Software Sustainability Characteristic for Software Development towards Long Living Software

RUZITA AHMAD, AZHAM HUSSAIN & FAUZIAH BAHAROM
School of Computing
University Utara Malaysia
06010 UUM Sintok
KEDAH
rita_azura08@yahoo.com, azham.h@uum.edu.my, fauziah@uum.edu.my
https://sites.google.com/site/drazhamhussain/

Abstract: Software sustainability is a kind of software endurance that capable to lead the software development to be surviving in a long time period with employing the rules of sustainability paradigm. Even though, the jumble of identification of characteristics and sub-characteristic towards software sustainability in the presenting works recently are un-certainty declaration in utilizing the justification of the requirements towards achieving software sustainability. This paper investigates the state-of-the-art of software development towards software sustainability by identifying the important characteristic and sub-characteristic with organizing into the three pillars dimension of sustainability which are environmental, economic, and social. The Systematic Literature Review (SLR) was used to discover the holistic understanding and views on software sustainability, including the recent processes, activities, performance, and limitations that relates to the important of characteristic and sub-characteristic of software development. The results show the important features for software development towards software sustainability with embedding the essential criteria for each characteristic that are related to sustainability paradigm.

Key-Words: - Software Sustainability, Sustainable Characteristic and Sub-characteristic, Systematic Literature Review, and Sustainability Paradigm

1 Introduction

In the new globalization era, software production is dressed in the new clothes as to claim a novelty with future requirements that starts from the existing software infrastructure and ends with tightly integrated software systems realized by distributed systems that can support the core business process of the company [6]. Nowadays, the software development is developed based on the reused software and has been driven by the different paradigms [46]. The concept of heterogeneous software acts as a main role in supporting the business process with the capability in sustaining the high dimensionality of data. For instance, the client server architectures and systems are distributed based on several databases, transactions of processing monitors, and involved the application of web servers. For example, the critical software in the industry organizations is constantly being integrated with reused of application provided by Software-as-a-Service (SaaS), Enterprise knowledge sharing (know-how), Enterprise Application Integration (EAI), and the latest one is Service Oriented Architecture (SOA) [3], [48].

Recently, the integration of applications has their own credibility and powerful role in ensuring the long living software in order to increase the accountability, reliability and functionality of the software systems. In addition, the capability of technologies is claimed to reduce costs, positive returns to the subscription models, and to improve usability in the software systems [26]. Literatures reveal that the issue to develop software for the long living system is closely related to sustainability paradigm and it is already highlighted in this new era's software production [5], [8], [19], [20], [21], [22], [23], [31], [33], [43], [53]. However, the concept of sustainability by integrating the three pillar dimensions which are environmental, economic, and social does not exist in the integration of technologies used today. Unfortunately, the integration of the applications in the critical software is claimed as non-sensitive in influencing the impacts of sustainability [7], [25], [28].

Besides, spending on Information and Communication Technology (ICT) equipments with un-sustainable features and application will be

influenced to the system development process and indirectly will be reflected in the environmental, economical and societal impacts [12]. Since SOA adoption is hired as a new integration approach for long living systems, therefore this study will be focused on identifying the characteristic and sub-characteristic based on the behavioral characteristic and sub-characteristic of SOA as the specific features in software sustainability.

The [70] has defined sustainability as meeting the needs of the present generation without compromising the ability of future generations to meet their own needs. Sustainability has been practiced in the various fields such as in manufacturing [24], [51], construction [50], restoration of natural disasters [49], [54], soils and erosions [49], [51], and ecosystems and biodiversity [49], [54]. Sustainability in software engineering is just began ended 2009 in which the issue has been recognized as an important topic that is needed to be highlighted in software development [4], [22], [34], [35].

As mentioned earlier, sustainability is strongly related to long living software in which the regardless to highlight sustainability in software development will be influenced to the system with poor quality. This scenario will be reflected into the strategies of efficiency to achieve profitability and also reliability with the aimed to improve and recover the risks of the system failures and errors in the future [1], [4], [5], [8] and [9]. Unfortunately, the systems architectures today are claimed as poor quality in handling the changes and transformation process to meet the goal in sustainability impacts [31], [32]. For instance, the software systems are lacking of consistency between the system and user in which the software architectures are not supporting the user action in handling the changes in the environment [20], [34].

In order to master all changes within the software development towards long living systems, the continuously of evaluation process in software sustainability is extremely important to lead the achievement of sustainability [8], [19], [20], [34]. Therefore, the compulsory to take into consideration in defining the characteristic and sub-characteristic of software production and the reused application by structuring the characteristic and sub-characteristic into the three dimensions of sustainability is significantly required.

2 Systematic Literature Review (SLR) Methods

The identification of sustainability characteristic and sub-characteristic of software development towards long living software in this study is developed by applying the SLR method. Several steps involve in SLR such as planning the review by identifying the SLR research question, conducting the review through the searching process, assessing the quality of searching query, select the primary studies through the inclusion and exclusion criteria, data analysis and synthesis data and reporting the results. The steps are further explained in the next sub-sections.

2.1 Planning the review

In the first step of SLR will be involved planning the review through the SLR research questions. The aimed of the review is to develop a body of knowledge on sustainability studies in software engineering. In order to ensure the research objective of this study such as to identify the characteristic and sub-characteristic of software development towards long living software and organizing into three dimensions of sustainability, several questions have been developed. This is further detailed in the following research questions are presented in the Table 1. Table 1 illustrates the questions created to generate the holistic view on sustainability for software engineering.

Table 1 Holistic view on sustainability for software engineering

No	Questions for software sustainability development
RQ1	How much the research activities in sustainability in the last 20 years?
RQ2	How much research activity in sustainability for software engineering?
RQ3	How was sustainability support performed in software development towards long living software?
RQ4	What are the characteristic and sub-characteristic of software development towards software sustainability?
RQ5	What are the characteristic and sub-characteristic of software development been proposed are stressing on the dimension of sustainability, i.e. environment, economic, and social individually?

2.2 Conducting the review

The research questions in Table 1 are used in the searching process in this study based on the following digital libraries presented in the Table 2 below.

Table 2 Search digital libraries

Digital libraries	Address
IEEE Digital Library	http://ieeexplore.ieee.org
ACM Digital Library	http://dl.acm.org
SpringerLink	www.springerlink.com
ScienceDirect / Scopus	http://www.sciendirect.com
Web of science	http://apps.webofknowledge.com

The search process is a manual search of specific digital libraries, including the references of journals, books, proceedings and other academic materials. The sources were selected because they were known to include either empirical studies or literature surveys and have been used as sources for other SLR studies related to sustainability in software engineering [31] and [32]. The aim of the search strategy is to capture and screening all results that relates to the SLR research questions that have been developed.

The searching process is based on "sustainability" term in general and focused in the context of software engineering such as "sustainability AND software engineering", "sustainability AND characteristic / sub-characteristic for software development", and "sustainability AND characteristic and sub-characteristic in three dimensions of sustainability". The aim of this phase is to investigate the history of software sustainability existence based on sustainability research in the literatures. An overview of sustainable development in software engineering discipline with observing the research activity and related topics, characteristic and sub-characteristic and limitation of the previous studies are examined. Therefore, the component and structure of characteristic and sub-characteristic of software development in the integration of dimensions are investigated.

2.3 Inclusion and exclusion criteria

In this step, almost 239,688 articles are related to sustainability via the mapping date from 1995 until 2015 is screening. In stage 1, identification of search topic is executed on each database. None related articles on sustainability in software engineering are excluded and the related articles are almost 3,645 articles are included in the second stages. The gathered articles are revised and reviewed based on titles, abstracts and conclusions that are related to software sustainability metric evaluation and classifies the papers and articles according to the type, topic, and domain. As the result, 1,232 articles are related and moved to the third stages. The articles are revised and reviewed based on characteristic and sub-characteristic of software development. Almost 720 articles are excluded in the database and 512 articles are included in the database that are related to the title, abstract, and conclusion. Once again, the total of included articles is revised and reviewed with characteristic and sub-characteristic of software development and organized in three dimensions of sustainability. As the results, 88 articles are most related and will be used to support the body of knowledge as to achieve the objective stated in this study.

2.4 Data analysis and synthesis data

The articles finding from the digital library database will be classified according to the research type facets and knowledge area [31] and [32]. The terms are distributed according to the metadata, knowledge area, research type facets, and domain. The details of the terms used are listed below:

1) **Metadata:** Authors, year of publication, title, source, keywords, research topic and institution.
2) **Knowledge area:** Software sustainability, software requirements, software engineering metric evaluation, software engineering process, software quality, sustainability in flood management, sustainability modeling, sustainability concept, sustainable software solution, software sustainability evaluation, sustainability standard, characteristic and sub-characteristic for software sustainability in three dimensions of sustainability.
3) **Research type facets:** Philosophical, exploratory, solution, validation, evaluation, opinion and experience.
4) **Domain:** Software sustainability, software sustainability metric evaluation and software engineering.

The related articles are distributed into the related questions in the SLR research question. The aimed is to select the articles to answer the SLR research question that relates to the characteristic and sub-characteristic of software development and will be used in organizing the characteristic and sub-characteristic into three dimensions of sustainability. The results will be performed in the next section.

3 Results

The results presenting in this section are the reporting results performed pertaining to answer the SLR research questions stated in this study and followed by the reporting of the final results in identifying the sustainability characteristic for software development towards long living software.

3.1 SLR research questions results

RQ1: How much the research activity in sustainability in the last 20 years?
The number of results for each database is listed in the Table 3 by using the searching query keyword such as "sustainability".

Table 3 Number of screening articles per database

Database	Mapping date	Results
IEEE Xplore	Sept 2013- Mac 2015	4,404
ACM Digital	Sept 2013- Mac 2015	3,721
Springerlink	Sept 2013- Mac 2015	90,288
ScienceDirect/ Scopus	Sept 2013- Mac 2015	92,820
Web of science	Sept 2013- Mac 2015	48,455
Total articles		**239,688**

The results performed in the Table 3 above are the screening of articles based on the selected databases. Almost 239,688 articles have been produced pertaining to sustainability in every domain of investigation since 20 years ago. As analyzed the results from the investigation, sustainability has been applied in every field and domain in the realize world, for example in manufacturing, construction, restoration of natural disasters, soils and erosion, ecosystem and biodiversity, natural sciences and so forth. The domain applied such as in robotics, digital ecosystems, e business, security, forestry, networking, environment management, agriculture and many more.

RQ2: How much research activity in sustainability for software engineering?
The results are performed per each database by searching using the query keywords:"sustainability in software engineering". Table 4 shows the selected article from the database.

Table 4 Selected article by database

Database	Results related to sustainability in software engineering
IEEE Xplore	226
ACM Digital	1,486
Springerlink	990
ScienceDirect/ Scopus	458
Web of science	485
Total	**3,645**

The articles that are related to the research activities in sustainability for software engineering have covered the issues in method, model, methodology, metric, framework, tool, approach, concept, review, case study, evaluation, assessment, and many more. Therefore, the categories of issues are investigated on the constructive, empirical and also in the discussion based on the related issues.

RQ3 How was sustainability support performed in software development towards long living software?
Dealing with software development towards long living software is related to sustainability paradigm. Sustainability has been support performed the software development through the concept, frameworks, model, methods, methodology, tool/approaches, characteristics, perspectives and metrics. Table 5 illustrates the number of included results related to objective in this study and Table 6 represented the number of related articles that relates to the contents of each database.

Table 5 Included results related to objective in this study

Database	Related results to objective in this study
IEEE Xplore	76 out of 226
ACM Digital	54 out of 1486
Springerlink	66 out of 990
ScienceDirect/ Scopus	15 out of 458
Web of science	14 out of 485
Total	**225 out of 3,645**

Table 6 Included results per content type

Category	Type of content	Number of results
Constructive	Method	6
	Model	17
	Methodology	4
	Metric	5
	Framework	15
	Tool/Approach	28
	Characteristic	22
Empirical	Concept	27
	Review	25
	Case study	7
	Evaluation	3
	Assessment	19
Discussion	Overview	5
	Challenges	3
	Integration	25
	Perspective	7
	Dimension	7

Some of the articles present an empirical study such as review paper, case study, evaluation and assessment. Furthermore, in the context of discussing the articles present an overview, challenges, analysis, perspective and dimension that relates to the software development towards long living software.

RQ4 What are the characteristic and sub-characteristic of software development towards software sustainability?
As referred to the Table 6, the included results performed 22 articles that are related to the characteristic and sub-characteristic in the software development. The results are fully discussed in the section 3.2.

RQ5 What are the characteristic and sub-characteristic of software development that have been proposed in stressing on the dimension of sustainability such as environmental, economic, and social individually?
The included results based on the articles of characteristic and sub-characteristic of software development is reviewed and revised in term of their contents as to find out either the researchers have been utilized the concept of sustainability or otherwise. As the results, the investigation found that none of the related articles are organizing the component of the characteristic and sub-characteristic into three dimensions of sustainability i.e. environment, economic, and social. For instance, several articles are only proposed the characteristic and sub-characteristic into environment and economic dimension without stressing on the social dimension. Furthermore, some of the articles are focused on environmental and social dimension without stressing on economic dimension. Consequently, the investigation met the conclusion on proposing the characteristic and sub-characteristic in the previous studies is claimed as sustainable with highlighting the issues of sustainability and regardless to stress on the dimensions individually. Further explanation is fully reviewed in the reporting the results.

3.2 Reporting the final results

The identification of sustainability characteristic and sub-characteristic of software development towards long living software are derived from the principal of sustainability criteria that has been produced by [70] and supported by the previous studies in the domain of software engineering. Several studies on sustainability characteristic performed more supported ideas in bringing the information and guideline in identifying the sustainability characteristic in the various domains, especially in software engineering [12], [33], [34], [47], [48], [54], [57] and [71]. The next sub-sections are furthered by the explanation on the reviewed of the important characteristic in identifying the sustainability characteristic and sub-characteristic of software development towards long living software.

3.2.1 Principal concept of sustainability characteristic

The principal concept of sustainability characteristics is mostly used in many domains towards developing the sustainability. The issue that has been highlighted by [70] is created in a form of question such as “what is to be sustained” will be pointed into the three major categories such as nature, life support systems, and community [45]. There are three distinct categories to be highlighted which are created in question like “what is to be developed”. Therefore, it is divided into the people, economy, and society. In order to create the characteristic as the principal characteristic for sustainability, most of the literatures had focused on the economic dimension as a productive sector with supplying the employment and desired consumption and wealth [70]. This means the economic dimension is presenting the incentives and it will be resulted in the investment as well as funds for environmental maintenance and restoration [67].

Sustainability is multifaceted concept and it is dynamic condition that can be applied in any ways of development. Each individual or organizations are needed to know the rules and criteria in sustainable development in order to achieve sustainability. This involves the knowledge of each party that requires to work with sustainability are compulsory to identify the general concept of appropriate characteristics as referred to the criteria and indicators published by the United Nation organization in their own categories of development, management or evaluation. The [70] suggested a set of principle characteristic that proposed by the Brundland Commission Report are most recently focused has to be pointed to people with stressed on human development, life expectancy increment, education, equity, and opportunity. Furthermore, all of the elements need to re-build in developing society emphasizing the well-being and also the security in the country, region, states, and institutions or organization. Thus, in the Brundlant Commission Report has suggested the created characteristics are compulsory accepted the objective and goals of sustainability, which is to fulfil the requirements and needs at the present until to the future generations for long living quality of life. Table 7 illustrates the taxonomy of sustainable development introduced by the Brundtland Commission Report in 1987 [45].

Table 7 Taxonomy of sustainable development (Source: Paris et al. 2003)

	What is to be sustained:	What is to be developed:
1	**Nature**	**People**
	Earth	Child survival
	Biodiversity	Life expectancy
	Ecosystems	Education
		Equity and Equal opportunity
2	**Life support**	**Economy**
	Ecosystems services	Wealth
	Resources	Productive sectors
	Environment	Consumption
3	**Community**	**Society**
	Cultures	Institutions
	Groups	Social capital
	Places	States
		Regions

3.2.2 Sustainability characteristic in software engineering

Sustainability characteristic is strongly related to the definition and the concept of sustainability paradigm. Most of the researchers have defined the set of sustainability characteristics and sub-characteristics based on their own experience, theory, views and deep understanding of sustainability. The concept and dimensions are the important elements in defining the characteristics towards sustainable development. In the context of software engineering, several studies had contributed a set of characteristics and indicators to sustainable software development. The most researchers are [12], [33], [34], [47], [48], [54], [57] and [71].

The research on sustainability characteristic in software engineering has initiated by [34]. The investigation proposes a set of characteristic in sustainability for long living software with highlighted the cost of maintaining the efficiency and evolved over the entire life cycle as the indicators. The proposed characteristics such are Maintainability, Modifiability, Portability, Evolvability, and Integrity. However, the researcher only focused on Maintainability characteristic with broken down into the sub-characteristics such as analyzability, stability, testability, and understandability. Besides, the researcher does not figure out the sub-characteristics for Modifiability, Portability, Evolvability, and Integrity. In addition, [34] has highlighted the Integrity characteristic as the important characteristic that need to be underlined, but unfortunately the researcher does not derive the integrity as the priority in the ranking of the sustainability characteristic as well.

According to the researcher, the sustainable development of software systems or products can be achieved by preparing a guideline to conduct the stakeholders through the sustainable requirements such as documenting, prioritizing, analyzing, and tracing functional and non-functional requirements for an industrial software system is an important prerequisite for sustainable [35], . They claimed that the sustainable requirements are much helpful in tracing and noticing for long term development systems or product as it preserves the knowledge about the decision making in every phase of development, such as in architecture, design, implementation, testing and maintenance [34], [35], [36] and [37].

In fact, each requirement should be analyzed for its potential impact on sustainability in the early stage as to improve the upfront design of the

systems. Therefore, the researcher suggested that the quality of software architectures determines the sustainability to a large extent.

In [12] had included the element of quality such as non-functional requirement in order to create a set of characteristic in sustainability. The researchers defined software sustainability as a means to improve a software product that is a part of software product's quality and recounted to non-functional requirements. They claimed that the non-functional requirements are extremely important to be highlighted in sustainability in order to focus on the requirements that need to be sustained in developing software product. In [12] has referred to the quality model such as ISO/IEC 25010 that is established in 2010 as the new standard of quality model after the revision of quality model of ISO/IEC 25000 which is the series of standards that are specified to system and software product quality requirement and evaluation namely SQuaRE model [23] and [24].

In addition, [12] has defined a set of characteristic for software sustainability that involves the elements: Energy consumption and Resource optimization which are inherited from ISO/IEC 25010 standard quality model. The ISO/IEC 25010 has proposed the quality characteristics under these two elements so called: Functionality, Reliability, Efficiency, Operability, Security, Compatibility, Maintainability and Portability that are also broken down into the sub-characteristic for each characteristic. Therefore, the researchers had added their own characteristic into the list namely Perdurability.

As remarked upon their existing work, Perdurability means the development of software products with features of long-lasting software as functionality, modifiability, reusability, changeability and adaptability [12] and [71]. These characteristics of long living software are composed as sub-characteristics of Perdurability. Therefore, the researcher has defined Perdurability as the degree to which a software product can be performed as the features of sub-characteristics in order to perform specified functions under specified conditions over a long period of time [12], [23], and [24].

In [71] had examined the characteristic and sub-characteristic of software sustainability by attaching the elements such as a composite requirement and non-functional requirements. Pertaining to the elements involved, the researchers had created a set of characteristics in developing sustainability and it is closely related to the software quality characteristic. There are Extensibility, Interoperability, Dependability, Portability, Reusability, and Scalability. The [71] had referred to the model of quality proposed by [34]. The McCall model has defined quality attributes into two levels which are quality factors and quality criteria. The quality factors can be characterized as external attributes that can be measured directly. While, quality criteria is un-measurable characteristic that can be measured in subjectively or objectively through the human expecting and behavior. Moreover, [71] applied the characteristics proposed by the McCall model as their benchmark to decompose several characteristic such as Maintainability, Reliability and Integrity as sub-characteristic of Dependability.

However, the researchers have added Safety characteristic as a new sub-characteristic into Dependability. The proposed of six attributes are then combined with the concept of threats and failure as to support the element of composite requirement. Therefore, the Efficiency attributes have been suggested as a new characteristic in the list and this makes the seven features in sustainability characteristics proposed by [71]. However, a set of characteristic proposed by [12] and [71] is totally focused on software quality model which are the attributes contributing to the quality of the system and product, while they do not explicitly state which are the attributes contributes to the concept of sustainability itself i.e. (environmental, economic and social dimension). These are the limitations occurred in a set of characteristic provided by the existing works that are needed to be catered in order to build the priority of the features in sustainability characteristic.

Next, the Software Sustainability Institute has been established in handling the development, management and evaluation of the software products. The Software Sustainability Institute is providing more information pertaining to the software sustainability and also presented a guideline to the researcher in assessing the software products towards sustainability. This private institute had developed a set of characteristic which is so called as "criterion" that is derived from ISO/IEC 9126-1 standard model in software quality. There are Usability, Sustainability and Maintainability [23] and [24]. All the criteria had been broken down into sub-criteria and each of the sub-criteria has been decomposed into a set of questions that are needed to be answered by the stakeholders during the evaluation on their software products.

Latest, [47] and [48] has investigated sustainability in the three variables, namely system, function and time that needs to be defined for setting scopes in proposing the sustainability characteristic. In [46] has provided a systematic literature review pertaining to the sustainability in software engineering that are based on two researchers such as [41] and [42]. Both investigations had fully supported [54] and [57] in proposing a guideline towards sustainability by focusing on IT changes behavior that has considerable effect on the society and the environment which are introduced by [41] and [42]. While, [47] and [48] are also focusing on the direct and indirect impacts that can be affected on economy, society, human beings, and the environment which is suggested by [41] and [42].

In [54] and [57], the presented of guideline in software sustainability is distinguished in four aspects such as two aspects focused on developing company and its processes which are development process and maintenance process aspect. Then, another two aspects are focusing on system production and system usage aspect. Each aspect will take into consideration on variables such as system, function and time. According to the researchers, the software sustainability is required to fulfil the requirement engineering in these three variables in order to achieve sustainable software.

In addition, they had proposed a set of indicator for each identified dimensions in their model such as Social, Economic, Environment, Individual and Technical. Furthermore, in their proposed generic sustainability model has been structured the guideline to achieve sustainability by values, indicators, regulations and activities which are related to each other. Table 8 summarizes the sustainability characteristic in software engineering proposed by previous studies.

All investigations in the presenting work are much help and support in identifying the features of software development towards software sustainability in this study. The following section discusses the identification of the characteristic of software development towards long living software as to achieve the objective stated in this study.

4 Development

The results provided by the systematic review applied in this study are much supported in developing the sustainability characteristic towards software sustainability. The following sections discuss the identification of the software sustainability characteristic, followed by the organization of the characteristic into sustainability dimensions and also the description of the organization characteristic into sustainability dimension.

Table 8 Sustainability characteristic in software engineering proposed by previous studies

	Researcher (s)	Types	Characteristic / Indicator
1	Koziolek (2011)	Char.	Maintainability Modifiability Portability Evolvability Integrity
2	Calero et al. (2013)	Char.	Reliability Maintainability Portability Perdurability
3	Venters et al. (2013)	Char.	Extensibility Interoperability Dependability Portability Reusability Scalability Efficiency
4	Software Sustainability Institute (2010)	Char.	Usability Sustainability Maintainability
5	Penzenstadler et al. (2013)	Indicator	Economic Environment Social Technical Individual

4.1 Identification of sustainability characteristic for software development towards long living software

Based on the investigation of a systematic review in [1] and [2] found more results based on characteristic and sub-characteristic of software development towards long living software. Several characteristics are derived from the sustainability characteristic in software engineering such as Maintainability, Functionality, Portability, Efficiency, Usability, User Conformity and Integrity are considered to be applied. The identification of these characteristics had some reasons. For instance, Maintainability is the ability of the system to reflect the changes with a degree of ease. The occurrence of the changes can impact to the components,

services, features and interfaces once the responses to fix the errors is done and involved the changeability of the application functions and time consuming [23] and [24]. Maintainability is the main characteristic that needs to be highlighted in improving the systems function in increasing the availability and reduce the effects of run time detection. In the context of sustainability, the sub-characteristic for Maintainability is broken down into Evolvability, Analyzability, Changeability, and Testability [23] and [24].

The second proposed characteristic is Functionality in which the ability of the system to do the task that has been intended. This characteristic is related to the task of the system which is the element involved in the system is working in a coordinated manner in completing the job cooperatively [23] and [24]. The element involved in the system will be assigned into the correct responsibilities for coordinating with other element in the system [24]. In the context of sustainability, the Functionality characteristic is needed to decompose into some modules in order to make it understandable and to support the variety of the purposes in the system. However, Functionality is independently of structure which is the modules can be interacted each other in reflecting to the errors, constraints, and other qualities in the system function. Therefore, this characteristic is divided into sub-characteristic such as Suitability, Accuracy, and Interoperability [7], [23] and [24].

Next, Portability is selected characteristic in the ranking towards software sustainability. Portability is the ability of a system to run under different computing environment for example, one application can be used and understandable in different language of the source code in executing and different platforms or protocol [7], [14], [23] and [24]. Usually, the environment types refer to the combination of hardware and software of the system [24]. Sub-characteristic for Portability such are Installability, Reusability, Adaptability, and Replacebility. In order to be a sustainable manner in software sustainability, this characteristic is very important to be highlighted due to the level of monitoring, coordinating and communication for the standards and procedures in handling the asset for example, in disaster management such as land and acquisition, re-housing, compensation development for flood victim and all the benefits involved in the management. In sustainability dimension such as in environmental, this characteristic is beneficial in evaluating the pilot program in order to protect and treats the diseases, accommodation, consumption, fairness or equity, and consensus [65], [66] and [69].

Next, Efficiency is the ability of the system functions in time required to complete any task given to the system. It is related to the utilization of processor capacity, disk space and memory efficiency that can be used in real time applications [24]. According to [12] and [71] suggested that Efficiency is a very important characteristic to be highlighted in order to achieve sustainability. The researchers are considered that the level achievement of Efficiency characteristic of software system is the higher performance in achieving sustainability of software systems. This is because the Efficiency characteristic enables the completeness of systems function in the required task rapidly that is related to the time behaviour of the system, the consumption of hardware and software of the system, and level of speed of the system running overtime and give the best result accurate to the users. This makes the sub-characteristics are divided into Time Behaviour, Resource Utilization, and Efficiency Compliance [12], [14], [71], [72] and [73]. In order to develop a sustainable software system, the performance of Efficiency characteristic is required to be highlighted in term of the interaction between data access performance and data management by assessing the memory, network and disk space management, coding practices, compliance with programming language and the databases. These criteria can be affected by the software system performance that is the potential operational performance bottlenecks and future scalability problems in the software system that is required for higher execution speed in dimensionality of data [12], [33], [34], [47], [48], [54], [57] and [71].

Furthermore, Efficiency characteristic will support to develop a software system in all dimensions of sustainability, i.e. (environment, economic, and social). In the environmental aspect, the evaluation will be focused on assessing the ecosystem health, public health, and environment protection. For example, the consumption of hardware for developing software maintenance should be not affected by the environment such as applying the toner coverage, ink, frame, and etc that can be recycled and reused in order to protect the environment affected by zoning ordinances caused by carbon dioxides. In economic aspect, the application of the utilization that can be recycled and reused can save the risks of management that can affected to the expenditure to protect the environment and also might be related to the

existence of any diseases to the country. Thus, the phenomenon can affect to the social aspect which influences the society, organization, and also the country risks and health [65], [66] and [69].

Usability characteristic is categorized into the user qualities experience which has involved the application of interfaces in the software system. The designation of the interfaces in the software system is focused on the user and consumer requirements and needs on the application of interfaces [23], [24], [72], and [73]. Thus, the Usability characteristic is most important to provide access for disabled users, provide a good overall user experience through the application in which it can be localized and globalized application [23] and [24]. In order to achieve sustainability in the software system, this characteristic can support to utilize the interaction between the interfaces is needed in order to control an excessive number of clicks in a required for a single task. This function is extremely important to produce the software system with the understandability function which is considering the incorporating workflows where appropriate to simplify multistep operation. Through this function, the software system can be learned any other changes occurred and can feedback to the user, especially for errors and exception [45] and [67]. Thus, the implementing technologies and techniques are also contributed to sustainability in software system in which involves the user interactivity such as Asynchronous JavaScript and External Mark up Languange (AJAX) that can be used the web pages and client-side input [45]. This is the way to improve the software development towards a long living system by assessing the level of usability of software system in strengthening the regional project planning capabilities. Consequently, the sub-characteristic is divided into Understandability, Learnability, and Operability.

User Conformity is related to the impact attribute such as user perception and user requirement. Based on the Pragmatic Quality Factor (PQF) model created by [72] and [73] believes that the attribute is extremely important to balance the quality model between technical measurement of software and human factor [14]. The characteristic refers to the impact characteristic, which is divided into two categories: user perceptions and user requirements. In particular, user perception measures the popularity, performance, trustworthiness, law and regulation, recommendation, environment, and adaptability, while user requirement measures the user acceptance and satisfaction. In order to achieve sustainability, User Conformity decomposes into User Perception and User Requirement as their sub-characteristic as to measure the level of sustainability in environment and social aspect [3], [14], [72] and [73]. This is because in the environment aspect involves the consensus feature which is the stakeholders will involve and acts as effected parties in handling the software systems and relates to the decision making. While in social aspect, this characteristic involves in changing the perception of people about the effect and caused by the software system, for instance in disposing the software equipment and encouraging people in building awareness [19], [45], [67] and [70].

The last but not least characteristic in this study is Integrity. Integrity is the ability to detect and manage invalid data coming into the system as well as the imposition of complete transaction or rollbacks [23] and [24]. In another meaning, Integrity is the ability of the system to protect the software system from unauthorized access to system functions, preventing information loss, ensure that the software is protected from virus infection, and also the privacy of data entered protection into the system [23], [24], [14], [72] and [73]. Integrity characteristic is focused to be achieved sustainability in software system due to their best practices in assessing security in the software engineering [3], [7], [14], [72], and [73]. It involves the measuring on application architecture that has been used in the system such as Software-as-a-Service (SaaS), Enterprise knowledge sharing (know-how), Enterprise Application Integration (EAI), and the latest one is Service Oriented Architecture (SOA) for developing the multi layer compliance design, the programming language applied, error detection and exception handling [1], [2], [3], and [19].

Table 9 below summarizes the proposed characteristic and sub-characteristic of software development towards long living software presented in this study. This is followed by Figure 1 summarizes the component of characteristic and sub-characteristic of software development towards long living software.

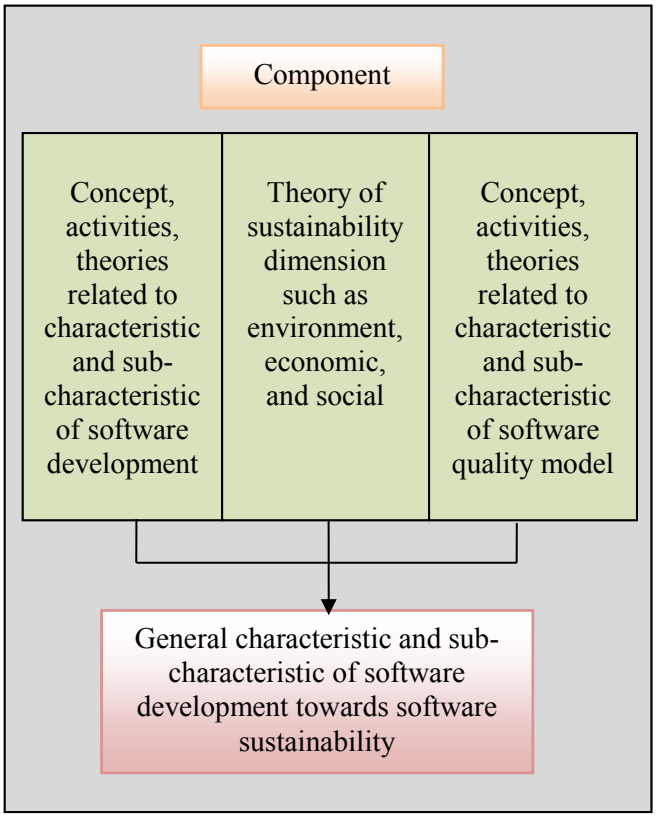
4.2 Organization of the characteristic into sustainability dimension

Based on the investigation of a systematic review in [1] and [2], there is still required a lot of research studies in proposing the characteristic and sub-characteristic of software development towards a long living system that organized in three dimensions of sustainability, i.e. (environment, ec-

Table 9 Proposed characteristic and sub-characteristic towards long living software

Characteristic	Sub-characteristic
Maintainability	Evolvability Analysability Changeability Testability
Functionality	Suitability Accuracy Interoperability
Portability	Installability Reusability Adaptability Replacebility
Efficiency	Time Behavior Resource Utilization
Usability	Understandability Learnability Operability
User Conformity	User perception User Requirement
Integrity	Security Data Protection

Fig.1 Component of characteristic and sub-characteristic of software development towards long living software



onomic and social dimension).

In this study, the proposed characteristic and sub-characteristic are structured and organizing into sustainability dimensions which are focused on the flooding management study as the domain of the research. As the results, the components of the proposed characteristics in this study are structured and organizing based on the priority of characteristic in handling the requirements of software development in flood management towards software sustainability. The organization of characteristic and sub-characteristic of software sustainability in this study is followed the theory and standard organization recommended by the International and Organizational Standard (ISO/IEC 9126) for characteristic and sub-characteristic in software quality. Table 10 summarizes the software sustainability characteristics that are structured and organized in three dimensions of sustainability in the domain of flood management. The description of the features structured is further explained in the following sections.

Table 10 Proposed characteristic organized into sustainability dimension

Sustainability dimension	Proposed characteristic
Environment	Efficiency Maintainability Functionality Portability User Conformity
Economic	Maintainability Usability Integrity
Social	User Conformity Efficiency Maintainability Functionality

4.3 Description of the organization characteristic into sustainability dimension

This section explains the reason of organizing several characteristics into environmental, economic and social dimension.

Description of environmental characteristic

1) Efficiency – This feature can assess the ecosystem health, public health and environmental protection such as flood awareness and flood warning. Sustaining the

structural measures and non-structural measures, for example structural measures refers to the large scale defences such as dikes, dams and flood control reservoirs, diversions, flood ways, improving channel conveyance capacity (widening, deepening, realignment, bank protection and etc. Example of non-structural measures i.e. (zoning ordinances, building codes, flood proofing, early detection and warning, emergency planning, education and flood insurance among others) [65], [66] and [69]. In fact, the feature can support to be used the equipments of software system that can be disposed without leaving the impacts to the environmental and societal.

2) Maintainability – Refers to the feedback in the element of sustainability of the biosphere. It has contributed to the success of a democratic system of government, the effectiveness of market based economic systems and the power of science [65], [66], [69] and [71]. A key feature that has made these 3 institutions effective the capacities for systematic feedback [23] and [24]. Towards sustainability, the needs to be addressed such weaknesses in our feedback systems if we are achieving sustainability in over the long run. The feature can support the durability of the equipments applied in building a software system by enhancing the corrective maintenance by using the recyclable of equipments.
3) Functionality – This feature can support to create the multi-purpose for the software system with a number of functions related to sustainability, such as flood protection, water supply, waterpower, navigation, and etc [65], [66], [69] and [71].
4) Portability– The system can support to monitoring, coordinating and communication for the standards and procedures in reallocating the land acquisition, re-housing and compensation development for the flood's victim benefit. For health and well-being, the system features can support to evaluate the pilot program in order to protect and treats the diseases like bilharzias, malaria and etc [65], [66], [69] and [71].
5) User Conformity – The system can make consensus means the stakeholders involved and affected parties should agree as to the program of flood protection and management. This criterion is related to the decision making [3], [72] and [73]

Description of economic characteristic

1) Maintainability – The feature can analyze the cost benefit technique which can be influenced to the cost effectiveness, cost efficiency and the return of investment (ROI) to the organization. The identification of the faulty item is necessary to enhance the fault detection in order to reduce the cost of maintenance [7]. The criteria will influence to the economic stability and well being of the peoples and communities in the country by increasing the employment opportunities, maintenance of software property prices that can bring to the improvement of the quality of software system because the lower level of identification of repair equipment is better than the higher level of repair equipment involved in upgrading the software system.
2) Usability – This feature can support to strengthen the regional project planning capabilities and built learning mechanisms that can help the user in handling the software system [19], [45] and [67].
3) Integrity – The software system can support features for data security and protection, and also supports in decision making as to decrease the cost of maintenance for software systems. This is much supported to achieve software sustainability with developing software system with trustfulness [8], [9], [10] and [14].

Description of social characteristic

1) User Conformity – The system features to support social acceptability of different flood damage control measures for different community in order to design an acceptable combination of measures. The changing perception of people about the effect and caused by flood and build awareness [1], [2], [3], and [19].
2) Maintainability – The software system is capable to provide user satisfaction to the stakeholders [14], [72], and [73]. This feature can support to maintain the users hours in maintaining the software system per year, month, day, hour, and maintenance action in order to upgrade the user satisfaction towards long living software.
3) Efficiency – The system features can establish the energy behaviour to support the

energy baseline in the software system [23] and [24]. Towards software sustainability, the level of carbon emission and carbon footprint in the CPU or processor equipments will be not affected to the social and environmental impacts. In fact, the level of energy that has been installed in the CPU or processor equipments can reduce the energy changes per hour when the system is not in use or function [20] and [21]. This feature can help in utilizing the effectiveness of energy behaviour and energy consumption.

4) Functionality – This feature is very important to support the software functionalities such as connection includes technologies that enable to connect between people in social networking [5], [8], [9] and [19]. Furthermore, the collaboration includes software that enables people to work each other on a shared problem or tasks are also discovered [23] and [24]. In addition, the communication includes software that allows people to communicate with each other either in synchronously or asynchronously with text, audio or video are also supported. Consequently, the suitability of the software system to support business in global collaborative processes will upgrades the decision making towards long living software.

5 Discussion

This section provides a discussion of the results pertaining to the discussion on the state-of-the-art, and discussion for a body of knowledge.

5.1 Discussion on a state-of-the-art

As analyzed from the included articles, less than expected on the state-of-the-art for sustainability in software engineering, especially in software development can be referred as to support the body of knowledge and achieve the objective stated in this study. As the result, the need to extend the inclusion of publications that is classified as a research with further investigating on sustainability in software engineering are very significant. Furthermore, there is still required a lot of research studies in proposing the characteristic and sub-characteristic of software development into three dimensions of sustainability such environment, economic and social dimension. This research will be beneficial in promoting the software sustainability features by highlighting the important characteristic and sub-characteristic of software development, and also the assessment criteria through the features that are needed to be underlined in order to achieve the goal of software sustainability.

In addition, the software sustainability features proposed in this study are very helpful to the software developer, software assessor, and researcher in making the decision to develop a software system in a sustainable manner. The proposed model is very advantageous to the stakeholders in providing the systematic review with efficiency, reliability and practicality of the sustainable way through the well-defined processes, activities, criteria, and the assessment criteria towards software sustainability. In addition, the result performed in the proposed model such as the achievement of software sustainability in the aspects of environment, economic, and social dimension is very useful and supportive for the stakeholders in upgrading and improving the way to develop a software system towards long living systems.

5.2 Discussion for a body of knowledge

The findings support a body of knowledge in this study is led by the issue of sustainability in software engineering with representation in the sustainability dimension, perspective, concept, and the characteristics and sub-characteristics that are structured and organized in three dimensions of sustainability. Figure 2 illustrates the body of knowledge areas that much supporting in generating the characteristic and sub-characteristic structured and organized into three dimensions of sustainability.

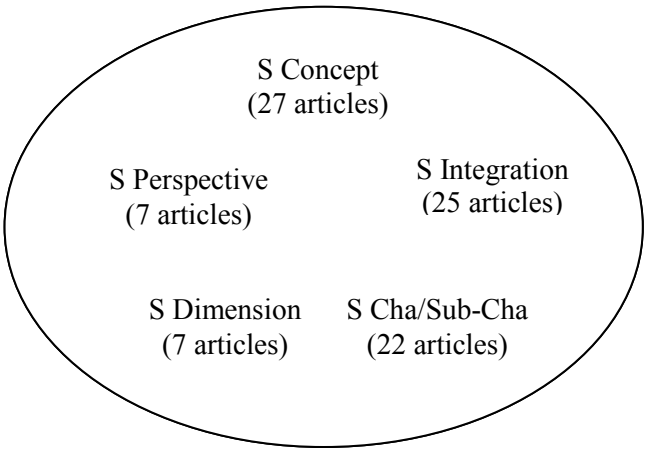


Fig. 2 Areas of the body of knowledge for sustainability in software development towards long living software

The proposed model presented in this study is very beneficial to software sustainability community because the issue related to the three dimensions of sustainability are directly highlighted with clearly defined in term of the indicators towards software sustainability. The important characteristics and sub-characteristics provided in this study are very helpful in encouraging the stakeholders to improve, extend, and adapt the suggested features of software sustainability into the software development. Next, the contents of this study provides more historical of sustainability from initial discussed in other disciplines to be exclusively discussed in the software engineering field. In addition, the assessment criteria provided through the proposed features are potentially contributing to the novelty of measurement criteria by evaluating the three dimensions of sustainability individually with mapping the value for action and improvement in future.

6 Conclusion

This paper has presented the extended results of systematic literature reviews on research activity of sustainability in software engineering [1] and [2]. The related topics of the included articles allow for building a body of knowledge as to achieve the objective in this study. Almost 3,645 articles are related to sustainability in software engineering and considered 225 articles are related and relevant to support the understanding and holistic views of sustainability in software engineering that relates to the area of this study. The articles have been revised and reviewed through the classification of the articles into the content, topic, application domain, and potential benefit for further investigation. Consequently, 88 articles are selected as the most relevant articles in supporting the body of knowledge for sustainability in software development. The critical review is stressed on the component of characteristic and sub-characteristic by structuring them into the dimensions of sustainability, such as environment, economic, and social with the description of each features towards software sustainability. Furthermore, the supporting knowledge based on the sustainability concept, perspective, integration environment, and dimension are also much helps in building the strong understanding on the issues of study.

Based on the contributions of previous works, none of them are integrating the three pillars dimension of sustainability by highlighting each dimension as the priority features in identifying the characteristic and sub-characteristic towards software sustainability. The identification of the characteristics and sub-characteristics in software sustainability are un-certainty because they are only based on theoretical approaches for developing a software product without fully investigating on the impacts of each characteristic and sub-characteristic towards sustainability dimensions. Consequently, they are failed to show the holistic views of characteristics and sub-characteristics in three dimensions individually as general indicators towards software sustainability.

Future work is to extend this study in developing a metric evaluation of software sustainability by introducing a metric for each proposed characteristic in evaluating each dimension of sustainability individually. Hopefully, this study can support to further investigation by clarifying the important features of software development that can be applied as a goal in the metric evaluation towards software sustainability achievement.

ACKNOWLEDGEMENT

We would like to thank the Malaysian Ministry of Higher Education for providing the funding for this research through LRGS entitled “A new Holistic Service Oriented model for wireless and sustainable flood management: From monitoring to emergency response”.

References:

[1] Ahmad, R., Baharom, F., & Hussain, A. (2015). A Systematic Review on Characteristic and Sub-Characteristic for Software Development towards Software Sustainability. *World Scientific and Engineering Academy and Society (WSEAS) 2015, 23 –25 April 2015, Kuala Lumpur, Malaysia http://www.wseas.org/cms.action*

[2] Ahmad, R., Baharom, F., & Hussain, A. (2014). A Systematic Literature Review on

Sustainability Studies in Software Engineering. *Knowledge Management International Conference (KMICe) 2014, 12–15 August 2014, Malaysia* http://www.kmice.cms.net.my/

[3] Ahmad, R., Yahaya, J.H., Deraman, A., Kamaruddin, S.S.,. (2011). Intelligent Software Quality Model: The Theoretical Framework. *The Proceedings of the 3rd International Conference on Computing and Informatics, ICOCI 2011, 8-9 June 2011, Bandung Indonesia, pp. 160-166.*

[4] Aier, S., & Schelp, M. (2006). Evaluating Integration Architectures – A Scenario-Based Evaluation of Integration Technologies. *TEAA 2005, LNCS 3888*, 2–14.

[5] Amsel, N., Ibrahim, Z., Malik, A., & Tomlinson, B. (2011). Toward Sustainable Software Engineering (NIER Track). *ICSE' 11*, *1*(978), 976–979.

[6] Araujo, V., & Vázquez, J. A. (2013). Business and Technical Requirements of Software-As-a-Service: Implications in Portuguese Enterprise Business Context. *International Journal in …*, *3*(6), 1–14. doi:10.5121/ijfcst.2013.3601

[7] Azuma, M. (1991). SQuaRE the next generation of the ISO/IEC 9126 and 14598 international standards series on software product quality. *Technical Report, ISO/IEC* JTC1/SC7/WG6.

[8] Bevan, N. (1997). Quality and usability: A new framework. *National Physical Laboratory*, United Kingdom.

[9] Bevan, N. (1999). Quality in use: Meeting user needs for quality. *The Journal of Systems and Software, 49, March 19*. 89-96.

[10] Bhatti, S. N. (2005). Why Quality? ISO 9126 Software Quality Metrics (Functionality) Support by UML Suite. *Advances in Engineering Software*, *30*(2), 1–5.

[11] Cabot, J., Easterbrook, S., Horkoff, J., & Maz, J. (2009). Integrating Sustainability in Decision-Making Processes_a Modelling Strategy. *Journal of Green Engineering*, 1–4.

[12] Calero, C., & Bertoa, M. F. (2013). Sustainability and Quality : icing on the cake. *Journal of Green Engineering*.

[13] Denning, P. J. (1992). What is Software Quality? *A Commentary from Communications of ACM* (January).

[14] Deraman, A. & Yahaya, J. H. (2010). Measuring the unmeasurable characteristics of software quality using pragmatic quality factor. Proceedings from: *2010 3rd IEEE International Conference on Computer Science and Information Technology, July 7-10, 2010, Chengdu, China,* ISBN:978-1-4244-5539-3, 197-202.

[15] Dick, M., & Naumann, S. (2010). Enhancing Software Engineering Processes towards Sustainable Software Product Design. *Integration of Environment Information in Europe*, 706–715.

[16] Dikbas, a, Ercoskun, P., & Ercoskun, K. (2010). Enabling Sustainability through SOA within the AEC / FM Domain, 1–7.

[17] Dromey, R. G., & Popper, K. (1994). A model for software product quality. *Software Quality Institute*, (October), 1–35.

[18] Durdik, Z., Klatt, B., Koziolek, H., Krogmann, K., Stammel, J., & Weiss, R. (2012). Sustainability guidelines for long-living software systems. *2012 28th IEEE International Conference on Software Maintenance (ICSM)*, 517–526. doi:10.1109/ICSM.2012.6405316

[19] Dustdar, S., Dorn, C., Li, F., Baresi, L., Cabri, G., Pautasso, C., & Zambonelli, F. (2010). A roadmap towards sustainable self-aware service systems. *Proceedings of the 2010 ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems - SEAMS '10*, 10–19. doi:10.1145/1808984.1808986

[20] Finkbeiner, M., Inaba, A., Tan, R., Christiansen, K., & Klüppel, H.-J. (2006). The New International Standards for Life Cycle Assessment: ISO 14040 and ISO 14044. *The International Journal of Life Cycle Assessment*, *11*(2), 80–85. doi:10.1065/lca2006.02.002

[21] Governor, J. (2009). SOA : An On Ramp To Sustainability. *Redmonk Greenpaper*, (March), 1–9.

[22] Hilty, L. M., & Hercheui, M. D. (2010). ICT and Sustainable Development 1 The Ethics of Sustainable Development and the Role of ICT. *IFIP Advances in Information and Communication Technology*, *328*, 227–235.

[23] IEEE. (1993). IEEE standard for a software quality metrics methodology. Retrieved August 20, 2010, from http://ieeexplore.ieee.org/xpl/standards.jsp.

[24] ISO/IEC 9126. (1996). Software quality characteristics and metrics-Part2: External metrics. *Technical Report, ISO/IEC* JTC1/SC7/WG6.

[25] Jackson, M., Crouch, S., & Baxter, R. (2011). Software evaluation: criteria-based assessment. *Software Sustainability Institute*, 1–13. Retrieved from http://software.ac.uk/sites/default/files/SSI-SoftwareEvaluationCriteria.pdf

[26] Johann, T. (2013). Position Paper: The Social Dimension of Sustainability in Requirements Engineering University of Hamburg University of Hamburg. *Journal of Systems and Software*.

[27] Johann, T., Dick, M., Kern, E., & Naumann, S. (2011). Sustainable Development , Sustainable Software , and Sustainable Software Engineering. *International Symposium on Humanities, Science and Engineering Research*.

[28] Jorgensen, M. (1999). Software quality measurement. *Advances in Engineering Software*, *30*(12), 907–912.

[29] Khamis, N., Misfian, A. M., & Md Noor, R. (2013). Towards sustainable software criteria: Rescue operation and disaster management system model. *2013 10th IEEE INTERNATIONAL CONFERENCE ON NETWORKING, SENSING AND CONTROL (ICNSC)*, 398–403. doi:10.1109/ICNSC.2013.654877.

[30] Kilidar, H., Cox, K., & Kitchenham, B. (2005). The use and usefulness of the ISO/IEC 9126 quality standard. *2005 International Symposium on Empirical Software Engineering, 2005.*, 122–128.

[31] Kitchenham, B. (2007). *Guidelines for performing Systematic Literature Reviews in Software Engineering* (pp. 1–53).

[32] Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering – A systematic literature review. *Information and Software Technology*, *51*(1), 7–15.

[33] Kocak, S. A. (2011). Green Software Development and Design for Environmental Sustainability. *Journal of Green Engineering*.

[34] S.N.E.S. Mohd Shokri And W.H. W.Badaruzzaman (2014). A Problem Based Sustainability Education Approach in Integrated Design Project Course. *International Conference on Education and Educational (EET '14)*. WSEAS Press.

[35] Koziolek, H., Domis, D., Goldschmidt, T., & Vorst, P. (2013). Measuring Architecture Sustainability. *IEEE Software*, *30*(6), 54–62. doi:10.1109/MS.2013.101

[36] Koziolek, H., Schlich, B., Becker, S., & Hauck, M. (2012). *Performance and reliability prediction for evolving service-oriented software systems*. *Empirical Software Engineering* (pp. 1–29). doi:10.1007/s10664-012-9213-0

[37] Koziolek, H., Weiss, R., Durdik, Z., Stammel, J., & Krogmann, K. (2011). Towards Software Sustainability Guidelines for Long-living Industrial Systems. *3rd Workshop of GI Working Group'Longliving Software Systems L2S2 Design for Future*.

[38] Kundzewicz, Z. W. (2009). Flood protection - Sustainability Issues. *Journal of Sciences Hydrologiques*, *44 (4)*(August 1999), 559–572.

[39] Laboratorio, M. O. (2002). A Systemic Quality Model for Evaluating Software Products. Retrieved Sept10, 2011, from http://www.lisi.usb.ve/publicaciones/02 calidad sistemica/calidad_24.pdf

[40] Lin, K. J., & Panahi, M. (2010). A real-time service-oriented framework to support sustainable cyber-physical systems. *IEEE International Conference on Industrial Informatics (INDIN)*, 15–21. doi:10.1109/INDIN.2010.5549473

[41] Mahaux, M., & Canon, C. (2013). Integrating the Complexity of Sustainability in Requirements Engineering 4. Requirements Engineering and impacts on the software life-. *Journal of Systems and Software*.

[42] Mahaux, M., Heymans, P., & Saval, G. (2011). Discovering Sustainability Requirements : An Experience Report. *WC Requirements Engineering: Foundation for Software Quality*.

[43] Malekzadeh, B. (2010). *Event-Driven Architecture and SOA in collaboration*.

[44] Mori, K., & Christodoulou, A. (2012). Review of sustainability indices and indicators: Towards a new City Sustainability Index (CSI). *Environmental Impact Assessment Review*, *32*, 94–106. doi:10.1016/j.eiar.2011.06.001

[45] Parris, T. M., & Kates, R. W. (2003). Characterizing and measuring sustainable development. *Annual Review of Environment and Resources*, *28*, 559–586. doi:10.1146/annurev.energy.28.050302.10555

[46] Penzenstadler, B. (2012). What does Sustainability mean in and for Software Engineering ? *International Journal of Software Engineering & Application (IJSEA)*.

[47] Penzenstadler, B. (2013). RE @ 21 : Time to Sustain! *Journal of Green Engineering*.

[48] Penzenstadler, B. (2013). Supporting Sustainability Aspects in Software Engineering. *3rd International Conference on Computational Sustainability*, 1–4.

[49] Penzenstadler, B., & Femmer, H. (2013). A generic model for sustainability with process- and product-specific instances. *Proceedings of the 2013 Workshop on Green In/by Software Engineering - GIBSE '13*, 3–7. doi:10.1145/2451605.2451609

[50] Penzenstadler, B., Bauer, V., & Fleischmann, A. (2013). Seminar : Sustainability in Software Engineering.

[51] Penzenstadler, B., Bauer, V., Calero, C., & Franch, X. (2012). Sustainability in software engineering: a systematic literature review. *16th International Conference on Evaluation & Assessment in Software Engineering (EASE 2012)*, 32–41. doi:10.1049/ic.2012.0004

[52] Penzenstadler, B., Femmer, H., & Richardson, D. (2013). Who is the advocate? Stakeholders for sustainability. *2013 2nd International Workshop on Green and Sustainable Software (GREENS)*, 70–77. doi:10.1109/GREENS.2013.6606424

[53] Penzenstadler, B., Fleischmann, A., & Linde-akademie, C. Von. (2011). Teach Sustainability in Software Engineering ? *24th IEEE-CS Conference on Software Engineering Education and Training, CSEE and T 2011*, 1–5.

[54] Penzenstadler, B., Irvine, U. C., Richardson, D., Franch, X., & Raturi, A. (2014). *Institute for Software Research Protocol and Results - Protocol and Results* (pp. 1–19).

[55] Penzenstadler, B., Khurum, M., & Petersen, K. (2013). Towards Incorporating Sustainability while Taking Software Product Management Decisions. *International Workshop on Software Product Management (IWSPM 2013)*.

[56] Penzenstadler, B., Mahaux, M., & Salinesi, C. (2013). RE4SuSy : Requirements Engineering for Sustainable Systems. *18th International Working Conference on Requirements Engineering: Foundation for Software Quality, (REFSQ)*, 8–12.

[57] Penzenstadler, B., Raturi, A., & Richardson, D. (2014). Safety, Security, Now Sustainability: The Nonfunctional Requirement for the 21st Century. *IEEE Software*, *31*, 40–47. doi:10.1109/MS.2014.22

[58] Penzenstadler, B., Tomlinson, B., & Richardson, D. (2013). RE4ES : Support Environmental Sustainability by Requirements Engineering. *18th International Working Conference on Requirements Engineering: Foundation for Software Quality, (REFSQ)*.

[59] Penzenstadler, B.,Norton, J., Stringfellow, A., Laviola, J. J., & Tomlinson, B. (2013). Plant Guild Composer : A Software System for Sustainability. *IEEE Computer Society*, 1–8.

[60] Qutaish, R. E. (2009). Measuring the software product quality during the software development life-cycle: An international organization for standardization standard perspective. *Journal of Computer Science, 5 (5),* 392-397.

[61] Rachuri, S., Sarkar, P., & Sriram, R. (2009). Metrics, Standards, and Industry Best Practices for Sustainable Manufacturing Systems (extended Abstract). *5th IEEE Conference on Automation Science and Engineering*, 472–477.

[62] Sahu, S. (2010). *Design and Implementation of a Heterogeneous Sensor-based Embedded System for Flood Management* (pp. 1–49).

[63] Shenoy, S., & Eeratta, R. (2011). Green Software Development Model. *India Conference (INDICON), 2011 Annual IEEE*, 1–6.

[64] Silcher, S., Minguez, J., Scheibler, T., & Mitschang, B. (2010). A service-based approach for next-generation Product Lifecycle Management. *2010 IEEE International Conference on Information Reuse and Integration, IRI 2010*, 219–224. doi:10.1109/IRI.2010.5558936

[65] Simonovic, S. P. (1998). *Decision Support System for Flood Management in the* (pp. 1–41).

[66] Simonovic, S. P. (2001). Two New Non-Structural Measures for Sustainable Management of Floods. *International Workshop on Non-Structural Measures for Sustainable Management of Floods*, *27*(1), 65–81.

[67] Solow, R. . (1991). Sustainability Ecomist perspective.pdf. *Economics of the Environment Selected Readings*, 1001–1010.

[68] Suryn, W., Abran, A. & April, A. (2003). ISO/IEC SQuaRE: The second generation of standards for software product quality. Retrieved December 20, 2010, from

http://www.lrgl.uqam.ca/publications/pdf/799.pdf.

[69] Takeuchi, K. (2001). Flood Management in Japan-From Rivers to Basin. *International Workshop on Non-Structural Measures for Water Management Problems*, *27*(1), 37–44.

[70] United Nations World Commission on Environment and Development (1987). Report of the World Commission on Environment and Development: Our Common Future In United Nations Conference on Environment and Development, 1987.

[71] M. Mohankumar. (2015). Empirical Study on Green and Sustainable Software Engineering. *Advances in Software Engineering and Systems*. WSEAS Press.

[72] Yahaya, J. H, Deraman, A. & Hamdan, A. R. (2008). Software quality from behavioural and human perspectives. *IJCSNS International Journal of Computer Science and Network Security, 8(8), August 30*, 53-63.

[73] Yahaya, J. H., Deraman, A. & Hamdan, A. R. (2010). Continuously ensuring quality through software certification: A case study. *The International Conference on Information Society (i-Society 2010), June 28-30, London, UK.*

[74] Zahran, S., Brody, S. D., Peacock, W. G., Vedlitz, A., & Grover, H. (2008). Social vulnerability and the natural and built environment : a model of flood casualties in Texas. *Journal Compilation @ Overseas Development Institute*. doi:10.1111/j.0361-3666.2008.01054.

[75] Zamagni, A. (2012). Life cycle sustainability assessment. *The International Journal of Life Cycle Assessment*, *17*, 373–376. doi:10.1007/s11367-012-0389-8

[76] Zhang, J., Zhou, C., Xu, K., & Watanabe, M. (2002). Flood disaster monitoring and evaluation in China. *Global Environmental Change Part B: Environmental Hazards*, *4*(2-3), 33–43.

[77] Zhang, Q. (2001). Quality dimensions, perspectives and practices: A mapping analysis. *International Journal of Quality & Reliability Management*, *18*(7), 708–722.