

# 西安电子科技大学

## 计算机学院

题目： JavaWeb 微博系统 .

班级： 1403013 .

姓名： 畅阳光 .

学号： 14030130040

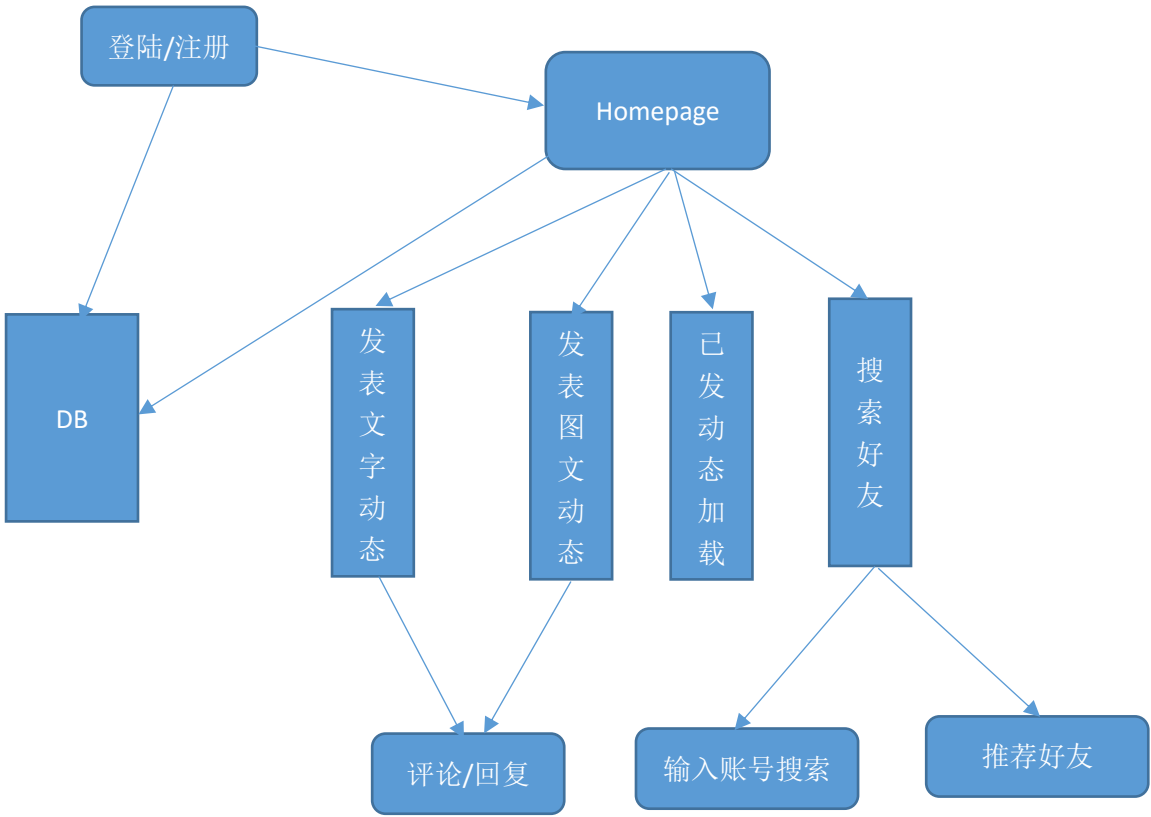
# 微博系统

## 一、设计要求：

使用 Java Web 开发和实现一个微博系统，可参考新浪微博，要求具备微博的基本功能：

- 1) 注册、登录和注销等功能；
- 2) 发布信息，如：文字、图片；
- 3) 用户发言、留言功能。

## 二、总体设计：



## 三、文件结构：

bin	201
node_modules	201
public	201
routes	201
views	201
~\$微博报告.docx	201
~WRL2365.tmp	201
app.js	201
package.json	201
README.md	201
微博.sql	201

app.js 为入口文件，routes 为路由文件，views 为界面渲染文件，public 中存放着网站用到的一些静态资源文件。运行过程为：npm start 开启服务器后，浏览器输入 url，然后 node.js 在 app.js 中找到对应的路由接口，然后根据接口去 routes 中找到处理路由，接着根据 routes 中指定的渲染文件，去 views 找到对应的渲染文件进行渲染。

渲染文件如下：

名称	修改日期	类型
 error.ejs	2017/4/9 16:01	EJS 文件
 Homepage.ejs	2017/6/22 0:01	EJS 文件
 index.ejs	2017/6/22 0:04	EJS 文件
 searchFriend.ejs	2017/6/20 20:17	EJS 文件

Index.ejs 为网站入口的渲染文件，包含用户的登陆和注册。Error.ejs 为 404 出错界面。Homepage.ejs 为微博主界面，包括动态的加载、动态的发表、好友列表的显示以及个人中心还有头像上传等。searchFriend.ejs 为好友搜索渲染文件，包含按账号搜索和好友推荐。

## 四、各部分具体实现：

### 1. 登陆注册模块：

实现功能：

用户注册，注册的过程中加入了表单验证，验证规则为：

- (1). 手机号必须为 11 位数字且必须是以 13/14/15/17/18/19 开头，因为 16 为服务号。
- (2). 学号为 11 位数字。
- (3). 密码长度必须大于 6。

用户登录时通过 Cookie 记录用户的登陆信息，下次登陆时无需再次输入。

Canvas 画出西电眺望塔。

主要代码：

#### a. 注册账号时的表单验证：

```
$(function () {  
    $('#logform').bootstrapValidator({  
        message: "This value is not valid",  
        feedbackIcons: { // 定义了检验通过与未通过时采用的图标  
            valid: 'glyphicon glyphicon-ok',  
            invalid: 'glyphicon glyphicon-remove',  
            validating: 'glyphicon glyphicon-refresh'  
        },  
    },
```

```

fields:
{
    tel:{
        message:"The tel is not valid",
        validators:{
            notEmpty:{
                message:"手机号不能为空"
            },
            stringLength:{
                min:11,
                max:11,
                message:"手机号必须为 11 位"
            },
            regexp:{
                regexp:/^1[34578]\d{9}$/,
                message:"手机号错误"
            }
            //手机号已经注册过未写
        }
    },
    StuId:{
        message:"The StuId is not valid",
        validators:{
            notEmpty:{
                message:"学号不能为空"
            },
            stringLength:{
                min:11,
                max:11,
                message:"学号必须为 11 位"
            },
            regexp:{
                regexp:/\d{11}/,
                message:"学号不正确"
            }
        }
    },
    password1:{
        message:"The value is not valid",
        validators: {
            notEmpty: {
                message: "密码不能为空"
            },
            stringLength: {

```

```

        min: 6,
        max: 20,
        message: "密码长度必须在 6~20 之间"
    }
}
},
password2: {
    message: "The value is not valid",
    validators: {
        notEmpty: {
            message: "密码不能为空"
        },
        identical: {
            field: 'password1',
            message: "两次密码输入不一致"
        }
    }
}
}
});

```

b. Cookie 记录用户登录信息函数：

Cookie 的生存周期为 7 天，路径为网站根目录 `path = "/"`。

函数功能为，当用户进入 index 界面后，程序先判断上次登录时候是否设置 Cookie，并判断是否到期，若有 Cookie 并没到期，则将上次设置到 Cookie 中的用户名和密码回填到表单中，用户选择是否记住密码，若选择记住，则此次登陆后，Cookie 中的内容被延至 7 天后到期，若未选中则 Cookie 被清空。

```

$(function () {
    //Cookie 模块

    if ($.cookie('loginname') != null' && $.cookie('logincode') != null' && $.cookie('loginname') && $.cookie('logincode')) //判断 Cookie 是否有效 若有效则显示出来
    {
        $('#loginname').val($.cookie('loginname'));
        $('#logincode').val($.cookie('logincode'));
    }

    $('#loginbut').click(function () { //登陆模块 发送 AJAX 请求到 loginroute 处理路由 （界面始终不跳转）

        var loginname = $('#loginname').val();
        var logincode = $('#logincode').val();
        var data = {"loginname": loginname, "logincode": logincode};

        $.ajax({

```

```

url:"http://localhost:3000/loginroute",
data:data,
success:function (data, stau) {
    if(stau == 'success')
    {
        if($("#remember").is(":checked")==true)//若选中设置 cookie
        {
            $.cookie('loginname', loginname, {path:"/", expires:7});
            $.cookie('logincode', logincode, {path:"/", expires:7});
        }
        else//若没选中 清除 cookie
        {
            $.cookie('loginname', null);
            $.cookie('logincode', null);
        }
        if(data.search('false')!=-1)
        {
            alert("用户名或密码错误_`false");
            return;
        }
        if(data.search('StuId'))
        {
            window.location.href="http://localhost:3000/Homepage?"+loginname+"&logincode="+logincode;

            return;
        }
    }
    else
    {
        alert(stau);
    }
},
error:function () {
    alert("用户名或密码错误_`");
},
});
});

```

## 2.动态模块与账号的注销退出：

实现功能：

个人信息编辑及头像上传。

发布文字动态。

发布图片和文字动态。

评论及回复模块。

加载用户及其关注的好友当天发布过的动态及评论。

点击查看更多，可查看用户之前发布过的动态及评论。

主要代码解释：

- a. 加载用户发布的动态，用户登录进入后，通过 ajax 请求服务器返回当天的动态，并通过动态创建表单展示到页面。

//加载当天动态

```
function load_dongtai(data)
{
    $.ajax({
        url: "/dongtai",
        data: data,
        success: function (dat, name)
        {
            if (name == "success")
            {
                var dat = JSON.parse(dat);
                //console.log(dat);
                var len = dat.length;
                //console.log(len);
                for (var i = len - 1; i >= 0; i--)
                {
                    dealOneDongtai(dat[i]);
                }
            } else
            {
                console.log(name);
            }
        }
    });
}
```

//通过动态创建表单，处理一条动态函数。

//单个动态处理函数

```
function dealOneDongtai(data)
{
    //console.log(data);
    var showDT = $('#showDT');
    var h2 = $('<h2>' + data['xuehao'] + ' :</h2><br>');
    showDT.append(h2);
    if (data['imageinfo'])
    {
```

```

        var img = $('');
        showDT.append(img);
    }
    if(data['dongtai'])
    {
        var dongtai = $('<p style="font-size: 18px">'+data['dongtai']+'</p>');
        showDT.append(dongtai);
    }

    var date = new Date(data['shijian']);
    var shijian = $('<p style="font-size: 15pt">'+date.getUTCFullYear()+"/"+(date.getMonth()+1)+"/"+date.getDate()+</p>');
    showDT.append(shijian);

    var Pbtn = $('<button class="ping_lunclass" onclick="pinlun_show(this)" id="'+data['flag']+'>评论</button>'); //评论按钮
    showDT.append(Pbtn);

    var pin_input = $('<form id="'+data['xuehao']+'>'+data['flag']+'><input type="text" /><button type="button" value="fabu" onclick="fabu_cancel(this)">发表</button><button type="button" value="cancel" onclick="fabu_cancel(this)">取消</button></form>');
    showDT.append(pin_input);
    pin_input.hide();
    var pinglun = data['liuyan'];
    if(pinglun)
    {
        showDT.append($('<hr style="border: 1px dotted gray">'));
        var tpinglun = pinglun.split('#');
        var pinglen = tpinglun.length;
        for(var i = 1;i<pinglen;i++)
        {
            showDT.append($('<p class="pinglun">'+tpinglun[i]+'<button class="huifuclass" id="'+i+'#"+data['flag']+'>回复</button></p>'));
        }
    }

    showDT.append($('<hr style="background-color: gray ;height: 1px;">'));
}

```

## b. 发表图片和文字动态：

通过序列化头像数据和文字消息，通过 ajax 采用 post 方式提交到后台，提交完成后刷新页面，头像上传和此处逻辑基本相似，不过头像上传完成后不用刷新页面，而是将头像 img 的 src 动态改变成所上传头像在服务器端的路径即可。

```

function uploadI() {
    var data = new FormData($('#myph_wordform')[0]);
    console.log(data);
}

```



```

data.append("loginname", loginname);
var url="http://localhost:3000/myph_word";
$.ajax({
    url:url,
    type:'post',
    data:data,
    processData:false, //默认为 true
    contentType:false, //string 类型 默认值: "application/x-www-form-urlencoded"。发送信息至服务器时内容编码类型。

    success:function (dat, name)
    {
        if(name == 'success' && dat!="false")
        {
            //console.log("success");
            location = location;
        }
        else
        {
            alert("动态发表失败-_-!");
        }
    }
});
}

```

### 3.搜索好友模块:

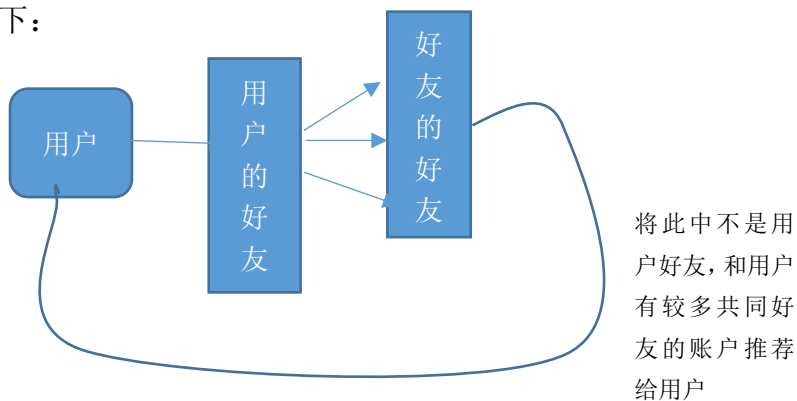
#### 实现功能:

用户输入账号查找到对应的账号信息，可点击关注或者取消关注。

好友推荐模块，会按照一定规则推荐 20 个非好友给你，每页展示 4 个，用户可以选择关注他们或不关注。

#### 主要代码解释:

- a. 好友推荐，推荐规则为：根据用户的好友的好友中，不在用户的好友列表中，且和用户有共同好友数最多的前 20 个好友给用户。用图表示如下：



五、数据库的设计：

Customers 用户注册时的信息表，用来记录用户的个人信息。主码为 StuId;

Column	Type	Default Value	Nullable	Character Set
StuId	char(11)		NO	utf8
username	varchar(255)		YES	utf8
code	varchar(255)		NO	utf8
tel	char(11)		NO	utf8
sex	varchar(4)		YES	utf8
addr	varchar(255)		YES	utf8
relation	varchar(255)		YES	utf8
photo_path	varchar(255)		YES	utf8
id	int(11)		NO	

Relaion 表，刻画人物之间的关系，A 和 B 账号在此表中有一条记录，表示 A 关注 B，A 关注了 B，A 就可以看到 B 发送的动态，而 B 不可以看到 A 的，如果 B 没有关注 A 的话。

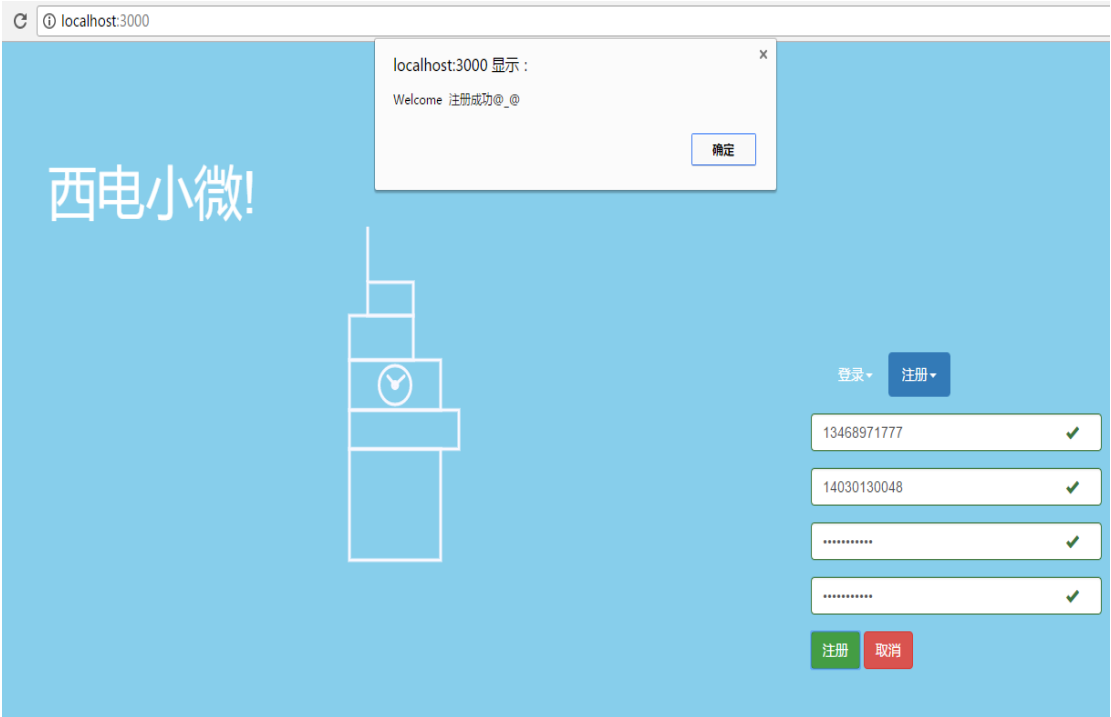
Column	Type	Default Value	Nullable	Character Set	Collation
xuehao	char(11)		NO	utf8	utf8_general_ci
xuehaoed	char(11)		NO	utf8	utf8_general_ci

Dongtai 表，记录所发布的动态和留言消息，主码为自增的标号 flag。

Column	Type	Default Value	Nullable	Character Set	Collation
flag	int(11)		NO		
xuehao	char(11)		NO	utf8	utf8_general_ci
dongtai	varchar(255)		YES	utf8	utf8_general_ci
imageinfo	varchar(255)		YES	utf8	utf8_general_ci
liuyan	varchar(255)		YES	utf8	utf8_general_ci
shijian	date		YES		

六、结果截图：

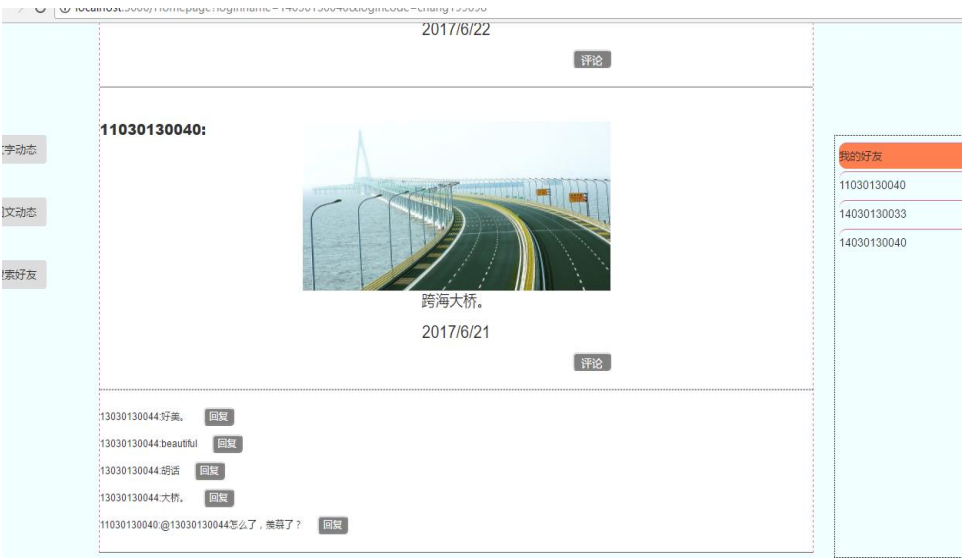
1.登陆注册界面：



2.动态加载界面：



3.查看更多功能及评论/回复：




4.按账号搜索好友：




5.好友推荐:


账号搜索

推荐好友

12030130050	默然	
女	海棠	
关注好友		

13030130044	傻傻	
女	竹园	
关注好友		

14030130060	天天	
女	丁香	
关注好友		

14030130089	null	
null	null	
关注好友		

七、总结:

从这次微博系统的总体设计，到各个模块之间关系的确定，再到最后每个模块具体功能的细化以及最终编码的实现，让我更清楚的理解和了解了网络上的各种协议及数据传输方式，特别是对 Ajax 技术及 Cookie 还有正则表达式，及对 DOM 操作的一些细节有了更清楚的认识，收益匪浅。