Báo cáo thực tập nhóm:

Tìm hiểu về SVN

Người thực hiện : Vũ Đức Độ

Lớp : CNTT14

# Tìm hiểu SVN

## Giới thiệu

* Subversion (viết tắt SVN) là một hệ thống quản lý version (version control system - VCS) được giới thiệu vào năm 2000 bởi công ty CollabNet (http://subversion.tigris.org). Đây là hệ thống hỗ trợ làm việc theo nhóm rất hiệu quả.  
  Phần mềm:  
  Cho client: TortoiseSVN, Download:http://tortoisesvn.net/  
  Cho server: VisualSVN – Server Download: http://tortoisesvn.net/downloads.html  
  Các site cung cấp dịch vụ:  
  [http://hostingviet.vn](http://hostingviet.vn/)<http://sourceforge.net>
* Khi một nhóm làm việc trên cùng một project, việc nhiều người cùng chỉnh sửa nội dung của một file là điều không thể tránh khỏi. SVN Subversion cung cấp các chức năng để có thể thực hiện việc này một cách đơn giản và an toàn.
* SVN Subversion được thiết kế với mục đích thay thế hệ thống quản lý phiên bản Concurrent Versioning System (CVS) đã cũ và có nhiều nhược điểm.
* Subversion có thể được sử dụng để quản lý bất cứ hệ thống phiên bản nào.
* SVN Subversion là hệ thống quản lý source code tập trung (Centralized).
* SVN Subversion là hệ thống quản lý phiên bản mạnh mẽ, hữu dụng, và linh hoạt.
* SVN Subversion quản lý tập tin và thư mục theo thời gian.
* SVN Subversion giống như một hệ thống file server mà các client có thể download và upload file một cách bình thường.
* Điểm đặt biệt của SVN Subversion là nó lưu lại tất cả những gì thay đổi trên hệ thống file: file nào đã bị thay đổi lúc nào, thay đổi như thế nào, và ai đã thay đổi nó.
* SVN Subversion cũng cho phép recover lại những version cũ một cách chính xác. Các chức năng này giúp cho việc làm việc nhóm trở nên hiệu quả và an toàn hơn rất nhiều.
* Thông thường, client và server kết nối thông qua mạng LAN hoặc Internet. Client và server có thể cùng chạy trên một máy nếu SVN Subversion có nhiệm vụ theo vết lịch sử của dự án do các nhà phát triển phần mềm phát triển trong nội bộ.
* SVN Subversion hỗ trợ khá nhiều giao thức để kết nối giữa client và server.
* Ví dụ bạn có thể dùng các giao thức của ứng dụng web như http:// hoặc https://, hay các giao thức của svn như svn:// hoặc svn+ssh://, hoặc nếu phần mềm client và server cài chung trên 1 máy thì có thể dùng file://.  
  Việc cho phép server hỗ trợ giao thức nào phụ thuộc vào lúc cấu hình.
* Cài đặt SVN Subversion (Client): tool dùng trên Client
* Cài đặt VisualSVN(Server): tool dùng cho Server

## Các khái niệm cơ bản

* **Repository (Kho chứa):** Một repository là trái tim của bất cứ hệ thống quản lý phiên bản. Nó là khu vực trung tâm để các lập trình viên giữ tất cả công việc của họ. Kho lưu không chỉ lưu các file mà còn lưu giữ cả lịch sử làm việc. Kho lưu được truy cập thông qua mạng, thực hiện vai trò như một máy chủ và công cụ điều khiển phiên bản là một máy khách. Các máy khách có thể kết nối với repository và sau đó họ có thể giữ/lấy lại những thay đổi họ thực hiện tới/từ repository. Bằng cách giữ các thay đổi , một khách tạo các thay đổi này tới người khác và bằng cách lấy lại những thay đổi, một khách có thể nhận được các thay đổi được thực hiện bởi người khác như là một bản sao công việc.
* **Trunk (Thân):** Một thân là một thư mục tại đó tất cả các phát triển chính xảy ra và thường được kiểm tra bởi các lập trình viên làm việc trên dự án.
* **Tags (Thẻ):** Thư mục các thẻ được sử dụng để giữ các ảnh chụp nhanh (snapshot) được đặt tên của dự án. Hoạt động thẻ cho phép đặt các tên để mô tả và dễ ghi nhớ cho các phiên bản cụ thể trong repository.

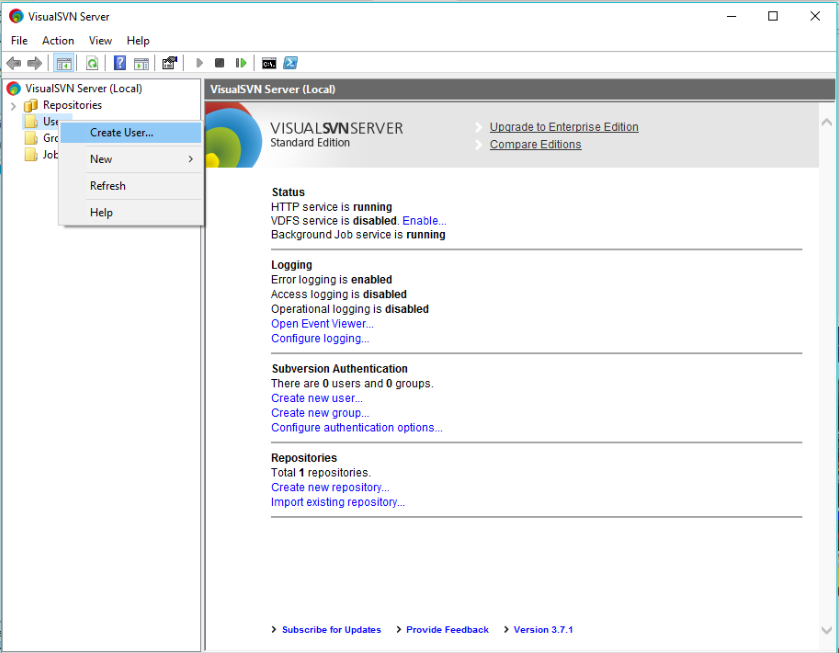
Ví dụ, LAST\_STABLE\_CODE\_BEFORE\_EMAIL\_SUPPORT thì dễ ghi nhớ hơn

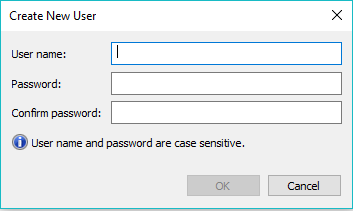
7ceef8cb-3799-40dd-a067-c216ec2e5247 của repository UUID và

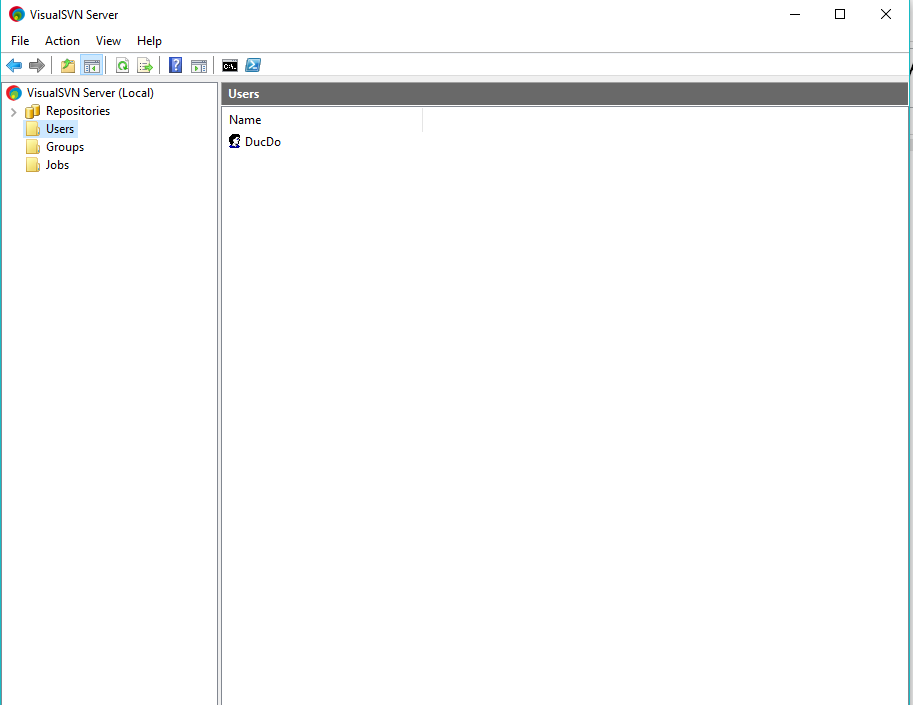
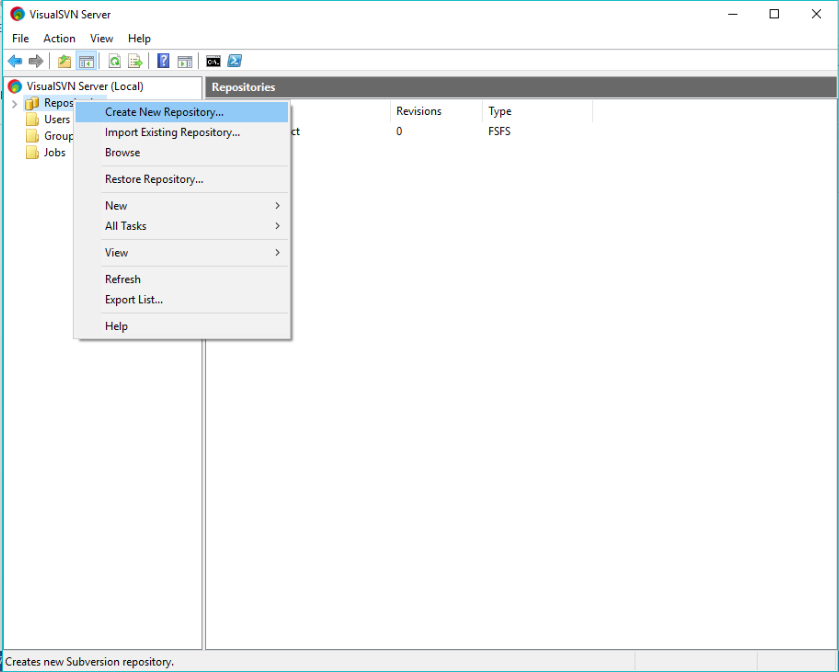
Revision: 13

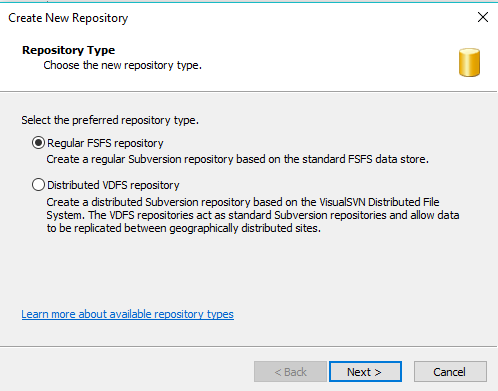
* **Branches (Nhánh):** Hoạt động nhánh được sử dụng để tạo một tiến trình khác của sự phát triển. Nó hữu ích khi bạn muốn tiến trình phát triển của bạn phân theo hai hướng khác nhau. Ví dụ, khi bạn công bố phiên bản 5.0, bạn có thể muốn tạo ra một nhánh để phát triển các tính năng phiên bản 6.0 mà được giữ riêng rẽ với phiên bản 5.0.
* **Working copy (Bản sao công việc ):** Bản sao công việc là hình chụp nhanh (snapshot) của repository. Kho lưu được chia sẻ bởi cả team, nhưng người ta không thể sửa nó một cách trực tiếp. Thay vào đó mỗi nhà lập trình kiểm tra bản sao công việc. Bản sao công việc là khu vực làm việc riêng, tại đó các nhà lập trình có thể thực hiện công việc của họ riêng rẽ một mình, cách biệt với phần còn lại của team.
* **Các thay đổi commit:** Commit là tiến trình của việc giữ các thay đổi từ khu vực làm việc riêng tới máy chủ trung tâm. Sau khi commit, các thay đổi sẵn sàng được sử dụng bởi các thành viên khác trong team. Các thành viên khác có thể lấy lại các thay đổi này bằng cách cập nhật bản sao công việc. Commit là một hoạt động nguyên tử (atomic operation) tức là không thể chia nhỏ hoạt động đó được. Hoặc là nó thành công hoặc là thất bại. Bạn không bao giờ nhìn thấy một commit hoàn thành giữa chừng.

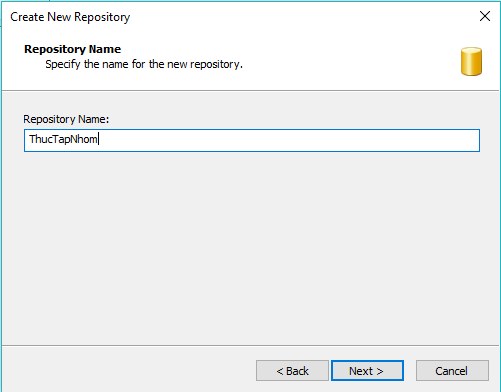
## Cài đặt SVN server và tạo user



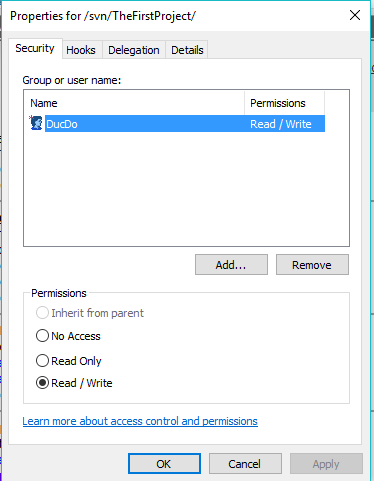


  
  
- Tạo một Reponsitory





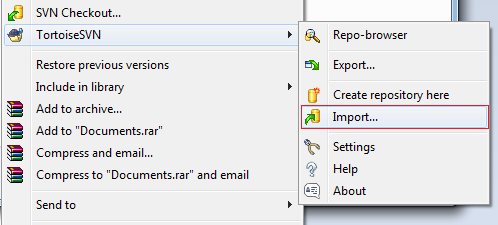
Tạo user: dùng để kiểm soát giới hạn quyền cho các thành viên trong dự án. Nhấp chuột phải vào **repository🡪 properties.** Nhấn Add để thêm user có quyền truy cập vào **Repository** này. Chọn quyền truy cập(Permissions): No Access- Read Only- Read/Write.

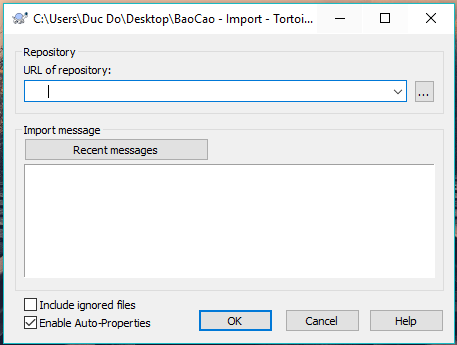


## Cách lưu trữ dữ liệu trên SVN,upload dữ liệu và download dữ liệu về từ Server

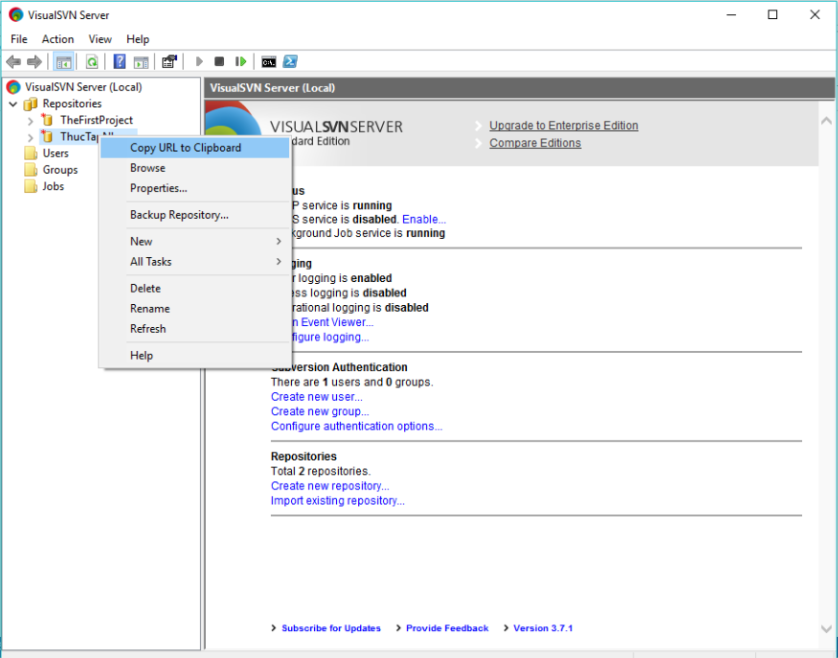
1. Import dữ liệu

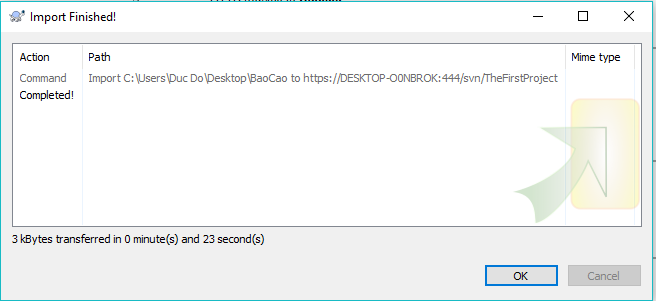
* Tại thư mục muốn đẩy dữ liệu lên server nhấp chuột phải chọn TortoiseSVN 🡪 Import…





* Tại URL of repository ta vào SVN server copy URL



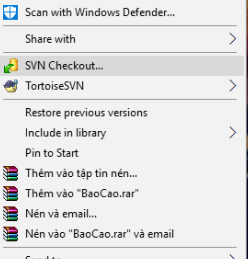


* Như vậy ta đã import thành công

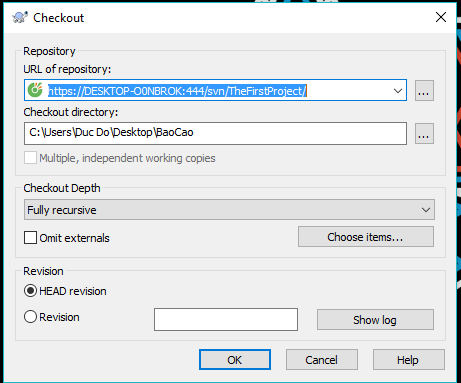
## Checkout dữ liệu từ máy client

Checkout là hành động lấy toàn bộ dữ liệu được chia sẻ trên Repository xuống máy địa phương lần đầu tiên.

Tất cả các thành viên nhóm sẽ checkout các dữ liệu từ trên Repository về máy tính của mình, bao gồm cả người import dữ liệu ban đầu vàoSVN Repository. Tại các máy tính của các thành viên, tạo mới một thư mục rỗng nào đó, và checkout dữ liệu từ SVN Repository về đó. Thư mục này sẽ là thư mục làm việc của mỗi thành viên trên máy tính của mình.



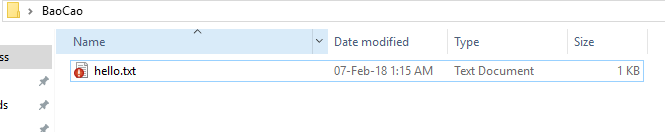
* Copy and paste URL cần checkout



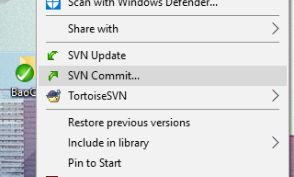
## Commit và update

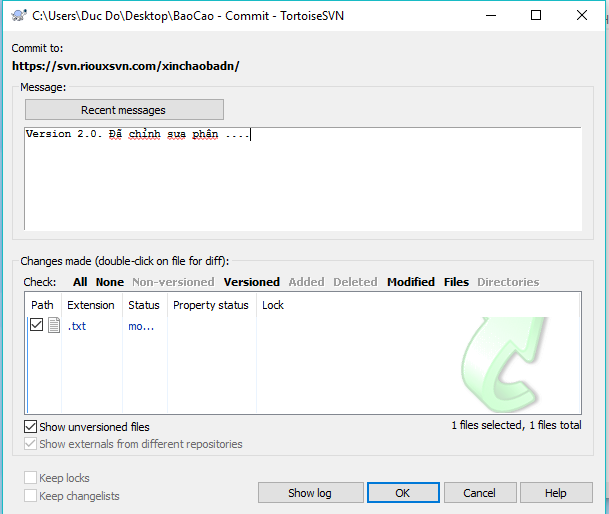
**Checkout**là hành động lấy toàn bộ dữ liệu được chia sẻ trên **Repository**xuống máy địa phương lần đầu tiên. Sau đó dữ liệu được sửa đổi tại địa phương. Việc đưa dữ liệu đã thay đổi lên **Repository**được gọi là **Commit**. Và việc cập nhập các dữ liệu đã thay đổi về là hành động **Update**.

Bây giờ chúng ta sẽ sửa (hoặc thêm) file trên các thư mục rồi **Commit**lên.



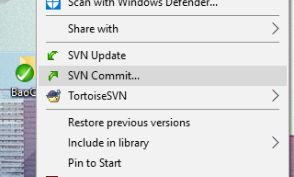
Thư mục sau khi chỉnh sửa sẽ hiện dấu đỏ. Ta tiến hành commit dữ liệu lên

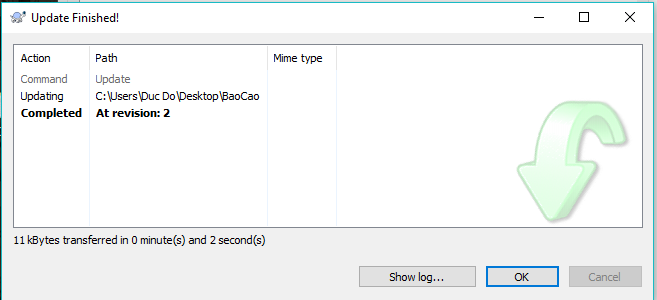




Để lại dòng tin nhắn để thông báo với các thành viên

* Update dữ liệu về máy mình
* Chọn phần update





Như vậy ta đã update dữ liệu về máy thành công.