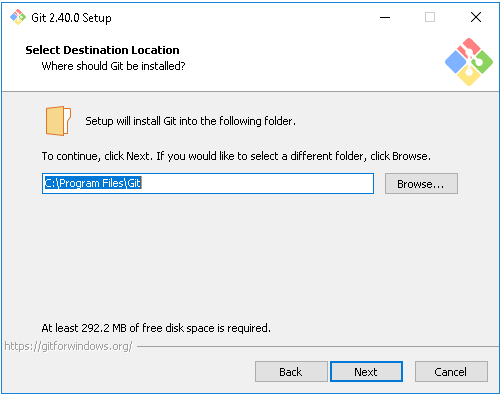
**LƯU TRỮ TÀI NGUYÊN VỚI GITHUB**

GitHub là một dịch vụ nổi tiếng cung cấp kho lưu trữ mã nguồn cho các dự án phần mềm. Đây là công cụ giúp quản lý source code tổ chức theo dạng dữ liệu phân tán và giúp đồng bộ source code của team lên 1 server. Mặc khác, nó hỗ trợ các thao tác kiểm tra source code trong quá trình làm việc

Cài Git từ trang : <https://git-scm.com/download/win>

Lưu ý: ghi nhớ đường dẫn khi cài git

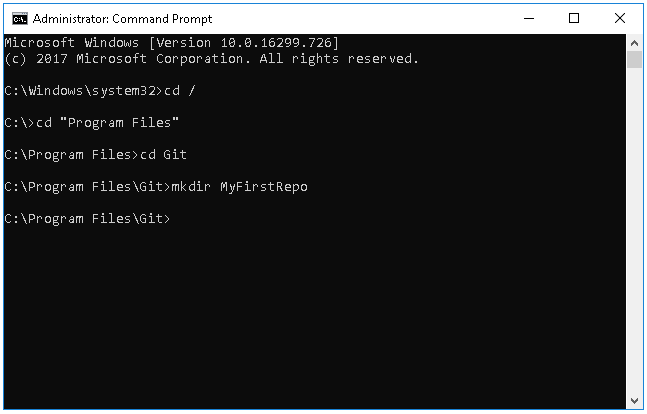


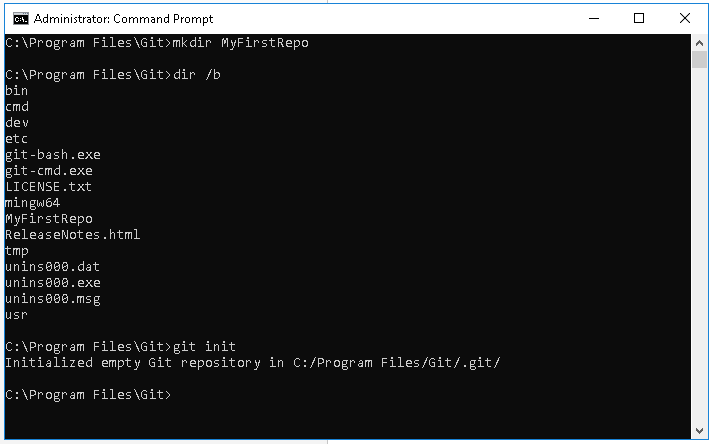
Tạo biến môi trường và đưa vào đường dẫn PATH theo các bước sau:

1. In the Start Menu or taskbar search, search for "environment variable".
2. Select "Edit the system environment variables".
3. Click the "Environment Variables" button at the bottom.
4. Double-click the "Path" entry under "System variables".
5. With the "New" button in the PATH editor, add C:\Program Files\Git\bin\ and C:\Program Files\Git\cmd\ to the end of the list.
6. Close and re-open your console.

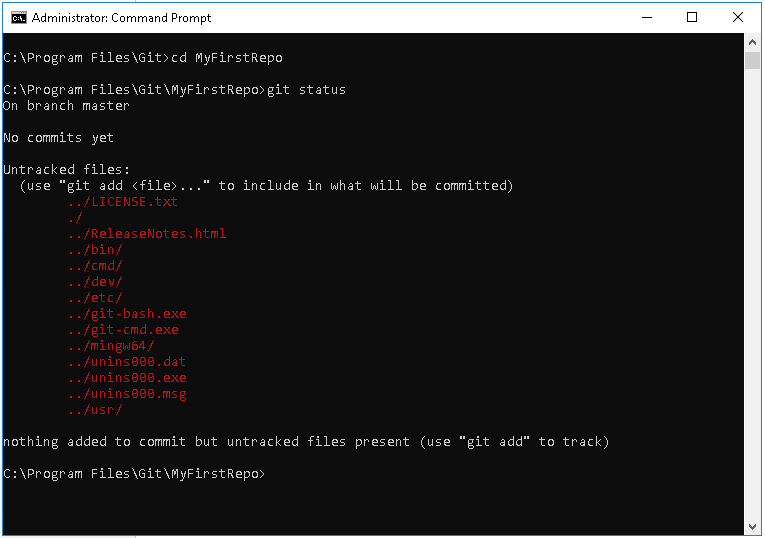
**TẠO THƯ MỤC GIT Ở MÁY TÍNH CÁ NHÂN VÀ ĐỒNG BỘ LÊN TÀI KHOẢN**

Bước 1. Tạo thư mục ở máy tính cá nhân

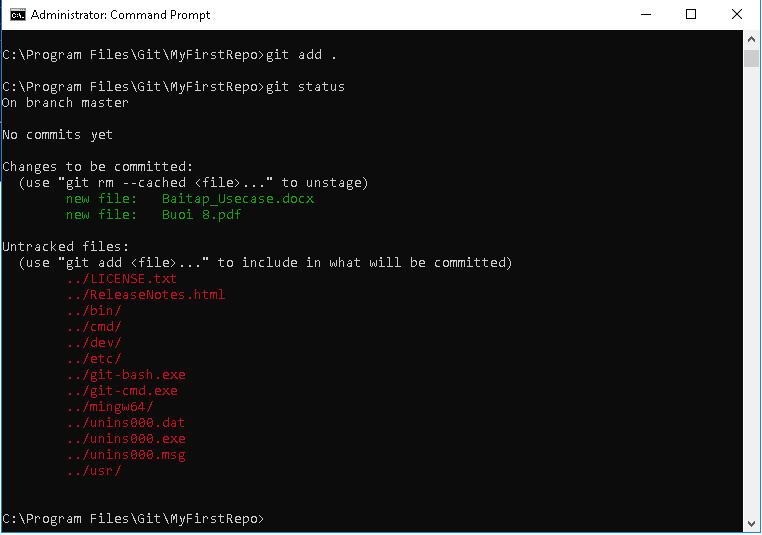


Bước 2: Chép một file docx bất kỳ vào thư mục MyFirstRepo vừa tạo, đồng thời khởi tạo git

Bước 3.Chép một file .pdf bất kỳ vào thư mục Git (MyFirstRepo). Tuy nhiên, Git vẫn chưa thấy file.

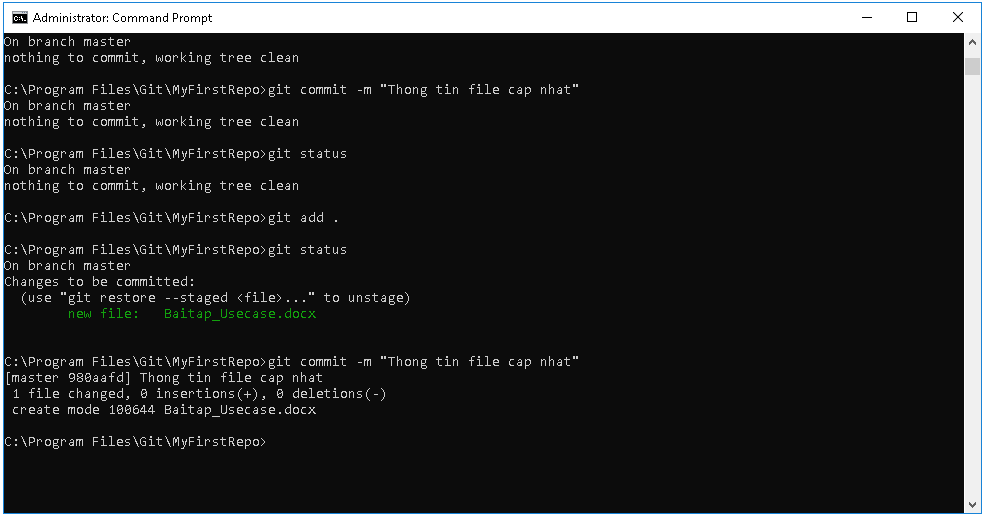


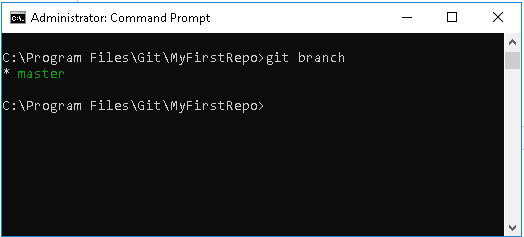
Phải thông báo tường minh cho Git biết là đã thêm file đó.



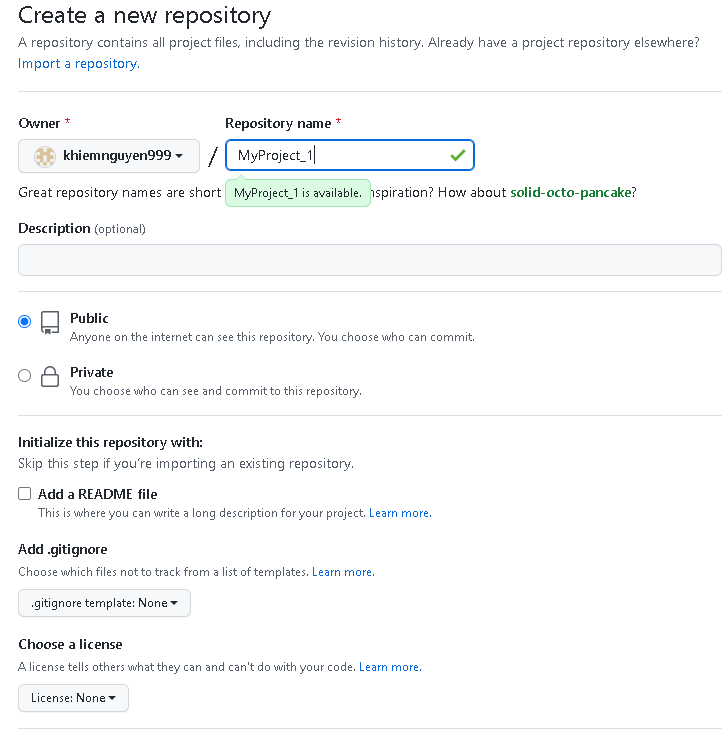
Bước 4: Ghi chú thông tin về các thao tác trên file để đồng bộ với GitHub trên môi trường mạng.

Ở đây, khi máy cục bộ yêu cầu cập nhật thông tin qua lệnh git commit

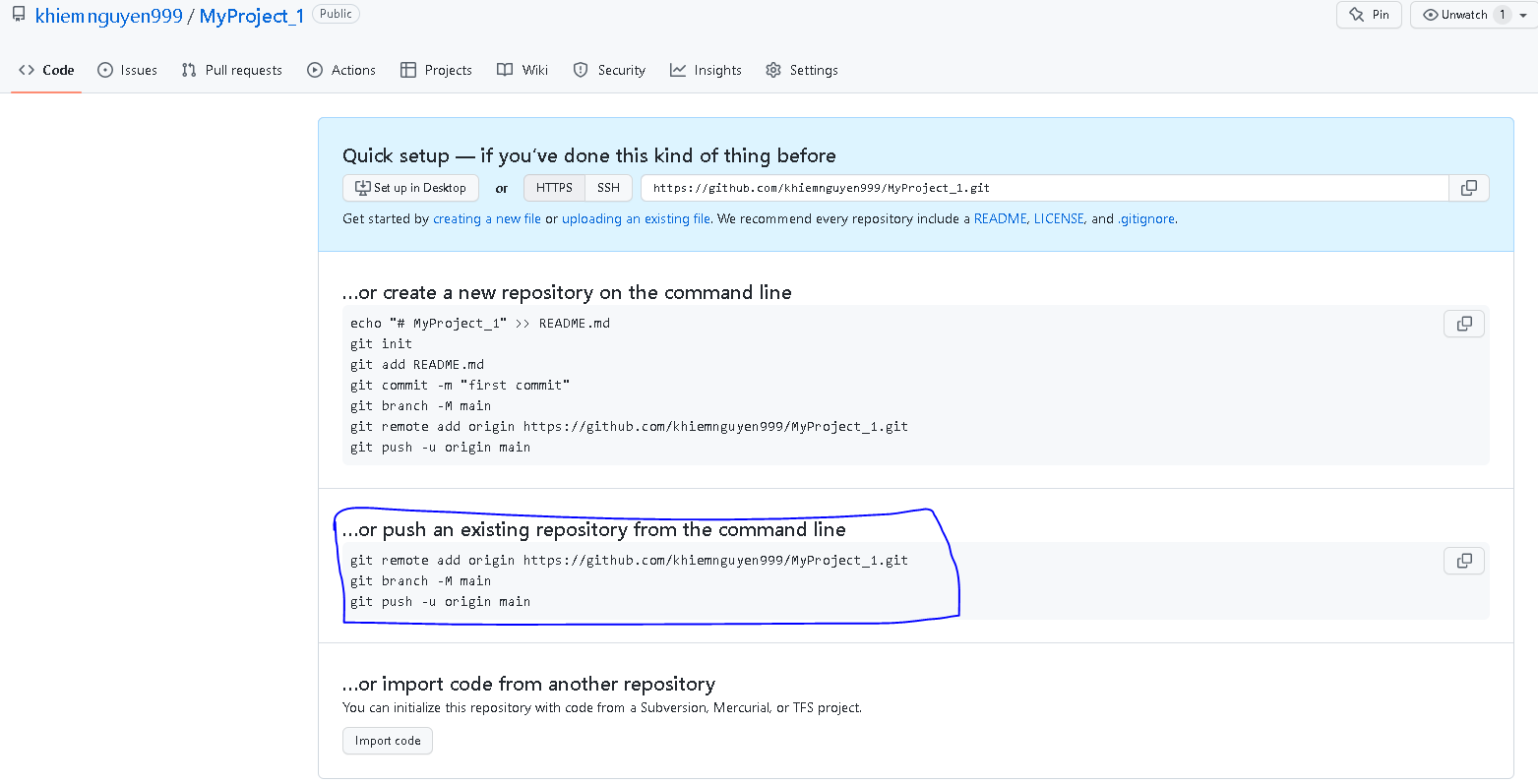


Bước 5: Tạo brach để đồng bộ tài liệu từ thư mục cá nhân MyFirstRepo lên GitHub

Tạo Repository trên Github

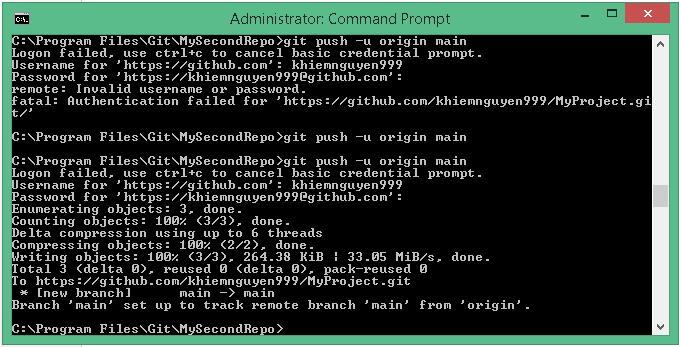


Hãy chạy lệnh được cung cấp từ Github sau khi tạo Repository sau vào của sổ command line

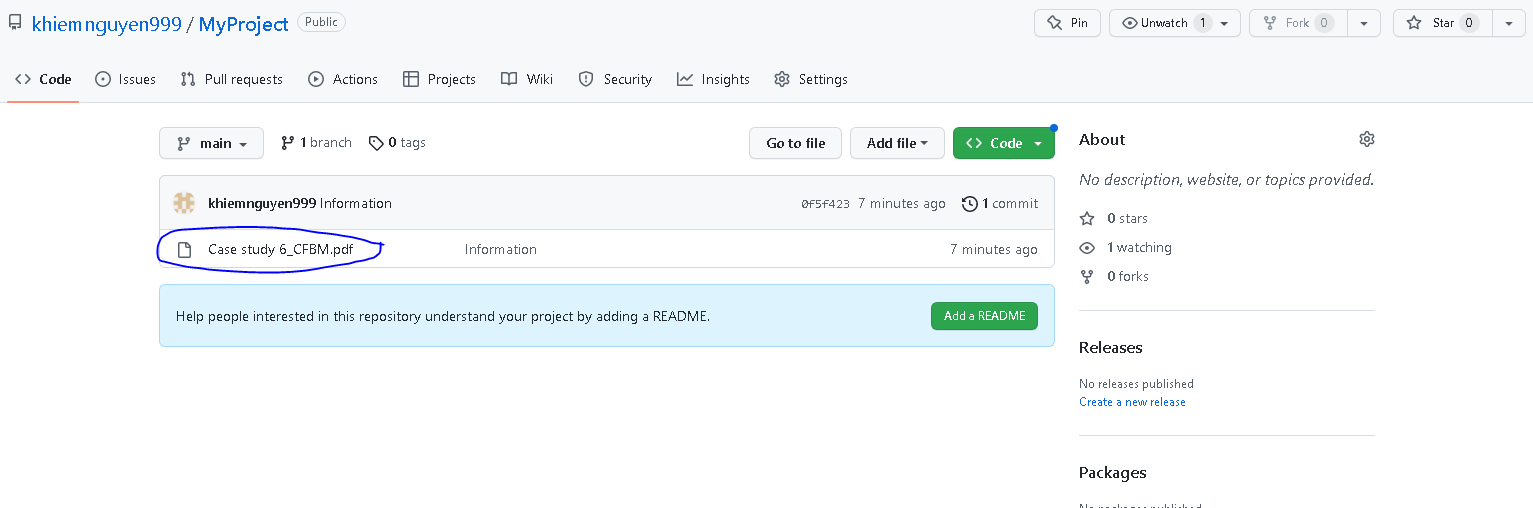


Sau khi chạy các lệnh trên, nếu Github có hỏi về username và password là do lỗi two-factor authentication (2FA). Nếu có lỗi này xảy ra thì khắc phục bằng cách tạo personal key thay thế cho password. Xem chi tiết ở trang: <https://mycyberuniverse.com/how-fix-fatal-authentication-failed-for-https-github-com.html>

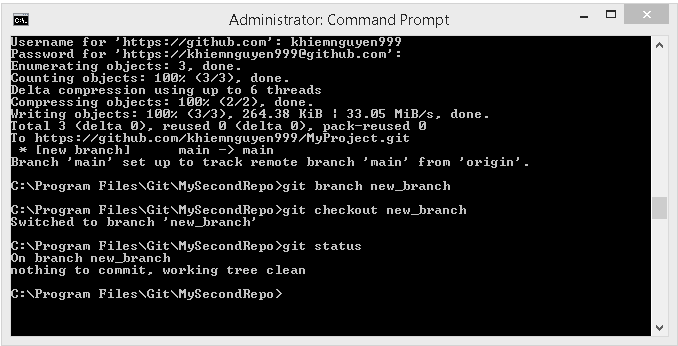
Sau khi lấy được personal Key thì điền vào, kết quả hiển thị như màn hình bên dưới



Vào GitHub xem dữ liệu đã được đồng bộ lên máy tính chưa

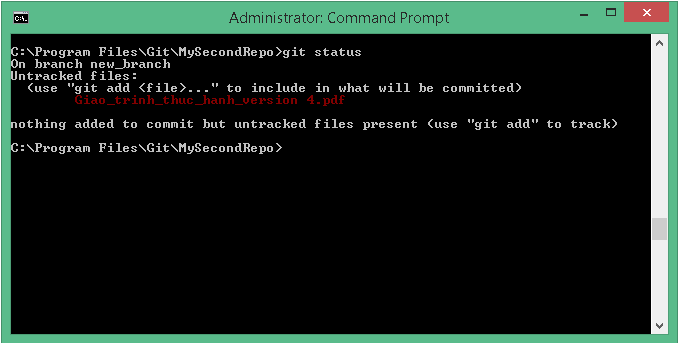


Bước 7: Tạo branch mới để đẩy dữ liệu lên GitHub

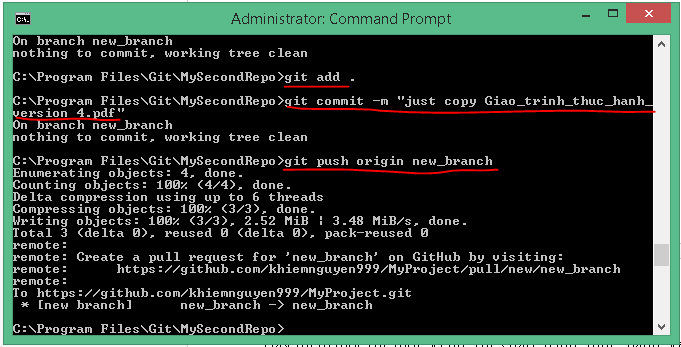


Hãy thêm một file mới, ví dụ, file Giao\_trinh\_thuc\_hanh\_version 4.pdf vào thư mực Git trên máy cá nhân

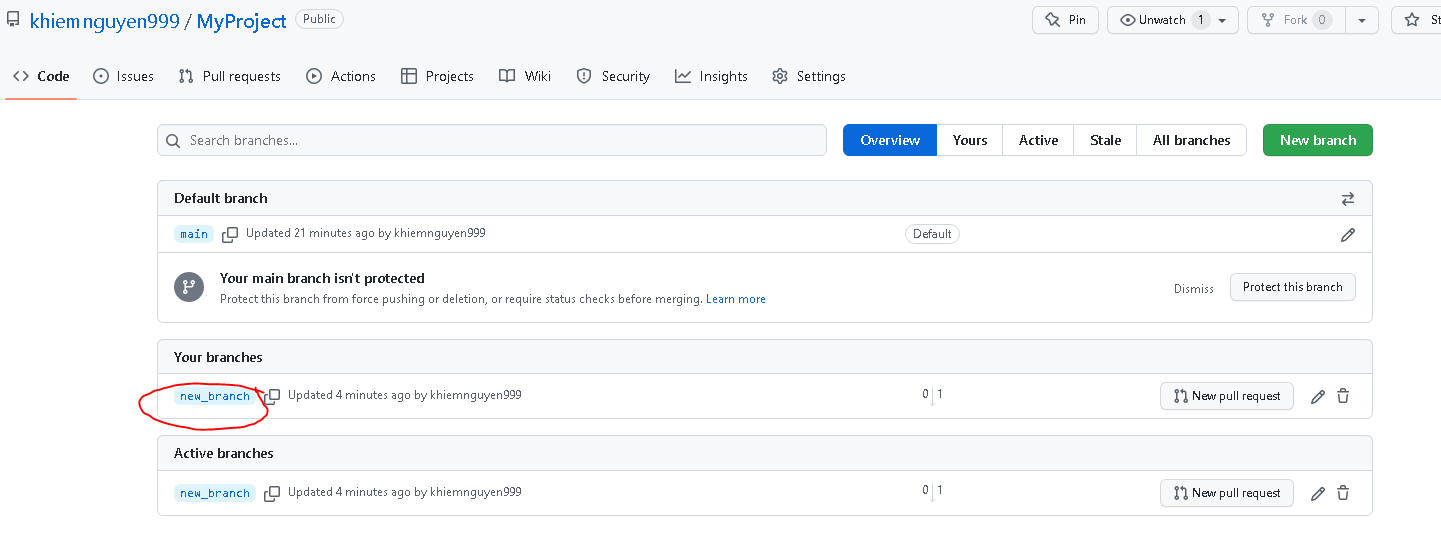
Sau đó chạy lại lệnh: git status



Đẩy dữ liệu lên branch mới (new\_branch)

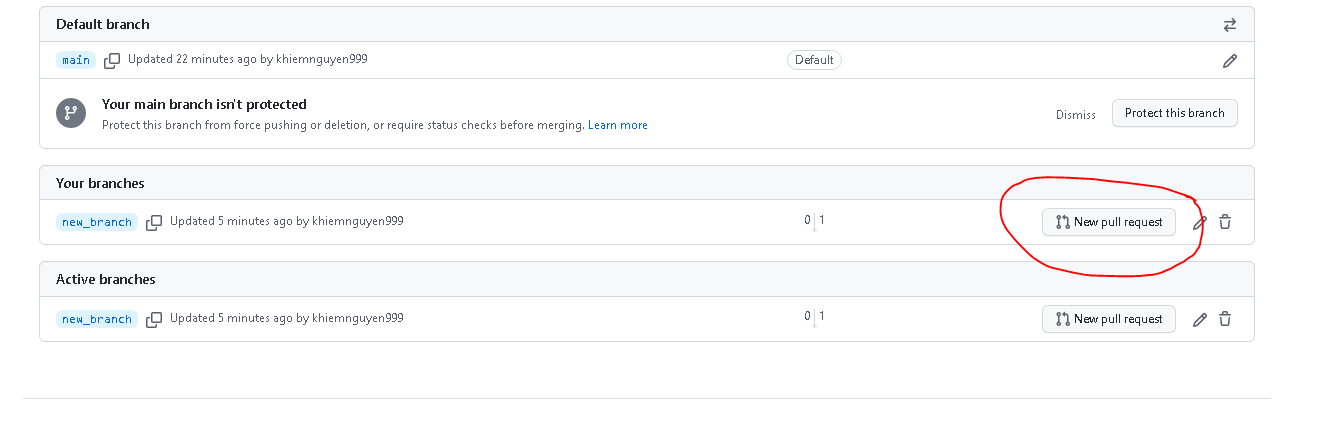


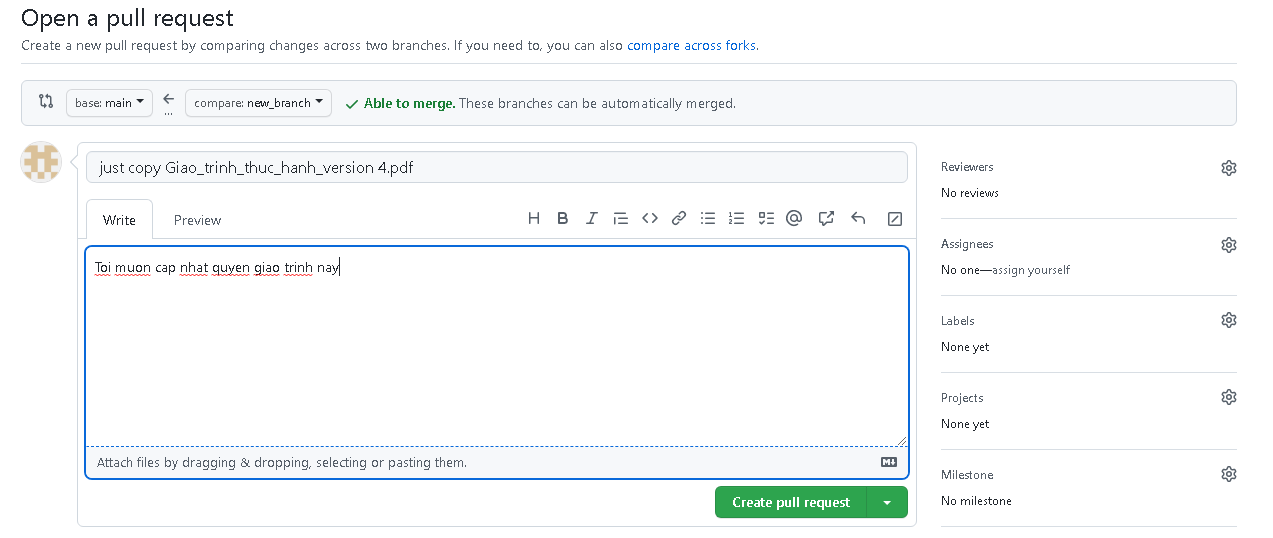
Trên GitHub, 2 branch sẽ xuất hiện



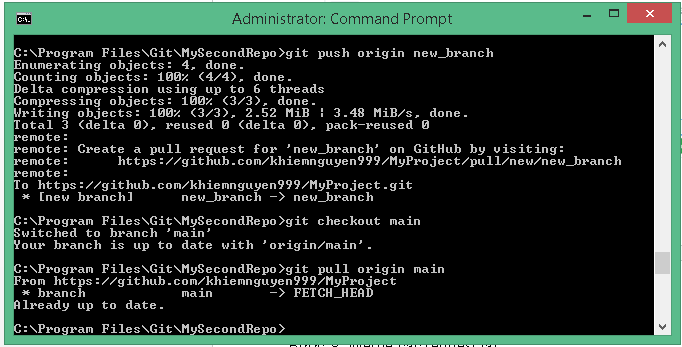
Trong các new\_brach sẽ có tài nguyên mới đẩy từ máy cục bộ lên

Bước 8: Tạo một yêu cầu đẩy dữ liệu lên

 Sau đó thêm các yêu cầu



Bước 9: Merge các request lại để lưu các thay đổi từ các người dùng khác nhau



**TẠO NODE JS, LƯU TRỮ TRONG GITHUB VÀ TRIỂN KHAI LÊN HEROKU**

Nodejs là một môi trường runtime chạy [JavaScript](https://vietnix.vn/javascript-la-gi/) đa nền tảng và có mã nguồn mở, được sử dụng để chạy các ứng dụng web bên ngoài trình duyệt của client. Nền tảng này được xem là một giải pháp hoàn hảo cho các ứng dụng sử dụng nhiều dữ liệu nhờ vào mô hình hướng sự kiện (event-driven) không đồng bộ.

1. Tải nodejs ở địa chỉ:

<https://nodejs.org/en>

Tiến hành cài đặt Nodejs. Giả sử, thư mục cài đặt nodejs ở …

2. Tạo file package.json bằng lệnh npm init

Ví dụ, package.json sẽ có dạng

{

"name": "coolnodeapp",

"version": "1.0.0",

"description": "node app ",

"main": "app.js",

"scripts": {

**"start": "node app.js"**

},

"repository": {

"type": "git",

"url": ""

},

"author": "",

"license": "ISC",

"bugs": {

"url": ""

},

"homepage": ""

}

Quan trọng nhất là :

"start": "node app.js"

3. Tại file app.js như sau. Chú ý, file app.js và package.js nên để cùng ở một thư mục

const http = require('http');

const port = process.env.PORT || 3000

const server = http.createServer((req, res) => {

res.statusCode = 200;

res.setHeader('Content-Type', 'text/html');

res.end('<h1>Hello World</h1>');

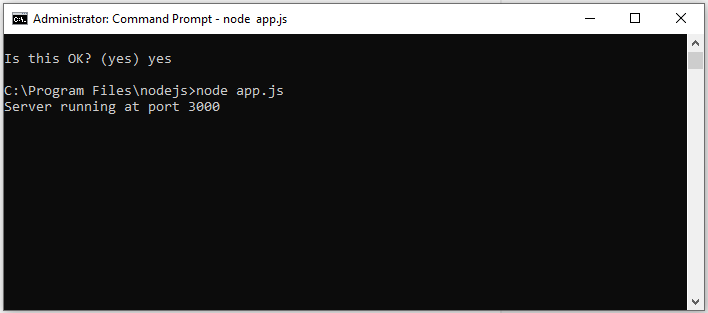
});

server.listen(port,() => {

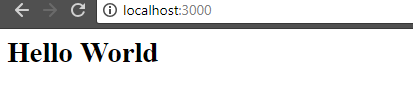
console.log(`Server running at port `+port);

});

Test app.js bằng cách gọi: node app.js



Và trên trình duyệt web gõ: <http://localhost:3000/>



Ở đây, anh/chị có thể tạo trang web tùy ý mình!!

4. Đẩy trang app.js lên Git

Đứng ở đường dẫn chứa file app.js, thực hiện các lệnh sau

git init

git add .

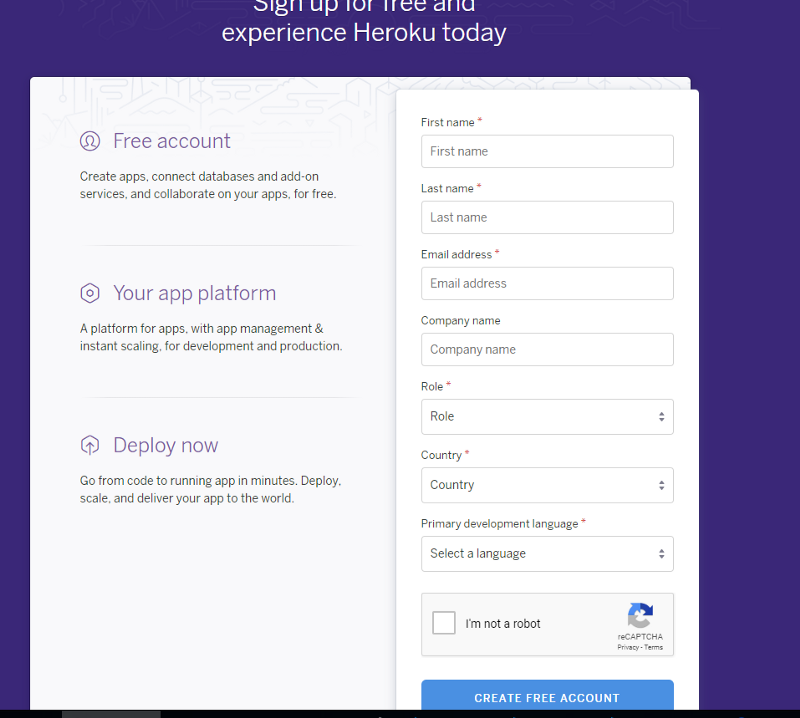
git remote add origin [https://github.com/<tencua](https://github.com/%3ctencua)USER>/<REPOSITORY\_CUAMINH>.git

git push — set-upstream origin master

Sau lệnh trên, sẽ thấy source code được up lên trang GitHub

5. Triển khai lên Heroku (tùy chọn, nếu có tính phí thì thôi ☺)

Tạo tài khoản Heroku tại: <https://signup.heroku.com/login> (tạo free, không cần điền credit card)



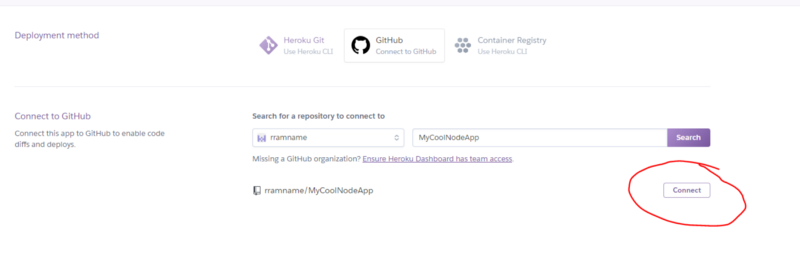
Sau khi tạo xong tài khoản, chọn New ở góc trên bên phải và “Create New app”

Đặt tên cho ứng dụng (nên bao gồm URL dẫn đến ứng dụng) 🡪 Click chọn Create app

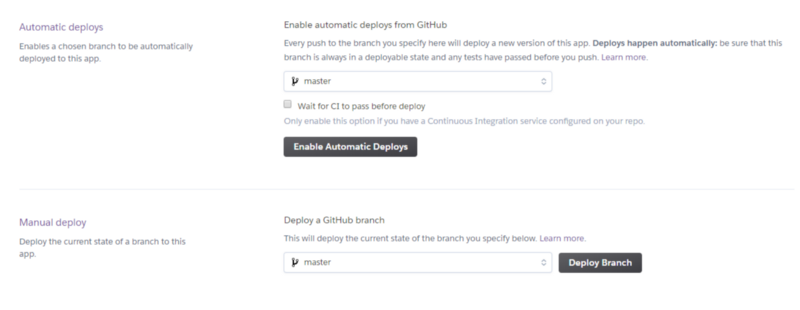
Trang dashboard mở ra. Chọn tabl Deploy và kéo xuống chọn “Deployment method”

Chọn GitHub để triển khai

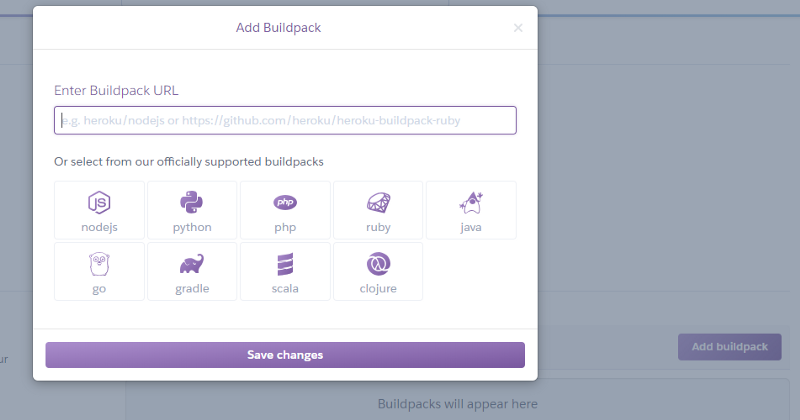
Kết quả tương tự như hình sau:



Sau khi kết nối xong với GitHub repository, click “Enable Automatic Deploys”



Tiếp tục mở tab settings và chọn Buildpacks, click chọn “Add buildpack”



Chọn Nodejs và Save Changes

Trở lại tab Deploy và click “Deploy Branch”

Mở tab seetings, chọn “Domais and certificates”. Ở đây thì URL mới của trang web app.js sẽ được tạo. Copy URL này dán vào trình duyệt web thì sẽ có kết quả như sau

