

Empirical Study on Green and Sustainable Software Engineering

M.MOHANKUMAR

Department of Computer Science,
Karpagam University,
Pollachi Main Road, Eachanri post, Coimbatore-641021
India
mohankumarcs@karpagamuniversity.edu.in

DR.M.ANAND KUMAR

Department of Information Technology,
Karpagam University,
Pollachi Main Road, Eachanri Post, Coimbatore-641021
India.
anandkumarit @karpagamuniversity.edu.in

Abstract- Green software engineering is a main software engineering process in the 21st century. In the previous era, software engineers concerned mainly with development of hardware or software without giving more importance to sustainability. There is no effort towards the technical, economical, environmental, social and individual aspects of the sustainability of the environment. In developed countries researches are mainly focused on green and sustainable software engineering. But in developing countries like India and china there is no more research on this, so more research works are to be carried on Green software engineering. The developing countries are developing more software applications and ICT products alone for the developed countries, but Green sustainable software engineering process is a major issue that is to be solved to make efficient software models. The developed software models should focus mainly on environment and on earth which are common for all the countries. This paper analyses about the Green and the sustainability of software engineering approach in the life cycle models for software development process. This work surveys the existing software models and their impact on environments.

Key words: Information and Communication Technology, Environmental protection Agency, Management Information System, Software Development Life Cycle, SBA Small Business Administration, Windows Event Tracing for Windows, key performance indicators.

1. Introduction:

The main challenges of the 21st century are the global impacts of the environment. The environment impacts may be in any form like noise, heat, radiations and power consumption. The above set of factors affects the environment [1] directly or indirectly which gives negative impacts on economy, society, human beings and

environment that result from development, deployment and usage of the software. In the development of a software application, there exists many lifecycle models, where each model has its own sustainability and environment threats. But most of the research activities focuses only on energy efficiency and energy consumption [2]. For example it has been estimated that the energy consumption of data centers in the

United States are increased from 28 billion (Kw/hour) in 2000 to 61 billion (Kw/hour) in 2006 on a global scale and a similar development has been recorded for power consumption which raise from 58 million (Kw/hour) in 2000 to 123 billion (Kw/hour) in 2005. Another environmental threat was [3] global carbon dioxide emissions reached 9.1 billion tones, which is the highest level in human history 49% higher than in 1990. At least 2 percent of global carbon dioxide emissions are attributed to IT systems and further increases are expected as New IT systems which are deployed daily. Thus its essential to reduce the energy consumption and related carbon dioxide emission of IT systems. The above factors are almost related to hardware and software factors that continue to affect all aspects of the environment in ever changing forms Leveraging of existing systems is a challenging task for many companies.

The remainder of this paper gives an overview of the literary review in the second chapter (section II). The third chapter describes about the various tools which are used for finding Green and sustainable software engineering methods. The fourth chapter describes the statement of the problem in SDLC(Software Development Life Cycle).Fifth chapter deals with the approaches for sustainable software engineering using Green technologies in SDLC phases and advantages in that approach. And the last chapter deals about the conclusion with an outlook on future works.

2. Literature Survey:

The study of Green and Sustainable development was first introduced at the World Convention on Environment and Development in 1987[4]. It defined the Green software process as “development that meets the needs of the present without compromising the ability of future generations to meet their own needs. It soon becomes a paradigm for software organizations. The research started with the software safety and security in the initial

stages and currently focus was shifted to reduce waste or raw materials and save the environment, the concept of Green software becomes increasingly complex [5].

The hardware aspects are covered by Green IT concepts where many publications on sustainability of data centers were published. For instance, the code of conduct for data center efficiency [6]. There is no execution model and software tools to identify the estimation of energy consumption in these early design stages[7].This mixed role puts technology organizations under tremendous conflicting pressures, internally they are eco friendly and externally they are expected to design new products that improve the sustainability of the society at large[8].In the quality aspect the commoditization has resulted in the software industry coming under increasing pressure to develop and to deliver greater volumes of high quality products and services within cost and schedule constraints that are tighter than ever before[9].

This paper [10] pointed out lot of aspects related to quality. The quality is one of the aspects in testing of software where codes written by an automatic interpreter have less efficiency than code written by experienced developers. The paper [11] shows how the power reduction can be achieved through improved software design. It also pointed out the role of software engineers in reducing the power consumption of the application. This work [12] outlined about the importance of classification for Green quality factors and defines about the Green metrics. The work [13] discussed about the approaches which lead to identify challenges based on quality, requirement and design.

- How does the sustainability differ from other approaches?
- What are the types of requirement gathering method that lead to sustainable s/w solutions?
- How do they differ from traditional Non Function Requirements (NFS) on requirement perspective?

The author [5] proposed software security considerations from 1980 to 2010 era, in which the developers focuses mainly on requirement and process, secure

computations, verification of systems, software piracy protection architecture and design of secure systems, trusting components are secure development factors. In the year 1990 to 2010 the developers were focused designing for safety, testing at different levels for reliability and safety, certification and standards resource available on software safety requirements and analysis and hazard analysis are the above factors that comes under the software safety considerations.

In the year 2010 software sustainability considerations safety is an emergent property that arises when the system components interact within an environment [14]. The supporting sustainability requires knowing the stakeholders as the key challenge and success factor for all the projects also for the sustainable and Green software. There is a lot of effort trying to combat pollution issues especially the European Union (EU) cap is the limited amount of emission from each country and it can be traded in the form of emission permit. Installations are able to hold credit but it will not be allowed to exceed the cap. However if the installations would like to emit more gas they have to pay in order to obtain credit, therefore not only carbon foot print or hazardous substance but also environment foot print has to be monitored closely by manufacturers. This method has to be set for software development industries [15]. Reducing the energy consumption and overall environmental impacts of data centers have become an important research area in future [16]. The European commissions Europe 2020 strategy has targeted three key areas for sustainable growth. They are

1. 20% increase in energy efficiency
2. 20% reduction of green house gas emissions and
3. Increase the share of renewable energy by 20%.

So more concentration is required in these areas in both developed and developing countries. [17]

Software development plays a specific role in creating rebound effects. The usual response of software engineers is to increase

the processing power and storage capacity available at a given price is to capture more of the same [18]. This paper focuses on the Green potential of clouds such as:

1. How they have to be deployed for different user levels highlighting the related environmental risks?
2. Giving the growing importance of cloud computing the question is not whether it is green as it is now but how it can become really green?
3. Awareness and responsible behaviors are background condition to achieve sustainable and Green cloud computing.

One of the pillars of the information society strategy of the European Union is the application of ICT (Information and Communication Technology) to improve the quality of life and to foster environmental care and sustainable development [28]. Over a third of organizations in Europe do not implement Green IT practices the most prominent reason given is that there is no official legislation in their countries in enforcing Green IT practices. Less than one fifth of the organizations actually monitor how employees reduce their energy consumption [19]. The paper empirically analyzed the energy consumption induced by comparable MIS (Management Information System) applications namely, ERP (Enterprise Resource Planning) CRM (Customer relation management) and DBMS and found out that: (i) not only infrastructural layers, but also the MIS application layer does impact energy consumption up to 70% (ii) different MIS applications are satisfying the same functional requirements to consume significantly different amounts of energy (differences up to 145%) and (iii) in some scenarios energy efficiency cannot be increased simply by improving time performance.

2.1 Design Perspective:

In software engineering, a design pattern is a general reusable solution to a commonly occurring problem in the software design process [20]. The work [21] pointed out the

considerations on Green and sustainability in the design process. This paper focused on a lot of ideas about how to promote Green software values. Energy efficiency should be given more important when it comes down to general algorithm efficiency where there is a need to make thing runs fast with less hardware. But this is amplified and driven by pricing schemas of cloud resources and cost saving: that need to write efficient software that can do more with limited resources. This will translate into power efficiency when there is a increase in the amount of work done per CPU Cycle.

There is a myth on Green software engineering which denotes that if the software is built more efficiently then it will consume less energy. So there is a need of global standard which should be bigger than any practitioner who promotes it. Otherwise it will be really hard to raise the awareness. The work [22] describes the best examples for studies published on the average electricity consumption of a single Google search query. In 2007 EPA (Environmental Protection Agency) data center reported to the US congress that by 2011 the peak load will be generated on the grid by data centers in US alone. It would be close to 12 GW which is equivalent to the output from 25 base load power plants where research community need to focus on this area.

In terms of chip manufacturing, a study [22] shows that the amount of resources and energy consumed are measured as a ratio against the weight of the final product (chip) which is one of the highest amongst all manufacturing industries. Thus the environmental impacts of these associated manufacturing industries should also be considered when analyzing the ecological impact of a computer. Many developing countries do not have efficient recycling facilities for packaging and shipping of computers .In the point of disposal many developed countries have introduced laws for e-waste recycling but owing to the high operational costs, most e-waste ends up in developing countries where appropriate recycling facilities and stringent environmental laws do not exist. The author

proposed a solution for these environmental issues with the use of virtualization with Green. But again that the use of virtualization with Green also raises many research oriented questions for future investigation. [23]

Designers of software technology are responsible for the long term consequences of software designs. There is a perception that sustainability is a distinct discipline of research and practice with few defined connections to software where as sustainability is a pervasive concern that translates into discipline-specific questions in each area it applies [24]. Programmers should write efficient algorithms via writing a compact design of codes and data structures based upon the application, programming language, and the architecture of the hardware but optimization comes from only through experience.[25] (Re-)Designing business models and business processes according to the principles of sufficiency (instead of maximizing port) and decoupling of resources from economic growth.

2.2 Green Metrics

There are different approaches that are related to Green metrics of the software [26]. This paper pointed out the Green software metrics that are defined in the software engineering literature and also raised two research related questions on Green software engineering and Green metrics such as

- What Green metrics have been proposed in software engineering literature?
- How Green metrics can be classified?

This paper also pointed out many Green software metrics are used in the software engineering literature. A Green factor defines the Green software must fulfill the properties. It needs one or more Green metrics which measures the factor fulfillment in software for instance if we one Green factor stillness. [10] This paper pointed out how to obtain Green quality is determined. The work presented [32] the

quality assurance techniques corresponding to the requirements which are needed; it necessitates future research in establishing sustainability metrics as well as assessment techniques. In [7] Green soft model has the ability to represent three categories of sustainability criteria and metrics for software products they are (I) common quality criteria and metrics, (II) directly related criteria and metrics and (III) indirectly related criteria and metrics. Depend on the classification the quality properties “Modifiability” and Reusability” take effect in the development phase, where as the properties portability, supportability, performance, dependability, usability, and accessibility” take effect in the usage phase. In [27] a Green model for sustainable software engineering energy awareness in systems can be obtained and calculated through Green metrics such as the Green Performance Indicators (GPIs) found in [28,29,30,31] here GPIs are classified into four classes: IT Resource Usage GPIs that compute resource usage, the Application Life cycle KPIs (key performance indicators) that define efforts required to develop or redesign applications and reconfigure IT-infrastructure, the Energy Impact GPIs that represent the environmental impact of data centers, and the Organizational GPIs that describe organizational factors. [32] Green IT will happen only when all the stakeholders involved in delivering and consuming IT services are aware of the functionality of the service delivers and the relationships between the associated Quality of service levels and environmental-impacts metrics. The work [33] presented that Information Technology (IT) can be regarded as a special resource for business processes. On one hand, IT is useful for collecting, processing and providing information in order to determine and to improve ecological indicators for business processes from the perspective of “IT for Green”. On the other hand, IT itself requires natural resources for the design, manufacturing, use, and disposal of the required hardware and software (perspective “Green for IT”). In the work [22] Green metrics are used as a tool to measure the actual carbon footprint of SBA (Small

Business Administration). This makes measure an eligible keyword for ones search strategy. The work [34] presented a generic metric to measure software and a method to apply it in a software engineering process.

2.3 Requirement Engineering Perspective

Requirement engineering involves eliciting, analyzing, documenting and maintaining the complex set of requirements for a software system. [35] In this paper [35] the author raised the 3 types of research related questions about support of environmental sustainability by requirement engineering such as:

- How to make environmental sustainability a first –class quality objective for development?
- How does the necessity can be implemented in a requirement approach?
- How can one assess the impacts of a given software system for environmental sustainability, including both direct and indirect effects and considering different groups of stakeholders?

3. Tools

There are few tools that are used for finding out solutions to the Green and sustainable software engineering [2]. “Green Power Indicator” visualizes when a webpage is hosted with Green electricity and depicts the power quality. This tool is an add-on for Mozilla Fire fox. [35] To analyze the applications energy efficiency it is recommended to use a range of tools which are available to address power related frameworks ,optimizations and measurement for example Intel Power Checker, Perfmon, PwrTest, Windows Event Viewer, Windows ETW (Event Tracing for Windows), Power Informer, Power Top, and Battery life toolkit etc., [36] GREEN TRACKER it measures the energy consumption of software and focuses on comparing different systems that serve the

same purpose instead of comparing among different versions of the same system.[37]"Joule Meter" a tool that estimates the prorated power consumption of virtualized servers running on one hardware server.[38] The author introduced "Green Analytics" a tool that visualizes the impact of websites on the environment by using data from Google Analytics. Soft Watt [39] is a tool to estimate power used by OS and applications. [30] LABVIEW (WWW.NI.COM/LABVIEW/) is the tool acquires and stores samples of current energy consumption every 4ms (i.e.) with a sampling frequency of 250MHz) and computes the total energy consumed over time by interpolating current consumption sample. [20] The Java tool called Workload Simulator that can record a given flow of operations and execute it a given number of times for a given number of simultaneous users this tool will help to identify the performance of an application. These are all the tools that are helpful to initiate the Green software engineering and sustainability approach. Joule energy awareness tool is a lightweight monitoring tool that consolidates and visualizes energy to use information across networked devices.

4. Statement of the Problem:

From the literature survey the following research challenges are identified for the use of SDLC (Software development life cycle) techniques in the area of sustainable software using green technologies.

4.1 Requirement Phase

In the previous study of the work there is no high importance for the requirement analysis related to Green. If much concentration is given to this phase then throughout the life cycle it becomes Green. In this analysis phase one has to raise a question in the aspect of sustainable software engineering using Green technologies. [18].The author presented a case study on business information system for event management agency. In contrast the author provided modeling means explicitly for integrating

sustainability into the software development process as a major objective.

4.2 Design Phase

In traditional method of design phase the software engineer created a design which is based on the SRS (software requirement specification) to develop a product. In design phase almost all the activities are energy related and the most of the researcher focused on the energy consumption and energy efficient and other aspects only. But this won't give a correct solution for the Green design point of view. Before progressing to the development one has to create a mockup design that mock design itself has to include the sustainable software engineering and Green Technologies. Now there are a lot of open source mock up tools which are available on internet one can use that tool and create a mock design. That mock design after creation one has to show to the customer or to the management and to get their approval for the development, if one follows like this it will reduce the risk after development. In the mock up design itself one has to show how the application is efficient in consuming the energy and about all the parameters one used [29].It is necessary to integrate the consideration of sustainability aspects as early as possible into design process. This is because it is a common knowledge with software processes that late changes to the product design are much more expensive than early ones.

The paper [31] discusses the implication of evolving releases of a widely used text processor and relates these to Green IT and SD (software development) as a solution to cope with sustainability issues during software designing and development, they propose "Grand Management Information Design", which tries to transfer the Bauhaus design principles to immaterial software products[32].To preserve the world for future generations, there is a need for change in consumer behavior with regard to environmental impact. Where action is currently limited compared to other economic activities. From the past research activities regarding security and safety requirements it is shown to establish

sustainability requirements and thereby make the world as eco-friendly. In this nearly all the serious accidents that software has been involved with over the past 20 years can be traced to show the requirement flaws but not the coding errors. The reason is that the requirements usually reflect incomplete or wrong assumptions about the operations or context. Extensive investigation into specification and analysis of requirements for safety critical systems began in the 1990's, in the area of formal methods. A security requirement is a "manifestation of a high level organizational policy into the detailed requirements of specific systems. Quality assurance techniques corresponding to the requirements are also needed, which necessitates future research in establishing sustainability metrics as well as assessment techniques.

4.3 Development Phase

In developing countries and in the developed countries developers are not aware about Green and sustainable software engineering technologies of code creation. Each developer has their own idea of developing a code. In this one need to follow the universal approach of developing a code in the point of Green and sustainable software technologies related to the developing team and if the developer develop a code in this perspective it will give the solution for various issues which are related to energy consumption, and energy efficiency etc.,[1]The first challenge for modern computers system and a complex network capacitors and for electronic storage unit is an attempt to maintain a constant voltage regardless of what computation is being performed that is difficult to discern fluctuations in power consumption. The second challenge is synchronizing power consumption with program execution location [33]. As designers of software technology, they are responsible for the long term consequences of their designs. Design is the process of understanding the world and articulating an alternative conception on how it should be shaped, according to the designer's

intentions. Through design, one can make a revival and shape the environment. In this work researcher discusses about the sustainability applies to both a system and its wider contexts. There are at least two spheres to consider in system design:(i) the sustainability of the system itself and how it affects overall sustainability of the wider system of which it will be part of he again describe system visibility is a necessary precondition and enabler for sustainability design. Strive to make the status of the system and its context visible at different levels of abstraction and perspective to enable participation and informed responsible choice.[34] The energy efficiency would translate into better design by software's overall runtime efficiency, which will translate into lower energy costs thus it is resulted in creating a Green and sustainable software engineering in today's scenario

4.4 Testing

In this phase tester has a tool for testing the functionality of the application and logic of the application. They either by using a tool or by manually they tested the application and given a release. But they don't have a tool to conduct a test based on the Green and sustainable software technologies point of view. So testing community need a tool in the point of Green and sustainable software engineering [1]. A review and a preview meeting to be held as team facilitation approach after round about two thirds of an iteration. This enables the development team to correct and to found design and implementation which flows within the same iteration without the necessity to shift these to the next iteration [19]. This commoditization has resulted in increasing pressure to develop and to deliver great volumes of high quality product and services within the cost and schedule constraints that are tighter than ever before in the software industry [20]. The demand for higher performances and new usage models will also continue to grow energy efficiency and it will be crucial for the computing industry to b the increase battery life for mobile platforms and to reduce the energy expenses for desktop and server platforms.

4.5 Implementation

Implementation of the developed software have some difficulties already (i.e.) user need a proper training in user friendly, and in fast accessibility. But it is must to think in the sustainable software engineering and Green using technologies point of view as a result, one lead to think about all the factors (i.e.) Operating system, backup devices, user manuals, and maintenance of that application[18].Stake holders are the key persons in determining the objectives to be achieved. Identifying the stake holders for sustainability is crucial in implementing sustainability support successfully in a given context. All these works provided are helpful insights on stake holder identifications in other domain, but one of them mentioned that stakeholders are for sustainability in the area of software engineering [21].It is common experience that code written by an automatic interpreter is less efficient than code written by an experienced developer. Shannon's entropy framework is referred that entropy denotes the programming style of different developers.

5. Discussion:

These papers discuss about the advantages of the sustainable software engineering using Green technologies in SDLC and its

draw backs about the traditional SDLC. If one implement the sustainable software engineering and Green technologies in each and every step of SDLC. It will reduce the carbon emission in the development time, also increase the efficiency of the product without affecting the environment and give the sustainable software product. Finally in the traditional SDLC methods all the problems that are described in the above chapters are for the betterment taken into consideration as a serious issue and for further research.

6. Conclusion

This paper presented a traditional SDLC method, how it affects the sustainability of the environment and what are all the problems in each phase of SDLC. Create if one has not think in this point of sustainable software engineering without using the Green technologies definitely it leads to the environmental threat. So it is concluded that here after all the software development companies should try to implement the sustainable software engineering approaches using Green technology in each and every software development life cycle approaches. If they tried to implement like that it will reduce the environment impact for sure and also increase the quality of software.

References:

1. Dick, M.Drangmeister, J.Kern, E.Naumann, S., "Green Software Engineering with Agile Methods," Green and Sustainable Software (GREENS), 2013 2nd International Workshop on , Vol., no., pp.78,85, 20-20 May, 2013.
2. Johann, T. Dick, M.; Kern, E.Naumann, S., "Sustainable Development, Sustainable Software, and Sustainable Software Engineering: An Integrated Approach," *Humanities, Science & Engineering Research (SHUSER)*, 2011 *International Symposium on* , Vol., no., pp.34,39, 6-7 June, 2011.
3. Bener, Ayse Basar; Morisio, Maurizio; Miranskyy, Andriy, "Green Software," *Software, IEEE* , vol.31, no.3, pp.36,39, May-June 2014 doi: 10.1109/MS.2014.62
4. Stefan Naumann, Markus Dick, Eva Kern, Timo Johann, The GREENSOFT Model: A reference model for green and sustainable software and its engineering, Sustainable Computing: Informatics and Systems, Volume 1, Issue 4, December 2011, Pages 294-304, ISSN 2210-5379

5. Sierszecki, K.; Mikkonen, T.; Steffens, M.; Fogdal, T.; Savolainen, J., "Green Software: Greening What and How Much?," *Software, IEEE*, vol.31, no.3, pp.64,68, May-June 2014
6. Eugenio Capra, Chiara Francalanci, Sandra A. Slaughter, Is software "green"? Application development environments and energy efficiency in open source applications, *Information and Software Technology*, Volume 54, Issue 1, January 2012, Pages 60-71, ISSN 0950-5849,
7. Sahin, C.; Cayci, F.; Clause, J.; Kiamilev, F.; Pollock, L.; Winbladh, K., "Towards power reduction through improved software design," *Energytech, 2012 IEEE*, vol., no., pp.1,6, 29-31 May 2012.
8. Juha Taina-Good,Bad,and Beautiful –In Search of Green software quality Factors-CEPIS upgrade European journal Exploring Initial challenges for green software engineering
9. Penzenstadler, B.; Femmer, H.; Richardson, D., "Who is the advocate? Stakeholders for sustainability," *Green and Sustainable Software (GREENS), 2013 2nd International Workshop on*, vol., no., pp.70,77, 20-20 May 2013
10. DR. BOB STEIGERWALD AND ABHISHEK AGRAWAL SOFTWARE & SERVICES GROUP, INTEL CORPORATION, FOLSOM, CA, USA-"Developing Green Software"- Download the Developing Green Software-A generic model for sustainability with process and product specific instances .Developing a green software - on June 23, 2011
11. Patricia Lago , Niklaus Meyer , Maurizio Morisio , Hausi A. Müller , Giuseppe Scanniello, Leveraging "energy efficiency to software users": summary of the second GREENS workshop, at ICSE 2013, ACM SIGSOFT Software Engineering Notes, v.39 n.1, January 2014
12. Penzenstadler, Birgit; Raturi, Ankita; Richardson, Debra; Tomlinson, Bill, "Safety, Security, Now Sustainability: The Nonfunctional Requirement for the 21st Century," *Software, IEEE*, vol.31, no.3, pp.40,47, May-June 2018
13. Trent A. Kroeger, Neil J. Davidson, Stephen C. Cook, "Understanding the characteristics of quality for software engineering processes: A Grounded Theory investigation, *Information and Software Technology*, Volume 56, Issue 2, February 2014,
14. Dr. Bob Steigerwald And Abhishek Agrawal Software & Services Group, Intel Corporation, Folsom, Ca, Usa-"Developing Green Software"- Download the Developing Green Software white paper. Pages 1- 11- on June 23, 2011
15. Giovanna Sissa;"utility computing: green opportunities and risks",CEPIS UPGRADE, The European journal for the informatics professional,Vol.XII,issue no.4,October -2011 page no :16
16. Juan-Carlos López-López, Giovanna Sissa, Lasse Natvig : "Green ICT: The information society commitment for environmental Sustainability", CEPIS UPGRADE, The European journal for the informatics professional,Vol.XII,issue no.4,October -2011 page no :2
17. Carol-Ann Kogelman on behalf of the CEPIS Green ICT Task Force– CEPIS Green ICT Survey – Examining Green ICT Awareness in Organisations: Initial Findings , CEPIS UPGRADE, The European journal for the informatics professional,Vol.XII,issue no.4,October -2011 page no :6

18. Authors: Christoph Becker, Ruzanna Chitchyan, Leticia Duboc, Steve Easterbrook, Martin Mahaux, Birgit Penzenstadler, Guillermo Rodriguez-Navas, Camille Salinesi, Norbert Seyff, Colin Venters, Coral Calero, Sedef Akinli Kocak, Stefanie Betz-“ The Karlskrona manifesto for sustainability design- [v1] Sat, 25 Oct 2014.
19. Paolo Bozzelli, Qing Gu, Patricia Lago “A systematic literature review on green softwaremetrics”-
www.sis.uta.fi/~pt/TIEA5_Thesis_Course/.../SLR_GreenMetrics.pdf
20. Birgit Penzenstadler,Bill Tomlinson and Debra Richardson “RE4ES Support Environmental Sustainability by Requirements Engineering “-2012Journal of International Workshop on Requirements Engineering for Sustainable Systems
21. Petter Larsson, WHITE PAPER, Intel Software Solutions Group Energy-Efficient Software Guidelines, April 2011
22. Aman Kansal,A Feng Zhao,Jie Liu,Nupur Kothari,Arka A. Bhattacharya “Virtual machine power metering and provisioning”,Proceedings of the 1st ACM symposium on Cloud computing, 978-1-4503-0036-0,Indianapolis, Indiana, USA P 39-50,2010, ACM
23. Sara S. Mahmoud and Imtiaz Ahmad, A Green Model for Sustainable Software Engineering , International Journal of Software Engineering and Its Applications Vol. 7, No. 4, July, 2013
24. Christoph Becker, Ruzanna Chitchyan, Steve Easterbrook, Martin Mahaux, Birgit Penzenstadler, Guillermo Rodriguez-Navas, Camille Salinesi, Norbert Seyff, Colin Venters,Coral Calero, Sedef Akinli Kocak, Stefanie Betz “The Karlskrona manifesto for sustainability design”, Version 0.3, October 2014, <http://arxiv.org/abs/1410.6968v1>
25. Penzenstadler, B.; Raturi, A.; Richardson, D.; Tomlinson, B., "Safety, Security, Now Sustainability: The Nonfunctional Requirement for the 21st Century," *Software, IEEE* , vol.31, no.3, pp.40,47, May-June 2014
26. N Chitnis, R Bhaskaran, T Biswas – “Going Green With Virtualization” Vol 9 no:1 2011Green IT: Go Green for Sustenance, 2011 <http://www.infosys.com/infosys-labs/publications/documents/green-it/going-green-virtualization.pdf>
27. Alexander Kipp, Tao Jiang, Mariagrazia Fugini, Ioan Salomie, Layered Green Performance Indicators, Future Generation Computer Systems, Volume 28, Issue 2, February 2012, Pages 478-489, ISSN 0167-739X,
28. S. S. Mahmoud and I. Ahmad, “Green Performance Indicators for Energy Aware IT Systems: Survey and Assessment”, Journal of Green Engineering, vol. 3, no. 1, (2012), pp. 33-6
29. F. Albertao, J. Xiao, C. Tian, Y. Lu, K. Q. Zhang and C. Liu, “Measuring the sustainability performance of software project”, 2010 IEEE 7th International Conference on e-Business Engineering (ICEBE), (2010), pp.369-373
30. D. M. Raffo, W. Harrison and J. Vandeville, “Software Process Decision support: making process tradeoffsusing a hybrid metrics, modeling and utility framework”, Proceedings of the 14th ACM international conference on Software engineering and knowledge engineering, (2002), pp. 803-809.
31. Reiter, M.; Fettke, P.; Loos, P., "Towards Green Business Process Management: Concept and Implementation of an Artifact to Reduce the Energy Consumption of

- Business Processes," *System Sciences (HICSS)*, 2014 47th Hawaii International Conference on , vol., no., pp.885,894, 6-9 Jan. 2014
32. Atkinson, C.; Schulze, T.; Klingert, S., "Facilitating Greener IT through Green Specifications," *Software, IEEE* , vol.31, no.3, pp.56,63, May-June 2014
 33. Johann, T.; Dick, M.; Naumann, S.; Kern, E., "How to measure energy-efficiency of software: Metrics and measurement results," *Green and Sustainable Software (GREENS)*, 2012 First International Workshop on , vol., no., pp.51,54, 3-3 June 2012
 34. Penzenstadler, Birgit; Raturi, Ankita; Richardson, Debra; Tomlinson, Bill, "Safety, Security, Now Sustainability: The Nonfunctional Requirement for the 21st Century," *Software, IEEE* , vol.31, no.3, pp.40,47, May-June 2014
 35. Torsten Wilde, Axel Auweter, Hayk Shoukourian "**The 4 Pillar Framework for energy efficient HPC data centers**", Computer Science - Research and Development August 2014, Volume 29, Issue 3-4, pp 241-251,
 36. Karin Kandananond, A Roadmap to Green Supply Chain System through Enterprise Resource Planning (ERP) Implementation, *Procedia Engineering*, Volume 69, 2014, Pages 377-382,
 37. carol-Ann kogelman "Examine green ICT awareness in organizations: Initial Findings" CEPIS UPGRADE, The European journal for the informatics professional, Vol.XII, issue no.4, October -2011 P:6-10
 38. Lorenz M.Hilty and wolfgang lohmann" The Five most neglected Issues in "Green IT" CEPIS UPGRADE, The European journal for the informatics professional, Vol.XII, issue no.4, October -2011 P:11-15
 39. Dick, M.; Drangmeister, J.; Kern, E.; Naumann, S., "Green software engineering with agile methods," *Green and Sustainable Software (GREENS)*, 2013 2nd International Workshop on , vol., no., pp.78,85, 20-20 May 2013