State-of-the-Art: JavaScript Language for Internet of Things

Fernando L. Oliveira, Júlio C. B. Mattos
Computer Science Graduate Program – PPGC
Technological Development Center - CDTec
Federal University of Pelotas – UFPel
Pelotas, Brazil
{fernando.oliveira, julius}@inf.ufpel.edu.br

***Abstract*—Currently, two technologies stand out in the field of research, the Internet of Things (IoT) and JavaScript (JS). On one hand, IoT enables everyday objects to connect to the network, analyze, capture, and interact with the environment. On the other hand, there is JavaScript, a programming language that was initially inside browsers but nowadays it is being used in several fields. This paper correlates IoT and JS technologies, showing how JavaScript can be applied in the context of the Internet of Things. This survey has considered the work of the last ten years and presents state of the art from JavaScript applied to the IoT-side.**

***Index Terms*—Internet of Things, IoT Platform, Smart Object, JavaScript, Survey**

I. Introduction

The Internet of Things (IoT) is a paradigm in which everyday objects can connect to the Internet and through sensors and actuators interact with the environment [1]. In the past few years, the IoT landscape has grown tremendously, and applications can be found in many domains such as smart home, smart city, agriculture, industry, and others. According to the Gartner, there will be approximately twenty billion IoT devices present by the year 2020 [2].

The growth of IoT devices is daily. However, the context of smart objects is wide and heterogeneous. These features promote several challenges and opportunities to introduce new technologies and approaches to improve performance, resource consumption, security, and communication. In this sense, JavaScript[1](JS) appears as a viable and promising alternative, due to its flexibility and large developer community.

The JavaScript is a highly dynamic scripting programming language that is becoming one of the essential programming languages in the world [3]. Recent surveys by Stack Overflow [4] show JS topping the rankings of programming, scripting, and markup languages.

JavaScript is present and diverse contexts, including in the Internet of Things ecosystem. IoT devices can be understood as an underlying platform, in which we have an operational system or kernel that performs a JS interpreter, supporting interactions with hardware like CPU, sensors, memory storage, disk storage, servo motors, camera among others [5].

[1]JavaScript is also known as ECMAScript; It is the name used for the language in its specification

This paper shows a study to identify how JS can be used in IoT devices. The focus is on identify the State-of-the-Art that use the JavaScript language on the IoT-side.

The outline of this paper is as follows. The introduction presents the essential concepts about IoT and JavaScript in sections II and III, respectively. Section IV presents the methodological process of the research. Section V shows the results, and finally, section VI presents the conclusions.

II. Internet of Things

According to [6] the term Internet of Things (IoT) initially understood the connection of all physical objects to the Internet, which would have the capacity to capture information through radio frequency identification (RFID) and sensing technologies, which would allow them to observe, understand and interact with the world regardless of people.

From a conceptual point of view, the smart object is based on three pillars, related to the ability of intelligent objects to (i) be identifiable, (ii) communicate, and (iii) interact, with each other, building networks of interconnected objects, or with end-users [7].

For [1], consider IoT with a novel paradigm in which the "objects that surround us will be on the network in one form or another". The objects or "things" can be any devices, such as lamps, sensors, televisions, cell phones, and others. These obtainable devices identified and connected to the Internet to exchange information and make decisions (without the aid of human intervention) to achieve common goals.

In a similar view, [8] states that IoT is characterized by giving to ordinary objects the ability to connect to the network, making them smarts and ability to capture, compute, store, send, and display information. Thus, objects become able to interact with the environment, generating a result of data in quantity, variety, and specificity [9].

In the past few years, the Internet of Things has become common in many knowledge areas towards innovation, moving several kinds of research projects around the world. The IoT capability of collecting, recording and correlating data, enables the inference of knowledge, generating business, and learning [10].

In this new paradigm, IoT stands out for its impact on people's daily lives. New habits are created, unique needs arise, and people adapt to hitherto unknown realities. From the private user's point of view, IoT applies to the work and residential fields, for corporate users, notably in industrial automation and manufacturing, logistics, process management, intelligent transportation of people, healthcare among others [11].

III. JavaScript

Script languages are becoming increasingly popular for developing applications in different areas. JavaScript is a highly dynamic scripting language with high popularity among developers [4], [12]. JavaScript is the programming language initially designed for Web. The vast majority of sites uses JS and modern web browsers on desktops, game consoles, tablets, and smartphones incorporate JavaScript interpreters, making it the most ubiquitous programming language in history [13], [14].

From the technical perspective, JavaScript is a high level, lightweight, dynamic, and untyped programming language, that is well-suited to functional-oriented and object-oriented programming paradigm. The language execution model is based on the interpretation of the source code [15], [13].

JavaScript is a language that is rapidly evolving. The reason for its success is its flexibility and the fact that it is the official language for web applications. Moreover, this is part of the set of standards that define the technology of the World Wide Web, which gives it tremendous popularity. [16], [17]. The institution responsible for language specification is ECMA (European Computer Manufactures Association). The ECM-262 group defines standards and regulations for JavaScript.

Over time, the language has gained new frontiers, leaving the standard browser and inserting the server-side, mobile applications, games, robotics, and Internet of Things [16], [18], [13], [3]. All the advancement is possible due to efficient virtual machines (VM) that interpreter JavaScript [19].

A. *JavaScript Environment for IoT*

For the execution of the JavaScript code, a virtual machine is required, which is assigned to interpret the scripts. The most popular interpreters are JavaScriptCore (JSC) engine of WebKit, SpiderMonkey written by Brendan Eich [2] and the V8[3] engine of Google Chrome [20].

Some IoT devices cannot run JavaScript. Most of the equipment is a simple platform like smart lamps, thermostats, remote electricity plugs, and sensors in overall. Such devices do not require a lot of processing and storage capacity; they are designed to be low-end devices; sometimes, they do not even have an operating system [18].

Specific devices are rated Language-Runtime Architectures [21]. In this model, some boards are built to support a particular built-in language runtime or virtual machine; for instance, the Espruino[4] and Tessel 2[5] boards are units that support JavaScript.

The segment of the IoT architecture that enables the execution of JavaScript are those that have an Operating System (OS). Generally, this type of equipment uses a Linux-based OS. It has significantly higher memory and CPU requirements, one good example are the Raspberry Pi family devices [18]. This kind of device enables to execute a JavaScript VM.

IV. Research Methodology

This section presents the research methodology, how the study was performed and shows the guidelines used. In this paper, a technique of Systematic Mapping Review commonly used in software engineering is applied. Overall, the purpose of this approach is to provide researchers with an overview of a research area and help them to identify possible gaps or actuation context [22].

The Systematic Mapping proposes a research process methodologically well established to identify, analyze and interpret the evidence related to a particular set of research, topic or phenomena of interest, in an unbiased way [23], Figure 1 shows the workflow applied.

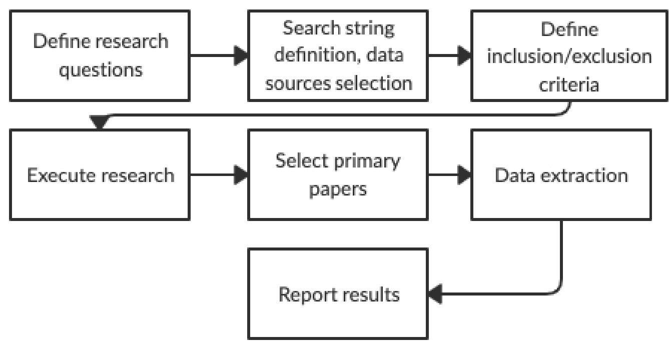

Fig. 1. Research process flow adopted from [23]

Besides, the guidelines proposed in [23], [24], every research process is motivated by a central question, in this case: "*How JavaScript can be used in programming for the Internet of Things*". Thus, to facilitate the conduct of research, the general problem is broken into a smaller set of Research Questions (RQs):

- RQ1: What are the purposes of using JavaScript in the IoT?
  - The intention is to discover the motivations to apply JavaScript on IoT.
- RQ2: What are the most investigated research topics and how have they changed over time?
  - The goal is to group the researchers by areas.
- RQ3: Which JavaScript interpreters (virtual machine) are found? What level of JavaScript API is present?
  - To define which engines are used, and which features are available.

[2]Creator of the JavaScript programming language

[3]V8 is the core of Node.js, the most popular JavaScript server-side execution environment.

[4]https://www.espruino.com

[5]https://tessel.io

- RQ4: What are the most challenges on IoT development for JavaScript?
  - The goal is to identify the limitations that JavaScript can find and opportunities for acting.
- RQ5: What are the tools or plugins for JavaScript in the Internet of Things context?
  - The goal is to identify the resources that are available to support integration among technologies.

From the research questions, a set of keywords is elaborated that will be used in the research process. In order to systematize this step, [24] suggests using the PICO process (Population, Intervention, Comparison, and Outcomes). PICO models use a process for framing a question, Table I shows the keywords generated, moreover Figure 2 presents the basic search string.

TABLE I
PICO MODEL FOR THE FRAMING QUESTION

Criteria	Goal	Words
Population	IoT devices	IoT OR Internet of Things
Intervention	JavaScript Interpreter	JavaScript;ECMAScript Interpreter
Comparison	not applied	
Outcomes	Integration between JavaScript and IoT	JavaScript AND IoT

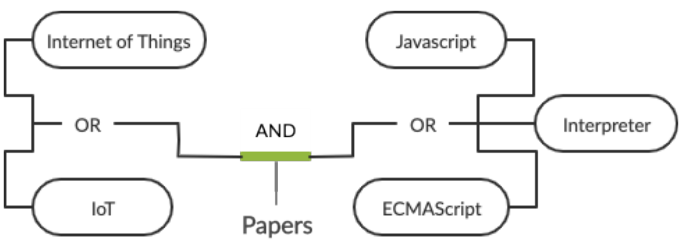

Fig. 2. Basic search string

Regarding the data sources selection, [23], [24] recommends the use of at least four sources, being two digital libraries and two indexing bases. Digital publisher libraries contain articles published by themselves, while indexers index articles from several digital libraries. For this work, we choose the IEEE Explorer[6], Elsevier[7] and ACM Digital Library[8] with libraries, as well as the Scopus[9] and Engineering Village[10] indexers.

The adoption of criteria is fundamental to guarantee quality in the results, establishing characteristics that a study must contain to be considered relevant in the context of the research (inclusion criteria - IC) and characteristics that lead to the exclusion of studies that do not meet the defined criteria (exclusion criteria - EC) [24]. Table II shows the criterions used.

[6]https://ieeexplore.ieee.org
[7]https://www.elsevier.com
[8]https://dl.acm.org
[9]https://www.scopus.com
[10]https://www.engineeringvillage.com

TABLE II
INCLUSION AND EXCLUSION CRITERIA.

Type	Ref	Description
Inclusion	IC1	Papers written in English
	IC2	Peer-reviewed articles in journals and conferences
	IC3	Published from 2009
	IC4	Papers related to JavaScript language applied to IoT
Exclusion	EC1	Duplicated papers
	EC2	Papers not available for downloading
	EC3	Paper lies outside the IoT and JavaScript language domain

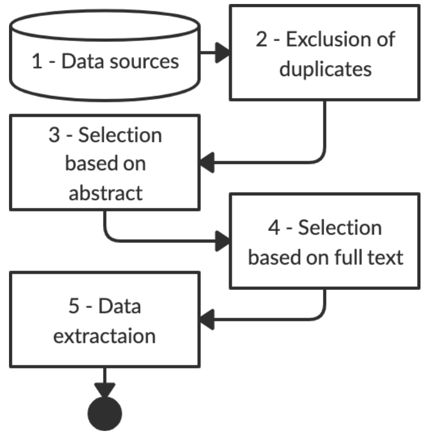

Fig. 3. Selection process

A. Conduct research

From the search string, the investigation was conducted at the chosen data source. The execution process of searching was performed manually using the interfaces provided by each data source. Overall, search engines allow filtering the results. Therefore, it is possible to apply some exclusion criteria in the initial search. However, the search result still needs to be analyzed in a specific selection process. Figure 3 shows the document selection flow.

TABLE III
AMOUNT OF PAPERS BY STEP.

	Result (# of papers)			
Source	Step 1	Step 2	Step 3	Step 4
IEEE Explorer	79	79	9	7
ACM Digital Library	101	101	5	4
Elsevier	7	7	0	0
Scopus	483	391	12	5
Engineering Village	104	83	7	0
Total				**16**

In summary, the process of selecting papers consists of four phases. Step one represents the execution of the research based on the search string; also, in this phase, the search engines of the databases already allow filtering the results applying the criteria of inclusion and exclusion. The second step consists of the discarding of duplicated items, applied to the indexed bases. Thirdly, a review by the title and abstract of the papers is carried out. Finally, in step four, the full text is analyzed. In Figure 3 still presents a fifth step, which refers to the extraction of the data of the articles were selected in the research process. The resulting quantity of each phase is presented in Table III.

V. Results and Discussion

Based on the selected papers (16), an analysis was performed in order to answer each of the research questions. Thus, in the data extraction process, the same article may have been used to answer more than one question. Figure 4 shows the temporal distribution of publications.

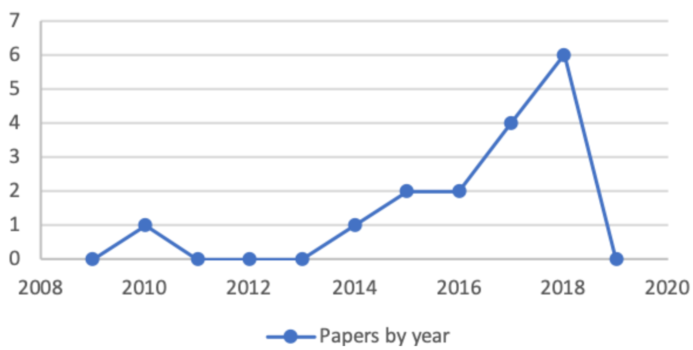

Fig. 4. Amount of papers by year

A. RQ1: What are the purposes of using JavaScript in the IoT?

The use of JavaScript in the context of the Internet of Things can be seen as a naturalness due to its ubiquity, which ensures a robust and active community.

Although it sounds like an empirical analysis, it is possible to verify that by adopting JS as a language in the device-edge, we are in practice, producing an environment that dialogs through a single programming language. Working with the same technology in both segments allows reuse of code, simplifying the logic, and hence, reducing the time and cost of development.

Another perspective of the use of JS and IoT is presented by [25], in which they use the JavaScript language to teach programming through physical objects, contextualizing theories, and practices.

Given the background, there is a constant evolution of the technical aspects from devices, and this performance gain stimulates script execution on the IoT-side. In this bibliographic survey, it was verified that the most significant part of the research focuses on environments (VM) for scripting or migration of states of objects between different equipment, evidencing the attention of researchers in the integration between technologies.

B. RQ2: What are the most investigated research topics and how have they changed over time?

Each article can deal with different topics and provoke discussions on several subjects. In order to classify and group them, it was decided to classify each paper in only one area to facilitate the identification of research trends and opportunities.

TABLE IV
Hot topic on IoT and JS

Topic	Papers
Security	[5]
VM	[26], [19], [27], [28]
Migration	[29], [30]
IoT/JS	[31], [32]
Prototyping	[25]
Interoperability	[16], [33], [34]
Interface	[35], [36], [37]

Each paper is framed in a given topic is:

- Security: Fits in this category papers that address security aspects in the execution of JavaScript.
- VM: Comprises papers that propose some virtual machine or improvements in the execution of JS in the IoT context.
- Migration: Set of the articles dealing with migration and stateful challenges, serialization of information for JavaScript language.
- IoT/JS: Understands papers that generally use both subjects (IoT and JS) without having a specific focus.
- Prototyping: Papers that focus on prototyping IoT solutions considering the scripting language to teach programming.
- Interoperability: Fits in this category papers that address interoperability through JavaScript.
- Interface: We consider the papers that present some application interface as a platform in which JavaScript is used as the basis for the construction of IoT solutions. It may include use case papers.

About changes over time, in the beginning, it was verified that the researches did not focus on JS or IoT as addressed in [32]. Such technologies acted as support. However, the concept of intelligent objects has gained space, and with that emerged challenges as interoperability, Security, resource consumption, and others.

C. RQ3: Which JavaScript interpreters (virtual machine) are found? What level of JavaScript API is present?

This question explores the alternatives to enable JavaScript execution on embedded systems. Table V presents the options identified, as well as their size characteristics and resource coverage according to ECMA-262 specification.

In this research, it was identified a polemic point. Discussing IoT devices, there are devices that are limited in technical terms. With this in mind, the WebletScript interpreter chooses not to implement some language specification definitions [27], so that it gets small binary file. On the one hand, by reducing

TABLE V
JAVASCRIPT ENGINE FOR IOT.

Engine	Size (KB)	JS Version	Compliance (%)
WebletScript	103	5.1	<60
v7	120	5.1	100
JerryScript	160	5.1	100
Duktape	184	5.1[a,b]	99,4
Espruino	231	5.1[b]	<70

Partial/initial support for ECMAScript
[a] version 6
[b] version 7

the size, it is possible to reach a higher number of devices. However, not covering some technical aspect of the specification can be an obstacle to the adoption of the interpreter.

Some features of the language are more commonly used [38], [12], but they are part of the language and should be present. That is a hard decision to make, or if you contemplate a more significant number of devices and limit the capabilities of the language; or if you implement all the features of the specification and reduce the number of compatible devices.

It is expected that the devices evolve in terms of technical capabilities, so the best strategy is to implement as much as possible of the language specification, this way the VM becomes compatible with a more significant number of libraries and consequently wins confidence and credibility in the community.

D. RQ4: What are the most challenges on IoT development for JavaScript?

Overall, IoT has many challenges in several dimensions, such as heterogeneity of hardware, different operational systems, programming languages, APIs, and distinct communication protocols [27]. The lack of a standard implies the manipulation of distributed and complex systems [26].

However, to enable JavaScript on embedded devices, the most evident aspect that represents the biggest barrier today, is the computational limitation of devices. Some devices have common characteristics that limit them; they are created for specific purposes; for example, sensors in general. Also, some devices have as a prerequisite the size, usually connected to the healthcare area. Therefore, any devices will always have a technical limitation to adoption JavaScript as the programming language [18], [39].

One concern JavaScript needs to have is about device resource consumption. Although this is a recurring interest in different languages, in particular, JS is an interpreted language, and this may penalize its performance. So issues such as energy efficiency, bandwidth consumption, and memory should be considered [18], [27], [1].

We identified as an opportunity and tendency, the computational processing at the edge of the network (i.e., edge computing). The evolution of the equipment's processing and storage capabilities allows it to concentrate some of the information processing on the device-edge. In this perspective, instead of sending the data to an external agent like cloud service, the information can be treated locally, innovation the classic IoT cloud-centric model. That approach avoids a network overhead and reduces the response time because the data is processed locally [34], [21]. In this context, the investigations about migrations of object state, corroborate to a distributed processing in the IoT context.

The opportunities for the application of IoT are numerous, by the way, its basic principle is to make everyday objects smart. To this end, objects need to understand end-users, one way to achieve this goal is through monitoring and observing their movements, gestures, location, and context [39]. From these data emerges an opportunity for action, it is possible to apply artificial intelligence techniques to understand the human being, the environment, and interact appropriately [1].

E. RQ5: What are the tools or plugins for JavaScript in the Internet of Things context?

Answering this question, it is presented the technologies found in this bibliographic survey. However, the virtual machines is not addressed.

- SmartJS: JavaScript-based middleware; that provides an environment of execution and development of IoT solutions [26].
- ThingsJS: JavaScript-based middleware; Abstracts several large-scale distributed systems considerations, such as scheduling, monitoring, and self-adaptation, using a rich constraint model, a multi-dimensional resource prediction approach [30].
- Jade: It is a framework that allows a developer to mix C and JavaScript, and the result is a hybrid language to develop IoT applications [33].
- ThingsMigrate: It is middleware to migration of stateful JavaScript applications across IoT devices [30].
- Opel: It is an IoT platform which enables developers to implement several services swiftly and efficiently via a JavaScript [35].
- Smart Block: It is a visual programming environment. Smart Block enable to untechnical users can write their application in this platform quickly [37].

VI. CONCLUSION

Internet of Things and JavaScript can be used together. However, there are several challenges and opportunities to use JavaScript language applied to IoT. There are hardware limitations that may restrict some applications. Furthermore, JavaScript is suitable to build a pretty stable, clean, and maintainable code base using modern frameworks and design patterns practices [31].

Bringing the language to the leading device-edge represents the unification of the programming language, opens up a range of options as tools and libraries, but above all, it simplifies and standardizes the development of applications, implying positively in a more favorable environment for interoperability creating a cross-platform development environment.

Finally, we believe that in the future, we will have highly context-sensitive applications with high levels of understanding and actuation, in which JavaScript will be the official language, given its importance and relevance.

Acknowledgment

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] M. Hung, *Leading the IoT: Gartner Insights on How to Leadin a Connected World*. Gartner, 2018.

[3] S. Delcev and D. Draskovic, "Modern javascript frameworks: A survey study," in *2018 Zooming Innovation in Consumer Technologies Conference (ZINC)*, pp. 106–109, May 2018.

[4] Stack Overflow, "Developer survey results 2019." https://insights.stackoverflow.com/survey/2019#development-environments-and-tools, 2019. Accessed: jun. 2019.

[5] A. Sahu and A. Singh, "Securing iot devices using javascript based sandbox," in *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pp. 1476–1482, IEEE, 2016.

[6] K. Ashton *et al.*, "That 'internet of things' thing," *RFID journal*, vol. 22, no. 7, pp. 97–114, 2009.

[7] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of things: Vision, applications and research challenges," *Ad hoc networks*, vol. 10, no. 7, pp. 1497–1516, 2012.

[8] C. González García, D. Meana Llorián, B. C. Pelayo García-Bustelo, and J. M. Cueva Lovelle, "A review about smart objects, sensors, and actuators," *International Journal of Interactive Multimedia and Artificial Intelligence*, 2017.

[9] A. Whitmore, A. Agarwal, and L. Da Xu, "The internet of things—a survey of topics and trends," *Information Systems Frontiers*, vol. 17, no. 2, pp. 261–274, 2015.

[10] C. Puliafito, E. Mingozzi, F. Longo, A. Puliafito, and O. Rana, "Fog computing for the internet of things: A survey," *ACM Transactions on Internet Technology (TOIT)*, vol. 19, no. 2, p. 18, 2019.

[11] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[12] A. M. Fard and A. Mesbah, "Javascript: The (un) covered parts," in *2017 IEEE International Conference on Software Testing, Verification and Validation (ICST)*, pp. 230–240, IEEE, 2017.

[13] D. Flanagan, *JavaScript: the definitive guide*. " O'Reilly Media, Inc.", 2006.

[14] D. Crockford, *JavaScript: The Good Parts: The Good Parts*. " O'Reilly Media, Inc.", 2008.

[15] T. Mikkonen and A. Taivalsaari, "Using javascript as a real programming language," 2007.

[16] J. Gascon-Samson, M. Rafiuzzaman, and K. Pattabiraman, "Thingsjs: Towards a flexible and self-adaptable middleware for dynamic and heterogeneous iot environments," in *Proceedings of the 4th Workshop on Middleware and Applications for the Internet of Things*, pp. 11–16, ACM, 2017.

[17] A. Saboury, P. Musavi, F. Khomh, and G. Antoniol, "An empirical study of code smells in javascript projects," in *2017 IEEE 24th international conference on software analysis, evolution and reengineering (SANER)*, pp. 294–305, IEEE, 2017.

[18] A. Taivalsaari and T. Mikkonen, "A taxonomy of iot client architectures," *IEEE Software*, vol. 35, pp. 83–88, May 2018.

[19] E. Gavrin, S.-J. Lee, R. Ayrapetyan, and A. Shitov, "Ultra lightweight javascript engine for internet of things," in *Companion Proceedings of the 2015 ACM SIGPLAN International Conference on Systems, Programming, Languages and Applications: Software for Humanity*, pp. 19–20, ACM, 2015.

[20] H. Park, W. Jung, and S.-M. Moon, "Javascript ahead-of-time compilation for embedded web platform," in *2015 13th IEEE Symposium on Embedded Systems For Real-time Multimedia (ESTIMedia)*, pp. 1–9, IEEE, 2015.

[21] A. Taivalsaari and T. Mikkonen, "A roadmap to the programmable world: software challenges in the iot era," *IEEE Software*, vol. 34, no. 1, pp. 72–80, 2017.

[22] C. Wohlin, P. Runeson, P. A. d. M. S. Neto, E. Engström, I. do Carmo Machado, and E. S. De Almeida, "On the reliability of mapping studies in software engineering," *Journal of Systems and Software*, vol. 86, no. 10, pp. 2594–2610, 2013.

[23] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering.," in *Ease*, vol. 8, pp. 68–77, 2008.

[24] B. Kitchenham, R. Pretorius, D. Budgen, O. P. Brereton, M. Turner, M. Niazi, and S. Linkman, "Systematic literature reviews in software engineering–a tertiary study," *Information and software technology*, vol. 52, no. 8, pp. 792–805, 2010.

[25] B. Peterson and B. Vogel, "Prototyping the internet of things with web technologies: Is it easy?," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 518–522, IEEE, 2018.

[26] J. Gascon-Samson, M. Rafiuzzaman, and K. Pattabiraman, "Smartjs: dynamic and self-adaptable runtime middleware for next-generation iot systems," in *Proceedings Companion of the 2017 ACM SIGPLAN International Conference on Systems, Programming, Languages, and Applications: Software for Humanity*, pp. 51–52, ACM, 2017.

[27] D. Li, B. Huang, L. Cui, and Z. Xu, "Webletscript: A lightweight distributed javascript engine for internet of things," in *2018 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, IEEE, 2018.

[28] Y. J. Heo, S. M. Oh, W. S. Chin, and J. W. Jang, "A lightweight platform implementation for internet of things," in *2015 3rd International Conference on Future Internet of Things and Cloud*, pp. 526–531, IEEE, 2015.

[29] J.-w. Kwon and S.-M. Moon, "Work-in-progress: Jsdelta: serializing modified javascript states for state sharing," in *2017 International Conference on Embedded Software (EMSOFT)*, pp. 1–2, IEEE, 2017.

[30] J. Gascon-Samson, K. Jung, S. Goyal, A. Rezaiean-Asel, and K. Pattabiraman, "Thingsmigrate: Platform-independent migration of stateful javascript iot applications," in *32nd European Conference on Object-Oriented Programming (ECOOP 2018)*, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[31] U. Jaimini and M. Dhaniwala, "Javascript empowered internet of things," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 2373–2377, IEEE, 2016.

[32] D. Guinard, V. Trifa, E. Wilde, *et al.*, "A resource oriented architecture for the web of things.," in *IoT*, pp. 1–8, 2010.

[33] D. Ghosh, F. Jin, and M. Maheswaran, "Jade: A unified programming framework for things, web, and cloud," in *2014 International Conference on the Internet of Things (IOT)*, pp. 73–78, IEEE, 2014.

[34] J. Gascon-Samson, K. Jung, and K. Pattabiraman, "Poster: Towards a distributed and self-adaptable cloud-edge middleware," in *2018 IEEE/ACM Symposium on Edge Computing (SEC)*, pp. 338–340, IEEE, 2018.

[35] H. Lee, D. Sin, E. Park, I. Hwang, G. Hong, and D. Shin, "Open software platform for companion iot devices," in *2017 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 394–395, IEEE, 2017.

[36] E. Baccelli, J. Doerr, O. Jallouli, S. Kikuchi, A. Morgenstern, F. A. Padilla, K. Schleiser, and I. Thomas, "Reprogramming low-end iot devices from the cloud," in *2018 3rd Cloudification of the Internet of Things (CIoT)*, pp. 1–6, IEEE, 2018.

[37] N. Bak, B.-M. Chang, and K. Choi, "Smart block: A visual programming environment for smartthings," in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2, pp. 32–37, IEEE, 2018.

[38] S. Gude, M. Hafiz, and A. Wirfs-Brock, "Javascript: The used s," in *2014 IEEE 38th Annual Computer Software and Applications Conference*, pp. 466–475, IEEE, 2014.

[39] M. H. Miraz, M. Ali, P. S. Excell, and R. Picking, "A review on internet of things (iot), internet of everything (ioe) and internet of nano things (iont)," in *2015 Internet Technologies and Applications (ITA)*, pp. 219–224, IEEE, 2015.