

GIT I.

Petr Šťastný

11.3.2021

České Budějovice

Partnery akce jsou:



Evropská unie
Evropský sociální fond
Operační program Zaměstnanost

CO NÁS DNES ČEKÁ

- Co jsou verzovací systémy
- Jak se pracuje se zdrojovými kódy
- Seznámení s GitHubem, vytvoření účtu
- Základní práce s nástrojem **Git**
- Domácí úkol, code review

VERZOVACÍ SYSTÉMY

PRÁCE SE ZDROJOVÝM KÓDEM

- Zdrojový kód (HTML, CSS, JS) + další soubory (obrázky)
- Potřebuji se v tom nějak orientovat
- Historie změn (verze) - kdo, co, kdy změnil
- Zálohování
- Různé verze projektu
- Organizace práce více lidí na jednom projektu

VERZOVACÍ SYSTÉM

- Na počítači mám složku s projektem (webovými stránkami)
- Ve složce zdrojový kód, edituji ho nějakým editorem (VS Code)
- Git bdí nad obsahem této složky, sleduje změny
- Repozitář - všechny soubory projektu, obvykle jeden centrální pro celý tým na nějakém serveru
- Z repozitáře si stahuji změny ostatních, odesílám své změny

GIT

- Jeden z mnoha verzovacích systémů
- Velmi oblíbený, bezplatný
- Pro malé i velké projekty (třeba i tisíce lidí)
- Ovládání přes příkazovou řádku
- Grafické nadstavby

GIT REPOZITÁŘ

- Potřebuji “centrální” repozitář pro sdílení kódu s kolegy
- Vlastní server, GitHub, GitLab, BitBucket
- Bezplatné (open source), placené (private)
- Úložiště kódu a veškeré historie
- Klikací webové rozhraní
- Týmová spolupráce - tikety, komentáře, code review

GITHUB

- www.github.com
- Nejoblíbenější služba pro Git repozitáře
- Budeme používat v této akademii
- Ukázka

BREAK ROOMS - REGISTRACE NA GITHUB

- www.github.com
- Zaregistrujte se
- Vytvořte si repozitář (třeba "test1")
 - Zvolte typ "Public" (veřejný)
 - Zaškrtněte "Add a README file"

ZÁKLADY GITU

PRÁCE S GITEM

- Budeme pracovat s příkazovým řádkem
- Na počátku si “naklonuji” projekt z hlavního repozitáře k sobě na počítač (git clone)
- Edituji u sebe kód, přidávám soubory (VS Code)
- Když je práce dokončena
 - Změněné soubory přidám do oblasti připravených změn (git add)
 - Provedu commit připravených změn (git commit) - vznik nové verze
 - Odešlu nové commity na server (git push)

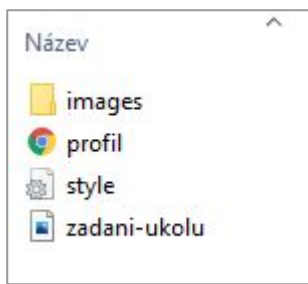
TERMINOLOGIE

- **Pracovní složka (working directory, working copy)** - kde mám kód u sebe na počítači, naklonovaná kopie z repozitáře
- **Staging area** - množina změněných souborů, které jsou připraveny pro commit
- **Commit** - potvrzení hotových změn, vytvoření nové verze (zachycení stavu kódu v čase), uložení historie
- **Push** - odesílám své změny do vzdáleného repozitáře
- **Pull** - stahuji změny ze vzdáleného repozitáře k sobě

VERZOVACÍ SYSTÉM

```
home > petr > Stažené > profil.html > ...  
1 <!DOCTYPE html>  
2 <html>  
3 <head>  
4 <meta charset="utf-8">  
5 <meta name="viewport" content="width=device-width, initial-scale=1">  
6 <title>Profil: Jana Czech  
7 <link href="https://fonts.googleapis.com/css?family=Roboto:400,700" rel="stylesheet">  
8 <link href="style.css" rel="stylesheet">  
9 </head>  
10 <body>  
11 <div class="profil">  
12 <div class="hlavicka">  
13 <div class="nazev">  
14 <div class="hadr">  
15
```

editor kódu



složka s projektem

sledování změn
add, commit



git

push

pull

vzdálený
repozitář



NASTAVENÍ GITU

- Je potřeba provést pár základních nastavení Gitu na vašem PC
- Platí pro všechny repozitáře

```
$ git config --global user.name "Jméno Příjmení"  
$ git config --global user.email "muj-email@example.com"  
$ git config --global core.editor "code --wait"  
$ git config --global pull.rebase true
```

NAKLONOVÁNÍ - GIT CLONE

- Prvotní stažení souborů z repozitáře ke mně
- Na mém PC v mé složce vznikne “working copy”
- Potřebuji znát URL adresu repozitáře k naklonování
 - GitHub: Code -> záložka HTTPS
- Otevřu si ve VS Code terminál, přepnu se do složky kde chci working copy (příkaz cd)

```
$ git clone https://github.com/pstast/test1.git
```

ALTERNATIVA KE KLONOVÁNÍ - GIT INIT

- Pokud chci vytvořit **nový repozitář**, mohu si jej nejprve vytvořit u sebe (zakládám úplně nový prázdný projekt)
- Musím ale ručně nastavit adresu vzdáleného repozitáře
 - Aby Git věděl, kam posílat změny (push/pull)
- Spíše používejte **git clone**

```
$ mkdir test
$ cd test
$ git init
$ git remote add origin https://github.com/pstast/test.git
```


STAV WORKING COPY - GIT STATUS

\$ **git status**

- Vypíše aktuální stav mé pracovní složky
- Neřeší změny v repozitáři
- Necomitnuté změny - **unstaged**, **staged**

PŘÍPRAVA ZMĚN KE COMMITU - GIT ADD

- Než budu commitovat, musím říci, jaké změny se mají do commitu zahrnout
- Nemusím zahrnout vše

```
$ git add README.md
```

```
$ git add *.html
```

- Anebo chci zahrnout všechny změněné soubory:

```
$ git add .
```

VÝPIS NECOMMITNUTÝCH ZMĚN - GIT DIFF

```
$ git diff
```

```
$ git diff README.md
```

- Podrobný výpis změn v souborech
- Diff = výpis rozdílů - přidané/odebrané/změněné řádky
- To samé vidíme také graficky ve VS Code
 - Kliknutím na barevné zvýraznění uvidíme detail změny
 - Je možné vrátit změnu zpět (také příkaz git restore)

GIT COMMIT

- Vytvoření nové verze
- S krátkým komentářem co se měnilo
- Commit se vytvoří lokálně na vašem PC, ještě se neodešle
- Commity nejsou číslovány, jsou označeny digitálním otiskem
 - "hash" - směs čísel a písmen, celkem 40 znaků, zkracuje se na 8 znaků

```
$ git commit -m "Doplnění README"
```

```
[main 8debf96] Doplnění README  
1 file changed, 3 insertions(+), 1 deletion(-)
```

ODESLÁNÍ ZMĚN NA SERVER - GIT PUSH

- Mnou lokálně vytvořené commity se odešlou do vzdáleného repozitáře (ze kterého jsme klonovali)
- Je nutné se ve VS Code přihlásit do GitHubu, protože se chystám provádět změny

\$ **git push**

STAŽENÍ ZMĚN ZE SERVERU - GIT PULL

- Stažení nových commitů ze serveru (vytvořené někým jiným)
- Změny provedené ostatních kolegy se aplikují do mé working copy

```
$ git pull
```

VÝPIS HISTORIE COMMITŮ - GIT LOG

- Výpis všech commitů
- Opět se pracuje jen s tím co znám lokálně, změny se nečtou ze vzdáleného repository
- Buď všechny commity nebo commity týkající se jen konkrétního souboru

```
$ git log
```

```
$ git log README.md
```

BREAK ROOMS - PRVNÍ COMMIT

- Naklonujte si svůj repozitář
- Na svém počítači proveďte změnu/přidání souboru
- `git add`
- `git commit`
- `git push` (vyžaduje přihlášení ke GitHubu ve VS Code)
- Prohlédněte si výsledek na GitHubu

GITHUB PAGES

Jak snadno publikovat web přes GitHub

GITHUB PAGES

- GitHub umožňuje obsah vašeho repository přímo publikovat jako webovou stránku na adrese **vaše-jméno.github.io**
- Tedy takový jednoduchý webhosting
- Pouze statické stránky (tj. žádný backend)
- Změna obsahu stránek = push do příslušného repository

PŘÍPRAVA WEBU PRO ZVEŘEJNĚNÍ

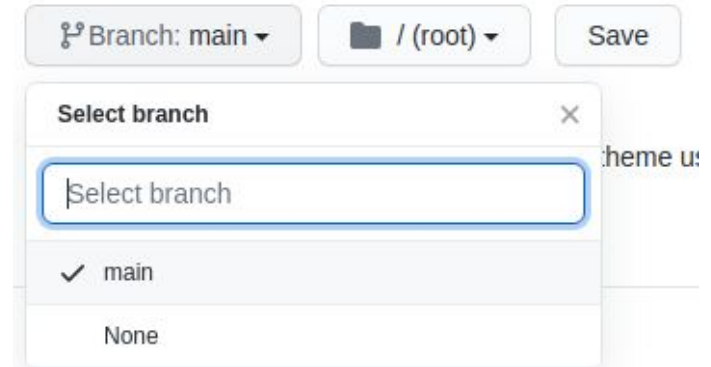
- Vytvoříme nový soubor **index.html** + add + commit + push

```
$ echo "Ahoj!" > index.html  
$ git add index.html  
$ git commit -m "Úvodní stránka webu"  
$ git push
```

ZVEŘEJNĚNÍ WEBU

- Na GitHubu v našem repository záložka **Settings**
- Sjedeme dolů do sekce **GitHub Pages**
- V rozbalovacím seznamu zvolíme “**main**”, uložíme **Save**

A je to: <https://pstast.github.io/test1/index.html>



BREAK ROOMS - GITHUB PAGES

- Vyzkoušejte si postup pro zřízení GitHub Pages

CODE REVIEW A ÚKOL 1

CODE REVIEW

- Kontrola nového kódu jinou osobou
- Základ týmové spolupráce, vzájemná revize práce
- Čtení cizího kódu
- Víc hlav/očí víc ví/vidí
- Na GitHubu lze přímo komentovat jednotlivé řádky kódu

ÚKOLY A VZÁJEMNÁ KONTROLA

- Během akademie celkem 6 domácích úkolů
- Úkoly si budete kontrolovat vzájemně - nácvik revize kódu
- V tabulce ke svému jménu napište vaše přihlašovací jméno na GitHubu + vložte odkaz na váš repozitář **piskvorky** (viz. dále)
- My vám náhodně přidělíme kdo bude kontrolovat vás a koho budete kontrolovat vy (pro každý úkol to bude jinak)

Podrobné instrukce:

<https://github.com/Czechitas-podklady-WEB/Uvod-do-code-review>

ZADÁNÍ ÚKOLU Č. 1

- V úkolech 1 až 5 budete programovat **piškvorky**
- **Úkol 1** - vytvořit základní statickou stránku s obrázkem a styly
- 7 dnů na vypracování + 2 dny na review
- Deadline: **18.3.**

Podrobné zadání:

<https://github.com/Czechitas-podklady-WEB/Ukol-Piskvorky-1>

Děkuji za pozornost

Dotazy a volná zábava

Petr Šťastný
petr@stastny.eu