Supporting Sustainability Aspects in Software Engineering

Birgit Penzenstadler

Institut für Informatik, Technische Universität München, Germany
penzenst@in.tum.de

Abstract. Sustainability is not supported by traditional software engineering methods. This lack of support leads to inefficient efforts to address sustainability or complete omission of this important concept.
Our aim is to support the dimensions of sustainability - human, social, economic, and environmental - within different phases of the software lifecycle, especially requirements engineering and quality assurance.
We contribute a description of the aspects of sustainability in software engineering. The application of sustainability actions on the basis of these aspects is sketched in usage scenarios.
Keywords— sustainability, environment, software engineering, requirements engineering, quality assurance, guidance

Although many people are by now aware of the general definition of sustainable development as "meeting the needs of the present without compromising the ability of future generations to meet their own needs" [13], there still is no concrete guidance for the different aspects of sustainability that are observable from the point of view of software engineering.

Problem Traditional software engineering has not fully supported sustainability as a relevant, first-class concern. This refers to green IT topics, which represent a part of the technical realization possibilities for a sustainable software system, as well as the broader role of software in understanding and tackling the issue, for example, examining the sustainability of the business processes supported by a software system in its application domain.

Contribution We describe the different aspects of sustainability from a point of view of software engineering and exemplarily illustrate their consideration during requirements engineering and quality assurance.

1 Aspects of sustainability in the software lifecycle

Sustainability aspects can be brought to bear both during the development and use of software systems. We distinguish four aspects of sustainability. The first two focus rather on the developing company and its processes, while the latter two have the system under development in scope.

Development process aspect Sustainability in the initial software development process (with responsible use of ecological, human, and financial resources). For example, Naumann et al. propose guidelines for environmentally sustainable web development [12]. Lago et al. propose to measure the environmental impact of software services by their energy consumption [10].

Maintenance process aspect Sustainability of the software system during its maintenance period until replacement by a new system. This includes continuous monitoring of quality and knowledge management. For example, Albertao measures sustainability performance of a software project according to standard quality properties [6].

System production aspect Sustainability of the software system as product with respect to its use of resources for production, for example, by using green IT principles and sustainably produced hardware components. An administrative tool for strategic sustainable development is the ISO 14001 Environmental System Management Standard [3], which is embedded into a planning framework by MacDonald [8].

System usage aspect Sustainability in the usage processes in the application domain triggered by the software system as product. This takes into account responsibility in the impact on the environment and using green business processes. The probably most important step for analyzing and optimizing business processes is taken during requirements engineering, as proposed by Mahaux et al. [9]. Hilty et al. [7] propose a classification of ICT application types as starting point of such an analysis.

All four aspects are relevant for an encompassing approach to supporting sustainability in software engineering. However, we particularly emphasize the system usage aspect, as our hypothesis is that it might have the biggest impact in terms of improvement potential.

2 Usage Scenarios

The following usage scenarios exemplarily describe how software engineers, in particular a requirements engineer and a quality engineer, can apply various actions improving the sustainability of the software system under development. Requirements engineer Rebecca and quality engineer Quentin are developing software systems at Sustainable Software Inc. A new project is coming up: a car-sharing platform, to be developed within 4 months.

2.1 Requirements Engineering

After the kick-off meeting with their customers, Rebecca sketches a first draft of a *Domain Model*[1] with a respective ontology. As a description of the operational

[1] Items in *italics* are part of a domain-independent requirements engineering content model available at `http://www4.in.tum.de/~penzenst/sources/DomainIndependent-RE-ContentModel.png`

and business context, the model serves as communication basis for discussing and deriving goals during the next meeting. The explicit modeling of sustainability aspects of the domain, e.g., emission reduction, community building, etc., triggers a discussion on how to emphasize these aspects in a *System Vision*. The goals arising from the workshop are described in a *Goal Model*.

Using life cycle analysis and impact assessment on the ideas depicted in the *System Vision*, Rebecca manages to convince their customers of a more environmentally sustainable business process.

The *Domain Model*, the *System Vision*, and the *Goal Model* serve as inputs for the *Usage Model*. The latter is detailed in use cases and their scenarios to elaborate functional requirements as well as user stories to describe quality goals. For example, the quality goal "The system shall be easily maintainable", Rebecca sketches a number of user stories, where one of them describes, e.g., how the database administrator edits tables at runtime. Her activities mainly contribute to the System Usage Aspect.

Rebecca hands over the results to her new colleague Quentin, curious how he likes the new way of representing the quality requirements. Quentin likes the user stories, as they enable a requirements engineer to illustratively describe quality requirements and to get the message across, thereby, providing a good interface to the quality model used by quality assurance.

2.2 Quality Assurance

At university, Quentin learned that the constructive part of quality engineering is always accompanied by the analytical part of quality assessment. For objectivity and traceability, Quentin wants to set up an assessment scheme with Key Performance Indicators (KPIs) that rely on common metrics, for example, the KPI library [4], the Environmental Sustainability Index [1], and the Sustainability Index [2].

For assessment of the project management, he uses the Sustainability Maturity Model [11], and for assessment of the product, he considers the sustainability metrics proposed by Albertao et al. [5]. His metrics show that Rebecca managed to achieve a considerable improvement in reducing energy consumption and emissions by optimizing the *Usage Model* of the car-sharing platform in her analysis and convincing her customer of the adaptations.

Following these activities, Quentin is continuously improving the support of sustainability in their product under development, and his progress can be measured. His activities mainly belong to the Development Process Aspect.

3 Conclusion and future work

This paper presented emerging research on supporting different aspects of sustainability via requirements engineering and quality assurance.

As preliminary evaluation, we are currently elaborating a case study with a major automotive company. This will be followed by a third-party evaluation via industrial collaboration in a different application domain.

Future Work We envision an encompassing approach to build sustainability into software products with a corresponding assessment model. The author is currently working on a first evaluation of the content model and the scenario activities in student projects in a series of seminars[2] and students' theses.

For the *Quality Model*, we intend to combine adequate standard KPIs [4] and metrics used for the Environmental Sustainability Index [1] and the Sustainability Index [2] into a new Software Sustainability Index (SSI). A dashboard tool for ease of use will support the SSI.[3]

References

1. Environmental sustainability index report 2005. www.yale.edu/esi/ESI2005_Main_Report.pdf.
2. Sam indexes gmbh: Dow jones sustainability indexes. http://www.sustainability-index.com.
3. ISO 14001 Environmental Management System Standard. International Standardization Organization, November 2009.
4. Key performance indicator library. www.kpilibrary.com, December 2011.
5. F. Albertao, Jing Xiao, Chunhua Tian, Yu Lu, Kun Qiu Zhang, and Cheng Liu. Measuring the sustainability performance of software projects. In *2010 IEEE 7th International Conference on e-Business Engineering (ICEBE)*, pages 369–373. IEEE, November 2010.
6. Felipe Albertao. Sustainable software engineering. CMU SEI online, 2004.
7. Lorenz Hilty et al. The relevance of information and communication technologies for environmental sustainability. *Environm. Modelling & Software*, 21(11):1618 – 1629, 2006.
8. Jamie MacDonald. Strategic sustainable development using the iso 14001 standard. *Journal of Cleaner Production*, 2005.
9. Martin Mahaux et al. Discovering Sustainability Requirements. In *17th Intl. Working Conf. on Requirements Engineering: Foundation for Software Quality*, 2011.
10. Patricia Lago et al. The service greenery - integrating sustainability in service oriented software. In *International Workshop on Software Research and Climate Change (WSRCC)*, 2010.
11. G. Silvius and R. Schipper. A maturity model for integrating sustainability in projects and project management. In *24th World Congress of the International Project Management Association*, 2010.
12. Stefan Naumann et al. The greensoft model. *Sustainable Computing: Informatics and Systems*, (0):–, 2011.
13. United Nations World Commission on Environment and Development. Report: Our Common Future. In *United Nations Conference on Environment and Development*, 1987.

[2] `http://www4.in.tum.de/~penzenst/teaching.shtml`

[3] The author would like to thank Bill Tomlinson and Debra Richardson for feedback on early ideas.