

使用 WebSocket 协议而不是 http 协议来实现多用户匹配。

WebSocket 服务器与用户都可以主动发送信息给对方，且接收信息后可以不立即应答。

匹配流程：

首先客户发送匹配的请求给后端服务器，服务器再发送客户信息给匹配系统。（这个匹配系统是通过微服务实现的）匹配系统将排名相近的两个客户匹配在一起并发送给服务器，服务器再将匹配信息发送给这两个客户。

在服务器端需要实现的功能：

1. 生成地图并把这个地图发送给两个客户；
2. 依次等待两个客户发送操作命令，收到命令才执行下一步操作，或者有个客户长时间未操作直接判定这个客户输；
3. 除了可以接收客户发送的命令外也可以接收机器人的代码输入；（实现代码指令操作机器人需要微服务实现）

服务器端具体细节怎么实现：

1. 等待指令：设置一个死循环，循环内执行等待 1 秒的指令，等待一秒后还没收到指令则重新执行循环语句，否则 break。循环体等待了 5 秒还没收到指令则判定当前等待的客户输；
2. 判定客户的指令是否合法：若客户的指令会让蛇撞墙则直接判定客户输；
3. 链接客户与服务器：给每个客户创建一个 WebSocketServer 类的对象，所有有关链接的信息都会存到这个类里面；

