Galileo documentation

For this project, you need :

1 – Arduino Galileo

1 – LCD Shield

1 – RJ45 Cable

1 – USB Cable

1 – Computeur for coding

1 – Raspberry pi to do server role.

And is recommanded to use a arduino software, I use the version 1.5.3.

Library to install :

WebSocketClient : « a websocket client for Intel Galileo », this library is from mauromezze, and you can see more information on:

https://github.com/mauromezze/IntelGalileo-SpacebrewWebsocket

aJson : « aJson is an Arduino library to enable JSON processing with Arduino. It easily enables you to decode, create, manipulate and encode JSON directly from and to data structures. »

https://github.com/interactive-matter/aJson

To setup :

1 – Install the LCD Shield.

2 – Power on the Galileo.

3 – Connect to Raspberry Pi with RJ45 cable.

4 – Connect to computeur with USB cable.

5 – Install the two library in the good folder.

6 – Download the source project in : https://github.com/Jeremy-Imerir/Architecture-Platforms/tree/master/Galileo .

7 – Start the last version, actually « FinalVersion ».

Program description :

First step :

Connection with Raspberry pi on HTTP server for a little request.

We do a « GET / » and the server respond a json weft (« trame » in french), in this weft give us the ip address and the port of the websocket server like this :

{"IP":"192.168.0.1","port":8000}

Second step :

Connection with Raspberry pi on WebSocket server with the informations of the precedent step.

Third step :

We read the message sent by the server, the server say 3 message :

1 – welcome

2 – Cab information

3 – Cab Request

The second message is for general information of the cab, like position, statuts, end of the traject.

Exemple :


```
{
    "cab": {
        "x": 5,
        "y": 5,
    "goTo": "vertex",
    "status": "busy",
    "distanceToEnd": 50
    }
}
```

The third message is for a cab request, a new client with this informations.

Exemple :

```
        {
        "cabRequest": {
                "area": "areaname",
                 "from": "a",
                 "to": "b"
}
}
```

Every informations is wrote on the LCD.

We can Accept or refuse the client, press Right button for accept and return the weft on server, or press Left to refuse.

For more technical information, I invited you to read the source and this comment in « english » or in french.

Thanks for reading !