Indian Journal of Science and Technology, *Vol 9(32), DOI: 10.17485/ijst/2016/v9i32/88499, August 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Green based Software Development Life Cycle Model for Software Engineering

M. Mohankumar[1]* and M. Anand Kumar[2]

[1]Department of Computer Science, Karpagam University, Coimbatore - 641021, Tamil Nadu, India; Mohankumar07@gmail.com
[2]Department of Information Technology, Karpagam University, Coimbatore - 641021, Tamil Nadu, India; anand2kumarm@gmail.com

Abstract

Objectives: The main objectives of this research work are to propose a new green based model for the Software engineering with no impact in the environment. A**nalysis:** In the beginning stages, more importance is given to ICT manufacturing companies and software developing companies "in Hardware and software. They don't think about the sustainability of resources used for the manufacturing of a product or development of an application from the statistical analysis it is identified that there is a need to minimize the energy utilization relate to CO_2 emission of IT equipment. **Result**: The experimental results are verified that the proposed model out performs the existing software engineering model with little impact on the environmental considerations. **Improvement/Application:** Experiments revealed that the proposed Green Based SDLC technique is able to reduce the power consumption.

Keywords: Capability Maturity Model, Environment and Development, Green Power Indicator, Green Tracker, Non Function Requirements, Sustainable Software Development Life Cycle

1. Introduction

In the electronics world, major challenges are the environmental impacts which is one of the type of noise, heat and power consumption. These aspects affect the environment[1] which give unenthusiastic impacts on the financial system, culture, humans and atmosphere that the result in improvement of the operation and practice of the software. In the software maturity there exists several methods, and each method has its own environment and sustainability risk. But 95% of the research work mainly concentrate on the energy competence and consumption of energy[2]. For illustration it has been predictable that the power expenditure of data centers in the US are greater than 30 billion (Kilowatts per hour) in 2001 and it increased to 72 billion (Kilowatts per hour) in 2009 in an international scale and a comparable progress has been documented for the power utilization which increases from 62 million (Kilowatts per hour) in 2003 and 134 billion (Kilowatts per hour) in 2008. Further the global ecological danger was[3] the CO_2 emissions that accomplish 9.4 billion tones in a year which is the higher rank in the individual times gone by 51% advance than in 1991. Mostly 3 % of worldwide CO_2 release is ascribed to the information technology systems and auxiliaring increases that anticipated as the modern information technology systems which are increasing day by day. So it is necessary to minimize the energy utilization and relate to CO_2 emission of IT equipments. The issues mentioned previously are more or less associated to hardware related or software aspects that persist to the influence, and all the factors of the atmosphere in constantly varying forms and leveraging of the available system is a demanding job in numerous software and hardware organizations.

2. Problem

Presently, there is some idea on how the software development people can contribute to sustainability, but there

*Author for correspondence

are no approaches found throughout the SDLC. How are such procedures and ideas needed to follow in each and every process involved in the SDLC phases? In the SDLC phase, how is the process related to CMM level aspect? This work focuses on the above mentioned two research questions

3. Contribution

GBMSSE model (Green Based Model for Sustainable Software Engineering) provides the solution for the above mentioned research questions. How does the SDLC phase relate to CMM level and sustainability of a software engineering process? The approach shows the aspects which are supported by the current ways of development team in the software development life cycle.

4. Related Work

There is a methodical literature review on various areas in the green and sustainable software engineering but so for no study has been carried out the investigation to track the relationship between SDLC phase and Green and sustainability based approach.

The research on Green computing was First submitted at the WCED (World Commission on Environment and Development) progress in 1987[4]. The report indicated that the green development as "growth that meets the needs of the current devoid of negotiating the capability of prospect next generations to get there has possession of requirements". The Figure 1 shows the Green Software reference model and the work consist of a holistic life-cycle model for the software products, sustainability criteria and metrics of software products, a "Green Software Engineering" procedure model, and recommendations for actions as well as tools. Here the work does not consider the entire life cycle of software Development process, the study taken to consider the development, usage, and End of life but the remaining process is not considered.

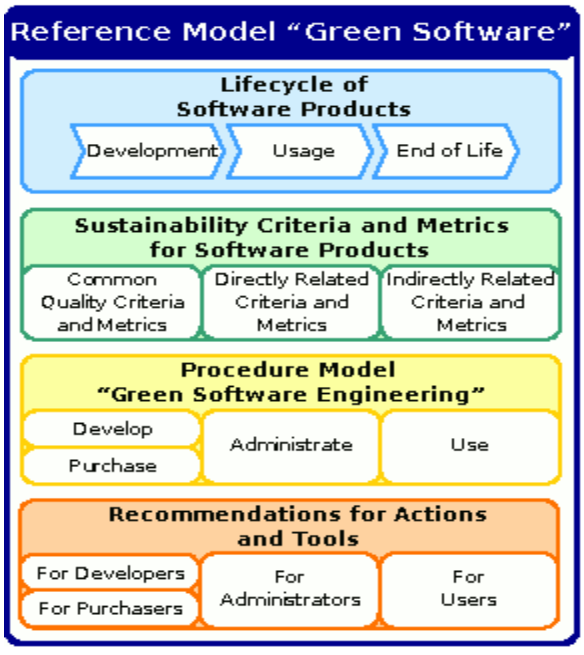

Figure 1. Reference model for green software engineering process.

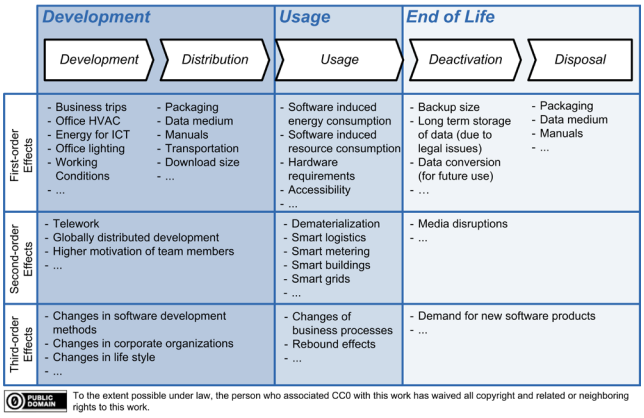

Figure 2. Overview of the life cycle of software.

Figure 2 depicts an overview of the life cycle of software and its relationship to the different levels of effects. It quickly becomes an illustration for the software industries. The study works and focuses on software green safety and green security in the early step and currently focuses on reducing the garbage waste or raw resources and hoard the atmosphere which is the perception of green software that becomes more and more complex[5]. The software, hardware and network related aspects are enclosed by the green information technology concepts which are published especially on the software and hardware sustainability of the data centers. Such as, the behavior of code for the data center efficiency[6]. In the early design stages not much awareness about the implementation techniques and software supporting tools to distinguish the inference of power consumption[7] are found. This scenario places a tremendous pressure on the technological organizations, internally and externally. And they need to develop an eco friendly and anticipated to propose new-fangled products which can get better sustainability in the

world at large[8]. The excellence related to the commercialization has resulted serious growth stress in the software industry to build up and bring bigger volumes of elevated superiority products and services related within the stipulated charge and time restraint that are tighter than ever before[9]. The manuscript[10] shows the power decline can be obtained by enhancing software design which points out the responsibility of the software developers in tumbling the energy utilization of the software function. This work[11] outlines the significance of categorization for the Green the excellence metrics and describe the concerning on the Green aspects. There are several methods which tend to blemish the contest mainly on the requirement, quality and design.

1. In what way the sustainability varies from one method to other method?
2. What are the several methods of prerequisite convention technique that tends to the sustainable software solutions?
3. what are the difference of conventional and Non Function Requirements on the requirement viewpoint?

The research work proposed an Energy frenzied by the electric utilities which was typically calculated in kilowatt-hours (kWh), and it meant which means the energy was being used at a stable rate (1 kW power) over a period of time (1 hour). In the following examples, the power utilization was given in kilowatt- years (kWy), meaning that an aspect rated at a power of kilowatt was being used over a period of one year.

In the previous years, the software sustainability considers only on the safety as a developing method that happens when the system tools interact inside the situation. The underneath sustainability requires the end user as the solution confront and in victory aspect for all the projects developed by the green software. A set of attempt was made to overcome the warfare in the greenhouse gas concerns particularly in the EU (European Union). It restricts the huge quantity of emanation in every nation and may be operated in the shape of free consent. Inventions are talented to keep acclaim which will not be permitted to go beyond the frontier. However if the mechanism like to release more gas, it should pay to regulate and find credit, therefore not only in the carbon footstep or perilous essence but also in the ecosystem trace that to be observed directly by the manufacturers. This technique has to be positioned in the software progressing industries. Green Performance Indicators take part in a key role in developing the more number of energy efficient data centers. Even though many metrics have been defined to measure all the factors which waste the energy, there is no typical or well known framework is found till now to define the "greenness" of a data center and it will become an important study area in future[12].

The current Software industry plays an essential role in creating the bounce back effects. The customary reply from the software engineers is to add the dispensation power and storage space capacity obtainable at a given price and confine the same. This work mainly focuses on the green methods such as

In what way the product to be implemented for the dissimilar user stage weight and linked either ecological risk?

What is the mounting substance of cloud work out the query is not whether it is green but how it can be truly green?

Wakefulness and answerable behaviors are the surrounding conditions to attain the sustainable and green cloud computing.

Software engineering has a considerable potential to support the "greening through IT"—that is, making civilizations more environmentally sustainable via IT interventions. To draw the attention to such issues in software engineering, the study argues that the sustainability must be treated as a first quality alongside the other critical attributes such as safety, security, efficiency, reliability, and usability[13]. More than one third of the software industries in Europe do not apply the green IT practices main and important cause is that there is no official law in their nations to enforce the green IT practices. Minor amounts of one fifth of the software industries really checking the ways of minimizing the employees power consumption[14]. The manuscript empirically analyzes a conceptual integration model for the energy consumption of business processes, applications and IT devices is introduced and prototypically implemented[15]. By applying concept and software, the energy consumption of business processes in office settings can be detected and also made transparent. Furthermore, the prototype enables the deduction of energy saving measures which can directly be derived from the energy consumption profiles of the applications which are used by the business process, but also the Management Information System function cover does the impact power consumption up to 92% and the different

Management Information System software's are fulfilling the similar efficient requirements to consume the different amounts of energy and in some situations energy efficiency cannot be improved just by humanizing the time performance, the Figure 3 shows the existing green software engineering process.

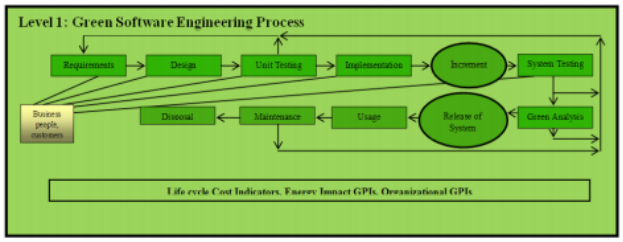

Figure 3. Green software engineering process.

5. Tools

There are some tools that to be used for finding the result using the Green and sustainable software engineering (Table 1).

Table 1. Different tools and performance

S.No	Tool Name	Performance
1	green power indicator	It is help full to analyze the web based application of the power quality
2	green tracker	Dealing the power utilization of the software with different versions.
3	Joule Meter	It helps to estimate the prorate power usage of the virtualized servers running the computer
4	Green Analytics	It support and show the impact effect of the WebPages on the environment
5	Soft Watt	It estimate the power used by the operating system and applications
6	Lab View	The tool gets and stores the current energy consumption for every 4 ms
7	Workload Simulator	This tool will produce the performance of a software application
8	Joulery	It visualizes the energy use of information across the network devices

6. Green based Model for the sustainable software engineering in SDLC

The above research study throws the light on the different problems in the life cycle phase of a software development in a traditional methodology. So the researcher tryes to introduce the GBMSSE method related to the Green Based Model for the Sustainable Software Engineering.

6.1 Requirement Phase

In the previous study, Requirement analysis was not given more importance related to the Green and sustainability. In future, it more importance is given to the requirement analysis phase in the beginning stage itself then the entire SDLC phase is slowly converted to green and sustainability the requirement for the observation time and the requirement of the analyzer have to raise questions related to green technologies to the customer, if the requirement analyzer gives the suggestion or guidelines to the customer about using a green technologies , and generic template which is needed to create based on the green and sustainable based requirement analysis if one who create that will be accepted by the globally because nowadays lot of research going on requirement engineering and requirement related sustainability, so there is need for common template and business community need accept that template[16]. The goal of our current research is to develop a "forward-looking" approach that incorporate well-timed metrics data with mock-up models of the software development method in order to support the software project managements control function.

6.2 Design Phase

In the early stages of design technique, the developer develops an application based on the requirement specification and a coding is developed based on that. In the design phase developers think that the coding technique only plays a main role related to energy usage and energy efficiency but they do not consider about the database design. The green and sustainable research community group need to create a technical Design documentation template with regard to the green and sustainability perspective that creates a template which is need to be approved by the business people[17]. It is very important to link the design and sustainability as early as possible. If

the software changes or new design technique related to customization of an application is not identified in the beginning then the new design product is much more expensive in time and cost point of view. Now a day's design concept has a tremendous growth. In the previous era only two application designs and product design were available but now web based application design, gaming design and app design were available fully related in the technical dimension of activities.

The paper[18] discusses the insinuation of the broadly used text processor as provide a solution in dealing the issues related to sustainability in the software designing and development. Green by software come into view some time ago. In general, green by software conceal the software developed for the field that works in the conservation of the atmosphere, as well as the software that helps to deal with energy-intensive applications. On the other hand, green in software is associated in making the software in a more sustainable way ensuring in a more sustainable product (this is called green software engineering). The next section will discuss this, of course the green in software also comprise other feature aside from the software development, such as supremacy. Sustainability requirements and thereby making the world eco-friendly. The cradle to grave trouble the industries of software that has been drawn in last 25 years can be indentified to show the requirement mistake the understanding methods differ from one person to another, this requirement document alone forwarded to the software development team if they develop an application as per the documentation and the entire application will give wrong results as it is not by the source code errors. The wrong requirement observation or wrong assumptions about the process, requirement specification analysis and for the protection, important systems began in the early 1990's to 2000, in the area of formal methods. Data base design also very important. As the designers of software technology and data base designer; they are accountable for the extended standing penalty of the designs of software and database. Data base design is one of the methods of accommodating the world and eloquent a choice commencement on how it must be created, based on the data base designer's objective. Through the data base design, one can create an optimized database that data base save the environment through energy consumption it's an indirect rebound effect.

6.3 Development Phase

If one needs to follow the general source code, it should be estimated that five software developers will carry out the software development phase for one year and that one engineer would provide support to the entire product's duration. One software engineer's personal device (a PC, development boards, debuggers, screens and so on) is rated at 600 watts. All the software engineers will split the IT infrastructure, consuming 3 kW (for example, consider the servers version control and build management systems). It is assumed that the software development devices run 24/7. Over 12 years, the power expenditure will be 29.5 kWy (4.5 kWy for development + 10 × 2.5 kWy for operation). Though, it was calculated that by using the IT equipment more correctly (switching it off, putting it in standby mode when not used, and so on), Then there will be decrease in consumption by 60 percent, to 11.8 kWy, the development of software is divided into three broad categories that application oriented, web based development and mobile based development. It is also related to technical dimension of aspects. The first confront for recent computers and a multifaceted network capacitors and for electronic storage to try to preserve a constant electrical energy and discriminate fluctuations in power consumption despite of what calculation is being performed, that is difficult to observe. In the current background, the major role of the in-house software project head is sensibly divided into three namely developing the individual, building the team, and achieving the task. When we are assessing the quality of both the development phase of the software as well as the quality of the end product, we are establishing a dialectical relationship between the two. The energy competences would interpret into improved design by software's taken as a whole runtime competence, which will interpret into lower energy costs. Defining a classes only when it is needed, closing the Database connection it saves the memory, in large arrays unset the variables to free the memory. Thus it will result in produce a Green based model for sustainable software engineering in today's situation.

6.4 Testing

In this division the tester has a device for testing the performance of the product, and the sense of the application. In 90 percentage of the company they either use an automated device or physically they test the applica-

tion and give a release. If they conduct a test with the help of a tool then its technical dimension of sustainability related approach and in this, testing team need to ensure the energy consumed by the tool and need to ensure the second and third order impacts ,suppose if they test the application manually then it is related to human energy sustainability related dimensions that is how many man powers are used and how many days the testing was conducted these aspect and methods need to be analyzed in this phase In testing Automation tools should be confident as they reduce the manual testing errors. They also highlight the re-use of test cases and regulate the testing process. This not only improves the accuracy of testing but also the efficiency and reduces the power consumed by additional resources in the manual testing process. A investigation[19] and a sample convention are to be held as squad facilitation and come close after a rounding about two or three of iterations. This enables the team members involved to exact and to find the design and performance which flows within the similar iteration without necessary to shift these to the next iteration. This commercialization has yield in rising the pressure to construct and to distribute massive volumes of high excellence manufactured goods and services bounded by the price and time limitation that are tighter than increasingly before in the software industry[20]. The order for advanced performances and innovative custom models[21] will also maintain to grow power competence and it will be vital for the computing production to be the augment battery days for mobile platforms and to decrease the power operating expense for PC and server platforms[22]. Role specific tools and processes are involved in the traditional approach the while the agile approach has collaborative platform with web-based tools, integration and practices. The plans in the traditional approach[23] are prone to false precision while the agile model has honest and evolving precision while it can resolve uncertainties. The traditional governance is based on the measurement of activities and artifacts while the agile approach measures the incremental outcomes and quality of the outcomes so the research is related to testing and is based on both the product and application oriented[24]. Assessment is a process of making a decision based on the rules and standards. Assessment is an on-going process, which combines the teaching and learning process. As like in the Figure 3, yellow box in the model that the customer and the business people should be a part of the requirement, design, implementation, and testing stages[25] but this proposed model having some short of difficulties to follow, if the development companies try to implement the green initiative at the end of all the process is over, if any small changes in the requirement, it increases the cost and time of the process.

6.5 Implementation

Completion of the developed application has a few difficulties already and now identified (i.e.) the customer needs and proper education in customer friendlness, and in ease of access one leads to consider all the factors(i.e.) OS of an computer, backup mediea , user manuals, and preservation of that application of first order, second and third impacts on sustainability because in this the study need to consider all the dimensions of technical, economical, human sustainability because all the sustainability factors are involved in this implementation and maintenance of a software development life cycle phase.

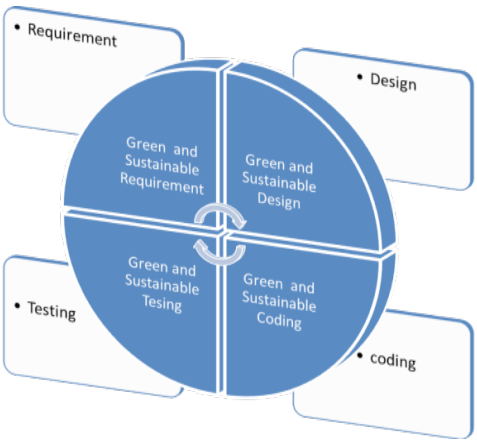

Figure 4. GBMSSE model in SDLC.

End users are the important people in influencing the implementation and the related actions to be achieved. To knowing the end user for greenability is critical in deploying the sustainability maintaience productively in a specified framework. Make use of Compact Disc and DVD(Digital Versatile Discs) or any kind of throw-away medium form a major source of e-Waste as they are configured and to be used on certain criteria such as exact number of installations, serial key based installation, machine or system based installation. An alternate approach could be the use of online license verification based installation to avoid the e-Waste created by the disposable media significantly. Separate Installation procedure to be avoided as it may consume a lot of disk space. Contemporary ways of reuseing the software installation

may also be working where a single copy of a software is deployed centrally and users can access it only when need arises It is universal understanding that a source-code printed by an automatic interpreter is less efficient than the sourcecode written by the skill developer. Shannon's entropy structure is referred that the entropy denotes the encoding style of different software engineers. This segment implements the SDD into some programming language that build up a software product. During this phase, various mechanism are generated, which are later incorporated into a final product. In the mode given below it shows only the Normal SDLC life cycle approach, It's not concerned with any Green and sustainable based analysis. This work introduces a new anticipated model for green and sustainable based in each and every phases of the SDLC phase and in the figure 4 shows the methodical description about the GBMSSE model analysis in SDLC.

In that Requirement, Design and coding, Testing and implementation takes many follow up and it is needed to achieve the green sustainability in software delvelopment life cycle phase. Figure 5 shows the iteration in SDLC implementation and the phases are described in the Figure 6.

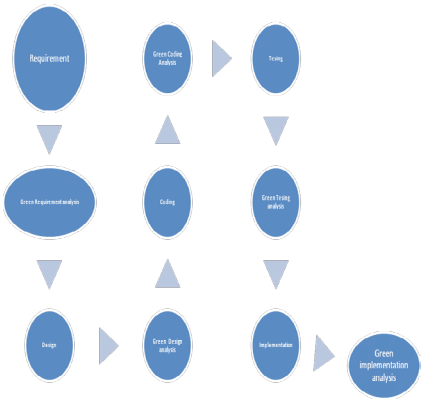

Figure 5. Iteration in SDLC.

7. Discussion

These credentials talk about the concerning the compensation of the sustainable software engineering using the green technologies in softare development life cycle and it is "drawbacks" with regard to the conventional SDLC. If one apparatus used by the software engineering in the sustainable way of using the energy efficiency and green IT in all footsteps of the software development life cycle, it will decrease the CO2 discharge in the progress time, also augment the competence of the creation without disturbing the surroundings and give the environmental friendly software. As a final point, in the conventional software development life cycle methods the ways that are going to be analyzed in GBSSE model illustrated in the work thought as a grave problem and for further research.

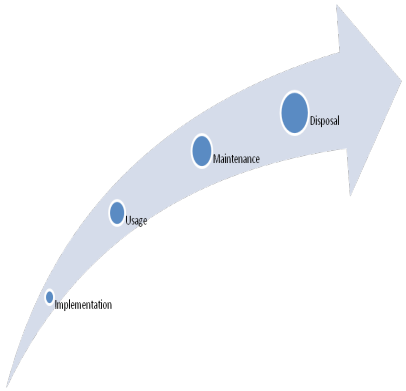

Figure 6. Implementation phase.

8. Conclusion

This work pointed out the initial phase of software engineering and it does not concentrate on sustainablilty in green techonologies software engineering which absolutely lead to the ecological threats. So it is to be realized that the initial step is forwarded to the software development companies insisting that they should attempt to apply the GBMSSE approaches using the green technology in every SDLC approaches. If this approach is implemented, it is sure that it resolve and decrease the atmosphere influence which leads to raise the software. quality.

9. References

1. Dick M, Drangmeister J, Kern E, Naumann S. Green software engineering with agile methods. 2nd International Workshop on Green and Sustainable Software (GREENS). 2013 May; 1(1). DOI: 10.1109/GREENS.2013.6606425.
2. Johann, T, Dick M, Kern E, Naumann S. Sustainable development, sustainable software, and sustainable software engineering: An integrated approach. 2011 International Symposium on Humanities, Science and Engineering Research (SHUSER). 2011 Jun 6–7. DOI: 10.1109/SHUSER.2011.6008495.
3. Bener AB, Morisio M, Miranskyy A. Green software. Software. 2014 May–Jun; 31(3):36–9. DOI: 10.1109/MS.2014.62.

4. Naumann S, Dick M, Kern E, Johann T. The GREENSOFT Model: A reference model for green and sustainable software and its engineering. Sustainable Computing: Informatics and Systems. 2011 Dec; 1(4):294–304. DOI: 10.1016/j.suscom.2011.06.004.
5. Sierszecki K, Mikkonen T, Steffens M, Fogdal T, Savolainen J. Green software: Greening what and how much? Software. 2014 May–Jun; 31(3):64–8. DOI: 10.1109/MS.2014.63.
6. Capra E, Francalanci C, Slaughter SA. Is software green? Application development environments and energy efficiency in open source applications. Information and Software Technology.2012 Jan; 54(1):60–71. DOI: 10.1016/j.infsof.2011.07.005.
7. Sahin C, Cayci F, Clause J, Kiamilev F, Pollock L, Winbladh K. Towards power reduction through improved software design. Energytech. 2012 May 29–31:1–6. DOI: 10.1109/EnergyTech.2012.6304705.
8. Taina-Good J. Bad and beautiful – In search of green software quality factors- CEPIS upgrade European Journal Exploring Initial Challenges for Green Software Engineering. 2011; 4.
9. Penzenstadler B, Femmer H, Richardson D. Who is the advocate? Stakeholders for sustainability. 2013 2nd International Workshop on Green and Sustainable Software (GREENS). 2013 May 20:70–7. DOI: 10.1109/GREENS.2013.6606424.
10. Penzenstadler B, Raturi A, Richardson D, Tomlinson B. Safety, security, now sustainability: The nonfunctional requirement for the 21st century. Software. 2018 May–Jun; 31(3):40–7. DOI: 10.1109/MS.2014.22.
11. Kroeger TA, Davidson NJ, Cook SC. Understanding the characteristics of quality for software engineering processes: A grounded theory investigation. Information and Software Technology. 2014 Feb; 5(2). DOI: 10.1016/j.infsof.2013.10.003.
12. Albertao F, Xiao J, Tian C, Lu Y, Zhang KQ, Liu C. Measuring the sustainability performance of software project. 2010 IEEE 7th International Conference on e-Business Engineering (ICEBE); 2010. p. 369–73. DOI: 10.1109/ICEBE.2010.26.
13. Kandananond K. A roadmap to green supply chain system through Enterprise Resource Planning (ERP) implementation. Procedia Engineering. 2014; 69: 377–82. DOI: org/10.1016/j.proeng.2014.03.002.
14. Dick M, Drangmeister J, Kern E, Naumann S. Green software engineering with agile methods. 2013 2nd International Workshop on Green and Sustainable Software (GREENS); 2013 May 20. p. 78–85. DOI: 10.1109/GREENS.2013.6606425.
15. Shenoy SS, Eeratta R. Green software development model: An approach towards sustainable software development. 2011 Annual IEEE India Conference, Hyderabad; 2011. p. 1–6. DOI: 10.1109/INDCON.2011.6139638.
16. Nagarajan R, Joseph AV, Sujatha S. Behavioural aspects of software project management- in- house software development. Indian Journal of Science and Technology. 2015 Feb; 8(S3). DOI: 10.17485/ijst/2015/v8iS3/58771.
17. Rashid E, Patnayak S, Bhattacherjee V. Estimation and Evaluation of change in software quality at a particular stage of software development. Indian Journal of Science and Technology. 2013 Oct; 6(10). DOI: 10.17485/ijst/2013/v6i10/38797.
18. Hilty LM, Lohmann W. The five most neglected issues in Green IT CEPIS upgrade. The European Journal for the Informatics Professional. 2011 Oct; XII(4):11–15.
19. Kogelman C-A. Examine green ICT awareness in organizations: Initial Findings CEPIS Upgrade. The European Journal for the Informatics Professional. 2011 Oct; XII(4):6–10.
20. Raffo DM, Harrison W, Vandeville J. Software process decision support: Making process tradeoffs using a hybrid metrics, modeling and utility frame work. Proceedings of the 14th ACM International Conference on Software Engineering and Knowledge Engineering; 2002. p. 803–9.
21. Penzenstadler B, Tomlinson B, Richardson D. RE4ES support environmental sustainability by requirements engineering. 2012 Journal of International Workshop on Requirements Engineering for Sustainable Systems; 2012.
22. Shahabuddin SM, Prasanth Y. Integration testing prior to unit testing: A paradigm shift in object oriented software testing of agile software engineering. Indian Journal of Science and Technology. 2016 May; 9(20). DOI: 10.17485/ijst/2016/v9i20/91223.
23. Bhatia MPS, Kumar A, Beniwal R. Ontologies for software engineering: Past, present and future. Indian Journal of Science and Technology. 2016 Mar; 9(9). DOI: 10.17485/ijst/2016/v9i9/71384.
24. Wankhede HS, Kiwelekar AW. Qualitative assessment of software engineering examination questions with bloom's taxonomy. Indian Journal of Science and Technology. 2016 Feb; 9(6). DOI: 10.17485/ijst/2016/v9i6/85012.
25. Mahmoud SS, Ahmad I. A green model for sustainable software engineering. International Journal of Software Engineering and its Applications. 2013 Jul; 7(4).