**华南农业大学珠江学院  
毕业设计**

**基于SpringBoot＋Vue的人事管理系统**

杨膨键

指导教师： 郑艳松（讲师）

学院： 信息工程学院 年级专业： 2018级计算机科学与技术

提交日期： 2021年x月xx日 答辩日期： 2021年x月xx日

答辩委员会主席（签名）：

评阅人（签名）：

年 月 日

摘 要

随着互联网信息技术的发展，企业的对人事管理系统的功能要求不断提高。一个高质量、功能齐全的人事管理系统不仅能够简化公司业务的工作流程，还能满足不同角色权限操作不同的功能，便于企业领导更全面的掌握每个员工的基本信息。它可以减少劳动力的使用，提高员工的工作效率。因此，对于企业来说，使用一套高质量的人事管理系统进行管理，为企业创造更高的效益是非常重要的。

本系统使用IntelliJ IDEA开发工具，数据库使用MySQL，后端采用SpringBoot、Spring和Mybatis流行框架进行整合开发，使用SpringSecurity安全框架进行登录认证和权限校验。前端采用Vue框架和element-ui等技术进行页面设计，采用axios、vuex等技术进行前后端数据的交互，是基于SpringBoot+Vue前后端分离进行设计与实现。本文的人事管理系统主要分为员工资料模块、人事管理模块、工资管理模块以及系统管理模块。使其成为功能齐全，能够真正服务于一般企业的人事管理系统。该系统能够切换不同角色权限来操作不同的模块，保证系统的安全性。

本文对该人事管理系统进行功能性测试，通过测试验证了软件的实现和预期的结果一致，能够使企业极大的提高企业人事管理的效率和安全性。

关键词：人事管理系统; SpringBoot; MySQL; 权限检验

**Abstract**

With the development of Internet information technology, the functional requirements of enterprise personnel management system are constantly improving. A high-quality and fully functional personnel management system can not only simplify the workflow of the company's business, but also meet the different roles and permissions to operate different functions, which is convenient for enterprise leaders to have a more comprehensive grasp of the basic information of each employee. It can reduce the use of labor force and improve the work efficiency of employees. Therefore, it is very important for enterprises to use a set of high-quality personnel management system for management and create higher benefits for enterprises.

The system uses the IntelliJ idea development tool, the database uses mysql, the back-end uses the popular frameworks of springboot, spring and mybatis for integrated development, and uses the springsecurity security framework for login authentication and permission verification. The front end uses Vue framework, element UI and other technologies for page design, and uses Axios, vuex and other technologies for front and rear data interaction. It is designed and implemented based on the separation of front and rear ends of springboot + Vue. The personnel management system of this paper is mainly divided into employee information module, personnel management module, salary management module and system management module. Make it a fully functional personnel management system that can really serve general enterprises. The system can switch different roles and permissions to operate different modules to ensure the security of the system.

This paper tests the function of the personnel management system, and verifies that the implementation of the software is consistent with the expected results, which can greatly improve the efficiency and security of enterprise personnel management.

**Keywords:** personnel management system; SpringBoot;MySQL; Permission check

设计说明

设计任务的来源。

本课题的设计任务来源于本人的毕业设计项目：“基于SpringBoot＋Vue的人事管理系统”。

随着计算机网络技术的迅猛发展，各行各业的运营趋向于网络化、自动化。对于企业来说，人事管理系统是企业现代化管理不可或缺的一部分，它贯穿企业生产和经营的全过程，并服务于企业可持续化发展。

本文设计并实现的基于SpringBoot＋Vue的人事管理系统改变了传统的以纸质为媒介的人事管理方法存在的弊端，如保密性差、效率低、数据容易遗失、无法进行数据的实时查询、更新、维护等。企业引入人事管理系统可以简化许多复杂繁琐的工作流程，大大提高工作效率和准确性，节约管理成本，提高企业管理水平。因此，开发一套适合企业的人事管理系统具有深远的意义。

设计的技术路线。

（1）系统需求分析

在基于SpringBoot＋Vue的人事管理需求分析。通过调查了解用户，分析系统需求，引入系统目标，系统可行性分析，业务流程分析，功能性和非功能性需求分析，整体把握各个模块之间的需求关系。

（2）系统设计

系统设计是通过需要分析后得出对系统设计的方案，其中包括设计原则、架构设计和功能模块设计及设计思想，各个功能模块的结构与内容、各个模块的关系。同时，对系统的功能模块、流程进行划分并说明。

（3）数据库设计

数据库的概念设计，概念模型设计和逻辑结构设计，对系统所涉及的实体、关系进行梳理，给出E-R图，同时列出该系统运行所需要的数据表。

（4）系统具体实现

该阶段主要实现系统开发所需的环境搭建、各个功能的操作和实现界面，功能模块逻辑代码逻辑和程序的设计与实现。

（5）系统测试

该阶段主要对开发完成的系统进行各个功能模块的测试，并使用不同的权限角色登录系统，验证是否实现不同的角色拥有不同权限。对系统通过单元测试、集成测试和等测试系统能否符合预期。

关键词：设计路线；概要设计；数据库设计；Spring Boot；人事管理系统

目 录

1 引言 1

1.1 课题背景 1

1.2 国内研究情况 1

1.3 国外研究情况 1

1.4 课题研究目标及意义 2

2 相关框架和技术简介 3

2.1 Spring框架 3

2.2 SpringBoot框架 3

2.3 Mybatis框架 4

2.4 Vue框架 4

3 系统分析 5

3.1 可行性分析 5

3.1.1 技术可行性 5

3.1.2 操作可行性 5

3.1.3 经济可行性 5

3.2 需求分析 5

3.2.1 功能性需求分析 5

3.2.2 非功能性需求分析 6

3.2.3 系统用例图 7

4 人事管理系统设计 9

4.1 系统设计原则 9

4.1.1 系统设计目标 9

4.1.2 系统设计原则 9

4.2 系统架构设计 10

4.3 系统功能模块设计 11

4.3.1 系统设计目标 11

4.3.2 用户登录流程 11

4.3.3 人事管理模块 12

4.3.4 部门管理模块 13

5 数据库设计 15

5.1 数据库概念模型设计 15

5.2 数据库逻辑结构设计 17

6 人事管理系统的实现 21

6.1 系统开发及运行环境 21

6.2 系统实现 21

6.2.1 项目目录结构 21

6.2.2 系统登录模块 22

6.2.3 员工资料模块 25

6.2.4 人事管理模块 26

6.2.5 薪资管理模块 27

6.2.6 系统管理模块 30

7 系统测试 36

7.1 单元测试 36

7.2 集成测试 36

7.3 功能测试 37

7.4 测试总结 38

8 总结和展望 39

8.1 总结 39

8.2 展望 39

参考文献 40

致谢 41

1 引 言

1.1 课题背景

如今作为一个信息化的时代，不管是中小型还是大型企业都是非常需要人事管理系统的。这不仅可以提高效率节省人力，还可以规范流程以及只需要手机或者电脑就可以随时随地操作。目前，许多的中小型企业虽然有使用人事管理系统，但有些企业使用的可能是前几年的老系统，首先老的系统上在系统界面上在设计上布局以及颜色都是较为过时，以及在企业发展的过程中，系统的功能已并不能完全满足企业的要求了，导致企业在这方面存在很多不便，从而会给企业带来巨大的经济损失，进而会影响了企业的效率和经济效益。总体来说就是这种管理方式缺乏规范化、系统化的管理手段，计算机资源没有得到实际应用。因此，开发一个有利于企业对员工管理的系统至关重要。

1.2 国内研究情况

经过计算机信息化深入我们生活的方方面面，企业里的人事管理系统也自然得到了很好的发展，涌现了大量的，优秀的人事管理系统，充分体现了我们国家人事管理行业的智能化、信息化。主要国内人事管理信息系统的开发技术非常成熟，主要原因在于两方面，一是计算机技术的发展使得更容易实现数字化人事管理，二是越来越多企业发展壮大，传统的人事管理系统已不能满足企业的需要。现国内出现了许多提供定制化的人力资源系统解决方案的企业，用户企业包括大规模企业、中小型企业以及政府事业单位。还有一些提供SaaS服务企业，例如钉钉、企业微信等，它的好处是由SaaS提供商负责全面应用的维护，降低了使用和维护的成本，企业只需要通过网络注册账号，做一些简单的配置即可使用。

1.3 国外研究情况

国外的人事管理系统比国内出现要早，该系统本身与其相应的计算机信息应用技术均非常成熟。国外公司的人事部或劳工部一般均建立了计算机信息系统，负责公司内部的人力资源或劳工管理。几乎所有发达国家均建立了自己的人力资源管理信息系统，这个是全国性的，覆盖了全国的人力资源管理。在一些发达国家的企业，非常注重自身人力资源管理与决策支持系统的开发应用。尤其在一些跨国公司，更是不惜巨大投资开发相应的人力资源管理与决策支持系统。人力资源管理与决策支持系统在跨国公司中，既不是一个普通的人事部，也不是一个福利部门，而是围绕提高工作效能而设立的由多种信息、数据和活动形成的一个开放性系统。

1.4 课题研究目标及意义

如今计算机技术发展迅速，可以利用计算机来完成大量的基础数据的收集以及处理这些数据。其中，人事管理系统的开发，对公司管理员工信息具有重大的意义。对于今天的企业来说，人才是企业发展的生命线。 关注企业人才，就像关注企业发展一样。 高效的人事管理系统可以简化工作流程，提高工作效率，使企业具有更强的凝聚力、组织性和竞争力。

高质量的人事管理系统不仅能够简化系统内不同角色员工的工作流程，还能满足不同的员工对不同功能的需求，这不仅能避免由于人工手动操作导致的工作中的错误，还能使员工的工作变得更加简单，进而提高了员工的工作效率，促进人事单位的不断发展。所以，一套好的人事管理系统可以提高企业人事工作的办公效率、推动人事管理的科学化、系统化、高效化、自动化。

2 相关框架和技术简介

2.1 Spring框架

Spring是一个开源的轻量级控制反转(IoC)和面向切面(AOP)的容器框架，Spring结构图如2.1所示。

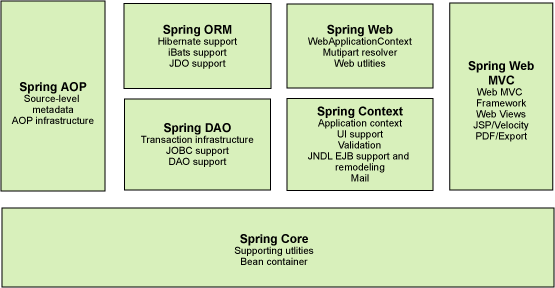


图2.1 Spring结构图

上图包含了Spring框架的所有模块，这些模块可以满足一切企业级应用开发的需求。Spring框架具有几个特性，非侵入式、控制翻转、依赖注入、面向切面编程、容器、组件化和一站式等。与之前程序员自己创建实例非常不同的是，Spring是通过IoC（控制反转）的方式，将应用程序依赖的对象注入bean容器，由bean容器把控创建对象。

2.2 SpringBoot框架

Spring Boot是在Spring的基础上再整合配置，简化Spring搭建时所需的大量配置，同时内嵌了Tomcat容器，配合Restful风格接口，实现了一键启动就能本地服务器访问。我们不再需要创建servlet、通过response将数据以json形式返回、配置web.xml、发布项目到Tomcat、启动Tomcat等这些操作。可以说，Springboot是Spring框架的框架，它将重复繁琐的配置在内部帮我们实现，使得程序员更多地关注业务代码的实现，从而最大程度地提高开发者的开发效率，简化项目的开发过程。

2.3 Mybatis框架

MyBatis是一款优秀的持久层框架，它和Hibernate都属于ORM框架，但具体来说，Hibernate是一个完全的ORM框架，而MyBatis是一个不完全的ORM框架，它内部封装了JDBC，开发时只需要关注SQL语句本身，不需要花费精力去处理加载驱动、创建连接、创建statement等繁杂的过程。它支持定制化 SQL、存储过程以及高级映射。MyBatis避免了几乎所有的JDBC代码和手动设置参数以及获取结果集。MyBatis可以使用简单的XML或注解来配置和映射原生信息，将接口和Java的 POJO(Plain Old Java Objects,普通的Java对象)映射成数据库中的记录。

2.4 Vue框架

Vue是一套用于构建用户界面的渐进式框架，是三大主流框架之一：Angular（庞大），React（精通移动端），Vue（吸取前两者优势，轻量级）。遵循先进的MVVM模式，即Model-View-ViewModel模式，其核心是提供对View和ViewModel的双向数据绑定，这使得ViewModel的状态改变可以自动传递给View，MVVM模式图如图2.2所示。因为Vue的双向数据绑定特性以及技术的成形，实现了项目的热加载，改完页面代码能立即在浏览器方面显示效果，提高开发效率。

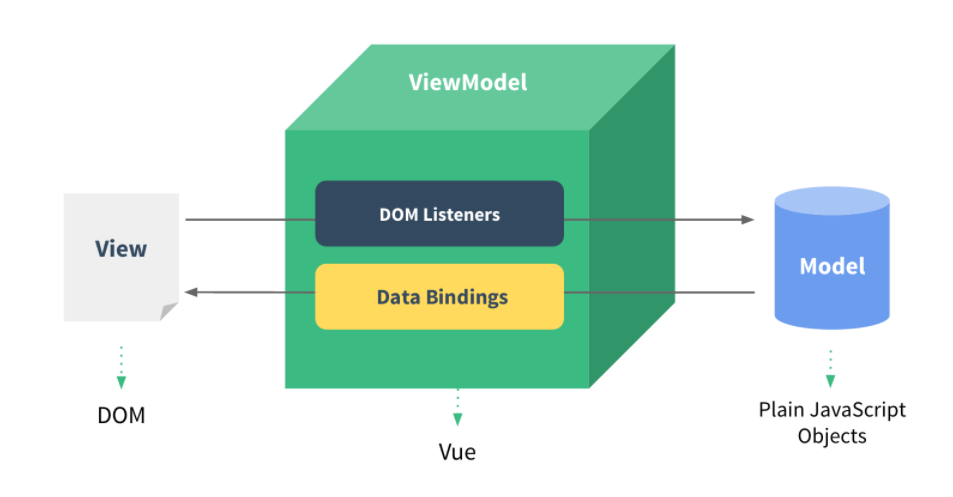


图2.2 MVVM模式图

3 系统分析

3.1 可行性分析

3.1.1 技术可行性

本系统开发使用目前流行的SpringBoot+Vue前后端分离的技术为基础，后端SpringBoot框架内嵌了Web服务容器Tomcat，以及使用Maven进行构建、依赖管理。只需要在pom.xml中添加依赖，由程序自动加载依赖jar包等配置文件，简化了Spring众多框架中所需的大量且繁琐的配置。数据库使用的MySQL5.7版本，前端使用Vue框架为基础，整合Vuex，axios等，实现前端单页面富应用的页面开发。能够把页面抽象成多个相对独立的模块为组件，实现代码重用，提高开发效率和代码质量，使代码易于维护。

综合上述几点，本人事管理系统是具有技术可行性的。

3.1.2 操作可行性

本系统的操作简单，使用者只需要在浏览器访问本人事管理系统，界面设计简洁明了，功能模块划分明确规范，不需要经过任何软件使用培训都能轻松操作。所以说本系统具有操作可行性。

3.1.3 经济可行性

本人事管理系统使用的技术框架例如SpringBoot、MySQL等都是开源的。对于传统的人工方式管理人事档案，不仅工作繁琐而且需要大量的人力，这样做的结果会造成记录出错率高、效率低下以及增加企业的运营成本，从而阻碍企业的发展。采用本人事管理系统的成本远低于人工管理的成本，只需要开发时候的购买服务器的费用，是一个以低成本就能换取高收益的项目，为企业发挥了巨大的作用，所以本系统具有经济可行性。

3.2 需求分析

3.2.1 功能性需求分析

通过对企业人事部门的调查和研究，根据他们的工作需求，把系统功能分为四个模块。

（1）员工资料模块：主要是管理企业员工的基本信息。在系统中，增加和删除员工资料信息只能由系统管理员来操作，其他权限的用户只能在这个模块中对员工基本资料的查询和更新。

（2）人事管理模块：主要包括员工奖惩和员工培训。管理员可以输入员工的奖惩情况和原因，以及员工的培训日期和内容。 同时，也可以让员工了解自己的奖惩情况。

（3）薪资管理模块：主要包括工资套账管理和员工套账设置。管理员可以添加工资套账和设置不同部门人员的工资套账内容的金额，以及可以修改员工所属部门的工资套账。

（4）系统管理模块：主要包括基础信息设置和操作员管理。基础信息设置里面有职位管理、职称管理、部门管理、权限组等。在部门管理里，管理员可以添加部门和删除部门，通过划分不同的部门可以搭建企业的组织架构。职位管理可以添加不同的职位，选择是否启用和删除职位。职称管理是可以添加不同的职称和选择该职称等级，选择是否启用和删除职称。权限组可以添加角色以及分配角色可访问的资源。

3.2.2 非功能性需求分析

软件产品非功能性定义不仅决定产品的质量，还在很大程度上影响产品的功能需求的定义。在系统完成基本用户业务需求而必须具有的功能以外，还需要非功能性的需求分析，如时间性，稳定性，易用性，安全性等。

（1）时间性：系统需要考虑企业信息变动频繁的情况，为了让员工信息变动后能及时地查看最新的信息，实现信息同步。因此本系统采用了前后端分离架构。只需要在浏览器刷新即可获取到服务器端的最新信息。

（2）稳定性：系统需具有持续正常工作的能力，能正常对数据进行读写等操作。

（3）易用性：系统的操作界面要求简介，选择使用户视觉上舒服的色彩搭配。各种功能操作命名规范，便于用户找到他们期望进行的操作。用户不需要专门学习操作该系统也能很快上手，在系统熟练使用后可以更快的进行各项操作。

（4）安全性：人事信息是企业非常重要且非常私密的资料，关乎到企业的利益和员工的隐私，所以系统的信息安全问题是非常重要的环节。因此系统需要设置登录权限，管理权限，不同权限的使用者只能使用被授予该层级的权限。

3.2.3 系统用例图

本系统的用户角色有，系统管理员、部门领导、人事领导、人事专员、普通员工。系统根据用户需要分为员工资料、人事管理、薪资管理、系统管理四大模块，由于用户权限不一样，所以能操作的模块也会不一样。通过系统用例图和各个模块管理用例图，可以了解各个功能之间的关系。系统用例图如图3.1所示。

人事领导

人事专员

普通员工

系统管理员

部门领导

图3.1系统用例图

人事管理模块包括员工奖惩和员工培训。普通职员只能查询相关信息，而不能具有修改和添加的权限。人事管理模块用例图如图3.2所示。

普通员工

人事专员

人事领导

部门领导

系统管理员

图3.2人事管理用例图

查询薪资、薪资管理、薪资核算、录入工资等都是薪资管理模块的子功能，同人事管理模块一致，普通职员只能对薪资进行查询，系统管理员具有添加、修改等所有的操作权限，部门领导、人事领导具有薪资管理的权限。薪资管理用例图如图3.3所示。

普通员工

人事领导

人事专员

部门领导

系统管理员

图3.3 薪资管理用例图

4 人事管理系统设计

4.1 系统设计原则

4.1.1 系统设计目标

本系统的设计目标在于研发一款满足企业的信息化需求的人事管理系统。方便企业人事管理部门的工作，以便于企业的人力资源部门能够全面、及时、正确的掌握企业员工的各类信息，提高其服务质量，改善其办公效率。使用本系统时，人力资源部工作人员可以对企业员工的基本信息、员工培训、员工奖惩、员工工资等进行管理，系统管理员可以添加、删除、修改、查询等系统管理操作上的所有信息。基于现代化的信息管理模式，对企业人事管理工作提供了高效、便利、安全可靠的服务。

4.1.2 系统设计原则

对于现今高度信息化的社会来说，企业的用户量、安全性、稳定性以及多种的特色功能，企业需要一个功能强大、安全性高、方便操作、界面美观、用户体验更好的人事管理系统。基于SpringBoot+Vue的人事管理系统以Web浏览器和Tomcat服务器组成B/S架构。本系统是一个具有用户群广、安全稳定、可扩展性和可维护性等特性的人事管理系统。结合企业人事部门现有的需求，在系统设计的过程中将遵循以下原则：

（1）可用性

因为客户和用户都很注重界面设计，用户感受到最直观的就是操作界面是否美观，界面友好、标准，操作流畅，有良好的用户体验。良好的网站设计，用户操作非常流畅，流程很容易让用户理解。简介明了，没有乱七八糟的东西干扰用户。数据的呈现方式也是经过整合的，不是所有的数据都呈现给用户，而是呈现对用户有用的数据。

（2）实用性

人事管理系统是应用于企业的系统，每个企业必备并且无时无刻都在使用的系统。其宗旨是满足企业人事管理的业务需求，保证人事部门人员管理的便捷高效，促进企业的发展壮大。

（3）先进性

在实用性的基础上，充分应用先进和成熟的技术，满足系统的要求，把科学的管理操作流程和先进的技术紧密结合起来，使系统具有较高性能，符合当今技术发展方向，确保系统具有较强的生命力。

（4）可靠性

在保证各项业务可靠性的基础上，需要一个健壮的架构来保证系统在出现软硬件错误的情况下仍能流畅运行。在系统可靠性设计上，采用统一的异常处理、错误隔离、告警、容错和恢复机制，可以在不损坏数据的情况下尽快恢复正常运行。

（5）安全性

系统使用了Spring Security安全管理框架对用户的登录进行了认证和授权。不同的用户拥有操作系统的权限也是不同的。所以系统管理员可以分配不同的权限以达到管理系统数据的安全性和保密性。

（6）可扩展性

应该全面考虑企业业务的扩展需求，各功能模块都应遵循完整统一的规范，系统在设计过程中，应具备良好的输入输出接口，可为各种扩展业务提供接口。

（7）可维护性

系统的编码的过程中，采用了MVC系统架构，清晰的包、类和xml配置文件的命名。项目代码具有良好的编码风格，采用了单一职责原则、接口隔离原则和开闭原则等设计模式原则，将对象解耦，提高内聚性，降低了类的复杂度，从而提高了可读性和可维护性。

4.2 系统架构设计

本系统是基于B/S架构的开发，在B/S架构中，系统标准的三层架构包括：表现层、业务层、持久层。

（1）表现层

表现层主要是展示界面，用于处理用户请求，并返回响应结果。通常客户端使用HTTP协议请求Web层，Web层需要接收HTTP请求，完成HTTP响应。表现层包括展示层和控制层：控制层负责接收请求，展示层负责结果的展示。表现层依赖业务层，接收到客户端请求一般会调用业务层进行业务处理，并将处理结果响应给客户端。

（2）业务层

业务层是处理系统大部分业务逻辑的层面。和我们开发项目的需求息息相关。Web层依赖业务层，但业务层不依赖Web层。业务层在业务处理时可能会依赖持久层，如果要对数据持久化需要保证事务一致性。

（3）持久层

持久层是直接跟数据库交互的层面，负责数据持久化，包括数据层即数据库和数据访问层，数据库是数据进行持久化的载体，数据访问层是业务层和持久层交互的接口，业务层需要通过数据访问层将数据持久化到数据库中。

4.3 系统功能模块设计

4.3.1 系统设计目标

本人事管理系统中的主要功能包括员工资料、人事管理、薪资管理和系统管理等功能。系统中的具体功能结构如图4.1所示。

人事管理系统

员工资料

薪资管理

人事管理

系统管理

基本资料

基本信息设置

员工套账设置

工资套账管理

员工奖惩

员工培训

更多资料

操作员管理

图4.1 系统功能模块结构图

4.3.2 用户登录流程

基于SpringBoot+Vue的人事管理系统用户登录流程图如图4.2所示，其登录序列图设计如图4.3所示。

结束

输入用户名、密码、验证码

输入是否正确

进入主界面

开始

图4.2 用户登录流程图

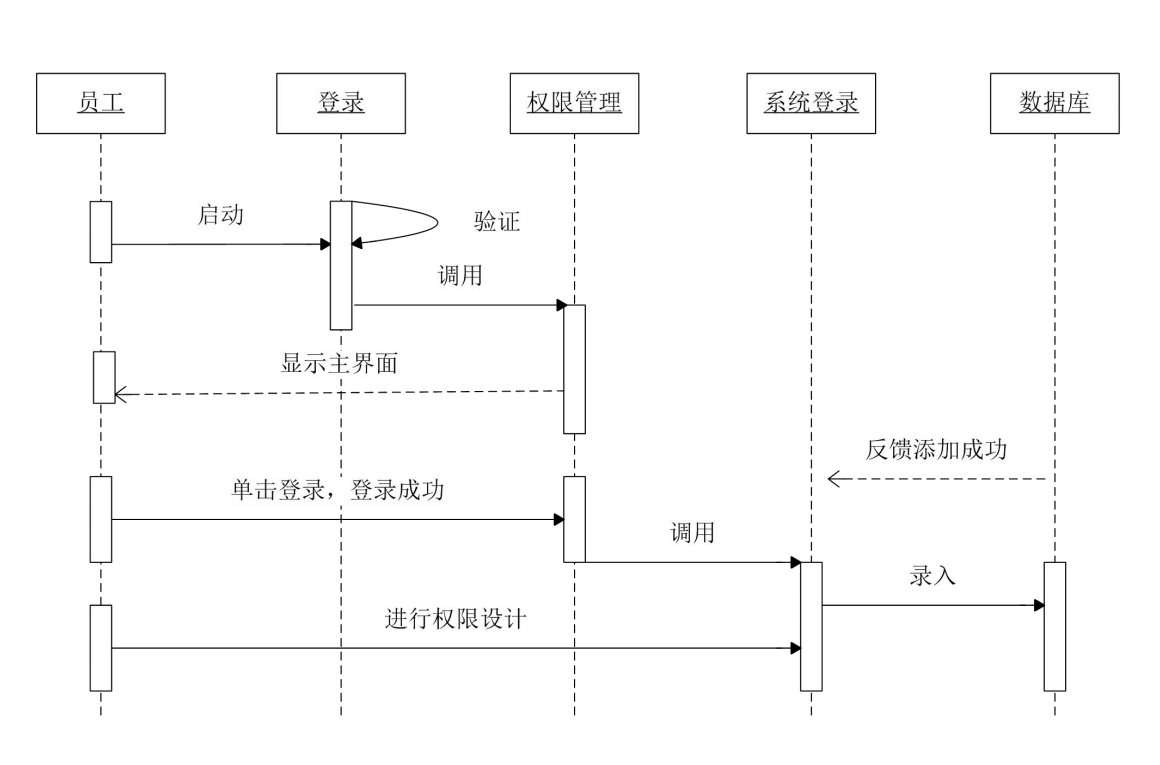


图4.3 员工登录序列图

4.3.3 人事管理模块

人事管理模块里包含员工培训和员工奖惩两个功能，所有操作用户可以查询人事管理模块里的信息，但要进行添加、删除和修改等功能必须是系统管理员或者获得该操作权限的用户才能进行操作。人事管理模块流程图如图4.4所示。

开始

人事管理界面

添加、修改和删除操作

显示查询员工人事信息

查询员工人事信息

没有操作权限

保存操作信息

是否具有操作权限

结束

Y

N

图4.4人事管理流程图

* + 1. 部门管理模块

部门管理是基础信息设置里的功能，以系统管理员的身份登录本系统，可以对部门信息进行管理。系统管理员可以对查询、添加和修改部门信息，每个部门可以创建自己的子部门，若部门下有子部门则不能删除该部门，应先删除父部门才能删除子部门。部门管理流程图如图4.5所示。

开始

部门管理界面

添加部门

显示查询部门结果

查询部门

保存部门信息

是否具有操作权限

结束

输入查询部门ID

输入部门详细信息

Y

N

图4.5 部门管理流程图

5 数据库设计

人事管理系统是企业中最常用的系统之一，员工日常的打卡、住址电话、薪资发放明细等都存储在系统中，数据库存储的员工信息以及数据是企业机密信息，绝对不能泄露，否则会直接影响到员工的隐私和企业的商业发展，数据库的安全性至关重要。因此数据库设计的一定要具有安全性，稳定性以及效率性，数据库的设计好坏，往往能直接影响系统的处理问题的能力和运行效率。

项目所涉及的系统数据库的表主要包括：基本资料表、角色表、培训表、薪资异动表、岗位异动表。

5.1 数据库逻辑结构设计

经过前两章系统分析和系统设计的需求分析后，我们可以画出E-R图模型，进行逻辑设计。因为概念模型能让我们更清楚实体间的联系，这也是在进行完需求分析后，设计数据库最先进行的工作。实体设置相应的属性，各实体间进行一对一、一对多或多对多的关联，用树状结构连接表示实体与实体间的联系。

本系统数据的实体对象主要包括员工、部门、员工培训、职位、职称等，系统之间的E-R图如图5.1所示。

员工

培训

部门

职称

职位

民族

员工

员工

岗位异动

包含

包含

包含

包含

包含

包含

包含

包含



图5.1 系统E-R图

员工

图5.1 员工属性图

（2）员工培训实体主要包括姓名、参加对象、地点、主讲老师和内容等属性，员工培训属性如图5.3所示。

员工培训

图5.2 员工培训属性图

（3）员工考勤实体主要包括姓名、正常打卡、工作天数、请假天数和迟到天数等属性，员工考勤属性如图5.4所示。

员工考勤

图5.3 员工考评属性图

（4）员工岗位异动调薪实体主要包括异动性质、生效时间、调前工资和调后工资等属性，员工岗位异动调薪属性如图5.5所示。

员工岗位异动调薪

图5.4 员工调薪属性表

5.2 数据库物理设计

通过数据库逻辑设计确定了E-R图后，由于本系统的功能尚有不足，以及规模较小，所以决定选择开源免费的MySQL数据库进行物理设计。在设计表以及字段的时候，严格按照企业级代码规范设计，每个表和字段的命名都使用有实际意义的命名，能让别人清楚该表该字段的用途。定义字段类型和长度时，不应盲目选择默认提供的，尽量选择合适的字段类型和字段长度，可以节省数据库的存储空间，降低数据库的压力。

在通过以上设计策略后，设计出了一个较为功能齐全、存储时间和空间合理的数据库。

系统所有模块功能围绕下面9个表展开。

员工信息表如表5.1所示。

表5.1 员工信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名 称** | **类 型** | **长 度** | **非 空** | **主 键** | **注 释** |
| id | int | 10 | 是 | 是 | 主键id |
| name | varcahr | 30 | 是 | 否 | 姓名 |
| sex | char | 1 | 是 | 否 | 性别 |
| sn | varchar | 100 | 是 | 否 | 工号 |
| phone | varchar | 20 | 是 | 否 | 电话 |
| position | varchar | 30 | 是 | 否 | 职位 |
| department | varchar | 30 | 是 | 否 | 部门 |
| email | varchar | 30 | 否 | 否 | 电子邮箱 |

员工培训表如表5.2所示

表5.2 员工培训表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名 称** | **类 型** | **长 度** | **非 空** | **主 键** | **注 释** |
| id | int | 10 | 是 | 是 | 主键id |
| name | varcahr | 30 | 是 | 否 | 姓名 |
| participants | varchar | 30 | 是 | 否 | 参加对象 |
| theme | varchar | 100 | 是 | 否 | 主题 |
| lecturer | varchar | 20 | 是 | 否 | 主讲老师 |
| content | varchar | 255 | 是 | 否 | 内容 |
| place | varchar | 30 | 是 | 否 | 地点 |
| share\_time | datetime | 0 | 否 | 否 | 分享时间 |

员工考勤表如表5.3所示

表5.3 员工考勤表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名 称** | **类 型** | **长 度** | **非 空** | **主 键** | **注 释** |
| id | int | 10 | 是 | 是 | 主键id |
| name | varcahr | 30 | 是 | 否 | 姓名 |
| work\_days | int | 4 | 是 | 否 | 工作天数 |
| normal\_clockin | int | 4 | 是 | 否 | 正常打卡 |
| late\_days | int | 4 | 是 | 否 | 迟到天数 |
| absent\_days | int | 4 | 是 | 否 | 旷工天数 |
| early\_days | int | 4 | 是 | 否 | 早退天数 |
| leave\_nums | int | 4 | 是 | 否 | 请假次数 |
| makeup\_nums | int | 4 | 是 | 否 | 补卡次数 |

员工岗位异动调薪表如表5.4所示

表5.4 员工岗位异动调薪表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **名 称** | | **类 型** | **长 度** | | **非 空** | | **主 键** | **注 释** |
| id | int | | 10 | | 是 | 是 | | 主键id |
| name | varcahr | | 30 | | 是 | 否 | | 姓名 |
| id\_card | char | | 18 | | 是 | 否 | | 身份证号码 |
| eid | varchar | | | 30 | 是 | 否 | | 员工编号 |
| change\_property | varchar | | | 30 | 是 | 否 | | 异动性质 |
| origin\_post | varchar | | | 30 | 是 | 否 | | 原岗位名称 |
| new\_post | varchar | | | 30 | 是 | 否 | | 现岗位名称 |
| adjust\_reason | varchar | | | 50 | 是 | 否 | | 调薪原因 |
| before\_salary | decimal | | | 10 | 是 | 否 | | 调前工资 |
| after\_salary | decimal | | | 10 | 是 | 否 | | 调后工资 |
| effect\_time | datetime | | | 0 | 是 | 否 | | 生效时间 |

会议管理表如表5.5所示

表5.5 会议管理表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名 称** | **类 型** | **长 度** | **非 空** | **主 键** | **注 释** |
| id | int | 10 | 是 | 是 | 主键id |
| title | varcahr | 30 | 是 | 否 | 名称 |
| meeting\_place | varchar | 30 | 是 | 否 | 会议地点 |
| convener | varchar | 30 | 是 | 否 | 召集人 |
| contact | varchar | 30 | 是 | 否 | 联系人 |
| start\_time | datetime | 0 | 是 | 否 | 开始时间 |
| end\_time | datetime | 0 | 是 | 否 | 结束时间 |
| status | char | 1 | 是 | 否 | 会议状态 |

会议室表如表5.6所示

表5.6 会议室表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名 称** | **类 型** | **长 度** | **非 空** | **主 键** | **注 释** |
| id | int | 10 | 是 | 是 | 主键id |
| name | varchar | 30 | 是 | 否 | 名称 |
| place | varchar | 30 | 是 | 否 | 位置 |
| capacity | int | 5 | 是 | 否 | 容纳人数 |
| scale | varchar | 10 | 是 | 否 | 规模 |

工资表如表5.7所示

表5.7 工资表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名 称** | **类 型** | **长 度** | **非 空** | **主 键** | **注 释** |
| id | int | 10 | 是 | 是 | 主键id |
| date | date | 0 | 是 | 否 | 日期 |
| name | varchar | 30 | 是 | 否 | 姓名 |
| base\_pay | decimal | 10 | 是 | 否 | 基本工资 |
| deduction | decimal | 10 | 是 | 否 | 缺勤扣款 |
| insurance | decimal | 10 | 是 | 否 | 社保 |
| acc\_fund | decimal | 10 | 是 | 否 | 公积金 |
| per\_tax | decimal | 10 | 是 | 否 | 个税 |
| net\_salary | decimal | 10 | 是 | 否 | 实发工资 |

6 人事管理系统的实现

通过前面章节的系统分析、系统设计以及数据库设计，本章是论述该人事管理系统的实现，主要介绍了实现系统开发的技术选型、项目环境和项目结构，各功能的模块的具体实现及界面设计展示。

6.1 系统开发及运行环境

本人事管理系统后台使用SpringBoot框架，整合Mybatis进行开发，前端页面使用Vue.js框架进行数据模型的绑定与展示，技术选型如表6.1所示。

表6.1 技术选型

|  |  |  |
| --- | --- | --- |
| **技 术** | **版 本** | **说 明** |
| Spring Boot | 2.5.5 | 容器+MVC框架 |
| Spring Security | 5.5.2 | 认证和授权框架 |
| Mybatis | 3.5.7 | ORM框架 |
| MybatisGenerator | 1.4.0 | 数据层代码生成 |
| Druid | 1.2.6 | 数据库连接池 |
| Lombok | 1.8.20 | 简化对象封装工具 |
| Vue.js | 2.6.12 | 前端框架 |
| Vue-router | 3.4.9 | 路由框架 |
| Vuex | 3.6.0 | 全局状态管理框架 |
| Element-UI | 2.15.6 | 前端UI框架 |
| Axios | 0.24.0 | 前端HTTP框架 |
| Js-cookie | 3.0.1 | Cookie管理工具 |

6.2 系统实现

6.2.1 项目目录结构

本系统后端项目结构如下所示。

vhrserver

com

yangpj

├── config – 配置类

├── controller – 前端控制器层

├── mapper – 数据接口访问层

├── model – 数据实体类层

├── service – 数据服务接口层

└── utils – 工具库类

本系统前端项目结构如下所示。

src -- 源码目录

├── assets -- 静态图片资源文件

├── components -- 通用组件封装

├── router -- vue-router路由配置

├── store -- vuex的状态管理

├── styles -- 全局css样式

├── utils -- 工具类

└── views -- 前端页面

├── emp -- 员工资料模块页面

├── error -- 错误提示页面

├── per – 人事管理模块页面

├── sal – 薪资管理模块页面

└── sys – 系统管理模块页面

6.2.2 系统登录模块

因为系统是本地部署的，所以将前端和后端服务运行起来后，首先的会是进入登录界面，界面是一个用户输入用户名和密码的表单，用户可能登录不同的角色获取不同的登录权限。界面登录界面如下图6.1所示。



图6.1 系统登录界面

本人事管理系统是通过Spring Security安全框架进行登录认证和角色授权的，下面是SpringSecurity执行原理过程。创建一个配置类SecurityConfig继承了抽象类WebSecurityConfigurerAdapter,重写configure(HttpSecurity http)方法，配置需要认证的登录url，以及提交表单的url，这里除了登录url不需要认证之外，其他的url都需要认证才能访问。

SecurityConfig的configure方法代码如下：

从上面代码中可以看出，当用户发起登录请求时，loginPage指定跳转到/login，进入/login接口后，有两种情况，一种是.successHandler登录成功回调，另一种是.failure登录失败回调。在数据源中获取到用户名和密码组成的token，用它和数据库的记录进行比较，如果匹配成功，则获取该用户的角色和权限，并跳转到主界面，加载出相应的菜单，同时把该用户的信息和权限列表存储在浏览器的session中，以便后续的查询对用户进行权限的检查。登录过程结束。

6.2.3 员工资料模块

员工资料模块是系统中占据着十分重要的地位，存储着整个企业人员的基本资料，模块的功能包括对员工增加、编辑、删除和搜索，其中增加、删除和编辑都是对数据库服务器数据的更新操作。还有使用了Apache POI组件设计，使用Java程序创建、修改和显示MS Office文件，实现了文件导入和导出数据的功能。员工基本资料界面如图6.2所示。

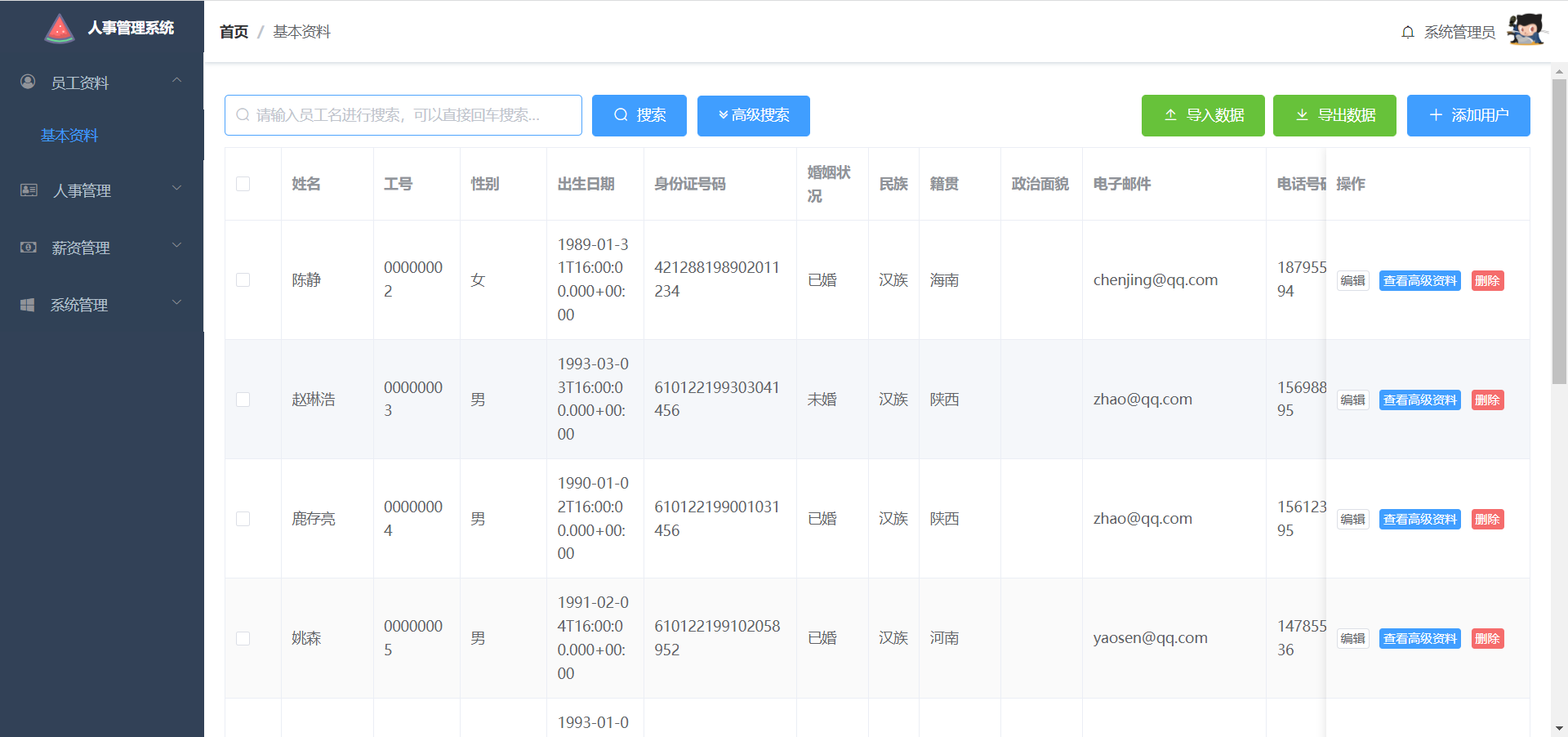


图6.2 员工基本资料

添加用户功能的核心代码如下所示。

添加用户界面分别如图6.3所示。



图6.3 员工资料-添加用户界面

6.2.4 人事管理模块

人事管理模块中包括员工奖惩和员工培训，主要是记录员工的培训内容、培训日期和员工奖惩原因和奖惩日期等。实现添加、编辑和删除操作。这两个界面类似，下面只展示图6.4员工培训界面。



图6.4 员工培训界面

6.2.5 薪资管理模块

薪资管理包括工资套账管理和员工套账设置模块，是本人事管理系统的核心和重点模块。因为每个公司都有不同类型的工作人员，企业中都需要至少3种不同的工资套账。

SalSob.vue进行展示所有信息，工资套账管理界面如下图6.5所示。



图6.5 工资套账管理界面

可以添加工资套账，添加用到了Steps步骤条，添加工资套账界面如下图6.6所示。



图6.6 添加工资套账界面

当出现多级表头时，可以利用el-table-culumn的嵌套从而实现多级表头。工资套账管理核心代码如下。

可以对员工的工资套账进行修改，当鼠标移动到套账标签时，会自动显示该套账详情信息，员工套账设置界面如下图6.7所示。

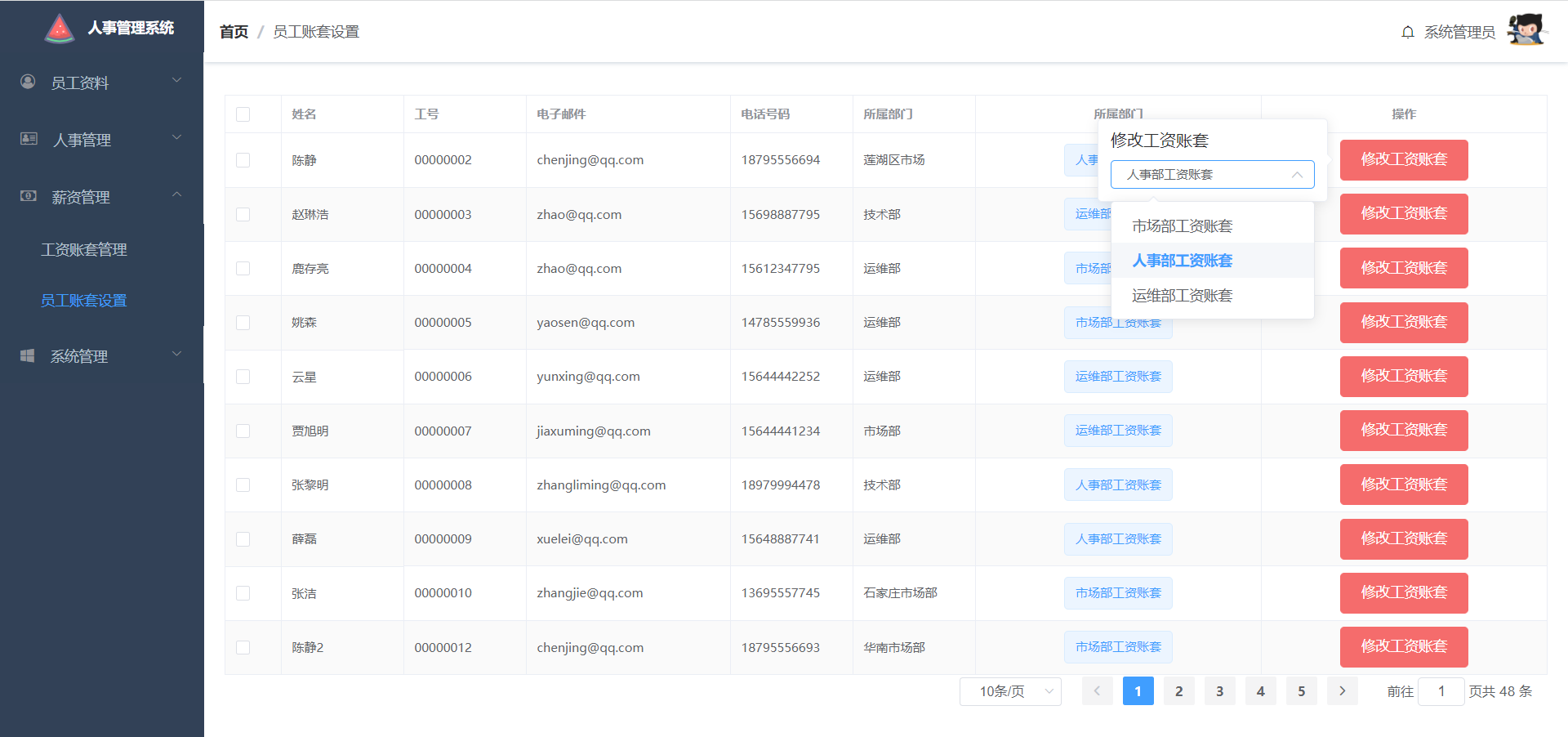


图6.7 员工套账设置界面

6.2.6 系统管理模块

系统管理是整个人事管理系统必不可缺的部分，系统管理有基本信息设置模块和操作员管理模块。其中基本信息设置包括了部门管理、职位管理、职称管理和权限组等重要功能。

部门管理中可以添加部门以及子部门，部门内可以添加多个子部门，或者兄弟部门，子部门还可以添加其自身的子部门，这样可以形成一个部门职责、层级分明的企业组织架构但要删除部门必须要把子部门都删除了才可以删除父部门。部门管理界面如下图6.8所示。

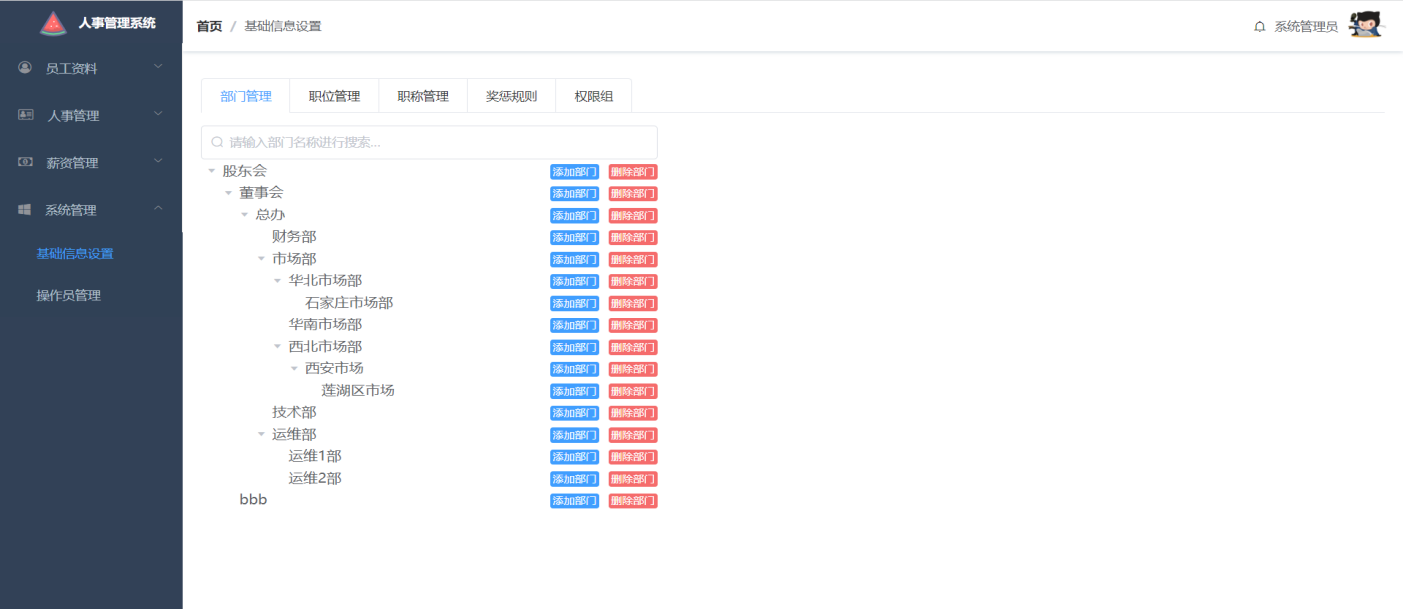


图6.8 部门管理界面图

职位管理和职称管理的界面和功能类似，都是可以对其进行添加、编辑和删除操作，编辑可以修改职位和职称，以及选择是否启用。

职位管理界面如下图6.9所示。

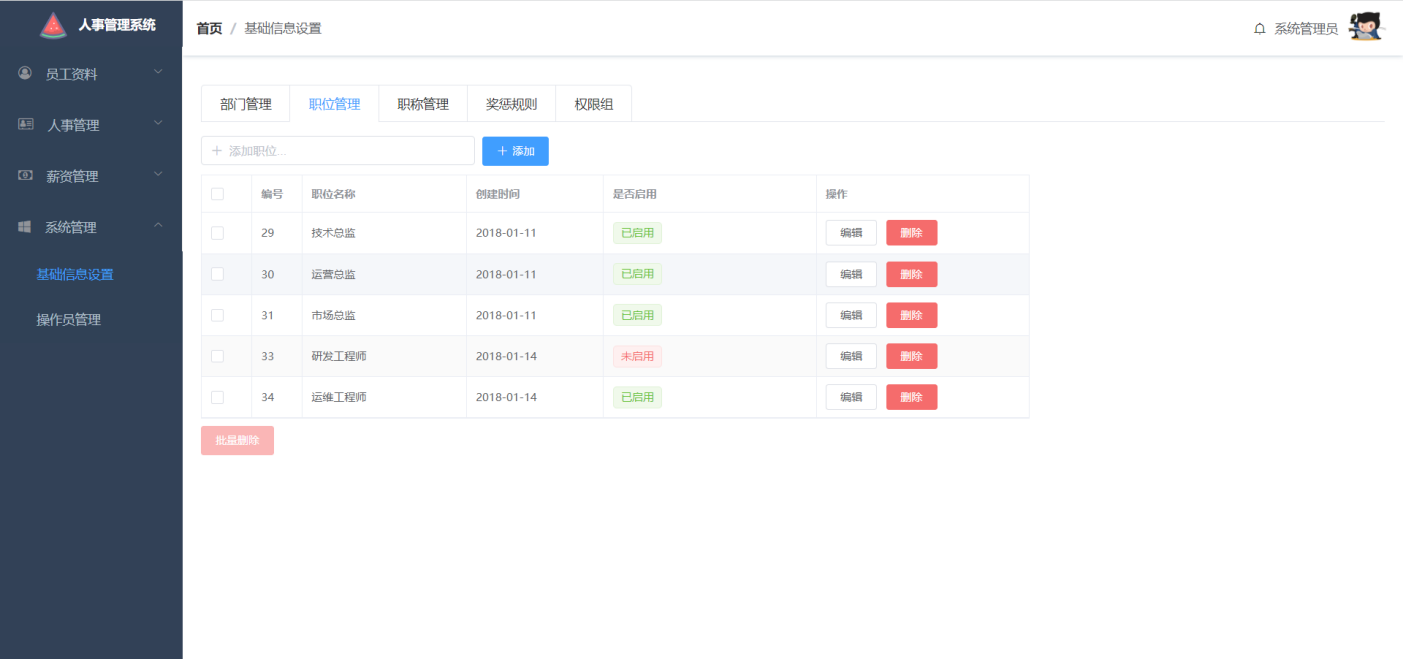


图6.9 职位管理界面图

职称管理界面如下图6.10所示。

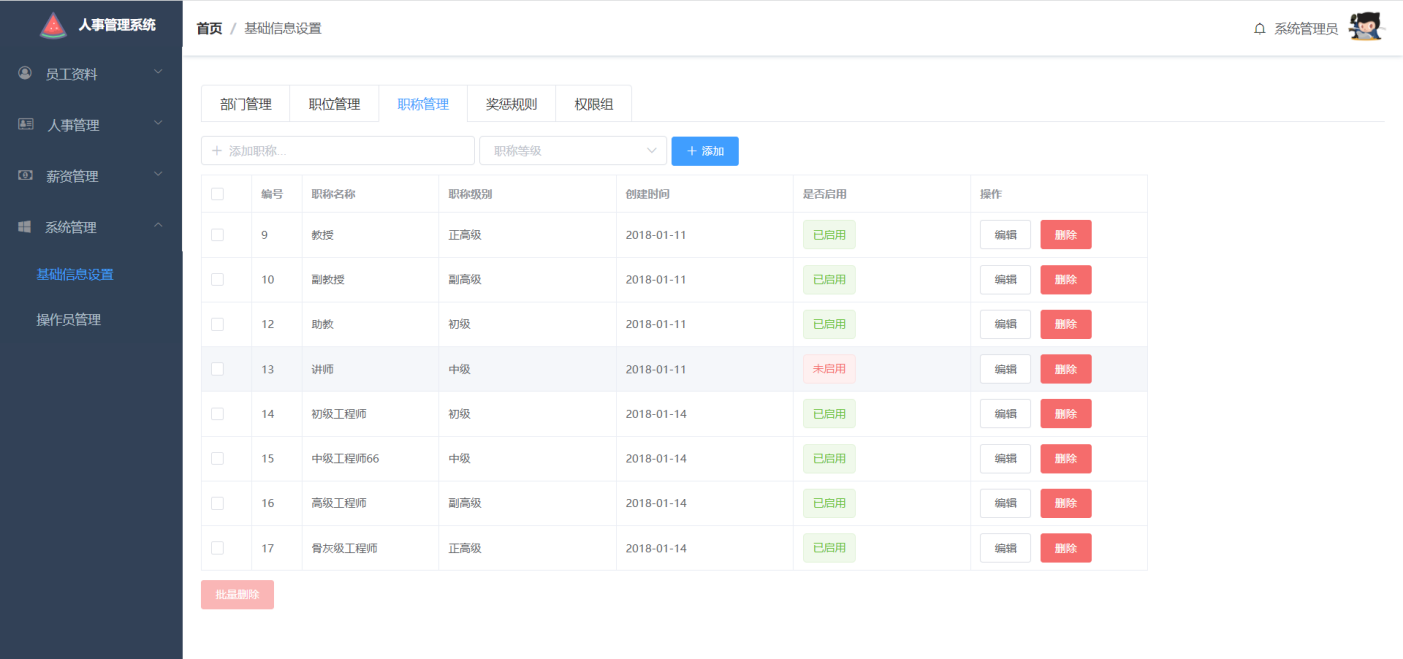


图6.10 职称管理界面图

权限组的角色展示采用了ElementUI中的Collapse 折叠面板，并且采用了手风琴模式，即一次只打开一个角色，权限组界面如下图6.11所示。

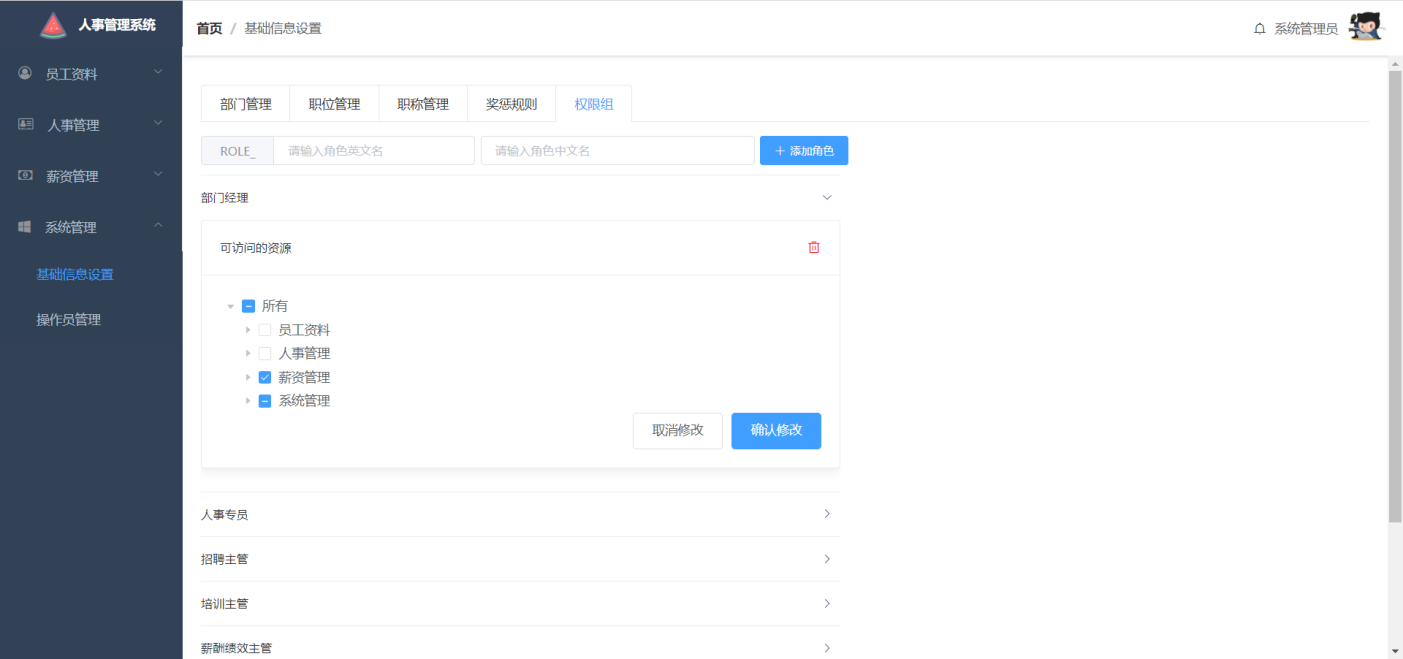


图6.11 权限组界面图

角色中资源的展示则采用了ElementUI中的树形控件，管理员可以直接直接点击勾选，然后点击修改按钮，进行资源的分配。

权限组核心代码如下：

操作员管理是给不同用户分配不同权限的，因为一个人事管理系统不可能所以用户都具有相同的权限，必须有明确的权限和职责划分，每个角色下的用户登录系统后完成相应的任务，实现相应的功能，不能出现越权操作的情况，操作员管理界面如图6.12所示。

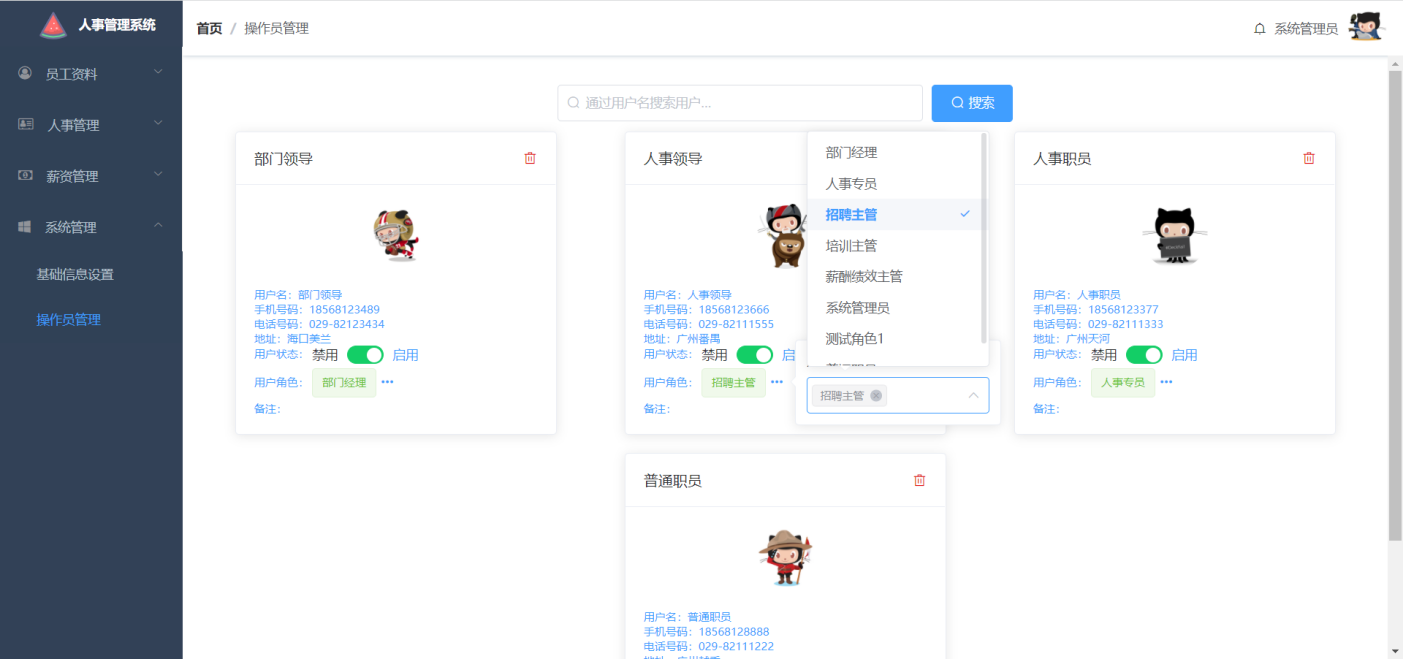


图6.12 操作员管理界面图

角色展示使用了ElementUI的Tag**标签**来实现，角色后面有一个more按钮，点击之后是一个Popover**弹出框，**Popover**弹出框**的里边是一个Select**选择器**，多选的，可以进行角色的分配。

操作员管理的核心代码如下。

7 系统测试

对于本人事管理系统来说，系统必须安全、可靠，能够真正满足企业内部不同角色人员的需求。每一个软件研发过程中存在错误是在所难免的，为了减少错误造成的损失，保证系统研发的质量，因此，在系统真正投入使用之前必须对其进行测试。整个测试过程不仅仅是在开发完成之后才进行，而是渗透在系统的整个开发过程中，在开发完成一个模块后，也要对相应的模块进行测试，否则会因为在上一个模块的问题还没解决，影响开发下一个模块。

7.1 单元测试

单元测试又称模块测试、功能点测试等，检查各个模块是否正确实现规定的功能，接口是否正确，从而发现模块在编码中或算法中的错误。

本文在系统中对各模块进行单元测试时使用的测试方法是“白盒测试”，在各个模块设置测试用例，检查是否能达到预计的功能实现。如果能得到预计的输出结果，则初步表面该模块实现正常，待后续进一步的测试；如果没能得到预计的输出结果，则说明该代码仍存在语法或是逻辑错误，需要代码进行仔细检查，表7.1中列举了部分测试用例。

表7.1登录测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **流程测试** | **测试案例** | **案例预计结果** | **测试结果** |
| 用户登录 | 用户名：admin  密码：123。 | 成功登录。 | 测试通过 |
| 用户登录 | 用户名：depleader  密码：123。 | 成功登录。 | 测试通过 |
| 用户登录 | 用户名：hrleader  密码：123。 | 成功登录。 | 测试通过 |
| 用户登录 | 用户名：staff  密码：123。 | 成功登录。 | 测试通过 |
| 用户登录 | 用户未输入内容，击登录。 | 提示用户名不能为空，密码不能为空。 | 测试通过 |

7.2 集成测试

在进行详细的单元测试后，然后就是集成测试，将模块按照设计要求组装成一个完整的系统进行测试。因为虽然单个模块能正常工作，但不能保证各个模块连接起来也能正常工作。集成测试的方法主要有自底向上测试和自顶向下测试，自底向上是最常用的集成测试方法。

本系统进行的集成测试是自底向上的测试方法，首先按照系统设计明确出本人事系统的模块组成，按照优先级顺序组合成为子系统，测试子系统是否能正常工作，最终形成整个人事管理系统，测试系统能否达到前文中的需求分析的要求。

对系统进行集成测试的部分测试用例如表7.2所示。

|  |  |  |  |
| --- | --- | --- | --- |
| **流程测试** | **测试案例** | **案例预计结果** | **测试结果** |
| 登录模块 | 用户输入用户名和密码。 | 成功进入系统，并显示用户可以操作的所有功能。 | 测试通过 |
| 人事管理模块 | 单击“人事管理”菜单，添加人员。 | 显示人事管理模块的全部功能。 | 测试通过 |
| 薪资管理模块 | 单击“薪资管理”菜单。 | 显示薪资管理模块的全部功能。 | 测试通过 |
| 系统管理模块 | 单击“系统管理”模块，然后选择“部门管理”添加部门。 | 显示系统管理模块的全部功能。 | 测试通过 |

7.3 功能测试

功能测试主要采用黑盒测试，结合测试内容对功能进行测试，功能主要分为功能模块测试和业务流程测试。

对本系统各个功能模块的部分测试用例如表7.3所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **流程测试** | **测试案例** | **案例预计结果** | | **测试结果** |
| 系统管理员 | 以系统管理员的用户名和密码登录系统。 | | 显示系统全部功能，并可以操作所以功能模块。 | 测试通过 |
| 一般部门职员 | 以一般部门职员的用户名和密码登录系统。 | | 显示系统全部功能，但是只能对个人基本信息进行增、删、改操作，可以查询全部模块。 | 测试通过 |
| 人事部门职员 | 以人事部门职员的用户名和密码登录系统。 | | 显示系统全部功能，能够修改所有人员的人事信息，对本系统所有的基本功能具有修改权限，但须人事部门领导审批通过才能进行相关操作。 | 测试通过 |
| 人事部门领导 | 以人事部门领导的用户名和密码登录系统。 | | 显示系统全部功能，可以对系统进行增删改查操作，但是部分功能只能查看不能修改、删除。 | 测试通过 |
| 人事管理模块 | 以人事部门职员的用户名和密码登录系统。 | | 显示人事管理模块的全部功能，包括对员工的增删改查等。 | 测试通过 |
|  |  | |  | 续表7.3 |
| **流程测试** | **测试案例** | | **案例预计结果** | **测试结果** |
| 薪资管理模块 | 以人事部门职员的用户名和密码登录系统。 | | 显示薪资管理模块的全部功能，员工的基础薪资和福利的增删改查等操作。 | 测试通过 |
| 系统管理模块 | 以系统管理员的用户名和密码登录系统。 | | 显示系统管理的全部功能，对整个系统管理的增删改查等操作。 | 测试通过 |

7.4 测试总结

软件测试是软件开发过程中必不可少环节之一，通过对本人事管理系统进行测试，发现其运行良好，在员工资料、人事管理、薪资管理和系统管理等功能模块中运行正常，不仅实现了用户的需求，而且性能上也达到了一定的标准，方便、快捷，大大地提高了企业人事管理的效率，为企业减少了很多不必要的开支，达到了预期的目的。

8 总结和展望

8.1 总结

企业人事管理系统是现代企业管理工作中极其重要的一部分，一个高质量的人事管理系统不仅能够使企业内部工作井然有序，提高员工的工作效率，还可以使企业人事管理质量得到保证。本文根据现代企业对人事管理的需要，总结了许多企业人力资源管理的有效方法，开发了一套基于现代化的企业人事管理系统。在系统设计与开发过程中，采用了Spring Boot框架，Spring MVC作为后台的调度中心，前端使用Vue.js框架，数据库使用MySQL，最终实现了基于SpringBoot+Vue的人事管理系统。本系统实现的主要功能包括如下：

（1）实现了基本的用户管理和员工信息管理功能。本文设计的人事管理系统中的所有用户都是由管理员添加的，以员工编号作为用户名，并为新添加的用户设置初始密码，用户登录系统后可自行修改密码；管理员对员工信息具有增删改查功能，实现了基本的人员管理功能。

（2）根据文中设计的角色和权限，每个用户将属于不同的角色，每个角色拥有不同的权限集合，因此可以操作不同的功能模块。系统使用了Spring Security框架对登录和权限进行控制。不同的用户登录系统会显示不同的菜单和权限。

（3）人事管理模块实现了对员工奖惩和员工培训信息的增删改查功能。

（4）薪资管理模块实现了工资账套管理和员工账套设置等功能。

（5）系统管理实现了部门管理、职位管理、职称管理和权限组等功能，是整个系统最重要的模块，只有系统管理员才具有权限操作。

8.2 展望

本文在开发过程，由于自身技术原因和时间有限，很多功能还没有细化和完善，后续还需要不断的进行改进。

界面的设计方面，还不够吸引用户，需要优化界面设计，增强用户体验。人事管理系统是一个非常庞大的系统，涉及的功能、角色非常多，里面的关系也比较复杂。本文还无法开发出一个完善、通用性较强的人事管理系统，在以后的日子里，我还需要不断的学习，以开发出一个能在企业中真正发挥作用的人事管理系统。

参考文献

[1] 王佳越. 基于Spring Cloud的企业人事管理系统的设计与实现[D].吉林大学,2018.

[2] 文鹏. 基于SSM框架的企业人事管理系统的设计与实现[D].湖北工业大学,2018.

[3] 燕雷. 某企业人事管理系统的设计与实现[D].北京工业大学,2018.

[4] 刘丽媛,徐天姿.企业人事管理系统的设计与实现[J].文存阅刊,2018(01):179.

[5] 路瑜. 基于B/S模式的高校人事管理系统设计与开发[D].石家庄铁道大学,2017.

[6] 王壮. 基于Nodejs框架的企业人事管理系统的设计与实现[D].吉林大学,2017.

[7] 郭炎. 基于B/S的高校人事管理系统设计与实现[D].北京工业大学,2016.

[8] 唐卫.企业人力资源管理信息化建设研究[J].商业观察,2021(32):89-91.

[9] 杨丽娟.企业人力资源管理信息化现状与发展对策[J].技术与市场,2021,28(09):188+190.

[10] 肖英.基于B/S人力资源系统设计与实现[J].数字通信世界,2020(11):267-268.

[11] 周昱伟.计算机软件测试技术研究[J].电子技术与软件工程,2021(17):49-50.

[12] Chu Yichen. Construction of Student Personal Information Management System Relying Computer[J]. Journal of Physics: Conference Series,2021,1915(4).

[13] Herasymova Vitaliia,Sidorenko Valentina. Improvement of Personnel Management System as Condition of Enhancement of Competitiveness of the Enterprise[J]. Modern Economics,2019,14.

[14] [A modularized operator interface framework for Tokamak based on MVC design pattern](https://sso.gzlib.org.cn/interlibSSO/goto/75/+jmr9bmjh9mds/kcms/detail/detail.aspx?filename=SJES14061700190877&dbcode=SJES&dbname=scholar_journal_SJES)[J] . Xuan Yin,Wei Zheng,Ming Zhang,Jing Zhang,G. Zhuang,T. Ding.  Fusion Engineering and Design . 2014.

[15] [Teaching object‐oriented programming with AEIOU](https://kns.cnki.net/kcms/detail/detail.aspx?filename=SJWD14042400000230&dbcode=SJWD&dbname=scholar_journal_SJWD)[J] . Guillermo Licea,Reyes Juárez‐ Ramírez,Carelia Gaxiola,Leocundo Aguilar,Luis G. Martínez.  Comput Appl Eng Educ . 2014 (2).

致 谢

岁月如梭，忙碌将近半年，终于较为圆满的完成了自己的毕业设计及论文，随着毕业论文的完成，四年本科的生涯即将结束。很感谢在撰写论文的过程中老师、同学及家人的帮助和指导。

首先，要感谢我的指导老师，在我毕业设计与实现的过程中认真引导，抽出时间对我进行耐心教导，帮助我解决一个个问题，给了我不止学术上，同时在生活上许多的宝贵意见。通过论文的撰写，我学到了书本以外的知识，培养了自己分析问题、解决问题的能力。

其次，要感谢我的同学及好友们，是他们在我毕业设计遇到挫折和迷茫时，对我进行及时的帮助和开导，结合自身经验与我进行学习和研究上的交流，帮助我走出困境，坚定信心，成功完成本次毕业论文。

最后，再次向他们表示我最衷心的感谢和最诚挚的敬意！