# THỰC HÀNH SỬ DỤNG SVN

## Yêu cầu

* Cài đặt phần mềm TortoiseSVN
* Tạo project bất kỳ để thực hành
* Đưa mã nguồn lên SVN
* Tạo ít nhất 10 commit – có ghi chú cho mỗi commit rõ ràng
* Xem lại lịch sử commit
* So sánh các phiên bản của mã nguồn
* Phục hồi một phiên bản bất kỳ
* Giả lập ít nhất hai tài khoản SVN, tạo xung đột mã nguồn 🡪 Giải quyết xung đột
* Tạo mởi branch/ Merge branch

# HƯỚNG DẪN SỬ DỤNG SUBVERSION (SVN)

## Giới thiệu.

Hệ thống SVN ([http://subversion.tigris.org/)](http://subversion.tigris.org/) là một hệ thống quản lý phần tài nguyên của một dự án. Hệ thống có khả năng tự cập nhật, so sánh và kết hợp tài nguyên mới vào phần tài nguyên cũ.

Phần mềm TortoiseSVN [(http://tortoisesvn.net/)](http://tortoisesvn.net/) là client tốt nhất của hệ thống SVN.

Các chức năng mà các bạn cần quan tâm nhất là chức năng Commit – để đưa các dữ liệu của các bạn lên thư mục dự án (chú ý là mỗi lần các bạn đưa lên thì sẽ là một phiên bản (revision) của dự án nên các bạn chỉ nên đưa lên server một khi đã hoàn thành cơ bản chạy được một tính năng nào đó không có lỗi để tránh trường hợp có người khác update về chạy bị lỗi ở phần của các bạn mà chương trình không chịu chạy, như vậy rất nguy hiểm). Và tính năng thứ 2, là update, để cập nhập tài nguyên của dự án, phần mềm sử tự so sánh mã nguồn từ server và của các bạn và tìm ra chỗ nào khác sẽ tự động nối (merge) vào mã nguồn của các bạn đảm bảo không bị lỗi). Các bạn mỗi khi commit khuyến khích nên update xong rồi hãy update để tránh tình trạng bị lỗi.

Hướng dẫn ở dưới chỉ có hướng dẫn vài tính năng cơ bản, các bạn nên đọc thêm phần hướng dẫn của tortoiseSVN để biết thêm nhiều tính năng rất mạnh của nó hoặc khi có vấn đề gì về lỗi conflic có thể mail về cho trưởng nhóm để nhận được hướng dẫn.

Tùy theo tình hình của mỗi dự án mà nhóm quản lý sẽ cấp user để các bạn tham gia dự án. Các bạn sẽ nhận được username và password và url đường dẫn đến thư mục của dự án trên mạng.

## Hướng dẫn sử dụng Subversioning (SVN) dựa trên TortoiseSVN client.

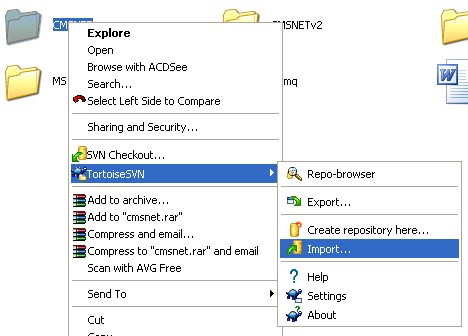
* **Mục tiêu chính của SVN :**
* Quản lý các phiên bản của một file nào đó (bất kỳ là file text, binary,...) vào server SVN.
* **Quá trình kiểm tra sẽ được SVN sẽ kiểm tra tại 2 nơi**
* 1. Thư mục dự án (Nơi thực hiện dự án).
* 2. Thông tin dự án được lưu trữ trên server.

khi kiểm tra file hiện hành có sự thay đổi so với trên server SVN là SVN sẽ tạo một phiên bản mới lưu lại (khi người dùng update -> commit). Việc quản lý phiên bản sẽ giúp việc phục hồi lại các version trước đó dể dàng mà không cần phải lưu backup nhiều mà không biết cái nào trước cái nào sau ... Khi lưu từng phiên bản sẽ cho phép lưu kèm theo các thông tin liên quan như ...ai là người thay đổi, thay đổi chổ nào, thay đổi cái gì ... giúp cho việc quản lý source nguồn một các rõ ràng. Việc lưu trữ version SVN sử **dụng** các thuật toán so sánh và chỉ lưu trữ các thành phần khác nhau của mỗi verision nên dung lượng lưu trữ sẽ được tối ưu so với việc backup từng file.

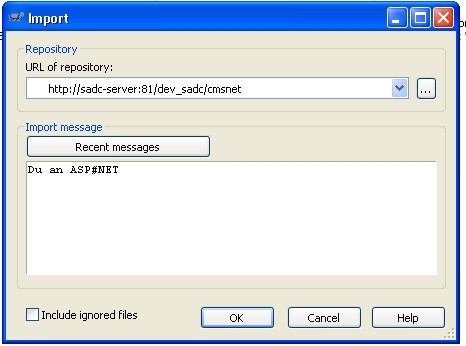
* **Các quy định khi làm việc với SVN:**
* Khi thực hiện mới dự án: Nếu là một dự án mới thì quá trình thực hiện dự án như thường lệ, **sau đó “Import” dự án vào SVN.** Cách thực hiện như sau: Nhấn chuột phải trên thư mục dự án chọn **TortoiseSVN → Import...**
* Khi thực hiện upgrade **từ một dự án có sẵn trong SVN :** Nếu phải thực hiện lại một dự án có sẵn thì phải “checkout” dự án đó từ SVN về thư mục thực hiện dự án. để SVN khởi tạo các thông tin quản lý source nguồn trên thư mục làm việc của dự án. Cách thực hiện như sau : Nhấn chuột phải trên thư mục (rỗng), nơi dự định để thực hiện dự án **TortoiseSVN** → **Checkout...**
* **Đăng ký tài khoản để lưu trữ mã nguồn dự án (server)**
* https://www.assembla.com/subversion
* https://deveo.com/svn-hosting/
* **Một số thao tác cơ bản với SVN**

Đưa một project vào SVN server: Mở windows explorer lên, chọn thư mục của dự án thực hiện "right click" và chọn **TortoiseSVN** → Import... Các bước như sau:

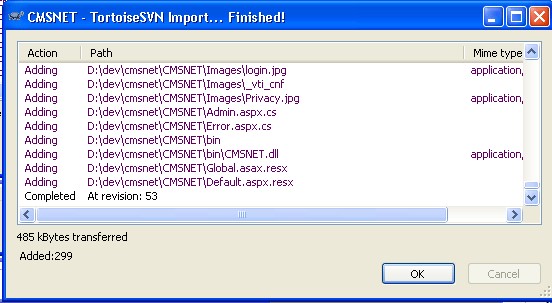
Bước 1: Chọn chức năng import



Bước 2: Điền các thông tin cần thiết (khi truy cập vào url server thì sẽ có dialog yêu cầu nhập thông tin username + password đã gửi)



Bước 3 : Hoàn thành.



**Lấy một project từ SVN**

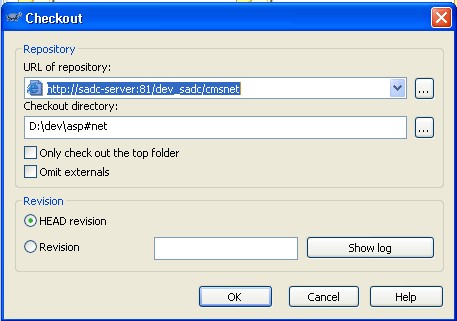
Khi cần lấy các source code từ server SVN thì thực hiện các bước như sau: Mở windows explorer lên, chọn (hoặc tạo mới) một thư mục để chưa các source code của dự án (có thể là thư mục của webserver)

@chú ý : thư mục phải rỗng.

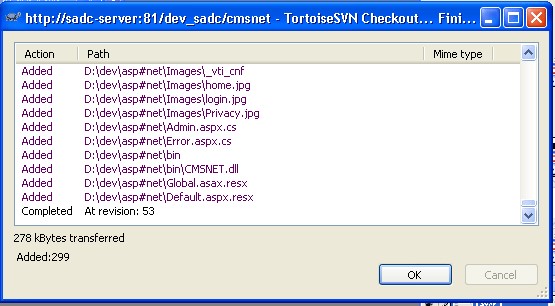
Bước 1 : chọn thư mục chứa dự án



Bước 2 : chọn thông tin dự án trên server



Bước 3: Hoàn thành, bạn đã có các thông tin của dự án trên máy của mình



**Kiểm tra các file bị thay đổi với server SVN**

**Kiểm tra sự thay đổi của source nguồn:**

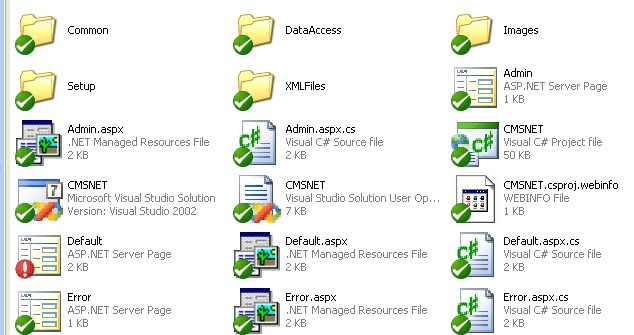
Nếu chưa có bất kỳ sự thay đổi nào thì icon của thư mục dự án như sau



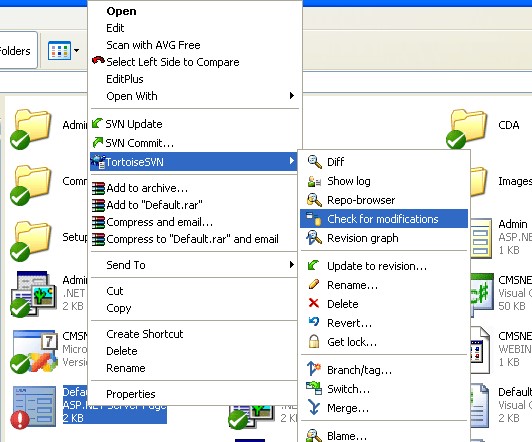
Nếu có một file, hay một thư mục nào có sự thay đổi thì icon cùa thư mục như sau



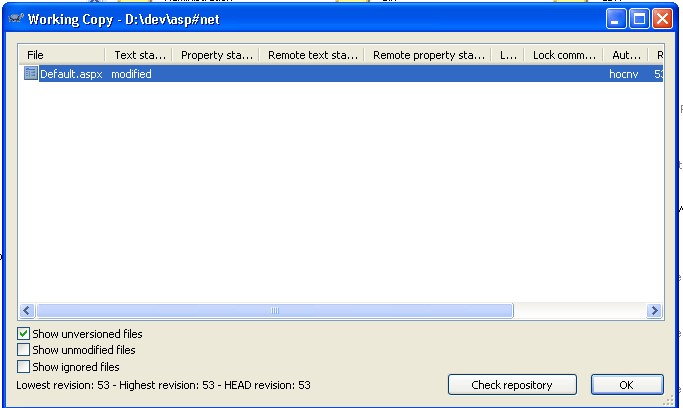
Cho biết file bị thay đổi



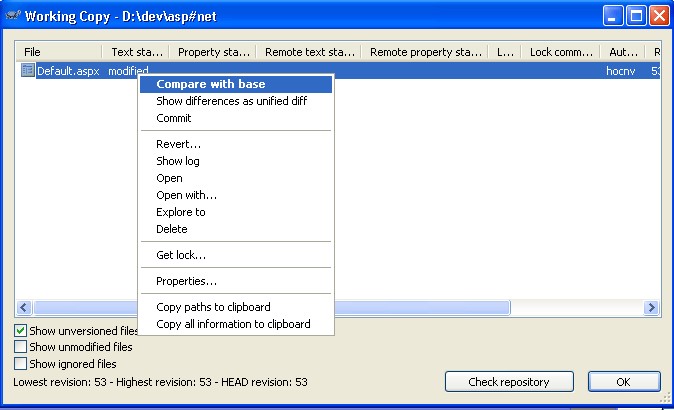
Xem file đó thay đổi nội dung gì



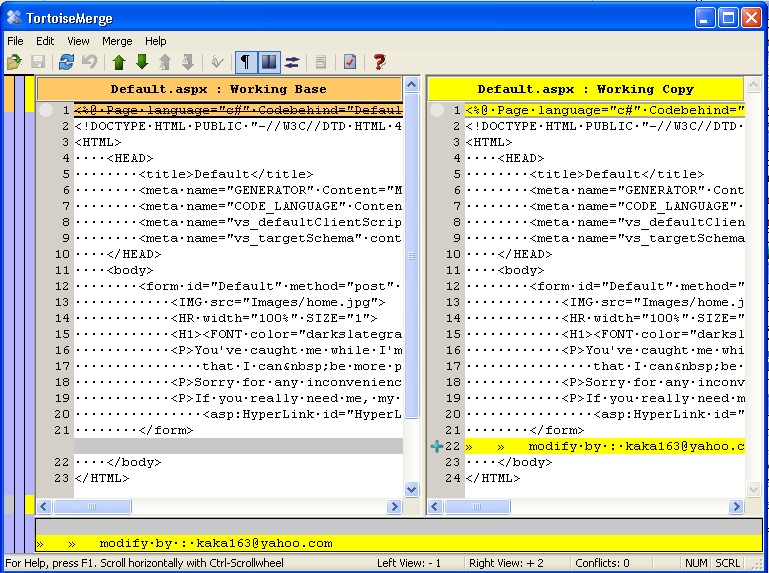
Thông tin thay đổi



Nhấn chuột phải ... tìm chức năng “Compare with base”



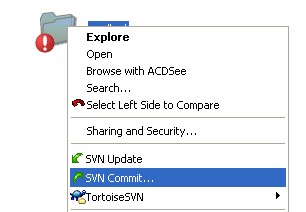
Tìm xem thử coi chi tiết là nó thay đổi cái gì ?



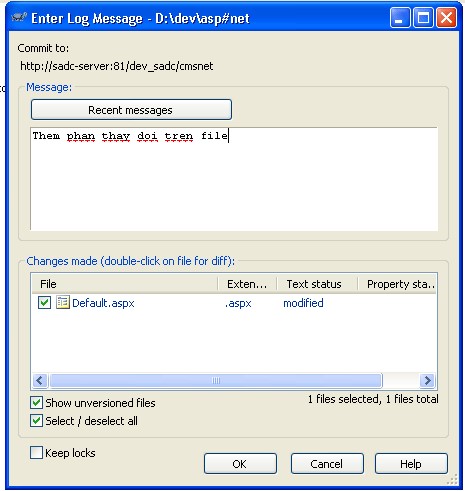
Vậy là biết cái file đó đã bị thay đổi như thế nào ....

Đ**ưa thông tin thay đổi vào server SVN**

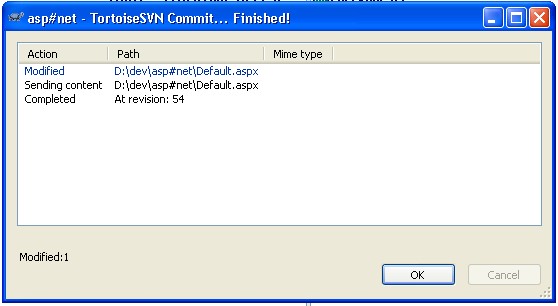
Dùng chức năng “Commit ...” để đưa thông tin vào server



Check vào các file cần commit, hoặc…bỏ bớt một số file chưa cầu commit (nếu là nhiều file)



Hoàn thành…



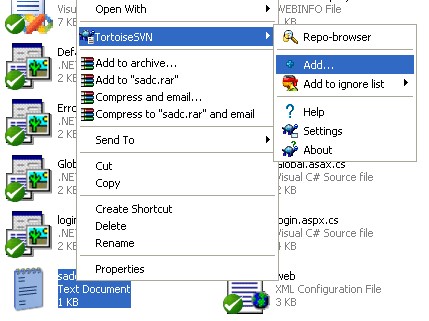
Sau khi commit các thông tin giữa clien và version mới nhất tại server hoàn toàn giống nhau thì icon của thư mục thư mục ..trở về như cũ.



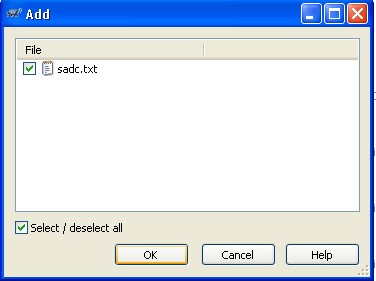
**Đưa file mới vào SVN**

Nếu trong quá trình phát triển dự án cần thêm một số file vào dự án .... thì sau khi thêm vào các bạn chọn chức năng “add” để đưa các file này vào SVN

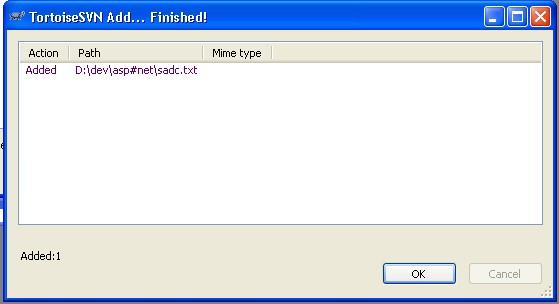
Bước 1: Nhấn chuột phải vào file chưa có icon SVN



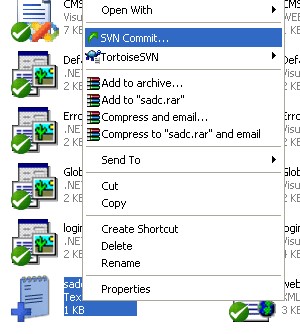
Bước 2: Chọn các file cần đưa .... nếu trong bước 1 là một thư mục



Bước 3: Ngồi đợi ...và xong .

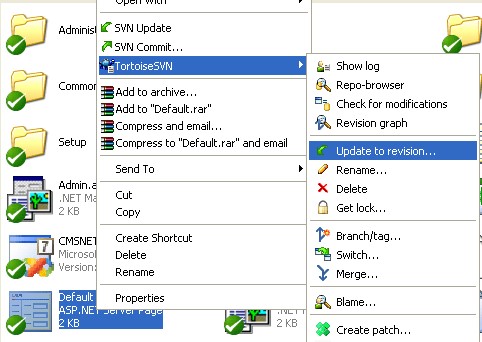


Bước 4: sau khi đưa lên thì phải ... commit các thông tin đó quá trình commit thì tương tự như các bước trên (cập nhật file có thông tin thay đổi).

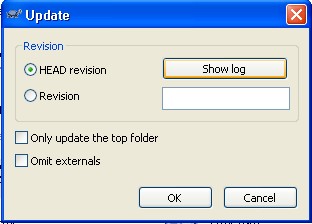


Lấy lại phiên bản trước đó về client (thư mục thực hiện dự án)

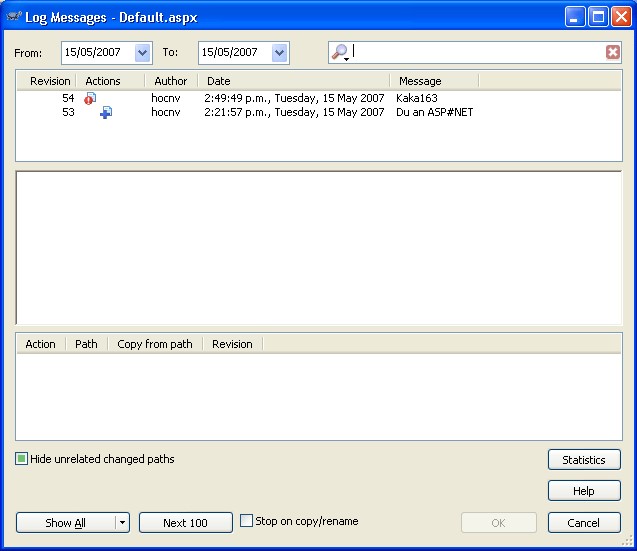
Chọn file cần muốn lấy lại các version (phiên bản) trước đây



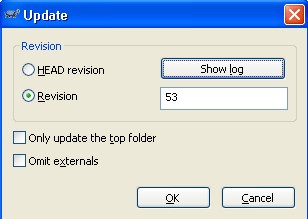
Dialog yêu cầu thông tin version nào



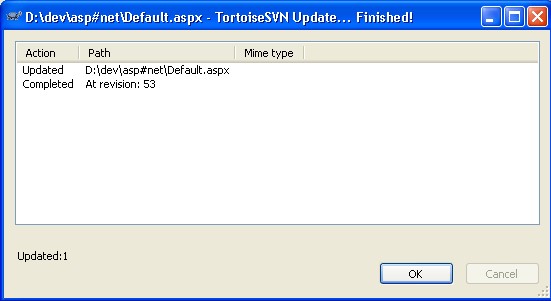
Không biết thì click vào “Show log”



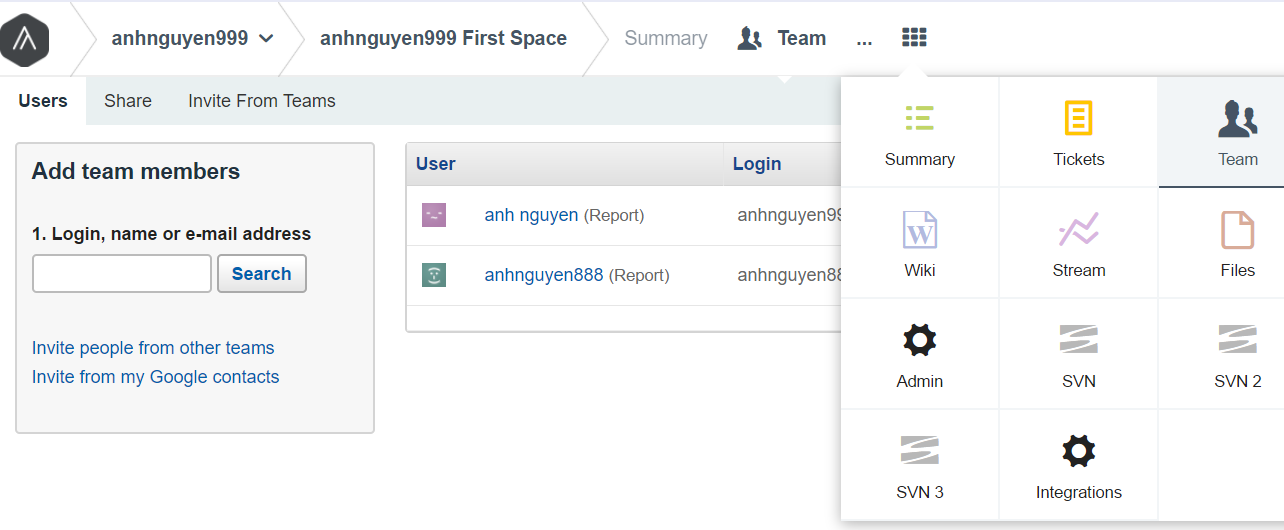
Chọn được ...một version ưng ý (con số 53)



Hoàn thành,… file trên thư mục local đã được lấy lại các thông tin của version 53 trước đó

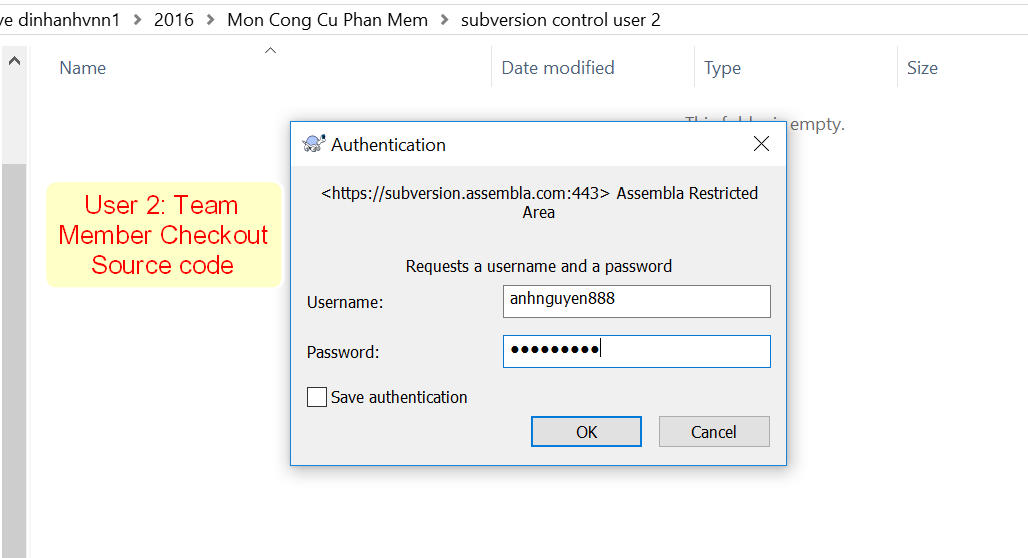


Thêm các thành viên trong dự án vào TEAM (Quản lý trên host site SVN)

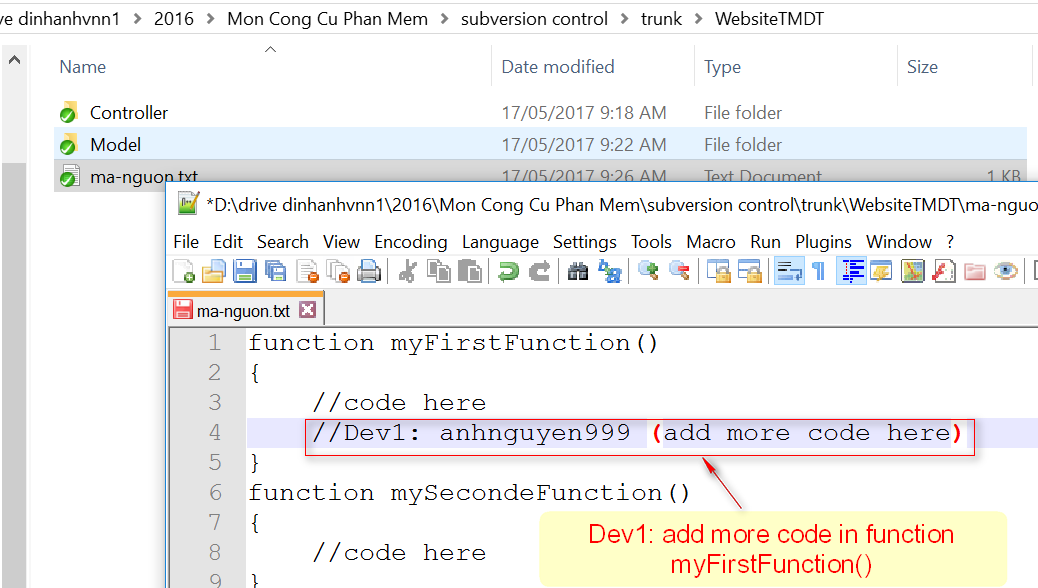


**Giải quyết xung đột mã nguồn**

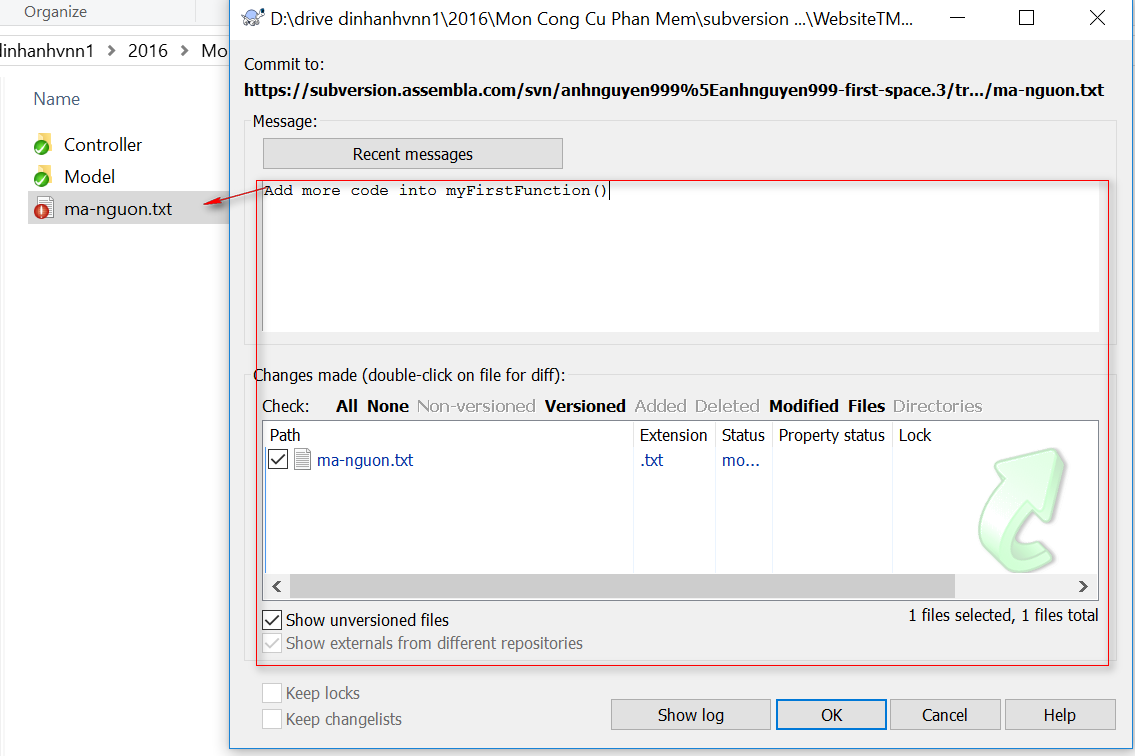
Dev2: Thành viên dự án checkout mã nguồn về local



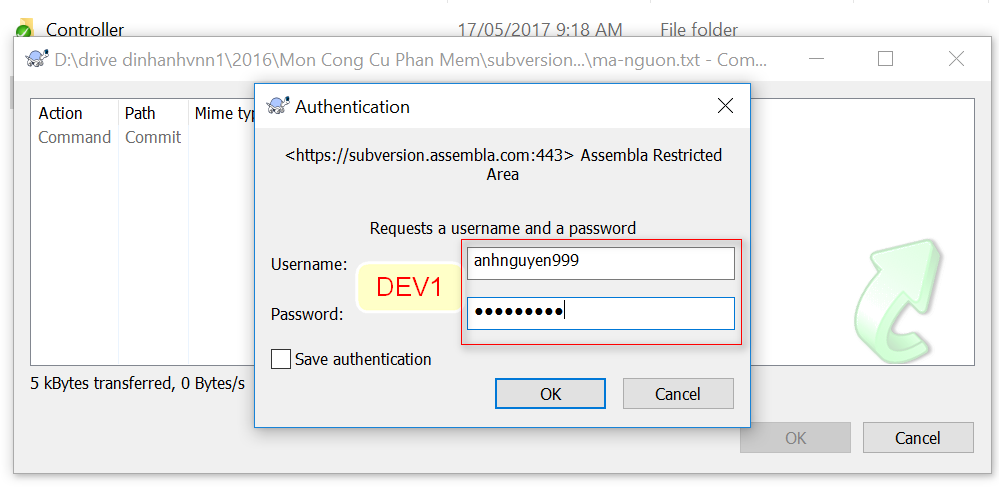
Dev1: Chỉnh sửa mã nguồn dự án và commit lên Server



Commit

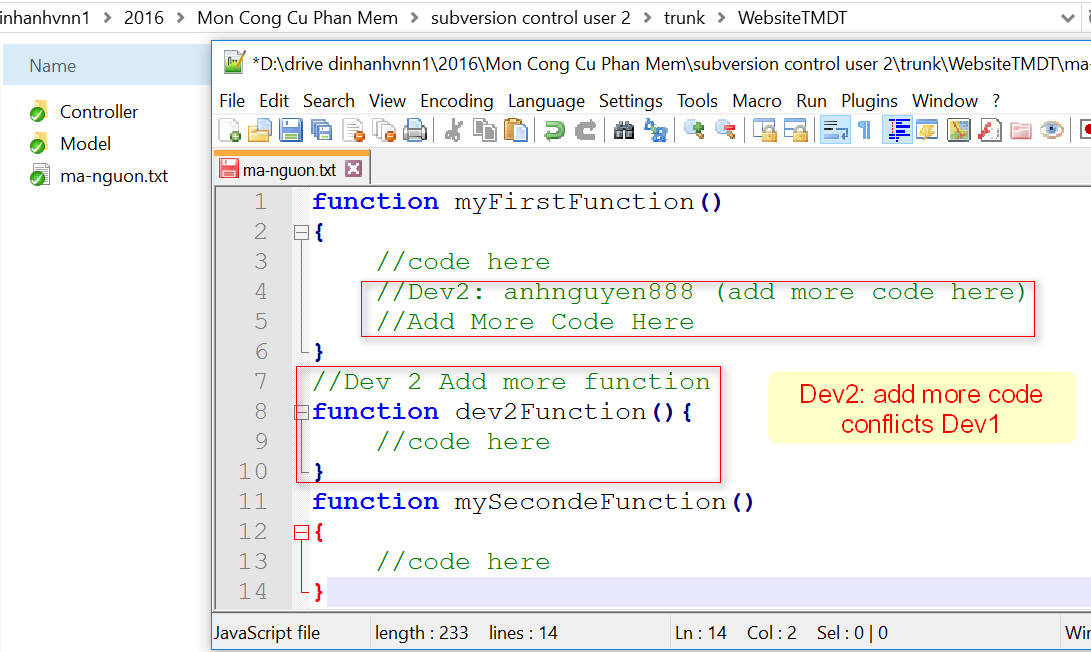


Nhập thông tin Dev1

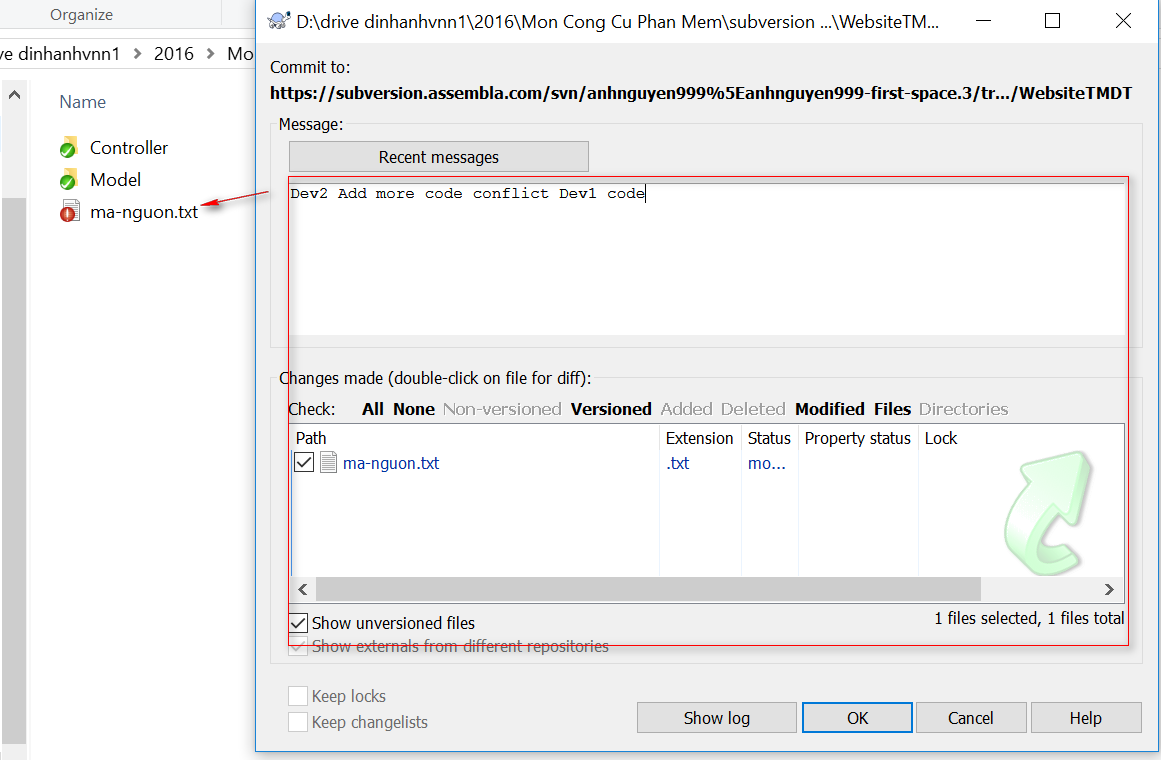


Hoàn thành commit sự thay đổi lên server

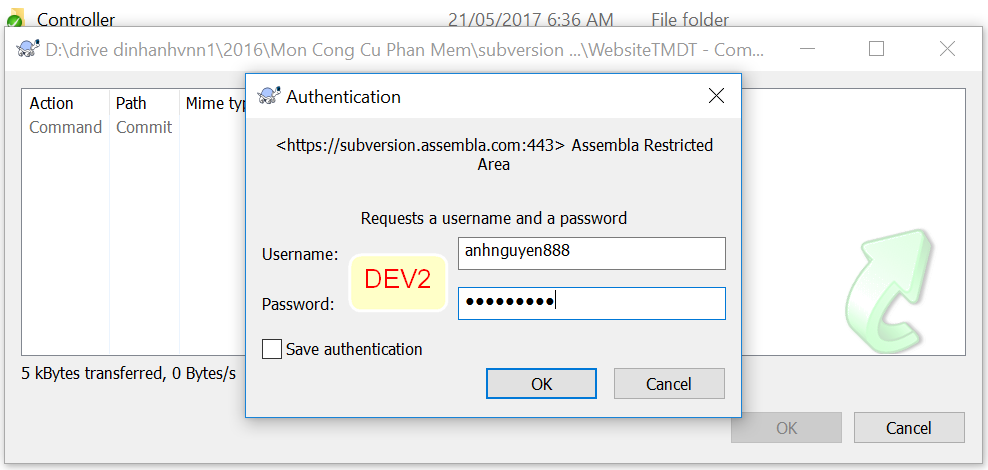
Trong khi đó Dev2 cũng đồng thời sửa trong cùng file ma-nguon.txt và tiến hành commit lên server tương tự Dev1



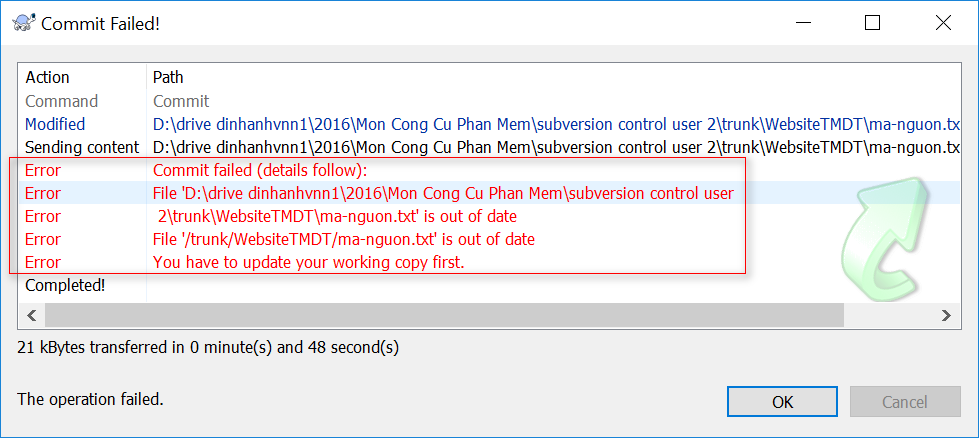
Lưu lại sự thay đổi trên file và commit lên server tương tự.



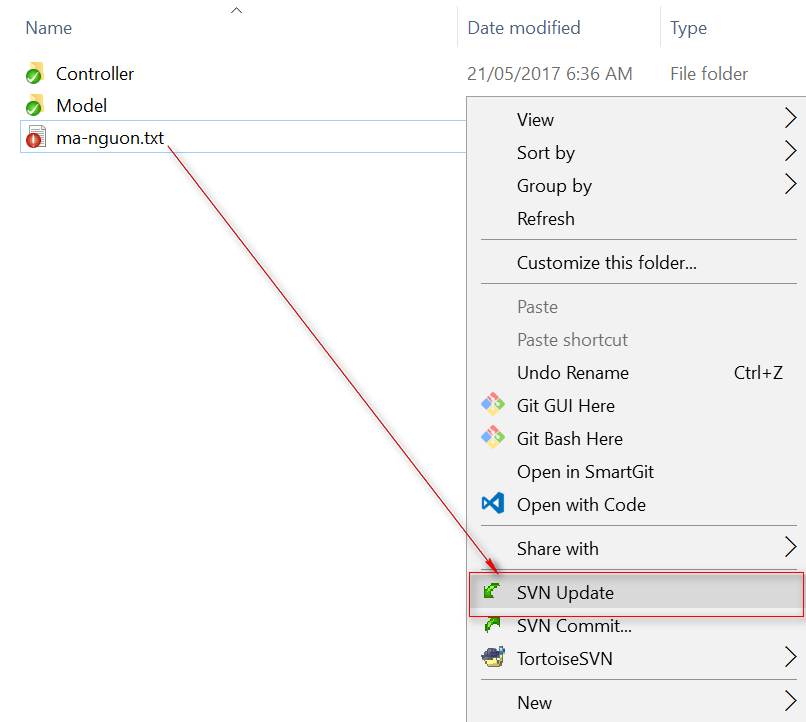
Commit sự thay đổi lên server 🡪 xảy ra hiện tượng conflict (xung đột mã nguồn)

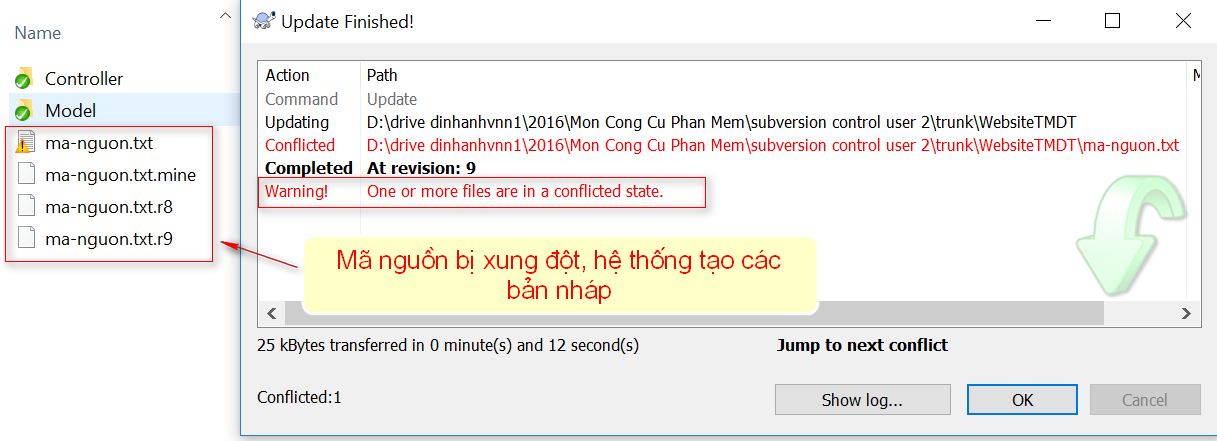


Conflict (xung đột)

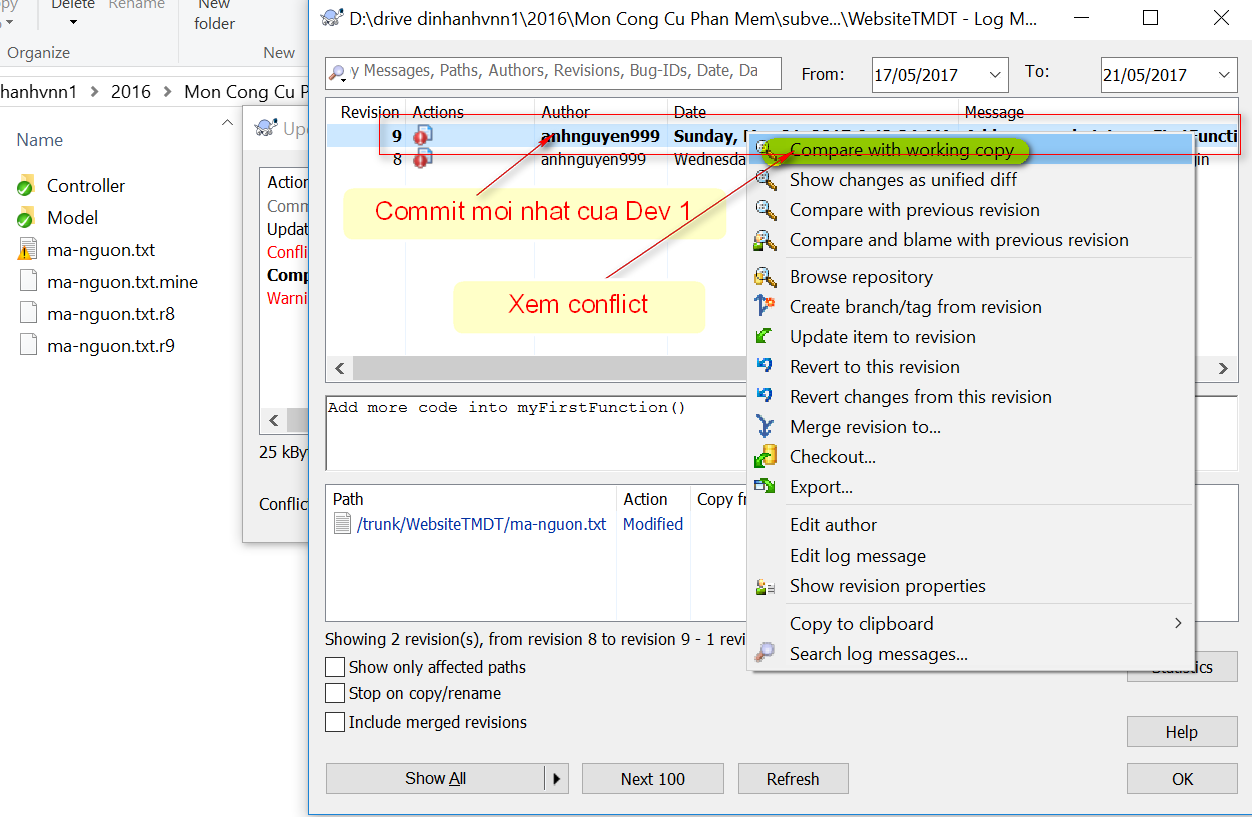


Update mã nguồn trước khi commit

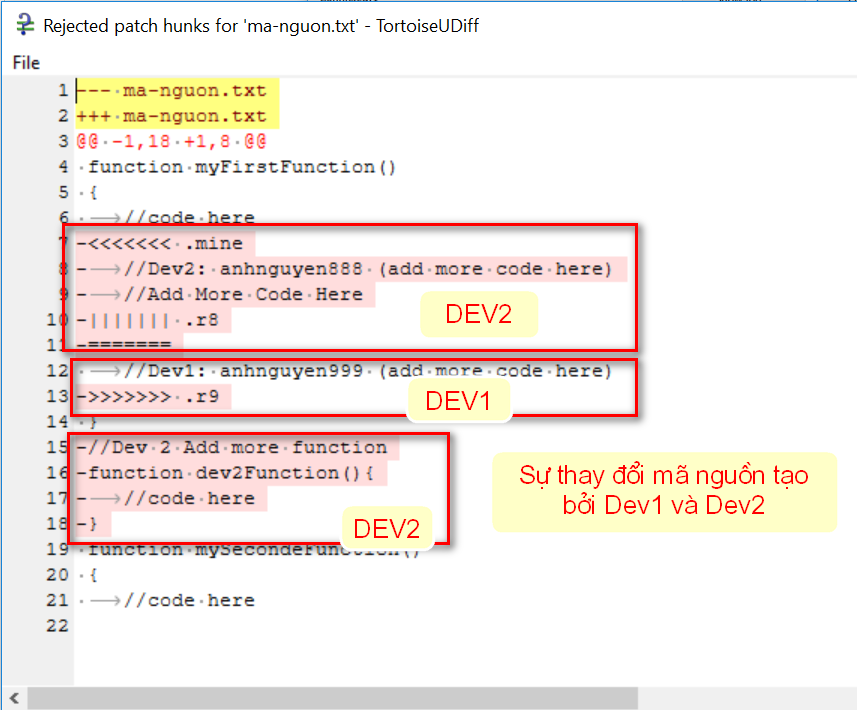




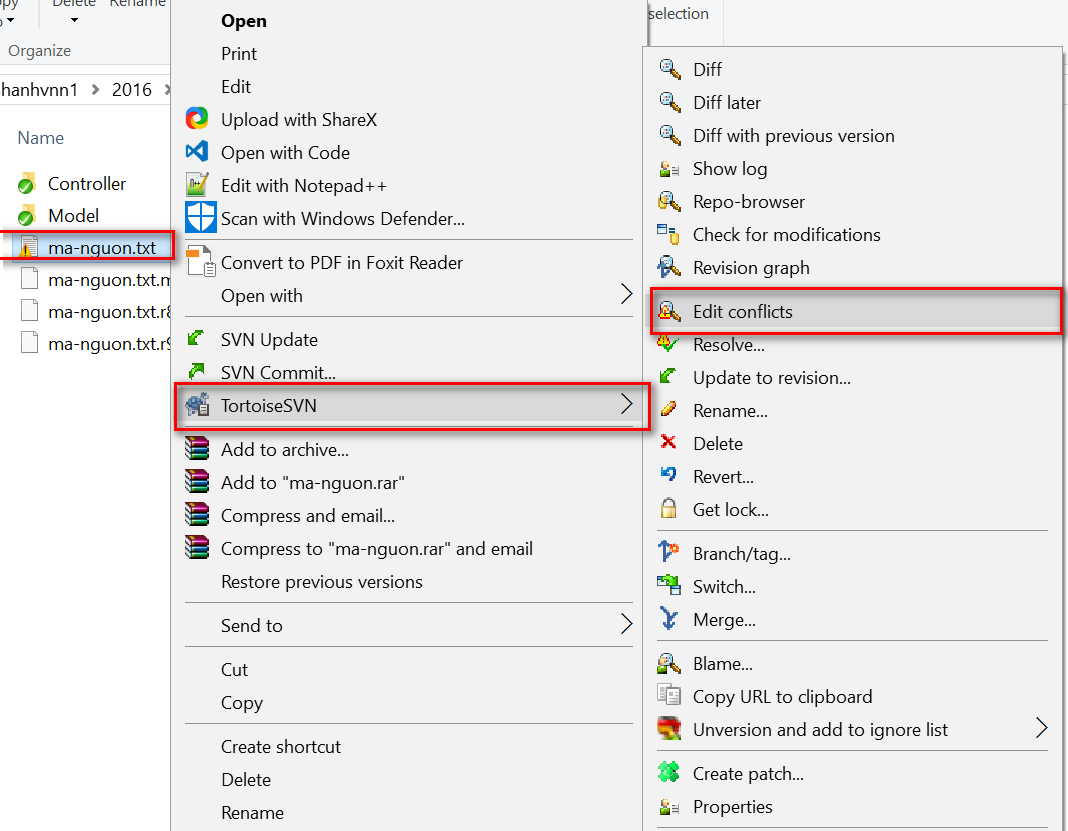
Nhấn vào nút Show log… để xem lỗi conflict



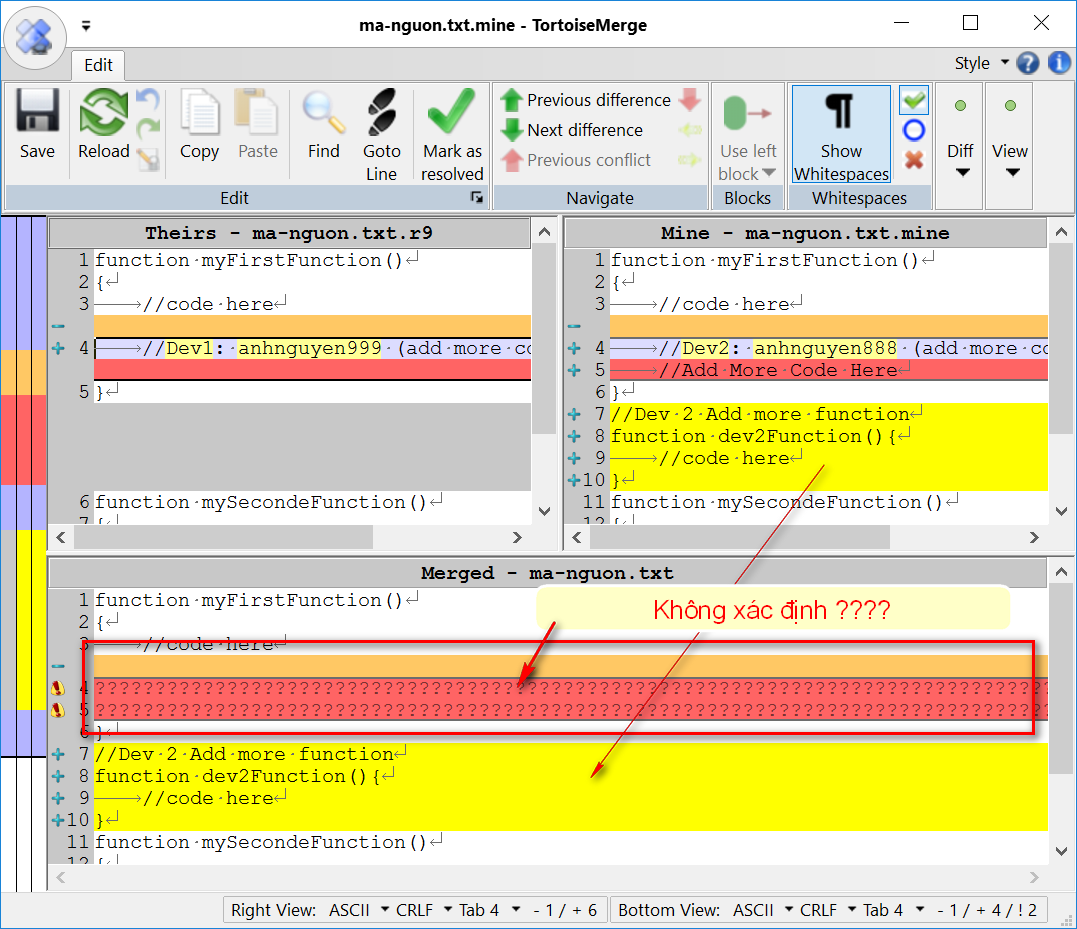
Chọn file cần xem conflict trong lần commit cuối cùng, kết quả hiển thị



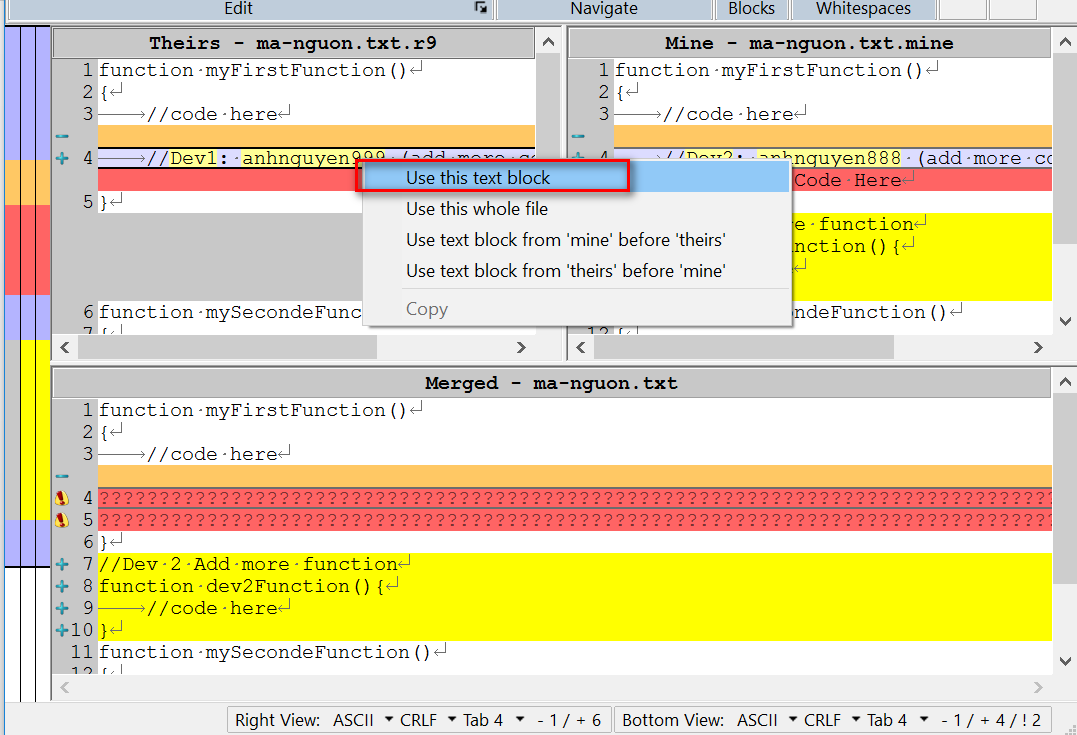
Giải quyết xung đột mã nguồn



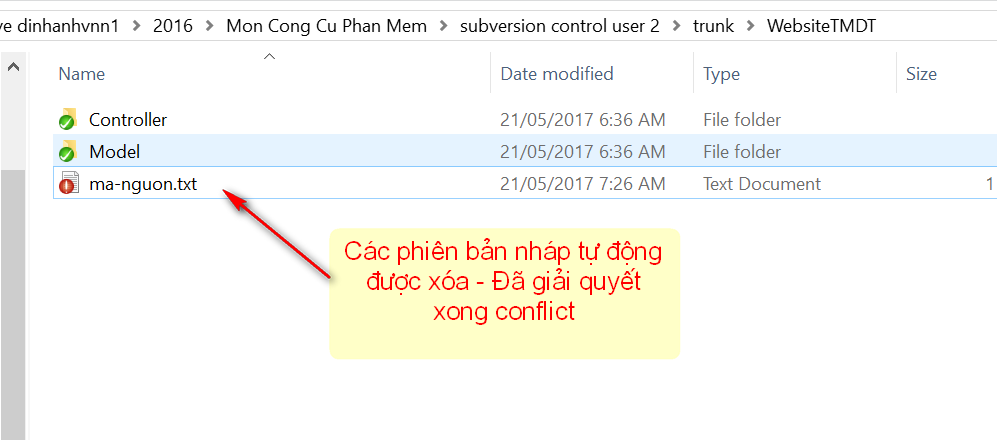
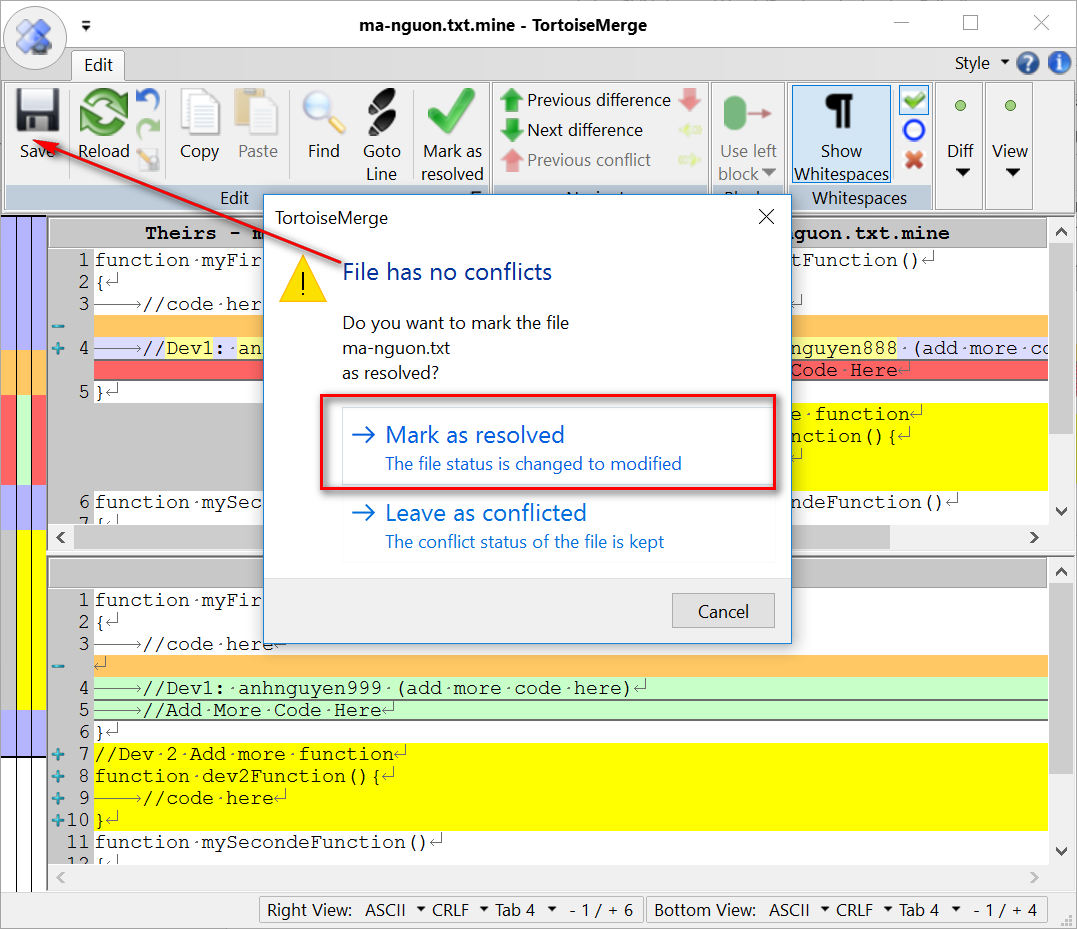
Màn hình tiếp theo



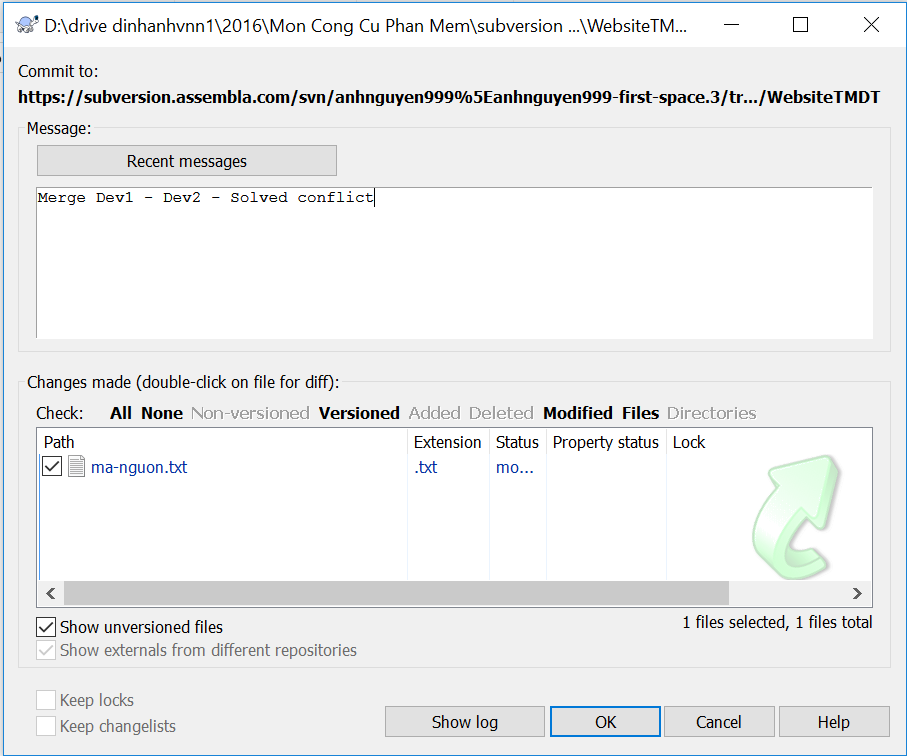
Chọn mã nguồn muốn giữ trong phiên bản Merge



Lưu thay đổi, sau khi hoàn thành việc conflict



Commit mã nguồn.



**Tạo mới branch 🡪 Khi thực hiện một chức năng (feature mới) mà không làm ảnh hướng tới code chính của dự án.**

(sinh viên tự tìm hiểu để thực hiện)

**Merge nhánh về code chính của dự án (trong thư mục /trunk)**

(sinh viên tự tìm hiểu để thực hiện)

- Chúc các bạn thực hành tốt -