**Tarefa 3: Testes de Unidade e Introdução ao TDD**

**Fundamentação teórica**

* **Prática para obtenção de requisitos de forma colaborativa.**
* **Aplicada em equipe ágeis.**
* **Baseia-se na ideia de exemplos concretos (Cenários de testes) e testes automatizados criam especificações executáveis.**
* **Deve ser realizada antes da reunião de planning com a finalidade de executar o Gromming Backlog – detalhamento.**
* **Na Sprint Planning o ATDD deve ser apresentado com requisitos minuciados em “Gromming Backlog”.**

Diagrama

Descrição gerada automaticamente

**Figura 1: Ciclo ATDD**

**Retirado de: https://www.infoq.com/br/articles/atdd-passo-a-passo/**

**Roteiro da prática – em equipe**

**Considere este estudo de caso:**

**GAMIFICAÇÃO PARA ENGAJAMENTO DE EDUCAÇÃO CONTINUADA**

Uma determinada plataforma vende cursos online e EAD no modelo de assinaturas. O aluno paga um valor mensal e tem acesso a um conjunto de cursos para assinatura básica. A cada curso terminado e com média acima de 7,0, o aluno tem direito a realização de mais 3 cursos. O aluno que escrever mais tópicos no fórum e ajudar outros participantes com seus comentários, ganha um curso no final do mês. Quando o aluno conquistar 12 cursos, seu plano de assinatura passa a ser “Premium” e ele passa a receber voucher para participar de projetos reais, durante os cursos, e receber 3 moedas, que podem ser convertidas em conhecimento (novos cursos), acumular ou receber por criptomoeda.

**ENTREGA**: Resolva as atividades de projeto abaixo. Para cada resposta, deve ter um “print”. Além disso, o projeto deve ser entregue via link do github, de forma que seja possível o clone.

1. Cada integrante da equipe deve escrever uma User Story conforme template da Figura 2.
   1. Se a equipe tem 5 integrantes deve se ter 5 User Story.
2. A equipe deve se reunir e escolher no mínimo 1 User Story escrita.
3. Para a User Story escolhida, cada integrante da equipe DEVE escrever um BDD conforme template da Figura 2. Exemplo: de 5 integrantes, 5 BDD para uma User Story.
4. A equipe deve julgar se é possível criar uma funcionalidade válida para esta User Story escolhida e BDDs, a ponto de ser possível criar uma API.
5. Desenvolver a funcionalidade de acordo com TDD – Test Driven Development, seguindo os passos:
   1. **Passo 1 do TDD**: Considerando um ambiente Maven de desenvolvimento (Gestão de Build e Dependências): Criar os casos de testes no JUNIT de acordo com BDD e fazer o teste falhar para todos os BDD. Printar aqui as imagens dos testes falhando e os stubs gerados.
   2. **Passo 2 do TDD**: Fazer os casos de testes passarem com código simples.
   3. **Passo 3 do TDD**: Refatorar o código, aplicando SOLID ou boas práticas de reuso de software,
6. Hospedar a aplicação no GIT e executar os testes via Jenkins.
7. Apresentar o relatório Jacoco, destacando a cobertura do teste via Jenkins.

<<usar este template da figura 2 – o arquivo XLS está no CANVAS – link da TAREFA 3.

Uma imagem contendo captura de tela

Descrição gerada automaticamente

Figura 2: Template ATDD – Grooming Planning