|  |  |
| --- | --- |
| 评分 | 指导教师签名 |
|  |  |





**信息系统分析与设计**

**课程项目**

|  |  |  |
| --- | --- | --- |
| 项目名称 | **：** | **中小型餐饮业人事管理系统——以时尚城“麻麻鱼府”为例** |
| 组长 | **：** | **1409030235：佘华扬** |
| 组员 | **：** | **袁梓倩 廖娴静 宫春燕 张颖** |
| 班级 | **：** | **信管1402** |
| 指导教师 | **：** | **杨怡光** |
| 完成时间 | **：** | **2017年01月12日** |

**修改记录：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **编写/修订说明** | **版本** | **修改日期** | **修改人** | **备注** |
| **1** | 需求分析 | 1.0 | 2016-10-23 | 全体 | 将选题细化后修改各部分内容 |
| **2** | 整体修改 | 1.1 | 2016-11-11 | 佘华扬 | 按要求修改全部 |
| **3** | OOA | 1.2 | 2016-11-11 | 宫春燕 | 修改流程图 |
| **4** | OOA | 1.3 | 2016-11-11 | 张颖 | 修改功能模型 |
| **5** | 整体修改 | 2.0 | 2016-11-25 | 佘华扬 | 按要求修改全部 |
| **6** | OOD | 2.1 | 2016-11-26 | 廖娴静 | 修改功能模型 |
| **7** | OOD | 2.2 | 2016-12-03 | 袁梓倩 | 修改OOD类图 |
| **8** | OOA | 2.3 | 2016-12-04 | 宫春燕 | 修改用例模型规约 |
| **9** | 整体修改 | 3.0 | 2016-12-13 | 佘华扬 | 按要求修改全部 |
| **10** | OOD | 3.1 | 2016-12-15 | 宫春燕 | 修改人机界面设计 |
| **11** | OOD | 3.2 | 2016-12-15 | 袁梓倩 | 修改系统总体方案设计 |
| **12** | OOI | 3.3 | 2016-12-17 | 张颖 | 修改数据库表 |
| **13** | OOI | 3.4 | 2016-12-20 | 佘华扬 | 修改软件测试 |
| **14** | OOI | 4.0 | 2016-12-25 | 袁梓倩 | 修改关键算法设计 |
| **15** | OOI | 4.1 | 2017-01-01 | 廖娴静 | 修改应用实例 |
| **16** | 整体修改 | 4.2 | 2017-01-03 | 佘华扬 | 按要求修改全部 |
| **17** | 结语 | 4.3 | 2017-01-07 | 张颖 | 修改系统难点 |
| **18** | 整体修改 | 5.0 | 2017-01-10 | 佘华扬 | 整体综合修改全部内容 |
| **19** | OOD | 5.1 | 2017-01-15 | 袁梓倩 | 修改容错设计 |
| **20** | OOD | 5.2 | 2017-01-15 | 廖娴静 | 修改接口设计 |
| **21** | 引言 | 5.3 | 2017-01-15 | 张颖 | 完善系统模型 |
| **22** | 整体修改 | 6.0 | 2017-01-16 | 佘华扬 | 整体综合修改全部内容 |

**贡献记录**：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **姓名** | | | **佘华扬** | **袁梓倩** | **张颖** | **廖娴静** | **宫春燕** |
| **系统分析(30%)** | **用例模型** | 功能模型 | 25% | 20% | 20% | 20% | 15% |
| 分析模型 | 20% | 20% | 20% | 25% | 15% |
| 辅助模型 | 20% | 20% | 20% | 20% | 25% |
| 其他加分因素 | 总控 |  |  | 设计操作 | 排版 |
| **分析模型** | OOA类图 | 20% | 20% | 25% | 15% | 20% |
| OOD类图 | 20% | 25% | 20% | 20% | 15% |
| 其他 | 总控 | 画图操作 |  |  | 排版 |
| **辅助模型** | 顺序图 | 20% | 24% | 20% | 22% | 14% |
| 协作图 | 20% | 24% | 20% | 22% | 14% |
| 活动图 | 20% | 24% | 20% | 22% | 14% |
| 状态图 | 20% | 24% | 20% | 22% | 14% |
| 其他加分因素 | 总控 | 画图操作 |  |  | 排版 |
| **系统设计(30%)** | 数据库设计 | | 20% | 10& | 30% | 30% | 10% |
| 界面设计 | | 20% | 20% | 20% | 20% | 20% |
| 算法控制驱动设计 | | 25% | 27.5% | 10% | 27.5% | 10% |
| 构件设计 | | 60% |  | 20% |  | 20% |
| 设计原则 | | 40% | 20% | 20% | 20% |  |
| 设计模式 | | 20% | 20% | 20% |  | 40% |
| 部署设计 | | 60% |  | 20% | 20% |  |
| 其他加分因素 | | 总控 |  | 设计 | 设计 | 排版 |
| **系统实现(40%)** | 功能测试 | | 100% |  |  |  |  |
| 性能测试 | |  | 100% |  |  |  |
| 存储测试 | |  |  | 100% |  |  |
| 并发测试 | |  |  |  | 100% |  |
| 容错性测试 | |  |  |  |  | 100& |
| 可靠性测试 | | 20% | 20% | 20% | 20% | 20% |
| 系统测试 | | 20% | 20% | 20% | 20% | 20% |
| 其他加分因素 | | 总控 |  |  |  | 排版 |
| **总分** | | |  |  |  |  |  |

**贡献系数**：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **姓名** | | | **佘华扬** | **袁梓倩** | **张颖** | **廖娴静** | **宫春燕** |
| **系统分析(30%)** | **用例模型** | 功能模型 | 1.25 | 1 | 1 | 1 | 0.75 |
| 分析模型 | 1 | 1 | 1 | 1.25 | 0.75 |
| 辅助模型 | 1 | 1 | 1 | 1 | 1.25 |
| 其他加分因素 | 总控 |  |  | 设计操作 | 排版 |
| **分析模型** | OOA类图 | 1 | 1 | 1.25 | 0.75 | 1 |
| OOD类图 | 1 | 1.25 | 1 | 1 | 0.75 |
| 其他 | 总控 | 画图操作 |  |  | 排版 |
| **辅助模型** | 顺序图 | 1 | 1.2 | 1 | 1.1 | 0.7 |
| 协作图 | 1 | 1.2 | 1 | 1.1 | 0.7 |
| 活动图 | 1 | 1.2 | 1 | 1.1 | 0.7 |
| 状态图 | 1 | 1.2 | 1 | 1.1 | 0.7 |
| 其他加分因素 | 总控 | 画图操作 |  |  | 排版 |
| **系统设计(30%)** | 数据库设计 | | 1 | 0.5 | 1.5 | 1.5 | 0.5 |
| 界面设计 | | 1 | 1 | 1 | 1 | 1 |
| 算法控制驱动设计 | | 1.25 | 1.375 | 0.5 | 1.375 | 0.5 |
| 构件设计 | | 3 |  | 1 |  | 1 |
| 设计原则 | | 2 | 1 | 1 | 1 |  |
| 设计模式 | | 1 | 1 | 1 |  | 2 |
| 部署设计 | | 3 |  | 1 | 1 |  |
| 其他加分因素 | | 总控 |  | 设计 | 设计 | 排版 |
| **系统实现(40%)** | 功能测试 | | 5 |  |  |  |  |
| 性能测试 | |  | 5 |  |  |  |
| 存储测试 | |  |  | 5 |  |  |
| 并发测试 | |  |  |  | 5 |  |
| 容错性测试 | |  |  |  |  | 5 |
| 可靠性测试 | | 1 | 1 | 1 | 1 | 1 |
| 系统测试 | | 1 | 1 | 1 | 1 | 1 |
| 其他加分因素 | | 总控 |  |  |  | 排版 |
| **整体系数** | | | 1.21 | 0.965 | 1.01 | 0.968 | 0.906 |

**目录**

[1.引言 8](#_Toc472336643)

[1.1系统概述 8](#_Toc472336644)

[1.2国内外现状 9](#_Toc472336645)

[1.3系统开发的意义 10](#_Toc472336646)

[1.4系统模型及开发步骤 11](#_Toc472336647)

[1.5可行性分析 16](#_Toc472336648)

[1.6系统创新点 16](#_Toc472336649)

[1.7系统开发框架 17](#_Toc472336650)

[1.7.1 Struts与Hibernate 17](#_Toc472336651)

[1.7.2 SSH与Jfinal的区别 20](#_Toc472336652)

[2. 面向对象分析 OOA 22](#_Toc472336653)

[2.1 OOA概述 22](#_Toc472336654)

[2.1.1 OOA的必要性 22](#_Toc472336655)

[2.1.2 OOA的任务 22](#_Toc472336656)

[2.1.3 OOA模型 22](#_Toc472336657)

[2.2 需求分析 23](#_Toc472336658)

[2.2.1 需求分析概述 23](#_Toc472336659)

[2.2.2功能分析与功能模型 24](#_Toc472336660)

[2.2.3业务建模 26](#_Toc472336661)

[2.2.4用例建模 35](#_Toc472336662)

[2.3 建立基本模型（OOA类图） 45](#_Toc472336663)

[2.3.1基本模型概述 45](#_Toc472336664)

[2.3.2基本模型 46](#_Toc472336665)

[2.4建立辅助模型 48](#_Toc472336666)

[2.4.1辅助模型概述 48](#_Toc472336667)

[2.4.2顺序图 48](#_Toc472336668)

[2.4.3状态图 52](#_Toc472336669)

[2.4.4活动图 55](#_Toc472336670)

[3.面向对象设计 OOD 61](#_Toc472336671)

[3.1 OOD概述 61](#_Toc472336672)

[3.1.1概念 61](#_Toc472336673)

[3.1.2系统设计模型 61](#_Toc472336674)

[3.1.3设计原则 62](#_Toc472336675)

[3.1.4设计模式 63](#_Toc472336676)

[3.2问题域设计 63](#_Toc472336677)

[3.2.1问题域设计任务 63](#_Toc472336678)

[3.2.2将OOA类图转换为OOD类图 76](#_Toc472336679)

[3.3人机界面设计 79](#_Toc472336682)

[3.3.1界面设计的意义 79](#_Toc472336683)

[3.3.2界面设计的内容 80](#_Toc472336684)

[3.3.3界面设计的一般原则 89](#_Toc472336685)

[3.4数据库设计 90](#_Toc472336686)

[3.4.1数据库设计概述 91](#_Toc472336687)

[3.4.2数据库设计依据 94](#_Toc472336688)

[3.4.3数据库表设计 96](#_Toc472336689)

[3.5控制驱动设计 104](#_Toc472336690)

[3.5.1控制驱动设计概述 104](#_Toc472336691)

[3.5.2控制驱动设计依据 105](#_Toc472336692)

[3.5.3系统并发控制设计 107](#_Toc472336693)

[3.5.6系统总体方案设计 108](#_Toc472336694)

[3.5.7软件体系结构设计 108](#_Toc472336695)

[3.6构件部署设计 109](#_Toc472336696)

[3.6.1面向对象构件方法与技术概述 109](#_Toc472336697)

[3.6.2构件设计 111](#_Toc472336698)

[3.7体系结构设计 114](#_Toc472336699)

[3.7.1系统的分层结构设计 114](#_Toc472336700)

[3.7.2系统的正交设计 116](#_Toc472336701)

[3.7.3系统的MVC模式设计 127](#_Toc472336702)

[3.8接口设计 130](#_Toc472336703)

[3.8.1用户接口 130](#_Toc472336704)

[3.8.2模块接口 136](#_Toc472336705)

[3.8.3数据接口 140](#_Toc472336706)

[3.8.4总结 148](#_Toc472336707)

[3.9容错设计 148](#_Toc472336708)

[3.9.1出错信息 148](#_Toc472336709)

[3.9.2补救措施 149](#_Toc472336710)

[4.面向对象实现OOI 150](#_Toc472336711)

[4.1OOI概述 150](#_Toc472336712)

[4.1.1面向对象实现 150](#_Toc472336713)

[4.1.2系统实现的内容 150](#_Toc472336714)

[4.1.3系统角色分析 154](#_Toc472336715)

[4.2系统初始化 155](#_Toc472336716)

[4.3关键业务逻辑|困难算法实现 155](#_Toc472336717)

[4.3.1解决查看不同人员详情的问题 155](#_Toc472336718)

[4.3.2解决动态向用户展示数据的问题。 158](#_Toc472336719)

[4.3.3解决动态数据存储问题 162](#_Toc472336720)

[4.4程序（代码）设计 176](#_Toc472336721)

[4.5软件测试 218](#_Toc472336722)

[4.5.1软件测试的概念 218](#_Toc472336723)

[4.5.2测试方法 218](#_Toc472336724)

[4.5.3测试实例 221](#_Toc472336725)

[4.6应用实例 226](#_Toc472336726)

[4.6.1登录界面 226](#_Toc472336727)

[4.6.2系统首页 227](#_Toc472336728)

[4.6.3人员管理界面 227](#_Toc472336729)

[4.6.4招聘管理界面 228](#_Toc472336730)

[4.6.5人事异动界面 230](#_Toc472336731)

[4.6.6培训管理界面 231](#_Toc472336732)

[4.6.7奖惩管理界面 232](#_Toc472336733)

[4.6.8绩效管理界面 233](#_Toc472336734)

[4.6.9薪金管理界面 234](#_Toc472336735)

[4.7创新应用实例 235](#_Toc472336736)

[4.7.1正则表达式应用 235](#_Toc472336737)

[4.7.2多设备的适应性 240](#_Toc472336738)

[4.7.3交互性实现 244](#_Toc472336739)

[5．总结与展望 245](#_Toc472336740)

[5.1系统开发总结 245](#_Toc472336741)

[5.1.1系统设计原则 245](#_Toc472336742)

[5.1.2系统设计模式 246](#_Toc472336743)

[5.1.3系统设计难点 254](#_Toc472336744)

[5.2系统开发展望 256](#_Toc472336745)

[5.2.1系统不足及改进方向 256](#_Toc472336746)

[5.2.2感悟 257](#_Toc472336747)

[参考文献 259](#_Toc472336748)

[代码： 260](#_Toc472336749)

[/HRmanagement/src/com.zhangying.action/EducateAction.java 260](#_Toc472336750)

[/HRmanagement/src/com.zhangying.action/EducateForm.java 264](#_Toc472336751)

[/HRmanagement/src/com.zhangying.action/InstitutionAction.java 270](#_Toc472336752)

[/HRmanagement/src/com.zhangying.action/InstitutionForm.java 274](#_Toc472336753)

[/HRmanagement/src/com.zhangying.action/JobAction.java 277](#_Toc472336754)

[/HRmanagement/src/com.zhangying.action/JobForm.java 280](#_Toc472336755)

[/HRmanagement/src/com.zhangying.action/PerformanceAction.java 287](#_Toc472336756)

[/HRmanagement/src/com.zhangying.action/PerformanceForm.java 291](#_Toc472336757)

[/HRmanagement/src/com.zhangying.action/StipendAction.java 294](#_Toc472336758)

[/HRmanagement/src/com.zhangying.action/StipendForm.java 298](#_Toc472336759)

[/HRmanagement/src/com.zhangying.action/UsersAction.java 302](#_Toc472336760)

[/HRmanagement/src/com.zhangying.action/UsersForm.java 305](#_Toc472336761)

[/HRmanagement/src/com.zhangying.dao/EducateDao.java 311](#_Toc472336762)

[/HRmanagement/src/com.zhangying.dao/InstitutionDao.java 314](#_Toc472336763)

[/HRmanagement/src/com.zhangying.dao/JobDao.java 316](#_Toc472336764)

[/HRmanagement/src/com.zhangying.dao/PerformanceDao.java 319](#_Toc472336765)

[/HRmanagement/src/com.zhangying.dao/StipendDao.java 321](#_Toc472336766)

[/HRmanagement/src/com.zhangying.dao/UsersDao.java 324](#_Toc472336767)

[/HRmanagement/src/com.zhangying.hb/HibernateSessionFactory.java 326](#_Toc472336768)

[/HRmanagement/src/com.zhangying.po/Educate.hbm.xml 329](#_Toc472336769)

[/HRmanagement/src/com.zhangying.po/Educate.java 330](#_Toc472336770)

[/HRmanagement/src/com.zhangying.po/Institution.hbm.xml 335](#_Toc472336771)

[/HRmanagement/src/com.zhangying.po/Institution.java 335](#_Toc472336772)

[/HRmanagement/src/com.zhangying.po/Job.hbm.xml 337](#_Toc472336773)

[/HRmanagement/src/com.zhangying.po/Job.java 338](#_Toc472336774)

[/HRmanagement/src/com.zhangying.po/Performance.hbm.xml 344](#_Toc472336775)

[/HRmanagement/src/com.zhangying.po/Performance.java 344](#_Toc472336776)

[/HRmanagement/src/com.zhangying.po/Stipend.hbm.xml 346](#_Toc472336777)

[/HRmanagement/src/com.zhangying.po/Stipend.java 347](#_Toc472336778)

[/HRmanagement/src/com.zhangying.po/Users.hbm.xml 350](#_Toc472336779)

[/HRmanagement/src/com.zhangying.po/Users.java 351](#_Toc472336780)

[/HRmanagement/src/com.zhangying.tool/ActionServlet.java 354](#_Toc472336781)

[/HRmanagement/src/com.zhangying.tool/DateUtil.java 355](#_Toc472336782)

[/HRmanagement/src/com.zhangying.po/StringUtil.java 357](#_Toc472336783)

[/HRmanagement/src/hibernate.cfg.xml 357](#_Toc472336784)

[\HRmanagement\WebRoot\css\common1.css 358](#_Toc472336785)

[\HRmanagement\WebRoot\css\control.css 361](#_Toc472336786)

[\HRmanagement\WebRoot\css\css.css 367](#_Toc472336787)

[\HRmanagement\WebRoot\css\head.css 390](#_Toc472336788)

[\HRmanagement\WebRoot\css\style.css 394](#_Toc472336789)

[\HRmanagement\WebRoot\css\style0.css 411](#_Toc472336790)

[\HRmanagement\WebRoot\css\tab.js 412](#_Toc472336791)

[\HRmanagement\WebRoot\Js\js.js 412](#_Toc472336792)

[\HRmanagement\WebRoot\Js\typem.js 422](#_Toc472336793)

[\HRmanagement\WebRoot\Js\xiangmu.js 425](#_Toc472336794)

[\HRmanagement\WebRoot\addeducate.jsp 426](#_Toc472336795)

[\HRmanagement\WebRoot\addinstitution.jsp 429](#_Toc472336796)

[\HRmanagement\WebRoot\addjob.jsp 431](#_Toc472336797)

[\HRmanagement\WebRoot\addperformance 435](#_Toc472336798)

[\HRmanagement\WebRoot\addstipend.jsp 437](#_Toc472336799)

[\HRmanagement\WebRoot\adduser.jsp 439](#_Toc472336800)

[\HRmanagement\WebRoot\detaileducation 442](#_Toc472336801)

[\HRmanagement\WebRoot\detailinstitution.jsp 446](#_Toc472336802)

[\HRmanagement\WebRoot\detailjob.jsp 448](#_Toc472336803)

[\HRmanagement\WebRoot\detailperformance 451](#_Toc472336804)

[\HRmanagement\WebRoot\error.jsp 453](#_Toc472336805)

[\HRmanagement\WebRoot\index.jsp 454](#_Toc472336806)

[\HRmanagement\WebRoot\left.jsp 454](#_Toc472336807)

[\HRmanagement\WebRoot\listeducate.jsp 464](#_Toc472336808)

[\HRmanagement\WebRootlistinstitution.jsp 468](#_Toc472336809)

[\HRmanagement\WebRoot\listjob.jsp 472](#_Toc472336810)

[\HRmanagement\WebRoot\listperformance.jsp 476](#_Toc472336811)

[\HRmanagement\WebRoot\liststipend.jsp 479](#_Toc472336812)

[\HRmanagement\WebRoot\listuser.jsp 483](#_Toc472336813)

[\HRmanagement\WebRoot\main.jsp 486](#_Toc472336814)

[\HRmanagement\WebRoot\mainfra.html 487](#_Toc472336815)

[\HRmanagement\WebRoot\manage.jsp 488](#_Toc472336816)

[\HRmanagement\WebRoot\MYJSP.jsp 489](#_Toc472336817)

[\HRmanagement\WebRoot\top.html 492](#_Toc472336818)

[\HRmanagement\WebRoot\updateeducate.jsp 493](#_Toc472336819)

[\HRmanagement\WebRoot\updateinstitution.jsp 497](#_Toc472336820)

[\HRmanagement\WebRoot\updatejob.jsp 500](#_Toc472336821)

[\HRmanagement\WebRoot\updateperformance.jsp 503](#_Toc472336822)

[\HRmanagement\WebRoot\updatestipend.jsp 506](#_Toc472336823)

[\HRmanagement\WebRoot\updateuser.jsp 509](#_Toc472336824)

[\HRmanagement\WebRoot\welcome.jsp 513](#_Toc472336825)

[\HRmanagement\WebRoot\WEB-INF\structs-config.xml 515](#_Toc472336826)

# 

# 1.引言

## 1.1系统概述

1.1.1 “麻麻鱼府”规模现状介绍

【名称】麻麻鱼府

【地理位置】中南财经政法大学南湖校区“南湖时尚城”商圈内

【开业日期】2016年10月8日

【经营范围】现有餐桌16张左右，高峰期可容纳约70人就餐

【经营状况】运营状况良好，顾客满意度较高，整体氛围较好，在学生范围内好评度较高。

【管理水平】人员管理较为混乱，制度化不够健全等。具体表现在上菜环节混乱人员较多但“可用的人较少”。

【总结】由此我方认为对其上线一套人事管理系统是有一点的必要性和意义的，以便于其更好的进行人员管理，为中南大学习提供更好的就餐环境等。 对该饭店调研后，小组认为其营业规模符合中小型餐饮业，较适合作为研究对象，故本小组将其作为调研实例。

1.1.2 项目介绍

【项目名称】中小型餐饮业人事管理系统——以时尚城“麻麻鱼府”为例

【背景】 企业人事管理信息系统即对企业人事相关进行管理的系统，负责整个企业的日常人事安排，人员的人事管理等。随着计算机技术和网络通讯的发展以及计算机在企业管理中的普及，利用计算机进行企业人事管理是不可避免的趋势。因此，对于各类型企业来说，利用企业管理信息系统高效率地完成人事管理的日常事务是适应现代化企业人事管理的必然要求。

现代社会，人们生活水平正逐步提高，餐饮业的发展也非常迅速。菜谱多样化和管理科学化已经成为餐饮业的发展趋势。在这种情况下，传统的管理手段已经不能适应餐饮企业的发展。目前，随着餐饮企业业务量的不断增加，企业对人事管理水平提出了更高的要求。至此，餐饮企业人事管理信息系统应运而生。因此，利用现代化信息技术，餐饮企业可以轻松而高效进行系统而全方位的规范和管理。[[1]](#footnote-1)

【目标】开发出完整的定制化的人事管理系统

【范围】由于“麻麻鱼府”是一家连锁店，小组讨论后决定，系统完成后可先在分店使用，视情况决定是否推广。

【开发条件】JAVA+HTML+CSS+JAVASCRIPT+MYSQL

【开发框架】SSH（struts+spring+hibernate）

【环境要求】（1）操作系统：Windows XP SP3或Windows 7

（2）开发环境：Eclipse Java EE IDE for Web Developers.

（3）1G以上的CPU处理器，512MB以上的内存

【初步设想】1、本系统根据需求设计实现，让系统的运行和维护简单化，界面交互简单清晰，提供良好的用户体验。

2、本系统结合我方研究对象——“**麻麻鱼府**”实际的人事管理制度，降低了人力资源管理成本，为该餐饮业决策提供强有力的支持，以达到协助管理者决策的目的。

3、传统的人工管理员工信息及其历史资料等存在许多不足，例如安全性差等，时间长久之后积累大量的文件信息，在更新数据及数据维护方面存在许多问题，本系统合理规避此类问题。

4、本系统采用结构化的开发方法，整体思路清晰，从全局出发保证质量。有利于系统开发过程中结构的把握和总体的管控。

## 1.2国内外现状

1.2.1背景

人事管理系统的发展史可以追溯到20世纪60年代末期。由于当时计算机已经进入实用阶段，同时大型企业用手工来计算和发放薪资既费时、费力又非常容易出差错，为了解决这个魔盾，第一代的人事管理系统营运而生。当时由于技术条件和需求的限制，用户非常少，而且系统充其量也只能算是自动计算薪资的供给而已。但是他的出现为人事的管理展现了美好的前景，即用计算机的高速度和自动化来替代手工的巨大工作量，用计算机的高准确性来避免手工的错误和误差，是大规模几种处理企业人事管理成为可能。

第二代的人事管理系统出现于20世纪70年代末。由于计算机技术的飞速发展无论是计算机的普及性还是计算机系统工具和数据库技术的发展都为人事管理系统的阶段性发展提供了可能。第二代人事管理系统基本上解决了第一代系统的主要缺陷。

人事管理系统的革命性改革出现在20世纪90年代末。由于市场的竞争需要，如何吸引和留住人才，激发员工的创造性、责任感和工作热情已成为关系企业兴衰的重要因素，人才已经成为企业最重要的核心资产之一。同时由于个人电脑的普及，数据库技术，客户/服务器技术，特别是Internet/Intranet技术的发展，是第三代人事管理系统的出现成为必然。

1.2.2现状

随着通讯网络技术、多媒体技术和人工智能技术等为代表的信息收集、处理、加工等技术的迅猛发展，国内的企业人事管理系统的开发技术已经较为成熟，人事管理系统在各类企业中基本上进入了普及应用阶段。国内有多种通用的人事管理系统软件供企业使用，应该说在我国该系统的开发与应用已经进入了成熟期。其中财务等应用较多的软件有金蝶等。

在国外，与我国的人事管理信息系统相对应的是人力资源管理系统。几乎所有发达国家都建有覆盖全国的人力资源管理系统，发达国家的企事业单位，非常注重自身人力资源管理系统与决策支持系统的开发与应用。其中，较为知名的人力资源管理系统有微软、SAP、Oracle等。

即使国内外相关技术和成果已经较为完善，因该类型系统需求量过大且我方系统针对具体研究对象——时尚城“麻麻鱼府”进行“定制化”开发，故本系统的开发具有一定的重要性和必要性。[[2]](#footnote-2)

## 1.3系统开发的意义

1.3.1理论研究意义：

因为本系统是定制化开发，针对具象的某个餐饮店面，根据真实需要进行模块设定，同时进行系统创新，在餐饮业人事系统开发研究上有一定的现实意义。

宏观上讲：顺应了社会信息化潮流，缩短了整个社会大生产的周期，加快社会生产率。

1.3.2实际应用价值

（1）大幅度提高工作效率，化繁为简，减少人事部门工作人员的工作压力，使其将主要经历投入到更为重要的决策中来。

（2）用计算机代替人工，使各类数据更为精准，大大减少了出错的可能性，对数据库的各类数据进行物理和逻辑验证，避免错漏和冗余。

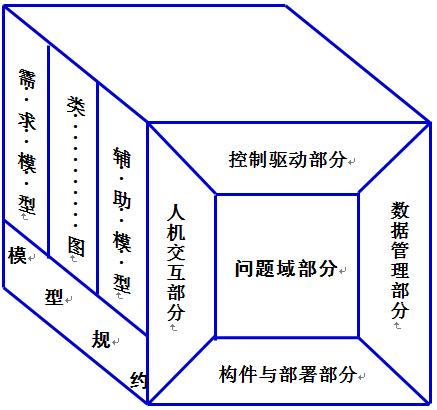
（3）使各部门的管理工作规范化，简化数据汇总和检查的工作量。

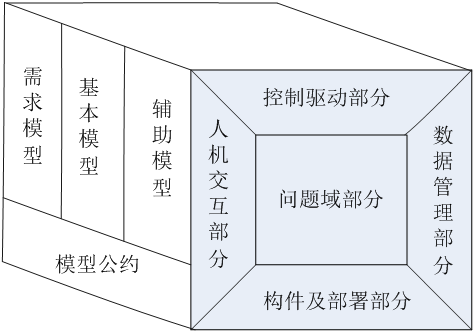
1.3.3经济效益和社会效益

本系统的开发减少了人工工作量，减轻了工作压力，在经济效益和社会效益上成果显著。

## 1.4系统模型及开发步骤

1.4.1信息系统模型





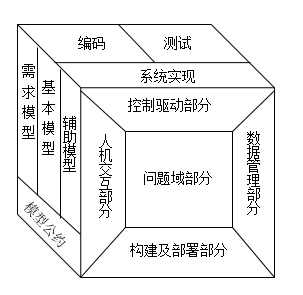


图1.4.1 信息系统模型图

1、从左侧面看，系统模型采用OOA的概念及模型组织方式：

（1）需求模型：需求模型用于捕获与描述用户要求，一般采用用况图。

（2）基本模型（类图）：  
对象层：给出所有与问题域和系统责任有关的对象，用类表示。  
特征层：定义每个类的属性和服务。  
关系层：通过已定义的关系描述类之间的关系。

（3）辅助模型：  
交互图：完成某项特定功能的一组对象之间的详细交互。  
活动图：一个服务的流程或业务流程。  
状态图：一个对象的状态变迁。  
包图：对关系密切的元素打包，帮助理解系统模型。

（4）模型规约：对模型中的所有元素进行详细说明。

2、从正面看，系统模型由核心和外围组成：

核心：问题域部分

外围分别是：控制驱动、人机交互、数据管理、构建与部署。

主要工作内容是以OOA模型为基础，按照实现的要求进行设计决策，全局性的决策和局部细节的设计。

（1）设计问题域部件。OOA的结果是OOD的问题域部件，分析的结果在OOD中可以被改动或增补，但基于问题域的总体组织框架是长时间稳定的。

（2）设计人机交互部件：人机交互部件在上述结果中加入人机交互的设计和交互的细节，包括窗口和输出报告的设计。可以用原型来帮助实际交互机制的开发和选择。

（3）设计任务管理部件：识别事件驱动任务，识别时钟驱动任务，识别优先任务和关键任务，识别协调者，审查每个任务并定义每个任务。

（4）设计数据管理部件：数据管理部件提供了在数据管理系统中存储和检索对象的基本结构，其目的是隔离数据管理方案（如普通文件、关系数据库和面向对象数据库等）对其他部分的影响。

3、从上面看：

（1）编码：将面向对象设计的结果翻译成用某种语言书写面向对象程序，即编码；面向对象程序的质量基本上由面向对象设计的质量决定，但是所采用的编程语言和编程风格也会对程序的可靠性、可重用性及可维护性产生影响。

（2）测试：对用某种语言编写的面向对象程序进行有效的测试，即测试。而保证软件可靠的主要措施是软件测试，软件测试的目标也是用尽可能低的测试成本和测试方案，发现尽可能多的错误。

（3）系统实现：编码和测试共同组建了系统实现，二者缺一不可。

1.4.2系统开发步骤

1、系统规划阶段

（1）含义：有针对性的就即将开发的系统进行规划，做出系列计划，以便日后工作的开展。

（2）目标任务：根据用户需求，给出系统开发计划和的总体方案，并对方案进行可行性分析，撰写系统开发计划书和可行性研究分析报告。

（3）成果：开发计划书和可行性研究分析报告。

2、系统分析阶段

（1）含义：根据开发计划，进行系统分析。

（2）目标任务：对市场上现有系统进行详细调查，描述现有系统的业务流程，指出不足，在现有的基础上进行创新，确定本系统的基本目标和逻辑模型。

（3）成果：系统分析报告。

3、系统设计阶段

（1）含义：根据系统分析报告，进行系统设计。

（2）目标任务：包括总体设计和模块设计。根据相关功能要求，结合实际条件，具体设计实现逻辑模型、物理模型以及最终系统。

（3）成果：系统设计说明书、完整系统。

4、系统实施阶段

（1）含义：系统设计完成后，开始系统实施。

（2）目标任务：进行初步上线，按照实施计划分阶段完成，相应阶段完成“实施进度报告”文档，根据上线效果和情况进行下一步工作安排。

（3）成果：系统使用说明书和系统测试报告

5、运行和维护阶段

（1）含义：最终环节，系统的运行和维护。

（2）目标任务：系统投入运行后，进行维护，记录系统运行情况，根据产生的问题和bug进行后续修改。

（3）成果：系统运行及维护报告

**本系统开发六大特点**

（1）预先明确用户要求 根据需求设计信息系统

（2）自顶向下设计

（3）严格按阶段进行

（4）工作文档规范化和标准化

（5）运用系统的分解和综合技术 使复杂的系统简单化

（6）强调阶段成果的审定和检验[[3]](#footnote-3)

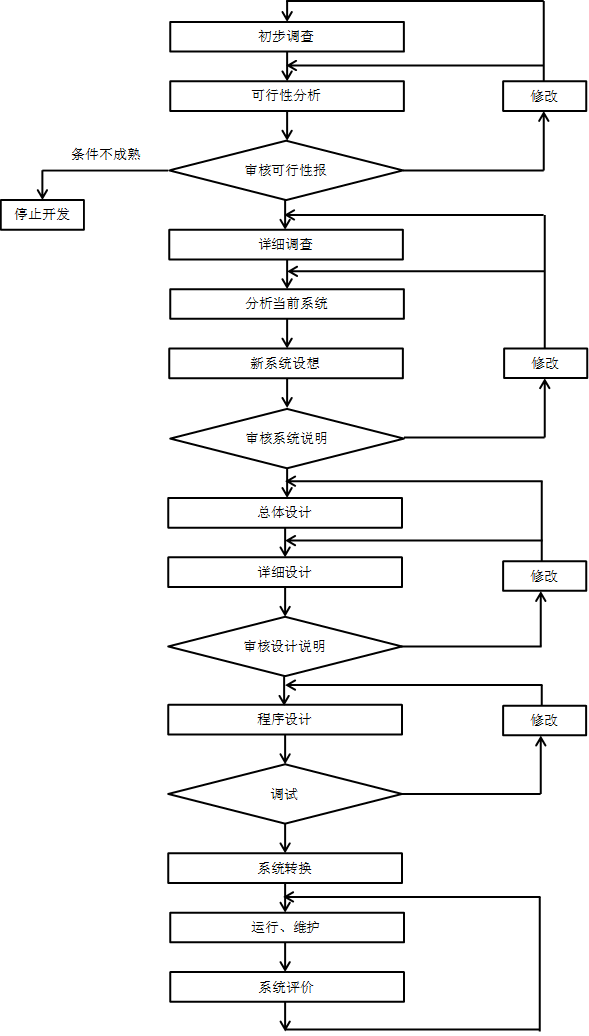


图1.4 系统开发流程图

## 1.5可行性分析

从设备条件方面来看，我们每人均配备计算机一台，且性能比较高，其CPU速度均在266MHz以上，内存均在32M以上，具备了运行目前大多数工具软件的能力，完全满足软件使用的要求。并且我们所在的区域网络良好，完全满足数据传输和通信的要求，为以后系统升级打下了基础。

从技术方面来看，目前的计算机硬件、软件技术以及网络技术完全能完成本系统的功能要求。

从经济方面来看，人事管理信息系统开发的预算可满足。就收益而言，人事处主要的效益是社会效益，系统一旦投入使用，将会极大地提高人事信息管理的工作效率和工作质量。

从组织方面来看，开发系统得到了管理人员的认可，改变现行系统的不完整性是十分必要的。并且我们在系统开发时充分注意了可操作性和易用性，现有员工通过简单培训或阅读操作手册完全能掌握系统并能方便地使用本系统。

通过以上几个方面的分析可知，开发人事管理信息系统是十分必要且可行的。

**1.6系统创新点**

**1、管理层面**

（1）界面简洁大方：为了避免繁杂，我们将设计简单大方的主操作界面，使交互简单明了，注重与用户之间的交流，方便用户使用。使用新颖却不繁复的前端页面设计，给人以舒适的用户体验。

（2）在原有系统的基础上，进行功能模块的创新，实现流程再造，“青出于蓝而胜于蓝。”

**2、技术层面**

（1）开发语言的选择：前端界面在html5、css3、jquery的基础上使用bootstrap框架。后端diamante在Java的基础上使用SSH框架。SSH不是一个框架，而是多个框架（struts+spring+hibernate）的集成，是目前较流行的一种Web应用程序开源集成框架，用于构建灵活、易于扩展的多层Web应用程序。

（2）在信息系统的**安全性**方面进行改进，减少“泄露企业秘密的可能性”。

（3）**数据库表和表**之间的逻辑关系更为清晰和全面，建立好各模块之间的联系。

3、研究对象的选择

在实地调研之后，具象化研究对象。以真实存在的一家中小型餐饮业为研究参照，实现“定制化”开发，使整个系统具有现实意义。

4、多设备的适应性

本系统在一定程度上添加了设备的自适应性功能，避免浏览器屏幕适应问题，使客户浏览页面更加舒适方便。

5、交互性的实现

本系统中的动态表单就是用户和网页进行交互的一种界面元素，有效的将企业与用户联系起来，以此实现了系统的交互性。

**1.7系统开发框架**

### 1.7.1 Struts与Hibernate

1.7.1.1 Struts

1、Struts的几个基本概念

struts是一个开源框架（frameset） ；struts是一个Web框架 ；struts是一个基于MVC的Web框架

2、为什么有struts

因为我们对MVC的理解不同，可能造成写程序的时候，规范不统一，这样不利于程序的维护和扩展以及提高开发效率，所以我们有必要用一个统一的规范来开发项目。所以出现了struts.

struts是通过采用Java的Servlet／JSP技术，实现了基于Java EE Web应用的Model-View-Controller（MVC）设计模式的应用框架，是MVC经典设计模式中的一个经典产品

**struts的优点如下：**

程序更加规范化

程序开发的效率提高

程序的可读性增加

程序的可维护性增加

3、struts的原理

以一个用户登录验证的例子说明。

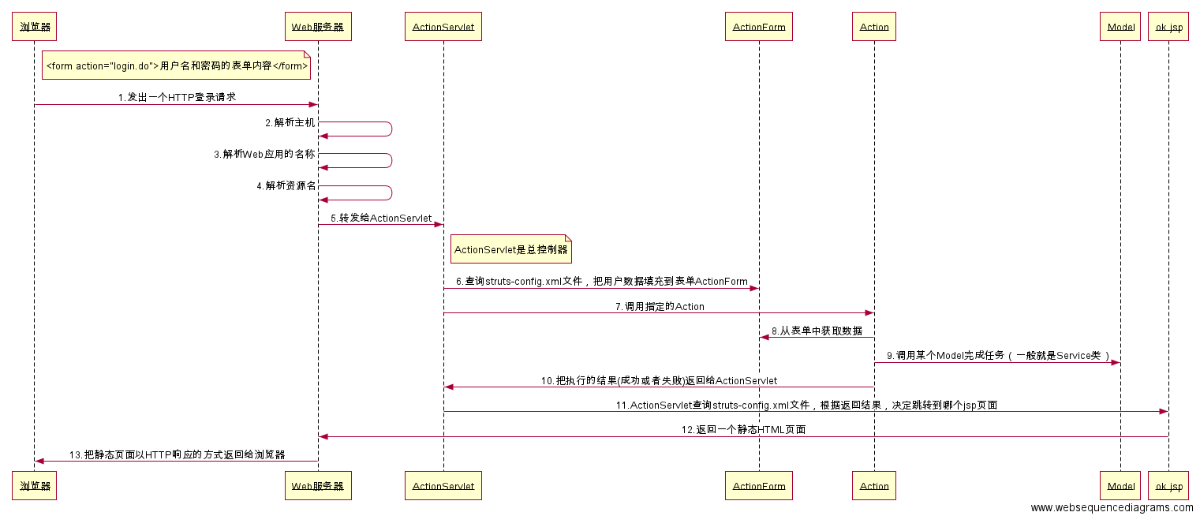


图1.7.1.1.1 时序图

（1）ActionServlet是总控制器，它是struts给我们提供的，我们无需自己写，只需要配置即可。ActionServlet管理struts-config.xml文件，而这个文件是struts的核心文件，该文件配置了ActionForm，还配置了Action，以及它们的对应关系等。

（2）ActionForm是表单，用于存放数据。开发ActionFrom必须要继承ActionForm类，这是规范。

（3）Action是分控制器，在Struts中，Action可以有多个.它的本质就是Servlet。开发一个Action也要继承Action类。

（4）Model(Java类，Service类，ejb等)

1.7.1.2 Hibernate

Hibernate，翻译过来是冬眠的意思，正好现在已经进入冬季，世间万物开始准备冬眠了。其实对于对象来说就是持久化。

持久化（Persistence），即把数据（如内存中的对象）保存到可永久保存的存储设备中（如磁盘）。持久化的主要应用是将内存中的对象存储在关系型的数据库中，当然也可以存储在磁盘文件中、XML数据文件中等等。

持久化是将程序数据在持久状态和瞬时状态间转换的机制。

JDBC就是一种持久化机制。文件IO也是一种持久化机制。

1、我们从三个角度理解一下Hibernate

（1）Hibernate是对JDBC进一步封装

原来没有使用Hiberante做持久层开发时，存在很多冗余，如：各种JDBC语句，connection的管理，所以出现了Hibernate把JDBC封装了一下，我们不用操作数据，直接操作它就行了。

（2）我们再从分层的角度来看

我们知道非常典型的三层架构：表示层，业务层，还有持久层。Hiberante也是持久层的框架，而且持久层的框架还有很多，比如：IBatis，Nhibernate，JDO，OJB，EJB等等。

（3）Hibernate是开源的一个ORM（对象关系映射）框架。

ORM，即Object-Relational Mapping，它的作用就是在关系型数据库和对象之间做了一个映射。从对象（Object）映射到关系（Relation），再从关系映射到对象。这样，我们在操作数据库的时候，不需要再去和复杂SQL打交道，只要像操作对象一样操作它就可以了（把关系数据库的字段在内存中映射成对象的属性）。

2、Hibernate的核心：

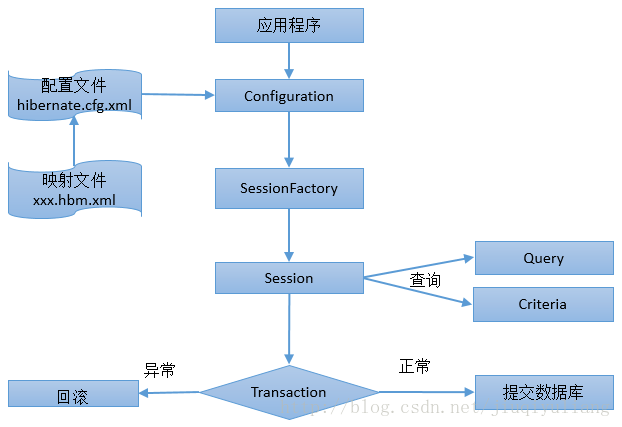


图1.7.1.2.1 核心图

从上图中，我们可以看出Hibernate六大核心接口，两个主要配置文件，以及他们直接的关系。Hibernate的所有内容都在这了。那我们从上到下简单的认识一下，每个接口进行一句话总结。

（1）Configuration接口:负责配置并启动Hibernate

（2）SessionFactory接口:负责初始化Hibernate

（3）Session接口:负责持久化对象的CRUD操作

（4）Transaction接口:负责事务

（5）Query接口和Criteria接口:负责执行各种数据库查询

注意：Configuration实例是一个启动期间的对象，一旦SessionFactory创建完成它就被丢弃了。

3、Hibernate的优点**：**

（1）更加对象化

以对象化的思维操作数据库，我们只需要操作对象就可以了，开发更加对象化。

（2）移植性

因为Hibernate做了持久层的封装，你就不知道数据库，你写的所有的代码都具有可复用性。

（3）Hibernate是一个没有侵入性的框架，没有侵入性的框架我们称为轻量级框架。对比Struts的Action和ActionForm，都需要继承，离不开Struts。Hibernate不需要继承任何类，不需要实现任何接口。这样的对象叫POJO对象。

（4）Hibernate代码测试方便。

（5）提高效率，提高生产力。

**总结：**

Hibernate可以使我们采用对象化的思维操作关系型数据库。

### 1.7.2 SSH与Jfinal的区别

1.7.2.1 SSH

SSH不是一个框架，而是多个框架（struts+spring+hibernate）的集成，是目前较流行的一种Web应用程序开源集成框架，用于构建灵活、易于扩展的多层Web应用程序。

Web层，就是MVC模式里面的“C”（controller）与“V”（View），负责控制业务逻辑层与页面的交互与展现，在SSH架构中，Struts充当controller，JSP充当View。

Service层（就是业务逻辑层），负责实现业务逻辑。业务逻辑层以DAO层为基础，通过对DAO组件的正面模式包装，完成系统所要求的业务逻辑。

DAO层，负责与持久化对象交互。该层封装了数据的增、删、查、改的操作。

PO，持久化对象。通过实体关系映射工具将关系型数据库的数据映射成对象，很方便地实现以面向对象方式操作数据库，该系统采用hibernate作为ORM框架。spring的作用贯穿了整个中间层，将Web层、Service层、DAO层及PO无缝整合，其数据服务层用来存放数据。

1.7.2.2 Jfinal

Final 是基于Java 语言的极速 web 开发框架，其核心设计目标是开发迅速、代码量少、学习简单、功能强大、轻量级、易扩展、Restful。在拥有Java语言所有优势的同时再拥有ruby、python等动态语言的开发效率。

1.7.2.3 二者比较

不可否认，SSH与jFinal都是非常优秀的架构，两者都有各自的优点，如何进行取舍取决于项目与项目组成员的实际情况。就本系统而言，使用SSH开发框架更为有利，方便成员代码地编写和逻辑地生成。

# 2.面向对象分析 OOA

## 2.1 OOA概述

顾名思义，OOA——面向对象的分析，就是运用面向对象方法对系统进行分析。OOA是一种分析方法，它充分地体现了面向对象的概念及原则。概而言之，OOA是一种基于对象、类、继承、封装、聚合、关联、消息和多态性等面向对象概念的分析方法。

### 2.1.1 OOA的必要性

在创建一个应用软件系统前，必须进行充分的分析，即明确所要解决的问题是什么。例如，开发一个人事管理系统软件，这个系统要实现什么样的功能？这就是一个分析问题。

设计强调的是问题的逻辑解决方案，即系统怎样才能达到要求。例如，人事信息管理系统如何实现远程查询？设计最终可以用硬件或软件方法来实现。

面向对象的分析与设计的精髓是按照对象的观点考虑问题域和逻辑解决方案。分析旨在发现并描述问题域中的对象。例如，在人事管理信息系统中，这些对象包括员工、[人事管理员](https://www.baidu.com/s?wd=%E5%9B%BE%E4%B9%A6%E7%AE%A1%E7%90%86%E5%91%98&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YzujwBrAR1njwhrjFWmhu-0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHR4PHmsPjc)等。

设计的重点在于定义逻辑软件对象，这些对象（具有属性和方法）最终能够用面向对象程序设计语言实现，即编写它们的类。

### 2.1.2 OOA的任务

OOA的基本任务：运用面向对象的概念对问题域进行分析与理解，对其中的事物和它们之间的关系产生正确的认识，将问题域中与系统责任有关的事物抽象为系统中的类和对象，定义这些类和对象的属性与操作，以及它们之间所形成的关系。最终的目的是产生一个符合用户需求，并能直接反映问题域的OOA模型及其规约。

### 2.1.3 OOA模型

完整的OOA文档包括需求模型、基本模型和辅助模型及其规约。基本模型表现了系统有哪些基本对象构成，每一类对象具有哪些属性和操作以及对象间的关系；需求模型清晰而规范地定义了系统的需求；辅助模型所含的各种图可在面向对象的建模中起到辅助作用。

模型基本框架如下：

**需求模型**

功能模型

业务模型

用例模型

**基本模型**

分析类图

**辅助模型**

顺序图

状态图

协作图

……

**模型规约**

对象层

特征层

关系层

图2.1 OOA模型

## 2.2 需求分析

### 2.2.1 需求分析概述

所谓"需求分析"，是指对要解决的问题进行详细的分析，弄清楚问题的要求，包括需要输入什么数据，要得到什么结果，最后应输出什么。

需求分析的基本任务：

（1）问题识别：双方确定对问题的综合需求，这些需求包括功能需求，性能需求，环境需求，用户界面需求。

（2）分析与综合，导出软件的逻辑模型

（3）编写文档：包括编写"需求规格说明书","初步用户使用手册","确认测试计划","修改完善软件开发计划"。

针对本系统，需求分析主要是了解餐饮行业管理者的需求有哪些，比如对员工基本信息进行修改、查询浏览等，对员工考勤记录以及相关薪金的浏览等。

#### 2.2.1.1需求分析的必要性

需求分析可以使得开发和测试更能够了解客户的需求，把一些技术难点和可能遇到的难点问题提出来，尽早解决，并且达到一致，便于以后的开发和测试。

#### 2.2.1.2需求分析模型

需求模型包括功能模型、业务模型和用例模型。功能模型即整体功能结构的建立。业务模型指的是用活动图对系统的业务用例实现业务流程进行描述；用例模型就是用用例对用户需求进行规范化描述。

### 2.2.2功能分析与功能模型

#### 2.2.2.1功能分析

常见的功能分析方法有功能分解法，结构化方法，信息建模及面向对象法。不同的方法基于不同的概念来建立系统地分析模型及设计模型，并给出不同的过程策略。

其中功能分解法是软件工程领域最早出现的建模方法。该方法从系统所需提供的功能出发来分析和设计系统。首先明确系统地各大功能模块，再对其进行逐步的分解，形成系统所需提供的大小合适的子功能。功能分解法图示如下：[[4]](#footnote-4)

系统

功能1

功能3

功能2

子功能1

子功能2

子功能3

子功能4

子功能5

子功能6

子功能

子功能

图2.2.1功能分解法

#### 2.2.2.2功能模型

2.2.2.2.1模型背景

我们通过对 “麻麻鱼府”进行调研了解到目前其现行的人事管理方式结合其所需将本系统主要分为人员管理、招聘管理、绩效管理、培训管理、奖惩管理、人事异动、薪金管理等主要功能模块。其中，人员管理包含人员信息录入、人员信息查看；招聘管理包含应聘信息录入、应聘信息查看、人才库浏览；人事异动包含人事异动管理；培训管理包含培训计划录入、培训计划查看、培训总结查看；绩效管理包含绩效信息登记、绩效信息查看；奖惩管理包含奖惩信息登记、奖惩信息查看；薪金管理包含薪金数据录入、薪金数据查看。

2.2.2.2.2功能结构图

应聘信息录入

应聘信息查看

人才库浏览

薪金数据录入

薪金数据查看

人事管理信息系统

招聘管理

人事异动

人员信息录入

人员信息查看

人事异动管理

奖惩信息登记

奖惩信息查看

绩效信息查看

薪金管理

奖惩管理

绩效管理

培训管理

培训计划录入

培训计划查看

培训总结查看

人员管理

绩效信息登记

图2.2.2.2.2功能结构图

2.2.2.2.3模型含义

人员管理模块：对员工基本信息进行管理，包括添加、修改、查询员工基本信息。其中除了最基本的姓名、联系方式等之外添加了一项体检结果。

招聘管理模块：分为应聘信息录入模块、应聘信息查看模块、人才库浏览模块。发布招聘信息模块是显示招聘计划；员工就职登记模块的功能是登记就职人员名单，给每个就职人员一个员工编号，用来唯一标识员工；历史信息模块统计本月新入人员名单。

绩效管理模块：主要包括绩效信息登记和绩效信息查看两个子模块。绩效信息登记模块包括对员工的绩效考核记录的添加、修改、删除与查询；绩效信息查看模块是各职位员工绩效考核记录的统计。

薪金管理模块：包括薪金数据录入模块和薪金数据查看模块两部分。包括查询员工的工资报表，查询方式有帐套查询，员工编号/姓名/工资月份查询。

奖惩管理模块：包括奖惩信息登记和奖惩信息查看两个子模块。可以对员工奖励记录、惩罚记录进行添加、删除、编辑和查询，以及按个人进行奖惩统计。

人事异动模块：主要有人事异动管理模块，是对员工异动记录进行添加、删除、编辑和查询。

培训管理模块：培训计划录入、培训计划查看、培训总结查看。负责对员工进行入职培训，入职培训的对象是刚进入公司的员工。[[5]](#footnote-5)

### 2.2.3业务建模

#### 2.2.3.1业务流程概述

中小型餐饮业人事管理系统业务是分模块来体现的。该系统包括以下几个模块：

1、招聘管理：

（1）因工作需要必须增加人员时，经管理层决策后纳入招聘计划，录入招聘信息

（2）应聘信息查看（招聘时录入前来应聘的人才的信息，其中包括体检），点击“入库”按钮则将该应聘者加入人才库（即企业所有员工信息表）

（3）人才库浏览（企业所有员工的信息都在该模块查看，可以从应聘信息查看模块添加企业员工）

2、人员管理：

（1）人员信息录入（该管理系统管理员信息录入）

（2）人员信息查看（查看所有该管理系统管理员信息）

3、培训管理:

（1）培训计划录入（员工因就职前业务操作不熟而要进行的相关培训）

（2）培训计划查看（包括历史信息与当前信息）

（3）培训总结查看（员工经培训后作出培训总结，包括历史信息与当前信息）

4、绩效管理

（1）绩效信息登记（对员工的工作能力与质量进行评估）

（2）绩效信息查看（查看员工的工作能力与质量）

5、薪金管理：

（1）薪金管理

（2）薪金数据录入（与绩效考核结果有密切关系）

（3）薪金数据查看（查看员工薪金详细情况）

6、人事异动管理：

（1）包括查询人事信息

（2）修改人事信息（外派、调岗、借调、待岗、晋升、平级调岗、离岗）

7、奖惩管理

（1）奖惩信息登记（记录员工相关奖惩信息，与薪金有密切关系）

（2）奖惩信息查看（查看员工奖惩信息详细情况）

#### 2.2.3.2 业务流程模型背景

业务流程模型采用活动图，活动图中包含的图形元素有动作状态、活动状态、动作流、分支与合并、分叉与汇合、泳道和对象流等。由于流程需要，我们把招聘管理、合同管理、人员管理放在一个流程图完成，表达了企业招收新员工的整个流程；在餐饮企业中，员工的入职培训是十分重要的，因此，员工培训管理流程图表达了企业在招收新员工后，对新员工进行入职培训的流程；根据我们调研的结果，餐饮企业中员工流动性较强，因此为人事异动管理做出了完整的流程图；薪金管理是每个企业必不可少的，因此有了薪金管理的流程图；而绩效管理、奖惩管理与薪金管理密切联系，因此作出绩效管理、奖惩管理的流程图十分必要。

根据各活动图各个活动状态等，员工培训管理活动图的泳道可以分为用户、系统中的员工培训管理模块；人事异动管理活动图的泳道可以分为用户、系统中的人事异动管理模块、系统中的员工基本信息管理模块；薪金管理活动图的泳道可以分为用户、系统中的薪金管理模块；绩效管理活动图的泳道可以分为用户、系统中的薪金管理模块；奖惩管理活动图的泳道可以分为用户、系统中的奖惩管理模块。

#### 2.2.3.3业务流程模型

2.2.3.3.1员工培训管理

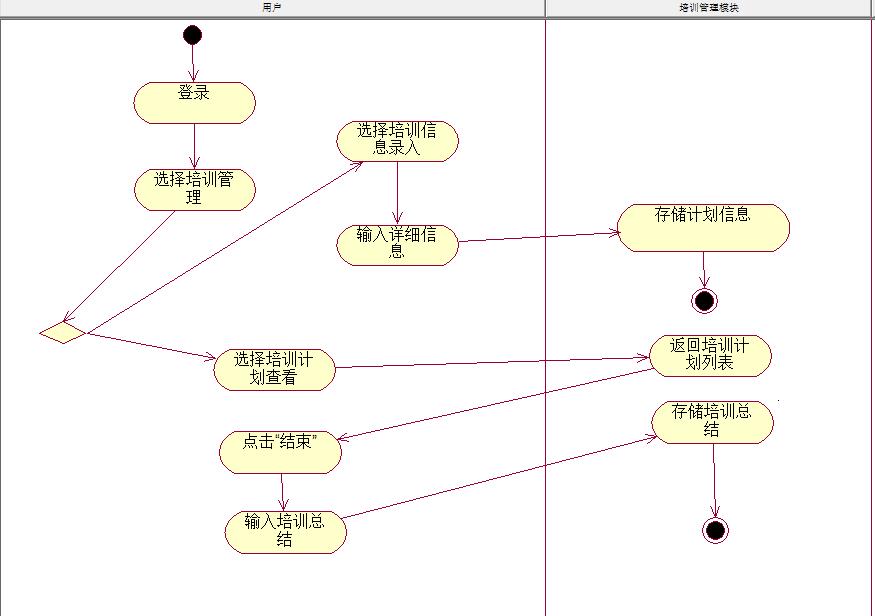


图2.2.3.3.1 员工培训管理活动图

模型含义：描述企业对员工进行培训的流程，用户在有培训员工需求的时候进入系统的员工培训管理模块，选择需要进行培训的员工安排培训计划；用户在需要查看员工培训情况的时候从员工培训管理模块请求员工培训信息并返回给用户；用户在需要查看员工培训总结的时候从员工培训管理模块请求员工培训信息并返回给用户。

2.2.3.3.2人事异动管理

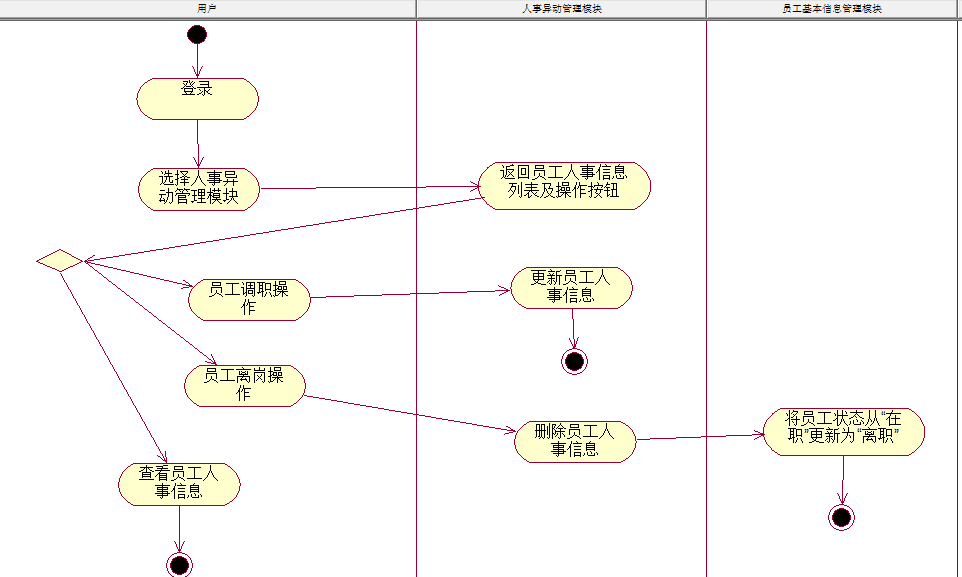


图2.2.3.3.2 人事异动管理活动图

模型含义：描述企业进行人事异动的流程图。用户在有人事异动需求（员工调职、离岗）的时候进入人事异动管理模块，系统将返回员工人事信息列表及操作按钮：选择员工调职操作的时候，人事异动管理模块将更新员工的职务信息；选择员工离岗操作的时候，人事异动模块将清除员工的人事信息（查询人事信息不会再显示）。用户在完成人事异动操作后再查询人事信息的时候系统将向人事异动管理模块请求人事信息并返回给用户。

2.2.3.3.3 薪金管理

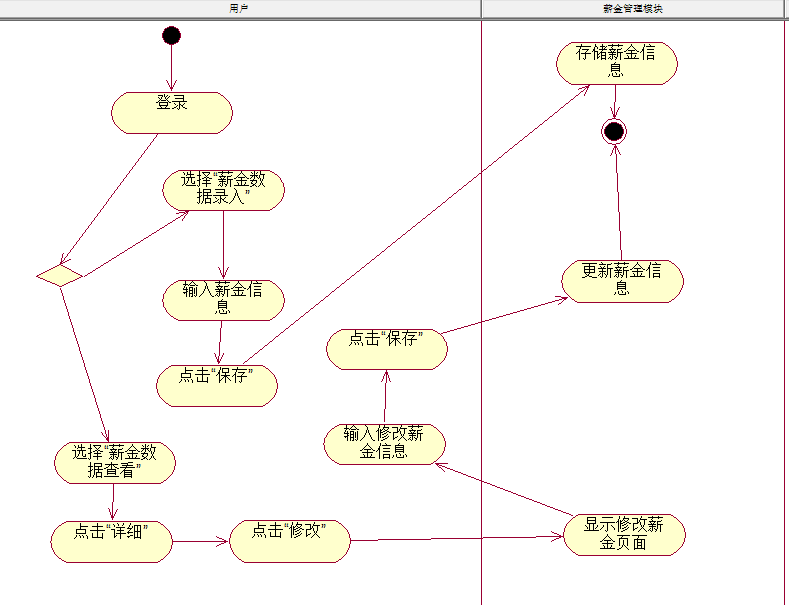


图2.2.3.3.3 薪金管理活动图

模型含义：描述企业薪金管理模块中修改基本工资业务流程和查询员工工资信息的流程。用户进入薪金管理模块选择薪金数据录入并填写其工资详细信息；用户查询员工工资信息的时候，系统将向薪金管理模块请求工资信息并返回给用户。

2.2.3.3.4 绩效管理

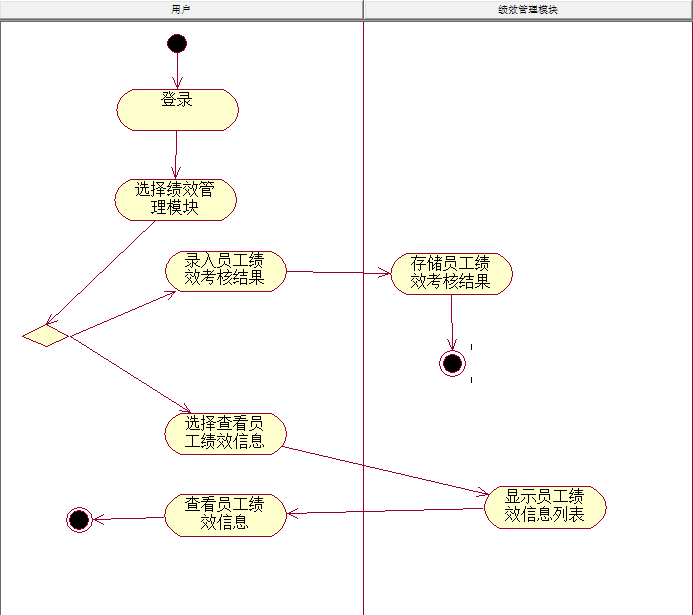


图2.2.3.3.4 绩效管理活动图

模型含义：描述了企业对员工进行绩效管理的流程。用户需要对员工绩效进行管理时进入绩效管理模块，选择录入员工绩效考核结果，同时储存在绩效管理模块。当用户需要查询员工绩效信息的时候，系统将向绩效管理模块请求员工绩效信息返回给用户。

2.2.3.3.5 奖惩管理

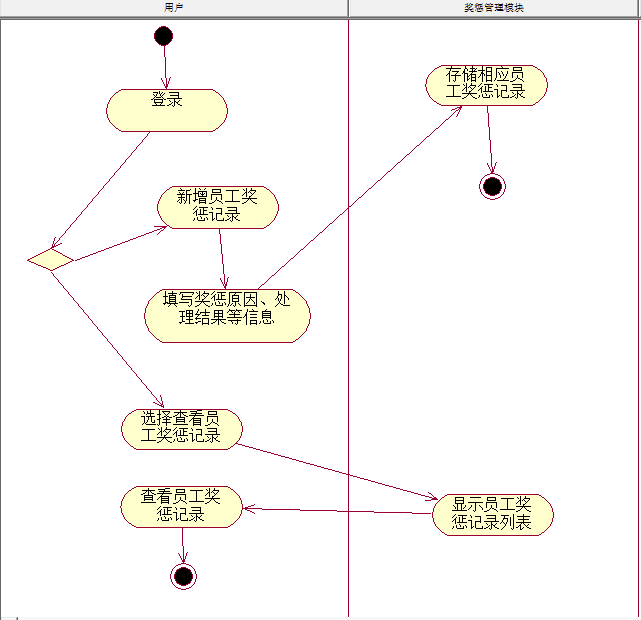


图2.2.3.3.5 奖惩管理活动图

模型含义：描述了人事管理者对员工奖励与惩罚的流程图。当用户需要增加员工奖惩记录时进入奖惩管理模块，选择录入员工奖惩相关信息（原因、结果等信息），同时，奖惩管理模块会根据记录奖惩的不同增加或者减少奖金金额；当用户需要查询员工奖惩记录时，系统将向奖惩管理模块请求员工奖惩信息返回给用户。

2.2.3.3.6 招聘管理活动图

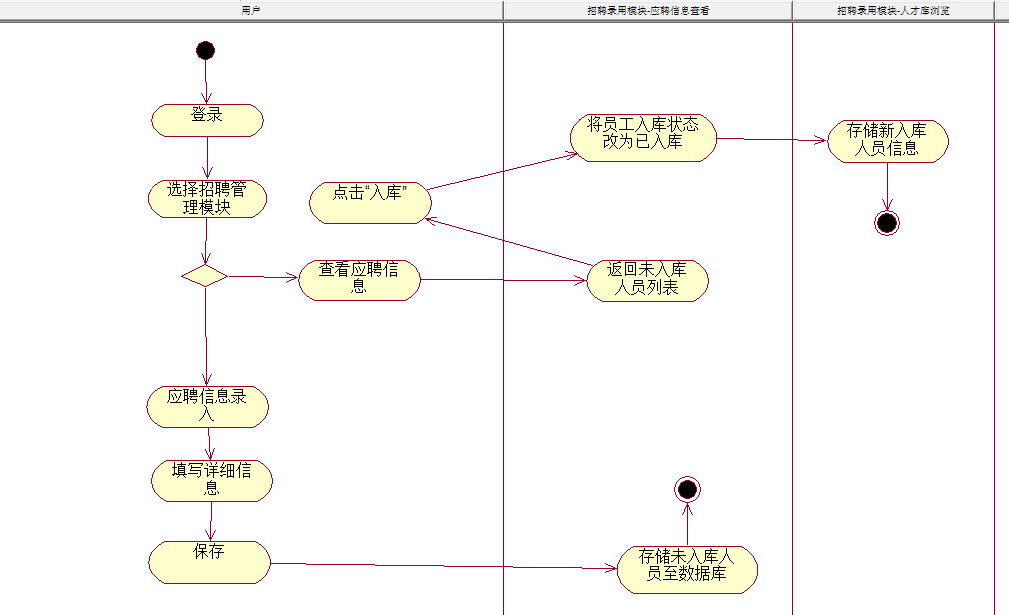


图2.2.3.3.6 招聘管理活动图

模型含义：描述企业招聘、应聘者应聘及员工基本信息登记的流程，用户在有应聘者应聘时进入系统的招聘录用模块，选择应聘信息录入应聘者详细信息；当用户需要查询应聘信息时，系统将向招聘录用模块-应聘信息查看模块请求应聘者信息返回给用户，当用户点击入库则将该应聘者信息纳入人才库（即聘用）。

#### 2.2.3.4业务流程模型规约[[6]](#footnote-6)

2.2.3.4.1 员工培训管理模型规约

该模型划分了用户、系统中的员工培训管理模块这两个泳道。用户是指使用中小型餐饮人事管理系统的用户，当然需要单独划分一个泳道。用户在有培训员工需求的时候进入系统的员工培训管理模块，选择需要进行培训的员工安排培训计划，同时将培训计划等信息存储在员工培训管理模块对应的数据表，在用户需要查看员工培训情况的时候再从员工培训管理模块请求员工培训信息并返回给用户；在用户需要查看员工培训总结的时候再从员工培训管理模块请求员工培训信息并返回给用户。因此在这个模型中要划分用户和员工培训管理模块两个泳道。

2.2.3.4.2 人事异动管理模型规约

该模型划分了用户、系统中的人事异动管理模块、系统中的员工基本信息管理模块这三个泳道。用户是指使用中小型餐饮人事管理系统的用户，当然需要单独划分一个泳道。用户在有人事异动需求（员工调职、离岗）的时候进入人事异动管理模块。用户在选择员工调职操作的时候人事异动管理模块将更新员工的职务信息并存储在该模块对应的数据表；用户在选择员工离岗操作的时候人事异动模块将清除员工的人事信息（查询人事信息不会再显示），并且人员管理模块将更新员工状态并存储在该模块对应的数据表。用户在完成人事异动操作后再查询人事信息的时候系统将向人事异动管理模块请求人事信息并返回给用户。因此在这个模型中要划分用户、系统中的人事异动管理模块、系统中的员工基本信息管理模块三个泳道。

2.2.3.4.3 薪金管理模型规约

该模型划分了用户、系统中的薪金管理模块这两个泳道。用户是指使用中小型餐饮人事管理系统的用户，当然需要单独划分一个泳道。用户在有修改员工基本工资需求的时候进入薪金管理模块，选择岗位并修改其基本工资，同时修改后的基本工资信息存储在薪金管理模块对应的数据表。用户在完成修改基本工资操作后再查询员工工资信息的时候，系统将向薪金管理模块请求工资信息并返回给用户。

2.2.3.4.4 绩效管理模型规约

该模型划分了用户、中的绩效管理模块这两个泳道。用户是指使用中小型餐饮人事管理系统的用户，当然需要单独划分一个泳道。用户在有员工绩效管理需求的时候进入绩效管理模块，选择录入员工绩效考核结果，同时绩效信息存储在绩效管理模块对应的数据表。当用户需要查询员工绩效信息的时候，系统将向绩效管理模块请求员工绩效信息返回给用户。因此在这个模型中要划分用户、绩效管理模块两个泳道。

2.2.3.4.5 奖惩管理模型规约

该模型划分了用户、系统中的奖惩管理模块这两个泳道。用户是指使用中小型餐饮人事管理系统的用户，当然需要单独划分一个泳道。用户在有增加员工奖惩记录需求的时候进入奖惩管理模块，选择录入员工奖惩相关信息（原因、结果等信息），同时奖惩记录将存储在奖惩管理模块对应的数据表。当用户需要查询员工奖惩记录的时候，系统将向奖惩管理模块请求员工奖惩信息返回给用户。因此在这个模型中要划分用户、奖惩管理模块两个泳道。[[7]](#footnote-7)

2.2.3.4.6 招聘管理模型规约

该模型划分了用户、系统中的招聘录用模块-应聘信息查看、系统中的招聘录用模块-人才库浏览这三个泳道。用户是指使用中小型餐饮人事管理系统的用户，当然需要单独划分一个泳道。用户在有应聘者应聘时进入系统的招聘录用模块，选择应聘信息录入录入应聘者详细信息，同时该应聘者信息将被储存在系统中的招聘录用模块-应聘信息查看模块对应的数据表；当用户需要查询应聘信息时，系统将向招聘录用模块-应聘信息查看模块请求应聘者信息返回给用户，当用户点击入库则将该应聘者信息纳入人才库（即聘用），同时在系统中的招聘录用模块-应聘信息查看模块将该应聘者入库状态改为已入库。因此在这个模型中要划分用户、系统中的招聘录用模块-应聘信息查看、系统中的招聘录用模块-人才库浏览这三个泳道。

### 2.2.4用例建模

#### 2.2.4.1用例模型

2.2.4.1.1模型背景

本系统涉及参与者只有人事管理者一个，因此参与者之间的关系不做讨论。

与参与者人事管理员关联的用例有人员管理、招聘管理、绩效管理、薪金管理、奖惩管理、人事异动、培训管理。其中，人员管理包含人员信息录入、人员信息查看；招聘管理包含应聘信息录入、应聘信息查看、人才库浏览；人事异动包含人事异动管理；培训管理包含培训计划录入、培训计划查看、培训总结查看；绩效管理包含绩效信息登记、绩效信息查看；奖惩管理包含奖惩信息登记、奖惩信息查看；薪金管理包含薪金数据录入、薪金数据查看。[[8]](#footnote-8)

2.2.4.1.2用例模型

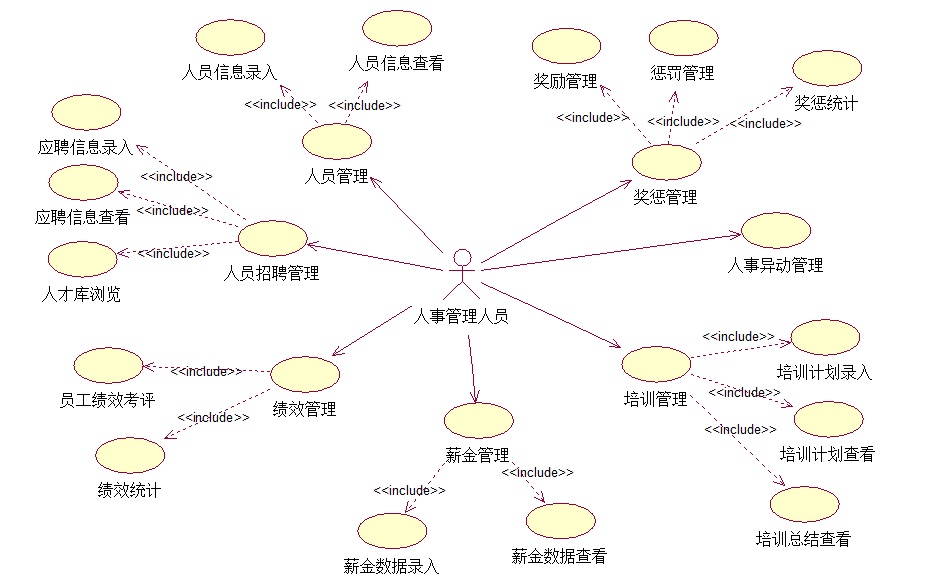


图2.2.4.1.2 人事管理者用例图

2.2.4.1.3模型含义

人事管理员在登录人事管理系统之后可以看到人员管理、招聘管理、绩效管理、薪金管理、奖惩管理、人事异动、培训管理七个大模块。

当选择人员管理时，可以进行人员信息录入、人员信息查看操作；

当选择招聘管理时，可以进行应聘信息录入、应聘信息查看、人才库浏览操作；

当选择绩效管理时，可以进行绩效信息登记、绩效信息查看；

当选择薪金管理时，可以进行薪金数据录入、薪金数据查看；

当选择奖惩管理时，可以进行奖惩信息登记、奖惩信息查看；

当选择人事异动时，可以进行查询人事信息、员工调职管理、员工离岗；

当选择培训管理时，可以进行培训计划录入、培训计划查看、培训总结查看。

#### 2.2.4.2用例模型规约

表2.2.4.2.1人员管理

|  |  |
| --- | --- |
| 用例名称 | 人员管理 |
| 用例描述 | 当企业增加或减少一名员工的时候是这个用例的开始，它处理有关员工基本信息增删改等问题，当同一名员工基本信息编辑完成的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 1 |
| 前置条件 | 已登录 |
| 后置条件 | 企业人员增加或减少 |
| 基本操作流程 | 点击“人员管理”，查看普通员工基本信息。 |
| 可选操作流程 | 点击“人员管理”，增加普通员工基本信息。  点击“人员管理”，删除普通员工基本信息。  点击“人员管理”，编辑普通员工基本信息。 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.2招聘管理

|  |  |
| --- | --- |
| 用例名称 | 招聘管理 |
| 用例描述 | 当人事管理者点击招聘管理的时候是这个用例的开始，它处理有关企业公布招聘信息、查看历史招聘信息等问题，当人事管理者完成对企业人事录用给的管理时，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 2 |
| 前置条件 | 已登录 |
| 后置条件 | 人才库增加；应聘信息增加 |
| 基本操作流程 | 点击“招聘管理”，选择“应聘信息录入”。 |
| 可选操作流程 | 点击“招聘管理”，选择“应聘信息查看”。  点击“招聘管理”，选择“人才库浏览”。 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 应聘信息录入、应聘信息查看、人才库浏览 |
| 被扩展的用例 | 无 |

表2.2.4.2.3 应聘信息录入

|  |  |
| --- | --- |
| 用例名称 | 应聘信息录入 |
| 用例描述 | 当人事管理者点击应聘信息录入的时候是这个用例的开始，它处理有关企业录入招聘信息、应聘者信息录入等问题，当人事管理者决定停止招聘新员工时，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 1 |
| 前置条件 | 已登录 |
| 后置条件 | 应聘信息增加；人才库增加 |
| 基本操作流程 | 点击“应聘信息录入”，编辑录入应聘者的详细信息。 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.4 应聘信息查看

|  |  |
| --- | --- |
| 用例名称 | 应聘信息查看 |
| 用例描述 | 当有应聘者前来应聘的时候是这个用例的开始，它处理有关应聘者信息查看等问题，当查看了某一名应聘者的信息时，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 2 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“应聘信息查看”，查看应聘者基本信息。 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.5 人才库浏览

|  |  |
| --- | --- |
| 用例名称 | 人才库浏览 |
| 用例描述 | 当人事管理者希望查看某一名员工的详细信息的时候是这个用例的开始，它处理有关员工信息查看等问题，当查看了某一名员工的信息时，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 3 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“人才库浏览”，查看普通员工基本信息。 |
| 可选操作流程 | 点击“人才库浏览”，点击“详细信息”，查看普通员工详细信息。  点击“人才库浏览”，点击“详细信息”，点击“修改”，编辑普通员工基本信息。  点击“人才库浏览”，点击“删除”，删除普通员工基本信息。 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.6 绩效管理

|  |  |
| --- | --- |
| 用例名称 | 绩效管理 |
| 用例描述 | 当企业员工开始工作的时候是这个用例的开始，它处理有关企业员工工作成绩和效率等问题，当企业员工结束工作的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 3 |
| 前置条件 | 已登录 |
| 后置条件 | 绩效信息登记改变 |
| 基本操作流程 | 点击“绩效管理”，选择“绩效信息登记”。 |
| 可选操作流程 | 点击“绩效管理”，选择“绩效信息查看”。 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 绩效信息登记、绩效信息查看 |
| 被扩展的用例 | 无 |

表2.2.4.2.7 绩效信息登记

|  |  |
| --- | --- |
| 用例名称 | 绩效信息登记 |
| 用例描述 | 当人事管理者点击绩效信息登记的时候是这个用例的开始，它处理有关企业录入绩效信息的问题，当人事管理者完成登记同一名员工的信息时，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 1 |
| 前置条件 | 已登录 |
| 后置条件 | 绩效信息增加 |
| 基本操作流程 | 点击“绩效信息登记”，编辑录入员工绩效的详细信息。 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.8 绩效信息查看

|  |  |
| --- | --- |
| 用例名称 | 绩效信息查看 |
| 用例描述 | 当人事管理者希望查看某一名员工的绩效信息的时候是这个用例的开始，它处理有关企业员工工作成绩和效率等问题，当人事管理者完成查看某一名员工的绩效信息的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 2 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“绩效信息查看”，查看普通员工绩效信息。 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.9 薪金管理

|  |  |
| --- | --- |
| 用例名称 | 薪金管理 |
| 用例描述 | 当企业员工开始工作的时候是这个用例的开始，它处理有关企业员工基本工资、提成工资和奖金等问题，当企业员工发工资的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 4 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“薪金管理”，选择“薪金数据录入”录入员工薪金详细信息，包括基本工资、提成、全勤奖、罚款、总计。 |
| 可选操作流程 | 点击“薪金管理”，选择“薪金数据查看”查看员工薪金详细信息。 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 薪金数据录入、薪金数据查看 |
| 被扩展的用例 | 无 |

表2.2.4.2.10 薪金数据录入

|  |  |
| --- | --- |
| 用例名称 | 薪金数据录入 |
| 用例描述 | 当人事管理者点击薪金数据录入的时候是这个用例的开始，它处理有关企业录入薪金数据的问题，当人事管理者完成录入同一名员工的薪金数据时，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 1 |
| 前置条件 | 已登录 |
| 后置条件 | 薪金数据增加 |
| 基本操作流程 | 点击“薪金数据录入”，编辑录入员工薪金的详细信息。 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.11 薪金数据查看

|  |  |
| --- | --- |
| 用例名称 | 薪金数据查看 |
| 用例描述 | 当人事管理者希望查看某一名员工的薪金数据的时候是这个用例的开始，它处理有关企业员工薪金金额等问题，当人事管理者完成查看某一名员工的薪金数据的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 2 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“薪金数据查看”，查看普通员工薪金相关数据。 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.12 奖惩管理

|  |  |
| --- | --- |
| 用例名称 | 奖惩管理 |
| 用例描述 | 当企业员工开始工作的时候是这个用例的开始，它处理有关企业员工奖励管理、惩罚管理和奖惩统计等问题，当企业员工结束工作的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 5 |
| 前置条件 | 已登录 |
| 后置条件 | 工资改变；人事异动改变 |
| 基本操作流程 | 点击“奖惩管理”，选择“奖励信息登记”编辑员工奖惩详细信息。 |
| 可选操作流程 | 点击“奖惩管理”，选择“奖惩信息查看”，点击“修改”，编辑员工奖惩详细信。  点击“奖惩管理”，选择“奖惩信息查看”查看员工奖惩详细信。 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 奖励信息登记、奖励信息查看 |
| 被扩展的用例 | 无 |

表2.2.4.2.13 奖惩信息登记

|  |  |
| --- | --- |
| 用例名称 | 奖惩信息登记 |
| 用例描述 | 当企业员工开始工作的时候是这个用例的开始，它处理有关企业员工奖励管理、惩罚管理和奖惩统计等问题，当企业员工结束工作的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 1 |
| 前置条件 | 已登录 |
| 后置条件 | 奖惩信息改变 |
| 基本操作流程 | 点击“奖惩信息登记”，编辑普通员工奖惩统计信息 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.14 奖惩信息查看

|  |  |
| --- | --- |
| 用例名称 | 奖惩信息查看 |
| 用例描述 | 当人事管理者希望查看某一名员工的奖惩信息的时候是这个用例的开始，它处理有关企业员工薪金金额等问题，当人事管理者完成查看某一名员工的奖惩信息的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 2 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“奖惩信息查看”，查看普通员工奖惩相关信息。 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.15 人事异动

|  |  |
| --- | --- |
| 用例名称 | 人事异动 |
| 用例描述 | 当一名员工进行人事调动的时候是这个用例的开始，它处理有关人事异动的编辑和授权等问题，当人事管理者完成了同一名员工的对话的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 6 |
| 前置条件 | 已登录 |
| 后置条件 | 人才库改变；应聘者信息改变 |
| 基本操作流程 | 单击“人事异动”模块，选择“人事异动管理”对相应员工进行晋升、离职等人事异动操作 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 员工人事异动、异动统计 |
| 被扩展的用例 | 无 |

表2.2.4.2.16人事异动管理

|  |  |
| --- | --- |
| 用例名称 | 人事异动管理 |
| 用例描述 | 当一名员工进行人事调动的时候是这个用例的开始，它处理有关人事异动的编辑和授权等问题，当人事管理者完成了同一名员工的对话的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 1 |
| 前置条件 | 已登录 |
| 后置条件 | 异动统计改变；员工基本信息改变；工资改变 |
| 基本操作流程 | 单击“人事异动管理”模块，选择相应员工进行“调岗”操作 |
| 可选操作流程 | 单击“人事异动管理”模块，选择相应员工进行“借调”操作  单击“人事异动管理”模块，选择相应员工进行“晋升”操作  单击“人事异动管理”模块，选择相应员工进行“待岗”操作  单击“人事异动管理”模块，选择相应员工进行“离职”操作 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.17 培训管理

|  |  |
| --- | --- |
| 用例名称 | 培训管理 |
| 用例描述 | 当企业管理者决定对新员工进行培训的时候是这个用例的开始，它处理有关企业新员工培训管理、培训信息查看等问题，当企业员工结束培训的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 7 |
| 前置条件 | 已登录 |
| 后置条件 | 工资改变；人事异动改变 |
| 基本操作流程 | 点击“培训管理”，选择“培训计划录入”录入员工培训计划。 |
| 可选操作流程 | 点击“培训管理”，选择“培训计划查看”查看员工培训详细情况。  点击“培训管理”，选择“培训总结查看”查看员工培训后总结的详细情况。 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 培训计划查看、培训总结查看 |
| 被扩展的用例 | 无 |

表2.2.4.2.18 培训计划录入

|  |  |
| --- | --- |
| 用例名称 | 培训计划录入 |
| 用例描述 | 当企业管理者决定对员工进行培训的时候是这个用例的开始，它处理有关企业新员工培训管理、培训信息查看等问题，当企业员工结束培训的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 1 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“培训计划录入”，编辑员工培训计划 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.19 培训计划查看

|  |  |
| --- | --- |
| 用例名称 | 培训计划查看 |
| 用例描述 | 当企业管理者决定对员工进行培训的时候是这个用例的开始，它处理有关企业新员工培训管理、培训信息查看等问题，当企业员工结束培训的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 2 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“培训计划查看”，查看员工培训计划，包括当前和往期的培训计划 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

表2.2.4.2.20 培训总结查看

|  |  |
| --- | --- |
| 用例名称 | 培训总结查看 |
| 用例描述 | 当员工完成培训后写培训总结的时候是这个用例的开始，它处理有关企业员工培训心得查看等问题，当企业员工完成培训总结的时候，它就结束了 |
| 参与者 | 人事管理者 |
| 优先级 | 3 |
| 前置条件 | 已登录 |
| 后置条件 | 无 |
| 基本操作流程 | 点击“培训总结查看”，查看员工培训总结 |
| 可选操作流程 | 无 |
| 被泛化的用例 | 无 |
| 被包含的用例 | 无 |
| 被扩展的用例 | 无 |

## 2.3 建立基本模型（OOA类图）

### 2.3.1基本模型概述

类图显示了系统的静态结构，而系统的静态结构构成了系统的概念基础。类图不仅描述了系统内部信息的结构，也包含了系统的内部行为，系统通过自身行为与外部事物进行交互。因此类图作为系统的基本模型是非常合适的。

中小型餐饮业人事管理系统类图包含人事管理者、招聘计划类、员工基本信息类、合同信息类、培训信息类、绩效考核类、考勤记录类、工资信息类、奖惩记录类。[[9]](#footnote-9)

### 2.3.2基本模型

2.3.2.1模型背景

该模型包含一个参与者：人事管理者，六个类：人事管理者类、应聘人员信息类、奖惩记录类、薪金信息类、绩效考核类和培训基本信息类。其中，应聘人员信息类、奖惩记录类、薪金信息类、绩效考核类和培训基本信息类与参与者人事管理者关联。

2.3.2.2基本模型

中小型餐饮业人事管理系统应当包括以下几个类：

1、人事管理者类

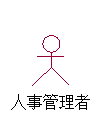


图2.3.2.2.1 人事管理者类图

2、应聘人员信息类

C:\Users\Administrator\AppData\Roaming\Tencent\Users\415689819\QQ\WinTemp\RichOle\RXF[T$[8LXTVSGNA4V$QTUV.png

图2.3.2.2.2 应聘人员信息类图

3、薪金信息类：

C:\Users\Administrator\AppData\Roaming\Tencent\Users\415689819\QQ\WinTemp\RichOle\NXQ[KP3D`QS2A`Y)I]AWY~S.png

图2.3.2.2.3 薪金信息类图

4、绩效考核类

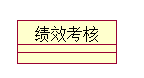


图2.3.2.2.4 考核绩效类图

5、培训信息类：

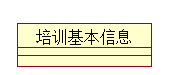


图2.3.2.2.5 培训信息类图

6、奖惩记录类：

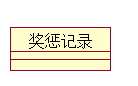


图2.3.2.2.6 奖惩记录类图

7、总体类图：

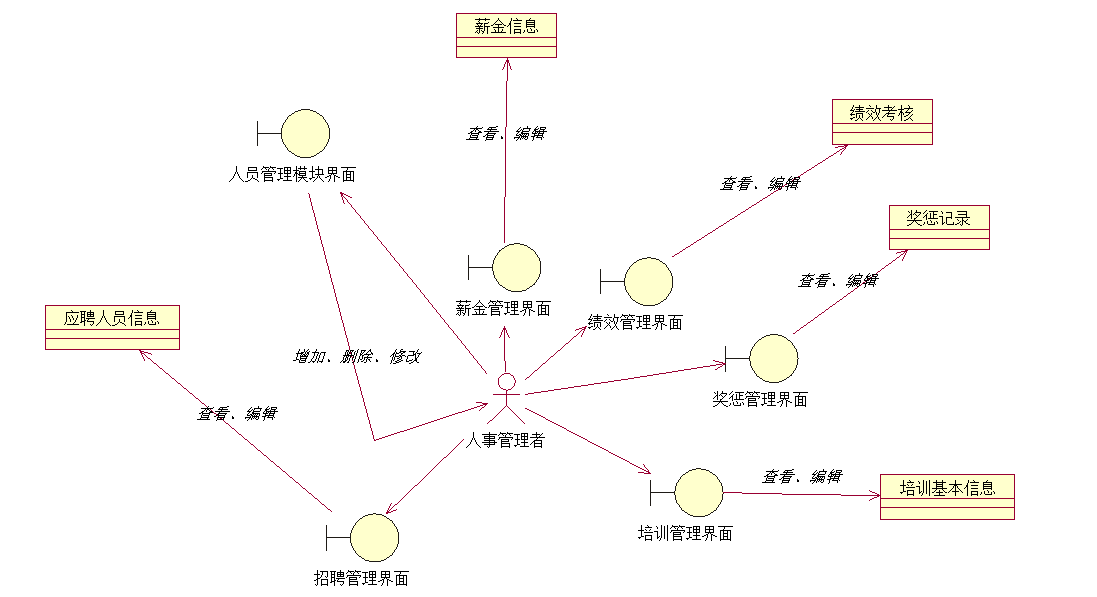


图2.3.2.2.7 总体类图

2.3.2.3模型含义

参与者人事管理者可以查看、编辑应聘人员信息类，可以在奖惩管理界面查看、编辑奖惩记录类，可以在薪金管理界面查看、编辑薪金信息类，可以在绩效考核界面查看、编辑绩效考核类，可以在培训管理界面查看、编辑培训基本信息类，可以在人员管理模块界面增加、删除、修改应聘人员信息类。

## 2.4建立辅助模型

### 2.4.1辅助模型概述

由于仅靠基本模型等是不足以良好的向客户展示本系统的，因此辅助模型的存在具有一定必要性。辅助模型包括顺序图、协作图和状态机图等。

### 2.4.2顺序图

#### 2.4.2.1顺序图概念

顺序图是将交互关系表示为一个二维图。纵向是时间轴，时间沿竖线向下延伸。横向轴代表在协作中各独立对象的类元角色。类元角色用生命线表示。当对象存在时，角色用一条虚线表示，当对象的过程处于激活状态时，生命线是一个双道线。一般用于确认和丰富一个使用情境的逻辑。

#### 2.4.2.2顺序图

2.4.2.2.1人事异动管理顺序图

2.4.2.2.1.1模型背景

该模型包含三个对象：用户、人事异动管理模块、数据管理；十条消息：用户发送消息

1给人事异动管理模块，人事异动管理模块发送消息

2给数据管理，数据管理返回消息

3给人事异动管理模块，人事异动管理模块返回消息

4给用户，用户发送消息

5给人事异动管理模块，人事异动管理模块返回消息

6给用户，用户发送消息

7给人事异动管理模块，人事异动管理模块发送消息

8给数据管理，数据管理返回消息

9给人事异动管理模块，人事异动管理模块返回消息

10给用户。

2.4.2.2.1.2人事异动管理顺序图

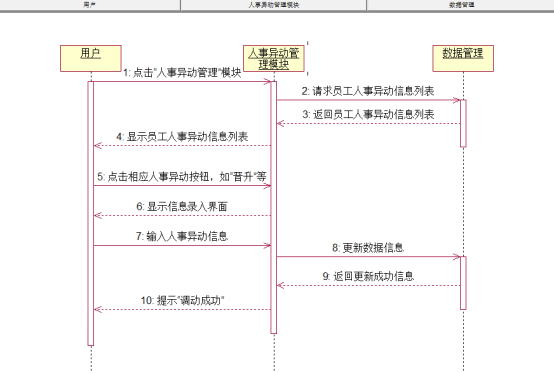
****

图2.4.2.2.1.2 人事异动管理顺序图

2.4.2.2.3.3模型含义

此模块内容如下：点击“人事异动管理”模块，向数据库请求员工人事异动信息列表，显示员工信息后，点击相应人事异动按钮，如“晋升”等，在信息录入界面输入人事异动信息，数据库更新数据，返回更新成功，界面提示“调动成功”。

2.4.2.2.2薪金数据录入顺序图

2.4.2.2.2.1模型背景

该模型包含三个对象：用户、工资管理模块、数据管理；八条消息：用户发送消息1给工资管理模块，工资管理模块返回消息2给用户，用户发送消息3给工资管理模块，用户发送消息4给工资管理模块，工资管理模块发送消息5给数据管理，数据管理返回消息6给工资管理模块，工资管理模块返回消息7给用户。

2.4.2.2.2.2薪金数据录入顺序图

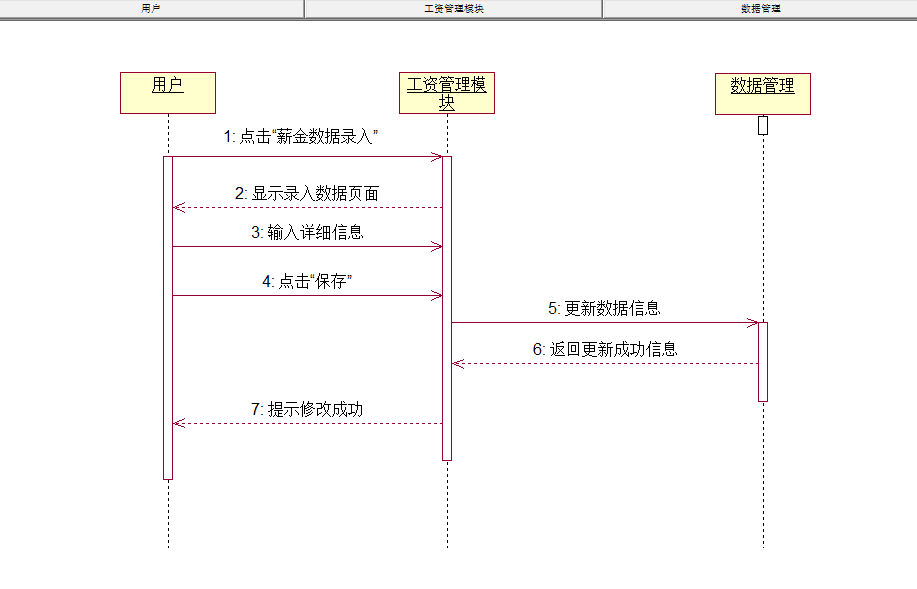


图2.4.2.2.2.2 薪金数据录入顺序图

2.4.2.2.2.3模型含义

此模块包括内容如下：点击“薪金数据录入”，在显示录入数据页面上输入薪金详细信息，点击保存按钮，数据库更新数据信息，返回更新成功信息，界面提示修改信息成功。

2.4.2.2.3薪金数据修改顺序图

2.4.2.2.3.1模型背景

该模型包含三个对象：用户、工资管理模块、数据管理；八条消息：用户发送消息

1给工资管理模块，工资管理模块发送消息

2给数据管理，数据管理返回消息

3给工资管理模块，工资管理模块返回消息

4给用户，用户发送消息

5给工资管理模块，用户发送消息

6给工资管理模块，用户发送消息

7给工资管理模块，工资管理模块发送消息

8给数据管理，数据管理返回消息

9给工资管理模块，工资管理模块返回消息

10给用户。

2.4.2.2.3.2薪金数据修改顺序图

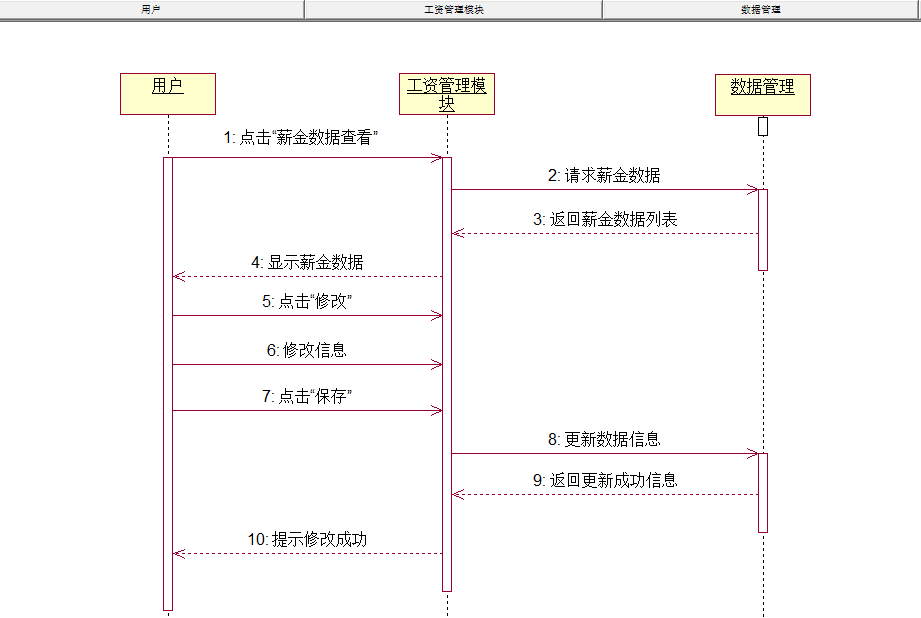


图2.4.2.2.3.2 薪金数据修改顺序图

2.4.2.2.3.3模型含义

此模块包括内容如下：点击“薪金数据查看”，在返回的薪金数据列表中点击修改，修改员工薪金相关信息并点击保存按钮，数据库更新数据信息，返回更新成功信息，界面提示修改信息成功。

2.4.2.2.4人事异动管理顺序图

2.4.2.2.4.1模型背景

该模型包含三个对象：用户、招聘管理模块、数据管理；五条消息：用户发送消息

1给招聘管理模块，招聘管理模块返回消息

2给用户，用户发送消息

3给招聘管理模块，招聘管理模块发送消息

4给数据管理，数据管理返回消息

5给招聘管理模块。

2.4.2.2.4.2招聘管理顺序图

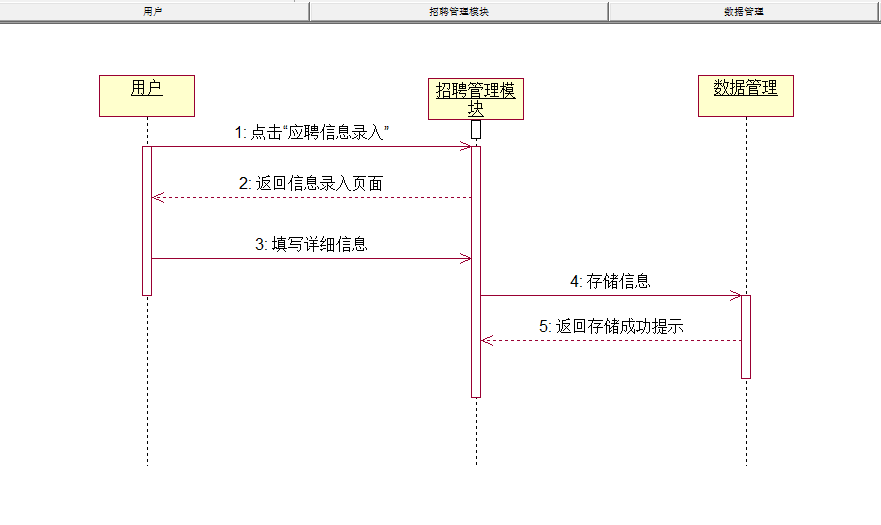
****

图2.4.2.2.4.2 招聘管理顺序图

2.4.2.2.4.3模型含义

此模块内容如下：点击“应聘信息录入”模块，在信息录入页面填写应聘者详细信息，数据库更新数据，返回存储成功提示。

### 2.4.3状态图

#### 2.4.3.1状态图概念

状态图(Statechart Diagram)是描述一个实体基于事件反应的动态行为，显示了该实体如何根据当前所处的状态对不同的事件做出反应。通常我们创建一个UML状态图是为了以下的研究目的：研究类、角色、子系统、或组件的复杂行为。

#### 2.4.3.2状态图

2.4.3.2.1人事异动状态图

2.4.3.2.1.1模型背景

该模型包含一个进入节点；八个状态：

用户登录成功、外派、员工调至相应岗位、员工调至借调后职务、待岗、员工晋升至相应职位、员工离岗、该人事信息清空；一个退出节点。

用户登录成功后，状态分为外派、员工调至相应岗位、员工调至借调后职务、待岗、员工晋升至相应职位、员工离岗，员工外派进入状态外派，员工调岗进入状态员工调至相应岗位，员工借调进入状态员工调至借调后职务，员工待岗进入状态待岗，员工晋升进入状态员工晋升至相应职位，员工离岗进入状态员工离岗，如果状态是员工离岗，则进入状态该人事信息清空。

2.4.3.2.1.2人事异动状态图

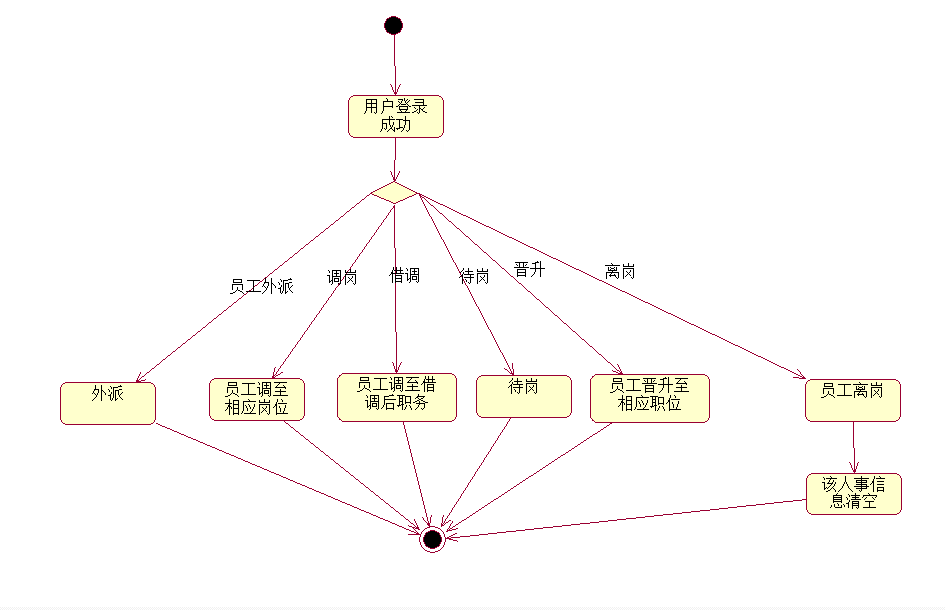
****

图2.4.3.2.1 人事异动状态图

2.4.3.2.1.3模型含义

人事管理员登录成功后，可以选择员工外派、员工调岗、员工借调、员工借岗、员工晋升、员工离岗，操作后员工的职务状态发生相应的改变。此外，员工离岗操作后，应该清除人事信息，员工基本信息从“在职”状态变为“离职”状态。

2.4.3.2.2员工培训状态图

2.4.3.2.2.1模型背景

该模型包含一个进入节点；五个状态：用户登录成功、员工开始培训、员工结束培训、员工数据录入正在培训员工列表、员工数据录入已培训员工列表；一个退出节点。

用户登录成功后，状态分为员工开始培训、员工结束培训，当选择了未培训员工，点击开始培训时，进入状态员工开始培训；如果状态是员工开始培训，则进入状态员工数据录入正在培训员工列表；当选择了正在培训员工，点击结束培训时，进入状态员工结束培训；如果状态是员工结束培训，则进入状态员工数据录入已培训员工列表。

2.4.3.2.2.2员工培训状态图

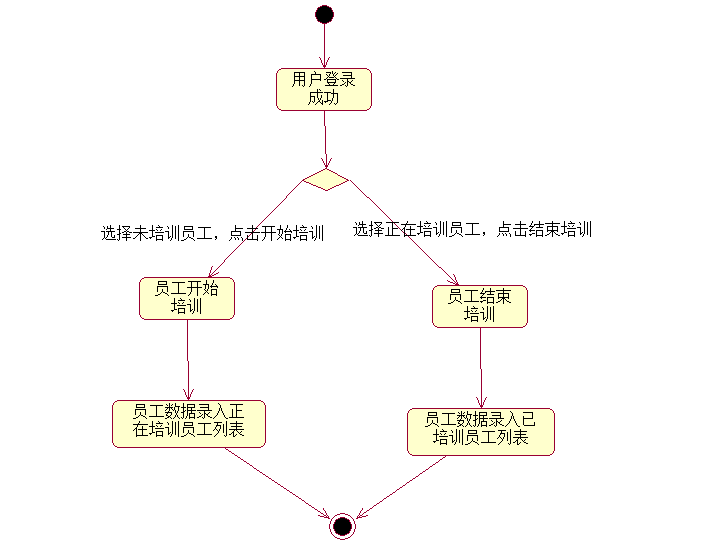
****

图2.4.3.2.2 员工培训状态图

2.4.3.2.2.3模型含义

人事管理员在成功登录后，选择未培训员工安排培训计划，员工进入“正在培训”的状态，数据录入培训管理模块对应的数据表（正在培训员工列表），培训结束后选择正在培训的员工点击结束培训，数据录入培训管理模块对应的数据表（已培训员工列表）。

### 2.4.4活动图

#### 2.4.1 活动图概念

业务建模（Business Modeling）是一种建模方法的集合，目的是对业务进行建模。这方面的工作可能包括了对[业务流程](http://baike.baidu.com/view/1368133.htm)建模，对[业务组织](http://baike.baidu.com/view/2290452.htm)建模，改进业务流程，领域建模等方面。此处，我们使用活动图对业务流程进行建模。活动图阐明了业务[用例](http://baike.baidu.com/view/706238.htm)实现的工作流程。业务工作流程说明了业务为向所服务的业务主角提供其所需的价值而必须完成的工作。工作流程通常包括一个基本工作流程和一个或多个备选工作流程。工作流程的结构使用活动图来进行说明。

#### 2.4.2活动图

2.4.2.1员工培训管理

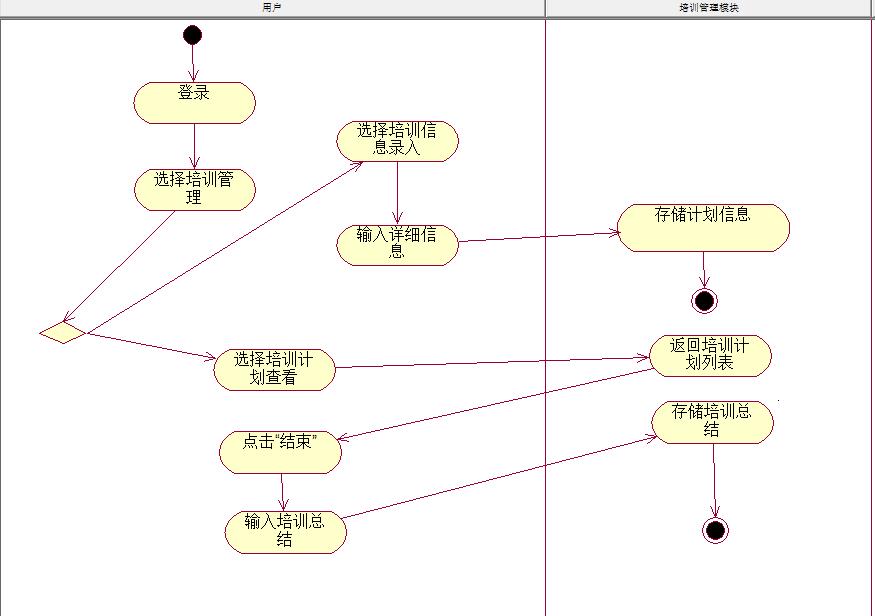


图2.4.2.1 员工培训管理活动图

模型含义：描述企业对员工进行培训的流程，用户在有培训员工需求的时候进入系统的员工培训管理模块，选择需要进行培训的员工安排培训计划；用户在需要查看员工培训情况的时候从员工培训管理模块请求员工培训信息并返回给用户；用户在需要查看员工培训总结的时候从员工培训管理模块请求员工培训信息并返回给用户。

2.4.2.2人事异动管理

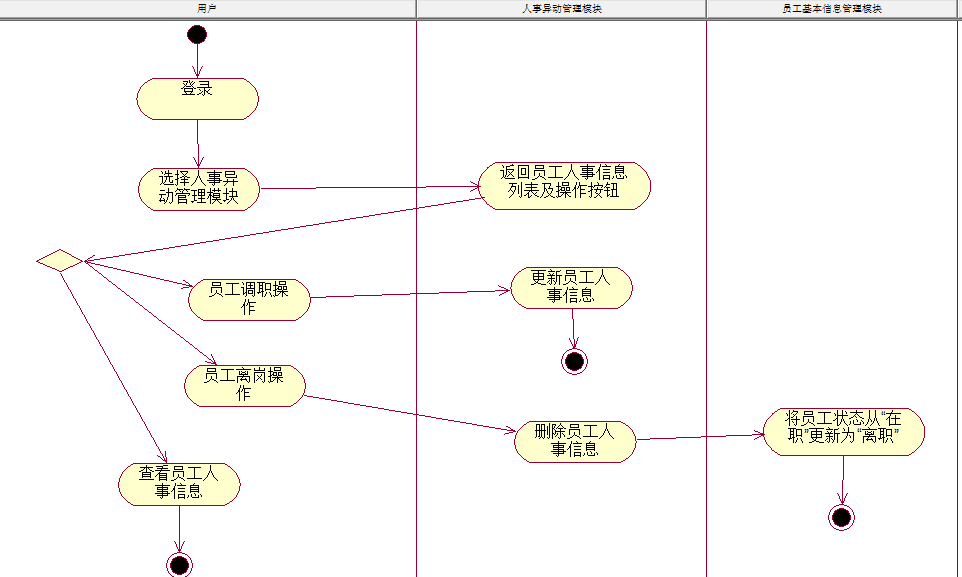


图2.4.2.2 人事异动管理活动图

模型含义：描述企业进行人事异动的流程图。用户在有人事异动需求（员工调职、离岗）的时候进入人事异动管理模块，系统将返回员工人事信息列表及操作按钮：选择员工调职操作的时候，人事异动管理模块将更新员工的职务信息；选择员工离岗操作的时候，人事异动模块将清除员工的人事信息（查询人事信息不会再显示）。用户在完成人事异动操作后再查询人事信息的时候系统将向人事异动管理模块请求人事信息并返回给用户。

2.4.2.3 薪金管理

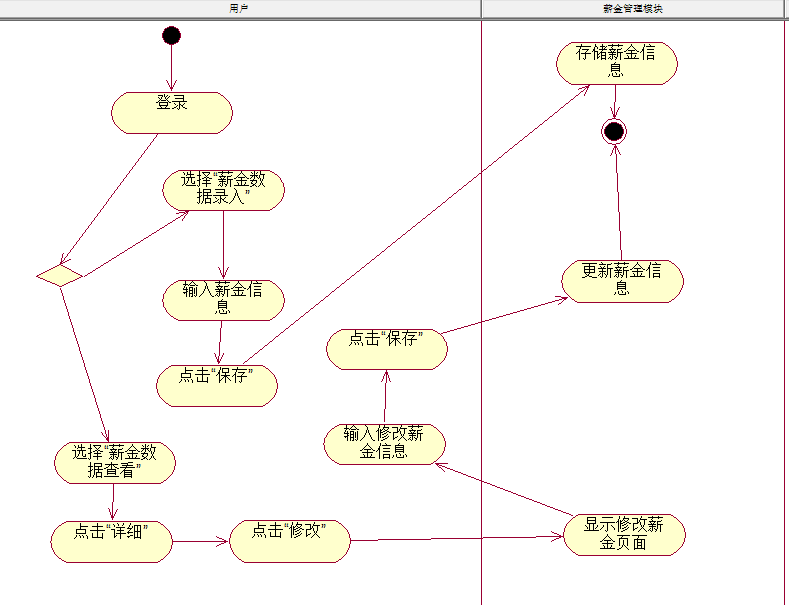


图2.4.2.3 薪金管理活动图

模型含义：描述企业薪金管理模块中修改基本工资业务流程和查询员工工资信息的流程。用户进入薪金管理模块选择薪金数据录入并填写其工资详细信息；用户查询员工工资信息的时候，系统将向薪金管理模块请求工资信息并返回给用户。

2.4.2.4 绩效管理

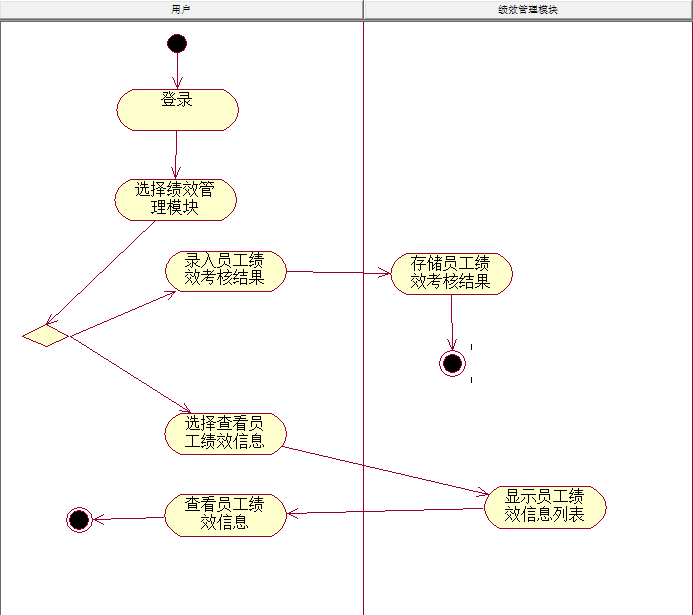


图2.4.2.4 绩效管理活动图

模型含义：描述了企业对员工进行绩效管理的流程。用户需要对员工绩效进行管理时进入绩效管理模块，选择录入员工绩效考核结果，同时储存在绩效管理模块。当用户需要查询员工绩效信息的时候，系统将向绩效管理模块请求员工绩效信息返回给用户。

2.4.2.5 奖惩管理

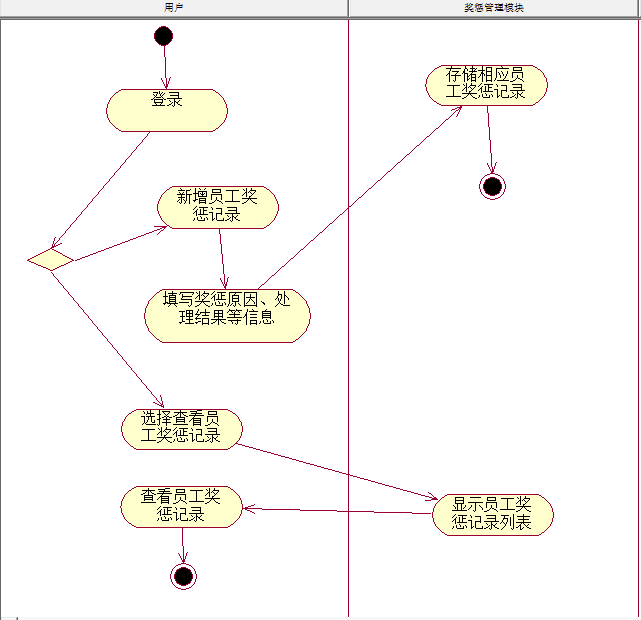


图2.4.2.5 奖惩管理活动图

模型含义：描述了人事管理者对员工奖励与惩罚的流程图。当用户需要增加员工奖惩记录时进入奖惩管理模块，选择录入员工奖惩相关信息（原因、结果等信息），同时，奖惩管理模块会根据记录奖惩的不同增加或者减少奖金金额；当用户需要查询员工奖惩记录时，系统将向奖惩管理模块请求员工奖惩信息返回给用户。

2.4.2.6 招聘管理活动图

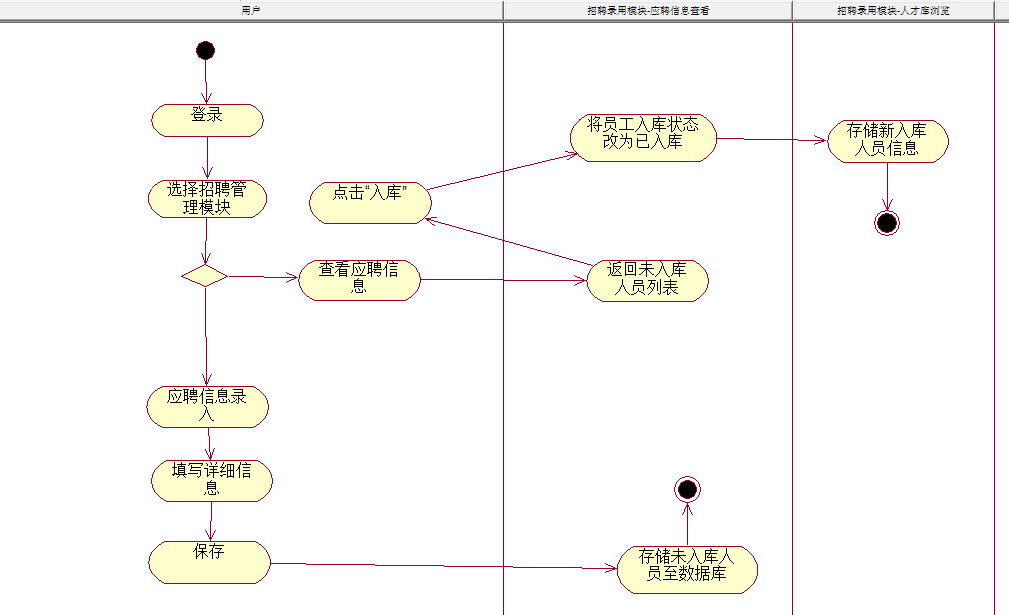


图2.4.2.6 招聘管理活动图

模型含义：描述企业招聘、应聘者应聘及员工基本信息登记的流程，用户在有应聘者应聘时进入系统的招聘录用模块，选择应聘信息录入应聘者详细信息；当用户需要查询应聘信息时，系统将向招聘录用模块-应聘信息查看模块请求应聘者信息返回给用户，当用户点击入库则将该应聘者信息纳入人才库（即聘用）。

# 3.面向对象设计 OOD

**3.1 OOD概述**

### 3.1.1概念

系统设计是新系统的物理设计阶段。根据系统分析阶段所确定的新系统的逻辑模型、功能要求，在用户提供的环境条件下，设计出一个能在计算机网络环境上实施的方案，即建立新系统的物理模型。即系统设计是在面向对象分析模型的基础上运用面向对象方法，产生符合具体实现条件的面向对象设计模型。

面向对象分析与面向对象设计的关系：

（1）面向对象的设计就是在面向对象分析的基础上运用面向对象方法，主要解决与现实有滚得问题，目标是产生一个符合现实条件的面向对象设计模型。与现实条件有关的因素有:图形用户系统、硬件、操作系统、网络、数据管理系统和编辑语言等。

（2）由于面向对象设计以面向对象分析模型为基础且面向对象分析与面向对象设计采用一致的表示方法，这使得从面向对象分析到面向对象设计不存在转换，只需做必要的修改和调整，或补充某些细节，并增加几个与现实关联的独立部分即可。因此面向对象分析与面向对象设计之间不存在分析与设计的鸿沟，二者紧密衔接，大大降低了从面向对象分析与面向对象设计的难度、工作量和出错频率。

（3）面向对象分析主要针对问题域，识别有关的对象以及它们之间的关系，产生一个映射问题域，满足用户需求，独立于实现的面向对象分析模型。面向对象设计主要解决与实现有关的问题，基于面向对象分析模型，针对具体的软、硬件条件（如机器、网络、OS、GUI、DBMS等）产生一个可实现的面向对象设计模型。[[10]](#footnote-10)

### 3.1.2系统设计模型

在面向对象分析阶段只考虑问题域和系统操作责任，在面向对象设计阶段要考虑与具体实现的问题。OOD模型如下：

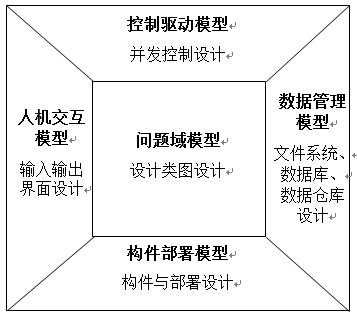


图3.1.2 系统设计模型

从图3.1中可以发现，面向对象设计模型包括五个主要部分——一个核心部分加四个外围部分：问题域部分、人机交换部分、控制驱动部分、数据管理部分、构建及部署部分。同时，面向对象设计模型每个部分仍采用面向对象分析的概念和表示方法，只是在辅助模型中要增加分别于描述构件模型和部署的构件图和部署图。

面向对象设计过程就是设计面向对象设计模型的五个部分，主要包括：问题域部分的设计、人机交互部分的设计、控制流管理部分的设计、数据管理部分的设计、构件部署设计。前四项不强调次序，每个部分均采用与面向对象分析一致的概念、表示法及活动，但具有自己独特的策略。进行构件部署设计要在其前面四个部分完成后进行。

### 3.1.3设计原则

（1）合理性原则。在系统设计时，充分考虑系统的容量及功能的扩充，方便系统扩容及平滑升级。系统对运行环境(硬件设备、软件操作系统等)具有较好的适应性，不依赖于某一特定型号计算机设备和固定版本的操作系统软件。

（2）可靠性原则。是指信息系统抗干扰的能力及受外界干扰时的恢复能力。一个成功的信息系统必须具有较高的可靠性，如安全保密性、检错及纠错能力、抗病毒能力等。

（3）可维护性原则。系统操作简单，实用性高，具有易操作、易维护的特点，系统具有专业的管理维护终端，方便系统维护。并且，系统具备自检、故障诊断及故障弱化功能，在出现故障时，能得到及时、快速地进行自维护。

（4）经济性原则。是指在满足系统需求的前提下，尽量节约成本。一方面，在硬件投资上不能盲目追求技术上的先进，而应以满足应用需要为前提。另一方面，信息系统设计中应尽量避免不必要的复杂化，各模块应尽量简洁，以便缩短处理流程、减少处理费用。

（5）开放性原则。系统设计遵循开放性原则，能够支持多种硬件设备和网络系统，软硬件支持二次开发。各系统采用标准数据接口，具有与其他信息系统进行数据交换和数据共享的能力。[[11]](#footnote-11)

### 3.1.4设计模式

设计模式的产生：如果某一解决方案对某类问题都很有用，就把它总结出来，从而产生了设计模式。设计模式是优秀的设计范例；从优秀设计方案中发现和总结出来的经验；在实践中反复出现的设计问题的优秀解决方案；设计者相互交流的基本术语：设计语言；培养优秀设计师的一条捷径。

设计模式的原则：

（1）Liskov 替换原则LSP。任何基类可以出现的地方，子类一定可以出现。LSP是继承复用的基石，只有当衍生类可以替换掉基类，软件单位的功能不受到影响时，基类才能真正被复用，而衍生类也能够在基类的基础上增加新的行为。

（2）开放-封闭原则OCP。关于开放封闭原则，其核心的思想是：软件实体应该是可扩展，而不可修改的。也就是说，对扩展是开放的，而对修改是封闭的。

（3）单一职责原则SRP。该原则规定一个类只有一个发生变化的原因，并且具有内聚性。

（4）接口隔离原则ISP。客户端不应该依赖它不需要的接口；一个类对另一个类的依赖应该建立在最小的接口上。

（5）依赖倒置原则DIP。高层模块不应该依赖低层模块，两者都应该依赖其抽象；抽象不应该依赖细节；细节应该依赖抽象。

**3.2问题域设计**

### 3.2.1问题域设计任务

（1）问题域部分就是对面向对象分析的结果按实现条件进行补充与调整。进行问题域部分设计，要继续运用面向对象分析的方法，包括概念、表示法及一部分策略。不但要根据实现条件进行面向对象设计，而且由于需求变化或新发现了错误，也要对面向对象分析的结果进行修改。问题域设计的重点是对面向对象分析的结果进行补充与调整，这部分工作主要不是细化，但面向对象分析未完成的细节定义要在面向对象设计完成。问题域设计主要包括以下几个方面：

①复用类设计

②提高性能

③增加一般类

④按OOPL调整继承与多态

⑤关联实现方式设计

⑥调整与完善属性

⑦构造及优化算法

⑧对象间的可访问性设计

⑨定义对象实例设计

⑩将类图转换为分析类图

（2）OOD问题域设计任务：以OOA类图为依据，进行如下设计将OOA类图逐渐转换为OOD类图。

#### 3.2.1.1增加根类和一般类

在OOD中，考虑一些类具有共同的实现策略，定义一般类集中地给出多个类的实现都要使用的属性与操作。

具体原则如下：

（1）增加根类：将所有具有相似协议的类组织在一起提供全系统通用的协议（例如：许多类都具有创建、删除、复制等操作，将其置于一般类，特殊类从中继承）。

（2）增加其他一般类：提供局部通用的协议（例如：将提供存储及检索功能的类置于一般类，特殊类从中继承）。

本系统中对员工的档案中的信息的查询、修改、添加可以设计出一个一般类。

#### 3.2.1.2复用类设计

如果在面向对象分析识别和定义的类是本次开发中新定义的，而且没有可复用的资源，则需要进一步设计和编程。如果已存在一些可复用的类，而且这些类既有分析、设计时的定义，又有源程序，那么，复用这些类即可提高开发效率与质量。

可复用的类可能只是与面向对象分析模型中的类相似，而不是完全相同对二者进行修改。

（1）如果完全相同，就把可复用的类直接加到问题域，并用{复用}标记所复用的类。

（2）如果大于，就把可复用的类直接加到问题域，并用{复用}标记所复用的类，所需要的类再继承它。

（3）如果大于，就把可复用的类直接加到问题域，删除可复用类中的多余信息，并用{复用}标记所复用的类。

（4）如果相似，按如下方法处理：

•把要复用的类加到问题域，标以 “复用”。

•划掉（或标出）不用的属性与服务。

•建立从复用类到问题域原有的类之间的泛化关系。

•由于问题域的类继承了“复用”类的特征，所以有些属性和服务不需要了应该把它们划掉。

•考虑修改问题域原有类的结构和连接，必要时移到“复用”类。

#### 3.2.1.3提高系统性能

（1）调整对象分布：把需要频繁交换信息的对象，尽量地放在一台处理机上；

（2）合并通信频繁的类；

（3）增加属性：保存中间结果减少重复计算；

（4）细化对象分类：针对具体情况，在原来一般类之下定义一些特殊类，并在每个特殊类中定义各自对象的操作；

（5）缩短对象存取时间：设立缓冲区；

（6）降低算法的计算复杂性：节省时、空的目标往往冲突，通常以空间换时间。[[12]](#footnote-12)

#### 3.2.1.4按OOPL特征调整多继承与多态

由于在面向对象分析强调如实地反映问题域，面向对象设计考虑实现问题，所选用的面向对象编程语言不支持多继承或多态性，需要根据编程语言调整面向对象分析模型，否则到了系统实现阶段由程序员随意处理，将很难保证模型与程序一致。面向对象分析阶段可能将某些与面向对象编程语言有关的对象细节推迟到面向对象设计阶段描述，在确定面向对象编程语言后，必须明确设计决策，因为所用语言不支持多继承，甚至不支持继承。

（1）对继承的调整。对继承的调整的方法有简单转化、重新定义对象类（压平方式）和保持分类剥离多继承信息。

（2）调整多态性。在继承结构中，具有相同名字的属性和操作，在不同的类中可以具有不同的类型和行为。这种在继承结构中对同一命令具有不同的含义的机制，就是继承的多态。如果编程的语言不支持多态，就需要把多态有关的属性和操作的名字分别赋予不同的含义，也即明确把他们是为不同的东西，不但如此，往往还要按实际要求，重新考虑对象的分类，并对属性和操作进行调整。

#### 3.2.1.5决定关联的实现方式

在 OOA 建立的模型中，可能含有关联类和 N 元关联，目前的 OOPL 并不支持这样的关系，为此需要将其转化为二元关联，另外从实现的角度考虑，对于多对多的二元关联，需要将其转化为一对多的二元关联；

1、转化复杂关联

2、将关联类和 N 元关联转化为二元关联：见 OOA 基本模型--对象之间的关联关系；2.关联的实现方式：

3、将多对多关联转化为一对多关联

对于 C++ 这样的语言，多对多关联对实现而言可能的麻烦是无论哪一端类的对象用指针指向另一端类的对象，类中设立的指针数目都是不定的，只有使用指针链表；

若不想使用指针链表，且不需要关联两端类的对象都要用指针指向对方；

类之间的关系种类： Generalization(泛化)、Dependency(依赖关系)、Association(关联关系)、Aggregation(聚合关系)、Composition(合成关系)。

4、对关联进行调整后，要考虑关联的实现方式。

**聚合：**

指的是整体与部分的关系。通常在定义一个整体类后，再去分析这个整体类的组成结构。从而早出一些组成类，该整体类和组成类之间形成了聚合关系。需求描述中“包含”、“组成”、“分为…部分”等词通常意味着聚合关系。

若用部分类定义整体类，可用部分类直接作为整体类中属性的数据类型；

把部分类用作指针变量（C++）或引用变量（Java）的基类型，再用这样的指针变量或引用变量定义整体类的属性，在整体类的相应操作中保证整体对象管理部分对象；

**关联：**

关联是指两个类之间存在某种特定的对应关系，例如客户和订单，一个订单只能属于某个客户，一个客户可能会有多张订单。根据方向，分为单向和双向。根据对应的数量分为一对一、一对多、多对多等。

通常，通过在对象中设立指针或对象标识以指向或记录另一端的对象的方法，来实现关联。如果是单向关联，就在源端的类中设立属性，用来标记另一端的类将来创建的对象。如果是双向关联，就在两端类中各设立属性，用来标记对方将来创建的对象。

若关联中对方类的多重性是 1，则在本方设立一个指向对方对象的指针，或设立一个记录对方对象引用的属性。

若对方类的多重性大于 1，则在本方设立一个指向对方对象的指针集合或引用集合。

若关联的某端有角色名，最好把其作为另一端类的属性名，以访问与角色名相邻的类。

#### [3.2.1.6调整与完善属性](#_Toc441672575)

按照语法：

[可见性] 属性名[‘：’类型][‘=’ 初始值]

对属性的定义进行完善。每一个属性或者包含单个值，或者包含作为一个整体的密切相关的一组值。

若要给出对属性的性质的约束，如“工龄<60”或“0≤英语成绩≤100”等，也要看语言是否对其直接支持，否则要在算法上考虑如何实现。

为了维护数据的完整性，必须要考虑需要一起更新的多个相关联的数据值。特别是，当基本的数据发生变化时，必须更新导出的属性。通过下列方法可以做到这一点：

1、显式的代码

因为每一个导出属性是根据一个或多个基本对象属性定义的，更新导出属性的一种方法是，在更新基本对象属性的操作中插入更新导出属性的代码。这种附加的代码将明确地更新依赖基本对象属性的导出属性，使得基本属性与导出属性的值同步。

2、批处理性的重计算

当基本数据以批处理的方式改变时，可能在所有的基本数值改变之后，再重新计算所有的导出属性的值。

3、触发器

凡是依赖基本属性的属性，都必须将它自己向基本属性注册。当基本属性的值被更新时，由专门设置的触发器更新导出属性的值。

我们小组对类的属性以及操作的可见性，类之间的关联关系做出如下详细说明：

（1）人事管理者类

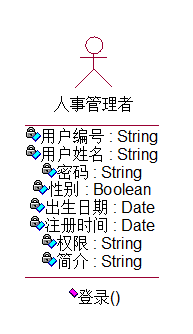


图3.2.1.6.1人事管理者类

**属性：**

private 用户编号 String：账号为用户在系统中的唯一ID，另由于该编号不可随意修改，因此设为private类型。

private 用户姓名 String：该姓名为用户的真实姓名，另由于本系统需要实行实名制，真实姓名不可随意修改，因此设为private类型。

private 密码 String：用户在登录系统时需要输入与账号匹配的密码方可成功登录。另由于考虑到安全性，密码仅本人可见，因此设为private类型。

private性别 Boolean：用户的性别。由于性别不可随意修改，因此设为private类型。

public出生日期Date：用户的出生日期。

private注册时间Date：该用户的创建时间。考虑到该创建时间不能随意修改，因此设为private类型。

private权限String：该用户的身份标识。考虑到权限不能随意修改，因此设为private类型。

Public简介String：该用户的简单描述。

**主要方法：**

public Integer 登录（）：用户需要输入账号以及相应的密码登录系统。若登录成功，本方法返回1；若用户名或密码错误，本方法返回2；若服务器或网络原因导致错误，本方法返回3。

（2）应聘人员信息类



图3.2.1.6.2应聘人员信息类

**属性：**

private编号 String：编号由系统生成，是应聘人员在系统中的唯一ID。另由于该编号不可随意修改，因此设为private类型。

private姓名 String：应聘人员的真实姓名，考虑到可能存在重名，因此编号才是唯一识别应聘人员的标志。另由于人员姓名不可随意修改，因此设为private类型。

private 职位 String：这里针对已经进行入库操作的应聘人员——也就是已录取的员工，企业会为其安排职位。另由于职位信息不可随意修改，因此设为private类型。

public 年龄Integer：应聘人员的年龄。

private 招聘时间 Date：应聘人员的应聘时间。该时间不可随意修改，因此设置为private类型。

private 电话 String：应聘人员的联系方式。由于联系方式不可随意修改，因此设为private类型。

private 邮箱 String：应聘人员的邮箱。由于邮箱不可随意修改，因此设为private类型。

private性别 Boolean：应聘人员的性别。由于性别不可随意修改，因此设为private类型。

private 专业 String：应聘人员主修的专业信息。由于专业信息不可随意修改，因此设为private类型。

private 工作经历 String：应聘人员的工作经历。由于工作经历不可随意修改，因此设为private类型。

private 学历 String：应聘人员的学历信息。由于学历不可随意修改，因此设为private类型。

private 学校 String：应聘人员毕业的学校信息。由于该信息不可随意修改，因此设为private类型。

public 内容 String：应聘简历具体内容。

private 简历是否入库 Boolean：应聘简历的当前状态（入库的操作等同于录取）。其中，本属性只可取值0、1。0对应“未入库”，1对应“已入库”。另由于该状态不可随意修改，因此设为private类型。

private 体检结果Boolean：应聘人员的体检结果。其中，本属性只可取值0、1。0对应“不合格”，1对应“合格”。另由于该结果不可随意修改，因此设为private类型。

**主要方法：**

public Integer 人员信息录入（）：用户可用本方法录入应聘人员的基本信息。如果录入成功，本方法返回1；如果服务器或网络原因导致错误，本方法返回3。

public void 人员信息查看（）：用户可用本方法查看人员的基本信息，但不可修改。

public Integer人事异动管理（）：用户可用本方法查看或修改员工的人事异动信息，包括对员工的升职、离岗等操作。若操作成功，本方法返回1；若服务器或网络原因导致错误，本方法返回3。

（3）绩效考核类

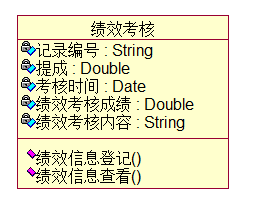


图3.2.1.6.5 绩效考核类

**属性：**

private 记录编号 String：记录编号由系统生成，为该条考核记录在系统中的唯一识别ID。另由于记录编号不可随意修改，因此设为private类型。

private 提出Double：由员工的绩效考核决定的提成金额数。另由于此金额不可随意修改，因此设为private类型。

private 考核时间 Date：考核时间由用户输入，为该条绩效考核记录的时间。另由于考核时间不可随意修改，因此设为private类型。

private 绩效考核成绩Double：考核成绩是员工绩效考核的结果。该成绩将决定提成的金额数。另由于考核结果不可随意修改，因此设为private类型。

private 绩效考核内容 String：员工绩效考核的具体内容。另由于该内容不可随意修改，因此设为private类型。

**主要方法：**

public Integer绩效信息登记（）：用户可用本方法录入员工的绩效考核信息。若录入成功，本方法返回1；若服务器或网络原因导致错误，本方法返回3。

public void绩效信息查看（）：用户可用本方法查看员工的绩效考核信息，但不可修改。

（4）薪金信息类

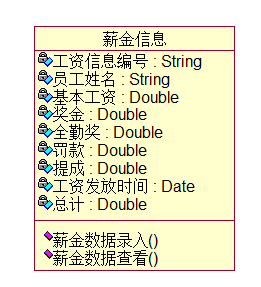


图3.2.1.6.6 薪金信息类

**属性：**

Private工资信息编号 String：工资信息编号在系统中不可重复，能够唯一标识一条工资信息记录。另由于该编号不可随意修改，因此设为private类型。

Private员工姓名 String：员工的真实名字。由于姓名不可随意修改，因此设为private类型。

private 基本工资 Double：基本工资由员工的岗位决定，不同岗位的基本工资不同。另由于基本工资不可随意修改，因此设为private类型。

private 奖金 Double：员工的奖励金额，能为员工工资加成。另由于金额数不可随意修改，因此设为private类型。

private 全勤奖 Double：员工的全勤奖金额数。另由于金额数不可随意修改，因此设为private类型。

private 罚款 Double：罚款为员工工资减少总额，其可能包括缺勤扣除金额、惩罚扣除金额等。另由于金额数不可随意修改，因此设为private类型。

private 提成Double：员工的绩效提成金额，能为员工工资增加总额。另由于金额数不可随意修改，因此设为private类型。

private 总计 Double：总计是员工的工资总额，工资总额=基本工资+工资加成（包括奖金、全勤奖、提成）-工资扣除（罚款），为最终发放给员工的工资。另由于金额数不可随意修改，因此设为private类型。

**主要方法：**

public Integer薪金数据录入（）：用户可用本方法录入各岗位员工的薪金信息。若录入成功，本方法返回1；若服务器或网络原因导致错误，本方法返回3。

public void 薪金数据查看（）：用户可用本方法查看员工的薪金信息，这里的薪金信息包括基本工资、奖金、全勤奖、罚款、提成，可查看但不可修改。

（5）培训基本信息类

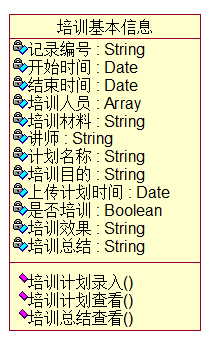


图3.2.1.6.7 培训基本信息类

**属性：**

private 记录编号 String：记录编号为该条培训信息记录所属的编号，由系统生成，在系统中不可重复。另由于记录编号不可随意修改，因此设为private类型。

private 开始时间 Date：开始时间由用户输入，为培训的开始时间。另由于开始时间不可随意修改，因此设为private类型。

private 结束时间 Date：结束时间由用户输入，为培训的结束时间。另由于结束时间不可随意修改，因此设为private类型。

private 培训人员 Array：培训人员为该次培训的参与员工列表，数组内元素为同一岗位的员工。另由于培训人员不可随意修改，因此设为private类型。

private 培训材料 String：培训材料由用户输入，为该具体培训教材、材料等等。另由于培训材料不可随意修改，因此设为private类型。

private 讲师 String：讲师由用户输入，为该次培训的具体讲师。另由于讲师不可随意修改，因此设为private类型。

private 计划名称 String：计划名称由用户输入，是该次培训计划总称。另由于计划名称不可随意修改，因此设为private类型。

private 培训目的 String：培训目的由用户输入，是该次培训计划的目的。另由于培训目的不可随意修改，因此设为private类型。

private 上传计划时间 Date：上传计划时间由用户输入，为培训计划的上传时间。另由于该时间不可随意修改，因此设为private类型。

private是否培训 Boolean：员工的培训状态。其中，本属性只可取值0、1。0对应“未培训”，1对应“已培训”。另由于该状态不可随意修改，因此设为private类型。

private 培训效果 String：受培训的员工的培训效果。由于培训效果不可随意修改，因此设为private类型。

private 培训总结 String：受培训的员工的培训总结。由于培训总结不可随意修改，因此设为private类型。

**主要方法：**

public Integer 培训计划录入（）：用户可用本方法在未培训的员工列表中选择一类员工（一般来说是岗位相同）进行培训，并设置培训开始于结束时间、培训计划名称等等。若操作成功，本方法返回1；若服务器或网络原因导致错误，本方法返回3。

public void培训计划查看（）：用户可用本方法查看已培训员工的培训计划信息，但不可修改。

public void培训总结查看（）：用户可用本方法查看已培训员工的培训总结信息，但不可修改。

（6）奖惩记录类

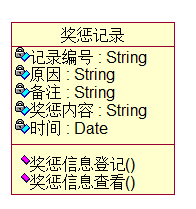


图3.2.1.6.8 奖惩记录类

**属性：**

private 记录编号 String：记录编号为该条奖惩记录的编号，由系统生成，在系统中不可重复。另由于记录编号不可随意修改，因此设为private类型。

private 原因 String：奖惩原因由用户输入，为该次奖惩发生的具体原因。另由于奖惩原因不可随意修改，因此设为private类型。

public 备注 String：备注由用户输入，为该次奖惩发生的一些细节方面以及企业规章制度方面的备注。

private 奖惩内容 String：奖惩内容由用户输入，为该次奖惩发生的具体记录，包括员工信息与奖金/罚款金额的记录。另由于奖惩内容不可随意修改，因此设为private类型。

private 时间 Date：时间由用户输入，为奖惩发生的时间。另由于时间不可随意修改，因此设为private类型。

**主要方法：**

public Integer 奖惩信息登记（）：用户可用本方法登记员工的奖惩信息。若录入成功，本方法返回1；若服务器或网络原因导致错误，本方法返回3。

public void 奖惩信息查看（）：用户可用本方法查看员工的奖惩信息，但不可修改。

#### [3.2.1.7构造及优化算法](#_Toc441672576)

对于需要设计的操作，要从如下几方面进行详细地定义：

1、按照定义操作的格式 。

2、从问题域的角度，考虑对象提供操作的方式。

3、操作过程前端设计美工结束后，要考虑编程语言是否能够支持的问题。

4、进一步地分析类的对象相关的所有交互图，找出其所有与之相关的消息根据其状态图获内部逻辑关系，设计算法的整体架构。

可用自然语言或半结构化语言（伪代码），也可使用程序流程图或活动图等图形语言。

在算法中要考虑对例外和特殊情况的处理，如考虑对输入错误、来自中间件或其它软硬件的错误的消息以及其它例外情况的处理。

在系统较为复杂或需要处理大批量的数据的情况下，若系统在性能上有要求，就要对系统的体系结构和算法进行优化。

针对异常的建模：在算法中要考虑可能出现的异常及对异常的处理，若异常情况较为复杂，可针对其进行建模型，通常用信号对异常建模。

#### [3.2.1.8完善对象细节](#_Toc441672577)

从类 A 的对象到类 B 的对象有 4 种访问性：

1、属性可见性：B 是 A 的一个属性（关联、聚合）；

class A

{ … ；B b；…}

2、参数可见性： B 的对象是 A 的一个方法的参数（依赖）；

【例】 A.amethod（B b） //间接地找到一个对象，并赋给 b；

3、局部声明可见性： B 的对象是在 A 的一个方法中声明的一个局部变量（依赖）；

class A::amethod

{ …； B b；…}

4、全局可见性： B 的对象在某种程度上全局可见（一般为依赖）；

声明 B 的全局实例变量；

【说明】对于后三种情况，从类 A 到类 B 间存在着依赖关系，在程序运行期间 A 的对象与 B 的对象存在着临时性的连接（临时链），而第一种情况中的链是由从类 A 到类 B 间的关联实例化而来的。

#### [3.2.1.9定义对象实例设计](#_Toc441672578)

一个类的对象实例可以分布到不同的处理机上，说明在它之上创建的每一个（或组）内存对象，也说明在它之上保存的外存对象；

1、根据不同的实现条件和实现策略，定义对象的方式：

（1）用相应的类定义内存中的全局性对象，包括静态声明和动态创建两种方式。可以一次针对一个对象定义一个变量，也可以成批地定义对象

（2）当系统需要通过从外存读取数据来创建一个对象时，先创建该对象，再从外存中读取这个对象数据，把数据赋值给对象的相应属性；

（3）按照一定的策略，内存中的永久对象要保存到外存中，请参看数据管理部分；

2、类的对象实例说明

{

处理机：<结点名>{，<结点名>}

内存对象：{<名称>[（n元数组）][<文字描述>]}

外存对象：{<名称>[<文字描述>]}

}

#### [3.2.1.10](#_Toc441672579)其他

（1）进行输入数据验证类；

（2）对来自中间件或其它软硬件的错误进行处理的类；

（3）对其它例外情况进行处理的类；

……

### 3.2.2将OOA类图转换为OOD类图

#### 3.2.2.1按类版型将OOA类转换为OOD类

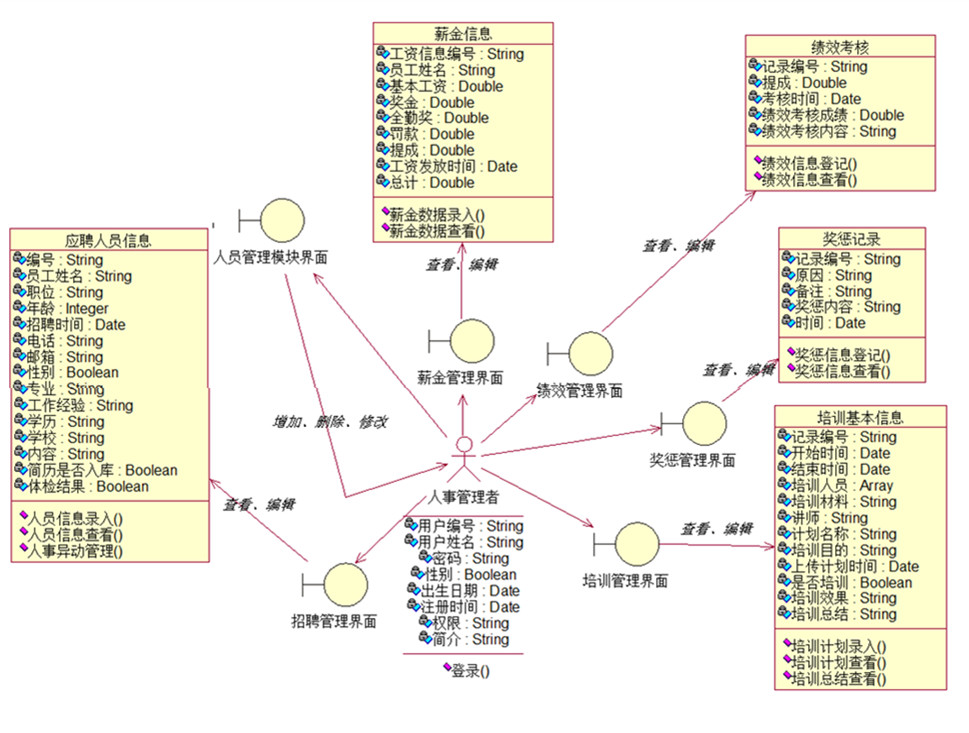


图3.2.2 OOD类图

#### **3.2.2.2编制数据字典**

表3.2.2.2.1应聘人员信息信息表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 应聘人员信息表（job） | | | | | |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 是否为主键/外键 |
| id | Id | 11 | 员工编号 | 否 | 主键 |
| name | varchar | 50 | 员工姓名 | 是 |  |
| sex | varchar | 2 | 性别 | 否 |  |
| age | int | 11 | 年龄 | 是 |  |
| job | varchar | 50 | 职位 | 是 |  |
| specialty | varchar | 50 | 专业 | 是 |  |
| experience | varchar | 50 | 工作经验 | 是 |  |
| studyeffort | varchar | 50 | 学历 | 是 |  |
| school | varchar | 50 | 学校 | 是 |  |
| tel | varchar | 50 | 电话 | 是 |  |
| email | varchar | 50 | 邮箱 | 是 |  |
| createtime | datetime | - | 应聘时间 | 是 |  |
| content | text | - | 内容 | 是 |  |
| isstock | bit | 1 | 简历是否入库 | 是 |  |
| health | varchar | 50 | 体检结果 | 是 |  |

表3.2.2.2.2培训信息表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 培训信息表（educate） | | | | | |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| id | int | 11 | 培训计划编号 | 否 | 主键 |
| name | varchar | 100 | 培训计划名称 | 否 |  |
| purpose | varchar | 500 | 培训目的 | 否 |  |
| begintime | datetime | - | 培训开始时间 | 是 |  |
| endtime | datetime | - | 培训结束时间 | 是 |  |
| datum | text | - | 培训材料 | 是 |  |
| teacher | varchar | 50 | 讲师 | 是 |  |
| student | varchar | 50 | 培训人员 | 是 |  |
| createtime | datetime | - | 上传计划时间 | 是 |  |
| educate | bit | 1 | 是否培训 | 是 |  |
| effect | varchar | 500 | 培训效果 | 是 |  |
| summarize | text | - | 培训总结 | 是 |  |

表3.2.2.2.3绩效信息表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 绩效信息表（performance） | | | | | |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| id | int | 11 | 绩效信息编号 | 否 | -1 |
| content | varchar | 50 | 绩效考核内容 | 是 | 0 |
| score | float | 8 | 绩效考核成绩 | 否 | 0 |
| addmoney | float | 8 | 提成 | 是 | 0 |
| createtime | datetime | - | 绩效考核时间 | 是 | 0 |

表3.2.2.2.4奖惩信息表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 奖惩信息表（institution） | | | | | |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| id | int | 11 | 奖惩编号 | 否 | 主键 |
| name | varchar | 50 | 奖惩内容 | 否 |  |
| reason | varchar | 50 | 原因 | 是 |  |
| explains | text | - | 备注 | 是 |  |
| createtime | datetime | - | 奖惩时间 | 是 |  |

表3.2.2.2.5管理人员信息表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 管理人员信息表（users） | | | | | |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| id | int | 11 | 用户编号 | 否 | 主键 |
| username | varchar | 50 | 用户姓名 | 否 |  |
| password | varchar | 50 | 密码 | 否 |  |
| sex | bit | 1 | 性别 | 是 |  |
| Birthday | datetime | - | 出生日期 | 否 |  |
| createtime | datetime | - | 注册时间 | 否 |  |
| Isadmin | bit | 1 | 权限 | 是 |  |
| Content | text | - | 简介 | 是 |  |

表3.2.2.2.6薪金信息表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 薪金信息表（stipend） | | | | | |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| Id | int | 11 | 工资信息编号 | 否 | 主键 |
| Name | varchar | 50 | 员工姓名 | 否 |  |
| Basic | float | 10 | 基本工资 | 是 |  |
| Reward | float | 8 | 奖金 | 是 |  |
| Duty | float | 8 | 全勤奖 | 是 |  |
| punishment | float | 8 | 罚款 | 是 |  |
| Other | float | 8 | 提成 | 是 |  |
| Granttime | datetime | - | 工资发放时间 | 是 |  |
| Totalize | float | 10 | 总计 | 是 |  |

#### 3.2.2.3按OO设计原则调整OOD类图

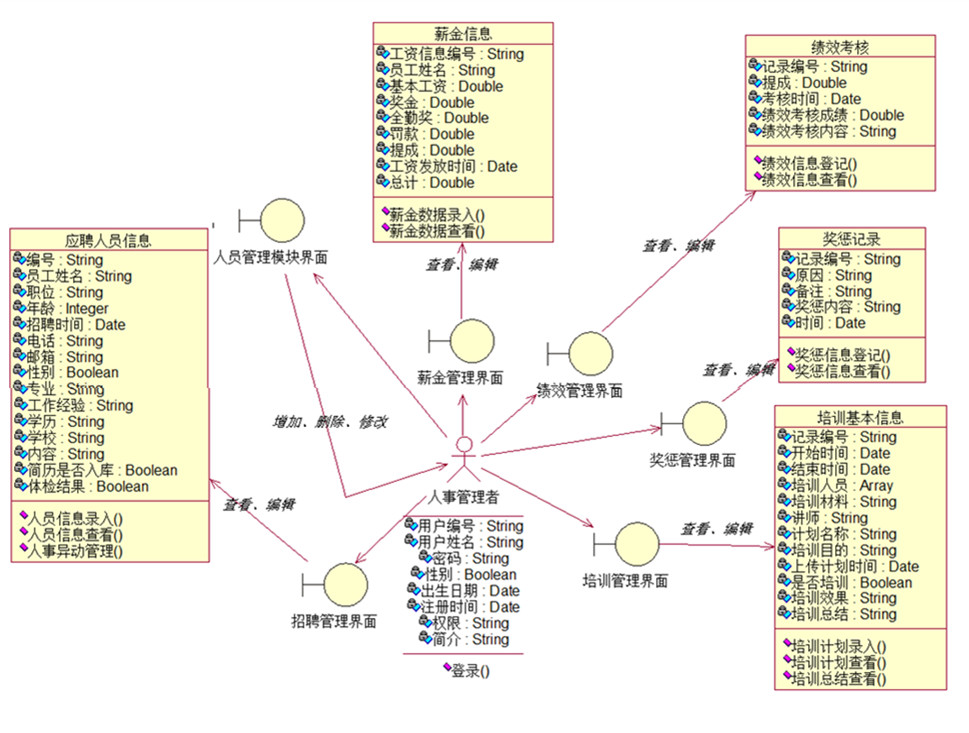


图3.2.2.3 OOD类图

## 3.3人机界面设计

### 3.3.1界面设计的意义

信息系统已成为用户直接操作和应用的主体，它应以美观实用、操作便捷为用户所青睐。用户界面设计的规范性显得尤为重要。

界面的总体色彩应该接近和类似系统界面的总体色调，色彩的强烈变化会影响用户的使用情绪，我们知道系统界面是根据用户审美习惯定制的，它应该有他们的审美群体。合理地结合系统界面进行设计还包括图标、[按钮](http://www.680.com/anniu/)的风格及在不同操作状态下的视觉效果。[[13]](#footnote-13)

于中国人来说，轿车并不仅仅是交通工具，在它之上往往负载了很多别的意义。设计除了在产品与用户之间充当实际存在的**联系纽带**之外，还应该是意义的纽带。

设计产品是建立联系的过程，与人们的需求建立联系，与人们的渴望建立联系，与他们的文化和所处的世界建立联系，有的时候，帮助人们在他们的情感和自我之间建立联系。这常常被视为品牌的领域，但它也是设计能够发挥效能的地方——好的产品、把产品交给客户的方式、客户每一次和企业接触的体验——这些都是设计的领地。

设计是关于界面、功能选择和意义，而不仅仅是让产品外形漂亮。把人的思维和行为转换成机器可以接受的方式，把机器的行为方式转换成人可以接受的方式，这个转换就是用户界面。检验一个界面的标准是最终用户的感受。所以界面设计要和用户体验紧密结合，是一个不断为最终用户设计满意视觉效果的过程。

### 3.3.2界面设计的内容

3.3.2.1 登录界面

（1）叙述性综述

表单：系统登录界面

用户：人事管理员

任务：用户打开该界面后可进行之后的登录操作

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

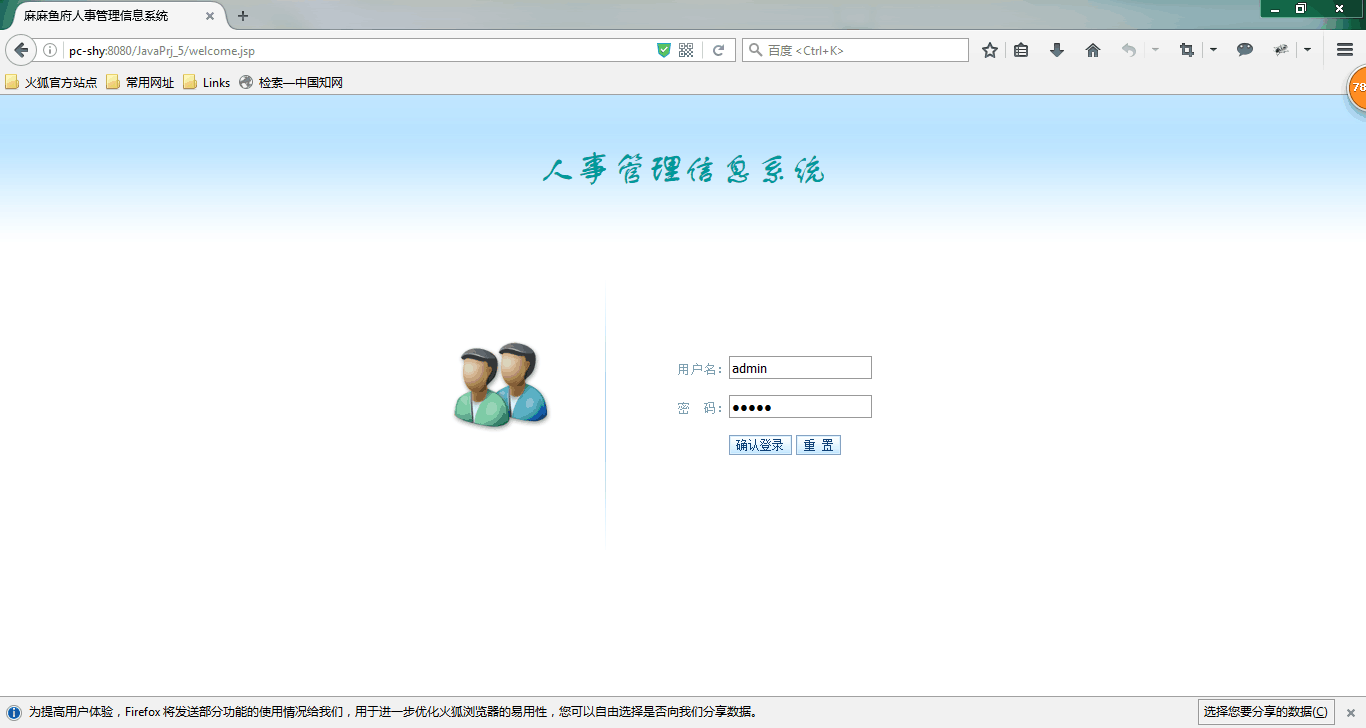


图3.3.2.1 系统登录界面

3.3.2.2 系统首页

(1)叙述性综述

表单：系统首页界面

用户：人事管理员

任务：人事管理员登录后可进行人员管理、招聘管理、人事异动、培训管理、奖惩管理、绩效管理以及薪金管理几个功能的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

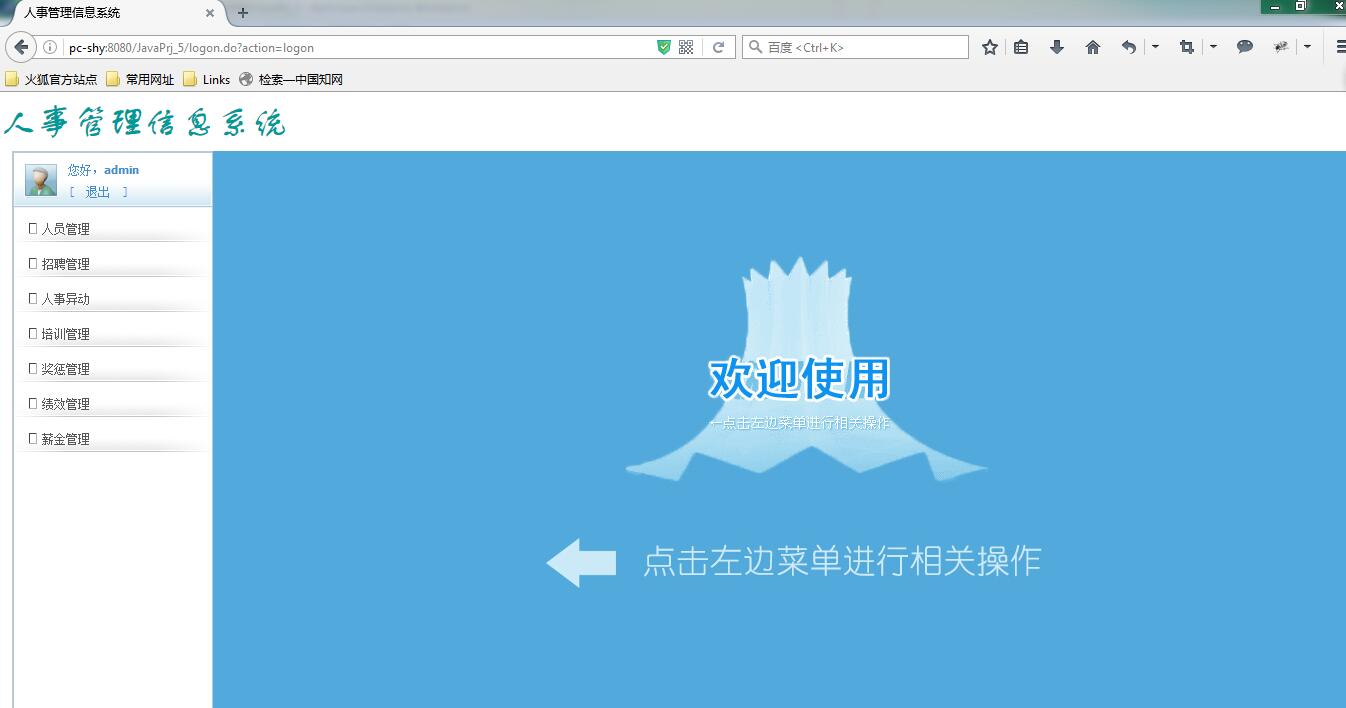


图3.3.2.2 项目首页

3.3.2.3 人员管理界面

(1)叙述性综述

表单：系统人员管理界面

用户：人事管理员

任务：人事管理员选择人员管理功能后可进行人员信息录入、人员信息查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

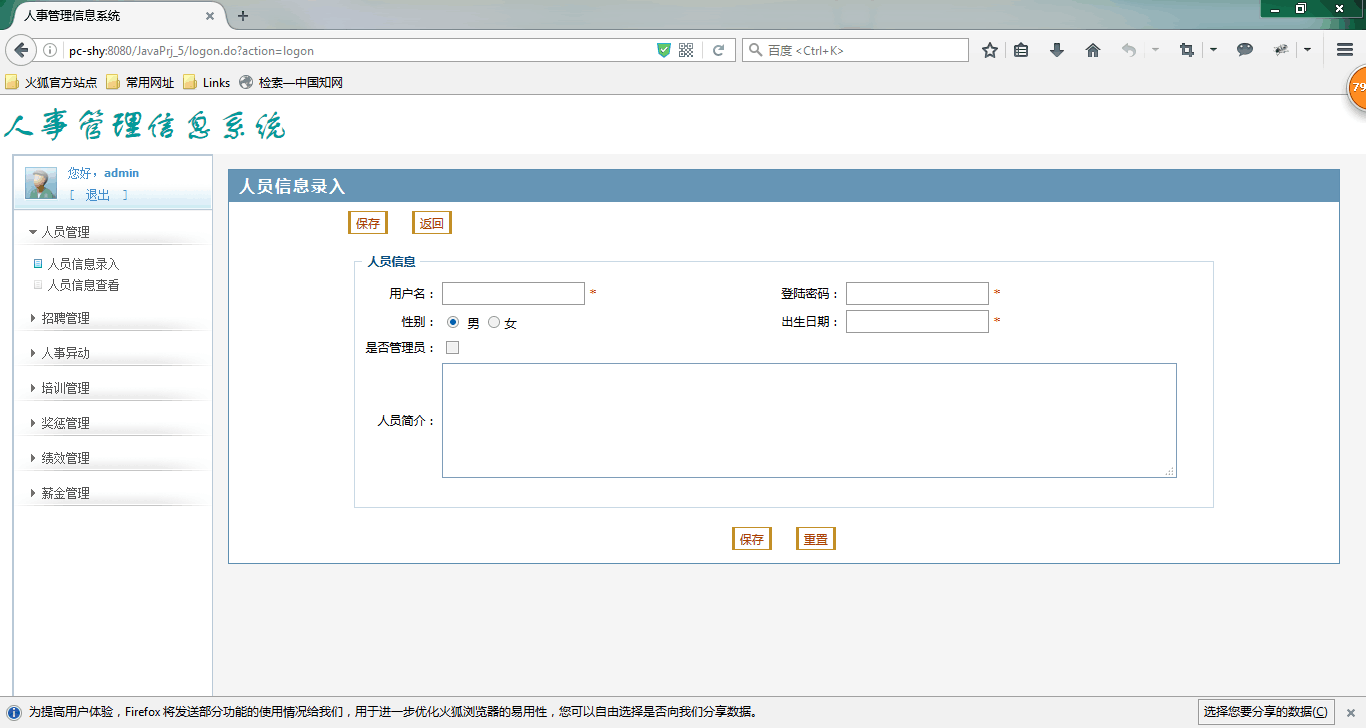


图3.3.2.3.1 人员信息录入

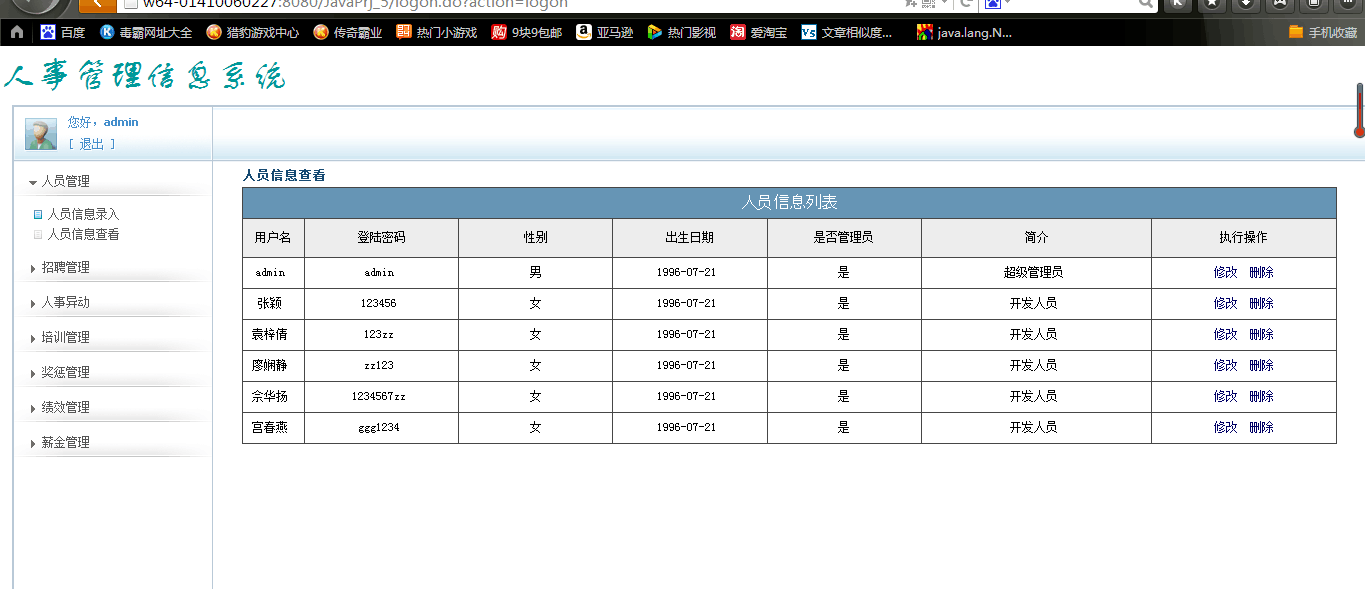


图3.3.2.3.2 人员信息查看

3.3.2.4 招聘管理界面

(1)叙述性综述

表单：系统招聘管理界面

用户：人事管理员

任务：人事管理员选择招聘管理功能后可进行应聘信息录入、应聘信息查看、人才库浏览的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计



图3.3.2.4.1 应聘信息查看

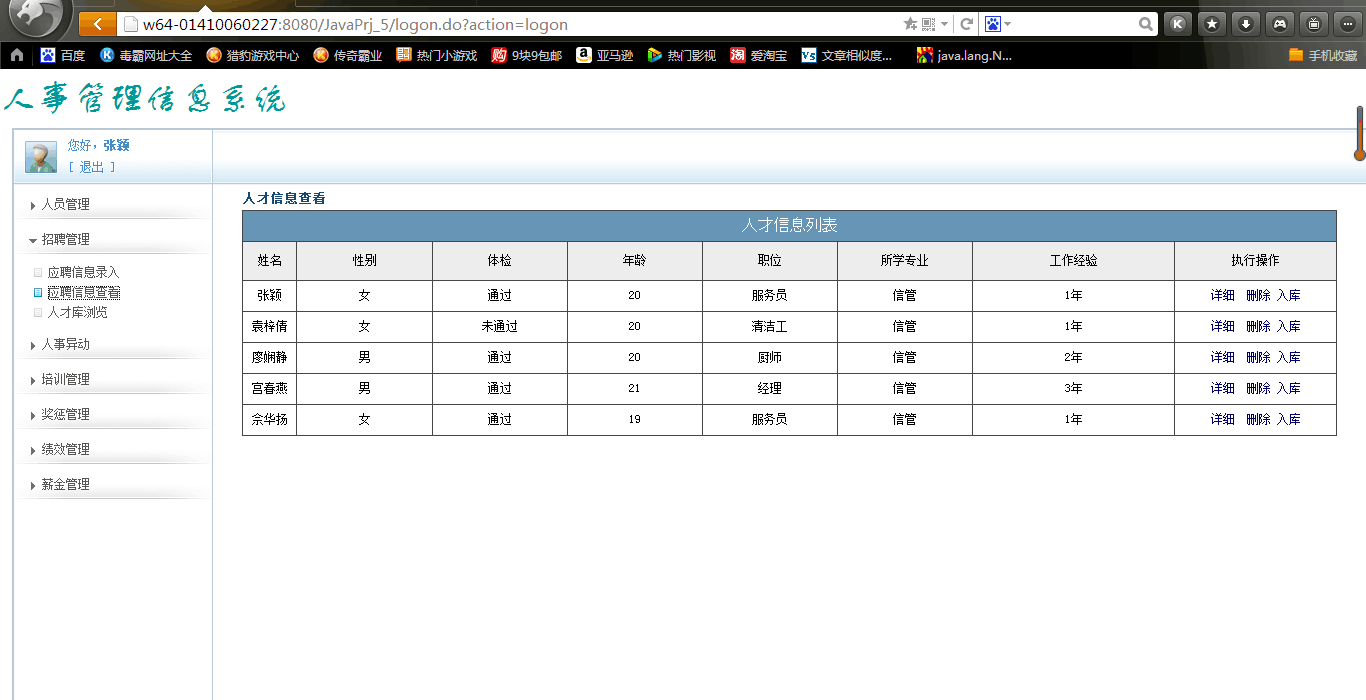


图3.3.2.4.2 应聘信息查看

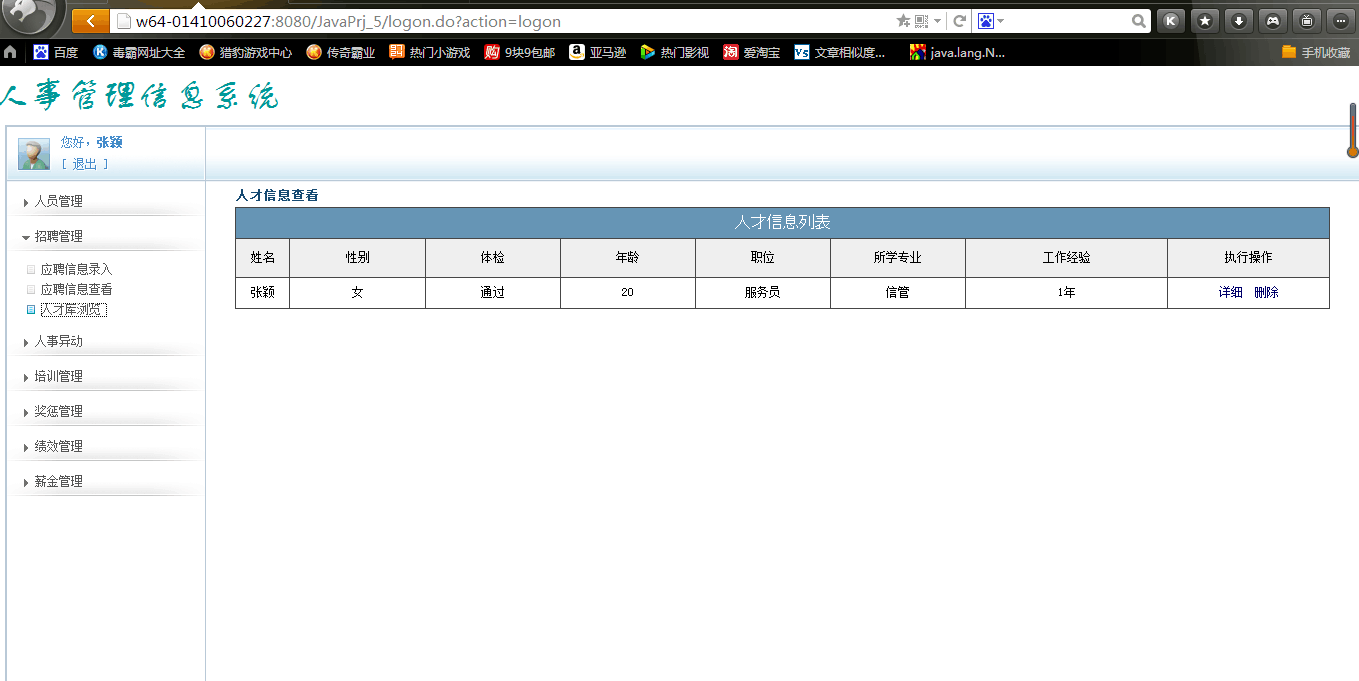


图3.3.2.4.3 人才库浏览

3.3.2.5 人事异动界面

(1)叙述性综述

表单：系统人事异动界面

用户：人事管理员

任务：人事管理员选择人事异动功能后可进行人事信息查看的选择，在人事信息查看的界面还可以选择查看详细信息、修改的操作

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计



图3.3.2.5.1 人事信息查看



图3.3.2.5.2 详细信息查看



图3.3.2.5.3 人事信息修改

3.3.2.6 培训管理界面

（1）叙述性综述

表单：系统培训管理界面

用户：人事管理员

任务：人事管理员选择培训管理功能后可进行培训计划录入、培训计划查看、培训总结查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

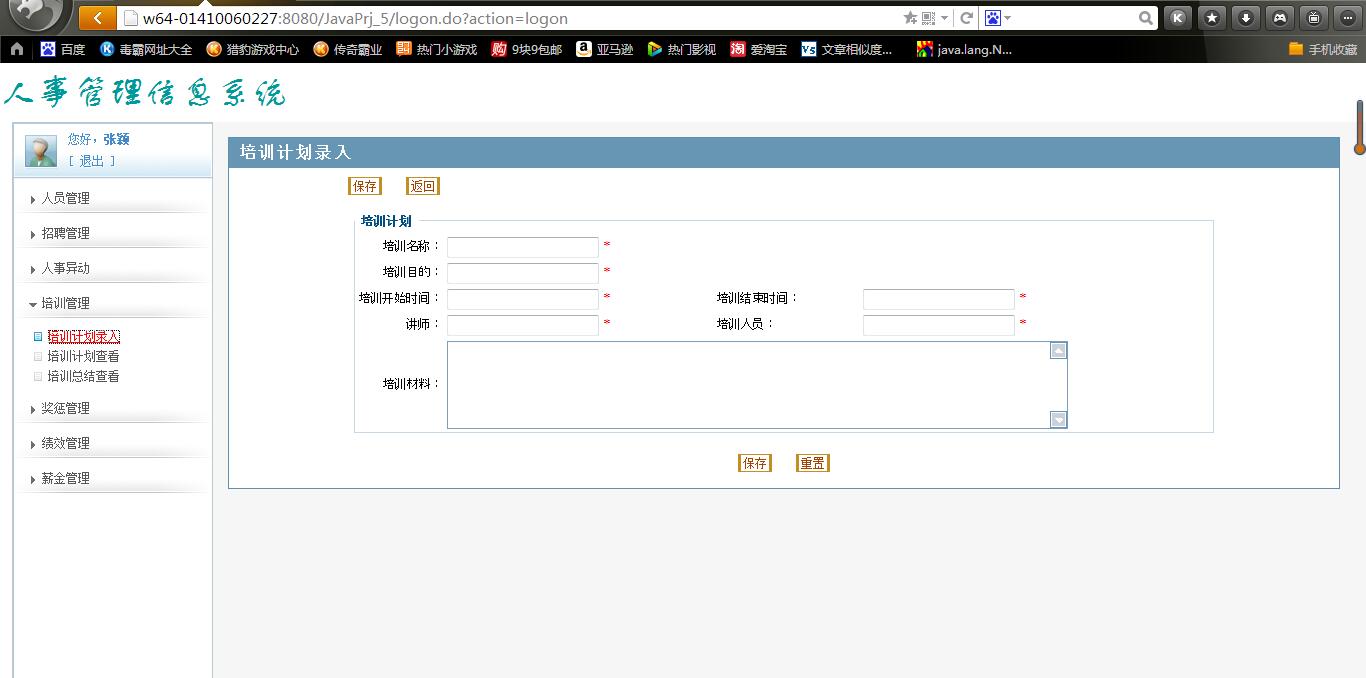


图3.3.2.6.1 培训计划录入

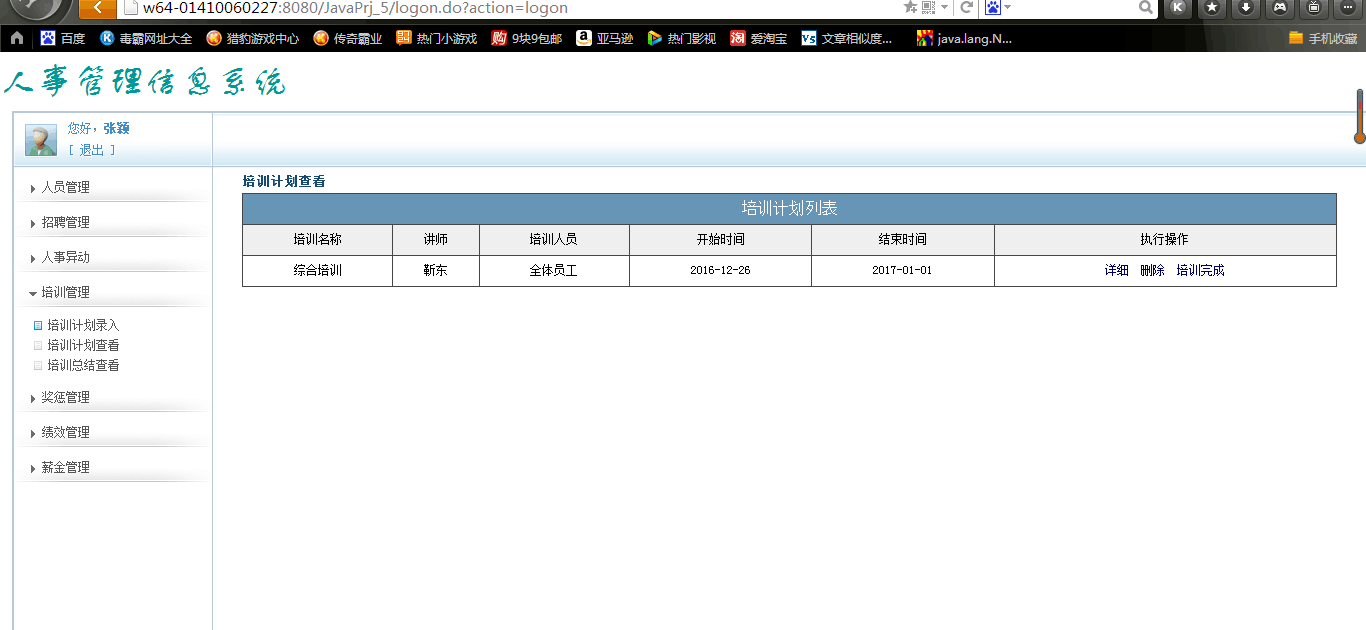


图3.3.2.6.2 培训计划查看

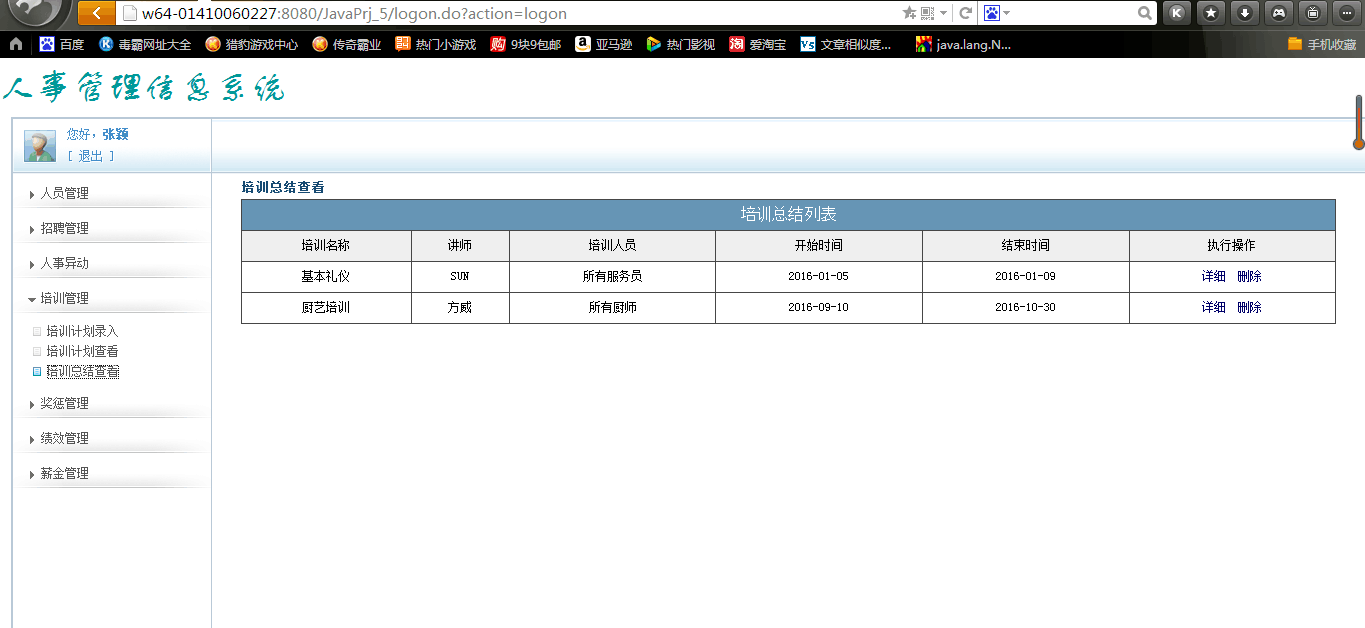


图3.3.2.6.3 培训总结查看

3.3.2.7 奖惩管理界面

(1)叙述性综述

表单：系统奖惩管理界面

用户：人事管理员

任务：人事管理员选择奖惩管理功能后可进行奖惩信息登记、奖惩信息查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计



图3.3.2.7.1 奖惩信息登记

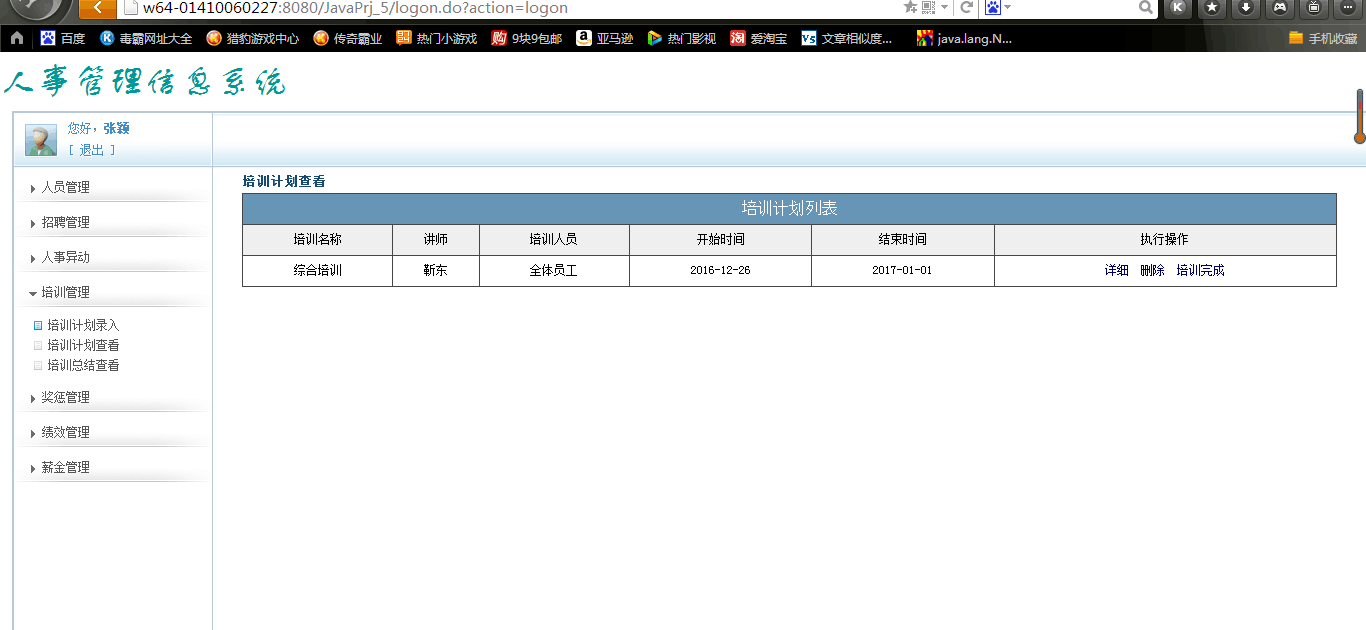


图3.3.2.7.2 奖惩信息查看

3.3.2.8 绩效管理界面

(1)叙述性综述

表单：系统绩效管理界面

用户：人事管理员

任务：人事管理员选择绩效管理功能后可进行绩效信息登记、绩效信息查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

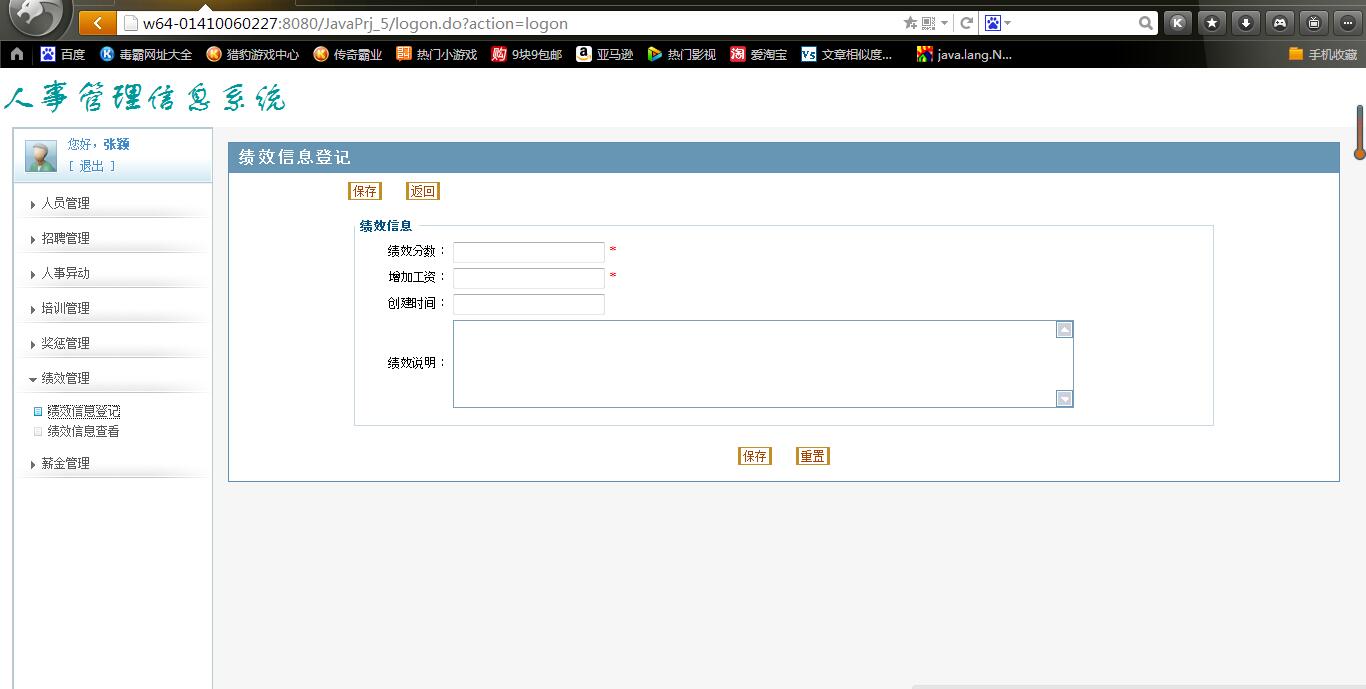


图3.3.2.8.1 绩效信息登记

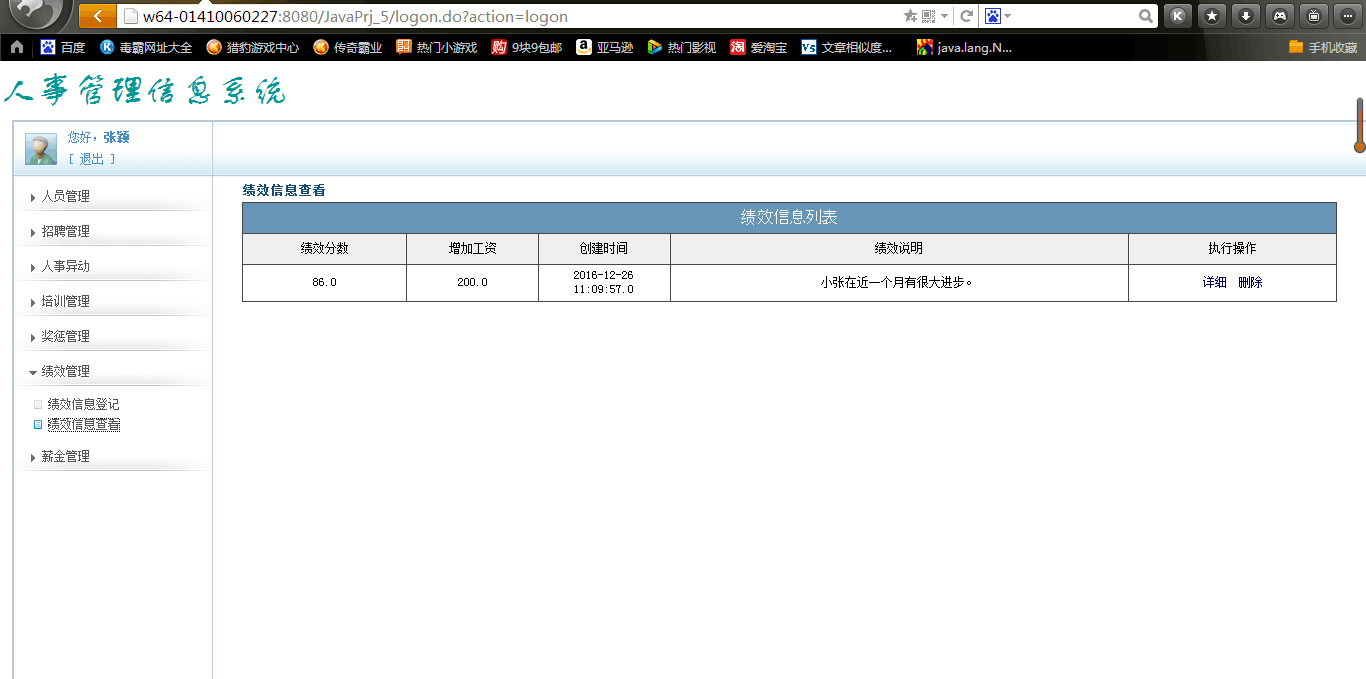


图3.3.2.8.2 绩效信息查看

3.3.2.9 薪金管理界面

(1)叙述性综述

表单：系统薪金管理界面

用户：人事管理员

任务：人事管理员选择薪金管理功能后可进行薪金数据录入、薪金数据查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计



图3.3.2.9.1 薪金数据录入

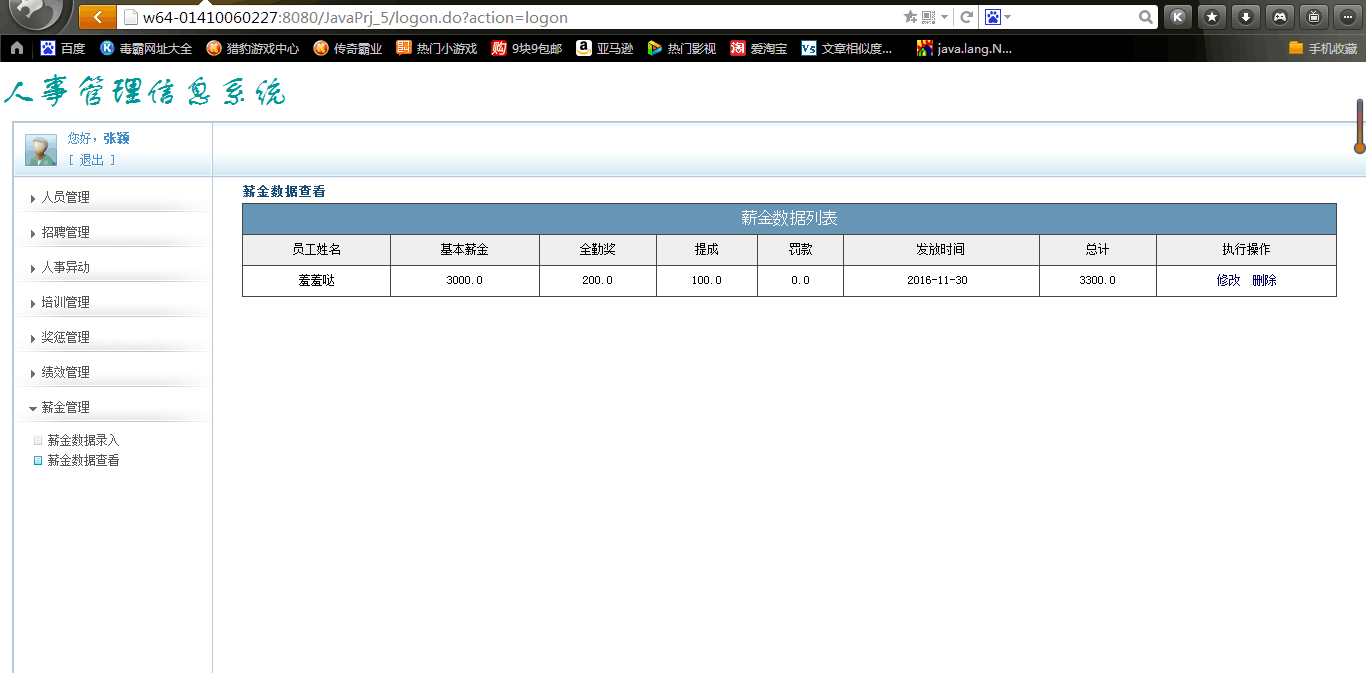


图3.3.2.9.2 绩效信息查看

### 3.3.3界面设计的一般原则

用户界面设计是屏幕产品的重要组成部分。界面设计是一个复杂的有不同学科参与的工程，认知心理学、设计学、语言学等在此都扮演着重要的角色。用户界面是用户与程序沟通的唯一途径，要能为用户提供方便有效的服务。

用户界面设计的三大原则是：置界面于用户的控制之下、减少用户的记忆负担、保持界面的一致性。详细说来有以下几点。

**1、简易性**

界面的简洁是要让用户便于使用、便于了解、并能减少用户发生错误选择的可能性。

**2、用户语言**

界面中要使用能反应用户本身的语言，而不是游戏设计者的语言。

**3、记忆负担最小**

人脑不是电脑，在设计界面时必须要考虑人类大脑处理信息的限度。人类的短期记忆极不稳定、有限，24小时内存在25％的遗忘率。所以对用户来说，浏览信息要比记忆更容易。

**4、一致性**

是每一个优秀界面都具备的特点。界面的结构必须清晰且一致，风格必须与游戏内容相一致

**5、清楚**

在视觉效果上便于理解和使用。

**6、用户的熟悉程度**

用户可通过已掌握的知识来使用界面，但不应超出一般常识。

**7、从用户的观点考虑**

想用户所想，做用户所做。用户总是按照他们自己的方法理解和使用。

通过比较两个不同世界(真实与虚拟)的事物，完成更好的设计。如：书籍对比竹。

**8、排列**

一个有序的界面能让用户轻松的使用。

**9、安全性**

用户能自由的作出选择，且所有选择都是可逆的。在用户作出危险的选择时有信息介入系统的提示。

**10、灵活性**

简单来说就是要让用户方便的使用，但不同于上述。即互动多重性，不局限于单一的工具(包括鼠标、键盘或手柄)。

**11、人性化**

高效率和用户满意度是人性化的体现。应具备专家级和初级玩家系统，即用户可依据自己的习惯定制界面，并能保存设置。[[14]](#footnote-14)

**3.4数据库设计**

在UML建模分析和系统功能分析后，就需要在此前的基础上建立相关的数据库，设计出能够满足用户需求的各种实体以及它们之间的关系，这些实体包含各种具体信息，以及通过相互作用形成的数据的流动。数据库设计包括需求分析、概念设计、逻辑设计、物理设计和实施维护。

### 3.4.1数据库设计概述

数据库设计的目标就是为用户的应用系统提供一个高效数据库系统。为了保证数据库设计更合理，数据库的设计工作通常分阶段进行，在不同的阶段采用不同的方法完成不同的设计内容。

数据库设计分为以下6个阶段：

（1）需求分析阶段。收集和分析用户对系统的信息需求和处理需求，得到设计系统所必需的需求信息，建立系统的需求说明文档。

（2）概念结构设计阶段。通过对用户的需求进行综合、归纳与抽象，形成一个独立于具体DBMS的概念模型。

（3）逻辑结构设计阶段。在概念模型的基础上导出一种DBMS支持的逻辑数据库模型，该模型应满足数据库存取、一致性及运行等各方面的用户需求。

（4）物理结构设计阶段。为逻辑数据模型选取一个最适合应用环境的物理结构。

（5）数据库实施阶段。根据逻辑设计和物理设计的结果建立数据库，编写与调试应用程序，将数据录入到数据库中，同时进行数据库系统的试运行。

（6）数据库运行与维护阶段。数据库应用系统试运行后，即可以投入正式运行。在此过程中必须不断地对其进行评价、调整与修改。

数据库采用本地数据库。数据库中有数张表：应聘人员信息表、培训信息表、绩效信息表、奖惩信息表、薪金信息表、管理人员信息表。每个表均只有一张表，存储其对应的所有信息。

**3.4.1.1需求分析**

经过可行性分析和初步的需求调查，确定了本系统的主要功能和功能边界。该系统应能完成下面的功能：

（1）人事管理者对人事信息的录入、更新与管理；

（2）开发管理员开发人事管理信息系统时结合实际调研为企业提供更多的功能；

（3）维护管理员是为人事管理信息系统提供后期的维护等。

**3.4.1.2概念结构设计**

数据库的概念设计是指针对现实世界，通过对其中信息实体的收集、分类、聚集和概括，建立数据库概念结构的过程，应该反映系统组织的业务模式、信息结构、信息间的相互制约关系，以及对信息存储、查询和加工的处理要求等。

人事管理信息系统中存在的实体是员工的人事信息。根据这个，设计出人员管理、招聘管理、绩效管理、奖惩管理、薪金管理、人事异动管理、培训管理等功能。

**3.4.1.3逻辑结构设计**

数据库逻辑结构设计阶段就是指出所需的实体和关系，并进行实体规范化。将上面的数据库概念结构转化为数据库所支持的逻辑数据模型。

本系统的数据库总共有6个表。

（1）应聘人员信息表。

这个表包含了所以应聘人员的基本信息，实际上也包含员工的基本信息。在这里，我们将员工基本信息表和应聘人员信息表融合了，也就是说，人员管理这一模块针对的员工基本信息在数据库层面来看是来源于应聘人员信息表的。本系统的员工基本信息是从应聘人员信息表提取的。所以表中需要一个类似权限的字段来判断人员到底是应聘人员还是已录取的员工。

下面来介绍一下该表中的各个属性。id是系统给应聘人员生成的编号，用来唯一用来区分不同的人员，name是人员的姓名，sex是人员的性别，birthday是人员的出生日期，age是人员的年龄，job是人员入职后的职位信息，specialty是人员的主修专业，experience是人员以前的工作经验，studyeffort是人员的学历，school是人员的学校，tel是人员的联系方式，email是人员的邮箱，createtime是人员应聘时间，content是人员的简历内容，isstock是简历是否入库的状态， health是员工入职前的体检结果。

（2）培训信息表。

id是培训计划编号，唯一标识不同培训记录，name是培训计划名称，purpose是这次培训计划的主要目的，begintime是人事管理者设置的培训开始时间，endtime是人事管理者设置的培训结束时间，datum是培训计划中所需要的教材、材料等等，teacher是这次培训的主要讲师，student是这次培训计划的人员，createtime是培训计划的上传时间，educate是员工培训的状态比如说未培训，effect是员工的培训效果，summarize是员工的培训总结。

（3）绩效信息表。

id是绩效信息编号，唯一标识不同考核时间、不同员工的绩效信息，content是绩效考核的具体内容，score是绩效考核成绩，addmoney根据绩效考核成绩得出的提成金额数，createtime是绩效考核的具体时间。

（4）奖惩信息表。

id是奖励编号，唯一标识每一条奖励记录，name是奖惩的具体内容，reason是奖惩原因，explains是有关于奖惩记录的细节方面或企业规章制度方面的描述，createtime是奖罚记录的时间。

（5）薪金信息表。

id是薪金编号，用来唯一区别每一条薪金记录，name是员工的姓名，basic是基本工资，reward是奖金，duty是全勤奖，punishment是罚金，other是提成，granttime是工资发放时间，totalize是工资总额，工资总额=基本工资+工资加成（提成、奖金、全勤奖）+工资扣除（罚金）。

（6）管理人员信息表。

id是系统给人事管理者生成的编号，用于唯一标识人事管理者，username是用户的真实姓名，password是用户的密码，sex是用户的性别，birthday是用户的出生日期，createtime是用户的创建时间，isadmin是用于判断该用户的身份与权限的字段，content是用户的简介。

3.4.1.4物理结构设计

数据库物理设计的任务是为上一阶段得到的数据库逻辑模式，即数据库的逻辑结构选择合适的应用环境的物理结构，既确定有效地实现逻辑结构模式的数据库存储模式，确定在物理设备上所采用的存储结构和存取方法，然后对该存储模式进行性能评价、修改设计，经过多次反复，最后得到一个性能较好、可用性较高的存储模式。数据库物理设计内容包括记录存储结构的设计，存储路径的设计。由于是测试实验，我们的数据库选用了MySQL的数据库，并将数据库架设在一台电脑上。

3.4.1.5数据库实施

数据库的实施主要是根据逻辑结构设计和物理结构设计的结果，在计算机系统上建立实际的数据库结构、导入数据并进行程序的调试。它相当于软件工程中的代码编写和程序调试的阶段。用具体的DBMS提供的数据定义语言(DDL)，把数据库的逻辑结构设计和物理结构设计的结果转化为程序语句，然后经DBMS编译处理和运行后，实际的数据库便建立起来了。

3.4.1.6数据库运行与维护

数据库系统投入正式运行，意味着数据库的设计与开发阶段的基本结束，运行与维护阶段的开始。数据库的运行和维护是个长期的工作，是数据库设计工作的延续和提高。

在数据库运行阶段，完成对数据库的日常维护，工作人员需要掌握DBMS的存储、控制和数据恢复等基本操作，而且要经常性地涉及物理数据库、甚至逻辑数据库的再设计，因此数据库的维护工作仍然需要具有丰富经验的专业技术人员(主要是数据库管理员)来完成。

数据库的运行和维护阶段的主要工作有：（1）对数据库性能的监测、分析和改善；（2） 数据库的转储和恢复；（3）维持数据库的安全性和完整性；（4）数据库的重组和重构。

### 3.4.2数据库设计依据

3.4.2.1数据库设计依据概述

人事管理信息系统是围绕人事管理者的需求开发的。首先是人事管理者自身的信息需要存储（验证密码后才能进入系统），所以建立一个管理人员表。

当人事管理者需要招聘员工时，应当先储存应聘人员信息，所以要建立应聘人员信息表存储应聘人员的基本信息（应聘成功的人员信息也来源于此表）。

当人事管理者有员工培训这一需求时，必须先编辑培训计划，所以要建立培训信息表存储员工培训信息，包括具体的培训内容、培训材料和讲师等等。

当人事管理者有员工薪金管理这一需求时，要从基本工资、绩效提成、奖金、罚金、全勤奖这五个方面综合计算总工资，所以要建立薪金信息表（存储员工总薪金信息）、绩效信息表（存储不同时间下不同员工的绩效考核结果以及提成）、奖惩信息表（存储不同时间下不同员工的奖惩记录以及奖惩金额），其中全勤奖单独由人事管理者输入金额。

当人事管理者有人事异动调整这一需求时，可使用应聘人员信息表修改员工的职位信息。

3.4.2.2数据库关系模型

**应聘人员信息**（员工编号（主键），员工姓名,性别,年龄.出生日期,联系电话,家庭住址,职位,工作经验,学历,学校,专业,邮箱,应聘时间,简历是否入库,体检结果）

**管理人员信息**（用户编号（主键），用户姓名，密码，性别，出生日期，注册时间，权限，简介）

**绩效信息**（绩效编号（主键），绩效考核内容，绩效考核时间，绩效考核成绩，提成）

**奖惩信息**（奖励编号（主键），奖惩内容，原因，备注，奖惩时间）

**薪金信息**（工资信息编号（主键），员工姓名，基本工资，提成，奖金，全勤奖，罚金，工资发放时间，总工资）

**培训信息**（培训计划编号（外键）,培训计划名称,培训目的,培训开始时间,培训结束时间,培训材料,培训人员,讲师,上传计划时间,是否培训,培训效果,培训总结）

3.4.2.3 DBMS和 OOD类图描述（互相校验）

与OOD类图相比，DBMS关系图体现的是数据库中各数据表之间的关系。其中用户表（对应OOD类图的人事管理者类，记录人事管理者用户名、密码等基本信息）、招聘计划表和招聘计划内容表（对应OOD类图的招聘计划类）与员工编号无关，因此在DBMS关系图中没有体现出来。下面具体分析OOD类图中的各个类在DBMS关系图中对应的数据表：

（1）应聘人员信息类对应应聘人员信息表job，记录人员的员工编号、姓名、联系方式等基本信息。

（2）培训基本信息类对应培训信息表education，记录不同员工在不同时段下的培训信息。

（3）奖惩记录类对应奖惩信息表institution，记录不同员工在不同时段下的奖惩信息。

（4）绩效考核类对应绩效信息表performance，记录不同员工不同时段下的绩效信息。

（5）薪金信息类对应薪金信息表stipend，记录不同员工的工资信息包括基本工资、提成、奖金、罚金、全勤奖。其中，基本工资随职位的不同而发生改变的。

（6）管理人员信息类对应管理人员信息表users，记录管理员的编号、姓名、登录密码、权限等基本信息。

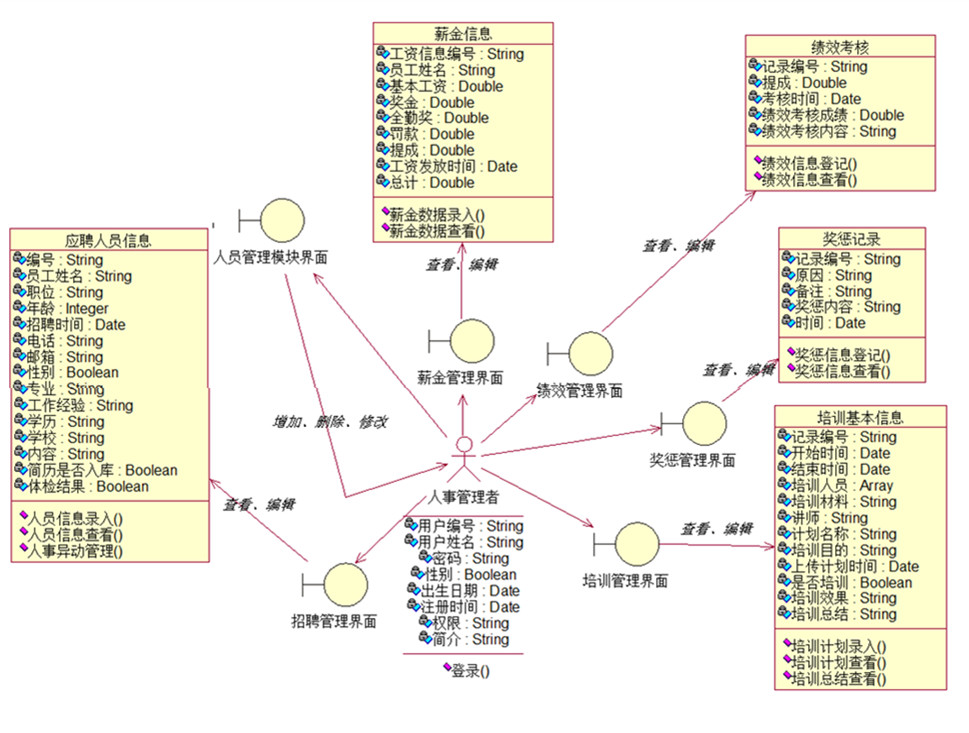


图3.4.2.3.1人事管理信息系统OOD类图

### 3.4.3数据库表设计

人事管理信息系统数据库总共包括12个表，它们分别为员工基本信息表、招聘计划表、招聘计划内容表、合同表、培训信息表、绩效信息表、考勤信息表、奖励信息表、惩罚信息表、工资信息表、基本工资表、用户表。

3.4.3.1表设计

（1）应聘人员信息表：

表3.4.3.1.1应聘人员信息表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 应聘人员信息表（job） | | | | | |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 是否为主键/外键 |
| id | Id | 11 | 员工编号 | 否 | 主键 |
| name | varchar | 50 | 员工姓名 | 是 |  |
| sex | varchar | 2 | 性别 | 否 |  |
| age | int | 11 | 年龄 | 是 |  |
| job | varchar | 50 | 职位 | 是 |  |
| specialty | varchar | 50 | 专业 | 是 |  |
| experience | varchar | 50 | 工作经验 | 是 |  |
| studyeffort | varchar | 50 | 学历 | 是 |  |
| school | varchar | 50 | 学校 | 是 |  |
| tel | varchar | 50 | 电话 | 是 |  |
| email | varchar | 50 | 邮箱 | 是 |  |
| createtime | datetime | - | 应聘时间 | 是 |  |
| content | text | - | 内容 | 是 |  |
| isstock | bit | 1 | 简历是否入库 | 是 |  |
| health | varchar | 50 | 体检结果 | 是 |  |

（2）培训信息表：

表3. 4.3.1.2培训信息表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| id | int | 11 | 培训计划编号 | 否 | 主键 |
| name | varchar | 100 | 培训计划名称 | 否 |  |
| purpose | varchar | 500 | 培训目的 | 否 |  |
| begintime | datetime | - | 培训开始时间 | 是 |  |
| endtime | datetime | - | 培训结束时间 | 是 |  |
| datum | text | - | 培训材料 | 是 |  |
| teacher | varchar | 50 | 讲师 | 是 |  |
| student | varchar | 50 | 培训人员 | 是 |  |
| createtime | datetime | - | 上传计划时间 | 是 |  |
| educate | bit | 1 | 是否培训 | 是 |  |
| effect | varchar | 500 | 培训效果 | 是 |  |
| summarize | text | - | 培训总结 | 是 |  |

（3）绩效信息表：

表3. 4.3.1.3绩效信息表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| id | int | 11 | 绩效信息编号 | 否 | -1 |
| content | varchar | 50 | 绩效考核内容 | 是 | 0 |
| score | float | 8 | 绩效考核成绩 | 否 | 0 |
| addmoney | float | 8 | 提成 | 是 | 0 |
| createtime | datetime | - | 绩效考核时间 | 是 | 0 |

（4）奖惩信息

表3. 4.3.1.4奖惩信息表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| id | int | 11 | 奖惩编号 | 否 | 主键 |
| name | varchar | 50 | 奖惩内容 | 否 |  |
| reason | varchar | 50 | 原因 | 是 |  |
| explains | text | - | 备注 | 是 |  |
| createtime | datetime | - | 奖惩时间 | 是 |  |

（5）管理信息表：

表3. 4.3.1.5管理人员信息表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| id | int | 11 | 用户编号 | 否 | 主键 |
| username | varchar | 50 | 用户姓名 | 否 |  |
| password | varchar | 50 | 密码 | 否 |  |
| sex | bit | 1 | 性别 | 是 |  |
| Birthday | datetime | - | 出生日期 | 否 |  |
| createtime | datetime | - | 注册时间 | 否 |  |
| Isadmin | bit | 1 | 权限 | 是 |  |
| Content | text | - | 简介 | 是 |  |

（6）薪金表：

表3. 4.3.1.6薪金信息表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 描述 | 可否为空 | 主键/外键 |
| Id | int | 11 | 工资信息编号 | 否 | 主键 |
| Name | varchar | 50 | 员工姓名 | 否 |  |
| Basic | float | 10 | 基本工资 | 是 |  |
| Reward | float | 8 | 奖金 | 是 |  |
| Duty | float | 8 | 全勤奖 | 是 |  |
| punishment | float | 8 | 罚款 | 是 |  |
| Other | float | 8 | 提成 | 是 |  |
| Granttime | datetime | - | 工资发放时间 | 是 |  |
| Totalize | float | 10 | 总计 | 是 |  |

3.4.3.3表实现代码

create database HRmanagement;  
use HRmanagement;  
CREATE TABLE `job` (

`id` int(11) NOT NULL auto\_increment,

`name` varchar(50) NOT NULL,

`sex` bit(1) default 0,

`age` int(11),

`job` varchar(50) default NULL,

`specialty` varchar(50) default NULL,

`experience` varchar(50) default NULL,

`studyeffort` varchar(50) default NULL,

`school` varchar(50) default NULL,

`tel` varchar(50) default NULL,

`email` varchar(50) default NULL,

`content` text default NULL,

`createtime` datetime default NULL,

`isstock` bit(1) default 0,

`health` varchar(10) default `合格`,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='应聘人员信息表';  
  
CREATE TABLE `educate` (

`id` int(11) NOT NULL auto\_increment,

`name` varchar(100) NOT NULL,

`purpose` varchar(50) NOT NULL,

`begintime` datetime default NULL,

`endtime` datetime default NULL,

`datum` text default NULL,

`teacher` varchar(50) default NULL,

`student` varchar(50) default NULL,

`createtime` datetime default NULL,

`educate` bit(1) NOT NULL,

`effect` varchar(50) default NULL,

` summarize ` text default NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='培训信息表';  
  
CREATE TABLE `performance` (

`id` int(11) NOT NULL auto\_increment,

`content` varchar(50),

`score` float NOT NULL,

`addmoney` float default 0,

`createtime` datetime default NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='绩效信息表';  
  
CREATE TABLE `institution` (

`id` int(11) NOT NULL auto\_increment,

`name` varchar(50) NOT NULL,

`reason` varchar(50) default NULL,

`explains` text default NULL,

`createtime` datetime default NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='奖惩信息表';

CREATE TABLE `stipend` (

`id` int(11) NOT NULL auto\_increment,

`name` varchar(50) NOT NULL,

`basic` float default NULL,

`reward` float default NULL,

`duty` float default NULL,

`punishment` float default NULL,

`other` float default NULL,

`totalize` float default NULL,

`granttime` datetime default NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='薪金信息表';

CREATE TABLE `users` (

`id` int(11) NOT NULL auto\_increment,

`username` varchar(50) NOT NULL,

`password` varchar(50) NOT NULL,

`sex` bit(1) default 0,

`isadmin` bit(1) default 0,

`content` text default NULL,

`birthday` datetime NOT NULL,

`createtime` datetime NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='人员信息表';

3.4.3.3数据库的表实现

（1）员工基本信息表：

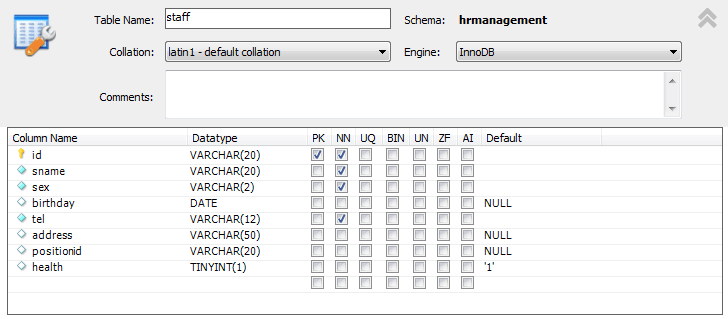


图3.4.3.3.1员工基本信息表

（2）招聘计划表：

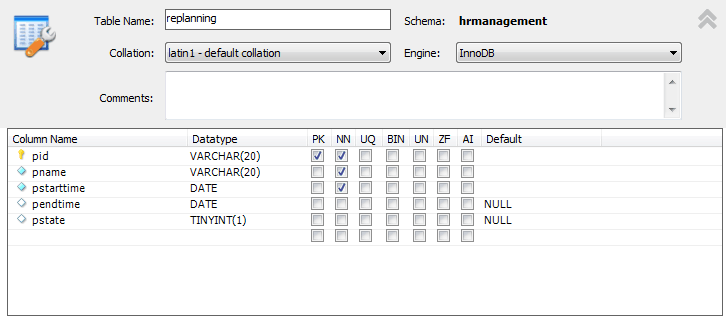


图3.4.3.3.2招聘计划表

（3）招聘计划内容表：

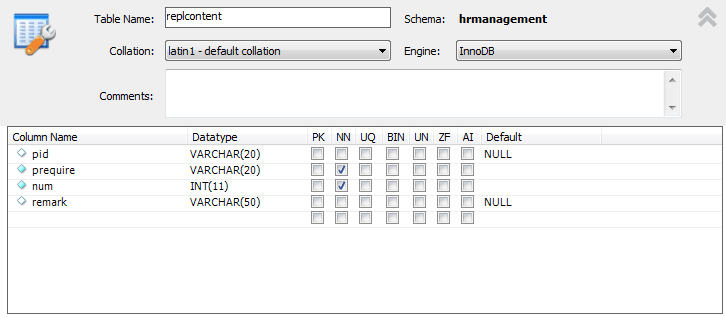


图3.4.3.3.3招聘计划内容表

（4）合同表：

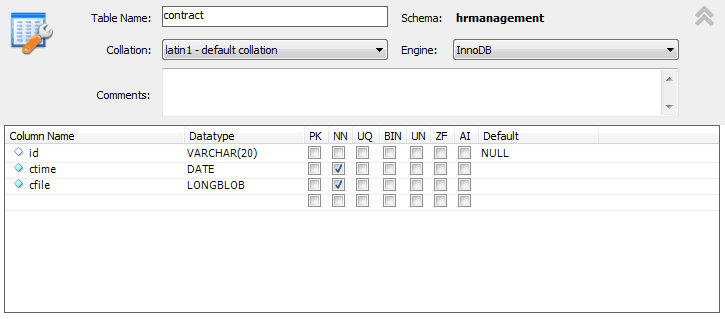


图3.4.3.3.4合同表

（5）培训信息表：

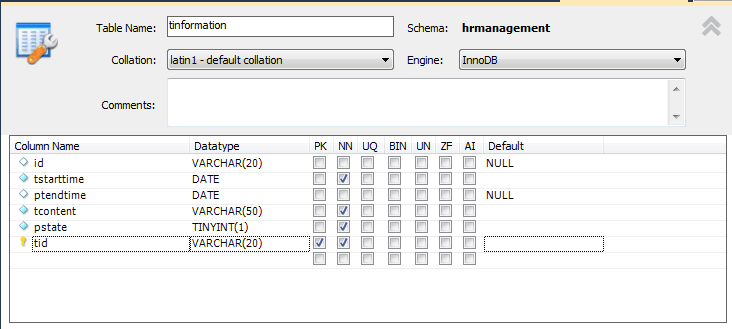


图3.4.3.3.5培训信息表

（6）绩效信息表：

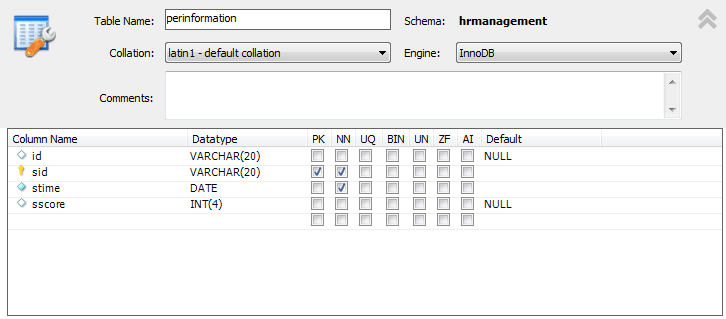


图3.4.3.3.6绩效信息表

（7）考勤信息表：

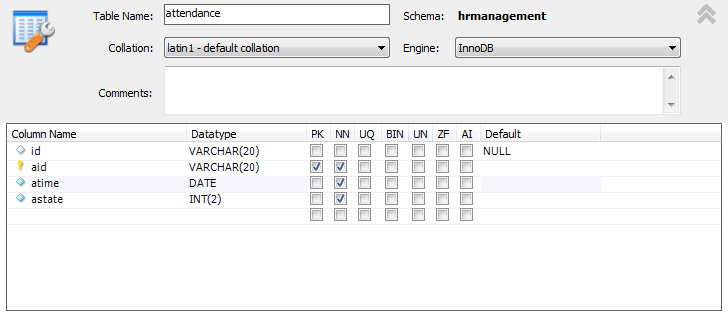


图3.4.3.3.7考勤信息表

（8）奖励信息表;

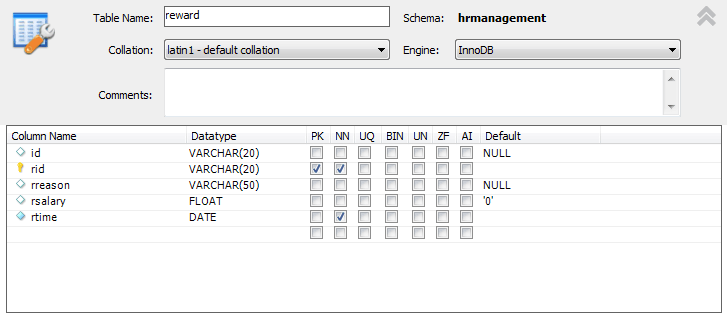


图3.4.3.3.8奖励信息表

（9）惩罚信息表：

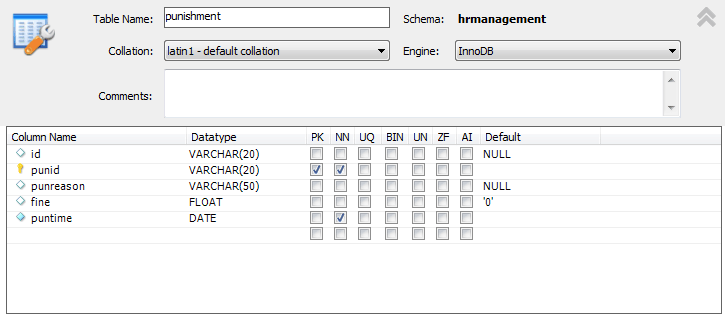


图3.4.3.3.9惩罚信息表

（10）工资信息表：

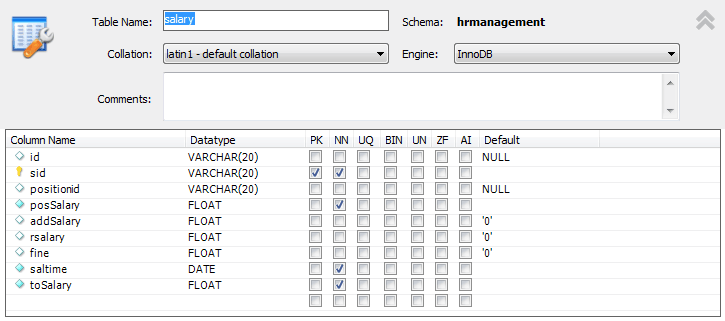


图3.4.3.3.10工资信息表

（11）基本工资表：

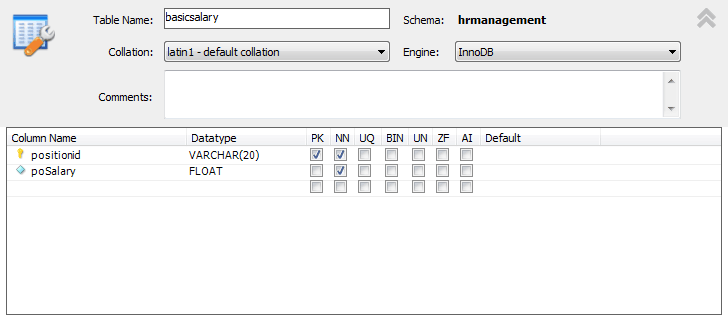


图3.4.3.3.11基本工资表

（12）用户表：

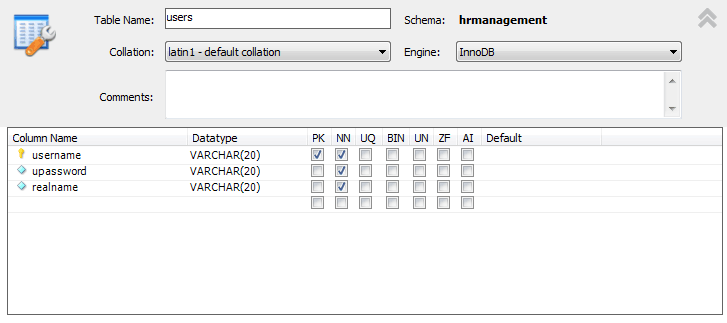


图3.4.3.3.12用户表

## 3.5控制驱动设计

### 3.5.1控制驱动设计概述

系统运行时，当一个主动对象被创建，其主动操作将被创建为一个控制流；所有的主动对象由主动类描述，所有的主动类构成系统设计模型的控制驱动部分。进程（process）是重量级控制流，具有一定独立功能的程序的一次执行过程。线程（thread ）是轻量级控制流，在一个进程内部可定义一些能分别占用处理机，而且要同时计算的执行单位。控制驱动概念：

（1）OOD 模型的外围组成部分之一，由系统中全体主动类构成；

（2）所有的主动类构成控制流驱动部分；

（3）主动类描述了整个系统中所有的主动对象；

（4）每个主动对象是系统中一个控制流的驱动者；

（5）用于定义和表示并发系统中的每个控制流（进程、线程的总称）。

系统有多个控制流并发进行称为系统的并发性，其基本思想是把并发程序分解成一些顺序执行的进程，使得每个进程内部不再包含并发行为，避免了并发问题，同时，多个进程之间是并发（异步）执行的，所以能够构成并发程序。

### 3.5.2控制驱动设计依据

1、OOD类图中以人事管理者类为例。人事管理者在进行人员招聘、绩效管理、人员管理、奖惩管理、培训管理、薪金管理、人事管理等具体操作时需要进入不同的模块，也就是说人事管理者进入多个不同的模块会产生多个进程。而人事管理者在不同的模块里再产生不同的子操作的时候（比如说在薪金管理模块里选择修改基本工资），会产生线程。根据人事管理者的需求，系统应该是可以多进程多线程的，比如说人事管理者既可以在工资管理模块里修改基本工资和查看员工总工资，同时也可以在人员管理模块查询员工信息。

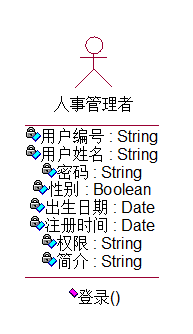


图3.5.2.1 人事管理者类图

2、泳道活动图中以人事异动管理活动图为例。人事管理者登录人事管理信息系统后选择进入人事异动管理模块，相当于系统创建了一个进程（完成人事异动后关闭）。在人事异动管理模块中，用户可以选择查看员工人事信息以及修改操作（包含员工调职、离岗），相当于系统可以同时创建了两个线程（完成操作后关闭）。其中，人事信息修改这一操作还涉及到人员管理模块的员工职位信息修改，也就是说，人事信息修改这一操作已完成，系统将自动创建一个线程将数据库对应的表中的员工职位状态进行修改。

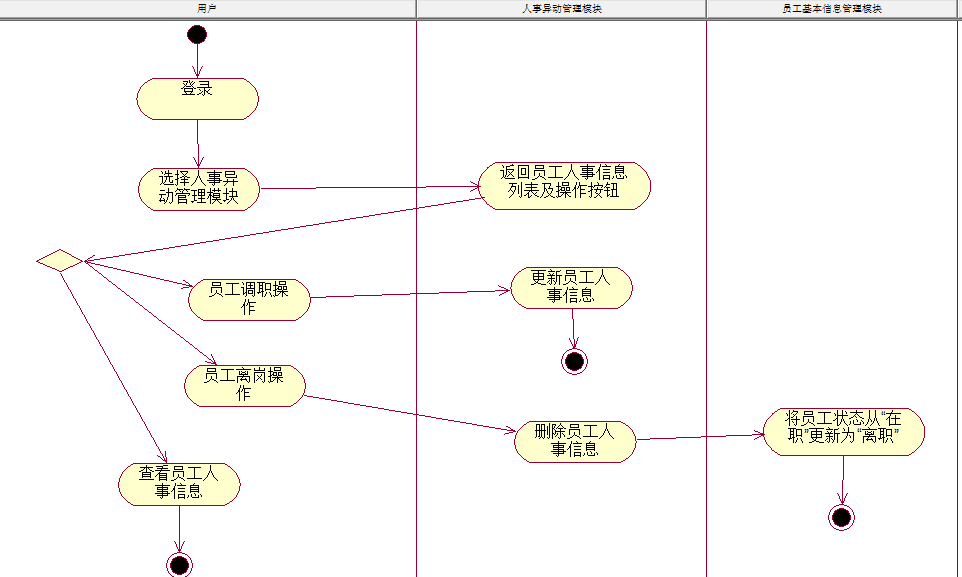


图3.5.2.2 人事异动管理活动图

3、状态图中以员工培训状态图为例。人事管理者登录人事管理信息系统后选择进入员工培训管理模块，相当于系统创建了一个进程（完成员工培训相关操作后关闭）。在员工培训管理模块中，用户可以选择发布培训计划（针对一类职位的员工进行培训，避免人员的遗漏），也可以选择查看历史培训的培训效果以及培训总结，相当于系统可以同时创建了三个线程（完成操作后关闭）。

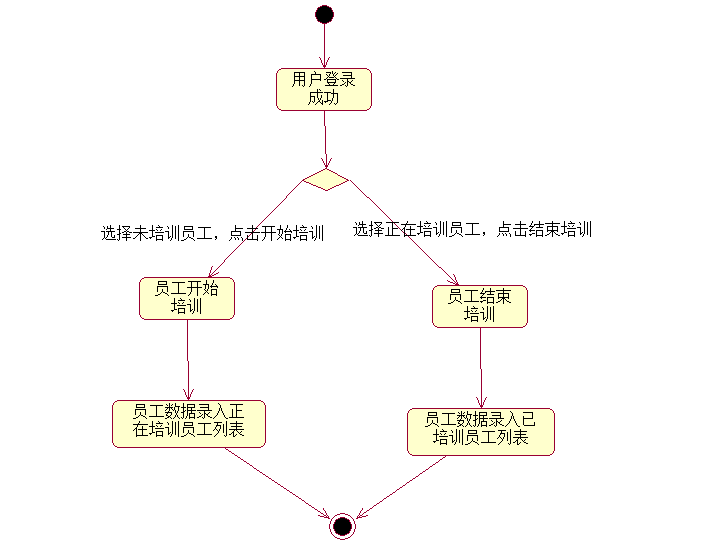
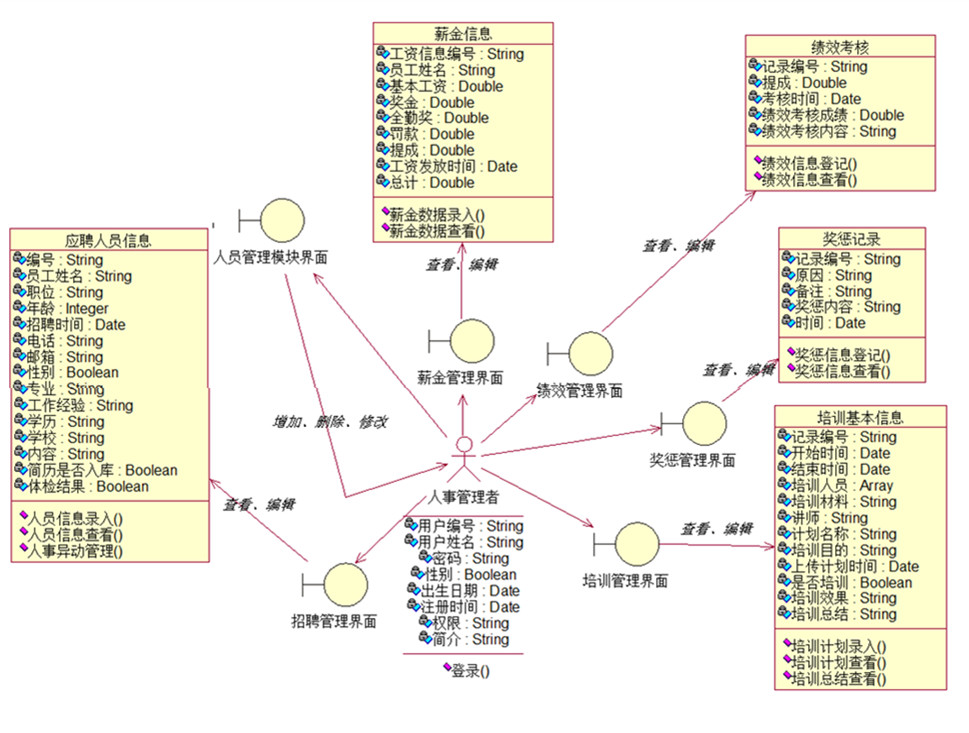
****

图3.5.2.3 员工培训状态图

### 3.5.3系统并发控制设计

当人事管理者登录人事管理信息系统时，“人事管理者”这个主动类创建了一个具体的主动对象“XX”，之后这个对象进行了一系列的操作，例如：人员管理（包含人员信息录入、人员信息查看）、招聘管理（包含应聘信息录入、应聘信息查看、人才库浏览）、人事异动（包含人事异动管理）、培训管理（包含培训计划录入、培训计划查看、培训总结查看）、绩效管理（包含绩效信息登记、绩效信息查看）、奖惩管理（包含奖惩信息登记、奖惩信息查看）、薪金管理（包含薪金数据录入、薪金数据查看）等，这一系列的操作形成了一个控制流。



控制流：

查看应聘人员信息

修改应聘人员信息

查看薪金信息

编辑薪金信息

查看绩效考核

编辑绩效考核

…….

图3.5.3以“人事管理者”主动类形成的控制流

### 3.5.6系统总体方案设计

由系统总体方案决定的实现条件：

计算机硬件：性能、容量和 CPU 数目；

操作系统：对并发和通信的支持：

网络方案：网络软硬件设施、网络拓扑结构、通信速率、网络协议等；

软件体系结构：抽象地说，软件体系结构描述了构成系统的元素、这些元素之间的相互作用、指导其组合的模式以及对这些模式的约束；-- Mary Shaw；

编程语言：对进程和线程的描述能力；

其他商品软件：数据管理系统、界面支持系统、构件库等对共享和并发访问的支持。

### 3.5.7软件体系结构设计

3.5.7.1典型的软件体系结构风格

管道与过滤器风格（pipe and filter style）；

数据抽象风格（data abstraction style）；

面向对象风格（object-oriented style）；

隐式调用风格（implicit invocation style）；

层次风格（layered style）；

仓库风格（repository style）；

黑板风格（blackboard style）；

解释器模型（interpreter model）；

进程控制风格（process control style）；

客户-服务器风格（client-server style） ；

3.5.7.2分布式系统的体系结构的风格

主机+仿真终端体系结构；

文件共享体系结构；

客户-服务器体系结构；

二层客户-服务器体系结构；

三层客户-服务器体系结构；

对等式客户-服务器体系结构；

瘦客户-服务器体系；

浏览器-服务器体系结构；

## 3.6构件部署设计

### 3.6.1面向对象构件方法与技术概述

3.6.1.1构件概述

UML 支持对逻辑构件（如业务构件、过程构件）和物理构件（如 EJB 构件、CORBA 构件、COM+ 构件和 .NET 构件以及 WSDL 构件）建模。

UML 支持对实现构件的制品、对可执行的构件的部署和对构件执行在其上的节点的规约，即意味着在基于构件的开发中，在不同的阶段有着不同的构件模型，供不同的开发人员使用。

3.6.1.2构件的相关概念

3.6.1.2.1构件的概念

（1）定义

在构件软件系统中具有相对独立功能、可明确辨识、接口由契约指定、语境有明显依赖关系、可独立部署且多由第三方提供的可组装软件实体。

（2）含义

一个构件表示系统的一个模块部分，而且是一个自包含的单元，它封装了其内部成分的状态和行为。

按照提供和请求接口，构件定义其行为。

构件具有属性、操作和可见性，并能参与关联和泛化。

可使用 用例图、活动图、顺序图、状态机图描述构件的外部行为。

用制品实现构件，部署到运行环境中运行。

（3）作用

可执行的构件是可实例化的。

可以用构件来装配大粒度的构件，方法为把所复用的构件作为大粒度构件的成分，并把它们的请求和提供接口连接在一起。

构件是可替换的单元，在设计时和运行时基于接口的兼容性，若一个构件能提供功能与另一个构件相同的功能，则前者就能替换后者。

（4）分类

基本构件：注重于把构件定义为在系统中可执行的元素。

包装构件：扩展了基本构件的概念，它注重于把构件定义为一组相关的元素，这组元素为开发过程的一部分，即包装构件定义了构件的命名空间方面，在构件的命名空间中，可以包括类、接口、构件、包、用例、依赖（如映射）和制品。

3.6.1.2.2构件的接口

（1）定义：

表示对一组相关的操作进行声明的一种建模元素，它也指定了一种契约，这契约必须由实现这个接口的构件的任何实例完成。

（2）接口的分类

提供接口（provided interfaces）：构件实现的接口，给其它构件提供服务；。

请求接口（required interfaces）：构件使用的接口，向其它构件请求服务时要遵循的接口。

（3）接口的含义：

构件及其子构件可以直接实现提供接口。

实现接口的构件支持由该接口所拥有的特征。

构件必须与接口拥有的约束相容。

一个构件可以实现多个接口，一个构件可以请求多个接口，一个接口可以由多个不同的构件实现。

只要一个构件服从由另一个构件的提供和请求接口表达的约束，它就能和这样的环境进行交互。

通过添加新增功能的新构件类型，能够扩展系统。

（4）构件和接口关系的表示

* 简略图符方式
* 扩展方式
* 分栏矩形方式

3.6.1.2.3构件的端口

（1）定义

一个封装构件的显式的对外窗口，所有进出构件的消息（调用或信号）都通过端口。接口对声明一个构件的总的行为而言是有用的，构件的实现仅需保证要实现在全部的提供接口中的操作。使用端口是为了能进一步控制这种实现。

（2）表示

一个构件边上和小方块，代表了一个穿过构件的封闭边界的窗口，可将接口附属到端口符上，端口有标识。

（3）作用

构件内部的实例能用端口名称标识收发消息的端口。

3.6.1.2.4连接

（1）端口连接

若一个端口提供一个特定的接口而另一个端口需要这个接口，且接口是兼容的，则两个端口是可连接的。构件可通过特定的端口与另一构件通信，且通信完全通过端口支持的接口描述。供接口说明通过端口提供服务，需接口说明通过端口从其他构件获取服务。使用端口能在更大程度上增加构件的封装性和可替代性。

（2）连连接

两个端口之间的衔接。连接端口意味着请求端口要调用提供端口中的操作，以得到服务。设立端口和接口的优点在于在设计时，两个构件彼此不需要了解对方的内部，只要其接口是相互兼容的即可。若需要使用新的构件，则可以把这些端口重新连接到其它提供相同接口的构件上。

分类：

装配连接件（assembly connector）：两个构件实例间的连接件，用于连接一个需接口或端口实例与一个供接口或端口实例。

委托连接件（delegation connector）：把外部对构件端口实例的请求分发到构件内部的部件实例进行处理，或通过构件端口实例将内部的部件实例向构件外部的请求分发出去。被委托的端口实例必须提供被委托的功能。必须在两个提供端口实例间或两个请求端口实例间定义委托连接件。具体的消息流将发生在所连接的端口实例之间，可能跨越多个层次，最终达到对消息进行处理的最终部件实例，即可使用委托连接件按层次分解的方式对构件建模。对于来自外部端口实例的消息被委托给内部部件实例，将委托连接件表示为从源端口实例到处理请求的目标对象的线段。

### 3.6.2构件设计

3.6.2.1构件的内部结构

构件由实现它的一个或多个部件组成，把这些部件以及其间的关系称之为它的内部结构。

装配连接件和委托连接件的关系：内外交互。

3.6.2.2构件间的关系

1、关联：构件、提供接口和请求接口以及通过依赖的线接示例。

在组合结构图中，通过定义内部构件和连接件，可以在角色或实例级详细描述衔接。装配连接件是关联的实例。

一旦构件之间建立的关联，在构件运行时，一个构件的实例就可以实例化另一个构件，或访问另一个构件的实例；一个构件实例可创建几个其它构件的实例，可能进而访问，也可能供其它实例访问；一个构件实例不创建另一个构件的实例，而通过获取的标识的方式再进行访问，但要遵从构件间的可见性。

2、聚合：构件v、x和w作为构件u的组成成分，构件z作为构件w的组成部分，在组成关系上还标有对外可见性。

3、依赖：构件间的访问关系实际上就是构件接口间的依赖（通常在结构图上），在结构上定义构件间的线接。但这种关系是一种概貌性的关系，一般是中间结果的产物，最终要精确到关键间的关联或聚合。

4、继承：构件之间可以有泛化关系，但只是应用在规约阶段；

特殊构件继承一般构件的规约，并对所继承的规约可进行修改，但不能添加不变式。

为了保证特化在逻辑上与泛化相符合，新操作的前置条件必须等于或弱于旧操作的前置条件，且新操作的后置条件必须等于或强于旧操作的后置条件，即，特化构件的行为模型必须在协议和结构上与泛化构件相符合；在状态机上，特化的要保留泛化的东西。

3.6.2.3对构件的动态行为的描述

可以在接口、端口上和构件本身上附属诸如协议状态机这样的行为。可以通过按调用操作的顺序显式地给出动态约束，以较精确地定义外部行为。可以把其它的行为（如用况、活动或交互规约）与接口或连接件相关联，以定义协作间的“契约”。

1、制品图

制品（Artifact）是具体的开发产物，用于对构件的物理实现进行建模，它要被部署到执行环境去运行。如源代码（«source»），可执行文件«executable»。

（1）对构件的源代码建模。识别出一组相关源代码文件集合。考虑设置一个标记值，用它给出源代码文件的版本号、作者名或最后修改日期等信息。用依赖关系对这些文件之间的编译依赖关系建模。

（2）对可执行体的发布建模。识别要建模的制品集合。通常，它应包括一个结点上的部分或全部制品，或者集跨越系统中所有结点的这样制品集的分布。

2、部署图

在运行环境中要部署（deployment）一个或几个制品，部署一个制品往往要有部署规约（deployment specification）。

部署制品和部署环境间的关系可定义在：以下两个级别上：类型级别： 在“application server” 上定义 “order entry request handler” 执行制品。实例级别： 在 3个具体 “appserver1” ... “app-server3” 上部署6个 “request handler\*” 实例。

节点(node)是可计算资源，制品要部署在其上执行。通过通信路径把节点相连，即定义了网络结构。典型的节点为设备和计算环境。

部署规约详述了一组性质，用于决定部署到节点上的构件制品执行参数。设备是物理的可计算资源，具有处理能力。即被部署的制品在其上执行。通信路径（及关联）连接部署目标，对其间的信号和消息建模。 在节点实例中放置执行环境（ ExecutionEnvironment ）实例，其间为组成关系。执行环境示例： «OS», «workflow engine», «database system», and «J2EE container».。执行环境可是嵌套的。

一个节点上可以有一个或多个制品，一个制品也可以部署在一个或多个节点上。节点之间最常见的关系是关联关系，用来表示节点之间的物理连接。对单机式、嵌入式和分布式系统拓扑结构中的处理器和设备，都可以用部署图进行建模。

3.6.3系统部署设计

1、部署图

在运行环境中要部署（deployment）一个或几个制品，部署一个制品往往要有部署规约（deployment specification）。

2、部署制品和部署环境间的关系

（1）类型级别：在“application server”上定义“order entry request handler”执行制品。

（2）实例级别：在 3个具体 “appserver1” ... “app-server3”上部署 6 个 “request handler”实例。

（3）节点（node）：可计算资源，制品要部署在其上执行。通过通信路径把节点相连，即定义了网络结构；典型的节点为设备和计算环境；一个节点上可以有一个或多个制品，一个制品也可以部署在一个或多个节点上；节点之间最常见的关系是关联关系，用来表示节点之间的物理连接。

## 3.7体系结构设计

### 3.7.1系统的分层结构设计

#### 3.7.1.1本系统分层结构

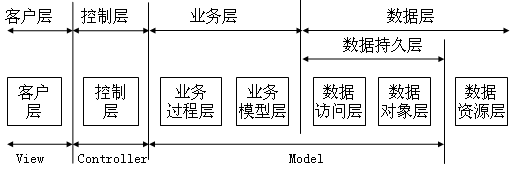


图3.7.1.1.1 系统分层结构图

#### 3.7.1.2架构与组件的选择

根据系统的分层结构，该系统选择符合J2EE规范的Tomcat中间件。客户层选用JSP组件；业务流程层选用Action组件；业务模型层、数据访问层和实体对象层选用JavaBean组件；控制层选用FilterDispatcher 组件。

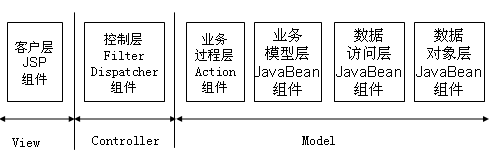


图3.7.1.2.1系统结构组件图

#### 3.7.1.3架构与组件的选择

运行环境设计包括网络结构，硬件、软件环境。

（1）本系统的网络拓扑参考图



图 3.7.1.3.1 网络拓扑参考图

本系统提供本地和远程两种访问系统的用户机，及局域网用户机和远程用户机。

数据库服务器：

 IP：192.168.1.114

 网关：192.168.1.1

 端口：3306

 用户名：root

 密码：123456

Web服务器：

IP：192.168.1.114

网关：192.168.1.1

端口：8080

域名：<http://pc-shy:8080/JavaPrj_5/logon.do?action=logon>

（2）硬件环境

下面是我们这个系统使用的硬件环境：

客户机：普通PC

CPU： P4 1.8GHz以上

Memory ：512MB以上

能运行IE6.0以上的机器

分辨率：推荐使用1024\*768

Web服务器

CPU： P4 2.4GHz

Memory ：1GB以上

HDD：80GB以上

网卡：KMb/s速度网卡

数据库服务器

CPU： P4 2.4GHz

Memory：1GB以上

HDD：80GB以上

（3）软件环境

软件环境是指操作系统，数据库系统以及一些开发工具的选用。本系统开发所需要的软件环境如下：

操作系统：Windows2000及以上版本

数据库：MySQL6.0；Navicat for MySQL

开发工具包：JDK Version 1.6

开发环境：MyEclipse6.0

Web服务器：Tomcat7.0

浏览器：IE6.0或以上版本

应用框架：Struts2.0，Hibernate，Spring

### 3.7.2系统的正交设计

#### 3.7.2.1正交设计的概念

（1）定义

正交设计是解决领域问题结构设计的一种解耦方法 。

（2）设计指导思想

以事件为线索，按照水平和垂直(横/纵)两个方向交叉进行找出V、M、E的设计内容。

（3）方法

先横后纵。

横向是链间解耦、层间依赖：按一个事件链执行顺序从源到尾找出层间依赖；

纵向是层间解耦链间综合：将各链按相同层对象进行综合考虑该层总体设计，就是划分出V、M、E三层，然后隔离开单独设计；

完成了V、M、E三层，层与层之间的联系就用架构或框架来沟通(这就是基于架构的设计)，比如C层设计采用MVC框架解决事件驱动管理，路由V/M间的联系；利用O/R映射框架设计DAO持久层并解决M/E之间的联系。

（4）步骤

步骤1：细化用例图，确定每条执行链的操作用例；

步骤2：横向设计执行链的各层对象（组件）间依赖关系；

步骤3：纵向按照MVE三大部分进行总体设计；

步骤4：对MVE三大部分进行装配；

<1>确定操作用例

系统在分析阶段建立的用例图有的可能表达不到操作用例，但正交设计就是分析到所有的操作用例。因此设计人员可以根据需求对分析模型的每个用例考虑是否是“叶子”，是否有必要补充这些“叶子”。

<2>横向按执行链设计组件间关系

每条执行链都表达了其各层组件之间的依赖关系(消息传递)。通过对这种依赖关系的分析，为各组件分配行为。

先不考虑链间关系，也就是无论链间如何纠结，都要先解耦。这样做可以使设计人员集中精力来寻找层间组件的依赖关系(横向关系)。

第一步的按链分析会做很多重复性的工作，它遵循的核心理念是以重复换简单，以简单换清醒，以清醒换全面。每个层的整合设计就是下面纵向设计考虑的问题。

执行链的各层组件就是由事先的体系结构设计确定的。组件的选取决定于选定的企业架构。每层组件值守在自己的层内，在事件发生时忠实地执行自己的使命。

<3>纵向按MVE三层结构设计

在横向按执行链寻找完组件之间的依赖关系后，按纵向四个层进行层内的结构设计。由于层间是隔离的，每个层可独立进行，达到层间分离、链间综合。

层间隔离：MVC分层的目的，就是为不同的层提供隔离的空间；而架构为不同层提供了相应的组件。MVC是将同类组件组合在同一个空间里，方便开发和维护。SSH组合框架，空间的隔离更是清晰。Struts管理Web层事件的驱动和响应，即提供C层来配置V/M路由；Hibernate管理DAO层，提供O/R映射建立数据对象与资源层的联系；资源层由DBMS管理；Spring管理事件的执行链即业务层。

链间综合：就是对在同一层内的对象进行总体设计。解决按链分解时存在的对象冗余。链间综合就是对层内的对象应用合并、接口、继承、关联、组合、多态等技术进行结构设计，优化软件结构。

<4>层间进行配置

系统从OOA进行分解之后，分解出的视图层V、逻辑层M和资源层E，三个部分从设计到实现，可以在总工程师的协调下交由三个开发团队一竿子到底分别完成。其中有三个配置是由三个团队高层处理：一个是V/M层间的映射控制器C；一个是M/E之间的映射O/R配置；再一个是业务层的事件执行链配置。

（5）总结

核心思想就是解耦，然后再装配。

先水平方向按链间解耦，水平方向是一个链的层间对象依赖关系；后垂直方向按MVCE进行结构划分(层间解耦)，层间解耦的基础是采用基于架构的组件。MVE三大结构分别由对应专业人员独立设计和实现，然后进行映射或组装；控制器C映射V和M，ORM映射O和R，Spring和SCA可以组装M。

#### 3.7.2.2正交设计之横向设计

（1）定义：

按分层体系结构寻找各层组件的属性与方法。属性一般较好确定，麻烦的是找出方法。确定组件中的方法要依据对象交互图(时序图/协作图)的消息传递来划分。方法是：根据功能模块图找到事件层，对每个功能模块中的单个事件采用正交设计按横向链画出对象交互图。

（2）步骤

<1>根据需求文档和系统总体结构细化各子模块结构，分解到最底层的操作用例，每个用例的业务功能已在OOA的用例描述中说明；

<2>依照用例功能描述设计执行链，既画出每个操作用例的对象时序图。注意这里单链不用画出C；

<3>依据上一步时序图的消息传递，寻找出各层对象封装的方法。每个链的设计结果用表格式列出。封装说明就是列出该组件的属性和方法；

<4>如果是Web软件，建议采用Web版型(提供web建模元素)画出单个执行链的类关系图。

（3）对人事异动管理事件画对象交互图为对象分配行为

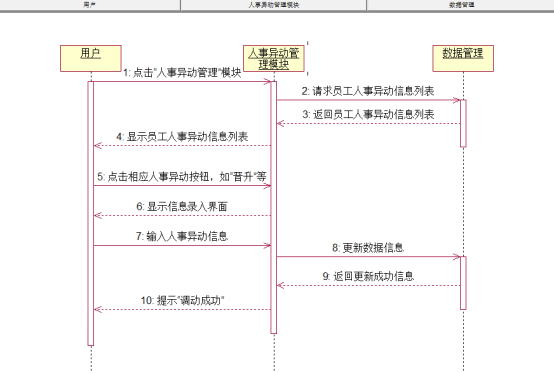
****

图3.7.1.2.1 人事异动管理顺序图

表3.7.1.2.1 人事异动管理模块横向设计说明

|  |  |  |  |
| --- | --- | --- | --- |
| **模块名称** | **程序文件** | **功能说明** | **封装说明** |
| 界面 | updatejob.jsp | 人事管理者选择“人事异动”，显示员工的部分信息（包括职位信息），每一项后面都有“修改”、“删除”按钮 | 输入项：员工需要修改的职位信息 |
| 业务流程BP | JobForm.java | 响应HTTP请求，获取前台表单数据id；  用获取的表单参数向BM层对象传递消息；  成功后返回到updatejob.jsp | reset()  validate() |
| 业务模型BM | JobAction.java | 显示员工职位信息  删除职位信息  更新职位信息 | Detailjob()  deleteJob()  updateJob()  execute() |
| 对象持久化DAO | JobDao.java | 为BM层提供对象持久化操作 | updateJob() |

（4）对薪金录入事件画对象交互图为对象分配行为

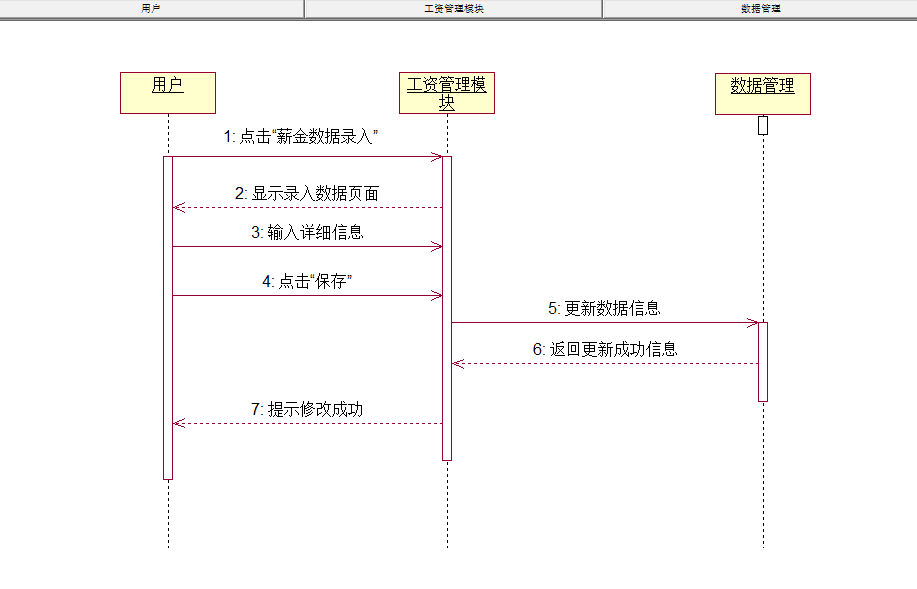


图3.7.1.2.2 薪金数据录入顺序图

表3.7.1.2.2 薪金数据录入横向设计说明

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 程序文件 | 功能说明 | 封装说明 |
| 界面 | addstipend.jsp | 人事管理者选择“薪金管理”下拉菜单中的“薪金数据录入”，显示需要填写的文本框，最后有“提交”、“重置”按钮 | 输入项：员工姓名、基本薪金、全勤奖、提成、罚款、发放时间 |
| 业务流程BP | StipendForm.java | 响应HTTP请求，获取前台表单数据；  用获取的表单参数向BM层对象传递消息；  成功后跳转到liststipend.jsp | reset()  validate() |
| 业务模型BM | StipendAction.java | 录入薪金 | addStipend()  execute() |
| 对象持久化DAO | StipendDao.java | 为BM层提供对象持久化操作 | addStipend ()  getCountTotalize() |

（5）对薪金数据修改事件画对象交互图为对象分配行为

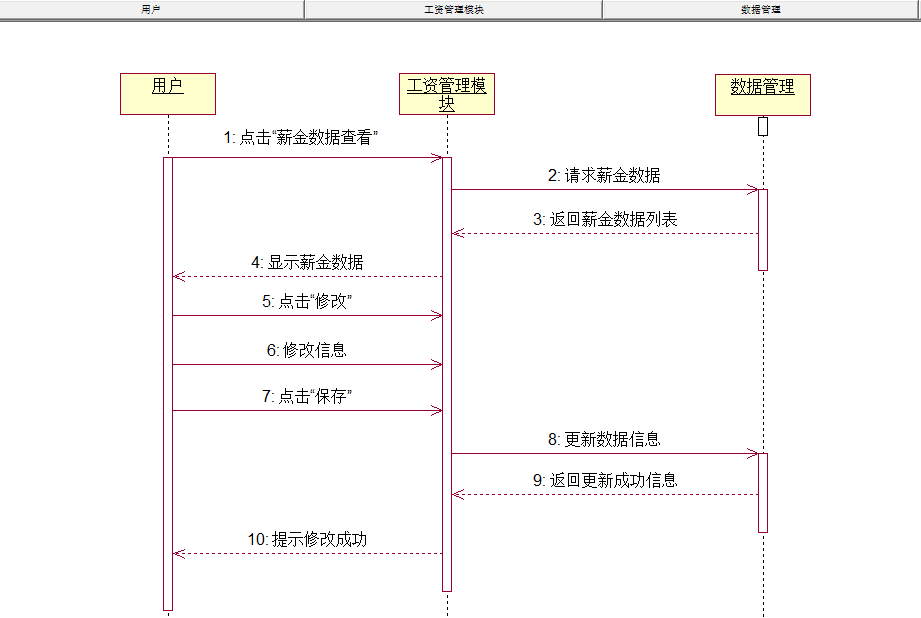


图3.7.1.2.3 薪金数据修改顺序图

表3.7.1.2.3 薪金数据修改横向设计说明

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 程序文件 | 功能说明 | 封装说明 |
| 界面 | updatestipend.jsp | 人事管理者选择“薪金管理”下拉菜单中的“薪金数据修改”，显示需要填写的文本框，最后有“提交”、“重置”按钮 | 输入项：基本薪金、全勤奖、提成、罚款 |
| 业务流程BP | StipendForm.java | 响应HTTP请求，获取前台表单数据；  用获取的表单参数向BM层对象传递消息；  成功后跳转到liststipend.jsp | reset()  validate() |
| 业务模型BM | StipendAction.java | 修改薪金数据 | updateStipend()  execute() |
| 对象持久化DAO | StipendDao.java | 为BM层提供对象持久化操作 | updateStipend ()  getCountTotalize() |

（6）对应聘信息录入事件画对象交互图为对象分配行为

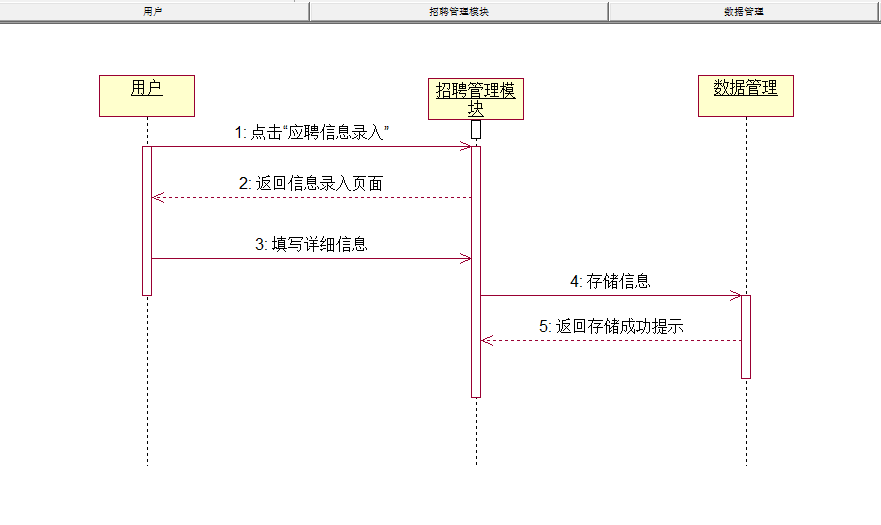
****

图3.7.1.2.4 应聘信息录入顺序图

表3.7.1.2.4 应聘信息录入横向设计说明

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 程序文件 | 功能说明 | 封装说明 |
| 界面 | addjob.jsp | 人事管理者选择“招聘管理”下拉菜单中的“招聘信息录入”，显示需要填写的文本框，最后有“提交”、“重置”按钮 | 输入项：姓名、性别、体检、年龄、职位、工作经验、所学专业、毕业学校、学历、电话、邮箱、详细经历 |
| 业务流程BP | JobForm.java | 响应HTTP请求，获取前台表单数据；  用获取的表单参数向BM层对象传递消息；  成功后跳转到listjob.jsp | reset()  validate() |
| 业务模型BM | JobAction.java | 增加应聘人员信息 | addJob()  execute() |
| 对象持久化DAO | JobDao.java | 为BM层提供对象持久化操作 | addJob () |

#### 3.7.2.3正交设计之纵向设计

（1）定义

按正交设计的纵向依MVE分割，既按对象类型的集合进行设计，也就是对各类组件进行纵向综合。

（2）按功能模块展开的设计

* 视图设计
* 逻辑设计
* 数据实体设计
* 控制器设计
* OR映射设计

（3）视图设计

<1>界面风格设计

* 要求界面风格统一，不同于一般性网页，要求简洁、清晰、操作方便。
* 避免过多的页面转换，一个操作最好集中在一个页面，但“输入”与“输出”在不同区域。
* 在不同模块显示激活的子流程，提示操作者处在哪个操作阶段。
* 树形选择区与菜单内容一致。
* 在“状态显示区”指示操作进程和状态信息。界面风格设计如下图。

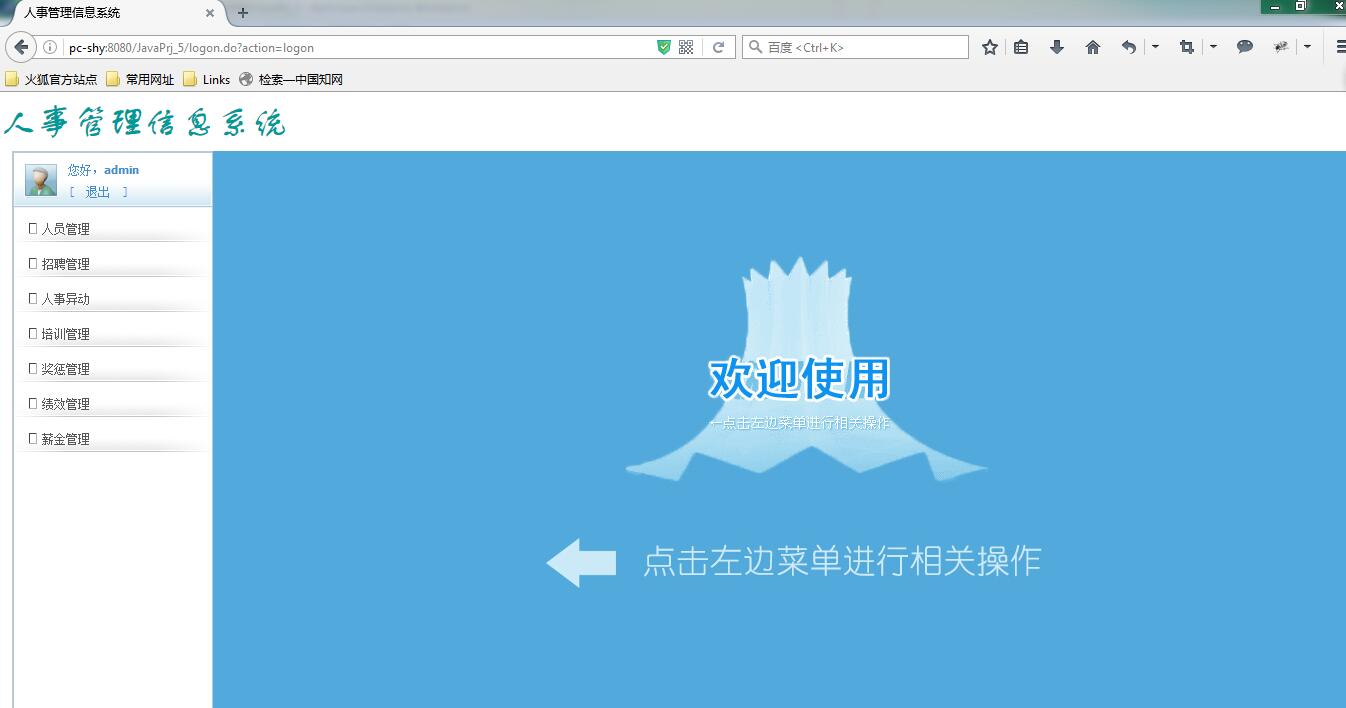


图3.7.1.2.5 项目主界面设计

具体界面设计详见4.5应用实例。

<2>命令与界面组织

表3.7.1.2.5 人事管理信息系统的命令与界面组织

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能划分 | | | | 角色参与 | 界面组件 |
| 第一层 | 第二层 | 第三层 | 底层 | 人事管理者 |  |
| 人事管理信息系统 | 招聘管理 | 应聘信息录入 | - | **√** | addjob.jsp |
| 应聘信息查看 | 详细 | **√** | listjob.jsp  detailjob.jsp |
| 删除 | **√** | listjob.jsp |
| 入 | **√** | listjob.jsp |
| 人才库浏览 | 详细 | **√** | listjob.jsp |
| 删除 | **√** | listjob.jsp |
| 人员管理 | 人员信息录入 | - | **√** | adduser.jsp |
| 人员信息查看 | 修改 | **√** | listuser.jsp  updateuser.jsp |
| 删除 | **√** | listuser.jsp |
| 绩效管理 | 绩效信息登记 | - | **√** | addperformance.jsp |
| 绩效信息查看 | 详细 | **√** | listperformance.jsp  detailperformance.jsp |
| 修改 | **√** | listperformance.jsp  updateperformance.jsp |
| 培训管理 | 培训计划录入 | - | **√** | addeducate.jsp |
| 培训计划查看 | 详细 | **√** | listeducate.jsp  detaileducation.jsp |
| 删除 | **√** | listeducate.jsp |
| 培训完成 | **√** | listeducate.jsp  updateeducate.jsp |
| 培训总结查看 | 详细 | **√** | listeducate.jsp  detaileducation.jsp |
| 删除 | **√** | listeducate.jsp |
| 奖惩管理 | 奖惩信息登记 | - | **√** | addinstitution.jsp |
| 奖惩信息查看 | 修改 | **√** | listinstitution.jsp  updateinstitution.jsp |
| 详细 | **√** | listinstitution.jsp  detailinstitution.jsp |
| 薪金管理 | 薪金数据录入 | - | **√** | addstipend.jsp |
| 薪金数据查看 | 修改 | **√** | liststipend.jsp  updatestipend.jsp |
| 删除 | **√** | liststipend.jsp |
| 人事异动 | 人事信息查看 | 详细 | **√** | listjob.jsp  detailjob.jsp |
| 删除 | **√** | listjob.jsp |
| 人事信息修改 | - | **√** | updatejob.jsp |

（4）逻辑设计

* 逻辑设计主要的工作是细化每条执行链的业务层组件和对象持久层组件；
* M＝业务流程BP+业务模型BM+数据访问DA+数据对象DO。但把数据对象DO割离出来归属到数据实体设计，因此重点考虑BP、BM和DA的设计；
* 方法是：将前一步按链设计的结果进行汇总，统一列表。然后按BP、BM、DA三层独立考虑综合问题。由于BP、BM、DA三者在横向是明确的依赖关系，对象间依赖已经解决，重点考虑是否需利用接口、继承关系提升类的设计。

（5）数据实体设计：面向对象的实体设计。由于一体化的方法本身就是沿用对象的路线，因此数据设计可以顺理成章的从对象侧进行，可以发挥面向对象的优势，方便与面向对象应用程序的接轨。该设计路线抛弃了面向数据的设计过程，从OOA分析开始用类模型逐步细化实体的数据结构。使用对象的封装、继承、关联等表达数据模型，模型设计好之后通过正向工程映射到资源层形成对应的表。

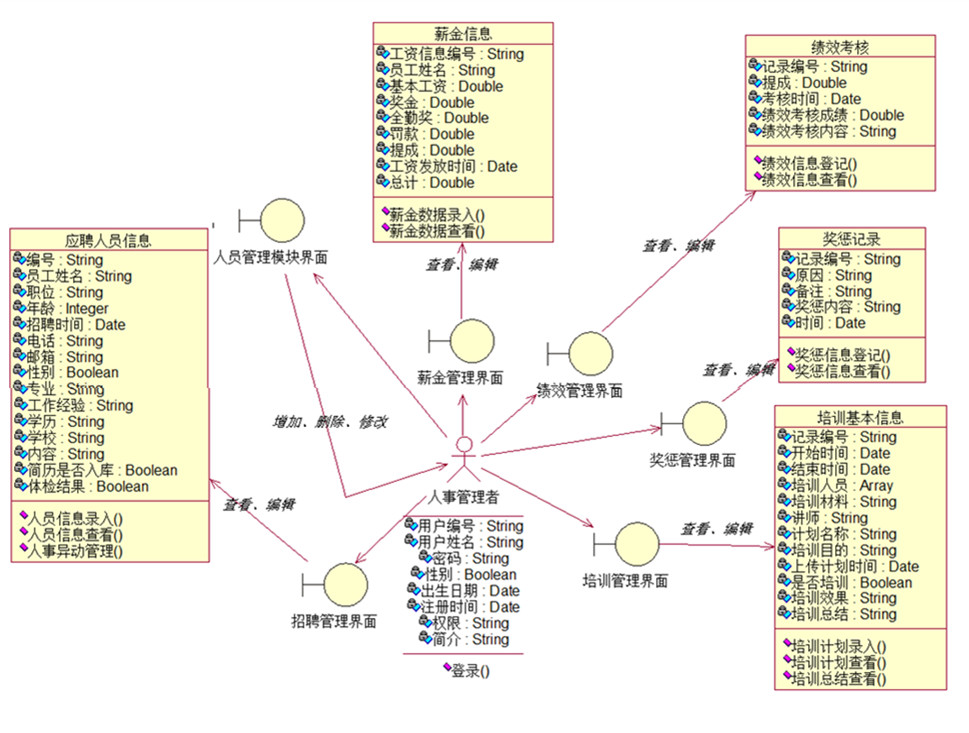


图3.7.1.2.6 实体关系设计

表3.7.1.2.6 各数据实体类内部结构设计

|  |  |  |
| --- | --- | --- |
| 实体类 | 英文名 | 作用说明 |
| 主流程 | MainFlow | 维护四个主流程开启或关闭，起始和截止日期 |
| 子流程 | SubFlow | 主流程下各子流程的开启或关闭 |
| 操作 | Operation | 命令操作既事件项的维护 |
| 人事管理者 | User | 该人事管理信息系统使用者 |
| 应聘人员信息 | Job | 前来应聘职位的人员信息 |
| 薪金信息 | Stipend | 员工薪金数据 |
| 绩效信息 | Performance | 员工绩效考核信息 |
| 奖惩记录 | Institution | 员工奖惩记录 |
| 培训基本信息 | Educate | 员工培训的基本信息以及总结 |

（6）控制器设计

控制器的设计是要找出所有执行链V与M的请求响应关系，设计出系统所有事件的请求/响应映射关系(路由表)。路由表也就是一个系统的事件汇总表。依然可以按功能模块进行，每个事件由“请求表单”出发，经过一个执行链，转向(Forward)到另一个页面。如果用Struts实现时，可根据该表配置Vregi(事件源)与BPm(响应头Action)以及Forward的关系。

* 控制器是系统的“核心机密”，由系统设计的高级技术人员掌握。
* 事件链汇总。
* 表中表达V/M事件映射关系的是“请求表单”栏与业务逻辑的“BP”栏，配置文件要依据这两栏的关系和“Forward”栏配出每个事件链。

（7）OR映射设计

* 采用Hibernate框架，依据数据实体类（O）与数据库表（R），对O和R的映射关系进行配置。

### 3.7.3系统的MVC模式设计

#### 3.7.3.1 MVC模式简介

MVC将程序分离为模型M(Model)、视图V(View)和控制器C(Controller)三部分。

视图(View)：是应用系统中与用户相关的部分，是用户看到的并能够与系统交互的操作界面，是事件的触发源头，选择系统履行的功能；也是数据展示层，完成数据的输入/输出。界面开发人员进行图形界面设计和编码。

模型(Model)：用于封装与应用程序的业务逻辑相关的数据以及对数据的处理方法。业务逻辑的处理过程对V层来说是黑箱操作，业务模型的设计可以采用广泛的技术，除一般的面向对象设计方法，可以采用前面所介绍的RMI、EJB、Spring、Hibernate、SOA、SCA等各种框架或组件。

控制器(Controller)： 在MVC模型中，V和M彼此不知道对方，由Controller连接起这两个部分，起到在V和M层间的组织作用：

负责转发V层的事件请求，找到M层处理事件的对应逻辑，就是完成事件的请求—响应的对接，它是一个事件分发器。用于控制业务流程，也就是导引完成这个事件的一系列活动，直到流程结束，再导航到相应界面显示执行结果。

#### 3.7.3.2 MVC模式分层中与框架分层

|  |  |
| --- | --- |
| **架构分层** | **MVC分层** |
| **基本三层：**客户层、业务层、数据层；可以细化为N层； | **只有三层**：视图V层、逻辑M层、控制C层 |
| **面向软件体系结构设计：**  是体系结构分层，简化软件体系结构的设计工作量，降低设计难度；  架构统一了分层的认识，规范了体系结构设计，使得三层架构约定俗成地成为了体系结构设计事实标准； | **面向领域问题的软件结构设计：**  按V、M和C三层空间隔离代码；  静动态代码分离的设计规范； |
| **提供编程组件：**  为不同层提供不同的组件，如客户层JSP、业务层Servlet、JavaBean、数据层JDBC，远程RMI； | **面向组件编程：**  对这些组件编程要按M、V、C三部分组织； |

#### 3.7.3.3 MVC模式分层中与框架分层的统一

View层=客户层

选用架构中的视图组件JSP，在视图组件中只编写静态代码；

Model层=业务层+对象持久化层=(BP+BM)+(DA+DO)

该层既囊括所有后台运算。选用架构中后台运算组件Servlet、JavaBean，在运算组件中编写动态代码；

Controller层与基本三层结构中无对应，在N层结构中就是控制层控制器C采用XML文件配置V←→M的映射，由“VM映射”的容器管理。

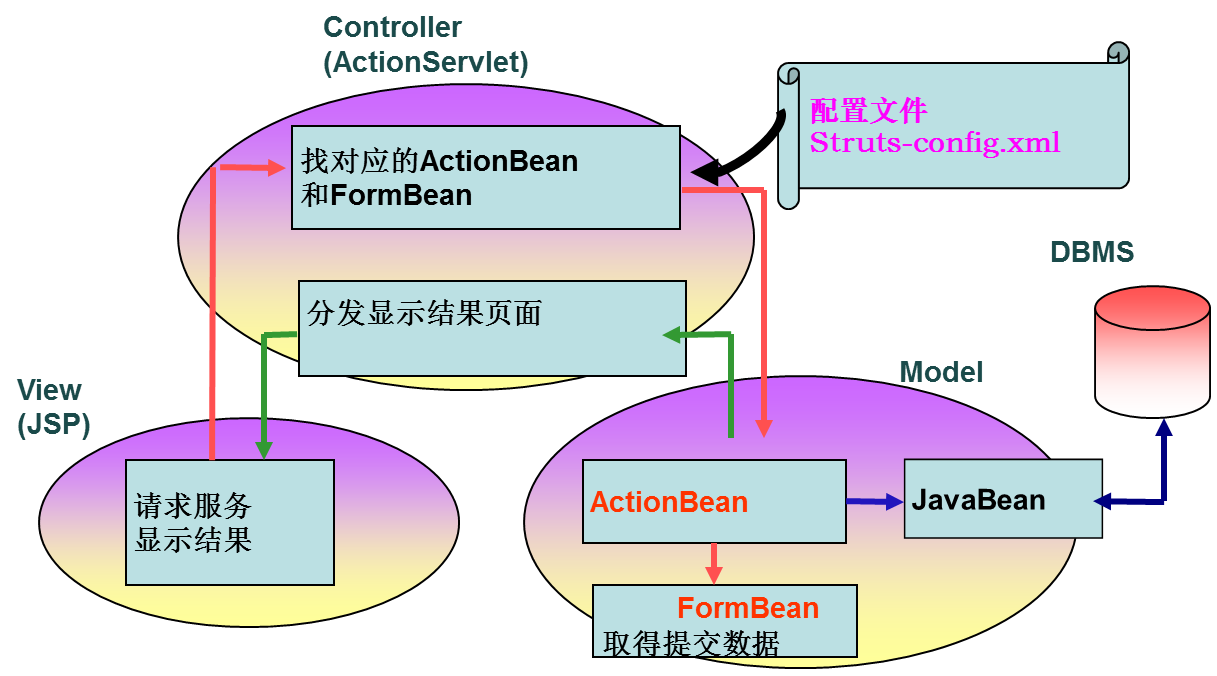
#### 3.7.3.4MVC模式Struts框架

C层：控制器C是ActionServlet，它是Servlet组件，负责事件请求和响应之间的路由（映射）关系，即事件源和执行链句柄Handle间关系，但无需对它进行编码，它根据配置文件Struts-config.xml来工作。系统设计的核心就是这个配置文件，是由开发团队的核心人物掌握。

M层：各事件执行链的Handle是Action组件，控制业务流程（因此有人将Action称为控制器），是M层中的BP层组件。在11.5.3节用Model2中取款的业务流程就移植到这里。M层的DAO由普通JavaBean构成。

V层：用JSP组件，Struts的JSP比普通JSP增加了许多Struts标签元素。另外提供一种VO组件ActionForm，ActionForm对象用于封装JSP的请求参数，该对象被ActionServlet转发给Action，Action提取ActionFrom携带的前台数据进行业务处理。ActionForm是具有特殊使命的M层组件，相当于M层派驻V层的“驻外使节”，V层通过ActionForm对象向M层递交“国书”。

#### 3.7.3.5本系统中运用Struts框架的工作流程：



人事管理信息系统具体用例--以登录为例：

Modeal:

(1)JavaBean：user.java

(2) ActionBean:UsersAction.java

(3)FormBean:UsersForm.java

View:welcome.jsp

Controller:struts-config.xml

## 3.8接口设计

### 3.8.1用户接口

1、本组人事管理信息系统采用图形方式的用户接口，提供“对话框”、“图标”、“菜单”、“按钮”等等图形用户接口。下面我们简单查看一下人事管理信息系统中导航栏具体对应的部分代码。

function tupian(idt){

var nametu="xiaotu"+idt;

var tp = document.getElementById(nametu);

tp.src="images/ico05.gif";//图片ico05为蓝色的正方形

for(var i=1;i<30;i++)

{

var nametu2="xiaotu"+i;

if(i!=idt\*1)

{

var tp2=document.getElementById('xiaotu'+i);

if(tp2!=undefined)

{tp2.src="images/ico06.gif";}//图片ico06白色的正方形

}

}

}

function list(idstr){

var name1="subtree"+idstr;

var name2="img"+idstr;

var objectobj=document.all(name1);

var imgobj=document.all(name2);

if(objectobj.style.display=="none"){

for(i=1;i<101;i++){

var name3="img"+i;

var name="subtree"+i;

var o=document.all(name);

if(o!=undefined){

o.style.display="none";

var image=document.all(name3);

//alert(image);

image.src="images/ico04.gif";

}

}

objectobj.style.display="";

imgobj.src="images/ico03.gif";

}

else{

objectobj.style.display="none";

imgobj.src="images/ico04.gif";

}

}

下图中左侧导航栏就是“菜单”这一图形用户接口。

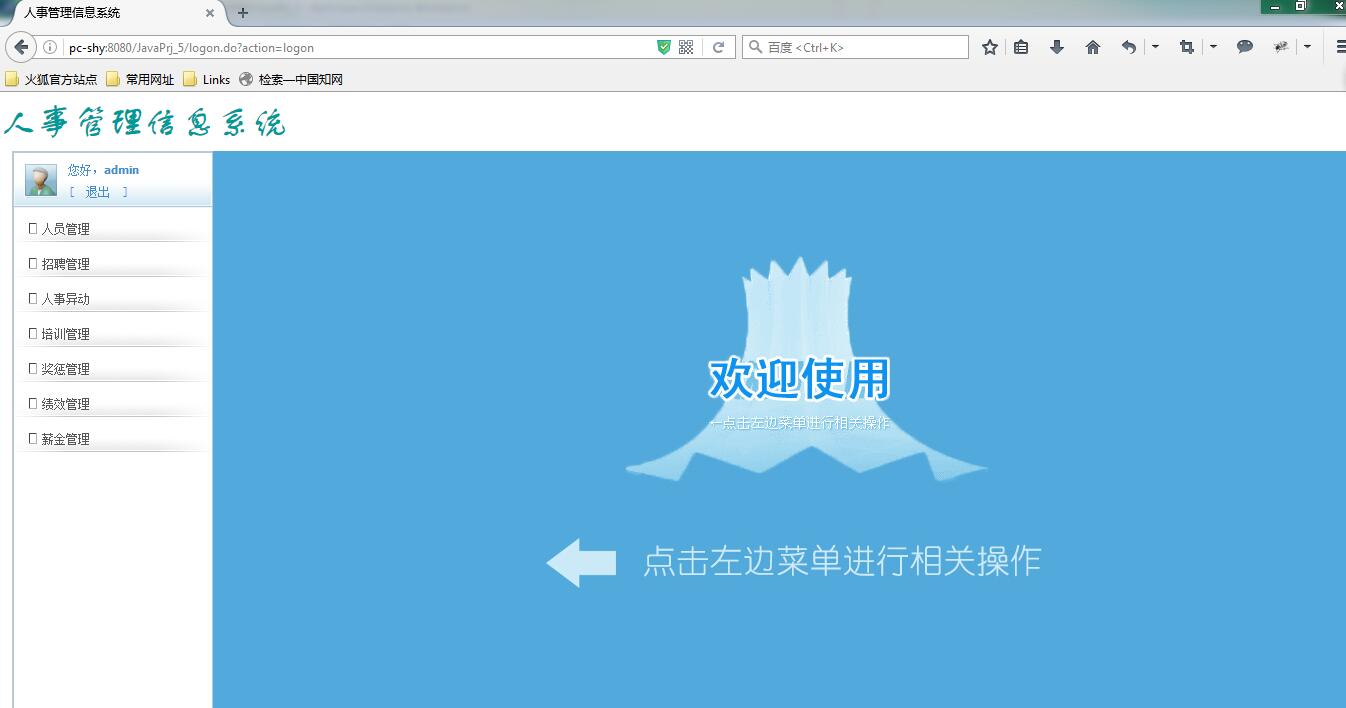


图5.1.3.1 人事管理信息系统导航栏

2、优秀的接口应当能够友好反馈错误信息给用户。下面我们查看一下错误信息反馈的部分简单代码。

function jobValidate(){

//alert(jobForm.job.value);

//return false;

if(jobForm.name.value==""){

alert("请添写姓名！");

jobForm.name.focus();

return false;

}

if(jobForm.job.value==""){

alert("请添写职位！");

jobForm.job.focus();

return false;

}

if(jobForm.specialty.value==""){

alert("请添写所学专业！");

jobForm.specialty.focus();

return false;

}

if(jobForm.experience.value==""){

alert("请添写工作经验！");

jobForm.experience.focus();

return false;

}

if(jobForm.studyeffort.value==""){

alert("请添写学历！");

jobForm.studyeffort.focus();

return false;

}

if(jobForm.school.value==""){

alert("请添写毕业学校！");

jobForm.school.focus();

return false;

}

if(jobForm.tel.value==""){

alert("请添写电话！");

jobForm.tel.focus();

return false;

}

if(isNaN(jobForm.tel.value))

{

alert("电话只能填写数字！");

jobForm.tel.focus();

return false;

}

if(jobForm.email.value==""){

alert("请添写Email！");

jobForm.email.focus();

return false;

}

var email = jobForm.email.value;

if (email != "") {

var reg = /^\w+((-\w+)|(\.\w+))\*\@[A-Za-z0-9]+((\.|-)[A-Za-z0-9]+)\*\.[A-Za-z0-9]+$/;

isok= reg.test(email );

if (!isok) {

alert("邮箱格式不正确，请重新输入！");

return false;

}

};

return true;

}

3、规范、转换用户输入结果，下面我们查看一下有关规范、转换格式的一些简单代码。

public class DateUtil {

public static final String yyMMdd="yy-MM-dd"; //短日期格式

public static final String yyyyMMdd="yyyy-MM-dd"; //长日期格式

public static final String HHmmss="HH:mm:ss"; //时间格式

public static final String yyyyMMddHHmmss

="yyyy-MM-dd HH:mm:ss"; //长日期时间格式

public static final String yyMMddHHmmss

="yy-MM-dd HH:mm:ss"; //短日期时间格式

public static Date parseToDate(String s, String style) {//字符串转换成日期时间

SimpleDateFormat simpleDateFormat

= new SimpleDateFormat(); //实例化日期格式化类

simpleDateFormat.applyPattern(style); //提交转换格式

Date date = null;

if(s==null||s.length()<8) //字符串为空或长度小于8

return null; //无法转换，返回null值

try {

date = simpleDateFormat.parse(s); //进行转换

} catch (ParseException e) {

e.printStackTrace();

}

return date;

}

public static String parseToString(String s, String style) {//格式化日期字符串

SimpleDateFormat simpleDateFormat

= new SimpleDateFormat(); //实例化日期格式化类

simpleDateFormat.applyPattern(style); //提交转换格式

Date date = null;

String str=null;

if(s==null||s.length()<8) //字符串为空或长度小于8

return null; //无法转换，返回null值

try {

date = simpleDateFormat.parse(s); //将字符串转换成日期格式

str=simpleDateFormat.format(date); //将日期格式化为字符串

} catch (ParseException e) {

e.printStackTrace();

}

return str;

}

public static String parseToString(Date date, String style) {//日期时间转换成字符串

SimpleDateFormat simpleDateFormat

= new SimpleDateFormat(); //实例化日期格式化类

simpleDateFormat.applyPattern(style); //提交转换格式

String str=null;

if(date==null) //如果日期时间为null

return null; //返回null值

str=simpleDateFormat.format(date); //将日期格式化为字符串

return str;

}

}

4、设置编码格式，下面我们查看一下有关设置编码格式的一些简单代码。

public class ActionServlet extends org.apache.struts.action.ActionServlet {

protected void process(HttpServletRequest httpservletrequest,HttpServletResponse httpservletresponse) throws IOException,ServletException {

setCharset(httpservletrequest);//设置编码

super.process(httpservletrequest, httpservletresponse);//调用父类处理方法

}

private void setCharset(HttpServletRequest httpservletrequest) {

String s = getServletConfig().getInitParameter("charset");//获得初始化参数

if (s == null || s.length() == 0) {//如果参数为空，或者为null

s="GBK"; //设置参数值为GBK

}

try {

httpservletrequest.setCharacterEncoding(s);//设置请求编码格式

} catch (UnsupportedEncodingException unsupportedencodingexception) {

log("set character encoding error", unsupportedencodingexception);

}

}

}

### 3.8.2模块接口

模块接口用于提供业务逻辑接口，将业务细分为独立单元。模块接口是模块之间进行对接交互的门户，我们在设计时至少应该遵循以下四个原则：

（1）简单原则。所谓简单，主要体现在模块接口的使用方法上，模块的使用者在不借助或借助很少的文档的情况下，就可以轻松使用模块所提供的功能。这首先要求接口方法的命名要规范，每个对外提供的方法名都应该有意义，让使用者可以通过名称猜测到方法的主要用途；其次要求接口中的相关参数的数据类型要尽可能的简单，尽量少使用嵌套层次多的数据结构，必要时可以构建全局应用的内存环境来保存模块间共同使用的数据，同时在这个内存环境之上提供不同数据的操作方法，从而减少模块间直接性的复杂数据的传递；其三，模块接口的方法尽可能单一，设计模式中的工厂模式是一种不错的选择。

（2）封闭原则。封闭原则要求的的是，模块功能的实现细节要完全对外封闭，而且在对模块内部的处理逻辑进行修改时，不会影响模块使用者的调用逻辑。

（3）完整性原则。做为功能模块，它所提供的功能应该是一个全面的整体，一些具有细微差别的功能应该被集中到一个模块中，这样我们可以方便利用继承、重载、覆写等技术手段来提高代码复用率，同时也可以提升模块使用的灵活度。

（4）可置换原则。我们很难保证一个功能模块所提供的功能会永不过时，因此在接口设计时应该尽可能的应用接口编程思想，为接口提供标准的接口规范，这样将来可以轻松的用遵循接口规范的新的模块置换原有的模块，而不会影响其到他相关模块的调用方式。

下面我们来查看一下人事管理信息系统中具体的模块接口设计的部分代码（以培训管理模块为例）

public class EducateAction extends Action {

private EducateDao dao=new EducateDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nEducateAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listeducate".equals(action)){

return listEducate(mapping,form,request,response);

}else if("addeducate".equals(action)){

return addEducate(mapping,form,request,response);

}else if("deleteeducate".equals(action)){

return deleteEducate(mapping,form,request,response);

}else if("detaileducate".equals(action)){

return detailEducate(mapping,form,request,response);

}else if("updateeducate".equals(action)){

return updateEducate(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updateEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

EducateForm educateForm=(EducateForm)form;

Educate e=educateForm.populate();

dao.updateEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Educate e = dao.loadEducate(id.longValue());

System.out.println(e.getName());

request.setAttribute("educate",e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deleteEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id").toString());

Educate e=new Educate();

e.setId(id);

dao.deleteEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

EducateForm educateForm=(EducateForm)form;

Educate e=educateForm.populate();

dao.addEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

String educate=request.getParameter("educate");

if(educate==null||"".equals(educate)||"0".equals(educate)){

request.setAttribute("list",dao.listEducate(new Byte("0").byteValue()));

}else{

request.setAttribute("list",dao.listEducate(new Byte("1").byteValue()));

}

return mapping.findForward("success");

}

}

### 3.8.3数据接口

#### 3.8.3.1 Hibernate核心接口

（1）SessionFactory接口：

SessionFactroy接口负责初始化Hibernate。它充当数据存储源的代理，并负责创建Session对象。这里用到了工厂模式。需要注意的是SessionFactory并不是轻量级的，因为一般情况下，一个项目通常只需要一个SessionFactory就够，当需要操作多个数据库时，可以为每个数据库指定一个SessionFactory。

（2）Session接口：

Session接口 Session 接口对于Hibernate 开发人员来说是一个最重要的接口。然而在Hibernate中，实例化的Session是一个轻量级的类，创建和销毁它都不会占用很多资源。这在实际项目中确实很重要，因为在客户程序中，可能会不断地创建以及销毁Session对象，如果Session 的开销太大，会给系统带来不良影响。但是Session对象是非线程安全的，因此在你的设计中，最好是一个线程只创建一个Session对象。 session可以看作介于数据连接与事务管理一种中间接口。我们可以将session想象成一个持久对象的缓冲区，Hibernate能检测到这些持久对象的改变，并及时刷新数据库。我们有时也称Session是一个持久层管理器，因为它包含这一些持久层相关的操作， 诸如存储持久对象至数据库，以及从数据库从获得它们。需要注意的是，Hibernate的session不同于JSP 应用中的HttpSession。当我们使用session这个术语时，我们指的Hibernate 中的session，而我们以后会将HttpSesion 对象称为用户session。

（3）Transaction接口

Transaction接口负责事务相关的操作，一般在Hibernate的增删改中出现，但是使用Hibernate的人一般使用Spring去管理事务。

（4）Query接口

Query负责执行各种数据库查询。它可以使用HQL语言或SQL语句两种表达方式。它的返回值一般是List。需要自己转换。

（5）Configuration接口：

Configuration对象用于配置并根启动Hibernate。Hibernate应用通过Configuration实例来指定对象—关系映射文件的位置或者动态配置Hibernate的属性，然后创建SessionFactory实例。

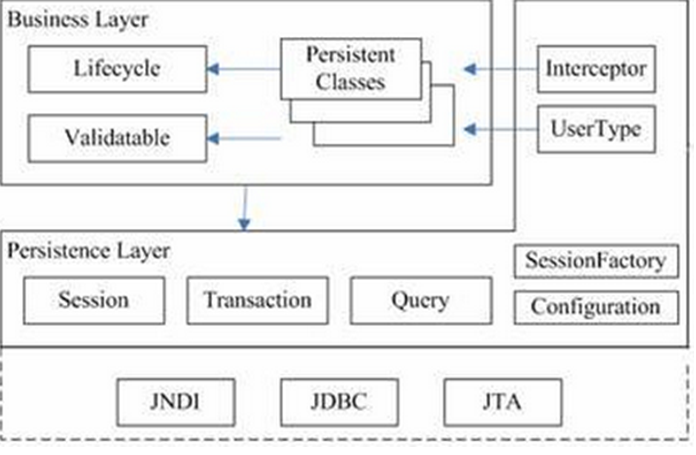


图5.1.3.2 Hibernate核心接口的关系图

下面我们查看一下人事管理信息系统源码中具体应用的SessionFactory接口（主要以培训管理模块为例）。

HibernateSessionFactory.java：

private static String CONFIG\_FILE\_LOCATION = "/hibernate.cfg.xml";

private static final ThreadLocal<Session> threadLocal = new ThreadLocal<Session>();

private static Configuration configuration = new Configuration();

private static org.hibernate.SessionFactory sessionFactory;

private static String configFile = CONFIG\_FILE\_LOCATION;

static {

try {

configuration.configure(configFile);

sessionFactory = configuration.buildSessionFactory();

} catch (Exception e) {

System.err.println("%%%% Error Creating SessionFactory %%%%");

e.printStackTrace();

}

}

private HibernateSessionFactory() {

}

public static Session getSession() throws HibernateException {

Session session = (Session) threadLocal.get();

if (session == null || !session.isOpen()) {

if (sessionFactory == null) {

rebuildSessionFactory();

}

session = (sessionFactory != null) ? sessionFactory.openSession(): null;

threadLocal.set(session);

}

return session;

}

public static void rebuildSessionFactory() {

try {

configuration.configure(configFile);

sessionFactory = configuration.buildSessionFactory();

} catch (Exception e) {

System.err.println("%%%% Error Creating SessionFactory %%%%");

e.printStackTrace();

}

}

public static void closeSession() throws HibernateException {

Session session = (Session) threadLocal.get();

threadLocal.set(null);

if (session != null) {

session.close();

}

}

public static org.hibernate.SessionFactory getSessionFactory() {

return sessionFactory;

}

public static void setConfigFile(String configFile) {

HibernateSessionFactory.configFile = configFile;

sessionFactory = null;

}

public static Configuration getConfiguration() {

return configuration;

}

下面我们查看一下人事管理信息系统源码中具体应用的其他接口（主要以培训管理模块为例）。

public void addEducate(Educate e) throws HibernateException{

e.setCreatetime(new java.util.Date());

e.setEducate(new Byte("0"));

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.save(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public void deleteEducate(Educate e) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.delete(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public Educate loadEducate(long id) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Educate e=(Educate)session.get(Educate.class,new Long(id));

tx.commit();

HibernateSessionFactory.closeSession();

return e;

}

public List listEducate(byte educate) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Query query = session.createQuery("select e from Educate as e where e.educate =:educate order by createtime");

query.setByte("educate",educate);

List list = query.list();

tx.commit();

HibernateSessionFactory.closeSession();

return list;

}

public void updateEducate(Educate educate) throws HibernateException{

Educate e =this.loadEducate(educate.getId().longValue());

if (educate.getBegintime()!=null){

e.setBegintime(educate.getBegintime());

}

if(educate.getDatum()!=null){

e.setDatum(educate.getDatum());

}

if (educate.getEffect()!= null) {

e.setEffect(educate.getEffect());

}

if (educate.getEndtime()!= null) {

e.setEndtime(educate.getEndtime());

}

if(educate.getName()!=null){

e.setName(educate.getName());

}

if(educate.getPurpose()!=null){

e.setPurpose(educate.getPurpose());

}

if(educate.getStudent()!=null){

e.setStudent(educate.getStudent());

}

if(educate.getSummarize()!=null){

e.setSummarize(educate.getSummarize());

}

if(educate.getEducate()!=null){

e.setEducate(educate.getEducate());

}

if(educate.getTeacher()!=null){

e.setTeacher(educate.getTeacher());

}

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.update(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

#### 3.8.3.2 DAO接口

DAO(Data Access Object)是一个数据访问接口，数据访问，顾名思义就是与数据库打交道。所以说，DAO接口是夹在业务逻辑与数据库资源中间的。为了建立一个健壮的J2EE应用，应该将所有对数据源的访问操作抽象封装在一个公共API中。用程序设计的语言来说，就是建立一个接口，接口中定义了此应用程序中将会用到的所有事务方法。在这个应用程序中，当需要和数据源进行交互的时候则使用这个接口，并且编写一个单独的类来实现这个接口在逻辑上对应这个特定的数据存储。

传统关系型数据库定义了四种数据操作：

插入 Insert

删除 Delete

更新 Update

查询 Query

可以说，这四种操作涵盖了所有的数据操作。并且，除了插入操作，所有的操作都是可以一次针对多条记录的。但是，其实从使用者的角度来看，这四种操作还是有所不同的。比如，查询返回的结果，很多时候仅仅是一条记录。 我们需要为这种情况进行优化。所以，DAO在传统关系型数据库数据操作的基础上定义了如下的数据操作：

表5.1.3.1 DAO数据操作

|  |  |  |
| --- | --- | --- |
| 插入 | Insert | 一条 SQL 插入一条记录或者多条记录 |
| 插入 | FastInsert | 一条 SQL ,通过batch插入多条记录 |
| 删除 | Delete | 一条 SQL 删除一条记录 |
| 更新 | Update | 一条 SQL 更新一条或者多条记录 |
| 获取 | Fetch | 一条 SQL 获取一条记录 |
| 查询 | Query | 一条 SQL 根据条件获取多条记录 |
| 清除 | Clear | 一条 SQL 根据条件删除多条记录 |
| 建表 | Create | 根据实体建表 |
| 删表 | Drop | 根据实体/表名称进行删表 |
| 聚合 | Func | 执行sum,count等操作 |

下面我们查看一下人事管理信息系统源码中具体应用的其他接口（主要以培训管理模块为例）。

public class EducateAction extends Action {

private EducateDao dao=new EducateDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nEducateAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listeducate".equals(action)){

return listEducate(mapping,form,request,response);

}else if("addeducate".equals(action)){

return addEducate(mapping,form,request,response);

}else if("deleteeducate".equals(action)){

return deleteEducate(mapping,form,request,response);

}else if("detaileducate".equals(action)){

return detailEducate(mapping,form,request,response);

}else if("updateeducate".equals(action)){

return updateEducate(mapping,form,request,response);

}

return mapping.findForward("error");

}

private ActionForward updateEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

EducateForm educateForm=(EducateForm)form;

Educate e=educateForm.populate();

dao.updateEducate(e);

return mapping.findForward("success");

}

private ActionForward detailEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Educate e = dao.loadEducate(id.longValue());

System.out.println(e.getName());

request.setAttribute("educate",e);

return mapping.findForward("success");

}

private ActionForward deleteEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id").toString());

Educate e=new Educate();

e.setId(id);

dao.deleteEducate(e);

return mapping.findForward("success");

}

private ActionForward addEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

EducateForm educateForm=(EducateForm)form;

Educate e=educateForm.populate();

dao.addEducate(e);

return mapping.findForward("success");

}

private ActionForward listEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

String educate=request.getParameter("educate");

if(educate==null||"".equals(educate)||"0".equals(educate)){

request.setAttribute("list",dao.listEducate(new Byte("0").byteValue()));

}else{

request.setAttribute("list",dao.listEducate(new Byte("1").byteValue()));

}

return mapping.findForward("success");

}

}

### 3.8.4总结

所有的接口，其实基本目的就是降低耦合度，以屏蔽具体的底层实现。当底层实现变动时，可以不用修改上层Service的实现，就可以很好的兼容底层的变动。总之一句话，接口对外封装过程，对内提供统一标准。

## 3.9容错设计

### 3.9.1出错信息

用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。

表5.1 出错信息及处理表

|  |  |  |
| --- | --- | --- |
| **序号** | **出错或故障情况** | **系统输出信息及处理** |
| 1 | 输入用户名和密码等必填信息为空 | 客户端拦截用户的请求，并弹出错误提示，提醒用户需输入必填信息。 |
| 2 | 输入密码输入中文 | 客户端通过正则表达式判断用户输入值的合法性，并拦截用户请求，弹出错误提示，提醒用户输入正确值。 |
| 3 | 输入工资等信息输入非数字的值 | 客户端通过正则表达式判断用户输入值的合法性，并拦截用户请求，弹出错误提示，提醒用户输入正确值。 |
| 4 | 输入邮箱地址格式不正确 | 客户端通过正则表达式判断用户输入的邮箱地址格式是否正确，若错误，拦截用户请求，弹出错误提示，提醒用户输入正确值。 |

### 3.9.2补救措施

系统通过正则表达式判断各种用户输入值的合法性。正则表达式，又称规则表达式。（英语：Regular Expression，在代码中常简写为regex、regexp或RE），计算机科学的一个概念，通常被用来检索、替换那些符合某个模式(规则)的文本。

首先，系统通过客户端获取到用户输入的值，再应用正则表达式对用户输入的值进行匹配处理，检查其中是否包含中英文、数字、格式是否正确等信息。若输入值不合法，客户端对用户进行请求的拦截和对用户的提醒，提示用户重新输入正确的值。此种举措可以防止不合法的值被传到后端，造成错误。

# 4.面向对象实现OOI

## 4.1 OOI概述

### 4.1.1面向对象实现

面向对象实现即是将系统设计阶段的结果在计算机上实现，并应用到实际管理工作之中的过程。即将纸面上的、类似于设计图式的新的管理信息系统方案（物理模型）转成可以实际运行的管理信息系统系统软件，并应用到实际管理工作之中。[[15]](#footnote-15)

### 4.1.2系统实现的内容

根据2.2.2功能模型，我们的系统实现了人员管理、招聘管理、绩效管理、培训管理、奖惩管理、人事异动、薪金管理七个主要模块的功能。其中，人员管理模块包含人员信息录入、人员信息查看两个子模块；招聘管理模块包含应聘信息录入、应聘信息查看、人才库浏览三个子模块；人事异动模块包含人事异动管理一个子模块；培训管理模块包含培训计划录入、培训计划查看、培训总结查看三个子模块；绩效管理模块包含绩效信息登记、绩效信息查看两个子模块；奖惩管理模块包含奖惩信息登记、奖惩信息查看两个子模块；薪金管理模块包含薪金数据录入、薪金数据查看两个子模块。下面将分别介绍七个主要功能模块及其子模块的实现内容。

4.1.2.1人员管理模块

点击左边菜单“人员管理”会出现“人员信息录入”和“人员信息查看”下拉菜单。

4.1.2.1.1人员信息录入页面

点击“人员信息录入”即出现“人员信息录入”页面。在该页面，用户可输入管理员相关信息，其中用户名、登录密码、出生日期为必填项目，不填或填错将在保存时有出错提示并不能保存，而性别、是否管理员及人员简介等信息为可选项目。填写完毕后，单击“保存”按钮则将输入的所有信息保存至数据库中，同时跳转至“人员信息查看”页面；单击“返回”按钮则返回上一个页面；点击“重置”按钮将清除所有填写信息。

4.1.2.1.2人员信息查看页面

该页面向用户展示了所有管理员的详细信息如用户名、密码、性别、出生日期、是否管理员、简介。可执行操作有修改和删除：点击“修改”将跳转至“人员信息修改”页面修改数据，在该页面点击更新按钮后会保存数据至数据库并跳转回“人员信息查看”页面，点击“重置”按钮可还原修改前的信息；点击“删除”将从数据库删除所选管理员信息，同时该管理员将不能登录系统。

4.1.2.2招聘管理模块

点击左边菜单“招聘管理”会出现“应聘信息录入”、“应聘信息查看”、“人才库浏览”下拉菜单。

4.1.2.2.1应聘信息录入页面

点击“应聘信息录入”即出现“应聘信息录入”页面。在该页面，用户可输入应聘者相关信息，其中姓名、职位、所学专业、学历、Email、工作经验、毕业学校、电话为必填项目，不填或填错将在保存时有出错提示并不能保存，而性别、体检、年龄及详细信息等信息为可选项目。填写完毕后，单击“保存”按钮则将输入的所有信息保存至数据库中，同时跳转至“应聘信息查看”页面；单击“返回”按钮则返回上一个页面；点击“重置”按钮将清除所有填写信息。

4.1.2.2.2应聘信息查看页面

该页面向用户展示了所有应聘者的详细信息如姓名、性别、体检、年龄、职位、所学专业、工作经验。可执行操作有“详细”、“删除”和“入库”：点击“详细”将跳转至“人才详细信息”页面查看该应聘者的详细信息，在该页面点击“修改”按钮后将跳转至“修改人事信息”页面，点击“保存”按钮可以保存数据至数据库并跳转回“应聘信息查看”页面，点击“重置”按钮可还原修改前的信息，点击“返回”将返回至“人才详细信息”页面；点击“删除”将从数据库删除所选应聘者信息；点击“入库”将招收该应聘者，其数据将不再显示在“应聘信息查看”页面，而显示在“人才库浏览”页面中。

4.1.2.2.3人才库浏览页面

该页面向用户展示了所有员工的详细信息如姓名、性别、体检、年龄、职位、所学专业、工作经验。可执行操作有“详细”和“删除”：点击“详细”将跳转至“人才详细信息”页面查看该员工的详细信息，在该页面点击“修改”按钮后将跳转至“修改人事信息”页面，点击“保存”按钮可以保存数据至数据库并跳转回“人才库浏览”页面，点击“重置”按钮可还原修改前的信息，点击“返回”将返回至“人才详细信息”页面；点击“删除”将从数据库删除所选员工信息。

4.1.2.3人事异动模块

4.1.2.3.1人事信息查看页面

该页面向用户展示了所有员工的详细信息如姓名、性别、体检、年龄、职位、所学专业、工作经验。可执行操作有“详细”和“删除”，在“详细”中可改变员工职位。点击“详细”将跳转至“人才详细信息”页面查看该员工的详细信息，在该页面点击“修改”按钮后将跳转至“修改人事信息”页面，在该页面，只能修改职位信息，点击“保存”按钮可以保存数据至数据库并跳转回“人才信息查看”页面，点击“重置”按钮可还原修改前的信息，点击“返回”将返回至“人才详细信息”页面；点击“删除”将从数据库删除所选员工信息。

4.1.2.4培训管理模块

点击左边菜单“培训管理”会出现“培训计划录入”、“培训计划查看”和“培训总结查看”下拉菜单。

4.1.2.4.1培训计划录入页面

点击“培训计划录入”即出现“培训计划录入”页面。在该页面，用户可输入培训计划相关信息，其中培训名称、培训目的、培训开始时间、培训结束时间、讲师、培训人员为必填项目，不填或填错将在保存时有出错提示并不能保存，而培训材料等信息为可选项目。填写完毕后，单击保存按钮则将输入的所有信息保存至数据库中，同时跳转至“培训计划查看”页面；单击“返回”按钮则返回上一个页面；点击“重置”按钮可清除所有已写信息。。

4.1.2.4.2培训计划查看页面

该页面向用户展示了所有培训计划信息如培训名称、讲师、培训人员、开始时间、结束时间等。可执行操作有“详细”、“删除”和“培训完成”：点击“详细”将跳转至“培训详细计划”页面查看该培训计划的详细信息；点击删除将从数据库删除所选计划相关信息；点击“培训完成”将跳转至“培训总结”页面，填写培训效果及培训总结，点击“保存”按钮可以保存数据至数据库并跳转至“培训总结查看”页面，点击“重置”按钮可还原修改前的信息，点击“返回”将返回至“培训计划查看”页面。

4.1.2.4.3培训总结查看页面

该页面向用户展示了所有培训计划信息如培训名称、讲师、培训人员、开始时间、结束时间等。可执行操作有“详细”和“删除”：点击“详细”将跳转至“培训详细总结”页面查看该培训总结的详细信息，在该页面点击“修改”按钮后将跳转至“修改培训总结”页面，修改完毕后点击“保存”按钮可以保存数据至数据库并跳转回“培训总结查看”页面，点击“重置”按钮可还原修改前的信息，点击“返回”将返回至“培训总结查看”页面；点击“删除”将从数据库删除所选培训总结相关信息。

4.1.2.5奖惩管理模块

点击左边菜单“奖惩管理”会出现“奖惩信息登记”和“奖惩信息查看”下拉菜单。

4.1.2.5.1奖惩信息登记

点击“奖惩信息登记”即出现“奖惩信息登记”页面。在该页面，用户可输入奖励措施或惩罚措施及其相关信息，其中奖惩名称、奖惩原因为必填项目，不填或填错将在保存时有出错提示并不能保存，而奖惩说明等信息为可选项目。填写完毕后，单击“保存”按钮则将输入的所有信息保存至数据库中，同时跳转至“奖惩信息查看”页面；单击“返回”按钮则返回上一个页面；点击“重置”按钮可清除所有已写信息。

4.1.2.5.2奖惩信息查看

该页面向用户展示了所有奖惩相关信息：奖惩名称、奖惩原因。可执行操作有“详细”和“删除”：点击“详细”将跳转至“奖惩信息详情”页面查看此次奖惩措施的详细信息，在该页面点击“修改”按钮后将跳转至“修改奖惩信息”页面，修改完毕后点击“保存”按钮可以保存数据至数据库并跳转回“奖惩信息查看”页面，点击“重置”按钮可还原修改前的信息，点击“返回”将返回至“奖惩信息查询”页面；点击“删除”将从数据库删除所选奖惩措施相关信息。

4.1.2.6绩效管理模块

点击左边菜单“绩效管理”会出现“绩效信息登记”和“绩效信息查看”下拉菜单。

4.1.2.6.1绩效信息登记

点击“绩效信息登记”即出现“绩效信息登记”页面。在该页面，用户可输入奖励措施或惩罚措施及其相关信息，其中绩效分数、增加工资为必填项目，不填或填错将在保存时有出错提示并不能保存，而创建时间、绩效说明等信息为可选项目，其中创建时间为系统当前时间。填写完毕后，单击“保存”按钮则将输入的所有信息保存至数据库中，同时跳转至“绩效信息查看”页面；单击“返回”按钮则返回上一个页面；点击“重置”按钮可清除所有已写信息。

4.1.2.6.2绩效信息查看

该页面向用户展示了所有绩效相关信息：绩效分数、增加工资、创建时间、绩效说明。可执行操作有“详细”和“删除”：点击“详细”将跳转至“绩效信息详情”页面查看该员工绩效方面的详细信息，在该页面点击“修改”按钮后将跳转至“修改绩效信息”页面，其中创建时间不能修改，修改完毕后点击“保存”按钮可以保存数据至数据库并跳转回“绩效信息查看”页面，点击“重置”按钮可还原修改前的信息，点击“返回”将返回至“绩效信息查询”页面；点击“删除”将从数据库删除该条绩效相关信息。

4.1.2.7 薪金管理模块

点击左边菜单“薪金管理”会出现“薪金数据录入”和“薪金数据查看”下拉菜单。

4.1.2.7.1薪金数据录入

点击“薪金数据录入”即出现“薪金数据录入”页面。在该页面，用户可输入员工薪金相关信息，其中员工姓名、基本薪金和发放时间为必填项目，不填或填错将在保存时有出错提示并不能保存，而全勤奖、提成、罚款等信息为可选项目。填写完毕后，单击“保存”按钮则将输入的所有信息保存至数据库中，同时跳转至“薪金数据查看”页面；单击“返回”按钮则返回上一个页面；点击“重置”按钮可清除所有已写信息。

4.1.2.7.2薪金数据查看

该页面向用户展示了所有薪金相关信息：员工姓名、基本薪金、全勤奖、提成、罚款、发放时间、总计。可执行操作有“详细”和“删除”：点击“详细”将跳转至“薪金数据详情”页面查看该员工薪金方面的详细信息，在该页面点击“修改”按钮后将跳转至“薪金数据更新”页面，修改完毕后点击“保存”按钮可以保存数据至数据库并跳转回“薪金数据查看”页面，点击“重置”按钮可还原修改前的信息，点击“返回”将返回至“薪金数据查询”页面；点击“删除”将从数据库删除该条薪金相关信息。

### 4.1.3系统角色分析

角色或执行者指与系统产生交互的外部用户或者外部系统。考虑到麻麻鱼府企业的实际情况，因该企业普通员工多不具有使用电脑、手机等设备的条件和能力，本系统用户角色唯一，系统仅对人事管理者开放。但系统内的单一角色下可包含多个用户。

**角色功能描述如下：**

人事管理者：负责员工信息的录入、修改等工作，同时还能对员工的薪金、奖惩、绩效、培训等信息进行录入和修改。

## 4.2系统初始化

（1）用户数据初始化：向系统中录入初始用户的信息，包括账号、密码等，保证至少有一个用户可以登录系统；

（2）员工数据初始化：向系统中录入模拟员工数据，包括各个员工的薪金、培训等状态信息，满足系统测试的最低要求。

## 4.3关键业务逻辑|困难算法实现[[16]](#footnote-16)

### 4.3.1解决查看不同人员详情的问题

解决方法：获取所查看人员的id，通过id辨识是哪个人员，查询其相关数据。

步骤一：获取所查看的员工的id。

步骤二：后台根据id查询员工信息。

步骤三：将信息发至前端，填充至相应的地方。

listjob.jsp: <a href="detailjob.do?action=detailjob&id=<%=j.getId()%>">详细</a>

JobAction.java:

private ActionForward detailjob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//获得参数id

//System.out.println("id="+id);

Job j=dao.loadJob(id.longValue());//加载该ID对应的应聘信息

request.setAttribute("job",j);//将应聘信息对象设置到request范围

return mapping.findForward("success");

}

detailjob.jsp: <%

Job j=(Job)request.getAttribute("job");

if(j!=null){

%>

<tr>

<td width="100%">

<fieldset style="height:100%;">

<table border="0" cellpadding="8" cellspacing="1" style="width:100%">

<tr>

<td align="right" width="9%">姓&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;名：</td>

<td width="36%"><%=j.getName()%></td>

<td width="20%"><div align="right">性&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;别：</div></td>

<td width="43%"><%=new Byte("1").equals(j.getSex())?"男":"女"%> </tr>

<tr>

<td width="20%"><div align="right">体&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;检：</div></td>

<td width="43%"><%=new Byte("1").equals(j.getHealth())?"通过":"未通过"%>

<td align="right" width="9%">年&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;龄：</td>

<td><%=j.getAge()%></td>

</td>

</tr>

<tr>

<td><div align="right">职&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;位：</div></td>

<td><%=j.getJob()%>

<td><div align="right">工作经验：</div></td>

<td><%=j.getExperience()%></td>

</td>

</tr>

<tr>

<td align="right">所学专业：</td>

<td><%=j.getSpecialty()%>

<td><div align="right">毕业学校：</div></td>

<td><%=j.getSchool()%></td>

</tr>

<tr>

<td align="right">学&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;历：</td>

<td><%=j.getStudyeffort()%>

</td>

<td align="right">电&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;话：</td>

<td><%=j.getTel()%>

</td>

</tr>

<tr>

<td><div align="right">Email：</div></td>

<td><%=j.getEmail()%></td>

<td align="right">详细经历：</td>

<td colspan="4"><%=j.getContent()%></td>

</tr>

<tr>

<td colspan="4" align="center">

<a href="updatejob.do?action=detailjob&id=<%=j.getId() %>">修改</a>&nbsp;&nbsp;

<a href="#" onclick="javascript:history.back();">返回</a></td>

</tr>

</table>

</fieldset>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="4" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

</table>



图4.2.1.1 未输入数据的人才库列表



图4.2.1.2 人才库列表



图4.2.1.3 人才库列表详细信息

### 4.3.2解决动态向用户展示数据的问题

解决方法：在页面加载时使用Hibernate框架查询数据库的相关数据，并填充进界面。

步骤一：调用相应方法，向数据库查询数据。

步骤二：将查询到的数据封装好，发至前端。

步骤三：JSP页面获取到数据后，填充进特定的地方，展示给用户。

JobAction.java:

private ActionForward listJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

String isstock=request.getParameter("isstock");//获得isstock参数

System.out.println("isstock="+isstock);

if(isstock==null||"".equals(isstock)){//如果isstock为null或者为空

request.setAttribute("list",

dao.listJob(new Byte("0").byteValue()));//查询所有没入库的应聘信息

}else{

request.setAttribute("list",

dao.listJob(new Byte("1").byteValue()));//查询所有入库的应聘信息

}

return mapping.findForward("success");

}

}

listjob.jsp:

<table width="100%" border="0" cellpadding="4" cellspacing="1" bgcolor="#464646" class="newfont03">

<tr class="CTitle" >

<td height="22" colspan="8" align="center" style="font-size:16px">人才信息列表</td>

</tr>

<tr bgcolor="#EEEEEE">

<td width="4%" align="center" height="30">姓名</td>

<td width="10%" align="center" >性别</td>

<td width="10%" align="center" >体检</td>

<td width="10%" align="center" >年龄</td>

<td width="10%" align="center" >职位</td>

<td width="10%" align="center" >所学专业</td>

<td width="15%" align="center" >工作经验</td>

<td width="12%" align="center" >执行操作</td>

</tr>

<% List list=(List)request.getAttribute("list");

if(list!=null&&list.size()>0){

Iterator it = list.iterator();

while (it.hasNext()) {

Job j = (Job) it.next();

%>

<%if(j.getIsstock()!=null&&j.getIsstock().intValue()==0){%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=j.getName()%></td>

<td height="22" align="center" ><%=new Byte("1").equals(j.getSex())?"男":"女"%></td>

<td height="22" align="center" ><%=new Byte("1").equals(j.getHealth())?"通过":"未通过"%></td>

<td height="22" align="center" ><%=j.getAge()%></td>

<td height="22" align="center" ><%=j.getJob()%></td>

<td height="22" align="center" ><%=j.getSpecialty()%></td>

<td height="22" align="center" ><%=j.getExperience()%></td>

<td height="22" align="center" ><a href="detailjob.do?action=detailjob&id=<%=j.getId()%>">详细</a>&nbsp;&nbsp;<a href="modifyjob.do?action=deletejob&&id=<%=j.getId()%>">删除</a>

<a href="modifyjob.do?action=updatejob&id=<%=j.getId()%>&isstock=1">入库</a></td>

<% }else{%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=j.getName()%></td>

<td height="22" align="center" ><%=new Byte("1").equals(j.getSex())?"男":"女"%></td>

<td height="22" align="center" ><%=new Byte("1").equals(j.getHealth())?"通过":"未通过"%></td>

<td height="22" align="center" ><%=j.getAge()%></td>

<td height="22" align="center" ><%=j.getJob()%></td>

<td height="22" align="center" ><%=j.getSpecialty()%></td>

<td height="22" align="center" ><%=j.getExperience()%></td>

<td height="22" align="center" ><a href="detailjob.do?action=detailjob&id=<%=j.getId()%>">详细</a>&nbsp;&nbsp;<a href="modifyjob.do?action=deletejob&isstock=1&id=<%=j.getId()%>">删除</a>

<%}%>

</tr>

<%}

}else{

%>

<tr bgcolor="#FFFFFF">

<td height="22" colspan="8" align="center" >对不起，没有信息！！！</td>

</tr>

<%}%>

</table>



图4.2.2.1 未输入数据的人才库列表



图4.2.2.2 人才库列表



图4.2.2.3 人才库列表详细信息

### 4.3.3解决动态数据存储问题

解决方法：使用form表单提交的方式，把用户输入的数据传至后台，后台再将相应数据存储入数据库。

步骤一：前端判断用户输入值的合法性后，将数据传至后台。

步骤二：后台获取数据，调用相应的dao进行数据库的增删改查。

步骤三：若操作成功，跳转至成功页面。

updateeducate.jsp:

<form name="educateForm" method="post" action="modifyeducate.do?action=updateeducate" onSubmit="return educateValidate();">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >培训总结</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<%

Educate e=(Educate)request.getAttribute("educate");

if(e!=null){

%>

<input type="hidden" name="id" value="<%=e.getId()%>"/>

<input type="hidden" name="educate" value="<%=request.getParameter("educate")%>"/>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>培训计划</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">培训名称：</td>

<td colspan="3"><input name="name" type="text" class="input" value="<%=e.getName()%>" disabled/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">培训目的：</td>

<td colspan="3"><input type="text" name="purpose" class="input" value="<%=e.getPurpose()%>" disabled/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">培训开始时间：</td>

<td width="29%"><input type="text" name="begintime" class="input" value="<%=StringUtil.notNull(DateUtil.parseToString(e.getBegintime(),DateUtil.yyyyMMdd))%>" disabled/>

<span class="red">\*</span></td>

<td width="18%">培训结束时间：</td>

<td width="42%"><input type="text" name="endtime" class="input" value="<%=StringUtil.notNull(DateUtil.parseToString(e.getEndtime(),DateUtil.yyyyMMdd))%>" disabled/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">讲师：</td>

<td><input type="text" name="teacher" class="input" value="<%=e.getTeacher()%>" disabled/>

<span class="red">\*</span></td>

<td>培训人员：</td>

<td><input type="text" name="student" class="input" value="<%=e.getStudent()%>" disabled/>

<span class="red">\*</span>

</td>

</tr>

<tr>

<td width="11%" nowrap align="right">培训材料：</td>

<td colspan="3"><textarea name="datum" cols="100" rows="6" class="input" disabled><%=e.getDatum()%></textarea></td>

</tr>

<tr align="center">

<td width="11%" height="22" >培训效果：</td>

<td height="22" colspan="3" align="left" ><input name="effect" type="text" class="input" value="<%=StringUtil.notNull(e.getEffect())%>" size="50"></td>

</tr>

<tr align="center">

<td width="11%" height="22" >培训总结：</td>

<td height="22" colspan="3" align="left" ><textarea name="summarize" cols="50" rows="6" class="input"><%=StringUtil.notNull(e.getSummarize())%></textarea></td>

</tr>

</table>

<br />

</fieldset> </TD>

</TR>

</TABLE>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="4" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

<TR>

<TD colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/>

</TD>

</TR>

</TABLE>

</div>

</form>

Js.js:

//function test()

//{

// alert

//}

function stipendValidate(){

if(stipendForm.name.value==""){

alert("请添写员工姓名！");

stipendForm.name.focus();

return false;

}

if(stipendForm.basic.value==""){

alert("请添写基本薪金！");

stipendForm.basic.focus();

return false;

}

if(isNaN(stipendForm.basic.value))

{

alert("基本薪金只能填写数字！");

stipendForm.basic.focus();

return false;

}

if(stipendForm.granttime.value==""){

alert("请添写发放时间！");

stipendForm.granttime.focus();

return false;

}

if(!checkIsValidDate(stipendForm.granttime.value)){

alert("输入的时间不合法！（年-月-日）");

stipendForm.granttime.focus();

return false;

}

return true;

}

function institutionValidate(){

if(institutionForm.name.value==""){

alert("请添写奖惩名称！");

institutionForm.name.focus();

return false;

}

if(institutionForm.reason.value==""){

alert("请添写奖惩原因！");

institutionForm.reason.focus();

return false;

}

return true;

}

function userValidate(){

if(userForm.username.value==""){

alert("请添写人员姓名！");

userForm.username.focus();

return false;

}

// if(!/^[\u4e00-\u9fa5]+$/gi.exec(userForm.username.value) || !/^\\w+$/gi.exec(userForm.username.value)){

// alert("用户请输入中文！");

// userForm.username.focus();

// return false;

// }

if(userForm.password.value==""){

alert("请添写登录密码！");

userForm.password.focus();

return false;

}

if(escape(userForm.password.value).indexOf("%u")>=0 ){

alert("密码不能包含中文！");

return false;

}

if(userForm.birthday.value==""){

alert("请添写出生日期！");

userForm.birthday.focus();

return false;

}

if(CheckDate(userForm.birthday.value)){

alert("输入的日期不合法！（年-月-日）");

userForm.birthday.focus();

return false;

}

return true;

}

function educateValidate(){

if(educateForm.name.value==""){

alert("请添写培训名称！");

educateForm.name.focus();

return false;

}

if(educateForm.purpose.value==""){

alert("请添写培训目的！");

educateForm.purpose.focus();

return false;

}

if(educateForm.begintime.value==""){

alert("请添写培训开始时间！");

educateForm.begintime.focus();

return false;

}

if(!checkIsValidDate(educateForm.begintime.value)){

alert("输入的时间不合法！（年-月-日）");

educateForm.begintime.focus();

return false;

}

if(educateForm.endtime.value==""){

alert("请添写培训结束时间！");

educateForm.endtime.focus();

return false;

}

if(!checkIsValidDate(educateForm.endtime.value)){

alert("输入的时间不合法！（年-月-日）");

educateForm.endtime.focus();

return false;

}

if(!checkDateEarlier(educateForm.begintime.value,educateForm.endtime.value)){

alert("起始日期不能晚于结束日期！");

educateForm.begintime.focus();

return false;

}

if(educateForm.teacher.value==""){

alert("请添写培训讲师！");

educateForm.teacher.focus();

return false;

}

if(educateForm.student.value==""){

alert("请添写培训人员！");

educateForm.student.focus();

return false;

}

return true;

}

function checkDateEarlier(strStart,strEnd)

{

if(checkIsValidDate(strStart) == false || checkIsValidDate(strEnd) == false)

return false;

//如果有一个输入为空，则通过检验

if (( strStart == "" ) || ( strEnd == "" ))

return true;

var arr1 = strStart.split("-");

var arr2 = strEnd.split("-");

var date1 = new Date(arr1[0],parseInt(arr1[1].replace(/^0/,""),10) - 1,arr1[2]);

var date2 = new Date(arr2[0],parseInt(arr2[1].replace(/^0/,""),10) - 1,arr2[2]);

if(arr1[1].length == 1)

arr1[1] = "0" + arr1[1];

if(arr1[2].length == 1)

arr1[2] = "0" + arr1[2];

if(arr2[1].length == 1)

arr2[1] = "0" + arr2[1];

if(arr2[2].length == 1)

arr2[2]="0" + arr2[2];

var d1 = arr1[0] + arr1[1] + arr1[2];

var d2 = arr2[0] + arr2[1] + arr2[2];

if(parseInt(d1,10) > parseInt(d2,10))

return false;

else

return true;

}//~~~

function checkIsValidDate(str)

{

//如果为空，则通过校验

if(str == "")

return true;

var pattern = /^((\d{4})|(\d{2}))-(\d{1,2})-(\d{1,2})$/g;

if(!pattern.test(str)){

return false;

}

var arrDate = str.split("-");

if(parseInt(arrDate[0],10) < 100)

arrDate[0] = 2000 + parseInt(arrDate[0],10) + "";

var date = new Date(arrDate[0],(parseInt(arrDate[1],10) -1)+"",arrDate[2]);

if(date.getYear() == arrDate[0]

&& date.getMonth() == (parseInt(arrDate[1],10) -1)+""

&& date.getDate() == arrDate[2]){

return true;

}else{

return false;

}

}//~~~

function CheckDate(INDate)

{ if (INDate=="")

{return true;}

subYY=INDate.substr(0,4)

if(isNaN(subYY) || subYY<=0){

return true;

}

//转换月份

if(INDate.indexOf('-',0)!=-1){ separate="-"}

else{

if(INDate.indexOf('/',0)!=-1){separate="/"}

else {return true;}

}

area=INDate.indexOf(separate,0)

subMM=INDate.substr(area+1,INDate.indexOf(separate,area+1)-(area+1))

if(isNaN(subMM) || subMM<=0){

return true;

}

if(subMM.length<2){subMM="0"+subMM}

//转换日

area=INDate.lastIndexOf(separate)

subDD=INDate.substr(area+1,INDate.length-area-1)

if(isNaN(subDD) || subDD<=0){

return true;

}

if(eval(subDD)<10){subDD="0"+eval(subDD)}

NewDate=subYY+"-"+subMM+"-"+subDD

if(NewDate.length!=10){return true;}

if(NewDate.substr(4,1)!="-"){return true;}

if(NewDate.substr(7,1)!="-"){return true;}

var MM=NewDate.substr(5,2);

var DD=NewDate.substr(8,2);

if((subYY%4==0 && subYY%100!=0)||subYY%400==0){ //判断是否为闰年

if(parseInt(MM)==2){

if(DD>29){return true;}

}

}else{

if(parseInt(MM)==2){

if(DD>28){return true;}

}

}

var mm=new Array(1,3,5,7,8,10,12); //判断每月中的最大天数

for(i=0;i< mm.length;i++){

if (parseInt(MM) == mm[i]){

if(parseInt(DD)>31){return true;}

}else{

if(parseInt(DD)>30){return true;}

}

}

if(parseInt(MM)>12){return true;}

return false;

}

function jobValidate(){

//alert(jobForm.job.value);

//return false;

if(jobForm.name.value==""){

alert("请添写姓名！");

jobForm.name.focus();

return false;

}

if(jobForm.job.value==""){

alert("请添写职位！");

jobForm.job.focus();

return false;

}

if(jobForm.specialty.value==""){

alert("请添写所学专业！");

jobForm.specialty.focus();

return false;

}

if(jobForm.experience.value==""){

alert("请添写工作经验！");

jobForm.experience.focus();

return false;

}

if(jobForm.studyeffort.value==""){

alert("请添写学历！");

jobForm.studyeffort.focus();

return false;

}

if(jobForm.school.value==""){

alert("请添写毕业学校！");

jobForm.school.focus();

return false;

}

if(jobForm.tel.value==""){

alert("请添写电话！");

jobForm.tel.focus();

return false;

}

if(isNaN(jobForm.tel.value))

{

alert("电话只能填写数字！");

jobForm.tel.focus();

return false;

}

if(jobForm.email.value==""){

alert("请添写Email！");

jobForm.email.focus();

return false;

}

var email = jobForm.email.value;

if (email != "") {

var reg = /^\w+((-\w+)|(\.\w+))\*\@[A-Za-z0-9]+((\.|-)[A-Za-z0-9]+)\*\.[A-Za-z0-9]+$/;

isok= reg.test(email );

if (!isok) {

alert("邮箱格式不正确，请重新输入！");

return false;

}

};

return true;

}

function adminChecked(){

if(userForm.isadminhelp.checked){

document.userForm.isadmin.value="1";

}else{

document.userForm.isadmin.value="0";

}

}

function performanceValidate(){

if(performanceForm.score.value==""){

alert("请添写绩效分数！");

performanceForm.score.focus();

return false;

}

if(isNaN(performanceForm.score.value))

{

alert("绩效分数只能填写数字！");

performanceForm.score.focus();

return false;

}

if(performanceForm.addmoney.value==""){

alert("请添写增加工资金额！");

performanceForm.addmoney.focus();

return false;

}

if(isNaN(performanceForm.addmoney.value))

{

alert("增加工资金额只能填写数字！");

performanceForm.addmoney.focus();

return false;

}

if(performanceForm.content.value==""){

alert("请添写绩效说明！");

performanceForm.content.focus();

return false;

}

if(CheckDate(performanceForm.createtime.value)){

alert("输入的日期不合法！（年-月-日）");

performanceForm.createtime.focus();

return false;

}

return true;

}

EducateAction.java:

private ActionForward listEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

String educate=request.getParameter("educate");

if(educate==null||"".equals(educate)||"0".equals(educate)){

request.setAttribute("list",dao.listEducate(new Byte("0").byteValue()));

}else{

request.setAttribute("list",dao.listEducate(new Byte("1").byteValue()));

}

return mapping.findForward("success");

}

}



图4.2.3.1 未输入数据的培训计划列表



图4.2.3.2 培训总结列表



图4.2.3.3 培训总结列表详细信息

## 4.4程序（代码）设计

整个程序完成了前后端的分离，实现了“低耦合”的效果。前端完成获取用户操作的内容，包括应聘人员的信息、培训计划的信息等等，同时得到配置的基本数据，将这些数据传送到后台处理；后台接收数据根据实现约定好的接口向数据库分类存储数据，维护管理员在对网站进行维护的过程中，也会动态实现数据库的增删查改，以及缓存的更新。用户浏览数据的时候也会实时显示最新内容。[[17]](#footnote-17)

Educate.java:

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

/\*\* 培训 \*/

public class Educate implements Serializable {

private Long id; //培训编号

private String name; //培训名称

private String purpose; //培训目的

private Date begintime; //培训开始时间

private Date endtime; //培训结束时间

private String datum; //培训材料

private String teacher; //培训讲师

private String student; //培训人员

private Date createtime;//创建时间

private Byte educate; //培训是否完成

private String effect; //培训效果

private String summarize;//总结

public Educate(String name, String purpose, Date begintime, Date endtime, String datum, String teacher, String student, Date createtime, Byte educate,String effect, String summarize) {

this.name = name;

this.purpose = purpose;

this.begintime = begintime;

this.endtime = endtime;

this.datum = datum;

this.teacher = teacher;

this.student = student;

this.createtime = createtime;

this.educate=educate;

this.effect = effect;

this.summarize = summarize;

}

/\*\* default constructor \*/

public Educate() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public String getPurpose() {

return this.purpose;

}

public void setPurpose(String purpose) {

this.purpose = purpose;

}

public Date getBegintime() {

return this.begintime;

}

public void setBegintime(Date begintime) {

this.begintime = begintime;

}

public Date getEndtime() {

return this.endtime;

}

public void setEndtime(Date endtime) {

this.endtime = endtime;

}

public String getDatum() {

return this.datum;

}

public void setDatum(String datum) {

this.datum = datum;

}

public String getTeacher() {

return this.teacher;

}

public void setTeacher(String teacher) {

this.teacher = teacher;

}

public String getStudent() {

return this.student;

}

public void setStudent(String student) {

this.student = student;

}

public Date getCreatetime() {

return this.createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public String getEffect() {

return this.effect;

}

public void setEffect(String effect) {

this.effect = effect;

}

public String getSummarize() {

return this.summarize;

}

public void setSummarize(String summarize) {

this.summarize = summarize;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Educate] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" begintime = " + this.begintime+ ";\n");

toStr.append(" endtime = " + this.endtime+ ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" purpose = " + this.purpose + ";\n");

toStr.append(" datum = " + this.datum + ";\n");

toStr.append(" teacher = " + this.teacher + ";\n");

toStr.append(" student = " + this.student + ";\n");

toStr.append(" educate = " + this.educate+ ";\n");

toStr.append(" effect = " + this.effect + ";\n");

toStr.append(" summarize = " + this.summarize + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

/\*\*

\* @return Returns the educate.

\*/

public Byte getEducate() {

return educate;

}

/\*\*

\* @param educate The educate to set.

\*/

public void setEducate(Byte educate) {

this.educate = educate;

}

}

Institution.java:

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

/\*\* 奖惩 \*/

public class Institution implements Serializable {

private Long id; //奖惩编号

private String name; //奖惩名称

private String reason; //奖惩原因

private String explain; //奖惩说明

private Date createtime;//创建时间

/\*\* full constructor \*/

public Institution(String name, String reason, String explain, Date createtime) {

this.name = name;

this.reason = reason;

this.explain = explain;

this.createtime = createtime;

}

/\*\* default constructor \*/

public Institution() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public String getReason() {

return this.reason;

}

public void setReason(String reason) {

this.reason = reason;

}

public String getExplain() {

return this.explain;

}

public void setExplain(String explain) {

this.explain = explain;

}

public Date getCreatetime() {

return this.createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Institution] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" reason = " + this.reason + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" explain = " + this.explain + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

Job.java:

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

public class Job implements Serializable {

private Long id; //应聘人员编号

private String name;//应聘人员姓名

private Byte sex; //性别

private Integer age;//年龄

private String job; //职位

private String specialty;//所学专业

private String experience;//工作经验

private String studyeffort;//学历

private String school; //毕业学校

private String tel; //电话号码

private String email; //Email地址

private Date createtime;//创建时间

private String content; //详细经历

private Byte isstock; //是否入库

private Byte health;//体检结果

public Job(Long id,String name, Byte sex, Integer age, String job, String specialty, String experience, String studyeffort, String school, String tel, String email, Date createtime, String content, Byte isstock,Byte health) {

this.id=id;

this.name = name;

this.sex = sex;

this.age = age;

this.job = job;

this.specialty = specialty;

this.experience = experience;

this.studyeffort = studyeffort;

this.school = school;

this.tel = tel;

this.email = email;

this.createtime = createtime;

this.content = content;

this.isstock = isstock;

this.health=health;

}

public Job() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public Byte getSex() {

return this.sex;

}

public void setSex(Byte sex) {

this.sex = sex;

}

public Integer getAge() {

return this.age;

}

public void setAge(Integer age) {

this.age = age;

}

public String getJob() {

return this.job;

}

public void setJob(String job) {

this.job = job;

}

public String getSpecialty() {

return this.specialty;

}

public void setSpecialty(String specialty) {

this.specialty = specialty;

}

public String getExperience() {

return this.experience;

}

public void setExperience(String experience) {

this.experience = experience;

}

public String getStudyeffort() {

return this.studyeffort;

}

public void setStudyeffort(String studyeffort) {

this.studyeffort = studyeffort;

}

public String getSchool() {

return this.school;

}

public void setSchool(String school) {

this.school = school;

}

public String getTel() {

return this.tel;

}

public void setTel(String tel) {

this.tel = tel;

}

public String getEmail() {

return this.email;

}

public void setEmail(String email) {

this.email = email;

}

public Date getCreatetime() {

return this.createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public String getContent() {

return this.content;

}

public void setContent(String content) {

this.content = content;

}

public Byte getIsstock() {

return this.isstock;

}

public void setIsstock(Byte isstock) {

this.isstock = isstock;

}

public Byte getHealth() {

return this.health;

}

public void setHealth(Byte health) {

this.health = health;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Job] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" sex = " + this.sex + ";\n");

toStr.append(" age = " + this.age + ";\n");

toStr.append(" job = " + this.job + ";\n");

toStr.append(" specialty = " + this.specialty + ";\n");

toStr.append(" experience = " + this.experience + ";\n");

toStr.append(" studyeffort = " + this.studyeffort + ";\n");

toStr.append(" school = " + this.school + ";\n");

toStr.append(" tel = " + this.tel + ";\n");

toStr.append(" email = " + this.email + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" content = " + this.content+ ";\n");

toStr.append(" isstock = " + this.isstock + ";\n");

toStr.append(" health = " + this.health + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

Performance.java:

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

/\*\* 绩效 \*/

public class Performance implements Serializable {

private Long id; //绩效编号

private float score; //绩效分数

private float addmoney; //增加工资

private String content; //绩效说明

private Date createtime;//创建时间

/\*\* full constructor \*/

public Performance(float score, float addmoney, String content, Date createtime) {

this.score = score;

this.addmoney = addmoney;

this.content = content;

this.createtime = createtime;

}

/\*\* default constructor \*/

public Performance() {

}

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public float getScore() {

return score;

}

public void setScore(float score) {

this.score = score;

}

public float getAddmoney() {

return addmoney;

}

public void setAddmoney(float addmoney) {

this.addmoney = addmoney;

}

public String getContent() {

return content;

}

public void setContent(String content) {

this.content = content;

}

public Date getCreatetime() {

return createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Performance] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" content = " + this.content + ";\n");

toStr.append(" score = " + this.score + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" addmoney = " + this.addmoney + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

Stipend,java:

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

/\*\* 薪金 \*/

public class Stipend implements Serializable {

private Long id; //薪金信息编号

private String name; //员工姓名

private float basic; //基本薪金

private Date granttime; // 工资发放时间

private float duty; //全勤奖

private float punishment;//罚款

private float other; //额外补助

private float totalize; //总计

public Stipend(String name, float basic, Date granttime, float duty, float punishment, float other, float totalize) {

this.name = name;

this.basic = basic;

this.granttime = granttime;

this.duty = duty;

this.punishment = punishment;

this.other = other;

this.totalize = totalize;

}

/\*\* default constructor \*/

public Stipend() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public Float getBasic() {

return this.basic;

}

public void setBasic(Float basic) {

this.basic = basic;

}

/\*\*

\* @return Returns the granttime.

\*/

public Date getGranttime() {

return granttime;

}

/\*\*

\* @param granttime The granttime to set.

\*/

public void setGranttime(Date granttime) {

this.granttime = granttime;

}

public float getDuty() {

return this.duty;

}

public void setDuty(float duty) {

this.duty = duty;

}

public float getPunishment() {

return this.punishment;

}

public void setPunishment(float punishment) {

this.punishment = punishment;

}

public float getOther() {

return this.other;

}

public void setOther(float other) {

this.other = other;

}

public float getTotalize() {

return this.totalize;

}

public void setTotalize(float totalize) {

this.totalize = totalize;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Stipend] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" basic = " + this.basic + ";\n");

toStr.append(" granttime = " + this.granttime + ";\n");

toStr.append(" duty = " + this.duty + ";\n");

toStr.append(" punishment = " + this.punishment + ";\n");

toStr.append(" other = " + this.other + ";\n");

toStr.append(" totalize = " + this.totalize + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

Users.java:

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

public class Users implements Serializable {

private Long id; //员工编号

private String username;//员工用户名

private String password;//登录密码

private Byte sex; //性别

private Date birthday; //生日

private Date createtime;//创建时间

private Byte isadmin; //是否为管理员

private String content; //人员简介

public Users(Long id,String username, String password, Byte sex, Date birthday, Date createtime, Byte isadmin, String content) {

this.id=id;

this.username = username;

this.password = password;

this.sex = sex;

this.birthday = birthday;

this.createtime = createtime;

this.isadmin = isadmin;

this.content = content;

}

public Users() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getUsername() {

return this.username;

}

public void setUsername(String username) {

this.username = username;

}

public String getPassword() {

return this.password;

}

public void setPassword(String password) {

this.password = password;

}

public Byte getSex() {

return this.sex;

}

public void setSex(Byte sex) {

this.sex = sex;

}

public Date getBirthday() {

return this.birthday;

}

public void setBirthday(Date birthday) {

this.birthday = birthday;

}

public Date getCreatetime() {

return this.createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public Byte getIsadmin() {

return this.isadmin;

}

public void setIsadmin(Byte isadmin) {

this.isadmin = isadmin;

}

public String getContent() {

return this.content;

}

public void setContent(String content) {

this.content = content;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Users] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" username = " + this.username + ";\n");

toStr.append(" password = " + this.password + ";\n");

toStr.append(" sex = " + this.sex + ";\n");

toStr.append(" birthday = " + this.birthday+ ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" isadmin = " + this.isadmin + ";\n");

toStr.append(" content = " + this.content + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

EducateAction.java:

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.EducateDao;

import com.zhangying.po.Educate;

/\*\*

\* @author BWeiMing

\*

\*/

public class EducateAction extends Action {

private EducateDao dao=new EducateDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nEducateAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listeducate".equals(action)){

return listEducate(mapping,form,request,response);

}else if("addeducate".equals(action)){

return addEducate(mapping,form,request,response);

}else if("deleteeducate".equals(action)){

return deleteEducate(mapping,form,request,response);

}else if("detaileducate".equals(action)){

return detailEducate(mapping,form,request,response);

}else if("updateeducate".equals(action)){

return updateEducate(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updateEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

EducateForm educateForm=(EducateForm)form;

Educate e=educateForm.populate();

dao.updateEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Educate e = dao.loadEducate(id.longValue());

System.out.println(e.getName());

request.setAttribute("educate",e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deleteEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id").toString());

Educate e=new Educate();

e.setId(id);

dao.deleteEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

EducateForm educateForm=(EducateForm)form;

Educate e=educateForm.populate();

dao.addEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

String educate=request.getParameter("educate");

if(educate==null||"".equals(educate)||"0".equals(educate)){

request.setAttribute("list",dao.listEducate(new Byte("0").byteValue()));

}else{

request.setAttribute("list",dao.listEducate(new Byte("1").byteValue()));

}

return mapping.findForward("success");

}

}

InstitutionAction.java:

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.InstitutionDao;

import com.zhangying.po.Institution;

/\*\*

\* @author BWeiMing

\*

\*/

public class InstitutionAction extends Action {

private InstitutionDao dao=new InstitutionDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nInstitutionAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listinstitution".equals(action)){

return listInstitution(mapping,form,request,response);

}else if("addinstitution".equals(action)){

return addInstitution(mapping,form,request,response);

}else if("updateinstitution".equals(action)){

return updateInstitution(mapping,form,request,response);

}else if("deleteinstitution".equals(action)){

return deleteInstitution(mapping,form,request,response);

}else if("detailinstitution".equals(action)){

return detailInstitution(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Institution i=dao.loadInstitution(id.longValue());

request.setAttribute("institution",i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deleteInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Institution i=new Institution();

i.setId(id);

dao.deleteInstitution(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updateInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

InstitutionForm institutionForm=(InstitutionForm)form;

Institution i=institutionForm.populate();

dao.updateInstitution(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

InstitutionForm institutionForm=(InstitutionForm)form;

Institution i=institutionForm.populate();

dao.addInstitution(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

request.setAttribute("list",dao.listInstitution());

return mapping.findForward("success");

}

}

JobAction.java:

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.JobDao;

import com.zhangying.po.Job;

public class JobAction extends Action {

private JobDao dao=new JobDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nJobAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listjob".equals(action)){

return listJob(mapping,form,request,response);

}else if("addjob".equals(action)){

return addJob(mapping,form,request,response);

}else if("deletejob".equals(action)){

return deleteJob(mapping,form,request,response);

}else if("updatejob".equals(action)){

return updateJob(mapping,form,request,response);

}else if("detailjob".equals(action)){

return detailjob(mapping,form,request,response);

}

return mapping.findForward("error");

}

private ActionForward detailjob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//获得参数id

//System.out.println("id="+id);

Job j=dao.loadJob(id.longValue());//加载该ID对应的应聘信息

request.setAttribute("job",j);//将应聘信息对象设置到request范围

return mapping.findForward("success");

}

private ActionForward updateJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

// System.out.println("111111111111112221111");

//System.out.println("form="+form);

JobForm jobform=(JobForm)form;

//System.out.println("jobform="+jobform);

Job j=jobform.populate();

j.setJob(jobform.getJob());

//System.out.println("j="+j);

dao.updateJob(j);

return mapping.findForward("success");

}

private ActionForward deleteJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//接受id参数

Job j=new Job();//新建一个Job对象

j.setId(id);//设置其id属性

dao.deleteJob(j);//调用dao对象完成Job对象的删除

return mapping.findForward("success");

}

private ActionForward addJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

JobForm jobform=(JobForm)form;//获得JobForm

Job job=jobform.populate();//获得表单提交的应聘信息

dao.addJob(job);//调用DAO完成应聘信息的保存

return mapping.findForward("success");//跳转到success逻辑视图

}

private ActionForward listJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

String isstock=request.getParameter("isstock");//获得isstock参数

System.out.println("isstock="+isstock);

if(isstock==null||"".equals(isstock)){//如果isstock为null或者为空

request.setAttribute("list",

dao.listJob(new Byte("0").byteValue()));//查询所有没入库的应聘信息

}else{

request.setAttribute("list",

dao.listJob(new Byte("1").byteValue()));//查询所有入库的应聘信息

}

return mapping.findForward("success");

}

}

PerformanceAction.java:

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.PerformanceDao;

import com.zhangying.po.Performance;

/\*\*

\* @author f(x)函数

\*

\*/

public class PerformanceAction extends Action {

private PerformanceDao dao=new PerformanceDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nPerformanceAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listperformance".equals(action)){

return listPerformance(mapping,form,request,response);

}else if("addperformance".equals(action)){

return addPerformance(mapping,form,request,response);

}else if("updateperformance".equals(action)){

return updatePerformance(mapping,form,request,response);

}else if("deleteperformance".equals(action)){

return deletePerformance(mapping,form,request,response);

}else if("detailperformance".equals(action)){

return detailPerformance(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailPerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Performance i=dao.loadPerformance(id.longValue());

request.setAttribute("performance",i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deletePerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Performance i=new Performance();

i.setId(id);

dao.deletePerformance(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updatePerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

PerformanceForm performance=(PerformanceForm)form;

Performance i=performance.populate();

dao.updatePerformance(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addPerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

PerformanceForm PerformanceForm=(PerformanceForm)form;

Performance i=PerformanceForm.populate();

dao.addPerformance(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listPerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

request.setAttribute("list",dao.listPerformance());

return mapping.findForward("success");

}

}

StipendAction.java:

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.StipendDao;

import com.zhangying.po.Stipend;

public class StipendAction extends Action {

private StipendDao dao=new StipendDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nStipendAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("liststipend".equals(action)){

return listStipend(mapping,form,request,response);

}else if("addstipend".equals(action)){

return addStipend(mapping,form,request,response);

}else if("deletestipend".equals(action)){

return deleteStipend(mapping,form,request,response);

}else if("updatestipend".equals(action)){

return updateStipend(mapping,form,request,response);

}else if("detailstipend".equals(action)){

return detailStipend(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Stipend s=dao.loadStipend(id.longValue());

request.setAttribute("stipend",s);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updateStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

StipendForm stipendForm=(StipendForm)form;

Stipend s=stipendForm.populate();

dao.updateStipend(s);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deleteStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id").toString());

Stipend s=new Stipend();

s.setId(id);

dao.deleteStipend(s);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

StipendForm stipendForm=(StipendForm)form;

Stipend s=stipendForm.populate();

dao.addStipend(s);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

request.setAttribute("list",dao.listStipend());

return mapping.findForward("success");

}

}

UsersAction.java:

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.UsersDao;

import com.zhangying.po.Users;

public class UsersAction extends Action {

private UsersDao dao=new UsersDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

if(action==null||" ".equals(action)){

return mapping.findForward("error");

}else if("listuser".equals(action)){

return listUser(mapping,form,request,response);

}else if("adduser".equals(action)){

return addUsers(mapping,form,request,response);

}else if("logon".equals(action)){

return logon(mapping,form,request,response);

}else if("updateuser".equals(action)){

return updateUser(mapping,form,request,response);

}else if("deleteuser".equals(action)){

return deleteUser(mapping,form,request,response);

}else if("selectuser".equals(action)){

return selectUser(mapping,form,request,response);

}

return mapping.findForward("error");

}

private ActionForward selectUser(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//获得id参数

Users u=dao.loadUsers(id);//加载该ID对应的人员信息

request.setAttribute("user",u);//将人员信息设置到request范围

return mapping.findForward("success");//跳转到success对应的逻辑视图

}

private ActionForward updateUser(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

UsersForm usersform=(UsersForm)form;//获得UsersForm

Users users=usersform.populate();//获得提交的人员信息

dao.updateUsers(users);//更新人员信息

return mapping.findForward("success");//跳转到success对应的逻辑视图

}

private ActionForward deleteUser(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//获得id参数

Users users=new Users();//新建一个users对象

users.setId(id);//设置该users对象的id

dao.deleteUsers(users);//删除该users对象

return mapping.findForward("success");

}

private ActionForward logon(ActionMapping mapping,ActionForm form,HttpServletRequest request,HttpServletResponse response) throws HibernateException {

UsersForm usersform=(UsersForm)form;

// System.out.println("userform="+form);

Users users=new Users();

users.setUsername(usersform.getUsername());

users.setPassword(usersform.getPassword());

//users.setUsername(request.getParameter("username"));

//users.setPassword(request.getParameter("password"));

boolean flag=dao.logonUsers(users);

if(flag){

request.getSession().setAttribute("users",users);

return mapping.findForward("success");

}else{

request.setAttribute("error", "登录失败");

return mapping.findForward("failed");

}

}

private ActionForward addUsers(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

UsersForm usersform=(UsersForm)form; //获得UsersForm

Users users=usersform.populate(); //获得提交的人员信息

dao.addUsers(users); //完成人员信息保存

return mapping.findForward("success"); //跳转到成功页面

}

private ActionForward listUser(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

request.setAttribute("list",dao.listUser());//将人员信息列表保存到request范围

return mapping.findForward("success"); //跳转到成功页面

}

}

**4.5软件测试**

### 4.5.1软件测试的概念

软件测试，即描述一种用来促进鉴定软件的正确性、完整性、安全性和质量的过程。软件测试的经典定义是：在规定的条件下对程序进行操作，以发现程序错误，衡量软件质量，并对其是否能满足设计要求进行评估的过程。

### 4.5.2测试方法

**4.5.2.1 白盒测试**

白盒测试是知道QuickCMS内部工作过程，可通过测试来检测产品内部动作是否按照规格说明书的规定正常进行，按照程序内部的结构测试程序，检验程序中的每条通路是否都有能按预定要求正确工作，而不顾它的功能.

**4.5.2.2黑盒测试**

黑盒测试是指软件测试人员以用户的角度，通过各种输入和观察QuickCMS(测试软件)的各种输出结果来发现软件存在的缺陷，而不关心程序具体如何实现的一种软件测试方法。

**4.5.2.3可移植性测试**

可移植性测试是指将QuickCMS移植到指定的硬件或软件平台上看是否能运行成功。

**4.5.2.4 UI测试**

用户界面测试是指将QuickCMS给用户测试，看界面的风格是否满足客户要求，文字是否正确，页面是否美观，文字，图片组合是否完美，操作是否友好等等。UI 测试的目标是确保用户界面会通过测试对象的功能来为用户提供相应的访问或浏览功能。

**4.5.2.5随机测试**

随机测试主要是根据测试者的经验对QuickCMS进行功能和性能抽查。根据测试说明书执行用例测试，保证测试覆盖完整性。主要是对QuickCMS的一些重要功能进行复测，也包括测试当前没有覆盖到的部分。另外，对于软件更新和新增加的功能要重点测试。重点对一些特殊点情况点、特殊的使用环境、并发性、进行检查。对出现的bug，还要再次测试。

**4.5.2.6功能测试**

**4.5.2.6.1功能测试步骤**

（1）系统启动正常，有相应的提示框、错误提示等

（2）每项功能符合实际要求

（3）系统的界面清晰、美观

（4）菜单、按钮操作正常、灵活，能处理一些异常操作

（5）能接受正确的数据输入，对异常数据的输入有提示、容错处理等

（6）数据的输出结果准确，格式清晰，可以保存和读取

（7）功能逻辑清楚，符合使用者习惯

（8）系统的各种状态按照业务流程而变化，并保持稳定

（9）支持各种应用的环境

（10）能配合多种硬件周边设备

（11）系统升级后，能继续支持旧版本的数据

（12）与外部应用系统的接口有效

**4.5.2.6.2功能测试的方法**

（1）等价类划分法

（2）边界值分析法

（3）错误推测法

（4）因果图法

（5）组合分析法

**4.5.2.7性能测试**

性能测试通常会使用特定的测试工具，来模拟超常的数据量、负载等，监测系统的各项性能指标，如CPU和内存的使用情况、响应时间、反应速度等。

**4.5.2.7.1性能测试的目的**

为了验证系统是否达到用户提出的性能指标，同时发现系统中存在的性能瓶颈，起到优化系统的目的。

**4.5.2.7.2性能测试指标的来源**

用户对各项指标提出的明确需求；如果用户没有提出性能指标则根据用户需求、测试设计人员的经验来设计各项测试指标。

**4.5.2.7.3主要的性能指标**

服务器的各项指标（CPU、内存占用率等）、后台数据库的各项指标、网络流量、响应时间

**4.5.2.7.4性能测试要点**

测试环境应尽量与产品运行环境保持一致，应单独运行尽量避免与其他软件同时使用。

性能测试一般使用测试工具和测试人员编制测试脚本来完成。

性能测试的重点在于前期数据的设计与后期数据的分析。

性能测试的用例主要涉及到整个系统架构的问题，所以测试用例一旦生成，改动一般不大，所以做性能测试的重复使用率一般比较高。

**4.5.2.7.5两种负载类型**

“flat”测试；ramp-up测试

**4.5.2.8验收测试**

在软件产品完成了功能测试和系统测试之后、产品发布之前所进行的软件测试活动它是技术测试的最后一个阶段,也称为交付测试。

**4.5.2.8.1前提**

系统或软件产品已通过了系统测试的软件系统。

**4.5.2.8.2测试内容**

验证系统是否达到了用户需求规格说明书（可能包括项目或产品验收准则）中的要求，测试试图尽可能地发现软件中存留的缺陷，从而为软件进一步改善提供帮助，并保证系统或软件产品最终被用户接受。主要包括易用性测试、兼容性测试、安装测试、文档（如用户手册、操作手册等）测试等几个方面的内容。

**4.5.2.8.3测试步骤**

制定测试计划，测试项，测试策略及验收通过准则，并经过客户参与的计划评审。

建立测试环境，设计测试用例，并经过评审。

准备测试数据，执行测试用例，记录测试结果。

分析测试结果，根据验收通过准则分析测试结果，作出验收是否通过及测试评价。

测试项目通过；

测试项目没有通过，并且不存在变通方法，需要很大的修改；

测试项目没有通过，但存在变通方法，在维护后期或下一个版本改进；

测试项目无法评估或者无法给出完整的评估。此时必须给出原因。如果是因为该测试项目没有说明清楚，应该修改测试计划。[[18]](#footnote-18)

### 4.5.3测试实例

**4.5.3.1测试的步骤**

1、测试人员要仔细阅读有关资料，包括设计文档、规格说明、测试大纲、使用说明、测试内容及测试的通过准则，做到整体了解系统，然后写测试的计划，测试用例，为测试做充足的准备。

2、为了保证测试的质量，将测试过程分成几个阶段，即：代码审查、单元测试、集成测试和验收测试。

3、代码会审：代码会审是由一组人通过阅读、讨论和争议对程序进行静态分析的过程。会审小组在充分阅读待审程序文本、控制流程图及有关要求、规范等文件基础上，召开代码会审会，程序员逐句讲解程序的逻辑，并展开热烈的讨论甚至争议，以揭示错误的关键所在[14]。实践表明，程序员在讲解过程中能发现许多自己原来没有发现的错误，而讨论和争议则进一步促使了问题的暴露。

4、单元测试： 单元测试集中在检查软件设计的最小单位-模块上，通过测试发现实现该模块的实际功能与定义该模块的功能说明不符合的情况，以及编码的错误。采用结构测试（白盒法）的用例，尽可能达到彻底测试，然后辅之以功能测试（黑盒法）的用例，使之能鉴别和响应任何合理和不合理的输入[15]。

5、集成测试：集成测试是将模块按照设计要求组装起来，同时进行测试，主要目标是发现与接口有关的问题。如数据穿过接口时可能丢失；把子功能组合起来可能不产生预期的主功能；全程数据结构可能有错误等。

6、验收测试：验收测试的目的是向未来的用户表明，系统能够像预定要求那样工作。经集成测试后，已经按照设计把所有的功能模块组装成一个比较完整的软件系统，验收测试的任务就是进一步验证软件的有效性，即如同用户所期待的那样，软件的功能、性能等其他方面都可以实现。

经过上述的测试过程对软件测试后，软件基本满足开发的要求，测试任务完成，验收完毕后，便可以将软件发送至用户。

**4.5.3.2 黑盒测试与白盒测试**

管理员及用户的登录模块，我们采用黑盒（白盒）测试的方式对系统进行测试。

**4.5.3.2.1 登录模块测试用例**

逻辑覆盖是白盒测试方案的一种技术。设计测试方案是测试阶段的关键技术问题。所谓测试方案包括具体的测试目的（例如，要测试的具体功能），应该输入的测试数据和预期的输出结果。通常又把测试数据和输出结果称为测试用例。

不同的测试数据发现程序错误的能力差别很大，为了提高测试效率降低测试成本，应该选用高效的测试数据。因为不可能进行穷尽的测试，选用少量“最有效的”测试数据，做到尽可能完备的测试就更重要了。有选择地执行程序中某些最有代表性的通路是对穷尽测试的唯一可行的替代办法。所谓逻辑覆盖是对一系列测试过程的总称，这组测试过程逐渐进行你越来越完整的通路测试。

语句覆盖：为了暴露程序中的错误，至少每个语句应该执行一次。语句覆盖的含义是，选择足够多的测试数据，是被测试程序中每个语句至少执行一次测试用例：

测试案例1：UN=佘华扬，PW=1234567zz

测试案例2：UN=廖娴静，PW=zz123

测试案例3：UN=宫春燕，PW=ggg1234

判定覆盖：又叫分支覆盖，它的含义是，不仅每个语句必须至少执行一次，而且每个判定的每种可能的结果都应该至少执行一次，也就是每个判定的每个分支都至少执行一次。

测试用例同语句覆盖的测试用例相同，此处省略。

黑盒测试：

运用等价分类法划分等价类补充用例，有效等价类如测试案例2；无效等价类如测试案例1（有效等价类、无效等价类较多，此处只举一例），然后用猜错法附加用例。

测试案例4：UN为空，PW为空（猜错法）

测试案例5：UN为空，PW=1234567zz猜错法）

测试案例6：UN=admin，PW=admin（猜错法）

测试案例7：UN=廖娴静，PW为空（无效等价类）

表4.5.3.2.1 登录模块测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **案例** | **应产生行为** | **实际测试结果** | **失败原因** |
| 1. UN=佘华扬，PW=zz123 | 提示用户用户名或密码错误 | 成功 |  |
| 1. UN=廖娴静，PW=zz123 | 提示用户登录成功，进入开发者管理界面 | 成功 |  |
| 1. UN为空，PW为空 | 提示用户输入用户名、密码 | 成功 |  |
| 1. UN为空，   PW=ggg1234 | 提示用户输入用户名 | 成功 |  |

**4.5.3.3压力测试（多用户使用测试）**

**4.5.3.3.1系统压力强度估算**

系统响应时间判断原则如下：

1. 系统业务响应时间小于2-5秒，判为优秀，用户对系统感觉很好；
2. 系统业务响应时间在5-10秒之间，判为良好，用户对系统感觉一般；
3. 系统业务响应时间超过15秒，判断为一般，用户体验不佳。

**4.5.3.3.2测试环境**

网络环境：校园内部的以太网，与服务器的连接速率为100.0M，与客户端的连接速率为10/100M自适应。

配置：

表4.5.3.3.2 测试环境配置表

|  |  |  |
| --- | --- | --- |
| 设备 | 硬件配置 | 软件配置 |
| 负载产生设备 | PC机（一台）  CPU：2.8  内存：1.5G | Windows 7 |

场景设计：

测试内容取：登录模块、后台管理模块。

场景设计思想是：逐步提高系统用户同时并发登录，并发配置，以检测系统的长期稳定性。

**4.5.3.3.3测试工具**

Apache JMeter是Apache组织开发的基于Java的压力测试工具。用于对软件做压力测试，它最初被设计用于Web应用测试，但后来扩展到其他测试领域。 它可以用于测试静态和动态资源，例如静态文件、Java 小服务程序、CGI 脚本、Java 对象、数据库、FTP 服务器， 等等。JMeter 可以用于对服务器、网络或对象模拟巨大的负载，来自不同压力类别下测试它们的强度和分析整体性能。另外，JMeter能够对应用程序做功能/回归测试，通过创建带有断言的脚本来验证你的程序返回了你期望的结果。为了最大限度的灵活性，JMeter允许使用正则表达式创建断言。

Apache jmeter 可以用于对静态的和动态的资源（文件，Servlet，Perl脚本，java 对象，数据库和查询，FTP服务器等等）的性能进行测试。它可以用于对服务器、网络或对象模拟繁重的负载来测试它们的强度或分析不同压力类型下的整体性能。你可以使用它做性能的图形分析或在大并发负载测试你的服务器/脚本/对象。

测试场景一：

1. 设置初始登录用户为：5人
2. 每30秒增加3个用户并发数
3. 逐步递增到15个用户并发数
4. 测试计算

一、登录模块

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 最小响应时间（s） | 最大响应时间（s） | 平均响应时间（s） |
| 第一次 | 4.00 | 6.00 | 5.00 |
| 第二次 | 4.02 | 7.00 | 5.51 |
| 第三次 | 3.21 | 5.23 | 4.22 |

二、配置数据模块

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 最小响应时间（s） | 最大响应时间（s） | 平均响应时间（s） |
| 第一次 | 10.50 | 22.50 | 16.50 |
| 第二次 | 13.01 | 20.99 | 17.00 |
| 第三次 | 12.21 | 23.29 | 17.75 |

三、后台管理模块。

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 最小响应时间（s） | 最大响应时间（s） | 平均响应时间（s） |
| 第一次 | 14.32 | 22.16 | 18.24 |
| 第二次 | 15.43 | 22.67 | 19.05 |
| 第三次 | 13.63 | 23.64 | 18.64 |

测试场景二：

1、设置初始登录用户为：5人

2、每30秒增加3个用户并发数

3、逐步递增到30个用户并发数

4、测试计算

一、登录模块

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 最小响应时间（s） | 最大响应时间（s） | 平均响应时间（s） |
| 第一次 | 12.05 | 12.19 | 12.11 |
| 第二次 | 12.06 | 14.99 | 12.20 |
| 第三次 | 12.22 | 12.30 | 12.26 |

二、配置数据模块

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 最小响应时间（s） | 最大响应时间（s） | 平均响应时间（s） |
| 第一次 | 10.85 | 28.44 | 19.65 |
| 第二次 | 11.47 | 28.51 | 19.49 |
| 第三次 | 12.33 | 29.32 | 18.33 |

三、后台管理模块。

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 最小响应时间（s） | 最大响应时间（s） | 平均响应时间（s） |
| 第一次 | 20.97 | 30.51 | 25.74 |
| 第二次 | 21.61 | 30.57 | 26.09 |
| 第三次 | 22．65 | 31．33 | 26.99 |

**4.5.3.3.4综述**

在系统测试过程中，系统在用户并发使用和反复运行中，系统未出现不良反应，系统发行良好，在大数据并发下载情况下，系统响应时间令人满意，系统稳定性比较可靠。

存在问题：系统配置数据模块在多用户并发下载数据压力下，速度降低，增加了用户等待时间，几个主要模块在几十个用户同时并发条件下的响应时间都超过10秒，制约了用户体验。

**4.6应用实例**

设计表单时包含一些预定义数据的一个业务文档，它通常包括一些区域，其中将填入另外的数据。以下是本系统表单设计规格说明：

### 4.6.1登录界面

（1）叙述性综述

表单：系统登录界面

用户：人事管理员

任务：用户打开该界面后可进行之后的登录操作

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

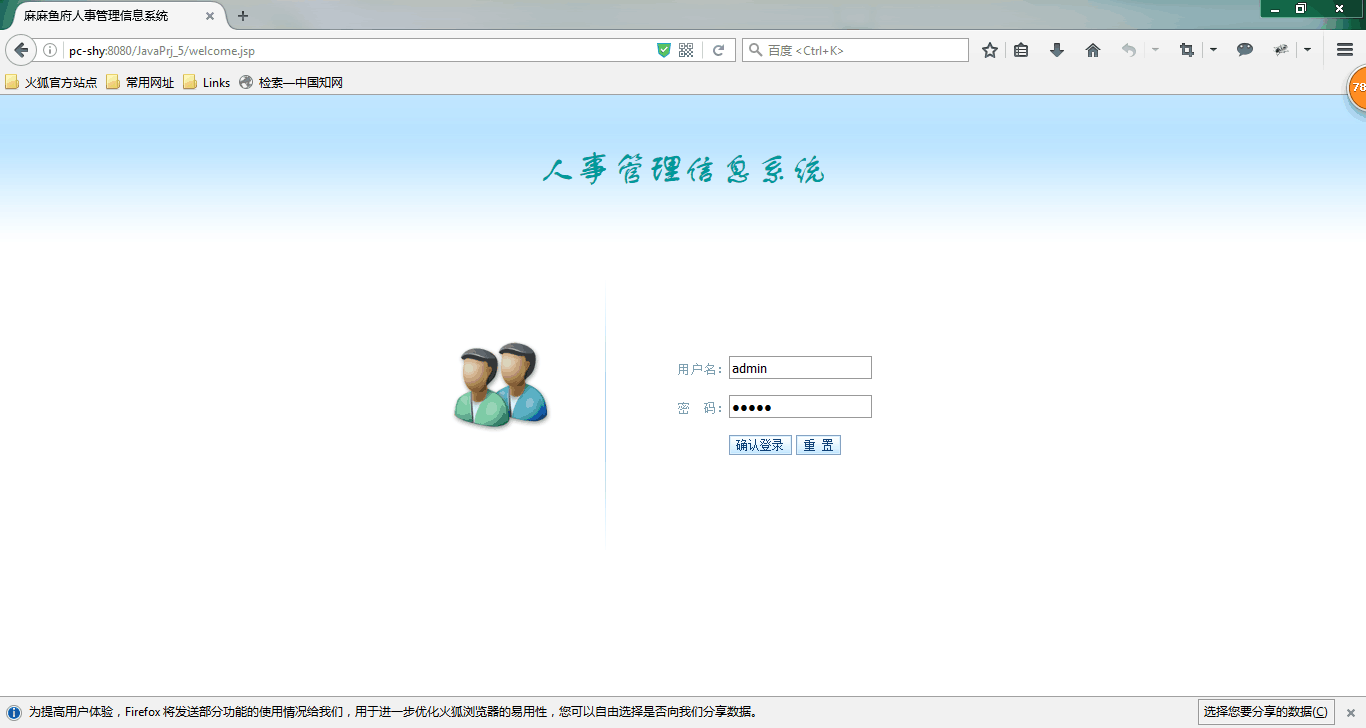


图4.6.1 系统登录界面

### 4.6.2系统首页

(1)叙述性综述

表单：系统首页界面

用户：人事管理员

任务：人事管理员登录后可进行人员管理、招聘管理、人事异动、培训管理、奖惩管理、绩效管理以及薪金管理几个功能的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

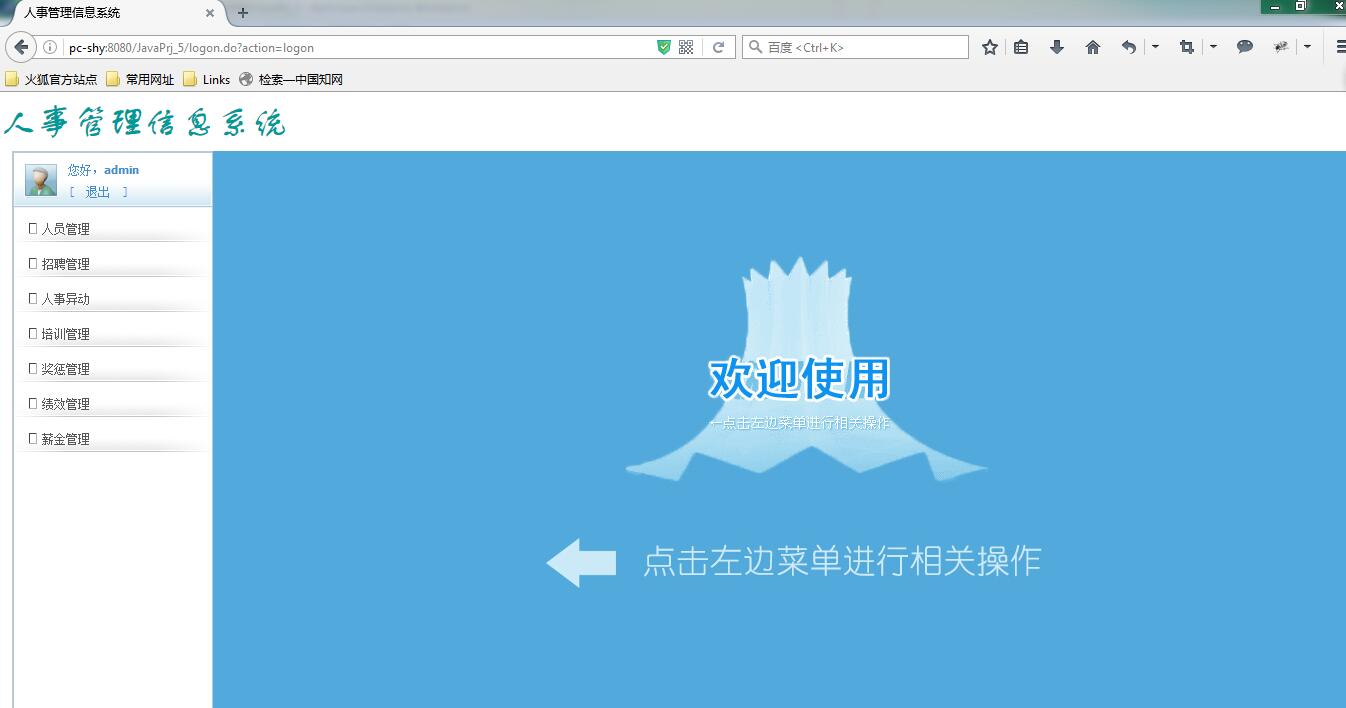


图4.6.2 项目首页

### 4.6.3人员管理界面

(1)叙述性综述

表单：系统人员管理界面

用户：人事管理员

任务：人事管理员选择人员管理功能后可进行人员信息录入、人员信息查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

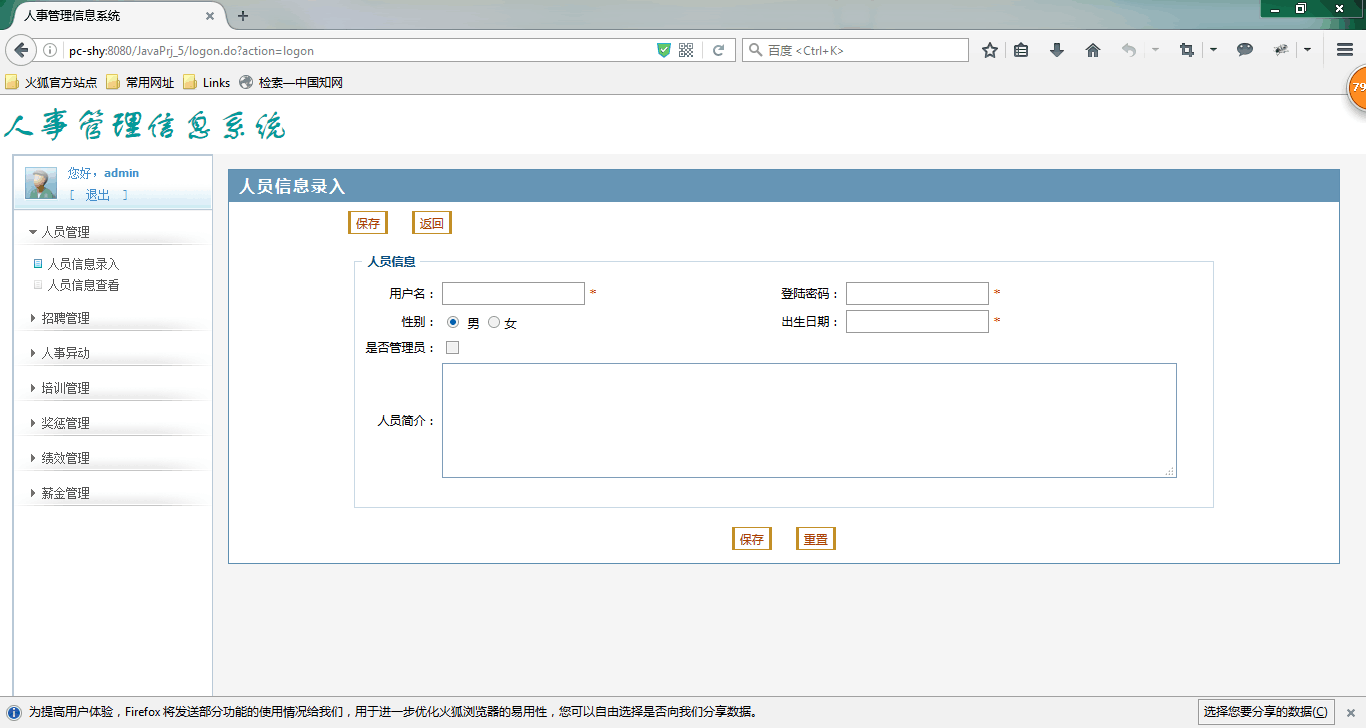


图4.6.3.1 人员信息录入

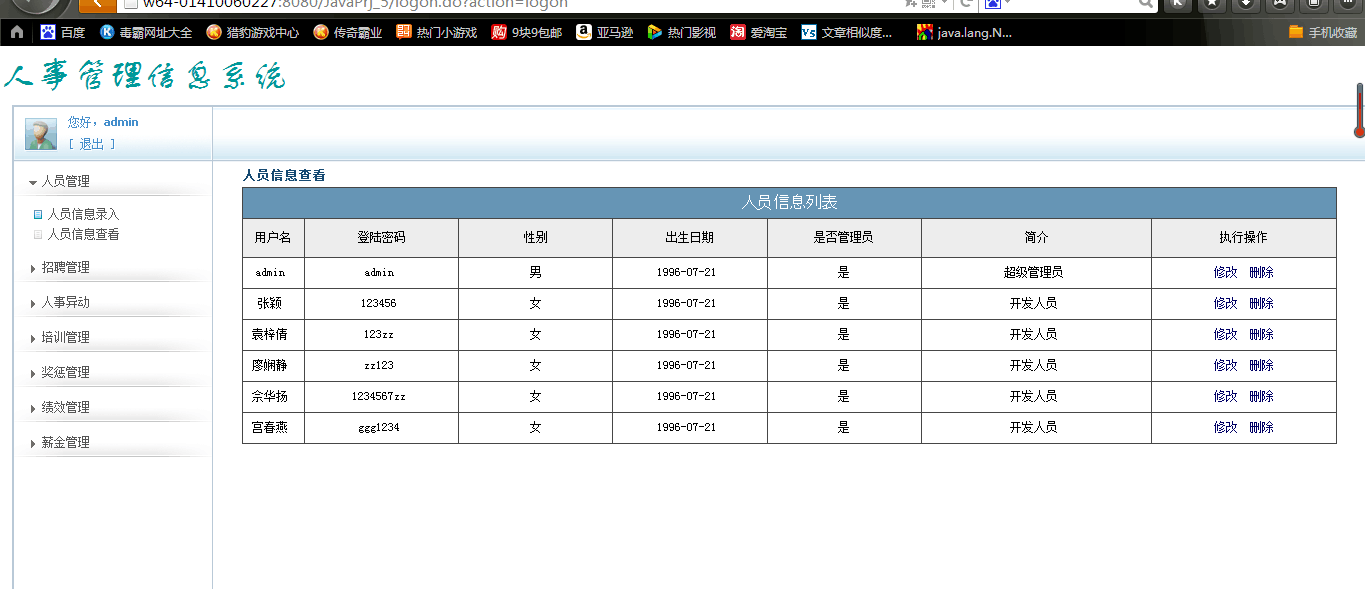


图4.6.3.2 人员信息查看

### 4.6.4招聘管理界面

(1)叙述性综述

表单：系统招聘管理界面

用户：人事管理员

任务：人事管理员选择招聘管理功能后可进行应聘信息录入、应聘信息查看、人才库浏览的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计



图4.6.4.1 应聘信息查看

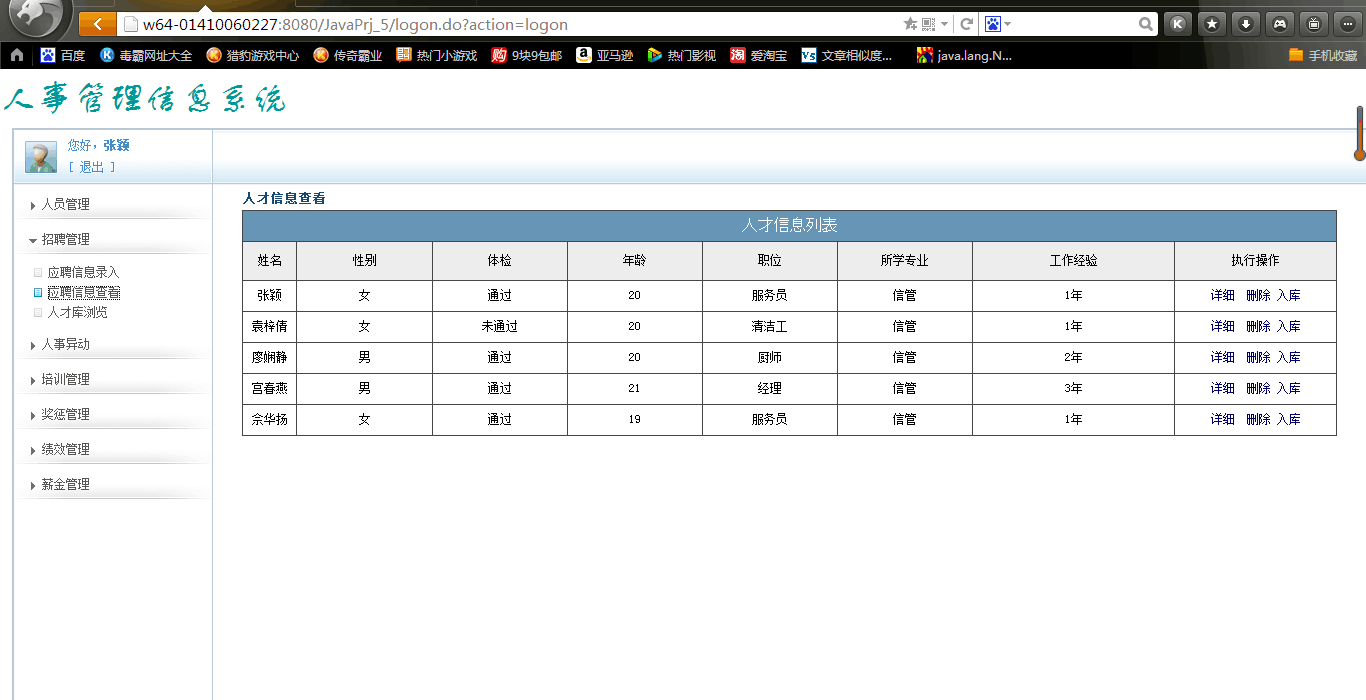


图4.6.4.2 应聘信息查看

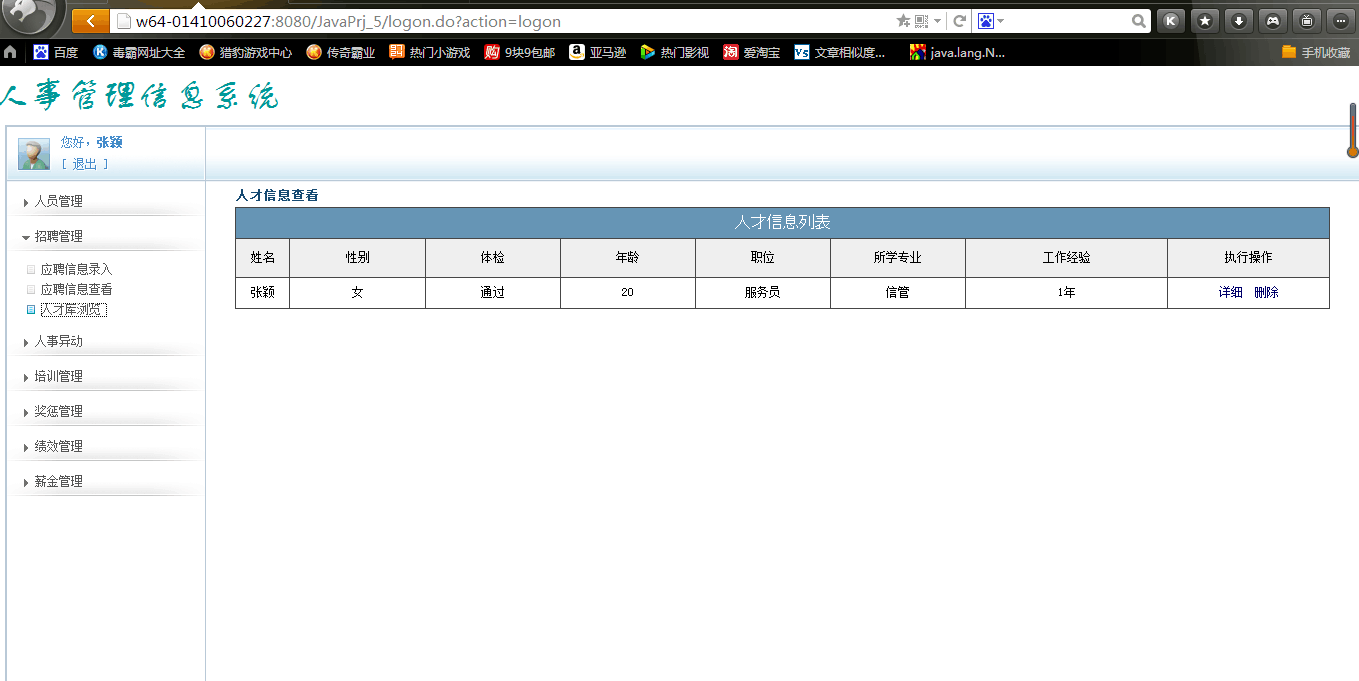


图4.6.4.3 人才库浏览

### 4.6.5人事异动界面

(1)叙述性综述

表单：系统人事异动界面

用户：人事管理员

任务：人事管理员选择人事异动功能后可进行人事信息查看的选择，在人事信息查看的界面还可以选择查看详细信息、修改的操作

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计



图4.6.5.1 人事信息查看



图4.6.5.2 详细信息查看



图4.6.5.3 人事信息修改

### 4.6.6培训管理界面

（1）叙述性综述

表单：系统培训管理界面

用户：人事管理员

任务：人事管理员选择培训管理功能后可进行培训计划录入、培训计划查看、培训总结查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

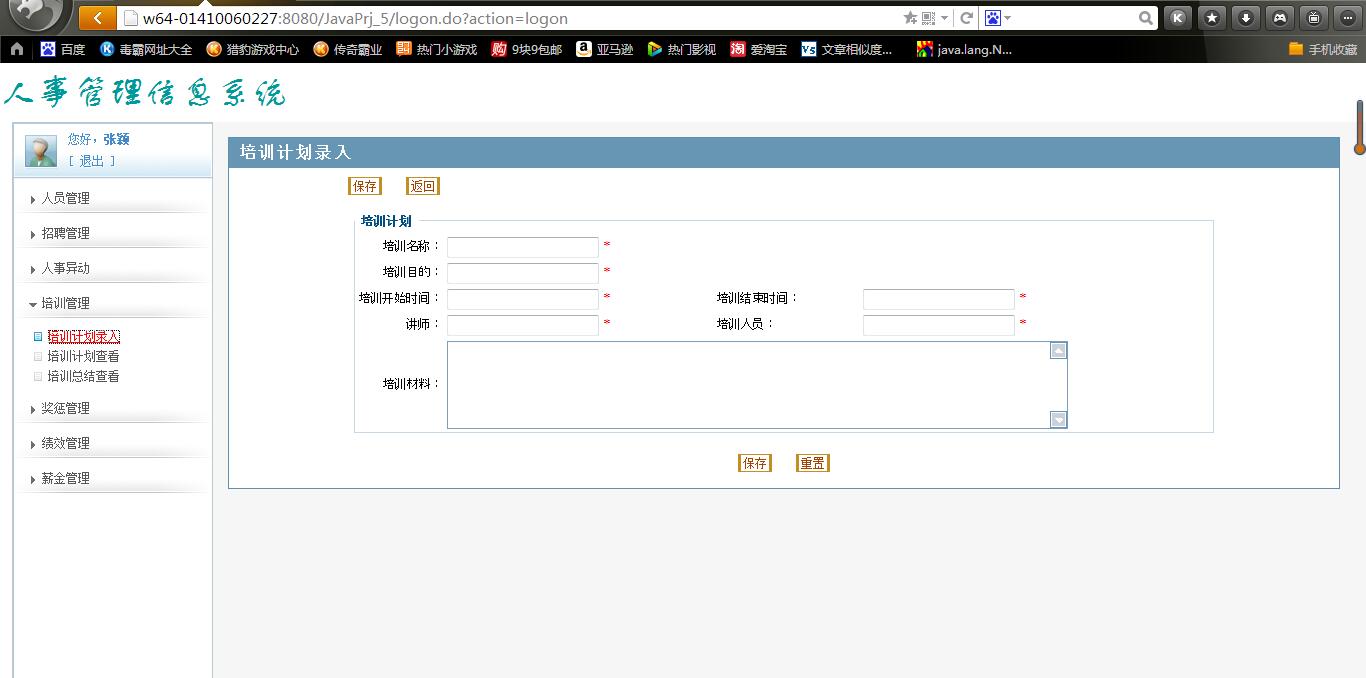


图4.6.6.1 培训计划录入

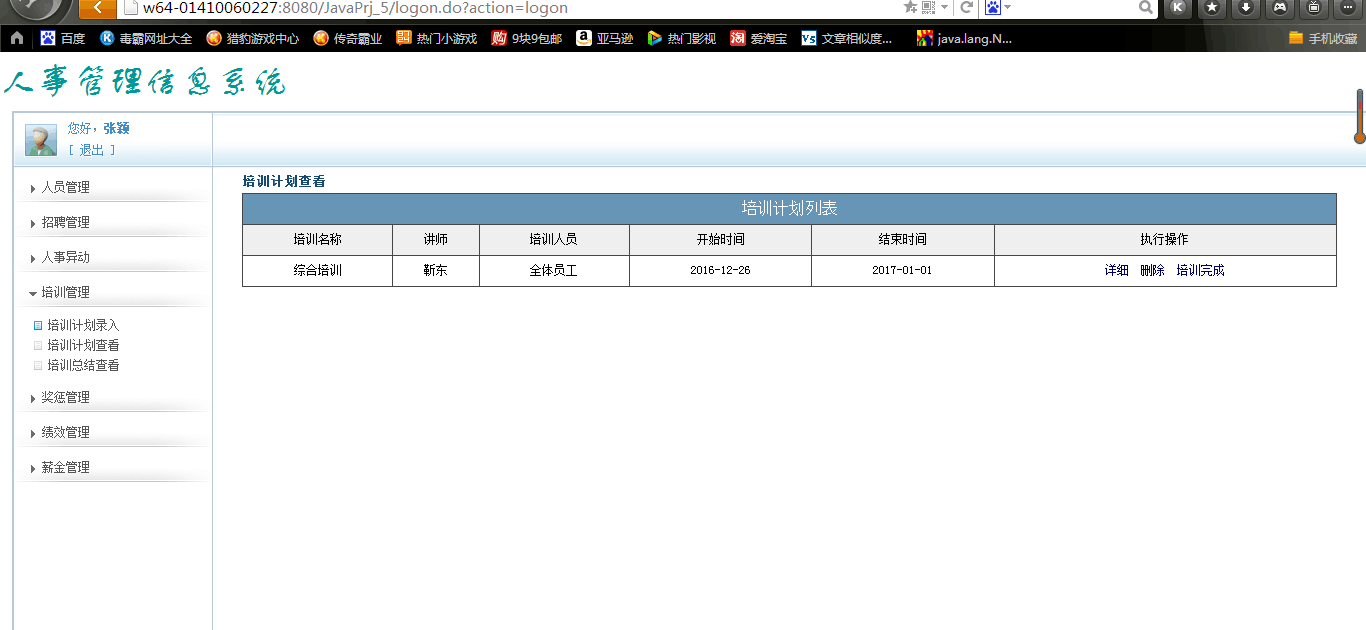


图4.6.6.2 培训计划查看

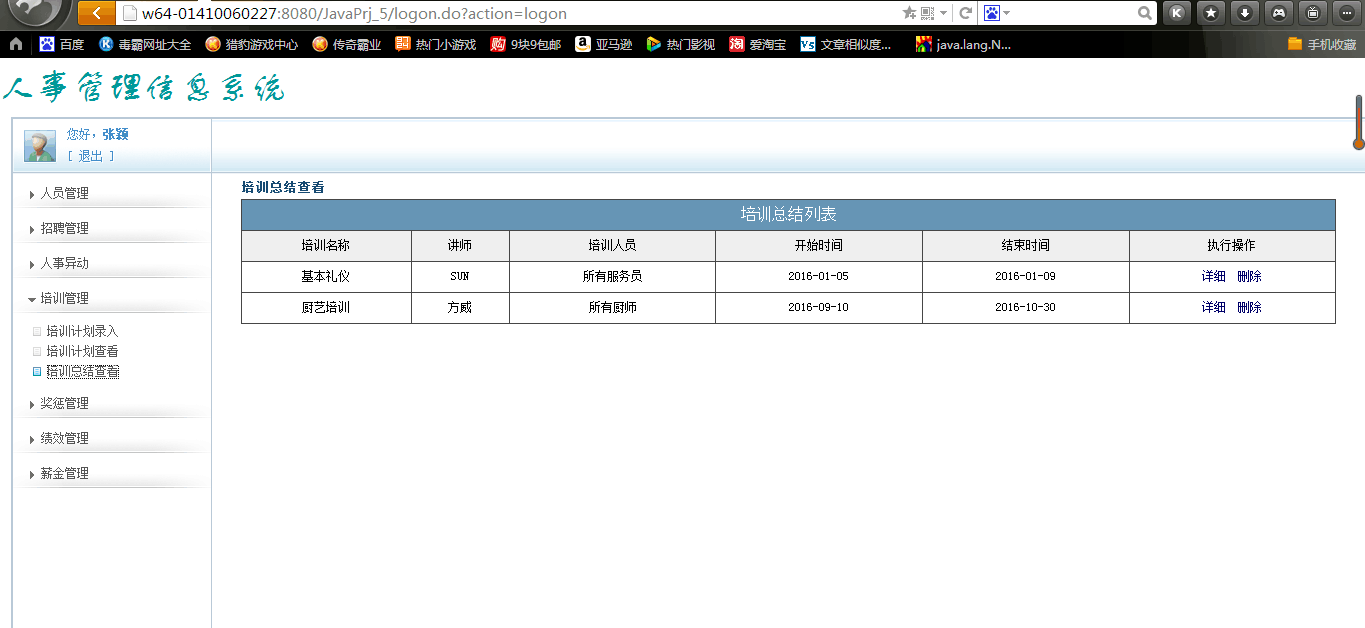


图4.5.6.3 培训总结查看

### 4.6.7奖惩管理界面

(1)叙述性综述

表单：系统奖惩管理界面

用户：人事管理员

任务：人事管理员选择奖惩管理功能后可进行奖惩信息登记、奖惩信息查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计



图4.6.7.1 奖惩信息登记

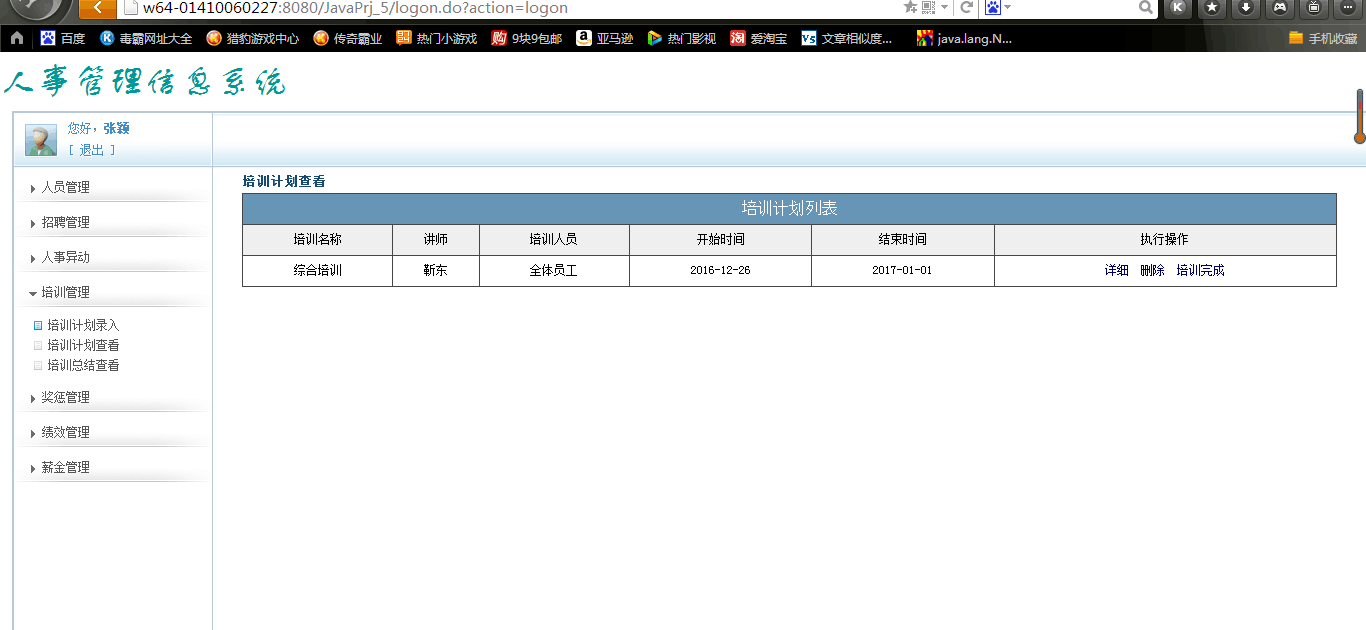


图4.6.7.2 奖惩信息查看

### 4.6.8绩效管理界面

(1)叙述性综述

表单：系统绩效管理界面

用户：人事管理员

任务：人事管理员选择绩效管理功能后可进行绩效信息登记、绩效信息查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计

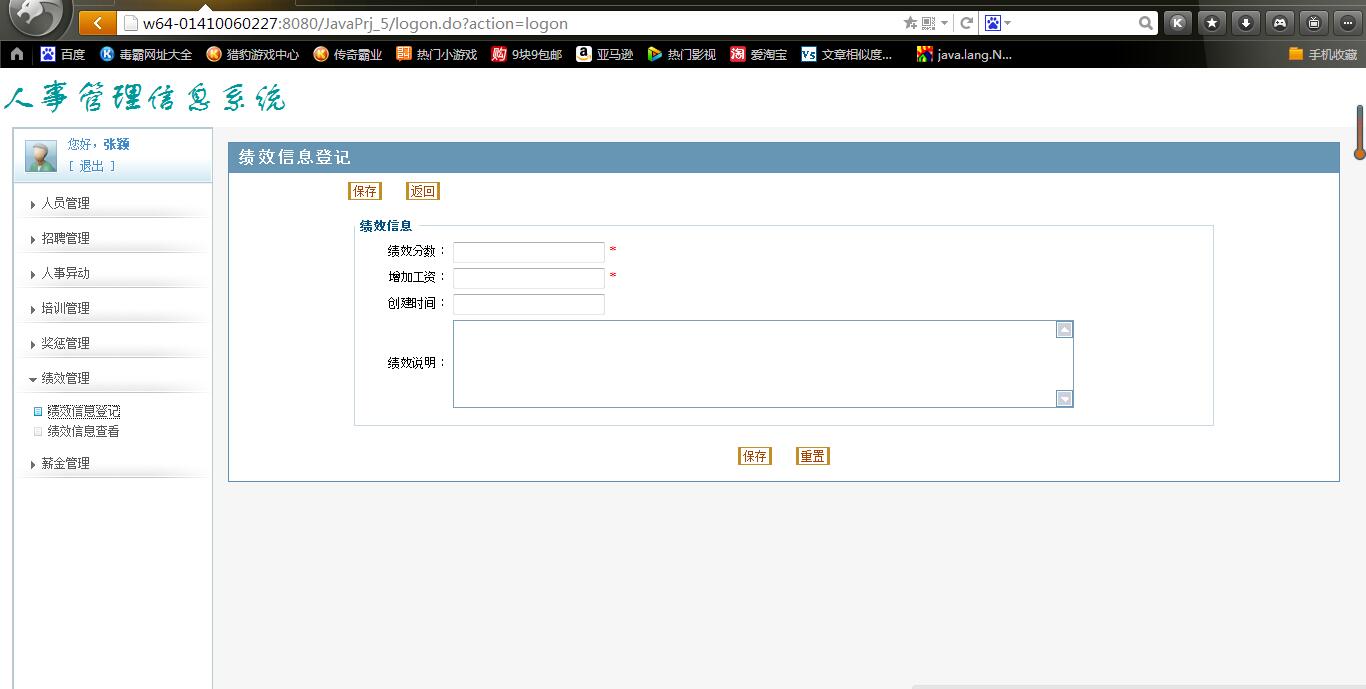


图4.6.8.1 绩效信息登记

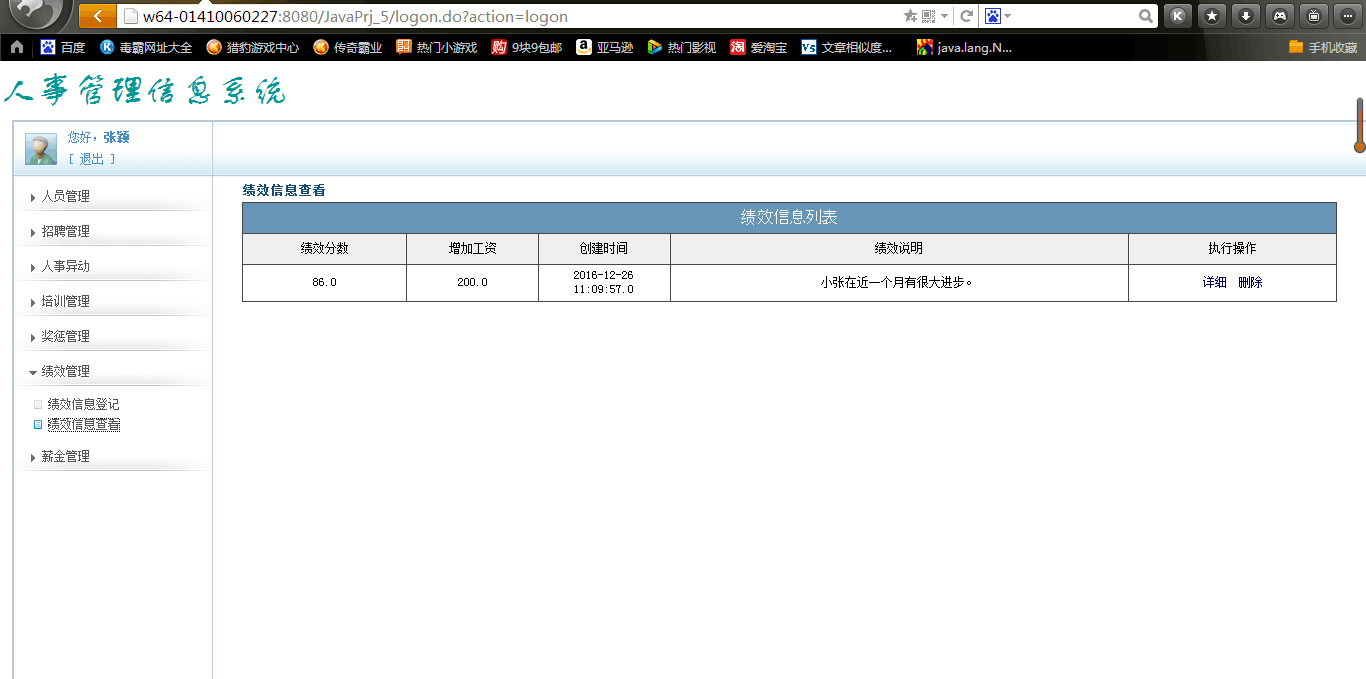


图4.6.8.2 绩效信息查看

### 4.6.9薪金管理界面

(1)叙述性综述

表单：系统薪金管理界面

用户：人事管理员

任务：人事管理员选择薪金管理功能后可进行薪金数据录入、薪金数据查看的选择

系统：Microsoft Windows

环境：普通网页环境

（2）样例设计



图4.6.9.1 薪金数据录入

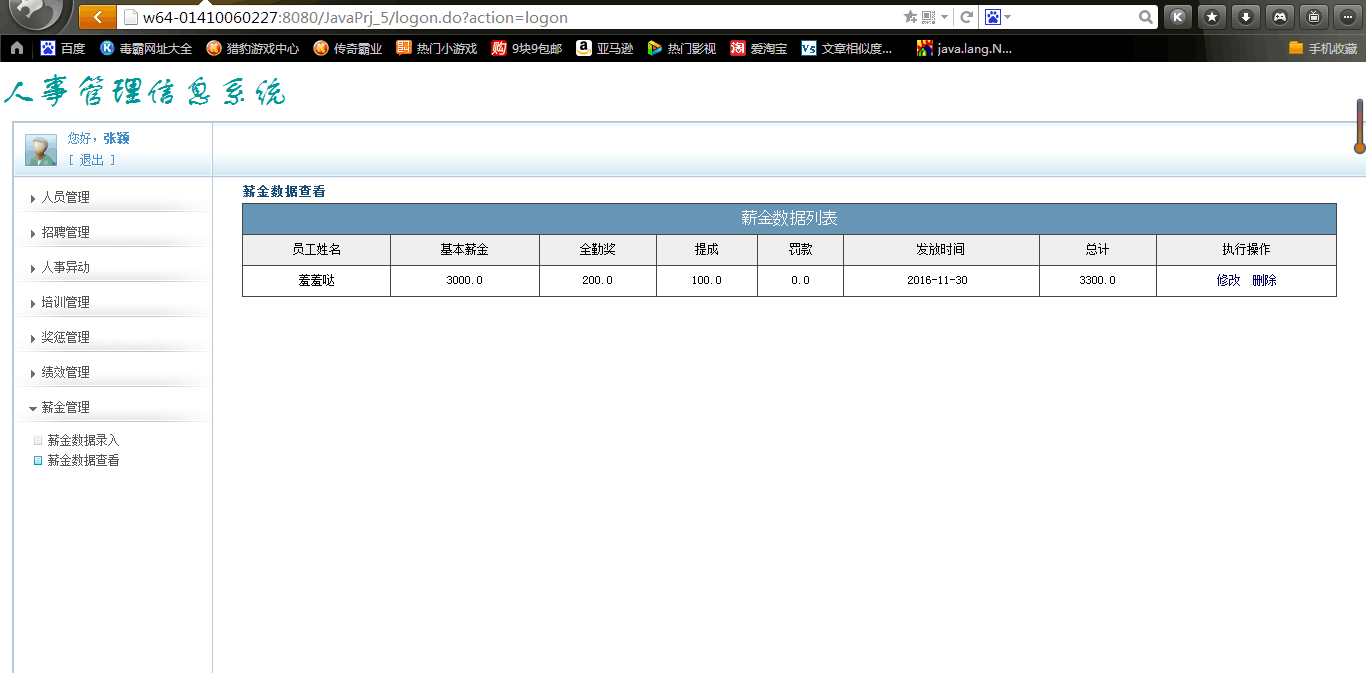


图4.6.9.2 绩效信息查看

**4.7创新应用实例**

### 4.7.1正则表达式应用

**（1）实例背景**：系统通过正则表达式判断各种用户输入值的合法性。正则表达式，又称规则表达式**。**首先，系统通过客户端获取到用户输入的值，再应用正则表达式对用户输入的值进行匹配处理，检查其中是否包含中英文、数字、格式是否正确等信息。若输入值不合法，客户端对用户进行请求的拦截和对用户的提醒，提示用户重新输入正确的值。此种举措可以防止不合法的值被传到后端，造成错误。

**（2）解决思路**：

部分代码如下：

function stipendValidate(){

if(stipendForm.name.value==""){

alert("请添写员工姓名！");

stipendForm.name.focus();

return false;

}

if(stipendForm.basic.value==""){

alert("请添写基本薪金！");

stipendForm.basic.focus();

return false;

}

if(isNaN(stipendForm.basic.value))

{

alert("基本薪金只能填写数字！");

stipendForm.basic.focus();

return false;

}

if(stipendForm.granttime.value==""){

alert("请添写发放时间！");

stipendForm.granttime.focus();

return false;

}

if(!checkIsValidDate(stipendForm.granttime.value)){

alert("输入的时间不合法！（年-月-日）");

stipendForm.granttime.focus();

return false;

}

return true;

}

function institutionValidate(){

if(institutionForm.name.value==""){

alert("请添写奖惩名称！");

institutionForm.name.focus();

return false;

}

if(institutionForm.reason.value==""){

alert("请添写奖惩原因！");

institutionForm.reason.focus();

return false;

}

return true;

}

function userValidate(){

if(userForm.username.value==""){

alert("请添写人员姓名！");

userForm.username.focus();

return false;

}

if(userForm.password.value==""){

alert("请添写登录密码！");

userForm.password.focus();

return false;

}

if(escape(userForm.password.value).indexOf("%u")>=0 ){

alert("密码不能包含中文！");

return false;

}

if(userForm.birthday.value==""){

alert("请添写出生日期！");

userForm.birthday.focus();

return false;

}

if(CheckDate(userForm.birthday.value)){

alert("输入的日期不合法！（年-月-日）");

userForm.birthday.focus();

return false;

}

return true;

}

function educateValidate(){

if(educateForm.name.value==""){

alert("请添写培训名称！");

educateForm.name.focus();

return false;

}

if(educateForm.purpose.value==""){

alert("请添写培训目的！");

educateForm.purpose.focus();

return false;

}

if(educateForm.begintime.value==""){

alert("请添写培训开始时间！");

educateForm.begintime.focus();

return false;

}

if(!checkIsValidDate(educateForm.begintime.value)){

alert("输入的时间不合法！（年-月-日）");

educateForm.begintime.focus();

return false;

}

if(educateForm.endtime.value==""){

alert("请添写培训结束时间！");

educateForm.endtime.focus();

return false;

}

if(!checkIsValidDate(educateForm.endtime.value)){

alert("输入的时间不合法！（年-月-日）");

educateForm.endtime.focus();

return false;

}

if(!checkDateEarlier(educateForm.begintime.value,educateForm.endtime.value)){

alert("起始日期不能晚于结束日期！");

educateForm.begintime.focus();

return false;

}

if(educateForm.teacher.value==""){

alert("请添写培训讲师！");

educateForm.teacher.focus();

return false;

}

if(educateForm.student.value==""){

alert("请添写培训人员！");

educateForm.student.focus();

return false;

}

return true;

}

function checkDateEarlier(strStart,strEnd)

{

if(checkIsValidDate(strStart) == false || checkIsValidDate(strEnd) == false)

return false;

//如果有一个输入为空，则通过检验

if (( strStart == "" ) || ( strEnd == "" ))

return true;

var arr1 = strStart.split("-");

var arr2 = strEnd.split("-");

var date1 = new Date(arr1[0],parseInt(arr1[1].replace(/^0/,""),10) - 1,arr1[2]);

var date2 = new Date(arr2[0],parseInt(arr2[1].replace(/^0/,""),10) - 1,arr2[2]);

if(arr1[1].length == 1)

arr1[1] = "0" + arr1[1];

if(arr1[2].length == 1)

arr1[2] = "0" + arr1[2];

if(arr2[1].length == 1)

arr2[1] = "0" + arr2[1];

if(arr2[2].length == 1)

arr2[2]="0" + arr2[2];

var d1 = arr1[0] + arr1[1] + arr1[2];

var d2 = arr2[0] + arr2[1] + arr2[2];

if(parseInt(d1,10) > parseInt(d2,10))

return false;

### 4.7.2多设备的适应性

（1）实例背景：现今浏览网页端程序并不局限于电脑，各种移动设备如手机平板等正超过桌面设备，成为访问互联网的最常见终端。于是，我们不得不面对一个难题：如何才能在不同大小的设备上呈现同样的网页？

（2）解决思路：

手机的屏幕比较小，宽度通常在600像素以下；PC的屏幕宽度，一般都在1000像素以上（目前主流宽度是1366×768），有的还达到了2000像素。同样的内容，要在大小迥异的屏幕上，都呈现出满意的效果，并不是一件容易的事。

很多解决方法，是为不同的设备提供不同的网页，比如专门提供一个mobile版本，或者iPhone / iPad版本。这样做固然保证了效果，但是比较麻烦，同时要维护好几个版本，而且如果一个网站有多个portal（入口），会大大增加架构设计的复杂度。

于是，我们就设想，能不能“一次设计，普遍适用”，让同一张网页自动适应不同大小的屏幕，根据屏幕宽度，自动调整布局？

#### 4.7.2.1自适应网页设计

1、“自适应性网页设计”的概念

2010年，Ethan Marcotte提出了“自适应网页设计”（Responsive Web Design）这个名词，指可以自动识别屏幕宽度、并做出相应调整的网页设计。他制作了一个范例，里面是《福尔摩斯历险记》六个主人公的头像。

如果屏幕宽度大于1300像素，则6张图片并排在一行；如果屏幕宽度在600像素到1300像素之间，则6张图片分成两行；如果屏幕宽度在400像素到600像素之间，则导航栏移到网页头部；如果屏幕宽度在400像素以下，则6张图片分成三行。

2、允许网页宽度自动调整

首先，在网页代码的头部，加入一行viewport元标签。举例如下：

<meta name=”viewport” content=”width=device-width, initial-scale=1″ />

viewport是网页默认的宽度和高度，上面这行代码的意思是，网页宽度默认等于屏幕宽度（width=device-width），原始缩放比例（initial-scale=1）为1.0，即网页初始大小占屏幕面积的100%。

所有主流浏览器都支持这个设置，包括IE9。举例如下:

<!–[if lt IE 9]>

<script src=”http://css3-mediaqueries-js.googlecode.com/svn/trunk/css3-mediaqueries.js”></script>

<![endif]–>

3、不使用绝对宽度

由于网页会根据屏幕宽度调整布局，所以不能使用绝对宽度的布局，也不能使用具有绝对宽度的元素。这一条非常重要。

具体说，CSS代码不能指定像素宽度：

width:xxx px;

只能指定百分比宽度：

width: xx%;

或者

width:auto;

4、相对大小的字体

字体也不能使用绝对大小（px），而只能使用相对大小（em）。举例如下：

body {

font: normal 100% Helvetica, Arial, sans-serif;

}

上面的代码指定，字体大小是页面默认大小的100%，即16像素。

h1 {

font-size: 1.5em;

}

然后，h1的大小是默认大小的1.5倍，即24像素（24/16=1.5）。

small {

font-size: 0.875em;

}

small元素的大小是默认大小的0.875倍，即14像素（14/16=0.875）。

5、流动布局（fluid grid）

“流动布局”的含义是，各个区块的位置都是浮动的，不是固定不变的。举例如下：

.main {

float: right;

width: 70%;

}

.leftBar {

float: left;

width: 25%;

}

float的好处是，如果宽度太小，放不下两个元素，后面的元素会自动滚动到前面元素的下方，不会在水平方向overflow（溢出），避免了水平滚动条的出现。

6、选择加载CSS

“自适应网页设计”的核心，就是CSS3引入的Media Query模块。它的意思就是，自动探测屏幕宽度，然后加载相应的CSS文件。举例如下：

<link rel=”stylesheet” type=”text/css”

media=”screen and (max-device-width: 400px)”

href=”tinyScreen.css” />

上面的代码意思是，如果屏幕宽度小于400像素（max-device-width: 400px），就加载tinyScreen.css文件。

<link rel=”stylesheet” type=”text/css”

media=”screen and (min-width: 400px) and (max-device-width: 600px)”

href=”smallScreen.css” />

如果屏幕宽度在400像素到600像素之间，则加载smallScreen.css文件。除了用html标签加载CSS文件，还可以在现有CSS文件中加载。

@import url(“tinyScreen.css”) screen and (max-device-width: 400px);

7、CSS的@media规则

同一个CSS文件中，也可以根据不同的屏幕分辨率，选择应用不同的CSS规则。举例如下：

@media screen and (max-device-width: 400px) {

.column {

float: none;

width:auto;

}

#sidebar {

display:none;

}

}

上面的代码意思是，如果屏幕宽度小于400像素，则column块取消浮动（float:none）、宽度自动调节（width:auto），sidebar块不显示（display:none）。

8、图片的自适应（fluid image）

除了布局和文本，”自适应网页设计”还必须实现图片的自动缩放。举例如下：

这只要一行CSS代码：

img { max-width: 100%;}

这行代码对于大多数嵌入网页的视频也有效，所以可以写成：

img, object { max-width: 100%;}

老版本的IE不支持max-width，所以只好写成：

img { width: 100%; }

此外，windows平台缩放图片时，可能出现图像失真现象。这时，可以尝试使用IE的专有命令：

img { -ms-interpolation-mode: bicubic; }

或者，Ethan Marcotte的imgSizer.js。

addLoadEvent(function() {

varimgs= document.getElementByIdx\_x(“content”).getElementsByTagName\_r(“img”);

imgSizer.collate(imgs);

}); else

return true;

}

### 4.7.3交互性实现

（1）实例背景：交互性就是网页和浏览者之间的息传递的双向性。网站它不是一个“被动”的媒体。和电视，电台等媒体不一样的是，网页是需要我们用鼠标去点击的。浏览者点击连接，网页显示出需要的内容然后在传递给浏览者。这就是网页的交互性

（2）解决思路：实现网页交互性的核心技术就是数据库中数据的调用、一些脚本语言、JavaScript等。本系统在实现过程中，良好的应用以上内容，交互流畅，用户使用感和体验感优秀。

# 5．总结与展望

## 5.1系统开发总结

### 5.1.1系统设计原则

本系统设计遵循了以下设计原则：

1、稳定压倒一切

系统的稳定性压倒一切，即使牺牲性能，也要确保系统的稳定。即使扩展性不强，也要保持系统稳定。即使用户界面不友好，也不要系统不稳定。

稳定性放第一位，再先进灵活的框架，再牛的功能，如果不够稳定，都不能满足需要。如采用统一面向方面的框架，统一的异常处理、错误隔离、报警、容错和恢复机制，考虑系统的冗余度，尽可能地避免单点故障，尽可能地保证一个进程故障都不会引起系统的瘫痪。

2、可扩展性，灵活性

毋庸置疑，一个先进的系统设计具有很强的可扩展性和灵活性。设计灵活的可扩展的架构，。在设计时应具有一定的前瞻性，充分考虑系统维护的可行性；并充分考虑如何大幅度提高业务处理的响应速度以及动态表单统计汇总的速度和精度。

3、可用性，用户友好性

设计以人为本，到头来还是给人用的。关注用户的行为和体验。我方倾听和收集用户对系统使用的需求、体验或不满，并进行艺术化的设计。

4、可维护性，可管理性

可维护性包括代码的可理解性，可测试性，可修改性和系统的可移植性。

可维护性设计应该采用灵活架构，采用复用的设计方法，减少相互之间的依赖项，采用成熟的工业应用级的产品和框架，采用代码审查机制等。可维护性最直接的体现是良好的软件结构和完整正确的文档体系。可重用性是可维护性的基本属性，最大限度地重用现存软件是软件维护方法学的重要思想原则。

为实现运行可管理性，系统采用了参数化的可配置的设计，模块间通信尽可能简单，

### 5.1.2系统设计模式

本系统设计运用了如下设计模式：

5.1.2.1工厂方法(Factory Method)模式

工厂方法(Factory Method)模式的意义是定义一个创建产品对象的工厂接口，将实际创建工作推迟到子类当中。核心工厂类不再负责产品的创建，这样核心类成为一个抽象工厂角色，仅负责具体工厂子类必须实现的接口，这样进一步抽象化的好处是使得工厂方法模式可以使系统在不修改具体工厂角色的情况下引进新的产品。

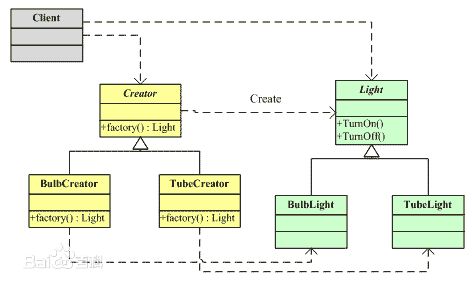


图5.1.2.1 工厂方法模式图

在本系统中，Hibernate架构里运用了工厂模式。

package com.zhangying.hb;

import org.hibernate.HibernateException;

import org.hibernate.Session;

import org.hibernate.cfg.Configuration;

/\*\*

\* Configures and provides access to Hibernate sessions, tied to the

\* current thread of execution. Follows the Thread Local Session

\* pattern, see {@link http://hibernate.org/42.html }.

\*/

public class HibernateSessionFactory {

/\*\*

\* Location of hibernate.cfg.xml file.

\* Location should be on the classpath as Hibernate uses

\* #resourceAsStream style lookup for its configuration file.

\* The default classpath location of the hibernate config file is

\* in the default package. Use #setConfigFile() to update

\* the location of the configuration file for the current session.

\*/

private static String CONFIG\_FILE\_LOCATION = "/hibernate.cfg.xml";

private static final ThreadLocal<Session> threadLocal = new ThreadLocal<Session>();

private static Configuration configuration = new Configuration();

private static org.hibernate.SessionFactory sessionFactory;

private static String configFile = CONFIG\_FILE\_LOCATION;

static {

try {

configuration.configure(configFile);

sessionFactory = configuration.buildSessionFactory();

} catch (Exception e) {

System.err.println("%%%% Error Creating SessionFactory %%%%");

e.printStackTrace();

}

}

private HibernateSessionFactory() {

}

/\*\*

\* Returns the ThreadLocal Session instance. Lazy initialize

\* the <code>SessionFactory</code> if needed.

\*

\* @return Session

\* @throws HibernateException

\*/

public static Session getSession() throws HibernateException {

Session session = (Session) threadLocal.get();

if (session == null || !session.isOpen()) {

if (sessionFactory == null) {

rebuildSessionFactory();

}

session = (sessionFactory != null) ? sessionFactory.openSession(): null;

threadLocal.set(session);

}

return session;

}

/\*\*

\* Rebuild hibernate session factory

\*

\*/

public static void rebuildSessionFactory() {

try {

configuration.configure(configFile);

sessionFactory = configuration.buildSessionFactory();

} catch (Exception e) {

System.err.println("%%%% Error Creating SessionFactory %%%%");

e.printStackTrace();

}

}

/\*\*

\* Close the single hibernate session instance.

\*

\* @throws HibernateException

\*/

public static void closeSession() throws HibernateException {

Session session = (Session) threadLocal.get();

threadLocal.set(null);

if (session != null) {

session.close();

}

}

/\*\*

\* return session factory

\*

\*/

public static org.hibernate.SessionFactory getSessionFactory() {

return sessionFactory;

}

/\*\*

\* return session factory

\*

\* session factory will be rebuilded in the next call

\*/

public static void setConfigFile(String configFile) {

HibernateSessionFactory.configFile = configFile;

sessionFactory = null;

}

/\*\*

\* return hibernate configuration

\*

\*/

public static Configuration getConfiguration() {

return configuration;

}

}

5.1.2.2装饰者模式（Decorator Pattern）

装饰者模式（Decorator Pattern）属于结构型设计模式。装饰者模式是在不必改变原类文件和使用继承的情况下，动态地扩展一个对象的功能。它是通过创建一个包装对象，也就是装饰来包裹真实的对象。

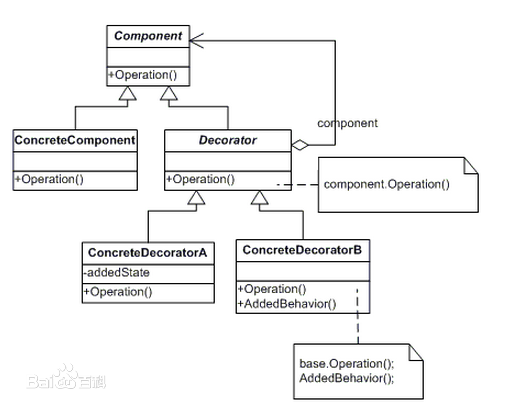


图5.1.2.2 装饰者模式图

本系统中，ActionMapping通过path属性来和一个指定的URI相关联。HTTP请求到来时，总控ActionServlet通过path属性来选择对应的ActionMapping对象。全部的ActionMapping对象都是存放在一个ActionMapping集合中。为了更好地使用Action，ActionMapping对象被用来作为Action对象的修饰符。它将一个或者多个URI交给一个Action，同时可以根据哪个URI被调用来传递不同的配置信息给Action。即ActionMapping可以动态赋予一个特定Action对象一些附加的要求和新功能，这些就是装饰者模式的要求。该设计模式相对于子类化方法扩展功能而言，具有更高的灵活性。ActionMapping对象是通过配置文件struts-config.xml来创建的。

5.1.2.3单例模式（Singleton Pattern）

单例模式（Singleton Pattern）是一种常用的软件设计模式。在它的核心结构中只包含一个被称为单例类的特殊类。通过单例模式可以保证系统中一个类只有一个实例而且该实例易于外界访问，从而方便对实例个数的控制并节约系统资源。如果希望在系统中某个类的对象只能存在一个，单例模式是最好的解决方案。

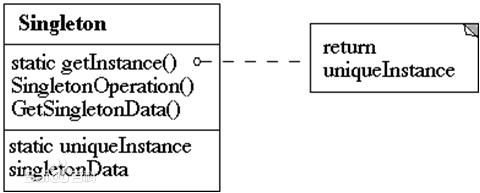


图5.1.2.3 单例模式图

在Java这样的多线程环境中，使用一个对象的多个拷贝并没有什么特别的好处，ActionServlet就是一个很好地例子，它是一个多线程对象，可以处理任意数量的请求而不需要创造新对象。本系统同样在Action对象上使用这一原则：一个应用程序仅仅实例化一个Action的实例，该实例会生成多个线程来处理后面的请求。这样做不仅保证使用的资源少，而且提高了应用程序的性能，同时我们还保证了Action类是线程安全的。

5.1.2.4MVC模式

[MVC](http://baike.baidu.com/view/31.htm)全名是Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC被独特的发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。

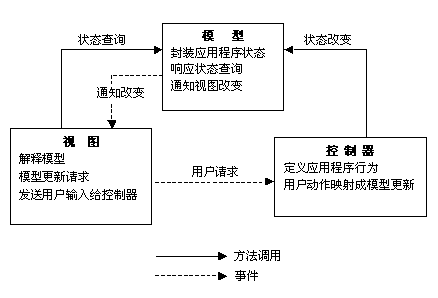


图5.1.2.4 MVC模式图

在本系统中运用的Struts框架是MVC模式的典型代表。Struts有一组相互协作的类（组件）、[Servlet](http://baike.baidu.com/view/25169.htm)以及jsp tag lib组成。基于struts构架的web应用程序基本上符合JSP Model2的设计标准，可以说是MVC设计模式的一种变化类型。Struts有其自己的控制器（[Controller](http://baike.baidu.com/view/1805013.htm)），同时整合了其他的一些技术去实现模型层（[Model](http://baike.baidu.com/view/2159664.htm)）和视图层（View）。在模型层，Struts可以很容易的与数据访问技术相结合，本系统中使用了与数据库访问接口Dao。在视图层，Struts能够与JSP, Velocity Templates,XSL等等这些表示层组件相结合，本系统采用了与JSP结合。

5.1.2.5责任链模式（Chain of responsibility）

责任链模式（Chain of responsibility）是一种对象行为型设计模式。在责任链模式里，很多对象由每一个对象对其下家的引用而连接起来形成一条链。请求在这个链上传递，直到链上的某一个对象决定处理此请求。发出这个请求的客户端并不知道链上的哪一个对象最终处理这个请求，这使得系统可以在不影响客户端的情况下动态地重新组织和分配责任。

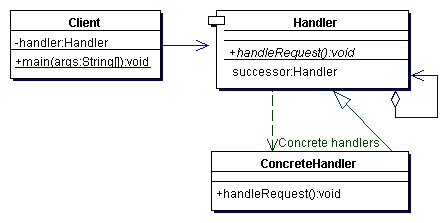


图5.1.2.5 责任链模式图

本系统中使用的struts架构隐含了责任链模式。在Struts2中，其拦截器结构的设计，是一个典型的责任链模式的应用。首先将整个执行划分成若干相同类型的元素，每个元素具备不同的逻辑责任，并将他们纳入到一个链式的[数据结构](http://lib.csdn.net/base/31)中（我们可以把堆栈结构也看作是一个递归的链式结构），而每个元素又有责任负责链式结构中下一个元素的执行调用。这样的设计，从代码重构的角度来看，实际上是将一个复杂的系统，分而治之，从而使得每个部分的逻辑能够高度重用并具备高度可扩展性。

5.1.2.6命令模式（Command pattern）

命令（Command Pattern）模式属于对象的行为模式。命令模式又称为行动（Action）模式或交易（Transaction）模式。命令模式把一个请求或者操作封装到一个对象中。命令模式允许系统使用不同的请求把客户端参数化，对请求排队或者记录请求日志，可以提供命令的撤销和恢复功能。

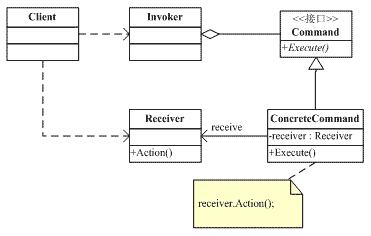


图5.1.2.6 命令模式图

命令模式是对命令的封装。命令模式把发出命令的责任和执行命令的责任分割开，委派给不同的对象。每一个命令都是一个操作：请求的一方发出请求要求执行一个操作；接收的一方收到请求，并执行操作。命令模式允许请求的一方和接收的一方独立开来，使得请求的一方不必知道接收请求的一方的接口，更不必知道请求是怎么被接收，以及操作是否被执行、何时被执行，以及是怎么被执行的。

本系统中，用户定义了Action Mapping在struts.xml里，在web container启动时，ConfigurationManager 就装载了struts.xml，当Request过来时，Struts Framework会根据当前的URL去找ConfigurationManager所对应的concreteCommand/Action,  然后这个Action会被set到ActionInvocation。虽然没有严格的按照其经典模型实现，但运用了command的思想，我们以损失一点点接口作为契约带来好处为代价，降低了用户实现对框架的依赖，增加了程序的可测性。通过配置文件+Reflection，我们可以做到在Runtime的时候给ActionInvocation注入不同的Action的目的。

package com.zhangying.tool;

import java.io.IOException;

import java.io.UnsupportedEncodingException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class ActionServlet extends org.apache.struts.action.ActionServlet {

protected void process(HttpServletRequest httpservletrequest,HttpServletResponse httpservletresponse) throws IOException,ServletException {

setCharset(httpservletrequest);//设置编码

super.process(httpservletrequest, httpservletresponse);//调用父类处理方法

}

private void setCharset(HttpServletRequest httpservletrequest) {

String s = getServletConfig().getInitParameter("charset");//获得初始化参数

if (s == null || s.length() == 0) {//如果参数为空，或者为null

s="GBK"; //设置参数值为GBK

}

try {

httpservletrequest.setCharacterEncoding(s);//设置请求编码格式

} catch (UnsupportedEncodingException unsupportedencodingexception) {

log("set character encoding error", unsupportedencodingexception);

}

}

}

### 5.1.3系统设计难点

**1、系统开发语言的选择**

根据需求分析确定开发时使用到的具体的技术是非常关键的一步。因为在大二的时候学过C#编程，大三学了Java编程，并且在课上听杨老师说这两种语言都可以，那么这两种语言哪一个能开发出更符合用户需求的产品？在开发前，经过讨论与分析并结合在网上的资料，我们确定了要把人事信息管理系统做成一个B/S架构，因此我们选择了Java。

**2、系统框架的搭建**

系统框架的搭建是系统最后能够实现的一个基石。在最开始选择了Java语言进行开发的基础上，怎样才能做到让整个系统架构清晰明了？怎样才能用最少的代码写出用户需求的系统？这时，就想起了金老师和杨老师上课讲过的MVC模式，这个模式把整个系统划分成三层甚至是多层，使系统的显示层、业务逻辑层和数据层分离。当时是想到了这个框架，可是到了具体的应用上并不知道要怎么开始，先做什么再做什么。所以系统开发的最开始是很艰难的。于是，我们利用课余时间到图书馆借阅相关书籍，并在网上查阅资料，先做了一个利用Struts框架实现的登录模块，这为其他模块的实现做了一个铺垫。

**3、数据库连接及操作**

由于我们所做的人事管理信息系统需要长期地保存数据，并且前端需要对数据进行查询、添加、修改、删除等操作，所以连接数据库是十分关键并且必要的。据我所知，JDBC一直是系统开发的难点，结合系统框架的搭建，对数据库的操作我们选择了Hibernate，可是我们只在金大卫老师的课上听过一点点，也不知道从哪里下手，可是利用传统的方式，代码又显得冗余复杂，于是充分利用图书馆的资源，系统地学习了一下这个框架包含的内容及用法，将系统中用到的对数据库的操作封装起来，用Hibernate提供的接口去实现，这也大大降低了代码量。

**4、对框架的配置文件的理解**

框架的使用相较不用框架还是有很多好处的，但是不用框架比框架理解起来更加容易，所以在前后端代码编写完成之后，怎么能够使得二者能够合二为一，成为一个整体的东西是至关重要的，这时配置文件就成为了关键点，怎样使得这个页面能够进行那个业务逻辑的判断跳转到另一个页面。既然选择了用框架，那么我们就要坚持弄清楚框架到底怎么通过配置文件来实现整个系统的功能，我们通过学习，理解了配置文件，并且也算是成功地实现了这个系统的全部功能。

**5、编码问题**

刚开始在前端页面插入数据库后再显示在页面上英文是可以的，但是中文却是乱码，我们以为由于JSP和数据库的编码不一致，所以就检查了所有JSP页面的编码和数据库的编码，发现都是UTF-8，我们选择分步检查，先看数据在添加到数据库中时是否能正常显示，于是发现也是乱码，所以就明确了问题出在连接数据库上，于是我检查连接数据库的URL发现确实少了一句关于编码的描述，最终添加上之后中文数据显示是正常的。

**6、表单得不到输入值的问题**

在整个系统实现的过程中，出现过好多次在页面显示的不是最开始输入的值，有时甚至是空的问题，首先是检查了一下前端代码是否是正确且合法的，其次检查Struts框架中Action Form中得到的是不是在前端页面填写的值，然后检查Action里用到的DAO是否有问题，一步一步检查下来发现有一些是前端代码中出现了错误，有一些是DAO中数据类型没有进行转换，有一些是Action中变量重复的问题，最后一一按照正确写法进行修改，解决了这个问题。

**7、JAR包的冲突问题**

由于本系统不仅使用了Struts框架还应用了Hibernate框架，在系统整合完成后运行时报了一个500的错误，当时觉得各个步骤都是合理的，出现错误也是很疑惑，进行了很长一段时间的错误查找，后来经过百度才发现Struts的antlr-2.7.2.jar和Hibernate中的antlr-2.7.6.jar冲突了，于是按照百度上的解决办法将低版本的jar删除之后就可以正常运行了。

**8、系统功能的实现问题**

由于在调研时，麻麻鱼府的老板或是经理对其店铺的绩效管理以及薪金的发放很是敏感，不愿说出具体的考核方法，所以我们在实现这个系统的时候为其提供了一个输入框，能够让其将按照考核方法之后的数据填入其中并进行记录，虽然是迫于现实情况的无奈之举，但是相关功能确实已实现，这一点上我们都认为处理的很恰当。

## 5.2系统开发展望

### 5.2.1系统不足及改进方向

本系统由于不可抗因素即一些客观原因，导致系统还存在一些不足，在日后兄推广的过程中，我们会不断完善本系统，想出一些更好的方法来进行适配，解现有难题和痛点，朝着更好的方向进行系统改进。

1. **功能新增**

在后续过程中，结合现有问题，本小组会进行多商家调研，不局限于某一餐饮业，切实了解用户需求，进行功能新增。由于代码的重用性非常高，所以功能新增的任务量不是特别大，这也是当初我们在进行系统设计的时候考虑到的问题，问日后系统的运维提供了接口，方便后续功能的增改。

1. **绩效管理完善**

由于不可抗因素，本小组没有拿到绩效考核方法（属商业机密范畴），导致改模块功能不是特别完善，但也可以达到我方调研用户需求。在后续过程中，本小组多商家调研后，拿到相关数据，对该模块功能和界面进行进一步完善，满足用户需求。

1. **权限设置**

结合实际情况的需要，对超级管理员和系统开发人员的使用权限进行进一步完善，方便人员管理和权限的分配。

1. **合理适配**

在现有的基础上，微调CSS文件中的内容，让本系统更加完美的适配PC端和移动端，包括电脑、平板、手机等。

### 5.2.2感悟

在一学期的学习过程中，《信息系统分析与设计》这门课程的设计与实现贯穿了我们小组一整个学期。每周都在写文档该文档，后期就是写代码改代码，改文档，每个人都受益匪浅。

老师对待这门课程也是非常认真，认真程度也是我大学快三年来仅见的。课上为我们讲解知识，分享师兄师姐的作业让我们累积经验；课下给我们分配任务，从系统分析开始完善文档，到系统设计的代码撰写，最后展示成果。老师会认真查看我们的文档并提出细节上的不足，用心批注，并提出指导意见，一直在后方为我们保驾护航。

首先要进行系统的规划和选择，接着就是系统的设计，包括设计人机界面和数据库设计。最后一步就是系统的实现与运行。

在做信息系统分析与设计的时候，通过学习，我们能更清晰地了解系统所要实现的功能。并经过不断讨论后逐步完善，一点点修正。在系统开发过程中，我们明白了理论与实际相结合的重要性，光有理论知识是远远不够的，只有把所学的理论知识与实践相结合起来，从理论中得出结论，提高自己的实际动手能力和独立思考的能力，才能解决问题。我们在编程过程中也遇到过很多困难与瓶颈，比如说对以前所学过的知识掌握得不够牢固等，但最后都一一克服，最终有了成果。

我们的人事管理系统还十分稚嫩，还有很多功能有待完善，也还存在不足之处，在未来我们也会添加进更多的特色模块和新功能，可以让系统满足更多用户的需求。首先人事管理系统应留给用户更大的空间，并提供更加灵活的应用程序接口，不同用户根据自己业务逻辑需求进行第二次开发，为每一个用户提供量身定做的解决方案。当用户的经营环境和业务模式改变时，系统也可以轻松的被加以改装，适应新的要求。

一个友好的用户界面的意义不仅在于它能高速高效的完成用户要求的操作流程，更重要的是它能在当今和谐社会的引导潮流下体现出一种人机和谐的良好氛围。一直以来，人们在开发某种技术或设备时，更多注重的是其多功能性和其软硬件的开发，而较少从人的主观意志去改善和提高。“以人为本”突出的就是一个人字，人性化的设计更是从本质上方便改善了我们的生活，是一个社会进步的本质体现，这是我们发展科学技术的最终目标。因此，信息系统，在做好的基础之上，下一个步骤是要做精。简而言之，我们要朝着人事管理系统“人性化”和“定制化”的方向而努力。

我们要借助于计算机及网络技术，对信息资源进行收集、整理、分类、分析，以满足用户对信息的需求。总而言之，要让人事管理系统渗入到各行各业的每一个领域，充分挖掘其潜在的可塑之处，模块化，人性化，个性化。分别从系统本身，使用者和开发者三个角度深入分析人事管理系统今后的发展方向，并向着这个方向努力。

本次开发这个系统。从分析到设计到编码实现，小组成员学习和收获了很多知识，清楚了系统开发的流程，同时也意识到了我们的能力有所欠缺。在接下来的日子里，不能懈怠，仍要不停的学习，多完善自己的能力，同时，对自己未来知识体系架构也有了新的规划，如未来努力的方向和查漏补缺的重点等。

再次感谢老师的付出和努力，您辛苦了。

## 参考文献

[1] 陶永明，企业人事管理系统设计. 东北财经大学，2011

[2] 郑環，企业人事管理系统的设计与实现. 电子科技大学，2010

[3] 陈燕，重庆大学人事管理信息系统的设计与开饭. 重庆大学，2002

[4] 蔡琴，餐饮管理系统的设计与开发. 电子科技大学，2007

[5] 戚云龙，大中型餐厅综合管理系统设计与实现. 大连理工大学，2012

[6] 许秀芳，徐森，基于Web的餐饮连锁企业工资人事管理系统设计. 中国电子学会，2016

[7] 李中明，大型连锁餐饮企业人力资源管理系统的研究. 上海财经大学，2010

[8] 周鹏飞，基于嵌入式的餐饮系统设计与实现. 电子科技大学，2014

[9] 刘学治，从“人本管理”谈餐饮业的人事管理. 四川烹饪，1997

[10] 吴姣，餐饮零售管理大提速——天维思专注供应链实时预测与计划协调. 物流时代，2016

[11] 邢美东，基于数据挖掘的餐饮管理系统的设计与实现. 东北大学，2008

[12] 董浩，内蒙古聚祥餐饮管理有限公司薪酬制度设计与研究. 内蒙古大学，2011

[13] 许丹，基于供应链的餐饮业管理系统. 南京理工大学，2007

[14] 瞿建，二连浩特酒店管理系统设计与实现. 大连理工大学，2012

[15] 杨超超，浙江省连锁餐饮企业人力资源管理存在问题及对策研究. 中国管理信息化，2016

[16] 王洪海，浅析我国餐饮管理现状及对策. 商业故事，2016

[17] 范海明，浅析管理信息系统在酒店中的应用. 电子制作，2015

[18] 付佳，人事管理系统的设计与实现. 大连海事大学，2015

[19] 巫晓聪，基于WEB的企业人事信息管理系统的设计与实现. 电子科技大学，2013

[20] 赵炳森，基于Hibernate的人事管理系统. 华东师范大学，2009

[21] 陈卫琴，基于Struts和Hibernate的人事管理系统的分析与设计. 南京理工大学，2007

[22] 史洪源，企业人事管理系统的设计与实现. 吉林大学，2014

[23] 李相会，人事管理系统的设计与实现. 电子科技大学，2013

## 代码：

共12875行。

/HRmanagement/src/com.zhangying.action/EducateAction.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.EducateDao;

import com.zhangying.po.Educate;

/\*\*

\* @author f(x)函数

\*

\*/

public class EducateAction extends Action {

private EducateDao dao=new EducateDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nEducateAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listeducate".equals(action)){

return listEducate(mapping,form,request,response);

}else if("addeducate".equals(action)){

return addEducate(mapping,form,request,response);

}else if("deleteeducate".equals(action)){

return deleteEducate(mapping,form,request,response);

}else if("detaileducate".equals(action)){

return detailEducate(mapping,form,request,response);

}else if("updateeducate".equals(action)){

return updateEducate(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updateEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

EducateForm educateForm=(EducateForm)form;

Educate e=educateForm.populate();

dao.updateEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Educate e = dao.loadEducate(id.longValue());

System.out.println(e.getName());

request.setAttribute("educate",e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deleteEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id").toString());

Educate e=new Educate();

e.setId(id);

dao.deleteEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

EducateForm educateForm=(EducateForm)form;

Educate e=educateForm.populate();

dao.addEducate(e);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listEducate(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

String educate=request.getParameter("educate");

if(educate==null||"".equals(educate)||"0".equals(educate)){

request.setAttribute("list",dao.listEducate(new Byte("0").byteValue()));

}else{

request.setAttribute("list",dao.listEducate(new Byte("1").byteValue()));

}

return mapping.findForward("success");

}

}

/HRmanagement/src/com.zhangying.action/EducateForm.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionMapping;

import com.zhangying.po.Educate;

import com.zhangying.tool.DateUtil;

/\*\*

\* @author f(x)函数

\*

\*/

public class EducateForm extends ActionForm {

/\*\* identifier field \*/

private Long id;

/\*\* nullable persistent field \*/

private String name;

/\*\* nullable persistent field \*/

private String purpose;

/\*\* nullable persistent field \*/

private String begintime;

/\*\* nullable persistent field \*/

private String endtime;

/\*\* nullable persistent field \*/

private String datum;

/\*\* nullable persistent field \*/

private String teacher;

/\*\* nullable persistent field \*/

private String student;

/\*\* nullable persistent field \*/

private String createtime;

/\*\* nullable persistent field \*/

private String effect;

/\*\* nullable persistent field \*/

private String summarize;

private Byte educate;

/\*\* full constructor \*/

public void reset(ActionMapping arg0, HttpServletRequest arg1) {

this.name = null;

this.purpose = null;

this.begintime = null;

this.endtime = null;

this.datum = null;

this.teacher = null;

this.student = null;

this.createtime = null;

this.educate=null;

this.effect = null;

this.summarize = null;

}

public ActionErrors validate(ActionMapping arg0, HttpServletRequest arg1) {

return null;

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public String getPurpose() {

return this.purpose;

}

public void setPurpose(String purpose) {

this.purpose = purpose;

}

public String getBegintime() {

return this.begintime;

}

public void setBegintime(String begintime) {

this.begintime = begintime;

}

public String getEndtime() {

return this.endtime;

}

public void setEndtime(String endtime) {

this.endtime = endtime;

}

public String getDatum() {

return this.datum;

}

public void setDatum(String datum) {

this.datum = datum;

}

public String getTeacher() {

return this.teacher;

}

public void setTeacher(String teacher) {

this.teacher = teacher;

}

public String getStudent() {

return this.student;

}

public void setStudent(String student) {

this.student = student;

}

public String getCreatetime() {

return this.createtime;

}

public void setCreatetime(String createtime) {

this.createtime = createtime;

}

public String getEffect() {

return this.effect;

}

public void setEffect(String effect) {

this.effect = effect;

}

public String getSummarize() {

return this.summarize;

}

public void setSummarize(String summarize) {

this.summarize = summarize;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Educate] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" begintime = " + this.begintime+ ";\n");

toStr.append(" endtime = " + this.endtime+ ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" purpose = " + this.purpose + ";\n");

toStr.append(" datum = " + this.datum + ";\n");

toStr.append(" teacher = " + this.teacher + ";\n");

toStr.append(" student = " + this.student + ";\n");

toStr.append(" educate = " + this.educate+ ";\n");

toStr.append(" effect = " + this.effect + ";\n");

toStr.append(" summarize = " + this.summarize + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

public Educate populate(){

Educate e=new Educate();

e.setBegintime(DateUtil.parseToDate(this.getBegintime(),DateUtil.yyyyMMdd));

e.setCreatetime(DateUtil.parseToDate(this.getCreatetime(),DateUtil.yyyyMMddHHmmss));

e.setDatum(this.getDatum());

e.setEffect(this.getEffect());

e.setEndtime(DateUtil.parseToDate(this.getEndtime(),DateUtil.yyyyMMdd));

e.setId(this.getId());

e.setName(this.getName());

e.setPurpose(this.getPurpose());

e.setStudent(this.getStudent());

e.setSummarize(this.getSummarize());

e.setEducate(this.getEducate());

e.setTeacher(this.getTeacher());

return e;

}

/\*\*

\* @return Returns the educate.

\*/

public Byte getEducate() {

return educate;

}

/\*\*

\* @param educate The educate to set.

\*/

public void setEducate(Byte educate) {

this.educate = educate;

}

}

/HRmanagement/src/com.zhangying.action/InstitutionAction.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.InstitutionDao;

import com.zhangying.po.Institution;

/\*\*

\* @author f(x)函数

\*

\*/

public class InstitutionAction extends Action {

private InstitutionDao dao=new InstitutionDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nInstitutionAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listinstitution".equals(action)){

return listInstitution(mapping,form,request,response);

}else if("addinstitution".equals(action)){

return addInstitution(mapping,form,request,response);

}else if("updateinstitution".equals(action)){

return updateInstitution(mapping,form,request,response);

}else if("deleteinstitution".equals(action)){

return deleteInstitution(mapping,form,request,response);

}else if("detailinstitution".equals(action)){

return detailInstitution(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Institution i=dao.loadInstitution(id.longValue());

request.setAttribute("institution",i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deleteInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Institution i=new Institution();

i.setId(id);

dao.deleteInstitution(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updateInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

InstitutionForm institutionForm=(InstitutionForm)form;

Institution i=institutionForm.populate();

dao.updateInstitution(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

InstitutionForm institutionForm=(InstitutionForm)form;

Institution i=institutionForm.populate();

dao.addInstitution(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listInstitution(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

request.setAttribute("list",dao.listInstitution());

return mapping.findForward("success");

}

}

/HRmanagement/src/com.zhangying.action/InstitutionForm.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionMapping;

import com.zhangying.po.Institution;

import com.zhangying.tool.DateUtil;

/\*\*

\* @author f(x)函数

\*

\*/

public class InstitutionForm extends ActionForm {

/\*\* identifier field \*/

private Long id;

/\*\* nullable persistent field \*/

private String name;

/\*\* nullable persistent field \*/

private String reason;

/\*\* nullable persistent field \*/

private String explain;

/\*\* nullable persistent field \*/

private String createtime;

/\*\* full constructor \*/

public void reset(ActionMapping arg0, HttpServletRequest arg1) {

this.name = null;

this.reason = null;

this.explain = null;

this.createtime = null;

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public String getReason() {

return this.reason;

}

public void setReason(String reason) {

this.reason = reason;

}

public String getExplain() {

return this.explain;

}

public void setExplain(String explain) {

this.explain = explain;

}

public String getCreatetime() {

return this.createtime;

}

public void setCreatetime(String createtime) {

this.createtime = createtime;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Institution] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" reason = " + this.reason + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" explain = " + this.explain + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

public Institution populate(){

Institution i=new Institution();

i.setCreatetime(DateUtil.parseToDate(this.getCreatetime(),DateUtil.yyyyMMddHHmmss));

i.setExplain(this.getExplain());

i.setId(this.getId());

i.setName(this.getName());

i.setReason(this.getReason());

return i;

}

public ActionErrors validate(ActionMapping arg0, HttpServletRequest arg1) {

return null;

}

}

/HRmanagement/src/com.zhangying.action/JobAction.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.JobDao;

import com.zhangying.po.Job;

public class JobAction extends Action {

private JobDao dao=new JobDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nJobAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listjob".equals(action)){

return listJob(mapping,form,request,response);

}else if("addjob".equals(action)){

return addJob(mapping,form,request,response);

}else if("deletejob".equals(action)){

return deleteJob(mapping,form,request,response);

}else if("updatejob".equals(action)){

return updateJob(mapping,form,request,response);

}else if("detailjob".equals(action)){

return detailjob(mapping,form,request,response);

}

return mapping.findForward("error");

}

private ActionForward detailjob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//获得参数id

//System.out.println("id="+id);

Job j=dao.loadJob(id.longValue());//加载该ID对应的应聘信息

request.setAttribute("job",j);//将应聘信息对象设置到request范围

return mapping.findForward("success");

}

private ActionForward updateJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

// System.out.println("111111111111112221111");

//System.out.println("form="+form);

JobForm jobform=(JobForm)form;

//System.out.println("jobform="+jobform);

Job j=jobform.populate();

j.setJob(jobform.getJob());

//System.out.println("j="+j);

dao.updateJob(j);

return mapping.findForward("success");

}

private ActionForward deleteJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//接受id参数

Job j=new Job();//新建一个Job对象

j.setId(id);//设置其id属性

dao.deleteJob(j);//调用dao对象完成Job对象的删除

return mapping.findForward("success");

}

private ActionForward addJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

JobForm jobform=(JobForm)form;//获得JobForm

Job job=jobform.populate();//获得表单提交的应聘信息

dao.addJob(job);//调用DAO完成应聘信息的保存

return mapping.findForward("success");//跳转到success逻辑视图

}

private ActionForward listJob(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

String isstock=request.getParameter("isstock");//获得isstock参数

System.out.println("isstock="+isstock);

if(isstock==null||"".equals(isstock)){//如果isstock为null或者为空

request.setAttribute("list",

dao.listJob(new Byte("0").byteValue()));//查询所有没入库的应聘信息

}else{

request.setAttribute("list",

dao.listJob(new Byte("1").byteValue()));//查询所有入库的应聘信息

}

return mapping.findForward("success");

}

}

/HRmanagement/src/com.zhangying.action/JobForm.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionMapping;

import com.zhangying.po.Job;

import com.zhangying.tool.DateUtil;

/\*\*

\* @author f(x)函数

\*

\*/

public class JobForm extends ActionForm {

/\*\* identifier field \*/

private Long id;

/\*\* nullable persistent field \*/

private String name;

/\*\* nullable persistent field \*/

private Byte sex;

/\*\* nullable persistent field \*/

private Integer age;

/\*\* nullable persistent field \*/

private String job;

/\*\* nullable persistent field \*/

private String specialty;

/\*\* nullable persistent field \*/

private String experience;

/\*\* nullable persistent field \*/

private String studyeffort;

/\*\* nullable persistent field \*/

private String school;

/\*\* nullable persistent field \*/

private String tel;

/\*\* nullable persistent field \*/

private String content;

/\*\* nullable persistent field \*/

private Byte isstock;

private String createtime;

private String email;

private Byte health;

/\*\*

\* @return Returns the email.

\*/

public String getEmail() {

return email;

}

/\*\*

\* @param email The email to set.

\*/

public void setEmail(String email) {

this.email = email;

}

/\*\* full constructor \*/

public void reset(ActionMapping arg0, HttpServletRequest arg1) {

this.id=null;

this.name = null;

this.sex = null;

this.health=null;

this.age = null;

this.job = null;

this.specialty = null;

this.experience = null;

this.studyeffort = null;

this.school = null;

this.tel = null;

this.createtime=null;

this.email=null;

this.content = null;

this.isstock = null;

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public Byte getSex() {

return this.sex;

}

public void setHealth(Byte health) {

this.health = health;

}

public Byte getHealth() {

return this.health;

}

public void setSex(Byte sex) {

this.sex = sex;

}

public Integer getAge() {

return this.age;

}

public void setAge(Integer age) {

this.age = age;

}

public String getJob() {

return this.job;

}

public void setJob(String job) {

this.job = job;

}

public String getSpecialty() {

return this.specialty;

}

public void setSpecialty(String specialty) {

this.specialty = specialty;

}

public String getExperience() {

return this.experience;

}

public void setExperience(String experience) {

this.experience = experience;

}

public String getStudyeffort() {

return this.studyeffort;

}

public void setStudyeffort(String studyeffort) {

this.studyeffort = studyeffort;

}

public String getSchool() {

return this.school;

}

public void setSchool(String school) {

this.school = school;

}

public String getTel() {

return this.tel;

}

public void setTel(String tel) {

this.tel = tel;

}

public String getContent() {

return this.content;

}

public void setContent(String content) {

this.content = content;

}

public Byte getIsstock() {

return this.isstock;

}

public void setIsstock(Byte isstock) {

this.isstock = isstock;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[JobForm] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" sex = " + this.sex + ";\n");

toStr.append(" age = " + this.age + ";\n");

toStr.append(" job = " + this.job + ";\n");

toStr.append(" specialty = " + this.specialty + ";\n");

toStr.append(" experience = " + this.experience + ";\n");

toStr.append(" studyeffort = " + this.studyeffort + ";\n");

toStr.append(" school = " + this.school + ";\n");

toStr.append(" tel = " + this.tel + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" email= " + this.email+ ";\n");

toStr.append(" content = " + this.content+ ";\n");

toStr.append(" isstock = " + this.isstock + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

public ActionErrors validate(ActionMapping arg0, HttpServletRequest arg1) {

return null;

}

public Job populate(){

Job j=new Job();

j.setAge(this.getAge());

j.setContent(this.getContent());

j.setCreatetime(DateUtil.parseToDate(this.getCreatetime(),DateUtil.yyyyMMddHHmmss));

j.setExperience(this.getExperience());

j.setId(this.getId());

j.setIsstock(this.getIsstock());

j.setJob(this.getJob());

j.setName(this.getName());

j.setSchool(this.getSchool());

j.setSex(this.getSex());

j.setHealth(this.getHealth());

j.setSpecialty(this.getSpecialty());

j.setStudyeffort(this.getStudyeffort());

j.setTel(this.getTel());

j.setEmail(this.getEmail());

return j;

}

/\*\*

\* @return Returns the createtime.

\*/

public String getCreatetime() {

return createtime;

}

/\*\*

\* @param createtime The createtime to set.

\*/

public void setCreatetime(String createtime) {

this.createtime = createtime;

}

}

/HRmanagement/src/com.zhangying.action/PerformanceAction.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.PerformanceDao;

import com.zhangying.po.Performance;

/\*\*

\* @author f(x)函数

\*

\*/

public class PerformanceAction extends Action {

private PerformanceDao dao=new PerformanceDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nPerformanceAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("listperformance".equals(action)){

return listPerformance(mapping,form,request,response);

}else if("addperformance".equals(action)){

return addPerformance(mapping,form,request,response);

}else if("updateperformance".equals(action)){

return updatePerformance(mapping,form,request,response);

}else if("deleteperformance".equals(action)){

return deletePerformance(mapping,form,request,response);

}else if("detailperformance".equals(action)){

return detailPerformance(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailPerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Performance i=dao.loadPerformance(id.longValue());

request.setAttribute("performance",i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deletePerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Performance i=new Performance();

i.setId(id);

dao.deletePerformance(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updatePerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

PerformanceForm performance=(PerformanceForm)form;

Performance i=performance.populate();

dao.updatePerformance(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addPerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

PerformanceForm PerformanceForm=(PerformanceForm)form;

Performance i=PerformanceForm.populate();

dao.addPerformance(i);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listPerformance(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

request.setAttribute("list",dao.listPerformance());

return mapping.findForward("success");

}

}

/HRmanagement/src/com.zhangying.action/PerformanceForm.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionMapping;

import com.zhangying.po.Performance;

import com.zhangying.tool.DateUtil;

/\*\*

\* @author f(x)函数

\*

\*/

public class PerformanceForm extends ActionForm {

/\*\* identifier field \*/

private Long id;

/\*\* nullable persistent field \*/

private float score;

/\*\* nullable persistent field \*/

private float addmoney;

/\*\* nullable persistent field \*/

private String content;

/\*\* nullable persistent field \*/

private String createtime;

/\*\* full constructor \*/

public void reset(ActionMapping arg0, HttpServletRequest arg1) {

this.score = 0;

this.addmoney = 0;

this.content = null;

this.createtime = null;

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public float getScore() {

return score;

}

public void setScore(float score) {

this.score = score;

}

public float getAddmoney() {

return addmoney;

}

public void setAddmoney(float addmoney) {

this.addmoney = addmoney;

}

public String getContent() {

return content;

}

public void setContent(String content) {

this.content = content;

}

public String getCreatetime() {

return createtime;

}

public void setCreatetime(String createtime) {

this.createtime = createtime;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Performence] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" score = " + this.score + ";\n");

toStr.append(" addmoney = " + this.addmoney + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" content = " + this.content + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

public Performance populate(){

Performance i=new Performance();

i.setCreatetime(DateUtil.parseToDate(this.getCreatetime(),DateUtil.yyyyMMddHHmmss));

i.setScore(this.getScore());

i.setId(this.getId());

i.setAddmoney(this.getAddmoney());

i.setContent(this.getContent());

return i;

}

public ActionErrors validate(ActionMapping arg0, HttpServletRequest arg1) {

return null;

}

}

/HRmanagement/src/com.zhangying.action/StipendAction.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.StipendDao;

import com.zhangying.po.Stipend;

public class StipendAction extends Action {

private StipendDao dao=new StipendDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

System.out.println("\nStipendAction\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*action="+action);

if(action==null||"".equals(action)){

return mapping.findForward("error");

}else if("liststipend".equals(action)){

return listStipend(mapping,form,request,response);

}else if("addstipend".equals(action)){

return addStipend(mapping,form,request,response);

}else if("deletestipend".equals(action)){

return deleteStipend(mapping,form,request,response);

}else if("updatestipend".equals(action)){

return updateStipend(mapping,form,request,response);

}else if("detailstipend".equals(action)){

return detailStipend(mapping,form,request,response);

}

return mapping.findForward("error");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward detailStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id"));

Stipend s=dao.loadStipend(id.longValue());

request.setAttribute("stipend",s);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward updateStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

StipendForm stipendForm=(StipendForm)form;

Stipend s=stipendForm.populate();

dao.updateStipend(s);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward deleteStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

Long id=new Long(request.getParameter("id").toString());

Stipend s=new Stipend();

s.setId(id);

dao.deleteStipend(s);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward addStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

StipendForm stipendForm=(StipendForm)form;

Stipend s=stipendForm.populate();

dao.addStipend(s);

return mapping.findForward("success");

}

/\*\*

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return

\* @throws HibernateException

\*/

private ActionForward listStipend(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response) throws HibernateException {

request.setAttribute("list",dao.listStipend());

return mapping.findForward("success");

}

}

/HRmanagement/src/com.zhangying.action/StipendForm.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionMapping;

import com.zhangying.po.Stipend;

import com.zhangying.tool.DateUtil;

public class StipendForm extends ActionForm {

public ActionErrors validate(ActionMapping arg0, HttpServletRequest arg1) {

return null;

}

/\*\* identifier field \*/

private Long id;

/\*\* nullable persistent field \*/

private String name;

/\*\* nullable persistent field \*/

private float basic;

/\*\* nullable persistent field \*/

private String granttime;

/\*\* nullable persistent field \*/

private float duty;

/\*\* nullable persistent field \*/

private float punishment;

/\*\* nullable persistent field \*/

private float other;

/\*\* nullable persistent field \*/

private float totalize;

/\*\* full constructor \*/

public void reset(ActionMapping mapping,HttpServletRequest request) {

this.name = null;

this.basic = 0;

this.granttime = null;

this.duty = 0;

this.punishment = 0;

this.other = 0;

this.totalize =0;

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public float getBasic() {

return this.basic;

}

public void setBasic(float basic) {

this.basic = basic;

}

public float getDuty() {

return this.duty;

}

public void setDuty(float duty) {

this.duty = duty;

}

public float getPunishment() {

return this.punishment;

}

public void setPunishment(float punishment) {

this.punishment = punishment;

}

public float getOther() {

return this.other;

}

public void setOther(float other) {

this.other = other;

}

public float getTotalize() {

return this.totalize;

}

public void setTotalize(float totalize) {

this.totalize = totalize;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Stipend] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" basic = " + this.basic + ";\n");

toStr.append(" granttime = " + this.granttime + ";\n");

toStr.append(" duty = " + this.duty + ";\n");

toStr.append(" punishment = " + this.punishment + ";\n");

toStr.append(" other = " + this.other + ";\n");

toStr.append(" totalize = " + this.totalize + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

public Stipend populate(){

Stipend s=new Stipend();

s.setBasic(this.getBasic());

s.setDuty(this.getDuty());

s.setId(this.getId());

s.setGranttime(DateUtil.parseToDate(this.getGranttime(),DateUtil.yyyyMMdd));

s.setName(this.getName());

s.setOther(this.getOther());

s.setPunishment(this.getPunishment());

s.setTotalize(this.getTotalize());

return s;

}

/\*\*

\* @return Returns the granttime.

\*/

public String getGranttime() {

return granttime;

}

/\*\*

\* @param granttime The granttime to set.

\*/

public void setGranttime(String granttime) {

this.granttime = granttime;

}

}

/HRmanagement/src/com.zhangying.action/UsersAction.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import org.hibernate.HibernateException;

import com.zhangying.dao.UsersDao;

import com.zhangying.po.Users;

public class UsersAction extends Action {

private UsersDao dao=new UsersDao();

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) throws Exception {

String action =request.getParameter("action");

if(action==null||" ".equals(action)){

return mapping.findForward("error");

}else if("listuser".equals(action)){

return listUser(mapping,form,request,response);

}else if("adduser".equals(action)){

return addUsers(mapping,form,request,response);

}else if("logon".equals(action)){

return logon(mapping,form,request,response);

}else if("updateuser".equals(action)){

return updateUser(mapping,form,request,response);

}else if("deleteuser".equals(action)){

return deleteUser(mapping,form,request,response);

}else if("selectuser".equals(action)){

return selectUser(mapping,form,request,response);

}

return mapping.findForward("error");

}

private ActionForward selectUser(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//获得id参数

Users u=dao.loadUsers(id);//加载该ID对应的人员信息

request.setAttribute("user",u);//将人员信息设置到request范围

return mapping.findForward("success");//跳转到success对应的逻辑视图

}

private ActionForward updateUser(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

UsersForm usersform=(UsersForm)form;//获得UsersForm

Users users=usersform.populate();//获得提交的人员信息

dao.updateUsers(users);//更新人员信息

return mapping.findForward("success");//跳转到success对应的逻辑视图

}

private ActionForward deleteUser(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

Long id=new Long(request.getParameter("id"));//获得id参数

Users users=new Users();//新建一个users对象

users.setId(id);//设置该users对象的id

dao.deleteUsers(users);//删除该users对象

return mapping.findForward("success");

}

private ActionForward logon(ActionMapping mapping,ActionForm form,HttpServletRequest request,HttpServletResponse response) throws HibernateException {

UsersForm usersform=(UsersForm)form;

// System.out.println("userform="+form);

Users users=new Users();

users.setUsername(usersform.getUsername());

users.setPassword(usersform.getPassword());

//users.setUsername(request.getParameter("username"));

//users.setPassword(request.getParameter("password"));

boolean flag=dao.logonUsers(users);

if(flag){

request.getSession().setAttribute("users",users);

return mapping.findForward("success");

}else{

request.setAttribute("error", "登录失败");

return mapping.findForward("failed");

}

}

private ActionForward addUsers(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

UsersForm usersform=(UsersForm)form; //获得UsersForm

Users users=usersform.populate(); //获得提交的人员信息

dao.addUsers(users); //完成人员信息保存

return mapping.findForward("success"); //跳转到成功页面

}

private ActionForward listUser(ActionMapping mapping, ActionForm form, HttpServletRequest request, HttpServletResponse response){

request.setAttribute("list",dao.listUser());//将人员信息列表保存到request范围

return mapping.findForward("success"); //跳转到成功页面

}

}

/HRmanagement/src/com.zhangying.action/UsersForm.java

package com.zhangying.action;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionMapping;

import com.zhangying.po.Users;

import com.zhangying.tool.DateUtil;

/\*\*

\* @author f(x)函数

\*

\*/

public class UsersForm extends ActionForm {

/\*\*

\* @return Returns the content.

\*/

public String getContent() {

return content;

}

/\*\*

\* @param content The content to set.

\*/

public void setContent(String content) {

this.content = content;

}

/\*\* identifier field \*/

private Long id;

/\*\* nullable persistent field \*/

private String username;

/\*\* nullable persistent field \*/

private String password;

/\*\* nullable persistent field \*/

private Byte sex;

/\*\* nullable persistent field \*/

private String birthday;

/\*\* nullable persistent field \*/

private String createtime;

/\*\* nullable persistent field \*/

private Byte isadmin;

/\*\* nullable persistent field \*/

private String content;

/\*\*

\* @return Returns the createtime.

\*/

public String getCreatetime() {

return createtime;

}

/\*\*

\* @param createtime The createtime to set.

\*/

public void setCreatetime(String createtime) {

this.createtime = createtime;

}

/\*\*

\* @return Returns the id.

\*/

public Long getId() {

return id;

}

/\*\*

\* @param id The id to set.

\*/

public void setId(Long id) {

this.id = id;

}

/\*\*

\* @return Returns the isadmin.

\*/

public Byte getIsadmin() {

return isadmin;

}

/\*\*

\* @param isadmin The isadmin to set.

\*/

public void setIsadmin(Byte isadmin) {

this.isadmin = isadmin;

}

/\*\*

\* @return Returns the password.

\*/

public String getPassword() {

return password;

}

/\*\*

\* @param password The password to set.

\*/

public void setPassword(String password) {

this.password = password;

}

/\*\*

\* @return Returns the username.

\*/

public String getUsername() {

return username;

}

/\*\*

\* @param username The username to set.

\*/

public void setUsername(String username) {

this.username = username;

}

public void reset(ActionMapping arg0, HttpServletRequest arg1) {

this.id=null;

this.username=null;

this.password=null;

this.isadmin=null;

this.createtime=null;

this.content=null;

this.sex=null;

this.birthday=null;

}

public ActionErrors validate(ActionMapping arg0, HttpServletRequest arg1) {

return null;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[UsersForm] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" username = " + this.username + ";\n");

toStr.append(" password = " + this.password + ";\n");

toStr.append(" sex = " + this.sex + ";\n");

toStr.append(" birthday = " + this.birthday+ ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" isadmin = " + this.isadmin + ";\n");

toStr.append(" content = " + this.content + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

/\*\*

\* @return Returns the birthday.

\*/

public String getBirthday() {

return birthday;

}

/\*\*

\* @param birthday The birthday to set.

\*/

public void setBirthday(String birthday) {

this.birthday = birthday;

}

/\*\*

\* @return Returns the sex.

\*/

public Byte getSex() {

return sex;

}

/\*\*

\* @param sex The sex to set.

\*/

public void setSex(Byte sex) {

this.sex = sex;

}

public Users populate(){

Users u=new Users();

u.setId(getId());

u.setIsadmin(getIsadmin());

u.setContent(getContent());

u.setUsername(getUsername());

u.setPassword(getPassword());

u.setSex(getSex());

u.setBirthday(DateUtil.parseToDate(getBirthday(),DateUtil.yyyyMMdd));

u.setCreatetime(DateUtil.parseToDate(getCreatetime(),DateUtil.yyyyMMddHHmmss));

return u;

}

}

/HRmanagement/src/com.zhangying.dao/EducateDao.java

package com.zhangying.dao;

import java.util.List;

import org.hibernate.HibernateException;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import com.zhangying.hb.HibernateSessionFactory;

import com.zhangying.po.Educate;

/\*\*

\* @author f(x)函数

\*

\*/

public class EducateDao {

public void addEducate(Educate e) throws HibernateException{

e.setCreatetime(new java.util.Date());

e.setEducate(new Byte("0"));

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.save(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public void deleteEducate(Educate e) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.delete(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public Educate loadEducate(long id) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Educate e=(Educate)session.get(Educate.class,new Long(id));

tx.commit();

HibernateSessionFactory.closeSession();

return e;

}

public List listEducate(byte educate) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Query query = session.createQuery("select e from Educate as e where e.educate =:educate order by createtime");

query.setByte("educate",educate);

List list = query.list();

tx.commit();

HibernateSessionFactory.closeSession();

return list;

}

public void updateEducate(Educate educate) throws HibernateException{

Educate e =this.loadEducate(educate.getId().longValue());

if (educate.getBegintime()!=null){

e.setBegintime(educate.getBegintime());

}

if(educate.getDatum()!=null){

e.setDatum(educate.getDatum());

}

if (educate.getEffect()!= null) {

e.setEffect(educate.getEffect());

}

if (educate.getEndtime()!= null) {

e.setEndtime(educate.getEndtime());

}

if(educate.getName()!=null){

e.setName(educate.getName());

}

if(educate.getPurpose()!=null){

e.setPurpose(educate.getPurpose());

}

if(educate.getStudent()!=null){

e.setStudent(educate.getStudent());

}

if(educate.getSummarize()!=null){

e.setSummarize(educate.getSummarize());

}

if(educate.getEducate()!=null){

e.setEducate(educate.getEducate());

}

if(educate.getTeacher()!=null){

e.setTeacher(educate.getTeacher());

}

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.update(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

}

/HRmanagement/src/com.zhangying.dao/InstitutionDao.java

package com.zhangying.dao;

import java.util.List;

import org.hibernate.HibernateException;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import com.zhangying.hb.HibernateSessionFactory;

import com.zhangying.po.Institution;

/\*\*

\* @author f(x)函数

\*

\*/

public class InstitutionDao {

public void addInstitution(Institution e) throws HibernateException{

e.setCreatetime(new java.util.Date());

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.save(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public void deleteInstitution(Institution e) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.delete(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public Institution loadInstitution(long id) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Institution e=(Institution)session.get(Institution.class,new Long(id));

tx.commit();

HibernateSessionFactory.closeSession();

return e;

}

public List listInstitution() throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Query query = session.createQuery("select e from Institution as e order by createtime");

List list = query.list();

tx.commit();

HibernateSessionFactory.closeSession();

return list;

}

public void updateInstitution(Institution ins) throws HibernateException{

Institution e =this.loadInstitution(ins.getId().longValue());

if (ins.getExplain()!=null){

e.setExplain(ins.getExplain());

}

if(ins.getName()!=null){

e.setName(ins.getName());

}

if (ins.getReason()!= null) {

e.setReason(ins.getReason());

}

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.update(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

}

/HRmanagement/src/com.zhangying.dao/JobDao.java

package com.zhangying.dao;

import java.util.List;

import org.hibernate.HibernateException;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import com.zhangying.hb.HibernateSessionFactory;

import com.zhangying.po.Job;

public class JobDao {

public void addJob(Job job) throws HibernateException{

job.setIsstock(new Byte("0"));//设置入库信息为0，表示暂时不入库

job.setCreatetime(new java.util.Date());//设置当前时间

Session session = HibernateSessionFactory.getSession();//获得Session对象

Transaction tx = session.beginTransaction();//开启事务处理

session.save(job);//保存应聘人员信息

tx.commit();//提交事务

HibernateSessionFactory.closeSession();//关闭Session对象

}

public List listJob(byte isstock) throws HibernateException{

Session session = HibernateSessionFactory.getSession();//获得Session对象

Query query = session.createQuery("select j from Job " +

"as j where j.isstock = :isstock order by createtime");//查询所有应聘信息

query.setByte("isstock",isstock);//设置是否入库

List list = query.list();//获得应聘信息列表

HibernateSessionFactory.closeSession();//关闭Session对象

return list;

}

public Job loadJob(long id){

Session session = HibernateSessionFactory.getSession();//获得Session对象

Job j = (Job) session.get(Job.class,id);//加载指定ID的应聘信息

HibernateSessionFactory.closeSession();//关闭Session对象

return j;//返回查询对象

}

public void deleteJob(Job job) throws HibernateException{

Session session = HibernateSessionFactory.getSession();//获得Session对象

Transaction tx = session.beginTransaction();//开启事务处理

session.delete(job);//删除该job对象

tx.commit();//提交事务

HibernateSessionFactory.closeSession();//关闭Session对象

}

public void updateJob(Job job){

Job j = loadJob(job.getId().longValue());

if (job.getAge()!=null){

j.setAge(job.getAge());

}

if(job.getContent()!=null){

j.setContent(job.getContent());

}

if (job.getExperience() != null) {

j.setExperience(job.getExperience());

}

if (job.getIsstock()!= null) {

j.setIsstock(job.getIsstock());

}

if(job.getJob()!=null){

j.setJob(job.getJob());

}

if(job.getName()!=null){

j.setName(job.getName());

}

if(job.getSchool()!=null){

j.setSchool(job.getSchool());

}

if(job.getSex()!=null){

j.setSex(job.getSex());

}

if(job.getHealth()!=null){

j.setHealth(job.getHealth());

}

if(job.getSpecialty()!=null){

j.setSpecialty(job.getSpecialty());

}

if(job.getStudyeffort()!=null){

j.setStudyeffort(job.getStudyeffort());

}

if(job.getTel()!=null){

j.setTel(job.getTel());

}

if(job.getEmail()!=null){

j.setEmail(job.getEmail());

}

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.update(j);

tx.commit();

HibernateSessionFactory.closeSession();

}

}

/HRmanagement/src/com.zhangying.dao/PerformanceDao.java

package com.zhangying.dao;

import java.util.List;

import org.hibernate.HibernateException;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import com.zhangying.hb.HibernateSessionFactory;

import com.zhangying.po.Performance;

/\*\*

\* @author f(x)函数

\*

\*/

public class PerformanceDao {

public void addPerformance(Performance e) throws HibernateException{

e.setCreatetime(new java.util.Date());

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.save(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public void deletePerformance(Performance e) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.delete(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public Performance loadPerformance(long id) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Performance e=(Performance)session.get(Performance.class,new Long(id));

tx.commit();

HibernateSessionFactory.closeSession();

return e;

}

public List listPerformance() throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Query query = session.createQuery("select e from Performance as e order by createtime");

List list = query.list();

tx.commit();

HibernateSessionFactory.closeSession();

return list;

}

public void updatePerformance(Performance ins) throws HibernateException{

Performance e =this.loadPerformance(ins.getId().longValue());

if (ins.getContent()!=null) {

e.setContent(ins.getContent());

}

if(String.valueOf(ins.getScore())!=null){

e.setScore(ins.getScore());

}

if (String.valueOf(ins.getAddmoney())!=null){

e.setAddmoney(ins.getAddmoney());

}

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.update(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

}

/HRmanagement/src/com.zhangying.dao/StipendDao.java

package com.zhangying.dao;

import java.util.List;

import org.hibernate.HibernateException;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import com.zhangying.hb.HibernateSessionFactory;

import com.zhangying.po.Stipend;

/\*\*

\* @author f(x)函数

\*

\*/

public class StipendDao {

/\*\*

\* 计算工资总数

\*\*/

private Stipend getCountTotalize(Stipend e){

float count=0;

count=count+e.getBasic().longValue();

count=count+e.getDuty();

count=count+e.getOther();

count=count-e.getPunishment();

e.setTotalize(new Float(count));

return e;

}

public void addStipend(Stipend e) throws HibernateException{

e=getCountTotalize(e);

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.save(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public void deleteStipend(Stipend e) throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.delete(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

public Stipend loadStipend(long id) throws HibernateException{

System.out.println("33333444");

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

System.out.println("33333");

Stipend e=(Stipend)session.get(Stipend.class,new Long(id));

System.out.println("e="+e);

tx.commit();

HibernateSessionFactory.closeSession();

return e;

}

public List listStipend() throws HibernateException{

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Query query = session.createQuery("select e from Stipend as e order by e.granttime");

List list = query.list();

tx.commit();

HibernateSessionFactory.closeSession();

return list;

}

public void updateStipend(Stipend stipend) throws HibernateException{

Stipend e =new Stipend();

Long l=stipend.getId();

e =this.loadStipend(l.longValue());

if (String.valueOf(stipend.getBasic())!=null){

e.setBasic(stipend.getBasic());

}

if(stipend.getGranttime()!=null){

e.setGranttime(stipend.getGranttime());

}

if(stipend.getName()!=null){

e.setName(stipend.getName());

}

if(String.valueOf(stipend.getOther())!=null){

e.setOther(stipend.getOther());

}

if(String.valueOf(stipend.getPunishment())!=null){

e.setPunishment(stipend.getPunishment());

}

e=getCountTotalize(e);

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

session.update(e);

tx.commit();

HibernateSessionFactory.closeSession();

}

}

/HRmanagement/src/com.zhangying.dao/UsersDao.java

package com.zhangying.dao;

import java.util.Iterator;

import java.util.List;

import org.hibernate.HibernateException;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import com.zhangying.hb.HibernateSessionFactory;

import com.zhangying.po.Users;

public class UsersDao {

public void addUsers(Users users){ //新增人员信息

users.setCreatetime(new java.util.Date());//设置创建时间

Session session = HibernateSessionFactory.getSession();//获得Seesion对象

Transaction tx = session.beginTransaction();//开启事务

session.save(users);//保存用户

tx.commit();//提交事务

HibernateSessionFactory.closeSession();//关闭Seesion对象

}

public void deleteUsers(Users users){//删除人员信息

Session session = HibernateSessionFactory.getSession();//获得Seesion对象

Transaction tx = session.beginTransaction();//开启事务

session.delete(users);//删除users对象

tx.commit();//提交事务

HibernateSessionFactory.closeSession();//关闭Seesion对象

}

public void updateUsers(Users users){//更新人员信息

Users u = loadUsers(users.getId());//加载该ID对应的人员信息

if (users.getBirthday() != null){

u.setBirthday(users.getBirthday());//设置出生日期

}

if(users.getSex()!=null){

u.setSex(users.getSex());//设置性别

}

if (users.getContent() != null) {

u.setContent(users.getContent());//设置人员简介

}

if (users.getIsadmin() != null) {

u.setIsadmin(users.getIsadmin());//设置是否为管理员

}

if (users.getPassword() != null) {

u.setPassword(users.getPassword());//设置登录密码

}

Session session = HibernateSessionFactory.getSession();//获得Session对象

Transaction tx = session.beginTransaction();//开启事务

session.update(u);//更新对象

tx.commit();//提交事务

HibernateSessionFactory.closeSession();//关闭Session对象

}

public Users loadUsers(long id){//加载人员信息

Session session = HibernateSessionFactory.getSession();//获得Session对象

Users u = (Users)session.get(Users.class, id);//加载指定ID的Users对象

return u;

}

public boolean logonUsers(Users users){//人员登录

Session session = HibernateSessionFactory.getSession();

Transaction tx = session.beginTransaction();

Query query = session.createQuery("select count(\*) from Users as u where username = :name and password =:password");

query.setString("name", users.getUsername());

query.setString("password", users.getPassword());

List list = null;

list = query.list();

tx.commit();

HibernateSessionFactory.closeSession();

if (list != null) {

Iterator it = list.iterator();

if (it.hasNext()) {

if ((Long) it.next() == 1) {

return true;

}

}

}

return false;

}

public List listUser(){//显示人员列表

Session session = HibernateSessionFactory.getSession();//获得Session对象

Query query = session.createQuery("select u from Users as u order by createtime");//执行查询

List list = query.list();//获得查询列表

HibernateSessionFactory.closeSession();//关闭Session对象

return list; //返回人员列表

}

}

/HRmanagement/src/com.zhangying.hb/HibernateSessionFactory.java

package com.zhangying.hb;

import org.hibernate.HibernateException;

import org.hibernate.Session;

import org.hibernate.cfg.Configuration;

/\*\*

\* Configures and provides access to Hibernate sessions, tied to the

\* current thread of execution. Follows the Thread Local Session

\* pattern, see {@link http://hibernate.org/42.html }.

\*/

public class HibernateSessionFactory {

/\*\*

\* Location of hibernate.cfg.xml file.

\* Location should be on the classpath as Hibernate uses

\* #resourceAsStream style lookup for its configuration file.

\* The default classpath location of the hibernate config file is

\* in the default package. Use #setConfigFile() to update

\* the location of the configuration file for the current session.

\*/

private static String CONFIG\_FILE\_LOCATION = "/hibernate.cfg.xml";

private static final ThreadLocal<Session> threadLocal = new ThreadLocal<Session>();

private static Configuration configuration = new Configuration();

private static org.hibernate.SessionFactory sessionFactory;

private static String configFile = CONFIG\_FILE\_LOCATION;

static {

try {

configuration.configure(configFile);

sessionFactory = configuration.buildSessionFactory();

} catch (Exception e) {

System.err.println("%%%% Error Creating SessionFactory %%%%");

e.printStackTrace();

}

}

private HibernateSessionFactory() {

}

/\*\*

\* Returns the ThreadLocal Session instance. Lazy initialize

\* the <code>SessionFactory</code> if needed.

\*

\* @return Session

\* @throws HibernateException

\*/

public static Session getSession() throws HibernateException {

Session session = (Session) threadLocal.get();

if (session == null || !session.isOpen()) {

if (sessionFactory == null) {

rebuildSessionFactory();

}

session = (sessionFactory != null) ? sessionFactory.openSession(): null;

threadLocal.set(session);

}

return session;

}

/\*\*

\* Rebuild hibernate session factory

\*

\*/

public static void rebuildSessionFactory() {

try {

configuration.configure(configFile);

sessionFactory = configuration.buildSessionFactory();

} catch (Exception e) {

System.err.println("%%%% Error Creating SessionFactory %%%%");

e.printStackTrace();

}

}

/\*\*

\* Close the single hibernate session instance.

\*

\* @throws HibernateException

\*/

public static void closeSession() throws HibernateException {

Session session = (Session) threadLocal.get();

threadLocal.set(null);

if (session != null) {

session.close();

}

}

/\*\*

\* return session factory

\*

\*/

public static org.hibernate.SessionFactory getSessionFactory() {

return sessionFactory;

}

/\*\*

\* return session factory

\*

\* session factory will be rebuilded in the next call

\*/

public static void setConfigFile(String configFile) {

HibernateSessionFactory.configFile = configFile;

sessionFactory = null;

}

/\*\*

\* return hibernate configuration

\*

\*/

public static Configuration getConfiguration() {

return configuration;

}

}

/HRmanagement/src/com.zhangying.po/Educate.hbm.xml

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">

<hibernate-mapping>

<class name="com.zhangying.po.Educate" table="educate">

<id column="id" name="id" type="long">

<generator class="native"/> <!-- 主键生成策略 -->

</id>

<property name="name" length="100" type="string"/>

<property name="purpose" length="500" type="string"/>

<property name="begintime" length="23" type="timestamp"/>

<property name="endtime" length="23" type="timestamp"/>

<property name="datum" length="2000" type="string"/>

<property name="teacher" length="50" type="string"/>

<property name="student" length="50" type="string"/>

<property name="createtime" length="23" type="timestamp"/>

<property name="effect" length="500" type="string"/>

<property name="educate" length="1" type="byte"/>

<property name="summarize" length="2000" type="string"/>

</class>

</hibernate-mapping>

/HRmanagement/src/com.zhangying.po/Educate.java

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

/\*\* 培训 \*/

public class Educate implements Serializable {

private Long id; //培训编号

private String name; //培训名称

private String purpose; //培训目的

private Date begintime; //培训开始时间

private Date endtime; //培训结束时间

private String datum; //培训材料

private String teacher; //培训讲师

private String student; //培训人员

private Date createtime;//创建时间

private Byte educate; //培训是否完成

private String effect; //培训效果

private String summarize;//总结

public Educate(String name, String purpose, Date begintime, Date endtime, String datum, String teacher, String student, Date createtime, Byte educate,String effect, String summarize) {

this.name = name;

this.purpose = purpose;

this.begintime = begintime;

this.endtime = endtime;

this.datum = datum;

this.teacher = teacher;

this.student = student;

this.createtime = createtime;

this.educate=educate;

this.effect = effect;

this.summarize = summarize;

}

/\*\* default constructor \*/

public Educate() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public String getPurpose() {

return this.purpose;

}

public void setPurpose(String purpose) {

this.purpose = purpose;

}

public Date getBegintime() {

return this.begintime;

}

public void setBegintime(Date begintime) {

this.begintime = begintime;

}

public Date getEndtime() {

return this.endtime;

}

public void setEndtime(Date endtime) {

this.endtime = endtime;

}

public String getDatum() {

return this.datum;

}

public void setDatum(String datum) {

this.datum = datum;

}

public String getTeacher() {

return this.teacher;

}

public void setTeacher(String teacher) {

this.teacher = teacher;

}

public String getStudent() {

return this.student;

}

public void setStudent(String student) {

this.student = student;

}

public Date getCreatetime() {

return this.createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public String getEffect() {

return this.effect;

}

public void setEffect(String effect) {

this.effect = effect;

}

public String getSummarize() {

return this.summarize;

}

public void setSummarize(String summarize) {

this.summarize = summarize;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Educate] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" begintime = " + this.begintime+ ";\n");

toStr.append(" endtime = " + this.endtime+ ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" purpose = " + this.purpose + ";\n");

toStr.append(" datum = " + this.datum + ";\n");

toStr.append(" teacher = " + this.teacher + ";\n");

toStr.append(" student = " + this.student + ";\n");

toStr.append(" educate = " + this.educate+ ";\n");

toStr.append(" effect = " + this.effect + ";\n");

toStr.append(" summarize = " + this.summarize + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

/\*\*

\* @return Returns the educate.

\*/

public Byte getEducate() {

return educate;

}

/\*\*

\* @param educate The educate to set.

\*/

public void setEducate(Byte educate) {

this.educate = educate;

}

}

/HRmanagement/src/com.zhangying.po/Institution.hbm.xml

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">

<hibernate-mapping>

<class name="com.zhangying.po.Institution" table="institution">

<id column="id" name="id" type="long">

<generator class="native"/>

</id>

<property name="name" length="50" type="string"/>

<property name="reason" length="1000" type="string"/>

<property name="explain" length="1000" type="string" column="explains"/>

<property name="createtime" length="23" type="timestamp"/>

</class>

</hibernate-mapping>

/HRmanagement/src/com.zhangying.po/Institution.java

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

/\*\* 奖惩 \*/

public class Institution implements Serializable {

private Long id; //奖惩编号

private String name; //奖惩名称

private String reason; //奖惩原因

private String explain; //奖惩说明

private Date createtime;//创建时间

/\*\* full constructor \*/

public Institution(String name, String reason, String explain, Date createtime) {

this.name = name;

this.reason = reason;

this.explain = explain;

this.createtime = createtime;

}

/\*\* default constructor \*/

public Institution() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public String getReason() {

return this.reason;

}

public void setReason(String reason) {

this.reason = reason;

}

public String getExplain() {

return this.explain;

}

public void setExplain(String explain) {

this.explain = explain;

}

public Date getCreatetime() {

return this.createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Institution] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" reason = " + this.reason + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" explain = " + this.explain + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

/HRmanagement/src/com.zhangying.po/Job.hbm.xml

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">

<hibernate-mapping>

<class name="com.zhangying.po.Job" table="job">

<id column="id" name="id" type="long">

<generator class="native"/> <!-- 主键生成策略 -->

</id>

<property name="name" length="50" type="string"/>

<property name="sex" length="1" type="byte"/>

<property name="age" length="10" type="integer"/>

<property name="job" length="50" type="string"/>

<property name="specialty" length="50" type="string"/>

<property name="experience" length="50" type="string"/>

<property name="studyeffort" length="50" type="string"/>

<property name="school" length="50" type="string"/>

<property name="tel" length="50" type="string"/>

<property name="email" length="50" type="string"/>

<property name="createtime" length="23" type="timestamp"/>

<property name="content" length="1073741823" type="string"/>

<property name="isstock" length="1" type="byte"/>

<property name="health" length="1" type="byte"/>

</class>

</hibernate-mapping>

/HRmanagement/src/com.zhangying.po/Job.java

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

public class Job implements Serializable {

private Long id; //应聘人员编号

private String name;//应聘人员姓名

private Byte sex; //性别

private Integer age;//年龄

private String job; //职位

private String specialty;//所学专业

private String experience;//工作经验

private String studyeffort;//学历

private String school; //毕业学校

private String tel; //电话号码

private String email; //Email地址

private Date createtime;//创建时间

private String content; //详细经历

private Byte isstock; //是否入库

private Byte health;//体检结果

public Job(Long id,String name, Byte sex, Integer age, String job, String specialty, String experience, String studyeffort, String school, String tel, String email, Date createtime, String content, Byte isstock,Byte health) {

this.id=id;

this.name = name;

this.sex = sex;

this.age = age;

this.job = job;

this.specialty = specialty;

this.experience = experience;

this.studyeffort = studyeffort;

this.school = school;

this.tel = tel;

this.email = email;

this.createtime = createtime;

this.content = content;

this.isstock = isstock;

this.health=health;

}

public Job() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public Byte getSex() {

return this.sex;

}

public void setSex(Byte sex) {

this.sex = sex;

}

public Integer getAge() {

return this.age;

}

public void setAge(Integer age) {

this.age = age;

}

public String getJob() {

return this.job;

}

public void setJob(String job) {

this.job = job;

}

public String getSpecialty() {

return this.specialty;

}

public void setSpecialty(String specialty) {

this.specialty = specialty;

}

public String getExperience() {

return this.experience;

}

public void setExperience(String experience) {

this.experience = experience;

}

public String getStudyeffort() {

return this.studyeffort;

}

public void setStudyeffort(String studyeffort) {

this.studyeffort = studyeffort;

}

public String getSchool() {

return this.school;

}

public void setSchool(String school) {

this.school = school;

}

public String getTel() {

return this.tel;

}

public void setTel(String tel) {

this.tel = tel;

}

public String getEmail() {

return this.email;

}

public void setEmail(String email) {

this.email = email;

}

public Date getCreatetime() {

return this.createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public String getContent() {

return this.content;

}

public void setContent(String content) {

this.content = content;

}

public Byte getIsstock() {

return this.isstock;

}

public void setIsstock(Byte isstock) {

this.isstock = isstock;

}

public Byte getHealth() {

return this.health;

}

public void setHealth(Byte health) {

this.health = health;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Job] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" sex = " + this.sex + ";\n");

toStr.append(" age = " + this.age + ";\n");

toStr.append(" job = " + this.job + ";\n");

toStr.append(" specialty = " + this.specialty + ";\n");

toStr.append(" experience = " + this.experience + ";\n");

toStr.append(" studyeffort = " + this.studyeffort + ";\n");

toStr.append(" school = " + this.school + ";\n");

toStr.append(" tel = " + this.tel + ";\n");

toStr.append(" email = " + this.email + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" content = " + this.content+ ";\n");

toStr.append(" isstock = " + this.isstock + ";\n");

toStr.append(" health = " + this.health + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

/HRmanagement/src/com.zhangying.po/Performance.hbm.xml

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">

<hibernate-mapping>

<class name="com.zhangying.po.Performance" table="performance">

<id column="id" name="id" type="long">

<generator class="native"/>

</id>

<property name="content" length="1000" type="string"/>

<property name="score" length="50" type="float"/>

<property name="addmoney" length="50" type="float" />

<property name="createtime" length="23" type="timestamp"/>

</class>

</hibernate-mapping>

/HRmanagement/src/com.zhangying.po/Performance.java

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

/\*\* 绩效 \*/

public class Performance implements Serializable {

private Long id; //绩效编号

private float score; //绩效分数

private float addmoney; //增加工资

private String content; //绩效说明

private Date createtime;//创建时间

/\*\* full constructor \*/

public Performance(float score, float addmoney, String content, Date createtime) {

this.score = score;

this.addmoney = addmoney;

this.content = content;

this.createtime = createtime;

}

/\*\* default constructor \*/

public Performance() {

}

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public float getScore() {

return score;

}

public void setScore(float score) {

this.score = score;

}

public float getAddmoney() {

return addmoney;

}

public void setAddmoney(float addmoney) {

this.addmoney = addmoney;

}

public String getContent() {

return content;

}

public void setContent(String content) {

this.content = content;

}

public Date getCreatetime() {

return createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Performance] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" content = " + this.content + ";\n");

toStr.append(" score = " + this.score + ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" addmoney = " + this.addmoney + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

/HRmanagement/src/com.zhangying.po/Stipend.hbm.xml

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">

<hibernate-mapping>

<class name="com.zhangying.po.Stipend" table="stipend">

<id column="id" name="id" type="long">

<generator class="native"/>

</id>

<property name="name" length="50" type="string"/>

<property name="basic" length="53" type="float"/>

<property name="granttime" length="23" type="timestamp"/>

<property name="duty" length="53" type="float"/>

<property name="punishment" length="53" type="float"/>

<property name="other" length="53" type="float"/>

<property name="totalize" length="53" type="float"/>

</class>

</hibernate-mapping>

/HRmanagement/src/com.zhangying.po/Stipend.java

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

/\*\* 薪金 \*/

public class Stipend implements Serializable {

private Long id; //薪金信息编号

private String name; //员工姓名

private float basic; //基本薪金

private Date granttime; // 工资发放时间

private float duty; //全勤奖

private float punishment;//罚款

private float other; //额外补助

private float totalize; //总计

public Stipend(String name, float basic, Date granttime, float duty, float punishment, float other, float totalize) {

this.name = name;

this.basic = basic;

this.granttime = granttime;

this.duty = duty;

this.punishment = punishment;

this.other = other;

this.totalize = totalize;

}

/\*\* default constructor \*/

public Stipend() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public Float getBasic() {

return this.basic;

}

public void setBasic(Float basic) {

this.basic = basic;

}

/\*\*

\* @return Returns the granttime.

\*/

public Date getGranttime() {

return granttime;

}

/\*\*

\* @param granttime The granttime to set.

\*/

public void setGranttime(Date granttime) {

this.granttime = granttime;

}

public float getDuty() {

return this.duty;

}

public void setDuty(float duty) {

this.duty = duty;

}

public float getPunishment() {

return this.punishment;

}

public void setPunishment(float punishment) {

this.punishment = punishment;

}

public float getOther() {

return this.other;

}

public void setOther(float other) {

this.other = other;

}

public float getTotalize() {

return this.totalize;

}

public void setTotalize(float totalize) {

this.totalize = totalize;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Stipend] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" name = " + this.name + ";\n");

toStr.append(" basic = " + this.basic + ";\n");

toStr.append(" granttime = " + this.granttime + ";\n");

toStr.append(" duty = " + this.duty + ";\n");

toStr.append(" punishment = " + this.punishment + ";\n");

toStr.append(" other = " + this.other + ";\n");

toStr.append(" totalize = " + this.totalize + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

/HRmanagement/src/com.zhangying.po/Users.hbm.xml

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">

<hibernate-mapping>

<class name="com.zhangying.po.Users" table="users">

<id column="id" name="id" type="long">

<generator class="native"/> <!-- 主键生成策略 -->

</id>

<property name="username" length="50" type="string"/>

<property name="password" length="50" type="string"/>

<property name="sex" length="1" type="byte"/>

<property name="birthday" length="23" type="timestamp"/>

<property name="createtime" length="23" type="timestamp"/>

<property name="isadmin" length="1" type="byte"/>

<property name="content" length="2000" type="string"/>

</class>

</hibernate-mapping>

/HRmanagement/src/com.zhangying.po/Users.java

package com.zhangying.po;

import java.io.Serializable;

import java.util.Date;

public class Users implements Serializable {

private Long id; //员工编号

private String username;//员工用户名

private String password;//登录密码

private Byte sex; //性别

private Date birthday; //生日

private Date createtime;//创建时间

private Byte isadmin; //是否为管理员

private String content; //人员简介

public Users(Long id,String username, String password, Byte sex, Date birthday, Date createtime, Byte isadmin, String content) {

this.id=id;

this.username = username;

this.password = password;

this.sex = sex;

this.birthday = birthday;

this.createtime = createtime;

this.isadmin = isadmin;

this.content = content;

}

public Users() {

}

public Long getId() {

return this.id;

}

public void setId(Long id) {

this.id = id;

}

public String getUsername() {

return this.username;

}

public void setUsername(String username) {

this.username = username;

}

public String getPassword() {

return this.password;

}

public void setPassword(String password) {

this.password = password;

}

public Byte getSex() {

return this.sex;

}

public void setSex(Byte sex) {

this.sex = sex;

}

public Date getBirthday() {

return this.birthday;

}

public void setBirthday(Date birthday) {

this.birthday = birthday;

}

public Date getCreatetime() {

return this.createtime;

}

public void setCreatetime(Date createtime) {

this.createtime = createtime;

}

public Byte getIsadmin() {

return this.isadmin;

}

public void setIsadmin(Byte isadmin) {

this.isadmin = isadmin;

}

public String getContent() {

return this.content;

}

public void setContent(String content) {

this.content = content;

}

public String toString() {

StringBuffer toStr = new StringBuffer();

toStr.append("[Users] = [\n");

toStr.append(" id = " + this.id + ";\n");

toStr.append(" username = " + this.username + ";\n");

toStr.append(" password = " + this.password + ";\n");

toStr.append(" sex = " + this.sex + ";\n");

toStr.append(" birthday = " + this.birthday+ ";\n");

toStr.append(" createtime = " + this.createtime+ ";\n");

toStr.append(" isadmin = " + this.isadmin + ";\n");

toStr.append(" content = " + this.content + ";\n");

toStr.append(" ];\n");

return toStr.toString();

}

}

/HRmanagement/src/com.zhangying.tool/ActionServlet.java

package com.zhangying.tool;

import java.io.IOException;

import java.io.UnsupportedEncodingException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class ActionServlet extends org.apache.struts.action.ActionServlet {

protected void process(HttpServletRequest httpservletrequest,HttpServletResponse httpservletresponse) throws IOException,ServletException {

setCharset(httpservletrequest);//设置编码

super.process(httpservletrequest, httpservletresponse);//调用父类处理方法

}

private void setCharset(HttpServletRequest httpservletrequest) {

String s = getServletConfig().getInitParameter("charset");//获得初始化参数

if (s == null || s.length() == 0) {//如果参数为空，或者为null

s="GBK"; //设置参数值为GBK

}

try {

httpservletrequest.setCharacterEncoding(s);//设置请求编码格式

} catch (UnsupportedEncodingException unsupportedencodingexception) {

log("set character encoding error", unsupportedencodingexception);

}

}

}

/HRmanagement/src/com.zhangying.tool/DateUtil.java

package com.zhangying.tool;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.Date;

public class DateUtil {

public static final String yyMMdd="yy-MM-dd"; //短日期格式

public static final String yyyyMMdd="yyyy-MM-dd"; //长日期格式

public static final String HHmmss="HH:mm:ss"; //时间格式

public static final String yyyyMMddHHmmss

="yyyy-MM-dd HH:mm:ss"; //长日期时间格式

public static final String yyMMddHHmmss

="yy-MM-dd HH:mm:ss"; //短日期时间格式

public static Date parseToDate(String s, String style) {//字符串转换成日期时间

SimpleDateFormat simpleDateFormat

= new SimpleDateFormat(); //实例化日期格式化类

simpleDateFormat.applyPattern(style); //提交转换格式

Date date = null;

if(s==null||s.length()<8) //字符串为空或长度小于8

return null; //无法转换，返回null值

try {

date = simpleDateFormat.parse(s); //进行转换

} catch (ParseException e) {

e.printStackTrace();

}

return date;

}

public static String parseToString(String s, String style) {//格式化日期字符串

SimpleDateFormat simpleDateFormat

= new SimpleDateFormat(); //实例化日期格式化类

simpleDateFormat.applyPattern(style); //提交转换格式

Date date = null;

String str=null;

if(s==null||s.length()<8) //字符串为空或长度小于8

return null; //无法转换，返回null值

try {

date = simpleDateFormat.parse(s); //将字符串转换成日期格式

str=simpleDateFormat.format(date); //将日期格式化为字符串

} catch (ParseException e) {

e.printStackTrace();

}

return str;

}

public static String parseToString(Date date, String style) {//日期时间转换成字符串

SimpleDateFormat simpleDateFormat

= new SimpleDateFormat(); //实例化日期格式化类

simpleDateFormat.applyPattern(style); //提交转换格式

String str=null;

if(date==null) //如果日期时间为null

return null; //返回null值

str=simpleDateFormat.format(date); //将日期格式化为字符串

return str;

}

}

/HRmanagement/src/com.zhangying.po/StringUtil.java

package com.zhangying.tool;

public class StringUtil {

public static String notNull(String s){

if(s==null||s.length()<1)

return "";

return s;

}

}

/HRmanagement/src/hibernate.cfg.xml

<?xml version='1.0' encoding='UTF-8'?>

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

<property name="dialect">

org.hibernate.dialect.MySQLDialect <!-- 数据库方言 -->

</property>

<property name="hibernate.connection.url">

jdbc:mysql://localhost:3306/db\_hr?unicode=true&amp;characterEncoding=UTF-8 <!-- 数据库连接URL -->

</property>

<property name="connection.username">root</property>

<!-- 数据库用户名 -->

<property name="connection.password">123</property>

<!-- 数据库用户密码 -->

<property name="connection.driver\_class">

com.mysql.jdbc.Driver<!-- 数据库驱动类 -->

</property>

<property name="show\_sql">true</property><!--显示SQL语句-->

<mapping resource="com/zhangying/po/Users.hbm.xml"/>

<mapping resource="com/zhangying/po/Job.hbm.xml"/>

<mapping resource="com/zhangying/po/Educate.hbm.xml"/>

<mapping resource="com/zhangying/po/Institution.hbm.xml"/>

<mapping resource="com/zhangying/po/Stipend.hbm.xml"/>

<mapping resource="com/zhangying/po/Performance.hbm.xml"/>

</session-factory>

</hibernate-configuration>

\HRmanagement\WebRoot\css\common1.css

.declare{background:#FFFDE8;border:1px solid #FFEBB6;margin-top:10px;padding:10px}

.FTGreen{color:#7DC818}

.price\_list\_head{border-bottom:2px solid #7F7F7F}

.price\_list\_main{border-bottom:1px solid #D7D7D7}

.SideItem{position:absolute;right:20px;top:7px}

.title{font:bold 14px "宋体";margin:16px 10px 0;position:relative}

.TitleInfo{position:absolute}

.TitleLink{text-align:center;position:absolute}

.TitleTop{text-align:center;position:absolute}

#auction\_list{padding-left:30px;}

#auction\_list li{float:left;text-align:center}

#comm{margin-top:10px;position:relative;width:100%}

#commend{margin-top:10px}

#commend\_head{font:bold 14px "宋体";height:20px;padding:6px 0 0 30px}

#commend\_list td{text-align:center}

#commend\_list{height:180px;padding-top:10px;margin:0 10px 10px}

#CommInfo{position:absolute;right:10px;top:0;width:240px}

#CommInfo div{text-align:left}

#CommMess{text-align:left;width:300px;height:250px;margin:0 auto}

#CommMess #city{width:90px}

#CommMess #TenpayNone{background:#FFFDE8;border:1px solid #FFEBB6;margin-top:5px;padding:5px 10px}

#CommMess #TenpayShow{background:url(http://pics.paipai.com/newsys/auction/auction\_btn\_pay.gif) center no-repeat;margin-top:5px;padding:7px 20px 7px 110px;border:none;line-height:140%}

#CommMess div{border-bottom:1px dashed #D4D4D4;padding:7px 0 4px 10px}

#CommMess div.position{border:none;margin:0;padding:0;position:relative;width:100%}

#CommMess span.PositionShow{left:160px;position:absolute;top:-19px}

#CommPics{left:10px;position:absolute;text-align:center;top:0;width:220px}

#CommPicsShow{height:200px;padding:5px 0;border: 1px solid #EDEDED;}

#CommShow{margin:0 240px 0 220px;text-align:center}

#FullPath{position:relative}

#FullPathShow{border-bottom:1px solid #E1569A;padding:7px 0}

#html\_tem\_leaveMsg td{line-height:150%;word-wrap:break-word;word-break:break-all}

#html\_tem\_price{color:#FF4E00;font:18px Arial}

#LeaveMsg{border:1px dashed #CCCCCC;height:50px;line-height:50px;margin:10px;text-align:center}

#MobileSeller{background:#FFFDE8;border:1px solid #FFEBB6;height:30px;margin-top:10px;padding:10px;text-align:left}

#name{height:28px;padding-top:12px}

#others{height:100%}

#others li{float:left;padding-left:30px;width:45%;border-bottom:1px dashed #D4D4D4;height:24px;line-height:24px}

#others ul{margin:10px 30px}

#pay\_desc1,#pay\_desc2{padding-left:20px}

#price{padding:10px 0 10px 30px;text-align:left}

#price li{float:left;height:30px;line-height:30px;margin-left:20px}

#price\_list{margin-top:10px}

#price\_list\_foot{background:#F7F7F7}

#rate{margin-top:10px}

#rate td{border-bottom:1px dashed #D7D7D7;line-height:150%;word-wrap:break-word;word-break:break-all}

#TenpayBank{line-height:150%;margin:0 0 10px 60px;text-align:left}

#UserHelp{margin-top:5px;padding-left:30px}

#UserHelp li{background:url(images/auction\_btn\_help.gif) 0 2px no-repeat;margin-bottom:3px;padding-left:10px}

#UserInfo{background:#F8F8F8;border:1px solid #D8D8D8}

#UserInfo ul{margin-bottom:7px;position:relative}

#UserInfo ul li.LiLeft{left:18px;position:absolute;top:2px;width: 62px;}

#UserInfo ul li.LiSide{padding-left:80px}

#UserInfo ul li.LiSide div img{padding-right:8px}

#UserInfoHead{background:#EDEDED;padding:4px 0 4px 18px;border-bottom:1px solid #D8D8D8;font-size:14px}

#UserInfoFoot{background:#EDEDED;padding:2px 0 2px 18px;border-top:1px solid #D8D8D8}

.font\_gray{color:#808080}

.btn\_gray{border:1px solid #CCC;color:#808080;background:#F7F7F7}

.input\_main{border:1px solid #CCC}

#DescShow{padding:10px;word-wrap:break-word;word-break:break-all}

#CommendSecond,#auction\_list{margin-top:16px}

.DescShow{padding:20px}

.shopSell{height:20px}

.shopPrize{color:#ff4e00;font-weight:bold;height:17px;padding-top:3px}

.shopTitle{height:50px;overflow:hidden;padding:0 5px;word-break:break-all;word-wrap:break-word}

.shopImg{height:85px}

#DiscountSpan1 {color:red;font-size:14px;font-weight:bold;padding-left:15px}

#DiscountSpan2 {color:red}

#showAttitute{margin:10px;border:1px dashed #8BA2CB;height:1px;background:#f6f6f6}

#ClassAttituteListUL{padding:20px;float:left;}

#ClassAttituteListUL li{float:left;white-space:nowrap;margin:0 10px 10px 0;width:210px;text-align:left;height:16px}

#ClassAttituteListUL li.newLine{border-top:1px solid transparent !important;border-top:0;clear:both}

#buyMore{margin:10px;border:1px solid #FFD488;height:1px;background:#FFFFE6}

#buyMore td{padding-top:10px}

#buyMore div{padding-bottom:10px}

#moreNow div{border-right:1px dashed #D8D8D8}

\HRmanagement\WebRoot\css\control.css

body{

margin:0px;

}

a:link,a:visited{

text-decoration:none;

color:#006

}

a:hover{

text-decoration:underline;

color:#0033CC

}

.headbody{

background:url('../HeadBack.jpg')

}

.headmain{

height:70px;

border-bottom:1px solid #000066;

}

.menuBody{

background:url('../menuB.jpg') repeat-y;

text-align:left;

}

.menuUL{

list-style:none;

margin:0px;

padding:0px;

}

.menuUL li{

display:block;

margin-bottom:5px;

}

.menuA:link,.menuA:visited{

width:100%;

height:35px;

display:block;

background:url('../menubutton.jpg') no-repeat 4px 0px;

font-size:12px;

color:#fff;

text-decoration:none;

}

.menuA:hover{

background:url('../menubutton.jpg') no-repeat 4px -36px;

color:#fff;

text-decoration:none;

}

.menuAS:link,.menuAS:visited{

width:100%;

height:35px;

display:block;

background:url('../menubutton.jpg') no-repeat 4px -72px;

font-size:12px;

color:#000;

text-decoration:none;

font-weight:bold;

}

.ContentBody{

background:#DBDFE8 url(../mainback.jpg) repeat-x fixed; margin:10px;

}

.MainDiv{}

.MenuIcon{

margin:1px 6px -12px 12px;

}

.CTitle{

background:#3E70A6;

padding:5px;

text-align:left;

color:#FFFFFF;

font-size:13px;

font-family: Verdana, Arial, Helvetica, sans-serif;

}

.CContent{

border-top: 1px solid #666666;

border-right: 1px solid #CCCCCC;

border-bottom: 1px solid #CCCCCC;

border-left: 1px solid #666666;

background-color: #FFFFFF;

}

.CPanel{

padding:5px;

font-size:12px;

font-family: Verdana, Arial, Helvetica, sans-serif;

}

legend{

color: #2c3e67;

font-weight:bold;

}

fieldset{

border-top: 1px solid #CCCCCC;

border-right: 1px solid #CCCCCC;

border-bottom: 1px solid #CCCCCC;

border-left: 1px solid #CCCCCC;

margin-top: 10px;

margin-left:10px;

margin-right:10px;

}

fieldset div{

padding:0px;

}

fieldset td {

font-size:12px;

}

.shuom{

padding:0px;

color:#999;

font-size:11px;

}

select{font-size:12px;}

input.text{

border:1px solid #4f6480;font-size:12px;font-family: Verdana, Arial, Helvetica, sans-serif;height:18px

}

button{background:#fff;border:1px solid #999;padding:2px 2px 0px 2px;margin:4px;border-width:1px 3px 1px 3px;}

input.button{

background:#fff;border:1px solid #999;padding:2px 2px 0px 2px;margin:4px;border-width:1px 3px 1px 3px;

}

input.buttonActive{

color: #FFFFFF;

background-color: #5C71AD;

border: 1px solid #000066;

height:20px;

cursor:default;

}

input.buttonUnActive{

color: #143E76;

background-color: #CCD8E6;

border: 1px solid #A4B9D2;

height:20px;

cursor:pointer;

background:url(../UAb.jpg)

}

textarea{ border:1px solid #4f6480;font-size:12px;font-family: Verdana, Arial, Helvetica, sans-serif;}

.SubMenu{

padding: 4px;

border-bottom: 1px solid #003366;

margin:-5px -5px 2px -5px;

background-color: #E6EBF7;

}

.SubButton{

margin-top: 6px;

padding: 4px;

text-align:center;

}

.TDHead{

background:url(../M.jpg);

border-bottom:1px solid #999;

height:16px;

font-weight:bold;

}

.TablePanel{

border:1px solid #999

}

.TablePanel td{font-size:12px;}

.alertTxt{background:#FFCC00 url('../aM.gif');padding:8px 6px 8px 6px !important;padding:8px 6px 8px 6px;font-weight:bold;font-size:14px;}

.textFlow{text-overflow:ellipsis;overflow:hidden;white-space:nowrap }

#msgInfo{margin-bottom:8px;height:33px;top:80px;position:absolute;text-align:center;width:100%;}

/\*---主内容页数--\*/

.pageContent{margin:0 auto;padding:3px;text-align:left;height:16px;font-size:12px;}

.NavTitle{float:left;font-weight:bold;}

.page{padding-right:5px;height:18px;}

.page ul{list-style:none;margin:0px;padding:0px}

.page ul li{float:left;padding:0px 3px 0px 3px;}

.page ul li.PageL {width:9px;height:16px;}

.page ul li.PageR {width:9px;height:16px;}

.PageLbutton{background:url('../BttnLeft.gif') 0px 0px;width:9px;height:16px;display:block;}

.PageRbutton{background:url('../BttnRight.gif') 0px 0px;width:9px;height:16px;display:block;}

/\*---Skin Viewer---\*/

.selectskin{margin:5px;padding:6px;border:1px solid #4d3a22;background:#ffffce;height:100px}

.unselectskin{margin:5px;padding:6px;border:1px solid #fff;height:100px}

.skinimg{width:100px;height:100px;float:left;margin-right:6px;border:1px solid #ccc}

.skinDes{line-height:150%;margin-bottom:12px;}

.cskin{position: absolute;margin-left:74px !important;margin-left:-34px;margin-top:-12px !important;margin-top:-14px}

.msgDiv{margin:4px;}

.msgDiv .item{width:auto !important;width:100%;border:1px solid #D6DEF1;margin-bottom:8px;}

.msgDiv .title{padding:4px;background:#E6EBF7;border-bottom:1px solid #AAC1E6;}

.msgDiv .title .date{font-size:10px;color:#666}

.msgDiv .title img{margin:0px 4px -3px 1px;}

.msgDiv .blogTitle{float:right;margin-top:3px;background:url(../BttnRight.gif) -2px right no-repeat;padding-right:12px}

.msgDiv .content{padding:3px 5px 3px 3px !important;padding:0px}

.msgDiv .contentTB{padding:4px;line-height:150%;word-break:break-all;overflow:hidden}

.msgDiv .reply{padding:0px;background:#F3F5FC;width:auto !important;width:100%;}

.msgDiv .reply h5{padding-top:2px;padding-left:3px;border-bottom:1px solid #DAE1F3;margin:0px;font-size:12px}

.msgDiv .reply h5 .tip{color:#f00;font-weight:100;margin-left:5px;}

.msgDiv textarea{width:100%;margin:auto !important;background:transparent;margin:3px;padding:1px;border:0px solid #fff;cursor:pointer;height:33px;overflow:hidden;line-height:140% !important;line-height:130%}

\HRmanagement\WebRoot\css\css.css

.top-font01 {

font-family: "宋体";

font-size: 12px;

color: #448CCB;

text-decoration: none;

}

a.top-font01:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

}

.left-table01 {

border-top-width: 2px;

border-top-style: solid;

border-top-color: #B8C9D6;

margin-left: 14px;

}

.left-table02 {

margin-top: 8px;

margin-bottom: 8px;

font-family: "宋体";

font-size: 12px;

color: #555555;

text-decoration: none;

}

.left-table03 {

margin-top: 6px;

font-family: "宋体";

font-size: 12px;

color: #555555;

text-decoration: none;

background-image: url(../images/nav02.gif);

}

a.left-table03:hover {

background-image: url(../images/nav03.gif);

}

.left-table04 {

font-family: "宋体";

font-size: 12px;

color: #555555;

text-decoration: none;

background-image: url(../images/nav03.gif);

margin-top: 6px;

}

.left-font01 {

font-family: "宋体";

font-size: 12px;

color: #448CCB;

text-decoration: none;

filter: DropShadow(Color=#FFFFFF, OffX=1, OffY=1);

}

a.left-font01:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

filter: DropShadow(Color=#FFFFFF, OffX=1, OffY=11,);

}

.left-font02 {

font-family: Tahoma;

font-size: 11px;

font-weight: bold;

color: #448CCB;

text-decoration: none;

filter: DropShadow(Color=#FFFFFF, OffX=1, OffY=1);

}

.left-font03 {

font-family: "宋体";

font-size: 12px;

color: #555555;

text-decoration: none;

}

a.left-font03:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

}

.right-table01 {

background-image: url(../images/button01.gif);

text-align: center;

height: 40px;

float: left;

width: 117px;

}

.right-font01 {

font-family: "宋体";

font-size: 12px;

color: #A1A1A1;

text-decoration: none;

filter:

Dropshadow(offx=1,offy=0,color=white)

Dropshadow(offx=0,offy=1,color=white)

Dropshadow(offx=0,offy=-1,color=white)

Dropshadow(offx=-1,offy=0,color=white);

}

a.right-font01:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

filter:

Dropshadow(offx=1,offy=0,color=white)

Dropshadow(offx=0,offy=1,color=white)

Dropshadow(offx=0,offy=-1,color=white)

Dropshadow(offx=-1,offy=0,color=white)

}

.right-font02 {

font-family: "宋体";

font-size: 12px;

font-weight: bold;

color: #4EA1CB;

text-decoration: none;

filter:

Dropshadow(offx=1,offy=0,color=white)

Dropshadow(offx=0,offy=1,color=white)

Dropshadow(offx=0,offy=-1,color=white)

Dropshadow(offx=-1,offy=0,color=white);

}

a.right-font02:hover {

font-family: "宋体";

font-size: 12px;

font-weight: bold;

color: #FF0000;

text-decoration: underline;

filter:

Dropshadow(offx=1,offy=0,color=white)

Dropshadow(offx=0,offy=1,color=white)

Dropshadow(offx=0,offy=-1,color=white)

Dropshadow(offx=-1,offy=0,color=white);

}

.right-table02 {

background-image: url(../images/button02.gif);

float: left;

height: 40px;

width: 117px;

text-align: center;

}

.right-table03 {

margin-top: 16px;

margin-bottom: 20px;

}

.right-table04 {

padding-top: 3px;

padding-left: 7px;

}

.right-table05 {

margin-left: 2px;

margin-top: 12px;

}

.right-table06 {

margin-top: 2px;

margin-right: 2px;

margin-left: 2px;

}

.right-table07 {

margin-top: 16px;

margin-left: 20px;

margin-bottom: 40px;

}

.right-table08 tr{wfsr:expression(this.bgColor=this.rowIndex%2?'#EBEBEB':'#FFFFFF');}

.right-table08 td{font-size:12px;font-family:宋体;height:30px;}

.right-table08 {

text-decoration:none;

color:#555555;

font-family: "宋体";

}

.right-table08 a:visited{

text-decoration:none;

color:#555555;

}

.right-table09 {

float: left;

margin-right: 16px;

}

.right-table10 {

margin-top: 16px;

margin-bottom: 20px;

margin-left: 16px;

}

.right-font03 {

font-family: "宋体";

font-size: 12px;

color: #555555;

text-decoration: none;

padding-top: 4px;

padding-left: 2px;

text-align: right;

}

a.right-font03:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

}

.right-font04 {

font-family: "宋体";

font-size: 12px;

font-weight: bold;

color: #FF0000;

text-decoration: none;

}

a.right-font04:hover {

font-family: "宋体";

font-size: 12px;

font-weight: bold;

color: #FF0000;

text-decoration: none;

}

.right-font05 {

font-family: "宋体";

font-size: 12px;

color: #769ED9;

text-decoration: none;

line-height: 18px;

}

.right-font06 {

font-family: "宋体";

font-size: 12px;

color: #6C7494;

text-decoration: none;

padding-top: 3px;

}

.right-font07 {

font-family: "宋体";

font-size: 12px;

font-weight: bold;

color: #4EA1CB;

text-decoration: none;

filter:

Dropshadow(offx=1,offy=0,color=white)

Dropshadow(offx=0,offy=1,color=white)

Dropshadow(offx=0,offy=-1,color=white)

Dropshadow(offx=-1,offy=0,color=white);

}

.right-font08 {

font-family: "宋体";

font-size: 12px;

color: #555555;

text-decoration: none;

}

a.right-font08:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

}

.right-text09 {

font-family: Tahoma;

font-size: 11px;

font-weight: bold;

color: #F26521;

text-decoration: none;

}

.right-text10 {

font-family: "宋体";

font-size: 12px;

font-weight: bold;

color: #333333;

text-decoration: none;

}

.right-table11 {

border-right-width: 1px;

border-bottom-width: 1px;

border-left-width: 1px;

border-right-style: solid;

border-bottom-style: solid;

border-left-style: solid;

border-right-color: #6C7494;

border-bottom-color: #6C7494;

border-left-color: #6C7494;

}

.right-table12 {

text-align: center;

float: left;

margin-right: 3px;

background-image: url(../images/button11.gif);

height: 24px;

width: 86px;

font-family: "宋体";

font-size: 12px;

color: #555555;

font-weight: bold;

padding-top: 4px;

}

.right-table13 {

background-image: url(../images/button12.gif);

text-align: center;

float: left;

height: 24px;

width: 86px;

margin-right: 3px;

font-family: "宋体";

font-size: 12px;

color: #555555;

padding-top: 4px;

}

.right-table14 {

margin-top: 16px;

margin-bottom: 20px;

margin-left: 8px;

margin-right: 0px;

}

.right-table15 {

background-color: #82BEFF;

padding: 6px;

border: 1px solid #E2EFF5;

font-family: "宋体";

font-size: 13px;

color: #FFFFFF;

text-decoration: none;

line-height: 18px;

}

.right-table16 {

margin-bottom: 8px;

font-family: "宋体";

font-size: 12px;

color: #555555;

text-decoration: none;

}

.right-textfield01 {

width: 306px;

}

.right-textfield02 {

height: 21px;

width: 142px;

}

.right-textfield03 {

font-family: Tahoma;

font-size: 11px;

color: #555555;

text-decoration: none;

height: 14px;

}

.right-textfield04 {

font-family: "宋体";

font-size: 12px;

color: #3C3C3C;

text-decoration: none;

height: 14px;

width: 86px;

}

.right-textfield05 {

font-family: "宋体";

font-size: 12px;

color: #3C3C3C;

text-decoration: none;

height: 21px;

width: 235px;

}

.right-textfield06 {

font-family: "宋体";

font-size: 12px;

color: #3C3C3C;

text-decoration: none;

height: 14px;

width: 210px;

}

.right-textfield07 {

font-family: "宋体";

font-size: 12px;

color: #3C3C3C;

text-decoration: none;

height: 14px;

width: 35px;

}

.right-textfield08 {

font-family: "宋体";

font-size: 12px;

color: #3C3C3C;

text-decoration: none;

height: 14px;

width: 132px;

}

.right-button01 {

font-family: "宋体";

font-size: 12px;

color: #174B73;

height: 20px;

width: 63px;

background-image: url(../images/button03.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

}

.right-button02 {

font-family: "宋体";

font-size: 12px;

color: #174B73;

height: 20px;

width: 45px;

background-image: url(../images/button04.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

}

.right-button03 {

font-family: "宋体";

font-size: 12px;

color: #174B73;

height: 20px;

width: 123px;

background-image: url(../images/button05.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

}

.right-button04 {

font-family: "宋体";

font-size: 12px;

color: #174B73;

height: 20px;

width: 87px;

background-image: url(../images/button06.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

}

.right-button05 {

font-family: "宋体";

font-size: 12px;

color: #174B73;

height: 20px;

width: 22px;

background-image: url(../images/button07.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

}

.right-button06 {

font-family: "宋体";

font-size: 12px;

color: #174B73;

height: 20px;

width: 24px;

background-image: url(../images/button08.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

}

.right-button07 {

font-family: "宋体";

font-size: 12px;

color: #174B73;

height: 20px;

width: 75px;

background-image: url(../images/button09.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

}

.right-button08 {

font-family: "宋体";

font-size: 12px;

color: #174B73;

height: 20px;

width: 99px;

background-image: url(../images/button10.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

}

.login-text01 {

font-family: "宋体";

font-size: 12px;

color: #7F9DB9;

text-decoration: none;

}

a.login-text01:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

}

.login-text02 {

font-family: "宋体";

font-size: 13px;

color: #7F9DB9;

text-decoration: none;

text-align: right;

padding-top: 4px;

}

.login-text03 {

font-family: "宋体";

font-size: 12px;

color: #0E69C7;

text-decoration: underline;

}

a.login-text03:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

}

.login-text04 {

font-family: "宋体";

font-size: 12px;

color: #ACACAC;

text-decoration: none;

line-height: 20px;

}

a.login-text04:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

line-height: 20px;

}

.login-text05 {

font-family: "宋体";

font-size: 14px;

color: #6890D2;

text-decoration: none;

}

.login-text06 {

font-family: "宋体";

font-size: 12px;

font-weight: bold;

color: #FFFFFF;

text-decoration: none;

}

.newtable01 {

font-family: "宋体";

font-size: 14px;

font-weight: bold;

color: #FFFFFF;

text-decoration: none;

text-align: center;

float: left;

background-image: url(../images/navbutton01.gif);

margin-right: 2px;

}

.navtable02 {

font-family: "宋体";

font-size: 12px;

color: #000000;

text-decoration: none;

text-align: center;

float: left;

background-image: url(../images/navbutton02.gif);

margin-right: 2px;

}

.newfont01 {

font-family: "宋体";

font-size: 14px;

font-weight: bold;

color: #FFFFFF;

text-decoration: none;

}

a.newfont01:hover {

font-family: "宋体";

font-size: 14px;

font-weight: bold;

color: #FFFFFF;

text-decoration: underline;

}

.newfont02 {

font-family: "宋体";

font-size: 12px;

color: #000000;

text-decoration: none;

}

a.newfont02:hover {

font-family: "宋体";

font-size: 12px;

color: #000000;

text-decoration: underline;

}

.newfont03 {

font-family: "宋体";

font-size: 12px;

color: #000000;

text-decoration: none;

}

a.newfont03:hover {

font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: underline;

}

.newfont04 {

font-family: Arial, Helvetica, sans-serif;

font-size: 13px;

color: #000000;

text-decoration: line-through;

}

.newfont05 {

font-family: Arial, Helvetica, sans-serif;

font-size: 13px;

color: #000000;

text-decoration: none;

}

.newfont06 {

font-family: "宋体";

font-size: 14px;

color: #000000;

text-decoration: none;

}

.newfont07 {

font-family: "宋体";

font-size: 13px;

font-weight: bold;

color: #174B73;

text-decoration: none;

}

.newfont08 {

font-family: Arial, Helvetica, sans-serif;

font-size: 13px;

color: #FF0000;

text-decoration: none;

font-weight: bold;

}

.newfont09 {

font-family: "宋体";

font-size: 13px;

line-height: 20px;

color: #000000;

text-decoration: none;

}

.newfont10 {

font-family: "宋体";

font-size: 13px;

font-weight: bold;

color: #000000;

}

.navtable04 {

font-family: "宋体";

font-size: 12px;

color: #000000;

text-decoration: none;

text-align: center;

float: left;

background-image: url(../images/navbutton04.gif);

margin-right: 2px;

}

.newtable03 {

font-family: "宋体";

font-size: 14px;

font-weight: bold;

color: #FFFFFF;

text-decoration: none;

text-align: center;

float: left;

background-image: url(../images/navbutton03.gif);

margin-right: 2px;

}

.mxfont01 {

font-family: "宋体";

font-size: 12px;

line-height: 20px;

color: #333333;

text-decoration: none;

}

.dhfont01 {

font-family: "宋体";

font-size: 16px;

font-weight: bold;

color: #174B73;

text-decoration: none;

}

.dhfont02 {

font-family: "宋体";

font-size: 16px;

font-weight: bold;

color: #9D080D;

}

.dhfont03 {

font-family: "宋体";

font-size: 16px;

line-height: 24px;

color: #000000;

text-decoration: none;

}

.dhfont04 {

font-family: "宋体";

font-size: 14px;

line-height: 24px;

color: #000000;

text-decoration: none;

}

.dhfont05 {

font-family: "宋体";

font-size: 16px;

font-weight: bold;

color: #177B2F;

text-decoration: none;

}

.dhfont06 {

font-family: "宋体";

font-size: 16px;

font-weight: bold;

color: #FF0000;

}

.dhfont07 {

font-family: Arial, Helvetica, sans-serif;

font-size: 16px;

color: #FF0000;

}

.dhtextfiled {

font-family: "宋体";

font-size: 16px;

height: 30px;

color: #000000;

padding-top: 5px;

}

.dhbutton01 {

font-family: "宋体";

font-size: 16px;

color: #004A80;

text-align: center;

background-image: url(../images/dhbutton01.gif);

height: 30px;

width: 128px;

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

border-top-style: none;

border-right-style: none;

border-bottom-style: none;

border-left-style: none;

}

.dhbutton02 {

font-family: "宋体";

font-size: 16px;

color: #803600;

text-align: center;

height: 30px;

width: 128px;

background-image: url(../images/dhbutton02.gif);

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

border-top-style: solid;

border-right-style: solid;

border-bottom-style: solid;

border-left-style: solid;

}

.dhbutton03 {

font-family: "宋体";

font-size: 16px;

color: #80003C;

text-decoration: none;

background-image: url(../images/dhbutton03.gif);

height: 30px;

width: 128px;

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

border-top-style: solid;

border-right-style: solid;

border-bottom-style: solid;

border-left-style: solid;

}

.dhbutton04 {

font-family: "宋体";

font-size: 16px;

color: #000980;

background-image: url(../images/dhbutton04.gif);

height: 30px;

width: 128px;

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

border-top-style: solid;

border-right-style: solid;

border-bottom-style: solid;

border-left-style: solid;

}

.dhbutton05 {

font-family: "宋体";

font-size: 16px;

color: #004A80;

background-image: url(../images/dhbutton05.gif);

height: 30px;

width: 81px;

border-top-width: 0px;

border-right-width: 0px;

border-bottom-width: 0px;

border-left-width: 0px;

border-top-style: solid;

border-right-style: solid;

border-bottom-style: solid;

border-left-style: solid;

}

\HRmanagement\WebRoot\css\head.css

/\* $TopStyle:URL=css/head.css$ \*/

/\* 通用样式表v2.0 \*/

body,p,td,input,option,textarea,select{color:#333333;font:12px "Verdana"}

body,form,ul,li,p,ol{margin:0;padding:0}

body{background:#fff;text-align:center}

img{border:0}

ul{list-style-type:none}

table{text-align:left}

.full{margin:0 10px;text-align:left}

.main{margin:0 auto;text-align:left;width:760px}

.clear{border-top:1px solid transparent !important;border-top:0;clear:both}

.debug{border:1px dashed #FF00FF}

.font14{font-size:14px}

.fontOrange{color:#FF4E00}

.b{font-weight:bold}

.line{line-height:140%;word-break:break-all;word-wrap:break-word}

.desc{line-height:150%;word-break:break-all;word-wrap:break-word}

.lineCenter{text-align:center}

.lineLeft{text-align:left}

.lineRight{text-align:right}

a:link,a:visited{color:#1835D1;text-decoration:none}

a:active,a:hover{color:#FF4e00;text-decoration:underline}

/\* 头部和尾部样式表 \*/

#head{font-family:"宋体"}

#head a:link,#head a:visited,#foot a:link,#foot a:visited{color:#1835D1;text-decoration:none}

#head a:active,#head a:hover,#foot a:active,#foot a:hover{color:#FF4e00;text-decoration:underline}

#head{height:80px;margin:0 auto;position:relative;width:760px}

#head\_find{border-top:1px solid #D0D0D0;float:right;margin-top:1px;padding-top:4px;text-align:right;width:516px}

#head\_find li{float:right;margin-left:3px}

#head\_link{float:right;margin:6px 0 2px;width:500px}

#head\_link li{float:right;margin-left:10px}

#head\_logo{left:0;position:absolute;top:0}

#head\_navi{float:right;width:500px}

#head\_navi li{float:right;margin-left:10px}

#head\_url{border-top:1px solid #D0D0D0;color:#808080;float:right;font-family:Verdana;margin-top:1px;padding:2px 21px 0 0;text-align:right;width:516px}

/\* 头部样式表V2 \*/

#headV2 a:link,#headV2 a:visited,#foot a:link,#foot a:visited{color:#1835D1;text-decoration:none}

#headV2 a:active,#headV2 a:hover,#foot a:active,#foot a:hover{color:#FF3300;text-decoration:underline}

#headV2 #headMenuSimple li a{color:#333;font-family:"宋体"}

#headV2 #menuFind a{color:#fff;text-decoration:underline}

#headV2 #menuFind a:hover{color:#ff4e00}

#foot{margin:10px 0 12px;text-align:center;line-height:150%}

#foot\_link{border-bottom:1px solid #D7D7D7;margin:7px auto;padding-bottom:4px;width:760px}

#headV2{margin:7px}

#headMain{width:760px;text-align:right;height:66px;overflow:hidden;position:relative;margin:0 auto}

#headLogo{position:absolute;top:0;left:0}

#headLink{background:url(http://pics.paipai.com/common/menu/line.gif) right bottom no-repeat;height:18px;font-family:Verdana}

#headInfo{padding:4px 40px 0 0;color:#545454;height:20px;text-align:right;font-family:Verdana}

#headMenu{padding-left:240px}

#headMenuSimple{margin-left:400px;background:#f0f0f0 url(http://pics.paipai.com/common/menu/menu\_left.gif) left no-repeat;height:21px}

#headMenuSimple ul{background:url(http://pics.paipai.com/common/menu/menu\_side.gif) right no-repeat;height:21px;padding-left:7px}

#headMenuSimple li{float:left;background:url(http://pics.paipai.com/common/menu/menu\_main.gif) right no-repeat;padding:5px 7px 0 3px;height:16px}

#headFind{height:34px;width:760px;margin:0 auto;overflow:hidden}

#headFindSimple{padding:3px 0}

#headForm{height:26px;padding:8px 20px 0;text-align:left;color:#fff}

.findInto{border:1px solid #808080;color:gray;height:16px;width:240px}

.findGoto{border:none;height:20px;background:url(http://pics.paipai.com/common/menu/find\_bg.gif) no-repeat}

#headV2 #menuItem{height:24px;font-family:"宋体"}

#headV2 #menuItem a{background:url(http://pics.paipai.com/common/menu/main\_left.gif) #EBEBEB left top no-repeat;display:block;float:left;height:24px;text-decoration:none}

#menuItem a span{background:transparent url(http://pics.paipai.com/common/menu/main\_side.gif) right top no-repeat;color:#000000;cursor:hand;display:block;float:left;height:17px;padding:7px 13px 0;text-align:center}

#menuItem li{float:left;height:24px;margin-right:3px}

#menuItem li#menuBook a:hover{background:url(http://pics.paipai.com/common/menu/book\_left.gif) #1D7D8B left top no-repeat}

#menuItem li#menuBook a:hover span{background:transparent url(http://pics.paipai.com/common/menu/book\_side.gif) right top no-repeat;color:#FFFFFF}

#menuItem li#menuDigital a:hover{background:url(http://pics.paipai.com/common/menu/digital\_left.gif) #8B94A2 left top no-repeat}

#menuItem li#menuDigital a:hover span{background:transparent url(http://pics.paipai.com/common/menu/digital\_side.gif) right top no-repeat;color:#FFFFFF}

#menuItem li#menuGame a:hover{background:url(http://pics.paipai.com/common/menu/game\_left.gif) #784544 left top no-repeat}

#menuItem li#menuGame a:hover span{background:transparent url(http://pics.paipai.com/common/menu/game\_side.gif) right top no-repeat;color:#FFFFFF}

#menuItem li#menuHome a:hover{background:url(http://pics.paipai.com/common/menu/home\_left.gif) #FFA730 left top no-repeat}

#menuItem li#menuHome a:hover span{background:transparent url(http://pics.paipai.com/common/menu/home\_side.gif) right top no-repeat;color:#FFFFFF}

#menuItem li#menuLady a:hover{background:url(http://pics.paipai.com/common/menu/lady\_left.gif) #F0447D left top no-repeat}

#menuItem li#menuLady a:hover span{background:transparent url(http://pics.paipai.com/common/menu/lady\_side.gif) right top no-repeat;color:#FFFFFF}

#menuItem li#menuQ a:hover{background:url(http://pics.paipai.com/common/menu/q\_left.gif) #32C1EC left top no-repeat}

#menuItem li#menuQ a:hover span{background:transparent url(http://pics.paipai.com/common/menu/q\_side.gif) right top no-repeat;color:#FFFFFF}

#menuItem li#menuSale a:hover{background:url(http://pics.paipai.com/common/menu/sale\_left.gif) #97C03D left top no-repeat}

#menuItem li#menuSale a:hover span{background:transparent url(http://pics.paipai.com/common/menu/sale\_side.gif) right top no-repeat;color:#FFFFFF}

#menuItem li#menuSport a:hover{background:url(http://pics.paipai.com/common/menu/sport\_left.gif) #60B900 left top no-repeat}

#menuItem li#menuSport a:hover span{background:transparent url(http://pics.paipai.com/common/menu/sport\_side.gif) right top no-repeat;color:#FFFFFF}

#menuItem li#menuStudent a:hover{background:url(http://pics.paipai.com/common/menu/student\_left.gif) #FF6100 left top no-repeat}

#menuItem li#menuStudent a:hover span{background:transparent url(http://pics.paipai.com/common/menu/student\_side.gif) right top no-repeat;color:#FFFFFF}

#headFind{background:url(http://pics.paipai.com/common/menu/home\_find\_left.gif) #FFA730 left top no-repeat}

#headForm{background:transparent url(http://pics.paipai.com/common/menu/home\_find\_side.gif) right top no-repeat}

\HRmanagement\WebRoot\css\style.css

body{

margin:0px;

}

html {

scrollbar-face-color:#CCD0DB;

scrollbar-highlight-color:#fff;

scrollbar-shadow-color:#fff;

scrollbar-3dlight-color:#6795B4;

scrollbar-arrow-color:#20669A;

scrollbar-track-color:#fff;

scrollbar-darkshadow-color:#6795B4;

border:0;

}

html { overflow-x: hidden; overflow-y: auto; }

TD{

font-size: 12px;

}

a:link,a:visited{

text-decoration:none;

color:#006;

}

a:hover{

text-decoration:underline;

color:#cc0000;

}

.headbody{

background:url('../images/HeadBack.jpg')

}

.headmain{

height:70px;

border-bottom:1px solid #014B56;

}

.menuBody{

background:url('../images/menuB.jpg') repeat-y;

text-align:left;

}

.menuUL{

list-style:none;

margin:0px;

padding:0px;

}

.menuUL li{

display:block;

margin-bottom:5px;

}

.menuA:link,.menuA:visited{

width:100%;

height:35px;

display:block;

background:url('../images/menubutton.jpg') no-repeat 4px 0px;

font-size:12px;

color:#fff;

text-decoration:none;

}

.menuA:hover{

background:url('../images/menubutton.jpg') no-repeat 4px -36px;

color:#fff;

text-decoration:none;

}

.menuAS:link,.menuAS:visited{

width:100%;

height:35px;

display:block;

background:url('../images/menubutton.jpg') no-repeat 4px -72px;

font-size:12px;

color:#000;

text-decoration:none;

font-weight:bold;

}

.ContentBody{

background:#F6F6F6 url(../images/mainbg.gif);

/\*url(../images/mainback.jpg) repeat-x fixed; \*/

margin:15px;

}

.MainDiv{}

.MenuIcon{

margin:1px 6px -12px 12px;

}

.CTitle{

background:#6795B4;

padding:5px;

text-align:left;

color:#FFFFFF;

font-size:13px;

font-family: Verdana, Arial, Helvetica, sans-serif;

}

.CContent{

border-top: 1px solid #5F8FB0;

border-right: 1px solid #5F8FB0;

border-bottom: 1px solid #5F8FB0;

border-left: 1px solid #5F8FB0;

background-color: #FFFFFF;

}

.CPanel{

padding:5px;

font-size:12px;

font-family: Verdana, Arial, Helvetica, sans-serif;

}

legend{

color: #014E82;

font-weight:bold;

padding:5px;

}

fieldset{

border-top: 1px solid #C8D9E4;

border-right: 1px solid #C8D9E4;

border-bottom: 1px solid #C8D9E4;

border-left: 1px solid #C8D9E4;

margin-top: 10px;

margin-left:10px;

margin-right:10px;

}

fieldset div{

padding:0px;

}

fieldset td {

font-size:12px;

}

select{font-size:12px;}

input.text{

border:1px solid #7F9DB9;font-size:12px;font-family: Verdana, Arial, Helvetica, sans-serif;height:18px

}

button{background:#fff;border:1px solid #C28F27;padding:2px 2px 0px 2px;margin:4px;border-width:1px 3px 1px 3px;font-size: 12px;color: #A74300;}

input.button{

background:#fff;border:1px solid #C28F27;padding:2px 2px 0px 2px;margin:4px;border-width:1px 3px 1px 3px;font-size: 12px;color: #A74300;

}

textarea{ border:1px solid #7F9DB9;font-size:12px;font-family: Verdana, Arial, Helvetica, sans-serif;}

.SubMenu{

padding: 4px;

border-bottom: 1px solid #5F8FB0;

margin:-5px -5px 2px -5px;

background-color: #EBF1F5;

}

.SubButton{

margin-top: 6px;

padding: 4px;

text-align:center;

}

.TDHead{

/\*background:url(../images/M.jpg);\*/

border-bottom:1px solid #5F8FB0;

height:16px;

font-weight:bold;

background-color: #C5D7E2;

}

.TablePanel{

/\* border:1px solid #999;\*/

background-color: #DBE0E6;

}

.TablePanel td{

font-size:12px;

/\*background-color: #fff;\*/

}

.dtxt{

font-family:verdana;

font-size:10px;

font-weight: normal;

padding-right:2px;

color: #808080;

}

.ctxt{

border:1px solid #DBE0E6;

background-color: #F6F8F9;

font-size:11px;

padding:2px;

color:#000;

text-justify :newspaper;

text-align:justify;

}

.ctxt UL {

list-style:none;

font-size:11px;

padding:0;

border:0;

margin:1px 1px 1px 1px;

}

.tblist{

padding:2px;

color:#003366;

}

/\*topbt\*/

#nav {

height:70px;

list-style:none;

float:right;

margin-top:0px;

width:369px;

padding-right:70px;

background:url(../images/pic4.jpg);

}

#nav li{

float:left;

font-size:11px;

font-family:verdana;

color:#fff;

}

#nav li a{

color:#FFFFFF;

text-decoration:none;

display:block;

width:40px;

height:70px;

text-align:center;

margin-left:40px;

}

#nav li a:hover{

color:#C2E0FC;

}

.btpading{

padding-top:54px;

}

#nav li #topbt1{

text-align:center;

background:url(../images/pic4.jpg);

background-position:-3px 0px;

}

#nav li #topbt2{

text-align:center;

background:url(../images/pic4.jpg);

background-position:-83px 0px;

}

#nav li #topbt3{

text-align:center;

background:url(../images/pic4.jpg);

background-position:-165px 0px;

}

#nav li #topbt4{

text-align:center;

background:url(../images/topbt.jpg);

background-position:-245px 0px;

}

#nav li #topbt5{

text-align:center;

background:url(../images/pic4.jpg);

background-position:-325px 0px;

}

/\*\*/

#datatable tr{wfsr:expression(this.bgColor=this.rowIndex%2?'#ffffff':'#FDFAF0')}

#datatable td{font-size:11px;font-family:verdana;height:19px}

#datatable a:link,a:visited{

text-decoration:none;

color:#0033CC;

}

#datatable a:hover{

text-decoration:underline;

color:#cc0000;

}

/\*\*/

A.p:link,A.p:active,A.p:visited{

TEXT-DECORATION:none;

Color:#666;

FONT-SIZE: 11px;

}

A.p:hover{

TEXT-DECORATION: none;

Color:#6795B4;

FONT-SIZE: 11px;

position: relative; left: 1px; top: 1px;

}

.pagetxt{

border:1px solid #666;

border-right: 0px solid #666;

font-size:11px;font-family: Verdana, Arial, Helvetica, sans-serif;

height:14px;

text-align: center;

color:#2C3E88;

width:23px;

}

.pagebt{

border:1px solid #666;

font-size:11px;font-family: Verdana, Arial, Helvetica, sans-serif;

height:18px;

color:#2C3E88;

background-color: #EBF1F5;

width:25px;

}

#page TD{

font-size:11px;font-family: Verdana, Arial, Helvetica, sans-serif;

Color:#666;

}

.webtxt{

text-align:left;

overflow:hidden;

margin:12px;

font-size:14px;

clear:both;

line-height: 160%;

word-wrap:break-word;

word-break:break-all;

text-justify :newspaper;

text-align:justify;

}

.sbt{

border:1px solid #7F9DB9;

font-size:12px;

font-family: Verdana, Arial, Helvetica, sans-serif;

height:22px;

}

.file\_input{

border:1px solid #7F9DB9;

font-size:12px;

font-family: Verdana, Arial, Helvetica, sans-serif;

height:22px;

}

A.btlink:link,A.btlink:visited{

text-decoration:none;

color:#006600

}

A.btlink:hover{

text-decoration:underline;

color:#cc0000;

}

ul.TabBarLevel{

list-style:none;

margin:0;

padding:0;

height:29px;

background-image:url(../images/tabbar\_level1\_bk.gif);

}

ul.TabBarLevel li{

float:left;

padding:0;

height:29px;

margin-right:1px;

background:url(../images/tb\_left\_bk.gif) left top no-repeat;

}

ul.TabBarLevel li a{

display:block;

line-height:29px;

padding:0 20px;

color:#006;

background:url(../images/tb\_right\_bk.gif) right top no-repeat;

white-space: nowrap;

/\*font-weight: bold;\*/

}

ul.TabBarLevel li.Selected{

background:url(../images/tb\_selected\_left\_bk.gif) left top no-repeat;

}

ul.TabBarLevel li.Selected a{

background:url(../images/tb\_selected\_right\_bk.gif) right top no-repeat;

}

ul.TabBarLevel li a:link,ul.TabBarLevel li a:visited{

color:#006;

text-decoration:none;

}

ul.TabBarLevel li a:hover,ul.TabBarLevel li a:active{

color:#0033CC;

text-decoration:none;

}

ul.TabBarLevel li.Selected a:link,ul.TabBarLevel li.Selected a:visited{

color:#cc0000;

text-decoration:none;

font-weight: bold;

}

ul.TabBarLevel li.Selected a:hover,ul.TabBarLevel li.Selected a:active{

color:#cc0000;

text-decoration:none;

}

div.HackBox {

padding : 5px 5px;

border-left: 1px solid #b1b1b1;

border-right: 1px solid #b1b1b1;

border-bottom: 1px solid #b1b1b1;

display:none;

margin:0;

line-height:1.5em;

font-size:12px;

/\*height:expression(document.body.clientHeight-80);\*/

}

.sumfont{

font-size:12px;

color:#666666;

padding-top:5px;

}

.reddoc{

position: relative;

bottom:-2;

padding-top:3px;

}

.tfont{

font-weight:bold;

font-size:12px;

color:#000000;FILTER: dropshadow(color=#ffffff,offx=1,offy=1,positive=1);

letter-spacing:1;

padding-top:5px;

}

.atfont{

font-size:12px;

color:#666666;FILTER: dropshadow(color=#ffffff,offx=1,offy=1,positive=1);

letter-spacing:1;

padding-top:5px;

padding-right:5px;

}

.gfont{

color: #666666;

}

.gtfont{

color: #666600;

letter-spacing:2;

font-weight: bold;

}

.mainlinkbg{

background-color: #ffffff;

/\*BORDER-RIGHT: #A3C7E2 1px solid;

BORDER-TOP: #A3C7E2 0px solid;

FONT-SIZE: 9pt;

BORDER-LEFT: #A3C7E2 1px solid;

BORDER-BOTTOM:#A3C7E2 1px solid;\*/

}

.main{

margin:0;

font-size:12px;

width: 100%;

margin-top:2px;

margin-bottom: 5px;

}

.main ul{

}

.main li{

width:390px;

float:right;

padding:3px 0px 0px 5px;

/\*background: url(../images/dott.gif) transparent no-repeat 3px 7px;\*/

height:17px;

overflow : hidden;

text-overflow:ellipsis;

border-bottom-width: 1px;

border-bottom-style: dotted;

border-bottom-color: #C0C0C0;

margin-top: 2px;

color:#666666;

text-overflow:ellipsis;

white-space:nowrap;

}

.main li a{

float:left;

width:310px;

text-overflow:ellipsis;/\*文字超长则用省略号\*/

white-space:nowrap;

overflow : hidden;

}

.main a:link,a:visited{

text-decoration:none;

color:#003288;

}

.main a:hover{

text-decoration:underline;

color:#cc0000;

}

.select\_tr{

background-color: #EAFDDB;

}

div{

font-size:12px;

font-family: Verdana, Arial, Helvetica, sans-serif;

}

/\*整合的CSS\*/

.tablestyle\_title{

/\*height: 20px; font-style: normal; font-variant: normal; font-weight: bold; font-size: 16px;letter-spacing: 2px; text-align: middle; border-top: 2px solid #ff7300; padding-left: 10px; background-color: #f0f0f0;text-align:center;\*/

background:#6795B4;

padding:5px;

text-align:left;

color:#FFFFFF;

font-family: Verdana, Arial, Helvetica, sans-serif;

height: 20px;

font-style: normal; font-variant: normal; font-weight: bold; font-size: 16px;letter-spacing: 2px; padding-left: 10px;

}

#datatabdiv .smdiv{

border:1px solid #CCEFF5;

background-color: #FAFCFD;

padding:5px;

line-height:18px;

color:#666;

width:98%;

margin-bottom:5px;

text-justify :newspaper;

text-align:justify;

}

#datatabdiv .tabinfodiv{

border:1px solid #E4E298;

background-color: #FFFFDD;

padding:5px;

line-height:18px;

color:#666;

width:98%;

margin-bottom:5px;

text-justify :newspaper;

text-align:justify;

}

.infodiv{

border:1px solid #E4E298;

border-top:2px solid #FF7300;

background-color: #FFFFDD;

padding:5px;

line-height:18px;

color:#666;

width:98%;

margin-bottom:5px;

text-justify :newspaper;

text-align:justify;

}

.red{color: #f00;}

.red2{color: #DF131F;}

.tablestyle\_td1{

border-top:#ffffff 3px solid; border-bottom:#ffffff 1px solid; background:#ffffff;

}

.tablestyle\_td2{

border-bottom:#ffffff 1px solid; background:#fff;

}

.tablestyle\_td3{border-top:#fff 1px solid; background:#ffffff;

}

.tablestyle\_td4{border-bottom:#fff 1px solid; background:#FFF;

}

\HRmanagement\WebRoot\css\style0.css

#FullPath{border-bottom:3px solid #FF9900}

#name{border-bottom:1px solid #D6D6D6}

#UserInfo{background:#F8F8F8;border:1px solid #D8D8D8}

#UserInfoHead{background:#EDEDED;border-bottom:1px solid #d8d8d8}

#UserInfoFoot{background:#EDEDED;border-top:1px solid #d8d8d8}

.TitleInfo{left:30px;top:6px}

.TitleLink{right:60px;top:6px;width:120px}

.TitleTop{right:0;top:5px;height:24px;width:60px}

#commend\_list{border:1px solid #BEC6D9;border-top:none}

#DescShow{border:1px solid #FF9900;border-top:none}

#auction\_list{background:transparent url(../images/menuMain.gif) repeat-x;height:28px;position:relative}

#auction\_list li{float:left;height:25px;margin-right:3px}

#auction\_list li a{background:transparent url(../images/menuWillMain.gif) left top no-repeat;display:block;float:left;height:25px;text-decoration:none}

#auction\_list li a span{background:transparent url(../images/menuWillSide.gif) right top no-repeat;color:#000000;cursor:hand;display:block;float:left;height:18px;padding:7px 20px 0;text-align:center}

#auction\_list li a:hover span{background:transparent url(../images/menuHereSide.gif) right top no-repeat;color:#FFFFFF}

#auction\_list li a:hover,#auction\_list li.li\_now a{background:transparent url(../images/menuHereMain.gif) left top no-repeat}

#auction\_list li.li\_now a span{background:transparent url(../images/menuHereSide.gif) right top no-repeat;color:#FFFFFF;font-weight:bold}

#main{margin:0 10px 0 210px;border:1px dashed #FFFFFF;text-align:left}

\HRmanagement\WebRoot\css\tab.js

function IsShow()

{

parent.document.all.layMnAboutPingSoft.style.visibility = 'visible';

}

function IsHide()

{

parent.document.all.layMnAboutPingSoft.style.visibility = 'hidden';

}

function ReLocation(url)

{

parent.document.location.href=url

//parent.document.all.CurrentLocation.innerText = CurrentLocationCaption;

//parent.document.location.href=url

}

function mOvr(src,clrOver) { if (!src.contains(event.fromElement)) { src.bgColor = '#ED1B23'; src.style.color = '#555555'}}

function mOut(src,clrIn,font) { if (!src.contains(event.toElement)) { src.bgColor = '#CCCCCC'; src.style.color = '#555555'}}

function mClk(src) { if(event.srcElement.tagName=='TD'){src.children.tags('A')[0].click();} }

function MM\_callJS(jsStr) { //v2.0

return eval(jsStr)

}

\HRmanagement\WebRoot\Js\js.js

//function test()

//{

// alert

//}

function stipendValidate(){

if(stipendForm.name.value==""){

alert("请添写员工姓名！");

stipendForm.name.focus();

return false;

}

if(stipendForm.basic.value==""){

alert("请添写基本薪金！");

stipendForm.basic.focus();

return false;

}

if(isNaN(stipendForm.basic.value))

{

alert("基本薪金只能填写数字！");

stipendForm.basic.focus();

return false;

}

if(stipendForm.granttime.value==""){

alert("请添写发放时间！");

stipendForm.granttime.focus();

return false;

}

if(!checkIsValidDate(stipendForm.granttime.value)){

alert("输入的时间不合法！（年-月-日）");

stipendForm.granttime.focus();

return false;

}

return true;

}

function institutionValidate(){

if(institutionForm.name.value==""){

alert("请添写奖惩名称！");

institutionForm.name.focus();

return false;

}

if(institutionForm.reason.value==""){

alert("请添写奖惩原因！");

institutionForm.reason.focus();

return false;

}

return true;

}

function userValidate(){

if(userForm.username.value==""){

alert("请添写人员姓名！");

userForm.username.focus();

return false;

}

// if(!/^[\u4e00-\u9fa5]+$/gi.exec(userForm.username.value) || !/^\\w+$/gi.exec(userForm.username.value)){

// alert("用户请输入中文！");

// userForm.username.focus();

// return false;

// }

if(userForm.password.value==""){

alert("请添写登录密码！");

userForm.password.focus();

return false;

}

if(escape(userForm.password.value).indexOf("%u")>=0 ){

alert("密码不能包含中文！");

return false;

}

if(userForm.birthday.value==""){

alert("请添写出生日期！");

userForm.birthday.focus();

return false;

}

if(CheckDate(userForm.birthday.value)){

alert("输入的日期不合法！（年-月-日）");

userForm.birthday.focus();

return false;

}

return true;

}

function educateValidate(){

if(educateForm.name.value==""){

alert("请添写培训名称！");

educateForm.name.focus();

return false;

}

if(educateForm.purpose.value==""){

alert("请添写培训目的！");

educateForm.purpose.focus();

return false;

}

if(educateForm.begintime.value==""){

alert("请添写培训开始时间！");

educateForm.begintime.focus();

return false;

}

if(!checkIsValidDate(educateForm.begintime.value)){

alert("输入的时间不合法！（年-月-日）");

educateForm.begintime.focus();

return false;

}

if(educateForm.endtime.value==""){

alert("请添写培训结束时间！");

educateForm.endtime.focus();

return false;

}

if(!checkIsValidDate(educateForm.endtime.value)){

alert("输入的时间不合法！（年-月-日）");

educateForm.endtime.focus();

return false;

}

if(!checkDateEarlier(educateForm.begintime.value,educateForm.endtime.value)){

alert("起始日期不能晚于结束日期！");

educateForm.begintime.focus();

return false;

}

if(educateForm.teacher.value==""){

alert("请添写培训讲师！");

educateForm.teacher.focus();

return false;

}

if(educateForm.student.value==""){

alert("请添写培训人员！");

educateForm.student.focus();

return false;

}

return true;

}

function checkDateEarlier(strStart,strEnd)

{

if(checkIsValidDate(strStart) == false || checkIsValidDate(strEnd) == false)

return false;

//如果有一个输入为空，则通过检验

if (( strStart == "" ) || ( strEnd == "" ))

return true;

var arr1 = strStart.split("-");

var arr2 = strEnd.split("-");

var date1 = new Date(arr1[0],parseInt(arr1[1].replace(/^0/,""),10) - 1,arr1[2]);

var date2 = new Date(arr2[0],parseInt(arr2[1].replace(/^0/,""),10) - 1,arr2[2]);

if(arr1[1].length == 1)

arr1[1] = "0" + arr1[1];

if(arr1[2].length == 1)

arr1[2] = "0" + arr1[2];

if(arr2[1].length == 1)

arr2[1] = "0" + arr2[1];

if(arr2[2].length == 1)

arr2[2]="0" + arr2[2];

var d1 = arr1[0] + arr1[1] + arr1[2];

var d2 = arr2[0] + arr2[1] + arr2[2];

if(parseInt(d1,10) > parseInt(d2,10))

return false;

else

return true;

}//~~~

function checkIsValidDate(str)

{

//如果为空，则通过校验

if(str == "")

return true;

var pattern = /^((\d{4})|(\d{2}))-(\d{1,2})-(\d{1,2})$/g;

if(!pattern.test(str)){

return false;

}

var arrDate = str.split("-");

if(parseInt(arrDate[0],10) < 100)

arrDate[0] = 2000 + parseInt(arrDate[0],10) + "";

var date = new Date(arrDate[0],(parseInt(arrDate[1],10) -1)+"",arrDate[2]);

if(date.getYear() == arrDate[0]

&& date.getMonth() == (parseInt(arrDate[1],10) -1)+""

&& date.getDate() == arrDate[2]){

return true;

}else{

return false;

}

}//~~~

function CheckDate(INDate)

{ if (INDate=="")

{return true;}

subYY=INDate.substr(0,4)

if(isNaN(subYY) || subYY<=0){

return true;

}

//转换月份

if(INDate.indexOf('-',0)!=-1){ separate="-"}

else{

if(INDate.indexOf('/',0)!=-1){separate="/"}

else {return true;}

}

area=INDate.indexOf(separate,0)

subMM=INDate.substr(area+1,INDate.indexOf(separate,area+1)-(area+1))

if(isNaN(subMM) || subMM<=0){

return true;

}

if(subMM.length<2){subMM="0"+subMM}

//转换日

area=INDate.lastIndexOf(separate)

subDD=INDate.substr(area+1,INDate.length-area-1)

if(isNaN(subDD) || subDD<=0){

return true;

}

if(eval(subDD)<10){subDD="0"+eval(subDD)}

NewDate=subYY+"-"+subMM+"-"+subDD

if(NewDate.length!=10){return true;}

if(NewDate.substr(4,1)!="-"){return true;}

if(NewDate.substr(7,1)!="-"){return true;}

var MM=NewDate.substr(5,2);

var DD=NewDate.substr(8,2);

if((subYY%4==0 && subYY%100!=0)||subYY%400==0){ //判断是否为闰年

if(parseInt(MM)==2){

if(DD>29){return true;}

}

}else{

if(parseInt(MM)==2){

if(DD>28){return true;}

}

}

var mm=new Array(1,3,5,7,8,10,12); //判断每月中的最大天数

for(i=0;i< mm.length;i++){

if (parseInt(MM) == mm[i]){

if(parseInt(DD)>31){return true;}

}else{

if(parseInt(DD)>30){return true;}

}

}

if(parseInt(MM)>12){return true;}

return false;

}

function jobValidate(){

//alert(jobForm.job.value);

//return false;

if(jobForm.name.value==""){

alert("请添写姓名！");

jobForm.name.focus();

return false;

}

if(jobForm.job.value==""){

alert("请添写职位！");

jobForm.job.focus();

return false;

}

if(jobForm.specialty.value==""){

alert("请添写所学专业！");

jobForm.specialty.focus();

return false;

}

if(jobForm.experience.value==""){

alert("请添写工作经验！");

jobForm.experience.focus();

return false;

}

if(jobForm.studyeffort.value==""){

alert("请添写学历！");

jobForm.studyeffort.focus();

return false;

}

if(jobForm.school.value==""){

alert("请添写毕业学校！");

jobForm.school.focus();

return false;

}

if(jobForm.tel.value==""){

alert("请添写电话！");

jobForm.tel.focus();

return false;

}

if(isNaN(jobForm.tel.value))

{

alert("电话只能填写数字！");

jobForm.tel.focus();

return false;

}

if(jobForm.email.value==""){

alert("请添写Email！");

jobForm.email.focus();

return false;

}

var email = jobForm.email.value;

if (email != "") {

var reg = /^\w+((-\w+)|(\.\w+))\*\@[A-Za-z0-9]+((\.|-)[A-Za-z0-9]+)\*\.[A-Za-z0-9]+$/;

isok= reg.test(email );

if (!isok) {

alert("邮箱格式不正确，请重新输入！");

return false;

}

};

return true;

}

function adminChecked(){

if(userForm.isadminhelp.checked){

document.userForm.isadmin.value="1";

}else{

document.userForm.isadmin.value="0";

}

}

function performanceValidate(){

if(performanceForm.score.value==""){

alert("请添写绩效分数！");

performanceForm.score.focus();

return false;

}

if(isNaN(performanceForm.score.value))

{

alert("绩效分数只能填写数字！");

performanceForm.score.focus();

return false;

}

if(performanceForm.addmoney.value==""){

alert("请添写增加工资金额！");

performanceForm.addmoney.focus();

return false;

}

if(isNaN(performanceForm.addmoney.value))

{

alert("增加工资金额只能填写数字！");

performanceForm.addmoney.focus();

return false;

}

if(performanceForm.content.value==""){

alert("请添写绩效说明！");

performanceForm.content.focus();

return false;

}

if(CheckDate(performanceForm.createtime.value)){

alert("输入的日期不合法！（年-月-日）");

performanceForm.createtime.focus();

return false;

}

return true;

}

\HRmanagement\WebRoot\Js\typem.js

var \_D //div

var div\_Top; //层顶坐标

var div\_Id; //层的名字

var div\_Left; //层左边坐标

var div\_Width; //层的宽度

var div\_Height; //层的高度

var doc\_Height; //窗口当前总高度

var doc\_Width; //窗口当前宽度

//style对象中的属性

var \_style;

var style\_Top;

var style\_Left;

var T = function (divId){

alert(divId);

//alert("T");

init(divId);

style\_Top=(arrayPageSize[1] - div\_Height)/2+arrayScroll[1];

style\_Left=(arrayPageSize[0] - div\_Width)/2+arrayScroll[0];

\_style.display="";

}

//获取属性

var init = function (divId){

\_D = document.getElementById(divId);

div\_Top = parseInt(document.getElementById(divId).style.top,10) //获取高

div\_Left = parseInt(document.getElementById(divId).style.left,10) //div的y坐标

div\_Height = parseInt(document.getElementById(divId).offsetHeight,10) //div的高度

div\_Width = parseInt(document.getElementById(divId).offsetWidth,10) //div的宽度

arrayScroll = getPageScroll();

arrayPageSize = getPageSize();

\_style = divId.style;

style\_Top= divId.style.top;

style\_Left= divId.style.left;

}

//增加的方法

var addTM = function (x,y){

//获取值

var td1 = document.getElementById("gztype").value;

var td2 = document.getElementById("jinge").value;

//表格插入

var \_T = document.getElementById("t2");

var R\_ind = \_T.rows.Length;

var objTR = objTable.insertRow(R\_ind);

var objTD\_1= objTR.insertCell(0);

var objTD\_2= objTR.insertCell(1);

objTD\_1.innerHTML="<td>"+td1+"</td>";

objTD\_2.innerHTML="<td>"+td2+"</td>";

}

//获取getPageSize

var getPageSize = function () {

var de = document.documentElement;

var w = window.innerWidth || self.innerWidth || (de&&de.clientWidth) || document.body.clientWidth;

var h = window.innerHeight || self.innerHeight || (de&&de.clientHeight) || document.body.clientHeight

arrayPageSize = new Array(w,h)

return arrayPageSize;

}

//获取pageScrollTop

var getPageScroll = function (){

var yScrolltop;

var xScrollleft;

if (self.pageYOffset || self.pageXOffset) {

yScrolltop = self.pageYOffset;

xScrollleft = self.pageXOffset;

} else if (document.documentElement && document.documentElement.scrollTop || document.documentElement.scrollLeft ){ // Explorer 6 Strict

yScrolltop = document.documentElement.scrollTop;

xScrollleft = document.documentElement.scrollLeft;

} else if (document.body) {// all other Explorers

yScrolltop = document.body.scrollTop;

xScrollleft = document.body.scrollLeft;

}

arrayPageScroll = new Array(xScrollleft,yScrolltop)

return arrayPageScroll;

}

\HRmanagement\WebRoot\Js\xiangmu.js

var init = function(){

var xiangmustyle = document.getElementById("xiangmu");

alert(xiangmustyle);

xiangmustyle.style.display = "";

}

//获取getPageSize

var getPageSize = function () {

var de = document.documentElement;

var w = window.innerWidth || self.innerWidth || (de&&de.clientWidth) || document.body.clientWidth;

var h = window.innerHeight || self.innerHeight || (de&&de.clientHeight) || document.body.clientHeight

arrayPageSize = new Array(w,h)

return arrayPageSize;

}

//获取pageScrollTop

var getPageScroll = function (){

var yScrolltop;

var xScrollleft;

if (self.pageYOffset || self.pageXOffset) {

yScrolltop = self.pageYOffset;

xScrollleft = self.pageXOffset;

} else if (document.documentElement && document.documentElement.scrollTop || document.documentElement.scrollLeft ){ // Explorer 6 Strict

yScrolltop = document.documentElement.scrollTop;

xScrollleft = document.documentElement.scrollLeft;

} else if (document.body) {// all other Explorers

yScrolltop = document.body.scrollTop;

xScrollleft = document.body.scrollLeft;

}

arrayPageScroll = new Array(xScrollleft,yScrolltop)

return arrayPageScroll;

}

\HRmanagement\WebRoot\addeducate.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="educateForm" method="post" action="modifyeducate.do?action=addeducate" onsubmit="return educateValidate();">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >培训计划录入</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>培训计划</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">培训名称：</td>

<td colspan="3"><input name="name" type="text" class="input"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">培训目的：</td>

<td colspan="3"><input type="text" name="purpose" class="input"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">培训开始时间：</td>

<td width="29%"><input type="text" name="begintime" class="input"/>

<span class="red">\*</span></td>

<td width="18%">培训结束时间：</td>

<td width="42%"><input type="text" name="endtime" class="input"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">讲师：</td>

<td><input type="text" name="teacher" class="input"/>

<span class="red">\*</span></td>

<td>培训人员：</td>

<td><input type="text" name="student" class="input"/>

<span class="red">\*</span>

</td>

</tr>

<tr>

<td width="11%" nowrap align="right">培训材料：</td>

<td colspan="3"><textarea name="datum" cols="100" rows="6" class="input"></textarea></td>

</tr>

</table>

</fieldset>

</TD>

</TR>

</TABLE>

</td>

</tr>

<TR>

<TD colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</TR>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\addinstitution.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="institutionForm" method="post" action="modifyinstitution.do?action=addinstitution" onSubmit="return institutionValidate()">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >奖惩信息登记</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>奖惩信息</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">奖惩名称：</td>

<td width="89%"><input name="name" type="text" class="input"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">奖惩原因：</td>

<td><input name="reason" type="text" class="input"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td width="11%" nowrap align="right">奖惩说明：</td>

<td><textarea name="explain" cols="100" rows="6" class="input" id="explain"></textarea></td>

</tr>

</table>

<br />

</fieldset> </TD>

</TR>

</TABLE>

</td>

</tr>

<TR>

<TD colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</TR>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\addjob.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="jobForm" method="post" action="modifyjob.do?action=addjob" onSubmit="return jobValidate();" >

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >应聘信息录入</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>应聘信息</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">姓&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;名：</td>

<td width="31%"><input name="name" type="text" class="input">

<span class="red">\*</span></td>

<td width="14%"><div align="right">性&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;别：</div></td>

<td width="44%">

<input name="sex" type="radio" value="1" checked>男

<input name="sex" type="radio" value="0">女

</tr>

<tr>

<td width="11%"><div align="right">体&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;检：</div></td>

<td width="31%">

<input name="health" type="radio" value="1" checked>通过

<input name="health" type="radio" value="0">未通过

<td nowrap align="right" width="14%">年&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;龄：</td>

<td><input name="age" type="text" class="input"></td>

</tr>

<tr>

<td><div align="right">职&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;位：</div></td>

<td><input name="job" type="text" class="input">

<span class="red">\*</span></td>

<td><div align="right">工作经验：</div></td>

<td><input name="experience" type="text" class="input">

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">所学专业：</td>

<td><input name="specialty" type="text" class="input">

<span class="red">\*</span></td>

<td><div align="right">毕业学校：</div></td>

<td><input name="school" type="text" class="input">

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">学&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;历：</td>

<td><input name="studyeffort" type="text" class="input">

<span class="red">\*</span></td>

<td nowrap align="right">电&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;话：</td>

<td><input name="tel" type="text" class="input">

<span class="red">\*</span></td>

</tr>

<tr>

<td><div align="right">Email：</div></td>

<td><input name="email" type="text" class="input">

<span class="red">\*</span></td>

</tr>

<tr>

<td width="11%" nowrap align="right">详细经历：</td>

<td colspan="3"><textarea name="content" cols="100" rows="6" class="input" id="content">

</textarea></td>

</tr>

</table>

</fieldset>

</TD>

</TR>

</TABLE>

</td>

</tr>

<TR>

<TD colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</TR>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\addperformance

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="performanceForm" method="post" action="modifyperformance.do?action=addperformance" onSubmit="return performanceValidate()">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >绩效信息登记</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="reset" value="返回" class="button"/>

</td></tr>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>绩效信息</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">绩效分数：</td>

<td width="89%"><input name="score" type="text" class="input"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">增加工资：</td>

<td><input name="addmoney" type="text" class="input"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">创建时间：</td>

<td><input name="createtime" type="text" class="input"></td>

</tr>

<tr>

<td width="11%" nowrap align="right">绩效说明：</td>

<td><textarea name="content" cols="100" rows="6" class="input" id="content"></textarea></td>

</tr>

</table>

<br />

</fieldset> </TD>

</tr>

</table>

</td>

</tr>

<tr>

<td colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</tr>

</table>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\addstipend.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="stipendForm" method="post" action="modifystipend.do?action=addstipend" onSubmit="return stipendValidate();" >

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >薪金数据录入</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<tr>

<td width="100%">

<fieldset style="height:100%;">

<legend>薪金数据</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" >员工姓名：</td>

<td width="36%"><input name="name" type="text" class="input" /><span class="red">\*</span></td>

<td nowrap align="right" >基本薪金：</td>

<td width="36%"><input name="basic" type="text" id ="basic" class="input" /><span class="red">\*</span></td>

</tr>

<tr>

<td nowrap height="36" align="right">全&nbsp;勤&nbsp;奖：</td>

<td><input name="duty" type="text" class="input" id="duty"></td>

<td nowrap align="right">提&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;成：</td>

<td><input name="other" type="text" class="input" id="other"></td>

</tr>

<tr>

<td nowrap height="36"><div align="right">罚&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;款：</div></td>

<td><input name="punishment" type="text" class="input" id="punishment"></td>

<td nowrap align="right">发放时间：</td>

<td><input name="granttime" type="text" class="input" id="granttime">

<span class="red">\*</span></td>

</tr>

<tr>

<td colspan="4"><div align="center">注：金钱单位（元/RMB）</div></td>

</tr>

</table>

<br />

</fieldset> </TD>

</tr>

</table>

</td>

</tr>

<tr>

<td colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></td>

</tr>

</table>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\adduser.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="userForm" method="post" action="modifyuser.do?action=adduser" onSubmit="return userValidate();">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >人员信息录入</th>

</tr>

<tr>

<td class="CPanel">

<table width="90%" border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>人员信息</legend>

<table width="100%" border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="9%">用户名：</td>

<td width="36%">

<input name="username" type="text" class="input" id="username"/>

<span class="red">\*</span></td>

<td width="12%"><div align="right">登录密码：</div></td>

<td width="43%">

<input name="password" type="text" class="input" id="password" />

<span class="red">\*</span></tr>

<tr>

<td nowrap align="right" width="9%">性别：</td>

<td>

<input name="sex" type="radio" value="1" checked> 男

<input name="sex" type="radio" value="0">女</td>

<td><div align="right">出生日期：</div></td>

<td>

<input name="birthday" type="text" class="input" id="birthday">

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">是否管理员：</td>

<td>

<input name="isadminhelp" type="checkbox" onClick="javascript:adminChecked();" >

<input type="hidden" name="isadmin" ></td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td nowrap align="right">人员简介：</td>

<td colspan="3">

<textarea name="content" cols="100" rows="6" class="input" id="content"></textarea></td>

</tr>

</table>

<br />

</fieldset>

</TD>

</TR>

</TABLE>

</td>

</tr>

<TR>

<TD colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</TR>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\detaileducation

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Educate"%>

<%@ page import="com.zhangying.tool.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="educateForm" method="post" action="modifyeducate.do?action=addeducate" onsubmit="return educateValidate();">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >

<%

String educate=null;

try{

educate=request.getParameter("educate").toString();

}catch(Exception e){

educate="0";

}

if("1".equals(educate)){

out.print("培训详细总结");

}else{

out.print("培训详细计划");

}

%>

</th>

</tr>

<%

Educate e=(Educate)request.getAttribute("educate");

if(e!=null){

%>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<table border="0" cellpadding="8" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">培训名称：</td>

<td colspan="3"><%=e.getName()%>

</td>

</tr>

<tr>

<td nowrap align="right" width="11%">培训目的：</td>

<td colspan="3"><%=e.getPurpose()%>

</td>

</tr>

<tr>

<td nowrap align="right">培训开始时间：</td>

<td width="29%"><%=StringUtil.notNull(DateUtil.parseToString(e.getBegintime(),DateUtil.yyyyMMdd))%>

</td>

<td width="18%">培训结束时间：</td>

<td width="42%"><%=StringUtil.notNull(DateUtil.parseToString(e.getEndtime(),DateUtil.yyyyMMdd))%>

</td>

</tr>

<tr>

<td nowrap align="right">讲师：</td>

<td><%=e.getTeacher()%>

</td>

<td>培训人员：</td>

<td><%=e.getStudent()%>

</td>

</tr>

<tr>

<td width="11%" nowrap align="right">培训材料：</td>

<td colspan="3"><%=e.getDatum()%></td>

</tr>

<%if("1".equals(educate)){%>

<tr>

<td width="11%" height="22" align="center" >培训效果：</td>

<td height="22" colspan="3" align="left" ><%=StringUtil.notNull(e.getEffect())%>&nbsp;</td>

</tr>

<tr>

<td width="11%" height="22" align="center" >培训总结：</td>

<td height="22" colspan="3" align="left" ><%=StringUtil.notNull(e.getSummarize())%>&nbsp;</td>

</tr>

<tr>

<td colspan="4" align="center">

<a href="updateeducate.do?action=detaileducate&educate=1&id=<%=e.getId()%>">修改</a>&nbsp;&nbsp;

<a href="#" onclick="javascript:history.back();">返回</a></td>

</tr>

<%}%>

</table>

</fieldset>

</TD>

</TR>

<%}else{ %>

<tr>

<td height="22" colspan="4" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\detailinstitution.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Institution"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >奖惩信息详情</th>

</tr>

<%

Institution e=(Institution)request.getAttribute("institution");

if(e!=null){

%>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr>

<td width="100%">

<fieldset style="height:100%;">

<legend>奖惩信息详情</legend>

<table border="0" cellpadding="8" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">奖惩名称：</td>

<td width="89%"><%=e.getName()%></td>

</tr>

<tr>

<td nowrap align="right" width="11%">奖惩原因：</td>

<td><%=e.getReason()%></td>

</tr>

<tr>

<td width="11%" nowrap align="right">奖惩说明：</td>

<td><%=e.getExplain()%></td>

</tr>

<tr>

<td height="22" colspan="2" align="center" >

<a href="updateinstitution.do?action=detailinstitution&id=<%=e.getId()%>">修改</a>&nbsp;&nbsp;

<a href="#" onClick="javascript:history.back();">返回</a>

</td>

</tr>

</table>

</fieldset>

</td>

</tr>

</table>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="2" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

</table>

</div>

</body>

</html>

\HRmanagement\WebRoot\detailjob.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Job"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >人才详细信息</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<%

Job j=(Job)request.getAttribute("job");

if(j!=null){

%>

<tr>

<td width="100%">

<fieldset style="height:100%;">

<table border="0" cellpadding="8" cellspacing="1" style="width:100%">

<tr>

<td align="right" width="9%">姓&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;名：</td>

<td width="36%"><%=j.getName()%></td>

<td width="20%"><div align="right">性&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;别：</div></td>

<td width="43%"><%=new Byte("1").equals(j.getSex())?"男":"女"%> </tr>

<tr>

<td width="20%"><div align="right">体&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;检：</div></td>

<td width="43%"><%=new Byte("1").equals(j.getHealth())?"通过":"未通过"%>

<td align="right" width="9%">年&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;龄：</td>

<td><%=j.getAge()%></td>

</td>

</tr>

<tr>

<td><div align="right">职&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;位：</div></td>

<td><%=j.getJob()%>

<td><div align="right">工作经验：</div></td>

<td><%=j.getExperience()%></td>

</td>

</tr>

<tr>

<td align="right">所学专业：</td>

<td><%=j.getSpecialty()%>

<td><div align="right">毕业学校：</div></td>

<td><%=j.getSchool()%></td>

</tr>

<tr>

<td align="right">学&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;历：</td>

<td><%=j.getStudyeffort()%>

</td>

<td align="right">电&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;话：</td>

<td><%=j.getTel()%>

</td>

</tr>

<tr>

<td><div align="right">Email：</div></td>

<td><%=j.getEmail()%></td>

<td align="right">详细经历：</td>

<td colspan="4"><%=j.getContent()%></td>

</tr>

<tr>

<td colspan="4" align="center">

<a href="updatejob.do?action=detailjob&id=<%=j.getId() %>">修改</a>&nbsp;&nbsp;

<a href="#" onclick="javascript:history.back();">返回</a></td>

</tr>

</table>

</fieldset>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="4" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

</table>

</td>

</tr>

</tabel>

</div>

</body>

</html>

\HRmanagement\WebRoot\detailperformance

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Performance"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >绩效信息详情</th>

</tr>

<%

Performance e=(Performance)request.getAttribute("performance");

if(e!=null){

%>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr>

<td width="100%">

<fieldset style="height:100%;">

<legend>绩效信息详情</legend>

<table border="0" cellpadding="8" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">绩效分数：</td>

<td width="89%"><%=e.getScore()%></td>

</tr>

<tr>

<td nowrap align="right" width="11%">增加工资：</td>

<td><%=e.getAddmoney()%></td>

</tr>

<tr>

<td nowrap align="right" width="11%">创建时间：</td>

<td width="89%"><%=e.getCreatetime()%></td>

</tr>

<tr>

<td width="11%" nowrap align="right">绩效说明：</td>

<td><%=e.getContent()%></td>

</tr>

<tr>

<td height="22" colspan="2" align="center" >

<a href="updateperformance.do?action=detailperformance&id=<%=e.getId()%>">修改</a>&nbsp;&nbsp;

<a href="#" onClick="javascript:history.back();">返回</a>

</td>

</tr>

</table>

</fieldset>

</td>

</tr>

</table>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="2" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

</table>

</div>

</body>

</html>

\HRmanagement\WebRoot\error.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>人事管理信息系统</title>

<meta http-equiv="Content-Type" content="text/html; charset=GBK">

<link href="css/css.css" rel="stylesheet" type="text/css">

</head>

<body>

<table width="100%" height="100%" border="1">

<tr>

<td align="center" >没有action！！！</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\index.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%

response.sendRedirect("index.do");

%>

\HRmanagement\WebRoot\left.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

background-image: url(images/left.gif);

}

-->

</style>

<link href="css/css.css" rel="stylesheet" type="text/css" />

</head>

<SCRIPT language=JavaScript>

function tupian(idt){

var nametu="xiaotu"+idt;

var tp = document.getElementById(nametu);

tp.src="images/ico05.gif";//图片ico05为蓝色的正方形

for(var i=1;i<30;i++)

{

var nametu2="xiaotu"+i;

if(i!=idt\*1)

{

var tp2=document.getElementById('xiaotu'+i);

if(tp2!=undefined)

{tp2.src="images/ico06.gif";}//图片ico06白色的正方形

}

}

}

function list(idstr){

var name1="subtree"+idstr;

var name2="img"+idstr;

var objectobj=document.all(name1);

var imgobj=document.all(name2);

//alert(imgobj);

if(objectobj.style.display=="none"){

for(i=1;i<101;i++){

var name3="img"+i;

var name="subtree"+i;

var o=document.all(name);

if(o!=undefined){

o.style.display="none";

var image=document.all(name3);

//alert(image);

image.src="images/ico04.gif";

}

}

objectobj.style.display="";

imgobj.src="images/ico03.gif";

}

else{

objectobj.style.display="none";

imgobj.src="images/ico04.gif";

}

}

</SCRIPT>

<body>

<table width="198" border="0" cellpadding="0" cellspacing="0" class="left-table01">

<tr>

<TD>

<table width="100%" border="0" cellpadding="0" cellspacing="0">

<tr>

<td width="207" height="55" background="images/nav01.gif">

<table width="90%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="25%" rowspan="2"><img src="images/ico02.gif" width="35" height="35" /></td>

<td width="75%" height="22" class="left-font01">您好，<span class="left-font02">${sessionScope.users.username}</span></td>

</tr>

<tr>

<td height="22" class="left-font01">

[&nbsp;<a href="welcome.jsp" target="\_top" class="left-font01">退出</a>&nbsp;]</td>

</tr>

</table>

</td>

</tr>

</table>

<!-- 任务系统开始 -->

<TABLE width="100%" border="0" cellpadding="0" cellspacing="0" class="left-table03">

<tr>

<td height="29">

<table width="85%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="8%"><img name="img8" id="img8" src="../images/ico04.gif" width="8" height="11" /></td>

<td width="92%">

<a href="javascript:" target="mainFrame" class="left-font03" onClick="list('8');" >人员管理</a></td>

</tr>

</table>

</td>

</tr>

</TABLE>

<table id="subtree8" style="DISPLAY: none" width="80%" border="0" align="center" cellpadding="0"

cellspacing="0" class="left-table02">

<tr>

<td width="9%" height="20" ><img id="xiaotu1" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="adduser.do" target="mainFrame" class="left-font03" onClick="tupian('1');">人员信息录入</a></td>

</tr>

<tr>

<td width="9%" height="21" ><img id="xiaotu2" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="user.do?action=listuser" target="mainFrame" class="left-font03" onClick="tupian('2');">人员信息查看</a></td>

</tr>

</table>

<!-- 任务系统结束 -->

<!-- 消息系统开始 -->

<TABLE width="100%" border="0" cellpadding="0" cellspacing="0" class="left-table03">

<tr>

<td height="29">

<table width="85%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="8%"><img name="img7" id="img7" src="../images/ico04.gif" width="8" height="11" /></td>

<td width="92%">

<a href="javascript:" target="mainFrame" class="left-font03" onClick="list('7');" >招聘管理</a></td>

</tr>

</table>

</td>

</tr>

</TABLE>

<table id="subtree7" style="DISPLAY: none" width="80%" border="0" align="center" cellpadding="0"

cellspacing="0" class="left-table02">

<tr>

<td width="9%" height="20" ><img id="xiaotu3" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%">

<a href="addjob.do" target="mainFrame" class="left-font03" onClick="tupian('3');">应聘信息录入</a></td>

</tr>

<tr>

<td width="9%" height="20" ><img id="xiaotu4" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%">

<a href="job.do?action=listjob" target="mainFrame" class="left-font03" onClick="tupian('4');">应聘信息查看</a></td>

</tr>

<tr>

<td width="9%" height="20" ><img id="xiaotu5" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%">

<a href="job.do?action=listjob&isstock=yes" target="mainFrame" class="left-font03" onClick="tupian('5');">人才库浏览

</a></td>

</tr>

</table>

<!-- 消息系统结束 -->

<!-- 人员系统开始 -->

<table width="100%" border="0" cellpadding="0" cellspacing="0" class="left-table03">

<tr>

<td height="29"><table width="85%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="8%" height="12"><img name="img100" id="img100" src="../images/ico04.gif" width="8" height="11" /></td>

<td width="92%"><a href="javascript:" target="mainFrame" class="left-font03" onClick="list('100');" >人事异动</a></td>

</tr>

</table></td>

</tr>

</table>

<table id="subtree100" style="DISPLAY: none" width="80%" border="0" align="center" cellpadding="0" cellspacing="0" class="left-table02">

<tr>

<td width="9%" height="20" ><img id="xiaotu15" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="job.do?action=listjob&isstock=yes" target="mainFrame" class="left-font03" onClick="tupian('15');">人事信息查看</a></td>

</tr>

</table>

<!-- 人员系统结束 -->

<TABLE width="100%" border="0" cellpadding="0" cellspacing="0" class="left-table03">

<tr>

<td height="29">

<table width="85%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="8%"><img name="img1" id="img1" src="../images/ico04.gif" width="8" height="11" /></td>

<td width="92%">

<a href="javascript:" target="mainFrame" class="left-font03" onClick="list('1');" >培训管理</a></td>

</tr>

</table>

</td>

</tr>

</TABLE>

<table id="subtree1" style="DISPLAY: none" width="80%" border="0" align="center" cellpadding="0"

cellspacing="0" class="left-table02">

<tr>

<td width="9%" height="20" ><img id="xiaotu6" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="addeducate.do" target="mainFrame" class="left-font03" onClick="tupian('6');">培训计划录入</a></td>

</tr>

<tr>

<td width="9%" height="20" ><img id="xiaotu7" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="educate.do?action=listeducate" target="mainFrame" class="left-font03" onClick="tupian('7');">培训计划查看</a></td>

</tr>

<tr>

<td width="9%" height="20" ><img id="xiaotu8" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="educate.do?action=listeducate&educate=1" target="mainFrame" class="left-font03" onClick="tupian('8');">培训总结查看</a></td>

</tr>

</table>

<!-- 项目系统结束 -->

<!-- 客户系统开始 -->

<table width="100%" border="0" cellpadding="0" cellspacing="0" class="left-table03">

<tr>

<td height="29"><table width="85%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="8%" height="12"><img name="img2" id="img2" src="../images/ico04.gif" width="8" height="11" /></td>

<td width="92%"><a href="javascript:" target="mainFrame" class="left-font03" onClick="list('2');" >奖惩管理</a></td>

</tr>

</table></td>

</tr>

</table>

<table id="subtree2" style="DISPLAY: none" width="80%" border="0" align="center" cellpadding="0" cellspacing="0" class="left-table02">

<tr>

<td width="9%" height="20" ><img id="xiaotu9" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="addinstitution.do" target="mainFrame" class="left-font03" onClick="tupian('9');">奖惩信息登记</a></td>

</tr>

<tr>

<td width="9%" height="20" ><img id="xiaotu10" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="institution.do?action=listinstitution" target="mainFrame" class="left-font03" onClick="tupian('10');">奖惩信息查看</a></td>

</tr>

</table>

<!-- 客户系统结束 -->

<!-- 人员系统开始 -->

<table width="100%" border="0" cellpadding="0" cellspacing="0" class="left-table03">

<tr>

<td height="29"><table width="85%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="8%" height="12"><img name="img3" id="img3" src="../images/ico04.gif" width="8" height="11" /></td>

<td width="92%"><a href="javascript:" target="mainFrame" class="left-font03" onClick="list('3');" >绩效管理</a></td>

</tr>

</table></td>

</tr>

</table>

<table id="subtree3" style="DISPLAY: none" width="80%" border="0" align="center" cellpadding="0" cellspacing="0" class="left-table02">

<tr>

<td width="9%" height="20" ><img id="xiaotu11" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="addperformance.do" target="mainFrame" class="left-font03" onClick="tupian('11');">绩效信息登记</a></td>

</tr>

<tr>

<td width="9%" height="20" ><img id="xiaotu12" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="performance.do?action=listperformance" target="mainFrame" class="left-font03" onClick="tupian('12');">绩效信息查看</a></td>

</tr>

</table>

<!-- 人员系统结束 -->

<!-- 人员系统开始 -->

<table width="100%" border="0" cellpadding="0" cellspacing="0" class="left-table03">

<tr>

<td height="29"><table width="85%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="8%" height="12"><img name="img99" id="img99" src="../images/ico04.gif" width="8" height="11" /></td>

<td width="92%"><a href="javascript:" target="mainFrame" class="left-font03" onClick="list('99');" >薪金管理</a></td>

</tr>

</table></td>

</tr>

</table>

<table id="subtree99" style="DISPLAY: none" width="80%" border="0" align="center" cellpadding="0" cellspacing="0" class="left-table02">

<tr>

<td width="9%" height="20" ><img id="xiaotu13" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="addstipend.do" target="mainFrame" class="left-font03" onClick="tupian('13');">薪金数据录入</a></td>

</tr>

<tr>

<td width="9%" height="20" ><img id="xiaotu14" src="../images/ico06.gif" width="8" height="12" /></td>

<td width="91%"><a href="stipend.do?action=liststipend" target="mainFrame" class="left-font03" onClick="tupian('14');">薪金数据查看</a></td>

</tr>

</table>

<!-- 人员系统结束 -->

</TD>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\listeducate.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Educate"%>

<%@ page import="com.zhangying.tool.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

}

.tabfont01 {

font-family: "宋体";

font-size: 9px;

color: #555555;

text-decoration: none;

text-align: center;

}

.font051 {font-family: "宋体";

font-size: 12px;

color: #333333;

text-decoration: none;

line-height: 20px;

}

.font201 {font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: none;

}

.button {

font-family: "宋体";

font-size: 14px;

height: 37px;

}

html { overflow-x: auto; overflow-y: auto; border:0;}

-->

</style>

<link href="css/css.css" rel="stylesheet" type="text/css" />

<script type="text/JavaScript">

</script>

<link href="css/style.css" rel="stylesheet" type="text/css" />

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="30"> <table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="62" background="images/nav04.gif">&nbsp;</td>

</tr>

</table></td></tr>

<tr>

<td><table id="subtree1" style="DISPLAY: " width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td><table width="95%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td height="20"><span class="newfont07"> <%

String educate=null;

try{

educate=request.getParameter("educate").toString();

}catch(Exception e){

educate="0";

}

if("1".equals(educate)){

out.print("培训总结查看");

}else{

out.print("培训计划查看");

}

%></span></td>

</tr>

<tr>

<td height="40" class="font42">

<table width="100%" border="0" cellpadding="4" cellspacing="1" bgcolor="#464646" class="newfont03">

<tr class="CTitle" >

<td height="22" colspan="6" align="center" style="font-size:16px">

<%

if("1".equals(educate)){

out.print("培训总结");

}else{

out.print("培训计划");

}

%>列表</td>

</tr>

<tr bgcolor="#EEEEEE">

<td height="22" align="center" >培训名称</td>

<td height="22" align="center" >讲师</td>

<td height="22" align="center" >培训人员</td>

<td height="22" align="center" >开始时间</td>

<td height="22" align="center" >结束时间</td>

<td height="22" align="center" >执行操作</td>

</tr>

<% List list=(List)request.getAttribute("list");

if(list!=null&&list.size()>0){

Iterator it = list.iterator();

while (it.hasNext()) {

Educate j = (Educate) it.next();

%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=j.getName()%></td>

<td height="22" align="center" ><%=j.getTeacher()%></td>

<td height="22" align="center" ><%=j.getStudent()%></td>

<td height="22" align="center" ><%=StringUtil.notNull(DateUtil.parseToString(j.getBegintime(),DateUtil.yyyyMMdd))%></td>

<td height="22" align="center" ><%=StringUtil.notNull(DateUtil.parseToString(j.getEndtime(),DateUtil.yyyyMMdd))%></td>

<td height="22" align="center" ><a href="detaileducate.do?educate=<%=educate%>&action=detaileducate&id=<%=j.getId()%>">详细</a>&nbsp;&nbsp;<a href="modifyeducate.do?action=deleteeducate&id=<%=j.getId()%>">删除</a><%if(!"1".equals(educate)){%>&nbsp;&nbsp;<a href="updateeducate.do?educate=1&action=detaileducate&id=<%=j.getId()%>">培训完成</a><%}%></td>

</tr>

<% }

}else{

%>

<tr bgcolor="#FFFFFF">

<td height="22" colspan="6" align="center" >对不起，没有信息！！！</td>

</tr>

<%}%>

</table></td>

</tr>

</table>

</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRootlistinstitution.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Institution"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

}

.tabfont01 {

font-family: "宋体";

font-size: 9px;

color: #555555;

text-decoration: none;

text-align: center;

}

.font051 {font-family: "宋体";

font-size: 12px;

color: #333333;

text-decoration: none;

line-height: 20px;

}

.font201 {font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: none;

}

.button {

font-family: "宋体";

font-size: 14px;

height: 37px;

}

html { overflow-x: auto; overflow-y: auto; border:0;}

-->

</style>

<link href="css/css.css" rel="stylesheet" type="text/css" />

<script type="text/JavaScript">

</script>

<link href="css/style.css" rel="stylesheet" type="text/css" />

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="30"> <table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="62" background="images/nav04.gif">&nbsp;</td>

</tr>

</table></td></tr>

<tr>

<td><table id="subtree1" style="DISPLAY: " width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td><table width="95%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td height="20"><span class="newfont07">奖惩信息查看</span></td>

</tr>

<tr>

<td height="40" class="font42">

<table width="100%" border="0" cellpadding="4" cellspacing="1" bgcolor="#464646" class="newfont03">

<tr class="CTitle" >

<td height="22" colspan="3" align="center" style="font-size:16px">奖惩信息列表</td>

</tr>

<tr bgcolor="#EEEEEE">

<td height="22" align="center" >奖惩名称</td>

<td height="22" align="center" >奖惩原因</td>

<td height="22" align="center" >执行操作</td>

</tr>

<% List list=(List)request.getAttribute("list");

if(list!=null&&list.size()>0){

Iterator it = list.iterator();

while (it.hasNext()) {

Institution j = (Institution) it.next();

%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=j.getName()%></td>

<td height="22" align="center" ><%=j.getReason()%></td>

<td height="22" align="center" ><a href="detailinstitution.do?action=detailinstitution&id=<%=j.getId()%>">详细</a>&nbsp;&nbsp;<a href="modifyinstitution.do?action=deleteinstitution&id=<%=j.getId()%>">删除</a></td>

</tr>

<% }

}else{

%>

<tr bgcolor="#FFFFFF">

<td height="22" colspan="3" align="center" >对不起，没有添加奖惩信息！！！</td>

</tr>

<%}%>

</table></td>

</tr>

</table>

</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\listjob.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Job"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

}

.tabfont01 {

font-family: "宋体";

font-size: 9px;

color: #555555;

text-decoration: none;

text-align: center;

}

.font051 {font-family: "宋体";

font-size: 12px;

color: #333333;

text-decoration: none;

line-height: 20px;

}

.font201 {font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: none;

}

.button {

font-family: "宋体";

font-size: 14px;

height: 37px;

}

html { overflow-x: auto; overflow-y: auto; border:0;}

-->

</style>

<link href="css/css.css" rel="stylesheet" type="text/css" />

<script type="text/JavaScript">

</script>

<link href="css/style.css" rel="stylesheet" type="text/css" />

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="30"> <table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="62" background="images/nav04.gif">&nbsp;</td>

</tr>

</table></td></tr>

<tr>

<td><table id="subtree1" style="DISPLAY: " width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td><table width="95%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td height="20"><span class="newfont07">人才信息查看</span></td>

</tr>

<tr>

<td height="40" class="font42">

<table width="100%" border="0" cellpadding="4" cellspacing="1" bgcolor="#464646" class="newfont03">

<tr class="CTitle" >

<td height="22" colspan="8" align="center" style="font-size:16px">人才信息列表</td>

</tr>

<tr bgcolor="#EEEEEE">

<td width="4%" align="center" height="30">姓名</td>

<td width="10%" align="center" >性别</td>

<td width="10%" align="center" >体检</td>

<td width="10%" align="center" >年龄</td>

<td width="10%" align="center" >职位</td>

<td width="10%" align="center" >所学专业</td>

<td width="15%" align="center" >工作经验</td>

<td width="12%" align="center" >执行操作</td>

</tr>

<% List list=(List)request.getAttribute("list");

if(list!=null&&list.size()>0){

Iterator it = list.iterator();

while (it.hasNext()) {

Job j = (Job) it.next();

%>

<%if(j.getIsstock()!=null&&j.getIsstock().intValue()==0){%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=j.getName()%></td>

<td height="22" align="center" ><%=new Byte("1").equals(j.getSex())?"男":"女"%></td>

<td height="22" align="center" ><%=new Byte("1").equals(j.getHealth())?"通过":"未通过"%></td>

<td height="22" align="center" ><%=j.getAge()%></td>

<td height="22" align="center" ><%=j.getJob()%></td>

<td height="22" align="center" ><%=j.getSpecialty()%></td>

<td height="22" align="center" ><%=j.getExperience()%></td>

<td height="22" align="center" ><a href="detailjob.do?action=detailjob&id=<%=j.getId()%>">详细</a>&nbsp;&nbsp;<a href="modifyjob.do?action=deletejob&&id=<%=j.getId()%>">删除</a>

<a href="modifyjob.do?action=updatejob&id=<%=j.getId()%>&isstock=1">入库</a></td>

<% }else{%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=j.getName()%></td>

<td height="22" align="center" ><%=new Byte("1").equals(j.getSex())?"男":"女"%></td>

<td height="22" align="center" ><%=new Byte("1").equals(j.getHealth())?"通过":"未通过"%></td>

<td height="22" align="center" ><%=j.getAge()%></td>

<td height="22" align="center" ><%=j.getJob()%></td>

<td height="22" align="center" ><%=j.getSpecialty()%></td>

<td height="22" align="center" ><%=j.getExperience()%></td>

<td height="22" align="center" ><a href="detailjob.do?action=detailjob&id=<%=j.getId()%>">详细</a>&nbsp;&nbsp;<a href="modifyjob.do?action=deletejob&isstock=1&id=<%=j.getId()%>">删除</a>

<%}%>

</tr>

<%}

}else{

%>

<tr bgcolor="#FFFFFF">

<td height="22" colspan="8" align="center" >对不起，没有信息！！！</td>

</tr>

<%}%>

</table>

</td>

</tr>

</table>

</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\listperformance.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Performance"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

}

.tabfont01 {

font-family: "宋体";

font-size: 9px;

color: #555555;

text-decoration: none;

text-align: center;

}

.font051 {font-family: "宋体";

font-size: 12px;

color: #333333;

text-decoration: none;

line-height: 20px;

}

.font201 {font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: none;

}

.button {

font-family: "宋体";

font-size: 14px;

height: 37px;

}

html { overflow-x: auto; overflow-y: auto; border:0;}

-->

</style>

<link href="css/css.css" rel="stylesheet" type="text/css" />

<script type="text/JavaScript">

</script>

<link href="css/style.css" rel="stylesheet" type="text/css" />

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="30"> <table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="62" background="images/nav04.gif">&nbsp;</td>

</tr>

</table></td></tr>

<tr>

<td><table id="subtree1" style="DISPLAY: " width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td><table width="95%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td height="20"><span class="newfont07">绩效信息查看</span></td>

</tr>

<tr>

<td height="40" class="font42">

<table width="100%" border="0" cellpadding="4" cellspacing="1" bgcolor="#464646" class="newfont03">

<tr class="CTitle" >

<td height="22" colspan="8" align="center" style="font-size:16px">绩效信息列表</td>

</tr>

<tr bgcolor="#EEEEEE">

<td height="22" width="5%" align="center" >绩效分数</td>

<td height="22" width="5%" align="center" >增加工资</td>

<td height="22" width="5%"align="center" >创建时间</td>

<td height="22" width="50%"align="center" >绩效说明</td>

<td height="22" width="15%"align="center" >执行操作</td>

</tr>

<% List list=(List)request.getAttribute("list");

if(list!=null&&list.size()>0){

Iterator it = list.iterator();

while (it.hasNext()) {

Performance j = (Performance) it.next();

%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=j.getScore()%></td>

<td height="22" align="center" ><%=j.getAddmoney()%></td>

<td height="22" align="center" ><%=j.getCreatetime()%></td>

<td height="22" align="center" ><%=j.getContent()%></td>

<td height="22" align="center" ><a href="detailperformance.do?action=detailperformance&id=<%=j.getId()%>">详细</a>&nbsp;&nbsp;<a href="modifyperformance.do?action=deleteperformance&id=<%=j.getId()%>">删除</a></td>

</tr>

<% }

}else{

%>

<tr bgcolor="#FFFFFF">

<td height="22" colspan="3" align="center" >对不起，没有添加绩效信息！！！</td>

</tr>

<%}%>

</table></td>

</tr>

</table>

</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\liststipend.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Stipend"%>

<%@ page import="com.zhangying.tool.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

}

.tabfont01 {

font-family: "宋体";

font-size: 9px;

color: #555555;

text-decoration: none;

text-align: center;

}

.font051 {font-family: "宋体";

font-size: 12px;

color: #333333;

text-decoration: none;

line-height: 20px;

}

.font201 {font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: none;

}

.button {

font-family: "宋体";

font-size: 14px;

height: 37px;

}

html { overflow-x: auto; overflow-y: auto; border:0;}

-->

</style>

<link href="css/css.css" rel="stylesheet" type="text/css" />

<script type="text/JavaScript">

</script>

<link href="css/style.css" rel="stylesheet" type="text/css" />

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="30"> <table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="62" background="images/nav04.gif">&nbsp;</td>

</tr>

</table></td></tr>

<tr>

<td><table id="subtree1" style="DISPLAY: " width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td><table width="95%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td height="20"><span class="newfont07">薪金数据查看</span></td>

</tr>

<tr>

<td height="40" class="font42">

<table width="100%" border="0" cellpadding="4" cellspacing="1" bgcolor="#464646" class="newfont03">

<tr class="CTitle" >

<td height="22" colspan="8" align="center" style="font-size:16px">薪金数据列表</td>

</tr>

<tr bgcolor="#EEEEEE">

<td height="22" align="center" >员工姓名</td>

<td height="22" align="center" >基本薪金</td>

<td height="22" align="center" >全勤奖</td>

<td height="22" align="center" >提成</td>

<td height="22" align="center" >罚款</td>

<td height="22" align="center" >发放时间</td>

<td height="22" align="center" >总计</td>

<td height="22" align="center" >执行操作</td>

</tr>

<% List list=(List)request.getAttribute("list");

if(list!=null&&list.size()>0){

Iterator it = list.iterator();

while (it.hasNext()) {

Stipend j = (Stipend) it.next();

%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=j.getName()%></td>

<td height="22" align="center" ><%=j.getBasic()%></td>

<td height="22" align="center" ><%=j.getDuty()%></td>

<td height="22" align="center" ><%=j.getOther()%></td>

<td height="22" align="center" ><%=j.getPunishment()%></td>

<td height="22" align="center" ><%=StringUtil.notNull(DateUtil.parseToString(j.getGranttime(),DateUtil.yyyyMMdd))%>&nbsp;</td>

<td height="22" align="center" ><%=j.getTotalize()%></td>

<td height="22" align="center" ><a href="updatestipend.do?action=detailstipend&id=<%=j.getId()%>">修改</a>&nbsp;&nbsp;<a href="modifystipend.do?action=deletestipend&id=<%=j.getId()%>">删除</a></td>

</tr>

<% }

}else{

%>

<tr bgcolor="#FFFFFF">

<td height="22" colspan="8" align="center" >对不起，没有添加薪金信息！！！</td>

</tr>

<%}%>

</table></td>

</tr>

</table>

</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\listuser.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Users"%>

<%@ page import="com.zhangying.tool.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

}

.tabfont01 {

font-family: "宋体";

font-size: 9px;

color: #555555;

text-decoration: none;

text-align: center;

}

.font051 {font-family: "宋体";

font-size: 12px;

color: #333333;

text-decoration: none;

line-height: 20px;

}

.font201 {font-family: "宋体";

font-size: 12px;

color: #FF0000;

text-decoration: none;

}

.button {

font-family: "宋体";

font-size: 14px;

height: 37px;

}

html { overflow-x: auto; overflow-y: auto; border:0;}

-->

</style>

<link href="css/css.css" rel="stylesheet" type="text/css" />

<script type="text/JavaScript">

</script>

<link href="css/style.css" rel="stylesheet" type="text/css" />

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="30"> <table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="62" background="images/nav04.gif">&nbsp;</td>

</tr>

</table></td></tr>

<tr>

<td><table id="subtree1" style="DISPLAY: " width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td>

<table width="95%" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td height="20"><span class="newfont07">人员信息查看</span></td>

</tr>

<tr>

<td height="40" class="font42">

<table width="100%" border="0" cellpadding="4" cellspacing="1" bgcolor="#464646" class="newfont03">

<tr class="CTitle" >

<td height="22" colspan="7" align="center" style="font-size:16px">人员信息列表</td>

</tr>

<tr bgcolor="#EEEEEE">

<td width="4%" align="center" height="30">用户名</td>

<td width="10%" align="center" >登录密码</td>

<td width="10%" align="center" >性别</td>

<td width="10%" align="center" >出生日期</td>

<td width="10%" align="center" >是否管理员</td>

<td width="15%" align="center" >简介</td>

<td width="12%" align="center" >执行操作</td>

</tr>

<% List list=(List)request.getAttribute("list");

if(list!=null&&list.size()>0){

Iterator it = list.iterator();

while (it.hasNext()) {

Users u = (Users) it.next();

%>

<tr bgcolor="#FFFFFF">

<td height="22" align="center" ><%=StringUtil.notNull(u.getUsername())%>&nbsp;</td>

<td height="22" align="center" ><%=StringUtil.notNull(u.getPassword())%>&nbsp;</td>

<td height="22" align="center" ><%=new Byte("1").equals(u.getSex())?"男":"女"%></td>

<td height="22" align="center" ><%=StringUtil.notNull(DateUtil.parseToString(u.getBirthday(),DateUtil.yyyyMMdd))%>&nbsp;</td>

<td height="22" align="center" ><%=new Byte("1").equals(u.getIsadmin())?"是":"否"%></td>

<td height="22" align="center" ><%=StringUtil.notNull(u.getContent())%>&nbsp;</td>

<td height="22" align="center" ><a href="selectuser.do?action=selectuser&id=<%=u.getId()%>">修改</a>&nbsp;&nbsp;<a href="modifyuser.do?action=deleteuser&id=<%=u.getId()%>">删除</a></td>

</tr>

<%}

}else{

%>

<tr>

<td height="22" colspan="7" align="center" >对不起，暂时没有人员信息！！！</td>

</tr>

<%}%>

</table>

</td>

</tr>

</table>

</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\main.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>麻麻鱼府人事管理信息系统</title>

<meta http-equiv="Content-Type" content="text/html; charset=GBK">

<link href="css/css.css" rel="stylesheet" type="text/css">

</head>

<body>

<table width="731" height="497" border="0" align="center" cellpadding="0" cellspacing="0">

<tr>

<td width="724" height="497" align="center" background="images/main.gif" class="text13">&nbsp;</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\mainfra.html

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 1px;

margin-right: 0px;

margin-bottom: 0px;

background:url("images/main.jpg") no-repeat center center;

background-color:#52a9db;

}

html { overflow-x: ; overflow-y: ; border:0;}

-->

</style>

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td></td>

</tr>

<tr>

<td><table width="768" height="500" border="0" cellpadding="0" cellspacing="0">

</table></td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\manage.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

</head>

<frameset rows="59,\*" cols="\*" frameborder="no" border="0" framespacing="0">

<frame src="top.html" name="topFrame" scrolling="no" noresize="noresize" id="topFrame" title="topFrame" />

<frameset cols="213,\*" frameborder="no" border="0" framespacing="0">

<frame src="left.jsp" name="leftFrame" scrolling="no" noresize="noresize" id="leftFrame" title="leftFrame" />

<frame src="mainfra.html" name="mainFrame" id="mainFrame" title="mainFrame" />

</frameset>

</frameset>

<noframes><body>

</body>

</noframes></html>

\HRmanagement\WebRoot\MYJSP.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="com.zhangying.po.Job"%>

<%@ page import="com.zhangying.tool.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="jobForm" method="post" action="modifyjob.do?action=updatejob" onSubmit="return jobValidate();" >

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >修改人事信息</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<%

Job ob=(Job)request.getAttribute("job");

if(ob!=null){

%>

<input type="hidden" name="id" value="<%=ob.getId()%>"/>

<input type="hidden" name="job" value="<%=request.getParameter("job")%>"/>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>人事信息</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="9%">姓名：</td>

<td width="36%"><input name="name" type="text" class="input" value="<%=ob.getName()%>" disabled>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="9%">年龄：</td>

<td><input name="age" type="text" class="input" value="<%=ob.getAge()%>" disabled ></td>

<td><div align="right">职位：</div></td>

<td><input name="job" type="text" class="input" value="<%=ob.getJob()%>" >

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">所学专业：</td>

<td><input name="specialty" type="text" class="input" value="<%=ob.getSpecialty()%>" disabled >

<span class="red">\*</span></td>

<td><div align="right">工作经验：</div></td>

<td><input name="experience" type="text" class="input" value="<%=ob.getExperience()%>" disabled >

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">学历：</td>

<td><input name="studyeffort" type="text" class="input" value="<%=ob.getStudyeffort()%>" disabled >

<span class="red">\*</span></td>

<td><div align="right">毕业学校：</div></td>

<td><input name="school" type="text" class="input" value="<%=ob.getSchool()%>" disabled >

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">电话：</td>

<td><input name="tel" type="text" class="input" value="<%=ob.getTel()%>" disabled >

<span class="red">\*</span></td>

<td><div align="right">Email：</div></td>

<td><input name="email" type="text" class="input" value="<%=ob.getEmail()%>" disabled >

<span class="red">\*</span></td>

</tr>

<tr>

<td width="9%" nowrap align="right">详细经历：</td>

<td colspan="3">

<textarea name="content" cols="100" rows="6" class="input" id="content" value="<%=ob.getContent()%>" disabled >

</textarea></td>

</tr>

</table>

</fieldset>

</TD>

</TR>

</TABLE>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="4" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

<TR>

<TD colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</TR>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\top.html

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

}

-->

</style>

<link href="../css/css.css" rel="stylesheet" type="text/css" />

<style type="text/css">

<!--

.STYLE1 { font-size: 36px;

color: #009999;

font-family: "方正舒体";

}

-->

</style>

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="59" background="../images/top.gif"><table width="99%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td width="35%"><span class="STYLE1">人事管理信息系统</span></td>

<td width="65%" align="right" style="font-size:12px;vertical-align:bottom;">&nbsp;</td>

</tr>

</table></td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\updateeducate.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Educate"%>

<%@ page import="com.zhangying.tool.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="educateForm" method="post" action="modifyeducate.do?action=updateeducate" onSubmit="return educateValidate();">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >培训总结</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<%

Educate e=(Educate)request.getAttribute("educate");

if(e!=null){

%>

<input type="hidden" name="id" value="<%=e.getId()%>"/>

<input type="hidden" name="educate" value="<%=request.getParameter("educate")%>"/>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>培训计划</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">培训名称：</td>

<td colspan="3"><input name="name" type="text" class="input" value="<%=e.getName()%>" disabled/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">培训目的：</td>

<td colspan="3"><input type="text" name="purpose" class="input" value="<%=e.getPurpose()%>" disabled/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">培训开始时间：</td>

<td width="29%"><input type="text" name="begintime" class="input" value="<%=StringUtil.notNull(DateUtil.parseToString(e.getBegintime(),DateUtil.yyyyMMdd))%>" disabled/>

<span class="red">\*</span></td>

<td width="18%">培训结束时间：</td>

<td width="42%"><input type="text" name="endtime" class="input" value="<%=StringUtil.notNull(DateUtil.parseToString(e.getEndtime(),DateUtil.yyyyMMdd))%>" disabled/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">讲师：</td>

<td><input type="text" name="teacher" class="input" value="<%=e.getTeacher()%>" disabled/>

<span class="red">\*</span></td>

<td>培训人员：</td>

<td><input type="text" name="student" class="input" value="<%=e.getStudent()%>" disabled/>

<span class="red">\*</span>

</td>

</tr>

<tr>

<td width="11%" nowrap align="right">培训材料：</td>

<td colspan="3"><textarea name="datum" cols="100" rows="6" class="input" disabled><%=e.getDatum()%></textarea></td>

</tr>

<tr align="center">

<td width="11%" height="22" >培训效果：</td>

<td height="22" colspan="3" align="left" ><input name="effect" type="text" class="input" value="<%=StringUtil.notNull(e.getEffect())%>" size="50"></td>

</tr>

<tr align="center">

<td width="11%" height="22" >培训总结：</td>

<td height="22" colspan="3" align="left" ><textarea name="summarize" cols="50" rows="6" class="input"><%=StringUtil.notNull(e.getSummarize())%></textarea></td>

</tr>

</table>

<br />

</fieldset> </TD>

</TR>

</TABLE>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="4" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

<TR>

<TD colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/>

</TD>

</TR>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\updateinstitution.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Institution"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="institutionForm" method="post" action="modifyinstitution.do?action=updateinstitution" onSubmit="return institutionValidate()">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >修改奖惩信息</th>

</tr>

<%

Institution e=(Institution)request.getAttribute("institution");

if(e!=null){

%>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr>

<td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td>

</tr>

<tr>

<td width="100%">

<fieldset style="height:100%;">

<legend>修改奖惩信息</legend>

<input type="hidden" name="id" value="<%=e.getId()%>"/>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">奖惩名称：</td>

<td width="89%"><input name="name" type="text" class="input" value="<%=e.getName()%>"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">奖惩原因：</td>

<td><input name="reason" type="text" class="input" value="<%=e.getReason()%>"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td width="11%" nowrap align="right">奖惩说明：</td>

<td><textarea name="explain" cols="100" rows="6" class="input" id="explain"><%=e.getExplain()%></textarea></td>

</tr>

</table>

</fieldset>

</td>

</tr>

</table>

</td>

</tr>

<tr>

<td colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="2" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

</tabel>

</table>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\updatejob.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Job"%>

<%@ page import="com.zhangying.tool.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="jobForm" method="post" action="modifyjob.do?action=updatejob&isstock=1" onSubmit="return jobValidate();" >

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >修改人事信息</th>

</tr>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<%

Job ob=(Job)request.getAttribute("job");

if(ob!=null){

%>

<input type="hidden" name="id" value="<%=ob.getId()%>"/>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>人事信息</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="9%">姓名：</td>

<td width="36%"><input name="name" type="text" class="input" value="<%=ob.getName()%>" disabled>

<span class="red">\*</span></td>

<td width="12%"><div align="right">性别：</div></td>

<td width="43%">

<input name="sex" type="radio" value="1" checked>男

<input name="sex" type="radio" value="0">女

</tr>

<tr>

<td nowrap align="right" width="9%">年龄：</td>

<td><input name="age" type="text" class="input" value="<%=ob.getAge()%>" ></td>

<td><div align="right">职位：</div></td>

<td><input name="job" type="text" class="input" value="<%=ob.getJob() %>" >

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">所学专业：</td>

<td><input name="specialty" type="text" class="input" value="<%=ob.getSpecialty()%>" disabled >

<span class="red">\*</span></td>

<td><div align="right">工作经验：</div></td>

<td><input name="experience" type="text" class="input" value="<%=ob.getExperience()%>" disabled >

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">学历：</td>

<td><input name="studyeffort" type="text" class="input" value="<%=ob.getStudyeffort()%>" disabled >

<span class="red">\*</span></td>

<td><div align="right">毕业学校：</div></td>

<td><input name="school" type="text" class="input" value="<%=ob.getSchool()%>" disabled >

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">电话：</td>

<td><input name="tel" type="text" class="input" value="<%=ob.getTel()%>" disabled >

<span class="red">\*</span></td>

<td><div align="right">Email：</div></td>

<td><input name="email" type="text" class="input" value="<%=ob.getEmail()%>" disabled >

<span class="red">\*</span></td>

</tr>

<tr>

<td width="9%" nowrap align="right">详细经历：</td>

<td colspan="3">

<textarea name="content" cols="100" rows="6" class="input" id="content" value="<%=ob.getContent()%>" disabled >

</textarea></td>

</tr>

</table>

</fieldset>

</TD>

</TR>

</TABLE>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="4" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

<TR>

<TD colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</TR>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\updateperformance.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Performance"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="performanceForm" method="post" action="modifyperformance.do?action=updateperformance" onSubmit="return PerformanceValidate()">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >修改绩效信息</th>

</tr>

<%

Performance e=(Performance)request.getAttribute("performance");

if(e!=null){

%>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr>

<td align="left">

<input type="submit"value="保存" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td>

</tr>

<tr>

<td width="100%">

<fieldset style="height:100%;">

<legend>修改绩效信息</legend>

<input type="hidden" name="id" value="<%=e.getId()%>"/>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="11%">绩效分数：</td>

<td width="89%"><input name="score" type="text" class="input" value="<%=e.getScore()%>"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">增加工资：</td>

<td><input name="addmoney" type="text" class="input" value="<%=e.getAddmoney()%>"/>

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right" width="11%">创建时间：</td>

<td width="89%"><input name="createtime" type="text" class="input" value="<%=e.getCreatetime()%>" disabled/>

<span class="red">\*</span></td>

</tr>

<tr>

<td width="11%" nowrap align="right">绩效说明：</td>

<td><textarea name="content" cols="100" rows="6" class="input" id="con"><%=e.getContent()%></textarea></td>

</tr>

</table>

</fieldset>

</td>

</tr>

</table>

</td>

</tr>

<tr>

<td colspan="2" align="center" height="50px">

<input name="提交" type="submit" class="button" value="保存"/>

<input name="重置" type="reset" class="button" value="重置"/></TD>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="2" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

</tabel>

</table>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\updatestipend.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.po.Stipend"%>

<%@ page import="com.zhangying.tool.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="stipendForm" method="post" action="modifystipend.do?action=updatestipend" onSubmit="return stipendValidate();" >

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >薪金数据更新</th>

</tr>

<%

Stipend e=(Stipend)request.getAttribute("stipend");

if(e!=null){

%>

<tr>

<td class="CPanel">

<table border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="更新" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td>

</tr>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>薪金数据</legend>

<table border="0" cellpadding="2" cellspacing="1" style="width:100%">

<input type="hidden" name="id" value="<%=e.getId()%>"/>

<tr>

<td nowrap align="right" >员工姓名：</td>

<td width="36%"><input name="name" type="text" class="input" value="<%=e.getName()%>"/><span class="red">\*</span></td>

<td nowrap align="right" >基本薪金：</td>

<td width="36%"><input name="basic" type="text" id ="basic" class="input" value="<%=e.getBasic()%>"/><span class="red">\*</span></td>

</tr>

<tr>

<td nowrap height="36" align="right">全&nbsp;勤&nbsp;奖：</td>

<td><input name="duty" type="text" class="input" id="duty" value="<%=e.getDuty()%>" /></td>

<td nowrap align="right">提&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;成：</td>

<td><input name="other" type="text" class="input" id="other" value="<%=e.getOther()%>"></td>

</tr>

<tr>

<td nowrap height="36"><div align="right">罚&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;款：</div></td>

<td><input name="punishment" type="text" class="input" id="punishment" value="<%=e.getPunishment()%>" /></td>

<td nowrap align="right">发放时间：</td>

<td><input name="granttime" type="text" class="input" id="granttime" value="<%=StringUtil.notNull(DateUtil.parseToString(e.getGranttime(),DateUtil.yyyyMMdd))%>" />

<span class="red">\*</span></td>

</tr>

<tr>

<td colspan="4"><div align="center">注：金钱单位（元/RMB）</div></td>

</tr>

</table>

<br />

</fieldset> </TD>

</TR>

</TABLE>

</td>

</tr>

<tr>

<td colspan="2" align="center" height="32">

<input name="更新" type="submit" class="button" value="更新"/>

<input name="重置" type="reset" class="button" value="重置"/>

</td>

</tr>

<%}else{ %>

<tr>

<td height="22" colspan="4" align="center" >该信息不存在！！！</td>

</tr>

<%}%>

</tabel>

</table>

</body>

</html>

\HRmanagement\WebRoot\updateuser.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<%@ page import="java.util.List"%>

<%@ page import="java.util.Iterator"%>

<%@ page import="com.zhangying.tool.\*"%>

<%@ page import="com.zhangying.po.Users"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>人事管理信息系统</title>

<link rel="stylesheet" rev="stylesheet" href="css/style.css " type="text/css" media="all" />

<script type="text/javascript" src="Js/typem.js"></script>

<script type="text/javascript" src="Js/js.js"></script>

<style type="text/css">

<!--

.atten {font-size:12px;font-weight:normal;color:#F00;}

-->

</style>

</head>

<body class="ContentBody">

<form name="userForm" method="post" action="modifyuser.do?action=updateuser" onSubmit="return userValidate();">

<div class="MainDiv">

<table width="99%" border="0" cellpadding="0" cellspacing="0" class="CContent">

<tr>

<th class="tablestyle\_title" >人员信息更新</th>

</tr>

<% Users u=(Users)request.getAttribute("user");

if(u!=null){

%>

<input type="hidden" name="id" value="<%=u.getId()%>"/>

<tr>

<td class="CPanel">

<table width="90%" border="0" cellpadding="0" cellspacing="0" style="width:80%" align="center">

<tr><td align="left">

<input type="submit"value="更新" class="button"/>

<input type="button" value="返回" onclick="javascript:history.back();" class="button"/>

</td></tr>

<TR>

<TD width="100%">

<fieldset style="height:100%;">

<legend>人员信息</legend>

<table width="100%" border="0" cellpadding="2" cellspacing="1" style="width:100%">

<tr>

<td nowrap align="right" width="9%">用户名：</td>

<td width="36%"><input name="username" type="text" class="input" id="username" value="<%=u.getUsername()%>">

<span class="red">\*</span></td>

<td width="12%"><div align="right">登录密码：</div></td>

<td width="43%"><input name="password" type="text" class="input" id="password" value="<%=u.getPassword()%>"/>

<span class="red">\*</span></tr>

<tr>

<td nowrap align="right" width="9%">性别：</td>

<td><input name="sex" type="radio" value="1" <%=new Byte("1").equals(u.getSex())?"checked":""%>> 男

<input name="sex" type="radio" value="0" <%=new Byte("1").equals(u.getSex())?"":"checked"%>>女</td>

<td><div align="right">出生日期：</div></td>

<td><input name="birthday" type="text" class="input" id="birthday" value="<%=StringUtil.notNull(DateUtil.parseToString(u.getBirthday(),DateUtil.yyyyMMdd))%>">

<span class="red">\*</span></td>

</tr>

<tr>

<td nowrap align="right">是否管理员：</td>

<td><input name="isadminhelp" type="checkbox" <%=new Byte("1").equals(u.getIsadmin())?"checked":""%> onClick="javascript:adminChecked();" >

<input type="hidden" name="isadmin" value="<%=u.getIsadmin()%>"></td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td nowrap align="right">人员简介：</td>

<td colspan="3"><textarea name="content" cols="100" rows="6" class="input" id="content"><%=u.getContent()%></textarea></td>

</tr>

</table>

<br />

</fieldset>

</TD>

</TR>

</TABLE>

</td>

</tr>

<%

}else{

%>

<tr>

<td height="22" colspan="2" align="center" >没有查到该人员信息！！！</td>

</tr>

<%}%>

<TR>

<TD colspan="2" align="center" height="50px">

<input type="submit" class="button" value="更新"/>

<input type="reset" class="button" value="重置"/></TD>

</TR>

</TABLE>

</div>

</form>

</body>

</html>

\HRmanagement\WebRoot\welcome.jsp

<%@ page contentType="text/html; charset=GBK" language="java" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>麻麻鱼府人事管理信息系统</title>

<style type="text/css">

<!--

body {

margin-left: 0px;

margin-top: 0px;

margin-right: 0px;

margin-bottom: 0px;

}

-->

</style>

<link href="css/css.css" rel="stylesheet" type="text/css" />

<style type="text/css">

<!--

.STYLE1 {

font-size: 36px;

color: #009999;

font-family: "方正舒体";

}

-->

</style>

</head>

<body>

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td height="147" background="images/top02.gif"><div align="center"><span class="STYLE1">人事管理信息系统</span></div></td>

</tr>

</table>

<table width="562" border="0" align="center" cellpadding="0" cellspacing="0" class="right-table03">

<tr>

<td width="221"><table width="95%" border="0" cellpadding="0" cellspacing="0" class="login-text01">

<tr>

<td><table width="100%" border="0" cellpadding="0" cellspacing="0" class="login-text01">

<tr>

<td align="center"><img src="images/ico13.gif" width="107" height="97" /></td>

</tr>

<tr>

<td height="40" align="center">&nbsp;</td>

</tr>

</table></td>

<td><img src="images/line01.gif" width="5" height="292" /></td>

</tr>

</table></td>

<td>

<form name="usersForm" method="post" action="logon.do?action=logon">

<table width="100%" border="0" cellspacing="0" cellpadding="0">

<tr>

<td colspan="2" align="center"><font color="red">${requestScope.error}</font></td>

</tr>

<tr>

<td width="31%" height="35" class="login-text02">用户名：<br /></td>

<td width="69%"><input name="username" type="text" id="username" size="20" /></td>

</tr>

<tr>

<td height="35" class="login-text02">密　码：<br /></td>

<td><input name="password" type="password" id="password" size="20" /></td>

</tr>

<tr>

<td height="35">&nbsp;</td>

<td><input type="submit" class="right-button01" value="确认登录"/>

<input type="reset" class="right-button02" value="重 置" /></td>

</tr>

</table>

</form>

</td>

</tr>

</table>

</body>

</html>

\HRmanagement\WebRoot\WEB-INF\structs-config.xml

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE struts-config PUBLIC "-//Apache Software Foundation//DTD Struts Configuration 1.3//EN" "http://struts.apache.org/dtds/struts-config\_1\_3.dtd">

<struts-config>

<form-beans>

<form-bean name="userForm" type="com.zhangying.action.UsersForm" />

<form-bean name="jobForm" type="com.zhangying.action.JobForm" />

<form-bean name="educateForm" type="com.zhangying.action.EducateForm" />

<form-bean name="institutionForm" type="com.zhangying.action.InstitutionForm" />

<form-bean name="stipendForm" type="com.zhangying.action.StipendForm" />

<form-bean name="performanceForm" type="com.zhangying.action.PerformanceForm" />

</form-beans>

<global-forwards>

<forward name="error" path="/error.jsp" />

</global-forwards>

<action-mappings>

<action forward="/welcome.jsp" path="/index" />

<action forward="/main.jsp" path="/main" />

<action forward="/adduser.jsp" path="/adduser" />

<action forward="/addjob.jsp" path="/addjob" />

<action forward="/addeducate.jsp" path="/addeducate" />

<action forward="/addinstitution.jsp" path="/addinstitution" />

<action forward="/addstipend.jsp" path="/addstipend" />

<action forward="/addperformance.jsp" path="/addperformance" />

<action name="userForm" path="/logon" scope="request" type="com.zhangying.action.UsersAction">

<forward name="success" path="/manage.jsp" />

<forward name="failed" path="/welcome.jsp" />

</action>

<action name="userForm" path="/user" scope="request" type="com.zhangying.action.UsersAction">

<forward name="success" path="/listuser.jsp" />

</action>

<action name="userForm" path="/modifyuser" scope="session" type="com.zhangying.action.UsersAction">

<forward name="success" path="/user.do?action=listuser" />

</action>

<action path="/selectuser" scope="request" type="com.zhangying.action.UsersAction">

<forward name="success" path="/updateuser.jsp" />

</action>

<action name="jobForm" path="/job" scope="request" type="com.zhangying.action.JobAction">

<forward name="success" path="/listjob.jsp" />

</action>

<action name="jobForm" path="/modifyjob" scope="session" type="com.zhangying.action.JobAction">

<forward name="success" path="/job.do?action=listjob" />

</action>

<action path="/detailjob" scope="request" type="com.zhangying.action.JobAction">

<forward name="success" path="/detailjob.jsp" />

</action>

<action path="/updatejob" scope="request" type="com.zhangying.action.JobAction">

<forward name="success" path="/updatejob.jsp" />

</action>

<action name="educateForm" path="/educate" scope="request" type="com.zhangying.action.EducateAction">

<forward name="success" path="/listeducate.jsp" />

</action>

<action name="educateForm" path="/modifyeducate" scope="session" type="com.zhangying.action.EducateAction">

<forward name="success" path="/educate.do?action=listeducate" />

</action>

<action path="/detaileducate" scope="request" type="com.zhangying.action.EducateAction">

<forward name="success" path="/detaileducate.jsp" />

</action>

<action path="/updateeducate" scope="request" type="com.zhangying.action.EducateAction">

<forward name="success" path="/updateeducate.jsp" />

</action>

<action name="institutionForm" path="/institution" scope="request" type="com.zhangying.action.InstitutionAction">

<forward name="success" path="/listinstitution.jsp" />

</action>

<action name="institutionForm" path="/modifyinstitution" scope="session" type="com.zhangying.action.InstitutionAction">

<forward name="success" path="/institution.do?action=listinstitution" />

</action>

<action path="/detailinstitution" scope="request" type="com.zhangying.action.InstitutionAction">

<forward name="success" path="/detailinstitution.jsp" />

</action>

<action path="/updateinstitution" scope="request" type="com.zhangying.action.InstitutionAction">

<forward name="success" path="/updateinstitution.jsp" />

</action>

<action name="stipendForm" path="/stipend" scope="request" type="com.zhangying.action.StipendAction">

<forward name="success" path="/liststipend.jsp" />

</action>

<action name="stipendForm" path="/modifystipend" scope="session" type="com.zhangying.action.StipendAction">

<forward name="success" path="/stipend.do?action=liststipend" />

</action>

<action path="/detailstipend" scope="request" type="com.zhangying.action.StipendAction">

<forward name="success" path="/detailstipend.jsp" />

</action>

<action path="/updatestipend" scope="request" type="com.zhangying.action.StipendAction">

<forward name="success" path="/updatestipend.jsp" />

</action>

<action name="performanceForm" path="/performance" scope="request" type="com.zhangying.action.PerformanceAction">

<forward name="success" path="/listperformance.jsp" />

</action>

<action name="performanceForm" path="/modifyperformance" scope="session" type="com.zhangying.action.PerformanceAction">

<forward name="success" path="/performance.do?action=listperformance" />

</action>

<action path="/detailperformance" scope="request" type="com.zhangying.action.PerformanceAction">

<forward name="success" path="/detailperformance.jsp" />

</action>

<action path="/updateperformance" scope="request" type="com.zhangying.action.PerformanceAction">

<forward name="success" path="/updateperformance.jsp" />

</action>

</action-mappings>

</struts-config>

HRmanagement\WebRoot\WEB-INF\web.xml

<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xmlns:web="http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd" version="2.5">

<servlet>

<servlet-name>action</servlet-name>

<servlet-class>com.zhangying.tool.ActionServlet</servlet-class>

<init-param>

<param-name>config</param-name>

<param-value>/WEB-INF/struts-config.xml</param-value>

</init-param>

<init-param>

<param-name>debug</param-name>

<param-value>3</param-value>

</init-param>

<init-param>

<param-name>detail</param-name>

<param-value>3</param-value>

</init-param>

<load-on-startup>0</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>action</servlet-name>

<url-pattern>\*.do</url-pattern>

</servlet-mapping>

<welcome-file-list>

<welcome-file>index.jsp</welcome-file>

</welcome-file-list>

</web-app>

1. [1] 陶永明，企业人事管理系统设计. 东北财经大学，2011 [↑](#footnote-ref-1)
2. [2]郑環，企业人事管理系统的设计与实现. 电子科技大学，2010

   [3]陈燕，重庆大学人事管理信息系统的设计与开饭. 重庆大学，2002 [↑](#footnote-ref-2)
3. [4] 蔡琴，餐饮管理系统的设计与开发. 电子科技大学，2007 [↑](#footnote-ref-3)
4. [5] 戚云龙，大中型餐厅综合管理系统设计与实现. 大连理工大学，2012 [↑](#footnote-ref-4)
5. [6] 许秀芳，徐森，基于Web的餐饮连锁企业工资人事管理系统设计. 中国电子学会，2016 [↑](#footnote-ref-5)
6. [7] 李中明，大型连锁餐饮企业人力资源管理系统的研究. 上海财经大学，2010 [↑](#footnote-ref-6)
7. [8] 周鹏飞，基于嵌入式的餐饮系统设计与实现. 电子科技大学，2014 [↑](#footnote-ref-7)
8. [9] 刘学治，从“人本管理”谈餐饮业的人事管理. 四川烹饪，1997 [↑](#footnote-ref-8)
9. [10] 吴姣，餐饮零售管理大提速——天维思专注供应链实时预测与计划协调. 物流时代，2016 [↑](#footnote-ref-9)
10. [11] 邢美东，基于数据挖掘的餐饮管理系统的设计与实现. 东北大学，2008

    [12] 董浩，内蒙古聚祥餐饮管理有限公司薪酬制度设计与研究. 内蒙古大学，2011 [↑](#footnote-ref-10)
11. [13] 许丹，基于供应链的餐饮业管理系统. 南京理工大学，2007 [↑](#footnote-ref-11)
12. [14] 瞿建，二连浩特酒店管理系统设计与实现. 大连理工大学，2012 [↑](#footnote-ref-12)
13. [15] 杨超超，浙江省连锁餐饮企业人力资源管理存在问题及对策研究. 中国管理信息化，2016

    [16] 王洪海，浅析我国餐饮管理现状及对策. 商业故事，2016 [↑](#footnote-ref-13)
14. [17] 范海明，浅析管理信息系统在酒店中的应用. 电子制作，2015 [↑](#footnote-ref-14)
15. [17] 范海明，浅析管理信息系统在酒店中的应用. 电子制作，2015

    [18] 付佳，人事管理系统的设计与实现. 大连海事大学，2015 [↑](#footnote-ref-15)
16. [19] 巫晓聪，基于WEB的企业人事信息管理系统的设计与实现. 电子科技大学，2013

    [20] 赵炳森，基于Hibernate的人事管理系统. 华东师范大学，2009

    [21] 陈卫琴，基于Struts和Hibernate的人事管理系统的分析与设计. 南京理工大学，2007 [↑](#footnote-ref-16)
17. [22] 史洪源，企业人事管理系统的设计与实现. 吉林大学，2014 [↑](#footnote-ref-17)
18. [23] 李相会，人事管理系统的设计与实现. 电子科技大学，2013 [↑](#footnote-ref-18)