package org.cibertec.edu.pe.controller;

import java.util.Optional;

import java.util.ArrayList;

import java.util.List;

import org.cibertec.edu.pe.model.Detalle;

import org.cibertec.edu.pe.model.Producto;

import org.cibertec.edu.pe.model.Venta;

import org.cibertec.edu.pe.repository.IDetalleRepository;

import org.cibertec.edu.pe.repository.IProductoRepository;

import org.cibertec.edu.pe.repository.IVentaRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.SessionAttributes;

*@Controller*

*@SessionAttributes*({"carrito", "total"})

public class ProductoController {

*@Autowired*

private IProductoRepository productoRepository;

*@Autowired*

private IVentaRepository ventaRepository;

*@Autowired*

private IDetalleRepository detalleRepository;

*@GetMapping*("/index")

public String listado(Model model) {

List<Producto> lista = new ArrayList<>();

lista = productoRepository.findAll();

model.addAttribute("productos", lista);

return "index";

}

*@GetMapping*("/agregar/{idProducto}")

public String agregar(Model model, *@PathVariable*(name = "idProducto", required = true) int idProducto) {

Optional<Producto> optionalProducto = productoRepository.findById(idProducto);

if (optionalProducto.isPresent()) {

Producto producto = optionalProducto.get();

List<Detalle> carrito = (List<Detalle>) model.getAttribute("carrito");

if (carrito == null) {

carrito = new ArrayList<>();

}

boolean productoExistente = false;

for (Detalle detalle : carrito) {

if (detalle.getProducto().getIdProducto() == idProducto) {

detalle.setCantidad(detalle.getCantidad() + 1);

detalle.setSubtotal(detalle.getCantidad() \* detalle.getProducto().getPrecio());

productoExistente = true;

break;

}

}

if (!productoExistente) {

Detalle nuevoDetalle = new Detalle();

nuevoDetalle.setProducto(producto);

nuevoDetalle.setCantidad(1);

nuevoDetalle.setSubtotal(producto.getPrecio());

carrito.add(nuevoDetalle);

}

model.addAttribute("carrito", carrito);

System.***out***.println("Contenido del carrito: " + carrito);

}

return "redirect:/index";

}

*@GetMapping*("/carrito")

public String carrito() {

return "carrito";

}

*@GetMapping*("/pagar")

public String pagar(Model model) {

// Codigo para pagar

return "pagar";

}

*@PostMapping*("/actualizarCarrito")

public String actualizarCarrito(Model model) {

// Codigo para actualizar el carrito

return "carrito";

}

*@ModelAttribute*("carrito")

public List<Detalle> getCarrito() {

return new ArrayList<Detalle>();

}

*@ModelAttribute*("total")

public double getTotal() {

return 0.0;

}

}