**课题名称：Web网站的开发**

**班级：2017级信息二班**

**组长：刘翠翠**

**组员：吕淑红**

**组员：于海芳**

**组员：张颖**

**摘要**

随着计算机网络的普及应用，Web网站已成为人们获取信息、传递信息的重要阵地，Web网站开发成为当代信息系统开发的新热点。作者在研究Web应用系统原理、Web门户网站开发技术、Web网站安全技术的基础上,设计并实现系统，并投入实际运行。

本文总结了研究开发工作，详细阐述了软件系统需求分析、总体设计、详细设计、编码实现等开发过程。运用系统结构规划、安全控制策略技术与配置等系统安全实现过程与技术。

**一、实现用户注册登陆相关功能：**

1、编写UserBean 类

package com.liuyanzhao;

public class UserBean {

    private String name;

    private String password;

    private String email;

    public String getName() {

        return name;

    }

    public void setName(String name) {

        this.name = name;

    }

    public String getPassword() {

        return password;

    }

    public void setPassword(String password) {

        this.password = password;

    }

    public String getEmail() {

        return email;

    }

    public void setEmail(String email) {

        this.email = email;

    }

}

2、编写 RegisterFormBean 类

package com.liuyanzhao;

import java.util.HashMap;

import java.util.Map;

public class RegisterFormBean {

    private String name;

    private String password;

    private String password2;

    private String email;

    private Map<String, String> errors = new HashMap<String,String>();

    public String getName() {

        return name;

    }

    public void setName(String name) {

        this.name = name;

    }

    public String getPassword() {

        return password;

    }

    public void setPassword(String password) {

        this.password = password;

    }

    public String getPassword2() {

        return password2;

    }

    public void setPassword2(String password2) {

        this.password2 = password2;

    }

    public String getEmail() {

        return email;

    }

    public void setEmail(String email) {

        this.email = email;

    }

    public void setErrors(Map<String, String> errors) {

        this.errors = errors;

    }

    public boolean validate() {

        boolean flag = true;

        if(name == null || name.trim().equals("")) {

            errors.put("name", "请输入姓名");

            flag = false;

        }

        if(password == null || password.trim().equals("")) {

            errors.put("password", "请输入密码");

            flag = false;

        } else if(password.length() > 12 || password.length() < 6) {

            errors.put("password", "请输入 6-12 个字符");

            flag = false;

        }

        if(password != null && !password.equals(password2)) {

            errors.put("password2", "两次输入的密码不匹配");

            flag = false;

        }

        if(email == null || email.trim().equals("")) {

            errors.put("email", "请输入邮箱");

            flag = false;

        } else if(!email.matches("[a-zA-Z0-9\_-]+@[a-zA-Z0-9\_-]+(\\.[a-zA-Z0-9]+)+")) {

            errors.put("email", "邮箱格式错误");

            flag = false;

        }

        return flag;

    }

    public void setErrorMsg(String err,String errMsg) {

        if((err != null) && (errMsg != null)) {

            errors.put(err, errMsg);

        }

    }

    public Map<String, String> getErrors() {

        return errors;

    }

}

3、编写 DBUtil 类

package com.liuyanzhao;

import java.util.HashMap;

public class DBUtil {

    private static DBUtil instance = new DBUtil();

    private HashMap<String,UserBean> users = new HashMap<String,UserBean>();

    private DBUtil() {

        UserBean user1 = new UserBean();

        user1.setName("Tom");

        user1.setPassword("123456");

        user1.setEmail("tom@gmail.com");

        users.put("Tom", user1);

        UserBean user2 = new UserBean();

        user2.setName("jerry");

        user2.setPassword("666666");

        user2.setEmail("jerry@gmail.com");

        users.put("jerry", user2);

    }

    public static DBUtil getInstance() {

        return instance;

    }

    public UserBean getUser(String userName) {

        UserBean user = (UserBean)users.get(userName);

        return user;

    }

    public boolean insertUser(UserBean user) {

        if(user == null) {

            return false;

        }

        String userName = user.getName();

        if(users.get(userName) != null) {

            return false;

        }

        users.put(userName, user);

        return true;

    }

}

4、编写 ControllerServlet 类

package com.liuyanzhao;

import java.io.IOException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class ControllerServlet extends HttpServlet{

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

        this.doPost(request, response);

    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

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

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

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

        String password2 = request.getParameter("password2");

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

        RegisterFormBean formBean = new RegisterFormBean();

        formBean.setName(name);

        formBean.setPassword(password);

        formBean.setPassword2(password2);

        formBean.setEmail(email);

        if(!formBean.validate()) {

            request.setAttribute("formBean", formBean);

            request.getRequestDispatcher("/register.jsp").forward(request, response);

            return;

        }

        UserBean userBean = new UserBean();

        userBean.setName(name);

        userBean.setPassword(password);

        userBean.setEmail(email);

        boolean b = DBUtil.getInstance().insertUser(userBean);

        if(!b) {

        request.setAttribute("DBMes", "你注册的用户已存在");

            request.setAttribute("formBean", formBean);

      request.getRequestDispatcher("/register.jsp").forward(request, response);

            return;

        }

    response.getWriter().print("恭喜你注册成功,3秒钟自动跳转");

      request.getSession().setAttribute("userBean", userBean);

       response.setHeader("refresh","3;url=loginSuccess.jsp");

    }

}

5、在 web.xml 添加映射

<?xml version="1.0" encoding="utf-8"?>

<web-app xmlns="http://java.sun.com/xml/ns/javaee"

  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee

  http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"

  version="3.0"

  metadata-complete="true">

    <servlet>

        <servlet-name>ControllerServlet</servlet-name>    <servlet-class>com.liuyanzhao.ControllerServlet</servlet-class>

    </servlet>

    <servlet-mapping>

        <!-- 映射为 ControllerServlet -->

        <servlet-name>ControllerServlet</servlet-name>

        <url-pattern>/ControllerServlet</url-pattern>

    </servlet-mapping>

</web-app>

6、编写 register.jsp

<%@ page language="java" contentType="text/html; charset=UTF-8"

    pageEncoding="UTF-8"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

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

    <title>用户注册</title>

    <style>

        h3 {

            margin-left:100px;

        }

        #outer {

            width:750px;

        }

        span {

            color:#ff0000;

        }

        div {

            height:20px;

            margin-bottom:10px;

        }

        .ch {

            width:80px;

            text-align:right;

            float:left;

            padding-right:5px;

        }

        .ip {

            width:500px;

            float:left;

        }

        .ip > input {

            margin-right:20px;

        }

        #bt {

            margin-left:50px;

        }

        #bt > input {

            margin-right:30px;

        }

    </style>

</head>

<body>

    <form action="ControllerServlet" method="post">

        <h3>用户注册</h3>

        <div id="outer">

            <div>

                <div class="ch">姓名:</div>

                <div class="ip">

   <input type="text" name="name" value="${formBean.name}" />

            <span>${formBean.errors.name}${DBMes}</span>

                </div>

            </div>

            <div>

                <div class="ch">密码:</div>

                <div class="ip">

              <input type="text" name="password" />

             <span>${formBean.errors.password} </span>

                </div>

            </div>

            <div>

                <div class="ch">确认密码:</div>

                <div class="ip">

             <input type="text" name="password2" />

             <span>${formBean.errors.password2} </span>

                </div>

            </div>

            <div>

                <div class="ch">邮箱:</div>

                <div class="ip">

  <input type="text" name="email" value="${formBean.email}" />

                    <span>${formBean.errors.email}</span>

                </div>

            </div>

            <div id="bt">

             <input type="reset" value="重置" />

             <input type="submit" value="注册" />

            </div>

        </div>

    </form>

</body>

</html>

7、编写 loginSuccess.jsp 类

<%@ page language="java" contentType="text/html; charset=UTF-8"

    pageEncoding="UTF-8"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

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

<title>登录成功</title>

</head>

<body>

    <%

        if(session.getAttribute("userBean") == null) {

    %>

    <jsp:forward page="register.jsp" />

    <%

            return;

        }

    %>

    <jsp:useBean id="userBean" class="com.liuyanzhao.UserBean" scope="session" />

    <div id="main">

        <div id="welcome">恭喜你，登录成功</div>

        <hr />

        <h3>你的信息</h3>

        <div>

            <ul>

               <li>您的姓名：${userBean.name} </li>

               <li>您的邮箱：${userBean.email} </li>

            </ul>

        </div>

    </div>

</body>

</html>

**二、运行程序**

启动 Tomcat 服务器

1、在地址栏输入 http://localhost:8080/JspModel2Demo/register.jsp

可以看到如下图所示

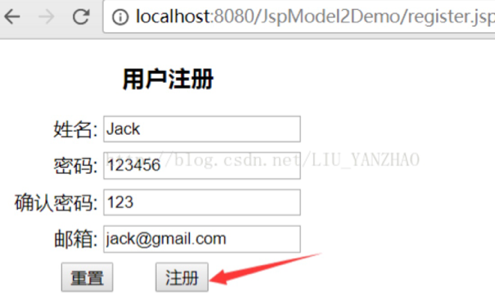


 2测试未填





测试两次密码不一致



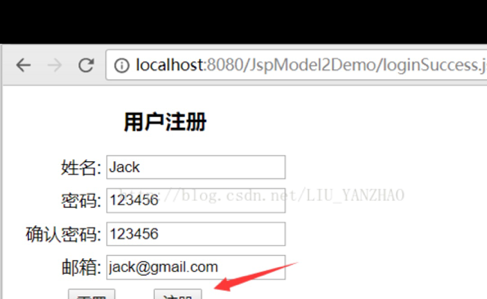


测试账号存在



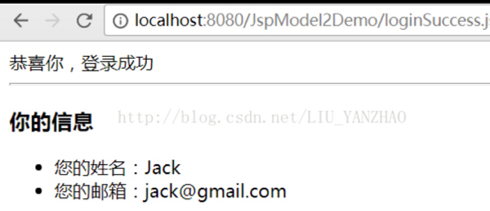


再次输入正确密码



点击注册按钮





实现修改密码

Jsp页面的html代码

<a class="easyui-linkbutton" data-options="iconCls:'icon-cut',plain:true" οnclick="editPsw()">修改密码</a>

修改密码的dialog弹出框

<%-- 修改密码dialog--%>

<div id="xgmmdlg" class="easyui-dialog" style="width:600px;height:400px;padding:10px 20px" closed="true" buttons="#dlg-buttons">

<f:form id="paswform" commandName="userPartner" action="userPartner/editPassword.do" method="post" >

<div class="fitem">

<label>新密码:</label>

<input style="width: 400px;" id="newLoginPwd" class="easyui-passwordbox" name="newLoginPwd" missingMessage="密码不能为空！" required="true"/>

</div>

<div class="fitem">

<label>确认密码:</label>

<input style="width: 400px;" id="ConfirmPassword" class="easyui-passwordbox" name="ConfirmPassword" missingMessage="密码不能为空！" required="true"/>

</div>

<div class="fitem">

<label>验证账号密码:</label>

<input style="width: 400px;" id="userLoginPwds" class="easyui-passwordbox" name="userLoginPwds" missingMessage="密码不能为空！" required="true"/>

</div>

</f:form>

<div class="fitem">

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

<tr>

<td style="text-align:center">

<a href="javascript:void(0)" class="easyui-linkbutton c6" iconCls="icon-ok" οnclick="editPassword()" style="width:90px">修改</a>

<a href="javascript:void(0)" class="easyui-linkbutton" iconCls="icon-cancel" οnclick="javascript:$('#xgmmdlg').dialog('close')" style="width:90px">关闭</a>

</td>

</tr>

</table>

</div>

</div>

选出一条记录打开新的dialog

//修改密码

function editPsw() {

var selected=$('#dg').datagrid('getSelections');

if(selected.length == 1){

var ids = [];

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

ids.push(selected[i].id);

}

$('#xgmmdlg').dialog('open').dialog('center').dialog('setTitle', '修改密码');

$('#fm').form('clear');

}else{

$.messager.alert('错误消息','您未选中或只能选中单条','error');

}

}

提交数据访问后台

//修改密码

function editPassword() {

var row = $("#dg").datagrid("getSelected");

var newPawd = $('#newLoginPwd').val();

var ConfirmPassword = $('#ConfirmPassword').val();

var userLoginPwds = $('#userLoginPwds').val();

if(newPawd == ConfirmPassword){

$.ajax({

url:'xxxxxxx.do?id='+row.id,

type:"POST",

data:{newPawd:newPawd,userLoginPwds:userLoginPwds},

success:function (data) {

if(data.code=='1'){

$.messager.alert('提示消息','修改失败,初始密码可能正确！','show');

}else {

$.messager.alert('提示消息','修改成功','show');

$('#paswform').form('clear');

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

$('#xgmmdlg').dialog('close');

}

}

})

}else{

$.messager.alert('错误消息','两次输入密码不匹配','error');

}

}

**三、修改密码Controller**

@RequestMapping("/editPassword")

@ResponseBody

public Map<String, Object> editPassword(UserPartner userPartner, String newPawd, String userLoginPwds, HttpServletRequest request) {

Map<String, Object> map = new HashMap<String, Object>();

try {

//根据id去查询

UserPartner loadPassWord = userPartnerService.load(userPartner);

//将输入的原始密码加密

String initPassWords = MD5Util.strToMD5(userLoginPwds);

if (!initPassWords.equals(loadPassWord.getUserLoginPwd())) {

map.put(CardGodConstant.MAP\_KEY\_MSG, CardGodConstant.MAP\_VALUE\_PARAMETER\_ERROR);

map.put(CardGodConstant.MAP\_KEY\_CODE, CardGodConstant.STR\_VALUE\_1);

return map;

}

if (initPassWords.equals(loadPassWord.getUserLoginPwd())) {

userPartner.setUserLoginPwd(MD5Util.strToMD5(newPawd));

userPartnerService.edit(userPartner, request);

map.put(CardGodConstant.MAP\_KEY\_MSG, CardGodConstant.MAP\_VALUE\_SUCCESS);

map.put(CardGodConstant.MAP\_KEY\_CODE, CardGodConstant.STR\_VALUE\_0);

return map;

} } catch (Exception e) {

logger.error(CardGodConstant.ERROR\_MSG + e);

e.printStackTrace();

map.put(CardGodConstant.MAP\_KEY\_MSG, CardGodConstant.MAP\_VALUE\_FAILURE);

map.put(CardGodConstant.MAP\_KEY\_CODE, CardGodConstant.STR\_VALUE\_1);

return map;

}

return map;

}

**四、数据库选择**

数据库系统选择是系统实现的关键之一。目前比较流行的数据库系统分为关系数据库系统和对象数据库系统。关系数据库系统分为两类：一类是小型的关系数据库系统；另一类是大型的关系数据库系统。

**五、数据库设计**

数据库是系统的数据中心，支撑着整个系统的运行。用户需要的所有信息都是应用程序通过中间件调用数据库数据而得到的。数据库的设计、开发中占有很重要的比重，设计良好的数据库不但可以使系统以最优的性能运行。而且能够使系统的开发变得简单，从而减轻开发人员的负担，从而缩短开发周期。

**六、数据库表设计图**

基于网络结构和访问控制设计，可以设计具体的数据库表。涉及用户管理子系统的数据库表如图所示：

