

Linux平台PHP服务端开发——

第十二讲 WCMS – API设计

WCMS：支持推送功能的CMS

- 依靠Swoole实现Websocket协议的服务端推送功能。
- 使用Medoo扩展操作数据库。
- 使用Slim框架开发API。
- composer搭建自动加载目录结构。
- 设计方式：Medoo+Slim+composer自动加载+Swoole仅仅是实现API，前端页面是完全分离的。可以部署在其他目录或者是其他服务器。

初步实现的功能

- 后台管理员：登录，root管理员其他管理员账户。
- 内容：创建，删除，更新，获取内容，搜索。
- 用户：注册，登录，查看，收藏。
- ！对内容的操作，用户仅仅是查看收藏，管理员具备修改功这里体现的增删改查并不是通常框架所提供的基础操作，那些操作无法满足需求。

API设计遇到的问题

- 由于同源策略，AJAX只能获取同一个域的数据。
- 什么是同源：两个网页的端口，域名，协议相同则是同源。
- 同源策略是一个实现约定，而不是技术限制，主要是考虑到安全问题，防止有恶意脚本获取其他网页的cookies等数据。由于同源策略，AJAX也无法跨域访问其他数据。
- 浏览器方面会进行检测并确定是不是同源，是否可以请求。
- 所以，设计API的时候，如果考虑到同源策略，前端页面就只能部署在和API同一个域名同一个端口下，使用相同的协议（HTTP或者是HTTPS）。

如何解决跨域访问

- 通常解决跨域使用的是JSONP方法，jQuery也提供了接口，但是JSONP只能发起GET请求。实际JSONP的本质就是使用<script src="****"></script>的方式解决的，<script>的方式是不受同源策略限制的。但是功能有限。
- CORS：跨源资源分享（Cross-Origin Resource Sharing）的缩写，W3C标准，这个是解决跨域请求的根本方法。
- 使用CORS解决AJAX跨域请求：PHP返回请求的头信息加上Access-Control-Allow-Origin:*
- *表示所有域名都可以访问，如果是具体的域名则写成：
Access-Control-Allow-Origin:a.com ; b.com
- 多个域名用；分隔。

目录组织结构

- 目录结构与功能：

| | |
|-----------|--------------------------|
| vendor/ | composer扩展安装目录 |
| model/ | 数据模型类目录，由action目录下的类调用 |
| action/ | 逻辑层，实际接口通过此目录的文件执行 |
| lib/ | 库函数目录，一些通用函数 |
| start.php | 接口定义文件，入口文件会引入此文件执行， |
| public/ | API根目录，其中的index.php是入口文件 |
| core | 一些通用的核心类文件 |

composer自动加载目录

- composer.json加入一下配置：

```
"autoload":{
    "files":["./lib/common.php"],
    "psr-4":{
        "mcore\\":"core",
        "model\\":"model",
        "action\\":"action"
    }
}
```

- 运行：composer.phar dump-autoload

API路由控制

- start.php文件编写代码作为测试接口：

```
use \mcore\DB;
$app = new \Slim\App;
$app->get('/a',
    function ($request, $response) {
        return $response->withStatus(200)
            ->withHeader('Access-Control-Allow-Origin', '*')
            ->write('Hello, PHP');
    }
);
$app->post('/b',
    function ($request, $response) {
        return $response->withStatus(200)
            ->withHeader('Access-Control-Allow-Origin', '*')
            ->write(json_encode($request->getParsedBody()));
    }
);
$app->run();
```


API测试

- 使用本地页面运行AJAX请求接口返回结果，F12查看调试信息，在响应头信息查看到 Access-Control-Allow-Origin

