# 目录结构

一个完整的 炸毛框架 包含：

* Console 控制台输入输出
* HTTP + WebSocket 服务器（开到一个端口）
* RemoteShell 远程调试 Telnet 端口

├── bin/ ---- 程序入口，可执行文件目录  
├── config/ ---- 框架所需的配置文件目录  
│   └── global.php ---- 框架核心配置文件  
├── resources/ ---- 资源文件夹  
│   └── html/ ---- 静态 HTML 页面存放目录  
├── src/ ---- 框架根目录  
│   ├── Custom/ ---- \*\*可修改\*\* 用户自定义添加的框架内对象  
│   │   ├── Annotation/ ---- 用户自定义添加的注解类  
│   │   ├── global\_function.php ---- 用户自定义添加的全局函数  
│   │   └── Connection/ ---- 用户自定义添加的 WebSocket 连接类型类  
│   ├── Framework/ ---- 框架引导和全局代码，不可修改  
│   │   ├── Console.php ---- 控制台输入输出的管理类  
│   │   ├── ZMBuf.php ---- 框架内的高速全局缓存变量储存的管理类  
│   │   └── global\_functions.php ---- 全局函数  
│   ├── Module/ ---- 用户编写模块代码的目录（此层内目录结构并无限制）  
│   │   └── Example/ ---- 模块子文件夹  
│   │      └── Hello.php ---- 默认自带的第一个模块代码  
│   └── ZM/ ---- 框架核心代码主目录，一般不需要动  
│   ├── API/ ---- API接口工具类目录  
│   │   ├── CQ.php ---- CQ码解析和生成的工具类  
│   │   └── CQAPI.php ---- 调用 CQHTTP 的 API 接口的工具类  
│   ├── Annotation/ ---- 框架的注解类目录  
│   │   ├── AnnotationBase.php ---- 注解类的基类  
│   │   ├── AnnotationParser.php ---- 注解类解析器  
│   ├── Connection/ ---- WebSocket 连接定义类  
│   │   ├── CQConnection.php ---- CQHTTP 来的 WebSocket链接  
│   │   ├── ConnectionManager.php ---- WS 链接管理类  
│   │   ├── UnknownConnection.php ---- 未知连接的类  
│   │   └── WSConnection.php ---- WS 连接的基类  
│   ├── DB/ ---- MySQL 数据库操作的类  
│   ├── Event/ ---- 事件触发类的目录  
│   │   ├── CQ/ ---- 酷Q 发来的消息、通知等事件处理类  
│   │   ├── EventHandler.php ---- 管理事件的类  
│   │   └── Swoole/ ---- Swoole 内部的绑定事件  
│   ├── Exception/ ---- 框架内部的 Exception 异常的类的目录  
│   ├── Http/ ---- 框架内继承或重写 Swoole 内部组件中间层  
│   ├── ModBase.php ---- 用户编写的模块类的基类  
│   └── Utils/ ---- 工具类的目录  
│   ├── DataProvider.php ---- 提供文件读写、缓存读写等功能  
│   ├── ZMRequest.php ---- HTTP 客户端工具类  
│   ├── SQLPool.php ---- 数据库连接池对象  
│   ├── ScheduleManager.php ---- 计时器管理类  
│   └── ZMUtil.php ---- 其他杂项函数集合的类  
└── zm\_data/ ---- 炸毛框架默认指定的数据目录

::: warning 注意  
一般情况下，使用框架进行编写机器人的功能仅需修改 src/Module/，src/Custom/ 目录下的文件，剩余的 其他 src/ 下的子目录为框架代码，谨慎或不要修改，除非你完全知晓框架的运行机制和流程代码。以上 src 中标黑的部分为用户开发部分。

比如自带的一问一答示例 **Hello** 模块类，就放到了 Module/ 下的 Example/ 目录中，你也可以将模块类直接放入 Module/ 根目录下，也可以放到子目录的子目录中，以此类推，根据自己项目的需求和结构灵活掌握。  
:::