Learning IoT skills in the context of student projects

Marek Beránek, Ilja Lisunov, Václav Vacek

Learning IoT skills in the context of student projects

Marek Beránek, Ilja Lisunov, Václav Vacek
Department of Information Technologies
Unicorn College
Prague, Czech Republic
info@unicorncollege.cz

Abstract - **Extensive use of various types of IoT devices has resulted in high demand for developers with IoT skills. We have identified the challenges facing IoT developers and designed suitable student projects that aim to address these challenges. It is our experience, that it is important to match the IoT platform used for the project to the skills of the students to achieve optimal learning outcomes. In this paper, we describe two case studies that use high-quality modular IoT kits to accelerate the learning process and to allow students with relatively low technical skills to implement working solutions to *real-world* problems.**

Keywords— IoT; Arduino; BigClown; Raspberry Pi

I. INTRODUCTION

The Internet of Things (IoT) refers to billions of physical devices around the world that are connected to the Internet, collecting and sharing data. Almost any physical object can be transformed into an IoT device if it can be connected to and controlled over the Internet that way. According to some experts [1], IoT can be regarded as the next step in the evolution of the Internet (Figure 1). Thousands of publications dealing with IoT topics are available and many companies see great business potential in IoT devices. Furthermore, IoT plays an important role in the emerging area known as Industry 4.0.

Fig. 1.   IoT as the next evolutionary step

One could easily get an impression that we are facing an entirely new trend, especially if you consider various predictions and a range of possibly affected areas of applications [2]. But, it is not completely true.

We have been surrounded by smart devices for decades; consider the following examples. We have been driving cars that are packed with sensors used for diagnostics during service and repairs. We are living with smart home appliances where signals from sensors are handled by embedded or PLC (Programmable Logic Control) devices. Automated manufacturing processes and robots have been used extensively using hardwired connectivity or closed local networks.

So, what has changed? Prices of components dropped dramatically, Internet connectivity became available at the high speed, and computing power of mobile devices is reaching a level of PC's only a few years ago. Young people entering their professional life have grown up in the era of massive development of various types of web services, and they spend significant part of the life in on-line. This ubiquity of the Internet creates a situation that makes it relatively easy to connect devices that represent physical objects and to control these devices remotely over the Internet. The preferred solutions for the "On-line" generation will use wireless connection available anytime and anywhere [3].

However, achieving this will not be so fast and easy. Similarly, to the situation some 24 year ago, when WWW and the HTML language emerged from the world of physics at CERN, it took nearly two decades to establish the web as an integral part of everyday life. It is highly probable that the implementation of IoT and Industry 4.0 projects will be more rapid than the development of the web, but we can already identify some factors that are slowing the adopting of IoT. A recent Cisco survey [4] found that nearly three quarters of all IoT projects are unsuccessful and suffer from the following problems caused by the lack of IoT expertise:

- IoT projects take a long time to complete.

- The quality of data collected is poor, or the data is not used effectively.

- IoT integration issues.

- The IoT projects exceed budgets.

Additionally, there is a need to improve communication between production engineers and IT specialists, and much more attention should be given to IoT security issues that are often neglected by enthusiasts entering the business. Many experts predict that there will be large demand for the IT specialists with at least some basic knowledge of engineering physics and some Big Data analyses experience to be successful in future IoT projects [5].

In this paper, we share our experiences gained conducting IoT projects at the Unicorn College with students being trained as IT specialists. Our IoT projects are classified into the following categories:

- *Hobby platforms* [6, 7] (Arduino, Raspberry, etc.) – inexpensive training and learning applications using low accuracy sensors and consequently low precision data that are not suitable for real-world applications.

- *Semi-professional platforms* – slightly more expensive components from suppliers that pay more attention to quality and precision of the hardware. We have very good experience with Big Clown [8, 10] products and solution by Hardwario [11]. Besides higher quality, the advantage is that components can be assembled like Lego parts without the need of wiring and soldering. This plays an essential role if only IT people are involved in the IoT projects.

- *Professional IoT platforms* – this category relies on traditional solutions based on PLC platforms, where one can take advantage of the long-term reliability of widely used and proven components in control and automation systems. The most suppliers adhere to international standards, and have been recently providing connectivity either to their own clouds or are enabling communication for commercially available cloud platforms.

In the next section, we discuss the steps involved in implementing successful IoT projects and the choice of a suitable IoT platform. In section III we present two examples of IoT projects, and section IV contains our conclusions.

II. BUILDING IoT APPLICATIONS

A. *What tasks need to be solved for a successful IoT project?*

Figure 2 illustrates the steps that a successful IoT project should follow.

Fig. 2. Basic steps leading to the well-defined IoT project

The choice of a Software Development Kit (SDK) is very much dependent on what developers need in order for them to successfully build software for a particular platform. The SDKs typically include APIs, Library Files, Development Tools with Utilities and Sample Applications. Scalability is essential for large-scale projects. Developers often initially ignore security issues and attempt to add security features in the later stages of the project. The use of a cloud platform is becoming necessary for all mobile solutions in order to provide relevant storage, and to ensure accessibility from anywhere at any time. Cloud platform also provide the necessary compute power to handle computationally intensive tasks necessary for Big Data analyses.

B. *How to choose a suitable plaform for IoT projects?*

There are many platforms to choose from today but it is recommended to begin with some of the most popular like Arduino (well suited for various measurements) or Raspberry Pi (with good compute power, running Linux OS but requiring permanent power supply because of high module power consumption) simply because of their availability and low price. The product information typically states that the devices are easy to assemble and program, but this may not be the case for IT students who lack experience and knowledge of electronics. Frequently, such attempts end in *spaghetti* wiring and poor functionality (Figure 3).
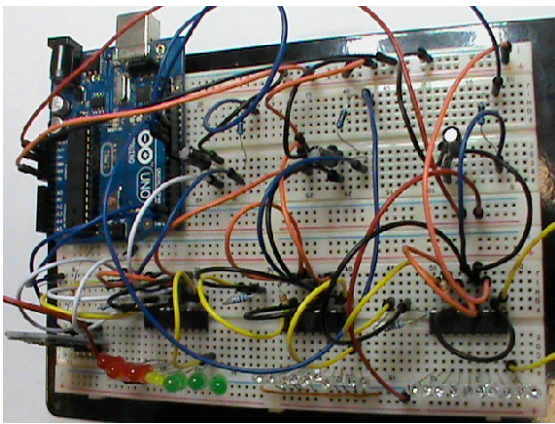
Fig. 3. *Spaghetti* wiring - typical result of student IoT projects

We have carefully evaluated two platforms at our college – Arduino based solutions and the Big Clown modular solution. Big Clown components generate quality data, and are easy and fast to develop solutions from the initial specification to the final implementation. We have compared the platforms on the basis of: hardware comparison, software, network options, design complexity, quality of documentation, coding complexity, price, extensibility, power consumption, sensor quality, sensor measurement time, and architecture implementation. Apart from the price, the BigClown platform is superior because of its simplicity of use, out-of-box readiness and shorter time needed to develop a functional and elegant solution. One of the most interesting features of the BigClown platform is in-build radio module. The radio module is very power efficient, allowing users to build isolated standalone solutions that can work for months (sometimes years) powered by four AAA batteries. The RF-module can work with distances of up to 500 meters. The BigClown platform is based on ARM Cortex M0+ 32-bit MCU and offers 192kB of flash and 20 kB of RAM memory. It also has various inbuilt modules such as temperature tags, radio, security chip, accelerometer, two buttons, LED and 18 GPIO pins. The BigClown platform supports multiple communication protocols: 3x UART, 2x I²C, 1x SPI, 5x ADC, and 2x DAC. It is important to note, that the ARM M0 CPU supports deep sleep mode, enabling the hardware to operate for a year or longer on a single battery charge. The BigClown component set also comes with SigFox license and it is prepared for use with LoRa. Both Sigfox [12] and LoRa [13] are specialized Low Power Wide Area Networks for data transfer.

Unlicensed networks, mostly based on LoRa and Sigfox, make up two-thirds of low-power wide-area (LPWAN) networks today. The Sigfox network covers 93% of the population of the Czech Republic at the moment.

C. How to solve IoT wireless connectivity?

Figure 4 illustrates one possible wireless IoT solution. Sensors are placed on a Sensor Node with a RF module transferring data to a similarly equipped IoT gateway that can communicate with a local database and can pass data to the cloud at the same time.
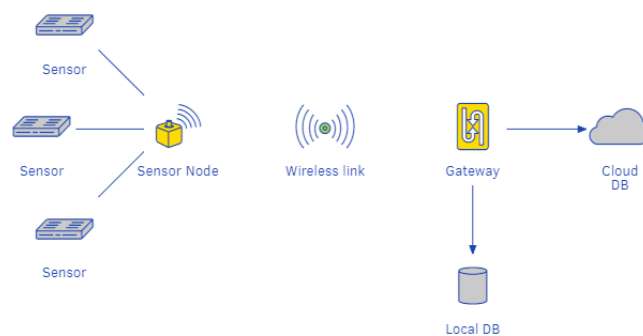

Fig. 4. Example of IoT wireless solution

One should take in account that nearly all wireless based IoT projects have to carefully consider power requirements to operate such systems for long time durations without the need for maintenance. Figure 5 illustrates a solution using the BigClown Kit.
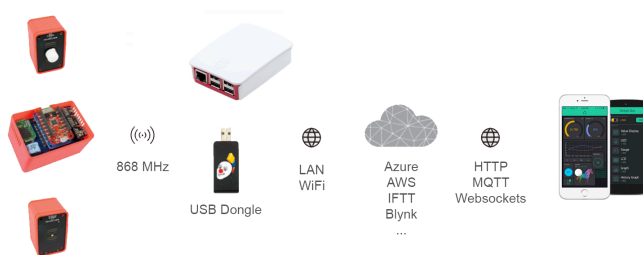

Fig. 5. Wireless IoT solution with BigClown Kit components

The IoT sensors shown in the very left column send data to the USB dongle via Sub-GHz radio using 868 MHz frequencies. The USB dongle can be used with a computer (e.g. Raspberry Pi) or smart router. Data are transmitted from the USB dongle to the computer through the serial line. Combination of the USB dongle and Raspberry Pi typically

represents gateway or field gateway that communicates with a cloud service (e.g. Azure, AWS, IFTT, Blynk, etc.). The gateway uses LAN or Wi-Fi for Internet connection. Different applications providing data visualization or enabling sensor settings, communicate directly with a cloud service often using HTTP, MQTT or Websockets.

III. EXAMPLES OF IoT PROJECTS

A. Server room monitoring system (use case 1)

The prototype was designed to monitor at least two key temperatures, humidity and a flood sensor in order to detect malfunctioning of the cooling system in a computer room with servers running 24/7. The hardware prototype assembled with basic hardware components contains a Base module, a Power module, and a Core module with attached temperature tags, a humidity tag and a flood sensor connected via the Sensor module to Core module. The BigClown prototype was connected to a Raspberry PI that simulates a small communication server and acts as a microcomputer to process and store data. Influx (DB) and Grafana (Dashboard) with its dependencies were installed to retrieve and display data from the BigClown platform. Additionally, we have installed the BigClown USB dongle to act as a MQTT gateway for radio devices on the Raspberry PI. This allows the implementation of a standalone wireless solution. Figure 6 shows a fully assembled wireless device with battery module, temperature tags, humidity tag and flood detector (right of the figure), and the wireless gateway module connected to Raspberry Pi server with an additional USB dongle for RF communication (left of the figure).
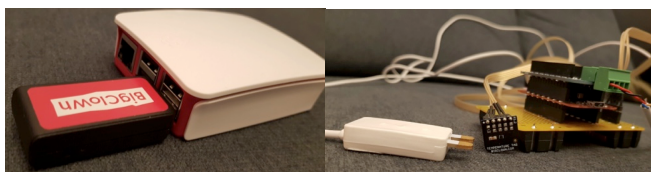

Fig. 6. Server monitoring system

Since the BigClown platform works together with a Raspberry PI that uses the Debian OS (Linux), we have access to multiple software solutions to save and track the data. Influx Database has been chosen for storing data, as a real-time database that is designed for monitoring and controlling sensors and devices and for storing data

metrics and events, which suits our application goals.

Fig. 7. Main Dashboard with graphical charts and data history for each of the three sensors

The Grafana web server [14] is used as a front-end application to view data retrieved from the database in real-time, and to build easy to understand charts (Figure 7).

B. Seat occupancy monitoring in meeting rooms (use case 2)

This project is aimed at monitoring meeting room occupancy and evaluation of room usage as compared to reservations. Additional technical requirements were specified as follows:

- The device should use RF-module for communication. Even though Bluetooth and Wi-Fi are very popular ways to connect devices, we have avoided those protocols for the sake of reducing power consumption and improving connectivity distance.

- Sensors will be heavily used (chairs are used daily for about 3-5 hours), so that avoiding mechanical and direct contact sensor components will reduce the risk of mechanical failure or damage of components.

- Sensor unit should monitor each chair or seat to increase accuracy of the collected data.

Many different sensors were considered and evaluated, including temperature sensors, strain gauge sensors, IR sensors, ultrasonic sensors, PIR sensors, lux sensors, piezoelectric sensors, capacitive sensors, etc. The BigClown platform was selected due to its flexibility and also because of the availability of a temperature module, PIR module and a lux meter module in the kit; most of the sensors provided traceable signals. The Figure 8 shows an example of PIR sensor data within a short sitting session, illustrating that even small

movements of a person sitting on the chair are detected.

Fig. 8. Chart of PIR signal values that display one sitting session

The PIR sensor solution seemed to be optimal when compared to other sensors we evaluated. Unfortunately, the overall cost of this solution was not acceptable when using a separate sensor for each seating place (chair). The estimated price for several meeting rooms with an average number up to 10 chairs per room became too high.

We have decided to implement an inexpensive solution using a custom made capacitive sensor modules. Capacitance sensing is a technology based on capacitive coupling that can detect and measure anything that is conductive or has a dielectric difference from air. This technology is commonly used in touch screens and track pads.

1) Capacitive sensor module

Capacitive sensor electrical signal goes through high-value resistor; the output of the resistor is connected to input pin of the MCU (Microcontroller unit). There is a "sensing area" connected to the output pin of the resistor. This area influences the time required to change input pin state of the MCU. Interestingly, the area doesn't require direct contact with source of the capacitance (i.e. human body) [9]. Step by step assembly procedure is outlined in the Figures 9 and 10.
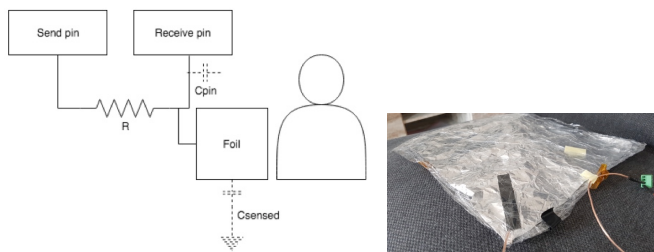

Fig. 9. Circuit diagram of CapSense module and Sensing area made of aluminium foil wrapped into plastic film
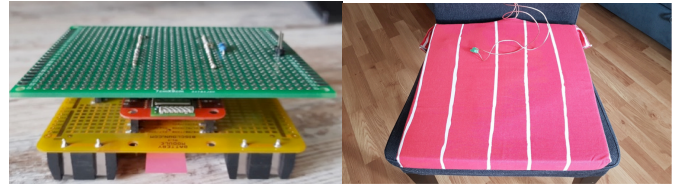
2) Practical implementation

Fig. 10. CapSense module implemented using the BigClown platform and Seat cover with an integrated sensing area

A custom made capacitive sensing module was placed on top of BigClown modules in similar manner to other BigClown native modules. The seat cover contains embedded sensing area. It is important for monitoring to keep the input pin discharged after the sensing is done to avoid the preceding values influencing later measurements, i.e. avoiding a situation where the sensor is continuously sending high signal values indicating seat occupancy. Sensing algorithm has to follow alternating switching of input and output pin values (LOW and HIGH). The algorithm includes delays needed to fully discharge the input pin and measuring time needed for signal to change the state on receive pin to HIGH.

The Figure 11 displays the events of the seat state changing to occupied and then going back to vacant.
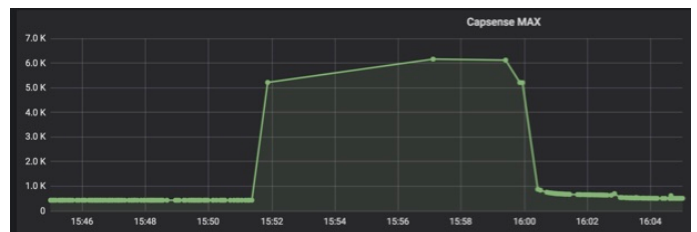

Fig. 11. Chart of CapSense module values that display one sitting session

All values are displayed in arbitrary units that indicate time needed to receive pin on MCU to change its state. As shown in Figure 11, around 15:51 the sensor signals rises from ~600 arbitrary units to over 5000 arbitrary units. It clearly indicates the moment when person sat down on the seat. At around 16:00 the person got up, resulting in a sharp signal fall shown on the chart, with signal returning to the baseline.

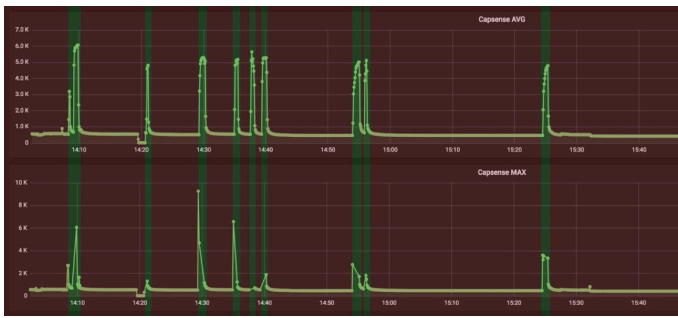A longer period of seat occupancy monitoring is shown in Figure 12.

Fig. 12. Colorized areas with occupied (green) and vacant (brown) seat state for CapSense module

We can clearly identify the time when a person sits down on the chair (value rises) and when a person leaves the chair (value drops back to the baseline). On top of that we have reduced sensor price to a negligible amount and have improved strength and reliability. The installation within the seat cover allows testing of the solution with up to 32 channels prior to the eventual incorporation of the setup into chairs.

IV. Summary and Conclusions

Proliferation of various types of IoT devices has resulted in a demand for developers with IoT skills that are able to design and implement IoT solutions using various types of IoT platforms. In response to this demand, we have introduced IoT topics into our courses at Unicorn College, and provide opportunities for learning IoT design and development techniques in the context of student projects. We have identified the challenges facing IoT developers and designed suitable student projects that aim to address these challenges. We have classified IoT development platforms into three categories: (a) Hobby platforms, (b) Semi-professional platforms and (c) Professional IoT project platforms.

It is our experience that it is important to match the IoT platform used for the project to the skills of the students to achieve optimal learning outcomes. As illustrated by the two case studies presented in this paper, the use of high-quality modular IoT kits can accelerate the learning process and allow students with relatively low technical skills to implement working solutions to *real-world*, practical problems.

References

[1] Coetzee, Louis, and Johan Eksteen. "The Internet of Things-promise for the future? An introduction." *IST-Africa Conference Proceedings, 2011.* IEEE, (2011).

[2] Press, Gil. "Internet of Things (IoT) Predictions from Forrester, Machina Research, WEF, Gartner, IDC." *Forbes Magazine,* (2016).

[3] Beranek, M.; Bory, P.; Kovar, V.; et al.: Methods used for Effectivity Improvement of Learning at Unicorn College, INTERNATIONAL JOURNAL OF EDUCATION AND INFORMATION TECHNOLOGIES, Volume: 10 Pages: 168-175 Published: 2016

[4] Cisco Survey:https://www.slideshare.net/CiscoBusinessInsights/journey-to-iot-value-76163389, May 20, (2017)

[5] Lund, Denise, et al. "Worldwide and regional internet of things (iot) 2014–2020 forecast: A virtuous circle of proven value and demand." *International Data Corporation (IDC), Tech. Rep* 1 (2014).

[6] Arduino Uno official website. https://store.arduino.cc/usa/arduino-uno-rev3

[7] Raspberry Pi documentation for the hardware. https://www.raspberrypi.org/documentation/hardware/raspberrypi/README.md.

[8] BigClown documentation: basics. https://www.bigclown.com/doc/basics/getting-started/#feature-2-bigclown-is-wireless

[9] Malý, M.: Hradla, Volty, JednočIpy (Úvod do bastlení), 1. vydání, Edice CZ.NIC, Praha 2017 pp. 514, ISBN 978-80-88168-26-3

[10] BigClown official Github repository. https://github.com/bigclownlabs/bc-hardware

[11] Novák, Jakub. *Průzkum trhu pro aplikace internetu věcí.* BS thesis. České vysoké učení technické v Praze. Vypočetní a informační centrum., (2018).

[12] Vejlgaard, Benny, et al. "Coverage and capacity analysis of sigfox, lora, gprs, and nb-iot." *Proceedings of the 2017 IEEE 85th Vehicular Technology Conference (VTC Spring), Sydney, Australia.* (2017)

[13] Bor, Martin, John Edward Vidler, and Utz Roedig. "LoRa for the Internet of Things." (2016): 361-366.

[14] Tortone, Gennaro, et al. "Web Based Visualization Tools for Epics Embedded Systems: An Application to Belle2." (2018): THPHA181.