

# Build an EF and ASP.NET Core 2.2 App HOL

This lab walks you through creating a View Component and cleans up some miscellaneous partial views. Prior to starting this lab, you must have completed Lab 8. All labs and files are available at [https://github.com/skimedid/dotnetcore\\_hol](https://github.com/skimedid/dotnetcore_hol).

## Part 1: Create the View Component Server-Side Code

- 1) Create a new folder in the MVC project named ViewComponents and add a new class named Menu.cs.
- 2) Add the following using statements:

```
using System.Threading.Tasks;
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.ViewComponents;
using SpyStore.Hol.Dal.Repos.Interfaces;
```

- 3) Make the class public, inherit from ViewComponent, and add the InvokeAsync method:

```
public class Menu : ViewComponent
{
    public async Task<IViewComponentResult> InvokeAsync() { }
```

- 4) Add a constructor that takes an instance of the ICategoryRepo and a private variable to hold the instance. This will be added into the class by the DI container:

```
private readonly ICategoryRepo _categoryRepo;
public Menu(ICategoryRepo categoryRepo)
{
    _categoryRepo = categoryRepo;
}
```

- 5) Implement the InvokeAsync method. This method uses the ICategoryRepo instance to get the list of Categories. If the call is successful, then the method returns the MenuView partial View (to be created soon). If the call fails, the method returns a ContentViewComponentResult with an error message:

```
public async Task<IViewComponentResult> InvokeAsync()
{
    return await Task.Run<IViewComponentResult>(() =>
    {
        var cats = _categoryRepo.GetAll();
        if (cats == null)
        {
            return new ContentViewComponentResult("There was an error getting the categories");
        }
        return View("MenuView", cats);
    });
}
```

## Part 2: Update the ViewImports

### Step 1: Update the \_ViewImports.cshtml File

- 1) In order to use the ViewComponent as a Tag Helper, the assembly must be registered in the \_ViewImports.cshtml file. Add the following to the end of the file:

```
@addTagHelper *, SpyStore.Hol.MVC
```

## Part 3: Create the Partial View

### Step 1: Create the MenuView partial view

- 1) Add a new folder named Components under the Views\Shared folder. Add a new folder named Menu under the Components folder. Add a new partial view named MenuView.cshtml in the new folder.
- 2) Update the code to match the following:

```
@model IEnumerable<Category>
<li><a asp-controller="Products" asp-action="Featured" class="nav-link">Featured</a></li>
@foreach (var item in Model)
{
    <li><a asp-controller="Products" asp-action="ProductList" asp-route-id="@item.Id"
        class="nav-link">@item.CategoryName</a>
    </li>
}
```

## Part 3: Update the Main Layout

- 1) Open the \_Layout.cshtml file in Views\Shared folder and add the view component as a tag helper between the Home and Privacy menu items:

```
<li class="nav-item">
    <a class="nav-link text-dark" asp-area="" asp-controller="Products" asp-action="Index">Home</a>
</li>
<vc:menu></vc:menu>
<li class="nav-item">
    <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="Privacy">Privacy</a>
</li>
```

## Summary

The lab created the View Component and its view as well as added the Layout.

## Next steps

In the next part of this tutorial series, you will create a custom tag helper