



INTERNATIONAL CONFERENCE INFORMATION AND COMMUNICATION TECHNOLOGIES IN BUSINESS AND EDUCATION



CONFERENCE PROCEEDINGS



Publishing house "Science and economics"
University of Economics - Varna

International Conference
INFORMATION AND COMMUNICATION
TECHNOLOGIES IN
BUSINESS AND EDUCATION

Conference proceedings

**INFORMATION AND COMMUNICATION
TECHNOLOGIES IN
BUSINESS AND EDUCATION**

**Proceedings of the International Conference
dedicated to the
50th anniversary
of the Department of Informatics**

2019

University publishing house “Science and economics”
University of Economics – Varna

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Permission for use must always be obtained from University publishing house “Science and economics”. Please contact izdatel@ue-varna.bg

Published reports have not been edited or adjusted. The authors are solely responsible for the content, originality and errors of their own fault.

With the support of ADASTRA, 411 Marketing Business Services.

© University publishing house “Science and economics”,
University of Economics – Varna, 2019.

ISBN 978-954-21-1004-0

PROGRAMME COMMITTEE

Abdel-Badeeh M. Salem, Egypt

Aijaz A. Shaikh, Finland

Aleksandar Dimov, Bulgaria

Alexander Feldman, USA

Alexander Sergeev, Russia

Antoanela Naaji, Romania

Avram Eskenazi, Bulgaria

Dariusz Mrozek, Poland

Dimitrina Polimirova, Bulgaria

Dmitriy Berg, Russia

Elena Nechita, Romania

Emil Denchev, Bulgaria

Georgi Dimitrov, Bulgaria

Ivan Popchev, Bulgaria

Jeanne Schreurs, Belgium

Julian Vasilev, Bulgaria

Kameliya Stefanova, Bulgaria

Krasimir Shishmanov, Bulgaria

Maciej Czaplewski, Poland

Maciej Pondel, Poland

Maksim Medvedev, Russia

Maria Hristova, Bulgaria

Marian Cristescu, Romania

Marina Medvedeva, Russia

Michael Voskoglou, Greece

Nadezhda Filipova, Bulgaria

Nikola Nikolov, Ireland

Niv Ahituv, Israel

Pavel Petrov, Bulgaria

Rami Malkawi, Jordan

Sara Paiva, Portugal

Silvia Parusheva, Bulgaria

Snezhana Sulova, Bulgaria

Todorka Atanasova, Bulgaria

Valentin Kisimov, Bulgaria
Valery Trofimov, Russia
Veselin Popov, Bulgaria
Violeta Kraeva, Bulgaria
Vladimir Sulov, Bulgaria
Vladimir Zanev, USA
Zdzislaw Polkowski, Poland

ORGANIZING COMMITTEE

Chairman: Prof. Vladimir Sulov, PhD

Members:

Prof. Avram Eskenazi, PhD
Prof. Julian Vasilev, PhD
Assoc. Prof. Todorka Atanasova, PhD
Assoc. Prof. Silvia Parusheva, PhD
Assoc. Prof. Snezhana Sulova, PhD
Assoc. Prof. Pavel Petrov, PhD

Secretary: Tijen Talib

FOREWORD

These proceedings contain the papers of the International Conference “Information and Communication Technologies in Business and Education” which took place at the University of Economics – Varna, Bulgaria, 18 October 2019.

The international scientific conference is dedicated to the **50th anniversary of the Department of Informatics at the University of Economics – Varna**. The conference is also dedicated to the 100th anniversary of the University. The included papers describe recent scientific and practical developments in the field of information and communication technologies, information systems, and their applications in business and education.

The papers in the Proceedings are peer reviewed and are checked for plagiarism.

C O N T E N T S

1. Kamelia Stefanova, Dorina Kabakchieva Challenges and Perspectives of Digital Transformation	13
2. Julian Vasilev, Marian Cristescu Approaches for information sharing from manufacturing logistics with downstream supply chain partners	24
3. Abdel-Badeeh M. Salem Computational Intelligence in Smart Education and Learning	30
4. Marian Niedźwiedziński Proposal of the Multi-aspect Analysis of ICT's Needs in Management	41
5. Krasimir Shishmanov Digital Transformation in Bulgaria	52
6. Tatyana Makarchuk, Valery Trofimov, Svetlana Demchenko Modeling the life cycle of e-learning course using Moodle Cloud LMS	62
7. Vanya Lazarova Exploring the functionality of Bulgaria eGovernment site	72
8. Veselin Popov, Petya Emilova Big data and information security	83
9. Luben Boyanov Approaches for enhancing digitalization and digital transformation in supply chain management	91

10. Monika Tsaneva	
Challenges of GDPR compliance in consumer financing companies	103
11. Todorka Atanasova	
Main factors influencing digitization in construction companies	116
12. Silvia Parusheva	
Digitalization and digital transformation in construction – benefits and challenges	126
13. Snezhana Sulova	
The Usage of Data Lake for Business Intelligence Data Analysis	135
14. Pavel Petrov	
Web security technologies used in banks of Estonia, Latvia and Lithuania	145
15. Tanka Milkova	
ABC analysis of inventory in MS Excel	155
16. Ivan Kuyumdzhev, Radka Nacheva	
Choosing Storage Devices with Consideration of Backup and Restore Performance in MS SQL SERVER	166
17. Yanka Aleksandrova	
Predicting Students Performance in Moodle Platforms Using Machine Learning Algorithms	177
18. Latinka Todoranova, Bonimir Penchev	
M-learning Applications	188
19. Boris Bankov	
The Impact of Social Media on Video Game Communities and the Gaming Industry	198

20. Velina Koleva, Svetoslav Ivanov	
Social media and the recruitment of IT professionals in Bulgarian IT companies	209
21. Deyan Mihaylov	
Modeling and Simulation of Some Functions of Two Independent Random Variables	225
22. Oumar Sy	
Using Association Rules as Semantics for Domain Ontologies	233
23. Ina Stanoeva	
Social media possibilities for improving arts marketing performance	247
24. Iskren Tairov, Vladislav Vasilev	
Perspectives on mobile devices adoption in healthcare sector	257
25. Angelin Lalev	
Deep Neural Networks for Detection of Credit Card Fraud	263
26. Miglena Stoyanova	
Priorities for Digital Transformation in Property Management	275
27. Mihail Radev	
Organizational Variants of IT Department at the University	284
28. Mariya Armyanova	
IoT problems and design patterns which are appropriate to solve them	291
29. Yavor Christov	
Corporate Big Data and Hybrid Integration Platforms	306

30. Ivan Belev	
Making financial data readable using Inline XBRL	314
31. Ivan Belev	
eXtensible Business Reporting Language (XBRL) – a technical overview	322
32. Anna Sobczak	
Entrepreneurship in the education of university students	331
33. Mihaela Markova	
E-commerce in the process of digital transformation: preparedness, requirements and technological tendencies	343
34. Petya Petrova	
Basic Soft Computing Methods In User Profile Modeling	353
35. Petar Dimitrov	
Methods for storing authentication data – a historical review	369
36. Svetoslav Ivanov	
Development stages of starting software company, problems and approaches for software development	382
37. Antonio Hadzhikolev	
The role of the smart city concept in the process of urban transformation	397
38. Oumar Sy	
Ontology-based materialized views for reverse engineering: Toward a set of conceptual guidelines	410

IOT PROBLEMS AND DESIGN PATTERNS WHICH ARE APPROPRIATE TO SOLVE THEM

Mariya Armyanova¹

¹ UE-Varna, Bulgaria, armianova@ue-varna.bg

Abstract

Internet of Things provides the possibility to cooperation and manage the devices with the global network. But the development of IoT systems is fraught with a variety of complicated problems and a good solution is to use design patterns. IoT patterns are still not strictly defined and classified as the domain is under development in the moment. The purpose of the report is to present common problems and opportunities to overcome them through IoT patterns. The applied approach includes research and analysis of scientific publications, official websites and design patterns classification. The report results are useful for IoT system developers who want to know about common problems and how to be solved and where to find appropriate design patterns to their problem. The main conclusion of the report is that in order to be easily found when needed, the IoT patterns are appropriate to be classified by the groups of problems they solve. In the report, the approach for classification of IoT design patterns is pointed out and by the defined criteria they are classified.

Keywords: *design patterns, IoT, security, Design, Architecture.*

INTRODUCTION

Different organizations and research institutes realize the IoT, by using manifold technologies, standards, which are involved by different organizations and sectors. So it often leads to conflicts of interest. Therefore, when developing an IoT system, there are various complex problems to solve. Problems are compounded by increasing the count of devices connected to the global network. Many factors are driving the process from reducing the device cost to the new communications technology capabilities. The concept of tying different devices to the Internet is seen as Internet of Things (IoT). The IoT is a symbiosis between various leading technologies that aim to enable device recognition. The device number on the global network is expected to reach 26.66 billion in 2019. The trend is a steady increase in devices connected in IoT. Globally, their number is expected to reach 75.44 billion by 2025, which is threefold increase in 5 years (SRD, 2016). The trend is presented in fig. 1.

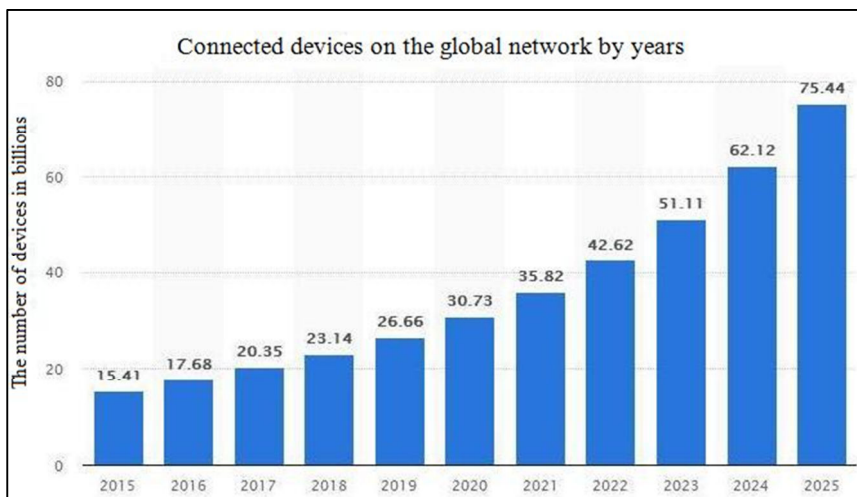


Figure 1. The trends in the device number for the period 2015-2025

The IoT is based on such technologies as machine learning, artificial intelligence, big data and intelligent analyzes. IoT platform is used in various domains from smart homes and smart cars to agriculture, healthcare and smart supply chain. Each domain and technology sets specific requirements, which are often contradictory. So this poses a variety of complicated problems to the development of systems, enabling the work of different devices into the IoT environment. For some of the problems, the development teams have already found an effective solution, which is formally described as a pattern. The patterns present a concept that enables developers to reuse the experience of problem solving. They can be seen as descriptions of common problems, their abstract solutions, and their consequences.

The individual manufacturers often develop devices only for a specific domain area, such as healthcare or smart cars, smart homes, so for such problems it is leading to an increase in the number of different solutions. The existence of too many IoT standards and technologies proposed by various device developers, as well as a lack of common understanding in the domain (Guth et al. 2016) again popularizes the use of design patterns as a possible solution to problems.

1. THE IOT AND ITS PROBLEMS

Internet of Things (IoT) is seen as a concept of linking different devices to the Internet so that they can be recognized by other devices. The term was introduced in 1999 by Kevin Ashton at his presentation.

The technical advancements and the communications developments also provide opportunities for the IoT development. Since the concept of IoT can be used in different domains, so the term refers the different systems, such as Smart Homes, Smart Office (Le Gal et al., 2001; Röcker, 2010), Smart Grids (Kopp et al., 2015), or the Smart City concept (Nam and Pardo, 2011; Su et al., 2011), the Industrial Internet (IIC, 2015), and others. However, the main idea is to send the results of the sensors to the Internet to enable the analysis of all the accumulated data. The results of the analysis are used to the better device management in the domain.

According to Voas, the IoT involves sensing, computing, communication, and actuation. The IoT implements device unification into a ubiquitous layer of cyber-physical devices would then be able to measure itself and its environment; combine, process, and analyze these measurements to gain new insights; and use these insights to optimize itself and the environment (Voas, 2016).

Some of the difficulties in the IoT system implementation is due to the numerous requirements for the IoT platforms. The requirements relate to performance, reliability, scalability, reuse, modification and interoperability. The main IoT platform features that make implementation difficult are, for example, the lack of centralized management, continuously device connection or disconnection to the system, user anonymity, etc. Despite their implementation difficulties, they have many advantages for IoT systems. For example, decentralization provides scalability and reliability because it involves the use of all resources, eliminating intensive traffic to only one point, which reduces delays and guarantees the reliable work of the system even in the event of a resource failure. Of course, for developers this architecture requires solving of many problems, such as security-related issues, dynamic device connectivity, decentralized data collection and processing, and more.

The IoT systems complexity is also due to the distribution of tasks across many devices. The situation is also complicated by the physical remoteness and distribution of roles in the system development between many different often independent participants. The device is developed by a manufacturer, then is installed by another company, and is often maintained by a company that is different from the software development company.

The many of the IoT system problems are related to security. In a network environment, the device needs to communicate with many potential applications and systems. Therefore, it is difficult to guarantee its security, because of the difficulty of distinguishing malicious users who want to gain control of the device or access its network.

Another problem is that a potential target for malicious attacks is the device network. It often is attacked by sending unsolicited communication requests from an unreliable communication partner. As a result, security trust issues arise. Device owners are afraid to provide access to their devices or their data without full control, because the device communication with third parties or its access security is not always well protected. The devices should react differently depending on the request sender.

The problems are compounded by the growing number of IoT users' devices. Rights management, data sharing, and device control are complicated, too. They are often accessed through great number of gateways, and data collection and processing is through different cloud systems.

The integration of devices into a united system causes and other problems. Some devices do often not support the networks communication technology or protocol, because they use different standards. In order to save energy or due to other network features, some devices are not permanently plugged into the general network environment, making it difficult to access them.

The main challenges that IoT system developers face can be reduced to the following categories of problems:

1. The choice of the right architecture for IoT applications which enable the acquisition and processing of huge amounts of sensor data simultaneously;
2. The combining of different devices, some of which do not support the communication protocol or the technology, into a common IoT system, or integrating multiple different protocols into an IoT platform;

3. The ensuring of the communication security and reliability with the devices, as well as the physical access to them;

4. The providing energy-saving capabilities and, at the same time, to ensuring that the device is switched on immediately when needed.

The security issues are becoming more important because of the increasing number of devices, which leads to more frequent hacking attacks as the number of potential targets increases. Moreover, the use of different technologies and mechanisms makes difficult to create effective protection for all of them.

The creating of an IoT system requires the provision of multiple aspects, which complicates the problems. The design patterns that summarize the experience of specialists offer the ability to solve some of the systems development problems.

2. THE IOT DESIGN PATTERNS

The IoT is a dynamically evolving domain, so new design patterns are constantly described to solve the problems that arise with the IoT systems development. Because of the many patterns there is confusion among developers regarding the use of the right pattern in the project situation. Therefore, the report offers an opportunity to classify the patterns to bring arrangement among set of IoT patterns. In 2018 (Bloom and etc., 2018) is proposed the grouping of patterns based on the architecture layers and categories of IIoT applications, which are only suitable for architectural patterns.

Washizaki (Washizaki et al., 2019) offers a patterns classification based on the level of abstraction. Here, the patterns are divided according to the architecture layer into: high-level or architectural patterns; mid-level or design patterns; and low-level or software-dependent patterns. Such a classification could direct the developer to an appropriate patterns for the architecture layer he is developing, but it is insufficient because most often the developer tries to solve a specific problem. The patterns are too many to find the right one. Therefore, it is necessary to add more classification criteria that are specific only to patterns in the domain area of IoT systems.

IoT patterns are a small part of all patterns used in the systems development. They are a subset derived from the classification of patterns

by the criterion Purpose in the general classification scheme (Armiyanova, 2018). Therefore, we propose the basic criteria for the classification of IoT patterns, the criterion Goal. The criterion Layer of the architecture is also appropriate, as it subdivides the patterns according to the development phases and, with such classification, the developer can easily find the appropriate pattern at the appropriate level.

The criterion Goal indicates the specific problem the pattern is addressing. According to Kardell (1997), the Goal is determined by the elements of the system that are influenced by the pattern: integration, functionality, communication, access, security, etc. This classification is applicable not only to design patterns, but also to architectural patterns that can also add goals. But the architectural patterns described to do moment are not so many, so they do not need to be classified.

Koster also presents IoT patterns according to their goals (Koster, 2014; Chandra and Mahindra, 2016). But the categories offered by the authors are many, as they classify all patterns that are suitable for use in IoT systems, and not specifically for IoT. Therefore, there are some categories that can be removed, such as information computing patterns. They are not specific for IoT systems and can be used to build cloud systems, too.

By the criterion Goal the patterns could be classified according to the main groups of problems that they addressed in the previous point. Therefore, design patterns can be divided into the following main groups, defined by the main problem categories: Integration patterns, Communication patterns, Security patterns, and Energy Saving patterns. By adding a group of Architectural patterns to the design patterns, a classification is obtained to allow easy detection of the patterns when needed (Fig. 2).

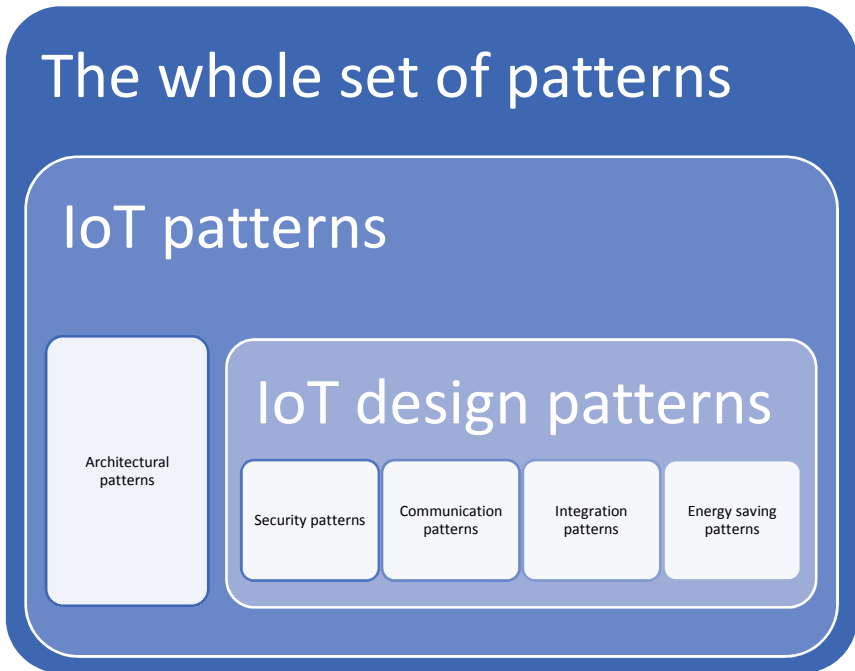


Figure 2. The classification scheme of the categories IoT patterns

The Security patterns are designed to prevent accidental insertion of code vulnerabilities and mitigate the effects of these vulnerabilities. They integrate knowledge of security and system structure, enabling the development and improvement of software. They allow to integrate security policy with software design and development. The Security patterns are numerous. Their levels range from architectural patterns involving system-wide design to execution-level patterns relating to the implementation of individual functions or methods in the system. Some of the security issues are common to all systems. Therefore, in the development of an IoT system, both specialized IoT patterns and patterns can be used that realize one or another aspect of security across all systems. For example, the Whitelisting Firewall patterns of (Villarreal et al., 2013) introduces a firewall to control communication and filter it only to authorized interlocutors without delaying it. Fernandez also offers simi-

lar patterns (Fernandez, 2013; Fernandez et al., 2014). He presents two firewall patterns to control network access to cloud resources. Hafiz cloud security patterns (Hafiz, 2013), Schumacher web application patterns (Schumacher et al., 2006), Dara symmetric and asymmetric encryption patterns (Dara, 2014), or the patterns of Romanovsky (Romanovsky, 2001) and Kienzle (Kienzle et al., 2002) are also suitable.

In addition to generic patterns, IoT-specific systems have been created. Our aim is to look at these patterns. Such are the Reinfurt patterns (Reinfurt et al., 2017). He offers Trusted Communication Partner, Outbound-Only Connection, Permission Control, Personal Zone Hub, Whitelist, and Blacklist design patterns. The Trusted Communication Partner pattern is intended to prevent the device or its network from being accessed by random communication partners by allowing the device to communicate with the device solely in a list of authorized pre-known partners. Another solution to prevent unauthorized access is offered in the Outbound-Only Connection pattern, which supposes only those incoming communications requests that respond to a device-initiated connection to be answered. The Permission Control pattern guarantees the security of device data by retaining on the backend server of the data access rights of the communication partners. The Personal Zone Hub involves the creation of a Hub that allows the management of the permissions, data sharing and control of all devices of a user. The Whitelist pattern involves creating a list of users that have access rights, with no one outside that list having access. The Blacklist pattern is the opposite. It involves creating a list of unreliable partners who are denied access rights.

The another major set of patterns are integration patterns that ensure way to combine devices located at different end nodes with different protocols in an IoT platform. Koster offers design patterns that can use different network and device technologies to solve problems with the physical infrastructure of IoT (Koster, 2014). However, its solutions are not described as typical patterns. Qanbari (Qanbari et al., 2016) also offers patterns for combining devices related to edge: Edge Provisioning Pattern, Edge Code Deployment Pattern, Edge Orchestration Pattern, Edge Diameter Of Things (DOT) Pattern. The Edge Provisioning Pattern provides control over a large number of hard-to-reach scattered end de-

vices, enabling them to be easily reconfigured, as well as to easily incorporate new ones. The Edge Code Deployment Pattern provides the ability to easily maintain device software through decentralized Git version control located on backend server. Edge Orchestration Pattern distributes functionality in such a way that it allows end nodes to control, configure and manage their own end devices, monitor their status, discover the services they need. The Edge DOT Pattern requires the creation of a Metering Server that unifies the way services are measured across endpoints, as different providers may use different usage patterns, such as event-based or time-based.

The next large group of patterns are communication patterns. Peter (Peter, 2016) offering such communication patterns: Request / Response, Event Subscription, Asynchronous Messaging, Reliable Messaging, Multicasting, Publish / Subscribe, Queues, Message Brokers, Federation, Discovery, and Delegation of Trust patterns are not described according to the strict definition of pattern documentation.

Reinfurt also provides patterns for solving communication problems (Reinfurt et al., 2019). Device Gateway, Device Shadow, Rules Engine, Device Wakeup Trigger, Remote Lock and Wipe, Delta Update, Remote Device Management, Visible Light Communication. The Device Gateway patterns is used when part of the devices does not support communication technology or protocol on the network. The pattern involves using an intermediary, such as a Gateway, to translate protocols from and to the device. The Device Shadow pattern is used to send requests to devices that are not permanently connected to the network. The pattern involves creating a virtual address to the device and usually interacts directly with the patterns in the other next group. The Rules Engine pattern allows users to customize the system through a set of simple rules. The Device Wakeup Trigger pattern involves sending a message over a communication channel to a device control unit that is not permanently connected to the network. The Remote Lock and Wipe pattern guarantees the security of the data stored on the device, allowing its data to be erased when the device is unlocked from the system. The Delta Update pattern helps reduce network traffic by sending only device data that has changed since the last time you sent it. The Remote Device

Management pattern allows local management of the device through a client installed on the device that interprets the server commands. The Visible Light Communication pattern involves sending communication signals to or from a remote device.

The last group of design patterns are energy saving patterns which is gaining popularity. Reinfurt (Reinfurt et al., 2017) also offers patterns for this category, dividing them into two types. The former describe the different types of devices by their energy requirements: Mains-Powered Device, Period Energy-Limited Device, Lifetime Energy-Limited Device, Energy-Harvesting Device. Therefore they are not interested in the software of the IoT system. The second type describes how to communicate with devices so that they save energy: Always-On Device and Normally-Sleeping Device. The Always-On Device pattern refers to cases where energy saving is inefficient. The other Normally-Sleeping Device pattern is used in cases where the device does not need to operate continuously and is implemented by deactivating all its energy elements. When the functionality of a such device is required, the control unit supplies power and awakens the device. Even if the control unit is not permanently connected to the network, it may be switched on at regular intervals to check for the need to wake up the device. The patterns that help communicate with similar devices from the reviewed communication patterns are Device Shadow, Device Wakeup Trigger.

The Architectural patterns determine how the elements of the system are organized and interacted. They introduce the basic characteristics and behaviour of the system, the principles and the multilayered structure into the system architecture. The IoT architectural patterns suggest building a single system in a heterogeneous environment. Therefore, components are introduced into the system when is necessary to allow the creation of different types of integration and communication of its components. These include Device-to-Device (D2D), Networking-to-Networking (N2N), Middleware-to-Middleware (MW2MW), Application & Services-to-Application & Services (AS2AS), Data & Semantics-to-Data & Semantics (DS2DS), and protection on different architecture levels that incorporates various mechanisms from authentication to authentication, use of authentication tags, or secure socket layer (SSL). The inclusion of one or the other architectural elements is also influenced by

the purpose of the system, since the systems are heterogeneous. For example, Smart Home and smart supply chain systems have their own specifics. But IEEE (Weyrich and Ebert, 2016) is trying to standardize the fuzzy concepts of definition, architecture and infrastructure. IEEE offers a three-tier architecture with Sensor layer, Networking and data communication layer and Application layer. Referring to this architecture, an IoT architecture pattern (Alreshidi and Ahmad, 2019, p.4) is defined, which implies a system with functions divided into three layers: User interfacing layer, which allows users to monitor and control devices remotely; Interconnection and logic layer, which ensures the coordination and collaboration of devices across the network; Computation and data storage layer, which manages the intelligent analysis and data storage.

CONCLUSION

The IoT systems improve application functionality, increase access to resources, save energy and reduce costs. But their development is attended by many problems. The report presents the IoT system main problems and the abilities for solving them by patterns. The aim is for developers to be easier to find the right patterns when using a top-down development approach. The developers are specialized in specific domains and issues, so the pattern classification by the criterion Aim, enables them to focus on known issue.

REFERENCES

1. ALRESHIDI, A. and AHMAD, A. (2019) *Architecting Software for the Internet of Thing Based Systems*. MDPI, Basel, Switzerland Available from: <https://www.mdpi.com/1999-5903/11/7/153/pdf> [Accessed 12/08/2019].
2. ARMIYANOVA, M. (2018) *Patterns Classification Criteria and Scheme*. Eastern Academic Journal, Bourgas, pp.70-89.
3. BLOOM, G., ALSULAMI, B., NWAFOR, E., AND BERTOLOTTI, I.G., (2018) *Design Patterns for the Industrial Internet of Things*. 14th IEEE International Workshop on Factory Communication Systems (WFCS). Available from: <https://ieeexplore.ieee.org/document/8402353> [Accessed 12/08/2019].

4. CHANDRA, G., MAHINDRA T. (2016) *Pattern language for IoT applications*. Conference on Pattern Languages of Programs (CPLoP). Available from: <https://hillside.net/plop/2016/papers/writinggroup/17.2.pdf> [Accessed 12/08/2019].
5. DARA, S. (2014) *Privacy Patterns in Public Clouds*. Proceedings of the Indian Conference on Pattern Languages of Programs (Guru-PLoP).
6. FERNANDEZ, E. (2013) *Security Patterns in Practice: Designing Secure Architectures Using Software Patterns*. Wiley.
7. FERNANDEZ, E. B. YOSHIOKA, N. and WASHIZAKI, H. (2014) *Patterns for Cloud Firewalls*. Proceedings of the Asian Conference on Pattern Languages of Programs (AsianPLoP).
8. GUTH, J., BREITENBÜCHER, U., FALKENTHAL, M., LEY-MANN, F. and REINFURT, L. (2016) *Comparison of IoT Platform Architectures: A Field Study based on a Reference Architecture*. In Proceedings of the International Conference on Cloudification of the Internet of Things (CIoT). IEEE. Available from: <https://www.iaas.uni-stuttgart.de/publications/INPROC-2016-48-Comparison-of-IoT-Platform-Architectures.pdf> [Accessed 12/08/2019].
9. HAFIZ, M. (2013) *Security Pattern Catalog*. Available from: <http://www.munawarhafiz.com/securitypatterncatalog/index.php> [Accessed 12/08/2019].
10. INDUSTRIAL INTERNET CONSORTIUM (IIC) (2015) Available from: <https://www.iiconsortium.org/pdf/IIC-Overview-11-24-15.pdf> [Accessed 12/08/2019].
11. KARDELL, M. (1997) *A Classification of Object-oriented Design Patterns. Master's thesis, Department of Computing Science*. Umeå University. Available from: <http://www8.cs.umu.se/~jubo/ExJobbs/MK/patterns.htm> [Accessed 12/08/2019]
12. KIENZLE, D., ELDER, M., TYREE, D. and EDWARDS-HEWITT, J. (2002) *Security Patterns Repository Version 1.0*. Available from: <http://www.scrypt.net/~celer/securitypatterns/repository.pdf> [Accessed 12/08/2019]

13. KOSTER, M., (2014) *Design Patterns for an Internet of Things*. Available from: <http://community.arm.com/groups/internet-of-things/blog/2014/05/27/design-patterns-for-an-internet-of-things> Available from: <http://iot-datamodels.blogspot.com/2014/05/design-patterns-for-internet-of-things.html> [Accessed 12/08/2019]
14. KOPP, O., Falkenthal, M., Hartmann, N., Leymann, F., Schwarz, H., Thomsen, J. (2015) *Towards a Cloud-based Platform Architecture for a Decentralized Market Agent*. INFORMATIK 2015, P-246, pp. 69–80. Gesellschaft für Informatik e.V. (GI), Bonn.
15. LE GAL, C., MARTIN, J., LUX, A., CROWLEY, J. (2001) *SmartOffice: Design of an Intelligent Environment*. IEEE Intelligent Systems 16, pp. 60–66.
16. NAM, T., PARDO, T. (2011) *Conceptualizing Smart City with Dimensions of Technology, People, and Institutions*. Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times, pp. 282–291. ACM, NY.
17. PETER W. Communication Patterns for the Internet of Things, June 1, 2016. Available from: <https://software.intel.com/en-us/articles/communication-patterns-for-the-internet-of-things> [Accessed 12/08/2019].
18. QANBARI, S., PEZESHKI, S., RAISI, R., MAHDIZADEH, S., RAHIMZADEH, R., BEHINAEIN, N., MAHMOUDI, F., AY-OUBZADEH, S., FAZLALI, P., ROSHANI, K., YAGHINI, A., AMIRI, M., FARIVARMOHEB, A., ZAMANI, A., and DUST-DAR, S. (2016) *IoT Design Patterns: Computational Constructs to Design, Build and Engineer Edge Applications*. IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI), pp. 277-282.
19. REINFURT, L., BREITENBÜCHER, U., FALKENTHAL, M., LEYMAN, F. and RIEGG, A., (2019) *Internet of Things Patterns for Communication and Management*. Springer-Verlag. Available from: <https://www.iaas.uni-stuttgart.de/publications/ART-2019-08-Internet-of-Things-Patterns-for-Communication-and-Management.pdf> [Accessed 12/08/2019].

20. REINFURT, L., BREITENBÜCHER, U., FALKENTHAL, M., LEYMANN, F. and RIEGG, A., (2017) *Internet of Things Patterns for Devices* Available from: <https://www.iaas.uni-stuttgart.de/publications/INPROC-2017-15-Internet-of-Things-Patterns-for-Devices.pdf> [Accessed 12/08/2019].
21. REINFURT, L., BREITENBÜCHER, U., FALKENTHAL, M., FREMANTLE, P., and LEYMANN, F. (2017) *Internet of Things Security Patterns*. HILLSIDE Proc. of Conf. on Pattern Lang. of Prog. Available from: <https://www.hillside.net/plop/2017/papers/proceedings/papers/20-reinfurt.pdf> [Accessed 12/08/2019].
22. RÖCKER, C. (2010) *Services and Applications for Smart Office Environments - A Survey of State-of-the-Art Usage Scenarios*. International Journal of Social, Behavioral, Educational, Economic, Business and Industrial Engineering 4, pp. 51–67.
23. ROMANOSKY S., (2001) *Security Design Patterns*, Technical report, Available from: [http://www.cgisecurity.com/lib /security DesignPatterns.pdf](http://www.cgisecurity.com/lib/security%20DesignPatterns.pdf) [Accessed 12/08/2019]
24. SCHUMACHER, M., FERNANDEZ, E. B., HYBERTSON, D., BUSCHMANN, F., and SOMMERLAD, P., (2006) *Security Patterns: Integrating Security and Systems Engineering*. John Wiley & Sons.
25. SRD, Statista Research Department (2016) *Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions)*. Available from: <https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide> [Accessed 2/08/2019].
26. SU, K., LI, J., FU, H. (2011) *Smart City and the Applications*. 2011 International Conference on Electronics, Communications and Control (ICECC), pp. 1028– 1031. IEEE, Piscataway, NJ.
27. VILLARREAL, N., FERNANDEZ, E., LARRONDO-PETRIE, M. and HASHIZUME, K. (2013) *A Pattern for Whitelisting Firewalls (WLF)*. PLoP 13. Available from: <http://www.lacpei.org/LACCEI2013-Cancun/ExtendedAbstracts/EA055.pdf> [Accessed 12/08/2019].
28. VOAS, J. (2016). *Networks of 'Things'*. NIST Special Publication 800-183. <http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-183.pdf>

29. WASHIZAKI, H., YOSHIOKA, N., HAZEYAMA, A., KATO, T., KAIYA, H., OGATA, S., OKUBO, T. and FERNANDEZ, E. (2019) *Landscape of IoT Patterns*, SERP4IoT '19 Proceedings of the 1st International Workshop on Software Engineering Research & Practices for the Internet of Things, pp. 57-60.
30. WEYRICH, M., EBERT, C. (2016) *Reference Architectures for the Internet of Things*. IEEE Softw. 33, pp. 112–116.