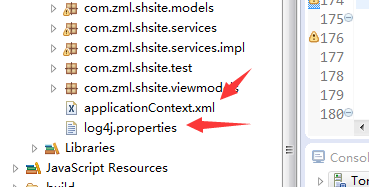
项目整体采用Spring4.3框架（包括IOC、DI、mvc等）实现项目整体架构和组件管理，使用Hibernate3.3和jdbc完成数据库访问读写，同时配置c3p0数据库连接池，访问高效，并且通过spring的支持，完美集成到了spring的bean容器中，同时对相关数据库实体访问采用了ehcache缓存框架，基于内存进行部分数据缓存。权限管理和认证基于spring security3.2,前端使用bootstrap UI框架，bootstrap validator校验框架，jqeury库等等

去网上找到spring mvc的访问示意图

一般是这样：

用户请求-相关拦截器（编码拦截器、权限拦截器）-前端控制器-后端控制器-视图解析器-返回用户

重要文件：



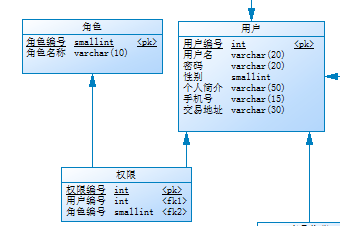
自上而下：spring上下文配置文件（数据库信息和连接池也配置在里面），日志配置文件



自上而下：spring securty配置文件、前端控制器文件、web根配置文件

**1.用户登录注册模块**

用户登录基于spring security3.2框架，与此相关的数据库表有三张：角色表、权限表、用户表，权限表中记录了用户与角色的关系，角色与用户多对多关系。系统预设两个角色（Admin管理员、User普通用户）,一个用户Admin



用户登录时，首先填写用户名和密码表单，点击登录，随后由spring框架进行管理，我们的代码逻辑建立在回调组件CustomUserDetailsServiceImpl里面，在这个bean里面进行用户查询和权限查询，随后框架会自动匹进行登录后的代码逻辑，如：将用户信息存入会话等

用户注册只需要填写相关表单信息，通过bootsrapvalidator校验后，点击注册，相关注册信息会传递到服务器，随后服务器进行用户增加，权限自动设置为普通用户，管理员用户是不准予注册的  
**2.二手商品浏览和查询模块**

二手商品浏览是通过对数据库商品表、用户评论表、商品收集表进行读取，获取商品信息，并统计商品评论个数和用户评论个数，随后渲染输出，返回给客户端浏览器，查询是通过在界面上方的查询表单，在上面输入商品名称，点击查询，商品名称便会通过post请求传递到服务器，服务器端会通过对商品名称进行数据库模糊查询（lke关键字），从而得出相关商品信息  
**3.二手商品详情浏览、收藏及留言模块**

二手商品浏览是通过读取了数据库对应商品、用户、评论以及收藏信息后，渲染给界面，同时在此运用到了spring security标签库，如果已经授权（登录）则可以查看完整信息，并且可以进行商品收藏和评论，否则全部看不到（你自己看看），商品可以进行收藏、取消收藏和评论，全部通过客户端js完成，通过jquery库ajax技术，页面无刷新  
**4.商品发布模块**

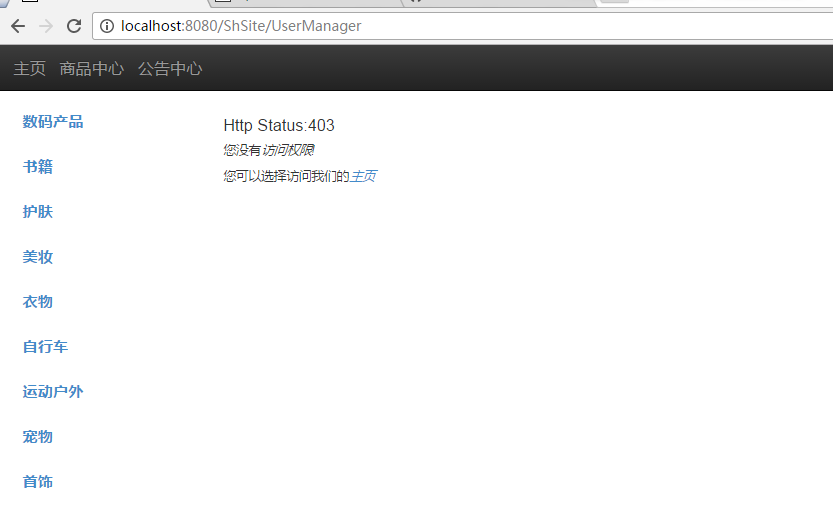
商品发布就是填写商品表单和上传图片，随后上传服务端。。。。  
**5.用户中心模块**

用户中心是通过查询当前登录用户的所有相关信息（用户信息、发布商品信息，收藏商品信息），然后渲染到界面上，然后用户可以修改基本信息，修改基本密码，修改发布商品，删除发布商品等等，前三者采用了bootstrapvalidator校验，密码修改还采用了远程校验，所谓校验就是输入正确信息才准许提交。其实我们网站只要是新增或者修改信息都用了校验  
**6.后台管理模块**

后台管理只针对于管理员，用了spring security进行安全控制，通过spring security标签，实现针对于不同用户显示不同界面，管理员用户可以看到如下导航栏：

而未登陆用户和普通用户：

 同时未登陆用户和普通用户如果想强制通过地址访问,会自动拦截和提示：



商品管理、用户管理主要就是对商品和用户的增加、删除、修改，几个表单和校验而已，没什么好说的  
**7.系统公告模块**

未登陆用户和普通用户可以浏览公告，

管理员用户可以增加、删除、修改公告，也就是几个表单和校验