

**《Java Web技术》课程设计**

**题目：学生信息管理系统**

**院、 系： 计算机信息与技术系**

**学科专业： 软件工程**

**学 号： B14060405**

**学生姓名： 范 卓**

**指导教师： 刘彩红**

×××2016年12月23日

**中文摘要**

近年来，我国高等职业教育迎来了蓬勃发展的新局面，各院校招生规模不断扩大，学校的教学管理负担越来越重。采用传统的学生信息管理模式(如直接手工管理或者开放性程度不高的管理系统)不可避免的浪费大量的人力、物力和财力。如何提高学生信息管理的水平，建立适合自己院校的学生信息管理系统，已成为学校信息化建设过程中亟特解决的重要问题。 从实用性和经济性考虑，构建基于WEB技术的网络管理平台是当前大多数院校教学管理的一个重要任务和发展方向。运用WEB技术，建设学校学生信息管理平台，旨在探索一种以互联网为基础的教学管理模式。通过这种新的管理模式，为学院营造一种新的教学管理环境，使管理突破时空限制，提高工作效率和管理水平，使学校管理者、教师和学生可以在任何时候、任何地点通过网络进行学习与交流。 本文主要论述了基于WEB架构，采用JSP技术设计与开发学生信息管理系统的实现方案。论文首先介绍了国内信息管理系统研究的现状与发展趋势，其次论述了采用C／S体系结构，MySql数据库系统的开发平台，从信息管理系统的需求分析入手，阐述了本系统的各个模块的功能，和后台数据库、系统登录、用户管理、学生信息管理的具体设计与实现。 本文充分考虑系统的安全性，对系统采用MVC设计模式，采用dao直接访问后台连接的util数据库，使数据访问过程简单，访问效率高，从而使整个信息管理系统始终能够高效、稳定的运行和应对比较大的访问流量。 实践证明，本系统的开发对于同类系统的开发和设计具有一定的推广和实用价值。

目 录

[第一章 项目概述 3](#_Toc470283370)

[1.1 需求分析 3](#_Toc470283371)

[1.2 功能描述 3](#_Toc470283372)

[第二章 系统总体设计 4](#_Toc470283373)

[2.1 系统功能图 4](#_Toc470283374)

[2.2数据库设计 4](#_Toc470283375)

[2.3框架设计 6](#_Toc470283376)

[第三章 详细设计 7](#_Toc470283377)

[3.1各模块设计 7](#_Toc470283378)

[3.1.1. Com.dawn.dao模块 7](#_Toc470283379)

[3.1.2. Com.dawn.model模块 9](#_Toc470283380)

[3.1.3. Com.dawn.util模块 9](#_Toc470283381)

[3.1.4. Com.dawn.web模块 11](#_Toc470283382)

[3.1.5. Jsp页面 15](#_Toc470283383)

[第四章 运行与测试 21](#_Toc470283384)

[第五章 设计体会与小结 26](#_Toc470283385)

[第六章 参考文献 27](#_Toc470283386)

第一章 项目概述

## 1.1 需求分析

1.设计一个较完善的学生信息管理系统；

2.用户模块：提供用户注册、登录、查询信息、修改密码、联系管理员，并考虑用户数量增减，需要用数据库技术；

3.后台登陆模块：实现与数据库相连，对管理员的身份进行验证；

4.后台管理模块：实现用户信息的增删改查，并能实现所有用户信息的浏览；

5.数据库模块：用户数据表（t\_user），管理员数据表（t\_admin）；

6.界面处理：要求操作简单，界面美化。

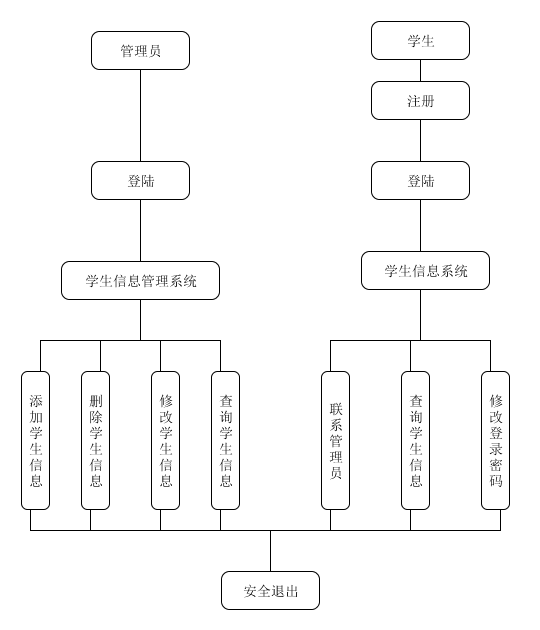
## 1.2 功能描述

1. 前端功能：用户注册、登录、查询信息、修改密码、联系管理员；

2. 后台管理：管理员登陆、对学生信息增删改查，以及查看所有用户信息并且实现分页和每页显示固定学生信息。

第二章 系统总体设计

## 2.1 系统功能图



## 2.2数据库设计

**根据实体的分析，对应的数据表，分别是：**

1. 学生信息表(t\_user)

DROP TABLE IF EXISTS `t\_user`;

CREATE TABLE `t\_user` (

`id` int(4) NOT NULL AUTO\_INCREMENT,

`userName` varchar(10) COLLATE utf8\_bin NOT NULL,

`password` varchar(20) COLLATE utf8\_bin NOT NULL,

`name` varchar(10) COLLATE utf8\_bin NOT NULL,

`age` int(4) NOT NULL,

`sex` varchar(2) COLLATE utf8\_bin NOT NULL,

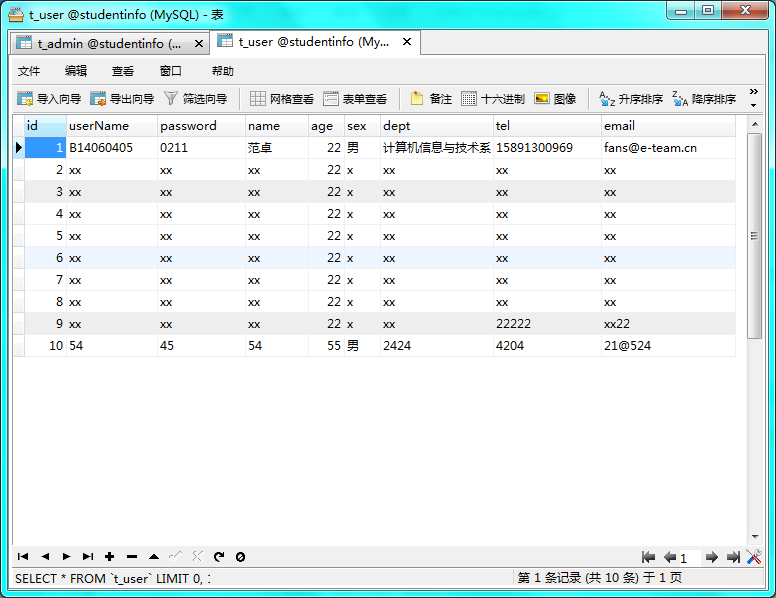
`dept` varchar(15) COLLATE utf8\_bin NOT NULL,

`tel` varchar(15) COLLATE utf8\_bin NOT NULL,

`email` varchar(30) COLLATE utf8\_bin NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=12 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;



1. 管理员登陆信息表(t\_admin)

DROP TABLE IF EXISTS `t\_admin`;

CREATE TABLE `t\_admin` (

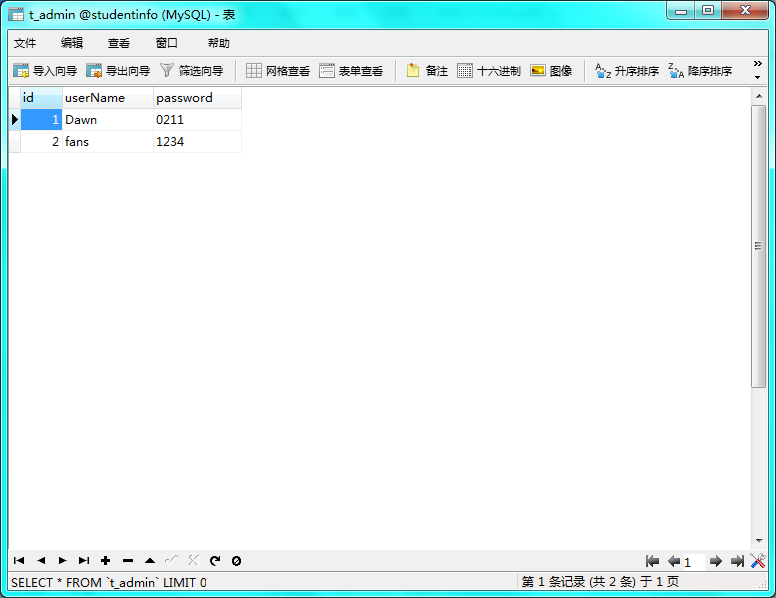
`id` int(4) NOT NULL AUTO\_INCREMENT,

`userName` varchar(10) COLLATE utf8\_bin NOT NULL,

`password` varchar(20) COLLATE utf8\_bin NOT NULL,

PRIMARY KEY (`id`)

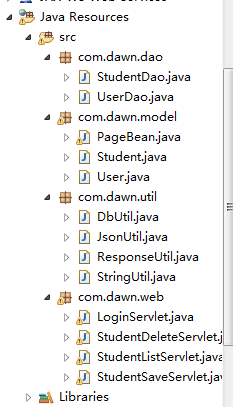
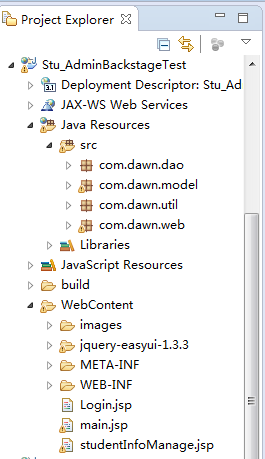
) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;



## 2.3框架设计

前端框架：（暂未实现）

后台框架：



第三章 详细设计

## 3.1各模块设计

3.1.1. Com.dawn.dao模块：

主要实现数据库的操作，从web获取到用户请求的功能对封装的util下的数据库进行增删改查操作；

类：StudentDao.java、UserDao.java

核心代码：

**public** **class** StudentDao {

**public** ResultSet studentList(Connection con,PageBean pageBean,Student student)**throws** Exception{

StringBuffer sb=**new** StringBuffer("select \* from t\_user");

**if**(StringUtil.*isNotEmpty*(student.getUserName())){

sb.append(" and userName like '%"+student.getUserName()+"%'");

}

**if**(pageBean!=**null**){

sb.append(" limit "+pageBean.getStart()+","+pageBean.getRows());

}

PreparedStatement pstmt=con.prepareStatement(sb.toString().replaceFirst("and", "where"));

**return** pstmt.executeQuery();

}

**public** **int** studentCount(Connection con,Student student)**throws** Exception{

StringBuffer sb=**new** StringBuffer("select count(\*) as total from t\_user");

**if**(StringUtil.*isNotEmpty*(student.getUserName())){

sb.append(" and userName like '%"+student.getUserName()+"%'");

}

PreparedStatement pstmt=con.prepareStatement(sb.toString().replaceFirst("and", "where"));

ResultSet rs=pstmt.executeQuery();

**if**(rs.next()){

**return** rs.getInt("total");

}**else**{

**return** 0;

}

}

//删除

**public** **int** studentDelete(Connection con,String delIds)**throws** Exception{

String sql="delete from t\_user where id in("+delIds+")";

PreparedStatement pstmt=con.prepareStatement(sql);

**return** pstmt.executeUpdate();

}

//添加

**public** **int** studentAdd(Connection con,Student student)**throws** Exception{

String sql="insert into t\_user values(null,?,?,?,?,?,?,?,?)";

PreparedStatement pstmt=con.prepareStatement(sql);

pstmt.setString(1, student.getUserName());

pstmt.setString(2, student.getPassword());

pstmt.setString(3, student.getName());

pstmt.setInt(4, student.getAge());

pstmt.setString(5, student.getSex());

pstmt.setString(6, student.getDept());

pstmt.setString(7, student.getTel());

pstmt.setString(8, student.getEmail());

**return** pstmt.executeUpdate();

}

//修改

**public** **int** studentModify(Connection con,Student student)**throws** Exception{

String sql="update t\_user set userName=?,password=?,name=?,age=?,sex=?,dept=?,tel=?,email=? where id=?";

PreparedStatement pstmt=con.prepareStatement(sql);

pstmt.setString(1, student.getUserName());

pstmt.setString(2, student.getPassword());

pstmt.setString(3, student.getName());

pstmt.setInt(4, student.getAge());

pstmt.setString(5, student.getSex());

pstmt.setString(6, student.getDept());

pstmt.setString(7, student.getTel());

pstmt.setString(8, student.getEmail());

pstmt.setInt(9, student.getId());

**return** pstmt.executeUpdate();

}

}

**public** **class** UserDao {

**public** User login(Connection con,User user) **throws** Exception{

User resultUser=**null**;

String sql="select \* from t\_admin where userName=? and password=?";

PreparedStatement pstmt=con.prepareStatement(sql);

pstmt.setString(1, user.getUserName());

pstmt.setString(2, user.getPassword());

ResultSet rs=pstmt.executeQuery();

**if**(rs.next()){

resultUser=**new** User();

resultUser.setUserName(rs.getString("userName"));

resultUser.setPassword(rs.getString("password"));

}

**return** resultUser;

}

}

### 3.1.2. Com.dawn.model模块：

主要实现变量的定义，以及数据库分页的实现；

类：PageBean.java、Student.java、User.java

核心代码：

**public** **class** PageBean {

**private** **int** page; // 第几页 当前页

**private** **int** rows; // 每页记录数

**private** **int** start; // 起始页0

**public** PageBean(**int** page, **int** rows) {

**super**();

**this**.page = page;

**this**.rows = rows;

}

Get、set方法略

}

**public** **class** Student {

**private** **int** id;

**private** String userName;

**private** String password;

**private** String name;

**private** **int** age;

**private** String sex;

**private** String dept;

**private** String tel;

**private** String email;

**public** Student(){

**super**();

}

**public** Student(String userName,String password,String name,**int** age,String sex,String dept,String tel,String email){

**super**();

**this**.userName = userName;

**this**.password = password;

**this**.name = name;

**this**.age = age;

**this**.sex = sex;

**this**.dept = dept;

**this**.tel = tel;

**this**.email = email;

}

Get、set方法略

}

**public** **class** User {

**private** **int** id;

**private** String userName;

**private** String password;

**public** User() {

**super**();

// **TODO** Auto-generated constructor stub

}

**public** User(String userName, String password) {

**super**();

**this**.userName = userName;

**this**.password = password;

}

Get、set方法略

}

### 3.1.3. Com.dawn.util模块：

主要实现数据库信息的封装；

类：DbUtil.java、JsonUtil.java、ResponseUtil.java、

核心代码：

**public** **class** DbUtil {

**private** String dbUrl="jdbc:mysql://localhost:3306/studentInfo";

**private** String dbUserName="root";

**private** String dbPassword="0211";

**private** String jdbcName="com.mysql.jdbc.Driver";

**public** Connection getCon() **throws** Exception{

Class.*forName*(jdbcName);

Connection con=DriverManager.*getConnection*(dbUrl,dbUserName,dbPassword);

**return** con;

}

**public** **void** closeCon(Connection con) **throws** Exception{

**if**(con!=**null**){

con.close();

}

}

**public** **static** **void** main(String[] args) {

DbUtil dbUtil=**new** DbUtil();

**try** {

dbUtil.getCon();

System.***out***.println("数据库连接成功");

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

**public** **class** JsonUtil {

**public** **static** JSONArray formatRsToJsonArray(ResultSet rs)**throws** Exception{

ResultSetMetaData md=rs.getMetaData();

**int** num=md.getColumnCount();

JSONArray array=**new** JSONArray();

**while**(rs.next()){

JSONObject mapOfColValues=**new** JSONObject();

**for**(**int** i=1;i<=num;i++){

mapOfColValues.put(md.getColumnName(i), rs.getObject(i));

}

array.add(mapOfColValues);

}

**return** array;

}

}

**public** **class** ResponseUtil {

**public** **static** **void** write(HttpServletResponse response,JSONObject jsonObject)**throws** Exception{

response.setContentType("text/html;charset=utf-8");

PrintWriter out=response.getWriter();

out.println(jsonObject.toString());

out.flush();

out.close();

}

}

**public** **class** StringUtil {

**public** **static** **boolean** isEmpty(String str){

**if**("".equals(str)|| str==**null**){

**return** **true**;

}**else**{

**return** **false**;

}

}

**public** **static** **boolean** isNotEmpty(String str){

**if**(!"".equals(str)&&str!=**null**){

**return** **true**;

}**else**{

**return** **false**;

}

}

}

### 3.1.4. Com.dawn.web模块：

主要实现控制功能，对jsp页面传递的参数进行传送到dao模块进行处理，最后返回给用户的另外jsp界面；

类：LoginServlet.java、StudentDeleteServlet.java、StudentListServlet.java、StudentSaveServlet.java

核心代码：

**public** **class** LoginServlet **extends** HttpServlet{

DbUtil dbUtil=**new** DbUtil();

UserDao userDao=**new** UserDao();

@Override

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

**this**.doPost(request, response);

}

@Override

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

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

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

request.setAttribute("userName", userName);

request.setAttribute("password", password);

**if**(StringUtil.*isEmpty*(userName)||StringUtil.*isEmpty*(password)){

request.setAttribute("error", "用户名或密码为空！");

request.getRequestDispatcher("Login.jsp").forward(request, response);

**return**;

}

User user=**new** User(userName,password);

Connection con=**null**;

**try** {

con=dbUtil.getCon();

User currentUser=userDao.login(con, user);

**if**(currentUser==**null**){

request.setAttribute("error", "用户名或密码错误！");

// 服务器跳转

request.getRequestDispatcher("Login.jsp").forward(request, response);

}**else**{

// 获取Session

HttpSession session=request.getSession();

session.setAttribute("currentUser", currentUser);

// 客户端跳转

response.sendRedirect("main.jsp");

}

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}**finally**{

**try** {

dbUtil.closeCon(con);

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

}

**public** **class** StudentDeleteServlet **extends** HttpServlet{

DbUtil dbUtil=**new** DbUtil();

StudentDao studentDao=**new** StudentDao();

@Override

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

**this**.doPost(request, response);

}

@Override

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

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

Connection con=**null**;

**try**{

con=dbUtil.getCon();

JSONObject result=**new** JSONObject();

**int** delNums=studentDao.studentDelete(con, delIds);

**if**(delNums>0){

result.put("success", "true");

result.put("delNums", delNums);

}**else**{

result.put("errorMsg", "删除失败");

}

ResponseUtil.*write*(response, result);

}**catch**(Exception e){

e.printStackTrace();

}**finally**{

**try** {

dbUtil.closeCon(con);

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

}

**public** **class** StudentListServlet **extends** HttpServlet{

DbUtil dbUtil=**new** DbUtil();

StudentDao studentDao=**new** StudentDao();

@Override

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

**this**.doPost(request, response);

}

@Override

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

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

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

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

**if**(userName==**null**){

userName="";

}

Student student = **new** Student();

student.setUserName(userName);

PageBean pageBean=**new** PageBean(Integer.*parseInt*(page),Integer.*parseInt*(rows));

Connection con=**null**;

**try**{

con=dbUtil.getCon();

JSONObject result=**new** JSONObject();

JSONArray jsonArray=JsonUtil.*formatRsToJsonArray*(studentDao.studentList(con,pageBean,student));

**int** total=studentDao.studentCount(con, student);

result.put("rows", jsonArray);

result.put("total", total);

ResponseUtil.*write*(response, result);

}**catch**(Exception e){

e.printStackTrace();

}**finally**{

**try** {

dbUtil.closeCon(con);

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

}

**public** **class** StudentSaveServlet **extends** HttpServlet{

DbUtil dbUtil=**new** DbUtil();

StudentDao studentDao=**new** StudentDao();

@Override

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

**this**.doPost(request, response);

}

@Override

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

request.setCharacterEncoding("utf-8");

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

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

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

**int** age=Integer.*parseInt*(request.getParameter("age"));

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

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

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

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

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

Student student=**new** Student(userName,password,name,age,sex,dept,tel,email);

**if**(StringUtil.*isNotEmpty*(id)){

student.setId(Integer.*parseInt*(id));

}

Connection con=**null**;

**try**{

con=dbUtil.getCon();

**int** saveNums=0;

JSONObject result=**new** JSONObject();

**if**(StringUtil.*isNotEmpty*(id)){

saveNums=studentDao.studentModify(con, student);

}**else**{

saveNums=studentDao.studentAdd(con, student);

}

**if**(saveNums>0){

result.put("success", "true");

}**else**{

result.put("success", "true");

result.put("errorMsg", "保存失败");

}

ResponseUtil.*write*(response, result);

}**catch**(Exception e){

e.printStackTrace();

}**finally**{

**try** {

dbUtil.closeCon(con);

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

}

3.1.5. Jsp页面：

主要实现界面的美化以及界面参数的传送接收等。

Jsp文件：Login.jsp、main.jsp、studentInfoManage.jsp

核心代码：

**Login.jsp**

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<title>学生信息管理系统登录</title>

<script type=*"text/javascript"*>

**function** resetValue(){

document.getElementById("userName").value="";

document.getElementById("password").value="";

}

</script>

<style type=*"text/css"*>

**a***:link***,a***:visited*{

font-size:*13px*;

text-decoration:*none*; /\*超链接无下划线\*/

}

**a***:hover*{

text-decoration:*underline*; /\*鼠标放上去有下划线\*/

color:*red*;

}

*.aa*{

font-size:*14px*;

font-weight:*bold*;

background:*aqua*;

}

*.bb*{

font-weight:*bold*;

background:*#82D900*;

}

*.button* {

width:*70px*; height:*25px*;

border-bottom-color:*aqua*;

border-left-color:*aqua*;

border-right-color:*aqua*;

border-top-color:*aqua*;

background:*#AE8F00*;

font-size: *16px*;

font-family: *"宋体"*;

font-weight:*bold*;

}

*.button:hover*{

color: *#339966*;

background-color: *#8CEA00*;

}

</style>

</head>

<body background=*"./images/bg.jpg"*>

<center>

<br/><br/><br/><br/><br/><br/><br/><br/>

<table border=*"0"* width=*"540"* height=*"80"* class=*"bb"*>

<tr>

<td align=*"center"*>

<h1>学生信息管理系统内容后台管理界面</h1>

</td>

</tr>

<table>

<form action=*"login"* method=*"post"*>

<table border=*"0"* width=*"540"* height=*"260"* class=*"aa"*>

<tr>

<td align=*"right"*>帐 &nbsp; 号： </td>

<td align=*"left"*><input type=*"text"* value=*"*${userName }*"* name=*"userName"* id=*"userName"*/></td>

</tr>

<tr>

<td align=*"right"*>密 &nbsp; 码： </td>

<td align=*"left"*><input type=*"password"* value=*"*${password }*"* name=*"password"* id=*"password"*/></td>

</tr>

<tr height=*"30"*>

<td width=*150* align=*"right"*><input type=*"submit"* value=*"登 陆"* class=*"button"*></td>

<td width=*350* align=*"center"*><input type=*"reset"* value=*"重 置"* onclick="resetValue()" class=*"button"*></td>

</tr>

<tr height=*"50"*>

<td width=*"40%"*></td>

<td colspan=*"3"*>

<font color=*"red"*>${error }</font>

</td>

</tr>

<tr height=*"20"*>

<font size=*"3px"*>

<td align=*"right"*>

版权所有：Dawn

</td>

<td align=*"left"*>

&nbsp; Email：fans@e-team.cn

</td>

</font>

</tr>

</table>

</form>

</center>

</body>

</html>

main.jsp

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=utf-8"*>

<title>学生信息管理系统主界面</title>

<%

// 权限验证

**if**(session.getAttribute("currentUser")==**null**){

System.out.println("error");

response.sendRedirect("Login.jsp");

**return**;

}

%>

<link rel=*"stylesheet"* type=*"text/css"* href=*"jquery-easyui-1.3.3/themes/default/easyui.css"*>

<link rel=*"stylesheet"* type=*"text/css"* href=*"jquery-easyui-1.3.3/themes/icon.css"*>

<script type=*"text/javascript"* src=*"jquery-easyui-1.3.3/jquery.min.js"*></script>

<script type=*"text/javascript"* src=*"jquery-easyui-1.3.3/jquery.easyui.min.js"*></script>

<script type=*"text/javascript"* src=*"jquery-easyui-1.3.3/locale/easyui-lang-zh\_CN.js"*></script>

<script type=*"text/javascript"*>

$(**function**(){

// 数据

**var** treeData=[{

text:"根目录",

children:[{

text:"学生信息管理",

attributes:{

url:"studentInfoManage.jsp"

}

}]

}];

// 实例化树菜单

$("#tree").tree({

data:treeData,

lines:**true**,

onClick:**function**(node){

**if**(node.attributes){

openTab(node.text,node.attributes.url);

}

}

});

// 新增Tab

**function** openTab(text,url){

**if**($("#tabs").tabs('exists',text)){

$("#tabs").tabs('select',text);

}**else**{

**var** content="<iframe frameborder='0' scrolling='auto' style='width:100%;height:100%' src="+url+"></iframe>";

$("#tabs").tabs('add',{

title:text,

closable:**true**,

content:content

});

}

}

});

</script>

</head>

<body class=*"easyui-layout"*>

<div region=*"north"* style="height: *80px*;background-color: *#E0EDFF*">

<div align=*"left"* style="width: *80%*;float: *left*"><img src=*"images/main.jpg"*></div>

<div style="padding-top: *50px*;padding-right: *20px*;">管理员： <font color=*"red"* >${currentUser.userName }</font></div>

</div>

<div region=*"center"*>

<div class=*"easyui-tabs"* fit=*"true"* border=*"false"* id=*"tabs"*>

<div title=*"首页"*>

<div align=*"center"* style="padding-top: *100px*;">

<font size=*"10"*>欢迎管理员 <font color=*"red"* >

${currentUser.userName }</font> 登陆学生信息管理系统</font>

</div>

</div>

</div>

</div>

<div region=*"west"* style="width: *150px*;" title=*"导航菜单"* split=*"true"*>

<ul id=*"tree"*></ul>

</div>

<div region=*"south"* style="height: *25px*;" align=*"center"*>版权所有：Dawn &nbsp; Email：fans@e-team.cn</div>

</body>

</html>

studentInfoManage.jsp

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=utf-8"*>

<title>学生信息管理</title>

<link rel=*"stylesheet"* type=*"text/css"* href=*"jquery-easyui-1.3.3/themes/default/easyui.css"*>

<link rel=*"stylesheet"* type=*"text/css"* href=*"jquery-easyui-1.3.3/themes/icon.css"*>

<script type=*"text/javascript"* src=*"jquery-easyui-1.3.3/jquery.min.js"*></script>

<script type=*"text/javascript"* src=*"jquery-easyui-1.3.3/jquery.easyui.min.js"*></script>

<script type=*"text/javascript"* src=*"jquery-easyui-1.3.3/locale/easyui-lang-zh\_CN.js"*></script>

<script type=*"text/javascript"*>

**function** searchStudent(){

$('#dg').datagrid('load',{

userName:$('#s\_userName').val()

});

}

**function** deleteStudent(){

**var** selectedRows=$("#dg").datagrid('getSelections');

**if**(selectedRows.length==0){

$.messager.alert("系统提示","请选择要删除的数据！");

**return**;

}

**var** strIds=[];

**for**(**var** i=0;i<selectedRows.length;i++){

strIds.push(selectedRows[i].id);

}

**var** ids=strIds.join(",");

$.messager.confirm("系统提示","您确认要删掉这<font color=red>"+selectedRows.length+"</font>条数据吗？",**function**(r){

**if**(r){

$.post("studentDelete",{delIds:ids},**function**(result){

**if**(result.success){

$.messager.alert("系统提示","您已成功删除<font color=red>"+result.delNums+"</font>条数据！");

$("#dg").datagrid("reload");

}**else**{

$.messager.alert('系统提示',result.errorMsg);

}

},"json");

}

});

}

**function** openStudentAddDialog(){

$("#dlg").dialog("open").dialog("setTitle","添加学生信息");

url="studentSave";

}

**function** openStudentModifyDialog(){

**var** selectedRows=$("#dg").datagrid('getSelections');

**if**(selectedRows.length!=1){

$.messager.alert("系统提示","请选择一条要编辑的数据！");

**return**;

}

**var** row=selectedRows[0];

$("#dlg").dialog("open").dialog("setTitle","编辑学生信息");

$("#fm").form("load",row);

url="studentSave?id="+row.id;

}

**function** closeStudentDialog(){

$("#dlg").dialog("close");

resetValue();

}

**function** resetValue(){

$("#userName").val("");

$("#password").val("");

$("#name").val("");

$("#age").val("");

$("#sex").val("");

$("#dept").val("");

$("#tel").val("");

$("#email").val("");

}

**function** saveStudent(){

$("#fm").form("submit",{

url:url,

onSubmit:**function**(){

**return** $(**this**).form("validate");

},

success:**function**(result){

**if**(result.errorMsg){

$.messager.alert("系统提示",result.errorMsg);

**return**;

}**else**{

$.messager.alert("系统提示","保存成功");

resetValue();

$("#dlg").dialog("close");

$("#dg").datagrid("reload");

}

}

});

}

</script>

</head>

<body>

<table id=*"dg"* title=*"学生 信息"* class=*"easyui-datagrid"* fitColumns=*"true"*

pagination=*"true"* rownumbers=*"true"* url=*"studentList"* fit=*"true"* toolbar=*"#tb"*>

<thead>

<tr>

<th field=*"cb"* checkbox=*"true"*></th>

<th field=*"userName"* width=*"50"*>学号</th>

<th field=*"password"* width=*"50"*>密码</th>

<th field=*"name"* width=*"50"*>姓名</th>

<th field=*"age"* width=*"30"*>年龄</th>

<th field=*"sex"* width=*"30"*>性别</th>

<th field=*"dept"* width=*"80"*>系别</th>

<th field=*"tel"* width=*"50"*>电话</th>

<th field=*"email"* width=*"60"*>邮箱</th>

</tr>

</thead>

</table>

<div id=*"tb"*>

<div>

<a href=*"javascript:openStudentAddDialog()"* class=*"easyui-linkbutton"* iconCls=*"icon-add"* plain=*"true"*>添加</a>

<a href=*"javascript:openStudentModifyDialog()"* class=*"easyui-linkbutton"* iconCls=*"icon-edit"* plain=*"true"*>修改</a>

<a href=*"javascript:deleteStudent()"* class=*"easyui-linkbutton"* iconCls=*"icon-remove"* plain=*"true"*>删除</a>

</div>

<div>&nbsp;学生学号：&nbsp;<input type=*"text"* name=*"s\_userName"* id=*"s\_userName"*/><a href=*"javascript:searchStudent()"* class=*"easyui-linkbutton"* iconCls=*"icon-search"* plain=*"true"*>搜索</a></div>

</div>

<div id=*"dlg"* class=*"easyui-dialog"* style="width: *400px*;height: *330px*;padding: *10px 20px*"

closed=*"true"* buttons=*"#dlg-buttons"*>

<form id=*"fm"* method=*"post"*>

<table>

<tr>

<td>学 号：</td>

<td><input type=*"text"* name=*"userName"* id=*"userName"* class=*"easyui-validatebox"* required=*"true"*/></td>

</tr>

<tr>

<td>密 码：</td>

<td><input type=*"password"* name=*"password"* id=*"password"* class=*"easyui-validatebox"* required=*"true"*/></td>

</tr>

<tr>

<td>姓 名：</td>

<td><input type=*"text"* name=*"name"* id=*"name"* class=*"easyui-validatebox"* required=*"true"*/></td>

</tr>

<tr>

<td>年 龄：</td>

<td><input type=*"age"* name=*"age"* id=*"age"* class=*"easyui-validatebox"* required=*"true"*/></td>

</tr>

<tr>

<td>性 别：</td>

<td><input type=*"text"* name=*"sex"* id=*"sex"* class=*"easyui-validatebox"* required=*"true"*/></td>

</tr>

<tr>

<td>系 别：</td>

<td><input type=*"text"* name=*"dept"* id=*"dept"* class=*"easyui-validatebox"* required=*"true"*/></td>

</tr>

<tr>

<td>电 话：</td>

<td><input type=*"text"* name=*"tel"* id=*"tel"* class=*"easyui-validatebox"* required=*"true"*/></td>

</tr>

<tr>

<td>Email：</td>

<td><input type=*"email"* name=*"email"* id=*"email"* class=*"easyui-validatebox"* required=*"true"*/></td>

</tr>

</table>

</form>

</div>

<div id=*"dlg-buttons"*>

<a href=*"javascript:saveStudent()"* class=*"easyui-linkbutton"* iconCls=*"icon-ok"*>保 存</a>

<a href=*"javascript:closeStudentDialog()"* class=*"easyui-linkbutton"* iconCls=*"icon-cancel"*>取 消</a>

</div>

</body>

</html>

第四章 运行与测试

后台系统登录界面



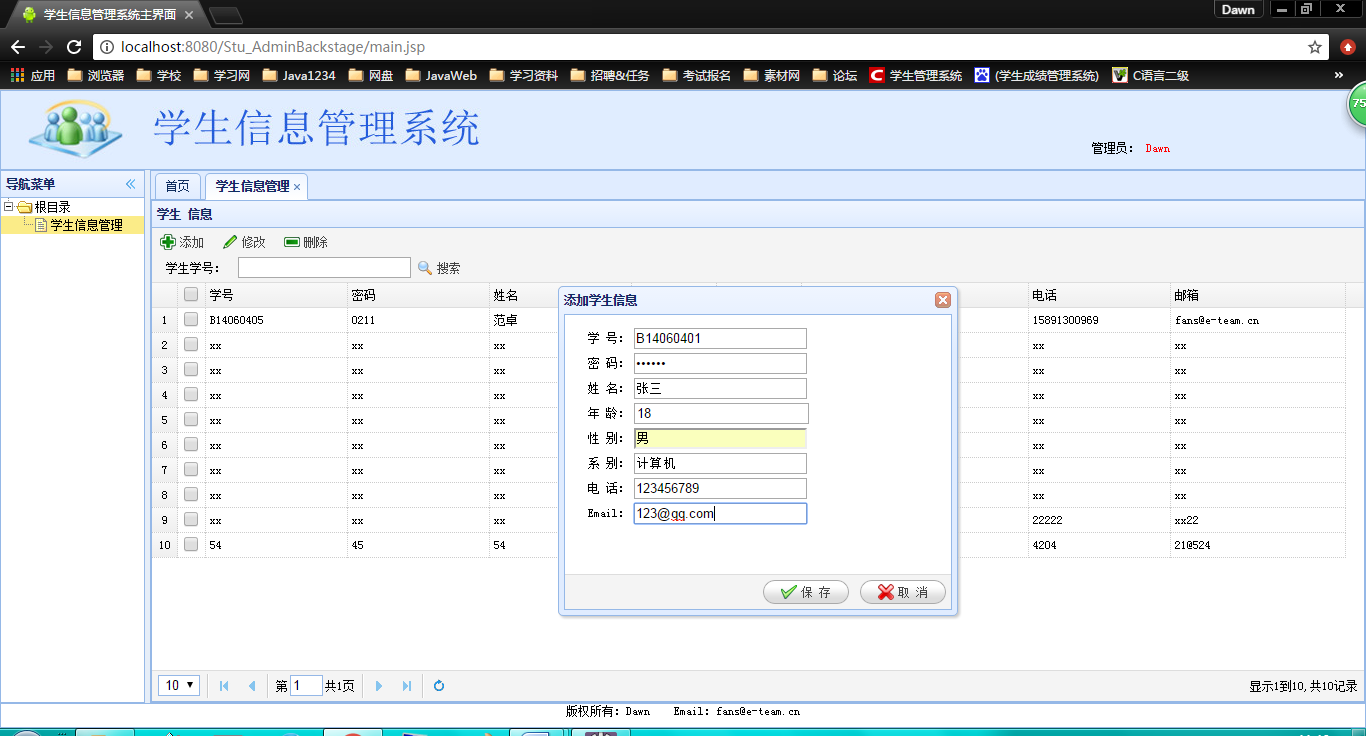
后台系统主界面



后台系统主要操作界面

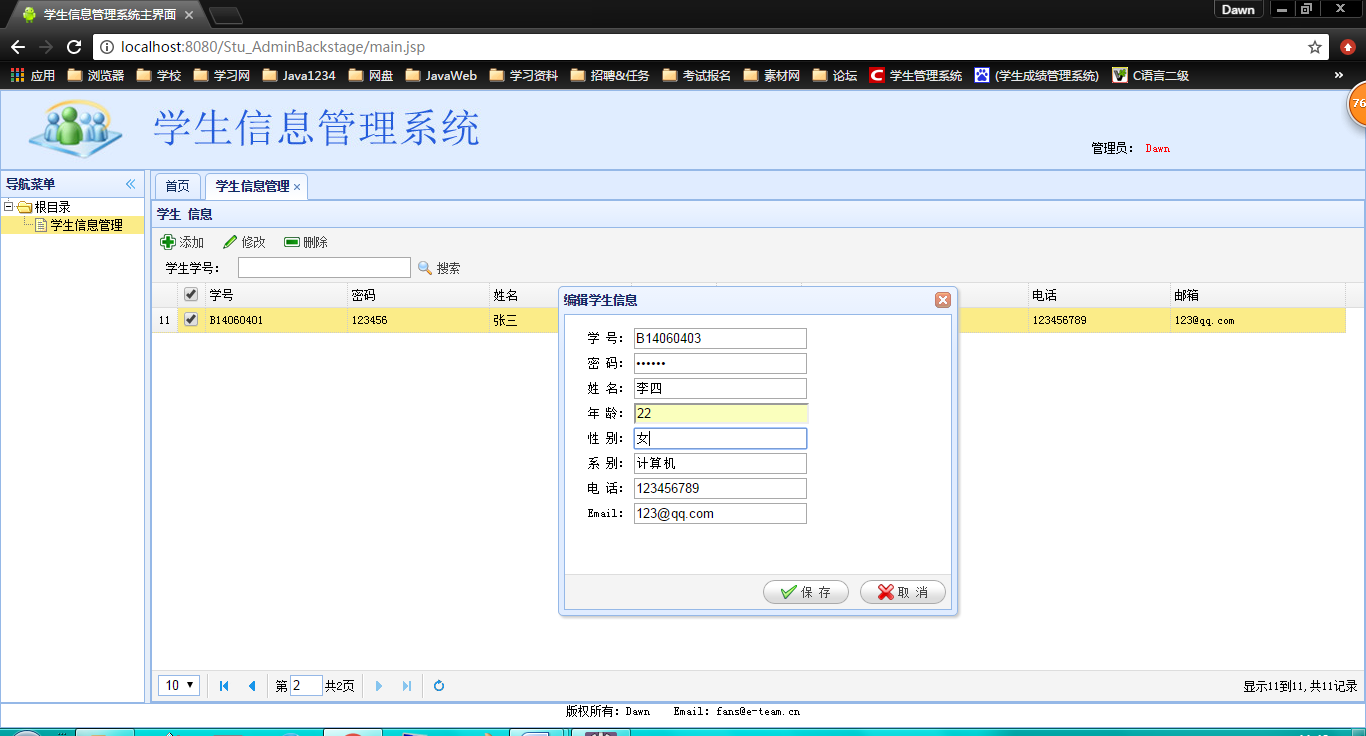


添加学生信息





修改学生信息



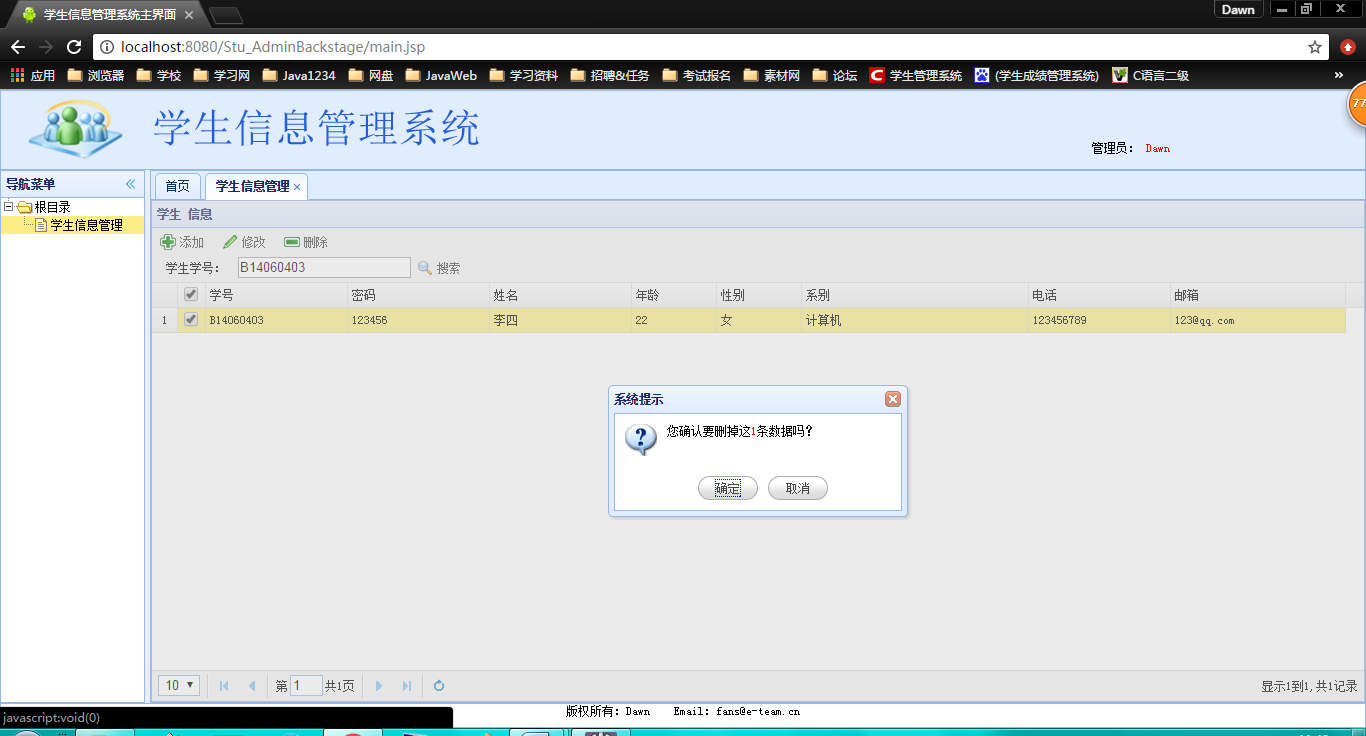


查询学生信息





删除学生信息





第五章 设计体会与小结

学生信息管理系统是针对学校人事处的大量业务处理工作而开发的管理软件，主要用于学校学生信息管理，总体任务是实现学生信息关系的系统化、科学化、规范化和自动化，其主要任务是用计算机对学生各种信息进行日常管理，如**查询、修改、增加、删除**，针对这些要求设计了学生信息管理系统。在本次实训中虽然时间有点短，但是针对本次开发我选择了MVC设计模式开发，因此，有许多框架都是固定的，都是可以拿之前开发的框架来填充的，这就从写代码上省掉许多时间。本次实训让我更加对MVC设计模式的了解和对Swing框架和数据库连接、操作的熟练使用，使框架开发更加熟悉。同时也提高了自己的设计能力，在此，我认为，开发前的设计思路是很重要的，只要有了思路，前面的准备工作完善后，再加上框架的实现，开发出一个良好的系统是不成问题的。由于时间原因，本次的前端用户登陆及用户功能没有完善，因此还不算是一个完整的系统。通过此次课设让我对JavaWeb开发网站有了很好的认识，同时也让我收获了许多。

第六章 参考文献

参考文献

[1] 邓良松 刘海岩 陆丽娜．软件工程[M]．西安：西安电子科技大学出版社，2000.11

[2] 闪四清.SQL Server2000实用教程[M]. 北京：人民出版社,2002.6

[3] 蔡剑 景楠著《Java Web 应用开发：J2EE 和Tomcat》清华大学出版社

[4] 侯晓强,徐春荣,勾海波.Java服务器编程实例[M].中国青年出版社,2004.53-97页.

[5] JSP 应用开发详解(含光盘) 电子工业出版社 2002.9

[6] 简兆权.一类智能管理信息系统的设计与实现[M].计算机应用研究,1998,15(2):43-45.

[7] 汪永明.基于B/S系统的在线考试系统.微机发展，2004(1)：32-35.

[8] 耿祥义，张跃平，JSP实用教程.清华大学出版社，2003.5.

[9] 蔡翠平等.Java程序设计[M].北京: 清华大学出版社,2002.10

[10] 肖刚.Java Web 服务器应用程序设计[M].西安：电子工业出版社，2001.6