**1.文件夹规划**

所有文件放在APP\SOME文件夹下，其中控制器为trait，其他被该控制器use，不为trait。

分为core、controller、Services、library4个文件夹，lang、conf放在core里。视图文件仍在laravel自带的view中不变

**2.全局过滤器**

（1）检查是否安装



（2）定义常量

1）路径常量

SOME\_ROOT: app()->path() . ‘/some/’

各个文件夹

**3.定义工具类Tool.php**

load\_config()

**4.一些替换**

（1）mca

mca依然通过list($m, $c, $a) = array\_pad(explode('.', Route::currentRouteName()), 3, '');解析。之前用namespace判断是否为admin，member不再用了，此时用m代替

**5.页面静态化**

均不考虑，以后看情况补充。

**6.模板**

鉴于laravel的前端资源文件均在public里面，因此，仍然设置模板文件。将前端模板文件放在views文件夹下，不需要另建子文件夹，前端资源文件放在public/assets/下，也不需要另建子文件夹，因为就那些文件

（1）上传模板时，先检查是否有对应的文件，分离，分别保存

（2）需要定义VIEW\_DIR用于扫描视图文件夹，定义ASSETS\_DIR用于上传前端文件

**7.功能规划**

（1）去掉站点功能

（2）去掉自定义url、可选静态化、回调方法、等功能

（3）去掉页面静态化

（4）去掉admin后缀选择

（5）去掉插件

【特别注意】

暂时关闭csrf检查

## 系统配置

## 目录

1.扩展菜单

后台左侧显示的目录中，会加上用户自定义的扩展项目，也就成了目录，是通过位于后台控制器基类的optionMenu中调用sysMenu方法实现，加上表单扩展+会员扩展

2.用户自定义菜单

在optionMenu中，简单的加上name和对应的url即可

## 权限

1.投稿权限

在栏目中设置，在后台控制器基类的adminPost()方法中判断



2.菜单操作权限

在后台控制器基类的checkUserAuth中判断，通过判断url是否在角色所拥有的url数组中

## 工具

load\_config：加载配置文件