

Elektro in računalniška šola, Šolski center Ptuj

**Poročilo pri predmetu**  
**VARNOST IN ZAŠČITA V INFORMATIKI**

**GITHUB**

**Žiga Plejnšek**

**Mentor: Hana Hrnjić**

Ptuj, september 2025

## KAZALO VSEBINE

<b>1</b>	<b>Uvod.....</b>	<b>3</b>
<b>2</b>	<b>OPIS GITHUBA.....</b>	<b>3</b>
<b>3</b>	<b>POSTOPEK NALAGANJA NA GITHUB.....</b>	<b>4</b>
<b>4</b>	<b>TEŽAVE.....</b>	<b>6</b>
<b>5</b>	<b>Ugotovitve.....</b>	<b>6</b>
<b>6</b>	<b>ZAKLJUČEK.....</b>	<b>7</b>
<b>7</b>	<b>Viri.....</b>	<b>7</b>

## 1 UVOD

V tej vaji sem se učil, kaj je GitHub in kako ga lahko uporabljamo. Moj cilj je bil narediti račun na GitHubu in ustvariti repozitorij z imenom VZI praksa. Na njega sem naložil vsa poročila ki sem jih naredil do zdaj vključno z to nalogo. Nato pa v to nalogo vstavil poročilo ki sem ga ustvaril.

## 2 OPIS GITHUBA

Kaj je GitHub?

GitHub je spletna stran, na kateri lahko shranjuješ programske projekte in sodeluješ z drugimi ustvarjalci. Temelji na sistemu Git, ki sledi vsem spremembam v datotekah. To pomeni da lahko kdaj koli pogledaš spremembe v repozitoriju.

Kako deluje?

Deluje tako da spremembo v datoteki shranimo kot commit. Vse te committe pošljemo z ukazom push. Če več ljudi dela na istem projektu, vsak dela na svojem branchu, nato pa svoje spremembe združijo v glavno vejo z merge.

Zakaj se uporablja?

GitHub se uporablja za varnostno kopiranje vseh datotek v oblaku, za pogled v zgodovino sprememb in da več ljudi lahko dela na istem projektu brže da bi si med seboj pokvarili kaj v kodi.

Kaj je merge?

Merge pomeni združitev sprememb iz ene veje v drugo. Na primer če daš novo funkcijo v veji login jo z mergom dodaš v glavno vejo main. To pa je zelo dobro pri skupinskem delu, ker se koda preveri preden pride v glavni projekt. [1], [2]

### 3 POSTOPEK NALAGANJA NA GITHUB

Najprej sem odprl Ubuntu v VirtualBoxu in v njem ustvaril mapo z imenom VZI praksa, kamor sem shranil vsa dosedanja poročila in prazno poročilo te vaje.

Nato sem v terminalu vpisal ukaz, da sem se premaknil v to mapo:  
`cd "/home/migi1212/Documents/VZI praksa"`

Tam sem inicializiral nov Git repozitorij z ukazom: `git init`

Z ukazom sem dodal vse datoteke v repozitorij: `git add .`

Potem sem naredil prvi commit, kjer sem shranil trenutno stanje projekta: `git commit -m "Prvi commit: dodana poročila in prazna vaja 2"`

Na GitHubu sem ustvaril repozitorij z imenom VZI-praksa, nato pa ga povezal z lokalnim repozitorijem v Ubuntu-ju: `git remote add origin https://github.com/migi13252/VZI-praksa.git`

Ker GitHub ne podpira več prijave z navadnim geslom, sem moral ustvariti Personal Access Token (PAT) v nastavitvah na GitHubu (Settings → Developer settings → Personal Access Tokens → Tokens (classic)). Ta token sem potem uporabil namesto gesla pri prvem `git push`.

Nato sem izvedel ukaze: `git branch -M main` `git push -u origin main`

S tem sem uspešno naložil vse datoteke na GitHub. Ko sem kasneje dopolnil poročilo z razlago o GitHubu, sem ponovil postopek: `git add .` `git commit -m "Dodan teoretični del o GitHubu"` `git push`

Po drugem pushu sem na GitHubu videl oba commita – prvega s praznim poročilom in drugega z dopolnjenim teoretičnim delom.

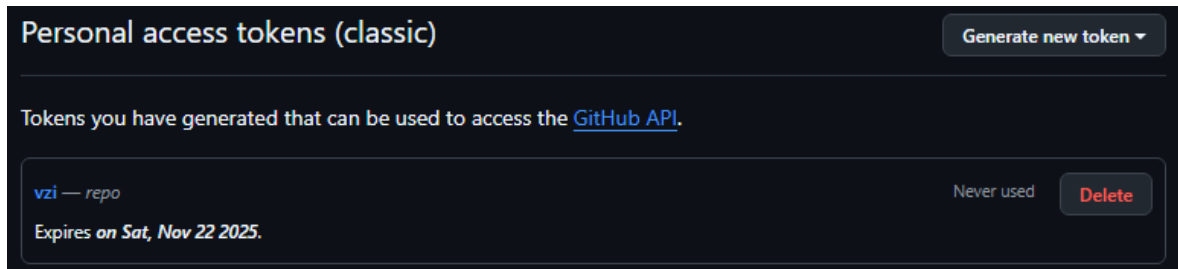
```
migi1212@ZIGA:~$ ^C
migi1212@ZIGA:~$ sudo apt update
[sudo] password for migi1212:
Hit:1 http://si.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://si.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://si.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
80 packages can be upgraded. Run 'apt list --upgradable' to see them.
migi1212@ZIGA:~$ sudo apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.43.0-1ubuntu7.3).
The following package was automatically installed and is no longer required:
  libllvm19
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 80 not upgraded.
migi1212@ZIGA:~$
migi1212@ZIGA:~$ git --version
git version 2.43.0
migi1212@ZIGA:~$
```

```
e
migi1212@ZIGA:~/Documents/VZI praksa$ git add .
migi1212@ZIGA:~/Documents/VZI praksa$ git commit -m "Test push s tokenom"
On branch main
nothing to commit, working tree clean
migi1212@ZIGA:~/Documents/VZI praksa$ git push -u origin main
Username for 'https://github.com': migi13252
Password for 'https://migi13252@github.com':
remote: Invalid username or token. Password authentication is not supported for
Git operations.
fatal: Authentication failed for 'https://github.com/migi13252/VZI-praksa.git/'
migi1212@ZIGA:~/Documents/VZI praksa$ git push -u origin main
Username for 'https://github.com': migi13252
Password for 'https://migi13252@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 408.76 KiB | 18.58 MiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/migi13252/VZI-praksa.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
migi1212@ZIGA:~/Documents/VZI praksa$
```

## 4 TEŽAVE

Pri tej vaji sem imel nekaj težav.

Ko sem hotel uporabiti ukaz `git push`, sem dobil napako “Permission denied (403)”. GitHub mi ni dovolil naložiti datotek, ker ne podpira več prijave z navadnim geslom. Težavo sem rešil tako, da sem ustvaril Personal Access Token (PAT) in ga uporabil namesto gesla. Po tem je ukaz `git push` deloval brez težav.



Imel sem tudi težavo, ker sem najprej poskušal pushati na vejo `main`, moj repozitorij pa je uporabljal vejo `master`. To sem popravil z ukazom `git branch -M main`, potem pa je vse delovalo normalno.

Kasneje sem opazil, da se repozitorij ni povezal, ker sem imel na koncu povezave do GitHub repozitorija presledek. Ko sem URL ponovno pravilno kopiral, je povezava delovala.

Ko sem vse te napake odpravil, sem lahko uspešno naložil datoteke na GitHub.

## 5 UGOTOVITVE

Pri vaji sem se naučil, kako deluje GitHub in kako pomembno je, da se spremembe redno shranjujejo in nalagajo.

Po prvem pushu sem imel na GitHubu prazno poročilo, po drugem pa dopolnjeno z vsebino.

V zavihku “**Commits**” sem lahko jasno videl obe verziji datoteke in spremembe med njima.

To je zelo koristno, saj omogoča sledenje spremembam in primerjavo stare ter nove verzije projekta.

Zdaj razumem, zakaj razvijalci uporabljajo GitHub pri skupinskem delu – omogoča preglednost, sodelovanje in varnostno kopijo vseh datotek.

## 6 ZAKLJUČEK

GitHub sem uspešno nastavil in na njega naložil obe poročili. Najprej prazno nato pa dopolnjeno s teorijo. Kljub nekaj težavam sem uspel vse pravilno prenest. Iz te vaje sem se veliko naučil, saj me je zanimalo, kako razvijalci skupaj sodelujejo pri projektih in kako lahko shraniš svoje delo na GitHubu.

## 7 VIRI

[1] GitHub Documentation. *About GitHub and Git*. [spletno]. Dostopno na: <https://docs.github.com>

[2] Atlassian. *What is Git?* [spletno]. Dostopno na: <https://www.atlassian.com/git/tutorials>