**丽 水 学 院**

**毕业设计（论文）**

**（2018届）**

**题 目 基于SSM框架的实习生招聘网开发**

**指导教师 沈伟华**

**院 系 工学院**

**专 业 计算机科学与技术**

**班 级 计142**

**学 号 14103330129**

**姓 名 巫坤**

2018年4月15日

基于SSM框架的实习生招聘网开发

专业：计算机科学与技术 班级：计142 姓名：巫坤 指导教师：沈伟华

摘要 实习生招聘网站是提供职位发布，职位搜索，简历投递一系列服务的平台。同时为了让用户有更多的自主性，提供了实习生用户模块、HR用户模块、后台管理系统三大系统，以供三类用户使用。为了保证以后网站运行的稳定性，选取合适的运行服务器则显得非常重要。Spring Mvc是一款优秀的MVC的开发框架，强大的性能和易用性，为网站在服务器端稳定安全地运行提供了保障；Spring对所有bean进行管理，并且与Mybatis整合。Require.js前端框架则实现了前端js模块化管理，有效解决js引用代号冲突问题，更加清晰统一化管理依赖js。

本论文介绍了基于spirng mvc+Spring+mybatis框架开发的实习生招聘网设计与实现，本项目以IDEA+Jetty+Mysql5.0为开发平台。

关键词Spring Mvc；Spring；Mybatis；框架

**Development of intern recruitment network based on SSM**

Major: Computer Science and technology class: 142 names: WuKun Instructor: ShenWeihua

**Abstract** The intern job site is a platform for job Posting, job search, and resume delivery. At the same time, in order to give users more autonomy, three systems of intern user module, HR user module and background management system are provided for the use of three types of users. In order to ensure the stability of the website, it is very important to select the proper running server. Spring Mvc is a good Mvc development framework with strong performance and ease of use, which provides guarantee for the stable and safe operation of the website on the server side. Spring manages all beans and integrates with Mybatis. The front-end framework of Require. Js implements the front-end js modularization management, effectively solves the conflict problem of js reference code, and makes it more clear and unified management dependence js.

This paper introduces the design and implementation of intern recruitment network based on spirng MVC +Spring+mybatis+redis framework. This project is a development platform with IDEA+Jetty+Mysql5.0.

Key Words**：**Spring Mvc；Spring；Mybatis；Framework

目 录

[引 言 3](#_Toc512093563)

[第一章 概述 4](#_Toc512093564)

[1.1 开发的背景 4](#_Toc512093565)

[1.2 系统功能概述 4](#_Toc512093566)

[1.3开发环境介绍 5](#_Toc512093567)

[第二章 需求分析 8](#_Toc512093568)

[2.1 用户需求分析 8](#_Toc512093569)

[2.1.1 实习生用户模块需求 8](#_Toc512093570)

[2.1.2 HR用户模块需求 8](#_Toc512093571)

[2.1.2 后台管理系统需求 8](#_Toc512093572)

[2.2 系统的特点 9](#_Toc512093573)

[2.3 数据流程图 9](#_Toc512093574)

[第三章 系统概要设计 12](#_Toc512093575)

[3.1 数据库概念设计 12](#_Toc512093576)

[3.2 数据库逻辑设计 13](#_Toc512093577)

[3.3 系统功能模块设计 20](#_Toc512093578)

[3.3.1 实习生功能模块 20](#_Toc512093579)

[3.3.2 HR功能模块 21](#_Toc512093580)

[3.3.3 后台管理系统模块 22](#_Toc512093581)

[第四章 详细设计与实现 24](#_Toc512093582)

[4.1系统文件结构的设计 24](#_Toc512093583)

[4.1.1 java类和接口 24](#_Toc512093584)

[4.2系统架构的设计 25](#_Toc512093585)

[4.2.1表现层设计 26](#_Toc512093586)

[4.2.2业务逻辑层设计 32](#_Toc512093587)

[4.2.3数据库持久层设计 36](#_Toc512093588)

[4.3界面功能设计 40](#_Toc512093589)

[4.3.1实习生用户模块界面设计 40](#_Toc512093590)

[4.3.2 HR用户模块界面设计 45](#_Toc512093591)

[4.3.3后台管理系统界面设计 50](#_Toc512093592)

[4.3.4公共页面界面设计 50](#_Toc512093593)

[第五章 测试与分析 53](#_Toc512093594)

[5.1 测试环境 53](#_Toc512093595)

[5.2 测试原理 53](#_Toc512093596)

[5.3 实例测试 53](#_Toc512093597)

[第六章 结论 55](#_Toc512093598)

[参考文献 56](#_Toc512093599)

[致谢 57](#_Toc512093600)

# 引 言

实习是大学面临毕业的同学必须经历的阶段，如何快速找到可靠有意义的实习工作对每一位大学生来说都尤为重要。对于目前常规的找工作方式，一般是要选择一个可靠的招聘网站，然后注册账号，按自己的条件去搜索实习工作，但是一般的招聘网站不仅仅面向大学生实习这一群体，还有其他各种形式的工作。而对于经验匮乏的毕业生来说，如何在信息量大而杂中找到合适并且可靠的职位，如何辨别这个公司是否是诈骗或者培训公司是非常困难的。

为此本论文阐述的就是专注于实习生招聘网的开发，服务对象就是即将毕业的大学生。目的是让实习生用户可以在更加简洁的环境下，从已发布的职位理性搜索实习职位，而职位对应的信息更加全面与透彻，工资、实习时间、工作地点等要素都会完整的展示。实习公司都会实名化并进行注册认证,增加了平台服务的安全性。

本论文将会剖析当下企业级应用开发最流行的SSM架构在本系统中的应用，去感受spring框架、mybatis持久层框架、spring mvc web框架的实用价值。

# 第一章 概述

网站开发伴随着这些年出现的spring mvc,Spring,Mybaits框架，Ajax技术，XML等新技术使一个单纯的网站项目开发越来越容易。本次项目开发采用SSM架构，数据库选用MySQL5.0。

## 1.1 开发的背景

伴随着中国互联网的迅猛发展，如今越来越多的领域朝向互联网靠近，自然而然“互联网+”成了国家战略的一部分。而就招聘领域来说，网上招聘成为当下最流行的应聘方式。国内最早的传统三大招聘网站有前程无忧，智联招聘以及中华英才网，他们的服务对象类似，对所有大众开放，涉及领域宽泛。针对上述传统网上招聘虽然内容多，由此也衍生出没有对某一特殊对象的精准化服务。由此产生了本论文的研究课题，即针对大学生群体的精细化就业服务平台。

毕业实习是大学学习阶段与步入社会就业阶段的过渡期，是大学实践性教学的重要组成部分，是每一位大学生必须经历的阶段。实习环节是毕业生未来职业生涯起点和校园生涯结束点的有效承接，而一份好的实习工作对未来的发展有极大的促进作用。

围绕着如何实现毕业生及时接受就业信息，引导毕业生快速抉择就业方向，根据个人专业情况以及未来规划提供更多的职位选择，毕业生精准投递简历，公司对简历的有效筛选以及更有效率的网罗到人才，为此通过spring,spring mvc,mybatis整合框架技术开发这个实习生招聘网。本平台的使用对象有三类，分别是实习生，公司hr，平台管理员。实习生可以使用平台提供的简历模板制作简历，选择心仪的职位投递。公司hr根据所发布职位获取对应投递的简历，进行条件筛选过滤部分简历，提高浏览简历决定面试对象的效率。

该平台系统由于只针对实习职位，可以让处于实习阶段的毕业生获得更加准确的信息，针对性有助于提高实习生寻找实习工作的效率以及公司网罗人才的效率。

## 1.2 系统功能概述

本系统主要是服务于实习生找实习工作以及企业寻找合适的可培养人才。本网站平台主要包括实习生用户模块、HR用户模块、后台管理模块等几大部分本平台具有良好的用户接口，用户交互，使用方便，具有完善的浏览已发布职位信息，实习生可以综合筛选出合适的职位进行投递简历，实习生可以对简历进行修改。HR用户所具有的注册认证公司，发布职位，简历管理等功能是本平台的重要组成部分。后台管理系统的作用相当于维护辅助，能及时、方便、灵活地进行查询、修改、删除等维护性操作。后台管理系统有对于实习生，HR用户有一定的管理，并对管理员用户的权限有一定的设置。系统主要实现以下几个功能模块：实习生用户模块, HR用户模块, 后台管理模块。

(1). 实习生用户模块

当实习生用户登录以后，在公共访问页面的顶端会显示用户登录的信息。个人中心则需要判断当前用户是否登录，登录后才允许访问个人管理界面。职位投递按钮点击后也需要验证实习生用户是否登录。

(2).HR用户模块

HR用户模块要有基本信息管理，发布职位功能，职位管理以及简历管理功能。HR用户只能注册一家公司，并且审核通过后才有发布职位的权限。

(3).后台管理模块

后台管理系统的服务对象主要是网站管理者，具有职位类别设置，企业增值服务提供，企业信息管理，实习生用户信息管理，HR信息管理。

(4).公共页面模块

实习页面部分：提供招聘职位展示多条件搜索服务，公司详细信息，职位详细信息查看。

校招页面部分：提供校招资讯，企业宣讲会信息。

朵喵学院页面部分：提供相关面试笔试资料和技能，为实习生用户提供帮助。

## 1.3开发环境介绍

本项目采用的开发组合是:IDEA + Maven + JDK1.8 + Jetty + MYSQL5.0 + Redis

+Git+Github+Windows10

IDEA是目前市面上最流行的java开发工具之一，其本身是用java语言开发的集成环境，界面舒适功能丰富，同时携带很多开发需要的插件譬如版本管理工具git,svn、spring相关、maven插件、Web服务器Jboss,Tomcat支持等。编码过程中会有智能代码提示，对于开发者来说可以提高写代码的效率。总之这是一款十分强大java开发软件。

本项目采用Maven工具进行管理依赖包，严格来说就是一个maven项目。Maven是Apache旗下的产品，它的定义是一个项目管理以及综合的工具，专业角度阐述提供了一个项目对象模型来管理项目构造。它可以帮助开发人员快速的构建一个具有完整生命周期。Maven具有三套相互独立的生命周期，而最重要的就是Default Lifecycle，其核心部分包括常用的编译（compile），测试(test)，打包(package),部署(site)命令。使用maven前需要下载该工具，并配置相应的本地仓库以及环境变量，为了提升依赖包下载速度在配置远程中央仓库时可以将阿里仓库作为首选，当找不到依赖包再向maven官方的中央仓库下载相关依赖包。

Java的开发环境采用JDK1.8版本。首先介绍jdk，jdk的中文释义是java语言的软件开发包工具，顾名思义可以看出它是整个java开发的核心。Jdk内包括了JRE,即java的运行环境。JRE内含JVM,JVM是java虚拟机，它是实现java语言一次编译到处运行的关键。JDK包含的基本组件有javac,javadoc,java,jar.javap..，常用的包有lang,io,nio,net,util,sql...。

Jetty是开发时用到的服务器，其特点在于可以通过XML来对它进行设置，体现其易用性。Jetty本身也是用java语言编写的，可以说他就是一个servlet容器。相比于Tomcat，Jetty显得更加轻量级。Jetty在IDEA中的结构如下图1- 1所示：

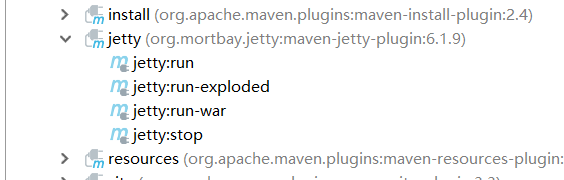


图1- 1 Jetty结构图

Jetty:run命令用于开发时启动项目在本地localhost:8083/duomiao/测试查看项目。

MYSQL5.0是关系型数据库开发工具，其可以支持大型的数据库可以处理上千万条记录。由于其快速轻巧，它多被应用于许多小型企业。开发环境下采用windows版本的mysql，部署环境则采用linux版的mysql。

Redis数据库作为本次项目的mybatis的二级缓存，它是基于内存亦可持久化的Key-Value数据库，即非关系型数据库。读写速度快，性能极高等特点使其在消息队列，热数据（经常读，但不经常修改的数据）领域很受欢迎。当然redis也支持数据备份，同时还有list,hash,set等存储结构。

Git是分布式版本管理工具，基于linux内核开发，所以其终端类似于shell。本项目用git进行版本管理，在项目根目录下运行 git init命令将项目纳入版本管理。Git由于是分布式，所以没有中心服务器，可以在本地建立本地仓库，当然也可以创建远程仓库当做中央仓库。Git常用命令有git init,git clone,git status,git log,git add,git diff,git commit,git revert,git rm,git rm –cached(不清除工作目录),git clean(清除未track的文件)。

Github是全球最大的开源项目托管平台，其原理也是将git作为版本管理的工具。

此次项目采用github和git结合管理。本地git仓库和github远程仓库采用ssh通讯方式（添加同一公钥），借助IDEA将创建好的项目纳入本地仓库，然后将仓库提交到github远程仓库，代码提交时同时向本地和远程commit push,这样既能方便项目更新又能很好的安全保存。

# 第二章 需求分析

## 2.1 用户需求分析

### 2.1.1 实习生用户模块需求

实习生用户注册账户后，有权自己的profile进行操作，但是为了防止用户恶意注册，账户默认状态是未激活。为此本项目只允许采用邮箱注册，并通过邮箱激活后账户才生效。实习生的个人信息，譬如提供密码展示修改，真实姓名修改，这些对于用户来说是常规功能。所以本项目提供基本信息模块对这些重要信息进行统一管理。

在实习生角度出发，个人简历如何创建，如何填写，如何修改都要从系统的完整性以及用户体验度出发。简历内容需要包括姓名，性别，年龄，籍贯，手机号码，邮箱以及专业，学校，学历，意向职位，工作经验，项目经验等等。实习生只有简历的修改查看权限。

实习生用户在浏览职位或企业的同时，可能有心仪的职位出现，这时收藏功能尤为重要。我只需要建立起收藏对象与实习生的联系即可，并提供其对收藏内容的查看，删除权限。

此外本系统提供了朵喵学院，这一板块提供一些关于就业技巧的资料信息，可供实习生查看下载资料。

### 2.1.2 HR用户模块需求

HR即招聘方，作为企业的人力资源管理者需要在最短的时间内找到符合要求的职员并不是一件容易的事情。如何找到优质实习生源，让实习生关注到自己的公司以及招聘岗位，这些都是HR最大的需求。为此本平台将实习生，HR两方资源汇聚在一起，作为两者连接的桥梁。HR需要发布职位，并且对已发布的职位进行管理，包括投递过来的简历进行筛选。

### 2.1.2 后台管理系统需求

平台运营方需要对平台进行必要的管理，比如职位类别的管理，这部分数据会影响到前端页面的展示。动态数据的建立，提高了系统的灵活度。后台管理最核心的需求就是对三类用户的信息管理，即实习生用户管理，企业信息管理（HR用户），后台管理员用户，当然密码部分没有查看权限。

## 2.2 系统的特点

系统需要实现以下特点：

1. 安全性，杜绝恶意注册，未认证HR的权限控制；数据存储方面的id采用UUID避免破解，持久层sql代码防止sql注入。
2. 高效性，精简代码，服务器优化，数据库连接池采用druid。
3. 交互良好，前端页面尽量简洁，按钮或者模块清晰，对应的业务逻辑流畅。
4. 扩展性高，代码采用面向接口的方式编程，尽量提高系统的可维护，可扩展性，满足项目的二次开发。

## 2.3 数据流程图

一、全局数据流程图，如图2- 1所示：

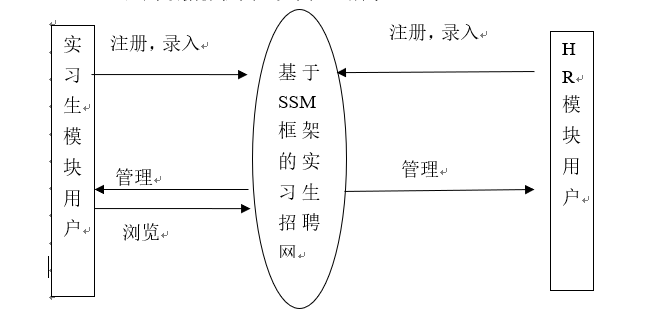


图2- 1 全局数据流程图

二、实习生模块数据流程图，如图2- 2所示

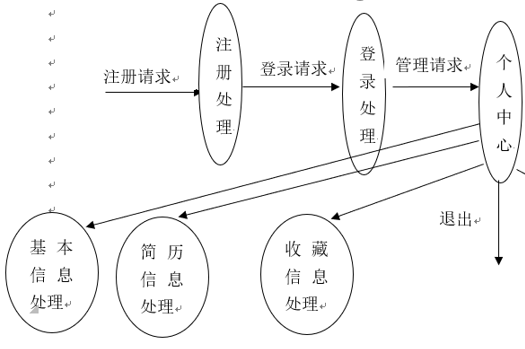


图2- 2 实习生模块数据流程图

三、HR模块数据流程图，如图2- 3所示

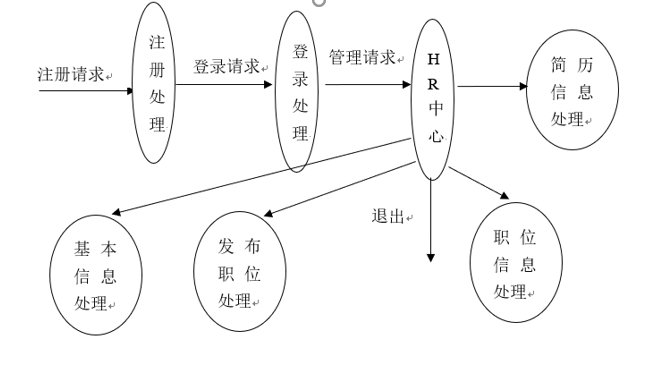


图2- 3 HR数据流程图

四、后台管理系统模块数据流程图，如图2- 4所示

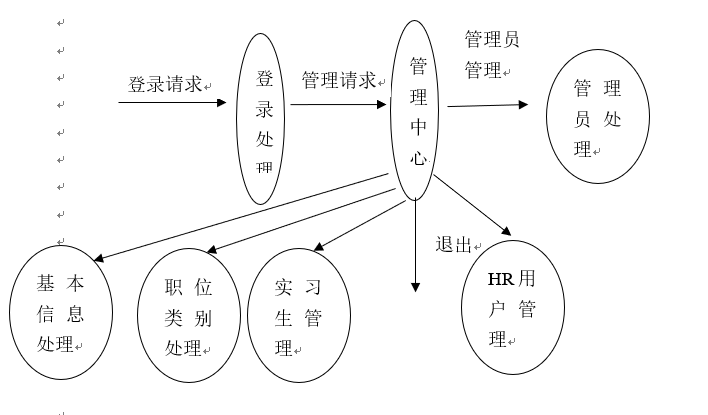


图2- 4 管理员数据流程图

# 第三章 系统概要设计

## 3.1 数据库概念设计

本系统主要的实体包括实习生用户和HR用户以及平台管理员用户，HR用户与实习生用户之间的关系主要体现在职位和简历模块。两者之间通过关系图来描述，关系图如图3- 1所示。

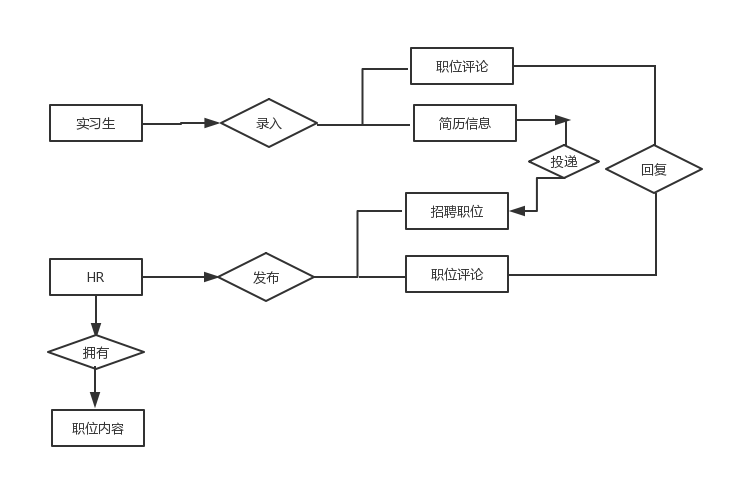


图3- 1 平台ER图

## 3.2 数据库逻辑设计

使用数据库： Mysql5.0

数据库特点：轻巧，功能强大，稳定

概述：本系统构建数据库名称为duomiaodb，该数据库共有十几张表。为了方便以后开发查看表信息，以及数据库的维护，采用一定的规约设计。

规约:

1. 表名的一般格式“DM\_TABLENAME”,“DM\_”代表前缀，“TABLENAME”此处填写表具体名称；
2. 表中字段也采用“[prefix]\_字段名”书写格式，prefix即前缀通常是表名的缩写，字段名即具体属性名称。
3. 每张表的序号字段统一用ID命名，其生成方式采用java代码生成，格式是32位的UUID，目的是为了降低重复率。

具体表说明如下表3- 1所示：

表3- 1 数据库表的详细设计

|  |  |  |
| --- | --- | --- |
| 表 名 | 中文名 | 功 能 说 明 |
| DM\_INTERN\_INFO | 实习生信息表 | 保存实习生账号信息的数据表，包括登录信息，相关状态 |
| DM\_RESUME\_INFO | 简历信息表 | 存储实习生简历内容的数据表，包括基本信息，工作经历，项目经验.. |
| DM\_ENTERPRISE\_INFO | 公司信息表 | 学生学习课程时，对课程的学习感受等内容保存在这张表里，主要对应课程笔记模块 |
| DM\_HR\_INFO | HR信息表 | 保存HR账户信息，包括其联系方式注册时间等等，在后台对其进行管理 |
| DM\_JOB\_REPLY | 评论回复存储表 | 存储评论信息，该表的主要字段是hr ID,实习生ID，职位ID |
| DM\_POSITION\_CATEGORY | 职位类别表 | 保存职位大类和小类信息，用于管理员在后台管理系统中对职位类别的设置 |
| DM\_PUBLISH\_JOB | 已发布职位表 | 本张表保存了已发布职位的基本信息，在HR用户模块里是最重要的表之一 |
| DM\_ENT\_RESUME | 企业与简历关系表 | 保存了企业所受的简历信息，通过企业ID，职位ID，简历ID三者简历联系。使HR能够获得相关简历的查看权限 |
| DM\_ENT\_COLLECT | 公司收藏表 | 保存了实习生用户所收藏所有的公司基本信息，主要对应实习生模块，本表有实习生ID，公司ID以及公司名称等字段 |
| DM\_JOB\_COLLECT | 职位收藏表 | 保存了实习生用户所有收藏的心仪职位的基本信息，由实习生用户来进行管理（增加、删除、查看），主要字段有实习生ID，收藏职位ID，职位名称，公司名称等等 |
|  |  |  |
|  |  |  |
|  |  |  |

在DM\_INTERN\_INFO（实习生信息表）中，存放了实习生用户的基本信息。数据表的具体内容如表3- 2所示。

表3- 2 DM\_INTERN\_INFO表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实习生信息表(DM\_INTERN\_INFO） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  |
| INT\_LOGIN\_NAME | 登录名 | varchar(20) | unique | 使用邮箱注册 |  |
| INT\_LOGIN\_PWD | 密码 | varchar(20) |  | 6-10位数字和字母混合 |  |
| INT\_REAL\_NAME | 真实姓名 | varchar(20) |  |  | |
| INT\_IMG\_URL | 头像路径 | varchar(255) |  |  | |
| INT\_DB | 朵币 | varchar(20) |  | 数字 | |
| INT\_FLAG | 唯一标识 | char(1) |  | 默认为I | |
| INT\_STATE | 状态 | char(1) |  | 0为无效，1为有效 | |
| INT\_REGISTER\_TIME | 注册时间 | datetime |  | 24小时制，精确到秒 | |

在 DM\_RESUME\_INFO（简历信息表）中，存放了实习生用户简历的基本信息。数据表的具体内容如表3- 3所示。

表3- 3 DM\_RESUME\_INFO表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 简历信息表（DM\_RESUME\_INFO） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  |
| RI\_NAME | 姓名 | varchar(20) |  |  | |
| RI\_MOTTO | 人生格言 | varchar(20) |  | 字数限制在12个字以内 |  |
| RI\_SEX | 性别 | varchar(20) |  | 0位女，1为男 |  |
| RI\_AGE | 年龄 | varchar(5) |  |  | |
| RI\_HP | 籍贯，出生地 | varchar(10) |  | 格式 |省份+城市 | |
| RI\_TEl | 手机号码 | varchar(20) |  | 必填 | |
| RI\_EMAIL | 邮箱 | varchar(22) |  |  | |
| RI\_MAJOR | 专业 | varchar(20) |  |  | |
| RI\_COLLOGE | 学校 | varchar(20) |  |  | |
| RI\_ACADEMIC | 学历 | varchar(5) |  |  | |
| 求职意向（部分） | | | | | |
| RI\_JOB\_NAME | 职位 | varchar(20) |  |  |  |
| RI\_WORKPLACE | 期望工作地点 | varchar(10) |  | 不能超过10个字 |  |
| RI\_EDUCATION | 教育背景 | longtext |  |  |  |
| RI\_CERT\_REWARD | 奖励证书 | longtext |  |  |  |
| RI\_SKILL | 专业技能 | longtext |  |  |  |
| RI\_WORK\_EXPERIENCE | 工作经验，实习经验 | longtext |  |  |  |
| RI\_PRO\_EXPERIENCE | 项目经验 | longtext |  |  |  |
| RI\_VALUATION | 自我评价 | longtext |  |  |  |

在 DM\_ENTERPRISE\_INFO（公司信息表）中，存放了公司相关的基本信息。数据表的具体内容如表3- 4所示。

表3- 4 DM\_ENTERPRISE\_INFO表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 公司信息表（DM\_ENTERPRISE\_INFO） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  | |
| EI\_ENINAME | 企业名称（全称） | varchar(20) | unique,非空 |  |  | |
| EI\_RESUME | 简要概述,一句话介绍 | varchar(20) |  | 字数不能超过20字 |  | |
| EI\_IMG\_URL | 公司图像 | varchar(255) |  |  |  | |
| EI\_INTRODUCTION | 介绍 | longtext |  |  |  | |
| EI\_ADDRESS | 地址 | varchar(30) |  |  |  | |
| EI\_TEL | 公司联系方式 | varchar(20) | 不为空 |  |  | |
| EI\_EMPLOYEENUM | 员工数 | int |  |  |  | |
| EI\_INDUSTRY | 行业方向，领域 | varchar(15) |  |  |  | |
| EI\_UNISCID | 统一社会信用代码 | varchar(30) |  |  |  | |
| EI\_REGTIME | 成立日期 | date |  |  |  | |
| EI\_REG\_CAPITAL | 注册资本 | varchar(15) |  |  |  | |
| EI\_WELFARE | 企业福利 | longtext |  |  |  | |
| EI\_STATUS | 企业审核状态 | char(1) |  | 0为无效，1为有效 |  | |
|  |  |  |  |  |  | |

在 DM\_HR\_INFO（hr信息表）中，存放了HR用户相关的基本信息。数据表的具体内容如表3- 5所示。

表3- 5 DM\_HR\_INFO表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| hr信息表（DM\_HR\_INFO） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  |
| HI\_HR\_NAME | 账户 | varchar(20) | unique,非空 | 任意注册 |  |
| HI\_PASSWORD | 密码 | varchar(32) |  | 6-10位数字和字母混合 |  |
| HI\_IMG\_URL | hr头像路径 | varchar(255) |  | 绝对路径 |  |
| HI\_ENTID | ID标识 | varchar(40) | null | 企业ID |  |
| HI\_FLAG | 唯一标识 | char(1) |  | 默认为H |  |
| HI\_TEL | hr联系方式 | varchar(20) |  |  |  |
| HI\_REGTIME | 注册时间 | datetime |  |  |  |

在 DM\_JOB\_REPLY（评论回复信息表）中，存放了评论回复相关的基本信息。数据表的具体内容如表3- 6所示。

表3- 6 DM\_JOB\_REPLY表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 评论回复储存表（DM\_JOB\_REPLY） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  |
| JR\_HR\_ID | hr ID | varchar(40) | 非空 | 邮箱注册，验证码，格式 |  |
| JR\_INT\_ID | 实习生ID | varchar(40) |  |  |  |
| JR\_MAIN\_INFO | 评论主体内容 | varchar(52) |  | 不能超过50字 |  |
| JR\_REPLY\_TIME | 回复时间 | datetime |  |  |  |
| JR\_JOB\_ID | 职位ID | varchar(40) |  |  |  |
| JR\_PARENT\_ID | 父ID | varchar(40) |  | 0为无 |  |
| JR\_REPLY\_NUM | 评论数量 | varchar(10) |  |  |  |
| JR\_AGREE\_NUM | 赞同数量 | varchar(10) |  |  |  |
| JR\_CLICK\_USER | 点过赞的用户UUID拼接在一起 | longtext |  | 初始值为""逗号作为分隔符 |  |

在 DM\_POSITION\_CATEGORY（职位类别信息表）中，存放了职位类别相关的基本信息。数据表的具体内容如表3- 7所示。

表3- 7 DM\_POSITION\_CATEGORY表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 职位类别表（DM\_POSITION\_CATEGORY） | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |
| PC\_TYPE\_NAME | 类别名称 | varchar(20) | 非空，unique | 不能重名 |
| PC\_FATHER\_ID | 父ID | varchar(40) |  | 空默认为0 |
| PC\_URL | 链接 | varchar(255) |  |  |

在 DM\_PUBLISH\_JOB（已发布职位信息表）中，存放了已发布职位相关的基本信息。数据表的具体内容如表3- 8所示。

表3- 8 DM\_PUBLISH\_JOB表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 已发布职位表（DM\_PUBLISH\_JOB） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  |
| PJ\_JOB\_NAME | 职位名称 | varchar(20) | 非空 |  | |
| PJ\_ENT\_ID | 所属公司 | varchar(40) |  | 空默认为0 | |
| PJ\_PC\_ID | 所属职位的类别 | varchar(40) |  |  | |
| PJ\_MIN\_SALARY | 最低薪水 | varchar(10) |  |  | |
| PJ\_MAX\_SALARY | 最高薪水 | varchar(10) |  |  | |
| PJ\_MIN\_EDUCATION\_LEV | 最低学位要求 | varchar(5) |  | 大专，本科，博士，硕士 四选一 |  |
| PJ\_WORK\_PLACE | 工作地点 | varchar(10) |  | 选择 市一级 下拉框 |  |
| PJ\_DESCRIBE | 职位描述 | longtext |  |  | |
| PJ\_WORK\_DAYS | 每周工作天数 | varchar(2) |  |  | |
| PJ\_WORK\_MONTH | 工作月数 | varchar(2) |  |  | |
| PJ\_UPDATE\_TIME | 刷新时间 | datetime |  |  | |
| PJ\_CUT\_DATE | 截止日期 | datetime |  | 2017-12-30 | |
| PJ\_URGENCY | 急招标志 | char(1) |  | 1代表急招，0代表默认，需要付钱 |  |
| PJ\_SEND\_NUM | 热门职位标志HOT | int |  | 投递次数最多 |  |

在DM\_ENT\_RESUME（企业与简历关系信息表）中，存放了企业与简历关系相关的基本信息。数据表的具体内容如表3- 9所示。

表3- 9 DM\_ENT\_RESUME表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 企业与简历关系表（DM\_ENT\_RESUME） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  |
| ER\_ENT\_ID | 企业ID | varchar(40) | 非空 |  | |
| ER\_PJ\_ID | 所属职位ID | varchar(40) |  |  | |
| ER\_RI\_ID | 简历ID | varchar(40) |  | 解决重复投递，唯一标识 |  |
| ER\_ADD\_TIME | 投递时间 | datetime |  |  | |
| ER\_SEX | 性别 | char(1) |  |  | |
| ER\_AGE | 年龄 | varchar(3) |  |  | |
| ER\_ACADEMIC | 学历 | varchar(5) |  | 大专，本科，博士，硕士 四选一 | |
| ER\_ISDEL | 是否删除 | char(1) |  | 默认为0没有删除 1为删除 | |
| ER\_DEL\_TIME | 删除时间 | datetime |  | 数据恢复后删除时间为null | |

在DM\_ENT\_COLLECT（公司收藏信息表）中，存放了所有收藏公司相关的基本信息。数据表的具体内容如表3- 10所示。

表3- 10 DM\_ENT\_COLLECT表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 公司收藏表（DM\_ENT\_COLLECT） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  |
| EC\_INT\_ID | 实习生ID | varchar(40) | 非空 |  | |
| EC\_ENT\_ID | 公司ID | varchar(40) |  |  | |
| EC\_URL | 链接 | varchar(255) |  |  | |
| EC\_ENT\_NAME | 公司名称 | varchar(20) |  |  | |
| EC\_INPUT\_TIME | 录入时间 | datetime |  |  | |

在DM\_JOB\_COLLECT（职位收藏信息表）中，存放了实习生用户所有对应收藏职位相关的基本信息。数据表的具体内容如表3- 11所示。

表3- 11 DM\_JOB\_COLLECT表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 职位收藏表（DM\_JOB\_COLLECT） | | | | | |
| 字段 | 中文名 | 类型 | 约束 | 备注 | |
| ID | ID标识 | varchar(40) | 主键 | UUID(通用唯一识别码) |  |
| JC\_INT\_ID | 实习生ID | varchar(40) | 非空 |  | |
| JC\_PJ\_ID | 收藏职位ID | varchar(40) |  |  | |
| JC\_PJ\_NAME | 职位名称 | varchar(20) |  |  | |
| JC\_GO\_URL | 去向路径 | varchar(255) |  |  | |
| JC\_ENT\_NAME | 公司名称 | varchar(20) |  |  | |
| JC\_INPUT\_TIME | 录入时间 | datetime |  |  | |

## 3.3 系统功能模块设计

### 3.3.1 实习生功能模块

实习生用户功能模块包含有以下6个功能模块，如图3- 2所示。

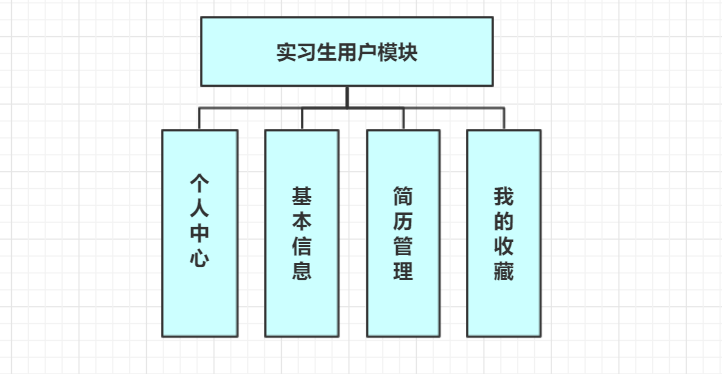


图3- 2 实习生功能模块功能划分图

(1) 个人中心，是整个实习生功能模块的中枢神经，有点类似于用户的后台管理界面；在个人中心模块下，包含基本信息模块、简历管理模块、我的收藏模块、我的猫侍模块以及我的课程模块共5个子模块，实习生用户的profile配置或者核心的简历填写生成以及简评组件都可以在这些子模块功能区中完成。

(2) 基本信息：该模块只在个人中心里展示，主要显示的信息有实习生用户的用户名，密码，真实姓名，头像图片。其中用户名只读不可修改，密码可以修改但不能为空，真实姓名可以为空，可以上传头像（不上传时采用默认头像显示）。

(3) 简历管理：本子模块用户展示实习生用户的简历，实习生可以对简历进行初始化创建，后期修改，但不具备删除简历的权限；关于简历内容的填写限制，姓名、性别、年龄、籍贯、手机号码、专业、学校、学历为必填项，此外可选填的有人生格言、邮箱、期望职位、期望工作地点、教育背景。当然还可以选填的有奖励证书、专业技能、工作经验、项目经验、自我评价部分。

(4) 我的收藏：我的收藏主要是展示实习生用户所收藏的企业和职位的信息；实习生用户对每个对象只能收藏一次，可以对已收藏对象进行删除操作，但不具备修改操作，对于收藏对象的数量也会有一定的限制。

(5) 公共页面：本项目的首页属于公共页面，该页面展示常规的导航栏进行跳转选择，页面包含有职位分类展示，banner企业或职位企划活动轮播图展示，按照职位名称或公司名称搜索框功能；该页面还具有实习生登录和公司登录注册入口。除了首页，实习模块旗下所含的所有页面，校招下的所有页面，朵喵学院的页面，猫侍社区的所有页面。

### 3.3.2 HR功能模块

HR用户功能模块主要实现了以下6个子功能模块，如图3- 3所示：

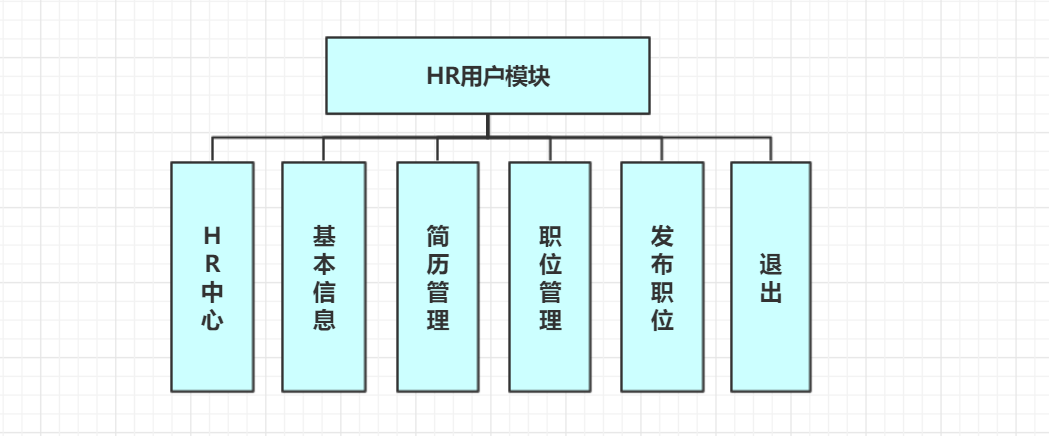


图3- 3 HR用户模块功能划分图

(1) HR中心

同样HR中心就是HR用户功能模块的中枢神经，HR中心负责调度以下基本信息、发布职位、职位管理、简历管理、退出功能一共6个子模块。本项目的公共首页有HR中心的链接，可以直接跳转到HR用户的管理中心主界面。

(2) 基本信息

在这个子模块中可以浏览到HR用户的账户信息，包括账号密码、联系方式等等；其中HR用户可以对密码、头像、联系方式进行修改，而账户名，关联企业ID，注册时间都是一经初始化就不可以修改的。

(3) 发布职位

该子模块功能是HR用户管理中心的核心功能，HR用户经过注册企业通过审核有效后才可以发布职位；发布职位过程中可填写项有职位名称，所属类别（选择），最低薪水，最高薪水，最低学位要求（限制填写范围和格式），工作地点（必填），职位描述，每周工作天数，工作月数，截止日期，急招标志；此外的ID标识，所属公司，刷新时间，热门职位标志都是有系统完成填写；急招标志需要支付一定代价才有机会设置。

(4) 职位管理

HR用户可以在这里浏览已经发布的职位信息，会将重要的信息缩减展示；HR用户觉得需要更新已发布职位时，可以对选定的职位进行修改；当然HR用户也具有对职位的删除权限，添加权限则体现在发布职位功能模块中。

(5) 简历管理

此子功能模块用于显示实习生用户所投递过来的简历信息，HR用户可以查看这些简历，并通过多条件进行筛选浏览，快速选择最优实习生简历进行评估；HR用户也可以对不满意的简历进行删除操作，删除后也可以对简历在一定的时间内进行恢复;简历的添加体现在实习生用户的投递，投递限制为一个职位只能投一次。

(6) 退出

退出操作，会将服务器上HR用户对应的session进行删除操作，页面会返回项目的公共主页面；如果长时间不操作，系统默认执行session清空操作。

### 3.3.3 后台管理系统模块

后台管理系统功能模块主要实现了以下几个功能模块，如图3- 4所示：

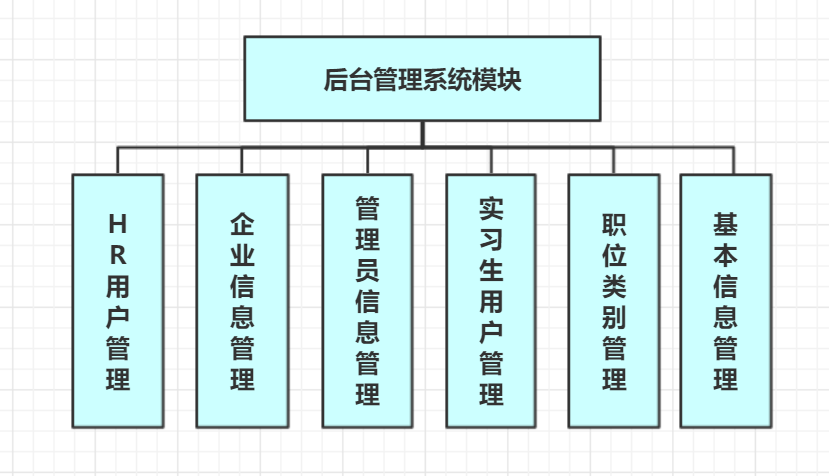


图3- 4 后台管理系统模块功能划分图

(1) HR用户信息管理：在这里可以查看和删除HR用户。可以查看的内容有账户名称，密码非明文显示，头像路径，关联企业，联系方式，注册时间信息。删除功能一般用不到，只有存在特殊情况才会使用此功能。

(2) 企业信息管理：在此处可以查看，删除企业的信息。

(3) 实习生用户管理：本功能包含可以查看实习生用户的部分信息，按条件筛选查询，可以在特殊情况下动用删除功能。

(4) 管理员信息管理：本子模块提供对管理员用户的增加、删除、修改、多条件查询功能。

(5) 职位类别管理：可以对职位类别进行添加，修改，删除以及查看。

(6) 基本信息：修改自己的密码。

(7) 退出：删除session中所存储有关管理员账户的信息。

# 第四章 详细设计与实现

## 4.1系统文件结构的设计

### 4.1.1 java类和接口

本项目的部分关键java类信息，java接口信息如下表4-1所示。

表4-1 java类或接口信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | 说明 | 类名 | 说 明 |
| IndexControl | 公共页面访问类 | InternInfo | 实习生实体类 |
| InternInfoControl | 实习生相关访问控制类 | ResumeInfo | 简历实体类 |
| RedisControl | Redis测试控制类 | EntCollect | 收藏企业实体类 |
| ResumeInfoControl | 简历访问控制类 | EnterpriseInfo | 企业信息实体类 |
| EntCollectControl | 企业收藏访问类 | EntResume | 企业与简历关系实体类 |
| EnterpriseInfoControl | 企业信息访问类 | HrInfo | HR实体类 |
| EntResumeControl | 企业简历关系访问类 | JobCollect | 收藏职位实体类 |
| HrInfoControl | HR用户访问类 | JobReply | 职位评论实体类 |
| JobCollectControl | 职位收藏访问类 | PositionCategory | 职位类别实体类 |
| JobReplyControl | 评论访问控制类 | PublishJob | 职位实体类 |
| PositionCategoryControl | 职位类别访问控制类 |  |  |
| PublishJobControl | 职位访问控制类 | RedisCacheUtil | 第三方缓存服务类 |
|  |  |  |  |
| InternInfoService | 实习生服务接口 | HrInfoService | HR用户服务接口 |
| InternInfoServiceImpl | 实习生服务接口实现类 | HrInfoServiceImpl | HR用户服务接口实现类 |
| PublishJobService | 职位服务接口 | EntResumeService | 企业简历关系服务接口 |
| PublishJobServiceImpl | 职位服务接口实现类 | EntResumeServiceImpl | 企业简历关系服务接口实现类 |
| JobReplyService | 职位评论服务接口 | EnterpriseInfoService | 企业信息服务接口 |
| JobReplyServiceImpl | 职位评论服务接口实现类 | EnterpriseInfoServiceImpl | 企业信息服务接口实现类 |
| JobCollectService | 收藏职位服务接口 | EntCollectService | 收藏企业服务接口 |
| JobCollectServiceImpl | 收藏职位服务接口实现类 | EntCollectServiceImpl | 收藏企业服务接口实现类 |
| PositionCategoryService | 职位类别服务接口 | ResumeInfoService | 简历服务接口 |
| PositionCategoryServiceImpl | 职位类别服务接口实现类 | ResumeInfoServiceImpl | 简历服务接口实现类 |
|  |  |  |  |
| InternInfoMapper | 实习生Dao接口 | ResumeInfoMapper | 简历Mapper接口 |
| PublishJobMapper | 职位Dao接口 | HrInfoMapper | HR用户Mapper接口 |
| JobReplyMapper | 职位评论Mapper接口 | EntResumeMapper | 企业简历关系Dao接口 |
|  |  |  |  |
| JobCollectMapper | 收藏职位Mapper接口 | EnterpriseInfoMapper | 企业信息Mapper接口 |
| PositionCategoryMapper | 职位类别Mapper接口 | EntCollectMapper | 收藏企业Mapper接口 |
|  |  |  |  |
| AjaxResult | Ajax返回实体类 | Consant | 常量类 |
| DateHelper | 时间格式处理工具类 | RedisCacheTransfer | Redis静态注入属性类 |
| UUIDBuilder | UUID生成调用类 |  |  |
|  |  |  |  |

## 4.2系统架构的设计

本系统是B/S项目，后台框架采用spring，spring mvc，mybatis整合架构。代码的编写格式符合表示层、业务逻辑层和数据访问层、数据层分层结构。数据层即数据库，本项目中使用的是Mysql5.0数据库。前端页面使用的插件或框架有jquery插件、bootstrap框架、require.js插件、layer.js插件、bootstrap-table表格插件。

### 4.2.1表现层设计

本项目的表示层采用spring mvc框架，它是一种符合MVC设计模式的web层架构。它是以DispatcherServlet为核心设计的，其工作流程大致分为4步:

1. 前端控制器DispatcherServlet接收到来自客户端的请求后，将其交给处理器映射器HandlerMapping;
2. 处理器映射器会根据请求路径去找到相对应的处理器适配器，然后选则对应的Control控制器执行对应的方法；
3. 处理完请求后，返回一个ModelandView视图模型对象；
4. 视图解析器对其进行处理，最后将数据渲染到视图上。

本项目中的所有页面的访问等都是由spring mvc的dispacher servlet来控制。

(1) 视图部分

①JSP页面

本系统所有页面采用jsp形式的页面类型，对于不算太庞大的项目使用服务器页面jsp可以使动态数据展示更加容易。Jsp支持多种标签语言，这里用到c标签语言处理判断，循环数据展示问题，数据展示部分也用到el表达式；Jsp的require,session等内置对象可以存储用户相关的信息。当然大部分数据都是采用ajax方式请求后台返回json数据，然后解析json格式数据展示到界面上。

(2) 控制部分

①Control类

每一个实体对象大多数情况下都对应一个control类，这些类都存放在com.duomiao.control包下面。Spring mvc收到的请求会根据路径匹配到相应的控制器，选择对应的方法并执行。

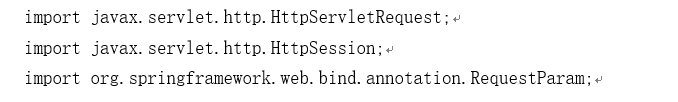
control类需要加上@Controller注解，以此来告诉 spring容器这是一个控制器类，这样spring就可以管理它的实例化单例对象了。当然还可以加上一个@RequestMapping注解来表示整个类的访问根路径，然后类里面的每个方法也有相对应的请求映射注解。最后浏览器可以输入对应的url路径来访问，测试接口的时候可以用postman工具来检测该http接口是否有效。

在com.duomiao.util工具包中，有一个名称为AjaxResult的工具类。它的作用是包装返回值对象，为了支持以json数据格式返回到前端，该类需要实现Serializable接口（代表该类可以序列化）。该类的success属性是布尔类型，用来代表这次操作是否成功；其message属性的类型是Map<String,Object>键值对集合，存放的是需要返回的数据。

这里我通过实习生信息访问控制类这个例子来展示controller类结构:

下面是InternInfoControl类代码片段，以图片加文字方式展示:

//导入外部依赖包



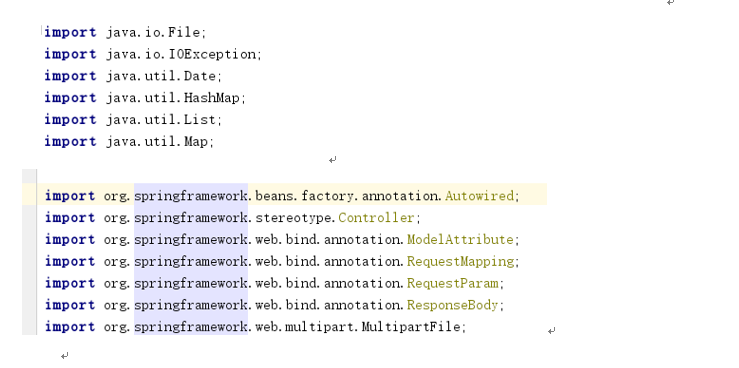


图4- 1 导包图

//InternInfoControl类

1. 该类加上了@Controller注解，使其能被component-scan扫描识别。
2. 配置了该类的访问基路径”/intern”。
3. enLoginAndRegisterPage方法用于访问实习生用户的登录页面。
4. doLogin方法用来处理实习生登录，验证用户名和密码是否有效。
5. doLoginOut方法的作用是处理实习生退出操作，具体执行内容包括删除Session。的SESSION\_INTERN\_INFO属性值，以及跳转到本系统首页。



图4- 2

//重名校验&注册功能

1. doCheckSameName方法的作用是检测该用户名是否已存在，当存在是返回的AjaxResult的success属性设置为true。
2. doRegister方法的作用是执行注册操作，其中系统设置关于注册用户对象的属性有db(朵币)、flag(I代表实习生用户)、state(默认为0,未激活),imgUrl（采用默认头像路径）。



图4- 3

//用户基本信息修改功能、一些页面访问

1. alertMainInfoByOwn方法用来处理用户修改请求，传入的参数有（internInfo，session）;根据用户的ID来修改部分信息。
2. enAdminLteIndex方法 作用：访问个人中心页面。
3. enBasicInfo方法 作用：访问基本信息页面。
4. enResumeInfo方法 作用：访问个人简历管理页面。
5. enMyCollect方法 作用：访问我的收藏页面。



图4- 4

//个人头像上传实现

1. upImg方法，传入的参数file代表经过spring mvc处理过的图片文件;

首先获得项目的图片存储位置的物理路径，然后用transferTo方法上传文件到指定位置，最后获取当图片的url并返回到客户端。



图4- 5

②配置文件

Web层的配置文件内容写在spring-mybatis.xml文件中。

1. 配置alibaba的fastjson数据转换器来负责本项目的json数据转换。
2. 文件上传采用spring自带的文件解释器，配置multipartResolver文件解释器。
3. 配置扫描com.duomiao.control包的包扫描器。
4. 设置静态资源访问，包括dmcss、dmjs、imgs等目录下的静态文件。
5. 配置登录相关的拦截器，编码过滤器。
6. 设置视图解析器，系统不拦截jsp页面，视图以jsp形式返回。



图4- 6

### 4.2.2业务逻辑层设计

(1) 业务逻辑层的分析

只负责处理业务，没有与数据直接打交道。主要体现出“高内聚，低耦合”的软件开发思想，业务逻辑层负责与表现层以及数据联络层相联系交互，处理逻辑性数据。对于数据的具体展示样式，则不在业务逻辑层考虑范围之内。

(2)业务逻辑层的实现

业务层的结构如下图4- 7所示：

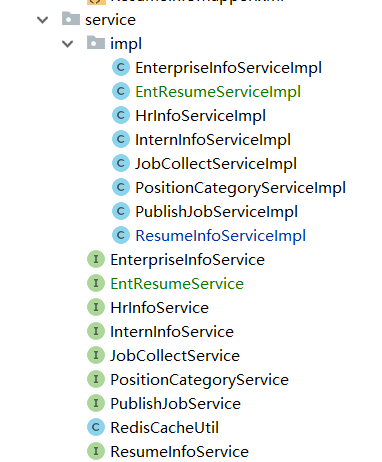


图4- 7 实现接口的类

每个业务都会建一个对应的服务接口，然后新建一个服务类去实现这个服务接口，提高了系统的可扩展性，可维护性。业务层的每个实现类都会加上@Service注解，将其交给spring容器管理，在初始化的时候就创建好这个bean类并且注入依赖。业务层的每个业务类对应的方法作为事务管理的最小单位。这里分析实习生服务类，下面展示代码：

1. InternInfoService接口

//导入自己创建的依赖包

import com.duomiao.entity.I9nternInfo;

//getInternInfoList方法用于多条件查询实习生用户列表信息

//batchDelInternInfos方法用于批量删除实习生用户

//updateInternInfoForOwn方法用于用户自己修改信息

//specialUpdateInternInfo方法用于系统自行处理某些特殊属性的更新

//specialDelInternInfoByState方法用于定时删除无效用户

//registerInternInfo方法用于处理用户注册行为

//validateSameName方法用于重名校验

如下图4- 8：

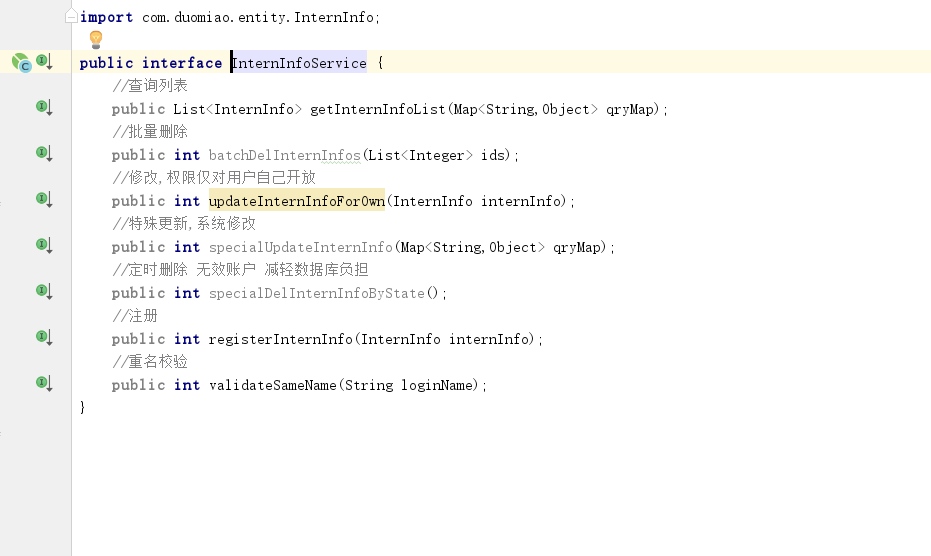


图4- 8

1. 该接口对应的InternInfoServiceImpl实现类:

//导入依赖包

import com.duomiao.entity.InternInfo;

com.duomiao.mapper.InternInfoMapper;

com.duomiao.service.InternInfoService;

//其他包如下图4- 9

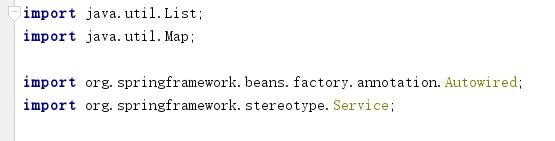


图4- 9

//具体实现代码 如图4- 10：



图4- 10

//ResumeInfoService接口（简历服务接口类）的代码展示

如下图4- 11：

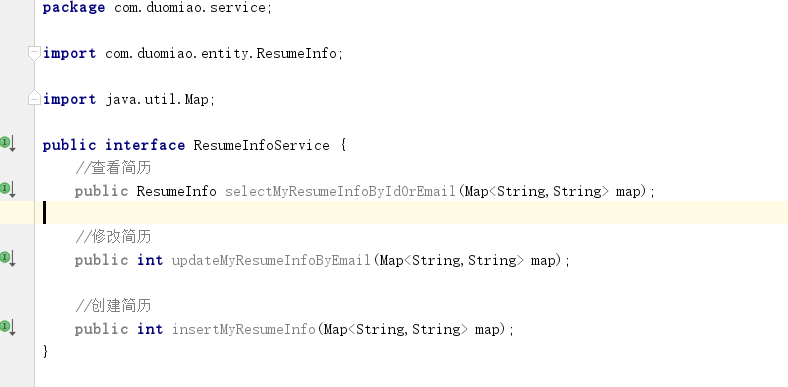


图4- 11

(3) 业务逻辑层的管理：业务层的类加上service注解后，会被spring的自动扫描组件识别。默认情况下bean属性是单例模式，所以spring容器初始化时会给每个类创建实例对象，并对其实时监控和管理。

如果给某个类设置属性值为prototype，则将采用原型模式，每次调用getBean()方法都会产生一个对象，这个对象是不受spring控制的（不处于其监控范围）。

### 4.2.3数据库持久层设计

(1)概述

本项目使用Mybatis框架作为持久层，它是支持高级映射、定制化Sql语句以及数据库存储过程的一种持久层架构。Mybatis框架的主要优点：

1. 使用灵活、提供映射标签；
2. 半自动orm框架，支持编写动态sql；
3. 可以借助第三方数据库作为二级缓存，缓解关系型数据库的压力。

(2) Mybatis配置

1.需要导入主要的jar包: org.mybatis. mybatis-spring、org.mybatis.mybatis、

org.mybatis.caches. mybatis-redis。

2.提供spring-mybatis.xml对关键部分进行配置。首先将dbconfig.properties数据库属性配置文件注入进来，然后配置datasource数据源，数据源采用阿里的druid开源数据源；接着配置sqlsessionFactory工厂bean,并注入datasource数据源，之后配置mapperscanner（数据联络层的自动扫描组件）；然后注入sqlsessionfactory单例对象供dao层实现类方法使用，component-scan扫描service层组件。Mybatis开启缓存支持，redis的具体配置在spring-redis.xml文件中。相关配置文件：

Spring-mybatis.xml

//加载dbconfig.properties数据库配置属性文件

//配置alibaba的druid开源数据库连接池

如下图4- 12：



图4- 12

//配置sqlSessionFactory工厂，datasource属性表示数据源，mapperLocation属性代表该session工厂所管理的映射文件。

//配置configurationProperties，开启二级缓存支持

如下图4- 13：



图4- 13 spring-mybatis配置图

//配置mapper文件扫描器，指定com.duomiao.mapper包，并给这些mapper层接口类提供sqlsession工厂对象使用

如下图4- 14：



图4- 14

(3)映射配置

数据库表与对象通过xml映射文件进行映射，执行mapper接口的实现类中的方法，就可以访问数据库以及实现对数据的操作。

(4) DAO层的设计

Dao层包含一个接口和一个xml文件，spring容器会自动给这个dao接口创建实现类，所以不需要自己写实现类。Dao接口的方法名和xml文件中每个Sql的id对应，通过执行方法发送Sql语句到数据库，最后执行。

## 4.3界面功能设计

下面将会对界面设计以及其具体实现做详细阐述。

### 4.3.1实习生用户模块界面设计

(1) 实习生模块个人中心

①功能描述:个人中心是实习生用户进入自己管理中心的首张页面，主要提供访问其他子模块的路径链接。

②显示效果如图4- 15所示：



图4- 15 个人中心

③实现原理：本后端提供该页面的访问方法，直接返回jsp页面。该页面在左侧位置用ul列表展示其他子模块的导航链接，导航的href链接拼接方式采用“系统的根目录（通过request内置对象获取）+具体url”。

④核心内容：此页放置于WEB-INF路径下，不能通过原有页面位置路径直接在请求访问。因此需要通过后台的服务器跳转的方式访问个人中心，即后台retrun “interncenter/adminlte/index”进行视图渲染。

(2) 实习生用户注册

①功能描述:为实习生提供注册账户功能；进入注册界面，然后输入邮箱名（注册名字）、密码和验证码，之后点击注册按钮。

②页面如图4- 16所示：

![C:\Users\哥斯达黎加\AppData\Roaming\Tencent\Users\3073415886\TIM\WinTemp\RichOle\LU1RZT~6%C]5(3]((8_SYEP.png](data:image/png;base64,)

图4- 16 实习生用户注册页面

③实现原理：前端注册数据需经过格式校验，才能提交数据。通过$.ajax()方法向后台所提供的注册接口发出请求，并返回json格式的确认成功与否数据。

④核心内容：插入用户数据时，其ID用UUID类方法生成；注册接口对应的方法将请求数据封装到internInfo对象中，然后调用业务层提供的插入数据方法进行数据新增。

(3) 实习生用户登录

①功能描述:用户注册功能；用户通过输入正确的邮箱号（用户名）和密码以及验证码后，可以进入实习生管理中心进行操作。

②如图4- 17所示:



图4- 17 实习生登录页面

③实现原理：前端通过后台提供的doLogin接口进行ajax请求，请求数据有用户名和密码，请求数据传输格式是默认的contentType。

④核心内容：登录接口接收的参数包含邮箱和密码，统一用map接收；调用internInfoService的查询方法，返回结果封装成AjaxResult中。

(4)基本信息

①功能描述:用户名只读，可以修改头像、密码、真实姓名。

②显示效果如图3- 8所示：

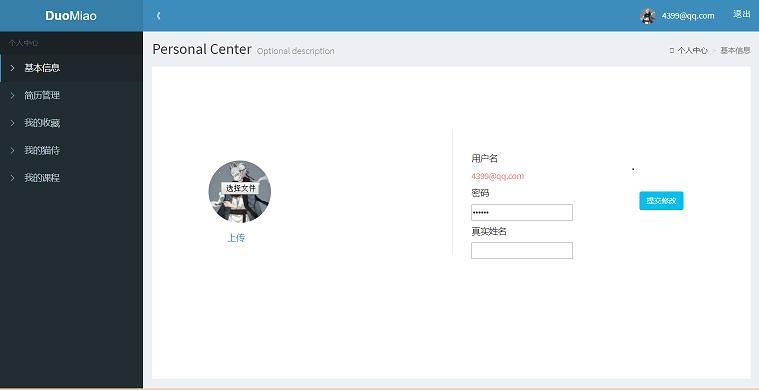


图4- 18 基本信息页面

③实现方法：进入页面，js代码加载时进行初始化操作（向后台接口请求基本信息数据内容）用于展示信息；修改时通过ajax请求doEditMainInfo接口。

④核心内容： doEditMainInfo接口，后台接收数据时封装成internInfo对象，调用internInfoService的更新数据方法（传入参数对象）。返回结果以JsonObject传输。

(5)简历管理模块

①功能描述:展示简历的内容，用户可以根据自己的要求修改简历内容。

②显示效果图4- 19所示：

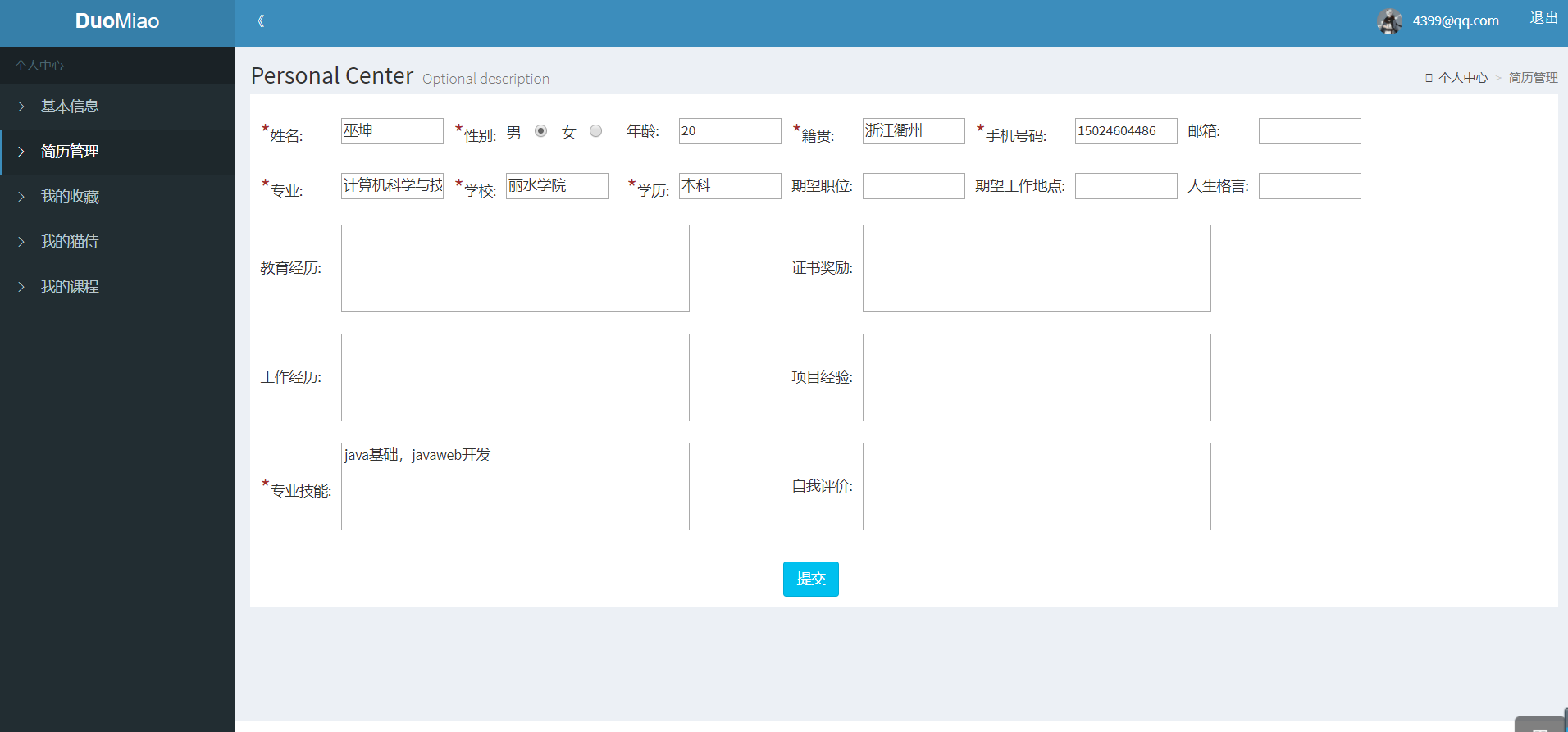


图4- 19 简历管理页面

③实现原理：首次注册的实习生，需要创建简历；创建简历也就是提交的时候访问的接口地址是“/resume/intern/doMakeResumeInfo”，进行简历新增操作；修改时提交的接口地址换成“/resume/intern/doEditResumeInfo”，进行简历修改。

④核心内容：前端数据输入js实现校验（判断必填是否已填、格式是否符合），通过ajax向后台接口发送请求，后台用Map接收数据，创建简历接口调用resumeInfoService

对象的新增方法（插入数据时ID手动生成）；修改简历接口调用resumeInfoService对象的修改方法（根据RI\_EMAIL字段（用户名）进行修改操作）。

(6)我的收藏

①功能描述:展示收藏的职位列表，可以删除选定的职位收藏记录。点击职位名称可以跳转到该职位的具体界面。

②显示效果图4- 20所示：

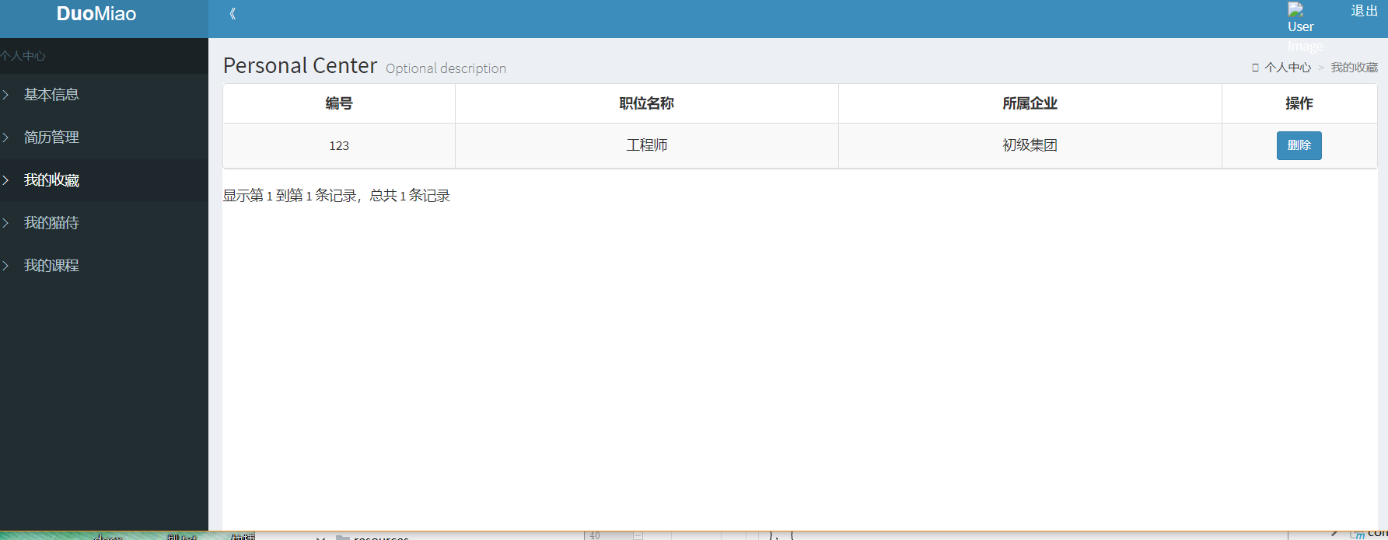


图4- 20 我的收藏页面

③实现原理： 展示职位信息，需要先从后台获取数据，前端采用bootstrap table插件进行数据渲染。删除功能通过ajax请求向后台删除接口传入id参数，根据主键进行删除本条记录。

④核心内容：前端页面通过$(“myResumeInfo”).bootstrapTable({//相关属性设置})方法创建表格，它的sidePagination属性设置为‘server’代表服务端分页，url属性设置为“CONSTANT.root\_url + "/entResume/hr/getValidEntResumes" 代表请求数据的接口路径。

queryParams属性用于设置本次请求自定义参数。该接口提供的方法内容：调用jobCollectService对象的查询方法获取列表和总记录，并将这些数据封装到BootsTable类的实例对象中返回给前端。删除时ajax请求成功后进行表格刷新。

### 4.3.2 HR用户模块界面设计

(1)HR用户模块界面HR中心首页

①功能描述: HR中心是HR用户进入自己管理中心的首张页面，主要提供访问其他子模块的路径链接。

②如图4- 21所示：



图4- 21 HR中心首页

③实现原理：和实习生个人中心类似，通过spring mvc的路径映射，经过服务器跳转，返回jsp页面。

(2) 基本信息界面，与实习生模块的基本信息界面样式一致

①功能描述:用户名只读，可以上传头像，修改密码、联系方式。

②如图4- 22所示。

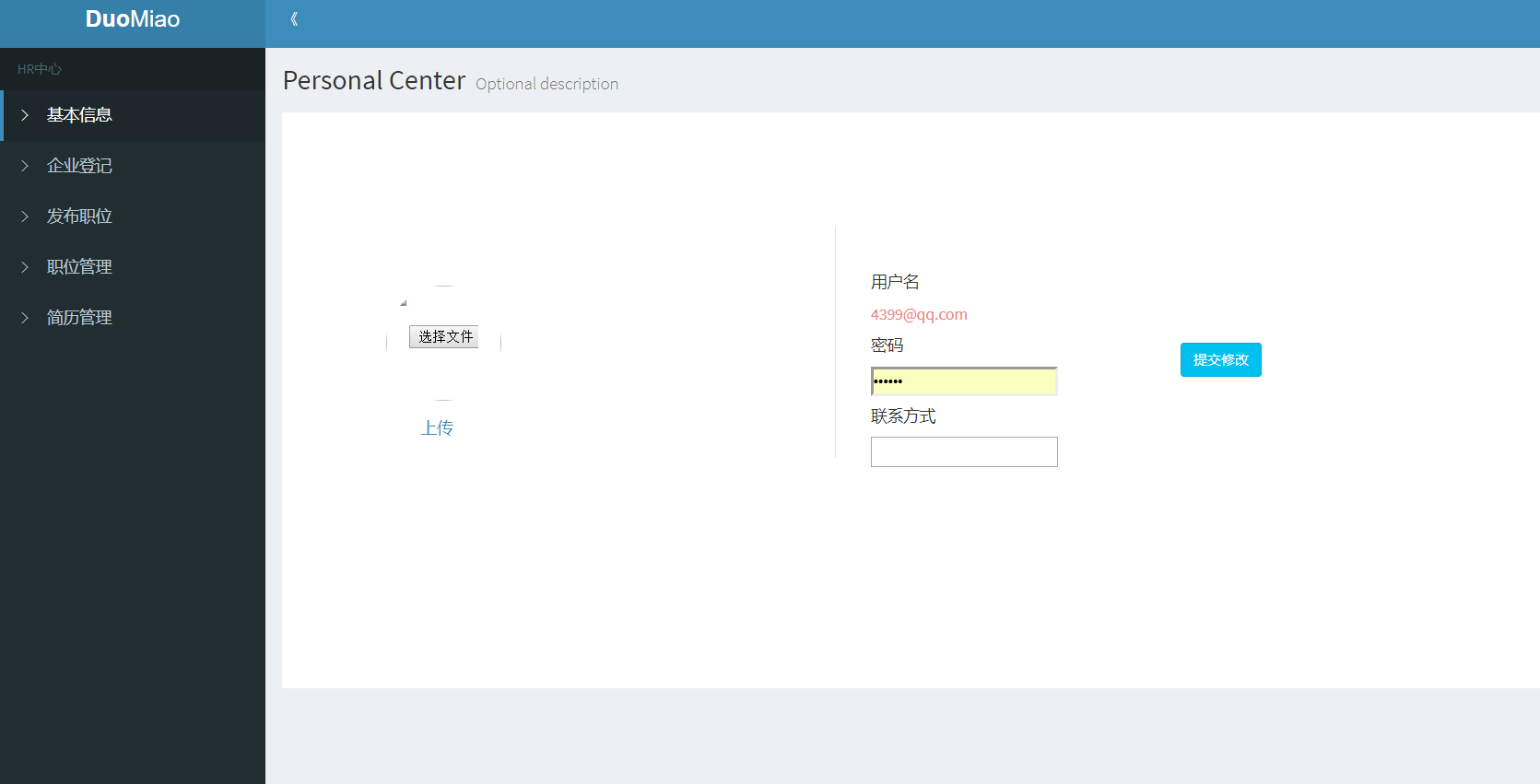


图4- 22 HR的基本信息页面

③实现原理：进入页面，js代码加载时进行初始化操作（向后台接口请求基本信息数据内容）用于展示信息；修改时通过ajax请求接口路径为“/hr /adminlte/doEditMainInfo”。

④核心内容： doEditMainInfo接口，后台接收数据时封装到Map中，调用hrInfoService的更新数据方法（传入参数对象）。返回结果以JsonObject传输。

(3) 发布职位

①功能描述:已经进行过企业登记的HR用户可以发布职位，输入相关信息后，点击提交即可发布职位。

②页面如图4- 23所示。

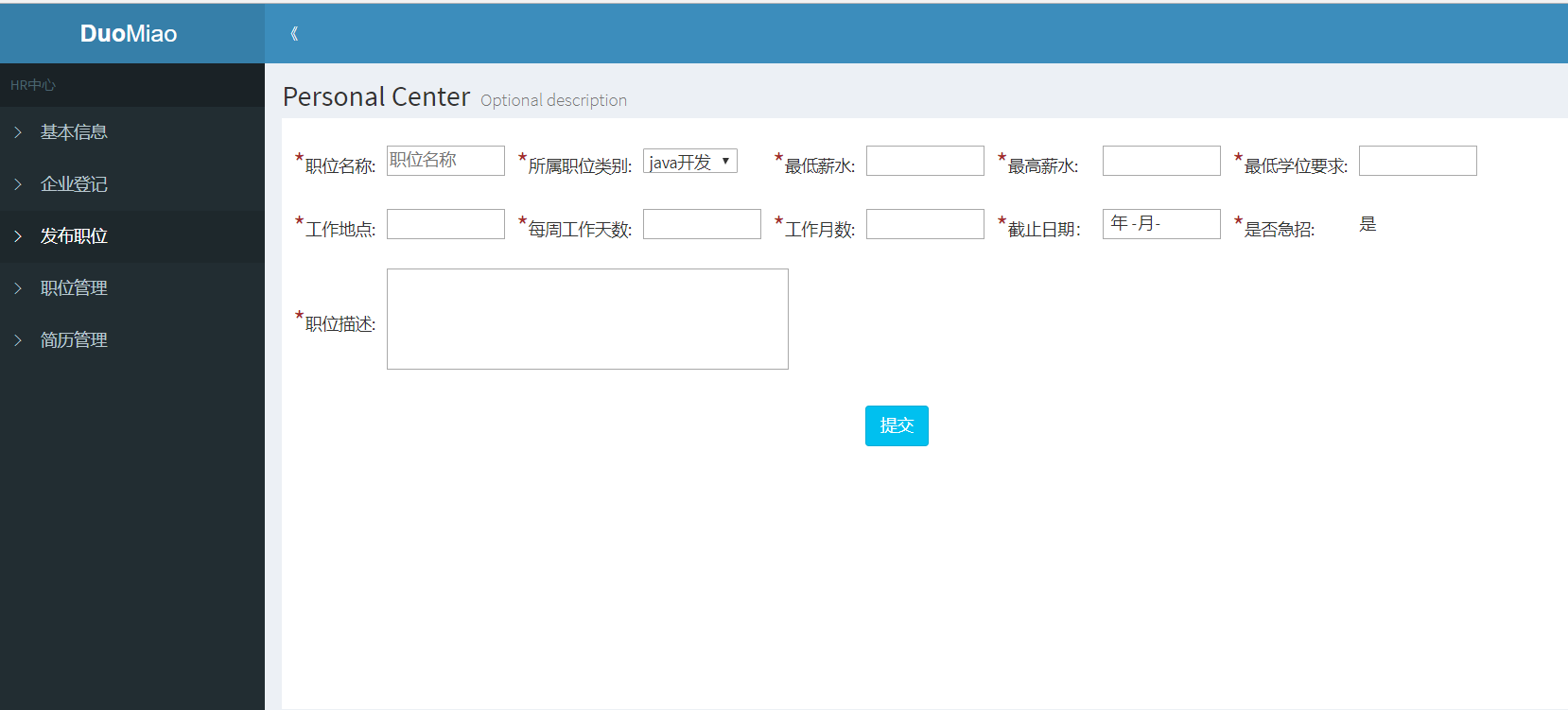


图4- 23 发布职位页面

③实现原理：职位信息填写完成后，通过表单的方式提交到url为 “/pubJob /hr/createPubJob”；后台通过publishJob对象（类型：职位实体类）接收表单的参数集，通过执行publishJobService.insertPubJob(publishJob)方法来创建职位。

④核心内容：publishJob参数对象需要后台添加id（主键），entId（公司ID）,sendNum（投递次数）以及updateTime（更新时间）属性后，再执行插入数据操作。

(4) 职位管理

①功能描述:对自己已发布的职位进行查看，点击删除按钮删除当前职位记录，点击修改对当前职位记录进行修改。

②页面如图4- 24所示：



图4- 24 职位管理页面

③实现原理：进入该页面时，在js中初始化bootstrap table表格，该表格通过访问“/pubJob/hr/getMyPubJobs”接口获取职位信息列表数据。后台执行 publishJobService.selectMyPubJobs(map（存有键名称为’entId’）)方法,根据企业id查询返回职位列表数据后封装到bootsTable对象中返回前端;删除操作通过ajax访问“/pubJob/hr/delPubJob”接口并传递id参数，后台获取id参数后，调用业务层的根据主键删除方法进行删除操作；删除操作通过ajax访问“/pubJob /hr/editMyPubJob”接口，后台用map接收参数，调用业务层updateMyPubJobById方法进行修改操作。

(5) 简历管理页面与上面的职位管理页面样式类似

①功能描述:对所对应的有效投递简历进行展示，可以进行删除企业简历关系记录和查看具体简历信息。

②页面设计效果如图4- 25所示：

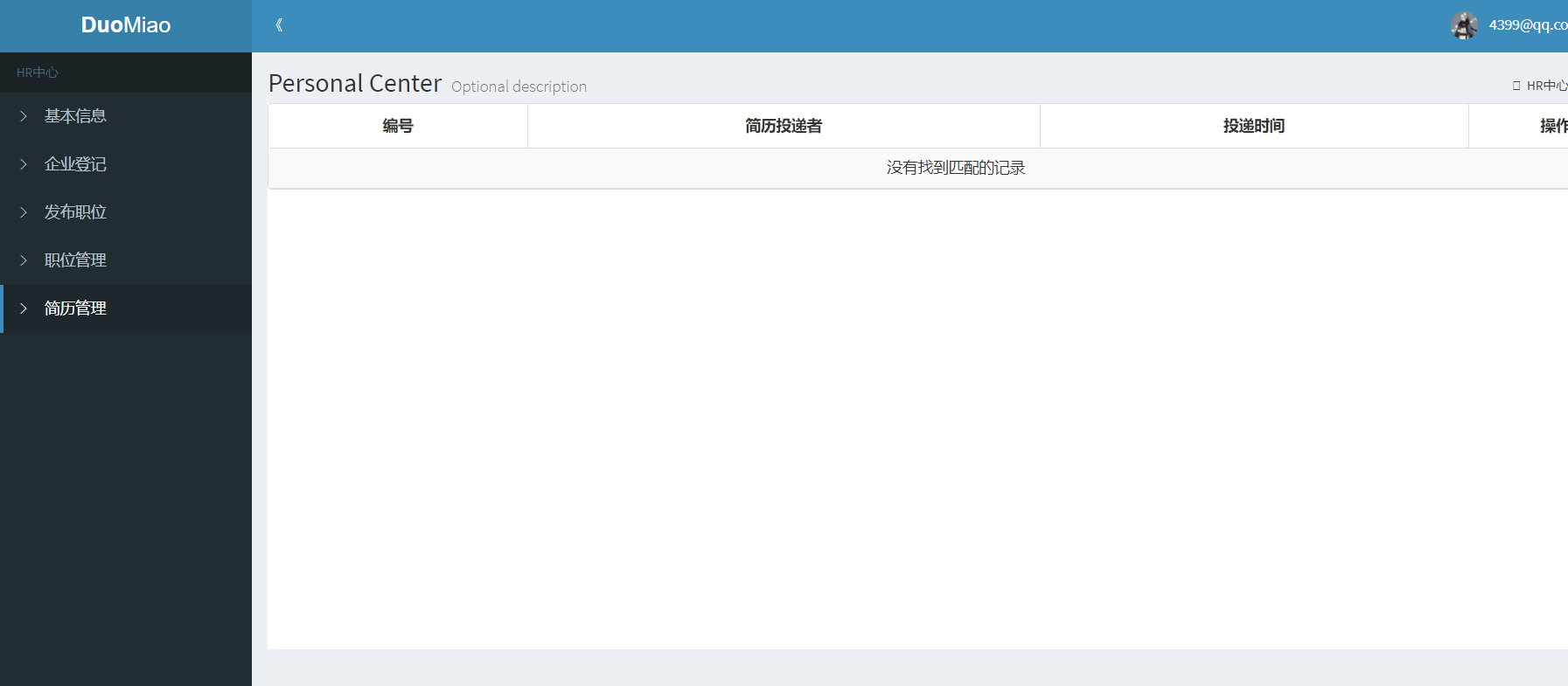


图4- 25 简历管理页面

③实现原理：

对于该页面有效数据展示，前端需通过访问“/entResume /hr/getValidEntResumes”接口来获取有效的简历记录展示；传递的参数在该接口对应的方法中用map来接收，调用entResumeService的selectByMap方法获取企业简历关系对象集合，最后将数据封装到bootsTable对象中返回前端。

查看具体简历信息，是通过请求“/resumehr/browseResumeInfo”接口地址，并将简历id作为参数传递；该接口地址对应的方法是ResumeInfoController类的browseResumeInfo方法，此方法通过调用对应业务类的selectMyResumeInfoByIdOrEmail的方法，传入参数id，最后返回ResumeInfo对象到前端。

删除记录，不是真正删除只是将对应的状态字段值进行修改，updateForFakeDel方法用于匹配删除请求所执行的方法，接收参数id采用map封装,在调用业务层方法updateForFakeDel(map)之前，需要为map增加’delTime’键存放删除时间。最后返回数据用AjaxResult封装发送给客户端。

### 4.3.3后台管理系统界面设计

本平台的后台管理系统首页设计效果如图4- 26所示：

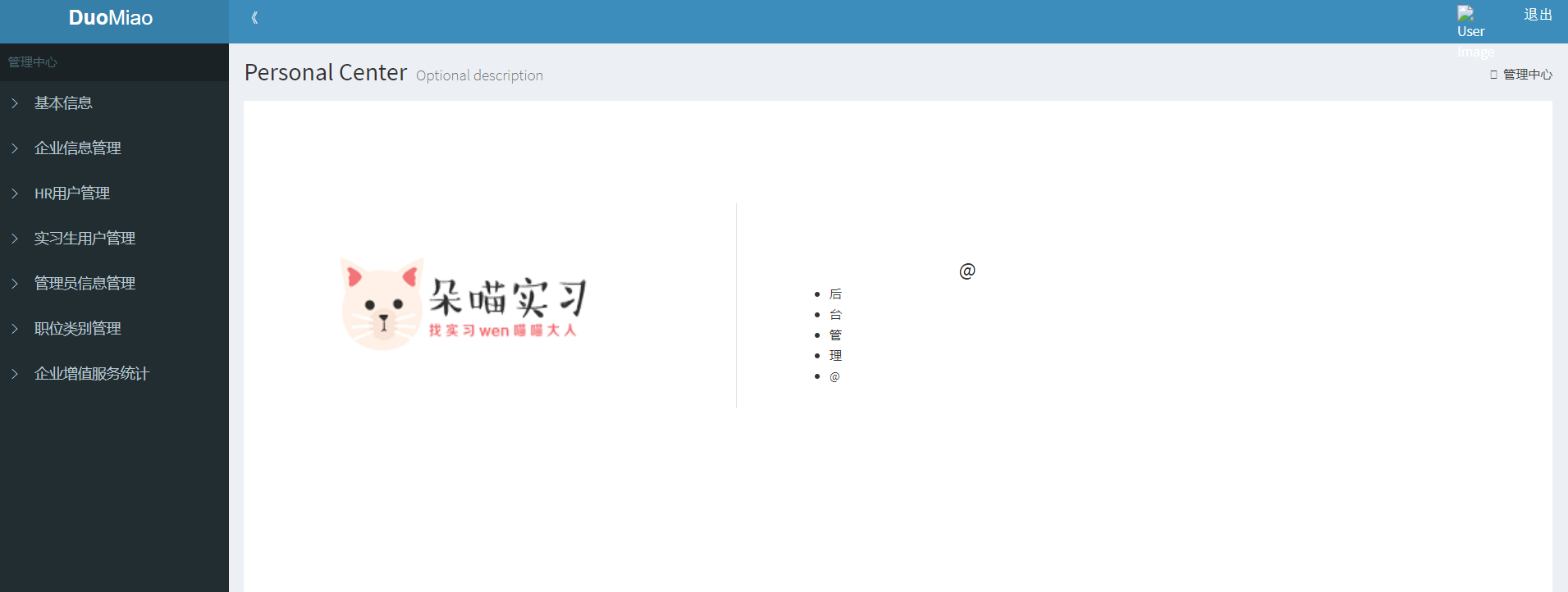


图4- 26 后台管理系统首页页面

1. 基本信息
2. 企业信息管理
3. HR用户管理
4. 实习生用户管理
5. 管理员信息管理
6. 职位类别管理

### 4.3.4公共页面界面设计

①功能概述：本平台的公共主页主要用于展示职位列表，提供搜索功能，展示热门职位和最新职位。

②如下图4- 27，图4- 28所示：

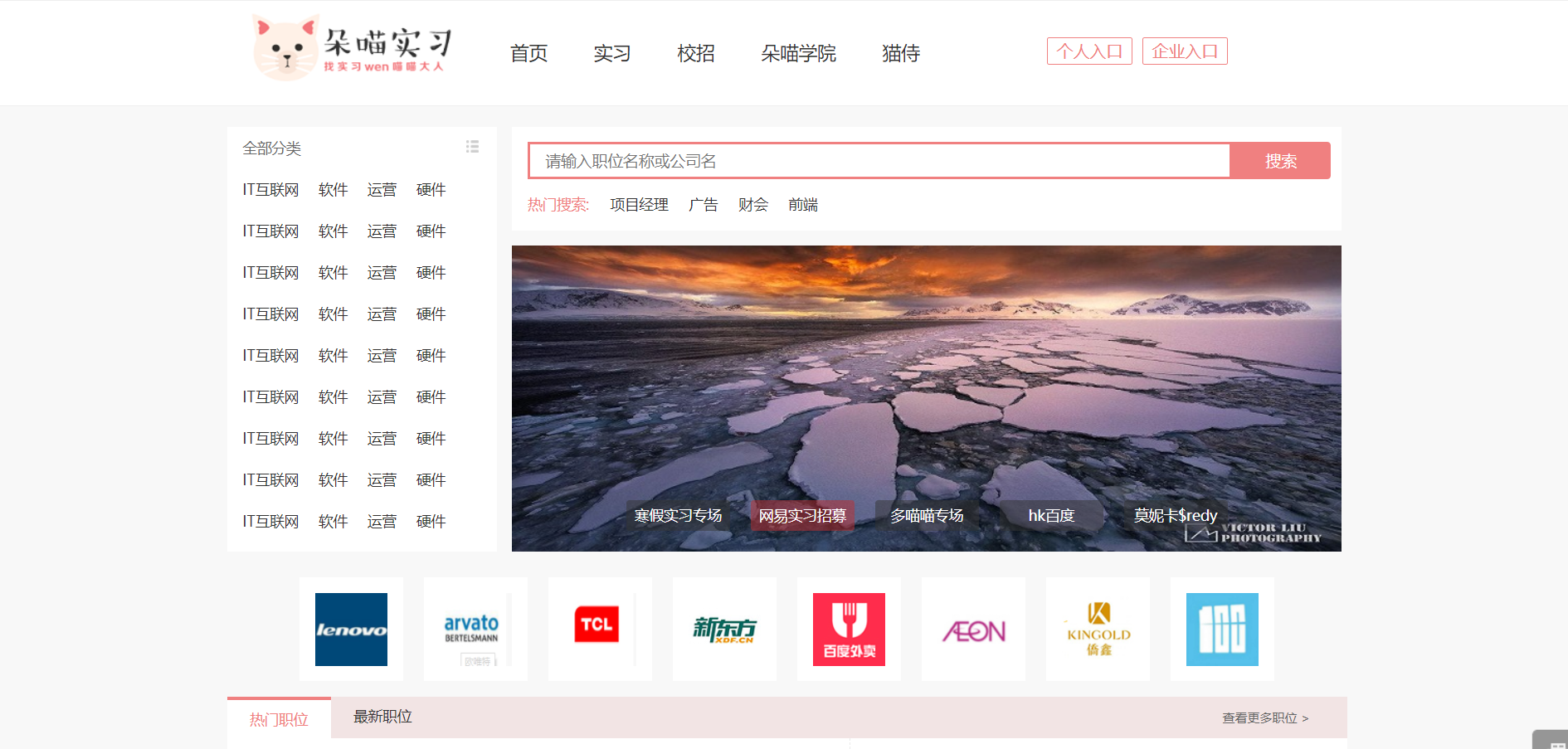


图4- 27 项目公共首页页面

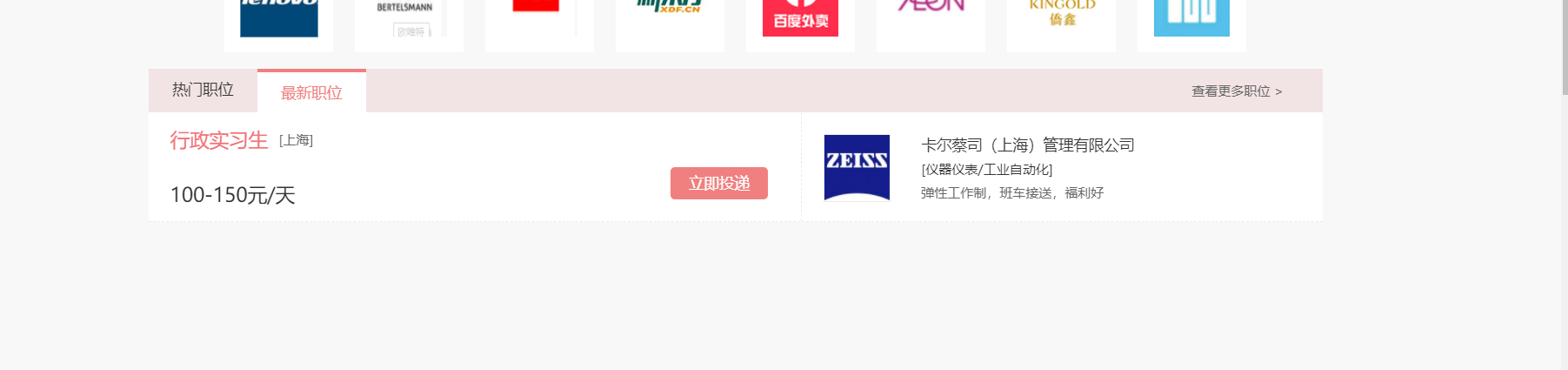


图4- 28 项目公共首页页面

③实现原理：

1. 职位列表展示部分

后台Controller层的listPositionCategory方法用于获取所有的职位类别信息，其内部实现是首先获取一级的职位类别，然后根据相应的职位类别获取二级职位类别，最后封装成BootsTable实例以json格式返回前端。前端数据展示在js代码中通过ajax的回调函数，对数据进行循环显示。

1. 搜索功能

PublishJobControl类的searchPubJods方法用map接收搜索条件参数[currentPage(当前页)，jobNameLike（职位名称全模糊），minSalary（每月最小工资），maxSalary(每月最多工资)，minEducationLev（什么学历），workPlace（工作地点）]，执行publishJobService.selectPublicMethodTwo (map)获取搜索记录，最后封装到AjaxResult中返回。

1. 展示热门职位

PublishJobControl类的getHotPubJob方法里面通过调用

publishJobService的selectHotPubJob()方法返回8个热门职位，以List<PublishJob>的json格式返回前端。

1. 展示最新职位

PublishJobControl类的getNewPubJob方法里面通过调用publishJobService的selectNewPubJob ()方法返回8个最新职位（查询数据库通过orderby ‘更新时间字段’ desc 排序后取前8条数据展示），最后以List<PublishJob>的json格式返回前端。

# 第五章 测试与分析

## 5.1 测试环境

这次测试在本机的虚拟机上进行。

硬件环境:

处理器I7-6700HQ

内存:4GB

硬盘:1TB

软件环境:

Windows10环境下Vmware安装的linux虚拟机

Linux CentOs

Tomcat7.0

Mysql

## 5.2 测试原理

本次测试采用黑盒测试，侧重于测试软件的基本功能是否运行稳定。

该测试方法本质在于不深究于系统的内部结构，而是从使用者的角度出发，根据系统规定的格式输入数据以此来检验输出的数据是否符合预期。它的优点是测试简单快速，缺点是测试结果依赖于测试用例，并且有可能出现漏测的功能。

在测试前需要罗列出测试用例，避免漏测。测试后需要记录数据尤其是出错的功能需要着重标记。

## 5.3 实例测试

首先运行项目:

(1)项目打成war包部署到linux服务器下tomcat的wepapp文件夹，解压。

(2)将数据库文件导入到linux服务器下的mysql数据库中。

(3)启动Tomcat服务器

(4)浏览器输入http:// 192.168.190.129:8080/duomiao/public/index

开始测试：

测试对象是针对实习生用户模块、HR用户模块、后天管理系统三大系统。

部分测试用例如下表5- 1所示

表5- 1 实例测试表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 测试项目 | 测试方法 | 测试结果 |
| 1 | 实习生注册 | 进入注册界面，填写不同的格式账号和密码，点击注册 | 不符条件的无法注册，符合条件的注册成功 |
| 2 | 实习生基本信息操作 | 修改用户名 | 只读无法修改 |
| 上传图像 | 当上传非图像文件时无效 |
| 3 | 实习生简历填写 | 名字不输，学历不输 | 提醒用户不能为空，并无法提交 |
| 按照要求范填写 | 提交后，提醒成功 |
| 4 | 删除 | 删除企业信息 | 提醒删除成功，数据库数据更新 |
| 5 | 查询HR用户 | 多条件查询 | 展示筛选出的数据 |
| 6 | 数据导入到录入窗体 | 导入每项数据 | 显示并计算出当前工作量 |
| 7 | 发布职位 | 乱填写 | 提示输入不符条件，无法提交 |
| 按照规定填写 | 提交成功，数据库添加数据 |
| 8 | 简历管理 | 删除职位的情况下 | 对应简历也被删除 |
| 职位保留的情况下 | 简历正常展示 |
| 9 | 职位管理 | 查询，删除职位 | 能删除数据 |

# 第六章 结论

通过这次毕业设计，深刻体会到采用spring mvc ，spring ， mybatis 三大框架爱整合的模式开发的优势，更低的成本，更快的效率，更清晰的项目架构这些都是在这个开发过程中切身的体会。以往我并未使用过该类型的MVC架构从头到尾的开发一个项目，最初只是采用原始的servlet+jsp模式开发一个简单的后台管理系统。由于之后自学这些框架后，发现其的使用广泛性和高效性后，决定将SSM框架应用到这次毕设中。

之所以会想到实习生招聘网站这个毕业设计主题，是自己本身就处于这个实习找工作阶段，自己也切身体会到工作不易。由于时间所限，本项目还有诸多瑕疵，因为项目本身比庞大所以需求方面也未考虑的十分细致，包括技术方面遇到的问题。不过经过这此项目的锤炼，遇到问题解决问题，学到了很多实际性的技术。

当然在公司的实习经历，对我做毕业设计也有很大帮助。我真切地感受到学校和公司对学习方式的极大不同。在公司通常都是通过不断在工作中去实现复杂需求，去攻克技术难点来提升自己，而这种方式恰恰是最锻炼人的，边实战边学习通常比学校的理论学习更加的快速且有意义，但是学校学的理论基础对日后的进一步提升也是非常重要的。这段是间，接触了struts2+spring+ibatis，spring+spring mvc+mybatis等不同模式的开发方式，在应用方面更加的熟悉了。也学习到团队开发的经验，知道更多的联合开发规范。

未来这类型的网站开发趋势都会朝彻底前后端分离方向发展，前端和后台互不干涉开发，后台只需要提供规范的接口即可，这样分工开来对项目的快速开发和部署尤为重要。当访问量大到一定程度时，通常需要解决高并发问题，这时候就会用到分布式架构，数据库更多优化。

# 参考文献

[1] 陈恒,楼偶俊,巩庆志,林徐. Spring MVC开发技术指南[M].北京:清华大学出版社, 2017：17-30.

[2] Josiah L. Carlson. Redis in Action [M].北京: 人民邮电出版社, 2015：20-30.

[3] 徐郡明. MyBatis技术内幕[M].北京：电子工业出版社, 2017:40-50.

[4]李杉,贾彦平,达虎.Mybatis逆向工程在JavaEE中的应用[J].通讯世界,2017(24):342.

[5]王鹤琴,汪炜玮,朱珍元.基于SSM框架技术的办公管理系统的研究[J].安徽警官职业学院学报,2017,16(03):118-122.

[6]吴东明,王丽娟.基于MySQL的中文全文搜索研究[J].电脑知识与技术,2017,13(32):7-8.

[7]宋成明.基于Shiro的某高校科研信息管理系统的设计与实现[J].智能计算机与应用,2017,7(04):62-63+66.

[8]张光平.Java多线程的分析[J].科技风,2018(01):71.

[9]荣艳冬.关于Mybatis持久层框架的应用研究[J].信息安全与技术,2015,6(12):86-88.

[10]Salvador Martínez,Valerio Cosentino,Jordi Cabot. Model-based analysis of Java EE web security misconfigurations[J]. Computer Languages, Systems & Structures,2017,49.

[11]Guido Barbaglia,Simone Murzilli,Stefano Cudini. Definition of REST web services with JSON schema[J]. Software: Practice and Experience,2017,47(6).

[12]Wang Bing,Dobosh Paul A,Chalk Stuart,Sopek Mirek,Ostlund Neil S. Computational Chemistry Data Management Platform Based on the Semantic Web.[J]. The journal of physical chemistry. A,2016.

[13]Aibek Sarimbekov,Lukas Stadler,Lubomír Bulej,Andreas Sewe,Andrej Podzimek,Yudi Zheng,Walter Binder. Workload characterization of JVM languages[J]. Software: Practice and Experience,2016,46(8).

[14]罗贵木. 基于微服务化的Web后台系统架构优化及实现[D].北京邮电大学,2017.

[15]北京市工商局网监处. 招聘网站履行审查义务相关问题探析[N]. 中国工商报,2017-09-05(003).

# 

# 致谢

毕业设计的结束也预示着大学生涯即将结束，我也将会投入到工作岗位中迈入新的生活，同时我想说在这次毕业设计中学到了很多以往未接触过的珍贵经验。

在这里我要感谢我的指导老师，在毕设开题的时候为我拟定毕业设计课题，并为我指定好设计方向和规划。开发过程中，老师对系统的见解和想法，都灌输到我的项目当中。最后在老师的帮助下完成这个项目以及毕业论文，整个过程学到了老师德严谨态度、创新意识等等。

同时要感谢平时遇到问题，帮助我解决的同学们，让我的项目开发变得更加顺畅。

其次要感谢我实习公司的前辈们，是他们让我感受到真实的企业项目开发，让我从一个懵懂的菜鸟在历练中成长。

最后感谢丽水学院机损及科学与技术的全体老师，四年来对我的教导和培育。