

Graduate Recruiting System Based on ThinkPHP Framework

Zhanyu Wang, Siling Feng*, Lu Li

College of Information Science & Technology, Hainan University, Haikou Hainan
Email: 1773309187@qq.com, *fengsiling2008@163.com, 1038745819@qq.com

Received: Jun. 8th, 2017; accepted: Jun. 25th, 2017; published: Jun. 28th, 2017

Abstract

The working principle of the ThinkPHP framework, Bootstrap framework and AJAX technology are introduced in detail, and the ThinkPHP framework that implements the MVC design pattern is introduced into the development of graduate recruitment system. The Bootstrap is used to develop a responsive layout page at the front-end; AJAX technology is used to achieve no Refresh update page, and concrete examples are combined to illustrate the application of the ThinkPHP framework to implement the MVC architecture design pattern, which embodies the superiority of the software framework in the development and application. The application is simple and the development efficiency is high.

Keywords

MVC, Recruitment System, ThinkPHP, Bootstrap, Ajax

基于ThinkPHP框架的毕业生招聘系统

王湛宇, 冯思玲*, 李 路

海南大学信息科学技术学院, 海南 海口
Email: 1773309187@qq.com, *fengsiling2008@163.com, 1038745819@qq.com

收稿日期: 2017年6月8日; 录用日期: 2017年6月25日; 发布日期: 2017年6月28日

摘 要

详细介绍了ThinkPHP框架的工作原理, Bootstrap框架以及AJAX技术, 将实现MVC设计模式的ThinkPHP框架引入到毕业生招聘系统的开发中, 在前端使用Bootstrap开发响应式布局网页, 利用AJAX*通讯作者。

技术实现无刷新更新页面，并结合具体的实例阐述了应用ThinkPHP框架实现MVC架构的设计模式，体现了软件框架在开发应用中的优越性，应用简单，开发效率高。

关键词

MVC, 招聘系统, ThinkPHP, Bootstrap, AJAX

Copyright © 2017 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

当前的网络技术飞速发展，生活的方方面面都与互联网有着一定的关系，联系方式由原来的书信变成了现在的电邮，支付手段不再局限于原来的金银纸币，电子支付也变成了常用的付款方式。可以说网络的发展方便了生活。在人才的竞争中，也不再局限于人才招聘市场的现场招聘，网络招聘异军突起，成为了各大商家和广大人才的交流中心。凭借着互联网的互通，快捷，网络招聘的优势突出，其低成本，信息量大，搜索便利成为了网招流行的关键因素。

由此可见，构建一个方便，快捷并且安全的网络招聘平台的必要性。本文就是基于以上原因，简化了部分流程。着重去实现了网络招聘中的应试者和招聘单位的需求，使得各方都能够及时的查询到自己心仪的职位，招聘到公司需要的人才，大大节省了人才招聘和应招时的花费。同时，本系统应用了当前较为流行的 ThinkPHP 框架和 AJAX 技术，在 Bootstrap 框架下构建的网页能够在各种平台上显示友好，充分展现了响应式布局的优势。

2. 系统实现的关键技术

2.1. ThinkPHP 框架

ThinkPHP 作为一款国产的 PHP 框架，以其快速、兼容和简洁的优势成为最为热门的 MVC 框架之一，支持多种服务器环境和多种数据库连接，简单易用，高度自动化的 CURD 操作极大简化了数据库操作，并且其开发效率高极大减轻了开发人员的负担。

在开发时，可以通过编写、配置入口文件，自动生成项目的目录[1]。清晰的项目结构目录为项目开发提供了很多便利。Conf 目录下储存配置文件，可用于对一些参数的更改，项目数据库的连接可以通过对配置文件的相关参数进行赋值即可完成对数据库的连接。Controller 目录下储存对 MVC 架构中的控制器文件，负责调度整个程序的运转。Model 目录下存储对 MVC 架构中的模型文件，主要负责处理业务数据。View 目录下储存 MVC 架构中的视图文件，负责呈现界面。

ThinkPHP 框架的开发流程为：

- 1) 项目命名并创建项目的入口文件
- 2) 完成项目的配置
- 3) 创建控制器类
- 4) 创建模型类
- 5) 创建模板文件

6) 运行和调试

其项目的目录结构如图 1。

2.2. MySQL 数据库

关系型数据库管理系统有很多，其中 MySQL 就是其中之一。关系型数据库是一种将数据保存在不同的表中，增强了数据的组织性，提高了数据的灵活性。在数据库语言的应用上，MySQL 应用的是用于访问数据库的最常用的标准化语言——SQL。这一数据库的特点突出，其体积小，源码开放，总体拥有成本低成为了一般中小型网站开发的首选数据库。其系统特性有：

- 1) MySQL 源码开放，使用是不需要支付其他费用。
- 2) 支持 Linux、Mac OS、Solaris、Windows 等多种操作系统。
- 3) 为多种编程语言提供了 API。这些编程语言包括 C、C++、Python、Java、Perl、PHP、Eiffel、Ruby 和 .NET 等。
- 4) 对于多线程的支持使得其能很好的利用 CPU 资源。
- 5) 对于查询算法的优化，能够很好的提高查询的效率。

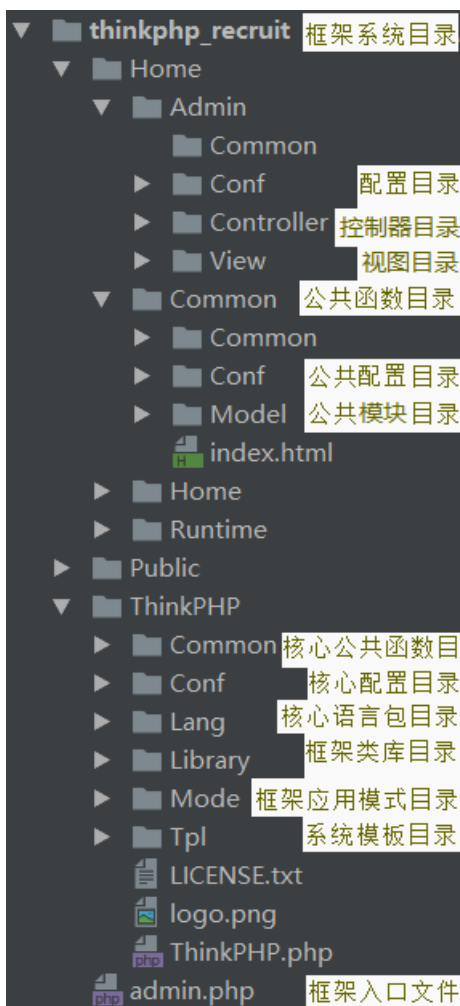


Figure 1. Structure chart: project directory

图 1. 项目目录结构图

6) 运用起来较为灵活，能单独运行于客户端服务器网络环境中，也可以作为嵌入到其他的软件中的一个库。

7) 提供多语言支持，常见的编码如中文的 GB2312、BIG5，日文的 Shift-JIS 等都可以用作数据表名和数据列名。

8) 对于 PHP 有极好的支持。

9) 管理、检查和优化数据库均有提供管理工具。

10) 能够处理数据量极大的大型数据库。

11) 支持多种存储引擎。

2.3. Bootstrap 框架

Bootstrap 作为一款前端框架，以其优秀的响应式布局设计，丰富的 HTML 和 CSS 组件，JQuery 插件等特点被广泛应用于前端页面的建设。

一方面它有精美的预设样式和 UI 设计。丰富的 class 样式，给 table, nav 等标签赋予了生动的表现，其内置的 Glyphicons 字体图标可以免费使用，应用便利。内置的 JQuery 插件可以通过 data 属性就能轻松使用，省去了写一行 js 代码的繁琐。

另一方面它在不同的设备和平台上都有很好的显示风格。基于 Bootstrap 创建的网站和应用能够有效地适配手机、平板、PC 设备。

2.4. AJAX 技术

AJAX 作为一种创建交互式网页应用的网页开发技术，能够快速创建动态网页，只需要通过后台与服务器及进行少量的数据交换，即可在不刷新整个网页的情况下，实现网页的异步更新。AJAX 技术使得 Internet 的应用程序更简洁，更高效。其工作方式如下图 2:

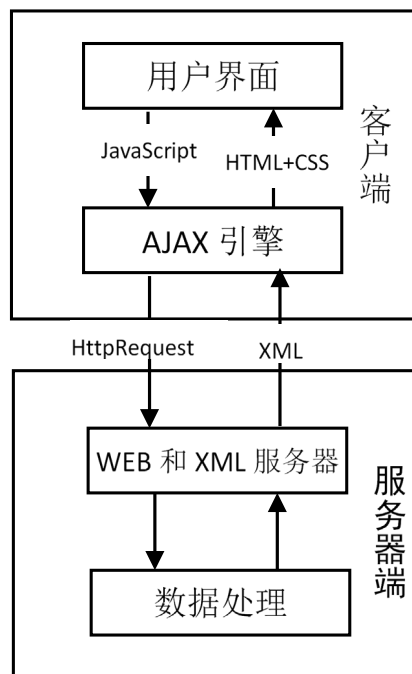


Figure 2. AJAX operation schematic diagram

图 2. AJAX 工作示意图

3. 系统需求分析

3.1. 系统概述

当前大学生求职问题已成为了当前社会的热点问题，全国高校面临的重点问题也是如何提高毕业生的就业率。过去的大学生就业主要就是通过现场的招聘会和人才交流会的途径来实现的，在互联网高速发展的今天，这一途径已经不能满足日益增长的毕业生数目，另外由于招聘会的限期和地域限制，让很多毕业生错过了与招聘单位的交流机会，一对一的交流对于毕业生和招聘单位尤为重要，所以，在线的交流平台，一对一的招聘网站就显得尤为重要，因其便捷，信息量大和即时性强的特点，成为了现今较为流行的人才招聘手段[2]。

3.2. 系统业务需求分析

3.2.1. 招聘信息发布流程

招聘系统要运行起来的首要条件是应该有企业发布招聘信息，然后待就业者才能去选择自己适合的岗位提交简历，在企业发布招聘信息之前，应该先有用人单位向人事反映需要应聘者需要达到的一些要求，比如说年龄，学历，工作经验等，接下来就是要确定招聘信息的内容，比如：单位名称，招聘的起止时间，需要的人数等。提交之后还应该给商家予以修改的机会，方便随时修改招聘条件，以便满足从业需求，最后提交审核。其示意图如图 3 所示：

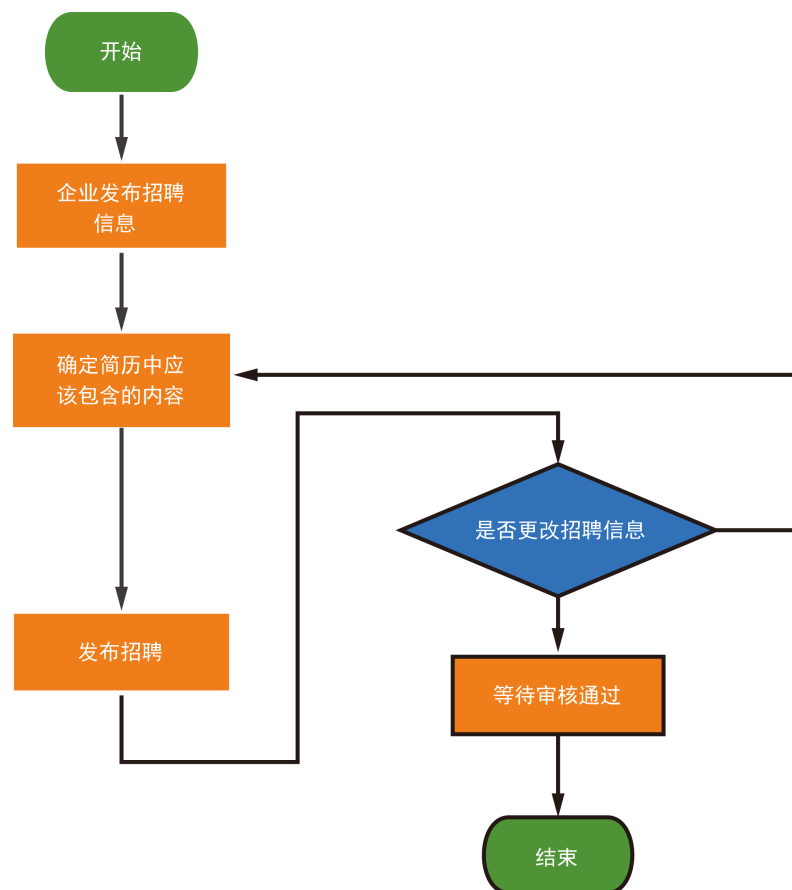


Figure 3. Recruitment flow chart

图 3. 招聘流程图

3.2.2. 应聘流程

当企业发布了招聘信息之后，应聘者应该能即时的看到招聘信息，并编辑自己的简历，投递简历。应聘者首先会浏览网站上的招聘信息，根据自己想要找到的职位和自己的个人状况来选择心仪的公司，根据其招聘要求填写简历，最后提交审核，如果需要修改简历，可随时进行查看改动，并可以即时的查看自己的审核状态。其示意图如图 4 所示。

3.2.3. 审核的流程

定期检查公司所发布的招聘公告，并当到达招聘信息的截止日期时，查看人事部门收到的所有简历，进行筛选，通过用人部门对简历的筛选，选出待面试的应聘者，并将面试通知发送给应聘者，如果应聘者面试通过，待招聘人数达标之后，删去招聘公告。其示意图如图 5 所示。

3.2.4. 系统前台业务流程

访客浏览毕业生招聘系统时，如果有账号可以直接登陆，反之，则需进行注册，根据个人用户和商家用户来个性化注册界面。注册登陆之后，可以根据自己的用户类型来显示特定的界面。管理员登陆毕业生招聘系统后，显示其数据库中的现有的商家和个人用户的相关信息，可以进行管理。其示意图如图 6 所示。

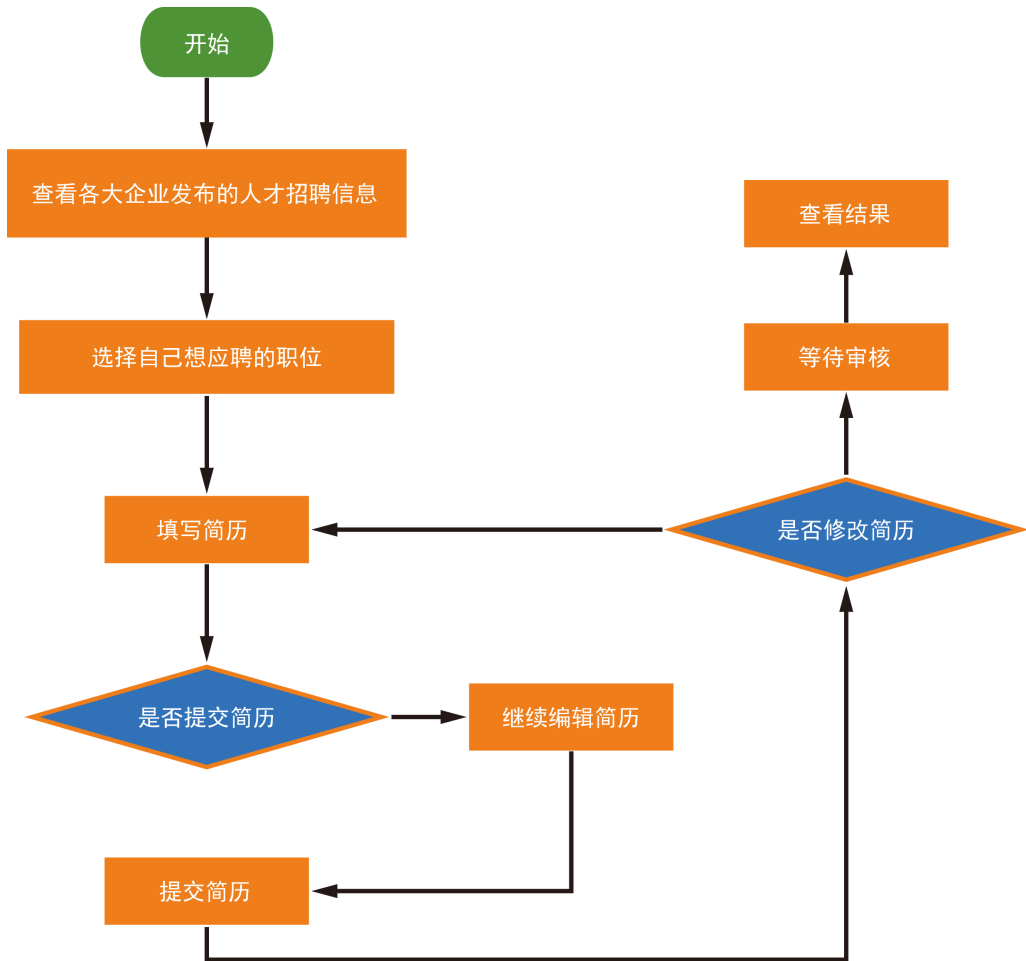


Figure 4. Apply flow chart

图 4. 应聘流程图

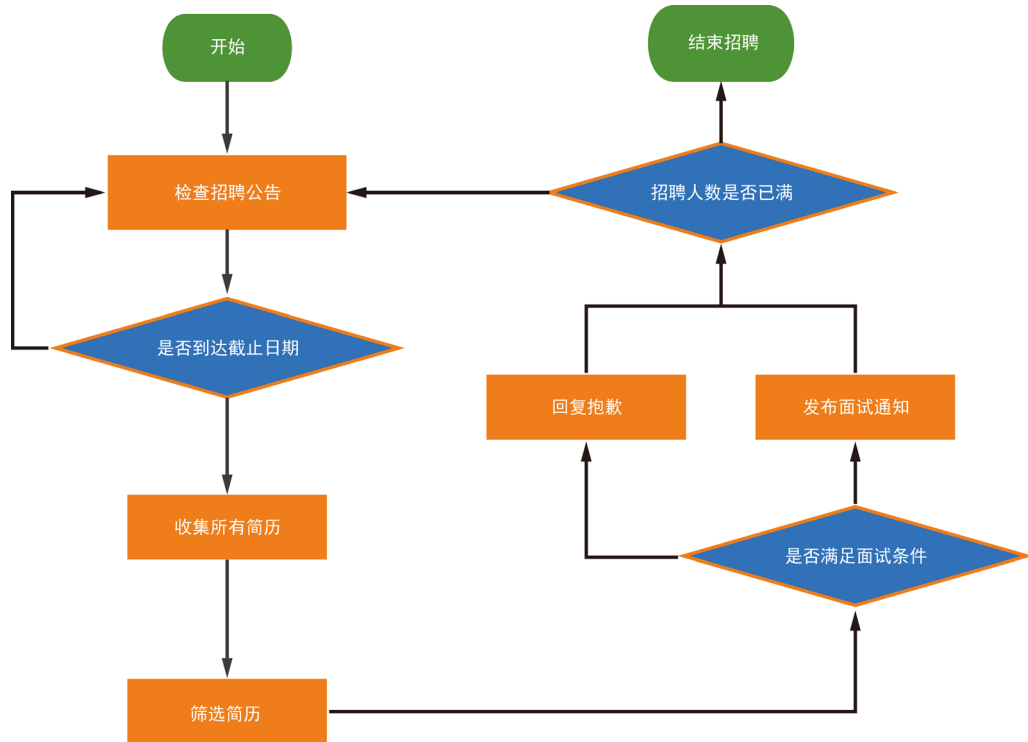


Figure 5. Audit flow chart
图 5. 审核流程图

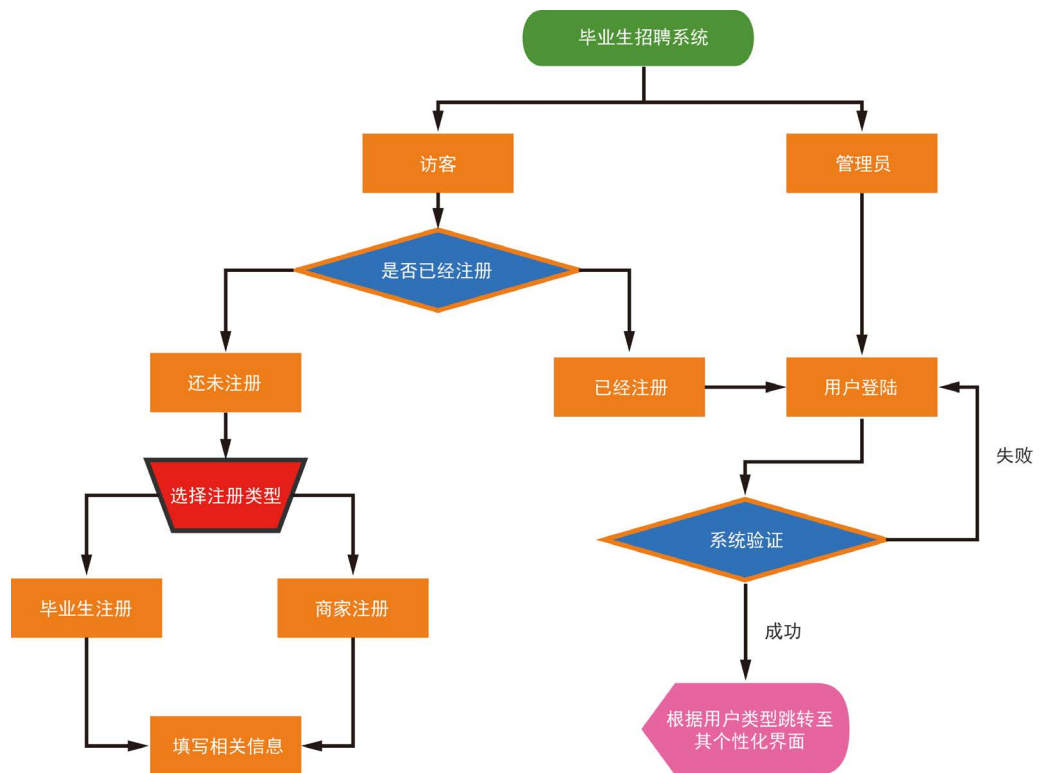


Figure 6. The flow chart of front desk business
图 6. 前台业务流程图

3.3. 系统功能需求分析

3.3.1. 招聘管理功能需求

招聘管理作为整个系统的核心内容，可将其大致分为两个部分：

- 1) 招聘信息提交，发布和审核：当公司的用人部门需要招聘人才时，可以在公司的招聘系统中发布招聘公告，由公司的人事部门审核。之后需要修改或者拓展的时候，还可以随时进行更改。
- 2) 人才筛选：人才筛选管理功能是基于发布了招聘公告之后，对应聘者投递的简历所进行的一系列的操作。其过程是由公司的人事部门进行的，筛选完成之后，分别对通过了筛选的应聘者发布面试通知。

3.3.2. 人才资料管理功能需求

人才资料主要管理与求职者相关的信息，分为人才的分类管理，人才基本信息管理，综合查询等。

- 1) 人才分类管理：由于每个公司招聘的人才方向不一，在庞大的数据库中筛选时会十分费时，为了有效对数据进行筛选分析，管理维护，系统设置有人才分类模块，人才分类信息可以根据需求由用户自定义进行维护。
- 2) 人才基本信息管理：这个功能的实现主要有赖于应聘者自己对自己的个人信息进行的更新和维护，另一方面，也可以由管理员进行操作，对违规的信息进行定时的清理。
- 3) 综合查询：这部分主要是为了在数据不断扩大之后，方便根据不同的条件来进行查询，比如说：性别，技能，创建时间等。

3.3.3. 权限管理功能需求

由于整个系统有三种人员要登陆使用，分别是发布招聘公告的公司，应聘职位的毕业生和管理员。所以有必要通过一定的手段来区分登陆用户的身份。

- 1) 角色和权限管理：权限控制是基于不同的角色而设定的，而角色又是基于不同的权限，角色具有系统操作的权限，不同的角色会有不同的权限，这些在登陆的时候应该基于判断，从而赋予不同系统管理功能。
- 2) 用户和角色管理控制：在新增用户的时候，应该分出是商家注册还是毕业生(应聘者)注册，并在数据库中给予给定的标识，这样在下次登陆的时候就能由系统根据其角色来提供特定的系统操作功能。

4. 系统总体设计

4.1. 系统总体模块设计

采用结构化设计方法来实现系统总体功能，将整个系统划分为几个功能模块，正确处理模块之间的关系和模块内部的联系，明确其数据的调用关系和联系，通过对一个个模块的具体实现来完成整个系统的功能。其系统功能图如图 1 所示。

本系统主要按照用户的不同分为了三个模块：毕业生模块，商家模块和管理员模块。其中毕业生模块主要实现的功能有查看已经投递的简历，填写/管理个人简历和浏览商家的招聘公告等，商家模块主要实现的功能有填写/管理招聘公告，查看已经收到的简历和对应聘者发送面试/落选通知等，管理员模块主要实现的功能有管理毕业生信息，管理公司信息 and 添加/删除用户等。其示意图如图 7 所示：

下面分述其具体的模块内容：

1) 毕业生模块：

当毕业生注册为系统用户后，毕业生模块提供了用户一系列的操作，如毕业生可以通过这个模块创建简历，投递简历，搜索职位等。用户基本资料管理，简历管理，岗位查询都应该是这一模块的组成成分。

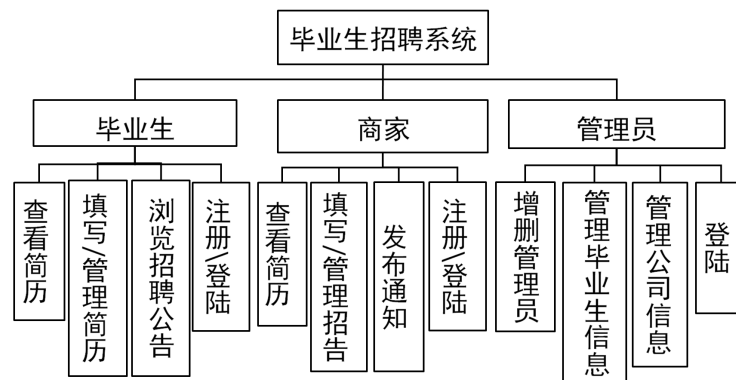


Figure 7. System function diagram

图 7. 系统功能图

求职者的基本资料：个人求职者可以查看和修改自己的个人信息，包括用户名，性别，出生日期，联系方式等。

求职者的简历管理：这里可以让求职者方便的创建自己的简历，在填写时，需要的信息有个人基本信息，教育背景，工作经验等。系统对简历编号，方便管理，在简历创建成功之后，用户可以对简历进行查看，修改，删除等操作。

求职岗位的查询：求职者可以在这个模块查看近期企业发布的招聘公告，求职者可以通过企业名称，职位名称，工作地点等来筛选自己想要的岗位[3]。

2) 企业会员模块：

当企业用户在首页完成注册操作之后，就可以使用企业会员模块，企业通过此模块可以进行的操作有：发布企业近期招聘公告，管理发布的招聘公告，查询求职者的信息等。企业会员模块包含的功能有：管理企业基本信息，管理招聘公告，查询求职者信息等。

管理企业基本信息：企业通过该功能模块时，可以对企业的基本信息进行修改。基本信息包括：企业名称，所在地，联系人，联系方式等。

管理招聘公告：企业的招聘公告由企业的人事部门来发布，在发布之后可以随时进行修改，更新，以便应聘者可以及时了解到企业的新情况。

查询求职者信息：企业可以通过查询模块来实现对近期收到简历的筛选，如可以根据求职者的性别，学历，面试岗位等来搜索。

3) 管理员管理模块：

该模块是对整个系统网站进行控制和设置的一个平台。管理员在这个模块中可以进行的操作有：管理个人会员信息、管理企业会员信息，添加/删除用户，对系统的基本数据进行分析等。

管理个人会员信息：通过这个模块，管理员可以对已经注册的个人会员信息进行查询、审核、删除等操作。

管理企业会员信息：通过这个模块，管理员可以对已经注册了的企业的信息进行查询、审核、删除等操作。

管理用户：这里管理员可以增加用户，也可以删除用户，通过这个模块可以增加用户，也可以删除违规的用户。

数据分析：可以在管理员模块中加入一个数据分析的模块，来进行一些简单的数据分析，比如，通过折线图来查看最近的个人用户注册的情况等。

4.2. 数据库设计

由系统的需求分析，在这里共构建了 5 个数据表，分别是记录用户基本信息的 md-userInfo，个人简历信息的 md-resume，公司基本信息的 md-compInfo，职位基本信息 md-workInfo 的和应聘职位记录表 md-applyInfo。

得到系统数据库的概念模型为图 8 所示。

得到系统数据库逻辑模型为：

- 1) 用户基本信息：{ID, 类型, 用户名, 密码, 性别, 年龄, 邮箱, 电话, 更新时间, 添加时间}。
- 2) 个人简历基本信息：{ID, 教育程度, 工作经验, 个人简历, 专业技能}。
- 3) 公司基本信息：{ID, 类型, 名称, 密码, 邮箱, 电话, 邮编, 地址}。
- 4) 职位基本信息表：{ID, 职位名, 人数, 联系人, 联系方式, 描述}。
- 5) 应聘职位记录表：{职位 ID, 个人简历 ID, 有无参加面试资格}。

数据表结构图如图 9 所示。

5. 系统实现

5.1. 系统的基础配置

5.1.1. 系统环境搭建

本系统采用 ThinkPHP 的三层架构，开发工具使用了 phpstorm，数据库 mysql 版本为 5.5.53，服务器采用 Apache，于 Windows 操作系统下进行开发使用，其前端框架使用了 Bootstrap，兼容大部分浏览器，其中包括 Chrome、Safari、Firefox、Internet Explorer (IE9 以上完美支持)和 Opera。

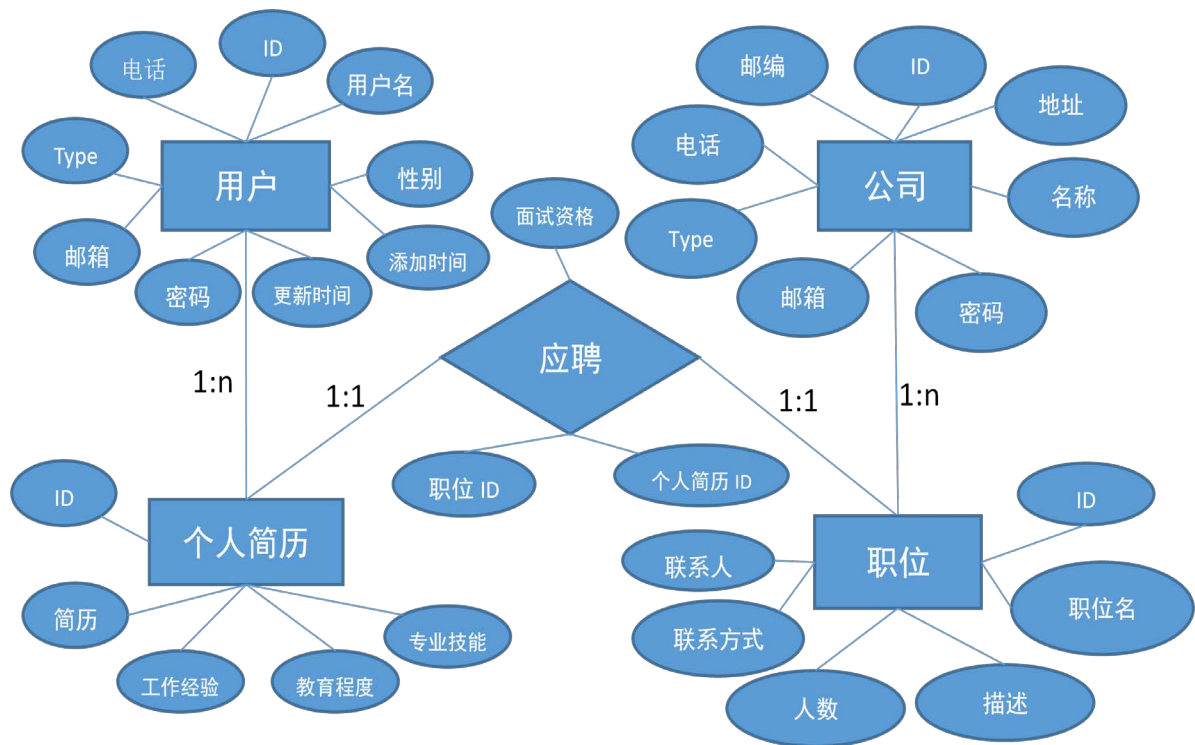


Figure 8. Entity-relation diagrams

图 8. E-R 图

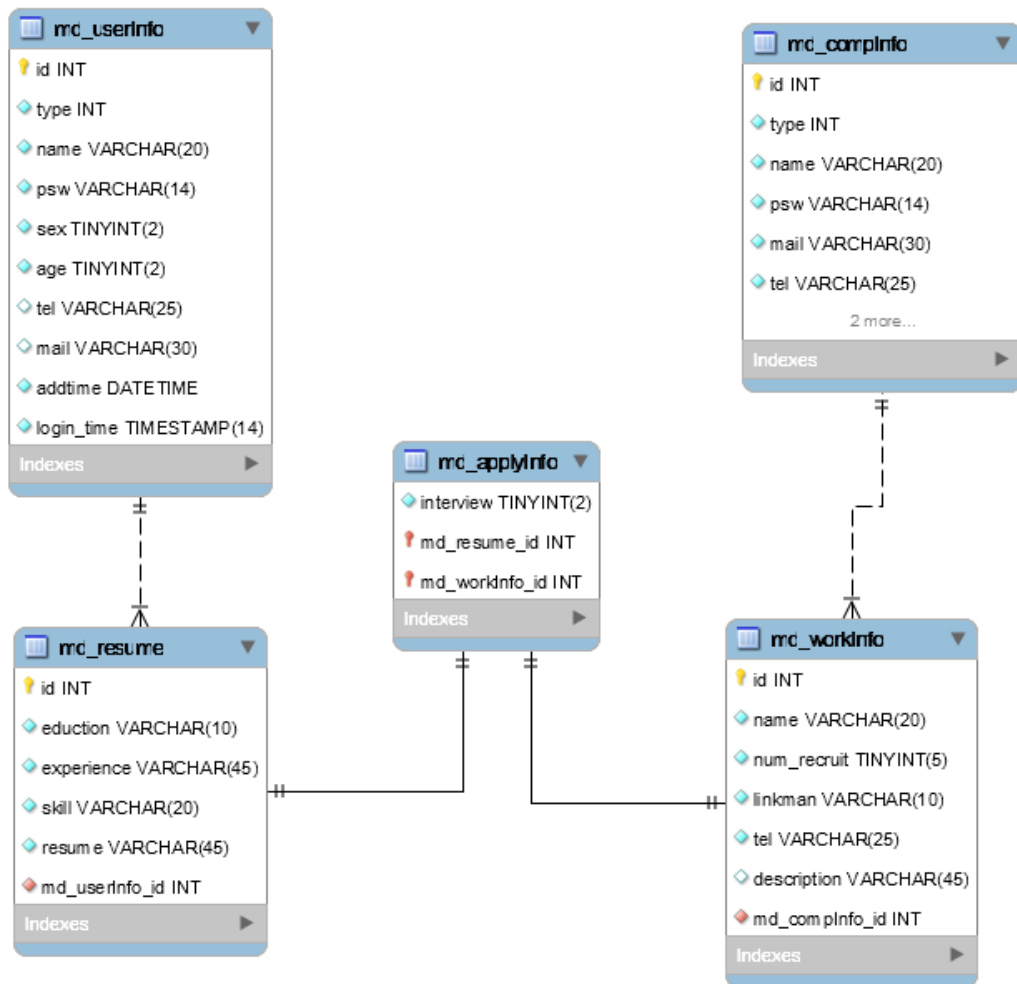


Figure 9. The structure chart of data table
图 9. 数据表结构图

5.1.2. 数据库的连接配置

本次系统开发使用的是 ThinkPHP 框架和 MySQL 数据库，由于 ThinkPHP 本身对 MySQL 数据库兼容性好，并且配置简单，只需要更改一下 config.php 这个项目下的配置项的值即可，所以十分简洁。具体配置如下：

```
//配置项 => '配置值'
'URL_MODEL'=>'0',//URL 模式
'DB_PREFIX'=>'md_',//数据库表前缀配置
'DB_TYPE'=>'mysql',//数据库类型
'DB_HOST'=>'localhost',//数据库的地址
'DB_NAME'=>'recruit',//数据库名
'DB_USER'=>'root',//数据库登录名
'DB_PWD'=>'root',//数据库登录密码
'DB_PORT'=>'3306',//数据库端口
'MD5_PRE'=>'modai',//定义用于 MD5 加密 psw 的后缀
```

之后就可以运用 TP 框架提供的操作函数直接连接到数据库进行操作。

5.2. 主页模块

系统的主界面如图 10 所示。

通过主页模块实现的功能为：根据 session 中的 User 的 type 来渲染模板，session 不存在的话就跳转到登陆主页。

其前端的设计大致为：头部放置网页 logo 和导航栏，中间的内容为一个轮播图，底部有一个版权声明。为了能在各种不同的设备上达到一个比较好的视觉效果和使用体验，运用了现今较为成熟的 bootstrap 前端框架，其优秀的响应式布局，显示效果和较为丰富的插件，为之后的模板设计提供了很大的便利，并且由于其流式栅格系统构成的响应式布局使得网页在电脑上和各个移动端都有很好的显示效果。

其后端实现代码为：

```
class Index Controller extends Controller {
public function index(){
//判断用户是否登陆过，通过 session
if(isset($_SESSION['User'])&&$_SESSION['User']!="){
if($_SESSION['User']['type']==1){
$this->redirect(U('Admin/index/index'));}
if($_SESSION['User']['type']==2){
$this->redirect('BysIndex/index');//显示后台页面}
if($_SESSION['User']['type']==3){
$this->redirect('SjIndex/index');//显示相应的后台页面}
}else{$this->display('index/index');}}
```

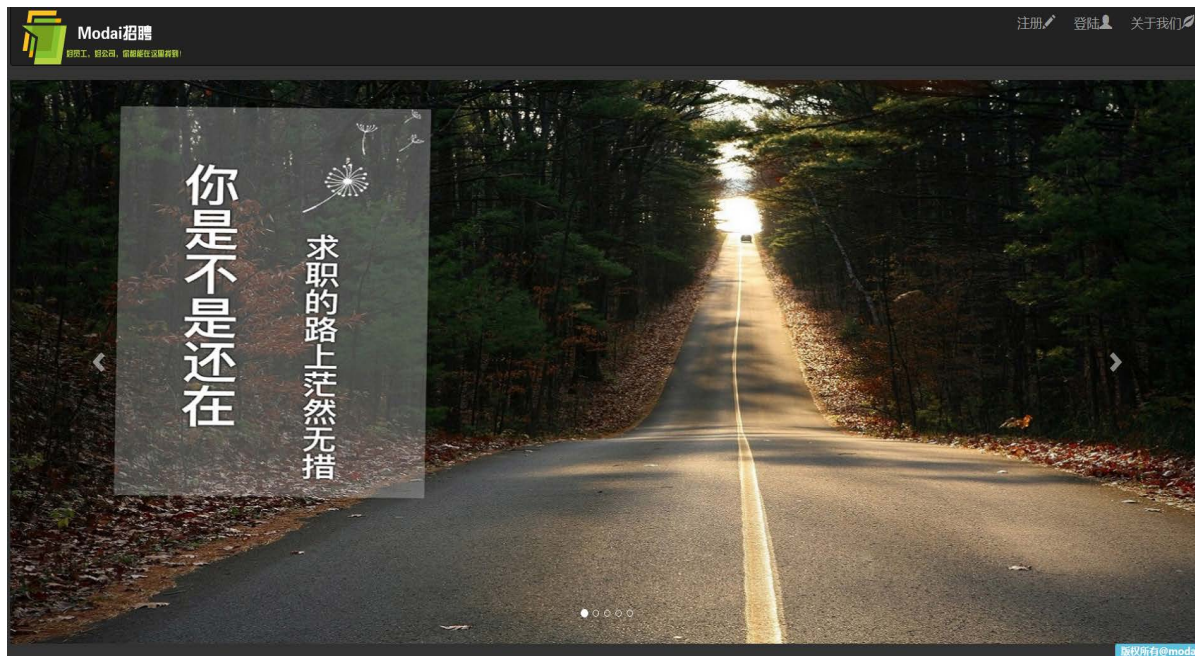


Figure 10. Homepage figure

图 10. 主页图

5.3. 系统注册模块

前端页面的编写使用了 bootstrap 这个前端框架，首先编写了头部的导航栏，之后进行了注册的表单的设计，其中 nav 导航栏是通过 TP 框架中的 include 标签来引入的。

后端的工作主要就是进行对提交的注册数据进行一定的防注入处理和加密后存入数据库，在注册成功之后跳转至相应的首页并存入 session 变量，反之，跳出错误界面，提示重试。

5.4. 毕业生模块

毕业生主页使用了一个模板文件，其中的各个页面的跳转是通过锚点实现的。为了实现动态加载网页，使用了 AJAX 技术无刷新动态生成页面。

其左侧导航栏是通过 bootstrap 的 tabbable 选项卡和 panel-group 手风琴切换嵌套应用实现的，头部的导航栏是通过 navbar 等 class 属性对 nav 标签进行样式控制实现的。其中头部导航的下拉菜单中的跳转应用了 js 中的 click 方法。毕业生主页如图 11 所示。

其后端的实现功能主要是页面的赋值和渲染，具体代码如下：

```
public function index(){
    if(isset($_SESSION['User'])&&$_SESSION['User']!=""){
        $User=session('User');
        $name=$User['name'];
        $data=D('User')->getUserByName($name);
        $id=$data['id'];
        session('id',$id);
        $num=M('resume')->where('md_userInfo_id="'.$id.'")->count();
        $data=array_merge($data,array('num'=>$num));
        $this->assign('data', $data);}
    $this->viewsume();
    $this->display('index_personal/bys_index');
```



Figure 11. Graduate home page figure

图 11. 毕业生主页图

毕业生的个人信息修改前端实现主要是通过一张提交表单，表单中默认填写有现在的登陆者的一些基本信息，方便用户进行订正，更改。后端的主要功能就是更新当前用户的数据库中的相关字段。

毕业生进行工作的查询时，可以选择搜索条件，当无条件的时候点击搜索按钮可以搜索全部的工作信息，之后点击“show details”后可以显示搜索的结果。其搜索结果图如图 12 所示。

前端的实现是通过一个判断条件的下拉菜单，一个搜索值填写框，搜索按钮和显示详情的按钮组成，其中的搜索值填写框通过 JS 来根据条件的不同，对其进行 disabled 与否的切换。搜索结果是由一个列表表单组成，其中操作一栏有一个按钮组，用于显示工作详情和填写简历。

后端的实现包括两个部分，一个部分是查找和显示搜索结果的方法，另一个就是通过 AJAX 实现的异步处理数据，实现了无刷新页面的渲染，通过这个方法很好的解决了 TP 中的 display 方法无法渲染当前页面的问题，通过对弹出层的赋值和页面生成实现对详情的展示。

后端的核心代码如下：

1. 查找功能和显示结果功能的实现：

```
public function findwork(){
    $searchdata=array(
        'condition'=>$_POST['condition'],
        'value'=>$_POST['value']);
    if($searchdata['condition']==0){
        $data= M('compwork')->select();
        return show(1,'查询成功!',$data);
    }
}
```



Figure 12. Search result chart
图 12. 搜索结果图

```

}elseif ($searchdata['condition']==1){
    $data= M('compwork')->where
('compname="'. $searchdata['value'].'"')->select();
    return show(1,'公司信息查询成功! ', $data);
}elseif ($searchdata['condition']==2){
    $data= M('compwork')->where
('workname="'. $searchdata['value'].'"')->select();
    return show(1,'职位查询成功! ', $data);
}else{return show(0,'查询出错! ');}
//显示工作列表
public function showdetail(){
    $wid=$_POST['wid'];
    $compname=M('compwork')->where
('wid="'. $wid.'"')->find();
    $workinfo=D('work')->getWorkById($wid);
    $work=array_merge($workinfo,array('compname'=>$compname));
return show(1,'查看成功! ', $work);}

```

2. AJAX 实现详情显示和搜索结果的显示，以搜索结果的异步数据传输为例：

```

search:function () {
var condition=$( 'select[name="condition"]').val();
var value=$( 'input[name="value"]').val();
if(condition!=0&&!value){ dialog.error("请输入要查询的值! ");}
varurl="index.php?m=Home&c=BysIndex&a=findwork";
var data={'condition':condition,'value':value};
//执行异步请求$.post
$.post(url,data,function (result){
    if(result.status==0){return dialog.error(result.message);}
    if(result.status==1){
        var info=result.data;
        var html="";
        for(vari=0;i<info.length;i++){
            if(i%2==0){html+="629
```

```

getElementById('+send'+).click()</button>;
    html+=' </div></td></tr>';}
document.querySelector('#result_show').innerHTML=html;}},'JSON');//AJAX 的 post 方法}

```

这里的 jquery 的 \$.post(url, data, function(result), 'JSON') 方法中的参数 url 地址值为向后台传输数据的地址，data 值为页面中获取到的条件和值所组成的数组，function(result) 方法中包含了对 result 这一数组中的 status 值的判读和对 data 值的写入，进而进行对相应的 div 元素进行赋值和 html 代码的写入，整个后台处理后的结果以 JSON 数组的形式进行返回。

5.5. 管理员模块

为了能让管理员更为直观的看到后台数据的相关情况，优化用户的体验，在毕业生的图表中主要设计了一个年龄占比的扇形图和一个统计不同性别人数的条形图，在商家的图表中主要设计了商家发布的职位数的条形图。统计图表如图 13 所示。

前端实现是通过引入 echart 插件，在预设好的 div 元素中进行图形的绘制。绘制出的图形在交互上十分友好，鼠标悬停可以看到具体的数目和信息。

后端主要的工作是将后台的数据进行统计整理之后，以 JSON 数组的形式传给执行图形绘制的操作函数，进而进行绘图。

以下为 TP 框架的相关操作的代码(毕业生图表相关后端操作):

```

public function postAgedata(){
    $age=D('User')->where('type=2')->field('age,count(age)')->
group('age')->select();
    $result=array();
    foreach ($age as $k=>$value){
        $result[$k]['value']=$value['count(age)'];
        $result[$k]['name']=$value['age'].'岁';}
    exit(json_encode($result));
}

```

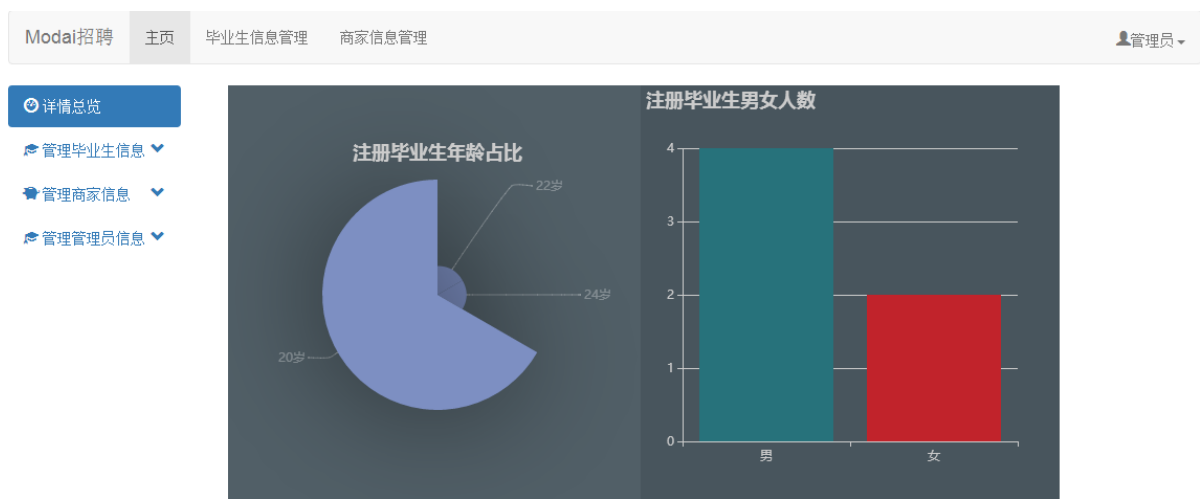


Figure 13. Statistical graph

图 13. 统计图表


```
    }//扇形图
public function postSexdata(){
    $sex=D('User')->where('type=2')->field('count(sex)')->group('sex')->select();
    $result=array();
    foreach ($sex as $value){
        $result[]=$value['count(sex)'];}
    exit(json_encode($result));}//条形图
```

6. 结束语

本文利用基于 ThinkPHP 框架的 MVC 设计模式构建了毕业生的网络招聘系统,其前端页面的开发利用了支持响应式布局的 Bootstrap 框架,为了实现根据用户的操作构建页面,使用了 JQuery 中的 AJAX 操作实现了无刷新更新页面。系统运行良好,可参见网址[4]。

基金项目

海南省自然科学基金项目(20156226);海南大学科研启动基金项目(kyqd1533)。

参考文献 (References)

- [1] 王俊芳,李隐峰,王池. 基于 MVC 模式的 ThinkPHP 框架研究[J]. 电子科技, 2014(4): 151-153+158.
- [2] 孟丽群. 大学生求职招聘系统开发的意义与应用[J]. 科技情报开发与经济, 2015(4): 139-140+144.
- [3] 魏晓海. 基于 Web 的人才招聘系统的设计与实现[D]: [硕士学位论文]. 吉林大学, 2015.
- [4] 毕业生管理系统. <http://modai.ecdus01.win>

期刊投稿者将享受如下服务:

1. 投稿前咨询服务 (QQ、微信、邮箱皆可)
2. 为您匹配最合适的期刊
3. 24 小时以内解答您的所有疑问
4. 友好的在线投稿界面
5. 专业的同行评审
6. 知网检索
7. 全网络覆盖式推广您的研究

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: csa@hanspub.org