Integrating Environmental Sustainability in Software Product Quality

Sedef Akınlı Koçak
Data Science Laboratory
Environmental Applied Science
Ryerson University
Toronto, Canada
sedef.akinlikocak6@ryerson.ca

Güllfem Işıklar Alptekin
Computer Engineering
Galatasaray University
Istanbul, Turkey
gisiklar@gsu.edu.tr

Ayşe Başar Bener
Data Science Laboratory
Mechanical and Industrial Engineering
Ryerson University
Toronto, Canada
ayse.bener@ryerson.ca

***Abstract*—Recently, sustainability in software engineering and especially in requirements engineering is an emerging field. Especially, increasing demand for energy and intensive use of software and software-related services are the key motivators for designing software products with environmental requirements. In this study, we identify the software practitioners perception of the energy-related impact of software quality in order to develop environmentally sustainable software product. We present the result of a survey study that is conducted with 53 software practitioners in 7 different companies. Through this survey, we aim to explore the correlation between software quality and energy efficiency. We found out that there are significant negative correlations between functional suitability and compatibility; performance efficiency and security; reliability and compatibility with respect to energy efficiency. We built regression models by using energy efficiency and resource efficiency. Regression models show that performance efficiency, reliability and usability attributes have significant energy-related impact on the quality of the environmentally sustainable software product. We believe that our analysis gives insights to software practitioners to consider energy efficiency in making requirements prioritization decisions.**

***Index Terms*—environmental sustainability; software quality; non-functional requirements; energy efficiency; decision making**

I. Introduction

It is important to define the sustainability of a software product by considering different sustainability dimensions such as economic, social, technical and environmental in order to evaluate the consequences of product or process decisions properly [1], [2]. We observe that decision-making in software development involves trade-offs either between different quality criteria such as performance versus reliability, or between economic dimension and quality, such as cost versus performance. Environmental dimension, especially, has big concerns since the immediate effect of energy efficiency and resource efficiency can easily be evaluated for any new product by understanding its functionality. Some studies argue that energy efficiency usually conflicts with quality goals (e.g. energy-performance trade-off)[3]. However, we believe that decision-making on sustainability-related software should complement quality rather than to replace it.

Recently, several authors have explicitly promoted sustainability in software engineering [4], [5]. Some studies claim that sustainability should be part of software quality models [4] such as in ISO/IEC 25010. Other studies reveal that sustainability has to be considered as a non-functional requirement (aka quality attribute) [5], [6]. Becker [7] stated that there is a need of understanding of the relationships between sustainability and software qualities supported by empirical studies. This reflects that sustainability has to be understood as a software quality and it should be part of design and development of software [8]. In order to integrate sustainability as a quality attribute, we believe that the starting point is the awareness of the practitioners. They need to be aware of sustainability concerns and understand the relationship between quality and sustainability requirements.

Although, it seems to be an abundance of information on sustainability, the level of awareness in the software engineering field is low in practice [9]. A recent study shows that developers consider sustainability at the level of the software artifact and they define sustainability in terms of maintainability and usability [10]. So far there is no consensus in the literature on how to relate sustainability in terms of software quality.

In most cases, sustainability has been associated with the use of environmental resources and it requires simultaneous consideration of environmental protection and direct effects of energy consumption. In contrast, as Hilty et al. [11] mentioned in their latest report, knowledge about designing and configuring software in an environmental manner is not sufficient today. Therefore, there is a need to have a common understanding of sustainability among software practitioners.

In this study, we are motivated to analyze the relationship between environmental sustainability and software quality from the points of view of software practitioners. Our analysis complements the literature on decision-making frameworks [12], [13] in the area of environmental sustainability and software engineering. We state our research question as:

- How does environmental sustainability relate to software product quality?

We studied how software practitioners consider the direct effect of product quality on environmental sustainability. We show a practical guide for practitioners to identify the important quality factors at early stage of software development. The data come from surveys that were completed by practitioners. We analyzed the correlations between software product quality criteria and environmental criteria. Then we built a regression model using product quality and environmental criteria. We identified a significant correlation between product quality attributes (i.e. functionality and performance), and sustainability attributes (i.e. energy efficiency and resource efficiency). The regression model is built using these criteria with significant correlations. The regression analysis results showed that resource efficiency measures may be useful to clarify environmental sustainability of software. Moreover, the results revealed a negative correlation between energy efficiency and quality. Our contributions are summarized as:

- Explore environmental dimension of sustainability in software engineering
- Identify and discuss the main challenges in the development of environmentally sustainable software quality attributes that are compatible with current software engineering quality practices.
- Perform an empirical study that integrates environmental sustainability into quality.
- Present the correlation between current software quality attributes and environmental attributes.
- Provide insights for practitioners to incorporate environmental sustainability in requirements selection decisions.

The paper is organized as follows. Section II represents related work. Section III provides with the methodology of our study. Analysis and results are given in Section IV. Section V presents threats to validity, and Section VI finalizes our work by giving a conclusion and future work.

II. Sustainability and Software Engineering

Brundtland commission [14] defines sustainability as "meeting the needs of the present without compromising the ability of future generations to meet their own needs", Seacord et al. [15] define sustainability in the field of software engineering as the "ability to modify a software system based on customer needs and deploy these modifications". Sustainability has different dimensions including economic, environmental, social, individual, and technical. Economic dimension relates to financial aspects and business value, whereas environmental dimension refers to usage and care of the natural resources. Social sustainability aims at preserving the societal communities in their solidarity and services and individual sustainability refers to maintaining individual human capital (e.g., health, education). Technical dimension relates to long-time usage of systems and their adequate evolution regarding changing surrounding conditions and related requirements [2], [6], [7], [16].

In the field of software engineering, recent discussion of sustainability tended to focus on the sustainability as a requirement in terms of technical, economic and environmental perspectives [17].

A. Sustainability in Decision Making

Considering sustainability into any research domain requires a multidisciplinary approach [18], [19]. Karlskrona Manifesto set out key principles and commitments for sustainability design [16]. One of the key principles of this manifesto is that sustainability needs to be integrated on multiple levels of the decision-making processes by using modeling and analysis techniques.

Software companies as well as software engineering research community have been mostly focused on the software quality attributes (e.g. reliability, performance and efficiency) and the cost-benefit analysis of these attributes in making decisions [20], [21]. These approaches mainly concentrate on the analysis of the economic and technical aspects.

Cabot et al. [22] implemented sustainability within the goal model [23] to support decision making, but they do not provide a methodological framework on how to apply or to use for decision support. Stefan et al. [24] used a quantitative approach with a case study where goals are given formal and measurable definitions. Mahaux et al. [25] performed a case study on a business information system but they did not discuss the challenges in the decision-making. Gu et al. [26] proposed a green strategy model that provides decision makers with the information to employ green strategies. They give a broader view on sustainable software engineering. Penzenstadler et al. [27] discussed sustainability as one of the major consideration in software product management decision-making and they defined a value-based approach for sustainability. However, they do not discuss how sustainability aspects could be measured in practice.

All of above the models in the literature are qualitative base models; therefore they offer a limited support for decision-making. In order to consider sustainability in decision-making process, a practical framework that includes all the relevant quality attributes is needed. These qualities need to characterize the sustainability concerns of software products and identify interdependency of qualities with respect to sustainability. In our previous studies [12], [13] a multi-criteria decision-making framework was built for prioritization of software requirements. The emphasis was on environmental sustainability. However, there are still open issues, especially on developing sustainability attribute, as they are interpreted as quality attribute, and incorporating them during the process of software development. The work in this paper is the first step towards capturing and analyzing the practitioners insights on the impact of quality and environmental sustainability. Therefore, the method and the results will provide evidence on how software practitioners may incorporate environmental sustainability in terms of energy efficiency and resource efficiency when they make software requirements selection decisions.

B. Sustainability as a Software Quality

Current discussions on the sustainability requirements are built on how to define, measure and assess sustainability as a quality attribute of software [28]. The recent quality model/standards are introduced by ISO (ISO/9126 and

ISO/IEC 25010) [29] but none of the sustainability dimensions are considered as quality attribute in the standards.

In the software engineering literature, the first quality model for green and sustainable software was developed by Kern et al [30]. It refers to a quality factors from ISO /IEC 25000 based on the direct and indirect quality attributes of software. The quality model gives an overview of potential aspects that may be taken as sustainability attribute as well as the metrics for software products. The model just considers the product quality factors, however, the quality aspects standardized in ISO /IEC 25000 are also related with the quality of software in use. Calero and Bertoa [4] consider sustainability as a new factor that affects software product and process quality. They presented a new quality model (ISO 2510+S) based on ISO/25010. In the model they differentiate the quality factors with respect to the sustainability impact and they describe related and unrelated sub-characteristics. All these studies discuss the relationship between the software quality and sustainability in general terms. They point out that the product as well as the quality in use needs to be considered when assessing the sustainability of the software. On the other hand, none of the studies have mentioned and investigated the impact of the quality on sustainability dimensions. Our study is the first one that attempts to analyze correlation between the standardized quality attributes and environmental sustainability attributes to identify their effects on environmental sustainability.

III. Methodology

We conducted an eight-week survey study at Galatasaray University, Center for Research and Decision Analysis and Applications in Turkey using a questionnaire. The study was conducted with 53 software practitioners in 7 different companies in Turkey. Four of them are large size information technology (IT) companies and the remaining three are small and medium size IT companies. The majority of the participants are working in the development and quality assurance teams.

A. Development of the Questionnaire

A questionnaire was designed to understand and evaluate the relationship of different criteria. In order to clarify the term we used “criteria“ as for attribute. Software practitioners were asked to prioritize the importance of different criteria regarding product quality and environmental sustainability in their decision-making process. The procedure was adopted from Wohlin and Aurum [31] when designing the questionnaire:

- A brainstorming session was held to identify suitable criteria of quality and environment to include in the questionnaire as well as to design the format of the questionnaire. The session included all the authors.
- The questionnaire was designed by the main author of this paper based on the outcome of the brainstorming session.
- The questionnaire was reviewed and updated by the authors to further improve the questions. Then it is sent to a contact person at different companies.

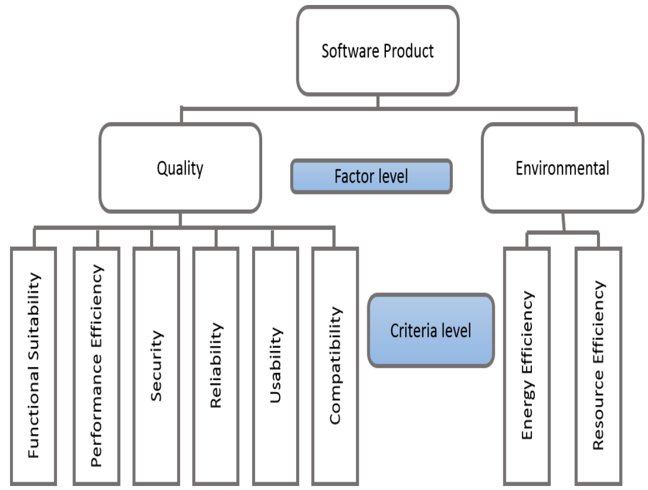

Fig. 1: Hierarchical structure of factors and criteria

The brainstorming session and the review process included some discussion about whether it was possible to identify dependencies in criteria. It was concluded that it would only be possible if the criteria were kept at a high level of abstraction. This would mean that few criteria would be evaluated and prioritized by the practitioners in the study. In summary, the objective was to judge the importance of an individual criterion among the other criteria. We identified the quality and environment as factors that are described as the external view of the software (as it is viewed by the users), and criteria that are described as the internal view of the software (as it is seen by the practitioners). Figure 1 shows the hierarchical structure that divides quality and environmental factors into criteria which may consist of sub-criteria. Please note that the terms factor and criteria are not part of the ISO/IEC standard, but we introduced them here for the sake of simplicity.

B. Description of Criteria

After literature review and brainstorming, we selected eight *criteria* (shown in Table I) that would be assessed by the practitioners. Many of the *criteria* were general in the sense that they were often referred in the literature when discussing software quality. It was agreed among the researchers that the eight *criteria* are covered by two factors, software product quality and environmental sustainability, although this grouping was not communicated to the practitioners.

1) Product Quality Criteria: Software product quality criteria were adopted from the ISO/IEC 25000 (SQuaRE) [29] series. ISO/IEC 25010 is a part of Square series that is composed of a quality in use and a product quality model. Considering the scope of our study, we used product quality model that categorizes product quality properties into eight characteristics (functional suitability, reliability, performance efficiency, usability, security, compatibility, maintainability and portability). While functional suitability, reliability, performance efficiency, usability, security and compatibility are defined as internal quality characteristics; maintainability and portability are external characteristics. For the purpose of this

TABLE I: Software quality and environmental factors and their related criteria

(Q)	**Software Quality** [29]
Q1 Functional Suitability	The product fits the functional requirements of the user and customer.
Q2 Performance Efficiency	How well the product responds to user requests and how efficient it is at execution time.
Q3 Reliability	The product produces failures and hence may not be available.
Q4 Usability	How easily the system can be used.
Q5 Security	Keep data intact and a secret as well as to maintain repudiation.
Q6 Compatibility	The quality that a product does not disturb or can even work together with other products.
(E)	**Environmental Sustainability** [12], [13]
E1 Energy Efficiency	The level of energy performance of the software and the amount of energy resources used, under stated conditions.
E2 Resource Efficiency	How efficiently the resources used by the product when performing its functions and/or serving useful workload.

study, we only adopt internal characteristic as quality criteria. The criteria defined by model are relevant to all software products and computer systems [29].

The main motivation of choosing the standard is that it becomes the most well-known software quality model in practice and it covers all well-known quality characteristics.

2) Environmental Criteria: Environmental criteria were adopted from our previous works [12], [13]. In order to analyze environmental dimension of sustainability, the classification is important since all the interdependencies related to both quality and environment are elicited for the analysis. Similar classification may also be used to identify other aspects of sustainability dimensions that correspond to quality requirements.

Environmental sustainability aims at improving human welfare while protecting natural resources [32]. If the software product is considered, this dimension aims at addressing ecological requirements including energy efficiency [33]. In software intensive systems, it does not consider only the energy efficiency and optimization. However, main motivation here is that energy efficiency is the most important attribute and it has direct effect *(first order effect)* on the environmental sustainability. First order effects are the immediate opportunities and effects created by the physical existence of software as a product and process involved in its design and production [33]. Although second- and third-order effects are very important for an informed decision-making on sustainability-related software, the main focus in this study is to identify relations on the first order effects. Computing resources (memory, processing, network bandwidth, and storage) are the principal source of consumption within the software system. Given a monitoring of energy consumption over certain time frames, energy efficient resource usage possibilities may be spotted and subsequently applied. For this reason, resource efficiency is adopted as another characteristic of environmental quality of the product.

C. Conducting the Questionnaire

The questionnaire was distributed online via e-mail to the participants. At the questionnaire the practitioners were given a short introduction which included positioning the questionnaire such that this is an international collaborative research project and the main research objective. The practitioners were also guaranteed anonymity.

The first part of the questionnaire contains an introduction and the context. The second part introduces the criteria that are shown in Table 1. The third part is the actual questionnaire. The eight criteria were listed in a table and the practitioners were asked to fill out different columns with respect to the given factor (quality and environment). The practitioners were asked questions to indicate the level of importance of the each criterion regarding the importance of the factors (quality and environment). For example, the they were asked "How important is functional suitability with respect to environmental sustainability", "How important is functional suitability with respect to resource efficiency". The questions were inspired from [34] We used Likert scale with nine response categories (1-9) [35]. The nine-point scale has been shown to reach the upper limits of the scales reliability [36]. A score of 1 indicates not important and 9 represents the extremely important. The higher the value, the more important the criterion is.

IV. Results and Analysis

The questionnaire was initially sent to one company to check if the questionnaire was easy to understand and if there is major problem existed. this provided a validation of the questionnaire. It was validated that the questions are clear and understandable, no changes were needed. Later, it is sent to six more companies. The results presented here are based on the responses of seven companies. In total, 53 practitioners responded from those seven companies. The participants have between 5 to 20 years of software engineering experience.

A. Data Analysis

We used Spearman correlation analysis in order to find the correlation between quality and environmental criteria. A correlation is the measurement of the relationship between two variables. A positive and negative correlation simply indicates that there is a relationship between the two variables. The most important concept is that correlation does not indicate causation. Spearman correlation is a measure of the existence and strength of the relationship between two variables [37]. Here we used Spearman test to determine the importance of the relationships between quality and environmental criteria. The important elements of the test are the data distribution does not necessarily follow the normal distribution and the data must be ordinal. As we used 1 to 9 Likert scale to determine the the rank of importance, Spearman correlation

TABLE II: Descriptive statistics and correlations for E1

	Q1	Q2	Q3	Q4	Q5	Q6
Q1						
Q2	0.22					
Q3	0.25	0.13				
Q4	-0.28*	0.01	-0.08			
Q5	0.03	-0.33*	0.43**	-0.20		
Q6	-0.42**	-0.13	-0.53**	0.34*	-0.42**	
*correlation is significant at $p < 0.05$						
**correlation is significant at $p < 0.01$						

test suits best to our data and distribution type. As a result of this analysis, the relationship between criteria is expressed by a value between -1 and 1. The values close to 1 or -1 indicate high correlation. Positive values represent positive correlation in the same direction while negative values indicate correlation in the opposite direction. We took the significance level as 0.05. We applied correlation analysis on each criterion separately and obtain p and StdErr values for each of them. We used SPSS Statistics software package for statistical analysis. After correlation analysis, we performed regression analysis on criteria whose correlation analysis yields significant results. We created and evaluated regression models in which the independent variables are quality criteria and the dependent variables are the environmental criteria. In order to evaluate the explanatory power of the regression models, we used the R^2 coefficient. It is the ratio of the regression sum of squares to the total sum of squares [37]. R^2 ranges from 0 to 1, and the higher the value is, the more variability is explained by the model, i.e., the better the explanatory power of the model is. Another indicator of the explanatory power used is the adjusted R^2. This takes into account the degrees of freedom of the independent variables and the sample population.

B. Correlation Analysis Results

We analyzed the relations regarding the importance of *energy efficiency (E1)* criteria. For six quality criteria using Spearman correlation (Table II), we found that functional suitability is significantly negative correlated with usability (-0.28) at the 0.05 level and compatibility (-0.42) at the 0.01 level. Another negative significant correlation was found between reliability and compatibility (-0.53); security and compatibility (-0.42); at the 0.05 level. A positive correlation was found between reliability and security (0.43) at the 0.01 level.

Regarding the importance of energy efficiency, the negative correlation reveals that functional suitability has negative correlation with usability and compatibility. That can be interpreted as the level of functional suitability increases with the decreasing level of usability and compatibility. However, this correlation does not imply the causality. Usability focuses on efficiency of use. Its goals are easy to accomplish quickly and with a few or no users errors, customer acceptance and how well the customer can use the product to complete the required task. The way in which energy is consumed is the result of the customers characteristics and the way in which they use the product. In this sense, energy efficient product usability requirements need to build around people and business objectives. In contrast, developers are most comfortable with the functions and tend to focus on them. As a result, in most development environments, usability requirements are less constructed. Instead, the developers agree on a basic functions that are the most desired. Principally, developers can and should embrace and care about energy efficiency and usability, just as much as they embrace functionality based development.

Another negative correlation lies between security and compatibility. Ability of the software to work with other systems provide compatibility of the product, on the other hand working with different platforms/operating systems may create a security risks. This is not only under the operating system level but also in resource consumption.

The positive correlation between reliability and security may help ascertain the faults and defects, preventing mishaps and be helpful to establish the behavior of the software product with respect to the energy efficiency of system that it is deployed on. Knowing this positive correlation also helps to distinguish system reliability failures and systems security failures for analysis at the time of system design. The negative correlation of reliability and compatibility implies that reliability is greatly influenced by the compatibility of the software. This may be used for the decision to adopt the software product. For example, even if the product is 95% compatible, the remaining 5%, may result in breakdown and this affects reliability and energy efficiency negatively.

Regarding *resource efficiency (E2)* we found the similar results with energy efficiency shown in Table III. There are negative significant correlations between functional suitability and usability (-0.28), compatibility (-0.50), respectively.

Another negative correlation is fond between performance efficiency and security (-0.41) at 0.01 level. The only positive significant correlation are found between reliability and security (0.51) at 0.01 level.

C. Regression Analysis Results

Following our correlation analysis, we applied regression. We built regression models using quality criteria as independent variables. We also tested the statistical significance of the regression models using the F-test. We ran the stepwise regression analysis to look at the contribution of each quality criteria (Table IV, V) to regression models. Table IV shows the stepwise regression models for energy efficiency.

Model 1 was run for functional suitability (Q1). The model is significant at the level of 0.05. When we add *performance efficiency (Q2)*, surprisingly, Model 2 leads to reduction of Adjusted R^2 from 0.11 to 0.10. However, addition of *reliability (Q3)* to regression model (Model 3) R^2 increases and *F change* is significant. These results indicate that, reliability is an

TABLE III: Descriptive statistics and correlations for E2

	Q1	Q2	Q3	Q4	Q5	Q6
Q1						
Q2	0.25					
Q3	0.17	-0.26				
Q4	-0.28*	0.04	-0.11			
Q5	0.09	-0.19	0.51**	-0.02		
Q6	-0.50**	-0.16	-0.41**	0.17	-0.43**	
*correlation is significant at $p < 0.05$						
**correlation is significant at $p < 0.01$						

TABLE IV: Stepwise Analysis-R^2 Changes for energy efficiency

Model	R^2	Adjusted R^2	R^2 Change	F Change	Sig. F Change
1	0.128	0.112	0.128	7.665	0.008
2	0.143	0.109	0.014	0.844	0.363
3	0.424	0.389	0.281	24.423	0.000
4	0.567	0.531	0.143	16.155	0.000
5	0.649	0.612	0.082	11.178	0.002
6	0.665	0.622	0.016	2.316	0.135

important quality criterion considering energy efficiency. This result is also supported by the correlation analysis results (see section VI-B) and the prioritization analysis results in Akinli Kocak et al [13]. In addition to reliability criterion, usability is also seen as an important criterion for energy efficiency (Model 4). Moreover, adding security (Table V-Model 5) also contributes to increase R^2. However, the contribution of compatibility criterion to the models is very low and the changes in F value are not significant.

As seen in the Table V, Model 1 was run only with *functional suitability (Q1)*. The model is sufficient, however when we added the *performance efficiency (Q2)*, this leads to

TABLE V: Stepwise Analysis-R^2 Changes for resource efficiency

Model	R^2	Adjusted R^2	R^2 Change	F Change	Sig. F Change
1	0.612	0.604	0.612	81.978	0.000
2	0.729	0.718	0.117	22.001	0.000
3	0.786	0.774	0.058	13.457	0.001
4	0.834	0.821	0.048	14.187	0.000
5	0.835	0.818	0.001	0.311	0.579
6	0.836	0.815	0.000	0.091	0.764

TABLE VI: Regression analysis Results

	Model A (DV: E1)		Model B (DV: E2)	
Variables	**Beta**	**VIF**	**Beta**	**VIF**
Q1	-0.239*	3.21	0.234*	3.60
Q2	-0.,223*	3.74	0.163*	3.43
Q3	0.467**	3.94	0.263*	3.68
Q4	0.834	3.11	0.424**	3.33
Q5	0.404**	2.53	-0.053	2.73
Q6	0.285	2.01	0.039	1.92
R^2	0.665			0.83
Adjusted R^2	0.622		0.81	0.091
F	15.559***		39.845***	
Durbin-Watson	2.022		1.52	
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$				

significant increase of R^2 (from 0.61 to 0.73) in Model 2. This means that performance efficiency has a high effect on resource efficiency. Similarly, usability and reliability significantly increase the R^2. Interestingly, we could not observe the same result for security and compatibility. This analysis results reveal that, how the system behaves with respect to energy efficiency is highly influenced by reliability, performance efficiency and usability.

We ran the regression to investigate the unique contribution of each criterion on both energy efficiency and resource efficiency (Table VI). We also tested for collinearity among any variables by calculating the variance inflation factor (VIF) for each of the regression coefficients. Since all the values of VIF are below 10, multi-collinearity is not a problem [38]. The regression results in Model A show that, reliability has the highest effect on energy efficiency. The reliability is a failure-free operation. This also means the actual usage time of the product by user. Therefore, the reliability is correlated with the efficiency of software over time. As time passes, the energy efficiency increases. Surprisingly, we found that security criterion has also high effect on energy efficiency. An increase in the functional suitability and performance efficiency lead to a decrease in the energy efficiency. We could not find any significant correlation between compatibility and energy efficiency.

In Model B, we ran the regression for resource efficiency. The results show that *usability criterion (Q5)* has the highest impact on resource efficiency, followed by reliability, functional suitability and performance efficiency. Even though performance efficiency increases with the increase in resource efficiency, the effect is not very pronounced. No significant correlations have been found regarding compatibility.

We can conclude that performance efficiency, usability and functional suitability and reliability measures are important indicators for the development of environmentally sustainable software product.

V. Threats to Validity

We discuss four types of threats to validity: construct validity, internal validity, external validity, and reliability.

1) Construct validity: Regarding construct validity, one limitation is the selection of the criteria. We chose the quality criteria from the well-known quality model [29]. We selected the environmental criteria from the related literature studies explained in Section III/B. Environmental concerns are rising more attention and will consequently gain more importance in software projects. Hence, the environmental criteria might be different for different applications. Moreover, in the scope of our study, we chose the criteria related to the direct effect on environmental sustainability.

Another limitation is mono-operation bias which refers to problems with single exemplars of a level of independent variable or a single measure of dependent variable This is minimized by including different companies into the study. Another threat to construct validity is evaluation stress which is limited by guaranteeing anonymity to the participants which had been done before the questionnaire.

2) Internal validity: Internal limitation may be the selection of practitioners for the questionnaire. In order to mitigate this threat we conducted our questionnaire with the practitioners who have a sufficient amount of experience (5-20 years) in software engineering domain, mostly in quality assurance. Additionally, confounding factors influencing measurements are another thread to internal validity. In order to mitigate this threat, we performed statistical regression. The focus in this study is determining the correlations not establishing a causal relationship. Therefore, we have only collected practitioners opinion on how the environmental sustainability relates to product quality.

3) External validity: It refers to generalization of the results of this work. The number of questionnaires filled by the practitioners may be seen as an external threat to validity. Various rules-of-thumb have been suggested in the literature for determining the minimum number of subjects required to conduct multiple regression analyses. Based on the literature review and Green‘s [39] approach, sample size of over 50 is sufficient for our study. The final threat is that we did not make any differentiation between different types of software engineering projects and the role of practitioners. These limit the range of issues that could be identified.

4) Reliability: It is a threat that the results of the research are influenced by interpretation. In order to mitigate this threat, data analysis had been conducted using statistical analysis techniques.

VI. Conclusion and Future Work

In this paper, we present an analysis of the relationship between environmental sustainability and software quality from the standpoint of software development practitioners. We analyze the effect of the quality criteria on sustainability.

The results from the survey conducted at seven companies with 53 practitioners are reported. Analysis results show that there is significant correlation between energy efficiency and quality criteria. The regression analysis results indicate that using quality and energy efficiency for designing and analyzing environmental sustainability of the product may be useful. Practitioners should be careful in using resource efficiency, since the resource efficiency measures may not be consistent. Our study may guide practitioners especially requirement engineers when defining quality requirements that specify what environmental sustainability means in terms of qualitative characteristics of a software product. We believe this work would provide hard evidence to requirement engineers on the relations of quality and environmental requirements while taking these requirements into account during the requirements process.

There is a tendency to threat environmental sustainability as a quality of the system once other priorities are set. However, we recommend that environmental sustainability should be considered in balance with the existing quality attributes of the system. The analysis in this paper also helps practitioners to better understand the relations and impact of environmental factors with quality attributes to make informed feasibility analysis.

Going forward, we would like to extend the set of criteria.

Acknowledgments

This research has been fully funded by the Mitacs Globalink Program under grant number 1-51-52278.

References

[1] R. Goodland, "Encyclopedia of global environmental change, chapter sustainability: Human, social, economic and environmental," 2002.

[2] 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.

[3] B. Penzenstadler, "Infusing green: Requirements engineering for green in and through software systems," *Christopher Arciniega, Birgit Penzenstadler TechReport UCI-ISR-14-2 June*, 2014.

[4] C. Calero and M. Bertoa, "25010+s: A software quality model with sustainable characteristics: sustainability as an element of software quality," in *Proceedings of the 2013 workshop on Green in/by software engineering*. ACM, 2013.

[5] C. C. Venters, L. Lau, M. K. Griffiths, V. Holmes, R. R. Ward, C. Jay, C. E. 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, p. e8, 2014.

[6] B. Penzenstadler, A. Raturi, D. Richardson, and B. Tomlinson, "Safety, security, now sustainability: The nonfunctional requirement for the 21st century," *Software, IEEE*, vol. 31, no. 3, pp. 40–47, 2014.

[7] C. Becker, "Sustainability and longevity: Two sides of the same quality?" *mental*, vol. 20, p. 21, 2014.

[8] C. Becker, R. Chitchyan, L. Duboc, S. Easterbrook, B. Penzenstadler, N. Seyff, and C. C. Venters, "Sustainability design and software: The karlskrona manifesto," in *ICSE'15: Proceedings of the International Conference on Software Engineering*, 2015.

[9] 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.

[10] M. R. de Souza, R. Haines, and C. Jay, "Defining sustainability through developers eyes: Recommendations from an interview study," Technical Report 1111925, figshare, 2014. http://dx. doi. org/10.6084/m9. figshare. 1111925, Tech. Rep., 2014.

[11] L. M. Hilty, W. Lohmann, S. Behrendt, M. Evers-Wlk, K. Fichter, and R. Hintemann, "Ict for sustainability: An emerging research field," Technical Report (UBA-FB) 001883/2,E, Tech. Rep., 2015.

[12] S. A. Koçak, G. G. Calienes, G. I. Alptekin, and A. B. Bener, "Requirements prioritization framework for developing green and sustainable software using anp-based decision making." in *EnviroInfo*, 2013, pp. 327–335.

[13] S. A. Koçak, G. I. Alptekin, and A. B. Bener, "Evaluation of software product quality attributes and environmental attributes using anp decision framework," in *Proceedings of the Third International Workshop on Requirement Engineering for Sustainable Systems (pp. pp. 37-44). Karlskrona: Central Europe Workshop Proceedings*, 2014.

[14] B. Commission *et al.*, "World commission on environment and development. our common future," 1987.

[15] R. C. Seacord, J. Elm, W. Goethert, G. A. Lewis, D. Plakosh, J. Robert, L. Wrage, and M. Lindvall, "Measuring software sustainability," in *null*. IEEE, 2003, p. 450.

[16] C. Becker, R. Chitchyan, L. Duboc, S. Easterbrook, M. Mahaux, B. Penzenstadler, G. Rodríguez-Navas, C. Salinesi, N. Seyff, C. C. Venters, C. Calero, S. A. Koçak, and S. Betz, "The Karlskrona manifesto for sustainability design," *CoRR*, vol. abs/1410.6968, 2014. [Online]. Available: http://arxiv.org/abs/1410.6968

[17] C. C. Venters, M. K. Griffiths, V. Holmes, R. R. Ward, and D. J. Cooke, "The nebuchadnezzar effect: Dreaming of sustainable software through sustainable software architectures," Technical Report 1112484, figshare, 2014. http://dx. doi. org/10.6084/m9. figshare. 1112484, Tech. Rep.

[18] H. Cabezas, U. Diwekar, J. Beck, B. Beloff, B. Bakshi, J. Crittenden, J. Farley, H. J. Fernando, S. P. French, A. Garmestrani *et al.*, *Sustainability: Multi-Disciplinary Perspectives*, 2012.

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

[20] J. Karlsson and K. Ryan, "A cost-value approach for prioritizing requirements," *Software, IEEE*, vol. 14, no. 5, pp. 67–74, 1997.

[21] B. Regnell and S. Brinkkemper, "Market-driven requirements engineering for software products," in *Engineering and managing software requirements*. Springer, 2005, pp. 287–308.

[22] J. Cabot, S. Easterbrook, J. Horkoff, L. Lessard, S. Liaskos, and J.-N. Mazón, "Integrating sustainability in decision-making processes: A modelling strategy," in *Software Engineering-Companion Volume, 2009. ICSE-Companion 2009. 31st International Conference on*. IEEE, 2009, pp. 207–210.

[23] E. S. Yu, "Towards modelling and reasoning support for early-phase requirements engineering," in *Requirements Engineering, 1997., Proceedings of the Third IEEE International Symposium on*. IEEE, 1997, pp. 226–235.

[24] D. Stefan, E. Letier, M. Barrett, and M. Stella-Sawicki, "Goal-oriented system modelling for managing environmental sustainability," in *International Workshop on Software Research and Climate Change (WS-RCC)*, 2010.

[25] M. Mahaux, P. Heymans, and G. Saval, "Discovering sustainability requirements: an experience report," in *Requirements engineering: foundation for software quality*. Springer, 2011, pp. 19–33.

[26] Q. Gu, P. Lago, and S. Potenza, "Aligning economic impact with environmental benefits: A green strategy model," in *Proceedings of the First International Workshop on Green and Sustainable Software*. IEEE Press, 2012, pp. 62–68.

[27] B. Penzenstadler, M. Kuhrum, and K. Petersen, "Towards incorporating sustainability while taking software product management decisions," in *in International Workshop on Software Product Management (IWSPM 2013)University of Duisburg-Essen*. IEEE Press, 2013.

[28] P. Lago, N. Meyer, M. Morisio, H. A. Müller, and G. Scanniello, "Leveraging energy efficiency to software users," 2014.

[29] ISO/IEC25010:2011, "Systems and software engineering – systems and software quality requirements and evaluation (square) – system and software quality models," 2011.

[30] E. Kern, M. Dick, S. Naumann, A. Guldner, and T. Johann, "Green software and green software engineering–definitions, measurements, and quality aspects," *Hilty et al.(2013)*, pp. 87–94, 2013.

[31] C. Wohlin and A. Aurum, "Criteria for selecting software requirements to create product value: An industrial empirical study," in *Value-based software engineering*. Springer, 2006, pp. 179–200.

[32] F. Berkhout and J. Hertin, "Impacts of information and communication technologies on environmental sustainability: Speculations and evidence," *Report to the OECD, Brighton*, vol. 21, 2001.

[33] L. M. Hilty, P. Arnfalk, L. Erdmann, J. Goodman, M. Lehmann, and P. A. Wäger, "The relevance of information and communication technologies for environmental sustainability–a prospective simulation study," *Environmental Modelling & Software*, vol. 21, no. 11, pp. 1618–1629, 2006.

[34] T. L. Saaty, *Decision making with dependence and feedback: The analytic network process*. RWS publications Pittsburgh, 1996, vol. 4922.

[35] R. Likert, "A method of constructing an attitude scale," *Scaling: A sourcebook for behavioural scientists*, pp. 233–243, 1974.

[36] I. E. Allen and C. A. Seaman, "Likert scales and data analyses," *Quality Progress*, vol. 40, no. 7, pp. 64–65, 2007.

[37] G. Keller, *Statistics for Management and Economics, Abbreviated*. Cengage Learning, 2015.

[38] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to linear regression analysis*. John Wiley & Sons, 2012, vol. 821.

[39] S. B. Green, "How many subjects does it take to do a regression analysis," *Multivariate behavioral research*, vol. 26, no. 3, pp. 499–510, 1991.