



DOCUMENTAÇÃO DE SOFTWARE EM PROJETOS DE CÓDIGO ABERTO: Desafios e soluções estratégicas

Lara D. VIANA¹; Paulo C. SANTOS²;

RESUMO

Este trabalho, na modalidade de Revisão Bibliográfica, analisa a importância estratégica da documentação de software. A documentação é crucial para a usabilidade e manutenção de sistemas, mas enfrenta desafios, especialmente em projetos *Free and Open Source Software (FOSS)*. A falta de documentação adequada dificulta a colaboração e a evolução do software. Este artigo aborda os principais desafios, como colaboração e qualidade (identificados por Pinho, 2024, em mapeamento sistemático) e soluções, como o uso de *READMEs* e Inteligência Artificial (IA). O objetivo é demonstrar como práticas eficazes de documentação otimizam a qualidade e a sustentabilidade dos projetos. A metodologia envolveu revisão bibliográfica e estudos de caso. Os resultados indicam que abordagens estruturadas melhoram a colaboração e a eficiência. Conclui-se que investir em documentação é uma necessidade estratégica para o sucesso dos sistemas.

Palavras-chave:

Engenharia de Software; Qualidade de Software; Manutenção de Software; Projetos *FOSS*.

1. INTRODUÇÃO

A crescente complexidade dos sistemas de software no cenário tecnológico atual exige mecanismos eficazes para garantir sua compreensão, manutenção e evolução. Nesse contexto, a documentação de software emerge como um componente crítico, servindo como principal meio de comunicação sobre a estrutura e funcionamento de um sistema (AGHAJANI *et al.*, 2020; PINTO, 2021). No entanto, apesar de sua importância reconhecida, a prática da documentação enfrenta desafios significativos, sendo frequentemente negligenciada em muitos projetos.

Considerando os desafios inerentes à colaboração distribuída, garantia de qualidade e manutenção em projetos de software, especialmente no contexto *FOSS*, surge a questão: como a negligência ou inadequação da documentação impacta a sustentabilidade, a eficiência do desenvolvimento e a adoção bem-sucedida desses sistemas?

Este artigo visa, portanto, analisar a importância estratégica da documentação de software, com foco nos desafios (colaboração, qualidade, incompletude) e soluções (*READMEs*, IA, ferramentas) em projetos *FOSS*, conforme identificado em Pinho (2024). Buscar-se-á discutir o impacto da qualidade da documentação na colaboração, manutenção e usabilidade, explorar abordagens emergentes para mitigar os desafios identificados e reforçar a necessidade de tratar a documentação como um componente integral e estratégico do processo de engenharia de software,

¹Discente da Ciência da Computação, IFSULDEMINAS – Campus Muzambinho. E-mail: laradomingos257@gmail.com.

²Orientador, IFSULDEMINAS – Campus Muzambinho. E-mail: paulo.santos@muz.ifsuldeminas.edu.br

ressaltando sua importância para a qualidade e longevidade dos sistemas.

2. FUNDAMENTAÇÃO TEÓRICA

A documentação de software é um artefato essencial que comunica informações sobre um sistema, facilitando sua compreensão, uso, manutenção e evolução por diferentes partes interessadas (AGHAJANI *et al.*, 2020; PINTO, 2021). Ela abrange desde especificações e diagramas até guias e comentários no código. Sua qualidade impacta diretamente a eficiência do desenvolvimento e a sustentabilidade do software.

No contexto de projetos de software livre e de código aberto (*FOSS*), a documentação assume importância adicional devido à natureza colaborativa, distribuída e voluntária. Projetos *FOSS* permitem acesso, modificação e redistribuição do código-fonte, mas enfrentam desafios como a alta rotatividade de colaboradores e a diversidade de experiências, tornando a documentação um pilar para a gestão do conhecimento e a integração de novos membros (PINHO, 2024).

3. MATERIAL E MÉTODOS

A metodologia adotada foi a de revisão bibliográfica. A seleção dos trabalhos foi realizada por meio de uma busca estruturada em bases de dados acadêmicas de relevância, como Google Scholar e IEEE Xplore, e em repositórios institucionais. Para a prospecção dos trabalhos, foram utilizadas combinações de palavras-chave como “documentação de software”, “engenharia de software”, “qualidade de software”, “projetos *FOSS*”, “desafios documentação” e “soluções documentação”.

Os critérios de seleção definidos foram temporais e temáticos, incluindo apenas artigos, dissertações e trabalhos de conclusão de curso publicados entre 2020 e 2025. Priorizaram-se estudos que abordassem diretamente a importância da documentação, seus desafios práticos — com especial atenção ao contexto *FOSS* —, soluções propostas e revisões da literatura sobre o tema.

Após a seleção, os documentos foram lidos na íntegra para uma análise. Foram extraídos e sintetizados dados pertinentes sobre a relevância da documentação, seus desafios recorrentes (colaboração, qualidade, incompletude) e as soluções emergentes, como o uso de *READMEs*, IA e abordagens como a *living documentation*. Todo o material foi organizado para fundamentar este artigo, seguindo as normas da ABNT para citações e referências. Os recursos materiais se limitaram a computadores com acesso à internet e softwares de edição de texto e leitura.

4. RESULTADOS E DISCUSSÃO

A literatura recente (2020-2025) evidencia desafios persistentes na documentação de software. A perspectiva dos praticantes, analisada por Aghajani *et al.* (2020), frequentemente revela

que, apesar de reconhecida sua importância, a documentação muitas vezes não recebe a prioridade devida nas rotinas de desenvolvimento. O mapeamento sistemático de Pinho (2024) em projetos *FOSS* corrobora essa percepção, identificando cinco desafios centrais: colaboração, qualidade, incompletude, manutenibilidade e categorização. Pinto (2021), ao analisar o papel da documentação em software *open source*, também destaca a complexidade de mantê-la útil e atualizada nesses ambientes colaborativos.

A inadequação documental impacta diretamente a integração de novos membros, como aponta Dias (2023) ao analisar dificuldades de novatos em *FOSS*, conectando-se aos desafios de colaboração e incompletude. A falta de suporte adequado na documentação inicial pode levar à desistência de voluntários, afetando a sustentabilidade do projeto.

Quanto às soluções, Pinho (2024) destaca três perspectivas investigadas nos estudos recentes: uso estratégico de *README*, adoção de IA e ferramentas de apoio. Silva (2020), ao comparar *living documentation* com a documentação tradicional, observou que, em seu contexto específico, a abordagem tradicional facilitou mais o entendimento das funcionalidades pelos desenvolvedores, sinalizando que novas abordagens automatizadas podem impor barreiras de compreensão e usabilidade.

A discussão revela que não há solução única. Documentação tradicional pode ser mais clara inicialmente (SILVA, 2020), mas sofre com a manutenibilidade, um dos desafios centrais apontados por Pinho (2024). Abordagens automatizadas aliviam a manutenção, mas exigem investimento e capacitação.

Em projetos *FOSS*, é estratégico investir em documentação de entrada clara para novatos (DIAS, 2023) e otimizar os processos de criação e manutenção (PINHO, 2024; PINTO, 2021). A combinação de abordagens — *READMEs* bem estruturados, guias tradicionais claros e ferramentas automatizadas — pode ser um caminho eficaz. Os resultados reforçam a documentação como parte integral e estratégica da engenharia de software (AGHAJANI *et al.*, 2020).

5. CONCLUSÃO

A documentação de software é estratégica para a qualidade, manutenção e evolução de projetos (AGHAJANI *et al.*, 2020; PINTO, 2021). Sua negligência gera desafios de colaboração, qualidade e manutenibilidade, especialmente em *FOSS* (PINHO, 2024). Soluções como *READMEs* otimizados, IA e ferramentas especializadas buscam mitigar esses problemas, mas a adoção deve ser contextualizada, considerando os desafios de implementação (SILVA, 2020). Investir em documentação de alta qualidade, acessível inclusive a novatos (DIAS, 2023), é crucial e exige uma cultura que a valorize como parte integrante da engenharia de software para garantir a longevidade e o sucesso dos sistemas.

REFERÊNCIAS

AGHAJANI, E. *et al.* Software documentation: The practitioners' perspective. **Information and Software Technology**, v. 126, p. 106348, out. 2020.

DIAS, Luiz Felipe Fronchetti. **Um estudo sobre a importância da documentação para iniciantes em projetos de software livre**. 2023. Dissertação (Mestrado em Ciência da Computação) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

Disponível em: <https://teses.usp.br/teses/disponiveis/55/55134/tde-14082023-161348/pt-br.php>.

Acesso em: 2 jun. 2025.

PINHO, Francisca Giniele do Nascimento. **Desafios e soluções na documentação em projetos de software livre e de código aberto: um mapeamento sistemático da literatura**. 2024. 50 f.

Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) – Universidade Federal do Ceará, Campus de Crateús, Crateús, 2024. Disponível em:

<https://repositorio.ufc.br/handle/riufc/77431>. Acesso em: 2 jun. 2025.

PINTO, Victor Hugo Miranda. **O papel da documentação no desenvolvimento de softwares open source: uma análise e um estudo de caso**. 2021. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2021.

Disponível em: <https://bccdev.ime.usp.br/tccs/2020/victorhmp/monografia.pdf>. Acesso em: 3 jun. 2025.

SILVA, André Santiago da Fonseca. **Documentação de software: uma análise comparativa entre documentação tradicional e living documentation**. 2020. Dissertação (Mestrado em Ciência da Computação) - Programa de Pós-Graduação em Ciência da Computação, Universidade Federal de Campina Grande, Campina Grande, 2020. Disponível em:

<http://dspace.sti.ufcg.edu.br:8080/jspui/handle/riufcg/13968>. Acesso em: 2 jun. 2025.