IT2030 Assignment 8 Matt Brown

**ProductListViewModel.cs**

namespace GuitarShop.Models

{

public class ProductListViewModel

{

public List<Category> Categories { get; set; }

public List<Product> Products { get; set; }

public string SelectedCategory { get; set; }

public string CheckActiveCategory(string category) =>

category == SelectedCategory ? "active" : "";

}

}

**ProductController.cs**

namespace GuitarShop.Controllers

{

public class ProductController : Controller

{

private ShopContext context;

public ProductController(ShopContext ctx)

{

context = ctx;

}

public IActionResult Index()

{

return RedirectToAction("List", "Product");

}

[Route("[controller]s/{id?}")]

public IActionResult List(string id = "All")

{

var categories = context.Categories

.OrderBy(c => c.CategoryID).ToList();

List<Product> products;

if (id == "All")

{

products = context.Products

.OrderBy(p => p.ProductID).ToList();

}

else

{

products = context.Products

.Where(p => p.Category.Name == id)

.OrderBy(p => p.ProductID).ToList();

}

var model = new ProductListViewModel

{

Categories = categories,

Products = products,

SelectedCategory = id

};

// bind products to view

return View(model);

}

public IActionResult Details(int id)

{

var categories = context.Categories

.OrderBy(c => c.CategoryID).ToList();

Product product = context.Products.Find(id);

string imageFilename = product.Code + "\_m.png";

// use ViewBag to pass data to view

ViewBag.Categories = categories;

ViewBag.ImageFilename = imageFilename;

// bind product to view

return View(product);

}

}

}

**List.cshtml**

@model ProductListViewModel

@{

ViewBag.Title = "Product List";

}

<h1>Product List</h1>

<div class="row">

<div class="col-sm-2">

<a asp-controller="Product" asp-action="List" asp-route-id="All"

class="list-group-item

@Model.CheckActiveCategory("All")">

All

</a>

@foreach (Category c in Model.Categories)

{

<a asp-controller="Product" asp-action="List" asp-route-id="@c.Name"

class="list-group-item

@Model.CheckActiveCategory(c.Name)">

@c.Name

</a>

}

</div>

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

<table class="table table-bordered table-striped">

<thead>

<tr>

<th>Name</th>

<th>Price</th>

<th></th>

<th></th>

</thead>

<tbody>

@foreach (var product in Model.Products)

{

<tr>

<td>@product.Name</td>

<td>@product.Price.ToString("C")</td>

<td>

<a asp-controller="Product" asp-action="Details"

asp-route-id="@product.ProductID"

asp-route-slug="@product.Slug">View Details</a>

</td>

<td>

<a asp-controller="Cart" asp-action="Add"

asp-route-id="@product.Slug">Add to Cart</a>

</td>

</tr>

}

</tbody>

</table>

</div>

</div>

**ProductController.cs**

namespace GuitarShop.Areas.Admin.Controllers

{

[Area("Admin")]

public class ProductController : Controller

{

private ShopContext context;

private List<Category> categories;

public ProductController(ShopContext ctx)

{

context = ctx;

categories = context.Categories

.OrderBy(c => c.CategoryID)

.ToList();

}

public IActionResult Index()

{

return RedirectToAction("List");

}

[Route("[area]/[controller]s/{id?}")]

public IActionResult List(string id = "All")

{

List<Product> products;

if (id == "All")

{

products = context.Products

.OrderBy(p => p.ProductID).ToList();

}

else

{

products = context.Products

.Where(p => p.Category.Name == id)

.OrderBy(p => p.ProductID).ToList();

}

var model = new ProductListViewModel

{

Categories = categories,

Products = products,

SelectedCategory = id

};

// bind products to view

return View(model);

}

[HttpGet]

public IActionResult Add()

{

// create new Product object

Product product = new Product(); // create Product object

product.Category = context.Categories.Find(1); // add Category object - prevents validation problem

// use ViewBag to pass action and category data to view

ViewBag.Action = "Add";

ViewBag.Categories = categories;

// bind product to AddUpdate view

return View("AddUpdate", product);

}

[HttpGet]

public IActionResult Update(int id)

{

// get Product object for specified primary key

Product product = context.Products

.Include(p => p.Category)

.FirstOrDefault(p => p.ProductID == id);

// use ViewBag to pass action and category data to view

ViewBag.Action = "Update";

ViewBag.Categories = categories;

// bind product to AddUpdate view

return View("AddUpdate", product);

}

[HttpPost]

public IActionResult Update(Product product)

{

if (ModelState.IsValid)

{

string userMessage = "";

if (product.ProductID == 0) // new product

{

context.Products.Add(product);

userMessage = "You just added the product " + product.Name;

}

else // existing product

{

context.Products.Update(product);

userMessage = "You just updated the product " + product.Name;

}

context.SaveChanges();

TempData["UserMessage"] = userMessage;

return RedirectToAction("List");

}

else

{

ViewBag.Action = "Save";

ViewBag.Categories = categories;

return View("AddUpdate", product);

}

}

[HttpGet]

public IActionResult Delete(int id)

{

Product product = context.Products

.FirstOrDefault(p => p.ProductID == id);

return View(product);

}

[HttpPost]

public IActionResult Delete(Product product)

{

context.Products.Remove(product);

context.SaveChanges();

return RedirectToAction("List");

}

}

}