

Sustentabilidade no Desenvolvimento de Software: uma Revisão Bibliométrica e Sistemática da Literatura

ROMULO RIEDER

CENTRO UNIVERSITARIO FEI

GABRIELA SCUR

CENTRO UNIVERSITARIO FEI

GETULIO K. AKABANE

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE SÃO PAULO (PUCSP)

RENATA MARIA NOGUEIRA DE OLIVEIRA

UNIVERSIDADE PRESBITERIANA MACKENZIE (MACKENZIE)

Sustentabilidade no Desenvolvimento de Software: uma Revisão Bibliométrica e Sistemática da Literatura

Abstract

O desenvolvimento de software desempenha um papel crucial na sociedade moderna, impulsionando a inovação, a eficiência e a produtividade em diversos setores. No entanto, surgem preocupações pertinentes sobre o impacto ambiental e social dessa indústria em rápida expansão. Nesse contexto a sustentabilidade no desenvolvimento de software surge como uma abordagem fundamental para minimizar os impactos negativos e maximizar os benefícios ambientais e sociais do software. Este artigo tem como objetivo realizar uma revisão bibliométrica e sistemática da literatura sobre sustentabilidade no desenvolvimento de software, onde através dessa análise, busca-se identificar e sintetizar modelos e abordagens propostas por pesquisas sobre esse tema, que recomendam a incorporação de aspectos de sustentabilidade na engenharia de software que podem ser aplicados durante o desenvolvimento e uso de sistemas de tecnologia, como minimizar os efeitos negativos no meio ambiente e produzir produtos de software sustentáveis. Espera-se que essa revisão possa contribuir como um instrumento inicial de leitura para outras pesquisas assim como um guia de apresentação de modelos que podem apoiar no avanço e adoção de práticas de desenvolvimento de software sustentável.

Palavras-chave: sustentabilidade em software, software sustentável, software verde, engenharia de software.

1 Introdução

O desenvolvimento de software desempenha um papel crucial na sociedade moderna, impulsionando a inovação, a eficiência e a produtividade em diversos setores. No entanto, à medida que a dependência de tecnologia digital aumenta, também surgem preocupações crescentes sobre o impacto ambiental e social dessa indústria em rápida expansão. A sustentabilidade no desenvolvimento de software emerge como uma abordagem fundamental para enfrentar essas preocupações, buscando minimizar os impactos negativos e maximizar os benefícios ambientais e sociais do software (MOURÃO; KARITA; DO CARMO MACHADO, 2018).

Uma revisão bibliométrica e sistemática da literatura sobre sustentabilidade no desenvolvimento de software é essencial para compreender o estado atual do conhecimento nessa área, identificar modelos e práticas propostas. Essa análise abrangente permite uma visão

crítica das contribuições acadêmicas, fornecendo uma base sólida para apoiar futuras pesquisas e práticas no campo da sustentabilidade em software.

Este artigo tem como objetivo realizar uma revisão bibliométrica e sistemática da literatura sobre sustentabilidade no desenvolvimento de software. Através de uma análise sistemática e quantitativa de um conjunto selecionado de artigos científicos e técnicos, busca-se identificar e sintetizar contribuições relevantes, propostas e abordagens adotadas na pesquisa sobre esse tema.

A revisão bibliométrica permitirá a análise quantitativa de aspectos como as principais fontes de publicação, os autores, instituições e países com maior produção científica bem suas distribuições cronológicas. Já a revisão sistemática fornecerá uma análise qualitativa mais detalhada, explorando as principais abordagens, metodologias, desafios e soluções propostas nos estudos selecionados.

A partir dessa revisão bibliométrica e sistemática da literatura, espera-se fornecer uma visão agregadora do estado atual do conhecimento sobre sustentabilidade no desenvolvimento de software, que possa possibilitar a identificação de oportunidades de pesquisa, bem como a definição de diretrizes e melhores práticas para apoiar o avanço e a adoção de práticas de desenvolvimento de software sustentável.

2 Motivação e Fundamentação

O software foi se tornando crucial e passou a fazer parte da nossa vida hoje, sendo vital para realizar trabalhos e tarefas com mais eficiência. Mas o software causa atividade de hardware e dessa forma é responsável pelo incremento do consumo de energia. Por isso o software sustentável e a sustentabilidade na engenharia de software são agora reconhecidas como preocupações importantes e atuais não apenas para pesquisadores, mas também para toda a indústria de software (BAMBAZEK; GROHER; SEYFF, 2023).

No entanto, é vasta a literatura abordando essa temática fazendo com que os pesquisadores motivados a contribuir com esse tópico tenham que investir muito tempo na busca de conhecimento através da pesquisa bibliográfica. Este artigo relata uma revisão da literatura com o objetivo de elaborar uma base pertinente de conhecimento que possa apoiar partes interessadas sobre sustentabilidade na engenharia de software.

2.1 Aspectos de Sustentabilidade na Engenharia de Software

A definição mais citada de desenvolvimento sustentável é atender às necessidades do presente sem comprometer a capacidade das futuras gerações de satisfazer suas próprias

necessidades. Segundo essa definição, o desenvolvimento sustentável deve satisfazer três dimensões: sociedade, economia e meio ambiente (BRUNDTLAND, 1987).

Os aspectos de sustentabilidade na engenharia de software podem ser aplicados tanto durante o desenvolvimento quanto durante o uso de sistemas de software. Assim, PENZENSTADLER *et al.* (2012) distinguem quatro aspectos de sustentabilidade na engenharia de software. O ponto de vista do processo de desenvolvimento inclui:

- Aspecto do processo de desenvolvimento: sustentabilidade no processo inicial de desenvolvimento do sistema (com uso responsável de recursos ecológicos, humanos e financeiros).
- Aspecto do processo de manutenção: sustentabilidade do sistema de software durante o período de manutenção até a substituição por um novo sistema (com monitoramento contínuo da qualidade e gerenciamento do conhecimento).

Já o ponto de vista do produto abrange os aspectos de sustentabilidade durante a operação e uso:

- Aspecto da operação do sistema: sustentabilidade do sistema de software como produto em relação ao uso de recursos para operação (utilizando princípios de TI verde e componentes de hardware produzidos de forma sustentável).
- Aspecto do uso do sistema: sustentabilidade nos processos de uso no domínio de aplicação desencadeados pelo sistema de software como produto (responsável pelo impacto no meio ambiente).

Esses aspectos implicam diferentes níveis de abstração e granularidade dependendo do sistema em análise.

2.2 Software Verde

A engenharia de software verde tem crescido e evoluído devido às preocupações com o desenvolvimento de software, a indústria e os consumidores. No entanto, ela se concentra nos aspectos fundamentais de eficiência energética e consumo de energia, que afetam o meio ambiente. Muitos estudos discutem o desenvolvimento de software verde, com foco na perspectiva de hardware em eficiência energética e consumo de energia. Os objetivos da engenharia de software verde são minimizar os efeitos negativos no meio ambiente e produzir produtos de software sustentáveis (SIERSZECKI *et al.*, 2014).

Como muitos estudos enfatizam o hardware verde, percebe-se, em comparação, uma escassez de estudos sobre software verde, embora os impactos sejam diretos no hardware, mas

o software também tem impactos indiretos. As iniciativas de software verde visam economizar recursos, como energia, eletricidade e recursos naturais, para preservar a estabilidade do ecossistema. Atualmente, o software não é mais visto apenas em termos de produtividade, mas precisa se concentrar em escalabilidade, usabilidade e qualidade, com os avanços recentes na demanda por novas tecnologias (IBRAHIM *et al.*, 2021).

3 Método

O método utilizado foi Revisão Sistemática da Literatura (SLR), onde o processo de busca para este estudo foi baseado em uma busca automatizada na base da biblioteca digital Elsevier/Scopus (<https://www.scopus.com/>), com o objetivo de capturar todos os resultados relacionados à sustentabilidade com engenharia de software.

A *string* de busca utilizada para pesquisa foi:

“sustainable software” OR “green software” OR “sustainability in software”

AND

“software engineering” OR “software development”

Com a utilização dessa *string* procurou-se combinar os aspectos desejados de forma a abranger todas as dimensões da sustentabilidade no que tange o processo de concepção de software, representando pelos termos engenharia e desenvolvimento que são comumente utilizados em Tecnologia da Informação. O objetivo também com esses termos foi evitar que pudesse retornar resultados envolvendo exclusivamente aspectos de sustentabilidade voltadas para hardware.

Como ação de refinamento, os seguintes critérios foram aplicados de forma a restringir o resultado:

- Área de conhecimento: Ciência da Computação, Engenharia, Ciências Sociais, Negócios e Gestão. O objetivo desse filtro foi evitar trabalhos principalmente das áreas de Matemática, Ciência Ambiental, Energia, dentre outras.
- Tipo de documento: Artigo;
- Tipo de fonte: Journal;
- Idioma: Inglês.

Após aplicação dos critérios acima para refinamento do resultado, os documentos encontrados foram exportados e considerados para a Revisão Bibliométrica utilizando como

apoio a ferramenta RStudio – software livre de ambiente de desenvolvimento integrado para R, uma linguagem de programação para gráficos e cálculos estatísticos – e a ferramenta Biblioshiny – interface visual para o Bibliometrix, software livre para pesquisa quantitativa incluindo os principais métodos bibliométricos de análise.

Por fim foram selecionados 5 artigos para uma Revisão Sistemática, utilizando como critério estabelecido pelo Autor, que leu todos os títulos e abstracts, e selecionou artigos que tratam sobre modelos ou frameworks, a fim de identificar estruturas de referência que possam ser aplicadas em contextos variados da indústria de software. Optou-se por uma seleção dessa forma com o objetivo de encontrar diretrizes, padrões ou métodos já formulados que possam orientar a implementação de práticas sustentáveis em software ou mesmo a sua validação. Ao focar em modelos ou frameworks, buscou-se analisar uma base sólida e estruturada para a implementação de iniciativas de sustentabilidade em software, utilizando abordagens recomendadas por pesquisadores do tema.

4 Resultados

Como resultado do processo de busca foram obtidos 61 artigos. Analisando-os quantitativamente, no que diz respeito a produção científica anual, é possível observar que o pico de concentração de publicações esteve no ano de 2018 e atualmente ainda há um volume de produção relevante, com índices em 2021 e 2022 superiores aos índices anteriores a 2018, conforme apresentado na Figura 1.

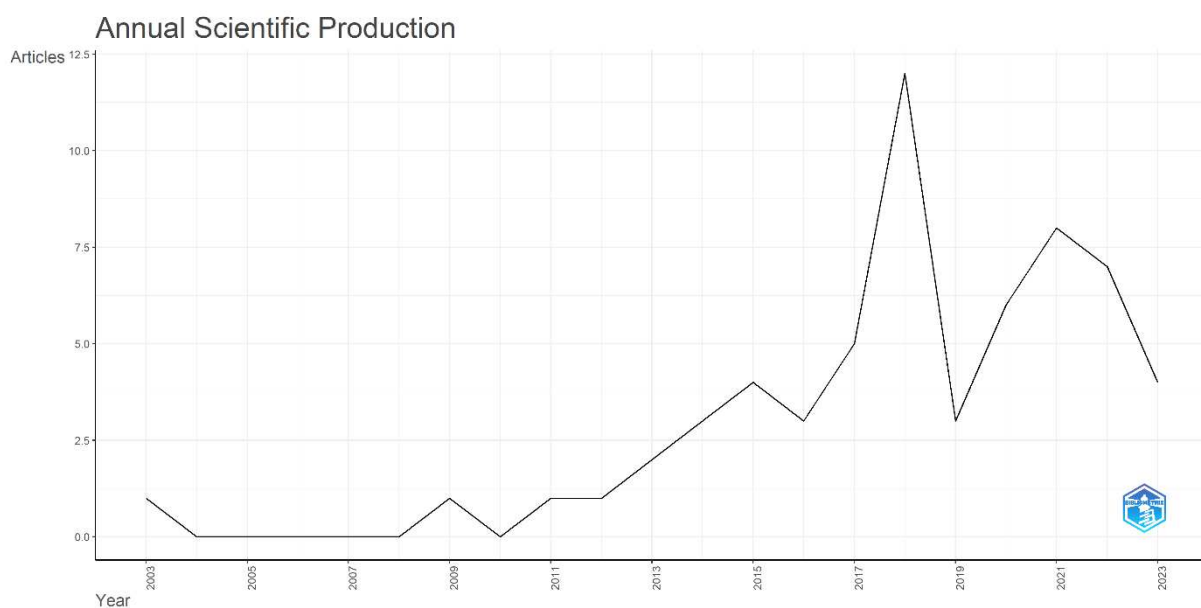


Figura 1 – Produção científica anual

Já no que diz respeito às Fontes de publicação, é possível observar na Figura 2 que os Journals mais representativos são: SUSTAINABILITY (SWITZERLAND) e IEEE SOFTWARE.

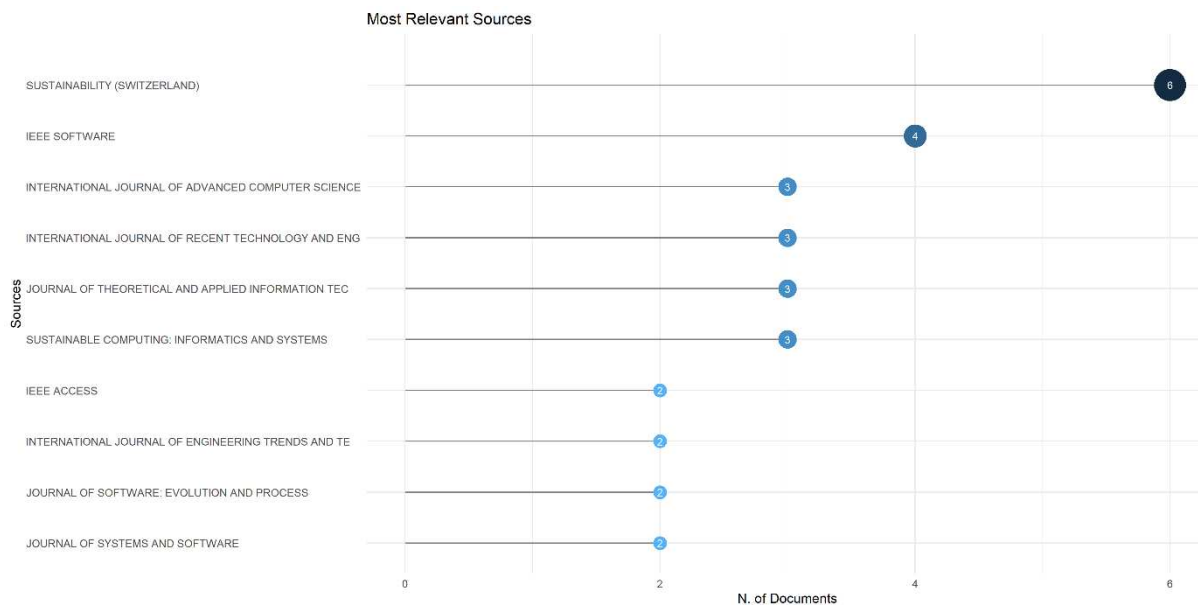


Figura 2 – Fontes mais relevantes

No que diz respeito aos Autores, temos como mais relevantes: YAHAYA J., com 5 artigos publicados; e em seguida com 4 artigos publicados temos os autores: DERAMAN A., KHAN SU e MOHANKUMAR M, conforme apresentado na Figura 3.

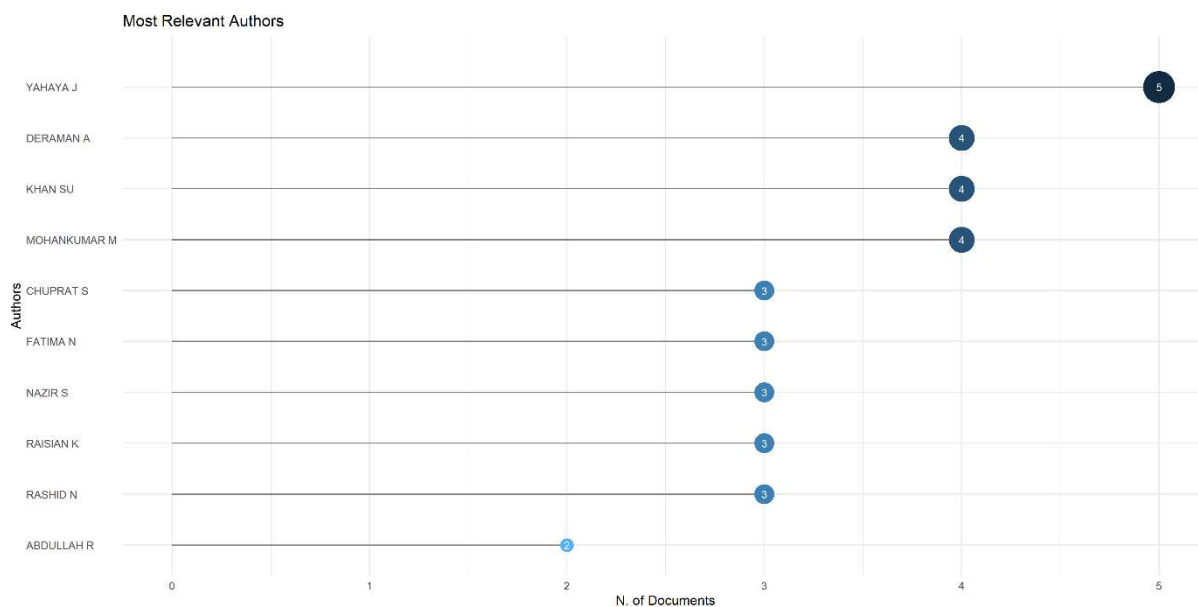


Figura 3 – Autores mais relevantes

Observando os documentos mais citados globalmente temos o artigo de Naumann *et al.* (2011) como o de maior referência apresentando uma quantidade bastante expressiva de citações (184), como mostra a Figura 4. Esse artigo mais adiante será discutido com mais detalhes na revisão sistemática.

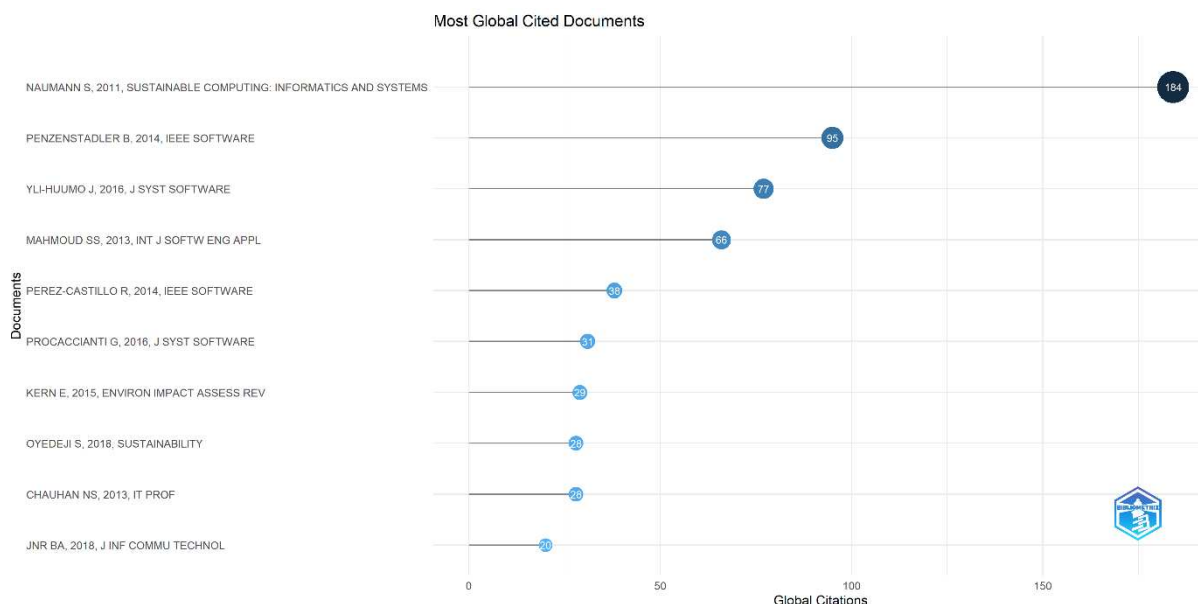


Figura 4 – Documentos mais globalmente citados

Analisando quantitativamente as palavras mais frequentes observa-se através da Figura 5 em destaque os termos: *sustainable development*, *sustainable softwares*, *software engineering*, *computer software* e *energy efficiency*.



Figura 5 – Nuvem de palavras

5 Discussões

Para a Revisão Sistemática foram selecionados os 5 artigos listados na Tabela 1, conforme critério previamente descrito na seção Método deste artigo.

Autores	Título	Ano	Fonte de publicação
Naumann S.; Dick M.; Kern E.; Johann T.	The GREENSOFT Model: A reference model for green and sustainable software and its engineering	2011	Sustainable Computing: Informatics and Systems
Mahmoud S.S.; Ahmad I.	A green model for sustainable software engineering	2013	International Journal of Software Engineering and its Applications
Oyedeki S.; Seffah A.; Penzenstadler B.	A catalogue supporting software sustainability design	2018	Sustainability (Switzerland)
Chauhan N.S.; Saxena A.	A green software development life cycle for cloud computing	2013	IT Professional
Sriraman G.; Raghunathan S.	A Systems Thinking Approach to Improve Sustainability in Software Engineering—A Grounded Capability Maturity Framework	2023	Sustainability (Switzerland)

Tabela 1 – Artigos selecionados pelo autor para Revisão Sistemática

O artigo de Naumann *et al.* (2011) aborda o modelo GREENSOFT que é um modelo conceitual e, como observado na revisão bibliométrica, também o mais referenciado pela literatura referente a software verde, tendo como objetivo auxiliar desenvolvedores de software, administradores e usuários de software a criar, manter e utilizar software de maneira mais sustentável. Conforme pode ser observado na Figura 6, o modelo abrange um ciclo de vida holístico para produtos de software, critérios e métricas de sustentabilidade para produtos de software, modelos de procedimento para diferentes partes interessadas e recomendações de ação, bem como ferramentas que apoiam as partes interessadas no desenvolvimento, compra, fornecimento e uso de software de forma verde e sustentável.

Esse modelo de referência contém um Ciclo de Vida dos Produtos de Software. Em contraste com os ciclos de vida tradicionais de software, voltados para o Pensamento do Ciclo de Vida (LCT), que segue o lema “do berço ao túmulo”, o LCT tem o objetivo de avaliar a compatibilidade ecológica, social, humana e econômica de um produto durante todo o seu ciclo de vida. Ele começa nas primeiras etapas do desenvolvimento do produto e termina com o descarte e reciclagem do produto. As descobertas obtidas dessas avaliações podem ser usadas para otimizar o produto de maneira equilibrada ou para comparar um produto com seus concorrentes.

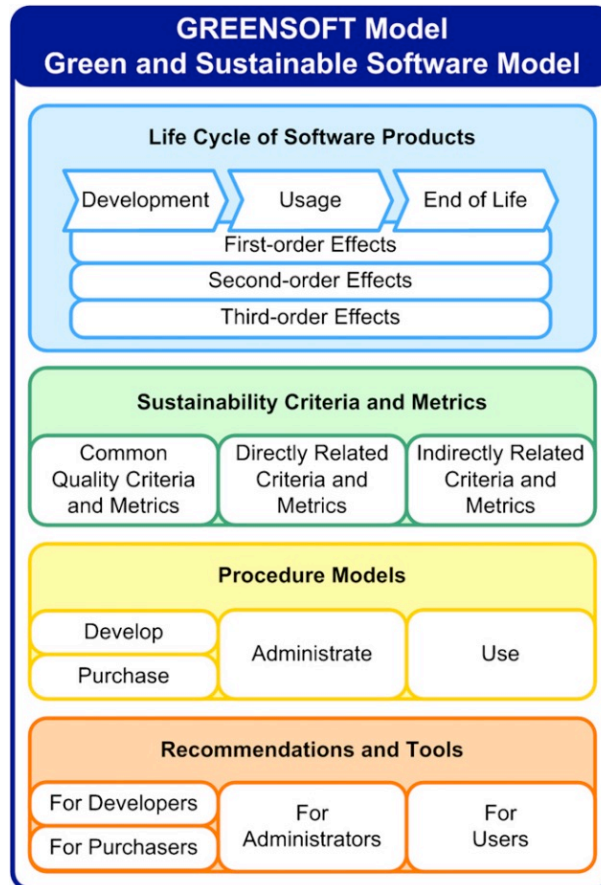


Figura 6 – O Modelo GREENSOFT, um modelo de referência para “Software Verde e Sustentável”

A segunda parte do Modelo GREENSOFT é chamada de Critérios e Métricas de Sustentabilidade. Ela abrange métricas e critérios comuns para a medição da qualidade do software e permite a classificação de critérios e métricas para avaliar a sustentabilidade de um produto de software. Critérios e métricas adequados podem incluir modelos para a medição da qualidade do software, modelos de procedimento para o desenvolvimento de software, bem como métodos emprestados da Análise do Ciclo de Vida (LCA). Aqui, são feitas distinções entre critérios e métricas diretos (relacionados a efeitos de primeira ordem) e aqueles que indiretamente dizem respeito à sustentabilidade (relacionados a efeitos de segunda e terceira ordem).

O componente Modelos de Procedimento possibilita classificar modelos de procedimento que abrangem aquisição e desenvolvimento de software, manutenção de sistemas de TI e suporte ao usuário. Como exemplo, foi proposta uma extensão genérica para processos de desenvolvimento de software ambíguos que permite a consideração sistemática de aspectos de sustentabilidade durante o desenvolvimento de software.

O último componente do modelo contém Recomendações e Ferramentas. Essas ferramentas auxiliam partes interessadas com diferentes níveis de habilidade profissional na aplicação de técnicas verdes ou sustentáveis em geral, no desenvolvimento, compra, administração ou uso de produtos de software. Os possíveis papéis incluem desenvolvedores de software, adquirentes de software, administradores, bem como usuários profissionais e particulares.

No artigo de Sriraman e Raghunatha (2023) os autores revisaram a literatura, modelos e práticas existentes, além de realizar pesquisas e entrevistas para compreender a realidade, práticas e desafios no campo da sustentabilidade de software. Eles identificaram um conjunto de questões de pesquisa para descobrir por que a sustentabilidade é importante, quais ações podem ser tomadas para melhorá-la e como e quando podem ser implementadas. De acordo com o estudo, a pesquisa existente é limitada por um foco estreito em domínios específicos, como meio ambiente e energia, ou em fases específicas do desenvolvimento de software. Portanto, o objetivo é abordar essas limitações por meio do proposto framework abrangente de capacidade de sustentabilidade de software (SSCF), que fornece um “prontuário” que permite a qualquer organização avaliar sua sustentabilidade de software atual e as capacidades e métricas nas quais podem se concentrar para melhorar a maturidade do software sustentável.

O modelo proposto de maturidade de capacidade é um framework que define as capacidades principais e práticas que as organizações precisam adotar para desenvolver software de forma sustentável. O modelo, apresentado na Figura 7, engloba uma variedade de métodos e abordagens que ajudam as organizações a reduzir o impacto ambiental do desenvolvimento de software, ao mesmo tempo em que melhoram a sustentabilidade econômica e social.

Como destaque, o framework define uma série de níveis de maturidade, desde *ad hoc* até líder, que as organizações podem usar para avaliar seu nível atual de maturidade na adoção de práticas operacionais de software sustentável.

O framework ainda define as capacidades e práticas essenciais que as organizações devem adotar para operar software de forma sustentável e as relaciona aos diferentes níveis de maturidade, e prove uma série de métricas de sustentabilidade que as organizações podem usar para medir e acompanhar seu progresso na busca de metas de sustentabilidade. Segundo os autores um roadmap claro com métricas potenciais fornece clareza e permite que as organizações foquem em resultados tangíveis que podem ser medidos.

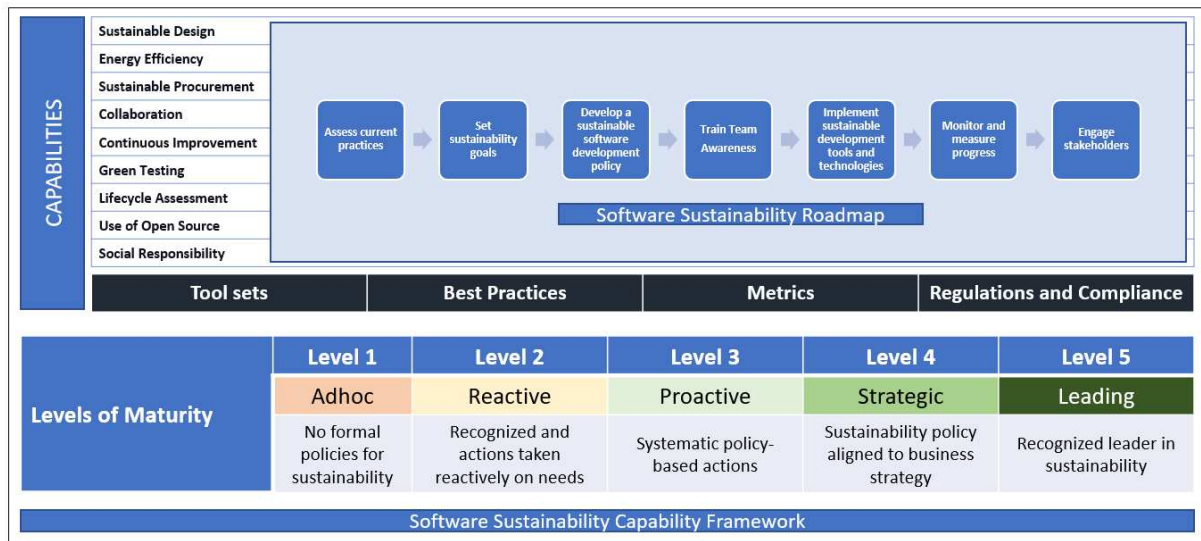


Figura 7 – Framework de Capacidade de Sustentabilidade de Software

Os autores concluem que o modelo de maturidade proposto não é focado em organizações de tamanho específico, mas é simples e adaptável para organizações de qualquer tamanho. O framework é construído com base em modelos existentes, onde o objetivo foi fornecer listas de verificação detalhadas e uma estrutura de avaliação para que as organizações possam iniciar sua jornada de sustentabilidade.

O artigo dos autores Chauhan e Saxena (2013) apresenta um framework de nuvem aprimorada que oferece uma visão holística do ambiente de nuvem e mapeia oportunidades de economia de energia para vários componentes de nuvem. O autor ressalta que frameworks de nuvem verde já foram propostos anteriormente, mas cobriam apenas as operações de virtualização e datacenter. Assim, a estrutura proposta nesse novo framework também abrange o desenvolvimento de software, o que pode melhorar significativamente a eficiência energética em ambientes de nuvem. Na Figura 8 é apresentado o Framework proposto pelos autores.

O artigo aborda duas questões relacionadas à computação em nuvem e eficiência energética. Em primeiro lugar, discute-se a importância dos Acordos de Nível de Serviço (SLAs) para estabelecer o nível de serviço esperado pelos clientes de provedores de serviços em nuvem. Os SLAs podem afetar a qualidade do serviço fornecido pelos provedores de nuvem e influenciar o consumo de energia da nuvem. A inclusão de termos e condições relacionados à energia nos SLAs permite rastrear o consumo de energia na nuvem e estabelecer limites e responsabilidades em relação ao uso de energia.

Em segundo lugar, destaca-se a importância do software, especialmente do software do sistema, para a eficiência energética na computação em nuvem. O design e a seleção adequada

do software do sistema, como sistemas operacionais e compiladores, podem melhorar significativamente a eficiência energética. Algoritmos de agendamento otimizados e esquemas de alocação de recursos podem contribuir para a eficiência energética. Além disso, o design do sistema operacional deve visar reduzir o tempo necessário para realizar cálculos e oferecer suporte à execução paralela por meio de multithreading.

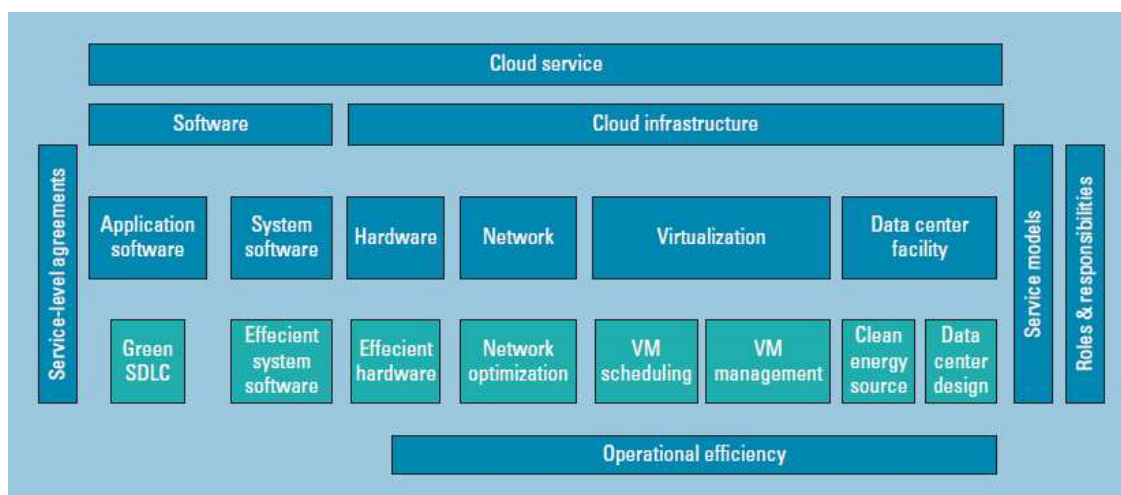


Figura 8 – Framework de computação em nuvem verde

No ambiente de nuvem, são comumente usadas máquinas virtuais (VMs) em vez de máquinas físicas dedicadas. Os desenvolvedores devem projetar o software das VMs e as instâncias dos sistemas operacionais com práticas de desenvolvimento de software energeticamente eficientes. Também é importante personalizar os elementos do sistema operacional, mantendo apenas aqueles necessários para executar a tarefa exigida. A criação de uma instância de VM leve pode economizar energia, reduzindo o tempo de inicialização e o armazenamento da imagem da VM.

Por fim, destaca-se que no ambiente de nuvem, as responsabilidades de desenvolvimento de aplicativos dependem do modelo de nuvem adotado. No caso do software como serviço (SaaS), o provedor de nuvem é responsável pelo desenvolvimento do software, enquanto na plataforma como serviço (PaaS), o provedor de nuvem fornece apenas a plataforma para o desenvolvimento de um aplicativo. Se o cliente tiver assinado uma infraestrutura como serviço (IaaS), todas as responsabilidades de desenvolvimento e manutenção de aplicativos são do cliente de nuvem.

No artigo de Mahmoud e Ahmad (2013) a contribuição é modelo que representa como obter um produto de software verde e sustentável cobrindo todo o ciclo de vida. Os autores recapitulam que o processo popular e genérico de engenharia de software é composto pelas

seguintes etapas: Requisitos, Desenho, Teste Unitário, Implementação, Teste Sistemático, Uso, Manutenção e Descarte. Reforçam ainda que existem muitos modelos com diferentes variações dessas etapas e muitas empresas adotam os seus próprios, mas todos tendem a ter padrões similares.

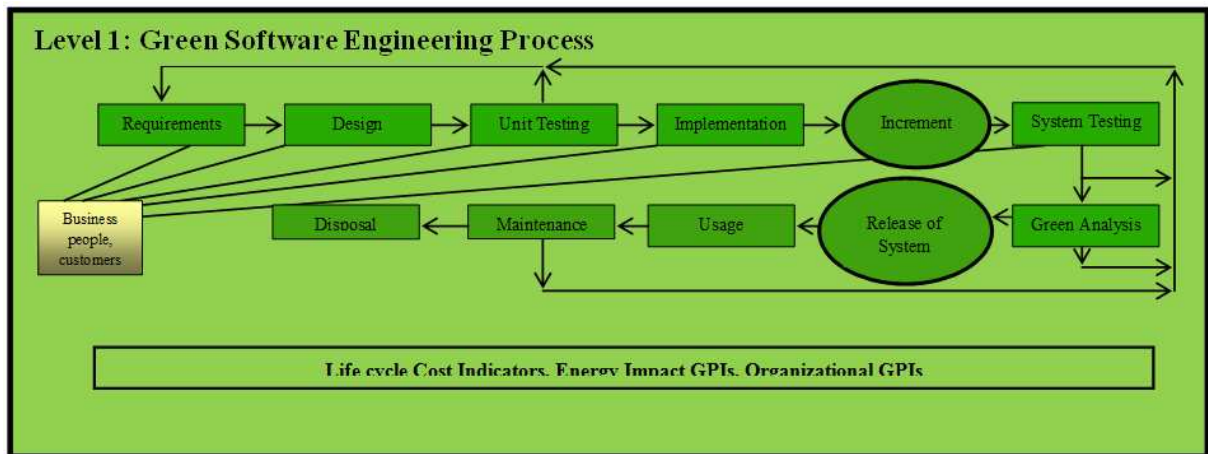


Figura 9 – Modelo de Processo de Engenharia de Software Verde

O estudo então propõe a adição de uma etapa de análise ambiental, chamada de estágio de “Análise Verde”, ao processo de engenharia de software para promover a eficiência energética, onde essa etapa busca considerar novas ideias sobre sustentabilidade ambiental que não foram previamente contempladas no processo de engenharia de software. Na Figura 9 pode-se observar o modelo proposto com o processo completo. Esse novo estágio da análise verde avalia o grau de sustentabilidade ambiental de cada incremento do software em desenvolvimento, atuando como uma etapa de teste específica para eficiência energética, e apoiado pela utilização de métricas para realização da análise.

O processo proposto dentro dessa etapa envolve:

- Definir quais métricas de recursos de TI e de qualidade devem ser utilizadas para o sistema desejado. Recomenda-se o uso de métricas de uso de recursos de TI, como uso da CPU, e métricas de qualidade, como desempenho;
- Coletar os dados do software e realizar as medições com base em fórmulas. Muitas ferramentas de software podem ser utilizadas nessa etapa para coletar os dados do hardware subjacente e mapear o desperdício de energia com as localizações no código.
- Analisar os resultados das fórmulas e determina se mudanças devem ocorrer nas etapas de requisitos, design ou implementação. Pode se tomar por base diretrizes definidas por

leis ou regulamentos relacionados à ecologia que estabelecem algum padrão para os níveis de energia que os sistemas devem seguir, auxiliando na tomada dessa decisão;

- Dependendo dos resultados, o modelo permite voltar ao estágio de requisitos para implementar mudanças que podem levar a um software mais eficiente em termos energéticos. Como os incrementos são pequenos, aplicar alterações não é caro. O software pode então ser liberado para os usuários. Para isso ocorrer, os autores destacam que a incorporação dos princípios ágeis no modelo proposto por eles, pois a mudança de requisitos, mesmo tarde no desenvolvimento, é um princípio ágil que pode promover a engenharia verde, pois é inevitável que ocorram mudanças e é necessário haver uma maneira energeticamente eficiente de lidar com esse problema.

Por fim, o artigo de Oyedeji, Seffah e Penzenstadler (2018) propõem um framework piloto que inclui um conjunto de metas, conceitos e métodos de sustentabilidade. Ele exemplifica como aplicar e quantificar a sustentabilidade.

Com base nas fases do ciclo de vida do desenvolvimento de software (SDLC), este framework piloto foi criado com o intuito de orientar as partes interessadas envolvidas no design e desenvolvimento de um sistema de software. A Figura 10 fornece um fluxo detalhado do framework piloto.

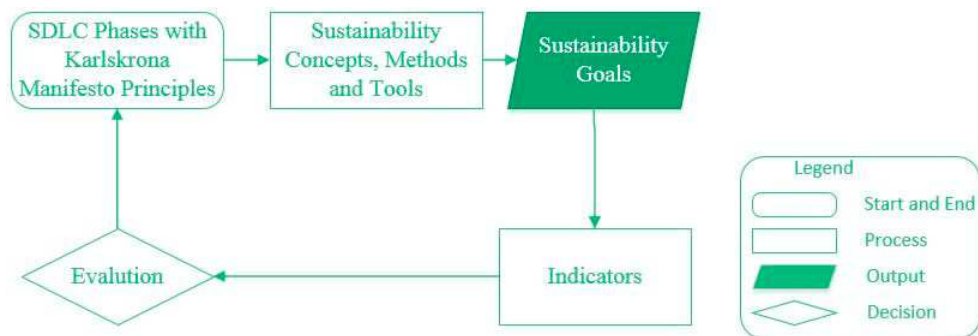


Figura 10 – Framework piloto para a sustentabilidade do design do sistema de software

O framework piloto é derivado do SDLC para auxiliar os desenvolvedores a incorporar sustentabilidade durante o design e desenvolvimento do sistema, abrangendo as fases do ciclo de vida do desenvolvimento de software. Para uma melhor compreensão, o framework piloto é apresentado abaixo em formato tabular para mostrar os conteúdos envolvidos no framework. É importante destacar que os indicadores usados no framework são influenciados pelos nove princípios do manifesto de Karlskrona (BECKER *et al.*, 2015) – manifesto que enfatiza a importância de incorporar considerações ambientais, sociais e econômicas no desenvolvimento

de software, além de promover a colaboração entre os envolvidos no processo de desenvolvimento, buscando incentivar a conscientização sobre a sustentabilidade e fornecer orientações para a criação de softwares mais sustentáveis.

6 Conclusões

Os artigos discutidos apresentam diferentes abordagens e propostas para o desenvolvimento de software verde e sustentável. No geral, eles mostram modelos diferentes para promover o desenvolvimento de software verde e sustentável, considerando aspectos como ciclo de vida do software, critérios de sustentabilidade, maturidade ágil e integração da sustentabilidade em diferentes fases do processo de desenvolvimento.

Contudo observa-se uma certa falta de aprofundamento nos aspectos sociais e humanos do desenvolvimento de software verde e sustentável. Os estudos poderiam explorar questões como:

- Impacto na qualidade de vida dos usuários: como o desenvolvimento de software verde pode contribuir para a melhoria da qualidade de vida dos usuários? Isso poderia envolver a consideração de aspectos como usabilidade, acessibilidade, segurança e privacidade dos sistemas de software.
- Inclusão social: como garantir que o desenvolvimento de software verde leve em consideração as necessidades e realidades de diferentes grupos sociais, incluindo pessoas com deficiência, comunidades marginalizadas e países em desenvolvimento? Isso pode envolver a adaptação de software para diferentes contextos culturais e socioeconômicos, além de garantir a acessibilidade para todos.
- Ética e responsabilidade social: quais são as considerações éticas e responsabilidades sociais dos desenvolvedores de software verde? Isso pode incluir a avaliação dos impactos sociais e éticos do software, como o uso de dados pessoais, vieses algorítmicos e questões relacionadas à segurança cibernética.
- Participação e engajamento dos usuários: como envolver os usuários no processo de desenvolvimento de software verde e sustentável? Isso pode incluir a co-criação de soluções, a incorporação de feedback dos usuários e a promoção da conscientização sobre a importância da sustentabilidade no uso de software.

Ao abordar esses aspectos sociais e humanos, os estudos podem fornecer uma visão mais abrangente do desenvolvimento de software verde e sustentável, considerando não apenas

os aspectos ambientais e econômicos, mas também seu impacto nas pessoas e na sociedade em geral.

Referências

BAMBAZEK, P.; GROHER, I.; SEYFF, N. **Requirements engineering for sustainable software systems: a systematic mapping study**. Requirements Engineering Springer Science and Business Media LLC, 7 jun. 2023.

BECKER, C. et al. **Sustainability Design and Software: The Karlskrona Manifesto**. 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, 2015.

BRUNDTLAND, G.H. **Our Common Future: Report of the World Commission on Environment and Development**. Geneva, UN-Dokument A/42/427, 1987. Disponível em: <http://www.un-documents.net/ocf-ov.htm>

CHAUHAN, N. S.; SAXENA, A. **A Green Software Development Life Cycle for Cloud Computing**. IT Professional Institute of Electrical and Electronics Engineers (IEEE), 2013.

IBRAHIM, S. R. A. et al. **The Development of Green Software Process Model**. International Journal of Advanced Computer Science and Applications. The Science and Information Organization, 2021.

MAHMOUD, S. S.; AHMAD, I. **A green model for sustainable software engineering**. International Journal of Software Engineering and Its Applications, v. 7, n. 4, p. 55-74, 2013.

MOURÃO, B. C.; KARITA, L.; DO CARMO MACHADO, I. **Green and Sustainable Software Engineering - a Systematic Mapping Study**. Proceedings of the XVII Brazilian Symposium on Software Quality. In: SBQS: 17TH BRAZILIAN SYMPOSIUM ON SOFTWARE QUALITY. ACM, 17 out. 2018.

NAUMANN, S. et al. **The GREENSOFT Model: A reference model for green and sustainable software and its engineering**. Sustainable Computing: Informatics and Systems. Elsevier BV, dez. 2011.

OYEDEJI, S.; SEFFAH, A.; PENZENSTADLER, B. **A Catalogue Supporting Software Sustainability Design**. Sustainability MDPI AG, 2018.

PENZENSTADLER, B. et al. **Sustainability in software engineering: a systematic literature review**. 16th International Conference on Evaluation & Assessment in Software Engineering, 2012.

SRIRAMAN, G.; RAGHUNATHAN, S. **A Systems Thinking Approach to Improve Sustainability in Software Engineering—A Grounded Capability Maturity Framework**. Sustainability MDPI AG, 2023.

SIERSZECKI, K. et al. **Green Software: Greening What and How Much?** IEEE Software Institute of Electrical and Electronics Engineers (IEEE), 2014.