**Lista de Exercícios**

1)

O time de desenvolvimento deve ser pequeno, multidisciplinar, bem estruturado e composto por profissionais qualificados com características e habilidades adequadas para organizar o próprio trabalho.

O time deve ser suficientemente pequeno para otimizar a comunicação e para que haja uma organização facilmente planejada.

2)

As equipes autogerenciáveis são aquelas que têm autonomia e responsabilidade o suficiente para cuidar das próprias entregas com excelência.

As equipes autogerenciáveis prometem dinamicidade e rapidez nas entregas graças a um alto grau de produtividade.

3)

SCRUM MASTER

Atua como um facilitador para a equipe. É responsável por certificar o andamento e o desenvolvimento do projeto. Esse papel é ocupado pela pessoa que tem mais conhecimento sobre a metodologia de trabalho.

PRODUCT OWNER

É um dos papéis principais do Scrum. É o principal responsável por direcionar o projeto, considerando as necessidades e desejos do cliente. O Product Owner é a pessoa que irá definir o cronograma, as prioridades e designar as funções para a equipe.

DEVELOPMENT TEAM

É composto por toda a equipe do projeto. O Scrum Team é o principal responsável por desenvolver o projeto, seguindo as orientações dadas pelo Product Owner citado anteriormente.

4)

PRODUCT BACKLOG

É a definição de uma lista de atividades que devem ser feitas durante o processo de desenvolvimento do produto ou serviço, normalmente ordenadas por prioridades. O Product Backlog é definido pelo Product Owner.

SPRINT BACKLOG

É uma lista de tarefas que o Development Team se compromete a fazer em um Sprint. Os itens do Sprint Backlog são extraídos do Product Backlog, pela equipe, com base nas prioridades definidas pelo Product Owner e a percepção da equipe sobre o tempo que será necessário para completar as várias funcionalidades.

5)

Na programação em pares é importante que os dois programadores possuam *skills* semelhantes. Isso evita perderem o foco da implementação, pois com outra pessoa te ajudando ou mitigando algum problema, a chance de entregar o projeto no tempo é muito maior.

Uma segunda vantagem importante é o “rodízio” de atividades, evitando assim a fadiga ou o cansaço pelo exercício repetitivo. Uma terceira vantagem muito explorada na metodologia ágil é o amadurecimento da dupla e confiança. Quanto mais tempo os pares passarem juntos, mais conhecimento (amadurecimento) e entrosamento (confiança) eles terão

6)

O SCRUM é uma forma de gestão ampla para projetos que não depende da área de conhecimento. Já o XP tem sua aplicação mais restrita, focada basicamente no mundo de desenvolvimento de sistemas de softwares. Entretanto, quando usamos o SCRUM como forma de gestão e para criação de sistemas de software, muitas das práticas contidas no XP são de grande competência, como por exemplo a criação de testes automatizados ou o uso de refatoração de código para que um trecho de código funcional seja alterado buscando um ganho de qualidade e garantindo que a manutenção futura seja simplificada.

7)

O TDD (Test-Driven Development) se baseia em pequenos ciclos de repetições, onde para cada funcionalidade do sistema um teste é criado antes. Este novo teste criado inicialmente falha, já que ainda não temos a implementação da funcionalidade em questão e, em seguida, implementamos a funcionalidade para fazer o teste passar

<https://blog.engage.bz/equipes-autogerenciaveis/>

<https://www.devmedia.com.br/agile-development-xp-e-scrum-em-uma-abordagem-comparativa/30808>

<https://www.redbelt.com.br/blog/2017/02/08/metodologia-xp/#:~:text=O%20XP%2C%20ou%20eXtreme%20Programming,para%20minimizar%20poss%C3%ADveis%20%E2%80%9Cbugs%E2%80%9D>

<https://www.devmedia.com.br/test-driven-development-tdd-simples-e-pratico/18533>