

Ementa e Horários

Integração Contínua e Testes Automatizados

- 01 Introdução a Integração Contínua
- 02 Ferramentas de Integração Contínua
- 03 Configuração de Pipelines de CI
- 04 Testes Automatizados na Integração Contínua
- 05 Boas Práticas e Padrões na Integração Contínua
- 06 Instalando e configurando uma ferramenta para pipeline de Integração Contínua
- 07 Jobs e Gatilhos de execução da pipeline
- 08 Adicionando testes automatizados para execução na pipeline
- 09 Gerando relatórios de execução dos testes
- 10 Distribuição e armazenamento de arquivos de execução dos testes
- 11 Análise de resultados de execução
- 12 Configurações especiais da ferramenta
- 13 Plugins
- 14 Boas práticas no uso da ferramenta

Datas:

23 e 24 de Maio
06 e 07 de Junho

Horários:

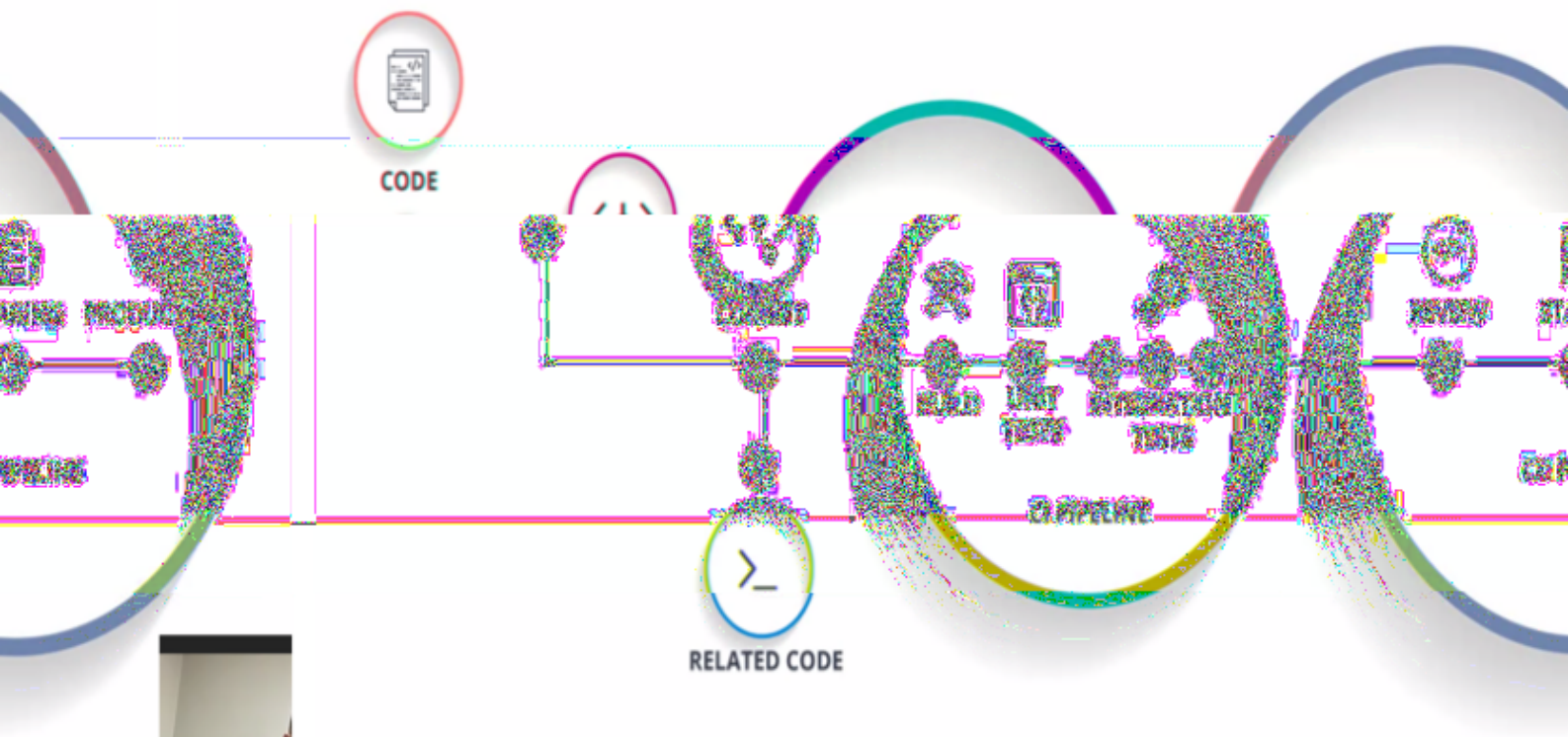
Sextas
19h00 - 20h30; 20h45 - 22h00

Sábados

09h00 - 10h30; 10h45 - 12h00;
13h30 - 15h00; 15h15 - 16h30;
16h45 - 18h00

Conceito

Introdução a Integração Contínua



3 amigos (UX/PO/QA)
Pre-Refinamento
Refinamento

Definição

API Contracts
Design System
Figma Flows

Implementação

Entrega

vProd (master/main)
vRelease
nFeatures
nStorys
nTarefas
vHotfix

- verificar se é possível executar o código (compilação)
- verificar se o código segue compatível com o código existente (código de outra pessoa ou código de produção)
- verificar se o código está padronizado (formatação e linting)
- verificar se o que estava funcionando, segue funcionando (regressão)
- verificar se o código "do jeito que está" pode ser publicado (deploy)

tarefa -> feature branch

- verificar se o código não quebrou
- verificar se os testes de unidade não quebraram

feature branch -> release

- verificar se o código não quebrou
- verificar se os testes de unidade não quebraram
- execução dos testes regressivos em x ambientes
- verificar se o percentual de cobertura é x

release -> branch principal (produção)

- ~~calcular o percentual de cobertura em todos os ambientes~~

Hotfix

- verificar se o código não quebrou
- verificar se os testes de unidade não quebraram
- verificar se o código "do jeito que está" pode ser publicado (deploy)

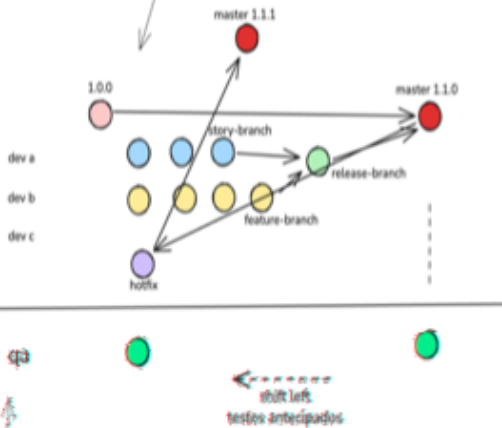
ciclos de feedback

- +rápido
- +frequente
- = melhor

500 testes de unidade - 1min

20 testes e2e - 30min

testes de mutação - 90min



Componentes da Integração Contínua

