



Vaasan yliopisto
UNIVERSITY OF VAASA

OSUVA Open
Science

This is a self-archived – parallel published version of this article in the publication archive of the University of Vaasa. It might differ from the original.

REFIoT: A Framework to Combat Requirements Engineering in IoT Applications and Systems

- Author(s):** Siakas, Errikos; Lampropoulos, Georgios; Rahanu, Harjinder; Georgiadou, Elli; Siakas, Dimitrios; Siakas, Kerstin
- Title:** REFIoT: A Framework to Combat Requirements Engineering in IoT Applications and Systems
- Year:** 2024
- Version:** Accepted manuscript
- Copyright** ©2024 Springer. This is a post-peer-review, pre-copyedit version of a book chapter published in *Systems, Software and Services Process Improvement: 31st European Conference, EuroSPI 2024, Munich, Germany, September 4–6, 2024, Proceedings, Part I*. The final authenticated version is available online at: <https://doi.org/10.1007/978-3-031-71139-8>

Please cite the original version:

Siakas, E., Lampropoulos, G., Rahanu, H., Georgiadou, E., Siakas, D., & Siakas, K. (2024). REFIoT: A Framework to Combat Requirements Engineering in IoT Applications and Systems. In M. Yilmaz, P. Clarke, A. Riel, R. Messnarz, C. Greiner, & T. Peisl (Eds.), *Systems, Software and Services Process Improvement: 31st European Conference, EuroSPI 2024, Munich, Germany, September 4–6, 2024, Proceedings, Part I* (pp. 80-96). Communications in computer and information science, vol. 2179. Springer. https://doi.org/10.1007/978-3-031-71139-8_6

REFIoT: A framework to combat requirements engineering in IoT applications and systems

Errikos Siakas¹, Georgios Lampropoulos², Harjinder Rahanu³, Elli Georgiadou³, Dimitrios Siakas⁴, Kerstin Siakas^{5,6}

¹*National Archaeological Museum, Thessaloniki, Greece, esiakas@culture.gr*

²*University of Macedonia, Thessaloniki, Greece, lamprop@gmail.com*

³*Middlesex University London, London, UK, h.rahanu@mdx.ac.uk, elli.georgiadou@mdx.ac.uk*

⁴*Häme University of Applied Sciences, dimitrios.siakas@gmail.com*

⁵*International Hellenic University, Thessaloniki, Greece, siaka@the.ihu.gr*

⁶*University of Vaasa, Finland, Kerstin.siakas@uwasa.fi*

Abstract

In this research, we investigate the way in which Requirements Engineering is handled in the development of Internet of Things (IoT) applications. Successful and correct requirements engineering improves the quality of the development process and the whole system, as well as of the resulting products and services. As a result of an extensive literature review, and based on the authors' collective longitudinal experience gained through industrial experience and ongoing research using a variety of software development approaches and practices, we identified and categorized the challenges in requirements elicitation and requirements engineering for IoT applications. We encapsulated the categorized requirements engineering challenges into REFIoT, a conceptual Requirements Engineering Framework for IoT, which can be used as a tool by requirements engineers in the development of IoT applications to identify risks so that they can be minimized at an early stage of systems development using the values of the SPI Manifesto we propose mitigation actions.

Keywords: Requirements Engineering, Requirements Management, Requirements Development, Requirements Elicitation, Internet of Things (IOT)

1 Introduction

Internet of Things (IoT) is an innovative and continuously growing technology comprising of a network of physical devices embedded with software and sensors that collect and share data over the Internet without human intervention. IoT is used in numerous applications, functions, and services covering common activities in various markets and industries. IoT enables monitoring and control of the environment through the installation of sensors and actuators that interact with each other and with Internet services [1]. The aim of IoT is to instill intelligence into physical devices and enable them to process data and to gain knowledge. Numerous technologies are combined to allow sensor and actuators to sense and gather valuable information, communicate and collaborate, provide smart data analysis, and make decisions.

IoT is increasingly utilized since its emergence from connecting simple individual devices to complex comprehensive systems transforming, thus, whole industries. IoT is a very promising technology aiming to be used in various domains and use cases and can also enhance quality of life (QoL) of humans [2]. Internet of Things is revolutionising the manufacturing process as a core technology underpinning cyber physical systems in smart factories, aka Industry 4.0 [3]. Hence, it is utilized in various business applications, including automotive industries, agriculture, healthcare, and education.

The benefits of IoT are personal, professional and economic, due to the deployment of these smart objects making intelligent decisions and providing services; hence, human involvement and intervention are decreased and quality of services (QoS), as well as quality of human life, is improved [1]. IoT is regarded as an innovation enabler and facilitator of new initiatives. However, there are challenges that need to be addressed for IoT to be fully exploited by various industries.

Madakam et al. [4] highlighted that IoT facilitates: i) remote monitoring of the current state of any physical object, ii) modification of conditions of the environment iii) data collection for prediction of events, and iv) making

decisions in real time regarding potential intervention. Hence, physical objects convert into digital objects that can be manipulated from anywhere through the Internet [5]. To fully utilize the potential of IoT, well-defined requirements and clear development methodologies are essential for improving the success rate of the development process and the quality of the resulting system. The role of Information and Communication Technologies (ICTs) as an innovation enabler has been enhanced due to the development of IoT [1]. This fact has increased the demands on software engineers to understand and deploy physical objects in a network with the aim of connecting and exchanging data with diverse devices and systems over the internet [1]. However, IoT systems have noteworthy differences with traditional Information Systems. In addition to software, IoT systems include two additional components, namely hardware (sensors / actuators), and communication mechanisms for the interaction between the hardware and the Internet [6]. Hence, a multidisciplinary approach is needed. The basic requirements for any IoT systems include i) sensing (gathering of data from the environment including temperature, humidity, sound and motion); ii) connectivity (devices need to have ability to connect to the internet through wired or wireless connection); and iii) intelligence. Sensors measure environmental characteristics in a context-aware manner. Thus, they provide the possibility to a device to communicate with humans and the outside world [8]. Nonetheless, the interaction between things and people also needs to be considered [9]. Askar et al., [8] propose the use of Named Data Networking (NDN), a viable networking design for future Internet, because it provides a potential for IoT due to built-in support for naming, caching, mobility, and security. They argue that NDN can take care of the following IoT characteristics i) QoS e.g. response times and network traffic minimized packet loss, ii) scalability (scaling up e.g. from prototype to production), iii) energy efficiency (transmission saving in the IoT network), iv) security (without third-party services), and v) heterogeneity (interoperability among different devices).

It is important that IoT applications are designed with caution, taking into consideration the intertwined requirements of multiple objectives and the overall cost of implementation without reducing the Quality of Experience (QoE) and QoS levels. A balance between production cost and Return of Investment (RoI) should be considered and IoT solutions should be designed taking life quality into account [1]. The communication requirements of industrial IoT include real-time operations with low latency, high reliability, flexibility, and security. They are enabled by fifth generation of cell-based mobile communication architecture (5G) [10]. Industries are expected to capitalize on the potential of IoT to improve operational efficiency and sustainability with increased levels of automation and environmental awareness [11; 25].

Minoli et al. [12] highlighted that IoT-based systems, such as in IoT enabled smart buildings, have inherited requirements in terms of thermal comfort, usability, security, and energy efficiency, but lack of comprehensive end-to-end standards, fragmented cybersecurity solutions, and vertical applications does not allow full utilization. Achieving diverse IoT goals also bring difficulties in ensuring universal access, scalability, robustness, and security.

IoT requirements of smart buildings comprise video surveillance, access management, energy management (including lighting), and environmental monitoring (including fire detection). Cano-Suñén et al. [13] argued that IoT offers modularity and interoperability for implementing smart buildings and smart cities, by improving energy efficiency and indoor / outdoor quality at a low cost. In energy management, efficiency is improved by IoT-based sensing, automation, and management. Security is a major challenge in smart cities. Transforming urban cities into smart cities raise security and privacy concerns because of the use of IoT devices in diverse smart city applications [14].

Majeed et al. [15] proposed that blockchain offers enhanced security via storing transactions in a transparent, decentralized, secure, and immutable ledger. Despite the fact that shortcomings and open issues have been linked to blockchain technology, the benefits compensate the drawbacks due to characteristics, such as decentralization, encryption and anonymization, that facilitate enhancement of customers' data integrity, security and protection through Zero-Trust security principles [16]. Iqal et al. [17] investigated access control for protecting IoT resources by implementing access restrictions to data, objects, devices, and services to ensure safe interconnectivity. Metallidou et al. [18] asserted that Internet of Energy (IoE) is an IoT application regarding distributed energy systems, which aim at increasing energy efficiency, reducing energy waste, and improving environmental conditions. IoE has a significant impact on the acceleration of the clean energy transition goal of the EU to reach climate neutrality by 2050.

Boutot et al. [7] emphasized that it is crucial to address the adaptive nature of IoT systems at the beginning of the development life cycle in order to develop a comprehensive and exact system specification. They recommended the use of A Use Case Modelling Environment for IoT Systems (UCM4IoT) (a case-based modelling language) for requirement elicitation and specification of IoT systems. UCM4IoT considers the heterogeneity of IoT systems

and provides domain-specific language constructs to model the diverse facets of IoT systems. The notion of exceptional situations and adaptive system behavior are both addressed by the language. A textual modelling environment assists requirement engineers in writing use cases. Syntax-directed editing, validation of use case models, and requirement analysis are also supported by the environment.

Requirements are the foundation of any system under development. Hence, requirement elicitation and requirements engineering represent the most significant aspects in the system development life cycle [19;20]. Lauesen [21] investigated five accidents due to failures in large Danish government software systems. It is notable that of the 37 identified root causes only one was programming-related and that the inefficient identification of user needs and prioritization of requirements were the main reasons for failures. Uncertain requirements contribute to project failures [22]. Requirements failures are costly in terms of time, revenue, and reputation losses; they can even be a matter of survival [23].

The requirements of a system are divided into functional requirements (e.g., specific business functions, tasks, or behaviors of the system to be developed) and non-functional requirements (e.g., quality attribute of a system not directly related to the functionality of the system, e.g. reliability, availability, and security [24, 25]. Non-functional requirements can be legal requirements of systems, such as General Data Protection Regulation (GDPR) regulating privacy and security [26], Health & Safety and Equality legislations overseeing accessibility and risk assessments. Lee et al. [27] emphasized that in IoT systems security is the most important non-functional requirement, due to the risk of privacy invasion and safety threat. Important factors in securing IoT are authentication and access control technology among devices and services [28]. The use of artificial intelligence and machine learning (ML) can further be used to amplify the capabilities and autonomy of IoT devices [29]. Khurshid et al. [30] investigated IoT-oriented Healthcare systems, which deliver monitoring services of patients' data and facilitate immediate steps in case of emergency. Machine learning based techniques were used to safeguard security. They introduced a novel classification ML technique to extract non-functional requirements from IoT-oriented healthcare requirement documents. For implementing the automatic classification technique, a dataset containing requirements including accuracy, reliability, security and privacy, performance, compatibility, usability, functional, and maintainability was created. Relevant features and ML algorithms were used to perform the classification, which was more accurate than using written natural language for extracting formal requirements from the requirement document.

Requirements elicitation includes the activity of identifying the right stakeholders and collection of adequate system requirements (functional and non-functional) to achieve the objectives of the project [31]. Requirements engineering is extensively studied in system and software development fields to signify the importance of a systematic handling of requirements. Requirements engineering provides a systematized mean for adequately understanding the domain and context of a system under study. It facilitates capturing, analysis, modelling, verification, and validation of stakeholder needs and requirements [32]. Requirements elicitation is the first phase in requirements engineering and includes a complex and iterative activity for uncovering, extracting, and surfacing user needs, expectations and preferences [33]. Pandey et al. [33] emphasized that the aim of the first phase of requirements engineering is the gathering of viewpoints and information regarding the application domain, standards and restrictions, business and security requirements, as well as user requirements. It is a concerted human activity with intensive communication, collaboration, and negotiation between the requirements engineers on one hand and various stakeholders (e.g., customers, end-users, domain experts, project owners) often possessing diverse knowledge domains on the other hand [26]. The requirements elicitation process comprises four main activities, namely communication, negotiation, cooperation with stakeholders and accomplishing requirements development together with prioritization [34; 35]. To develop a successful system, it is important that the requirements engineers fully understand the system domain. Requirements engineering is considered a particularly significant and complicated process since at this stage accurate requirements are identified and important software development decisions are taken [20]. Detailed descriptions of the different phases in requirements engineering can be found in [36; 37; 38].

The focus of this paper is on the requirements engineering challenges when developing IoT systems. As a result of a literature overview and longitudinal experience from different software development approaches, the challenges in requirements elicitation and engineering are pinpointed. A conceptual framework is created to increase awareness of the importance of requirements engineering in the development of IoT systems. Adequate requirements engineering will improve the quality of the software process and the whole system, as well as of the resulting products and services. The framework can be used as a tool for requirements engineers in development of IoT to identify risks so that they can be minimized at an early stage.

The remainder of this paper is organized as follows. In the following section, we introduce prior work identifying requirements engineering challenges and practices. The next section looks into at the challenges through the lens of requirements elicitation and a classification of the challenges in people, process, business environment and principles of conduct. Finally, we demonstrate the IoT Requirements Framework that aims to provide a viewpoint of the challenges related to the requirements and raise awareness of potential conflicts that may occur on a multiplicity of dimensions and interactions of the different entities in the framework. The framework proposes mitigation actions to be taken in order to address the conflicts and reveal the expected benefits of an adequate requirements engineering process. Finally, conclusive statements and directions for future research directions are provided.

2 Challenges in RE

In this paper we look at the challenges through the lens of requirements elicitation which is the first step in requirements engineering and which is the basic step for understanding the aims, objectives, domain and context of the IoT system to be developed.

2.1 Stakeholder Identification

A stakeholder is an individual or group of people who affect or is/are affected directly or indirectly by the core activities of the system to be developed [38; 19; 20; 39]. Stakeholders are identified based on type of project, and their activities and interest in the system [38; 39]. Stakeholders work closely with the requirements engineer in requirements elicitation to determine project requirements and to set priorities, to realize project objectives, and to provide feedback regarding the outcome of the project. Challenges in stakeholder identification may include i) inability to identify main stakeholder (e.g. due to large number of stakeholders), ii) identification of inappropriate stakeholder, iii) different culture of stakeholders, and iv) delays in stakeholder identification.

Table 1: Quality elements affected by stakeholder identification [adopted from 13]

Elements	The influence of stakeholder identification
Requirements Quality	Reliable, available, consistent and reusable requirements will be gathered.
System Functionality	There will be a suitable product, compliance and accurate results.
System Reliability	The system function will perform properly with high integrity and recoverability.
System Performance	There will be interconnection and sequence between the system's functions, and the system will perform its functions perfectly in the required time.
System Usability	The system will be easy to use, confident and flexible.
System Sustainability	The three dimensions of sustainable development will be achieved: economic sustainability, environmental sustainability, social sustainability.

2.2 Communication

In requirements elicitation a communication gap is inherited in the communication process, due to the fact that at least two professional cultures are involved. On one side is the user who views the system from a business perspective and on the other side the requirements engineer who views the system from a technological perspective [26]. In IoT requirements even more professional cultures may be involved in the process because in addition to software, sensors and actuators are also used. Stakeholders' divergent values in national, organizational, and professional cultures impact adversely on communication required for successful requirements elicitation. The appropriate stakeholders may even be out of organizational reach, in particular in Global Software Development (GSD), where additional communication obstacles include time zone difference, lack of informal communication, incompatibility of processes, knowledge sharing, standards, task allocation and follow-up documentation and technical issues, such as ICTs.

The challenges of inadequate communication in the requirements elicitation process are [26]:

- *Missing requirements:*
 - Stakeholders do not mention all requirements due to lack of:
 - comprehension of potential solutions that a new system can offer;
 - understanding of what is necessary;
 - interest in the proposed system.
 - The requirements engineer forgets to ask some specific questions.

- *Reluctant participation*: stakeholders not interested in participating in the elicitation process because of:
 - mistrust in consequences of automation;
 - conflicts among stakeholders;
 - politics and disagreements regarding aim and objectives of the new system.
- *Misperception*: requirements engineer incorrectly captures requirements because of:
 - different terminology arising from different professional, organizational or national culture;
 - linguistic ambiguity due to terminological discrepancies which may occur between stakeholders that belong to different technical domains (different professional cultures).
 - lack of cross-cultural awareness.
 - inadequate negotiation and communication skills.
 - insufficient transfer of implicit obvious domain knowledge from stakeholders to requirements engineer.
 - inexperience of requirements engineers.
- *Disagreement*: requirements engineers and stakeholders disagree about certain requirements. Requirements negotiation aims to resolve any contradictory requirements. Requirements conflicts must be resolved for completing the requirements validation processes and progressing to the next phase of the development life-cycle.

Siakas et al. [40] carried out a study regarding requirements elicitation in multicultural environments. The Multicultural Requirements Elicitation (McRE) framework was developed which aims to help minimizing prejudice, conflicts, misunderstandings, and misinterpretations arising from cultural differences.

2.3 Domain Knowledge

If requirement engineers have project domain knowledge prior to the elicitation process then this impacts positively on the requirements engineering processes by fostering communication, and by facilitating a mutual understanding of the needs. Hadar et al. [41] state that the most common elicitation technique is interviews. Palomares et al. [42] carried out a study involving 24 practitioners from 12 Swedish Information Technology (IT) companies. Their results showed that group interaction techniques (meetings and workshops), are the most popular elicitation techniques practiced in the industry. They also revealed that in every project a variety of elicitation techniques are used. They identified a number of challenges including difficulties in understanding and prioritizing stakeholder needs. They also noted that requirements instability (i.e. volatility) was a predominant challenge.

When the requirements engineer does not understand the project domain the Requirement Specification Document is likely to be ambiguous and unclear and there will be a lack of requirements traceability throughout the project life-cycle. Requirements traceability is the ability to trace a requirement forwards and backwards within the system development lifecycle, i.e. from the origin, through the requirements specification and the system development to subsequent deployment and use [43]. Traceability from requirements to code assist software engineers regarding which parts of software code implement a specific requirement. This decreases the risk of code quality degradation during requirements volatility [44].

2.4 Requirements volatility

Requirements volatility is “*the emergence of new requirements or modification or removal of existing requirements*” [45]. More defects, lower performance, project delays and cost overruns are usually created by requirements volatility [22], which is a major, usually underestimated, risk factor [46]. Requirements are often not completely known or understood when a project starts. New requirements and changes to requirements may occur because of different reasons such as i) increased knowledge brought on by development activities, ii) new needs may surface because of unexpected changes in organizational or environmental policies, structures, and iii) work roles of intended end users [39].

The volatility of requirements includes the following sub-factors [47]:

- *Scope creep*: uncontrolled growth in the scope without customer approval.
- *Ambiguous system and project requirements*: unclear or unspecified requirements.
- *Continually changing system and project requirements*: unstable requirements.
- *Ill-defined project goals*: neglected functionality.
- *Abundance of features*: more features than needed in pursuit of customer/user.
- *Assumptions and Ambiguities*: implicit and invalid assumptions.

Elwahab et al. [45] assert that the causes of requirements volatility are attributed to:

- *Business environment changes: government regulations, factors related to market, competition, financial, policy, technological and legal changes etc.*
- *Development requirements iterations: requirements error, conflicting requirements, missing requirements, evolving user and technological needs, missing team members, cost upgrades etc.*

2.5 Classification of requirements challenges

The requirements challenges can be classified into:

- *People Challenges*
 - Inadequate stakeholder identification [38].
 - Communication gaps between requirements engineer and stakeholders [26];
 - Limited understanding of project domain knowledge [41; 42];
 - Inadequate negotiation and prioritization of requirements [48]
 - Stakeholders with unreasonable timelines and limited knowledge of what they want [34; 35].
- *Process Challenges*
 - No defined requirements engineering processes [23];
 - Processes not followed [23];
 - No process measure implemented [49];
 - No Process Improvement in place [49].
- *Business Environment Challenges*
 - Changes in government regulations [37];
 - Competitors [39];
 - Policy [45];
 - Technology [6];
 - Legal changes [22].
- *Principles of Conduct Challenges*
 - Values and principles are not established / not followed [49];
 - Methodology (e.g. Agile development) is not established / not followed [9];
 - Requirements Volatility [22; 39];
 - Ethical and Professional principles not established / not followed [50].

3 The Requirements Engineering Framework for IoT Applications

The Requirements Engineering Framework for IoT (REFIoT) aims to raise awareness of potential conflicts that may occur on a multiplicity of dimensions and interactions of the different entities in the framework.

Level 1 (L1) show the classification of the requirements challenges into four groupings (people, process, business environment, principles of conduct) including the underlying reasons to the requirements challenges (e.g. lack of or poor stakeholder identification). All the entities in L1 comprise of an effort (e.g. prioritization) and their impact on each other (e.g. poor communication has an impact on negotiation and poor measures have an impact on project domain knowledge). As a result, the different entities are intertwined in a complex effort-impact representation that gives rise to potential failures in the determination of unambiguous, correct, and complete system requirements as needed and required by the customer / user.

Level 2 (L2) receives inputs from the effort-impact matrix (L1), which also in turn impacts on the entries in L1. Diverse cultural values include national, organizational, team and professional cultures and, depending on the context, different cultures may be involved in all the entries of L1. For example, the multidisciplinary character of IoT requirements elicitation process (including entities such as software, sensors/actuators, communication mechanisms, business, sustainability) by nature include divergent values of the requirements engineers and the different stakeholders (professional values). This may create difficulties in the requirements negotiation process which need to be addressed.

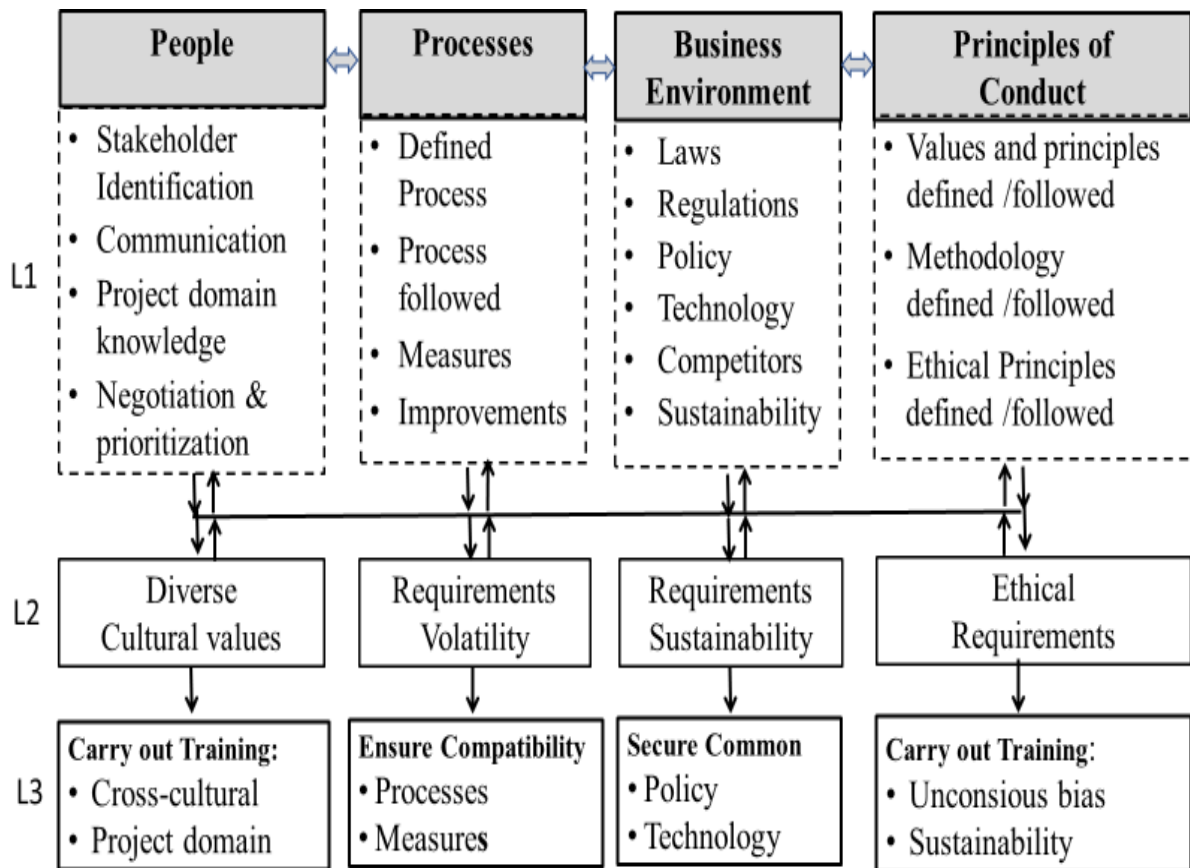


Figure 1: The Requirements Engineering Framework for IoT (REFIoT)

Level 3 (L3) proposes some initial mitigation actions to address discrepancies in viewpoints and considerations. Understanding the current situation and carrying out suitable cross-cultural, unconscious bias and sustainability training will help people involved in the requirements elicitation process to overcome cultural mismatches and prevent for misunderstandings and conflicts. Also training in the project domain may be useful in particular for people who have less experience of the project domain. Compatibility of processes and measures used by people in their daily work is imperative, because differences will be confusing and thus slowing down the effectivity of the requirements elicitation. It is also important that the same policy and technology are used for avoiding misunderstandings and unnecessary difficulties.

4. How the SPI Manifesto could address the IoT Requirements Challenges

Communication between and among requirements engineers and diverse stakeholders is crucial for eliciting unambiguous, complete and explicit requirements. Each case is different and the context dictates how explicitly or implicitly communication and knowledge (both explicit and tacit) sharing takes place [26].

Mapping the identified challenges (C1 to C4 - shown in the four columns of Figure 1) to the three SPI Manifesto Values, namely V1: People, V2: Business and V3: Change, reveals the main manifestations (of the potential problems) which are shown in the cells of Table 1. These problems result in detrimental effects and long-lasting impacts in the context of the three SPI Manifesto Values.

Table 1. Challenges mapped to the three SPI Manifesto Values

Challenges due to	V1: People	V2: Business	V3: Change
C1. Diversity of roles of multidisciplinary stakeholders [26; 39; 40]. National, Organizational, Team, and Professional Cultures	Cultural prejudice Misunderstandings & conflicts between roles due to different norms, terminology, language, beliefs, habits, experiences, and knowledge Lack of co-operation Acrimony	Erroneous or deficient requirements elicitation Loss of competitiveness Loss of data Diminished productivity Inefficiency Disorganization	Resistance to updates and evolution Stifling innovation Opposition to transformation and early adoption of innovations Loss of trust
C2. Diversity of processes of the multidisciplinary partners (individuals, teams, organizations, teams) [40; 50]	Frustration when Key Process Areas (KPAs) & improvement goals are not compatible between partners and teammates	Failed projects Cost (Budget and Time) overruns Dissatisfied customers	Capability maturity conflicts Resistance, malpractice
C3. Diversity of business environment [16; 40]	Confusions due to different policies, goals, rules, regulations	Divergence of working styles and approaches slows down the processes, and results in lower effectivity	Power clashes Entrenchment in own camp, waste of resources and loss of innovative ideas
C4. Principles of Conduct [3; 50; 51]	Misunderstandings and frustration due to different ways of understanding things	Slowed down workflows Reduced efficiency Decreased productivity	Fast staff turnover, loss of expertise and tacit knowledge

For each of the challenges identified in Table 1, above, we propose respective strategies for the prevention and mitigation of the detrimental effects of the potential problems. These are presented in Table 2, below.

As shown above, the values of the SPI Manifesto can inform and guide the identification of challenges. In turn experiences from industry can be fed to the debate in the SPI Manifesto community thus contributing any extensions/refinements to the Manifesto itself.

The set of recommendations that we propose will also assist companies implementing SPI (Software, System and Service Improvement) through achieving a better understanding, and towards addressing the challenges of IoT Requirements. A requirements elicitation process that adopts one or several of these proposals can help alleviate the challenges invoked in the requirements engineering process, thus leading to systems development and deployment that much better reflects the requirements/needs of diverse stakeholders and users. In addition, our recommendations will contribute to the on-going debate on the review and extension of the SPI manifestation, for example the suggestions made in Rahanu et al. [52] regarding a fourth Value, which we named Society (public interest), and includes the following six proposed principles:

- Principle 1. Ensure that all technology improves the Quality of Life for all users
- Principle 2. Respect the Use of Power by safeguarding solutions from demarcating the powerful and the less powerful
- Principle 3. Acknowledge that all technology comes with inherent Risks and Reliability issues
- Principle 4. Have a respect for Property Rights and fair use provision
- Principle 5. Ensure the right to Privacy is respected for all citizens
- Principle 6. To ensure Equity and Access for all citizens to technology.

Principles 1, 3, 6 are particularly important regarding recent IoT systems.

Table 2. Strategies for the prevention and mitigation of the detrimental effects of the potential problems

Challenges References	V1: People	V2: Business	V3: Change
C1 [26; 39; 40].	<p>Ensure that everybody is aware of the differences and produce advice as to how the differences might be addressed.</p> <p>Organize at the early stages of the project awareness and appreciation training on different norms, beliefs, but also on history and current political conflicts.</p> <p>Allocate roles and responsibilities to each individual considering their capabilities (knowledge, strengths or weaknesses) in a transparent manner.</p>	<p>Establish regular, transparent, and timely reviews, use of estimation and measurement.</p> <p>Ensure transparency and collaboration. Monitor progress through measurements and through sharing good practices.</p> <p>Handle corrections openly and collaboratively.</p>	<p>Hold brainstorming sessions, carry out training and take early and preventative action.</p> <p>Provide early notification and sharing of new ideas.</p> <p>Invite ideas from everybody evaluate them and adopt by consensus, thus creating and reinforcing an atmosphere of common purpose.</p> <p>Abandon blame culture, consult with all involved, encourage openness which in turn builds trust.</p>
C2 [1; 11; 16; 25; 28; 29; 40; 50]	<p>Organize at the early stages of the project training on processes capability models, and the role of new technologies.</p> <p>Hold open discussion at all stages prior to allocating personnel to tasks.</p>	<p>Hold open discussion regarding vision, mission, aims and objectives, KPAs and improvements.</p> <p>Base estimation and measurement on data from earlier projects and monitor progress.</p> <p>Agree measurable and achievable targets.</p>	<p>Carry out a self-assessment of capability of the organization and of the project team. If the respective organizations have prior professional assessments share the results in order to learn from each other and achieve alignment.</p> <p>Awareness of current developments in technologies and innovation will facilitate change.</p>
C3 [16; 40]	<p>Familiarize relevant staff from all organizations / constituencies, and align the respective practices e.g., using a comparative table.</p>	<p>Compare working styles and processes and identify common ground as a starting point in order to create a collaborative climate.</p>	<p>Establish regular meetings, progress reports, with opportunities for airing concerns, and integration of common goals.</p>
C4 [3; 50; 51]	<p>Familiarize staff with the similarities and differences of experiences. Start from identifying the similarities in order to develop understanding and build confidence.</p>	<p>Measure and monitor progress through the identification of strengths and successes</p> <p>Create inter and intra organizational trust.</p>	<p>Learn from each other's strengths and use change management.</p> <p>Replicate successes reinforcing appreciation and trust.</p>

In view of the new opportunities, and the challenges posed by the IoT requirements elicitation and specification, the solutions proposed are multiple and multidisciplinary.

5 Conclusions and Further Research Directions

The aim, of the research presented in this paper, was to identify and report on challenges in IoT during the requirements engineering, elicitation and specification processes. The development of IoT applications differ

from development of software (they also include devices with sensors / actuators, and communication mechanisms for the interaction between the hardware and the Internet) necessitating an interdisciplinary approach in requirements elicitation. The Requirements Engineering Framework for IoT (REFIoT) was developed based on an extensive literature review and the authors' collective longitudinal experience gained in industry as well as through the development of a number of studies of software development approaches and practices [1; 3; 11; 16; 25; 26; 28; 29; 39; 40; 50; 51].

The identified requirements challenges were:

- i) People Challenges (stakeholder identification, communication, domain knowledge and requirements negotiation & prioritization);
- ii) Process Challenges (processes defined, followed, measured and improved);
- iii) Business Environment Challenges (laws, regulations, policy, technology, competitors, sustainability); and
- iv) Principles of Conduct Challenges (values and principles, methodology, and ethical and professional principles defined and followed).

The complex interaction between all these challenges influences and is influenced by diverse cultural values of the people involved in the requirements elicitation process, the level of requirements volatility and the degree to which the requirements are ethical and sustainable. The REFIoT also proposes mitigation actions aiming to address discrepancies in viewpoints and attitudes. The REFIoT can be used as a tool by requirements engineers in the development of IoT applications to identify and minimize risks at an early stage. Mapping the IoT requirements challenges to the Values of the SPI Manifesto enabled the specification of a number of mitigation actions.

It is aimed to carry out an expert validation of REFIoT as the whole IoT field is gaining momentum. Experts experiences will deepen understanding and will provide a stepping stone towards future research directions including requirements characteristics, such as security and privacy concerns as well as scalability and interoperability. Another important research direction concerns Internet of Energy (IoE) for reaching the transition goal of EU regarding climate neutrality by 2050. IoE comprise distributed energy systems aiming to increase energy efficiency, reduce energy waste, and improve environmental conditions.

Reference

1. Lampropoulos, G., Siakas, K., & Anastasiadis, T. Internet of Things (IoT) in Industry: Contemporary Application Domains, Innovative Technologies and Intelligent Manufacturing. *International Journal of Advances in Scientific Research and Engineering*, 4(10), 109–118. 2018. <https://doi.org/10.31695/ijasre.2018.32910>
2. Haddadpajouh, H., Dehghantanha, A., Parizi, R. M., Aledhari, M., & Karimipour, H. A survey on internet of things security: Requirements, challenges, and solutions. *INTERNET OF THINGS*, 14, 100129. 2021. <https://doi.org/10.1016/j.iot.2019.100129>
3. Rahanu, H., Georgiadou, E., Siakas, K. and Ross, M. (2021). Ethical Issues Invoked by Industry 4.0. In: Yilmaz, M., Clarke, P., Messnarz, R., Wöran, B. (eds) *Systems, Software and Services Process Improvement. EuroSPI 2021. Communications in Computer and Information Science*, vol 1646. Springer, Cham, pp. 589-606, https://doi.org/10.1007/978-3-030-85521-5_39
4. Madakam, S., Ramaswamy, R. & Tripathi, S. *Internet of Things (IoT): A Literature Review. Journal Computer Communication*, 3, pp. 164–173. 2015.
5. Stankovic, J.A. Research Directions for the Internet of Things. *IEEE Internet Things Journal*, 1, pp. 3-9, 2014.
6. Memon, S.K., Nisar, K., Hijazi, M.H.A., Chowdhry, B.S., Sodhro, A.H., Pirbhulal, S. & Rodrigues, J.J.P.C. A Survey on 802.11 MAC Industrial Standards, Architecture, Security & Supporting Emergency Traffic: Future Directions. *J. Ind. Inf. Integr.* 24, 100225, 2021.
7. Boutot, P., Tabassum, M. R., Abedin, A., & Mustafiz, S. Requirements development for IoT systems with UCM4IoT. *Journal of Computer Languages*, 78, 101251. 2024. <https://doi.org/10.1016/j.cola.2023.101251>
8. Askar, N. A., Habbal, A., Alden, F. Z., Wei, X., Alaidaros, H., Guo, J., & Yu, H. (2023). Forwarding Strategies for Named Data Networking Based IoT: Requirements, Taxonomy, and Open Research Challenges. *IEEE access*, 11, 78363-78383. <https://doi.org/10.1109/ACCESS.2023.3276713>

9. Guerrero-Ulloa, G., Rodríguez-Domínguez, C., & Hornos, M. J. Agile Methodologies Applied to the Development of Internet of Things (IoT) Based Systems: A Review. *Sensors (Basel, Switzerland)*, 23(2), 790. 2023. <https://doi.org/10.3390/s23020790>
10. Varga, P., Peto, J., Franko, A., Balla, D., Haja, D., Janky, F., . . . Toka, L. 5G support for Industrial IoT Applications- Challenges, Solutions, and Research gaps. *Sensors (Basel, Switzerland)*, 20(3), 828. 2020. <https://doi.org/10.3390/s20030828>
11. Lampropoulos, G., Rahanu, H., Georgiadou, E., Siakas, D., Siakas, K. Reconsidering a Sustainable Future Through Artificial Intelligence of Things (AIoT) in the Context of Circular Economy. In: Misra, S., Siakas, K., Lampropoulos, G. (eds) *Artificial Intelligence of Things for Achieving Sustainable Development Goals. Lecture Notes on Data Engineering and Communications Technologies* (pp. 1–20), vol 192. Springer, 2024. Cham. https://doi.org/10.1007/978-3-031-53433-1_1
12. Minoli, D., Sohrawy, K., & Occhiogrosso, B. IoT Considerations, Requirements, and Architectures for Smart Buildings-Energy Optimization and Next-Generation Building Management Systems. *IEEE internet of things journal*, 4(1), 269-283.2017. <https://doi.org/10.1109/JIOT.2017.2647881>
13. Cano-Suñén, E., Martínez, I., Fernández, Á., Zalba, B., & Casas, R. Internet of Things (IoT) in Buildings: A Learning Factory. *Sustainability (Basel, Switzerland)*, 15(16), 12219. 2023. <https://doi.org/10.3390/su151612219>
14. Kilicay-Ergin, N., Barb, A., & Chaudhary, N. Knowledge elicitation methodology for evaluation of Internet of Things privacy characteristics in smart cities. *Systems engineering*, 27(2), 354-367. 2023. <https://doi.org/10.1002/sys.21726>
15. Majeed, U., Khan, L. U., Yaqoob, I., Kazmi, S. A., Salah, K., & Hong, C. S. Blockchain for IoT-based smart cities: Recent advances, requirements, and future challenges. *Journal of network and computer applications*, 181, 103007.2021. <https://doi.org/10.1016/j.jnca.2021.103007>
16. Lampropoulos, G., Siakas, K., Viana, J., & Reinhold, O. Artificial Intelligence, Blockchain, Big Data Analytics, Machine Learning and Data Mining in Traditional CRM and Social CRM: A Critical Review. In *2022 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)* (pp. 504–510). Niagara Falls, Ontario, Canada. IEEE. 2022. <https://doi.org/10.1109/WI-IAT55865.2022.00080>
17. Iqal, Z. M., Selamat, A., & Krejcar, O. A Comprehensive Systematic Review of Access Control in IoT: Requirements, Technologies, and Evaluation Metrics. *IEEE access*, 12, 1. 2024. <https://doi.org/10.1109/ACCESS.2023>
18. Metallidou, C. K., Psannis, K. E., & Egyptiadou, E. A. Energy Efficiency in Smart Buildings: IoT Approaches. *IEEE access*, 8, 63679-63699. 2020. <https://doi.org/10.1109/ACCESS.2020.2984461>
19. Khan, F. M., Khan, J. A., Assam, M., Almasoud, A. S., Abdelmaboud, A., & Hamza, M. A. M. A Comparative Systematic Analysis of Stakeholder's Identification Methods in Requirements Elicitation. *IEEE access*, 10, 30982-31011. <https://doi.org/10.1109/ACCESS.2022.3152073>. 2022.
20. Pacheco, C., & Garcia, I. A systematic literature review of stakeholder identification methods in requirements elicitation. *The J. of systems and software*, 85(9), 2171-2181. 2012. <https://doi.org/10.1016/j.jss.2012.04.075>
21. Lauesen, S. IT project failures, causes and cures. *IEEE Access* 8:72059–72067, 2020.
22. Dasanayake, S., Aaramaa, S., Markkula, J., & Oivo, M. Impact of requirements volatility on software architecture: How do software teams keep up with ever-changing requirements? *Journal of Software: Evolution and Process*, DOI: 10.1002/smr.2160. 2019.
23. Beecham, S., Hall, T., & Rainer, A. Defining a Requirements Process Improvement Model. *Software quality journal*, 13(3), 247-279. 2005. <https://doi.org/10.1007/s11219-005-1752-9>
24. Lee, R.Y. Requirements Elicitation (Chapter 5), *Software Engineering: A Hands-On Approach*, DOI: 10.2991/978-94-6239-006-5_5, Atlantis Press, 2013.

25. Siakas, D., Lampropoulos, G., Rahanu, H., Georgiadou, E. & Siakas, K. Emerging Technologies Enabling the Transition Toward a Sustainable and Circular Economy: The 4R Sustainability Framework. In: Yilmaz, M., Clarke, P., Riel, A., Messnarz, R. (eds) Systems, Software and Services Process Improvement. Communications in Computer and Information Science, vol 1891. Springer, Cham. 2023, https://doi.org/10.1007/978-3-031-42310-9_12
26. Siakas, E., Rahanu, H., Georgiadou, E. & Siakas, K. Towards Reducing Communication Gaps in Multicultural and Global Requirements Elicitation. In: Yilmaz M., Clarke P., Messnarz R., Reiner M. (eds) Systems, Software and Services Process Improvement. Communications in Computer and Information Science, vol 1442. Springer, Cham. pp. 257-277, 2021. https://doi.org/10.1007/978-3-030-85521-5_17
27. Lee, Y., Lim, J., Jeon, Y., & Kim, J. *Technology trends of access control in IoT and requirements analysis*. <https://doi.org/10.1109/ICTC.2015.7354730>. 2015.
28. Lampropoulos, G., & Siakas, K. Enhancing and securing cyber-physical systems and Industry 4.0 through digital twins: A critical review. *Journal of Software: Evolution and Process*, 35(7), e2494. 2022. <https://doi.org/10.1002/smr.2494>
29. Lampropoulos, G. Artificial Intelligence, Big Data, and Machine Learning in Industry 4.0. In J. Wang (Ed.), *Encyclopedia of Data Science and Machine Learning* (pp. 2101–2109). IGI Global. 2023. <https://doi.org/10.4018/978-1-7998-9220-5.ch125>
30. Khurshid, I., Imtiaz, S., Boulila, W., Khan, Z., Abbasi, A., Javed, A. R., & Jalil, Z. Classification of Non-Functional Requirements from IoT Oriented Healthcare Requirement Document. *Frontiers in public health*, 10, 860536. 2022. <https://doi.org/10.3389/fpubh.2022.860536>
31. Wong, L.R., Mauricio, D.S. & Rodriguez, G.D.: A systematic literature review about software requirements elicitation, *Journal of Engineering Science and Technology*, Vol. 12, no. 2, pp. 296-317, 2017.
32. Ali, N. & Lai, R. Requirements Engineering in Global Software Development: A Survey Study from the Perspectives of Stakeholders, *Journal of Software*, 13(10):520-532. 2018.
33. Pandey, D., Suman, U. & Ramani, A.K. An Effective Requirement Engineering Process Model for Software Development and Requirements Management. *Int. Conference on Advances in Recent Technologies in Communication and Computing*, IEEE Comp. Soc., pp. 287 – 291. 2010.
34. Grunbacher, P. & Braunsberger, P. Tool Support for Distributed Requirements Negotiation: Lessons Learned. Cooperative Methods and Tools for Distributed Software Processes. *IEEE*, pp. 46–55. 2003.
35. Grunbacher, P., Halling, M., Biffel, S., Kitapci, H. & Boehm, B.W. Integrating Collaborative Processes and Quality Assurance Techniques: Experience from Requirements Negotiation. *Journal of Management Information System*, 20(4): pp. 9-29. 2004
36. CMMI Requirements Development (RD), accessed 12.03.2024, available at <https://www.cmmi.co.uk/cmmi/RD.htm>.
37. ISO/IEC/IEEE 26511:2018 Systems and software engineering - Requirements for managers of information for users of systems, software, and services, accessed 12.03.2024 available at <https://www.iso.org/standard/70879.html>.
38. Elneel, D. A., Fakharudin, A. S., Ahmed, E. M., Kahtan, H., & Abdullateef, M. Stakeholder Identification Overview and Challenges in Requirements Engineering Prospective. *2nd Int. Conference on Computing and Information Technology (ICCIT)*, IEEE, <https://doi.org/10.1109/ICCIT52419.2022.9711653>. 2022.
39. Siakas, E., Rahanu, H., Georgiadou, E. & Siakas, K. Requirements Volatility in Multicultural Situational Contexts. In: Yilmaz, M., Clarke, P., Messnarz, R., Wöran, B. (eds) Systems, Software and Services Process Improvement. EuroSPI 2022. Communications in Computer and Information Science, vol 1646, pp. 633-655, Springer, Cham. 2022. https://doi.org/10.1007/978-3-031-15559-8_45
40. Siakas, K., Georgiadou, E., Rahanu, H., Siakas, E., Meggoudis, N. & Siakas, D. A Multicultural Requirements Elicitation Framework, *Journal of Software Engineering Research and Development (JSERD)*, Springer (accepted, in press). 2024.

41. Hadar, I., Soffer, P., & Kenzi, K. The role of domain knowledge in requirements elicitation via interviews: An exploratory study. *Requirements engineering*, 19(2), 143-159. <https://doi.org/10.1007/s00766-012-0163-2>. 2014.
42. Palomares, C., Franch, X., Quer, C., Chatzipetrou, P., López, L., & Gorschek, T. The state-of-practice in requirements elicitation: An extended interview study at 12 companies. *Requirements engineering*, 26(2), 273-299. 2021. <https://doi.org/10.1007/s00766-020-00345-x>
43. Torkar, R., Gorschek, T., Feldt, R., Svahnberg, M., Uzair Akbar, R., & Kamran, K. Requirements Traceability: A Systematic Review and Industry Case Study. *International journal of software engineering and knowledge engineering*, 22(3), 385. 2012. <https://doi.org/10.1142/S021819401250009X>
44. Al-Msie'deen, R. Requirements Traceability: Recovering and Visualizing Traceability Links Between Requirements and Source Code of Object-oriented Software Systems. *International Journal of Computing and Digital System (Jāmi'at al-Baḥrayn. Markaz al-Nashr al-'Ilmī)*, 14(1), 279-295. 2023. <https://doi.org/10.12785/ijcds/1401123>
45. Elwahab, K.A., Latif, M.A.E. & Kholeif, S. Identify and Manage the Software Requirements Volatility: Proposed Framework and Case Study, *International Journal of Advanced Computer Science and Appl.*, 7(5):64-71. 2016.
46. Dev, H., & Awasthi, H. A Systematic Study of Requirement Volatility during Software Development Process, *International Journal of Computer Science Issues*, 9(2):528-533. 2012.
47. Christel, M.G. & Kang, K.C. Issues in Requirements Elicitation, Technical Report, CMU/SEI-92-TR-012, ESC-TR-92-012, Carnegie Mellon University, Pittsburgh, Pennsylvania, US. 1992.
48. Akram, F., Ahmad, T., & Sadiq, M. Recommendation systems-based software requirements elicitation process—a systematic literature review. *Journal of engineering and applied science (Online)*, 71(1), 29-21. 2024. <https://doi.org/10.1186/s44147-024-00363-4>
49. El Emam, K., & Birk, A. Validating the ISO/IEC 15504 measure of software requirements analysis process capability. *IEEE transactions on software engineering*, 26(6), 541-566. 2000. <https://doi.org/10.1109/32.852742>
50. Siakas, E., Rahanu, H., Loveday, J., Georgiadou, E., Siakas, K. & Ross, M. Managing Ethical Requirements Elicitation. In: Yilmaz, M., Clarke, P., Riel, A., Messnarz, R. (eds) *Systems, Software and Services Process Improvement. Communications in Computer and Information Science*, vol 1890. Springer, Cham. 2023. https://doi.org/10.1007/978-3-031-42307-9_19
51. Rahanu, H., Loveday, J., Siakas, E., Georgiadou, E., Siakas, K., Ross, M. (2022). The Proposal of Adding a Society Value to the Software Process Improvement Manifesto. In: Yilmaz, M., Clarke, P., Messnarz, R., Wöran, B. (eds) *Systems, Software and Services Process Improvement. Communications in Computer and Information Science*, vol 1646. Springer, Cham, pp. 673–687, https://doi.org/10.1007/978-3-031-15559-8_47