

## Podcast

### Disciplina: **Qualidade de software com Clean Code e técnicas de usabilidade**

**Título do tema:** **Técnicas de Usabilidade: melhorando a qualidade da experiência do usuário.**

**Autoria:** **Stella Marys Dornelas Lamounier**

**Leitura crítica:** **Marco Ikuro Hisatomi**

#### Abertura:

Olá, ouvinte! No podcast de hoje vamos entender a importância de se tratar erros utilizando Clean Code para garantir uma excelente usabilidade para os usuários de software.

Desenvolver e realizar manutenção em sistemas computacionais é uma tarefa que envolve, além de muito conhecimento técnico, criatividade e experiência.

Os profissionais da área admitem que habilidades com técnicas e procedimentos particulares é algo único, isto é, cada um pode ser capaz de criar seu próprio código-fonte da sua maneira, utilizando métodos bem particulares. Mas com a evolução tecnológica criar sistemas deste modo já não são mais permitidos nas empresas de desenvolvimento de software; este modelo já se encontra em desuso e por sinal, muito obsoleto.

É preciso muito mais que saber levantar requisitos e criar sistemas rapidamente, é preciso dotar-se de técnicas que elevem a qualidade do código-fonte, criá-lo de forma que quem irá utilizá-lo saiba interpretá-lo e imediatamente consiga lê-lo e alterá-lo. Foi assim que surgiu o Clean Code, ou código limpo por volta de 2008, uma técnica relativamente nova que está fortemente ligada às metodologias ágeis que também está sendo muito utilizada nas empresas modernas. O Clean Code auxilia o programador a ter facilidade de entendimento, manutenção de código e realização de testes, seguindo uma série de orientações que garantem que o código adequado à manutenção com qualidade.

Neste contexto, além de padronização de variáveis, classes e métodos curtos, que devem falar por si só, determinando suas funcionalidades dentro do próprio código-fonte, é necessário que haja o mínimo de comentários possíveis.

Sobretudo, o código-fonte deve ser capaz de relatar para o que foi criado, sem rodeios, mantendo sempre simples e de fácil entendimento, mesmo depois de inúmeras alterações que geraram várias releases/versões.

Bem, agora vamos falar sobre o tratamento de erros! Deve ser realizado de forma correta para não prejudicar a usabilidade do software e assim garantir o bom funcionamento do sistema. O tratamento de erros é responsável por exibir

mensagens esclarecedoras do erro encontrado ou até mesmo de uma prevenção futura de erros.

Pode parecer estranho, mas tratar erros é também uma preocupação a parte quando se fala em usabilidade e Clean Code; saber especificar a escrita, criar mensagens limpas e claras são importantes. Tanto do ponto de vista do usuário quanto do profissional desenvolvedor para prevenir erros e reverter ações que ocasionalmente acontecem.

Sabemos que erros são cometidos, são muito indesejados, podem diminuir a produtividade, frustrando o usuário final. Portanto, devem ser detectados imediatamente e corrigidos o mais rápido possível! Vejamos. Deve se atentar para uma boa interface de software, imaginem um software em que a interface é confusa, mal planejada, de difícil entendimento, o programador gastaria horas ou até dias para a sua correção, trazendo prejuízos enormes para as empresas.

Portanto desenvolvedor, lembre-se que aplicar boas práticas de programação, seguir padrões geram confiança, facilidade e customização para o quesito usabilidade, tão importante nos dias de hoje. Propor sistemas que, ao mesmo tempo, possua um código bem entendível e de fácil manutenção, é sinal de facilidade de uso para o programador o que corrobora na criação de sistemas mais eficazes, eficientes e que satisfazem ao usuário final.

### **Fechamento:**

Este foi nosso podcast de hoje! Até a próxima!