



**Modernas izstrādes vides
uzstādīšana**

GIT

Pasākumu finansē: Eiropas Jūrlietu un zivsaimniecības fonds, projekts "Viedo ciemu attīstība piekrastes teritorijās". Nr. 19-00-F043.0443-000001.
VRG "Partnerība laukiem un jūrai", sadarbībā ar vadošo partneri VRG "Liepājas rajona partnerība"



**NACIONĀLAIS
ATTĪSTĪBAS
PLĀNS 2020**



EIROPAS SAVIENĪBA
Eiropas Jūrlietu un
zivsaimniecības fonds



Atbalsta Zemkopības ministrija un Lauku atbalsta dienests

Mēs zinām

- kas ir WEB lapa
- kas ir pašā taisīta mājas lapa
- funkcionālas/nefunkcionālās prasības
- personas kartiņas
- Mājas lapas imidž un lietojamība

Esam uztaisījuši paši savas mājaslapas .html .css un .json failos

Šodien uzzināsim

- kā programmētāji sadarbojas
- kas ir versiju kontroles sistēmas
- izveidosim mājaslapas glabātuvī, lai viegli tai piekļūtu jebkurā vietā un laikā
- padarīsim mūsu smilšukasti pieejamu citiem

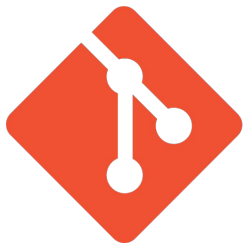
Tupināsim darbu pie kopienai vērtīgas mājas lapas padarot to pieejamu citiem

Problēmas... Kāpēc vajadzētu VCS?

- Vairāki cilvēki vienā projektā
- Koplietošana
- Pieeja no jebkura/cita datora
- Dokumentācija(kas? Ko? Kā?)
- Dublējumu saglabāšana
- Dalīšanās ar failiem, to glabāšana



Pieejamie rīki



git



git
bash

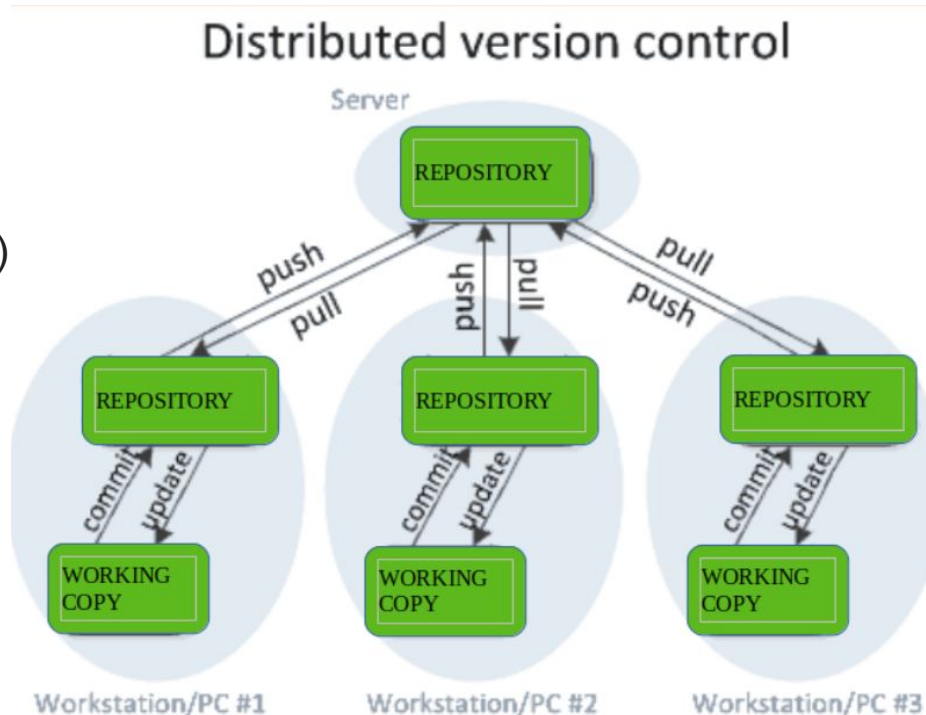


Bitbucket

Versiju kontroles sistēma

Lai citi iegūtu izmaiņas

- Tu saglabā izmaiņas lokāli(commit)
- Tu augšupielādē izmaiņas(push)
- Citi lejupielādē jaunākās izmaiņas(pull)



Zarošanās un versijas glabāšana

Commit – versijas glabāšana

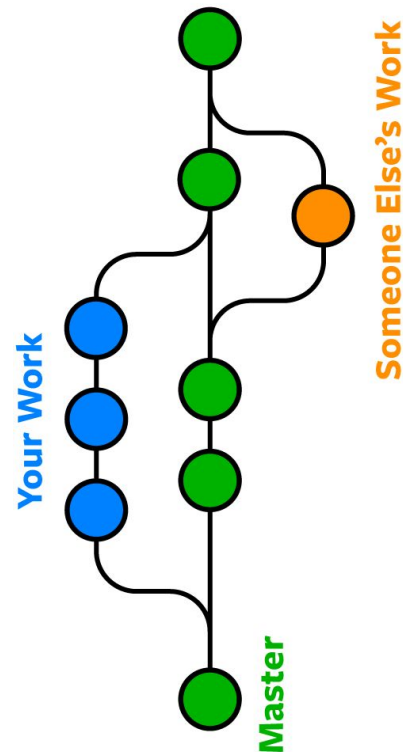
Katru reizi izdarot git commit, tiek saglabāta pašreizējā versija(kods, failu stāvoklis). Katru reizi tam tiek piešķirts hash kods, kas apslēpti satur datus par autora informāciju, laiku u.t.t... Pateicoties šai glabāšanai, var atgriezt kādu agrāko versiju izpildot komandu – git checkout <hash_kods>

Zarošanās

Zars jeb branch ir norāde ar nosaukumu uz kādu versiju.

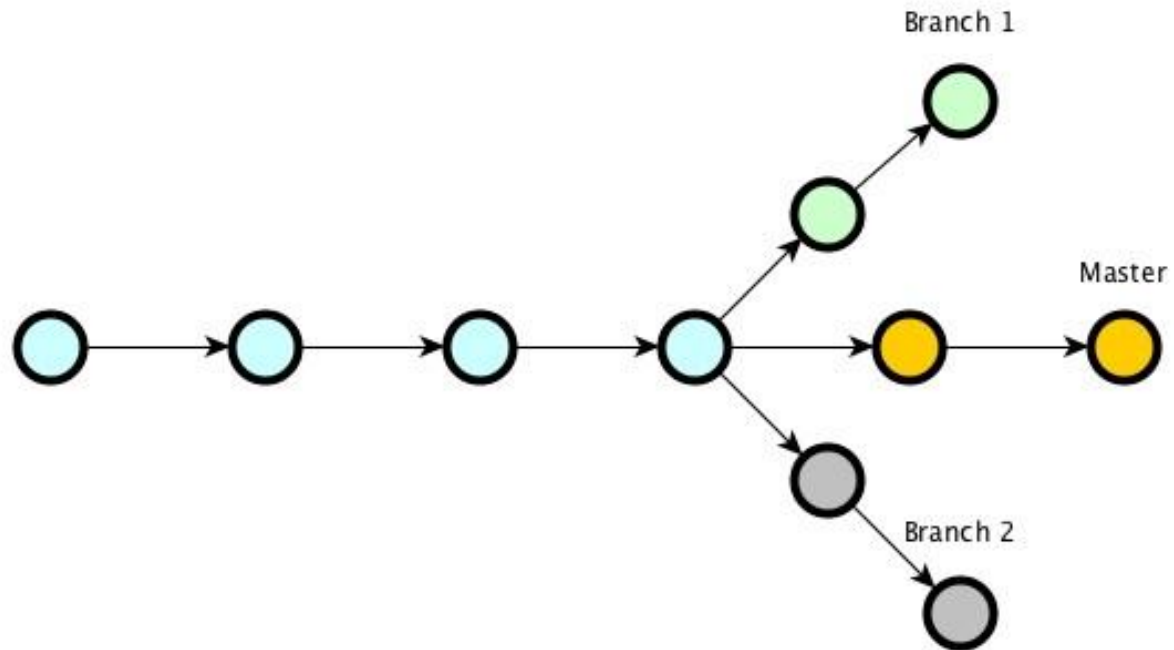
Repositorija sākumā vienmēr ir vismaz 1 zars, ko sauc par master, taču var izveidot vairākus zarus, kur eksperimentēt un tad savienot vairākus zarus vienā, galvenajā – master zarā.

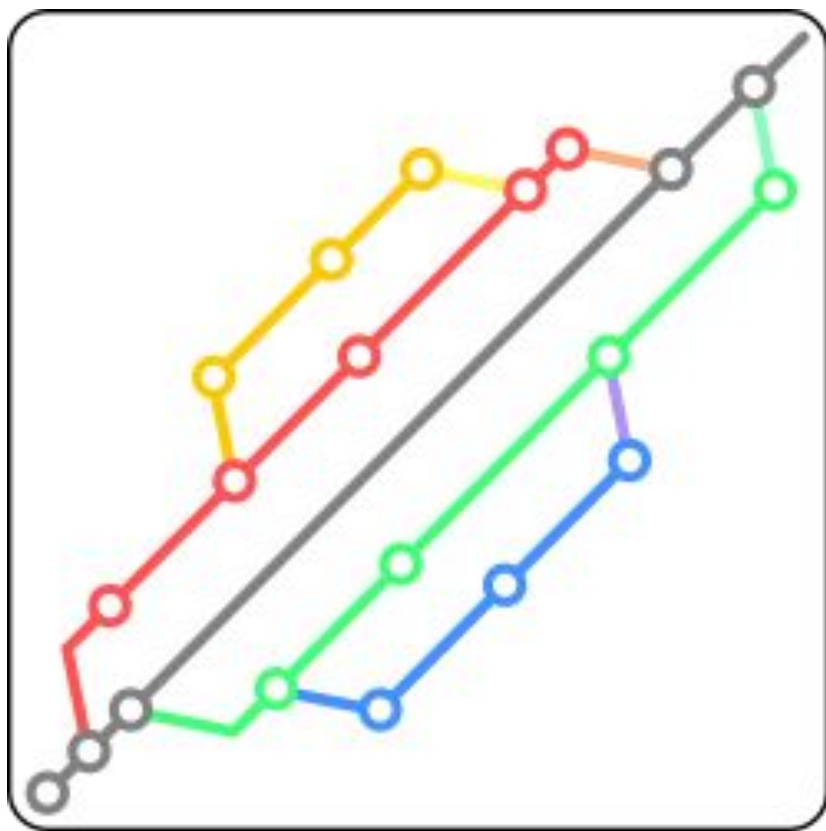
Piemērs: Var izveidot kādu papildus zaru, nokopēt kodu no master zara un ar to bez raizēm eksperimentēt. Ja neizdodas – master zarā joprojām ir iepriekšējā versija. Ja viss izdodas, tad varam master zarā ielikt izveidoto eksperimentālo versiju.

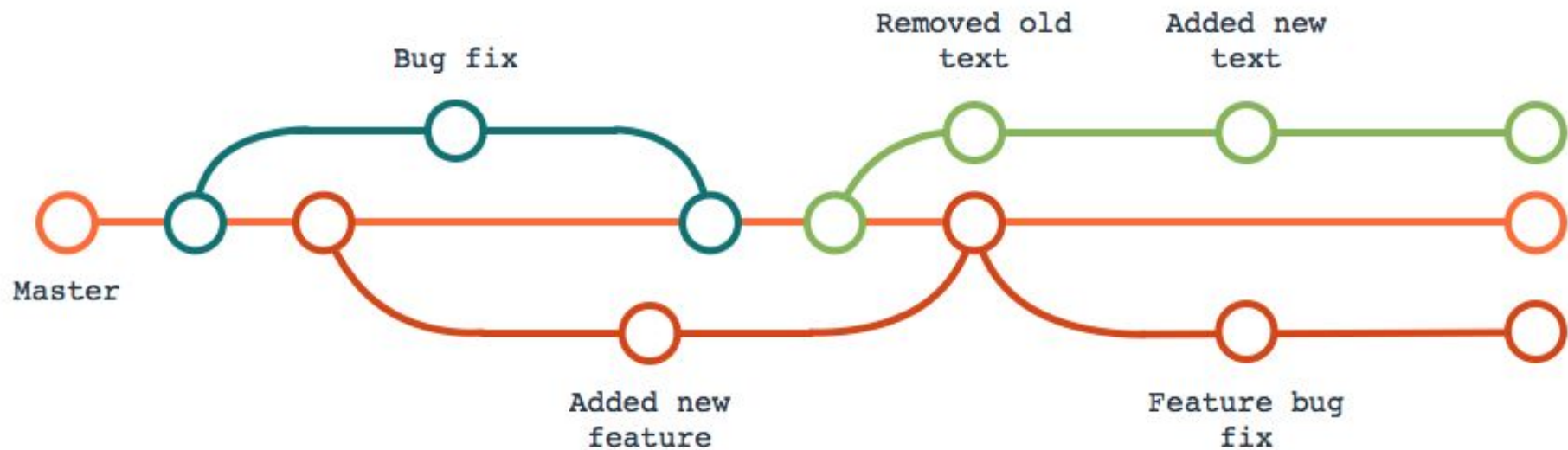


GIT koncepts

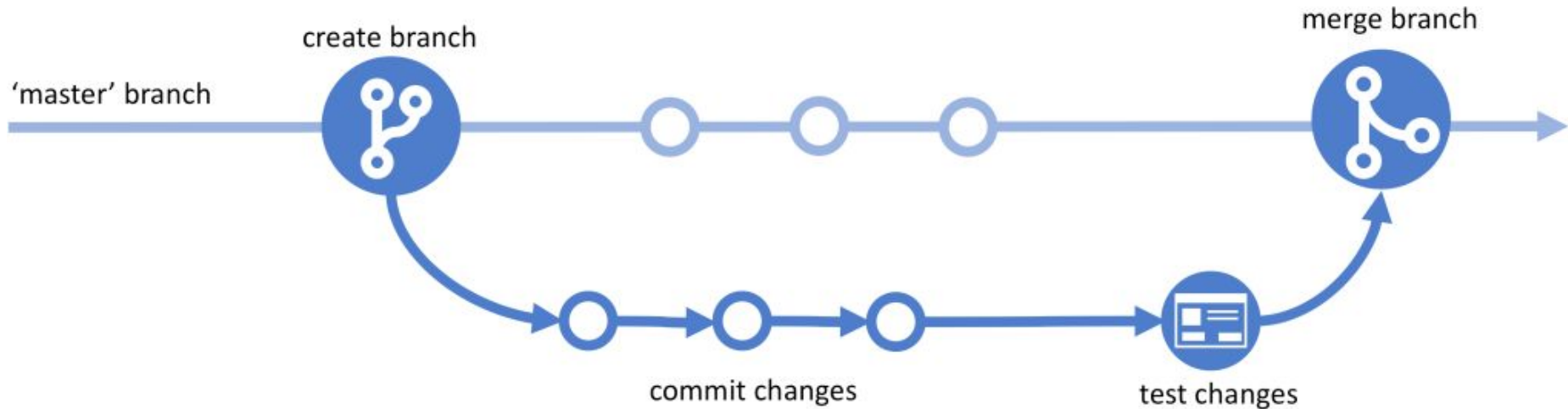
- Atcerās vēsturi
- Glabā katru failu
- Tu vari izvēlēties kad saglabāt failu veidojot commit
- Var apskatīties jebkuru iepriekšējo commitu jebkurā laikā
- Var izvēlēties, kurus failus augšupielādēt, kurus nē

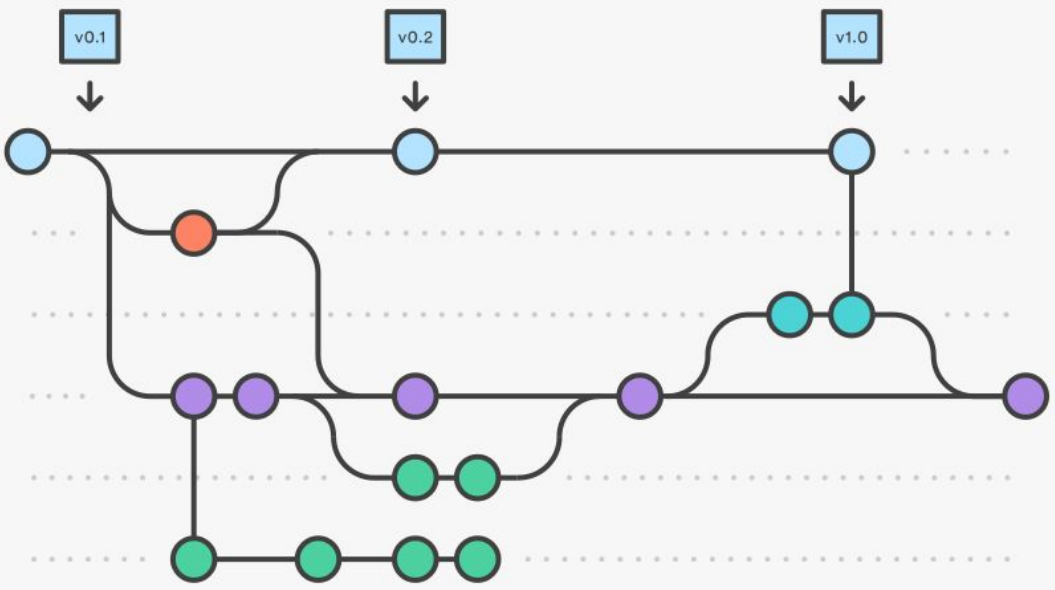






Simplified Git Flow





Komandas

- `$ git config --global user.name "John Doe"`
- `$ git config --global user.email johndoe@example.com`
-

Komandas

- git init – izveido glabātuvi
 - git checkout –b <nosaukums> – izveido jaunu zaru
 - git checkout <nosaukums> - pārslēdzas uz izvēlēto zaru
 - git status – parāda izdarītās izmaiņas glabātvē
 - git add --all – pievieno VISAS izmaiņas
 - git commit – saglabā izmaiņas lokāli
 - git push – augšupielādē izmaiņas
 - git log – parāda info par izdarītajām izmaiņām glabātvē
- Esam master branch → git merge <nos> - iepludina <nos> iekš master

HEAD



```
martins.lorencs@LVAGPLTP4118 MINGW64 ~/Desktop/-E TECH (master)
```

```
$
```

Noderīgas saites

<https://github.com/git-game/git-game>

<http://rogerdudler.github.io/git-guide/>

<https://learngitbranching.js.org/>

<https://www.freecodecamp.org/news/the-ultimate-guide-to-git-merge-and-git-rebase/>

<https://www.toptal.com/git/the-advanced-git-guide>

Praktiskais stūrītis 1

Izveidojam katrs savu zaru esošajā glabātuvē vai paši savu glabātuvi un izveidojam savu github lapu

//<https://github.com/L0renco/-ETECH>

git clone <https://github.com/L0renco/-ETECH.git>

git init

git checkout -b (my branch name)

git pull

(pievieno failus)

git status

git add --all

git commit -m "(komentārs)"

git push


Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner * / Repository name * ⚠

Great repository names The repository L0renco.github.io already exists on this account. antastic-octo-bassoon?

Description (optional)

 **Public**
Anyone on the internet can see this repository. You choose who can commit.

 **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

Add a README file
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

Piemēri... Varam pamēģināt

Tā kā github lapas ir ar open-source licenci, tad varam arī noklonēt citas github lapas un paskatīties/aizņemties to kodus un paņemt kā paraugu.

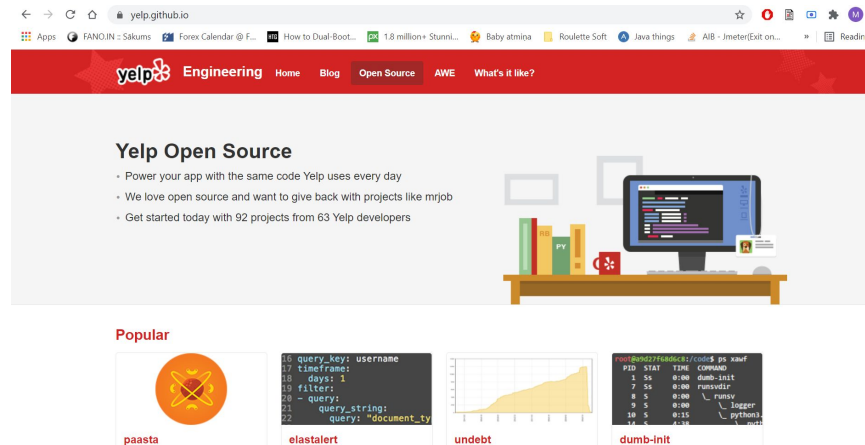
<https://github.com/collections/github-pages-examples>

git clone.....?

<https://twitter.github.io/>

<https://jekyllthemes.io/github-pages-themes>

<https://yelp.github.io/>



The screenshot shows the 'Yelp Open Source' page on GitHub. The page has a red header with the Yelp logo and navigation links: Engineering, Home, Blog, Open Source, AWE, and What's it like?. The main content area features the title 'Yelp Open Source' followed by a bulleted list: 'Power your app with the same code Yelp uses every day', 'We love open source and want to give back with projects like mrjob', and 'Get started today with 92 projects from 63 Yelp developers'. To the right is an illustration of a desk with a computer monitor, keyboard, and books. Below the main content is a 'Popular' section with four project cards: 'paasta' (with a red circular icon), 'elasticsearch' (with a terminal screenshot), 'undeb' (with a line graph), and 'dumb-init' (with a terminal screenshot).

Nākama lekcija

Plānojam nākamajā lekcijā mācīt integrēt Jūsu mājaslapā pieprasījumus no citiem avotiem, piemēram, dinamisku temperatūras rādītāju vai interaktīvu karti.

Mājas lapas arhitektūra



PALDIES

Apmācības nodrošina NVO "Piekrastes Konvents". Vairāk info - info@piekrasteskonvents.lv