Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej Nr 52

IV Konferencja

e-Technologies in Engineering Education eTEE'2017

Politechnika Gdańska, 27-28 kwietnia 2017

SENIOR LAB PROJECTS FOR TEACHING THE INTERNET OF THINGS IN A SOFTWARE ENGINEERING PROGRAM

Fernando GONZALEZ, Dahai GUO, Adam NOWICKI, Janusz ZALEWSKI

Dept. of Software Engineering, Florida Gulf Coast University, USA
tel.: +1 (239)590-7317 e-mail: {fgonzalez,dguo,anowicki,zalewski}@fgcu.edu

Abstract: This paper addresses student laboratories for courses related to the Internet of Things (IoT) in an undergraduate Software Engineering program. It reviews the concept of the IoT, first, then discusses benefits of IoT in education, especially in engineering, and finally presents examples of projects, reviewing some more fundamental concepts of introducing such labs. Specific examples of IoT projects include software development for: a robotic arm accessed through AWS, GPS tracker with Sparkfun data stream service, online health monitoring with a smartwatch and Google Cloud, and remote relay access from a phone with MQTT service.

Keywords: Internet of Things, cloud computing, engineering education, software engineering education, online labs.

1. INTRODUCTION

E-technologies, such as, Internet of Things (IoT) or Cloud Computing are definitely making their way into teaching and learning, but there is very little experience or information how to use them effectively in education, especially, in education of engineers. It is the fact of the matter that especially the IoT is a disruptive technology in many industries and in business in general. Actual numbers may vary by source but the consensus is that the volume of IoT connected devices will grow somewhat unpredictably to billions of units in the next decade. So will grow the market value, likely reaching trillions of dollars in the same period.

Famously, the early predictions, 2010-11, were a bit off. For example, Ericsson [1] gave an estimate of 50 billion devices interconnected by 2020, and a CISCO executive independently reconfirmed this early estimate [2]. Even though these specific predictions were inaccurate in absolute numbers, it is necessary to realize, what impact the IoT will have on the world population. As Figure 1 [2] shows, growth of the number of connected devices per person appears to double every five years, at least, in the reported period.

These early predictions are now followed by more realistic statements based on updated research. For example, as summarized in [3], current estimates include the following numbers of devices by 2020:

- 28 billion, as corrected value by Ericsson (by 2021)
- 30 billion, corrected by former CISCO executive
- 30.7 billion by IHS Markit
- 28.1 billion by International Data Corp., and
- 30.7 billion in a study by Gartner.

This is all reflected in Figure 2, including previously given inaccurate estimates [4].

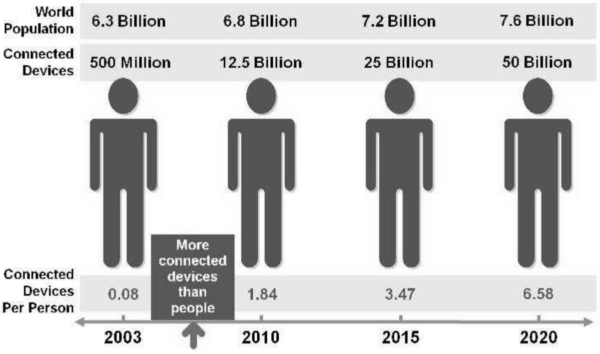

Fig. 1. Predicted increase in the number of connected devices per person (adopted from [2])

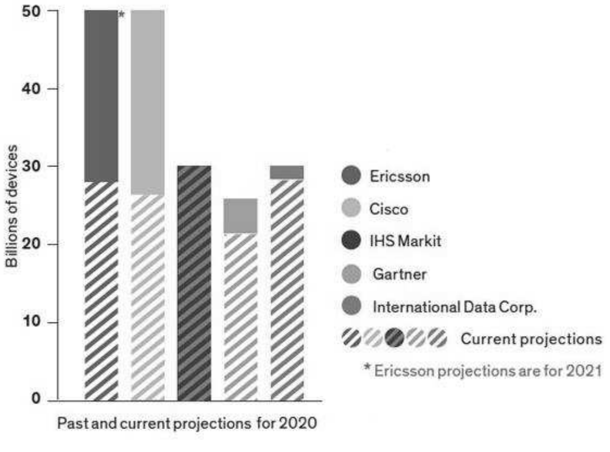

Fig. 2. Past and current projection for the number of interconnected devices (adopted from [4])

It is, therefore, clear that with numbers this big, education will be heavily impacted, but for the moment nobody knows how? Thus, given the pervasive nature of IoT, it is necessary to address the problem in education of engineers. Teaching how to design and implement IoT is essential to the profession. The rest of this paper is organized as follows. Section 2 gives an overview of the Internet of Things from the technical perspective, Section 3 discusses other work, Section 4 presents the IoT labs themselves, and Section 5 ends the paper with conclusion.

2. INTERNET OF THINGS TECHNICAL OVERVIEW

2.1. Basic Architecture

The IoT does not appear to have a single, widely adopted, definition. However, one particular definition should appeal more to the professionals, since it comes from an engineering society and reads as follows [5]-[6]:

„Internet of Things (IoT) is a system consisting of networks of sensors, actuators, and smart objects whose purpose is to interconnect "all" things, including everyday and industrial objects, in such a way as to make them intelligent, programmable, and more capable of interacting with humans and each other".

There are a number of characteristics, which can be attributed to the IoT. The most important ones are its architectural components, which can be listed as follows:

- smart devices at the user end
- communication infrastructure for connectivity
- computing cloud to provide data storage, and
- analytics tools at the cloud level.

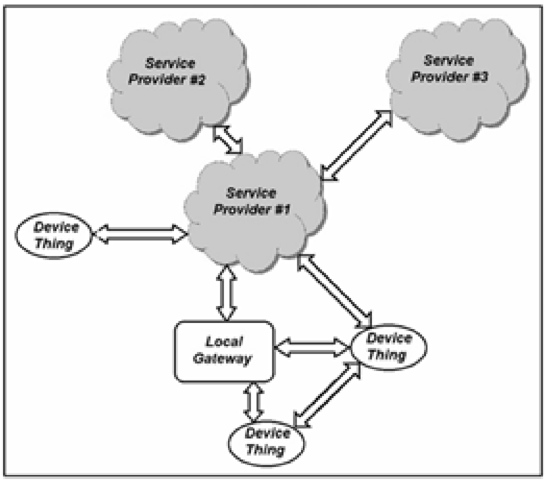

Fig. 3. Overall architecture of the Internet of Things

As shown in Figure 3, there are multiple devices ("things", some smart) at the user end, a communication infrastructure with devices accessing the cloud directly or via intermediaries, such as local gateways, and service providers in the cloud equipped with appropriate analytical tools. These are the critical constituents of the IoT, forming its architecture compliant with the one adopted by Intel [7].

2.2. Device Things Layer

Figure 4, adopted from [8], shows from a different angle, how the IoT definitions map on the practical architecture of the Internet of Things. There are always multiple data sources, these "things". They are represented by instances of: pressure transmitters, lighting system, coffee maker, washing machine, dishwasher, guitar, car and more.

The layer of "things" may include all sorts of data sources but also data sinks, that is, devices that are just recipients of data, for example, for display or control: light emitting diodes (LED's), LCD displays, street lights, door locks, relays, rotors and motors, 3D printers, even speakers, etc. So, one has to think about this layer as a device layer, which includes sensing and actuating devices, that is, data sources (senders) but also data sinks (recipients). These device things, as it is clear from both sample lists above, may have various knowledge about themselves and the surrounding world, that is, rudimentary intelligence.

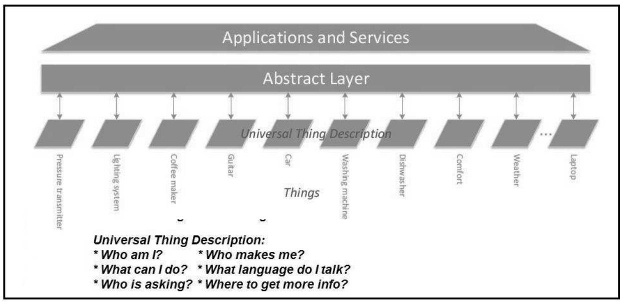

Fig. 4. IEEE P2413 IoT Application Framework [8]

An Abstract Layer must consist of the communication infrastructure as well as the servers, which host the services. The "communication infrastructure" is the Internet itself, but can be any network, and the "servers" are just computers embedded in the cloud. The Applications and Services layer becomes an "intelligence" layer, offering related processing services, analytics, and decision support.

From the technical standpoint, to develop the IoT at the Device Thing level, it is important to understand building individual components and programming them, with knowledge of the communication infrastructure. The low-level communication is also important and involves wireless standards, such as Bluetooth, Zigbee, RFID and NFC.

2.3. Merging with the Cloud

Given the large variety of sensors that can be deployed ubiquitously in an IoT system, a large volume of data may be generated at a high velocity. With the rapid growth of cloud computing, many of the Big Data challenges have been effectively addressed. The two front runners in providing cloud computing, the Amazon Web Services (AWS) [9] and Microsoft Azure [10] offer such services,.

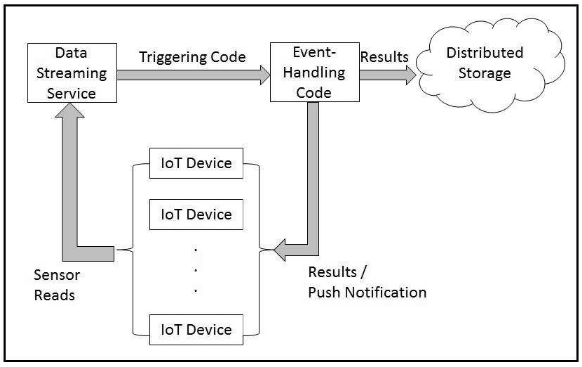

Fig. 5. Example of using cloud services to build an IoT system

An essential service in the cloud is called auto-scaling, which allows engineers to dynamically provision resources (computing power and storage) based on the current demand. Both AWS and Azure offer services that support data streaming, push notifications, event-driven architecture, which can be used in developing software to respond to individual reads. Figure 5 shows the devices linked via the Internet to the data streaming service which then invokes different programs to respond to different events. Results are stored in a distributed storage or pushed to other devices.

Using these services, software engineers only need to focus on how each read should be handled, without worrying about networking, programming dependencies, runtime environments, data consistency, synchronization, and dynamic provisioning. Cloud vendors even provide IoT services, e.g., AWS IoT [9] and Azure IoT Suite [10].

3. PERSPECTIVES FOR IoT USE IN EDUCATION

From the perspective of this paper, one can distinguish three essential and significantly different levels of IoT use in education: (1) education in general, (2) higher education, and (3) engineering education; all topics are discussed next.

3.1. IoT Use in Education in General

Prospects of using IoT in education, in general, are articulated the most vocally, among others, by computer companies, which sense a big business just around the corner. Such examples are CISCO [4] and Intel [11], which beyond hidden advertising provide valuable insight into the use of IoT in education. For example, CISCO considers the following key factors for successful implementation of IoT in education: security, data integrity, and education policies [4]. Intel [11] advocates that IoT has the potential to trigger enablers to create the "synthesizing mind", which include: programming (commonly understood as "coding"), science, and making (in a sense of "makers movement").

There is, however, an independent study by the British Computer Society [12], which emphasizes the enormous significance of IoT in education for the future: *"The impact of the Internet of Things is likely to be revolutionary in all areas of education. This will be a consequence of speed of deployment, ubiquity, global scale, low cost and connectivity of billions of intelligent sensors and actuator devices generating unprecedentedly huge amounts of data. The interconnectivity and cutting across silos will place more demand on hybrid skills throughout ICT and beyond."*

Existing academic studies, although relatively few, confirm all such observations, for example, referring to new educational opportunities, which IoT will bring into "rural underprivileged areas" [13].

With respect of using IoT in higher education, the situation is similar, with the industry taking the lead. Most notably, in a special issue of Educause Review [14], executives from Salesforce, Google, Extreme Networks, IBM and CISCO present their views on the IoT impacts on higher education, followed by some sobering thoughts of one of the Information Technology directors at a major U.S. university: *"The IoT and IoT systems have the potential to provide substantial value to higher education institutions. But the implementation of those systems creates seams with our existing IT and information management ecosystems."*

The academic research falls far behind the industry and there are only a handful of studies analyzing impacts of IoT on higher education in the forthcoming years. In one, rather superficial paper [15], the author lists a number of changes the educators and administrators will face due to the introduction of IoT, including: changes in teaching and learning, experimental and practical changes, need for a change in management, etc. Brief discussion of using IoT in engineering education is presented, as well.

Another article [16] focuses on presenting the needs for adopting IoT technology on campus in e-learning, calling it smart i-campus. It points to a number of issues facing those who implement the i-campus, related mostly to the use of new technologies, but omitting completely the changes in pedagogy resulting from adopting the new approach.

One other paper [17] presents academic experiences on learning the IoT technology for purposes of e-business courses. The described model relies on using cheap, general-purpose boards based on Raspberry Pi and Arduino microcontrollers. The authors outline the course structure and its pilot implementation, sharing their first experiences and feedback received from students.

3.2. Status of IoT in Engineering Education

Papers specific to IoT in engineering education are very interesting, because they focus on addressing the practical aspects. Five papers are briefly reviewed here, in chronological order of their publication.

The oldest and the most substantive paper, by Kortuem et al. [18], discusses at full length the introduction of an Internet of Things based course in the introductory computer science curriculum, at Open University in the UK. The main objective of developing this course was to diverge from a traditional way of thinking about computing education, which starts with giving students good theoretical background and postpones introduction and acquisition of programming and artefact building skills until later years. Instead, the Open University project proposed "a radical departure from a traditional computer science curriculum", and implemented an infrastructure developed locally and composed of three essential elements:

- a unique networked sensor device (the SenseBoard)
- newly developed visual programming language, named Sense, similar to Scratch, and
- cloud services hosted on university servers.

This endeavor required, of course, "a multi-year effort by a large group of dedicated educators and a significant investment in people and technology" [18], which the Open University embarked on. As a result, students "learn with IoT technology, rather than merely learning about the IoT".

Other projects whose descriptions were found in the literature are on a much smaller scale and tend to use publicly available resources.

The impact of the IoT on engineering education is discussed in [19], where the authors advocate the use IoT to improve the quality, scalability and breadth of education. They present a teaching platform where students can use cloud services ranging from accommodating simple devices based on Raspberry Pi to more complicated broker services developed for remote sensor network. Their experiences are also discussed, with a main statement that "the major impact of IoT based learning environment is that the traditional teacher and student roles change significantly".

In a more recent paper [20], Uskov et al. present an ongoing project aimed at identifying main IoT features that can be used in support of a Smart Learning concept leading to a smart engineering education. The main result of the project, thus far, is an identification of several factors in the IoT, which contribute to Smart Learning in engineering education. Even though it has been one of the project's initial goals, the paper stops short of identifying "types of pedagogy that may be effectively supported by IoT".

Two other recent papers, [21]-[22], focus on device aspects of using the IoT in acquiring knowledge. The former stresses the software application aspects of IoT by introducing ThingWorx Android phone app and using it with a pre-built Android web UI. The major objective of the exercise is to expose undergraduate students to IoT in their early years and improve recruitment and retention. The latter paper discusses a successful approach in gradually introducing complexity to IoT projects with Raspberry Pi, to students who have not had any hardware or Linux based experience before.

4. SENIOR PROJECTS FOR IoT

The Software Engineering program at the authors' institution aims at creating a full IoT specialization. For the time being, prospective courses for the specialization have been defined and are in the process of approval. The projects described here are a part of this process aiming at the selection of the most convenient and the most effective teaching tools. Each project has a small embedded device, sensor or/and actuator, and targets a specific cloud platform. The choice of both the device and the platform has been given to students with instructor's approval.

4.1. Smart Home with Remote Device Access from iOS

The first project focused on studying remote device access in a smart home. The objective was to have a small microcontroller installed at home, with enough processing power and connectivity to have the capability of allowing a user to operate it remotely for some simple operations, from an iPhone running iOS 9.

The microcontroller selected for the project is an ESP8266 (Figure 6 [23]). The ESP8266 is a newer chip that is WiFi enabled. It is fully programmable with two GPIO ports. Because of its small size, 21.1mm by 13.2mm, and price of about $5, it is a perfect fit for a project like this.

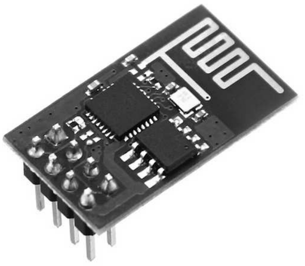

Fig. 6. ESP8266 microcontroller board

This is to control a lamp or outlet, which is powered by 120V power source, via a relay shield. The relay chosen is the SunFounder relay shield. It is able to successfully switch the 120V AC load on and off. The iPhone and microcontroller shall communicate over a UDP socket connection, through a Message Queue Telemetry Transport (MQTT) protocol service, as illustrated in Figure 7.

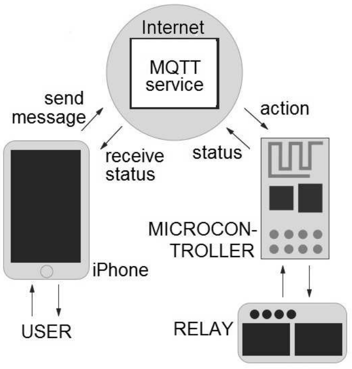

Fig. 7. Connectivity of the iPhone with a home appliance

The task was to design, implement and test software for both the microcontroller and the iPhone, with the use of MQTT service subscription.

4.2. GPS Tracker Using the FONA 808 Breakout Board

This project's objective was to track multiple FONA 808 devices (Figure 8) from a remote user terminal using the cellular GSM connection and GPS antenna via the Internet. The Arduino microcontroller is used to communicate with the FONA 808 to transmit GPS coordinates and use a database that stores information for multiple FONA devices to a user interface in order for the user to interact with the devices remotely.

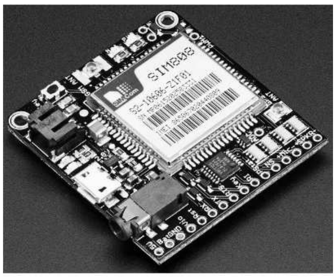

Fig. 8. FONA Board [24]

The overall connectivity diagram, involving a local user/client at a terminal, remote FONA board with Arduino, and Sparkfun data stream service is shown in Figure 9. The software development task involved designing, implementing and testing the following components: GUI and database software for the client side, Arduino GPS command and communication software for the FONA board, and interaction with the data stream service for data collection.

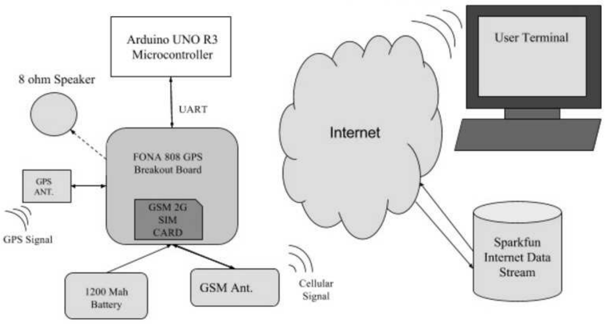

Fig. 9. Connectivity of the remote user with FONA board.

One of several difficulties that arose during this project was the reliability of the Sparkfun data stream service. It seemed to experience frequent down times and was not available for connection on multiple occasions. This affected both data collection from the FONA to the data stream as well as transmissions from the data stream to the GUI.

4.3. Remote Vision for Telecontrol of a Robotic Arm

The AL5A Robotic arm with the SSC-32 microcontroller (Figure 10) incorporates four degrees of freedom featuring base rotation, single plane shoulder, elbow, and wrist motion. In addition, the arm provides a functional gripper as well as a wrist rotate.

The project involved connecting the robotic arm to the Internet in order to relay commands remotely. It was previously implemented using a NearBus server, which is a cloud connector that receives client commands wirelessly, and most recently with Amazon Web Services, which allow more efficient operations. Services such as Simple Queue Service (SQS) and AWS Lambda support wireless connection to the cloud as well as the ability to execute software based on signal calls. These commands were issued via a client user interface, which permitted users to manually move the robotic arm whichever way they wanted to, given movement restrictions.

Fig. 10. AL5A robotic arm with SSC-32 controller board [25]

Current implementation incorporated two extensions. First, the Arduino Yun microcontroller was added to control the arm via its SSC-32 microcontroller, allowing for increased processing power as well as memory storage and connectivity through an Ethernet port. Secondly, and most importantly, a webcam was added on the base of the robotic arm in order to detect objects in view. OpenCV is a computer vision and machine learning software library [26] that provides this capability along with other features such as 3D model extractions, and object tracking. Information such as shapes, color, and distances can be analyzed. The robotic arm can then use this information to retrieve the foreign object in view via x, and y coordinates. Both these changes are illustrated on the connectivity diagram in Figure 11.

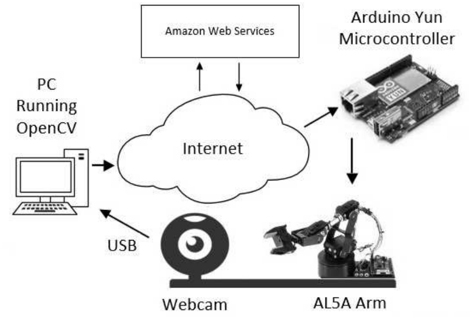

Fig. 11. Connectivity of the camera with AL5A robotic arm.

The diagram shows a webcam connected to a computer via a USB connection. This provides a continuous video feed that the OpenCV software analyzes in real time. The project objective is to analyze objects that come into view and successfully retrieve them with no user interaction.

After detecting an object, the computer sends commands to an Amazon AWS server, which holds the machine instructions for moving the robotic arm that can then be sent to the Arduino Yun for execution. The Arduino Yun connects to the robotic arm via the SSC-32 microcontroller that controls all of the on-board motors.

This way all user interactions with the robot are avoided and fully automatic control of the robot is implemented. The robotic arm is able to operate on its own using the webcam as a reference of where the object is that needs to be removed. Commands sent to the cloud (Amazon Web Services) are relayed to the Arduino Yun to perform the needed operations. This project could be used as a reference for manufacturing plants that would like to automate the assembly line.

4.4. Online Health Monitoring with a Smartwatch

The last project involved using a Moto 360 smartwatch [27] to implement a person's monitoring health parameters for use by a doctor and a person themselves. Initially, only one parameter, heart rate, was measured, but a completely operational system was implemented, with full connectivity to a Google Cloud, as shown in Figure 12.

Fig. 12. Connectivity of the smartwatch with Android and cloud.

As much as all previous projects did, this particular one also focused on the usability, which is illustrated in Figure 13. By reading the user's heart rate and sending the data via smartphone to the health monitoring server, which stores the history of all the information it receives, the application allows a doctor and a patient to stay in touch over a secure network, regarding all actions necessary to monitor health.

Fig. 13. Practical usefulness of the smartwatch solution.

5. CONCLUSION

Four student projects were described, developed on an experimental basis in a senior software engineering course, to illustrate the principles and advantages of the Internet of Things. Their emphasis was, first, on compatibility with the basic architecture of the IoT, and, second, on practical aspects and usability of the technology. The applications included smart home, real-time GPS tracking, automatic telecontrol, and online health monitoring. All projects were successful in bringing to operation the low-end "device things", whether sensors or actuators, and making connectivity with the services provided by the cloud, thus, demonstrating the value of the technology.

Future work will involve incorporating the ideas and experiences of this exercise into other courses across the software engineering curriculum, to form the degree specialization in the Internet of Things.

6. ACKNOWLEDGEMENTS

This work has been supported by a grant from NASA through University of Central Florida's NASA-Florida Space Grant Consortium (UCF-FSGC 66016015). Partial support was provided by a WIDER grant from the National Science Foundation, Award No. DUE-1347640. Views and findings expressed herein are those of the authors and not of the funding agencies.

The following students of the Software Engineering program at Florida Gulf Coast University are gratefully acknowledged for their contributions to the design and implementation of projects described in this paper: Wes Wickwire (iOS), Gio Gastaldo and Hilton Wise (FONA), Mike DeSeno and Derek Pike (AL5A), Steve Joy-Volk, Merzier Petit-Frere and Rudi Trevino (Smartwatch).

7. REFERENCES

1. Vestburg H., *CEO to Shareholders: 50 Billion Connections 2020*. Ericsson, April 13, 2010. URL: https://www.ericsson.com/thecompany/press/releases/2010/04/1403231
2. Evans D., *The Internet of Things How the Next Evolution of the Internet Is Changing Everything*. Cisco Internet Business Solutions Group. April 2011. URL; https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf
3. Nordrum A., The Internet of Fewer Things. *IEEE Spectrum Magazine*, September 23, 2016. URL: http://spectrum.ieee.org/telecom/internet/the-internet-of-fewer-things
4. Selinger M., Sepulveda A., Buchan J., *Education and the Internet of Everything*. Cisco Systems, San Jose, CA, 2013
5. IEEE Standards Association. *P2413 – Standard for an Architectural Framework for the Internet of Things (IoT)*. Institute of Electrical and Electronics Engineers, New York, 2016
6. IEEE Standards Association. *IEEE Standards Association (IEEE-SA) Internet of Things (IoT) Ecosystem Study*. Institute of Electrical and Electronics Engineers, New York, 2014.
7. *The Intel IoT Platform: Architecture Specification*. White Paper. Intel Corporation, 2016. URL: http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/iot-platform-reference-architecture-paper.pdf
8. Kraemer B., *P2413 - Standard for an Architectural Framework for the Internet of Things*. Institute of Electrical and Electronics Engineers, New York, 2016.
9. Amazon Web Services. URL: https://aws.amazon.com/
10. Microsoft Azure. URL: https://azure.microsoft.com/en-us/
11. *Transforming Education with the Internet of Things*, Intel Corp., Santa Clara, Calif., 2015.
12. British Computer Society, *Internet of Things Working Group: Report to PPAB*, BCS Policy and Public Affairs Board, June 12, 2013.
13. Pruet P. et al., Exploring the Internet of Educational Things (IoET) in Rural Underprivileged Areas, *Proc. ECTI-CON2015, 12th Int'l Conference on Electrical Engineering/Electronics, Computer, Telecom. & Info. Technology*, Hua Hin, Thailand, June 24-27, 2015.
14. Asseo I. et al., The Internet of Things: Riding the Wave in Higher Education, *Educause Review*, pp. 11-31, July/August 2016.
15. Zhang T., The Internet of Things Promoting Higher Education Revolution, *Proc. MINES 2012, 4th Int'l Conference on Multimedia Information Networking and Security*, Nanjing, China, November 2-4, 2012.
16. Veeramanickam M.R.M., Mohanapriya M., IoT Enabled Future Smart Campus with Effective e-Learning: iCampus, *GSTF Journal of Engineering Technology*, Vol. 3, No. 4, pp. 81-87, April 2016.
17. Bogdanovic Z. et al., A Platform for Learning Internet of Things, *Proc. Int'l Conference on e-Learning*, Lisbon, Portugal, July 15-19, 2014, pp. 259-266.
18. Kortuem G. et al., Educating the Internet-of-Things Generation, *IEEE Computer*, Vol. 46, No. 2, pp. 53-61, 2013.
19. Vujovic V., Maksimovic M., The Impact of the Internet of Things on Engineering Education, *Proc. ICOFE 2015, 2nd Int'l Conference on Open and Flexible Education*, Hong Kong, July 16-17, 2015, pp. 135-144.
20. Uskov V. et al, Smart Engineering Education: The Ontology of Inrnet-of-Things Applications, *Proc. EDUCON-2016, 7th IEEE Global Engineering Education Conference*, Abu-Dabi, UAE, April 10-13, 2016, pp. 476-481.
21. Abraham S., Using Internet of Things (IoT) as a Platform to Enhance Interest in Electrical and Computer Engineering, *Proc. ASEE-2016, 123rd Ann. Conference and Exposition*, New Orleans, La., June 26-29, 2016.
22. Zhing X., Liang Y., Raspberry Pi: An Effective Vehicle in Teaching the Internet of Things in Computer Science and Engineering, *Electronics (Switzerland)*, Vol. 5, No. 3, Article 5, 2016.
23. Seeed Studio, WiFi Serial Transmission Module with ESP8266. URL: https://www.seeedstudio.com/WiFi-Serial-Transceiver-Module-w-ESP8266-p-1994.html
24. Adafruit, FONA 808 – Mini Cellular GSM. URL: https://www.adafruit.com/products/2542
25. Lynxmotion, AL5D Robotic Arm with SSC-32. URL: http://www.lynxmotion.com/p-1035-lynxmotion-al5d-4-degrees-of-freedom-robotic-arm-combo-kit-no-software-w-ssc-32u.aspx
26. OpenCV, Multiple Object Detection with Color. URL: https://i.ytimg.com/vi/hQ-bpfdWQh8/hqdefault.jpg
27. Motorola 360 Smartwatch. Product Video. URL: https://www.bhphotovideo.com/c/product/1196765-REG/motorola_00865nartl_moto_360_sport_smartwatch.html

LABORATORIA DLA „INTERNETU RZECZY” W PROGRAMIE INŻYNIERII OPROGRAMOWANIA

W artykule omówiono zagadnienia laboratoriów studenckich dla kursów związanych z „Internetem rzeczy”, w programie studiów Inżynierii Oprogramowania. Przedstawiono koncepcję Internetu rzeczy i omówiono korzyści płynące z użycia tej technologii w kształceniu, szczególnie w dyscyplinach inżynierskich, dyskutując podstawowe problemy z tym związane. W szczególności, opisano praktyczne rozwiązania problemów laboratoryjnych, z użyciem robota, systemu GPS, inteligentnego zegarka i zdalnego przekaźnika, oraz przekazywania odpowiednich danych do przetwarzanuia w chmurze.

Słowa kluczowe: Internet rzeczy, przetwarzanie w chmurze, kształcenie inżynierów, inżynieria oprogramowania, zdalne laboratoria.