

Trabalho Prático - Etapa 3:
Continuous Integration/Continuous Delivery-Deployment (CI/CD)
OU
Análise de Qualidade de Código

1. Especificação

Esta seção apresenta as características fundamentais do **Trabalho Prático - Etapa 3** relativas à sua especificação.

1.1 Informações gerais

Na **Etapa 3** do trabalho, os grupos poderão escolher **uma das opções** abaixo:

1.1.1. Implementação de um pipeline de Continuous Integration/Continuous Delivery-Deployment (CI/CD)

No resumo sobre CI/CD, vocês já estudaram os conceitos e se familiarizaram. Agora é hora de implementar um pipeline de integração contínua e entrega e implantação contínua da aplicação desenvolvida pelo grupo. Vocês podem utilizar uma instalação de Jenkins ou GitHub Actions, se usar o GitHub Actions, não esqueça de fazer a conta de estudante e cuidado com os limites de armazenamento e tempo de uso.

1.1.2. Análise de Qualidade de Código

Uma ferramenta de análise de qualidade de código é um aplicativo de software que examina automaticamente o código-fonte para identificar potenciais defeitos, inconsistências e outros problemas que podem impactar negativamente a qualidade, confiabilidade e manutenibilidade do software. Essas ferramentas podem ajudar os desenvolvedores a melhorar suas práticas de codificação, detectar erros no início do processo de desenvolvimento e, finalmente, produzir software de maior qualidade.

O SonarQube é uma ferramenta bastante utilizada por empresas de tecnologia e possui uma versão Community que é free e open source:

<https://www.sonarsource.com/open-source-editions/sonarqube-community-edition/>

A ideia é fazer uma instalação de SonarQube Community edition e rodar para o código do trabalho de vocês.

2. Apresentação

O grupo deve preparar um conjunto de slides para apresentar as partes principais do trabalho desenvolvido para o professor e os colegas. O tempo de apresentação deve ser de no máximo 20 minutos. A apresentação deve mostrar obrigatoriamente:

2.1. Implementação de um pipeline de Continuous Integration/Continuous Delivery-Deployment (CI/CD)

- Qual ferramenta de CI/CD foi utilizada Jenkins ou GitHub Actions?
- Como foi realizada a instalação/configuração?
- Demonstrar uma integração de código que faça falhar os testes e parar o pipeline
- Demonstrar uma integração de código que faça passar os testes e concluir o pipeline
- Quais as dificuldades encontradas?

2.2. Análise de Qualidade de Código

- Como foi realizada a instalação/configuração do SonarQube Community?
- Demonstrar a execução da análise de código
- Apresentar pelo menos o Test Coverage e o Issues Report
- Algum outro report causou surpresa ao time? Qual e por quê?
- Quais as dificuldades encontradas?

A ordem de aplicação será definida por sorteio no início da aula, onde todos os alunos devem estar presentes. É importante prestar atenção no tempo limite e dividir a apresentação previamente entre os membros do grupo para que todos consigam participar da apresentação.

Como são ferramentas que talvez vocês tenham dificuldade de configurar no laboratório, caso necessário, as demonstrações podem ser gravações de vídeo de tela e prints para ajudar na explanação. Algumas ferramentas que podem ajudar: Loom (versão free grava 5 minutos de vídeo), OBS... (vocês devem conhecer outras).

3. Entrega

Este trabalho (Etapa 3) deve ser entregue exclusivamente via commit no GitHub do grupo conforme data combinada pelo professor, como um único arquivo no formato **pdf da apresentação** com nome **"INF01120_grupoXX_Etapa3.pdf"**, onde **XX** é o **número do grupo** correspondente descrito na planilha da disciplina. **Deverá ser entregue apenas um arquivo por grupo.** Não esqueçam de adicionar a Professora no repositório. Como entrega será considerada a última versão "commitada" do arquivo da data combinada (localizada através do histórico do arquivo no GitHub). Versões "commitadas" após a data de entrega serão desconsideradas.

Bom trabalho!