Integrating the Sustainability Awareness Framework in Undergraduate Software Engineering Education

Peter Bambazek[1,*,†], Iris Groher[1,†]

[1]Johannes Kepler University Linz (JKU), Linz, Austria

Abstract

[Context and motivation] In recent years, sustainability has emerged as an important concern in requirements engineering. The Sustainability Awareness Framework, the leading approach in this area, offers practical support for practitioners to identify the potential sustainability impacts of software systems. [Question/problem] Despite its relevance, a critical question remains unanswered: how can we effectively enhance sustainability awareness among students and integrate it into software engineering education? [Principal ideas/results] In this paper, we present early results of an ongoing undergraduate practical software engineering course in which we integrated the Sustainability Awareness Framework. We report on how students performed in identifying sustainability effects of a new software system for a hypothetical company. [Contribution] Our primary contribution is a proposal on how the Sustainability Awareness Framework can be integrated in university-level courses. Moreover, our findings reveal that students, even without prior related knowledge, can identify significant sustainability aspects, supporting the framework's educational value and accessibility.

Keywords

Sustainability, Requirements Engineering, Sustainability Awareness Framework, Higher Education

1. Introduction

In recent years, several approaches have been proposed in the field of requirements engineering that promote the development of sustainable software systems [1, 2, 3, 4]. Among these, the Sustainability Awareness Framework (SusAF, cf. Figure 1) [5, 6] has emerged as a prominent approach, as highlighted in a previously conducted mapping study [7].

SusAF is distinguished by its practical orientation, providing a structured guide for facilitating workshops with stakeholders, including a set of instructions and questions to guide in-depth discussions on the sustainability implications of software systems. The primary goal of the framework is to raise awareness of the potential effects of software systems, which is also in line with the principle of *"Good Digital Design is sustainable and creates sustainability"* as stated by IREB [9].

In: D. Mendez, A. Moreira, J. Horkoff, T. Weyer, M. Daneva, M. Unterkalmsteiner, S. Bühne, J. Hehn, B. Penzenstadler, N. Condori-Fernández, O. Dieste, R. Guizzardi, K. M. Habibullah, A. Perini, A. Susi, S. Abualhaija, C. Arora, D. Dell'Anna, A. Ferrari, S. Ghanavati, F. Dalpiaz, J. Steghöfer, A. Rachmann, J. Gulden, A. Müller, M. Beck, D. Birkmeier, A. Herrmann, P. Mennig, K. Schneider. Joint Proceedings of REFSQ-2024 Workshops, Doctoral Symposium, Posters & Tools Track, and Education and Training Track. Co-located with REFSQ 2024. Winterthur, Switzerland, April 8, 2024.

*Corresponding author.

†These authors contributed equally.

peter.bambazek@gmail.com (P. Bambazek); iris.groher@jku.at (I. Groher)

0000-0002-4447-3083 (P. Bambazek); 0000-0003-0905-6791 (I. Groher)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

The Sustainability Awareness Framework (SusAF) presents a guideline to carry out workshops with stakeholders, and a set of instructions and questions to guide discussions on the sustainability effects of software systems.

Sustainability Dimensions. The guidance questions to identify potential effects cover the following sustainability dimensions and related topics:

- Social: e.g. sense of community, trust, diversity, equity, communication
- Individual: e.g. health, lifelong learning, privacy, safety, agency
- Environmental: e.g. resources, pollution, energy, biodiversity, land use
- Economic: e.g. value, customer relationship, supply chain, governance, innovation
- Technical: e.g. maintainability, usability, adaptability, security, scalability

Orders of Effects. Each dimension is further divided into three orders of effects to indicate the degree of each identified effect:

- Immediate: caused by the direct function of the system or its development
- Enabling: arising from the application of a system over time
- Structural: referring to persistent changes that can be observed at the macro level

Figure 1: Overview of the Sustainability Awareness Framework [5, 8]

So far, SusAF has been evaluated in educational contexts and in several industrial case studies [10, 11, 12]. To date, over 200 students have engaged with SusAF [12]. Penzenstadler et al. assessed the sustainability analysis diagram, a component of SusAF, with 20 students across five teams, receiving positive feedback on its suitability and usefulness [13]. Duboc et al. explored SusAF's feasibility with 47 computer science students, demonstrating its effectiveness in fostering discussions on sustainability and its accessibility to students without prior knowledge [5]. Furthermore, Penzenstadler et al. reported on SusAF's application during a summer school in Uganda, involving 30 participants. This evaluation confirmed SusAF's simplicity and effectiveness in raising awareness about the sustainability impacts of software systems within an educational setting [14]. A comprehensive and recent literature review on sustainability in computing education was conducted by Peters et al. [15] in 2024. Their work examines the integration and exploration of sustainability concepts in computing education, focusing on different perspectives and applications related to sustainability, computing, and educational methodologies.

However, our research is different from existing studies in its focus, as it is the first attempt to embed SusAF comprehensively within a software engineering course structure. We present our approach to incorporating SusAF into a Business Informatics Bachelor's course at the Johannes Kepler University Linz (JKU). This includes an outline of the implementation strategies and pedagogical methods employed. Additionally, we present preliminary findings from this study, with a particular focus on the effectiveness of SusAF in enhancing student awareness regarding the potential sustainability impacts of software systems.

The remainder of this paper is structured as follows. Section 2 presents the setting of the course. The findings from our ongoing research are detailed in Section 3, followed by an in-

depth discussion in Section 4 and a summary of identified limitations in Section 5. The paper concludes with Section 6, where we summarize our key insights and propose directions for future research.

2. Course Setting

The course, mandatory in the fourth semester of the Bachelor's program in Business Informatics, carries 6 ECTS and focuses on practical aspects of software engineering. Students are grouped into teams of three, working collaboratively on a hypothetical software application project.

This semester, the project's objective was to develop a digital logbook application for managing trips. The application's core functionalities include the ability to create, edit, and delete trip entries. Additionally, it includes features for sorting, filtering, and analyzing the list of trips. Moreover, the application is designed to support both local and cloud-based backup options for trip data. In the current semester, 15 students grouped into 5 teams attended the course.

The developmental process was divided into three sprints, encompassing a total of 20 user stories. This structure not only provided a realistic software development experience but also facilitated the application of agile methodologies in a controlled educational setting. The course's design aims to equip students with practical skills in software development, team collaboration, and agile project management. During the sprints, students concentrate on various key areas including object-oriented design, implementation, testing methodologies, user interface design, analysis of code quality, and comprehensive documentation.

The course timeline and the associated tasks focusing on sustainability are detailed in Table 1. This paper specifically discusses the outcomes of the first three sustainability tasks (T 1-3).

Table 1
Timeline of the Practical Software Engineering Course

ID	Tasks	Oct.	Nov.	Dec.	Jan.	Feb.
T1	Sustainability Literature					
T2	SusAF Workshop					
T3	Survey #1					
T4	Analysis of User Stories					
T5	Survey #2					
T6	Design					
T7	Development and Test					
T8	Quality and Documentation					

In the initial task (T1), we equipped students with a selection of relevant literature [5, 16, 8, 17] to introduce them to the topic of sustainability. This step aimed to provide a theoretical background for their practical work. Subsequently, for T2, students were presented with a comprehensive scenario involving a fictional company, a detailed description of the proposed software solution, and a suite of 20 user stories. Utilizing this information, they were tasked with conducting a SusAF workshop. The primary objective here was to identify potential sustainability impacts associated with the software system. Following this, an online survey (T3) was conducted, aimed at gathering feedback on the application of SusAF and to acquire

additional insights. In T4, we directed the students to further analyze the provided user stories. Their task was to document how these stories either aligned with the previously identified sustainability effects or unveiled new sustainability aspects. Thereafter, in T5, a second online survey was conducted to evaluate how students' perceptions of sustainability evolved over the course of the semester. Tasks T6 through T8, which are associated with the development process of the software system, were conducted concurrently with the sustainability-oriented tasks T1 to T5. In the first month (T6), students focused on creating a basic design for their software, encompassing architecture, UI/UX considerations, etc. The main portion of the semester (T7) was then devoted to the actual software development. Finally, T8 required students to concentrate on code quality analysis and the documentation of their software project at the semester's end.

3. Results

In the conducted SusAF workshops, the five teams together identified 97 sustainability effects (including duplicates). On average, each team recognized approximately 19.4 effects, with a variation ranging from a minimum of 16 to a maximum of 24 effects across the teams. The majority of these effects, accounting for nearly two-thirds (60 effects), were classified as positive. Approximately one-third (34 effects) were deemed negative. Additionally, there were three effects for which the impact was determined to be dependent on user behavior, potentially being either positive or negative. An aggregation of the identified effects, noting that several effects were recognized by more than one team, reveals a total of 52 different effects: economic (11), environmental (8), social (7), individual (12), and technical (14).

The voting scale of the survey was defined from 1 (strongly disagree) to 5 (strongly agree). Table 2 illustrates the outcomes of specific questions or statements from the initial online survey that students completed following their SusAF workshops. The survey employed a rating scale ranging from 1, signifying strong disagreement, to 5, indicating strong agreement.

Table 2
Example questions/statements of first survey

Question/Statement	Median	Modal	(Min,Max)
The division of sustainability into 5 dimensions is understandable.	4,00	5,00	(1,5)
We identified more potential impacts of the software system across the various sustainability dimensions than I expected.	4,00	5,00	(1,5)
Discussing potential sustainability aspects of the software system in the team helped to better understand the product vision of the system.	4,00	4,00	(1,5)
Potential impacts on the sustainability of a software system should be taken into account as early as possible in software development.	5,00	5,00	(3,5)
Discussing sustainability aspects of software systems corresponds to my expectations of studying business informatics.	4,00	4,00	(3,5)

"The sustainability tasks provided me with a novel perspective on software projects, which I didn't have before."

"Initially, my understanding of sustainability was limited to its environmental aspect. However, learning about the existence of additional dimensions and the impact a system can have on these dimensions was interesting."

"Through this course I learned that sustainability is more than just ecological concerns and I will try to integrate this understanding into my daily work."

"It became increasingly fascinating to identify the links between sustainability dimensions and our user stories. This offered not only theoretical insights, but also positively influenced the technical development of our project. Taking sustainability aspects into account helped us to focus our work more precisely also with an awareness of their long-term effects."

Figure 2: Statements from course participants

Detailed results including the course documents can be accessed in the corresponding online repository[1].

Throughout the course, we gathered student feedback and received exclusively positive reactions from seven participants regarding the sustainability tasks. As illustrated in Figure 2, the student comments highlight not only an enhanced understanding of the different aspects of sustainability but also recognize the benefits of these insights beyond the course.

4. Discussion

The most noteworthy aspect is the comprehensive outcomes provided by the student teams. Each team identified potential sustainability effects across all dimensions. The results highlight a diverse set of sustainability issues and illustrate how the adoption of a new software system can impact a company's overarching sustainability objectives. Notably, the students reported a predominance of positive effects over negative ones. This trend aligns with findings from a prior case study we conducted involving practitioners, as detailed in [10].

Secondly, it's important to note that the student teams engaged with the SusAF workbook without significant prior knowledge. We made relevant literature available for self-study but did not offer any explicit introduction regarding the definition of sustainability, its five dimensions, or the specifics of SusAF. In the initial survey, a majority of the students reported lacking professional experience in software engineering and having not previously addressed sustainability aspects within this field. Given this context, the results achieved through the application of SusAF are particularly encouraging.

Thirdly, we received very positive feedback from the students regarding the implementation of sustainability in the software project course. As depicted in Table 2, the students acknowledged

[1] https://github.com/peter1123581321/jku-win-se-2023, created and last accessed March 2024

that the integration of sustainability discussions aligns well with their expectations of a business informatics program. They agreed that the impacts of sustainability in software systems should be explored as early as possible. Moreover, engaging in discussions about sustainability effects was seen to enhance their comprehension of the product vision for the system. The students also agreed that they were able to identify a greater number of sustainability impacts than they initially anticipated.

Fourthly, a notable aspect of the project was the voluntary contributions made by some student teams. Three teams went beyond the required scope, providing additional insights into their comprehension of sustainability and the product vision for the software system. Remarkably, one team proposed an innovative feature for the system: a feedback mechanism in the driver's logbook that informs users about the CO2 emissions associated with their driving style.

In summary, the outcomes and student feedback from this exercise are highly encouraging, providing strong motivation to replicate and refine the integration of sustainability concerns in future iterations of this course and other courses of the study program.

5. Limitations

Our study is subject to a number of limitations. Firstly, the student sample represents a convenient sample, indicating that the study participants were not selected based on any demographic criteria.

Secondly, the sample size is relatively small. Only five student teams participated in the study and an evaluation involving additional teams would thus be desirable. Also, each team consisted of only three students. Therefore, we do not know how larger teams would have performed.

Thirdly, the fictional company and its demand for a digital driver's logbook was carefully selected, as the user stories imply some obvious sustainability effects, especially with respect to the ecological dimension (e.g. CO2 emissions) and the individual dimension (e.g. privacy). Consequently, the number of identified sustainability effects is likely influenced by the product vision of the envisioned software system.

Lastly, the potential for collusive behavior among the teams and the extent to which they may have influenced each other remains unknown.

6. Conclusion and Future Work

This paper shares preliminary findings from a study where students are tasked with identifying potential sustainability impacts of a fictional software system they develop in a software project course. The feedback has been predominantly positive, supporting the value of concentrating on sustainability in both educational and research contexts. Accordingly, we aim to inspire fellow academics to explore the integration of sustainability into their pedagogical methods.

Our long term goal is to design and evaluate a comprehensive curriculum that integrates sustainability concepts into various stages of business informatics and computer science education. This involves developing new courses or modules specifically focused on sustainability in software engineering.

References

[1] I. Groher, N. Seyff, T. Iqbal, Towards automatically identifying potential sustainability effects of requirements., in: RE4SuSy@ RE, 2019.

[2] I. Groher, R. Weinreich, An interview study on sustainability concerns in software development projects, in: 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), IEEE, 2017, pp. 350–358. doi:10.1109/SEAA.2017.70.

[3] N. Seyff, S. Betz, L. Duboc, C. Venters, C. Becker, R. Chitchyan, B. Penzenstadler, M. Nöbauer, Tailoring requirements negotiation to sustainability, in: 2018 IEEE 26th international requirements engineering conference (RE), IEEE, 2018, pp. 304–314. doi:10.1109/RE.2018.00038.

[4] N. Seyff, S. Betz, I. Groher, M. Stade, R. Chitchyan, L. Duboc, B. Penzenstadler, C. Venters, C. Becker, Crowd-focused semi-automated requirements engineering for evolution towards sustainability, in: 2018 IEEE 26th International Requirements Engineering Conference (RE), IEEE, 2018, pp. 370–375. doi:10.1109/RE.2018.00-23.

[5] L. Duboc, S. Betz, B. Penzenstadler, S. Akinli Kocak, R. Chitchyan, O. Leifler, J. Porras, N. Seyff, C. C. Venters, Do we really know what we are building? raising awareness of potential sustainability effects of software systems in requirements engineering, in: 2019 IEEE 27th International Requirements Engineering Conference (RE), IEEE, 2019, pp. 6–16. doi:10.1109/RE.2019.00013.

[6] L. Duboc, B. Penzenstadler, J. Porras, S. A. Kocak, S. Betz, R. Chitchyan, O. Leifler, N. Seyff, C. C. Venters, Requirements engineering for sustainability: an awareness framework for designing software systems for a better tomorrow, Requirements Engineering 25 (2020) 469–492. doi:10.1007/s00766-020-00336-y.

[7] P. Bambazek, I. Groher, N. Seyff, Requirements engineering for sustainable software systems: a systematic mapping study, Requirements Engineering (2023). doi:10.1007/s00766-023-00402-1.

[8] B. Penzenstadler, L. Duboc, S. Akinli Kocak, C. Becker, S. Betz, R. Chitchyan, S. Easterbrook, O. Leifler, J. Porras, N. Seyff, C. C. Venters, The SusAF Workshop - improving sustainability awareness to inform future business process and systems design, 2020. doi:10.5281/zenodo.3676514.

[9] IREB, Digital design professional handbook, 2024. URL: https://www.digitaldesign.org/media/pages/downloads/2c6f2fa605-1706682600/ddp_foundationlevel_handbook_en_v2.0.0.pdf.

[10] P. Bambazek, I. Groher, N. Seyff, Application of the sustainability awareness framework in agile software development, in: 2023 IEEE 31st International Requirements Engineering Conference (RE), IEEE, 2023, pp. 264–274. doi:10.1109/RE57278.2023.00034.

[11] S. Lieu, A. Wong, B. Penzenstadler, The 'vattn'case: Analysing sustainability impacts in a software startup, in: 2022 International Conference on ICT for Sustainability (ICT4S), IEEE, 2022, pp. 150–159. doi:10.1109/ICT4S55073.2022.00026.

[12] B. Penzenstadler, S. Betz, L. Duboc, N. Seyff, J. Porras, S. Oyedeji, I. Brooks, C. C. Venters, Iterative sustainability impact assessment: When to propose?, in: 2021 IEEE/ACM International Workshop on Body of Knowledge for Software Sustainability (BoKSS), IEEE, 2021, pp. 5–6. doi:10.1109/BoKSS52540.2021.00010.

[13] B. Penzenstadler, Sustainability analysis and ease of learning in artifact-based requirements engineering: The newest member of the family of studies (it's a girl!), Information and Software Technology 95 (2018) 130–146. doi:10.1016/j.infsof.2017.11.011.

[14] B. Penzenstadler, L. Duboc, R. Hebig, A. Dearden, B. Kanagwa, M. Chaudron, E. Bainomugisha, E. Umuhoza, D. Okello, Raising awareness for potential sustainability effects in uganda: A survey-based empirical study, in: Proceedings of the 8th International Workshop on Requirements Engineering for Sustainable Systems co-located with 27th International Conference on Requirements Engineering, volume 2541, CEUR-WS, 2020.

[15] A.-K. Peters, R. Capilla, V. C. Coroamă, R. Heldal, P. Lago, O. Leifler, A. Moreira, J. P. Fernandes, B. Penzenstadler, J. Porras, et al., Sustainability in computing education: A systematic literature review, ACM Transactions on Computing Education (2024). doi:10.1145/3639060.

[16] C. Becker, R. Chitchyan, L. Duboc, S. Easterbrook, B. Penzenstadler, N. Seyff, C. C. Venters, Sustainability design and software: The karlskrona manifesto, in: 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, volume 2, IEEE, 2015, pp. 467–476. doi:10.1109/ICSE.2015.179.

[17] B. Penzenstadler, A. Raturi, D. Richardson, B. Tomlinson, Safety, security, now sustainability: The nonfunctional requirement for the 21st century, IEEE software 31 (2014) 40–47. doi:10.1109/MS.2014.22.