**Parte Prática**

* Cria um repositório local em uma pasta chamada “REP”
* Adiciona um ficheiro de texto e escreve um conteúdo qualquer.
* Adiciona esse ficheiro ao teu repositório local e posteriormente ao remoto. Descreve todos os passos que tiveste de realizar para realizar ho pretendido.
* Pede a outro grupo para fazer cópia do teu repositório remoto, alterar o teu ficheiro e realizar um pedido de pull request. Caso dê conflito efetua os passos necessários para resolver.

**Parte Teórica**

* Explica o que entendes por ***software* de controlo de versões**.
* Um software controlo de versões é um sistema onde podemos trabalhar em conjunto e ao mesmo tempo guardar todas as alterações(backup) no site.Também podemos trabalhar em paralelo sem interferir no trabalho dos outros colegas sdasdasd
* Indica **três vantagens** da utilização de Sistemas de Controlo de Versões.
* Três vantagens da utilização do controlo de versões são:

- Trabalhar no projeto sem alterar no projeto principal, assim não há funcionalidades que fiquem mal com as minhas alterações;

-Ter vários backups no caso de algo correr mal podemos sempre voltar atrás

-Trabalhar em simultâneo sem atrapalhar o trabalho dos outros e no final juntando todas as partes

* Considera as imagens seguintes:

 

* De que tipo de Sistemas de controlos se trata?
* **Explica a diferença dos seus procedimentos.**

**- Na primeira imagem estamos perante um sistema centralizado onde tem um repositório remoto na qual nós fazemos as alterações, ou seja todos trabalham no mesmo repositório, enquanto na segunda imagem estamos perante um sistema distribuído onde cada um fica com uma cópia do projeto e trabalham no seu repositório local, e depois os repositórios locais de cada juntam-se no repositório remoto.**

* Quando estamos perante um sistema de versão de controlo, surgem os termos***branch****,* ***merge***e***commit****.* Explica e associa cada um dos termos anteriores, utilizando a imagem seguinte.



Um branch é uma ramificação do trabalho, existe o MASTER que é a ramificação principal e podemos ter vários branches fora do master onde trabalhamos , e após estar tudo feito fazemos merge que significa juntar as branchs que eu tenho à master , commit significa fazer uma cópia para o repositório remoto.

* Os repositórios locais consistem em três "árvores" mantidas pelo git.

O working dir é onde temos os trabalhos com alterações que depois adicionamos ao index, fazendo um commit para adicionar ao head, (repositório remoto)



* Explica a imagem anterior, descrevendo o funcionamento dos repositórios locais.

**Os repositórios locais consistem em três "árvores" mantidas pelo git. a primeira**

**delas é a Working Directory que contém os ficheiros vigentes; a segunda Index**

**(Staging Area) que funciona como uma área temporária e finalmente**

**a HEAD que aponta para o último commit (confirmação) que foi feito.**

* Descreve os passos necessários para criar um repositório local através do **GIT CMD**. Para além de criar um repositório imagina que ainda queres criar uma nova funcionalidade para o teu projeto e adicionar ao projeto inicial.

Criar um pasta : Mkdir “nome”

Inicializar o repositório : git init (dentro da pasta).