



**MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE CIÊNCIAS DA SAÚDE DE PORTO ALEGRE
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM TECNOLOGIAS DA INFORMAÇÃO E
GESTÃO EM SAÚDE**

Mayandre Jacob Oliveira de Bona

**INTEGRANDO UM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE
COM OS OBJETIVOS DE DESENVOLVIMENTO SUSTENTÁVEL: UMA ANÁLISE
DOS FATORES DE IMPACTO NA SUSTENTABILIDADE**

Porto Alegre

2023

Mayandre Jacob Oliveira de Bona

**Integrando um Processo de Desenvolvimento de Software com os
Objetivos de Desenvolvimento Sustentável: uma Análise dos Fatores de
Impacto na Sustentabilidade**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Tecnologias da Informação e Gestão em Saúde da Universidade Federal de Ciências da Saúde de Porto Alegre, como requisito parcial para a obtenção do título de Mestre em Tecnologias da Informação e Gestão em Saúde.

**Orientadora: Prof. Dra. Juliana
Silva Herbert**

Universidade Federal das Ciências da Saúde de Porto Alegre

Porto Alegre – RS

2023

Catálogo na Publicação

Bona, Mayandre Jacob Oliveira de
Integrando um Processo de Desenvolvimento de Software
com os Objetivos de Desenvolvimento Sustentável: uma
Análise dos Fatores de Impacto na Sustentabilidade /
Mayandre Jacob Oliveira de Bona. -- 2023.
120 p. : il., graf., tab. ; 30 cm.

Dissertação (mestrado) -- Universidade Federal de
Ciências da Saúde de Porto Alegre, Programa de
Pós-Graduação em Tecnologias da Informação e Gestão em
Saúde, 2023.

Orientador(a): Prof. Dra. Juliana Herbert.

1. sustentabilidade. 2. objetivos do desenvolvimento
sustentável. 3. desenvolvimento de software. I. Título.

Sistema de Geração de Ficha Catalográfica da UFCSPA com os dados
fornecidos pelo(a) autor(a).

Mayandre Jacob Oliveira de Bona

**Integrando um Processo de Desenvolvimento de Software com os
Objetivos de Desenvolvimento Sustentável: uma Análise dos Fatores de
Impacto na Sustentabilidade**

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em
Tecnologias da Informação e Gestão em Saúde da Universidade Federal de
Ciências da Saúde de Porto Alegre, como requisito parcial para a obtenção do título
de Mestre em Tecnologias da Informação e Gestão em Saúde.

Orientadora: Profa. Dra. Juliana Silva Herbert.

Aprovado em 15 de agosto de 2023.

BANCA EXAMINADORA:

Prof. Dr. Guilherme Horta Travassos
Universidade Federal do Rio de Janeiro (UFRJ)

Prof. Dr. Diego Vallespir
Universidad de la República (Udelar)

Prof. Dr. Airton Telbon Stein
Universidade Federal de Ciências da Saúde de Porto Alegre (UFCSPA)

AGRADECIMENTOS

Agradeço a todos aqueles que estiveram ao meu lado nessa jornada árdua e desafiadora.

Especialmente à minha orientadora, professora Juliana Herbert, por ser um exemplo de grande profissional e pessoa, compreensiva, carinhosa, sempre disponível, por estar ao meu lado desde a graduação, me orientando em monitorias, Programa de Iniciação à Docência, no meu trabalho de conclusão de curso e até organizando viagens! Agradeço por todas as trocas de conhecimentos, pelos conselhos e dicas para a minha carreira profissional, por todos os votos de confiança ao longo da jornada do mestrado e, principalmente, por acreditar na minha capacidade em momentos que eu mesma duvidei.

Agradeço a Vanessa Siviero e ao Marcos Mendonça, por serem meus companheiros ao longo desta etapa. A parceria que construímos vai além da pós-graduação e sou muito feliz por termos nos conhecido e nos apoiado durante todo este tempo.

Agradeço aos meus amigos e familiares pela torcida, pelo apoio, carinho e por compreender as minhas ausências constantes em suas vidas enquanto eu precisava dedicar boa parte do meu tempo a esta pesquisa.

“Tem quem analisa, tem quem banaliza, [...] quem fica de cima e não vê.

Não esqueça que tem poder para mudar o mundo, você tem poder para mudar o mundo! [...]”

(BaianaSystem. **Bola de Cristal**. Máquina de Louco/Universal MGB/Som Livre. 2019)

RESUMO

INTRODUÇÃO: O conceito de sustentabilidade não é restrito apenas a questões ambientais, podendo ser expandido a contextos sociais, econômicos, políticos dentre outros. Em virtude desta ampla gama de aplicações, a Organização das Nações Unidas (ONU) desenvolveu a Agenda 2030, composta pelos Objetivos do Desenvolvimento Sustentável (ODSs) associados a ações em diversas áreas para promover a melhoria da qualidade de vida desta geração e das futuras. Considerando que os ODSs envolvem diversos fatores de impacto social, ambiental e econômico, propõe-se aqui uma associação entre ODSs e desenvolvimento de soluções em software visando a sustentabilidade como uma forma de contribuição para o alcance destes Objetivos. **OBJETIVO:** Elaborar um framework que ofereça suporte ao desenvolvimento de sistemas de software autossustentáveis, bem como à promoção da sustentabilidade por meio do software. **MÉTODO:** Foi realizado um estudo de revisão sistemática da literatura para embasamento sobre como o tema sustentabilidade era considerado em soluções em saúde digital. Em seguida, estabeleceu-se relações entre softwares e sustentabilidade para responder à pergunta de pesquisa “como um software pode ser sustentável e contribuir à sustentabilidade?”. Para tal, foram selecionados 7 ODSs, princípios da ISO 25010:2011 de qualidade de software e fatores do processo de desenvolvimento de software. A partir desta etapa foi desenvolvido um framework visando o conceito de sustentabilidade pelo software. Foi conduzida a avaliação qualitativa do framework proposto com um grupo focal selecionado, para adequação e refinamentos. **RESULTADOS:** Evidenciou-se, desde a realização do estudo de revisão sistemática, a carência bibliográfica sobre o tema sustentabilidade pelo software. Os artefatos do framework foram refinados conforme necessidade, após a avaliação com grupo focal, e disponibilizados em um site, para que todos os interessados possam acessar e incorporar fatores de sustentabilidade pelo software em seus projetos. **CONCLUSÕES:** Foi construído um framework associando requisitos que devem ser considerados num processo de desenvolvimento de software com ODSs selecionados e princípios da ISO 25010:2011, para ser utilizado no desenvolvimento de novos projetos ou na avaliação de um software já construído, com o intuito de auxiliar pesquisadores, profissionais da área de tecnologia da informação e comunicações e demais stakeholders interessados à buscarem a incorporação do conceito de sustentabilidade pelo software em seus projetos.

Palavras-chave: sustentabilidade; objetivos do desenvolvimento sustentável; desenvolvimento de software.

ABSTRACT

INTRODUCTION: The concept of sustainability is not restricted to environmental issues; it extends to social, economic, political contexts, among others. Due to its broad applicability, the United Nations (UN) developed the 2030 Agenda, comprising the Sustainable Development Goals (SDGs), which encompass actions in various areas to enhance the quality of life for current and future generations. Given that the SDGs involve multiple factors with social, environmental, and economic impacts, this paper proposes an association between SDGs and the development of software solutions aimed at sustainability to contribute to achieving these Objectives. **OBJECTIVE:** The objective is to develop a framework that supports the development of self-sustainable software systems and promotes sustainability through software. **METHOD:** A systematic literature review was conducted to understand how sustainability has been considered in digital health solutions. Relationships between software and sustainability were established to address the research question: "How can software be sustainable and contribute to sustainability?" A framework was then developed based on the concept of sustainability through software. The proposed framework was qualitatively evaluated with a selected focus group to ensure appropriateness and refine it further. **RESULTS:** The systematic review study revealed a lack of literature on the topic of sustainability through software. The artifacts produced within the framework were refined as necessary based on the evaluation with the focus group. These artifacts were made available on a website, enabling interested parties to access and incorporate sustainability factors through software into their projects. **CONCLUSIONS:** A framework was created that associates requirements to be considered in the software development process with selected SDGs and ISO 25010:2011 principles. This framework can be utilized in the development of new projects or in evaluating existing software, offering guidance to researchers, IT professionals, and other stakeholders interested in incorporating the concept of sustainability through software into their projects.

Keywords: sustainability; sustainable development goals; software development.

SUMÁRIO

1 INTRODUÇÃO.....	10
2 OBJETIVOS.....	12
2.1 OBJETIVO GERAL	12
2.2 OBJETIVOS ESPECÍFICOS	12
3 REFERENCIAL TEÓRICO	13
3.1 ENGENHARIA DE SOFTWARE	13
3.2 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	14
3.3 SUSTENTABILIDADE	16
3.4 OBJETIVOS DO DESENVOLVIMENTO SUSTENTÁVEL - AGENDA 2030.....	17
3.5 SAÚDE DIGITAL E SUSTENTABILIDADE	22
4 MÉTODO	24
4.1 REVISÃO SISTEMÁTICA DA LITERATURA	24
4.2 DEFINIÇÃO DAS RELAÇÕES ENTRE SOFTWARE E SUSTENTABILIDADE	25
4.3 AVALIAÇÃO QUALITATIVA.....	26
5 RESULTADOS	30
5.1 RESULTADOS DA AVALIAÇÃO QUALITATIVA	30
5.2 ARTIGO	38
5.3 ARTIGO DE REVISÃO SISTEMÁTICA.....	38
6 CONSIDERAÇÕES FINAIS.....	39
REFERÊNCIAS	41
APÊNDICE A – ODS 4, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	43

APÊNDICE B - ODS 7, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	46
APÊNDICE C - ODS 8, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	50
APÊNDICE D - ODS 9, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	53
APÊNDICE E - ODS 10, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	57
APÊNDICE F - ODS 12, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	64
APÊNDICE G - ODS 17, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	70
APÊNDICE H – CHECKLIST PARA DESENVOLVIMENTO DE SOFTWARE	74
APÊNDICE I – CHECKLIST PARA AVALIAÇÃO DE SUSTENTABILIDADE PELO SOFTWARE EM UM SOFTWARE JÁ CONSTRUÍDO	79

1 INTRODUÇÃO

Rotineiramente utilizamos soluções em software para facilitar nosso dia a dia. A adoção de soluções de software em saúde, por exemplo, já é amplamente difundida e adotada por instituições, profissionais e pacientes, indo ao encontro da era de digitalização em que vivemos. As soluções utilizadas são as mais variadas possíveis, indo de aplicações simples voltadas aos usuários para monitorar sua própria saúde, até complexas aplicações utilizadas para gestão de serviços de saúde, por exemplo. Os benefícios de sua utilização são diversos, como empoderamento do paciente, acesso a seus dados, controles de estoque e finanças, dentre muitos outros. O uso de saúde digital pode expandir o acesso à saúde de todos, independente dos locais onde vivem (MITCHELL, *et al.*, 2019).

A Organização das Nações Unidas (ONU) propôs os Objetivos de Desenvolvimento Sustentável (ODSs), no contexto da Agenda 2030, visando melhorar as condições de vida da população mundial, econômica, social e ambientalmente. Estes ODSs foram pensados para serem adotados por todos os países, de maneira universal, o que os torna um tanto quanto ousados. Eles “se baseiam nos princípios de equidade (não deixar ninguém para trás), direitos humanos, prestação de contas e sustentabilidade” (OLU *et al.*, 2019).

Há um grande desafio na obtenção de êxito em atingir os ODSs propostos. Portanto, medidas que contribuam para o alcance dos ODSs são necessárias mesmo que em microcenários. Tendo isso em mente, dá-se a importância de desenvolver soluções em software para saúde voltadas à sustentabilidade. “Embora haja um consenso crescente de que a saúde digital será uma mudança positiva na prestação de cuidados de saúde em países de baixa e média renda, ainda há desafios significativos para sua implementação e crescimento” (MITCHELL *et al.*, 2019).

Existe uma carência bibliográfica envolvendo o tema software e sustentabilidade de publicações que relacionem o processo de desenvolvimento de software com a sustentabilidade, a fim de contribuir para que desenvolvedores de software e profissionais da saúde possam utilizá-lo como base ao propor novas soluções ou, até mesmo, para a adequação de soluções já existentes, garantindo que este produto de software estará em consonância com os ODSs.

Os desafios ambientais, sociais e econômicos enfrentados atualmente fazem da sustentabilidade uma preocupação global. Nesse contexto, o setor de tecnologia desempenha um papel fundamental na busca de soluções sustentáveis. O desenvolvimento de software, em particular, apresenta uma oportunidade significativa para promover a sustentabilidade, pois pode impactar diretamente a forma como uma organização opera e como ela interage com o meio ambiente.

A sustentabilidade é uma meta amplamente reconhecida, mas muitas vezes negligenciada no processo de desenvolvimento de software. Muitas empresas e desenvolvedores se concentram principalmente na funcionalidade e eficiência do software, ignorando o impacto de longo alcance que isso pode ter no meio ambiente e na sociedade. Portanto, é importante considerar e entender como o software pode ser projetado e desenvolvido de forma sustentável, alinhado aos Objetivos de Desenvolvimento Sustentável (ODSs).

Ao vincular elementos do processo de desenvolvimento de software aos ODSs e a norma ISO 25010:2011, este trabalho fornece uma abordagem prática para quaisquer interessados em desenvolver projetos de software. Ao fazê-lo, esperamos enfatizar a importância de considerar não apenas as funcionalidades e a eficiência do software, mas também os impactos sociais e ambientais que um software pode ter.

Ao mostrar como o desenvolvimento de sustentabilidade pelo software pode contribuir para a sustentabilidade em geral, este trabalho oportuniza a conscientização sobre o assunto e incentiva a adoção de práticas mais responsáveis no setor de tecnologia. Além disso, fornece insights valiosos aos profissionais de software, incentivando-os a considerar a sustentabilidade como parte integrante de seus projetos e a buscar soluções inovadoras que possam promover um futuro mais sustentável para todos.

Este trabalho, portanto, contribui para o preenchimento das lacunas existentes na literatura, vinculando os ODSs e os princípios da norma ISO 25010:2011 ao processo de desenvolvimento de software e fornecendo uma perspectiva prática e abrangente sobre a sustentabilidade pelo software. Espera-se que as proposições apresentadas tenham um impacto positivo na área de tecnologia e conduzam mudanças significativas para um desenvolvimento de software mais sustentável e responsável.

2 OBJETIVOS

2.1 OBJETIVO GERAL

Elaborar um framework que ofereça suporte ao desenvolvimento de sistemas de software autossustentáveis, bem como à promoção da sustentabilidade por meio do software.

2.2 OBJETIVOS ESPECÍFICOS

1. Entender como os ODSs estão relacionados ao desenvolvimento de software.
2. Caracterizar fatores de desenvolvimento de software relacionados à sustentabilidade, considerando tanto o software sustentável quanto o software para sustentabilidade.
3. Definir características do processo de desenvolvimento de software que considere os ODSs sob o ponto de vista de desenvolvedores de software.
4. Definir um modelo de avaliação de software com relação aos ODSs.
5. Propor uma forma de avaliação do processo de desenvolvimento de software e de softwares já construídos, através de checklists, compostos por critérios e itens de verificação relacionados à sustentabilidade.
6. Realizar a avaliação dos checklists.

3 REFERENCIAL TEÓRICO

O objetivo do referencial teórico é apresentar conceitos que são necessários para o desenvolvimento da pesquisa. A seguir são listados os conceitos-chave necessários para o embasamento deste estudo. Estes conceitos foram elencados com o intuito de relacioná-los durante a pesquisa, portanto Engenharia de Software, Processo de desenvolvimento de software, Sustentabilidade e os Objetivos de Desenvolvimento Sustentável são conceitos indispensáveis neste caso. Adicionalmente, Saúde Digital e Sustentabilidade foi um conceito escolhido como um nicho de aplicabilidade das Tecnologias da Informação para a proposição do estudo de revisão sistemática, conforme o contexto do Programa de Pós-Graduação.

3.1 ENGENHARIA DE SOFTWARE

Uma das definições de Engenharia de Software a conceitua como uma área que abrange “todos os aspectos de produção de software”, tendo como objetivo apoiar o desenvolvimento profissional de software, incluindo técnicas de especificação, projeto e evolução de sistemas (SOMMERVILLE, 2011). Esta área vai além das características técnicas de um produto de software, englobando, também, o gerenciamento do projeto e desenvolvimento de outros fatores que apoiam a produção de software, tais como: ferramentas, métodos e teorias (SOMMERVILLE, 2011).

Praticar engenharia de software é resolver problemas seguindo um conjunto de princípios básicos. Esta prática possibilita a garantia de qualidade na construção de sistemas complexos dentro dos prazos previstos, de acordo com processos, métodos e ferramentas (PRESSMAN, 2011).

A engenharia de software consiste na aplicação de habilidades computacionais para abordar desafios em diversas disciplinas, resultando em soluções robustas, confiáveis e de fácil manutenção (LAWLOR *et al.*, 2015). Sistemas de software podem ser caros, complexos e necessitam entregar valor a um usuário final, portanto é necessário que sua manutenção seja simplificada e que tenham alto grau de confiabilidade.

Ainda de acordo com Sommerville (2011), um bom produto de software possui as características: manutenibilidade, confiança e proteção, eficiência e aceitabilidade. Sistemas de software precisam evoluir conforme as necessidades dos clientes ou

usuários. Portanto, devem ser desenvolvidos de forma a facilitar sua manutenção. Quando falamos em confiabilidade e proteção, cabe destaque à segurança das informações contidas em um sistema. Os dados dos usuários devem estar protegidos, tornando o sistema confiável e não causando prejuízos com seu uso. Uma solução construída de maneira eficiente não desperdiça recursos de programação e uso de hardware. Garantir a aceitabilidade de um produto de software por seus usuários é uma premissa básica para o sucesso de um projeto.

A engenharia de software abrange uma ampla gama de elementos e fases, incluindo gerenciamento, modelagem, projeto e codificação (ZHU *et al.*, 2006). As funções compreendidas pela engenharia de software, como por exemplo: organização de *sprints*, criação de casos de teste, desenhos e refatorações, visam a compreensão e expressão de atributos para otimizar as decisões que maximizam os resultados almejados por aquele projeto (OZKAYA, 2021).

Podemos citar, ainda, duas importantes razões para a prática da engenharia de software:

1. Cada vez mais, indivíduos e sociedades dependem dos sistemas de software avançados. Temos de ser capazes de produzir sistemas confiáveis econômica e rapidamente.
2. Geralmente é mais barato, a longo prazo, usar métodos e técnicas da engenharia de software para sistemas de software, em vez de simplesmente escrever os programas como se fossem algum projeto pessoal. Para a maioria dos sistemas, a maior parte do custo é mudar o software depois que ele começa a ser usado (SOMMERVILLE, 2011, p. 5).

3.2 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

Podemos usar como definição de processo: identificação de uma sequência de etapas que envolve a execução de atividades, considerando restrições e recursos específicos, com o objetivo de alcançar um propósito definido (ISO/IEC/IEEE, 2017).

A seguir são apresentados quatro itens fundamentais em processos de desenvolvimento de software, independentemente do tipo de software trabalhado:

1. Especificação de software, em que clientes e engenheiros definem o software a ser produzido e as restrições de sua operação.
2. Desenvolvimento de software, em que o software é projetado e programado.
3. Validação de software, em que o software é verificado para garantir que é o que o cliente quer.

4. Evolução de software, em que o software é modificado para refletir a mudança de requisitos do cliente e do mercado (SOMMERVILLE, 2011, p. 6).

Cada projeto de software contará com processos específicos que atendam a realidade do contexto em que está inserido. Um processo de desenvolvimento de software não deve ser rígido, mas sim adaptável para as tarefas que uma equipe de desenvolvimento irá executar (PRESSMAN, 2011).

Existem modelos clássicos de processo de desenvolvimento de software, como os modelos cascata, espiral e iterativo-incremental. O modelo cascata prevê que todas as atividades do processo de desenvolvimento do software sejam definidas e planejadas antes do início do trabalho de desenvolvimento em si (SOMMERVILLE, 2011). Ainda de acordo com Sommerville (2011), no modelo espiral, cada volta simboliza uma etapa do processo de desenvolvimento de software, contendo a natureza iterativa da prototipação unida a aspectos sistemáticos e controlados como no modelo cascata (PRESSMAN, 2011), possibilitando, inclusive, entregas de versões do projeto.

Já no modelo iterativo-incremental, conforme Pressman (2011), seu foco é direcionado à entrega de um produto operacional em cada incremento realizado, ou seja, as atividades não precisam ser totalmente definidas antes do início do desenvolvimento, o planejamento se dá a cada ciclo de incremento e a especificação é gerada em conjunto com a codificação. Algumas vantagens desse modelo é que ele não utiliza protótipos, mas sim entregas de partes reais do sistema, assim os clientes envolvidos podem obter ganhos dessas entregas (SOMMERVILLE, 2011).

O modelo de processo iterativo-incremental é amplamente utilizado atualmente, sendo frequentemente considerado no contexto dos métodos ágeis. Neste contexto, a documentação do sistema é simplificada, definindo os requisitos mais importantes ao projeto (SOMMERVILLE, 2011). O *Scrum* é um dos frameworks mais utilizados no contexto ágil. De forma simplificada, pode-se dizer que, usando *Scrum*, divide-se um tempo para execução de determinadas tarefas numa *sprint*, baseado num *backlog* que é a lista com as tarefas a serem desenvolvidas pela equipe, a qual interage e se comunica diariamente através de reuniões curtas para melhorar a comunicação e viabilizar a resolução de impedimentos e dificuldades de cada membro da equipe (PRESSMAN, 2011).

3.3 SUSTENTABILIDADE

O conceito de sustentabilidade vem sendo usado de forma indistinta (FAUSTINO *et al.*, 2016). Este fato pode causar certa confusão ao compreender o que este conceito significa em determinado cenário, a ausência de um quadro de referência teórico abrangente que possa relacionar de forma sistemática as diversas contribuições dos discursos e campos de conhecimento específicos evidencia a falta de precisão do conceito de sustentabilidade (RATTNER, 2006).

O mais importante avanço na evolução do conceito de sustentabilidade é representado pelo consenso crescente que esta requer e implica democracia política, equidade social, eficiência econômica, diversidade cultural, proteção e conservação do meio ambiente (RATTNER, 2006).

Podemos entender a sustentabilidade como um conceito que abrange aspectos econômicos, sociais, culturais e ambientais da sociedade humana (LIMA, 2006).

Desenvolvimento sustentável é um termo que surge com o propósito de que a geração atual consiga atender suas necessidades sem esgotar recursos para as próximas gerações, considerando que o planeta em que vivemos tem suas limitações, como recursos naturais, o capital humano e econômico e a capacidade da biosfera em lidar com emissões e resíduos (LARREA-GALLEGOS *et al.*, 2022). Para que este conceito seja implementado na prática, é necessário considerar diversos fatores, como: meio ambiente, economia e sociedade (LARREA-GALLEGOS *et al.*, 2022). Ainda de acordo com Larrea-Gallegos *et al.* (2022) este conceito pode ser adaptado para ser aplicado a diversas áreas.

Trazendo os conceitos de sustentabilidade para as áreas de tecnologia, Kunkel *et al.* (2021) afirmam que pesquisas transdisciplinares devem explorar e promover formas alternativas de pensar e implementar a digitalização, contando com o envolvimento de formuladores de políticas, empresas e sociedade civil, considerando que a tendência da era digital em que vivemos mudará a forma como o desenvolvimento econômico acontece e, portanto, o avanço da digitalização urge ser sustentável a fim de alcançar outros objetivos de sustentabilidade, como a mitigação das mudanças climáticas (ODS 13) (KUNKEL *et al.*, 2021).

3.4 OBJETIVOS DO DESENVOLVIMENTO SUSTENTÁVEL - AGENDA 2030

Há mais de três décadas, o desenvolvimento sustentável tem sido um objetivo político de grande importância para a comunidade internacional (BONNEDAHL *et al.*, 2022).

Durante a 70ª Assembleia Geral da Organização das Nações Unidas, que contou com a presença de lideranças de 193 países, em 25 de setembro de 2015, foram propostos os Objetivos do Desenvolvimento Sustentável - Agenda 2030, alinhados com os Objetivos de Desenvolvimento do Milênio (ODM). Foram definidos 17 Objetivos de Desenvolvimento Sustentável, detalhados em 169 metas, propondo que todos os países conseguissem alcançar os objetivos e metas até 2030. O cumprimento destes objetivos culminaria na necessidade de os países realizarem mudanças significativas, dentro de uma curta lacuna temporal, no contexto de uso de recursos, tecnologias, infraestrutura, organizações e relações sociais (BIGLARI *et al.*, 2021).

A concretização dos ODSs requer estratégias, práticas e medidas que ultrapassem as fronteiras nacionais, uma vez que esses objetivos são globais (MOHR *et al.*, 2021 *apud* KOLK, 2016; VAN TULDER *et al.*, 2021; VAN ZANTEN E VAN TULDER, 2018)¹. Porém, conforme Biglari *et al.* (2021), diversas nações não obtiveram o sucesso almejado na construção dos processos que levariam ao sucesso ao atingir os objetivos. Podemos citar, como razões para isso:

O fracasso financeiro de governos e setores privados em investir no desenvolvimento sustentável, particularmente em países de baixa renda, barreiras na transferência de tecnologia de regiões desenvolvidas para regiões menos desenvolvidas, impedimentos comerciais que desfavorecem países de renda média e baixa, fracasso governamental em estabelecer sistemas integrados de longo prazo, planos de desenvolvimento sustentável e falta de parcerias coerentes, conceber métricas personalizadas pouco propícias e mecanismos de avaliação inadequados e incompetentes (BIGLARI *et al.*, 2021).

¹MOHR, A., SCHUMACHER, C., KIEFNER, V. **Female executives and multinationals' support of the UN's sustainable development goals.** Journal of World Business, v. 57. 2021. DOI: <https://doi.org/10.1016/j.jwb.2021.101304> *apud* Kolk, A. **The social responsibility of international business: From ethics and the environment to CSR and sustainable development.** Journal of World Business, v. 51, p. 23–34. 2016. Van Tulder, R., Rodrigues, S. B., Mirza, H., & Sexsmith, K. **The UN's Sustainable Development Goals: Can multinational enterprises lead the Decade of Action?** Journal of International Business Policy, v. 4, p. 1–21. 2021. van Zanten, J. A., & van Tulder, R. **Multinational enterprises and the Sustainable Development Goals: An institutional approach to corporate engagement.** Journal of International Business Policy, v. 1, p. 208–233. 2018.

Salo *et al.* (2022) afirmam que nenhum país no mundo está a caminho de alcançar os ODSs até 2030. De acordo com os relatórios da ONU sobre o progresso em relação aos ODSs, constata-se que o desenvolvimento tomou uma direção contrária (BONNEDAHL *et al.*, 2022). Os cenários desiguais entre as regiões do mundo contribuem para dificultar a obtenção de êxito em atingir os objetivos. Países subdesenvolvidos, em desenvolvimento e desenvolvidos possuem grandes disparidades em termos da disponibilidade de recursos para trabalhar os ODSs. Apenas esforços nacionais individuais e isolados não são suficientes para atingir os ODSs, segundo (BIGLARI *et al.*, 2021). As metas estipuladas são ambiciosas e necessitam que sejam estabelecidas parcerias globais que promovam o compartilhamento de conhecimento, transferência de tecnologia e facilitem o acesso a recursos financeiros, especialmente para países subdesenvolvidos e em desenvolvimento (BIGLARI *et al.*, 2021 *apud* GA, 2015; STAFFORD-SMITH *et al.*, 2017; TABEAU E MEIJL, 2020)². Não podemos deixar de destacar a pandemia causada pelo vírus da Covid-19 que impactou o mundo todo de diversas formas negativas.

Outro ponto de destaque é que os 17 Objetivos de Desenvolvimento Sustentável estão interligados de alguma forma. Ou seja, a implementação de um Objetivo impacta em outro (BIGLARI *et al.*, 2021).

Os Objetivos e metas de Desenvolvimento Sustentável não são independentes e, pelo contrário, geralmente implicam interações na forma de sinergias (enquanto o cumprimento de um objetivo facilita o cumprimento de outro) ou tradeoffs (enquanto o cumprimento de um objetivo impede o progresso no outro) (BIGLARI *et al.*, 2021 *apud* Nilsson *et al.*, 2016; Pradhan *et al.*, 2017; Waage *et al.*, 2015)³.

² BIGLARI, Shirin; BEIGLARY, Shevva; ARTHANARI, Arthanari. **Achieving sustainable development goals: Fact or Fiction?**. *Journal of Cleaner Production*, v. 332. 2021. DOI: <https://doi.org/10.1016/j.jclepro.2021.130032> *apud* GA, U.N. **Transforming Our World: the 2030 Agenda for Sustainable Development**. Div. Sustain. Dev. Goals, New York, NY, USA. 2015. Retrieved from. [https://sustainabledevelopment.un.org/content/documents/21252030 Agenda for Sustainable Development web.pdf](https://sustainabledevelopment.un.org/content/documents/21252030%20Agenda%20for%20Sustainable%20Development%20web.pdf). (Accessed November 2020). Stafford-Smith, M., Griggs, D., Gaffney, O., Ullah, F., Reyers, B., Kanie, N., Stigson, B., Shrivastava, P., Leach, M., O'Connell, D. **Integration: the key to implementing the sustainable development goals**. *Sustain. Sci.* 12, p. 911–919. 2017. Tabeau, A., Meijl, H. Van. **Snakes and Ladders : World Development Pathways Synergies and Trade-Offs through the Lens of the Sustainable Development Goals**. 267. 2020. <https://doi.org/10.1016/j.jclepro.2020.122147>

³ BIGLARI, Shirin; BEIGLARY, Shevva; ARTHANARI, Arthanari. **Achieving sustainable development goals: Fact or Fiction?**. *Journal of Cleaner Production*, v. 332. 2021. DOI: <https://doi.org/10.1016/j.jclepro.2021.130032> *apud* Nilsson, M., Griggs, D., Visbeck, M. **Policy: map the interactions between sustainable development goals**. *Nature* v. 534, p. 320–322. 2016. Pradhan, P., Costa, L., Rybski, D., Lucht, W., Kropp, J.P. **A systematic study of Sustainable Development Goal (SDG) interactions**. *Earth's Futur* v. 5, p. 1169–1179. 2017. Waage, J., Yap, C., Bell, S., Levy, C., Mace, G., Pegram, T., Unterhalter,

Logo, pode-se considerar frustradas as tentativas de atingir cada meta individualmente, ignorando as dependências entre os ODSs, podendo ser chamados de insucessos dos Objetivos da Agenda 2030 (BIGLARI *et al.*, 2021 *apud*. COLSTE, 2017)⁴.

Salo *et al.* (2022) trazem em seu estudo que, considerando as dificuldades enfrentadas por todos os países em atingir os Objetivos do Desenvolvimento Sustentável até 2030, devemos nos concentrar nas mudanças menores, porém positivas, que foram impulsionadas ao tentar atingir os ODSs, que, para muitas nações, já são grandes ganhos.

No presente trabalho selecionamos alguns ODSs e metas nos quais identificamos meios para incorporar, durante o processo de desenvolvimento de software, fatores que contribuam para seu alcance. A Figura 1 apresenta os ODSs escolhidos. Todos os ODSs acabam se interligando de alguma maneira, porém alguns não se encaixariam nesse processo de desenvolvimento de software, por exemplo, Objetivo 2: “Acabar com a fome, alcançar a segurança alimentar e melhoria da nutrição e promover a agricultura sustentável” (ONU, s.d.), como estamos abordando um processo genérico de desenvolvimento de qualquer software, alguns ODSs, como o citado, requereriam muita especificidade se tratando de um software, sendo contemplado no propósito geral do software, como o Objetivo 3: “Assegurar uma vida saudável e promover o bem-estar para todas e todos, em todas as idades”, onde qualquer software voltado a área da saúde poderia dizer-se que contribui ao ODS 3.

O Objetivo 4: “Assegurar a educação inclusiva e equitativa e de qualidade, e promover oportunidades de aprendizagem ao longo da vida para todas e todos” (ONU, s.d.), versa sobre educação e acreditamos que seja possível incluir fatores educativos num software, não apenas para os usuários finais, mas também para a equipe de desenvolvimento e stakeholders envolvidos no processo. Destacamos a Meta 4.7,

E., Dasandi, N., Hudson, D., Kock, R. Governing the UN sustainable development goals: interactions, infrastructures, and institutions. *Lancet Glob. Heal.* v. 3, p. 251–e252. 2015.

⁴ BIGLARI, Shirin; BEIGLARY, Shevva; ARTHANARI, Arthanari. **Achieving sustainable development goals: Fact or Fiction?**. *Journal of Cleaner Production*, v. 332. 2021. DOI: <https://doi.org/10.1016/j.jclepro.2021.130032> *apud* Collste, D. **Policy coherence to achieve the SDGs: using integrated simulation models to assess effective policies**. *Sustain. Sci.* 2017. <https://doi.org/10.1007/s11625-017-0457-x>.

em que a contribuição a ela se dá pela disseminação do conhecimento dos ODSs durante o processo de desenvolvimento de software, por exemplo.

4.7 Até 2030, garantir que todos os alunos adquiram conhecimentos e habilidades necessárias para promover o desenvolvimento sustentável, inclusive, entre outros, por meio da educação para o desenvolvimento sustentável e estilos de vida sustentáveis, direitos humanos, igualdade de gênero, promoção de uma cultura de paz e não violência, cidadania global e valorização da diversidade cultural e da contribuição da cultura para o desenvolvimento sustentável (ONU, s.d.).

O Objetivo 7 trata das questões energéticas, diz: “Assegurar o acesso confiável, sustentável, moderno e a preço acessível à energia para todas e todos” (ONU, s.d.) entendemos que para isso são necessárias políticas públicas e em algumas das metas do Objetivo, também medidas da iniciativa privada poderiam contribuir, as quais não conseguiríamos discorrer sobre como incorporar especificamente no processo de desenvolvimento de software. Já na Meta “7.3 Até 2030, dobrar a taxa global de melhoria da eficiência energética” (ONU, s.d.) entendemos que podem haver contribuições do ponto de vista de evitar o desperdício de energia, por exemplo, através de otimizações do processo de desenvolvimento e do produto final de software, em si. Esta meta relaciona-se estreitamente com a Meta 8.4. Além deste também acreditamos na possibilidade de incorporar o Objetivo 8 e as metas:

Objetivo 8. Promover o crescimento econômico sustentado, inclusivo e sustentável, emprego pleno e produtivo e trabalho decente para todas e todos

8.2 Atingir níveis mais elevados de produtividade das economias por meio da diversificação, modernização tecnológica e inovação, inclusive por meio de um foco em setores de alto valor agregado e dos setores intensivos em mão de obra

8.4 Melhorar progressivamente, até 2030, a eficiência dos recursos globais no consumo e na produção, e empenhar-se para dissociar o crescimento econômico da degradação ambiental, de acordo com o Plano Decenal de Programas sobre Produção e Consumo Sustentáveis, com os países desenvolvidos assumindo a liderança

O Objetivo 9 e sua Meta 9.c podem ser facilmente relacionados a processos de desenvolvimento de software, pois fala em inovação e acesso a tecnologias da informação. Objetivo 9: “Construir infraestruturas resilientes, promover a industrialização inclusiva e sustentável e fomentar a inovação” (ONU, s.d.) e “9.c Aumentar significativamente o acesso às tecnologias de informação e comunicação e se empenhar para oferecer acesso universal e a preços acessíveis à internet nos

países menos desenvolvidos, até 2020”, não teríamos como sugerir preços de internet, por exemplo. Mas podemos propor estratégias de mitigação para uso de sistemas de software com conexões à internet limitadas ou inexistentes.

Não optamos especificamente por utilizar o Objetivo 5: “Alcançar a igualdade de gênero e empoderar todas as mulheres e meninas” (ONU, s.d.) pois ele traria uma especificidade muito grande de critérios e por entendermos que, num contexto de software, contribuir para o Objetivo 10 contemplaria as contribuições ao Objetivo 5. O Objetivo 10 e suas metas escolhidas foram:

Objetivo 10. Reduzir a desigualdade dentro dos países e entre eles

10.2 Até 2030, empoderar e promover a inclusão social, econômica e política de todos, independentemente da idade, gênero, deficiência, raça, etnia, origem, religião, condição econômica ou outra

10.3 Garantir a igualdade de oportunidades e reduzir as desigualdades de resultados, inclusive por meio da eliminação de leis, políticas e práticas discriminatórias e da promoção de legislação, políticas e ações adequadas a este respeito (ONU, s.d.).

Contribuir para o “Objetivo 12. Assegurar padrões de produção e de consumo sustentáveis” (ONU, s.d.) relaciona-se com evitar a produção de software desenfreada e/ou desorganizada resultando em produtos muito semelhantes à toa, sistemas de software sem manutenibilidade e reusabilidade. Na Meta: “12.8 Até 2030, garantir que as pessoas, em todos os lugares, tenham informação relevante e conscientização para o desenvolvimento sustentável e estilos de vida em harmonia com a natureza” (ONU, s.d.) inclui-se a questão da disseminação das medidas de contribuição aos ODSs que estão sendo adotadas durante o processo de desenvolvimento de software.

Por último, incluímos também o ODS 17: “Fortalecer os meios de implementação e revitalizar a parceria global para o desenvolvimento sustentável”, focando na Meta 17.18, quando menciona a disponibilidade de dados e sua qualidade:

17.18 Até 2020, reforçar o apoio à capacitação para os países em desenvolvimento, inclusive para os países menos desenvolvidos e pequenos Estados insulares em desenvolvimento, para aumentar significativamente a disponibilidade de dados de alta qualidade, atuais e confiáveis, desagregados por renda, gênero, idade, raça, etnia, status migratório, deficiência, localização geográfica e outras características relevantes em contextos nacionais (ONU, s.d.).



Figura 1 – Ilustração dos ODSs selecionados, elaborado pela autora (2023).

3.5 SAÚDE DIGITAL E SUSTENTABILIDADE

É necessário compreender como as mudanças de diferentes magnitudes interagem na política de desenvolvimento sustentável (SALO *et al.*, 2022). Como já exposto, atingir em sua totalidade os Objetivos de Desenvolvimento Sustentável pode ser algo inalcançável, portanto, como Salo *et al.* (2022) ressaltam em seu estudo, é importante visualizar as pequenas mudanças e êxitos como algo concreto que pode crescer até culminar em medidas de grande escala. Assim como Salo *et al.* (2022), nos propomos a focar nessas mudanças menores, no reino dos "4Is", que representam instituições, interesses, informações e ideias (SALO *et al.*, 2022).

Ningrum *et al.* (2021) trazem que, para atingir os ODSs, devem ser propostas ações locais, de baixo para cima, pois o resultado dessas ações atingiria os sistemas ambientais e sociais em sua abrangência. Para tal, é necessário adotar uma abordagem de governança participativa e inclusiva, que envolva governos locais e nacionais, sociedade civil, empresas, organizações não governamentais, pesquisadores e cientistas (NINGRUM *et al.*, 2021).

No mundo atual, onde a digitalização cresce e ganha forças a cada dia, a personalização e a interconectividade são fatores importantes a serem levados em consideração nesse processo (KUNKEL *et al.*, 2021).

O governo e o setor público, no entanto, precisarão desempenhar um papel fundamental na promoção do acesso a soluções de código aberto e dados abertos – ou seja, moldar o contexto do sistema e/ou economia política da digitalização - bem como promover iniciativas locais que priorizem os aspectos de sustentabilidade social e ambiental do desenvolvimento e aplicação de software/hardware (KUNKEL *et al.*, 2021).

Propor que sistemas de software sejam construídos, desde sua concepção, visando a sustentabilidade contribui em um nível micro para atingir objetivos macro. Com base na definição de Pressman (2011), de que: um framework estabelece a base para um processo abrangente de engenharia de software, por meio da identificação de um conjunto reduzido de atividades estruturais que podem ser aplicáveis a diversos projetos de software, independentemente de seu tamanho ou complexidade. Ou seja, reforça-se aqui a importância de envolver diversos atores, como pesquisadores e a sociedade num todo, em projetos que culminem em atingir os Objetivos do Desenvolvimento Sustentável, a proposição de um guia específico, voltado à saúde digital, auxiliará na construção dos passos para que possamos nos aproximar de êxito na prática dos ODSs.

4 MÉTODO

Este projeto de pesquisa tem cunho exploratório e experimental quanto aos objetivos e experimental qualitativa com relação à abordagem.

A tabela a seguir apresenta a relação entre os objetivos específicos e as etapas do método.

Tabela 1 - Relação entre objetivos e método

Objetivos específicos	Etapa do método
1. Entender como os ODSs estão relacionados ao desenvolvimento de software.	Revisão sistemática da literatura.
2. Caracterizar fatores de desenvolvimento de software relacionados à sustentabilidade, considerando tanto o software sustentável quanto o software para sustentabilidade.	Definição das relações entre software e sustentabilidade.
3. Definir características do processo de desenvolvimento de software que considere os ODSs sob o ponto de vista de desenvolvedores de software.	Revisão sistemática da literatura e definição das relações entre software e sustentabilidade.
4. Definir um modelo de avaliação de software com relação aos ODSs.	Definição das relações entre software e sustentabilidade.
5. Propor uma forma de avaliação do processo, através de checklists, compostos por critérios e itens de verificação relacionados à sustentabilidade.	Definição das relações entre software e sustentabilidade.
6. Realizar a avaliação dos checklists.	Avaliação qualitativa.

As etapas do projeto de pesquisa podem ser brevemente descritas, como:

4.1 REVISÃO SISTEMÁTICA DA LITERATURA

Uma revisão sistemática de literatura, com protocolo definido a partir das diretrizes apresentadas por Kitchenham *et al.* (2015) e Dybå & Dingsøyr (2008), foi realizada objetivando identificar como as ferramentas de saúde digital podem ser

empregadas para impulsionar o alcance dos Objetivos de Desenvolvimento Sustentável, e de que forma as técnicas de engenharia de software são aplicadas na construção dessas ferramentas, com o intuito de distinguir e explorar:

- características de desenvolvimento de software que levem em consideração a sustentabilidade;
- práticas e técnicas de engenharia de software considerando o desenvolvimento sustentável para o contexto especificado.

Esta etapa teve por finalidade a identificação de elementos que foram considerados no desenvolvimento do framework proposto. O artigo produzido consta na Seção 5.3.

4.2 DEFINIÇÃO DAS RELAÇÕES ENTRE SOFTWARE E SUSTENTABILIDADE

Para responder à pergunta de pesquisa “como um software pode ser sustentável e contribuir à sustentabilidade?”, foram propostas relações entre fatores de desenvolvimento de software considerando os ODSs selecionados, a norma ISO 25010:2011, baseado nos conceitos do referencial teórico, em pesquisa exploratória, no conhecimento da equipe e nos resultados obtidos a partir da revisão sistemática. Essas relações entre estes fatores compõem um framework para auxiliar no processo de desenvolvimento de software, seguindo o conceito de sustentabilidade pelo software.

Este framework poderá ser utilizado por qualquer pessoa da área de tecnologia da informação ou stakeholders em geral para proposição e construção de produtos de software que contribuam para atingir os ODSs. Baseado no modelo iterativo-incremental que prevê entregas contínuas e por etapas durante o desenvolvimento de um software, abordamos desde as etapas iniciais de construção do software, tentando responder a questões como: “o que fazer ao ter uma ideia para propor a elaboração do projeto?” incluindo fatores que devem ser considerados desde estas etapas iniciais vislumbrando os ODSs, passando pelas etapas de construção do software com itens a serem considerados no backlog, avaliação e validação das entregas, sugerindo fatores a serem considerados durante os testes do software baseado nos ODSs e chegando a etapa final de entrega. Também propomos fatores a serem avaliados em

sistemas de software já construídos para verificar se estão consonantes com a sustentabilidade.

A utilização do modelo iterativo-incremental foi escolhida para embasamento do framework pois a entrega dos incrementos do sistema como funcionalidades para os usuários “ajuda a compreender suas necessidades para incrementos posteriores” (SOMMERVILLE, 2011, p.32), o que pode ser muito valioso vislumbrando os ODSs já que nas primeiras funcionalidades entregues já seria possível identificar dificuldades e corrigi-las evitando, assim, a *pilotite*, que segundo Lampariello *et al.* (2021) significa um software caindo em desuso ainda na fase de implantação do projeto piloto.

Cabe ressaltar que não foi desenvolvido um produto de software em si ou um protótipo, mas sim artefatos para serem utilizados durante o desenvolvimento de projetos de software, com o objetivo de cumprimento dos ODSs. Baseado em modelos já existentes na área de engenharia de software, e em modelos que consideraram os ODSs de alguma maneira, foram propostas customizações para que o objetivo já citado seja alcançado.

O framework completo foi disponibilizado através do endereço <https://www.meussoftwaresustentavel.com/> para livre acesso. Foi realizada uma construção teórica relacionando o processo de desenvolvimento de software com os ODSs selecionados, para cada um dos ODSs, conforme os Apêndices de A até G. A partir desta, foram construídos dois checklists, um para ser utilizado durante o processo de desenvolvimento de software (Apêndice H) e outro para ser utilizado na verificação de softwares já construídos, a fim de avaliar se este é consonante à sustentabilidade (Apêndice I).

4.3 AVALIAÇÃO QUALITATIVA

Foi conduzida uma avaliação qualitativa dos artefatos propostos no framework para avaliação de qualidade e, conforme necessário, refinamento da proposição. Já houve pesquisas mostrando que até 50% de pesquisas de engenharia de software não realizavam avaliações de suas teorias (BARCELOS, 2006 *apud* TRAVASSOS,

1999)⁵. Estudos experimentais possuem diversas vantagens nesses casos e são considerados uma boa abordagem (BARCELOS, 2006 *apud* TRAVASSOS, 1999)⁵.

A avaliação foi realizada através de um grupo focal, composto por 8 participantes, com estudantes de áreas diversas entre saúde e tecnologia da informação e/ou outras, utilizando um modelo de avaliação baseado no *checklist* definido por Patton (2002), que ressalta questões como a necessidade de atenção às questões e a coleta de dados qualitativos que facilite a análise.

Primeiramente, foram selecionados dois sistemas de software de interesse da equipe, um denominado Voe Bem, proposto durante a dissertação de um colega deste programa de pós-graduação e outro denominado Balance, desenvolvido por alunos do curso de Informática Biomédica da UFCSPA, durante a disciplina de Engenharia de Software II. O primeiro, trata-se de um web site para que usuários respondam um questionário antes de viajar de avião, para verificar se estão aptos para o embarque, conforme suas condições de saúde, disponível em: <http://voebem.app.br/>. O segundo, é um aplicativo mobile que realiza o exame de posturografia utilizando o acelerômetro do smartphone, para auxiliar fisioterapeutas, disponível para download em: <https://play.google.com/store/apps/details?id=br.edu.ufcspa.balance>.

A autora deste trabalho realizou a avaliação dos dois sistemas de software utilizando o checklist para avaliar a sustentabilidade pelo software em softwares já desenvolvidos, onde o avaliador não participa do processo de desenvolvimento, e registrou suas respostas antes de conduzir a avaliação com o grupo focal.

O grupo focal era composto por:

1. Mestranda do Programa de Pós-graduação em Tecnologias da Informação e Gestão em Saúde da UFCSPA, bacharel em Comunicação Social - Jornalismo, pela UFRGS.

⁵ BARCELOS, R. F. **Uma abordagem para inspeção de documentos arquiteturais baseada em checklist**. Universidade Federal do Rio de Janeiro, p. 90. 2006. Acesso em: 01/05/2023. Disponível em: <<https://www.cos.ufrj.br/index.php/pt-BR/publicacoes-pesquisa/details/15/1922>> *apud* TRAVASSOS, G. H., SCHULL, F., FREDERICKS, M., BASILI, V. R. Detecting Defects In Object-Oriented Designs: Using Reading Techniques To Increase Software Quality. **Proceedings of The Conference Object-Oriented Programming Systems, Language & Applications**. 1999

2. Mestranda do Programa de Pós-graduação em Tecnologias da Informação e Gestão em Saúde da UFCSPA, bacharel em Informática Biomédica, pela UFCSPA.

3. Mestranda do Programa de Pós-graduação em Tecnologias da Informação e Gestão em Saúde da UFCSPA, bacharel em Informática Biomédica, pela UFCSPA.

4. Mestrando do Programa de Pós-graduação em Tecnologias da Informação e Gestão em Saúde da UFCSPA, bacharel em Informática Biomédica, pela UFCSPA.

5. Mestrando do Programa de Pós-graduação em Tecnologias da Informação e Gestão em Saúde da UFCSPA, bacharel em Informática Biomédica, pela UFCSPA.

6. Mestranda do Programa de Pós-graduação em Tecnologias da Informação e Gestão em Saúde da UFCSPA, bacharel em Fisioterapia, pela UFCSPA.

7. Graduando em Medicina, pela PUCRS.

8. Graduanda em Medicina, pela PUCRS.

Foi feito contato prévio com os 8 membros do grupo, em 15/05/2023, para verificar o interesse na participação no grupo focal. As 8 pessoas aceitaram e no dia 24/05/2023 foram realizadas duas sessões online, a primeira das 18:30 às 19:00 e a segunda das 19:30 às 20:00, devido a disponibilidade manifestada pelos participantes.

As sessões foram conduzidas conforme o roteiro:

- a) Apresentação pessoal da autora proponente deste trabalho.
- b) Apresentação do tema proposto nesta dissertação.
- c) Apresentação do site contendo o framework <https://www.meusoftwaresustentavel.com/>.
- d) Breve apresentação do software Voe Bem.
- e) Breve apresentação do software Balance.
- f) Orientações gerais: os participantes foram orientados a lerem o conteúdo do site <https://www.meusoftwaresustentavel.com/> antes de preencherem o checklist. Foi informado que seria enviado por e-mail os links dos dois softwares

para avaliação e do site contendo o framework proposto, além de um arquivo com a planilha contendo o checklist para que respondessem individualmente e retornassem também por e-mail. Os participantes foram orientados a responder os checklists de modo avaliativo quanto à sustentabilidade e não como um teste exploratório de software. Também no e-mail foram enviadas 4 perguntas a cada um:

- I. O que você achou do checklist utilizado como instrumento para verificação dos sistemas de software? Os itens estavam claros, de fácil compreensão, o número de itens é bom, quaisquer comentários ou sugestões são bem-vindos.
- II. Você acha que seria viável utilizar o checklist do processo de desenvolvimento de software em algum projeto futuro seu?
- III. Você acha que seria viável utilizar o checklist para sistemas de software já prontos em alguma outra avaliação de software de seu interesse?
- IV. Você indicaria os checklists para alguém?

Dois dos participantes não conseguiram comparecer às sessões online e receberam todas as instruções do protocolo citado por e-mail.

5 RESULTADOS

Nesta seção abordaremos os resultados obtidos através da avaliação qualitativa (Subseção 5.1) e os dois artigos científicos que foram desenvolvidos ao longo deste trabalho, um artigo com o conteúdo desta dissertação (Subseção 5.2) e um artigo de revisão sistemática (Subseção 5.3), para futura submissão à e formatados de acordo com o exigido.

5.1 RESULTADOS DA AVALIAÇÃO QUALITATIVA

O preenchimento de cada um dos itens do checklist foi somado para comparar com as respostas da autora, pois caso a maioria das respostas a cada item fosse semelhante a resposta da autora, aquele item teria atingido seu objetivo.

As Tabelas 2 e 3 ilustram a comparação das respostas da autora com as quantidades das respostas do grupo focal, às avaliações do software Voe Bem e Balance, respectivamente. Um dos participantes não pôde avaliar o software Balance por não possuir acesso a um smartphone com sistema operacional Android, sendo que a aplicação só funciona no referido sistema operacional.

As respostas do grupo focal fizeram a própria autora mudar a sua resposta ao item número 4 na avaliação do software Voe Bem e ao item número 3 na avaliação do software Balance, evidenciando que se, na prática, a avaliação de um software utilizando o checklist proposto for realizada por mais de uma pessoa, o resultado poderá ser mais enriquecedor por promover uma discussão acerca dos itens.

Tabela 2 – Comparativo entre as respostas da autora e quantidade de respostas dos participantes ao software Voe Bem

Softwares já construídos								
Voe Bem								
Fatores a serem verificados num software já existente, onde o processo de desenvolvimento não foi acompanhado pelo avaliador.			Respostas da autora			Quantidade de respostas dos participantes		
Item nº	Contribuições à sustentabilidade	Item de verificação	Sim	Não	Não se aplica	Sim	Não	Não se aplica

1	Qualquer ODS.	O propósito do software contribui à algum dos 17 ODS de maneira geral?	x			7	1	
2	Contribui ao ODS 12 e à Meta 7.3.	Você conhece algum software semelhante ao que está sendo avaliado?		x			8	
3	Contribui aos ODS 10 e 4 e às Metas 10.2 e 10.3.	Aparentemente, o software possui fatores inclusivos em geral?	x			5	2	
4	Contribui ao ODS 10 e suas Metas 10.2 e 10.3 Abrange o princípio da usabilidade, especialmente o item de acessibilidade da ISO 25010:2011.	Os elementos em tela, para os usuários, são inclusivos?	x			3	5	
5	Contribui à Meta 7.3. Abrange o princípio da usabilidade e da adequação funcional da ISO 25010:2011.	Os fluxos de telas dos usuários são otimizados? A utilização é intuitiva?	x			8		
6	Contribui ao ODS 4 e 10 e às Metas 4.a e 9.c.	Existem fatores que podem fazer com que usuários sintam-se excluídos do uso do software?		x		4	4	

7	Contribui à Meta 9.c. Abrange o princípio da portabilidade especialmente o item adaptabilidade da ISO 25010:2011.	Foram consideradas as possíveis limitações de acesso à internet e a hardware?		x			5	3
8	Contribui ao ODS 9 e às Metas 9.c, 10.2 e 10.3. Abrange o princípio da portabilidade, nos itens de adaptabilidade e instalabilidade da ISO 25010:2011.	O software pode ser acessado de diferentes dispositivos em diversos ambientes?	x			8		
9	Contribui com o ODS 12 e com as Metas 8.4 e 7.3. Abrange o princípio de manutenibilidade no critério reusabilidade da ISO 25010:2011.	Foi reutilizado código para as funcionalidades? Foram consideradas boas práticas e padrões de programação?			x	1		7
10	Contribui com a Meta 7.3, 8.4 e com o ODS 12.	Foi possível medir o gasto energético dos dispositivos de hardware utilizados durante o desenvolvimento do projeto?			x			8
11	Contribui para a Meta 17.18. Abrange o princípio da segurança da ISO 25010:2011.	Está claro aos usuários as coletas de dados que estão sendo realizadas?	x			7	1	

12	Contribui à Meta 17.18. Abrange o princípio da segurança da ISO 25010:2011.	O sistema passa segurança aos usuários quanto a possíveis ataques?	x			2	2	4
13	Contribui aos ODS 17, 12 e à Meta 8.4.	As funcionalidades estão implementadas corretamente?	x			8		
14	Contribui aos ODS 17, 12 e à Meta 8.4.	Há funcionalidades que geram valor e/ou são diferenciais do sistema para o usuário?	x			8		
15	Contribui à Meta 12.8.	Foram utilizados recursos para contribuição aos ODS no projeto?	x			6	1	1
16	Contribui ao ODS 12.	A recepção do produto de software, de maneira geral, é positiva?	x			7		1
17	Contribui à Meta 12.8.	Por fim, é possível enxergar os esforços que foram tomados consonantes aos ODS?			x	5	2	

Tabela 3 – Comparativo entre as respostas da autora e quantidade de respostas dos participantes ao software Balance

Softwares já construídos								
Balance								
Fatores a serem verificados num software já existente, onde o processo de desenvolvimento não foi acompanhado pelo avaliador.			Respostas da autora			Quantidade de respostas dos participantes		
Item nº	Contribuições à sustentabilidade	Item de verificação	Sim	Não	Não se aplica	Sim	Não	Não se aplica
1	Qualquer ODS.	O propósito do software contribui à algum dos 17 ODS de maneira geral?	x			6	1	

2	Contribui ao ODS 12 e à Meta 7.3.	Você conhece algum software semelhante ao que está sendo avaliado?		x		7		
3	Contribui aos ODS 10 e 4 e às Metas 10.2 e 10.3.	Aparentemente, o software possui fatores inclusivos em geral?	x			2	4	1
4	Contribui ao ODS 10 e suas Metas 10.2 e 10.3 Abrange o princípio da usabilidade, especialmente o item de acessibilidade da ISO 25010:2011.	Os elementos em tela, para os usuários, são inclusivos?		x		4	3	
5	Contribui à Meta 7.3. Abrange o princípio da usabilidade e da adequação funcional da ISO 25010:2011.	Os fluxos de telas dos usuários são otimizados? A utilização é intuitiva?		x		7		
6	Contribui ao ODS 4 e 10 e às Metas 4.a e 9.c.	Existem fatores que podem fazer com que usuários se sintam excluídos do uso do software?		x		3	4	
7	Contribui à Meta 9.c. Abrange o princípio da portabilidade especialmente o item adaptabilidade da ISO 25010:2011.	Foram consideradas as possíveis limitações de acesso à internet e a hardware?		x		2	3	2

8	Contribui ao ODS 9 e às Metas 9.c, 10.2 e 10.3. Abrange o princípio da portabilidade, nos itens de adaptabilidade e instalabilidade da ISO 25010:2011.	O software pode ser acessado de diferentes dispositivos em diversos ambientes?		x		6	1	
9	Contribui com o ODS 12 e com as Metas 8.4 e 7.3. Abrange o princípio de manutenibilidade no critério reusabilidade da ISO 25010:2011.	Foi reutilizado código para as funcionalidades? Foram consideradas boas práticas e padrões de programação?			x	1		6
10	Contribui com a Meta 7.3, 8.4 e com o ODS 12.	Foi possível medir o gasto energético dos dispositivos de hardware utilizados durante o desenvolvimento do projeto?			x			7
11	Contribui para a Meta 17.18. Abrange o princípio da segurança da ISO 25010:2011.	Está claro aos usuários as coletas de dados que estão sendo realizadas?		x		4	3	
12	Contribui à Meta 17.18. Abrange o princípio da segurança da ISO 25010:2011.	O sistema passa segurança aos usuários quanto a possíveis ataques?		x			3	4
13	Contribui aos ODSs 17, 12 e à Meta 8.4.	As funcionalidades estão implementadas corretamente?	x			7		
14	Contribui ao ODS 17, 12 e à Meta 8.4.	Há funcionalidades que geram valor e/ou são diferenciais do sistema para o usuário?	x			7		

15	Contribui à Meta 12.8.	Foram utilizados recursos para contribuição aos ODS no projeto?		x		5	2	1
16	Contribui ao ODS 12.	A recepção do produto de software, de maneira geral, é positiva?	x			6		1
17	Contribui à Meta 12.8.	Por fim, é possível enxergar os esforços que foram tomados consonantes aos ODS?		x		5	2	

Com base nos resultados obtidos, os itens em que a maioria das respostas estava discordante da resposta da autora foram refinados. Também foi refinado o primeiro item e removido o décimo item de acordo com sugestões recebidas nas perguntas descritivas enviadas aos participantes. Os itens de número 15 e 17 foram mantidos mesmo com a diferença entre o número de respostas dos participantes e a resposta da autora, por concluir-se que houve uma confusão por parte de alguns dos participantes que tenderam a pensar que os sistemas de software avaliados, contribuiriam robustamente aos ODSs, já que este é o tema da presente dissertação. Os itens 2 e 6 foram reescritos, o primeiro para melhor compreensão e o segundo para que sua resposta ideal fosse “sim”, a fim de quantificar uniformemente as respostas que levariam um software a satisfazer o conceito de sustentabilidade pelo software. A reescrita do item 6, acabou levando ao refinamento do item 5 do checklist de processo de desenvolvimento de software e motivando mais uma revisão geral dos itens. O resultado dos checklists, após os refinamentos, foi atualizado no framework disponível em <https://www.meusoftwaresustentavel.com/> e constam nos Apêndices H e I.

Foram quantificadas em 4 faixas de resultados, baseadas na quantidade de respostas “sim” aos checklists, o quanto o software está satisfazendo o conceito de sustentabilidade pelo software. Os itens dos checklists são de pesos iguais, viabilizando uma divisão quantitativa simples para as faixas. As quantidades e o grau de satisfação ao conceito proposto são apresentados na Tabela 4, para o uso do checklist durante o processo de desenvolvimento de software, sendo que nesse caso, cabe ressaltar que se espera que o maior número possível de itens seja incorporado ao longo do processo. Na Tabela 5, são apresentadas as faixas de resultados para o

checklist de avaliação de softwares já construídos, onde o avaliador não participou do processo de desenvolvimento.

Tabela 4 – Classificação do processo de desenvolvimento de software conforme o checklist.

Classificação de acordo com o checklist para utilização durante o processo de desenvolvimento de software	
Quantidade de respostas sim	Classificação
De 0 a 6	Insatisfatório
De 7 a 13	Pouco contribui à sustentabilidade pelo software
De 14 a 19	Contribui à sustentabilidade pelo software
De 20 a 25	Contribui satisfatoriamente à sustentabilidade pelo software

Tabela 5 – Classificação de um software já construído avaliado conforme o checklist.

Classificação de acordo com o checklist para utilização durante o processo de desenvolvimento de software	
Quantidade de respostas sim	Classificação
De 0 a 4	Insatisfatório
De 5 a 8	Pouco contribui à sustentabilidade pelo software
De 9 a 12	Contribui à sustentabilidade pelo software
De 13 a 16	Contribui satisfatoriamente à sustentabilidade pelo software

5.2 ARTIGO

Suprimido do texto para submissão à periódicos.

5.3 ARTIGO DE REVISÃO SISTEMÁTICA

Suprimido do texto para submissão à periódicos.

6 CONSIDERAÇÕES FINAIS

O enfoque deste trabalho surge da necessidade de responder à pergunta: “como um software pode ser sustentável e contribuir à sustentabilidade?”, motivado por entendermos e acreditarmos que um software pode ser sustentável por si só ao mesmo tempo em que contribui com a sustentabilidade.

Os Objetivos de Desenvolvimento Sustentável – Agenda 2030 são ambiciosos e, provavelmente, difíceis de alcançar por muitos países que participaram da referida agenda. Assim, quaisquer contribuições para os ODSs, em cenários de quaisquer tamanhos, são válidas e devem ser executadas, pois a cada microcenário que se busca atingir, se contribui para um macro cenário.

Assim, o objetivo deste estudo foi elaborar um framework que ofereça suporte ao desenvolvimento de sistemas de software autossustentáveis, bem como à promoção da sustentabilidade por meio do software.

Foi realizado um estudo de revisão sistemática com o objetivo de identificar como soluções em saúde digital podem ser empregadas para impulsionar o alcance dos ODSs, e a forma com a qual técnicas de engenharia de software são utilizadas na construção dessas soluções. A partir desse estudo foram levantados diversos pontos importantes para se atentar na proposição do framework.

O framework proposto relacionou etapas do processo de desenvolvimento de software, com os ODSs selecionados e com princípios da ISO 25010:2011 visando garantir que um software construído utilizando o framework contribuísse ao alcance do conceito de sustentabilidade pelo software. O fato de as proposições terem sido avaliadas por um grupo focal diverso, tanto por pessoas que trabalham com software quanto não, mas que podem ser stakeholders em projetos, e terem recebido avaliações positivas, corrobora com a conclusão de que o presente estudo tem uma boa aplicabilidade.

Para o futuro, espera-se que o tema proposto se torne presente no ambiente acadêmico, especialmente na UFCSPA, espera-se que os artefatos criados possam ser utilizados nas pesquisas tanto da pós-graduação quanto da graduação. Além da utilização do que já foi produzido, também pode ser dada continuidade a esta pesquisa, expandindo-a, refinando-a ou adaptando conforme necessidades

específicas do cenário de uso. Os artefatos produzidos poderão ser utilizados no processo de desenvolvimento em geral, como fatores a serem considerados para validação de uma das entregas incrementais, pontos a serem levantados na concepção do projeto, dentre outros. Alguns trabalhos em desenvolvimento neste programa de pós-graduação em tecnologias da informação e gestão em saúde, já estão considerando o tema proposto, há em curso uma pesquisa que visa aumentar o engajamento de stakeholders envolvidos no desenvolvimento de projetos junto aos alunos do curso de informática biomédica, na disciplina de engenharia de software, que já utilizou o tema na fase de qualificação do projeto. Ainda na melhoria dos projetos da disciplina mencionada, está sendo desenvolvido um estudo de como aplicar micro serviços no desenvolvimento desses projetos, que pode englobar vários dos fatores apresentados no framework proposto neste trabalho. Também em curso, há uma dissertação sobre ferramentas de saúde digital para a população negra, que se relaciona com as proposições aqui feitas, especialmente no que cerne o ODS 10. Especialmente no que tange as proposições deste framework para a Meta 17.18, está sendo realizada uma pesquisa para propor um processo de desenvolvimento de software biomédico contendo técnicas relacionadas ao ciclo de vida e qualidade dos dados, visando garantir a viabilidade das coletas, processamento, usabilidade e acesso aos dados gerados.

A relevância dessa pesquisa, sob o conceito de sustentabilidade pelo software, vai desde a pouca quantidade de referências bibliográficas sobre o tema atualmente, até suas contribuições à sustentabilidade, conforme princípios de igualdade, equidade, acessibilidade, economia consciente e fatores ambientais, através da inovação e eficiência em produtos de software.

REFERÊNCIAS

OLU, Olushayo; MUNEEENE, Derrick; BATARINGAYA, Juliet E.; NAHIMANA, Marie-Rosette; BA, Housseynou; TURGEON, Yves; KARAMAGI, Humphrey C.; DOVLO, Delanyo. How Can Digital Health Technologies Contribute to Sustainable Attainment of Universal Health Coverage in Africa? A Perspective. **Frontiers in Public Health**, vol. 7. 2019. DOI: 10.3389/fpubh.2019.00341

MITCHELL, Marc; KAN, Lena. Digital Technology and the Future of Health Systems. **Health Systems & Reform**, vol 5. 2019. DOI: <https://doi.org/10.1080/23288604.2019.1583040>

SOMMERVILLE, Ian. **Engenharia de software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

PRESSMAN, Roger S. **Engenharia de software: Uma abordagem profissional**. 7. ed. Porto Alegre: Bookman, 2011.

LAWLOR, Brendan.; WALSH, Paul. Engineering bioinformatics: building reliability, performance and productivity into bioinformatics software. **Bioengineered**, v. 6, n. 4, p. 193–203. 2015.

ZHU, Haibin.; MENGCHU, Zou.; ZEGUIN, Pierre. Supporting Software Development With Roles. **IEEE transactions on systems, man, and cybernetics—part a: systems and humans**, v. 36, n. 6. 2006

Ozkaya, Ipek. A Watershed Moment for Search-Based Software Engineering. **IEEE Software**, vol. 38, n. 4, p. 3-6. 2021. DOI: 10.1109/MS.2021.3075108.

ISO/IEC/IEEE 24765. Systems and software engineering – Vocabulary: International Standard. 2ª edição: Geneva, Suíça. Nova Iorque, EUA. 2017.

FAUSTINO, Manuel; AMADOR, Filomena. O conceito de “sustentabilidade”: migração e mudanças de significados no âmbito educativo. **Indagatio Didactica**, Portugal, v. 8 n. 1. 2016

RATTNER, Henrique. Sustentabilidade - uma visão humanista. **ANPPAS - Revista Ambiente e Sociedade Anppas**, São Paulo, SP, Brasil. 2006.

LIMA, Sérgio S. Introdução ao conceito de sustentabilidade, aplicabilidade e limites. **Cadernos da Escola de Negócios**, Brasil, v. 1, n. 4. 2006.

BONNEDAHL, Karl J; HEIKKURINEN, Pasi; PAAVOLA, Jouni. Strongly sustainable development goals: Overcoming distances constraining responsible action. **Environmental Science and Policy**, v. 129, p. 150–158. 2022. DOI: <https://doi.org/10.1016/j.envsci.2022.01.004>

LARREA-GALLEGOS, Gustavo; BENETTO, Enrico; MARVUGLIA, Antonino; GUTIERREZ, Tomás Navarrete. Sustainability, resilience and complexity in supply networks: A literature review and a proposal for an integrated agent-based approach, Sustainable Production and Consumption. 2022. DOI: <https://doi.org/10.1016/j.spc.2022.01.009>

KUNKEL, Stefanie; TYFIELD, David. Digitalisation, sustainable industrialisation and digital rebound – Asking the right questions for a strategic research agenda.

Energy Research & Social Science, v. 82. 2021. DOI: <https://doi.org/10.1016/j.erss.2021.102295>

SALO, Hanna H; BERG, Annukka; KORHONEN-KURKI, Kaisa; LÄHTEENOJA, Satu. Small wins enhancing sustainability transformations: Sustainable development policy in Finland. **Environmental Science and Policy**, v. 128, p. 242–255. 2022. DOI: <https://doi.org/10.1016/j.envsci.2021.11.024>

BIGLARI, Shirin; BEIGLARY, Shevva; ARTHANARI, Arthanari. Achieving sustainable development goals: Fact or Fiction?. **Journal of Cleaner Production**, v. 332. 2021. DOI: <https://doi.org/10.1016/j.jclepro.2021.130032>

NINGRUM, Dianty; MALEKPOUR, Shirin; RAVEN, Rob; MOALLEMI, Enayat A. Lessons learnt from previous local sustainability efforts to inform local action for the Sustainable Development Goals. **Environmental Science and Policy**, v. 129, p. 45–55. 2021. DOI: <https://doi.org/10.1016/j.envsci.2021.12.018>

MOHR, Alexander; SCHUMACHER, Christian; KIEFNER, Valentin. Female executives and multinationals' support of the UN's sustainable development goals. **Journal of World Business**, v. 57. 2021. DOI: <https://doi.org/10.1016/j.jwb.2021.101304>

Kitchenham, Barbara Ann; Budgen, David; Brereton, Pearl. Evidence-Based Software Engineering and Systematic Reviews. Flórida: CRC Press, 2015.

Dybå, Tore & Dingsøy, Torgeir. Strength of Evidence in Systematic Reviews in Software Engineering. **ESEM'08: Proceedings of the 2008 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement**. 178-187, 2008.

ORGANIZAÇÃO DAS NAÇÕES UNIDAS (ONU). Objetivos de Desenvolvimento Sustentável. Disponível em: <<https://brasil.un.org/pt-br/sdgs/10>>. Acesso em: 25/01/22.

PATTON, Michael Q. **Qualitative Research & Evaluation Methods - Integrating Theory and Practice**. 3. ed. London: Sage Publications, 2002

Lampariello, R., Ancellin-Panzani, S. Mastering stakeholders' engagement to reach national scale, sustainability and wide adoption of digital health initiatives: lessons learnt from Burkina Faso. **Farm Med Community Health**. 2021. DOI: [10.1136/fmch-2021-000959](https://doi.org/10.1136/fmch-2021-000959)

APÊNDICE A – ODS 4, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

ODS 4 e metas em destaque:

ODS 4: “Assegurar a educação inclusiva e equitativa e de qualidade, e promover oportunidades de aprendizagem ao longo da vida para todas e todos” (ONU, s.d.)

4.a: “Construir e melhorar instalações físicas para educação, apropriadas para crianças e sensíveis às deficiências e ao gênero, e que proporcionem ambientes de aprendizagem seguros e não violentos, inclusivos e eficazes para todos” (ONU, s.d.)

4.7: “Até 2030, garantir que todos os alunos adquiram conhecimentos e habilidades necessárias para promover o desenvolvimento sustentável, inclusive, entre outros, por meio da educação para o desenvolvimento sustentável e estilos de vida sustentáveis, direitos humanos, igualdade de gênero, promoção de uma cultura de paz e não violência, cidadania global e valorização da diversidade cultural e da contribuição da cultura para o desenvolvimento sustentável” (ONU, s.d.)

O ODS 4 trata, em sua totalidade, de educação. Entendemos que não há como separar uma educação de qualidade de sistemas de software produzidos com qualidade, afinal novas tecnologias surgem repentinamente e é necessário estudá-las, mesmo que fora do ambiente acadêmico, para colocá-las em prática.

Durante a pandemia da Covid-19 diversas regiões do mundo adotaram como única alternativa o uso de tecnologias da informação para manter aulas acontecendo e mitigar os prejuízos que alunos de todos os níveis teriam ao não frequentar as instituições de ensino. Porém isso evidenciou exponencialmente as desigualdades de acesso entre os alunos. Rangel-Pérez *et al.* (2021) afirmam em seu estudo que “a exclusão digital gerada pela pandemia compromete seriamente as metas estabelecidas no ODS 4”, diz ainda que esta exclusão ocorre mesmo em países desenvolvidos, com mais tecnologia e professores com altos graus de formação.

Considerando que todos os ODSs estão interligados de alguma forma e, ainda, que o texto do ODS 4 cita “oportunidades de aprendizagem ao longo da vida”, incluímos o referido ODS neste trabalho e falaremos como podemos considerá-lo num processo de desenvolvimento de software.

Considerar fatores inclusivos para o software: como cita Lampariello *et al.* (2021) os fatores humanos não podem ser negligenciados em um processo de software. Usabilidade também é um dos princípios da ISO 25010:2011. Porém deve-se ter um olhar diversificado ao considerar estes fatores, como o uso do software por diferentes usuários. A diversificação dos stakeholders pode auxiliar nesta missão, além de pesquisas diversas. No estudo de revisão sistemática realizado para esta dissertação, fatores como nível de escolaridade dos usuários-alvo, religião, questões culturais e individuais das pessoas envolvidas no uso dos sistemas de software. Assim se contribui aos pontos de inclusão e equidade e oportunidades para todos, conforme o ODS 4.

Alertar para fatores excludentes ao software: sistemas que necessitam de hardwares específicos para uso ou conexões específicas a internet, podem não abranger uma parcela de usuários, principalmente quando são para uso pessoal. Por exemplo, aplicativos para dispositivos móveis que só funcionam nas 3 últimas versões mais atuais do sistema operacional, excluirão todas as pessoas que possuem um dispositivo funcional com sistema operacional anterior às últimas três versões. Mc Kena *et al.* (2019) citam um software que necessita de um smartphone ou tablet com conectividade à internet para o treinamento de profissionais da saúde em Ebola, utilizado em distritos afastados de Serra Leoa, o que acarretava em barreiras para sua utilização. Além da contribuição ao ODS 4, podemos acrescentar a Meta 4.a se traçarmos um paralelo imaginário entre instalações físicas e hardware conseguimos ver a importância de o uso de um equipamento não ser excludente.

Disseminação de conteúdos sobre desenvolvimento sustentável: tanto a equipe de desenvolvimento, quanto os stakeholders e outras pessoas em qualquer nível de envolvimento com o projeto, independente das suas funções, devem ficar cientes da razão pelas quais as medidas estão sendo tomadas e como elas contribuem para a sustentabilidade, encontrando assim a Meta 4.7, apesar de neste contexto estas pessoas não serem alunos, têm-se a disseminação de informações sobre o desenvolvimento sustentável.

REFÊRENCIAS

ONU. “ODS 4 - Educação de qualidade”. Disponível em: <<https://brasil.un.org/pt-br/sdgs/4>>. Acesso em: 08 de mai. de 2023.

Rangel-Pérez, C.; Gato-Bermúdez, M.-J.; Musicco-Nombela, D.; Ruiz-Alberdi, C. (2021). "The Massive Implementation of ICT in Universities and Its Implications for Ensuring SDG 4: Challenges and Difficulties for Professors." <https://doi.org/10.3390/su132212871>

Mc Kenna, P., Babughirana, G., Amponsah, M., Egoeh, S. G., Banura, E., Kanwagi, R. and Gray, B. (2019). "Mobile training and support (MOTS) service-using technology to increase Ebola preparedness of remotely-located community health workers (CHWs) in Sierra Leone". doi: 10.21037/mhealth.2019.09.03

APÊNDICE B - ODS 7, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

ODS 7 e metas em destaque:

ODS 7 “Assegurar o acesso confiável, sustentável, moderno e a preço acessível à energia para todas e todos” (ONU, s.d.) trata, em sua totalidade, sobre questões relacionadas à energia, como a garantia de energia elétrica a todos e uso de energias renováveis.

7.3: “Até 2030, dobrar a taxa global de melhoria da eficiência energética” (ONU, s.d.)

Destacamos a Meta 7.3, onde se pode traçar paralelos entre o gasto energético consumido durante o desenvolvimento de um projeto de software e, posteriormente, seu uso em produção.

Pailman *et al.* (2022) trazem um estudo sobre as questões energéticas no continente africano e a necessidade da formação de profissionais especialistas em energia, contemplando a igualdade de gênero entre esses profissionais e o quanto a formação acadêmica deve ser ampliada para tal. Considerando que os ODSs se interligam entre si, podemos dizer que quaisquer profissionais deveriam ter consciência das questões energéticas envolvidas no seu campo de atuação. Tem-se então, a importância de trazer a energia como um dos pontos a ser observado durante o processo de desenvolvimento de software. Ratificando o exposto, podemos citar Elavarasan *et al.* (2021) que afirmam “o mundo pós-Covid exige o desenvolvimento de uma sociedade resiliente que trabalhe em pé de igualdade com o meio ambiente”. Ainda podemos afirmar que “dependendo de como o software é implementado, pode haver influência na potência usada pelo hardware” (GARCIA-BERNA *et al.*, 2021 *apud* WILKE *et al.*, 2011)¹.

Outro ponto que podemos considerar para contribuir com o não-desperdício de energia num processo de desenvolvimento de software é descrito na seção sobre o ODS 12, que consiste em fazer uma avaliação de mercado antes de iniciar o processo de desenvolvimento em si. Evitando culminar no desperdício total de energia elétrica gasta num software que caia em desuso.

Definição de escopo: as funcionalidades que serão desenvolvidas dentro do escopo do projeto devem estar bem especificadas, a fim de evitar um esforço desnecessário criando funcionalidades que não serão utilizadas de maneira satisfatória, afinal desperdiçar recursos criando funcionalidades que não serão plenamente utilizados vai contra o proposto pelo ODS 7.

Automação de processos, reuso de código e escolha de ferramentas: processos automatizados, como deploy e testes, tendem a diminuir o tempo de execução das tarefas, resultando na diminuição de tempo gasto com energia do hardware, assim como: verificar se há a possibilidade de reutilizar algum código já produzido pela própria equipe de desenvolvedores ou disponibilizado de maneira aberta. Apesar de simples, esses são exemplos de fatores que diminuem o intervalo de tempo que um desenvolvedor precisaria ficar utilizando o hardware para executar as tarefas citadas, consequentemente diminuindo o consumo de tempo energético do dispositivo. Podemos exemplificar com: “uma determinada escolha de API em que diferentes tamanhos de buffer foram selecionados teve impactos significativos no custo de energia, economizando até 76% da energia inicialmente consumida” (GARCIA-BERNA *et al.*, 2021 *apud* SINGH *et al.*, 2015)².

Medição do gasto energético dos dispositivos de hardware utilizados durante o desenvolvimento do projeto: existem dispositivos de hardware que podem ser acoplados ao computador para medir o gasto energético da máquina, como o EET que possui sensores que podem medir a energia de três diferentes partes do hardware: processador, disco rígido e placa gráfica (GARCIA-BERNA *et al.*, 2021). É uma estratégia interessante quando possível de ser empregada para medir o consumo energético durante o trabalho dos desenvolvedores. Com os resultados que podem ser obtidos através da medição do gasto energético durante o processo de desenvolvimento, podem ser tomadas medidas para a redução do desperdício ou gasto exacerbado de energia, caso necessário.

Testes de caixa preta: para evitar o desperdício de tempo e recursos energéticos, caso sejam utilizados testes de caixa preta no projeto, os casos de teste devem estar previstos e documentados para serem executados de forma objetiva. Além disso, ressaltamos a importância do uso de testes automatizados para que possíveis bugs sejam identificados já durante a construção do código do software

evitando desperdício de recursos durante os testes de caixa preta. Estes fatores também vão ao encontro do cumprimento do critério de testabilidade do software, descrito no princípio de manutenibilidade da ISO 25010:2011.

Usabilidade: a usabilidade do sistema é um ponto de alto impacto no consumo de energia do hardware. Se os usuários conseguem realizar as tarefas propostas de maneira clara, ágil e intuitiva, sem precisar de muitos passos desnecessários, se os elementos em tela não demandam tempo de carregamento e não tiram o foco do usuário, o tempo de uso do software é otimizado, logo o tempo de energia gasta pelo hardware diminui. Garcia-Berna *et al.* (2021) trazem em seu estudo avaliações de gasto energético no uso de sistemas de software de registros pessoais em saúde e aponta casos em que um grande número de usuários acessa um portal web ao mesmo tempo, afirmando que “uma pequena economia na energia consumida para uma determinada funcionalidade pode implicar em uma grande economia na energia total necessária para atender todos os clientes do portal web”, ou seja, uma boa usabilidade do sistema também contribui às questões relacionadas ao funcionamento deste sistema, do ponto de vista energético. Baseado no estudo comparativo do gasto energético, o estudo traz pontos que podem ser adotados num software para melhorar seu consumo, como: tons de cores adequados ao tipo de tecnologia do monitor que será utilizado, widgets devem ter tamanho significativo para que sejam atingidos, as ações a serem executadas devem estar reunidas de forma coerente, diminuição do uso de elementos como barras de progresso e animações, funcionalidades com preenchimento automático, dentre outros (GARCIA-BERNA *et al.*, 2021). Além disto, uma usabilidade adequada encontra os princípios da usabilidade e da adequação funcional da ISO 25010:2011.

REFÊRENCIAS

¹Garcia-Berna, J. A., Fernandez-Aleman, J. L., de Gea, J. M. C., Toval, A., Mancebo, J., Calero, C., Garcia, F. Energy efficiency in software: A case study on sustainability in personal health records. *Journal of cleaner production*, v. 282. 2021. DOI: <https://doi.org/10.1016/j.jclepro.2020.124262> apud Wilke, C., Gotz, S., Cech, S., Waltsgott, J., Fritzsche, R. Aspects of software's energy consumption. *Inst. fur Softw. und Multimediatechnik*. 2011.

²Garcia-Berna, J. A., Fernandez-Aleman, J. L., de Gea, J. M. C., Toval, A., Mancebo, J., Calero, C., Garcia, F. Energy efficiency in software: A case study on sustainability in personal health records. *Journal of cleaner production*, v. 282. 2021. DOI: <https://doi.org/10.1016/j.jclepro.2020.124262> apud Singh, J., Naik, K., Mahinthan, V. Impact of Developer Choices on Energy Consumption of Software on Servers. *Procedia Computer Science*, v. 62, p. 385-394. 2015. <https://doi.org/10.1016/j.procs.2015.08.423>.

ONU. “ODS 7 - Energia limpa e acessível”. Disponível em: <<https://brasil.un.org/pt-br/sdgs/7>>. Acesso em: 08 de mai. de 2023.

Pailman, W., Groot, J. Rethinking education for SDG 7: A framework for embedding gender and critical skills in energy access masters programmes in Africa. *Energy Research & Social Science*, v. 90. 2022. DOI: <https://doi.org/10.1016/j.erss.2022.102615>.

Elavarasan, R. M., Pugazhendhi, R., Jamal, T., Dyduch, J., Arif, M. T., Kumar, N. M., Shauhrat, C. S., Nadarajah, M. (2021). Envisioning the UN Sustainable Development Goals (SDGs) through the lens of energy sustainability (SDG 7) in the post-Covid-19 world. *Applied Energy*, v. 292. 2021. DOI: <https://doi.org/10.1016/j.apenergy.2021.116665>

Garcia-Berna, J. A., Fernandez-Aleman, J. L., de Gea, J. M. C., Toval, A., Mancebo, J., Calero, C., Garcia, F. Energy efficiency in software: A case study on sustainability in personal health records. *Journal of cleaner production*, v. 282. 2021. DOI: <https://doi.org/10.1016/j.jclepro.2020.124262>

APÊNDICE C - ODS 8, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

ODS 8 e metas em destaque:

ODS 8: "Promover o crescimento econômico sustentado, inclusivo e sustentável, emprego pleno e produtivo e trabalho decente para todas e todos" (ONU, s.d)

8.2 "Atingir níveis mais elevados de produtividade das economias por meio da diversificação, modernização tecnológica e inovação, inclusive por meio de um foco em setores de alto valor agregado e dos setores intensivos em mão de obra" (ONU, s.d)

8.4 "Melhorar progressivamente, até 2030, a eficiência dos recursos globais no consumo e na produção, e empenhar-se para dissociar o crescimento econômico da degradação ambiental, de acordo com o Plano Decenal de Programas sobre Produção e Consumo Sustentáveis, com os países desenvolvidos assumindo a liderança" (ONU, s.d)

O ODS 8 aborda questões muito voltadas ao trabalho e existem estudos acerca deste ODS, na maioria voltados às questões de direitos trabalhistas e previdenciários, condições de trabalho e políticas públicas. Aqui, focaremos no cenário micro de desenvolvimento de software e como algumas medidas podem contribuir a este ODS e às Metas 8.2 e 8.4, respaldando a importância de pensarmos nesse nicho, citamos Khalique *et al.* (2021): "De todos os desenvolvimentos que impulsionam os negócios globais hoje, aquele que está transformando a gestão internacional mais do que qualquer outro é o rápido avanço no setor de TI/ITeS", ainda Ufua *et al.* (2021) dizem que a transformação digital, do ponto de vista dos atores, "inclui liderança transformadora, capacidades gerenciais e organizacionais, cultura da empresa e ambiente e ambiente de trabalho.

Disseminação de conteúdos sobre sustentabilidade à equipe de desenvolvimento: A equipe de desenvolvimento, em qualquer nível de envolvimento com o projeto, independente das suas funções, deve ficar ciente da razão pelas quais as medidas estão sendo tomadas e como elas contribuem para a sustentabilidade,

pois tendo os conhecimentos sobre sustentabilidade adquiridos através de seu trabalho, fará com que as pessoas da equipe tomem consciência da importância da incorporação dos fatores contribuintes ao alcance dos ODSs e que elas mesmas tenham iniciativa para propor diversificação, modernização e inovação culminando numa melhor produção e diminuição da degradação ambiental.

Definição da equipe de desenvolvimento: Uma equipe com papéis bem definidos clarifica as necessidades de conhecimento para que as as funções de cada um seja desempenhada satisfatoriamente, Nazir *et al.* (2020) afirmam que a o conhecimento específico é fundamental para processos de engenharia de software sustentável e, ainda, que a falta destes conhecimentos específicos pode ser um risco à sustentabilidade individual. A clara definição de papéis e tarefas também é importante para que uns não sejam sobrecarregados em detrimento de outros, afinal sobrecarregar uma pessoa ou forçar que ela desempenhe papéis numa equipe além de suas capacidades gera um impacto negativo na sua saúde mental, podendo resultar num trabalho adoecedor, pois “estados afetivos negativos entre os desenvolvedores de software podem potencialmente levar a experiências adversas, como baixa autoestima, altos níveis de estresse e ansiedade e burnout” (KURIAN *et al.*, 2022 *apud* GRAZIOTIN *et al.*, 2017)¹. Com os papéis bem definidos, também, pode-se focar nos ODSs que serão mais pertinentes a cada um, fazendo com que estas pessoas possam ter novas ideias de contribuições aos ODSs dentro daquele projeto.

Funcionalidades e objetivos do projeto: os objetivos do projeto e as funcionalidades incluídas no escopo devem estar bem claras e descritas para que as funções dos membros da equipe sejam designadas assertivamente, ainda com a mesma motivação de conhecimentos específicos de Nazir *et al.* (2020), citada no item anterior, pois a partir daí a equipe de desenvolvedores poderá se atentar a questões quanto a reutilização de código, uso de padrões de programação, automação de processos, dentre outros fatores, que contribuirão com os princípios de qualidade de software definidos pela ISO 25010:2011 e com a Meta 8.4.

Automação de processos, reuso de código e boas práticas de programação: a partir do momento em que os processos deixam de ser manuais e passam a ser automatizados as pessoas têm mais tempo para dedicar-se a outras atividades mais

produtivas, inclusive podendo dedicar-se a estudar soluções para atingir a Meta 8.2, o reuso de código também tem o mesmo impacto. Nazir *et al.* (2020) citam a importância de educação e conhecimento e a sustentação de competências e habilidades que podem melhorar a produtividade de engenheiros de software. Logo, melhorar a produtividade para que os membros da equipe possam aproveitar melhor seu tempo para buscar conhecimentos, como descrito anteriormente, enquadra-se no conceito de sustentabilidade individual descrita por Nazir *et al.* (2020). Adotar boas práticas de programação visa o futuro do software, enquadrando-se no princípio de manutenibilidade num todo, especialmente nos critérios de reusabilidade e modificabilidade da ISO 25010:2011 e contribuindo a Meta 8.4.

Interoperabilidade do software: além do descrito sobre este fator na seção sobre o ODS 12, mesmo que não seja uma demanda imediata, deve estar previsto o caso de num futuro, o sistema precisar ser interoperável com outros, sem demandar grandes esforços de desenvolvimento, visando a Meta 8.2

REFÊRENCIAS

¹Kurian, R. B., Thomas, S. Importance of positive emotions in software developers' performance: a narrative review. *Theoretical Issues in Ergonomics Science*. 2022. DOI: 10.1080/1463922X.2022.2134483 apud Graziotin, D., X. Wang, and P. Abrahamsson. Happy Software Developers Solve Problems Better: Psychological Measurements in Empirical Software Engineering. 2014. DOI:10.7717/peerj.289.

ONU. "ODS 8 - Trabalho decente e crescimento econômico". Disponível em: <<https://brasil.un.org/pt-br/sdgs/8>>. Acesso em: 08 de mai. de 2023.

Khalique, F., Madan, P., Puri, G., Parimoo, D. Incorporating SDG 8 for Decent Work Practices: A study of MNC Subsidiaries in India. *Australasian Accounting, Business and Finance Journal*, v. 15(5). 2021. DOI: <http://dx.doi.org/10.14453/aabfj.v15i5.7>

Aljarallah, S., Lock, R. Software sustainability from a user perspective a case study of a developing country (Kingdom of Saudi Arabia). 2018 International Conference on Computing, Electronics & Communications Engineering. IEEE, p. 1-6. 2018. DOI: 10.1109/iCCECOME.2018.8659275

Nazir, S., Fatima, S., Chuprat, N., Sarkan, H., Nilam, N. F., Sjarif, N. A. Sustainable Software Engineering: A Perspective of Individual Sustainability Challenges. In 11th Malaysian Software Engineering Conference, v. 10, p. 21. 2018.

APÊNDICE D - ODS 9, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

ODS 9 e metas em destaque:

ODS 9: “Construir infraestruturas resilientes, promover a industrialização inclusiva e sustentável e fomentar a inovação” (ONU, s.d.)

9.c: “Aumentar significativamente o acesso às tecnologias de informação e comunicação e se empenhar para oferecer acesso universal e a preços acessíveis à internet nos países menos desenvolvidos, até 2020” (ONU, s.d.)

O ODS 9 traz a inovação como um dos fatores para promoção da sustentabilidade e questões relacionadas à infraestrutura e industrialização, na Meta 9.c têm-se especificamente a questão das tecnologias de informação e comunicação. Nesse quesito, entendemos que é impossível que uma equipe de desenvolvimento de software consiga definir preços da internet em um país, como diz o texto da Meta 9.c. mas é possível tomar medidas para que os sistemas de software construídos sejam utilizáveis por uma ampla gama de usuários, em diversos cenários, de maneira equitativa, contribuindo ao acesso universal. Ufua *et al.* (2021) versam, em seu estudo, sobre a transformação digital na Nigéria e recomenda “a adoção de uma abordagem multidisciplinar para intervenções de transformação digital voltadas para o desenvolvimento para os ODSs 4 e 9 na Nigéria, por meio de um processo de engajamento efetivo das partes interessadas e sinalização institucional transparente”, logo pode-se inferir a importância do desenvolvimento de software numa abordagem multidisciplinar e como parte interessada no engajamento do processo. Ainda, no estudo de revisão sistemática elaborado nesta dissertação, ficaram evidenciadas questões desafiadoras relacionadas à infraestrutura tecnológica para implantação de projetos de software.

Os itens a seguir relacionam-se entre si, porém cabe abordá-los um a um para melhor exemplificar cada situação.

Conexão com a internet: Swartz *et al.* (2021) aborda que apesar do software apresentado em seu estudo ter uma aceitação dos profissionais da saúde que o utilizam e ser um modelo inovador de coleta de dados de mortalidade infantil e

neonatal, esbarra na dificuldade de ser necessária sua instalação num computador, sendo que este processo é burocrático aos usuários, porém foi a única forma encontrada para utilização diante das dificuldades de conexão à internet em unidades de saúde da África do Sul. Portanto, desenvolver funcionalidades que necessitem de conexões à internet de alta velocidade ou que necessitem de conexão com a internet o tempo todo para uso, exclui usuários que não teriam este acesso, dependendo de sua localidade geográfica ou faixa de renda. Se no exemplo citado, um software de uso corporativo (usado em unidades de saúde naquele contexto) já não conseguiria acesso à internet, sistemas de software de uso pessoal precisam de ainda mais atenção nesse quesito. Prover um software instalável que não necessita de conexão à internet em tempo real, mas que tem uma instalação pouco amigável aos usuários, também não é um cenário ideal. Portanto, quando se avalia os requisitos não funcionais de um software, deve-se considerar essas questões.

Sistemas operacionais: cabe destaque a este item, pois um projeto de software desenvolvido para funcionar em apenas um sistema operacional, poderá excluir um grande número de usuários, por exemplo, aplicativos para dispositivos móveis que só funcionam nas 3 últimas versões mais atuais do sistema operacional, excluirão todas as pessoas que possuem um dispositivo funcional com sistema operacional anterior as últimas três versões. Salvo se o software for desenvolvido para usuários-alvo muito bem definidos, com a certeza do sistema operacional que eles utilizam. James (2002) cita que o acesso universal às TICs pode ser alcançado através do acesso provido em instituições como escolas, correios e bibliotecas. Para isso precisa-se também de uma redução de custos para implantação, sendo assim, a opção por software livre torna-se uma estratégia possível, cita que: “O caso do México é ilustrativo a esse respeito porque o governo mexicano decidiu em 1998 instalar o sistema Linux em 140.000 laboratórios escolares em todo o país” (JAMES, 2002). Então ao pensar em software instalável, por exemplo para mitigar as dificuldades de acesso à internet citadas anteriormente, não se deve negligenciar os diferentes sistemas operacionais em que podem ser utilizados.

Hardware: sistemas de software ou determinadas funcionalidades de um software que exigem hardwares específicos para uso, quando são para uso pessoal, acabam excluindo as pessoas que não possuem as especificidades necessárias de hardware. Mc Kena *et al.* (2019) abordam em seu artigo um software para capacitação

de profissionais da saúde frente ao Ebola, no país de Serra Leoa, além das possíveis dificuldades de conexão à internet, é mencionada a necessidade dos usuários possuírem um smartphone ou tablet para uso do aplicativo. Conforme bem ilustrado por James (2022) “Certas instituições em países em desenvolvimento, como escolas, orfanatos e bibliotecas, muitas vezes recebem doações de computadores recondicionados e relativamente antigos de organizações não governamentais (ONGs) [...]” sendo que estes computadores são avaliados para serem doados em perfeito funcionamento, porém não fogem do fator tempo e, num mundo onde novas tecnologias surgem rapidamente, provavelmente estes computadores não sejam de muita robustez no momento em que são doados. Logo, precisam ser pensadas em estratégias para que um software possa funcionar mitigando possíveis limitações de hardware ao fazer o levantamento de requisitos não funcionais, como por exemplo, uso de memória RAM, espaço ocupado em disco, dentre outros.

Todos os fatores citados anteriormente seguem o princípio da portabilidade, principalmente o item adaptabilidade, da ISO 25010:2011.

É desafiador pensar em um software que possa contemplar os 3 critérios mencionados, sem comprometer uns aos outros, portanto é importante investigar o cenário de uso do software, para mitigar os efeitos em um critério ou em outro conforme a necessidade.

REFÊRENCIAS

ONU. “ODS 9 - Indústria, inovação e infraestrutura”. Disponível em: <<https://brasil.un.org/pt-br/sdgs/9>>. Acesso em: 08 de mai. de 2023.

Ufua, D. E., Emielu, E. T., Olujobi, O. J., Lakhani, F., Borishade, T. T., Ibidunni, A. S., Osabuohien, E. S. Digital transformation: A conceptual framing for attaining Sustainable Development Goals 4 and 9 in Nigeria. *Journal of Management & Organization*, v. 27(5), p. 836-849.2021. DOI:10.1017/jmo.2021.45

James, J. Universal access to information technology in developing countries. *Regional Studies*, v. 36, n. 9, p. 1093-1097. 2002. DOI: <https://doi.org/10.1080/0034340022000024312>

Swartz, A., LeFevre, A. E., Perera, S., Kinney, M. V., and George, A. S. Multiple pathways to scaling up and sustainability: an exploration of digital health solutions in South Africa. *Global Health*. 2021. DOI: <https://doi.org/10.1186/s12992-021-00716-1>

Mc Kenna, P., Babughirana, G., Amponsah, M., Egoeh, S. G., Banura, E., Kanwagi, R. and Gray, B. Mobile training and support (MOTS) service-using technology to increase Ebola preparedness of remotely-located community health workers (CHWs) in Sierra Leone. 2019. DOI: 10.21037/mhealth.2019.09.03

APÊNDICE E - ODS 10, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

ODS 10 e metas em destaque:

ODS 10: "Reduzir a desigualdade dentro dos países e entre eles" (ONU, s.d.)

10.2: "Até 2030, empoderar e promover a inclusão social, econômica e política de todos, independentemente da idade, gênero, deficiência, raça, etnia, origem, religião, condição econômica ou outra" (ONU, s.d.)

10.3: "Garantir a igualdade de oportunidades e reduzir as desigualdades de resultados, inclusive por meio da eliminação de leis, políticas e práticas discriminatórias e da promoção de legislação, políticas e ações adequadas a este respeito"(ONU, s.d.)

Este ODS é muito importante para a sociedade num todo e, qualquer medida que possa ser tomada em sua contribuição, deverá ser. Num processo de desenvolvimento de software, devemos ter este ODS em mente, com a proposta de contribuir num microcenário para atingir o macro. Ufua *et al.* (2021) citam “[..] a marginalização de alguns stakeholders minoritários no processo, cujo interesse pode não ser devidamente considerado no processo de transformação digital e implementação dos ODSs” como motivo para “a resistência de certas partes interessadas que são afetadas ou envolvidas no processo de transformação digital”. De fato, se as necessidades de um usuário, por exemplo, não são contempladas no uso de um software, ele irá resistir à utilização deste. Swartz *et al.* (2021) citam que as questões relacionadas às pessoas envolvidas num processo de implantação de software, características deles e suas relações são fatores mais desafiadores do que dificuldades de acesso a tecnologias, ainda Heijsters *et al.* (2023) afirma a importância de envolver a percepção dos usuários para permitir a adaptação do software a especificidades como idade, alfabetização ou deficiências.

Os fatores listados na Meta 10.2 podem ser considerados como critérios de inclusão num software e a Meta 10.3 quando fala em garantias de oportunidades podemos entender que a oportunidade de usar um software qualquer deve ser dada

a todos, conforme o objetivo do software, independente dos fatores listados na Meta 10.2.

Desde a concepção do sistema até a fase de testes deve ser considerada se a usabilidade abrange as necessidades de diferentes usuários.

Nas fases iniciais do projeto de software, desde a concepção até a prototipação, por exemplo, deve-se considerar o uso do software por diferentes usuários em diferentes contextos. Através de pesquisas, consultas a especialistas ou com o envolvimento de usuários chave nessas etapas, considera-se que o software poderá ser utilizado por quaisquer pessoas independentemente de suas particularidades. Segundo Martínez *et al.* (2022), afirmam que “projetar e desenvolver uma ferramenta [...] e, depois de concluída, aplicar um novo ciclo de desenvolvimento para torná-la acessível, por exemplo, para pessoas com deficiência visual. Esta é uma maneira errada de apresentar a solução para o problema. Ao fazer isso, a adaptação do aplicativo final é afetada por decisões de design anteriores e ambos, os desenvolvedores e as empresas que financiarão o projeto, serão forçados a realizar dois desenvolvimentos: primeiro eles implementarão o aplicativo seguindo uma metodologia padrão, e então eles vão adaptá-lo para torná-lo acessível. Esta adaptação implica um aumento dos custos de desenvolvimento, o que significa, em muitos casos, que não é implementada para poupar custos” (MARTINEZ *et al.*, 2022).

À luz do princípio da usabilidade, especialmente o item de acessibilidade da ISO 25010:2011, podem-se adotar medidas simples, por exemplo: no caso de uso de textos em tela usar fontes conhecidas como Times New Roman ou Georgia (fontes serifadas), ou Arial, Helvetica ou Verdana (fontes não-serifadas) que são facilmente reconhecidas e contribuem para a compreensão de usuários com baixa acuidade visual e dificuldades relacionadas à alfabetização, pois estas são fontes populares, antigas e de fácil compreensão. Por exemplo, Tarita-Nistor *et al.* (2013) trazem em seu estudo que os melhores resultados obtidos nos testes com um grupo que possuía degeneração macular relacionada à idade, foi utilizando a fonte Courier, que é serifada. Minakata *et al.* (2023) cita que a grande popularidade das fontes serifadas aumenta as chances de reconhecimento das letras e, por consequência, da leitura. Nos estudos citados também é falado sobre o tamanho das fontes, portanto opções

de aumentar ou diminuir o tamanho da fonte no software também pode ser uma medida válida a ser adotada.

Ainda neste nicho da escrita, podemos dizer empiricamente, já que ainda não há conclusões acadêmicas e bibliografia suficiente sobre o uso de linguagem neutra, que é necessário ter cautela ao utilizar-se este tipo de recurso, pois testes com leitores de tela podem não reconhecer palavras quando escritas com “@” ou “x” no final, como “del@” ou “delx”, o uso do “u” ao final também pode levar a uma confusão no contexto, por uma geração de parônimos como “elu”. Portanto deve-se atentar aos usuários-alvo do software para não excluir um grupo em detrimento de outro.

A grande maioria da população possui a capacidade de comunicação oral, “considera-se a incidência de surdos pré-linguísticos na base de 1 a 4 a cada 1.000 habitantes” (SILVA *et al.* 2021 *apud* BUBBICO *et al.*, 2007)¹, sendo que surdos pré-linguísticos são as pessoas que nasceram surdas ou adquiriram surdez antes de aprender uma linguagem oral. Existem ferramentas para tradução dos textos em tela para linguagem de sinais. No Brasil, por exemplo, temos o VLibras, disponibilizado pelo governo federal que traduz textos em português para a linguagem brasileira de sinais, utilizando um avatar, conforme Silva *et al.* (2022) e aponta que o VLibras “é importante como um projeto sem fins lucrativos com um modelo livre, aberto e colaborativo com amplos impactos positivos na comunidade surda” (SILVA *et al.*, 2022). Nos estudos de Silva *et al.* (2021) são apontados problemas na tradução feita pelo VLibras, como textos grandes terem pausas durante a tradução, dificultando a tradução simultânea. Todavia é uma ferramenta válida que, apesar de precisar de aperfeiçoamento, auxilia na utilização de websites por pessoas surdas.

Uma ferramenta interessante e amplamente difundida é a cartilha de acessibilidade da W3C (grupo de colaboração internacional para auxiliar no crescimento da web através de padrões), a WCAG 2.1, desenvolvida por um grupo de trabalho focada com acessibilidade na web, e a partir daí surgem também, diretrizes para acessibilidade em aplicativos móveis (CHICANELLI *et al.*, 2018). Resumidamente, as diretrizes para acessibilidade da WCAG 2.1, segundo Todorov *et al.* (2022) são:

“- Perceptível - as informações e os componentes da interface do usuário devem ser apresentáveis aos usuários de forma que possam ser percebidos;

- Operável - Os componentes da interface do usuário e a navegação devem ser operáveis. A interface não pode exigir interação que um usuário não possa realizar;

- Compreensível - os usuários devem ser capazes de entender as informações, bem como a operação da interface do usuário;

- Robusto - o conteúdo deve ser robusto o suficiente para que possa ser interpretado de forma confiável por uma ampla variedade de agentes de usuário, incluindo tecnologias assistivas” (TODOROV *et al.*, 2022).

Em sua pesquisa Todorov *et al.* (2022) trazem a avaliação de sites de museus na Bulgária, através de testes automatizados baseados na WCAG 2.1 e de testes manuais por usuários cegos ou com deficiência visual, concluindo que a maioria dos sites avaliados não atendem a média de critérios de acessibilidade. Sites que serão disponibilizados a toda população, como no exemplo, em que não se pode definir as particularidades de cada usuário, devem contar com o máximo de acessibilidade possível, podem ser utilizadas ferramentas que auxiliem nos testes, “o software de teste de acessibilidade da Web existente para sites é muito útil para desenvolvedores da Web e criadores de sites, bem como para instituições” (TODOROV *et al.*, 2022), nestes casos de aplicações disponíveis para grandes públicos deve-se considerar estratégias como esta ferramenta de teste automatizado, já na concepção dos testes, sendo que a acessibilidade de usuário não pode ser negligenciada desde a concepção inicial do projeto.

O nicho de raça e etnia jamais pode ser negligenciado na construção de um software. Souza *et al.* (2021) demonstram em seu estudo o quanto algoritmos utilizados na justiça criminal americana são tendenciosos ao reconhecimento de culpados quando analisam pessoas negras ou com traços hispânicos, evidenciando o racismo enraizado na sociedade. Sandvig *et al.* (2016) abordam o exemplo do algoritmo das câmeras de computadores HP, que na época, não detectaram a imagem de rostos negros, fato que foi rapidamente corrigido pela fabricante após a repercussão negativa do caso. Sandvig *et al.* (2016) afirmam, ainda, que provavelmente os desenvolvedores que trabalharam na solução assim como a empresa em si, não tinham a intenção de serem racistas nesse caso. Pode-se concluir que houve uma falha na inclusão de diferentes amostras de rostos para detecção do algoritmo nessa situação. Pensando nos tempos atuais e nos ODSs, tal falha seria

inadmissível na construção de um software. No exemplo da HP, uma fabricante industrial, com produção massiva que não teria como inferir sobre todos os usuários de um produto, o que podemos fazer é ampliar ao máximo a visão holística sobre os possíveis usuários, para evitarmos este tipo de situação. Cabe a nós, como pesquisadores e desenvolvedores, atentar para que estas situações não aconteçam, inclusive em fatores simples como uso de avatares num software.

Nunes *et al.* (2023) abordam que a área de tecnologia da informação e comunicação (TIC) ainda possui formas de discriminação de gênero, sendo uma área ainda dominada por homens, ilustra que “as mulheres representam apenas 3% dos graduados em TIC em todo mundo” (NUNES *et al.*, 2023 *apud* UNESCO, 2017)². Essa disparidade de gênero pode resultar em produtos com viés misógino e/ou machista, visto que a maioria das equipes são formadas apenas por homens, Nunes *et al.* (2023) citam a criação de tecnologias cor de rosas e simplistas quando voltadas a mulheres, como jogos digitais sobre moda, ou a utilização de elementos em tela representando o gênero masculino “que normalmente são considerados neutros, não apenas impactam negativamente o sentimento de pertencimento das usuárias, mas também a maioria dos usuários em potencial” (NUNES *et al.*, 2023), podemos deduzir um exemplo de alguma imagem ou avatar de um homem branco num sistema, que não é representativo para mulheres, negros e pessoas LGBTQIA+. Pequenos cuidados no desenho das interfaces gráficas podem solucionar ou ao menos mitigar essa sensação de não representatividade.

Em sua pesquisa, Nunes *et al.* (2023) propõem requisitos para que o software não seja discriminatório, requisitos que podem ser integrados aos requisitos funcionais e não funcionais do projeto, colocando como um dos pontos chave a importância da análise das características dos usuários do sistema para avaliar quem está priorizado e negligenciado “para a construção de um sistema de software que beneficiará todos os seus usuários, independentemente das diferenças de gênero na vida e nas responsabilidades de trabalho dos atores humanos” (NUNES *et al.*, 2023).

Outros fatores que podem ser considerados para a não exclusão de pessoas dos projetos estão descritos na seção sobre o ODS 9.

REFERÊNCIAS

¹Silva, A. L. D. C., Sá, T. M. D., Diniz, R. S., Ferreira, S. B. L., Siqueira, S. W. M., Bourguignon, S. C. When just Ok, is not Ok: An Experimental Study through Sequential Chronological Cuts, with Prescriptive and Semantic Analyzes on the Dynamic Translation by VLibras Avatar. Proceedings of the XX Brazilian Symposium on Human Factors in Computing Systems. 2021. DOI: <https://doi.org.ez41.periodicos.capes.gov.br/10.1145/3472301.3484343> apud Bubbico, L., Rosano, A., Spagnolo, A. Prevalence of prelingual deafness in Italy. *Acta otorhinolaryngologica Italica*. v. 27, p.17. 2007

²Nunes, I., Moreira, A., Araujo, J. (2023). GIRE: Gender-Inclusive Requirements Engineering. *Data & Knowledge Engineering*, v. 143. 2023. DOI: <https://doi.org/10.1016/j.datak.2022.102108> apud UNESCO. Cracking the code: Girls' and women's education in science, technology, engineering and mathematics (STEM). 2017.

ONU. "ODS 10 - Redução das desigualdades". Disponível em: <https://brasil.un.org/pt-br/sdgs/10>. Acesso em: 08 de mai. de 2023.

Swartz, A., LeFevre, A. E., Perera, S., Kinney, M. V., and George, A. S. Multiple pathways to scaling up and sustainability: an exploration of digital health solutions in South Africa. *Global Health*. 2021. DOI: <https://doi.org/10.1186/s12992-021-00716-1>

Heijsters, F. A. C. J., Van Loon, G. A. P., Santema, J. M. M., Mullender, M. G., Bouman, M., De Bruijne, M. C., Van Nassau, F. A usability evaluation of the perceived user friendliness, accessibility, and inclusiveness of a personalized digital care pathway tool. *International Journal of Medical Informatics*, v.177. 2023. DOI: <https://doi.org/10.1016/j.ijmedinf.2023.105070>

Ufua, D. E., Emielu, E. T., Olujobi, O. J., Lakhani, F., Borishade, T. T., Ibidunni, A. S., Osabuohien, E. S. Digital transformation: A conceptual framing for attaining Sustainable Development Goals 4 and 9 in Nigeria. *Journal of Management & Organization*, v. 27(5), p. 836-849.2021. DOI:10.1017/jmo.2021.45

Martínez, C. M., Piorno, J. R., Escribano Otero, J. J., Mata-García, M. G. Responsive inclusive design (RiD): a new model for inclusive software development. *Universal Access in the Information Society*. 2022. DOI: <https://doi.org/10.1007/s10209-022-00893-9>

Tarita-Nistor, L., Lam, D., Brent, M. H., Steinbach, M. J., González, E. G. Courier: a better font for reading with age-related macular degeneration. *Canadian Journal of Ophthalmology*, v. 48, p. 56-62. 2013. DOI: <https://doi.org/10.1016/j.jcjo.2012.09.017>.

Minakata, K., Eckmann-Hansen, C., Larsen, M., Bek, T., Beier, S. The effect of serifs and stroke contrast on low vision reading. *Acta Psychologica*, v. 232. 2023. DOI: <https://doi.org/10.1016/j.actpsy.2022.103810>.

Chicanelli, R.T., de Souza, P.C., Borges, L.C.L.d.F. Mobile-PrivAccess: Method for Analyzing Accessibility in Mobile Applications from the Privacy Viewpoint Abiding by W3C. In: Antona, M., Stephanidis, C. (eds) *Universal Access in Human-Computer Interaction. Methods, Technologies, and Users*. UAHCI 2018. Lecture Notes in Computer Science(), v. 10907. 2018. DOI: https://doi.org.ez41.periodicos.capes.gov.br/10.1007/978-3-319-92049-8_2

andvig, C., Hamilton, K., Karahalios, K., & Langbort, C. (2016). Automation, algorithms, and politics| when the algorithm itself is a racist: Diagnosing ethical harm in the basic components of software. *International Journal of Communication*, v. 10. 2016

ilva, A. L. D. C., Sá, T. M. D., Diniz, R. S., Ferreira, S. B. L., Siqueira, S. W. M., Bourguignon, S. C. Prescriptive and Semantic Analysis of an Automatic Sign Language Translation: Cases on VLibras Avatar Translation Using Video Interviews and Textual Interactions With a Chatbot. *Interacting with Computers*. 2022. DOI: <https://doi.org/10.1093/iwc/iwac020>

Silva, A. L. D. C., Sá, T. M. D., Diniz, R. S., Ferreira, S. B. L., Siqueira, S. W. M., Bourguignon, S. C. When just Ok, is not Ok: An Experimental Study through Sequential Chronological Cuts, with Prescriptive and Semantic Analyzes on the Dynamic Translation by VLibras Avatar. *Proceedings of the XX Brazilian Symposium on Human Factors in Computing Systems*. 2021. DOI: <https://doi-org.ez41.periodicos.capes.gov.br/10.1145/3472301.3484343>

Souza, M. C. S. de, Oliveira, M. S. M. de. O Racismo Algorítmico E A Caverna De Platão: Como As Novas Tecnologias Modulam O Estigma Do Criminoso. *Anais do 12º congresso internacional de ciências criminais da PUCRS*. 2021.

Todorov, T., Bogdanova, G., Todorova–Ekmekci, M. Accessibility of Bulgarian Regional Museums Websites. *International Journal of Advanced Computer Science and Applications*, v. 13(3). 2022. DOI:10.14569/IJACSA.2022.0130305

Nunes, I., Moreira, A., Araujo, J. (2023). GIRE: Gender-Inclusive Requirements Engineering. *Data & Knowledge Engineering*, v. 143. 2023. DOI: <https://doi.org/10.1016/j.datak.2022.102108>

APÊNDICE F - ODS 12, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

ODS 12 e metas em destaque:

ODS 12: “Assegurar padrões de consumo e produção sustentáveis” (ONU, s.d.).

12.8: “Até 2030, garantir que as pessoas, em todos os lugares, tenham informação relevante e conscientização para o desenvolvimento sustentável e estilos de vida em harmonia com a natureza” (ONU, s.d.)

Gaspar *et al.* (2019) afirmam que as metas deste ODS não clarificam objetivamente medidas específicas para alcançar cada uma das metas, mas que “ênfatisa abordagens políticas voluntárias e indiretas para alcançar os padrões de consumo e produção sustentáveis”, também afirma que o ODS conta com a “fé na capacidade humana de administrar os impactos ambientais adversos do crescimento econômico sem fim, [...], por meio de inovação tecnológica e cooperação” (GASPER *et al.*, 2019).

Considerando assim que o ODS 12 aborda de forma muito ampla as questões de consumo e produção, adotar pequenas medidas conscientes num processo de desenvolvimento de software já contribui com o objetivo. De maneira geral, todas as medidas sugeridas na presente dissertação se interligam na contribuição direta ou indireta ao ODS 12.

Na prática, o que podemos adotar:

Pilotite, explicado por Lampariello *et al.* (2021) como projetos de software que não obtêm êxito já na fase piloto, é um fenômeno que vai na contramão da produção sustentável, já que os recursos utilizados no desenvolvimento do projeto são desperdiçados pois o software não é utilizado. Para evitar este fenômeno, realizar uma pesquisa prévia ao início do desenvolvimento do projeto, buscando sistemas de software semelhantes aos já existentes, se estes têm boa aceitação no mercado e/ou do público-alvo, é uma boa estratégia para verificar se vale a pena iniciar o projeto.

Sugere-se listar alguns destes sistemas de software já existentes e suas principais funcionalidades para fazer um comparativo e a partir daí pensar em funcionalidades diferenciais, inclusive, que contribuam aos ODSs.

Deve ficar evidente aos stakeholders qual o enfoque do software e como ele se destaca de soluções já existentes e, caso, não pareça viável o desenvolvimento do projeto, por mais que haja interesse e demanda dos stakeholders para produção, criar produtos repetidos ou que não tenham funcionalidades que possam destacar o produto, visando o sucesso da sua utilização, pode criar um produto obsoleto e contribuir para uma produção demasiada de “mais do mesmo”.

Definição clara dos stakeholders: no estudo de revisão sistemática inicialmente desenvolvido nesta dissertação, evidencia-se a importância da identificação de stakeholders do projeto, para que o levantamento de requisitos, alinhamento de expectativas e negociações. Sendo assim, os stakeholders devem ser listados, sejam eles usuários diretos do software, clientes, órgãos governamentais, dentre outros, pois estes serão pontos fundamentais na disseminação das informações sobre os ODSs e as medidas para contribuição com esses ODSs que serão adotadas durante o projeto, contribuindo à Meta 12.8.

Ao término do projeto, seja na fase piloto, ou nas entregas parciais, os stakeholders são as primeiras pessoas que devem ter uma boa aceitação do projeto.

Objetivos e diferenciais da proposta: é necessário evidenciar a geração de valor que o projeto trará a seus usuários, quais os diferenciais que o software possuirá, para que ele não caia no fenômeno da *pilotite*. Explicitar essas informações justifica o desenvolvimento do projeto e motiva, além dos stakeholders envolvidos, a equipe que vai trabalhar, dentre outros possíveis personagens envolvidos. A contribuição ao ODS 12 vem do fato de o objetivo do desenvolvimento do projeto ser pautado em um software que realmente obtenha êxito na sua implantação, não produzindo um produto irrelevante.

Tendo as funcionalidades abrangidas e descartadas do escopo bem definidas, evita-se que sejam criadas, por parte dos stakeholders, expectativas que não serão atingidas e fazendo com que as funcionalidades abrangidas no escopo possam ser desenvolvidas de maneira consistente, pois a partir daí a equipe de desenvolvedores

poderá se atentar a questões quanto a reutilização de código, uso de padrões de programação, automação de processos, dentre outros fatores, que contribuirão com os princípios de qualidade de software definidos pela ISO 25010:2011. Também é a partir dessa definição preliminar de funcionalidades que se pode trabalhar a elicitação de requisitos do projeto, sendo que em “sistemas de grande porte é o caso de existir uma fase da engenharia de requisitos claramente identificável antes de se iniciar a implementação do sistema” (SOMMERVILLE, 2011 p.58). Em projetos ágeis, os requisitos podem variar ou sofrerem alterações ao longo do desenvolvimento, sendo assim podem ser escritos como histórias de usuário, por exemplo (SOMMERVILLE, 2011). Mas ainda assim é importante ter um ponto de partida.

Definição da equipe de desenvolvimento: uma equipe com papéis definidos clarifica as necessidades de conhecimento para que as funções de cada um sejam desempenhadas e, também, para que uns não sejam sobrecarregados em detrimento de outros. Conforme Imtiaz *et al.* (2016) a divisão das tarefas em uma equipe deve considerar, dentre outros, fatores como: experiência, conhecimento e disponibilidade dos membros do time. A equipe deve estar a par das medidas adotadas para a contribuição no cumprimento dos ODSs, a fim de que uma cultura de sustentabilidade possa ser disseminada, contribuindo para a Meta 12.8.

Reutilização de código e boas práticas de programação: verificar se há a possibilidade de reutilizar algum código já produzido pela própria equipe de desenvolvedores ou disponibilizado de maneira aberta a fim de economizar tempo e recursos de programação. A equipe de desenvolvimento deve ter em mente o princípio da manutenibilidade, ou seja, o código não ficará estático para sempre, é necessário que seja possível alterá-lo facilmente se necessário e reutilizá-lo (todo ou em partes) em outros projetos, assim enquadrando-se no princípio de manutenibilidade num todo, especialmente nos critérios de reusabilidade e modificabilidade da ISO 25010:2011. Jha *et al.* (2005) afirmam que a reutilização de código aumenta a produtividade e a qualidade do software, sendo que o reuso não descarta a personalização conforme a necessidade.

Medição de gasto energético: apesar deste item contribuir diretamente com o ODS 7, as medidas que podem ser tomadas para evitar o desperdício energético

durante o desenvolvimento do projeto e uso do software contribuem indiretamente ao ODS 12.

Interoperabilidade: a interoperabilidade de um sistema pode ser um dos fatores chave para seu sucesso, mesmo que não seja uma demanda imediata, deve estar previsto o caso de num futuro, o sistema precisar ser interoperável com outros, sem demandar grandes esforços de desenvolvimento. Destaca-se, por exemplo, sistemas de software voltados à área da saúde, a adoção de padrões internacionais que garantam que o sistema seja interoperável com outros sistemas é de suma importância. Segundo Tapsoba *et al.* (2020): “Para uma melhor coordenação dos serviços de saúde, a maioria dos hospitais precisa se comunicar por meio do compartilhamento de conhecimento. A interoperabilidade entre os sistemas de informação hospitalar é, portanto, um imperativo” (TAPSOBA *et al.*, 2020). Considera-se também o item de interoperabilidade que está no princípio da compatibilidade da ISO 25010:2011, pois com os incontáveis sistemas de software que existem em diversas áreas atualmente, um software projetado para existir apenas individual e isoladamente vai na contramão de qualquer possibilidade de inovação, inclusive podendo comprometer o sucesso do produto.

Confiabilidade e segurança: os usuários precisam sentir-se confortáveis ao usar o software, principalmente se ele conta com preenchimento de dados, é necessário que o usuário se sinta seguro na utilização, que o sistema passe confiabilidade, que conte com políticas de privacidade quando for o caso, do contrário as pessoas podem vir a abandonar ou evitar o uso do sistema. Lampariello *et al.* (2021) citam a importância de considerar o “fator humano” ao desenvolver um software, inclusive quando este fator é negligenciado, há chances de cair no fenômeno da *pilotite*.

É importante que sejam realizados testes automatizados ao longo do projeto para evitar e/ou diminuir a ocorrência de erros na aplicação, porém, conforme o público-alvo, testes de caixa preta também são de suma importância. Swartz *et al.* (2021) afirma que fatores relacionados às personas envolvidas no processo podem ser mais desafiadores do que questões tecnológicas em si. No estudo de revisão sistemática realizado para esta dissertação é evidenciado as dificuldades de implementação de sistemas de software na área da saúde devido às diferenças

culturais, sociais e geográficas em que os usuários estão inseridos. Portanto, cenários de teste de caixa preta bem escritos e objetivos conforme os critérios de aceitação das funcionalidades, contribuem para a abordagem centrada no usuário citada anteriormente e aumentam a qualidade do software.

Questões como um fluxo de sistema intuitivo e otimizado, com todas as funcionalidades atendendo as expectativas, de acordo com o contexto, devem estar incluídas no cenário de teste.

Assim, além de contribuir indiretamente ao ODS 12, também contribuem ao princípio de usabilidade e ao critério de testabilidade do software, descrito no princípio de manutenibilidade da ISO 25010:2011.

Ao final do projeto num todo, ou ao final de entregas parciais, dependendo da abordagem utilizada, deve ser avaliada se a recepção do produto de software, de maneira geral, é positiva, pois os stakeholders devem estar satisfeitos com o produto entregue para que, quando for colocado em produção, obtenha êxito. Por mais óbvio que isto pareça em qualquer projeto de software, do ponto de vista dos ODSs, um software com potencial de sucesso, conforme os princípios de qualidade da ISO 25010:2011 e baseado nos fatores que objetivaram o alcance de ODS e Metas, contribui ainda mais para o ODS 12 e, logo, para todos os outros adotados durante o processo de desenvolvimento.

Por fim, após o produto final ser homologado, deve-se viabilizar a possibilidade de divulgar os esforços que foram tomados consonantes aos ODSs, pois é importante que estas informações sejam divulgadas para que o maior número de pessoas possível, além da equipe de desenvolvimento e stakeholders, tenha conhecimento sobre os ODSs, o que contribui para a Meta 12.8.

REFÊRENCIAS

ONU. "ODS 12 - Consumo e produção responsáveis". Disponível em: <<https://brasil.un.org/pt-br/sdgs/12>>. Acesso em: 08 de mai. de 2023.

Gaspar, D., Shah, A., Tankha, S. The framing of sustainable consumption and production in SDG 12. *Global Policy*, v. 10, p. 83-95. 2019. DOI: <https://doi.org/10.1111/1758-5899.12592>

Intiaz, S., Ikram, N. Dynamics of task allocation in global software development. *Journal of Software: Evolution and Process*, v. 29(1). 2017. DOI: 10.1002/smr.1832

Jha, M., Maheshwari, P. Reusing Code for Modernization of Legacy Systems. 13th IEEE International Workshop on Software Technology and Engineering Practice (STEP'05), p. 102-114. 2005. DOI: 10.1109/STEP.2005.21.

Tapsoba, L. S., Traore, Y. e Malo, S. Towards an Architecture for the Interoperability of Hospital Information Systems in Burkina Faso. The Importance of Health Informatics in Public Health during a Pandemic, p. 159. 2020. DOI: 10.3233/SHTI200518

Sommerville, I. Engenharia de software. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

Swartz, A., LeFevre, A. E., Perera, S., Kinney, M. V., and George, A. S. Multiple pathways to scaling up and sustainability: an exploration of digital health solutions in South Africa. Global Health. 2021. DOI: <https://doi.org/10.1186/s12992-021-00716-1>

Lampariello, R., and Ancellin-Panzani, S. Mastering stakeholders' engagement to reach national scale, sustainability and wide adoption of digital health initiatives: lessons learnt from Burkina Faso. Family Medicine and Community Health. v. 9(3). 2021. DOI: 10.1136/fmch-2021-000959

APÊNDICE G - ODS 17, METAS ESCOLHIDAS E RELAÇÃO COM PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

ODS 17 e metas em destaque:

ODS 17: “Fortalecer os meios de implementação e revitalizar a parceria global para o desenvolvimento sustentável” (ONU, s.d.)

17.18: “Até 2020, reforçar o apoio à capacitação para os países em desenvolvimento, inclusive para os países menos desenvolvidos e pequenos Estados insulares em desenvolvimento, para aumentar significativamente a disponibilidade de dados de alta qualidade, atuais e confiáveis, desagregados por renda, gênero, idade, raça, etnia, status migratório, deficiência, localização geográfica e outras características relevantes em contextos nacionais” (ONU, s.d.)

Durante a pandemia da Covid-19, por exemplo, ficou evidente a necessidade de parcerias globais, contribuindo em pesquisas, recursos e esforços para vencer aquele período, evidenciando a necessidade do ODS 17, conforme seu texto descritivo. Sobre a Meta 17.18, atualmente a grande maioria das coletas de dados são feitas através de sistemas de software ou transcritas a eles, portanto durante o processo de desenvolvimento de software deve-se atentar a qualidade dos coletados para que eles possam ser usados, quando houver necessidade, inclusive fortalecendo as parcerias globais. Um informativo da Organização Panamericana de Saúde, esclarece que “Dados desagregados de alta qualidade, acessíveis, seguros, atuais, abertos e confiáveis são fundamentais a fim de gerar informações valiosas para a tomada de decisões em tempo real” (PAHO, s.d.), esta afirmação ficou muito clara e evidente durante a pandemia da Covid-19.

Victoria *et al.* (2019) abordam as dificuldades nas análises de dados desagregados em saúde, especificamente no âmbito da saúde materna, infantil e neonatal. Este é um exemplo de que, se os dados fossem coletados, armazenados e seguros, sua utilização em pesquisas ou levantamentos seriam mais fáceis e assertivos.

Coletas, armazenamento e disponibilidade de dados: Atualmente, a maneira mais utilizada para coletar dados são os sistemas de software. No estudo de revisão sistemática produzido para esta dissertação, foram abordados estudos sobre sistemas de software que realizam coletas de dados de usuários para diversos fins, como: monitoramento de doenças cardiovasculares, saúde mental, indicadores de mortalidade infantil e rastreamento de Covid-19, por exemplo. Durante o desenvolvimento de um software deve-se considerar que ele seja capaz de coletar dados de maneira segura e com qualidade. Deve estar claro e bem documentado quais dados serão coletados e armazenados durante o uso do software e, ainda, qual será o seu destino. Tudo isso precisa estar documentado e adequado à legislação de onde o software será utilizado, por exemplo, no Brasil, temos a Lei Geral de Proteção de Dados, em vigência desde 18/09/2020, já a União Europeia possui o Regulamento Geral sobre Proteção de Dados (BHAIMIA, 2018). Portanto, dependendo da localidade de uso de software ou se ele será utilizado internacionalmente, deverá ser adequado às normas do seu local de uso. A exemplo da legislação brasileira, a política de privacidade do uso do sistema deve estar clara e de fácil acesso aos usuários, que devem entender a razão pela qual as coletas de dados são necessárias, as possíveis destinações desses dados, a adequação à legislação e a segurança no armazenamento.

Padronização de dados: a utilização de padrões internacionais é fundamental para a qualidade dos dados coletados, para contribuir justamente com o fortalecimento de parcerias globais. Em Dam *et al.* (2017), por exemplo, o sistema utiliza o padrão CDISC ODM, um modelo utilizado em registros de saúde. Matsumura *et al.* (2014), em seu estudo, convertem um software para os padrões CDISC ODM. Sendo que este é apenas um dos padrões adotados em sistemas de software em saúde, podemos citar como outro exemplo o padrão DICOM que “é o padrão de comunicação referente ao processamento de imagens médicas” (SCHÜTZE *et al.*, 2004), logo, um software que lida com imagens de exames de pacientes, deve adotar o padrão DICOM para tal.

Confiabilidade e segurança: “A confiabilidade e a segurança de um sistema estão relacionadas, mas um sistema confiável pode ser inseguro e vice-versa”

(SOMMERVILLE, 2011, p. 209). Idealmente podemos unir os conceitos de confiabilidade e segurança para o desenho do sistema, sendo que, segundo Sommerville (2011, p. 224) a “confiabilidade geral de um sistema depende da confiabilidade do hardware, da confiabilidade do software e da confiabilidade dos operadores do sistema” e ainda: “A segurança é um atributo do sistema que reflete sua capacidade de se proteger de ataques externos, sejam acidentais ou deliberados” (SOMMERVILLE, 2011, p. 2011), considerando o conceito de segurança exposto, deve-se avaliar se as medidas adotadas de segurança da informação realmente são eficazes contra ataques externos, pois não são raros os casos de invasões, sequestro e roubo de dados, em diversos sistemas de software utilizados em diversos âmbitos, portanto a segurança do sistema deve ser o mais imbatível possível, evitando que possíveis situações como as citadas aconteçam o que pode culminar no fracasso total do projeto do software em questão. Do ponto de vista de usuários, a confiabilidade é um fator importante, pois estes sentir-se confortáveis ao usar o software, principalmente se ele conta com preenchimento de dados, é necessário que o usuário se sinta seguro na utilização, que o sistema passe confiabilidade, que conte com políticas de privacidade quando for o caso, do contrário as pessoas podem vir a abandonar ou evitar o uso do sistema.

Todos os pontos citados anteriormente também vão ao encontro ao princípio da segurança da ISO 25010:2011.

Avaliação das funcionalidades do software: Após os testes, espera-se concluir que as funcionalidades previstas dentro do escopo estejam corretamente implementadas e em pleno funcionamento, contribuindo ao ODS 17, pois apesar deste ODS versar sobre implementações internacionais em diversos âmbitos, como financeiro, comercial, tecnológico e político, podemos restringi-lo a uma visão micro, afinal se não formos capazes de implementar funcionalidades de um projeto de software para que sua implantação seja exitosa a um grupo de usuários, independente de quão numeroso seja esse grupo, jamais conseguiremos implementar quaisquer tipos de tecnologia internacionais, ou sequer, sermos conscientes e críticos das implementações internacionais que ocorrem.

REFÊRENCIAS

ONU. “ODS 17 - Parcerias e meios de implementação”. Disponível em: <<https://brasil.un.org/pt-br/sdgs/17>>. Acesso em: 08 de mai. de 2023.

BRASIL. Lei nº 13709, de 14 de Agosto de 2018. Lei Geral de Proteção de Dados Pessoais (LGPD). Disponível em: <https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm>. Acesso em 11/05/23.

Victora, C., Boerma, T., Requejo, J., Mesenburg, M. A., Joseph, G., Costa, J. C., Vidaletti, L. P., Ferreira, L. Z., Hosseinpoor, H. R., Barros, A. J. D. Analyses of inequalities in RMNCH: rising to the challenge of the SDGs. *BMJ Global Health*, v. 4. 2019. DOI: <http://dx.doi.org/10.1136/bmjgh-2018-001295>

Matsumura, Y., Hattori, A., Manabe, S., Takeda, T., Takahashi, D., Yamamoto, Y., Murata, T., Mihara, N. Interconnection of electronic medical record with clinical data management system by CDISC ODM. *e-Health–For Continuity of Care*, p. 868-872. 2014. DOI: 10.3233/978-1-61499-432-9-868

Bhaimia, S. The general data protection regulation: the next generation of EU data protection. *Legal Information Management*, v. 18.1, p. 21-28. 2018. DOI: <https://doi.org/10.1016/j.ejrad.2003.08.001>

Schütze, B., Kroll, M., Geisbe, T., Filler, T. J. Patient data security in the DICOM standard. *European Journal of Radiology*, v. 51, p. 286-289. 2004. DOI: <https://doi.org/10.1016/j.ejrad.2003.08.001>.

Dam, J. V., Onyango K. O., Midamba B., Groosman, N., Hooper, N., Spector, J., Pillai G.C and Ogutu B. (2017) “Open-source mobile digital platform for clinical trial data collection in low-resource settings”.

Sommerville, I. Engenharia de software. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

PAHO. Por que a desagregação de dados é essencial durante pandemias. Disponível em <<https://www3.paho.org/ish/images/docs/Data-Disaggregation-Factsheet-Portuguese.pdf>>. Acesso em 11/05/2023. [s.d.]

APÊNDICE H – CHECKLIST PARA DESENVOLVIMENTO DE SOFTWARE

Processo de desenvolvimento de software					
Etapas Iniciais					
Fatores a serem verificados nas etapas de concepção inicial, definições, brainstorming, criação de documentos como visão do produto, elicitação de requisitos.					
Item nº	Contribuições à sustentabilidade	Item de verificação	Sim	Não	Não se aplica
1	Contribui ao ODS 12 e à Meta 7.3.	Foi feita uma pesquisa de mercado sobre soluções em software semelhantes a que será proposta?			
2	Contribui aos ODSs 8 e 12.	Todos os objetivos estão descritos sem ambiguidades, evidenciando os diferenciais da proposta?			
3	Contribui aos ODSs 12 e 9 e às Metas 7.3, 8.2 e 8.4.	As funcionalidades que estão dentro e fora do escopo estão bem definidas?			
4	Contribui aos ODSs 10 e 4 e às Metas 10.2 e 10.3.	Estão sendo considerados fatores inclusivos na concepção do software?			
5	Contribui aos ODSs 4 e 10 e às Metas 4.a e 9.c.	Estão sendo considerados fatores que podem excluir usuários do software, para não incorporá-los ao desenvolvimento?			
6	Contribui às Metas 12.8 e 4.7.	Os stakeholders do projeto são apresentados corretamente? Foi considerado agregar a disseminação de informações sobre sustentabilidade aos stakeholders do projeto?			

7	Contribui à Meta 4.7 e ao ODS 8, com suas Metas 8.2 e 8.4, ao ODS 9 e à Meta 12.8.	Foi definida uma equipe de desenvolvimento com as funções que cada um desempenhará? Ao longo do projeto será possível agregar a disseminação de conteúdos sobre sustentabilidade à equipe de desenvolvimento?			
Etapas Intermediárias					
Itens que devem ser considerados na fase de definição do backlog, desenvolvimento e testes.					
Item nº	Contribuições à sustentabilidade	Item de verificação	Sim	Não	Não se aplica
8	Contribui à Meta 7.3, ao ODS 8 e suas Metas 8.2 e 8.4.	Os processos que podem ser automatizados estão previstos?			
9	Contribui com o ODS 12 e as Metas 8.4 e 7.3. Abrange o princípio de manutenibilidade no critério reusabilidade da ISO 25010:2011.	É possível a reutilização de código para as funcionalidades?			
10	Contribui com o ODS 12 e as Metas 8.4 e 7.3. Abrange o princípio da manutenibilidade num todo, especialmente nos critérios de reusabilidade e modificabilidade da ISO 25010:2011.	Estão sendo consideradas boas práticas e padrões de programação?			

11	Contribui com as Metas 7.3, 8.4 e o ODS 12.	Seria possível medir o gasto energético dos dispositivos de hardware utilizados durante o desenvolvimento do projeto?			
12	Contribui à Meta 7.3. Abrange o princípio da usabilidade e da adequação funcional da ISO 25010:2011.	Os fluxos de telas dos usuários serão otimizados?			
13	Contribui ao ODS 10 e suas Metas 10.2 e 10.3 Abrange o princípio da usabilidade, especialmente o item de acessibilidade da ISO 25010:2011.	Os elementos em tela, para os usuários, são inclusivos?			
14	Contribui à Meta 9.c. Abrange o princípio da portabilidade especialmente o item adaptabilidade da ISO 25010:2011.	Foram consideradas as possíveis limitações de acesso à internet e a hardware?			
15	Contribui ao ODS 9 e às Metas 9.c, 10.2 e 10.3. Abrange o princípio da portabilidade, nos itens de adaptabilidade e instalabilidade da ISO 25010:2011.	O software pode ser acessado de diferentes dispositivos em diversos ambientes?			
16	Contribui à Meta 17.18. Abrange o princípio da	O software consegue coletar, armazenar e disponibilizar, se necessário, dados organizados, com qualidade e aderentes às legislações específicas da localidade onde está sendo desenvolvido? Isto está documentado?			

	segurança da ISO 25010:2011.				
17	Contribui com o ODS 12 e Meta 8.2. Abrange o princípio da compatibilidade, especialmente o item de interoperabilidade da ISO 25010:2011.	Está sendo prevista a interoperabilidade do software?			
18	Contribui com o ODS 12 e a Meta 7.3. Abrange o princípio de manutenibilidade, no item de testabilidade, da ISO 25010:2011.	Os cenários de teste estão documentados e escritos com base nas user stories propostas? Os critérios de aceitação das funcionalidades estão listados?			
19	Contribui para a Meta 17.18. Abrange o princípio da segurança da ISO 25010:2011.	Está claro aos usuários as coletas de dados que estão sendo realizadas?			
20	Contribui à Meta 17.18. Abrange o princípio da segurança da ISO 25010:2011.	O sistema é seguro quanto a possíveis ataques?			
21	Contribui aos ODSs 17, 12 e à Meta 8.4.	As funcionalidades previstas estão implementadas corretamente? Os fatores que agregariam valor e/ou são diferenciais do sistema realmente se destacam?			

22	Contribui à Meta 12.8.	A equipe de desenvolvimento está ciente dos fatores que estão sendo considerados para contribuir com os ODS?			
Etapa Final					
Após o desenvolvimento, testes e homologação do software, fatores que podem ser considerados:					
Item nº	Contribuições à sustentabilidade	Item de verificação	Sim	Não	Não se aplica
23	Contribui à Meta 12.8.	Foram utilizados os recursos previstos para contribuição aos ODS ao longo do projeto?			
24	Contribui ao ODS 12.	A recepção do produto de software, de maneira geral, é positiva?			
25	Contribui à Meta 12.8.	Por fim, após o produto final ser homologado, é possível divulgar os esforços que foram tomados consonantes aos ODS?			

APÊNDICE I – CHECKLIST PARA AVALIAÇÃO DE SUSTENTABILIDADE PELO SOFTWARE EM UM SOFTWARE JÁ CONSTRUÍDO

Softwares já construídos					
Fatores a serem verificados num software já existente, onde o processo de desenvolvimento não foi acompanhado pelo avaliador.					
Nome do Software:					
Item nº	Contribuições à sustentabilidade	Item de verificação	Sim	Não	Não se aplica
1	Qualquer ODS.	O propósito do software (sua temática, sua destinação) contribui a algum dos 17 ODS de maneira geral?			
2	Contribui ao ODS 12 e à Meta 7.3.	Caso você conheça algum software semelhante, o software que está sendo avaliado possui funcionalidades diferenciais?			
3	Contribui aos ODS 10 e 4 e às Metas 10.2 e 10.3.	Aparentemente, o software possui fatores inclusivos em geral?			
4	Contribui ao ODS 10 e suas Metas 10.2 e 10.3 Abrange o princípio da usabilidade, especialmente o item de acessibilidade da ISO 25010:2011.	Os elementos em tela (como fonte dos textos, cores, avatares, animações, dentre outros), para os usuários, são inclusivos?			

5	Contribui à Meta 7.3. Abrange o princípio da usabilidade e da adequação funcional da ISO 25010:2011.	Os fluxos de telas dos usuários são otimizados?			
6	Contribui ao ODS 4 e 10 e às Metas 4.a e 9.c.	São ausentes no software fatores que possam levar os usuários a se sentirem excluídos do uso?			
7	Contribui à Meta 9.c. Abrange o princípio da portabilidade especialmente o item adaptabilidade da ISO 25010:2011.	Foram consideradas as possíveis limitações de acesso à internet e a hardware?			
8	Contribui ao ODSs 9 e às Metas 9.c, 10.2 e 10.3. Abrange o princípio da portabilidade, nos itens de adaptabilidade e instalabilidade da ISO 25010:2011.	O software pode ser acessado de diferentes dispositivos em diversos ambientes?			
9	Contribui com o ODSs 12 e as Metas 8.4 e 7.3. Abrange o princípio de manutenibilidade no critério reusabilidade da ISO 25010:2011.	Caso, o acesso ao código esteja disponível: foi reutilizado código para as funcionalidades? Foram consideradas boas práticas e padrões de programação?			
10	Contribui para a Meta 17.18. Abrange o princípio da	Está claro aos usuários as coletas de dados que estão sendo realizadas, como eles são armazenados, qual sua destinação, etc?			

	segurança da ISO 25010:2011.				
11	Contribui à Meta 17.18. Abrange o princípio da segurança da ISO 25010:2011.	O sistema passa segurança aos usuários quanto a possíveis ataques?			
12	Contribui aos ODSs 17, 12 e à Meta 8.4.	As funcionalidades estão implementadas corretamente?			
13	Contribui aos ODSs 17, 12 e à Meta 8.4.	Há funcionalidades que geram valor e/ou são diferenciais do sistema para o usuário?			
14	Contribui à Meta 12.8.	Foram utilizados os recursos previstos para contribuição aos ODS ao longo do projeto?			
15	Contribui ao ODSs 12.	A recepção do produto de software, de maneira geral, é positiva?			
16	Contribui à Meta 12.8.	Por fim, é possível enxergar os esforços que foram tomados consonantes aos ODSs?			