### 站群系统开发手册

###### 一、指导思想

1. 分层思想：将应用分为不同层，M（模型层）,V(视图层),C（业务逻辑层），C（配置层）
2. 微服务重用思想，将之前单体应用拆分为各个小的微应用。具体表现为，之前Controller角色变化为服务组装器负责获取基础数据，并根据需求对各个小服务(Action)进行组装；之前各个模块应用拆分为小的Action，每个Action都拥有自己的MVC分层
3. 配置化思想，将系统数据及应用数据进行配置。具体表现为，前端资源配置化，路由重写配置化，依赖注入配置化，缓存依赖配置化，各个应用数据配置化
4. 标准化思想，对各层、程序进行标准化。具体表现为，ajax返回数据格式标准化，model层标准化，url生成表转化，图片展示标准化。
5. 热插拔、插件化思想，以事件机制实现插件化

###### 二、本系统需要解决的问题

1. 各个功能重复开发，不能重用
2. 功能相互耦合，牵一发而动全身
3. 不能做到部分相同部分不同的效果
4. 无法插件化，不能做到功能的热插拔

###### 三、系统目录结果说明

/@runtime *#运行时文件，包括logs和cache*

/logs *#日志文件 格式为:*

*#/logs/camfere-com/年月/日/\*.log*

/cache *#缓存文件目录*

/common *#系统公共目录*

public.php *#入口文件*

/widgets *#小部件目录,注意小部件视图位于:*

*# /common/fragments/widgets 目录下*

/plugins *#插件目录*

/test *#test插件*

Plugin.php *#插件声明类*

/controllers *#插件控制器*

/models *#插件model*

/views *#插件视图文件*

/... *#其他插件*

/models *#公用model层*

/helpers *#助手类文件目录*

/fragments *#视图片段，小部件视图及action视图*

/widgets *#小部件视图片段*

/... *#action视图片段*

/events *#事件*

/environments *#配置文件(三种环境test,dev,prod)*

/common *#公共配置*

/config *#配置目录*

/main.php *#生产环境主配置*

/main-dev.php *#开发环境主配置*

/main-test.php *#测试环境主配置*

/parmas.php *#生产环境参数配置*

/params-dev.php *#开发环境参数配置*

/params-test.php *#测试环境参数配置*

/camfere-com *#网站camfere-com 配置文件*

/config *#配置目录*

/main-local.php *#生产环境主配置*

/main-local-dev.php *#开发环境主配置*

/main-local-test.php *#测试环境主配置*

/parmas-local.php *#生产环境参数配置*

/params-local-dev.php *#开发环境参数配置*

/params-local-test.php *#测试环境参数配置*

/... *#其他网站配置*

/docs *#文档 说明书*

/components *#系统核心组件*

/behaviors *#行为*

/actions *#各个action*

/camfere-com *#应用网站目录*

/controllers *#应用控制器目录*

/models *#应用models*

/views *#应用视图*

/modules *#应用模块*

/... *#其他网站*

/vendor *#第三方应用及框架文件*

###### 四、配置文件说明

**(注意:系统会根据php.ini中设置的环境变量 env及本地目录中的'xxx.txt'文件匹配，如果php.ini中配置了env则PHP采用其配置，否则将匹配本地的txt文件。php.ini env有三个值，分别为'uat'、'prod'、'dev'，分别代表'测试环境','生产环境','开发环境'。txt文件有三种,分别为'test.txt'、'prod.txt'、'test.txt'，分别代表'测试环境','生产环境','开发环境')**

1. 主配置文件说明(main.php main-dev.php main-test.php main-local.php main-local-test.php main-local-dev.php)

**return** [

'id' => DOMAIN,*//应用id 如cafamere-com*

'basePath' => SYS\_PATH . DOMAIN,*//应用的基础目录*

'vendorPath' => SYS\_VENDOR\_PATH,*//verder目录*

'runtimePath' => SYS\_RUNTIME\_PATH,*//runtime 目录*

'controllerNamespace' => 'app\controllers',*//应用controller的命名空间*

'bootstrap' => [*//启动阶段加载的组件*

'plugin', *//插件*

'log',*//日志*

'diBox',*//依赖注入容器*

],

'components' => [

'cacheCleaner' => [*//缓存清理组件*

'class' => 'common\components\ComCacheCleaner',

'cacheVersions' => [*//各个缓存的版本 'key' => '版本号'*

'vdvert' => 'V1.0.6',

'categoryBg' => 'V1.0.6',

'allCategories' => 'V1.0.6',

],

],

'diBox' => [*//依赖注入容器管理*

'class' => 'common\components\ComDiBox',

'box' => [*//需要加入到容器中的类、对象 '对象名称' => ['def' => '文件路径']*

'CountryModel' => [

'def' => [

'class' => 'common\models\Country',

],

],

],

],

'plugin' => [*//插件管理*

'class' => 'common\components\ComPlugin',

'lists' => [

'facebookLogin' => [ *//插件名称*

'available' => 'camfere.com', *//可用网站*

'sort' => 0 *//排序*

],

'googleLogin' => [

'available' => 'camfere.com',

'sort' => 1

],

],

],

'urlManager' => [*//url模块管理*

'class' => 'common\components\AppUrlManager',

'enablePrettyUrl' => **true**,

'rules' => [*//各个重写规则*

*/\*+++首页重写+++\*/*

'index.html' => 'default/index',

'/' => 'default/index',

*/\*+++首页重写+++\*/*

*/\*+++文章页重写+++\*/*

'/site-map.html' => 'cms/sitemap',

'<url:[a-zA-Z-&\s]+>.html' => 'cms/details',

*/\*+++文章页重写+++\*/*

],

],

'assetManager' => [*//资源文件管理*

'bundles' => [

],

],

'reqApi' => [*//接口管理*

'class' => 'common\components\ComApi',

'sites' => [*//API站点*

'baseApi' => ['host' => 'http://base.api.tomtop.com'],

'memberApi' => ['host' => 'http://member.api.tomtop.com'],

'productApi' => ['host' => 'http://product.api.tomtop.com'],

'loyaltyApi' => ['host' => 'http://loyalty.api.tomtop.com'],

'cartApi' => ['host' => 'http://cart.tomtop.com'],

'shippingApi' => ['host' => 'http://logistics.api.tomtop.com'],

'advertApi' => ['host' => 'http://advert.api.tomtop.com'],

'ttApi' => ['host' => 'http://www.tomtop.com']

],

'apis' => [*//各种API {#xxx}部分代表api站点 如{#baseApi} 代表baseApi的主机*

*//baseApi*

'getCountries' => [*//获取国家接口*

'url' => '{#baseApi#}/base/country/v1',

],

'getCurrencies' => [*//获取货币接口*

'url' => '{#baseApi#}/base/currency/v1',

],

],

],

'cache' => [*//默认使用文件缓存*

'class' => 'common\components\AppFileCache',

'serializer' => ['igbinary\_serialize', 'igbinary\_unserialize'],*//序列化|反序列化 函数 必须开启php\_igbinary扩展*

'cachePath' => '@runtime/cache/' . DOMAIN, *//文件缓存目录*

],

*//国际化*

'i18n' => [

'translations' => [

'\*' => [

'class' => 'yii\i18n\PhpMessageSource',

'sourceLanguage' => 'en',

'fileMap' => [],

],

],

],

'redis' => [*//redis缓存管理*

'class' => 'yii\redis\Cache',

'serializer' => ['json\_encode', 'json\_decode'],

'redis' => [

'hostname' => 'redis01.tomtop.com',

'port' => 6379,

'database' => 14,

'password'=>'jg8ZxxT1#lmc',

],

],

'view' => [

'class' => 'common\components\AppView',

],

'mailer' => [

'class' => 'yii\swiftmailer\Mailer',

'viewPath' => '@common/mail',

*// send all mails to a file by default. You have to set*

*// 'useFileTransport' to false and configure a transport*

*// for the mailer to send real emails.*

'useFileTransport' => **true**,

],

'log' => [*//日志系统*

'targets' => [

'file' => [

'class' => 'common\components\AppFileTarget',

'levels' => ['trace', 'info', 'error', 'warning', 'profile'],

*// 'categories' => ['yii\\*'],*

],

],

],

],

'alias' => [*//别名*

],

1. 参数配置文件说明 参数配置文件均已'params'开头，在程序中使用如下代码访问

Yii::$app->params['参数名']

1. 配置加载及解析原理说明 当运行一个网站后，会加载公用配置和相关网站的配置，比如在生产环境运行'[camfere.com](http://camfere.com)'这个网站，分别会加载如下文件：

/common/environments/common/config/main.php /common/environments/common/config/params.php /common/environments/camfere-com/config/main-local.php /common/environments/camfere-com/config/params-local.php

且后面的配置文件会以'merge'的形式覆盖前面的配置，比如在'/common/environments/common/config/main.php'中配置了'cache'属性，若在'/common/environments/camfere-com/config/main-local.php'也配置了'cache'属性,则后面的将会以'merge'的形式覆盖前面的配置

###### 五、各个目录及功能命名空间

| **目录名称** | **对应的命名空间** |
| --- | --- |
| /common | /common |
| /common/widgets | /common/widgets |
| /common/models | /common/models |
| /common/helpers | /common/helpers |
| /common/events | /common/events |
| /common/components | /common/components |
| /common/behaviors | /common/behaviors |
| /common/actions | /common/actions |
| /camfere-com | /app |
| /camfere-com/controllers | /app/controllers |
| /camfere-com/views | /app/views |
| /camfere-com/models | /app/models |
| ... | ... |
| /common/pugins/lab | /pugins/puginLab |
| /common/pugins/test | /pugins/puginTest |
| ... | ... |

###### 六、常用类方法说明

//生成链接相关

Url::home();//获取主页URL

Url::current();//当前页码URL

Url::toRoute(['module/controller/action', 'params1' => '', ..., 'paramsN' => '']);//生成URL

TTHelper::getProductUrl($urlPrefix);//生成商品URL

TTHelper::staticPrefix();//CDN域名前缀

Controller::redrect(['module/controller/action', 'params1' => '', ..., 'paramsN' => '']);//重定向

//Json相关

Json::encode();

Json::decode();

//cookie相关

TTHelper::getCookie($name, $defaultValue = null);//获取cookie

TTHelper::setCookie($name, $value, $expire = 0, $domain = '', $path = '/', $secure = false, $httpOnly = false);//设置cookie

TTHelper::unsetCookie($name, $domain = '');//删除cookie

//语言相关

TTHelper::getLangPkg();//获取语言包

TTHelper::getSiteLang($key, $sitePrefix = '');//或者站点指定语言

//请求相关

Yii::$app->request->get('xxx');//获取get类型的参数

Yii::$app->request->post('xxx');//获取post类型的参数

Controoler::resAjax(['ret' => 1, 'data' => [], 'msg' => '']);

//网站SEO相关

Controller::seoMeta(['title' => '', 'description' => '', 'keywords' => '']);//生成SEO信息

//渲染视图相关

Action::renderPartial('viewFile', 'params');//Action渲染视图片段

Controller::render('viewFile', 'params');//controller渲染整个视图

Controller::renderPartial('viewFile', 'params');//controller渲染视图片段

//配置相关

Yii::$app->params['键名'];//获取params配置

//缓存相关

Cache::set($key, $value, $duration = 0, $dependency = null);//设置缓存

Cache::get($key, $defaultValue);//获取缓存

//Api请求相关

AppCurl::get($api, $queryData = **array**());//get请求

AppCurl::head($api);//head请求

AppCurl::post($api, $queryData = '', $jsonEncode = true);//post请求

AppCurl::put($api, $queryData = '', $jsonEncode = true);//put请求

AppCulr::delete($api, $raw = true);//**delete**请求

###### 七、规范说明

1. 'vendor'目录为第三方应用程序目录，凡第三方的开源程序必须放到该目录下面，且该目录平时push权限关闭
2. 各个层分明，禁止在'view'层调用'model'层属性/方法，需通过'controller'层以参数形式传递给'view'层
3. 禁止在'controller'层书写复杂逻辑，将单个应用变大的十分庞大。应将应用拆分为各个小的'Action',然后在'controller'层进行组装。例如如下程序

**<?php**

**namespace** **app**\**controllers**;

**use** **Yii**;

**use** **yii**\**helpers**\**Url**;

*/\*\**

*\* 默认控制器 首页入口*

*\*/*

**class** **DefaultController** **extends** **BaseController**

{

*//...*

*/\*\**

*\* 首页.*

*\**

*\* @return mixed*

*\*/*

**public** **function** **actionIndex**()

{

*//seo信息*

$this->seoMeta(

[

'title' => 'Action Camera,360 Panorama Camera, Flashes, Tripod and Photography Acc | Camfere.com',

'description' => 'Shop for Action Camera,360 Degree Camera, Camcorders, Flashes, Tripod, and Camera Accessories at Bargain Price | Camfere.com',

'keywords' => 'action camera,360 degree camera,360 panorama camera,VR camera,digital camcorder,flashes,speedlite,ringlight,tripod,monopod,digital photo frame,Binoculars,Telescopes,Mobile Photography Accessories'

]

);

**return** $this->render(

'index',

[

*//sliderBanners 的功能*

'sliderBanners' => (**new** \common\actions\index\SliderBannerAction($this))->run(),

*//rightBanners 的功能*

'rightBanners' => (**new** \common\actions\index\RightBannerAction($this))->run(),

*//newArrivals 的功能*

'newArrivals' => (**new** \common\actions\index\NewArrivalsAction($this))->run(),

*//featured 的功能*

'featured' => (**new** \common\actions\index\FeaturedAction($this))->run(['cpath' => '', 'depotName' => 1])

]

);

}

*//...*

}

1. 凡是涉及到接口的model层，必须放在'/common/models'目录下
2. 所有的model层返回的数据必须是数组，且拥有三个元素, 分别是'ret'|'data'|'msg',如：

**<?php**

*/\*\**

*\* 购物车类*

*\**

*\* @author caoxl*

*\*/*

**namespace** **common**\**models**;

**use** **Yii**;

**use** **common**\**components**\**AppModel**;

**use** **common**\**components**\**AppCurl**;

**use** **yii**\**helpers**\**Json**;

**class** **Cart** **extends** **AppModel**

{

*//...*

*/\*\**

*\* 获取购物车总共商品的数量*

*\**

*\* @return array*

*\*/*

**public** **function** **getShoppingCartTotalNumber**()

{

$shoppingCartHistory = Yii::$container->get('TTHelper')->getCookie(Yii::$app->params['cartCookiesName']);

$cartProductNumber = 0;

**if**(!**empty**($shoppingCartHistory))

{

$shoppingCart = Json::decode($shoppingCartHistory);

**foreach**($shoppingCart **as** $key => $value)

{

$cartProductNumber += count($value);

}

}

**return** ['ret' => 1, 'data' => $cartProductNumber, 'msg' => ''];

}

*//...*

}

1. 所有的ajax请求必须使用控制器的'resAjax'方法，**禁止使用'exit'函数强制退出**，且必须返回三个元素的JSON，分别是'ret'|'data'|'msg',如

**<?php**

**namespace** **app**\**controllers**;

**use** **Yii**;

**use** **yii**\**helpers**\**Json**;

**use** **common**\**actions**\**detail**\**AttrsAction**;

**use** **common**\**actions**\**detail**\**ReviewsAction**;

**use** **common**\**actions**\**detail**\**HotEventAction**;

**use** **common**\**actions**\**detail**\**TabsAction**;

**use** **common**\**actions**\**detail**\**ImgZoomAction**;

**use** **common**\**actions**\**detail**\**StorageAction**;

*/\*\**

*\* 分类控制器*

*\*/*

**class** **DetailController** **extends** **BaseController**

{

*//...*

*/\*\**

*\* 异步获取商品收藏数*

*\**

*\* @return string json*

*\*/*

**public** **function** **actionAjaxfavorites**(){

$listingId = Yii::$app->request->get('listingId');

$re = Yii::$container->get('ProductModel')->getFavoritesNumber($listingId);

$data = ['ret' => $re['ret'], 'data' => []];

$re['ret'] == 1 && $data['data']['client'] = $re['collectCount'];

**return** $this->resAjax($data);

}

*//...*

}

1. 禁止在应用中将各种url写死或者自行组装，必须使用'Url::toRoute','Url::home','TTHelper->getProductUrl','Controller::redirect'等方法，如：

Url::toRoute(['cate/index', 'cpath' => $category3['cpath']]); //生成url

$productUrl = $TTHelper->getProductUrl($data['url']);//获取商品URL

$index = Url::home();//获取主页地址

$this->redirect(['member/userlogin','backUrl'=>$currentUrl]);//跳转

1. 禁止将图片尺寸写死，必须使用配置形式,以"Yii:$ app->params['xxxWidth']"和"Yii:$ app->params['xxxHeight']"方式获取
2. 禁止在视图文件中直接注册资源文件，必须使用"AppAsset::register($this, 'index');"方式进行注册，如：

**<?php**

**use** **common**\**components**\**AppAsset**;

*//注册资源文件*

AppAsset::register($this, 'index');*//此处的`index`便是配置中资源文件的键名*

**?>**

1. 如果要渲染视图片段请使用'renderPartial'函数，禁止将控制器的'layout'设置为'false'，如:

$banners = Yii::$container->get('AdvertModel')->getAdvert('HOME','BANNER-TOP');

$banners = $banners['ret'] == 1 ? $banners['data'] : [];

*#渲染视图片段的*

**return** $this->renderPartial(

'//layouts/topBanners',

[

'banners' => $banners

]

);

1. Action寻找视图的规则为先在各个应用的目录下找，然后到'/common/fragments'目录下寻找
2. 如果要使用语言包，必须使用'$TTHelper->getSiteLang'函数获取，禁止自行组装或者写死，如：

$TTHelper->getSiteLang('account.reviewsList')；

1. 需要获取CDN的域名，禁止写死，必须使用'$TTHelper->staticPrefix()'获取，例如：

<img src="**<?**=$TTHelper->staticPrefix() **?>**/icon/logo.png" />

1. 请求接口必须使用'AppCurl'类，该方法有'get'|'post'|'put'|'head'|'delete'等方法，例如：

$appCurl = **new** AppCurl;

*#该函数有两个参数 前一个为数组 后一个也为数组*

$re = $appCurl->get(['api' => 'getArticleDetails'], ['url' => $url]);

1. 凡是涉及到公用的'model'|'action'|'widget'必须将注释写细致，要考虑到他人使用的方便
2. 凡是公用且会多次使用的类都放到'diBox'中，禁止多次生成对象，如:

$TTHelper = Yii::$container->get('TTHelper');

1. 所有项目文档说明请将文档文件放到'/common/docs'目录下