

Einstaklingsverkefni – Git og GitHub (8%)

GIT spurningar (2%)

Spurningum er svarað í **git_svor.txt** samkvæmt leiðbeiningum hér neðar í „Git verkefni“

GIT útgáfustýring (6%)

1. 3% **Git_verkefni**
 - a. Vinnusvæði vaktað með Git útgáfustjórnun og verkefnið leyst samkvæmt verklýsingu hér neðar.
2. 3% **Github_verkefni**
 - a. https://github.com/VSH24/github_verkefni afritað á eigin *GitHub reikning* (*GitHub Account*) með aðferðinni „**Fork Repository**“. Geymslan (*repository*) er síðan sett á local vinnusvæði með „**Git clone**“ Leysið verkefnið samkvæmt verklýsingu hér neðar.

Allar aðgerðir og skráningar eiga að vera í sögu skjalana á GitHub.
(*Branches / Commits / History*)

GIT spurningar (2%)

Svaraðu spurningunum sem eru í meðfylgjandi skrá: **git_spurningar.txt**
Svörin á síðan að setja í nýja skrá samkvæmt leiðbeiningum í Git útgáfustýringu.

GIT útgáfustýring (6%)

1. Git verkefni (3%)

- a. Stofnaðu reikning á GitHub.com (e. *Create Account*)
- b. Búðu til geymslu (e. *Create Repository*)
- c. Nefndu geymsluna „**git_verkefni**“
- d. Ekki gera neitt annað í bili á GitHub.

Notaðu Command Line Interface (CLI) - **Git Bash** í Windows til að leysa verkefnið
Opnaðu **Git Bash** og - **framkvæmdu eftirfarandi aðgerðir**

1. Athugaðu á hvaða drifi *Git Bash* vinnur - **pwd**, (/h/ = H: drif innan skólans).
2. Búðu til möppu með heitinu **git_verkefni** - **mkdir**
3. Vísðu *Git Bash* á möppuna - **cd**
4. Búðu til skrána README.md - **echo >> README.md**
5. Stofnaðu Git útgáfustýringu í möppunni - **git init**

```
gjjg@TV622-00 MINGW64 ~  
$ pwd  
/h/  
gjjg@TV622-00 MINGW64 ~  
$ mkdir git_verkefni  
gjjg@TV622-00 MINGW64 ~  
$ cd git_verkefni  
gjjg@TV622-00 MINGW64 /h/git_verkefni  
$ echo >> README.md  
gjjg@TV622-00 MINGW64 /h/git_verkefni  
$ git init  
Initialized empty Git repository in H:/git_verkefni/.git/  
gjjg@TV622-00 MINGW64 /h/git_verkefni (master)  
$
```

6. Skráðu vinnuferlið í Git (*staging area*) með - **git add .** (ekki gleyma .)
7. Skráðu síðan - **git status** til að sjá stöðuna.
8. Skráðu - **git commit -m "Add Readme"** í útgáfukerfið

```
gjjg@TV622-00 MINGW64 /h/git_verkefni (master)  
$ git status  
On branch master  
  
Initial commit  
  
Untracked files:  
  (use "git add <file>..." to include in what will be commit  
  ted)  
  
      README.md  
  
nothing added to commit but untracked files present (use "gi  
t add" to track)  
gjjg@TV622-00 MINGW64 /h/git_verkefni (master)  
$ git commit -m "Add Readme"
```

9. Opnaðu vefrita t.d. "**Sublime text editor**"
 - a. Opnaðu README.md skrána
 - b. Skrifaðu þar „#Fyrsta Git verkefnið“
10. Skráðu - **git add .** og síðan - **git commit -m "Readme titill"**
11. Til að hægt sé að senda gögn yfir á GitHub þá skráum við
 - **git config --global user.name 'pittNotendanafn'** og
 - **git config --global user.email 'pitt@email.com'** á GitHub reikningnum

```
MINGW64/c/Users/Notandi/Desktop/Git  
Notandi@Notandi-THINK MINGW64 ~  
$ git config --global user.name 'vsh24'  
Notandi@Notandi-THINK MINGW64 ~  
$ git config --global user.name  
vsh24  
Notandi@Notandi-THINK MINGW64 ~  
$ git config --global user.email 'vsh24@tskoli.org'  
Notandi@Notandi-THINK MINGW64 ~  
$ git config --global user.email  
vsh24@tskoli.org  
Notandi@Notandi-THINK MINGW64 ~  
$ cd /c/Users/Notandi/Desktop/Git  
Notandi@Notandi-THINK MINGW64 ~/Desktop/Git  
$ |
```

12. Næst setjum við Readme skrána á GitHub

`git remote add origin https://github.com/pinnReikningur/git_verkefni.git`

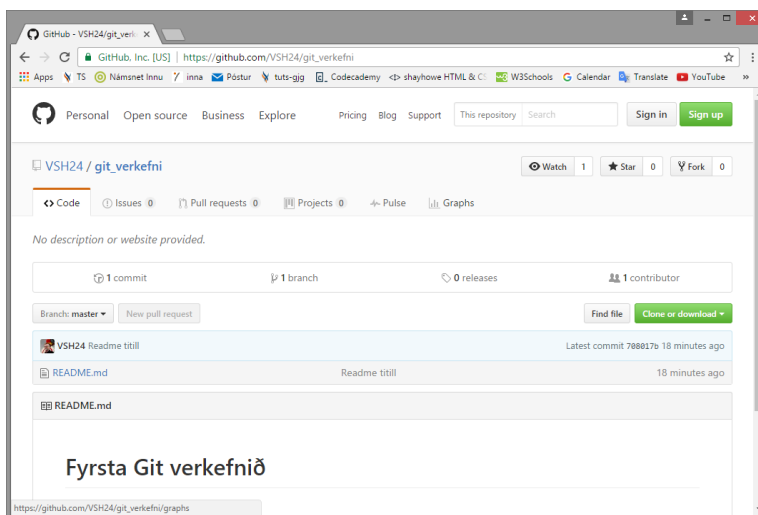
13. Skráðu - `git push -u origin master`

```
MINGW64/h/git_verkefni
gjjg@TV622-00 MINGW64 /h/git_verkefni (master)
$ git remote add origin https://github.com/VSH24/git_verkefni.git

gjjg@TV622-00 MINGW64 /h/git_verkefni (master)
$ git push -u origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 232 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/VSH24/git_verkefni.git
* [new branch]      master -> master
Branch master set up to track remote branch master from origin.

gjjg@TV622-00 MINGW64 /h/git_verkefni (master)
$
```

Ef allt gengur vel þá birtist valgluggi sem biður um notendanafn og lykilorð á GitHub reikning síðan hlaðast gögnin inn á GitHub geymsluna.



Á Github ætti geymslan að birtast svona.

14. Opnaðu vefrita t.d. “**Sublime text editor**”

a. búðu til skrá: `git_svor.txt`

b. skrifaðu efst í skrána “Git spurningar og svör”

c. vistaðu skrána í möppuna `git_verkefni`

15. Skráðu - `git add .` og síðan - `git commit -m "git svör"`

16. Afritaðu 1. spurningu úr `git_spurningar.txt` ásamt eigin svörum og settu í skrána `git_svor.txt`

17. Skráðu - `git add .` og síðan - `git commit -m "1. spurning"`

18. Gerðu eins fyrir hverja spurningu ásamt svari (16) þar til þú hefur klárað þær allar og bættu við jafnóðum skráningum (`commits`) í git (17).

19. Endurskoðun aðgerða (Checkout).

Notaðu skipunina - **git log** til að sjá allar skráningar (commits).

```
MINGW64:/c:/Users/Notandi/Desktop/Git/git_verkefni
commit 653efa44bad85ac2500dc8c3dac2e9d5112c029c
Author: vsh24 <vsh24@tskoli.org>
Date: Mon Nov 14 14:28:57 2016 +0000

    2. spurning

commit 594f61bec265b36ff79269c5a53b1eebf6066c94
Author: vsh24 <vsh24@tskoli.org>
Date: Mon Nov 14 14:19:08 2016 +0000

    1.spurning - OK

commit 4e2e4f2c491ff7bc370225f53f98aed07f113c1f
Author: vsh24 <vsh24@tskoli.org>
Date: Mon Nov 14 14:14:09 2016 +0000

    bætti -

commit 98cb37df2b80b076e6f634100216490c8b752cc1
:
```

Til að komast út úr log skrá smelltu á **q**

20. Til að fara í fyrstu útgáfu af verkefninu (Commit) 1.spurning

- afritaðu talnafærslna (velja og hægri smella - copy)
- git checkout** (talnarunan – paste)

```
MINGW64:/c:/Users/Notandi/Desktop/Git/git_verkefni
do so (now or later) by using -b with the checkout command again. Example:

    git checkout -b <new-branch-name>

HEAD is now at 594f61b... 1.spurning - OK

Notandi@Notandi-THINK MINGW64 ~/Desktop/Git/git_verkefni ((594f61b...))
$
```

pá ertu komin í 1. útgáfu sem var vistuð í byrjun verkefnisins

21. Opnaðu **git_svor.txt** í Sublime text til að sjá 1. útgáfu (e. version) af skjalinu

22. Notaðu - **git checkout** til að velja aðra útgáfu og síðast nýjustu útgáfuna (master)

23. Greinar (Branches)

Búðu til grein með skipuninni - **git branch hofundur**

24. Skoðaðu stöðu greina - **git branch**

25. Skráðu út greinina - **git checkout hofundur**

26. Bættu nafninu þínu neðst í **git_svor.txt** í Sublime text

27. Skráðu svo **git add .** og **git commit -m "höfundur"**

28. Færðu þig aftur á master grein - **git checkout master**, athugaðu hvort höfundarnafn sé í **git_svor.txt**

29. Skoðaðu síðan: **git log --graph --oneline master hofundur**

30. Samruni (Merge).

Á **master** grein skráðu eftirfarandi skipun: **git merge hofundur** Skoðaðu síðan **git_svor.txt** í Sublime text

31. Eyddu greininni „hofundur“ með skipuninni - **git branch -d hofundur**

32. Uppfærðu geymsluna (repository) á Github – **git push**

Ábending: Ef þú festist í CLI þá dugar að ýta á **q** til að hætta (án þess að loka CLI)

2. 3% **Github_verkefni**

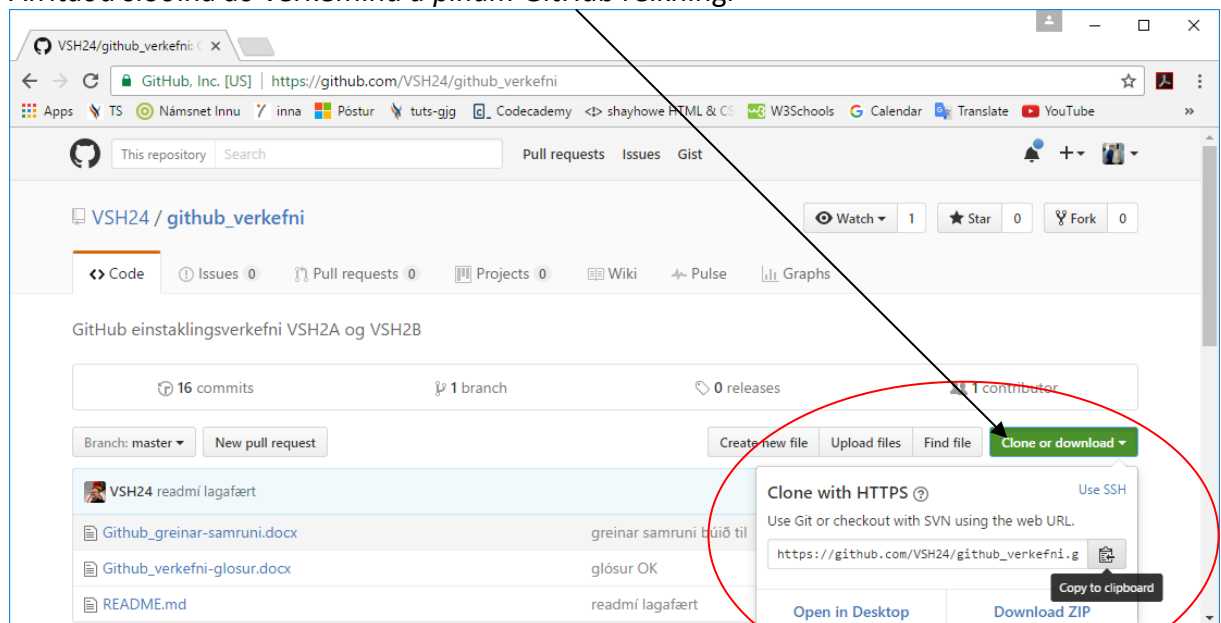
- afritað https://github.com/VSH24/github_verkefni á eigin *GitHub reikning* (*GitHub Account*) með goggunaraðferðinni „**Fork Repository**“.
- Geymsan (*repository*) er sett á local vinnusvæði með „**Git clone**“ í Git Bash.
- Verkefnið leyst samkvæmt verklýsingu hér neðar

Allar aðgerðir og skráningar eiga að vera í sögu skjalana á GitHub.
(*Branches / Commits / History*)

Greinar og samruni – *GitHub brances and merging*

Eftir að hafa afritað „**github_verkefni**“ á GitHub reikningin þinn (*account*) þá afritar þú það yfir á (local) vinnusvæði þitt samkvæmt eftirfarandi lýsingu.

Afritaðu slóðina að verkefninu á þínum GitHub reikningi



Git-Bash opnað

1. `cd` = vísað á vinnusvæðið
2. `git clone` = „github_verkefni“ afritað á H drifið
3. `cd` = vísið á „github_verkefni“
4. `(master)` sýnir að git útgáfustjórn er í gangi

gjjg@TV620-00 MINGW64 ~

```
1 $ cd /h/github_verkefni
```

gjjg@TV620-00 MINGW64 ~

```
2 $ git clone https://github.com/þú/github_verkefni.git
Cloning into 'github_verkefni'...
remote: Counting objects: 41, done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 41 (delta 13), reused 31 (delta 8), pack-
reused 0
unpacking objects: 100% (41/41), done.
checking connectivity... done.
```

```
gjjg@TV620-00 MINGW64 ~  
3 $ cd /h/github_verkefni  
gjjg@TV620-00 MINGW64 /h/github_verkefni (master)  
4 $
```

Í geymslunni „github_verkefni“ er mappan „**litaval**“ með vefsíðu og stílsíðu sem þú átt að vinna með í GitHub verkefninu.

Fylgdu eftirfarandi aðgerðum:

1. Gerðu nýja **grein** (Branch) í Git - **git branch vetur**
2. \$ **git checkout -b vetur** (Switched to a new branch 'vetur')
3. Bættu nýju litapema fyrir .vetur í stílsíðu og skiptu yfir í class="vetur" í HTML skjali
4. \$ **git add .**
5. \$ **git commit -m 'skipti yfir í vetur litapema'**
6. \$ **git status**
7. Skoðaðu vefsíðuna í vafra
8. Skiptu aftur yfir í master í Git og ræstu (refresh - F5) vafrann
9. Vistaðu aðgerðina og uppfærðu geymsluna á GitHub \$ **git push**

Breyting á Github

1. Gerðu breytingar á *README.md* skjali á Github

Samanburður og leiðrétting

2. Settu saman greinina „**vetur**“ við „**master**“ þannig að **vetur** litapema er á **master** útgáfunni.
3. Sendu breytingarnar yfir á Github (ef það er hægt)
4. Skoðaðu mismun á **Git** (local) og **Github** notaðu \$ **git pull** og \$ **git diff** - Lagfærðu ágreining* (Conflict)
5. Skilaðu Git yfir í Github (í lagi - Your branch is up-to-date with 'origin/master'.)

```
Notandi@Notandi-THINK MINGW64  
~/Desktop/Git/github_verkefni (master)  
$ git status  
On branch master  
Your branch is up-to-date with 'origin/master'.
```

- Í skjalinu „Árekstur-Conflict.docx“ er dæmi um ágreining

*Git Bash – Windows! Do not forget to add a comment when you commit your files. Strategic commenting in Git is as important as commenting code. If you accidentally forget to add a comment and end up in a strange screen where you can no longer enter any commands and **q** wont work*

- press ESC
- type **:q!**
- ENTER

Námsmat miðast við hvernig vinnuferlið er á github

https://github.com/pinnReikningur/github_verkefni/Commits

Bjargir:

Git: <https://git-scm.com/>

Git startup: <https://www.taniarascia.com/getting-started-with-git/>

Github-git-cheat-sheet: <https://services.github.com/kit/downloads/github-git-cheat-sheet.pdf>

Pro git bók: <https://git-scm.com/book/en/v2>

Nánar um Git Branching og Merging:

<https://git-scm.com/book/en/v2/Git-Branching-Basic-Branching-and-Merging>

How to Use Git and GitHub: <https://www.udacity.com/course/how-to-use-git-and-github--ud775>

Linux skipanir: <https://community.linuxmint.com/tutorial/view/244>

Námsmat:

GIT spurningar (2%)

Spurningum er svarað í **git_svor.txt** samkvæmt leiðbeiningum

GIT útgáfustýring (6%)

- 1 3% **Git_verkefni** Vinnusvæði vaktað með Github útgáfustjórnun og verkefnið leyst samkvæmt verklýsingu.
- 2 3% **Github_verkefni** afritað með goggunaraðferðinni „**Fork Repository**“ af https://github.com/VSH24/github_verkefni á eigin **GitHub reikning (GitHub Account)**.

Allar aðgerðir og skráningar eiga að vera í sögu skjalana á GitHub.
(*Branches / Commits / History*)

Verkefnaskil á Innu:

Skilaðu tengli á GitHub geymsluna þína í öllum áföngunum sem eru með hópverkefnið.