Guide för Git-användning

# Git versionshantering

Varje enskilt projekt har en egen ”övermapp” som används som Repository. Här samlas allting som har med det projektet att göra.

För gitsystemet används en ”Remote Repository” (Global Git-mapp som ligger på servern) som du synkar din lokala ”Repository” (Lokal Git-mapp som ligger på din dator) till. Versionshantering sparas då i två steg, först till din lokala git-mapp och sedan till den globala git-mappen.

På så vis kan du göra flera ändringar i projektet och versionshantera lokalt för varje steg, innan du väljer att ladda upp det du sparat till servern i ett svep.

Med hälp av versionshanteringen är det sen möjligt att återgå (”Revert”) till hur filen såg ut i det steget.

## Branches

Branches (”grenar”) används för att kunna jobba på flera versioner av projektet samtidigt.

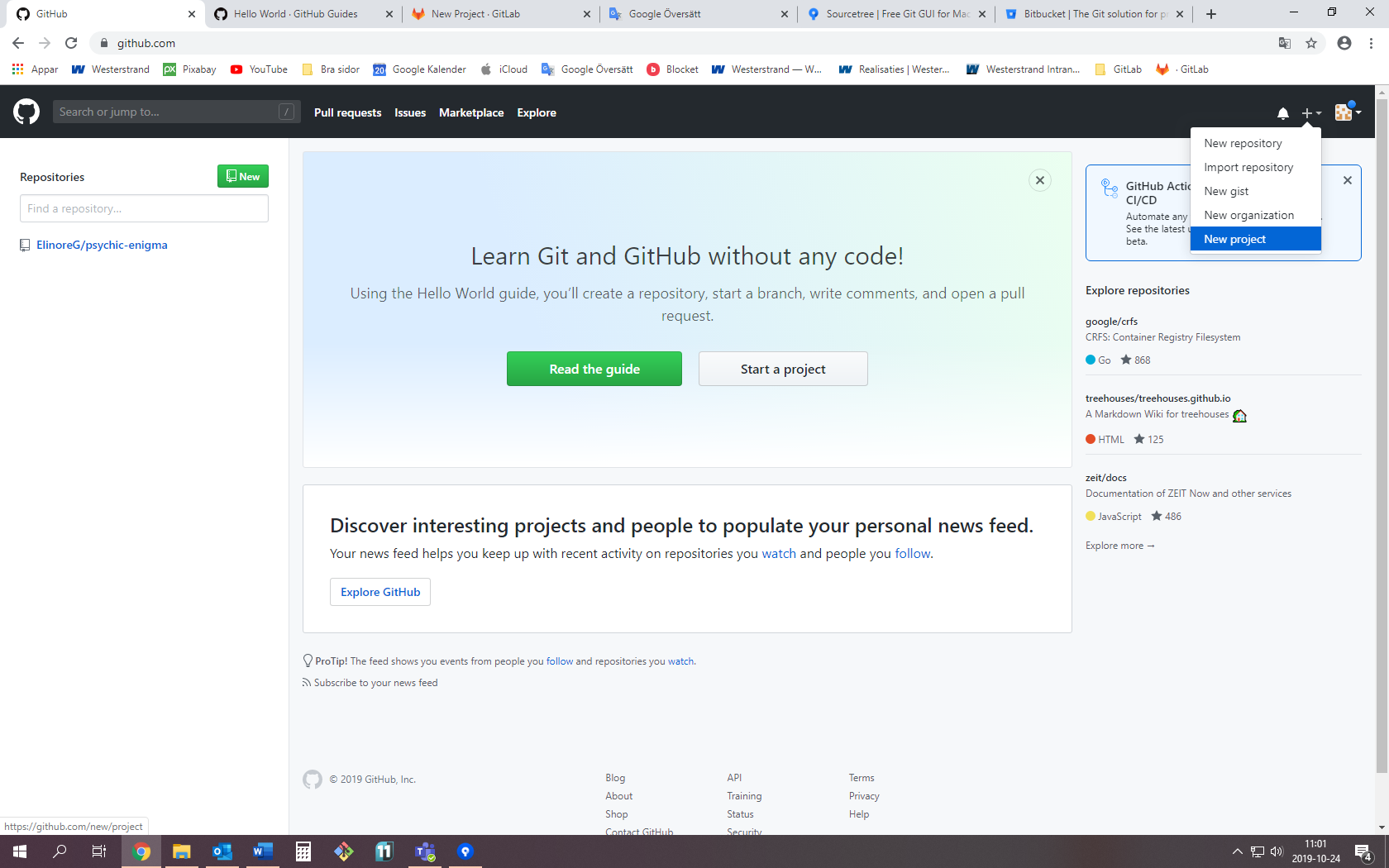
Till exempel om två personer ska jobba med projektet samtidigt. Om person 1 ska utveckla ***funktion X*** i projektet och person 2 ska utveckla ***funktion Y.*** Då hämtar båda personerna först hem projektet i dess huvud-gren **master branch.** Därefter skapar person 1 en ny branch, ger den ett beskrivande namn typ ”**branch-funktion-x**” och jobbar med projektet i den branchen. Person 2 skapar också en ny branch t.ex ”**branch-funktion-y**”.

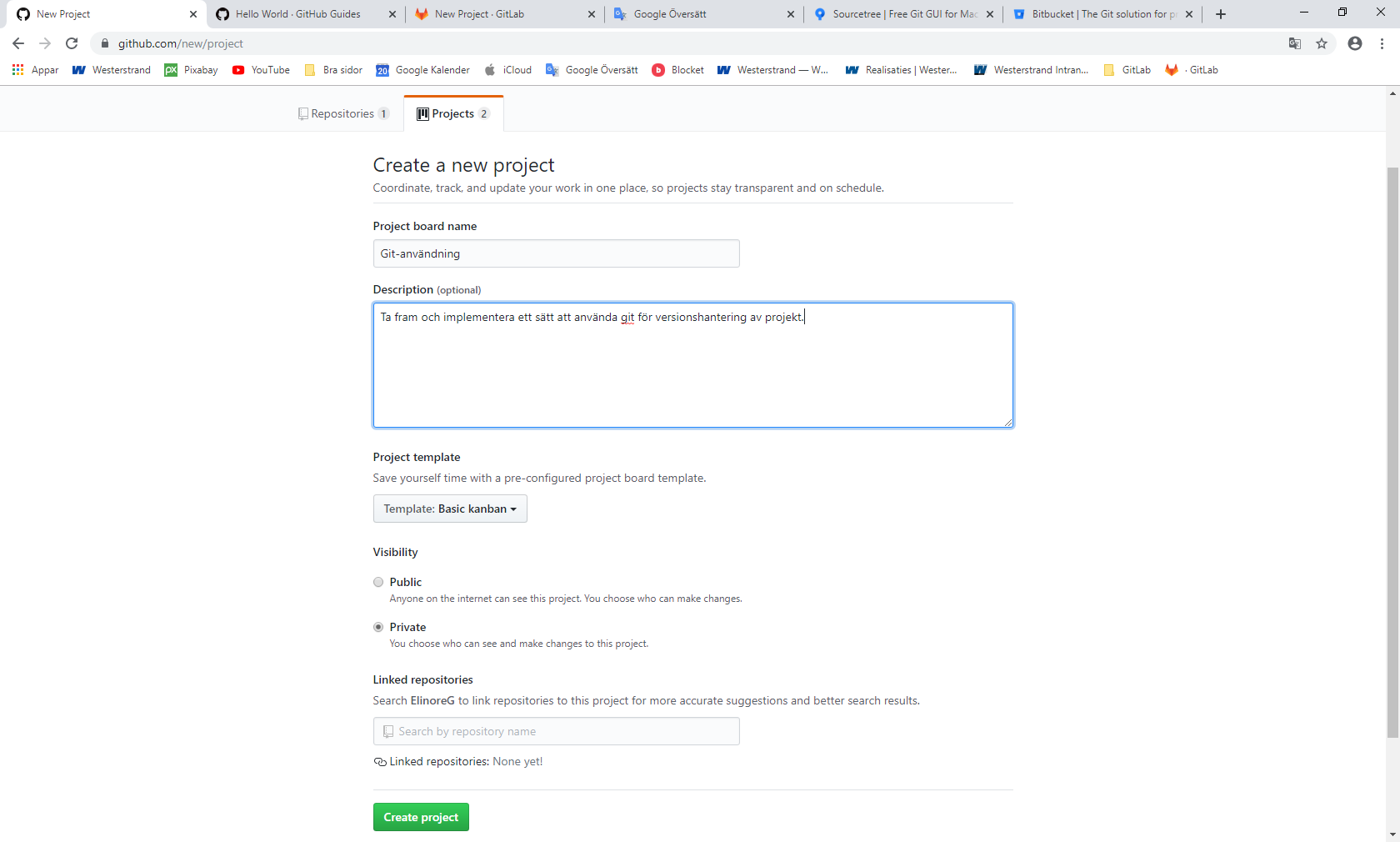
Både person 1 och person 2 kan då jobba med projektet utan att påverka varandra eller master-branchen. När person 1 är klar med funktion x så kan den testas och diskuteras och om allt är okej så kan **branch-funktion-x** sammanfogas (”merge”) med master-branchen. Person 2 kan då hämta hem ändringarna (”Pull”) i master branchen och jobba vidare med funktion y tills den klar. När sen funktion Y är klar kan den också sammanfogas till master-branchen.

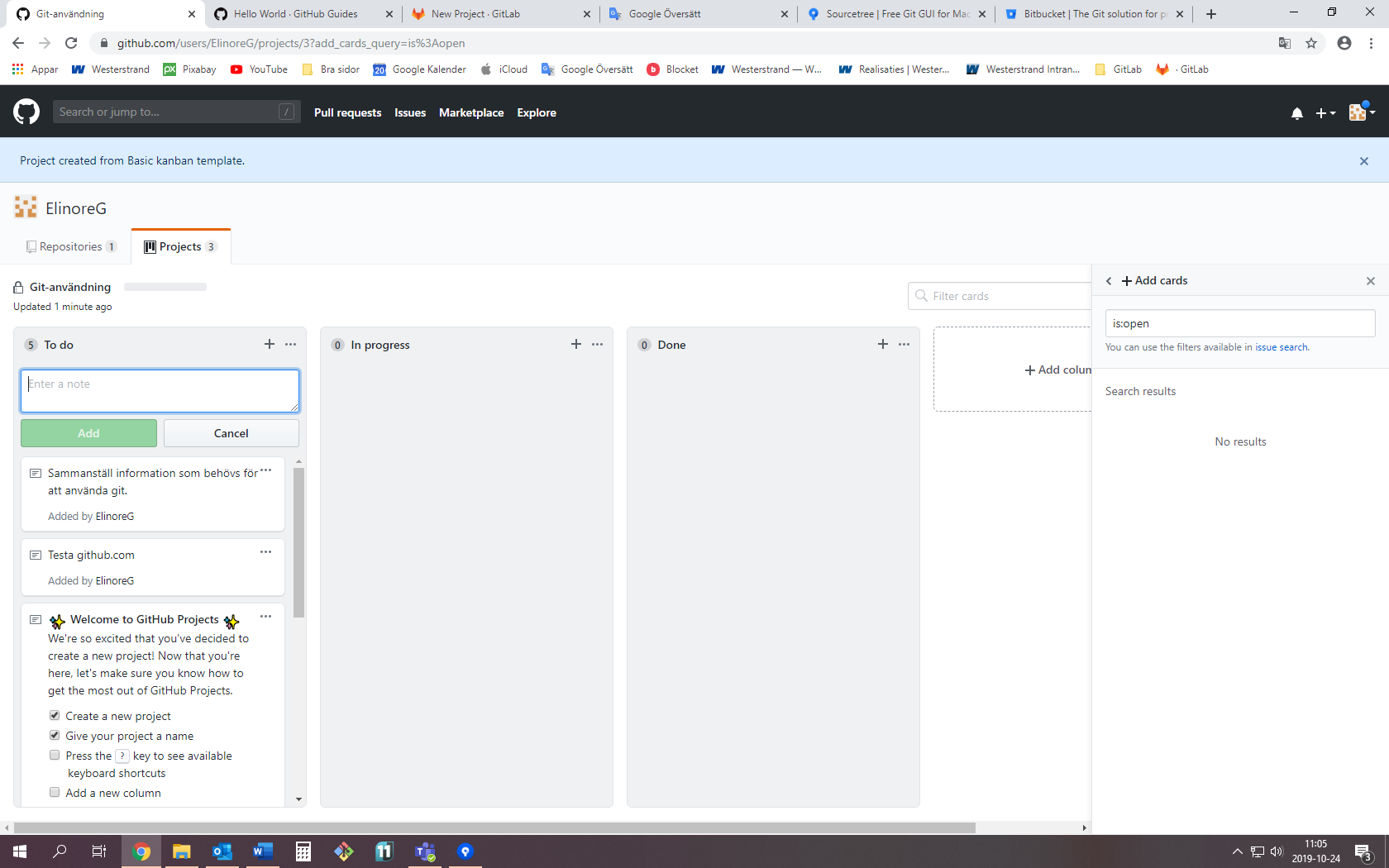
# GitHub

## Användarkonto

## Projekt



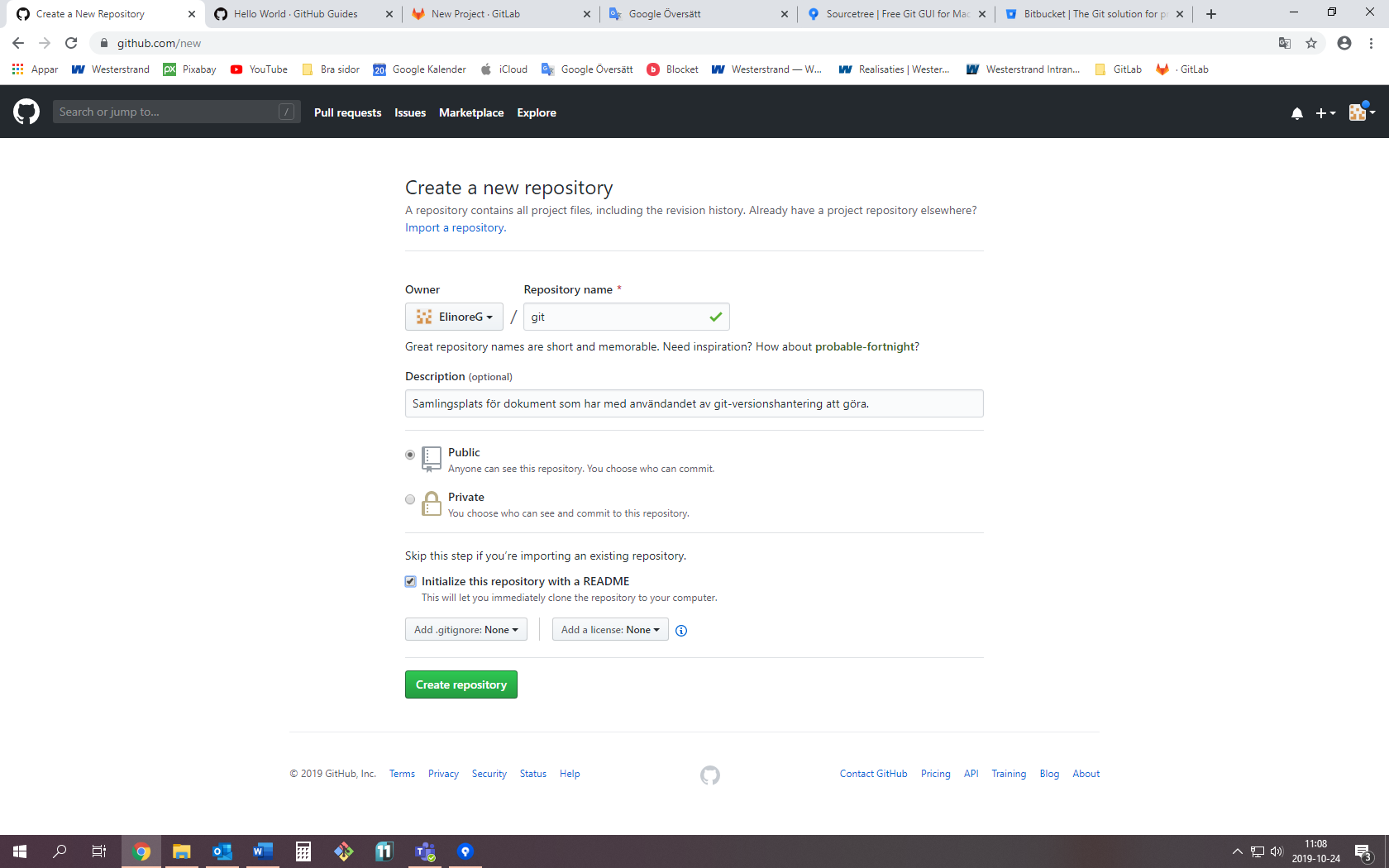




New repository – ny git-mapp

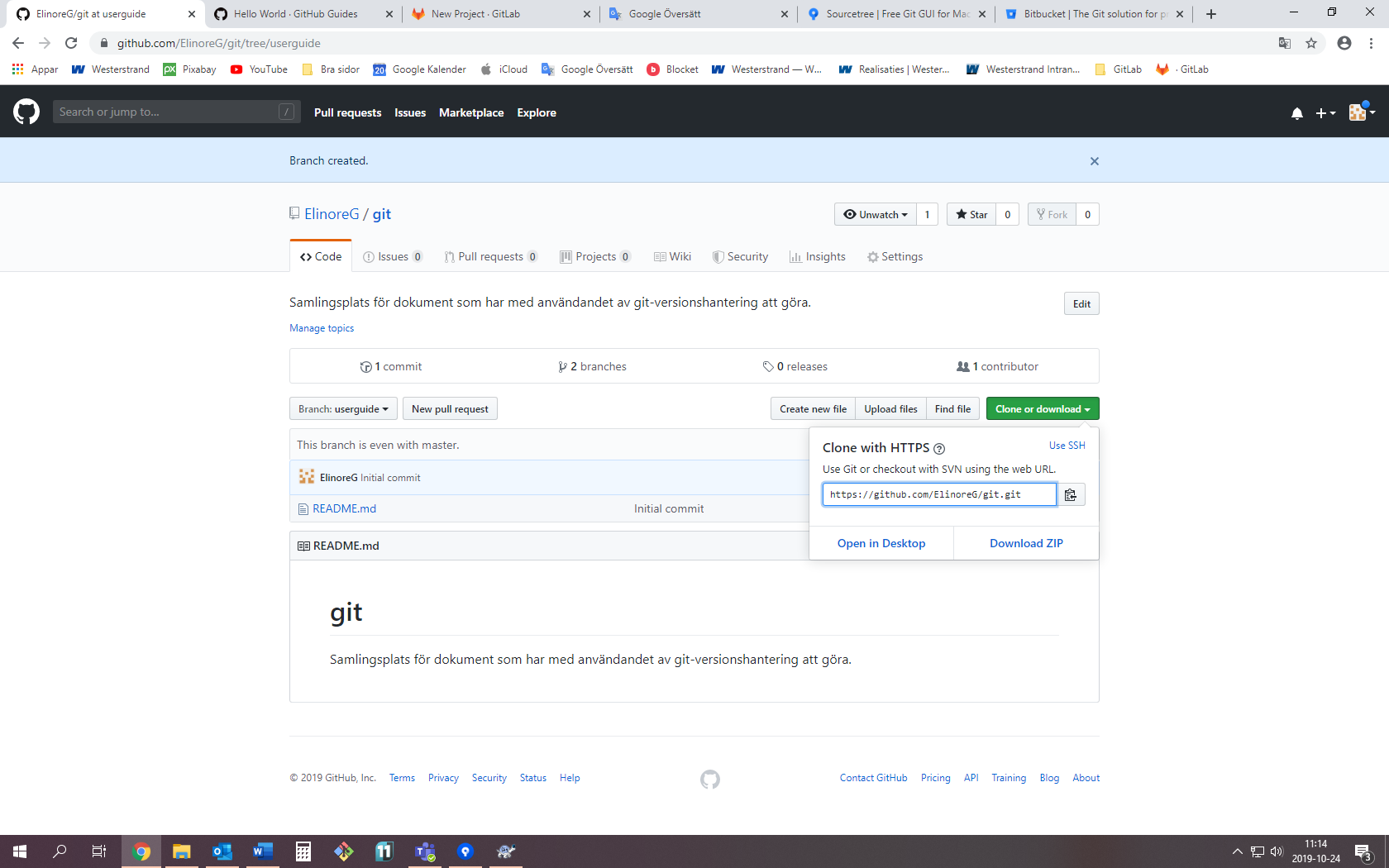


Skriv in information och tryck på Create Repository.



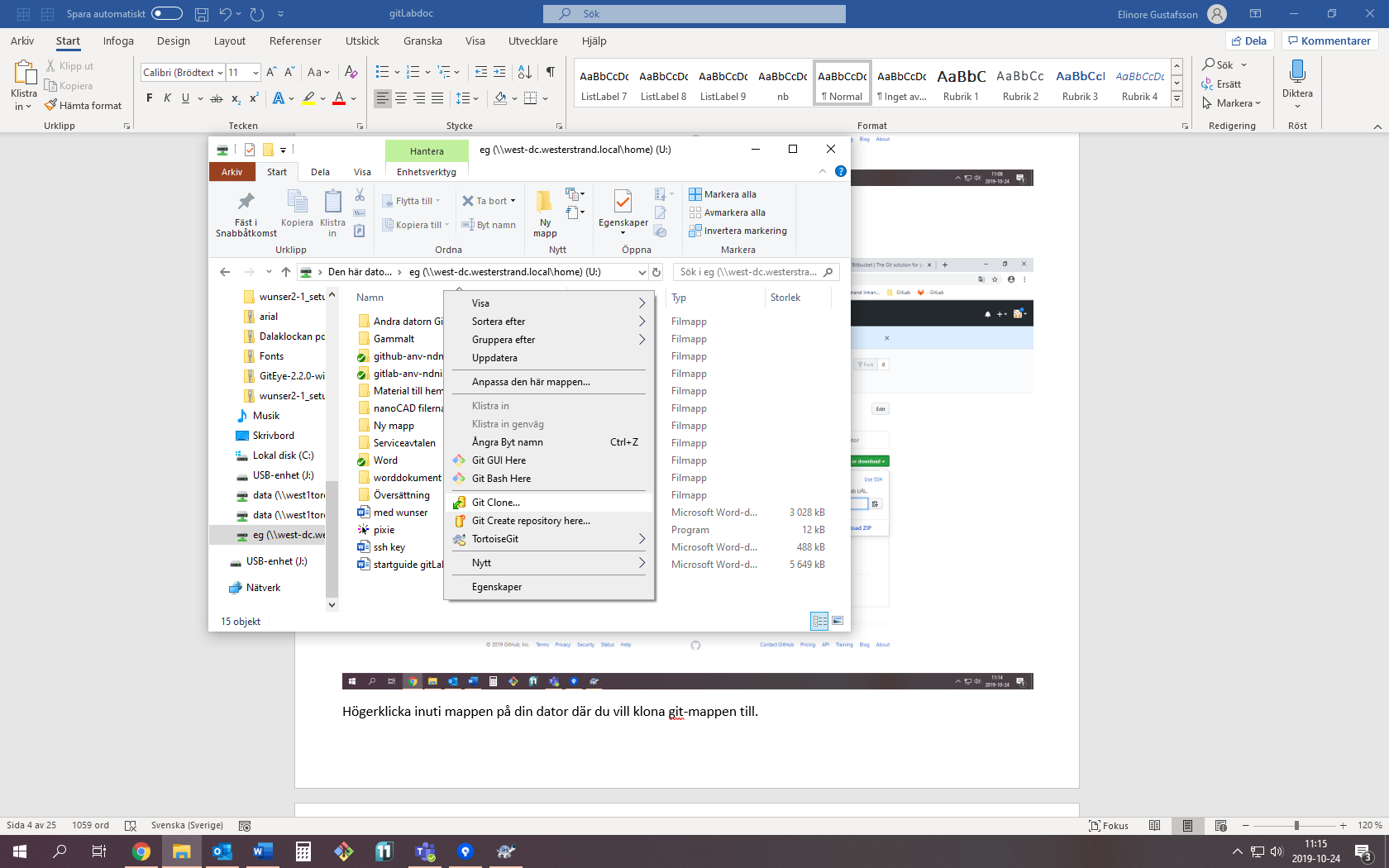
Klona git-mappen till lokalt på din dator:

Kopiera kloninslänken



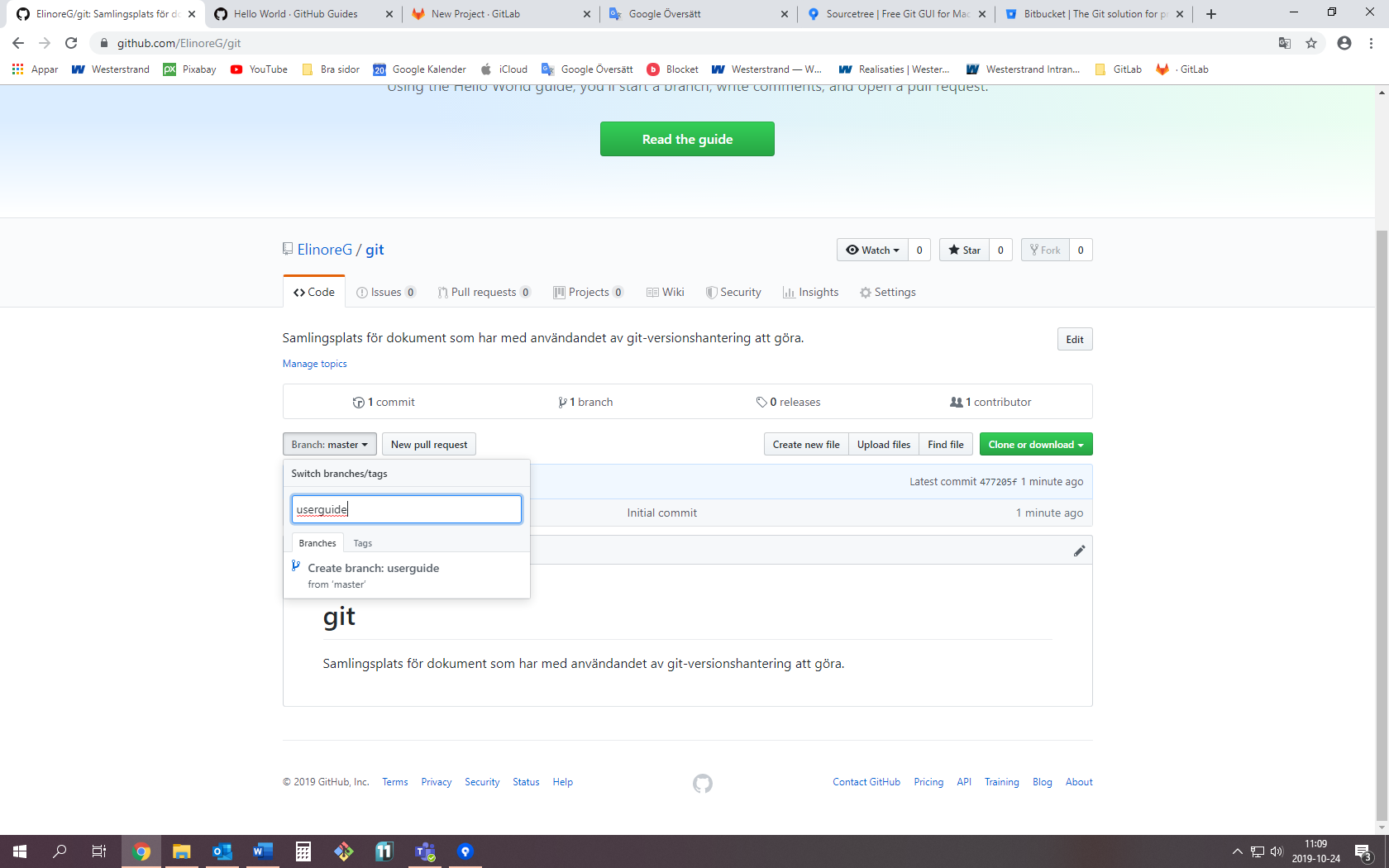
Högerklicka inuti mappen på din dator där du vill klona git-mappen till.

Välj Git Clone.

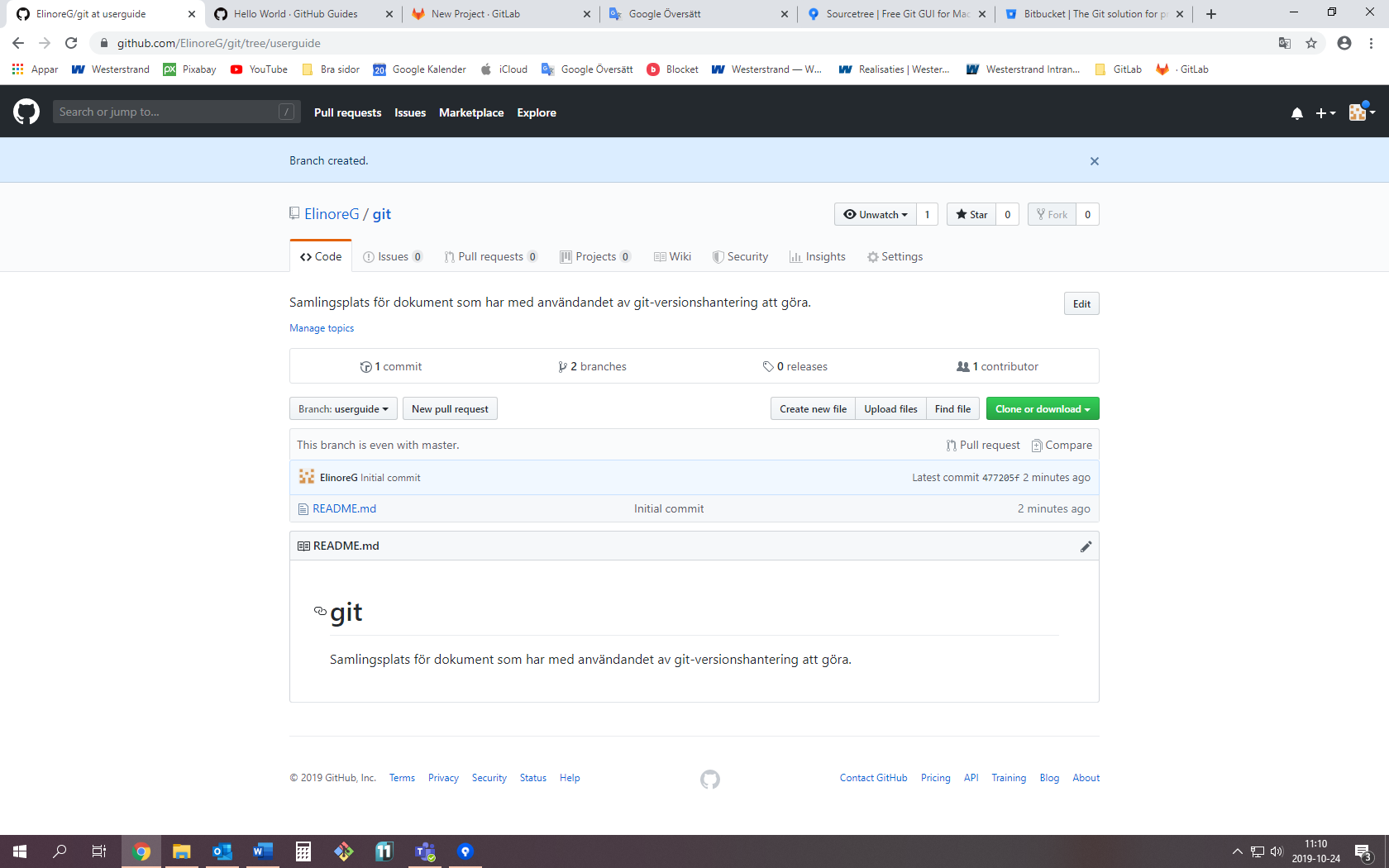


Sen samma som för gitlab

Skapa en branch att jobba i. Ge den ett kort beskrivande namn. Tryck på create branch.

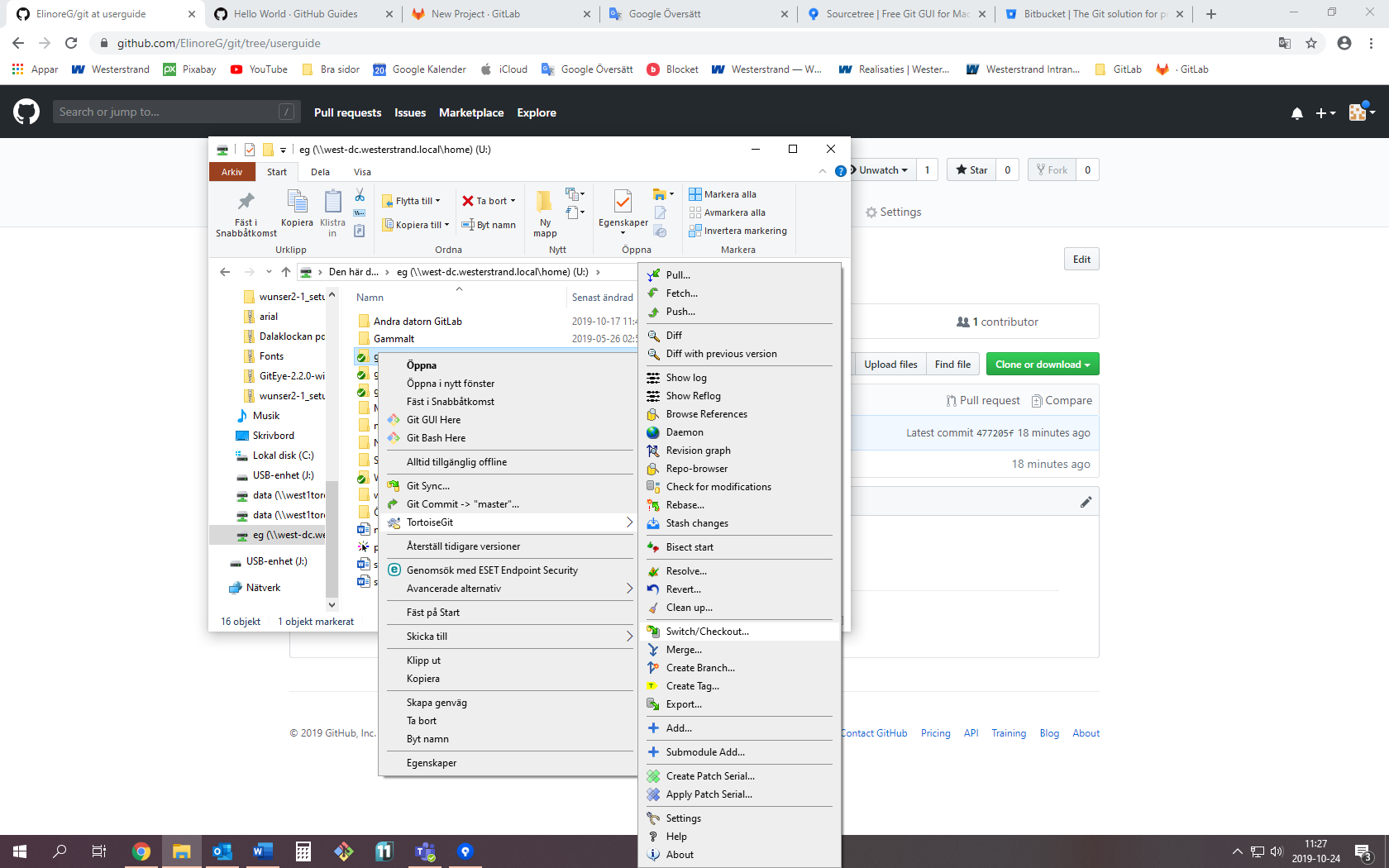


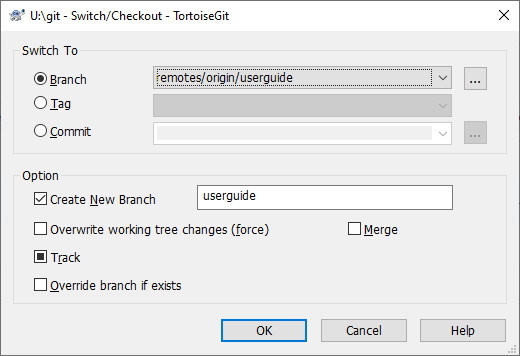
Se vilken branch som visas.



## Byt branch lokalt

Högerklicka på den lokala git-mappen. Välj Switch/Checkout.



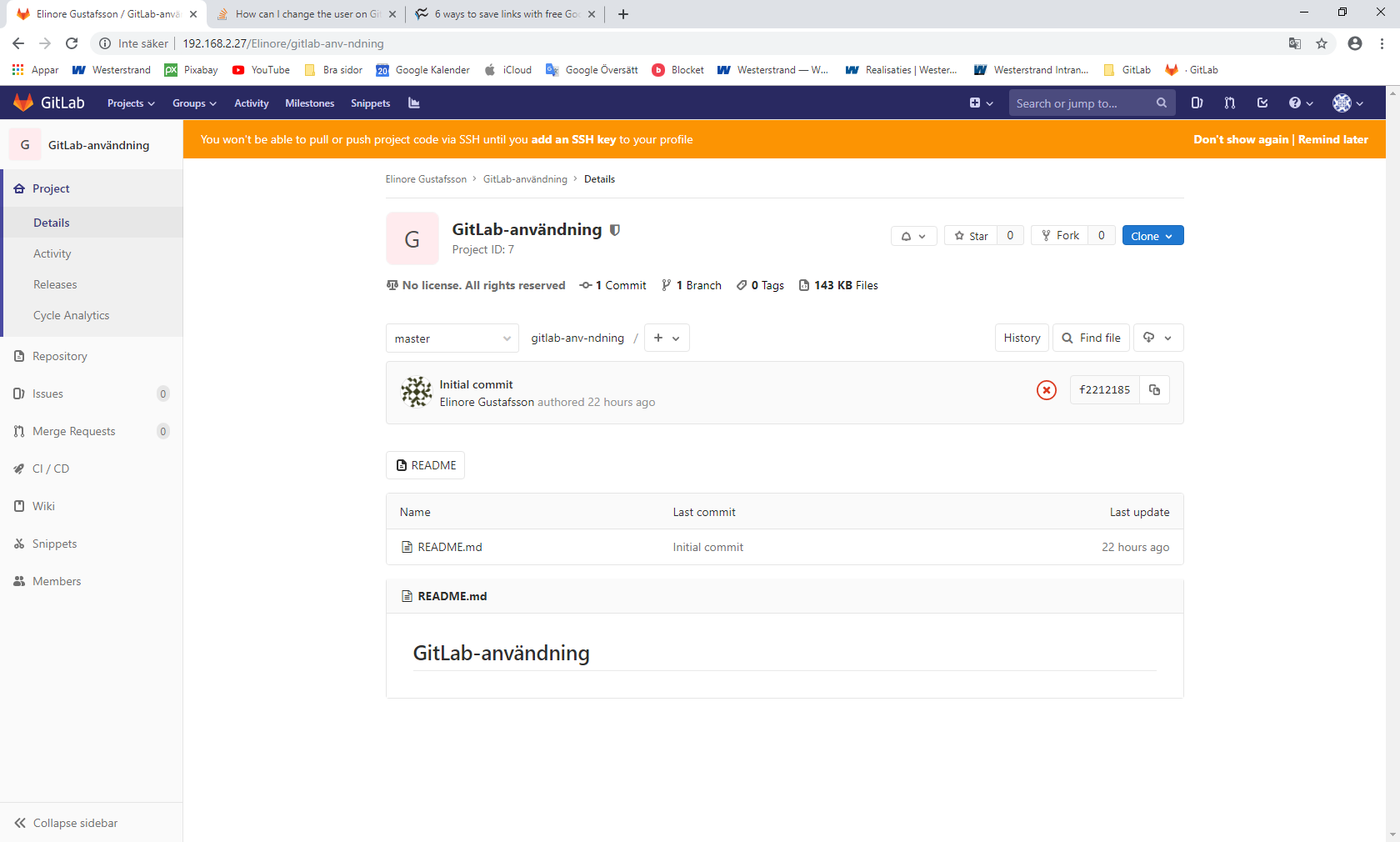


Välj den branch du vill jobba i, i listan.

Låt optionen ”create new branch” vara ikryssad om det är första gången du byter till denna branch (för att skapa branchen lokalt på datorn), samma branchnamn används lokalt.

# GitLab

## Användarkonto



## Projekt

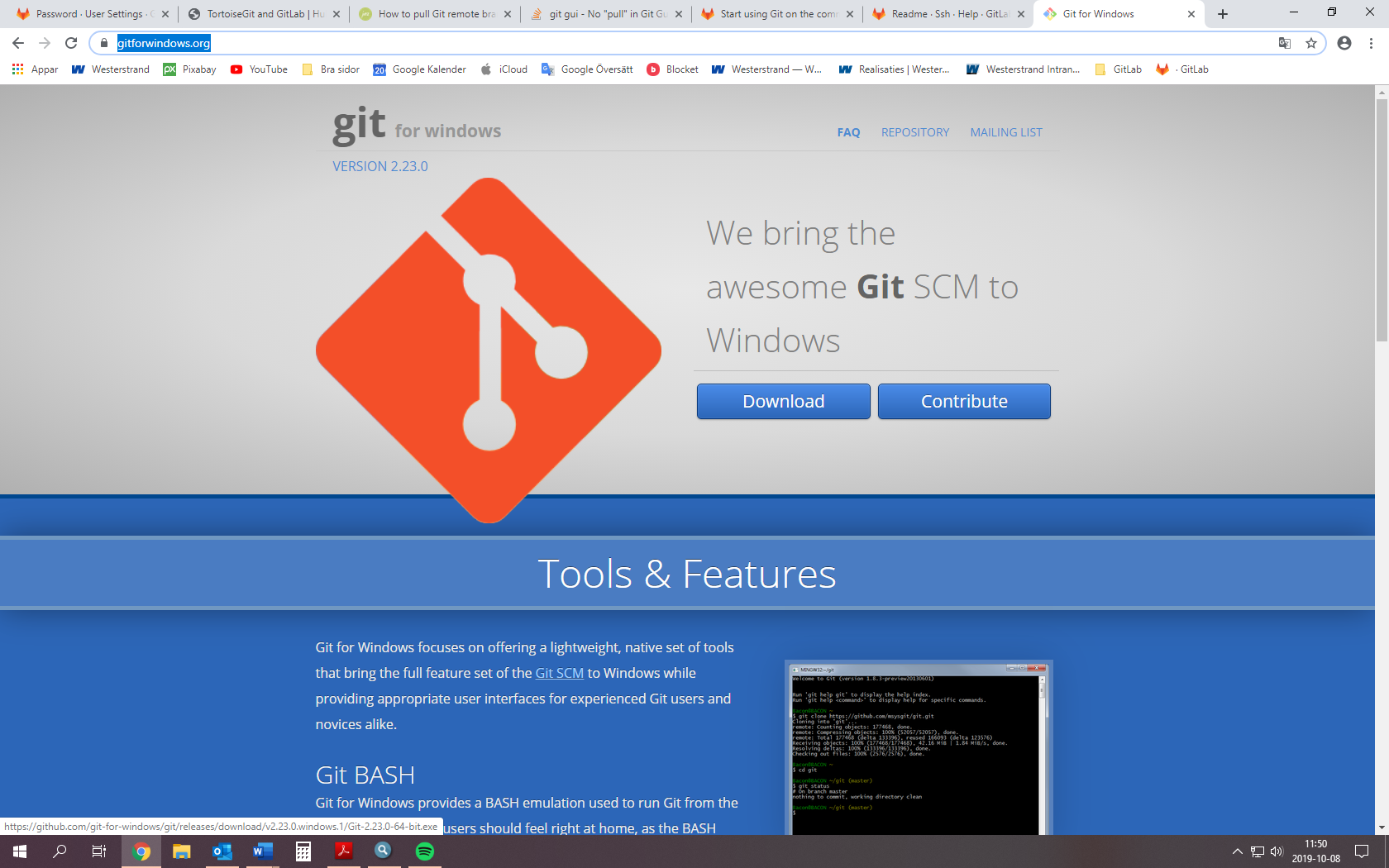
Varje projekt har en egen git-mapp som användare hämtar och laddar upp till från sin lokala git-mapp.

## Grupper

Grupper används för att gruppera projekt/mappar, så en grupp fungerar som en övermapp. De som ska ha tillgång till gruppen läggs till som medlemmar under ”Members”

# Git for windows

Ladda hem Git for Windows från den här länken <https://gitforwindows.org/>



# SSH nyckel (”SSH Key”)

SSH-nyckeln används som just en nyckel för kopplingen mellan din dator och GitLab. Så när du hämtar/skickar data mellan din dator och GitLab så vet GitLab vem som är i andra änden. Lösenordet du väljer får du sen ange för att verifiera när du ska hämta/skicka till GitLab.

SSH nyckeln skapas först på datorn och läggs sedan till på ditt konto via inställningar på GitLab.

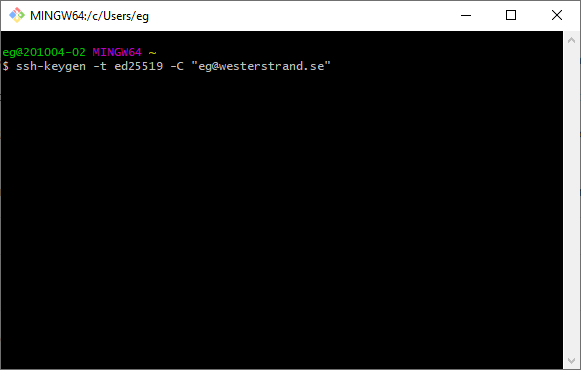
## Skapa en ”SSH key” på din dator

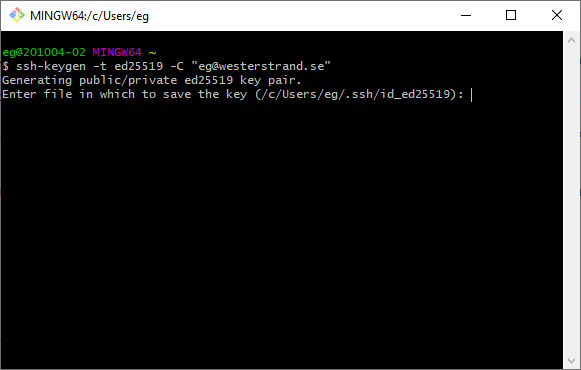
Öppna Git Bash genom att dubbelklicka på GitBash ikonen på skrivbordet.

Skriv in

$**ssh-keygen -t ed25519 -C ”dinepost@westerstrand.se”**

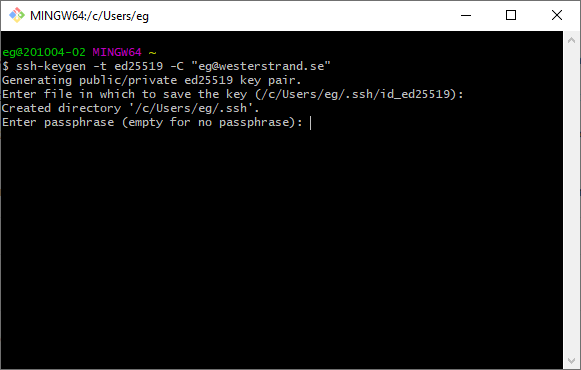
Men byt ut till din egen e-postadress. Tryck på Enter.





Tryck Enter för att spara som föreslås.

Nu får du välja ett lösenord för din ssh-nyckel, det används när du ska skicka/hämta data till/från GitLab sen.   
Så välj ett kort lösenord som du lätt kommer ihåg.



Skriv in önskat lösenord och tryck Enter.

Du får upprepa lösenordet och tryck Enter igen.

Nu bör diverse info och en s.k Randomart bild visas.

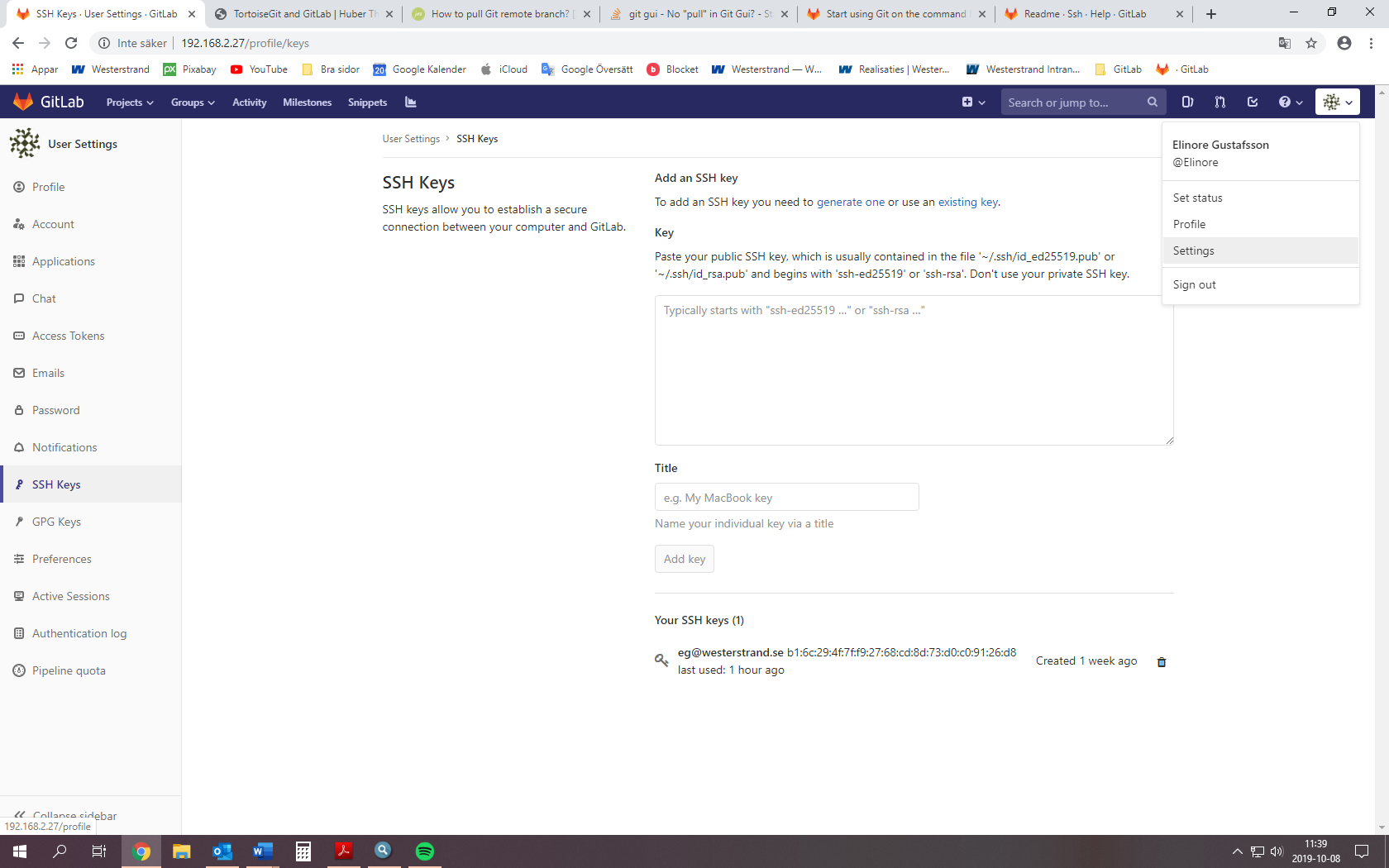
Skriv in följande kommando för att kopiera den bilden:

$**cat ~/.ssh/id\_ed25519.pub | clip**

(Använd tangenten **[alt gr]** och **[ > < ]** samtidigt för att göra vertikalstrecket | )

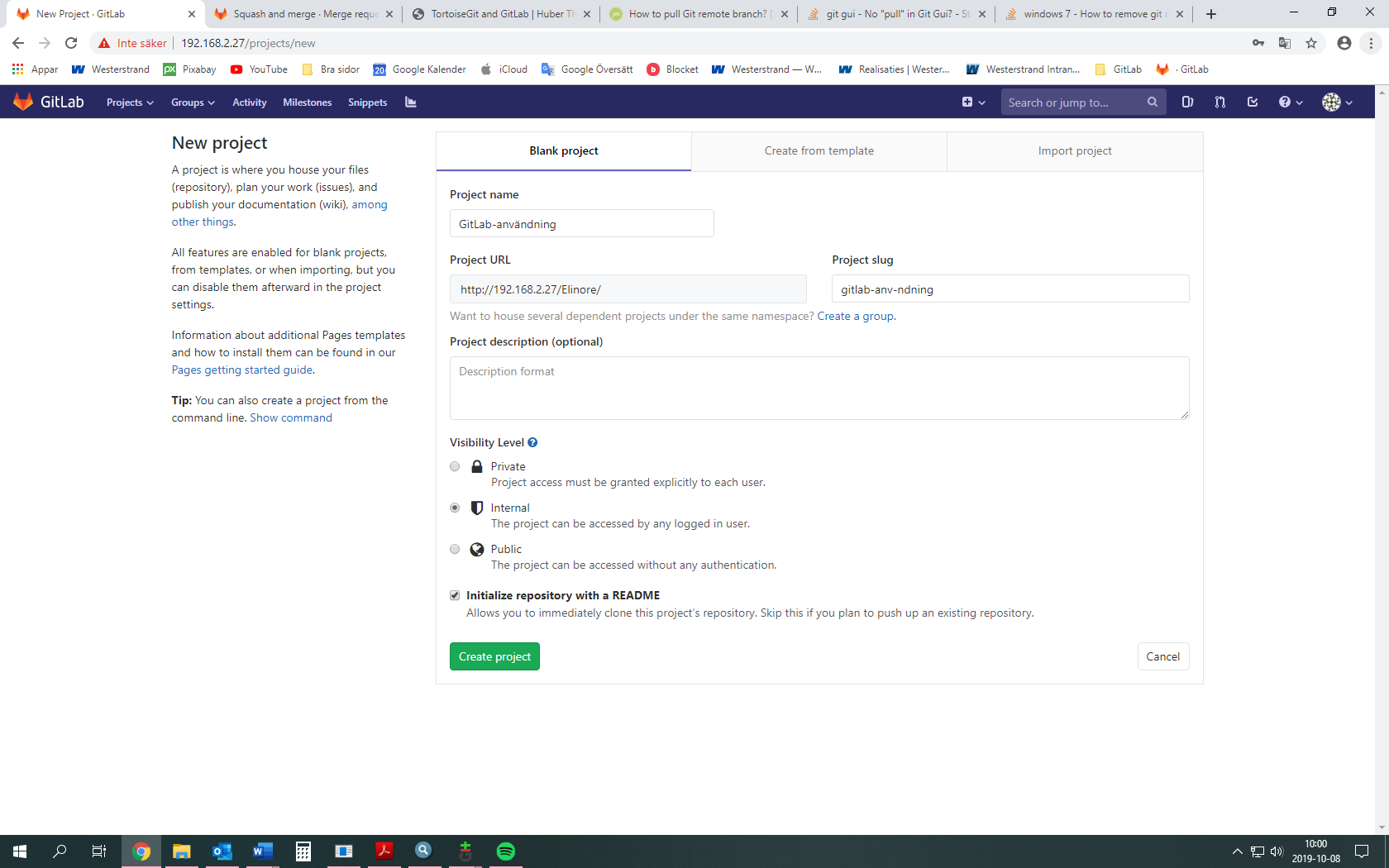
## Lägg till ssh-nyckeln till ditt användarkonto på GitLab

Logga in på GitLab, tryck på ikonen med din avatar längst till höger i menyn och välj Settings.



I den vänstra menyn under User Settings, tryck på SSH Keys.

# Skapa ett nytt projekt på GitLab

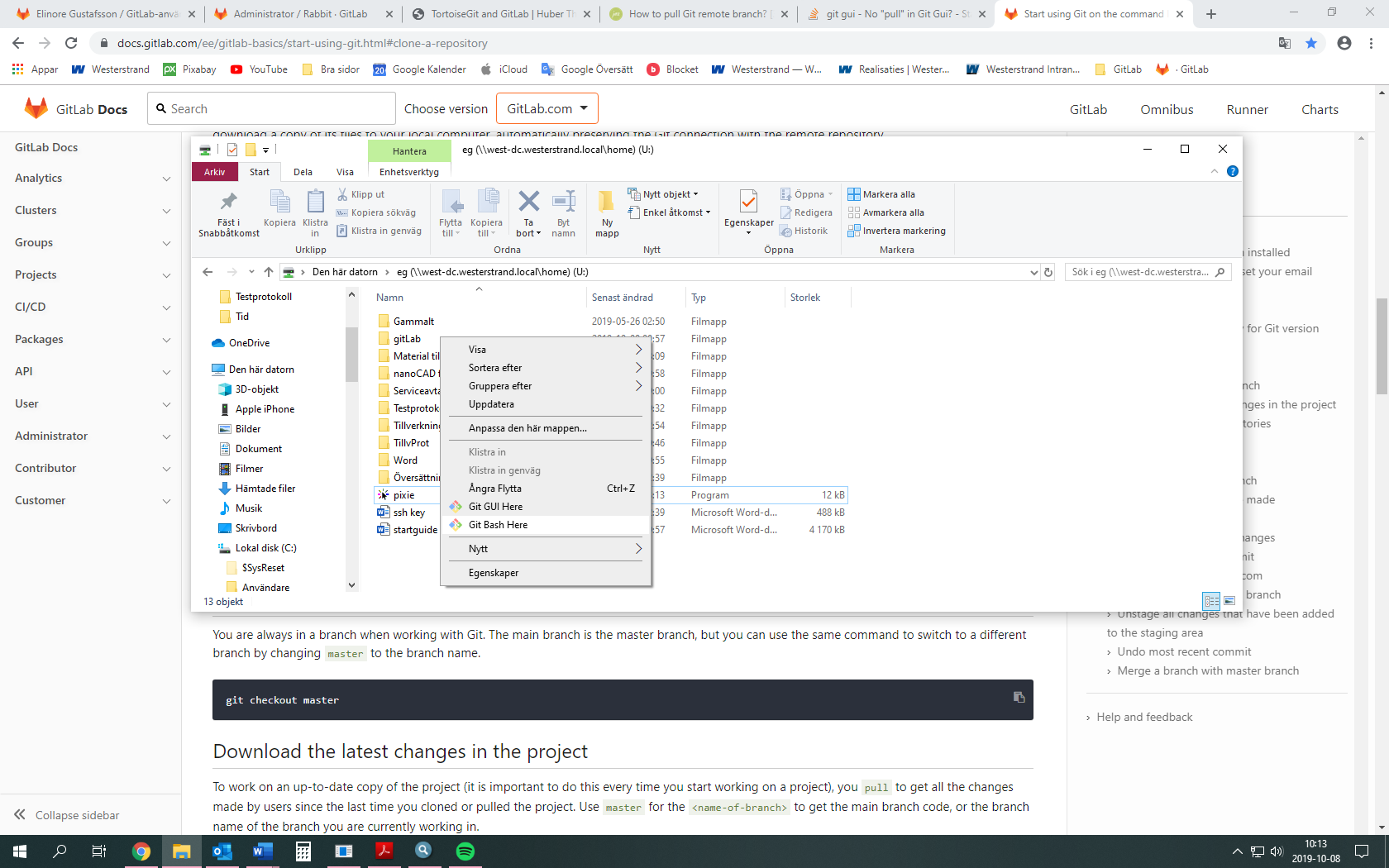


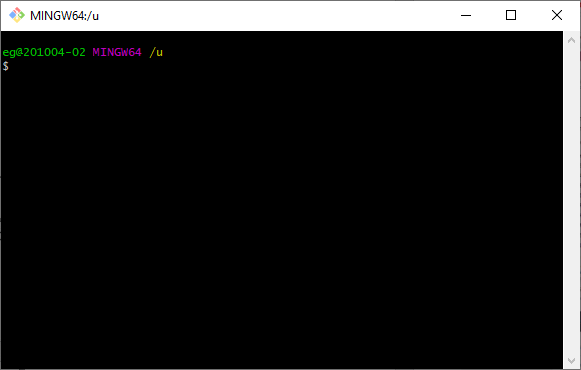
# Klona ett projekt från git

## Klona med GitBash

Gå till den mappen du vill att projektmappen från git ska hamna. Högerklicka någonstans inuti mappen på din dator.

Välj **GitBash here.**





GitBash kommandofönster öppnas.

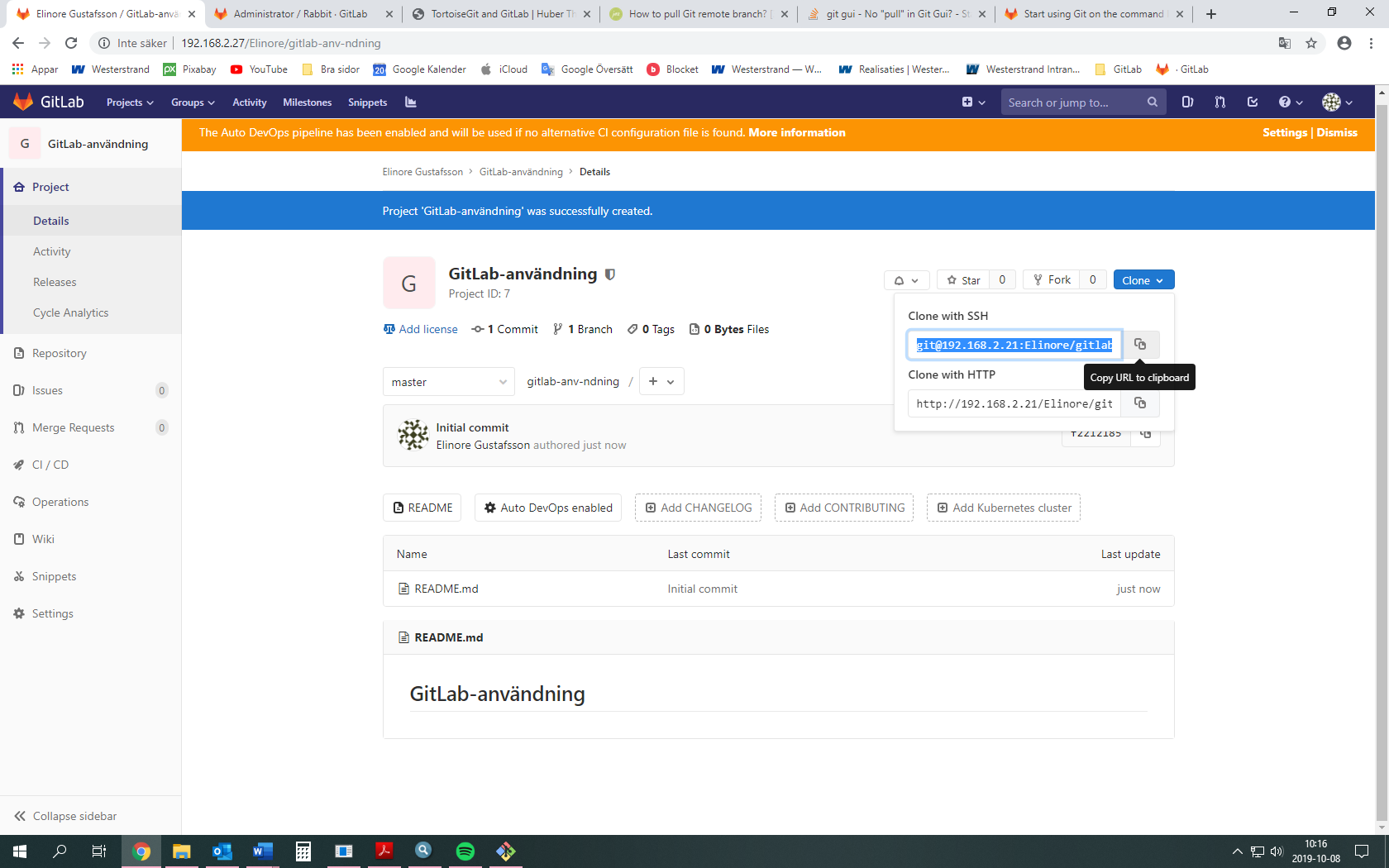
Skriv in

$**git clone git@192.168.2.27:Elinore/gitlab-anv-ndning.git**

**Clone** kommandot är för att klona befintlig mapp från git

**git@192.168.2.27:** är adressen som vårt git-system ligger på

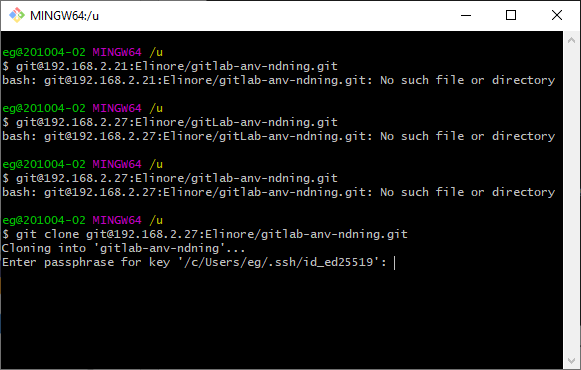
Den sista delen är länken till den mapp vi skall klona från git (den syns i adressfältet på projektsidan, se bild….)



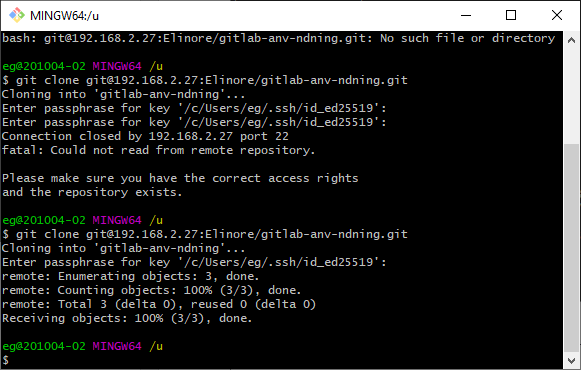


Det är viktigt att skriva helt rätt, se sista försöket här.

Du blir ombedd att ange ditt **ssh lösenord** som du valt själv när du skapade en **ssh key.**



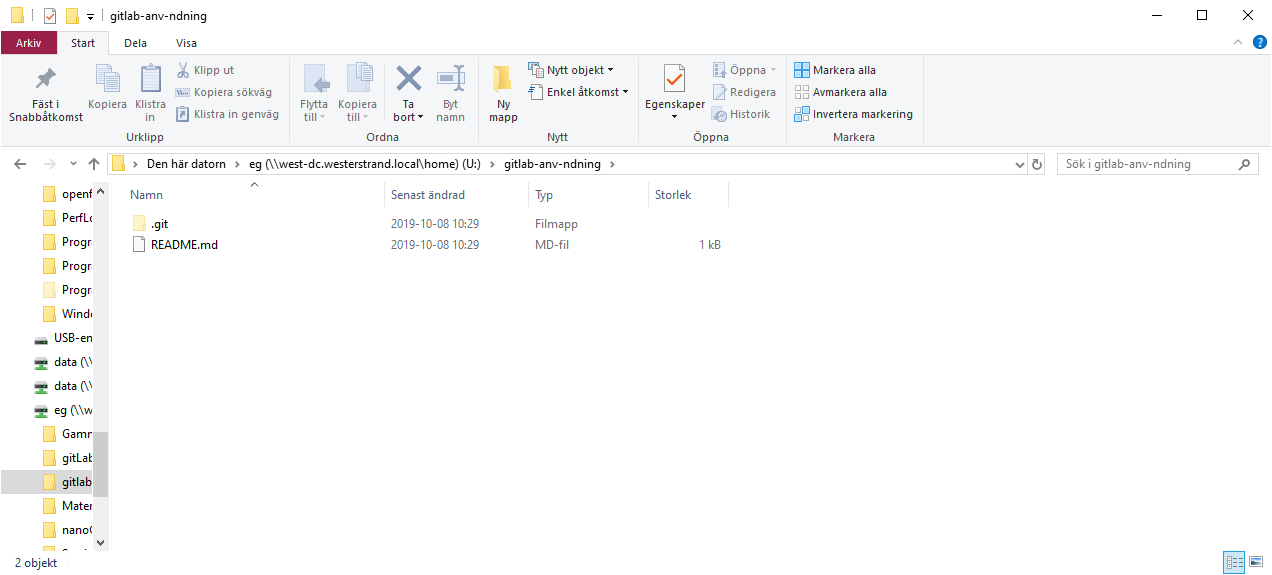
Ange ditt lösenord för ssh och tryck enter.



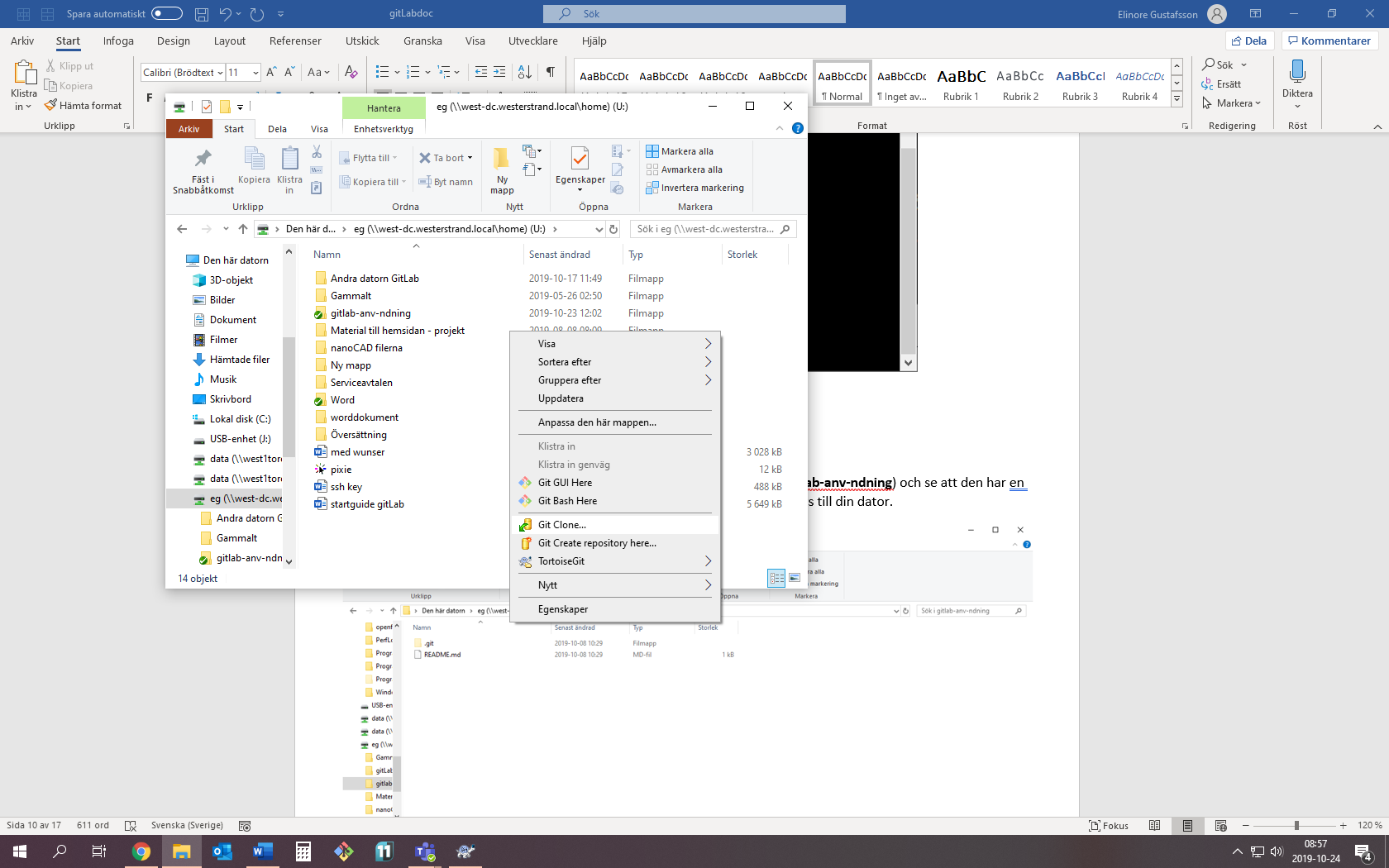
När det lyckas ska det se ut som i stil med sista försöket här.

Stäng sedan ned GitBash och gå till mappen på din dator (i det här fallet **u:** ),

kontrollera att projektmappen har hämtats hem (klonats), öppna mappen (**gitlab-anv-ndning**) och se att den har en **.git -mapp** inuti samt filerna som fanns i projektet uppe på gitLab har nu klonats till din dator.

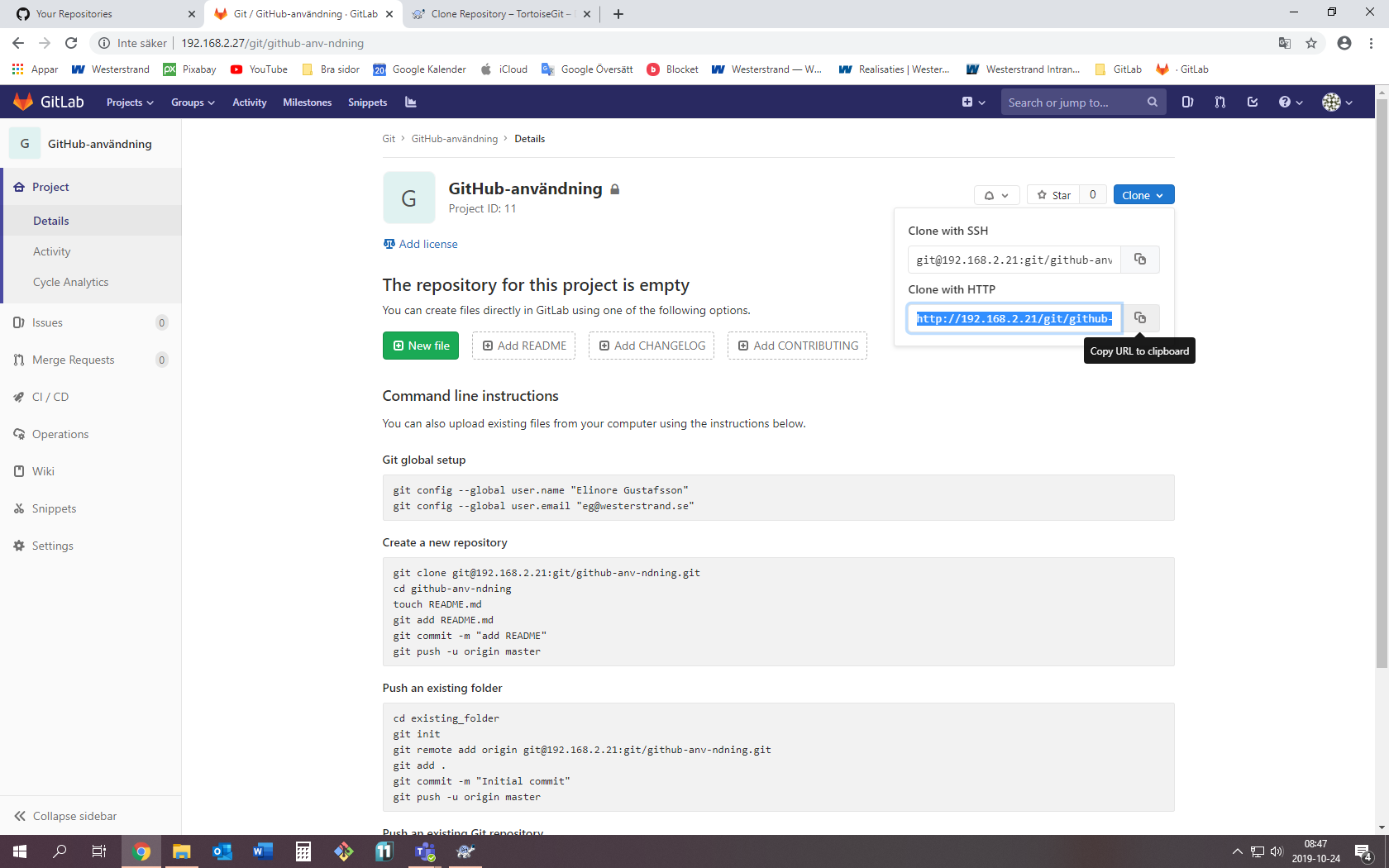


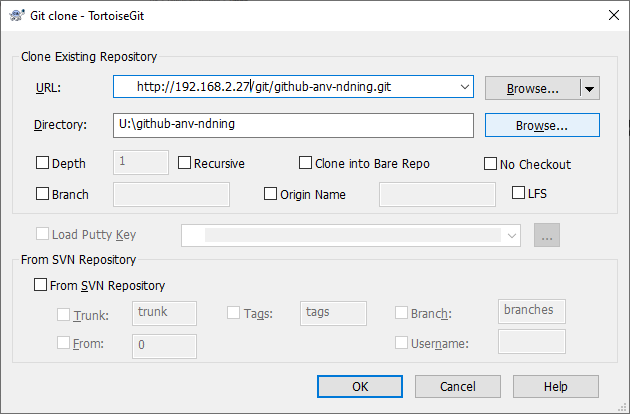
## Klona med TortoiseGit

Börja med att högerklicka inuti mappen på din dator där du vill att mappen från GitLab skall klonas till. Välj Git Clone.

Git clone - TortoiseGit öppnas.

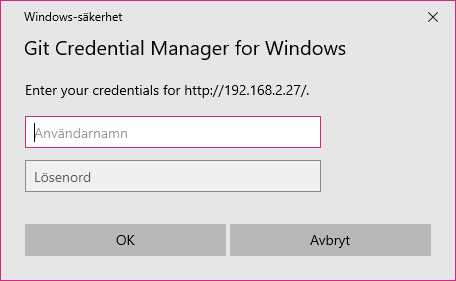
Gå nu till projektets sida på GitLab och kopiera klonings-länken från projektet på GitLab, ”Clone with http”.



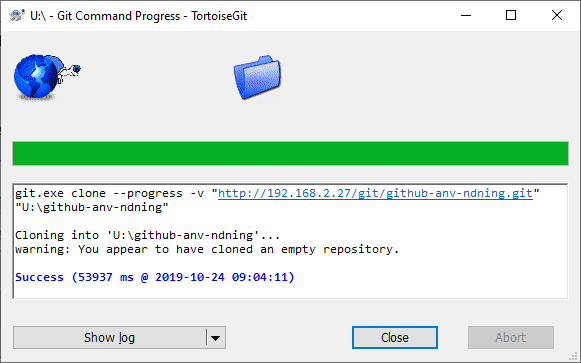


Klistra in ”Clone with http” länken som URL. Ändra Directory om det inte blev som du tänkt dig.

Tryck på OK.



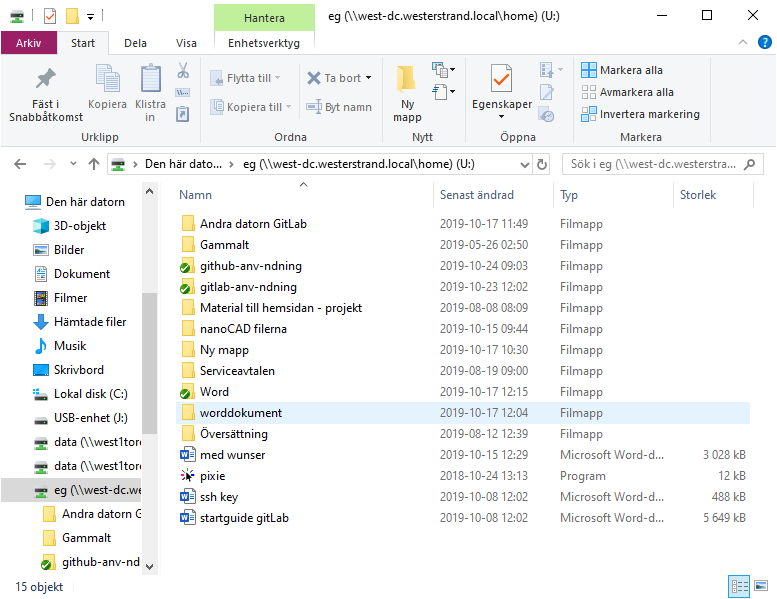
Skriv in ditt användarnamn och lösenord för GitLab.



Vänta tills kloningen är klar.

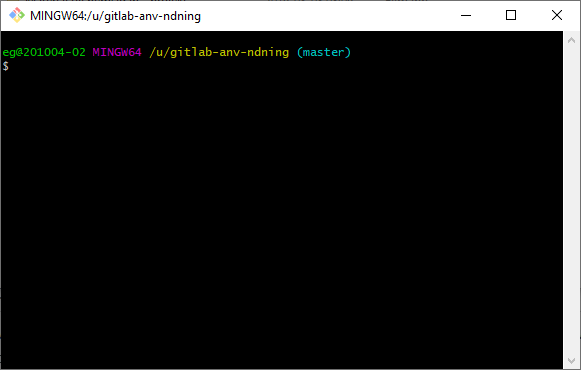
Tryck på Close.

Kolla sedan i din mapp på datorn att git-mappen du klonat nu hamnat där.



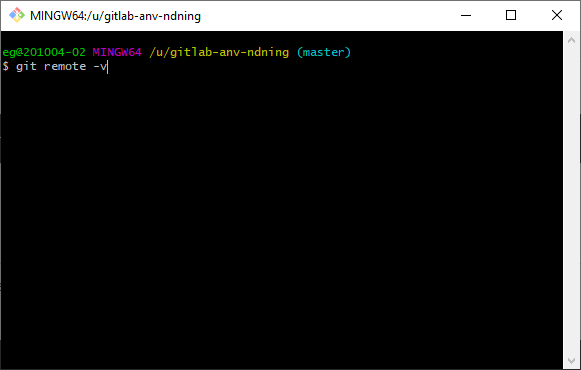
## För att se kopplingen till gitLab mappen:

Högerklicka på mappen på din dator och välj GitBash here.



Skriv in

$**git remote -v**  och tryck enter.



Du bör se att mappen har koppling till mappen på GitLab.



Clone

Add, Commit, Push

(Pull) Fetch, merge

Egen branch för att jobba på ? ansöka om merge, tex på common och en som är ansvarig

Issues

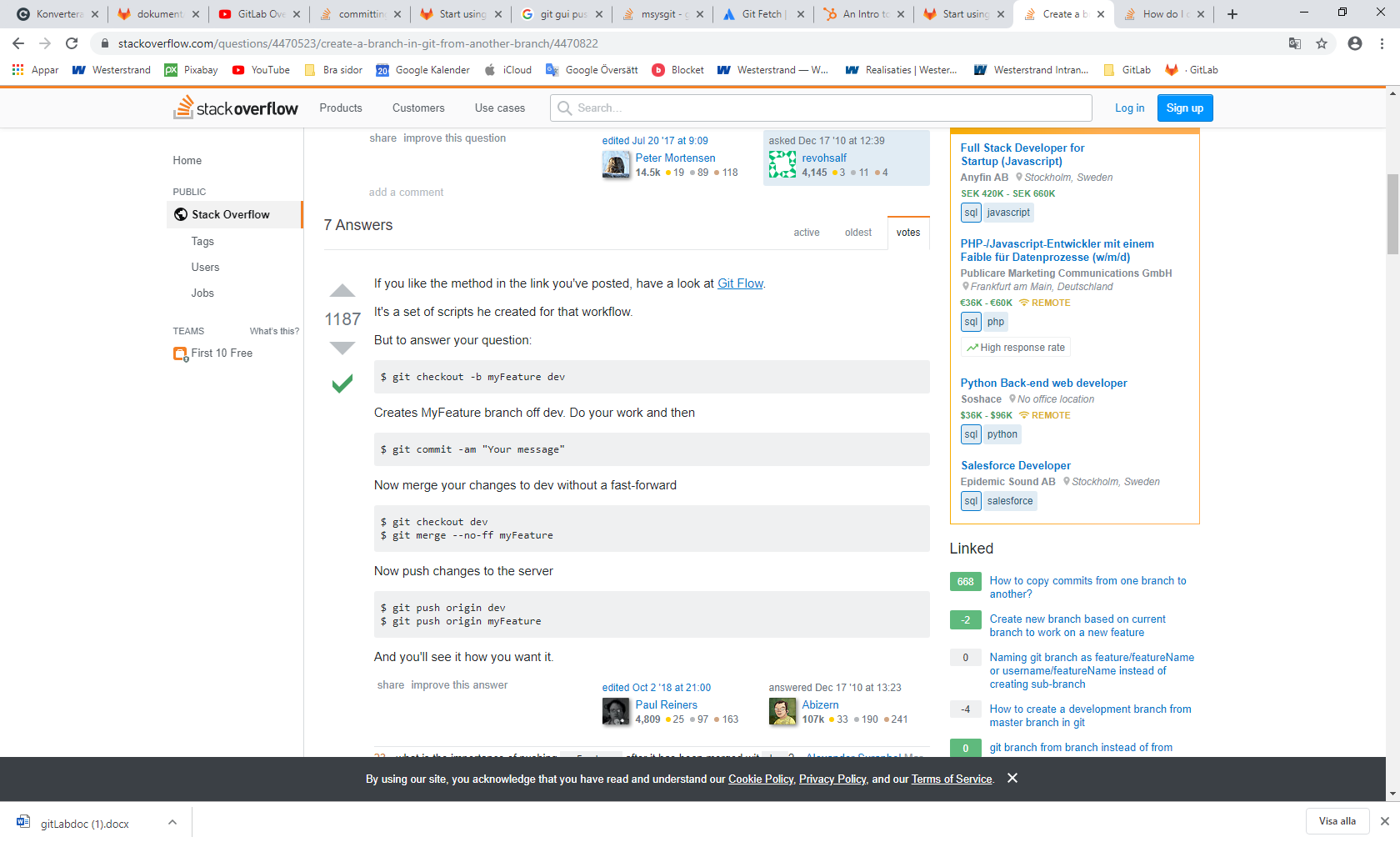
Hämta alla ändringar från master innan arbete påbörjas

Git bash

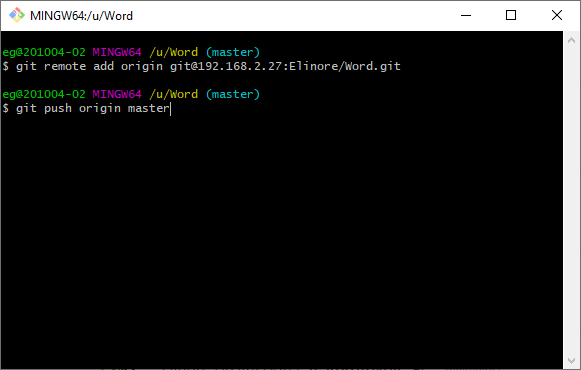
$git pull origin master

GIT GUI

Är det alltid bara en som kan spara direkt till master

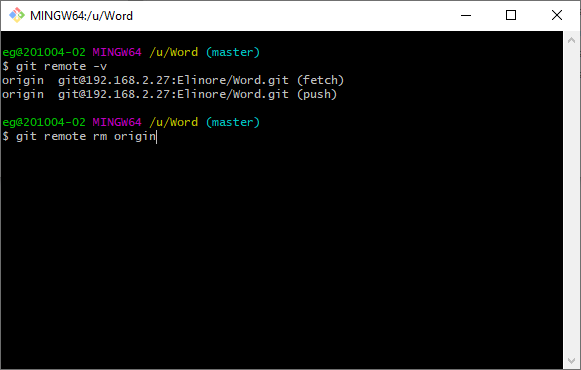


Skapa ett nytt projekt

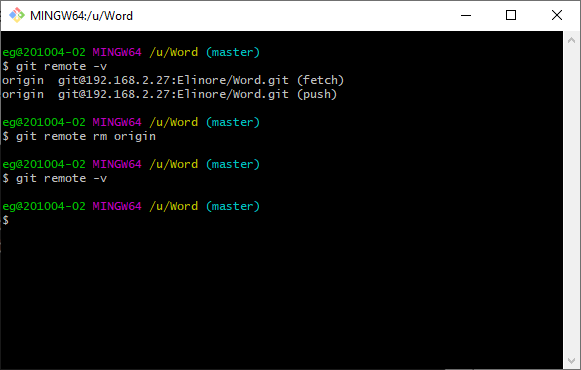


Push utan att mapp skapats innan på gitlab = nja.

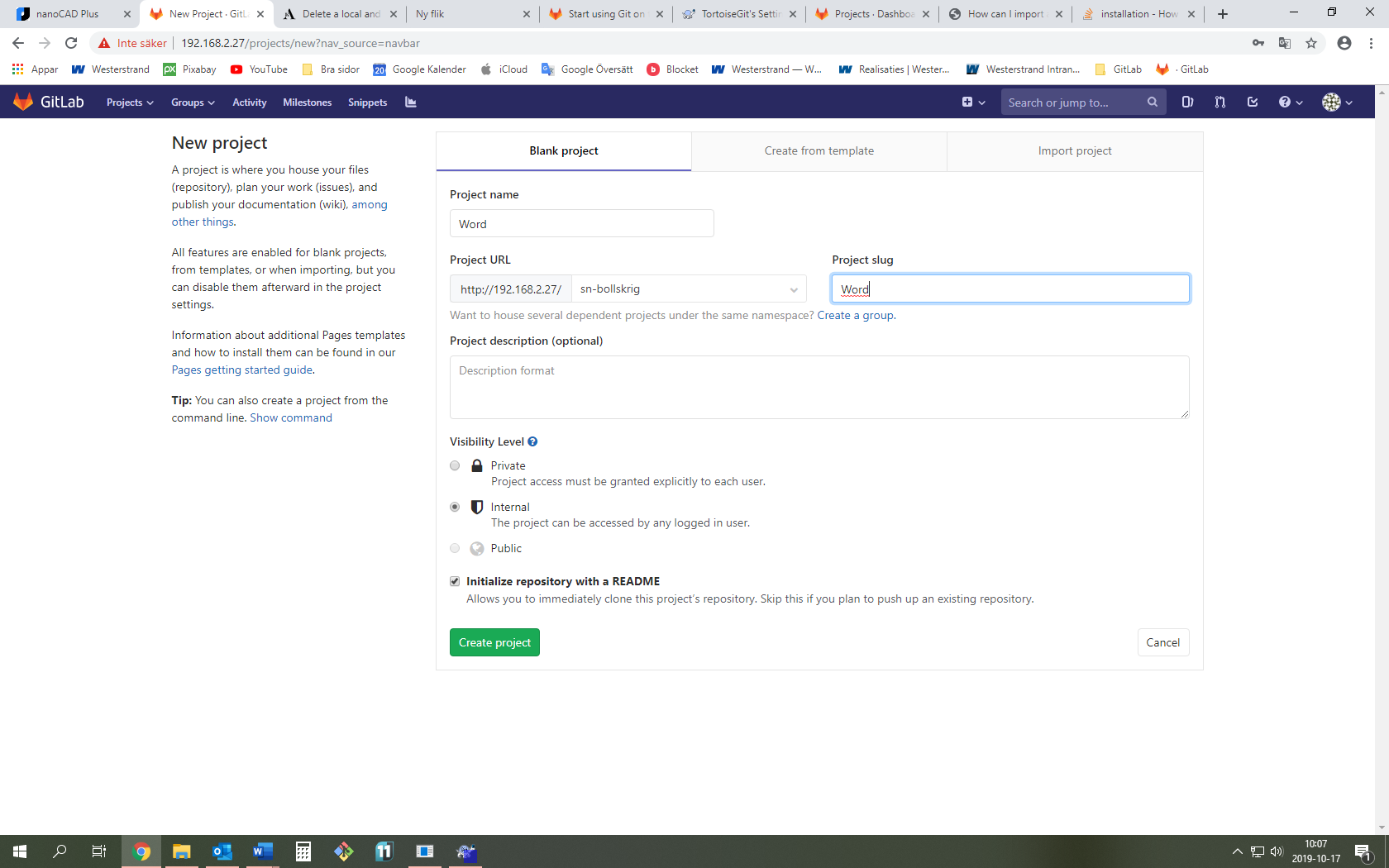
Ta bort felaktig remote länk som lagts till:



Verifiera att den tagits bort



Skapa projektmapp på GitLab först.



Högerklicka på din lokala mapp, välj GitBash here

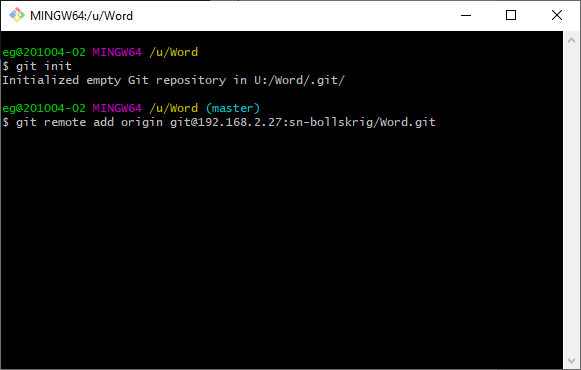
Skriv in

$git init

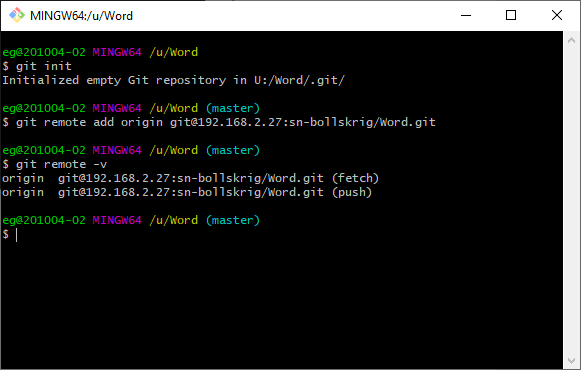


Lägg till länk till mappen du skapat på GitLab:

git remote add <source-name> <repository-path>



[git@192.168.2.27:sn-bollskrig/Word.git](mailto:git@192.168.2.27:sn-bollskrig/Word.git)



Verifiera att ”remoten” lagts in.