

Lab 07: CẤU TRÚC CÁC TRANG TIN VÀ MENU TỪ DATABASE

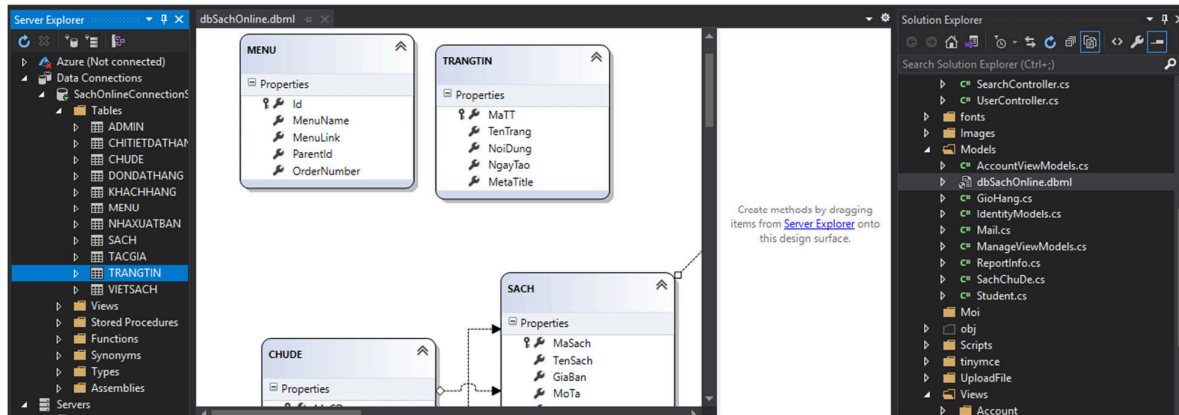
A. MỤC TIÊU:

- Xây dựng các trang Giới thiệu, Cơ cấu tổ chức (trang tin) động
- Xây dựng Menu động

B. NỘI DUNG:

1. Tạo thêm 2 table trong database SachOnline và ánh xạ vào model

- Chạy file menu.sql và trangtin.sql để tạo thêm 2 table trong database SachOnline
- Thêm 2 table này vào model: Mở file dbSachOnline.dbml, chọn Server Explorer → Kéo 2 table mới vào file



2. Hiện thị menu lên NavPartial

- Cập nhật lại action NavPartial trong SachOnlineController

```
/// <summary>
/// GetNav
/// </summary>
/// <returns>ReturnNav</returns>
[ChildActionOnly]
0 references
public ActionResult NavPartial()
{
    List<MENU> lst = new List<MENU>();
    lst = data.MENUS.Where(m => m.ParentId == null).OrderBy(m => m.OrderNumber).ToList();
    int[] a = new int[lst.Count()];
    for (int i = 0; i < lst.Count; i++)
    {
        var l = data.MENUS.Where(m => m.ParentId == lst[i].Id);
        a[i] = l.Count();
    }
    ViewBag.lst = a;
    return PartialView(lst);
}
```

- Cập nhật lại view NavPartial:

Khai báo model:

`@model IEnumerable<SachOnline.Models.MENU>`

Cập nhật phần hiển thị menu lên Nav:

```
<ul class="nav navbar-nav">
    @foreach (var item in Model)
    {
        int i = 0;
```

```

        foreach (var item in Model)
        {
            if (ViewBag.lst[i] != 0)
            {
                <li class="dropdown">
                    <a href="@item.MenuLink" class="dropdown-toggle"
                        data-toggle="dropdown" role="button" aria-expanded="false" aria-
haspopup="true">
                        @item.MenuName<span class="caret"></span>
                    </a>
                    @Html.Action("LoadChildMenu", "SachOnline", new { parentId =
item.Id })
                </li>
            }
            else
            {
                <li><a href="~/@item.MenuLink">@item.MenuName</a> </li>
            }
            i++;
        }
    }
</ul>

```

- Tạo thêm action LoadChildMenu trả về 1 PartialView:

```

[ChildActionOnly]
0 references
public ActionResult LoadChildMenu(int parentId)
{
    List<MENU> lst = new List<MENU>();
    lst = data.MENUs.Where(m => m.ParentId == parentId).OrderBy(m => m.OrderNumber).ToList();
    ViewBag.Count = lst.Count();
    int[] a = new int[lst.Count()];
    for (int i = 0; i < lst.Count; i++)
    {
        var l = data.MENUs.Where(m => m.ParentId == lst[i].Id);
        a[i] = l.Count();
    }
    ViewBag.lst = a;
    return PartialView("LoadChildMenu", lst);
}

```

- Thêm view LoadChildMenu dạng partial trong thư mục Shared với nội dung:

```

@model IEnumerable<SachOnline.Models.MENU>
@{
    int i = 0;
    @:<ul class="dropdown-menu">
        foreach (var item in Model)
        {
            if (ViewBag.lst[i] != 0)
            {
                <li class="dropdown-submenu">
                    <a href="#" class="test list-group-item">
                        @item.MenuName <span class="caret"></span>
                    </a>
                    @Html.Action("LoadChildMenu", "SachOnline", new { parentId = item.Id })
                </li>
            }
            i++;
        }
    }
}

```

```

    }
    else
    {
        <li>
            <a href="@item.MenuLink" class="list-group-item">@item.MenuName</a>
        </li>
    }
    i++;
}
@:</ul>
}

```

- Thêm đoạn css và jquery sau vào cuối NavPartial:

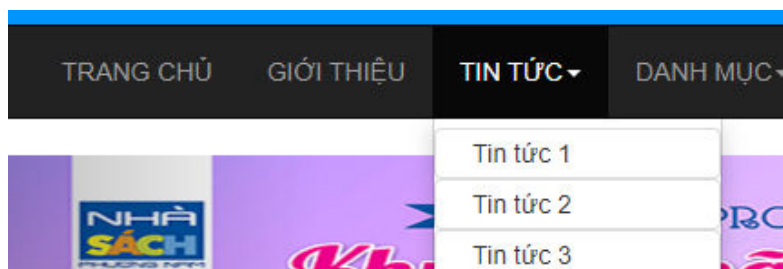
```

<style>
    .dropdown-submenu {
        position: relative;
    }

    .dropdown-submenu .dropdown-menu {
        top: 0;
        left: 100%;
        margin-top: -1px;
    }
</style>
<script>
    $(document).ready(function () {
        $('.dropdown-submenu a.test').on("click", function (e) {
            $(this).next('ul').toggle();
            e.stopPropagation(); /*Phương thức stopPropagation() của đối tượng event
            * được sử dụng để ngăn không cho sự kiện lan toả lên các phần tử cha
            * của phần tử diễn ra sự kiện.*/
            e.preventDefault(); /*Phương thức preventDefault() của đối tượng event
            * được sử dụng để ngăn chặn xử lý mặc định của trình duyệt khi xảy ra sự kiện.*/
        });
    });
</script>

```

- Kết quả:



Vào table MENU thêm dữ liệu để kiểm tra menu cấp 2, 3, ...

3. Xây dựng chức năng hiển thị TrangTin:

- Bổ sung action TrangTin() vào SachController:

```

public ActionResult TrangTin(string metatitle)
{
    var tt = (from t in data.TRANGTINS where t.MetaTitle == metatitle select t).Single();
    return View(tt);
}

```

- Tạo view TrangTin với nội dung:

```
@model SachOnline.Models.TRANGTIN
```

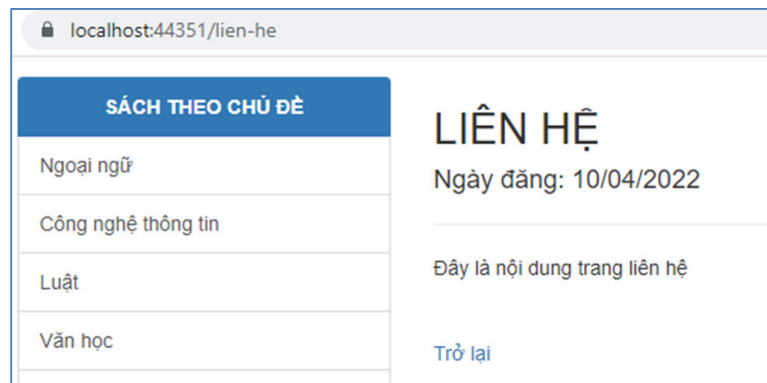
```
@{  
    ViewBag.Title = "TrangTin";  
    Layout = "~/Views/Shared/_LayoutUser.cshtml";  
}  
  
<h3>@Model.TenTrang</h3>  
<h5>Ngày đăng: @Html.FormatValue(@Model.NgayTao, "{0:dd/MM/yyyy}")</h5>  
  
<hr />  
  
<div>  
    @Html.Raw(Model.NoiDung)  
</div>  
  
    <br /><br />  
  
    <p>  
        @Html.ActionLink("Back to List", "Index")  
    </p>
```

- Tạo Route để thực hiện liên kết mở trang: Trong file RouteConfig.cs tạo route sau:

```
routes.MapRoute(  
    name: "Trang tin",  
    url: "{metatitle}",  
    defaults: new { controller = "SachOnline", action = "TrangTin",  
        metatitle = UrlParameter.Optional },  
    namespaces: new string[] { "SachOnline.Controllers" }  
);  
  
routes.MapRoute(  
    name: "Default",  
    url: "{controller}/{action}/{id}",  
    defaults: new { controller = "SachOnline", action = "Index", id = UrlParameter.Optional }  
    //Thêm hàng sau để tránh xung đột giữa các controller Home  
    , namespaces: new[] { "SachOnline.Controllers" }  
);
```

- Kết quả khi click chuột vào mục GIỚI THIỆU và LIÊN HỆ trên menu:





4. Xây dựng các chức năng thêm, sửa, xóa TRANG TIN Admin

- Tạo file class String.cs trong thư mục Controller của Admin rồi copy code từ file String.cs trong thư mục Lab09 vào.
- Tạo controller TrangTinController trong thư mục Controller của Admin
- Tạo action Index để trả về danh sách các tin trong table TrangTin và view Index:

```
public ActionResult Index()
{
    return View(db.TRANGTINs.ToList());
}
```

```
@model IEnumerable<SachOnline.Models.TRANGTIN>

@{
    ViewBag.Title = "Index";
    Layout = "~/Areas/Admin/Views/Shared/_LayoutAdmin.cshtml";
}

<h2>DANH SÁCH TRANG TIN</h2>
<p>
    @Html.ActionLink("Create New", "Create")
</p>
<table class="table">
    <tr>
        <th>
            @Html.DisplayNameFor(model => model.TenTrang)
        </th>
        <th>
            @Html.DisplayNameFor(model => model.NoiDung)
        </th>
        <th>
            @Html.DisplayNameFor(model => model.NgayTao)
        </th>
        <th>
            @Html.DisplayNameFor(model => model.MetaTitle)
        </th>
        <th></th>
    </tr>

    @foreach (var item in Model) {
```

```

<tr>
  <td>
    @Html.DisplayFor(modelItem => item.TenTrang)
  </td>
  <td>
    @Html.DisplayFor(modelItem => item.NoiDung)
  </td>
  <td>
    @Html.DisplayFor(modelItem => item.NgayTao)
  </td>
  <td>
    @Html.DisplayFor(modelItem => item.MetaTitle)
  </td>
  <td>
    <a href="@Url.Action("Edit", new { id = item.MaTT })" class="btn btn-warning"
role="button">Sửa</a>

    <a href="@Url.Action("Delete", new { id = item.MaTT })" class="btn btn-danger"
role="button">Xóa</a>
  </td>
</tr>
}
</table>

```

Kết quả:

DANH SÁCH TRANG TIN					
Create New					
Ten Trang	NoiDung	NgayTao	MetaTitle		
GIỚI THIỆU	Đây là nội dung trang giới thiệu	12/04/2021 12:00:00 AM	gioi-thieu	Sửa	Xóa
LIÊN HỆ	Đây là nội dung trang liên hệ	10/04/2022 12:00:00 AM	lien-he	Sửa	Xóa

- Xử lý phần Thêm mới TRANG TIN:

Action:

```

[HttpGet]
0 references
public ActionResult Create()
{
    return View();
}

[HttpPost]
[ValidateInput(false)]
0 references
public ActionResult Create(TRANGTIN tt)
{
    if (ModelState.IsValid)
    {
        tt.MetaTitle = tt.TenTrang.RemoveDiacritics().Replace(" ", "-");
        tt.NgayTao = DateTime.Now;
        db.TRANGTINs.InsertOnSubmit(tt);
        db.SubmitChanges();
        return RedirectToAction("Index");
    }
    return View();
}

```

View:

```
@model SachOnline.Models.TRANGTIN

@{
    ViewBag.Title = "Create";
    Layout = "~/Areas/Admin/Views/Shared/_LayoutAdmin.cshtml";
}
<script src="~/ckeditor/ckeditor.js"></script>

<h2>Create</h2>

@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()

    <div class="form-horizontal">
        <h4>TRANGTIN</h4>
        <hr />
        @Html.ValidationSummary(true, "", new { @class = "text-danger" })
        <div class="form-group">
            @Html.LabelFor(model => model.TenTrang, htmlAttributes: new { @class =
"control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.TenTrang, new { htmlAttributes = new { @class
= "form-control" } })
                @Html.ValidationMessageFor(model => model.TenTrang, "", new { @class =
"text-danger" })
            </div>
        </div>
        <div class="form-group">
            @Html.LabelFor(model => model.NoiDung, htmlAttributes: new { @class = "control-
label col-md-2" })
            <div class="col-md-10">
                <textarea name="NoiDung" id="nd"></textarea>
                <script>CKEDITOR.replace('nd');</script>
                @Html.ValidationMessageFor(model => model.NoiDung, "", new { @class = "text-
danger" })
            </div>
        </div>
        <div class="form-group">
            <div class="col-md-offset-2 col-md-10">
                <input type="submit" value="Create" class="btn btn-default" />
            </div>
        </div>
    </div>

    @Html.ActionLink("Back to List", "Index")
</div>
```

Kết quả:


```

{
    @Html.AntiForgeryToken()
    <div class="form-horizontal">
        <h4>TRANG TIN</h4>
        <hr />
        @Html.ValidationSummary(true, "", new { @class = "text-danger" })
        @Html.HiddenFor(model => model.MaTT)

        <div class="form-group">
            @Html.LabelFor(model => model.TenTrang, htmlAttributes: new { @class =
"control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.TenTrang, new { htmlAttributes = new { @class
= "form-control" } })
                @Html.ValidationMessageFor(model => model.TenTrang, "", new { @class =
"text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.NoiDung, htmlAttributes: new { @class = "control-
label col-md-2" })
            <div class="col-md-10">
                <textarea name="NoiDung" id="nd">@Model.NoiDung</textarea>
                <script>CKEDITOR.replace('nd');</script>

                @Html.ValidationMessageFor(model => model.NoiDung, "", new { @class = "text-
danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.NgayTao, htmlAttributes: new { @class = "control-
label col-md-2" })
            <div class="col-md-10">
                <input type="datetime" name="NgayTao"
value="@Convert.ToDateTime(Model.NgayTao)" class="form-control" />

                @Html.ValidationMessageFor(model => model.NgayTao, "", new { @class = "text-
danger" })
            </div>
        </div>
        <div class="form-group">
            <div class="col-md-offset-2 col-md-10">
                <input type="submit" value="Save" class="btn btn-default" />
            </div>
        </div>
    </div>
    @Html.ActionLink("Back to List", "Index")
</div>
}

```

Kết quả:


```

<dd>
    @Html.DisplayFor(model => model.TenTrang)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.NoiDung)
</dt>

<dd>
    @Html.DisplayFor(model => model.NoiDung)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.NgayTao)
</dt>

<dd>
    @Html.DisplayFor(model => model.NgayTao)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.MetaTitle)
</dt>

<dd>
    @Html.DisplayFor(model => model.MetaTitle)
</dd>

</dl>

@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()

    <div class="form-actions no-color">
        <input type="submit" value="Delete" class="btn btn-default" /> |
        @Html.ActionLink("Back to List", "Index")
    </div>
}
</div>

```

-----Hết Lab 07-----