# Towards a Software Sustainability-Quality Model: Insights from a Multi-Case Study

Nelly Condori-Fernandez
*Computer Science Department*
*Universidade da Coruña, Vrije Universiteit Amsterdam*
Coruña, Spain
Amsterdam, The Netherlands
n.condori.fernandez@udc.es, n.condori-fernandez@vu.nl

Patricia Lago
*Computer Science Department*
*Vrije Universiteit Amsterdam*
Amsterdam, The Netherlands
p.lago@vu.nl

***Abstract*—Background. Software sustainability is defined in terms of multiple and interdependent dimensions (economic, social, technical and environmental). Preliminary initiatives have investigated the contribution of certain quality attributes to sustainability dimensions.**

**Problem. Despite these valuable efforts, the characterization of software sustainability is still a key challenge. This entails how sustainability can be embraced in the design of software systems by identifying the relevant software quality attributes (QAs) and their dependencies. Both attributes and dependencies vary heavily with amongst others the type of software system and its operational context.**

**Aim and Method. We followed a multiple case study research method with the main objective ofinvestigating the applicability of the sustainability model in different contexts. We aim also to enrich our model, by means of identifying missing quality attributes or new contributions to the sustainability dimensions. We selected two software projects as cases of our study, where each one was independently conducted in specific situations.**

**Results. The results of the study show that the relevant quality requirements identified in both projects (cases) are covered by most of the QAs related to the social (82%) and technical (83%) dimensions. Moreover, some QAs that were not addressed in the corresponding projects, their relevance like context completeness, and flexibility were acknowledged. These results suggest that the software sustainability model could support the identification of relevant QAs. The case study also contributed to identify QAs that had not been considered in the economic, technical and social dimensions of the sustainability-quality model.**

***Index Terms*—Software quality, sustainability, multiple case study.**

## I. Introduction

Maintenance, evolution and adaptation can be extremely costly and painful for organizations due to the continuous and fast evolution of technology (e.g. paradigms, programming languages, etc.), which challenges the of software-intensive systems to be efficiently used in the future [1]. As a result, in recent years, there has been growing interest in understanding what sustainability means in the field of software engineering. In the literature, we can distinguish at least two distinct viewpoints:

978-1-7281-4844-1/19/18/$31.00 ©2019 IEEE

1) Sustainability IN software: it is concerned with the principles, practices, and processes that contribute to software endurability (e.g. [2], [3], [4])
2) Sustainability BY software: It is concerned with the achievement of sustainability goals by the help of software (e.g. [5], [6], [7]).

Both viewpoints are in line with software sustainability as defined by Lago et al. [8] and Venters et al. [9]. They agree on defining software sustainability in terms of multiple and interdependent dimensions (social, economic, technical and environmental). According to Lago et al. [8], these dimensions are defined as follows: The *economic dimension* aims to ensure that software-intensive systems can create economic value. The *social dimension* focuses on ensuring current and future generations have the same or greater access to social resources by pursuing generational equity. The *environmental dimension* seeks to avoid that software-intensive systems harm the environment they operate in. And, the *technical dimension* is concerned with supporting long-term use and appropriate evolution/adaptation of software-intensive systems in constantly changing execution environment.

Based on this multidimensional nature of sustainability, current efforts have been put in the definition of a Software Sustainability Model (e.g. [10], [11], [12], [13]). With the purpose of characterizing each sustainability dimension, Condori-Fernandez and Lago. [14] investigated the contribution and relevance of quality attributes from different perspectives. Given a quality attribute can contribute to one or more sustainability dimensions, preliminary direct dependencies among sustainability dimensions were also identified [14]. As the relevance and contribution of these quality attributes are very context-dependent, in this paper we propose an holistic and exploratory multiple case study with the main objective of investigating the applicability of our preliminary sustainability model in different contexts. To that end, we selected two industrial cases from different domains that cover the two sustainability viewpoints discussed above(sustainability IN/BY software).

The contribution of this work is twofold. First, it offers insights regarding the applicability of the sustainability model in different contexts. Second, it generates new insights into the characterization of the sustainability dimensions, by means of identifying missing relevant quality attributes and corresponding relations among dimensions.

The rest of the paper is organized as follows. Section 2 introduces the exploratory case study, which includes objective, research questions, data collection techniques, and data analysis. Then we present the results and summarize the main findings in section 3. Section 4 discusses threats to validity of the cases studies. Section 5 presents the related works. Finally, Section 6 concludes the paper.

## II. Research method: Case study

We used an holistic multiple-case study ( [15], [16]) in order to gain insights for investigating the applicability of a sustainability model for software-intensive systems. Figure 1 shows the main activities that compose the process to perform the multiple case study. These activities are: plan design, data collection, data analysis, and reporting results.

### A. Plan design

*1) Objective and research questions:* The **objective** of our multiple case study approach is to explore the applicability of the sustainability model for identifying relevant quality requirements in different contexts (i.e. organization, type of project and domain).

Moreover, as a consequence of using the model in multiple cases, an iterative improvement of the sustainability model is expected, by adding new quality attributes or extending the definition of the existing ones due to their actual contribution to other sustainability dimensions that were not originally considered.

To achieve this objective, the underlying **research question** is formulated as follows:

(RQ) *"How applicable is the sustainability model in practice for identifying the relevant sustainability-quality requirements?"*

In the context of our study, applicability is investigated by analyzing if the quality attributes (QAs) of the model do cover the relevant sustainability-quality requirements of the software project at hand.

*2) Cases and Units of Analysis:* In this paper, we investigated two different cases, which were selected by convenience. The companies of both cases were selected from the contact network of the researchers.

Regarding **Case A**, the investigation focused on a project evolving a pre-existing *Customer Relationship Management (CRM) system*. It involved the employees in charge of the project, i.e. business analysts, functional administrators, a project manager, an architect and business managers. The start of the implementation of the CRM system was five years ago. The system was part of a bigger project, aimed at making the whole organization more customer-driven (as opposed to product-driven).

Regarding **Case B**, the investigation focused on devising a new software solution required for reducing *food waste* within a Dutch Airline company. Particularly, such software solution should contribute to improve the business processes from the catering companies, cabin crew and planning department. The goal of this new software system is food waste-reduction and increasing user friendliness.

For each case, Table I summarizes the related context and unit of analysis.

The cases are adequate to investigate whether our sustainability model can encompass a broad range of quality aspects of the respective software system. Through the application of our model, by means of a multiple case study research method, we are going also to be able to identify missing quality requirements, detect incorrect dependencies (contributions), and confirm the contribution of those qualities to each sustainability dimension identified in [14]. In particular, the selected cases allow us to investigate sustainability from two complementary angles:

1) Sustainability *in* software, where the sustainability impact can be observed in the software system itself (e.g. energy efficiency). Case A mainly targets this level of sustainability, since a key objective of the resulting software is to be itself better maintainable.
2) Sustainability *by* software, where the sustainability impact can be observed in the processes supported by the software system. Case B mainly targets this level of sustainability, since a key objective of the resulting software is to help reduce food waste.

### B. Data collection

As shown in Figure 1, we used various data collection techniques (see blue arrows in Figure 1). In particular, for Case A the data was collected by using three different techniques (document analysis, user test, and interviews) then combined, whereas A/B testing and interviews were the techniques used for Case B.

*1) Data collection techniques used in Case A:* **Document Analysis.** The data gathered from the analysis of internal documents was obtained in several iterations in order to understand (i) the organizational goals that motivated the evolution of the pre-existing software system (i.e. CRM system); (ii) the functional and non-functional requirements; and (iii) the design of the software system. One of the authors spent about two weeks on the analysis of the documentation provided by the company. This analysis mostly served to prepare the first iteration of interviews with the team members involved in the project.

**User Testing.** In order to identify which requirements were addressed in the CRM system, a user test was performed with one business analyst from the company, who did not have any previous experience in using the CRM system while having an interaction with a customer. Thanks to this lack of experience, the user was able to pay extra attention to certain quality aspects of the system. The test took around 40 minutes. The results of the user testing helped to prepare the second iteration of interviews.

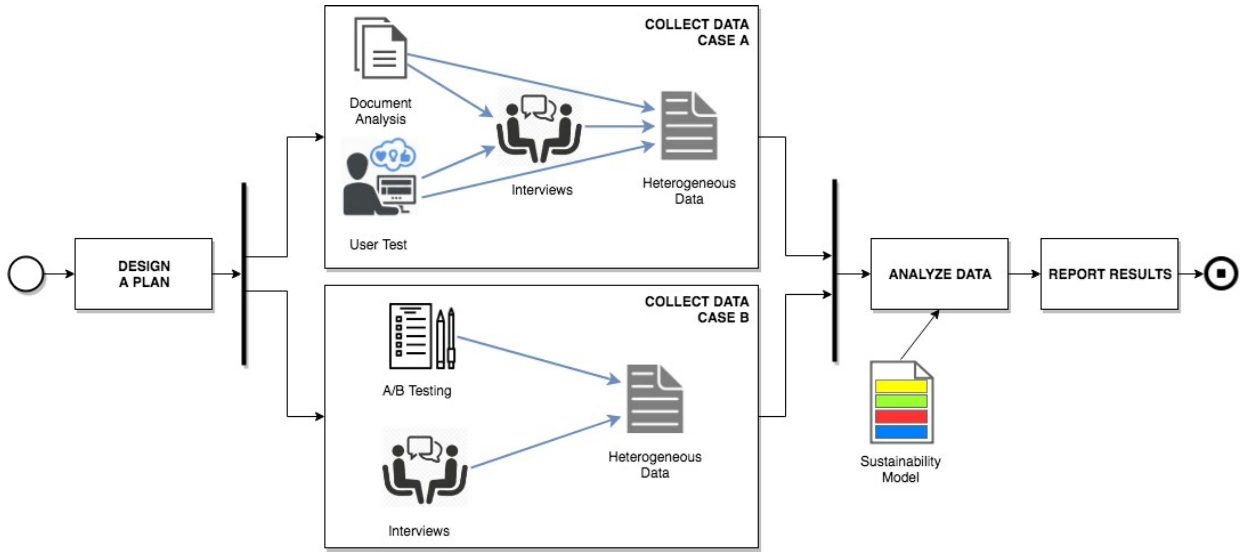


Fig. 1. Process of the multiple Case Study approach

TABLE I
CASES, CONTEXT, AND UNIT OF ANALYSIS

| | **Context** | **Sw system** | **Domain** | **Sust. angle** |
|---|---|---|---|---|
| **Case A** | The organization is one of the world leading providers of insurances, asset management and pensions, with more than 170 years of business experience. It counts about 28,000 employees worldwide. | Pre-existing | Customer Relationship Management (CRM) | IN |
| **Case B** | The organization is a multinational leader in end-to-end IT and business consulting services. It counts over 72,000 employees worldwide. | New | Food Waste Management | BY |

**Interviews.** The semi-structured interviews served the purpose of understanding how stakeholders interpret software sustainability in terms of quality requirements.

In Case A, semi-structured interviews were carried out with nine employees who were involved in the implementation of the pre-existing CRM system. The duration of each interview was between 30 and 60 minutes. Interviewees were provided with a shortlist of quality requirements identified by Condori-Fernandez and Lago [14]. This preliminary shortlist was made with the results found from the document analysis and user tests. Therefore, the purpose of the interviews was mainly confirmatory in nature.

*2) Data collection techniques used in Case B:* **Interviews.** Several stakeholders with different backgrounds were involved. The data gathered from *cabin crew members* with 3 or more years of experience allowed us to understand the behaviour of passengers on different types of flights and their buying habits; the data gathered from *managers from catering companies* provided us with an overview of the requirements that should be considered from the side of these catering companies, and hence gain more information about the feasibility of the proposed solution. The duration of the interviews oscillated between one to two hours.

All interviews were conducted face-to-face in The Netherlands between February and May 2018.

**A/B testing.**As a prototype app was implemented, it was used for validating the requirements of the project by means of A/B testing. This technique is conducted under experimental conditions, where two or more variants of a software product are shown to users at random. A/B testing is used to determine which variation performs better for a given goal. In Case B, two variations of the created prototype app for passenger and cabin crew were used. For instance, variant A showed the products by displaying them as tiles with 2 products per row, where variant B displayed them as a list/table. As part of the A/B tests, an online questionnaire was designed and conducted with the participation of practitioners from different areas of expertise (e.g., aviation, data analytics). The questionnaire took about 5 minutes.

### C. Data analysis

In this subsection, we present the data analysis carried out for answering our research question. As shown in Figure 2, researchers in collaboration with stakeholders identified the quality requirements from the collected data. Then, stakeholders related the identified qualities to the sustainability dimensions (i.e., economic, technical, environmental and social), which could contribute to identify new quality requirements for the project. Finally, these quality requirements were mapped on the corresponding quality attributes of the sustainability model [17].

Next, we list the quality requirements considered in each case.

*1) Case A: Customer Relationship Management:* According to the data found from the **document analysis**, the selection of possible CRM systems available on the market was based on three fundamental aspects: i) a centralized customer-view, ii) CRM case management, where cases are issues associated with a customer account, and iii) availability of the CRM system. Among the documented quality requirements that were derived from these three aspects are: functional suitability, interoperability, integration, performance, maintainability and availability. The best CRM system fulfilling these requirements was chosen.

Through the **user tests**, additional quality requirements like usability were found. For instance, users considered that the selected CRM system is appropriate for their needs, and it is easy to use. Moreover, as the CRM system is a cloud-based solution, we found security, availability, portability and satisfaction requirements as relevant, too.

Through the **interviews**, we found a list of quality requirements that were considered as relevant to contribute to the social, technical, economic, and environmental sustainability dimensions (see Table II). The interviewees where also asked to prioritize the quality requirements based on how important this requirement was in the selection of the CRM system and its development phase. The five most frequent quality requirements where (ranked from top to bottom) maintainability, usability, compatibility, security and functional suitability.

*2) Case B: Food waste management:* In contrast to Case A, most of quality requirements were identified through **(group and face-to-face) interviews** conducted with personnel of the airline company. The purpose of the interviews was to elicit requirements of the new software system to be developed. For instance, usability, availability, and fault tolerance were considered as the most important for responsible of the cabin crew. Other requirements considered also as relevant were: scalability, compatibility, performance, privacy, and security. And user satisfaction was tested by means of A/B tests.

Table II shows the list of quality requirements that were identified in the corresponding project and considered as contributors to the different sustainability dimensions from the stakeholders perspective.

In the next section, we present the results of mapping the list quality requirements (Table II) to the quality attributes of the sustainability model [17].

## III. Results

As a result of the mapping (See Figure 3), we focus on the findings related to (i) the applicability of the model for identifying quality requirements in software projects; and (ii) the enrichment of the model by identifying other relevant QAs to the sustainability dimensions as well as their dependencies.

### *A. Applicability of the Sustainability-quality model*

According to Figure 3, for the applicability of the sustainability model we highlight two type of findings:

- **QAs covered:** QRs of the software projects (Case A and Case B) that were *already covered* by the QAs of the model are shown in the Appendix, Tables III, IV, VI, and V (see QAs marked by the symbol ✓and listed above the line).
- **QAs discovered:** The QAs marked by the symbol ‡ represent those qualities that were not addressed in the corresponding projects but were identified as relevant after using the Sustainability-quality model.

In the following, both findings are discussed per dimension, by highlighting some of the qualities of the model considered as important requirements in the corresponding projects:

*1) Environmental sustainability:* Similar to economic dimension, few QAs of the sustainability-quality model(Table V) were confirmed as relevant environmental sustainability-quality requirements (i.e., reusability, modifiability, resource utilization, time behaviour).

*2) Economic sustainability:* In comparison to social and technical dimensions, several QAs of the economic dimension shown in Table VI were not considered as relevant by stakeholders of both cases (e.g. satisfaction, reliability, economic risk mitigation, functional completeness). It is possible, therefore, that the economic sustainability dimension of our model requires more validation, specially with those QAs that were not considered as relevant as well as the new ones added to the model (see next subsection).
The lack of stability in both economic and environmental dimensions can be due to the lack of competences of ICT experts in sustainability. Social and technical implications of software systems is a much more consolidated practice than the competences related to economic and environmental sustainability, which was also reflected in [14].

### *B. Enrichment of the Sustainability-quality model*

In this sub-section, we discuss i) the new QAs that were *added to the model*; and ii) the extension of QAs's contributions (direct dependencies) to other dimensions not detected in the original model (see QAs listed below the line in the same tables III, IV, V, VI).

*1) New QAs in the Model:*

- **Data Privacy** According to Barker et al. [18], privacy concerns arise wherever personally identifiable information is collected, stored, or used. In case B, as the app must handle private user data for product orders, data privacy was a requirement to be addressed in the project. Given the contribution of this quality to the social sustainability, Data privacy was also added to the sustainability model.

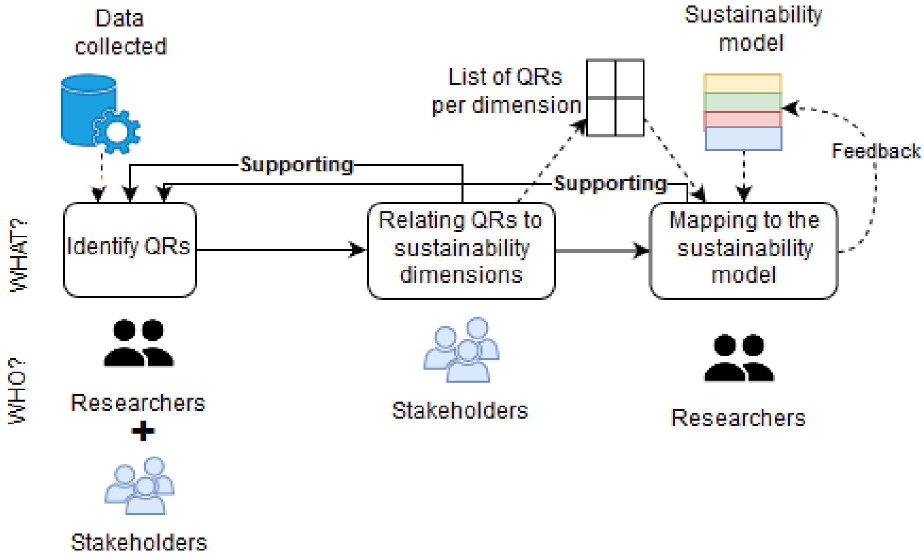


Fig. 2. Data Analysis carried out per case study

TABLE II
CONTRIBUTION OF QUALITY REQUIREMENTS TO EACH SUSTAINABILITY DIMENSION FROM STAKEHOLDERS PERSPECTIVE

| Dimension | Case A: Sustainability IN software | Case B: Sustainability BY software |
|---|---|---|
| **Technical** | Functional Suitability, compatibility, effectiveness, maintainability | Compatibility, reliability, maintainability, functional suitability |
| **Social** | Security, effectiveness, usability, compatibility | Security, satisfaction, effectiveness, usability, persuasiveness, data privacy |
| **Environmenal** | Maintainability | Resource utilization, environmental risk mitigation, performance efficiency |
| **Economic** | Effectiveness, functional suitability, replaceability, compatibility, maintainability, usability, portability, capacity | Effectiveness, functional suitability, efficiency, automatable. |

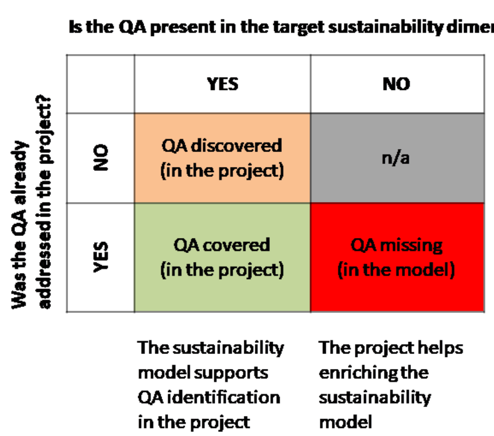


Fig. 3. Main findings from the multiple case study

- **Persuasiveness** In case B, designers aim to persuade end-users to pre-order products before their flight. Thus designing a persuasive mobile app, customer experience is expected to increase by enabling the passengers to order food and beverages upfront with a discount. By doing so, the passengers are sure that their product will be available during the flight, hence improving the current service offering. Accordingly, thanks to its positive impact to the social dimension, we added this QA to the sustainability model by defining persuasiveness as the ability of the software system to persuade its users to take action.
- **Automatable**
The automation of business processes can contribute to the improvement of economic, environmental and social sustainability dimensions in longer term. For instance in Case B, the original business process of an airline company has a standard trolley inventory that is loaded on each plane before flight. However, as the goal is to reduce the food waste, it is no longer possible to serve only standard equipped food trolleys. The business process on the side of the catering companies needs to change.
Although this initially can have a negative economic impact for the organization, the implementation of a predictive data analytics module in the system can contribute to achieve the goal of food waste reduction, by calculating what products and how many of them should be loaded

onto each plane.

- **Replaceability** According to the ISO standard, replaceability is defined as the degree to which a product can be replaced by another specified software product for the same purpose in the same environment. In Case A, this quality was considered as relevant to the *technical* and *economical dimension*. The reasoning of the interviewees was that if the maintenance of a system become so expensive, and harder to fit with the newest systems in the future, replacing the software system that is part of a system of systems can contribute to the sustainability of such systems of systems. Therefore, from this perspective, replaceability could fit to the technical and economic dimensions if we target to replace a software system that is part of a system of software systems.
- **Scalability** This QA was addressed in Case B. The new proposed solution fits in the target architecture, which relies on API-led connectivity. It is an architecture pattern designed for microservices. Although, currently most of the modules to external services are already in place (e.g. the booking- and crew systems), scalability of the system is needed for enabling the connection between the catering companies.

*2) Extending the contributions of QAs to other dimensions:* In comparison to our original model, we found that the contribution of certain QAs to sustainability could be extended to other dimensions that had not been considered (by respondents of a survey [14]). For example, integrity could also contribute to the technical dimension because thanks to this QA the system can be able to prevent (in certain extent) unauthorized access to the system and prevent that the software system cannot be compromised for example in terms of their functionality or availability.

Other QAs such as compatibility (coexistence), usability (user error protection and learnability), maintainability (modifiability), portability (adaptability) and performance efficiency (Capacity) were identified also as good contributors to the economic dimension.

In order to visualize all the **direct dependencies** between dimensions identified through the case studies, we use a Venn diagram notation (See figure 4). It shows the QAs that belong to the corresponding intersection areas, where a sustainability dimension is defined as a set of sustainability quality attributes. For example, according to the right Venn Diagram shown in Figure 4, the direct dependency between Technical and Economic dimensions consists of seven new ordered pairs whose qualities attributes are: modifiability (A22), coexistence(A16), interoperability (A18), adaptability (A25), capacity (A33), scalability (B4) and replaceability (B3).

Next, we explain the new contributions of these QAs:

- **Security** The relevance of security to social sustainability was confirmed in both cases. However, security was also considered as relevant to the **technical dimension** since the system can be threatened if security is not adequately addressed (Integrity). Therefore, securing the system through the implementation of any mechanism that enables a long-term use of any software-intensive system is also technical, contribution that had not been considered in the original model.
- **Compatibility** In both cases, compatibility (i.e. interoperability and coexistence) was also related to the **economic dimension**. For example, in Case A, before the CRM system was implemented, there was no a central point where all products of a customer could be found. When a customer needed to change her/his address, he or she had to notify all the different business lines separately. Now with the CRM system it is possible that with just one notification, all the connected systems are informed that the new customer´s address need to be updated. As this creates economic value for the organization, both compatibility attributes (A16, A18) resulted as relevant to the **economic dimension**, which had not been identified in the original model.
- **Usability** We found that some of the usability attributes are related not only to the social dimension but also to the economic dimension. The reasoning is that making the system usable (e.g. easy to learn), users will be able to perform their tasks faster, which in turn can help organizations to decrease costs and thus create an economic value, too. For instance, in Case A, the average handle-time of a customer question will decrease since the system can really help in getting the information without errors (user error protection). The quality experience of customers will increase as well, with a positive impact to both **social and economic** sustainability dimensions. The new dependencies correspond to learnability (A13) and User error protection (A12) attributes, which are located in the intersection areas between social and economic dimensions.
- **Maintainability** Maintaining the system is also creating economic value in the organization. If, however, a system becomes a legacy system, the modification of software usually becomes more expensive (e.g. to find the suitable resources to maintain it, lack of compatibility with new systems). This implies a negative impact to the economic and environmental dimensions. Therefore, modifiability (A22) that had been considered as contributor to technical and economic dimensions, now it is extended to the **environmental dimension** as well (See the right diagram in Figure 4.
- **Adaptability** By means of the first case study (Case A), this attribute was acknowledged as good contributor not only to the technical dimension but also to the economic dimension. As the CRM program is a software-as-a-service solution, it is important that the program can effectively and efficiently be adapted not only for different hardware or software, but also for different usage environments (e.g. noisy environments, customers speaking in foreign language, etc.). So practitioners considered that the implementation of adaptation capabilities is technical, but also economic because it also might imply higher costs to the organization. Although we agree

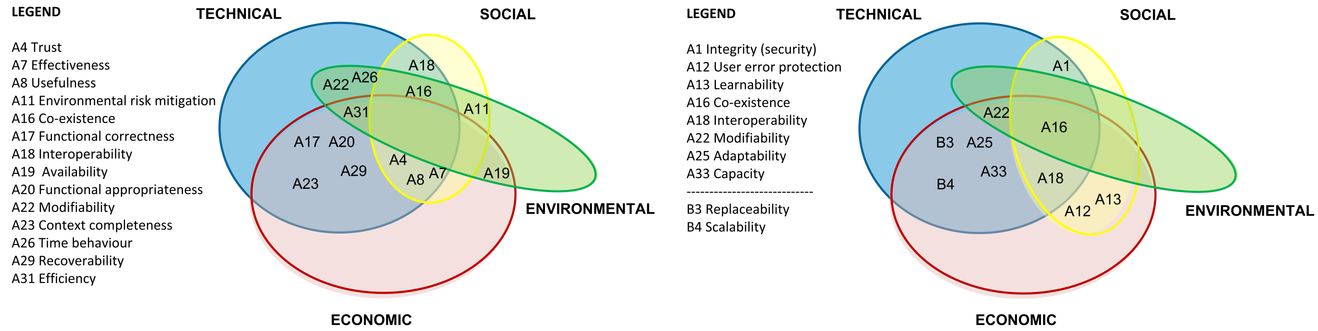


Fig. 4. Direct dependencies identified in the original sustainability model (left), new direct dependencies identified from the cases studies (right)

on such economic implication, we think that adaptability (as technical sustainability requirement) could be related directly to the economic risk mitigation attribute from the economic dimension. However, as this attribute was not considered by any participant of both cases, we decided consider this potential extension for adaptability (A25) as contributor to the **economic dimension** as well, but it still needs more studies to be confirmed.

- **Capacity**
  Practitioners of Case A (CRM project) acknowledged the relevance of this QA to the economic dimension. As in our original model, capacity was only identified as contributor of the technical dimension, we asked the practitioners some examples of situations to verify how capacity was being understood. Some of these examples are: ”If the number of users are exceeding the number of licenses that are bought, there is no option to let more users use the system or the organization needs to invest in the licenses. Moreover, If the servers running the system are not having enough capacity, the system will slow down, which will result in more time needed for answering questions and providing the customer with information”; ”When a system is running out of capacity, the organization needs to buy extra resources. Creating extra capacity is not possible without buying more resources, like licenses.” As a result of these examples, we conclude that whether capacity needs to be addressed, the economic value of the organization will be also impacted. Therefore, although this QA was not acknowledged as relevant for the technical dimension, we decided to extend the contribution of capacity to the economic dimension. As we keep the contribution to the technical dimension, capacity (A33) is located in the intersection between the technical and economic dimensions, represented as sets in Figure 4.

Scalability (B3) and Replaceability(B4), as we discussed in the previous subsection, both attributes were not present in the original model. And as their corresponding contributions were acknowledged to the **economic and technical** dimensions, both are represented also in Figure 4.

## IV. Threats to Validity

In the following, we discuss the validity threats of our case studies regarding reliability, internal, and external validity [19], [20].

*1) Reliability:* It is related to the repeatability of the study. As the execution of the case study was carried out only by one (junior)researcher, the data collection could have been threatened due to the lack of experience of the researcher. To mitigate this threat, both validation and execution activities were always guided by one senior researcher from the university. Moreover, thanks to the several meetings with stakeholders, inconsistencies and misinterpretations were avoided as much as possible.

*2) Internal validity:* It is related to factors that researchers are unaware of, or cannot control regarding their effect on the variables under investigation. To mitigate the threats related to investigator bias, at least two researchers were involved during the design of the instrumentation. To mitigate threats related to the interviewee bias (e.g. getting false positive answers), people with different roles were interviewed and involved in user tests, which facilitated us to identify possible false positive answers.

*3) Construct validity:* It reflects how well the measures used do represent the constructs the study intends to measure. In this study, the applicability of the sustainability model is investigated with respect to the QRs identified as relevant in a project. The identification of these requirements is crucial for determining the coverage of the sustainability model. To reduce the threat of missing relevant QRs in the projects, we used different data collection techniques (interviews, document analysis, users tests, A/B tests). This triangulation by method(e.g. interviews, user tests) and data source (data collected from different people) allowed us to reduce this threat.

*4) External validity:* It is concerned with the generalization of the findings. Since we employed the case study research method, the findings are bound by the context of the selected cases [21]. To slightly reduce this threat, we carefully selected two cases in different companies and different domains, which covered also a different sustainability angle (IN, BY). However, replications of each single-case study is definitively needed for increasing the external validity of our research.

## V. Related Work

Being able to assess the impact of quality requirements on sustainability is the first step towards developing software-intensive systems that fulfill sustainability concerns *by design* [14]. The assessment based on the notion of sustainability as a software quality requirement, however, is to date poorly addressed in the literature.

From a purely technical perspective, software sustainability maps on more traditional quality requirements and has been linked, for example, to the notion of longevity [22], stability [23], and evolvability [4]. In our work we cover this perspective by identifying the quality attributes (or aspects thereof) falling under one of the possible sustainability dimensions – namely, technical sustainability.

Through an analysis of the literature, Venters et al. [10] and later on Wolfram et al. [24] discussed the notion of software sustainability. From both analyses it emerges that sustainability is a multi-faceted concept and argue for a quantitative approach [10] that makes explicit the extent to which sustainability concerns are addressed in engineering software systems [24].

Although a dedicated literature exists on approaches for dealing with quality requirements (e.g. [25], [26], [27], [28]), there are still a lack of agreement regarding their definition and classification [29].

Our sustainability model sets a first step in this direction, by offering a toolkit of quality attributes classified along the four sustainability dimensions. To the best of our knowledge there are no sustainability quality models in the literature yet that, similar to ours, address sustainability from a product point of view.

The only exception is the work of Calero et al. [11]: based on the ISO/IEC 25010 Standard, the authors provide a preliminary discussion of which quality characteristics should be considered in addressing software sustainability. As a next step, they propose the definition of a quality model where sustainability is part of the quality of software products. In contrast to our work, Calero et al. defined sustainability only in terms of energy consumption, resource optimization and perdurability (reusability, modifiability, and adaptability). These three quality attributes would map on the environmental (for energy consumption) and technical (for resource optimization and perdurability) sustainability dimensions. Moreover, our work shows that the same quality attribute (e.g. adaptability) does not necessarily impact a single sustainability dimension.

Some research works have addressed sustainability from the perspective of the development process, or even the quality of the whole organization.

Lautenschutz et al. [30] compared and contrasted various Green ICT maturity models. Inspired by the CMM, Hankel and Lago [31] defined the SURF Green ICT Maturity Model (SGIMM) with the aim to assess the maturity of overall organizations with respect to Green ICT. To this aim, the SGIMM includes criteria in four areas, including *greening of ICT* and *greening of the primary processes*. In terms of software systems, these correspond to software energy efficiency and software energy awareness, respectively, which would be mapped on both technical- and environmental sustainability.

Hankel et al. [32] collected factors of influence regarding the environmental impact of ICT, and used them to create a taxonomy, which in turn can be used for quantitative quality evaluations. This work builds upon a pre-existing framework defined by Hilty [3]. This combines the possible orders of impact of Green ICT (namely, direct, indirect and systemic [33]) with whether the environmental impact is positive (part of solution) or negative (part of problem) into a matrix where specific effects are described.

In a similar vein, the GREENSOFT model proposed by Naumann et al. [13] was designed to incorporate quality requirements along the above-mentioned orders of effects. It remains, however, at a mostly process-oriented and very high-level of abstraction, without explicit reference to specific quality attributes.

## VI. Conclusion

In this paper, we investigated the applicability of a sustainability model with the purpose of i) understanding how the model supports QA identification; and ii) enriching the sustainability model through a characterization of the sustainability dimensions in terms of relevant quality attributes. To do so, we carried out a multiple-case study, where two software projects from different domains and developed at different companies, were selected by convenience.

The results of this study show that the relevant quality requirements identified in both projects are covered by most of the QAs related to the social (82%) and technical (83%) dimensions, whereas QAs regarding economic (50%) and environmental (38%) dimensions were partially covered. Moreover, despite some QAs of the model were not addressed in the corresponding projects (e.g. context completeness, flexibility), their relevance was acknowledged. These findings suggest that the software sustainability model could support software engineers in the identification of relevant QAs. In fact, practitioners found the sustainability model as very useful not only for identifying requirements but also for prioritizing requirements, by giving higher importance to those that contribute to different sustainability dimensions (e.g. integrity, user error protection, learnability, interoperability, modifiability, adaptability, capacity, co-existence). The relevance of the different dimensions, however, depends on the type of software system [34] or industry sector.

As a consequence of applying the model in both cases, another interesting finding is that the selected projects (cases) helped enriching our sustainability model, by means of:

1) identifying QAs that had not been considered as relevant to the respective dimensions. The results show that most of the new QAs were added to the economic dimension (e.g. automatable, scalability capacity), followed by the technical (integrity, replaceability, scalability) and the social dimension (i.e. data privacy and persuasiveness).
2) extending the definition of some QAs because new relations (direct dependencies) with other dimensions were

identified. Some of these QAs are security (related also to the technical dimension), maintainability, usability, compatibility (related also to the economic dimension).

We consider that more case studies investigating the applicability of the sustainability model in a holistic way are needed for identifying QAs as well as for uncovering the related cross-dependencies. In this paper, we focused only on direct dependencies identified among sustainability dimensions. As a future work, we are going to extend our model with other type of dependencies that can be identified by means of a nichesourcing approach [35].

## ACKNOWLEDGMENT

The authors are very thankful to the employees of the two Dutch companies involved in this research, as well as to the anonymous reviewers for their valuable feedback. We want also to thank our Master students Martine Reijm and Mathijs Oggel for their valuable help in the data collection process of this research.

The research presented in this paper is partially funded by the Spanish Ministry of Economy, Industry and Competitiveness with the project: TIN2016-78011-C4-1-R.

## REFERENCES

[1] A. Jansen, A. Wall, and R. Weiss, "Techsure-a method for assessing technology sustainability in long lived software intensive systems," in *Software Engineering and Advanced Applications (SEAA), 2011 37th EUROMICRO Conference on*. IEEE, 2011, pp. 426–434.

[2] S. Hettrick, "Research software sustainability: Report on a knowledge exchange workshop," 2016.

[3] L. M. Hilty and B. Aebischer, "ICT for sustainability: An emerging research field," in *ICT Innovations for Sustainability*. Springer, 2015, pp. 3–36.

[4] H. Koziolek, "Sustainability evaluation of software architectures: A systematic review," in *Proceedings of the Joint ACM SIGSOFT Conference – QoSA and ACM SIGSOFT Symposium – ISARCS on Quality of Software Architectures – QoSA and Architecting Critical Systems – ISARCS*, ser. QoSA-ISARCS. New York, NY, USA: ACM, 2011, pp. 3–12.

[5] A. Andersson, L. W. Hiselius, and E. Adell, "Promoting sustainable travel behaviour through the use of smartphone applications: A review and development of a conceptual model," *Travel Behaviour and Society*, vol. 11, pp. 52 – 61, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2214367X17301643

[6] N. Condori Fernandez and P. Lago, "The influence of green strategies design onto quality requirements prioritization," in *Requirements Engineering: Foundation for Software Quality*, E. Kamsties, J. Horkoff, and F. Dalpiaz, Eds. Cham: Springer International Publishing, 2018, pp. 189–205.

[7] F. Suni Lopez and N. Condori-Fernandez, "Design of an adaptive persuasive mobile application for stimulating the medication adherence," in *Intelligent Technologies for Interactive Entertainment*, R. Poppe, J.-J. Meyer, R. Veltkamp, and M. Dastani, Eds. Cham: Springer International Publishing, 2017, pp. 99–105.

[8] P. Lago, S. A. Koçak, I. Crnkovic, and B. Penzenstadler, "Framing sustainability as a property of software quality," *Communications of the ACM*, vol. 58, no. 10, pp. 70–78, 2015.

[9] C. Venters, L. Lau, M. Griffiths, V. Holmes, R. Ward, C. Jay, C. Dibsdale, and J. Xu, "The blind men and the elephant: Towards an empirical evaluation framework for software sustainability," *Journal of Open Research Software*, vol. 2, no. 1, 2014.

[10] C. C. Venters, C. Jay, L. Lau, M. K. Griffiths, V. Holmes, R. Ward, J. Austin, C. E. Dibsdale, and J. Xu, "Software sustainability: The modern tower of babel," *RE4SuSy: International Workshop on Requirements Engineering for Sustainable Systems*, 2014.

[11] C. Calero, M. A. Moraga, and M. F. Bertoa, "Towards a software product sustainability model," *CoRR*, vol. abs/1309.1640, 2013. [Online]. Available: http://arxiv.org/abs/1309.1640

[12] B. Penzenstadler and H. Femmer, "A generic model for sustainability with process-and product-specific instances," in *Proceedings of the 2013 workshop on Green in/by software engineering*. ACM, 2013, pp. 3–8.

[13] S. Naumann, M. Dick, E. Kern, and T. Johann, "The greensoft model: A reference model for green and sustainable software and its engineering," *Sustainable Computing: Informatics and Systems*, vol. 1, no. 4, pp. 294–304, 2011.

[14] N. Condori-Fernandez and P. Lago, "Characterizing the contribution of quality requirements to software sustainability," *Journal of systems and software*, vol. 137, pp. 289–305, Mar. 2018.

[15] F. Shull, J. Singer, and D. I. K. Sjøberg, Eds., *Guide to Advanced Empirical Software Engineering*, 2008. [Online]. Available: https://link.springer.com/book/10.1007/978-1-84800-044-5

[16] J. M. Verner, J. Sampson, V. Tosic, N. A. Abu Bakar, and B. A. Kitchenham, "Guidelines for industrially-based multiple case studies in software engineering," in *in Research Challenges in Information Science, 2009. RCIS 2009. Third International Conference on*, 2009. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.377.7021

[17] N. Condori Fernandez and P. Lago, *A Sustainability-quality Model: (version 1.0)*. VU Technical Report, November 2018.

[18] K. Barker, M. Askari, M. Banerjee, K. Ghazinour, B. Mackas, M. Majedi, S. Pun, and A. Williams, "A data privacy taxonomy," in *Dataspace: The Final Frontier*, A. P. Sexton, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 42–54.

[19] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering*, vol. 14, no. 2, p. 131, Dec 2008. [Online]. Available: https://doi.org/10.1007/s10664-008-9102-8

[20] R. Wieringa, "Towards a unified checklist for empirical research in software engineering: first proposal," in *16th International Conference on Evaluation & Assessment in Software Engineering, EASE 2012, Ciudad Real, Spain, May 14-15, 2012. Proceedings*, 2012, pp. 161–165. [Online]. Available: https://doi.org/10.1049/ic.2012.0020

[21] S. Ghaisas, P. Rose, M. Daneva, K. Sikkel, and R. J. Wieringa, "Generalizing by similarity: Lessons learnt from industrial case studies," in *Proceedings of the 1st international workshop on conducting empirical studies in industry*. IEEE Press, 2013, pp. 37–42.

[22] P. Avgeriou, M. Stal, and R. Hilliard, "Architecture sustainability [guest editors' introduction]," *Software, IEEE*, vol. 30, no. 6, pp. 40–44, 2013.

[23] M. Salama and R. Bahsoon, "Analysing and modelling runtime architectural stability for self-adaptive software," *The Journal of systems and software*, vol. 133, pp. 95–112, Nov. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0164121217301620

[24] N. J. Wolfram, P. Lago, and F. Osborne, "Sustainability in software engineering," in *The Fifth IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT)*, Dec. 2017, p. 7.

[25] 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," in *13th IEEE International Conference on Requirements Engineering (RE'05)*, Aug 2005, pp. 373–382.

[26] M. Khurum, T. Gorschek, and M. Wilson, "The software value map — an exhaustive collection of value aspects for the development of software intensive products," *Journal of Software: Evolution and Process*, vol. 25, no. 7, pp. 711–741, 2013. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/smr.1560

[27] M. Galster and E. Bucherer, "A taxonomy for identifying and specifying non-functional requirements in service-oriented development," in *2008 IEEE Congress on Services - Part I*, July 2008, pp. 345–352.

[28] N. Condori-Fernandez and P. Lago, "Can we know upfront how to prioritize quality requirements?" in *2015 IEEE Fifth International Workshop on Empirical Requirements Engineering (EmpiRE)*, Aug 2015, pp. 33–40.

[29] M. M. Hasan, P. Loucopoulos, and M. Nikolaidou, "Classification and qualitative analysis of non-functional requirements approaches," in *Enterprise, Business-Process and Information Systems Modeling*, I. Bider, K. Gaaloul, J. Krogstie, S. Nurcan, H. A. Proper, R. Schmidt, and P. Soffer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 348–362.

[30] D. Lautenschutz, S. Espana, A. Hankel, S. Overbeek, and P. Lago, "A comparative analysis of green ICT maturity models," in *International Conference on ICT for Sustainability*, ser. ICT4S, 2018.

[31] A. Hankel, L. Oud, M. Saan, and P. Lago, “A maturity model for green ICT: The case of the SURF green ICT maturity model,” in *28th International Conference on Informatics for Environmental Protection (EnviroInfo)*. BIS Verlag, 2014, pp. 33–40.

[32] A. Hankel, G. Heimeriks, and P. Lago, “A systematic literature review of the factors of influence on the environmental impact of ICT,” *Technologies*, vol. 6, no. 3, p. 85, Sep. 2018.

[33] A. Köhler and L. Erdmann, “Expected environmental impacts of pervasive computing,” *Human and Ecological Risk Assessment: An International Journal*, vol. 10, no. 5, pp. 831–852, Oct. 2004.

[34] P. Lago, “Architecture design decision maps for software sustainability,” in *IEEE/ACM International Conference on Software Engineering: Software Engineering in Society Track*, ser. ICSE SEIS, 2019.

[35] N. Condori-Fernandez, P. Lago, M. Luaces, and A. Catala, “A nichesourcing framework applied to software sustainability requirements,” in *IEEE 13th International Conference on Research Challenges in Information Science*. IEEE, 2019.

## APPENDIX

TABLE III
RELEVANT SOCIAL SUSTAINABILITY- QUALITY REQUIREMENTS

| **Characteristic** | **Attribute** | **Case A** | **Case B** |
|---|---|---|---|
| Security | Confidentiality | ✓ | ✓ |
| Security | Authenticity | ✓ | ✓ |
| Security | Accountability | ✓ | ✓ |
| Satisfaction | Trust | ✓ | ✓ |
| Freedom from risk | Health and safety risk mitigation | | |
| Security | Integrity | ✓ | ✓ |
| Effectiveness | Effectiveness | ‡ | ‡ |
| Satisfaction | Usefulness | ✓ | ✓ |
| Usability | Operability | ✓ | ✓ |
| Compatibility | Interoperability | | ✓ |
| Freedom from risk | Environmental risk mitigation | | |
| Usability | User error protection | ✓ | ✓ |
| Usability | Learnability | ✓ | ✓ |
| Accessibility | Accessibility | | |
| Usability | Appropriateness recognizability | ✓ | ✓ |
| Compatibility | Co-existence | ✓ | ✓ |
| Data Privacy | Data Privacy | | ✓ |
| Persuasiveness | Persuasiveness | | ✓ |

TABLE IV
RELEVANT TECHNICAL SUSTAINABILITY-QUALITY REQUIREMENTS

| **Characteristic** | **Attribute** | **Case A** | **Case B** |
|---|---|---|---|
| Functional suitability | Functional correctness | ✓ | ✓ |
| Compatibility | Interoperability | ✓ | ✓ |
| Reliability | Availability | ✓ | ✓ |
| Functional suitability | Functional appropriateness | ✓ | ✓ |
| Satisfaction | Usefulness | | |
| Reliability | Fault tolerance | | ✓ |
| Maintainability | Modifiability | ✓ | ✓ |
| Satisfaction | Trust | | |
| Context coverage | Context completeness | ✓ | ‡ |
| Effectiveness | Effectiveness | ‡ | |
| Robustness | Robustness | | |
| Portability | Adaptability | ✓ | |
| Performance efficiency | Time behaviour | ✓ | |
| Maintainability | Modularity | ✓ | ✓ |
| Maintainability | Testability | ✓ | ✓ |
| Reliability | Recoverability | | ✓ |
| Compatibility | Coexistence | ✓ | |
| Reliability | Maturity | | ✓ |
| Efficiency | Efficiency | | |
| Survivability | Survivability | | |
| Performance efficiency | Capacity | | |
| Security | Integrity | ✓ | ✓ |
| Replaceability | Replaceability | ✓ | |
| Scalability | Scalability | | ✓ |

TABLE V
RELEVANT ENVIRONMENTAL SUSTAINABILITY-QUALITY REQUIREMENTS

| **Characteristic** | **Attribute** | **Case A** | **Case B** |
|---|---|---|---|
| Maintainability | Reusability | ✓ | |
| Maintainability | Modifiability | ✓ | |
| Performance efficiency | Resource utilization | | ✓ |
| Freedom from risk | Environmental risk mitigation | | ‡ |
| Performance efficiency | Time behaviour | | ✓ |
| Reliability | Availability | | |
| Efficiency | Efficiency | | |
| Compatibility | Co-existence | | |

TABLE VI
RELEVANT ECONOMIC SUSTAINABILITY-QUALITY REQUIREMENTS

| **Characteristic** | **Attribute** | **Case A** | **Case B** |
|---|---|---|---|
| Effectiveness | Effectiveness | ✓ | ✓ |
| Reliability | Availability | | |
| Satisfaction | Trust | | |
| Satisfaction | Usefulness | | |
| Freedom from risk | Economic risk mitigation | | |
| Context coverage | Context completeness | ✓ | ‡ |
| Context coverage | Flexibility | ✓ | ‡ |
| Functional suitability | Functional appropriateness | ✓ | ✓ |
| Functional suitability | Functional correctness | ✓ | ✓ |
| Reliability | Recoverability | | |
| Efficiency | Efficiency | | ✓ |
| Functional suitability | Functional completeness | | |
| Automatable | Automatable | | ✓ |
| Replaceability | Replaceability | ✓ | |
| Compatibility | Interoperability | ✓ | ✓ |
| Compatibility | Coexistence | ✓ | ✓ |
| Maintainability | Modifiability | ✓ | |
| Usability | Error protection | ✓ | ✓ |
| Usability | Learnability | ✓ | ✓ |
| Portability | Adaptability | ✓ | |
| Performance efficiency | Capacity | ✓ | |
| Scalability | Scalability | ✓ | |