,然后右下角由一个 ul 构成。
- (3) 分析 head. jsp 的布局之后,那么就可以动手去写具体的代码去实现了。
- (4) 具体的代码实现步骤如下所示:
  - 1) 新建一个 head. jsp, jsp 还需要自己的 CSS 样式文件,因此在 WebRoot 下新建一个 CSS 文件夹,然后在里面新建一个 head. css 文件。
  - 2) 当 head. css 文件写好之后,那么就可以直接将此 css 文件拖拽至 head. jsp 文件。
  - 3) 当然我们也可以将此新建的项目先部署在 Tomcat 上,然后启动 Tomcat,

那么通过浏览器来一边浏览我们所写的 head. jsp 的样式内，然后一边在修改。

- 4) 在 head. jsp 中根据自己开始所分析的布局去分别引入图片资源和写入 ul 代码。
- 5) 最终 jsp 页面的话只需要去刷新浏览器即可就可以查看最新的修改之后的 jsp 了。

5. 首页面的中间部分的页面设计如下图 4.5:

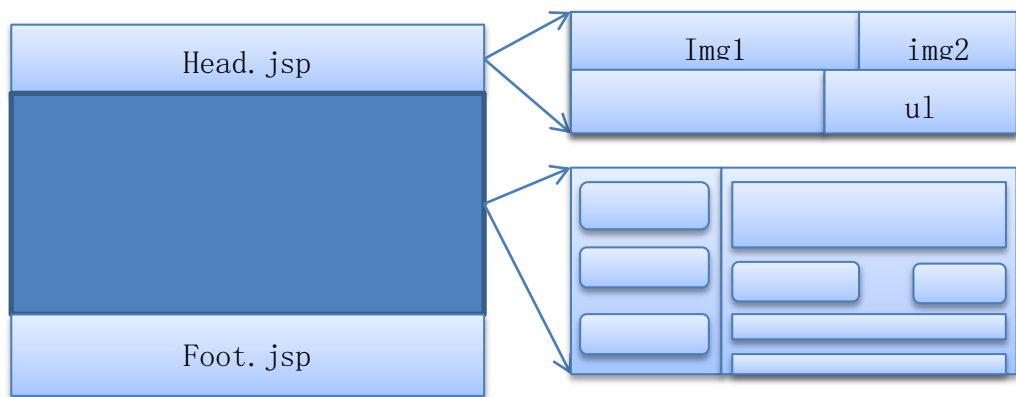


图 4.5 中间部分设计图

- (1) 中间部分按照上图显示看，全部设计为 div 块来设计。
- (2) 首先是左侧的竖着的用户注册以及登录块。
- (3) 右边横着的图片块。
- (4) 下面横着的文字块。

6. 用户注册页面的设计

(1) 注册界面:

- 1) 设计要点 1: 在本页面中的注册页面上选择不同人群的时候，那么不同人群所出现的注册信息输入框是有变化的，所以这个界面是很难写的，那么我们的策略是先将所有人群的输入框的信息全部考虑到位，然后全部先都写在这张页面上，那么后期的话我们可以利用 js 做事件函数处理，当点击某一个人群的时候，那么就让这个人群的那些输入框显示出来，同时让不是这个人群该有的输入框就隐藏起来即可。
- 2) 设计要点 2: 本页面中的身份选择框中只有 4 个数据，那么可以直接在 jsp 界面中写一个复选框就可以解决了； 但是在本页面中的当我们选择

学校的输入框时，那么因为全中国的学校简直太多了，我们不可能将这些数据存放在 jsp 页面中的，那么就必须得将这些数据存放到数据库中去了，因此设计到这里的时候，那么就必须得涉及到数据库了，但是 jsp 页面又不能去直接访问数据库，访问数据库的话只能通过搭建好的 Spring + Struts2 + Heribanate 环境框架去访问数据库的，所以说当这个界面设计到这里的时候，那么我们就得要开始去搭建 SSH 环境了，并搭建好数据库中的表数据，然后才可以继续去设计这个界面的。

- (2) 设计的过程，用户注册界面中的中间那部分用户信息的输入模块要用 table 表格来做，用表格来做的话，那么多的行列才能成行成列规整起来的，若用其他的标签做，界面会很乱，不易控制。

#### 7. Home 界面的设计

Home 界面原图为：如下图 4.6 所示



图 4.6 home 界面图

### 4.3.3 建立数据库数据表

1. 分析项目数据库表与表之间的关系,即业务逻辑。

- (1) 在设计用户注册的前端界面的时候遇到需要数据库的作用的，设计用户注册所需要用到的所有的数据表的关系。
- (2) 找出用户注册界面所需要的所有的数据库表，找出每张表都需要的字段。
- (3) 分析表与表之间的关系。

（4）分析哪张表，这张表又与哪些表有关系，又有着怎样的关系，就这样建立起来一个数 10 张数据表之间的关系网。

2. 向数据库中导入我们提前写好的各种数据表的 SQL 语句。

#### 4.3.4 整合 SSH

1. 引入 spring 的所有的 jar 包，并测试。引入 spring 的所有的 jar 包之后，要进行一些测试来证明所引入的这些 jar 包是正确的。可以发挥作用的 jar 包。

2. 引入的 spring 的 jar 包。

3. 写的测试类。

4. 配置 bean 零件。如下图 4.7 所示

```
<!-- 下面这句代码是测试代码： -->
<bean id="testS" class="com.xiaonei.test.TestService">
    <property name="name" value="ok"/>
</bean>
```

图 4.7 配置 bean 对象图

5. 运行测试类，看能否正确输出想要的结果。看到已经正确的输出了想要的结果：OK。说明 Spring 系列 jar 包是已经正确的导入并可以发挥 spring 的作用。

6. 在 src 下创建一个 beans.xml 文件。

7. 整合 Hibernate，整合 Hibernate 有两种方式：

（1）利用 eclipse 的导向组件： 关系模型→对象模型。先写好整个项目所有的数据表，再使用 MyEclipse 提供的工具去自动生成每一张数据表所对应的实体类对象 domain 即可。干系模子：就是数据库中的各种数据表及表之间的关系。组件模子：便是 Hibernate 框架中的表对应的各类实体类。

1) 连接数据库：

A. 点击 Windows + open perspective + MyEclipse database explorer 就可以打开数据库。然后在 DB Browser 框中右键单击选择 new 新建一个连接本地 MySQL 数据库的连接。

B. 点击 add jars，然后选中在 D 盘 lib 文件夹下的 MySQL 的驱动 jar 包，然后点击测试连接成功即可。

2) 在上一步中 MyEclipse 已经成功的连接到本地 MySQL 数据库了，接下来要导入 Hibernate 的 jar 包，但是只能使用 MyEclipse 中提供的工具去

导入 Hibernate 的 jar 包，否则我们使用不了 MyEclipse 的逆向工程。

- 3) 由于在上一步中我们是用 MyEclipse 中自带的工具导入的 Hibernate 的 jar 包，但是可能它导入的 Hibernate 的 jar 包不是我们想要的兼容的版本，那么我们可以先将它导入的版本包删掉，然后导入我们自己所需要的兼容的 Hibernate3.3 的 jar 包。
- 4) 使用 Eclipse 的导向工具，将数据表反射成 domain 实体类组件。一旦可以成功的使用 Eclipse 的工具映射自动生成数据表的 domain 对象的时候，那么证明 Hibernate 的 jar 包就导入成功了。
  - A. 将窗口切换到 DB Browser 窗口。
  - B. 在选择要使用逆向工程的数据表上右击，选择 Eclipse Reverse Enineering，按下图填好各参数后点击 next。如下图 4.8:

图 4.8 逆向工程参数图

- C. 按下图所示选择好 id Generator 选项为 increment 即可，type mapping 选择 java types 即可。如下图 4.9:

图 4.9 选择参数图

- D. 选择 next，并点击 finish 完成数据表的逆向反射。



E. 接下来将窗口切换到 java 窗口，然后去 com.xiaonei.domain 包下去看看是否逆向生成了刚刚那个数据表所对应的逆向实体类对象。

(2) 不使用逆向工程：对象模型--→关系模型。这种比较难，先在 MyEclipse 里面先写好各种数据表对应的实体类对象，然后在去映射出各种数据表，要是自己去写每个数据表的实体类的话，那么工作量会非常大的。

#### 8. 整合 Struts:

(1) 建议：使用 MyEclipse 提供的 Struts 的设计工具，需要通过 MyEclipse 导入 Struts 的 jar 包，然后再删除刚才 MyEclipse 自动为我们导入的 Struts 的 jar 包，然后再手动添加升级 Struts 的 jar 包即可。

(2) 使用 MyEclipse 中提供的工具去导入 Struts 的 jar 包。

(3) 由于 MyEclipse 给我们导入的 Struts 的 jar 包的版本可能不是我们想要的，所以我们要删除这些自动导入的 jar 包，如何删：在项目上右键单击，选择 properties，然后按照下图操作即可正确删除 jar 包。

(4) 删除 MyEclipse 为我们自动导入的 jar 包之后，我们就可以将我们自己的 Struts 版本的 jar 包复制一份到项目的 lib 目录下即可。

(5) MyEclipse 提供的 Struts 的设计工具：当我是通过 MyEclipse 导入的 Struts 的 jar 包的时候，那么 MyEclipse 就会自动的为我们生成一个 Struts 的设计工具，这个工具文件就是 struts-config.xml 文件。

#### 9. 单独测试 Struts

(1) 说明：到目前为止，我们已经成功的通过 MyEclipse 中的工具导入了 Spring，Hibernate，Struts 三者的所有 jar 包。接下来就需要做一些简单的测试，去测试我们导入的这些 jar 包是否可以正常使用。

(2) 编写一个测试程序，看看在没有把 Struts (action) 交给 Spring 容器管理的时候，Struts (action) 本身能否协同工作。

1) 测试案例链条：test.jsp(测试页面)--→\*.do 经过 web.xml(接收所有的 \*.do 请求)--→找到 struts 配置文件--→TestAction--→ShowCountry.jsp。

2) 在 WEB-INF 下新建一个 test 文件夹，专门用于写测试文件的。

3) 在 test 中新建一个 test.jsp 页面。

4) 在 struts 配置文件里面新建一个 Action：这步直接用点击选中窗口左侧的 Action—Add Action 按钮，然后再右边的窗口中点击一下，即可弹出

下框。

- 5) 然后再在 test 下新建并编写代码的 showCountry.jsp 文件。
- 6) 当 test.jsp 与 showCountry.jsp 两个页面都写完之后，那么直接用鼠标左键将这两个 jsp 拖拽至 struts-config.xml 之中即可，然后在用 struts-config.xml 设计工具的左侧的 Connection 按钮将这三者的连接起来，即可打通这次测试的所有环节。
- 7) 重新部署项目，启动 tomcat，更改 index.jsp 中的链接地址，然后打开浏览器访问本项目，看能否出现预期的结果页面。
- 8) 当我们没有将 Struts 托管给 spring 容器的时候，即 spring 容器没有自动创建 action 对象的权利的时候，那么我们就只能自己通过写代码的方式在 action 中创建一个 spring 容器对象，然后获取我们想要调用的 service 对象。一旦当我们后续将 Struts 托管给 spring 容器去管理的时候，那么上图中的那两句话就退出历史的舞台了。到时候我们可以直接使用的 spring 的 IOC 依赖注入的强大功能直接将要调用的 service 对象注入到对应的 action 中即可实现调用。

#### 10. 将 Struts 交给 Spring 去管理

- (1) 在 Struts 配置文件中书写请求处理器的代码：如下图 4.10 所示：

```
<!-- 配置请求处理器,就是把action的创建权交给spring容器去处理。就是*.do请求经过web.xml进入到本  
struts-config.xml文件，然后看所想要接收请求的action的path名，然后就直接去spring容器中去找与此  
path名相等的action，然后spring容器并创建该action对象。  
-->  
<controller>  
    <set-property property="processorClass"  
        value="org.springframework.web.struts.DelegatingRequestProcessor"/>  
</controller>
```

图 4.10 action 示例图

- (2) 在配置文件 beans.xml 中配置 registerAction 的 bean 代码如下图 4.11 所示：

```
<!-- 配置action -->
<bean name="/test" class="com.xiaonei.web.action.TestAction"/>

<bean name="/register" class="com.xiaonei.web.action.RegisterAction">
  <property name="countryService" ref="countryServiceImpl" />
  <property name="provinceService" ref="provinceServiceImpl"/>
  <property name="universityService" ref="universityServiceImpl"/>
  <property name="userService" ref="userServiceImpl"/>
  <property name="useruniversityService" ref="useruniversityServiceImpl"/>
</bean>
```

图 4.11 action bean 配置图

- (3) 接下来使用 ioc 依赖注入在每一个 action 中都注入器所需要的 service 业务层的 bean。但是要确保一点：确保 Struts 配置文件中 action 的 path 属性值要与 spring 的配置文件 beans.xml 文件中的 action 的 bean 的名称属性值相同。
- (4) 当我们修改了 action 的时候，是不用去重新部署项目的。
- (5) 将 Spring3.3、Struts1.3、Hibernate3 三者成功的整合在一起了。

#### 11. 添加一个新功能的开发步骤：

- (1) 先写 jsp 页面。
- (2) 用 Struts 配置文件将线路打通。先不考虑数据库。
- (3) 写 action 和 service，并将这两者配置到 spring 容器中去，此时考虑数据库。
- (4) 综合测试看是否成功。

### 4.3.5 用户注册功能

1. 注册界面跳转至 home 界面过程简介：由注册界面中 table 表格外面的 form 表单的提交按钮，然后将请求提交给 RegisterAction，经 action 处理之后，则跳转至网站个人主页，即 home.jsp 界面显示。用户注册过程如下图 4.12 所示：

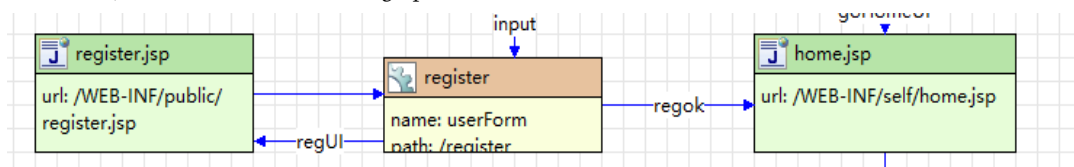


图 4.12 用户注册流程图

2. register.jsp 界面，register.jsp 中的 table 表格外面的 form 表单的提交按钮。如下图所示：当表单提交之后，则跳转至的 action 的 URL=/xiaoneinew/register.do。即要去 struts-config.xml 中去找 register.do

那个 action，找其中方法名是 flag=regUser 的那个方法。如下图 4.13 所示：

```
<form action="/xiaoneinew/register.do?flag=regUser" method="post">
    <table> <!-- 这里面可以加上属性border="1"来设置表格中的小格子的样式的。 -->
</form>
```

图 4.13 表单配置图

3. RegisterAction 的书写，其中的一个方法名为 regUser 的函数。如下图 4.14 所示：

```
//注册用户,真的添加到数据库.
public ActionForward regUser(ActionMapping mapping, ActionForm form,
    HttpServletRequest request, HttpServletResponse response) {

    //从该action对应的表单中,取出数据
    //注册成功,则跳转到home.jsp
    return mapping.findForward("regok");
}
```

图 4.14 action 代码图

4. home.jsp 界面，在上面的 RegisterAction 中的 regUser 方法的最后是返回了一个“regok”的字符串，这个字符串刚好对应于 struts-config.xml 中 RegisterAction 与 home.jsp 之间的连接线之上的字符串。
5. 实现真正的用户注册功能：
- (1) 之前只是实现了由注册界面到 home 界面的连通，但是并没有实现将用户注册的数据提交到数据库中去，接下来实现这个功能。
  - (2) 在 Struts 配置文件中右击 registerAction 组件，然后点击 new 新建一个名为 UserForm 的 Form 表单类。这个 form 表单中的属性都是一个新注册的用户可能所要具有的属性，到时候用户注册填的信息都由这个 UserForm 表单来存储。
  - (3) 将 UserForm 表单关联到对应的 registerAction，在 Struts 配置文件中的 registerAction 组件上右击选择 Properties，然后选择 Form，然后填入刚刚自己新建的那个 UserForm。将表单 form 与 action 关联起来之后呢，那么当 jsp 页面中的表单向此 action 中传数据的话，那么就直接将数据传入到与这个 action 相关联的那个 Form 对象中去存着了。
  - (4) 回到 register.jsp 页面中用户注册时点击提交按钮之后向哪里发起的请求，可看到是由 registerAction 中的 regUser 方法接收此次处理。

- (5) 编写 registerAction 中的 regUser 类中的代码。
6. 保存数据表字段的默认值。用户注册的时候，会发现最后插入到用户表中的有默认值的字段都没有默认值，那么这个时候我们只需要在 domain 包下的对应的 xxx.hbn.xml 中的 class 标签里面加上一个属性：dynamic-insert="true"，这样的话那么就可以在注册用户时，对应数据表里面字段的默认值就可以存到数据表里了。
7. 用户注册功能 2——由登录界面跳转至注册界面。在这个由 index.jsp 界面跳到用户注册界面的过程中，貌似看是不需要经过任何的 action 处理数据就可以直接跳转过去的，但是事实上，当我们在从登录界面跳转至注册界面的过程中，是需要提前准备一些数据的，比如当用户点击输入框时，就会自动弹出一些可供用户选择的数据或下拉框，那么这些数据肯定不可能是当用户点击输入框的那一刻去数据库提取得，那样会导致整个界面的效率非常低，在实际开发的时候，那些界面弹出的数据都是由一个界面跳转至另一个界面的过程中，经由一个 action 中的一个方法去提取从数据库中获取到，然后当界面进入注册界面的时候，那么当用户点击输入框的时候，那么数据框就会立马弹出，这样不会影响用户注册时的界面反应时间，提示用户的使用效果。
8. 由登录界面跳转至注册界面实现步骤：
- (1) 在 RegisterAction 中写上第二个方法：regUI。此类是 index.jsp 跳至注册界面时所要经过的 action 中的又一个处理方法：regUI。
  - (2) 在 Struts 配置文件中先写好页面到页面之间的流程配置好。
  - (3) 在 index.jsp 中的 head.jsp 中的“注册”超链接中加上 href="/xiaoneinew/register.do?flag=regUI"。路径指向了登录 action，并会去访问这个 action 里面名为 regUI 的方法。
  - (4) 写 registerAction 类的代码。
  - (5) 打通界面之后，那么当我们点击注册界面中的大学选择框的时候，是要弹出一个大学选择弹框的，那么这个弹框怎么写呢？这个弹框肯定是早已经在 register.jsp 界面上写好代码的，只是设置了隐藏功能而已，当用户注册时一点击就会变成显示立马弹出的。
  - (6) register.jsp 界面上有了这个弹框后，起初是没有数据的，我们要去在 registerAction 中的 regUI 这个方法中写代码去调用指定的 regService 去数据库中去获取数据的。这个时候我们可以通过分析写什么来获取我们想要获取到的数据：我们要分别获取所有的国家、省份或直辖市、大学。

(7) 获取所有的国家数据，那么操作的是 Country 数据表，对应 Hibernate 中的 domain 对象是 Country.java。然后操作 Country 对象肯定是需要有一个 CountryServiceInter 接口的，必然就需要一个 CountryServiceImpl 实现类的。这样下去这个 registerAction 中的 regUI 这个方法中肯定同时需要国家表，省份表，大学表这三张表的 Service 的接口和实现类的。那么这样的话这三张表肯定是有有一些公共的代码的，所以对于这种具有一些公共代码的类，将其公共的部分写成一个基类接口和基类的实现类。

#### 9. 界面转至注册界面之后注册界面的一些操作

- (1) 在由登录界面跳转至注册界面的过程中，虽然要经过一个 action 中的一个方法去数据库中获取默认要显示的所有国家，默认中国的所有省份，默认北京市中的所有大学，但是我们并不是这一次就将每一个国家的省份都取了出来，也不是将全世界所有的大学都取了出来，我们只是取出来了默认要显示的国家，省份，大学而已。但是当我们在注册界面的时候，点击选择大学的那个弹框的时候，那么需要可以弹出任何国家任何地方的大学的，那么这样的话就需要我们的 ajax 技术去实现这个功能的。
- (2) Ajax 简介：ajax 可以认为是一种临时异步的在界面上从后端获取数据并将数据显示到前端界面上的一种技术。
- (3) 本模块中 Ajax 的写法步骤：以用户随便点击一个中国的省份，然后显示这个省份的大学名为例。
  - 1) register.jsp 中的省份框，点击一个省份，触发 onclick 函数。
  - 2) 创建一个 ajax 文件，然后在里面写 ajax 函数代码。
  - 3) 将 ajax 文本文件引入此登录的 jsp 文件中
  - 4) 在 2) 中我们看到此 ajax 是发送了一个请求给 ChangeInfo 这个 servlet，所以我们还需要在项目 src 下新建一个包来存放 servlet。
  - 5) 接下来在 servlet (ChangeInfo) 中书写用于从数据库中去获取本次用户所点击的那个省份下的所有的大学数据。
  - 6) 最后在那个 ajax 文件下写一个函数方法，将本次获取到的大学数据拼成一个 table，table 去替换掉原先默认显示的那些大学的 table。

#### 4.3.6 上传头像功能

1. 将即将要上传的头像更新掉原用户表中的 photo 字段数据值。用户的头像图片存

储的位置其实就是数据库中用户那张表的固有字段照片字段中的值，那里面的值只是存储了这个图片的名字而已。所以上传头像其实就是要将用户表中用户默认的 photo 照片替换成我们自己选的照片，所以这是一个 update(修改) sql 语句。

2. 将该用户的头像上传至服务器。

- (1) 要能自动的在 Server 端自动的生成一定目录结构的用于存储用户头像图片的结构，我们设想给每个用户都弄一个固定的结构目录，那么最后所有的头像我们就不会弄混了。
- (2) 先在项目的 WebRoot 根目录下新建一个用于存储本项目所有需要的工具方法的工具类的 util 包，这个包下可以存储好多的工具方法，进而供我们在写业务逻辑代码的时候使用。并书写 MyTools.java 类。
- (3) 有的时候我们可能会经常换头像，那么就会导致一个用户编号目录下会有多张头像图片，这样会占用内存空间资源的，所以就产生了一个问题，如何在我们更新头像之前，先将此用户目录下之前的头像图片全删掉呢？解决办法：我们可以将每次上传的头像名都更改为用户 ID.jpg 的形式，那么就会产生覆盖的，即这次的头像覆盖掉上次的头像的。

3. Struts 中文件的上传与下载。

#### 4.3.7 用户登录功能

1. 写用户登录的界面。
2. 用 Struts 配置文件将线路打通。
3. 写用户登录 action 和用户登录的 service，并将这两者配置到 spring 的配置文件中即可，并考虑调用数据库去 获取相关的数据并返回显示。
4. 最后在综合测试看一下是否成功。

## 第五章 结束语

### 5.1 论文总结

本次毕设课题以开发一个具有自己特色的社交网站平台为宗旨，基本完成了用户登录，注册以及注销的功能，还完成了重置用户头像的功能。

遇到的困难及其解决办法：SSH 三大框架的整合问题：在这三大框架的整合过程中，应该是最难解决的地方，其中出现了整合错误的地方，导致程序不能正常衔接运行。最后经过检测是写 spring 配置文件的时候不小心导致 XML 配置写错；MySQL 数据库查数据出现中文乱码的问题：最后经过自己的百度搜索以及询问老师和同学最后发现原来是自己设置的 MySQL 数据库的字符编码有误，最后将 `character_set_client`、`character_set_connection`、`character_set_results` 三者由 UTF-8 改为 GBK 之后，那么数据库就可以正常插入中文数据已经正常查出中文数据了；前端界面的设计过程中出现了自己不会的问题：由于自己一直主要从事后端的设计，所以对于前端不是特别懂，但是在做一个完整的项目的时候确实是要各方面的知识点都要会一点点的，于是自己就通过询问同学以及百度搜索等知识学习了些前端设计的简要代码设计，设计出的界面也不是很美观，但是基本上可以实现最本质的要求。

心得体会：本次毕设完全是自己一个人完成的 JavaWeb 项目，基本上完成了需要的功能，但是由于时间以及自己学识上的有限，仅仅只是完成了一部分功能，虽然还有一部分功能没有具体的实现出来，但是基本上的框架，大体上的思路在自己的脑海中以及形成了，那么实现只是时间上的问题了。经过本次毕设学习到了前端界面如何设计，数据库如何搭建，Java 网站平台后端框架环境如何搭建，可以说是系统的学习到了一个系统从零到一再到十的过程。

### 5.2 未来工作

随着社交平台系统的不断发展，相信以后还会产生更加完美功能的社交系统出来的，自己对于软件编写有一定的兴趣爱好，希望可以用代码编织出自己大脑中对于这个世界的一些像发。



## 致 谢

在此感谢侯老师对我的毕设的指导，侯老师经常主动与我交流我的毕业设计的主要思想以及我自己大脑中对于这个课题的理解程度。在侯老师悉心的指导下，我完成了毕业设计和论文。同时感谢同学对我的帮助，有了同学的帮助，有些问题便迎刃而解了。最后感谢自己的两位好朋友，在遇到一些技术难题的时候，在他们的帮助下，技术提升很快。

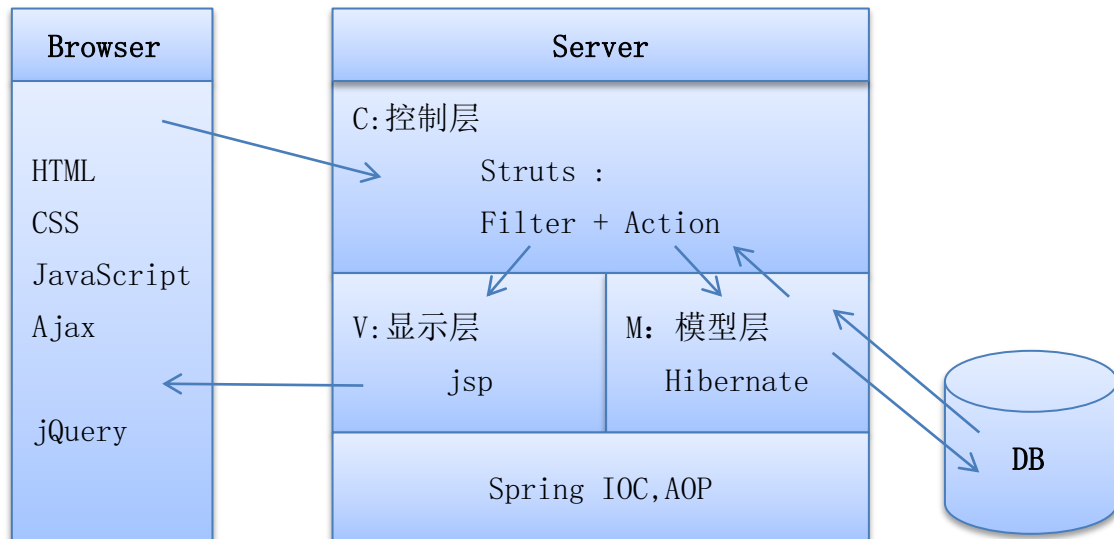
在此临近毕业之际，感谢母校给予自己学习与提升的环境，感谢遇到了许多如侯老师一样的老师，感谢遇到了好多志同道合的好朋友。人生在世，若能时刻心存感激之情，方能时刻感受社会的美好，更加努力的回馈国家与社会。

## 参考文献

- [1]钱乐秋. 软件工程[M]. 北京：清华大学出版社. 2017.
- [2]王珊，萨师煊. 数据库系统概论[M]. 北京：高等教育出版社. 2015.
- [3]陈雄华. Spring 企业级应用开发详解[M]. 北京：电子工业出版社. 2017.
- [4]李刚. Struts2 权威指南[M]. 北京：电子工业出版社. 2016.
- [5]陶勇，李晓军. Hibernate ORM 最佳实践[M]. 北京：清华大学出版社. 2015.
- [6]程杰. 大话设计模式[M]. 北京：清华大学出版社. 2013.
- [7]雷之宇，朱训雨，张麟. Java 实用组件集[M]. 北京：电子工业出版社. 2016.
- [8]熊节，林仪明，张凯峰，陈玉泉. 主编[《CSDN 社区电子杂志—Java 杂志》创刊号]
- [9]孙鑫. Java Web 开发详解[M]. 北京：电子工业出版社. 2014.
- [10]蔡剑，景楠. Java Web 应用开发：J2EE 和 Tomcat[M]. 北京：清华大学出版社. 2013.
- [11]Javey.S. “CONCEPT OF ‘CORRESPONDENCE’ IN JSP”,Proceedings of the Hawaii International Conference on System Science,Vol2,2013.
- [12]W.Clay,Richardson,Donald, “Avondolio.The Java high class weves a distance : JDK 5”,Scientific & Technology Book Review,No.3,2014.
- [13]Markus Aleksy,Axel Korthaus, “Martin Schader.Use Java and CORBA realization distribute type system”, Journal of Pingxiang College,No.4,2013.

## 附录 A

系统原理图：



## 附录 B

显示结果图：

The screenshot shows a registration interface. On the left is a registration form titled "简单一步，快速注册" (Simple one step, fast registration). The form includes fields for: 真实姓名 (Real Name) with value "张三", 性别 (Gender) with radio buttons for 男 (Male) and 女 (Female), 身份 (Identity) with a dropdown menu showing "已工作" (Already working), "大学生" (University student), "中学生/中专技校" (High school student/Vocational school), and "以上都不是" (None of the above), 类型 (Type) with a dropdown menu showing "大学生" (University student), 大学 (University) with value "西邮" (Xiyou), 电子邮箱 (Email), 设置密码 (Set password), and 确认密码 (Confirm password). Below the form is a 验证码 (Captcha) field with the text "请输入以下验证码，看不清楚一张？" (Please enter the following captcha, can't see clearly one more?) and a "免费注册" (Free registration) button. On the right is a list of universities and colleges, including: 中国, 美国, 加拿大, 英国, 澳大利亚, 法国, 德国, 新西兰, 新加坡, 韩国, 俄罗斯, 日本, 意大利, 爱尔兰, 荷兰, 马来西亚, 瑞士, 泰国, 乌克兰, 南非, 芬兰, 瑞典, 西班牙, 比利时, 挪威, 丹麦, 菲律宾, 波兰, 印度, 奥地利, 北京, 上海, 天津, 重庆, 黑龙江, 吉林, 辽宁, 山东, 山西, 陕西, 河北, 河南, 湖北, 湖南, 海南, 江苏, 江西, 广东, 广西, 云南, 贵州, 四川, 内蒙古, 宁夏, 甘肃, 青海, 西藏, 新疆, 安徽, 浙江, 福建, 香港, 澳门, 台湾, 西安交大, 长安大学, 西北工大, 西北大学, 陕西师大, 西安电子科大, 西安理工, 西安科大, 西安工大, 西外, 西邮, 西安医学院, 西安财经, 西北政法, 西安体育学院, 西安美院, 西安音乐学院, 西安文理学院, 西京学院, 西安翻译学院, 培华学院, 欧亚学院, 西安外事, 西安石油, 西安建筑科大, 第四军医大学, 西安电子科技大学高等职业技术学院, 西北工业大学明德学院, 陕西师范大学高等职业技术学院, 长安大学兴华学院, 延安大学, 陕西中医学院, 咸阳师范学院, 陕西科大, 宝鸡文理学院, 渭南师范, 延安大学, 咸阳师范学院, 渭南师范. A "关闭窗口" (Close window) button is at the bottom right.

图 1：注册界面图



图 2: 上传图像界面图

## 附录 C

部分程序代码:

```
<?xml version="1.0" encoding="utf-8" ?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:context="http://www.springframework.org/schema/context"
    xmlns:aop="http://www.springframework.org/schema/aop"
    xmlns:tx="http://www.springframework.org/schema/tx"
    xsi:schemaLocation="http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans-2.5.xsd
        http://www.springframework.org/schema/context
        http://www.springframework.org/schema/context/spring-context-2.5.xsd
        http://www.springframework.org/schema/aop
        http://www.springframework.org/schema/aop/spring-aop-2.5.xsd
        http://www.springframework.org/schema/tx
        http://www.springframework.org/schema/tx/spring-tx-2.5.xsd">
    <bean id="testS" class="com.xiaonei.test.TestService">
        <property name="name" value="ok"/>
    </bean> -->
    <context:annotation-config/>
    <!-- <context:component-scan base-package="com"/> -->
    <bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource"
        destroy-method="close">
        <property name="driverClassName" value="org.gjt.mm.mysql.Driver"/>
```

```

        <property name="url"
value="jdbc:mysql://localhost:3306/xiaonei?useUnicode=true&characterEncoding=utf-8"
/>

        <property name="username" value="root"/>
        <property name="password" value="root"/>
        <property name="initialSize" value="3"/>
        <property name="maxActive" value="500"/>
        <property name="maxIdle" value="2"/>
        <property name="minIdle" value="1"/></bean>
    <bean id="sessionFactory"
class="org.springframework.orm.hibernate3.LocalSessionFactoryBean">
        <property name="dataSource" ref="dataSource"/>
        <property name="mappingResources">
            <list>
                <value>com/xiaonei/domain/City.hbm.xml</value>
                <value>com/xiaonei/domain/Country.hbm.xml</value>
                <value>com/xiaonei/domain/Junior.hbm.xml</value>
                <value>com/xiaonei/domain/Primaryschool.hbm.xml</value>
                <value>com/xiaonei/domain/Province.hbm.xml</value>
                <value>com/xiaonei/domain/Senior.hbm.xml</value>
                <value>com/xiaonei/domain/Techschooll.hbm.xml</value>
                <value>com/xiaonei/domain/Town.hbm.xml</value>
                <value>com/xiaonei/domain/University.hbm.xml</value>
                <value>com/xiaonei/domain/Usernet.hbm.xml</value>
                <value>com/xiaonei/domain/Users.hbm.xml</value>
                <value>com/xiaonei/domain/Usersenior.hbm.xml</value>
                <value>com/xiaonei/domain/Useruniversity.hbm.xml</value>
                <value>com/xiaonei/domain/Album.hbm.xml</value>
                <value>com/xiaonei/domain/Photo.hbm.xml</value>
                <value>com/xiaonei/domain/Photocomment.hbm.xml</value>
            </list>
        </property>
        <property name="hibernateProperties">
            <value>
                hibernate.dialect=org.hibernate.dialect.MySQLDialect
                hibernate.hbm2ddl.auto=update
                hibernate.show_sql=false
                hibernate.format_sql=false
            </value>
        </property>
    </bean>

```

```

        </property></bean>
    <bean id="testService" class="com.xiaonei.service.TestService">
        <property name="sessionFactory" ref="sessionFactory"/></bean>
    <bean id="countryServiceImpl" class="com.xiaonei.service.impl.CountryServiceImpl">
        <property name="sessionFactory" ref="sessionFactory" /></bean>
    <bean id="provinceServiceImpl"
class="com.xiaonei.service.impl.ProvinceServiceImpl">
        <property name="sessionFactory" ref="sessionFactory"/></bean>
    <bean id="universityServiceImpl"
class="com.xiaonei.service.impl.UniversityServiceImpl">
        <property name="sessionFactory" ref="sessionFactory"/></bean>
    <bean id="userServiceImpl" class="com.xiaonei.service.impl.UserServiceImpl">
        <property name="sessionFactory" ref="sessionFactory"/></bean>
    <bean id="useruniversityServiceImpl"
class="com.xiaonei.service.impl.UseruniversityServiceImpl">
        <property name="sessionFactory" ref="sessionFactory"/></bean>
    <bean id="albumServiceImpl" class="com.xiaonei.service.impl.AlbumServiceImpl">
        <property name="sessionFactory" ref="sessionFactory"/></bean>
    <bean name="/test" class="com.xiaonei.web.action.TestAction"/>
    <bean name="/register" class="com.xiaonei.web.action.RegisterAction">
        <property name="countryService" ref="countryServiceImpl" />
        <property name="provinceService" ref="provinceServiceImpl"/>
        <property name="universityService" ref="universityServiceImpl"/>
        <property name="userService" ref="userServiceImpl"/>
        <property name="useruniversityService" ref="useruniversityServiceImpl"/></bean>
    <bean name="/login" class="com.xiaonei.struts.action.LoginAction">
        <property name="userService" ref="userServiceImpl" /></bean>
    <bean name="/profile" class="com.xiaonei.web.action.ProfileAction">
        <property name="userService" ref="userServiceImpl"/></bean>
    <bean name="/album" class="com.xiaonei.struts.action.AlbumAction">
        <property name="albumService" ref="albumServiceImpl"/></bean>
    <bean id="txManager"
class="org.springframework.orm.hibernate3.HibernateTransactionManager">
        <property name="sessionFactory" ref="sessionFactory"/></bean>
    <tx:annotation-driven transaction-manager="txManager"/>
</beans>

```

