



Instituto Superior Politécnico de Tecnologias e Ciências
(ISPTEC)

Departamento de Engenharia e Tecnologias

Engenharia de Software

Licenciatura em Engenharia Informática

2022/2023

Prática Laboratorial #05

Judson Paiva
judson.paiva@isptec.co.ao

INTRODUÇÃO

O Git é uma ferramenta poderosa de controle de versão que oferece várias operações básicas que podem ser feitas com ele. Algumas dessas operações básicas incluem:

1. **Clonar um repositório:** para começar a trabalhar com um repositório Git, você pode cloná-lo para o seu computador usando o comando `git clone <url_do_repositório>`.
2. **Adicionar arquivos ao staging:** para incluir arquivos no próximo commit, você precisa adicioná-los ao staging. Você pode fazer isso usando o comando `git add <nome_do_arquivo>` ou `git add .` para adicionar todos os arquivos alterados.
3. **Realizar commit:** para salvar as alterações no repositório, você precisa realizar um commit. Use o comando `git commit -m "mensagem_do_commit"` para realizar um commit com uma mensagem explicativa sobre as alterações feitas.
4. **Verificar o status:** você pode verificar o status do seu repositório local usando o comando `git status`. Ele mostra quais arquivos foram alterados, adicionados ou removidos e quais estão no staging.
5. **Atualizar o repositório local:** para buscar as últimas alterações do repositório remoto para o seu repositório local, você pode usar o comando `git fetch`.
6. **Enviar alterações para o repositório remoto:** para enviar suas alterações locais para o repositório remoto, use o comando `git push`.
7. **Criar e trabalhar com branches:** você pode criar uma nova branch usando o comando `git branch <nome_da_branch>` e mudar para ela usando o comando `git checkout <nome_da_branch>`. Você pode

trabalhar em uma branch separada sem afetar o código-fonte principal.

8. **Mesclar branches:** você pode mesclar as alterações feitas em uma branch com outra usando o comando `git merge <nome_da_branch>`. Isso permite que você incorpore alterações de uma branch em outra.

Essas são algumas das operações básicas do Git. Há muitas outras, mas essas devem ser suficientes para começar.

OBJECTIVOS DA PRÁTICA LABORATORIAL

1. Desenvolver habilidades de gestão de projetos.
2. Trabalhar em equipe e colaborar.
3. Desenvolver habilidades tecnológicas.
4. Promover a aprendizagem ativa e o engajamento.
5. Desenvolver habilidades de pensamento crítico.
6. Promover a aprendizagem autônoma.

ATIVIDADES

Antes de começar este laboratório, é necessário ter o Git instalado no computador. Além disso, é recomendável ter algum conhecimento básico de programação e de linha de comando.

1. Configurar o Git

Antes de começar a usar o Git, é necessário configurá-lo com o seu nome de usuário e endereço de e-mail. Use os seguintes comandos no terminal para fazer isso:

```
git config --global user.name "Seu nome"
```

```
git config --global user.email "seu-email@exemplo.com"
```

2. Criar um repositório Git

Para criar um novo repositório Git, crie uma nova pasta no seu computador e, em seguida, abra a mesma pasta usando linhas de comando ou terminal e execute o seguinte comando dentro da pasta:

```
git init
```

3. Adicionar arquivos ao repositório

Agora que o repositório foi criado, você pode adicionar arquivos ao repositório usando o seguinte comando:

```
git add <nome_do_arquivo>
```

Ou para adicionar todos os arquivos da pasta atual ao repositório, use o seguinte comando:

```
git add .
```

4. Realizar um commit

Depois de adicionar os arquivos ao repositório, você pode realizar um commit para salvar as alterações no repositório. Use o seguinte comando para realizar um commit:

```
git commit -m "Mensagem do commit"
```

5. Verificar o status do repositório

A qualquer momento, você pode verificar o status do repositório usando o seguinte comando:

```
git status
```

Este comando mostra quais arquivos foram modificados, adicionados ou removidos.

6. Trabalhar com branches

Uma das principais vantagens do Git é a capacidade de trabalhar com branches. Use o seguinte comando para criar uma nova branch:

```
git branch <nome_da_branch>
```

Para mudar para uma branch existente, use o seguinte comando:

```
git checkout <nome_da_branch>
```

Você também pode criar e mudar para uma nova branch usando o seguinte comando:

```
git checkout -b <nome_da_branch>
```

7. Mesclar branches

Depois de trabalhar em uma branch, você pode mesclar as alterações de uma branch com outra usando o seguinte comando:

```
git merge <nome_da_branch>
```

Isso permite que você incorpore as alterações feitas em uma branch em outra.

8. Trabalhar com um repositório remoto

O Git também permite trabalhar com um repositório remoto. Para fazer isso, você precisa clonar o repositório remoto para o seu computador usando o seguinte comando:

```
git clone <url_do_repositório>
```

Depois de clonar o repositório, pode enviar as suas alterações para o repositório remoto usando o seguinte comando:

```
git push
```

Também pode obter as alterações mais recentes do repositório remoto usando o seguinte comando:

```
git pull
```

9. Trabalhar Tags

O que são tags? Porque usamos tags? Como associar em nosso repositório tags?

Para a aula de hoje deverão aplicar todos os 9 conceitos acima em Projecto que criarão com os elementos de vosso grupo de trabalho, e farão um relatório com prints de cada comando e o resultado que obtiveram.

Neste relatório deverá conter em detalhes, conceitos relacionados as tarefas básicas:

- Iniciar um repositório Git em um diretório existente;
- Clonar um repositório Git para o seu computador;
- Adicionar arquivos ao seu repositório local;
- Confirmar mudanças no seu repositório local;
- Verificar o histórico de confirmações;

- Criar e trabalhar com branches;
- Integração de alterações entre branches;
- Trabalhar com repositórios remotos, incluindo clonagem, envio e recebimento de alterações;
- Resolver conflitos de mesclagem em alterações conflitantes;
- Desfazer mudanças em um arquivo específico ou em todo o branch.

Nota: Este laboratório foi projetado para ensinar os conceitos básicos de controle de versões com o Git. No entanto, há muitos outros recursos e comandos disponíveis no Git, e recomenda-se explorar a documentação do Git para aprender mais sobre como usá-lo de forma mais avançada.

Deverão partilhar os vossos perfis do GitHub com todos os vossos colegas do curso de Engenharia Informática. Têm até ao dia 14 as 23h59 para enviarem o relatório do Laboratório.