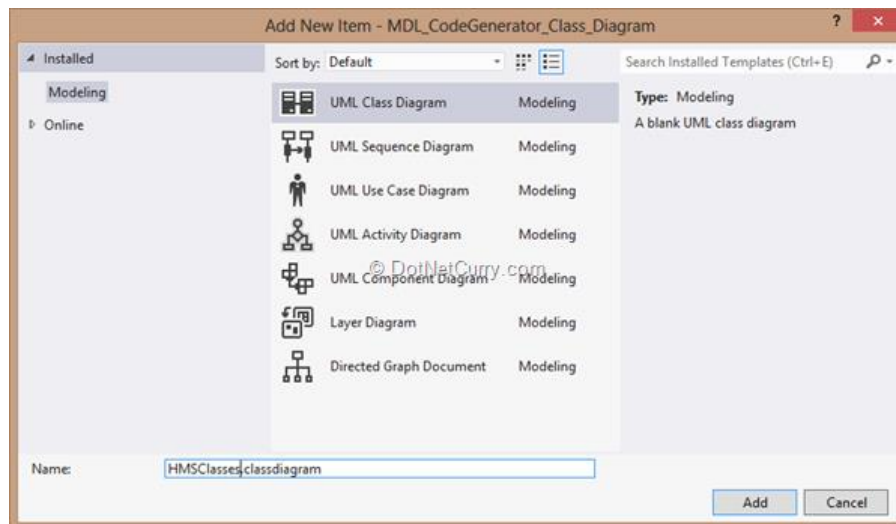


Môn: Phát triển phần mềm hướng đối tượng

Bài thực hành 2

SẢN SINH CODE TỰ ĐỘNG TỪ UML CLASS DIAGRAM TRONG VISUAL STUDIO 2012

Để thiết kế các Diagram trong Visual Studio 2012, chúng ta vào menu "Architecture" -> "New Diagram", lúc đó sẽ hiển thị cửa sổ dưới đây và cho ta tùy chọn nhiều loại Diagram khác nhau:



- Class Diagram:

- Dùng để mô tả các đối tượng được sử dụng trong ứng dụng.
- Giúp ta thấy rõ các đối tượng và mối quan hệ giữa chúng trong ứng dụng.

- Sequence Diagram:

- Dùng để mô tả sự tương tác giữa các đối tượng trong ứng dụng
- Các Diagram này cung cấp thông tin cơ bản về các thông điệp được truyền tải giữa các đối tượng khác nhau trong hệ thống.

- Use Case Diagram:

- Thể hiện sự tương tác giữa người dùng cuối và hệ thống.

- Activity Diagram:

- Thể hiện luồng dữ liệu của hệ thống theo hướng quản lý quy trình ứng dụng.
- Diagram này được xem như là bản mô tả chi tiết của Use Case Diagram.

- Component Diagram:

- Thể hiện mô hình của hệ thống ứng dụng ở mức độ tổng quan, bao gồm các thành phần khác nhau được sử dụng trong ứng dụng, ví dụ như: các thành phần giao diện, các hệ thống Web Server, cơ sở dữ liệu...

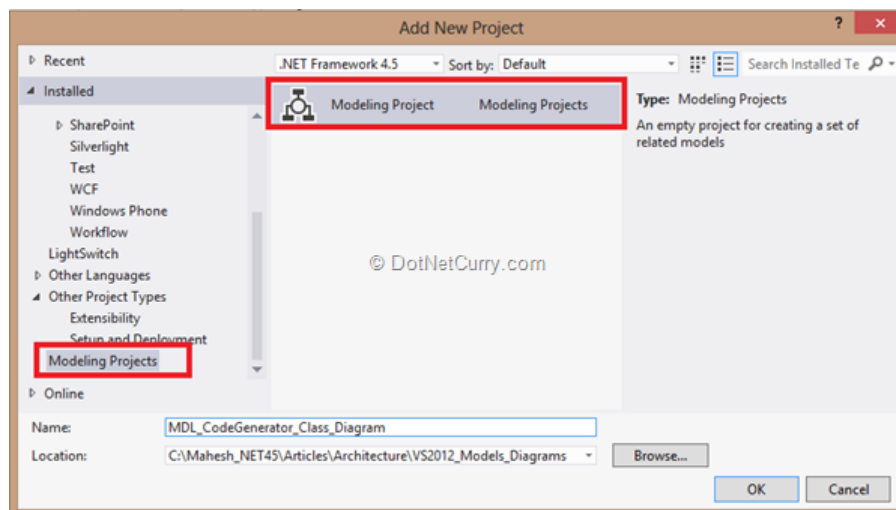
- Layer Diagram:

- Đây là Diagram quan trọng nhất trong thiết kế phần mềm.
- Thể hiện kiến trúc các tầng của hệ thống ở mức logic như: Data Access Layer, Business Logic Layer, UI Layer...
- Giúp kiểm tra về cấu trúc cũng như mối quan hệ giữa các tầng ứng dụng trong hệ thống sau khi coding.

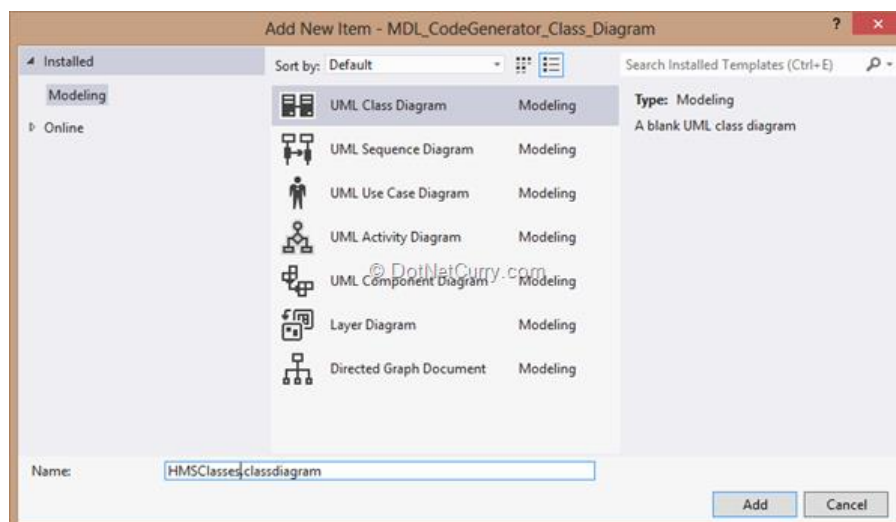
Class Diagram

Như đã đề cập ở trên, sơ đồ lớp (Class Diagram) cung cấp thông tin về các đối tượng được sử dụng trong hệ thống cũng như mối quan hệ giữa chúng. Mục đích chính của sơ đồ lớp là tập trung vào khía cạnh logic của các lớp mà không bị ảnh hưởng bởi các thể hiện của nó. Mối quan hệ giữa các đối tượng như thừa kế, kết hợp... được thể hiện một cách dễ dàng và rõ ràng thông qua sơ đồ lớp. Do đó, các kiến trúc sư hệ thống có thể sử dụng cùng IDE với lập trình viên như Visual Studio để thể hiện các thiết kế hệ thống.

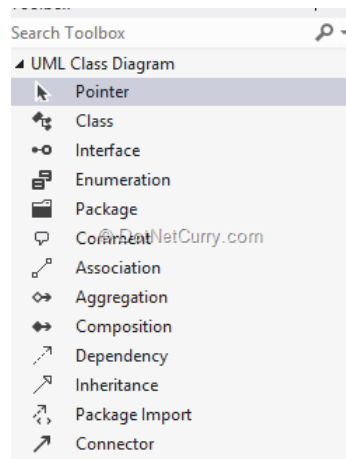
Bước 1: Mở Visual Studio và tạo một Solution mới, đặt tên là "VS2012_Models_Diagrams". Trong Solution này, click phải chuột và thêm vào một Modeling Project mới, đặt tên Project này là "MDL_CodeGenerator_Class_Diagram" như hình dưới đây:



Bước 2: Click phải vào Project mới tạo, chọn "Add New Item" -> "UML Class Diagram", đặt tên cho Diagram mới là "HMSClasses" như hình sau:

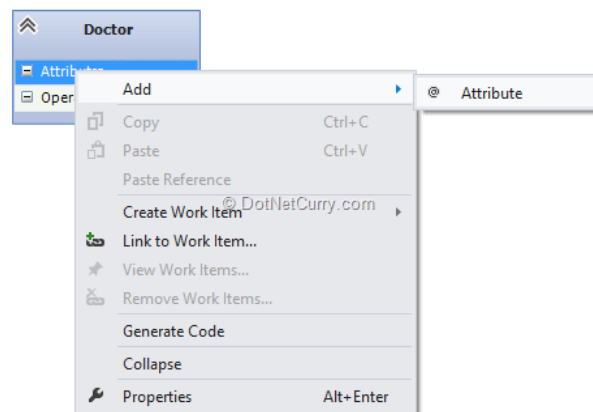


Khi đó, ta sẽ thấy một cửa sổ thiết kế Class Diagram và một thanh công cụ đi kèm:

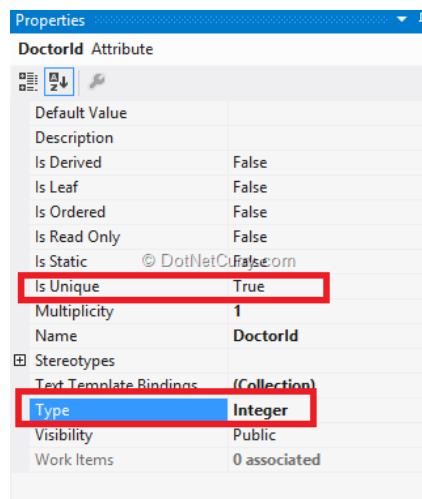


Thanh này cung cấp các công cụ thường được sử dụng trong thiết kế ứng dụng để thao tác với Class, Interface, Enum, Inheritance...

Bước 3: Kéo thả một lớp (Class) từ Toolbox vào màn hình thiết kế và đổi tên lớp thành "Doctor". Tại đây, ta có thể thêm các thuộc tính - Attributes (public properties) và các phương thức - Operations (methods) của lớp. Click phải chuột vào mục "Attributes" và chọn mục "Add Attribute" như hình dưới đây:



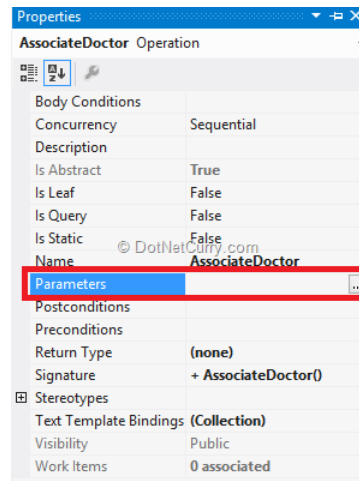
Thêm một thuộc tính mới tên là "DoctorId" và cấu hình thuộc tính này theo như hình sau:



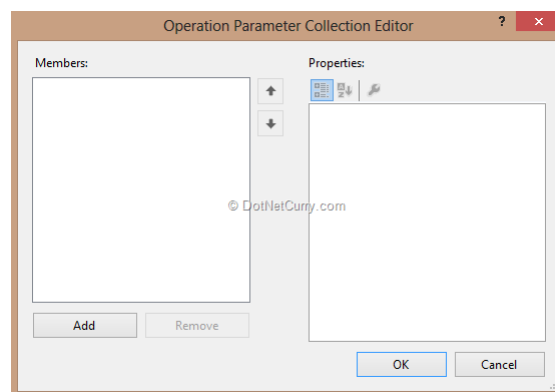
Ở đây, chúng ta đặt loại dữ liệu của thuộc tính bằng cách tùy chọn ở mục "Type". Ta có thể đặt cho thuộc tính là duy nhất thông qua mục "Is Unique".

Sau đó chúng ta tiếp tục thêm vào các thuộc tính cần thiết cho lớp tùy theo yêu cầu cụ thể.

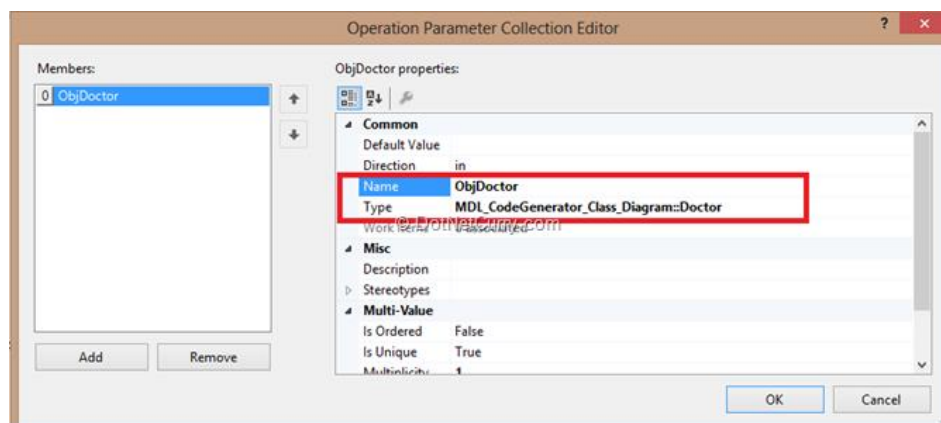
Để thêm vào các phương thức (Method), ta click phải chuột vào mục "Operation" và chọn "Add Operation". Tại đây, chúng ta có thể cấu hình các thông tin của phương thức bao gồm cả danh sách các tham số đầu vào và đầu ra như hình sau:



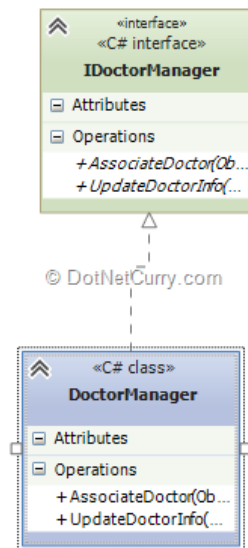
Để chỉ định các tham số đầu vào của phương thức, ta click vào dấu "..." ở mục "Parameters", khi đó cửa sổ sau sẽ xuất hiện:



Click vào nút "Add". Lúc này, chúng ta có thể chỉ định tên của tham số và loại dữ liệu của tham số theo như hình sau:

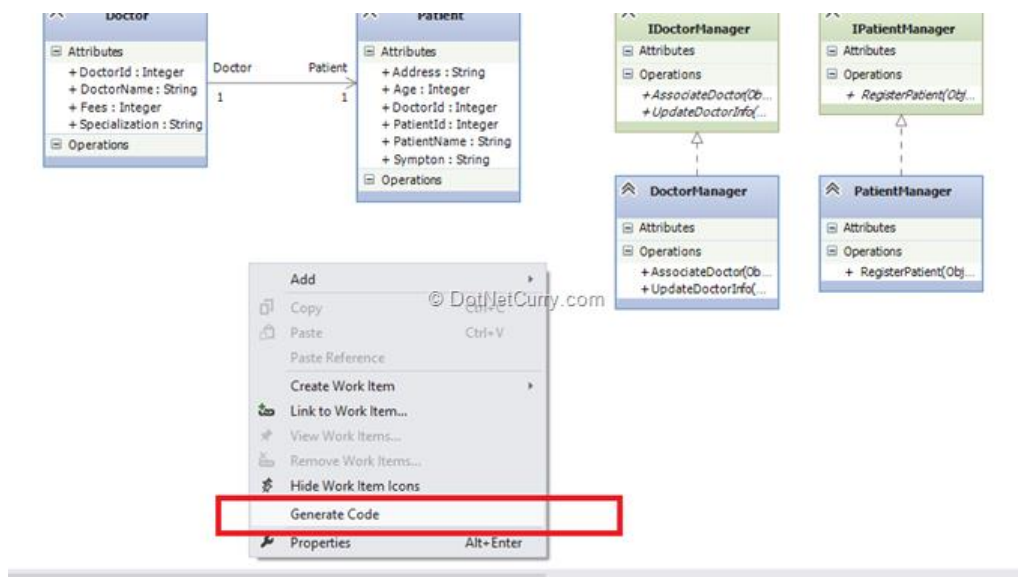


Tương tự như trên, ta có thể thêm vào các lớp (Class), các (giao diện) Interfaces và mối quan hệ giữa chúng. Ví dụ như một lớp kế thừa 1 giao diện, ta có thể sử dụng biểu tượng "Inheritance" trong thanh công cụ và kéo nó từ lớp sang Interface như hình sau:

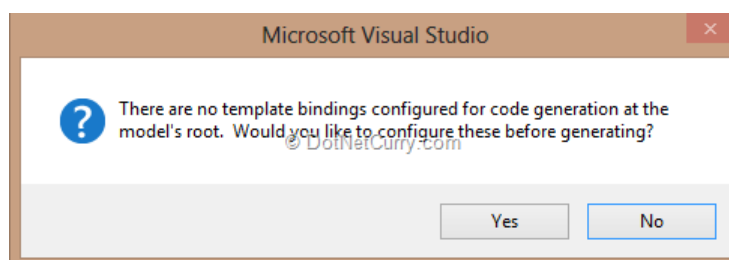


Sau khi hoàn tất, ta lưu lại các thiết kế này.

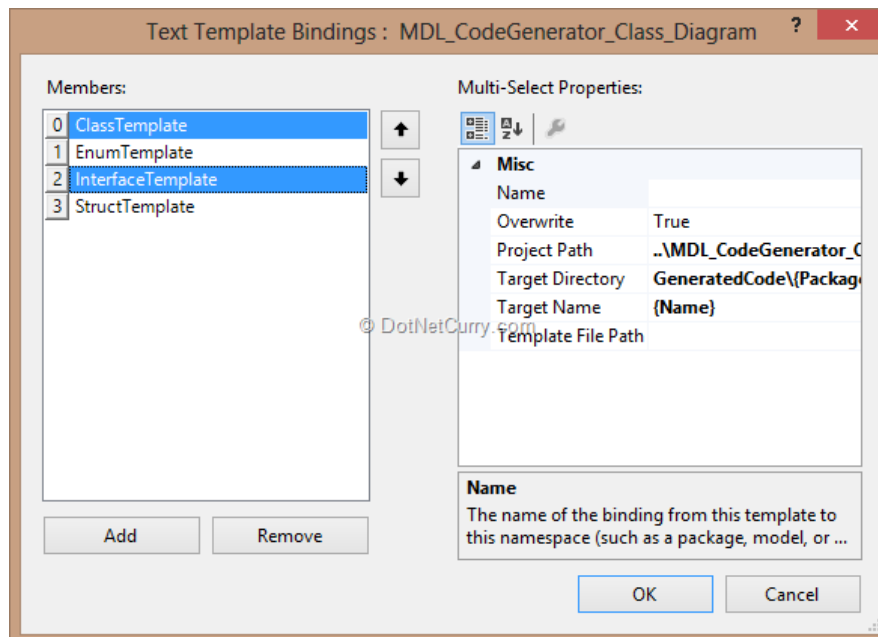
Bước 4: Click phải vào cửa sổ thiết kế Diagram của VS 2012 và chọn "Generate code" như hình sau:



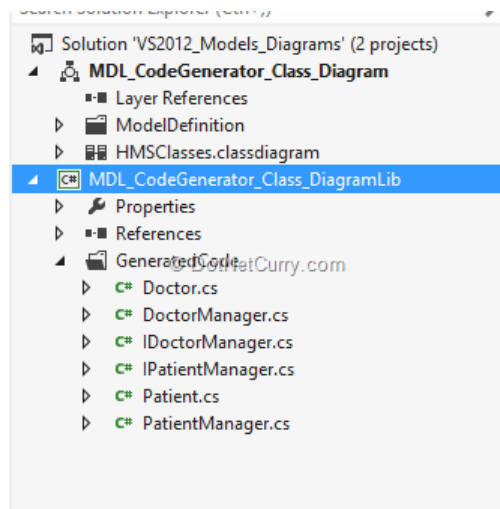
Tại đây, ta sẽ được hỏi về cấu hình "Template binding" như hình sau:



Do trong ví dụ này, chúng ta có sử dụng cả "Class Template" và "Interface Template", do đó ta click chọn "Yes" để tiếp tục, lúc này cửa sổ Template Bindings sẽ xuất hiện như sau:



Chọn "ClassTemplate" và "InterfaceTemplate", click chọn "OK". Lúc này, một Project mới sẽ được tự động thêm vào Solution với source code của tất cả các lớp (Class), giao diện (Interface) được thiết kế trong ứng dụng.



Kết luận:

Bộ công cụ vẽ sơ đồ UML và công cụ tự động sản sinh Source Code giúp các kiến trúc sư hệ thống đưa ra bộ khung thiết kế chương trình cho cả team, giúp hạn chế những xung đột trong thiết kế cũng như trong triển khai phần mềm.

Bài tập thực hành:

Phân tích, thiết kế sơ đồ lớp (UML Class Diagram) bao gồm các Class và Interface cần thiết, sản sinh source code tự động để tạo khung chương trình cho trò chơi Pacman.