An Innovative Smart System based on IoT Technologies for Fire and Danger Situations

Giorgio Cavalera[1], Roberto Conte Rosito[2], Vincenzo Lacasa[2], Marina Mongiello[3], Francesco Nocera[3], Luigi Patrono[4], and Ilaria Sergi[4]

[1] Cavalera s.r.l.
Galatone (LE), Italy

[2] ITS
Matera, Italy

[3] Dept. of Electric and Information Engineering
Politecnico of Bari
Bari, Italy

[4] Dept. of Innovation Engineering
University of Salento
Lecce, Italy

Abstract— **The SAFETY project aims to create a system able to support rescuers such as fire fighters in presence of danger events, especially when they have to intervene in large buildings with many floors and rooms, with different access points, and with numerous users. In these cases, in fact, it is very important to provide a smart decision support system able to promptly guide rescuers to the critical points of the building where there is the certainty that there are users to be rescued or artworks to be kept safe, without wasting resources. In order to achieve this goal, an innovative system architecture based on the use of technologies enabling the Internet of Things including sensors, mobile devices, Cloud technologies, mobile Apps, and embedded devices have been designed and developed in the SAFETY project.**

Keywords— Smart Building; Embedded System; Internet of Things; Cloud; Decision Support System; Mobile App; Proof-of-concept.

I. Introduction

The evolution of Information and Communications Technology (ICT) has enabled infrastructure and cities to be "smarter", leading to the emergence of the so-called smart cities characterized by the presence of innovative services and systems [1, 2, 3]. Innovative services introduced in smart cities can relate, for example, to waste and energy management, the elderly care and monitoring [4, 5, 6, 7], and so on. Moreover, in the context of smart cities and, more generally, in the creation of intelligent environments by exploiting new technologies, various problems have also been addressed, including energy efficiency [8, 9] and security [10] just to name a few. In recent years, the building industry has also been involved in digital evolution, which has paved the way for more and more innovations that can make buildings more and more attentive to energy saving, personal comfort and safety. From here the concept of "Smart Building"comes, an intelligent and connected building. In particular, security is an important element in the development of smart buildings. Nowadays, the topic of safety is very much felt by users of any type of building and the Internet of Things (IoT) [11] has allowed a digital revolution also in monitoring systems, with devices always connected with a high control of the building. Through the use of IoT-based systems, safety can be guaranteed in relation to any building and to any danger events. Related to the safety in buildings, special attention is paid to buildings hosting people like elderly and children (e.g., schools, residences for the elderly, etc.) and buildings containing high-value goods, such as museums or art galleries, where it is important the safety of both people and artworks.

Several research works deal with the safety in buildings. The system in [12] proposes a solution for fire monitoring exploiting a ZigBee-WiFi gateway able to transform a ZigBee network into a WiFi network. The system identifies the building where the fire occurred and notifies the building safety personnel. In [13], the proposed fire monitoring system is based on an IoT-aware approach exploiting embedded systems and Cloud technologies. This system includes a service for the building evacuation. [14] proposes a system based on the Arduino Mega board equipped with a smoke sensor and a camera. Data coming from sensors and cameras are sent to a monitoring system by using a wireless module connected to the Arduino board. Arduino Uno R3 was used in [15] for the realization of a home fire monitoring and warning system able to alert the building owner via GSM network. The work in [16] uses an approach based on Complex Event Processing (CEP) to monitor and manage the data provided by sensors. Anyway, the use of CEP does not prove to be as efficient and effective with respect to the new approach proposed in this work. Use of modern software architecture for monitoring data in IoT-aware environments have been largely explored based on middleware pattern such as reflection, adaptation or context-aware modeling [17, 18, 19]. As reported in previous works [16, 20], the main deficiency of most of the systems proposed in the literature as well as of commercial systems is the absence of a map of the building that clearly indicates the real time status of the room. Moreover, in many systems there is no direct connection with the rescue teams useful to support them indicating where intervene in case of emergency.

To fill these gaps, the main SAFETY project challenge is to realize an innovative and complete system, both hardware and software, able on the one hand, to guarantee the constant and real-time monitoring of certain parameters within a building and, on the other hand, to provide rescue teams with simple and immediate tools capable of supporting them in rescue operations in response to an event such as a fire.

The paper is structured as follows. In the Section II, an overall description of the SAFETY project is presented while Section III and IV describe the system architecture and some details of main components respectively. Finally, conclusions are drawn in Section IV.

II. Project Description

The project proposes an efficient implementation of an IoT-based system used for monitoring and controlling several environmental parameters through custom smart boxes installed in each room of the building. The primary goal of the research is to design, built and use an efficient software platform, cloud based, that can handle lot of requests from many devices and capable of managing and monitoring every element of the infrastructure (starting from device registration and location mapping, ending to his remote configuration and alert management checking).

The project involves as partners several companies and institutions such as "Informatica Tecnologie & Servizi s.r.l." (ITS), Cavalera s.r.l., Elettrosud soc. coop., Palmiotta s.r.l., Leader soc. coop., Sinapsi s.r.l., Polytechnic of Bari. In addition to the previous partners, the project foresees the presence of a research laboratory, IDA Lab of the University of Salento, Lecce (Italy) that is in charge of the scientific coordination of the project.

The project exploits the Living Labs approach, i.e. the use of real-life, experimentation and development environments where relevant stakeholders come together to explore, co-create and evaluate services offered by the proposed system. In particular, four different type of buildings have been identified and IoT devices will be installed in some of their rooms in the testing phase: two middle schools (in Lecce and in Bari, Italy), one Social and Healthcare Residence for the Elderly in Lecce, one services company in Bari and one public office such as the Municipality of Bari.

Furthermore, the provincial command of the fire fighters of Lecce will intervene to test the system effectiveness as a support tool for rescue teams.

III. Description of System Architecture

Figure 1 shows the architecture of the proposed system. Main components are:

- *Ambient monitoring Smart Boxes*. The Smart Box has two main features. The first is to periodically sample the value of some environmental parameters through a set of sensors, while the second is to send the collected data to the back-end server, where they will be processed to activate possible interventions. The proposed device is characterized by a modular structure, since the exact hardware equipment of the detection node and its communication interfaces will depend on many variables, such as the type of building (museum, school, hotel, etc.) in which the system will be installed, its size and topology, and so on. In this case, the main physical parameters to be detected are the temperature and humidity of the air, the presence of smoke, carbon monoxide, and Liquefied Petroleum Gas (LPG), the presence of people in the rooms of the building.
- *Installer Mobile App*. It is a multiplatform mobile application developed by using the Ionic Framework v4 [21]. This mobile application allows the Smart Box installer to use the building plan to quickly and easily associate each environmental monitoring device with its exact location with reference to the building layout.
- *Back-end server*. It is responsible for receiving data from monitoring devices (Smart Boxes) offering a suitable REST interface, for storing them in a persistent manner and processing them according to a well defined business logic able to establish whether an alert should be generated to the rescuers. The alert can consist of a notification message in the front-end application for rescuers and/or in a call to established telephone numbers. A very important aspect in this module is the data management. The data managed within the system must be appropriately modeled, so that it can be easily exchanged among modules of the system. Furthermore, these data must also be formatted according to appropriate standards in order to make them usable both internally and externally by third parties for further processing.
- *Front-end application*. It is a responsive Web application used to view the state of the building in real time, highlighting graphically on the building's layout the parameter values monitored by each Smart Box. This information will be used by the rescuers to have a global and concise view of the state of the building when an event occurs (e.g., a fire) and to decide promptly in which rooms of the building to intervene.

The software system architecture is modeled as a result of a decision-making process [22, 23] to evaluate the following relevant issues:

- The system must be up 24/7 since it is an environmental monitoring system.
- The amount and volume of data is huge since sensor`s data are constantly monitored and checked, hence the whole amount of stored data can reach a gigantic number.

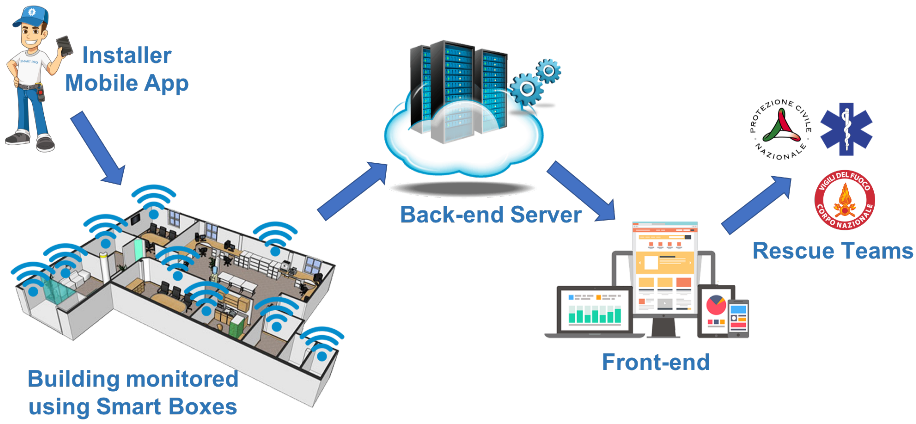

Figure 1. Overall system architecture.

- The system must be fault tolerance because a possible fall of the system would compromise the safety of people, objects of inestimable value and buildings.
- The system must be highly scalable.
- The system must ensure ease of updating functions and/or addition of other functionalities without turning the system off.

In order to satisfy these requirements, we decided to optionally use a distributed object model based on the client-server paradigm. The client-server model is a three-tier model. The client will be of the "thin" type, meaning that it only provides an interface to access information. Main components included in the architecture are a login interface, an administrator interface, an OpenData interface, an Alarm report.

To satisfy four main non-functional requirements such as security, usability, reliability, and supportability, the following patterns are included in the architectural model as shown in Figure 2 (a, b, c, d), client, controller, model and database server and remote components respectively.

Broker for maintainability and flexibility: it manages the communication between the Smart Boxes and the model.

Security proxy: two proxies are planned; the first one enables access to the system only through user recognition. The second deals with encrypting and decoding of data sent by the Smart Boxes.

Repository for data persistence, responsible for accessing data.

Database access layer for portability: the data recovered from the DBs are reprocessed through a layer that formats data so that they can be transferred to the system without problems; this layer makes the system transparent to the RDMBS technology.

Model view controller for low coupling and usability: the use of a modular vision of the system allows having a low level of coupling between the various components and this allows substantial modifications without affecting the other components.

Adapter for adaptability: the system includes a mobile App hence, to ensure that it interacts perfectly with the server, the adapter pattern will act as a link to the mobile app.

Publish/subscribe: it allows communications among the various distributed components.

Moving the software architecture to the cloud architecture involves adding the following patterns:

Stateless Component in the Controller component of Figure 2b allows increasing the elasticity and robustness of an application by providing an external component to the application that deals with managing the status of the services and the scaling-out.

Watchdog pattern in the Controller component of Figure 2b allows the detection of errors in instances created by the Stateless Component pattern. Components are resized and multiple instances are distributed over redundant resources. Component instances are monitored by a separate Watchdog component and replaced in case of errors.

Strict Consistency in the remote components of Figure 2c allows data to be replicated in different locations in order to increase system availability, improve response times and avoid data loss due to malfunctions, all while maintaining consistent data.

IV. Implementation Details

A. Smart Box

The Smart Box is an intelligent device able to manage several environmental sensors connected to it. In particular, six different sensors have been foreseen: humidity, temperature, carbon monoxide, smoke, presence (PIR sensor), and LPG. It is equipped with a universal power supply with several opportunities in terms of source. Additionally, it has a backup battery in case of temporary absence of main power. The Smart Box is also designed for direct supply from renewable sources, typically photovoltaic modules. It is equipped with a processing unit capable of acquiring and pre-processing in real time signals coming from the sensors.

Connectivity is one of the strengths of the Smart Box. In fact, it allows cable connections using the most common standards, such as ethernet, but can also use wireless

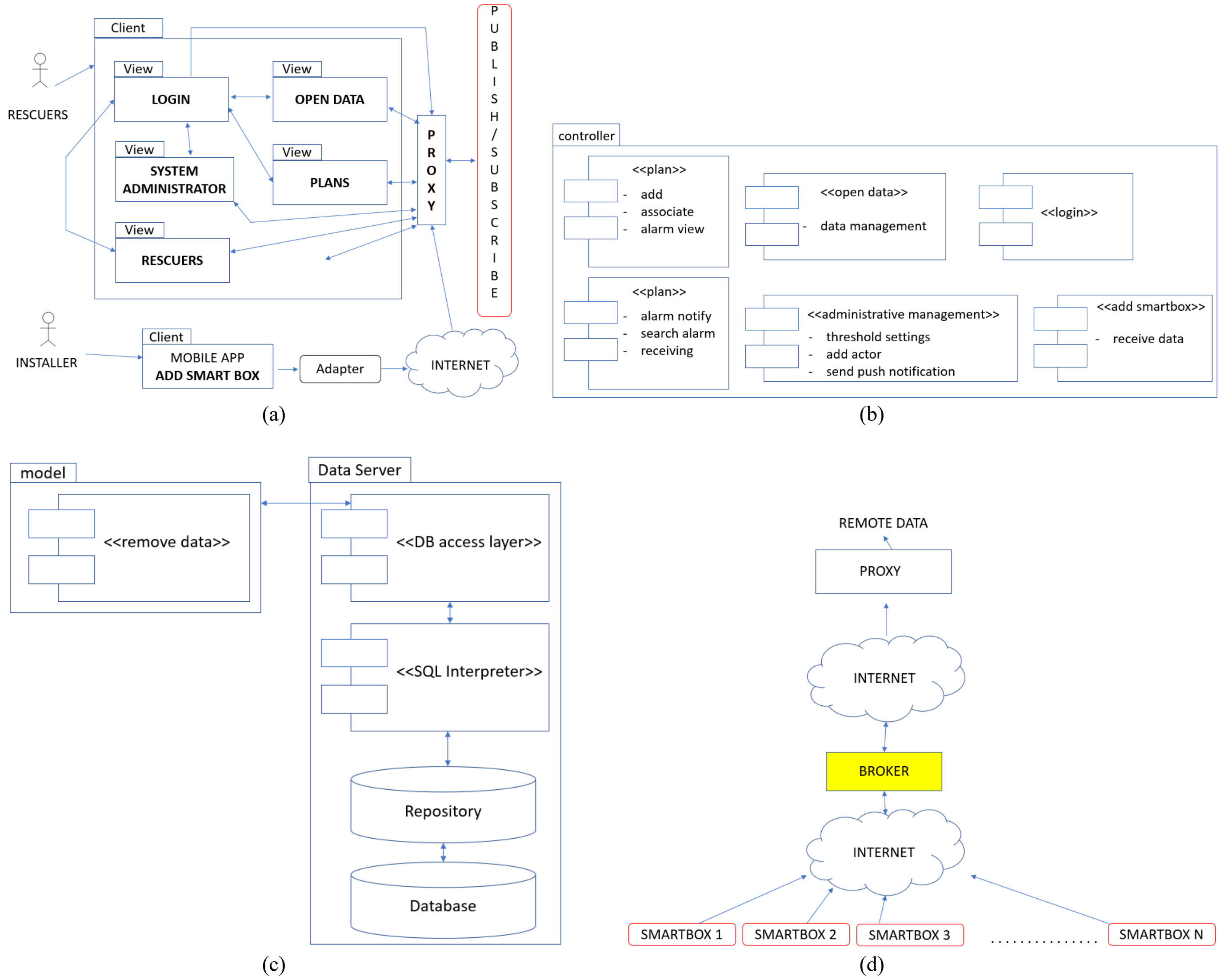

Figure 2. Software Architecture of the Safety Platform. Client component (a), Controller component (b), Model and database server components, and Remote components of the Safety Platform (d).

connectivity for the connection to Wi-Fi access points, to portable Bluetooth Low Energy devices, to existing networks in the Smart City, and to wireless home automation systems.

The Smart Box is able to send data in Cloud to be stored, processed and analyzed.

B. *Mobile App and Front-end*

The front-end of the system architecture is a multiplatform App used by the installer to initialize the Smart Boxes when they are installed in a room. The installer uses it also for the optimized management of their ordinary and extraordinary maintenance, with the possibility of receiving push notifications directly from the platform for the management of the intervention.

A typical scenario consists of an operator that scans the Smart Box, selects the building, the floor and the room where the Smart Box will be located. The “Smart Box census” will add the information about the new Smart Box in the repository of the whole infrastructure enabling the system to recognize its identifier and the place where it is located.

Several non-functional requirements are satisfied by the app: Portability, Scalability, Fault Tolerance, Reliability, Security.

Portability is reached using the Ionic Framework [21] that allows developing an app using only web languages (e.g. Typescript, CSS, JavaScript) and cross-compiling it for many platforms: Android, IOS, Windows Phone.

The scalability is achieved using the Firebase framework. In fact, although the entire project has a client-server style, the server used is Firebase [24], that provides many methods to implement the scalability requirement. As far as the Data Formats is concerned, we used the “Firebase Storage” to host two files: *namebulding.json*, *namebuilding.xml*, that contain the blueprints about the building in which we want to add the

Smart Box. These files are used in order to retrieve the information about the localization of the operator. In fact, once the operator has scanned the smart box, s/he selects the position (e.g., hotel, floor, room) and adds the Smart Box. Actually, to reach the scalability goal, we have implemented the Strategy Pattern to validate and handle the XML and JSON format. As a result, it is quite simple to add every other well-known desired format.

To satisfy the fault tolerance property we have handled the errors generated by the system so it will not crash.

Reliability requirement is intrinsically achieved by using Firebase, according to the Service Level Agreement of the service.

To ensure the security property we have used the built-in methods available in Firebase, "Firebase Security Rules".

C. Back-end Server

To fit common scenarios in this field, we are working on a complete infrastructure, built using open-source technologies, able to manage data transaction, between IoT devices and central cloud platform, using common protocol: MQTT [25] and REST.

The back-end server exploits several technologies for its sub-components: PHP [26] for the backend server, MariaDB [27] as principal data storage, in join with MQTT for real-time data management and orchestration. We will work with lot of time series that needs to be stored and handled in a strong environment. With MQTT we will ensure fast, reliable, service infrastructure allowing users to see real time data on the frontend, MariaDB will ensure strong storage platform that will be used for many operation: alert management, big data analysis and reporting (using specific data mart configuration necessary to ensure good performance when accessing data).

V. Conclusions

The SAFETY system offers an innovative solution useful to intervene promptly and in the best way inside buildings in emergency situations, such as when there is a fire. The present project aims at the realization of a complete system not present on the market that has the ambition to improve safety inside buildings by offering, on the one hand, useful tools for a timely intervention in case of emergency, but from the another side, focusing on constant monitoring of buildings and therefore on the prevention of risk situations.

Acknowledgments

SAFETY (Smart Aid System for Fire and Danger situations based on IOT technology) is a research project with code 6RPCRL1. ("Intervento cofinanziato nell'ambito del POR Puglia FESR-FSE 2014-2020 - Asse prioritario 1 - Ricerca, sviluppo tecnologico, innovazione - Azione 1.4.b - BANDO INNOLABS – SOSTEGNO ALLA CREAZIONE DI SOLUZIONI INNOVATIVE FINALIZZATE A SPECIFICI PROBLEMI DI RILEVANZA SOCIALE). Public Notice and subsequent reissues. AD n. 144/13 of 08/02/2017- Section DD Research and Innovation Capacity 'Institutional n. 37 of 28 March 2017.

References

[1] Mainetti, L., Patrono, L., Rametta, P., "Capturing behavioral changes of elderly people through unobtruisive sensing technologies", 2016 24th International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2016, art. no. 7772126. DOI: 10.1109/SOFTCOM.2016.7772126.

[2] Mainetti, L., Patrono, L., Stefanizzi, M.L., Vergallo, R., "A Smart Parking System based on IoT protocols and emerging enabling technologies", IEEE World Forum on Internet of Things, WF-IoT 2015 - Proceedings, art. no. 7389150, pp. 764-769. DOI: 10.1109/WF-IoT.2015.7389150.

[3] Mainetti, L., Mighali, V., Patrono, L., "An IoT-based user-centric ecosystem for heterogeneous Smart Home environments",(2015) IEEE International Conference on Communications, 2015-September, art. no. 7248404, pp. 704-709.

[4] Mainetti, L., Patrono, L., Secco, A., Sergi, I.,"An IoT-aware AAL system for elderly people",(2016) 2016 International Multidisciplinary Conference on Computer and Energy Science, SpliTech 2016, art. no. 7555929.

[5] Mighali, V., Patrono, L., Stefanizzi, M.L., Rodrigues, J.J.P.C., Solic, P., "A smart remote elderly monitoring system based on IoT technologies", (2017) International Conference on Ubiquitous and Future Networks, ICUFN, art. no. 7993745, pp. 43-48.

[6] Mulero, R., Aitor, A., Gorka, A., Abril-Jiménez, P., Waldmeyer, M.T.A., Castrillo, M.P., Patrono, L., Rametta, P., Sergi, I.,"An IoT-aware approach for elderly-friendly cities",(2018) IEEE Access, 6, pp. 7941-7957.

[7] Almeida, A., Mulero, R., Rametta, P., Urošević, V., Andrić, M., Patrono, L., "A critical analysis of an IoT—aware AAL system for elderly monitoring",(2019) Future Generation Computer Systems, 97, pp. 598-619.

[8] Alessandrelli, D., Mainetti, L., Patrono, L., Pellerano, G., Petracca, M., Stefanizzi, M.L.,"Performance evaluation of an energy-efficient MAC scheduler by using a test bed approach",(2013) Journal of Communications Software and Systems, 9 (1), pp. 84-96.

[9] Alessandrelli, D., Mainetti, L., Patrono, L., Pellerano, G., Petracca, M., Stefanizzi, M.L.,"Implementation and validation of an energy-efficient MAC scheduler for WSNs by a test bed approach",(2012) 2012 20th International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2012, art. no. 6347615.

[10] De Rubertis, A., Mainetti, L., Mighali, V., Patrono, L., Sergi, I., Stefanizzi, M.L., Pascali, S., "Performance evaluation of end-to-end security protocols in an Internet of Things", (2013) 2013 21st International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2013, art. no. 6671893.

[11] Mainetti, L., Patrono, L., Vilei, A., "Evolution of wireless sensor networks towards the Internet of Things: A survey", 2011 International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2011, art. no. 6064380, pp. 16-21.

[12] L. Yunhong and Q. Meini, "The design of building fire monitoring system based on zigbee-wifi networks," in Measuring Technology and Mechatronics Automation (ICMTMA), 2016 Eighth International Conference on. IEEE, 2016, pp. 733–735.

[13] H. M. Poy and B. Duffy, "A cloud-enabled building and fire emergency evacuation application," IEEE Cloud Computing, vol. 1, no. 4, pp. 40–49, 2014.

[14] M. S. A. Azmil, N. Ya'Acob, K. N. Tahar, and S. S. Sarnin, "Wireless fire detection monitoring system for fire and rescue application," in Signal Processing & Its Applications (CSPA), 2015 IEEE 11th International Colloquium on. IEEE, 2015, pp. 84–89.

[15] S. Suresh, S. Yuthika, and G. A. Vardhini, "Home based fire monitoring and warning system," in ICT in Business Industry & Government (ICTBIG), International Conference on. IEEE, 2016, pp. 1–6.

[16] Mongiello, M., Patrono, L., Di Noia, T., Nocera, F., Parchitelli, A., Sergi, I., Rametta, P., "A Complex Event Processing based smart aid system for fire and danger management," 7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI), Vieste, Italy, June 15-16, 2017.

[17] Marina Mongiello, Tommaso Di Noia, Francesco Nocera, Eugenio Di Sciascio, Angelo Parchitelli. Context-aware design of reflective middleware in the Internet of Everything. 1st International Workshop on Formal to Practical Software Verification and Composition (VeryComp 2016) - Co-located with Software Technologies: Applications and Foundations (STAF 2016), July 2016.

[18] Tommaso Di Noia, Marina Mongiello, Francesco Nocera, Eugenio Di Sciascio. Persistence on different databases via reflective IoT Middleware. 24th Italian Symposium on Advanced Database Systems (SEBD 2016) - june 2016. ReIOS: Reflective Architecting in the Internet of Objects.

[19] Marina Mongiello, Gennaro Boggia, Eugenio Di Sciascio 4th International Conference on Model-driven Engineering and Software Development (Modelsward 2016) – 2016.

[20] M. Mongiello, F. Nocera, A. Parchitelli, L. Patrono, P. Rametta, L. Riccardi, I. Sergi, "A Smart IoT-Aware System for Crisis Scenario Management," Journal of Communications software and Systems, vol. 14, no.1, March 2018, pp. 91-98.

[21] Ionic Framework. Available on: https://ionicframework.com/. March, 2019.

[22] Niko Makitalo, Francesco Nocera, Marina Mongiello, Stefano Bistarelli Architecting the Web of Things for the Fog Computing Era. IET Software Journal, Volume 1751-8806 – 2018.

[23] Francesco Nocera, Marina Mongiello, Eugenio Di Sciascio, Tommaso Di Noia, MoSAIC: a Middleware-induced Software ArchIteCture design Decision Support System. Proceedings of the 12th European Conference on Software Architecture – 2018.

[24] Firebase. Available on: https://firebase.google.com/. March, 2019.

[25] MQTT protocol. Available on: http://mqtt.org/. March, 2019.

[26] PHP Scripting Language. Available on: https://www.php.net/. March, 2019.

[27] MariaDB. Available on: https://mariadb.org/. March, 2019.