目录

摘要 2

1.绪论 3

1.1 研究背景 3

1.2 研究意义 3

2.需求分析 4

3.系统分析 5

4.概要设计 6

5.数据库设计 8

5.1 数据库ER总图 8

5.2 数据库逻辑结构设计 8

6.详细设计与实现 10

6.1 用户认证和退出登录模块的设计与实现 10

6.2 用户管理模块的设计与实现 12

6.3 部门管理模块的设计与实现 13

6.4 职位管理模块的设计与实现 14

6.5 员工管理模块的设计与实现 15

6.6 公告管理模块的设计与实现 16

6.7 下载模块的设计与实现 17

7.系统测试 18

8.总结 20

致谢 21

参考文献 22

摘要

在科技高速发展的今天，信息的自动化管理是信息管理的发展方向，而在其中，不管是国家政府，军队还是企业的建设又避不开一个能有效管理人力资源的人事管理系统。这次设计就是要实现一个后端使用java语言的Spring+SpringMVC+Mybitis框架+Mysql数据库，前端使用Ant design框架的人事管理系统。该系统使用SpringMVC来处理前端View到后端Controller之间的映射,使用Spring的IOC容器来管理持久层组件,业务逻辑组件,以及控制层组件,充分利用了Spring的依赖注入的优势,增强系统的解耦性,从而提高应用的可扩展性,降低系统重构的成本，使用Mybitis来面向对象的管理持久层进行数据库操作，层次分明，性能优秀并且简化了代码的编程，使用Ant Design来设计前端视图,由于Ant Design是基于React的组件式的框架,能像后端一样清晰的表明视图的结构,并且组件式的设计能很好的提高代码的可复用性,Ant Design提供了丰富的组件供给我选择,降低了代码编程的成本。

关键字：Spring,SpringMVC,Mybitis,Ant Design,人事管理

1.绪论

1.1研究背景

现在，在全球范围内，普遍使用了人事管理系统来管理人力资源，同时也出现了基于各行各业的实现各种不同需求的人事管理系统，其中有Client/Server也有Brower/Server还有其他不同的模式。随着网络的快速发展和普及，基于Web的信息管理系统成为了信息管理系统研究的主流。一个好的人事管理系统能方便的高效的管理好人力资源，我的目标就是开发一个基于B/S架构的,后端使用SSM框架,前端使用Ant Design的人事管理系统。

1.2研究意义

在信息与计算机网络技术高速发展的今天，基于计算机应用的信息管理系统必然会取代传统的以纸为媒介的管理方式。不管是国家单位还是企事业单位，在信息越来越复杂的今天，使用计算机应用来管理各种事务是不可避免的而且是顺应时代潮流的。人事管理系统作为信息管理系统的一种，对人力资源的管理起着重要的作用。

1. 需求分析

（1）用户认证与退出登录模块

该模块实现对用户的认证功能以及用户退出登录的功能,认证通过则跳转到系统主界面,认证失败则返回登录界面,登出后跳转到登录界面。

（2）用户管理模块

该模块实现对用户基本信息的管理,包括用户的用户名，密码和权限（超级管理员和普通用户）等。

（3）部门管理模块

该模块实现对部门信息的管理,通过该模块能对部门信息进行增删改查操作。

（4）职位管理模块

该模块实现对职位信息的管理,通过该模块能对职位信息进行增删改查操作。

（5）员工管理模块

改模块实现对员工信息的管理,包括员工的姓名，性别，年龄，手机，邮箱，地址，身份证，部门和职位等

（6）公告管理模块

该模块实现对公告的管理，通过该模块能对公告进行增删改查操作。

（7）下载中心

该模块实现了文件的上传，下载和删除功能，可以根据文件名称进行模糊查询。

3.系统分析

3.1经济可行性分析

实现本次开发项目，需要软硬件两方面的支持。硬件方面值需要一台电脑作为开发服务器即可；软件方面，只需要MyExlipse10，Mysql数据库和浏览器即可。经济成本低，在经济上是可行的。

3.2技术可行性分析

本系统是基于Windows操作系统环境下的,采用了Java语言，Mysql数据库进行开发设计。Java语言具有面向对象，解释型，分布式，简单性以及安全可靠等优点，是开发者使用最多的编程语言，本次设计实现的系统为小型系统，Mysql数据库的体积小、速度快以及总体拥有成本低等优点特别适合用作本系统数据库。在计算机技术高速发展的今天，普通的笔记本电脑就能运行此系统，在技术上是可行的。

3.3操作可行性

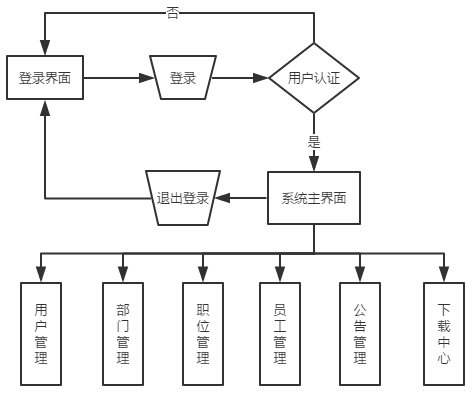
本系统界面简洁，层次分明，各种功能一目了然,哪怕没有计算机基础的也能很快的上手操作本系统。所以本系统具有操作可行性。

4.概要设计

4.1开发环境

系统开发需要配置软件环境，本系统需要配置的软件环境主要包括JDK1.8,Myexlipse10以及Mysql数据库。

4.2系统业务流程



浏览器打开登录界面,输入用户名和密码，点击登录进行用户认证，认证通过则跳转到系统主界面，失败则在登录界面弹出失败提示框。

进入系统主界面后，有退出登录，用户管理，部门管理，职位管理，员工管理，公告管理和下载中心等功能模块：

（1）点击退出登录则注销token并且跳转到登录界面。

（2）点击用户管理,则在页面右侧显示全部用户信息，点击添加按钮会弹出一个模态框，输入要添加的用户信息后点击提交后若通过数据校验则成功添加一个新的用户。在需要编辑的用户那行点击编辑按钮后就能对该行进行编辑，编辑后点击提交若通过数据校验则成功更新改用户信息。在所有需要删除的用户那行的checkbox勾上勾,然后点击删除按钮就能批量删除用户。点击列参数名字旁的搜索图标，弹出一个输入框，输入所需要查找的数据后点击确定就能只显示出筛选后的数据。

（3）点击部门管理,则在页面右侧显示全部部门信息，点击添加按钮会弹出一个模态框，输入要添加的部门信息后点击提交后若通过数据校验则成功添加一个新的部门。在需要编辑的部门那行点击编辑按钮后就能对该行进行编辑，编辑后点击提交若通过数据校验则成功更新改部门信息。在所有需要删除的部门那行的checkbox勾上勾,然后点击删除按钮就能批量删除部门。点击列参数名字旁的搜索图标，弹出一个输入框，输入所需要查找的数据后点击确定就能只显示出筛选后的数据。

（4）点击职位管理,则在页面右侧显示全部职位信息，点击添加按钮会弹出一个模态框，输入要添加的职位信息后点击提交后若通过数据校验则成功添加一个新的职位。在需要编辑的职位那行点击编辑按钮后就能对该行进行编辑，编辑后点击提交若通过数据校验则成功更新改职位信息。在所有需要删除的职位那行的checkbox勾上勾,然后点击删除按钮就能批量删除职位。点击列参数名字旁的搜索图标，弹出一个输入框，输入所需要查找的数据后点击确定就能只显示出筛选后的数据。

（5）点击员工管理,则在页面右侧显示全部员工信息，点击添加按钮会弹出一个模态框，输入要添加的员工信息后点击提交后若通过数据校验则成功添加一个新的员工。在需要编辑的员工那行点击编辑按钮后就能对该行进行编辑，编辑后点击提交若通过数据校验则成功更新改员工信息。在所有需要删除的员工那行的checkbox勾上勾,然后点击删除按钮就能批量删除员工。点击列参数名字旁的搜索图标，弹出一个输入框，输入所需要查找的数据后点击确定就能只显示出筛选后的数据。

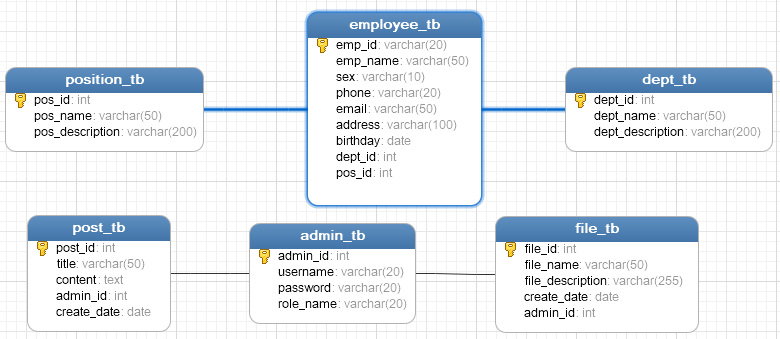
（6）点击公告管理,则在页面右侧显示全部公告信息，点击添加按钮会弹出一个模态框，输入要添加的公告信息后点击提交后若通过数据校验则成功添加一个新的公告。在需要编辑的公告那行点击编辑按钮后就能对该行进行编辑，编辑后点击提交若通过数据校验则成功更新改公告信息。在所有需要删除的公告那行的checkbox勾上勾,然后点击删除按钮就能批量删除公告。点击列参数名字旁的搜索图标，弹出一个输入框，输入所需要查找的数据后点击确定就能只显示出筛选后的数据。

（7）点击下载中心,则在页面右侧显示全部文件信息，点击上传按钮后选择要上传的文件后点击确定后上传文件,上传成功弹出成功提示，失败则弹出失败提示。在需要下载的文件那行点击下载按钮后点击确定即开始下载文件。选择需要删除的文件后点击删除按钮即可删除文件。

5.数据库设计

本系统使用Mysql作为数据库，数据库命名为hrms\_db(Human Resource Manage System DataBase)。

5.1数据库ER总图



5.2数据库逻辑结构设计

（1）admin\_tb---用户表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | 数据类型 | 主键 | 描述 |
| **admin\_id** | Int | 是 | 用户编号 |
| **username** | Varchar(20) | 否 | 用户名 |
| **password** | Varchar(20) | 否 | 密码 |
| **role\_name** | Varchar(20) | 否 | 权限 |

（2）dept\_tb---部门表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | 数据类型 | 主键 | 描述 |
| **dept\_id** | Int | 是 | 部门编号 |
| **dept\_name** | Varchar(50) | 否 | 部门名称 |
| **dept\_description** | Varchar(200) | 否 | 详细描述 |

（3）position\_tb---职位表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | 数据类型 | 主键 | 描述 |
| **pos\_id** | Int | 是 | 职位编号 |
| **pos\_name** | Varchar(50) | 否 | 职位名称 |
| **pos\_description** | Varchar(200) | 否 | 详细描述 |

（4）employee\_tb---员工表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | 数据类型 | 主键 | 描述 |
| **emp\_id** | Varchar(20) | 是 | 员工编号 |
| **emp\_name** | Varchar(50) | 否 | 员工姓名 |
| **sex** | Varchar(10) | 否 | 性别 |
| **phone** | Varchar(20) | 否 | 电话 |
| **email** | Varchar(50) | 否 | 邮箱 |
| **address** | Varchar(100) | 否 | 住址 |
| **birthday** | data | 否 | 生日 |
| **dept\_id** | int | 否 | 部门编号 |
| **pos\_id** | int | 否 | 职位编号 |

（5）post\_tb---公告表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | 数据类型 | 主键 | 描述 |
| **post\_id** | Int | 是 | 公告编号 |
| **title** | Varchar(50) | 否 | 标题 |
| **content** | text | 否 | 内容 |
| **admin\_id** | int | 否 | 发布人 |
| **create\_data** | date | 否 | 发表日期 |

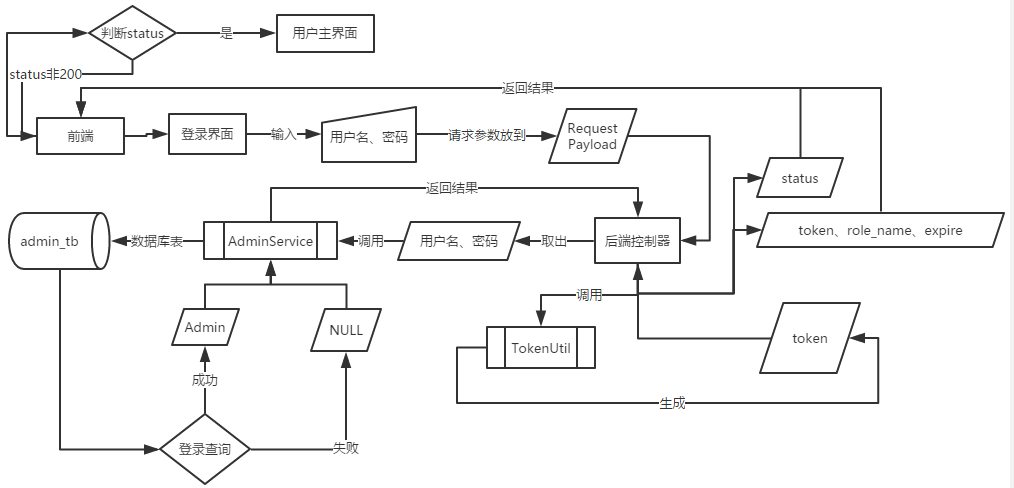
（6）file\_tb---文件表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | 数据类型 | 主键 | 描述 |
| **file\_id** | Int | 是 | 文件编号 |
| **file\_name** | Varchar(50) | 否 | 文件名 |
| **file\_description** | Varchar(255) | 否 | 详细描述 |
| **admin\_id** | int | 否 | 上传用户 |
| **create\_data** | date | 否 | 上传日期 |

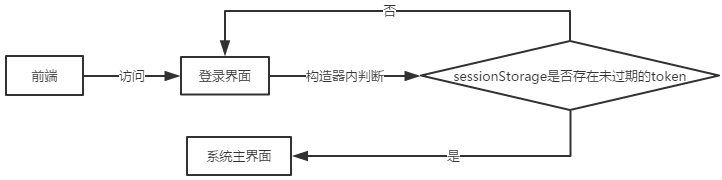
6.详细设计与实现

6.1登录与登出模块的设计与实现

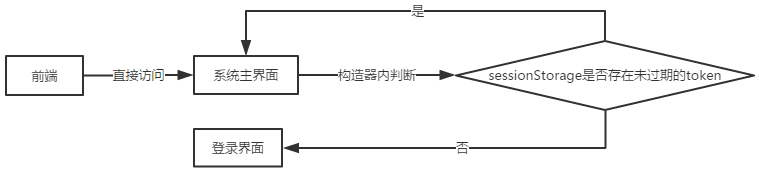
（1）首次登录流程



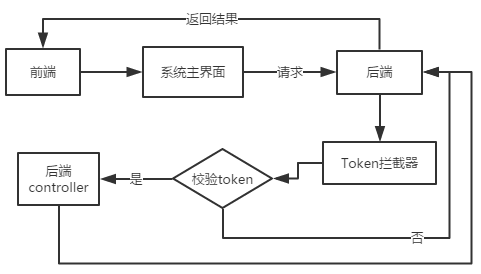
（2）访问登录界面



（3）访问系统主界面



（4）对后端发起请求



（5）退出登录



此次设计采用的是前后端分离的方式,后端和前端只进行数据交换,后端不对视图的跳转进行控制。因此使用token来实现用户认证。

前端提交的表单数据请求到后端controller后，将其绑定到Admin实体，然后调用service层的登录查询方法查询数据库是否有相同的Admin用户，若在数据库查询到相同的用户则调用自定义的TokenUtil的createToken生成一个token，和token过期时间以及查询到的用户权限放到一个map里在转化为String类型后一起reponse到前端,前端收到后把它们以及用户名一起存储到sessionStorage，然后跳转到用系统主界面。

之后在系统主界面的每次对后端发起的请求都要先从sessionStorage查询token是否存在以及利用sessionStorage存储的过期时间判断是否过期,如果token不存在或者当前时间大于过期时间,则视图页面重定向到登录页面,反之请求成功到达后端,在后端也实现了一个自定义的拦截器,对处理登录之外的请求的token进行验证,若token不存在或者签名不一致或者已经过期则返回401状态码,前端收到非200状态码后会重定向到登录页面。

用户每次访问登录页面都会先从sessionStorage查询是否有未过期的token，如果有则自动重定向到系统主页面。

用户每次直接访问系统主界面也会先从sessionStorage查询是否有未过期的token,如果没有则重定向到登录页面。

系统主界面点击退出登录后清空sessionStorage的数据,然后跳转到登录界面。

6.2 用户管理模块的设计与实现

6.3 部门管理模块的设计与实现

6.4 职位管理模块的设计与实现

6.5 员工管理模块的设计与实现

6.6 公告管理模块的设计与实现

6.7 下载模块的设计与实现

7.系统测试