Atividade Prática 04

Alunos: Lucas Lourenço, Gustavo Kremer, Leonardo Rorato

Jenkins

O Jenkins é uma ferramenta de automação de código aberto amplamente utilizada para integração contínua e entrega contínua (CI/CD). Com o Jenkins, as organizações podem acelerar o processo de desenvolvimento de software automatizando-o.

O Jenkins tem principal função de gerenciar toda a pipeline de entrega de software, controlar os processos de entrega em todos os ciclos do nosso software, desde a documentação até os testes e implantação.

Funcionalidades

Flexibilidade e Extensibilidade: O Jenkins destaca-se por sua flexibilidade e extensibilidade, oferecendo uma vasta gama de plugins que permitem a integração com diversas ferramentas e serviços. Essa capacidade facilita a personalização do pipeline de CI/CD conforme as necessidades específicas do projeto.

Suporte a Multiplataforma: O Jenkins é compatível com uma variedade de sistemas operacionais, permitindo sua implementação em diferentes ambientes de desenvolvimento. Isso torna a ferramenta versátil para equipes que trabalham em diversas plataformas.

Integração com Controle de Versão: A integração nativa com sistemas de controle de versão, como Git e SVN, simplifica a configuração de pipelines de CI/CD. Isso facilita o rastreamento de mudanças no código e a automação de builds.

Facilidade de Uso

Interface Gráfica: O Jenkins oferece uma interface gráfica para gerenciamento e monitoramento de Jobs e pipelines. No entanto, a interface pode parecer complexa para usuários iniciantes devido à abundância de opções.

Configuração: A configuração inicial do Jenkins pode ser desafiadora devido à variedade de opções disponíveis. No entanto, uma vez configurado corretamente, o sistema oferece eficiência e poder de personalização.

Vantagens

Flexibilidade
Jenkins oferece uma vasta gama de plugins que permitem a integração com diversas ferramentas e serviços, tornando-o altamente flexível.

Suporte Multiplataforma
Compatibilidade com diversos sistemas operacionais,

Comunidade Ativa
Jenkins possui uma comunidade ativa e muito abrangente, resultando em suporte contínuo, atualizações frequentes e uma variedade de recursos.

Integração com Controle de Versão
Integração nativa com sistemas de controle de versão, como Git, facilitando o rastreamento de mudanças no código.

Histórico e Visualização de Builds
Oferece um histórico detalhado de builds e uma interface gráfica para visualização fácil dos resultados.

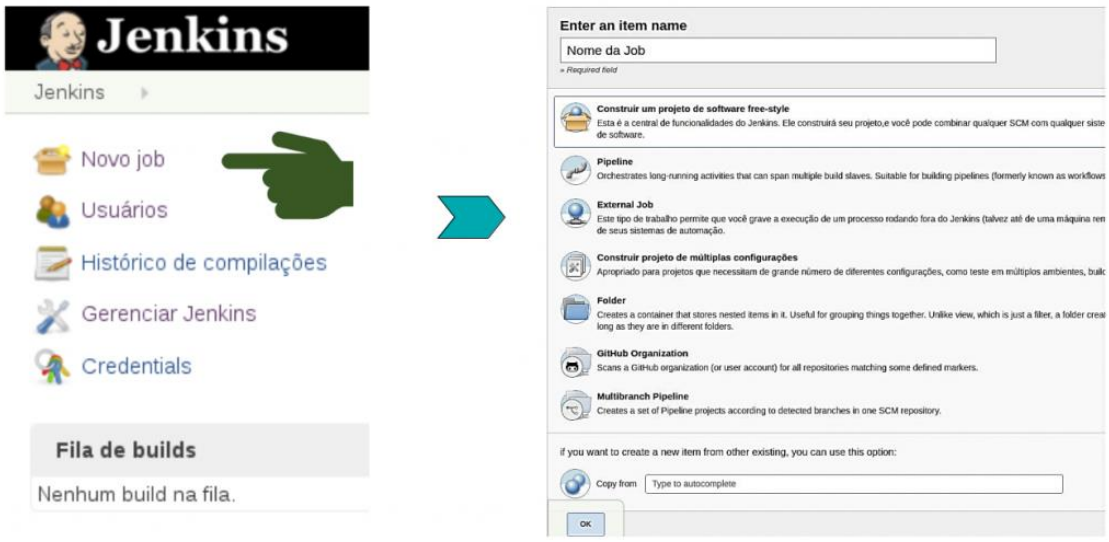
Desvantagens

Curva de Aprendizado
A configuração inicial pode ser desafiadora para usuários iniciantes, devido à variedade de opções e configurações disponíveis.

Complexidade na Configuração Inicial:
A configuração inicial pode ser complexa e requer atenção aos detalhes, especialmente para ambientes mais complexos.

Manutenção dos Plugins
A gestão e atualização de plugins podem se tornar desafiadoras à medida que a quantidade de plugins utilizados aumenta.

Tela do Jenkins



Exemplo configuração docker com uso de Jenkins

```
# Use a imagem base do Jenkins
FROM jenkins/jenkins:lts

# Instale ferramentas adicionais necessárias para os builds
USER root
RUN apt-get update && apt-get install -y \
    build-essential \
    curl \
    && rm -rf /var/lib/apt/lists/*

# Defina variáveis de ambiente, se necessário
ENV PATH="/usr/local/bin:${PATH}"

# Exemplo de instalação de uma ferramenta adicional (Node.js)
RUN curl -sL https://deb.nodesource.com/setup_14.x | bash -
RUN apt-get install -y nodejs

# Volte ao usuário Jenkins
USER jenkins
```

CircleCI

O CircleCI é uma plataforma de CI/CD na nuvem conhecida por sua facilidade de uso e configuração rápida. Esta ferramenta permite que as equipes liberem rapidamente o código em que confiam, automatizando o processo de compilação, teste e entrega para aplicativos moveis e da web, na nuvem ou em um próprio servidor privado.

Funcionalidades

Configuração Simplificada: O CircleCI destaca-se pela configuração simplificada. A plataforma adota a filosofia de "configuração por convenção", reduzindo a necessidade de extensas configurações manuais.

Integração Rápida: A integração rápida com repositórios do GitHub permite que os desenvolvedores comecem a usar o CircleCI com facilidade. Essa abordagem acelera a implementação de pipelines.

Provisionamento Rápido de Ambientes: O CircleCI oferece um provisionamento rápido de ambientes para execução de jobs, reduzindo o tempo necessário para construir, testar e implantar.

Facilidade de Uso

Interface Intuitiva: A interface do CircleCI é intuitiva, proporcionando uma experiência amigável para os usuários. A navegação simples e a organização clara dos dados contribuem para uma curva de aprendizado suave.

Configuração por Convenção: A abordagem de "configuração por convenção" do CircleCI facilita a criação de pipelines sem a necessidade de extenso conhecimento prévio da ferramenta, o que significa que, em muitos casos, não é necessária uma configuração extensa. O sistema tenta inferir configurações com base nas estruturas de diretórios e arquivos do seu projeto, facilitando a implementação de pipelines.

Pronto para Uso com Diversas Linguagens e Estruturas: O CircleCI é compatível com uma ampla variedade de linguagens de programação e estruturas de projeto. Ele oferece suporte nativo a muitas ferramentas e frameworks populares, tornando-o acessível para diferentes tipos de projetos.

Vantagens

Configuração Simples
O CircleCI utiliza arquivos de configuração YAML, simplificando a definição de pipelines e jobs.

Provisionamento Rápido de Ambientes
Oferece provisionamento rápido de ambientes para execução de builds, resultando em tempos de execução mais curtos.

Facilidade de Integração com Repositórios

integra-se rapidamente com repositórios populares como GitHub e Bitbucket, facilitando a configuração inicial.

Configuração por Convenção
Adota uma abordagem de configuração por convenção, o que significa que muitas configurações são inferidas automaticamente.

Suporte a Diversas Linguagens e Frameworks
Compatível com uma ampla variedade de linguagens de programação e frameworks, atendendo a diferentes tipos de projetos.

Integração Contínua com Controle de Versão
Oferece integração contínua, monitorando automaticamente repositórios e iniciando builds em resposta a alterações no código.

Desvantagens

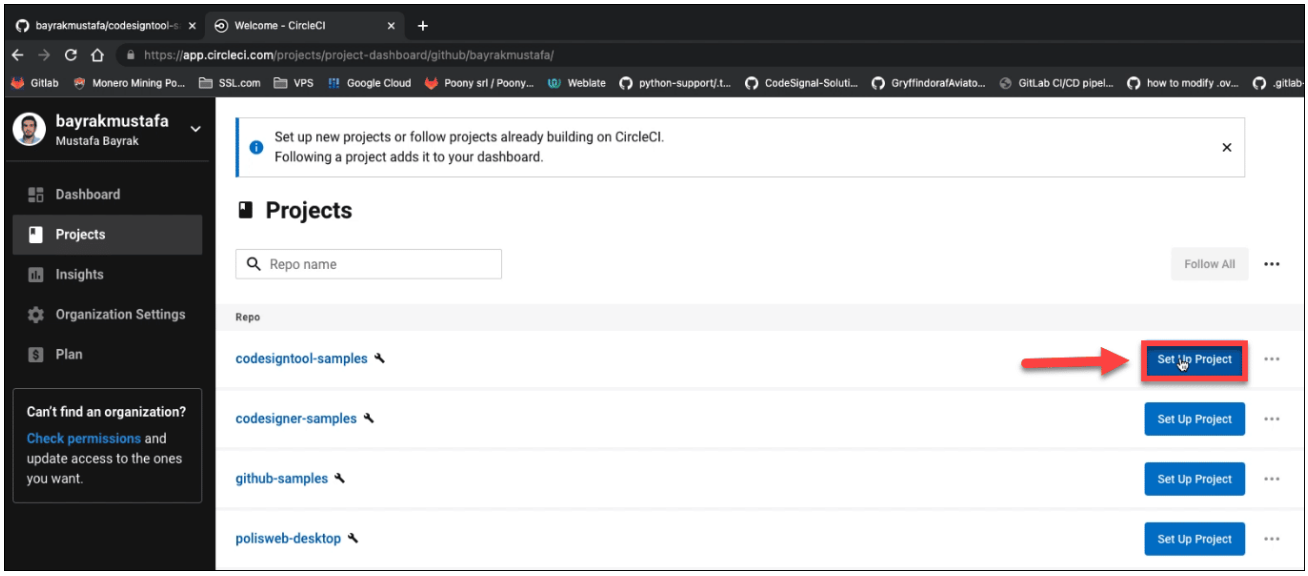
Custo em Projetos de Grande Escala
O custo pode aumentar em projetos de grande escala devido à sua estrutura de precificação.

Menos Flexibilidade Comparada a Ferramentas Especializadas
Pode ser menos flexível em comparação com ferramentas mais especializadas em determinados cenários complexos.

Necessidade de Conexão à Internet
Algumas funcionalidades podem depender de uma conexão à internet, o que pode ser uma limitação em ambientes restritos.

Limitações em Recursos Gratuitos
A versão gratuita pode ter limitações em recursos e tempo de execução para projetos mais complexos.

Tela do CircleCI



Exemplo configuração

Configuração Dockerfile

```
# Use uma imagem Node.js
FROM node:14

# Defina o diretório de trabalho
WORKDIR /app

# Copie os arquivos do projeto
COPY . .

# Instale as dependências
RUN npm install

# Comando padrão para executar os testes
CMD ["npm", "test"]
```

Configuração yaml

Bamboo

O Bamboo é uma ferramenta de integração contínua e entrega contínua (CI/CD) desenvolvida pela Atlassian. Projetado para integrar-se perfeitamente ao ecossistema Atlassian, o Bamboo oferece uma plataforma unificada que simplifica e otimiza o ciclo de vida de desenvolvimento de software.

Funcionalidades

Planos de Compilação: Permite criar planos de compilação que definem os processos de compilação, testes e implantação. Os planos de compilação podem ser configurados de acordo com as necessidades específicas do projeto.

Agentes Remotos: Utiliza agentes remotos para a execução de builds e deploys. Esses agentes podem ser distribuídos em ambientes diferentes, permitindo uma distribuição eficiente da carga de trabalho.

Execução Paralela de Jobs: A capacidade de executar jobs em paralelo acelera o processo de build e testes, melhorando a eficiência do pipeline. Essa funcionalidade é especialmente útil para projetos grandes e complexos.

Relatórios e Monitoramento
Relatórios de Build: Fornece relatórios detalhados sobre o status e os resultados das compilações e testes.
Monitoramento em Tempo Real: Oferece recursos de monitoramento em tempo real para que as equipes possam acompanhar o progresso das compilações.

Facilidade de Uso

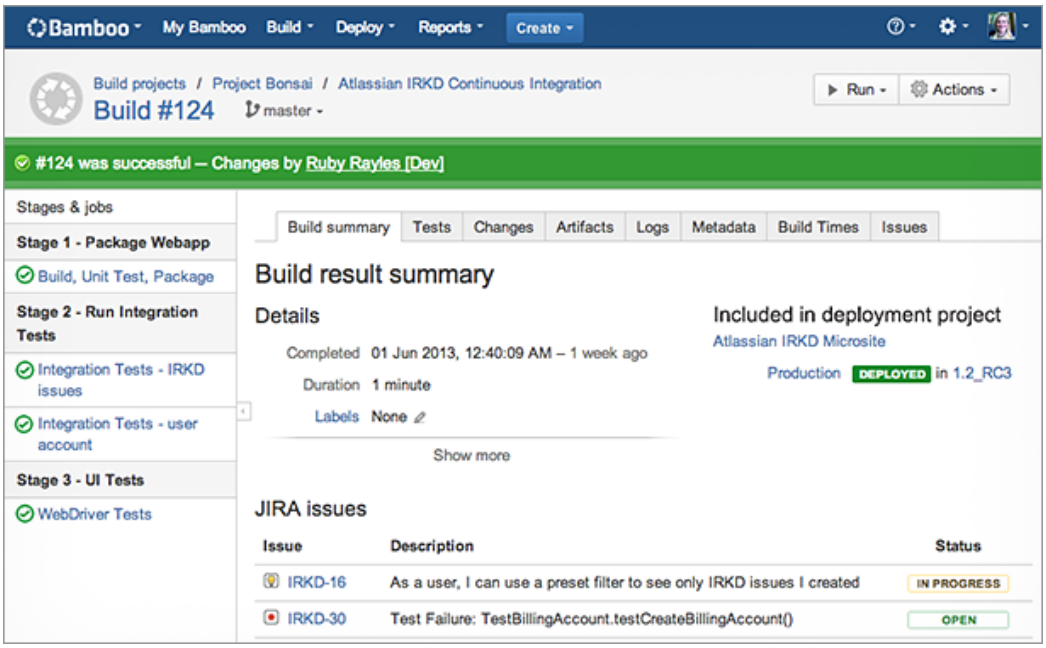
Interface Gráfica Intuitiva: A interface do Bamboo é projetada de forma intuitiva, permitindo que os usuários configurem e gerenciem ambientes, planos de construção e integrações de maneira visual.

Assistentes de Configuração: A presença de assistentes de configuração simplifica a criação inicial de projetos, planos de construção e integrações, reduzindo a curva de aprendizado.

Fácil Configuração de Repositórios: Integrar repositórios de código ao Bamboo é geralmente uma tarefa simples, com suporte para diferentes tipos de repositórios, como Git e Mercurial.

Recursos de Ajuda e Documentação: Oferece recursos de ajuda e documentação detalhada, facilitando a resolução de problemas e o entendimento das funcionalidades.

Exemplo tela Bamboo



Exemplo configuração bamboo com docker

bamboo.yaml

```yaml

image: node:14  # Imagem base para as etapas do pipeline

pipelines:
  default:
    - step:
        name: Build and Test
        script:
          - npm install
          - npm test

    - step:
        name: Build Docker Image
        services:
          - docker
        script:
          - docker build -t my-node-app .

    - step:
        name: Push to Docker Registry
        services:
          - docker
        script:
          - echo "$DOCKER_PASSWORD" | docker login -u "$DOCKER_USERNAME" --password-stdin
          - docker tag my-node-app:latest my-docker-registry/my-node-app:latest
          - docker push my-docker-registry/my-node-app:latest

    - step:
        name: Deploy to Production
        deployment: production
        trigger: manual
        script:
          - kubectl apply -f kube/deployment.yaml

```

Plano de implementação

Jenkins

Inicie com uma avaliação das necessidades do projeto, especialmente se flexibilidade e extensibilidade são fundamentais.

Escolha plugins de acordo com as integrações desejadas e personalizações necessárias.

Ofereça treinamento focado na configuração inicial e no uso básico da interface gráfica.

Quando usar o Jenkins

O Jenkins é adequado quando a flexibilidade é crucial, e há a necessidade de integração com uma variedade de ferramentas e serviços.

Considere a adoção do Jenkins no início de projetos complexos ou que exigem personalizações extensivas na pipeline de CI/CD.

CircleCI

Realize uma análise detalhada da estrutura do projeto para garantir que a abordagem de "configuração por convenção" do CircleCI seja adequada.

Identifique áreas onde a simplicidade na configuração é uma prioridade.

Configure rapidamente a integração do CircleCI com os repositórios do GitHub ou Bitbucket.

Utilize as integrações nativas dessas plataformas para simplificar o processo de implementação.

Promova sessões de treinamento focadas na configuração por convenção, destacando a filosofia de mínimo esforço.

Incentive a equipe a explorar e compreender a relação direta entre a estrutura do projeto e a configuração automática pelo CircleCI.

Quando usar o CircleCI

O CircleCI é a escolha ideal quando a simplicidade na configuração é crucial e o projeto utiliza GitHub ou Bitbucket.

Considere a adoção do CircleCI em projetos que exigem configuração rápida e que podem se beneficiar da abordagem de "configuração por convenção".

Bamboo

Mapeie os processos de compilação, testes e implantação para criar planos personalizados no Bamboo.

Planeje a distribuição eficiente de agentes remotos, considerando a carga de trabalho prevista.

Treine a equipe na interface gráfica intuitiva do Bamboo e nos assistentes de configuração.

Quando usar o Bamboo

O Bamboo é recomendado quando a integração perfeita com o ecossistema Atlassian é uma prioridade.

Considere a adoção do Bamboo em projetos que fazem uso extensivo de outras ferramentas da Atlassian e quando a interface gráfica intuitiva é valorizada.

Se a equipe valoriza uma interface gráfica intuitiva, o Bamboo oferece uma experiência de usuário amigável. Sua interface visual simplifica a configuração e o gerenciamento de pipelines, sendo uma escolha sólida para equipes que preferem uma abordagem mais visual.

Projetos que Demandam Relatórios Detalhados e Monitoramento em Tempo Real

Considerações

Recomenda-se uma abordagem de progressão gradual na implementação das ferramentas de CI/CD. Iniciar com a ferramenta mais apropriada ao contexto atual da equipe e do projeto permite uma transição suave, evitando sobrecarga e facilitando a adaptação progressiva.

A avaliação contínua é essencial para medir o progresso e a satisfação da equipe durante a adoção das ferramentas de CI/CD. Periodicamente, reveja a eficácia das ferramentas implementadas, coletando feedback e ajustando a estratégia conforme necessário. Isso garante que as escolhas feitas estejam alinhadas às necessidades em constante evolução.

A manutenção de treinamentos contínuos é crucial para o sucesso da equipe no uso das ferramentas escolhidas. Garanta que a equipe esteja familiarizada e confortável com as ferramentas por meio de sessões de treinamento regulares. Isso promove a atualização constante das habilidades e conhecimentos, permitindo que a equipe aproveite ao máximo as funcionalidades oferecidas.