iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VII **Month of publication:** July 2022

DOI: https://doi.org/10.22214/ijraset.2022.45731

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

Green Based Software Engineering Approach for Sustainable Protocol

Neeta Verma[1], Dr. Varsha Jotwani[2]
[1]Research Scholar (CS&IT), Rabindranath Tagore University
[2] HoD (CS&IT), Rabindranath Tagore University

Abstract: ***A sturdy green software model is the prominent necessity for various governments fields and banking areas for Energy utilization processing. The model should give all the Energy consumption aspects namely power consumption, utilization of Memory, (Co2) Carbon dioxide release which is related to development of software and make it energy efficient. For Software Development Life Cycle phases this research paper proposes a new green software model for the virtual host machine, which is the base for all the Green based Software Development process.***
Keyword: Carbon dioxide, Green Software Model, Host Machine, Requirement, Testing

I. INTRODUCTION

In a software development, maintenance of sustainability and creating green kind of environment while developing of Software is generally considered as an important task. In such kind of circumstances, it is believed that such development procedure will be handled in an effective manner in form of storage and transmission to keep away from unnecessary resource usage by the hardware and software machine.

The most effective concept of green in this regard is called Green and Sustainable Software Engineering, which focus on the importance of creating and managing green and sustainable software development process.

The process of software development usually involves requirement analysis, design, coding, testing and implementation or maintenance process. These procedures operate into iteration of substitutions and variations of the original procedure in the form of software application [1].

Several strategies regarding Green based techniques can be arranged to prepare the software and hardware mechanisms. These techniques have been detailed in the literature survey. In reality, all the current models have few constraints such as: a) they are fixed only for fixed phases, b) they emphasize only fixed tasks, and c) they have issue which is related to performance.

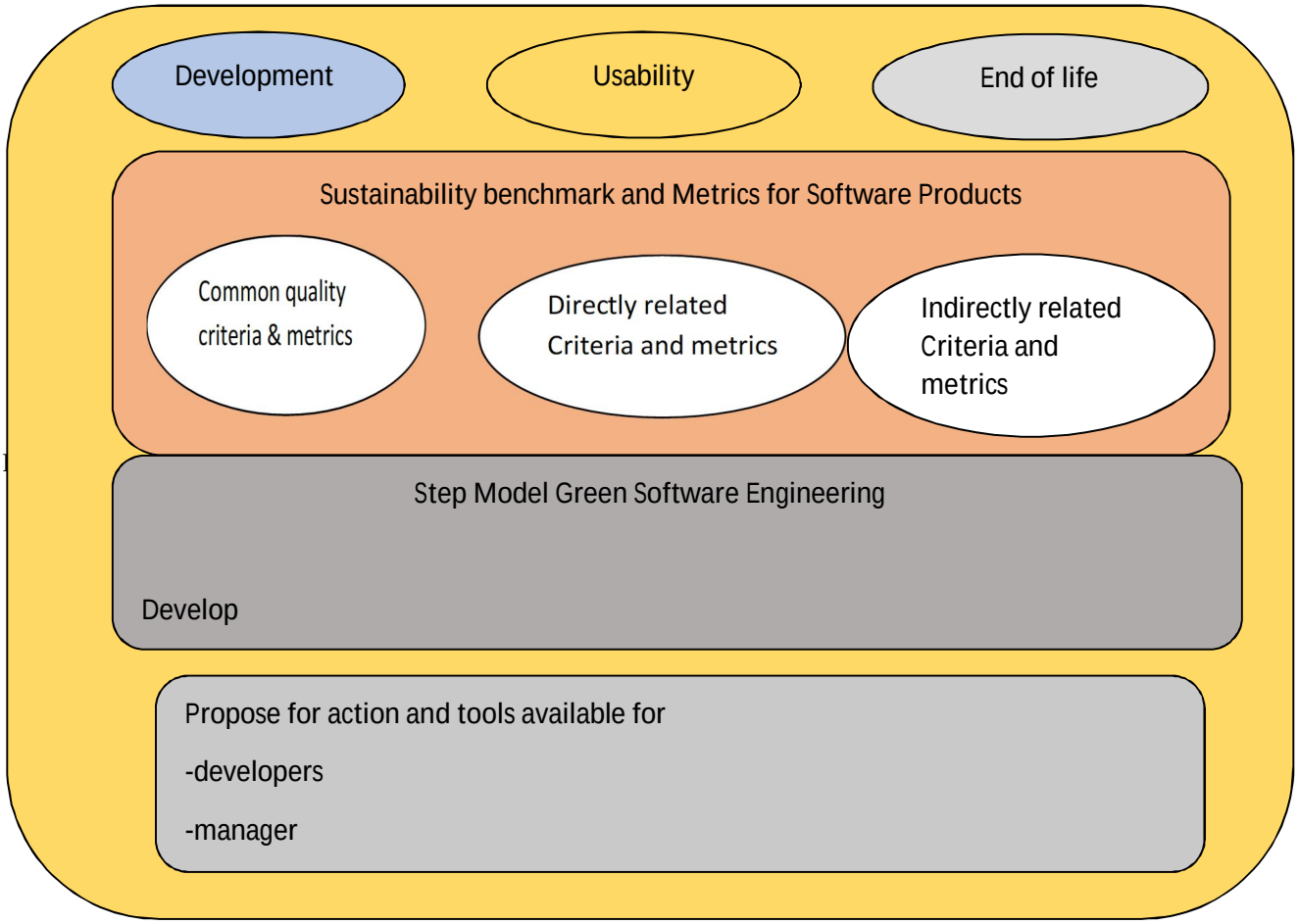
II. PROPOSED MODEL FOR GREEN SOFTWARE DEVELOPMENT FRAMEWORK

This research adopts the proposed Model framework Figure 2.1 shown proposed Model Framework below. This framework focuses on four key processes or activities in order to attain the research goals.

The four key processes are as follows:

1) Requirement,
2) Design and Implementation Stage
3) Testing
4) Green Review Stage

The current green based model is as shown in Figure 2.1.1 A new standard called Green and Sustainable Protocol put into effect. It is added as an effective phase of the existing Green based model. The green based analysis was done on each and every phase of the current model of Software Development Life Cycle process. The proposed model consists of both green and sustainability in each and every stage of Software Development Life Cycle process. Green based model for Sustainable Software Engineering is a new model proposed for Green Software Engineering. The two existing models viz. Testimonial model and Software Engineering Green process are both chooses to implement in the proposed architecture.



The architecture will be implemented in Software Engineering process. The analysis shows that the proposed model mainly falls on the iterative approach instead of the sequential approach that can lead to it to be Green. Figure 2.1 shows the Green Software Engineering Process Model3.

A. Requirement Stage

Requirement stage is the starting phase in which building of software product has been done. An energy efficient is comprise of the Requirement Engineering process was proposed as an energy-efficient process. It consists of five phases viz. feasibility, shape of service to be provided, categorize development and allotment of service to increments these two processes were recursive development based. There is a risk analysis which is one of the fourth phase and the last phase of the Requirement Test in terms of Power Competence. The Figure 3.5 shows the Green Requirement Engineering process.

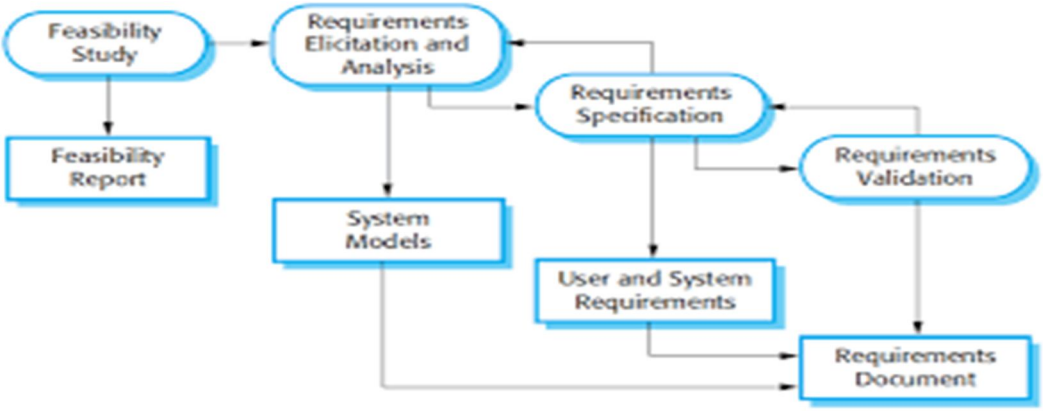


Figure: -2.1.1 Requirement Stage

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

B. Design and Implementation Stage

The overall system framework based on the requirement will be developed in the design phase. The fundamental software system abstractions and their relationships are different. This process of design, that is purely for component reuse in an energy-efficient manner, consist of the following points: search of component, selection of component, component customization and component validation.

Figure: -2.1.2 Design and Implementation Stage

C. Testing Stage

In this Green and Energy-efficient software testing phase, in the requirement tests should be developed in advance like in the requirement phase, to guarantee the tester and developer, that the requirements are correct and conform to the user's expectations. Testing Engineering process consist of four parts is of Requirement testing, Component testing, Integration testing, and Acceptance /System testing. Figure 2.1.3 shows the Testing Engineering Process

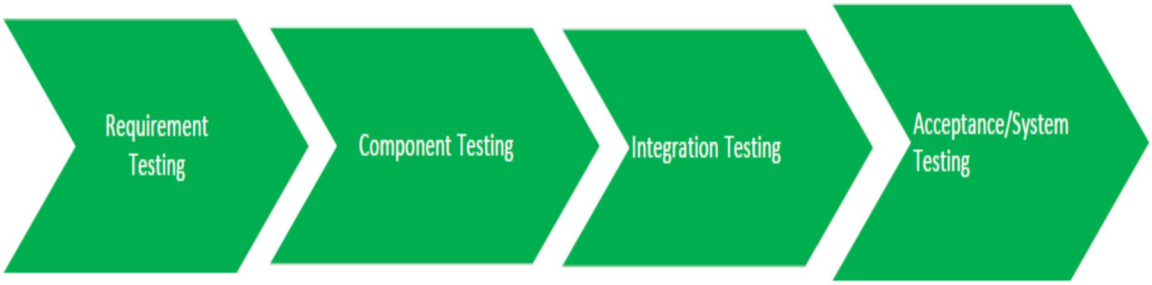


Figure:2.1.3 Testing Stage

D. Green Review Stage

Level1: The Green review stage starts with a fixed IT resource and quality metrics, collect data and take measurement, compare results against expected results and bring changes in requirementand development stages. Figure 2.1.3 shows Green Analysis process

Figure 2.1.3: Green Analysis Process

Level 2-This describes in which manner various software tools plays an important role in energy-efficient use of software applications. The tools and concepts are mentioned with examples and detailed information about the numerous energy related tools are also given at this level. Figure 2.1.4 depicts the software that encourage Green ICT.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

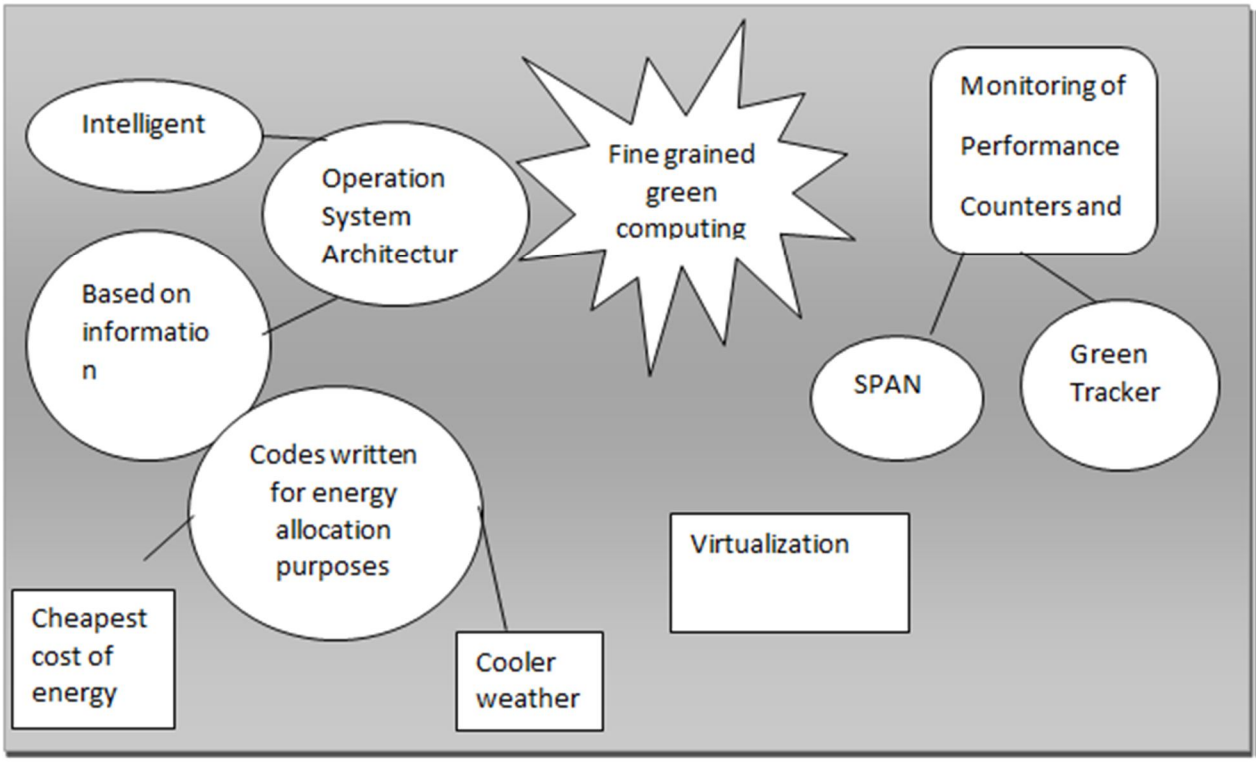


Figure:-3.4 Level2

III. PROBLEM AND CONTRIBUTION IN EXISTING MODELS:

Currently, there is some outline about to increase the viability of green concept in software engineering process but still there is some drawback still prevailing in software development life cycle, this work thus put a focus on the research questions:

1) What is the requirement for such procedures and ideas in each and every process indulge in the Software Development Life Cycle stages?

2) In the Software Development Life cycle phase, it is matter of fact to determine the process related to Complexity Maturity Model level?

Contribution:

GBFVSE Model (Green Based framework for viable Software Engineering) provides the breakthrough for these research questions. How does the Software Development Life Cycle stages linked to Complexity maturity model level and viability of a Software Engineering process? This kind of steps helps in developing a software which is helpful for generating the Software Development Life Cycle.

A. Green Based framework for the viable Software Engineering in SDLC

In traditional methodology the research study focuses on various problems in the life cycle phase. So, the researcher puts an effort to implement the GBFVSE (Green Based Framework for Viable Software Engineering).

1) Requirement Phase

In the initial phases, studies about Software requirement analysis did not give much significance to the Green and viability. But in the present scenario, the software industries give more significance for the requirement analysis phase. In the beginning stage itself one phase is given significance for Green and Sustainability and later the whole Software Development Life Cycle phase is slowly converted to Green and viability [2].

For raising question related to green and viability there is necessity of monitor time and analyzer it is relevant to the demand provides the consumer with suggestion or advice on how to use green technologies. In the todays world requirement observation is carried out by new technology like mail, teleconferencing, skype call, DVD exchanges between the customer and software development companies, etc. and a standard template that is required based on the Green and viable requirement review, globally it

was welcome, because lot of efforts were put into the gathering of requirement engineering and requirement linked viability and non-working requirement, so there is a necessity for basic protocol and trading community requires to accept that protocol.
In the formal procedures the wrong observation of the requirement and assumptions about the process, specification of requirement analysis and for the protection important systems start in the early 1990's to 2000, in the area of formal methods. Therefore, the objective of the present research is to create a "forward-looking" approach in which software project managements control the function to involve well defined metrics data by developing artificial protocol.

2) Design Phase

The developer produced an application based on the software requirements specification at the first design idea phase, and subsequent coding was made based on that. During the design process, developers believe that the coding technique plays the most essential function, which is linked to power production and efficiency, but database architecture is not a major worry. Making technical design documentation The Green and Viable Research Community Group needs a template with regard to the Green and Sustainability point of view, which produces a protocol that must be endorsed by business people. (Ekbal Rashid 2018). Only two designs were available in the previous scenario, which were tied to application and product design, but today web-based application design, gaming configuration, and app design are all fully related in the technical dimension of activities.
In his paper, Lorenz M. Hilty (2011) [10] addresses the insinuation of a widely used text processor as a solution to difficulties linked to viability in software design and development. It is an endeavour to build software that is environmentally friendly in order to make the world a better place. The software industry has evolved over the last 25 years to depict one person to another, and this requirement document is forwarded to the software development team alone.
If they try to build an application according to the documentation, the entire application will fail because the problems are not in the source code. The database design was also crucial. They are responsible for the extended standing penalty of the software and database designs as software technology and database designers. It is an indirect rebound effect that one can construct an optimised database through database design that can save the environment through energy usage.

3) Development Phase

If one is crucial to collect the general source code, it is predicted that five software developers will spend an additional year in the software development phase, and product engineers will be required for the complete development. The personal device of one software engineer (a PC, development brands, debuggers, screens, etc.) is rated at 600 watts. The software engineer will split it up in the IT infrastructure, which will absorb 3 kW. (for example, consider the servers version control and build management systems). The software development machines are said to run 24 hours a day. The total power consumption over the next 12 years will be 28.6 kWy (2.6 kWy for development + 10 *2.6 kWy for operation).
Though, it was found that by using IT equipment in a more responsible manner (turning it off when not in use, placing it in standby mode when not in use, and so on), absorption would increase by 60%, to 11.8 percent. KWy, software development is divided into three categories: application-oriented development, web-based development, and mobile-based development. It also has to do with technical considerations.
The first piece for current computers and a multifarious network capacitor and for electronic storage is tough to monitor calculation to save electrical energy and differentiate fluctuation in power usage. In the current context, the main role of the in-house software project head can be divided into three categories: developing the individual, forming the team, and completing the task.
1. Single Quotes should be applied (Double quotes are applicable for Strings),
2. Apply = = = instead of = =
3. Looping side:-
Avoid

```
for( $m=0; i< count($arr); $m++){
echo count($arr);
}
while ($m<= $limit-2)
do {loop code}
```

Alternative Option

```
$len = count(arr);
for( $m=0; m< $len; $m++){
echo $len;
}
$t = $limit-2;
while ($i <= $t)
do {loop code}
```

4. When it is required define Classes
5. Implement the techniques of caching (Reduce the Database burden & operations),
6. Close the Database connection (It helps to reduce the loss of memory).
7. Lessen the number of hits to Database. (Ex: Bulk insert, update automatically instead of each row insert or update)
8. For better output result aggregate style and script files,
9. To large array unset the variables to free memory.
10. Instead of If statement it is better to recommended to implement select statements (switch () {case ”: }), than multi if and else if statements.

For programming concept, the above points are the better practice. For today’s scenario the outcome will produce a green based protocol for viable software engineering in today’s circumstances.

4) Testing

There is a tester that acts as a gadget and the sense of the application for testing the product's performance. The automated gadget is used in the majority of the company, with 90 percent of the company either using it or physically checking and releasing the application. If they use a tool to test, they must examine the tool's sustainability in a technical manner in order to maintain the linked approach. To assure the second and third order impacts, the testing team must determine the energy absorbed by the tool. If they physically verify the application, it is tied to human energy. The manner in which manpower is implemented and for how many days this will be put into functioning condition is a sustainability related component. Automation tools should be used in testing since they reduce manual testing errors. They also focus on test case reuse and testing process regulation. This not only reforms the testing process, but it also makes it more efficient and reduces the power absorbed by unnecessary resources in manual testing.

After two or three iterations of rounding, it is discovered in an investigation (Carol-Ann 2011) that sample conventions are to be held as squad facilitation and get near after two or three iterations.

This permits team members to precisely and collect the design and performance that flows inside the same iteration without having to migrate from one recurrent iteration to the next. This type of employment has increased the need to generate and maintain high quality. They distribute a large volume of goods and services, as well as price value and time constraints (D.M. Raffo 2002)[3]. The demand for high performance and creative custom models (Birgit Penzenstadler 2012)[4] will also be able to nurture the development of power competence, and it will be critical for calculating production to improve battery days for mobile platforms while lowering power working expenses for PC and server platforms. (Shaik Mohammad Shahabuddin, 2016)[5].

Traditional approaches use role-specific tools and work, whereas agile approaches use a collaborative platform with web-based tools, integration, and practises. The plans of the traditional approach (M. P. S. Bhatia 2016)[6] are prone to misleading precision, but the agile model has honest and changing accuracy while also being able to deal with uncertainties. Traditional governance measures incremental breakthrough and outcome quality, whereas agile governance evaluates incremental breakthrough and outcome quality. As a result, research is closely related to testing and is entirely focused on both the product and the application. Due to complexity of network structure, it is not convenient to implement centralized cloud system because it delays to handle the congestion of network [7]. (Hansaraj S 2016) The process of making a choice based on norms and criteria is known as assessment. Assessment is a continuous process that incorporates both teaching and learning. As shown in Figure 3, the customer and business people should be involved in the requirement, design, implementation, and testing stages (Sara S 2013). However, this proposed model may be difficult to follow if development companies try to implement the green initiative at the end of the process, if there are any minor changes in the requirements.

5) Implementation

Overall, the developed application had a few issues that were previously unidentified; that is, the customer requires proper knowledge in developing friendliness among customers, and in order to access easily, one must work on all factors, including the computer's operating system, backup media, user manuals, and the preservation of that application of first, second, and third order. The end users are the ones that have the most say in the execution and activities that must be undertaken. To effectively deliver sustainability maintenance within a specified framework, it is critical to understand who the end user is for Green-ability. Because they are configured and to be used based on specific criteria such as the exact number of installations, serial key-based installation, machine or system-based installation, Compact Discs (CD) and Digital Versatile Discs (DVD), as well as any other single/one-use medium, are a major source of e-Waste.

Another approach is to prevent the electronic waste caused by disposable media by using online licencing verification, which does not require installation. Current ways of reusing software installation, in which a single copy of the software is installed centrally and users can access the services only, when necessary, may also be effective. An automatic interpreter that generates source code is less efficient than the way source code is generated by knowledge.

Shannon's entropy structure, which defines the entropy characterises various software engineers' encoding approach, is used by a variety of software engineers. This module converts the SDD into a software product by translating it into a few computer languages. At this period, several techniques will be created, which will later be included into a final product. In the mode below, it only shows the Normal System Development life cycle technique; it does not focus on any green or viable analysis.

Based on each step of the software development life cycle, a new projected model for green and viable software development is generated in each phase. This work proposes a new anticipated model for Green and Sustainable based on each phase of the SDLC, as seen in Figure 3.1, and the phases listed in Figure 3.2.

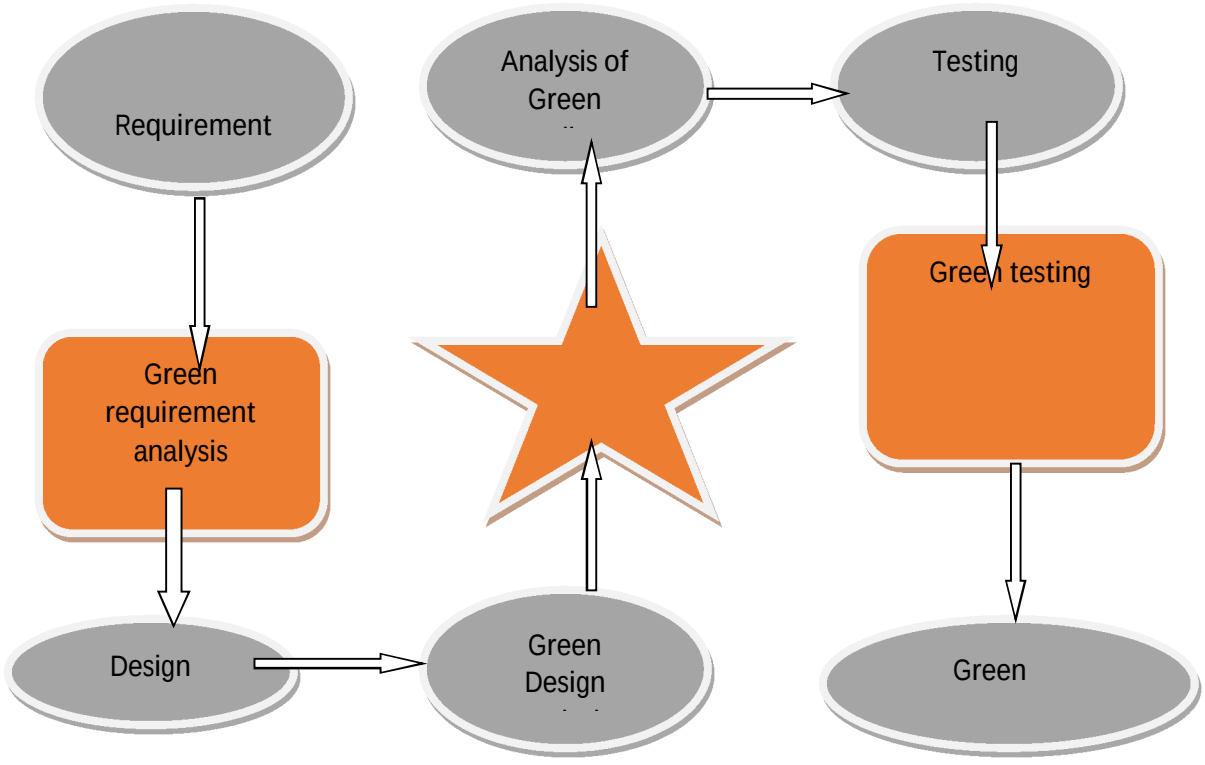


Figure 3.2 SDLC Phases

Comparison Among Different Engineering Process

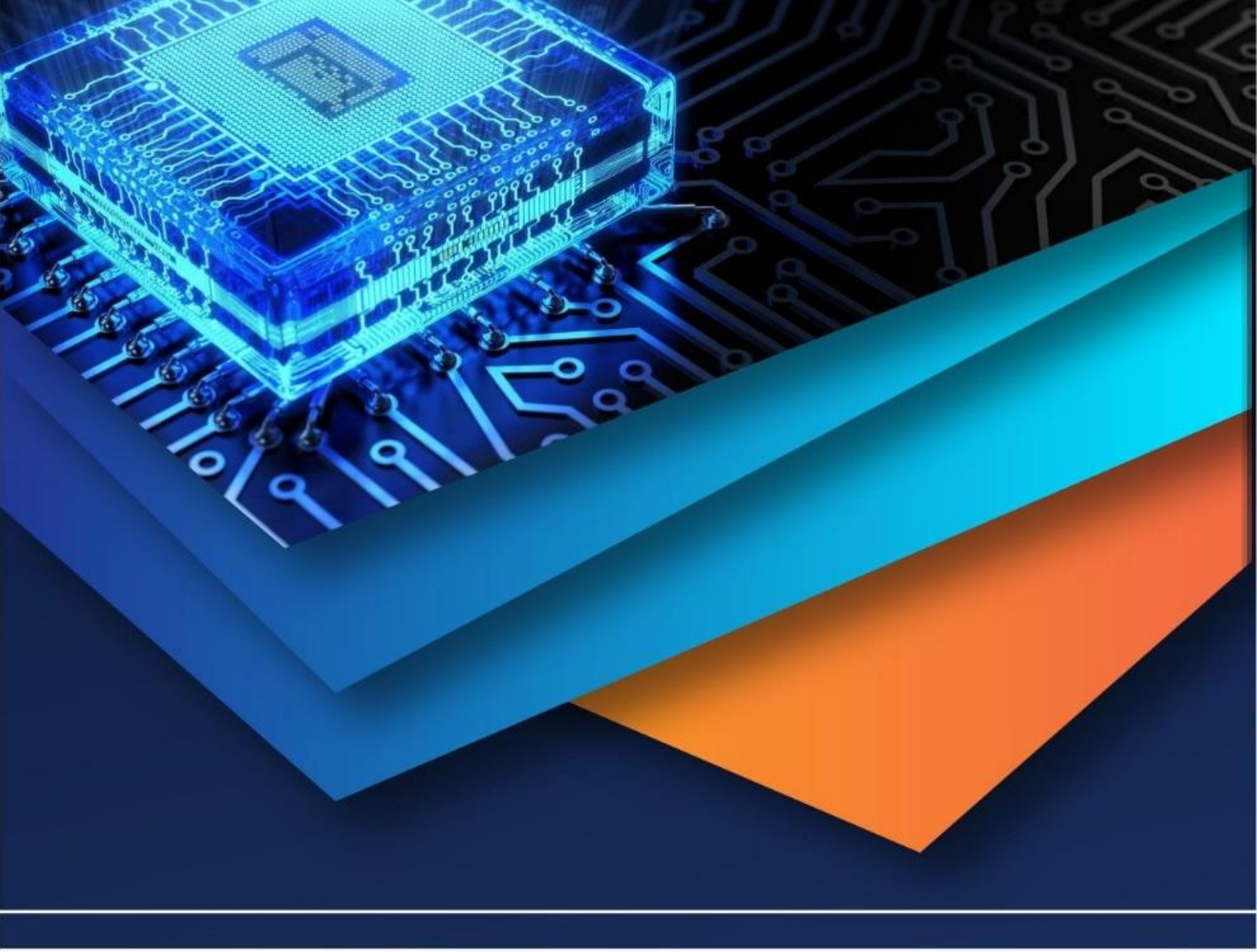
S.No	Electronically Green Software Engineering Process	Green and viable Software engineering	Green based model for sustainable software engineering (GMSSE)
1	It has been incorporated during the planning phase.	It has been incorporated during the planning phase.	It has been adopted during the life cycle of development of software.
2	A customised green review phase must be built to organise the amount of iteration processes and actions.	Analysis and preview are needed post second and third repetitions.	With numerous advanced hardware and software devices, a green analysis phase plan is set up at every level of the software development life cycle.
3	There are no preliminary measures in the initial step, and the viability of the supplier is under control.	It was unable to find an equivalent.	In the software development life cycle, no equivalent could be found.
4	In requirement engineering, elements that affect viability should be determined.	It is necessary to recognise the schematic life cycle procedure, as well as its exemplary issues and influence.	Identifies the factors that influence the viability of a design step and a quality attribute.
5	The requirements are linked in EGSEP, and the green analysis is set up.	GSSE has a analysis and preview procedure in place for estimating development.	The green approach is used in all aspects of the GBMSSE analysis at all stages of the software development life cycle.
6	It is feasible to put the changes in the requirements to the test.	It tries to keep track of requests for changes during development review and preview.	Give priority to the viability of the engineering process for SDLC during the SDLC phase.
7	It made an effort to create and recognise many components during the reference process.	Attempts to depict how the viability of aims will be achieved	Attempt to cover all aspects of the software development life cycle.
8	This will review the modification made in the requirement	It puts efforts to manage the changes requests during the duration of development review and preview	May control modification request in SDLC stages
9	It helps to monitor on the sustainability engineering process related to implementation.		

IV. CONCLUSION

The proposed green architecture for the software development life cycle model was provided Work has been presented in four stages. The requirement, design, coding, testing, and implementation four states has been presented for Green based analysis, by applying the Sustainable Software Engineering technique which adopt green based model. Four alternative forms of analysis, as well as the current model's strengths and weaknesses, were given. The results suggest that the proposed model was well-suited for analysing the Green based Software Development Life Cycle process for software development organisations and developers.

REFERENCES

[1] StefanNaumann, "The GREENSOFT Model: A reference model for green and sustainable software and its engineering," Sustainable Computing: Informatics and Systems, vol. 1, no. 4, p. 294, 2011.
[2] J. Sarkis, "A strategic decision framework for green supply chain management," Journal of Cleaner Production, vol. 11, no. 4, pp. 397-409, 2003
[3] D. M. Raffo, "Software process decision support: making process tradeoffs using a hybrid metrics, modeling and utility framework," Proceedings of the 14th international conference on Software engineering and knowledge engineering, pp. 803-809, 2002.
[4] L. M. Hilty, "ICT for Sustainability: An Emerging Research Field," ICT Innovations for Sustainability, pp. 3-36, 2014.
[5] S. M. Shahabuddin, "Integration Testing Prior to Unit Testing: A Paradigm," Indian Journal of Science and Technology, vol. 9, 2016.
[6] M. Bhatia, "Ontology based framework for detecting ambiguities in software requirements specification," 3rd International Conference on Computing for Sustainable Global Development (INDIACom), 2016.
[7] D. Jotwani, "Low latency Placement for Effective Fog based Infrastructure," International Journal of Science and Research , vol. 9, no. 6, pp. 901-905, 2020.
[8] M.Kumar, "International Journal of Education and Management Engineering," vol. 6, pp. 59-68, 2018.
[9] D.Costa, "A systematic review of life cycle sustainability assessment: Current state, methodological challenges, and implementation issues," Science of The Total Environment, vol. 686, pp. 774-787, 2019.
[10] W. L. -. a. LM Hilty, "The five most neglected issues in "Green IT" CEPIS UPGRADE,," 2011 .



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY