Github là gì?

[GitHub](https://github.com/) là một dịch vụ nổi tiếng cung cấp kho lưu trữ mã nguồn [Git](https://topdev.vn/blog/git-la-gi/" \t "_blank) cho các dự án phần mềm. Github có đầy đủ những tính năng của Git, ngoài ra nó còn bổ sung những tính năng về social để các developer tương tác với nhau.

Vài thông tin về GIT:

* Là công cụ giúp quản lý source code tổ chức theo dạng dữ liệu phân tán.
* Giúp đồng bộ source code của team lên 1 server.
* Hỗ trợ các thao tác kiểm tra source code trong quá trình làm việc (diff, check modifications, show history, merge source, …)

Chức năng của Github

GitHub được coi là một mạng xã hội dành cho lập trình viên lớn nhất và dễ dùng nhất với các chức năng cốt lõi như:

Wiki, issue, thống kê, đổi tên project, project được đặt vào namespace là user.

Watch project: theo dõi hoạt động của project của người khác. Xem quá trình người ta phát triển phầm mềm thế nào, project phát triển ra sao.

Follow user: theo dõi hoạt động của người khác.

Chức năng hàng đầu của GitHub là “forking” – sao chép một kho lưu trữ từ tài khoản của người dùng này sang tài khoản khác. Điều này cho phép bạn tham gia vào một dự án mà bạn không có quyền ghi và sửa đổi trực tiếp lên đó bằng tài khoản của bạn.

Ba tính năng “forking”, “pull request” và “merge” là những gì làm cho GitHub trở nên mạnh mẽ và đặc biệt cuốn hút với các nhà phát triển.

Có 2 cách tiếp cận GitHub: Tạo project của riêng mình Contribute cho project có sẵn: fork project có sẵn của người khác, sửa đổi, sau đó đề nghị họ cập nhật sửa đổi của mình (tạo pull request).

Cách hoạt động:

 Có hai nền tảng để bạn làm việc với **GitHub** là **local workflow** và **server workflow**.

Ở **local workflow**, bạn có thể thay đổi source code. Sau đó xác nhận các thay đổi đó tại **server workflow**. Lưu ý, bản xác nhận trên server phải chứa tính năng hoàn chỉnh hoặc có thể chạy được. Việc xác nhận các đoạn code chưa hoàn chỉnh, còn dở dang sẽ ảnh hưởng các thành viên khác khi họ cùng sử dụng kho lưu trữ.

Từ kho lưu trữ của **GitHub**, bạn có thể tạo bản build cho production site bằng cách gửi các ***source code*** thay đổi lên đó. Khi sử dụng kho lưu trữ của server, người dùng sẽ phải cung cấp mã chứng nhận, hệ thống sẽ so sánh **SSH key** ở **local** của bạn và **SSH key** trên **server** tương ứng với **account** mà đã đăng ký trước đó.

### Thao tác với repository ở local

Với 2 command thường dùng là git add và **git commit**

* **Git add**: add file đã thay đổi vào stage
* **Git commit** các file đã add vào stage lên repository ở local Ngoài ra bạn xem một số command khác

### ***Làm việc với repository ở server GitHub***

Cuối cùng khi có một bản ổn định và hoàn tất ta sẽ quyết định cập nhật nó lên **repository server** với:

* **push**: push thay đổi từ **repository local** lên **repository server**
* **fetch**: cập nhật thay đổi từ **repository server** về **repository local**
* **pull/rebase**: sao chép source code từ **server** về **local workspace** (tương đương checkout của SVN)

Hướng dẫn tải mã nguồn lên Github:

**Bước 1: Tạo tài khoản Github**

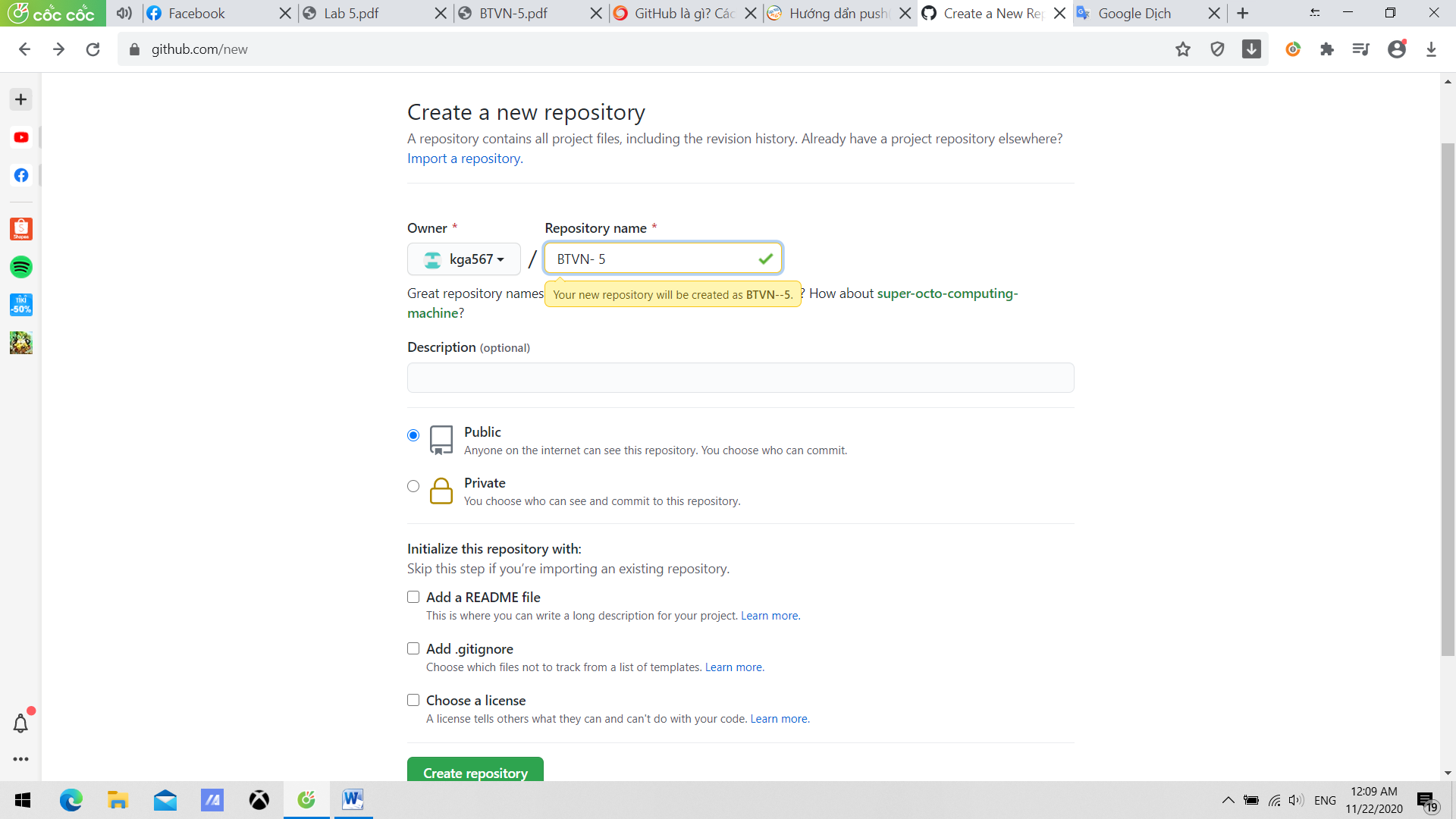
- Các bạn truy cập vào trang **Github.com** và đăng kí tài khoản

**Bước 2: Tạo một repository mới trên Github**

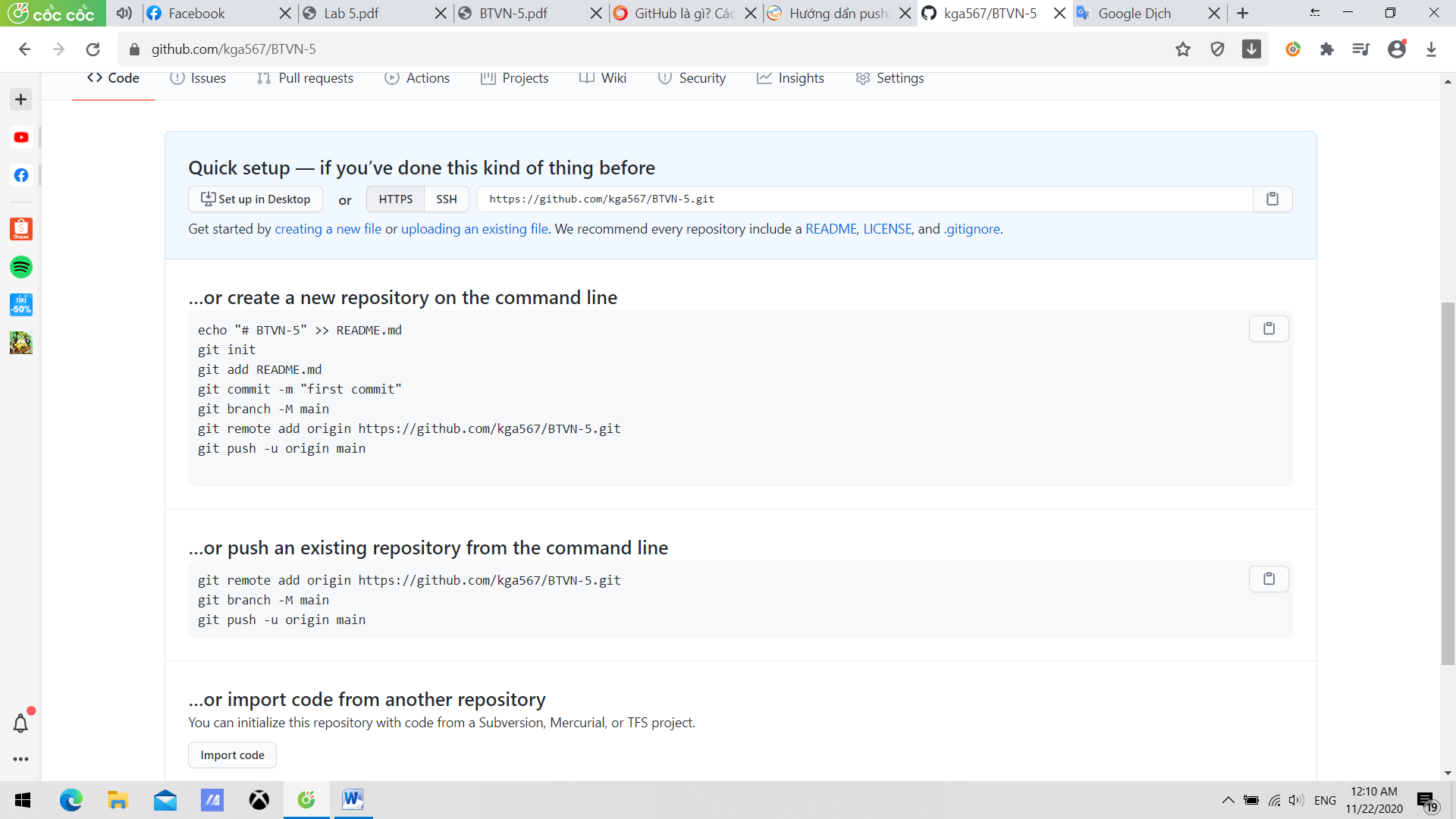
- Sau khi đăng kí tài khoản thành công, tạo một repo mới bằng cách nhấp vào mũi tên nơi avatar, chọn **your repositories**, trang repositories xuất hiện, nhấn nút **New**màu xanh bên phải.

- Đặt tên cho repo mới, sau đó click **Create Repository**.

Lưu ý: Nếu các bạn muốn triển khai một website tĩnh trên Github, các bạn phải đặt tên repo theo cú pháp sau: [username của tài khoản Github của bạn].github.io



 - Sau khi đặt tên cho repo xong thì các bạn click **Create repository**



**Bước 3: Clone repo về máy**

- Copy đường dẫn của repo mới tạo



- Clone repo về máy bằng lệnh sau:

*git clone https://github.com/kga567/BTVN-5.git*

**Bước 4: Thiết lập Git**

Muốn thiết lập tên người dùng / địa chỉ email

*git config --global user.name <username>*

*git config --global user.email <mailaddress>*

Khi không thêm lựa chọn--global thì chỉ những repository có liên quan mới trở nên hữu hiệu với thiết lập.

**Bước 5: Thêm/sửa/xóa file/thư mục trên repo vừa clone về, sau đó push lên Github**

-  Thêm/sửa/xóa các file/thư mục trên repo vừa clone về, sau đó lần lượt chạy từng lệnh sau:

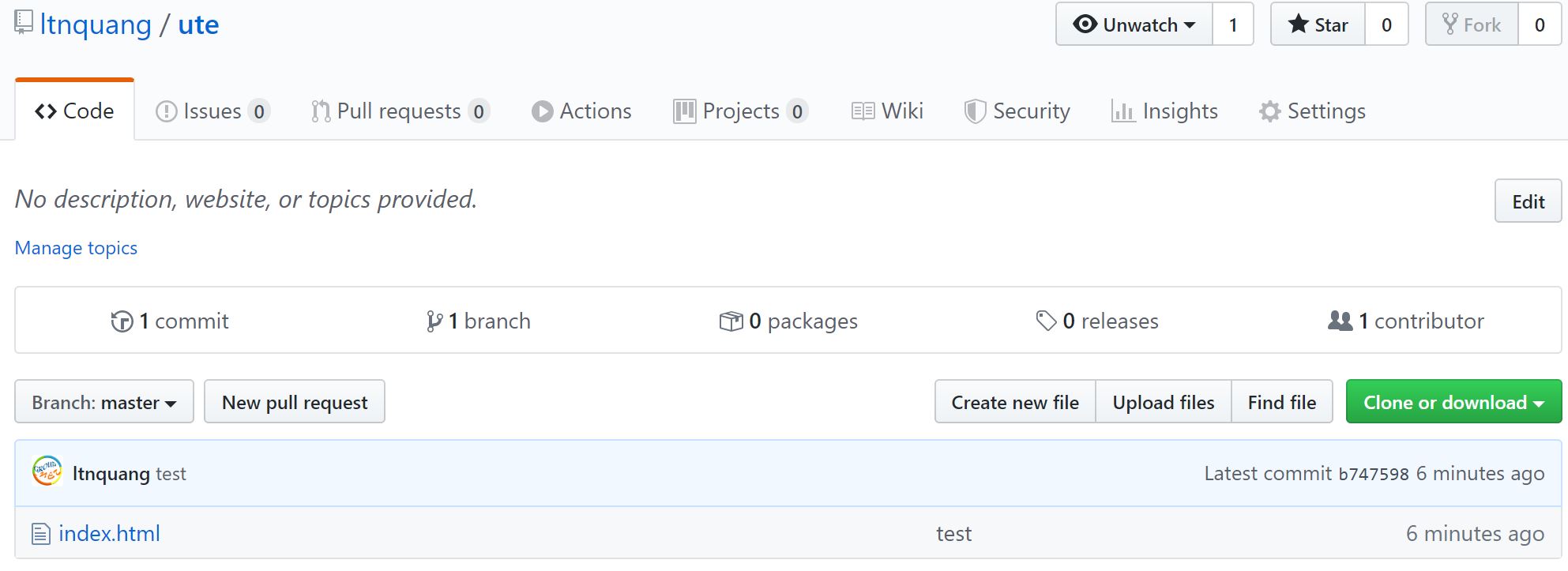
*git add .*

*git commit -m "điền nội dung commit vào đây"*

*git push origin master*

**Chú ý:**Trỏ đúng thư mục mà chúng ta clone git về, ví dụ cd 'tên thư muc' để chuyển đến

- Vào Github repo để kiểm tra:



Nguồn:  
<https://wiki.matbao.net/github-la-gi-cach-chia-se-ma-nguon-tai-code-de-dang-tren-github/?fbclid=IwAR2aTd18fH9VvLD3tD41DAvvz7eSNe0PsfYVMuUALWHaqm944jxozbK6PtM>

<http://dotnet.edu.vn/TinTucND/Huong-dan-pushday-code-len-Github-1884.aspx>