

**本科毕业设计**

**题目 基于WEB的人事管理系统**

|  |  |
| --- | --- |
| **学院** | **计算机工程学院** |
| **专业** | **计算机科学与技术** |
| **年级** | **2017级** |
| **学号** | **192017113175** |
| **姓名** | **王海波** |
| **指导教师** | **何俊梅** |

**2021年 5 月**

**目 录**

[一、绪论 5](#_Toc12787)

[（一）选题的背景及意义 5](#_Toc24478)

[（二）课题研究现状 5](#_Toc9165)

[（三）研究内容 6](#_Toc18908)

[二、开发环境介绍 7](#_Toc12614)

[（一）MVVM+Chrome浏览器 7](#_Toc6644)

[（二）Html5+css3 7](#_Toc16410)

[（三）JavaScript简介 8](#_Toc13896)

[（四）LAYUI技术 8](#_Toc15955)

[（五）Node+MYSQL+Visual Studio Code 8](#_Toc14626)

[三、系统分析 9](#_Toc22480)

[（一）系统可行性分析 9](#_Toc13987)

[1．背景 9](#_Toc28937)

[（二）系统可行性分析 9](#_Toc26347)

[1．技术可行性 9](#_Toc22776)

[2．操作可行性 9](#_Toc32485)

[3. 经济可行性 9](#_Toc29522)

[4. 管理可行性分析 9](#_Toc5151)

[（三）需求分析 10](#_Toc25967)

[1．功能需求分析 10](#_Toc3206)

[2．系统需求分析 10](#_Toc25496)

[四、系统设计与实现 11](#_Toc24849)

[（一）系统架构设计 11](#_Toc27021)

[（二）模块设计 11](#_Toc29330)

[1.基本信息管理 11](#_Toc16520)

[2.软件结构图 11](#_Toc27210)

[3.系统数据流图 12](#_Toc1541)

[（三）数据库设计 13](#_Toc11260)

[（五） 详细设计 15](#_Toc16917)

[1.登录界面设计 15](#_Toc19839)

[2.主页面设计 15](#_Toc20748)

[3. 员工信息页面设计 16](#_Toc26676)

[4.人事管理页面设计 16](#_Toc17344)

[5.工资管理页面设计 17](#_Toc31425)

[五、系统的实现 18](#_Toc12990)

[（一）各项功能实现 18](#_Toc1531)

[1．登录功能展示 18](#_Toc9239)

[2.主页面功能展示 18](#_Toc9716)

[3.员工资料的展示 19](#_Toc24765)

[4.添加员工的展示 19](#_Toc12407)

[5.删除员工的展示 20](#_Toc15740)

[6.员工考勤的展示 20](#_Toc27778)

[7.发布公司通知的展示 21](#_Toc15643)

[8.重置密码的功能的展示 21](#_Toc21122)

[六、 系统的测试 22](#_Toc23592)

[（一）系统测试目的与意义 22](#_Toc849)

[（二）测试过程 22](#_Toc7290)

[（三）测试结果分析 23](#_Toc21576)

[参考文献 24](#_Toc21872)

[致 谢 25](#_Toc242)

[附录A 26](#_Toc14193)

**基于WEB人事管理系统的设计与实现**

**摘要：**在随着现在计算机技术的不断进步，以及社会的发展，企业单位人事管理是非常繁琐的，传统的管理技术不再满足企业的需要，越来越多的企业注重计算机信息管理系统，人事信息管理系统是典型的计算机信息管理系统之一，企业借助于它进行人力资源管理，达到事半功倍。

本系统采用前后端分离的开发模式，采用NODE开发API接口，采用MYSQL数据库，主要功能包括，管理员登录，员工信息，员工考勤，员工工资，员工调薪模块等，本系统还采用高级加密的方式保存管理员的密码，系统界面美观，简单实用，安全性高，能够基本上满足企业人事信息管理的需要，实现了企业员工信息的整体化，系统化，规范化和自动化。

**关键词：**HTML 人事管理 MVVM 数据库设计 JavaScript

**Abstract：**In this era of information developed, with the rise of the technology, from the previous simple page to now very rich display of the 3 d interface, the interface of the display by the original basic text and graphic display to show around the form now, very fully illustrates the improvement of technology, now the employment pressure also big, with the enlargement of the company, also become very common information management way, the development of the personnel management system is necessary, so that we can improve the efficiency of the company, make management more efficient to perform a task, development of personnel management system, adopted the pattern of separation of before and after the end,The background will use MYSQL database and Node development, front-end using framework and JavaScript technology to achieve the website.

**Key words**: HTML, Administration, MVVM, Database Design, JavaScript

# 

一、绪论

## **（一）选题的背景及意义**

社会的进步，信息化时代的到来，大数据已成为了现在的主流，对于企业来说，人事管理系统是一个企业单位必不可少的，为人事管理和高层决策领导提供可靠性高的参考数据，所以我们要解决这些问题，为企业的利益提供必要的帮助，要尽最大能力为企业提供应有的帮助。现在社会是一个全新的社会，在当下，到处都是人工智能，各种科技技术日益更新，在更智能化的情况下提升公司的利益，更好的去管理员工，人事管理系统可以更高效的提高管理者的效率，让管理者更轻松。

面对竞争，我们只能迎头而上，人才是当今社会最稀缺的，谁占领了人才市场的制高点，就会让其在市场竞争中脱颖而出，竞争越激烈，企业人事管理就越显示其的重要性，成为企业非常重要的一个考察机制，每个企业都会严格要求自己的员工，从而提高其企业的价值，更好的展现给社会从而吸引更多的有才之士相投，使用人事管理系统，人事管理系统是非常符合公司的发展，提高公司的价值，是必不可缺的。在当代社会，人才是极度缺少的，那么管理好人才也是非常重要的，要做到如何并且恰当的处理员工的工作岗位，分析员工的数据，员工的考勤等等都需要管理者花费大量的时间去进行操作，那么就需要人事管理系统来让管理者更加便捷。

## **（二）课题研究现状**

如今，随着大数据时代的到来，我们应紧随时代的步伐，现在许多单位已都已采用电子化的人事管理系统，事管理系统对基础数据的收集和储存，使企业的人力资源管理效率得以提升。举例：员工管理、档案管理、考勤排班、薪酬管理、招聘培训等日常人事工作，如果人工统计核算，要占据HR大量的时间和精力，出错是难以避免的。人事管理系统，只要规则设置好，可以一键快速处理大量的数据工作，解放了HR，并为业务部门提供数据支持。同时，通过人事管理系统，员工可查询、修改、提交审批等活动，并与同事、领导、HR等进行交流和互动，大大提升了沟通效率。不同角色员工的自助管理平台，但也有必须面临的问题，员工信息复杂化，大量的信息数据不精确，复杂程度也可想而知，想要靠几个人就能处理大量的数据已再无可能，面对当下的问题，只有研究更高效，更快捷，更方便的技术，然而人事管理就是最佳的选项，不仅能提高企业的办事效率，通过高科技技术信息采集和处理，辅助提高企业人事管理水平，还能提高企业对于员工更深度的考核，对企业的将来提供更高质量的保障。所以说，人事管理发展前景非常的好。它不仅可以让管理者更加轻松，而且还让公司人员能看到自己的工作效率，记录自己的工作时间和自己的工作资料等等，生成一份电子版的档案，也让自己在以后的工作中有更多的选择机会。是非常不错的选择，提高工作效率的必要选择。

## **（三）研究内容**

首先对项目进行调查研究、案例研究，整理出相关的资料，然后确定项目，在确定项目以后，向指导老师请教课题的含义与研究方向，经过老师指导后进行网上查阅相关专业知识，主要在万方，维普，知网等学术平台查阅有关的文献资料，然后再对所需要的技术环境的查找一些资料，可以通过视频等方式进行学习。此项目采用前后端分离进行开发，然后对环境进行搭建，前端环境需要谷歌浏览器和VSCODE工具，后端环境需要Node环境，数据库采用MYSQL数据库，前端框架采用LAYUI框架和响应式的Bootstrap框架，在MYSQL中创建数据表，最后进行数据的调用，前后端交互的连接等。最后把项目进行打包。最后对项目进行必要的测试，首先打开数据库和软件，然后导入下项目，开启服务，进行测试，最后总结测试结果。

（1）网站页面设计：此系统包含前台和后台页面，前台页面是登录模块，后台页面又分为多功能模块。登录页面只能是管理员才能登录，用于管理和查看员工的信息和员工的薪资等问题。

（2）功能模块：人事管理系统有登录、退出功能，修改员工信息功能，增加员工功能，考勤管理功能，员工管理功能，权限管理功能。

（3）数据库逻辑设计：利用MYSQL数据库进行基本功能处理，实现数据的增删改查等功能，采用的是NODE开发，非常好的利用浏览器就可以弄完的后台项目。

（4）系统流程设计：开发环境介绍即采用前后端分离的方式，前端采用HTML+CSS+JavaScript搭建前端框架LAYUI，页面布局采用响应式布局，开发工具是Chrome浏览器和Visual Studio Code工具进行开发，数据库采用MYSQL数据库简单方便，后端采用Node。JS技术，编写后端API接口可以直接利用JavaScript技术直接进行编写。启动监听程序，实现页面的交互，系统架构图即画出系统的结构图，流程图，E-R图等。

# 二、开发环境介绍

## **（一）MVVM+Chrome浏览器**

MVVM它本质上就是MVC 的升级版。对于前端来说，就是模型，视图，模型视图，他们能实现响应式，边修改边渲染，这样让人的开发变得简单，查看起来更加方便，让有焕然一新的感觉，能充分体现技术酷炫之处。

Google是一款非常好用的浏览器，浏览器就是前端的运行框架，是开发前端界面的必要工具。谷歌浏览器的访问速度和加载速度都是非常快的，所以我们采用谷歌浏览器进行开发是非常有必要的。一个浏览器没有什么加速器，但是它无论冷启动、还是加载网页都是最快的，一个浏览器竟然能快速推动WEB的发展，最重要的是标准的统一和与时俱进，程序员的福音，一个浏览器可以做到全平台同步，登录账号就可以全平台同步填充密码，一个浏览器你不用等着它更新什么功能，而是直接去插件市场找又推出哪些新插件。

## **（二）Html5+css3**

Html5是超文本标记语言的升级版，css3是CSS的升级版，他们一个控制结构一个控制样式，他们之间完美结合，让页面展现的更加直观，让浏览者更加直观，让操作者更加便捷的去进行操作，有了这两个技术，在页面的展示，他们就是强者中的强者，更加直观，他们只需要浏览器这个运行环境，就能展现出更强大的功能。是前端开发必不可少的技术。HTML5本身是由W3C推荐出来的，它的开发是通过谷歌、苹果，诺基亚、中国移动等几百家公司一起酝酿的技术，这个技术最大的好处在于它是一个公开的技术。换句话说，每一个公开的标准都可以根据W3C的资料库找寻根源。另一方面，W3C通过的HTML5标准也就意味着每一个浏览器或每一个平台都会去实现。

CSS对于一个好的程序员来说是必要的工具，不必再把烦杂的样式定义编写在文档中，可以将所有有关文档的样式指定内容全部脱离出来，在行定义、在标题中定义，甚至作为外部样式文件供HTML调用。同时，在定义的时候，不用考虑到浏览器兼容的问题。首先，他简化了页面的编写，外部的样式表还会被浏览器保存在缓存中，加快了加载速度，也减少了需要上传的代码数量，只要修改保存着网站格式的CSS样式表文件，就可以更改整个网站的风格和特色，这对于大型网站，多页面网站有着重要作用。避免了逐个页面的修改，缩短了修改页面的时间。

## **（三）JavaScript简介**

JavaScript是一种解释型的脚本语言，受对象和环境的驱动，风险相对较低，也可以用多种书面语言进行客户端的网页浏览，一般可以用来增强HTML页面的移动功能，如使用control，JavaScript的关键功能之一就是发挥作用。基于组件的软件开发，可以直接、系统、可重用地设计应用程序，根据HTML应用程序，使用JavaScript设计交互式网站。显示非常丰富的信息和非常好的内容。许多JavaScript设计使用这个系统。他是负责让页面动起来的技术，可以展现更加直观生动的形态，也让开发者更有兴趣，让需要开发的系统更加活性化。

## **（四）LAYUI技术**

LAYUI是一款绝好的前端 UI 框架，遵循原生 HTML/CSS/JS 的书写与组织形式，门槛极低，拿来即用。其外在极简，却又不失饱满的内在，体积轻盈，组件丰盈，从核心代码到 API 的每一处细节都经过精心雕琢，非常适合界面的快速开发。当然，想让一个页面从无到有，只凭基础的样式肯定是很难的，那样会让你的代码变的非常多，那么我们就需要框架，LAYUI就是一个非常适合的框架，他拥有非常强大的功能，我们只需要学会一些简单的操作，就可以避免过多的代码，可以直接进行代码的复用即可，拥有非常简洁的画面，各种模块都有文档说明，他拥有很多内置模块，是JS封装而成的，我们只需对它进行改善，来实现自己的项目。

## **（五）Node+MYSQL+Visual Studio Code**

微Node在非浏览器端运行的JavaScript，包括前端环境搭建，云构建，服务，中间件，都有node.JS 的应用场景。他提供了一个前端的JS运行环境，让代码更加丰富化，具有非常强大的功能，是前端开发者掌握的一门最简单的后端技术的必要。

MYSQL是一个关系型数据库管理系统，他是一种小型的数据库，同样他也是非常简单的，他的操作很简单，实现原理也很容易理解，这样就避免需要花大量的时间进行学习和研究，我们只需要掌握简单的增删改查即可，同样MYSQL是一款免费的软件，对我们的开发和学习是非常友好的。

Visual Studio Code（简称“VS Code”）是Microsoft在2015年4月30日Build开发者大会上正式宣布一个运行于 Mac OS X、Windows和 Linux 之上的，针对于编写现代Web和云应用的跨平台源代码编辑器，他拥有很强大的功能，他有各种各样的插件展示，让开发变得不再无趣，他是一款运行及环境搭配齐聚的软件，是非常适合页面设计的开发人员。这是一款集成式的软件，最大的好处是支持多种文件格式的文件，对开发者非常友好，里面的插件可以进行英汉互译，对英语不好的开发者是相当重要的，让开发变得简单。

**三、系统分析**

## **（一）系统可行性分析**

### **1．背景**

随着公司的发展，一个公司的人员也逐渐增多，那么对管理者就是一个考验，该如何管理员工，要做到智能化的管理，通过员工的考勤，项目能力，学习能力的检测，可以检测一个公司的强大，现在是人才的时代，“得人才者得天下”也不再是空谈，所以一个好的公司都想培养自己的人才，让自己的公司更加具有吸引力，这就需要管理者对员工进行监督，那么就需要一个人事管理系统，更加方便的检测员工的各项信息及自己的能力表现等。

## **[（二）系统可行性分析](#_Toc137272364)**

### **1．技术可行性**

人事管理系统，采用的是前后端分离的技术，我们不需要非常强大的技术，只需要能做到查看员工的基本信息，员工上下班的考勤，员工的调薪，员工的培训等，需要简单的前后端技术即可实现，所以在技术可行性这个方面，是非常可行的。

### **2．操作可行性**

通过开发人员进行对软件系统的分析，只需要一台电脑，以及服务器，即可实现，软件等都有免费的，只要开发人员花点时间，只要掌握关键的开发技术，那么操作可行性是没有问题的。

### **3. 经济可行性**

现在都是互联网时代，人事管理系统开发的软件都是免费的，除了技术，可能需要一台服务器，这种经济需求是非常小的，只需要在腾讯或者阿里去采购一个，那就能轻松的解决问题，所以经济的可行性是可行的。

### **4. 管理可行性分析**

软件的开发，我们采用的是GITHUB分布式管理，GITHUB是免费的多人开发管理平台，这样我们开发把代码进行上传，如果中途出现错误也可以进行弥补，这样就不会导致整个项目的瘫痪，所以管理可行性是可行的。

## **[（三）需求](#_Toc137272364)分析**

### **1．功能需求分析**

人事管理系统，我们需要对普通的员工进行加密，只有管理员才有权限访问，查看员工信息，调薪信息，考勤信息等，对员工的信息进行各种操作和管理。

### **系统需求分析**

本人事管理系统是一款针对人事管理的软件，它的目的主要是对企业员工进行有效的管理来提高企业的运 转效率和适应现代社会的快节奏。它的功 能比较强大，能够对员工的信息进行管理查询汇总，使用简单，方便对员工的管理。

性能需求分析:

1. 硬件环境：

处理器： Intel Core Duo 或更高,内存： 512MB 以上,软件环境： Visual Studio 2019操作环境： Windows 数据库： MYSQL

1. 功能需求 :

管理员登录，查看员工信息，员工考勤，员工的调动，员工的调薪，员工的培训，员工的工资等。员工各种信息的输入，包括员工的基本信息、学历信息、考勤信息、工资信息。对用户进行授权。员工各种信息的修改。对于转出、辞职、退休员工信息的删除。 按照某种条件，查询统计符合条件的员工信息。对查询、统计的结果输出。人事系统的使用帮助。

1. 环境需求

尽量采用现有软硬件环境，及先进的管理系统开发方案，从而达到充分利用现有资源，提高系统开发水平和应用的目的。系统应符合人事管理的规定，满足日常人事管理工作需要，并达到操作过程中的直观、方便、实用、安全等要求。

1. 适配需求

系统采用 C/S 体系结构， Client （客户端）负责提供表达逻辑、显示用户界面信息、访问数据库服务器； Server （服务器端）则用于提供数据服务。系统分析等前期工作应尽量详细完善，以便学校以后体系结构的改变，对于一些安全性要求不高的信息可以方便地采用 Brower/Server 的方式进行访问。 系统采用模块化程序设计方法，既便于系统功能的各种组合和修改，又便于未参与开发 的技术维护人员补充、维护。 系统应具备数据库维护功能，及时根据用户需求进行数据的添加、删除、修改、备份等操作。

# 四、系统设计与实现

## **（一）系统架构设计**

在整个项目中，我们需要使用正确的软件，写开发文档并且能保存代码的完整性，采用GITHUB进行管理，书写正确的代码，保证逻辑关联的正确关系。

在进行系统开发时，我需要保证最基本的功能，使用渐进增强的开发模式，设计出更加完美的项目。

1.可行性。设计简单的架构，保证基本功能的可行性。

2.安全性。采用TOKEN的登录方式，数据库采用MYSQL使用插件对账号密码以及员工的信息进行加密，保证公司的资产不能泄露。

3.可扩展性。初期保证最基本的功能，后续可以在项目上进行更多的完善，让项目更加美观，让管理员操作更加直观和简单。

4.可维护性。系统运行期间发现和排除BUG，减小对程序的运行负担。采用GITHUB管理方式，随时都能进行管理和更新。

5.可升级性。采用渐进增强的开发模式，保证这个项目后续的开发。

8.简单性。系统操作必须简单、保证各种软件的适配。

## **（二）模块设计**

**1.基本信息管理**

人事管理、员工信息管理，员工调薪管理，员工培训管理，员工工资管理等。人事管理完成用户的分类查询和添加、修改、删除；权限管理完成用户权限的建立、查询、修改、删除；员工信息管理限定用户的权限，添加、修改、删除员工信息等。对员工考勤的信息，调薪等进行各项查看和修改。管理员对自己的信息管理，对账号密码的管理等。

**2.软件结构图**

软件结构图能清晰直观的查看开发系统的功能及结构。让管理员更直观的进行对系统的操作，并且了解软件的系统构架。如图1-1所示。

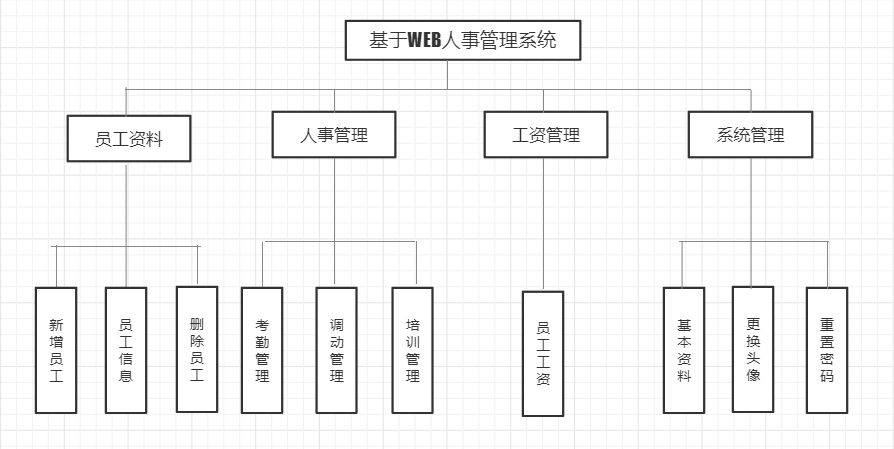


图1-1 人事管理结构图

**3.系统数据流图**

流程图使用几何将是为了更清晰明了的显示每个技术过程的逻辑关系。流程图能更直观的展示操作方式。如图1-2所示。

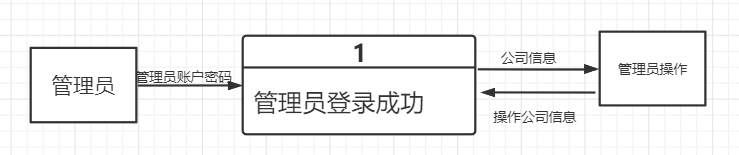


图1-2 系统数据流图

当管理员通过账户密码验证进入到系统，登录到管理系统的后台，进入系统后就能查看人事管理、员工管理、员工信息，系统管理的功能界面，我们不仅能够进行权限的管理，对用户的角色信息进行查看，通过部门机构调出员工的信息。

进入到每个功能界面后我们还能进行具体的操作，具体的部门信息，员工的在职情况等详细的个人信息。

## **数据库设计**

1. **数据库设计原则**

在数据库的建设中，最关键的一步就是设计合理的数据库，即依托合适的环境开发科学、公平的数据库模型，创建一个等价的系统，满足不同用户存储不同内容数据的需要。

(1)范式统一原则

设计数据库首先要确定其具有规范化特点，规范化范式具有多样化的特点，通常情况下选择的类型为3NF。然而，较高的范式级别并不意味着效果也更加理想，我们可以采用增加数据量来提高数据库性能，具体如何实现还应在进一步考虑。

(2)命名原则

应该选取通俗易懂的文字在表示，如果有的名字使英文单词，可以选择用首字母缩写。名字应该保证精简字数不能太长。牢记一点数据重要的内容则都是英文如果出现中文则系统无法进行识别，最后导致出现乱码情况。这一点应该着重注意。

(3)保证数据完整

一个可以独立运行的程序系统，数据的完整是它的运行保证，我们不应盲目追求功能的作用，随意添加与数据库无法匹配的数据，应该采用试错法进行逐步分析功能，做到每一个数据都有它的价值。

(4)数据增加功能

数据库构建的初期，应该做好必要的功课，对数据库影响程序的开发进行预判，了解该程序未来发展的空间与前景，做到可以根据用户意见做到增删数据。

### **2．数据库E-R图构架原理**

数据库的E-R图是数据库中实体互相联系的图表，构成ER图的基本要素就是实体、属性与关系。实体是指客观存在并可相互区分的事问特；可以清晰的展现出架构的流程。

数据库的概念设计实体属性E-R图如图3-1所示：



图3-1管理员信息实体E-R图

此系统采用的是NODE的后端开发技术，采用的是MYSQL数据库，通过数据库创建人事管理系统所需的数据表，导入员工的基本信息，创建表，然后在NODE中，通过监听调用的方法，通过SQL查询，添加，修改，删除等指令对数据库进行操作，实现人事管理系统的基本功能。保存的数据是存在安全性的，所以需要对数据库加密，登录数据库是需要权限的，以防止数据信息被盗，造成不必要的损失。

人事管理的员工信息表。如图3-1-1所示。

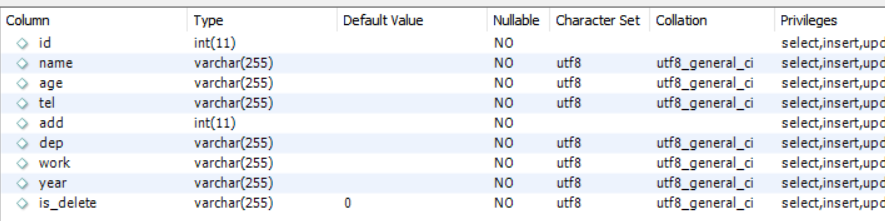


图3-1-1 员工信息表

员工考勤和工资表.如图3-2所示。

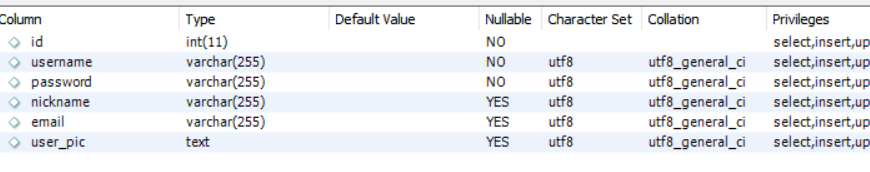


图3-2 管理员信息表

基础信息表.如图3-3所示。

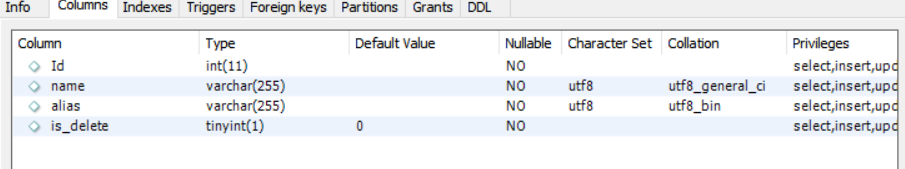


图3-3 员工资料表

## **详细设计**

**1.登录界面设计**

每位管理员都有属于自己的账号密码，当输入正确的账号密码后就会进行跳转，进入到后台的主页面，然后管理员可以进行管理的一系列操作行为，拥有管理员权限，可以对员工的各项信息和数据进行操作和修改，也可以修改自己的密码和头像等。如图5-1所示。

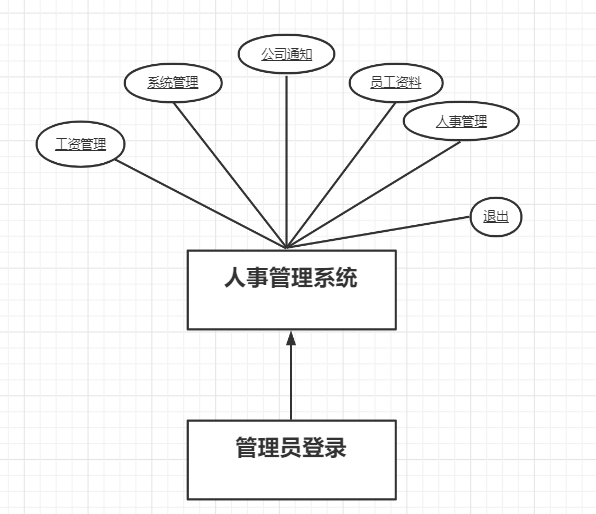
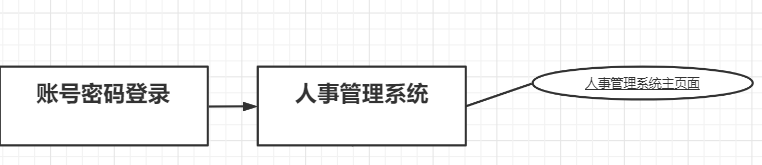


图5-1 人事管理E-R图

**2.主页面设计**

当管理员输入正确的账号密码以后就会进入主页面，可以查看公司的整体情况，也可以进行员工的操作。如图5-2所示。



如图5-2 主页面E-R图

1. **员工信息页面设计**

当管理员登录以后，才有权限去查看和修改员工的基本信息。如图5-3所示。

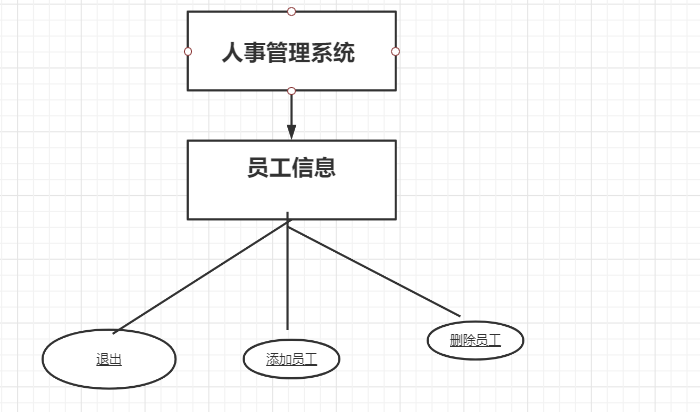


图5-3 员工信息E-R图

**4.人事管理页面设计**

当管理员登录以后，可以对公司员工的考勤管理，调动管理，培训管理，调薪管理等进行查看和操作。如图5-4所示。

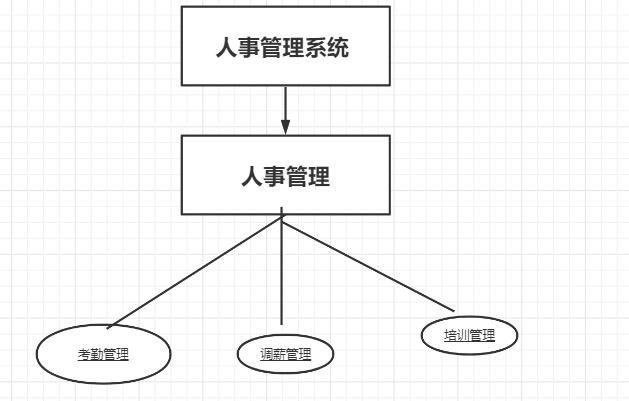
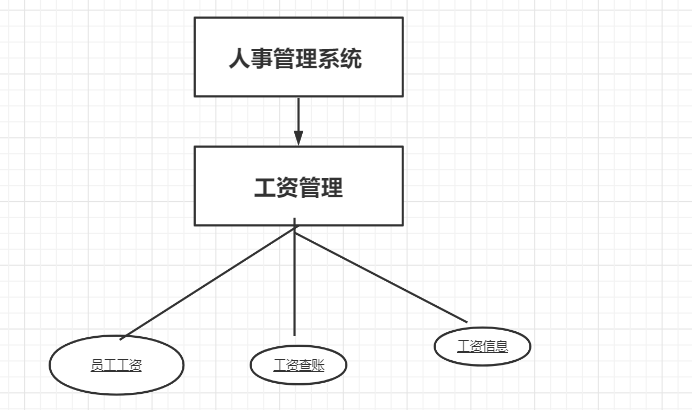


图5-4 员工考勤E-R图

**5.工资管理页面设计**

当管理员登录以后，可以对公司员工的工资进行查看和管理。如图5-5所示。



# 五、系统的实现

## **（一）各项功能实现**

**1．登录功能展示**

第一步打开软件，通过浏览器双击进入登录页面。然后输入正确的账号密码即可进入到主页面。如图6-1所示。



图6-1 登录页面展示图

**2.主页面功能展示**

当输入正确的账号密码以后，就会直接进入到后台的主页面，左边是一些系统的功能，主页面展现的是公司的一些基本信息。如图6-2所示。

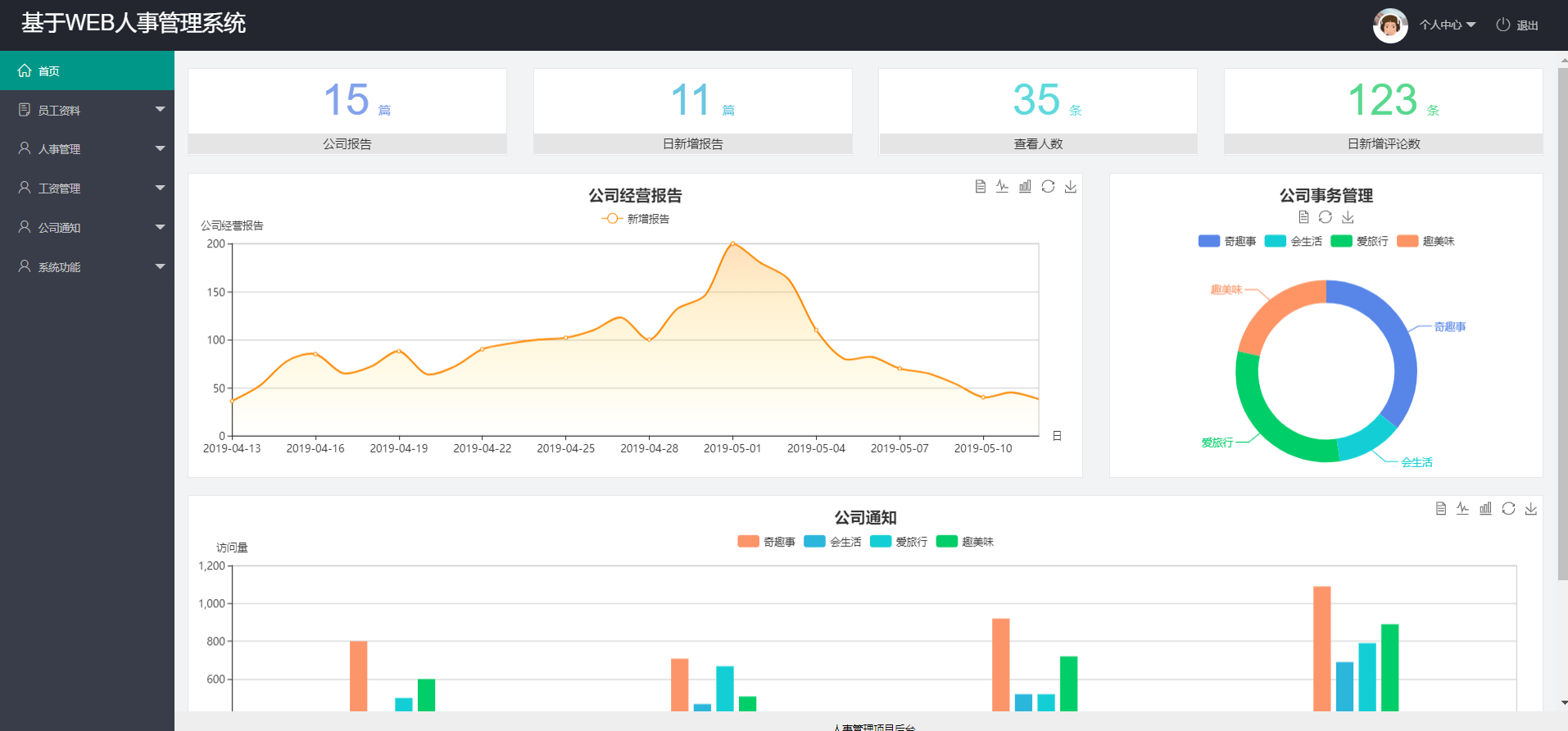
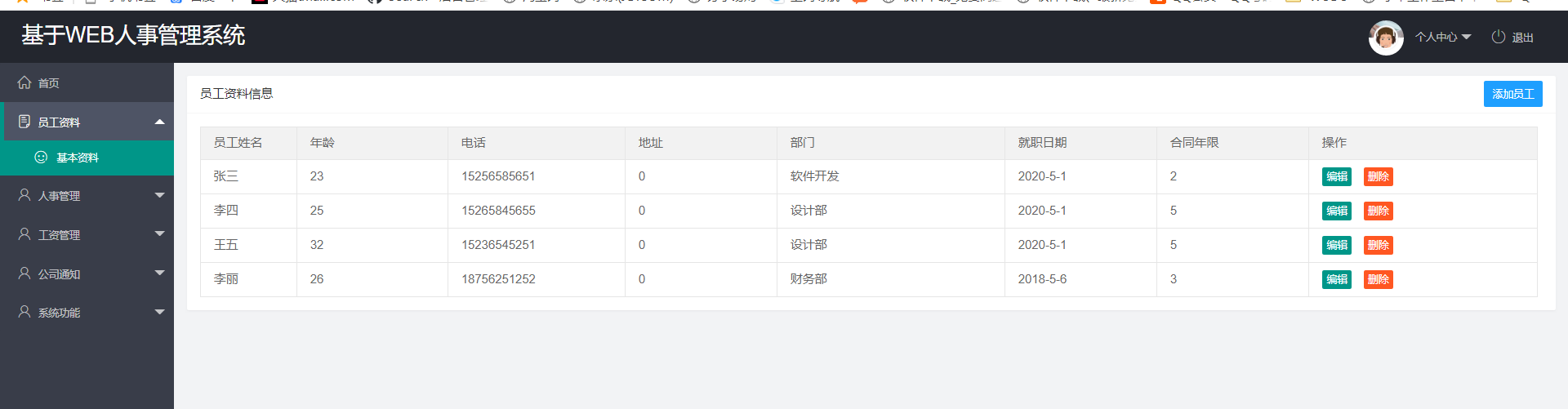


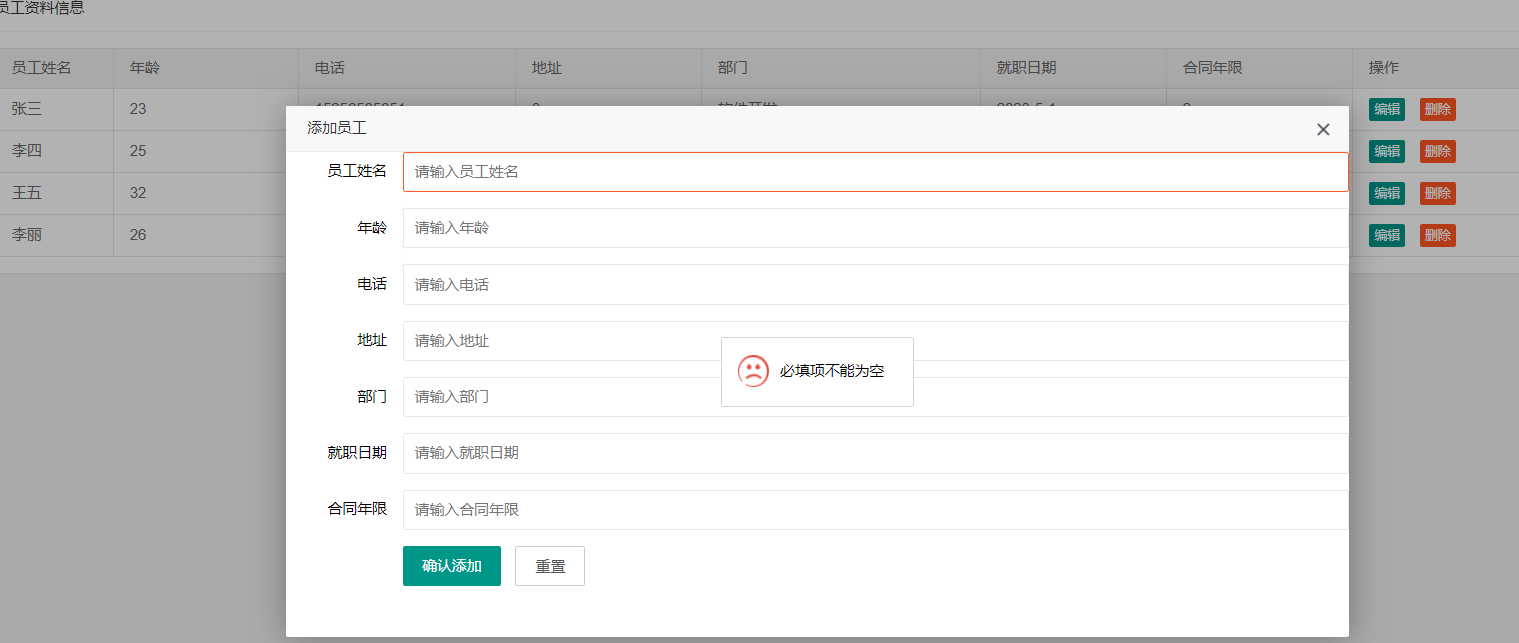
图6-2

1. **员工资料的展示**



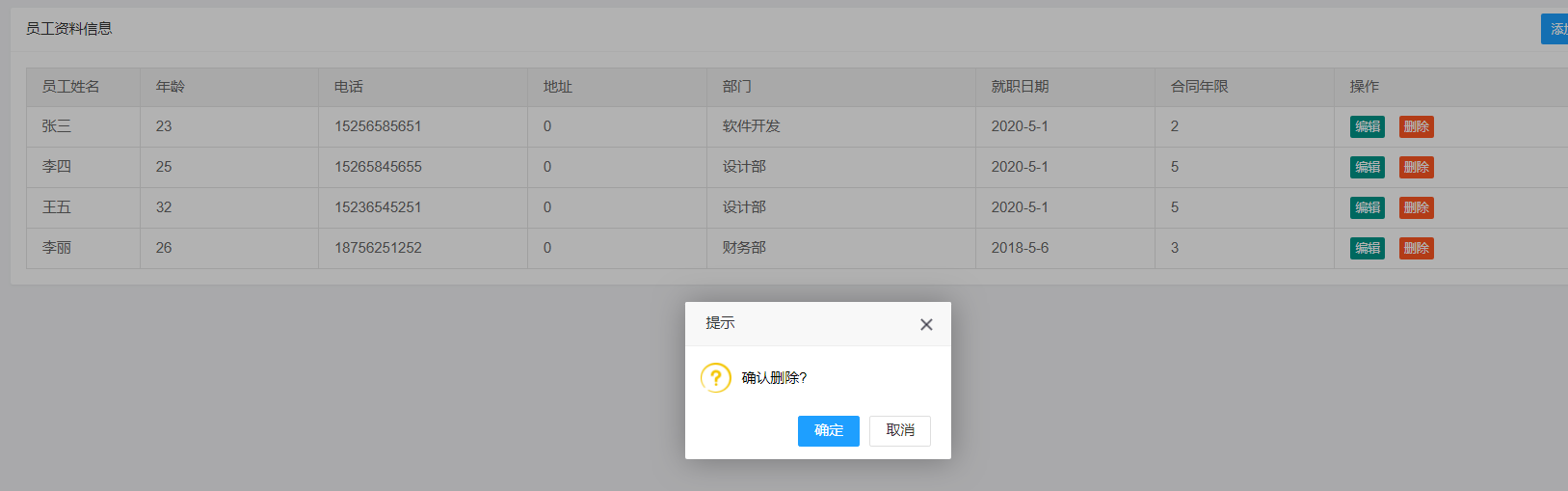
如6-3 主页面展示图

**4.添加员工的展示**



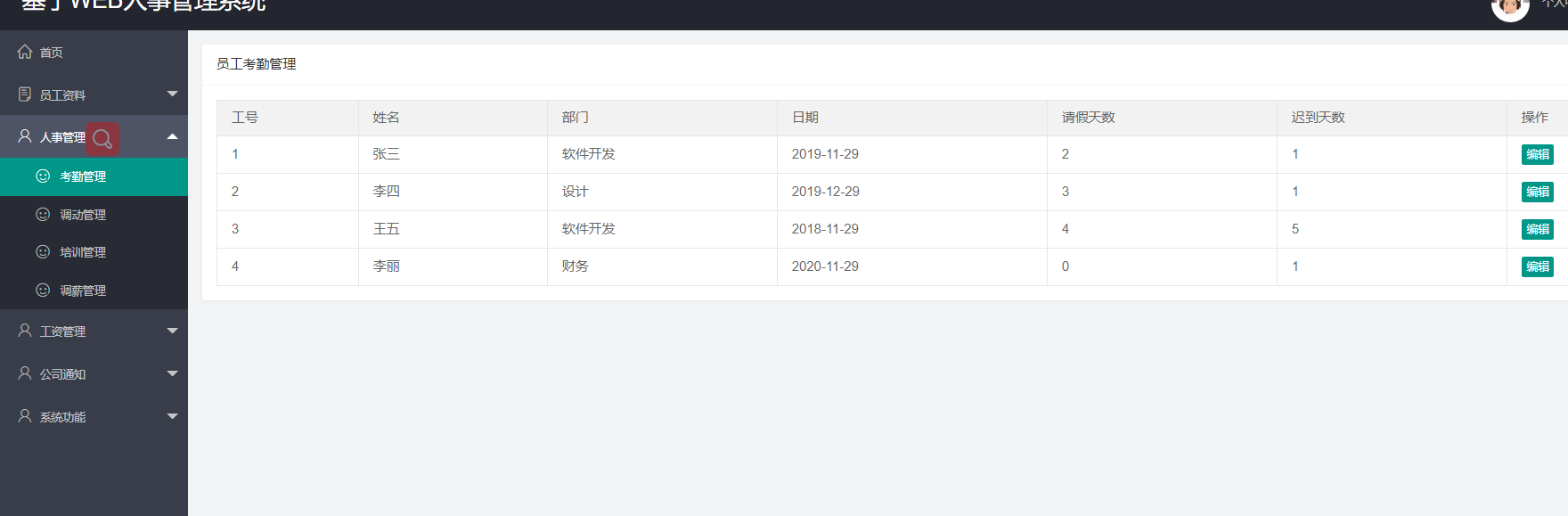
如6-4 添加员工图

1. **删除员工的展示**



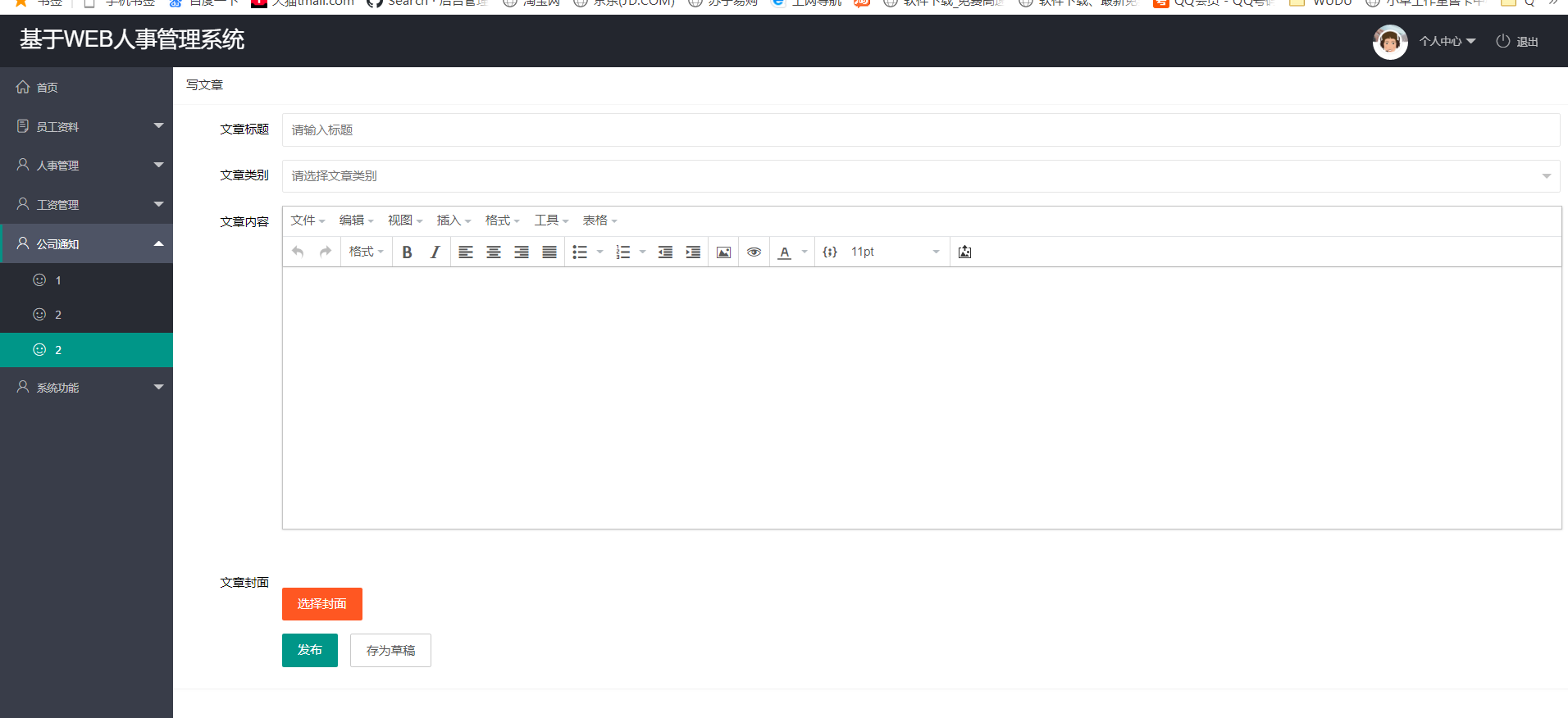
如6-5 删除员工图

**6.员工考勤的展示**



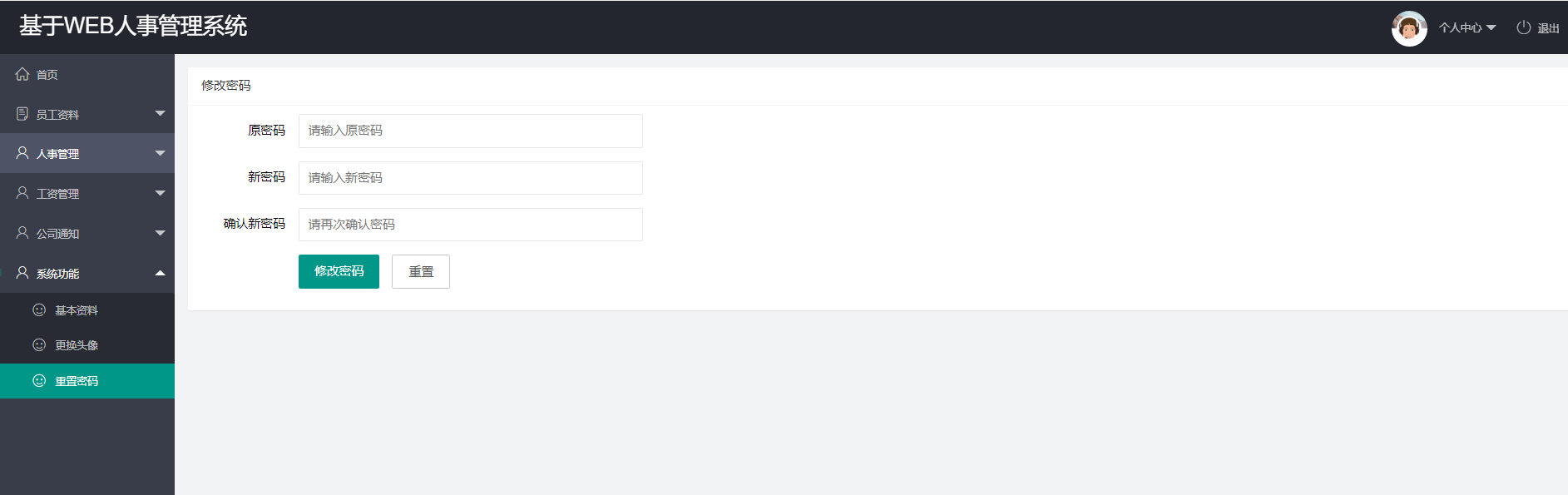
如6-6 员工考勤图

1. **发布公司通知的展示**



如6-7 公司发布通知图

1. **重置密码的功能的展示**



如6-8 重置密码图

# 系统的测试

## **（一）系统测试目的与意义**

人事管理系统的成功与否最重要的是系统的测试环节，可以说系统的测试非常重要，他决定了我此次程序开发的质量好坏，可以让我清楚的了解是否满足用户的要求，通过进一步测试让我更清晰的了解我所设计的程序中的不足与缺点，从而及早的进行查漏补缺，降低用户使用过程中的风险。测试不仅仅验证系统程序的流程性与是否能到达预期的效果，还有一个重要的工作是查找软件系统的逻辑漏洞，检查输入的数据是否可以完整的运行，提高鉴别能力，保障系统始终处于高水平高性能的工作状态当中，降低出错误的问题。遇到问题，还可以及早的进行补救。反复的测试是对自己编写程序的一种责任，要保证程序的正常执行，保证数据的完整性，因为如果是程序出现问题，那么可能造成公司不可估量的损失，现在是个大数据时代，数据就是价值，如果数据库不能正常运行，耽搁的就是公司的利益，这样肯就会造成很大的损失，所以我们要保证自己开发的程序拥有较高的质量，这样让公司和自己都能放心，也保障了客户的利益。

## **（二）测试过程**

首先打开PC端测试，打开数据库的服务，保证数据库的监听和服务正常运行，然后打开自己编写的程序，打开API接口，通过Node进行对程序的执行，执行node app.JS执行后端的程序，然后打开自己的程序，输入账号密码，测试其各个功能，对各个模块的信息进行查询，验证数据的正确与否，各个功能解决以后，用手机端打开进行测试，通过账号密码进行登录查询，最近进行整理，通过文档的形式进行总结。

表5-1 测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块名称 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 登录模块 | 用户名：null 密码：null | 弹出错误提示，请输入用户名 | 登陆失败，提示请输入用户名 | 通过 |
| 登录模块 | 用户名：admin  密码：null | 弹出错误提示，请输入密码 | 失败，提示输入密码 | 通过 |
| 登录模块 | 用户名：admin  密码：1 | 弹出错误提示，用户名或者密码错误 | 登陆失败，提示用户名或者密码错误 | 通过 |
| 登录模块 | 用户名：admin  密码：123456 | 弹出验证成功提示 | 登陆成功 | 通过 |
| 添加员工信息模块 | 输入员工信息后提交 | 添加成功 | 添加成功 | 通过 |
| 修改员工信息模块 | 点击编辑，修改员工信息后提交 | 修改成功 | 修改成功 | 通过 |
| 删除员工信息模块 | 选择员工信息，点击删除 | 删除成功 | 删除成功 | 通过 |

## **（三）测试结果分析**

由于缺乏经验，是第一次独立进行程序的开发在编写代码时存在一些不明显的系统缺陷。在编写代码时，遇到的问题最多是理论与实际相冲突的地方。比如，代码编写不规范就会出现系统无法执行出现报错，并且功能不能实现。最后经过了几次的测试以后，服务器和数据库都能完美运行，也保证了最基本的模块功能，加载速度都进行了反复的测试，人事管理系统的基本功能已能正常运行，在不同环境下，只需要有浏览器，就能进行测试，各个功能执行正常，管理员的登录退出正常，员工信息模块的展示正常，员工的工资模块正常，员工的考勤模块正常，测试结果都为正常，人事管理系统功能合格。

**参考文献**

[1]面向软件开发实践能力的《Java程序设计》教学资源建设[J].张峰,刘迪,仇丽青,赵卫东.软件工程.2020(04)

[2]事业单位人事劳资管理策略分析[J].马江燕.中国标准化. 2019(16)

[3]大数据时代西部高校人事档案管理系统信息化建设探究——以六盘水师范学院为例[J].王月霞,韩世明.办公室业务.2019(11)

[4]关于高校人事管理信息化建设的思考[J].李允奇,海浩.人力资源.2019(06)

[5]高校人事管理系统的设计与实现[J].吴三柱.信息通信.2019(01)

[6]高校人事管理信息系统的研究[J].陆玉发,范开兵,陈洪磊.教育现代化.2018(51)

[7]高校人事管理系统的设计与实现[J].张海,陈延东.产业与科技论坛.2017(14)

[8]高校人事管理信息化建设的探讨[J].关发辉.教育观察(上半月).2015(07)

[9]高校综合人力资源管理系统的数据库优化策略研究[J].王强.电脑知识与技术. 2010(10)

[10]企业人力资源开发与管理的现状与对策[J].秦辉.现代商业.2009(21)

[11]CNNIC发布第44次《中国互联网络发展状况统计报告》[J].于朝晖.网信军民融合. 2019(09)

[12]基于Node.js的教育技术学资源网站的设计与开发[J].姚佳花,彭楚瑶.无线互联科技.2019(09)

[13]计算机应用软件的需求分析与开发[J].徐鉴.数字技术与应用.2019(04)

[14]基于HTML5＿Canvas的3D粒子图形动画的设计[J].王青.中国科技信息.2019(05)

[15]浅谈数据库技术发展趋势[J].高瞻,王飞鹏.黑龙江科技信息.2016(28)

[16]Node.js中Express框架路由机制的研究[J].程桂花,沈炜,何松林,张珂杰.工业控制计算机.2016(08)

[17]Node.js服务器技术初探[J].骆文亮.无线互联科技.2014(03)

[18]服务器端javascript技术研究[J].高原.信息与电脑(理论版). 2012(01)

**致 谢**

一晃几个月的时间过去了，在这里，我要感谢我的论文指导老师，在我的论文写作期间，老师对我进行了长时间的精选辅导，给我指导性的意见，还提出了更多宝贵的意见，对于这次的论文，老师是投入了相当多的心血和努力，在这段时间里，我自己也学到了很多东西，一开始，自己感觉不能成功完成这个论文，因为自己的知识实在太少了，需要花很多时间去努力，去查阅资料，慢慢的在老师和同学的帮助下，我完成了论文的设计，在这四年的本科学习期间，我非常的感谢老师的帮助，从论文的选题，然后编写开题报告，到最后的论文编写完成，是非常困难的，也非常的感谢帮助我的同学和老师。

在此我要感谢计算机学院的老师，是你们的辛勤付出，才让我们学习到更多的知识，在这次的论文写作中，我受益匪浅，我从一开始的文档编写到代码的基本编写，以及软件工程图，这都是一个非常长的过程，也是非常不容易的，最后我也要感谢学校在过去的几年里为投入的精力，学校的老师们孜孜不倦地为我们授课和辅导，使我们能够更好地掌握专业知识，提高我们处理实际问题的能力。最终运用这些知识完成我的毕业设计，我也想向你们说一句谢谢你们老师，你们辛苦了，这次的程序将成为我难忘的经历。

**附录A**

程序主要源代码如下：

前台页面源代码：

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>人事管理系统后台</title>

<!-- 导入layui的样式 -->

<link rel="stylesheet" href="assets/lib/layui/css/layui.css">

<!-- 导入自己的样式 -->

<link rel="stylesheet" href="assets/css/login.css">

</head>

<body>

<!--头部LOGO区域-->

<div class="layui-main"><!-- <img src="assets/images/logo.png" alt=""> --></div>

<!--登录注册区域-->

<div class="loginAndRexBox">

<div class="title-box">

</div>

<!--登录的div-->

<div class="login-box">

<!--登陆的表单-->

<form class="layui-form" id="form\_login">

<!--登陆-->

<div class="layui-form-item">

<i class="layui-icon layui-icon-username"></i>

<input type="text" name="username" required lay-verify="required" placeholder="请输入用户名" autocomplete="off" class="layui-input">

</div>

<div class="layui-form-item">

<i class="layui-icon layui-icon-password"></i>

<input type="password" name="password" required lay-verify="required|pwd" placeholder="请输入密码" autocomplete="off" class="layui-input">

</div>

<!--按钮-->

<div class="layui-form-item">

<!--表单提交按钮和普通按钮的区别就是lay-submit属性 而且layui-btn-fluid独占一行-->

<div class="anniu">

<button class="layui-btn layui-btn-normal" lay-submit lay-filter="formDemo">登陆</button>

<button type="reset" class="layui-btn layui-btn-primary">重置</button>

</div>

</div>

<!-- <div class="layui-form-item links">

<a href="javascript:;" id="link\_reg">去注册账号</a></div> -->

</form>

</div>

<!--注册的div-->

<div class="reg-box">

<!-- 注册的表单 -->

<form class="layui-form" id="form\_reg">

<!--注册用户名-->

<div class="layui-form-item">

<i class="layui-icon layui-icon-username"></i>

<!-- lay-verify="required"表单验证的选项 -->

<input type="text" name="username" required lay-verify="required" placeholder="请输入用户名" autocomplete="off" class="layui-input">

</div>

<!--注册密码-->

<div class="layui-form-item">

<i class="layui-icon layui-icon-password"></i>

<input type="password" name="password" required lay-verify="required|pwd" placeholder="请输入密码" autocomplete="off" class="layui-input"></div>

<!--注册确认密码-->

<div class="layui-form-item">

<i class="layui-icon layui-icon-password"></i>

<input type="password" name="repassword" required lay-verify="required|pwd|repwd" placeholder="再次密码确认" autocomplete="off" class="layui-input"></div>

<!--注册按钮-->

<div class="layui-form-item">

<!--表单提交按钮和普通按钮的区别就是lay-submit属性 而且layui-btn-fluid独占一行-->

<button class="layui-btn layui-btn-fluid layui-btn layui-btn-normal" lay-submit lay-filter="formDemo">注册</button>

</div>

<div class="layui-form-item links">

<a href="javascript:;" id="link\_login">去登录</a>

</div> </form> </div>

前台首页源代码：

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>后台主页</title>

<link rel="stylesheet" href="assets/lib/layui/css/layui.css">

<!--导入layui的样式表-->

<link rel="stylesheet" href="assets/css/index.css">

<!--导入自己的样式-->

<link rel="stylesheet" href="assets/fonts/iconfont.css">

</head>

<body class="layui-layout-body">

<div class="layui-layout layui-layout-admin">

<div class="layui-header">

<span class="renshi">基于WEB人事管理系统</span>

<!-- <div class="layui-logo">

<img src="assets/images/logo.png" alt=""></div> -->

<!-- 头部区域（可配合layui已有的水平导航） -->

<ul class="layui-nav layui-layout-right">

<li class="layui-nav-item">

<a href="javascript:;" class="userinfo">

<span class="text-avatar"> </span>

<img src="http://t.cn/RCzsdCq" class="layui-nav-img"> 个人中心

</a>

<dl class="layui-nav-child">

<dd><a href="user/user\_info.html" target="fm"><i class="layui-icon layui-icon-app"></i> 基本资料</a></dd>

<dd><a href="user/user\_avatar.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 更换头像</a></dd>

<dd><a href="user/user\_pwd.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 重置密码</a></dd>

</dl>

</li>

<li class="layui-nav-item">

<a href="javascript:; " id="btnLogout"> <span class="iconfont icon-tuichu"></span>退出</a>

</li>

</ul>

</div>

<div class="layui-side layui-bg-black">

<div class="layui-side-scroll">

<!-- <div class="userinfo">

<img src="http://t.cn/RCzsdCq" class="layui-nav-img">

<span class="text-avatar"> </span>

<span id="welcome"></span>

</div> -->

<!-- 左侧导航区域（可配合layui已有的垂直导航） -->

<ul class="layui-nav layui-nav-tree" lay-filter="test" lay-shrink="all">

<!-- lay-shrink="all"只允许展开一个导航栏 -->

<li class="layui-nav-item layui-this"><a href="home/dashboard.html" target="fm"><span class="iconfont icon-home"></span>首页</a></li>

<li class="layui-nav-item">

<a class="" href="javascript:;"> <span class="iconfont home

icon-16"></span>员工资料</a>

<dl class="layui-nav-child">

<!-- <dd><a href="article/art\_cate.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 基本资料</a></dd> --><dd><a href="article/ygzl.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 基本资料</a></dd>

<!-- 利用layui的图标自定义，后面的才是控制图标的 -->

<!-- <dd><a href="article/art\_list.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 文章列表</a></dd>

<dd><a href="article/art\_pub.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 发布文章</a></dd> -->

</dl>

</li>

<li class="layui-nav-item">

<a href="javascript:;"><span class="iconfont user

icon-user"></span>人事管理</a>

<dl class="layui-nav-child">

<dd><a href="article/kqgl.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 考勤管理</a></dd>

<dd><a href="article/ddgl.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 调动管理</a></dd>

<dd><a href="article/pxgl.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 培训管理</a></dd>

<dd><a href="article/txgl.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 调薪管理</a></dd>

</dl>

</li>

<li class="layui-nav-item">

<a href="javascript:;"><span class="iconfont user

icon-user"></span>工资管理</a>

<dl class="layui-nav-child">

<dd><a href="article/grgz.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 员工工资</a></dd>

<!-- <dd><a href="article/gsgz.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 公司账套</a></dd> -->

</dl>

</li>

<li class="layui-nav-item">

<a href="javascript:;"><span class="iconfont user

icon-user"></span>公司通知</a>

<dl class="layui-nav-child">

<dd><a href="article/art\_cate.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 1</a></dd>

<dd><a href="article/art\_list.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 2</a></dd>

<dd><a href="article/art\_pub.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 2</a></dd>

</dl>

</li>

<li class="layui-nav-item">

<a href="javascript:;"><span class="iconfont user

icon-user"></span>系统功能</a>

<dl class="layui-nav-child">

<dd><a href="user/user\_info.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 基本资料</a></dd>

<dd><a href="user/user\_avatar.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 更换头像</a></dd>

<dd><a href="user/user\_pwd.html" target="fm"><i class="layui-icon layui-icon-face-smile"></i> 重置密码</a></dd>

</dl>

</li>

</ul>

</div>

</div>

<div class="layui-body">

<!-- 内容主体区域 -->

<iframe src="home/dashboard.html" name="fm" frameborder="0"></iframe>

<!-- src home/dashboard.html 默认打开就是这个主页 后台的内容主体 -->

</div>

<div class="layui-footer">

<!-- 底部固定区域 -->

人事管理项目后台

</div>

</div>

后端源代码：

// 导入 express 模块

const express = require('express')

// 创建 express 的服务器实例

const app = express()

// 导入并注册用户路由模块

const joi = require('@hapi/joi')

// 导入 cors 中间件

const cors = require('cors')

// 将 cors 注册为全局中间件

app.use(cors())

app.use(express.urlencoded({ extended: false }))//配置表单解析数据的中间件

// 响应数据的中间件，需要在导入路由之前

app.use(function (req, res, next) {

// status = 0 为成功； status = 1 为失败； 默认将 status 的值设置为 1，方便处理失败的情况

res.cc = function (err, status = 1) {

res.send({

// 状态

status,

// 状态描述，判断 err 是 错误对象 还是 字符串

message: err instanceof Error ? err.message : err,

})

}

next()

})

// 导入配置文件 一定要在路由配置解析token中间件

const config = require('./config')

// 解析 token 的中间件

const expressJWT = require('express-jwt')

// 使用 .un less({ path: [/^\/api\//] }) 指定哪些接口不需要进行 Token 的身份认证

app.use(expressJWT({ secret: config.jwtSecretKey }).unless({ path: [/^\/api\//] }))

//path: [/^\/api\//] }排除api接口开头的中间件

//导入员工信息模块

const ygzl = require('./router/ygzl')

// 为文章分类的路由挂载统一的访问前缀 /my/article

app.use('/my', ygzl)

//导入考勤模块

const attendance = require('./router/attendance')

app.use('/my', attendance)

//导入路由模块

const userRouter = require('./router/user')

app.use('/api', userRouter)

// write your code here...

// 导入并使用用户信息路由模块

const userinfoRouter = require('./router/userinfo')

// 注意：以 /my 开头的接口，都是有权限的接口，需要进行 Token 身份认证

app.use('/my', userinfoRouter)

// 导入并使用文章分类路由模块

const artCateRouter = require('./router/artcate')

// 为文章分类的路由挂载统一的访问前缀 /my/article

app.use('/my/article', artCateRouter)

// 导入并使用文章路由模块

const articleRouter = require('./router/article')

// 为文章的路由挂载统一的访问前缀 /my/article

app.use('/my/article', articleRouter)

// 托管静态资源文件

app.use('/uploads', express.static('./uploads'))

// 错误中间件

app.use(function (err, req, res, next) {

// 数据验证失败

if (err instanceof joi.ValidationError) return res.cc(err)

// 未知错误

// 捕获身份认证失败的错误

if (err.name === 'UnauthorizedError') return res.cc('身份认证失败！')

// 未知错误...

return res.cc(err)

})

// 调用 app.listen 方法，指定端口号并启动web服务器

app.listen(8888, function () {

console.log('api server running at http://127.0.0.1:8888')

})

后端程序源代码：

const express = require('express')

const router = express.Router()

// 导入用户路由处理函数模块

const userHandler = require('../router\_handler/user')

// 1. 导入验证表单数据的中间件

const expressJoi = require('@escook/express-joi')

// 2. 导入需要的验证规则对象

const { reg\_login\_schema } = require('../schema/user')

// 注册新用户

// 3. 在注册新用户的路由中，声明局部中间件，对当前请求中携带的数据进行验证

// 3.1 数据验证通过后，会把这次请求流转给后面的路由处理函数

// 3.2 数据验证失败后，终止后续代码的执行，并抛出一个全局的 Error 错误，进入全局错误级别中间件中进行处理

router.post('/reguser', expressJoi(reg\_login\_schema), userHandler.regUser)

// 登录

router.post('/login', expressJoi(reg\_login\_schema),userHandler.login)

module.exports = router

员工工资源代码：

// 导入 express

const express = require('express')

// 创建路由对象

const router = express.Router()

// 获取员工信息

const ygzl=require('../router\_handler/rsgl/ygzl')

//挂在路由attendance 考勤

router.get('/ygzl', ygzl.ygzl)

//新增员工信息

router.post('/inygzl',ygzl.inygzl)

//根据ID删除员工信息

router.get('/sygzl/:id',ygzl.sygzl)

//根据ID获取员工信息

router.get('/hygzl/:id',ygzl.hygzl)

//更新员工信息

router.post('/gygzl',ygzl.gygzl)

// 向外共享路由对象

module.exports = router