

# Kollaboration med Git och Github

## Övning 1: Samarbete med PR (Pull request)

Träna på att versionshantera i grupp.

Med PR kan man jobba i ett flöde där man ständigt granskar, diskuterar och justerar koden innan det mergas in i main-branch. Det hjälper gruppen att skriva bättre kod som är mindre buggig, samtidigt som man lär sig ifrån varandra.

1. En i gruppen skapar en ny projektmapp med filen index.html lokalt på sin dator
2. Versionshantera det nya projektet lokalt. Dvs initiera git och gör första commiten
  - a. `git init`
  - b. `git add .`
  - c. `git commit -m "First commit"`
3. Samma gruppmedlem skall nu skapa ett nytt tomt repo i sin github
4. Länka lokala repot med det nya github repot
  - a. `git branch -M main`
  - b. `git remote add origin <din github repo url>`
  - c. `git push -u origin main`
5. Kolla att ändringarna har skett i github
6. Samma gruppmedlem måste bjuda in resterande medlemmar som Collaborators, i det nya github repot
7. Alla medlemmar hämtar det nya repot, genom att kлона
  - a. `git clone <github repo url>`
8. **Samarbete börjar: Turas om att göra följande steg:**
9. Stå på main-branch och synka med github
  - a. `git pull origin main`
10. Förgrena ut till egen branch som heter exempelvis. "feature/header"
  - a. `git checkout -b feature/header`
11. Gör en ändring i index.html **och sen versionhantera**
  - a. `git add .`
  - b. `git commit -m "Ditt meddelande"`
12. Skicka upp er branch genom att pusha branchen till github
  - a. `git push origin feature/header`
13. I github kan ni skapa en pull request på er branch, förklara vad det är för PR samt sätt vem som skall granska er PR. Sätt även eventuella taggar för bättre översikt om vilka PR som skall behandlas, granskas, mergas osv
14. Er utvalda granskare skall granska och kommentera er kod. Antingen godkänner medlemmen eller så måste det diskuteras kring eventuella justeringar
15. Var ödmjuka i era diskussioner. Det viktiga är att lära sig från varandra, INTE vinna ett argument. Viktigt att ge positiv feedback varje gång ni granskar/kommenterar.
16. När granskaren har godkänt, kan ni merge er PR till main.
17. Radera er lokala förgrening.
18. Upprepa stegen 9-17 för varje medlem, tills alla känner sig bekväma med processen. Det bästa vore om ni övar tills hela flödet och alla dess kommandon sitter i ryggmärken, så att ni kan fokusera all er energi på att utveckla koden, istället för att haka upp er och irritera er på versionshanteringen. Övningen gynnar er när ni jobbar i projekt, då detta är flödet ni kommer jobba med när ni versionshanterar kodbasen tillsammans.

## Övning 2: Fixa en merge conflict

Framkalla en merge conflict genom att ni ändrar på samma rad. Kom ihåg att alltid hämta senaste versionen från det avlägsna main (git pull) innan ni skickar till det avlägsna repot (git push).

19. Alla hämtar senaste versionen av github projektet
  - a. git pull
20. Alla skapar en egen branch för att jobba på en specifik feature
  - a. Git checkout -b feature/name-of-your-choice
21. Alla ändrar samma fil, på samma rad. Sedan versionshanterar:
  - a. git add .
  - b. git commit -m "*Ditt meddelande*"
22. Medlem 1 skickar upp sin branch
  - a. git push origin feature/name-of-your-choice
23. Skapa en ny PR, granska och merga till main i github
24. Medlem 2 skickar upp sin branch
  - a. git push origin feature/name-of-your-choice
25. Kommer synas i PR i github att det finns en MERGE CONFLICT
26. För att hämta ner merge conflict lokalt för att lösa konflikten, måste man stå på main-branch, synka med github, stå sedan på feature/name-of-your-choice och merga main-branch i feature/name-of-your-choice. Detta framkallar konflikten lokalt som man kan se i editorn
  - a. git checkout main
  - b. git pull
  - c. git checkout feature/name-of-your-choice
  - d. git merge main
27. Diskutera och kom överens om vilken kod som skall behållas, spara och sedan skicka till github
  - a. git add .
  - b. git commit -m "*Ditt meddelande*"
  - c. git push origin feature/name-of-your-choice
28. Resten av medlemmarna repeterar stegen 24-27
29. Repetera 19-28, dvs hela merge conflict flödet. Tills alla i gruppen känner sig bekväma med merge conflicts

## Övning 3: Publicera ert repository i github pages

30. Publicera ert repository i github pages: <https://pages.github.com/>