Tuy đã có 1 bài note về git nhưng thấy cái này thú vị quá nên cũng “chôm” về để dành =))

#Hướng dẫn dùng Github cơ bản theo phong cách “mì ăn liền”

https://daynhauhoc.com/t/huong-dan-dung-github-tren-windows-co-ban-step-by-step/6526

----



Đã có khá nhiều bài hướng dẫn sử dụng Github rồi nhưng mình vẫn muốn viết bài này để giúp những bạn muốn tìm hiều về Github có thể sử dụng một số chức năng cơ bản một cách “mì ăn liền” nhất có thể, tức là sẽ tập trung tối đa vào phần thực hành, lí thuyết thì nghịch xong rồi học thêm hehe 

Mà bạn chưa biết Github là cái gì á? Nói chung là Daynhauhoc search hoặc Google search nhá 

## **Chuẩn bị**

**Đầu tiên đăng kí tài khoản Github** [https://github.com/ 797](https://github.com/)

các bạn đăng kí 2 cái nhá, để mò mẫm các chức năng của Github cho dễ

**Sau đó là cài đặt Github for Windows(GFW)** [https://windows.github.com/ 1.4k](https://windows.github.com/)

Cài xong thì nhớ bạn đăng nhập vào GFW 1 tài khoản Github nhá, và mở trình duyệt khác rồi đăng nhập tài khoản 2 vào, ví dụ tài khoản A của mình đăng nhập trên firefox và trên GFW, còn tài khoản B mình đăng nhập trên Google Chrome.

## **Xong bước chuẩn bị, bây giờ là thực hành**

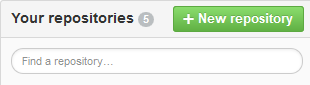
Github có 2 thể loại thế này:

1. Tạo một kho lưu code (sau này gọi là repo) cho một project của mình,.
2. Tham gia vào một project nào đó.

Bây giờ ta sẽ cho tài khoản A tạo 1 cái repo chứa 1 file test.txt chẳng hạn, tài khoản B sẽ tham gia vào project của tài khoản A.

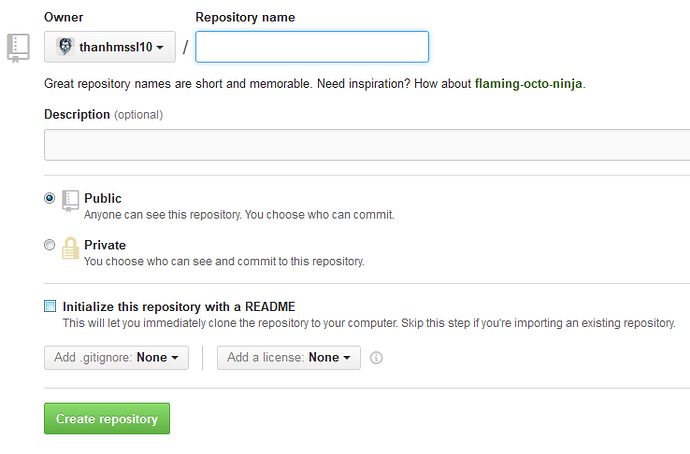
**Tạo repo cho A**

Trên giao diện web của A ấn nút New Repository



New Repository

Điền tên repo với description là được, mấy cái còn lại thì cứ kệ nó nhá (kệ vì nó không ảnh hưởng lắm đến bài viết này, chứ không phải không quan trọng đâu nhá :blush: )

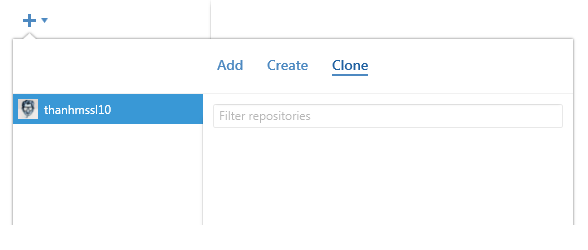


[**newrepo2.png**](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/7/76a4800ae5002b93adffbf520f94c13ef214f265.png)**[738x486 25 KB](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/7/76a4800ae5002b93adffbf520f94c13ef214f265.png)**

Creat Repository là xong

**Clone về máy**

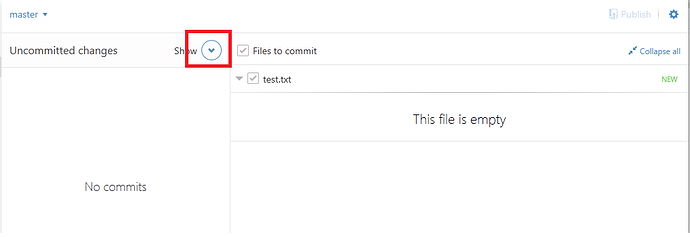
Mở Github for Windows (GFW) để clone repo về máy



Ấn vào dấu cộng góc trên cùng bên trái, chọn Clone, rồi nó sẽ hiện ra repo mình vừa tạo ở trên, ấn Clone Repository

**Thực hiện commit đầu tiên nào!!!**

Clone xong thì trên máy có một folder mang tên repo của bạn, bạn cho test.txt vào đó, khi đó quay lại GFW ta sẽ thấy “cảnh tượng” như này 



[**commitfirst.png**](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/d/d93bd9352edb18ce5bc09569d8021990ebe50c20.png)**[1018x350 14.3 KB](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/d/d93bd9352edb18ce5bc09569d8021990ebe50c20.png)**

Ấn Show rồi commit

Vậy là thay đổi của bạn giờ được lưu vào lịch sử thay đổi trên máy của bạn, sau đó ta sẽ cho lên repo trên github

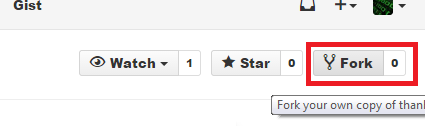


Ấn Publish để đưa lên repo trên github,chữ publish sau lần đầu tiên sẽ đổi thành sync

**Fork về tài khoản B**

Bây giờ đến lượt ông B, ông B sẽ “chôm” repo của ông A về tài khoản của mình (hành động này gọi là Fork)

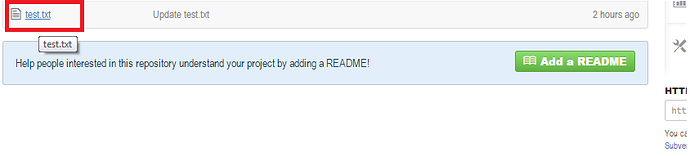
Từ giao diện web của B tìm đến repo của A rồi ấn nút Fork



Bây giờ B đã có 1 repo chứa dữ liệu giống hệt repo của A

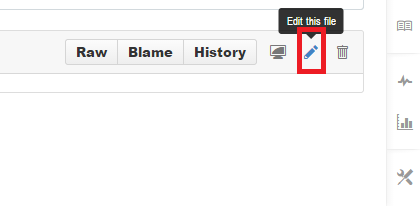
**Chỉnh sửa test.txt và Pull request (sau này gọi là PR) từ B tới A**

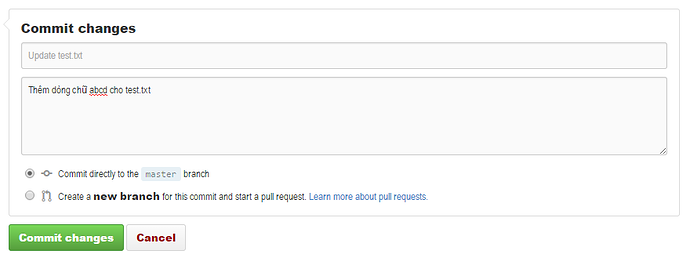
Bây giờ B sẽ chỉnh sửa file test.txt, (chỉnh sửa trực tiếp trên giao diện web của B), khi đó file test.txt giữa A và B đã có sự khác nhau và ta có thể gửi PR từ B tới A , PR nghĩa là B nói với A:“B đã sửa file test.txt, yêu cầu A hợp nhất vào repo của A”, nếu như A đồng ý với yêu cầu đó,thì file test.txt trên repo của A sẽ thay đổi theo file trên repo của B



[**edit1.png**](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/4/4a57bd85bba934e420cc3fe8121bf7816daa75a5.png)**[844x192 12.3 KB](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/4/4a57bd85bba934e420cc3fe8121bf7816daa75a5.png)**

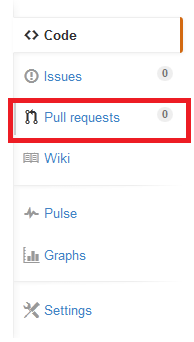
Bắt đầu chỉnh sửa file test.txt trên repo B

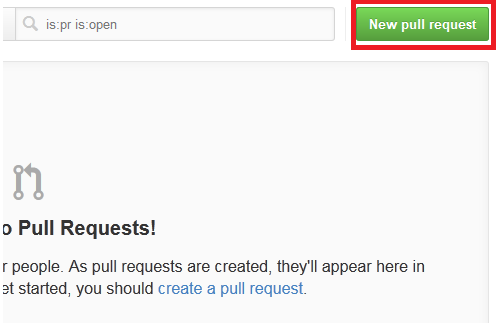


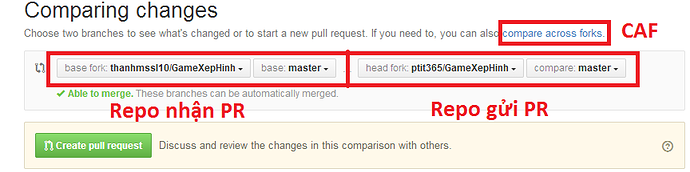


[**edittest.png**](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/c/cbca3186197a922416e00fc3d8bee635bad8a9e2.png)**[880x338 17.3 KB](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/c/cbca3186197a922416e00fc3d8bee635bad8a9e2.png)**

Và đây là quá trình gửi PR





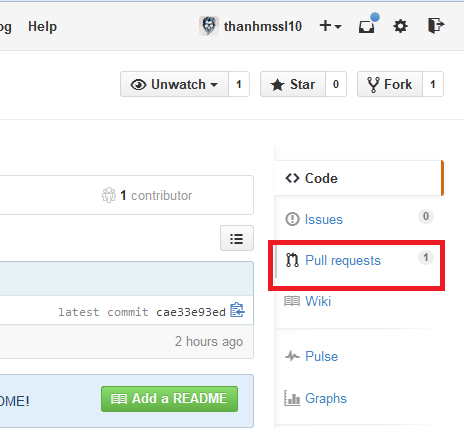


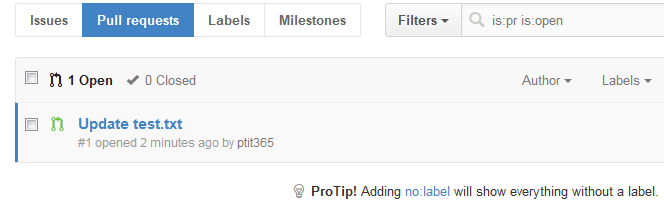
[**pr3.png**](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/d/dd462fd7f560d55e86dbe8c0e0ffb6942768fcf3.png)**[864x216 21.3 KB](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/d/dd462fd7f560d55e86dbe8c0e0ffb6942768fcf3.png)**

Thông thường thì github tự động hiện bên nhận PR là repo của A và bên gửi PR là repo của B, nhưng nếu không hiện đúng như vậy thì bạn phải tự chỉnh bằng tay, trong lúc đổi bên nhận gửi PR, thì sẽ có thời điểm là bên nhận và bên gửi là cùng 1 repo, khi đó phải dùng nút CAF để giải quyết

**Sang bên A xem có gì xảy ra nào**

Sau khi B gửi PR đến A, thì bên A sẽ hiện thông báo có PR, bây giờ ta sẽ chấp nhận PR đó, nội dung của file test.txt trên repo A sẽ bị thay đổi theo file test.txt của bên B





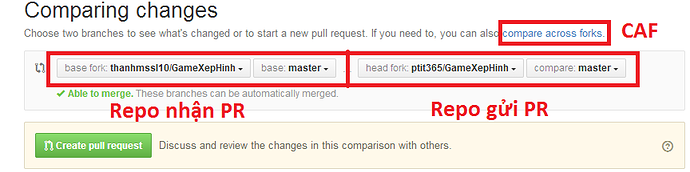


[**merge3.png**](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/4/4ba20155168df8e7040acf0fa1a084365aae81b5.png)**[774x186 17.4 KB](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/4/4ba20155168df8e7040acf0fa1a084365aae81b5.png)**

Và bây giờ những thay đổi của B đối với file test.txt đã được chấp thuận và được hợp nhất vào file test.txt của A

**Update cho B**

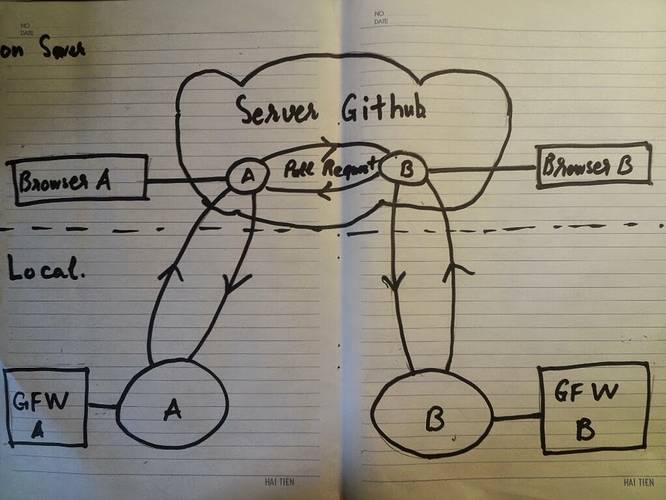
Một ngày đẹp trời A sửa file test.txt trên repo của A, nếu B muốn file test.txt của mình cũng được thay đổi giống như A thì sao nhỉ? Xoá repo đi fork lại? gọi điện cho A bảo: “Alo ông gửi pull request cho tôi đi để tôi hợp nhất vào repo của tôi”? Không cần rắc rối như vậy, github cho phép B “tự sướng” tạo một PR từ A đến B rồi tự đồng ý để merge (hợp nhất) vào repo của mình.



[**pr3.png**](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/d/dd462fd7f560d55e86dbe8c0e0ffb6942768fcf3.png)**[864x216 21.3 KB](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/d/dd462fd7f560d55e86dbe8c0e0ffb6942768fcf3.png)**

Nghĩa trên giao diện web của B, chỉ cần tạo PR : bên nhận là repo B còn bên gửi là repo A

Đây là sơ đồ khái quát quá trình trên, cái GFW của B ta không dùng đến vì ta sửa file trực tiếp trên Browser B ,trong thực tế rất ít khi làm như vậy, hầu hết là chỉnh sửa trên local trước rồi mới cho lên (vẽ tay cho nhanh, khi nào phải vẽ lại trên máy tính cho nó chuyên nghiệp mới được  )



[**upload\_-1.jpg**](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/c/c64a6a85b35aae3b196eaff79eecf0982716a2a7.jpg)**[1024x768 125 KB](https://daynhauhoc.s3-ap-southeast-1.amazonaws.com/original/2X/c/c64a6a85b35aae3b196eaff79eecf0982716a2a7.jpg)**

Rồi, bây giờ thì bạn có thể ngồi nghịch với 2 tài khoản A, B đó cho quen, rồi có thể mò thêm các cái khác như là tạo readme.md, tạo thêm branch, “cỗ máy thời gian”,… 

Có vấn đề gì thì cứ hỏi tại đây nhé 