**中信中收搜索模块框架设计及开发指导**

# 系统架构图

# 分布式及负载均衡方式说明

本系统作为信用卡商城的内部系统，所有接口都只供内部系统间调用，不对最终用户直接开放。所有请求消息均为无状态式消息，系统内部不保存任何状态。可以通过nginx做反向代理，进行水平扩展。

# 使用框架说明

本框架使用 spring、mybatis、druid、mysql、redis、solrj常用开源技术，方便进行扩展。

## 代码目录结构说明



## 基类

BaseController

所有controller都必须继承自此基类，此基类中定义了log变量和统一的异常处理方式。具体的异常处理方式在“异常规则说明”章节介绍。参数验证失败统一处理方法：validateError，在参数验证报错时必须调用。详见“参数校验”章节。

## 参数配置、使用方法说明

所有参数必须通过conf/ config.properties进行配置，通过${XXXXXXX}的方式进行引用。

禁止在代码或配置文件中直接引用pom中设置的参数。因为运维人员只在config.properties中改动参数。

## 全局变量配置、使用方法说明

宏定义：

com.citiccard.emall.search.web.util.Constants类中以静态常亮的形式定义。

提示文字定义：

禁止在代码中直接编写提示文字。

在conf/words.properties中定义，如 just\_for\_test=测试

使用方式：WebPropertyHolder. getContextProperty("just\_for\_test ")

## 前后台数据交互规则说明

所有信息以json串的形式返回，无前台页面。

## 前端框架使用说明

所有信息以json串的形式返回，无前台页面。

## 前后台参数校验方式说明

无前台验证。

后台验证使用hibernate-validation。

hibernate-validation使用详见：http://jinnianshilongnian.iteye.com/blog/1990081

错误提示文字需要写在words.porperties中，禁止直接写在代码中

参数验证失败必须调用基类中的validateError方法。

详细使用方法见：DemoController. testvalidator

## 日志编写规则说明

使用aop加入controller中“@ResponseBody”注解的请求（LogAspect）。记录请求ip、url、请求参数、返回值、执行时间。所有信息写入debug级别的日志，info级别的日志只记录前25个字节。

2016-10-28 10:37:49,483 DEBUG [qtp210278405-19] (LogAspect.java:98) - 127.0.0.1 request URL /testvalidator.ajax parameter {} body{"loginTime":"sdfds","token":"sdfdsf","userId":"123"}

2016-10-28 10:37:49,529 INFO [qtp210278405-19] (LogAspect.java:101) - 127.0.0.1 request URL /testvalidator.ajax parameter {} body{"loginTime":"sdfds","token":"sdfdsf","userId":"123"}

2016-10-28 10:37:50,989 DEBUG [qtp210278405-19] (LogAspect.java:109) - 127.0.0.1 request URL /testvalidator.ajax output {"result":{"code":0,"data":{"loginTime":"1477622269856","token":"E8C1E9377B8A7257D808EAB09164B719","userId":"12345678"}}} used 1662ms

2016-10-28 10:37:50,989 INFO [qtp210278405-19] (LogAspect.java:112) - 127.0.0.1 request URL /testvalidator.ajax output {"result":{"code":0,"data used 1662ms

在拦截到异常后会记录error日志

logger.error("catch exception:"+ e.getMessage(), e);

开发人员需要加入日志的位置：

与搜索引擎通信的事件必须以debug的形式记录：记录请求参数、消耗时间；无需记录返回内容，搜索引擎会记录。

## json与object相互转换方式说明

使用fastjson进行互转。

类转json：JSON.toJSON

json转类：JSON.parseObject

## 权限校验方式说明

内部接口系统暂无权限控制

## 多语言实现方式说明

不需要

## 图片上传、展示方法说明

不需要

## 文件上传、下载功能说明

不需要

## excel导入、导出功能说明

不需要

## 分页功能说明

不需要

## 异常规则说明、使用方法、捕获方式

在controller层定义了统一的异常拦截，并记录日志、返回错误结果。

开发人员无需手动拦截异常。

controller捕获到异常后会把异常解析为返回值result并记录日志。

**public** BaseResult exception(HttpServletRequest request, Exception e) {

BaseResult result = **new** BaseResult();

result.setCode(-1);

result.setMessage("服务端执行失败:"+ e.getMessage());

logger.error("catch exception:"+ e.getMessage(), e);

**return** result;

}

## 通用工具类说明

Constants：宏定义类

MD5Util：MD5加密使用

WebPropertyHolder：获取文本常量

## 数据库连接方式说明

使用druid作为数据库连接池，配置文件在

conf/ search-db-config.xml

开发中各开发人员的数据库连接地址在pom.xml中，各自设置。

<pom.jdbc.search.url>jdbc:mysql://172.16.75.48:3306/solrtest?characterEncoding=UTF-8

</pom.jdbc.search.url>

<pom.jdbc.search.username>root</pom.jdbc.search.username>

<pom.jdbc.search.password>123456</pom.jdbc.search.password>

## 数据库事务

在com.citiccard.emall.search.model包中通过aop加入了事务，要求包下所有类的命名形式为 \*Model。

默认不加入事务，符合以下命名的方法会加入事务，开发人员在编写方法时需要考虑是否使用事务：

<tx:method name=*"find\*"* propagation=*"REQUIRED"* read-only=*"true"* />

<tx:method name=*"query\*"* propagation=*"REQUIRED"* read-only=*"true"* />

<tx:method name=*"list\*"* propagation=*"REQUIRED"* read-only=*"true"* />

<tx:method name=*"search\*"* propagation=*"REQUIRED"* read-only=*"true"* />

<tx:method name=*"get\*"* propagation=*"REQUIRED"* read-only=*"true"* />

<tx:method name=*"save\*"* propagation=*"REQUIRED"* />

<tx:method name=*"update\*"* propagation=*"REQUIRED"* />

<tx:method name=*"delete\*"* propagation=*"REQUIRED"* />

<tx:method name=*"create\*"* propagation=*"REQUIRED"* />

<tx:method name=*"insert\*"* propagation=*"REQUIRED"* />

<tx:method name=*"do\*"* propagation=*"REQUIRED"* />

<tx:method name=*"auto\*"* propagation=*"REQUIRED"* />

## session共享方式

目前不需要使用session

## 缓存使用及存储内容

使用redis做缓存，链接地址在pom.xml中设置。

框架中使用redisTemplate作为缓存的接口，使用方法参见：UserRedisTemplate类。

不要在逻辑层直接操作缓存，必须封装成对应数据结构放在：com.citiccard.emall.search.web.redis

## 定时任务

不需要

## 中间件配置及使用方法说明

使用中间件druid，不但作为数据库连接池，还作为spring和mybatis的监控。

在发布地址的/druid下可以看到监控界面，账号密码在web.xml中配置







## 通用过滤器、拦截器、切面

网络访问拦截器NetInterceptor，拦截所有访问/\* 。备用，目前为加入具体操作。

数据库事务切面，详见“数据库事务”章节。

访问日志切面，详见“日志编写规则说明”章节。

## 框架其他功能说明

返回值接口

BaseResult类，用于对所有restful返回值进行封装。所有请求的返回值必须使用封装。

返回码：1 成功，默认值

-1 发生异常，异常信息在message中

-2请求参数错误，错误信息在message中。

## 内部开发测试环境说明

内部svn地址 http://172.16.75.209/svn/mi-emall

测试环境暂时为开发者本机。

# 其他需要开发时特殊注意的问题

文档中标红位置需要开发人员特别关注。