

Fase 4 - Activitat 6.1: Iniciació a Git i Github

0- Identificació del grup i activitat:

Curs: ASIX2

Projecte: PJ9 DevOps i CI/CD

Fase: 4

Activitat: 6.1

Grup/Individual:

Membres/Alumne:

0- Objectius de l'activitat

Els objectius d'aquesta activitat són:

- Instal·lació i configuració inicial del programari del sistema de control de versions dins del nostre sistema.
- Creació d'un compte en el núvol per hostatjar projectes utilitzant el mateix programari de control de versions.
- Creació de dipòsits locals i remots.
- Sincronització entre els dipòsits locals i remots.

1- Conceptes sobre versions de controls

[a\) Breu explicació sobre sistemes de control de versions](#)

[b\) Introducció al sistema de control de versions Git](#)

[c\) Què és Github?](#)

2- Creació d'un compte de Github

2.1- Creació d'un compte de Github

Ja sabem que **Github** és un servei per hostatjar projectes en el núvol que permet utilitzar com a sistema de control de versions el programa **Git**. Per poder utilitzar-lo caldrà que creem un compte. Per fer-ho he de seguir els següents passos:

- Accedir a **Github** --> <https://github.com/>
- Accedir a la secció **Sign up**.
- Crear un compte:
 - Utilitza el correu de l'escola o qualsevol altra correu particular ([Proton Mail](#), [GMail](#), etc...).
 - Escull la contrasenya que tu creguis convenient.
 - Crea el nom d'usuari que tu creguis convenient.
 - Introdueix el codi de verificació que hauràs rebut al teu compte de correu i accedeix per primera vegada a **Github**.
 - Per a la resta d'opcions de configuració inicial, utilitza el criteri que creguis més convenient.

2.2- Creació d'un PAT (Personal Authentication Token) per accedir a dipòsit remot de Github

a) **Github** demana crear i utilitzar un **PAT** (Personal Authentication Token) per poder pujar noves versions de codi des d'un dipòsit local a un dipòsit hostatjat als seus servidors. Per crear un **PAT** segueix la documentació que trobaràs [aquí](#) amb les següents consideracions en el moment de crear-lo:

- Crea un Token de tipus **classic**
- A la secció **Note** escriu: **pj9**
- A la secció **Expiration** → Selecciona **No expiration**
- A la secció **Select scopes** → Selecciona totes les opcions

c) Copia el teu token en un lloc segur perquè no tornaran a ser visibles un cop hagis sortit de la web de creació de PATs.

3- Instal·lació i configuració inicial de Git

a) Descarrega i instal·la el programari de control de versió **Git** per la teva versió de sistema operatiu:

- Per **Debian GNU/Linux** executa:
 - **sudo aptitude update**
 - **sudo aptitude install git**
- Per **Windows**:
 - Descarrega **Git** des de l'enllaç <https://git-scm.com/download/win>
 - Executa l'instal·lador amb les opcions per defecte.

NOTA: Es convenient que tinguis **git** a la teva màquina física i les teves màquines virtuals.

b) Configuració inicial de Git:

- Per **Debian GNU/Linux** i altres distribucions, executa dins del teu directori personal:
 - **git config --global user.name "nom_usuari_github"**
 - **git config --global user.email "adreça_correu_github"**
 - **git config --global core.editor nano**
 - **git config --global init.defaultBranch main**
- Per **Windows** executa dins del teu directori personal:
 - **git config --global user.name "nom_usuari_github"**
 - **git config --global user.email "adreça_correu_github"**
 - **git config --global core.editor notepad.exe**
 - **git config --global init.defaultBranch main**

NOTA 1: Pensa que **nom_usuari_github** és el teu nom d'usuari real de **GitHub**.

NOTA 2: Pensa que **adreça_correu_github** és l'adreça real que vas utilitzar pel teu compte de **GitHub**.

4- Creació d'un projecte amb un dipòsit local Git. Pujant el codi al dipòsit local

a) Crea una carpeta de nom **projectes**, i dins de **projectes**, crea una altra carpeta de nom **loginteller**. Dins de la carpeta **loginteller** descarrega els següents tres fitxers:

- El codi font del programa de nom [loginteller.c](#)
- El fitxer per la compil·lació del codi [Makefile](#)
- El fitxer de documentació [README](#)

b) Dins de la carpeta **loginteller**, executa les següents ordres per crear un dipòsit local **Git** dins del projecte i pujar la versió actual del projecte al dipòsit:

- **git init**
- **git add loginteller Makefile README**
- **git commit -m "Commit 1 del projecte loginteller"**

c) Executa → **git log** i comprova que el resultat és similar a aquest:

```
dacomo@daw2m08:~/loginteller$ git log
commit ea3cefc26bae5fdc41550490255e84ca5f598282 (HEAD -> main)
Author: dacomo <dacomo2223daw2@proton.me>
Date:   Wed Sep 14 12:26:08 2022 +0200

    Commit 1 del projecte loginteller
dacomo@daw2m08:~/loginteller$
```

d) Executa → `git show --pretty="" --name-only`


i comprova que el resultat és similar a aquest:

```
dacomo@daw2m08:~/loginteller$ git show --pretty="" --name-only
Makefile
README
loginteller.c
dacomo@daw2m08:~/loginteller$
```

e) Comprova que dins de la carpeta **loginteller** s'ha creat una carpeta de nom **.git** (oculta en Linux).

f) Si el resultats mostrats pel teu sistema estan d'acord amb allò que es mostra als apartats c), d) i e), llavors vol dir que el sistema Git funciona correctament, que has creat un dipòsit local Git dins del teu projecte loginteller i que la primera versió dels fitxers ja ha pujat al dipòsit local.

5- Creació d'un dipòsit remot dins de Github. Pujada del contingut del dipòsit local al remot

a) Fes clic a la icona  (a la part superior a l'esquerra) dins del teu compte de **GitHub** i a continuació fes clic a **Create repository**.

b) Crea un nou dipòsit de **GitHub** de nom **loginteller** i d'accés **Públic**. Deixa la resta d'opcions sense omplir per defecte. Finalment, fes clic a **Create repository**.

c) Copia l'adreça URL del teu dipòsit remot de **GitHub** que trobaràs a la secció **Quick setup** del nou dipòsit un cop acabat de crear.

d) Dins del teu projecte local **loginteller** executa:

- `git remote add origin https://github.com/nom_usuari_github/loginteller.git`
- `git branch -M main`
- `git push -u origin main`

NOTA 1: Per la primera ordre, recorda que **nom_usuari_github** és el teu nom d'usuari real de **GitHub**.

NOTA 2: Per la tercera ordre et demanarà el **nom d'usuari** del teu compte de **GitHub** que vas crear a l'apartat 2.1 i el **Token** que vas crear a l'apartat 2.2.

e) Comprova que el sistema mostra alguna cosa similar a:

```
dacomo@daw2m08:~/projectes/loginteller$ git push -u origin main
Username for 'https://github.com': dacomo2223daw2
Password for 'https://dacomo2223daw2@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 1.68 KiB | 861.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/dacomo2223daw2/loginteller.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
```

f) Ara refresca la pàgina web d'administració del dipòsit loginteller dins de **GitHub** i comprova que el resultat és similar a aquest: <https://github.com/asix2pj9/loginteller>

g) Si el resultats mostrats pel teu sistema estan d'acord amb allò que es mostra als apartats e) i f) llavors, vol dir que el teu dipòsit local **Git** està sincronitzat amb el dipòsit remot hostjat a **GitHub**.

6- Clonació d'un projecte remot dins del teu sistema creant un nou dipòsit local

a) Accedeix a la carpeta **projectes** que vas crear a l'apartat 4 de l'activitat. Un cop dins de la carpeta **projectes** executa la següent ordre:

- `git clone https://github.com/globproj2/m08uf1pr3.git`

b) Comprova que dins de la carpeta **projectes** s'ha creat un nou directori de nom **m08uf1pr3**. Accedeix al nou directori i comprova que hi ha 3 fitxers de nom **LICENSE**, **LLEGEIX-ME** i **Vagrantfile**. Comprova també que s'ha creat la carpeta **.git** amb el dipòsit local clonat del dipòsit remot.

c) Executa → `git log` i comprova que el resultat és similar a aquest:

```
daniel@sh3:~/tmp/m08uf1pr3$ git log --pretty=oneline
de9db105ce3bb92aa6500555f9689fdb1082ee7 (HEAD -> main, origin/main, origin/HEAD) Commit 6 de m8uf1pr3 DAW2 FJE CLOT 2021-2022
e650cd4f26d207f2af395d335ff05027528c76d9f Commit 5 de m8uf1pr3 DAW2 FJE CLOT 2021-2022
b21a0585c9fa46bf57f238d8ab22bab875f83451 Commit 4 de m8uf1pr3 DAW2 FJE CLOT 2021-2022
e4ba44b2952d6ff2d4e01308e7b009bb5fdaa458 Commit 3 de m8uf1pr3 DAW2 FJE CLOT 2021-2022
0f58e0f40bc3dd121fb983ea76229a190aa0c865 Commit 2 de m8uf1pr3 DAW2 FJE CLOT 2021-2022
7e7a22343f21b6cb55258f313a2fa3679e8facde Update LLEGEIX-ME
587f9148f26872b81476d8599b62978804cacdf7 Commit 1 de m8uf1pr3 DAW2 FJE CLOT 2021-2022
```

Lliurament de l'activitat

a) Envia un correu a **cf@collados.org** amb:

- Assumpte: **asix2_cognom_nom_pj9f4a6.1**.
- El nom d'usuari del teu compte de **GitHub** (només el nom, no vull saber la contrasenya).
- El teu compte de correu associat (només el nom, no vull saber la contrasenya).
- L'adreça URL del teu dipòsit de **GitHub**.
- Una captura de pantalla des d'un terminal del resultat d'executar l'ordre `git remote -v` dins de la carpeta del projecte **loginteller**.
- Una captura de pantalla des d'un terminal del resultat d'executar l'ordre `git log` dins de la carpeta del projecte **loginteller**.
- Una captura de pantalla des d'un terminal del resultat d'executar l'ordre `git log --pretty=oneline` dins de la carpeta clonada del projecte **m08uf1pr3**.

b) Data límit: **Dimecres 23-09-24** fins a les **16.50**. Fora d'aquest límit, aquesta activitat tindrà una menor puntuació.