Validation Study of a Framework for Sustainable Software System Design and Development

Shola Oyedeji
LUT School of Engineering Science (LENS)
LUT University
Lappeenranta, Finland
shola.oyedeji@lut.fi

Mikhail .O. Adisa
IT Service Management Consultant
IT Solutions
Abuja, Nigeria
olamikhx@gmail.com

Birgit Penzenstadler
Department of Computer Engineering and Computer Science,
California State University Long Beach
Long Beach, USA
LUT School of Engineering (LENS)
LUT University
Lappeenranta, Finland
birgit.penzenstadler@csulb.edu

Annika Wolf
LUT School of Engineering Science (LENS)
LUT University
Lappeenranta, Finland
annika.Wolff@lut.fi

***Abstract*—Sustainability in software design is an evolving area that requires more practical guidance on how software engineers and businesses could innovate and design software systems that consider sustainability as a guiding principle for supporting a sustainable environment, reducing the negative impact of ICT and at the same time promoting software system design for sustainability. This paper presents our early results for validating a Framework for Sustainability of Software System Design (FSSSD) based on the Software Sustainability Design Catalogue (SSDC). The SSDC exemplifies the use of Karlskrona Manifesto principles for sustainability design and how to promote sustainability design principles for software systems.**

***Index Terms*—Sustainable design, sustainability, software sustainability, information and communication technology, Karlskrona manifesto, Sustainability design principles**

I. Introduction

Sustainability is receiving a wide range of research from different sectors. Currently, there is not enough research results with guidelines and frameworks to support software designers and companies on how to design and develop software with sustainability at the core [1]. One of the main problems for sustainability in software design is that for software designers there are few existing tools that wrap core principles of sustainability together which can support effective software sustainability design and development [2]. For companies, the challenge is that there is little understanding of how sustainability can be understood by software and requirements engineering professionals to facilitate sustainability design as an established part of the software development process within companies [3][4][5].

The sustainable development goals (SDGs) [6] in 2015 got signed by more than 190 world leaders, this shows the importance of sustainability today in all aspects of our lives. Though there is no direct mention of software sustainability in the 17 SDGs, software as a catalyst for all sectors of the economy [7] serves as a key element for the implementation and actualization of those SDGs. According to the 2016 mobile industry impact report [8], the United Nations Sustainable Development Goals provide the opportunity for engagement to address the most pressing global challenges, but they cannot be realized without the business community. The report stresses the need for companies to implement the SDGs, working with governments and the international community to expand connectivity, lower barriers to access, and build a future of dignity and opportunity, where no one is left behind and ensure that tools and applications are developed with vulnerable communities in mind [8].

Sustainable development is also driving software innovations for creating new opportunities of cutting costs, adding value and for gaining competitive advantage [9]. García-Berna et al. [10] points out the practices applied by practitioners in companies for sustainability and the need for standards as a way of seeking more sustainable software businesses. The importance of sustainability as a driving force for companies is further highlighted in these reports: Sustainability Nears a Tipping Point [11]; Ericsson energy and carbon report [12]; Microsoft 2015 Citizenship Report [13]. In summary, software is a core of all human activities today and a major facilitator in the way humans produce and use products and services [14]. The way software is designed and the requirements to ensure sustainability in software design are factors that are challenging for software designers, requirement engineers and companies [15].

The Karlskrona Manifesto for Sustainability Design (KMSD) [16] was initiated as a starting point for tackling this challenges in software engineering. Based on these KMSD

principles and the Software Sustainability Design Catalogue (SSDC) [1], the Framework for Sustainability of Software System Design (FSSSD) was created [1]. This paper presents the first results of applying the Framework for Sustainability of Software System Design (FSSSD) [1].

The next section covers related research work. Section III presents the study design. Section IV covers the first case study and section V details the second case study. Discussion is in section VI and concluding remarks in section VII.

II. Background

Software development practices and processes that are widely used in industry for software design and development lack in addressing sustainability [17]. There is currently no single point of reference for researchers and practitioners where the sustainability measures are gathered and exemplified [26]. The issue of lack of understanding on how to effectively and efficiently integrate the different sustainability dimensions (economic, social, individual, environmental and technical) [18] into software design, development and wider engineering processes [9] [19] has hindered the adoption of sustainability in software development.

There have been different research efforts suggesting the need to further research on how sustainability can be supported in software requirements and design stages for all the different sustainability dimensions [20] [21] [22]. Further research also shows sustainability requires multidimensional and interdisciplinary approach [3][7][23][24][25] in order to fully achieve sustainability in software design, development and measurement.

From the requirements engineering phase, sustainability has been considered as a non-functional requirement [26][27][28], and Roher et al. [29] suggests the use of sustainability requirement patterns (SRPs) as a way to guide software requirements engineers in eliciting sustainability requirements in the requirements engineering process. However, there is a lack of examples to show how these are applied in the industry.

Researchers from the Human Computer Interaction (HCI) community believe sustainable HCI can facilitate and support sustainability in the design and development of new interfaces to promote sustainability awareness [30]. Froehlich et al. [31] show eco feedback can serve as a key way of promoting sustainability awareness among users of software systems. One key example of an eco-feedback application [32] shows a positive result in persuading and changing users habit towards sustainability. Successful application of eco feedback is when information has been tailored to encourage users towards sustainability through user emotional engagement [33] [34].

Some of the design issues in design of sustainability for better user experience of software systems are highlighted by Kem-Laurin [35]. Kem-Laurin propose the use of sustainability user experience framework as a way to guide designers to mitigate these problems. The challenge according to Eli Blevis [36] and Fallman [37] is that sustainability is not yet a core part of HCI. This has hindered the ability of designers to properly evaluate design choices for software systems especially with the different sustainability dimensions.

The challenges covered in this background section motivate the application of FSSSD to two case studies in order to show and suggest how to better support sustainability in software design and development.

III. Study Design for Framework Validation

This section describes the Framework for Sustainability of Software System Design (FSSSD) and the rationale behind choosing the two case studies used in the research.

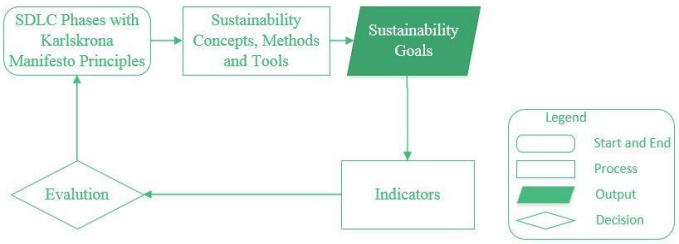

Figure 1. Framework for Sustainability of Software System Design (FSSSD) [1]

The FSSSD (Figure 1) was created to assist developers to incorporate sustainability goals and requirements during software system design and development covering the software development life-cycle (SDLC) phases. For the purpose of better understanding, the FSSSD (Figure 1) is transformed into tabular form (Table 1) [1].

TABLE I. Framework for Sustainability of Software System Design (FSSSD) [1]

SDLC phases and KMSD principles	**Sustainability goals**	**Sustainability concepts, Methods and Tools**	**Indicators**
Phase 1. Project Definition, P1, P2 and P3	Design for sustainable efficiency, reusability	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.	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	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, sustainability awareness, efficiency, design for easy	Biomimicry, cradle to cradle	Number of coding choices influenced by sustainability, number of features (functions) added to systems to inform users about sustainability through

	service and maintenance		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 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, possible second-order (enabling) and potential third order (structural) impacts of the 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 improvements to system based on sustainability requirements either from users' feedback or developers.

The approach applied in the selection of each case study was to choose two different case studies where one case study has the ultimate goal of sustainability from the beginning and the other case study uses the framework to improve an existing system.

The goal is to see what difference will occur from these two different case studies in different application context. The first case study - about a pension benefit tracker application - does not have sustainability as the central core and the second case study - about an energy usage display for university staff and students - is motivated by sustainability.

IV. Case Study One: Pension Benefit Tracker Application

The pension benefit tracker is an application from a pension company in Nigeria that wants to track pension benefit applications submitted by clients from all over the company's branches in different states of Nigeria. Currently, the pension applications are done manually from each branch and those applications are sent via courier service to the head office. This usually causes the following problems:

1. Zonal managers don't have direct access to know the status of applications submitted through them and have to directly place phone calls to the Head office to know the application status.
2. Customer service staff are unable to know why an application is pending, unless they contact the benefit department.
3. Time consumption, as all status updates are through customer service at the head office alone.
4. Files can go missing in transit because application files are handled manually.
5. Double application and too much physical involvement because of follow up in person

The company intended to develop a new pension benefit application tracker application for these key stakeholders, the benefit department, the customer service unit, the zonal managers and the clients with the aim of:

1. Identifying ways of improving the pension benefit application process and enhance communication.
2. Designing and implementing a web-based solution that will ensure effective and efficient benefit processing for users.

The below Figure 2 is the first Use case diagram for the application.

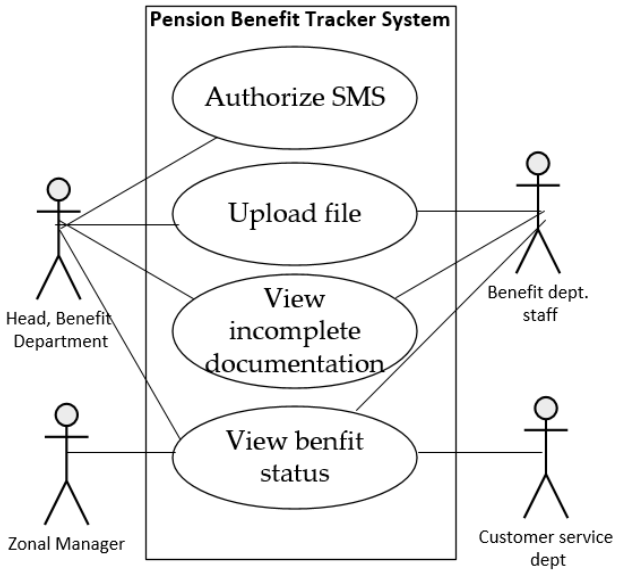

Fig 2. Use Case diagram pension benefit tracker

Figure 2 shows the use case diagram of the system for pension benefit tracker application after initial analysis. Figure 3 presents the process model of the pension benefit application after a second analysis, factoring in all the aforementioned problems without using FSSSD. Figure 3 shows that sustainability was not the core of this case study, based on the process model, as stakeholders are just interested in solving the problems stated in the case study.

Table 2 presents the details for applying FSSSD to the pension benefit tracker application (case study one). The documentation

for this case study using FSSSD covers the project initiation, user requirements and system requirements phases only (see Table 2) because that is the current development stage of the project.

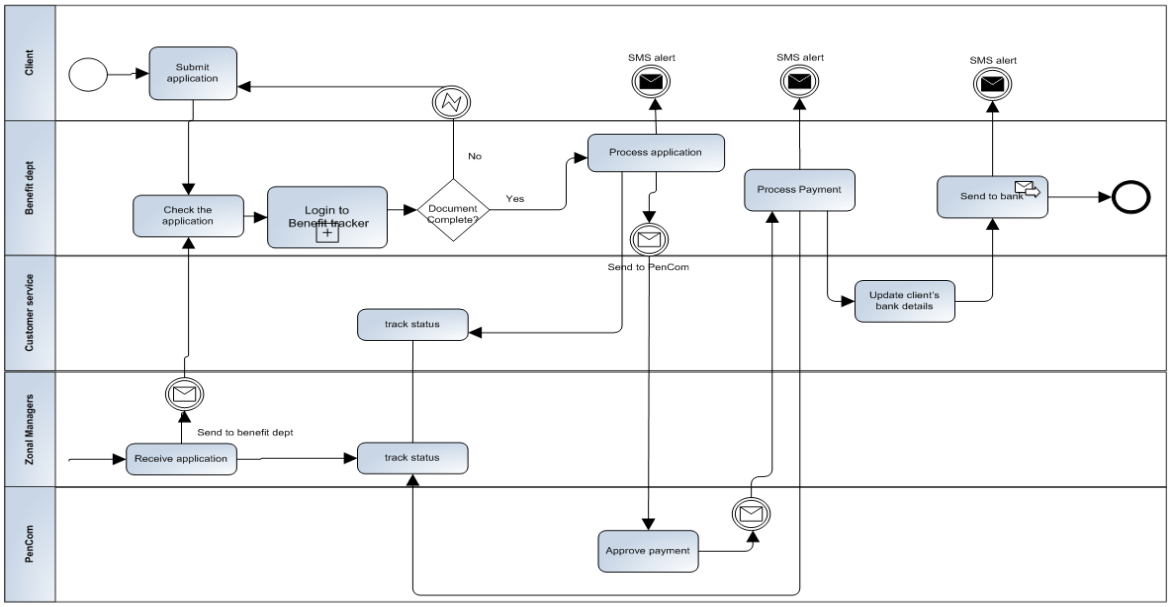

Fig 3. New Process Model for Pension Application after second analysis

TABLE II. APPLICATION OF FSSSD IN CASE STUDY ONE

SDLC Phases and Karlskrona Manifesto Principles	**Sustainability Goals**	**Sustainability Concepts, Methods and Tools**	**Indicators /Measure / Metric**
Phase 1. Project Definition Provide end users with easy to use interface for tracking pension payment, ensure each module for tracking can be updated to include new branches, Provide flexibility such as bulk and single upload, ensure easy integration with other existing pension systems, present report of system usage to track energy consumption in a way to educate users about sustainability, add bug reports	Design for: Easy integration, Reusability, Developers work satisfaction, Maintainability, Energy efficiency	Motivated by the cradle to cradle approach ensuring that the pension tracker application is design and developed in a way that it can be reused for future pension related purposes and easily integrated with other bigger pension system within the company	1. How many state branches can easily integrate the systems with less Backlog Management Index (BMI)? 2. What is the number of reports from IT staff about how to improve system energy efficiency? 3. How satisfied are the developers with the development of the application
Phase 2. User Requirements Definition 1. Provide tracking of pension benefit payment application from request submission to payment 2. Status notification should be sent to users after each stage of the pension benefit application	Reduce development cost, increase efficiency	Sustainability requirement Template	How efficient is benefit department able to track new pension benefit applications and send notification successfully
Phase 3. System Requirements Definition 1. The pension tracker application should be accessible online via web at any branch 2. The application should have ability to enable Managers, pensioners and other stakeholders check application status 3. Provide automatic status communication and notification at each stage of benefit application 4. Allow bulk or single file upload 5. Provide SMS authorization from managers in benefit department 6. Send SMS notification to applicants 7. Send Incomplete documentation notification to	Design for efficiency, sustainability awareness	Social and individual dimension of sustainability	1. How satisfied are users with visual problem with the magnifying display? 2. Do users use the option of email notification and does it reduce company cost for sending SMS? 3. How many positive responses came from users base on the "Save the planet, Reduce environmental waste" tag message?

benefit department staff 8. Provide email notification as an option for all users 9. Provide option of different display to magnify fonts for users with visual problems 10. Provide option to preview pension application and save electronically 11. Add a tag message below each notification "Save the planet, Reduce environmental waste" 12. Provide energy report for system usage			4. How many initiatives were suggested from IT department base on the system energy report?

After application of the FSSSD with the sustainability design catalogue (SSDC), see Table 2, the IT department made some changes to the system requirements such as addition of the following system requirements in Table 2, SDLC phase 3:

1. Email notification option instead of only SMS function as seen in Figure 3 in which only SMS is shown (system requirement 8 in Table 2).
2. Provide option of different display to magnify fonts for users with visual problems especially older staff (system requirement 9 in Table 2).
3. Provide option to preview pension application and save electronically instead of printing and filling locally to reduce cost, paper waste and energy usage (system requirement 10 in Table 2)
4. Add a tag message below each notification "Save the planet, and reduce environmental waste" to raise sustainability awareness among staff and clients (system requirement 11 in Table 2).
5. An energy report that enables developers to improve efficiency (system requirement 12 in Table 2).

V. Case Study Two: Energy Usage and Carbon Emission Display for University Staff and Students

This is a university setting project to raise the awareness of the public (university staff and students) about energy usage and the carbon emissions through activities in the university. The project requires a web application interface which will display the energy usage and carbon emission. The goal is to let the public know more about the electricity consumption of each building in the university and understand the relation between the electricity consumption and carbon emission (CO2).

Using the FSSSD, the involved students and their supervisors documented the project to show how sustainability was considered in the project (see Table 3). Figure 4 shows the interface design for the project and Figure 5 covers an overview of the sustainability business canvas for the project.

TABLE III. FSSSD application in case study two (Energy usage and carbon emission display for staff and students)

SDLC Phases and Karlskrona Manifesto Principles	**Sustainability Goals**	**Sustainability Concepts, Methods and Tools**	**Indicators /Measure**
Phase 1. Project Definition Raise awareness from the public (university staff and students) about energy usage and the carbon emissions through activities in the university.	Design for sustainability awareness, efficiency, reusability, easy integration, maintainability and energy efficiency	Sustainable Business Canvas was used to breakdown the project goals and scope into environment, society, economy, process, value and people in order to have better clarity on the sustainability goals of the project and derive basic benchmarks for evaluating the project at the end.	1. What is the impact of the project on promoting sustainability awareness within the university? 2. How many users participate in the weekly sustainability challenge? 3. What are the new initiatives from departments towards sustainability based on the application usage?
Phase 2. User Requirements Definition 1. Provide information on energy usage within the university 2. Show the carbon emission 3. Allow weekly sustainability challenge and show winners 4. Section for user community to connect and discuss 5. Provide feature to share things to social media	Increase sustainability awareness through energy usage and carbon emission information to users	Sustainability requirement template (template that shows the sustainability analysis of the five dimensions and the three orders of effects from the design catalogue) [1]	1. Can users see information about energy usage and carbon emission? 2. How effective is the weekly sustainability challenge? 3. How many users participate in the weekly sustainability challenge? 4. Do users share their experience via social media portal?
Phase 3. System Requirements Definition 1. Information about energy usage and carbon emission should be available via the central display screen and web portal	Design for sustainability awareness, maintainability and energy efficiency	Environmental, Social and individual dimension of sustainability	1. Can users understand the energy and carbon emission information presented? 2. How easy can users join the

2. The application should translate the carbon emission data base on energy usage into meaningful information for better user understanding such as distance between Lappeenranta and other cities 3. The web interface should allow users participate in the weekly challenge 4. Users are able to share their weekly challenge results via Facebook and Twitter. 5. The application should allow users form community of interest for different sustainability goals. 6. Provide API to allow for easy integration with other applications			weekly challenge? 3. Does the application to form community of different sustainability goals? 4. Can users successfully share their weekly challenge on Facebook and Twitter? 5. Does the API allow easy information access?
Phase 4. Analysis and Design 1. Identify the first, second and third order impact of the application on user energy usage and sustainability awareness 2. Find areas to improve the application implementation base on the different sustainability dimensions especially environment, social and technical dimensions	Design for sustainability awareness, reuse, efficiency and localization	Sustainability analysis radar chart was used for the sustainability analysis to show the he first, second and third (immediate, enabling, and structural) impacts of the application.	1. What is the potential percentage of energy usage reduction in the university? 2. What is the level of user awareness overtime about energy usage and carbon emission? 3. What is the impact of the user community for users' motivation towards sustainability within the university?
Phase 5. Development	Design for sustainability awareness, efficiency, reuse, design for module replicability, design for easy service and maintenance	Cradle to cradle concept influence the development to develop each module in the application in a way that support evolution as user requirements changes over time and ensuring sustainability is the core of all development	1. What is the defect density of the application? 2. What is the energy efficiency of the application? 3. How many modules relating to sustainability awareness was successfully developed? 4. Can users successfully use the application for all application functions such as join a community, participate and share weekly sustainability results, understand displayed energy usage and carbon emission information?

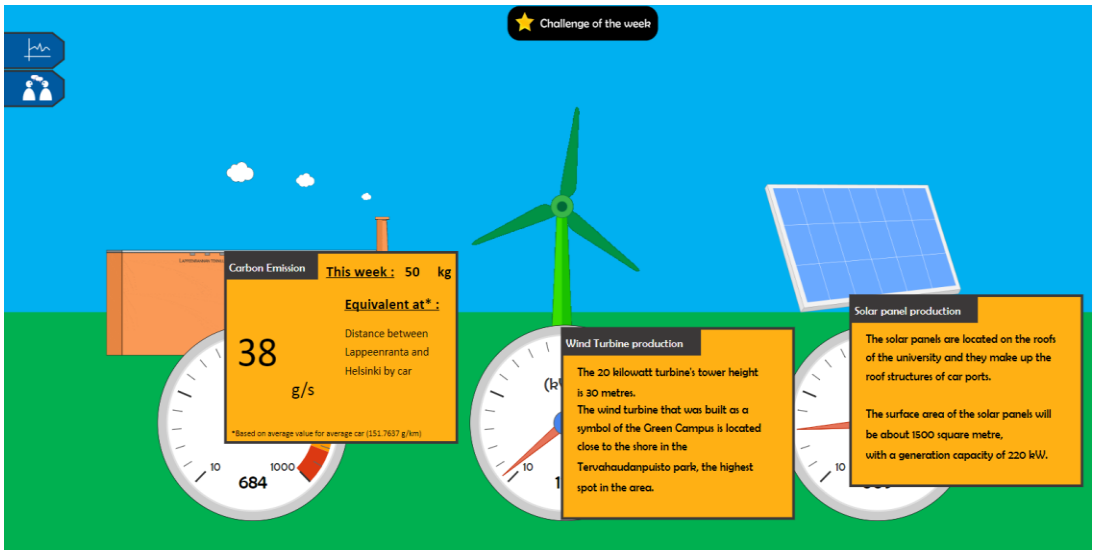

Fig 4.Sustainability awareness via energy usage interface

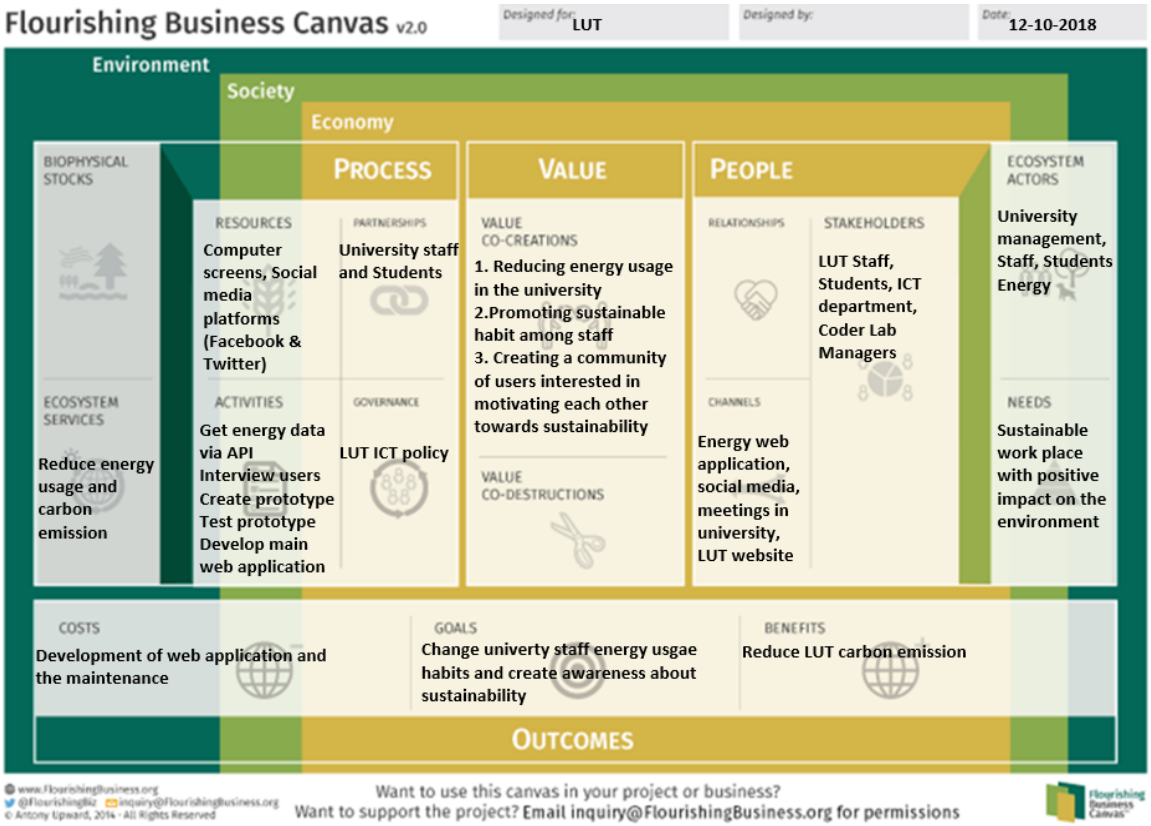

Fig 5. Sustainable Business Canvas for Case Study Two [38]

VI. DISCUSSION

For the project initiation in the first case study, normally project managers will only evaluate projects by considering whether the software system meets all user requirements after development and testing as a yardstick for satisfying all project requirements. The application of FSSSD in case study one (Table 2) shows that indicators used for evaluating the project up to the current development stage included the level of developer satisfaction (individual dimension of sustainability) and the number of IT staff reporting on how to improve the system energy efficiency (environment and technical dimension). This confirms a new perspective towards software project evaluation with sustainability dimensions now considered by stakeholders in case study one. The use of FSSSD also led to new system requirements (Table 2) with the potential to improve the system efficiency and consideration of sustainability based on the system context.

Based on the initial response from stakeholders in case study one, it indicates that as a company their major interest was to check if FSSSD - as guide in the application of sustainability in software system design and development - would save them cost and improve staff productivity. The use of developers satisfaction for the pension benefit tracker is one example because the company believes if there is means of checking staff satisfaction, it could offer a means of improving working conditions which will in turn improve productivity over time. This will help them reduce the cost of operations and improve profit margin.

Case study two provides a different use of FSSSD as sustainability is the core of the application design. As noted in [33] [34], with better tailored information through eco feedback, user habits can change positively towards sustainability over time. The second case study (see Table 3, Figures 4 and 5) shows the presentation of energy usage data converted into carbon emission. With the use of FSSSD as guide, the application in case study two was designed in a way that the carbon emission information was displayed in order to educate users about their energy consumption habits in each department. The system presented the percentage of carbon emission in form of distance between one city to another with the goal to provide better understanding for the public about the impact of their energy consumption on the environment.

Feedback and comments (Table 4) from stakeholders in case study one and two indicates that developers and engineers complained there are few industry case studies for software development that shows how sustainability was applied. The second challenge was in motivating software requirements engineers and designers to incorporate the use of the new sustainability artifacts for sustainability in requirements and software development because most of them are used to the old ways of developing software systems and therefore require extensive discussion on the usage of the artifacts in FSSSD.

In general, the early feedback and comments (Table 4) from case study one and two shows that the Framework for Sustainability of Software System Design (FSSSD) provides guidance and support for sustainability in software design requirements and development. The tools, methods and concepts provided as

sample in the framework helped in providing new insights into how sustainability can be incorporated into software project design and development especially the Sustainable Business Canvas, Goal model, Sustainability Requirement Template, Biomimicry, Cradle to cradle concept and Sustainability Analysis Radar Chat diagram. In addition, FSSSD also persuades stakeholders to rethink their software project with sustainability as a means of developing a better product that is cost effective over a long time and supports good corporate social responsibility. Table 4 summarizes the feedback on the usage of FSSSD from the case studies.

TABLE IV. DIRECT QUOTES FEEDBACKS AND COMMENTS FROM PARTICIPANTS AND STAKEHOLDERS IN USING FSSSD (CASE STUDY ONE AND TWO)

Role	**SDLC Phase**	**Positive**	**Challenges**
CTO	Project Definition	1. The SSDC was good way to understand the different aspect of sustainability for different kind of software system. The SSDC made it possible for me and my team to know more about sustainability in software development with those guidelines provided for each software system. 2. The FSSSD provides new insight for sustainability in software project with consideration of sustainability principles 3. Combination of the SSDC and FSSSD provides an avenue to consider our software impacts and see how we can minimise it. 4. FSSSD introduces new methods for evaluating our applications especially the environmental and individual dimensions of sustainability 5. The Sustainable Business Canvas brings in a totally new factors into software project definition with sustainability concepts and dimensions as guide	1. Very difficult to understand how to apply some of the sustainability concepts because its new to me and my team 2. We have a challenge to find concrete examples online to see how sustainability was applied to software project definition especially in industry 3. It was challenging to give my staff additional task of reading the Framework manual to understand how to apply it
Software developer, Project coordinator	User requirement definition	1. The sustainability requirement template was useful as guide during requirement gathering because it provides us with means of discussing sustainability with users and categorising user requirements base on sustainability dimensions	It was difficult at first to understand how to explain the different dimensions of sustainability to key stakeholders (users) during discussion gathering requirements on how to improve the existing system
System analyst, software developer	System Requirements Definition	1. I was able to learn new things about how sustainability can influence gathering system requirements and identifying new system requirements using the FSSSD 2. The goal model diagram is really a good tool to breakdown sustainability goals base on requirements into business, usage and system goal. 3. The goal model diagram made it easy to explain, discuss and improve the project goals and system requirements using the business, usage and system goal diagram.	1. The only issue is lack of examples to show how sustainability has been used in different software requirements elicitation at the beginning when using FSSSD but after couple of meetings discussing about sustainability with the research guy things became clearer. 2. Some of the research especially about sustainability in system requirements I saw on google from some researchers are too complex to apply
System analyst, Programmers, Software developer	Analysis and Design	1. The sustainability goals and suggested tools from FSSSD was a good starting point to guide us during the analysis and design phase. 2. The sustainability analysis radar chat was a new interesting tool because it shows some new requirements to add after brainstorming on each of the first, second and third impacts	Brainstorming on how to connect the first, second and third order impact in each of the sustainability dimensions was not easy because each of us have different views on what is the right thing to put but eventually we looked at some of the examples provided by the researcher guy in using FSSSD.

VII. CONCLUSION

Software design and development in the real world is continuously changing with the adoption of new software development methods and paradigms, such as agile, to reduce the development time from different SDLC phases and shortened time to market. However, sustainability is currently not at the core of the general development methodology in companies. Sustainability as a main principle and value provides a competitive advantage for companies and software designers /developers but the major challenge is the lack of understanding on how to institutionalize sustainability in software design and development projects.

This paper summarizes our early results on applying the Framework for the Sustainability of Software System Design (FSSSD) (Figure 1 and Table 1) in two case studies. The FSSSD provides support for sustainability in software design through the aspect of promoting sustainability goals at each stage of a software development life cycle phase with aid from different sustainability concepts, tools and methods as seen in case study 1 (Table 2) and case study 2 (Table 3 and Figure 4, 5). It also encourages a sustainability-oriented software development mindset over time with usage of FSSSD, because sustainability becomes part of the core fundamental values for software design and development practice.

Discussions with stakeholders and feedback in each of the case studies (Table 4) shows the major challenge in application of sustainability to software design and development is the lack of readily available software system industry examples and best practices of how core principles of sustainability are applied and exemplified in software projects.

Another challenge is in shifting developers' mindsets to adopting sustainability in a way that translates into their software design and development decisions and practices. The concept of sustainability dimensions (social, individual, environmental, economic, and technical) only becomes interesting to apply in software design if it can provide companies with opportunities for cutting costs and offer a competitive advantage in one way or another through usage of the framework.

The next phase is to repeatedly apply the FSSSD to different kinds of software projects and record best practices from each of these projects that can then be disseminated to interested stakeholders. Our template for documenting software sustainability requirement elicitation best practice during software design and development [39] can serve as template for such documentation.

ACKNOWLEDGMENT

The authors would like to thank Tom Mistretta and Alexandre Devinez for their work in the Living Lab at LUT University to support this research work.

REFERENCES

[1] S. Oyedeji, A. Seffah, and B. Penzenstadler, "A catalogue supporting software sustainability design," *Sustainability*, vol. 10, no. 7, pp. 1–30, 2018.

[2] N. Shedroff, *Design is the Problem:The Future of Design Must be Sustainable*. Rosenfeld Media, 2009.

[3] R. Chitchyan, L. Duboc, C. Becker, S. Betz, B. Penzenstadler, and C. C. Venters, "Sustainability Design in Requirements Engineering : State of Practice," pp. 533–542, 2016.

[4] M. Mahaux and C. Canon, "Integrating the Complexity of Sustainability in Requirements Engineering," *First Int. Work. Requir. Eng. Sustain. Syst.*, 2012.

[5] U. K. Jannat, "Green Software Engineering Adaption In Requirement Elicitation Process," vol. 5, no. 08, pp. 94–98, 2016.

[6] United Nations, "Sustainable Development Goals Available at: https://www.un.org/sustainabledevelopment/sustainable-development-goals/ Accessed on 28-12-2018," no. September 2000, pp. 8–23, 2015.

[7] S. Oyedeji, A. Seffah, and B. Penzenstadler, "Classifying the Measures of Software Sustainability," in *Proceedings of the 4th International Workshop on Measurement and Metrics for Green and Sustainable Software Systems co-located with 12th International Symposium on Empirical Software Engineering and Measurement (ESEM 2018)*, 2018.

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

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

[10] J. García-Berna, J. Carrillo de Gea, B. Moros, J. Fernández-Alemán, J. Nicolás, and A. Toval, "Surveying the Environmental and Technical Dimensions of Sustainability in Software Development Companies," *Appl. Sci.*, vol. 8, no. 11, p. 2312, 2018.

[11] H. Knut, R. Martin, S. Ingridvon, A. Michael, K. David, and K. Nina, "Sustainability Nears a Tipping Point Sustainability Nears a Tipping Point," *MIT Sloan Manag. Rev.*, vol. 53, no. 2, pp. 69–74, 2012.

[12] Ericsson, "Energy and Carbon Report," no. June, p. 12, 2013.

[13] Microsoft, "Microsoft 2015 Citizenship Report," 2015.

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

[15] 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.

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

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

[18] 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.

[19] 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.

[20] M. Al Hinai and R. Chitchyan, "Engineering Requirements for Social Sustainability," *Proc. ICT Sustain. 2016*, 2016.

[21] B. Penzenstadler, "Software Engineering for Sustainability."

[22] E. Blevis, C. Preist, D. Schien, and P. Ho, "Further Connecting Sustainable Interaction Design with Sustainable Digital Infrastructure Design," *Proc. 2017 Work. Comput. Within Limits - LIMITS '17*, pp. 71–83, 2017.

[23] G. G. Calienes, "Requirements Prioritization Framework for

developing Green and Sustainable Software using ANP - based Decision Making," pp. 1–9, 2013.

[24] B. Penzenstadler, "RE4ES: Support Environmental Sustainability by Requirements Engineering," *First Int. Work. Requir. Eng. Sustain. Syst.*, 2012.

[25] P. Bozzelli, Q. Gu, and P. Lago, "A systematic literature review on green software metrics," *Sis.Uta.Fi*, 2013.

[26] 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.

[27] C. C. Venters *et al.*, "The Blind Men and the Elephant Towards an Empirical Evaluation Framework for Software Sustainability," vol. 2, no. 1, pp. 1–6, 2014.

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

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

[30] L. Bonanni, D. Busse, J. Thomas, E. Blevis, M. Turpeinen, and N. J. Nunes, "Visible - Actionable - Sustainable: Sustainable Interaction Design in Professional Domains," *Proc. CHI 2011*, pp. 2413–2416, 2011.

[31] J. Froehlich, L. Findlater, J. Landay, and C. Science, "The Design of Eco-Feedback Technology," pp. 1999–2008, 2010.

[32] A. Spagnolli *et al.*, "Eco-feedback on the go: Motivating energy awareness," *Computer (Long. Beach. Calif).*, vol. 44, no. 5, pp. 38–45, 2011.

[33] R. J. Yun, A. Aziz, and B. Lasternas, "Design Implications for the Presentation of Eco- feedback Data," *Arch. Des. Res.*, vol. 28, no. 4, pp. 95–106, 2015.

[34] P. Kotzé, G. Marsden, G. Lindgaard, J. Wesson, and M. Winckler, *Interaction – INTERACT 2013 14th IFIP TC 13 International Conference*, no. 1. 2013.

[35] K.-L. Kramer, *User Experience in the Age of Sustainability. Available at: http://dx.doi.org/10.1016/b978-0-12-387795-6.00001-9*. 2012.

[36] E. Blevis, "Sustainable Interaction Design : Invention & Disposal , Renewal & Reuse," pp. 503–512, 2007.

[37] D. Fallman, "Design-oriented Human — Computer Interaction," no. 5, pp. 225–232, 2003.

[38] "Strongly Sustainable Business Model Canvas Http://www.ssbmg.com/ Accessed on 18-04-2019."

[39] S. Oyedeji and B. Penzenstadler, "Karlskrona Manifesto: Software requirement engineering good practices," *CEUR Workshop Proc.*, vol. 2223, pp. 15–23, 2018.