

COMPANHIA DE DESENVOLVIMENTO DOS VALES DO SÃO FRANCISCO E DO PARNAÍBA

CARGO 11: ANALISTA EM DESENVOLVIMENTO REGIONAL ÁREA: TECNOLOGIA DA INFORMAÇÃO

Prova Discursiva

Aplicação: 20/10/2024

PADRÃO DE RESPOSTA

- 1 As principais características dos métodos ágeis são:
 - Iteratividade: ao contrário de uma gestão cascata, os métodos ágeis são baseados em ciclos iterativos e incrementais de trabalho. O gerenciamento com essa abordagem é dividido em iterações menores, o que permite que partes funcionais do produto sejam entregues em intervalos regulares, em vez de esperar até o fim do projeto para entregar um produto final. Os requisitos e as necessidades do projeto podem evoluir ao longo do tempo, e, em vez de resistir a mudanças, os métodos ágeis abraçam-nas e estão preparados para se adaptar rapidamente aos novos contextos;
 - Colaboração: os métodos ágeis incentivam tanto a colaboração quanto a comunicação entre todas as pessoas da equipe, bem como com os *stakeholders* do projeto. Uma das características das equipes ágeis é trabalhar de forma interdisciplinar, promovendo a troca constante de ideias, conhecimentos e experiências;
 - Autonomia das equipes: a colaboração gera mais autonomia e empoderamento das equipes, pois as pessoas são incentivadas a se auto-organizarem, a tomar decisões e a definir o melhor caminho para atingir os objetivos do projeto; e
 - Entrega de valor: os atributos dos métodos ágeis e sua cultura de experimentação favorecem a entrega de valor, pois, ao priorizarem entregas contínuas e incrementos de valor em cada iteração, permitem que clientes e *stakeholders* obtenham benefícios tangíveis mais cedo e possam fornecer *feedbacks* para orientar o desenvolvimento do projeto.
- 2 O Scrum é uma metodologia com mecânica simples, baseada em um modelo incremental e iterativo, aliada à prática de valores ágeis. As equipes Scrum são formadas de: *product owner*, *scrum master* e membros da operação, cada um com suas responsabilidades.
 - *Product owner* (PO): compreende as necessidades dos *stakeholders* e conecta esse conhecimento com a equipe. Comanda as *sprints planning*, determina as prioridades e cria o *backlog* de produto, atua junto ao time para se certificar de que o andamento dos trabalhos corresponde ao esperado, faz ajustes de rota na *sprint*, aprova funcionalidades desenvolvidas e dialoga com os *stakeholders*.
 - *Scrum master*: guia o projeto em termos de agilidade, dá suporte e apoio a sua construção em comum com o time e eleva a prática do Scrum.
 - Membros: profissionais da execução; podem incluir desenvolvedores, *designers* UI e UX, arquitetos, entre outros profissionais.

A mecânica do Scrum centra-se no conceito de *sprint*: uma iteração limitada no tempo, normalmente entre 20 e 30 dias. Dentro de uma *sprint*, acontecem os seguintes ritos:

 - *Planning*: reunião da equipe com o objetivo de construir o *backlog* da *sprint*, sequenciar, estimar o tempo e distribuir as demandas.
 - *Daily*: reunião curta diária (15 minutos) para inspeção (o que fez, o que vai fazer e obstáculos) e ajustes de última hora.
 - *Review*: reunião ao final da *sprint* em que as funcionalidades são mostradas ao PO, a usuários e a *stakeholders* para a obtenção de *feedbacks*.
 - Retrospectiva: reunião do time ao final de cada *sprint* com o objetivo de fazer uma análise do que passou.
- 3 O Kanban, assim como o Scrum, funciona em um quadro. A forma mais básica dele tem três listas com os *kanbans* (cartões): a de tarefas por fazer, a de tarefas em andamento e a de tarefas concluídas. Cada cartão é uma atividade a ser puxada para a lista seguinte sempre que atender aos requisitos, como uma fila.
- 4 A XP (*Extreme Programming*) é uma metodologia ágil que ajuda times de desenvolvimento a fazer códigos flexíveis, que se prestam à mudança. Muitas de suas práticas são bem focadas em programação. As práticas XP são agrupadas em quatro categorias:
 - Práticas de programação: *test-first programming* ou *test-driven development* (TDD) – recomenda a escrita de testes automatizados antes da produção do código que será testado;

- Práticas de integração: *10-minute build* – recomenda-se a criação de uma *build* automatizada para o código base inteiro, que rode em menos de 10 minutos, incluindo os testes e gerando relatórios dos que passaram e falharam. O objetivo é que ela seja rodada frequentemente para revelar o nível de qualidade do código. Integração contínua: para evitar mudanças conflitantes no código, recomenda-se que o time rode a *build* constantemente e monitore erros;
- Práticas de planejamento: ciclos semanais – iterações de uma semana. *Stories* – mesmo conceito de *user stories*. Ciclos trimestrais – planejamento de longo prazo, com uma visão mais geral sobre os temas que vão direcionar o projeto, mas também um olhar para trás sobre o progresso feito e problemas internos e externos, *bugs* recorrentes e ajustes. *Slack* – tarefas ligadas a histórias com baixa prioridade deixadas para o final da iteração, para, no caso de problemas inesperados, poderem ser cortadas sem afetar as entregas; e
- Prática do time: sentar juntos – manter o acesso fácil aos colegas do time para consultas, aconselhamento e trocas. Ambiente de trabalho informativo – o espaço comunica informações sobre o projeto, por meio de quadros e *burndown charts*, por exemplo, para que todos estejam sempre na mesma página. Abraçar a mudança – assunção das mudanças como algo natural a um projeto, com a criação de mecanismos para gerenciá-la, para minimizar impactos tanto no escopo do projeto quanto na qualidade do código.

QUESITOS AVALIADOS

Quesito 2.1 Descrição de três características dos métodos ágeis

Conceito 0 – Não atendeu ao solicitado ou o fez de forma totalmente equivocada.

Conceito 1 - Descreveu corretamente apenas uma característica.

Conceito 2 - Descreveu corretamente apenas duas características.

Conceito 3 - Descreveu três características, de forma parcialmente correta.

Conceito 4 - Descreveu corretamente três características.

Quesito 2.2 Descrição da metodologia Scrum, detalhamento de dois papéis exercidos pela equipe e de dois ritos praticados na mecânica de funcionamento dessa metodologia

Conceito 0 - Não atendeu ao solicitado ou o fez de forma totalmente equivocada.

Conceito 1 - Descreveu corretamente apenas a metodologia OU detalhou corretamente apenas um papel OU apenas um rito.

Conceito 2 – Descreveu corretamente a metodologia E apenas detalhou corretamente um papel; OU descreveu corretamente a metodologia E apenas detalhou corretamente um rito; OU detalhou corretamente apenas dois papéis OU apenas dois ritos.

Conceito 3 - Descreveu corretamente a metodologia E apenas detalhou corretamente dois papéis; OU descreveu corretamente a metodologia E apenas detalhou corretamente dois ritos; OU detalhou corretamente dois papéis E apenas um rito OU vice-versa.

Conceito 4 - Descreveu corretamente a metodologia E detalhou corretamente dois papéis E apenas um rito; OU descreveu corretamente a metodologia E detalhou corretamente dois ritos e apenas um papel; OU detalhou corretamente apenas dois papéis e dois ritos.

Conceito 5 - Descreveu a metodologia E detalhou dois papéis E dois ritos, de forma parcialmente correta.

Conceito 6 - Descreveu corretamente a metodologia E dois papéis E dois ritos.

Quesito 2.3 Descrição da metodologia Kanban

Conceito 0 - Não descreveu a metodologia ou o fez de forma totalmente equivocada.

Conceito 1 - Descreveu a metodologia de forma parcialmente correta ou incompleta.

Conceito 2 – Descreveu, corretamente e de forma completa, a metodologia.

Quesito 2.4 Descrição da metodologia XP e de duas de suas práticas

Conceito 0 - Não atendeu ao solicitado ou o fez de forma totalmente equivocada.

Conceito 1 - Descreveu corretamente apenas a metodologia XP OU apenas uma de suas práticas.

Conceito 2 - Descreveu corretamente a metodologia XP E apenas uma de suas práticas; OU descreveu corretamente apenas duas práticas.

Conceito 3 - Descreveu a metodologia XP E duas de suas práticas, de forma parcialmente correta.

Conceito 4 - Descreveu corretamente a metodologia XP E duas de suas práticas.