Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

Article

Proposal for a Software Sustainability Design Catalogue

Shola Oyedeji [1,*], Birgit Penzenstadler [2] and Ahmed Seffah[3]

[1] Department of Software Engineering, Lappeenranta University of Technology, Lappeenranta, Finland; shola.oyedeji@lut.fi
[2] Department of Computer Engineering and Computer Science California State University Long Beach Long Beach, California, USA; birgit.penzenstadler@csulb.edu
[3] Department of Software Engineering, Lappeenranta University of Technology, Lappeenranta, Finland; ahmed.seffah@lut.fi
* Correspondence: shola.oyedeji@lut.fi; Tel.: +3584-1369-7708

Abstract: Like other ICT communities, sustainability in software engineering is a major research and development concerns. Current research focusses on eliciting the meanings of sustainability and proposing approaches for its engineering and integration into the mainstream software development lifecycle. However, few concrete guidelines that software designers can apply effectively are available and applicable. Such guidelines are needed for the elicitation of sustainability requirements and testing software against these guidelines. This paper introduces a sustainability design catalogue to assist software developers and managers in eliciting sustainability requirements, and then in measuring and testing software sustainability. The paper reviews the current research on sustainability in software engineering which is the grounds for the development of the catalogue. Four different case studies were analyzed using the Karlskrona manifesto on sustainability design. The output from this research paper is a software sustainability design catalogue through which a pilot framework is proposed that includes a set of sustainability goals, concepts and methods. The integration of sustainability for/in software systems requires a concrete framework that exemplifies how to apply and quantify sustainability. The paper demonstrates how the proposed software sustainability design catalogue provides a step towards this direction through a series of guidelines.

Keywords: sustainability, software sustainability, information and communication technology, software design, sustainability requirement, software sustainability analysis, software sustainability guidelines, karlskrona manifesto

---

1. Introduction

Software sustainability and sustainable software development are now recognized as an important concerns not only for researchers, but also for the entire software industry and standardization bodies such as ISO and IEEE. An IBM global CEO study shows that 47% of organizations have begun to redesign their entire software business models to incorporate sustainability [1]. Sustainable development is also driving software innovations for creating new opportunities of cutting costs, adding values and for gaining competitive advantages [2]. As software is the catalyst of economic and social activities [3] and the pillar for all industries, there is a huge pressure on the software industry from regulators and the civil society to consider green practices while developing software to support the green economy and green software that uses less energy [4].

The Ericsson sustainability report shows that ICT could help reduce global greenhouse Gas (GHG) emissions by up to 15%, and forecasts that by 2021, 28 billion devices will be connected with each other [5] which increase energy consumption. Another energy and carbon report from Ericsson

Peer-reviewed version available at *Sustainability* **2018**, *10*, 2296; doi:10.3390/su10072296

forecasts is that in 2018, 90% of the world's population will have mobile coverage, and 60% will have the ability to access high-speed LTE data networks. Such reports are clear indicator of the huge sustainability impact of ICT. Overall, the ICT sector contributes around 2% of the global CO2 emissions. It is also accountable for approximately 8% of the European Union's (EU) electricity consumption and 2% of the carbon emissions from ICT devices and services [2]. It is therefore important to look at avenues of how to reduce the impact of ICT on the environment and how sustainability can be incorporated better into the software development lifecycle.

However, the current software development practices provide non-sufficient support for sustainability. Some of the largely used processes such as agile totally lack to address sustainability [6]. Practitioners have not yet been able to figure out where and how sustainability can be integrated efficiently and effectively. Where the sustainability ingredients should be considered? Indeed, the different sustainability dimensions have no reference framework that can assist software and application developers and designers. Researchers also highlighted the vital need to define measures of sustainability and search for avenues for their integration in the wider engineering processes [2] [7] .

Software design is a key milestone where supporters and pioneers largely recognized the importance of sustainability. Varying perspectives have been discussed such as the design of sustainability and sustainability by design. [8]. Sustainability of software system design is crucial factor to achieve sustainability because design is a crucial and an integrating activity for software engineering [9]. Robillard presents the importance of integrating the concept of sustainability for software design into existing catalogue of design quality attributes [10]. Our research focusses on sustainability perceptions in design and design practices while proposing a method that support quantifying sustainability, its integration in the design loop. In this paper, we propose a pilot framework based on "Software Sustainability Design Catalogue (SSDC)". The SSDC is covers set of recommendations for sustainability by design. It is grounded in Karlskrona Manifesto for Sustainability Design (KMSD). The Karlskrona manifesto is a focal point with a common ground of principles and values for sustainability design in and for software systems [11- 12].

The concepts of sustainable and green are often used interchangeably, in many communities including software engineering. This article consider that "green software" and being "sustainable software" is not the same concern. Green is usually defined as "products, systems and services that has limited impact on human health and environment. As it will be defined in this paper, sustainability include green and it goes beyond green. It is represented by a five-pillars for environmental, social and financial as well as human and technical responsibility. For example with green, the sustainability consideration about people and social is limited to direct exposures from products or services. More and more green is being replaced by sustainability as a much broader term that looks at the implications of software products including hardware devices, systems and services used over a much longer period of time, and considers social and human alongside with financial, technical and environmental impacts as well. Overall all, the paper discusses the following questions:

- How does the Karlskrona Manifesto principles relate to software development life cycle phases (SDLC)
- What are the first, second and third order impacts of software sustainability as well the five dimension of sustainability?
- How can these principles be applied while ensuring that sustainability is fully achieved during design?
- How should these principles be integrated in the current software design practices for the current version of a software system and then for the diverse ones in the future?

The usage of SSDC is presented through an example to showcase how the SSDC can benefit software developers, companies, researchers using the pilot framework. Building on these studies, we will draw a roadmap for the further development and incorporation of industrial case studies into the SSDC that support developers and/or automate the use of the guidelines detailed in the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

catalogue. The SSDC will address concerns from the five sustainability dimensions: Economic, Environment, Social, Individual and Technical [13].

Section 2 covers information about background and related research of sustainability in software design and development. Section 3 presents the foundation of the Karlskrona manifesto principles and its relation to software development life cycle phases. Section 4 details the structure and components of Software Sustainability Design Catalogue (SSDC). Section 5 discusses the feasibility study used in creating the SSDC and the proposed pilot framework for sustainability of software design. Section 6 provided a practical example for the usage of the SSDC and the pilot framework. Section 7 covers discussion about the benefits of the SSDC and pilot framework. Section 8 contains conclusion with comments on future research work.

2. Sustainability in Software Design: Background and Related Research

The design of software with sustainability consideration is still an evolving area that has been discussed by different stakeholders interested in this area. There is still very few concrete sustainability guidelines for designers of software systems to apply during software development. Although design is a central phase of any software development process [9], there has been limited research work on sustainability design. Mahaux et al. [14] stated that sustainability is one of grand challenges of our civilization because of their pervasiveness, the way we design, and consequently use, software-intensive systems has a significant impact and can influence human perceptions of sustainably greatly. The most relevant related works are listed and described in the following.

Chitchyan et al. [15] explained that there is no single point of reference for researchers or practitioners where the sustainability measures are gathered and exemplified. Shedroff [16] also highlight the difficulty of the problem for software designers, stating that even with a systems approach, there are few existing tools that wrap core principles of sustainability together. Instead, designers must learn to patch together a series of disparate sustainability understandings, and frameworks in order to address the different dimensions of sustainability. Bibir [17] suggest that alternative design solutions can be elaborated based on sustainable design practices that use most efficient energy required over ICT's life cycle.

Linda Musthaler [18] also highlight the importance of software design as a key factor that can help reduce energy consumption by 30% to 90% because software provides the real energy saving that tells hardware what to do and how to function. Along the same lines, Durdik et al. [19] propose a catalogue of sustainability guidelines that incorporates all phase of system development life cycle to support project managers, software architects, and developers during system design, development, operation, and maintenance. The proposed catalogue by Durdik et al. [19] is not as detailed as the SSDC proposed in this article as it does not cover all the different sustainability dimensions (Economic, Environment, Social, Individual, and Technical), the first, second and third order of impacts of software systems and metrics/indicators to evaluate the effectiveness of guidelines in the catalogue .

Robillard [10] introduced the concept of sustainability for software design, and calls for its integration into the existing catalogue of design quality attributes. Elucidating that sustainability in software design means design decisions and its rational should clearly be reflected in the host technology in a manner that is traceable for conformance with code and resilient to fading. The author also states that there are several software design quality models that include attributes like flexibility and reusability, but there is no attribute that captures how cost-efficient are set of design decisions over time. This kind of attribute would be called sustainability.

Johann et al. [20] proposed a life cycle model that helps to develop green and sustainable software products. The paper uses a cradle to grave approach to analyse and indicate impacts of each software product life cycle phase (development, distribution, usage, deactivation and disposal). The model proposed by the authors is a good approach to improve design and integration of sustainability into software development processes with a life cycle approach. Mahaux et al. [21] contribute an experience report from a case of applying standard requirements engineering methods to analyse sustainability aspects.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1
Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

Penzenstadler provides a characterization of sustainability from a software engineering perspective [6] and describes how to support different aspects of sustainability in software development processes, software system analysis for production, and usage phases of the lifecycle [13]. In the same vein, Erdélyi [22] studies the lifecycle activities of software development with a focus on environmental protection, proposing a formula to calculate software waste to encourage the development of green software. The author highlights key activities during software design and development with key factors at each stage of software design. Erdélyi [22] states how each of this factors relates to a green aspect in software development. According to Erdélyi [22], thoroughly designed and implemented software uses energy efficiently through computational and data efficiency.

Becker et al. [12] highlight how different research domains from past to present have been trying to tackle the issue of sustainability through collaborative work via conferences and workshops. Dick et al. propose a generic model for improving the general software development process. Although the model does not currently cover sustainability benchmarks, it provides a sound basis for its future integration [23].

Mahaux et al. [21] paper reports on a software project in which sustainability requirements were treated as first class quality requirements, and as such systematically elicited, analysed and documented. This is corroborated by Oyedeji et al. [24] stating the need to characterize software sustainability as a quality factor in requirements elicitation. It also highlighted some points for contributions to a sustainability requirements research agenda. Penzenstadler [25] presents a checklist and guide approach that demonstrates how to include the objective of environmental sustainability from the very early steps of software development. Penzenstadler [25] shows how green requirements engineering may be conducted within the scope of general purpose requirements engineering and accommodate the new objective of improving the environmental sustainability of software systems.

Roher et al. [26] paper suggests the use of sustainability requirement patterns (SRPs), which will provide software engineers with guidance on how to write specific types of sustainability requirements with the aim to overcome the barriers of incorporating environmental sustainability into the requirements engineering process. Raturi et al. [27] paper focused on how to develop sustainability as a non-functional requirement (NFR) using NFR framework informed by sustainability models and how it can be used to correctly obtain and describe sustainability related requirements of the software system to be developed.

Colmant et al. [28] presented research on how to improve the software-energy efficiency on multi-core systems. Their motivations were driven by the huge impact of the ICT on the world CO_2 emissions which represents 2%. Lami et al. [29] stated that there are few studies and suggestions about 'what' aspects of sustainability to measure and 'how' to do it with regards to ICT.

Calero et al. [30] highlighted that nowadays, sustainability is a key factor that should be considered in software quality models, though there has been less research channelled towards it. Seacord et al. [31] stated that planning and management of software sustainment is impaired by a lack of consistently applied, practical measures. Without these measures, it is impossible to determine the effect of efforts to improve sustainment practices.

The above research shows the importance of sustainability in/for software systems within the ICT domain. There is still a problem of where and how to start for designers of software systems because most designers and developers are not equipped with the right guidelines and best practice on how to design and develop software systems with sustainability considerations. The following are the main conclusions from the background and related work:

- There is no single reference point where measures of software sustainability are gathered and exemplified
- Design is key to achieve software sustainability, thus the need to show how software designers can incorporate sustainability during software design to improve ICT energy usage and CO_2 emission

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

- The need for a framework or model to assist and guide developers during software design to incorporate sustainability requirements.

These conclusions are the reasons for initiating the creation of a software sustainability design catalogue (SSDC) that can be used by researchers and developers to create new frameworks, tools, guidelines and practices for software design and development. An example of such framework is the proposed pilot framework in this article (see Figure 1 and Table 6) as guide for both experienced and infant software designers during software design and development.

3. The Foundations of Sustainability by Design: The Karlskrona Manifesto

The Karlskrona Manifesto for Sustainability Design (KMSD) has its roots in the Third International Workshop on Requirements Engineering for Sustainable Systems (RE4SuSy)[32], held at RE'14 in Karlskrona, Sweden. Christoph Becker's paper [33] about the relationship between the concerns of sustainability and longevity provided one of the motives for the creation of the manifesto. The key goal was to blend the diverse aspects of sustainability to clarify its scope, objectives and challenges of the perceptions of sustainability leading to an interdisciplinary platform of researching sustainability [33].

The manifesto brings together input from researchers of various disciplines in the field of software engineering with sustainability research interests as the creators of the design manifesto [34] [11]. The Karlskrona Manifesto for Sustainability Design include nine principles of sustainability design [11]. They provide the basis for creating a reference point that can be applied during software design by different stakeholders (Table 1). The manifesto is accessible via the Web[34], where those interested in supporting the Manifesto can sign it.

Table 1. Description of the Karlskrona Manifesto Principles, Adapted from [11]

Principle Number	Principle	Description
P1	Sustainability is systemic	Sustainability is never an isolated property. It requires transdisciplinary common ground of sustainability as well as a global picture of sustainability within other properties
P2	Sustainability has multiple dimensions.	We have to include those dimensions into our analysis if we are to understand the nature of sustainability in any given situation
P3	Sustainability transcends multiple disciplines.	Working in sustainability means working with people from across many disciplines, addressing the challenges from multiple perspectives.
P4	Sustainability is a concern independent of the purpose of the system.	Sustainability has to be considered even if the primary focus of the system under design is not sustainability.
P5	Sustainability applies to both a system and its wider contexts.	There are at least two spheres to consider in system design: the sustainability of the system itself and how it affects the sustainability of the wider system of which it will be part of.
P6	System visibility is a necessary precondition and enabler for sustainability design.	Strive to make the status of the system and its context visible at different levels of abstraction and perspectives to enable participation and informed responsible choice.
P7	Sustainability requires action on multiple levels.	Seek interventions that have the most leverage on a system and consider the opportunity costs: Whenever you are taking action towards sustainability, consider whether this is the most effective way of intervening in comparison to

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

		alternative actions (leverage points).
P8	Sustainability requires to meet the needs of future generations without compromising the prosperity of the current generation	Innovation in sustainability can play out as decoupling present and future needs. By moving away from the language of conflict and the trade-off mind-set, we can identify and enact choices that benefit both present and future.
P9	Sustainability requires long-term thinking.	Multiple timescales, including longer-term indicators in assessment and decisions should be considered.

The Karlskrona manifesto principles aim to be a practical guide to the entire community like the Agile manifesto [35], the Business Rules manifesto [36], the SOA manifesto [37] [38], and the Recomputation manifesto [39]. It will support stakeholders in industry and academia (Companies, standardization organization, software practitioners, researchers and students) for promoting and developing sustainability design and practices in software development [33] [11]. The Karlskrona manifesto also serves as a facilitator for thinking about the broad effects of software on society and the need to embody longer-term thinking, ethical responsibility, and an understanding of how to integrate sustainability into the design of software systems [12].

Table 2 below shows how these Karlskrona principles can be related to software development phases [40] and at which phase a principle can be applied. Relating these principles to each software development phase will provide an avenue for better practical usage of the principles especially for evaluation of different software systems.

Table 2. Karlskrona Manifesto principles in relation to SDLC Phases

SDLC Phases	Karlskrona Manifesto Principles
Phase 1. Project Definition	**P1-** This ensures that the project initiation considers sustainability in the overall project definition from the beginning. **P2-** Software sustainability has different dimensions that have to be considered from the beginning for better project management with different stakeholders. **P3-** Software project usually involves stakeholders from different domains, incorporating their sustainability concerns provides better management of those concerns from multiple perspectives which can help the incorporation of sustainability for the software.
Phase 2. User Requirements Definition	**P2-** It is important to take note of user requirements in relation to each of the sustainability dimensions in order to have better sustainability analysis during the analysis and design phase
Phase 3. System Requirements Definition	**P4-** It is important during elicitation of system requirements to consider sustainability concerns for the system during the requirements definition even when it is not a core part of the user requirements. **P5-** It is also important to cross evaluate the consequential impacts of the system sustainability requirements and the environment in which the system will function.
Phase 4. Analysis and Design	**P2-** Applying this principle provides a blueprint for system evaluation from all sustainability dimensions (Economic, environment, social, individual and technical). **P4-** At this phase, this principle helps to encourage analysis of system design based on sustainability in order to facilitate better sustainable system. **P6-** Application of this principle enables better visual and visible overview of the system from different levels of abstraction.

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

	P8- This will provide better understanding during analysis to make better choices that will help the potential users of the system in present and in future when the system evolves.
Phase 5. Development	**P2-** This will encourage developers during this phase to consider different sustainability dimensions especially technical, social and individual dimensions **P4-** Encourage the search for better avenues to make the system sustainable from the development perspective (developers) and also the functions of the system to aid longevity.
Phase 6. Integration and Testing	**P2-** Provides integration and test team to have a sustainability template that can be used to test the system for all sustainability dimensions based on the sustainability requirement output from phase 2, 3, and 4. **P4-** Application of this principle will aid consideration of sustainability in this phase even if the primary focus of system is not about sustainability.
Phase 7. Implementation	**P5-** Provides a beforehand reasoning for the development team to consider sustainability of the system, its production environment and when push live for use. **P7-** Based on principle 5 (P5), this principle will aid consideration of seeking the involvement of different stakeholders to make the actualization of the system sustainability possible in the production environment and when pushed live.
Phase 8. Sustainment / Maintenance	**P9-** This principle at this stage help to create the conscious awareness so that when the system is in live environment, there will be continuous evaluation to assess the system sustainability and think of ways for optimizing and improving sustainability of the system from the different dimensions.

Table 2 highlights the first step towards putting the Karlskrona manifesto principles into practice of software development. The Karlskrona manifesto focused on high level principles, not techniques [12], which means there is need to exemplify the principles to show the practical usage with techniques. The following are the limitations of the manifesto that motivate the development of the software sustainability design catalogue (SSDC):

- The principles abstract and generic to serve all the possible stakeholders interested in sustainability in all the stages of the software development and management phases.
- The principles are on a high level of abstraction, missing many details for their practical usage.
- Some of them are closely related, making a trade-off among them difficult, especially for novice in the field of sustainability.
- The principles are not connected to tangible measures but serve as guide to create measures.

4. Structure of the Proposed Software Sustainability Design Catalogue

The software sustainability design catalogue (SSDC) contains information (recommendations) that can serve as guidelines (which are concrete actions) implementing one or more of the nine principles from the manifesto. The SSDC guidelines are collected during the analysis of several case studies. The contents of the SSDC cover different types of systems that are characterized and analysed using the manifesto principles, sustainability dimensions, orders of impact, current/future applications of manifesto principles, stakeholders, their goals, and indicators that will serve as source of evaluation.

SSDC is divided into two core components with the following content (see Table 4 and 5). The first component is Sustainability Analysis of Systems using the Karlskrona Principles. This covers the practical application of the Karlskrona Principles as criteria for evaluating current systems and

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

incorporating sustainability concerns into system design and development. The following are the core elements used in the analysis:

1. Karlskrona Principle: The right principle or principles are selected from the 9 principles (see **Table 1**) for evaluation of each system category. The principles are identified using a tag of **P1 to P9** (Principle 1 to 9).
2. Goal/Requirement: This highlights the desired end result for each system category based on sustainability consideration.
3. Current Principle Usage in software: This covers the current application of the principle in existing system design and development, even if it is not explicitly stated in current system documentation.
4. Future Principle Usage in software: Based on the evaluation of the current principle application in existing system design and development, a potential usage of the principle in future system enhancement and design is suggested.
5. Stakeholders: Those responsible for implementing the goals/requirement.
6. Question: Questions characterize each goals. From each goal, a set of questions is derived that will determine if each goal is being met.
7. Indicators: Specific indicators are used to answer the questions as a way to evaluate if the goals were achieved.

The second component is the **sustainability dimensions according to their order of impacts**. It covers a sustainability analysis using:

I. The Orders of Impact [41], [42] cover all the positive and negative effects of software on the environment which are decomposed into three orders of magnitude. The first order impacts (Immediate effects) are about the direct effects of the development and use of software system. The second order impacts (Enabling effects) are about the indirect impacts related to the effects of using the software system in its application domain. The third order impacts (Structural effects) are the cumulative long-term effects as a result of the accumulating first and second order impacts over time.

II. The Sustainability Dimensions including: [13]

- Economic sustainability aims at maintaining assets. For Software Engineering (SE): How can software systems be created so that the stakeholders' long term investments are as safe as possible from economic risks and increase profit?
- Environment sustainability seeks to improve human welfare by protecting natural resources. For SE: How does software affect the environment during and after development?
- Social sustainability means maintaining social capital and preserving the societal communities. For SE: What effects do software systems and applications have on the society (e.g. communication, interaction, government)?
- Individual sustainability refers to the maintenance of individual human capital. For software engineering (SE), it means: How can software be created and maintained in a way that enables user friendly solution for users and also ensure developers are satisfied with their job over a long period of time?
- Technical sustainability has the central objective of long-time usage of systems and their adequate evolution with changing conditions. For SE: How can software be created so that it can easily adapt to future change?

5. Feasibility Study and Framework

The feasibility study covers 4 case studies (See Table 3) of different kinds of systems used in creating the SSDC summarized in 8 Tables. The first author gathered a large set of data for analysing the four different case studies according to the Karlskrona manifesto principles and the orders of impact to create the design catalogue. The second and third authors then cross-validated the data collected. Based on the aggregated data, a first draft of catalogue was developed. Then, the proposed software sustainability design catalogue has be refined using more data using various types of applications.

Table 3. System types for feasibility studies

System Category	System Type
Cyber Physical System	Smart home system
Embedded System	Washing machine
Gaming	Angry bird
Desktop Application	Microsoft office

Table 4 and 5 below covers cyber Physical System (CPS) defined as the integrations of computation, networking, and physical processes that are tightly connect with its users. The sample system used in the catalogue is "Smart Home." Tables 4 and 5 provide information of the SSDC for Cyber Physical System (Smart Home) that can assist companies and software developers identify key areas that relate to sustainability and recognize strategic avenues on how current and future smart home solutions should be designed in a more sustainable manner. This enables them make good sustainability decision during and after design of smart home solution. Tables 7-12 which can be found in the appendix; cover:

- Embedded Systems that are composed of electrical and mechanical components completely encapsulated by the device they control. The sample case study used in this category is a "Washing Machine"[43 - 48] (see Table 7 and 8 in appendix) .
- Mobile Games as an application design that runs on mobile devices. The game case study used in this category is "Angry Bird Game" [49 -53](see Table 9 and 10 in appendix).
- Desktop Applications that run on standalone computers. The sample application used in this category is "Microsoft Office" [54 -57] (see Table 11 and 12 in appendix).

Table 4. Sustainability Analysis of Cyber Physical System (smart home) based on Karlskrona Principles [58 -61]

Karlskrona Principle and Goal	Current / Future Principle Usage	Stakeholders	Question	Indicator
(P2) Cross platform compatibility	**Current**: Smart home appliances are compatible with only few other manufacturers devices in the market. **Future**: Smart home appliances should be compatible with other devices from different manufacturers based on standard interface to avoid increase in energy usage. This can be achieved by enforcing device standards that can be used for cross platform compatibility. Home automation appliances should be economic and at same time environmental friendly.	Business Analyst	Can device function with other device from different manufacturer?	Device cross platform compatibility
(P4) Educate Users	Current Smart home solutions provides graphical information about energy usage but not necessarily	Software developers Business Analyst	Are users aware of their actions relating to	Usage data over time to detect changes in

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

	educate users on how to be energy conscious based on their daily habit over a period of time. **Future**: User data from smart home solution should be used for educating users thereby encouraging users to behave more sustainably. User data could also be used for prediction aim at optimizing the use resources such as water and energy within a household or company. The solutions could help interconnect other systems that can help save resource like water and electricity in a household or company.		electrical appliances in the house or company?	user habits
(P9) Reduce production and solution cost	**Current**: There are currently few cost effective solutions that will encourage user to adopt home automation solution on the long term **Future:** Use cheap environmental friendly resources in the production of home automation device (hardware) that can reduce production cost. If the cost of production reduces, the overall cost of smart home solution will also reduce which will increase its affordability among users. There can also be low cost solution for poor countries to assist in the use of water and energy judiciously (reduce waste).	Business Analyst	Did we manage to reduce costs compared to previous years (before solution was a smart home)?	Net cost of smart home solution

Table 4 of the SSDC (Cyber Physical System - Smart Home) highlights one important issue that standardization authorities in this domain can work on, which is the cross platform compatibility for smart home devices. Smart home appliances should be compatible with other devices from different manufacturer based on standards to avoid increase in energy usage. Smart home solutions should provide meaningful graphical information that can educate its users thereby encouraging users to behave more sustainably.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

Table 5. Sustainability Dimensions Order of Impacts for Cyber Physical System (smart home) [62 -65]

Order of Impacts	Environment	Economic	Technical	Social	Individual
1st	Increase in the use of natural resources in the production of hardware for smart home devices and pollution of the environment from toxic material used in production	Creates new business opportunities for those in this sector (setup and installation of devices at home)	Pave way for improving existing technologies and development of new tools to meet new market demands for sustainable usage of these technologies	Breeds new communities of users and suppliers	Users rely on devices to control some aspects of their lives at home and in offices
2nd	Reduce household energy consumption	Reduce household bill on energy consumption	High demand for security of user personal data (privacy).	Increase comfort, safety, flexibility, and security of user.	Demand for sustainable user friendly solution for home users
3rd	Increase in the use of toxic material for production of hardware. Less energy consumption over a long period of time.	Decrease on cost of energy through optimized solution over time	Efficient provision of sound technical solutions to avoid technical glitch that could lead to high energy usage. Encourage innovation on how to create cost effective technologies and devices to reduce household and company energy usage	Encourage users to form communities to share data as a way of encouraging each other to be energy conscious	Induces sustainable behaviour among users

Table 5 of the SSDC for Cyber Physical System (Smart Home) provides different insights on the direct, indirect and structural impact of home automation design and deployment from the different dimension of sustainability. From Table 3 companies and stakeholders will be able to incorporate the following sustainability goals for the design and development of home automation solutions:

- Environment: Reduce household energy consumption
- Economic: Reduce household cost on energy
- Individual: Provide user friendly solution for home users with easy to use user interface and information to induce sustainable behaviour among user.
- Technical: Provide good security for user personal data and avoid technical glitch that could lead high energy usage

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

- Social: Encourage users to form communities to share data as a way of encouraging each other to be energy conscious and environmentally aware of the consequences of their actions and inaction while using smart home solutions.

The application of these principles from the catalogue offers explicit goals and opportunities for sustainability integration in system design through multiple perspectives for systems with sustainability as its core goal and those system without sustainability as it main goal. The below are detailed description of the principles used in providing information on how best to engineer and think of sustainability for smart home solution (**see Table 4 and 5**) from the catalogue.

Sustainability has multiple dimensions (P2): The application of this principle provides an overview of the fundamental issues and positive opportunities that could encourage stakeholders in the smart home domain to cross reference in the design and development of smart home solutions. Especially during solution requirements from users and choosing appropriate boundaries.

Smart home design and deployment in this domain requires getting inputs for the effect of design solutions on the environment from natural resources used in building hardware devices, energy consumption of the devices, social behaviour and interaction between people in family (household), company and other places where these solutions will be deployed. This means all sustainability dimensions (environment, economic, social, individual, technical) will be analysed for better design output.

Sustainability is a concern independent of the purpose of the system (P4): The goal of most smart home solutions is to provide comfort and reduce energy consumption for its users, but it is important to consider an encompassing view of sustainability. This is to be able to get even more benefits such as reducing pollution through the use of environmental friendly materials in producing hardware devices used for smart home solution. The smart home solution can be used to educate and inform users about the negative consequential effect of their behaviour and habits. This can help induce sustainable behaviour among users. For a smart home solution design to be effective and meet user needs, it will require the expertise of a psychologist or at least an adequately educated interaction designer to help provide information according to the level of comfort and technical expertise of those in manufacturing, transportation, electrical, business and ICT discipline.

Sustainability requires long-term thinking (P9): It is important to think of how the smart home solution provided today will evolve to meet the requirement of current users and be adaptive enough to satisfy future user needs. This will require looking at measures to capture user behaviours over time through computational intelligence to predict future actions of users through data generated from time to time.

Based on the analysis of all the case studies in the feasibility study (refer to Table 3), a pilot framework to guide stakeholders involved in the design and development is proposed. Figure 1 provides a detailed flow of the pilot framework.

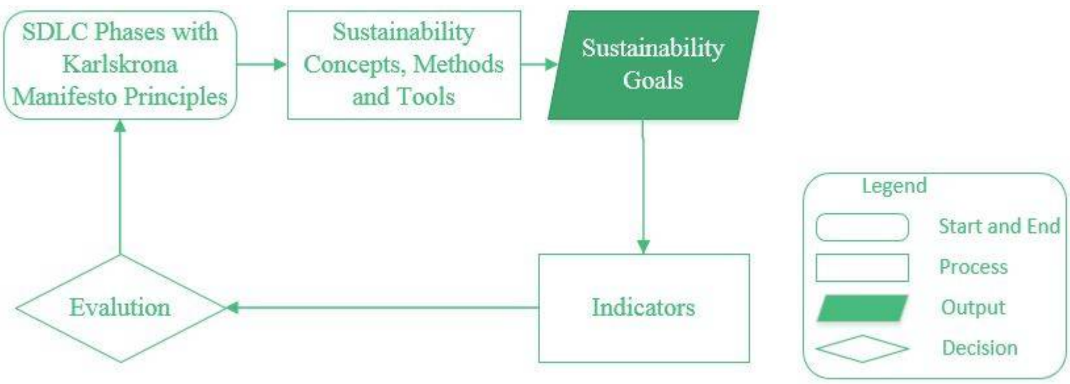

Figure 1. Pilot Framework for Sustainability of software system design based on SSDC

The pilot framework is the first derivative from the SSDC to assist developers incorporate sustainability during system design and development covering the software development life cycle (SDLC) phases. For better understanding, the pilot framework is presented below in a tabular form to show contents that are involve in the framework. Table 6 contains all contents of the framework. It is important to highlight that the indicators used in the framework (Table 6) are influenced by the nine Karlskrona manifesto principles mapped to each of the software development life cycle (see Table 2).

Table 6. Contents of Pilot Framework for Sustainability of Software System Design

SDLC Phases and Karlskrona Manifesto Principles	Sustainability Goals	Sustainability Concepts, Methods and Tools	Indicators
Phase 1. Project Definition, P1, P2 and P3	Transmaterilization, Design for sustainable efficiency, Reusability	Cradle to cradle, Biomimicry, Sustainable Business Canvas,	Carbon footprint, material footprint, end of life footprint
Phase 2. User Requirements Definition, P2	Increase sustainability awareness among users	Helix of Sustainability	Total number of sustainability requirements, Priority assign to sustainability requirements
Phase 3. System Requirements Definition, P4, and P5	Design for efficiency, sustainability awareness and Interoperability	Biomimicry, Cradle to cradle, Goal Model	Total number of system goals relating to sustainability dimensions
Phase 4. Analysis and Design, P2, P4, P6 and P8	Design for reuse and efficiency, localization, Interoperability	Biomimicry, Helix of Sustainability, Life cycle sustainability assessment, Social return on investment, Sustainability analysis radar chart	Number of first, second and third order impacts of system identified
Phase 5. Development, P2 and P4	Design for reuse, Design for module replicability, Design for efficiency, Design for sustainability awareness, Design for efficiency, Design for easy service and maintenance	Biomimicry, Cradle to cradle	Number of coding choice influenced by sustainability, number of features (functions) added to systems to inform users about sustainability through functions like eco feedback.
Phase 6. Integration and Testing, P2 and P4	Design for easy assembly and disassembly, Design for durability,	Cradle to cradle, Sustainability analysis radar chart, Life cycle sustainability assessment,	How much of information from sustainability analysis chart was used during integration and testing such as the number of systems functions tested against sustainability concerns such as the first order (immediate) impact and possible second order (enabling) impacts of the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

			system
Phase 7. Implementation, P5 and P7	Design for easy use, design to induce conscious sustainability awareness, Design to educate users about sustainability, Design for easy recycle	Biomimicry, Cradle to cradle	The priority assign to sustainability by developers and the system owners /users during after implementation
Phase 8. Sustainment / Maintenance, P9	Proper design for serviceability, Design for easy replacement of code modules, Design for continuous user engagement through sustainability awareness	Life cycle sustainability assessment, Sustainability analysis radar chart, Cradle to cradle	Number of improvement to system based on sustainability requirements either from users feedback or developers

6. Application of the SSDC and the Pilot Framework

In order to showcase the application of SSDC and pilot framework, an excerpt from a Cyber Physical System (Smart Home) is used here. Consider an application scenario below:

"A software engineer called Henry has the task of eliciting and documenting the requirements for a new smart home system. Being aware of his responsibility for the software system sustainability he creates, and its impacts, he takes the template of the sustainability analysis of the five dimensions and the three orders of effects from the design catalogue with him to meet the customer during their first meeting. The customer is curious about these additional analysis ideas, and Henry explains to his client what they mean and gives his client couple of examples. Then, together with the customer, he fills out the template applying the concepts from the design catalogue (SSDC) to find out what those dimensions and orders of impact mean for the smart home system the customer wants for his house. The information from the activity goes into the requirements analysis that is subsequently conducted and used as a measurement yardstick during the smart home system development and deployment."

To showcase the use of the framework in the above scenario, the following explanation breakdown how the pilot framework for software sustainability design was used in creating the smart home system from the planning to requirement phase and finally delivery of system.

Phase 1(Project Definition) with Karlskrona principles 1, 2 and 3: Henry uses the sustainable business canvas to show value that can be generated through sustainability consideration and how it can help improve the product. Henry was able to pinpoint two sustainability goals from this phase, which is design for sustainable efficiency, and to create sustainability awareness through the smart home system by facilitating a community of users willing to share their energy usage to motivate each other.

Phase 2 (User Requirements Definition) with Karlskrona principle 2: From the information gathered in phase one and a discussion with the client, Henry was able to identify the goal of increasing the sustainability awareness among users of the system once it is created based on the sustainability helix concept. These were the indicators from this phase: percentage of reduce energy usage of the household, amount of feedback on the environment impact of energy used (CO_2) by the family through eco feedback, number of suggestions provided on how to improve household energy usage based on usage patterns.

Phase 3 (System Requirements Definition) with Karlskrona principles 4 and 5: The goal in the phase of system requirements is to design for efficiency and sustainability awareness based on the overall system goal from phase 1. He uses the goal model to showcase how the system goals were broken into smaller piece based on the system requirements in order to identify requirement conflicts that might occur. Some of the smaller goals based on the overall goal in this phase includes:

reduce energy consumption, reduce co2 emissions, establish community of users sharing energy usage data, ensure high availability of system, and provide eco-feedback.

Phase 4 (Analysis and Design) with Karlskrona principles 2, 4, 6 and 8: In this phase, the main goals are design for easy usage, efficiency and sustainability awareness. Using the sustainability analysis chart (see Figure 2). Figure 2 portrays the sustainability analysis of the smart home system design for the first, second and third (Immediate, Enabling, and Structural) impacts of smart home solutions from the different sustainability dimensions. Information from the analysis provides avenue for evaluating while guiding different stakeholders (Managers, Developers, and Users) on the benefits to aspire for sustainability in smart home solutions.

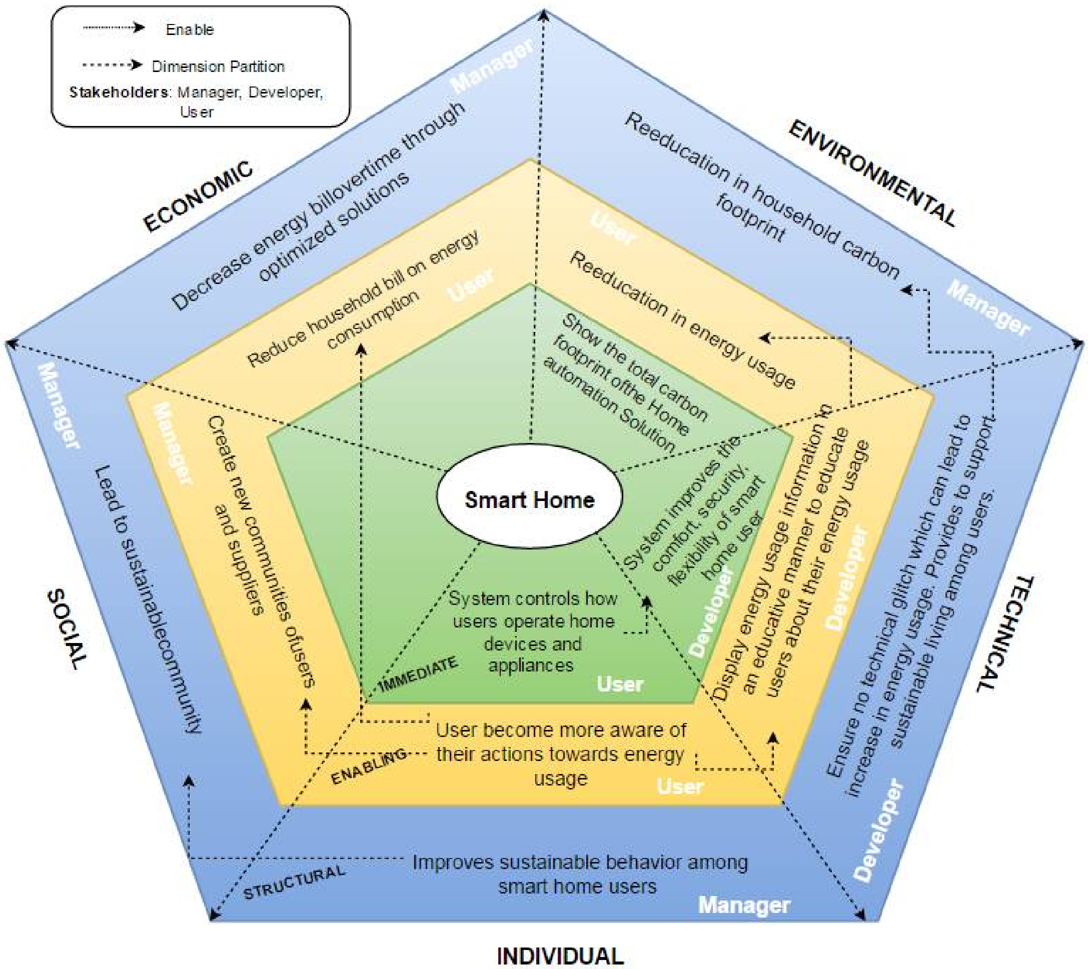

Figure 2. Immediate, Enabling and structural effects of Smart Home Solution in Sustainability Dimension

The impacts described in Figure 2 are then taken into account during the requirements refinement phase such that they will actually get deployed and implemented in the system. The indicators from this phase are the immediate, enabling and structural impacts of the system identified from the sustainability analysis chart.

Phase 5 (Development) with Karlskrona principles 2 and 4: The goal of the development phase is to design a smart home system with the ability to create sustainability awareness among users and reduce household energy consumption. The sustainability concept that influences developers during this stage is biomimicry. This concept encourage developers to rethink how to create the smart home system with functions that will help reduce energy consumption of the software during usage, provide eco-feedback to users in a way to create sustainability awareness and also to be used an enabler for reducing household energy consumption. The biomimicry concept is visible in the energy UI dashboard of the smart home system where a full grown tree is used to mimic energy consumption. As the energy consumption increases, the tree leaf starts to change

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

colour to brown (indicating that is dead) and when energy consumption reduces the tree becomes more green (indicating the tree is back to life).The following are functions added to the system which are used as indicators in this phase: Percentage of energy saved during system usage, the accuracy in the eco-feedback based on the energy usage pattern of users, total number of times users shared their energy usage percentage with friends on social media, the level of user comprehension and understanding of the displayed information on CO2 emission and indicator for amount of saved trees based on reduce energy usage over time.

Phase 6 (Integration and Testing) with Karlskrona principles 2 and 4: This stage goal is the design for easy assembly and disassembly using the sustainability analysis chart (see Figure 2) and indicators from the other phases as guide for integrating and testing the smart home system.

Phase 7 (Implementation) with Karlskrona principles 5 and 7: The following were goals of this phase: design for easy use, design to induce conscious sustainability awareness and design to educate users about sustainability which was influenced by the sustainability concept of biomimicry. These are used as indicators: The change in developers coding practice based on sustainability consideration, the effectiveness of functions added due to sustainability requirements such as percentage of energy saved during system usage, the accuracy in the eco-feedback based on the energy usage pattern of users, understandability of presented information on CO2 emission and amount of saved trees based on reduce energy usage over time.

Phase 8 (Sustainment / Maintenance) with Karlskrona principle 9: This phase covers the long term goal of the smart home system such as design for serviceability and design for continuous user engagement through sustainability awareness (eco-feedback). The indicators used in this phase are: the efficiency of data generated in optimizing the smart home system, the effectiveness of the eco-feedback to improve user behaviour, the total percentage of energy saved over time and the backlog management index (BMI).

Switching to a process perspective, we now try to understand what the effects of the principles detailed in the SSDC for smart home solution (see Table 4 and 5) mean for a process engineer based the above example for smart home system. For improving the software development process, the principles considered relevant for the example at hand were principles 2 (multi-dimension), 4 (independent of purpose), and 9 (long-term thinking). Principles 2, 4 and 9 are further explained here as a way to show how the Karlskrona manifesto principles influence decisions for development of the solution for the above scenario of smart home.

Principle 2 from the SSDC highlighted the need for cross platform compatibility, which comes during the project definition of the Software Development Life Cycle (SDLC) as an initial step to put this goal in motion throughout the whole software development. It also becomes relevant during the user requirements elicitation when speaking to different stakeholders all the way through the implementation and training.

Principle 4 emphasizes the need to educate smart home users about how their action and inaction affects the environment through their usage of smart home devices. This principle comes into play from the user requirements stage down to the implementation. Even if the users are not interested in sustainability, as software developers, designers and engineers, it is our responsibility to inform users about the benefit of sustainability during requirement gathering [66], as seen in the example of Henry from the earlier example in the beginning of this section. Efforts to educate users also go into the documentation processes, as at some point in the system development, the developer needs to create user documentation and it should include information about energy usage, as well as ways of saving resources.

Principle 9, which is about reducing production and solution cost, plays a vital role from the beginning of the SDLC in project definition and planning from management side. It is important to identify choices that benefit both the current and future users, as well as to identify how the solution can be cost beneficial to encourage more users. The business analyst takes charge of this before moving on to the user requirements stage and this issue is monitored throughout the whole project development.

Again, to see the practical application of this, let's consider an **application scenario:**

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

A company named Energy Life, based on the SSDC analysis decides to provide a game like menu to control smart home devices for a family named Miralles. The family only wants to reduce energy cost in their household. Mark, the deployment manager, has read the SSDC for smart home (see Table 4 and 5), pilot framework (Figure 1) Table 6 (Framework description) and the sustainability analysis for smart home solution (see Figure 2), he realizes the best way to help reduce the energy cost of the Miralles is to create an effective way to educate them about their daily habits through the smart home technologies that he will deploy in their home.

Mark therefore creates a game-like menu for the Miralles to control their smart home devices. It provides information about the energy consumption of each home device and notifies them of their energy usage. Within a few days the Miralles are able to see the flow of their energy usage and how their daily habits impact the unnecessarily high-energy use within their household. The game provides the family tips on how they can save energy and also based on the information about their energy usage, they are able to identify the amount of energy consumed by their washing machine. So, for example, they decide not to run a half empty washing machine again as they can see that this happens almost every day, and even with the half-load mode it would make more sense to run a full one every two days instead. The Miralles also started a new habit of reminding each other to switch off the computer, TV and other household appliances when they are not in use.

Later on, when the company wants to update their system, they revisit those orders of effect and the different dimensions and thought about what may and may not have changed in order to improve their system to be more efficient and effectively with regards to sustainability.

Base on the definition of a smart home by Nicholl et al. [67] as "dwellings that use integrated communication systems to monitor and manage the performance of the home, and to support the lifestyle choices of the occupants.", the above scenario shows potential avenues of using automation to support sustainability in smart home solution through:

- Automatic analysis of user data to provide educative information in the GUI to induce sustainable behaviour among users,
- Alerting user through notification when electrical devices or appliances are running without being used,
- Automatic scheduling of task such as washing cloths and dishes when energy rate is low during the day, and
- Projecting when to turn on/off heating and lighting based on season and user behaviour (prediction).

Though the challenge of most smart homes is that most of the technology today does not have a standard interface that can facilitate cross platform compatibility as highlighted by Makonin et al. [65] and in the design catalogue (see Table 1), there is still potential to improve and overcome this challenge by smart home devices companies.

7. Discussion

The SSDC provides a comprehensive overview of sustainability design considerations and requirements for systems and applications in different domains. The pilot framework for sustainability of software design exemplifies the use of the SSDC. For example the sustainability analysis chart during the analysis phase of SDLC for the smart home system provides a variety of information for different stakeholders for the direct, indirect and structural effects of sustainability in smart home design and deployment. This information can then be used to create or enhance processes, methods and tools that can automate the incorporation of sustainability into the design of smart home solutions. Specifically, in Section 6 we propose use of a template for Sustainability Analysis (see Figure 2), a sustainability analysis diagram that contains a sector for each dimension, and uses the rings around the centre to depict the orders of impact.

The indicators from each phase of the SDLC while applying the pilot framework provide a way to evaluate the process and derivative from each of the SDLC phase influenced by different sustainability concepts which aided sustainability goals for each of the SDLC phases. The application of sustainability methods and tools used illustratively, such as sustainability business canvass, goal model and sustainability chart diagram, provides developers and engineering with a way to structurally elicit and manage sustainability requirements and monitor system impacts (immediate, enabling and structural).

The analysis diagram allows for the elicitation of facts around the impact of the system on its application environment, society, economy, and the natural environment, and gives an overview of potential long-term consequences. Figure 2 depicts an instance of such a Sustainability Analysis diagram for the smart home solution. In addition to dimensions (sectors), orders of impact (rings), and the actual effects (black text in the fields), the roles of responsible stakeholders are identified in the white text in the fields.

The design catalogue and the underlying pilot framework can be beneficial for the following stakeholders interested by sustainability, its engineering and integration in/for software systems design and development:

- For companies and software developers, serve as guide on how sustainability can be incorporated into software design and development. It can also enable them to identify effects of their project on technical, economic, social, individual, and environmental sustainability. Furthermore, we support the current revision of the ACM code of ethics and propose to incorporate sustainability principles and explicitly acknowledge the need to consider sustainability as part of professional practice [11].
- For the standardization organizations can benefit from it to create future standards for software and organizational sustainability. SSDC shows areas where software applications can impact the environment and humans, and this information can help create standards that would encourage companies and stakeholders to improve existing and new applications and policies to promote sustainability.
- For Public Authorities will be able to use the information from the catalogue to enact new laws persuading industry practitioners to design software systems, application and devices in a more sustainable manner.
- For academic institutions, it helps to identifying avenues to advance research on the sustainability by design, sustainability design patterns and tools support, among others.

8. Conclusions

An effective sustainability engineering and integration requires clarifying the current perceptions of sustainability and defining a concrete framework for its engineering and measuring. As first milestone, this paper presented a catalogue that quantify sustainability via a series of guidelines that can be used for incorporating sustainability into the design loop.

By analysing how the principles defined in the Sustainability Design Manifesto can be applied for some specific systems, we are able to identify a series of guidelines and develop the foundations for a “sustainability by design” approach. First, we reviewed the current perceptions of sustainability for various types of systems. Further, based on how sustainability has been perceived in three other fields, and in light of the basic assumptions behind the Karlskrona Manifesto, the SSDC has been defined. Each guideline is defined as a set of principles, dimensions of sustainability, orders of impact, and indicators. The usage and applicability of the catalogue have been demonstrated for four types of systems.

Future research includes examining other types of systems and the application of the guidelines in an industry settings. This will give better insights for the development of the guidelines for different types of systems and its usages by the diverse stakeholders in the software development lifecycle. An important aspect of its validity is that the catalogue was created based on the expertise of the wide set of researchers involved in the sustainability design manifesto. Considering the fact that sustainability in software engineering is still evolving, the SSDC provides a common ground for further research. Another limitation of its validity is the industry is not yet highly involved in sustainability concerns. Consequently, the provided theoretical validation can only serve as outline for an empirical evaluation to be conducted in future.

The SSDC also has automation potential in the future. The design catalogue can become the basis for a recommender system. This would help developers to identify and apply effectively the

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

sustainability guidelines. However, this requires more case studies to building a knowledge base required by a recommender system. The automation will provide a practical guide to enable developers during the each stage of the design and development to understand and incorporate the Karlskrona principles, sustainability goals, concepts, tools and methods with indicators that can help in evaluation of the software system.

Finally, this paper provides a foundation (via the SSDC and pilot framework) for the software engineering community to design and engineer sustainability.

Author Contributions: For S.O. conceived the idea of the paper; S.O. and B.P designed the research methodology; Investigation and Data collection S.O.; Data Validation B.P. and A.S.; S.O. wrote the paper and all authors contributed to reviewing all section; Review & Editing Supervision B.P and A.S.

Funding: This research received no external funding

Acknowledgments: Authors are grateful to the researchers who initiated the Karlskrona manifesto principles which was used in this article.

Conflicts of Interest: The authors declare no conflict of interest

Appendix A

Table 7. Sustainability Analysis of Embedded System (Washing Machine) Based on Karlskrona Principles

Karlskrona Principles and Goal	Current Principle Usage	Future Principle Usage	Stakeholders	Question	Indicator
(P 9) Efficient water usage	Current washing machine has some design features to help reduce water wastage during washing circle.	Good mechanism within the washing machine to aid efficient use of water during washing cycle and also display the amount of water saved to users. This will serve as means of educating users about water wastage. This will aid positive impact on the amount of water usage in a household.	Software Developer	Does the washing machine reduce water usage?	Washing Cycle/Total amount of water used
(P 8, 6) Energy Efficiency	Some current sets of washing machine has eco-friendly features to reduce energy usage.	One good feature to reduce energy usage of washing is to turn off or hibernate automatically after washing cycle if idle for 2 minutes. It will help reduce energy cost (P 6) and also reduce resource usage on the long term (P 8). This will help reduce energy consumption	Software Developer	Does the machine use too much energy for a single washing cycle?	Energy Efficiency (Washing Cycle /Total amount of energy used)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

		when machine is idle. Incorporate the use of scheduler to the washing machine as a way to time when the washing cycle should start during the period of the day when energy cost is less.			
(P 8/ 9) Water efficiency	Allows collection of grey water from washing machine	Encourage reuse of greywater (with biodegradable laundry detergent) in garden watering.	Business Analyst	How much of the greywater gets reused afterwards?	Percentage (%) of reused water

Table 8. Sustainability Dimensions Order of Impacts for Embedded System (Washing Machine)

Order of Impacts	Environment	Economic	Technical	Social	Individual
1st	Increase in the use of natural resources in the production of washing machine components such as iron, copper, Medium-density fibreboard, EPDM rubber (ethylene propylene diene monomer).	High demand for natural resources boost the economies of countries with those natural resources	Increase demand for new technologies, tools and equipment for extracting raw materials.	New job opportunities for people	Increase the risk of having skin diseases due to toxic material exposure
2nd	Increase in water and energy usage when using a washing machine as opposed to manual washing	High energy cost for household as a result of increase in energy usage due to ease of washing cloths (convenience factor and little manual labour).	Demand for energy and water saving mechanism in washing machine.	More job opportunities for technicians with knowledge of washing machine technologies	Increase the ease of washing for users
3rd	Over a long time it lead to increase water usage and a culture of washing a lot	Increase in profit for washing machine production	Over the time there will be pressure to build a more energy and	Increase job creation over time both from industry for skilled	Over a long time it lead to increase water usage and a culture

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 May 2018 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

	(e.g., California ;) In turn, that leads to a higher wear & tear of the hardware.	companies with high demand for washing machine from users.	water efficient washing machine from manufacturers to have competitive edge over other competitors.	workers and in household for technicians to fix minor issues of washing machine.	of washing a lot (e.g., California ;) In turn, that leads to a higher wear & tear of the hardware

Table 7 and 8 provides information on sustainability analysis of embedded system (washing machine) using the Karlskrona principles and sustainability dimensions based on usage of order of impacts. Based on output from Table 7 and 8, water and energy efficiency as key objectives of sustainability for a washing machine.

Table 9. Sustainability Analysis of Mobile Games (Angry Bird) Based on Karlskrona Principles

Karlskrona Principles and Goal	Current Principle Usage	Future Principle Usage	Stakeholders	Question	Indicator
(P 1) Energy Efficient	Currently game development focus more on usability and fun factor aspect than sustainability aspect.	Create the mobile game architecture with sustainability consideration. Since sustainability is systemic, it should be core of the application structure. Consider energy efficiency during game development Incorporate Green patterns to game application development	Software Developer	Is the mobile energy Efficient?	(Energy Efficiency) useful-work done/energy used
(P 1) Reduce wear and tear of hardware	Though sustainability is not the core of current game development practices, though game developer tries to ensure optimal use of	Ensure that the mobile game during operation uses hardware resources (Memory, CPU etc.) in efficient and sustainable manner to	Software Developer	What is the impact of the game on hardware component s like CPU and RAM	Does the game use too much hardware resources?

	hardware resources to make game run faster and better on hardware (phone, computer, tablets)	reduce wear and tear.			
(P 2) Sense of belonging to a community and connectedness to other people	Current Game development provide sort of community for it users in form of forums and groups online.	There is a digital community surrounding most games and the question is how compares does to face to face communities for gamers. Create a community that make gamer's feel connected to both the digital and real world. Make gamers feel like they are part of something (people always want to belong to a tribe).	Business Analysts, Software Developer	Is there a sense of community amongst players?	Connectedness in community? Number of 'friends' in a particular gaming community?
(P9) Good user experience	Current game development incorporates user experience into their production to gain more user base and profit.	Game should use reasonable lighting effect for the game display which can help reduce energy usage and also incorporate sustainability concept in the total overall design of the game. It can also add features that will educate users about sustainability.	UX and software Developer	Can users complete their task easily?	Gateway Metrics

Table 10. Sustainability Dimensions Order of Impacts for Mobile Games (Angry Birds)

Order of Impacts	Environment	Economic	Technical	Social	Individual
1st	Increase in energy usage from computers and mobile devices used for developing and testing game application	Cost of production for game development companies	Increase in demand of sophisticated hardware and software for game development	Open job opportunities for game developers	Provides avenues for leisure activities for users
2nd	Increase in energy usage because user are using mobile phone, laptops, iPad to play game. There will also be need for charging of these devices coupled with the energy consumption when playing the game	Company makes profit from game purchase. Increase demand and user cost for hardware (computers, phones and tablets).	Demand for better graphics and quick game response from users.	Create community sense among angry bird users.	Demand for good user experience while interacting with the game
3rd	Overtime leads to hardware wear and tear because of continuous game time from user side on computer/Phones and continuous game development from game production company side.	Increase in profit from huge user and fan base for the game company.	Demand for newer features and innovation from customers	Increase the sense of belonging in form of community among users	Lead to game addiction and lack of social interaction with outside world.

Table 9 and 10 provides information about sustainability analysis for mobile game (Angry Bird*)* centred on Karlskrona principles and sustainability dimensions based on order of impacts. Table 9 provides points at energy efficiency, reduction of wear and tear of hardware and creating sense of

 doi:10.20944/preprints201805.0464.v1

Peer-reviewed version available at *Sustainability* **2018**, *10*, 2296; doi:10.3390/su10072296

belong to community among users as key goals of sustainability for stakeholders. In addition, details from Table 10 prompts the need to aspire for these goals from all sustainability dimension:

- **Environment**: Optimize energy and computing resource consumption during game development and when users are playing game
- **Economic**: Provide continuous innovation on the game features to encourage current users to keep playing the game and attract new users and ensure game is maintainable (longevity)
- **Technical**: Ensure that game does not encourage quick hardware wear and tear and at same time has the ability to evolve with new demands of the market
- **Individual**: Good user experience while interacting with the game and serves as a medium of inducing sustainable behaviour.
- **Social**: Create good community sense among angry bird users and educate them about sustainability

Table 11. Sustainability Analysis of Desktop Application (Microsoft Office) Based on Karlskrona Principles

Karlskrona Principles and Goal	Current Principle Usage	Future Principle Usage	Stakeholders	Question	Indicator
(P1) Incorporate sustainability into development process	Current development framework allow the use of scrum and version control	Provide a development framework that support sustainability focusing on the software and those developing the software itself.	(Development) Process Engineer	Are guidelines available?	Boolean (Yes or No)
(P8) Offer reasonable amount of features	Currently all office application comes with all the features which sometimes are rarely used by users	Provide all basic features for office application and allow users to add other features when needed.	Business Analyst and Software Developer	What is the amount of changes to be made to add new features?	Rework Metric
(P9) Add green print (and let user know how many pages they save over time)	Users can print any document as their need requires	Incorporate Green print to office application that inform users whenever they want to print a document for the second time that they can skip few pages because changes were not made on those pages	Software Developer	Do people print less by using this green print button?	Number of pages printed compared to if there was no green button

Peer-reviewed version available at Sustainability 2018, 10, 2296; doi:10.3390/su10072296

Table 12. Sustainability Dimensions Order of Impacts for Desktop Application (Microsoft Office)

Order of Impacts	Environment	Economic	Technical	Social	Individual
1st	Reduce the amount of manual writing on papers which in turn reduce the amount of papers used by people for writing.	Open new potential market for company to explore	Ensure the ability of Word application to meet new market demands from the technical implementation aspect	Creates new job opportunities for those that are expert with most of the office applications.	Provide efficient way of doing daily task such as documentation, project management and design.
2nd	Reduce paper resource wastage	Increase in profit for Microsoft and other partner companies through sales of office application	Increase in use of computers for development of add-ons in and plugins for office applications	Provides a community feeling among users.	Guarantee that user will easily finish their task while using the application for their day to day work.
3rd	It will increase energy usage over time do to ease of doing daily task on computers, phones and tablets.	Rise in company's profit through product innovation as demands changes from generation to generation	Demand for new features to meet new demands	Improve connectedness among users through collaboration online	Improves the overall ability of users completing their task (documentation, project management and design)

Table 11 and 12 provides information on sustainability analysis of Desktop application (Microsoft Office) using the Karlskrona principles and sustainability dimensions based on order of impacts.

References

[1] IBM, "Global CEO Study: The enterprise of the future," *Available online: Https://www-935.ibm.com/services/uk/gbs/pdf/ibm_ceo_study_2008.pdf Accessed on 11-10-2017*, p. 76, 2010.

[2] C. Calero and M. Piattini, "Introduction to Green in software engineering," *Green Softw. Eng.*, pp. 1–327, 2015.

[3] B. Christoph *et al.*, "Requirements: The key to sustainability," *IEEE Softw.*, vol. 33, no. 1, pp. 56–65, 2016.

[4] S. A. . Koçak, G. I. . Alptekin, and A. B. . Bener, "Evaluation of software product quality attributes and

Peer-reviewed version available at *Sustainability* **2018**, *10*, 2296; doi:10.3390/su10072296

environmental attributes using ANP decision framework," *CEUR Workshop Proc.*, vol. 1216, pp. 37–44, 2014.

[5] Ericsson, "Technology for Good," *Available online: https://www.ericsson.com/assets/local/about-ericsson/sustainability-and-corporate-responsibility/documents/2015-corporate-responsibility-and-sustainability-report.pdf Accessed on 30-11-2017*, 2014.

[6] B. Penzenstadler, "What does Sustainability mean in and for Software Engineering ?," *1st Int. Conf. ICT Sustain.*, 2013.

[7] C. C. Venters *et al.*, "Software sustainability: The modern tower of babel," *3rd Int. Work. Requir. Eng. Sustain. Syst. Work. Proc.*, vol. 1216, pp. 7–12, 2014.

[8] J. R. Ehrenfeld, *Sustainability by Design: A Subversive Strategy for Transforming Our Consumer Culture*. .

[9] P. Freeman, "The Central Role of Design in Software Engineering: Implications for Research," *Softw. Eng. Res. Dir.*, pp. 121–132, 1980.

[10] M. P. Robillard, "Sustainable Software Design," pp. 920–923.

[11] C. Becker *et al.*, "The Karlskrona manifesto for sustainability design," *arXiv1410.6968 [cs]*, vol. 20, no. May, p. 2014, 2014.

[12] C. Becker *et al.*, "Sustainability Design and Software: The Karlskrona Manifesto," *Proc. - Int. Conf. Softw. Eng.*, vol. 2, pp. 467–476, 2015.

[13] B. Penzenstadler and H. Femmer, "A generic model for sustainability with process- and product-specific instances," *GIBSE 2013 - Proc. 2013 Work. Green Softw. Eng. Green by Softw. Eng.*, no. June 2015, pp. 3–7, 2013.

[14] G. Saval, M. Mahaux, and P. Heymans, "Discovering Sustainability Requirements: An Experience Report," *Uucs*, 2014.

[15] R. Chitchyan, S. Betz, L. Duboc, B. Penzenstadler, C. Ponsard, and C. C. Venters, "Evidencing sustainability design through examples," *CEUR Workshop Proc.*, vol. 1416, pp. 45–54, 2015.

[16] N. Shedroff, "Design is the Problem:The Future of Design Must be Sustainable," p. 582, 2009.

[17] M. Bibri, "Sustaining ICT for Sustainability," 2009.

[18] L. Musthaler, "Energy-aware software design can reduce energy consumption by 30% to 90%," *Netw. World*, 2014.

[19] Z. Durdik, B. Klatt, H. Koziolek, K. Krogmann, J. Stammel, and R. Weiss, "Sustainability guidelines for long-living software systems," *IEEE Int. Conf. Softw. Maintenance, ICSM*, pp. 517–526, 2012.

[20] T. Johann, M. Dick, E. Kern, and S. Naumann, "Sustainable development, sustainable software, and sustainable software engineering: An integrated approach," *2011 Int. Symp. Humanit. Sci. Eng. Res.*, pp. 34–39, 2011.

[21] G. saval Martin, mahaux, patrick heymans, "Requirements Engineering: Foundation for Software Quality," *Requir. Eng. Found. Softw. Qual.*, vol. 4542, no. January, pp. 247–261, 2007.

[22] K. Erdélyi, "Special factors of development of green software supporting eco sustainability," *SISY 2013 - IEEE 11th Int. Symp. Intell. Syst. Informatics, Proc.*, pp. 337–340, 2013.

[23] M. Dick and S. Naumann, "Enhancing software engineering processes towards sustainable software product design," *24th Int. Conf. Informatics Environ. Prot. (EnviroInfo 2010)*, vol. 2010, pp. 706–715, 2010.

[24] S. Oyedeji, A. Seffah, and B. Penzenstadler, "Sustainability Quantification in Requirements Informing Design," *6th Int. Work. Requir. Eng. Sustain. Syst.*, vol. i, 2017.

[25] B. Penzenstadler, "Infusing green: Requirements engineering for green in and through software systems," *3rd Intl. Work. Requir. Eng. Sustain. Syst. 2014*, vol. 1216, no. 1, pp. 44–53, 2014.

[26] K. Roher and D. Richardson, "Sustainability requirement patterns," *2013 3rd Int. Work. Requir. Patterns, RePa 2013 - Proc.*, pp. 8–11, 2013.

[27] A. Raturi, B. Penzenstadler, B. Tomlinson, and D. Richardson, "Developing a sustainability non-functional requirements framework," *Proc. 3rd Int. Work. Green Sustain. Softw. - GREENS 2014*, pp. 1–8, 2014.

[28] M. Colmant, R. Rouvoy, and L. Seinturier, "Improving the energy efficiency of software systems for multi-core architectures," *Proc. 11th Middlew. Dr. Symp. MDS 2014 - co-located with ACM/IFIP/USENIX 15th Int. Middlew. Conf.*, pp. 2–5, 2014.

[29] G. Lami and L. Buglione, "Measuring software sustainability from a process-centric perspective," *Proc. 2012 Jt. Conf. 22nd Int. Work. Softw. Meas. 2012 7th Int. Conf. Softw. Process Prod. Meas. IWSM-MENSURA 2012*, pp. 53–59, 2012.

[30] C. Calero, M. F. Bertoa, and M. Angeles Moraga, "A systematic literature review for software sustainability measures," *Green Sustain. Softw. ({GREENS)}, 2013 2nd Int. Work.*, pp. 46–53, 2013.

[31] R. Seacord *et al.*, "Measuring Software Sustainability," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[32] B. Penzenstadler, M. Martin, and S. Camille, "RE4SuSy: Requirements engineering for Sustainable systems," *CEUR Work. Proceedings, Retrieved from Http//ceur-ws.org/Vol-1216/*, vol. 995, 2013.

[33] B. Christoph, "Sustainability and longevity: Two sides of the same quality?," *CEUR Workshop Proc.*, vol. 1216, pp. 1–6, 2014.

[34] C. Becker *et al.*, "Website for The Karlskrona manifesto for sustainability design," *arXiv1410.6968 [cs] Available online Http//sustainabilitydesign.org/karlskrona-manifesto/ Accessed 10-10-2017*, vol. 20, no. May, p. 2014, 2014.

[35] M. Fowler and J. Highsmith, "The agile manifesto," *Softw. Dev.*, vol. 9, no. August, pp. 28–35, 2001.

[36] B. R. Group, "The Business Rules Manifesto," *Bus. Rules Group. Version Available online http//www.businessrulesgroup.org/brmanifesto.php Accessed 12-11-2017*, no. c, pp. 1–2, 2003.

[37] A. Arsanjani *et al.*, "SOA Manifesto," *SOAManifesto*, vol. 9, no. 1, pp. 82–8, 2009.

[38] L. Felikson, "The SOA Manifesto : Establishing a Common Understanding About SOA and Service-Orientation History of The SOA Manifesto," no. October, 2010.

[39] I. Gent, "THE RECOMPUTATION MANIFESTO," *Available online: https://www.software.ac.uk/blog/2016-10-05-recomputation-manifesto Accessed on 12-11-2017*, p. 9479.

[40] A. M. Langer, "Guide to Software Development - Designing and Managing the Life Cycle," *Guid. to Softw. Dev.*, p. 402, 2016.

[41] H. J. Berkhout Frans, "Impacts of Information and Communication Technologies on Environmental Sustainability : speculations and evidence," vol. 5.

[42] L. M. Hilty and B. Aebischer, "ICT for Sustainability : An Emerging Research Field What Is Sustainability ?," *ICT Innov. Sustain. Adv. Intell. Syst. Comput. 310*, no. August, pp. 1–34, 2014.

[43] ETH Sustainability Summer School 2011, "Washing Machine," pp. 1–28, 2009.

[44] C. Reports, "Washing machines that save water and money," pp. 2015–2017, 2017.

[45] N. Geographic, "Washing Machines Buying Guide," pp. 4–5, 2017.

[46] T. Jack, "The dirt on clothes: Why washing less is more sustainable," *J. Home Econ. Inst. Aust.*, vol. 20, no. 2, p. 44, 2013.

[47] C. Ross, "The Damage I Cause When I Wash My Clothes," pp. 16–21, 2015.

[48] D. Markham, "How to reuse grey water in the home and," pp. 1–4, 2017.

[49] R. Entertainment, "UN honours ' Angry Birds , Happy Planet ' campaign," pp. 2016–2017, 2017.

[50] G. Deloittte, "2016 Mobile Industry Impact Report: Sustainable Development Goals," 2016.

[51] H. Arndt, B. Dziubaczyk, and M. Mokosch, "Information Technology in Environmental Engineering," pp. 13–24, 2014.

[52] S. Pictures, "MAKE THE ANGRY BIRDS HAPPY! TAKE CLIMATE ACTIO SEND A TWEET , RECORD A VINE , TAKE A PHOTO THAT SHOWS RAISING YOUR VOICE FOR GOOD CAUSE," *Available online: http://www.angrybirdshappyplanet.net/ Accessed on 13-10-2017*, p. 2017, 2017.

[53] S. Pictures, "THE ANGRY BIRDS MOVIE Themed " Angry Birds for a Happy Planet," *Available online: http://www.sonypicturesgreenerworld.com/articles/angry-birds-for-a-happy-planet Accessed on 14-10-2017*, p. 2017, 2017.

[54] Accenture, "Microsoft, Accenture and WSP Environment & Energy Study Shows Significant Energy and Carbon Emissions Reduction Potential from Cloud Computing," *Available online: https://newsroom.accenture.com/article_display.cfm?article_id=5089 Accessed on 20-11-2017*, pp. 1–2, 2017.

[55] S. Microsoft, "Microsoft products Our products are empowering people and organizations to achieve more while improving efficiency and reducing carbon emissions . Cloud Services," *Available online: https://www.microsoft.com/about/csr/environment/solutions/cloud/ Accessed on 11-11-2017*, pp. 3–5, 2017.

[56] A. Ishfaq and R. Sanjay, "Handbook of Energy-Aware and Green Computing," p. 2017, 2017.

[57] Accenture, "Cloud Computing and Sustainability: The Environmental Benefits of Moving to the Cloud," *Available online http//download.microsoft.com/download/A/F/F/AFFEB671-FA27-45CF-9373-0655247751CF/Cloud Comput. Sustain. - Whitepaper - Nov 2010.pdf Accessed 22-08-2017*, p. 16, 2010.

[58] Queensland Department of Public Works, *Smart and Sustainable Homes Design Objectives*. 2016.

[59] C. Wilson, T. Hargreaves, and R. Hauxwell-Baldwin, "Benefits and risks of smart home technologies," *Energy Policy*, vol. 103, no. September 2016, pp. 72–83, 2017.

[60] A. Bhati, M. Hansen, and C. M. Chan, "Energy conservation through smart homes in a smart city: A lesson for Singapore households," *Energy Policy*, vol. 104, no. September 2016, pp. 230–239, 2017.

[61] A. Monaghan, "Seven things you need to know about the UK economy," *Available online: https://www.theguardian.com/sustainable-business/2015/apr/17/things-need-know-sustainable-smart-technology Accessed on 18-12-2017*, 2014.

[62] V. K. Solanki, V. Muthusamy, and S. Katiyar, "Think Home : A Smart Home as Digital Ecosystem," no. June, pp. 1976–1991, 2010.

[63] R. Y. M. Li, "The usage of Automation System in Smart Home to provide a Sus- tainable Indoor Environment : A Content Analysis in Web 1 . 0," *Int. J. Smart Home*, vol. 7, no. 4, pp. 47–60, 2013.

[64] S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy, and J. Albrecht, "Smart*: An Open Data Set and Tools for Enabling Research in Sustainable Homes," *SustKDD*, no. August, p. 6, 2012.

[65] S. Makonin, L. Bartram, and F. Popowich, "a Smarter Smart Home," *Pervasive Comput.*, pp. 58–66, 2013.

[66] D. Spinellis, "the Social Responsibility of," no. December 2006, pp. 4–6, 2017.

[67] M. P. Alison Nicholl, "Smart home systems and the Code for Sustainable Homes: A BRE guide," *Proc. ICE - Eng. Sustain.*, pp. 1–1, 2012.