CT70A2000
Requirements Engineering

Shola Oyedeji

LUT University

Sustainability Awareness Survey

https://link.webropolsurveys.com/S/96FE82A700071BB6

10 minutes to fill

LECTURE 4

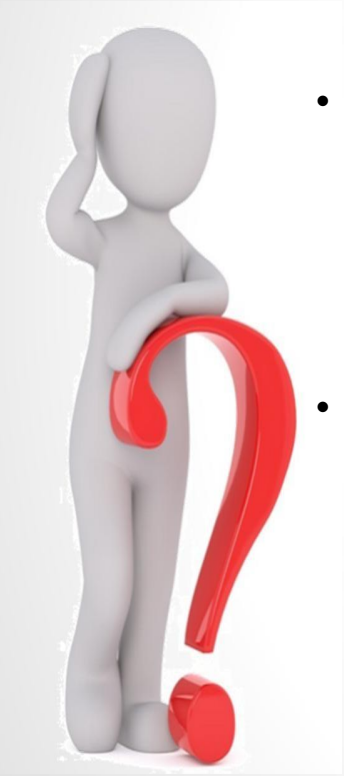
Sustainability – What & Why ?

Requirements and sustainability issues

Sustainability and Software

Sustainability Analysis of Software Requirements Specification

What is Sustainability ?



- **Sustainability** is also defined as the ability to continue a defined behaviour indefinitely with consideration of the environment, society and economy.
- **Sustainability** is a vision for the world in which current and future humans are reasonably healthy; communities and nations are secure, peaceful and thriving; there is economic opportunity for all; and the integrity of the life-supporting biosphere is restored and sustained at a level necessary to make these goals possible.
- Today, **sustainability values** from the environmental, economic, and social dimensions such as healthy environment, vibrant economy and equitable society are those we all aspire to achieve.

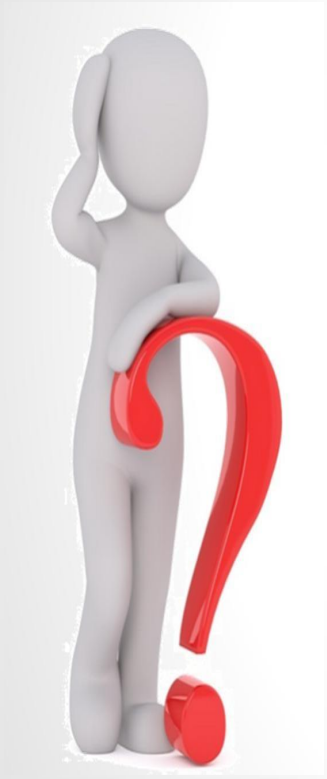
Sustainability - Why?

"Surely we have a responsibility to leave for future generations a planet that is healthy and habitable by all species"

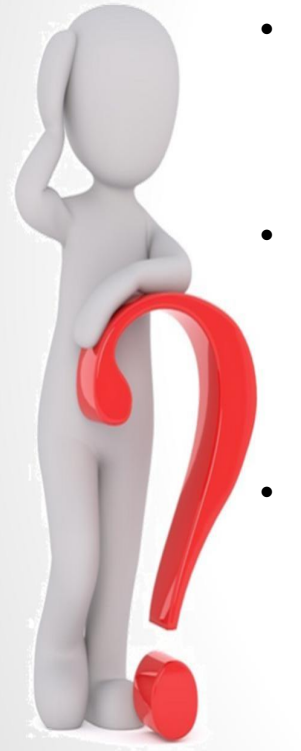
– Sir David Attenborough

Sustainability - Why?

- The importance of sustainability in all aspects of human life and development are highlighted by the United Nations Sustainable Development Goals (SDGs).



Sustainability - Why?



- Today we live in a digital world where software affects all aspects of our lives and the society in general.
- Software design is a process to transform user requirements into some suitable form, which helps the programmer in software coding and implementation.
- The design phase of software development deals with transforming the customer requirements as described in the Software Requirement Specification (SRS) documents into a form implementable using a programming language.

LECTURE 4

Sustainability – What & Why ?

Requirements and sustainability issues

Sustainability and Software

Sustainability Analysis of Software Requirements Specification

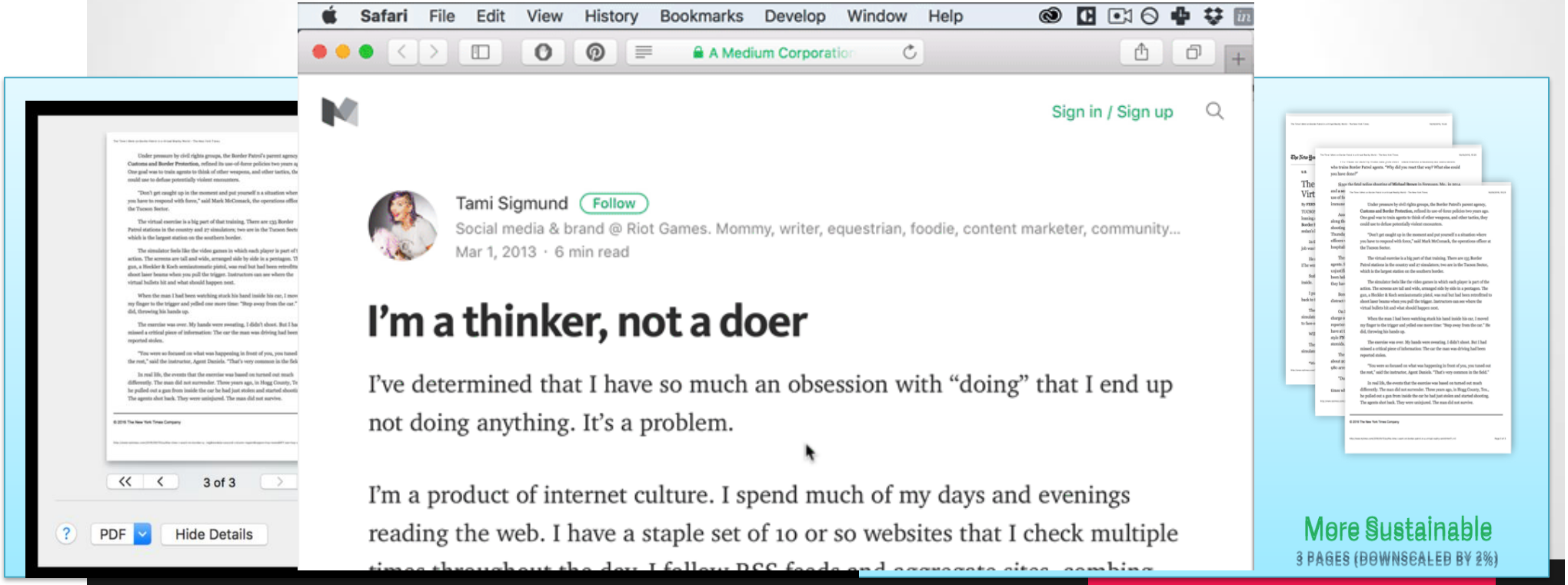
As a requirement engineer or analyst do you have a role towards sustainable development?

What can you do as Requirement Engineer or Analyst ?

- First, Requirement engineers or analyst should understand the breadth of sustainability and the strategies for developing more sustainable solutions.
- Become advocates of sustainability issues for their own organizations and clients, partners, and other stakeholders addressing sustainability issues in our projects whether the clients and organizations appreciate them or not.
- Over time, you can address more issues and integrate more sustainability strategies into software requirement gathering and design work naturally.

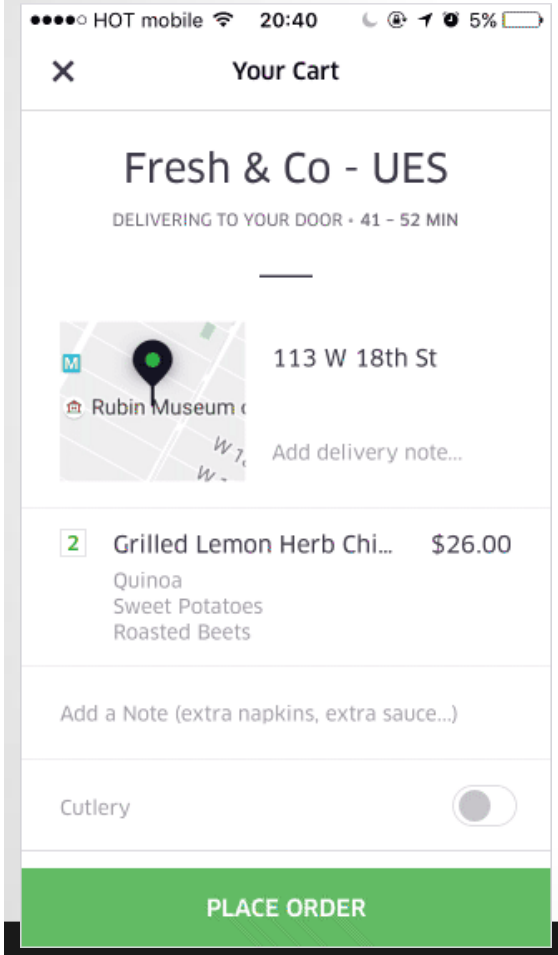
Requirements and sustainability issues

Mac OS on Paper waste
Chrome don’t allow scale down option but minimize margin options from more settings

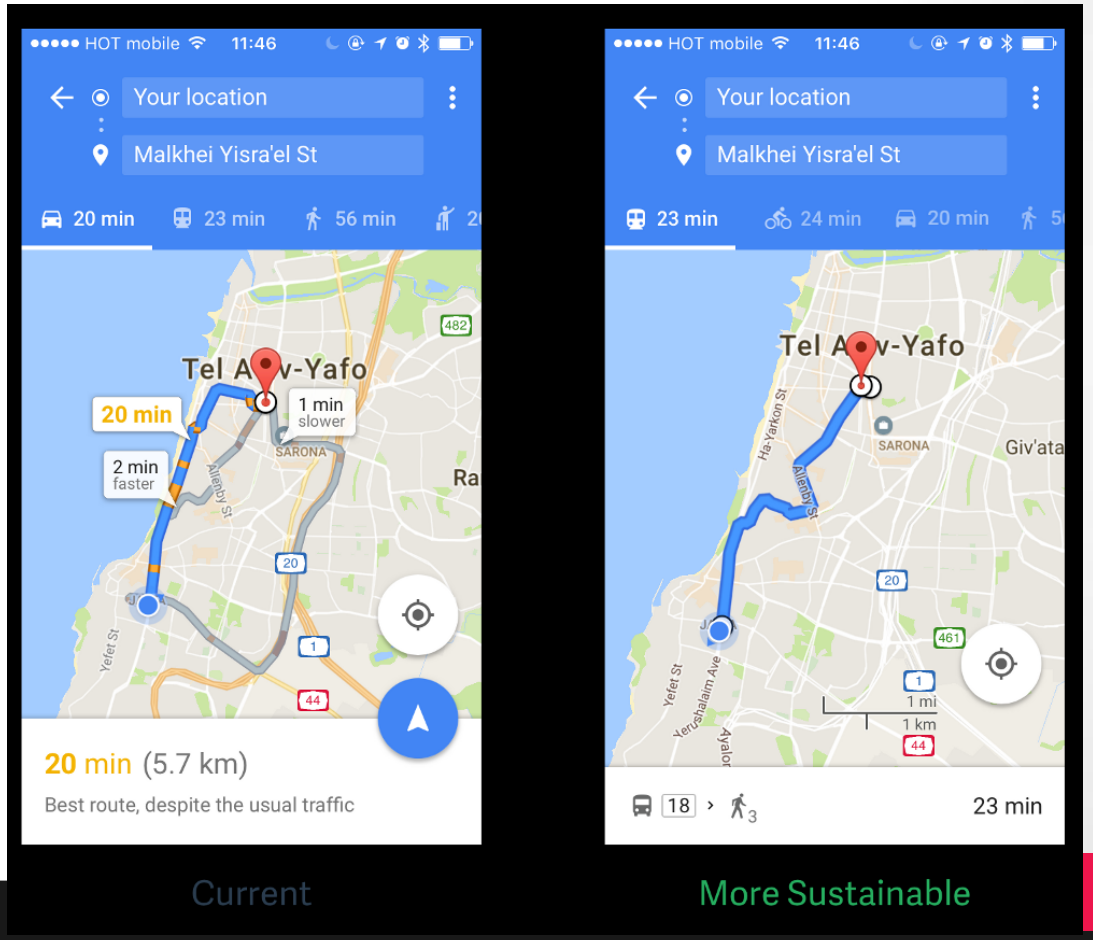


Requirements and sustainability issues

Food deliveries should promote less waste



Google Maps should encourage people to prefer public transportation and bikes over cars



LECTURE 4

Sustainability – What & Why ?

Requirements and sustainability issues

Sustainability and Software

Sustainability Analysis of Software Requirements Specification

Definitions: Sustainable Use & Sustainable Development

Sustainable use of a system S with regard to a function F and a time horizon T [Hilty 2015]
→ “use S in a way that does not compromise its ability to fulfil F for a period of T”

Sustainable development is “meeting the needs of the present without compromising the ability of future generations to meet their own needs ” [UN 1987]
→ acknowledging the concept of changing stakeholder requirements

Sustainability and Software

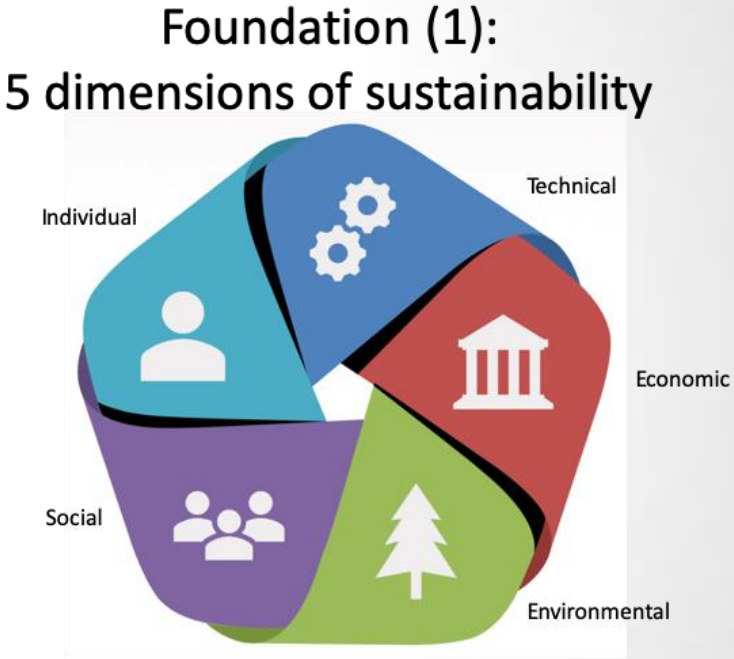
- In software engineering or software development, sustainability is categorized into five dimensions, namely, economic, environmental, social, individual and technical.

  - **The environmental dimension** refers to the use and maintenance of natural resources. ‘How does software impact and affect the environment and energy consumption’.
  - **The economic dimension** is the financial aspects: maintaining financial capital and added value. The focus is on how to design and develop software systems in a cost-effective manner.
  - **The social dimension** covers relationships between individuals and groups. ‘What are the impacts of software systems and applications on the society?’ (Example: communication, sense of belonging, interaction and equality).
  - **The individual dimension** refers to the maintenance of individual human capital, and equal access to services. It includes the ability of individuals (end users) to thrive and exercise their rights.
  - **The technical dimension** refers to maintenance and evolution, resilience, and the ease of system transitions. ‘How can software be designed and developed for easy evolution, maintainability, adaptability to changes in the future’?

Sustainability Dimensions

The five dimensions of sustainability allows software development stakeholders (requirement engineers, analysts, developers, companies) to evaluate the impacts of softwares using the:

- **First order impact**: (immediate effects) are about the direct effects of the development and use of a software system.
- **Second order impa**cts : (enabling effects) are about the indirect impacts related to the effects of using the software system in its application domain.
- **Third order impacts**: (structural effects) are the cumulative long-term effects resulting from accumulating first and second order impacts over time.



[GIBSE'13] Birgit Penzenstadler, Henning Femmer
A Generic Model for Sustainability with Process- and Product-specific Instances.
1st Intl. Workshop on Green In Software Engineering, Green By Software Engineering (at AOSD'13)

The Karlskrona Manifesto for Sustainability Design

https://www.youtube.com/watch?v=kcr8acz_q-Y

Introduction

Version 1.0, May 2015

As software practitioners and researchers, we are part of the group of people who design the software systems that run our world. Our work has made us increasingly aware of the impact of these systems and the responsibility that comes with our role, at a time when information and communication technologies are shaping the future. We struggle to reconcile our concern for planet Earth and its societies with the work that we do. Through this work we have come to understand that we need to redefine the narrative on sustainability and the role it plays in our profession.

What is sustainability, really? We often define it too narrowly. Sustainability is at its heart a systemic concept and has to be understood on a set of dimensions, including social, individual, environmental, economic, and technical[1].

Sustainability is fundamental to our society. The current state of our world is unsustainable in more ways that we often recognize. Technology is part of the dilemma and part of possible responses. We often talk about the immediate impact of technology, but rarely acknowledge its indirect and systemic effects. These effects play out across all dimensions of sustainability over the short, medium and long term.

Software in particular plays a central role in sustainability. It can push us towards growing consumption of resources, growing inequality in society, and lack of individual self-worth. But it can also create communities and enable thriving of individual freedom, democratic processes, and resource conservation. As designers of software technology,

There is a perception that sustainability is a distinct discipline of research and practice with a few defined connections to software.
Whereas sustainability is a pervasive concern that translates into discipline-specific questions in each area it applies.

There is a perception that sustainability is a problem that can be solved, and that our aim is to find the 'one thing' that will save the world.
Whereas it is a 'wicked problem' - a dilemma to respond to intelligently and learn in the process of doing so; a challenge to be addressed, not a problem to be solved.

There is a perception that there is a tradeoff to be made between present needs and future needs, reinforced by a common definition of sustainable development, and hence that sustainability requires sacrifices in the present for the sake of future generations.
Whereas it is possible to prosper on this planet while simultaneously improving the prospects for prosperity of future generations.

There is a tendency to focus on the immediate impacts of any new technology, in terms of its functionality and how it is used.
Whereas the following orders of effects have to be distinguished: *Direct, first order effects* are the immediate opportunities and effects created by the physical existence of software technology and the processes involved in its design and production. *Indirect, second order effects* are the opportunities and effects arising from the application and usage of software. *Systemic, third order effects*, finally, are the effects and opportunities that are caused by wide-scale use of software systems over time.

There is a tendency to overly discount the future. The far future is discounted so much that it is considered for free (or worthless). Discount rates mean that long-term impacts matter far less than current costs and benefits.
Whereas the consequences of our actions play out over multiple timescales, and the cumulative impacts may be irreversible.

There is a tendency to think that taking small steps towards sustainability is sufficient, appropriate, and acceptable.
Whereas incremental approaches can end up reinforcing existing behaviours and lure us into a false sense of security. However, current society is so far from sustainability that deeper transformative changes are needed.

There is a tendency to treat sustainability as a desirable quality of the

http://www.sustainabilitydesign.org

The Karlskrona Manifesto for Sustainability Design

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	Transmaterialization, 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 choices 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 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 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.

Oyedeji, Shola, Ahmed Seffah, and Birgit Penzenstadler. 2018. "A Catalogue Supporting Software Sustainability Design" *Sustainability* 10, no. 7: 2296. https://doi.org/10.3390/su10072296

LECTURE 4

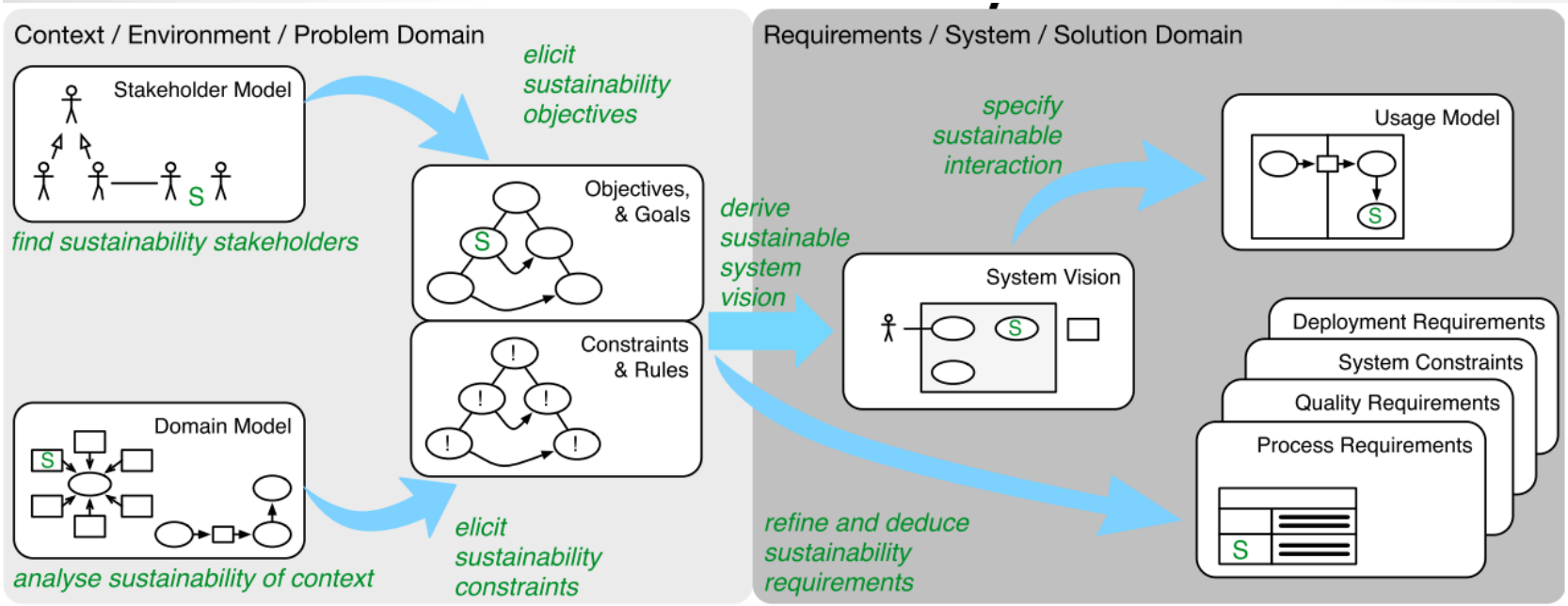
Sustainability – What & Why ?

Requirements and sustainability issues

Sustainability and Software

Sustainability Analysis of Software Requirements Specification

Requirement Engineering for Sustainability



Example checklist for analyzing sustainability of a software

Guiding Questions:

1. Dose the system have an explicit sustainability purpose ?
2. Which impact does the system have on the environment and people?
3. Is there a stakeholder for environmental, individual and social sustainability?
4. What are the sustainability goals and constraints for the system?

Requirement Engineering for Sustainability

1. Stakeholder

What is a stakeholder ?

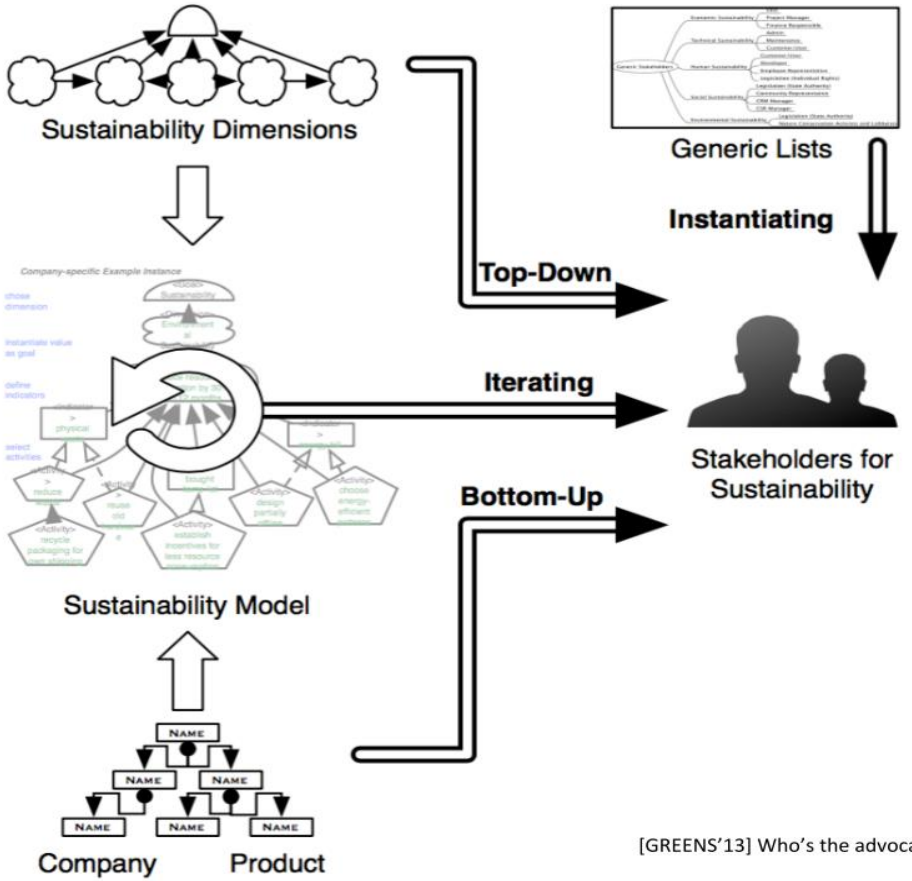
A stakeholder is a person or organization who influences a system's requirements or who is impacted by that system.

Stakeholder Model

- Stakeholder Model shows and describe all stakeholders involved in a project.
- This provides you a better overview and a way to categorize all your stakeholders into user groups.
- User groups are a specialization of stakeholders interacting with the system.

Requirement Engineering for Sustainability

1. Stakeholder
Stakeholder Model –Example



[GREENS'13] Who's the advocate?

TABLE I
A GENERIC LIST OF SUSTAINABILITY STAKEHOLDERS

Dimension	Stakeholder	Description/Rationale
Individual	User	The user is affected by the system in various ways. For example, users of online learning courses educate themselves through software.
	Developer	The developer is heavily involved in creating the system. Aspects like sustainable pace and growth of the developer must be considered.
	Employee represent.	The mental and physical safety of individuals needs to be maintained. Employee representatives watch rights of employees involved.
	Legislation (indiv. rights)	Systems must respect the rights of their users. A legislation representative is a proxy for privacy and data protection laws.
Social	Legislation (state authority)	The state has a strong interest in understanding a system's influence on the society. In contrary to the individual rights legislation representative, the state authority representative speaks from the perspective of the state as a whole.
	Community represent.	In addition to the state authority, other communities such as the local government (e.g. the mayor) or non-government clubs might be affected by a software system. A complete analysis must take their views into account.
	CRM	The Customer Relationship Manager (CRM) is in charge of establishing long-term relationships with their customers and creating a positive image of the company.
	CSR manager	Some companies created the dedicated position of the Corporate Social Responsibility (CSR) manager, who develops a company-specific vision of social responsibility.
Economic	CEO	The chief executive officer integrates sustainability goals into a company's vision.
	Project manager	It is very important to have the project manager agree in what ways the project should support sustainable aspects as he decides on prioritization with conflicting interests.
	Finance responsible	As sustainable software engineering often also affects the budget, many financial decisions have to be made to implement a sustainable software engineering model in a company.
Environm.	Legislation (state authority)	Environment protection laws are in place to ensure sustainability goals. These laws must be reflected in the model.
	CSR manager	The CSR manager is often also responsible for environmental aspects.
	Activists /Lobbyists	Nature conservation activists and lobbyists (e.g., WWF, Greenpeace, BUND)
Technical	Admin	The administrator of a software system has a strong motivation for long-running, low-maintenance systems, makinghis work easier.
	Maintenance	The hardware maintenance is interested in a stable, long-term strategy for installation of hardware items.
	Customer	Users are interested in certain longevity of the systems they are using. This refers to user interface and required soft- and hardware.

Lect. slides guest course Softw. Eng. Sustain. Lappeenranta Univ. Technol. 2016. Slides available online at http//birgit.penzenstadler.de/teach/LUT.html.

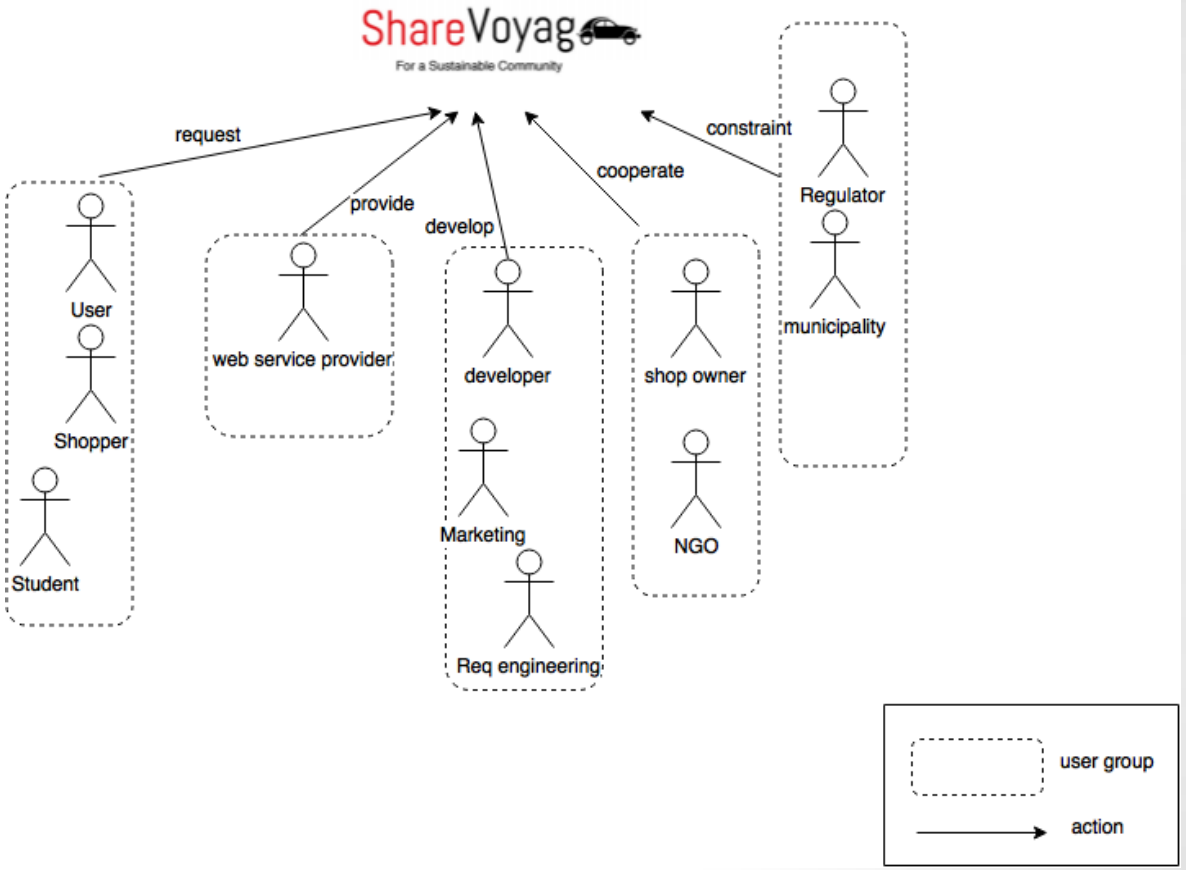
Requirement Engineering for Sustainability

1. Stakeholder
Stakeholder Model –Example

Example:shareYoyage (car pooling)

Goal: The goal of this project is to provide students with a web based and mobile application called shareYoyage for car pooling and food sharing in order to build a sustainable community of students willing to use resources efficiently.

Impact: The proposed application connects and ensure collaboration between users to travel together with low cost which reduce co2 emission and aid food sharing which reduce food wastage.



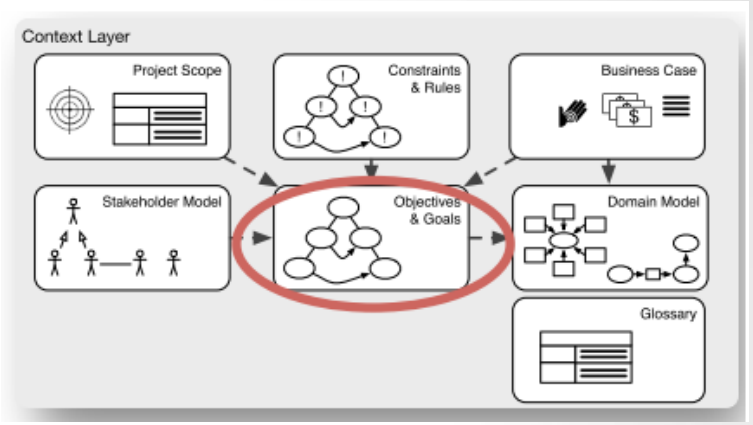
Requirement Engineering for Sustainability

2. Goal

What is Goal?

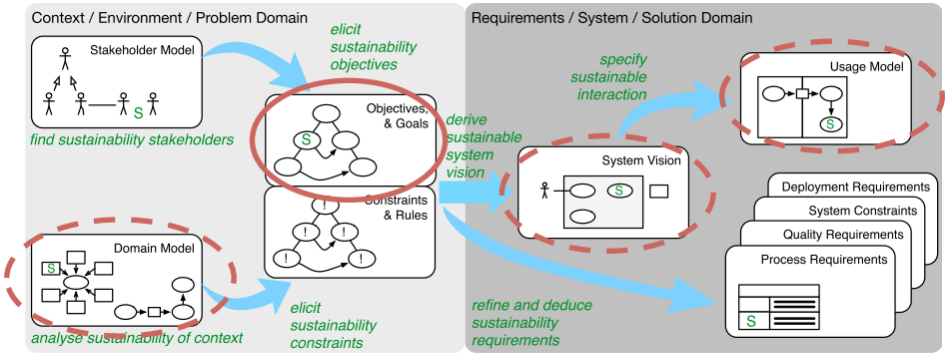
Discretionary abstract characteristic, which

1. the system shall fulfill in it‘s operational environment or
2. the development process of the system shall fulfill



Goal Model

- Provides a holistic grouping of software application goals into business goal, usage goal and system goals to identify conflicts early in order to resolve them with consideration of the five sustainability dimensions
- Modeling of the system behavior on different levels of abstraction (Domain Model, System Vision, Usage Model)

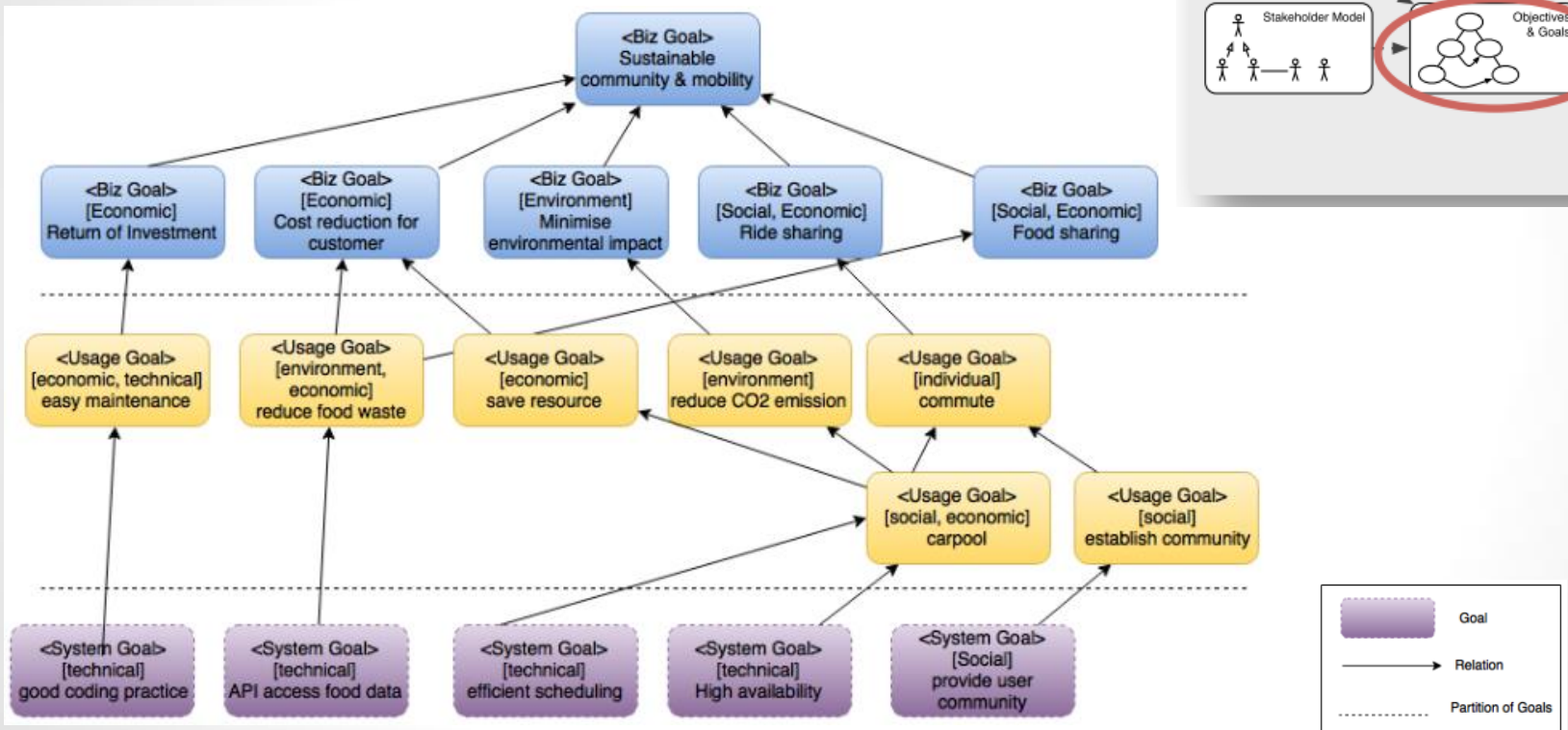


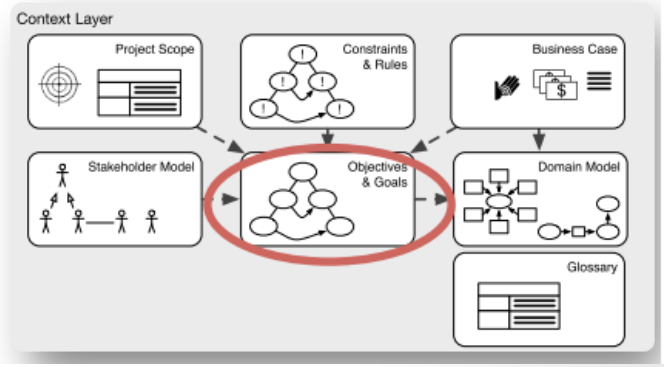
Penzenstadler, B Lect. slides guest course Softw. Eng. Sustain. Lappeenranta Univ. Technol. 2016. Slides available online at http//birgit.penzenstadler.de/teach/LUT.html.

Requirement Engineering for Sustainability

2. Goal

Goal Model – Example

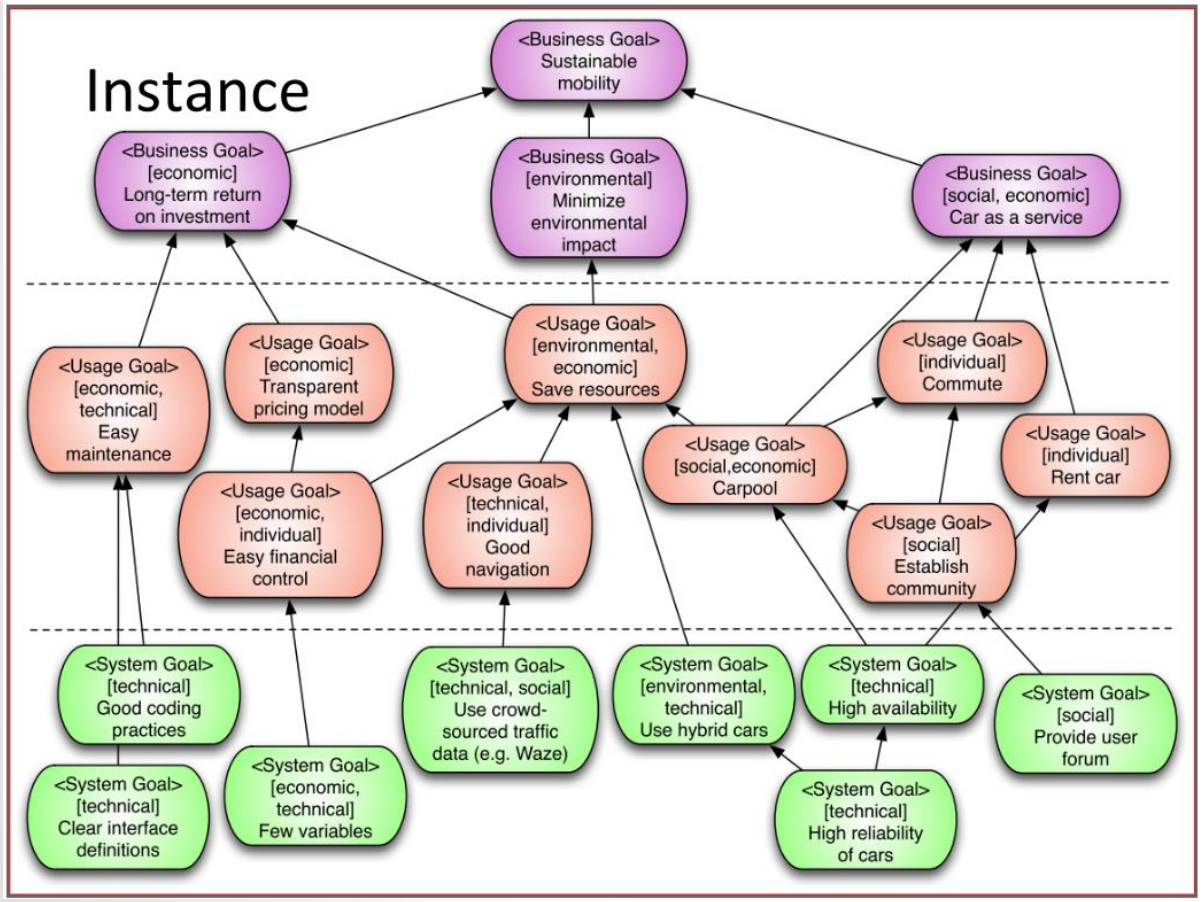


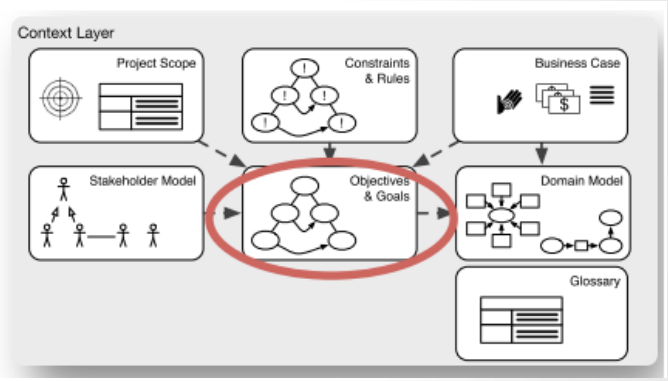


Requirement Engineering for Sustainability

2. Goal

Goal Model – Example





Penzenstadler, B Lect. slides guest course Softw. Eng. Sustain. Lappeenranta Univ. Technol. 2016. Slides available online at http//birgit.penzenstadler.de/teach/LUT.html.

Requirement Engineering for Sustainability

3. System Vision

What is System Vision?

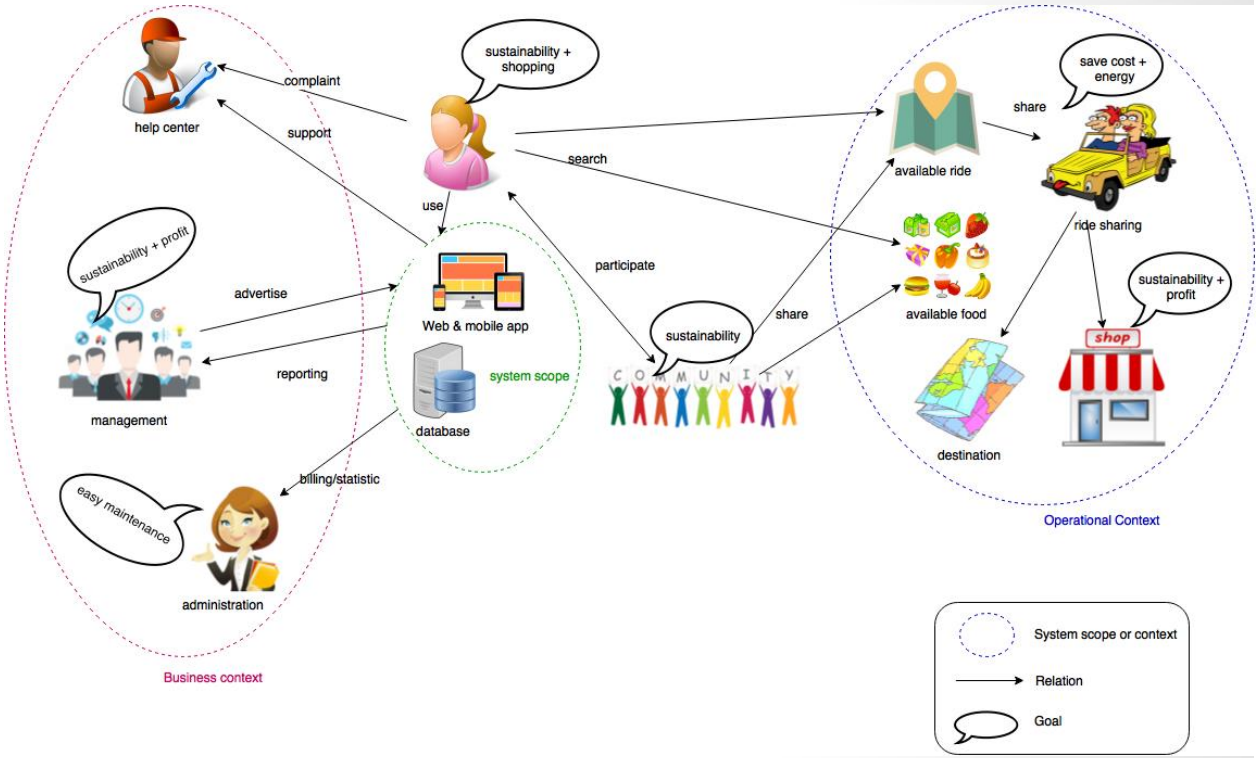
The system vision is a joint vision of the system agreed upon by all active stakeholders

Characteristics

- Big picture
- Abstract

Purpose

- Agreement on what this project is about
- Easy communication with stakeholders



Framework for Sustainability of Software System Design (FSSSD)

FSSSD is a framework covering all software development life cycle phases with a structural guide on how stakeholders (requirement engineers, analyst, developers, companies) can incorporate sustainability goals and link requirement to sustainability for better evaluation of software sustainability focusing all dimensions (environmental, economic, social, individual and technical).

SDLC phases	Sustainability goals	Sustainability concepts, Methods and Tools	Indicators
Phase 1. Project definition	Design for sustainable efficiency, reusability	Sustainable business canvas	Carbon footprint, material footprint, end of life footprint
Phase 2. User requirements definition	Increase sustainability awareness among users	Sustainability requirement template	Total number of sustainability requirements, priority assign to sustainability requirements
Phase 3. System requirements definition	Design for efficiency, sustainability awareness and interoperability	Cradle to cradle, goal model	Total number of system goals relating to sustainability dimensions

Oyedeji, S.; Seffah, A.; Penzenstadler, B. A Catalogue Supporting Software Sustainability Design. *Sustainability* **2018**, *10*, 2296. https://doi.org/10.3390/su10072296

Framework for Sustainability of Software System Design (FSSSD) Example

The pension benefit tracker is an application from a pension company that wants to track pension benefit applications submitted by clients from all over the company's branches in different states. 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:

- Zonal managers don't have direct access to know the status of applications submitted through them but have to directly place phone calls to the Head office to know the application status.
- Customer service staff are unable to know why an application is pending, unless they contact the bene-fit department.
- Time consumption, as all status updates are through customer service at the head office alone.
- Files can go missing in transit because application files are handled manually.
- 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 man-agers and the clients with the aim of:

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

Framework for Sustainability of Software System Design (FSSSD)

SDLC Phases	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 • The pension tracker application should be accessible online via web at any branch • Allow pension benefit application submission • Provide tracking of pension benefit payment • Send Status notification 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?

Framework for Sustainability of Software System Design (FSSSD)

SDLC Phases	Sustainability Goals	Sustainability Concepts, Methods and Tools	Indicators /Measure / Metric
Phase 3. System Requirements Definition • Provide automatic email status notification after each stage of benefit application to applicant and managers • Allow bulk or single file upload • Provide SMS authorization from managers in benefit department • Send SMS notification to applicants • Send Incomplete documentation notification to benefit department staff • Provide option of different display to magnify fonts for users with visual problems • Provide option to preview pension application • Add a tag message below each notification “Save the planet, Reduce environmental waste” • Provide energy report for system usage	Design for efficiency, sustain-ability 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? 4. How many initiatives were suggested from IT department base on the system energy report?

Sustainability Requirement Template

Sustainability Requirement Template: provide requirement engineers / analyst a way of categorizing software requirements into the five sustainability dimensions. The template foster better thinking on how software requirements relates to each sustainability dimension and provide an avenue to understand the requirement categorization for stakeholders

Requirement	Sustainability Dimension	Explanation
User Requirement: State each of the requirement in a way that makes it possible to associate the requirement to at least one or more of the sustainability dimensions **System Requirement:** State each of the requirement in a way that makes it possible to associate the requirement to at least one or more of the sustainability dimensions	Highlight which of the sustainability dimension relates to all the stated requirements. These are the general explanation of the five sustainability dimensions: • **Individual** sustainability refers to maintaining human capital (e.g., health, education, skills, knowledge, leadership, and access to services). It includes the ability of individuals (end users) to thrive and exercise their rights. • **Social** sustainability aims at preserving the societal communities in their solidarity and services. 'What are the impacts of software systems and applications on the society?' (Example: communication, sense of belonging, interaction and equality). • **Economic** sustainability aims at maintaining capital and added value. The focus is on how to design and develop software systems in a cost-effective manner. • **Environmental** sustainability refers to improving human welfare by protecting the natural resources: water, land, air, minerals and ecosystem services. 'How does software impact and affect the environment and energy consumption'. • **Technical** sustainability refers to longevity of information, systems, and infrastructure and their adequate evolution with changing surrounding conditions. 'How can software be designed and developed for easy evolution, maintainability, adaptability to changes in the future'?	Provide an explanation for your decision to associate each requirement to a particular sustainability dimension.