## 工作日志

学号：1830090115

姓名：李响

* 学习内容：

|  |  |
| --- | --- |
| 序号 | 内容 |
| 1 | 商品搜索功能。 |
| 2 | 搭建搜索服务层。 |
| 3 | 利用HttpClient调用taotao-search服务。 |
| 4 | 商品详情界面实现。 |

* 心得体会：

cd /usr/local/solr/tomcat  
bin/startup.sh  
tail -f logs/catalina.out

文本

描述已自动生成

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成



Solr服务不需要连接Mysql，查索引库。

<?xml version="1.0" encoding="UTF-8" ?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:context="http://www.springframework.org/schema/context" xmlns:p="http://www.springframework.org/schema/p"  
 xmlns:aop="http://www.springframework.org/schema/aop" xmlns:tx="http://www.springframework.org/schema/tx"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-4.0.xsd  
 http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.0.xsd  
 http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop/spring-aop-4.0.xsd http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-4.0.xsd  
 http://www.springframework.org/schema/util http://www.springframework.org/schema/util/spring-util-4.0.xsd">  
  
 <!-- 包扫描器，扫描带@Service注解的类 -->  
 <context:component-scan base-package="top.lixiang007.taotao.search.service, top.lixiang007.taotao.search.dao"/>  
  
 <!-- 配置单机版SolrServer -->  
 <bean id="httpSolrServer" class="org.apache.solr.client.solrj.impl.HttpSolrServer">  
 <constructor-arg name="baseURL" value="http://10.211.55.12:8080/solr" />  
 </bean>  
  
</beans>

需要把数据库中的商品信息导入索引库，需要商品的id、商品的名称、商品的卖点、商品的价格、商品的图片、商品的分类名称。

原则：需要展示给用户的字段、需要搜索的字段需要添加到索引库。

1. 创建一个webapp的maven工程，类似taotao-rest；

2. 搭建Solr服务；

3. 完成商品信息导入；

4. 搜索服务发布的实现。

1. 利用HttpClient调用taotao-search服务；

2. 解决get参数乱码以及修改pojo的get方法解决图片显示问题。

package top.lixiang007.taotao.portal.controller**;**import org.springframework.beans.factory.annotation.Autowired**;**import org.springframework.stereotype.Controller**;**import org.springframework.ui.Model**;**import org.springframework.web.bind.annotation.RequestMapping**;**import org.springframework.web.bind.annotation.RequestParam**;**import top.lixiang007.taotao.portal.pojo.SearchResult**;**import top.lixiang007.taotao.portal.service.SearchService**;**import java.io.UnsupportedEncodingException**;***/\*\*  
 \* 前端搜索实现  
 \*/  
/\*\*  
 \** ***@Author*** *Xiang Li  
 \** ***@Date*** *2021/9/2 上午10:42  
 \** ***@Since*** *version-1.0  
 \** ***@青软实训*** *\*/*@Controller  
public class SearchController {  
  
 @Autowired(required = false)  
 private SearchService searchService**;** @RequestMapping("/search")  
 public String search(@RequestParam("q") String keyword**,** @RequestParam(defaultValue = "1") Integer page**,** @RequestParam(defaultValue = "60") Integer rows**,** Model model) {  
 //get乱码处理  
 try {  
 keyword = new String(keyword.getBytes("iso8859-1")**,** "utf-8")**;** } catch (UnsupportedEncodingException e) {  
 keyword = ""**;** e.printStackTrace()**;** }  
 SearchResult searchResult = searchService.search(keyword**,** page**,** rows)**;** model.addAttribute("query"**,** keyword)**;** model.addAttribute("totalPages"**,** searchResult.getPageCount())**;** model.addAttribute("itemList"**,** searchResult.getItemList())**;** model.addAttribute("page"**,** searchResult.getCurrentPage())**;** return "search"**;** }  
}

日历

描述已自动生成

1、Taotao-portal接收页面请求，接收到商品id。

2、调用taotao-rest提供的商品详情的服务，把商品id作为参数传递给服务。接收到商品详细详细。

3、渲染结果，展示商品详细页面

4、为了提高响应速度，商品详情页的内容需要分步骤加载。

第一步：先展示商品基本信息，例如商品的名称、图片、价格。

第二步：展示商品的描述，此时商品页面已经展示完毕，需要ajax请求商品描述。展示的时机是页面加载完毕后一秒钟。

第三步：展示商品规格。当用户点击商品规格选项卡时展示。使用ajax请求商品规格。

5、在搜索页面点击商品的图片或者商品名称请求商品详情页面。

商品详情页请求的url：/item/{itemId}.html

商品描述请求的url：/item/desc/{itemId}.html

商品规格请求的url：/item/param/{itemId}.html

* 遇到问题：

|  |  |
| --- | --- |
| 序号 | 问题 |
| 1 | {  "status": 500,  "msg": "org.apache.solr.client.solrj.SolrServerException: IOException occured when talking to server at: http://10.211.55.12:8080/solr\n\tat |

* 解决方案：

|  |  |
| --- | --- |
| 序号 | 方案 |
| 1 | 再导入一次！哈哈哈哈哈哈哈哈哈哈。 |

图形用户界面, 文本, 应用程序

描述已自动生成