

# JavaWeb大作业报告

王鹏 14030130101

## 1. 题目要求

实现一个web微博系统，须具有以下功能：

1. 注册、登录和注销功能。
2. 布信息，如：文字、图片。
3. 用户发言、留言功能。

## 2. 系统功能结构

此系统的主要模块有三个，分别是：我的微博模块、最新动态模块、留言板模块。在我的微博模块中有查看我的微博和查看我的好友功能，还有更多设置功能；在最新动态模块中可以发表微博和查看历史微博内容；在留言板模块中可以发表留言和查看历史留言信息等；

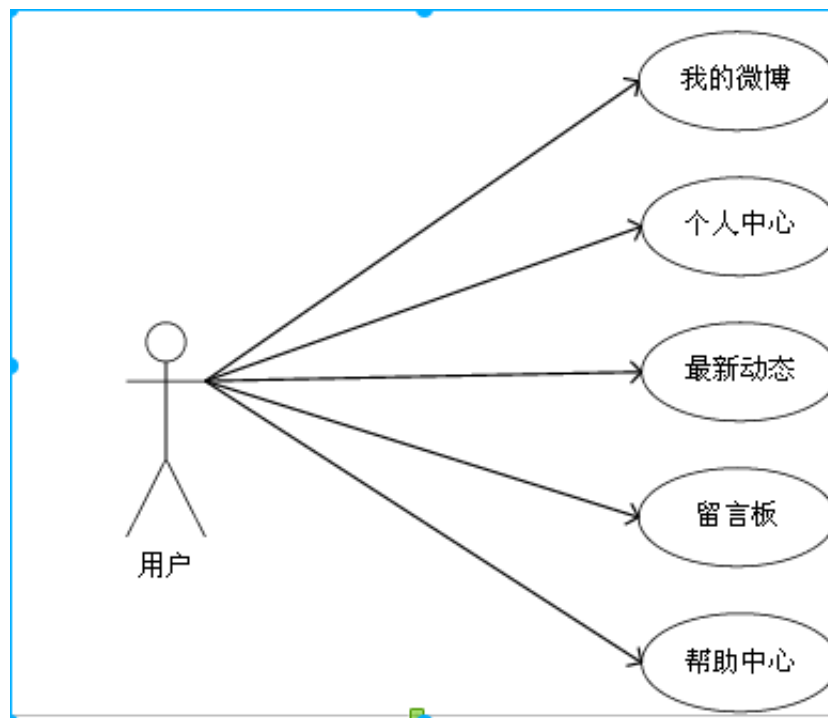


Figure 1. 系统总图

### 2.1. 我的微博模块功能分析

在我的微博模块中有查看我的微博和查看我的好友功能，还有更多设置功能。在这个模块中用户可以根据自己的需求选择相应的功能进行使用，具体如下图：

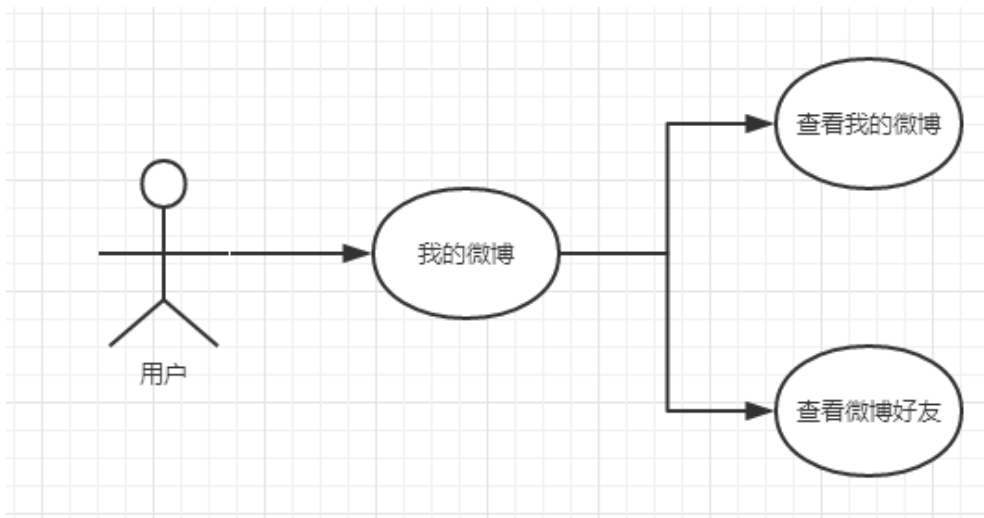


Figure 2. 我的微博

在这里，查看我的微博功能可以跳转到微博信息界面，用户可以查看历史的微博信息并对其进行操作。查看我的好友功能可以跳转到好友信息界面，可以查看好友的详细信息并且对好友进行相应的操作。更多设置功能是跳转到个人中心界面，可以对个人信息等信息进行操作。

## 2.2. 个人中心模块

在个人中心模块中可以对好友进行管理，还可以完善用户的基本信息和密码修改等操作。

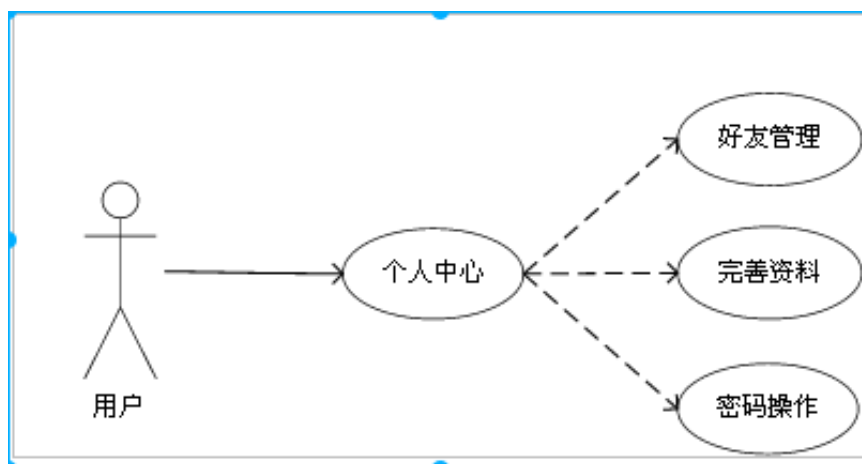


Figure 3. 个人中心

## 2.3. 最新动态模块

最新动态模块。用户可以在最新动态模块中发表微博信息，还可以查看历史微博信息并对其进行操作。

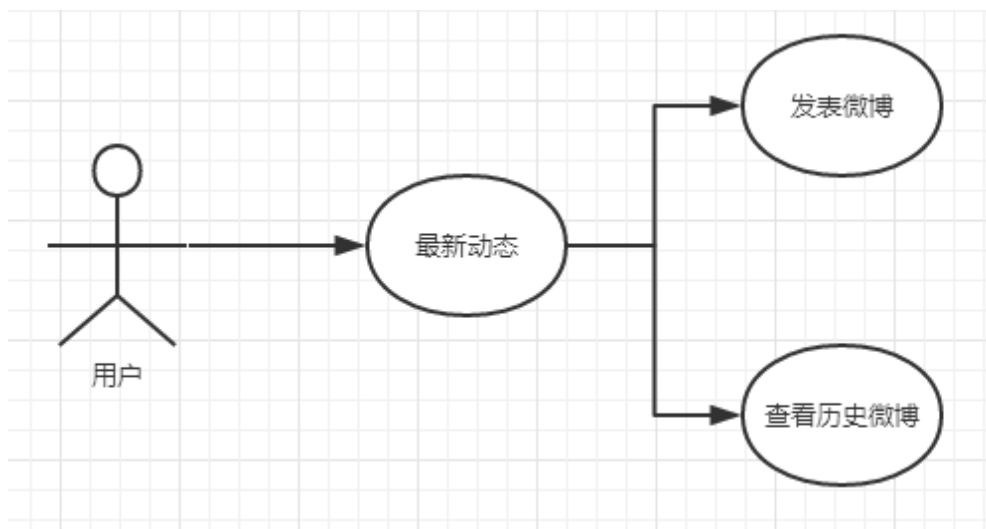


Figure 4. 最新动态

## 2.4. 留言板模块

留言板模块。用户可以在留言板模块中发表留言信息，还可以查看历史留言信息并对其进行操作。

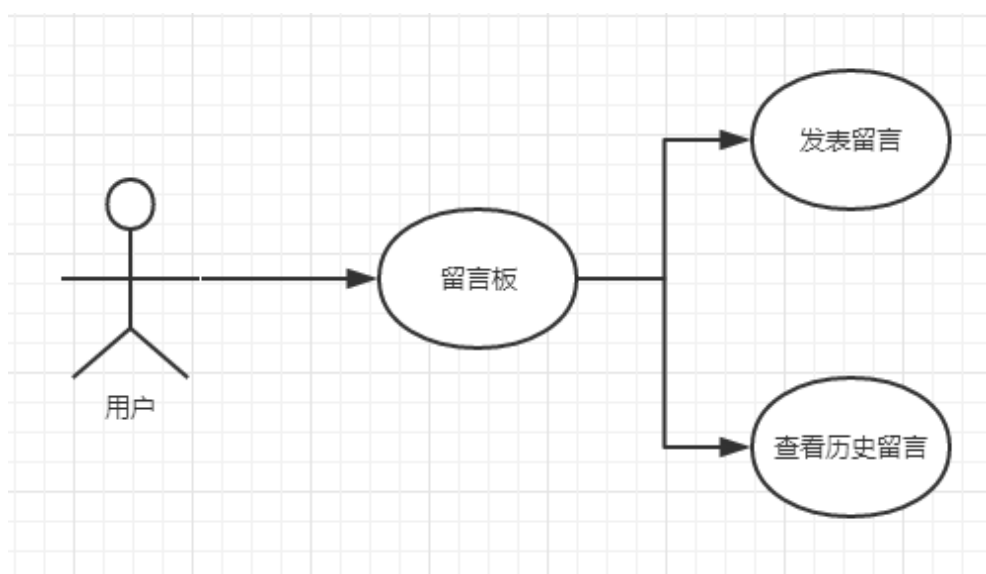


Figure 5. 留言板

## 2.5. 帮助中心模块

帮助中心这个模块中主要是针对一些常见问题进行解答。例如对于初用者的提问最多莫过于“如何使用微博系统”、“如何发表微博信息”等，还有就是主要的一些微博规则等

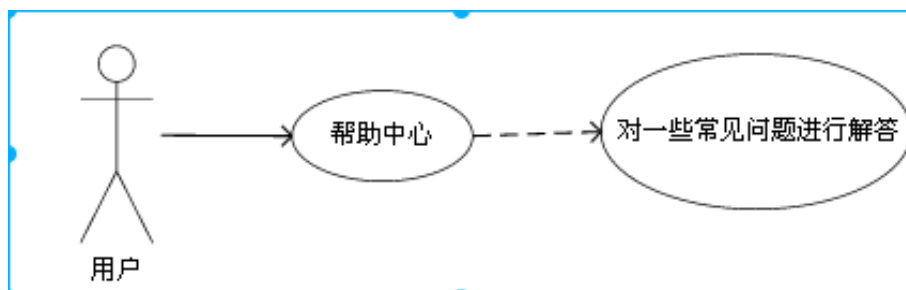


Figure 6. 帮助中心

### 3. 系统操作过程

1. 首先，用户会进入到系统的登录界面，此时要求用户输入账号和密码，只有存在并且正确的账号和密码才能通过验证并进入到系统使用页面。
2. 如果用户没有账号，可以直接注册，注册时只需要填写用户名和密码就好，进入系统使用页面后继续补充好其他信息即可。
3. 登录完成后进入系统操作页面，用户可以根据自己的需求选择我的微博、个人中心、最新动态、留言板、帮助中心功能，选择之后可以选择进一步的子功能。
4. 选择相应子功能后浏览器中会显示相应的操作供用户操作。
5. 使用完毕后点击退出，此时会弹出对话框询问用户是否真的退出，点击“是”则退出系统，直接返回到系统登录时的界面；点击“否”则留在刚刚的界面。

### 4. 系统总体结构图

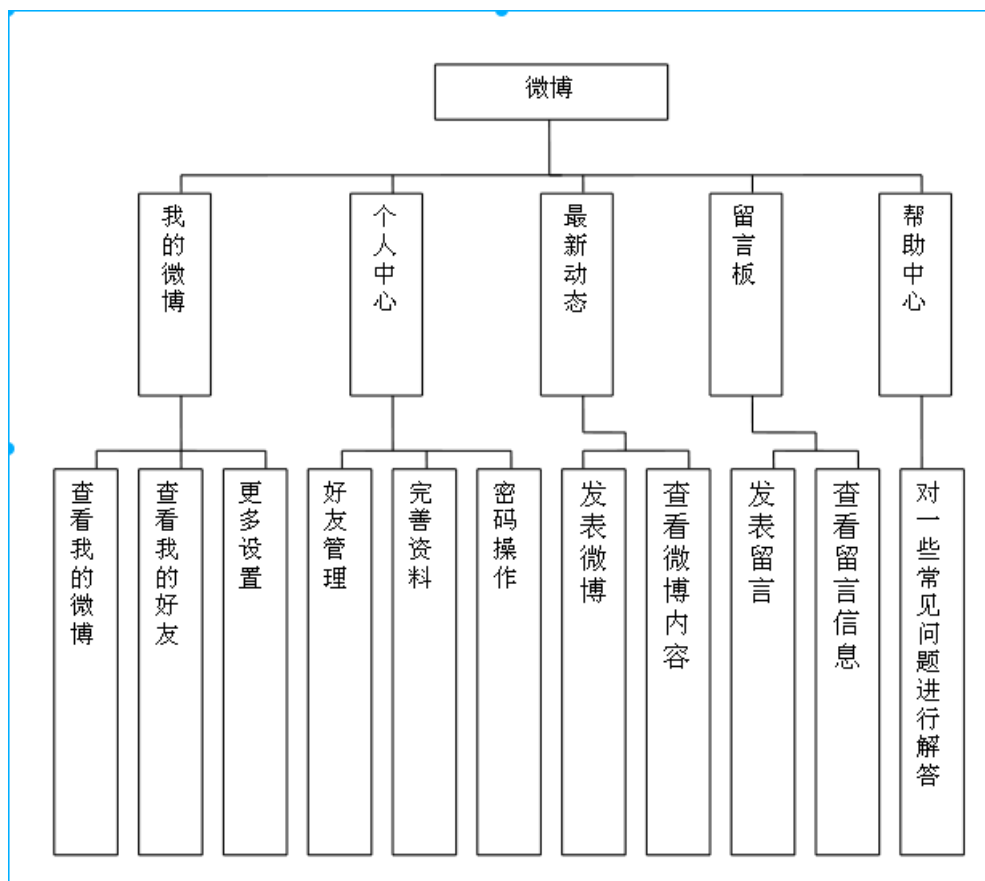


Figure 7. 系统总体结构图

## 5. 数据库设计

### 5.1. 数据库选择

本系统使用的数据库是关系型数据库MySQL。MySQL是一个小型关系型数据库管理系统，开发者为瑞典MySQL AB公司。Sun公司于2008年1月16号收购了瑞典MySQL AB公司。而在2009年,Oracle又收购了SUN。当时，没有任何人对MySQL的前途抱乐观的态度。但是目前，MySQL被广泛地应用在Internet上的中小型网站中。由于其速度快、体积小、总体拥有成本低，尤其MySQL是开源的这一特点，所以许多中小型网站为了降低网站总体拥有成本而选择了MySQL作为网站数据库。

### 5.2. 数据库表描述

本系统设计的所有数据表必须满足三范式要求，即：

1. 表中的每一列都是不可再分割的原子值。
2. 表中的所有非主属性值必须完全函数依赖于主属性值。
3. 表中不存在传递函数依赖，即非主属性值不能函数依赖于非主属性值。

#### 5.2.1. 用户信息表(Cust)

该表主要完成用户登录时信息的验证功能和存储用户的个人信息。在此表中定义了10个字

段来储存信息。

标识符	数据类型	长度	非空	描述	主键	外键	说明
<u>cid</u>	<u>varchar</u>	10	Y	用户编号	Y		自增
<u>cname</u>	<u>varchar</u>	20	Y	用户姓名			
<u>cpwd</u>	<u>varchar</u>	20	Y	用户密码			
<u>csex</u>	<u>varchar</u>	4	N	用户性别			
<u>cage</u>	<u>int</u>	11	N	用户年龄			
<u>ctel</u>	<u>int</u>	10	N	用户联系方式			
<u>caddress</u>	<u>varchar</u>	40	N	用户地址			
<u>did</u>	<u>varchar</u>	10	Y	动态 ID		Y	与动态表外键关联
<u>lid</u>	<u>varchar</u>	10	Y	留言 ID		Y	与留言表外键关联
<u>hid</u>	<u>varchar</u>	10	Y	好友编号		Y	与好友表外键关联

5.2.2. 动态信息

该表主要保存用户添加的动态信息，主要包括动态信息的ID、添加动态信息的时间还有动态信息的内容。 在此表中定义了以下4个属性来存储相关信息。其中用户编号是外键，对应于用户表的用户编号。 动态信息表(Dongtai)

标识符	数据类型	长度	非空	描述	主键	外键	说明
<u>did</u>	<u>varchar</u>	10	Y	动态 ID	Y		自增
<u>ddate</u>	<u>date</u>	20	Y	时间			
<u>dcontent</u>	<u>varchar</u>	200	Y	内容			
<u>cid</u>	<u>varchar</u>	10	Y	用户编号		Y	与用户表外键关联

5.2.3. 留言信息表(Liuyan)

该表主要保存用户添加的留言信息，主要包括留言信息的ID、添加留言信息的时间以及留言信息的内容。 在此表中定义了以下4个属性来存储相关信息

标识符	数据类型	长度	非空	描述	主键	外键	说明
<u>lid</u>	<u>varchar</u>	10	Y	留言 ID	Y		自增
<u>ldate</u>	<u>date</u>	20	Y	时间			
<u>lcontent</u>	<u>varchar</u>	200	Y	内容			
<u>cid</u>	<u>varchar</u>	10	Y	用户编号		Y	与用户表外键关联

5.2.4. 好友信息表(Haoyou)

该表主要用来保存好友信息，主要包括。在此表中定义了5个属性来存储相关的信息。其中管理员ID是外键，对应于管理员表的管理员ID

标识符	数据类型	长度	非空	描述	主键	外键	说明
<u>select table row</u>	<u>varchar</u>	10	Y	好友编号	Y		自增
<u>xcid</u>	<u>varchar</u>	10	Y	好友 x 的 ID			
<u>ycid</u>	<u>varchar</u>	10	Y	好友 y 的 ID			
<u>xcname</u>	<u>varchar</u>	20	Y	好友 x 姓名			
<u>ycname</u>	<u>varchar</u>	20	Y	好友 y 姓名			
<u>cid</u>	<u>varchar</u>	10	Y	用户编号		Y	与用户表外键关联

## 6. 系统实现

### 6.1. 登录注册

微博系统的登录功能的设计采用用户的编号和密码两条信息对用户的合法性进行验证，要登录的用户需输入编号和密码，单击“登录”按钮进行登录，这时后台会从登录页面中读取到用户填写的编号和密码与数据库中的数据进行比较，只有用户编号存在并且对应的密码正确才会登录成功跳转到主页面，否则则会显示相应的错误信息如“用户不存在”或“密码错误”等。注册功能。首先用户会进入到登录页面，但是没有账号所以可以点击注册按钮，此时浏览器会跳转到注册页面，用户可以填写自己的信息，后台会从页面获取到用户所填写的信息进行验证是否合法，如果不正确就需要用户重新填写不合法的信息，如果正确就可以注册，注册成功就会显示成功信息，注册失败也会提示相应的错误信息。

#### 6.1.1. 功能流程图

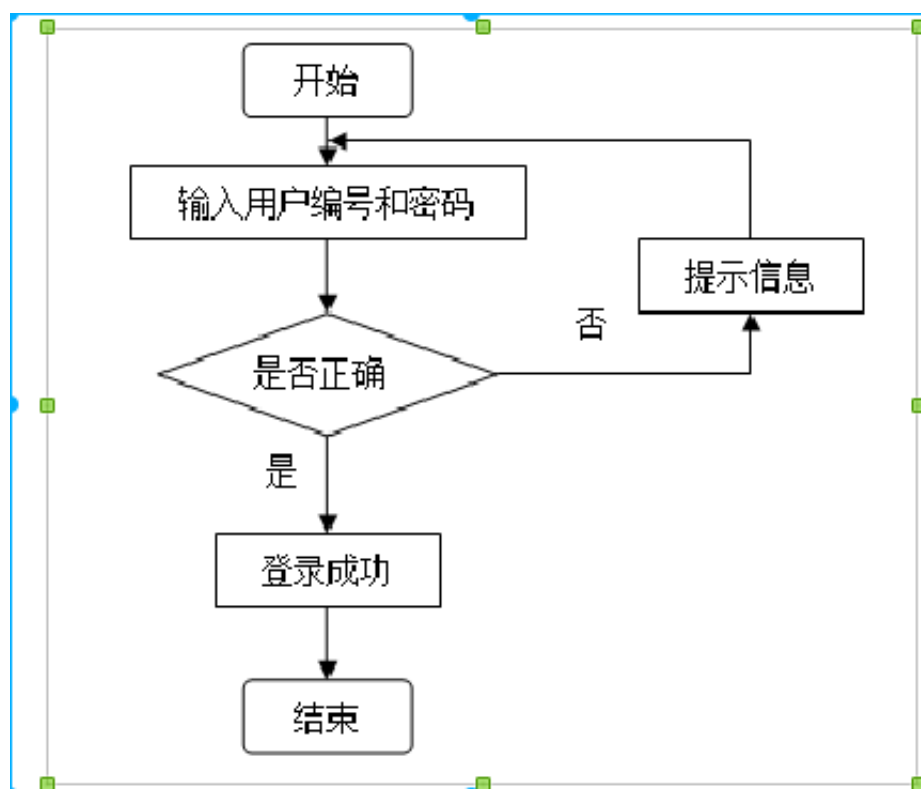


Figure 8. 系统登录流程图

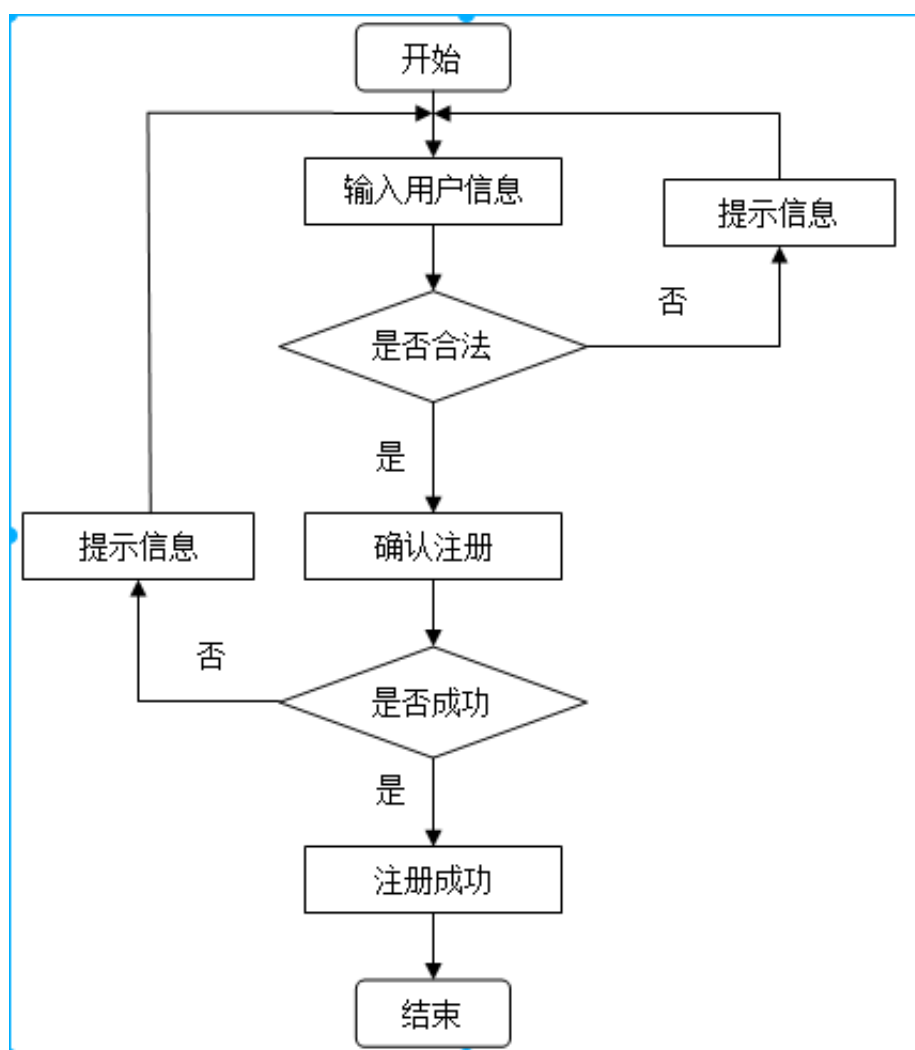


Figure 9. 系统注册流程图

### 6.1.2. 系统界面



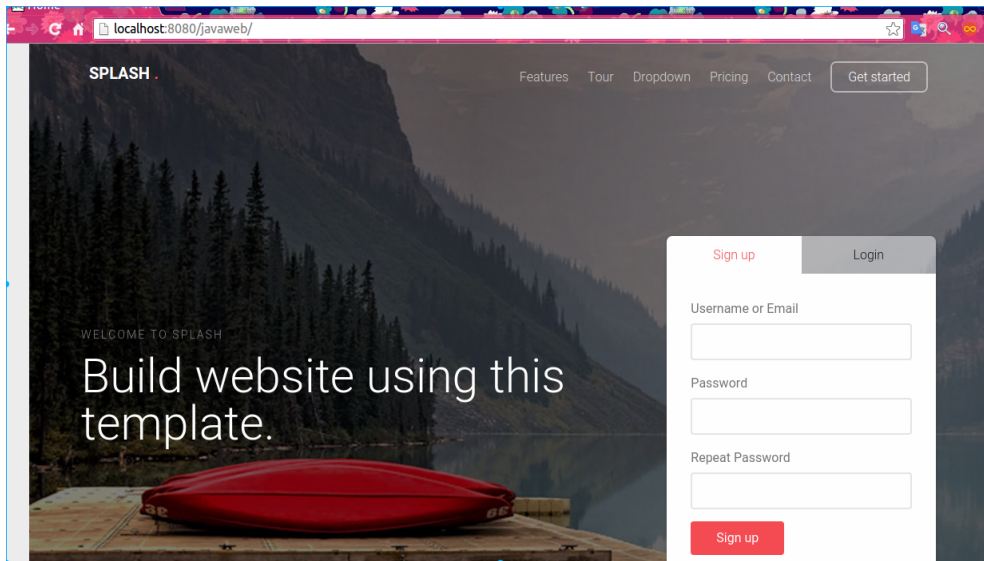


Figure 10. 登录注册界面

实现代码

```
public String login(){
    try {
        request.removeAttribute("login");
        String cid = request.getParameter("cid");
        String cpwd = request.getParameter("cpwd");
        if (!"".equals(cid)&&!"".equals(cpwd)) {
            ActionContext ac = ActionContext.getContext();
            Map<String, Object> session = ac.getSession();
            Cust cust = icsservice.findById(cid);
            if (cust != null && cust.getCpwd().equals(cpwd)) {
                session.put(cid, cust);
                HttpSession hs = request.getSession();
                hs.setAttribute("login", cust);
            }else{
                this.addActionError("用户名或密码错误!");
                return "index";
            }
        }else{
            return "index";
        }
    } catch (Exception e) {
        e.printStackTrace();
        return "index";
    }
    return "call";
}
```

## 6.2. 添加留言信息

添加留言信息功能，首先，用户会进入到添加留言信息页面，用户填写留言信息点击“提交留言”按钮后，检查留言信息是否为空，如果出现错误则提示相应的错误信息，重新写入正确的留言信息并且数据正确存入数据库之后，显示留言添加成功，失败则提示相应的错误信息。添加之后信息则会按照添加时间显示在所有留言信息中。

### 6.2.1. 功能流程图

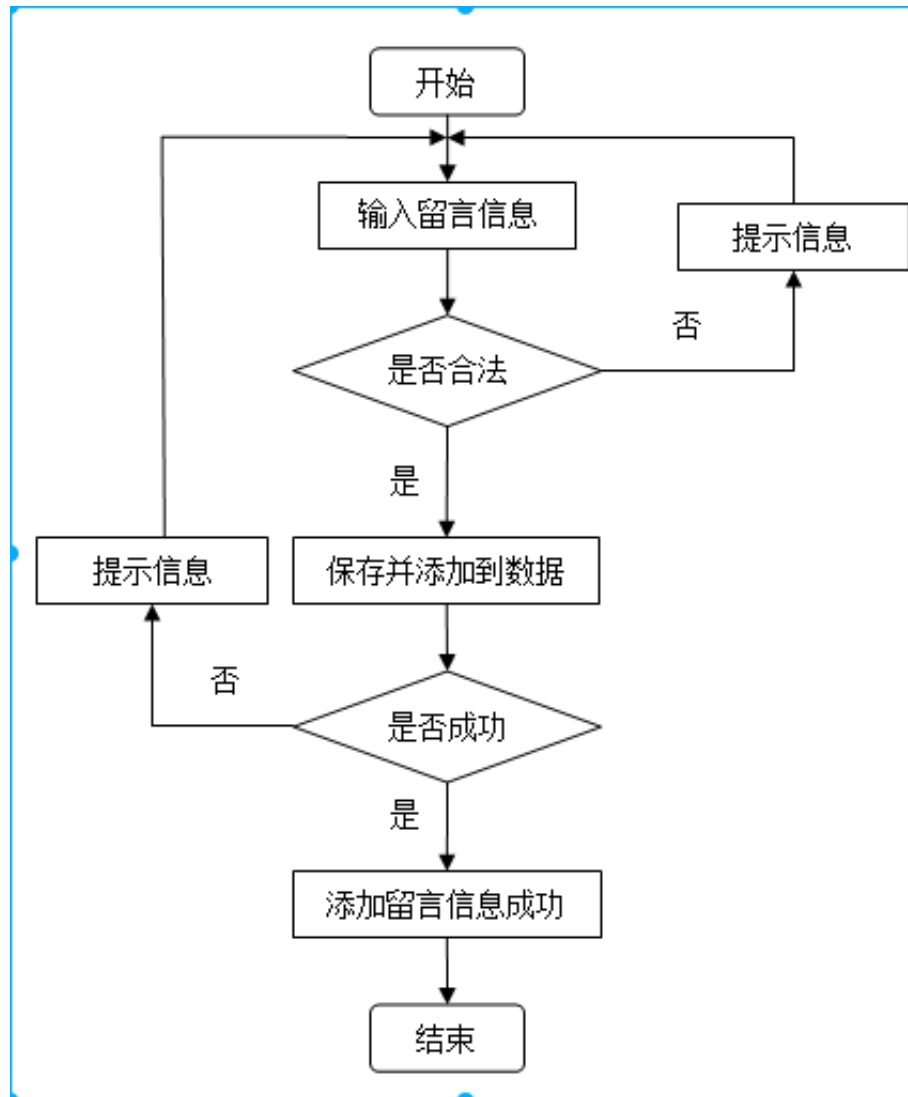


Figure 11. 界添加留言信息流程图

### 6.2.2. 系统界面



Figure 12. 添加留言信息界面

### 6.2.3. 实现代码

```
public String insertLiuyan(){
    try {
        HttpSession session = request.getSession();
        Cust cust= (Cust)session.getAttribute("login");
        Liuyan ly = new Liuyan();
        ly.setLcontent(request.getParameter("lcontent"));
        ly.setCid(cust.getCid());
        ly.setLdate(GetDate.getTime());
        ilservice.saveLiuyan(ly);
    } catch (Exception e) {
        e.printStackTrace();
    }
    return selectAllLiuyan();
}
```

## 6.3. 查询功能

点击查询功能可以查看所有留言

### 6.3.1. 功能流程图

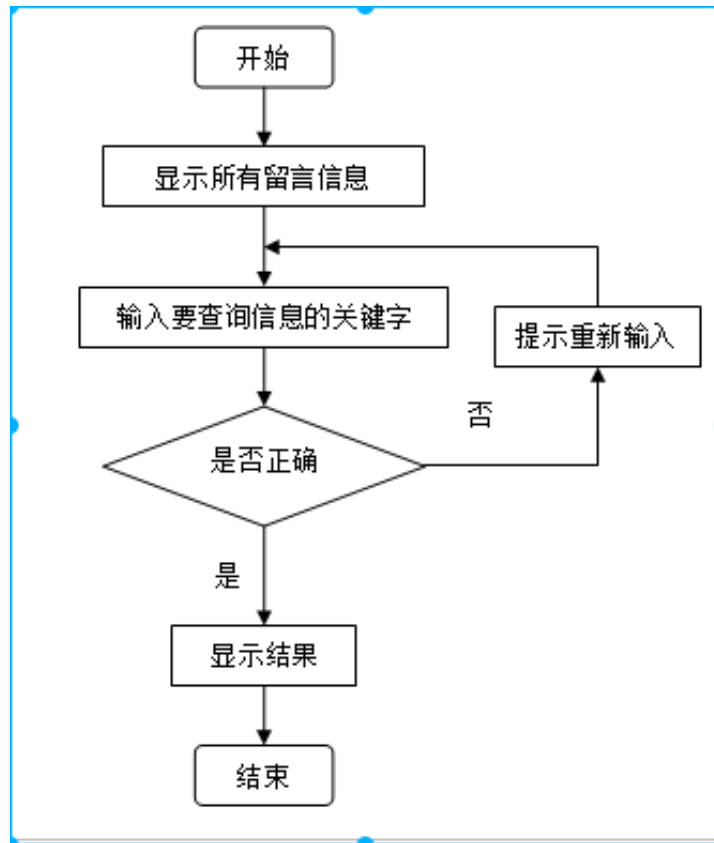


Figure 13. 查询功能流程图

### 6.3.2. 实现代码

```

public String deleteByHids(){
    try {
        String[] hids = request.getParameterValues("hd");
        ihservice.deleteByCids(hids);
    } catch (Exception e) {
        e.printStackTrace();
    }
    return selectAllHaoyou();
}

```

### 6.4. 其他功能

略

## 7. 心得

通过完成本次作业，我了解了一个web系统开发的一般过程，在完成web系统的开发过程中，运用了软件工程课程中学到的知识。首先，要明确系统功能，即做好需求分析。其次，要明确各系统功能模块的大概实现，即概要设计。在完成本次作业的过程中，也认识到了自己的不足。由于没有java基础，在编程上有一定的困难，需要大量的参考别人的代码或者模板，今后在这方面要加强学习。

## 8. 附:(开发过程展示)

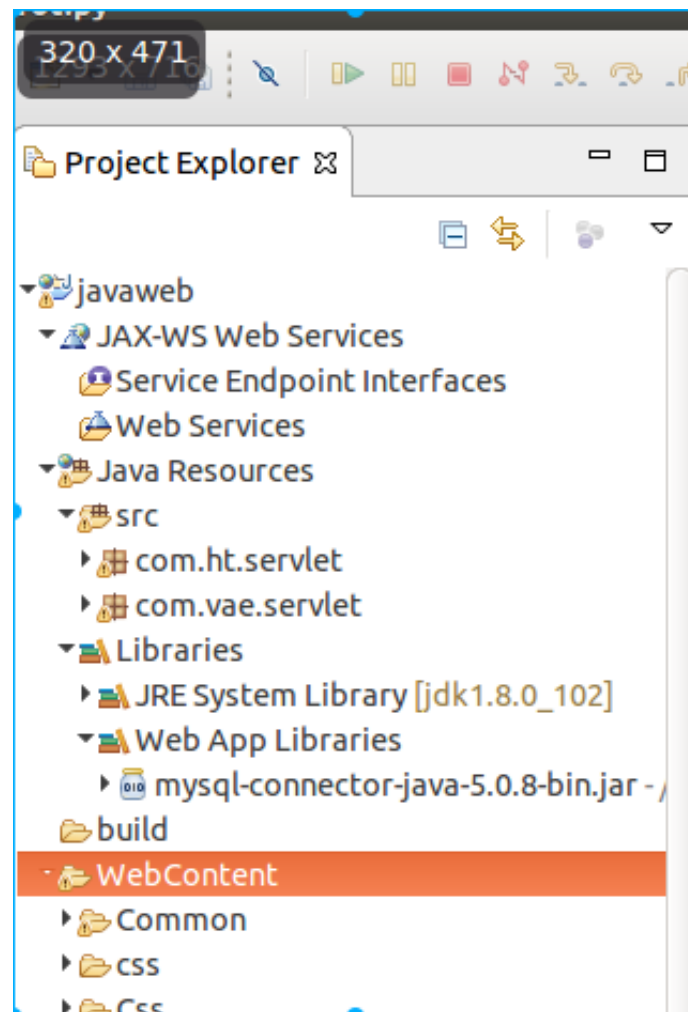


Figure 14. 工程目录1

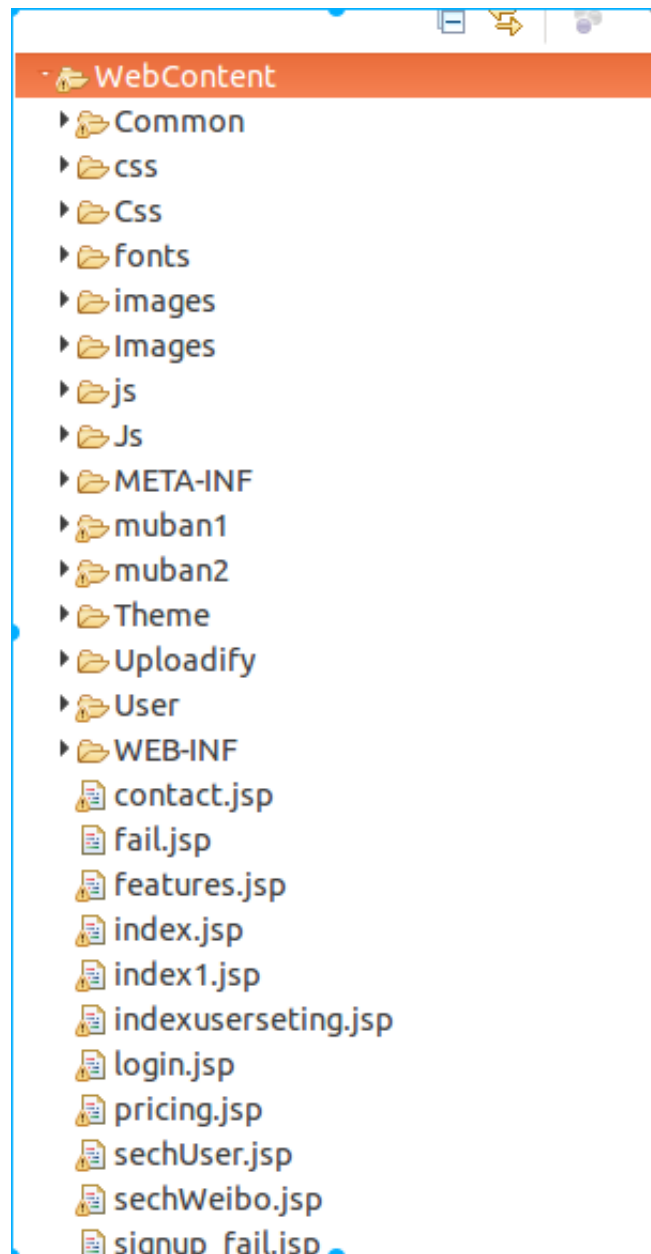


Figure 15. 工程目录2

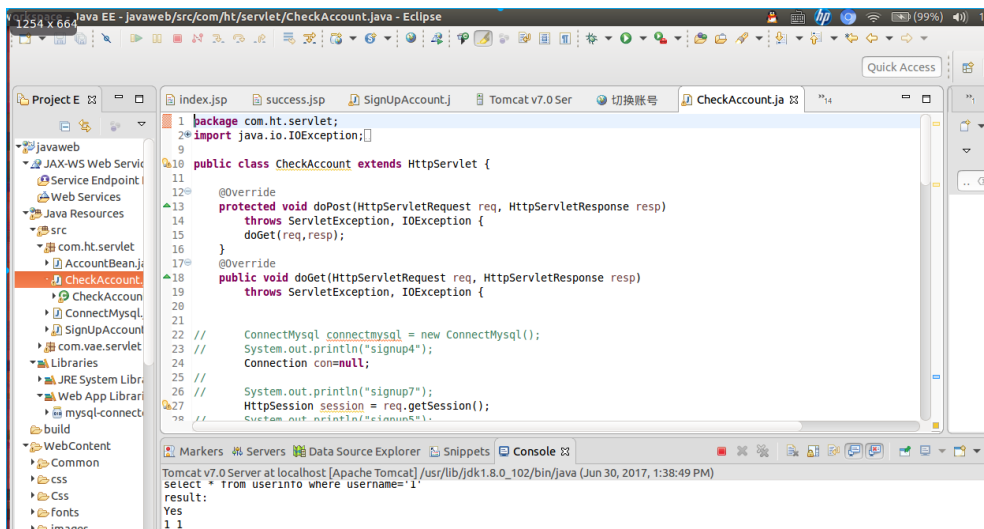


Figure 16. 登录注册实现(示例)

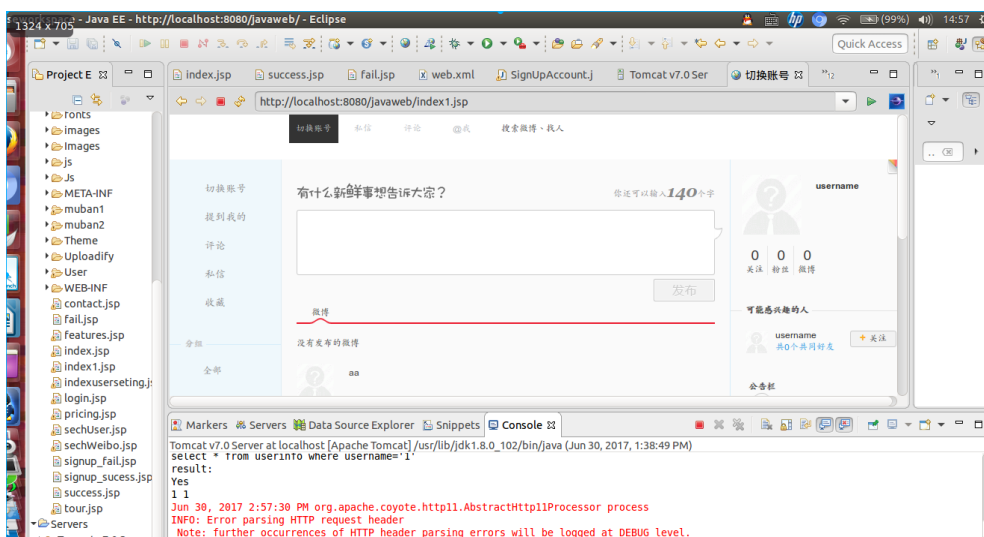


Figure 17. 界面展示