SpaceMonitor

Abstract

by Gaetan Martin, Florian Henriet, Alexey Tukalo

Introduction

The main idea of the project was to build the system with multiple transmission of data via network. The concept of IoT[1] fits for this purpose well.

We decided to create system for storage space monitoring. The system contains:

1. **Microcontroller**
   We chose Intel Galileo[2] as microcontroller for this project, because it is compatible with usual Arduino[3] framework, but much more powerful than traditional Arduinos. It has build-in Ethernet port and the fact makes the data transmission to the server easier for us. We connected 3-pin Ultrasonic sensor to the board via analog port. The sensor allows the devices measure the amount of space available on a shelf, if the volume is changed the Galileo will send the information to the server. The data is transmitted with help of MQTT[4] as JSON object with tree parameters: size of shelf, free space and name of the shelf.

2. **Server**
   We store our server inside IBM BlueMix[5].The logic is implemented via Node-RED [6]. The data is received by IBM IoT App In node, after that server adds date and time to the data and put the JSON object inside MongoDB[7]. The data can be requested by webpage form the server throughout HTTP GET request, the server sends an array of JSON objects from the database as response.

3. **Webpage**
   The webpage uses RESTfull API provided by our server to receive data. AJAX[8] is implemented with help of D3.js[9]. The received array is preprocessed inside callback function. The preprocessing is performed to separate values from different shelves. SVG image basically consists several independent shapes and together with smart system of event handling implemented via chain syntax it makes D3.js very powerful tool for really interactive visualisation of data. The information about storage space is represented in 5 different graphs: two of them are donut charts, a bar chart and a plot. The layout of the page is made in according with the concept of fluid interface, so it is able to change position of the legend for different screen sizes.

Conclusion

Our system shows the information about usage of storage place in real time, for this reason we transmit data from the sensors on the storage to the server and from the server to the web-browser of the storage owner. The end user is able to use our visualisation for analysing of efficiency of the shelves system and control amount of available space.

References:

[1] D3.js documentation (November 19, 2015)
[2] D3.js tutorial by Scott Murray (November 19, 2015)
[3] Interactive Data Visualization for the Web by Scott Murray (November 19, 2015)
[4] Ultrasonic Range detector tutorial (November 19, 2015)
[5] Build a cloud-ready temperature sensor with the Arduino Uno and the IBM IoT Foundation (November 19, 2015)
[6] Galileo Getting Started Guide (November 19, 2015)

[1] Physical objects connected via network with other devices or server. In most cases they are used to perform measurements or control under some infrastructure.
[2] Microcontroller board based on a 32-bit Intel Pentium-class system on a chip
[3] An open-source electronics platform for anyone making interactive projects
[4] Message Queue Telemetry Transport - asynchronous light weight data transmission protocol for IoT
[5] Cloud platform that helps developers rapidly build, manage and run web and mobile applications.
[6] Node-RED is a tool for wiring together hardware devices, APIs and online services in new and interesting ways.
[7] Database stores the data as set of JSON objects.
[8] Web development techniques used on the client-side to create asynchronous Web applications.
[9] Data-driven document - JavaScript library made for custom data visualisation with SVG inside webpage