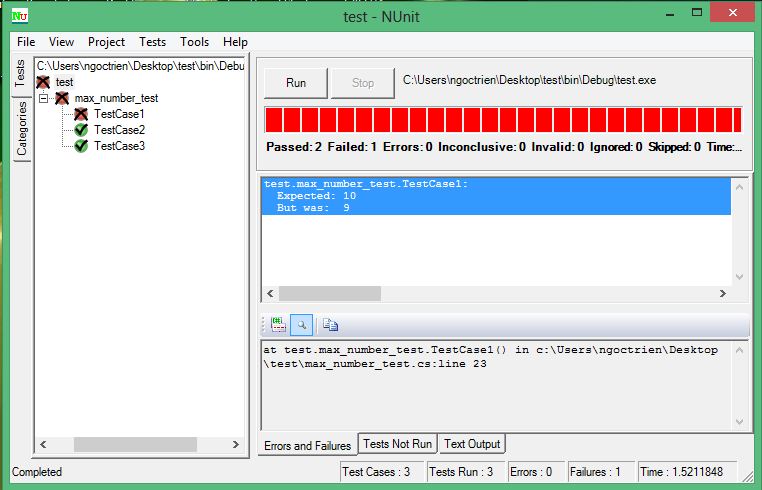
I.CÔNG CỤ KIỂM THỬ NUNIT

* 1. Giới thiệu

NUnit là khung kiểm thử đơn vị chương trình (như lớp, hàm hay module) có mã nguồn mở. Được phát triển theo mô hình Junit (công cụ kiểm thử nổi tiếng cho Java), nhưng được Nunit viết bằng C# và khai thác được các ưu điểm của các ngôn ngữ .NET. NUnit cho phép chúng ta viết cac hàm kiểm thử lỗi theo ngôn ngữ lựa chọn để kiếm thử một chức năng cụ thể của chương trình. Nunit cho phép kiểm thử đơn vị và là cách thức tốt để kiểm thử hoạt động của các đoạn mã viết mới, và cũng là phương thức để kiểm thử hồi quy ứng dụng. Các hàm kiểm thử lỗi có thể lưu và chạy lại mỗi khi chúng ta sửa đổi mã, điều này giúp phát hiện lỗi dễ dàng và đảm bảo phát triển ứng dụng tốt hơn.

Hình 1: Giao diện đồ họa của công cụ Nunit

* 1. Lớp Assert

Lớp Assert được sử dụng trong trường hợp muốn so sánh giữa hai giá trị. Ví dụ, sau khi khởi chạy hàm tính toán các phép tính đơn giản, để khẳng định rằng kết quả trả về có giá trị như dự kiến, thì sử dụng phương thức Assert.Equals.

Các phương thức trong lớp Assert:

* Phương thức tĩnh:
  + NUnit.Framework.Assert.AreEqual
  + NUnit.Framework.Assert.False
  + NUnit.Framework.Assert.AreNotEqual
  + NUnit.Framework.Assert.Contains
  + NUnit.Framework.Assert.True
  + NUnit.Framework.Assert.Fail
  + NUnit.Framework.Assert.Greater
  + NUnit.Framework.Assert.Less
  + NUnit.Framework.Assert.NotNull
  + NUnit.Framework.Assert.Null
  + NUnit.Framework.Assert.ReferenceEquals
* Phương thức động
  + [GetHashCode](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfSystemObjectClassGetHashCodeTopic.asp) (inherited from Object): Phục vụ như một hash chức năng cho một loại, thích hợp cho sử dụng trong các thuật toán hashing và dữ liệu cấu trúc giống như một bảng hash.
  + [GetType](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfSystemObjectClassGetTypeTopic.asp) (inherited from Object): Lấy kiểu của đối tượng hiện hành.
  + [ToString](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfSystemObjectClassToStringTopic.asp) (inherited from Object): Trả về một kiểu chuỗi của đối tượng hiện hành.
  + [Finalize](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfSystemObjectClassFinalizeTopic.asp) (inherited from Object): Cho phép một đối tượng cố gắng thử với mã nguồn mở và thực hiện các hoạt động trước khi đối tượng phản đối lại do tập hợp các dữ liệu vô nghĩa hoặc không tương thích.
  + [MemberwiseClone](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfSystemObjectClassMemberwiseCloneTopic.asp) (inherited from Object): Tạo ra một bản sao của đối tượng hiện hành.
  + [Assert Constructor](http://www.csunit.org/documentation/classdoc.1.8/csUnit.AssertConstructor.html): Khởi tạo một instance mới của lớp Assert.
  1. Các thuộc tính trong Nunit

Người ta thường tạo một project dạng class library dành cho các test class. Project này theo đúng tên gọi của nó chỉ có ý nghĩa để test và không có vai trò gì trong sản phầm phần mềm cuối cùng. Thực ra NUnit có thể test bất kí test class nào bên trong một assembly bất kì nên project test có thể là Console application, window application, v.v nhưng thông thường người ta sẽ chọn project loại class library. Có một lưu ý là test class của bạn phải được khai báo public, test method cũng thế. Khi sử dụng NUnit.Framework, các bạn sẽ phải làm quen với những Attribute như [TestFixture], [Test], [Setup], [TearDown],…

*- [TestFixture]:* dùng để đánh đấu 1 class là test class, những class khác không có Attribute này sẽ mặc định bị ignore khi NUnit test assembly của bạn.

*- [Test]:* dùng để đánh dấu 1 method là test method, ý nghĩa của nó tương tự như TestFixture nhưng scope ở cấp method.

*- [Setup]:* dùng để đánh dấu một method sẽ được gọi trước khi môt test case được goi. Nếu trong một test class có mười method test, thì mỗi lần một method test được chạy thì Nunit sẽ chạy method được đánh dấu với Setup trước tiên.

*- [TearDown]:* ngược với Setup, chạy sau mỗi test method.

*- [TestFixtureSetup]:* tương tự như Setup nhưng ở cấp của class, khi một test class được test thì methid nào được đánh dấu với attribute này sẽ được chạy trước tiên.

*- [TestFixtureTearDown]:* ngược với TestFixtureSetup.

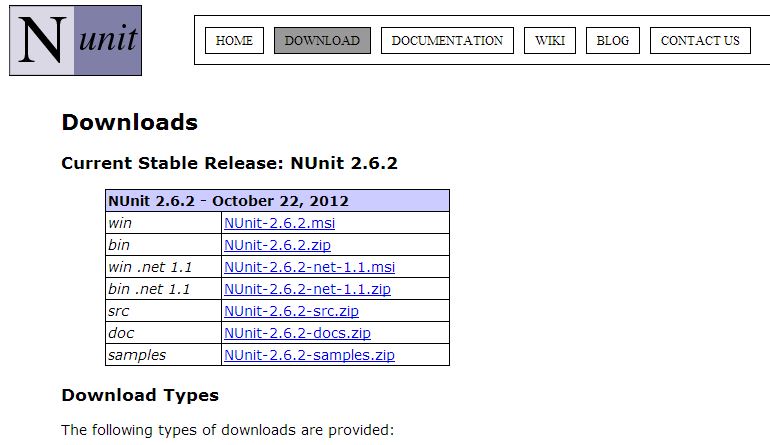
Vậy để apply NUnit Test thì công việc vô cùng đơn giản: tạo một project class library, thêm reference đến .dll nunit.framework, thêm 1 class mới, khai báo nó thành public, thêm using nunit.framework, thêm attribute **[TestFixture]** vào đầu của class, viết một method test và khai báo với attribute **[Test]**. Cơ bản như vậy là đủ để test, bạn có thể kết hợp nhiều attribute khác cũng như nguyên tắc Inheritance của lập trình hướng đối tượng để có một project test uyển chuyển. Người ta thường sử dụng **[Setup]** để mở một transaction scope, sau đó dùng **[TearDown]** để roll back transaction khi test các Dao, như vậy sẽ không có dữ liệu bị thêm xóa vào database và bảo đảm dữ liệu test sẽ như nhau trước khi test các method.

II. HƯỚNG DẪN SỬ DỤNG CÔNG CỤ NUNIT

* 1. Download và cài đặt công cụ
     1. Download công cụ

Để download và sử dụng công cụ, truy cập website:

<http://www.nunit.org/index.php?p=download>



Hình 2: Website download công cụ miễn phí

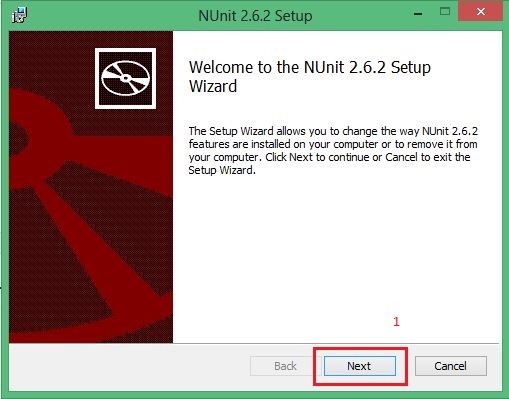
Phiên bản hiện tại (23/02/2013): Nunit 2.6.2

Chọn định dạng file tương ứng để tải.

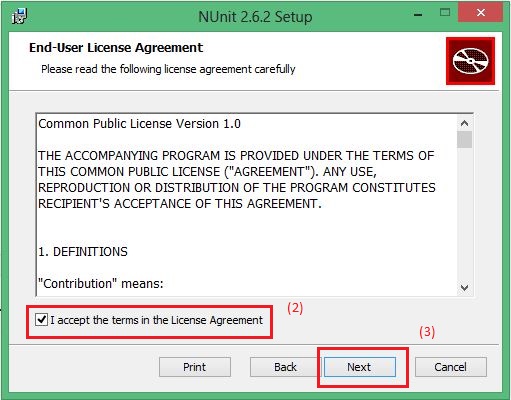
* + 1. Cài đặt công cụ

**Bước 1:** Sau khi download công cụ về ta được file thực thi có tên:

**NUnit-2.6.2.msi.** Chạy file này, xuất hiện màn hình, chọn “**Next**” (1) để tiếp tục.



Hình 3: Tiến trình cài đặt chương trình

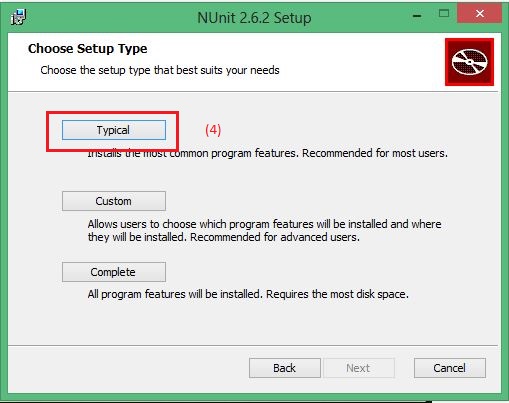
**Bước 2:** Chọn vào checkbox “***I accept the terms in the License Agreement***” (2) để chấp nhận các điều khoản khi sử dụng và tiếp tục chọn “**Next**” (3).

Hình 4: Tiến trình cài đặt chương trình (tiếp theo)

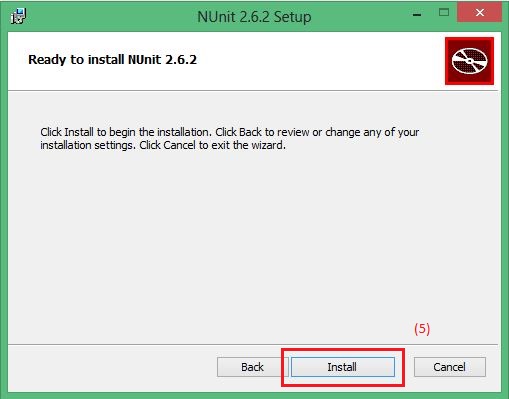
**Bước 3:** Tại đây bạn có thể chọn một trong ba kiểu cài đặt.

* Typical: Cài đặt những ứng dụng phổ biến.
* Custom: Người dùng tùy chọn các ứng dụng muốn cài đặt.
* Complete: Cài đặt tất cả các ứng dụng liên quan.

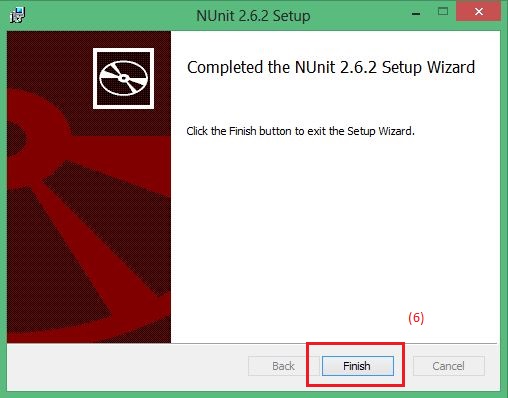
Khuyến cáo chọn Typical (4)



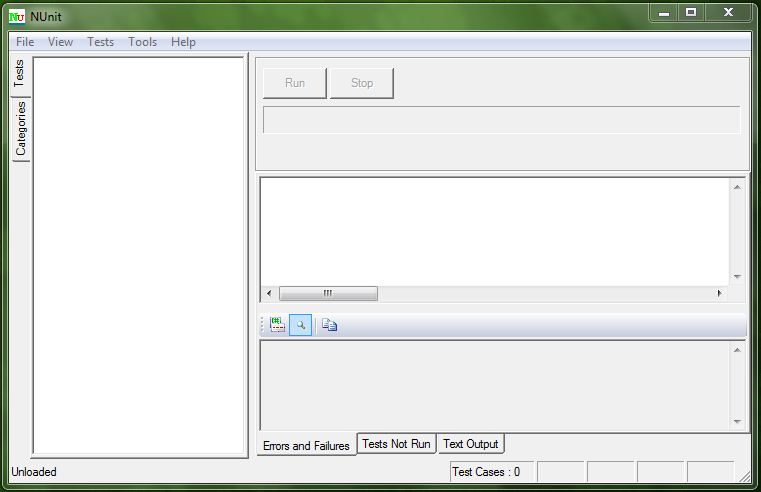
Hình 5: Tiến trình cài đặt chương trình (tiếp theo)

**Bước 4:** Chọn “Install” (5) để tiến hành cài đặt.

Hình 6: Tiến trình cài đặt chương trình (tiếp theo)

Quá trình này diễn ra trong một thời gian ngắn, sau đó chọn “Finish” (6) để kết thúc quá trình cài đặt.

Hình 7: Kết thúc quá trình cài đặt NUnit

***Giao diện chương trình sau khi cài đặt thành công***

Hình 8: Giao diện chương trình sau khi cài đặt thành công

**III.Link video hướng dẫn sữ dụng công cụ.**

**1.link hướng dẫn sữ dụng.**

**-https://www.youtube.com/watch?v=LvBR\_BMmjIc&t=33s**

**-https://www.youtube.com/watch?v=LNw4vwYqedo**