# A Classification Approach of Sustainability Aware Requirements Methods

Gabriel Alberto García Mireles
Departamento de Matemáticas
Universidad de Sonora
Hermosillo, México
mireles@mat.uson.mx

Ma. Ángeles Moraga, Félix García, Mario Piattini
ITSI, Universidad de Castilla – La Mancha
Ciudad Real, España
{MariaAngeles.Moraga, Felix.Garcia,
Mario.Piattini}@uclm.es

*Abstract* **— Sustainability concept is interpreted in different ways within and among disciplines. Thus, software developers must be aware of the extent to which methods used in requirements engineering are supporting this issue. So, this paper aims to classify methods used to develop or manage environmental sustainability (green) requirements as regards three categories: the sustainability notions they rely on, sustainability software view, and sustainability requirements view. The classification approach was validated with 16 papers selected by a Systematic Mapping Study method. As a result, it was found that the reviewed papers focused on sustainability goals related to second order effects and the case studies addressed mainly early requirements during development stage of the software life cycle. Few papers reported requirements validation practices. In a nutshell, the classification approach allowed to identify that requirements engineering methods should consider others sustainability notions and to provide methodological support to derive software requirements from sustainability goals.**

***Keywords - environmental sustainability; green software; software requirements; systematic mapping study.***

## I. INTRODUCTION

Requirements practices are of great importance to identify environmental sustainability goals. Requirements are considered as a main approach to address sustainable development, since they can support the identification of relevant sustainability goals [1]. Penzendstadler [2] pointed out that sustainability requirements can be considered as another objective in the context of requirements and that the methods used in requirements engineering (RE) can be adopted to address sustainability issues.

Sustainable development is considered a multidisciplinary field that needs to integrate research effort from different communities [3]. However, the notion of sustainability and their dimensions is an open issue among the software engineering community [3-5]. Few practitioners has a broad and systemic scope about sustainability while others relate it with natural resources availability and waste reduction [6]. Since the RE function is to mediate between domains of the acquirer and the supplier when system requirements are established for a system or software [7], software developers should be aware of different notions of sustainability. In addition, they should understand how sustainability requirements methods can be used to effectively define the sustainable requirements that customer wants to be implement in software.

The aim of this work is to understand the extent to which RE methods and practices address sustainability notions and how they are applied in case studies. So, this paper proposes a classification approach of sustainability aware requirements methods and practices. Sustainability aware requirement is a requirement that contributes to sustainable development. The classification considers three aspects: general sustainability notion, green software life cycle view, and RE view. Although a common view of sustainable development is to take into account the three sustainability dimensions (environmental, social, and economic), in this work the focus is on the environmental (green) dimension during software development.

As a result of classifying a set of papers addressing sustainability RE methods and practices, a catalog was obtained. This catalog may support practitioners when they need to select an appropriate method to address environmental sustainability stakeholders' needs. To the research field, this work provides a conceptual framework to classify RE methods and allows the identification of research gaps too.

Section II provides a brief summary of related work while Section III describes the suggested classification for the requirements engineering proposals. Section IV describes the methodology (systematic mapping study) to identify the set of papers. Section V presents a summary of main results and the discussion is showed in Section VI. Finally, conclusions and future work are depicted in Section VII.

## II. RELATED WORK

Since sustainability in software engineering is a recent research topic, there are few studies that consolidate the knowledge in software engineering disciplines. So an approach to identify RE methods and practices is reviewing systematic literature reviews about sustainability in software engineering. One of them, Penzenstadler et al. [8], has a search string aims focusing on sustainability or environmental issues with software engineering or requirements for software systems. Despite requirements are in the search string, the results do not provide a requirements-based classification. Most of the papers categorized as requirements belong to green by IT category. In a follow-up study [9], authors reported 11 papers belonging to software requirements area, which applied methods such as goal modeling, stakeholder modeling, agent modeling, service modeling, process modeling and simulation.

978-989-98434-7-9/17/$31.00 ©2017 AISTI 

A paper that tried to consolidate RE practices is the one by Chitchyan et al. [10]. They reviewed papers discussing requirements engineering activities and sustainability in order to provide an integrate perspective of the topic. They used data from two case studies to describe how requirements engineering practices can be used to support sustainability. However, both examples address only the green by IT perspective.

## III. SUSTAINABILITY RE METHODS CLASSIFICATION

Sustainability is defined as the capacity of enduring, but the most common definition used in research is "development that meets the needs of the present without compromising the ability of future generations to meet their own needs" [11]. Indeed, sustainability for software developers "means working with people across many disciplines" [3]. However, software engineering faces the challenge of dealing with several meanings for sustainability [12].

Concerning this challenge, Lankoski [13] provides a conceptual framework to analyze sustainability conceptions of organizations. It has three dimensions: sustainability scope, substitutability approach and goal orientation. Sustainability scope is *narrow* when it focuses on protecting the natural environment, but it is *broad* when it comprises issues of the environmental, social and economic dimensions. Substitutability approach addresses "the extent to which poor performance in one sustainability issue can be compensated by good performance in other" [13]. When substitution of sustainability issues belonging to different dimensions is allowed, it is called *weak* sustainability. When this type of substitution is not allowed, it is called *strong* sustainability. By last, the goal orientation dimension can be absolute or relative. *Absolute* means that performance is assessed by achieving a critical outcome, and *relative* means that performance is assessed as regards a reference framework.

Within the software engineering field, there is confusion about the meaning of software sustainability and how it can be measured [5]. While a definition considers green software as an application that produces as little waste as possible during its development and operation [14], another definition sees sustainability as a broader concept. For instance, green and sustainable software is defined as "software whose direct and indirect negative impacts on economy, society, human beings, and environment that result from development, deployment and usage of the software are minimal and/or which has a positive effect on sustainable development" [15].

Given that IT, including software, can either be part of the problem or part of the solution, some researchers distinguish between *green in software* and *green by software* [4, 16]. The former concept is related to the optimization of energy efficiency and resources used by software itself while the later supports goals related to sustainable development in fields distinct from IT. However, Taina [17] noted that green software should achieve both saving resources and reducing waste during its execution and supporting sustainable development. Thus, software can support both purposes, called in this paper *green in/by software*.

Naumann et al. [15] present the GREENSOFT model, a model that addresses both types of software: green by software and green in software. Two main components of the model are the software life cycle and the level of impact of environmental sustainability and other dimensions. The former component comprises three main stages: *development*, *usage* and *end of life*. The latter includes nested order of effects. The *first order effects* refer to the direct impact of activities used in producing, using, and disposing software. The *second order effects* refer to the application of the software as an enabler of application of technology for sustainable development. Finally, the *third order effects,* centered on systemic effects, are long-term effects in social values and economical structures [18].

Software requirements can be addressed in several abstraction levels and using different representations. Loucopoulos et al. 2013 [19] consider two categories in the abstraction level category: early requirements and late requirements. *Early requirements* model and analyze stakeholders' needs and interests and how they can be addressed in a new system [20]. *Late requirements* focus on the specification of functionality and system software quality in order to support the development of system components [19, 20].

RE methods and practices can be classified by means of a process perspective, which is based on SWEBOK 3.0 [21]. The process activities are requirements elicitation, requirements analysis, requirements specification, and requirements validation. *Requirements elicitation* is concerned with the source of requirements and the way they can be collected. *Requirements analysis* studies the user and stakeholder requirements in order to detect and resolve conflicts between requirements and to derive software requirements. It includes conceptual modeling and classification of requirements. *Requirements specification* refers to documenting software requirements in a way they can be evaluated and support the design of a software system. Finally, *requirements validation* practices ensure that software specification defines the software stakeholders expect.

## IV. METHOD

In order to determine the utility of the classification approach, a set of papers addressing RE methods and practices to develop or manage sustainability requirements was identified by conducting a Systematic Mapping Study (SMS) [22]. The general research question is the following: What practices, methods or techniques are addressing sustainability requirements in the context of software development? The specific research questions are: 1) What sustainability notions are considered? 2) What are the focus of sustainability requirements methods as regards green software life cycle? and, 3) What are the RE activities supported by these methods?

The search string used has two parts (Table I). The final search string is formed by String A OR String B. The search string was executed in Web of Science (http://apps.webofknowledge.com), IEEE Xplore (http://ieeexplore.ieee.org/Xplore/home.jsp), Scopus database (https://www.scopus.com/home.uri), and ACM Digital Library (http://dl.acm.org/).

TABLE I. SEARCH STRING

| String | Expression |
|---|---|
| String A | "sustainability requirements" |
| String B | ( sustainability OR sustainable OR green OR greenability ) AND software AND requirements |

The time span was up to year 2016. Inclusion criteria for selected papers were as follows: peer-reviewed papers written in English; discussing methods, techniques or practices used to identify environmental sustainability requirements in software developing (green in software); and papers including some evidence of their usage (e.g.., an example or a case study).

The data extracted from the papers included: objectives, methods or practices used, extracts supporting the concept of sustainability and the type of green software supported by the evidence. From the extracted data, frequencies of different categories were obtained in order to identify trends in sustainability RE methods. In addition, the classification approach described in Section III was applied.

## V. RESULTS

A set of 1142 records were obtained from the four databases mentioned in the previous section, from which 16 papers fulfilled the selection criteria to be fully read and extract data (Table II). The results are presented in three categories: sustainability notion, green software life cycle view, and support for requirements engineering activities.

### A. *Sustainability notion*

The most common definition found for sustainability was the one by Bruntland report (13 out 16 papers), although it was also defined as the "capacity to endure" [1]. Meanwhile, 12 papers lacked of a definition for sustainable software or sustainable software engineering, while the rest of them (s6, s7, s8, s15) provided, at least, one definition, being all of them different in scope.

As regards the type of requirement, 12 papers mentioned a category for sustainability requirements. 4 papers addressed sustainability requirements as requirements, while another one labeled them as software objective. When sustainability was treated as a quality attribute, it took several names, including nonfunctional requirements, quality requirements, softgoal, and sustainability quality requirements. The latter covers traditional quality requirements and sustainability requirements. However, s8 noted that labeling sustainability requirements as quality requirements is not appropriate since they can be derived into functional requirements and restrictions.

Based on Lankoski's [13] sustainability scope category, it was found that seven papers had a narrow scope, while nine were based on broad one (Table III). The former focused on environmental sustainability while the latter addressed environmental, social, and economic sustainability. In addition, several papers addressing the broad scope also included technical (7 papers) and individual (5 papers) dimensions. As regards substitutability category [13], six papers did not mention issues related to balance or trade-offs between requirements belonging to different sustainability dimensions.

TABLE II. SELECTED PAPERS

| ID | Goal | Evidence | Domain |
|---|---|---|---|
| s1 [23] | Modelling sustainability issues using i* | Case study | Conference organization |
| s2 [24] | Applying requirements methods to discover sustainability requirements | Experience report | Events organization |
| s3 [25] | Classifying sustainability requirements using a sustainability NFR framework | Examples | Monitoring resource tools |
| s4 [26] | Identifying sustainability requirements using RE4S framework | Case study | Monitoring medication adherence tool |
| s5 [1] | Conducting requirements activities for identify sustainability requirements | Case study | Procurement system |
| s6 [27] | Analyzing sustainability dimension using a framework | Case study | Climate moni-toring system |
| s7 [28] | Exploring sustainability quality requirements using a sustainability analysis framework | Case study | Paper mill control system, Car sharing system |
| s8 [29] | Eliciting sustainability requirements for a decision support system | Case study | Meal planner system |
| s9 [30] | Specifying sustainability requirements by using sustainability patterns | Student projects | Monitoring resources in hotel system |
| s10 [31] | Proposing an energy profiling tool to describe green efficiency requirements | Experiments | Mobile app. |
| s11 [32] | Identifying sustainable requirements in software | Example | N/A |
| s12 [33] | Proposing a sustainability meta-model to describe sustainability requirements | Example | Car sharing platform |
| s13 [34] | Identifying stakeholders who advocate sustainable development | Case study | Car sharing platform |
| s14 [35] | Applying the IMAGINE approach to assess sustainability of system scenarios | Case study | Car sharing platform |
| s15 [36] | Analyzing relationships between environmental sustainability and quality | Survey | IT company |
| s16 [37] | Determining relationships between quality and environmental attributes | Survey | IT company |

TABLE III. NOTIONS OF SUSTAINABILITY MODEL

| Category | Value | Papers |
|---|---|---|
| Scope | Narrow | s1, s2, s8, s9, s10, s15, s16 |
| | Broad | s3, s4, s5, s6, s7, s11, s12, s13, s14 |
| Substituta-bility | N/A | s8, s9, s10, s12, s13, s14 |
| | Weak | s1, s2, s3, s4, s5, s6,s7, s11, s15, s16 |
| | Strong | None |
| Goal orientation | N/A | s1, s2, s8, s9, s15, s16 |
| | Relative | s3, s4, s5, s6, s7, s10, s11, s12, s13, s14 |
| | Absolute | None |

In the goal orientation category [13], six papers lacked of information on how software can be assessed as regards sustainability requirements (Table III). On the other hand, ten papers mentioned the need to establish criteria or indicators to assess the extent to which sustainability requirements are achieved. However, these papers did not address explicitly whether the goal orientation was either absolute or relative. Indeed, they do not address environmental critical indicators.

TABLE IV. APPROACH TO SUSTAINABLE SOFTWARE

| Category | Value | Papers |
|---|---|---|
| (G)reen software | G. by software | s1, s2, s4, s5, s6, s8, s11, s12, s13, s14 |
| | G. in software | s10, s15, s16 |
| | G. in/by software | s3, s7, s9 |
| Life cycle | Development | s1, s2, s3, s4, s5, s6, s7, s8, s9, s10, s11, s13, s14 |
| | Usage | s5, s6, s11 |
| | End of life | None |
| Order effects | First order | s3, s4, s5, s7, s9, s10, s15, s16 |
| | Second order | s1, s2, s3, s4, s6, s7, s8, s11, s12, s13, s14 |
| | Third order | s5 |

### *B. Green software view*

Concerning the green software perspective showed in the case studies and examples, it was found that most of the papers described goals, requirements or sustainability issues in terms of the specific application domain (papers in G. by software value in Table IV). Three of the papers discussed requirements techniques that can be applied to determine green in software requirements. The papers also addressed both types of software purposes, as s3, s7, and s9 showed.

Based on the GREENSOFT model [15], it can be said that the requirements of the selected papers focused on identifying relevant requirements to develop software systems, most of them discussing software systems goals can contribute to sustainability. However, few papers considered that the software system also has an impact during usage stage (they consume energy as regards usage scenarios). About usage stage, three papers said to monitor the software system performance during the software operation stage (papers in usage value category from Table IV). However, among the selected papers, no paper explicitly mentioned how to address the end of life stage of the software system under study.

Regarding order effects in the environmental sustainability dimension, evidence of case studies and examples showed that there is a trend to consider the second order effects, since the software system purpose is established as regards the potential effect of software when it might be operable in application domains distinct from Information Technology. Half of the selected papers addressed goals or requirements to reduce energy consumption or minimize the usage of resources (First order value in Table IV). However, only one paper mentioned a third order effect.

### *C. Requirements engineering activities coverage*

In selected papers, requirements are addressed at high level of abstraction considering needs and goals of stakeholders. The majority of papers (12) were categorized as early abstraction level. Three papers described approaches to address specific software requirements while one papers present the two categories of requirements (Table V). The abstraction level used to describe requirements focused on early requirements, i.e. methods addressing stakeholders' needs and goals. However, it is needed support to derive late requirements from the early ones. Proposals to address sustainability during RE activities focused on elicitation and analysis (Table V). Few papers addressed validation practices.

In the elicitation activity, the source of requirements were sustainability related guidelines in the application domain, general catalogues of both sustainability goals and indicators, interviews and surveys with relevant stakeholders. Three papers discussed the identification of sustainability-related stakeholders: sustainability chair, sustainability actor, anti-sustainability actor, as well as approaches to identify them as regards organizational structure or sustainability dimensions under study. Methods to discuss sustainability goals with stakeholders included the following: elaboration of a rich picture and business case analysis, development of a system vision, construction of usage models, analysis cost-benefit of alternatives and Goal-Question-Metric approach.

Within analysis activities, proposals addressed goal-based methods such as i*, KAOS, and other goal models. Other papers considered classification methods and sustainability-based models to discover sustainability requirements. In addition, a paper provided a meta-model to characterize sustainability requirements with concepts related to dimension, value, indicators, and activities. As regards prioritizing requirements, some papers used statistical analysis methods and Analytical Network Process. Moreover, a paper related energy efficiency requirements with an energy profiling tool to establish a baseline for energy consumption in software.

Specification of requirement activity addressed UML class diagrams and a method to describe detailed requirements using requirements facets and rationale. Another paper used requirements specification based on a template that included sustainability criteria checklists. In addition, a paper presented a UML class diagram to describe sustainability requirements. These included relevant parameters and evaluation indicators. Describing sustainability requirements can also be based on sustainability requirements patterns. Finally, identifying measures for energy efficiency can contribute to specify sustainability requirements.

As regards validation activities, the selected papers addressed two approaches. The first of them, considered that sustainability requirements should be reviewed by the customer. The other approach proposed to monitor the system during the operation stage considering an appropriate set of indicators related to sustainability dimensions under study.

TABLE V. REQUIREMENTS APPROACHES

| Category | Value | Papers |
|---|---|---|
| Abstraction level | Early | s1, s4, s5, s6, s7, s8, s11, s12, s13, s14, s15, s16 |
| | Late | s3, s9, s10 |
| | Both | s2 |
| Process | Elicitation | s1, s2, s3, s4, s5, s6, s7, s8, s9, s10, s11, s13, s14, s15 |
| | Analysis | s1, s2, s3, s4, s5, s6, s7, s8, s10, s11, s12, s15, s16 |
| | Specification | s2, s3, s4, s5, s7, s9, s10 |
| | Validation | s1, s2, s4, s5, s6 |

## VI. Discussion

Despite the majority of the papers relied on the sustainable development definition of Bruntland's report [11], the classification of 16 sustainability aware requirements papers showed that there are many interpretations when methods and techniques are proposed and applied in case studies. In this review, the Lankoski's sustainability model [13] was used to identify different notions of sustainability concept. Both narrow and broad sustainability scope were represented evenly in the selected papers. This means that some RE methods are focused only on the environmental dimension while the rest of them, at least, take into account the three bottom line sustainability dimensions. However, the other two Lankoski's categories showed that RE methods are addressing them partially.

As regards substitutability approach, the ten papers considered appropriate to carry out a balance among dimensions or carry out trade-offs (weak substitutability). However, discussion of the effect of either balancing or trading-off analysis among sustainability dimensions was only discussed as a technical issue of proposed methods. There was no discussion about different stakeholders' views working in other disciplines. For instance, Lankoski [13] noted that a strong conception of sustainability is preferred among ecologists but economists prefer the weak conception. In addition, Lago et al. [16] pointed out that environmental sustainability needs much greater contextual information to ensure sound reasoning and this topic is subject of future research. Thus, given that software can be developed in any application domain, requirements engineers must be aware of this difference.

Goal orientation focused on how the indicators used to measure the impact on sustainability are considered in software (our application domain). Ten papers mentioned the possibility to assess the extent to which sustainability might be achieved in software; however there were no detailed arguments about the way the software indicators can be compared with other benchmarks or environmental critical values. Indeed, Lago et al. [38] pointed out that "the major challenge in kick starting sound research in green software is to investigate how to break down the definition of sustainability [relative understanding vs absolute understanding] so that it can be applied to software engineering."

As regards software sustainability, the set of papers under study showed that few authors documented the sustainability software definition or addressed the meaning of sustainable software. In addition, the majority of the papers provided evidence for the green by software category and focused on providing goals to address second order effects to discover requirements in the development stage of software life cycle. Thus, this result shows that methods for improving the development and management of sustainability requirements are in an exploratory stage.

The evidence provided in case studies and examples showed that researchers are working with early requirements. So, there is a need for additional research work to derive late requirements taking into account the early requirements. About process activities, the selected papers described RE methods that have been adopted to elicit and analyze sustainability requirements. In addition, some proposals talked about how to characterize sustainability requirements to address the sustainability dimension relevant for stakeholders.

Sustainability can be interpreted in different ways and this can impact the agreements between software developers and stakeholders. Within software engineering literature, the main definitions of sustainability are related to the capability to endure and sustainable development. However, there are other interpretations [13]. In this work, a model used in business context was used to explore its potential to classify sustainability aware RE methods. Although the model allowed to classify the selected papers, it can be adapted to the software engineering field. In addition, the selection of papers was focused on those that addressed the environmental sustainability dimension. Hence, a full view of sustainability aware requirements methods need to address other sustainability dimensions.

As regards validity threats related to the SMS, bias in selection of publications and inaccuracy in data extraction are considered. The search string was piloted and executed in databases recommended to use in SE [39]. In addition, the set of selected papers include both conference and workshop papers focused on software sustainability. The reduced number of selected papers gathered in this SMS is consistent with other literature review papers [9, 10]. However, one author carried out the selection procedure and the other authors reviewed the results. The data extraction bias was mitigated by using a template to extract data. In addition, the classification framework (Section III) describes the main concepts to look for in selected papers. Furthermore, SWEBOK [21] and GREENSOFT model [15], were used to classify primary papers. Nevertheless, the extracting data procedure was executed by one author and reviewed by the remaining authors.

## VII. Conclusions

This paper presented a classification approach for sustainability aware requirements methods. The classification had three main parts: sustainability notion view, sustainable software view, and requirements view. The classification approach was applied to 16 papers that were identified as a result of conducting a Systematic Mapping Study. The classification approach allowed to identify the main notions as regards sustainability as well as how sustainable software is managed in the context of requirements engineering methods. In addition, papers were focused on exploring sustainability in elicitation and analysis activities of software requirements knowledge area. Therefore, this result suggested that the reviewed papers show a research area in an exploration stage.

For future work, the understandings about the different interpretations of sustainability concept might provide an approach to develop new sustainability aware requirements methods or practices. In addition, addressing different stages of software life cycle and the three order impacts on sustainable development is needed. Finally, the proposed methods require validation in industrial settings to understand the extent to which they might contribute to sustainable development.

## ACKNOWLEDGEMENT

This work has been funded by the project: GINSENG (Ministerio de Economía y Competitividad y Fondo Europeo de Desarrollo Regional FEDER, TIN2015-70259-C2-1-R).


## REFERENCES


[1] Becker, C., Betz, S., Chitchyan, R., Duboc, L., Easterbrook, S.M., Penzenstadler, B., Seyff, N., Venters, C.C.: Requirements: The key to sustainability. IEEE Software. 33, 56-65 (2016)

[2] Penzenstadler, B.: Infusing Green: Requirements Engineering for Green In and Through Software Systems. In: RE4SuSy@ RE, pp. 44-53. (2014)

[3] Becker, C., Chitchyan, R., Duboc, L., Easterbrook, S., Penzenstadler, B., Seyff, N., Venters, C.C.: Sustainability Design and Software: The Karlskrona Manifesto. In: Proceedings - International Conference on Software Engineering, pp. 467-476. (2015)

[4] Calero, C., Piattini, M.: Introduction to green in software engineering. In: Calero, C.Piattini, M., (eds.). Green in Software Engineering. pp. 3-27. (2015)

[5] Venters, C.C., Jay, C., Lau, L.M.S., Griffiths, M.K., Holmes, V., Ward, R.R., Austin, J., Dibsdale, C.E., Xu, J.: Software sustainability: The modern tower of babel. In: CEUR Workshop Proceedings, pp. 7-12. (2014)

[6] Chitchyan, R., Becker, C., Betz, S., Duboc, L., Penzenstadler, B., Seyff, N., Venters, C.C.: Sustainability design in requirements engineering: state of practice. In: Proceedings of the 38th International Conference on Software Engineering Companion, pp. 533-542. ACM, (2016)

[7] IEEE. SE vocabulary - Requirements Engineering. 2017; Available from: https://pascal.computer.org/sev_display/search.action.

[8] Penzenstadler, B., Bauer, V., Calero, C., Franch, X.: Sustainability in software engineering: A systematic literature review. In: IET Seminar Digest, pp. 32-41. (2012)

[9] Penzenstadler, B., Raturi, A., Richardson, D., Calero, C., Femmer, H., Franch, X.: Systematic mapping Study on Software Engineering for Sustainability (SE4S). In: ACM International Conference Proceeding Series, pp. 1-10. (2014)

[10] Chitchyan, R., Betz, S., Duboc, L., Penzenstadler, B., Ponsard, C., Venters, C.C.: Evidencing sustainability design through examples. In: CEUR Workshop Proceedings, pp. 45-54. (2015)

[11] United-Nation, W.C.o.E.D., Report of the world comission on environment and development: our common future, in United Nation conference on environmental development. (1987).

[12] Katz, D.S., Choi, S.-C.T., Lapp, H., Maheshwari, K., Löffler, F., Turk, M., Hanwell, M.D., Wilkins-Diehr, N., Hetherington, J., Howison, J.: Summary of the first workshop on sustainable software for science: Practice and experiences (WSSSPE1). arXiv preprint arXiv:1404.7414(2014)

[13] Lankoski, L.: Alternative conceptions of sustainability in a business context. Journal of Cleaner Production. 139, 847-857 (2016)

[14] Erdélyi, K.: Special factors of development of green software supporting eco sustainability. In: SISY 2013 - IEEE 11th International Symposium on Intelligent Systems and Informatics, Proceedings, pp. 337-340. (2013)

[15] Naumann, S., Dick, M., Kern, E., Johann, T.: The GREENSOFT Model: A reference model for green and sustainable software and its engineering. Sustainable Computing: Informatics and Systems. 1, 294-304 (2011)

[16] Lago, P., Meyer, N., Morisio, M., Müller, H.A., Scanniello, G.: Leveraging "energy efficiency to software users": summary of the second GREENS workshop, at ICSE 2013. ACM SIGSOFT Software Engineering Notes. 39, 36-38 (2014)

[17] Taina, J.: Good, bad, and beautiful software -- in search of green software quality factors. CEPIS UPGRADE. 11, 22-27 (2011)

[18] Hilty, L.M., Aebischer, B., Ict for sustainability: An emerging research field, in Advances in Intelligent Systems and Computing. (2015). p. 3-36.

[19] Loucopoulos, P., Sun, J., Zhao, L., Heidari, F.: A systematic classification and analysis of NFRs. In: 19th Americas Conference on Information Systems, AMCIS 2013 - Hyperconnected World: Anything, Anywhere, Anytime, pp. 208-217. (2013)

[20] Kavakli, E.: Goal-oriented requirements engineering: A unifying framework. Requirements Engineering. 6, 237-251 (2002)

[21] Bourque, P., Fairley, R.E., eds. Guide to the Software Engineering Body of Knowledge, Version 3.0. 2014, IEEE Computer Society.

[22] Kitchenham, B., Charters, S., Guidelines for performing systematic literature reviews in software engineering, in Technical report, Ver. 2.3 EBSE Technical Report. EBSE. (2007).

[23] Cabot, J., Easterbrook, S., Horkoff, J., Lessard, L., Liaskos, S., Mazón, J.N.: Integrating sustainability in decision-making processes: A modelling strategy. In: 2009 31st International Conference on Software Engineering - Companion Volume, ICSE 2009, pp. 207-210. (2009)

[24] Mahaux, M., Heymans, P., Saval, G., Discovering sustainability requirements: An experience report, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). (2011). p. 19-33.

[25] Raturi, A., Penzenstadler, B., Tomlinson, B., Richardson, D.: Developing a sustainability non-functional requirements framework. In: 3rd International Workshop on Green and Sustainable Software, GREENS 2014 - Proceedings, pp. 1-8. (2014)

[26] Penzenstadler, B., Mehrabi, J., Richardson, D.J.: Supporting Physicians by RE4S: Evaluating Requirements Engineering for Sustainability in the Medical Domain. In: Proceedings - 4th International Workshop on Green and Sustainable Software, GREENS 2015, pp. 36-42. (2015)

[27] Saputri, T.R.D., Lee, S.W., Incorporating sustainability design in requirements engineering process: A preliminary study, in Communications in Computer and Information Science. (2016). p. 53-67.

[28] Lago, P., Koçak, S.A., Crnkovic, I., Penzenstadler, B.: Framing sustainability as a property of software quality. Communications of the ACM. 58, 70-78 (2015)

[29] Huber, M.Z., Hilty, L.M., Glinz, M.: Uncovering sustainability requirements: An exploratory case study in canteens. In: CEUR Workshop Proceedings, pp. 35-44. (2015)

[30] Roher, K., Richardson, D.: A proposed recommender system for eliciting software sustainability requirements. In: 2013 2nd International Workshop on User Evaluations for Software Engineering Researchers, USER 2013 - Proceedings, pp. 16-19. (2013)

[31] Beghoura, M.A., Boubetra, A., Boukerram, A.: Green software requirements and measurement: random decision forests-based software energy consumption profiling. Requirements Engineering. doi: 10.1007/s00766-015-0234-2, (2015)

[32] Alharthi, A.D., Spichkova, M., Hamilton, M.: Sustainability profiling of long-living software systems. In: CEUR Workshop Proceedings, pp. 12-19. (2016)

[33] Penzenstadler, B., Femmer, H.: A generic model for sustainability with process- and product-specific instances. In: GIBSE 2013 - Proceedings of the 2013 Workshop on Green in Software Engineering, Green by Software Engineering, pp. 3-7. (2013)

[34] Penzenstadler, B., Femmer, H., Richardson, D.: Who is the advocate? Stakeholders for sustainability. In: 2013 2nd International Workshop on Green and Sustainable Software, GREENS 2013 - Proceedings, pp. 70-77. (2013)

[35] Rodriguez, A., Penzenstadler, B.: An assessment technique for sustainability: Applying the IMAGINE approach to software systems. In: CEUR Workshop Proceedings, pp. 1-8. (2013)

[36] Koçak, S.A., Alptekin, G.I., Bener, A.B.: Integrating environmental sustainability in software product quality. In: CEUR Workshop Proceedings, pp. 17-24. (2015)

[37] Koçak, S.A., Alptekin, G.I., Bener, A.B.: Evaluation of software product quality attributes and environmental attributes using ANP decision framework. In: CEUR Workshop Proceedings, pp. 37-44. (2014)

[38] Lago, P., Meyer, N., Morisio, M., Muller, H.A., Scanniello, G.: 2nd International workshop on green and sustainable software (GREENS 2013). In: Proceedings - International Conference on Software Engineering, pp. 1523-1524. (2013)

[39] Petersen, K., Vakkalanka, S., Kuzniarz, L.: Guidelines for conducting systematic mapping studies in software engineering: An update. Information and Software Technology. 64, 1-18 (2015)