From RGB led laboratory to servomotor control with websockets and IoT as educational tool

German Carro Fernandez, Elio Sancristobal Ruiz, Manuel Castro Gil, Francisco Mur Perez
Electrical and Computer Engineering Department
Spanish University for Distance Education (UNED)
Madrid, Spain
germancf@ieee.org, elio@ieec.uned.es, mcastro@ieec.uned.es, fmur@ieec.uned.es

***Abstract*—The technology is advancing so unstoppable even in times of crisis. Sometimes it is difficult to take advantage of it because in no time a new and improved version of the above techniques appears to continue this progress. Within educational technologies aimed at enhancing remote laboratories is the same. From HTTP to websockets communication protocols the technology offers new alternatives to improve bidirectional communications between the client and the server. This paper shows how an integrated development initially HTTP has evolved to lead to a websockets-supported system for controlling a servomotor. The introduction of this communication technology is simplified to allow reuse existing laboratories, thus allowing reuse HTTP previous architectures, hardware and services, including a more advanced environment that uses websockets as communications support. How couldn't it be otherwise, the characteristic of the IoT hardware, in this case Arduino and Raspberry Pi, has been essential in this task. The goal: to reuse existing remote laboratories improve their communication protocol and expand the possibilities of client-server interaction retaining its character as an educational tool.**

Keywords—websockets; IoT; Raspberry Pi; Arduino; remote laboratories

I. INTRODUCTION

In 2013 and within a set of remote electronic laboratories aimed at schools and young students, it began to design a laboratory that used a RGB led as an educational tool to supplement based on optical sensing ILS (Inquiry Learning Space), the physics of colors , and mechanical performance of the optic nerve in humans, insects and animals.

Initially this laboratory was HTTP and PHP supported and its initial integration responded to its inclusion as a real example into a suite for integration of remote laboratories (SiLaRR) and later encompassed by the Techno Museum project and integrated into the GoLab project. This laboratory was named "The color of the light". Following the success obtained in its use as an educational tool, and given the opportunity to develop new laboratories using the technology developed for it, it came a new approach that would allow a better integration of the system that stand in the IoT (Internet of Things), [1].

Since the laboratory used Arduino, it was decided to add a new complementary hardware to allow its portability. For this, the Raspberry Pi Model B was selected to continue the evolution of development, and because GoLab project needs, websockets protocol was introduced to replace the previous HTTP system.

Since the system should allow reuse in other laboratories, it was decided to replace the RGB led by a servomotor. The result allowed us to check the possibilities of integrating the system with a new communications technology, its versatility by reusing it with new hardware, and adaptability to technological progress.

No modifications to the source code of the Arduino, which remains its basic hardware, the new laboratory "Remote Servo Control" allowed one step in the reuse of hardware/software preserving educational use.

The success of this intermediate step is allowing, currently, developing an integration of a robotic arm that uses Raspberry Pi, Arduino and websockets to let students/teachers in schools can access it and control it remotely.

II. FIRST THERE WAS LIGHT

The laboratory "The color of the light" was tested and used successfully in several schools, and its development was closed as regards to HTTP through its final stable version.

However, a second integrated in LabView version proved that it could also be used with Arduino Mega with the same efficiency as was done with Arduino UNO. The final integration results were shown in previous publications, [2].

Following that success he began work on a new development that reused the existing system in this laboratory, but adapted to websockets. Since the objective was to advance the IoT and allows the user to access the maximum capacity of laboratories to develop, various changes were undertaken:

A. Hardware

New hardware was added, more powerful than the Arduino and like it allows portability of the system: Rapsberry Pi. At that time it enlisted the Raspberry Pi model B due to its technical characteristics.

B. Software

Due to the characteristics of the Raspberry Pi; and its use of Rapbian (a flavor of Linux), it decided to modify the software

978-1-4799-7839-7/15/$31.00 ©2015 IEEE

support and use Javascript instead of PHP. Also due to use of HTTP against to websockets, it was decided to use Node.js against Apache.

With the above changes the relevant modifications are initiated to adapt the old system to the new. As has been indicated, initially the lab Led RGB interacted directly with Arduino UNO, [3], allowing client controlling the color of the light emitted by the RGB Led remotely using HTTP, [4], as the communication protocol. However the source code uploaded to the Arduino was versatile enough to be reused with a few changes.

The main objective was to continue retaining user interaction with the system, [5].

III. THEN WAS THE WHEEL

So the next step was to set the Raspberry Pi and Raspbian OS, [6], to communicate with Arduino UNO using a USB port. Likewise it was necessary to define what criteria should be followed to determine the communication via websockets, [7]. To solve both problems we had two options:

1) Use JavaScript exploiting the interaction with Node.js, [8].

2) Use one of the most programming languages used to Raspberry Pi, Python, and for that, use Tornado, [9] as a framework for the new system.

Either choice would decide the future final system development.

The option finally chosen, for the reasons cited above, was JavaScript, but both are described at length below.

A. *Javascript and Node.js*

Node.js is a server that allows you to use JavaScript as a basic programming language. While its initial value is reduced to the characteristics of any server functions, has the added advantage of having several libraries available that provide it with greater functionality.

Specifically NPM repository, [10], is the place where you can download and adding the above libraries that subsequently can be added to Node.js.

In our case and because it was needed interaction with Arduino UNO via USB port, it was necessary to have a library for this purpose. For this was chosen the *serialport* library for its simplicity and ease of use. In this way and simply stating the characteristics of USB data transmission protocol, the Raspberry Pi would communicate directly with Arduino UNO by the USB port in real time.

On the other hand we had the issue of websockets. Within NPM repository there are many libraries to work with websockets but previously there was a problem to be solved. Initially the websockets protocol must have a compatible browser from the client side.

While the latest versions of the most used browsers on the market support this protocol, there are still browsers or older versions of the aforementioned browsers that do not support websockets protocol, [11].

Since the objective of a remote laboratory is disclosure as an educational tool, and that means that it must be accessible by as many users as possible, [12], it was necessary to find a library that allows access by websockets as a priority, but at while allowing access to the system by other browsers using other older protocols. It was therefore resorted the library *Socket.io*.

B. *Tornado and Python*

The other option available was to use Python to implement the system. Python is a language traditionally used in Raspberry Pi. Its simplicity and power allow managing the hardware in an optimal way and in our case contains libraries to manage USB ports.

Specifically has the library *serial* that facilitates this process greatly.

However it was also necessary to have a framework to undertake the work necessary to launch server and deploy the system on the Internet. If we talk about Python, Tornado can be that framework.

Again we must consider that the required communication protocol should be websockets, thereby Tornado within the library *websocket* was used to perform this function, although in this case taking into account the limitations of use for unsupported browsers as unlike Socket.io this library does not have multiprotocol function.

Once defined the tools to use, the system was ready to be tested with both options. For this and how proof of concept we decided to use a servomotor and see what possibilities offered in each case.

IV. CONTROL YOUR DEVICE USING IOT CAPABILITIES

Tests with JavaScript and Tornado were satisfactory, but given the final choice by JavaScript, the steps used with this option will be described.

The philosophy of IoT allows to control remotely devices, information and events that currently are not control. In fact the IoT rise from the need to track manufactured goods, but later leads to what we all know as the Internet of things, something much more general and for the general public and that is currently taking part of its content from another new discipline such as Big Data and smart transmission of pool information data, [13].

From this point of view the goal of our research was to design a system that could be deployed on the Internet and could be used by students; schools initially and later in academia; and use the basis of IoT regarding control, interactivity, simplicity and immediacy.

Thus we proceeded to set up the scheme shown in Figure 1. On the one hand the server hosted on the Raspberry Pi has direct communication with the Arduino UNO via USB. When server receives client information, server sends to client side the current system settings in real time using the traditional JSON protocol basis of technical standard REST.

The client side hosts a standard web page HTML with CSS library that for this case has been implemented using Bootstrap but can also be implemented with jQuery or any other existing library.

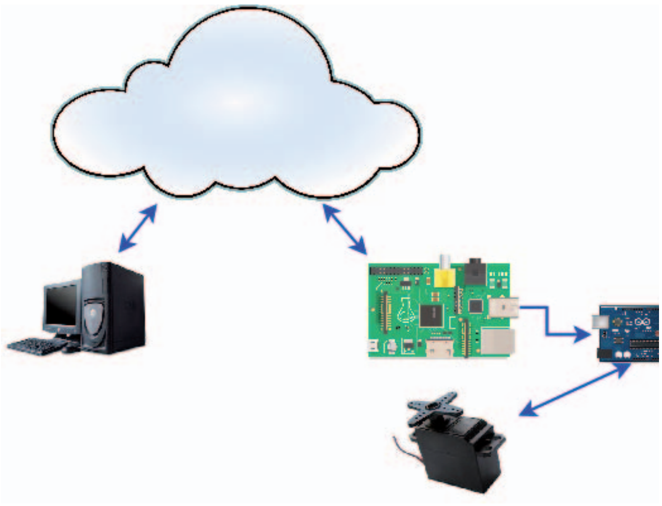

Figure 1. Architecture of the system.

The system enables a stable two-way communication supported by the communication possibilities of the Raspberry Pi and Arduino UNO hardware.

Due to the simplicity of deployment, the web is accessible from any mobile device. Also thanks to the versatility of the system is fully configurable for other electronics laboratories as will be seen below.

With this system, as a basis, the possibility of increasing the bidirectional information received and / or emitted is only a matter of customizing the relevant JSON.

Also if you want to manage Big Data or information to a lower level, you can easily integrate a database to store and process the desired information.

Once successfully tested the operation locally, we switched to production testing in a real environment. A school as evidence and a sample of a class of 20 students aged between 10 and 12 years was used in this case.

The website was totally intuitive to what students should only move a slider to fix the servo to the desired position.

The Raspberry Pi holds up right 20 parallel connections although it was detected that at times there were delays in receiving data.

However these tests only served to verify the direct interaction with the system and move to the next and final phase.

V. Finally was the Robotic Arm

Preliminary testing of the system leading finally to the development of remote configuration and control of a complete robotic equipment, in this case the robot arm of Figure 2, and the design of a learning environment (ILS) suitable for different educational targets from 10 to 18 years.

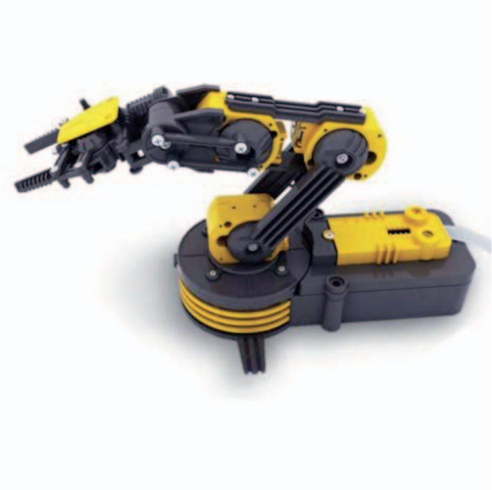

Figure 2. Robotic arm used.

Hardware scheme kept the same except for the fact of replace the test servo for the several actuators or services of robotic arm.

The increased complexity of the new hardware configuration forced small adjustments in the electronics using H-bridges between the Arduino UNO and the robotic arm to ensure adequate electronic stability of the response of the actuators, [14].

Furthermore the software required deeper adaptations that are explained below:

a) Increasing the number of websockets channels:

Because each actuator receives and transmits its own data, it was decided to implement a single channel websocket for each of them, thereby encapsulating all the information individually.

b) Adaptating of Arduino source code:

Moreover the source code of Arduino UNO had to be modified to handle the total number of services available (6 in total: led, clip, wrist, elbow, shoulder and base).

c) Modification JSON's information and content:

Likewise the information contained in JSON was splited and costumed for each service listing and assigning its characteristics, name, possible values and currently values recorded in real time. All this for the client can know at all times what happens whenever he/she interacts with the system.

d) Development of an Inquire Learning Space (ILS):

It was then necessary to draw up an appropriate learning path for the end remote laboratory was useful. For this purpose it was developed an specific ILS, [15], which focused its content in areas such as mechatronics, mechanical explanation of analog and digital devices such as motors and servomotors, evolution of robotics, science fiction literature, history of robotics in east and west societies, gadgets and robotic mechanisms, importance of the microprocessor in the current robotics, industrial and consumer robotics and robotics and economics.

All this to allow the use of the ILS; with minor changes by the teacher; with children of different ages from 10 to 18 years.

After adjustments and modifications listed above the remote laboratory was tested at the previous cited school.

This time was tested in different classes with different ages and with a total of 70 children and 4 teachers. As in the case of the servomotor, the interface was designed intuitively as can be seen in Figure 3.

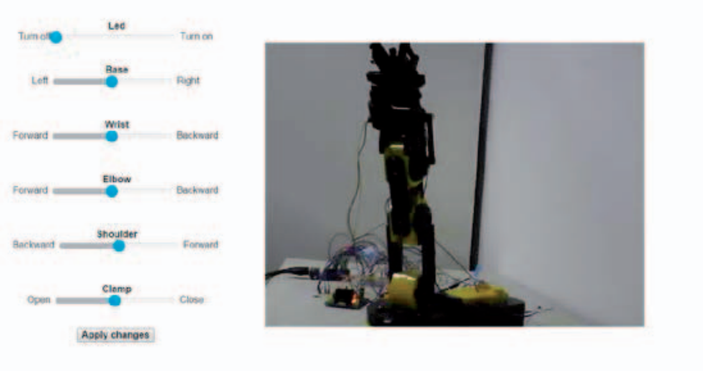

Figure 3. Client user interface of robotic arm laboratory.

With those adjustments and final ILS the remote laboratory became a useful tool for technology classes.

The feedback received during the sessions which it was used (total of 4) has allowed us to collect data to improve the system and develop new laboratories using it as a base.

VI. Conclusions

New technologies greatly facilitate the deployment of information, data and services using the Internet. It is very important to translate these new tools to the educational environment, especially when we talk about robotics in a world so technological like this, [16].

Throughout this paper explained an experience that has allowed using these technologies and specifically the philosophy of IoT to display different remote laboratories on the Internet and convert them into useful tools for the educational environment in an area of between 10 and 18 years.

Although the process has been on an increasing scale of difficulty, the ultimate goal of integrating a robotic laboratory or a servomotor into a classroom is, not only, a real test of IoT opportunities for education, but a new experience for teachers and students.

Teachers have seen how technology can help them teaching their sessions and the students are a little more motivated to learn due to the novelty of using a robotic tool remotely.

To reach the ultimate goal it was necessary to go through intermediate steps slowly, first with Led RGB and from a traditional protocol based on HTTP and then to the servomotor and the necessary modifications for using websockets in an environment developed with JavaScript.

All these steps have helped to know the limitations and to improve the initial development of the hardware and software tools used, and finally to understand how we can move the progress made at school level using an ILS or educational itinerary that fit with what might be useful for teachers and students.

With the information gathered and feedback acquired we are ready to take a step beyond the system and the new goal is to design a suite of laboratories that take advantage of this experience to offer a greater variety of remote laboratories and continue contributing to the development of tools that facilitate learning, face to face or remotely.

Acknowledgment

The authors acknowledge the support provided by the Engineering Science School of UNED, the Computer Science School of UNED, the UNED Rectorate, and the IEEE Student Branch of UNED, and “Techno-Museum: Discovering the ICTs for Humanity” (IEEE Foundation Grant #2011-118LMF), “e-Madrid: Research and development of learning technologies” (S2013/ICE-2715) and “Go-Lab: Global Online Science Labs for Inquiry Learning at School” (FP7-ICT-2011-8 – Project number 317601).

Authors are especially grateful to the Electrical and Computer Engineering Department (DIEEC) of UNED for its support and advice in the preparation of this paper.

References

[1] Zheng, J. Simplot-Ryl, D. Bisdikian, C. Mouftah H.T..“The internet of things”, Communications Magazine, IEEE, Volume: 49, Issue: 11, Page(s): 30- 31, 2011

[2] Carro, G.; Castro, M.; Sancristobal, E.; Diaz, G.; Mur, F.; Latorre, M.; Chaparro, M.; Lopez-Rey, A.; Salzmann, C.; Gillet, D., "The color of the light: A remote laboratory that uses a smart device that connects teachers and students," Global Engineering Education Conference (EDUCON), 2014 IEEE , vol., no., pp.854,860, 3-5 April 2014 doi: 10.1109/EDUCON.2014.682

[3] Arduino UNO Official site: http://arduino.cc/en/pmwiki.php?n=Main/ArduinoBoardUno consulted on 8th january 2015.

[4] HTTP Protocol, refernce site: http://www.w3.org/Protocols/ consulted on 8th january 2015.

[5] Fernandez, G.C.; Gil, M.C.; Perez, F.M., "Remote robotic laboratory as nexus between students and real engineering," *Interactive Collaborative Learning (ICL), 2012 15th International Conference on*, vol., no., pp.1,4, 26-28 Sept. 2012 doi: 10.1109/ICL.2012.6402136 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6402136&isnumber=6402022

[6] Raspbian, offcial site: http://raspbian.org/ consulted on 8th january 2015.

[7] Web Sockets protocol, reference site: http://www.w3.org/TR/2009/WD-websockets-20091222/ consulted on 8th january 2015.

[8] Node.js, official site: http://nodejs.org/ consulted on 8th january 2015.

[9] Tornado, official site: http://www.tornadoweb.org/en/stable/ consulted on 8th january 2015.

[10] NPM repository, official site: https://www.npmjs.com/ consulted on 8th january 2015.

[11] Autobahn WebSockets Testsuite Report: http://autobahn.ws/testsuite/reports/clients/index.html consulted on 8th january 2015.

[12] Carro Fernandez, G.; Gutierrez, S.M.; Ruiz, E.S.; Gil, M.C.; Perez, F.M., "Formation in robotics, the key to integration in industrial environments,"*Interactive Collaborative Learning (ICL), 2012 15th International Conference on*, vol., no., pp.1,4, 26-28 Sept. 2012 doi: 10.1109/ICL.2012.6402135 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6402135&isnumber=6402022

[13] Sowe, S.K.; Kimata, T.; Mianxiong Dong; Zettsu, K., "Managing Heterogeneous Sensor Data on a Big Data Platform: IoT Services for Data-Intensive Science," Computer Software and Applications Conference Workshops (COMPSACW), 2014 IEEE 38th International , vol., no., pp.295,300, 21-25 July 2014 doi: 10.1109/COMPSACW.2014.52 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6903145&isnumber=6903069

[14] Gupta, V., "Working and analysis of the H - bridge motor driver circuit designed for wheeled mobile robots," Advanced Computer Control (ICACC), 2010 2nd International Conference on , vol.3, no., pp.441,444, 27-29 March 2010 doi: 10.1109/ICACC.2010.5486818, URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5486818&isnumber=5486614

[15] Altin, H.; Pedaste, M.; Aabloo, A., "Educational Robotics and Inquiry Learning: A Pilot Study in a Web-Based Learning Environment," Advanced Learning Technologies (ICALT), 2011 11th IEEE International Conference on , vol., no., pp.224,226, 6-8 July 2011 doi: 10.1109/ICALT.2011.72, URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5992308&isnumber=5992239

[16] Carro Fernandez, G.; Gutierrez, S.M.; Ruiz, E.S.; Mur Perez, F.; Castro Gil, M., "Robotics, the New Industrial Revolution," *Technology and Society Magazine, IEEE*, vol.31, no.2, pp.51,58, Summer 2012 doi: 10.1109/MTS.2012.2196595 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6213867&isnumber=6213848