

Atividade CI/CD e Devops

- Entrega 11 dez em 23:59
- Pontos 5
- Perguntas 7
- Disponível 3 dez em 0:00 - 11 dez em 23:59
- Limite de tempo Nenhum

Instruções

Veja os vídeos abaixo e depois responda as perguntas sobre CI/CD e pipelines de CI/CD.

<https://www.youtube.com/watch?v=zxPG-fCyVFU> ➞ <https://www.youtube.com/watch?v=zxPG-fCyVFU>



<https://www.youtube.com/watch?v=zxPG-fCyVFU>

<https://www.youtube.com/watch?v=O2TbkEhpfBE> ➞ <https://www.youtube.com/watch?v=O2TbkEhpfBE>



<https://www.youtube.com/watch?v=O2TbkEhpfBE>

<https://www.youtube.com/watch?v=pcb0UdcBCCY> ➞ <https://www.youtube.com/watch?v=pcb0UdcBCCY>



<https://www.youtube.com/watch?v=pcb0UdcBCCY>

Histórico de tentativas

	Tentativa	Tempo	Pontuação
MAIS RECENTE	Tentativa 1	6 minutos	5 de 5

❗ As respostas corretas estão ocultas.

Pontuação deste teste: 5 de 5

Enviado 11 dez em 19:56

Esta tentativa levou 6 minutos.



Pergunta 1

0,5 / 0,5 pts

Em uma pipeline de CI/CD, é correto afirmar que a principal característica de um contêiner é o (a):

- ☐ segurança de dados.
- ☒ isolamento de recursos.
- ☐ automação de testes.
- ☐ baixa tolerância a falhas.



Pergunta 2

1 / 1 pts

Uma das etapas na implementação do DevOps é o processo de *build* que envolve tarefas necessárias para se conseguir executar um programa, como, por exemplo, vinculação com bibliotecas, compilação, *download* e resolução de dependências, conforme a linguagem de programação que se estiver utilizando.

- ☐ Totalmente errado.
- ☐ Parcialmente certo
- ☐ Parcialmente errado
- ☒ Totalmente certo



Pergunta 3

1 / 1 pts

Build: É o momento em que o código-fonte é transformado em um artefato executável. Isso pode incluir a compilação de código, resolução de dependências, execução de testes automatizados e geração de pacotes prontos para deploy. Em essência, é o processo que converte seu código em algo que pode ser executado, testado e, finalmente, lançado. No Ci/CD a ferramenta Jenkins é:



Uma ferramenta poderosa integrada ao GitHub, que permite automatizar todo o ciclo de vida do desenvolvimento. Você pode configurar workflows que iniciam o build sempre que houver um novo commit, pull request, ou até mesmo em horários específicos.



Uma das ferramentas de CI/CD mais populares e amplamente utilizadas, Jenkins oferece uma abordagem flexível para configurar e automatizar builds, além de pipelines complexas.

- ☐ Ferramenta que não pode automatizar o processo de build e não tem benefícios
- ☐ Não permite fazer pipeline



Pergunta 4

0,5 / 0,5 pts

A integração contínua, entrega contínua e implantação contínua otimizam o processo de combinar o trabalho de equipes separadas em um produto coeso. A CI/CD oferece um repositório único para armazenar o trabalho e automatiza de forma constante a integração e os testes contínuos.

Um pipeline de CI/CD é uma série de etapas que otimizam o processo de entrega de software. Por meio de uma abordagem DevOps ou da Engenharia de Confiabilidade de Site, a CI/CD melhora o desenvolvimento de aplicações usando monitoramento e automação. Isso é muito útil quando se trata de integração e testes contínuos, que são tipicamente difíceis de realizar, consomem tempo e exigem a criação de stubs e drivers.

Pipelines automatizados podem ajudar a prevenir erros decorrentes de processos manuais, permitir iterações rápidas do produto e proporcionar feedback uniforme durante o processo de desenvolvimento. Cada etapa de um pipeline de CI/CD é um subconjunto de tarefas agrupadas em etapas do pipeline. São etapas de um pipeline, EXCETO:

- ☒ Documentação do projeto
- ☐ Build
- ☐ Testes automatizados
- ☐ Deploy



Pergunta 5

1 / 1 pts

As afirmativas abaixo caracterizam um bom processo de pipeline de CI/CD, EXCETO:

- ☒ Dificuldade de manutenção, acarretando mais funcionários e maior custo.
- ☐

Velocidade: a integração contínua deve ser rápida e apresentar feedback instantâneo, caso contrário, o fluxo é interrompido e a produtividade cai.

- ☐ Confiabilidade: o pipeline deve ser confiável, com código de teste à prova de falhas e resultados estáveis.
- ☐ Precisão: automatizar o processo de implantação exige precisão extrema para evitar interferência humana.



Pergunta 6

0,5 / 0,5 pts

Estabelecer um pipeline de CI/CD completo, especialmente incorporando um pipeline de implantação contínua, melhora significativamente a eficiência e a confiabilidade dos projetos de desenvolvimento de software. Ao automatizar a execução de jobs desde cada commit em um branch até a implantação, esse tipo de pipeline garante que todas as alterações sejam testadas e integradas de forma completa. Essa automação aproveita os recursos potentes das ferramentas de CI/CD para otimizar processos em todo o repositório de código.

Isso não só acelera o ciclo de desenvolvimento, mas também ajuda a manter altos padrões de qualidade de código, garantindo que cada commit contribua positivamente para os resultados do projeto. Um pipeline de integração contínua melhora a qualidade de código ao garantir que todas as alterações feitas nele passem pelo mesmo processo. As alterações de código são validadas em relação a outras alterações nas quais está sendo feito commit no mesmo repositório de código compartilhado. Compilações e testes automatizados diminuem a chance de erro humano, criando iterações mais rápidas e código de melhor qualidade.

São etapas do pipeline:

- ☐ Fonte - Compilação - Testes
- ☐ Testes - Implantação
- ☐ Fonte - Testes
- ☒ Fonte - Compilação - Testes - Implantação



Pergunta 7

0,5 / 0,5 pts

CI é uma prática recomendada de DevOps e um estágio no ciclo de vida de DevOps em que os desenvolvedores fazem check-in de código em seu repositório de código compartilhado, geralmente várias vezes ao dia. Idealmente, sempre que isso acontece, uma ferramenta de build automatizada verifica o check-in ou a ramificação para garantir que não haja erros e que esteja pronto para entrar em produção. O principal benefício é que os problemas geralmente são detectados cedo, antes que se transformem em problemas maiores.

I) Praticar a CI significa integrar pequenos subconjuntos de mudanças em um período de tempo mais curto em vez de atualizações substanciais que demoram mais e com menos frequência. Automatizar fluxos de trabalho para teste, mesclagem e check-in de alterações em um repositório compartilhado significa que as equipes podem fornecer códigos mais limpos em um ritmo mais rápido. Um código mais limpo significa validação mais rápida, versões mais limpas e um pipeline de desenvolvimento mais eficiente e mais fácil de dimensionar.

II) A implementação do DevOps varia muito entre as organizações, mas em sua essência, o DevOps não pode ser realizado sem a CI/CD. Um pipeline de CI/CD está intrinsecamente vinculado a uma cultura de DevOps e seu processo de lançamentos menores e frequentes.

Sobre as 2 frases acima:

- ☐ Apenas II está correta

- ☐ Apenas I está correta
- ☐ Nenhuma está correta
- ☒ I e II estão corretas

Pontuação do teste: 5 de 5