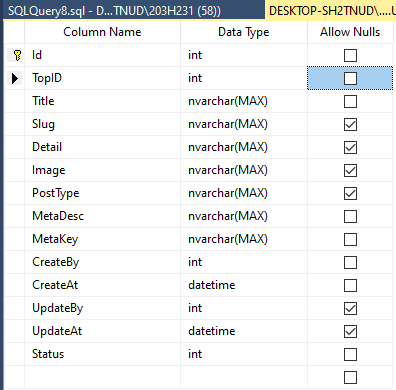
# Bài thực hành 9: XÂY DỰNG MENU CRUD CHO POSTS

Thiết kế lại CSDL



Mục đích của TopID bắt buộc khác NULL để tránh trường hợp bài viết không có chủ đề

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.ComponentModel.DataAnnotations;

using System.ComponentModel.DataAnnotations.Schema;

namespace MyClass.Model

{

[Table("Posts")]

public class Posts

{

[Key]

public int Id { get; set; }

[Required(ErrorMessage = "Chủ đề bài viết không được để trống")]

[Display(Name = "Chủ đề bài viết")]

public int TopID { get; set; }

[Required(ErrorMessage = "Tên bài viết không được để trống")]

[Display(Name = "Tên bài viết")]

public string Title { get; set; }

[Display(Name = "Liên kết")]

public string Slug { get; set; }

[Display(Name = "Chi tiết")]

public string Detail { get; set; }

[Display(Name = "Ảnh bài viết")]

public string Image { get; set; }

[Display(Name = "Kiểu bài viết")]

public string PostType { get; set; }

[Required(ErrorMessage = "Mô tả không được để trống")]

[Display(Name = "Mô tả")]

public string MetaDesc { get; set; }

[Required(ErrorMessage = "Từ khóa không được để trống")]

[Display(Name = "Từ khóa")]

public string MetaKey { get; set; }

[Required(ErrorMessage = "Người tạo không được để trống")]

[Display(Name = "Người tạo")]

public int CreateBy { get; set; }

[Required(ErrorMessage = "Ngày tạo không được để trống")]

[Display(Name = "Ngày tạo")]

public DateTime CreateAt { get; set; }

[Display(Name = "Cập nhật bởi")]

public int? UpdateBy { get; set; }

[Display(Name = "Ngày cập nhật")]

public DateTime? UpdateAt { get; set; }

[Required(ErrorMessage = "Trạng thái không được để trống")]

[Display(Name = "Trạng thái")]

public int Status { get; set; }

}

}

# create

## POST controller

/////////////////////////////////////////////////////////////////////////////////////

// GET: Admin/Post/Create: Them moi mot mau tin

public ActionResult Create()

{

ViewBag.TopList = new SelectList(topicsDAO.getList("Index"), "Id", "Name");[[1]](#footnote-1)

return View();

}

// POST: Admin/Post/Create: Them moi mot mau tin

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create(Posts posts)

{

if (ModelState.IsValid)

{

//Xu ly cho muc Slug

posts.Slug = XString.Str\_Slug(posts.Title);

//chuyen doi dua vao truong Name de loai bo dau, khoang cach = dau -

//xu ly cho phan upload hình ảnh

var img = Request.Files["img"];//lay thong tin file

if (img.ContentLength != 0)

{

string[] FileExtentions = new string[] { ".jpg", ".jpeg", ".png", ".gif" };

//kiem tra tap tin co hay khong

if (FileExtentions.Contains(img.FileName.Substring(img.FileName.LastIndexOf("."))))

//lay phan mo rong cua tap tin

{

string slug = posts.Slug;

string id = posts.Id.ToString();

//Chinh sua sau khi phat hien dieu chua dung cua Edit: them Id

//ten file = Slug + Id + phan mo rong cua tap tin

string imgName = slug + id + img.FileName.Substring(img.FileName.LastIndexOf("."));

posts.Image = imgName;

string PathDir = "~/Public/img/post/";

string PathFile = Path.Combine(Server.MapPath(PathDir), imgName);

//upload hinh

img.SaveAs(PathFile);

}

}//ket thuc phan upload hinh anh

//xu ly cho muc PostType

posts.PostType = "post";

//Xu ly cho muc CreateAt

posts.CreateAt = DateTime.Now;

//Xu ly cho muc CreateBy

posts.CreateBy = Convert.ToInt32(Session["UserId"]);

//xu ly cho muc Topics

if (postsDAO.Insert(posts) == 1)//khi them du lieu thanh cong

{

Links links = new Links();

links.Slug = posts.Slug;

links.TableId = posts.Id;

links.Type = "post";

linksDAO.Insert(links);

}

//Thong bao thanh cong

TempData["message"] = new XMessage("success", "Thêm bài viết thành công");

return RedirectToAction("Index");

}

ViewBag.TopList = new SelectList(topicsDAO.getList("Index"), "Id", "Name");

return View(posts);

}

## create.cshtml

### Tiêu đề

@model MyClass.Model.Posts

@{

ViewBag.Title = "Thêm mới bài viết";

Layout = "~/Areas/Admin/Views/Shared/\_LayoutAdmin.cshtml";

}

#### Lưu ý:

Trong form create chúng ta có sử dụng: upload hình ảnh

@using (Html.BeginForm("Create", "Post", FormMethod.Post, new { @enctype = "multipart/form-data" }))

### Thiết kế lại trang nhập thông tin

@model MyClass.Model.Posts

@{

ViewBag.Title = "Thêm mới bài viết";

Layout = "~/Areas/Admin/Views/Shared/\_LayoutAdmin.cshtml";

}

@using (Html.BeginForm("Create", "Post", FormMethod.Post, new { @enctype = "multipart/form-data" }))

{

@Html.AntiForgeryToken()

<!-- Content Wrapper. Contains page content -->

<div class="content-wrapper">

<!-- Main content -->

<section class="content my-2">

<!-- Default box -->

<div class="card">

<div class="card-header">

<div class="row">

<div class="col-md-6">

<strong class="text-danger">THÊM MỚI BÀI VIẾT</strong>

</div>

<div class="col-md-6 text-right">

<a href="@Url.Action("Index", "Post")" class="btn-sm btn-info"> <i class="fa fa-long-arrow-alt-left"></i> Quay lại</a>

</div>

</div>

</div>

<div class="card-body">

@\*Phan NOI DUNG\*@

<div class="row">

@\* Phần nhập tay \*@

<div class="col-md-9">

<div class="form-group">

<label>Tên bài viết <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.EditorFor(model => model.Title, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.Title, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Mô tả chi tiết <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.TextAreaFor(model => model.MetaDesc, new { @class = "form-control", @rows = "2" })

@Html.ValidationMessageFor(model => model.MetaDesc, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Từ khóa <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.TextAreaFor(model => model.MetaKey, new { @class = "form-control", @rows = "2" })

@Html.ValidationMessageFor(model => model.MetaKey, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Chi tiết bài viết <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.TextAreaFor(model => model.Detail, new { @class = "form-control", @rows = "2" })

@Html.ValidationMessageFor(model => model.Detail, "", new { @class = "text-danger" })

</div>

</div>

</div>

@\* Phần lựa chọn \*@

<div class="col-md-3">

<div class="form-group">

<label>Ảnh đại diện <span class="text-danger"></span></label>

<div class="col-md-10">

<input name="img" type="file" class="form-control" />

</div>

</div>

<div class="form-group">

<label>Chủ đề <span class="text-danger"></span></label>

<div class="col-md-10">

@Html.DropDownListFor(model => model.TopID, (SelectList)@ViewBag.TopList, "Chủ đề", new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.TopID, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Trạng thái <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.DropDownListFor(model => model.Status, new[] { new SelectListItem { Text = "Không xuất bản", Value = "2" }, new SelectListItem { Text = "Xuất bản", Value = "1" } }, new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.Status, "", new { @class = "text-danger" })

</div>

</div>

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<button class="btn-sm btn-success" type="submit" name="create"> Lưu [Thêm] <i class="fa fa-save"></i> </button>

</div>

</div>

</div><!-- /.card-body -->

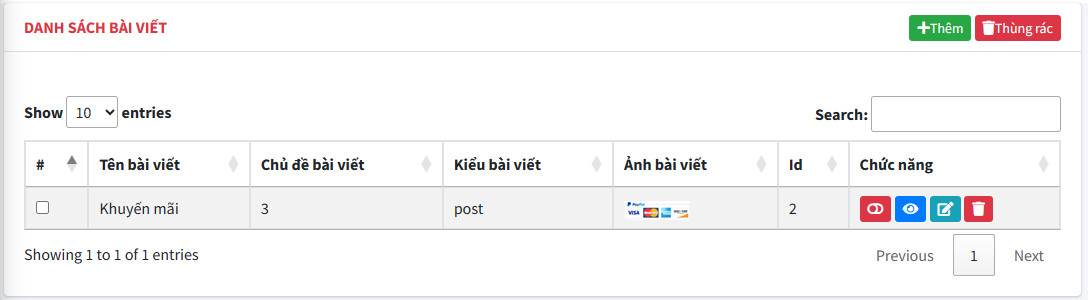
</div><!-- /.card -->

</section><!-- /.content -->

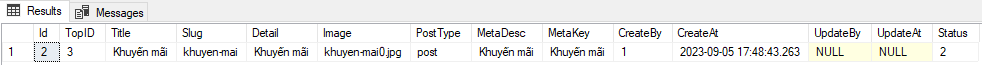
</div><!-- /.content-wrapper -->

}

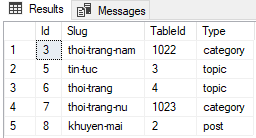
# Kết quả



## Kiểm tra Posts



## Kiểm tra links



1. Trả về danh sách Tên chủ đề dựa vào ID [↑](#footnote-ref-1)