

## 1. O que é Git?

Git é um tipo de sistema de controle de versão que torna mais fácil gerir alterações nos ficheiros. O Git guarda um registo do que foi feito num projeto e permite ao utilizador reverter alterações para versões anteriores se for necessário. Facilita a colaboração dentro de equipas pois permite a diferentes utilizadores fazer alterações a um único projeto.

## 2. Existem outros sistemas de controle de versão?

Existem três tipos de sistemas de controle de versão: Local, Centralizado e Distribuído. O local mantém o registo de alterações de ficheiros num sistema local. O centralizado possui um servidor central que contém toda a informação dos ficheiros e uma lista dos clientes que têm acesso aos mesmos (Ex: Tortoise SVN). O distribuído permite aos clientes clonar o repositório incluindo todos os seus registos. Permite assim que, caso o servidor falhe, um cliente possa copiar o repositório de novo para o servidor e assim restaurá-lo (Ex: Git).

## 3. Quais as vantagens de se utilizar um sistema de controle de versão como o Git?

Rapidez, autonomia, facilidade de partilha e fusão de projetos, confiabilidade e redução de custos com o servidor.

## 4. Qual o comando para indicar que uma pasta deve ser gerenciado pelo Git?

Git init

## 5. Ao adicionar/editar arquivos dentro de uma pasta gerenciada pelo Git, quais os comandos necessários para adicionar estas mudanças, de fato, ao Git?

Primeiro `git add <nome_ficheiro>` ou `.` (adiciona todos os ficheiros da diretoria) e depois `git commit -m "Informação sobre as alterações"`

## 6. Qual a função do comando "checkout"?

Permite ao utilizador navegar entre os branches criados no repositório através do comando `git branch`. Fazer checkout a um branch atualiza os ficheiros para a versão guardada nesse respetivo branch e faz com que o Git passe a novas alterações nesse branch.

## 7. Pesquise e explique o que é uma "branch"?

Uma branch corresponde a uma linha independente de desenvolvimento no repositório. Permite criar diferentes ramificações que posteriormente podem ser usadas para reverter o projeto atual para o estado numa outra ramificação.

## 8. Diversas plataformas existem no mercado para suportar o uso de Git via Web. Procure algumas destas plataformas. Procure informações como preço, número de pessoas, se oferece o serviço gratuitamente, se possui repositórios privados. Obs: No mínimo 3 plataformas.

- AWS CodeCommit
  - Grátis para os cinco primeiros usuários ativos com repositórios ilimitados, 50gb de armazenamento por mês e 10000 solicitações git por mês. Cada usuário adicional tem um custo de 1\$ por mês com repositórios ilimitados, 10gb de armazenamento por mês por usuário ativo e 2000 solicitações git por mês por usuário ativo.
  - Permite repositórios privados.

- GitHub
  - Grátis com repositórios públicos e privados ilimitados e numero ilimitado de colaboradores. Possui planos pagos com maior espaço de armazenamento e outras funcionalidades como revisão de código.
- Microsoft Azure DevOps
  - Possui planos individuais e coletivos. Nos planos coletivos existe numero ilimitado de repositórios privados. O plano básico e grátis para os 5 primeiros utilizadores e depois custa 6\$ por utilizador adicional. Oferece 2GB de armazenamento.