

皖江工学院
WANJIANG UNIVERSITY OF TECHNOLOGY



毕 业 设 计（论 文）

题目： 校友录管理系统的设计与实现

专业年级 计算机科学与技术 2017 级

学 号 170310223

姓 名 路稳

指导教师 林金珠

评 阅 人 王杨

2021 年 6 月

中国 马鞍山

郑 重 声 明

本人呈交的毕业设计（论文），是在指导老师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本设计（论文）的研究成果不包含他人享有著作权的内容。对本设计（论文）所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本设计（论文）的知识产权归属于培养单位。

本人签名：_____

日期：_____

摘要

校园是个结交朋友、分享生活、分享经验的好地方。在过去，同学们只能口头分享自己的经验，这大大的限制了校园里的交流。但如今，随着科技的发展，互联网已经成为了主流。如果有一个大家都能够分享自己种种事迹的平台，这不仅可以促校友之间的交流，也能形成一起学习交流的校园氛围。

本文讨论的系统就是以互联网为基础打造的一个校友录系统。该系统参照了现流行的前后端分离技术，基于前端的 Vue 框架和后端的 SSM 框架而开发的。以便后期维护更加简洁、方便。作者在该系统开发中所承担的任务包括校友公告管理、社区管理、校友情况发展管理和用户管理等模块。公告管理可以增删改查一些公告信息，社区管理可以让同学们发一些动态评论等等，大大促进了校园的交流。该系统以简洁的界面为用户展示了整个校友录管理功能。

关键词：学校，分享，信息化，互联网，SSM

ABSTRACT

The campus is a great place to make friends, share life, and share experiences. In the past, students could only share their experiences orally, which greatly restricted the communication on campus. But now, with the development of technology, the Internet has become the mainstream. If there is a platform where everyone can share their various deeds, this will not only promote exchanges between alumni, but also create a campus atmosphere for learning and communicating together.

The system discussed in this article is an alumni record system based on the Internet. The system is developed based on the front-end Vue framework and the back-end SSM framework with reference to the popular front-end and back-end separation technology. So that the later maintenance is more concise and convenient. The tasks undertaken by the author in the development of the system include alumni announcement management, community management, alumni development management, and user management. Announcement management can add, delete, modify and check some announcement information, and community management can allow students to post some dynamic comments, etc., which greatly promotes campus communication. The system displays the entire alumni record management function for users with a simple interface.

Key words:School,Share,Informatization,Internet,SSM

目录

| | |
|------------------------|-----|
| 摘要..... | I |
| ABSTRACT..... | II |
| 目录..... | III |
| 第 1 章 绪论..... | 1 |
| 1.1 课题背景..... | 1 |
| 1.2 课题研究意义..... | 1 |
| 1.3 国内外研究现状..... | 2 |
| 1.4 主要内容..... | 3 |
| 1.5 论文组织..... | 3 |
| 第 2 章 系统相关技术..... | 5 |
| 2.1 前端技术..... | 5 |
| 2.1.1 Vue CLI..... | 5 |
| 2.1.2 Vue..... | 5 |
| 2.1.3 Webpack..... | 5 |
| 2.1.4 Axios..... | 5 |
| 2.1.5 Babel..... | 5 |
| 2.1.6 Element..... | 5 |
| 2.2 后端技术..... | 6 |
| 2.2.1 SSM 框架..... | 6 |
| 2.2.2 Spring boot..... | 6 |
| 2.2.3 Druid..... | 6 |
| 2.2.4 Maven..... | 6 |
| 2.3 软件安装..... | 6 |
| 2.3.1 JDK 安装..... | 6 |
| 2.3.2 IDEA 安装..... | 10 |
| 2.3.3 WebStorm 安装..... | 11 |
| 2.3.4 MySQL 安装..... | 13 |
| 2.3.5 Node.js 安装..... | 14 |
| 2.2 环境搭建..... | 17 |
| 2.2.1 前端环境搭建..... | 17 |
| 2.2.2 后端环境搭建..... | 18 |
| 第 3 章 需求分析..... | 21 |
| 3.1 总体目标..... | 21 |
| 3.2 可行性分析..... | 21 |
| 3.2.1 技术可行性..... | 21 |
| 3.2.2 操作可行性..... | 22 |

| | |
|--------------------------------|-----------|
| 3.2.3 运行可行性..... | 22 |
| 3.3 功能性需求..... | 22 |
| 3.4 非功能性需求..... | 23 |
| 第4章 概要设计..... | 24 |
| 4.1 数据库设计..... | 24 |
| 4.2 系统总体设计..... | 24 |
| 4.3 环境设计..... | 25 |
| 第5章 详细设计..... | 26 |
| 5.1 数据库设计..... | 26 |
| 5.1.1 用户表 (user) | 26 |
| 5.1.2 动态表 (talk) | 26 |
| 5.1.3 评论表 (comment) | 27 |
| 5.1.4 公告表 (announcement) | 27 |
| 5.2 流程设计..... | 28 |
| 5.2.1 登陆功能流程设计..... | 28 |
| 5.2.2 注册功能流程设计..... | 28 |
| 5.2.3 退出功能流程设计..... | 28 |
| 5.2.4 公告管理流程设计..... | 28 |
| 5.2.5 社区功能流程设计..... | 29 |
| 5.2.6 校友状况功能流程设计..... | 30 |
| 5.2.7 查看公告功能流程设计..... | 30 |
| 5.2.8 修改个人信息功能流程设计..... | 31 |
| 5.2.9 修改密码功能流程设计..... | 31 |
| 5.2.10 用户管理功能流程设计..... | 31 |
| 5.3 界面设计..... | 32 |
| 5.3.1 主页页面..... | 32 |
| 5.3.2 社区页面..... | 32 |
| 5.3.3 校友状况页面..... | 32 |
| 5.3.4 个人中心页面..... | 32 |
| 5.3.5 公告管理页面..... | 33 |
| 第6章 程序设计..... | 34 |
| 6.1 登陆实现..... | 34 |
| 6.2 主页实现..... | 36 |
| 6.3 发布动态实现..... | 39 |
| 6.4 评论动态实现..... | 41 |
| 6.5 公告管理实现..... | 43 |
| 6.6 个人中心实现..... | 45 |
| 第7章 系统测试..... | 49 |
| 7.1 系统测试的目的和意义..... | 49 |
| 7.2 功能测试..... | 49 |

| | |
|---------------------|----|
| 7.2.1 发布动态测试..... | 49 |
| 7.2.2 评论动态测试..... | 50 |
| 7.2.3 修改个人信息测试..... | 50 |
| 7.2.4 新增公告测试..... | 51 |
| 7.3 可用性测试..... | 52 |
| 7.4 测试结果分析..... | 53 |
| 总结与展望..... | 54 |
| 致谢..... | 55 |
| 参考文献..... | 56 |
| 附录一： | 57 |
| 附录二： | 60 |

第 1 章 绪论

21 世纪是互联网的时代,随着网络越来越普遍地进入了人们的生活,利用网络来交流是人们如今经常使用的方式。本系统将从课题背景,研究意义和国内外现状三方面阐述本校友录系统的优点,让我们对网络交流的实用性和重要性有更深刻的理解。

1.1 课题背景

近年来,各种通信软件极大地便利了人与人之间的联系和交流,使千里之外的人可以随时联系,改变了人与人联系的方式。由于它对互联网的依赖,不受时间和空间的限制,它越来越受到大多数人的欢迎。校友是学校的重要资源,他们作为学校和社会之间的桥梁和纽带,在学校的发展中扮演着重要的角色。在高校自主发展和跨越式发展的过程中,最大限度地开发校友资源,发挥校友在高校发展过程中的重要作用,是高校发展的关键和可行途径之一。从而推动高校不断增强自身实力,提高我国高等教育水平。

随着知识经济和信息经济时代的到来,随着信息技术的飞速发展,管理信息系统在提高管理效率和效率方面的作用越来越明显。通过构建校友管理信息系统平台,可以对校友进行有效的管理,方便校友信息的查询、增加和更新。校友也可以通过开放平台的方式自行修改或完善个人信息。因此,校友管理信息系统是实现校友动态、有效管理的重要途径。

1.2 课题研究意义

如今我们的日常生活中,网络几乎已经成为我们每天都要接触的,计算机也成为方便我们生活和我们了解世界的重要工具,这也成为人与人交流的工具,极大方便了人与人之间的沟通与交流。

交流是人与人之间的桥梁,但要建成这座桥梁又被种种因素所阻挡着。学业繁忙等等因素都会减少同学们之间的生活检验分享。但如今,随着时代的发展,已是信息时代。人们可以动动手指通过网络就可以了解各种新闻,各种琐事。当然也可以动动手指让自己的生活经验通过网络分享给同其他人。该系统正式旨着这一观点而设计的。同学们可以在该系统上注册自己的信息,去发布分享自己的种种事迹,和大家一起讨论生活中的趣事、学习的交流和活动的组织。

同学录,是一种为用户提供网上交流的网络工具,它可以使你你的朋友、同学、老师等在网上有一个互相交流的机会。假如你是在校学生,拥有同学录,你可以通过它了解同学、老师的情况或者给他们留言;同时,你如果毕业了,依然可以通过同学录得到大家的最

近情况,更好地促进同学的感情,同学录它有留言本,可以长期保留你和他人的留言以供你和他查看。它比 E-MAIL 的优势在于它有相册,操作方便,安全,这些是 E-MAIL 所达不到的。与 BBS 相比,其实同学录是另一个意义上的论坛,你同样可以发表自己的看法,不同的是,这是一个表明身份的论坛,不是一个匿名的论坛,它的指向性很明显,因为当你在表达某些看法时,处在集体当中、熟悉你的人可以结合你的背景及对你的了解去理解你的言论,方便实用的留言版让每一个学生都有畅所欲言的权利,在这上面,你可以怀旧,可以展望,总之只要管理员不干预,你可以发表任何正当言论,现在互联网的虚拟世界中成为一个信息传递的平台。

1.3 国内外研究现状

1.3.1 国外研究现状

现在世界上各种形式的同学录网站大概有二十多万个,以目前存在的同学录网站来看,大致有以下几类:(1)按收费方式分,有收费的和免费的,其中以免费的居多,例如友联同学录、网易同学录等大多数提供的是免费校友录服务;(2)按提供同学录服务所在网站分,有院校自己网站上的同学录;有专门单独的同学录网站;有综合网站上的同学录,其中综合网站上的校友录居多;(3)还有网站校友录和手机校友录之分,手机校友录可以说是网站校友录的一种扩展,扩大了其使用范围,增加了便利性。

1.3.1 国内研究现状

目前的校友录正朝着“校友录 2.0”的方向发展,“校友录 2.0”是第二代产品,是在对免费校友录进行了功能升级和全新版面设计、用户可以在保持现有校友录免费服务完全不受影响的情况下,选择这一付费校友录形式的。“校友录 2.0”增加十多项新功能,如“班级留声机”“班级重建计划”等,而“校友录 2.0”的独立服务器和带宽很大提高了访问速度。中国人校友录还增添了信息服务内容,可以进行天气预报、热点新闻的信息定制,并对毕业班同学为他们提供全面就业信息及咨询。可以说“校友录 2.0”的经营已经大大超越了传统的校友录,由校友录衍生的信息服务满足了网民的需要,也让信息的传递更为全面。班级同学录的形式和班级自身的构成是一样的,由班级管理员负责维护界面。方便实用的留言板让每一个班级成员都有畅所欲言的权利;班级相册是同学录最直观、最具吸引力的一个功能,通过简单快捷的方式把照片上传到同学录供朋友们观看,可以让同学不会因为时间的流逝而忘记自己的样子。班级通讯录可以让同学们随时随地都可以联系到想联系的同学,同学录这些功能也使得它与其他网络工具相比有着其独特的优势。网

络发展前景无限，及早与网络结合，与信息时代同步，与高科技汇合，定会给社会各行各业的发展注入新鲜的活力。

1.4 主要内容

主要有用户管理模块，校友公告管理模块，社区管理模块，校友情况发展管理模块。

1.4.1 用户管理模块

①新用户注册，用户登录、注销。②系统用户分为一般用户和系统管理员两种。一般用户：可以修改自己的信息和密码，发布、删除、评论动态等。系统管理员：有一定的权限，可以删除用户，发布、修改、删除公告。社区管理，如删除评论，删除动态等。③信息录入：新用户要填写必要的信息，系统每年定期提醒用户更新自己的信息。

1.4.2 校友公告管理模块

此模块主要设计给系统管理员使用，管理员通过此平台发布校级新闻和活动通知。通过此平台，校友可以及时了解学校的信息，对提高校友对学校的归属感有很好的作用。

1.4.3 社区管理模块

社区是一个提供给校友进行交流的地方，大家可以分享自己的生活动态，其他用户可以在此动态下面评论。

1.4.4 校友情况发展管理模块

用户可根据自己的要求去了解校友的发展情况，如：在搜索栏里输入年份，就可以查到对应级的毕业生现在的发展情况。输入研究生，就可以看到有多少毕业生，正在读研、已获得硕士学位或博士学位。输入五百强，就可以了解到有多少校友目前在世界五百强企业工作等。

1.5 论文组织

本论文从第1章到第7章主要组成包括

第1章为绪论：描述校友录系统的背景、目的及研究现状；

第2章系统相关技术：对设计系统使用到的相关技术的进行介绍；

第3章需求分析：描述校友录系统的需求；

第4章概要设计：描述用户的数据设计、体系结构和模块设计；

第5章详细设计：针对每个模块的功能进行具体细节的描述，要把对功能的描述转化为精确的、结构化的描述过程；

第6章程序设计：针对校友录系统的功能进行程序设计实现，包括代码及界面实现；

第 7 章系统测试：软件测试包括对软件系统的功能分别进行测试分析运行结果；最后结语总结是对本次毕设的感想和领悟。

第 2 章 系统相关技术

2.1 前端技术

2.1.1 Vue CLI

Vue CLI 俗称 Vue 脚手架，它可以帮助我们快速的构建 Vue 框架。他可以用可视化界面显示的方式来引导我们如何创建一个 Vue 框架。如果需要引入相关的依赖，也可以在可视化界面中查找想要的依赖并添加。Vue CLI 最大的好处就是将 Vue 基础标准化，极大的节省了创建项目的时间，让我们有更多的时间去集中代码的书写。无论是新手老手都可以很方便的利用可视化界面来完成 Vue 的创建。

2.1.2 Vue

Vue 是一套构建用户界面的渐进式框架。它与其他框架不同的是，Vue 着重单页面开发，它的代码编写只着重于视图层，如果想在代码中加入其他组件库，可以很好的集成进来而不影响功能。

2.1.3 Webpack

前端的开发不外乎遇到很多的资源文件，如一些静态资源 CSS、JS 和图片等。这些资源不像动态资源，他需要页面打开时一一加载，这就导致的了显示页面的缓慢。Webpack 就很好的解决了这个问题。它是一个资源打包工具，他可以将这些静态资源打成一个模块，并且很好的处理这些模块的关系，极大的帮助了我们对静态资源的管理。

2.1.4 Axios

Axios 是一个 Http 库，它可以帮助我们完成前后端 Ajax 交互并且能自动的将数据转换成 JSON 格式。

2.1.5 Babel

Babel 是一个 JavaScript 编译器，它对外宣传的是下一代的 JavaScript 语法。他可以帮助我们兼容不同版本的 js 语法。

2.1.6 Element

Element 是一个基于 Vue 的组件库。它利用蓝白界面，简洁的展示了一些界面。利于 UI 设计师设计页面。

2.2 后端技术

2.2.1 SSM 框架

SSM 是 Spring + Spring MVC + MyBatis 的简称。

(1) Spring

Spring 是一个开源框架，它是为了解决企业级系统开发的复杂性而设计的。框架的最受认同的是 IoC 和 DI，容器将创建对象，而不用人工去 new 对象。这样就可以使 Spring 容器去控制对象，而不是人工去控制它。它也为 J2EE 的体系提供集成。然而，Spring 的用途不仅仅于这些。从整体来说，任何 Java 应用都可从中受益。

(2) Spring MVC

Spring MVC 沿用了 Web MVC 的设计思想再加以改进。它对 Web 进了解耦，使每一层的模块体现了与它最相关的内容。M 指的是模型层，V 指的是视图层，C 指的是控制层。

(3) MyBatis

MyBatis 是一个轻量的持久层框架。它使用 XML 来配置，也可以用注解的形式简化了配置。它使用 POJO 和接口等来与数据库中的表字段来映射。

2.2.2 Spring boot

SpringBoot 是为了简化创建 Spring 项目而开发的一种框架。它集成了 Spring 的优秀功能。该框架基于 Starter 方式来配置项目，从而使开发人员不再为繁琐的配置而苦恼。

2.2.3 Druid

Druid 是一个阿里巴巴公司开发的数据库连接池。它在功能、扩展性等方面都优于其他连接池，是目前现阶段最好的连接池。并且可以监控数据库的状态。

2.2.4 Maven

Maven 主要用于项目的构建和依赖管理，它有一个 pom.xml 文件，用来管理关于项目的所有 jar 包。使用代码就能简化导包的过程，极大的省去了导包的时间。

2.3 软件安装

2.3.1 JDK 安装

1、首先去官网下载 JDK1.8 的安装包，如图 2-1 所示。



图 2-1 JDK 安装步骤图

2、点击下载，会出现下如图 2-2 界面，首先勾选协议，再点击对应的电脑位数。

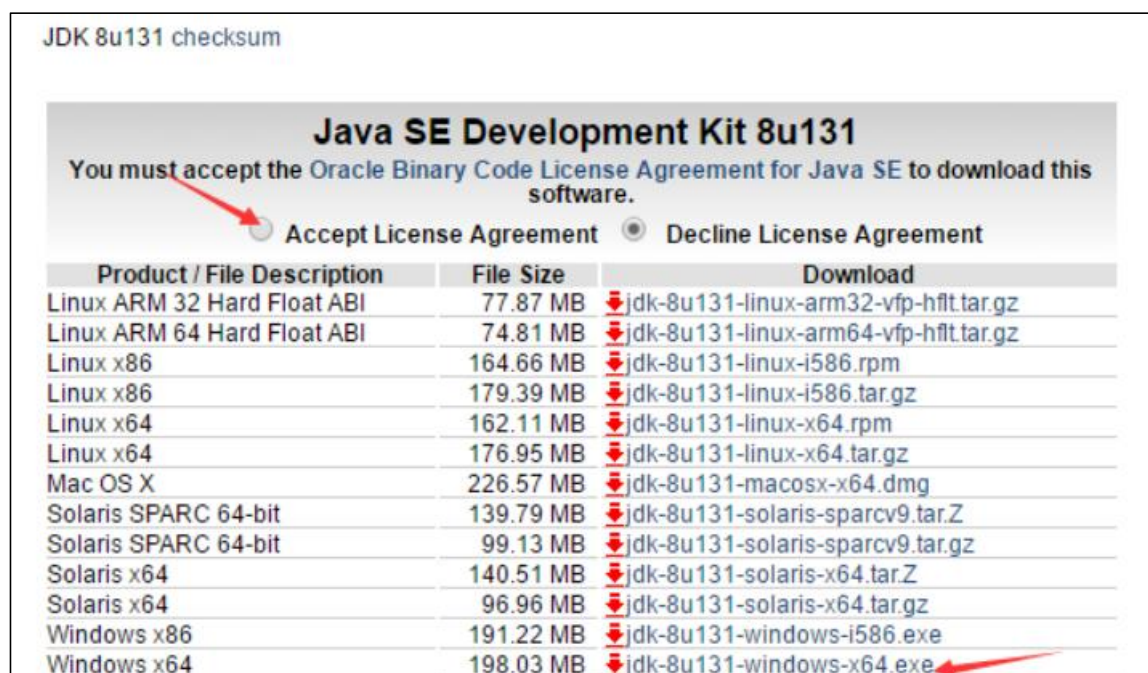


图 2-2 JDK 安装步骤图

3、双击安装包，点击下一步，选择安装路径，具体界面如下图 2-3 所示。



图 2-3 JDK 安装步骤图

4、之后点击下一步，等待安装，安装完出现如下图 2-4 界面，即安装成功。



图 2-4 JDK 安装步骤图

5、安装完成后，配置环境变量，右键我的电脑—属性----高级系统设置，就会看到如下图 2-5 界面。



图 2-5 JDK 安装步骤图

6、点击环境变量，新建环境变量，输入如下图 2-6 所示。

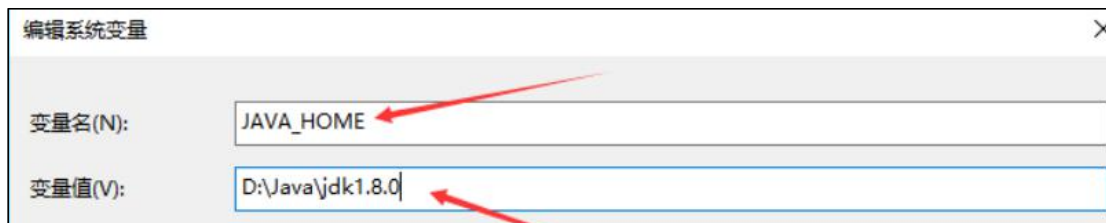


图 2-6 JDK 安装步骤图

7、继续新建一个变量，其变量值如下图 2-7 所示：

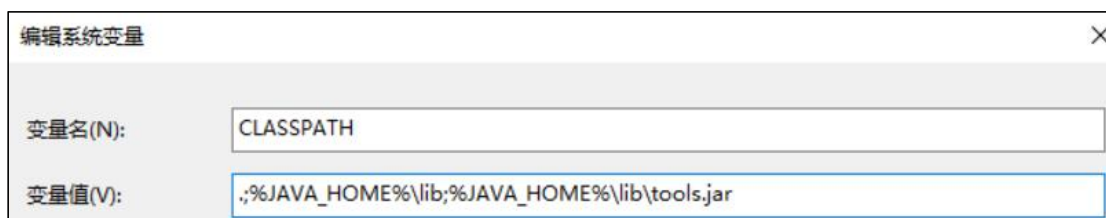


图 2-7 JDK 安装步骤图

8、测试 jdk 是否安装成功，WINDOWS+R 键，输入 cmd，进入命令行界面，输入 java -version 命令，可以出现如下图 2-8 的提示，你可以看你安装的 JDK 版本。


```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.18362.778]
(c) 2019 Microsoft Corporation. 保留所有权利。

C:\Users\feifei>java -version
java version "1.8.0_231"
Java(TM) SE Runtime Environment (build 1.8.0_231-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.231-b11, mixed mode)
```

图 2-8 JDK 安装步骤图

2.1.2 IDEA 安装

- 1、首先去官网下载安装包。Community 即为社区版，免费试用，功能相对而言不是很丰富，但是不影响开发使用。如果非要比较的话类似于 eclipse。
- 2、下载好了安装包，确认已经安装好了 JDK ， 双击开始安装。点开后的界面具体如下图 2-9 所示。

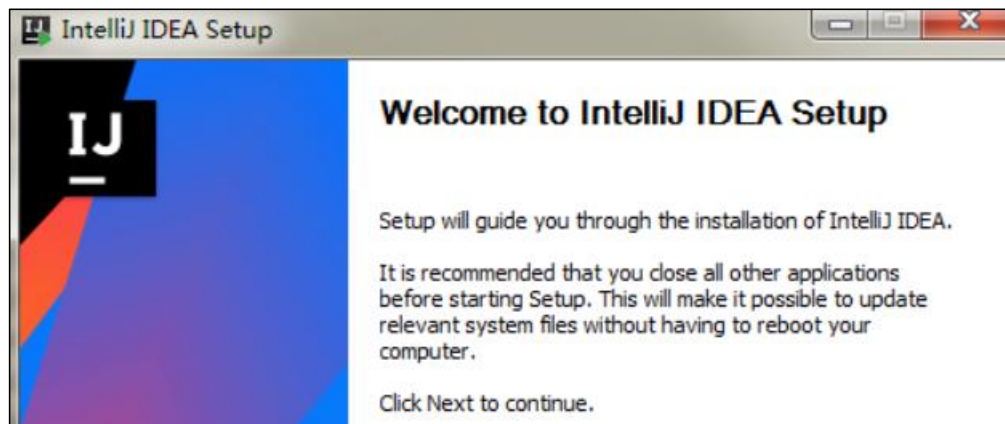


图 2-9 IDEA 安装步骤图

- 1、点击下一步，选择安装路径，具体界面如下图 2-10 所示。

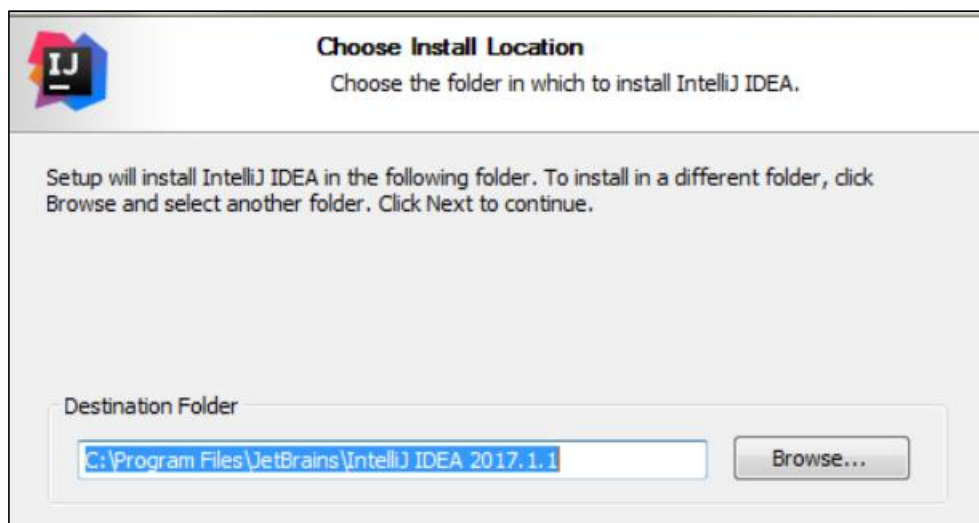


图 2-10 IDEA 安装步骤图

2、选择 32 位或 64 位的桌面快捷方式。选择关联相应的文件。具体界面如下图 2-11 所示。

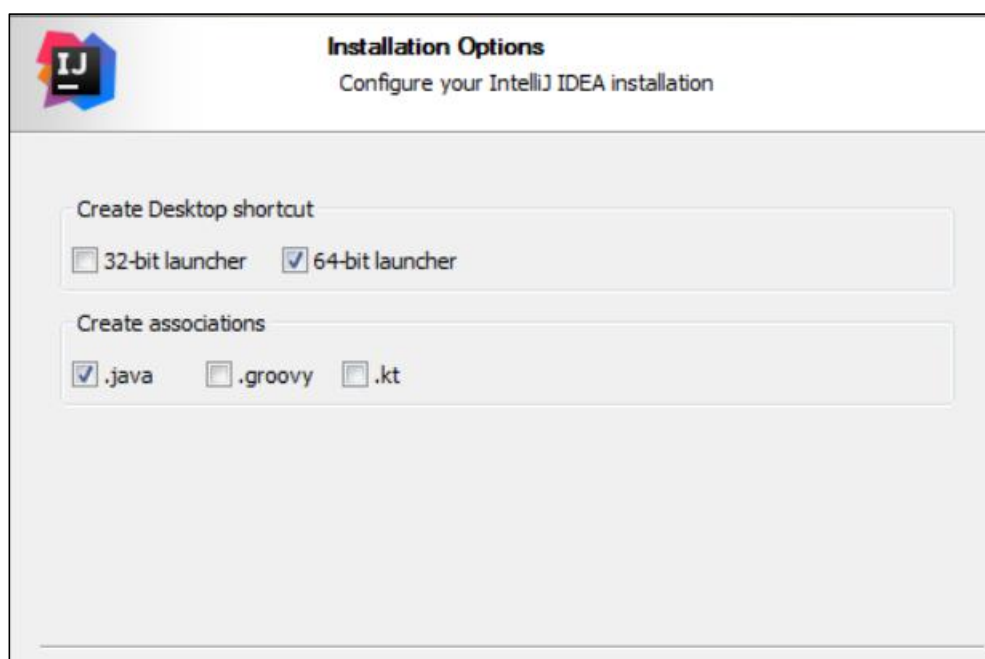


图 2-11 IDEA 安装步骤图

3、一路 next，等待安装，出现如下图 2-12 界面，即安装成功。

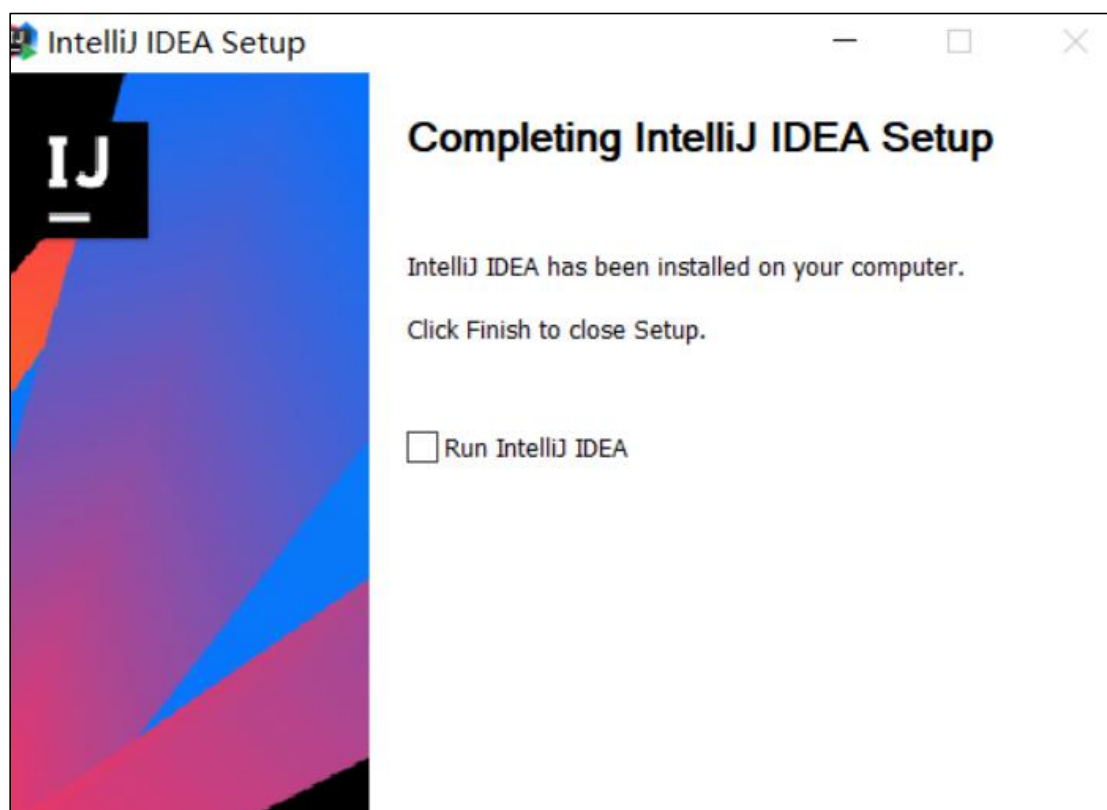


图 2-12 IDEA 安装步骤图

2.1.3 WebStorm 安装

1、先下载最新的安装包，双击打开。出现如下图 2-13 界面。

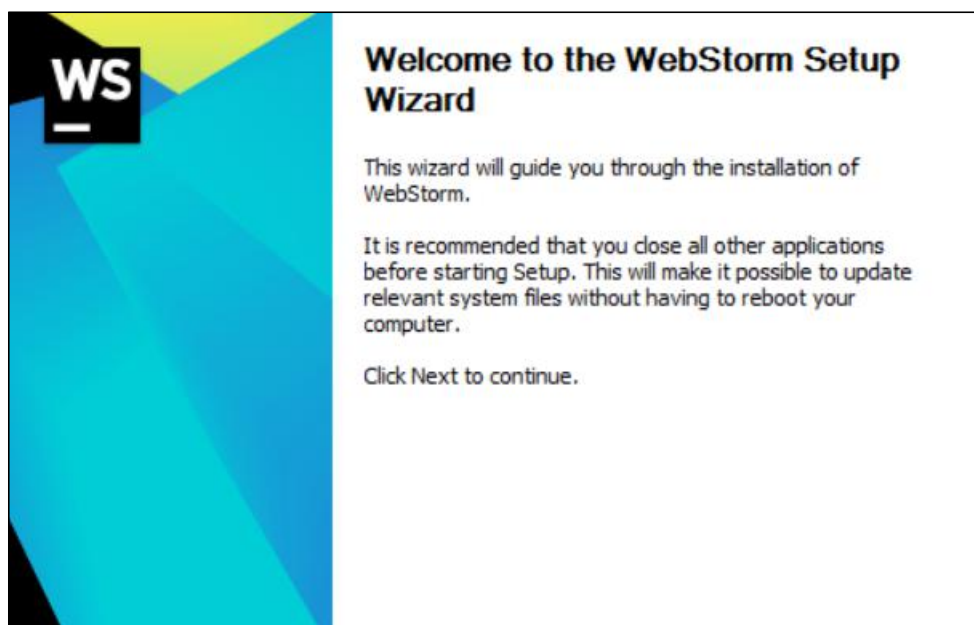


图 2-13 WebStorm 安装步骤图

2、点击 next，选择安装路径，具体界面如图 2-14 所示。

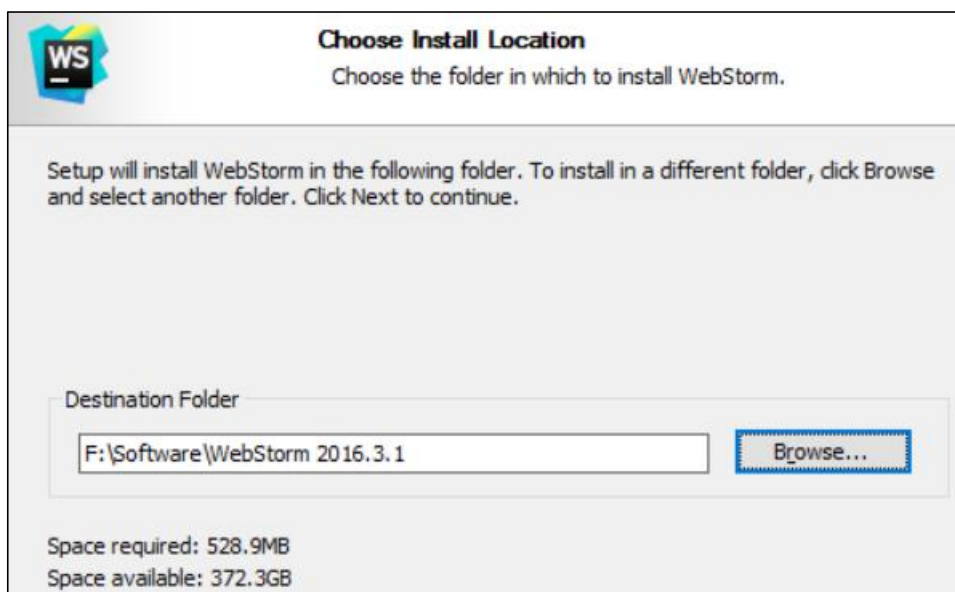


图 2-14 WebStorm 安装步骤图

3、选择 32 位或 64 位的桌面快捷方式。选择关联相应的文件，可以不勾选。具体界面如下图 2-15 所示。

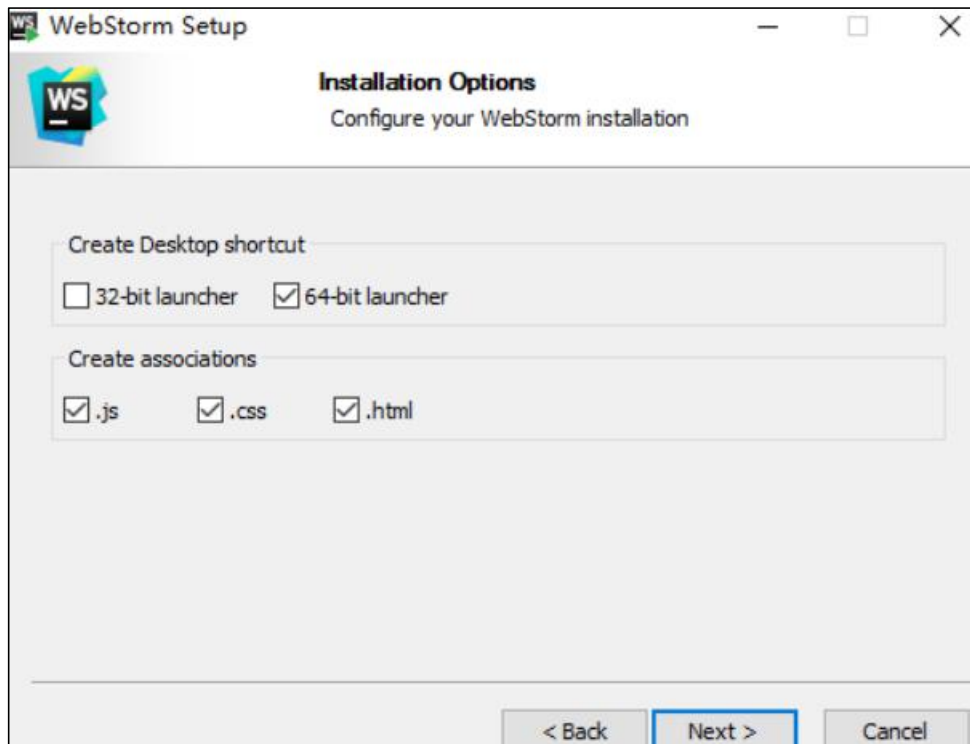


图 2-15 WebStorm 安装步骤图

4、一路 next，等待安装，出现如下图 2-16 界面，即安装成功。

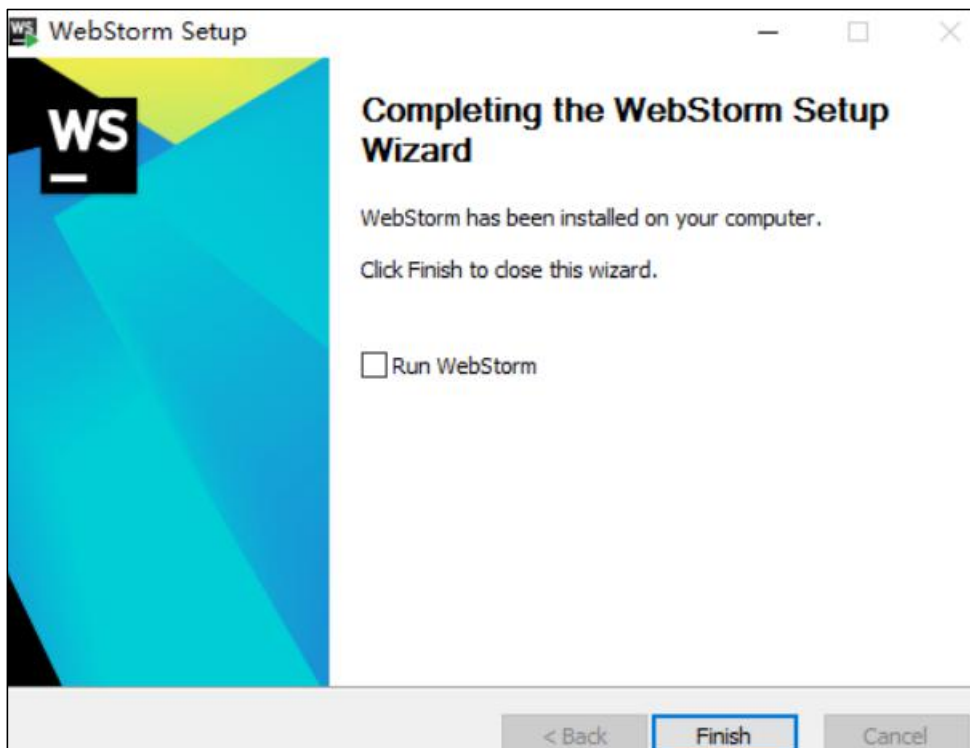


图 2-16 WebStorm 安装步骤图

2.1.4 MySQL 安装

1、从官网上下载 mysql-5.6.31-winx64.zip,将其解压到指定文件夹。打开终端进入此文件夹。

2、输入命令 `mysqld install`。具体界面如下图 2-17 所示。

```
C:\Users\feifei>mysqld install
```

图 2-17 MySQL 安装步骤图

3、输入命令 `net start mysql` 启动 MySQL 服务。具体界面如图 2-18 所示。

```
C:\Users\feifei>net start mysql
```

图 2-18 MySQL 安装步骤图

4、配置环境变量。具体界面如下图 2-19 所示。

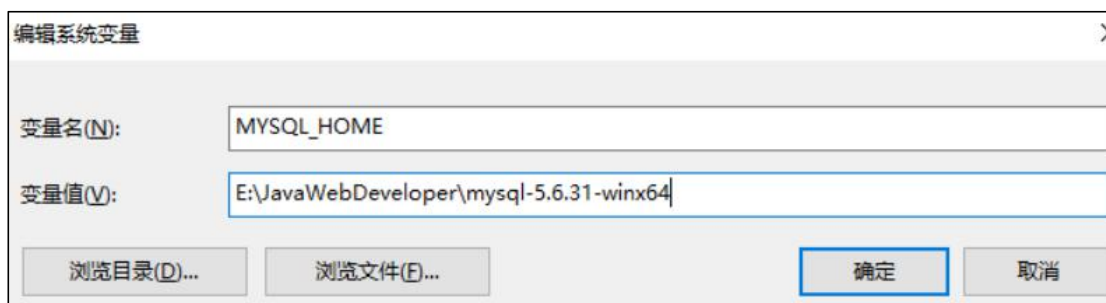


图 2-19 MySQL 安装步骤图

5、输入 `mysql -u root -p`，密码默认为空。进入 `mysql`。具体界面如下图 2-20 所示。

```
C:\Users\feifei>mysql -u root -p
```

图 2-20 MySQL 安装步骤图

2.1.5 Node.js 安装

1、去官网下载最新安装包，双击打开安装，出现如下如 2-21 界面。

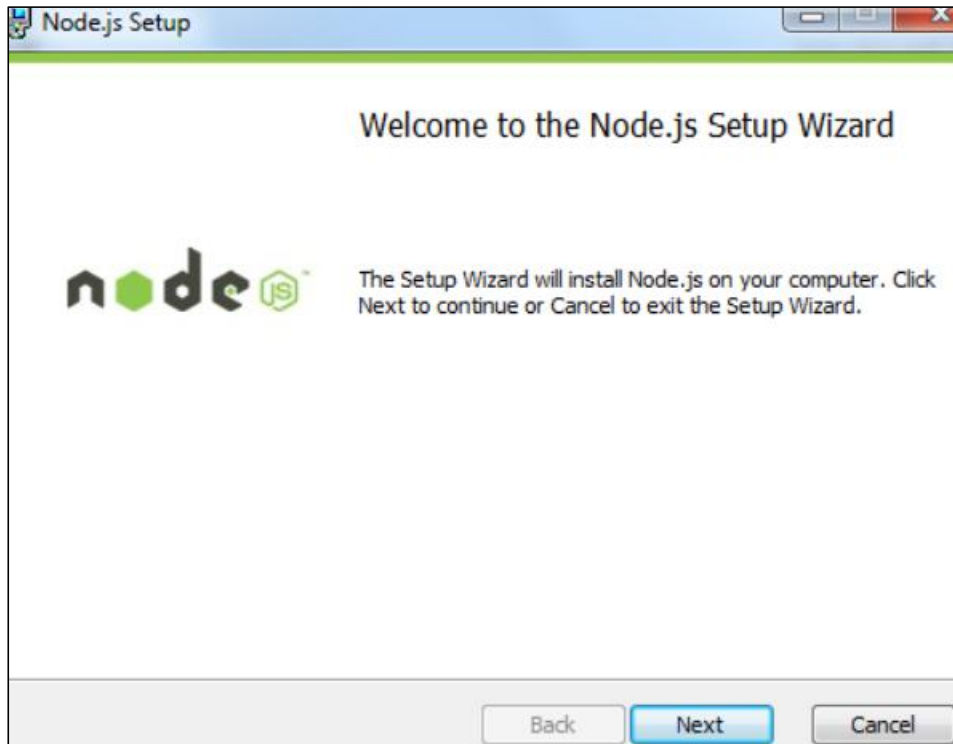


图 2-21 Node.js 安装步骤图

2、点击 next，勾选接受协议，点击 next，选择安装路径，具体界面如下图 2-22 所示。

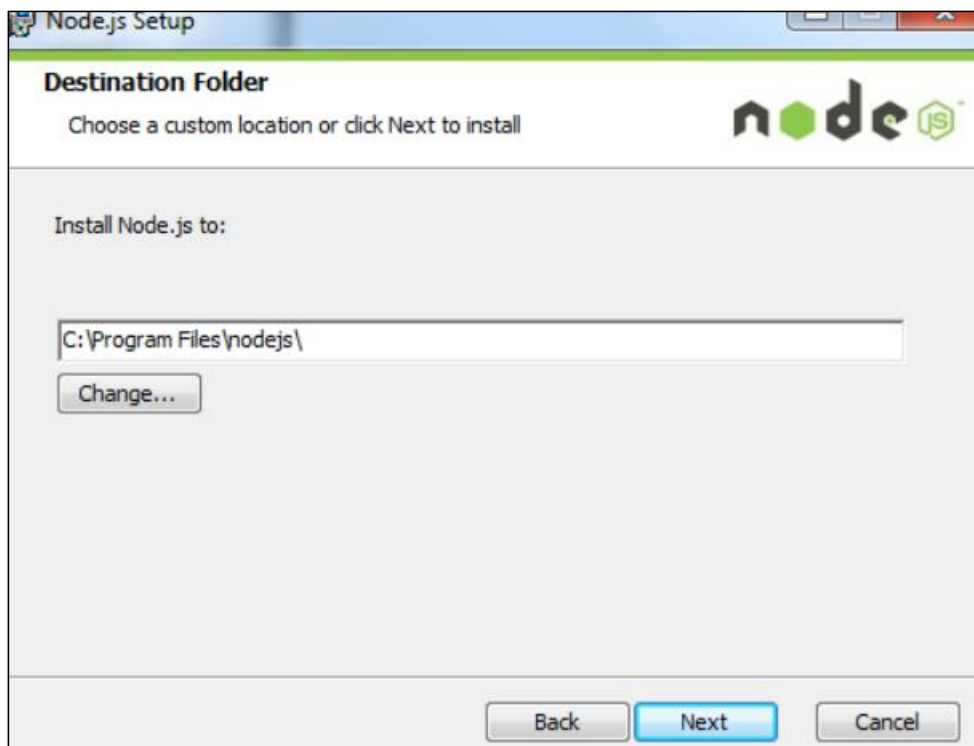


图 2-22 Node.js 安装步骤图

3、点击 next，选择要安装的模式，点击 next，具体界面如下图 2-23 所示。

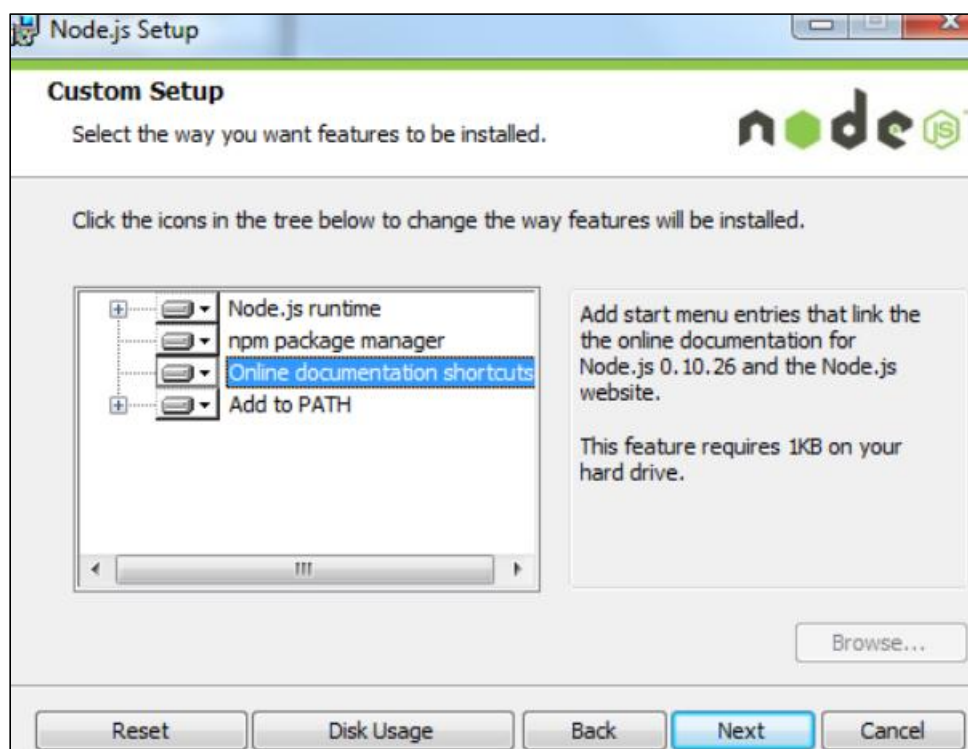


图 2-23 Node.js 安装步骤图

4、点击安装，等待完成，出现如下图 2-24 界面。

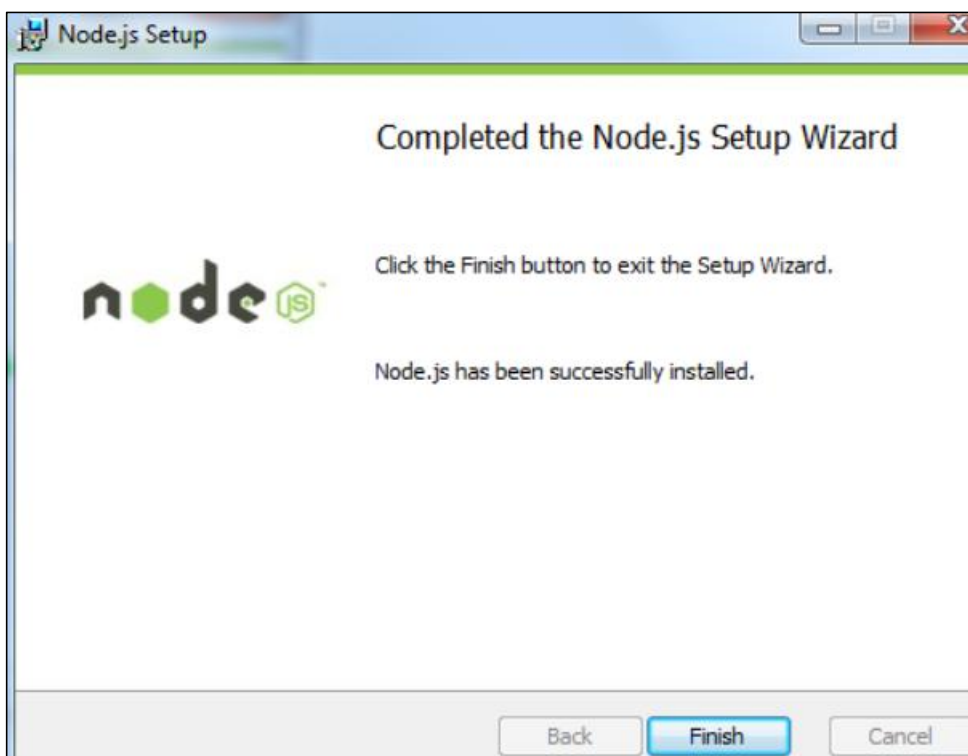


图 2-24 Node.js 安装步骤图

5、进入终端，输入 `node --version`，出现如下图 2-25 界面即安装成功。


```
C:\Users\feifei>node --version  
v8.9.4  
  
C:\Users\feifei>
```

图 2-25 Node.js 安装步骤图

2.2 环境搭建

2.2.1 前端环境搭建

- 1、打开终端，输入命令 `npm install -g @vue/cli`，下载 vue 脚手架。
- 2、下载完成后输入命令 `vue ui` 会自动弹出创建 vue 项目的可视化界面。
- 3、选择创建，选择合适的路径位置，点击在此创建新项目，具体界面如下图 2-26 所示。



图 2-26 前端环境搭建步骤图

- 4、输入项目名，点击下一步，具体界面如下图 2-27 所示。



图 2-27 前端环境搭建步骤图

5、选择默认，点击创建项目，等待自动创建完成即可。

2.2.2 后端环境搭建

1、打开 IDEA，点击 file->new->project，具体界面如下图 2-28 所示。

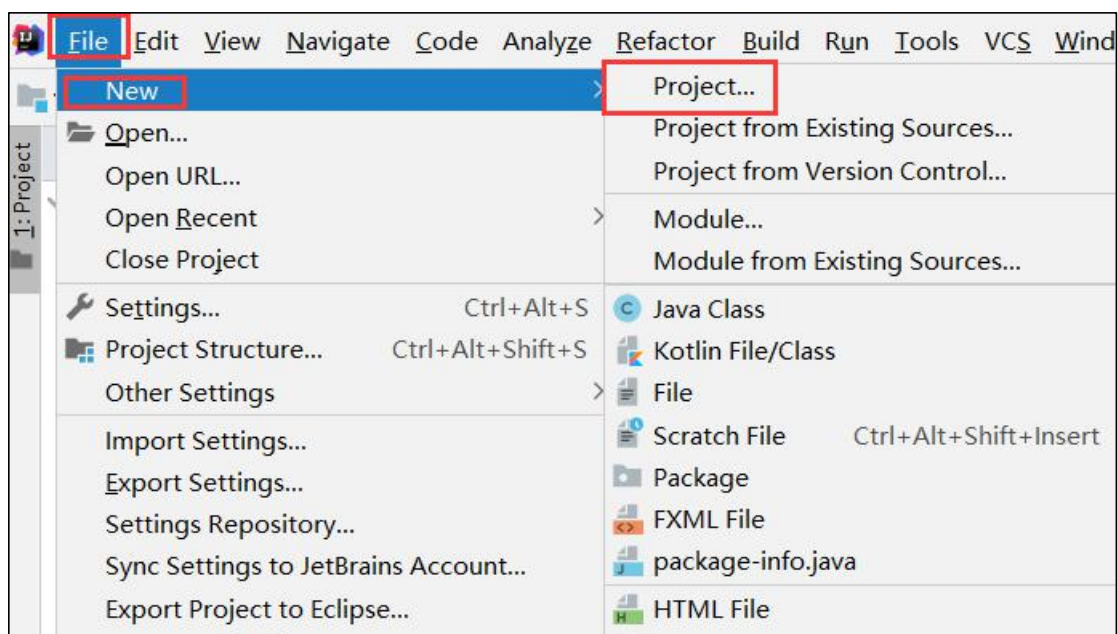


图 2-28 后端环境搭建步骤图

2、点击 spring initializr，选择 sdk，点击下一步，具体界面如下图 2-29 所示。

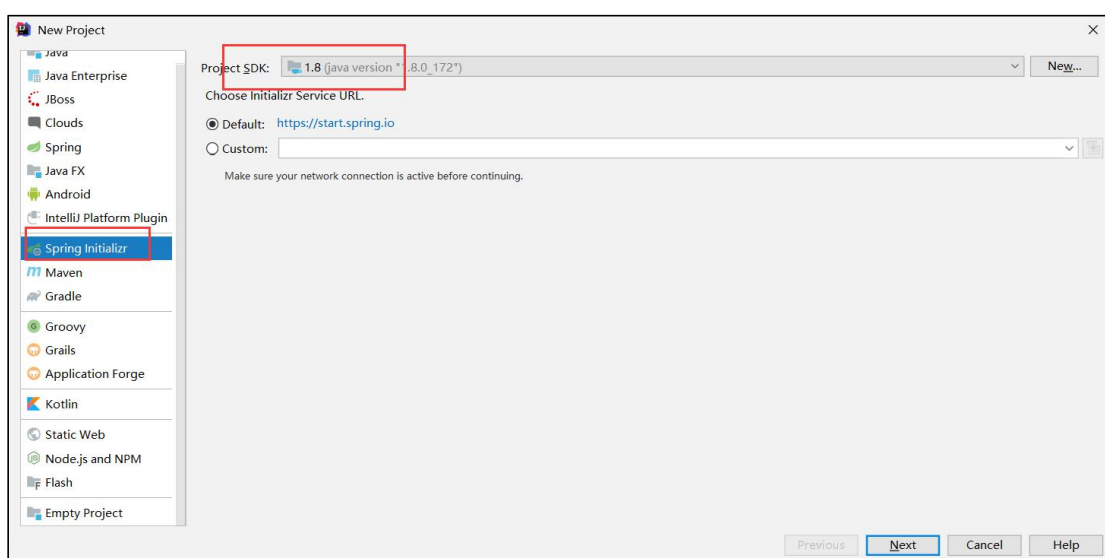


图 2-29 后端环境搭建步骤图

3、输入项目信息，点击下一步。具体界面如下图 2-30 所示。

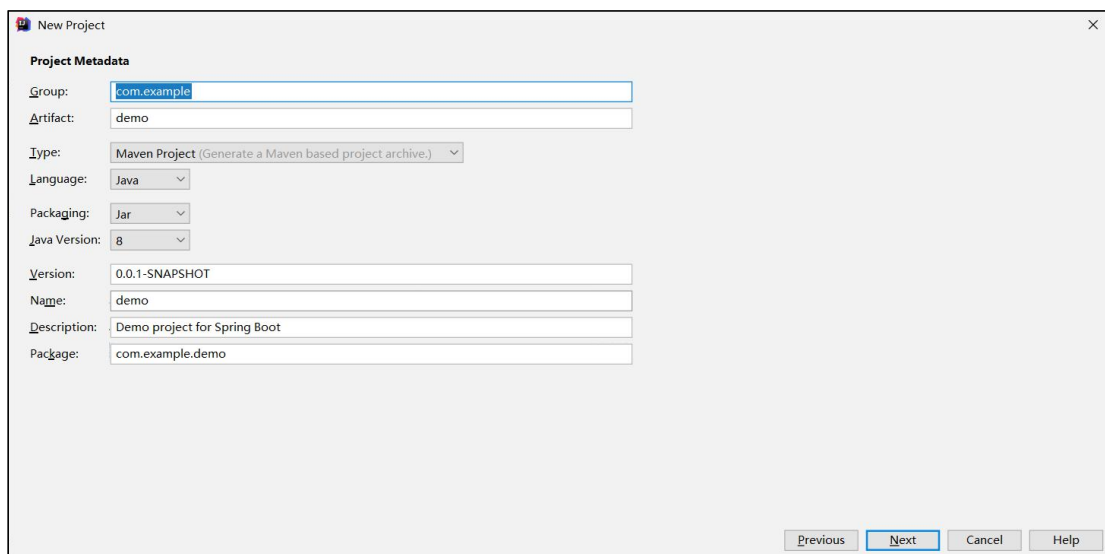


图 2-30 后端环境搭建步骤图

4、选择 web->spring web，点击下一步。具体界面如下图 2-31 所示。

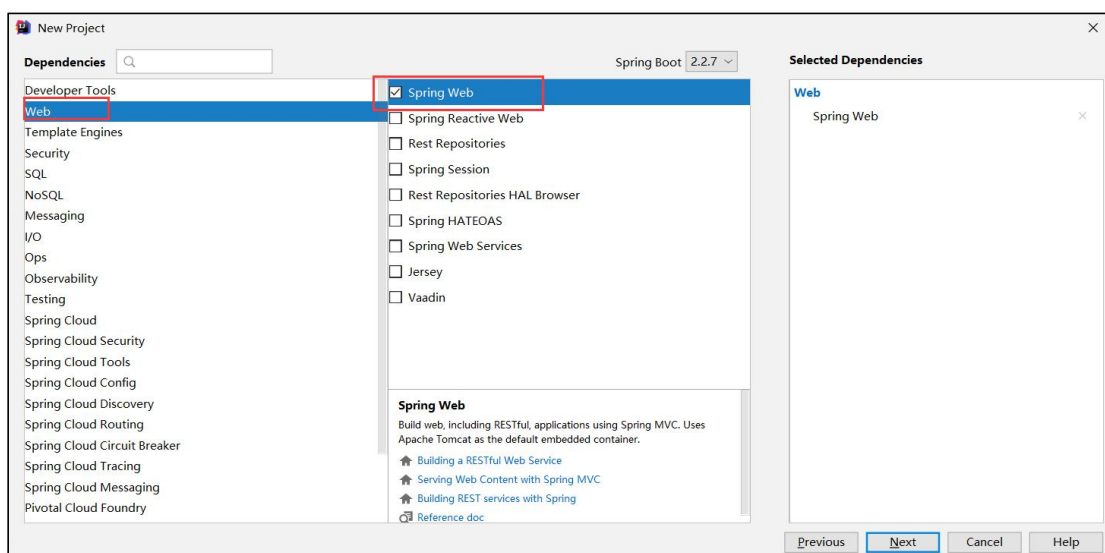


图 2-31 后端环境搭建步骤图

5、最后输入项目名，选择创建路径，点击 finished，等待片刻即创建完成。具体界面如下图 2-32 所示。



图 2-32 后端环境搭建步骤图

第3章 需求分析

本章节主要是对整个系统的功能进行需求分析，分析关于校友录系统的各个功能的实现。分别从社区页面、校友状况页面、个人中心页面、用户管理页面、公告管理页面对系统进行相应的分析。结合页面，给出详细的说明。

3.1 总体目标

页面以简洁的蓝白颜色展示了整个系统。用户通过简单的操作就可以在系统中发布管理评论动态，查看新闻活动等。本系统主要针对校园内的学生服务，通过网络，可以简单便捷的分享自己的生活。

管理员功能：

- 1、公告管理：管理员可以任意的管理校园的新闻和活动公告。包括对公告的增加、删除、修改。
- 2、用户管理：管理员可以对注册的用户进行的统一管理。可以删除一些违反规则的用户。
- 3、管理员拥有其他普通注册用户的权限

用户功能：

- 1、查看公告功能：用户登陆会跳转到首页，首页会展示近期管理员发布的活动或新闻。
- 2、社区功能：用户可以在社区模块发布自己的动态、评论别的动态以及管理自己的动态。
- 3、校友状况功能：用户可以根据条件查看系统中校友的信息情况。
- 4、个人中心功能：用户可以修改自己的基础信息以及密码。

3.2 可行性分析

3.2.1 技术可行性

关于操作平台方面，本系统采用 win10，系统稳定可靠。电脑只需要很低的配置就可以流畅的运行。

关于开发环境方面，本系统用的是 jdk8 进行的开发，后端 IDE 用的是当今比较流行的 idea，前端 IDE 用的是 webstorm。

关于开发技术方面，后端用的是比较火的 springboot 继承框架，它是简化创建 Spring 项目而开发的一种框架。它集成了 Spring 的优秀功能。该框架基于 Starter 方式来配置项目，从而使开发人员不再为繁琐的配置而苦恼。前端用的比较流行的三大 js 框架之一的 vue，它是一套构建用户界面的渐进式框架。它与其他框架不同的是，Vue 着重单页面开发，它的代码编写只着重于视图层，如果想在代码中加入其他组件库，可以很好的集成进来而不影响功能。

3.2.2 操作可行性

页面简洁的展示了整个系统，对于每个操作，都会有小提示来向用户进行说明，不会出现用户对按钮或者页面不理解的地方。这很大的提升了用户的操作感。

3.2.3 运行可行性

本系统的每个功能都比较完善健全，各个功能模块相对独立，所以不会出现 bug、延迟等现象。

3.3 功能性需求

1、社区功能：用户点击社区菜单，即可进入社区页面。在社区页面，右上角有两个按钮，第一个是发布动态按钮，第二个是进入我的动态按钮。点击发布动态按钮，会在右侧弹出抽屉框，用户可以在此框中输入想要发布的内容，点击发布即可。发布完后，页面会即可刷新最近的动态。点击第二个进入我的动态按钮，即可进入我的动态页面。在此页面上，用户可以查看自己发布过的历史动态，并可以对动态进行删除管理。在社区页面上，每个动态下方都会有个评论区，用户可以在评论区内输入想要评论的内容，点击发布，即可在此动态上评论。

2、校友状况功能：用户点击校友状况菜单，即可进入校友状况页面。在此页面上，用户可以根据自己想要的条件去查询想要了解的校友情况。例如姓名、学历、工作、入学时间等等。

3、个人中心功能：用户点击个人中心菜单，即可进入个人中心页面。在此页面上，分为两个 tab 页，第一个是基础信息修改，第二个是密码修改。点击个人信息，在此页面上，可以修改关于自己的一些基础信息。点击修改密码，可以修改自己的密码。

4、公告管理功能：管理员点击公告管理菜单，即可进入公告管理页面。在此页面上，管理员可以增删改查一些公告信息。以便发布最新的公告告诉此系统中的用户。

3.4 非功能性需求

对于一个系统的好坏，也并不是针对于系统功能设计的实现怎么样。对于一些非功能性的要求，例如稳定事实也是有必要的。

该校友录系统整体功能基本完善。整体界面使用的是蓝白颜色，简洁大方，格局框架分割的很整齐，不会使用户看了感觉别扭。对于一些提示也很人性化的体现出来。每个功能都用分级菜单很好的展示出来。用户可以简单的发布动态。表格清晰明了的显示了每个数据的信息。对于每个按钮都有提示信息，可以提示用户该按钮的意思。对于操作的成功失败都用绿色和红色体现出来，便于用户分辨。

第 4 章 概要设计

4.1 数据库设计

本系统有四大实体，分别是用户、公告、评论和动态。用户实体包括用户名、密码、手机号、邮箱、用户类型和入学时间等属性。公告实体包括公告名和公告内容等属性。评论实体包括评论内容、评论人、动态 id 和创建时间等属性。动态实体包括教动态内容、动态创始人和创建时间等属性。

其中用户与动态的联系为 1:N。用户和评论的联系为 1:N。动态和评论的联系为 1:N。

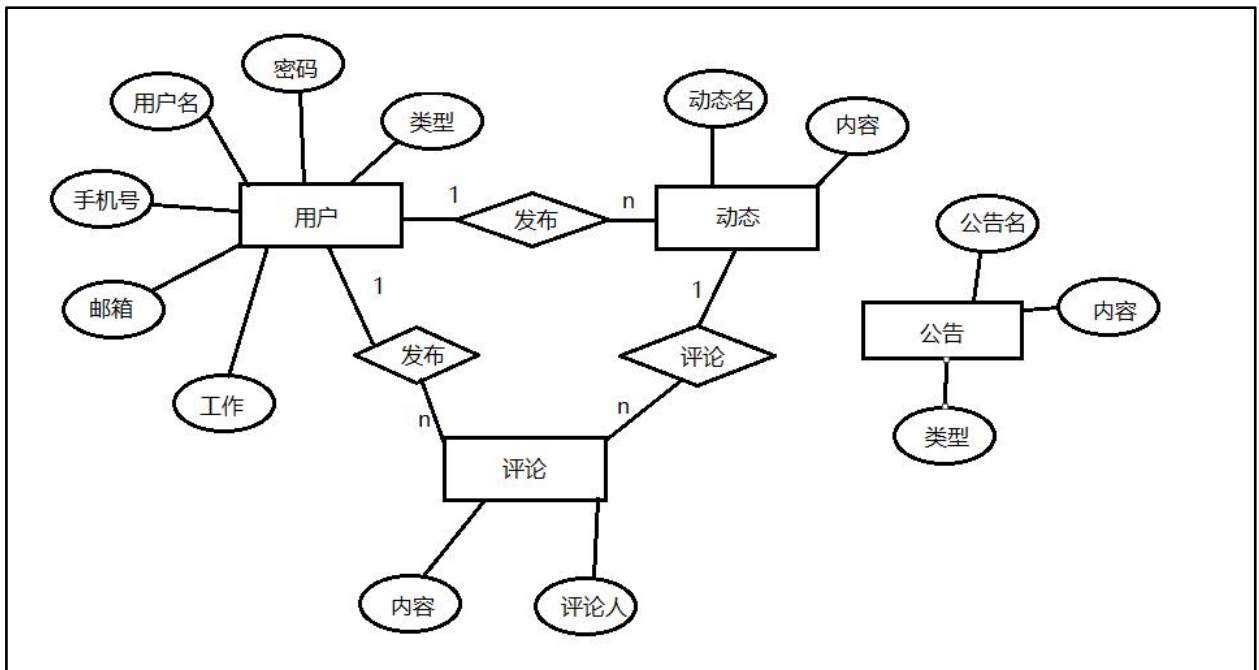


图 4-1 数据分析 E-R 图

4.2 系统总体设计

该系统的用户权限分为 2 种，只有一个超级管理员，其他为普通用户。超级管理员拥有所有用户权限的功能，并且可以管理用户信息、发布公告。用户可以查看公告、发布动态、评论动态和修改自己信息等。

详细系统功能结构可如图 3-2 所示：

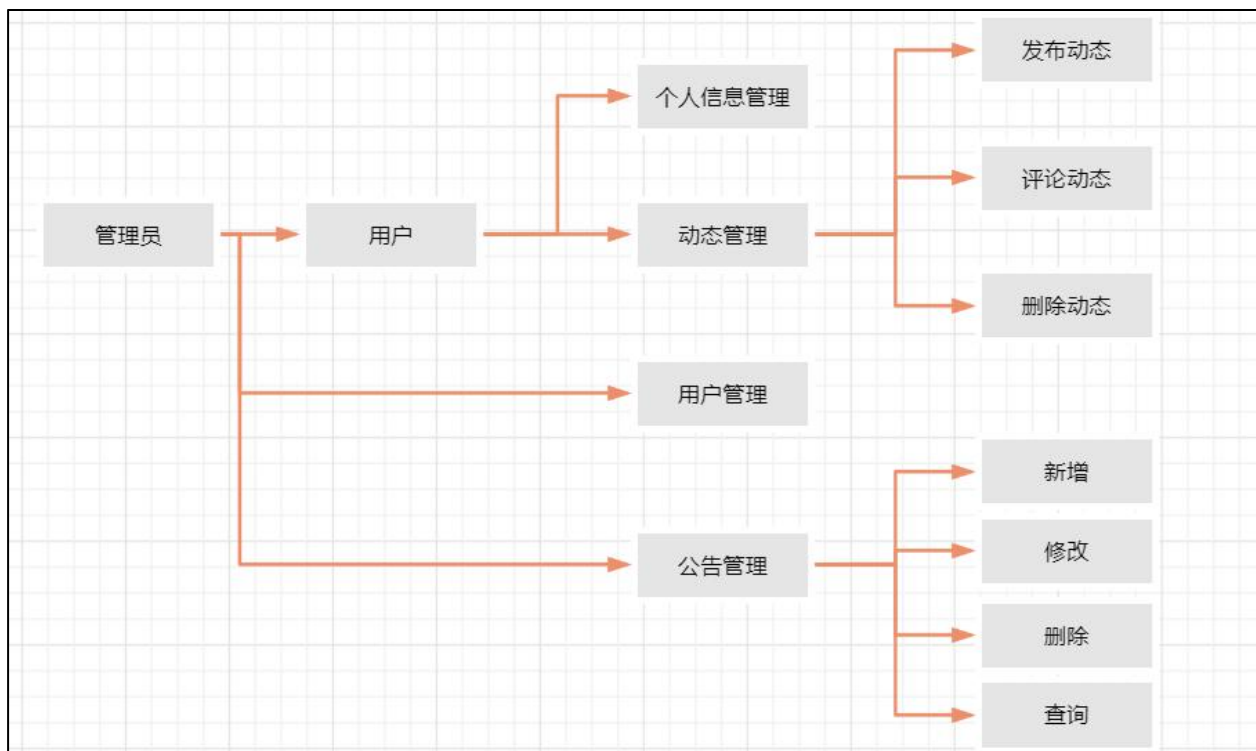


图 4-2 系统功能结构图

4.3 环境设计

- 推荐 Chrome 以及使用 Chrome 内核的较新版本浏览器访问前端可视化操作界面。
- 推荐 Tomcat 8.0+
- 推荐 JDK 1.8+
- 推荐 MySQL 5.6+
- 推荐 Nginx 1.6.2+

第 5 章 详细设计

5.1 数据库设计

5.1.1 用户表 (user)

用户表主要是存储用户信息，字段有主键 id、用户名、密码、用户类型、创建时间、最近修改时间、入学时间、最高学历、工作、电话号码和邮箱。具体设定如表 5-1 所示：

表 5-1 用户表

| | | | | | | |
|----|----------------|----------|-----|------|------|-----------------------------------|
| 表名 | user | | | | | |
| 说明 | 用户表 | | | | | |
| 主键 | id | | | | | |
| 索引 | | | | | | |
| No | 字段名 | 数据类型 | 长度 | 允许为空 | 默认值 | 约束条件和说明 |
| 1 | id | int | 11 | N | | 主键 id, 自增 |
| 2 | name | varchar | 256 | Y | Null | 用户名 |
| 3 | password | varchar | 256 | Y | Null | 密码 |
| 4 | type | int | 2 | Y | Null | 用户类型 1: 普通用户 2: 管理员 |
| 5 | create_time | datetime | | Y | Null | 创建时间 |
| 6 | update_time | datetime | | Y | Null | 最近修改时间 |
| 7 | admission_time | datetime | | Y | Null | 入学时间 |
| 8 | highest_degree | varchar | 2 | Y | Null | 最高学历。1-本科，2-读研中，3-研究生，4-读博中，5-博士生 |
| 9 | phone | varchar | 256 | Y | Null | 电话号码 |
| 10 | mail | varchar | 256 | Y | Null | 邮箱 |
| 11 | work | varchar | 2 | Y | Null | 工作。1-普通企业，2-世界 500 强 |

5.1.2 动态表 (talk)

动态表主要是存储关于动态的信息，字段有主键 id、动态内容、动态创始人和创建时间。具体设定如表 5-2 所示：

表 5-2 动态表

| | | | | | | |
|----|------|--|--|--|--|--|
| 表名 | talk | | | | | |
| 说明 | 动态表 | | | | | |
| 主键 | id | | | | | |
| 索引 | | | | | | |

| No | 字段名 | 数据类型 | 长度 | 允许为空 | 默认值 | 约束条件和说明 |
|----|-------------|----------|-----|------|------|-----------|
| 1 | id | int | 11 | N | | 主键 id, 自增 |
| 2 | content | varchar | 512 | Y | Null | 动态内容 |
| 3 | create_name | varchar | 256 | Y | Null | 动态创始人 |
| 4 | create_time | datetime | | Y | Null | 创建时间 |

5.1.3 评论表 (comment)

评论表主要是储存评论信息, 字段有主键 id、评论内容、评论人、创建时间和动态 id。具体设定如表 5-3 所示:

表 5-3 评论表

| 表名 | comment | | | | | |
|----|--------------|----------|-----|------|------|-----------|
| 说明 | 评论 | | | | | |
| 主键 | id | | | | | |
| 索引 | | | | | | |
| No | 字段名 | 数据类型 | 长度 | 允许为空 | 默认值 | 约束条件和说明 |
| 1 | id | int | 11 | N | | 主键 id, 自增 |
| 2 | content | varchar | 512 | Y | Null | 评论内容 |
| 3 | comment_name | varchar | 256 | Y | Null | 评论人 |
| 4 | create_time | datetime | | Y | Null | 创建时间 |
| 5 | talk_id | int | 11 | Y | Null | 动态 id |

5.1.4 公告表 (announcement)

公告表主要是储存公告信息, 字段有主键 id、公告名、公告内容、创建人姓名、创建人 id、公告类型、创建时间和更新时间。具体设定如表 5-4 所示:

表 5-4 公告表

| 表名 | announcement | | | | | |
|----|--------------|----------|------|------|------|---------------------|
| 说明 | 公告表 | | | | | |
| 主键 | id | | | | | |
| 索引 | | | | | | |
| No | 字段名 | 数据类型 | 长度 | 允许为空 | 默认值 | 约束条件和说明 |
| 1 | id | int | 11 | N | | 主键 id, 自增 |
| 2 | name | varchar | 512 | Y | Null | 公告名 |
| 3 | content | varchar | 1024 | Y | Null | 公告内容 |
| 4 | create_name | varchar | 256 | Y | Null | 创建人姓名 |
| 5 | create_id | varchar | 256 | Y | Null | 创建人 id |
| 6 | type | varchar | 2 | Y | Null | 公告类型。1-校级新闻, 2-活动通知 |
| 7 | create_time | datetime | | Y | Null | 创建时间 |
| 8 | update_time | datetime | | Y | Null | 更新时间 |

5. 2 流程设计

5. 2. 1 登陆功能流程设计

用户通过登陆窗口，输入用户名和密码。前端将数据通过接口发送给服务器，服务器先会从数据库中查询是否存在该用户，如果不存在，直接将错误结果返回给前端。如果存在，再去判断该用户的 session 状态，如果 session 超时或失效，会创建新的 session。如果存在，返回正确结果给前端，前端收到结果进行处理后跳转到相应的界面。

5. 2. 2 注册功能流程设计

用户通过注册窗口，输入用户名、密码和确认密码。前端将数据发给后端，后端通过一些逻辑判断去数据库存取信息，再返回结果给前端。提示成功或失败。

5. 2. 3 退出功能流程设计

用户从界面点击退出功能，会触发相应的接口传给服务器。服务器接到请求后，会清除该用户相应的 session，然后返回结果给前端。前端接受结果判断是否退出成功，如果不成功，会提示错误原因。如果成功，会返回到最初的登陆页面。

5. 2. 4 公告管理流程设计

（1）查询功能流程设计

管理员可以通过搜索功能来查询查看对应的公告信息，搜索条件如表 5-5：

表 5-5 查询条件参数表

| 参数 | 可以为空 |
|-------|------|
| 公告名 | 是 |
| 创建人姓名 | 是 |
| 公告类型 | 是 |

管理员从界面输入或选择条件，点击查询，前端通过接口将条件参数请求给服务器，服务器接受数据后处理参数并从数据库中根据条件参数查出对应的公告信息返回给前端，前端处理数据后并渲染查询结果。

（2）添加公告功能流程设计

管理员从页面点击发布公告，触发相应的添加事件，填入对应的公告信息，公告信息如下表 5-6：

表 5-6 公告信息添加参数表

| 参数 | 可以为空 |
|------|------|
| 公告名 | 否 |
| 内容 | 否 |
| 公告类型 | 否 |

管理员输入完信息后，点击添加按钮，前端通过接口将填写的数据传给后端。后端经过一些逻辑判断对数据进行数据操作后，将添加结果给前端，前端接受数据处理后，再将结果渲染。

（3）修改公告功能流程设计

管理员从界面点击修改按钮，前端会将该条数据的 id 传给服务器，服务器根据 id 查询该条公告的信息并返回给前端，前端处理数据后渲染成修改公告对话框，提供给管理员修改数据。可修改的信息如下表 5-7：

表 5-7 公告信息修改参数表

| 参数 | 可以为空 |
|------|------|
| 公告名 | 否 |
| 内容 | 否 |
| 公告类型 | 否 |

管理员修改完信息后，点击确认按钮，前端会通过接口将数据传递给服务器。服务器接收后处理数据，再根据公告的 id 修改该公告在数据库中的信息。修改完后将结果返回给前端，前端处理后将结果渲染。

（4）删除公告功能流程设计

管理员点击界面上的删除按钮，前端会将该公告信息的 id 通过接口传递给服务器，服务器会根据 id 从数据库中删除该公告。之后将删除结果返回给前端，前端接受结果处理后，渲染界面提示是否删除成功。

5.2.5 社区功能流程设计

用户可以通过社区来查看该系统所有用户分享的动态。

（1）发布动态功能流程设计

用户点击界面上的发布按钮，会展示发布动态的模态框，用户输入内容点击发布，前端会将数据发送给服务器，服务器接收后经过处理将数据存入数据库，再将结果返回给页面。添加信息如下表 5-8：

表 5-8 发布动态参数表

| 参数 | 可以为空 |
|------|------|
| 动态内容 | 否 |

(2) 评论动态功能流程设计

用户可以在喜欢的动态下发布评论交流，在每条动态下方有个评论区域，输入完内容后，点击评论，前端把评论的内容传输给服务器，服务器做完处理后存入数据库中，然后再将结果返回。

(3) 查看我的动态功能流程设计

用户点击页面上的我的动态，即可调到自己发布的动态页面。

(4) 删除我的动态流程设计

用户在自己动态页面上会有个删除按钮，点击删除按钮，前端会把这条数据的 id 传输给后端，后端会根据该 id 去数据库将数据删除。再将结果返回给前端。

5.2.6 校友状况功能流程设计

(1) 查询功能分析

用户可以通过条件来查询对应的校友状况的信息，默认是显示所有校友的信息。查询条件如下表 5-9:

表 5-9 查询校友信息条件参数表

| 参数 | 可以为空 |
|------|------|
| 姓名 | 是 |
| 手机号 | 是 |
| 邮箱 | 是 |
| 学历 | 是 |
| 入学时间 | 是 |
| 工作 | 是 |

用户从界面输入或选择条件，点击查询，前端通过接口将条件参数请求给服务器，服务器接受数据后处理参数并从数据库中根据条件参数查出对应的教材信息返回给前端，前端处理数据后并渲染查询结果显示给用户。

5.2.7 查看公告功能流程设计

(1) 查询功能流程设计

用户每次登陆系统后，会自动跳到主页，主页会展示近期管理员发布的活动以及新闻。

5.2.8 修改个人信息功能流程设计

用户点击基本信息二级菜单，即可弹出修改个人信息页面。可输入的信息参数如下表

5-10:

表 5-10 个人信息修改参数表

| 参数 | 可以为空 |
|-----|------|
| 手机号 | 否 |
| 邮箱 | 否 |
| 学历 | 否 |
| 工作 | 否 |

用户从界面输入信息，点击保存，前端通过接口将条件参数请求给服务器，服务器接受数据后处理参数并从数据库中根据参数修改对应的用户信息。之后将修改结果响应给前端，前端处理数据后并渲染页面提示用户。

5.2.9 修改密码功能流程设计

用户点击密码修改二级菜单，即可弹出修改个人密码页面。可输入的信息参数如下表

5-11:

表 5-11 个人密码修改参数表

| 参数 | 可以为空 |
|-------|------|
| 旧密码 | 否 |
| 新密码 | 否 |
| 确认新密码 | 否 |

用户从界面输入信息，点击修改，前端通过接口将参数请求给服务器，服务器接受数据后处理参数并从数据库中根据参数修改对应的用户信息。之后将修改结果响应给前端，前端处理数据后并渲染页面提示用户。

5.2.10 用户管理功能流程设计

(1) 查询用户功能流程设计

管理员可以通过输入用户名来查询用户信息，默认是显示所有用户信息。查询条件如下表 5-12:

表 5-12 查询用户信息参数表

| 参数 | 可以为空 |
|------|------|
| 姓名 | 是 |
| 手机号 | 是 |
| 邮箱 | 是 |
| 学历 | 是 |
| 入学时间 | 是 |
| 工作 | 是 |

管理员从界面输入用户名，点击查询，前端通过接口将参数请求给服务器，服务器接受数据后处理参数并从数据库中根据参数查出对应的用户信息返回给前端，前端处理数据后并渲染查询结果显示给管理员。

（2）删除用户功能流程设计

管理员点击界面上的删除按钮，前端会将该用户的 id 通过接口传递给服务器，服务器会根据 id 从数据库中删除该用户。之后将删除结果返回给前端，前端接受结果处理后，渲染界面提示管理员是否删除成功。

5.3 界面设计

5.3.1 主页页面

用户登陆成功后，会自动跳转到主页页面。主页面会展示近期管理员新增的新闻或者活动公告。

5.3.2 社区页面

点击社区菜单，进入社区页面。在社区页面中，可以看到其他用户发布的动态。当然也可以评论他们的动态。在页面的右上方有发布动态按钮，点击发布动态，会弹出抽屉框，输入内容，点击发布即可。还有一个我的动态按钮，点击会跳转到我的动态页面。在我的动态页面，用户可以看到自己的历史动态和删除动态。

5.3.3 校友状况页面

点击校友状况菜单，进入校友状况页面。用户可以根据条件查找相应的校友信息。

5.3.4 个人中心页面

点击个人中心菜单，或者页面右上角个人设置按钮。用户可以进入自己的信息页面。

用户在此页面可以修改自己的个人信息以及修改密码。

用户管理页面

管理员点击用户管理菜单，可以进入用户管理页面。此页面会展示系统所有用户信息，并且管理员可以管理这些信息。

5.3.5 公告管理页面

管理员点击公告管理菜单，可以进入公告管理页面。此页面会展示系统所有公告信息，并且管理员可以管理这些信息。

第 6 章 程序设计

6.1 登陆实现

用户进入系统，首先进入的是登陆界面，在登录框中输入用户名和密码，才可进入系统。具体实现如图 6-1 所示。

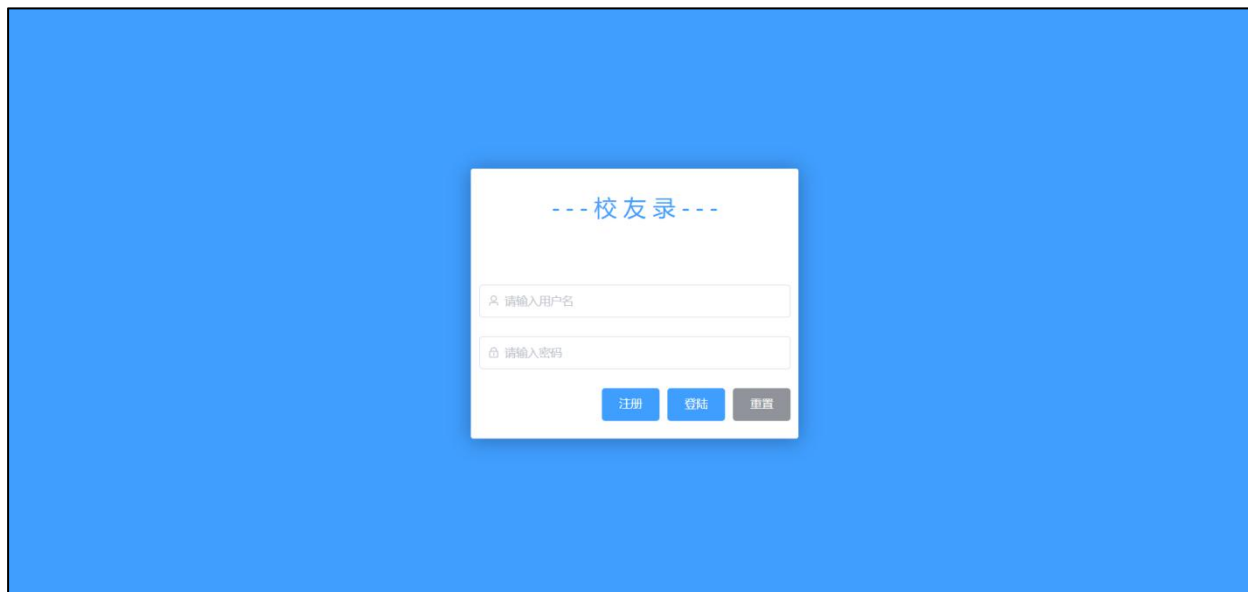


图 6-1 登陆界面图

登陆界面的代码为 `login.vue`，输入完用户名密码点击登陆按钮，前端表单通过 `axios` 发送异步请求给后端接口 `/user/login`，提交方式为 `post`，后端通过 `SpringMVC` 去接受发送过来的用户名和密码。后端接受到数据后，调用 `userService` 的 `login` 方法去处理逻辑。处理完逻辑后，调用 `userMapper` 去与数据库交互，若在数据库中根据用户名和密码查到该用户的信息，则返回成功给页面，即进入系统。

关键代码如下代码 6-1 所示。

代码 6-1:

```
/**
 * 用户登陆
 *
 * @param user
 * @return
 */
```

```
@RequestMapping("/login")

public String login(@RequestBody User user, HttpServletResponse response) {

    User loginUser = userService.login(user);

    if (loginUser == null) {

        return JSON.toJSONString(new Result<>("202", "没有该用户", null));

    }

    if (!loginUser.getPassword().equals(DigestUtils.md5DigestAsHex(user.getPassword().getBytes()))) {

        return JSON.toJSONString(new Result<>("201", "密码不正确", null));

    }

    QueryWrapper<Session> queryWrapper = new QueryWrapper<>();

    queryWrapper.eq("user_id", loginUser.getId());

    Session session = sessionMapper.selectOne(queryWrapper);

    Date now = new Date();

    //如果有该用户的 session

    if (session != null) {

        //还在 session 失效时间之内

        if (now.getTime() - session.getLoginTime().toInstant(ZoneOffset.of("+8")).toEpochMilli() <=

7200 * 1000) {

            //更新登录时间

            session.setLoginTime(now.toInstant().atOffset(ZoneOffset.of("+8")).toLocalDateTime());

            UpdateWrapper updateWrapper = new UpdateWrapper();

            sessionMapper.updateById(session);

        } else {

            //如果已超过时间，先删了该 session

            sessionMapper.deleteById(session.getSessionid());

        }

    } else {

        //没有的话创建新的 session

        session = new Session();

    }

}
```

```
        session.setSessionid(UUID.randomUUID().toString());

        session.setUserId(loginUser.getId());

        session.setLoginTime(now.toInstant().atOffset(ZoneOffset.of("+8")).toLocalDateTime());

        sessionMapper.insert(session);
    }

    String sessionId = session.getSessionid();

    UserVo userVo = new UserVo();

    userVo.setId(loginUser.getId())

        .setName(loginUser.getName())

        .setPassword(loginUser.getPassword())

        .setPhone(loginUser.getPhone())

        .setMail(loginUser.getMail())

        .setType(loginUser.getType())

        .setCreateTime(loginUser.getCreateTime())

        .setUpdateTime(loginUser.getUpdateTime())

        .setAdmissionTime(loginUser.getAdmissionTime())

        .setHighestDegree(loginUser.getHighestDegree())

        .setWork(loginUser.getWork())

        .setSessionId(sessionId);

    response.addCookie(new Cookie("sessionId", sessionId));

    return JSON.toJSONString(new Result<>("200", "成功", userVo));
}
```

6.2 主页实现

用户可以在主页查看公告，具体实现如图 6-2。

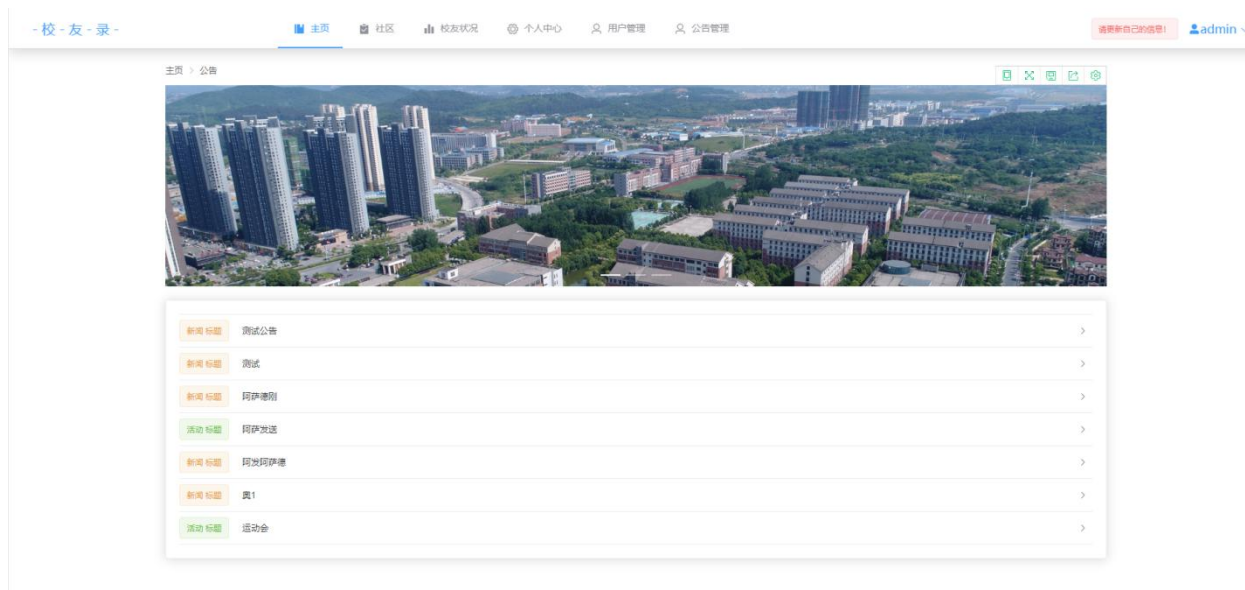


图 6-2 主页

用户登陆系统后会自动跳到主页面。主页面分为上下两块，上面一块的图片是前端 `vue` 获取静态图片，然后展示到页面中。下面一块当前端进入主页面中则会自动调用后端接口 `/announcement/queryAllAnnouncement`，后端接收到数据后会调用 `announcementService` 的 `queryAllAnnouncement` 方法，在 `announcementService` 处理完逻辑后，会调用 `announcementMapper` 去与数据库交互，查出所有的公告信息。然后后端返回给前端，前端通过 `vue` 在浏览器上渲染出页面。

关键代码如下代码 6-2 所示：

代码 6-2：

```
<!--          路由占位符-->

<router-view></router-view>

<el-row>

  <el-col :span="4">

    <div class="grid-content bg-purple-dark"></div>

  </el-col>

  <el-col :span="16">

    <div class="grid-content bg-purple-dark">

      <!--          面包屑导航区-->

      <el-breadcrumb separator-class="el-icon-arrow-right" style="margin-top: 25px">

        <el-breadcrumb-item>主页</el-breadcrumb-item>
```

```

        <el-breadcrumb-item>公告</el-breadcrumb-item>

        <!--          <el-breadcrumb-item>基本信息</el-breadcrumb-item>-->
    </el-breadcrumb>

    <el-card>

        <el-collapse v-model="activeNames" @change="handleChange">

            <el-collapse-item v-for="(item,i) in announcements" :title="item.name"
                                :name="item.id">

                <template slot="title">

                    <el-tag      v-if="item.type      ===      '1'"      type="warning"
style="margin-right: 20px">新闻  标题

                    </el-tag>

                    <el-tag      v-if="item.type      ===      '2'"      type="success"
style="margin-right: 20px">活动  标题

                    </el-tag>

                    {{item.name}}

                </template>

                <div>

                    <el-alert

                        v-if="item.type === '2'"

                        :title="item.content"

                        type="success"

                        :closable="false">

                    </el-alert>

                    <el-alert

                        v-if="item.type === '1'"

                        :title="item.content"

                        type="warning"

                        :closable="false">

                    </el-alert>

                </div>

            </el-collapse-item>

        </el-collapse>

    </el-card>

```

```

        </el-collapse-item>

    </el-collapse>

</el-card>

</div>

</el-col>

<el-col :span="4">

    <div class="grid-content bg-purple-dark"></div>

</el-col>

</el-row>

</el-container>

</template>

```

6.3 发布动态实现

用户点击发布动态可以弹出发布模态框。具体实现如图 6-3 所示。

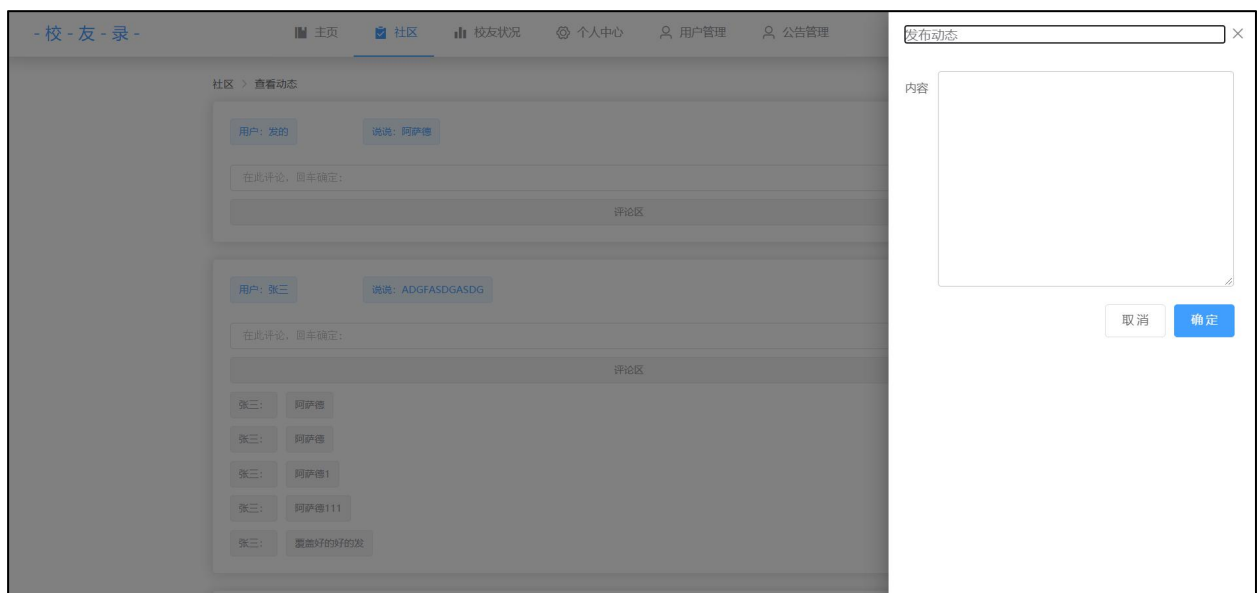


图 6-3 发布动态界面图

用户点击发布动态按钮，则会出发抽屉框。页面会从右边弹出发布动态内容的抽屉框。用户在该框中输入完内容，点击确定时，前端则会通过 `axios` 异步调用后端接口 `/talk/insert` 接口。后端接受到动态的数据后会调用 `talkService` 的 `insert` 方法。在 `talkService` 方法中处理完逻辑后，会调用 `talkMapper` 与数据库交互，将新增的动态插入数据库中。插入成功则

会返回状态码 1 给前端，失败则返回 0。

关键代码如下代码 6-3 所示：

代码 6-3：

```
<div>
  <el-input
    :id="item.id"
    type="textarea"
    placeholder="在此评论，回车确定:"
    v-model="comment[item.id]"
    size="small"
    rows="1"
    autosize
    ref="commentRef"
    style="width: 93%">

  </el-input>
  <el-button size="small" type="primary" style="margin-left: 1.2%"
    @click="submitComment(item.id)">评论
  </el-button>
</div>
<div style="margin-top: 10px">
  <el-tag type="info" style="margin-right: 10px;width: 100%;text-align:
center">评论区</el-tag>
</div>
<div style="margin-top: 10px" v-for="com in commentList" v-if="com.talkId ===
item.id">

  <el-tag type="info" style="margin-right: 10px">{{com.commentName}} :

</el-tag>

  <el-tag type="info">{{com.content}}</el-tag>
</div>
</el-card>
<el-pagination
  @current-change="handleCurrentChange"
  background
  layout="total, prev, pager, next"
  :total="talk.total"
  :page-size="5">

</el-pagination>
</div>
</el-col>
<el-col :span="4">
  <div class="grid-content bg-purple-dark">
    <div>
      <el-tooltip class="item" effect="dark" content="发布动态" placement="right">
```

```

        <el-button type="primary" icon="el-icon-edit" circle
            style="margin-left: 80px;margin-top: 55px"
            @click="insertTalkVisible = true"></el-button>
    </el-tooltip>
</div>
<di>
    <el-tooltip class="item" effect="dark" content="我的动态" placement="right">
        <el-button type="warning" icon="el-icon-star-off" circle
            style="margin-left: 80px;margin-top: 30px"
            @click="toMyTalk"></el-button>
    </el-tooltip>
</di>
</div>
</el-col>
</el-row>

<el-drawer
    title="发布动态"
    :visible.sync="insertTalkVisible"
    direction="rtl"
    custom-class="demo-drawer"
    ref="drawer"
>
    <div class="demo-drawer__content">
        <el-form :model="insertTalkForm">
            <el-form-item label="内容" label-width="60px">
                <el-input v-model="insertTalkForm.content" type="textarea" style="width:
360px"
                    rows="12"></el-input>
            </el-form-item>
        </el-form>
        <div class="demo-drawer__footer" style="text-align: right;margin-right: 40px">
            <el-button @click="cancelForm">取 消</el-button>
            <el-button type="primary" @click="insert" :loading="insertTalkLoading">{{
                loading ? '提交中 ...' : '确 定' }}
            </el-button>
        </div>
    </div>
</div>

```

6.4 评论动态实现

用户可以在动态的下方评论动态。具体实现如图 6-4 所示。

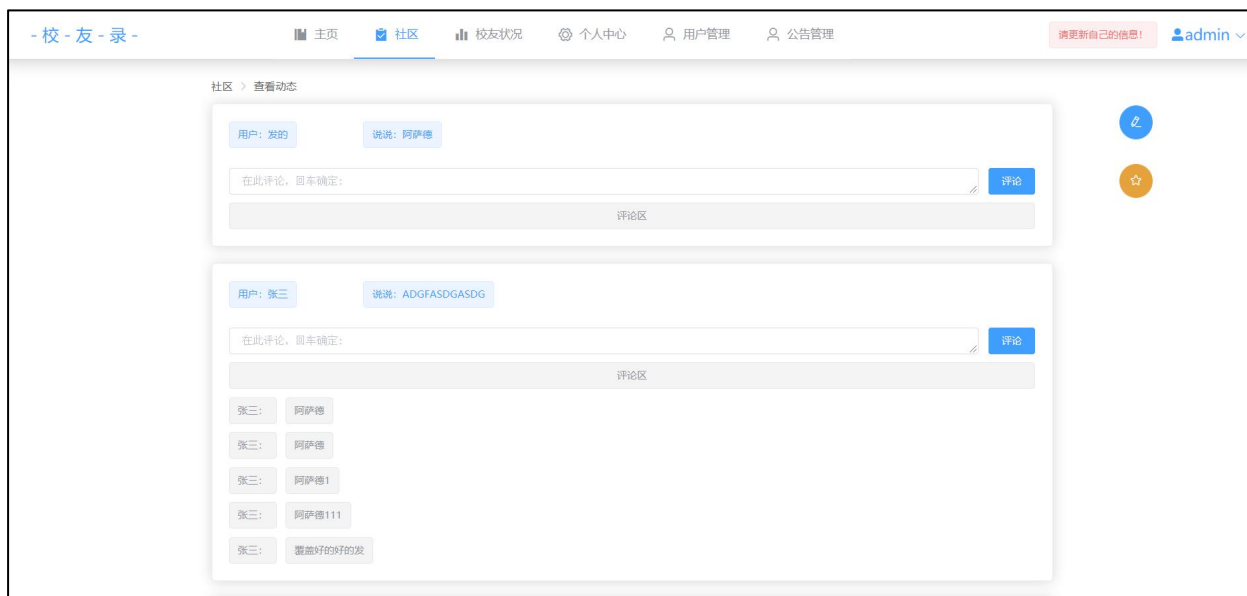


图 6-4 评论动态界面

用户在页面的评论 input 框中输入评论的内容，点击评论后，前端会把评论的内容、评论人和评论的动态 id 通过 axios 异步发给后端接口 `/comment/insert`。后端接受到数据后会调用 `commentService` 的 `insert` 方法。`commentService` 会将这些数据交给 `commentMapper` 去给数据库交互，将评论内容插入数据库中。插入成功后，会给前端返回状态码 1，失败则为 0。如果为 1，页面会将此动态 id 继续通过 axios 异步调用传递给后端接口 `/comment/queryAllComment`，后端接收后数据后调用 `commentService` 的 `queryAllComment` 方法，该方法会将数据给 `commentMapper` 与数据库交互，查出该条动态所有的评论，然后返回给前端到浏览器上渲染结果。如果为 0，则会提示评论失败。

关键代码如下代码 6-4 所示：

代码 6-4：

```
/**
 * 评论某个动态
 * @param comment
 * @return
 */
@RequestMapping("/comment")
public String comment(@RequestBody Comment comment,@RequestAttribute(value = "session.user")
User loginUser) {

    comment.setCommentName(loginUser.getName());
```

```

int result = commentService.comment(comment);

if (result > 0) {

    List<Comment> commentList = commentService.getCommentByTalkId(comment);

    return JSON.toJSONString(new Result<>("200", "更新成功,获得该动态的所有评论",
commentList));

}

return JSON.toJSONString(new Result<>("201", "更新失败", result));

}

```

6.5 公告管理实现

管理员可以进入公告管理页面去管理公告，具体实现如图 6-5 所示。

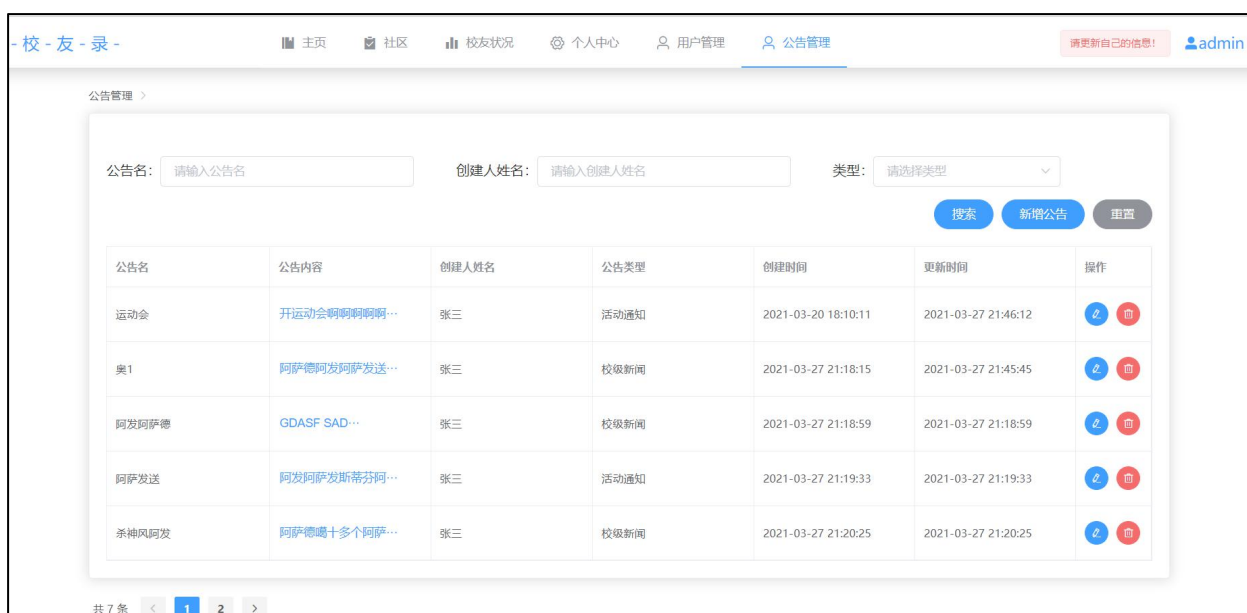


图 6-5 公告管理界面图

管理员可以进入公告管理界面，进入该界面后，前端会自动调用后端的/announcement的分页查询公告的方法。然后调用 announcementService 去做逻辑处理，处理完逻辑后，调用 announcementMapper 去与数据库交互，将公告信息按分页信息查出返回给前端。前端将数据通过 element 组件将表格按钮渲染出来。

关键代码如下代码 6-5 所示：

代码 6-5：

```

@ApiOperation("默认导出所有登记教材")
@GetMapping("/exportAllTextBook")

```

```
public void exportAllTextBook(@RequestAttribute(value = "session.user") User loginUser,
                               HttpServletResponse response) {
    try {int index = 0, flag = 0, flag1 = 0, flag2 = 0, flag3 = 0, flag4 = 0, flag5 = 0;
        List<CellRange> cellRangeList = new ArrayList<>();
        logger.info("用户-" + loginUser.getUserName() + "-导出所有登记教材");
        //查询所有登记教材
        List<TextBook> textBookList = textBookService.getTextBookList();
        //创建 excel 导出对象
        ExcelData excelData = new ExcelData();
        String[] head = {
            "编号", "课程号", "课程名"
            , "课程属性", "开课院系部", "任课教师"
            , "任课教师手机", "教材名称", "作者"
            , "出版社", "出版日期", "ISBN 码"
            , "出版版次", "教材级别（规划教材/一般教材/自编教材）", "所在学期"};
        excelData.setFileName("皖江工学院教材登记表");
        excelData.setHead(head);
        excelData.setTitle("皖江工学院教材登记表（全部）");
        excelData.setList(textBookList);
        //导出
        //创建个 excel
        HSSFWorkbook workbook = new HSSFWorkbook();
        //创建个 sheet
        HSSFSheet sheet = workbook.createSheet("教材登记表（全部）");
        //合并单元格
        sheet.addMergedRegion(new CellRangeAddress(0, 0, 0, 14));
        //创建标题
        ExcelUtil.setTitle(workbook, sheet, excelData);
        //设置表格表头
        ExcelUtil.setHead(workbook, sheet, excelData.getHead(), 1);
        //设置值
        //设置格式
        HSSFCellStyle style = workbook.createCellStyle();
        HSSFFont font = workbook.createFont();
        font.setFontName("宋体");
```

```
style.setFont(font);
//自动换行
style.setWrapText(true);
//垂直居中
style.setAlignment(HorizontalAlignment.LEFT);
style.setVerticalAlignment(VerticalAlignment.CENTER);
//设置单元格宽度
sheet.setColumnWidth(0, 5 * 256);
//合并单元格
for (int i = 0; i < cellRangeList.size(); i++) {
    sheet.addMergedRegion(new CellRangeAddress(
        cellRangeList.get(i).getStartRow(),
        cellRangeList.get(i).getEndRow(),
        cellRangeList.get(i).getStartCol(),
        cellRangeList.get(i).getEndCol()));
}
//设置浏览器下载
ExcelUtil.setBrowser(response, workbook, excelData.getFileName());
} catch (Exception e) {
    logger.error("用户-" + loginUser.getUserName() + "-导出所有登记教材失败", e);
}
}
```

6.6 个人中心实现

用户进入个人中心，可以修改个人信息密码，具体实现如图 6-6 所示。

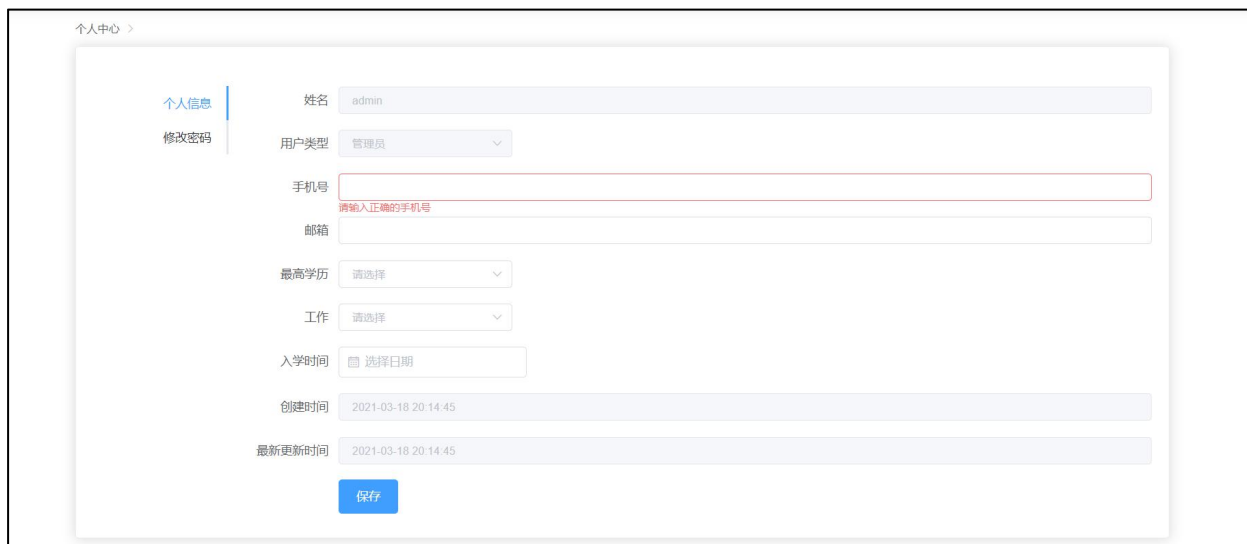


图 6-6 个人中心界面

用户进入个人中心界面。将想要修改的信息填入页面的表单中，点击保存。前端会调用后端的 `/user/update` 接口。该接口会通过 `springmvc` 将前端传入的表单信息全部接受到，然后封装到一个 `pojo` 实体类中。然后后端将会把这个实体类传递给 `userService` 的 `update` 方法，在其中做逻辑处理，处理完逻辑后，会调用 `userMapper` 与数据库交互，将该用户修改的信息与数据中的原数据替换。成功则会返回给前端状态码 `1`，失败则为 `0`。前端再根据该状态码去给用户反馈结果。

关键代码如下代码 6-6 所示：

代码 6-6：

`@RequestMapping("/update")`

```
public String update(Integer id, String name, String phone, String mail, String admissionTime, String highestDegree, String work) {
```

```
    DateTimeFormatter df = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
```

```
    User user = new User();
```

```
    user.setName(name)
```

```
        .setId(id)
```

```
        .setPhone(phone)
```

```
        .setMail(mail)
```

```
        .setAdmissionTime(LocalDate.parse(admissionTime, df))
```

```
        .setHighestDegree(highestDegree)
```

```
        .setWork(work);
```

```
    int result = userService.update(user);
```

```
    return JSON.toJSONString(new Result<>("200", "更新成功", result));
```

```
}

@RequestMapping("/delete/{id}")
public String delete(@PathVariable("id") Integer id) {
    boolean remove = userService.removeById(id);
    return JSON.toJSONString(new Result<>("200", "删除成功", remove));
}

@RequestMapping("/queryAllUser")
public String queryAllUser(int currentPage, int pageSize, String name, String phone, String mail, String
admissionTime, String highestDegree, String work) {
    Page<User> userList = userService.queryAllUser(currentPage, pageSize, name, phone, mail,
admissionTime, highestDegree, work);
    return JSON.toJSONString(new Result<>("200", "查询用户成功", userList));
}

@GetMapping("/getLoginUser")
public String getLoginUser(@RequestAttribute(value = "session.user") User loginUser) {
    return JSON.toJSONString(new Result<>("200", "查询用户", loginUser));
}

//退出登录
@PostMapping(value = "/signOut")
public String signOut(@RequestAttribute(value = "session.user") User loginUser,
    HttpServletRequest request) {

    try {
        sessionService.signOut(loginUser);
        request.removeAttribute("session.user");
        return JSON.toJSONString(new Result<>("200", "退出成功", null));
    } catch (Exception e) {
        return JSON.toJSONString(new Result<>("201", "退出失败", null));
    }
}

@PostMapping("/modifyPassword")
```

```
public String modifyPassword(@RequestBody JSONObject json, @RequestAttribute(value =
"session.user") User loginUser) {
    String oldPassword = json.getString("oldPassword");
    if (!loginUser.getPassword().equals(DigestUtils.md5DigestAsHex(oldPassword.getBytes())) {
        return JSON.toJSONString(new Result<>("201", "旧密码不一致", null));
    }
    String newPassword = json.getString("newPassword");
    User newUser = new User();

    newUser.setId(loginUser.getId()).setPassword(DigestUtils.md5DigestAsHex(newPassword.getBytes()));
    userService.update(newUser);
    return JSON.toJSONString(new Result<>("200", "修改密码成功", loginUser));
}
```

第 7 章 系统测试

7.1 系统测试的目的和意义

系统测试目的是尽可能多地找出错误，提高系统的可靠性。防止在线上环境产生问题。为了能稳定的运行，需要开发者在开发阶段就需要不断的调试。只有经历过无数遍的调试，才能完成一个完整的稳定的系统。系统测试的意义是为了让系统更加安全稳定的运行下去。

7.2 功能测试

7.2.1 发布动态测试

在发布动态的抽屉框里输入信息后点击发布，具体如下图 7-1 所示：

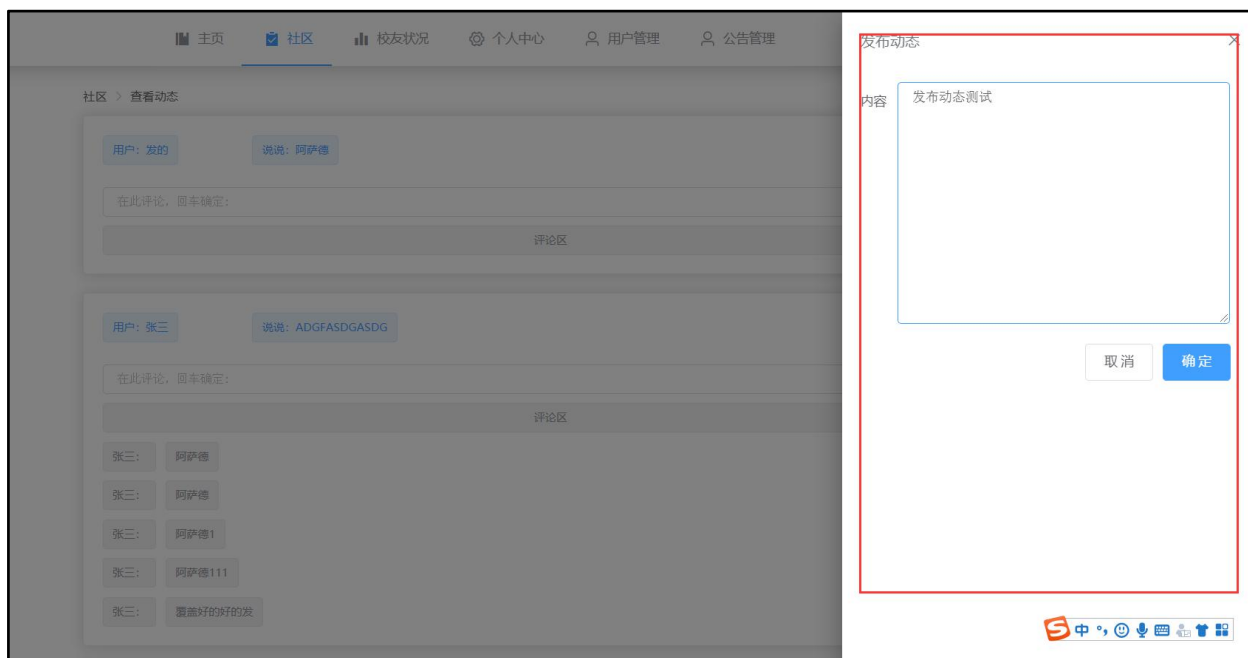


图 7-1 发布动态测试图

点击发布后，可以看见最新的动态已经更新到页面上。具体如下图 7-2 所示：

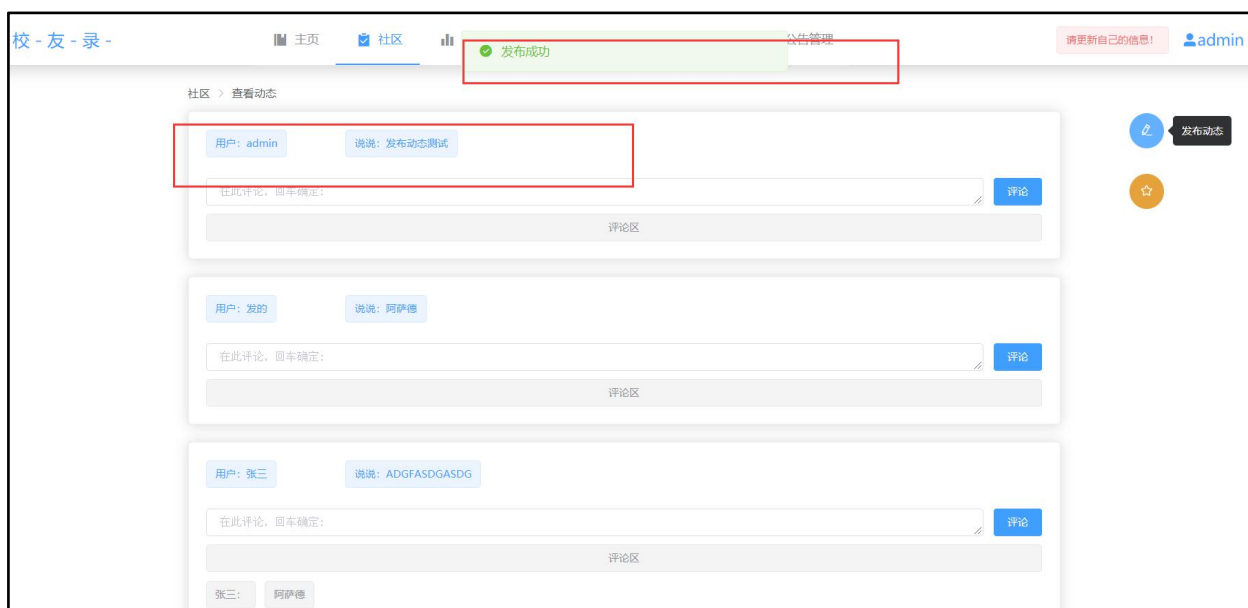


图 7-2 发布动态成功测试图

7.2.2 评论动态测试

在动态评论区内输入内容，点击提交评论，具体如下图 7-3 所示：

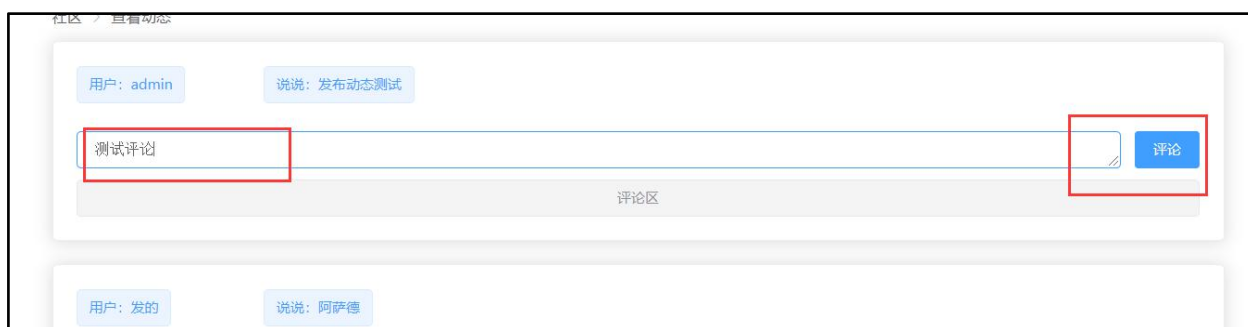


图 7-3 评论测试图

评论成功后，会提示评论成功并且动态下会出现该评论。具体如下图 7-4 所示：

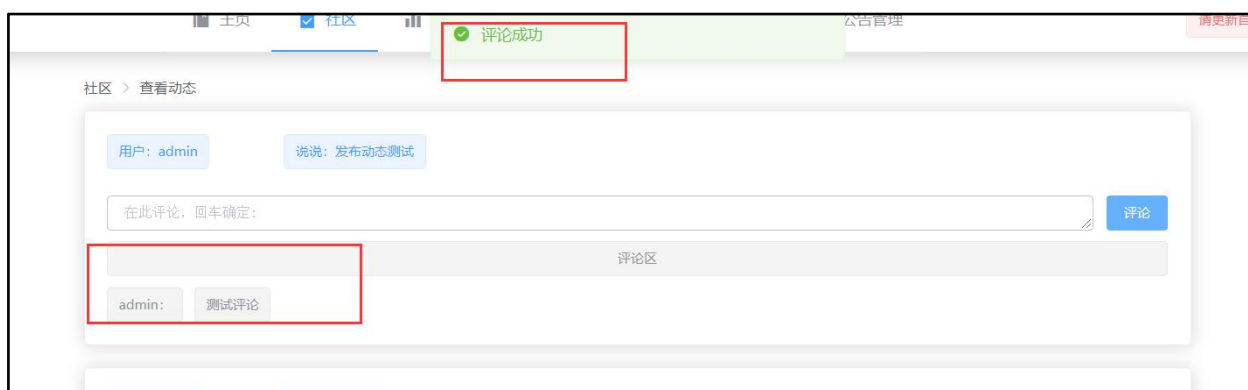


图 7-4 评论成功测试图

7.2.3 修改个人信息测试

在个人信息处可以修改自己的信息。这些信息有的会有验证，具体如下图 7-5 所示：

个人信息

修改密码

姓名 admin

用户类型 管理员

手机号 请输入正确的手机号

邮箱

最高学历 请选择

工作 请选择

入学时间 选择日期

创建时间 2021-03-18 20:14:45

最新更新时间 2021-03-18 20:14:45

保存

图 7-5 修改个人信息测试图

输入正确的信息后，点击保存，会提示修改信息成功，并展示最新的信息。具体如下图 7-6 所示：

修改信息成功

个人中心

个人信息

修改密码

姓名 张三

用户类型 普通用户

手机号 15695555555

邮箱 1242阿萨德3@qq.com

最高学历 读研中

工作 普通企业

入学时间 2021-03-23

创建时间 2021-03-18 20:14:45

最新更新时间 2021-04-11 21:29:13

保存

图 7-6 修改信息成功测试图

7.2.4 新增公告测试

管理员在新增公告模态框输入公告内容，这些内容会有一定的验证，具体如下图 7-7

所示：

新增公告

*

公告名称

请输入公告名

*

类型

请选择类型

请选择类型

*

内容

请输入内容

请输入内容

取消

确定

图 7-7 新增公告测试图

输入正确的信息后，点击确定，即可新增公告，新增成功后会提示新增成功，并在页面上显示。具体如下图 7-8 所示：

公告名：请输入公告名

创建人姓名：请输入创建人姓名

类型：请选择类型

搜索

新增公告

重置

| 公告名 | 公告内容 | 创建人姓名 | 公告类型 | 创建时间 | 更新时间 | 操作 |
|------|------|-------|------|---------------------|---------------------|-------------------------|
| 测试 | 测试 | admin | 校级新闻 | 2021-03-28 20:53:02 | 2021-03-28 20:53:02 | <div></div> <div></div> |
| 测试公告 | 测试公告 | admin | 校级新闻 | 2021-04-11 21:33:22 | 2021-04-11 21:33:22 | <div></div> <div></div> |

共 7 条

<

1

2

>

图 7-8 测试公告成功图

7.3 可用性测试

可用性测试主要测试一些操作上和理解上的一些测试。具体测试如下表 7-1：

表 7-1 可用性测试结果

| | |
|-----|----|
| 测试项 | 结果 |
|-----|----|

| | |
|-------------------------|---|
| 窗口移动、大小改变、关闭等操作是否正常 | 是 |
| 操作模块是否友好 | 是 |
| 模块、提示内容等文字描述是否正确 | 是 |
| 模块布局是否协调、合理 | 是 |
| 模块的状态是否正确（对选中项能否发生对应切换） | 是 |
| 鼠标、键盘操作是否支持 | 是 |
| 所需数据项是否正确显示 | 是 |
| 操作流程是否合理 | 是 |
| 是否提供帮助信息 | 是 |

7.4 测试结果分析

经过一系列的测试，根据测试的结果，该系统的功能非常完善并且操作简便。很符合之前的分析和设计。本系统在设计的时候也参考了许多文档。该系统的成功离不开各方各面的帮助。

总结与展望

总结

该校友录系统整体功能基本完善。整体界面使用的是蓝白颜色，简洁大方，格局框架分割的很整齐，不会使用户看了感觉别扭。对于一些提示也很人性化的体现出来。每个功能都用分级菜单很好的展示出来。用户可以简单的发布动态。表格清晰明了的显示了每个数据的信息。对于每个按钮都有提示信息，可以提示用户该按钮的意思。对于操作的成功失败都用绿色和红色体现出来，便于用户分辨。

展望

该系统只实现了一些基础的功能，还有很多功能没考虑到，没有实现。例如好友功能、关注功能和二级评论等功能。这次的系统设计让我对业务有了很深的重视。我坚信再接下来的时间，我会更加努力的学历技术和业务方面的知识来完善这次系统。没有最好，只有更好，只有跟上时代的进步才能把握住未来。

致谢

这次毕设的完成，离不开同学老师对我的支持。首先我要感谢自己的导师，在导师的细心指导下，让我熟悉了这次课题的一些业务。然后感谢在我开发的过程中，遇上的一些问题那些帮助过我的同学。通过这次的毕设，也让我明白了自学的重要性。每当我想不到技术解决问题的时候，我都先会百度去寻找问题，如果解决不到，再回去和同学老师讨教。因为在未来的社会生活中，只有靠自己才能展望未来。

最后，也感谢学校这四年来对我的辛勤耕耘。我相信，我能在未来走的更好更远。人无完人，学无止境。在完成毕设后我越发觉得自己需要更多的学习，以应对不断变化的需求，生命的光华只有在不断的升腾中才能得到最佳的呈现，生命的精彩只有在不断的超越中才能拥有完美的体验。

参考文献

- [1]肖龙星. 基于微信小程序的高校校友录系统设计与实现[J]. 中国战略新兴产业, 2019,(22):253,255.
- [2]徐化祥. 基于 Java 技术的校友录系统研究[J]. 电脑编程技巧与维护, 2015,(22):30,63. DOI:10.3969/j.issn.1006-4052.2015.22.016.
- [3]符宇. 浅谈校友录系统设计[J]. 科技资讯, 2013,(27):173,239. DOI:10.3969/j.issn.1672-3791.2013.27.113.
- [4]王文越. 校友录管理信息系统研究[J]. 中小企业管理与科技, 2012,(30):271. DOI:10.3969/j.issn.1673-1069.2012.30.176.
- [5]李玮琦. 基于 Java Web 的校友录系统[J]. 网友世界·云教育, 2013,(5):13.
- [6]麻灵. 基于 C/S 和 B/S 混合结构的校友录管理系统[J]. 数字技术与应用, 2013,(1):153.
- [7]杨露希. 基于 Android 的移动校友录的设计与实现[D]. 四川:电子科技大学, 2014. DOI:10.7666/d.D500808.
- [8]郭康康. 基于 B/S 结构的校友录系统设计与实现[D]. 四川:电子科技大学, 2012. DOI:10.7666/d.D772230.
- [9]麻灵, 冉婧. 基于 ASP 的校友录信息系统设计[J]. 信息通信, 2014,(3):91-91.
- [10]户文贵. 基于 J2ME 的手机校友录系统的设计[D]. 江西:南昌大学, 2010. DOI:10.7666/d.y1850536.
- [11]刘环. 基于校友管理的创业信息平台的设计与研究——以山西省长治医学院为例[J]. 农业网络信息, 2016,(12):127-128.
- [12]何荣毅. 基于 JSP 的网络校友录的实现[J]. 科技信息(科学·教研), 2007,(5):45-46,48. DOI:10.3969/j.issn.1001-9960.2007.05.033.

附录一：

基于 J2EE 的研究项目虚拟账户系统的开发

引言

随着国内外经济的发展，项目管理中的信息越来越多。今天，项目管理的专业领域是多种多样的。因此，有必要采用合理的处理方法对大量信息进行及时整理和分类。这样，管理系统就可以根据各种法规进行排列组合，形成报表，然后通过传统方式或通过互联网作为分析和决策的基础，与领导进行沟通，传递和报告。项目管理的能力很重要，因为它被认为对项目绩效以及业务绩效有重大影响。为了改变手工计算带来的被动肤色，并建立合同和支出的管理标准以适应数据统计的需求，近年来出现了各种项目管理软件或方法。它们在开发基于信息的项目管理中扮演着非常重要的角色。然而，目前的管理软件总体上具有以下趋势：每种软件都是在其特殊的系统软件平台上开发的，其自身具有一套代码库，以适应具有不同特性和目的的管理。因此，当使用计算机管理研究项目时，就会出现以下问题：①对软件运行环境的要求非常苛刻。它需要单一类型的系统软件来支持，因此其不同平台上的操作能力很弱。②每种软件都有自己的系统。代码不统一，数据结构也不相同。实现的功能也不同。该软件缺乏数据文件之间的互换性。项目经理被动地输入各种数据，广大教职员工无法主动、实时地参与科研项目的管理和监督。③某些软件存在漏洞，某些网络潜在的安全隐患密码薄弱，IIS 漏洞，数据库漏洞等，这些在给不同部门的资料进行核实时给科研项目统计报告和科研贡献报告带来很多不便。因此，本文讨论了基于网络安全性以满足各种统计需求的大学研究项目管理系统的开发。此外，大学的研究工作模式可以从被动管理转变为主动管理，并可以提升大学科研资源的信息化水平。

系统开发方案

模型-视图-控制器-数据（MVCD）模型与 MVC 模型类似，它只是抽象数据源以形成独立的数据层。数据层将从模型层接收的数据转换为与指定数据库具有相同存储格式的数据，并将其存储到指定数据库。该模型将 JSP，JavaBean 和 JDBC 有机地结合在一起。数据类是一种特殊的 JavaBean，用作 Session Bean 和数据库之间的桥梁。大多数 Web 应用程序都有大量数据，需要与数据库链接。Session Bean 以结构形式收集所有数据，并将其返回给 JSP。介绍了 MVCD 模型在开发科学会计系统中的应用，具体如下：

年度科学技术报告表和验证数据表用于促进人员以及横向和纵向项目清单。一个具

体的开发方案是在项目主管的现有资金帐户的基础上建立一个二级帐户（虚拟帐户）。然后，导演将某些资金分配给参与者。此外，在给定的时间段内，该软件将自动计算大学，研究所，部门，项目负责人，项目参与方等所有项目的资金库存，以满足年度报告统计的要求。

虚拟账户系统的数据库包括资金分配表，合同表，收入分配表和项目资金分配表等。每个表都包含科学项目的直接和间接信息。例如，资金分配表包括诸如项目名称，资金状况，检查节点，主管，参与者和项目来源等信息。同时，项目进行时可以将最新的项目进度输入该数据库。这样，它可以掌握所有科研项目的进度，协调解决需要及时注意的难点和程序。所有这些都有助于确保科学项目的和谐。

虚拟账户系统采用面向对象的开发方法。它以 weblogic 8.0 为服务器，通过 mysql 和访问来构建后台支持者的数据库。在此基础上，开发了查询数据库，升级数据库，修改数据库，处理汉字，限制 IP 地址，加密登录密码和管理项目资金等模块。系统的主要开发模块如下：

1 员工登录模块的设计

当工作人员通过浏览器访问 WWW 上的虚拟帐户系统主页时，第一页是工作人员登录对话框和新工作人员注册按钮。如果员工想要分配项目资金或查看其他人分配给他的资金，则他必须使用自己的帐户登录。新员工按照姓名和工资编号注册他的帐户。帐户和密码是确认员工身份的唯一证明。考虑到安全性，不允许同一帐户，并且密码同时进行相应的加密措施，以确保即使对数据库进行了分析和解释也找不到员工的登录密码。同时，为了方便教师在忘记密码的情况下及时找到密码，使用 java 的邮件程序包开发了电子邮件模块，可以在需要时将密码发送到相应用户的邮箱中。找回密码。

2 互联网项目信息查询模块

在 Internet 上查询项目信息的模块是虚拟帐户系统的重要组成部分。例如，系统建立了国家科研经费的纵向数据库，企业项目的横向数据库，项目分类数据库，经费分配数据库和职工信息数据库。这些数据库不仅具有输入，查询，修改，删除，打印，计数等常规功能，而且还可以根据员工的需求和权限实现汇总和动态调用。

3 分配模块

开发了资金分配模块，供项目负责人分配一定项目的资金，使项目负责人可以方便地增加，修改和删除项目参与人的资金，从而使项目得以顺利进行。

虚拟账户属于资金总监，并进行相应的加密。同时，只有给定的 IP 地址用户是合法用户，系统限制了非法 IP 用户的访问，从而提高了网络的安全性。

总结

（1）为了满足开发国家行政部门信息办公系统的需要。本文分析了可在 WEB 开发软件中有效使用的 MVCD 模型。以该模型为指导，设计了科研项目虚拟账户系统的整体框架结构，并开发了该系统的各个功能模块，即用户登录模块，项目详细查询模块，资金分配模块和模块。参考和修改年度报告。

（2）虚拟账户系统改善了管理处方，实现了高效无差错管理。在一年的运营过程中，每个机构管理层的项目甚至都没有故障。系统的顺利运行验证了基于网络安全性的 MVCD 模型在项目管理系统中的可行性和有效性。

附录二:

Development of Virtual Account System for Research Projects Based on J2EE

Introduction

With the economy developing at domestic and abroad, information in project management is growing increasingly. The professional field of project management today is diverse. So it is necessary to arrange and classify a large amount of information in time with a rational dealing method. In this way, the management system can put up permutation and combination in terms of various regulations to bring up report forms, then communicate, transfer and report to the leadership by traditional means or through internet as the basis of analysis and decisionmaking. Competence of project management personnel is important as it is considered to have a major impact on project performance and therefore on business performance. In order to change the passive complexion brought by handwork calculation and establish the management criterion of contract and outlay to adapt the needs of data statistics, various kinds of project management software or approaches emerge in recent years. They play a very important part in the developing of information-based projects management. However the present management software at large has the following trend: each kind of software is developed on its special system software platform with a set of code libraries of itself to suit for managements with different character and purpose. So problems appear when the research projects are managed with computer: ① the demand of the software's running environment is very harsh. It needs a single type of system software to support, so its operation ability over different platforms is weak. ② each kind of software has the system of its own. The codes are not unified and the structure of data is not the same. Also the function realized is different. The software lacks of interchangeability between data files. Project manager inputs various kinds of data passively and the masses of teaching and administrative staff can't participate in the management and supervision of the scientific research projects initiatively and real-timely. ③ some software has bugs, such potential safety hazards of the network as certain weak password, IIS loophole, database loophole, etc, And all these bring a lot of inconvenience to the scientific project statistics report and the report of scientific research

contribution when verifying the material of different departments. Therefore, this paper discusses the development of management system of university research projects based on network security to satisfy various kinds of statistics demands. Further, the research work mode of university can be transformed from passive management to initiative management and the information-based level of university scientific research resources can be promoted.

Developing Scheme of The System

Model-View-Controller-Data (MVCD) model is similar to MVC model, It just abstracts the data source to form an independent data layer. The data layer transforms data received from the model layer into data in the same storage format with the appointed database and stores it to appointed database. The model combines JSP, JavaBean and JDBC together organically. Data Class is a special JavaBean, which is served as the bridge between Session Bean and database. Most web application programs have plenty of data and need to link with database. Session Bean collects all the data in a structure and returns it to JSP. The application of MVCD model to develop scientific account system is introduced as follows:

The annual scientific and technology report form and the verify datum forms are used to promote staff and horizontal and vertical project inventory. A concrete developing scheme is to set up a secondary account (virtual account) on the basis of the existing fund account of the project director. Then the director distributes certain funds to the participants. In addition, in a given time slot, the software will count all projects' fund inventory automatically in terms of university, institute, department, project director, project participant to meet the request of annual report statistics.

The database of virtual account system includes fund allocation form, contract form, income distribution form and project fund assign form, etc. Each form includes direct and indirect information of scientific projects. For example, the fund allocation form includes such section of information as project's name, funds situation, examine node, director, participant and project source. Meanwhile, newest progress of project can be input to this database when the project is carried on. In this way, it can grasp the progress of all scientific projects, coordinate and solve the difficult point appearing and program needing noticing in time. All these .help to guarantee the harmony of scientific projects.

The virtual account system adopts the object oriented development approach. It regards

weblogic 8.0 as server, structures backstage supporter's database with mysql and access. On this basis, it develops such modules as inquiring database, upgrading database, modifying database, dealing with Chinese character, limiting IP address, encrypting login password and managing project funds, etc. The main developing modules of system are as follows:

1 The design of staff login module

When the staff visits the homepage of virtual account system on WWW through the browser, the first page is a dialog box for staff to login and a button for new staff to register. If the staff wants to distribute project funds or see the funds that others distribute to him, he must login with his own account. New staff registers his account in terms of his name and salary number. The account and password is the only evidence of confirming staff's identity. In consideration of security, identical account is not allowed and the password is carried on the corresponding encryption measure at the same time, in order to guarantee that staffs' login password can't be found even if the database is analyzed and explained. Meanwhile, in order to make it convenient for teachers to find the passwords in time in case of forgetting them, E-mail module is developed by using the mail package of java and it can send the password to the corresponding user's mailbox when he requires to get his password back.

2 The module of inquiring project information on Internet

The module of inquiring project information on Internet is an essential part of the virtual account system. For example, the system sets up vertical database of national scientific research funds, horizontal database of enterprise's project, database of project classification, database of fund distribution and database of staffs' information. These databases not only have the routine functions of inputting, inquiring, revising, deleting, printing, counting, etc, but also can realize summation and dynamic calls according to the staff's needs and his authority.

3 The module of distributing funds

The module of distributing funds is developed for the project director to assign funds of a certain project, through which the director can add, modify and delete funds of project participant conveniently, thus the project can be carried on successfully.

The virtual account belongs to the funds director, which is carried on corresponding encrypting. At the same time, only given IP address users are legal users and the system limits visiting of illegal IP users, thus the security of network is improved.

Conclusions

(1) In order to meet the needs of developing the information office system of national administrative department? the paper analyzes the MVCD model which is valid used in WEB developing software. Regard this model as guidelines,it designs the whole frame structure of the virtual account system of scientific research projects and develops each function module of the system, namely, module of users login, module of inquiring the project's detail, module of distributing funds and module of referring and modifying the annual report.

(2) The virtual account system has improved the prescription of management and achieved high-efficient error-free management. During the one-year operation, there is not even a fault in the projects of each institute management. The smooth operation of the system has verified the feasibility and validity of the MVCD model in the project management system based on network security.