

# TARS 前台开发文档

## 开始

1. 部署文件，并配置 nginx（或其它HTTP服务）。
2. 部署并配置后台服务 `pkg-api` 站点。
3. 修改配置文件 `./tars.ini`。配置基本信息、后台服务API地址等。
4. 调整 `./www/assets/js/app.js` 中的一些前台配置项。

## 目录结构

```
tars-console/
├── logs/                                # 日志
│   ├── curl.error.log                 # 接口请求错误日志
│   ├── curl.notice.log               # 接口请求访问日志
│   └── curl.options.error.log         # 接口请求错误对应的请求参数日志
├── src/                                # PHP 源文件
│   ├── common/                       # 公共类库
│   ├── controller/                   # 控制器
│   ├── remote/                       # 外部请求类
│   └── views/                         # 视图文件
├── www/                                # Web 源文件
│   ├── assets/                       # 静态资源
│   │   ├── css/
│   │   │   └── default.css           # 样式文件由 `../scss/default.scss` 自动生成
│   │   ├── js/
│   │   │   ├── app.js                # 应用入口 JS（前台配置项、页面 state 配置等）
│   │   │   ├── controllers.js        # Angular 控制器
│   │   │   ├── directives.js         # Angular 指令（分页）
│   │   │   └── services.js           # Angular 服务（过滤器、API请求、工具类）
│   │   ├── lib/                      # 第三方库
│   │   └── scss/                     # SASS 源文件
│   ├── templates/                   # 页面模板
│   └── index.php                     # Web 入口文件
└── tars.ini                          # 基本配置文件
```

## 开发

主要依赖：

- Angular.js
- Bootstrap

- AngularUI Router
- Flight - PHP 微型框架，用于构建 RESTful 的 Web 应用

```
cd tars-console
npm install -g gulp # NPM 全局安装构建工具 gulp
npm install         # NPM 本地安装 gulp, gulp-sass
gulp               # 启动 gulp 任务，自动监听 scss 文件变更，生成 css
```

## 启用第三方登录

### 1. 修改配置文件 `./tars.ini`

```
[auth]
auth.use_third_part = yes
; 第三方登录后，回调地址
auth.redirect_url = "http://www.tars.com/auth"
auth.api_url = "http://api.third.com/"
auth.appid = "12345"
auth.appkey = "abcdef"
; 系统管理员列表，使用分号 `;` 分隔
auth.admin_list = "zhangsan;lisi"
```

### 2. 编写第三方登录逻辑 `./src/controller/Index.php`

```
class Index extends BaseController {
    public function signin() {
        if (Config::get('auth.use_third_part')) {
            // 这里通常跳转到第三方登录页
        }
    }

    public function signout() {
        // 先注销本地 SESSION
        if (Config::get('auth.use_third_part')) {
            // 这里通常跳转到第三方退出页（或调用第三方退出接口）
        }
    }

    public function auth() {
        // 第三方登录后的回调
        // 这里通常调用第三方接口解析登录票据 ticket
        // 然后写本地 SESSION
    }
}
```