



Integração contínua e entrega contínua ▾

Introdução à integração contínua e à entrega contínua
Aula 1



Professor Gustavo Dias



Introdução à integração contínua de software

Você já pensou que sistemas automatizados podem fazer esse papel, otimizando a velocidade do processo?



Integração contínua ou CI (continuous integration)

- A integração contínua, frequentemente abreviada como CI (do inglês *continuous integration*), refere-se a uma prática de desenvolvimento de software na qual as alterações no **código-fonte são automaticamente testadas e integradas a um projeto existente** em uma frequência regular – em alguns casos, várias vezes ao dia.
- Essa prática é crucial para **identificar e corrigir problemas rapidamente, garantindo que o software permaneça em um estado funcional** ao longo do desenvolvimento.



Vamos entender mais sobre integração contínua?



© Getty Images

Exposição



Fonte:

<https://www.youtube.com/watch?v=nI3ljYcBGiU>



Jenkins

Relacionamento entre Git e CI

- Em um ambiente de CI, desenvolvedores frequentemente fazem *commit* de seu código em um repositório centralizado. Após cada *commit*, os sistemas de **CI compilam** e testam o projeto inteiro **automaticamente**, para garantir, assim, que a nova alteração não gere falhas.
- Isso torna o **processo de integração menos doloroso** e ajuda a **identificar problemas** previamente no ciclo de desenvolvimento.

POR QUE A CI É IMPORTANTE?

DETECÇÃO DE DEFEITOS

- **Deteccção Rápida de Defeitos:**
- Identifica e corrige bugs imediatamente após a submissão do código.

REDUÇÃO DE RISCOS

- **Redução de Riscos:**
- Minimiza a possibilidade de problemas graves no software.

QUALIDADE DO SOFTWARE

- **Melhoria da Qualidade do Software:**
- Garante um produto final mais estável e confiável.

CONFIANÇA NO SOFTWARE

- **Aumento da Confiança no Software:**
- Inspira maior confiança na capacidade do software de atender às expectativas.