教育平台及WEB产品开发框架

**开发说明文档**

|  |  |  |
| --- | --- | --- |
| **撰写人** | **版本** | **最后更新时间** |
| 方益文 | V1.0 | 2014-3-5 |
|  |  |  |
|  |  |  |
|  |  |  |

1. **关于开发框架**

教育产品框架是基于ThinkPHP框架进行开发，已对框架文件进行了修改，请不要使用ThinkPHP新版本进行更新（切记）

* 1. **对ThinkPHP的修改：**
     1. **文件：/ThinkPHP/Lib/Core/View.class.php**
* 修改parseTemplate方法

增强模板定位功能，增加父类分组功能

当本分组模板不存在时，自动到父类分组获取，配合2.1点

要使用此功能，必须通过配置C('PARENT\_MODULE')设置父类分组，

* + 1. **文件：/ThinkPHP/Common/common.php**
* 修改T函数

增强模板继承及导入功能，增加父类分组功能

当导入的模板或继承的父类模版不存在时，自动到父类分组获取,配合1.1点

要使用此功能，必须通过配置C('PARENT\_MODULE')设置父类分组。

* + 1. **文件：/ThinkPHP/Extend/Library/ORG/Net/UploadFile.class.php**

从旧版本中导入此文件，以支持文件上传并兼容老版本

* + 1. **文件：/ThinkPHP/Extend/Library/ORG/Util/Image.class.php**

从旧版本中导入此文件，以支持文件上传并兼容老版本

* + 1. **文件/ThinkPHP/Extend/Library/ORG/Util/Page.class.php**
* 修改分页链接样式，支持图片按钮链接
  + 1. **文件/ThinkPHP/common/functions.php**
* 修改S函数

1. **编码约定**
   1. **命名规则**

* 数据库表名：小写及下划线（产品KEY\_功能名)，如：center\_product代表平台的产品列表；center\_product\_config代表平台的产品配制表
* 数据表字段名：首字母小写的驼峰写法,如：proKey代表产品KEY；addTime代表添加时间
* ./Application下的目录名：以首字母大写
* ./Public下的目录名：小写
* 以类的形式存在的，文件名以首字母大写的驼峰写法，如CommonController.class.php
* 以非类形式存在的，文件名以小写加下划线方式，如config.php
* 方法名以首字母小写的驼峰写法
* 函数以小写字母加下划线方式写法
* 前端标签ID，使用小写字母及下划线
* 前端CSS样式名：使用首字母小写的驼峰法
* JS函数以首字母小写的驼峰写法
  1. **资源文件存放**
* 根据Public/static/下面的目录分类进行存放
* 如全局共用存放于common/common\_v1下
* 如父类共用存放于common/sd\_v1或common/hd\_v1下
* 业务分组存放于theme/xxx\_v1下
* 图片资源请按产品目录存放
* 对于css和js，除了公用的以外，请直接在模版中实现

1. **目录结构**

Application 程序主体原码

Common 公共分组（公共配置、核心逻辑、数据模型,不可直接访问）

area\_XX适配分组 地区适配分组，以适配原则划分，可直接访问

Sd 父类分组 标清版本父类分组，不可直接访问

Hd 父类分组 高情版本父类分组，不可直接访问

System 系统管理分组，可以直接访问

Public 站点入口（包括静态资源、上传文件等等）（可以直接访问）

logs 日志文件日录

upfiles 管理后台上传的资源文件目录

tempfiles 开发临时存放资源文件，部署时删除，开发时代替upfiles

static 静态资源目录

common 公共资源文件目录

common\_vx 通用资源文件目录

images 图片，继续按产品分类

css 样式表

js JS

hd\_vx 高清父类分组文件目录，子目录参与common\_vx

sd\_vx 标清父类分组文件目录，子目录参与common\_vx

theme 业务分组文件目录

90010\_sd\_vx 地区适配分组文件目录，子目录参与common\_vx

Runtime 运行时库 （不可直接访问）

ThinkPHP Thinkphp框架文件（不可直接访问）

1. **关于分组**
   1. **地区适配分组**

如无特殊适配需求，则此分组中只需要以下二个文件

Conf/config.php (存放本分组的配制信息）

Controller/EmptyController.class.php (空操作类，直接继承公共分组的CommonBaseController即可，无须任何其它代码

必须在配置文件中指定父类分组，配制项为：PARENT\_MODUL

如存在其它特殊适配需求，则从对应父类分组中Copy相应Controller或View文件后，只保留需要修改相应的act操作方法，不存在的操作及模板会自动从父类分组中调用

* 1. **父类分组**

用于标清及高清版本的通用分组，这样无须在适配分组中实现功能，以减少后期适配及修改开发工作量

* 1. **分组加载流程**

基于ThinkPHP正常流程上，当访问的适配分组不存在相应页面，则自动调用父类分组页面，父类分组页面也不存在时则报错。

* 1. **部署时有效分组**

某个地区适配分组 + 父类分组 + 公共分组 + 管理分组

1. **缓存**
   1. **创建及更新缓存**

以文件缓存为主，需要缓存的数据在相应Logic类里加上updateCache方法

updateCache有两种情况：

a、无参数：该类只有一个缓存

b、$exName参数：表示该类有多个缓存，$exName指缓存关键字后缀

自动创建缓存：当调用读取缓存函数时，如不存在会自动创建该缓存；

手动更新缓存：在更改数据时，手动调用updateCache进行更新缓存

批量更新缓存：使用后台的批量更新缓存功能，必须请在/System分组配制config.php文件内CACHE\_LIST配制项中添加相关逻辑类名

* 1. **读取缓存：**

使用get\_cache($key)函数， 如：

get\_cache('SysPro')，一个缓存模式,相对应的会自动调用SysPro->updateCache()方法创建缓存

get\_cache('GasstationClass\_1')，多个缓存模式,相对应的会自动调用GasstationClass->updateCache('1')方法创建缓存

配置高清时候再每个配置文件里面加了'DATA\_CACHE\_SUBDIR'=>true,'DATA\_PATH\_LEVEL'=>2, 正式部署去掉

1. **配置**
   1. **添加配制**

在/Application/Common/Conf/下添加配置模板，然后在后台添加即可

* 1. **读取配制**

get\_pro\_config\_content($key)函数：读取关键字为$key的配置项内容(content)

* 1. **读取优先规则**

缓存 > 数据库 > 模板。如果前一级别不存在，则使用后一级别。

* 1. **配制模板格式**

请参考：/Application/Common/Conf/p\_config.php文件

1. **附件上传**
   1. **上传文件**

**调用JS函数：upfileDialog(fileType,thumb,isMore,isSel,textId,datagrid)**

必选参数：fileType-文件类型(1-image,2-j2me)具体参考center\_upfile配置文件，

可选参数：thumb-缩略图，isMore-[0|1]是否可多文件上传，isSel-[0|1]是否可选择服务器端文件，textId-接收结果ID，datagrid-列表ID

缩略图：只对文件类型1-image有效,格式如：后缀1=宽1\*高1,后缀2=宽2\*高2

* 1. **读取文件：**

**调用函数get\_upfile\_url($url,$sufix='',$path='')**

$url－上传文件时返回的相对保存路径

$sufix－一般只用于读取某个缩略图，值为上传文件时的缩略图后缀

$path － 文件访问目录，默认UPFILE\_PATH常量

1. **积分**
2. **消息框**
   1. **浮动消息框（屏幕中央显示，3秒后自动关闭）**

JS方式：showFloatMessage(msg)

PHP方式：

a、向模板传递变量值：$floatMessage = 'content'

b、调用方法：$this->addFloatMessage(msg)，必须继承CommonBaseController控制器基类

* 1. **页面消息框**

只支持PHP方式

调用方法：$this->showMessage($message,$icon,$buttons,$template)

必须继承CommonBaseController控制器基类

1. **统计PV**

调用函数function savePv($pvType,$paramId='',$pageRemak='',$resKey='',$buttonKey='')

1. **前端开发**
   1. **前端路径变量**

\_\_COMMON\_\_ /static/common/common\_vx

\_\_HD\_\_ /static/common/hd\_vx

\_\_SD\_\_ /static/common/sd\_vx

\_\_THEME\_\_ /static/theme/XXXX\_vx

* 1. **前端公共JS调用**
* 快捷按键：直接在模板文件中定义keyCodes[按键代码] = "js代码";
* 获取元素：$()函数，

ID方式：$('#id')；

标签名称方式：$('name'); 标签名称方式返回的是数组，使用方式如：$('name')[0]

* 添加window.onload事件：addLoadEvent(t);t参数为函数
* 更换图片：changeImage(id,src)
  1. **默认焦点框**

将要设置默认焦点框的对象样式设置成“focus”即可

* 1. **默认返回**

将返回按钮或对象的ID设置为“a\_back”并设置访问地址。

如果页面上不希望显示，可以将其设置display:none进行隐藏

每个页面都必须要有该返回对象，不管是否显示，也无需再设置该操作的快捷按键代码