

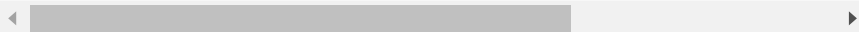
目录

介绍	1.1
站点架构	1.2
前端	1.3
前端规范	1.3.1
webpack	1.3.2
pc端	1.3.3
wap端	1.3.4
后端	1.4
后端规范	1.4.1
第三方插件(模块开发)	1.4.2
varnish	1.4.3
数据库	1.5

- [文档简介](#)

文档简介

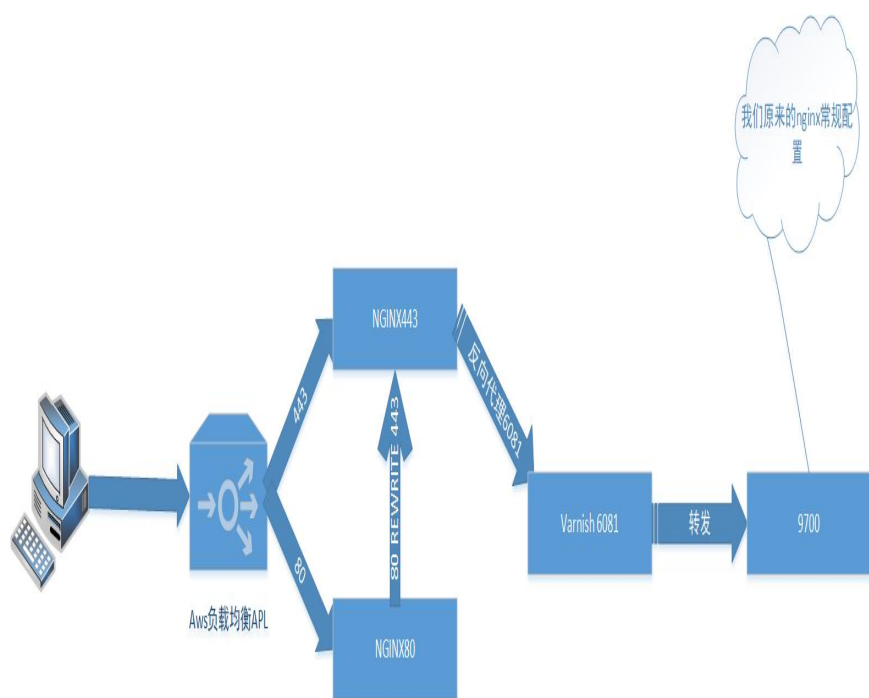
此文档主要就介绍了在julia站点在重构过程中遇到的问题及相关注意事项,



- 服务器架构调整

服务器架构调整

为了使用新的缓存机制varnish，我们需要对服务器的架构进行调整，基本架构



对于varnish的具体介绍与用法我们会单独汇总

- 前端介绍

前端介绍

本次重构的核心就是前端的调整，主要有以下几个方面：

- 1、前端采用BEM（模块名 + 元素名 + 修饰名）命名规范
- 2、js 库基于jqeury(Zepto),原有prototype弃用
- 3、第三方代码均放于gtm，修改原有的硬编码为软编码
- 4、通过webpack进行文件的管理及压缩
- 5、css 转为less 进行书写（提供代码的复用性）

- 前端规范手册
 - 一、规范目的
 - 二、前端主题结构
 - 三、框架及命名规范
 - 四、网页细节要求：
 - 五、css.js引入规则
 - 六、css命名规范（BEM命名法：模块名 + 元素名 + 修饰器名）
 - 七、css书写规范（提高可读性，减少维护成本）
 - 八、注释规范
 - 九、js规则
 - 十、image图片
 - 十一、phml视图文件

前端规范手册

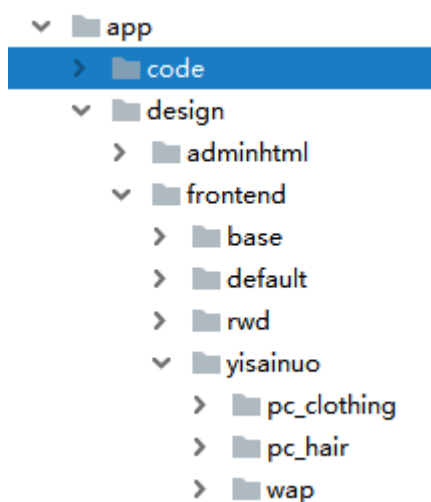
一、规范目的

为提高团队协作效率，便于后台人员添加功能及前端后期优化维护，
输出高质量的文档，特制订此文档。本规范文档一经确认，
前端开发人员必须按本文档规范进行前台页面开发。

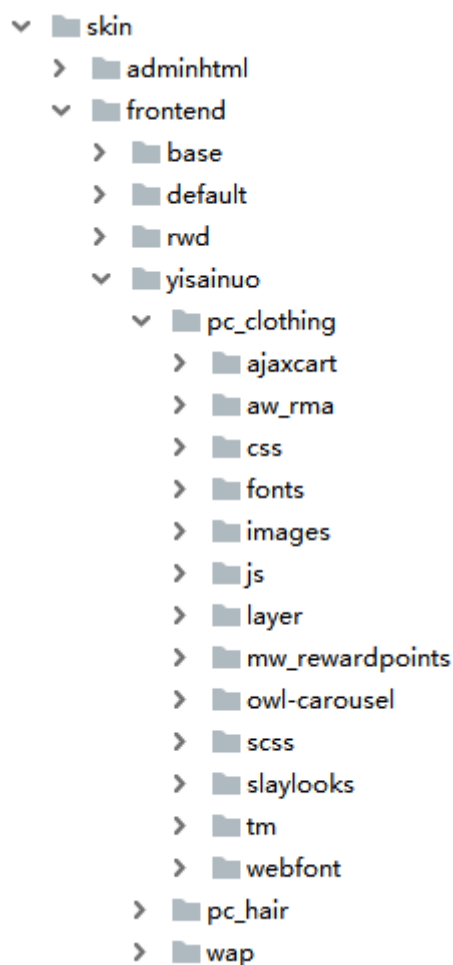
(注释：细节规则还在进一步完善)

二、前端主题结构

desgin



skin



针对有大型的活动或者针对某个栏目进行板式的调整;进行新的视图创建;原则上不影响先有的站点布局及板式

三、框架及命名规范

文件夹及文件名称统一用小写的英文字母、数字和下划线的组合,其中不得包含汉字、空格和特殊字符(做到见名知意);命名原则的指导思想:
一是使得你自己和工作组的每一个成员能够方便的理解每一个文件的意义,
二是当我们在文件夹中使用“按名称排列”的命令时,同一种大类的文件能够排列在一起,以便我们查找、修改、替换等操作。

所有页面在body上都有一个class,名称与控制器名称保持一致
(eg:cms-index-index,catalog-category-view,catalog-product-view)

四、网页细节要求：

- 1、Alt和Title 对应 `img` , `href` 标签 必须添加这两个属性
- 2、`href` 空链接: `javasc`
- 3、`h` 标签页面应用: (待定)
- 4、所有与产品相关的 单个产品的最外层需要添加 `data-id`, `data-category` 属性 (配合第三方代码部署)
- 5、所有图片必须使用延迟加载

五、css,js引入规则

- 1、基础: 该类别中包含的是默认的 CSS 样式。作为其他样式的基础—`base.css`。
- 2、不同页面分别引入:
自定义.css 文件 格式为 `longqi.使用页面名称(功能名称).css`
(
首页: `longqi.home.css`;
产品列表页: `longqi.catalog.category.css`;
详情页面: `longqi.catalog.product.css`;
支付页面: `longqi.checkout.css`;
用户页面: `longqi.customer.css`;
其他活动页面: `longqi.对应活动名称.css`
);
- 3、主题: 该类别和状态类似, 只不过是用来改变布局和模块的视觉效果`theme.css`。(大型活动)
- 4、css 正式站引入均是压缩过的 (`.min.`)
- 5、基于less 完成

六、css命名规范 (BEM命名法: 模块名 + 元素名 + 修饰器名)

- a. - 中划线 : 仅作为连字符使用, 表示某个块或者某个子元素的多单词之间的连接记号。
- b. __ 双下划线 (改成_) : 双下划线用来连接块和块的子元素
- c. -- 双中划线: 单下划线用来描述一个块或者块的子元素的一种状态
- d. 其中名称用统一用小写英文单词 (可以缩写, 但需要在注释中说明全称)
比如分页组件, 那么该组件模块就名为`page-btn`, 组件内部的元素命名都必须加上模块名:

```
<div class="page-btn">
    <button type="button" class="page-btn_prev page-btn--act
    <button type="button" class="page-btn_next">下一页</buttc
</div>
```

七、css书写规范（提高可读性，减少维护成本）

待完善....

八、注释规范

```
/* Header */
内容区
/* End Header */

/**
 * 对模块进行说明。
 * 如果模块类名缩写也说明一下。
 * Firecheckout 缩写成fc
 *
 */
```

九、js规则

以上css规则同样适用js规则

十、image图片

所有图片转化为webp格式----这的转化通过nginx嵌入lua完成

十一、phml视图文件

1、所有视图文件中不要出现命名js方法的现象出现

2、做好注释

```
/**
 * 评论数量和星级
 *
 * @see Mage_Catalog_Block_Product_List
 */
```

任何添加的代码都要做好注释

3、不要在视图代码中出现大面积后端逻辑代码，理论上视图中的后端代码仅负责传送数据

- [webpack 在我们传统站点的作用](#)
- [webpack 压缩问题 \(author:徐彦广\)](#)
 - [压缩的配置](#)

webpack 在我们传统站点的作用

- 1、统一管理css,js
- 2、对css进行压缩
- 3、对js进行混沌压缩

webpack 压缩问题 (author:徐彦广)

压缩的配置

- 1) 获取需要压缩的所有js、css的文件名称的字符串
- 2) 将字符串转化成可以被webpack识别的对口文件以及想对应路径
- 3) 全局的函数需要写成`window.name=function(){};`的形式,告知压缩插件不需要对此函数进行混淆,因为混淆后的函数 无法被全局调用。

```
/*将字符串编辑成对象格式 */
function mapCSS(str, output, input, type) {
    var cssArr = str.split('.') + type;
    var cssObj = {};
    for (let i = 0; i < cssArr.length; i++) {
        let key = cssArr[i].replace(/\s*/g, "");
        if (key == '') {
            continue;
        }
        cssObj[output + "/" + key] = input + "/" + key + "." + type;
    }
    return cssObj;
};
/*end 将字符串编辑成对象格式 */
```

需要用到以下插件及loader:

```
"css-loader": "^3.4.2",
  "file-loader": "^5.0.2",
  "mini-css-extract-plugin": "^0.9.0",
  "optimize-css-assets-webpack-plugin": "^5.0.3",
  "style-loader": "^1.1.2",
  "uglifyjs-webpack-plugin": "^2.2.0",
  "url-loader": "^3.0.0",
  "webpack": "^4.41.5",
  "webpack-cli": "^3.3.10",
  "webpack-fix-style-only-entries": "^0.4.0"
const MiniCssExtractPlugin = require('mini-css-extract-plugi
const optimizeCss = require('optimize-css-assets-webpack-plu
const UglifyJsPlugin = require('uglifyjs-webpack-plugin');//
const FixStyleOnlyEntriesPlugin = require('webpack-fix-style
```

具体的细节请看站点目录下 `webpack.config.js` 文件

- 一、目前存在的问题及处理（author:徐彦广）：
 - 1、flex布局兼容问题
 - 2、横向滚动问题
 - 3、头部导航问题
 - 4、首屏图片懒加载问题
 - 5、ajax动态加载js书写注意问题
 - 6、swiper 使用问题
 - 7、cookie作用域问题
 - 8、产品详情页主图滚动问题
 - 9、评论上传图片大小限制问题
 - 10、产品价格前端计算逻辑
- 二、通用弹框方法
- 三、具体js,css引入的位置及说明会在webpack.config.js 进行
- 四、第三方库说明
- 五、第三方工具的引入：

一、目前存在的问题及处理（author:徐彦广）：

1、flex布局兼容问题

```
display: -webkit-box; /* Chrome 4+, Safari 3.1, iOS Safari 3.2+
display: -moz-box; /* Firefox 17- */
display: -webkit-flex; /* Chrome 21+, Safari 6.1+, iOS Safari 7+
display: -moz-flex; /* Firefox 18+ */
display: -ms-flexbox; /* IE 10 */
display: flex; /* Chrome 29+, Firefox 22+, IE 11+, Opera 12.1/17
```

但是这样也只能兼容到ie10，想要兼容ie9还要自己写适合自己网站polyfillcss

2、横向滚动问题

屏幕适配时注意要对绝对定位的元素进行响应式处理，避免发生出现横向滚动条

3、头部导航问题

header导航栏部分为了节约元素数量，把一级导航和其对应的二级导航放在了一起，需要注意是用每个一级导航的位置和其二级导航的位置把两者关联起来的，动画收缩的背景是写在包含导航盒子的外面的。

4、首屏图片懒加载问题

注意首屏显示的图片是不需要懒加载的
导航上图片，**banner**的首个图片，产品第一个主图...

5、ajax动态加载js书写注意问题

- 1) **ajax**.动态添加的产品列表 需要重新启动懒加载。
- 2) 动态添加会导致绑定事件失效，需把事件委托给**document**或者**body**
- 3) 动态添加的表单需要重启动校验

6、swiper 使用问题

每个轮播都需要**new**不同的**swiper**对象进行配置和区分。

7、cookie作用域问题

在js命名**cookie**的时候要指定作用域

8、产品详情页主图滚动问题

左侧产品图片详情随进度条滑动效果，使用的是**display: fixed**，这样更加流畅，需要注意的是需要判断头部是否有**advert**推广，调整在不同的高度触发

9、评论上传图片大小限制问题

上传评论图片功能 限制图片大小的时候，需要注意**ie**和**firefox**的在查看文件大小时有区别。

```
//获取文件大小
if (Sys.firefox) {
    filesize = input.files[0].size;
} else if (Sys.ie) {
    var fileobject = new ActiveXObject("Scripting.FileSystemObjec
    var file = fileobject.GetFile(input.value); //获取上传的文件
    filesize = file.Size; //文件大小
}
```

10、产品价格前端计算逻辑

计算产品价格，利用calPrice函数计算选中选项的价格，
用updatePrice函数分别计算旧价格与新价格以及折扣掉的价格，
用roundTwo函数保留两位小数，并输出到对应的盒子中。

二、通用弹框方法

```
1、openLoading();    加载框
2、closePopup('loading'); 关闭
3、openMessage (msgObject, oneBtn, callBack); 确认框
/**
    1.msgObject可以为字符串，也可以写成{
        string:'',//段落
        title:'',//标题
    }
    2.oneBtn为'yes',只添加yes按钮 为'no'只添加no按钮,为window.将为y
    3.callBack 只有在当oneBtn 为function是 回味 no按钮 添加回调函
*/
4、openTips (string, isFail, delay);
/**
    1.string, 提示框内容
    2.是否为报错内容
    3. 停留时间默认为3000ms
**/
5、openPopup (ElementID) ; //ElementID为需要打开或者关闭的弹窗的id
6、closePopup (ElementID)
```

三、具体js,css引入的位置及说明会在webpack.config.js 进行

四、第三方库说明

Jquery: 用于方便实现复杂的js功能
elevatezoom (jquery): 用于实现详情页产品图片细节预览效果,
form-validator (jquery): 基于jquery的校验插件, 需要手动更改部分代码
velocity (jquery): 用于链式有顺序的执行animate动画, 方便写出复杂的
baguetteBox: 原生js灯箱插件, 兼容性良好体积小, 主要用于详情页预览评论
idangerous.swiper: 2.7.6 版本的swiper, 可兼容到ie9, 用于网站整个轮播
lozad: 依赖于原生js中intersection-observer功能的懒加载插件。性能高,
intersection-observer: 用于lozad兼容ie9的polyfill。

五、第三方工具的引入:

- 1、refersion 联盟
- 2、bizrate 第三方评论
- 3、结构化数据
- 4、bing
- 5、facebook
- 6、google评论
- 7、onesignal
- 8、salesiq

- 注：这里需要说明的是，第三方代码的引入对手机端的评分还是有一定的影响的，尤其是facebook,salesiq 影响较明显
- 一、目前存在的问题及处理（author:徐彦广）
 - 1、由于使用zepto库替换jquery,这里会使用tab 事件替换click事件
 - 2、处理有遮罩层下方依然滚动问题
 - 3、header 结构问题
 - 4、回到顶部问题
 - 5、产品列表js处理
 - 6、产品详情
- 二、通用弹框
- 三、用到的第三方库
- 四、第三方代码
- 五、输入框通用结构说明（author:石爱朵）
 - 1、输入框通用结构
 - 2、输入框右侧有apply 按钮结构的
 - 3、多选框通用结构
 - 4、单选框通用结构

注：这里需要说明的是，第三方代码的引入对手机端的评分还是有一定的影响的，尤其是facebook,salesiq 影响较明显

一、目前存在的问题及处理（author:徐彦广）

1、由于使用zepto库替换jquery,这里会使用tab 事件替换 click事件

2、处理有遮罩层下方依然滚动问题

添加以下两个全局方法


```

var t = 0;
window.unScroll = function() {
    t = $('header').offset().top;
    if ($('#xker')[0]) {
        t = t - $('#xker').height();
        if (t < 0) {
            t = 0
        }
    }
    $(document.body).addClass(function() {
        $(this).css('top', -t + 'px');
        return 'unScroll';
    });
    window.scrollToT = function() {
        $(document.body).removeClass('unScroll').css('top',
        window.scrollTo(0, t);
    }
}

```

3、header 结构问题

Header部分需注意四个部分组成，分别是nav 、 navbar、 navbar左侧滑出后的

4、回到顶部问题

因为ios对scroll事件的处理不同于安卓浏览器，这个动画效果需要借助原生js

5、产品列表js处理

产品目录页面的产品下拉刷新的工程需要借助原生js中的
IntersectionObserver 对象来完成。该页面的需要注意动态添加
的内容较多，绑定时间需要委托给父级元素、document或body

6、产品详情

- 1)产品详情页tab选项卡这一块滚动动画依旧需要使用回到顶部所用的原生js
- 2)产品详情页新增上传图片增加缩略图的功能，需要用到原生js中的FileReade

```
function readAndPreview(file) {
    // Make sure `file.name` matches our extensions crit
    if (/\. (jpe?g|png|gif)$/i.test(file.name)) {
        var reader = new FileReader();
        reader.addEventListener("load", function() {
            var image = new Image();
            image.title = file.name;
            image.src = this.result;
            preview.appendChild(image);
        }, false);
        reader.readAsDataURL(file);
    }
}
```

二、通用弹框

弹窗部分与PC虽异曲同工但有所出入，样式上弹窗宽度统一为96%，并且优化了补

三、用到的第三方库

Zepto: 用于移动端代替jquery的一款更适合移动端需求的框架，较与jq
 validator: 轻量级的验证插件，原生js编写，兼容性良好，为满足开发需
 baguetteBox: 与pc一致
 lozad: 与pc一致
 swiper: 与pc一致

四、第三方代码

与 pc一致

五、输入框通用结构说明（author:石爱朵）

1、输入框通用结构

```

<div class="form-list">
  <div class="form-list_item form-list_item--all">
    <label class="form-label form-label--validator" for="
    <input class="form-input" type="text" name="email" id
  </div>
</div>
<!--
form-label: 表单标题class;
form-label--validator: 必填项特殊标记添加class;
form-list_item--all: input 框宽度为整屏幕时添加class
form-list_item--sub : input 框宽度为屏幕一半的宽度时添加class
-->

```

2、输入框右侧有apply 按钮结构的

```

<div class="form-list">
  <div class="form-list_item form-list_item--all position-rela
    <input class="form-input" type="text" name="reward-point
    <button class="button checkout-reward_btn">Apply</button
  </div>
</div>

```

注释：需要在之前输入框结构的基础上 **form-list_item class**上增加**class: position-relative**,并在该标签中添加**button**;

3、多选框通用结构

```

<div class="form-control">
  <input class="form-control_input form-control_input--checkout"
  <label class="form-control_label" for="checkout-save-address">S

```

4、单选框通用结构

```

<div class="form-control">
  <input class="form-control_input form-control_input--radio"
  <label class="form-control_label" for="checkout-mathod1">FREE SH
</div>

```

注释：多选框需要在type为checkbox的input标签上添加form-control_input、form-control_input--checkboxout两个class 单选框要在type为radio的input标签上添加form-control_input、form-control_input--radio两个class

- 后端介绍

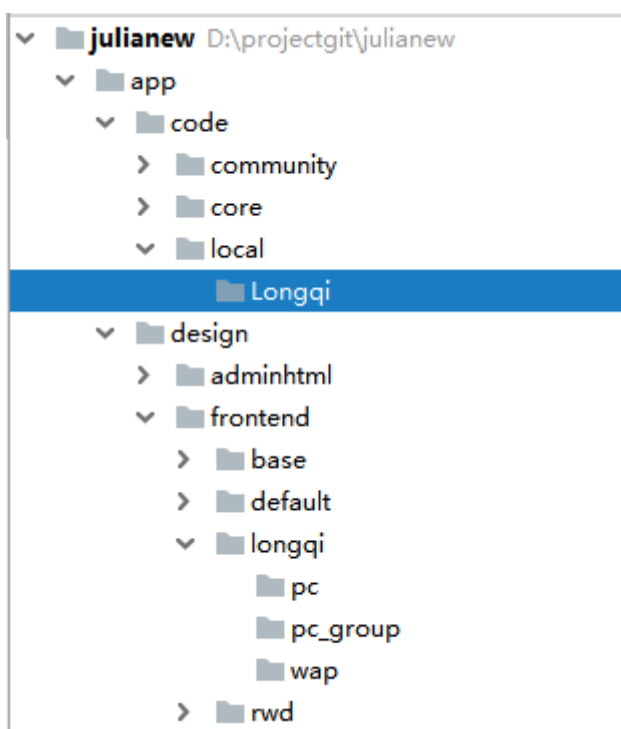
后端介绍

本次后端重构主要处理以下几个方面

- 1、移除冗余第三方插件
- 2、对评论模块进行大的调整
- 3、弃用lua，fpc缓存机制，启用varnish
- 4、核心流程均使用ajax异步加载机制，提高性能及用户体验

- 一、网站整体架构
- 二、注释规则
 - block,controllers,model,helper
 - xml
 - phtml
- 三、大的功能模块
- 四、缓存标签

一、网站整体架构



此次依然延续现在的mcbv 架构；对b，v端进行深度的优化调整
全面移除无用xml，v层全部重构

二、注释规则

block,controllers,model,helper

1、新建类必须注释，格式为

```
/**
 * @author 作者
 * @date 时间
 * @description 类描述
 */
```

2、创建的常量必须注释

3、方法必须注释，格式为

```
/**
 * @author 作者
 * @date 时间
 * @description 描述
 * @param $cart 变量描述
 * @return 返回值类型
 */
```

xml

所有新引入的xml必须注释，格式为

```
<!--引入说明-->
```

phtml

在页面添加的block块，引入getChildHtml,调用block里面的方法。。均必须写清楚注释，格式为

```
<?php
/**
 * 描述
 */
?>
```

三、大的功能模块

功能性开发需必须有文档说明

- 1、流程图，
- 2、数据字典，
- 3、核心代码及说明

四、缓存标签

首页---home
列表---product_list
详情---product_view
购物车--checkout
支付页面---firecheckout
用户中心---customer
其他通用--common
栏目促销折扣--catalog_promotion
评论缓存----customerreview

- 一、关闭的系统默认插件
- 二、使用的第三方插件
- 三、独立开始第三方插件
- 四、重构模块
 - 1、评论模块
 - 2、ajax异步完成
 - 3、支付页面
- 五、启用产品，分类flat模型
- 六、修改xml，支持底部移除css.js(功能暂未使用)
- 七、通过nginx-lua支持将图片动态修改webp格式
- 八、minify组件完成html压缩操作
- 九、varnish完成新的缓存机制
- 十、css.js同时支持min及常规两种形式
- 十一、开发中遇到的问题及解决方案（author:王佳康）
 - 1、xml布局文件
 - 2、首页整体调用调整
 - 3、调用指定栏目产品方法重构
 - 4、收藏产品红心标注
 - 5、配合varnish缓存机制配置
- - 6、翻页通过ajax异步加载
 - 7、手机端属性筛选

一、关闭的系统默认插件

```

1、配置文件关闭：
配置产品模块
Mage_Rss,
Mage_Authorizenet,
Mage_GiftMessage,
Mage_ConfigurableSwatches,
Mage_Downloadable,
Mage_Weee,
Phoenix_Moneybookers
2、后端关闭：
Mage_AdminNotification
Mage_Authorizenet
Mage_Backup
Mage_Bundle
Mage_ConfigurableSwatches
Mage_Downloadable
Mage_PaypalUk
Mage_Rss
Mage_Weee
Mage_XmlConnect
Phoenix_Moneybookers

```

二、使用的第三方插件

```

1、Firecheckout 需要较大调整
2、Mw(积分) 需要调整优化
3、Aschroder-- 邮件发送插件smtp
4、AW--blog
5、Inic--faq
6、Hetinfoway--video
7、aoe---cron
8、DigitalPianism--ajax login (保留功能)
9、RocketWeb--google shipping
10、Zestard--客户属性后台自定义
11、订单异步测试
12、Vstudio---menu(自定义导航)
13、Apptrian--minify
14、Fishpig--wordpress
15、Imsc--refersion
16、Inchoo--SocialConnect
17、Nexcessnet--Turpentine

```

三、独立开始第三方插件

- 1、wholesale
- 2、staylooks
- 3、luckydraw
- 4、paypal error
- 5、banner
- 6、sms

四、重构模块

1、评论模块

主要讲rating,移动到评论详情表,同时移除多余的字段
移除表review_author_ips,review_helpful,review_proscons,rview_proscons
Review_detail 移除字段: Cons,pros,body_type,sizing,location,age,

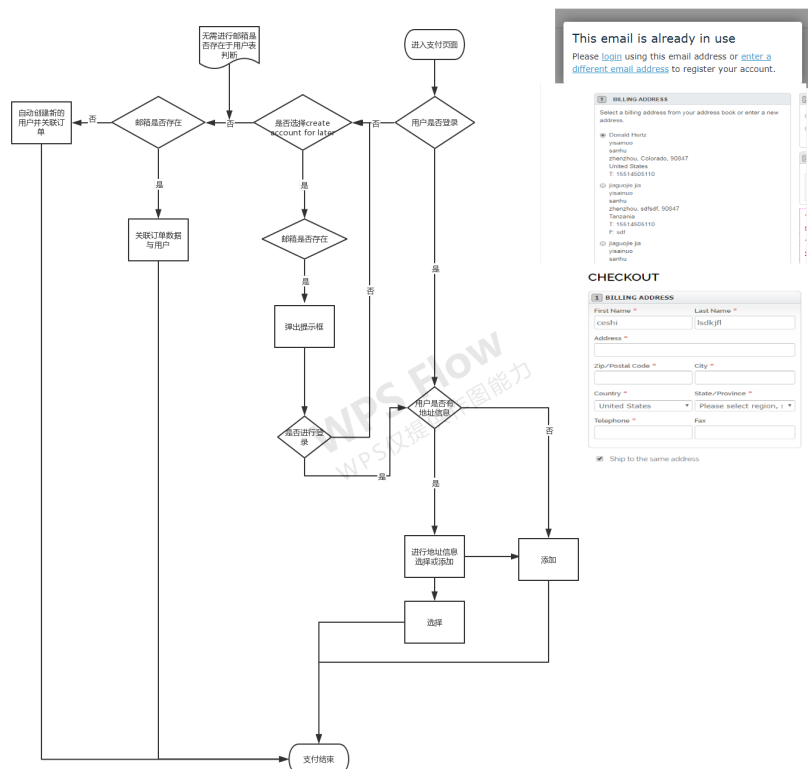
2、ajax异步完成

所有翻页,属性筛选,加购,购物车(删除,更新),支付页面

3、支付页面

支付页面的所有逻辑重新梳理重写(后端基于firecheckout方法)
移除字段 tm_field1 , tm_field2 , tm_field3, tm_field4, tm_field5,
移除表tm_orderattachment;

tm Additional 移除
firecheckout_geoip 功能移除
Address Verification (USPS) 功能移除
对应地址栏逻辑进行梳理:



五、启用产品，分类flat模型

六、修改xml，支持底部移除css,js(功能暂未使用)

七、通过nginx-lua支持将图片动态修改webp格式

八、minify组件完成html压缩操作

九、varnish完成新的缓存机制

十、css,js同时支持min及常规两种形式

十一、开发中遇到的问题及解决方案
(author:王佳康)

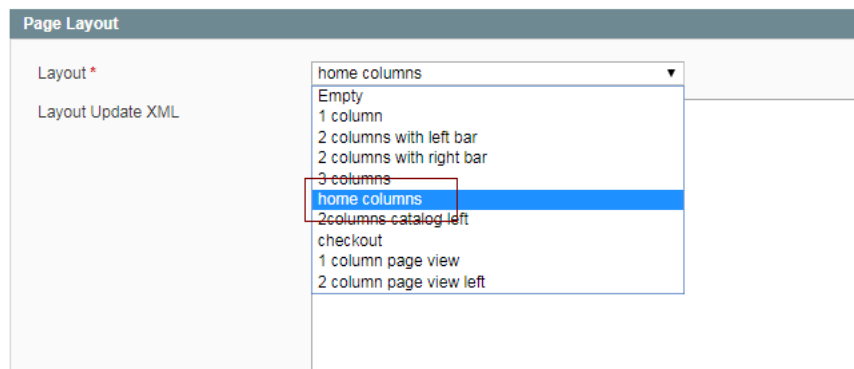
1、xml布局文件

添加新的栏目布局，如果需要可以在后台进行设置，需在xml中进行配置。

注：如果不需要在后台进行设置，则无需进行此配置，直接在xml中配置对应的布局模板即可。

D:\web\julianew\app\code\local\Longqi\Public\etc\config.xml

```
<page>
  <layouts>
    <home_columns module="page" translate="label">
      <label>home columns</label>
      <template>page/1column-home.phtml</template>
      <layout_handle>page_home_columns</layout_handle>
    </home_columns>
  </layouts>
</page>
```



2、首页整体调用调整

系统默认首页的内容是在后台cms->page中的单页填写的，因不利于后期维护，故将其重写至phtml页面中。

D:\web\julianew\app\design\frontend\longqi\pc\layout\local.xml



```

<cms_index_index>
<reference type="page/html" name="root" output="toHtml" template="page/1column-home.phtml">
<!-- 查看首页模板 -->
<block type="public/home" name="content" as="content" template="home/index.phtml">
<!-- banner -->
<block type="banner/banner" name="banner" template="home/banner.phtml" />
<!-- banner -->
<!-- 订阅 -->
<block type="newsletter/subscribe" name="newsletter" as="newsletter" template="newsletter/subscribe.phtml" />
<!-- 订阅 -->
</block>
</reference>
<reference name="head">
<!-- 首页swipercss 引入 -->
<action method="addItem"><type>skin_css</type><name>css/lib/swiper/1dangerous.swiper.css</name><params></action>
<!-- 首页swiper 引入 -->
<!-- 首页css引入 -->
<action method="addItem"><type>skin_css</type><name>css/longqi.home.css</name><params></action>
<!-- 首页css引入 -->
<!-- [if IE 9] -->
<action method="addItem"><type>skin_css</type><name>css/ie/indexPolyfill.css</name><params><!-- IE 9 --></action>
<!-- [if IE 9] -->
<!-- [if IE 9] -->
<action method="addItem"><type>skin_css</type><name>css/ie/polyfill.css</name><params><!-- IE 9 --></action>
<!-- [if IE 9] -->
<action method="addItem"><type>skin_js</type><name>js/lib/swiper/1dangerous.swiper.js</name><group>defer</group><params></action>
<!-- 首页js引入 -->
<action method="addItem"><type>skin_js</type><name>js/longqi.home.js</name><group>defer</group><params></action>
<!-- 首页js引入 -->
</reference>
<reference name="Footer1css">
</reference>
</cms_index_index>

```

3、调用指定栏目产品方法重构

系统默认在block中调取分类是没有数量限制的，自定义方法进行数量限制

D:\web\julianew\app\code\local\Mage\Catalog\Block\Product\List.

```

public function getProductCollection()
{
    if($this->getCategoryId()) {
        $_productCollection = Mage::getModel('catalog/category')->load($this->getCategoryId());
        if($_productCollection->getId()) {
            $_productCollection->getProductCollection()
                ->addAttributeToSelect('*')
                ->addAttributeToFilter('status', 1)
                ->addAttributeToFilter('visibility', 4)
                ->addAttributeToSort('position', 'ASC');
            if($this->getLimit()) {
                $_productCollection->setPageSize($this->getLimit());
            }
        }
        return $_productCollection;
    }
    return false;
}

```

4、收藏产品红心标注

登录情况下已收藏的产品红心的显示，因varnish缓存影响故不能用使用PHP对其判断，监听客户登录成功，将客户已收藏的产品ID存储到COOKIE中，然后在js中读取cookie进行红心展示。

```

public function resetCustomerWishlistCookie()
{
    /**
     * 配合心愿单产品红心使用 用户的心愿Id单存储cookie
     */
    $customer = Mage::getSingleton('model/class' 'customer/session')->getCustomer();
    $store_id = Mage::app()->getStore()->getId();
    $wishlist = Mage::getModel('model/class' 'wishlist/item')->getCollection();
    $wishlist->addFieldToSelect('product_id');
    $wishlist->addCustomerIdFilter($customer->getId());
    $wishlist->addFieldToFilter('store_id', $store_id);
    if($wishlist->getSize() > 0) {
        $productIds = array_column($wishlist->getData(), 'column' 'product_id');
        $productIds = implode('@##', $productIds);
        Mage::app()->getCookie()->set($name' wishlist_product_id', $productIds, $period, $path, $domain, $secure, $httponly);
    }
}

if (isLogin) {
    var wishlist_product_id = getCookie( cname: 'wishlist_product_id');
    if (wishlist_product_id != '' && wishlist_product_id != null) {
        var wishlist_product_id_arr = new Array();
        wishlist_product_id_arr = wishlist_product_id.split('-');
        for (var i = 0; i < wishlist_product_id_arr.length; i++) {
            className = '.wishlist_product_id_' + wishlist_product_id_arr[i];
            $(className).removeClass( name: 'icon-heart2').addClass('icon-heart1');
        }
    }
}
}

```

5、配合varnish缓存机制配置

因varnish缓存的缓存机制是全局缓存，然而一些不需要缓存的block或controller是需要要在xml中指定不进行缓存的，如头部登录状态，头部购物车。

```

<!--varnish 缓存block配置-->
<default>
    <reference name="top_user">
        <action method="setEsiOptions">
            <params>
                <access>private</access>
                <flush_events>
                    <persistent_session_expired/>
                </flush_events>
            </params>
        </action>
    </reference>
    <reference name="topcart">
        <action method="setEsiOptions">
            <params>
                <access>private</access>
                <flush_events>
                    <sales_quote_save_after/>
                </flush_events>
            </params>
        </action>
    </reference>

```

注：指定不缓存block的xml需放在default中进行配置

```

<firecheckout_index_index>
    <turpentine_cache_flag value="0"/>
</firecheckout_index_index>

<public_adminhtml_index_resetCatalogFilterLimit>
    <turpentine_cache_flag value="0"/>
</public_adminhtml_index_resetCatalogFilterLimit>

```

6、翻页通过ajax异步加载

分页数据改为ajax请求，原系统默认的分页链接不变改为ajax方式对其链接进行请求，在后台获取block将phtml内容转换为json格式进行返回。

注：因varnish缓存会根据链接进行缓存，所以ajax请求一定要添加参数与非a

```
/**
 * ajax请求分类列表数据
 * @param url
 */
window.ajaxCatalog = function(url) {
    openLoading();
    $.ajax( config: {
        url: url,
        type: 'GET',
        data: { ajaxcatalog: true },
        dataType: 'json',
        success: function(data) {
            closePopup('loading');
            if (data.status == 'success') {
                /**
                 * html 添加方式
                 */
                if (data.hasOwnProperty( v: 'add_manner')) {
                    if (data.add_manner == 'cover') {
                        /**
                         * 直接覆盖 replaceWith
                         */
                        $(' .sortlist').replaceWith(data.html);
                    }
                } else {
                    //默认追加
                    $(' .sortlist').html(data.html);
                }
            }
        }
    });
}

/**
 * ajax 异步请求处理
 */
if ($this->getRequest()->isAjax() && $this->getRequest()->isXmlHttpRequest() && $this->getRequest()->getParam( key: "ajaxcatalog")) {
    $response = array();
} else {
    $catalog_filter = $this->getLayout()->getBlock( name: 'catalog.filter')->toHtml(); // Generate catalog.filter
    $product_list_html = $this->getLayout()->getBlock( name: 'product_list')->toHtml(); // Generate product list
    $response['status'] = 'success';
    $response['html'] = $catalog_filter.$product_list_html;
}

$this->getResponse()->setHeader( name: 'Content-type', value: 'application/json');
$this->getResponse()->setBody(Mage::helper( name: 'core')->jsonEncode($response));
return;
```

7、手机端属性筛选

M端分类列表下拉属性筛选移至后台进行重置

产品通用配置

most popular 产品ID

5

▲ 只允许填写一个分类ID，区分店铺请认准PC还是WAP，注

most popular 产品显示数量

12

▲ 区分店铺请认准PC还是WAP，注意切换店铺，不填默认8

产品详情不显示优惠信息的分类

▲ 多个用逗号隔开

new in 产品分类ID

5

▲ 只允许填写一个分类ID，区分店铺请认准PC还是WAP，注

m端下拉属性筛选重置

Reset Catalog Filter Limit

```

protected function _getAddRowButtonHtml($title)
{
    $buttonBlock = $this->getElement()->getForm()->getParent()->getLayout()->createBlock('adminhtml/widget_button');
    $_websiteCode = $buttonBlock->getRequest()->getParam('website', null);
    $params = array();
    if (!empty($_websiteCode)) {
        $params['website'] = $_websiteCode;
    }
    // TODO: for real multi-store self-testing, the test button (and other configuration options)
    // should probably be set to show in website. Currently they are not.
    $url = Mage::helper('adminhtml')->getUrl('*/*/reset/index', $params);
    $buttonHtml = $this->getLayout()->createBlock('type: adminhtml/widget_button')
        ->setType('button')
        ->setLabel($this->__( $title ))
        ->setOnClick("window.location.href='".$url."'")
        ->toHtml();
    return $buttonHtml;
}

/**
 * 一级导航
 */
$categories = Mage::getModel('catalog/category')->getCollection()->addAttributeToSelect('*')->addIsAcl

foreach ($categories as $category) {
    $category->setData('store_id', 6);
    $categoryId = $category->getId();
    $layer = Mage::getModel('catalog/layer');
    $layer->setCurrentCategory($category);
    $attributes = $layer->getFilterableAttributes();
    $layer->setData('current_rate', 1);
    $layer->setData('store_id', 6);
    $filter_limit = explode(' ', $category->getCustomFilterName());
    if (empty($filter_limit)) continue;
    $attributeCodeArr = [];
    foreach ($attributes as $attribute) {
        if (in_array($attribute->getAttributeCode(), $filter_limit)) {
            $attributeCode = $attribute->getAttributeCode();
            $items = Mage::getModel('catalog/layer_filter_attribute')->setLayer($layer)->setStoreId(6);
            foreach ($items as $option) {
                $attributeCodeArr[$attributeCode]['value'][] = $option->getValue();
                $attributeCodeArr[$attributeCode]['label'][] = $option->getLabel();
                $attributeCodeArr[$attributeCode]['attribute_label'][] = $attribute->getFrontendLabel();
            }
        }
    }
    $this->insertLimitFilteres($categoryId, $attributeCodeArr);
}

```

- [varnish简单介绍](#)
- [varnish的具体使用及与站点的结合我们会有专门的介绍及分享](#)

varnish简单介绍

Varnish Cache是一个Web 应用加速器，也是一个知名的反向代理程序。你可！
我们此次修改为varnish的原因是

- 1、原有的lua缓存有各种限制（货币，不同用户组折扣，仅能通过url区分）
- 2、varnish 本身支持集群功能，为我们后续的扩展做准备
- 3、varnish同样缓存与内存中
- 4、varnsih的 vcl配置，及esi更加灵活

varnish的具体使用及与站点的结合我们会有专门的介绍及分享

- 数据库调整

数据库调整

1、移除评论插件相关字段

```
delete from eav_attribute where backend_model='detailedreview/ca
```

2、配合移除俄语站点

```
DELETE FROM `core_store` WHERE `core_store`.`store_id` = 5;  
DELETE FROM `core_store_group` WHERE `core_store_group`.`group_i  
DELETE from cms_page_store where store_id=5;
```

3、后端栏目报错:

```
DELETE from eav_attribute where attribute_code = "bs_menuadvance  
DELETE from eav_attribute where attribute_code = "cb_category_ty
```

4、后端报错（系统升级导致）

```
alter table sales_best sellers_aggregated_yearly add COLUMN produ  
alter table sales_best sellers_aggregated_monthly add COLUMN prod  
alter table sales_best sellers_aggregated_daily add COLUMN produc
```

5、移除无用表

```
DROP TABLE `permission_variable`;    #不一定执行  
DROP TABLE `permission_block`;      #不一定执行  
drop table `customer_flowpassword`;  
drop table `ordersyncquerunner_que`;  
drop table `vstudio_venu_menuitem`;  
drop table `vstudio_venu_menu`;
```

7、评论相关

```
DROP TABLE `review_author_ips`;  
DROP TABLE `review_helpful`;  
DROP TABLE `review_proscons`;  
DROP TABLE `review_proscons_store`;  
ALTER TABLE `review_detail` DROP `cons`;  
ALTER TABLE `review_detail` DROP `pros`;  
ALTER TABLE `review_detail` DROP `body_type`;  
ALTER TABLE `review_detail` DROP `sizing`;  
ALTER TABLE `review_detail` DROP `location`;  
ALTER TABLE `review_detail` DROP `age`;  
ALTER TABLE `review_detail` DROP `height`;
```

8、订单表操作

```
alter table sales_flat_order drop column tm_field1,drop column t  
drop table tm_orderattachment;
```

10、flat 表移除重新生成

```
drop table `catalog_category_flat_store_1`;  
drop table `catalog_category_flat_store_2`;  
drop table `catalog_category_flat_store_3`;  
drop table `catalog_category_flat_store_4`;  
drop table `catalog_category_flat_store_5`;  
drop table `catalog_category_flat_store_6`;  
drop table `catalog_product_flat_1`;  
drop table `catalog_product_flat_2`;  
drop table `catalog_product_flat_3`;
```

```
drop table `catalog_product_flat_4`;  
drop table `catalog_product_flat_5`;  
drop table `catalog_product_flat_6`;
```

11、关闭rss

```
UPDATE `core_config_data` SET `value` = '0' WHERE `core_config_d  
UPDATE `core_config_data` SET `value` = '0' WHERE `core_config_d  
UPDATE `core_config_data` SET `value` = '0' WHERE `core_config_d  
UPDATE `core_config_data` SET `value` = '0' WHERE `core_config_d  
UPDATE `core_config_data` SET `value` = '0' WHERE `core_config_d  
UPDATE `core_config_data` SET `value` = '0' WHERE `core_config_d  
UPDATE `core_config_data` SET `value` = '0' WHERE `core_config_d  
UPDATE `core_config_data` SET `value` = '0' WHERE `core_config_d
```

