Applying Quality Requirements Framework to an IoT System and its Evaluation

Tsuyoshi Nakajima
Department of Computer Science and Engineering
Shibaura Institute of Technology
Tokyo, Japan
e-mail: tsnaka@shibaura-it.ac.jp

Toshihiro Komiyama
Software Engineering Division
NEC Corporation
Tokyo, Japan
e-mail: t-komiyama@bk.jp.nec.com

***Abstract*—Modern information and communication technology systems are more focused on their quality requirements since they have been increasing their complexity. This paper shows how the quality requirements framework of the ISO/IEC 25030 can be applied to an Internet of things application. The results of this application are qualitatively evaluated to show the usefulness of the framework for defining quality requirements, and also its problems to be solved.**

Keywords—Quality requirements; SQuaRE; IoT.

I. INTRODUCTION

Information and Communication Technology (ICT) systems are increasingly used to perform a wide variety of organizational functions and personal activities. The quality of these products enables and impacts various business, regulatory and information technology stakeholders. High-quality ICT systems are hence essential to provide value, and avoid potential negative consequences, for the stakeholders.

To develop such high-quality ICT systems, it is important to define quality requirements, because finding the right balance of quality requirements, in addition to well-specified functional requirements, is a critical success factor to meet the stakeholders' objectives.

Furthermore, the complexity of ICT systems has grown exponentially with the advent of modern digital technologies like Internet of Things (IoT). This has also led to focus on more and more quality requirements that are critical to modern ICT systems.

ISO/IEC 25030 quality requirements was published in 2007, and its revision process has been going on to expand its scope from software to ICT systems [2]. The standard belongs to ISO/IEC 25000 series: Systems and software Quality Requirements and Evaluation (SQuaRE) has been developed as the successor of the other standards on product-related quality, including ISO/IEC 9126.

The quality requirements framework is applied to an IoT system in our previous work [1]. This paper fleshes out the contents to provide detailed discussion and evaluation of the framework. Section II explains the quality requirements framework and Section III describes the target IoT system, and then the framework is applied to the system in Section IV, and results of the application are qualitatively evaluated in Section V. Section VI reviews the related works, and finally, Section VII concludes this study.

II. QUALITY REQUIREMENTS FRAMEWORK

A. Architecture of the SQuaRE series

The SQuaRE series consists of five main divisions and one extension division. The main divisions within the SQuaRE series are:

- **ISO/IEC 2500n - Quality Management Division**. The standards that form this division define all common models, terms and definitions used by all other standards in the SQuaRE series. The division also provides requirements and guidance for the planning and management of a project.
- **ISO/IEC 2501n - Quality Model Division**. The standards that form this division provide quality models for system/software products, quality in use, data, and IT services. Practical guidance on the use of the quality model is also provided.
- **ISO/IEC 2502n - Quality Measurement Division**. The standards that form this division include a system/software product quality measurement reference model, definitions of quality measures, and practical guidance for their application. This division presents internal measures of software quality, external measures of software quality, quality in use measures and data quality measures. Quality measure elements forming foundations for the quality measures are defined and presented.
- **ISO/IEC 2503n - Quality Requirements Division**. The standard that forms this division helps specifying quality requirements. These quality requirements can be used in the process of quality requirements elicitation for a system/software product to be developed, designing a process for achieving necessary quality, or as inputs for an evaluation process.
- **ISO/IEC 2504n - Quality Evaluation Division**. The standards that form this division provide requirements, recommendations and guidelines for system/software product evaluation, whether performed by independent evaluators, acquirers or developers. The support for documenting a measure as an Evaluation Module is also presented.

B. Quality requirements and quality models/measures

Quality in use is the extent to which the influence (behavioral and attitudinal outcomes and consequences) of use of an ICT product or service meets the needs of users or other stakeholders in specific contexts of use (Figure 1). Therefore, quality in use requirements (QIURs) specify the required levels of quality from the stakeholders' point of view. These requirements are derived from the needs of various stakeholders. QIURs relate to the outcomes and consequences when the product is used in a particular context of use, and QIURs can be used as the target for validation of the product.

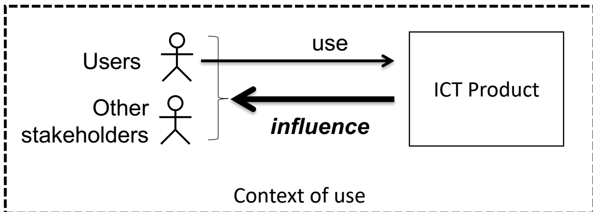

Figure 1. Quality in use

QIURs can be specified using quality in use model (ISO/IEC 25010 [3]) and measures (ISO/IEC 25022 [5]). Figure 2 describes characteristics and subcharacteristics of quality in use model.

Effectiveness	Efficiency	Satisfaction	Freedom from risk	Context coverage
Effectiveness	Efficiency	Usefulness Trust Pleasure Comfort	Economic risk mitigation Health and safety risk mitigation Environmental risk mitigation	Context completeness Flexibility

Figure 2. Quality in use model [3]

Product/Data quality is the capability of an ICT product/data that enables stakeholders to meet their needs (Figure 3).

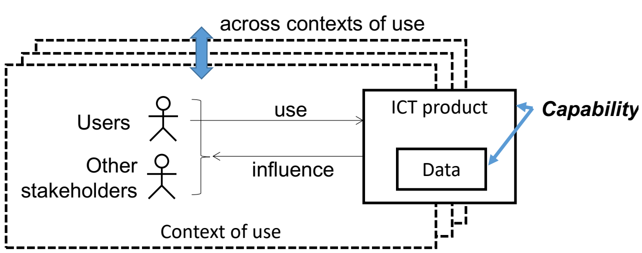

Figure 3. Product quality and data quality

Product quality requirements (PQRs) specify levels of quality required from the viewpoint of the ICT product. Most of them are derived from stakeholder quality requirements including QIURs, which can be used as targets for verification and validation of the target ICT product.

PQRs can be specified using product quality model (ISO/IEC 25010 [3]) and measures (ISO/IEC 25023 [6]). Figure 4 describes characteristics and subcharacteristics of product quality model.

Functional suitability	Performance efficiency	Compatibility	Usability
Functional completeness Functional correctness Functional appropriateness	Time-behavior Resource utilization Capacity	Co-existence Interoperability	Appropriateness recognisability Learnability Operability User error protection Use interface aesthetics Accessibility

Reliability	Portability	Maintain-ability	Security
Maturity Availability Fault tolerance Recoverability	Adaptability Installability Replaceability	Modularity Reusability Analysability Modifiability Testability	Confidentiality Integrity Non-repudiation Accountability Authenticity

Figure 4. Product quality model [3]

Data quality requirements (DQRs) specify levels of quality required for the data associated with the ICT product. These can be derived from related QIURs and PQRs. DQRs can be used for verification and validation from the data side.

DQRs can be specified using data quality model (ISO/IEC 25012 [4]) and measures (ISO/IEC 25024 [7]). Figure 5 describes 15 characteristics of data quality model, which are categorized by inherent and/or system dependent.

Inherent	Inherent & System dependent	System dependent
Accuracy Completeness Consistency Credibility Currentness	Accessibility Compliance Confidentiality Efficiency Precision Traceability Understandability	Availability Portability Recoverability

Figure 5. Data quality model [4]

C. *System hierarchy and scope of quality requirements*

Figure 6 describes the system hierarchy the SQuaRE series suppose and the scope for each type of quality requirements.

An information system, as the scope of QIURs, includes at least one ICT product, one user and relevant environments, and also can include other stakeholders such as developers, acquirers, regulatory bodies and society at large.

An ICT product, includes software, and also can include data, hardware, communication facilities, and other ICT products as its ICT components. PQRs are defined for the ICT product or its constituents (including sub-ICT products, hardware, communication facilities, software, and in some case software components), and DQRs are defined for the data inside the ICT product.

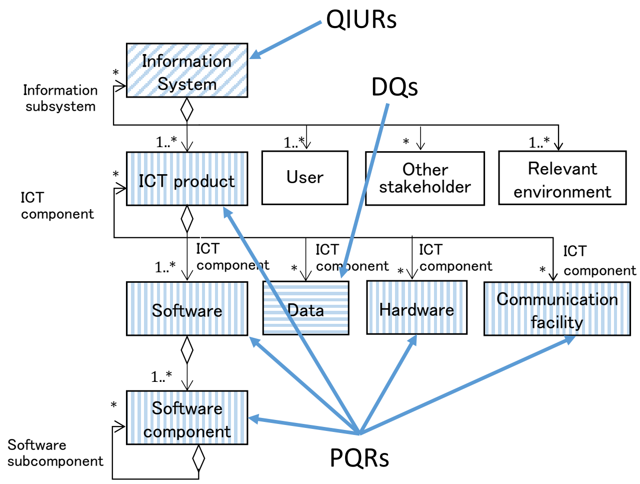

Figure 6. System hierarchy and scope of quality requirements

Figure 7 shows an example of mapping of a small IoT system, named Room open/close monitoring system, to the SQuaRE system hierarchy. The system judges whether the room is open or close based on luminance data measured under the room light, and user can know it through web.

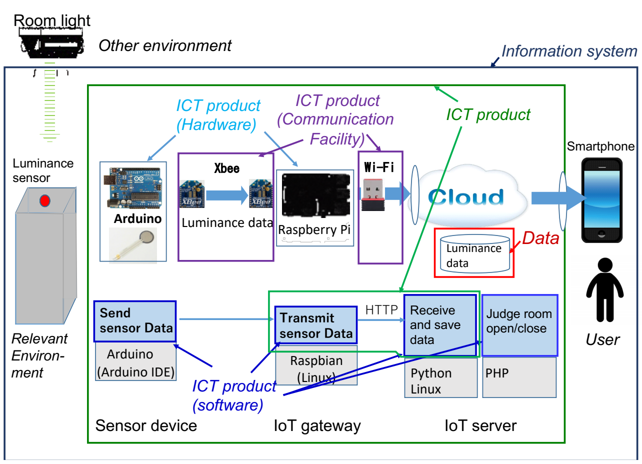

Figure 7. Mapping of Room open/close monitoring system to the SQuaRE system hierarchy

The *information system* includes a target *ICT system*, its *users* and *relevant environment*, which includes smartphones (as non-target ICT products) and the physical layout and phenomena in the room. The target *ICT product* includes a sensor device and an IoT gateway as *hardware*, Xbee and Wi-Fi as *communication facilities*, several software components (such as "send sensor data," "transmit sensor data," "receive and save data" and "judge room open/close") as *software*, and "luminance data" as *data*. If an ICT product is to be developed, the quality of all the target entities must be addressed and managed.

D. *Quality requirements framework*

The revision of ISO/IEC DIS 25030 [2] will provide a framework for quality requirements, which consists of concept of the quality requirements, and processes and methods to elicit, define, use and govern them.

There are three important points:

- To elicit quality requirements, not only direct users of the ICT product but also indirect users (using results of the product) and other stakeholders, such as developers, regulatory body, and society at large should be taken into account. TABLE I shows which type of stakeholders is a source of, a user of and relevant to which type of quality requirements.

TABLE I. STAKEHOLDERS AND TYPES OF QUALITY REQUIREMENTS

Stakeholder		Quality requirements		
		QIUR	PQR	DQR
User	Primary User	S	S	S
	Secondary User	S	S	S
	Indirect User	S		S
Other stakeholder	Developer	U	S, U	S, U
	Acquirer	U	U	U
	Regulatory body	S	S	S
	Society	R		

S: a source of / U: a user of / R: relevant to

- QIURs should be considered first because most of PQRs are derived from QIURs, and they should be deployed into PQRSs and DQRs of its sub-products (smaller ICT products, software, data, hardware and communication facilities) to meet them. Figure 8 describes how quality requirements derive others in the system hierarchy.

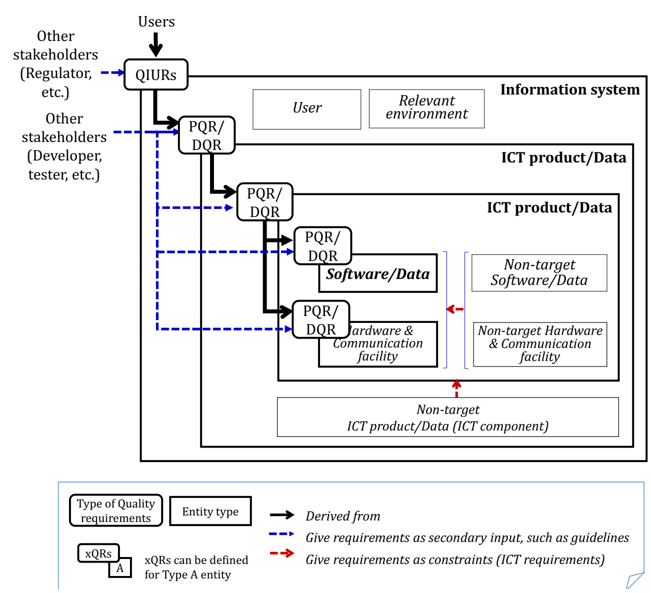

Figure 8. Derivation of quality requirements [2]

- Quality requirements should be defined quantitatively, in order not to be vague and unverifiable requirements that depend on subjective judgement for their interpretation. To specify the quality measure, ISO/IEC 25022 for QIUR, ISO/IEC 25023 for PQR and ISO/IEC 25024 for DQR should be used.

III. IoT SYSTEM AND TARGET SYSTEM

A. *Characteristics of IoT systems*

The IoT envisages the future in which digital and physical things or objects can be connected by means of suitable information and communication technologies, to enable a range of applications and services. The IoT's characteristics include [8]:

- many relevant stakeholders involvement
- device and network heterogeneity and openness
- resource constrained
- spontaneous interaction
- increased security attack-surface

These characteristics will make development of the diverse applications and services a very challenging task.

B. *Target system*

The target IoT system, to which SQuaRE's quality requirements framework is applied, is Elderly monitoring system. Figure 9 shows its system architecture.

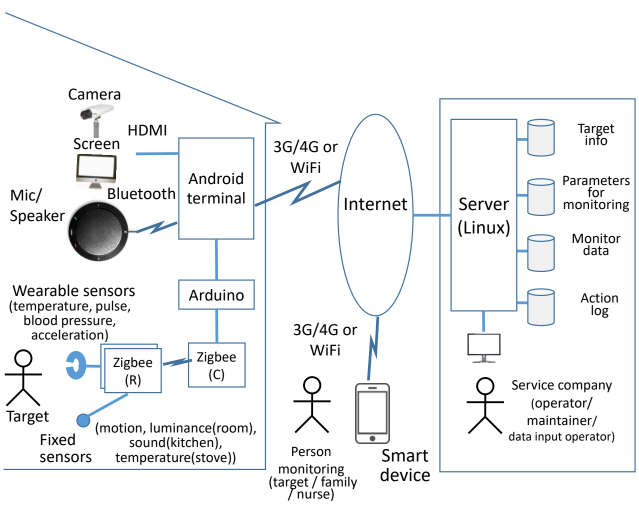

Figure 9. Elderly monitoring system [9]

The sensor devices of the system gathers sensor data of the target elderly living alone. The sensors include motion, luminance, temperature, sound (microphone) and vision (camera) sensors at fixed points in the elderly's house, and wearable sensors to measure body temperature, pulse, blood pressure, and acceleration of the target. These sensor data are sent to the server of the service company, which monitors and analyzes actions and body conditions of the target to provide several services, such as informing the designated persons (persons to monitor) of the dangerous situation about the target, directly give advices to the target through the speakers, and so on.

TABLE II shows the important data for this system in the site of the service company.

TABLE II. IMPORTANT DATA FOR THE SYSTEM

Data	Description	Data items
Target Info	Personal data about targets, including their medical history	Target ID, Name, Birthday, Medical history, Physical info, Place
Parameters for monitoring	Parameters and rules about what and how to monitor	Target ID, Sensor data (type, range, accuracy, unit), Sensor configuration, Abnormity: (data, range)->action Persons monitoring the target
Monitor data	Time series of data for targets and system components monitored from sensors	Target ID, Sensor data with time, Status of system components with time
Action log	Time series of targets' actions the system guesses	Target ID, Sensor data with time, Status of system components with time

Figure 10 describes all the use cases of elderly monitoring system.

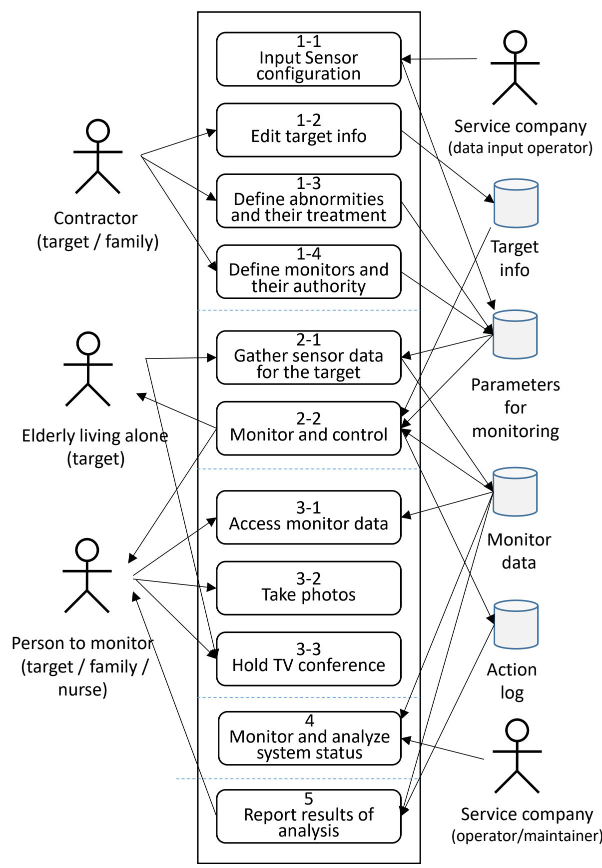

Figure 10. Use cases of the elderly monitoring system (written by the author)

IV. Application of the Framework

A. Stakeholder identification and selection of important QIURs

In the first step, stakeholders of the target system are to be identified, in which the quality requirements framework provides categories of stakeholders: direct users, indirect user and other stakeholders. Other stakeholders include users of quality requirements (developers, acquirers and independent evaluators), regulatory bodies, and society at large.

The stakeholders identified for the elderly monitoring system are:

- Direct user: contractor, elderly living alone, family, nurse, and service company's operators
- Indirect user: service company's managers
- Other stakeholder: Developer, Ambulance

In the second step, the outcomes and consequences the target ICT product is required to provide should be identified. First, for all stakeholders, their goals to achieve through using the target system are extracted. In case of the direct users, there must be some use cases of the system **(**Figure 10), in which they are involved to achieve their goal. In case of indirect users, who uses not the product itself but the outputs of the product, and other stakeholders, which do not use it but may get influenced from it, there are no use cases relevant to their goals.

Since the quality in use model (its characteristics/ subcharacteristics), shown in Figure 2, categorizes outcomes and consequences that the ICT product provide, this step is simplified into selecting important quality in use characteristics/ subcharacteristics for achieving stakeholders' goals (and use cases).

TABLE III shows an example of selecting important QIURs for direct users. One example is about contractor, which has the goal to inform the service company of what he/she wants them to do. This goal corresponds to Use case 1-2, 1-3 and 1-4, and therefore, efficiency (of operation for input) and freedom from risks (of inputting wrong parameters) are selected as important subcharacteristics of quality in use. Another example is about the elderly living alone has two goals: to detect the designated abnormalities on himself/ herself to take the designated actions, and to obtain useful information on his/her current body conditions and behavioral patterns. The former goal corresponds to Use case 2-2 "Monitor and control," and therefore effectiveness (early medical treatment) and trust (on getting correct results on proper timing) are selected as important subcharacteristics of quality in use. The latter goal corresponds to Use case 5 "Report results of analysis," and therefore effectiveness (obtaining useful information on current body conditions and behavioral patterns to provide objective insights) is selected.

These QIURs, which consist of selected subcharacteristics and their brief description, are a starting point for further enhancement to detailed quality requirements and for derivation of PQRs and DQRs, which is described in Section B.

TABLE III. QIURs SELECTION FOR DIRECT USERS

Stakeholder	Goal	Use case	QIUR (with target outcomes and consequences)
Elderly living alone (**direct user**)	Detect designated abnormalities for the target, and take actions.	2-2	**Effectiveness**: early medical treatment **Trust**: correct results on proper timing
	Obtain his/her own current body condition and behavioral pattern.	5	**Effectiveness**: obtain info on current body condition and behavioral pattern to provide objective insights.
Family (**direct user**)	Confirm target's normality.	3-1 3-2	**Effectiveness**: see target's condition anytime and anywhere
	Be informed of target's serious abnormalities.	2-2	**Trust**: correct results on proper timing
			Freedom from risks: prevention from * overlook of serious abnormalities * unnecessary notice on trivial abnormalities
Nurse (**direct user**)	Confirm target's normality.	3-1 3-2	**Effectiveness**: remote nursing **Efficiency**: early notice of patient's abnormalities
	Be informed of target's all abnormalities.	2-2	**Effectiveness**: early treatment **Trust**: correct results on good timing
			Freedom from risks: prevention from overlook of serious abnormalities
	Create reports for asking doctors to diagnose abnormalities.	5	**Efficiency**: automatic reporting
Service company's operator (**direct user**)	Monitor all equipment, and take actions if something wrong with them.	4	**Efficiency**: system monitor and control **Effectiveness**: preventive actions before disfunction or malfunction
	Maintain and update system and equipment.	1-1	**Efficiency:** maintenance activities
Contractor (**direct user**)	Inform the service company of what he/she wants them to do.	1-2 1-3 1-4	**Efficiency**: operation for input **Freedom from risks**: prevention from wrong input

TABLE IV shows an example of selecting important QIURs for indirect users and other stakeholders.

TABLE IV. QIURs SELECTION FOR INDIRECT USERS AND OTHER STALEHOLDERS

Stakeholder	Goal	QIUR (with target outcomes and consequences)
Service company's manager (**indirect user**)	Customer satisfaction	**Usefulness** **Trust**
	Prevention from incidents	**Freedom from risks:** prevention from * incidents by system faults or malfunctions * incidents by normal operation * privacy leakage * malfunction by malicious attack
Developer (**Other stakeholder**)	Achieve QCD goal	**Efficiency**: development activities
	Update the system to implement new functions periodically	**Efficiency**: maintenance activities
Ambulance (**Other stakeholder**)	Dispatch ambulance cars on demand (by nurse's call)	**Freedom from risks**: prevention from unnecessary dispatches of ambulance cars

For instance, the service company's manager has two goals: to get customer satisfaction, and to prevent from incidents that may affect the company's business. To achieve the former goal, usefulness (of the product) and trust (on getting good services) are selected as important subcharacteristics of quality in use. To achieve the latter, freedom from risks (of system faults, security incidents and so on) is selected.

Other stakeholders, such as developers and regulatory bodies, also give some quality requirements on the target entities.

B. *Derivation of PQRs and DQRs*

As described in previous section, QIURs for each stakeholder have been elicited and documented. In the next step, they recursively evolve into PQRs and DQRs for the target entities at the lower level of the system hierarchy.

Figure 11 is a mapping of Elderly monitoring system to the SQuaRE system hierarchy after the concept design finished, in which half-tone processed IoT devices, data and software components are the target entities whose quality must be managed.

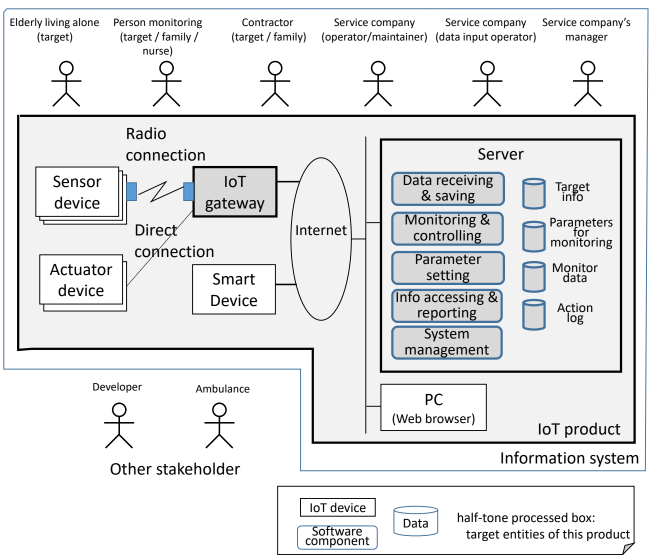

Figure 11. Target entities for quality requirements for Elderly monitoring system

The other entities, which are non-target ones, may influence and give some constraints to the target entities that include or interact with them.

To meet the corresponding QIURs, important product quality characteristics/ subcharacteristics (shown in Figure 4) for PQRs, and data quality characteristics/ subcharacteristics (shown in Figure 5) for DQRs are selected. Some PQRs for the target ICT product may be deployed into subcomponents to meet them (denoted with ->). DQRs are identified for the data files or data base used in the product.

TABLE V exemplifies how to derive PQRs and DQRs from QIURs of the service company's manager. From freedom from risks of incidents by system faults or malfunctions, three PQRs of availability for server, and maturity and time-behavior of the whole ICT product are identified. Availability of server (PQRs) entails recoverability of all the data on server (DQRs). Maturity of the whole ICT product (PQRs) is deployed into maturity of all the subcomponents (PQRs), including IoT devices, and software components, and accuracy, completeness, and consistency of all the data (DQRs). Time-behavior of the whole ICT product (PQRs) is deployed into throughput of server (PQRs), which entails efficiency and accessibility of monitor data (DQRs).

From freedom from risks of privacy leakage (QIURs), confidentiality of monitor data, target info and action log (DQRs) is derived, and then, one of monitor data entails confidentiality of all the devices and communication facilities from sensors to server, and one of target info and action log entails confidentiality of server and web.

TABLE V. DERIVATION OF PQRs AND DQRs FROM QIURs OF SERVICE COMPANY'S MANAGER

Stakeholder	QIUR	PQR	DQR
Service company's manager (**indirect user**)	**Freedom from risks:** prevention from		
	* incidents by system faults or malfunctions	**Availability** of server	**Recoverability** of all the data on server
		Maturity ->**Maturity** of all sub-components	**Accuracy, Completeness and Consistency** of all data
		Time-behavior ->Throughput of server	**Efficiency** and **Accessibility** of Monitor data
	* incidents by normal operation	**Maturity**:	**Accuracy, Consistency** and **Currentness** of Monitor data
	* privacy leakage	**Confidentiality** of all the devices and communication facilities from sensors to server	**Confidentiality** of Monitor data
		Confidentiality of server and web	**Confidentiality** of Target info and Action log
	* malfunction by malicious attack	**Integrity**: IoT devices, network	**Traceability** of Parameters for monitoring

TABLE VI exemplifies how to derive PQRs and DQRs from QIURs of the contractors (direct user). Use case 1-2, 1-3 and 1-4 are associated with "parameter setting," which is a software component on the server, and therefore, the derived PQRs and DQRs are respectively for the component and for "parameters for monitoring" as its input/output data. From efficiency (of operation for input), operability and accessibility of parameter setting as PQRs are derived, which entails understandability of parameters for monitoring (DQRs). From freedom from risks (of inputting wrong parameters) (QIURs), learnability and user error protection of parameter setting (PQRs) are derived, which entails understandability of parameters for monitoring (DQRs).

When considering freedom from risks about an IoT system, it is necessary to consider not only the risks relating to the integrity of the system and the confidentiality of its important data, but also the risks that the system gives some damage to the other systems; e.g., some IoT devices in the system infect malware to contributes to distributed denial-of-service (DDoS) attacks [10][11]. This means that the product quality requirements for IoT devices connecting to the Internet should include general internet security requirements.

TABLE VI. DERIVATION OF PQRs AND DQRs FROM QIURs OF CONTRACTOR

Stakeholder	Use case	QIUR	PQR	DQR
Contractor (**direct user**)	1-2 1-3 1-4	**Efficiency**: operation for input	**Operability** and **Accessibility** of Parameter setting (web)	**Understandability** of Parameters for monitoring
		Freedom from risks: prevention from wrong input	**Learnability** and **User error protection** of Parameter setting (web)	**Accuracy**, **Completeness** and **Consistency** of Parameters for monitoring

C. *Specifying quality requirements*

Quality requirements framework requires to quantitatively specify all the QIURs, PQRs and DQRs specified by using the quality requirements structure, shown in Figure 12.

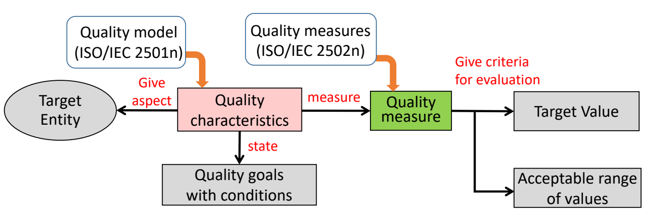

Figure 12. Quality requirements structure

Selected important quality subcharacterisrics selected and derived in Sections A and B are enhanced through it into complete quality requirements.

The following example describes a PQR for "User error protection of Parameter setting" in TABLE VI.

- **Target entity:** Parameter setting
- **Selected characteristic:** User error protection
- **Quality goal with conditions:**
  Parameter setting assist contractor to correctly input parameters for monitoring through web.
- **Quality measure:** Avoidance of user operation error (Uep-1-G [6])
- **Target value:** 1
- **Acceptable range of values:** 0.98 - 1.00

V. QUALITATIVE EVALUATION

The following evaluation results are obtained from the application of the quality requirement framework defined in ISO/IEC DIS 25030 [2] to an IoT product of Elderly monitoring system:

A. *Stakeholder identification and selection of important QIURs*

- Merits:
  - ➢ The categorization and examples of roles for stakeholders makes it easy to identify stakeholders, especially not overlooking indirect users and other stakeholders other than direct users. In addition, provided categories and roles ease to guess stakeholders' essential goals.
  - ➢ Knowing the goals of stakeholders and their use cases, it is easy to find the quality sub-characteristics related to them.
- Issues:
  - ➢ There may be a high possibility that relationship patterns between the stakeholder's roles and quality sub-characteristics can be developed.

B. *Derivation of PQRs and DQRs*

- Merits:
  - ➢ Because extracting the QIURs first and then associating them with PQRs and DQRs, the necessity and the priority of PQRs and DQRs are much more visible than extracting them alone. These would be useful in the steps of prioritizing of and resolving conflicts between quality requirements, which are in the quality request framework but not applied this time.
  - ➢ Mapping the target information system and the ICT product to the SQuaRE system hierarchy provides two advantages:
    - ✧ to ease to clarify the target entities whose quality should be managed, and
    - ✧ to support to derivate PQRs and DQRs recursively along with the hierarchy.
- Issues:
  - ➢ It is difficult to check whether PQRs and DQRs it are comprehensively derived from QUIRs. The framework does not support how to check it. There may be a high possibility to develop patterns of mapping from the types of ICT products to a set of important quality sub-characteristics.

C. Specifying quality requirements

- Merits:
  - Because the quality requirement structure provides a list of items required to quantitatively describe the quality requirements and a list of quality measures corresponding to the quality sub-characteristics, it is very smooth to refine the quality requirements if an appropriate measures can be found.
- Issues:
  - The set of measures provided in ISO/IEC 25022-25024 are not enough to find the right ones for this application. Especially for engineering purposes, measures for a function and component are needed, but such measure are very few.

VI. Related work

There are few reports on application of quality requirements standards to somewhat large and complex systems. Doerr et al. [10] reports their experience with using the ISO 9126 [13] and IEEE-830 [14] as quality requirements methods in three different settings, concluding that the methods led to more complete quality requirements. Jardim-Goncalves et al. [15] propose a test and evaluation framework to assess quality of ICT product in the architectural design, supported by the SQuaRE and Generalized Net.

Elicitation for of quality requirements is one of the most important issues [16]. Robertson et al. [17] address that use case is a good but not-always-useful method to elicit quality requirements because some quality requirements can be linked directly to a functional requirement, while some apply to the product. To help elicit quality requirements, they classified quality requirements into eight types: look and feel, usability and humanity, performance, operational, maintainability, security, cultural and political, and legal. Plosch et al. [18] propose an elicitation method for quality requirements using goal-oriented approach, which consist of four steps: identify goals, specify quality aspects, derive measurable factors and derive quality requirements. The quality requirements framework of ISO/IEC 25030 provides all the aspects which the above approaches have.

It is important to develop and update the quality requirements techniques in order to deal with new technologies. Noorwali et al. [19] propose an approach for specifying quality requirements in the context of big data. Knass et al. [20] propose a knowledge management framework for knowledge about quality requirements, so that a development team in agile can properly establish, share and maintain them. The quality requirements framework of ISO/IEC 25030 will be continuously updated so that it can be applicable to new technologies.

VII. Conclusion and Future work

Modern ICT systems like IoT systems should put more focus on their quality requirements. This paper provides the brief introduction of ISO/IEC 25000 (SQuaRE) series, which define quality models and measures, and how to define quality requirements and evaluate quality of the ICT products.

And then, the IoT systems' unique characteristics compared to the other information systems are mentioned, including many relevant stakeholders' involvement, device and network level heterogeneity and openness, resource constrained, spontaneous interaction, and increased security attack-surface, which may make development of the diverse applications and services a very challenging task.

To solve this problem, we apply the quality requirements framework of the ISO/IEC 25030 revision to an IoT system, Elderly monitoring system [1], and this paper fleshes out the contents and provide detailed discussion. The results of this application make us understand the usefulness and limitations (some issues to impede its smooth use) of the framework. The usefulness of the framework includes: the stakeholder categorization makes it easy to identify stakeholders; extracting the QIURs first, from which PQRs and DQRs are derived, makes their necessity and priority visible; the SQuaRE system hierarchy clarifies the target entities whose quality should be managed, and supports to derivate PQRs and DQRs recursively along with it; the quality requirement structure makes it smooth to quantitatively refine the quality requirements.

More application of the framework to a variety of IoT systems and much larger scale ones and some quantitative evaluation should be needed to ensure its usefulness and to clarify its limitations and problems.

References

[1] T. Nakajima. "Applying Quality Requirements Framework to an IoT System," The Fourth International Conference on Fundamentals and Advances in Software Systems Integration, Venice (Italy), September 16 - 20, 2018.

[2] ISO/IEC DIS 25030, Systems and Software engineering — Quality requirements framework.

[3] ISO/IEC 25010:2011, Systems and Software engineering — System and software quality models.

[4] ISO/IEC 25012:2008, Systems and Software engineering — Data quality model.

[5] ISO/IEC 25022:2016, Systems and Software engineering — Measurement of quality in use.

[6] ISO/IEC 25023:2016, Systems and Software engineering — Measurement of system and software product quality.

[7] ISO/IEC 25024:2015, Systems and Software engineering — Measurement of data quality.

[8] M. A. Razzaque, M. Milojevic-Jevric, A. Palade, and S. Clarke, "Middleware for internet of things, a survey," IEEE Internet of Things Journal, Vol. 3, No. 1, pp. 70-95, 2016.

[9] S. Okazaki et al., "An Intelligent Space System and its Communication Method to Achieve the Low Energy Consumption," IEEJ-C Vol. 136, No. 12, pp. 1804-1814, 2016 (in Japanese).

[10] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas. "DDoS in the IoT: Mirai and other botnets," Computer, Vol. 50, No. 7, pp. 80-84., 2017.

[11] E. Bertino and N. Islam. "Botnets and internet of things security," Computer, Vol. 50, No. 2, pp. 76-79, 2017.

[12] J. Doerr, D. Kerkow, T. Koenig, T. Olsson, and T. Suzuki. "Non-functional requirements in industry-three case studies adopting an

experience-based NFR method." 13th IEEE International Conference on Requirements Engineering (RE'05). IEEE, 2005.

[13] ISO/IEC 9126, Software engineering — Product quality -- Part 1: Quality model.

[14] IEEE 830-1998 - IEEE Recommended Practice for Software Requirements Specifications.

[15] R. Jardim-Goncalves, Ricardo, and V. Taseva. "Application of SQuaRE and Generalized Nets for extended validation of CE systems." 2009 IEEE International Technology Management Conference (ICE). IEEE, 2009.

[16] S. Ullah, M. Iqbal, and A. M. Khan. "A survey on issues in non-functional requirements elicitation." International Conference on Computer Networks and Information Technology. IEEE, 2011.

[17] S. Robertson, and J. Robertson. Mastering the requirements process: Getting requirements right. Addison-wesley, 2012.

[18] R. Plosch, A. Mayr, and C. Korner. "Collecting quality requirements using quality models and goals." 2010 Seventh International Conference on the Quality of Information and Communications Technology. IEEE, 2010.

[19] I. Noorwali, D. Arruda, and N. H. Madhavji. "Understanding quality requirements in the context of big data systems." Proceedings of the 2nd International Workshop on BIG Data Software Engineering. ACM, 2016.

[20] E. Knauss, G. Liebel, K. Schneider, J. Horkoff, and R. Kasauli. "Quality requirements in agile as a knowledge management problem: more than just-in-time." 2017 IEEE 25th International Requirements Engineering Conference Workshops (REW). IEEE, 2017.