

此文档用于对系统整体开发各部分协调作详细安排

GUI 设计

GUI 设计部分与其他本分的交互主要在与数据库的 post 和 get 请求上：

1.Post：提交 command 指令。

数据库部分定义 post 接口为 post_command.php; 提交指令格式按照一下数据库表的结构：

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	commandID	int(11)			No	None			Change Drop More
<input type="checkbox"/> 2	start_x	float			No	None			Change Drop More
<input type="checkbox"/> 3	start_y	float			No	None			Change Drop More
<input type="checkbox"/> 4	start_z	float			No	None			Change Drop More
<input type="checkbox"/> 5	end_x	float			No	None			Change Drop More
<input type="checkbox"/> 6	end_y	float			No	None			Change Drop More
<input type="checkbox"/> 7	end_z	float			No	None			Change Drop More
<input type="checkbox"/> 8	time	time			No	None			Change Drop More
<input type="checkbox"/> 9	IDs	binary(1)			No	None			Change Drop More

GUI 部分负责生成 command，使用 jquery post: \$.post (URL,data,callback) ;

其中，post_command.php 由数据库部分提供。

2.Get：查询智能车的 location 及 post 信息，以确定其状态。

数据库部分定义 get 接口为

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	ID	int(11)			No	None			Change Drop More
<input type="checkbox"/> 2	X	float			No	None			Change Drop More
<input type="checkbox"/> 3	Y	float			No	None			Change Drop More
<input type="checkbox"/> 4	Z	float			No	None			Change Drop More
<input type="checkbox"/> 5	speed	float			No	None			Change Drop More
<input type="checkbox"/> 6	pitch	float			No	None			Change Drop More
<input type="checkbox"/> 7	roll	float			No	None			Change Drop More
<input type="checkbox"/> 8	azimuth	float			No	None			Change Drop More
<input type="checkbox"/> 9	time	time			No	None			Change Drop More

get_all.php;该接口返回每个 ID 下的最新（根据 TIME 排序）的车辆的信息。

get_single.php.该接口接收一个参数（GUI 指定 ID），返回该 ID 的最新信息。

GUI 部分负责:

1. 使用 `get_all` 查询所有车辆的信息，然后根据时间（设定一个超时判断）在左边栏第一部分显示出小车的状态信息。此外，实时刷新 `map` 里面 `echarts` 显示的小车位置。

ID	TIME	STATE
01	09:00	

2. 使用 `get_single` 查询用户指定的特定车辆的信息，显示在左边栏的第二部分。当用户在右侧 `map` 点击某一车辆，自动发送 `get` 请求，然后在左边栏出现其信息。

ID		01	
Pich		01	
Row		01	
Yaw		01	
Location		01	

3. 将 GUI 获得的所有原始信息在左边栏 message 不断显示出来。自下而上不断刷新。

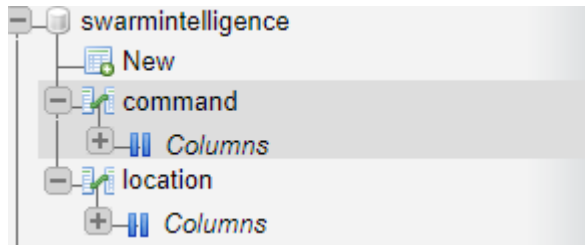
Message

```
000000000000000000000000000000000000000000000000000  
000000000000000000000000000000000000000000000000000  
00
```

数据库及通信部分

数据库部分:

数据库的结构:



数据库提供的接口：

1. 面向 GUI 提供：
 - post_command.php: 将 GUI 提交的 command 存入数据库的 command 表，同时解析其中的 IDs 包括哪些 cars，然后多播。
 - get_all.php: 该接口返回每个 ID 下的最新（根据 TIME 排序）的车辆的信息。
 - get_single.php: 该接口接收一个参数（GUI 指定 ID），返回该 ID 的最新信息。
2. 面向智能车提供：
 - 接收 cars 上传的数据，存入数据库
 - 多播 command

通信

对 GUI 提供 php 接口 (如上文)，对 cars 提供 TCP 与 UDP 通信，并将上传信息存入数据库。