Software Engineering: Tutorial 2

David Voigt

November 4th, 2022

Agenda

1. Discuss most the common errors in the last homework
2. Brief recap of Git Branching
3. Do some exercises on https://learngitbranching.js.org/
4. Group exercise

Homework discussion

Task 1

2. Welche Dateien befinden sich in der Working Area nach dem mit A markierten Befehl? Welche in der Staging Area?

   - Staging Area: `foo.txt`
   - Working Area: `bar.txt, foo.txt`

3. Wie wird ein Commit (konzeptuell) von git repräsentiert?

   A snapshot of the staging area is hashed together with the commit message, author and a timestamp. A new references is created, pointing from the new commit to its parent.

4. Wie sieht der Commit-Graph aus?

   'HEAD -> main -> "Last commit" -{parent}-> "Bad commit message"

Task 2

- Wie kann man `Student.scala` wieder auf den Zustand der Staging Area zurücksetzen?
  - `$ git restore Student.scala`
  - `$ git checkout Student.scala`

  Relevant StackOverflow Answer:
  https://stackoverflow.com/a/3044694
- Wie kann man `Student.scala` im Index wieder auf den Zustand von `HEAD` zurücksetzen?
  - `$ git restore --staged Student.scala`
  - `$ git restore --staged --worktree Student.scala`
  - `$ git reset Student.scala`
  - `$ git checkout HEAD Student.scala`

Task 2

- Wie kann man schauen, welche Änderungen zur Staging Area hinzugefügt wurden?

  ```
  $ git diff --staged
  ```

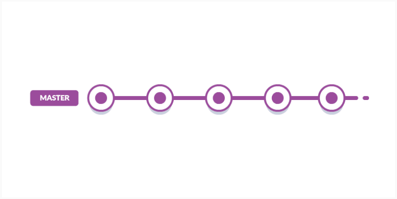
Git Branching

Branches

- Branches are for grouping development efforts into logically seperable units
- There are different possible workflows teams can use to organize their development process
    - Feature Branch Workflow
    - Gitflow Workflow

Workflows: Basic

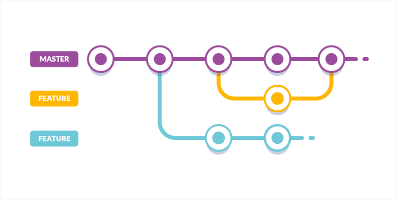
- Usually found on small repository with a handful of developers
- Cannot be used in production
- Plenty of merge errors bound to happen



Source: https://buddy.works/blog/5-types-of-git-workflows

Workflows: Feature Branch

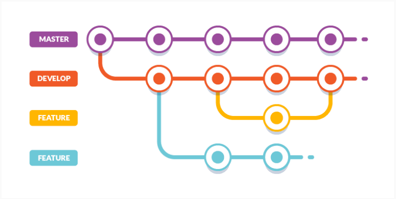
- The main branch is always production-ready
- Feature branches are used making changes to main
- If the feature is tested it may be merged into main



Source: https://buddy.works/blog/5-types-of-git-workflows

Workflows: Gitflow

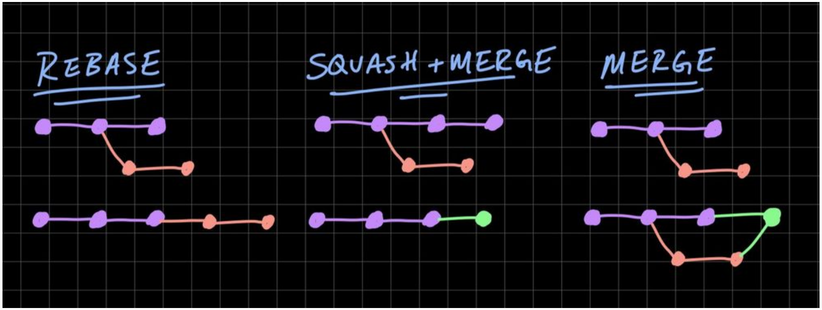
- The main branch is only used for new releases
- Features for new releases are gathered on the develop branch
- New feature branches, branch from the develop branch
- Besides the develop branch, only hotfixes directly branch from and back into main



Source: https://buddy.works/blog/5-types-of-git-workflows

Git Merging

merge vs. rebase vs. squash



https://matt-rickard.com/squash-merge-or-rebase

- merge creates a merge commit pointing to both parents
- rebase rewrites the history of the other branch on top of main
  - Information of branch-off is lost
- squash is something between a merge and a rebase
  - multiple commits are squashed into one. The squashed commit is added on top of main

Demo

- Exercises: https://learngitbranching.js.org
- Sandbox: https://learngitbranching.js.org/?NODEMO