课程论文

学年学期 **20** —**20** 学年第 学期

课程名称 软件工程综合实践二

题 目 人事管理系统

姓 名 秦梓豪

学 号 2020041223

学 院 计算机科学与信息工程学院

任课教师 林琳

论文评语（由任课教师填写）：

成 绩

**摘要**：随着社会的发展，互联网的利用率也在快速提高，在互联网高速发展的同时也大大提高了人们的生活质量。在企业人事管理这方面，计算机的普及大大减少了公司的管理成本，而且效率也得到了很大的提高。人事管理系统主要从公司人事管理员的角度管理管理公司的员工信息。

**关键词：**人事管理；人事管理系统；效率

**第1章 项目说明**

**1.1项目内容**

介绍自己项目的基本情况、包括项目功能、项目开发平台、项目针对用户等信息

**1.1.1项目功能**

人事管理系统功能包括人事通讯、员工信息、人事考评、奖惩、培训管理、薪资管理、统计分析、和系统管理六大模块、对应人事工作基本流程：新员工入职档案建立、调动、辞职、员工信息的查询及工资管理等方面。系统管理可以根据不同的角色分配菜单权限设置，不同的用户授予不同角色，对人事结构、单位结构进行整体调配设置。在线聊天可以实现操作员之间讯息的及时通讯。

**1.1.2项目开发**

项目基于MVVM的前后端分离开发模式进行开发。MVVM即模型（Model）-视图（View）-视图模型（View Model），实现了数据视图的双向绑定。相对于MVC模式和MVP来说，MVVM是一个比较新的开发架构，它是一种将MVP模式与WPF相结合应用方式发展演变而成的新型开发架。

前后端分离是指将前端和后端从之前的全部由后端负责中分离开来，不再共用一个Server，前端作为一个独立Serve存在。前后端通过接口使用HTTP协议交互，本项目使用vu实体属性outer做前端路由处理。页面跳转不在由后端处理，前后端只是数据的交互。前后端分离的好处在于降低了前后端的耦合性。当面对不同的硬件场景时，需要构建不同的界面，前后端分离之后，只需要扩展前端项目即可，不需要修改后端服务。

**1.1.3项目用户**

项目主要面向公司，用于公司人事的管理。

**1.2 项目背景**

人事管理是企业管理的一项重要内容，随着时代的进步以及科技的发展，如何管理好企业内部员工让人事管理变得规范化，是企业管理的重要内容。为了将人力资源管理规范化，人事管理系统成为国内外许多企业都热捧的产品。人事管理的自动化有利于企业对人力资源充分利用，同时方便员工对自己考勤、工资、职位的查看与管理。人事管理系统大大减轻了企业人事部门的工作压力，也方便了企业对员工进行管理。目前，许多的中小型企业管理存在着制度缺失，组织管理结构的不健全、员工管理工作无法统一、信息化自动化管理水平偏低。这些是导致企业人力资源流失的主导因素。人力资源的流失无疑对各个企业造成了很大的损失，很多企业对人事管理仍不重视，依旧利用人工进行管理，这样大大降低了企业对人力资源的利用率，无疑会对企业的发展产生制约和负面影响。显然基于纸质档案人工查询或修改员工信息已经不再适用于企业对人力资源的管理与运用。总体来说就是这种管理方式缺乏系统化、规范化的管理手段，使得计算机资源没有得到实际应用。因此，开发一个有利于企业与员工的管理系统十分重要。

**第2章 需求分析**

**2.1功能需求**

图2.1 XXX图—宋体五号字，图、图题居中，其中图编号图x.y ，x为章号，y为章内流水号

**2.1.1登录功能**

分为管理员登录和用户登录，系统需要先根据用户所输入的身份信息进行登录验证来判断该用户是否存在，若不存在则进行提示，若存在则进入各自界面：管理员界面和用户界面，来进行各自操作。

**2.1.2员工资料功能**

实现对员工基本资料的增删改查。实现员工基本资料表Excel导入导出，多条件搜索。添加员工由数据库自动生成工号，其他信息为必填项。修改根据id进行数据更新。

**2.1.3人事管理功能**

人事管理主要用于对员工奖惩、培训、考评、调动信息的管理。员工奖惩主要对员工奖惩模块的增删改查的实现。在UI设计上使用穿梭框和tab标签，逻辑处理上员工实体和奖惩信息是一对多的关系,即对同一员工在相同或者不同时间都可以添加多个奖惩，奖惩管理页面显示有奖惩信息的全部员工,可以查看奖惩详情。

**2.1.4工资管理功能**

实现对员工工资账套管理，员工账套设置，工资表管理，查询一系列对员工工资管理的功能。工资账套管理主要对员工奖金,基本工资,提成等各项的增刪改查. 设置当前套账信息。

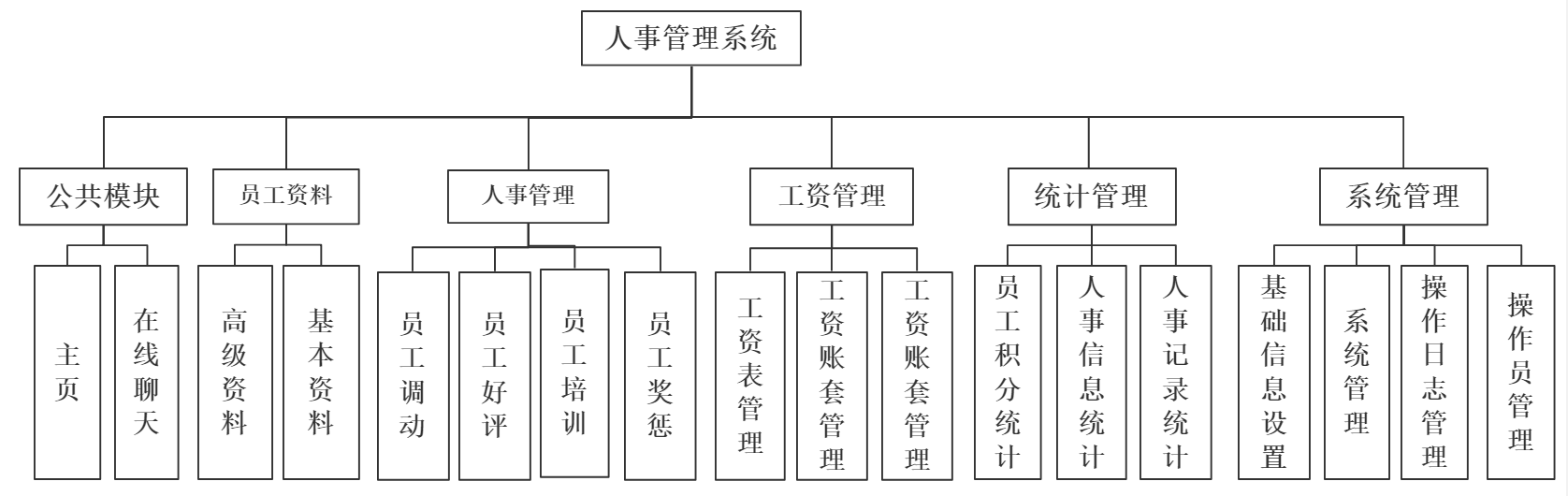
**2.1.5统计管理功能**

统计管理主要对员工资料的动态分析统计，包括员工积分统计、人事信息统计、人事记录统计。

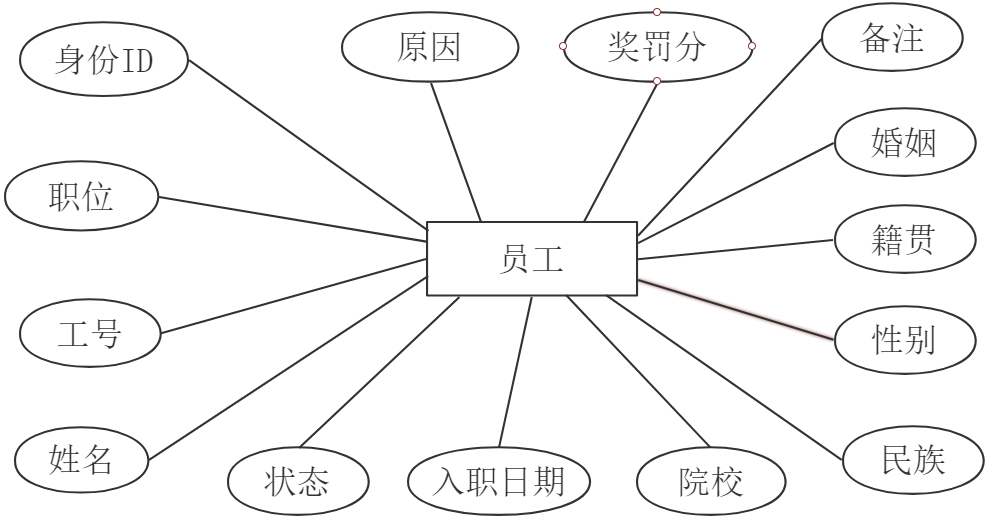
**第3章 项目设计**

**3.1项目设计**

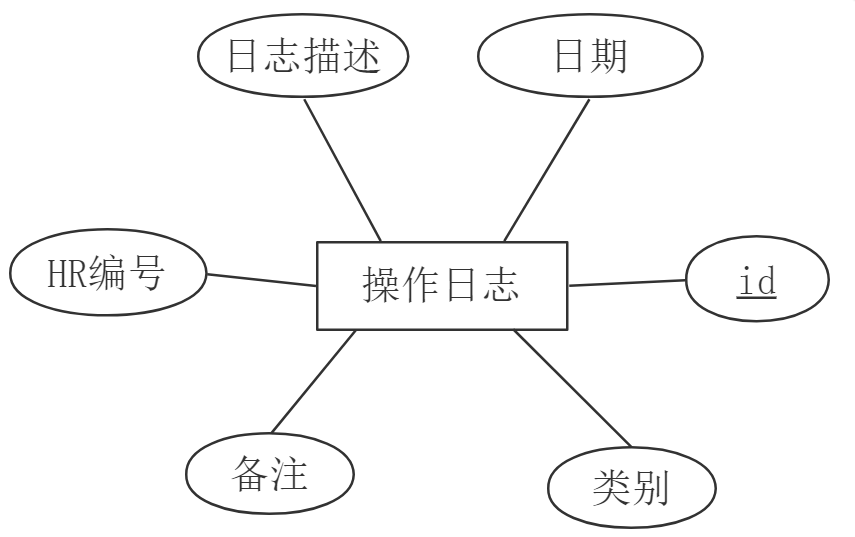
**3.1.1系统功能模块图**



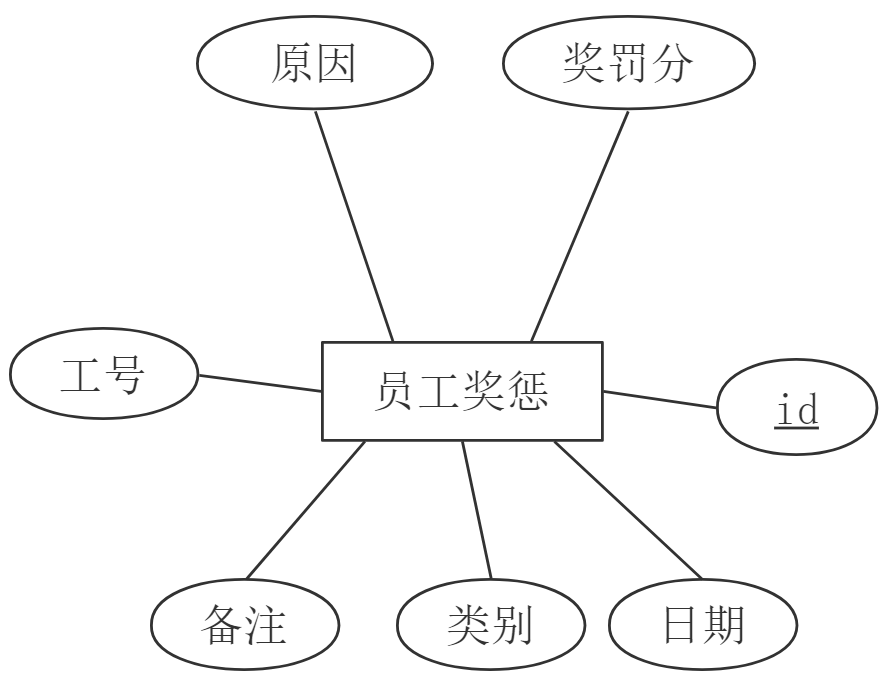
**3.1.2员工实体属性图**



**3.1.3员工奖惩实体属性图**



**3.1.4操作日志实体属性图**

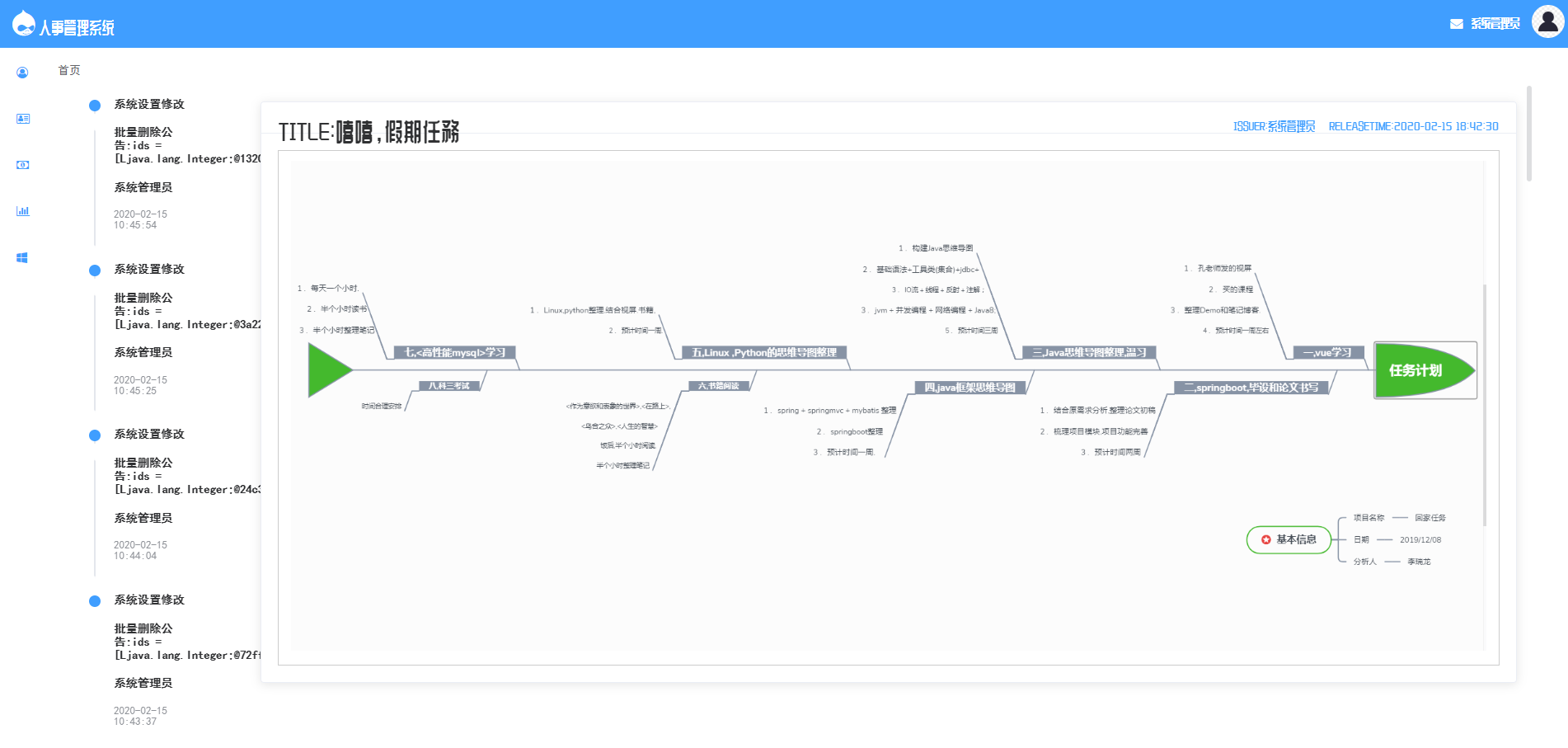


**第4章 项目实现**

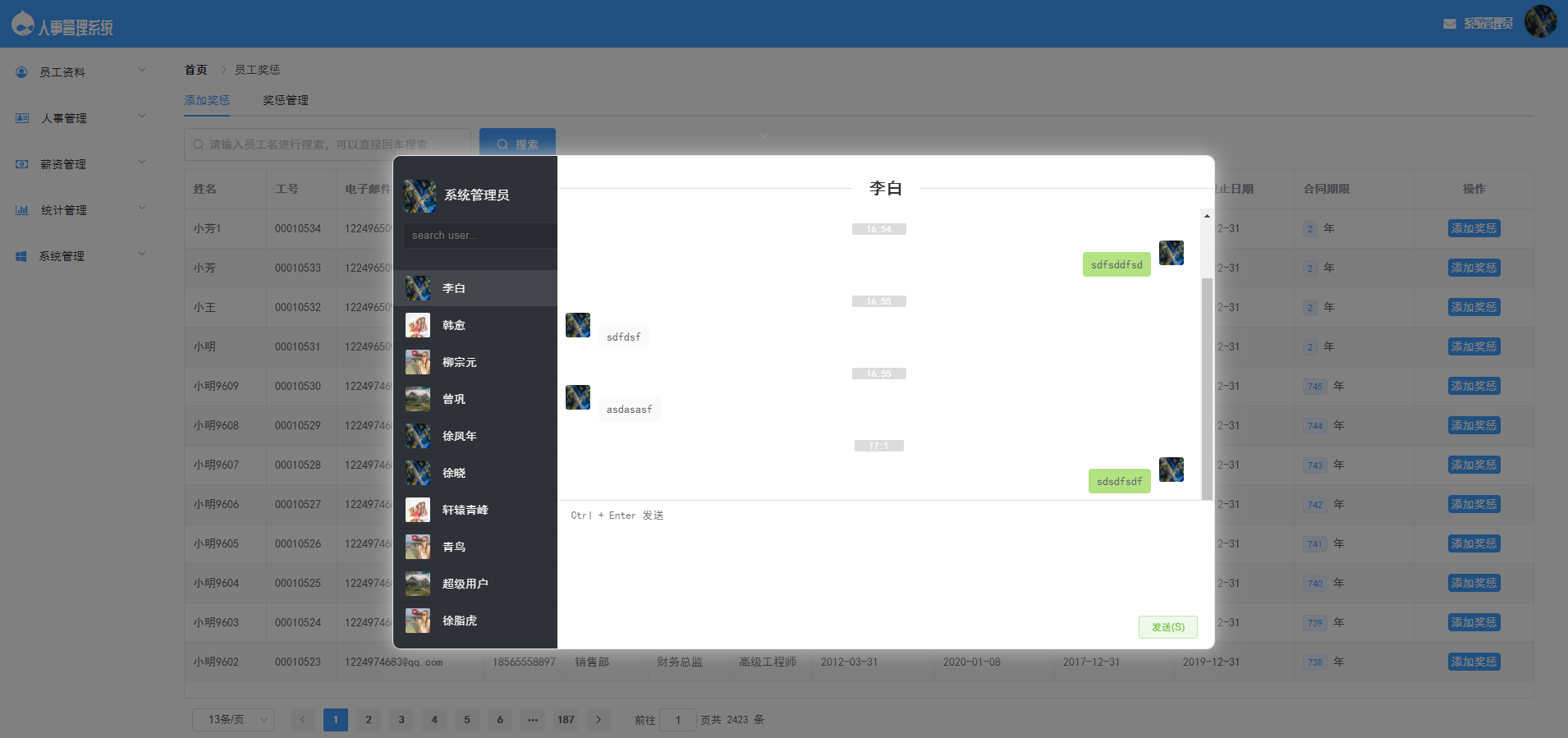
将该项目中重要的功能及用法的实现图展示出来。同时连接有数据库的将数据库部分的实现与构建也要展示出来

**4.1项目实现图**

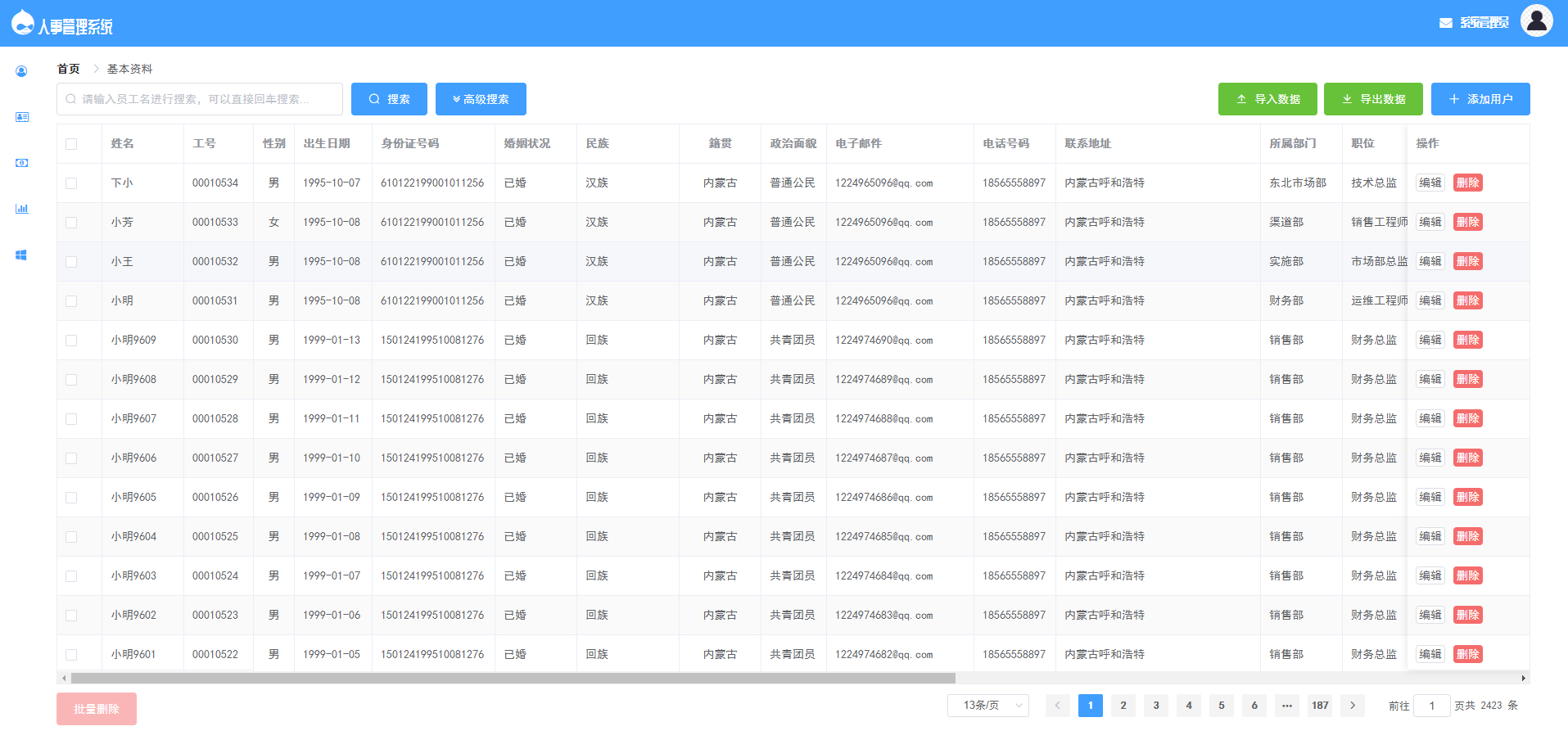
**4.1.1系统主页**



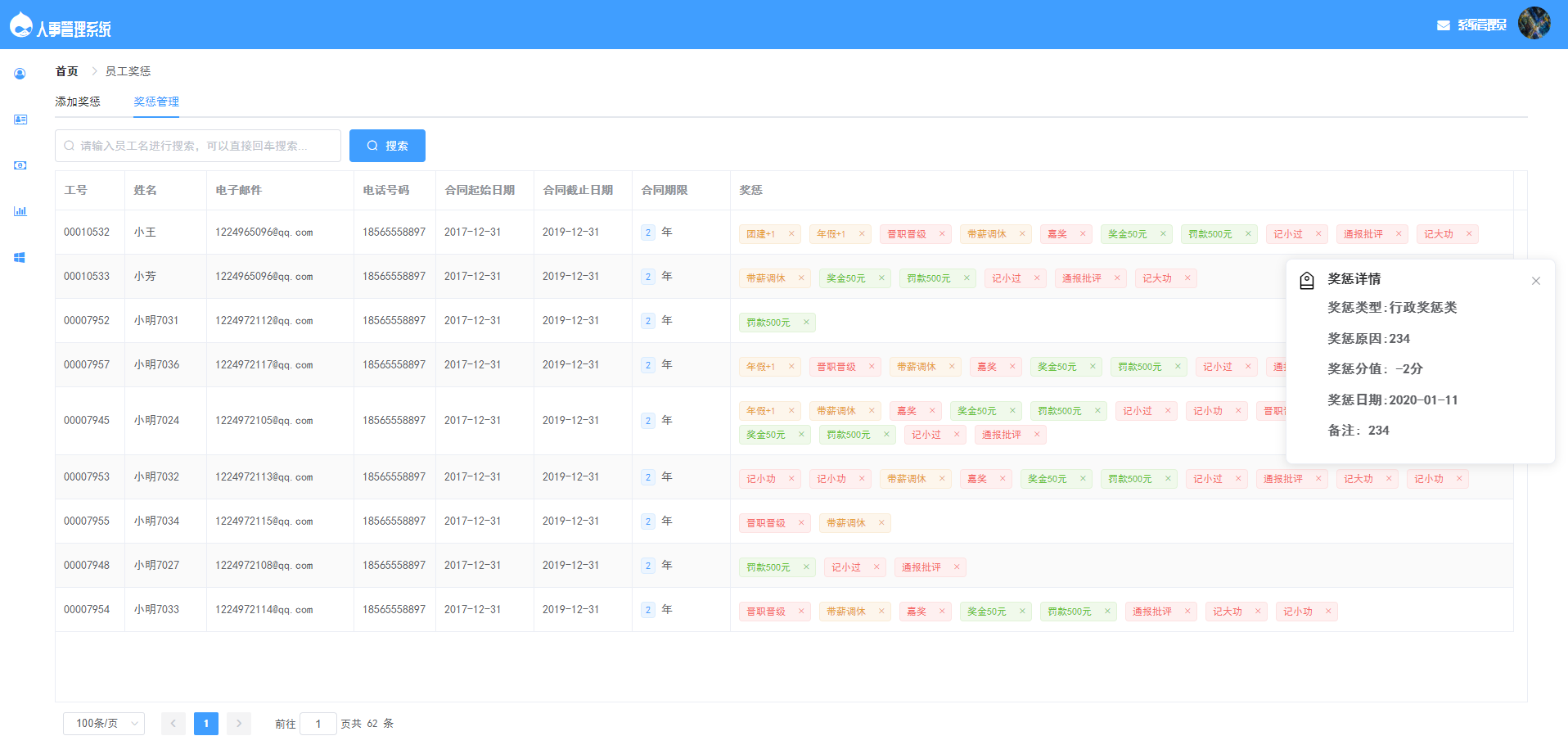
**4.1.2在线聊天**



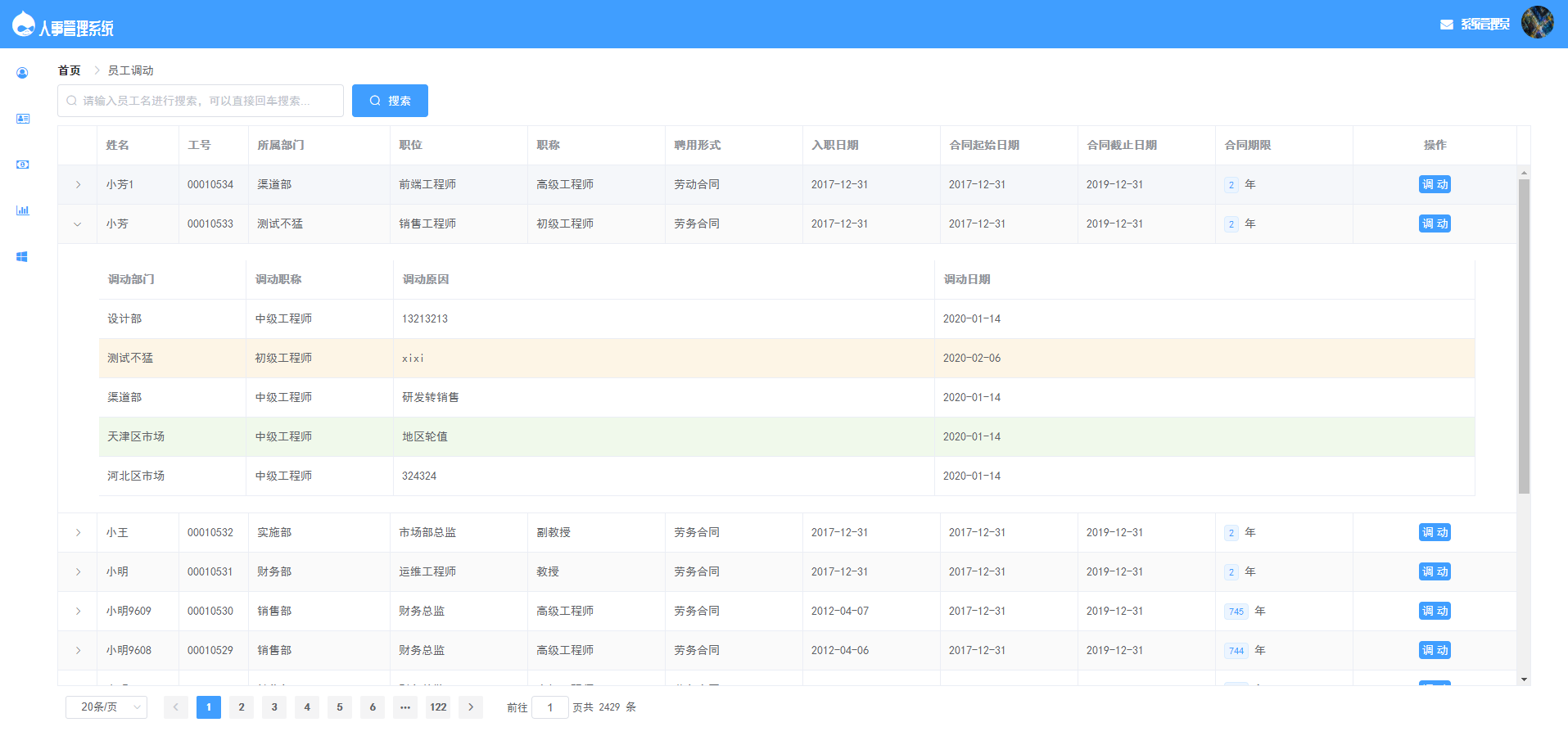
**4.1.3员工资料**



**4.1.4员工奖惩管理**



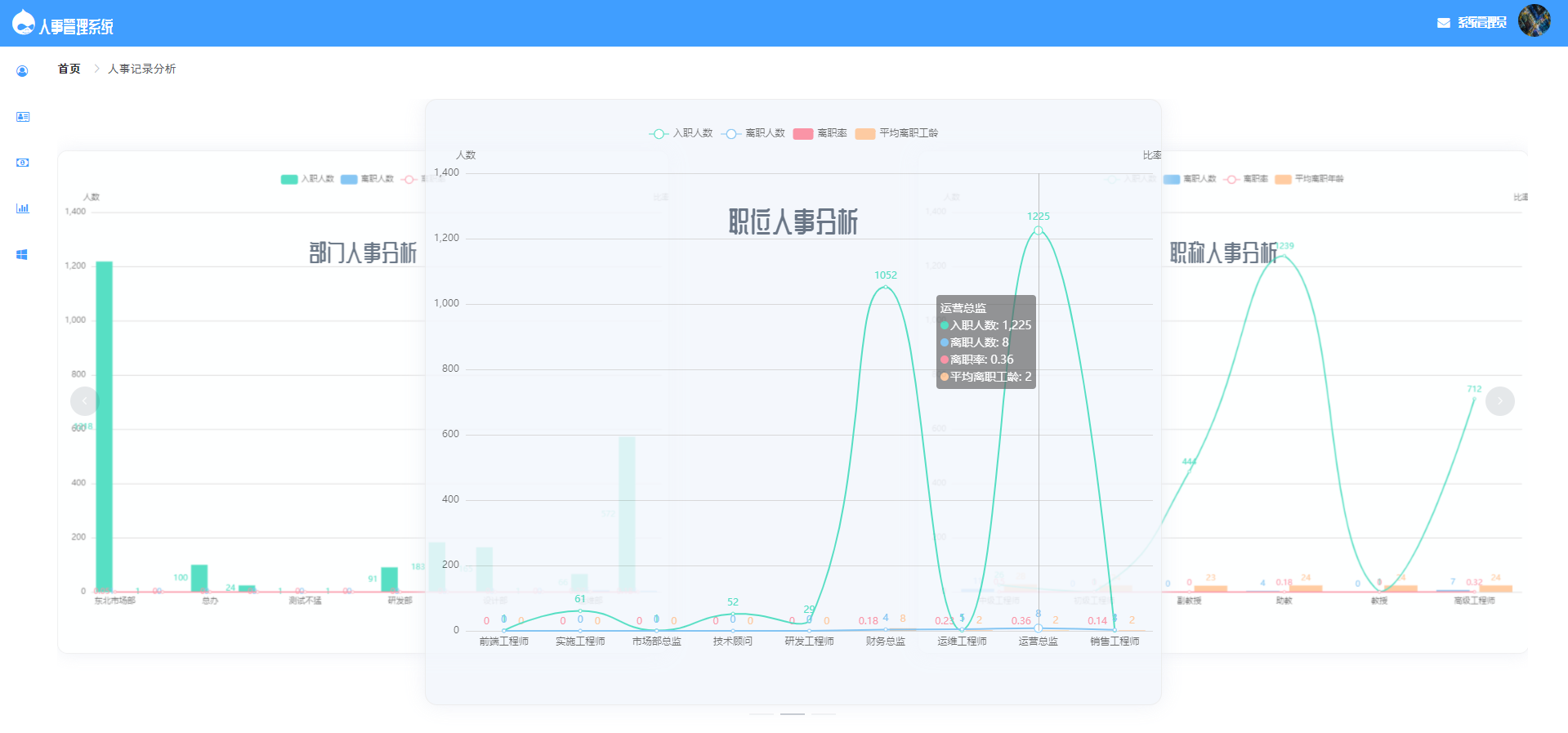
**4.1.5员工调动**



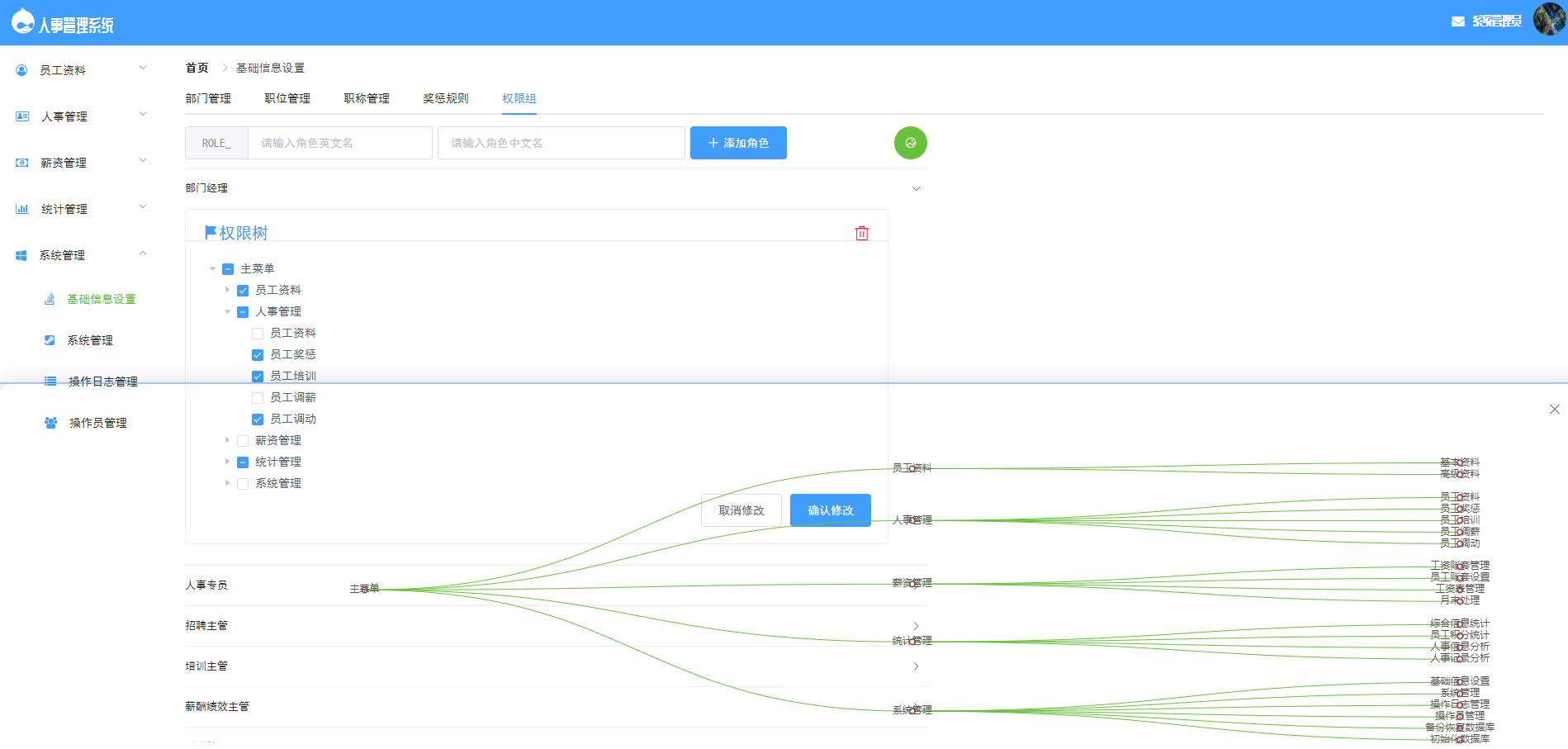
**4.1.6员工工资管理**



**4.1.7人事记录统计**



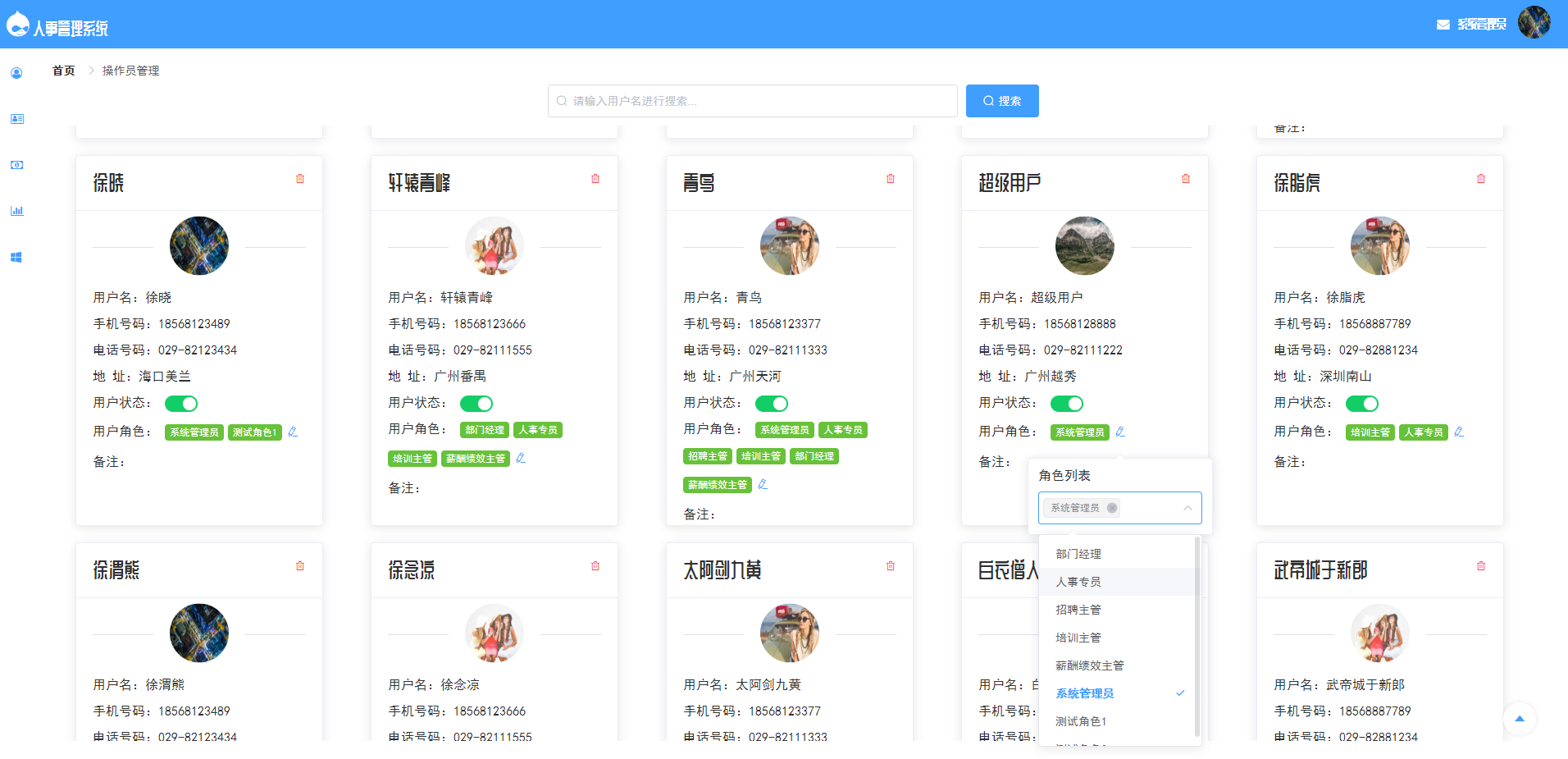
**4.1.8用户权限管理**



**4.1.9公告编辑**



**4.1.10操作员管理**



**4.2数据库表**

**4.2.1appeaise表**

|  |  |  |  |
| --- | --- | --- | --- |
| **列名** | **数值类型** | **是否为空** | **备注** |
| id | int(11) | NOT NULL | 主键id |
| eid | int(11) | NULL | 编号 |
| appDate | date | NULL | 考评日期 |
| appResult | varchar(255) | NULL | 考评结果 |
| appContent | varchar(255) | NULL | 考评内容 |
| remark | varchar(255) | NULL | 备注 |

**4.2.2employeeec表**

|  |  |  |  |
| --- | --- | --- | --- |
| **列名** | **数值类型** | **是否为空** | **备注** |
| id | int(11) | NOT NULL | 主键id |
| eid | int(11) | NULL | 员工编号 |
| ecDate | date | NULL | 奖罚日期 |
| ecReason | varchar(255) | NULL | 奖罚原因 |
| ecPoint | int(11) | NULL | 奖罚分 |
| ecType | int(11) | NULL | 奖罚类别 |
| remark | varchar(255) | NULL | 备注 |

**4.2.3employee表**

|  |  |  |  |
| --- | --- | --- | --- |
| **列名** | **数值类型** | **是否为空** | **备注** |
| id | int(11) | NOT NULL | 员工编号 |
| name | varchar(10) | NULL | 员工姓名 |
| gender | char(4) | NULL | 性别 |
| birthday | date | NULL | 出生日期 |
| idCard | char(18) | NULL | 身份证号 |
| wedlock | enum('已婚','未婚','离异') | NULL | 婚姻状况 |
| nationId | int(8) | NULL | 民族 |
| nativePlace | varchar(20) | NULL | 籍贯 |
| politicId | int(8) | NULL | 政治面貌 |
| email | varchar(20) | NULL | 邮箱 |
| phone | varchar(11) | NULL | 电话号码 |
| address | varchar(64) | NULL | 联系地址 |
| departmentId | int(11) | NULL | 所属部门 |
| jobLevelId | int(11) | NULL | 职称ID |
| posId | int(11) | NULL | 职位ID |
| engageForm | varchar(8) | NULL | 聘用形式 |
| tiptopDegree | enum('博士','硕士…… | NULL | 最高学历 |
| specialty | varchar(32) | NULL | 所属专业 |
| school | varchar(32) | NULL | 毕业院校 |
| beginDate | date | NULL | 入职日期 |
| workState | enum('在职','离职') | NULL | 在职状态 |
| workID | char(8) | NULL | 工号 |
| contractTerm | double | NULL | 合同期限 |
| conversionTime | date | NULL | 转正日期 |
| notWorkDate | date | NULL | 离职日期 |
| beginContract | date | NULL | 合同起始日期 |
| endContract | date | NULL | 合同终止日期 |
| workAge | int(11) | NULL | 工龄 |

**4.2.4salary表**

|  |  |  |  |
| --- | --- | --- | --- |
| **列名** | **数值类型** | **是否为空** | **备注** |
| id | int(11) | NOT NULL | 主键id |
| basicSalary | int(11) | NULL | 基本工资 |
| bonus | int(11) | NULL | 奖金 |
| lunchSalary | int(11) | NULL | 午餐补助 |
| trafficSalary | int(11) | NULL | 交通补助 |
| allSalary | int(11) | NULL | 应发工资 |
| pensionBase | int(11) | NULL | 养老金基数 |
| pensionPer | float | NULL | 养老金比率 |
| createDate | timestamp | NULL | 启用时间 |
| medicalBase | int(11) | NULL | 医疗基数 |
| medicalPer | float | NULL | 医疗保险比率 |
| accumulationFundBase | int(11) | NULL | 公积金基数 |
| accumulationFundPer | float | NULL | 公积金比率 |
| name | varchar(255) | NULL | 名称 |

**4.2.5oplog表**

|  |  |  |  |
| --- | --- | --- | --- |
| **列名** | **数值类型** | **是否为空** | **备注** |
| id | int(11) | NOT NULL | 主键id |
| addDate | date | NULL | 添加日期 |
| operate | varchar(255) | NULL | 操作内容 |
| hrid | int(11) | NULL | 操作员ID |

1. **项目测试**

**5.1系统测试**

软件在设计之后必须进行测试，主要从系统性能测试、功能测试和测试结果分析进行描述。

**5.1.1软件测试的目的**

软件在设计之后必须进行测试，主要从系统性能测试、功能测试和测试结果分析进行描述。

1. 、确认软件的质量，即软件是否满足了所有功能或性能需求。
2. 、将测试信息提供给开发、运营或项目经理，为之后的风险评估等做准备。
3. 、确保整个软件开发过程的质量，如果一个软件产品开发完成之后，发现了很多问题，这 说明此软件开发过程很可能是有缺陷的。

**5.1.2功能测试**

功能测试主要是基于黑盒测试的方法,也就是不管某个功能后台代码的业务 实现逻辑是怎样,只需要根据开发人员提供的接口相关的测试数据,从前端界面或使用第三方工具如Postman等对各模块功能中的主要操作进行了功能测试.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能测试** | 员工基本信息管理 | | | | |
| **测试目的** | 操作是否得到响应 | | | | |
| **测试内容** | **操作描述** | **数据** | **期望结果** | **实际结果** | **测试状态** |
| **删除员工** | 单机删除  按钮 | -- | 员工列消失,数据数据库没有  数据 | 员工列消失,数据数据库没有  数据 | 与预期  结果相同 |
| **修改员工**  **信息** | 单机员工列  修改按钮,输  入修改数据 | 姓名：小明106  改为：小明107 | 员工列数据  变成小明  107 | 员工列数据  变成小明  107 | 与预期  结果相同 |
| **搜索员工**  **信息** | 输入员工信  息,单机搜索 | 搜索框输入：  小明107 | 小明107  列显示 | 小明107  列显示 | 与预期  结果相同 |
| **添加员工** | 单机添加,输入数据 | 小明108….. | 小明108  列显示 | 小明108  列显示 | 与预期  结果相同 |

**5.1.3测试结论**

在开发中，开发流程是先设计界面，然后写接口，之后使用Postman进行接口测试，确认返回的JSON没有问题之后，写前端逻辑。最后会进行数据测试，项目中Spring Boot集成Juntil框架提拱了很好的测试手段，可以快速构建测试案例，同时基于断言策略很好的提供了对功能逻辑的正确性检查，在测试过程中，我发现对于异常的处理，不能在项目开始就定义全局异常处理器，即使定义，也不能直接定义父类异常，如果直接定义父类异常，Spring Boot会直接处理异常返回白页(空白页)，这样的话，不利于溯源异常点。

**第6章 总结**

**6.1总结**

通过对基于Spring Boot框架的人事系统的开发，我对系统开发有一个更清楚的了解，系统的开发过程中，不但熟悉了人事系统业务需求，同时，也拓宽了我对Spring Boot的了解。

在人事管理的业务方面，本系统涉及业务逻辑较浅，与人事关联的考勤、工资、日常审批的业务，本系统都没有涉及到。在系统架构方面，本系统使用基本单模块、单服务、功能之间耦合性强。在此基础上，可以随着业务的深入进行模块服务解耦，实现多服务、接口交互架构。在多服务的基础上还可以进行云原生微服务架构。在使用场景方面，除了PC端，还可以开发面向微信的流应用，即微信小程序。所以，本系统的有很大的迭代升级空间。