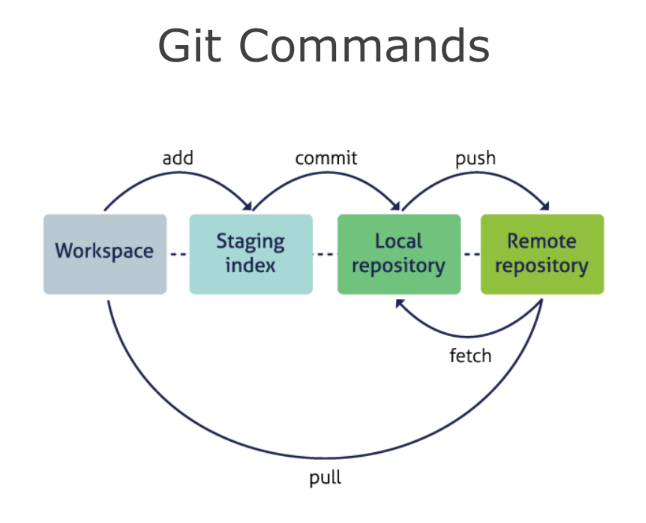
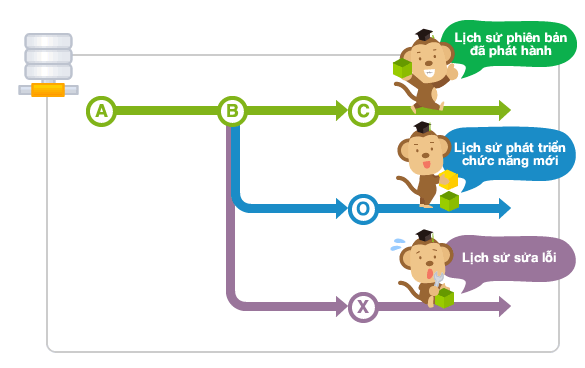
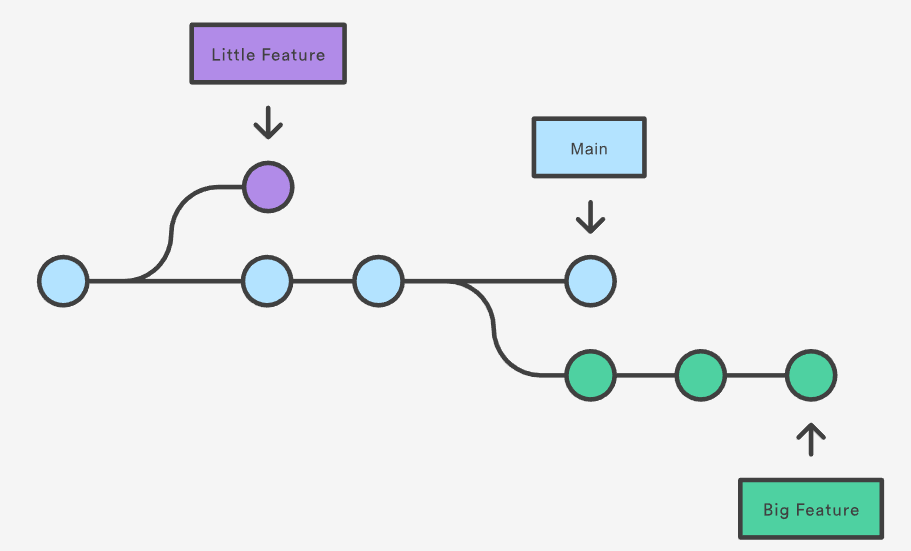
**Lab1. Hướng dẫn sử dụng Git**

1. **Nội dung chính:**

* Hướng dẫn sử dụng Git trong việc quản lý Source Code.
* Git là hệ thống kiểm soát phiên bản phân tán mã nguồn mở.
* Git được sử dụng để lưu trữ nội dung, source code.

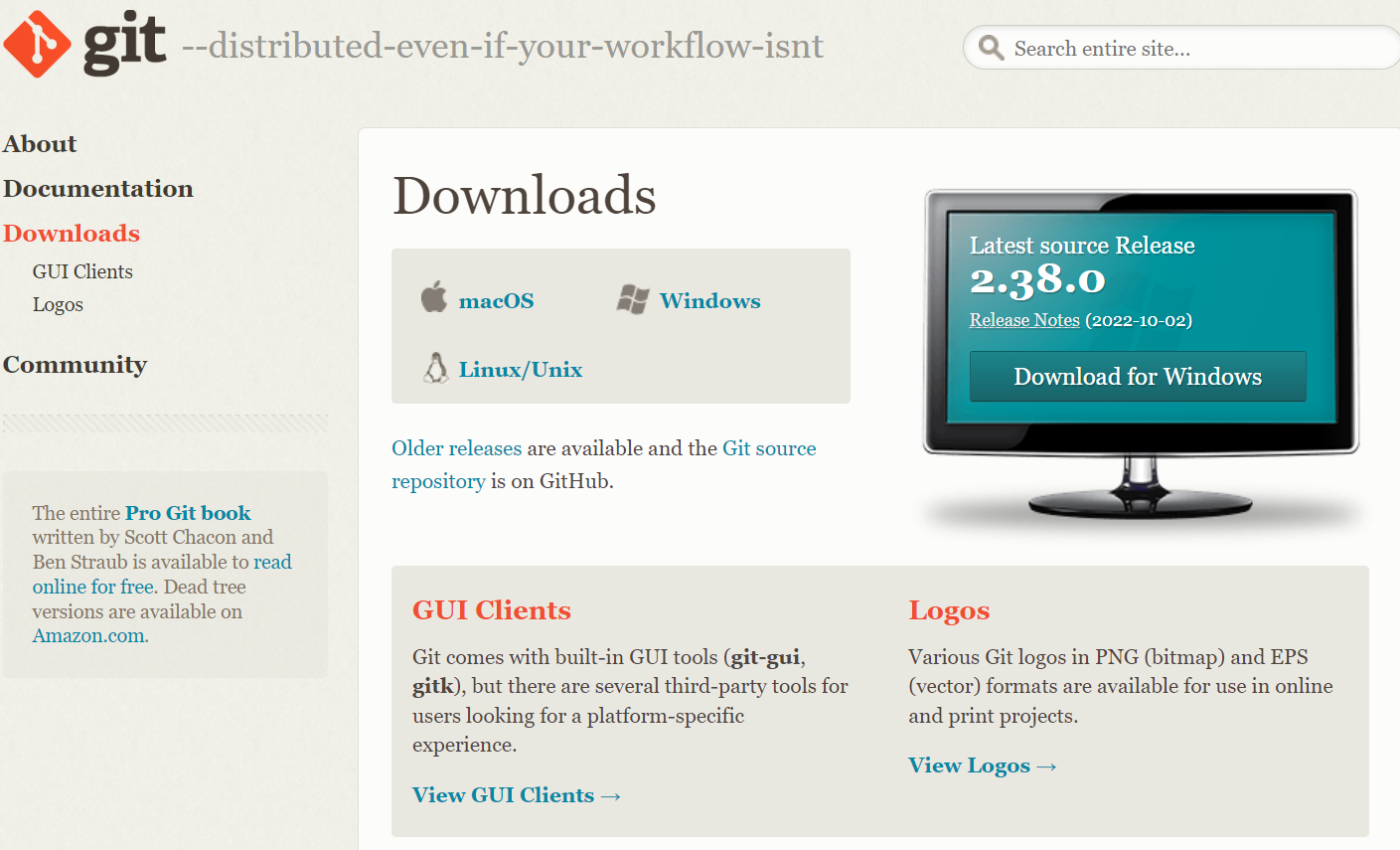


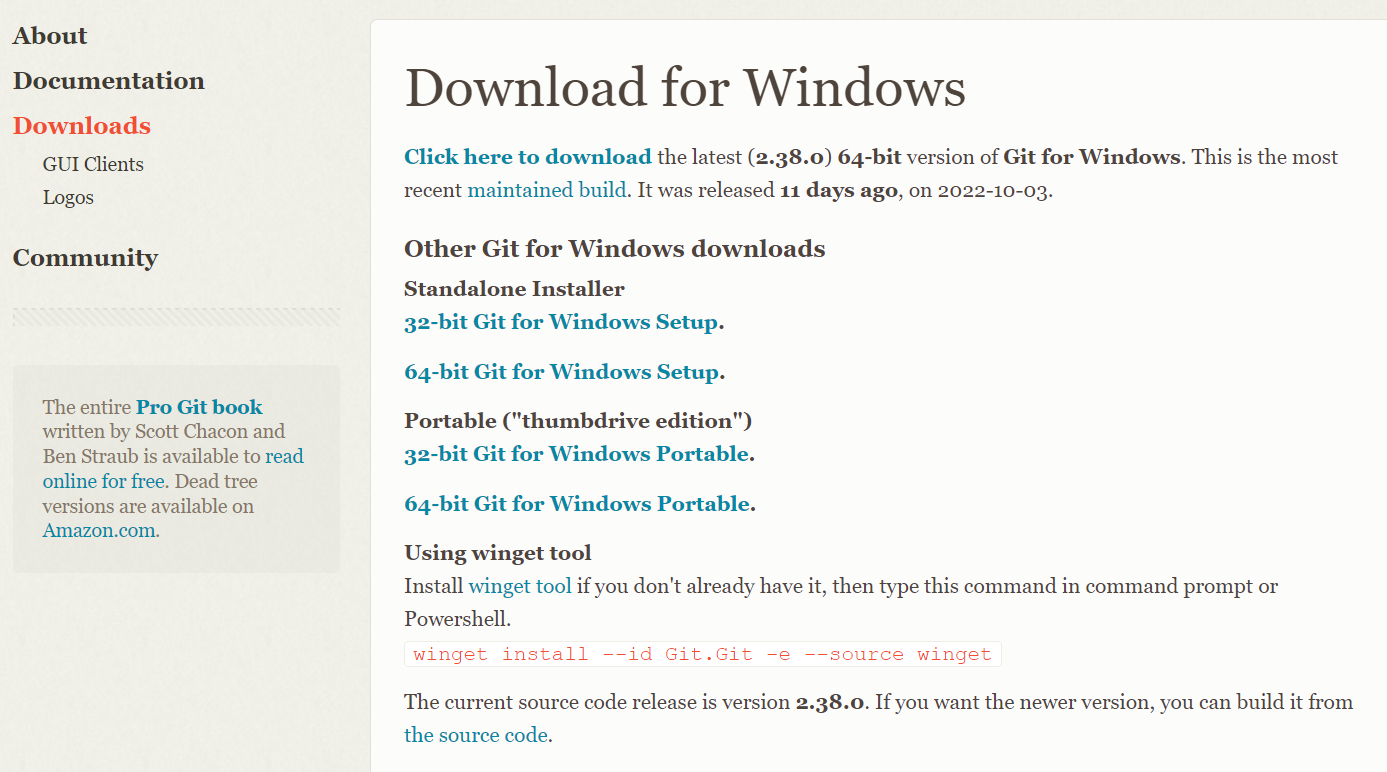




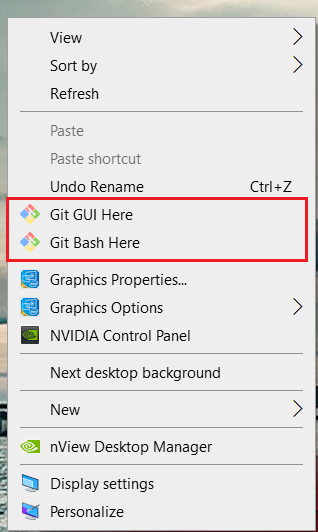
1. **Cài đặt Git:**

* Thực hiện download:<https://git-scm.com/downloads> :





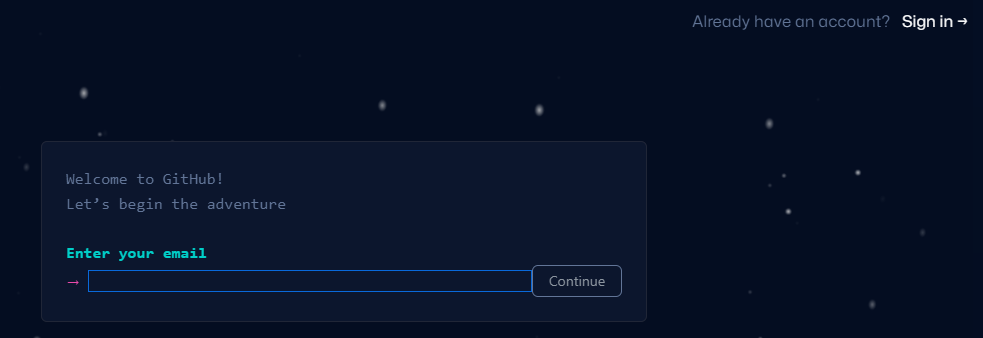
* Lựa chọn phiên bản phù hợp với hệ điều hành.
* Thực hiện cài đặt như những phần mềm bình thường.
* Sau khi cài đặt xong, kiểm tra lại menu chuột phải:



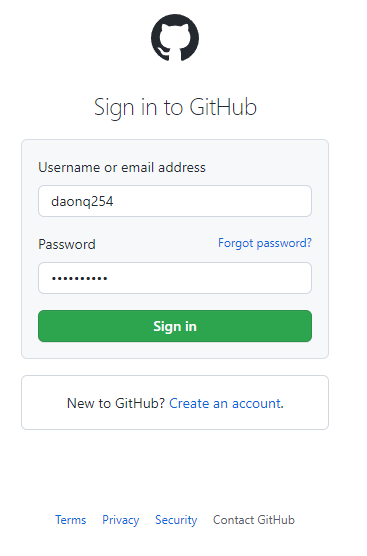
1. **Đăng ký tài khoản GitHub tạo Repository:**

* **Đăng ký tài khoản GitHub**
* Vào đường link: <https://github.com/>

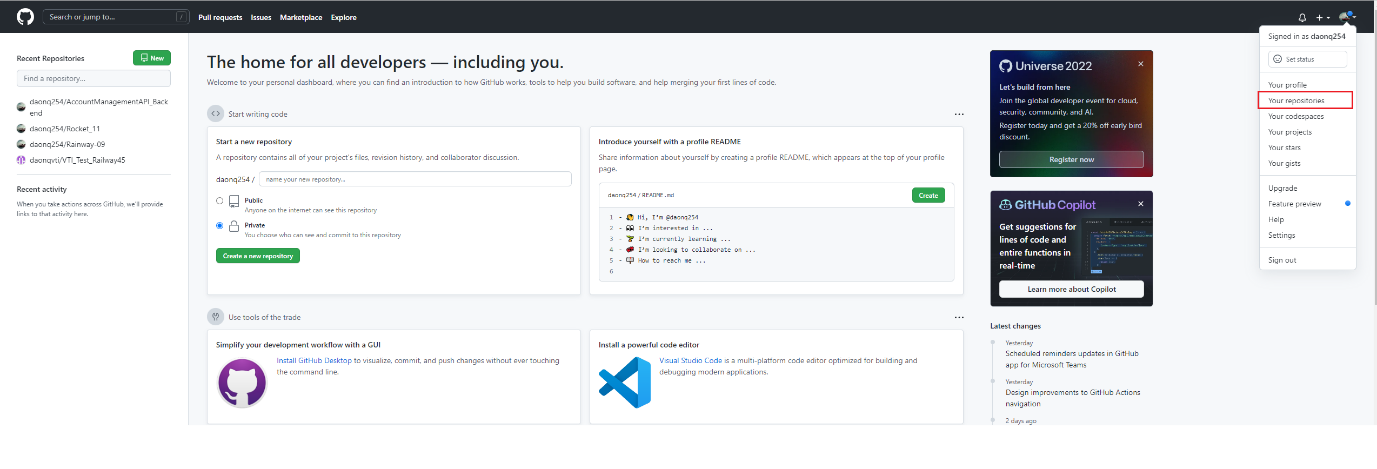




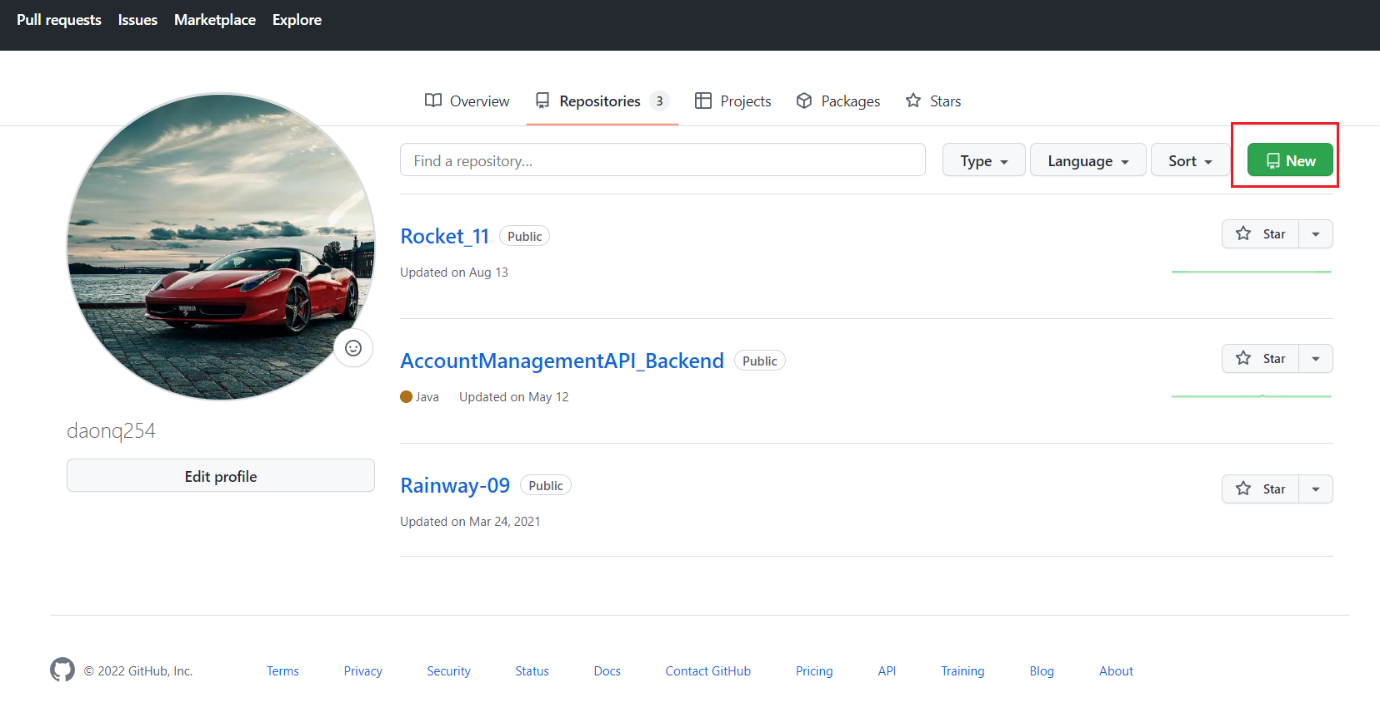
* Thực hiện điền các thông tin Emai, Username, Password để đăng ký tài khoản.
* Sau khi đăng ký xong thực hiện lưu trữ lại các thông tin này.
* Thực hiện đăng nhập vào tài khoản vừa tạo.



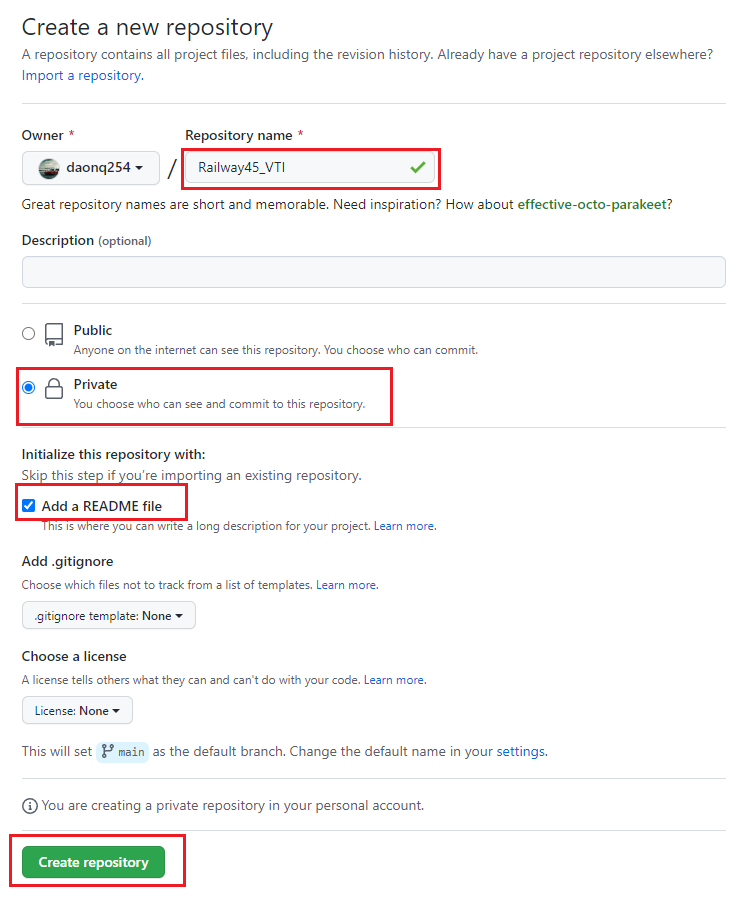
* **Tạo Repository mới**
* Chọn như hình.



* Chọn New.



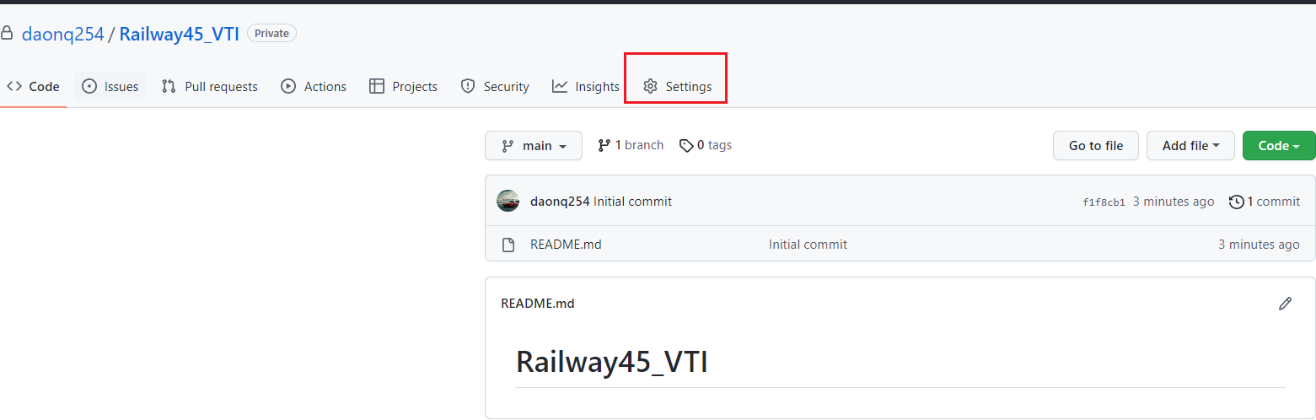
* Điền các thông tin để tạo ra 1 Repository mới với tên Railway45\_VTI



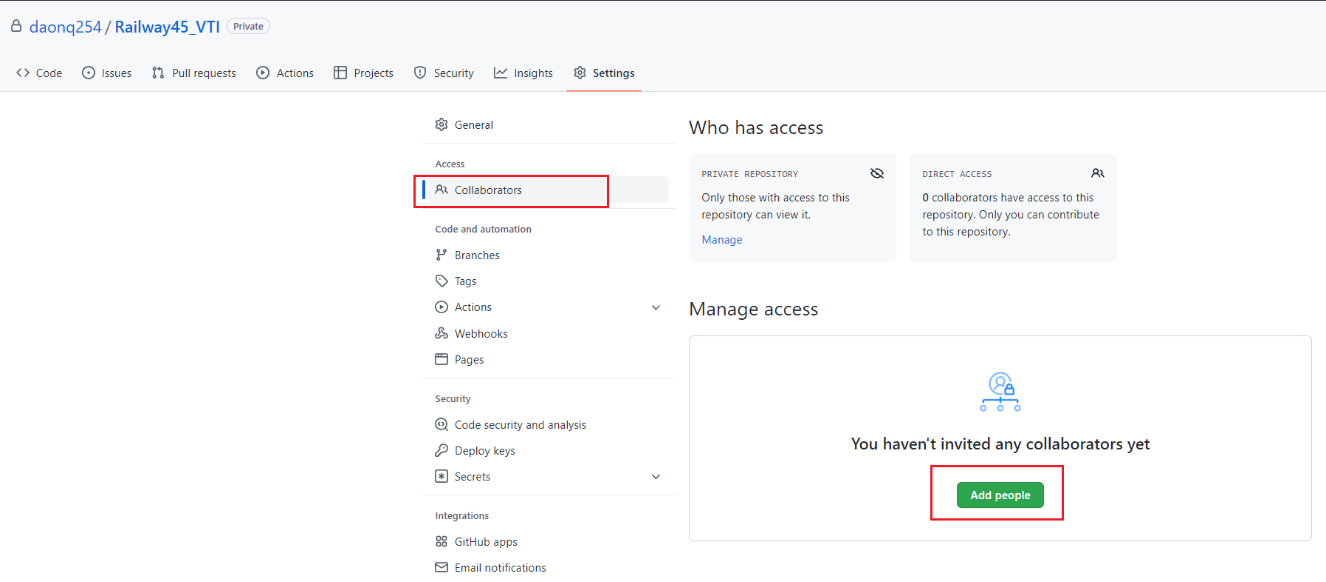
* Kết quả:



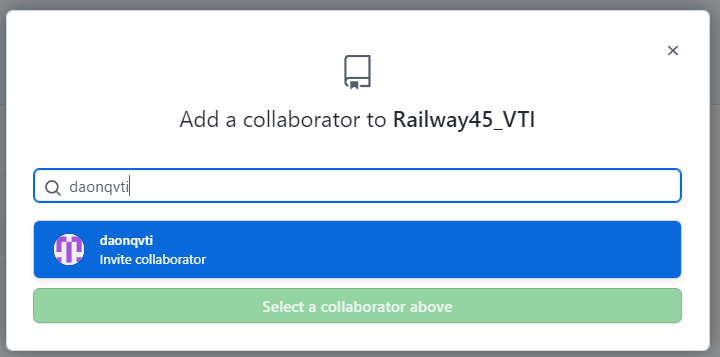
* **Add thêm các thành viên trong Team vào Git:**
* Chọn Tab Setting:



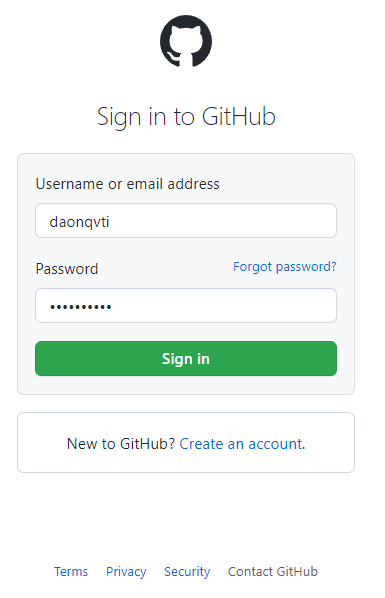
* Chọn tiếp



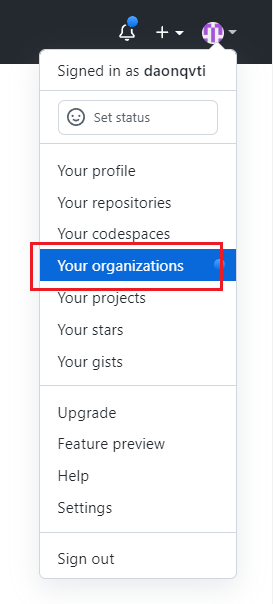
* Thực hiện Add các thành viên vào Github, add theo Username



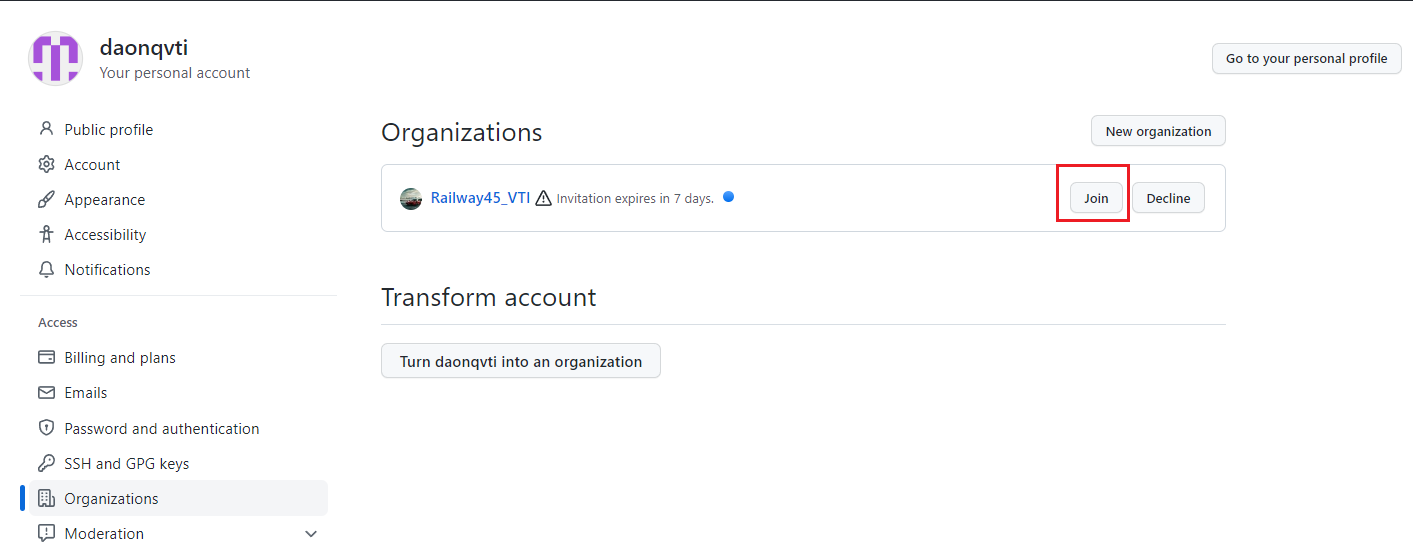
* Với các thành viên sau khi nhận được Invite cần Accept để join vào GitHub
* Thành viên login vào Tk của mình:

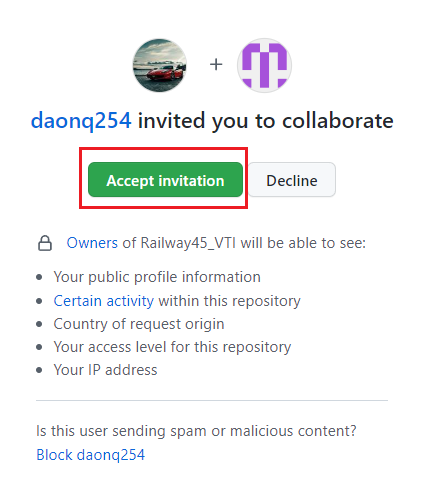


* Sau đó chọn:



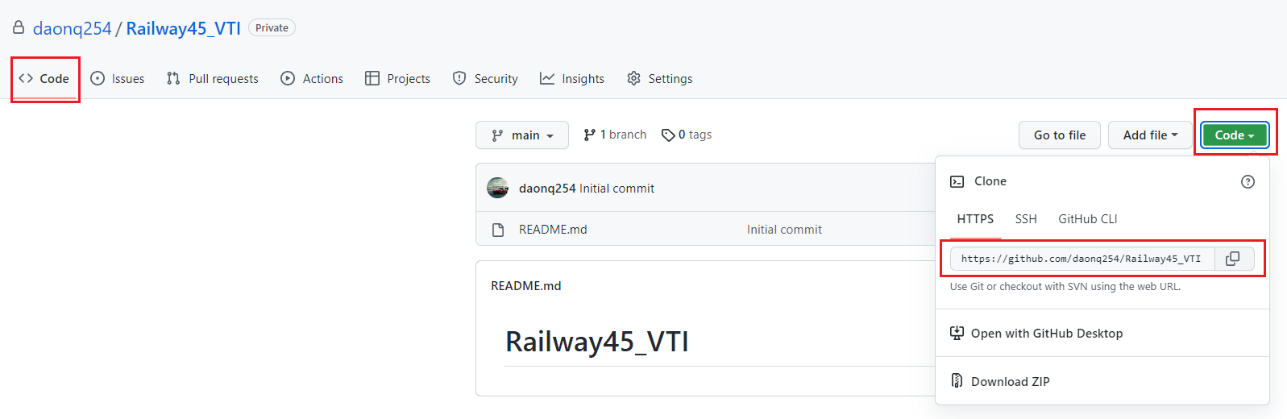
* Chọn tiếp để join.



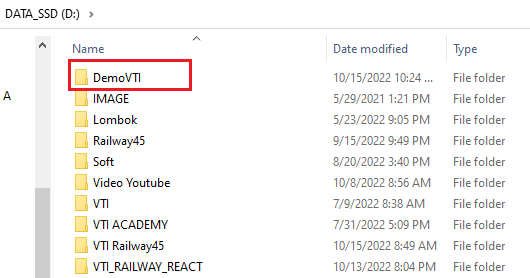


1. **Clone Repository từ Server về máy:**

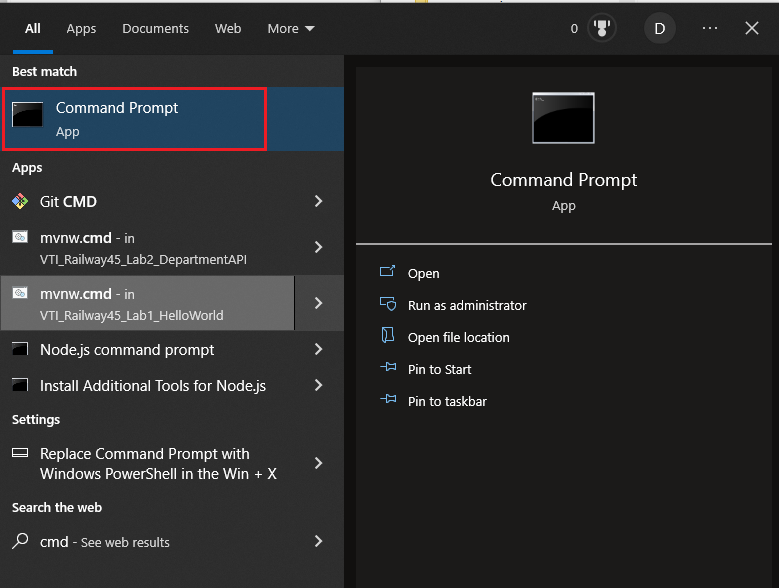
* Thực hiện Copy lại đường dẫn Repository để Clone về máy:

****

* Trên máy tính thực hiện tạo 1 thư mục: **DemoVTI**

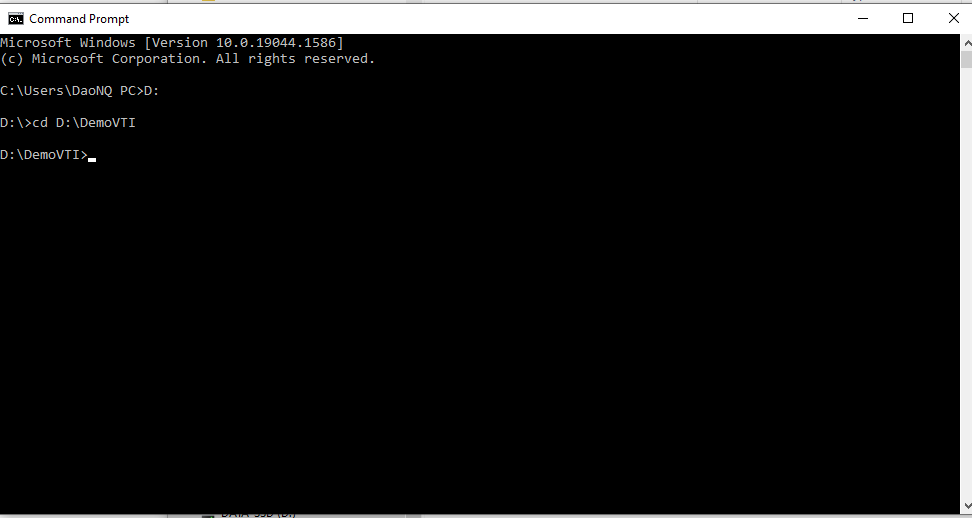


* Mở cửa sổ cmd trên máy:



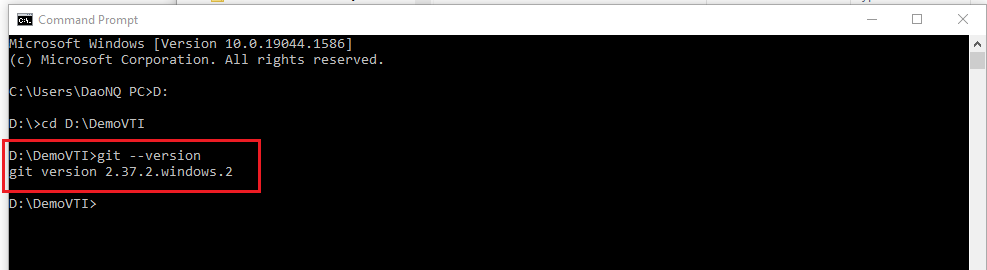
* Chuyển đến thư mục vừa tạo bên trên:

|  |
| --- |
| D:  cd D:\DemoVTI |



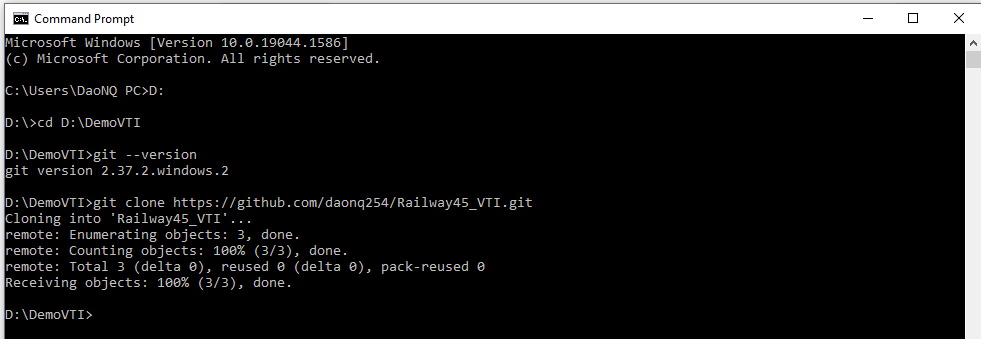
* Kiểm tra lại lần nữa xem Git đã được cài đặt hay chưa:

|  |
| --- |
| git --version |

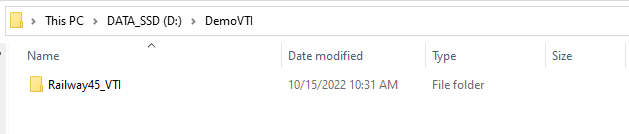


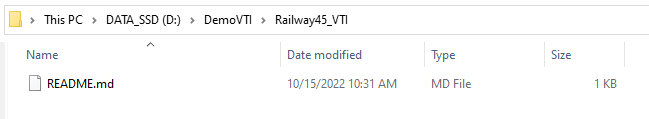
* Clone Repository về máy:

|  |
| --- |
| git clone <https://github.com/daonq254/Railway45_VTI.git>   * Đường link Repository lấy được từ bước trên. |



* Kiểm tra lại kết quả:



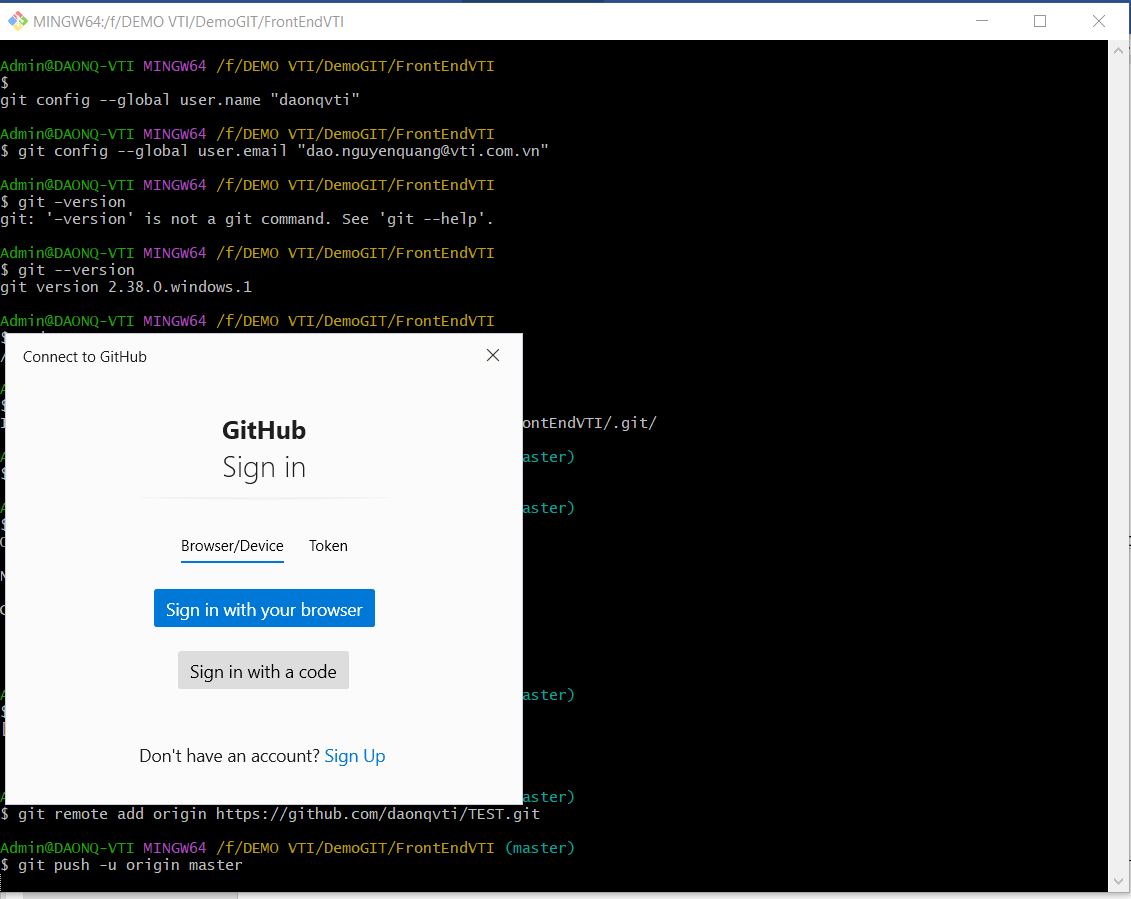


1. **Các lệnh hay sử dụng khi làm việc với Git**

* Các lệnh.

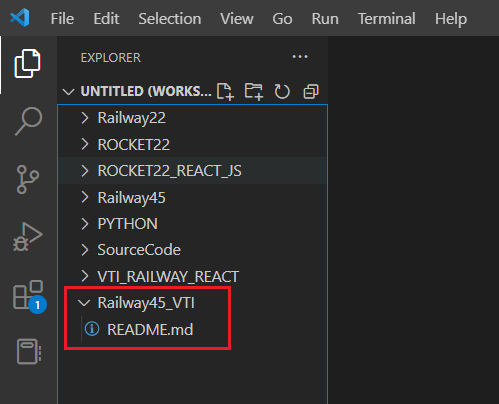
|  |
| --- |
| **🡺 Cài đặt Username, Password khi làm việc với Git**  git config --global user.name "daonq254"  git config --global user.email "[dao.nq254@gmail.com](mailto:dao.nq254@gmail.com)"  git config --global user.name "daonqvti"  git config --global user.email "[dao.nguyenquang@vti.com.vn](mailto:dao.nguyenquang@vti.com.vn)"  **🡺 Kiểm tra phiên bản Git**  git –version  **🡺 Tạo thư mục lưu trữ và add lên Local Repository**  Tạo thư mục DemoCode, sau đó vào trong thư mục và chạy lệnh  git init  **🡺 Add thêm file vào Stage**  git add demo.txt  git add file1 file2 file3 🡺 add nhiều file  git add . 🡺 add tất cả các file  **🡺 Lưu vào Local Repository**  git commit -m "message"  **🡺 Xem trạng thái các file**  git status  🡺 Có thể viết tắt  git add demo.txt git commit -m "message"  **🡺 Clone Repository**  git clone <https://github.com/daonqvti/VTI_Test_Railway45.git>  git clone -b <branch> <remote\_repo> 🡺Clone ở 1 nhánh nào đó  git clone -b master <https://github.com/daonqvti/TEST.git>  git clone -b daonq254 <https://github.com/daonqvti/TEST.git>  **🡺 Tạo nhánh**  git branch daonq254  git checkout daonq254 🡺 Chuyển qua 1 nhánh để làm việc  **🡺 Reset về phiên bản trước**  git log 🡺 kiểm tra lại lịch sử phiên bản  git reset –hard CodeVersion 🡺 Chuyển về 1 phiên bản trước đó: CodeVersion: Lấy từ lệnh git log  **🡺 GitHub**  **🡺 Trỏ Local Repository tới Remote Repository**  git remote add origin [repository url]  🡺 **push source từ Local Repository tới nhánh main**  git push  **🡺 push source từ Local Repository tới GitHub, nhánh master**  git push -u origin master  **🡺 Kéo code từ GitHub về Local**  git pull origin master  **🡺 Xem thông tin user git**  git config --list  git config user.email  git config user.name |

* Khi sử dụng git lần đầu, sẽ yêu cầu đăng nhập

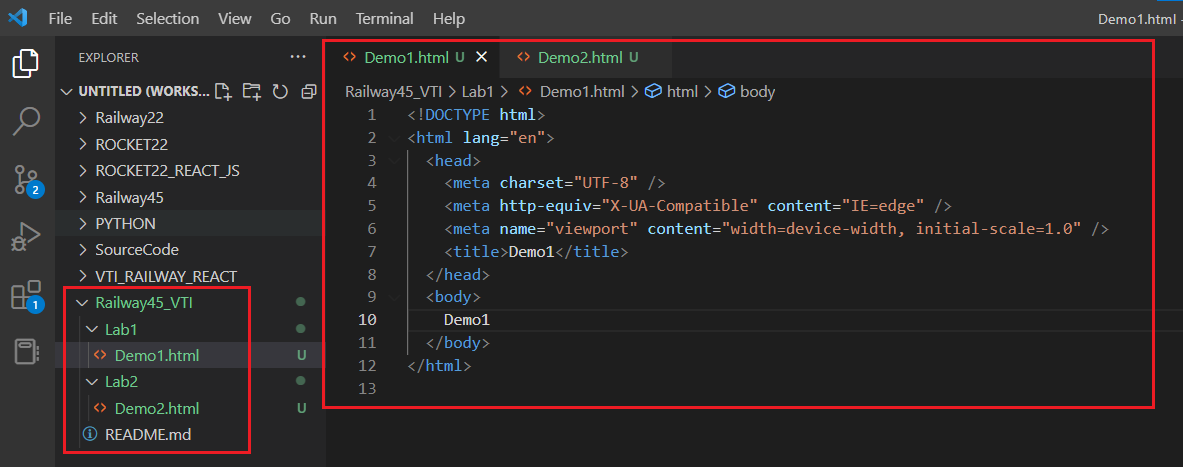


1. **Sử dụng Git cùng Visual Studio Code:**

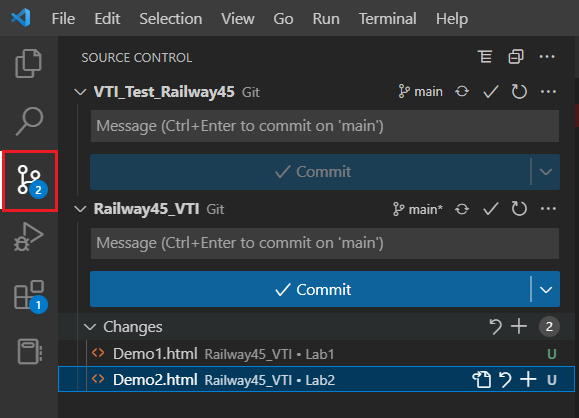
* Thực hiện mở thư mục vừa Clone trong VS Code.



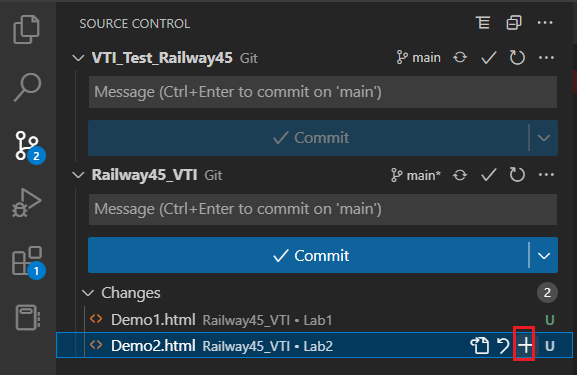
* Thực hiện tạo 1 vài file để Test

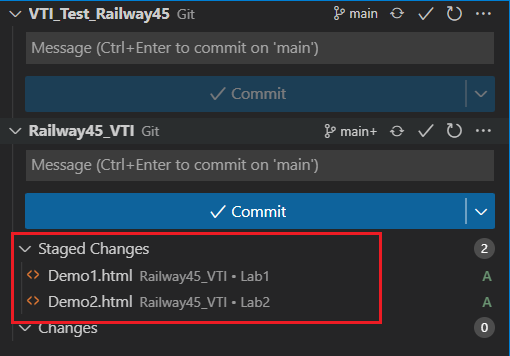


* Thực hiện các bước để đưa file code vừa tạo lên GitHub
* Chọn:



* Chọn: Để lưu các thay đổi vào Stage

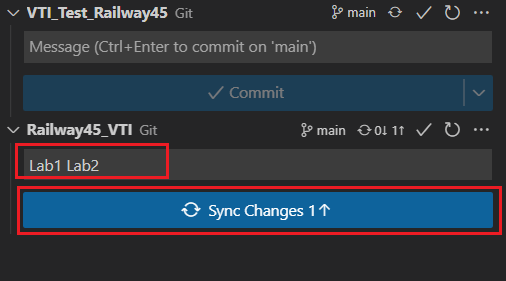


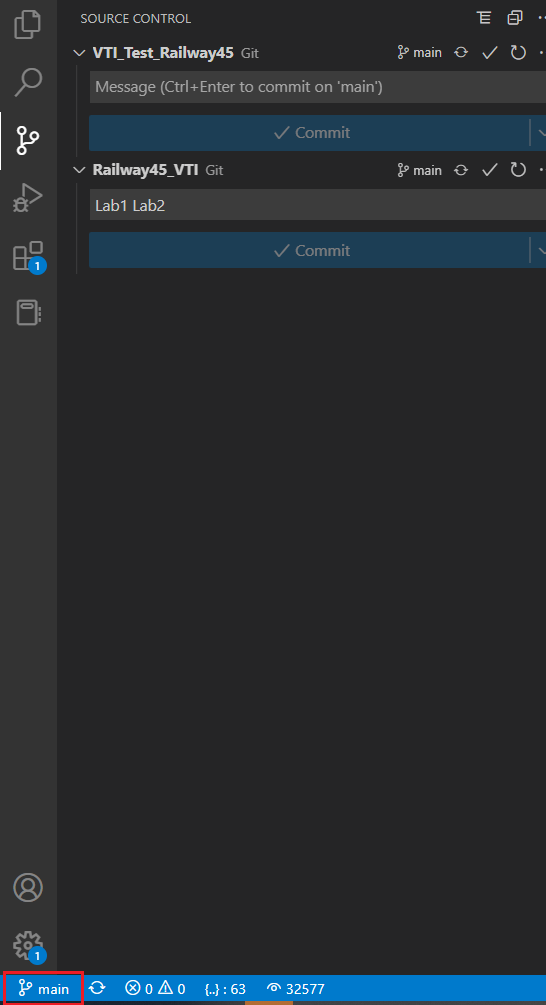


* Chọn: Để lưu các thay đổi vào Local Repository, chú ý thêm nội dung Comment trước khi Commit

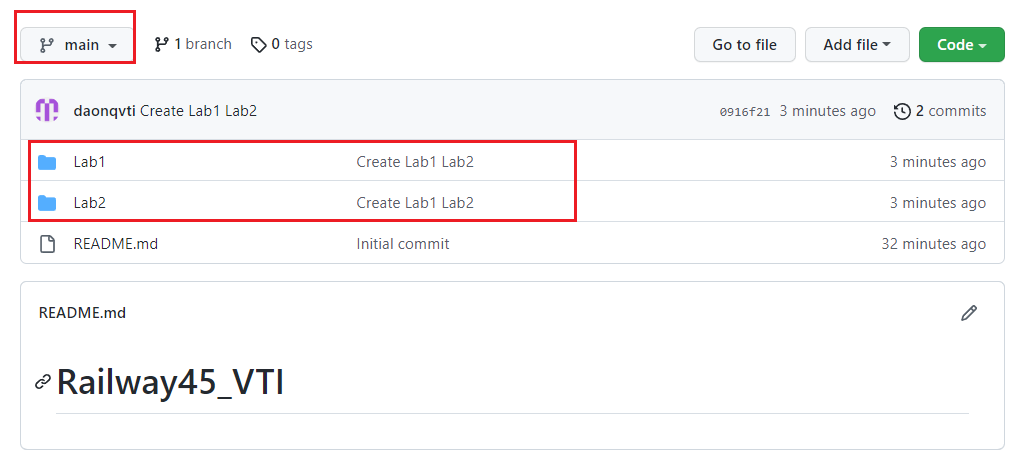


* Chọn: Để Push Code lên GitHub, chú ý thêm nội dung Comment.



****

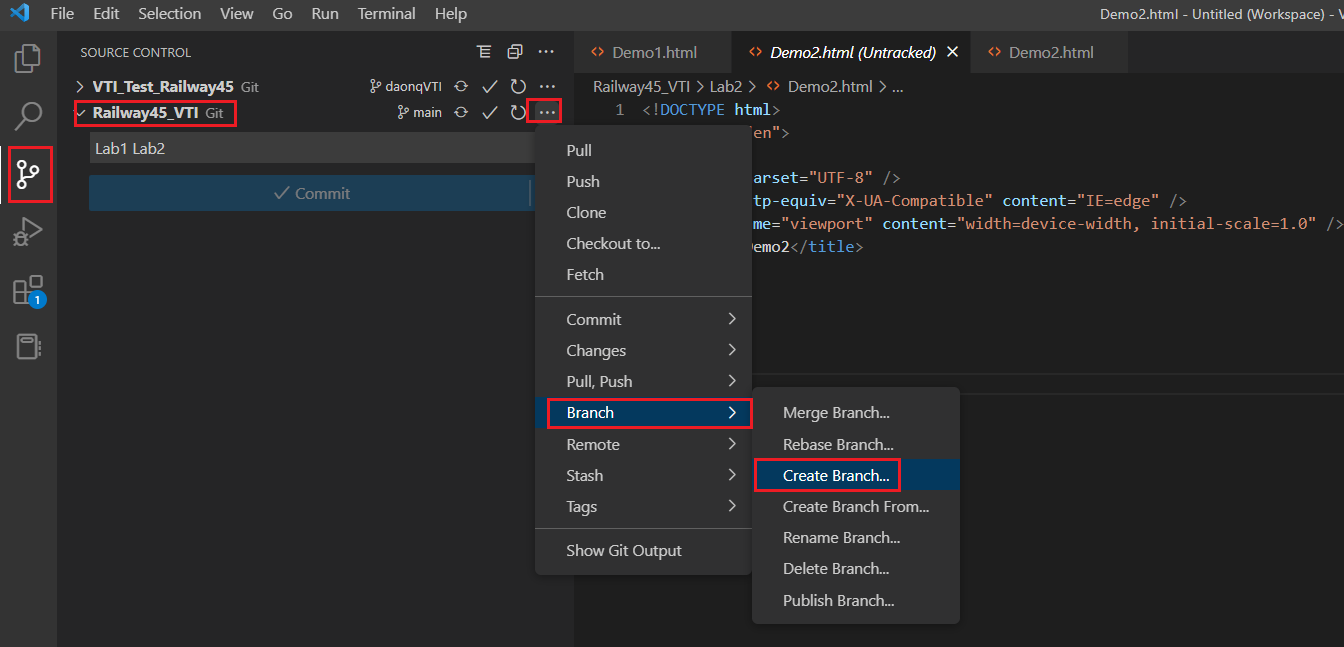
* Kiểm tra lại trên GitHub, Code đã được Push lên nhánh Main



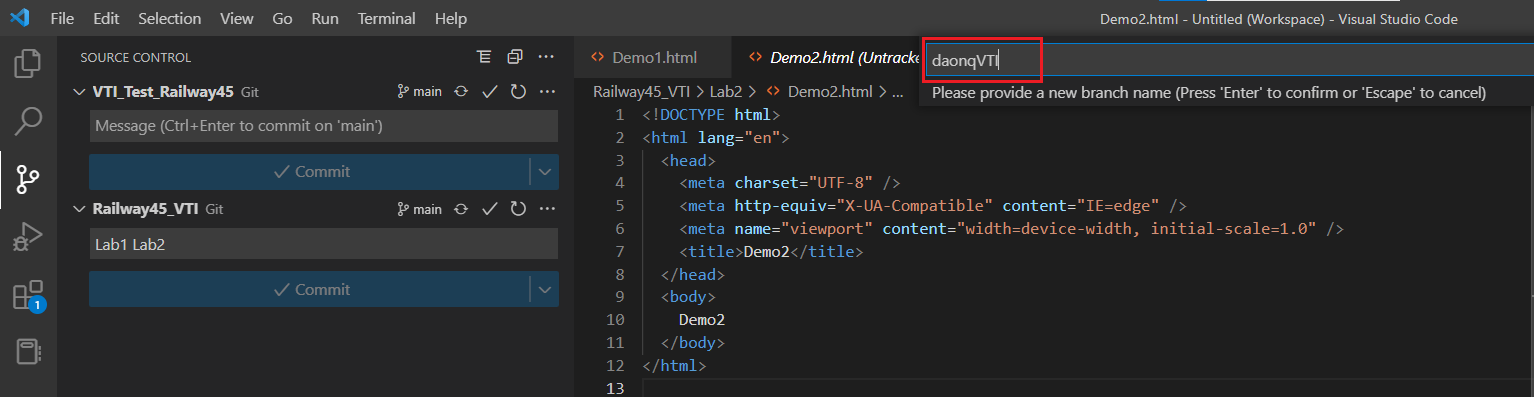


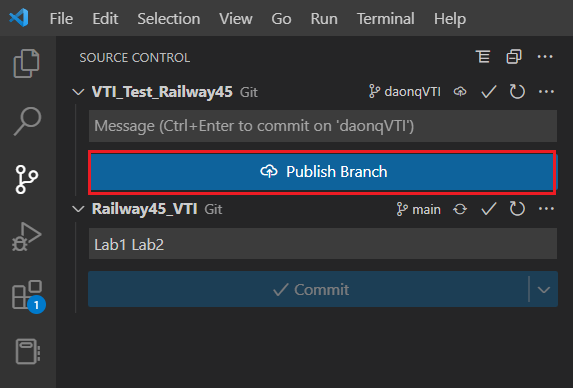
1. **Tạo các Branch để làm việc nhóm:**

* Thực hiện chọn như hình.

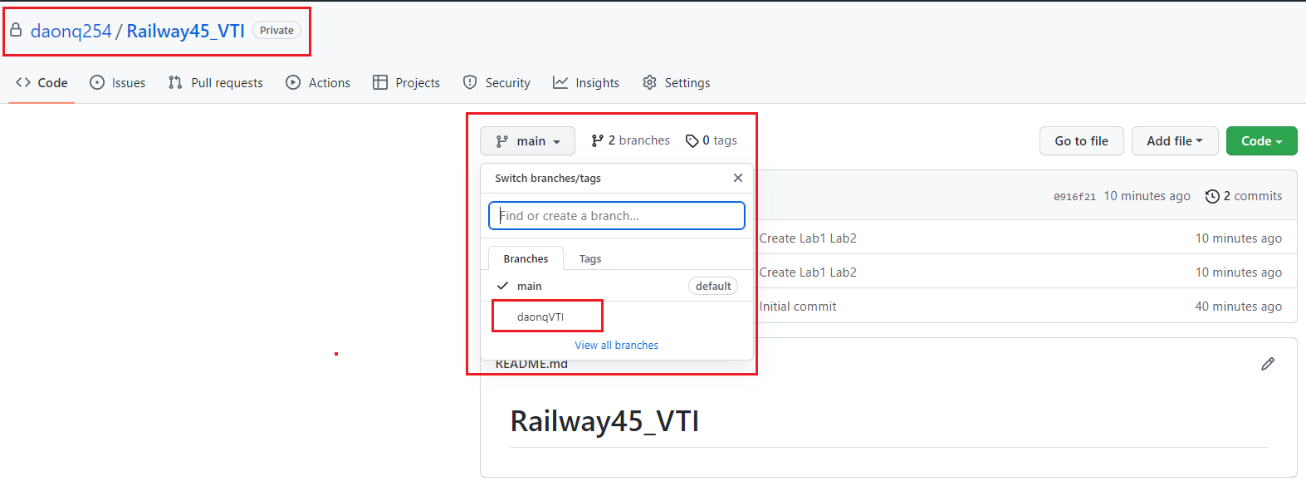


* Tạo 1 nhánh mới với tên: **daonqVTI**

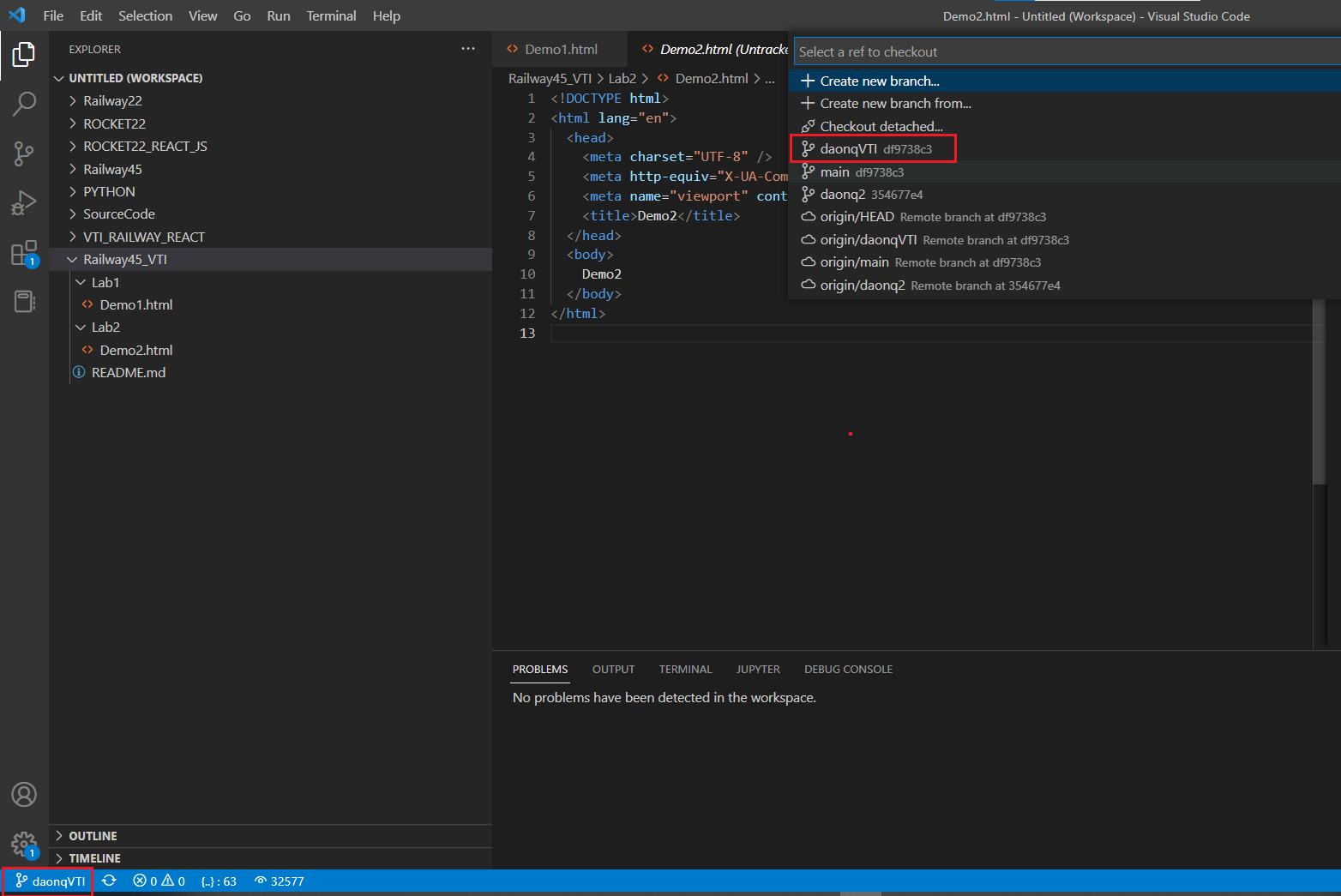




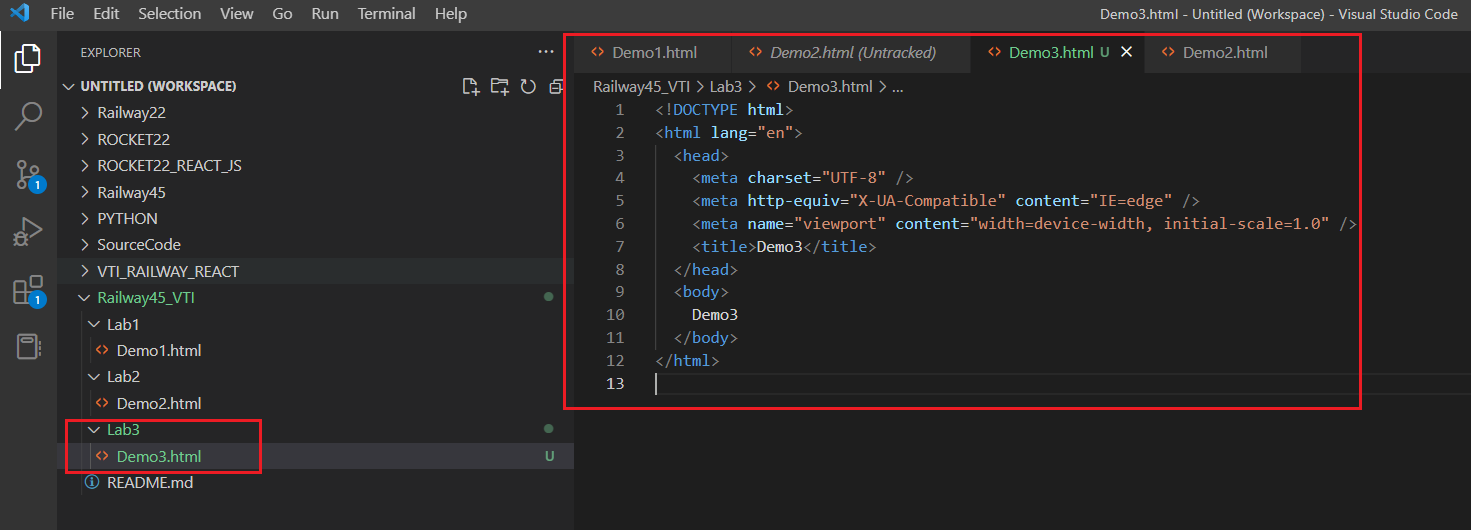
* Kiểm tra lại nhánh vửa tạo trên GitHub:



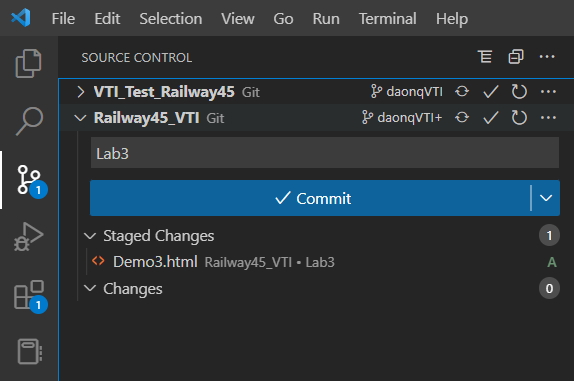
* Tại VS Code để chuyển sang nhánh vừa tạo chọn:

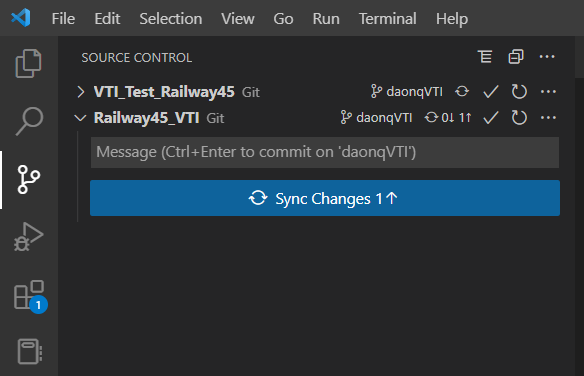


* Bây giờ khi thực hiện các lệnh Git sẽ được push lên Branch này.
* Thực hiện tạo thêm 1 bài Lab3 để Test.

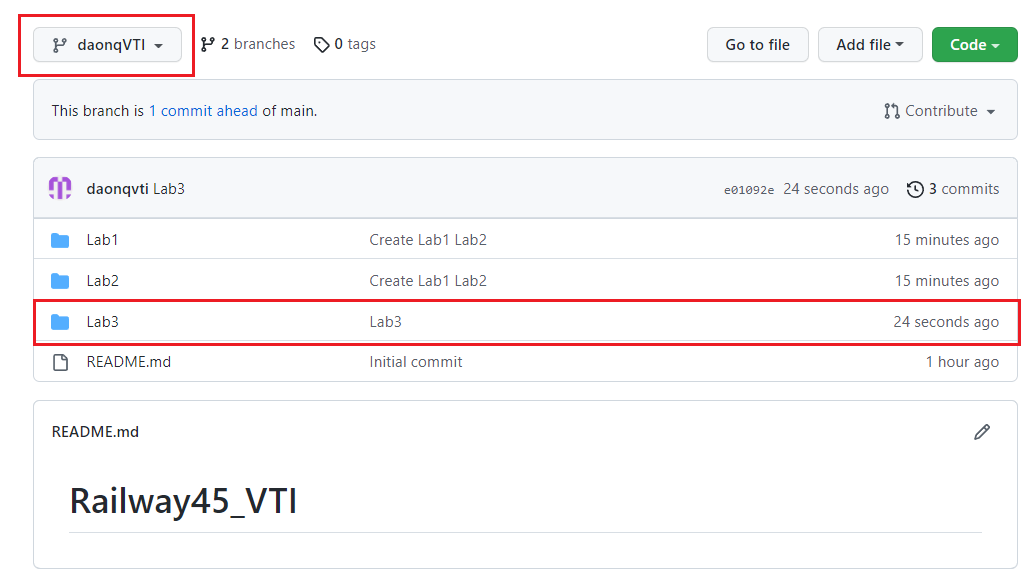


* Thực hiện các thao tác như bên trên để Push Code lên GitHub.



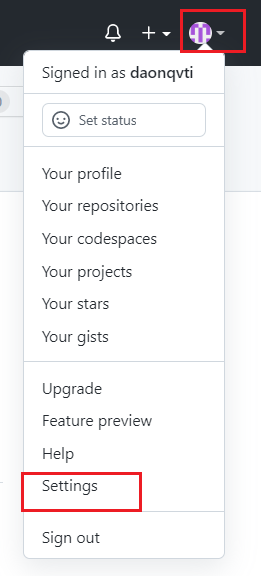


* Kiểm tra lại nhánh daonqVTI trên GitLab, Code được Push lên Branch này.

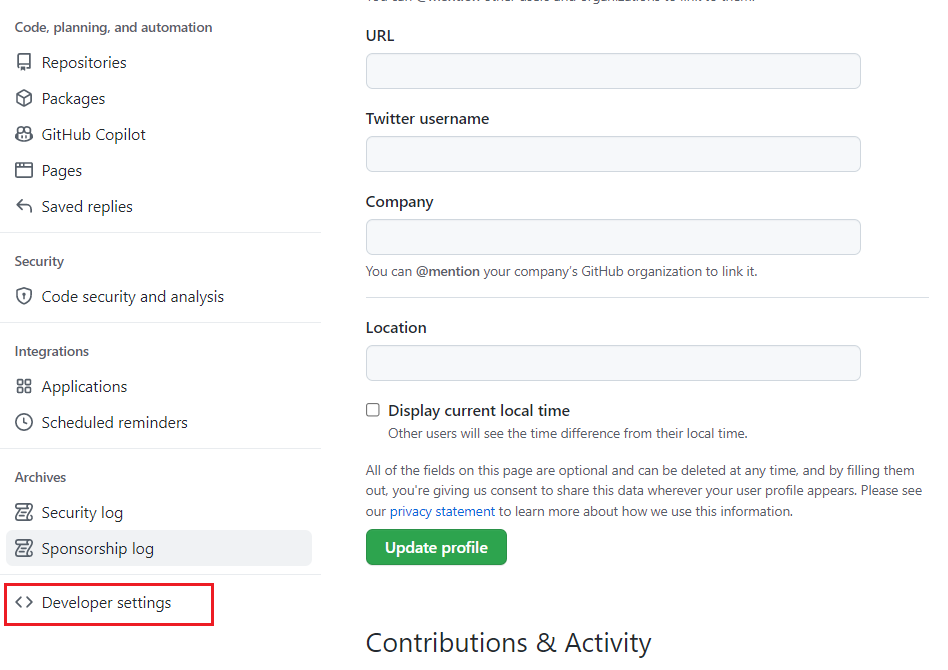


1. **Cách lấy Token GitHub:**

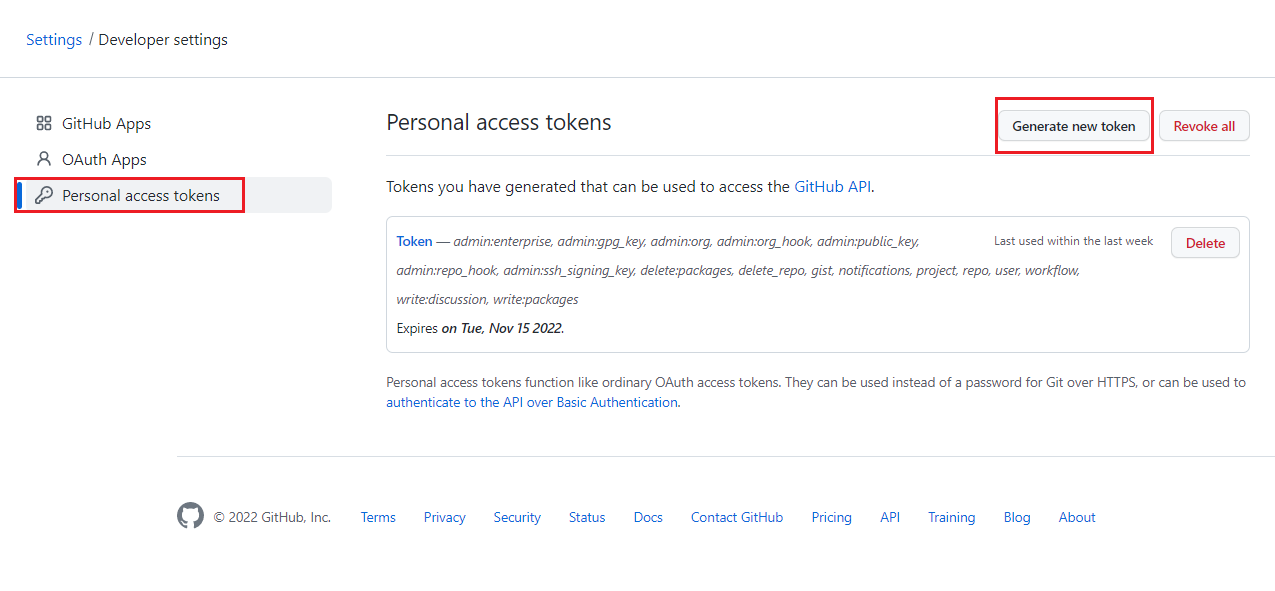
* Giao diện GitHub:



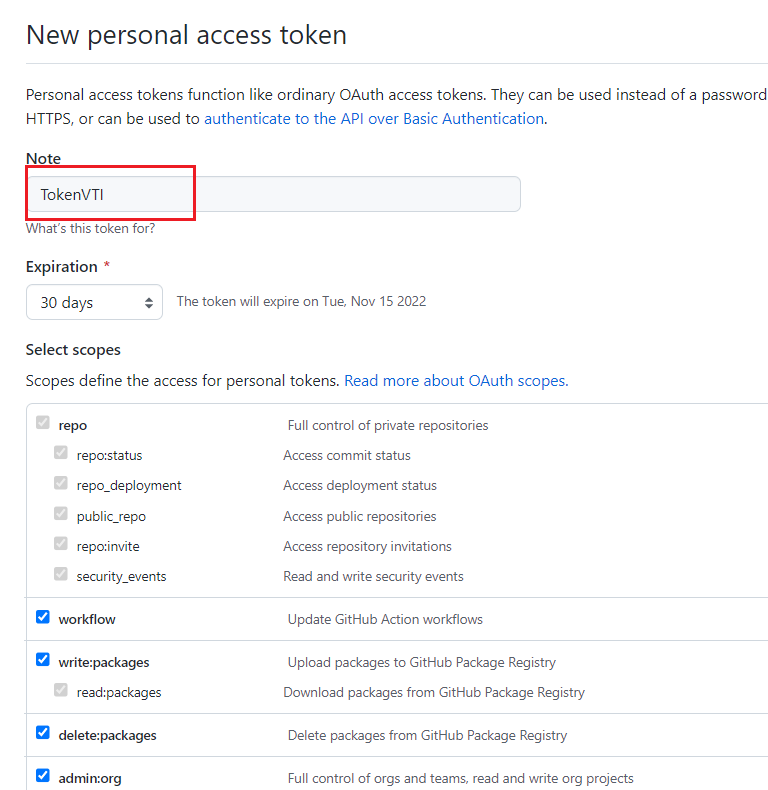
* Chọn:



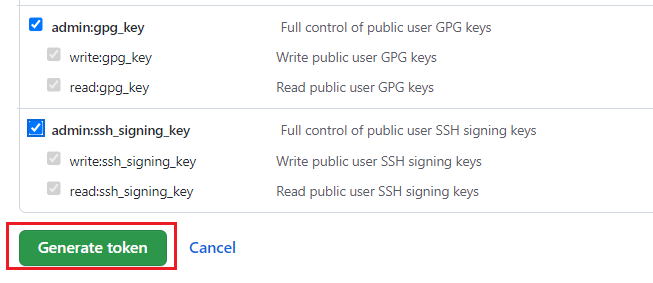
* Chọn:



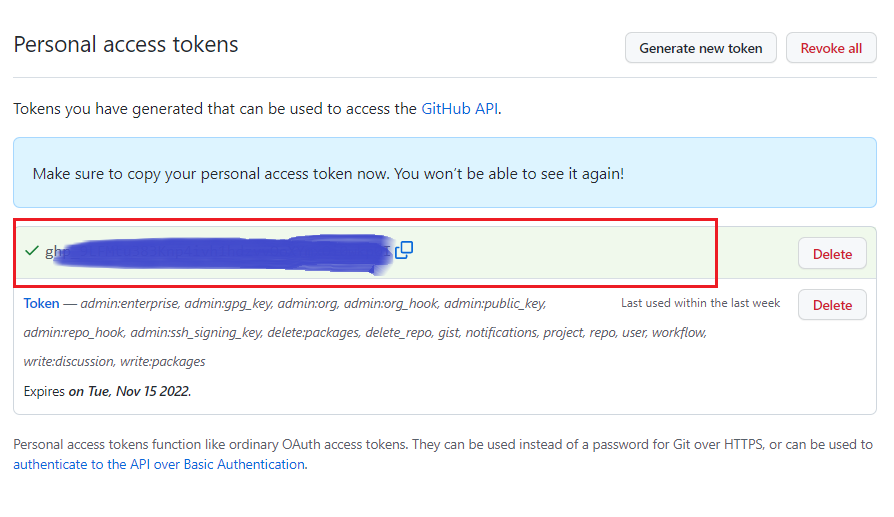
* Điền tên Token và chọn Full Role:



* Chọn:

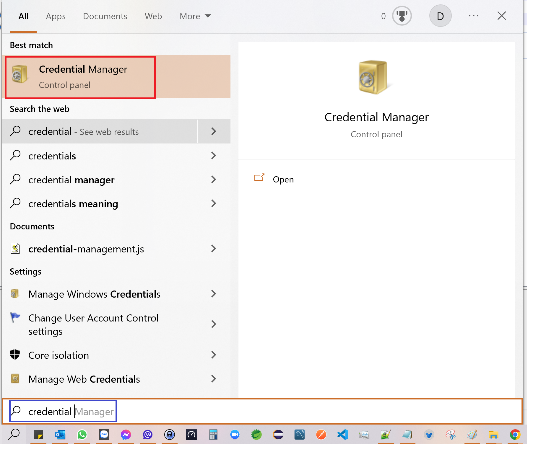


* Lưu lại mã Token để sử dụng:

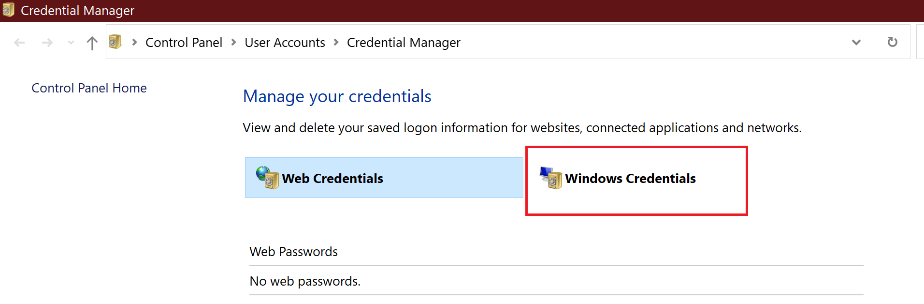


1. **Thay đổi tài khoản Local Github:**

* Chuyển User Github ở Local:
* Thực hiện tìm Credential ở window



* Chọn:



* Tìm tài khoản Github đang lưu trữ trên máy và xóa:

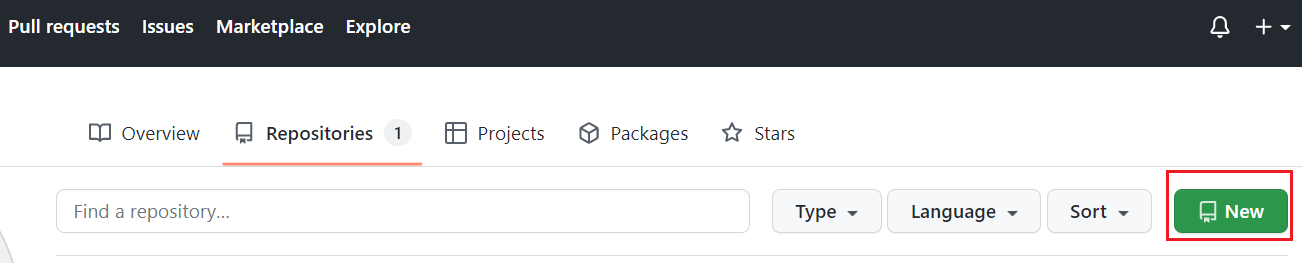


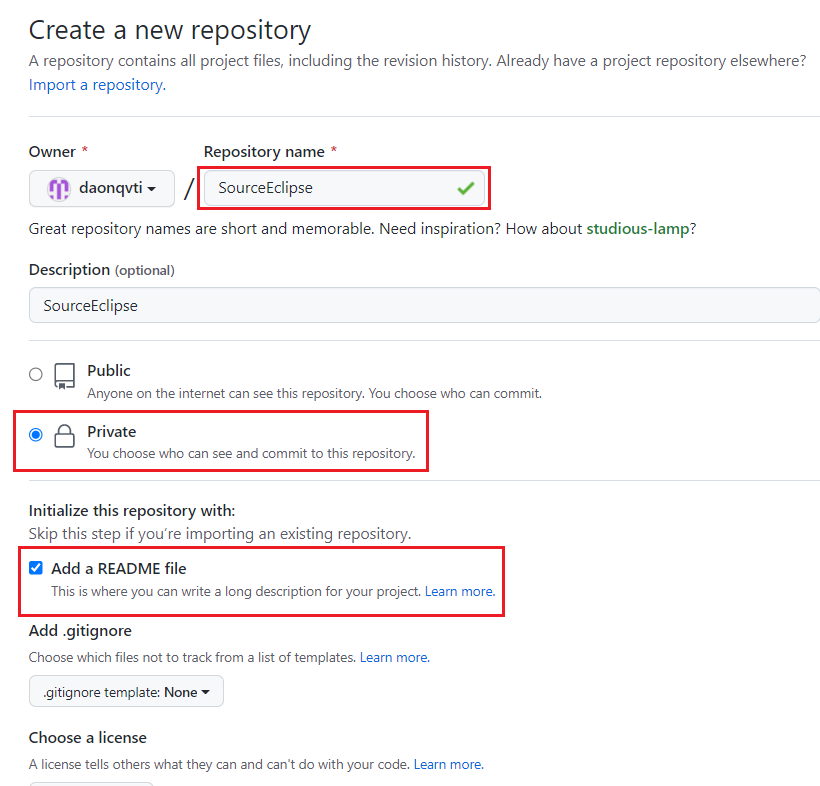
* Sử dụng các lệnh dưới để cài đặt lại user Github cần sử dụng:

|  |
| --- |
| git config --global user.name "daonq254"  git config --global user.email "[dao.nq254@gmail.com](mailto:dao.nq254@gmail.com)" |

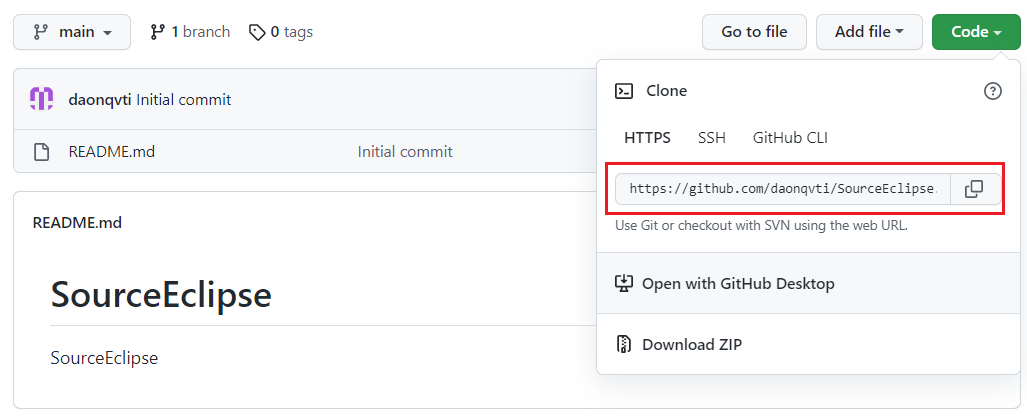
1. **Sử dụng Git với Eclipse:**

* **Bước 1: Tạo Repository trên GitHub để lưu Source Code.**

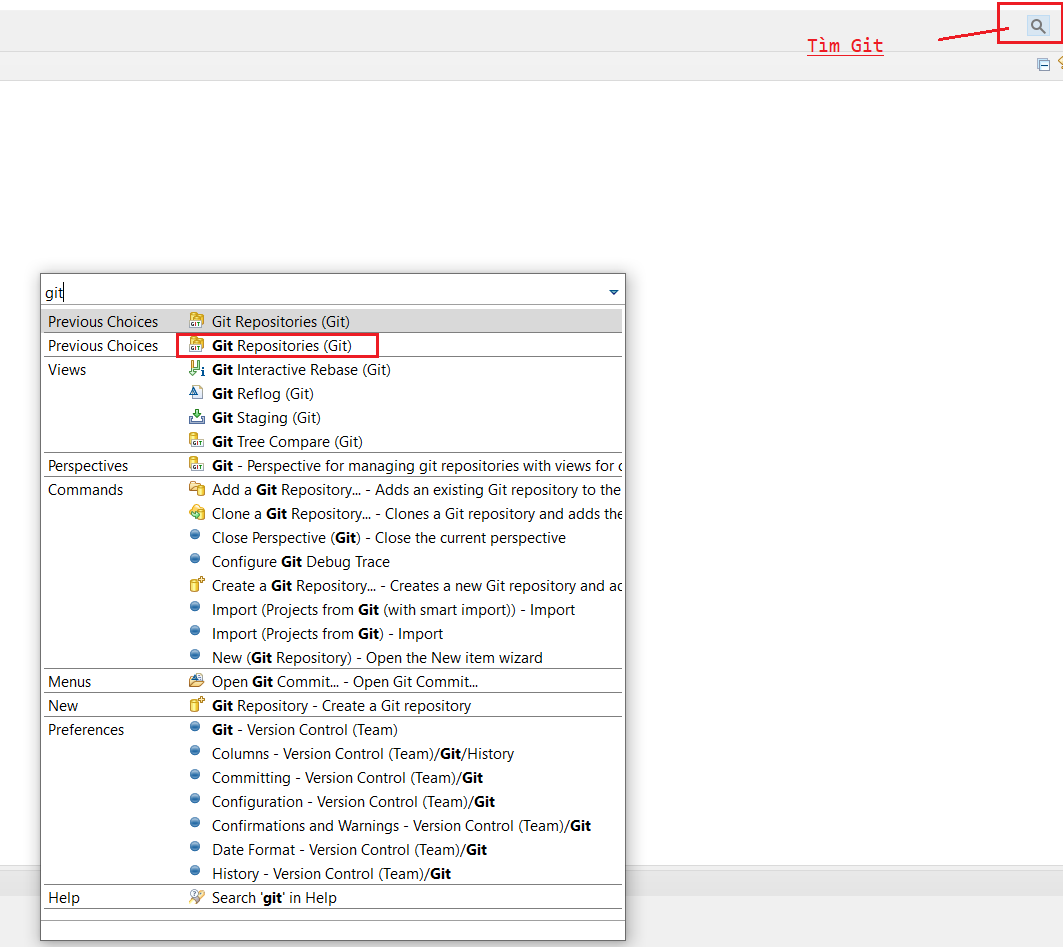




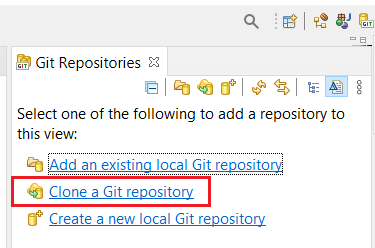
* Lưu đường dẫn của Repository



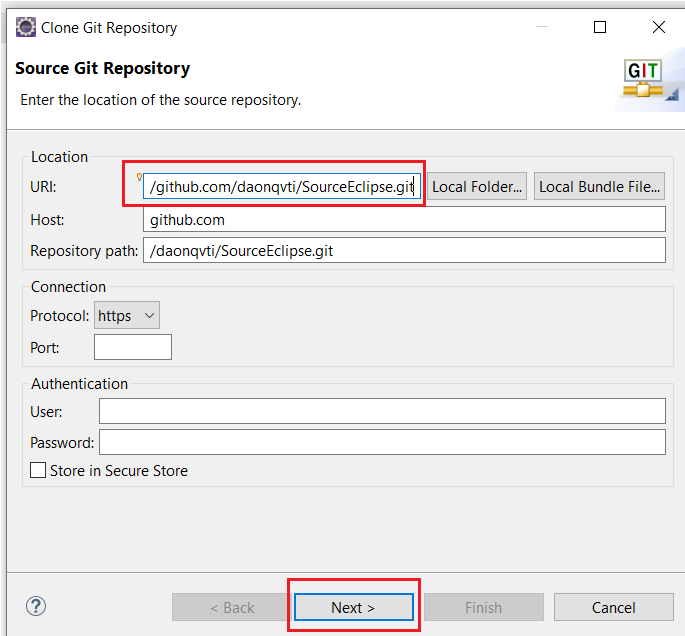
* **Bước 2: Kết nối Eclipse tới Repository GitHub vừa tạo bước trên.**
* Thực hiện mở Eclipse, tìm Git



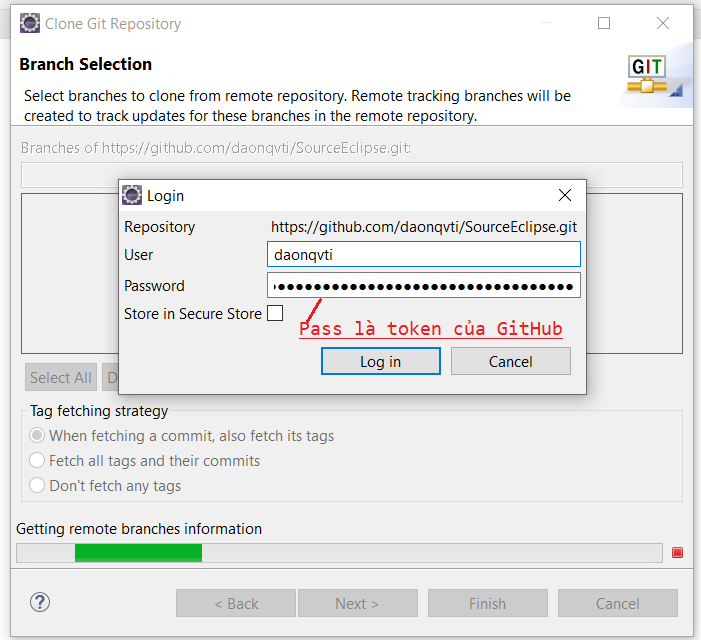
* Chọn:



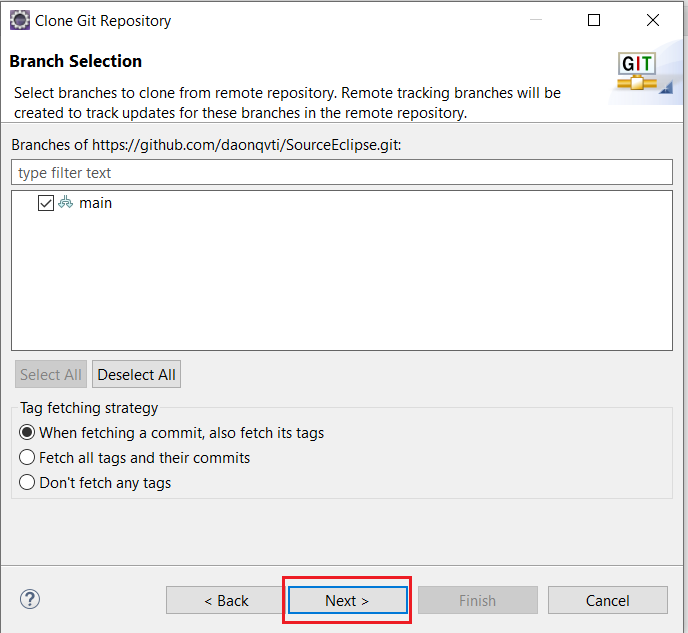
* Điền đường dẫn của Repository trên Git Hub để kết nối:



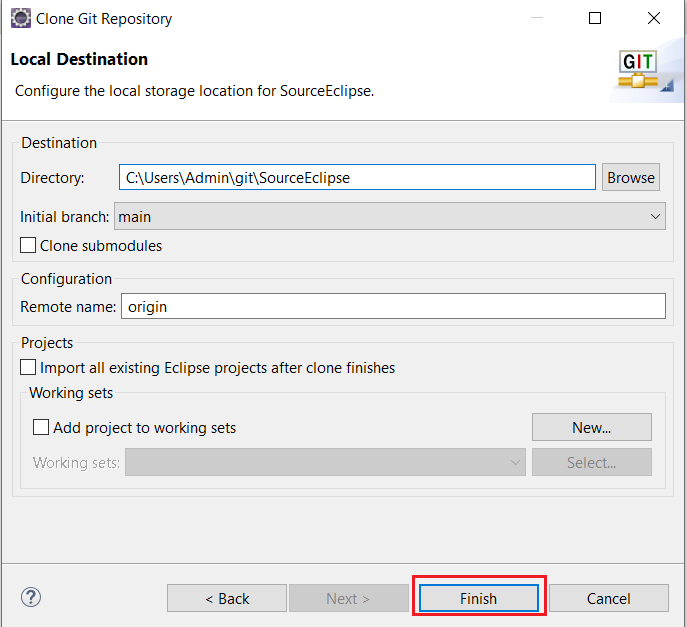
* Điền thông tin xác thực:



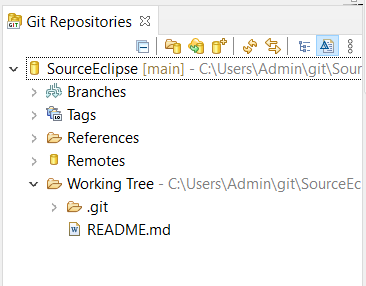
* Chọn:



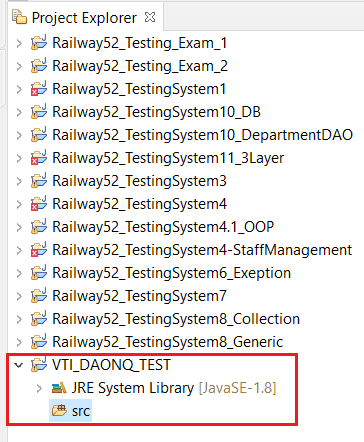
* Chọn:



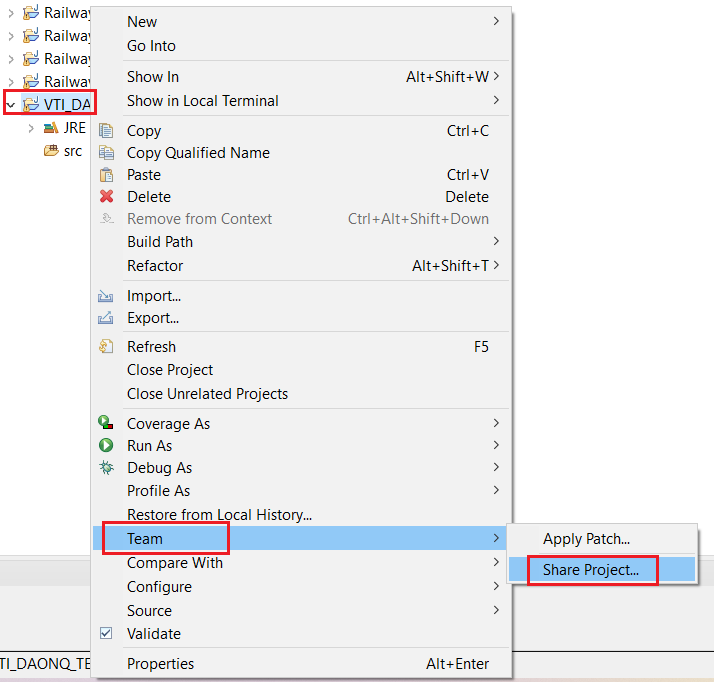
* Kết quả:



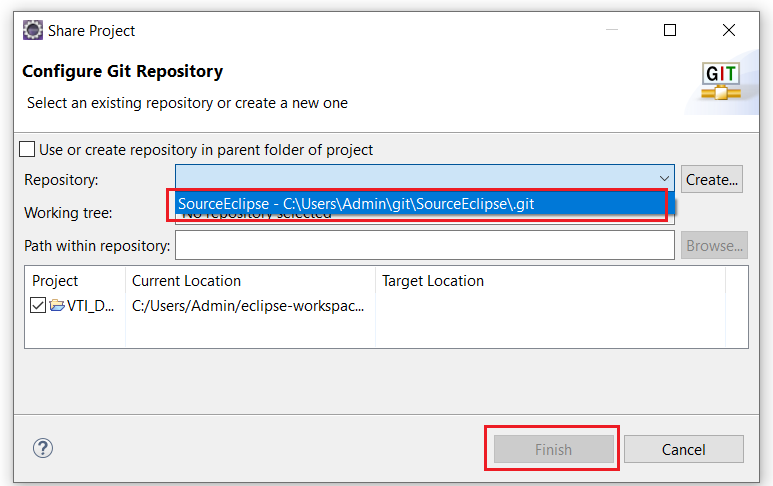
* **Bước 3: Tạo Project Eclipse, Link tới Git để quản lý**
* Thực hiện tạo 1 Project Java:

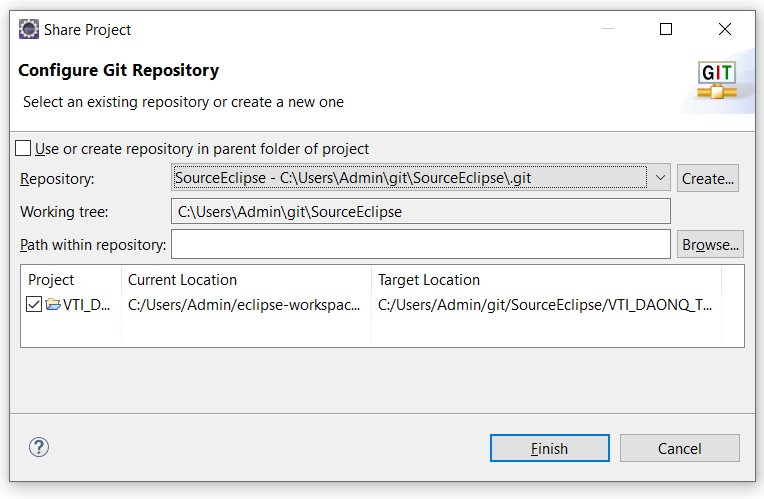


* Chọn:



* Chọn: Để link Project vừa tạo tới Git

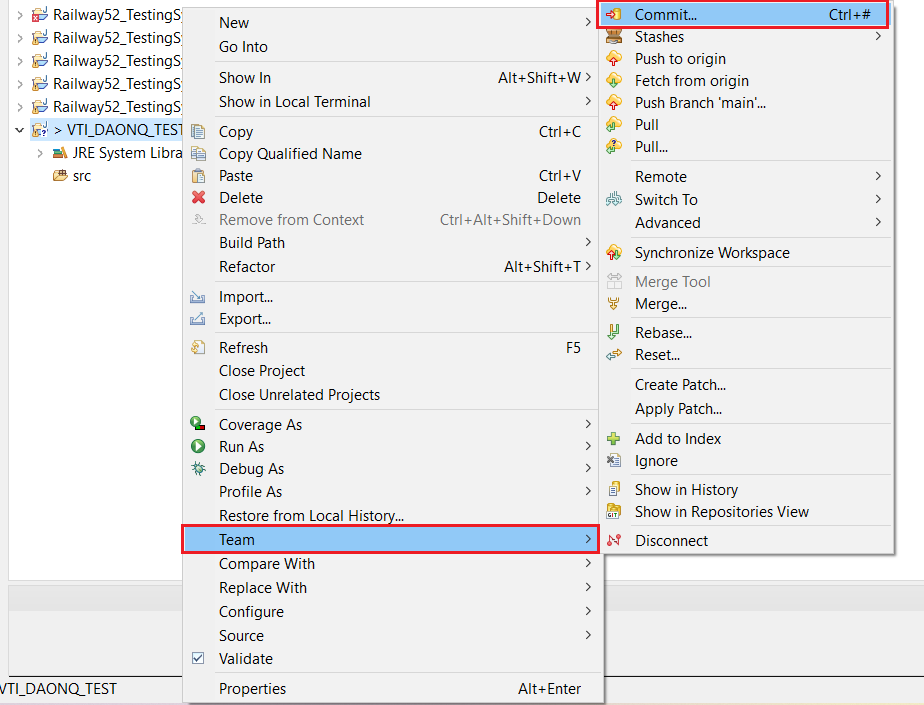


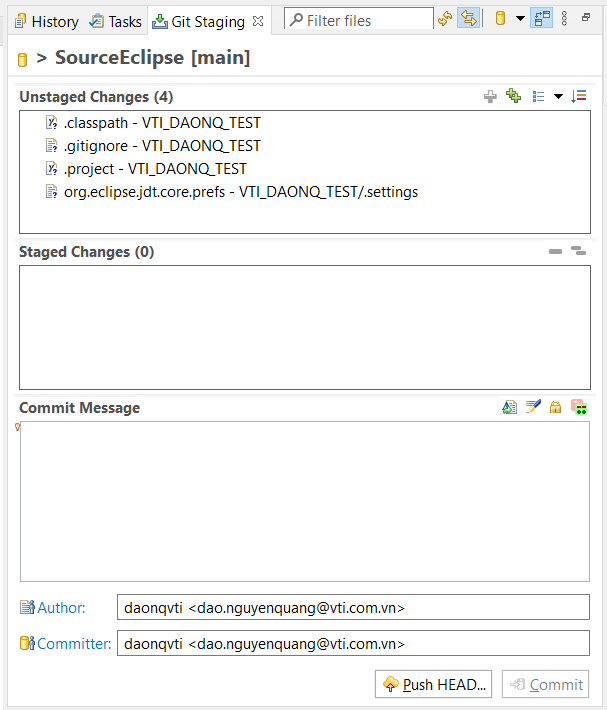


* Kết quả:

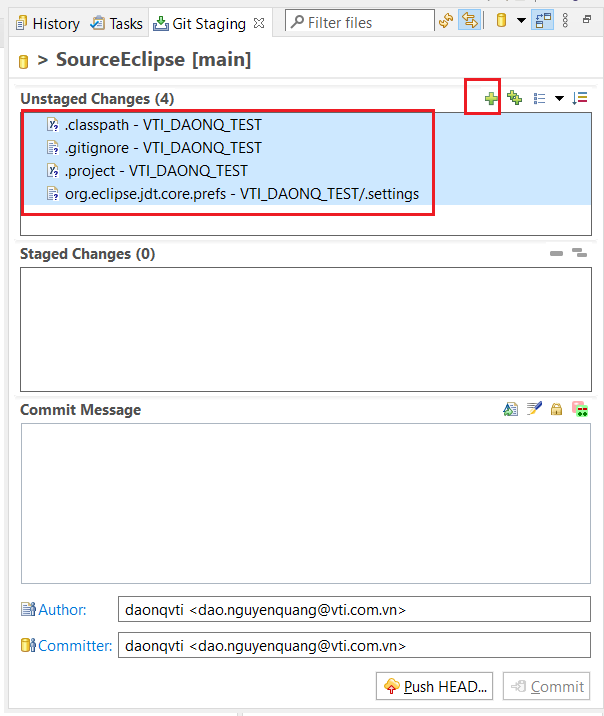


* Commit Project vừa tạo tới GitHub

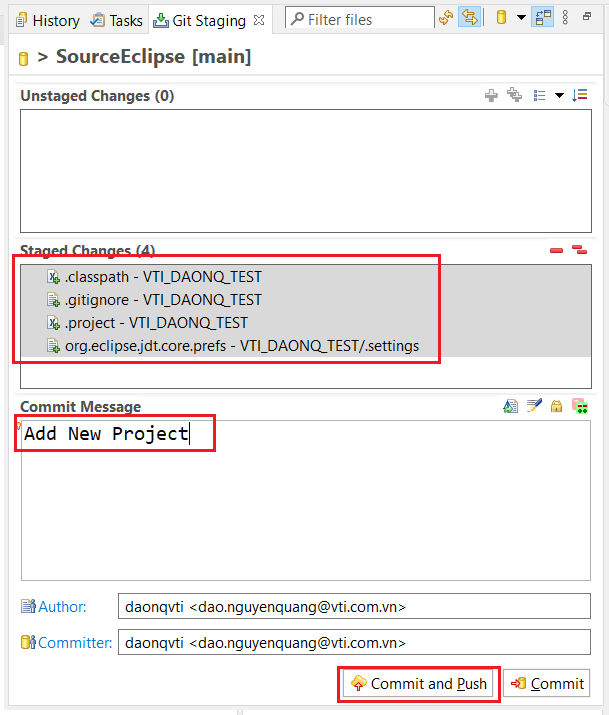




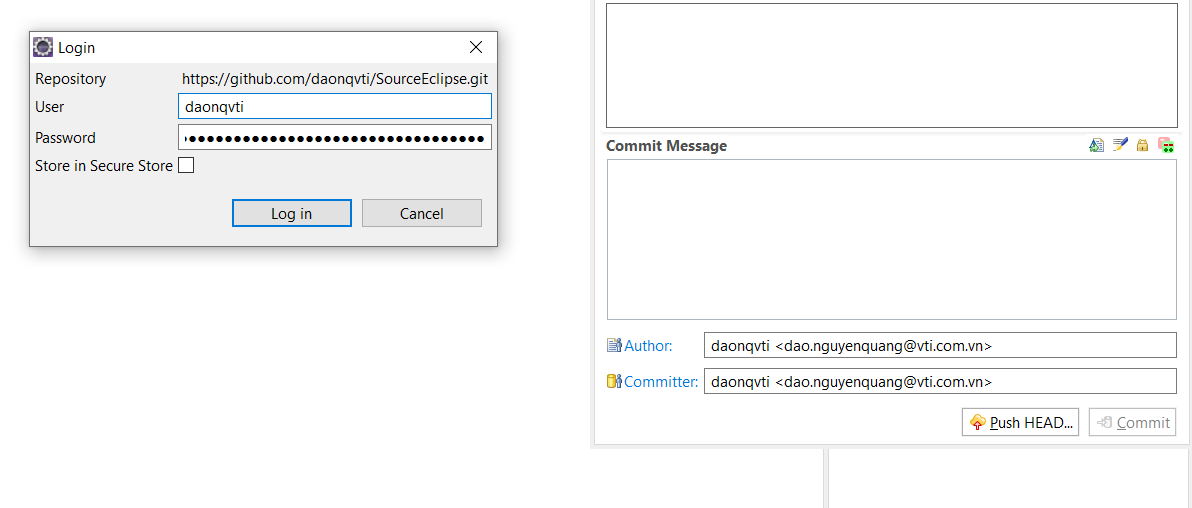
* Add các file tới Stage:



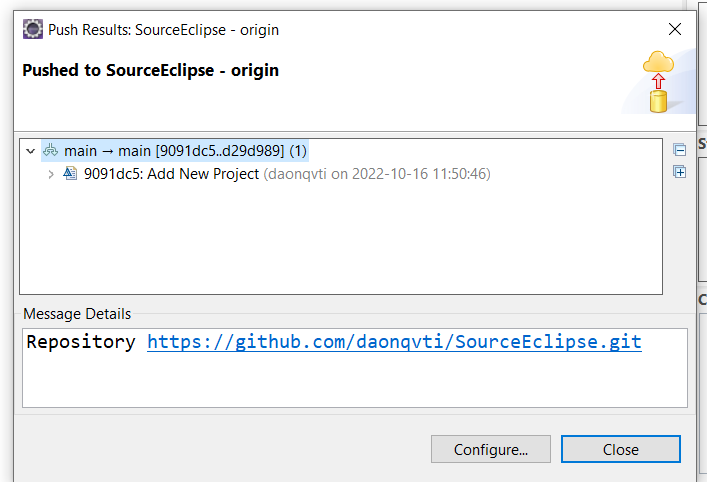
* Chọn Commit để lưu vào LocalRepository hoặc Commit And Push để lưu vào LocalRepository và Push tới GitHub

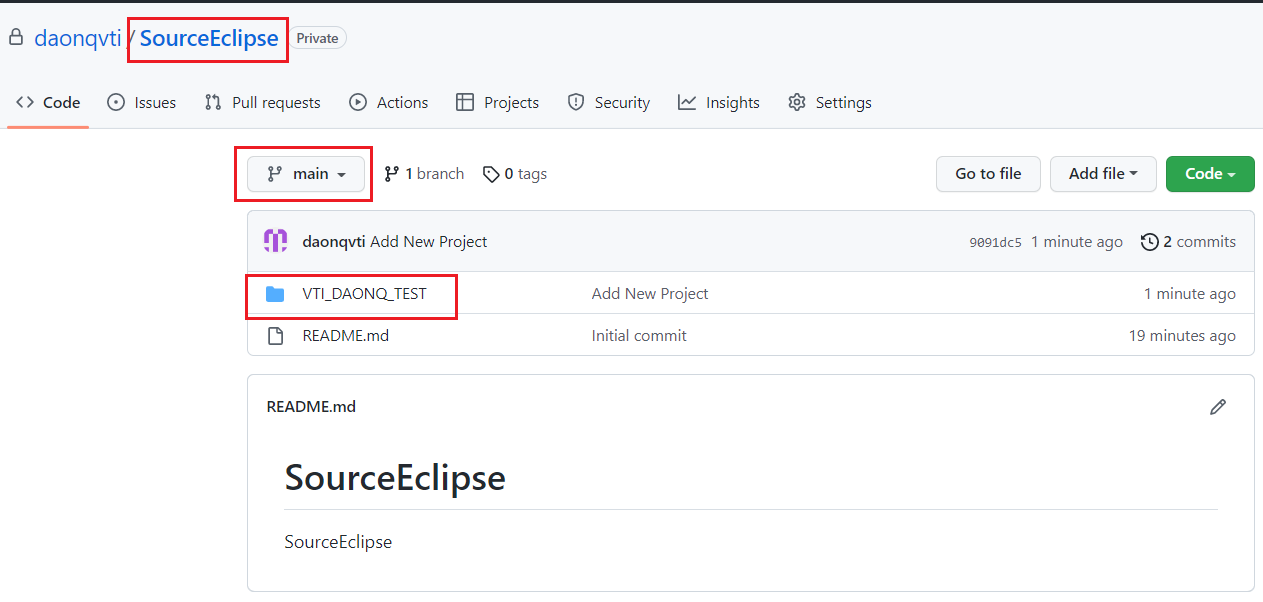


* Điền thông tin Xác thực: Password điền token



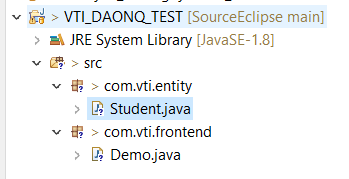
* Kết quả:







* **Bước 4: Thực hiện tạo thêm các file SourceCode khác và đưa lên GitHub**
* Thực hiện tạo các package: com.vti.entity, com.vti.frontend



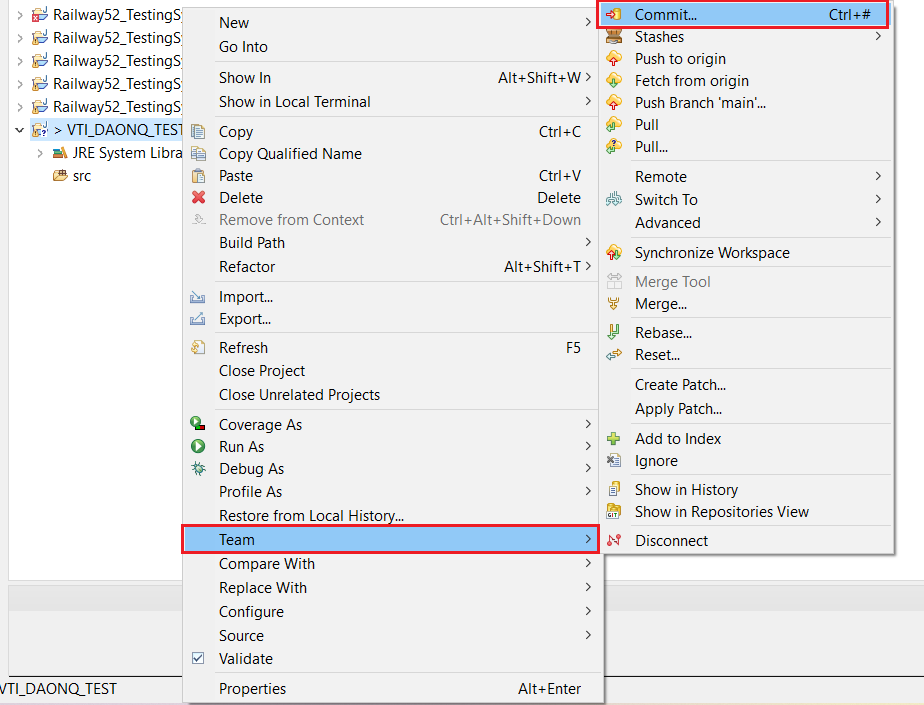
* Tạo các file Code: Student, Demo
* File Student:

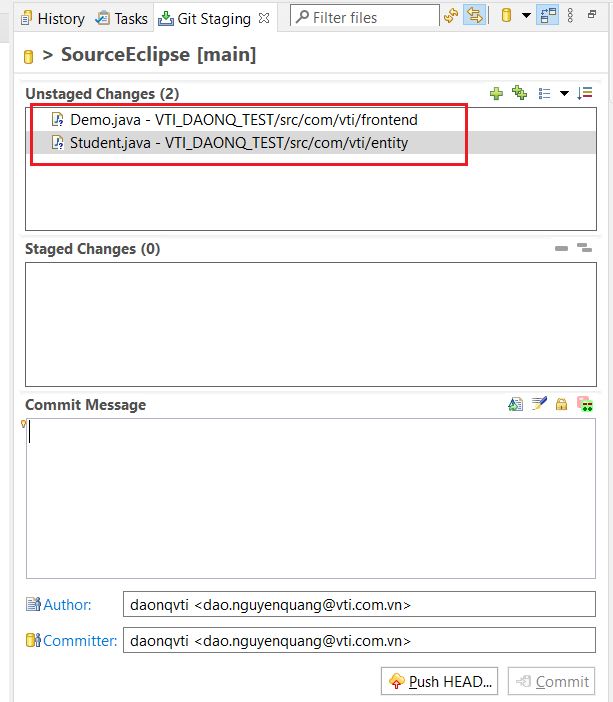
|  |
| --- |
| **package** com.vti.entity;  **public** **class** Student {  **private** **int** id;  **private** String name;  **public** **int** getId() {  **return** id;  }  **public** **void** setId(**int** id) {  **this**.id = id;  }  **public** String getName() {  **return** name;  }  **public** **void** setName(String name) {  **this**.name = name;  }  @Override  **public** String toString() {  **return** "Student [id=" + id + ", nameString=" + name + "]";  }  } |

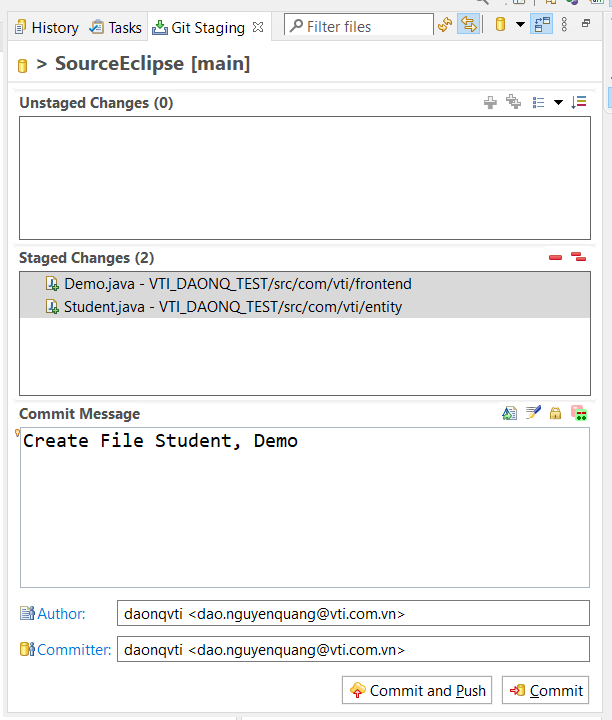
* File Demo:

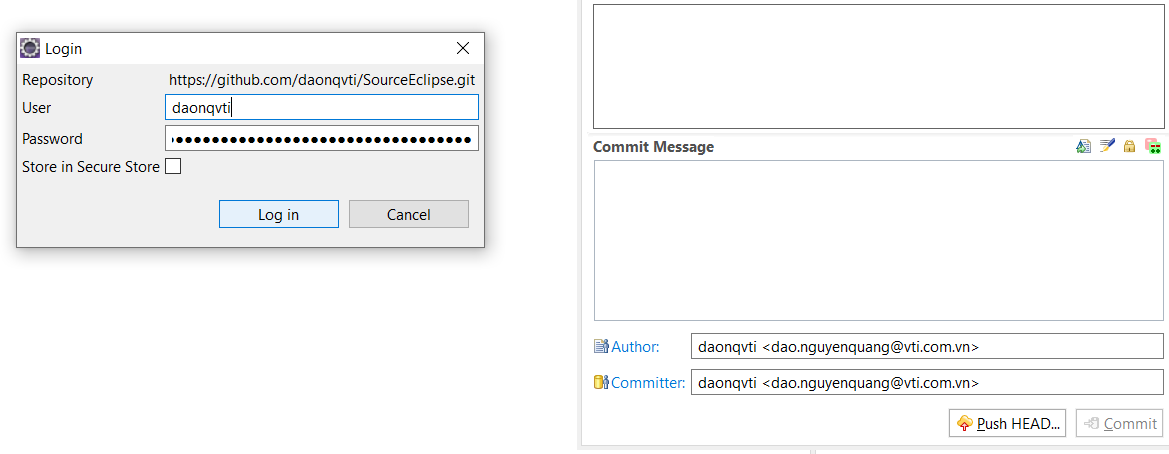
|  |
| --- |
| **package** com.vti.frontend;  **import** com.vti.entity.Student;  **public** **class** Demo {  **public** **static** **void** main(String[] args) {  Student student1 = **new** Student();  student1.setId(1);  student1.setName("DaoNQ");  System.***out***.println("Thông tin Sinh viên: " + student1.toString());  }  } |

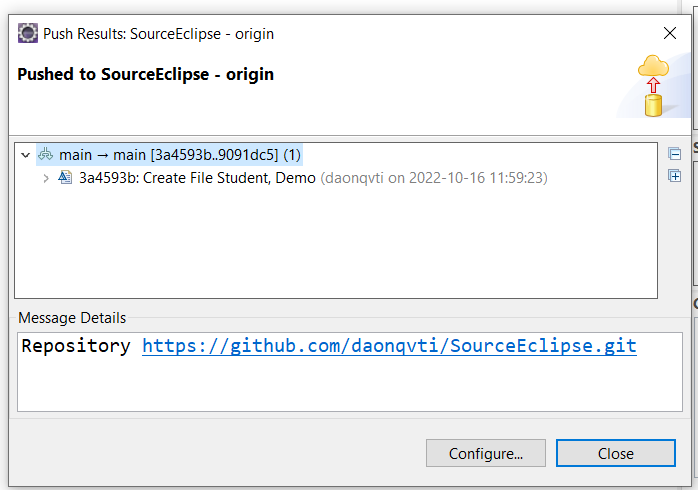
* Thực hiện Commit và Push các file Source lên GitHub:



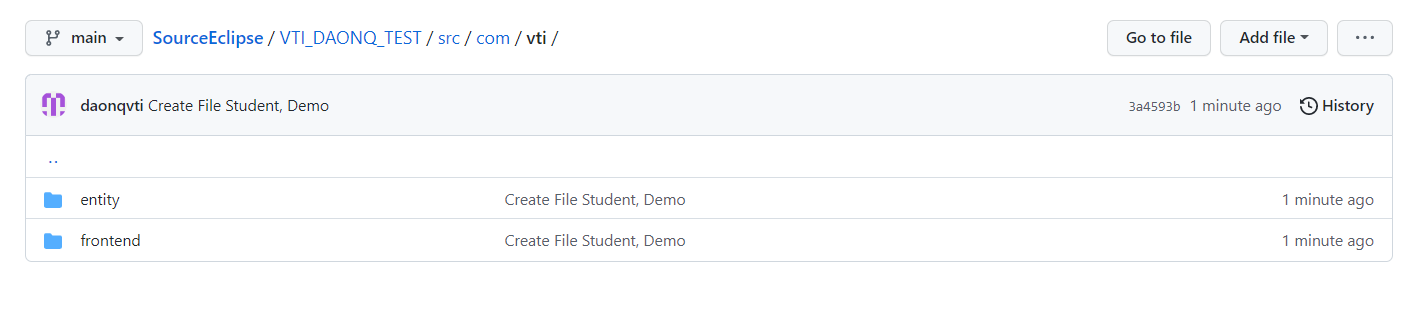


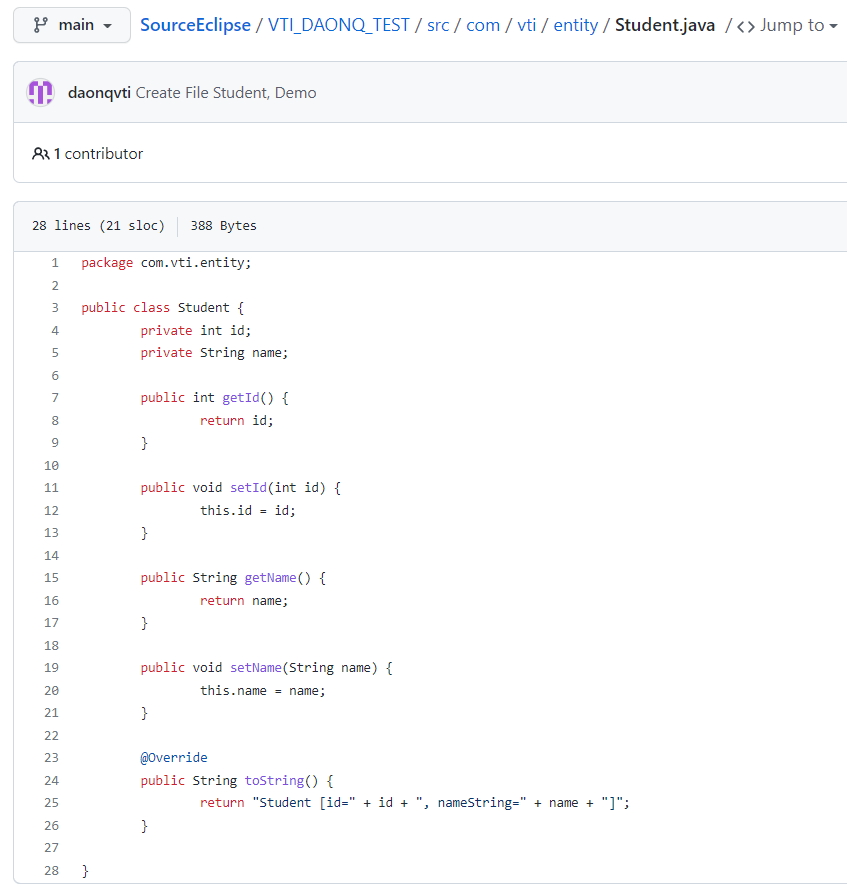




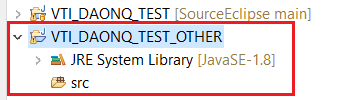


* Kết quả:

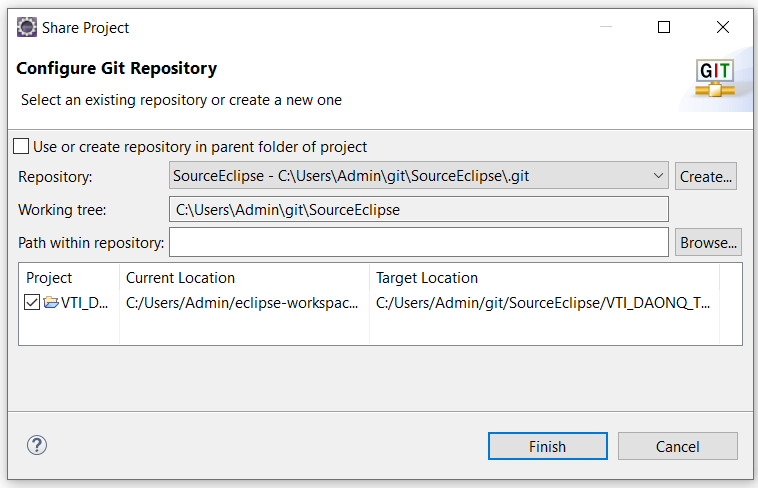


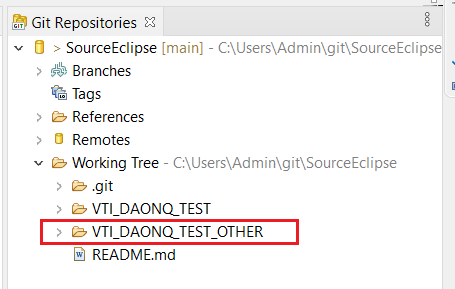


* **Bước 5: Tạo thêm 1 Project khác, quản lý cùng Repository này ở GitHub**
* Tạo Project: **VTI\_DAONQ\_TEST\_OTHER**

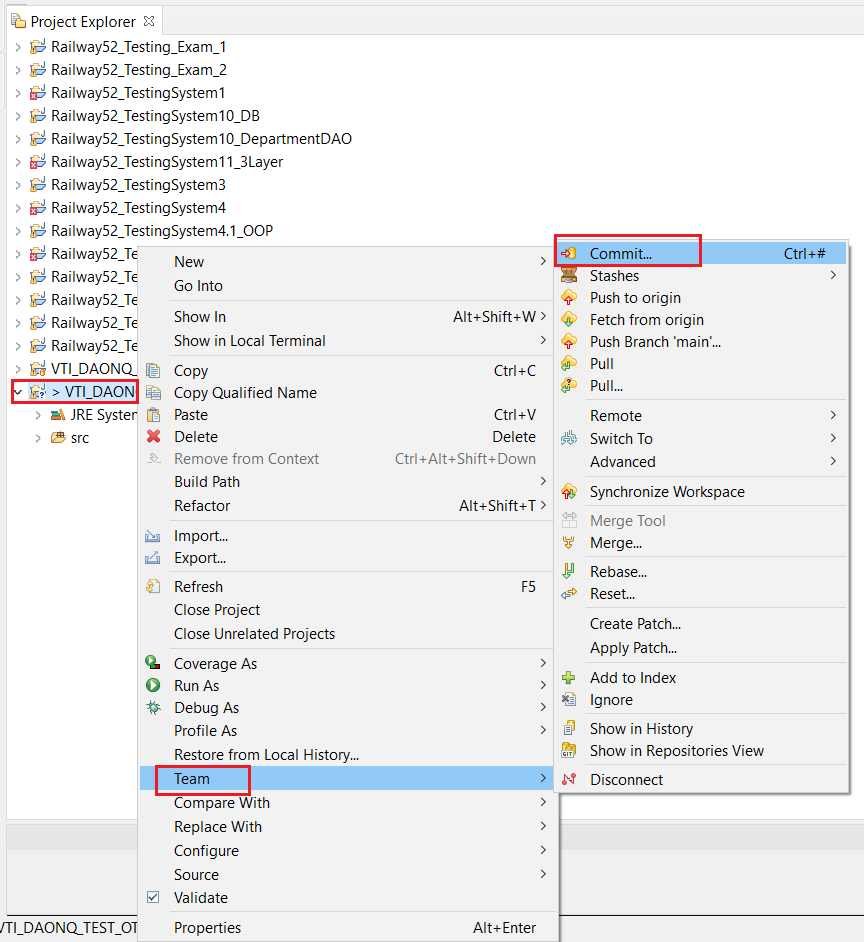


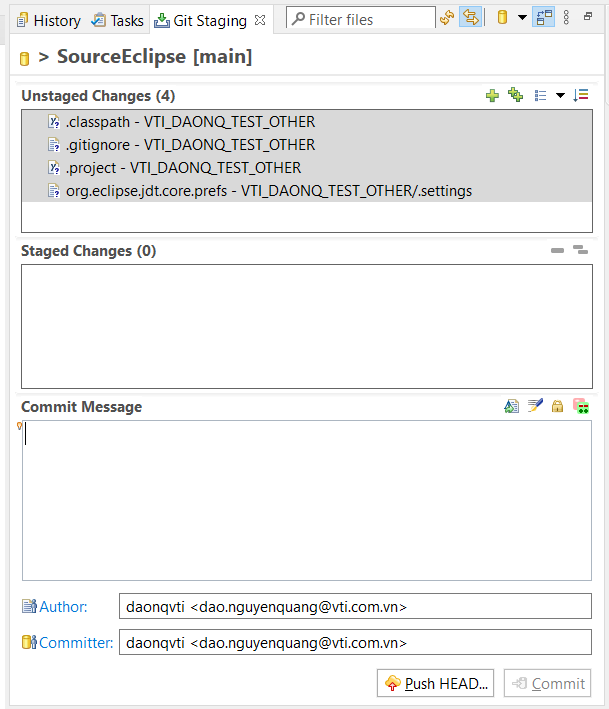
* Thực hiện Link Project vừa tạo tới Repository GitHub bên trên:

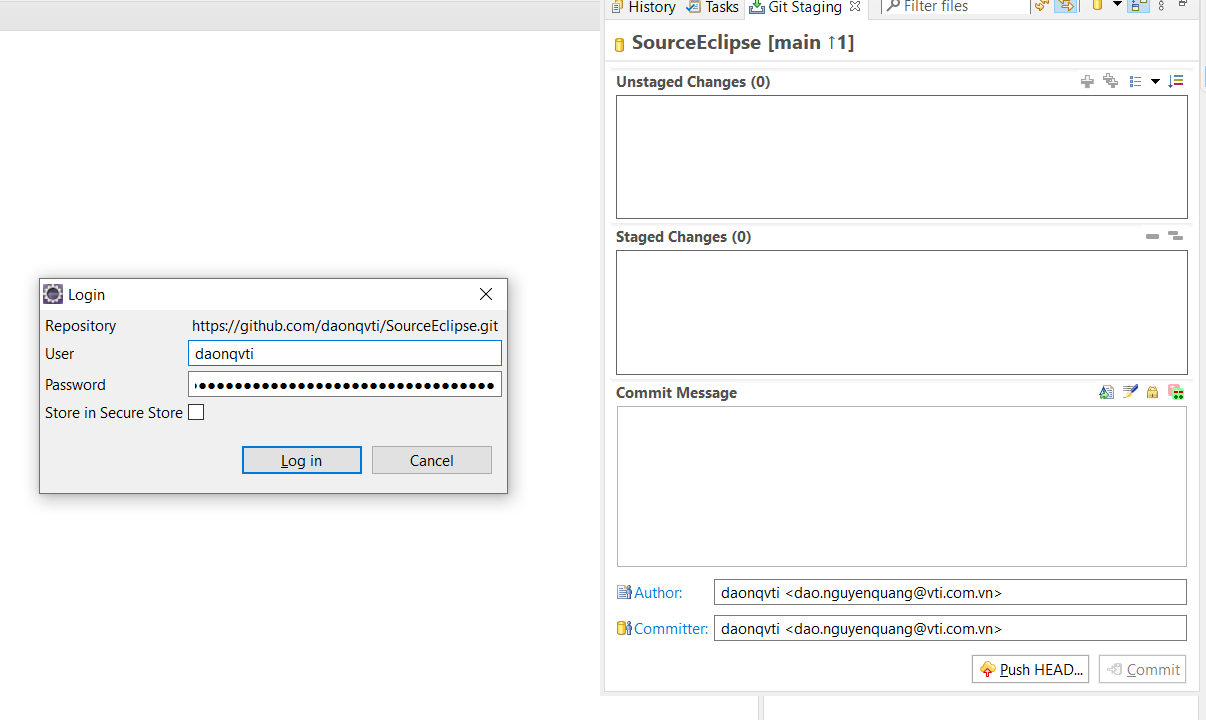


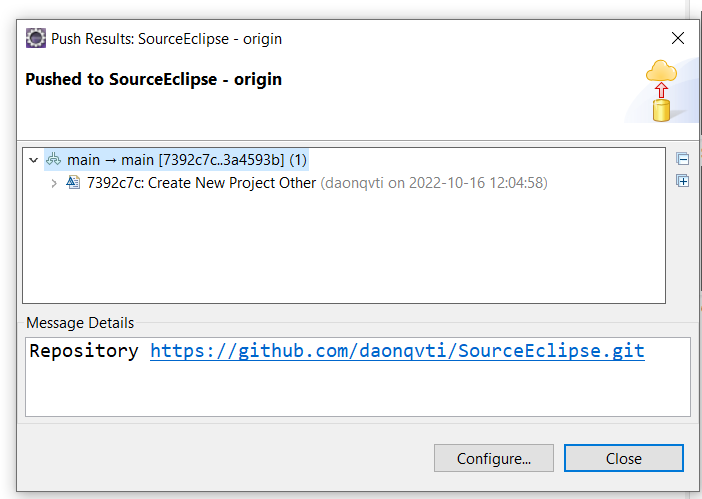


* Thực hiện Commit

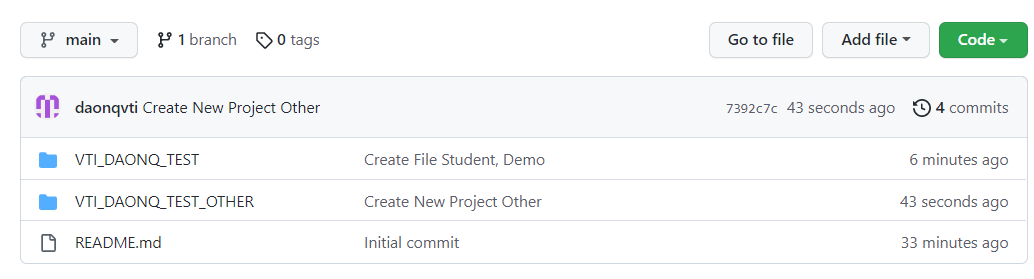


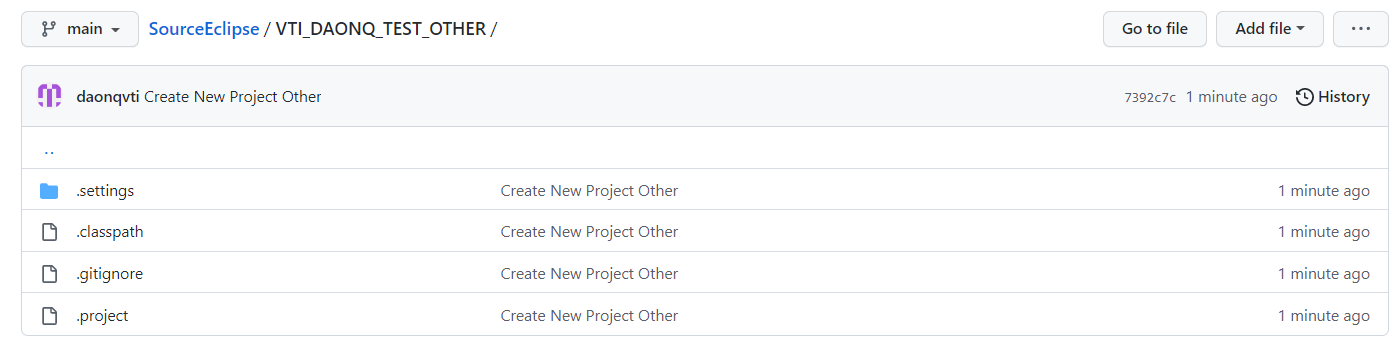




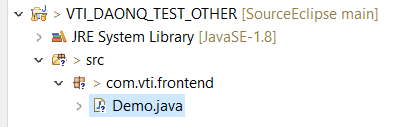


* Kết quả:



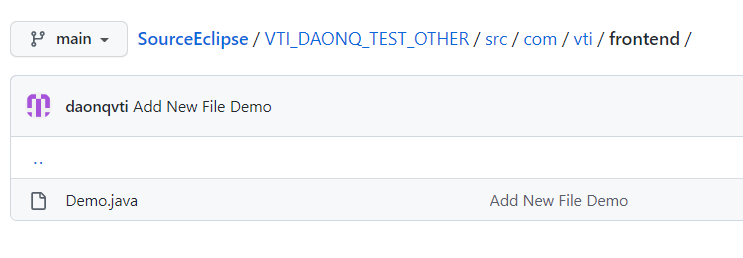


* Tạo thêm các file Code trong Project này và Commit And Push để kiểm tra kết quả:

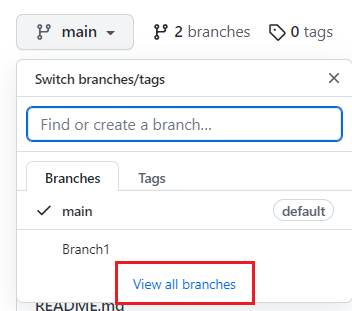


Thực hiện các bước để Commit và Push Code:

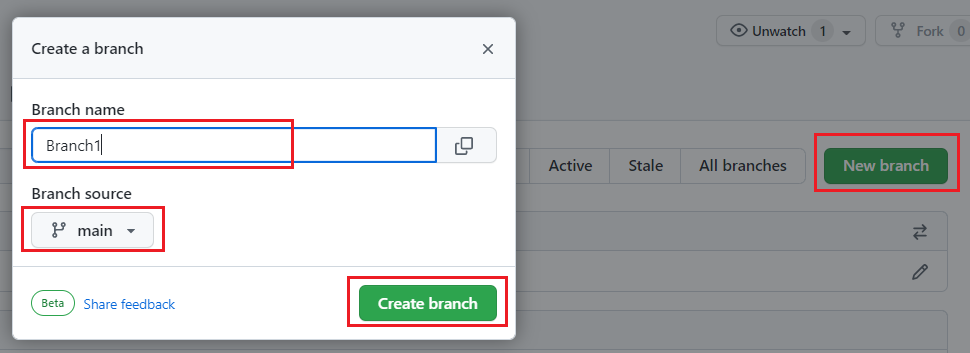
Kết quả trên GitHub:



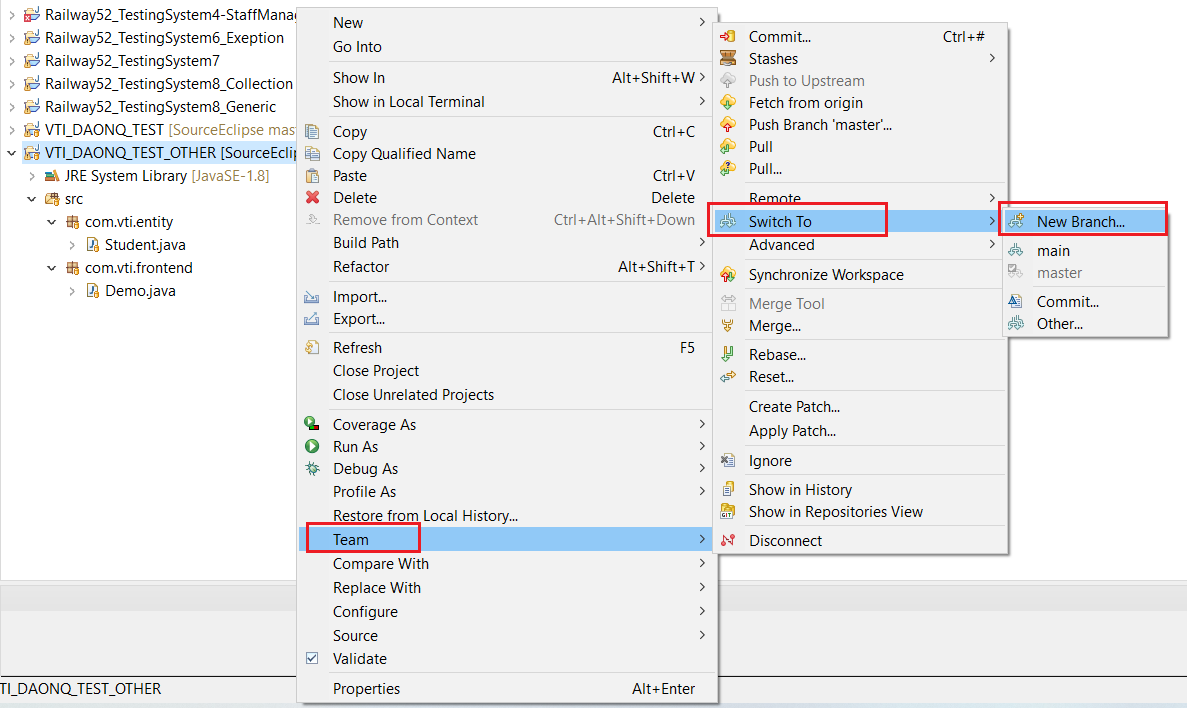
* **Bước 6: Thực hiện tạo các Branch để làm việc**
* Từ giao diện GitHub:

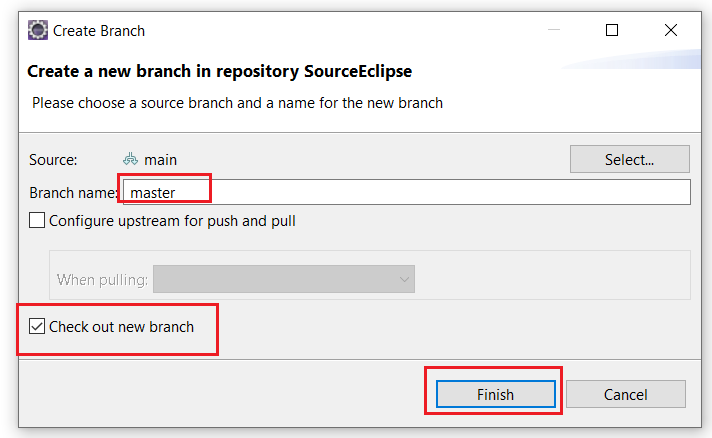


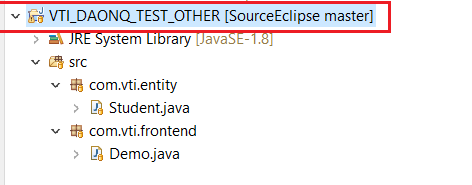
* Thực hiện tạo nhánh mới: Branch1



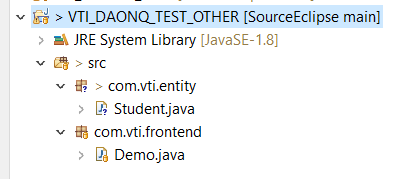
* Thực hiện tạo 1 nhánh mới: **master**





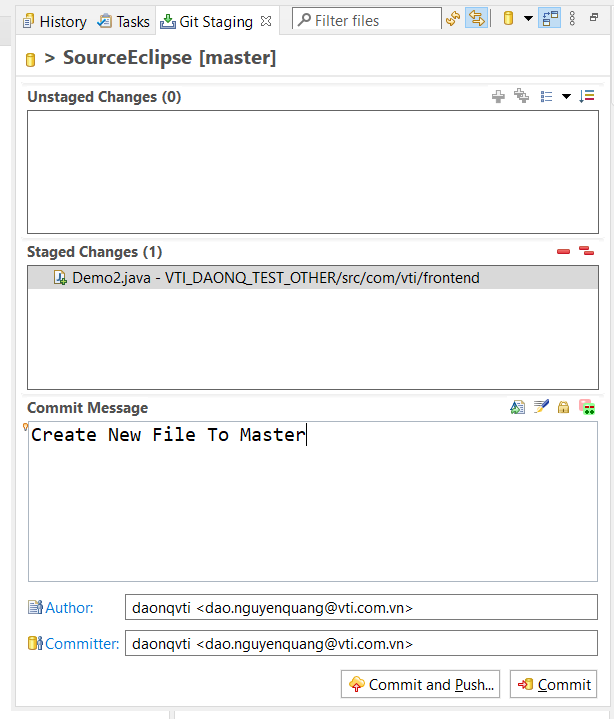


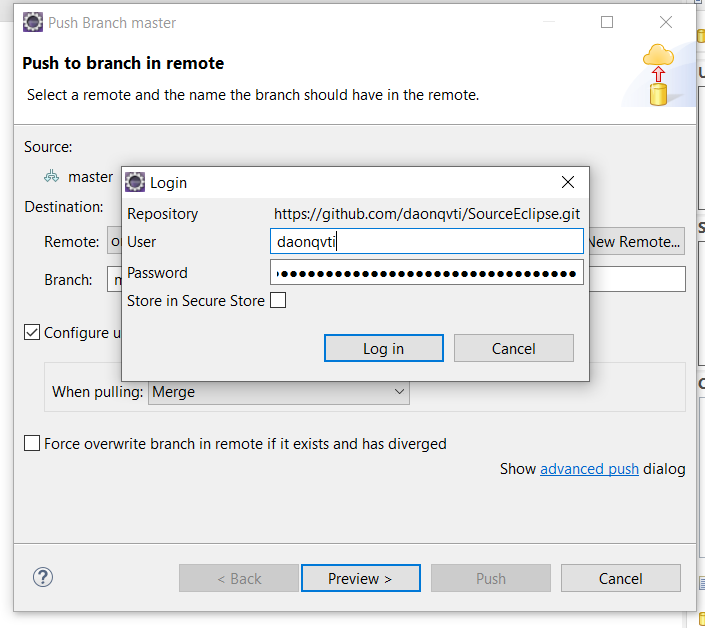
* Tại Project: VTI\_DAONQ\_TEST\_OTHER, tạo thêm 1 vài file Code mới:

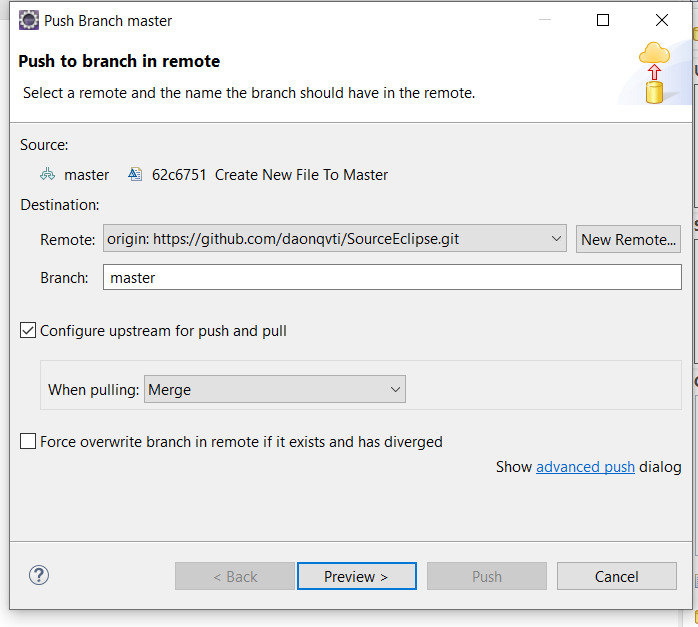
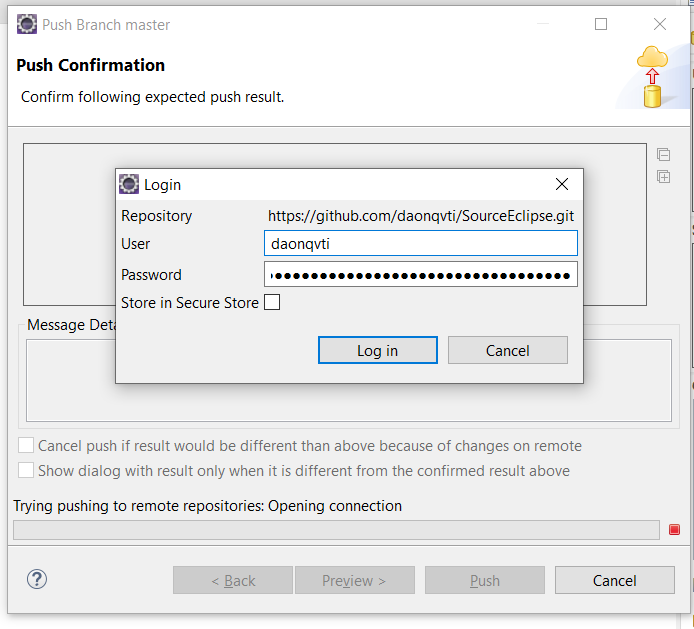
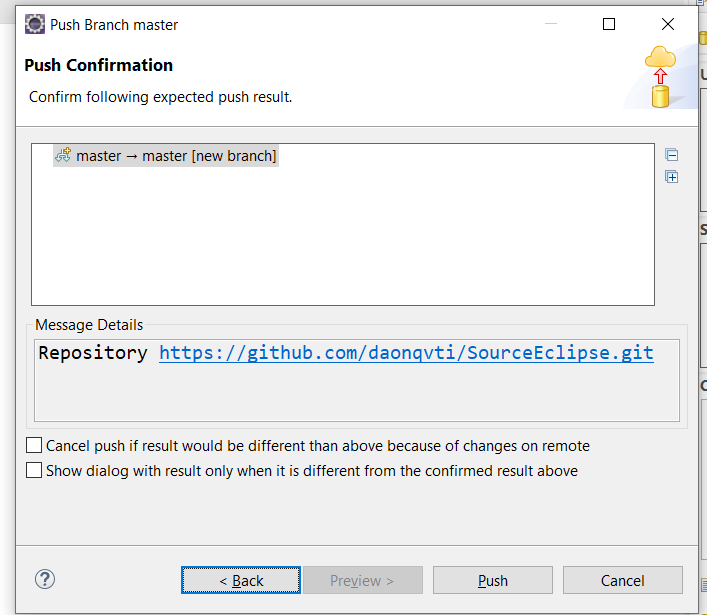


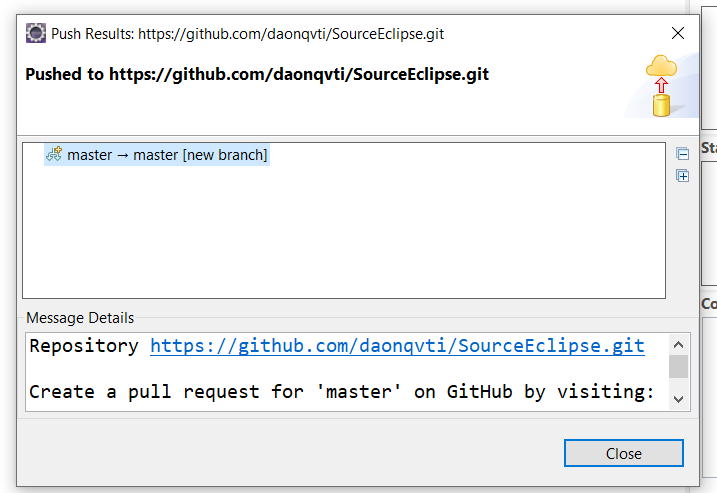
* Thực hiện Commit vào Push lên trên nhánh Branch1 vừa tạo:









* Kiểm tra lại trên GitHub:

