1. **Server + data Process**
   1. **Device iOT**

-> data son format

{

distance : 20; // centimeters

degree : 320; // degrees

order\_nb : 1;

(remaning\_time : 100 // milliseconds)

}

* 1. **Router / Server technology nodejs**

-> loop

- wifi connection / network

* + - * receive data
      * parse json
      * calc/store position
    1. verify last remaining time
    2. *if the angle is not ok try to correct —> final step*
    3. determine where is the car on the map
    4. draw the map
    5. define target destination
    6. set direction
    7. store commands
       - serialize json

{

rotation : 20; // degree

direction : 0 // boolean (0: forward, 1: back )

time : 1000 // milliseconds

order\_nb : 1;

}

* + - * sent back the action
      * show the output on a webpage / or terminal

1. **Map**

legend

forbidden postion

car 🚓

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | 🚓 | 7 | 6 | 5 | 4 | 1 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | 3 | 2 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |