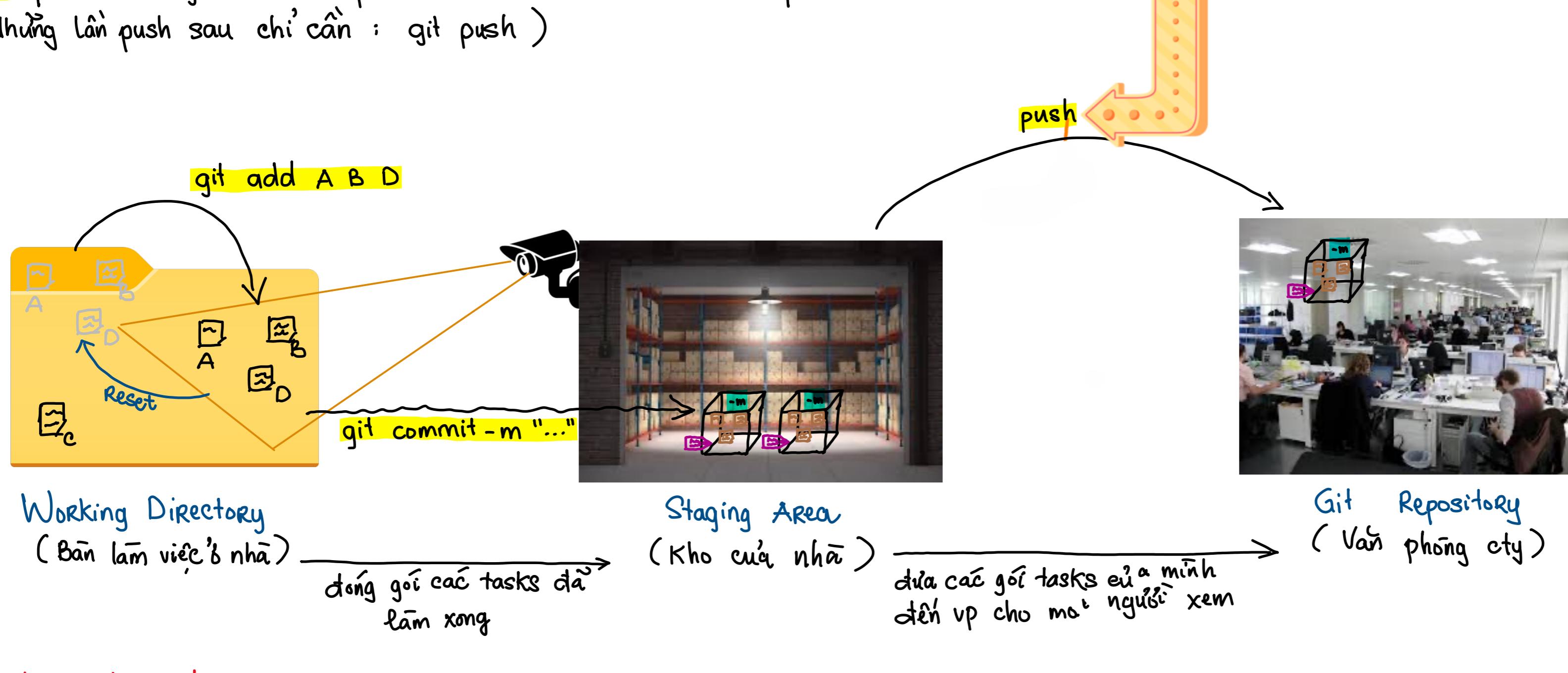


1. Tạo Repo Rỗng

- None hết

2. Upload code lên Repo

- git remote add origin <link-repo> trong folder project
- git push -u origin master push các commits ở local lên repo
(Nhưng lần push sau chỉ cần : git push)

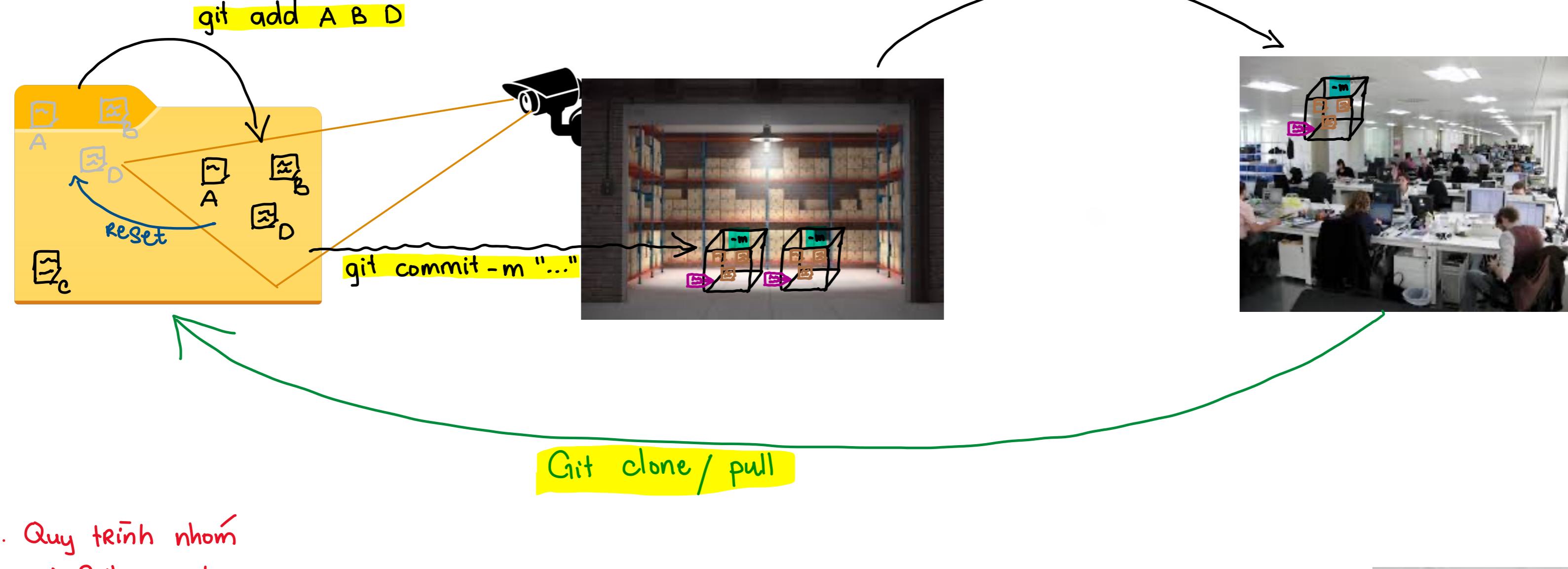


3. Git Credential

- Lưu lại user-name & password cho pushing
- Có nhiều cách:
 - git ssh (search google)

4. Clone & pull

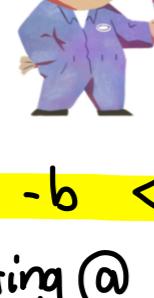
- Git clone : copy git Repo về local (Dùng 1 lần để down pj về)
- Git pull : dùng thường xuyên để update code



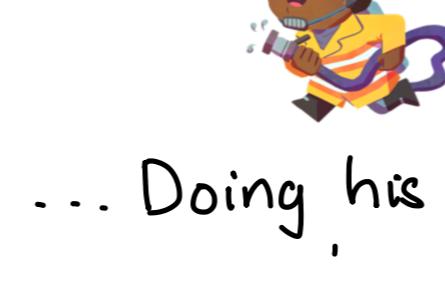
5. Quy trình nhóm

a) Pull Request

- Để làm 1 task :



Hi, my name is A

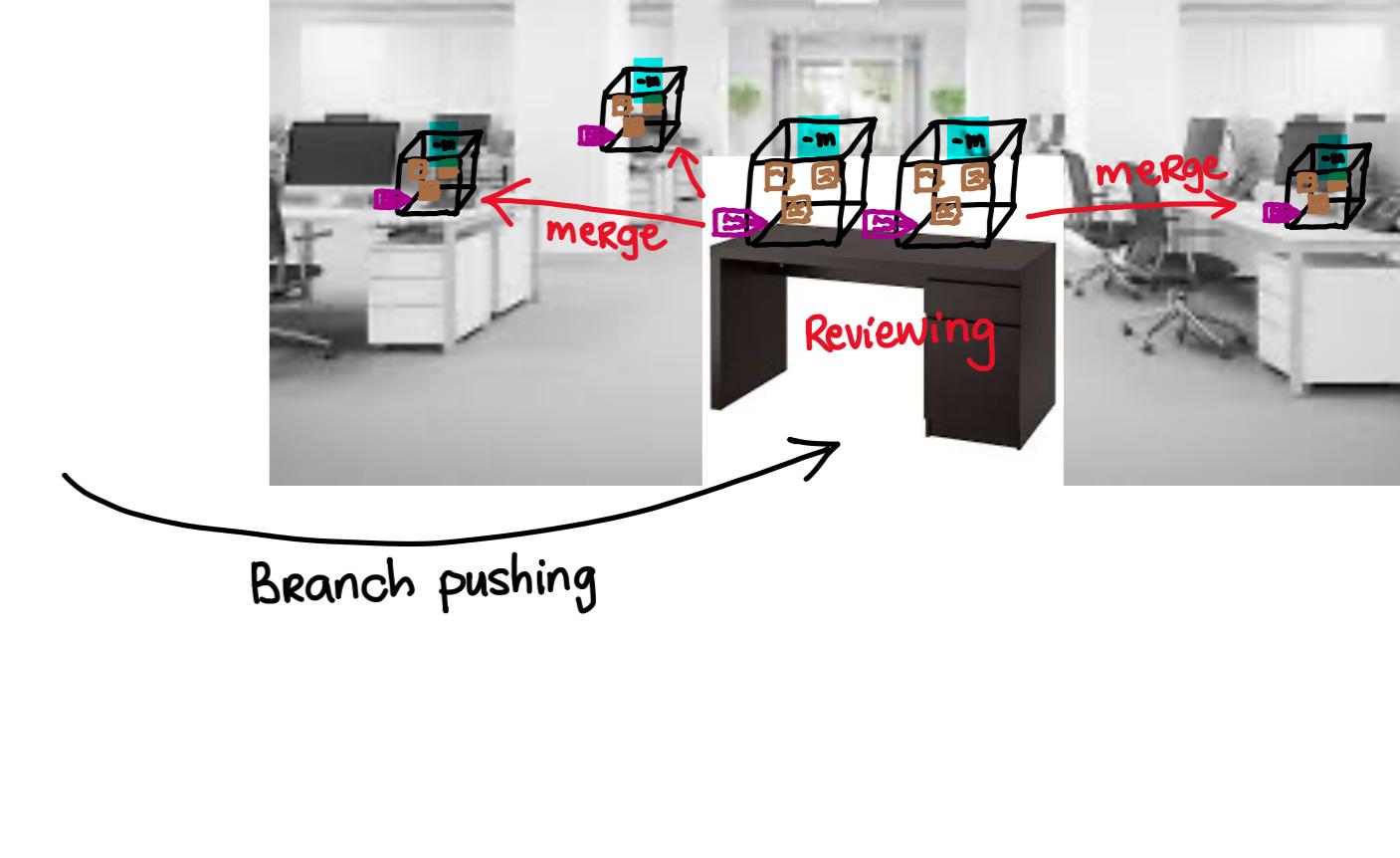


... Doing his tasks (B) ...

- + git checkout -b <feature-branch>
- committing @ local
- + git push origin <feature-branch>
- + create pull request to master

+ merging (xoá remote branch)

+ review code of A



b) Review Code

- b.1. Review code online on the Repo page

- b.2. Review at B's local

- Git fetch <reviewing-branch> : down branch về
- Git checkout <reviewing-branch>
- {npm install : cài missing packages mà máy A dùng | chạy code}
- {Review online (b.1) → Approve (A merge & delete branch ; B pull) / comment | checkout master, remove local <reviewing branch>}

c) Conflicts : xung đột giữa các commit

- Khi A, B modify same file, same line
- A del X.txt, B mod X.txt

Yêu cầu resolve khi merge pull request

c.1. Resolve by Rebase

- ở branch có conflict (branch chưa merge được)

git rebase master

branch cần merge vào

- Mở editor lên sửa conflict

- git add <conflicting files>

- git rebase --continue

- git push origin <branch> -f → pull request tự động update, (able to merge)

c.2. Resolve by merge (3-way merge)* (ưu tiên sử dụng)

- Update master { git checkout master

- git pull

- git checkout <conflicting-branch>

- git merge master

- Mở editor sửa conflict

- git add <conflicting files>

- git commit -m "fix conflict:..."

- git push origin <conflicting branch>

→ pull request tự động update (able to merge)