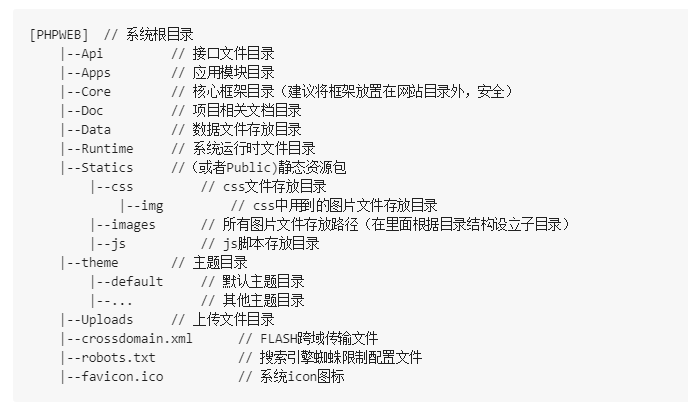
**ELO算法的可视化在五子棋游戏中应用**

我们项目的代码及其他文件信息都放在了GitHub网站上，每次有代码和文件更新都会及时上传。

首先，我们的项目分为客户端（backgammon\_online\_c）和服务端（backgammon\_online\_s），客户端用到的技术栈为Element+Vue + Vuex + Vue-router + axios + socket.io-client，服务端技术栈为node.js + express + socket.io，为B/S（C/S）体系结构。



客户端体系结构：

**[Backgammon\_online\_c] // 系统根目录**

|--public // 用于存放静态资源

|--src // 存放系统源码

|--assets // 存放小型资源（图片）

|--images // 存放图片

|--components // 存放组件级Vue文件

|--content // 存放内容区域组件

|--game // 存放游戏组件

|--header // 存放头部组件

|--side\_bar // 存放侧边栏组件

|--views // 存放视图级Vue文件

|--App.vue // 项目实例入口

|--main.js // 项目配置入口

|--router.js // 存放路由定义

|--store.js // 存放全局数据

|--.eslintrc.js // eslint配置文件

|--.gitignore // Git忽略文件配置

|--.postcssrc.js // postcss配置文件

|--babel.config.js // ES6转ES5配置文件

|--package-lock.json // 日志

|--package.json // 项目依赖文件

|--vue.config.js // Vue项目配置文件

服务端体系结构：

**[Backgammon\_online\_s] // 系统根目录**

|--bin // 服务器启动文件

**|--log // 日志文件**

**|--models // 数据库模型**

**|--Common // 公用模型**

**|--Users // 用户模型**

**|--Usually // 常用模型**

**|--mysql // 数据库建表记录文件**

**|--routes // 路由**

**|--socket\_io // websocket模块**

**|--nameSpace // 命名空间模块**

**|--views // 视图模块**

**|--.gitignore // Git忽略文件配置**

**|--app.js // 服务器项目入口**

**|--package,json // 项目第三方依赖**

**|--package-lock.json // 日志**