****

软件开发实训

说明书

题 目： **留 言 板**

学 院： 计算机与信息安全学院

专 业： 计算机科学与技术

学生姓名： 庞 俊 龙

学 号： 1400310223

指导教师： 梁 海

2017年12月25日

目录

[一、 详细设计 1](#_Toc502044044)

[1.1概述 1](#_Toc502044045)

[1.2系统功能 1](#_Toc502044046)

[1.3系统功能模块图 2](#_Toc502044047)

[1.4数据库设计 2](#_Toc502044048)

[二、 具体实现 3](#_Toc502044049)

[2.1使用技术 3](#_Toc502044050)

[2.2注册功能 3](#_Toc502044051)

[2.3发送留言 5](#_Toc502044052)

[2.4管理员审核 7](#_Toc502044053)

[三、 功能测试 8](#_Toc502044054)

[四、 总结 11](#_Toc502044055)

1. 详细设计

1.1概述

该留言板实现了发留言和管理员审核留言的功能。当用户或者游客发送留言后，并不能马上显示在首页，需要管理员审核通过之后才可以在首页显示。为此我们需要两个实体，分别为用户和留言信息实体。

用户实体

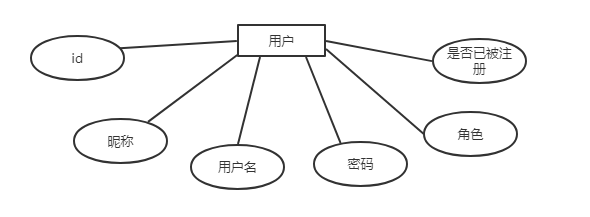


图1

留言信息实体

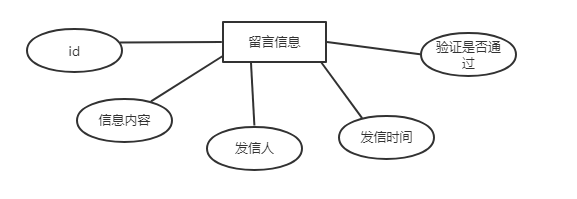


图2

## 1.2系统功能

1）打开页面后自动加载留言信息，并且按照时间降序排序显示；

2）可以不登陆或者登陆发表留言；

3）管理员有审核和删除留言的权利；

4）用户的一切操作都不需要跳转页面即可实现

## 1.3系统功能模块图

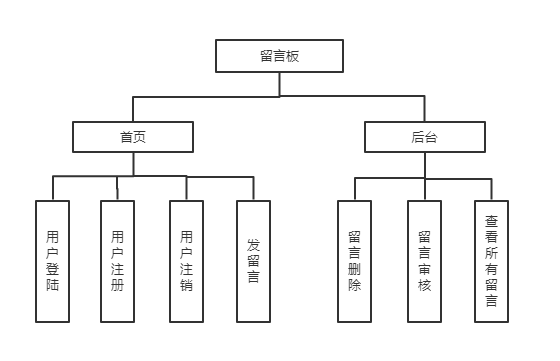


图3

## 1.4数据库设计

数据库采用mysql，该数据库为轻量级的数据库，方便且功能较为强大。

用户实体表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名字 | 类型 | 长度 | 可否为空 | 是否主键 | 默认值 | 属性描述 |
| Id | Int | 11 | 否 | 是 |  | 用户id |
| Nickname | Varchar | 255 | 是 | 否 |  | 用户昵称 |
| Username | Varchar | 255 | 是 | 否 |  | 用户名 |
| Password | Varchar | 255 | 是 | 否 |  | 用户密码 |
| Roll | Int | 11 | 是 | 否 |  | 角色  0为普通用户/1为管理员 |
| isRegis | Int | 11 | 是 | 否 |  | 是否已被注册 |

表1

isRegis用来返回给前端页面时让js来判断用户是否成功注册。

留言信息实体表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名字 | 类型 | 长度 | 可否为空 | 是否主键 | 默认值 | 属性描述 |
| Id | Int | 11 | 否 | 是 |  | 留言id |
| messageDetails | Varchar | 255 | 是 | 否 |  | 留言内容 |
| createPerson | Varchar | 255 | 是 | 否 |  | 留言发送人昵称 |
| createTime | Varchar | 255 | 是 | 否 |  | 发送时间 |
| isVal | Int | 11 | 是 | 否 |  | 是否通过验证 |

表2

1. 具体实现

2.1使用技术

这次实现的功能较为简单，但是使用了restful风格的框架jersey。

Jersey以注释的形式来获取前端传送的参数，并且决定了返回给前端的数据类型，至于前端如何接受如何显示，这与后端的接口无关，从而实现了前后端分离的概念。数据库框架使用了hibernate，该框架能非常方便的进行数据的增删改查，大大减少了开发难度。

2.2注册功能

流程图如图

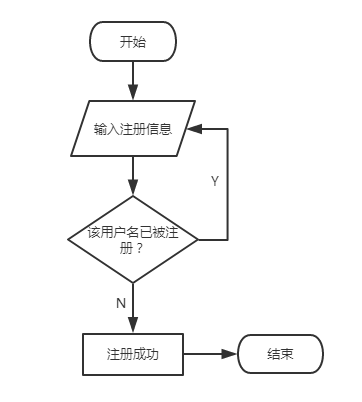


图4

注册请求发送后不会刷新页面，通过前端的js方法来提示你是否成功注册，如图所示

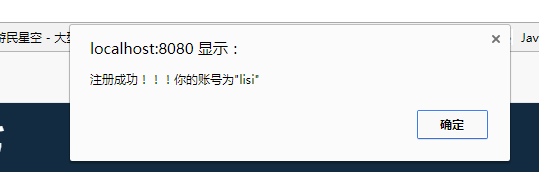


图5

所以不刷新的请求和返回值这里我们采用ajax异步技术，js方法代码如下：

function regis(){

if($("#pa2ssword").val()==$("#rePassword").val()){

$.ajax({

url:"rest/user/regis",

type:"post",

data:$("#regisForm").serialize(),

success:function(data){

if(data.isRegis == 1){

alert("你的账号已被注册！！");

}else{

alert("注册成功！！！你的账号为"+JSON.stringify(data.username));

}

}

});

}else{

alert("两次密码输入不一致！！！");

}

}

返回值通过json对象中的isregis属性来判断是否已被注册。后台代码如下所示：

@POST

@Path("/regis")

@Produces(MediaType.APPLICATION\_JSON)

public User regis(@FormParam("nickname") String nickname,

@FormParam("username") String username,

@FormParam("password") String password){

User user = new User();

user.setNickname(nickname);

user.setUsername(username);

user.setPassword(password);

user.setRoll(0);

User oUser = us.getUser(user.getUsername());

if(oUser != null){

return oUser;

}else{

us.regis(user);

user.setIsRegis(0);

return user;

}

}

登陆功能与注册大同小异，这里不做赘述

2.3发送留言

流程图如图

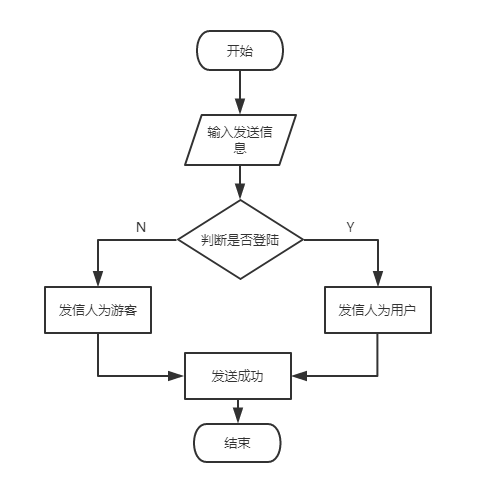


图6

发送同样通过ajax来发送请求，代码如下：

function send(){

if($("#messageArea").val() == ""){

alert("不能为空！！");

}else{

$.ajax({

url:"rest/Message/showMessage",

type:"post",

data:$("#MessageDetails").serialize(),

success:function(data){

//alert(JSON.stringify(data));

$("#showList").empty();

var html = $("#list").render(data.message);

$("#showList").append(html);

}

});

}

}

后台接受代码如下：

@POST

@Path("/showMessage")

@Consumes(MediaType.APPLICATION\_FORM\_URLENCODED)

@Produces(MediaType.APPLICATION\_JSON)

public List<Message> showMessage(@FormParam("message") String message,

@FormParam("sendMessageUserName") String username){

System.out.println(message);

if(username.equals("")){

username = "游客";

}

Date day=new Date();

SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

System.out.println(df.format(day));

Message messageDetails = new Message();

messageDetails.setMessageDetails(message);

messageDetails.setCreatePerson(username);

messageDetails.setIsVal(0);

System.out.println(messageDetails.toString());

messageDetails.setCreateTime(df.format(day));;

ms.sendMessage(messageDetails);

List<Message> list = ms.getPaMessage();

return list;

}

这里发送之后就直接刷新了一次首页的留言信息，实际上发送的留言还要通过管理员的审核通过之后才可以发布到首页。

2.4管理员审核

首先查询出来所有的未通过审核的留言，这里的遍历留言使用了js的一个模板引擎jsrender，具体js代码如下：

function showNonMessage(){

$.ajax({

url:"../rest/Message/showNonMessage",

type:"post",

success:function(data){

$("#collapseTwo").empty();

var html = $("#AllNonlist").render(data.message);

$("#collapseTwo").append(html);

}

})

}

然后模板处的代码如下：

<script id="AllNonlist" type="text/x-jsrender">

<div class="panel-body">

<table class="table table-hover">

<tbody>

<tr>

<td style="width:40%;">{{:messageDetails}}</td>

<td style="width:30%;">{{:createPerson}}</td>

<td><a href="#" style="color: #122B40;" onclick="javascript:PassMessage({{:messageId}});"><span

class=" glyphicon glyphicon-ok" ></span> 通过</a></td>

</tr>

</tbody>

</div>

</script>

大概原理就是，把json数据取出后赋值给该脚本代码，该代码模板会根据json数据的长度创建如长度一样的个数模板内的内容并且为此赋值。

然后管理员的通过按钮也是通过ajax来实现，将该条信息的isval设为1，具体代码如下：

@POST

@Path("/PassMessage")

@Produces(MediaType.APPLICATION\_JSON)

public List<Message> PassMessage(@FormParam(value = "a1") Integer messageId){

System.out.println(messageId);

ms.PassById(messageId);

List<Message> list = ms.getNonMessage();

return list;

}

Dao层代码如下：

public void PassById(Integer messageId) {

Session session = HibernateUtils.getCurrentSession();

Message message = session.load(Message.class, messageId);

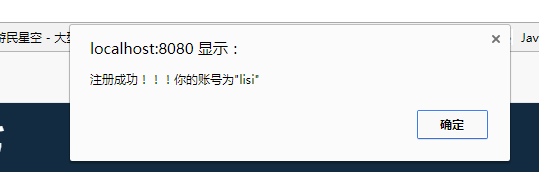
message.setIsVal(1);

session.update(message);

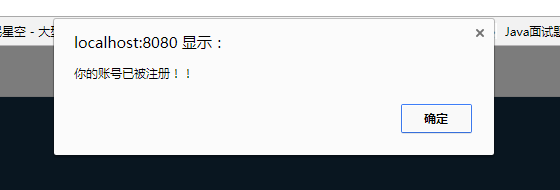
}

1. 功能测试

注册成功

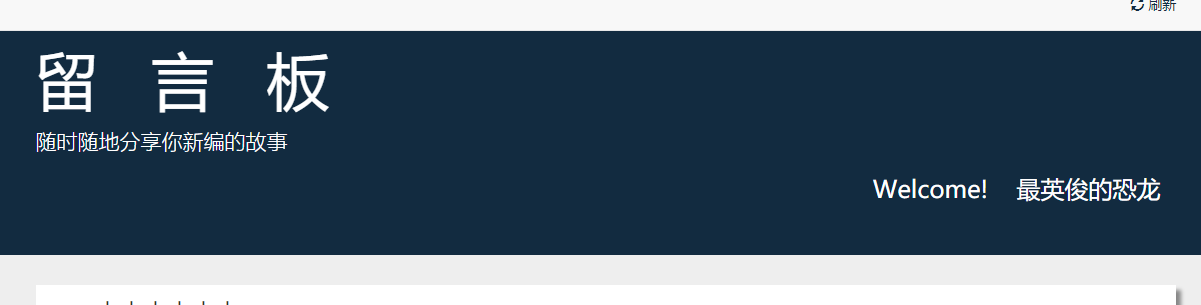


注册失败，用户名已被注册时

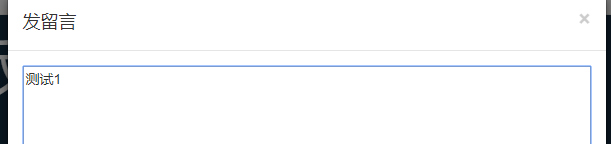


登陆成功后

会在留言板右上角显示该用户的昵称



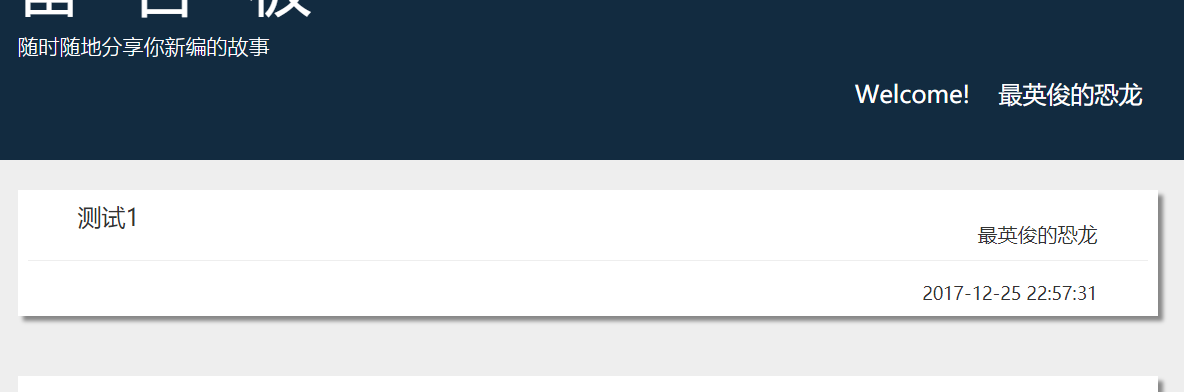
发送信息



发送后后台管理中未通过审核的留言处将显示该条信息



然后管理员点击通过按钮，该条信息便会发布出去



1. 总结

通过这次课程设计，虽然简单，但是也学习到了不一样的东西，比如git开源代码托管平台，和jersey框架，并且对restful也有一定程度的了解，和加强了hibernate的使用熟练度。

通过这次课程设计使我们都更加懂得并亲身体会到了理论与实际相结合的重要性，只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，从实践中得出结论，才能真正为社会服务，从而提高自己的实际动手能力和独立思考的能力。在设计的过程中遇到很多问题，可以说是困难重重，并且在设计的过程中发现了自己的很多不足之处，发现自己对之前所学过的知识理解得不够深刻，掌握得不够牢固，有待加强。  
虽然我们的课程设计在规定的时间内按时完成了任务，但是，由于知识和经验的欠缺，整个系统的开发功能还不算太完善，在设计的过程中也遇到了很多困难，比如由于早期的需求分析不充分，给后来系统的具体实现带来了很大的困难，所以中间做了一部分的无用功，但通过我们的不懈努力和老师、同学的帮助，设计才得以圆满按时的完成，所以仅凭个人的力量想设计出一个完美的系统几乎是不可能的。

老师为我们安排此次课程设计的基本目的，在于通过理论与实际的结合、人与人的沟通，进一步提高思想觉悟，尤其是观察、分析和解决问题的实际工作能力，以便培养成能够主动适应社会主义现代化建设需要的高素质人才。进一步激发了我们对专业知识学习的兴趣，并能够结合实际存在的问题在专业领域内进行更深入的学习。