**HƯỚNG DẪN SỬ TOOL GENCODE**

# Thông tin chung

* Tool gen code giúp bạn có thể tạo một chức năng với đầy đủ chức năng cơ bản như thêm mới, sửa, xóa, tìm kiếm, import excel, export excel
* Địa chỉ: [**http://192.168.1.100:8793**](http://192.168.1.100:8793)
* Phạm vi áp dụng: Các project sử dụng framework với CSDL Sql server

# Các bước thực hiện tạo một module mới

## Tạo file excel import hệ thống

* Truy cập dự án cần gencode

Graphical user interface, text, application, email

Description automatically generated

* Chọn Import collection

Graphical user interface, application

Description automatically generated

* Tại màn hình thực hiện tải mẫu excel import bằng cách chọn nút “Tải xuống”
* File excel mẫu sẽ có dạng như sau:

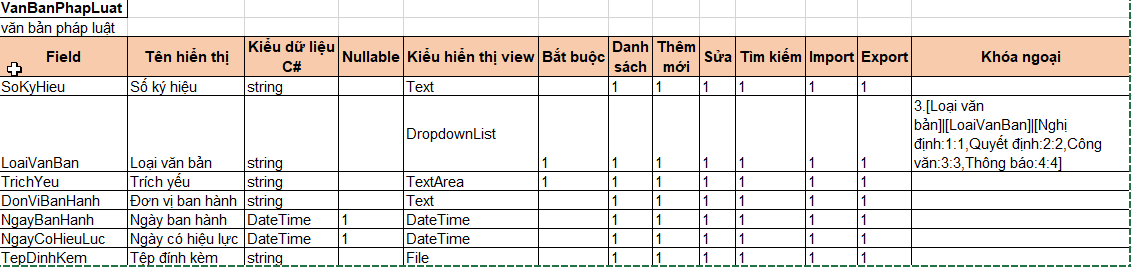
Table

Description automatically generated

* Vị trí số 1: Sẽ nhập tên bảng VD: HocSinh, SinhVien, NguoiDung. Nên đặt tên theo quy tắc Pascal (Viết hoa các chữ cái đầu) và đặc biệt tránh sử dụng dấu gạch dưới, kết thúc bằng các chữ cái hoa, kết thúc hoặc bắt đầu bằng tên các bảng khác.
* Vị trí số 2: sẽ nhập tên hiển thị của bảng. Tên này sử dụng để hiển thị cho người dùng cuối do vậy nên viết tiếng việt có dấu với ngữ nghĩa rõ ràng VD thông tin người dùng hoặc thông tin học sinh
* Vị trí số 3: là khu vực định nghĩa các trường thông tin trong bảng. Mỗi dòng là 1 trường thông tin.
* Ý nghĩa của các cột như sau:

|  |  |  |
| --- | --- | --- |
| **STT** | **Cột** | **Ý nghĩa** |
| **1** | **Field** | Tên của tường thông tin. Viết liền không dấu. Nên viết theo định dạng Pascal  VD: HoTen, NgaySinh, GioiTinh |
| **2** | **Tên hiển thị** | Tên hiển thị của trường thông tin của bảng. Nội dung này được hiển thị trên form khi người dùng kê khai thông tin và trên bảng danh sách.  Nên: Viết tiếng việt có dấu, ngắn gọn, ý nghĩa tường minh  VD: Họ và tên, Ngày sinh |
| **3** | **Kiểu dữ liệu C#** | Kiểu dữ liệu của trường trong ngôn ngữ C# như (string, int, long,…) File excel được thiết kế với ràng buộc dữ liệu chọn. Do đó nên copy từ dòng đầu tiên ra để giữ lại được format  Trường hợp gõ tay thì có thể xem ở sheet 2 “DM” để biết tất cả các giá trị điền |
| **4** | **Nullable** | Cột này xác định là kiểu dữ liệu có cho phép null hay không VD Kiểu dữ liệu ngày sinh nhưng không phải lúc nào người dùng nhập vào cũng có đầy đủ thông tin ngày sinh. Chúng ta nên để kiểu dữ liệu là DateTime?. Trường hợp này thì kiểu dữ liệu C# chúng ta chọn là DateTime. Tại cột Nullable chúng ta để số “1” để hệ thống hiểu rằng phải thiết lập kiểu dữ liệu cho phép null |
| **5** | **Kiểu hiển thị view** | Tương tự kiểu dữ liệu C# thì kiểu hiển thị View đã được thiết lập các tùy chọn. Có thể xem trong sheet 2. Hệ thống chỉ nhận các giá trị trong bảng này.  Đối với kiểu hiển thị các lựa chọn có ý nghĩa như sau:   * Text: Hiển thị ô text bình thường * Nummeric: Hiển thị ô nhập có áp dụng Autonumeric để có thể giúp người dùng nhìn các dấu phân cách hàng nghìn vd 100.125 * Editor: HIển thị khung soạn thảo có áp dụng editor * Radio: Hiển thị nút radio (chức năng này chưa phát triển hoàn thiện) * CheckBox: Hiển thị check box (Yêu cầu kiểu dữ liệu là bool) * DateTime: Hiển thị ô nhập có áp dụng datepicker * Number: Nhập kiểu số bình thường * Hidden: Hidden input * TextArea: Hiển thị textarea bình thường * File: Hiển thị kiểu nhập là File * CheckList: Hiển thị check list (chức năng này chưa hoàn chỉnh) * DropdownList: Kiểu này sẽ hiển thị dropdownlist để người dùng có thể chọn từ một tập dữ liệu có sẵn. Người dùng có thể thiết lập dữ liệu hiển thị từ một bảng, 1 nhóm trong danh mục dùng chung hoặc từ 1 class Constant. Chi tiết cách thiết lập sẽ được trình bày ở dưới |
| **6** | **Bắt buộc** | Nếu trường thông tin bắt buộc nhập để số 1 ngược lại thì để trống |
| **7** | **Danh sách** | Nếu để số 1 thì trường thông tin được hiển thị trên bảng danh sách |
| **8** | **Thêm mới** | Nếu để số 1 thì trường thông tin được hiển thị trên form thêm mới |
| **9** | **Sửa** | Nếu để số 1 thì trường thông tin được hiển thị trên form sửa |
| **10** | **Tìm kiếm** | Nếu để số 1 thì trường thông tin được hiển thị trên form tìm kiếm |
| **11** | **Import** | Nếu để số 1 thì trường thông tin được import từ file excel |
| **12** | **Export** | Nếu để số 1 thì trường thông tin được export ra file excel |
| **13** | **Khóa ngoại** | Cột này sử dụng để cấu hình khi kiểu hiển thị view là dropdownlist   |  |  | | --- | --- | | Cấu hình khóa ngoại | Diễn giải | | 1.[TenBang]|[value]|[display] | Tham chiếu đến bản khác | | 2.[TenConstant]|[type]|[n1:val1,n2:val2] | Tham chiếu đến 1 class Constant | | 3.[TenDM]|[MaDM]|[n1:val1,n2:val2] | Tham chiếu đến danh mục động trên hệ thống |   Có thể sử dụng chức năng “Công cụ” trên tool để hỗ trợ tạo nhanh các danh mục |

* VD về một file import excel



* Sau đó import file excel đã làm vào hệ thống
* Kết quả thu được như sau:

Graphical user interface, text, application, email

Description automatically generated

## Tạo code

* Ấn nút **Tạo code** màu xanh lá cây để tạo
* Hệ thống chuyển sang màn thiết lập trước khi gencode như sau

Graphical user interface, text, application, email

Description automatically generated

* Trước tiên để sau khi gen code không cần phải truy cập phần mềm để tạo thao tác sau đó phân quyền thì chúng ta có thể thiết lập trên tool như sau:
  + Ấn nút thêm mới chỗ nhóm chức năng để chèn thêm module vào CSDL của hệ thống

Graphical user interface

Description automatically generated with low confidence

* + Sau khi thêm mới thì load lại trang và chọn nhóm chức năng là chức năng mình đang gen code

Graphical user interface, application

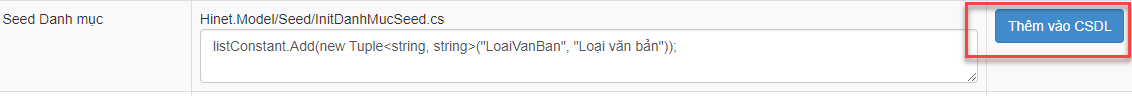
Description automatically generated

* + Tiếp theo gán các thao tác vào nhóm chức năng bằng cách ấn lần lượt các nút sau

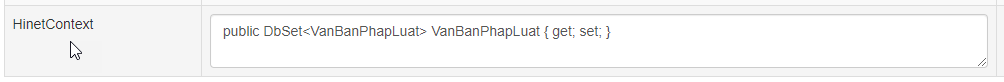
Graphical user interface, text, application, email

Description automatically generated

* + Trường hợp có bảng có thiết lập danh mục dùng chung thì chúng ta cũng có thể thêm danh sách giá trị vào CSDL bằng nút trên tool gen code



* + Thêm các cấu hình vào Solution thực hiện lần lượt như sau
  + Thêm thiết lập trong HinetContext

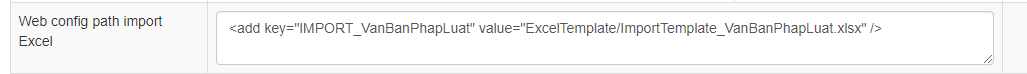


* + Copy nội dung vào file HinetContext.cs

Graphical user interface, text, application, email

Description automatically generated

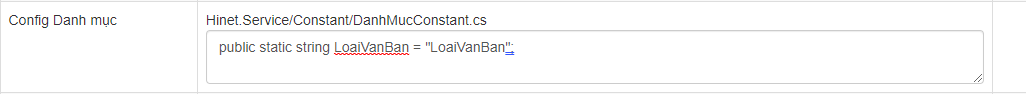
* + Copy cấu hình mẫu import excel vào web.config



Text

Description automatically generated

* + Trường hợp trong cấu hình có sử dụng danh mục dùng chung mới. Chúng ta thêm giá trị và DanhMucConstant để dễ sử dụng



* Tiếp đó kích nút **Tạo code** để hệ thống tải về code



* Và tải xuống

Graphical user interface, text, application

Description automatically generated

## Copy code vào project

* Sau khi tải về và giải nén ra. Chúng ta được file và folder như sau

Graphical user interface, text, application

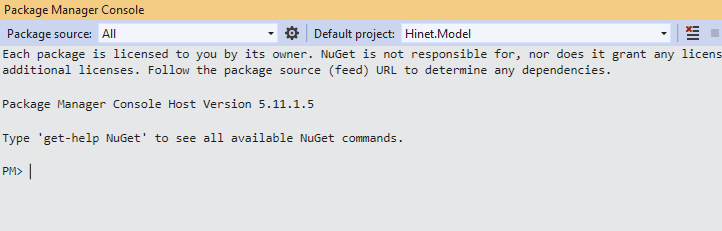
Description automatically generated

* Thực hiện copy lần lượt vào trong Visual Studio

## Migration dữ liệu

Fw đang sử dụng cách tiếp cận Code first để kết nối với database. Do đó sau khi chúng ta tạo code thì cần thực hiện một số lệnh sau để tạo bảng trong CSDL

Mở công cụ command trên visual studio



Lưu ý default project cần chọn: Hinet.Model

Sau đó gõ các lệnh sau:

* add-migration [messsage]. Lệnh này dùng để tạo migration cho bảng mới tạo. [messsage] là nội dung tên file migration để gợi nhớ
* update-database. Lệnh này dùng để cập nhật migration lên CSDL
* OK. Giờ thì run thôi.

## Kiểm tra code gen

* Chạy project lên. Truy cập với tài khoản admin

