

服务器通信协议

Web 与服务器接口协议

一. 通信协议

1. 协议类型

Web 与服务器之间通过 WebSocket 协议通信：
通信端口为：3000；
通信地址如："ws://192.168.1.101:3000";其中 192.168.1.101 为服务器 ip 地址。

二. 发送 JSON 数据格式

发送数据到服务器执行 JSON 格式：
JSON.stringify({"device": "table", "topic": topic, "data": data //数据信息})
A) device:固定写为 table，类型为字符串。
B) topic: 数据类别，字符串格式。
C) data: 详细数据，字符串格式。

三. 接收 JSON 数据格式

`{"device": "table", "topic": "yaml", "id": "robot-uuid", "nickname": "table_nick", "data": "{\"x\n\":" -20.000000\", \"y\":" -20.000000\"}"}`

接收到的数据为 JSON 格式：

topic: 类型	string
id: 机器人 id	string
nickname: 工位名	string
data: JSON 数据	string

四. server 端协议详情：

1. 页面登陆：

Server 接收：

topic: login	登陆验证	类型：string
data: {'pad':2017}	登陆密码	类型：json

server 回复：

topic: msg_reply	server 端返回信息	类型：string
data.type: login_reply	登陆返回信息	类型：string
data.content:	返回内容	类型：string
content:		
1. login_success	登陆成功	类型：string
2. login_failure	登陆失败	类型：string

2. 控制机器人流程：

A. 请求机器人

Server 接收：

topic: applyRobot	请求机器人	类型：string
data:	请求工位信息	类型：json
1.nickname	工位名	类型：string
2.type	工位类型	类型：string

Server 返回:

topic:msg_reply	server 端返回信息	string
data:		
1.type: msg_ApplyRobot	请求返回信息	string
2.id	请求到机器人 id	string
3.tablename	请求到工位的名字	string
4.nk	请求到的任务序列	init

B. 锁定机器人

JSON.stringify({"device": "table", "topic": "lock", "data": { "id": 小车 id, "type ": 工位类型, "nickname": 工位名, "nk": 任务序列}})

Server 接收:

topic:lock	锁定机器人	string
data :		
1.id	机器人 id	string
2.type	工位类型	string
3.nickname	工位名	string
4.nk	任务序列	init

Server 返回:

topic:msg_reply	server 端返回信息	string
id	锁定机器人 id	string
data:		
1.type: msg_lock	锁定机器人返回信息	string
2.replay:	锁定机器人返回值	init
1	锁定成功	
0	锁定失败	
3.nickname	工位名	string
4.nk	任务序列	init

C. 移动机器人

JSON.stringify({"device": "table", "topic": "target", "data": { "id": robotID, "nickname": tablename, "type": "取料", "nk": nk}})

Server 接收:

Topic: target	移动机器人	string
Data:		
1.id	机器人 id	string
2.nickname	工位名	string
3.nicktype	工位类型	string
4.nk	任务序列	init

Server 返回:

topic:msg_reply	server 端返回信息	string
id	机器 id	string
data:		
1.type: msg_RobotReached	机器人到达反馈	string
2.nk	任务序列	init

D. 停止机器人

Server 接收:

Topic: cease	停止机器人	string
Data:		
1.id	机器人 id	string

Server 返回:

topic:msg_reply	server 端返回信息	string
id	机器 id	string
data:		
1.type: msg_RobotReached	机器人到达反馈	string
2.nk	任务序列	init

E. 解锁机器人

JSON.stringify({"device": "table", "topic": "unlock", "data": { "id": robotID})

Server 接收:

topic: unlock	解锁机器人	string
data:		
1.id	机器人 id	string

Server 返回:

topic:msg_reply	server 端返回信息	string
id	机器人 id	string
data:		
1.type: msg_unlock	解锁机器人返回信息	string
2.locked:	解锁状态值	init
1	解锁失败	
0	解锁成功	

3. 获取机器人位置

机器人位置是实时传递的

{device: "table", topic: "pose", id: "WT_001", data: "{ \"y\" :-0.8816405465350527, \"x\" :

1.9060866026786485 , \"a\" :-0.7048353026749968, \"w\":0.7093709862286769}"} }

Server 实时返回:

Topic: pose	机器人位置信息	string
id	机器人 id	string
data:		
1.x	机器人 x 坐标	float
2.y	机器人 y 坐标	float
3.a	机器人 a 坐标	float
4.w	机器人 w 坐标	float

4. 获取地图配置信息

地图配置信息是页面登陆成功后服务器立即发送的

Server 返回:

Topic: yaml	地图位置信息	
Data:		
1.x	地图 x 坐标	
2.y	地图 y 坐标	

5. 获取路径信息

路径信息是登陆成功后服务器立即发送的

```
{device: "table", topic: "roadpath", id: "table-uuid", nickname: "table_nick", data: "{
" path_main.txt": " \\"1302\\n-0.052000  -3.269000 \\...02000  -2.801426 \\n0.002000
-2.842000 \\n\\\""}"}
```

Server 返回:

Topic: roadpath	路径信息	string
Data:		
1. path_main.txt	路径文件名	string

6. 位置点信息

位置点信息是登陆成功后服务器立即发送的

```
{device: "table", topic: "posPositions", id: "table-uuid", nickname: "table_nick", data:
"[{"x":0.005036812805627733,"y":-4.296907716686555,"a":0.7092592236111406,
"w":0.7092592236111406,"nick":"002", "zone":"取料"}]}
```

Server 返回:

Topic: posPosition	位置点信息	string
Data	位置数组数据	json 数组
单组数据中:		
1.x	x 坐标	
2.y	y 坐标	
3.a	a 坐标	
4.w	w 坐标	
5.nick	位置名	string
6.zone	位置类型	string

7. 机器人上线、下线通知

Server 返回:

机器人上线:

topic: msg		
data:		
1.type: msg_robotbutton	机器人上线	string
2.uuid	机器人 id	string

机器人下线:

topic: msg		
data:		
1.type: msg_robotdownline	机器人下线	string
2.uuid	机器人 id	string

8. 添加、移除位置点

Server 接收:

Topic: robot_cmd	指令	string
Data:		
1.cmd	指令	init
5	添加位置点	
6	移除位置点	
2.nickName	位置点名称	string

3.data	位置点坐标信息	json
Server 返回:		
Topic: posPosition	位置点信息	string
Data	位置数组数据	json 数组
单组数据中:		
1.x	x 坐标	
2.y	y 坐标	
3.a	a 坐标	
4.w	w 坐标	
5.nick	位置名	string
6.zone	位置类型	string

9.机器人扫码

Server 接收:

```
JSON.stringify({"device": "table", "topic": "scanCode", "data": { "id": 小车 id, 'cmd': 命令值 }})
```

Topic: scanCode	扫码操作	string
Data:		
1.id	机器人 id	string
2.cmd	指令	init
0	备用	
1	获取二维码信息	
2	二维码精准矫正	

Server 返回:

```
{device: "table", topic: "msg_reply", id: "机器人 ID", nickname: "table_nick", data: {"type": "scanCode", "cmd": "1", "replay": 1, "reply_data": 2921}}
```

topic: msg_reply	返回信息	string
id	机器人 id	string
data:		
1.type: scanCode	扫码返回	string
2.cmd	指令	init
1	获取二维码信息	
3.replay	执行	init
1	成功执行	
4.reply_data	2921	二维码数据（内容待定）

10.机器人顶升

Server 接收:

```
JSON.stringify({"device": "table", "topic": "upDown", "data": { "id": 小车 id, 'cmd': 命令值 }})
```

topic: upDown	顶升	string
data:		
1.id	机器人 id	string
2.cmd	指令	init
0	激光对准矫正	
1	上升	

Server 返回:

```
{device: "table", topic: "msg_reply", id: "机器人 ID", data: {"type": "upDown ",  
"cmd": "1", "replay": 1, "reply_data": 0}}
```

topic: msg_reply	返回信息	string
id:	机器人 id	string
data:		
1.type: upDown	顶升返回	string
2.cmd	指令	init
1	上升	
3.reply	执行状态	init
1	成功执行	
4.reply_data:0(无)		