**五邑大学**

**智能制造学部**



**J2EE开发技术报告**

**题 目：基于JavaEE架构设计与开发的人事管理Hrm系统**

**专 业**：电子信息工程（信息安全）

**班 级**：160808

**学 号**：3116003980

**姓 名**：游大满

**指导教师**：　　何芳明

**起止日期**：2019年3月21日 至2019年7月6日

递交的材料清单：

1. 源代码一份(小于10M)
2. 软件设计报告一份(word)
3. 项目演示视频一份

**指导教师审阅意见及成绩评定：**

指导老师签章：

日期： 年 月 日

目录

[基于JavaEE架构设计与开发的人事管理Hrm系统 1](#_Toc13751874)

[一、 **JavaEE三层架构** 1](#_Toc13751875)

[1、 概念简介 1](#_Toc13751876)

[2、 概述 1](#_Toc13751877)

[3、 三层结构原理 1](#_Toc13751878)

[4、 各层的作用 1](#_Toc13751879)

[5、 优缺点 2](#_Toc13751880)

[二、 **实现过程** 2](#_Toc13751881)

[1、 功能介绍 2](#_Toc13751882)

[2、 界面介绍 10](#_Toc13751883)

[3、 完成情况 10](#_Toc13751884)

[三、 **调试运行中碰到的问题及解决办法** 12](#_Toc13751885)

[1、 问题一 12](#_Toc13751886)

[2、 问题二 12](#_Toc13751887)

[四、 **课程学习体会** 13](#_Toc13751888)

**基于JavaEE架构设计与开发的人事管理Hrm系统**

1. **JavaEE三层架构**

三层架构(3-tier architecture) 通常意义上的三层架构就是将整个业务应用划分为：表现层（UI）、业务逻辑层（BLL）、数据访问层（DAL）。区分层次的目的即为了“高内聚，低耦合”的思想。

1. 概念简介
   1. 表现层（UI）

通俗讲就是展现给用户的界面，即用户在使用一个系统的时候他的所见所得。

* 1. 业务逻辑层（BLL）

针对具体问题的操作，也可以说是对数据层的操作，对数据业务逻辑处理。

* 1. 数据访问层（DAL）

该层所做事务直接操作数据库，针对数据的增添、删除、修改、查找等。

1. 概述

在软件体系架构设计中，分层式结构是最常见，也是最重要的一种结构。微软推荐的分层式结构一般分为三层，从下至上分别为：数据访问层、业务逻辑层（又或称为领域层）、表示层。

1. 三层结构原理

3个层次中，系统主要功能和业务逻辑都在业务逻辑层进行处理。

所谓三层体系结构，是在客户端与数据库之间加入了一个“中间层”，也叫组件层。这里所说的三层体系，不是指物理上的三层，不是简单地放置三台机器就是三层体系结构，也不仅仅有B/S应用才是三层体系结构，三层是指逻辑上的三层，即使这三个层放置到一台机器上。

三层体系的应用程序将业务规则、数据访问、合法性校验等工作放到了中间层进行处理。通常情况下，客户端不直接与数据库进行交互，而是通过COM/DCOM通讯与中间层建立连接，再经由中间层与数据库进行交互。

1. 各层的作用
   1. 数据访问层

主要是对原始数据（数据库或者文本文件等存放数据的形式）的操作层，而不是指原始数据，也就是说，是对数据的操作，而不是数据库，具体为业务逻辑层或表示层提供数据服务。

* 1. 业务逻辑层

主要是针对具体的问题的操作，也可以理解成对数据层的操作，对数据业务逻辑处理，如果说数据层是积木，那逻辑层就是对这些积木的搭建。

* 1. 表示层

主要表示WEB方式，也可以表示成WINFORM方式，WEB方式也可以表现成：aspx，如果逻辑层相当强大和完善，无论表现层如何定义和更改，逻辑层都能完善地提供服务。

在Java web 项目中：

dao层：数据访问层,操作数据库，对数据进行增删改查

service层：业务逻辑层,对数据进行处理

web层：表示层，给页面传递数据

1. 优缺点

优点：

* 1. 开发人员可以只关注整个结构中的其中某一层；
  2. 可以很容易的用新的实现来替换原有层次的实现；
  3. 可以降低层与层之间的依赖；
  4. 有利于标准化；
  5. 利于各层逻辑的复用。
  6. 结构更加的明确
  7. 在后期维护的时候，极大地降低了维护成本和维护时间

缺点：

1. 降低了系统的性能。这是不言而喻的。如果不采用分层式结构，很多业务可以直接造访数据库，以此获取相应的数据，如今却必须通过中间层来完成。
2. 有时会导致级联的修改。这种修改尤其体现在自上而下的方向。如果在表示层中需要增加一个功能，为保证其设计符合分层式结构，可能需要在相应的业务逻辑层和数据访问层中都增加相应的代码。
3. 增加了开发成本。
4. **实现过程**
5. 功能介绍
6. 登录功能

查询数据库中用户表，防止非法登录。用户退出则回到登录界面。



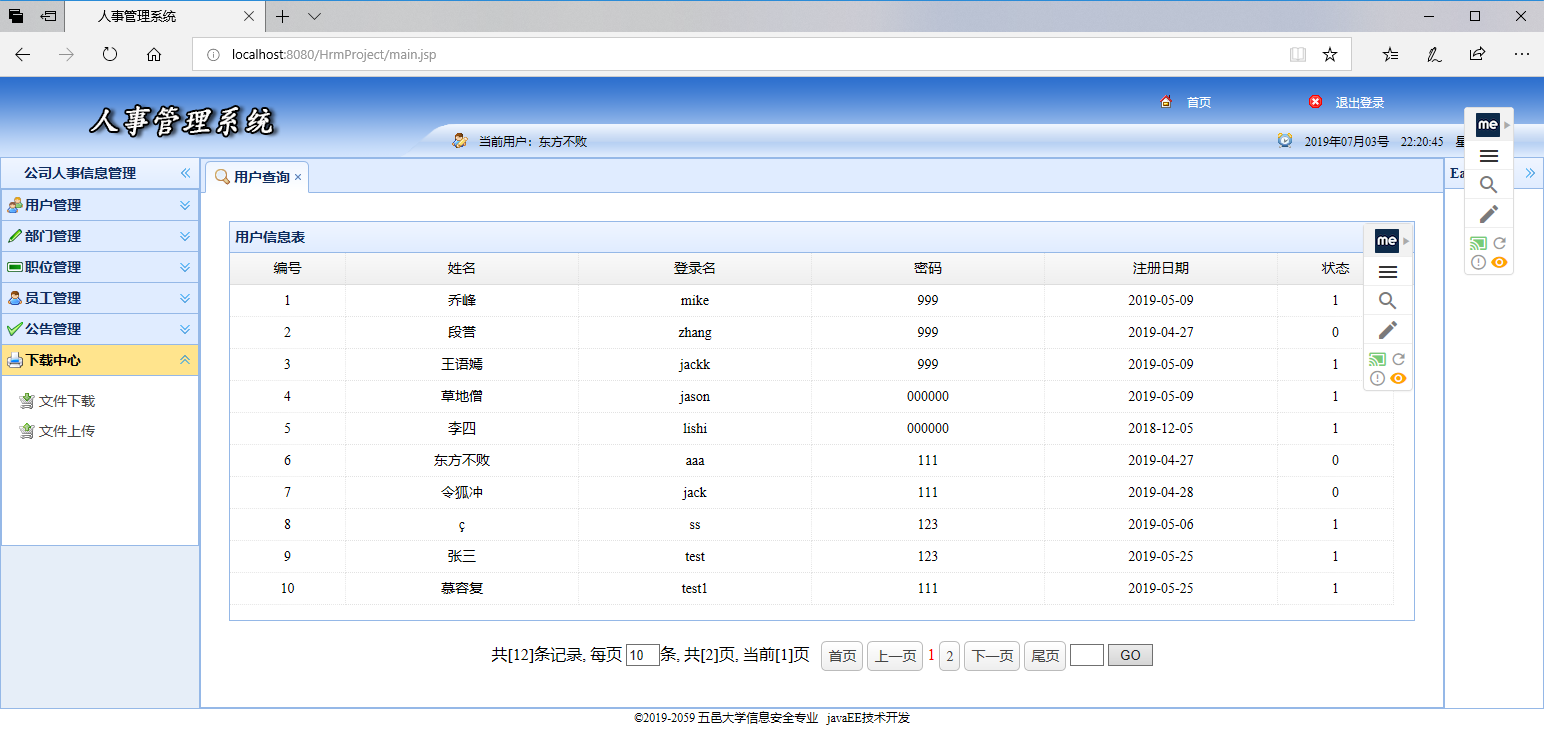
使用JOptionPane.showMessageDialog(null, "没有这个用户", "登录失败", JOptionPane.ERROR\_MESSAGE);弹出提示。

然后执行response.sendRedirect("/HrmProject/login.jsp");跳回登录界面。



使用JOptionPane.showMessageDialog(null, "密码错误", "登录失败", JOptionPane.ERROR\_MESSAGE); 弹出提示。

然后执行response.sendRedirect("/HrmProject/login.jsp");跳回登录界面。

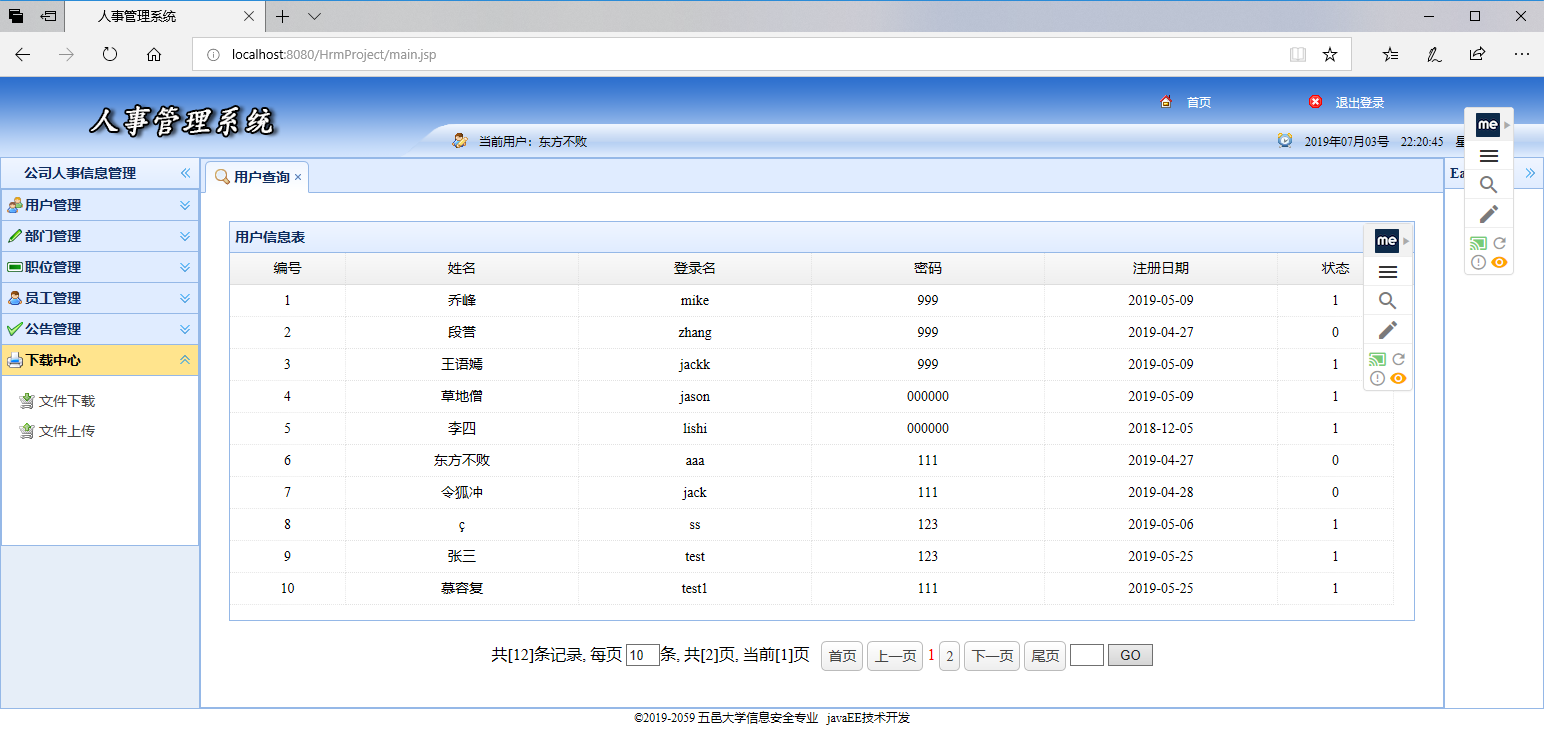


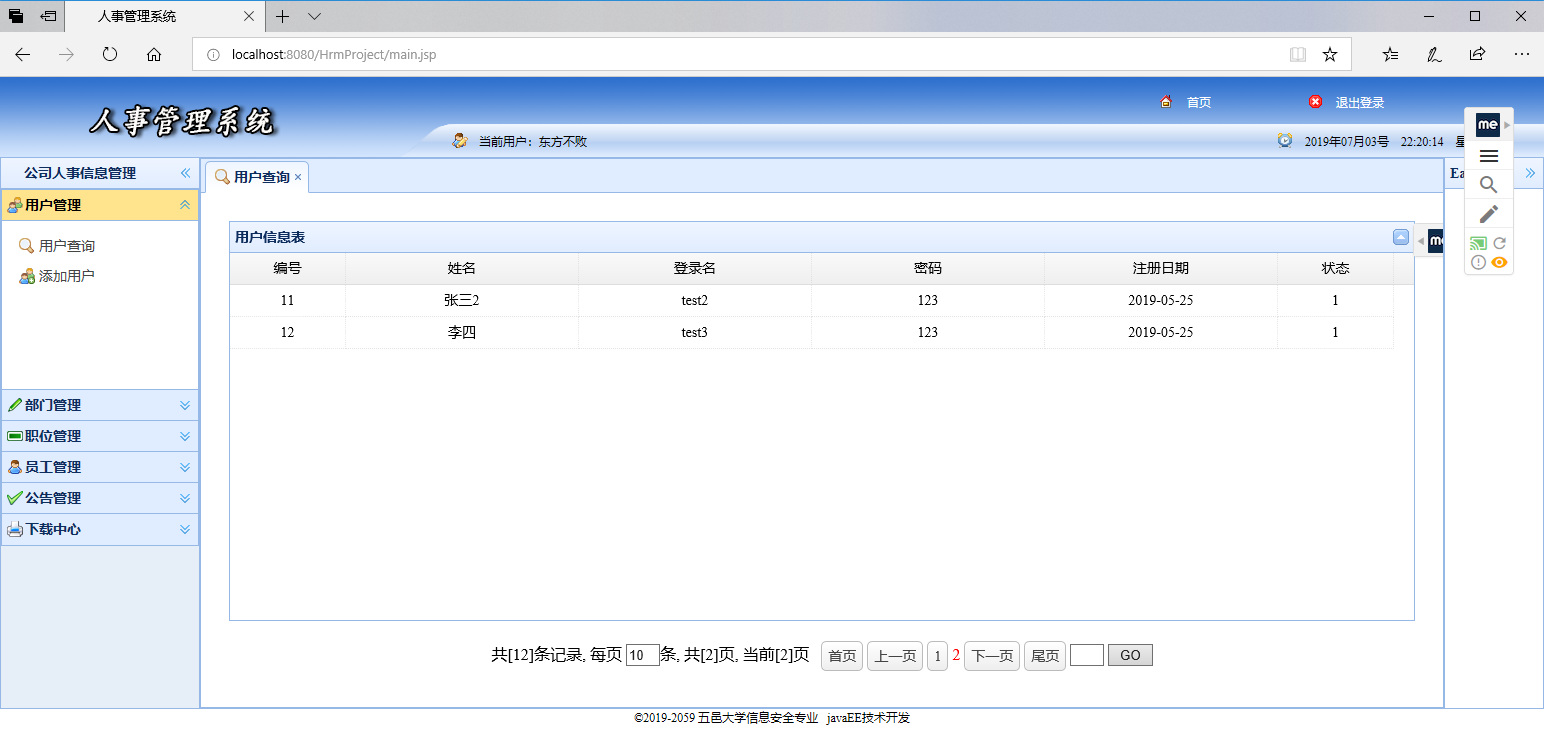
若登录成功，先执行 HttpSession session = request.getSession();设一个session，然后执行session.setAttribute("Username", user.getUsername());获得用户名，并存入Username字段内，使用session传值，最后执行response.sendRedirect("/HrmProject/main.jsp");跳转到主页main.jsp，在主页中通过session的Username字段获取当前用户的用户名，对应的代码为：

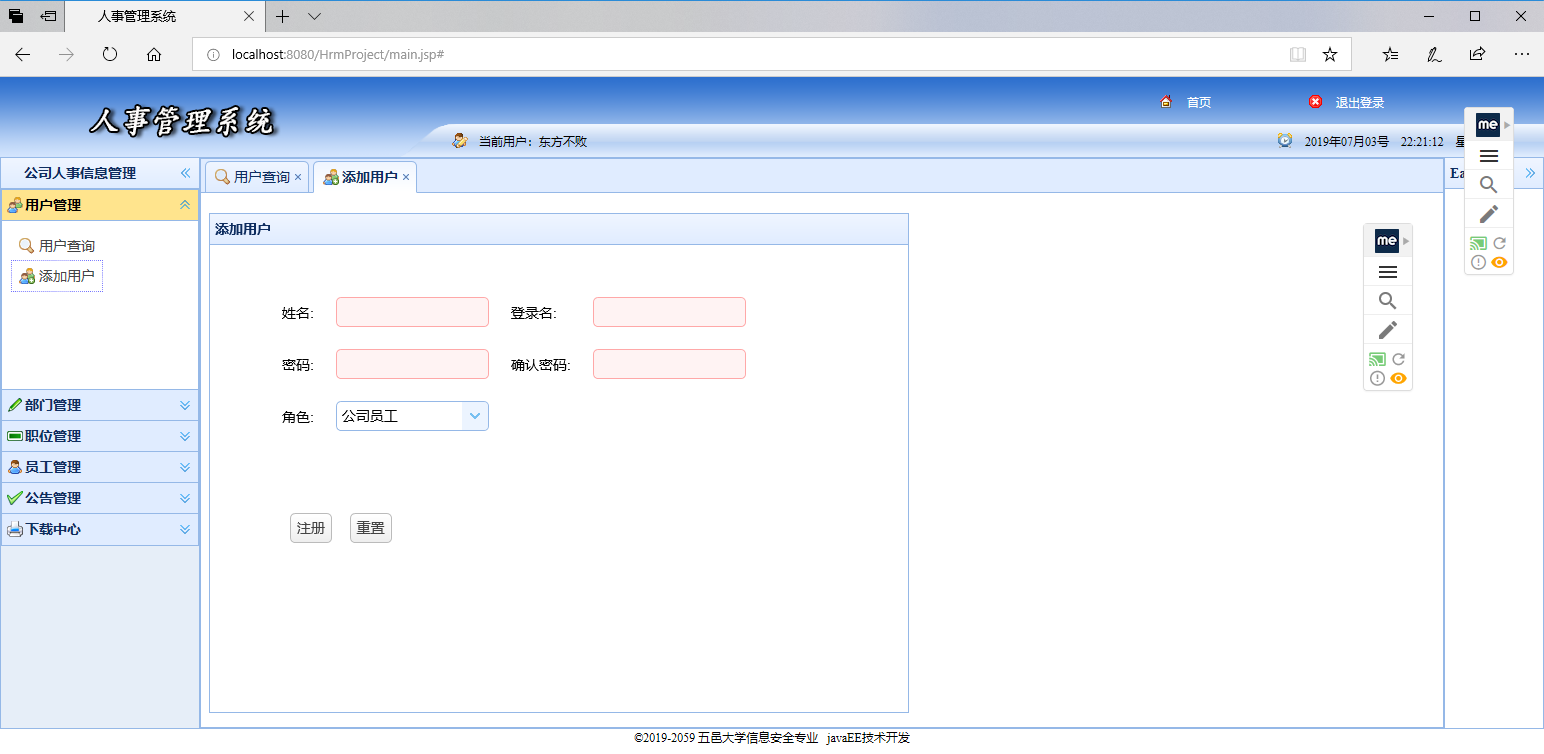
<td>当前用户：<span><%=session.getAttribute("Username")%></span></td>

1. 用户管理

实现对用户表的分页查询和添加功能。



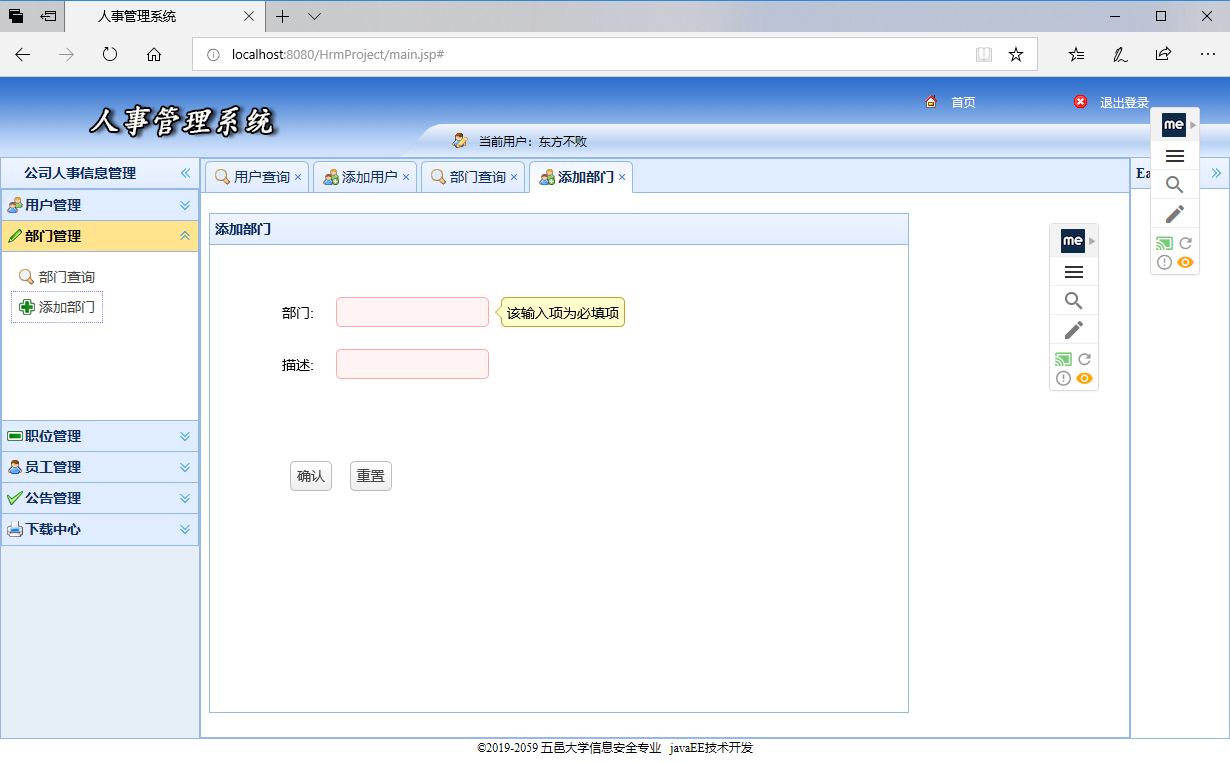




1. 部门管理

实现对部门表的分页查询和添加功能。





1. 职位管理

实现对职位表的分页查询和添加功能。

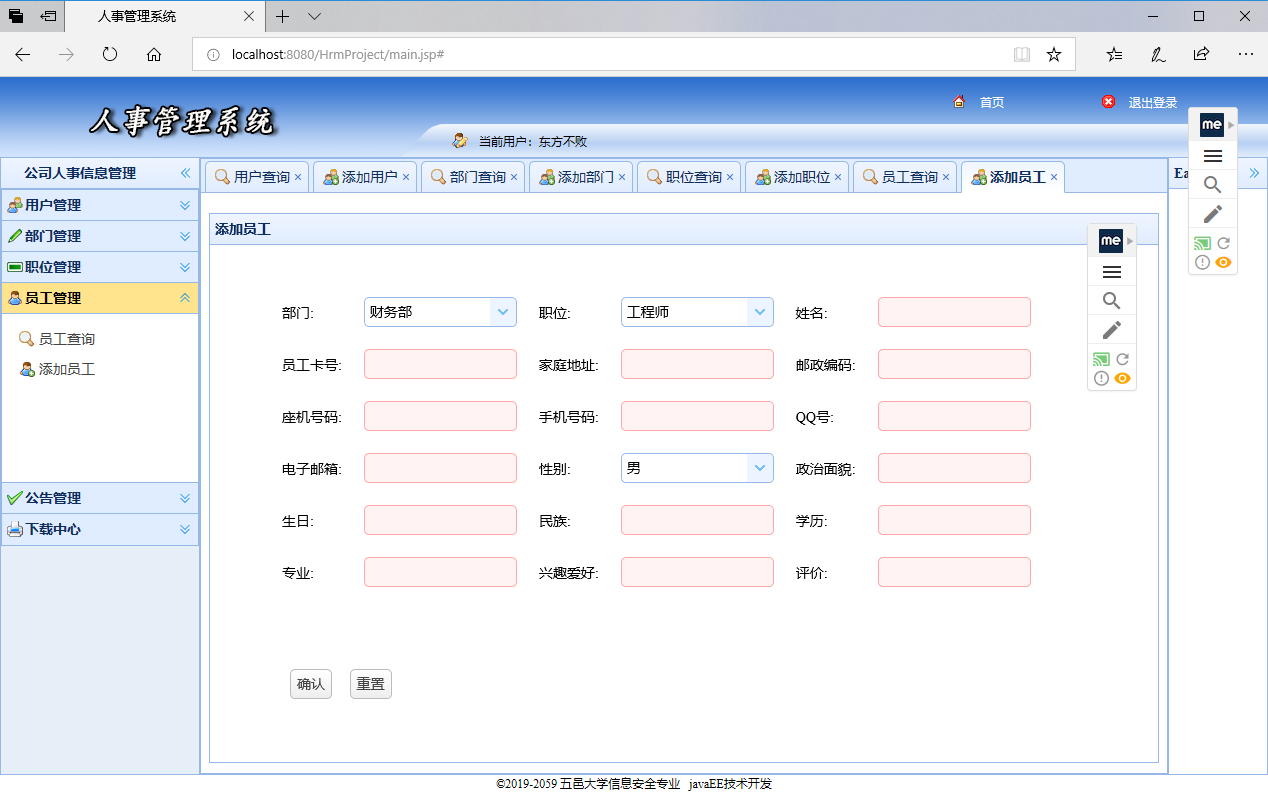




1. 员工管理

实现对员工表的分页查询和添加功能。

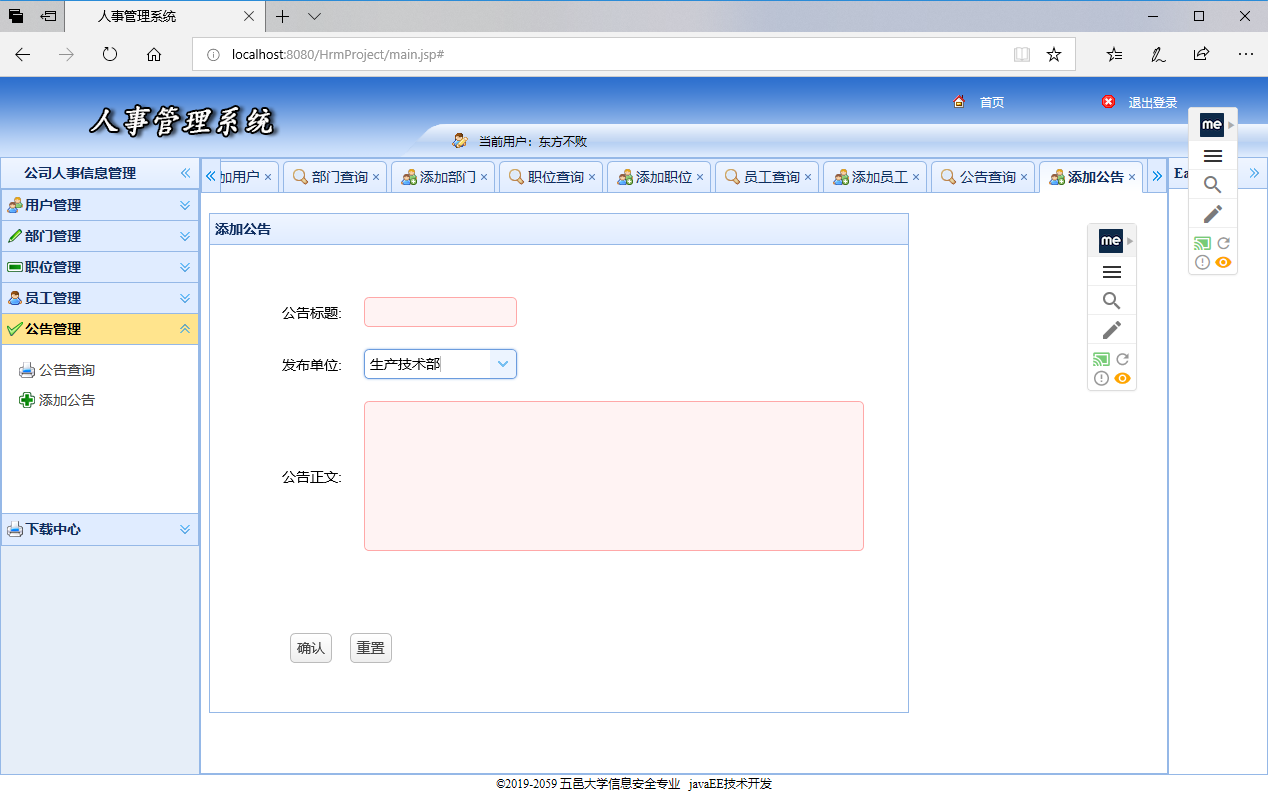




1. 公告管理

实现对公告表的分页查询和添加功能。



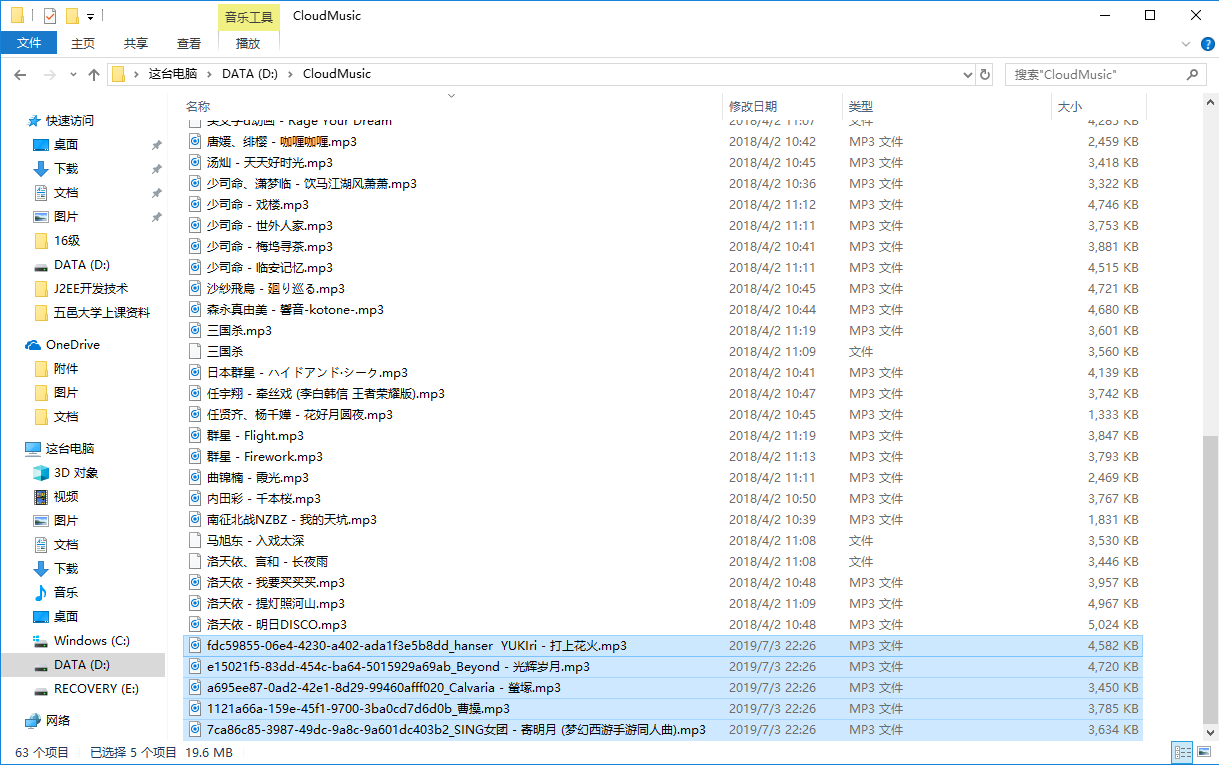


1. 下载中心

实现多文件的上传和下载功能。







1. 界面介绍

页面使用easyui编写完成后移植到jsp中。主页布局为BorderLayout 边界布局，按东西南北中五个方向来布局，各个功能页面在中间的div内addTab显示出来。

1. 完成情况

项目使用dao、service和web三层架构，以员工管理为例。

员工表在数据库中的表名为employee\_inf，首先设Emp.java、EmpDao.java、EmpService.java、EmpDaoImpl.java和EmpServiceImpl.java五个java文件。

Emp.java代码：

在它里面不仅为employee\_inf表的所有字段设置了成员和对应的set、get函数，而且还为部门编号和职业编号对应的部门名称和职业名称也设置了成员和函数。为的就是在后面的查询中能保存它们。

EmpDao.java代码：

定义了添加和查询的接口。

EmpService.java代码：

定义了添加和查询的接口。

EmpDaoImpl.java：

定义了添加和查询的函数。

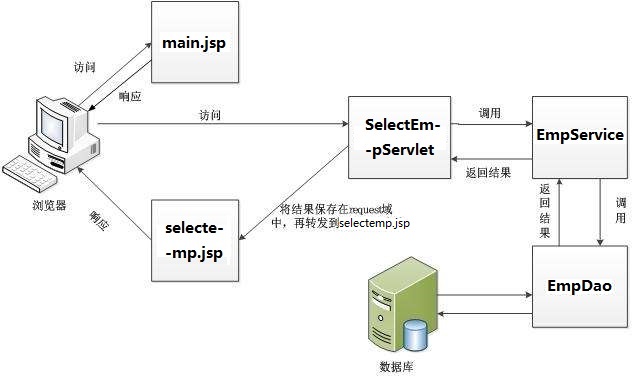
在添加函数中使用SimpleDateFormat将时间转换成指定的格式。

在查询语句在使用inner join内连接查询部门表和职位表的中对应的名称，并使用limit实现分页查询。

EmpServiceImpl.java：

实现了EmpService。

查询的流程如下图所示：



浏览器访问main.jsp页面并得到响应，随后点击员工查询，访问SelectEmpSevrlet，它会调用EmpService（提供接口），而EmpService的实现类又去调用EmpDao（提供接口），由EmpDao的实现类通过调用JDBC工具类操作数据库完成查询操作，并得到数据库返回的结果，接着EmpDao将结果返回给EmpService，而EmpService又会将结果返回给SelectEmpSevrlet，它将结果保存在request域中，再转发到selectemp.jsp页面，最后是selectemp.jsp响应浏览器完成此次查询。

其他如用户、部门、职位和公告的查询及添加功能的流程与此基本一致，详情见项目代码。

添加员工涉及两个文件，addemp.jsp和AddEmpService.java。

具体流程是：

首先addemp.jsp页面用表单<form></form>收集数据，然后提交给AddEmpService.java。接着AddEmpService.java将数据存入字符串，并封装到对象中，然后调用Servlet层的添加方法，最后往addemp.jsp页面放回数据，addemp.jsp则根据返回的data弹出不同的easyui信息提示框添加成功或添加失败。

查询员工涉及两个文件，selectemp.jsp和SelectEmpService.java。

具体流程是：

首先SelectEmpService.java执行doGet，定义一个QueryInfo info初始化页面的页数和页面大小，接着new一个EmpServiceImpl名字是service，然后设PageBean pagebean = service.pageQuery(info);完成后将查询结果转发到员工信息显示页面selectemp.jsp。

在selectemp.jsp中的${c.id}一类的元素，里面c点后面的成员必须之前在Emp.java中定义过，否则会找不到。

1. **调试运行中碰到的问题及解决办法**
2. 问题一

上传文件涉及到的UploadServlet.java文件中，通过String filename = fileItem.getName();来获取文件名，在火狐浏览器、306浏览器、ChromeCore浏览器和谷歌浏览器下会返回文件名，但是如IE浏览器和Microsoft Edge浏览器下都会返回了“路径”+“文件名”。

针对这种情况，我在filename = fileItem.getName();的后面加上filename = filename.substring(filename.lastIndexOf('\\')+1,filename.length());使用lastIndexOf()函数找到到斜杠最后一次出现的位置，然后使用substring()函数获取从到斜杠最后一次出现的位置后一个开始到字符串末尾的内容，也就是所需的文件名。

1. 问题二

上传文件涉及到的UploadServlet.java文件中，通过不论是否有选择文件，点击上传都会上传成功。查看对应的文件夹D:\CloudMusic，发现里面增加了大小为0KB的空文件，文件名都是由filename = UUID.randomUUID().toString() + "\_" + filename;产生的。

针对这种情况，我在filename = UUID.randomUUID().toString() + "\_" + filename;前面增加了if (0 < filename.length())的判断来确保选择了文件。点击上传后，反应和上传成功虽然是一样的，但文件夹中不会增加空文件。

UploadServlet.java代码：

1. // 先获取文件的名称
2. String filename = fileItem.getName();
3. filename=filename.substring(filename.lastIndexOf('\\')+1,filename.length());
4. // 获取唯一的字符串:通用唯一识别码
5. **if** (0 < filename.length()){
6. filename = UUID.randomUUID().toString() + "\_" + filename;
7. // 获取文件的输入流
8. InputStream in = **new** BufferedInputStream(fileItem.getInputStream());
9. // 项某个文件中写入
10. // 获取输出流
11. OutputStream os = **new** BufferedOutputStream(**new** FileOutputStream(path + "\\/" + filename));
12. // io拷贝
13. **int** len = 0;
14. **byte**[] b = **new** **byte**[1024];
15. **while** ((len = in.read(b)) != -1) {
16. os.write(b, 0, len);
17. }
18. in.close();
19. os.close();
20. }
21. **课程学习体会**

学习javaEE是离不开数据库的，基于j2ee的项目大多都是MIS系统，所以访问数据库是javaEE技术的核心，因此我们在日常编码过程中有意无意都要怀有数据库的意识，怀有数据模型的概念，这样我们才能编出具有企业价值，具有实用价值的代码。

这学期通过对JavaEE的学习，掌握了java的基本开发方法，灵活应用所学javaEE知识，编写程序实现系统所需功能。提高了综合运用所学的理论知识和方法分析和解决问题的能力；并用系统的的观点进行系统设计，巩固、深化了理论知识，提高编程水平，并在此过程中培养出严谨的科学态度和良好的编程风格。