Infrastructure & Design of Embedded Connected-Object Services: Application to Activity Daily Live monitoring

M-A Peraldi-Frati, J-L Salvat, N Le Thanh, T-H Hoang, T.-H.-H Nguyen

LABORATOIRE

INFORMATIQUE, SIGNAUX ET SYSTÈMES DE SOPHIA ANTIPOLIS

UMR7271

Infrastructure & Design of Embedded Connected-Object Services:
Application to Activity Daily Live monitoring

M-A. Peraldi-Frati[1], J-L. Salvat[2], N. Le Thanh[1], T.T-H Hoang[3], T.-H.-H. Nguyen[4]
[1] Côte d'Azur Univ. - I3S Lab. CNRS- UMR 7271 F-06900 Sophia Antipolis – France
[2] Nice Sophia Antipolis University F-06041 Nice Cedex
[3]Danang Univ. of Economics, The Danang University, 41 Le Duan, Danang, VN
[4]VietNam-Korea friendship IT College, Danang, VN
map@unice.fr, nlt@unice.fr, salvat@unice.fr, ha.htt@due.edu.vn, huenth@viethanit.edu.vn

Rapport de Recherche

Mai-2018

Laboratoire d'Informatique, Signaux et Systèmes de Sophia-Antipolis (I3S) - UMR7271 - UNS CNRS
2000, route des Lucioles – Les Algorithmes - bât. Euclide B – 06900 Sophia Antipolis – France
http://www.i3s.unice.fr

Infrastructure & Design of Embedded Connected-Object Services:

Application to Activity Daily Live monitoring

Abstract: The design of services based on a network of connected objects requires the integration of multiple constraints from different domains (electronics, communication, software) interacting throughout the value chain. When these services are applied to monitoring the activities of daily living (ADLs) of the elderly, additional constraints related to respect for private and family life must also be taken into consideration. The paper presents the IoT infrastructure that supports the corresponding service. A software and hardware architecture is proposed; as well as the various technical choices; all being guided by these requirements. Experimental results developed in a real environment are presented and provide us with feedback to improve the solution.

Key-words: IoT services, Connected Objects Infrastructure, Wireless Sensor Network, ADL (Activity of Daily Living), Smart Home, Elderly Monitoring.

Infrastructure et conception de services pour objets connectés embarqués :

Application au suivi d'activité

Abstract: La conception de services basés sur un réseau d'objets connectés nécessite l'intégration de contraintes multiples provenant de différents domaines (électronique, communication, logiciel) interagissant tout au long de la chaîne de valeur. Lorsque ces services sont appliqués au suivi des activités de la vie quotidienne (ADL) des personnes âgées, des contraintes supplémentaires liées au respect de la vie privée et familiale doivent également être prises en considération. Le document présente l'infrastructure IoT qui supporte le service correspondant. Une architecture logicielle et matérielle est proposée, ainsi que les différents choix techniques, le tout guidé par les exigences et contraintes de la chaîne de valeur. Les résultats expérimentaux développés dans un environnement réel sont présentés et nous fournissent un retour d'expérience pour améliorer la solution.

Key-words: Services IoT, Infrastructure pour objets connéctés, Réseaux de capteurs, ADL (Activity of Daily Living), Maison connectée, Suivi de personne.

Infrastructure & Design of Embedded Connected-Object Services:

Application to Activity Daily Live monitoring

M-A. Peraldi-Frati[1], J-L. Salvat[2], N. Le Thanh[1], T.T-H Hoang[3], T.-H.-H. Nguyen[4]

[1] Côte d'Azur Univ. - I3S Lab. CNRS- UMR 7271 F-06900 Sophia Antipolis – France
[2] Nice Sophia Antipolis University F-06041 Nice Cedex
[3]Danang Univ. of Economics, The Danang University, 41 Le Duan, Danang, VN
[4]VietNam-Korea friendship IT College, Danang, VN
map@unice.fr, nlt@unice.fr, salvat@unice.fr, ha.htt@due.edu.vn, huenth@viethanit.edu.vn

Abstract. The design of services based on a network of connected objects requires the integration of multiple constraints from different domains (electronics, communication, software) interacting throughout the value chain. When these services are applied to monitoring the activities of daily living (ADLs) of the elderly, additional constraints related to respect for private and family life must also be taken into consideration. The paper presents an IoT infrastructure that supports the corresponding service. A software and hardware architecture is proposed; as well as the various technical choices; all being guided by the requirements and constraints of the value chain. Experimental results developed in a real environment are presented and provide us with feedback to improve the solution.

Keywords: IoT services, Connected Objects Infrastructure, Wireless Sensor Network, ADL (Activity of Daily Living), Smart Home, Elderly Monitoring.

1 Introduction

The value chain for designing services based on connected objects requires different areas of expertise. This is particularly true when these services are applied to the remote monitoring of elderly people in their daily activities. Through these services, it is a question of identifying changes in people's behavior or of detecting a critical situation such as a fall or a lasting lack of activity so that families can react and have the appropriate response to this situation (visit, activation of help, information from the attending physician, etc.).

Thus, the development of connected object services for ADL management requires the integration of multiple and heterogeneous requirements in:

Electronics: for the design of multimodal sensors nodes and embedded architecture of different technologies that must meet requirements linked to performance, energy consumption, data transmission, spatial positioning, packaging, reliability, cost,….

Communication protocols and standards: for the interconnection of sensors with the local or remote parts of the IoT infrastructure which manage at the same time, the configuration of the sensors, the transmission policy and the range of the data flows, their security,...

Social and human sciences: by integrating the specific needs of users of these services such as the acceptability, the portability and non-intrusiveness of the equipment used, but also respect for the user's privacy (confidentiality of information, trust in the service, integrity of people, cost, etc.)

IT: by providing a software environment able to capture the previously-mentioned requirements for modeling the IoT infrastructure and the services deployed on it.

This article presents results obtained within the framework of the SLEGO project, carried out the DNIIT Institute. The project aims at developing domain specific services based on sensors network systems, which enable caregivers to remotely verify activities of elderly people under their care. SLEGO addresses the overall value chain of this design, and associated multi-disciplinary requirements. Thus, SLEGO offers both an open electronic infrastructure and the software support to program it. This package is based on royalty-free technologies. SLEGO provides a software environment dedicated to the providers of these services and those who describe the ADL scenarios to follow. The environment is based on the definition of a domain-specific language (DSL) to model and simulate IoT activities and infrastructure. This environment relies on the Gemoc Studio and encompasses Gemoc's modelling and simulation capabilities.

The document is organized as follows: Section 2 identifies work in progress and platforms related to the detection and monitoring of ADLs, assessing for each of them, their main characteristics and their adequacy with the value chain requirements. Section 3 provides a case study, and lists and classifies the requirements of the value chain. Section 4 presents the different stages of the software and hardware infrastructure that implements the services. Then, we conclude on the results of this experiment and the work in progress.

2 Societal and technical context for services

The ratio of population [1] aged of 65+ in Western Europe will be 37,8% in 2030 (30,6 in 2015). Maintaining elderly people at home is identified as an important societal challenge. A lot of smart-home research platforms have been tested in different countries. Let us cite GerHome Project in France CSTB [2], Tech Aware Home (Georgia Tech), Casas in Spain [3] which are based on sensors, cameras and associated services to monitor the activity of the person by maintaining a permanent interaction with caregivers, family, and medical staff.

These experimental platforms have been developed for generating datasets for the research community interested in real data analysis. The Placelab [4] or Care project [5] are examples of effective datasets used by researcher to evaluate their methods of activity recognition. These platforms are not intended for the consumer market and the use of cameras does not respect the neutrality of the application in term of privacy.

Concerning the electronic part, these platforms are based on sensor nodes also developed in the framework of research. These nodes are mainly based on a communication platform and the sensors connected either on I/O ports [6,7,8] or directly on board [9]. The PCB and the communication medium are carefully chosen according to some optimization objectives for signal strength, transmission reliability and consumption. These solutions are often proprietary and expensive because of the low quantities of units produced. They do not fit with the low cost constraints mandatory for consumer application.

Since 2010, the programming revolution of open platform such as Arduino [10] or the Mbed[11] solution for Arm processor, has enabled the creation of faster and cheaper board prototypes.

Thanks to theses open-source solutions, a sensor node can be bought and built easily, and the associated software can be implemented rapidly. Sensebender[12] or Moteino[13] are example of low cost Printed Circuit Board (PCB). These solutions have changed the way to implement low cost Wireless Sensor Nodes (WSN) compared to previous more costly and complex solutions.

Others wireless standardized solutions are used to communicate between sensors[14]. More expensive are platforms that implement the new communication protocols dedicated to the IoT and IPv6 such as Thread and 6LoWPAN [15]. For sensors nodes, these technologies prove to be not optimal: using Wifi consumes too much energy, while BLE operates at a range that is too low, the Z-wave and 6LoWPAN are either too expensive or too complex to implement (6LoWPAN needs at least MCU ARM 32 bits to be implemented).

These later protocols are more adapted for the Gateway to bridge with the Internet. The top candidate platforms for the Gateway are the Linux Raspberry or his clone (Nanopi, Orange pi, Banana Pi,…) thanks to his connectivity as Ethernet, WiFi or 3G and his General Purpose Input/Output (GPIO) header where RF communication board can be plugged. On the gateway open source protocols can be deployed such as MQTT [16] a publish-subscribe-based lightweight messaging protocol for small sensors and mobile devices, REST [17] allowing state transmission between a web-based client and a server or JSON [18] for data-interchange. All these protocols furnish open source APIs for connecting to a Cloud solution. As a consequence, these solutions allow accelerating the time and limiting the cost for developing a prototype necessary to experiment activity learning in multiple environment.

3 Objective & architecture of SLEGO

3.1 Context

The SLEGO project is developed in the context of the DNIIT institute[1], a joint research institute of the University of Danang and the University Côte d'Azur.

The project aims at developing services for connected objects dedicated to specific application domains. We consider objects endowed with multi-modal sensing or actuating capabilities, connected through heterogeneous communication media. Services are described through scenarios by using Domain specific languages (DSL) able to tackle the different needs of these IoT services. The overall system can be qualified as a cyber physical system.

3.2 Case study and associated requirements

The case study focuses on monitoring the activities of daily living (ADL) of a person with loss of autonomy. The system includes a network of sensors deployed in the elderly person's apartment. The activities to be followed are expressed through behavioral scenarios. Anticipated and unexpected scenarios are considered. Services are distributed on a computing and communication infrastructure at different levels.

The use case focuses on two activities. The first concerns the detection of showering. The second is an assessment of a person's daily physical activity, that is, how they move from one room to another.

A list of requirements for these services is given below and categorized as Function-al (F), Non-functional (NF) :

- ***Req1-F****: the service must detect & memorize a person taking his shower,*
- ***Req2-F****: the date, duration of the shower must be calculated & memorized*
- ***Req3-F:*** *the service must detect & memorize the physical activity of the person.*
- ***Req4-F:*** *the movement tracking information allow a reconstruction of the path taken by the person in the apartment.*
- ***Req5-F****: Configuration of services must be done at software level.*
- ***Req6-N-F-Archi:*** *Sensor/Actuator positions should be reported to services.*
- ***Req7-N-F-Archi:*** *Sensor/Actuator information can be shared by # services.*
- ***Req14-NF-Archi:*** *Electrical network of the apartment will not be modified*
- ***Req8-NF-Privacy:*** *Sensors/actuators should be not intrusive for a person*
- ***Req9-NF-Security:*** *Sensor Data should be reliable (time, space, value)*
- ***Req10-NF- Security:*** *Data should be kept confidential*
- ***Req11-NF-Perf.****: Power autonomy of sensor should be at least 2 years*
- ***Req12-NF-Perf.:*** *Sensor acquisition policies should be either periodic or on interrupt*
- ***Req13-NF-Archi:*** *The amount of data transferred over the internet should be limited*

[1] DNIIT: Da Nang Institute of Information and Technology

4 Hardware and Software Infrastructure for Services

Based on these requirements, some choices have been made to organize the computation and communication infrastructure. **Fig. 1** draws the hardware and software multi-layered infrastructure. The infrastructure is organized into layers with dedicated objectives.

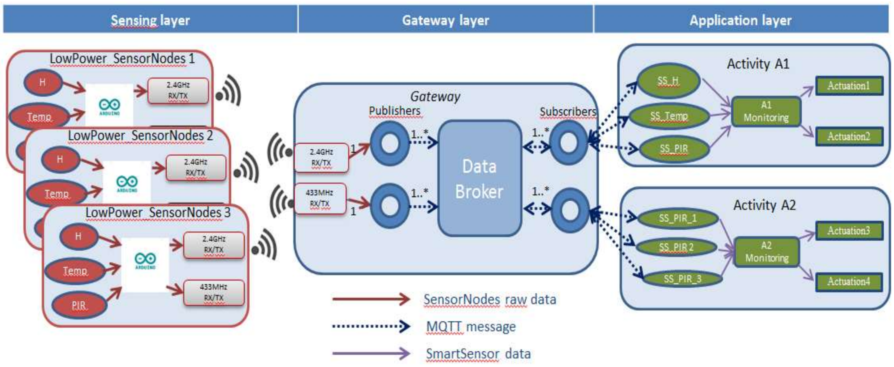

Fig. 1. IoT Hardware/Software infrastructure

The sensing layer: includes different kinds of sensors: humidity, PIR, temperature, luminosity. Multiple sensors can be combined and embedded on a single platform called a Sensor node. Sensor nodes provide computing and communication capabilities. They embeds multiple sensors/actuators, a PCBoard, and communication modules. A Sensor node produces, and transmits raw data from embedded sensors to the gateway with different policies either periodically or on interrupt via communication channel *(Req12).* The node must be low power and comply with energy constraints *(Req11 & Req14)*.

The Gateway layer collects and dispatches the raw data through different protocols. This layer embeds *communication modules* for reading/writing the environment (sensor node side) and transfers data with parsimony *(Req13)* to the software layer side. Features for securization and confidentiality of transported data are available on the gateway *(Req9 & 10).* The layer includes a *Sensor Data Broker* which collects sensor data sent by agents called publishers and makes these data available to subscriber agents which have subscribed to a specific set of data called a topic *(Req7).*

The software layer contains multiple activity nodes. Each one tracks a specific activity (e.g., taking of shower, taking of meal, rest period in armchair, night's rest, …). *(Req 1 to Req5).* Each activity needs specific data. To limit the amount of data, agents called SmartSensor manage raw data and filter it in order to detect specific patterns (*Req13*). In this layer, SmartSensors combine and fuse multiple data and store information in a database that can be visualized in a monitor This layer has also a monitor which implements a control determining the execution of a specific activity.

Details on these layers are given bellows.

4.1 Sensors system technological choices

Technological choices regarding sensors, are of the highest importance in the value chain of connected-objects services. The main criteria are the topology of the sensor network for the coverage, protocols and their openness for an efficient transfer of data, the communication technology for a good balance between distance and range, and the object autonomy linked to the sensor technology and their usage context.

- **Network topology:** In the context of network sensors, the topology can be either a mesh mainly for outdoor applications or a star network for indoor systems. Mesh architecture propagates information through nodes with additional routers or repeaters powered by AC line to relay messages to a gateway. This topology does not correspond to our needs (all sensors node are powered by battery). The alternative we chose is a star topology with a centralized controller as a gateway that receives and sends messages to the nodes. This topology is simplest to implement and allows sensors nodes on battery.
- **Network protocol:** Sensor nodes should to be cheap, used by community to permit a large choice of low-cost boards (communication platforms without or with sensors). Solution with Hope RF RFM69CW or Nordic nrf24L01 with an Arduino MCU are currently the cheapest. Protocols proposed for these two RF platforms include CRC verification, and in the case of RF24 or RF69, these protocols are implemented in mysensors library that allows acknowledgement and recovery of message due to interference or collision.
- **Communication technology:** Reliability of transmission is an important requirement. For radio communications, the balance between range of frequency and gain of antenna must be found. Depending of obstructions and reflections of walls, the range of sub-GHz radio is almost 3 (868MHz) to 5 times (433MHz) longer than 2.4GHz frequency, The shape and size of the antenna must comply with mechanical constraints of the box where to put the electronics and battery. Based on these criteria we opted to a helicoidal antenna with the RFM69CW (433Mhz) communication protocol.
- **Sensors technology:** The main requirements for sensors embedded in sensors nodes are their price, their surface and their power consumption. Multiple sensor kinds were required: luminosity sensors, temperature and humidity sensors …. We have chosen sensors available in PCB board, easily soldering, low price and very low power. Apart from these sensors, the project also requires a movement sensor. Passive Infra-Red (PIR) sensors are very sensible to Fresnel Lens so we opted to reliable commercial PIR sensor 433 MHz, with a price less than 5$ and compatible with RCswitch library in order to communicate with the gateway

4.2 Experimentation of sensor nodes.

Experimenting a DLA service need the installation of 7 to15 sensor nodes per apartment. The deployment of sensors nodes in an apartment must be possible without any electrical infrastructure changes *(Req14)*. In our case, sensors nodes must be energetically autonomous and in live for two years *(Req11)*. Three solutions of node sensor platforms have been tested in the effective operating environment i.e: the apartment of an elderly person.

1. A first node sensor, not represented here, uses a low power MCU atmega328 (sensebender) with temp/hum sensor and communicate through a NRF24L01+ (2.4GHz). The receptor was a banana Pi M2+ with an nRF24L01+ ultra- long-range and a whip antenna (Gain = 2dBi). Tests raise some reliabilities issues i.e.: loss of data with node sensors the most distant from the gateway.
2. The second platform (**Fig.2a**) uses Promini board on a PCB designed to integrate sensors and RFM69CW with a helicoidal antenna. A PCB board (not assemble) was designed and manufactured (150 units, 2$ per unit). The receptor was a banana Pi M2+ with a RFM69HCW and a whip antenna (Gain = 2dBi).
3. The third platform (**Fig.2b**) is a board with MCU, sensors and RFM69CW + SMD antenna assembled This solution was tested on a prototype ($500) but the price (50 units, $30 per unit) was too expensive to go further even if the range of antenna was better than solution 2.

After the installation, comes the exploitation of the service. The node consumption is another important criterion during this phase. Harvesting techniques associated with sensors, MCU, and RF modules, manage energy with different policies for collecting sensors data. SensorNode consumption depends on a sampling policy, a rate of sensors, an amount of data sent by the MCU, a speed of communication and a potential local processing on the MCU.

To limit the consumption, the policy we implemented on our nodes consists of a lazy evaluation of the environment based on the wake-up/ asleep of the MCU processor according to a parametrable period of time and/or on an interrupt. The consumption has been evaluated on different boards. For ex. CPU Atmega328p consumes 500 uA at 1 MHz and 4,5 uA in sleep mode. Implementing these techniques allow node sensors to be powered by batteries.

(a) (b) (c)

Fig. 2. : Prominino & Nodeus PCBs, & Communication gateway

We measured the consumption per node. Results are reported in Table 1. Raw data from sensors needs 14 octets to be sent. At a rate of 38400 bauds it takes 2,9ms to deliver data with a power of 13dBm. In this context consumption of RF module is 45mA. If data are sent every 5min, the average consumption is 45mA*0,0029/(5*60)<1uA and if period equal 30s, consumption is

45mA*0,0029/30<5uA. In between, MCU and RFM69CW are sleeping and consume 4,5uA + 0.9uA. Total consumption without sensors is less than 10.4uA if data are sent every 30s.

RFM69VW state	Power Consumption
Sending (38.2kbps) 13dBm	45mA
Sending (38.2kbps) 0dBm	12,6mA
Reception	16mA
Sleeping	900nA

Table 1. Consumption of RF module

4.3 Embedded system: gateway design

The gateway is the second layer of the infrastructure. As its name suggests, the gateway provides a link between multiple communication technologies. In our case, the gateway speaks 868 MHz with sensor nodes (with RFM69CW RF module) and sends data to the application layer via Wifi 802.11 frames or Ethernet. **Fig.2c.** shows two antennas that support communication with node sensors 868MHz and PIR sensors 433MHz. A dongle for GSM communication can be added. In this experiment, we used Ethernet.

Processing on raw-data is provided by a banana pi M2+ through a publish-subscribe policy based on topics. A client publishes sensor data; tagged with a certain topic; to a broker which in turn publishes these data to one or multiple subscribers of this topic. A topic is a hierarchical naming structure that references the sensors of your application. The broker is based on a lightweight messaging transport protocol dedicated to IoT, called MQTT [16]. The payload part of a MQTT frame has an ad hoc format. In our case the format for a Sensor (S) data includes different fields:

Version	TimeZone	Date	Hour	UserId	SKind	SUnit	SValue	SId	SName	SLocation

Fig. 3. : MQTT Payload format

The definition of these fields, as well as their size has been driven by the application needs and the exploitation of data.

4.4 Software layer: smart sensors and activity design

The software layer performs data processing at the edge of the network infrastructure instead of transmitting the data and postponing their processing on external servers. Edge computing is a local and ad-hoc computing differs greatly in complexity compared to processing in the cloud. Three kinds of processing are performed:

Smart sensor computing:

A Smart sensor receives a large amount of data coming from sensors nodes. It takes multiple samples of raw data as inputs and can play two main roles. If data come from one sensor, it filters the data by marking changes in the data's evolution. The second role is to fuse raw data coming from multiple sensors to corroborate and build the environment state. In our application, an example of smart sensor is the threshold detection for humidity level that marks that a shower is starting or stopping. When raw data from humidity sensors are produced every 5 minutes, the corresponding SmartSensor deliver 2 or 4 datasets of information per day.

Activity processing:

An activity is a specific action that an elderly person carries out and on a daily basis. An example of activity is "taking a shower". The objective is to report when the activity starts and stops. The Shower activity takes inputs from multiple smart sensors. From these smart data, a monitor tracks a normal activity by comparing to an expected pattern or detects an abnormal activity by using rules. In case of abnormal behaviors, activity monitor can raise an alert to end- user.

Actuation processing:

As a reactive system can do, the result of the monitoring may give rise to different actions: actuation (turning on a light, switching off the heating,...), reporting (sending an alarm via an email or sms) or for or recording (raw or smart data recording in the associated databases in the cloud).

5 Conclusion

This paper gives the results of the SLEGO project which aims at developing the hardware and software infrastructure for services dedicated to the supervision of ADLs of the elderly.

The originality of this work is to consider the entire value chain for ADL services from the sensors to the cloud, including hardware, software and communication parts and mixing the associated technologies.

The different technological choices of this three-layered infrastructure have been presented, driven by specific application requirements. These requirements can be of different types. For example, a choice of sensors are based on: the respect for private life and the comfort of the user, their location linked to the building structure, the energy efficiency linked to the usage and the expression of ADL scenario with a specific syntax.

The paper mainly focuses on two layers of the infrastructure i.e.: the sensor node and the gateway layers. The presentation of Domain specific language for scenario description is out of the scope.

A proof of concept of this platform and the associated services has been deployed in DNIIT institute.

We will capitalize on the results obtained concerning the layers 1 and 2 to enrich and extend the Domain Specific Language i.e.: the scenario language for activity modelling but also a language to model the allocation of applications onto this hardware and the location of hardware/sensors inside a building.

References

1. DeSA, U.N. *World population prospects: the 2012 revision*. Population division of the dep. of economic and social affairs of United Nations, New York.
2. Pomponio, L., Le Goc, M., Anfosso, A. & Pascual, E. (2012). *Levels of abstraction for behavior modeling in the gerhome project*. Int. Journal of E-Health and Medical Com., 3(3), 12-28.
3. Cook, D. J., Crandall, A. S., Thomas, B. L., & Krishnan, N. C. (2013). CASAS: *A smart home in a box*. Computer, 46(7), 62-69.
4. Intille, S., Larson, K., Tapia, E., et al. *Using a live-in laboratory for ubiquitous computing research.* Pervasive Computing, 349-365, (2006).
5. Van Kasteren, T., Noulas, A., Englebienne, G., & Kröse, B. *Accurate activity recognition in a home setting.* In Proc.10th ACM Int. Conf. on Ubiquitous Computing (pp. 1-9). Sept. 2008.
6. García-Hernando, A. B., Martínez-Ortega, J. F., López-Navarro, et al. *Hardware Platforms for WSNs*. In Problem Solving for Wireless Sensor Networks (pp. 1-34). 2008.
7. Akl, R., & Saravanos, Y. (2007, September). *Hybrid energy-aware synchronization algorithm in wireless sensor networks*. In Personal, Indoor and Mobile Radio Communications. PIMRC 2007. IEEE 18th Int. Symp. on (pp. 1-5).
8. Prayati, A., Antonopoulos, C., Stoyanova, T., Koulamas, C., & Papadopoulos, G. (2010). *A modeling approach on the TelosB WSN platform power consumption.* Journal of Systems and Software, 83(8), 1355-1363.
9. Wille, S., Wehn, N., Martinovic, I., Kunz, S., & Göhner, P. (2010). *AmICA-Design and implementation of a flexible, compact, and low-power node platform.*
10. Barrett, S. F. (2013). *Arduino microcontroller processing for everyone! Synthesis Lectures on Digital Circuits and Systems*, *8*(4), 1-513
11. Toulson, R., & Wilmshurst, T. (2016). *Fast and effective embedded systems design: applying the ARM mbed*. Newnes
12. SenserBender micro gateway https://www.mysensors.org/hardware/micro
13. Moteino microcontroller https://lowpowerlab.com/guide/moteino/
14. Reiter, G. (2014). *Wireless connectivity for the Internet of Things*. Europe, 433, 868MHz.
15. Olsson, J. (2014). *6LoWPAN demystified*. Texas Instruments, 13.
16. Tang, K., Wang, Y., Liu, H., Sheng, Y., Wang, X., & Wei, Z. (2013, September). *Design and implementation of push notification system based on the MQTT protocol.* In Int. Conf. on Information Science and Computer Applications (ISCA 2013) (pp. 116-119).
17. Colitti, W., Steenhaut, K., De Caro, N., Buta, B., & Dobrota, V. (2011, October). *REST enabled wireless sensor networks for seamless integration with web applications.* In IEEE 8th International Conference on Mobile Adhoc and Sensor Systems (MASS), 2011 (pp. 867-872).
18. Crockford, D. (2013). *The json data interchange format.* Tech. report, ECMA Int..