Hướng dẫn phát triển module trên DNN 7.1.2 sử dụng DAL2 Compiled C# template

***Tài liệu này là một phần bản nháp của tài liệu giải pháp phát triển, mở rộng chức năng hệ thống đề tài Công sở Điện tử***

Mặc định người phát triển đã cài DotNetNuke Module Development Project Templates cảu christoc’s download tại link <https://christoctemplate.codeplex.com/releases/view/114819>

Và đã cài template tùy biến của HueCIT.

Tham khảo tài liệu giới thiệu về lập trình truy xuất dữ liệu sử dụng DAL2 tại link

<http://www.dnnsoftware.com/community-blog/cid/147417/dal-2--repository-deep-dive-1>

Mục lục

[I. Hướng dẫn viết module truy vấn dữ liệu và phân trang. 2](#_Toc386027011)

[1. Cấu trúc thư mục mã nguồn của module như hình sau 3](#_Toc386027012)

[2. Lập trình cho module 4](#_Toc386027013)

[a. Lập trình lớp ItemController 4](#_Toc386027014)

[b. Lập trình cấu hình cho module 7](#_Toc386027015)

[c. Lập trình lớp MyNewsModuleBase 12](#_Toc386027016)

[d. Lập trình hiển thị các records 14](#_Toc386027017)

[e. Lập trình để hiển thị các records có phân trang ở control View.ascx 17](#_Toc386027018)

[II. Tạo module hỗ trợ multiple view NewsMultipleView, khi kích link bài viết thì chi tiết bài viết được hiển thị 21](#_Toc386027019)

[1. Lập trình lớp ItemController 21](#_Toc386027020)

[2. Lập trình phần cấu hình cho module 24](#_Toc386027021)

[a. Giao diện Settings.ascx 24](#_Toc386027022)

[b. Code behind Settings.ascx.cs 24](#_Toc386027023)

[3. Lập trình lớp ModuleBase NewsMultipleViewModuleBase 27](#_Toc386027024)

[4. Lập trình tạo các view dùng để hiển thị dữ liệu 29](#_Toc386027025)

[4.1. Thư mục ListControls lưu các usercontrol dùng để trình bày danh sách các records 30](#_Toc386027026)

[4.2. DetailControls để lưu các usercontrol dùng để trình bày chi tiết của một record 38](#_Toc386027027)

[5. Lập trình chọn view và hiển thị dữ liệu ở control View.ascx 39](#_Toc386027028)

[a. Giao diện control View.ascx 39](#_Toc386027029)

[b. Hàm Page\_Load ở code behind của control View.ascx.cs 40](#_Toc386027030)

[6. Biên dịch và cài đặt, chạy thử 40](#_Toc386027031)

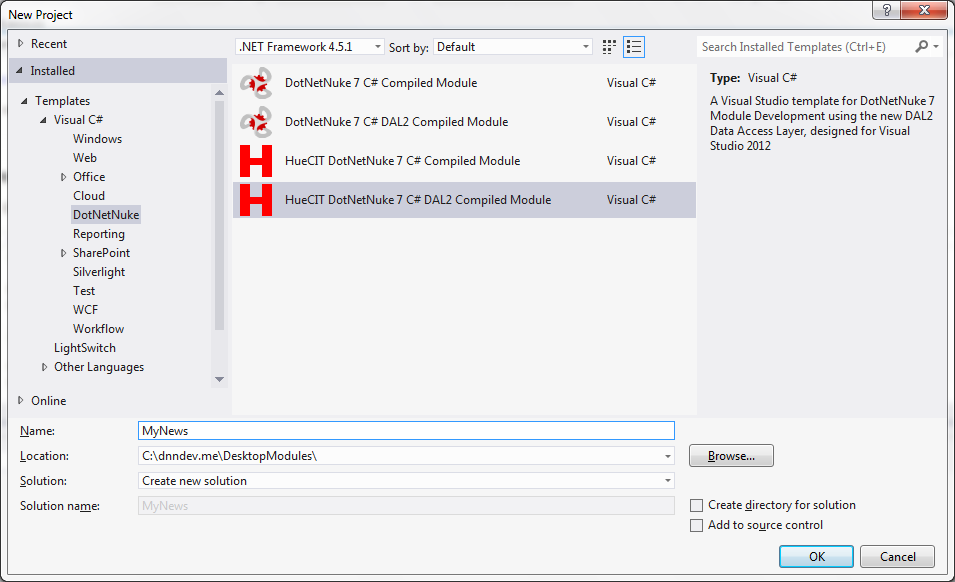
[a. Giao diện module khi chọn view trình bày danh sách theo dạng liệt kê dùng thẻ ul 40](#_Toc386027032)

[b. Giao diên module khi chọn view trình bày danh sách theo dạng bảng dùng thẻ table 41](#_Toc386027033)

# I. Hướng dẫn viết module truy vấn dữ liệu và phân trang.

Mở visual studio 2012 chạy ở chế độ Run As Administrator

Tạo mới project



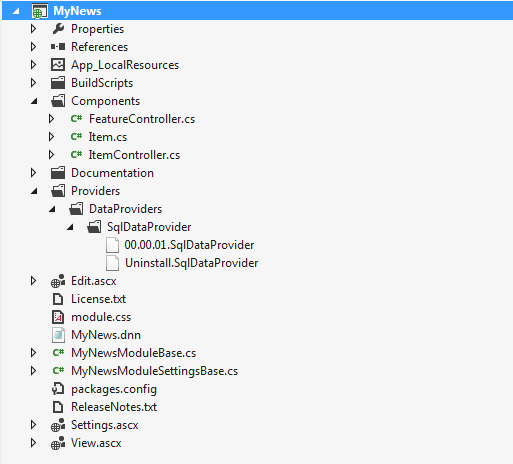
Đặt tên project là MyNews

Chọn template là HueCIT Dotnetnuke 7 C# DAL2 Compiled Module

Chọn vị trí lưu trữ module là thư mục DesktopModules của bản cài môi trường phát triển Dotnetnuke

Kích ok.

## 1. Cấu trúc thư mục mã nguồn của module như hình sau



* User control Edit.ascx dùng để thêm mới, hiệu chỉnh record
* User control Setting.ascx dùng để thiết lập cấu hình cho module
* User control View.ascx dùng để hiển thị danh sách records
* Lớp MyNewsModuleBase.cs kế thừa từ lớp PortalModuleBase cung cấp các thuộc tính, phương thức dùng dùng cho module
* Lớp MyNewsModuleSettingsBase kế thừa từ lớp MyNewsModuleSettingsBase cung cấp các thuộc tính, phương thức dùng dùng để thiết lập cấu hình cho module
* File MyNews.dnn lưu thông tin định nghĩa về module
* Thư mục Providers lưu các file script cài đặt các bảng để lưu dữ liệu cho module
* Thư mục Component lưu các lớp thực thể dữ liệu của module và các lớp nghiệp vụ thao tác dữ liệu

## 2. Lập trình cho module

### a. Lập trình lớp ItemController

Các hàm cơ bản mặc định của lớp ItemController

ĐỊnh nghĩa các phương thức thao tác dữ liệu sử dụng petapoco

/\*

' Copyright (c) 2014 huecit.vn

' All rights reserved.

'

' THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

' TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

' THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

' CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

' DEALINGS IN THE SOFTWARE.

'

\*/

using System.Collections.Generic;

using DotNetNuke.Data;

namespace HueCIT.Modules.MyNews.Components

{

class ItemController

{

public void CreateItem(Item t)

{

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

rep.Insert(t);

}

}

public void DeleteItem(int itemId, int moduleId)

{

var t = GetItem(itemId, moduleId);

DeleteItem(t);

}

public void DeleteItem(Item t)

{

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

rep.Delete(t);

}

}

public IEnumerable<Item> GetItems(int moduleId)

{

IEnumerable<Item> t;

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

t = rep.Get(moduleId);

}

return t;

}

public Item GetItem(int itemId, int moduleId)

{

Item t;

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

t = rep.GetById(itemId, moduleId);

}

return t;

}

public void UpdateItem(Item t)

{

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

rep.Update(t);

}

}

}

}

Khi mới tạo module lớp này cung cấp sẵn một số hàm cơ bản

* void CreateItem(Item t) Thêm mới item vào csdl
* void DeleteItem(int itemId, int moduleId) Xóa item có idemID, thuộc module có moduleId khỏi csdl
* void DeleteItem(Item t) xóa đối tượng item khỏi csdl
* IEnumerable<Item> GetItems(int moduleId) trả về danh sách các items thuộc module có moduleId
* Item GetItem(int itemId, int moduleId) trả về item có itemnId thuộc module có moduleId
* void UpdateItem(Item t) cập nhật một item

Truy vấn dữ liệu phân trang vào lớp ItemController

Bên cạnh đó tuy theo yêu cầu bài toán chúng ta có thể bổ sung các hàm phục vụ yêu cầu nghiệp vụ của bài toán chúng ta cần.

Ví dụ, thường số lượng bài viết rất lớn nếu đọc tất cả các bài viết thì sẽ rất chậm và đối khi người dùng cũng không đọc hết => chúng ta muốn phần trang để hiển thị mỗi lần chỉ một số bản ghi (pageSize) để hiển thị và hiển thị liên kết phân trang để người dùng chọn trang muốn xem bài viết.

Để đáp dứng yêu cầu trên chung ta bổ sung hàm để truy vấn danh các các bảng ghi của một trang nào đó. Đoạn code dưới đây là 2 hàm truy vấn dữ liệu phân trang

* GetItems(int pageIndex, int pageSize) trả về dữ liệu của trang có chỉ mục trang là pageIndex, với số bảng ghi trên một trang là pageSize
* IEnumerable<Item> GetItems(int moduleId, int pageIndex, int pageSize) trả về dữ liệu của trang thuộc modulemoduleId, có chỉ mục trang là pageIndex, với số bảng ghi trên một trang là pageSize

//public IEnumerable<Item> GetItems(int pageIndex, int pageSize)

public IPagedList<Item> GetItems(int pageIndex, int pageSize)

{

//IEnumerable<Item> t;

IPagedList<Item> t;

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

t = rep.GetPage(pageIndex, pageSize);

// truong hop phan trang kem dieu khien loc du lieu

//t = rep.Find(pageIndex, pageSize, "WHERE ItemName like N'%@0%' ORDER BY ItemName", "My");

}

return t;

}

//public IEnumerable<Item> GetItems(int moduleId, int pageIndex, int pageSize)

public IPagedList<Item> GetItems(int moduleId, int pageIndex, int pageSize)

{

//IEnumerable<Item> t;

IPagedList<Item> t;

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

t = rep.GetPage(moduleId, pageIndex, pageSize);

// truong hop phan trang kem dieu khien loc du lieu

//t = rep.Find(pageIndex, pageSize, "WHERE ModuleId = @0 AND ItemName like @1 ORDER BY ItemName", moduleId,"%My%");

}

return t;

}

### b. Lập trình cấu hình cho module

Cấu hình cho module được lập trình ở usercontrol Settings.ascx

Ví dụ chúng ta muốn cấu hình số bảng ghi sẽ hiện thị chỗ một trang nôi dung, chúng ta sẽ lập trình ở control Settings.ascx

Mặc định khi mới tạo module control Settings.ascx có mã nguồn như sau

<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="Settings.ascx.cs" Inherits="HueCIT.Modules.MyNews.Settings" %>

<!-- uncomment the code below to start using the DNN Form pattern to create and update settings -->

<%--

<%@ Register TagName="label" TagPrefix="dnn" Src="~/controls/labelcontrol.ascx" %>

<h2 id="dnnSitePanel-BasicSettings" class="dnnFormSectionHead"><a href="" class="dnnSectionExpanded"><%=LocalizeString("BasicSettings")%></a></h2>

<fieldset>

<div class="dnnFormItem">

<dnn:Label ID="lblSetting1" runat="server" />

<asp:TextBox ID="txtSetting1" runat="server" />

</div>

<div class="dnnFormItem">

<dnn:label ID="lblSetting2" runat="server" />

<asp:TextBox ID="txtSetting2" runat="server" />

</div>

</fieldset>

--%>

Khai báo giao diện để cho phép người dùng nhập số bảng ghi sẽ hiện thị chỗ một trang

<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="Settings.ascx.cs" Inherits="HueCIT.Modules.MyNews.Settings" %>

<!-- uncomment the code below to start using the DNN Form pattern to create and update settings -->

<%@ Register TagName="label" TagPrefix="dnn" Src="~/controls/labelcontrol.ascx" %>

<h2 id="dnnSitePanel-BasicSettings" class="dnnFormSectionHead"><a href="" class="dnnSectionExpanded"><%=LocalizeString("BasicSettings")%></a></h2>

<fieldset>

<div class="dnnFormItem">

<dnn:Label ID="lblPageSize" runat="server" />

<asp:TextBox ID="txtPageSize" runat="server" />

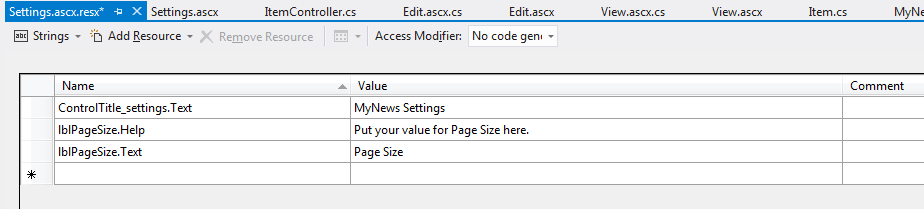
</div>

</fieldset>

Chú ý nhằm nâng cao tính mềm dẽo cho module ví dụ hỗ trợ đa ngữ thì text cho các nhãn trên form không được thiết kế cứng mà được lưu ở file resources có tên Settings.ascx.resx

Ví dụ text cho nhãn lblPageSize là Page Size

Mở file Settings.ascx.resx để soạn nội dung cho các nhãn như hình bên dưới



Lập trình để lưu thông tin cấu hình page size khi người dùng cài vào size và cấu hình

Mở file Settings.ascx.cs

Ban đầu mặc định module chưa có cấu hình gì nên code file này như sau

/\*

' Copyright (c) 2014 huecit.vn

' All rights reserved.

'

' THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

' TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

' THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

' CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

' DEALINGS IN THE SOFTWARE.

'

\*/

using System;

using DotNetNuke.Entities.Modules;

using DotNetNuke.Services.Exceptions;

namespace HueCIT.Modules.MyNews

{

/// -----------------------------------------------------------------------------

/// <summary>

/// The Settings class manages Module Settings

///

/// Typically your settings control would be used to manage settings for your module.

/// There are two types of settings, ModuleSettings, and TabModuleSettings.

///

/// ModuleSettings apply to all "copies" of a module on a site, no matter which page the module is on.

///

/// TabModuleSettings apply only to the current module on the current page, if you copy that module to

/// another page the settings are not transferred.

///

/// If you happen to save both TabModuleSettings and ModuleSettings, TabModuleSettings overrides ModuleSettings.

///

/// Below we have some examples of how to access these settings but you will need to uncomment to use.

///

/// Because the control inherits from MyNewsSettingsBase you have access to any custom properties

/// defined there, as well as properties from DNN such as PortalId, ModuleId, TabId, UserId and many more.

/// </summary>

/// -----------------------------------------------------------------------------

public partial class Settings : MyNewsModuleSettingsBase

{

#region Base Method Implementations

/// -----------------------------------------------------------------------------

/// <summary>

/// LoadSettings loads the settings from the Database and displays them

/// </summary>

/// -----------------------------------------------------------------------------

public override void LoadSettings()

{

try

{

if (Page.IsPostBack == false)

{

//Check for existing settings and use those on this page

//Settings["SettingName"]

/\* uncomment to load saved settings in the text boxes

if(Settings.Contains("Setting1"))

txtSetting1.Text = Settings["Setting1"].ToString();

if (Settings.Contains("Setting2"))

txtSetting2.Text = Settings["Setting2"].ToString();

\*/

}

}

catch (Exception exc) //Module failed to load

{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

/// -----------------------------------------------------------------------------

/// <summary>

/// UpdateSettings saves the modified settings to the Database

/// </summary>

/// -----------------------------------------------------------------------------

public override void UpdateSettings()

{

try

{

var modules = new ModuleController();

//the following are two sample Module Settings, using the text boxes that are commented out in the ASCX file.

//module settings

//modules.UpdateModuleSetting(ModuleId, "Setting1", txtSetting1.Text);

//modules.UpdateModuleSetting(ModuleId, "Setting2", txtSetting2.Text);

//tab module settings

//modules.UpdateTabModuleSetting(TabModuleId, "Setting1", txtSetting1.Text);

//modules.UpdateTabModuleSetting(TabModuleId, "Setting2", txtSetting2.Text);

}

catch (Exception exc) //Module failed to load

{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

#endregion

}

}

Sửa code ở hàm LoadSettings để đọc giá trị lưu thông tin cấu hình page size và hiển thị lên textbox txtPageSize ở form settings nếu có

/// -----------------------------------------------------------------------------

/// <summary>

/// LoadSettings loads the settings from the Database and displays them

/// </summary>

/// -----------------------------------------------------------------------------

public override void LoadSettings()

{

try

{

if (Page.IsPostBack == false)

{

//Check for existing settings and use those on this page

//Settings["SettingName"]

if(Settings.Contains("PageSize"))

txtPageSize.Text = Settings["PageSize"].ToString();

}

}

catch (Exception exc) //Module failed to load

{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

Lập trình lấy giá trị thông tin cấu hình page size từ form settings và lưu vào csdl

/// -----------------------------------------------------------------------------

/// <summary>

/// UpdateSettings saves the modified settings to the Database

/// </summary>

/// -----------------------------------------------------------------------------

public override void UpdateSettings()

{

try

{

var modules = new ModuleController();

//the following are two sample Module Settings, using the text boxes that are commented out in the ASCX file.

//module settings

modules.UpdateModuleSetting(ModuleId, "PageSize", txtPageSize.Text);

}

catch (Exception exc) //Module failed to load

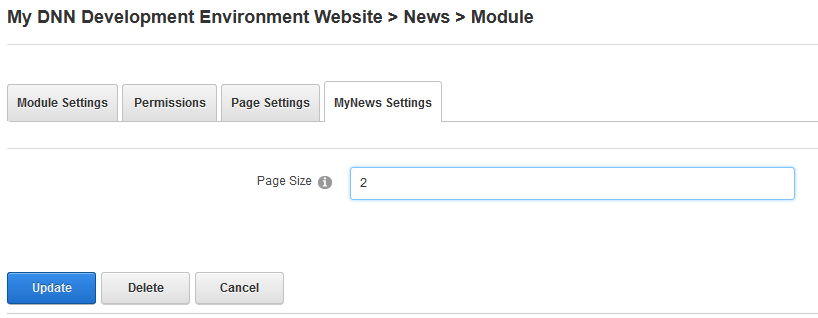
{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

Khi cài module vào site, và chọn Settings form settings sẽ như hình bên dưới



Nhập Page Size và Update

### c. Lập trình lớp MyNewsModuleBase

Bổ sung các thuộc tính PageSize và PageIndex vào lớp ModuleBase (MyNewsModuleBase) vào của module để lấy các giá trị pageSize và pageIndex

Ban đầu lớp MyNewsModuleBase như sau

/\*

' Copyright (c) 2014 huecit.vn

' All rights reserved.

'

' THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

' TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

' THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

' CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

' DEALINGS IN THE SOFTWARE.

'

\*/

using System;

using DotNetNuke.Entities.Modules;

namespace HueCIT.Modules.MyNews

{

public class MyNewsModuleBase : PortalModuleBase

{

public int ItemId

{

get

{

var qs = Request.QueryString["tid"];

if (qs != null)

return Convert.ToInt32(qs);

return -1;

}

}

}

}

Thuốc tính ItemID trả về id của record người sử dụng đang chọn để xem chi tiết

Lớp MyNewsModuleBase sau khi bổ sung 2 thuộc tính PageSize và PageIndex

/\*

' Copyright (c) 2014 huecit.vn

' All rights reserved.

'

' THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

' TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

' THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

' CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

' DEALINGS IN THE SOFTWARE.

'

\*/

using System;

using DotNetNuke.Entities.Modules;

namespace HueCIT.Modules.MyNews

{

public class MyNewsModuleBase : PortalModuleBase

{

public int ItemId

{

get

{

var qs = Request.QueryString["tid"];

if (qs != null)

return Convert.ToInt32(qs);

return -1;

}

}

public int PageIndex

{

get

{

var qs = Request.QueryString["p"]; // p: standard for page

if (qs != null)

return Convert.ToInt32(qs);

return 0;

}

}

public int PageSize

{

get

{

if (Settings.Contains("PageSize"))

return Convert.ToInt32(Settings["PageSize"]);

return 2; // default a page have two records for test

}

set

{

var mc = new ModuleController();

mc.UpdateModuleSetting(ModuleId, "PageSize", value.ToString());

}

}

}

}

### d. Lập trình hiển thị các records

Nội dung các records của module được hiển thị ở control View.ascx, mã nguồn giao diện control View.ascx

<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="View.ascx.cs" Inherits="HueCIT.Modules.MyNews.View" %>

<asp:Repeater ID="rptItemList" runat="server" OnItemDataBound="rptItemListOnItemDataBound" OnItemCommand="rptItemListOnItemCommand">

<HeaderTemplate>

<ul class="tm\_tl">

</HeaderTemplate>

<ItemTemplate>

<li class="tm\_t">

<h3>

<asp:Label ID="lblitemName" runat="server" Text='<%#DataBinder.Eval(Container.DataItem,"ItemName").ToString() %>' />

</h3>

<asp:Label ID="lblItemDescription" runat="server" Text='<%#DataBinder.Eval(Container.DataItem,"ItemDescription").ToString() %>' CssClass="tm\_td" />

<asp:Panel ID="pnlAdmin" runat="server" Visible="false">

<asp:HyperLink ID="lnkEdit" runat="server" ResourceKey="EditItem.Text" Visible="false" Enabled="false" />

<asp:LinkButton ID="lnkDelete" runat="server" ResourceKey="DeleteItem.Text" Visible="false" Enabled="false" CommandName="Delete" />

</asp:Panel>

</li>

</ItemTemplate>

<FooterTemplate>

</ul>

</FooterTemplate>

</asp:Repeater>

Mặc định khi module được lập trình để hiển thị tất cả các bảng ghi

Mã nguồn control View.ascx.cs

/\*

' Copyright (c) 2014 huecit.vn

' All rights reserved.

'

' THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

' TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

' THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

' CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

' DEALINGS IN THE SOFTWARE.

'

\*/

using System;

using System.Web.UI.WebControls;

using HueCIT.Modules.MyNews.Components;

using DotNetNuke.Security;

using DotNetNuke.Services.Exceptions;

using DotNetNuke.Entities.Modules;

using DotNetNuke.Entities.Modules.Actions;

using DotNetNuke.Services.Localization;

using DotNetNuke.UI.Utilities;

namespace HueCIT.Modules.MyNews

{

/// -----------------------------------------------------------------------------

/// <summary>

/// The View class displays the content

///

/// Typically your view control would be used to display content or functionality in your module.

///

/// View may be the only control you have in your project depending on the complexity of your module

///

/// Because the control inherits from MyNewsModuleBase you have access to any custom properties

/// defined there, as well as properties from DNN such as PortalId, ModuleId, TabId, UserId and many more.

///

/// </summary>

/// -----------------------------------------------------------------------------

public partial class View : MyNewsModuleBase, IActionable

{

protected void Page\_Load(object sender, EventArgs e)

{

try

{

var tc = new ItemController();

rptItemList.DataSource = tc.GetItems(ModuleId);

rptItemList.DataBind();

}

catch (Exception exc) //Module failed to load

{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

protected void rptItemListOnItemDataBound(object sender, RepeaterItemEventArgs e)

{

if (e.Item.ItemType == ListItemType.AlternatingItem || e.Item.ItemType == ListItemType.Item)

{

var lnkEdit = e.Item.FindControl("lnkEdit") as HyperLink;

var lnkDelete = e.Item.FindControl("lnkDelete") as LinkButton;

var pnlAdminControls = e.Item.FindControl("pnlAdmin") as Panel;

var t = (Item)e.Item.DataItem;

if (IsEditable && lnkDelete != null && lnkEdit != null && pnlAdminControls != null)

{

pnlAdminControls.Visible = true;

lnkDelete.CommandArgument = t.ItemId.ToString();

lnkDelete.Enabled = lnkDelete.Visible = lnkEdit.Enabled = lnkEdit.Visible = true;

lnkEdit.NavigateUrl = EditUrl(string.Empty, string.Empty, "Edit", "tid=" + t.ItemId);

ClientAPI.AddButtonConfirm(lnkDelete, Localization.GetString("ConfirmDelete", LocalResourceFile));

}

else

{

pnlAdminControls.Visible = false;

}

}

}

public void rptItemListOnItemCommand(object source, RepeaterCommandEventArgs e)

{

if (e.CommandName == "Edit")

{

Response.Redirect(EditUrl(string.Empty, string.Empty, "Edit", "tid=" + e.CommandArgument));

}

if (e.CommandName == "Delete")

{

var tc = new ItemController();

tc.DeleteItem(Convert.ToInt32(e.CommandArgument), ModuleId);

}

Response.Redirect(DotNetNuke.Common.Globals.NavigateURL());

}

public ModuleActionCollection ModuleActions

{

get

{

var actions = new ModuleActionCollection

{

{

GetNextActionID(), Localization.GetString("EditModule", LocalResourceFile), "", "", "",

EditUrl(), false, SecurityAccessLevel.Edit, true, false

}

};

return actions;

}

}

}

}

Ở mã nguồn trên tất cả các records được hiển thị, dữ liệu chưa được phân trang

Để hiển thị dữ liệu có phần trang thì chỉ việc sửa code ở hàm Page\_Load để gọi đến hàm có hỗ trợ phần trang chúng ta đã bổ sung

* IPagedList<Item> GetItems(int pageIndex, int pageSize)
* IPagedList<Item> GetItems(int moduleId, int pageIndex, int pageSize)

Để đọc dữ liệu phân trang chúng ta cần chú ý 2 tham số quan trọng

* pageIndex: trang muốn xem dữ liệu, do người dùng chọn trang muốn xem từ thành điều hướng trang (pagenigation)
* pageSize: kích thước dữ liệu của mỗi trang, đọc từ settings

### e. Lập trình để hiển thị các records có phân trang ở control View.ascx

Hàm Page\_Load của control View.ascx hỗ trợ phân trang

protected void Page\_Load(object sender, EventArgs e)

{

try

{

var tc = new ItemController();

//rptItemList.DataSource = tc.GetItems(ModuleId);

rptItemList.DataSource = tc.GetItems(ModuleId, PageIndex, PageSize);

rptItemList.DataBind();

}

catch (Exception exc) //Module failed to load

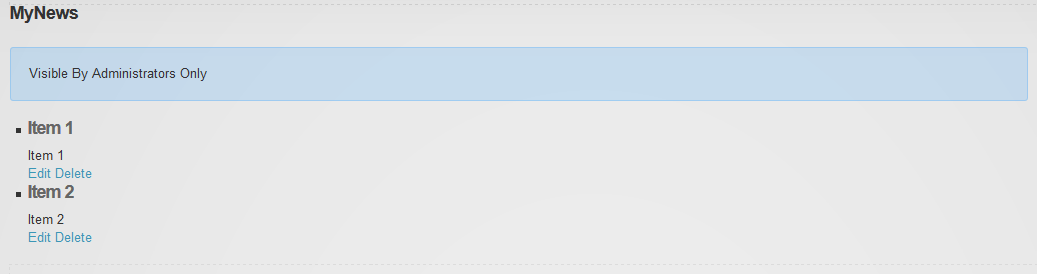
{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

Test xem kết quả phân trang bằng cách cài module vào site và thêm mới một số records ví dụ 6 records, settings page size = 2



Kết quả hiển thị 2 records, nhưng chưa có pagenigation

Lập trình hiển thị pagenigation để cho phép người dùng chọn trang để xem

Để hiển thị pagenigation trước hết phải biết tổng số records, số records trên mỗi trang (đọc từ file cấu hình)

Các hàm hỗ trợ phần trang trả về IPagedList<T>, Interface IPagedList cài đặt IList<T> và có thêm các thuộc tính như sau

namespace DotNetNuke.Collections

{

public interface IPagedList<T> : IList<T>, ICollection<T>, IEnumerable<T>, IEnumerable

{

bool HasNextPage { get; }

bool HasPreviousPage { get; }

bool IsFirstPage { get; }

bool IsLastPage { get; }

int PageCount { get; set; }

int PageIndex { get; set; }

int PageSize { get; set; }

int TotalCount { get; set; }

}

}

Thông qua ta sẽ biết được các thông tin như tổng số records, số trang…

Mở file control View.ascx để thêm các điều khiển dùng để hiển thị thông tin phân trang

<asp:Panel ID="pnlPaging" runat="server">

<asp:HyperLink ID="lnkPrevious" runat="server" resourcekey="lnkPrevious" Visible="false"

CssClass="lnkPrevious"></asp:HyperLink>

<asp:HyperLink ID="lnkNext" runat="server" resourcekey="lnkNext" Visible="false"

CssClass="lnkNext"></asp:HyperLink>

</asp:Panel>

Mở file View.ascx.cs bổ sung hàm lập trình hiển thị phân trang

private void BuildPageList(int pageCount)

{

NameValueCollection queryString = Request.QueryString;

SetPagingLink(queryString, lnkNext, PageIndex + 1 < pageCount, PageIndex + 1, TabId);

SetPagingLink(queryString, lnkPrevious, PageIndex - 1 > -1, PageIndex - 1, TabId);

}

private static void SetPagingLink(NameValueCollection queryString

, HyperLink link, bool showLink, int linkedPageId, int tabId)

{

if (showLink)

{

link.Visible = true;

queryString = new NameValueCollection(queryString);

queryString["p"] = linkedPageId.ToString(CultureInfo.InvariantCulture);

var additionalParameters = new List<string>(queryString.Count);

for (int i = 0; i < queryString.Count; i++)

{

if (string.Equals(queryString.GetKey(i), "TABID", StringComparison.OrdinalIgnoreCase))

{

int newTabId;

if (int.TryParse(queryString[i], NumberStyles.Integer, CultureInfo.InvariantCulture, out newTabId))

{

tabId = newTabId;

}

}

else if (!string.Equals(queryString.GetKey(i), "LANGUAGE", StringComparison.OrdinalIgnoreCase))

{

additionalParameters.Add(queryString.GetKey(i) + "=" + queryString[i]);

}

}

link.NavigateUrl = DotNetNuke.Common.Globals.NavigateURL(tabId, string.Empty, additionalParameters.ToArray());

}

else

{

link.Visible = false;

}

}

Sửa hàm Page\_Load để truy vấn dữ liệu và phân trang

protected void Page\_Load(object sender, EventArgs e)

{

try

{

var tc = new ItemController();

IPagedList<Item> items = tc.GetItems(ModuleId, PageIndex, PageSize);

rptItemList.DataSource = items;

rptItemList.DataBind();

BuildPageList(items.PageCount);

}

catch (Exception exc) //Module failed to load

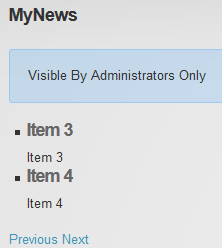
{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

Kết qủa



# II. Tạo module hỗ trợ multiple view NewsMultipleView, khi kích link bài viết thì chi tiết bài viết được hiển thị

Mở visual studio 2012 chạy ở chế độ Run As Administrator, tạo mới project đặt tên project là NewsMultipleView

Chọn template là HueCIT Dotnetnuke 7 C# DAL2 Compiled Module

Chọn vị trí lưu trữ module là thư mục DesktopModules của bản cài môi trường phát triển Dotnetnuke

Module NewsMultipleView dự định hỗ trợ các tính năng sau

* Cho phép lựa chọn hiển thị danh sách bài viết ở nhiều hình thức khác nhau
* Hiển thị thông tin chi tiết của một bài viết ở nhiều hình thức khác nhau

Các công việc cần thực hiện

## 1. Lập trình lớp ItemController

/\*

' Copyright (c) 2014 huecit.vn

' All rights reserved.

'

' THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

' TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

' THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

' CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

' DEALINGS IN THE SOFTWARE.

'

\*/

using System.Collections.Generic;

using DotNetNuke.Data;

using DotNetNuke.Collections;

namespace HueCIT.Modules.NewsMultipleView.Components

{

class ItemController

{

public void CreateItem(Item t)

{

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

rep.Insert(t);

}

}

public void DeleteItem(int itemId, int moduleId)

{

var t = GetItem(itemId, moduleId);

DeleteItem(t);

}

public void DeleteItem(Item t)

{

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

rep.Delete(t);

}

}

public IEnumerable<Item> GetItems(int moduleId)

{

IEnumerable<Item> t;

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

t = rep.Get(moduleId);

}

return t;

}

public IPagedList<Item> GetItems(int pageIndex, int pageSize)

{

IPagedList<Item> t;

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

t = rep.GetPage(pageIndex, pageSize);

// truong hop phan trang kem dieu khien loc du lieu

//t = rep.Find(pageIndex, pageSize, "WHERE ItemName like N'%@0%' ORDER BY ItemName", "My");

}

return t;

}

public IPagedList<Item> GetItems(int moduleId, int pageIndex, int pageSize)

{

IPagedList<Item> t;

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

t = rep.GetPage(moduleId, pageIndex, pageSize);

// truong hop phan trang kem dieu khien loc du lieu

//t = rep.Find(pageIndex, pageSize, "WHERE ModuleId = @0 AND ItemName like @1 ORDER BY ItemName", moduleId,"%My%");

}

return t;

}

public Item GetItem(int itemId, int moduleId)

{

Item t;

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

t = rep.GetById(itemId, moduleId);

}

return t;

}

public void UpdateItem(Item t)

{

using (IDataContext ctx = DataContext.Instance())

{

var rep = ctx.GetRepository<Item>();

rep.Update(t);

}

}

public static string GetItemLink(int tabId, int itemId)

{

return DotNetNuke.Common.Globals.NavigateURL(tabId, string.Empty, "tid=" + itemId);

}

}

}

## 2. Lập trình phần cấu hình cho module

* Cấu hình page size dùng để phân trang
* Cấu hình chọn view trình bày danh sách bài viết
* Cấu hình chọn view trình bày chi tiết bài viết

### Giao diện Settings.ascx

<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="Settings.ascx.cs" Inherits="HueCIT.Modules.NewsMultipleView.Settings" %>

<!-- uncomment the code below to start using the DNN Form pattern to create and update settings -->

<%@ Register TagName="label" TagPrefix="dnn" Src="~/controls/labelcontrol.ascx" %>

<h2 id="dnnSitePanel-BasicSettings" class="dnnFormSectionHead"><a href="" class="dnnSectionExpanded"><%=LocalizeString("BasicSettings")%></a></h2>

<fieldset>

<div class="dnnFormItem">

<div class="dnnFormItem">

<dnn:Label ID="lblPageSize" runat="server" />

<asp:TextBox ID="txtPageSize" runat="server" />

</div>

</div>

<div class="dnnFormItem">

<dnn:Label ID="lblListControls" runat="server" />

<asp:DropDownList ID="ddlListControls" runat="server" Width="60%">

</asp:DropDownList>

</div>

<div class="dnnFormItem">

<dnn:Label ID="lblbDetailControls" runat="server" />

<asp:DropDownList ID="ddlDetailControls" runat="server" Width="60%">

</asp:DropDownList>

</div>

</fieldset>

### Code behind Settings.ascx.cs

/\*

' Copyright (c) 2014 huecit.vn

' All rights reserved.

'

' THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

' TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

' THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

' CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

' DEALINGS IN THE SOFTWARE.

'

\*/

using System;

using DotNetNuke.Entities.Modules;

using DotNetNuke.Services.Exceptions;

using System.Web.UI.WebControls;

using System.IO;

using Microsoft.VisualBasic;

namespace HueCIT.Modules.NewsMultipleView

{

/// -----------------------------------------------------------------------------

/// <summary>

/// The Settings class manages Module Settings

///

/// Typically your settings control would be used to manage settings for your module.

/// There are two types of settings, ModuleSettings, and TabModuleSettings.

///

/// ModuleSettings apply to all "copies" of a module on a site, no matter which page the module is on.

///

/// TabModuleSettings apply only to the current module on the current page, if you copy that module to

/// another page the settings are not transferred.

///

/// If you happen to save both TabModuleSettings and ModuleSettings, TabModuleSettings overrides ModuleSettings.

///

/// Below we have some examples of how to access these settings but you will need to uncomment to use.

///

/// Because the control inherits from NewsMultipleViewSettingsBase you have access to any custom properties

/// defined there, as well as properties from DNN such as PortalId, ModuleId, TabId, UserId and many more.

/// </summary>

/// -----------------------------------------------------------------------------

public partial class Settings : NewsMultipleViewModuleSettingsBase

{

#region Base Method Implementations

private System.Web.UI.WebControls.DropDownList FillListControls(System.Web.UI.WebControls.DropDownList ddlTemplate, String path, string selectedValue, string fileExtension)

{

ListItem item;

if (Directory.Exists(path))

{

string[] files = Directory.GetFiles(path, fileExtension);

foreach (string file in files)

{

item = new ListItem();

item.Text = Path.GetFileNameWithoutExtension(file);

item.Value = Path.GetFileName(file);

if (item.Value == selectedValue)

{

item.Selected = true;

}

ddlTemplate.Items.Add(item);

}

}

return ddlTemplate;

}

/// -----------------------------------------------------------------------------

/// <summary>

/// LoadSettings loads the settings from the Database and displays them

/// </summary>

/// -----------------------------------------------------------------------------

public override void LoadSettings()

{

try

{

if (Page.IsPostBack == false)

{

//Check for existing settings and use those on this page

//Settings["SettingName"]

if (Settings.Contains("PageSize"))

txtPageSize.Text = Settings["PageSize"].ToString();

//Load the list controls

string listTemplate = "ULList.ascx";

if (Settings.Contains("ListTemplate"))

{

listTemplate = Settings["ListTemplate"].ToString();

}

ddlListControls = FillListControls(ddlListControls, Request.MapPath(this.ControlPath + "ListControls"), listTemplate, "\*.ascx");

//Load the detail controls

string detailTemplate = "ItemDetaila.ascx";

if (Settings.Contains("DetailTemplate"))

{

listTemplate = Settings["DetailTemplate"].ToString();

}

ddlDetailControls = FillListControls(ddlDetailControls, Request.MapPath(this.ControlPath + "DetailControls"), detailTemplate, "\*.ascx");

}

}

catch (Exception exc) //Module failed to load

{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

/// -----------------------------------------------------------------------------

/// <summary>

/// UpdateSettings saves the modified settings to the Database

/// </summary>

/// -----------------------------------------------------------------------------

public override void UpdateSettings()

{

try

{

var modules = new ModuleController();

//the following are two sample Module Settings, using the text boxes that are commented out in the ASCX file.

//module settings

//modules.UpdateModuleSetting(ModuleId, "Setting1", txtSetting1.Text);

//modules.UpdateModuleSetting(ModuleId, "Setting2", txtSetting2.Text);

//tab module settings

//modules.UpdateTabModuleSetting(TabModuleId, "Setting1", txtSetting1.Text);

//modules.UpdateTabModuleSetting(TabModuleId, "Setting2", txtSetting2.Text);

modules.UpdateModuleSetting(ModuleId, "PageSize", txtPageSize.Text);

if (ddlListControls.SelectedValue != null)

modules.UpdateModuleSetting(ModuleId, "ListTemplate", ddlListControls.SelectedValue);

if (ddlDetailControls.SelectedValue != null)

modules.UpdateModuleSetting(ModuleId, "DetailTemplate", ddlListControls.SelectedValue);

}

catch (Exception exc) //Module failed to load

{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

#endregion

}

}

## 3. Lập trình lớp ModuleBase NewsMultipleViewModuleBase

Định nghĩa các thuộc tính cần thiết cho module ở lớp ModuleBase (NewsMultipleViewModuleBase)

Bổ sung các thuộc tính PageSize và PageIndex, ListTemplate, DetailTemplate vào lớp ModuleBase (NewsMultipleViewModuleBase) vào của module để lấy các giá trị pageSize và pageIndex, tên view dùng để hiển thị danh sách record, và tên view dùng để hiển thị chi tiết một record

Bổ hàm trả về liên kết đến một item GetItemLink

/\*

' Copyright (c) 2014 huecit.vn

' All rights reserved.

'

' THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

' TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

' THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

' CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

' DEALINGS IN THE SOFTWARE.

'

\*/

using System;

using DotNetNuke.Entities.Modules;

using HueCIT.Modules.NewsMultipleView.Components;

namespace HueCIT.Modules.NewsMultipleView

{

public class NewsMultipleViewModuleBase : PortalModuleBase

{

public int ItemId

{

get

{

var qs = Request.QueryString["tid"];

if (qs != null)

return Convert.ToInt32(qs);

return -1;

}

}

public int PageIndex

{

get

{

var qs = Request.QueryString["p"]; // p: standard for page

if (qs != null)

return Convert.ToInt32(qs);

return 0;

}

}

public int PageSize

{

get

{

if (Settings.Contains("PageSize"))

return Convert.ToInt32(Settings["PageSize"]);

return 10;

}

set

{

var mc = new ModuleController();

mc.UpdateModuleSetting(ModuleId, "PageSize", value.ToString());

}

}

public string ListTemplate

{

get

{

if (Settings.Contains("ListTemplate"))

return Settings["ListTemplate"].ToString();

return "ListControls/ULList.ascx";

}

set

{

var mc = new ModuleController();

mc.UpdateModuleSetting(ModuleId, "ListTemplate", value);

}

}

public string DetailTemplate

{

get

{

if (Settings.Contains("Detailemplate"))

return Settings["Detailemplate"].ToString();

return "DetailControls/ItemDetaila.ascx";

}

set

{

var mc = new ModuleController();

mc.UpdateModuleSetting(ModuleId, "Detailemplate", value);

}

}

public string GetItemLink(string itemId)

{

return ItemController.GetItemLink(TabId, Convert.ToInt32(itemId));

}

}

}

## 4. Lập trình tạo các view dùng để hiển thị dữ liệu

Tạo các view ở các thư mục tương ứng để hiển thị danh sách record và chi tiết của một record

### 4.1. Thư mục ListControls lưu các usercontrol dùng để trình bày danh sách các records

Tạo một số control dùng để trình bày danh sách các records

#### 4.1.1. Control ULList.ascx dùng để hiển thị danh sách các record ở dạng liệt kê dùng thẻ ul

##### 4.1.1.1 Giao diện

<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="ULList.ascx.cs" Inherits="HueCIT.Modules.NewsMultipleView.ULList" %>

<asp:Repeater ID="rptItemList" runat="server" OnItemDataBound="rptItemListOnItemDataBound" OnItemCommand="rptItemListOnItemCommand">

<HeaderTemplate>

<ul class="tm\_tl">

</HeaderTemplate>

<ItemTemplate>

<li class="tm\_t">

<h3>

<asp:HyperLink ID="lnkItemName" runat="server" NavigateUrl='<%# GetItemLink(DataBinder.Eval(Container.DataItem,"ItemId").ToString())%>'><%# System.Web.HttpUtility.HtmlEncode(DataBinder.Eval(Container.DataItem,"ItemName").ToString()) %></asp:HyperLink>

</h3>

<asp:Label ID="lblItemDescription" runat="server" Text='<%#DataBinder.Eval(Container.DataItem,"ItemDescription").ToString() %>' CssClass="tm\_td" />

<asp:Panel ID="pnlAdmin" runat="server" Visible="false">

<asp:HyperLink ID="lnkEdit" runat="server" ResourceKey="EditItem.Text" Visible="false" Enabled="false" />

<asp:LinkButton ID="lnkDelete" runat="server" ResourceKey="DeleteItem.Text" Visible="false" Enabled="false" CommandName="Delete" />

</asp:Panel>

</li>

</ItemTemplate>

<FooterTemplate>

</ul>

</FooterTemplate>

</asp:Repeater>

<asp:Panel ID="pnlPaging" runat="server">

<asp:HyperLink ID="lnkPrevious" runat="server" resourcekey="lnkPrevious" Visible="false"

CssClass="lnkPrevious"></asp:HyperLink>

<asp:HyperLink ID="lnkNext" runat="server" resourcekey="lnkNext" Visible="false"

CssClass="lnkNext"></asp:HyperLink>

</asp:Panel>

##### 4.1.1.2 . Code behind

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using HueCIT.Modules.NewsMultipleView.Components;

using DotNetNuke.Security;

using DotNetNuke.Collections;

using DotNetNuke.Services.Exceptions;

using DotNetNuke.Entities.Modules;

using DotNetNuke.Entities.Modules.Actions;

using DotNetNuke.Services.Localization;

using DotNetNuke.UI.Utilities;

using System.Collections.Specialized;

using System.Globalization;

namespace HueCIT.Modules.NewsMultipleView

{

public partial class ULList : NewsMultipleViewModuleBase, IActionable

{

protected void Page\_Load(object sender, EventArgs e)

{

try

{

var tc = new ItemController();

IPagedList<Item> items = tc.GetItems(ModuleId, PageIndex, PageSize);

rptItemList.DataSource = items;

rptItemList.DataBind();

BuildPageList(items.PageCount);

}

catch (Exception exc) //Module failed to load

{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

protected void rptItemListOnItemDataBound(object sender, RepeaterItemEventArgs e)

{

if (e.Item.ItemType == ListItemType.AlternatingItem || e.Item.ItemType == ListItemType.Item)

{

var lnkEdit = e.Item.FindControl("lnkEdit") as HyperLink;

var lnkDelete = e.Item.FindControl("lnkDelete") as LinkButton;

var pnlAdminControls = e.Item.FindControl("pnlAdmin") as Panel;

var t = (Item)e.Item.DataItem;

if (IsEditable && lnkDelete != null && lnkEdit != null && pnlAdminControls != null)

{

pnlAdminControls.Visible = true;

lnkDelete.CommandArgument = t.ItemId.ToString();

lnkDelete.Enabled = lnkDelete.Visible = lnkEdit.Enabled = lnkEdit.Visible = true;

lnkEdit.NavigateUrl = EditUrl(string.Empty, string.Empty, "Edit", "tid=" + t.ItemId);

ClientAPI.AddButtonConfirm(lnkDelete, Localization.GetString("ConfirmDelete", LocalResourceFile));

}

else

{

pnlAdminControls.Visible = false;

}

}

}

public void rptItemListOnItemCommand(object source, RepeaterCommandEventArgs e)

{

if (e.CommandName == "Edit")

{

Response.Redirect(EditUrl(string.Empty, string.Empty, "Edit", "tid=" + e.CommandArgument));

}

if (e.CommandName == "Delete")

{

var tc = new ItemController();

tc.DeleteItem(Convert.ToInt32(e.CommandArgument), ModuleId);

}

Response.Redirect(DotNetNuke.Common.Globals.NavigateURL());

}

private void BuildPageList(int pageCount)

{

NameValueCollection queryString = Request.QueryString;

SetPagingLink(queryString, lnkNext, PageIndex + 1 < pageCount, PageIndex + 1, TabId);

SetPagingLink(queryString, lnkPrevious, PageIndex - 1 > -1, PageIndex - 1, TabId);

}

private static void SetPagingLink(NameValueCollection queryString

, HyperLink link, bool showLink, int linkedPageId, int tabId)

{

if (showLink)

{

link.Visible = true;

queryString = new NameValueCollection(queryString);

queryString["p"] = linkedPageId.ToString(CultureInfo.InvariantCulture);

var additionalParameters = new List<string>(queryString.Count);

for (int i = 0; i < queryString.Count; i++)

{

if (string.Equals(queryString.GetKey(i), "TABID", StringComparison.OrdinalIgnoreCase))

{

int newTabId;

if (int.TryParse(queryString[i], NumberStyles.Integer, CultureInfo.InvariantCulture, out newTabId))

{

tabId = newTabId;

}

}

else if (!string.Equals(queryString.GetKey(i), "LANGUAGE", StringComparison.OrdinalIgnoreCase))

{

additionalParameters.Add(queryString.GetKey(i) + "=" + queryString[i]);

}

}

link.NavigateUrl = DotNetNuke.Common.Globals.NavigateURL(tabId, string.Empty, additionalParameters.ToArray());

}

else

{

link.Visible = false;

}

}

public ModuleActionCollection ModuleActions

{

get

{

var actions = new ModuleActionCollection

{

{

GetNextActionID(), Localization.GetString("EditModule", LocalResourceFile), "", "", "",

EditUrl(), false, SecurityAccessLevel.Edit, true, false

}

};

return actions;

}

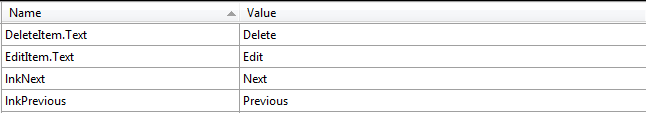
}

}

}

##### 4.1.1.3 . Resource file ULList.ascx.resx

Thêm mới thư mục ở thư mục ListControls thêm mới thư mục App\_LocalResources rồi thêm mới file resource ULList.ascx.resx



#### 4.1.2. Control TableList.ascx dùng để hiển thị danh sách các record ở dạng bảng dùng thẻ table

##### 4.1.2.1 Giao diện

<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="TableList.ascx.cs" Inherits="HueCIT.Modules.NewsMultipleView.TableList" %>

<asp:Repeater ID="rptItemList" runat="server" OnItemDataBound="rptItemListOnItemDataBound" OnItemCommand="rptItemListOnItemCommand">

<HeaderTemplate>

<table>

<tr>

<td>Item Name</td>

<td>Item Description</td>

<td>Actions</td>

</tr>

</HeaderTemplate>

<ItemTemplate>

<tr>

<td>

<asp:HyperLink ID="lnkItemName" runat="server" NavigateUrl='<%# GetItemLink(DataBinder.Eval(Container.DataItem,"ItemId").ToString())%>'><%# System.Web.HttpUtility.HtmlEncode(DataBinder.Eval(Container.DataItem,"ItemName").ToString()) %></asp:HyperLink>

</td>

<td>

<asp:Label ID="lblItemDescription" runat="server" Text='<%#DataBinder.Eval(Container.DataItem,"ItemDescription").ToString() %>' CssClass="tm\_td" />

</td>

<td>

<asp:Panel ID="pnlAdmin" runat="server" Visible="false">

<asp:HyperLink ID="lnkEdit" runat="server" ResourceKey="EditItem.Text" Visible="false" Enabled="false" />

<asp:LinkButton ID="lnkDelete" runat="server" ResourceKey="DeleteItem.Text" Visible="false" Enabled="false" CommandName="Delete" />

</asp:Panel>

</td>

</tr>

</ItemTemplate>

<FooterTemplate>

</table>

</FooterTemplate>

</asp:Repeater>

<asp:Panel ID="pnlPaging" runat="server">

<asp:HyperLink ID="lnkPrevious" runat="server" resourcekey="lnkPrevious" Visible="false"

CssClass="lnkPrevious"></asp:HyperLink>

<asp:HyperLink ID="lnkNext" runat="server" resourcekey="lnkNext" Visible="false"

CssClass="lnkNext"></asp:HyperLink>

</asp:Panel>

##### 4.1.2.2 . Code behind

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using HueCIT.Modules.NewsMultipleView.Components;

using DotNetNuke.Security;

using DotNetNuke.Collections;

using DotNetNuke.Services.Exceptions;

using DotNetNuke.Entities.Modules;

using DotNetNuke.Entities.Modules.Actions;

using DotNetNuke.Services.Localization;

using DotNetNuke.UI.Utilities;

using System.Collections.Specialized;

using System.Globalization;

namespace HueCIT.Modules.NewsMultipleView

{

public partial class TableList : NewsMultipleViewModuleBase, IActionable

{

protected void Page\_Load(object sender, EventArgs e)

{

try

{

var tc = new ItemController();

IPagedList<Item> items = tc.GetItems(ModuleId, PageIndex, PageSize);

rptItemList.DataSource = items;

rptItemList.DataBind();

BuildPageList(items.PageCount);

}

catch (Exception exc) //Module failed to load

{

Exceptions.ProcessModuleLoadException(this, exc);

}

}

protected void rptItemListOnItemDataBound(object sender, RepeaterItemEventArgs e)

{

if (e.Item.ItemType == ListItemType.AlternatingItem || e.Item.ItemType == ListItemType.Item)

{

var lnkEdit = e.Item.FindControl("lnkEdit") as HyperLink;

var lnkDelete = e.Item.FindControl("lnkDelete") as LinkButton;

var pnlAdminControls = e.Item.FindControl("pnlAdmin") as Panel;

var t = (Item)e.Item.DataItem;

if (IsEditable && lnkDelete != null && lnkEdit != null && pnlAdminControls != null)

{

pnlAdminControls.Visible = true;

lnkDelete.CommandArgument = t.ItemId.ToString();

lnkDelete.Enabled = lnkDelete.Visible = lnkEdit.Enabled = lnkEdit.Visible = true;

lnkEdit.NavigateUrl = EditUrl(string.Empty, string.Empty, "Edit", "tid=" + t.ItemId);

ClientAPI.AddButtonConfirm(lnkDelete, Localization.GetString("ConfirmDelete", LocalResourceFile));

}

else

{

pnlAdminControls.Visible = false;

}

}

}

public void rptItemListOnItemCommand(object source, RepeaterCommandEventArgs e)

{

if (e.CommandName == "Edit")

{

Response.Redirect(EditUrl(string.Empty, string.Empty, "Edit", "tid=" + e.CommandArgument));

}

if (e.CommandName == "Delete")

{

var tc = new ItemController();

tc.DeleteItem(Convert.ToInt32(e.CommandArgument), ModuleId);

}

Response.Redirect(DotNetNuke.Common.Globals.NavigateURL());

}

private void BuildPageList(int pageCount)

{

NameValueCollection queryString = Request.QueryString;

SetPagingLink(queryString, lnkNext, PageIndex + 1 < pageCount, PageIndex + 1, TabId);

SetPagingLink(queryString, lnkPrevious, PageIndex - 1 > -1, PageIndex - 1, TabId);

}

private static void SetPagingLink(NameValueCollection queryString

, HyperLink link, bool showLink, int linkedPageId, int tabId)

{

if (showLink)

{

link.Visible = true;

queryString = new NameValueCollection(queryString);

queryString["p"] = linkedPageId.ToString(CultureInfo.InvariantCulture);

var additionalParameters = new List<string>(queryString.Count);

for (int i = 0; i < queryString.Count; i++)

{

if (string.Equals(queryString.GetKey(i), "TABID", StringComparison.OrdinalIgnoreCase))

{

int newTabId;

if (int.TryParse(queryString[i], NumberStyles.Integer, CultureInfo.InvariantCulture, out newTabId))

{

tabId = newTabId;

}

}

else if (!string.Equals(queryString.GetKey(i), "LANGUAGE", StringComparison.OrdinalIgnoreCase))

{

additionalParameters.Add(queryString.GetKey(i) + "=" + queryString[i]);

}

}

link.NavigateUrl = DotNetNuke.Common.Globals.NavigateURL(tabId, string.Empty, additionalParameters.ToArray());

}

else

{

link.Visible = false;

}

}

public ModuleActionCollection ModuleActions

{

get

{

var actions = new ModuleActionCollection

{

{

GetNextActionID(), Localization.GetString("EditModule", LocalResourceFile), "", "", "",

EditUrl(), false, SecurityAccessLevel.Edit, true, false

}

};

return actions;

}

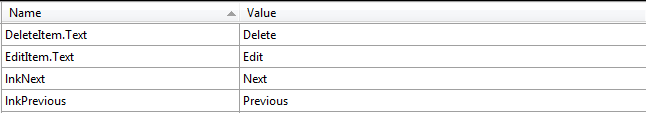
}

}

}

##### 4.1.2.3 . Resource file TableList.ascx.resx

Thêm mới file resource TableList.ascx.resx



### 4.2. DetailControls để lưu các usercontrol dùng để trình bày chi tiết của một record

#### 4.2.1. Control ItemDetaila.ascx dùng để hiển thị thông tinchi tiết của một record

##### 4.2.1.1 Giao diện

<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="ItemDetaila.ascx.cs" Inherits="HueCIT.Modules.NewsMultipleView.ItemDetaila" %>

<asp:Panel ID="pnlItemName" runat="server"/>

<asp:Panel ID="pnlItemDescription" runat="server"/>

<asp:Panel ID="pnlAdmin" runat="server" Visible="false">

<asp:HyperLink ID="lnkEdit" runat="server" ResourceKey="EditItem.Text" Visible="false" Enabled="false" />

<asp:LinkButton ID="lnkDelete" runat="server" ResourceKey="DeleteItem.Text" Visible="false" Enabled="false" CommandName="Delete" />

</asp:Panel>

##### 4.2.1.2 . Code behind

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using HueCIT.Modules.NewsMultipleView.Components;

using DotNetNuke.Security;

using DotNetNuke.Collections;

using DotNetNuke.Services.Exceptions;

using DotNetNuke.Entities.Modules;

using DotNetNuke.Entities.Modules.Actions;

using DotNetNuke.Services.Localization;

using DotNetNuke.UI.Utilities;

using System.Collections.Specialized;

using System.Globalization;

namespace HueCIT.Modules.NewsMultipleView

{

public partial class ItemDetaila : NewsMultipleViewModuleBase

{

protected void Page\_Load(object sender, EventArgs e)

{

try

{

if (!Page.IsPostBack)

{

if (ItemId > 0)

{

var tc = new ItemController();

Item curItem = tc.GetItem(ItemId, ModuleId);

//display item info on the view control

pnlItemName.Controls.Add(new LiteralControl(curItem.ItemName));

pnlItemDescription.Controls.Add(new LiteralControl(Server.HtmlDecode(curItem.ItemDescription)));

}

}

}

catch (Exception exc)

{

Exceptions.ProcessModuleLoadException(this, exc);

}

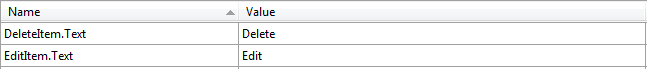
}

}

}

##### 4.2.1.3 . Resource file ItemDetaila.ascx.resx

Thêm mới thư mục ở thư mục ListControls thêm mới thư mục App\_LocalResources rồi thêm mới file resource ItemDetaila.ascx



## 5. Lập trình chọn view và hiển thị dữ liệu ở control View.ascx

### Giao diện control View.ascx

<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="View.ascx.cs" Inherits="HueCIT.Modules.NewsMultipleView.View" %>

<asp:PlaceHolder ID="phTemplate" runat="server"></asp:PlaceHolder>

### Hàm Page\_Load ở code behind của control View.ascx.cs

protected void Page\_Load(object sender, EventArgs e)

{

PortalModuleBase objListTemplate;

var controlToLoad = ListTemplate;

if (ItemId > 0)

{

controlToLoad = DetailTemplate;

}

objListTemplate = (PortalModuleBase)LoadControl(controlToLoad);

objListTemplate.ModuleConfiguration = ModuleConfiguration;

objListTemplate.ID = System.IO.Path.GetFileNameWithoutExtension(controlToLoad);

phTemplate.Controls.Add(objListTemplate);

}

## 6. Biên dịch và cài đặt, chạy thử

### Giao diện module khi chọn view trình bày danh sách theo dạng liệt kê dùng thẻ ul



### Giao diên module khi chọn view trình bày danh sách theo dạng bảng dùng thẻ table

