

GitHub un versiju kontrole

Darba veids: Patstāvīgā izpēte un praktiskais darbs

Sasniedzamais rezultāts:

1. Izmantota versiju kontroles sistēma *Git* un *GitHub* projekta failu glabāšanai un pārvaldībai.
2. Dokumentētas projekta izmaiņas ar *commit* un *push* darbībām, ievērojot strukturētu darba procesu.
3. Darbības pierādītas ar ekrānuzņēmumiem (*screenshots*).

Stundas tēmas apguvei ieteicamie video:

1. <https://www.youtube.com/watch?v=YX0RUxH4-tc&t=49s>
2. <https://www.youtube.com/watch?v=el3Fk4wb3dU&t=26s>
3. <https://www.youtube.com/watch?v=PhG72KirgLo>

Driķst izvēlēties arī citrus, tēmai atbilstošus video vai citas pamācības.

1. uzdevums. Izpēte. Skaties norādītos video un atbildi uz jautājumiem.

Jautājumi, uz kuriem jāsniedz atbildes:

1. Kāda ir atšķirība starp *Git* un *GitHub*?
2. Kādas trīs svarīgas komandas vai darbības tika demonstrētas videoklipā (piemēram, *commit*, *push*)?
3. Kāpēc ir svarīgi dokumentēt projekta izmaiņas (piemēram, *commit* ziņojumos) strādājot ar versiju kontroli?

2. uzdevums. Praktiskā daļa. Izveido savu *GitHub* profilu, repozitoriju un izmanto *GitHub* (vai *GitHub Desktop*) lietotni, lai praktiski augšupielādētu datnes, veiktu *commit* un *push* darbības. Katram posmam pievieno ekrānuzņēmumu!

1. solis – *GitHub* profila izveide

1. Atver vietni <https://github.com>
2. Klikšķini **Sign up** → izveido kontu (e-pasts, lietotājvārds, parole).
3. Apstiprini e-pastu un ieej profilā.
4. Ja vēlies, vari pievienot profila attēlu.

Ekrānšāviņš Nr. 1: Mans izveidotais *GitHub* profils (redzams lietotājvārds un profila lapa).

2. solis – Jauna repozitorija izveide

1. Augšējā labajā stūrī nospied + → **New repository**.
2. Ievadi nosaukumu, piemēram:
12kl_git_izpete_TavsVards.
3. Atzīmē opciju “**Initialize this repository with a README**”.
4. Klikšķini **Create repository**.

Ekrānšāviņš Nr. 2: Izveidotais repozitorījs (redzams repozitorija nosaukums un README fails).

3. solis – Failu augšupielāde *GitHub* tīmeklī

1. Atver savu repozitoriju *GitHub*.
2. Klikšķini **Add file** → **Upload files**.
3. Velc vai izvēlies failu (piem., **projekts.txt**, **index.html** u.c.).
4. Ieraksti *commit* ziņojumu, piemēram:
“*Pievienots pirmais projekta fails*”.
5. Klikšķini **Commit changes**.

Ekrānšāviņš Nr. 3: *Commit log* ar pirmo pievienoto failu un commit ziņojumu.

4. solis – Darbs ar *JitHub* vai *GitHub Desktop*

1. Atver lietotni **JitHub** vai **GitHub Desktop** un pieslēdzies savam GitHub kontam.
2. Izvēlies **Clone repository** → lejupielādē savu repozitoriju uz datoru.
3. Veic izmaiņas failos lokāli (piem., pievieno rindkopu **README.md**).
4. Lietotnē ieraksti commit ziņojumu, piem.:
“*Atjaunināts README ar secinājumiem*”.
5. Klikšķini **Push origin**, lai izmaiņas nosūtītu uz GitHub.

Ekrānšāviņš Nr. 4: *GitHub Desktop / JitHub logs* ar commit un push darbībām.

Ekrānšāviņš Nr. 5: *GitHub* tīmekļa lapa ar redzamu jauno commit vēsturi.

5. solis – Pārbaude un dokumentēšana

Pārbaudi, vai:

- redzami **vismaz 2 commit** ar skaidriem ziņojumiem;
- faili ir **atjaunināti** un redzami *GitHub*;
- **ekrānšāviņi** pievienoti dokumentā (PDF/Word).

3. uzdevums. Refleksija. Atbildi uz jautājumiem!

1. Kādas grūtības radās, izmantojot *GitHub* pirmo reizi?
2. Ko Tu iemācījies par versiju kontroli?
3. Kā *GitHub* varētu noderēt nākotnes darbā vai studijās?

Vērtēšanas kritēriji

Kritērijs	Punkti
Atbildes uz ievada jautājumiem (izpratne par <i>Git</i> un <i>GitHub</i>)	2
<i>GitHub</i> profils un repozitorijs izveidots (ar ekrānuzņēmumu pierādījumiem)	2
Veikti vismaz 2 <i>commit</i> ar skaidriem ziņojumiem	2
Izmaiņas veiksmīgi nosūtītas uz <i>GitHub</i> (<i>push</i>) un redzamas	2
Refleksija + vismaz 5 ekrānšāviņi pievienoti dokumentā	2
Kopā	10 punkti