International Journal of Knowledge, Innovation and Entrepreneurship
Volume 4 Nos. 1-3, 2016, pp. 54—65

Green Software Engineering with Agile Methodology

SAMRA SIDDIQUI, MARIAM KHAN, HINA ABRAR &
NADEEM MAJEED CHOUDHARY
University of Engineering and Technology Taxila, Pakistan
Received 07 August 2015; received in revised form 27 February 2016; approved 30 April 2016

Abstract

Sustainable development holds strong impact in the field of Information Communication Technology (ICT) because of increasing demands for resources and energy which are required to build software and hardware products. There are several hardware solutions for Green IT. Numerous concepts of hardware resolutions, involvement of software in the field of Green IT has not till better examined. It consists of production as well as impact of using software in the field of energy utilization. Proficient software will ultimately use less amount of energy through utilization of fewer equipment of hardware for executing. In our paper, we argue that involvement. Especially, a model is presented in our paper for integrating Green IT features in the process of software engineering with agile methods to construct "greener" software from scrape. We will also give details regarding software itself may also be use as a tool for supporting green computing via monitoring reserves in an energy resourceful way.

Keywords: Green Computing, Green ICT, Green and Sustainable Software Engineering, Agile Methods

Introduction

With rising demand for complex applications of software, Information and Communication Technology (ICT) has largely negative impact on environment because of its rising resources plus expenditure of power. In the area of green computing, sustainable software development is the most popular subject. Sustainable development is taking the economic, ecological and communal impacts of the society for the utilization of resources to meet requires of individuals. Though ICT is trying to discover well-organized resolutions for atmosphere, it is not obvious whether energy resource and energy economy by ICT would go beyond its own consumption of resources. Features of software are accountable for emissions of CO_2 the same as components of hardware. Software has no direct consequence on atmosphere via managing and operating the hardware that is executing underlying. Some solutions that are software based may consume and examine resources proficiently as well as further solutions may be sustainable sufficient for limiting hardware resources but would add extra hardware owing to revises. Unluckily, there are not enough proposed research models in the field of green software engineering.

In social, economical and environmental crisis that progressed in the recent century there is an ultimate need of a manipulative and profligate mode of life for potential generations to be alive in the globe. The social societies require producers for incorpo-

rating sustainability principles to plan products as well as manufacture these artifacts in an economical mode.
Agile software management strategies are becoming significant with time due to the low observance and frequently shift problem for the development of software product from processes anywhere in a rising software organization. Therefore, the development process becomes out of the sight of developers, mainly for the development team to take the responsibility of integration in their own design decisions.

To face these challenges, this paper proposes a modification of development processes of software by means of agile extension to aim at an untimely combination of issues of sustainability in random software growth processes. It places a managerial frame to allow development of software team for taking decisions related to the design process having straight impact on environment concerning hardware and also possess communal impacts of software invention. Furthermore defines promotion of green as well as sustainable process of software engineering by software tools.

Literature Review

Multiple aspects for the representation of green software engineering are covered in GREENSOFT Model (Agarwal, 2012), which is an abstract orientation model that supports IT professionals and common users in the sustainable software development. The four parts of GREENSOFT model are software product life cycle, metrics and criteria, process models, as well as tools. The social, ecological and environmental aspects of software in the Life cycle of software products are more than the entire life of the product. Tischner, Dietz, Maßelter, Schmincke, Prösler, Rubik and Hirschl (2000) argued that in accordance to Thinking of Life Cycle, the ultimate motto is followed "from cradle to grave". The subsequent phases of life cycle of software product includes: Development, allocation, achievement, exploitation, Usage, preservation, deactivation, and removal. (Capra, Formenti, Francalanci and Gallazzi, 2010). The impacts and effects of software invention on sustainable growth of software are being evaluated based on some metrics and criteria. (Dick, Kern, Johann, Naumann and Gülden, 2012).To compute these criteria, an effective quality model has been introduced for green and sustainable software. To regard metrics, it is vital to relate methods of measurement, such as to calculate, approximate, as well as rate energy utilization of ICT (Kern, Dick, Naumann and Guldner, 2013). Procaccianti, Vetro, Ardito, and Morisio (2011) argued about some instances to measure the energy expenditure of software. Such type of models is included in orientation model for examining the growth, administrating, and purchasing as well as software usage. Shenoy and Eeratta (2011) present other advancement of green software development. Additionally, tools, suggestions and recommendations for the support of vast variety of diverse stakeholders are also integrated in the Model of GREENSOFT (Agarwal, Asoke and Dipayan, 2012).

A number of additional ecological performances in growth as well as accomplishment of systems of software considered entire software development (Albertao, Xiao, Tian, Lu, Zhang and Liu, 2010). Software is "environment friendly", Metrics is being introduced to assess in genuine project of software. The ultimate concern is to improve feasibility of software projects regarding sustainability measurements via metrics over frequent iterations.

Affiliation between Sustainable Development of Software and ICT

Software products assist to optimize usage of energy as well as expenditure within ICTs. Thus the consideration of green software products would be really insufficient; software should utilize and monitor resources proficiently. The impact of Sustainable development on ICT might be negative or positive. Consequences of ICT might be revised into impacts of three orders into which one impact is result of preceding impact. An impact of second order is the result of impact of first order and impact of third order is the result of impact of second order. For identifying ICT impacts on sustainable development, contributions are followed as it covers social, human and environmental sustainability issues. Performance requirement is resulted in first order impacts that are most obvious, product covering and bandwidth of network. Impacts of second order are those consequences that is outcome of using ICT indirectly i.e., those effects that have been appeared on other products life cycle by using services that ICT provide.

Currently ICT offers services via software. Software is also being utilized to optimize design, construction, and discarding of product that is being manufactured. Hence it is clear that software is taking part in the process of software engineering. Impact of third order, famous for difficulty in assessment and prediction, have long lasting indirect environmental influences resulted from systematic ICT effects.

Many descriptions are being presented for sustainable and green development software and there is no standard description. Green and Sustainable Software Engineering are to develop green and sustainable software engineering product which utilizes minimum energy resources. So, it defines and develops products of software in such a way that positive and negative impacts on sustainable development are assessed continuously. Manuscripts and metrics are used for auxiliary optimization of software products.

Green and sustainable software effects on society, economy, environment and human beings that are resulted from growth, operation, and software usage possess positive impact on sustainable development. Our work in this paper is bestowed mainly to create ecologically environment friendly software processes that are being recounted towards software processes. Software process founded in ISO/IE312207 standard is the compilation of wide-ranging definitions of activities that are interrelated and might be performed correctly during software product life cycle. For obtaining software sustainable product whichever process that is causative towards its life cycle ought to be sustainable. Green and Sustainable Software Process: That Software that congregates its objectives of sustainability, often articulate in expression of indirect and direct effects on economy, society, economy ,environment and human beings that are being resulted from its classification as well as operation.

As novel technologies are invented with the progressions in the field of software engineering life novel advancement to development of software life cycle that might promise further process of sustainable development that bring great concentration. This novel advance process development is called Agile Software Development (ASD). This standard is brought by 17 professionals of software that are involved in making this innovation that depends upon top practices as well as preceding skills along with frequent software development tasks.

Some set of principles are followed by every agile methodology as diverse range of methodologies are adopted. Such principles endorse mostly close teamwork between business teams and software development, head to head communication, un-

timely and recurrent increments delivery of software and to accept altering prerequisites by clients. Agile is diverse from mature as well as conventional ones in such a way that do not depend on deep printed requirement manuscripts that are not accepted to some changing requirement, does not avert customers and business people for working hand in hand along with developers of software, and it also endorse sustainable development by considering that users, developers, as well as supporters ought to uphold working in even pace. For instance, it promotes lean development via evolutionary and iterative advance.

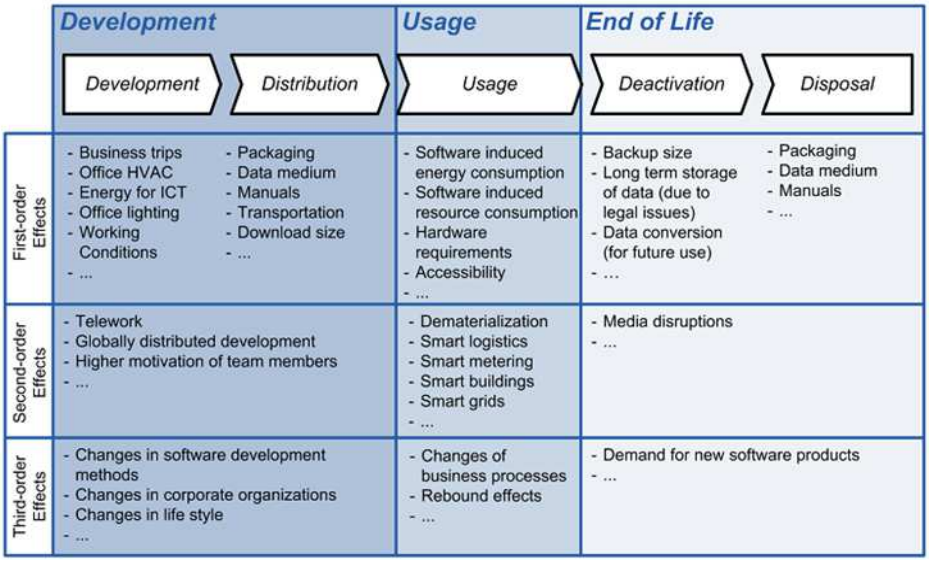

Fig. 1 Life Cycle Thinking inspired life cycle for software products (revised to comply with international standards footprints on carbon)

Approaches for Green Agile and Sustainable Software Engineering

Software Process enhancements for sustainability of agile

An innovative tool to offer clues on actors of impacts and effects is offered by the life cycle of Agile Development process model that should be reflected on development processes of software. Generally, enhancements comprise of two nonstop enhancement cycles: one meeting cycle is to study impacts and effects of software process of development itself, while other focal point is on effects that have been resulted from usage as well as distributing software product. Both enhancement cycles are driven actively throughout development process of software. Though designing, decision making plus manufacturing the objects parts of software artifact such as enhancements do not cover printed guides and are not here for design of product life cycle that comprises development life cycle of software. Development team design, develop and test as well as improve product of software regarding usage stage that orient issues of sustainability. Sustainability executive organize sustainability tasks of centric as well as incidents of development process of software and assess impacts of general design and development process. Enhancements bring small process of ceremony that is vigorous

in methodologies of agile software development. A general idea of process is shown in Fig. 2.

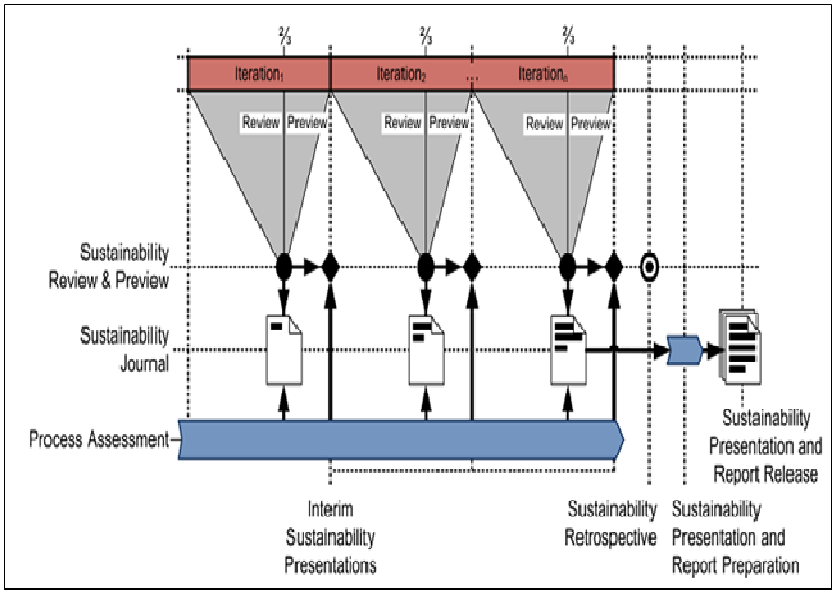

Fig. 2 Process Enhancements for Green and Sustainable Software Engineering

Process evaluation

The improvement cycle is implemented in process evaluation which concentrates upon all impacts of software processes. Sustainability executive conducts this process. The ultimate purpose is life cycle assessment or performing calculations of carbon footprint, so the collected data should be extensive enough. The information about collected data is provided in the development phase (see Fig. 1). This phase of the process should begin when plan for the product is developed, perhaps with start of the preliminary phase of product design life cycle.

Process Reviews and Previews for sustainability

The implementation of improvement cycle which focuses on the effects of distribution and usage of software product is carried out in this phase. This phase is directly carried out by development team and its major cause is non-functional requirements (e.g. efficiency). As far as the distribution phase is concerned, developers might have an impact upon the total size of the downloaded product but they might not have any effect upon the utilized download infrastructures or update strategies. The meeting of Review and Preview is carried out as a team approach and it should be after about two-thirds of iteration. In these meetings, the team can find flaws and mistakes in design or implementation stages in the same iteration and need not of shifting them towards next iterations. The executive should be a participant of meeting as well as encourage the development team for conducting such reviews. However, schedule and conduction of meetings is expected to be carried out by development team.

As suggested by its name, two more or less distinct parts are here. The total work done concerning sustainability issues is reviewed in review section. The solutions addressing discovered issues are proposed and its degree of success is evaluated in the preview section. Tools and methodologies helpful in this phase are runtime measurements of efficiency and performance, code reviews, energy efficiency measurements, other sustainability metrics and calculation of hardware obsolescence metrics.

Sustainability Journal

Sustainability Journal combines both of the Process evaluation and Reviews and Previews for sustainability. It can be thought as an information hub of process enhancements of agile methodologies. The influences and impacts of software projects with material product parts might also be documented in this journal. However, this journal tends to be a short report but not an extensive one.The maintenance and continuity of the journal is the ultimate responsibility of sustainability executive, as well as the development team due to the fact that the results and out comings of process evaluation and Reviews and Previews are documented here.

Provisional Sustainability Presentations

For decreasing negative effects and increasing positives which might arise by using the product, development team present reports at the end of an iteration, to the representatives of customers documenting the steps taken during iteration. Usually this report includes discovered problems during sustainability Reviews and Previews, solutions developed and the extent of success of solutions. The potential effects that result from the software projects such as greenhouse emissions, consumed person days, etc. are reported by sustainability executive.

Final Presentation of Sustainability and Report Release

The task of preparation of sustainability presentation is carried out at the end of software development process, by sustainability executive. He prepares final report from the provisional presentations and Journal of sustainability.
The executive also delivers the final presentation on sustainability and event of report release, where report is delivered to the customer representatives in closing ceremony of formal process. The development team might be there in the occasion.

Sustainability Retrospective

Sustainability Retrospective is performed by the development team and sustainability executive after the software project has finished with the aim of improvement in the sustainability of upcoming projects. This team facilitation approach is very useful and its probable outcomes are the learned lessons, the best approaches concerning sustainability problems of software or development processes, judgment of upcoming projects, combined thoughts on impacts, and effects of software or development processes. The results and conclusions should be saved in a knowledge base or handbook of process for upcoming projects.

Going agile: enhancing scrum with green and sustainable software engineering

The enhancements of sustainable and green software engineering do not provide a full description of software development process. Hence, its integration with software process is a necessity. All the actions, tasks and activities of enhancements should not be necessarily visible or implemented as separate actions for the resulting process. They should have a major contribution in tasks and actions of the immediate software engineering or development process. The ultimate goal of Scrum methodology is that, in every iteration, a potentially shippable increment of the software product is delivered. The model consists of different iterations known as Sprints. At the start of the Sprint, a Planning Meeting is held and a Review Meeting and Retrospective is held at the end. The potential features to be implemented in the upcoming sprints are decided in the Planning Meetings by the development team and Product Owner. During Sprints, development team held a short standup meeting (max. 15 min) known as Daily Scrum. In this meeting, all the developers explains the activities they worked out since the previous daily scrum, the tasks planned to be done until next one and the obstacles faced by them that potentially delayed ongoing progress in work. The potentially shippable features are presented by development team during Sprint Review Meetings. Features not considered to be potentially shippable are incomplete and shifted towards upcoming sprint. Product owner accepts or rejects the complete features (potentially shippable). Then, the completed features can be put into operation afterwards.

At the end of Sprint Review Meeting, whole team meets up for performing Sprint Retrospective. Its ultimate purpose is improvement of development process of software product, teamwork, and discovery of obstacles that slows down development process. The Retrospective is conducted in the form of team meeting. The team argues and evaluates the previous Sprint and finalizes changes to be implemented in the next sprint. Figure 3 shows suggestions on the integration of Green and Sustainable Software Engineering Processes into Scrum.

After the completion of two-third of the Sprint, the Reviews and Previews concerning sustainability should take place. Process Evaluation is also carried out as mentioned before. The integration of provisional Sustainability Presentations with Review meetings should be carried out. In this scenario, most of the actors participate: Product Owner performing the Customer Representative's role as well and the development team. In addition, the Executive attends the resulting combined meeting and give reports. The Product Owner either agrees to or rejects exercised features and the steps taken for the improvement of sustainability of the software product. So, this meeting is vital to the agile process.

The Sustainability Retrospective is carried out just prior to the ending of previous Sprint Review. This enables the team to document and provide report to the Product owner about the combined assessment results. The Sprint Retrospectives should not be combined with the Sustainability Retrospective as they are performed at a different time concerning process flow or focus. On the other hand, Sprint Retrospectives are carried out after every Sprint and focus upon the hurdles of last Sprint and improvement of upcoming Sprint, Whereas, Sustainability Retrospective emphasizes on preservation of experience of the contributing team about problems of sustainability for upcoming products. This enables the team to agree and discuss different aspects without getting pressure and other problems in mind that would otherwise lead to poor results which are unacceptable in any case.

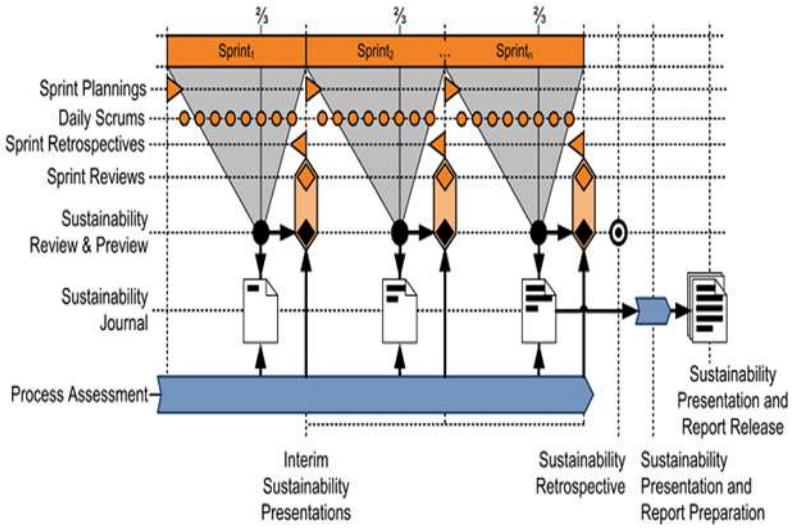

Fig. 3 Integration of enhancements of Sustainable and Green Software Engineering into Scrum

Green Analysis Stage

For promotion of energy efficiency, this stage is carried out at the completion of every increment of the agile methodology. New ideas about environmental sustainability are brought out by this stage which has not been known before to be added to the software processes. It evaluates greenness of every developing increment of this process. This stage tests for energy efficiency of the process or methodology. For analysis in this stage, metrics are utilized here. As the criteria of measurement, metrics of IT resource usage e.g.; CPU usage and the metrics of Quality e.g. performance from the Green Performance Indicators could be utilized. For this stage, we propose the process illustrated in Figure 4. The process starts by defining quality and IT resource metrics utilized by the system, and then continues by collection of data and information from software and measuring using the formulas by EU Project GAMES. Numerous tools can be utilized during this stage of the process for the collection of material from core hardware and for mapping wastage of energy with those locations of code. The results are analyzed using formulas and probable changes are determined in the third stage. The eco-related rules and laws that define some guidelines which put up standards regarding levels of energy systems should tolerate for determining them. The changes needs to be implemented based on the results are exercised in the final step.

Figure 4. The process of Green Analysis

Role of Software tools in energy efficient software applications

For the promotion of green computing, software tools have a vital impact regarding energy efficient utilization of various software applications. The five categories of software concepts and tools are defined in Figure 5. On computer systems, as opposed to hardware, the software gives a greater impact in increasing environmental sustainability. As an example, power-efficient software can be attained by the utilization of software and monitoring counters for quantifying dissipation of power in various pieces of code.

Another approach of utilizing software for management of the utilization of resources of hardware is the application of features of power scheme in fresh operating systems. After the review of various concepts of software such as fine tuning, software approaches are categorized into five types: fine grained green computing, operating systems frameworks, codes written for energy allocation purposes, and virtualization and performance monitoring counters and metrics. Concerning operating system, different frameworks are projected for running application's energy monitoring. For example, for the minimization of power dissipation of systems, the frameworks which make intelligent power profile can be inserted in code of operating system. Applications which are not used from some instant are shut down or hibernated which results in less dissipation of heat and minimized power consumption as well as minimizing CPU's average work load. Some other frameworks depend upon operating systems for reporting energy dissipation of processes with the help of data gathered from the hardware or the operating system dynamically. The information about CPU or network usage is collected using sensors.

The power analyzers for software rely upon the performance monitoring counters and information about frequency from CPUs for quantification of power consumption on systems is among some other used approaches. Such approaches such as SPAN can associate estimate of power with source codes of applications via API calls. An additional approach is the GREENTRACKER, a software tool that measures dissipation energy of software with emphasize on comparison of various computer systems with same intention rather than comparison between diverse versions of similar systems. Some other approaches are there which map design of software with the measurements of dissipation of power of audio, CPU, Ethernet core, video, and memory. All of the approaches in this group have goal to provide software designers vital information regarding power trends of system for better results concerning energy dissipation levels.

The approaches concerning fine grained green computing tends to be much more particular about running applications in the scenario of not utilizing hardware e.g.; I/O peripherals which is not in use by application contrasting coarse grained computing that has no contribution towards energy efficient utilization of computing resources of processes. Some other general approaches can be the codes documented particularly for the purposes of energy allocation that might fruitfully route traffic towards the locality such as the data centers having least costs of energy. Another approach is the concept of virtualization that plays vital job in green computing ideology and is somewhat software. It is operating system based upon partitions that enables execution of multiple applications on single system. Number of needed systems and the power needed for green computing is reduced in this way.

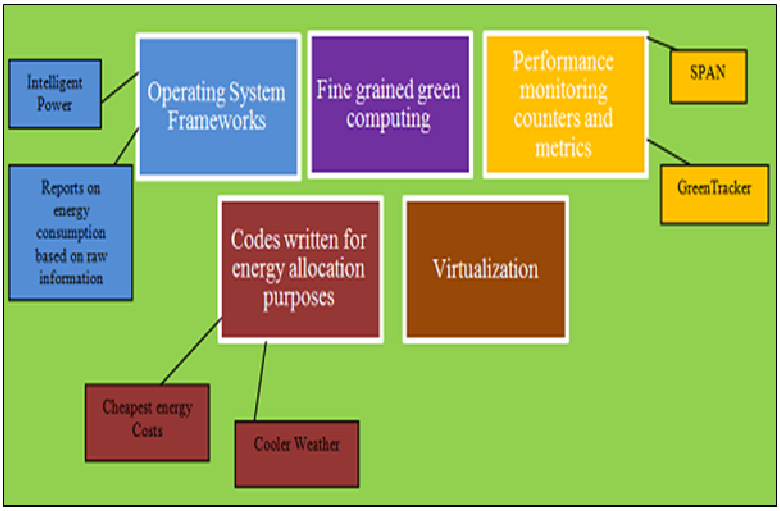

Figure 5: Tools used for sustainable development

Conclusion and Future work

Software has an indirect but vital effect on the environment just as hardware which contributes to direct impact. The development process of software from requirements to delivery should be performed to save energy and new greener software engineering processes must be produced so that the limitations of energy should be fulfilled. It would lessen the negative consequences of ICT towards green and sustainable development. Software tools which monitors and tracks different resources in energy efficient way and participate in features of power scheme should be utilized. In this paper, sustainability aspects are incorporated in the development process with the presentation of enhancements in generic process model as well as integration of enhancements of Sustainable Software Engineering with Scrum. In this way, software engineers have been encouraged in the development of sustainable and green software. This is evident that in the absence of measurement, monitoring, and organization of these aspects, no knowledge would be there regarding the performance of software regarding these issues. We also propose various approaches carried out by software itself for contribution in green computing. We categorized these approaches and conceptions into five major categories but the impact of carbon footprint is not measured in our work.
In the future work, a detailed model is to be developed and assessed for measurement of the carbon footprint for software production.

Correspondence
Samra siddiqui
University of Engineering and Technology Taxila
Pakistan
Email: samrasiddiqui7@yaho.com,

References

Agarwal, S., Asoke, N. and Dipayan, C. (2012). Sustainable Approaches and Good Practices in Green Software Engineering, *International Journal of Research and Reviews in Computer Science (IJRRCS)*. (3) .p.1425-1428.

Albertao, F., Xiao, J., Tian, C., Lu, Y., Zhang, K. Q., and Liu, C. (2010). Measuring the Sustainability Performance of Software Projects, *IEEE 7th International Conference on e-Business Engineering, Shanghai, China*. p.369–373.

Capra, E., Francalanci, C., and Slaughter, SA. (2012). Measuring Application Software Energy Efficiency. *IT Professional*. (14). p.54-61.

Capra, E., Formenti, G., Francalanci, C., and Gallazzi, S. (2010). The Impact of MIS Software on IT Energy Consumption, *18th European Conference on Information Systems,* Pretoria, South Africa.

Dick, M. and Naumann, S. (2010). Enhancing Software Engineering Processes towards Sustainable Software Product Design, *Proceedings of the 24th International Conference on Informatics for Environmental Protection*. Cologne/Bonn, Germany. p. 706–715.

Dick, M., Kern, E., Drangmeister, J., Naumann, S., and Johann, T. (2011). Measurement and Rating of Software-induced Energy Consumption of Desktop PCs and Servers, *Proceedings of the 25th International Conference Enviro Info*. Ispra, Italy. p. 290–299.

Dick, M., Kern, E., Johann, T., Naumann, S., and Gülden, C. (2012). Green Web Engineering- Measurements and Findings. *Proceedings of the 26th International Conference Enviro Info*. p.599–606.

Dirlewanger, W. (2006). Measurement and rating of computer systems performance and of software efficiency. *An introduction to the ISO/IEC 14756 method and a guide to its application*. Kassel: Kassel University Press.

Johann, T., Dick, M., Kern, E., and Naumann, S. (2012). How to Measure Energy-Efficiency of Software: Metrics and Measurement Results. *Proceedings of the First International Workshop on Green and Sustainable Software (GREENS)*. Zurich, Switzerland: IEEE.

Kern, E., Dick, M., Naumann, S., and Guldner, AJ. (2013). Green Software and Green Software Engineering – Definitions, Measurements, and Quality Aspects. *In Proceedings of the First International Conference on Information and Communication for Sustainability*. ETH Zurich, Switzerland.

Naumann, S., Dick, M., Kern, E. and Johann, T. (2011). The GREENSOFT Model: A reference model for green and sustainable software and its engineering, *Sustainable Computing: Informatics and Systems*. (1). p. 294-304.

Amsel, N., Ibrahim, Z., Malik, A., Tomlinson, B. (2011). Toward sustainable software engineering: NIER track. *33rd international conference on software engineering (ICSE)*. p. 976– 979.

Procaccianti, G., Vetro, A., Ardito, L., and Morisio, M. (2011). Profiling Power Consumption on Desktop Computer Systems. *Information and Communication on Technology for the Fight against Global Warming*. p.110-123.

Shenoy, SS. and Eeratta, R. (2011). Green software development model: An approach towards sustainable software development. *In India Conference, Annual*, IEEE.

Taina, J. (2011). Good, Bad, and Beautiful Software - In Search of Green Software Quality Factors. *CEPIS UPGRADE XII*. (4). p.22–27.

Tischner, U., Dietz, B., Maßelter, S., Schmincke, E., Prösler, M., Rubik, F. and Hirschl, B. (2000). How to do EcoDesign? A guide for environmentally and economically sound design. Verlag form, Frankfurt am Main. *GREENS '1 Proceedings of the 2nd International Workshop on Green and Sustainable Software.* IEEE Press Piscataway: USA.

Wang, S., Chen, H., and Shi, W. (2011). SPAN: A software power analyzer for multicore computer systems. *SUSCOM*. (1). p.23–34.