Documento do Pipeline de CI/CD - Easy Queue

**Visão Geral**

Este documento apresenta a simulação do pipeline de CI/CD (Integração Contínua e Deploy Contínuo), demonstrando as práticas de DevOps aplicadas a um fluxo de desenvolvimento de software moderno.  
  
O objetivo é garantir que, o código possa ser continuamente testado e implantado com o mínimo de intervenção manual, aumentando a confiabilidade e a eficiência dos processos de entrega.

**Ferramentas Utilizadas (Simuladas)**

| **Função** | **Ferramenta** | **Justificativa** |
| --- | --- | --- |
| Repositório remoto | GitHub | Plataforma utilizada no projeto; permite controle de versões e colaboração |
| CI/CD | GitHub Actions | Ferramenta nativa do GitHub e de fácil integração |
| Testes automatizados | Jest | Framework popular para testes automatizados em projetos Node.js |
| Deploy | Railway (simulado) | Permite integração com Git e realiza deploy automático em ambientes online |

**Descrição do Pipeline de CI/CD**

1. Disparo automático  
   Sempre que um \*push\* é realizado na branch `main`, o pipeline é automaticamente iniciado.
2. Instalação de dependências  
   Um ambiente virtual é configurado, onde as dependências do projeto são instaladas via npm install.
3. Execução de testes automatizados  
   Com o ambiente preparado, os testes definidos via Jest são executados automaticamente. Caso algum falhe, o processo é interrompido.
4. Deploy simulado  
   Com os testes aprovados, seria realizado o deploy para o ambiente de staging da Railway.
5. Notificação  
   Opcionalmente, poderíamos configurar notificações via e-mail ou Slack para informar a equipe sobre o status do pipeline.

**Arquivo YAML Simulado do Pipeline**

# .github/workflows/ci-cd.yml  
name: CI/CD Pipeline - Pizzaria  
  
on:  
 push:  
 branches:  
 - main  
  
jobs:  
 build-and-deploy:  
 runs-on: ubuntu-latest  
  
 steps:  
 - name: Checkout do repositório  
 uses: actions/checkout@v3  
  
 - name: Setup do Node.js  
 uses: actions/setup-node@v3  
 with:  
 node-version: '18'  
  
 - name: Instalar dependências  
 run: npm install  
  
 - name: Executar testes  
 run: npm test  
  
 - name: Deploy (simulado)  
 run: echo "Deploy para ambiente de staging realizado com sucesso!"

**Considerações Finais**

O pipeline apresentado demonstra claramente as etapas essenciais de um processo de integração e entrega contínua, e como essas etapas se aplicariam ao desenvolvimento do EasyQueue em um cenário real de desenvolvimento ágil com DevOps.