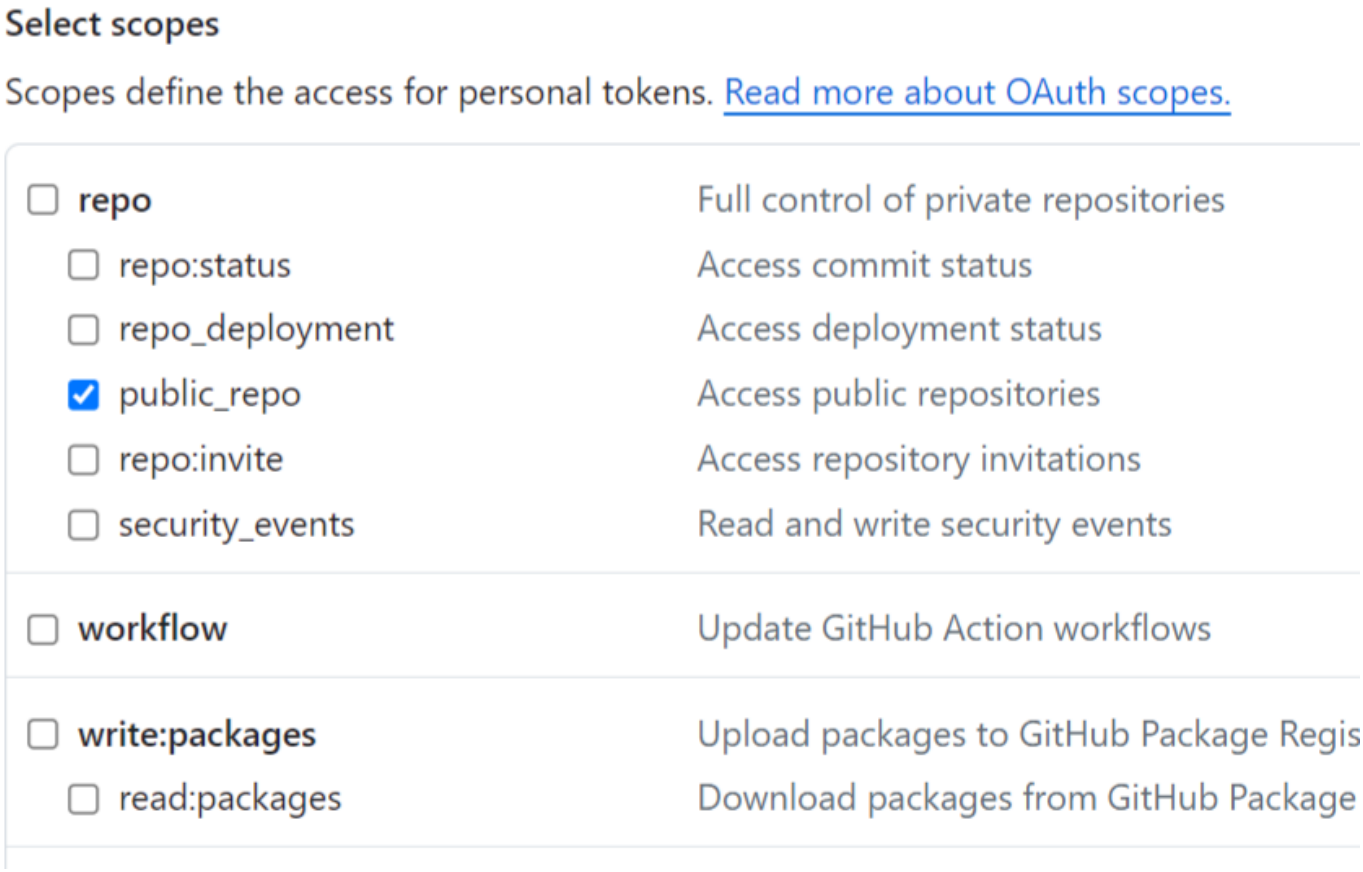
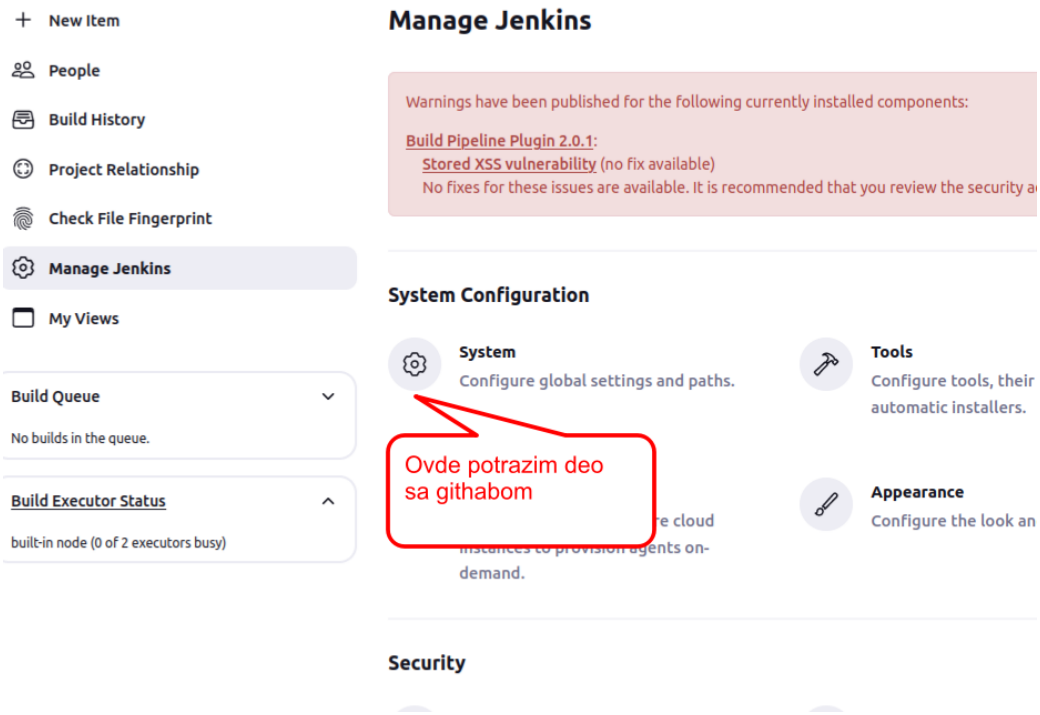
Kako se povezati Jenkins sa svojim githab-nalogom

Generisaćemo token na nasem git hab nalogom I od ponudjenih opcija uzećemo samo

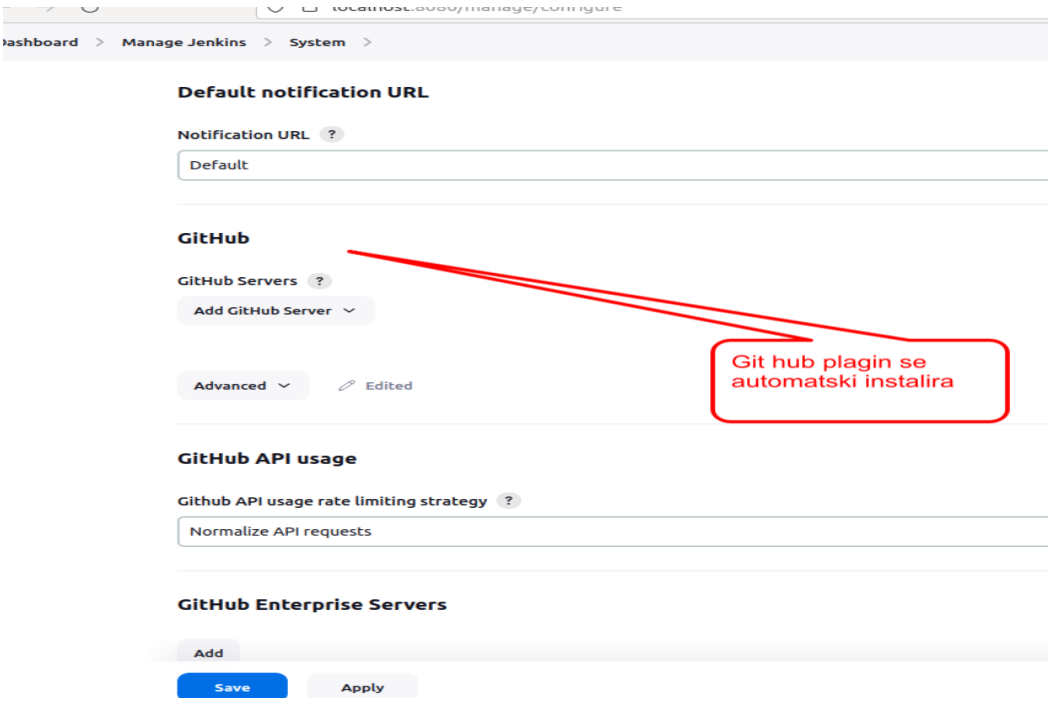


Vratim se sada u moj Jenkins i idem u Manage Jenkins pa u Configure System.

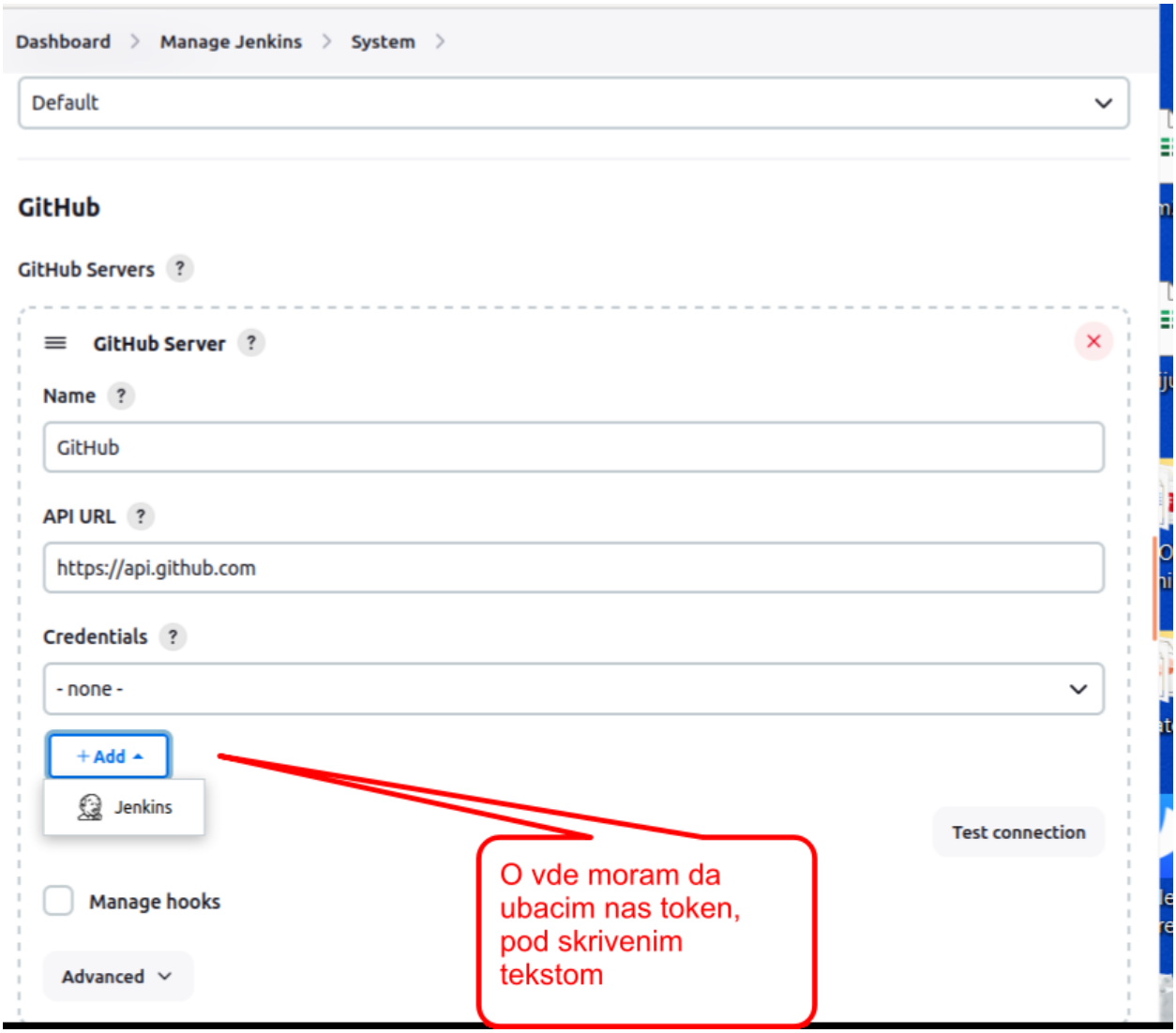
Idem dole sve do odseka sa Githabom.

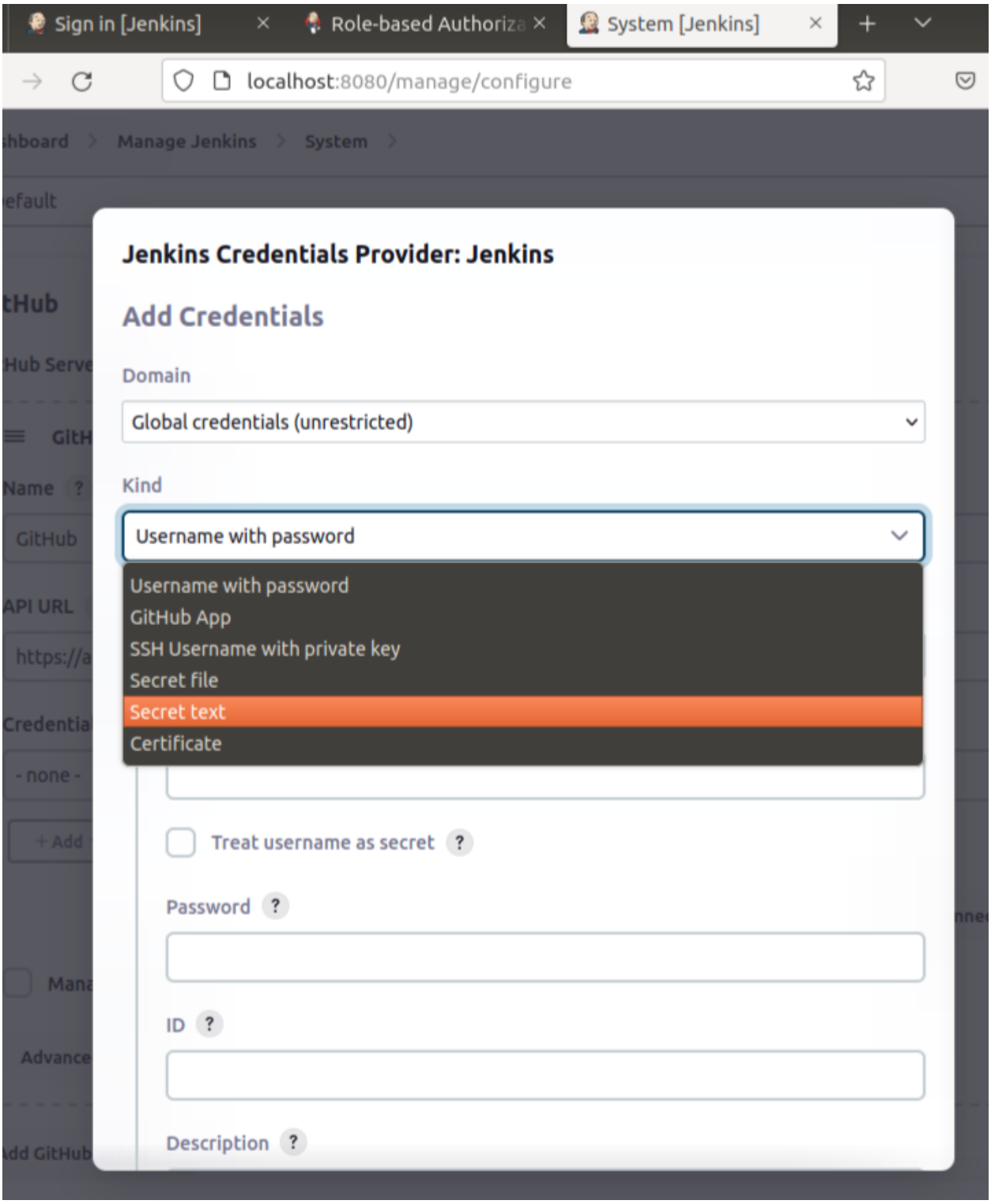


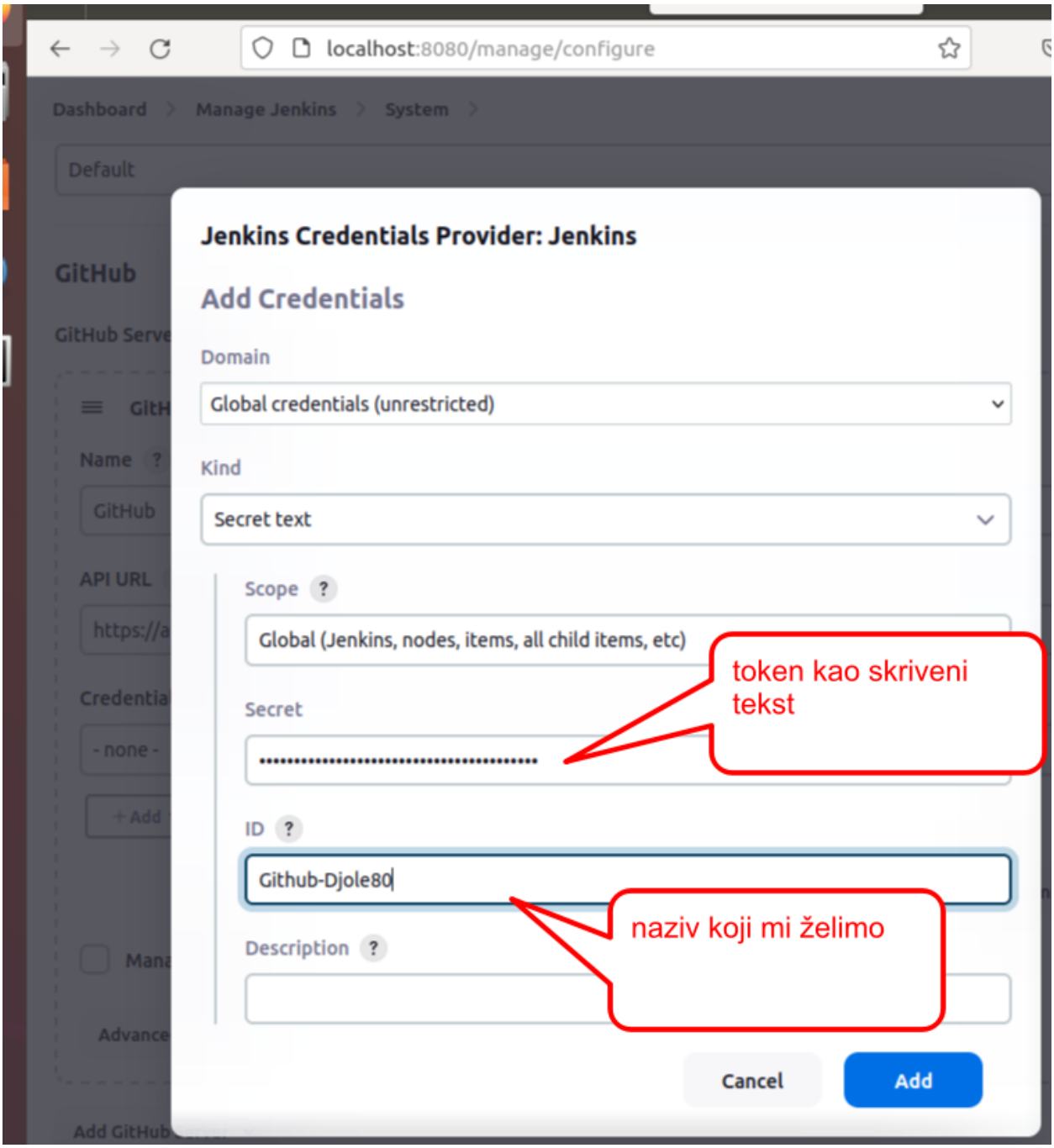
Kada instalirate predložene plaginove vi ste instalirali i plagin za github



Kada dodjemo da izaberemo vezu sa našim nalogom na github-u izabracemo skriveni tekst gde ćemo token iskoristiti za povezivanjem sa našim nalogom.


Dashboard
Manage Jenkins
System
Default
GitHub
GitHub Servers
GitHub Server
Name
GitHub
API URL
https://api.github.com
Credentials
- none -
+ Add
Jenkins
Test connection
Manage hooks
Advanced
O vde moram da ubacim nas token, pod skrivenim tekstom


Sign in [Jenkins]
Role-based Authoriza
System [Jenkins]
localhost:8080/manage/configure
Manage Jenkins
System
Jenkins Credentials Provider: Jenkins
Add Credentials
Domain
Global credentials (unrestricted)
Kind
Username with password
Username with password
GitHub App
SSH Username with private key
Secret file
Secret text
Certificate
Treat username as secret
Password
ID
Description

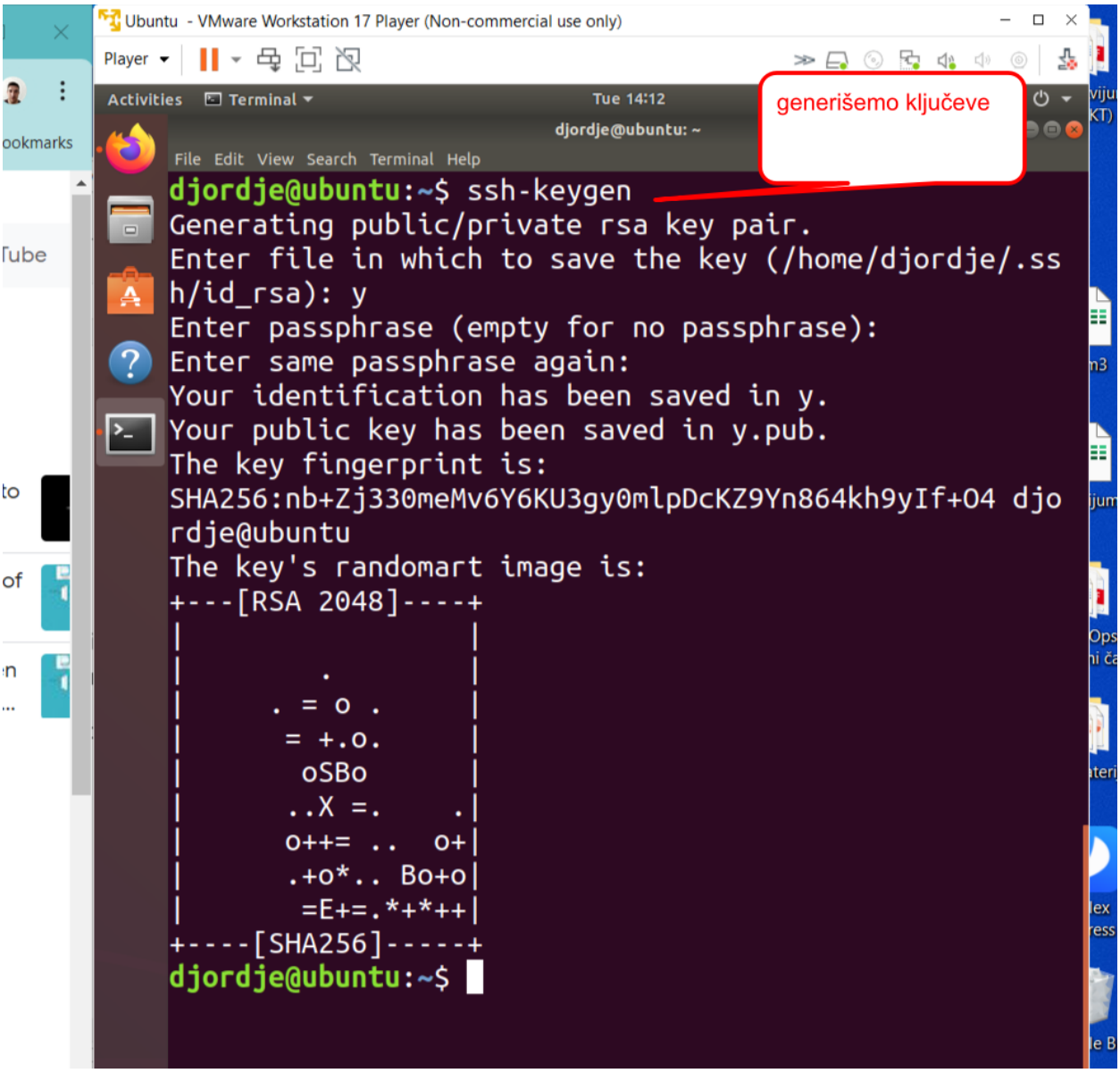


Proveriomo sada podešavanjem na našu konekicju da li ona rati . To meožemo pritiskom na test conection taster.

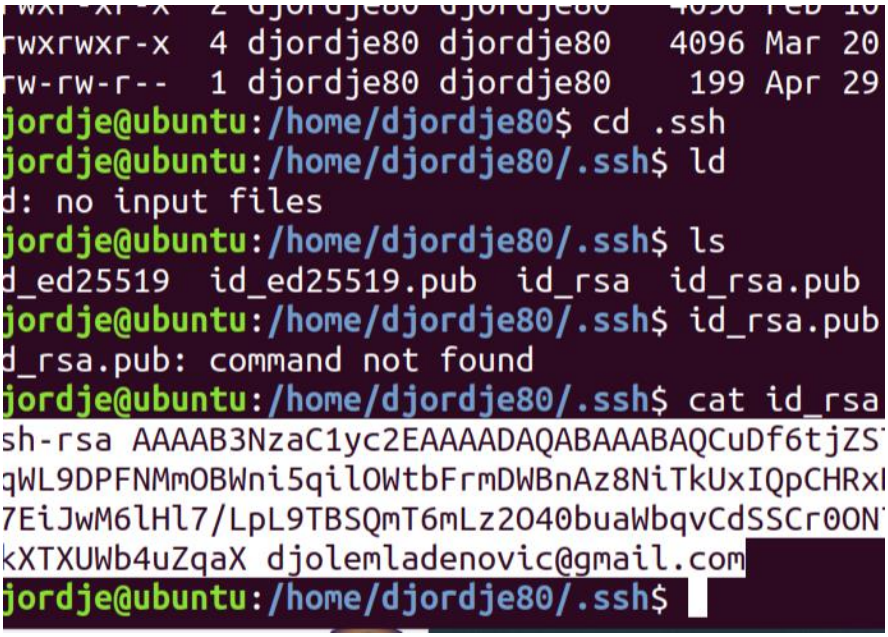
Sledeće što ćemo uraditi je da se povežemo sa SSH ključem

Na taj način moćićemo da cloniramo repozitorijum na Jenkinso lokalnom prostoru. Takođe gde god moj Github ima pristum imaće i Jenkins.

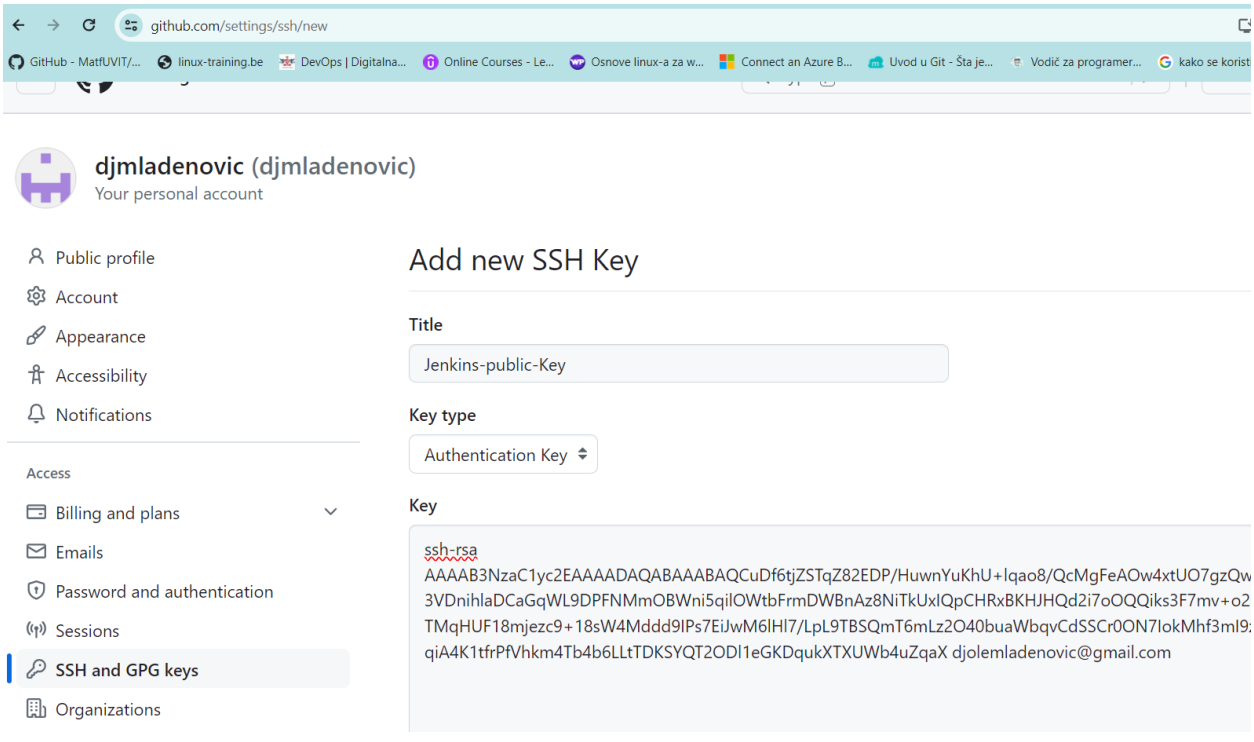
Da vidimo kako to možemo , moramo prvo u terminalu da generišemo privatni i javni ključ ssh.



Da vidimo sada javni (public key) ključ koji je



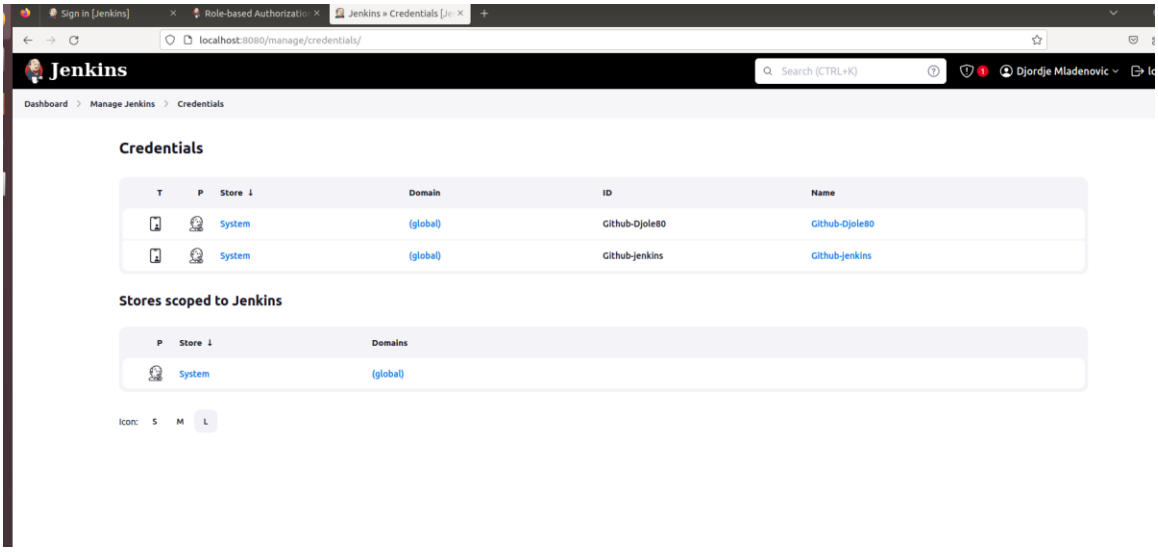
Sada idem na moj Github nalog da postavim taj javni ključ.

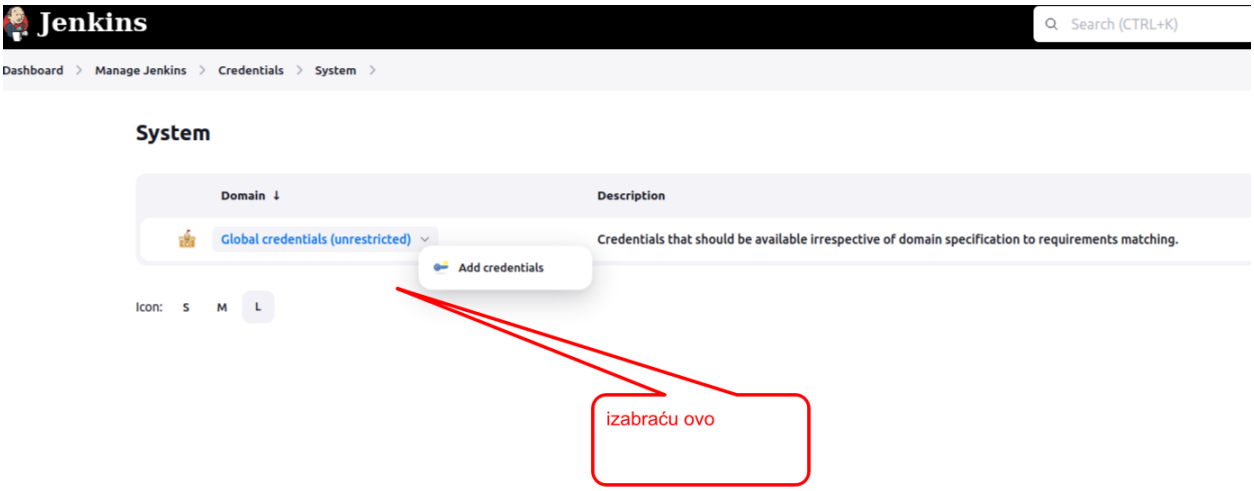


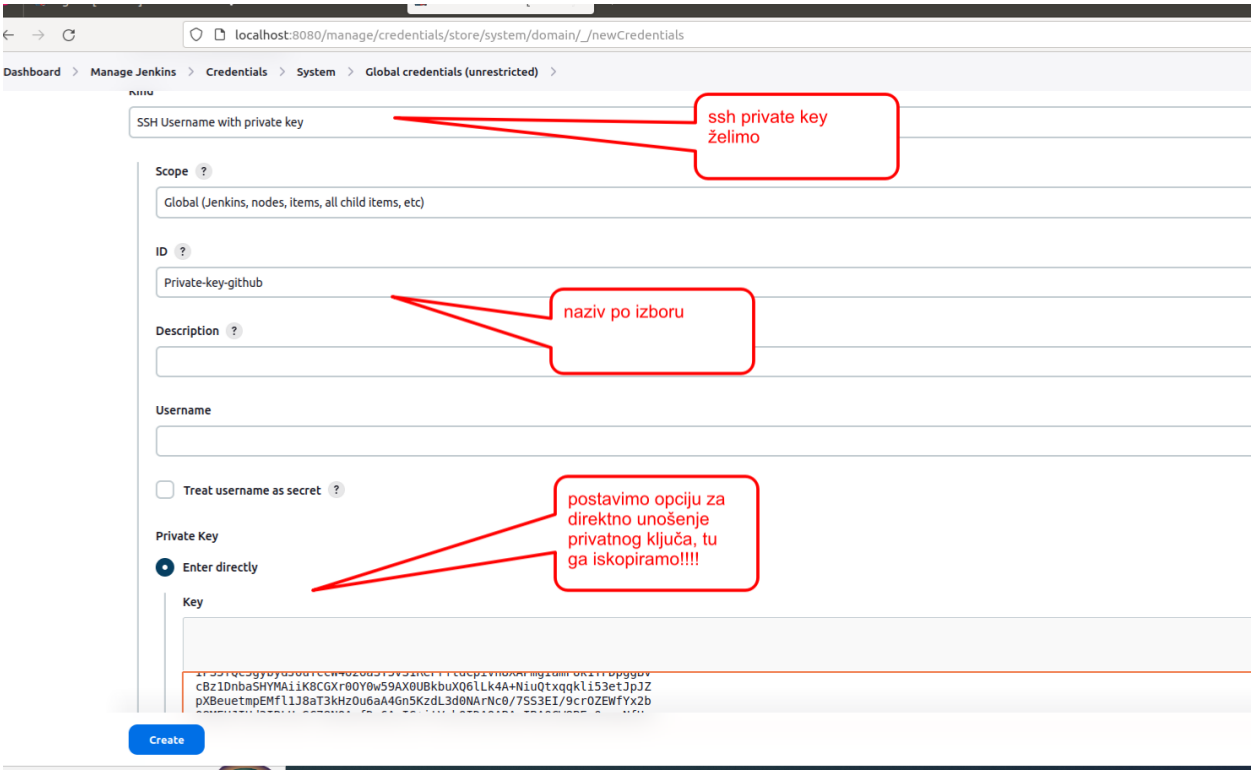
Idemo sada da iskopiramo naš private key u Jenkinsu gde treba.

```
GsVjIGJS22JYGhcSxs22/Tqq+8g9Ll9BzmDJvJ0mi5hxxbijahpJPm9LVF4Wt+10KmJWHx
GRu5dDqUuTgD42K5C3GqqSWLnd60mklmlcF6562akQx+XUnxpPeQfM67poDgafkrN0vd3Q
oDIgb6K1XJt djordje@ubuntu
rdje@ubuntu:~/.ssh$ ^C
rdje@ubuntu:~/.ssh$ cat id_rsa
---BEGIN RSA PRIVATE KEY-----
EpQIBAAKCAQEA3hARovKwQJCoEGXVoVF5tfLVr3xl0N/Euv2vBl0eRZttWDPu
4et8oqHBrwB39C5Ra/wXI7wjvD9No+a/m5ochrFYyBiUttiWBoXEsbNtv06qvv
55fQc5gybydJouYccW4o2oaST5vS1ReFrftdCpiVh8XAPmgIamFUk1YFDpggBv
:1DnbaSHYMAiiK8CGXr0OY0w59AX0UBkbuXQ6lLk4A+NiuQtxqqkli53etJpJZ
3euetmpEMfl1J8aT3kHzOu6aA4Gn5KzdL3d0NArNc0/7SS3EI/9crOZEWfYx2b
1EHJIHd3IRLHoSC78NOAefDw6AyIG+itVybQIDAQABAoIBAQCW8BEs0wzeNfUm
F/HyQ9ejuotkN2gEL2XuySxMSIx6bBJztXFLRodG++/shcMDJGCPp6AK2BzqHZ
pPecSlpNa0pPZ/XdesYu1Mnnh3B3cn84nsuqtBwqFi9hwjCZGxQGPbxjHMlidy
/Zm1sn+BbkllKhML2CawOEwGgZeGGOOoL67vkEsyWobv7O1gd6/AfXLkU3Pul7
```

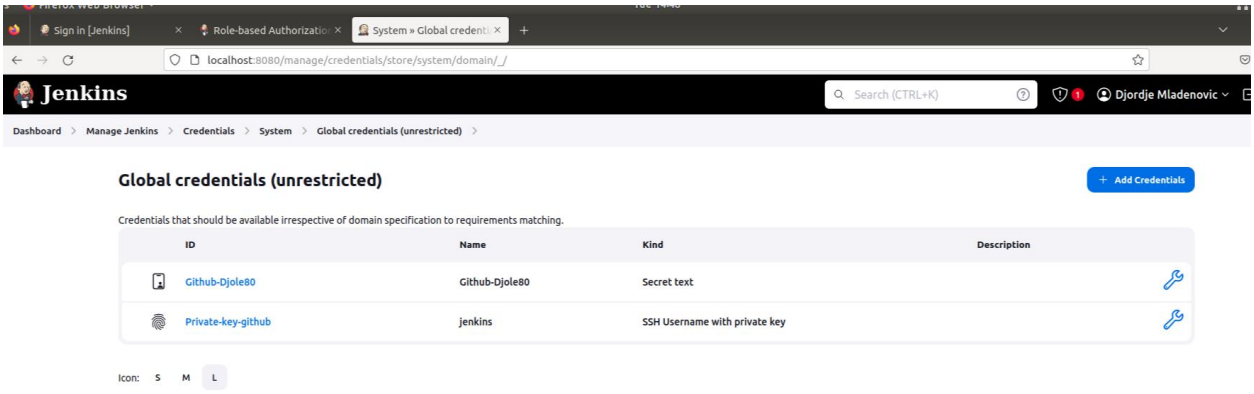
Sada u Jenkinsu idemo u **Manage Jenkins** pa u deo o sigurnosti pronađemo **Manage Credentials**



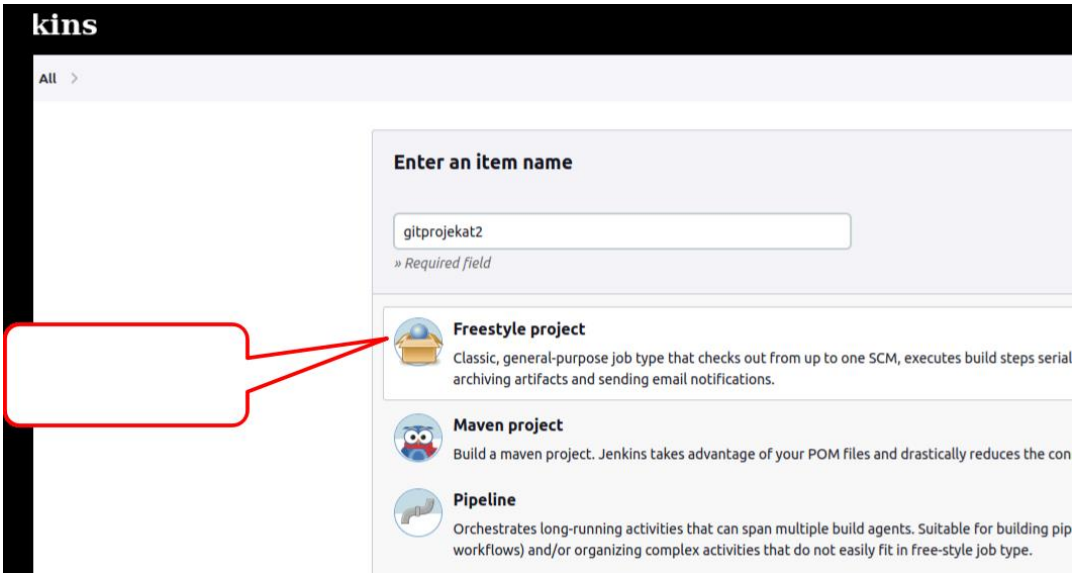




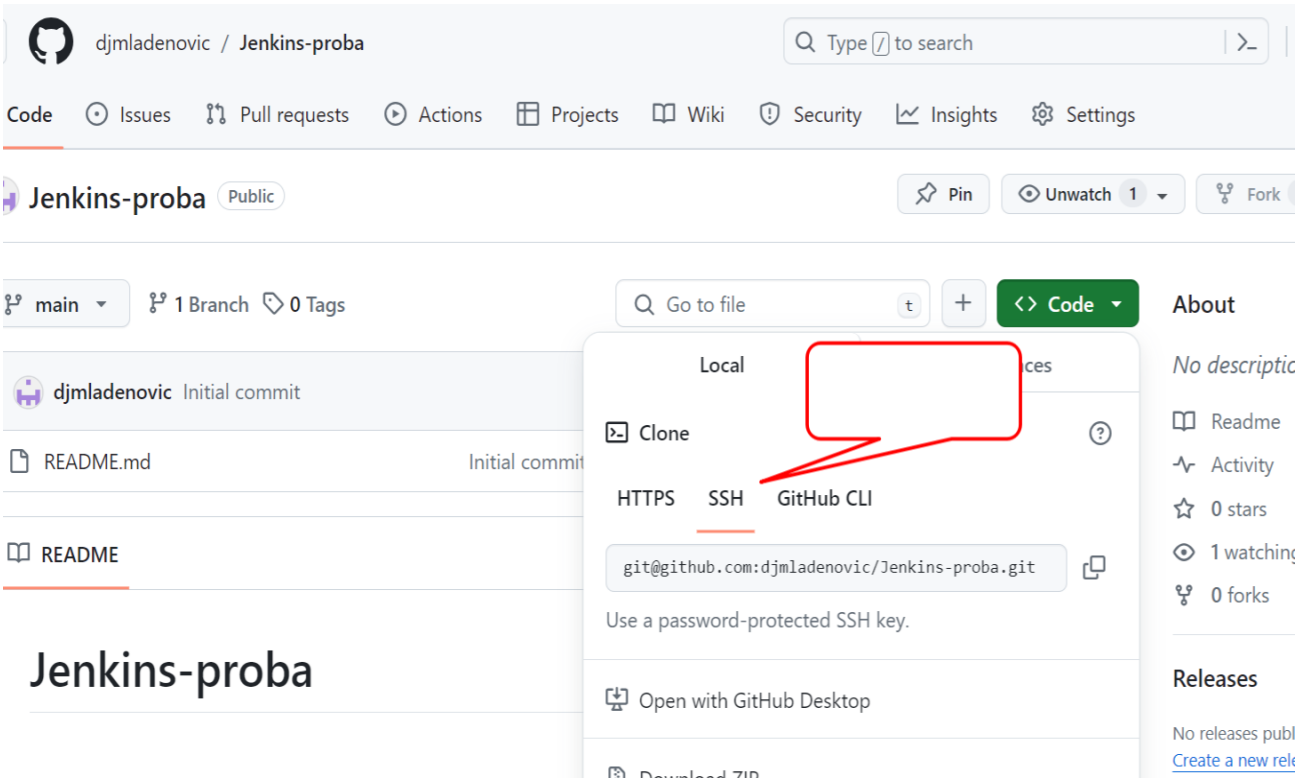
I dobili smo željene konekcije sa github-om



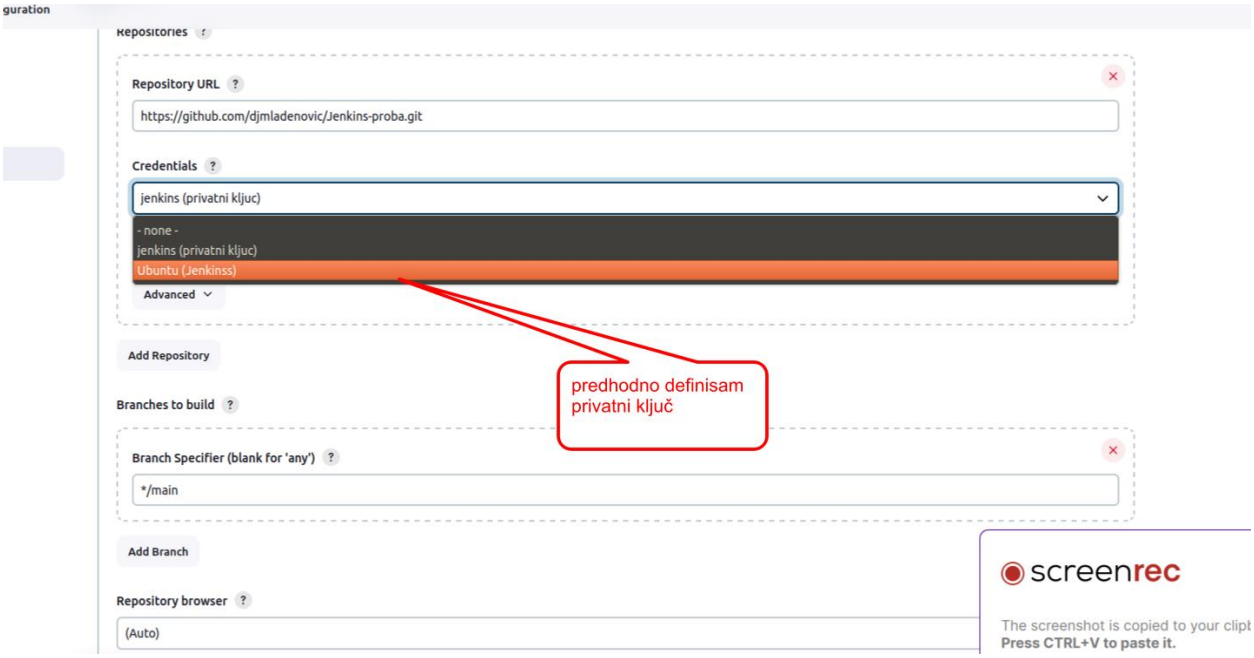
On bi sada trebao da može da klonira repositorijum uz pomoć ssh ključa.



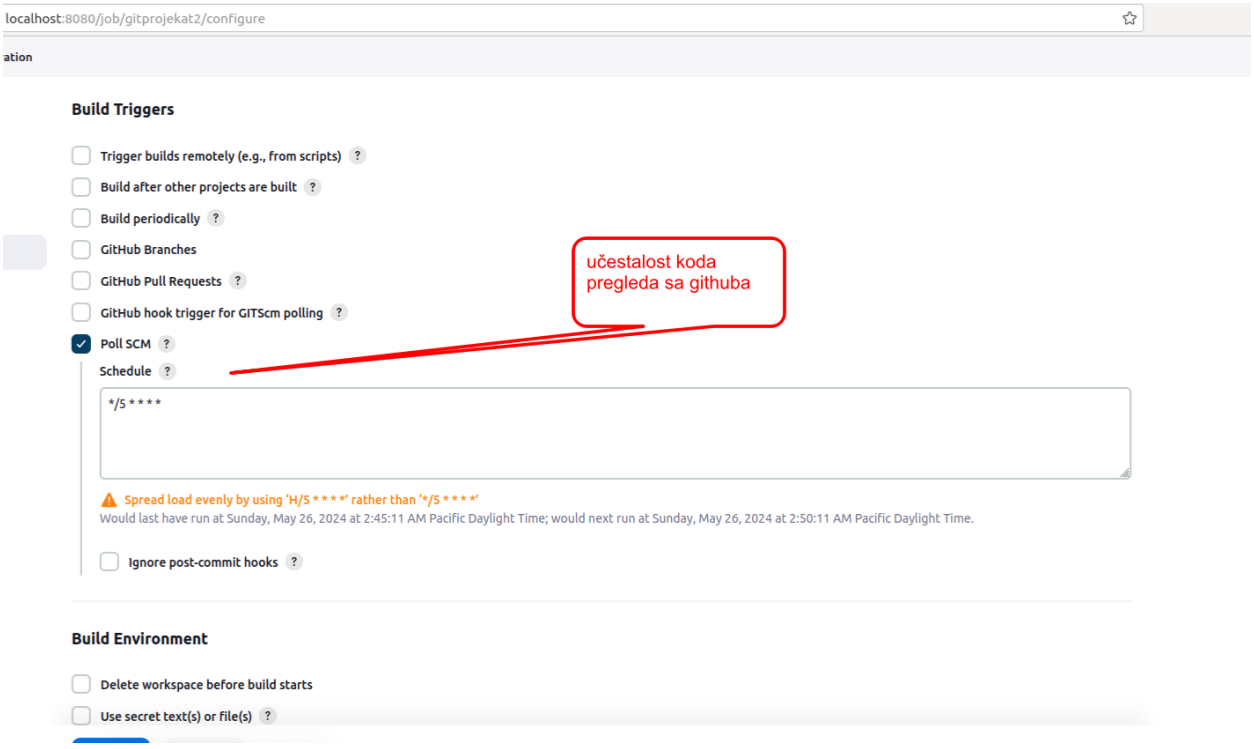
Koristicu na githubu neki prosti primer gde imam samo jedan readme file. Uzeću njegovu ssh konekciju i to ću iskopirati.



U postavkama iz priloženog obratite pažnju na branch koji postavljate. Postavljen je master koji morate promeniti na main. Naravno zavisi gde je željeni program koji želite da preuzmete na Jenkinsu tj. Na kojij grani se nalazi željena verzija .



Dodali smo i opciju (* * * * *) sa kojom možemo pratimo promene koje se dešavaju na GitHubu i njih bildujemo. Učestalost zavisi od nas i našeg podešavanja.



I kao što vidimo bild radi.

Console Output


```
Started by user Djordje Mladenovic
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/gitprojekat2
The recommended git tool is: NONE
using credential kljuc
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/gitprojekat2/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/djmladenovic/Jenkins-proba.git # timeout=10
Fetching upstream changes from https://github.com/djmladenovic/Jenkins-proba.git
 > git --version # timeout=10
 > git --version # 'git version 2.17.1'
using GIT_SSH to set credentials privatni kljuc
Verifying host key using known hosts file
You're using 'Known hosts file' strategy to verify ssh host keys, but your known_hosts file does not exist, please go to 'Manage Jenkins' -> 'Security' -> 'Git Host Key
Verification Configuration' and configure host key verification.
 > git fetch --tags --progress -- https://github.com/djmladenovic/Jenkins-proba.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 3747524ed59d77092a1d1a893ba91f98788595f3 (refs/remotes/origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 3747524ed59d77092a1d1a893ba91f98788595f3 # timeout=10
Commit message: "Add files via upload"
 > git rev-list --no-walk 3747524ed59d77092a1d1a893ba91f98788595f3 # timeout=10
Finished: SUCCESS
```