

Ensino e Implantação de Práticas de Inspeção Contínua na Manutenção de Software: Um Relato de Experiência na Disciplina de Manutenção de Software

Antônia Naelly Freire de Lima¹, Jacilane de Holanda Rabelo²

¹Universidade Federal do Ceará (UFC) - Russas, CE - Brazil

naellyfreire@alu.ufc.br, jacilane.rabelo@ufc.br

Abstract. *Continuous inspection is an essential practice to ensure the delivery of quality software, but teaching these techniques presents challenges in higher education institutions. In the job market, it is fundamental in improving the quality and efficiency of the final product. This study focused on the implementation of the SonarCloud tool during a refactoring project in the course of Software Maintenance course (2023.2) at the Federal University of Ceará - Campus Russas, with the aim of evaluating its impact on students' education. The results showed that students valued the adoption of the tool, recognizing its importance for delivering high-quality software. This experience report highlights the relevance of integrating industry tools into the academic environment, promoting learning that is more aligned with professional practices and preparing students for the challenges of the software industry.*

Resumo. *A inspeção contínua é uma prática essencial para garantir a entrega de software de qualidade, mas o ensino dessas técnicas enfrenta desafios nas instituições de ensino superior. No mercado de trabalho, ela é fundamental para melhorar a qualidade e a eficiência do produto final. O presente trabalho focou na implantação da ferramenta Sonar Cloud durante um projeto de refatoração na disciplina de Manutenção de Software (2023.2) da Universidade Federal do Ceará - Campus de Russas, com o objetivo de avaliar seu impacto na formação dos alunos. Os resultados mostraram que os alunos valorizaram a adoção da ferramenta, reconhecendo sua importância para a entrega de software de alta qualidade. Este relato de experiência destaca a relevância de integrar ferramentas do mercado de trabalho ao ambiente acadêmico, promovendo um aprendizado mais alinhado com as práticas profissionais e preparando os alunos para os desafios da indústria de software.*

1. Introdução

O processo de manutenção de *software* é essencial no ciclo de vida de qualquer sistema, não apenas para garantir seu funcionamento contínuo, mas também para acompanhar as demandas e a complexidade resultantes das tecnologias que surgem constantemente [Magalhães *et al.* 2020]. Isso incita a busca por abordagens inovadoras e eficazes, não apenas para corrigir falhas pontuais, mas para promover a evolução dos sistemas.

Com isso em mente, entra a inspeção contínua, uma prática que envolve a análise constante do código-fonte para detectar problemas como *bugs*, violações de padrões e

vulnerabilidades [Ferreira *et al.* 2023]. Este estudo examina sua integração na fase de refatoração na disciplina de Manutenção de *Software* da Universidade Federal do Ceará - UFC, buscando melhorar a qualidade do código e preparar os alunos para o mercado, além de incentivar práticas mais eficazes na engenharia de *software*.

Este trabalho está estruturado da seguinte forma: a Seção 2, *Background* e Trabalhos Relacionados apresenta os conceitos fundamentais e estudos que contribuíram para este artigo; a Seção 3, Metodologia descreve a condução do estudo e suas etapas; a Seção 4, Resultados destaca a influência positiva do trabalho no contexto aplicado; e, por fim, a Seção 5, Conclusão com um resumo e conclusões sobre o trabalho.

2. Background e Trabalhos Relacionados

A manutenção de *software* é um processo definido como um conjunto de atividades realizadas após a entrega do sistema para garantir seu bom funcionamento, sendo estas a manutenção corretiva, preventiva, adaptativa e perfectiva [ISO/IEC/IEEE 2022]. Visando o melhor desempenho do sistema, existe a refatoração de *software*, uma alteração feita na estrutura do código-fonte para torná-lo mais fácil de ser entendido e menos custoso de ser modificado, sem alterar o seu comportamento externo [Fowler 1999].

Seguindo a premissa de melhoria do *software* entra a inspeção de *software*, uma técnica que garante a qualidade do sistema e de seus artefatos, proporcionando maior eficiência e economia a longo prazo. Na inspeção contínua o foco é na análise automatizada e regular do código durante o desenvolvimento, utilizando ferramentas de análise estática para identificar *bugs*, vulnerabilidades, complexidade excessiva e violações de padrões de codificação [de Andrade Gomes *et al.* 2017].

Essas ferramentas podem melhorar a velocidade e a precisão do desenvolvimento de *software* [Eilertsen 2020]. No mercado, há uma variedade grande de *softwares* que têm por objetivo a automatização do processo de revisão. Dentre essas, destacam-se ferramentas como o Visual Expert¹, Sonar Qube² para soluções locais e o Sonar Cloud³.

[Lu 2020] avaliaram, ao longo de dois anos, o impacto da inspeção contínua na qualidade da programação com 48 alunos, mostrando que a prática ajudou a corrigir hábitos inadequados e reduzir erros, usando GitLab e Sonar Qube. Já a pesquisa de [Gomes 2021] propôs melhorias no ensino, por meio do Sonar Qube e Teacher Mate, o que permitiu que os professores analisassem os erros dos alunos, além de permitir aos alunos desenvolver habilidade de codificação.

Por sua vez, no estudo de [Lopes *et al.* 2013], o foco está no processo de ensino-aprendizado de inspeção de *software* através de jogos educativos. Assim, o artigo apresenta o jogo InspSoft 2.0, que auxilia na aprendizagem da inspeção de *software*. O jogo pode aprimorar a aprendizagem da inspeção de *software*, formando profissionais mais qualificados para a indústria.

Ao contrário dos estudos mencionados, este trabalho se destaca por incorporar na refatoração, práticas e ferramentas de análise de código. Isso torna a abordagem mais completa, prática e acessível, facilitando o desenvolvimento e o aprimoramento do código

¹<https://www.visual-expert.com/>

²<https://www.sonarsource.com/products/sonarqube/downloads/>

³<https://www.sonarsource.com/products/sonarcloud/signup/>

em um ambiente acadêmico. Além disso, busca aproximar os estudantes das demandas e ferramentas que eles encontram no mercado de trabalho.

3. Metodologia

Esta Seção foi dividida da seguinte forma: (1) Experiência Profissional e Revisão da Literatura; (2) Refinamento do escopo do relato; (3) Implementação e Estudo de caso e (4) Coleta de dados e avaliação dos resultados.

3.1. Experiência Profissional e Revisão da Literatura

A primeira fase deste trabalho, envolveu o estudo da experiência da primeira autora e a revisão da literatura sobre temas relacionados. Com formação acadêmica e atuação como analista de sistemas e Engenheira *DevOps*, a pesquisadora tem uma visão prática dos desafios da manutenção de *software*, o que reforça a relevância do estudo. Além disso, foram realizadas buscas em bases acadêmicas, como *IEEE Xplore*, *ACM Digital Library* e *Scopus*, para identificar lacunas no ensino de inspeção contínua.

3.2. Refinamento do escopo do relato

Esta etapa refinou o escopo deste trabalho, selecionando as ferramentas de automação e definindo o perfil dos participantes. Optou-se pelo Sonar Cloud, para inspeção de código, e o GitHub, para automação de integração e hospedagem dos trabalhos dos estudantes. A escolha desta ferramenta de inspeção considerou à facilidade de configuração, suporte a linguagens, integração com GitHub, adoção no mercado e experiência da pesquisadora.

Além disso, foram escolhidos as fontes de estudo, dando preferência a estudos acadêmicos, e utilizou-se a disciplina de Manutenção de *Software*. A escolha se justifica pela relação direta entre manutenção de *software* e inspeção de código, abrangendo conceitos essenciais como identificação e correção de falhas, boas práticas de programação e impacto prático para os estudantes.

3.3. Implementação e Estudo de caso

A inspeção contínua na disciplina de Manutenção de *Software* foi implementada em quatro etapas: definição do escopo, coleta de dados prévia ao estudo, integração de ferramentas nos projetos dos alunos e coleta e análise dos dados posteriores. Para avaliar a efetividade, um estudo de caso foi conduzido com 48 alunos dos cursos de Engenharia de *Software* e Ciência da Computação, organizados em 9 grupos. Foi solicitado aos alunos que cadastrassem no Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA) a equipe e projetos, informando linguagem de programação escolhida e o *link* do repositório no GitHub. Antes da implementação das ferramentas nos projetos, uma aula sobre inspeção contínua e ferramentas de inspeção, com foco no Sonar Cloud, foi ministrada pela aluna orientada deste trabalho. O material apresentado aos alunos pode ser encontrado no seguinte *link*: [Arquivos Artigo](#).

A implantação das práticas de inspeção contínua ocorreu de novembro a início de dezembro do semestre de 2023.2. Os alunos implementaram as ferramentas, realizaram a refatoração e documentaram as correções. A entrega dos trabalhos foi em 24/11/2023, com apresentações até 01/12/2023. Após isso, a docente avaliou os projetos com base nos critérios definidos e atribuiu as pontuações correspondentes.

3.4. Coleta de dados e avaliação dos resultados

Antes da implementação das práticas de inspeção e automação, um questionário baseado na escala *Likert* foi aplicado para avaliar o conhecimento prévio dos alunos, dividido em seis níveis. Ao final do projeto, um segundo questionário foi utilizado para avaliar a experiência dos alunos, incluindo perguntas repetidas para comparação. Para análise motivacional, foram empregados o Inventário da Motivação Intrínseca (IMI) e o Inventário da Motivação de Materiais Instrucionais (IMMS).

Em ambos os questionários, foi incluído um termo de ética explicando o objetivo do estudo, o uso dos dados e garantindo ao aluno o direito de recusa sem afetar na sua nota na disciplina. Ou seja, todos os participantes contribuíram de maneira voluntária e assinaram o Termo de Consentimento Livre e Esclarecido (TCLE). Os questionário e os termos podem ser acessados no seguinte *link*: [Arquivos Artigo](#).

O IMI foi avaliado em escala *Likert* de 1 a 7, onde 1 representa “Discordo totalmente” e 7 “Concordo totalmente”. O instrumento inclui subescalas para avaliar aspectos como interesse/prazer, competência percebida, esforço/importância, pressão/tensão, percepção de escolha e valor/utilidade.

A média de cada subescala foi calculada pela fórmula:

$$\text{media_subescala} = \frac{\sum(\text{frequência_escala} \times \text{peso_escala})}{\text{qtde_total_resp_escala}}$$

Os níveis de interpretação foram definidos como: ≤ 3.0 (não motivado), > 3.0 e < 5.0 (pouco motivado) e ≥ 5.0 (muito motivado). Para questões invertidas, a escala foi ajustada conforme referências [Azevedo *et al.* 2019].

O IMMS foi utilizado para medir a motivação em três dimensões do modelo ARCS: Relevância, Satisfação e Atenção [Oliveira 2024]. Para a análise, utilizou o Grau de Favorabilidade (GF), que soma as respostas “Concordo” e “Concordo totalmente” para identificar a percepção positiva dos alunos sobre a abordagem utilizada em sala de aula [Oliveira 2024].

4. Resultados

Primeiramente, foi detalhado o perfil das pessoas respondentes dos formulários. Embora 48 estudantes estivessem matriculados na disciplina, apenas 40 responderam aos questionários. Desses, 27 responderam ao primeiro e 36 ao segundo. Entre os respondentes 23 (57,5%) alunos cursam Engenharia de Software e 17 (42,5%) Ciência da Computação. Além disso, 31 (77,5%) são do gênero masculino e 9 (22,5%) feminino. A faixa etária dos acadêmicos é composta por 7 (17,5%) estudantes entre 17 a 20 anos, 25 (62,5%) entre 21 a 23 anos, 6 (15,0%) entre 24 e 26 anos e 2 (5,0%) respondentes estão entre 27 a 30 anos. Todos os alunos participantes assinaram o termo de ética concordando em participar do estudo.

Foi feita uma análise para verificar o progresso dos alunos em relação ao conhecimento sobre inspeção contínua. Para isso, comparou-se os dados coletados nos dois formulários aplicados na disciplina. Para os participantes que responderam apenas o primeiro formulário foram classificados como retrocesso, enquanto aqueles que responderam apenas o segundo foram considerados avanço.

Os dados mostram uma melhora significativa na compreensão dos alunos sobre inspeção contínua. Após a atividade, 37 alunos (86%) demonstraram avanço no conhecimento, enquanto 4 (9,3%) apresentaram retrocesso, possivelmente por dificuldades na compreensão inicial dos conceitos. Além disso, 2 alunos (4,7%) mantiveram seu nível de conhecimento após a atividade.

Ademais, foi perguntado aos alunos sobre o conhecimento de ferramentas de inspeção/análise de *software* antes do estudo. Dos participantes, 24 (92.6%) afirmaram não terem conhecimento de nenhuma ferramenta, enquanto 2 (7.4%) declararam estar familiarizados com alguma. Esses dados indicam uma lacuna nos conhecimentos dos alunos sobre ferramentas de inspeção de *software*.

Perguntados sobre o uso de ferramentas de análise de *software* nos estudos e/ou trabalho, 10 (37.0%) afirmaram nunca terem utilizado qualquer tipo de ferramenta de inspeção antes da prática, e 6 (22.2%) utilizam raramente. Por outro lado, 5 (18.5%) declararam utilizá-las com frequência. Adicionalmente, 4 (14.8%) afirmaram utilizá-las eventualmente, e 2 (7.4%) as empregam com muita frequência. Esses dados, evidenciam a importância de promover uma cultura de análise de *software* entre os estudantes.

Além disso, sobre a importância do ensino e estudo do processo de inspeção no ambiente acadêmico, 18 (66.7%) participantes afirmaram concordar totalmente com a afirmação, enquanto 8 (29.6%) concordaram e 1 (3.7%) demonstrou indecisão sobre o assunto. Observando os dados coletados, a maioria dos participantes 96,3% (26) concorda ou concorda totalmente com a relevância, destacando a necessidade de integrá-lo no ensino das universidades.

Por fim, na questão se a utilização de ferramentas de análise de código pode contribuir para o processo de refatoração. 27 participantes, 19 (70.4%) afirmaram concordar integralmente com a afirmação, enquanto 8 (29.6%) apenas concordaram. A integração dessas práticas juntamente com treinamento e experiência prática pode preparar melhor os estudantes para o mercado de trabalho.

4.1. Inventário da Motivação Intrínseca (IMI)

No segundo questionário foi aplicado o IMI, uma técnica que investiga a motivação da experiência vivida pelos sujeitos durante a realização de uma atividade [Azevedo et al. 2019]. Os dados foram organizados em gráficos por subescalas, sendo estas “**Interesse/Prazer**”, “**Pressão/Tensão**”, “**Percepção de Escolha**”, “**Percepção de Competência**” e “**Valor/Utilidade**”, ilustrada na Figura 1.

A subescala “**Interesse/Prazer**” apresentou média de 5,80 indicando alta motivação intrínseca dos alunos. 30 alunos tiveram uma visão positiva sobre a refatoração, e 29 consideraram a inspeção contínua agradável. Ademais, 28 acharam a refatoração divertida, enquanto 25 gostaram das atividades. Além disso, 25 discordaram que a atividade foi chata. No geral, 31 alunos consideraram a inspeção contínua muito interessante, com apenas um tendo percepção negativa.

Além disso, na subescala “**Pressão/Tensão**”, apresentou uma média baixa (2,91), indicando que os alunos não se sentiram excessivamente pressionados, o que favorece o aprendizado. A maioria dos estudantes (19) não se sentiu tensa durante a refatoração, enquanto 11 relataram tensão e 6 ficaram neutros. Quanto ao nervosismo,

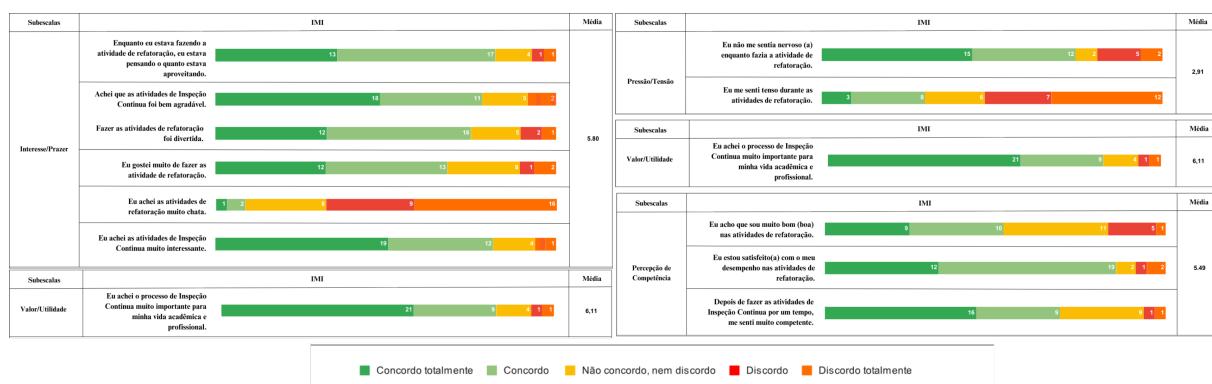


Figura 1. Dados coletados a partir do Inventário de Motivação Intrínseca (IMI)

27 alunos afirmaram não sentir, enquanto 7 demonstraram algum desconforto e 2 foram neutros. Esses dados sugerem que a metodologia aplicada proporcionou um ambiente de aprendizado mais confortável e eficiente.

Na análise da subescala “**Percepção de Escolha**” apresentou uma média alta (5,47), indicando que os alunos sentiram autonomia no processo de refatoração com inspeção contínua, fator essencial para o engajamento. Na afirmação, 26 alunos sentiram que participaram voluntariamente. Por outro lado, 10 foram neutros ou não tiveram essa percepção. Isso sugere que a atividade foi amplamente vista como uma decisão própria, fortalecendo a motivação dos estudantes.

Ademais, na subescala “**Percepção de Competência**” a média foi de 5,49, indicando que os alunos se sentiram confiantes e capazes ao realizar a inspeção contínua. Na afirmação sobre habilidade na refatoração, 19 estudantes avaliaram positivamente sua competência, enquanto 11 foram neutros e 6 discordaram. Sobre a satisfação com o desempenho, 31 estudantes concordaram, 2 foram neutros e 3 discordaram. Quanto à sensação de competência após a prática, 25 alunos concordaram, 11 foram neutros ou discordaram. Isso sugere que a maioria se sentiu capaz e envolvida durante o processo.

A subescala “**Valor/Utilidade**”, a média foi maior (6,11), demonstrando que os alunos consideram a inspeção contínua altamente relevante para suas vidas acadêmicas e profissionais. Na questão sobre a importância do processo, 30 alunos concordaram com sua relevância, enquanto apenas 2 discordaram e 4 ficaram neutros.

4.2. Inventário da Motivação de Materiais Instrucionais (IMMS)

Ademais outro técnica utilizada no segundo questionário foi o IMMS, que avaliam a motivação de sujeitos, após exposição a diferentes estratégias e materiais de ensino [Monteiro *et al.* 2015]. Que foi divida nestas dimensões: “**Atenção**”, “**Relevância**” e “**Satisfação**”, ilustrada na Figura 2.

Detalhando a dimensão “**Atenção**” percebe-se que houve um alto grau de favorabilidade 75% < GF < 98% por parte dos estudantes, evidenciando que o material de ensino das práticas de inspeção estimularam a curiosidade, o entusiasmo, o interesse e a participação ativa dos alunos. Na afirmação sobre novos conhecimentos, teve um grau de favorabilidade de 91% (33), onde 53% (19) dos participantes concordaram totalmente e 38% (14) concordaram. Apenas 6% (2) dos estudantes discordaram e 3% (1) ficaram

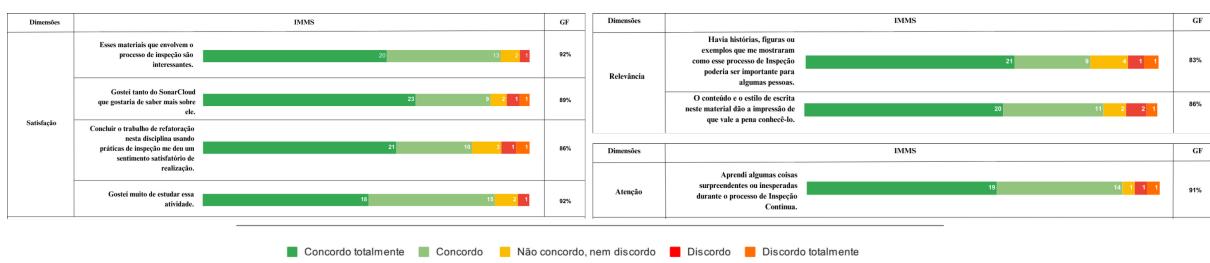


Figura 2. Dados do Inventário de Motivação de Materiais Instrucionais (IMMS)

neutros.

Já na dimensão “**Relevância**” houve alto grau de favorabilidade $81\% < GF < 100\%$ sugere que despertou na maioria dos estudantes um sentimentos de conexão por meio das dinâmicas e o processo de aprendizagem, alinhando o conteúdo apresentado com suas experiências, necessidades e preferências. Na primeira afirmação “Havia histórias, figuras ou exemplos que me mostraram como esse processo de Inspeção poderia ser importante para algumas pessoas”, obteve um grau de favorabilidade de 83% (30). Nesse caso, 58% (21) dos alunos concordaram totalmente, e 25% (9) concordaram, enquanto 6% (2) discordaram e 11% (4) ficaram neutros. Já na questão “O conteúdo e o estilo de escrita neste material dão a impressão de que vale a pena conhecê-lo” foi totalmente aceita por 56% (20) dos participantes, com 30% (11) concordando. Apenas 6% (2) ficaram neutros, 6% (2) discorda, e 2% (1) discorda totalmente, resultando em um grau de favorabilidade de 86% (33).

Por fim, a dimensão “**Satisfação**”, mostram um alto grau de favorabilidade (88% a 100%), indicando que o processo de inspeção contínua impactou positivamente os alunos, gerando um sentimento de realização. A afirmação “Esses materiais que envolvem o processo de inspeção são interessantes” teve 92% de favorabilidade, com 56% dos alunos concordando totalmente e 36% concordando. Na questão sobre o Sonar Cloud, 89% dos alunos tiveram uma percepção positiva, com 64% concordando totalmente. Quanto à afirmação “Concluir o trabalho de refatoração com práticas de inspeção me deu um sentimento satisfatório de realização”, obteve 86% de favorabilidade, com 58% dos alunos concordando totalmente. A afirmação “Gostei muito de estudar essa atividade” teve o maior grau de favorabilidade, 92%, com 50% dos alunos concordando totalmente.

4.3. Experiência com a Ferramenta Sonar Cloud

Sobre a experiência dos estudantes com o Sonar Cloud na prática apresentada no presente trabalho, a Figura 3 mostra as respostas dos 36 participantes.

Uma grande maioria dos estudantes (33) recomendaria a utilização do Sonar Cloud para outros colegas. Apenas 3 estudantes ficaram neutros ou discordaram. Da mesma forma, 34 estudantes consideram que o Sonar Cloud ajudou a melhorar a qualidade do código, mostrando a eficácia da ferramenta na detecção e correção de problemas. Apenas dois estudante ficaram neutros ou discordaram. 34 estudantes tiveram uma percepção positiva sobre o Sonar Cloud ajudar a identificar erros que não seriam facilmente encontrados em seus códigos. Apenas dois estudante ficaram neutros ou discordaram.

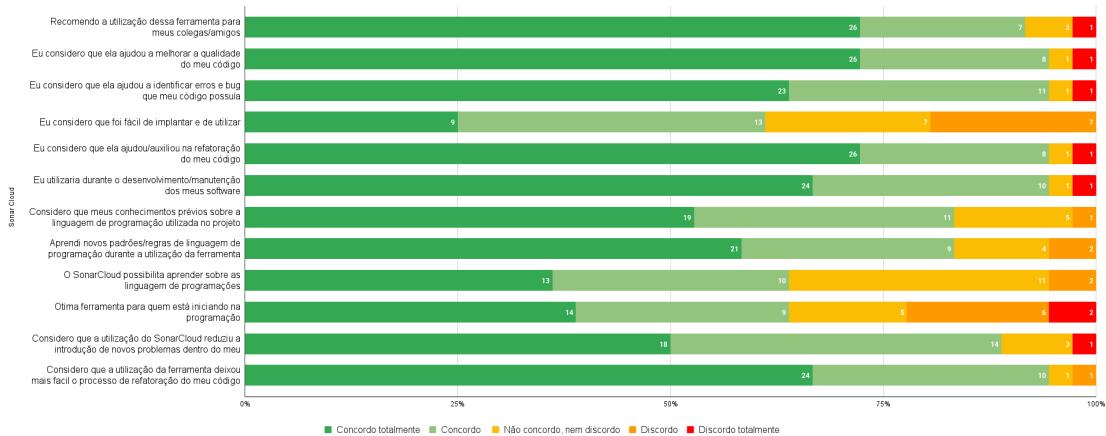


Figura 3. Respostas dos alunos sobre sua experiência com a ferramenta Sonar Cloud

No entanto, a facilidade de implantação e uso da ferramenta dividiu mais as opiniões: 22 estudantes concordaram, enquanto 7 ficaram neutros e outros 7 discordaram. Representando uma grande maioria, 34 pessoas concordam que a ferramenta auxiliou na refatoração do código e apenas dois ficaram neutros ou discordaram. Destacando o papel crucial da ferramenta na melhoria contínua do *software*. 24 estudantes afirmaram que utilizariam o Sonar Cloud em outros projetos, dois estudantes ficaram neutros ou discordaram.

Além disso, 30 estudantes consideraram que seus conhecimentos prévios sobre inspeção foram suficientes para utilizar o Sonar Cloud. Apenas 6 ficaram neutros e um discordou, indicando que a maioria se sentiu preparada, mas ainda há uma necessidade de preparação adicional para alguns. Ademais, 30 estudantes aprenderam novos padrões/regras de linguagem de programação com o Sonar Cloud. Apenas 6 ficaram neutros ou discordaram, sublinhando o valor educacional da ferramenta. Segundo a análise, 23 estudantes consideram que o Sonar Cloud possibilita aprender sobre as linguagens de programação utilizadas. 13 estudantes ficaram neutros ou discordaram, indicando que a ferramenta pode oferecer mais *insights* específicos sobre as linguagens de programação. 23 estudantes acham que o Sonar Cloud é uma ótima ferramenta para iniciantes na área de desenvolvimento. No entanto, 5 ficaram neutros, 8 discordaram, sugerindo que a ferramenta pode ser melhor adaptada para iniciantes.

Por fim, 32 estudantes concordaram que a utilização do Sonar Cloud reduziu a introdução de novos problemas no código. Apenas 4 ficaram neutros ou discordaram, destacando a eficiência da ferramenta em acelerar o processo de revisão, um dos objetivos da utilização dessas ferramentas. 34 estudantes consideram que a ferramenta tornou o processo de refatoração mais fácil. Apenas dois estudantes permaneceram neutros ou apresentaram discordância, evidenciando uma melhoria na produtividade desses alunos.

4.4. Resultados - Experiência com o processo de Inspeção Contínua e Ferramentas de Análise de código

O Figura 4 ilustra os resultados do questionário em relação à experiência dos alunos sobre o processo de inspeção contínua e o uso de ferramenta de análise de código em geral.

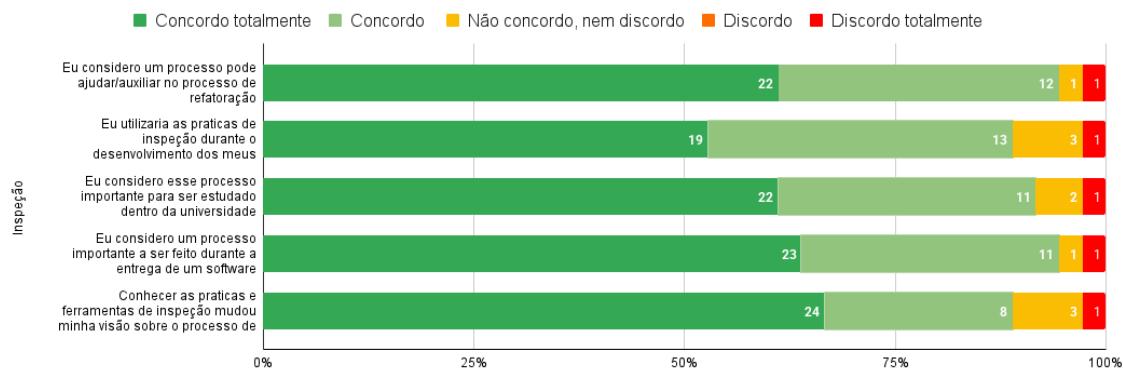


Figura 4. Respostas dos alunos sobre sua experiência com o processo de Inspeção Contínua e Ferramentas de Análise de Código

A maioria dos participantes reconhece os benefícios da inspeção contínua na refatoração de seus códigos (34 concordaram, 1 neutro). Isso demonstra uma clara percepção dos benefícios dessas práticas nessa fase acadêmica dos alunos. Em relação à utilização em outros projetos, 32 estudantes concordaram, três ficaram neutros e nenhum discordou. A importância de aprender o processo de inspeção na universidade também foi destacada, com 33 estudantes apoiando e apenas dois ficaram neutros, e nenhum discordou.

Além disso, 34 estudantes tiveram uma percepção positiva e apenas um ficou neutro para a afirmação sobre a importância do processo de inspeção ser realizado no desenvolvimento de *software*. Por fim, 32 acreditam que essas práticas ajudarão a mudar sua visão sobre o processo de inspeção, três estudantes ficaram neutros.

4.5. Resultado da Análise Qualitativa dos dados

Esta seção analisa qualitativamente as respostas dos estudantes sobre inspeção contínua e ferramentas de análise, antes e depois de atividades práticas. Para garantir a confidencialidade, os alunos foram identificados de R01 a R36.

4.5.1. Experiência durante o processo de refatoração

Os estudantes foram questionados sobre sua experiência com o trabalho em equipe durante o processo de refatoração. Conforme os relatos apresentados a seguir, os participantes destacaram aspectos positivos, como a boa divisão de tarefas (R05), que facilitou o processo, e a percepção de que essa atividade foi uma preparação para o mercado de trabalho (R11). No entanto, desafios também foram mencionados, especialmente a falta de familiaridade com a ferramenta e a linguagem escolhida, o que impactou a eficiência no início (R08).

Pergunta 01: *Como foi o trabalho em equipe durante o processo de refatoração?*

Respostas:

- R11: “Interessante, uma amostra do que seria no mercado de trabalho.”

- R05: “Dividimos cada code smell cada um deveria fazer, assim fincado [sic] mais fácil e tranquilo para todos.”
- R08: “Por ser a primeira vez que utilizei a ferramenta, o processo levou mais tempo do que deveria. Nem todos os membros da equipe utilizavam a linguagem escolhida, além de que a implementação do Sonar Cloud foi um pouco confusa para todos os membros no começo, pois ninguém havia utilizado ela previamente.”

4.5.2. Opinião sobre a inspeção continua e suas ferramentas

Os estudantes foram questionados sobre a inspeção contínua e o uso de ferramentas para análise de código. Conforme os dados apresentados a seguir, os participantes reconheceram a relevância desse processo no desenvolvimento de *software*. Além disso, sugeriram melhorias, como melhor organização do repositório (R07) e usar a ferramenta integrada ao VS Code desde o início (R010). Destacou-se a relevância para o mercado (R01, R03), a eficácia na identificação de problemas e ensino de boas práticas (R02) e o impacto no ambiente corporativo (R24).

Pergunta 01: Qual sua opinião sobre o processo de Inspeção Continua?

Respostas:

- R01: “Muito interessante pro mercado de trabalho.”
- R03: “É um processo de extrema importância em qualquer desenvolvimento de software.”

Pergunta 02: O que você achou da ferramenta que foi apresentada?

Respostas:

- R02: “Muito boa! Além de apontar os potenciais problemas, ainda aprendi bastante sobre boas práticas de código.”
- R24: “Uma ferramenta que pode impactar o mercado, quanto mais se adaptar-se ao mundo corporativo de maneira mais prática.”

5. Considerações Finais

O trabalho mostrou que a implementação de um processo de inspeção contínua e o uso de ferramentas de análise de código em um ambiente acadêmico influencia positivamente a percepção dos alunos sobre importância deste processo no desenvolvimento de *software* de qualidade, além de incentivar uma cultura de refatoração. Os dados dos inventários IMI e IMMS mostraram que os alunos melhoraram seus conhecimentos sobre inspeção e refatoração de código após a prática.

A metodologia de ensino aplicada, que utilizou o Sonar Cloud, despertou interesse e conforto nos alunos, estimulando a colaboração em equipe e aprimorando habilidades individuais. Isso preparou os estudantes para aplicar esses conhecimentos em contextos acadêmicos e profissionais, evidenciando a importância da inspeção contínua para a qualidade do *software*.

Contudo, foram identificados desafios, como a necessidade de os alunos se familiarizarem com a ferramenta escolhida e a linguagem de programação utilizada no projeto, além da sugestão de uma melhor estrutura para os processos de refatoração. Essas questões apontam caminhos para melhorias em futuras implementações.

Referências

(2019). Intrinsic motivation of medical students from a college with active methodology in brazil: a cross-sectional study. *Revista Brasileira de Educação Médica*, 43(1):12–23.

(2017). Teaching software quality via source code inspection tool. In *2017 IEEE Frontiers in Education Conference (FIE)*, pages 1–8.

Eilertsen, A. M. (2020). Refactoring operations grounded in manual code changes. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings*, ICSE '20, page 182–185, New York, NY, USA. Association for Computing Machinery.

(2023). Continuous inspection of software quality in an automotive project. In *2023 18th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6.

Fowler, M. (1999). *Refactoring: Improving the Design of Existing Code*. Addison-Wesley.

Gomes, P. H. d. A. (2021). *Inspeção de código-fonte como subsídio para o processo de ensino e aprendizagem de qualidade de software*. Universidade Estadual Paulista (Unesp), São José do Rio Preto.

ISO/IEC/IEEE (2022). Iso/iec/ieee 14764: Software engineering — software life cycle processes — maintenance. *International Standard*, pages 1–36.

(2013). Inspsoft 2.0: Um jogo para o ensino de inspeção de software. In *Anais do XII Simpósio Brasileiro de Qualidade de Software*, pages 95–107, Porto Alegre, RS, Brasil. SBC.

Lu, Yao. Mao, X. W. T. Y. G. L. Z. (2020). Improving students' programming quality with the continuous inspection process: a social coding perspective. *Frontiers of Computer Science*, 14.

(2020). Suporte às atividades de manutenção de software em bases de dados abertas e distribuídas. In *Anais do XXI Simpósio em Sistemas Computacionais de Alto Desempenho*, pages 227–238, Porto Alegre, RS, Brasil. SBC.

(2015). Intrinsic motivation inventory: Psychometric properties in the context of first language and mathematics learning. *Psicologia: Reflexão e Crítica*, 28(3):434–443.

Oliveira, M. F. (2024). Avaliação do uso de serious games em educação e treinamento corporativo: um case de montadora de veículos. Dissertação (mestrado em engenharia de produção), Faculdade de Engenharia e Ciências, Universidade Estadual Paulista, Guaratinguetá.