A Review of Middleware Platforms in Internet of Things: A Non – Functional Requirements Approach

Hafiz Wahab Raza[1], Muhammad Ayoub Kamal[2], Muhammad Alam[3], M.S. Mazliham Su'ud[4]

***Abstract*—Internet of things is like an umbrella that covers all connected things through the internet. The main objective behind all these connected devices is to share data, collect data and information, in the existing ecosystem (people, system, devices, etc.) to perform smartly and help to make human life better, easier, and comfortable. IoT is being used in multiple industries for different purposes such as manufacturing, healthcare, automation, vehicle transportation, etc. which is now called the Industrial Internet of Things (IIoT). The industrial revolution has made IoT very complex, crowded, and complicated. The recent technological developments of IIoT expect to create assorted applications in diverse domains of IIoT without human effort. Middleware is a system designed to be the intermediary between IoT devices and applications. The number of organizations depends on integrated solutions even these solutions are very complex for their requirements due to the successful communication among applications from various vendors. Generally, middleware provides ease in the development process through heterogeneous communications of devices and computing and supports interoperability among assorted services and applications. There has been a number of protocols and middleware in IIoT. This paper presents the review of IoT middleware used for diverse environments with respect to the various non - functional requirements.**

***Keywords*— Functional Requirements, Heterogeneous Devices, IIoT, Internet of Things, Middleware, Non – Functional Requirements.**

I. INTRODUCTION

This is the era of technological enhancement in every aspect of life. Due to the vast and fast development in communication and computing, many objects are being equipped with actuators, memories, sensors, communication modules without inference of human for the successful communication between objects, which forms Internet of Things (IoT).

Hafiz Wahab Raza[1], Muhammad Ayoub Kamal[2], Muhammad Alam[3], M.S. Mazliham Su'ud[4]

Malaysian Institute of Information Technology (MIIT) UniKL[1], Institute of Business and Management (IoBM)[2],
Malaysian France Institute (MFI), UniKL[3]
(email : hafizwahabraza@gmail.com)[1]

A. Internet of things (IoT)

IOT is like umbrella which covers all connected things through internet. The main objective behind all these connected devices is to share data, collect data and information, in the existing system which is ecosystem (people, system, devices, etc.) to perform smartly and help to make human life better, easier, and comfortable [1], [2]. IoT is very complex [3], crowded [4] and complicated field [5]. It covers many types of communication channels, protocols, middleware, architectures, devices and many more [3]. IoT confirms the interconnection, communication, and interoperability among smart devices like laptops, watches, computers, tablets, mobile phones and many other handheld devices. These devices are equipped with sensors/actuators which sense, understand, and then intelligently decide independently the environment or communicate with other devices to make the decision. In general, IoT aims to provide computer-based logic to multiple things which are also known as objects to control or monitor by analytics or engines [6]. Internet of things was firstly used for commercial purpose only but by the time it has evolved for industry as well [7]. Fig.1 illustrated the basic IoT architecture which consist of three layers named as perception layer, transportation/network layer and application layer.

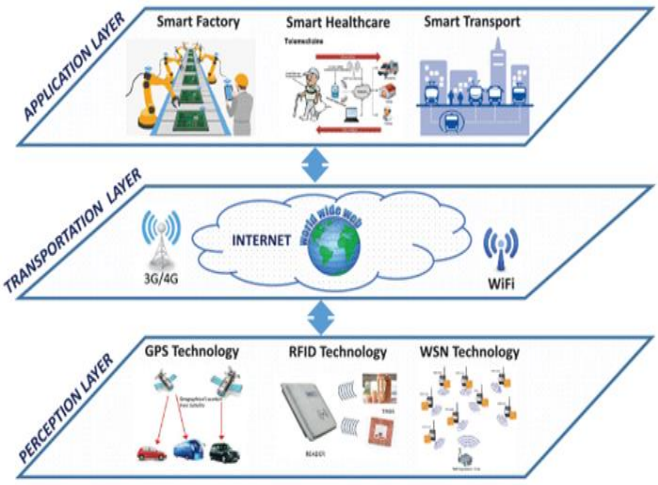

Figure 1. Basic IoT System Model [8]

Industrial Internet of Things (IIoT): As IoT touches many environments like personal electric devices, smart home, agriculture or healthcare, the Industrial Internet of Things (IIoT) confirms the environment of industry [9], [10]. The IoT refers the solutions to transform the role and operation of many industrial existing systems like manufacturing and

transportation system. For example, when IoT is used for the creation of intelligent system for transportation, the authority will keep track of vehicle's movement, its current location and predict the road traffic and its future location [9]. IIoT allows the integration of internet infrastructure, communication protocols and wireless sensor networks with the progression enabling intelligent operations of industry for analysis, management, and monitoring. This internetwork of smart things in the production system helps in industrial operation to improve efficiency, intelligence, and safety [6]. IoT allows the communication of numerous devices among each other. The handling of this huge data generated by these devices poses some challenges for the researchers. A successful communication is in which the collocutors use a common language. There are numbers of heterogeneous devices are used in IoT due to which the interoperability of data from these devices is the main concern. This challenge can be deal by using a standard solution which is very difficult, or the second solution is middleware [11]. Middleware is a software which plays key role and responsible for intelligence in IoT for integration of data from devices, communication among devices and make decisions on the behalf of collected data [12].

B. Middleware

Ubiquitous computing such as IoT produces huge amount of data from heterogeneous sensor-based devices and infrastructure along with applications. The creation of ubiquitous application is challenging task in IoT. Middleware provides interoperability and concatenation of services and applications functioning on different levels of IoT. The services include device identification, authorization, authentication, and security [13]. Middleware is a system designed to be the intermediary between IoT devices and applications [8]. Number of organizations depends on integrated solutions even these solutions are very complex for their requirements due to the successful communication among applications from various vendors[14]. Developers need to use new software specification when integrated software package if not using middleware, which is very difficult also time consuming [15]. There are many open-source and proprietary middleware from technology providers, very similar as compared to features provided. There are four different components or preliminaries of IoT which are: WSN (Wireless Sensor networks), RFID (Radio Frequency Identification), M2M (Machine – to – machine) and SCADA (Supervisory Control and Data Acquisition) as shown in Fig. (2). All these four pillars of IoT need IoT middleware for functioning fully [16],[17].

Figure 2. Examples of Device Heterogeneity in IoT [18]

This paper presents the comparative analysis of different IoT middleware used for various purposes in various environments with respect to non – functional requirements of IoT middleware which are (Scalability, Real Time, Availability, Security, Privacy, Ease of Deployment/use/Maintenance, Interoperability, Adoptability & Flexibility and Multiplicity) which is the main contribution of this paper. Rest of the paper consist of different sections which are: Section II consist of the literature review regarding the middleware while comparative analysis is elaborated in section III. Section IV illustrated the requirements of IoT middleware and discuss the analysis of middleware with reference to non – functional requirements and some open issues for researchers and section V concludes the paper.

II. LITERATURE REVIEW

Due to the vast enhancement enabling IoT technologies, it is predicted that our environment such as home, factories, healthcare, automation etc. will be equipped with wide range of IoT devices which mostly be sure of heterogeneous in nature as hardware systems within various networks. IoT has very broad spectrum[19], in which middleware plays a pivotal role of software glue to cover the gap of various systems of IoT by providing a bridge in between them, for helping them to communicate and collaborate each other in a working environment. In fact, IoT middleware should be responsible for providing solution to IoT devices having translation with sensor's measured data and control commands from actuators, available on the internet as resources.

A. IoT Middleware Platforms

nCube having standard based middleware oneM2M platform for heterogeneous devices in IoT having actuation and sensing proficiencies. The nCube permits actuation commands and sensing values to be transformed into standardized oneM2M resources which are accessible through REST APIs in a standardized way. An android application is made for the user to deal with the measured data generated by sensors and actuators available on the internet as resources. The source code will help developers to build application and IoT products in future in a standardized way [20].

Lightweight service IoT mashup middleware is based on the architecture which REST-style for various applications of IoT. This middleware focuses on the implementation and design for OSGi framework based on protocol stack management and uniform devices access. It includes distributed subscribe/publish based message service, and a mashup of IoT services approach which facilitates the integration of various data services, besides this decision-making process mashup which can be integrated for the creation of situational and composite application, to apply REST principle for defining the extensible interface for the building situational and composite applications. This middleware generated good results as expected in the coal mine system [21].

A framework for Semantic Integration of Heterogeneous sEnsor Data (SIGHTED)[22] is presented in this paper to provide and collect heterogeneous data of sensors from multiple sources and based on the linked data principles and semantic web. On the behalf of SIGHTED framework, DotThing platform is introduced which is based on layer structure to publish and consume sensor data by the programs. A python program having SPARQL library is created which expended DotThing dynamically by defining the query set and then system's time response is measured for monitoring the performance of the system. The results stat that the queries which uses small size of data provide results quickly while queries suing large amount of data upto 1Mb take too much time which reduces the efficiency of the system. So, scalability should be enhanced as it shows undesirable restriction on the stored data having large size and complex queries.

IoTOne [23] aims to provide support for IoT heterogeneous devices, allow strong control of all the connected devices without over-privileging, and allow secure and strong communication among devices in the system. IoTOne solution also provide third party applications for the user to connect their smart devices with IoTOne system. Different vender's devices can be hosted by IoTOne solution to overcome the limited compatibility issues. It is user friendly because it provides many IoT services and configure database and openHab server. OpenHab server comes with installed HABmin2 which is professional, portable, and modern interface to provide both administrative and user functionalities. This platform also supports third party applications but, in this case, security is the main concern. To overcome this main flaw endpoint code is implemented by the developer of the Smart things which ensures that the server code uses secure principles of programming.

IoT based Semantic Interoperability Model (IoT-SIM)[24] which aims to monitor and tracking of human diseases as per the doctors' prescribed medicine in healthcare domain. Cloud Services, Semantic Interoperability and User Interface are the main components of the proposed model. For the collection of data from the sensor enable IoT devices, data analytic technique is applied. All sensor devices have API network which is used to filter data then this filtered data is forwarded to web service of the web by which these sensors communicate with the world. After the identification of the disease by Lightweight model, system suggests medicine itself. If the prescribe medicine matches with the identified medicine, then it is correct otherwise the prescribe medicine is wrong. Storage intelligent health cloud stores right and wrong medicine with doctor and patient's identification. Classified diseases of multiple categories regarding the healthcare domain then forwarded to tagging where diseases are automatically annotated semantically or manually by Resource Description Framework to make readable by machine and human. SPARQL query is used to extract all patients' records from RDF. Physicians/doctors can query for the current situation of the patient from the database anytime from IoT devices remotely. End users do not have concern about distance, time, and hardware.

There are three levels on the gateway which are: (1) integration (2) Virtual Sensor and (3) Semantic annotation. The existing gateway gathers, filter and submits data bundles over multiple protocols from multiple sensors. The collected data of sensors converted to semantic data on the behalf of predefined criteria because sensor data is very hard to work with each other due to the lack of semantics as semantic data always work as the data is accurate to the real world. Further it connects them to the domain of the plant growth and stores the semantic data on the data storage. At the end, all users have interfaces for anticipating data linked with assigned services[25]. To implement the proposed work two use cases were evaluated in agriculture domain. One is for monitoring soil condition and other one is for monitoring environmental pollution. In these use cases three types of user were selected. One user was an IT professional who is familiar with the GSN environment and configuration process. Second user was an IT professional who is not familiar with the existing GSN environment and third user was non-IT person. The results show that all the users who are using the proposed middleware saves time of configuration and still it is very user friendly.

Figure 3. Middleware Implementation Scenario [26]

MSOAH – IoT is a middleware platform based on service-oriented architecture (SOA) of the IoT for the integration of heterogeneous data/information collected from heterogeneous sources[26].

This middleware searches the devices which are connected by different networks and provide its services through interface which is available through web services using REST API. This middle also ensures the heterogeneity by implementation of this middle in three different scenarios. In the first scenario it ensures the data management from heterogeneous IoT devices connected by Bluetooth or Wi – Fi, and in the second scenario the devices are being connected using different technologies. In the last scenario the interfaces were activated by using JAVA technology to offer the connected devices as the web services to deal with the generated data as described in Fig. (3). "the implementation of middleware in an IoT environment".

Syntactical interoperability is a type of interoperability in which all the devices in IoT network should be connected through IoT protocols. This middleware deals with this kind of problem. It provides interoperability through multi-protocol using WebSocket, MQTT and CoAP. It is created by event-based architecture using publish/subscribe pattern[27]. The middleware was tested on the behalf of success rate and delay time in responding data. This system consists of humidity and temperature sensor which are used with MQTT and CoAP as publisher and application using WebSocket as subscriber. The success rate is above 90% of the data transmission and delay time is less than 1 second while data loss ratio is 1% to 25%.

Centralized action-based framework[28] allows multiple users to control smart home appliances via common platform. In this framework controller is the main entity which acts as middleware between users and heterogeneous devices of smart home appliances. Middleware accepts user commands via API and convert it into instructions as per devices. Authentic user will get access to the appliances of the smart home while maintaining the integrity of the system. The results illustrated the performance of the framework is better when applying on local vicinity.

Abstraction middleware[29] allows the communication among heterogeneous devices connected in a smart home network using high level interface. The proposed middleware presents the suitable results permitting the use of interface through applications for the communication with the heterogeneous devices, besides this it shows the response time which is approximately 30 milliseconds for the request of the communication between devices.

PICO (Platform Independent Communications) middleware[30] facilitates the secure communication between devices and data storage in smart grid environment. It uses web services based on REST. The data interface makes it available and useable for all kinds of devices and operating systems. Middleware is the responsible for the real-time all kinds of security and operations. PICO proved the feasibility of security, scalability related to memory, throughput, and latency.

MsM (Microservice Middleware)[31] is the integration of WSNs and IoT. It tackles heterogeneity, scalability, and many other features. The architecture of MsM is adopted from ANN (Artificial Neural Network) which allows the connection between components of system and microservices. It also manages the existing and new services without concerning their technical details. Furthermore, it detects load balancing of the services, system bugs and data faults. MsM is compared with other middleware and result showed better performance of the Microservice Middleware.

III. Analysis of IoT Middleware Platforms

Comparative analysis of different middleware described in various environments in the following table 1 with respect to non – functional requirements of IoT middleware.

Table 1
Comparative Analysis of Middleware

Requirements → / Middleware ↓	Scalability	Real Time	Availability	Security	Privacy	Ease of Deployment / Use / Maintenance	Interoperability	Spontaneous Interaction	Multiplicity	Adaptability & Flexibility
nCube [20]	✓	✓		✓	×	✓	✓	✓	×	×

Lightweight Service Mashup [21]	✓	✓	✓			✓	✓	✓	✓	✓
SIGHTED [22]	×	✓	✓	✓	✓	✓	✓			✓
IoTOne [23]	×	✓		✓	✓	✓	✓	✓		✓
IoT-SIM [24]		✓			✓	✓	✓	✓		✓
Middleware using Semantic Web Techniques [25]	✓	✓		✓	✓	✓	✓	✓		✓
MSOAH – IoT [26]	✓	✓		✓	✓	✓	✓	✓	✓	✓
Event-based Middleware [27]		✓	✓			✓	✓		✓	✓
Centralized action-based framework [28]	✓	✓		✓			✓	✓		✓
Abstraction Middleware [29]	✓	✓	✓			✓	✓	✓		✓
PICO Middleware [30]	✓	✓		✓	✓	✓	✓		✓	✓
MsM [31]	✓	✓		✓			✓	✓		✓

The above table contains three different types of information where ✓ means the requirement exist in the respective middleware while × shows that it does not exist, and blank cell describes no information regarding specific requirement in the respective IoT middleware. In this analysis, it has been observed that many of the non – functional requirements such as interoperability, ease of development / use / maintenance, real – time, spontaneous interaction and adoptability & flexibility are used in almost every middleware but some to the requirements still need researcher's attention to provide IoT solution regarding the future need which are scalability, availability, multiplicity, security, and privacy. Although some work has done to deal with security and privacy of the middleware for various IoT solutions but still it needs attention for future work in the field of IoT middleware.

IV. Discussion and Open Issues

Middleware is the abstraction of complexities of hardware or any system in IoT which enables the application developers to concentrate to find out or solved the problem [32]. The requirements of the middleware are categorized as functional and non-functional requirements.

A. *Functional Requirements*

Functional requirements are consisting of (i) Data Management, (ii) Resource Discovery, (iii) Event Management and (iv) Resource Managements[33].

(i). Data Management: It deals with the generated data from sensors and actuators, any network data, infrastructure data or information of interest to the application because data plays a role of key in IoT services/applications [32]. Data management means data storage, data acquisition and data processing in the middleware.

(ii). Resource Discovery: In IoT, resources include device energy and power, heterogeneity of hardware (sensors, actuators, RFID, smartphone etc.), communication module, A/D (Analogue to Digital), network level or infrastructural information (protocol and topology of network) and all these services assorted by the devices [34]. As human intervention for resource discovery is infeasible, it is very important requirement that it must be automated. In the absence of any infrastructure network, every device must announce its existence and the services it provides.

(iii). Event Management: There are huge amounts of events occur in the domain of IoT, so it is the responsibility of middleware to tackle all these events [35]. Event management simply altered the generated events into informative events. It should analyze the real time data having high velocity so that the applications must provide real time, accurate information, and intelligence.

(iv). Resource Management: It is concerned with the management of resources where QoS (Quality of Services) impact is constrained in the environment like IoT where application services are provided through these resources. It means that resources should be allocated properly, monitored and conflicts occurred should be resolved accordingly to satisfy the needs of applications [36], [32], [33], [35].

Non – Functional Requirements: Non – functional requirements are scalability, real – time, availability, security, privacy, ease of deployment/use/maintenance, interoperability, spontaneous interaction, adoptability & flexibility, and multiplicity [37].

B. *Scalability*

Scalability concerns with the expansion of devices and their work in the network so that of data as well [38]. A middleware should be able to provide enough QoS to bear the expandability of the network when more objects are added.

C. *Real Time*

Real time deals with the continuity of data and updated data, so it must be updated at the same time. On the other hand, user should not bare the delay of data, so the time between receiving data must be minimum.

D. *Availability*

Availability ensures the platform to be available all the time for executing task to be operational while experiencing some kinds of failure as well [38]. Fault tolerance is ensured when availability and reliability work together.

E. *Security*

One of the main aspects in all the application to deal with the security. In IoT, it is more crucial due to the heterogeneity issue. Due to which a compromised object may bear some sort of attacks, reveal user's sensitive information like location, live video or regular schedule [37]. The consequences of these types of information are limitless. So, the middleware must adopt some standardized way to ensure the user's data security and offer some mechanism for intrusion detection.

F. Privacy

IoT nodes connected in IoT network generate huge amount of raw data which may be beneficial to the third party to collect information about the node holder. A temperature reading can exploit the person's absence or presence in a room by temperature sensor in smart environment. Data must be private by the platform so that only related stakeholder could access the data with compromising the privacy. Privacy deals with the discloser of the user data only to the authentic and authorized user in the network [39].

G. Ease of Deployment/use/Maintenance

The platforms are being used and tacked by users, who may not be technically sound. These platforms or solutions must be user friendly so that average person could be able to adopt, install, maintain, or use it easily [35]. Applications which are easy to use and maintain are preferable by the people and usability without compromising security is one of the major concerns in the success of IoT solution.

H. Interoperability

IoT solutions should be harmonious with other applications and devices with slightly change by the developers. If platform supports various devices, then automatically it will become popular, and scalability also enhanced. Interoperability also enhanced when it supports other IoT protocols such as MQTT and CoAP besides HTTP(S). It should also enhance the APIs for the developers to create more heterogeneous applications for the users without sharing the code of the software/application [36], [34], [35].

I. Spontaneous Interaction

In IoT network more devices added time to time and sometime reposition. These are situational changes mostly occur in remote network. So, the IoT solution should be able to interact and connect devices any time without human intervention or minimum human interaction [34].

J. *Multiplicity*

Various IoT devices connect in a network and communicate each other and even provide same services. So, the platform should provide and decide which one is the best service among all. When any device is selected as the best service provider then it must be smarter player than other devices [37]. Sometimes, better devices did not provide best services due to the following issues which are memory limit as sometime huge number of requests are needs to process, and sometime its distance which are the issues related to multiplicity.

K. Adoptability & Flexibility

IoT solution must be flexible to be altered any time for short period of time and it must be able to adopt those changes which can help to run and work for a long time. It must be worthwhile to work in various scenarios [35], [37], [39].

The terms middleware, IoT middleware, IoT Platform and IoT middleware framework are used interchangeably. The major enabling technology in IoT is middleware [11].

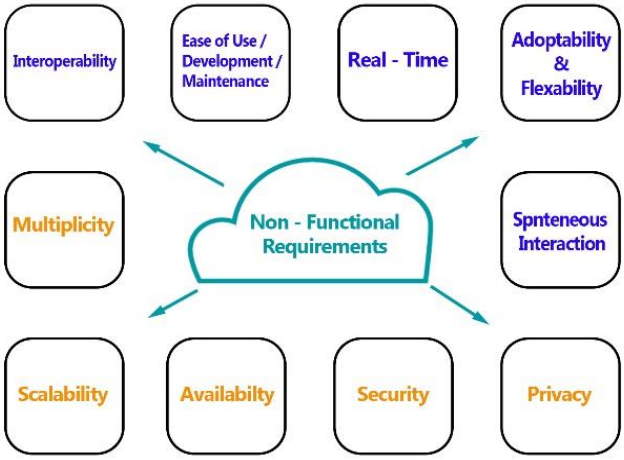

Figure 4. Middleware Implementation Scenario

This paper presents comparative analysis specifically in terms of non-functional requirements and in middleware requirements. There are some specific requirements where researchers need to work which are in terms of security, privacy, multiplicity, scalability, availability as many IoT devices are being connect in IoT network and make it wide so these are issues which need researcher's attention in future. In fig. 4 non – functional requirements are displayed where the requirements in orange color need researchers' attention and motivation for upcoming research.

Open Issues: There are significant opportunities and research challenges which need researcher's and industry's attention in almost every walk of life. The software which is chosen as IoT solution is a long-term and specific deed or commitment of any organization, specifically in IoT. There are hundreds of Middleware available to choose as IoT solution as per the required criteria for any organization, if not then it needs to be. There must be an objective to compare middleware, it gives clear understanding for the readers to choose which one is best for their environment and fulfill their requirements, but sometimes this approach is quite confusing for the readers in terms of theory only. Readers consider that

most of the middleware are same and providing the solution for them.

V. CONCLUSION

IoT is a complex and crowded network for favorable scenario where most of the devices are limited to resources, Which emphasis on the intelligence of the entity which is more efficient to deliver. This entity is software or application which is mostly recognized as IoT Middleware or sometime middleware platform or IoT solution and often it is identified as IoT platform even though it is not single platform. The critical decision is the selection of IoT solution for the specific scenario which can be much fruitful and even bad if the selection went go wrong because it will run for long period of time. The most important thing about choosing a middleware is which middleware is capable, recognize and accomplish the requirements of an individual or organization which is interested in IoT market. The developers must spend some more time to create the middleware solution more user-friendly keeping security in mind as most critical aspect and requirement, as the usability and quality would be the major concern of this crowded market. This paper presents the comparative analysis of multiple middleware regarding non-functional requirements as these plays vital role in selecting the suitable middleware for a specific environment.

REFERENCES

[1] O. B. Sezer, E. Dogdu, and A. M. Ozbayoglu, "Context-Aware Computing, Learning, and Big Data in Internet of Things: A Survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 1–27, 2018, doi: 10.1109/JIOT.2017.2773600.

[2] M. A. Kamal, M. K. Kamal, and M. Alam, "Context-Aware Perspective Analysis working of RFID Anti-Collision Protocols .," no. 2, pp. 19–32, 2018, doi: 10.31645/jisrc/(2018).16.2.02.

[3] L. Atzori, A. Iera, and G. Morabito, "Understanding the Internet of Things: definition, potentials, and societal role of a fast evolving paradigm," *Ad Hoc Networks*, 2017, doi: 10.1016/j.adhoc.2016.12.004.

[4] A. Čolaković and M. Hadžialić, "Internet of Things (IoT): A review of enabling technologies, challenges, and open research issues," *Comput. Networks*, vol. 144, pp. 17–39, 2018, doi: 10.1016/j.comnet.2018.07.017.

[5] A. Vision, A. Elements, S. Issues, N. Mpstme, and J. Ram, "Internet of Things (IoT)," pp. 492–496, 2017.

[6] Q. Wang, X. Zhu, Y. Ni, L. Gu, and H. Zhu, "Blockchain for the IoT and industrial IoT: A review," *Internet of Things*, 2019, doi: 10.1016/j.iot.2019.100081.

[7] T. M. Fernandez-Carames and P. Fraga-Lamas, "A Review on Human-Centered IoT-Connected Smart Labels for the Industry 4.0," *IEEE Access*, vol. 6, no. c, pp. 25939–25957, 2018, doi: 10.1109/ACCESS.2018.2833501.

[8] M. Frustaci, P. Pace, G. Aloi, and G. Fortino, "Evaluating critical security issues of the IoT world: Present and future challenges," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2483–2495, 2018, doi: 10.1109/JIOT.2017.2767291.

[9] E. M. Rúbio, R. P. Dionísio, and P. M. B. Torres, "Industrial IoT devices and cyber-physical production systems: Review and use case," *Lect. Notes Electr. Eng.*, vol. 505, pp. 292–298, 2019, doi: 10.1007/978-3-319-91334-6_40.

[10] S. Li, L. Da Xu, and S. Zhao, "5G Internet of Things: A survey," *J. Ind. Inf. Integr.*, vol. 10, pp. 1–9, 2018, doi: 10.1016/j.jii.2018.01.005.

[11] A. Farahzadi, P. Shams, J. Rezazadeh, and R. Farahbakhsh, "Middleware technologies for cloud of things: a survey," *Digit. Commun. Networks*, 2018, doi: 10.1016/j.dcan.2017.04.005.

[12] A. H. Ngu, M. Gutierrez, V. Metsis, S. Nepal, and Q. Z. Sheng, "IoT Middleware: A Survey on Issues and Enabling Technologies," *IEEE Internet Things J.*, 2017, doi: 10.1109/JIOT.2016.2615180.

[13] Y. Justin Dhas and P. Jeyanthi, "A review on internet of things protocol and service oriented middleware," in *Proceedings of the 2019 IEEE International Conference on Communication and Signal Processing, ICCSP 2019*, 2019, doi: 10.1109/ICCSP.2019.8698088.

[14] M. Alaa, A. A. Zaidan, B. B. Zaidan, M. Talal, and M. L. M. Kiah, "A review of smart home applications based on Internet of Things," *Journal of Network and Computer Applications*. 2017, doi: 10.1016/j.jnca.2017.08.017.

[15] M. A. A. da Cruz, J. J. P. C. Rodrigues, A. K. Sangaiah, J. Al-Muhtadi, and V. Korotaev, "Performance evaluation of IoT middleware," *J. Netw. Comput. Appl.*, 2018, doi: 10.1016/j.jnca.2018.02.013.

[16] L. Wang and R. Ranjan, "Processing distributed internet of things data in clouds," *IEEE Cloud Comput.*, 2015, doi: 10.1109/MCC.2015.14.

[17] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *IEEE Commun. Surv. Tutorials*, 2014, doi: 10.1109/SURV.2013.042313.00197.

[18] G. Fersi, "Middleware for internet of things: A study," in *Proceedings - IEEE International Conference on Distributed Computing in Sensor Systems, DCOSS 2015*, 2015, doi: 10.1109/DCOSS.2015.43.

[19] S. A. Chelloug and M. A. El-Zawawy, "Middleware for internet of things: Survey and challenges," *Intell. Autom. Soft Comput.*, 2018, doi: 10.1080/10798587.2017.1290328.

[20] J. Yun, I. Y. Ahn, J. Song, and J. Kim, "Implementation of sensing and actuation capabilities for IoT devices using oneM2M platforms," *Sensors (Switzerland)*, 2019, doi: 10.3390/s19204567.

[21] B. Cheng, S. Zhao, J. Qian, Z. Zhai, and J. Chen, "Lightweight Service Mashup Middleware with REST Style Architecture for IoT Applications," *IEEE Trans. Netw. Serv. Manag.*, 2018, doi: 10.1109/TNSM.2018.2827933.

[22] A. M. Nagib and H. S. Hamza, "SIGHTED: A Framework for Semantic Integration of Heterogeneous Sensor Data on the Internet of Things," in *Procedia Computer Science*, 2016, doi: 10.1016/j.procs.2016.04.251.

[23] N. Gyory and M. Chuah, “IoTOne: Integrated platform for heterogeneous IoT devices,” in *2017 International Conference on Computing, Networking and Communications, ICNC 2017*, 2017, doi: 10.1109/ICCNC.2017.7876230.

[24] F. Ullah, M. A. Habib, M. Farhan, S. Khalid, M. Y. Durrani, and S. Jabbar, “Semantic interoperability for big-data in heterogeneous IoT infrastructure for healthcare,” *Sustain. Cities Soc.*, 2017, doi: 10.1016/j.scs.2017.06.010.

[25] N. M. Htaik, N. A. M. Maung, and W. Zaw, “Enhanced IoT-based interoperable and configurable middleware using semantic web techniques,” in *ECTI-CON 2018 - 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, 2019, doi: 10.1109/ECTICon.2018.8620032.

[26] Y. Mesmoudi, M. Lamnaour, Y. El Khamlichi, A. Tahiri, A. Touhafi, and A. Braeken, “A Middleware based on Service Oriented Architecture for Heterogeneity Issues within the Internet of Things (MSOAH-IoT),” *J. King Saud Univ. - Comput. Inf. Sci.*, 2020, doi: 10.1016/j.jksuci.2018.11.011.

[27] E. S. Pramukantoro and H. Anwari, “An event-based middleware for syntactical interoperability in internet of things,” *Int. J. Electr. Comput. Eng.*, 2018, doi: 10.11591/ijece.v8i5.pp3784-3792.

[28] A. Banerjee, F. Sufyanf, M. S. Nayel, and S. Sagar, “Centralized framework for controlling heterogeneous appliances in a smart home environment,” in *2018 International Conference on Information and Computer Technologies, ICICT 2018*, 2018, doi: 10.1109/INFOCT.2018.8356844.

[29] E. J. Mendes, M. M. Silveira, M. B. Araujo, J. Celestino, and R. L. Gomes, “Abstraction of Heterogeneous IoT Devices for Management of Smart Homes,” *Proc. - 2019 IEEE Latin-American Conf. Commun. LATINCOM 2019*, 2019, doi: 10.1109/LATINCOM48065.2019.8937985.

[30] J. Chen, E. Cañete, D. Garrido, M. Díaz, and K. Piotrowski, “PICO: A platform independent communications middleware for heterogeneous devices in smart grids,” *Comput. Stand. Interfaces*, 2019, doi: 10.1016/j.csi.2019.01.005.

[31] A. Benayache, A. Bilami, S. Barkat, P. Lorenz, and H. Taleb, “MsM: A microservice middleware for smart WSN-based IoT application,” *J. Netw. Comput. Appl.*, 2019, doi: 10.1016/j.jnca.2019.06.015.

[32] S. Shapsough and I. Zualkernan, “Requirements for an IoT Middleware for Utility-Scale Distributed Solar Farms,” in *2019 6th International Conference on Internet of Things: Systems, Management and Security, IOTSMS 2019*, 2019, doi: 10.1109/IOTSMS48152.2019.8939251.

[33] A. Palade, C. Cabrera, F. Li, G. White, M. A. Razzaque, and S. Clarke, “Middleware for internet of things: an evaluation in a small-scale IoT environment,” *J. Reliab. Intell. Environ.*, 2018, doi: 10.1007/s40860-018-0055-4.

[34] S. Jeon and I. Jung, “MinT: Middleware for cooperative interaction of things,” *Sensors (Switzerland)*, 2017, doi: 10.3390/s17061452.

[35] M. A. A. Da Cruz, J. J. P. C. Rodrigues, J. Al-Muhtadi, V. V. Korotaev, and V. H. C. De Albuquerque, “A Reference Model for Internet of Things Middleware,” *IEEE Internet of Things Journal*. 2018, doi: 10.1109/JIOT.2018.2796561.

[36] M. A. Razzaque, M. Milojevic-Jevric, A. Palade, and S. Cla, “Middleware for internet of things: A survey,” *IEEE Internet Things J.*, vol. 3, no. 1, pp. 70–95, 2016, doi: 10.1109/JIOT.2015.2498900.

[37] V. Bastidas, M. Helfert, and M. Bezbradica, “A Requirements Framework for the Design of Smart City Reference Architectures,” in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018, doi: 10.24251/hicss.2018.317.

[38] M. Vierhauser, J. Cleland-Huang, J. Burge, and P. Grunbacher, “The interplay of design and runtime traceability for non-functional requirements,” in *Proceedings - 2019 IEEE/ACM 10th International Workshop on Software and Systems Traceability, SST 2019*, 2019, doi: 10.1109/SST.2019.00010.

[39] L. Calderoni, “Preserving context security in AWS IoT Core,” *ACM Int. Conf. Proceeding Ser.*, 2019, doi: 10.1145/3339252.3340499.