**Recuperação** - Luana Petters e Luís Felipe Francisco

* 1. Maria está trabalhando em um projeto de desenvolvimento de software e decide adotar a abordagem do Test Driven Development (TDD) para melhorar a qualidade do código e a colaboração entre os membros da equipe. Ela começa a implementar uma nova funcionalidade seguindo os princípios do TDD. Assinale a alternativa que apresenta corretamente o processo que Maria deve seguir.

1. Escrever o teste, escrever o código, executar o teste, refatorar e, por fim, integrar o código ao repositório principal.
2. Escrever o código, executar o teste, escrever o teste, refatorar e, por fim, integrar o código ao repositório principal.
3. Escrever o teste, executar o teste, escrever o código, executar o teste novamente, refatorar e, por fim, integrar o código ao repositório principal.
4. Escrever o código, escrever o teste, executar o teste, refatorar e, por fim, integrar o código ao repositório principal.
5. Escrever o teste, escrever o código, executar o teste, integrar o código ao repositório principal e, por fim, refatorar.

1.2) Justifique a resposta

R.: Ao falar em abordagem TDD, parte-se do princípio que seu desenvolvimento parte do desenvolvimento de testes, para na sequência escrever o código, om base nos testes previamente realizados, executar o teste para validar se está a implementação está de acordo, refatorar e, por fim, subir para o repositório, como apresentado na resposta. Isso se dá, pois o TDD trata de um fluxo cíclico e focado nos processos e, consequentemente, em um código menos passível de falhas.

* 1. Como você dividiria uma Epic em várias User Stories? Dê um exemplo.

R.: Epic: Comprar roupa

I. Como cliente, defino meu orçamento para a compra

Critérios de aceitação: usuário estabelece um valor limite esteja dentro da sua condição financeira.

II. Como cliente, quero ir em lojas tanto físicas, quanto online, que vendam roupas de acordo com o estilo que me interessa.

Critérios de aceitação: usuário deve saber quais lojas físicas quer visitar e quais lojas onlines acessar.

III. Como cliente, quando em loja física, quero poder provar as roupas e, quando em loja on-line, adicionar as opções ao meu carrinho.

Critério de aceitação: usuário deve experimentar as roupas em loja física ou adicionar as roupas ao meu carrinho.

IV. Como cliente, quero finalizar a compra, de acordo com o que me serviu/gostei, sendo assim pagar, receber nota e afins.

Critério de aceitação: usuário deve concluir a compra e recolher/receber (via e-mail, mensagem) a nota fiscal.

* 1. Quais são os critérios de aceitação e por que são importantes em User Stories?

R.: São importantes, pois são capazes de gerar um melhor entendimento e definição do escopo e expectativas de determinada tarefa, visto tratarem de descrições mais detalhadas das condições que devem ser atendidas.

3.1) O que é um teste unitário e qual é o seu objetivo?

R.: São códigos que têm como objetivo testar uma parte do código, por exemplo um método. Seu objetivo baseia-se no aumento da eficiência, garantindo a confiabilidade dos componentes de forma individual.

3.2) Explique o conceito de "mocking" em testes unitários.

R.: O objetivo do *mocking* em testes unitários é isolar e controlar o comportamento de dependências externas, permitindo que a unidade de código seja testada de forma independente, reduzindo efeitos colaterais e acelerando a execução.

* 1. O que é refatoração e como ela se encaixa no XP?

R.: Refatoração é o processo de melhorar o código-fonte para torná-lo mais limpo e eficiente sem alterar seu comportamento externo. No XP, é uma prática essencial para manter a qualidade e a flexibilidade do software.

* 1. Explique o conceito de "programação em pares" e seus benefícios.

R.: Prática de desenvolvimento em que dois programadores desenvolvem uma funcionalidade/projeto simultaneamente, revisando o que cada um faz e dando feedback quanto ao caminho seguido. Alguns de seus principais benefícios são melhora na qualidade do código, comunicação eficiente, menos erros, aprendizado colaborativo.

* 1. Como o TDD (Test-Driven Development) é aplicado no XP?

R.: Em um primeiro momento são definidos os requisitos, na sequência são escritos os testes automatizados antes do código real, garantindo que o código cumpra os requisitos propostos. Os testes automatizados inicialmente criados falham, seguida pela implementação do código necessário para que os testes passem. Esse ciclo resulta em código confiável, bem testado e adaptável a mudanças.

5.1) Quais métricas você usaria para avaliar o sucesso de uma implementação de Scrum ou XP?

R.: Customer satisfaction: avalia a satisfação do cliente com os produtos entregues, capaz de atender suas necessidades;

Velocity: mede a quantidade de trabalho entregue em cada Sprint, capaz de avaliar a produtividade e previsão de entrega da equipe;

Lead Time: mede o tempo necessário para transformar um requisito em um produto ou recurso, destacando a eficiência do processo e capacidade de resposta.

5.2) Explique como User Stories e Epics podem ser usadas para facilitar a comunicação entre desenvolvedores e stakeholders não técnicos.

R.: São ferramentas ágeis que simplificam a comunicação ao descreverem funcionalidades em termos de ações do usuário e negócio. Tornando mais fácil a comunicação entre o dev e o stakeholder não técnico, ajudando a alinhar as expectativas e prioridades.