ATIVIDADE 01

Gestão e Qualidade de Software-ADS1AN-BUC1

Caio de Sousa Passiano-824136218

[824136218@ulife.com.br](mailto:824136218@ulife.com.br)

Tema 2:

**Negligência e Responsabilidade Civil**, de quem deveria ser a responsabilidade por erros e falhas ou danos cometidos por esse profissional? A responsabilidade civil fala sobre a exigência do usuário que vem aumentando nos últimos anos, fazendo com que muitas das vezes o usuário cobre por reparo em erros cometidos, tantos danos morais como patrimoniais, existindo artigos no código civil que exigem esse reparo.

O Código Civil, em seus artigos 186 e 927, definem a responsabilidade civil como sendo:

Art. 186 CC: “Todo aquele que, por ação ou omissão voluntária, negligência ou imprudência, violar direito e voluntária, negligência ou imprudência, violar direito e causar dano a outrem, ainda que exclusivamente moral, causar dano a outrem, ainda que exclusivamente moral, comete ato ilícito.”

Art. 927 CC: Aquele que, por ato ilícito (arts. 186 e 187), causar dano a outrem, fica obrigado a repará-lo.

Fonte: [Engenheiros de Software e suas responsabilidades | by César França | Aspectos Humanos e Sociais na Computação | Medium](https://medium.com/aspectos-humanos-e-sociais-na-computa%C3%A7%C3%A3o/engenheiros-de-software-e-suas-responsabilidades-3b3a3ba4271d)

Pensando nisso já fica fácil lembrar dos **riscos** da qualidade de software que, muitas vezes por causa da pressão, são gerados muitos atalhos que causam a redução de testes e revisões, causando também a priorização da funcionalidade em vez da qualidade.

Esse assunto já faz pensar em um bom questionamento, o que seria um **Software “bom o suficiente”?** tendo em vista as exigências dos usuários o software ideal é aquele que cumpre as funções necessárias de forma satisfatória, sem recursos avançados ou adicionais. Ele resolve o problema sem ser o melhor ou mais caro disponível na situação.

As **ações administrativas**, também são fatores importantes para que esse software seja bom o suficiente e que tenha o mínimo de erros possíveis na hora da entrega do projeto, as ações administrativas desempenham um papel crucial na qualidade do software, influenciando diretamente os processos de desenvolvimento, a motivação da equipe, o direcionamento de recursos e consequentemente, o produto final.

Definição de Requisitos: Uma gestão eficaz garante que os requisitos sejam bem definidos e compreendidos, evitando ambiguidades e mudanças frequentes que possam comprometer a qualidade do software.

Cronograma Realista: Administradores que planejam prazos realistas permitem que a equipe de desenvolvimento tenha tempo suficiente para realizar testes rigorosos e resolver bugs, resultando em um software mais robusto.

Gerenciamento de Riscos: A identificação e mitigação de riscos desde o início do projeto ajudam a prevenir problemas que possam impactar a qualidade do software.

Mais uma serie de coisas são importantes para que um projeto corra bem, como os profissionais contratados, as tecnologias ideias para o projeto, uma boa comunicação e o mais importante na minha opinião que é deixar claro os padrões de qualidade claros e fácil de serem compreendidos para todos.