Git使用

智能系统实验室

清华大学基础工业训练中心

一个作家的自述

- 我是一个作家, 每天的日常就是伏案写作, 一天得写好几千字才能糊口. 最宝贵的财产是一张桃花芯木书桌, 上面有一个盒子, 只有自己觉得写的比较好的稿子, 才会放到盒子里面, 不然就揉成一团丢给脚边的Tom玩.
- 每天晚上, 我都会整理盒子里面的稿件, 认真修订一遍, 把它们订成一个小册子, 放到书桌后面的书架上, 才能安心去睡觉. 这似乎都成了一种仪式, 要是小册子太薄或者太潦草, 我都很难睡的踏实.
- 编辑每周都会来催稿. 然后把书架上面的小册子装进公文包带走, 当然更多的是带来一些读者的意见. 这往往就意味着改.再怎么自以为绝妙的想法, 在销量面前, 都是无力的. 只有些运气比较好的小册子, 会有出单行本的可能. 我的报酬, 就能更多一点. 但这也意味着更多的工作, 重新修订不说, 好的插画师也很难找…
- 很多时候我也常幻想有一种药,能让灵感源源不断涌现. 没灵感的时候确实很痛苦, 这时候我会回愤怒地把书桌上所有的东西全推到地上, 摸摸Tom. 然后等怒气平息了, 再一点点把稿件捡回来.

抽取概念和行为

- 我是一个作家, 每天的日常就是伏案写作, 一天得写好几千字才能糊口. 最宝贵的财产是一张桃花芯木书桌, 上面有一个盒子, 只有自己觉得写的比较好的稿子, 才会放到盒子里面, 不然就揉成一团丢给脚边的Tom玩.
- 每天晚上, 我都会整理<mark>盒子</mark>里面的稿件, 认真修订一遍, 把它们订成一个<mark>小册子</mark>, 放到书桌后面的<mark>书架</mark>上, 才能安心去睡觉. 这似乎都成了一种仪式, 要是小册子太薄或者太潦草, 我都很难睡的踏实.
- 编辑每周都会来催稿.然后把书架上面的小册子装进公文包带走,当然更多的是带来一些读者的意见. 这往往就意味着改.再怎么自以为绝妙的想法,在销量面前,都是无力的.只有些运气比较好的小册子,会有出单行本的可能.我的报酬,就能更多一点.但这也意味着更多的工作,重新修订不说,好的插画师也很难找…
- 很多时候我也常幻想有一种药,能让灵感源源不断涌现. 没灵感的时候确实很痛苦, 这时候我会回愤怒地把书桌上所有的东西全推到地上, 摸摸Tom. 然后等怒气平息了, 再一点点把稿件捡回来.

抽取概念

自然语言

- 书桌
- 书架
- 盒子(放)
- 公文包(放)
- 单行本
- TOM
- 编辑部

Git语言

- Repo Dir (local)
- Add (放入盒子)
- Commit (放上书架)
- 丢弃
- Push Remote (编辑带走)
- Repo Remote (编辑部)
- Merge (读者意见)
- Release

梳理流程

create & clone

create new repository

clone local repository

clone remote repository

git init

git clone /path/to/repository

git clone username@host:/path/to/repository

add & remove

add changes to INDEX

add all changes to INDEX

remove/delete

git add <filename>

git add *

git rm <filename>

commit & synchronize

commit changes

push changes to remote repository

connect local repository to remote repository

update local repository with remote changes

git commit -m "Commit message"

git push origin master

git remote add origin <server>

git pull

多个作家共同创作

branches create new branch git checkout -b
branch> e.g. git checkout -b feature_x switch to master branch git checkout master delete branch git branch -d <branch> **push branch** to remote repository git push origin
 tranch> merge merge changes from another branch git merge <branch> **view changes** between two branches git diff <source_branch> <target_branch> e.g. git diff feature_x feature_y tagging create tag git tag <tag> <commit ID> e.g. git tag 1.0.0 1b2e1d63ff

get commit IDs

git log

实践Git

- Git-gui Client软件
- 下载链接: https://git-scm.com/download/
- 命令行工具Git-bash;
- Gitlab使用: 账户注册
- http://gitlab.icenter.tsinghua.edu.cn
- Github使用: 账户注册
- https://github.com

谢谢!

zhenchen@Tsinghua.edu.cn