Theme Article: Special Section on Consumer Technologies for Smart Agriculture

An IoT Based Smart Irrigation System

Cristina Stolojescu-Crisan
Politehnica University of Timisoara

Calin Crisan
SafeFleet Telematics

Bogdan-Petru Butunoi
West University of Timisoara

***Abstract*—Smart home applications are fast becoming a staple of modern houses, leading to an increased interest in consumer electronics which enable the automation of common household tasks. However, high initial costs, complex solutions, and closed systems mean home automation is still out of reach for a large number of potential customers. A cheap, open, extensible solution is therefore a must if IoT is to become a standard option for modern homes. In this article, we present an innovative solution to this problem in the form of an IoT framework called qToggle: a system for interconnecting sensors, actuators, and other data sources based on open source solutions and easily accessible components. To give a practical example of the uses of the new framework, we implemented a smart irrigation system, which can simplify garden care, while reducing water and energy consumption. The article describes details of the design, instrumentation, and software (mobile application) for the real time control of this system.**

The world's population is drawing close to 7.8 billion people. One third of this does not have access to sufficient drinking water with climate change and increased water demand being the main factors responsible. On the other hand, inefficient and primitive usage methods have led to 70% of the total water usage worldwide being generated by irrigation.[1] Increasingly, this problem is being tackled by the use of automated, smart irrigation systems, powered by the Internet of Things (IoT), which have become a key component in precision gardening and agriculture. The IoT has given a boost to smart agriculture, in terms of productivity and resource optimization. It has increased efficiency and has minimized the cost of production.[2] Smart homes,

Digital Object Identifier 10.1109/MCE.2021.3084123
Date of publication 28 May 2021; date of current version 3 May 2022.

as part of the IoT paradigm, involve a networked system composed of various communication technologies and consumer electronics automating various tasks, including automated irrigation systems.[3] Each component of the smart home solution represents a key player in the overall "smart lifestyle" paradigm and needs to fulfill certain conditions for it to be optimal: ease of access to the public, open connectivity and ease of coupling with other components, online availability, and ease of usage. Automated irrigation systems help people control water usage in their gardens or fields and, thus, avoid water waste and minimize water bills. Moreover, smart irrigation systems aim to improve the quality of plant growth in gardens and fields by irrigating at correct times, with the appropriate amount of water, taking into consideration soil moisture levels and particular irrigation requirements in a given location. Replacing manual irrigation with an automatic valve system also eliminates the human error component. Smart irrigation systems also contribute by saving energy, time, and valuable resources. An IoT-based irrigation system is based on a smart microcontroller (e.g., an ESP chip). Soil moisture sensors are placed in the ground and send real-time data to the microcontroller, which automatically switches on the water pump when the values of the considered soil parameters are out of a specific reference interval. There is an entire discussion regarding the reference interval for these parameters (humidity, temperature, PH, etc.), because they are influenced by a series of factors: season, month, geography (continent, country, regions of a country: mountains, hill, riverside, etc.), type of soil, type of plants (grass, vegetables, cereals, etc.), and so on. In this article, we propose a simple but efficient smart irrigation system based on the Raspberry Pi. The irrigation system is part of an IoT framework called qToggle, a system for interconnecting sensors, actuators, and other data sources, with the aim of multiple automations, as shown in Figure 1.

Most smart irrigation controllers currently on the market are overly complex, with a wide range of capabilities, making them far too elaborate and expensive for small independent gardeners. On the other hand, smart irrigation companies usually only offer the irrigation controller and the app, we have proposed a complete irrigation solution, hardware, and software. Our goal is to provide a "do it yourself" irrigation system for gardens, along with three different scenarios: manual, automatic, and sensor-based automation.

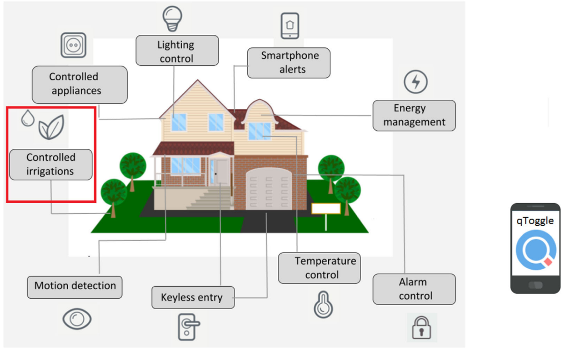

Figure 1. Smart irrigation system as part of the IoT based smart home controlled by qToggle.

STATE OF THE ART

Irrigation is one of the primary needs of agriculture. A lot of research has been done in this context and various irrigation systems, based on different technologies and using different devices, have been proposed in the literature. Some of them are cheaper than others, with different complexity or efficiency. Most of the proposed irrigation systems are dedicated to rural agriculture.[4–6] These systems are usually complex and too expensive for a small, independent garden, or are prohibitive due to the financial barrier.

Garcia *et al.*[1] presented a survey aimed at summarizing the current state of the art of the research regarding smart irrigation systems. They evaluate 128 papers published between 2014 and 2019 and present an overview of the most utilized technologies employed to implement IoT based smart irrigation systems. Analyzing the papers already published in the literature, some conclusions can be drawn. The best node for an IoT irrigation system is chosen considering the necessities and the characteristics a user wants for the system. Arduino boards are the most used nodes for the implementation of IoT irrigation systems. Koprda *et al.*[7] proposed the automated control and remote management of an irrigation system using an Arduino board and the Android operating system as a solution. The system is fully controlled by an online interface and requires an active connection to the Internet. Arduino boards are used by Karpagam *et al.*[8–11] as well. Arduino is highly flexible, open source, not expensive, and easy to program. However, Arduino is not designed to handle the large complexity of more complex projects due to scaling issues. Garcia *et al.*[1] claim that 59 papers used an Arduino board. The Raspberry family was frequently used as well since they are more potent than the Arduino boards, having powerful computing abilities that allow the implementation of more demanding software and algorithms, hence the choice of the Raspberry Pi board for the proposed system. The selection of the processor (controller) depends on the characteristics of the IoT irrigation system considering the type of crop and the irrigation needs. The microcontroller used for the proposed irrigation system is the ESP8266 chip. The communication technology represents a key point to achieve successful operation. In many papers in the literature, the authors combine several communication technologies, for example, the authors use either a wired or a wireless technology to connect the sensors with the nodes, and a wireless technology to send data from nodes to storage centers. The most used communication technology in most papers is WiFi, probably due to its accessibility. Most low-cost devices for IoT usually support WiFi and small farms are able to provide enough wireless coverage with several low-cost devices. GSM and ZigBee are widely used wireless technologies as well, for example, Goap *et al.*[10,12] GSM provides long-range communication at the cost of a mobile plan of the service provider that operates in the area. ZigBee provides low energy consumption, but it has lower data rates than other technologies and its range imply the deployment of many nodes. The communication used in this project is Wi-Fi or Ethernet. Most of the papers in the literature use sensors based on the conductivity between two electrodes, the most popular being the YL69 sensor, which was used in this article as well. Wireless sensor network (WSN) technology has also been used in irrigation projects by Astutiningtyas *et al.*[9,10,13–15] Most irrigation systems do not have access to the power grid or may only receive power during a given time period. The use of solar energy reduces the energetic costs, which is an advantage for irrigation systems intended for developing areas. Many surveyed papers use solar energy to power their proposed irrigation system. Starting this year, the irrigation system proposed will be using solar energy thanks to the photovoltaic panels installed.

The contributions of this article are both hardware and software. The hardware part of project, including the system inside well and the piping part described in section "QTOGGLE as an IoT Framework," has been designed and installed by the authors. In addition, the qToggle system, including the qToggle app described in section "Irrigation System Architecture," has been fully developed by our team. Another contribution of the proposed system resides in the fact that it lowers the barrier of entry to smart homes by reducing upfront costs and preventing vendor lock in. This allows a wider consumer pool to access smart home solutions both for residential and industrial needs.

QTOGGLE AS AN IOT FRAMEWORK

The proposed irrigation system represents a small part of the qToggle project, which is a complex automation project, with many functionalities that could transform a normal house into a smart home. The system can be monitored and controlled from any part of the world very easily, using a mobile app, entirely developed by our team. qToggle proposes a standard way of interconnecting devices. qToggle is built around a flexible, but powerful application programming interface (API), allowing devices to work together. The idea behind qToggle is to control programmable systems having a TCP/IP stack via simple HTTP requests. These systems can be, for example, single-board computers or TCP/IP-enabled microcontrollers. Features that make qToggle special are the following: a unitary and consistent solution that integrates all required features, device provisioning and management, firmware update over the same unique API used by all devices, the use of expressions allowing intelligent and complex rules to be implemented between various sensors and actuators inside a network, hierarchical master–slave topology offering flexibility and scalability, user data staying within the premises of the local network, a cloud connection not being needed (for security and privacy reasons), and the integrated web app that works well on all major platforms (both desktop and mobile). qToggle provides a friendly user interface, which comes in the form of a progressive web application (PWA). The advantages of a PWA include the fact that updates are immediate (not being served by a vendor platform such as the Play Store or App Store) and there is one single code base shared among all platforms. qToggle is designed to be used in both mobile and desktop environments. qToggle has been thought as a way of interconnecting sensors, actuators, and other data sources with the purpose of multiple home automations, control, monitoring, and security, a system that could be continuously developed and improved. One of the key features of the qToggle IoT platform is its ability to run without a cloud connection. More precisely, many users are concerned with their data leaving their premises. Therefore, we designed the web app that runs on the mobile phone in such a way that it can securely connect to the hub directly, using HTTPS, without the need of an intermediate server. Simplifying the system in this way has the advantages of reducing the number of parties involved (and thus the overall latency as well as operational costs), while increasing the security. Probably one of the biggest differences between qToggle and its competitors is that our target audience includes users without a technical background, who prefer intuitive, off-the-shelf solutions with minimal setup requirements and a smooth learning curve. qToggle also caters to users who require an IoT solution that is cheap and easy to install, does not lock one in to a specific vendor and scales very well, thus reducing the entry barrier for home and garden automation, one of the major impediments to the adoption of smart home solutions.

IRRIGATION SYSTEM ARCHITECTURE

The basic elements of the proposed irrigation system are: the well, the master hub connected to the Raspberry Pi, and the sprinklers network (including the links). When sprinkler heads are

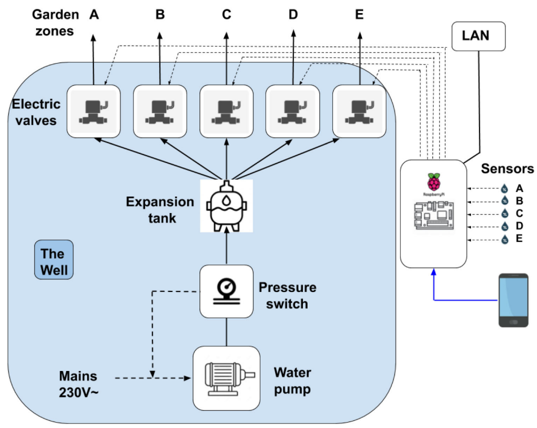

Figure 2. Diagram of the irrigation system and the system inside the well (the blue rectangle).

placed in the ground, they must follow a precise "architecture," called "head-to-head" spacing, for the best coverage. This means that each sprinkler must overlap part of the area watered by the adjacent sprinkler. Even though this overlap might seem like a water waste, it is a real necessity. Without this overlap, it would be impossible to obtain uniform water coverage. Water pressure is another important parameter when designing an irrigation system. If the water flow is limited, the irrigation system is divided into zones to ensure there is always enough water flow to operate one zone in the irrigation system. For the case study presented in this article, we have divided the space into five zones (A-E). Thus, the system inside the well (shown in Figure 2) contains: five electric valves (Rain Bird DV/DVF valves), each one corresponding to a zone, one water pump, a Raspberry Pi board connected to the house LAN, a pressure switch, and an expansion tank. Additionally, for the scenario involving moisture sensors, Figure 2 includes the five sensors A-E, corresponding to each of the five zones. Each humidity sensor is connected to an ESP8266 chip.

The Rain Bird DV/DVF valves (Diaphragm valves)[16] are premium products, representing the first choice of industry leaders worldwide, with more than 20 years of proven performance. The type of water pump is a standard 1.1 kW pump, with a maximum water debit of 70 l/min. In Figure 2, the water circuit is represented with solid lines and the electrical circuit with dashed lines. The pressure switch is an electronic device that starts and stops the pump motor according to the detected pressure. We used the following model: Easy Press II, with a maximum pressure of 10 bar and a flow of 12 000 l/hr. The expansion tank has a volume of 24 liters and a maximum pressure of 8 bars. The aim of the expansion tank is to reduce the water pump ON–OFF switching rate. This is accomplished by accumulating up to 8 bars of pressure inside it. By limiting the ON–OFF-switching rate, the cost is not only reduced, due to power consumption, but the life of the water pump is increased as well. The irrigation controller will ideally be controlled in turn by a master hub. While both are powered by Raspberry Pi boards, the controller is a terminal node in the network and is usually placed close to the valves. The master hub is typically installed in a central location as it serves as the nexus for managing all other interconnected devices. Even though there are various boards and platforms that could be used for this purpose,[17] our choice was the Raspberry Pi board. The reason for choosing Raspberry Pi boards, other than the low price, was the energy efficiency, the programming facility (Raspberry Pi uses Python, a programming language with relatively fewer lines of code and less complexity), small hardware footprint (RAM and CPU), the fact that it is future proof, the many available GPIOs, and its several roles in the IoT system, at the same time. Moreover, most of the software and projects done on Raspberry Pi are open source and are maintained by online user communities, always excited about new challenges. The humidity sensors chosen in this project, for the sensor-based scenario, are the popular YL69 from SparkFun and the microcontroller is the ESP8266 chip, a low-price Wi-Fi module perfectly suited for projects in the IoT field. The humidity sensors are connected to the irrigation controller and continuously send values for the relative humidity of the soil. These values are then grouped by zone and used to compute the min/max and average humidities for each zone. The automation consists in deciding how much of the total allocated watering time will be used for the respective zone. The proposed irrigation system can be monitored and controlled very easily, using a mobile app, the qToggle app. Using the qToggle app, users will be able to monitor and control the irrigation system from any part of the world. An important metric related to an irrigation system is the adequate quantity of water used for irrigation. There are various factors that influence the calculation of this amount: the geographical area, the season/weather, the type of soil, the type of crops, the type of sprinklers (each of them having their own range of precipitation rates and flow, depending on the choice of nozzle), water pressure, etc. Each type of nozzle has its own operating specifications, given by datasheets, for example, pressure (in bar), radius (in m) or flow rate (in l/hr). Here we employ nozzles with producer specifications as in Li *et al.*[5] Lawns usually need 15 to 25 mm of water per week, either from natural or artificial sources, usually a combination of both. To estimate potential water usage, we employ a simple but effective approximation: one liter of water will cover one square meter to a depth of one millimeter. Thus, to apply 15 mm of irrigation weekly, 15 liters of water are needed for each square meter. The necessary watering time can be computed as: $t = \frac{Q_n * A}{F}$, where $Q_n = 15$ liters needed for each square meter, A is the area in square meters, corresponding to each zone, F is the total flow (measured in cubic meters per hour) calculated as a sum of flows corresponding to all sprinklers in a zone. These calculations are no longer needed when using moisture sensors. The type of sensors used in this project are YL69,[18] a simple, low cost sensor used for measuring moisture in different types of materials. The output voltage for this sensor ranges between 3.3 and 5 V and the output values are related to soil moisture: 0 to 300 for a dry soil and 300 to 700 for a humid soil, respectively. Each sensor is connected to the analog digital converter (ADC) input of a microcontroller (ESP8266 chip in this case). The ADC converts the input voltage into digital data that can be processed by the qToggle server running on top of the Raspberry Pi. In the end, the qToggle server is provided with five input values representing the soil humidity percentages for each of the five zones.

A SPECIFIC CASE STUDY

A suite of irrigation systems is described in the literature or already available on the market. The combination of technology and setup difficulty associated with each on them varies, as do

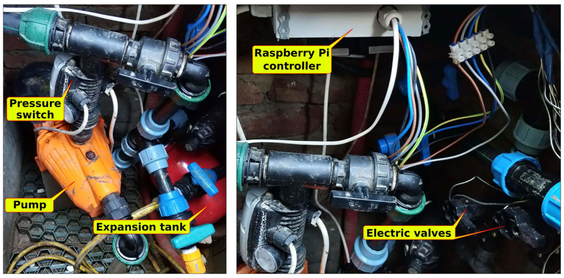

Figure 3. Real picture of the system inside the well.

their pros and cons regarding complexity, costs, performances, and so on. Our goal in this article is to present a "do it yourself" system that is simple, cheap, and performant. Figure 3 shows real pictures for the irrigation system inside the well.

In what follows, we consider three potential irrigation scenarios.

Scenario 1: Manual irrigation systems. The manual scenario is based exclusively on human interaction. The user starts and stops the irrigation corresponding to the areas of interest whenever he wants, and in whichever zone(s) he wants. This scenario is useful when, for example, the weather changes suddenly (in case of a storm).

Scenario 2: Schedule based irrigation systems. The second scenario involves irrigation schedules. The user can set schedules, by turning ON the Daily Schedule function on the qToggle app. Adjustments can be made using: Morning factor, Evening factor, Morning time, and Evening time. The user can select the time in the morning and evening when the irrigation should start and also select the amount of water by adjusting the Morning/Evening factor. We have computed (using 1) the minimum necessary run time for the sprinkler system, for each zone and we have obtained the following results: 10 min/day for zones B and D, 4 min/day for zone A and 6 min/day for zone C. These results should be adjusted taking into consideration the position of the garden (north/south), if there are shaded areas, if the summer is very hot, etc. If the proper quantity of water is not calculated, the tendency is to overwater.

Figure 4 presents the irrigation setup on the qToggle app, for Scenario 1 and Scenario 2.

Scenario 3: Sensor-based automatic systems. Humidity sensors are used to determine the appropriate schedule for irrigation. The humidity sensors are set to measure the soil humidity

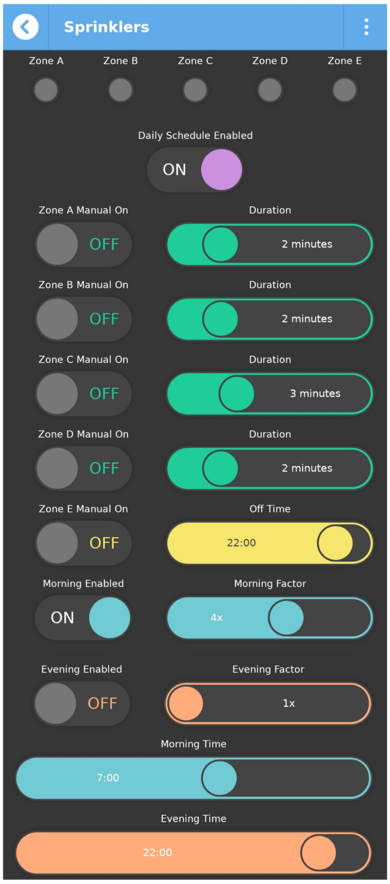

Figure 4. qToggle app for irrigations.

at the time of the day when the schedules are set. If the soil is dry, the sprinklers will start, if not, the irrigation is postponed until the next scheduled time. The concept is straightforward: the sensor measures the soil electrical resistance. A low resistance value indicates a high moisture content in the soil, while a high value is a sign of soil dryness. This signal is given to the microcontroller (the ESP chip), which transmits it to the Raspberry Pi. The Raspberry Pi processes the received moisture content data from the ESP chip and makes a decision regarding the water needs of the field, that is to say, open the electric valves or keep them shut. The valve in turn, acts on the pressure switch which commands the water pump. The sensor must be calibrated to reach the highest effectiveness of the irrigation system and may vary depending on environmental factors such as soil type, temperature, or electric conductivity. Details on sensor calibration are given in.[13] The equipment with the microcontroller (ESP chip) and the humidity

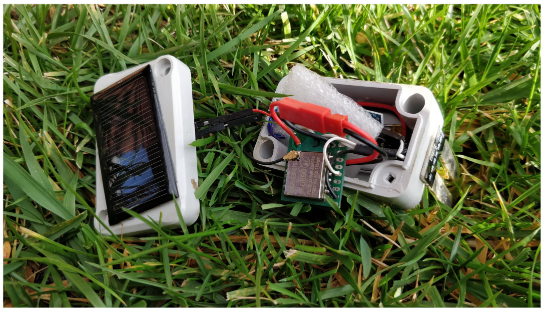

Figure 5. Equipment with the microcontroller and the humidity sensor.

sensor is shown in Figure 5. The ESP chip is powered by solar energy.

When using sensors, the climatic parameters make the irrigation routine more efficient. Smart irrigation systems bring savings, in terms of lower water waste, but they also diminish the need for manual labor. Also, by replacing manually operated watering systems with an automatic one, the chances of crops dying due to excessive or insufficient watering are minimized. Some results are given in Table 1, for three months in 2020, in a specific village in Romania, where the system has been implemented. The results correspond to Scenario 3. It should be mentioned that zone B and C are south-oriented, Zone A is west oriented, and placed between two buildings, so it is mostly shaded, while zone D has trees and is also partially shaded. Without sensors, water waste would have occurred in specific days when the soil was still wet.

The results obtained for water usage have been calculated taking into account the necessary run time for the sprinkler system, for each zone, the water debit of the pump, the number of rainy days (first column in Table 1), and the number of days when we have irrigated for each month (third and fourth columns in Table 1). The values presented in the last column represent the total amount of water, corresponding to all zones.

Regarding the price, the whole irrigation system costs around 760 USD, including the system inside the well, the irrigation controller, nozzles, and piping network, while the irrigation controller itself retails for around 70 USD. The qToggle app is open source and free for the consumer. In comparison, commercial vendors typically offer their own built in system and application packages at higher costs. This also implies a lock-in factor for the user, as the applications and control systems are not compatible with other off the shelf components. As an example, the closest similar solution we could identify on the market is OpenSprinkler, which is an open-source, web-based smart sprinkler controller for lawn and plant watering. Nevertheless, just the controller costs alone can vary between 80 and 186 USD in this case.

Table 1. Results for Three Months, Scenario 3.

Month	Rainy days	Zones A, D (days)	Zones B, C (days)	Saved water (mc)
June	9	9	11	22.96
July	10	8	10	24.5
August	6	17	19	14

CONCLUSIONS AND FUTURE WORK

In this article, we have proposed a simple, low cost, easily scalable, smart irrigation system that has all the above-mentioned characteristic. Our approach can be crucial in reducing water usage and waste in the house, prevent unwanted flooding or droughts, and reduce the overall time spent in watering and caring for the garden. Further, we have shown that by using real-life data from smart sensors, we can incorporate weather patterns and soil properties into the irrigation feedback loop. Because we chose off the shelf, open components, our system is readily available to anyone at a reduced cost, compared to other commercial solutions. Also, by avoiding vendor lockdown, our project offers seamless integration with other components and the qToggle environment, allowing the smart irrigation system to become an active component in the smart home with both offline and online control and reporting functionality. Finally, our simple approach to both the hardware setup and software application means most users will easily be able to make use of our system with little training required. In this context, we believe our smart irrigation system, as an integral component of the qToggle framework, can become a key player in the smart home market and is in line with current trends in consumer electronics and the main topics of interest in the field, specifically, interconnection, automation, and data awareness.

Sprinkler systems require a certain water flow or pressure to operate properly. If there is too little flow or pressure, the irrigation system will not work properly. Even with the zone division of the lawn, sometimes, due to the undersizing of the water network in some villages. For example, in areas with a lot of new houses using the same water network, the irrigation system can face problems. Therefore, future versions will rely on artificial intelligence, which will monitor water pressure trends and will determine the proper interval for watering. Another future idea is the use of sustainable energy and photovoltaic panels for the valves and water pump. A prototype for the later is already being tested with initial results being promising.

REFERENCES

1. L. García, L. Parra, J. M. Jimenez, J. Lloret, and P. Lorenz, "IoT-based smart irrigation systems: An overview on the recent trends on sensors and IoT systems for irrigation in precision agriculture," *Sensors*, vol. 20, no. 4, Feb. 2020, Art no. 1042.
2. P. K. Tripathy, A. K. Tripathy, A. Agarwal, and S. P. Mohanty, "MyGreen: An IoT-enabled smart greenhouse for sustainable agriculture," *IEEE Consum. Electron. Mag.*, early access, Feb. 2021, doi: 10.1109/MCE.2021.3055930.
3. S. P. Mohanty, "Consumer technologies build smart homes," *IEEE Consum. Electron. Mag.*, vol. 9, no. 1, pp. 4–5, 2019.
4. A.-B. Opare, A. Abdul-Rahman, and J. O. Agyemang, "An intelligent irrigation system for rural agriculture," *Int. J. Appl. Agricultural Sci.*, vol. 5, no. 3, pp. 75–81, 2019.
5. J. Li, W. Gu, and H. Yuan, "Research on IoT technology applied to intelligent agriculture," in *Proc. 5th Int. Conf. Elect. Eng. Autom. Control*, pp. 1217–1224, 2016.
6. B. G. Martini *et al.*, "IndoorPlant: A model for intelligent services in indoor agriculture based on context histories," *Sensors*, vol. 21, no. 5, 2021, Art no. 1631.
7. S. Koprda, Z. Balogh, D. Hruby, and M. Turcam, "Proposal of the irrigation system using low-cost arduino system as part of a smart home," in *Proc. IEEE 13th Int. Symp. Intell. Syst. Inform.*, Sep. 2015, Art. no. 229–233.
8. J. Karpagam, I. I. Merlin, P. Bavithra, and J. Kousalya, "Smart irrigation system using IoT," in *Proc. 6th Int. Conf. Adv. Comput. Commun. Syst.*, 2020, pp. 1292–1295.
9. M. Astutiningtyas, M. Nugraheni, and S. Suyoto, "Automatic plants watering system for small garden," *Int. J. Interactive Mobile Technol.*, vol. 15, no. 2, 2021, Art. no. 200.
10. A. Goap, D. Sharma, A. K. Shukla, and C. R. Krishna, "An IoT based smart irrigation management system using machine learning and open source technologies," *Comput. Electron. Agriculture*, vol. 155, pp. 41–49, 2018.
11. B. V. Raju, "An IoT based low cost agriculture field monitoring system," *J. Appl. Sci. Comput.*, vol. 4, no. 4, pp. 128–136, Apr. 2019.
12. S. K. Roy, S. Misra, N. S. Raghuwanshi, and S. K. Das, "AgriSens: IoT-based dynamic irrigation scheduling system for water management of irrigated crops," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 5023–5030, Mar. 2021.
13. J. C. Ekanayake and C. B. Hedley, "Advances in information provision from wireless sensor networks for irrigated crops," *Wireless Sensor Netw.*, vol. 10, pp. 71–92, 2018.
14. J. Gutierrez, J. F. Villa-Medina, A. Nieto-Garibay, and M. A. Porta-Gandara, "Automated irrigation system using a wireless sensor network and GPRS module," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 1, pp. 166–176, Jan. 2014.
15. H. Loubna and N. Bouchaib, "Integration of irrigation system with wireless sensor networks: Prototype and conception of intelligent irrigation system," in *Proc. World Congr. Eng. Comput. Sci.*, vol. I, pp. 56–62, 2018.
16. The Rain Bird DV/DVF Valve. Accessed on: Nov. 20, 2020. [Online]. Available: https://www.rainbird.com/products/dvdvf-series-valves
17. K. J. Singh and D. S. Kapoor, "Create your own Internet of Things: A survey of IoT platforms," *IEEE Consum. Electron. Mag.*, vol. 6, no. 2, pp. 57–68, Apr. 2017.
18. YL69 SparkFun Humidity Sensor. Accessed: Nov. 20, 2020. [Online]. Available: https://www.sparkfun.com/products/13322
19. Arc Spray Nozzles. Accessed: Nov. 20, 2020. [Online]. Available: https://www.rainbird.com/products/van-series-variable-arc-spray-nozzles.

Cristina Stolojescu-Crisan is an Assistant Professor with the Polyethnic University of Timisoara, Faculty of Electronics, Telecommunications and Informational Technologies, Communications Department, Timisoara, Romania. Contact her at cristina.stolojescu-crisan@upt.ro.

Bogdan-Petru Butunoi is currently working toward the Ph.D. degree with the West University of Timisoara, Faculty of Mathematics and Informatics, Computer Science Department, Timisoara, Romania. Contact him at bogdan.butunoi94@e-uvt.ro.

Calin Crian received the M.S. degree in electronic engineering from Telecom Bretagne, France, in 2010. Contact him at ccrisan@gmail.com.