



INSTITUTO SUPERIOR POLITÉCNICO DE TECNOLOGIAS E CIÊNCIAS

DEPARTAMENTO DE ENGENHARIAS E TECNOLOGIAS

ENGENHARIA DE SOFTWARE II

RELATÓRIO

GIT E GITHUB

ESTUDANTE	
ALBERTO PEDRO MULONDE – 20201030 ANTÓNIO ANDRÉ KUDIEZO – 20200630 ÉDNER LEOCÍNIO MATEUS – 20201775 EZEDÉLIO MACUETE GARCIA – 20200306	
CURSO: Engenharia Informática TURMA: EINF6-M1	DOCENTE: JUDSON PAIVA

I – INTRODUÇÃO

Este relatório foi elaborado em grupo e tem como objetivo fornecer uma visão detalhada das tarefas básicas relacionadas ao uso do Git, um sistema de controle de versão amplamente utilizado no desenvolvimento de software. O Git permite que equipes de desenvolvimento colaborem de maneira eficiente e mantenham um histórico completo das alterações feitas em um projeto.

Por outro lado, o GitHub é uma plataforma baseada em nuvem que oferece hospedagem para repositórios Git. Ele fornece uma interface web amigável para trabalhar com repositórios Git e uma variedade de recursos adicionais que ajudam no desenvolvimento colaborativo de projetos de software. O GitHub permite que desenvolvedores compartilhem e colaborem em seus projetos, além de fornecer ferramentas para rastrear problemas, solicitar e revisar alterações de código, bem como para implantar aplicativos.

II – INICIAR UM REPOSITÓRIO GIT EM UM DIRETÓRIO EXISTENTE

Iniciar um repositório Git é o primeiro passo para controlar versões de um projeto. Para iniciar um repositório Git em um diretório existente, utilizamos o comando "git init". Isso criará um novo repositório Git no diretório escolhido, permitindo que o Git rastreie e registre todas as alterações feitas nos arquivos.

```
C:\Users\Lil Miao\Documents\Git\lab-cinco>git init
Initialized empty Git repository in C:/Users/Lil Miao/Documents/Git/lab-cinco/.git/

C:\Users\Lil Miao\Documents\Git\lab-cinco>git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    estiloIndex.css
    index.html

nothing added to commit but untracked files present (use "git add" to track)
```

III – CLONAR REPOSITÓRIO GIT PARA O SEU COMPUTADOR

Clonar um repositório Git é o processo de obter uma cópia completa de um repositório remoto para o seu computador local. Utilizamos o comando "git clone" seguido do URL do repositório remoto. Isso cria uma cópia local do repositório, incluindo todo o histórico de commits e todas as branches existentes.

IV – ADICIONAR ARQUIVOS AO SEU REPOSITÓRIO LOCAL

Após iniciar um repositório ou clonar um repositório existente, é possível adicionar arquivos ao repositório local. Utilizamos o comando "git add" seguido do nome dos arquivos ou diretórios que desejamos adicionar. Isso prepara os arquivos para serem incluídos no próximo commit.

```
C:\Users\Lil Miao\Documents\Git\lab-cinco>git add .

C:\Users\Lil Miao\Documents\Git\lab-cinco>git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   estiloIndex.css
        new file:   index.html

C:\Users\Lil Miao\Documents\Git\lab-cinco>git commit -m "Estado inicial dos ficheiros"
[main (root-commit) 2a6e83c] Estado inicial dos ficheiros
 2 files changed, 59 insertions(+)
 create mode 100644 estiloIndex.css
 create mode 100644 index.html

C:\Users\Lil Miao\Documents\Git\lab-cinco>git branch -M main

C:\Users\Lil Miao\Documents\Git\lab-cinco>git remote add origin https://github.com/albpedmul/lab-cinco.git
```

V – ADICIONAR ARQUIVOS AO SEU REPOSITÓRIO LOCAL e CONFIRMAR MUDANÇAS NO SEU REPOSITÓRIO LOCAL

Após iniciar um repositório ou clonar um repositório existente, é possível adicionar arquivos ao repositório local. Utilizamos o comando "git add" seguido do nome dos arquivos ou diretórios que desejamos adicionar. Isso prepara os arquivos para serem incluídos no próximo commit.

Para registrar as mudanças feitas nos arquivos adicionados, utilizamos o comando "git commit". Isso cria uma nova confirmação (commit) no histórico do repositório, incluindo uma mensagem que descreve as alterações realizadas. O commit é uma etapa importante para manter um registro claro e organizado das alterações feitas no projeto.

```
C:\Users\Lil Miao\Documents\Git\lab-cinco>git add .

C:\Users\Lil Miao\Documents\Git\lab-cinco>git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   estiloIndex.css
        new file:   index.html

C:\Users\Lil Miao\Documents\Git\lab-cinco>git commit -m "Estado inicial dos ficheiros"
[main (root-commit) 2a6e83c] Estado inicial dos ficheiros
 2 files changed, 59 insertions(+)
 create mode 100644 estiloIndex.css
 create mode 100644 index.html

C:\Users\Lil Miao\Documents\Git\lab-cinco>git branch -M main

C:\Users\Lil Miao\Documents\Git\lab-cinco>git remote add origin https://github.com/albpedmul/lab-cinco.git
```

VI – VERIFICAR O HISTÓRICO DE CONFIRMAÇÕES

Para visualizar o histórico de commits em um repositório, utilizamos o comando "git log". Isso exibe uma lista detalhada de todas as confirmações realizadas, incluindo o autor, data e mensagem associados a cada commit. Essa funcionalidade é útil para rastrear o progresso do projeto e revisar as alterações feitas.

```
C:\Users\Lil Miao\Documents\cursogit\projeto1>git log
commit 864e4d1f64edf9254c2c64547814168cb23a3345 (HEAD -> login, origin/main, main)
Author: albpedmul <albertopedromulonde3@gmail.com>
Date:   Fri May 19 05:09:26 2023 +0100

    Finalizado sistema de login

commit 3592e0746e1bbc8ab48371333f2791549d7c001b (page_home)
Author: albpedmul <albertopedromulonde3@gmail.com>
Date:   Fri May 19 05:04:06 2023 +0100

    Finalizado segunda parte da home

commit 7d1888b79da94ecd9aed6d6b69c609bb449aa1bd
Author: albpedmul <albertopedromulonde3@gmail.com>
Date:   Fri May 19 04:55:31 2023 +0100

    Finalizando page home

commit 505f1393a704c566b35365c73006bb1d62dca5e9
Author: albpedmul <albertopedromulonde3@gmail.com>
Date:   Fri May 19 04:31:46 2023 +0100

    Criando descricao da pagina home

commit 39a55cfd3603a71aba40274046b84519ab5ba24a (page_contato)
Author: albpedmul <albertopedromulonde3@gmail.com>
Date:   Fri May 19 04:03:57 2023 +0100

    Adicionando gitignore ao projeto

commit ced634f085d35c4711882bdf00e860cc147c149d
Author: Alberto Pedro Mulonde <119819228+albpedmul@users.noreply.github.com>
Date:   Fri May 19 00:37:30 2023 +0100
```

VII – CRIAR E TRABALHAR COM BRANCHES

Branches são ramificações independentes do desenvolvimento de um projeto. Podemos criar novos branches para trabalhar em funcionalidades ou correções de bugs sem interferir no branch principal. Utilizamos o comando "git branch" para criar um novo branch e "git checkout" para alternar entre branches existentes.

```
C:\Users\Lil Miao\Documents\cursogit\projeto1>git branch
* login
  main
  page_contato
  page_home
```

VIII – CRIAR E TRABALHAR COM BRANCHES

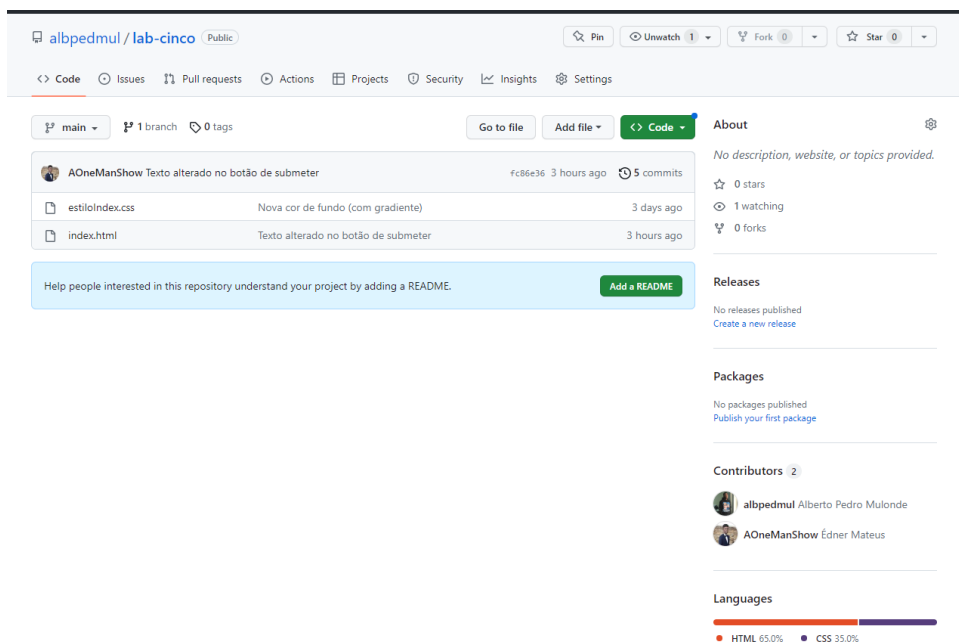
Para integrar alterações de um branch para outro, utilizamos o comando "git merge". Isso mescla as alterações feitas em um branch com outro, permitindo que as modificações sejam combinadas. O Git tentará mesclar automaticamente as alterações, mas em alguns casos podem ocorrer conflitos de mesclagem.

```
C:\Users\Lil Miao\Documents\cursogit\projeto1>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

C:\Users\Lil Miao\Documents\cursogit\projeto1>git merge page_home
Updating 7d1888b..3592e07
Fast-forward
 home.js      | 3 +++
 index.html   | 3 +++
 2 files changed, 6 insertions(+)
 create mode 100644 home.js
```

IX – TRABALHAR COM REPOSITÓRIOS REMOTOS

O Git possibilita o trabalho com repositórios remotos, que são versões do projeto hospedadas em servidores.



INSTITUTO SUPERIOR POLITÉCNICO DE TECNOLOGIAS E CIÊNCIAS

AV. Luanda Sul, Rua Lateral Via S10, Talatona – Município do Belas – Luanda/Angola
Telefones: +244226430334/44226430330 – Email: geral@isptec.co.ao