



Co je Git a GitHub? *-*

Ing. Lukáš Mázl



Co je Git? a GitHub?

- **Git** je nástroj pro správu verzí vyvíjené aplikace
 - První vydání **7. dubna 2005**
 - Vytvořen **Linusem Torvaldsem** (říká Vám to jméno něco?)
-
- **GitHub** je webová služba podporující vývoj softwaru pomocí **Git** nástroje
 - Jednoduchá vizualizace kódu, vytváření větví nebo pull requestů
 - Aktuální verze kódu aplikace je uložena v **Repozitáři**

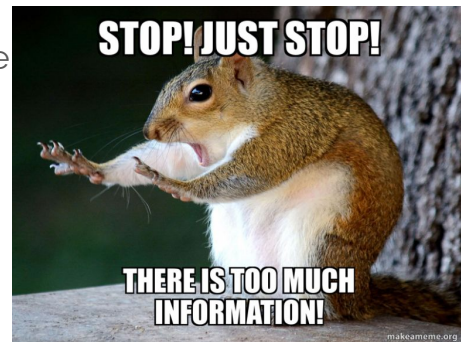
Co je Git? a GitHub?

- Kombinace **Git** a **GitHub** umožňuje týmům vyvíjet jednoduše společnou aplikaci
- Výsledný kód je zálohovaný a sdílený
- Je možné pracovat na více verzích najednou
- V kódu jsou viditelné jednotlivé změny
- Umožňuje hlášení chyb, kontrolu kvality kódu



Základní pojmy v Git/GitHub

- **repository** - Repozitáře s kódem vyvíjené aplikace. Obsahuje větve, tags, commits...
- **branch** - větev ve které probíhá vývoj. Základní větví je **master/main**, ve které je aktuálně nasazená verze
- **commit** - potvrzení vytvořených změn, commit má identifikátor v podobě hashID (**asd8dd18f**).
- **tag** - štítek pro hashID commitu, které umožní označit novou verzi aplikace
- **clone** - stažení aktuální verze na do lokálního prostředí
- **merge** - „spojení“ jedné větve s novou
- **pull request** - umožňuje nabídnout svoje změny do cizího repozitáře
- **fork** - clone celého repozitáře



Proč je Git/GitHub potřeba?

- Představte si:
 - máme aplikaci, jejíž vývoj nám zabral 6 měsíců
 - za pár dní je potřeba odevzdat
- několik dní před odevzdáním dojde ke ztrátě kódu (smaže se, odejde SSD či úmysl?)
- Vznikl „Doufám, že“ problém. Co něco takového znamená?
 - Doufám, že půjde SSD opravit
 - Doufám, že jsem si udělal kopii
 - Doufám, že si toho nikdo nevšimne
 - Doufám, že jsem nějakou subverzi poslal kolegovi a budu to mít v odeslaných e-mailech
- pokud byl během vývoje použitý **Git** stačí pouze stáhnout kopii do jiného zařízení





Proč je Git/GitHub potřeba? (Protože jeden důvod nestačí!)

- Představte si:
 - Na projektu je používán **Git**, ale vývoj je pouze v **main** větvi
 - Včera večer byla nasazena nová verze aplikace
 - Vývojový tým už začal připravovat další verzi (v **main** je ještě neotestovaná verze)
- Během dne přijde několik stížností, že aplikace nefunguje správně a bohužel chyba je v nové verzi
 - Musí dojít k rychlému pře-nasazení verze (tzv. rollback)
- Pokud je na projektu „Tag-ování“ (štítkování), je možné vzít minulou verzi a tu nasadit



Proč je Git/GitHub potřeba? (Protože dva důvody nestačí!)

- Představte si:
 - Na projektu je používán **Git**
 - Vytvoříte si novou větev **develop**, kde je potřeba vytvořit první změny, které jsou na delší dobu
- Klient si uvědomí, že potřebuje nové změny dostat do produkčního prostředí rychle
 - Je tedy potřeba vytvořit „**develop2**“ větev odvozenou z **masteru** větve
- Když verze s **develop2** je v produkci, musí dojít k „za mergování“ změn do **develop**

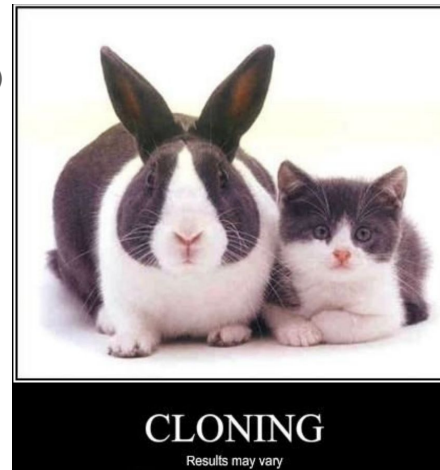
Jak začít?

- nejlépe tady <https://github.com> a tady <https://docs.github.com/en>
 - seznámím se s prostředím na <https://github.com> (viz. další slide)
- používám Linux -> terminál 😊
- používám Windows -> command line 😐 😐
 - využiji nějakého pomocníka (git bash, git cmd, git kraken, fork,...)
 - nainstaluji si pomocníka dle svého výběru a preferencí (ve škole máme git bash)



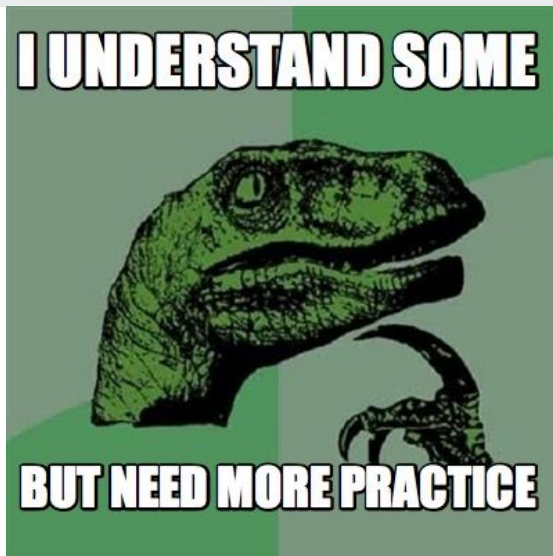
Seznámení s prostředím webové služby

- přejděte na <https://github.com>
- najděte si na nějaký repozitář (trending repo/dev, podle topicu, ...)
- seznámte se strukturou a významem jednotlivých prvků
- zkuste si clone vybraného repozitáře
 - https
 - ssh
- co se teď stalo?
- co mohu dělat dále?
- toto bylo pouze pro úvodní seznámení



Init, Add, Commit, Push

- příkazy znovu a prakticky 😊
- **git init** - inicializují repozitář v daném adresáři
- **git add** (název souboru, (*/-A : přidá všechny změny))
- **git commit -m** "Popis toho, co jsem změnil"
- **git push origin název větve (main/master...)** - nahrání změn na server
- **git pull** - stáhne verzi na webu na váš počítač
- čas na praktické procvičení a dotazy



Github, úkoly a markdown

- úkol se bude odevzdávat pomocí **pull request**
 - co je zač **pull request**
- online editor na markdown (např.: <https://dillinger.io/>, <https://pandao.github.io/editor.md/en.html>)
- markdown syntax (např.: <https://docs.github.com/en/get-started/writing-on-github/getting-started-with-writing-and-formatting-on-github/basic-writing-and-formatting-syntax>)
- nejlépe se učí praxí, takže pojďme na to

