

Published Online: 15th July, 2024

Awareness and Utilisation of Green Software Engineering Practices among Computer Education Students in Enugu State

Dr. Uzochukwu Peter Obichukwu

Department of Computer and Robotics Education, University of Nigeria, Nsukka

As society becomes increasingly conscious of environmental sustainability, there is a growing imperative to align software development practices with green principles. This study investigated the awareness and utilisation of green software engineering among computer education students. The study participants were 109 students selected using a total population sampling technique from the Department of Computer and Robotics Education, University of Nigeria, Nsukka, Enugu State. They comprised 78 males and 31 females. Their ages ranged from 19 to 24 years ($M = 21.13$; $SD = 2.31$). Data was collected using the Awareness of Green Software Engineering Practice Questionnaire and the Utilisation of Green Software Engineering Practices Questionnaire. The data was analysed using means and standard deviations in SPSS version 25. The findings of the study revealed that computer and robotics education students are highly aware of green software engineering practices; however, green software engineering practices are hardly utilised by students. As the demand for energy-efficient technology continues to grow, the importance of green software engineering will only increase, making it an essential component of modern computer science education. Hence, the study's findings have implications for integrating green software engineering into the curriculum, providing faculty training, offering practical assignments, and leveraging industry partnerships to equip students with the skills necessary to develop sustainable software.

Keywords: Awareness; Computer Education; Green Coding Practices; Green Software Engineering; Sustainable Software Development; Utilisation

Introduction

The rapid development of technology has made the global adoption of environmentally friendly software engineering practices necessary. Global warming is one of humanity's biggest threats. Due to the severity of the situation, the climate must be considered in all aspects of our professional and personal lives. As technology advances rapidly, the environmental impact of the information technology (IT) industry is becoming increasingly evident. From the production and disposal of electronic devices to the energy consumption of data centres, the IT sector contributes significantly to carbon emissions and resource depletion. Mubarak and Augie (2020) assert that information and communication technology (ICT) usage in educational

institutions is prevalent today. Evidence suggests that this usage increases carbon dioxide emissions. These ICT usages may result in high energy consumption and harmful waste. Green coding practices aim to mitigate these adverse effects by promoting efficient use of resources and reducing waste in software development processes.

Mubarak and Augie (2020) suggest that education institutions should adopt green computing programmes to reduce energy consumption, carbon emissions, and ICT waste due to the negative impact of ICT. Energy accounts for three-quarters of greenhouse gas emissions, and over 80% of our energy comes from fossil fuels (Ritchie et al., 2022). In their study, Mahmoud and Ahmad (2013) defined

green software engineering as applying principles, practices, and techniques to develop energy-efficient and environmentally friendly software systems. Meanwhile, Mohan and Murugesan (2014) see green software engineering as the process of designing, developing, deploying, and maintaining software systems in an environmentally sustainable manner by reducing their negative impacts on the environment and enhancing energy efficiency.

Developing software systems with minimal environmental impact is the primary focus of the field of study known as green software engineering, which is also sometimes referred to as sustainable software engineering (Jnr et al., 2018). They further revealed that green technology, also known as code green, is a developing discipline encompassing guidelines, ideas, and know-how for creating and managing sustainable software programs. Code Green is an initiative called green software engineering, which is a field where software practices and architecture, electricity markets, hardware, data centre design, and climate science intersect to try and make software engineering more sustainable. Code Green aims to create code modifications, architectural adjustments, and lifestyle decisions that minimise the amount of carbon dioxide the program emits (Herelius, 2022). It covers every stage of a software product's lifecycle, including the design, usage, and effects on the economy, society, and environment. It mixes hardware, data centre design, energy markets, software methodologies and architecture, and climate science. Green software engineering is a field that focuses on reducing the adverse environmental effects of software systems, such as energy use, carbon emissions, and electronic waste.

Herelius (2022) further opined that green software engineering involves implementing strategies to decrease energy consumption, optimise resource utilisation, and mitigate the environmental impact of software systems. Green software engineering is a new field that combines software development principles with climate science, electricity markets, data centre design, architecture, and computer hardware. Reducing computer energy use is crucial with global warming and rising energy costs (Moshnyaga, 2013). Computer systems consist of hardware like memory, CPU, input, output circuits, etc. and software that instructs the hardware to perform specific operations. When powering the motherboard circuit, hardware uses energy. This approach aims to raise awareness among students and educators about the environmental impacts of software systems and equip them with the knowledge and skills to develop energy-efficient, sustainable software that minimises its carbon footprint. Green software engineering is essential in the field of education, especially computer education.

Departments of Computer and Robotics Education in various educational institutions are charged with training professionals in computer hardware maintenance, information technology, software development and application, database management, and programming. Computer education refers to the general study of computers, their parts, and their application to human endeavours. Ibezim and Obi (2010) defined computer and robotics education as a course of study in higher institutions that aims to train people in skill acquisition in using computers to solve problems. According to Nwangwu (2010), computer and robotics education could be seen as the study of computers

and their applications, which involves knowing how computers work and how they are used to solve different problems. Computer education is enriched with hands-on courses such as word processing, hardware maintenance and management, software development and applications (Ibezim, 2018). Computer education is vital in schools as it prepares students for a digital future, provides various career opportunities, and equips them with critical thinking and problem-solving skills. Computer education aims to assist students in acquiring the abilities necessary for efficient and successful computer use. Learning how to utilise technology and software, access and use information, and produce and distribute material are all included in this.

Awareness of green coding practices, such as green software engineering among computer education students, is a critical first step towards widespread adoption. Awareness includes knowledge of specific techniques, such as code optimisation and efficient data structures, as well as an understanding of the broader environmental implications of software development. However, research indicates that awareness levels of green coding practices are generally low (Capra et al., 2012; Dick et al., 2013). This low level of awareness may be attributed to the academic curriculum contents (Murugesan, 2013), the prioritisation of performance and functionality over energy efficiency by industries (Capra et al., 2013), and the lack of textbooks and resources focused on green coding practices (Penzenstadler & Femmer, 2013). According to Hanzelik et al. (2022), the awareness and utilisation of green software engineering in computer education is crucial for promoting sustainable development in software engineering.

On the other hand, utilisation of green software engineering practices refers to computer education students' practical application of green software engineering principles in their coding projects and assignments. It involves actively implementing energy-efficient algorithms, optimising code, and considering the environmental impact of software during the development process. Research indicates that the utilisation of green coding practices among students is also minimal. A study by Dick et al. (2013) showed that while students might be aware of basic optimisation techniques, they rarely apply these methods in their projects unless explicitly instructed to do so. The study highlighted that most students focus on meeting functional requirements and deadlines with little consideration for energy efficiency. By writing clean, efficient code that minimises energy consumption and resource usage, students can enhance the overall performance of their software and reduce its carbon footprint. This can lead to faster and more responsive applications that deliver a better user experience while reducing the software's energy costs.

In addition to the environmental and economic benefits of green coding practices, there are social and ethical reasons for computer and robotic education students to adopt sustainable development practices in their coding work. By incorporating green coding techniques into their projects, students can contribute to a more equitable and just society by promoting environmental sustainability and social responsibility in the technology industry. This can help create a more inclusive and sustainable future for all members of society, including marginalised communities disproportionately impacted by environmental degradation and climate change.

Furthermore, green coding practices can help computer and robotic education students develop valuable skills in high demand in the technology industry. As more companies and organisations prioritise sustainability and environmental responsibility, students proficient in green coding techniques will have a competitive advantage in the job market. By demonstrating their commitment to sustainability and showcasing their ability to develop eco-friendly software solutions, students can position themselves as valuable assets to employers looking to reduce their environmental impact and align with green technology trends.

However, despite the growing interest in green coding practices among computer and robotic education students, there are still challenges and barriers towards the awareness and utilisation of green coding practices (Raisian et al., 2016). One of the main obstacles is the lack of understanding and education about sustainable development and green coding in many computer science and software engineering programs. A study by Capra et al. (2012) revealed that many students lack a basic understanding of how their coding practices can influence energy consumption. While they may be familiar with the concept of green IT, they often do not associate it directly with their everyday coding activities. More so, Dick et al. (2013) found that while students understood the importance of reducing energy consumption, they were unaware of specific coding practices that could achieve this goal. Without proper guidance and training, students may struggle to understand the importance of green coding practices and how to effectively integrate them into their projects.

Another challenge is the complexity and technicality of green coding practices, which can be daunting for students just starting their coding journey. Murugesan

(2013) found that green coding is often perceived as a complex and advanced topic, which deters students from implementing these practices. Green coding requires a deep understanding of energy efficiency, resource optimisation, and sustainable development principles, which may require additional training and expertise beyond what is typically covered in university computer science courses. Students may find implementing green coding practices in their projects challenging without access to the necessary resources and support. They may revert to traditional coding methods out of convenience.

In the contemporary digital age, the field of software engineering plays a pivotal role in technological advancements. As society becomes increasingly conscious of environmental sustainability, there is a growing imperative to align software development practices with green principles. Studies have shown the importance of green coding, which includes creating and integrating environmental consciousness in software development, development of energy-efficient computer software, development of sustainable software, and development of software that can minimise carbon footprint, among others (Atadoga et al., 2024). Studies have also revealed that integrating green software in education, especially in computer education, can help to identify areas of improvement such as carbon reduction during usage of computers in teaching and learning, reduction in energy consumption during software development and implementation and integration of green disposal system in the course of teaching and learning of computer education (Junger et al., 2024).

The impact of green software engineering cannot be over-emphasised, but one could wonder to what extent computer education students know and utilise green software

engineering. Therefore, this work assesses the awareness and utilisation of green software engineering practices among computer education students at the University of Nigeria, Nsukka. To achieve this aim, the study addressed the following research questions:

1. What is the current level of awareness of green software engineering practices among computer and robotics education students?
2. To what extent do computer and robotics education students utilise green software engineering practices in their software development projects?

Methods

Participants

The population of this study consisted of 109 computer and robotic education students (78 males and 31 females) from the University of Nigeria Nsukka. Their ages ranged from 19 to 24 years ($M = 21.13$; $SD = 2.31$). At the time of data collection, about 124 students were admitted to the Department of Computer and Robotics Education, University of Nigeria, Nsukka, across different levels of study. Due to the population's manageable size, participants were selected using a total population sampling technique, whereby the whole population of students enrolled in the Department at the time of data collection were invited to participate in the survey. However, only those who gave their consent proceeded to complete the survey. For the participants' level of education, 34 were in the first year, 29 were in the second year, 22 were in the third year, and 24 were in the final year.

Instrument

A questionnaire comprising three sections was used for data collection: the participants' demographic characteristics section (age, gender and level of education), the Awareness of Green

Software Engineering Practices Questionnaire, and the Utilisation of Green Software Engineering Practices Questionnaire, developed by the researcher. The Awareness of Green Software Engineering Practices Questionnaire assesses participants' knowledge of specific techniques for green software engineering. It consists of ten items responded to on a 4-point rating scale: 1 = Not Aware; 2 = Fairly Aware; 3 = Moderately Aware; and 4 = Highly Aware. An example of an item of the scale is "I am knowledgeable about energy-efficient coding techniques used in green software engineering". Total scores on the scale range from 10 to 40, with higher scores indicating higher awareness of green software engineering practices. The researcher obtained an acceptable Cronbach's alpha of .85 for the instrument, which shows that the items were internally consistent. The Utilisation of Green Software Engineering Practices Questionnaire assesses participants' practical application of green software engineering practices in their coding projects and assessments. It consists of seven items responded to on a 5-point rating scale: 1 = Not Used At All; 2 = Somewhat Used; 3 = Used; 4 = Highly Used; and 5 = Very Highly Used. An example of an item of the scale is "I consciously implement sustainable software design principles to reduce resource consumption in my projects". Total scores on the scale range from 7 to 35, with higher scores indicating higher utilisation of green software engineering practices. The researcher obtained an acceptable Cronbach's alpha of .87 for the instrument, which shows that the items were internally consistent. Three lecturers from the Department of Computer and Robotics Education, University of Nigeria Nsukka, further validated the research instruments. They were required to assess

the instrument grammatically, its adequacy and relevance, sentence structure and suitability of each item, with particular reference to the purpose of the study and research questions.

Procedure

With the assistance of four research assistants, who were class representatives for each level of education, the researcher distributed copies of the questionnaires to the Department of Computer and Robotics Education students. Before the distribution, the researcher instructed the research assistants on administering the instrument, who should get it, and how to retrieve it. The briefing aimed to inform the intended respondents about the research and emphasise the value of confidentiality. A total of 125 questionnaires were printed and distributed. However, 117 copies were

returned to the researcher. After cleaning and sorting the returned copies, eight were excluded from the data analysis, as they were either incomplete or completed wrongly.

Design and Data Analysis

The study adopted a descriptive survey research design. The data collected from this study was analysed using mean and standard deviation in the Statistical Package for Social Sciences (SPSS) version 25. Any item with a mean of 2.50 or above was considered high, while items below 2.50 were considered low.

Result

Research Questions 1

What is the current level of awareness of green software engineering practices among computer and robotics education students?

Table 1: Respondents' mean ratings and standard deviation on the awareness of green software engineering practices

S/N	ITEMS	\bar{X}	SD	DECISION
1	I am aware of the concept of green software engineering practices.	3.59	0.62	Highly Aware
2	I understand the environmental impact of traditional software development practices.	3.51	0.83	Highly Aware
3	I am knowledgeable about energy-efficient coding techniques used in green software engineering.	3.53	0.80	Highly Aware
4	I am aware of sustainable software design principles that reduce resource consumption.	3.59	0.79	Highly Aware
5	I have received education or training related to green software engineering practices	3.57	0.74	Highly Aware
6	I actively seek information and resources to stay updated on green software engineering practices	3.56	0.84	Highly Aware
7	I believe that integrating green software engineering practices is important for environmental sustainability.	3.55	0.81	Highly Aware
8	I think that incorporating green practices in software development can positively impact society.	3.50	0.74	Highly Aware
9	I consider environmental responsibility to be an ethical aspect of software development.	3.54	0.79	Highly Aware
10	I believe that green software engineering practices can enhance the quality of software products.	3.58	0.67	Highly Aware
CLUSTER MEAN		3.56		Highly Aware

\bar{X} = Mean; SD = Standard Deviation

The results in Table 1 showed the mean ratings on the awareness of green software engineering practices among computer and robotic education students. On a total score of 4.0 for each of the items, the responses by the participants had mean scores of 3.50 and above, with a total mean score of 3.56 by the participants, which shows a high level of awareness of the green

software engineering practices by the computer and robotics education students.

Research questions 2

To what extent do computer and robotics education students utilise green software engineering practices in their software development projects?

Table 2: Respondents' mean ratings and standard deviation on the utilisation of green software engineering practices

S/N	ITEMS	\bar{X}	SD	DECISION
1	I actively incorporate energy-efficient coding techniques in my software development projects.	1.00	0.67	Not Used
2	I consciously implement sustainable software design principles to reduce resource consumption in my projects.	1.27	0.61	Not Used
3	I make an effort to optimise software code for energy efficiency when developing applications.	1.16	0.60	Not Used
4	I use tools and technologies that promote eco-friendly software development in my projects.	1.26	0.99	Not Used
5	I seek guidance and support from faculty or mentors to incorporate green software engineering practices in my software development work.	1.24	0.65	Not Used
6	I actively seek opportunities to apply green software engineering practices in real-world software development projects.	1.21	1.13	Not Used
7	I believe that using green software engineering practices can result in more efficient and cost-effective software solutions.	1.20	0.58	Not Used
CLUSTER MEAN		1.98		Not Used

\bar{X} = Mean; SD = Standard Deviation

The results in Table 2 showed the mean ratings on computer and robotic education students' usage of green software engineering practices. On a total score of 5.0 for each of the items, the responses by the participants had mean scores of 1.26 and below, with a total mean score of 1.98 by the participants, which shows a low level of utilisation of the green software engineering practices by the computer and robotics education students.

Discussion

The present study hoped to examine the level of awareness and the utilisation of green software engineering practices among computer and robotic education students. The study findings revealed that the students were highly aware of green software engineering practices. This finding aligns with the assertions of Gupta (2010) that shortly after the Energy Star programme was established in 1992, the term "green computing" began to gain popularity. The Energy Star programme is a voluntary programme that recognises products that conform to its high energy efficiency standards. Meanwhile, in Nigeria, the National Information Technology Development Agency has started addressing the importation of climate-friendly computer systems. Other agencies like the Computer Registration Council of Nigeria Nigeria Computer Society ensure that computers and other peripherals imported into the country are approved and environmentally friendly (Nwankwo, 2007). This increased awareness among students could be a result of certain factors, such as the growing awareness of climate change and the need for sustainable development (Agboola & Emmanuel, 2016; Esakkimuthu & Banupriya, 2023) and the growing demand for environmentally responsible technologies in the market

(Leelavathi & Satyanarayana, 2024; Mustofa et al., 2023). Mubarak and Augie (2020) reported that there had been works on green computing in Nigeria, yet its take-up has not been impactful.

The study also revealed that computer and robotics education students hardly utilised green software engineering practices. This is in line with the findings of Suleiman (2021), who states that the level of acceptance and practices of green computing is significantly low. Meanwhile, Mubarak and Augie (2020) revealed that Nigeria has failed miserably in promoting environmentally friendly computing due to a lack of education and focus. They further showed that some African business people are considering making money from the situation, while many American and European companies are trying to reduce carbon dioxide emissions. Okewu (2017) claimed that the computer industry in Nigeria is just as responsible for accelerating climate change as the aviation sector. The fact that some users will keep their system running even when it is not being used results from ignorance regarding green computing and the energy it consumes. In their study on the application of green computing at Nigerian tertiary institutions, Ogbene et al. (2022) found that proper ventilation is not provided in the classrooms where computer systems are being used, making the room temperature very high, which is very hazardous to human health.

Implication of the Study Findings

The findings of this study have several practical implications for the students, the educators, the university administrators, and the policymakers. The use of green software engineering practices by computer and robotic education students is a significant and necessary step towards building a more sustainable future for the technology

industry. By integrating sustainability into their coding work, students can reduce the environmental impact of software development, create more efficient and resource-friendly applications, and contribute to a more equitable and just society. While there are challenges and barriers to overcome, the benefits of green software engineering practices far outweigh the costs, and students and educators alike must prioritise environmental responsibility in their software engineering projects and coursework. By embracing green software engineering techniques, computer and robotic education students positively impact the world and pave the way for a more sustainable future. Students must understand the power they hold in shaping the technology industry and the environment as a whole. By incorporating green software engineering practices into their projects, students gain valuable skills and knowledge and contribute to a more significant movement towards environmental consciousness in the tech sector. As the next generation of developers, students can lead by example and inspire others to follow suit. Together, they can drive innovation, promote sustainability, and create a better world for future generations.

On the part of educators and university administrators, integrating green software engineering into the core computer science curriculum is essential for raising awareness and encouraging utilisation. The educators and administrators can achieve this by incorporating relevant topics into existing courses, such as algorithms and systems design. Including dedicated modules or elective courses explicitly focused on sustainable software development can further enhance students' understanding and skills. Assignments and projects requiring students to apply green software engineering

principles can reinforce theoretical knowledge. These practical tasks should be designed to challenge students to optimise code for energy efficiency and consider the environmental impact of their software. Providing feedback on these aspects can help students refine their skills and appreciate the importance of green software engineering. Additionally, Equipping educators with the knowledge and resources to teach green software engineering is crucial. Professional development programs and workshops for faculty can help them stay updated on the latest practices and tools in sustainable software engineering. This, in turn, will enable them to convey these concepts to their students effectively.

Furthermore, various educational interventions have shown promise in improving the awareness and utilisation of green software engineering practices. Penzenstadler and Femmer (2013) reported on the success of workshops and specialised courses designed to teach sustainable software engineering. These programs provided students with hands-on experience in applying green coding techniques, resulting in improved knowledge and practical skills. Developing and disseminating educational resources focused on green software engineering is vital. Textbooks, online courses, and tutorials covering sustainable software development can be valuable reference materials for students and educators. Creating optimised code examples and case studies repositories can provide practical guidance and inspiration.

More so, collaborations between educational institutions and industry partners can also enhance the integration of green software engineering into the curriculum. For example, projects sponsored by companies prioritising sustainability can provide real-

world contexts for students to apply green software engineering practices. These collaborations can also help align educational goals with industry standards, making green software engineering more relevant and appealing to students. Instances where green coding has been successfully implemented in educational settings offer valuable insights. A case study by Dick et al. (2013) highlighted a university program that integrated green software engineering into its software engineering courses. The program emphasised practical applications, requiring students to optimise existing codebases and measure the impact of their changes on energy consumption. This hands-on approach increased student engagement and demonstrated the tangible benefits of green software engineering.

Conclusion

This study aimed to assess the level of awareness and utilisation of green software engineering practices among computer education students. The findings of this study showed that the students were highly aware of green software engineering practices, although they hardly utilised these practices. The awareness and utilisation of green software engineering practices among computer education students are critical for fostering a generation of environmentally conscious software developers. While current application levels are limited, strategic educational interventions can bridge this gap. By integrating green software engineering into the curriculum, providing faculty training, offering practical assignments, and leveraging industry partnerships, academic institutions can equip students with the knowledge and skills necessary to develop sustainable software. As the demand for energy-efficient technology continues to grow, the importance of green software engineering

will only increase, making it an essential component of modern computer science education.

References

- Agboola, O. S., & Emmanuel, M. (2016). Awareness of climate change and sustainable development among undergraduates from two selected universities in Oyo State, Nigeria. *World Journal of Education*, 6(3), 70-81.
- Ahmad, T. B., & Nordin, M. S. (2014). University Students' Subjective Knowledge of Green Computing and Pro-Environmental Behavior. *International Education Studies*, 7(2), 64-74.
- Atadoga, A., Umoga, U. J., Lottu, O. A., & Sodiya, E. O. (2024). Advancing green computing: Practices, strategies, and impact in modern software development for environmental sustainability. *World Journal of Advanced Engineering Technology and Sciences*, 11(1), 220-230.
<https://doi.org/10.30574/wjaets.2024.11.1.0052>
- Capra, E., Francalanci, C., & Slaughter, S. A. (2012). Is software "green"? Application development environments and energy efficiency in open source and proprietary software. *Information and Software Technology*, 54(1), 60-71.
- Capra, E., Francalanci, C., & Slaughter, S. A. (2013). Is green IT your IT? Consistent and competitive suppliers for IT services. *Business &*

- Information Systems Engineering*, 5(5), 353-363.
- Dick, M., Naumann, S., & Kuhn, N. (2013). A model and selected instance of green and sustainable software. *Proceedings of the 2013 International Conference on Information and Communication Technologies for Sustainability (ICT4S)*, 246-254.
- Dookhitram, K., Narsoo, J., Sunhaloo, M., & Sukhoo, A., (2012). *Green computing: an awareness survey among University of Technology, Mauritius students*. International Conference on Higher Education and Economic Development, Mauritius.
- Esakkimuthu, K., & Banupriya, S. (2023). Awareness about Climate Change among Students: A Sustainable Future. *ComFin Research*, 11(4), 1-6.
<http://dx.doi.org/10.34293/commerce.v11i4.6677>
- Ganesan, M., Kor, A. L., Pattinson, C., & Rondeau, E. (2020). Green cloud software engineering for big data processing. *Sustainability*, 12(21), 9255.
- Gupta, S. (2010). Computing with green responsibility. *Proceedings of the International Conference and Workshop on Emerging Trends in Technology*.
<http://dl.acm.org/citation.cfm?id=1741959>
- Hanzelik, P. P., Kummer, A., and Abonyi, J. (2022). Edge-computing and machine-learning-based framework for software sensor development. *Sensors*, 22(11), 4268.
<https://doi.org/10.3390/s22114268>
- Herelius, S. (2022). Green coding: Can we make our carbon footprint smaller through coding? *Blekinge Institute of Technology, Faculty of Computing*, 2022, 1-39.
- Jnr, B. A., Majid, M. A., & Romli, A. (2018). An empirical study on predictors of green sustainable software practices in Malaysian electronic industries. *Journal of Information and Communication Technology*, 17(2), 347-391.
- Junger, D., Westing, M., Freitag, C. P., Guldner, A., Mittelbach, K., Obergöcker, K., ... & Wohlgemuth, V. (2024). *Potentials of Green Coding--Findings and Recommendations for Industry, Education and Science--Extended Paper*. arXiv preprint arXiv:2402.18227.
- Leelavathi, S., & Satyanarayana, S. (2024). Influence of green labels on green purchase intention: A study on consumers in Bangalore. *Educational Administration: Theory and Practice*, 30(5), 12908-12916.
- Mahmoud, S. S., & Ahmad, I. (2013). A green model for sustainable software engineering. *International Journal of Software Engineering and Its Applications*, 7(4), 55-74.
- Mubarak, A., & Augie, A. I. (2020). Awareness level of green computing among computer users in Kebbi State, Nigeria. *International Journal of Environmental and Ecological Engineering*, 14(10), 111-123

- Murugesan, S. (2013). Harnessing green IT: Principles and practices. *IT Professional*, 10(1), 24-33.
- Mustofa, M., Suseno, B., & Basrowi, B. (2023). Technological innovation and the environmentally friendly building material supply chain: Implications for a sustainable environment. *Uncertain Supply Chain Management*, 11(4), 1405-1416.
- Okewu, E. (2017). Optimising Green Computing Awareness for Environmental Sustainability and Economic Security as a Stochastic Optimisation Problem. *International Journal of Computer Science and Mobile. Application*, 4(7), 1-6.
- Omobowale, O. A. (2013). Tokunbo ICT. *International Journal of Sociology & Social Policy*, 33(78), 509-523. www.emeraldinsight.com.
- Patterson, M. K., Poese, M. E., & Moore, J. (2013). Deploying energy-efficient data centers. *Communications of the ACM*, 60(2), 72-80.
- Pazienza, A., Baselli, G., Vinci, D. C., and Trussoni, M. V. (2024). A holistic approach to environmentally sustainable computing. *Innovations in Systems and Software Engineering*, 1-25.
- Penzenstadler, B., & Femmer, H. (2013). A generic model for sustainability with process- and product-specific instances. *Proceedings of the 2013 International Conference on Software Engineering*, 943-946.
- Perkins, D., Brune Drisse, M., Nxele, T., & Sly, P. (2014). E-waste: a global hazard. *Annals of Global Health*, 80(4), 286-295. www.annalsofglobalhealth.org.
- Raisia, K., Jamaiah, Y., & Aziz, D. (2016). Current challenges and conceptual model of green and sustainable software engineering. *Journal of Theoretical and Applied Information Technology*, 94(2), 428-443.
- Ritchie, H., Roser, M., & Rosado, P. (2022). 'Energy mix,' Our World in Data. *Global Change Data Lab, Oxford Martin School, University of Oxford*. <https://ourworldindata.org/energy-mix>
- Savvas, A. (2012). Cut Costs with a Green Network. *Computer Weekly*, 24(12), 12-13.
- Sendall, P., Shannon, L., Peslak, A., & Saulnier, B. (2011). The greening of the information systems curriculum. *Information Systems Education Journal*, 9(5), 33-41. www.isedj.org.
- Servaes, J. (2012). Introduction to 'Green ICT'. *Telematics and Informatics*, 29(4), 335-
- Suleiman, A. A. (2021). Investigation on level of awareness, acceptance and practices of green computing in universities in north western region of Nigeria. *British International Journal of Education and Social Sciences*, 8(10), 4519 – 6511
- Suryawanshi, K., & Narkhedeb, S. (2015). Green ICT for Sustainable Development: A higher education

perspective. *Procedia Computer Science*, 70, 701 – 707.

Taruna, S., Singh, P., & Joshi, S. (2014). Green computing in developed and developing countries. *International Journal in Foundations of Computer Science & Technology*, 4(3), 97-102.
<https://doi:10.5121/ijfcst.2014.4309>