**模块:** 账号管理及登录

**功能:**

1.实时检测当前输入用户名是否已被注册,通过向findUsernameServlet发送get请求，输入json格式：

{username: xxx}

期望返回数据格式：  
 {

"userExsit" : ”false"

“msg" : "Username is legal"

} 或

{

"userExsit" : “true"

“msg” : "Username has already been registered"

}；

1. 通过post向RegisterServlet发送注册表单：输入格式为：  
    {

'username': uname,

'password': upwd

}

期望返回数据：

{msg : xxx}

Xxx = {

“true” 注册成功的时候

“has been registered” 用户名已被注册

“false” 其它未知原因

}

**网页名：**register.html

**对象名说明：**

用于检测用户名检测get处理接口：findUsernameServlet

用于处理注册页面的登记表单的post接口：RegisterServlet

**模块:** 账号管理及登录

**功能:**

1. -实现对账户的用户名及密码进行判断，输入正确则跳转至对应界面，错误则提示具体错误原因。

通过向LoginServlet发送一个get请求，数据为：

{

username : username,

password : password

}

期望返回数据：  
{

error\_type : {0，登录成功的时候

1. 账号错误的时候
2. 密码错误的时候

}

Id : {

0, 当前账号为顾客

1，当前账号为商家

}

}

**网页名：**login.html

**对象名说明：**

用于对账号密码进行判断的get接口：LoginServlet

**模块:** 展示界面

**功能:**

-1.实现将数据库中所有的商家的所有分类的所有商品进行一个展示

2.实现对应用户的一个欢迎标语

**网页名：**index.html

**对象名说明：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 对象类别 | 对象属性 | | | |
| user | name（当前顾客的用户名） | | | |
| sellers | 无属性，这是一个由所有seller组成的集合 | | | |
| seller | name（商家名） | | | kinds(a List of kind)（当前商家所拥有所有的商品类别） |
| kind | name（商品类别名） | | | items(a list of item)(该商家当前类下的所有商品) |
| item | item\_title(商品名) | price(商品价格) | seller\_info(商品描述信息) | |

**模块:** 数据修改

**功能:**

-1.实现对应商家上架商品，往数据库中写入新的商品信息

Seller页面会通过”add\_goods”这个url通过get方式传递一个json，格式为：  
{Name:Name,

Price:Price,

Kind:Kind,

Description:Description}

目的是为了往此商家的对应种类中加入商品信息。

检查是否有当前类，没有则创建对应类，并写入商品信息，有就直接往对应类写入信息。

期望返回数据json格式：  
{"type":true,}

或

{"type":false}

2.实现对当前商家的一个欢迎标语。

**网页名：**seller.html

**对象名说明：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 对象类别 | 对象属性 | | | |
| seller | name（当前商家名） | | | kinds(a List of kind)（当前商家所拥有所有的商品类别） |
| kind | name（商品类别名） | | | items(a list of item)(该商家当前类下的所有商品) |
| item | item\_title(商品名) | price(商品价格) | seller\_info(商品描述信息) | |

用于处理商品信息登记的get接口：add\_goods