**Universidad ORT Uruguay**

Facultad de Ingeniería   
Escuela de Tecnología

Obligatorio 2 - Programación 3

**Rodrigo Suarez   
 303425**

**Grupo: N3G**

**Docente: Pilino Gañi**

**Junio de 2024**

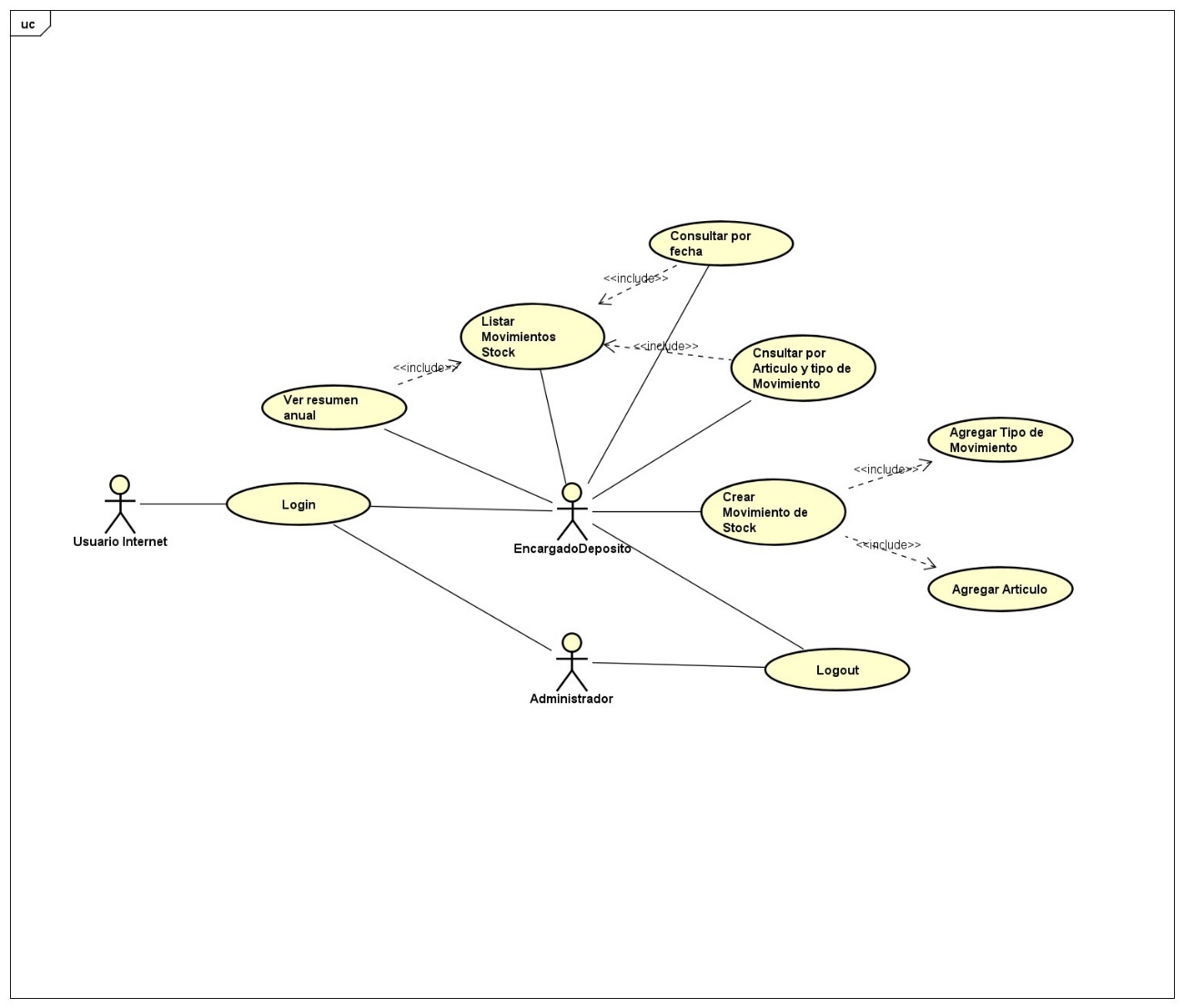
Tabla de contenido

[Diagrama de casos de uso 3](#_Toc169578655)

[Diagrama de clases 4](#_Toc169578656)

[Datos precargados 5](#_Toc169578657)

# Diagrama de casos de uso



# Diagrama Descripción generada automáticamente con confianza mediaDiagrama de clases

# Datos precargados

USE GestionDeposito

GO

SET DATEFORMAT DMY

GO

## -- INSERTS TIPOS DE MOVIMIENTO

INSERT INTO TiposMovimiento (NombreTipoMovimiento\_Valor, EsIncremento)

VALUES ('Venta', 0),

('Devolución', 1),

('Compra', 1)

## -- INSERTS ARTICULOS

/\*PROMPT: puedes crear 80 registros con similares propiedades para una papeleria:

INSERT INTO Articulos (Nombre\_Valor, Descripcion\_Valor, CodigoArticulo\_Valor, Precio\_Valor, Stock\_Valor)

VALUES('Lapicera BIC', 'Caja lapicera azul de uso general', 0000000000001, 876, 20)\*/

INSERT INTO Articulos (Nombre\_Valor, Descripcion\_Valor, CodigoArticulo\_Valor, Precio\_Valor, Stock\_Valor, ParametroId)

VALUES

('Lapicera BIC', 'Caja lapicera azul de uso general', 0000000000001, 876, 100, 1),

('Cuaderno Norma', 'Cuaderno universitario de 100 hojas', 0000000000002, 234, 100, 2),

('Resaltador Faber-Castell', 'Resaltador color amarillo', 0000000000003, 123, 100, 3),

('Borrador Pelikan', 'Borrador para lápiz y tinta', 0000000000004, 45, 100, 4),

('Tijeras Maped', 'Tijeras escolares 13 cm', 0000000000005, 321, 100, 5),

('Pegamento Pritt', 'Pegamento en barra 22g', 0000000000006, 98, 100, 6),

('Marcador Sharpie', 'Marcador permanente negro', 0000000000007, 65, 100, 7),

('Carpeta A4', 'Carpeta tamaño A4 color rojo', 0000000000008, 189, 100, 8),

('Regla Staedtler', 'Regla de 30 cm transparente', 0000000000009, 37, 100, 9),

('Corrector Paper Mate', 'Corrector líquido 20ml', 0000000000010, 56, 100, 10),

('Calculadora Casio', 'Calculadora científica FX-82', 0000000000011, 1290, 100, 11),

('Agenda Moleskine', 'Agenda diaria 2024', 0000000000012, 654, 100, 12),

('Pluma Parker', 'Pluma estilográfica negra', 0000000000013, 8765, 100, 13),

('Cuaderno Scribe', 'Cuaderno profesional 200 hojas', 0000000000014, 320, 100, 14),

('Goma Eva', 'Lámina de goma eva color verde', 0000000000015, 15, 100, 15),

('Cartulina', 'Cartulina blanca tamaño carta', 0000000000016, 5, 100, 16),

('Lápiz de color Prismacolor', 'Caja de 24 lápices de colores', 0000000000017, 789, 100, 17),

('Libro de colorear', 'Libro para colorear infantil', 0000000000018, 120, 100, 18),

('Bolsa para regalo', 'Bolsa de regalo tamaño mediano', 0000000000019, 45, 100, 19),

('Papel crepé', 'Rollo de papel crepé azul', 0000000000020, 12, 100, 20),

('Bolígrafo Pilot', 'Bolígrafo de gel negro', 0000000000021, 34, 100, 21),

('Rotulador Bic', 'Rotulador de pizarra blanca', 0000000000022, 65, 100, 22),

('Estuche Kimmidoll', 'Estuche escolar con diseño', 0000000000023, 210, 100, 23),

('Compás Staedtler', 'Compás escolar metálico', 0000000000024, 98, 100, 24),

('Lápiz mecánico Pentel', 'Lápiz mecánico 0.5 mm', 0000000000025, 76, 100, 25),

('Papel fotocopia', 'Resma de papel A4 500 hojas', 0000000000026, 250, 100, 26),

('Tijeras Titanium', 'Tijeras de acero inoxidable 20 cm', 0000000000027, 180, 100, 27),

('Cuaderno de dibujo', 'Cuaderno de dibujo espiralado', 0000000000028, 150, 100, 28),

('Pegamento UHU', 'Pegamento líquido 33ml', 0000000000029, 110, 100, 29),

('Regla metálica', 'Regla de acero inoxidable 50 cm', 0000000000030, 75, 100, 30),

('Portaminas Faber-Castell', 'Portaminas con goma', 0000000000031, 80, 100, 31),

('Plastilina', 'Caja de plastilina 12 colores', 0000000000032, 70, 100, 32),

('Papel Kraft', 'Rollo de papel kraft 1m', 0000000000033, 45, 100, 33),

('Cuaderno cuadriculado', 'Cuaderno cuadriculado 100 hojas', 0000000000034, 190, 100, 34),

('Sacapuntas metálico', 'Sacapuntas doble uso', 0000000000035, 23, 100, 35),

('Tinta Epson', 'Botella de tinta negra 100ml', 0000000000036, 140, 100, 36),

('Block de notas', 'Block de notas adhesivas', 0000000000037, 65, 100, 37),

('Cuadro de corcho', 'Tablero de corcho 60x90 cm', 0000000000038, 450, 100, 38),

('Carpeta de anillas', 'Carpeta de anillas A4 azul', 0000000000039, 170, 100, 39),

('Cinta adhesiva', 'Rollo de cinta adhesiva 50m', 0000000000040, 30, 100, 40),

('Perforadora', 'Perforadora de 2 agujeros', 0000000000041, 120, 100, 41),

('Grapas', 'Caja de grapas N° 10', 0000000000042, 20, 100, 42),

('Portafolios', 'Portafolios de plástico A4', 0000000000043, 80, 100, 43),

('Papel de regalo', 'Rollo de papel de regalo', 0000000000044, 35, 100, 44),

('Marcador de pizarra', 'Marcador borrable negro', 0000000000045, 55, 100, 45),

('Cinta correctora', 'Cinta correctora de 10m', 0000000000046, 45, 100, 46),

('Pinceles', 'Set de 5 pinceles variados', 0000000000047, 150, 100, 47),

('Hojas para álbum', 'Hojas transparentes A4', 0000000000048, 45, 100, 48),

('Caja organizadora', 'Caja plástica organizadora', 0000000000049, 90, 100, 49),

('Rotulador permanente', 'Rotulador permanente azul', 0000000000050, 65, 100, 50),

('Cuaderno de notas', 'Cuaderno de notas A5', 0000000000051, 110, 100, 51),

('Borrador de pizarra', 'Borrador para pizarra blanca', 0000000000052, 55, 100, 52),

('Rotulador borrable', 'Rotulador de pizarra blanca', 0000000000053, 60, 100, 53),

('Lápiz de grafito', 'Lápiz de grafito HB', 0000000000054, 15, 100, 54),

('Cinta doble cara', 'Cinta adhesiva doble cara', 0000000000055, 35, 100, 55),

('Papel milimetrado', 'Block de papel milimetrado', 0000000000056, 120, 100, 56),

('Set de geometría', 'Set de regla, escuadra y transportador', 0000000000057, 75, 100, 57),

('Archivador', 'Archivador metálico A4', 0000000000058, 300, 100, 58),

('Papel bond', 'Resma de papel bond A4', 0000000000059, 200, 100, 59),

('Lapicera gel', 'Lapicera gel azul', 0000000000060, 30, 70, 60),

('Cuaderno de tapa dura', 'Cuaderno tapa dura 150 hojas', 0000000000061, 250, 100, 61),

('Pegamento blanco', 'Botella de pegamento blanco 120ml', 0000000000062, 60, 100, 62),

('Notas adhesivas', 'Paquete de notas adhesivas', 0000000000063, 55, 100, 63),

('Lápices acuarelables', 'Set de 24 lápices acuarelables', 0000000000064, 450, 100, 64),

('Papel glasé', 'Paquete de papel glasé colores', 0000000000065, 50, 100, 65),

('Portadocumentos', 'Portadocumentos A4 con cierre', 0000000000066, 80, 100, 66),

('Láminas de dibujo', 'Paquete de láminas A3', 0000000000067, 90, 100, 67),

('Rotulador textil', 'Rotulador para tela negro', 0000000000068, 45, 100, 68),

('Carpeta colgante', 'Carpeta colgante verde', 0000000000069, 65, 100, 69),

('Marcador fluorescente', 'Marcador fluorescente rosa', 0000000000070, 30, 100, 70),

('Notas autoadhesivas', 'Notas autoadhesivas 76x76 mm', 0000000000071, 35, 100, 71),

('Tablero magnético', 'Tablero magnético 45x60 cm', 0000000000072, 320, 100, 72),

('Goma moldeable', 'Goma moldeable blanca', 0000000000073, 80, 100, 73),

('Rotulador de punta fina', 'Rotulador punta fina azul', 0000000000074, 25, 100, 74),

('Carpeta escolar', 'Carpeta escolar tamaño oficio', 0000000000075, 70, 100, 75),

('Portaminas Pentel', 'Portaminas 0.7 mm', 0000000000076, 65, 100, 76),

('Papel adhesivo', 'Rollo de papel adhesivo', 0000000000077, 50, 100, 77),

('Bolígrafo bic', 'Bolígrafo azul de punta fina', 0000000000078, 15, 100, 78),

('Papel fotográfico', 'Paquete de papel fotográfico A4', 0000000000079, 300, 100, 79),

('Caja archivadora', 'Caja archivadora de cartón', 0000000000080, 100, 100, 80);

## -- INSERTS PARAMETRO DE ARTICULOS

/\*

PROMP: Puedes generar 80 registros con el siguiente formato: INSERT INTO ParametroArticulos (ArticuloId, TopeMovimiento\_Valor)

VALUES(1, 5)

El ArticuloId debe ir del 1 al 80 y el TopeMovimiento\_valor debe variar entre 1 y 20

\*/

INSERT INTO Parametros(Nombre, Valor\_Valor)

VALUES

('Producto1', 4),

('Producto2', 8),

('Producto3', 7),

('Producto4', 12),

('Producto5', 2),

('Producto6', 15),

('Producto7', 5),

('Producto8', 12),

('Producto9', 15),

('Producto10', 13),

('Producto11', 4),

('Producto12', 19),

('Producto13', 18),

('Producto14', 12),

('Producto15', 4),

('Producto16', 4),

('Producto17', 12),

('Producto18', 15),

('Producto19', 5),

('Producto20', 1),

('Producto21', 10),

('Producto22', 18),

('Producto23', 15),

('Producto24', 15),

('Producto25', 2),

('Producto26', 19),

('Producto27', 2),

('Producto28', 11),

('Producto29', 4),

('Producto30', 14),

('Producto31', 16),

('Producto32', 7),

('Producto33', 9),

('Producto34', 3),

('Producto35', 9),

('Producto36', 17),

('Producto37', 3),

('Producto38', 18),

('Producto39', 17),

('Producto40', 18),

('Producto41', 8),

('Producto42', 5),

('Producto43', 12),

('Producto44', 13),

('Producto45', 1),

('Producto46', 20),

('Producto47', 3),

('Producto48', 20),

('Producto49', 13),

('Producto50', 7),

('Producto51', 12),

('Producto52', 8),

('Producto53', 7),

('Producto54', 20),

('Producto55', 7),

('Producto56', 9),

('Producto57', 13),

('Producto58', 6),

('Producto59', 16),

('Producto60', 16),

('Producto61', 16),

('Producto62', 14),

('Producto63', 15),

('Producto64', 8),

('Producto65', 15),

('Producto66', 4),

('Producto67', 12),

('Producto68', 6),

('Producto69', 20),

('Producto70', 8),

('Producto71', 16),

('Producto72', 6),

('Producto73', 6),

('Producto74', 11),

('Producto75', 12),

('Producto76', 2),

('Producto77', 8),

('Producto78', 19),

('Producto79', 4),

('Producto80', 4);

## -- INSERTS USUARIOS

/\*PROMPT: Podrias crear 6 inserts diferentes con propiedades similares a el siguiente: INSERT INTO Usuarios (Email\_Valor, Nombre\_Valor, Apellido\_Valor, Contrasenia\_Valor, TipoUsuario)

VALUES('admin@algo.com', 'Juan', 'Rodriguez', '12345', 'EncargadoDeposito')\*/

INSERT INTO Usuarios (Email\_Valor, Nombre\_Valor, Apellido\_Valor, Contrasenia\_Valor, TipoUsuario)

VALUES ('usuario@algo.com', 'Juan', 'Rodriguez', '12345', 'EncargadoDeposito'),

('usuario1@algo.com', 'Pedro', 'González', 'abcde', 'EncargadoDeposito'),

('usuario2@algo.com', 'María', 'López', 'qwerty', 'EncargadoDeposito'),

('admin@algo.com', 'Carlos', 'Martínez', '123456', 'Administrador'),

('admin1@algo.com', 'Laura', 'Fernández', 'abcdef', 'Administrador'),

('admin2@algo.com', 'Ana', 'Sánchez', 'password', 'Administrador');

## -- INSERTS MOVIMIENTO STOCK

/\*PROMPT: Puedes generarme 40 registros similares al siguiente: INSERT INTO MovimientosStock (Cantidad\_Valor, UsuarioId, ArticuloId, Fecha, TipoMovimientoId)

VALUES(4, 4, 1, '11-06-2024', 1004). Teniendo en cuenta que el UsuarioId debe estar entre 4 y6. Ademas los ArticulosId deben estar entre 1 y 81. La fecha maximia debes tomar el dia de hoy y los tipos de movimiento son los siguientes: INSERT INTO TiposMovimiento (NombreTipoMovimiento\_Valor, EsIncremento)

VALUES ('VENTA', 0),

('DEVOLUCION', 1),

('COMPRA', 1). Todos los articulos tienen un stock de 100 y un tope por movimiento: INSERT INTO ParametroArticulos (ArticuloId, TopeMovimiento\_Valor)

VALUES (1, 4),....

(79, 4),

(80, 4);

INSERT INTO TiposMovimiento (NombreTipoMovimiento\_Valor, EsIncremento)

VALUES ('VENTA', 0),

('DEVOLUCION', 1),

('COMPRA', 1) \*/

INSERT INTO MovimientosStock (Cantidad\_Valor, UsuarioEmail, ArticuloId, Fecha, TipoMovimientoId)

VALUES (4, 'usuario@algo.com', 1, '2024-06-01', 1),

(8, 'usuario@algo.com', 2, '2024-06-02', 2),

(7, 'usuario@algo.com', 3, '2024-06-03', 3),

(12, 'usuario@algo.com', 4, '2024-06-04', 1),

(2, 'usuario@algo.com', 5, '2024-06-05', 2),

(15, 'usuario@algo.com', 6, '2024-06-06', 3),

(5, 'usuario@algo.com', 7, '2024-06-07', 1),

(12, 'usuario@algo.com', 8, '2024-06-08', 2),

(15, 'usuario@algo.com', 9, '2024-06-09', 3),

(13, 'usuario@algo.com', 10, '2024-06-10', 1),

(4, 'usuario@algo.com', 11, '2024-06-11', 2),

(19, 'usuario@algo.com', 12, '2024-06-01', 3),

(18, 'usuario@algo.com', 13, '2024-06-02', 1),

(12, 'usuario@algo.com', 14, '2024-06-03', 2),

(4, 'usuario@algo.com', 15, '2024-06-04', 3),

(4, 'usuario@algo.com', 16, '2024-06-05', 1),

(12, 'usuario1@algo.com', 17, '2024-06-06', 2),

(15, 'usuario1@algo.com', 18, '2024-06-07', 3),

(5, 'usuario1@algo.com', 19, '2024-06-08', 1),

(1, 'usuario1@algo.com', 20, '2024-06-09', 2),

(10, 'usuario1@algo.com', 21, '2024-06-10', 3),

(18, 'usuario1@algo.com', 22, '2024-06-11', 1),

(15, 'usuario1@algo.com', 23, '2024-06-01', 2),

(15, 'usuario1@algo.com', 24, '2024-06-02', 3),

(2, 'usuario1@algo.com', 25, '2024-06-03', 1),

(19, 'usuario1@algo.com', 26, '2024-06-04', 2),

(2, 'usuario1@algo.com', 27, '2024-06-05', 3),

(11, 'usuario1@algo.com', 28, '2024-06-06', 1),

(4, 'usuario1@algo.com', 29, '2024-06-07', 2),

(14, 'usuario1@algo.com', 30, '2024-06-08', 3),

(16, 'usuario2@algo.com', 31, '2024-06-09', 1),

(7, 'usuario2@algo.com', 32, '2024-06-10', 2),

(9, 'usuario2@algo.com', 33, '2024-06-11', 3),

(3, 'usuario2@algo.com', 34, '2024-06-01', 1),

(9, 'usuario2@algo.com', 35, '2024-06-02', 2),

(17, 'usuario2@algo.com', 36, '2024-06-03', 3),

(3, 'usuario2@algo.com', 37, '2024-06-04', 1),

(18, 'usuario2@algo.com', 38, '2024-06-05', 2),

(17, 'usuario2@algo.com', 39, '2024-06-06', 3),

(18, 'usuario2@algo.com', 40, '2024-06-07', 1);

## --Parámetro para paginación

INSERT INTO Parametros VALUES('Paginacion', 2)

# Código

## MVC

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\ArticuloDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class ArticuloDTO

{

public int Id { get; set; }

public string Nombre { get; set; }

public string Descripcion { get; set; }

public long CodigoArticulo { get; set; }

public decimal Precio { get; set; }

public int Stock { get; set; }

public int TopeDeArticuloPorMovimiento { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\LoginDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class LoginDTO

{

public string Email { get; set; }

public string Contrasena { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\MovimientoStockDTO.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class MovimientoStockDTO

{

public int Id { get; set; }

[DisplayName("Fecha de Movimiento")]

public DateTime FechaMovimiento { get; set; }

public int Cantidad { get; set; }

[DisplayName("Articulo")]

public string ArticuloNombre { get; set; }

[DisplayName("Email Usuario")]

public string UsuarioEmail { get; set; }

[DisplayName("Tipo de Movimiento")]

public string TipoMovimientoNombre { get; set; }

public int ArticuloId { get; set; }

public int TipoMovimientoId { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\TipoMovimientoDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class TipoMovimientoDTO

{

public int Id { get; set; }

public string Nombre { get; set; }

public bool EsIncremento { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\UsuarioDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class UsuarioDTO

{

public string Email { get; set; }

public string Contrasena { get; set; }

public string Nombre { get; set; }

public string Apellido { get; set; }

public string ContrasenaEncriptada { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\UsuarioLogueadoDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class UsuarioLogueadoDTO

{

public string EmailUsuario { get; set; }

public string RolUsuario { get; set; }

public string Token { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\obj\Debug\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\obj\Debug\net8.0\DTOs.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("DTOs")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("DTOs")]

[assembly: System.Reflection.AssemblyTitleAttribute("DTOs")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\obj\Debug\net8.0\DTOs.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\obj\Release\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\obj\Release\net8.0\DTOs.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("DTOs")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Release")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("DTOs")]

[assembly: System.Reflection.AssemblyTitleAttribute("DTOs")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\DTOs\obj\Release\net8.0\DTOs.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\Program.cs

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.

builder.Services.AddControllersWithViews();

builder.Services.AddSession();

var app = builder.Build();

// Configure the HTTP request pipeline.

if (!app.Environment.IsDevelopment())

{

app.UseExceptionHandler("/Home/Error");

// The default HSTS value is 30 days. You may want to change this for production scenarios, see https://aka.ms/aspnetcore-hsts.

app.UseHsts();

}

app.UseHttpsRedirection();

app.UseStaticFiles();

app.UseRouting();

app.UseSession();

app.UseAuthorization();

app.MapControllerRoute(

name: "default",

pattern: "{controller=Usuarios}/{action=Login}/{id?}");

app.Run();

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\ClasesAuxiliares\HerramientasAPI.cs

using DTOs;

using Newtonsoft.Json;

using System.Text.Json;

using WebMVC.Controllers;

namespace WebMVC.ClasesAuxiliares

{

public class HerramientasAPI

{

public static string LeerContenidoRespuesta(HttpResponseMessage respuesta)

{

HttpContent content = respuesta.Content;

var tarea2 = content.ReadAsStringAsync();

tarea2.Wait();

string cuerpo = tarea2.Result;

return cuerpo;

}

public static UsuarioDTO CrearUsuarioDTO(HttpResponseMessage respuesta)

{

HttpContent content = respuesta.Content;

var tarea2 = content.ReadAsStringAsync();

tarea2.Wait();

string cuerpo = tarea2.Result;

if (string.IsNullOrEmpty(cuerpo))

{

throw new Exception("El contenido de la respuesta está vacío.");

}

UsuarioDTO usuario = JsonConvert.DeserializeObject<UsuarioDTO>(cuerpo);

return usuario;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\Controllers\MovimientoStockController.cs

using DTOs;

using WebMVC.Filtro;

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Mvc;

using Newtonsoft.Json;

using System.Net.Http.Headers;

using System.Net.Http;

using WebMVC.ClasesAuxiliares;

using WebMVC.Models;

using Humanizer;

using System.Reflection;

using NuGet.Protocol;

using System.Linq.Expressions;

namespace WebMVC.Controllers

{

public class MovimientoStockController : Controller

{

public string UrlApi { get; set; }

public HttpClient client = new HttpClient();

public MovimientoStockController(IConfiguration config)

{

UrlApi = config.GetValue<string>("UrlAPI");

}

// GET: MovimientoStockController/ListarMovimientosStock

[HttpGet]

[EncargadoDeposito]

public ActionResult ListarMovimientosStock()

{

IEnumerable<MovimientoStockDTO> movimientos = new List<MovimientoStockDTO>();

try

{

string url = UrlApi + "MovimientoStock";

client.DefaultRequestHeaders.Authorization = new System.Net.Http.Headers.AuthenticationHeaderValue("Bearer",

HttpContext.Session.GetString("Token"));

var tarea1 = client.GetAsync(url);

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

string cuerpo = HerramientasAPI.LeerContenidoRespuesta(respuesta);

movimientos = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(cuerpo);

}

}

catch

{

ViewBag.error = "No se encontraron movimientos para mostrar";

}

return View(movimientos);

}

// GET: MovimientoStockController/Create

[HttpGet]

[EncargadoDeposito]

public ActionResult Create()

{

AltaMovimientoStockViewModel vm = new AltaMovimientoStockViewModel();

vm.ArticuloDTOs = ArticuloDTOs();

vm.TiposMovimientoDTO = TipoMovimientoDTOs();

return View(vm);

}

// POST: MovimientoStockController/Create

[HttpPost]

[ValidateAntiForgeryToken]

[EncargadoDeposito]

public ActionResult Create(AltaMovimientoStockViewModel vm)

{

try

{

MovimientoStockDTO mvDTO = new MovimientoStockDTO()

{

UsuarioEmail = HttpContext.Session.GetString("EmailUsuario"),

Cantidad = vm.Cantidad,

TipoMovimientoId = vm.TipoMovimientoId,

ArticuloId = vm.ArticuloId,

};

string url = UrlApi + "MovimientoStock";

client.DefaultRequestHeaders.Authorization = new System.Net.Http.Headers.AuthenticationHeaderValue("Bearer",

HttpContext.Session.GetString("Token"));

var tarea1 = client.PostAsJsonAsync(url, mvDTO);

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

return RedirectToAction("ListarMovimientosStock");

}

else if ((int)respuesta.StatusCode == StatusCodes.Status400BadRequest

|| (int)respuesta.StatusCode == StatusCodes.Status500InternalServerError)

{

ViewBag.error = respuesta.Content.ReadAsStringAsync().Result;

}

}

catch { }

vm.TiposMovimientoDTO = TipoMovimientoDTOs();

vm.ArticuloDTOs = ArticuloDTOs();

return View(vm);

}

[HttpGet]

public IEnumerable<ArticuloDTO> ArticuloDTOs()

{

IEnumerable<ArticuloDTO> articulos = new List<ArticuloDTO>();

string url = UrlApi + "Articulo";

var tarea1 = client.GetAsync(url);

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

string cuerpo = HerramientasAPI.LeerContenidoRespuesta(respuesta);

articulos = JsonConvert.DeserializeObject<IEnumerable<ArticuloDTO>>(cuerpo);

}

return articulos;

}

[HttpGet]

public IEnumerable<TipoMovimientoDTO> TipoMovimientoDTOs()

{

IEnumerable<TipoMovimientoDTO> tipoMovimientos = new List<TipoMovimientoDTO>();

string url = UrlApi + "TipoMovimientos";

var tarea1 = client.GetAsync(url);

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

string cuerpo = HerramientasAPI.LeerContenidoRespuesta(respuesta);

tipoMovimientos = JsonConvert.DeserializeObject<IEnumerable<TipoMovimientoDTO>>(cuerpo);

}

return tipoMovimientos;

}

[HttpGet]

[EncargadoDeposito]

public ActionResult ConsultarMovimientosPorFecha()

{

var fechaActual = DateTime.Now;

ViewBag.FechaActual = fechaActual.ToString("yyyy-MM-dd HH:mm:ss");

return View(new MovimientoConsultaViewModel());

}

[HttpPost]

[EncargadoDeposito]

public ActionResult ConsultarMovimientosPorFecha(MovimientoConsultaViewModel consulta, int page)

{

if (consulta.FechaInicio > consulta.FechaFin)

{

ViewBag.Mensaje = "Fecha de Inicio siempre debe ser menor a Fecha de Fin";

return View();

}

try

{

IEnumerable<MovimientoStockDTO> movimientosTotales = new List<MovimientoStockDTO>();

IEnumerable<MovimientoStockDTO> movimientosPagina = new List<MovimientoStockDTO>();

HttpClient client = new HttpClient();

client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("Bearer", HttpContext.Session.GetString("Token"));

string url = UrlApi + "MovimientoStock";

var fechaInicioSinHora = consulta.FechaInicio.ToString("yyyy-MM-dd");

var fechaFinSinHora = consulta.FechaFin.Date.ToString("yyyy-MM-dd");

var tarea1 = client.GetAsync(url + $"/fechaInicio{fechaInicioSinHora}/fechaFin{fechaFinSinHora}");

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

var contenido = HerramientasAPI.LeerContenidoRespuesta(respuesta);

movimientosTotales = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(contenido);

ViewBag.Paginas = ObtenerCantidadPaginas(movimientosTotales);

try

{

var tarea2 = client.GetAsync(url + $"/fechaInicio{fechaInicioSinHora}/fechaFin{fechaFinSinHora}/pagina{page}");

tarea2.Wait();

var respuesta2 = tarea2.Result;

if (respuesta2.IsSuccessStatusCode)

{

var contenido2 = HerramientasAPI.LeerContenidoRespuesta(respuesta2);

movimientosPagina = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(contenido2);

MovimientoConsultaViewModel nuevoVM = new MovimientoConsultaViewModel();

nuevoVM.Articulos = ArticuloDTOs();

nuevoVM.TiposMovimiento = TipoMovimientoDTOs();

nuevoVM.Movimientos = movimientosPagina;

nuevoVM.FechaInicio = consulta.FechaInicio;

nuevoVM.FechaFin = consulta.FechaFin;

return View(nuevoVM);

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error al obtener los movimientos por pagina";

}

}

catch (Exception) {

ViewBag.Mensaje = "Ha ocurrido un error en el servidor";

}

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error al obtener los movimientos";

}

}

catch (Exception ex)

{

ViewBag.Mensaje = "Ha ocurrido un error en el servidor";

}

return View(consulta);

}

[EncargadoDeposito]

[HttpGet]

public ActionResult ConsultarMovimientosPorFechaQuery(string fechaInicio,string fechaFin, int pagina)

{

DateTime FechaInicio = DateTime.Parse(fechaInicio);

DateTime FechaFin = DateTime.Parse(fechaFin);

if (FechaInicio > FechaFin)

{

ViewBag.Mensaje = "Fecha de Inicio siempre debe ser menor a Fecha de Fin";

return View();

}

try

{

IEnumerable<MovimientoStockDTO> movimientosTotales = new List<MovimientoStockDTO>();

IEnumerable<MovimientoStockDTO> movimientosPagina = new List<MovimientoStockDTO>();

HttpClient client = new HttpClient();

client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("Bearer", HttpContext.Session.GetString("Token"));

string url = UrlApi + "MovimientoStock";

var fechaInicioSinHora = FechaInicio.ToString("yyyy-MM-dd");

var fechaFinSinHora = FechaFin.Date.ToString("yyyy-MM-dd");

var tarea1 = client.GetAsync(url + $"/fechaInicio{fechaInicioSinHora}/fechaFin{fechaFinSinHora}");

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

var contenido = HerramientasAPI.LeerContenidoRespuesta(respuesta);

movimientosTotales = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(contenido);

ViewBag.Paginas = ObtenerCantidadPaginas(movimientosTotales);

try

{

var tarea2 = client.GetAsync(url + $"/fechaInicio{fechaInicioSinHora}/fechaFin{fechaFinSinHora}/pagina{pagina}");

tarea2.Wait();

var respuesta2 = tarea2.Result;

if (respuesta2.IsSuccessStatusCode)

{

var contenido2 = HerramientasAPI.LeerContenidoRespuesta(respuesta2);

movimientosPagina = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(contenido2);

MovimientoConsultaViewModel nuevoVM = new MovimientoConsultaViewModel();

nuevoVM.Articulos = ArticuloDTOs();

nuevoVM.TiposMovimiento = TipoMovimientoDTOs();

nuevoVM.Movimientos = movimientosPagina;

nuevoVM.FechaInicio = FechaInicio;

nuevoVM.FechaFin = FechaFin;

return View(nuevoVM);

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error al obtener los movimientos por pagina";

}

}

catch (Exception)

{

ViewBag.Mensaje = "Ha ocurrido un error en el servidor";

}

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error al obtener los movimientos";

}

}

catch (Exception ex)

{

ViewBag.Mensaje = "Ha ocurrido un error en el servidor";

}

return View();

}

[HttpGet]

[EncargadoDeposito]

public ActionResult ConsultarMovimientosPorArticuloYTipo()

{

MovimientoConsultaViewModel vm = new MovimientoConsultaViewModel();

vm.Articulos = ArticuloDTOs();

vm.TiposMovimiento = TipoMovimientoDTOs();

return View(vm);

}

[HttpPost]

[EncargadoDeposito]

public ActionResult ConsultarMovimientosPorArticuloYTipo(MovimientoConsultaViewModel consulta, int page)

{

if (page == null || page == 0)

{

page = 1;

}

try

{

IEnumerable<MovimientoStockDTO> movimientosTotales = new List<MovimientoStockDTO>();

IEnumerable<MovimientoStockDTO> movimientosPagina = new List<MovimientoStockDTO>();

HttpClient client = new HttpClient();

client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("Bearer", HttpContext.Session.GetString("Token"));

var tarea1 = client.GetAsync(UrlApi + $"MovimientoStock/articulo/{consulta.ArticuloId}/tipo/{consulta.MovimientoId}");

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

var contenido = HerramientasAPI.LeerContenidoRespuesta(respuesta);

movimientosTotales = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(contenido);

ViewBag.Paginas = ObtenerCantidadPaginas(movimientosTotales);

try

{

var tarea2 = client.GetAsync(UrlApi + $"MovimientoStock/articulo/{consulta.ArticuloId}/tipo/{consulta.MovimientoId}/pagina{page}");

tarea2.Wait();

var respuesta2 = tarea2.Result;

if (respuesta2.IsSuccessStatusCode)

{

var contenido2 = HerramientasAPI.LeerContenidoRespuesta(respuesta2);

movimientosPagina = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(contenido2);

MovimientoConsultaViewModel nuevoVM = new MovimientoConsultaViewModel();

nuevoVM.Articulos = ArticuloDTOs();

nuevoVM.TiposMovimiento = TipoMovimientoDTOs();

nuevoVM.Movimientos = movimientosPagina;

nuevoVM.ArticuloId = consulta.ArticuloId;

nuevoVM.MovimientoId = consulta.MovimientoId;

return View(nuevoVM);

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error al obtener los movimientos por pagina";

}

}

catch (Exception ex)

{

}

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error al obtener los movimientos";

}

}

catch (Exception ex)

{

ViewBag.Mensaje = "Ha ocurrido un error en el servidor. Inténtelo mas tarde";

}

consulta.Articulos = ArticuloDTOs();

consulta.TiposMovimiento = TipoMovimientoDTOs();

return View(consulta);

}

[HttpGet]

[EncargadoDeposito]

public ActionResult ConsultarMovimientosPorArticuloYTipoQuery(int articulo, int tipo, int pagina)

{

MovimientoConsultaViewModel consulta = new MovimientoConsultaViewModel();

try

{

IEnumerable<MovimientoStockDTO> movimientosTotales = new List<MovimientoStockDTO>();

IEnumerable<MovimientoStockDTO> movimientosPagina = new List<MovimientoStockDTO>();

HttpClient client = new HttpClient();

client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("Bearer", HttpContext.Session.GetString("Token"));

var tarea1 = client.GetAsync(UrlApi + $"MovimientoStock/articulo/{articulo}/tipo/{tipo}");

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

var contenido = HerramientasAPI.LeerContenidoRespuesta(respuesta);

movimientosTotales = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(contenido);

ViewBag.Paginas = ObtenerCantidadPaginas(movimientosTotales);

try

{

var tarea2 = client.GetAsync(UrlApi + $"MovimientoStock/articulo/{articulo}/tipo/{tipo}/pagina{pagina}");

tarea2.Wait();

var respuesta2 = tarea2.Result;

if (respuesta2.IsSuccessStatusCode)

{

var contenido2 = HerramientasAPI.LeerContenidoRespuesta(respuesta2);

movimientosPagina = JsonConvert.DeserializeObject<IEnumerable<MovimientoStockDTO>>(contenido2);

MovimientoConsultaViewModel nuevoVM = new MovimientoConsultaViewModel();

nuevoVM.Articulos = ArticuloDTOs();

nuevoVM.TiposMovimiento = TipoMovimientoDTOs();

nuevoVM.Movimientos = movimientosPagina;

nuevoVM.ArticuloId = articulo;

nuevoVM.MovimientoId = tipo;

//return RedirectToAction("ConsultarMovimientosPorArticuloYTipo", nuevoVM);

return View(nuevoVM);

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error al obtener los movimientos por pagina";

}

}

catch (Exception ex)

{

}

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error al obtener los movimientos";

}

}

catch (Exception ex)

{

ViewBag.Mensaje = "Ha ocurrido un error en el servidor. Inténtelo mas tarde";

}

consulta.Articulos = ArticuloDTOs();

consulta.TiposMovimiento = TipoMovimientoDTOs();

return View(consulta);

}

[HttpGet]

[EncargadoDeposito]

public ActionResult ConsultarResumenAnual()

{

try

{

var client = new HttpClient();

string url = UrlApi + "MovimientoStock/resumen";

client.DefaultRequestHeaders.Authorization = new System.Net.Http.Headers.AuthenticationHeaderValue("Bearer",

HttpContext.Session.GetString("Token"));

var tarea1 = client.GetAsync(url);

tarea1.Wait();

var respuesta = tarea1.Result;

if (respuesta.IsSuccessStatusCode)

{

var contenido = HerramientasAPI.LeerContenidoRespuesta(respuesta);

if (contenido.StartsWith("{") || contenido.StartsWith("["))

{

var resumen = JsonConvert.DeserializeObject<string>(contenido);

return View("ConsultarResumenAnual", resumen);

}

else

{

return View("ConsultarResumenAnual", contenido);

}

}

else

{

ViewBag.Mensaje = "No se encontraron movimientos para mostrar";

return View();

}

}

catch (Exception ex)

{

ViewBag.Mensaje = "Ocurrió un error en el servidor, no se pudo obtener el resumen";

}

return View();

}

public double ObtenerCantidadPaginas(IEnumerable<MovimientoStockDTO> lista)

{

double cantidadPaginas = 0;

try

{

HttpClient cliente = new HttpClient();

cliente.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("Bearer", HttpContext.Session.GetString("Token"));

var tarea = cliente.GetAsync(UrlApi + $"MovimientoStock/cantidadPaginas/{lista.Count()}");

var respuesta = tarea.Result;

if (respuesta.IsSuccessStatusCode)

{

var contenido = HerramientasAPI.LeerContenidoRespuesta(respuesta);

contenido = contenido.Replace(".", ",");

double.TryParse(contenido, out cantidadPaginas);

return cantidadPaginas;

}

else if ((int)respuesta.StatusCode == StatusCodes.Status400BadRequest || (int)respuesta.StatusCode == StatusCodes.Status500InternalServerError)

{

cantidadPaginas = -1;

}

}

catch (Exception ex)

{

throw;

}

return cantidadPaginas;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\Controllers\UsuariosController.cs

using DTOs;

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Mvc;

using Newtonsoft.Json;

using WebMVC.ClasesAuxiliares;

namespace WebMVC.Controllers

{

public class UsuariosController : Controller

{

public string UrlApi { get; set; }

public UsuariosController(IConfiguration config)

{

UrlApi = config.GetValue<string>("UrlAPI") + "Usuarios/";

}

[HttpGet]

public ActionResult Login()

{

return View();

}

[HttpPost]

public IActionResult Login(LoginDTO usuario)

{

try

{

if (ModelState.IsValid)

{

HttpClient client = new HttpClient();

string url = UrlApi + "Login";

var tarea = client.PostAsJsonAsync(url, usuario);

tarea.Wait();

var respuesta = tarea.Result;

string cuerpo = HerramientasAPI.LeerContenidoRespuesta(respuesta);

if (respuesta.IsSuccessStatusCode)

{

UsuarioLogueadoDTO usuarioLogueado = JsonConvert.DeserializeObject<UsuarioLogueadoDTO>(cuerpo);

if (usuarioLogueado != null)

{

HttpContext.Session.SetString("EmailUsuario", usuarioLogueado.EmailUsuario);

HttpContext.Session.SetString("RolUsuario", usuarioLogueado.RolUsuario);

HttpContext.Session.SetString("Token", usuarioLogueado.Token);

return RedirectToAction("ListarMovimientosStock", "MovimientoStock");

}

ViewBag.Mensaje = "Email o Contraseña incorrectos";

}

else if ((int)respuesta.StatusCode == StatusCodes.Status400BadRequest

|| (int)respuesta.StatusCode == StatusCodes.Status404NotFound

|| (int)respuesta.StatusCode == StatusCodes.Status500InternalServerError)

{

ViewBag.Mensaje = cuerpo;

}

else

{

ViewBag.Mensaje = "Ha ocurrido un error";

}

}

else

{

ViewBag.Mensaje = "Error, debe ingresar Email y Contraseña";

}

}

catch (Exception ex)

{

ViewBag.Mensaje = "Error interno. Inténtelo mas tarde";

}

return View("Login");

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Logout()

{

try

{

string token = HttpContext.Session.GetString("Token");

if (!string.IsNullOrEmpty(token))

{

HttpContext.Session.Clear();

return RedirectToAction("Login");

}

else

{

ViewBag.Mensaje = "Error al cerrar sesión";

return View("Index");

}

}

catch (Exception)

{

ViewBag.Mensaje = "Error interno. Inténtelo más tarde.";

return View("Index");

}

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\Filtro\EncargadoDeposito.cs

using Microsoft.AspNetCore.Mvc.Filters;

using Microsoft.AspNetCore.Mvc;

namespace WebMVC.Filtro

{

public class EncargadoDeposito: Attribute, IAuthorizationFilter

{

public void OnAuthorization(AuthorizationFilterContext context)

{

string rol = context.HttpContext.Session.GetString("RolUsuario");

if (string.IsNullOrEmpty(rol))

{

context.Result = new RedirectResult("/Usuarios/Login");

}

else if (rol != "EncargadoDeposito")

{

context.Result = new RedirectResult("/Usuarios/Login");

}

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\Models\AltaMovimientoStockViewModel.cs

using DTOs;

namespace WebMVC.Models

{

public class AltaMovimientoStockViewModel

{

public IEnumerable<ArticuloDTO> ArticuloDTOs = new List<ArticuloDTO>();

public IEnumerable<TipoMovimientoDTO> TiposMovimientoDTO { get; set; }

public int ArticuloId { get; set; }

public int TipoMovimientoId { get; set; }

public int Cantidad { get; set; }

public string EmailUsuario { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\Models\ErrorViewModel.cs

namespace WebMVC.Models

{

public class ErrorViewModel

{

public string? RequestId { get; set; }

public bool ShowRequestId => !string.IsNullOrEmpty(RequestId);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\Models\MovimientoConsultaViewModel.cs

using DTOs;

namespace WebMVC.Models

{

public class MovimientoConsultaViewModel

{

public DateTime FechaInicio { get; set; }

public DateTime FechaFin { get; set; }

public int ArticuloId { get; set; }

public int MovimientoId { get; set; }

public IEnumerable<ArticuloDTO> Articulos { get; set; }

public IEnumerable<TipoMovimientoDTO> TiposMovimiento { get; set; }

public IEnumerable <MovimientoStockDTO> Movimientos { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\Models\MovimientoStockViewModel.cs

using System.ComponentModel;

namespace WebMVC.Models

{

public class MovimientoStockViewModel

{

public int Id { get; set; }

[DisplayName("Fecha de Movimiento")]

public DateTime FechaMovimiento { get; set; }

public int Cantidad { get; set; }

[DisplayName("Articulo")]

public int ArticuloId { get; set; }

[DisplayName("Email Usuario")]

public string UsuarioEmail { get; set; }

[DisplayName("Tipo de Movimiento")]

public int TipoMovimientoId { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\obj\Debug\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\obj\Debug\net8.0\WebMVC.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("WebMVC")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("WebMVC")]

[assembly: System.Reflection.AssemblyTitleAttribute("WebMVC")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\obj\Debug\net8.0\WebMVC.GlobalUsings.g.cs

// <auto-generated/>

global using global::Microsoft.AspNetCore.Builder;

global using global::Microsoft.AspNetCore.Hosting;

global using global::Microsoft.AspNetCore.Http;

global using global::Microsoft.AspNetCore.Routing;

global using global::Microsoft.Extensions.Configuration;

global using global::Microsoft.Extensions.DependencyInjection;

global using global::Microsoft.Extensions.Hosting;

global using global::Microsoft.Extensions.Logging;

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Net.Http.Json;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebMVC\WebMVC\obj\Debug\net8.0\WebMVC.RazorAssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: Microsoft.AspNetCore.Mvc.ApplicationParts.ProvideApplicationPartFactoryAttribute("Microsoft.AspNetCore.Mvc.ApplicationParts.ConsolidatedAssemblyApplicationPartFact" +

"ory, Microsoft.AspNetCore.Mvc.Razor")]

## API

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\ArticuloDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class ArticuloDTO

{

public int Id { get; set; }

public string Nombre { get; set; }

public string Descripcion { get; set; }

public long CodigoArticulo { get; set; }

public decimal Precio { get; set; }

public int Stock { get; set; }

public int TopeDeArticuloPorMovimiento { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\LoginDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class LoginDTO

{

public string Email { get; set; }

public string Contrasena { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\MovimientoStockDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class MovimientoStockDTO

{

public int Id { get; set; }

public DateTime FechaMovimiento { get; set; }

public int Cantidad { get; set; }

public string? ArticuloNombre { get; set; }

public string? TipoMovimientoNombre { get; set; }

public string UsuarioEmail { get; set; }

public int ArticuloId { get; set; }

public int TipoMovimientoId { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\MovimientoStockListDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class MovimientoStockListDTO

{

public int Id { get; set; }

public DateTime? FechaMovimiento { get; set; }

public int Cantidad { get; set; }

public string ArticuloNombre { get; set; }

public string UsuarioEmail { get; set; }

public string TipoMovimientoNombre { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\TipoMovimientoDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class TipoMovimientoDTO

{

public int Id { get; set; }

public string Nombre { get; set; }

public bool EsIncremento { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\UsuarioDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class UsuarioDTO

{

public string Email { get; set; }

public string Contrasena { get; set; }

public string Nombre { get; set; }

public string Apellido { get; set; }

public string ContrasenaEncriptada { get; set; }

public string Rol { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\UsuarioLogueadoDTO.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs

{

public class UsuarioLogueadoDTO

{

public string EmailUsuario { get; set; }

public string RolUsuario { get; set; }

public string Token { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\MappersDTOs\MapperArticulo.cs

using LogiaNegocio.Dominio;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs.MappersDTOs

{

public class MapperArticulo

{

public static IEnumerable<ArticuloDTO> ToListDtoSimple(IEnumerable<Articulo> articulos)

{

return articulos.Select(a => new ArticuloDTO()

{

Id = a.Id,

Nombre = a.Nombre.Valor,

Descripcion = a.Descripcion.Valor,

CodigoArticulo = a.CodigoArticulo.Valor,

Precio = a.Precio.Valor,

Stock = a.Stock.Valor,

//TopeDeArticuloPorMovimiento = a.Parametro.TopeMovimiento.Valor PARA AREGLAR DESPUES DE CREAR TOPES

})

.ToList();

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\MappersDTOs\MapperMovimientoStock.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.ValueObjects.MovimientosStock;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs.MappersDTOs

{

public class MapperMovimientoStock

{

public static MovimientoStock ToMovimiento(MovimientoStockDTO ms)

{

DateTime fechaMov = DateTime.Now;

MovimientoStock movimientoStock = new MovimientoStock()

{

Fecha = fechaMov,

Cantidad = new Cantidad(ms.Cantidad),

UsuarioEmail = ms.UsuarioEmail,

Articulo = new Articulo()

{

Id = ms.ArticuloId

},

TipoMovimiento = new TipoMovimiento()

{

Id = ms.TipoMovimientoId

}

};

return movimientoStock;

}

public static MovimientoStockDTO ToMovimientoStockDTO(MovimientoStock ms)

{

MovimientoStockDTO movimientoStockDTO = new MovimientoStockDTO()

{

Id = ms.Id,

FechaMovimiento = ms.Fecha,

Cantidad = ms.Cantidad.Valor,

ArticuloId = ms.Articulo.Id,

TipoMovimientoId = ms.TipoMovimientoId,

UsuarioEmail = ms.UsuarioEmail

};

return movimientoStockDTO;

}

public static IEnumerable<MovimientoStockDTO> ToListMovimientoDTO(IEnumerable<MovimientoStock> movimientos)

{

return movimientos.Select(mov => new MovimientoStockDTO

{

Id = mov.Id,

FechaMovimiento = mov.Fecha,

Cantidad = mov.Cantidad.Valor,

ArticuloNombre = mov.Articulo.Nombre.Valor,

TipoMovimientoNombre = mov.TipoMovimiento.NombreTipoMovimiento.Valor,

UsuarioEmail = mov.UsuarioEmail,

ArticuloId = mov.Articulo.Id,

TipoMovimientoId = mov.TipoMovimientoId,

});

}

public static IEnumerable<MovimientoStock> ToListMovimientoStock(IEnumerable<MovimientoStockDTO> lista)

{

return lista.Select(mov => new MovimientoStock

{

Fecha = mov.FechaMovimiento,

Cantidad = new Cantidad(mov.Cantidad),

UsuarioEmail = mov.UsuarioEmail,

Articulo = new Articulo()

{

Id = mov.ArticuloId

},

TipoMovimiento = new TipoMovimiento()

{

Id = mov.TipoMovimientoId

}

});

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\MappersDTOs\MapperTipoMovimiento.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.ValueObjects.ArticuloVOs;

using LogiaNegocio.ValueObjects.TipoMovimiento;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using static System.Runtime.InteropServices.JavaScript.JSType;

namespace DTOs.MappersDTOs

{

public static class MapperTipoMovimiento

{

public static IEnumerable<TipoMovimientoDTO> ToListTipoMovimientoDTO(IEnumerable<TipoMovimiento> tipoMovimientos)

{

return tipoMovimientos.Select(tm => new TipoMovimientoDTO

{

Id = tm.Id,

Nombre = tm.NombreTipoMovimiento.Valor,

EsIncremento = tm.EsIncremento

});

}

public static TipoMovimientoDTO ToTipoMovimientoDTO(TipoMovimiento tp)

{

if (tp != null)

{

TipoMovimientoDTO tipoMovimientoDTO = new TipoMovimientoDTO()

{

Id = tp.Id,

Nombre = tp.NombreTipoMovimiento.Valor,

EsIncremento = tp.EsIncremento

};

return tipoMovimientoDTO;

}

return null;

}

public static TipoMovimiento ToTipoMovimiento(TipoMovimientoDTO tpDTO)

{

TipoMovimiento tipoMovimiento = new TipoMovimiento()

{

NombreTipoMovimiento = new NombreTipoMovimiento(tpDTO.Nombre),

EsIncremento = tpDTO.EsIncremento,

Id = tpDTO.Id

};

return tipoMovimiento;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\MappersDTOs\MapperUsuario.cs

using LogiaNegocio.Dominio;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTOs.MappersDTOs

{

public class MapperUsuario

{

public static UsuarioDTO ToUsuarioDTO(Usuario usuario)

{

if (usuario != null)

{

UsuarioDTO usuarioDTO = new UsuarioDTO()

{

Nombre = usuario.Nombre.Valor,

Apellido = usuario.Apellido.Valor,

Email = usuario.Email.Valor,

Contrasena = usuario.Contrasenia.Valor,

ContrasenaEncriptada = usuario.ContraseniaEncriptada,

Rol = usuario.GetType().Name

};

return usuarioDTO;

}

return null;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\obj\Debug\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\obj\Debug\net8.0\DTOs.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("DTOs")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("DTOs")]

[assembly: System.Reflection.AssemblyTitleAttribute("DTOs")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\obj\Debug\net8.0\DTOs.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\obj\Release\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\obj\Release\net8.0\DTOs.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("DTOs")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Release")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("DTOs")]

[assembly: System.Reflection.AssemblyTitleAttribute("DTOs")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\DTOs\obj\Release\net8.0\DTOs.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\Dominio\Administrador.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.Dominio

{

public class Administrador: Usuario

{

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\Dominio\Articulo.cs

using LogiaNegocio.InterfacesDominio;

using LogiaNegocio.ValueObjects.ArticuloVOs;

using LogicaNegocio.Dominio;

using Microsoft.EntityFrameworkCore;

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.Dominio

{

public class Articulo

{

[Key]

public int Id { get; set; }

public NombreArticulo Nombre { get; set; }

public DescripcionArticulo Descripcion { get; set; }

public CodigoArticulo CodigoArticulo { get; set; }

public Precio Precio { get; set; }

public Stock Stock { get; set; }

public Parametro Parametro { get; set; }

public int ParametroId { get; set; }

public void ActualizarStock(int cant, bool esIncremento)

{

if (esIncremento)

{

cant = cant + Stock.Valor;

}

else

{

cant = Stock.Valor - cant;

}

Stock = new Stock(cant);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\Dominio\EncargadoDeposito.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.Dominio

{

public class EncargadoDeposito: Usuario

{

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\Dominio\MovimientoStock.cs

using LogiaNegocio.ExcepcionesPropias;

using LogiaNegocio.InterfacesDominio;

using LogiaNegocio.ValueObjects.ArticuloVOs;

using LogiaNegocio.ValueObjects.MovimientosStock;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.Dominio

{

public class MovimientoStock

{

public int Id { get; set; }

public int TipoMovimientoId { get; set; }

public Cantidad Cantidad { get; set; }

public string UsuarioEmail { get; set; }

public Articulo Articulo { get; set; }

public DateTime Fecha { get; set; }

public TipoMovimiento TipoMovimiento { get; set; }

public void CalcularMovimientoStock(int cant)

{

if (cant <= 0)

throw new DatosInvalidosException("Solo se pueden mover cantidad de articulos con numeros enteros positivos.");

if (cant > Articulo.Parametro.Valor.Valor)

throw new DatosInvalidosException($"No se pueden mover más {Articulo.Nombre.Valor} de los permitido por el límite de movimiento. El tope es {Articulo.Parametro.Valor.Valor}");

if (!TipoMovimiento.EsIncremento)

{

if (cant > Articulo.Stock.Valor)

throw new DatosInvalidosException($"No hay articulos suficientes. Solo contamos {Articulo.Stock.Valor}");

if (cant > Articulo.Stock.Valor && Articulo.Stock.Valor - cant < 0)

throw new DatosInvalidosException($"No hay articulos suficientes. Solo contamos {Articulo.Stock.Valor}");

}

Articulo.ActualizarStock(cant, TipoMovimiento.EsIncremento);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\Dominio\Parametro.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesDominio;

using LogicaNegocio.ValueObjects.ParametroArticulo;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaNegocio.Dominio

{

public class Parametro

{

public int Id { get; set; }

public string Nombre { get; set; }

public ParametroValor Valor { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\Dominio\TipoMovimiento.cs

using LogiaNegocio.InterfacesDominio;

using LogiaNegocio.ValueObjects.TipoMovimiento;

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.Dominio

{

public class TipoMovimiento

{

[Key]

public int Id { get; set;}

public NombreTipoMovimiento NombreTipoMovimiento { get; set; }

public bool EsIncremento { get; set; }

public IEnumerable<MovimientoStock> MovimientosStock { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\Dominio\Usuario.cs

using LogiaNegocio.InterfacesDominio;

using LogiaNegocio.ValueObjects.UsuarioVOs;

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.Dominio

{

public class Usuario

{

public int Id { get; set; }

public EmailUsuario Email { get; set; }

public NombreUsuario Nombre { get; set; }

public NombreUsuario Apellido { get; set; }

public Contrasenia Contrasenia { get; set; }

public string? ContraseniaEncriptada { get; set; }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ExcepcionesPropias\DatosInvalidosException.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ExcepcionesPropias

{

public class DatosInvalidosException : Exception

{

public DatosInvalidosException() { }

public DatosInvalidosException(string message) : base(message) { }

public DatosInvalidosException(string message, Exception innerException) : base(message, innerException) { }

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\InterfacesDominio\IValidable.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.InterfacesDominio

{

public interface IValidable

{

void Validar();

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\InterfacesRepositorios\IRepositorio.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.InterfacesRepositorios

{

public interface IRepositorio<T>

{

T FindById(int id);

IEnumerable<T> FindAll();

void Add(T obj);

void Remove(int id);

void Update(T obj);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\InterfacesRepositorios\IRepositorioArticulos.cs

using LogiaNegocio.Dominio;

namespace LogiaNegocio.InterfacesRepositorios

{

public interface IRepositorioArticulos

{

IEnumerable<Articulo> FindAll();

Articulo FindById(int id);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\InterfacesRepositorios\IRepositorioMovimientosStock.cs

using LogiaNegocio.Dominio;

using System;

using System.Collections.Generic;

using System.Diagnostics.Contracts;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.InterfacesRepositorios

{

public interface IRepositorioMovimientosStock

{

MovimientoStock FindById(int id);

IEnumerable<MovimientoStock> FindAll();

void Add(MovimientoStock obj);

IEnumerable<MovimientoStock> ObtenerMovimientosPorArticuloYTipo(int articuloId, int tipoMovimiento);

IEnumerable<MovimientoStock> ObtenerMovimientosPorArticuloYTipo(int articuloId, int tipoMovimiento, int pagina);

IEnumerable<MovimientoStock> ObtenerMovimientosPorRangoDeFechas(DateTime inicio, DateTime fin);

IEnumerable<MovimientoStock> ObtenerMovimientosPorRangoDeFechas(DateTime inicio, DateTime fin, int pagina);

public string ObtenerResumenMovimientos();

public double ObtenerCantidadPaginas(int totalMovimientos);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\InterfacesRepositorios\IRepositorioTipoMovimiento.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaNegocio.InterfacesRepositorios

{

public interface IRepositorioTipoMovimiento : IRepositorio<TipoMovimiento>

{

TipoMovimiento BuscarPorNombre(string nombre);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\InterfacesRepositorios\IRepositorioUsuarios.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaNegocio.InterfacesRepositorios

{

public interface IRepositorioUsuarios

{

Usuario FindByEmail(string email);

Usuario Login(string email, string password);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\obj\Debug\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\obj\Debug\net8.0\LogiaNegocio.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("LogiaNegocio")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("LogiaNegocio")]

[assembly: System.Reflection.AssemblyTitleAttribute("LogiaNegocio")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\obj\Debug\net8.0\LogiaNegocio.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\obj\Debug\net8.0\LogicaNegocio.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("LogicaNegocio")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("LogicaNegocio")]

[assembly: System.Reflection.AssemblyTitleAttribute("LogicaNegocio")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\obj\Debug\net8.0\LogicaNegocio.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\obj\Release\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\obj\Release\net8.0\LogicaNegocio.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("LogicaNegocio")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Release")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("LogicaNegocio")]

[assembly: System.Reflection.AssemblyTitleAttribute("LogicaNegocio")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\obj\Release\net8.0\LogicaNegocio.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\ArticuloVOs\CodigoArticulo.cs

using LogiaNegocio.ExcepcionesPropias;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.ArticuloVOs

{

public class CodigoArticulo

{

public long Valor { get; init; }

private CodigoArticulo() { }

public CodigoArticulo(long valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (Valor <= 0) throw new DatosInvalidosException("EL codigo no puede ser menor a 0.");

if (Valor == 0) throw new DatosInvalidosException("El codigo de articulo no puede estar vacio.");

string cantValor = Valor.ToString();

if (cantValor.Length != 13) throw new DatosInvalidosException("El codigo de articulo debe tener 13 digitos.");

}

public override bool Equals(object? obj)

{

CodigoArticulo otro = obj as CodigoArticulo;

if (otro == null) return false;

return otro.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\ArticuloVOs\DescripcionArticulo.cs

using LogiaNegocio.ExcepcionesPropias;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.ArticuloVOs

{

public class DescripcionArticulo

{

public string Valor { get; init; }

private DescripcionArticulo() { }

public DescripcionArticulo(string valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (string.IsNullOrEmpty(Valor)) throw new DatosInvalidosException("La descripcion no puede estar vacia.");

if (Valor.Length < 3) throw new DatosInvalidosException("La descripcion no puede tener menos de 3 caracteres.");

}

public override bool Equals(object? obj)

{

DescripcionArticulo otro = obj as DescripcionArticulo;

if (otro == null) return false;

return otro.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\ArticuloVOs\NombreArticulo.cs

using LogiaNegocio.ExcepcionesPropias;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.ArticuloVOs

{

public class NombreArticulo

{

public string Valor { get; init; }

private NombreArticulo() { }

public NombreArticulo(string valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (string.IsNullOrEmpty(Valor)) throw new DatosInvalidosException("El nombre de articulo no puede estar vacio.");

if (Valor.Length < 3) throw new DatosInvalidosException("El nombre no puede tener menos de 3 caracteres.");

}

public override bool Equals(object? obj)

{

NombreArticulo otro = obj as NombreArticulo;

if (otro == null) return false;

return otro.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\ArticuloVOs\Precio.cs

using LogiaNegocio.ExcepcionesPropias;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.ArticuloVOs

{

public class Precio

{

public decimal Valor { get; init; }

private Precio() { }

public Precio(decimal valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (Valor <= 0) throw new DatosInvalidosException("El precio no puede ser inferior o igual a 0.");

}

public override bool Equals(object? obj)

{

Precio otro = obj as Precio;

if (otro == null) return false;

return otro.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\ArticuloVOs\Stock.cs

using LogiaNegocio.ExcepcionesPropias;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.ArticuloVOs

{

public class Stock

{

public int Valor { get; init; }

private Stock() { }

public Stock(int valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (Valor < 0) throw new DatosInvalidosException("El stock no puede ser inferior a 0.");

}

public override bool Equals(object? obj)

{

Stock otro = obj as Stock;

if (otro == null) return false;

return otro.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\MovimientosStock\Cantidad.cs

using LogiaNegocio.ExcepcionesPropias;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.MovimientosStock

{

public class Cantidad

{

public int Valor { get; init; }

private Cantidad() { }

public Cantidad(int valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if(Valor <=0)throw new DatosInvalidosException("Únicamente se permiten movimientos con números naturales positivos.");

}

public override bool Equals(object? obj)

{

Cantidad otra = obj as Cantidad;

if(otra == null) return false;

return otra.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\ParametroArticulo\ParametroValor.cs

using LogiaNegocio.ExcepcionesPropias;

using LogiaNegocio.ValueObjects.MovimientosStock;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaNegocio.ValueObjects.ParametroArticulo

{

public class ParametroValor

{

public int Valor { get; init; }

private ParametroValor() { }

public ParametroValor(int valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (Valor <= 0) throw new DatosInvalidosException("Únicamente se permiten topes de movimientos con números valores iguales o inferiores a 0.");

}

public override bool Equals(object? obj)

{

ParametroValor otra = obj as ParametroValor;

if (otra == null) return false;

return otra.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\TipoMovimiento\NombreTipoMovimiento.cs

using LogiaNegocio.ExcepcionesPropias;

using LogiaNegocio.ValueObjects.UsuarioVOs;

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.TipoMovimiento

{

public class NombreTipoMovimiento

{

[MaxLength(100)]

[MinLength(3, ErrorMessage = "El nombre debe contener al menos 3 caracteres")]

[RegularExpression(@"^[a-zA-Z]+([' -][a-zA-Z]+)\*$", ErrorMessage = "La cadena debe contener solo caracteres alfabéticos, " +

"espacios, apóstrofes o guiones del medio, sin caracteres no alfabéticos al principio o al final.")]

public string Valor { get; init; }

private NombreTipoMovimiento() { }

public NombreTipoMovimiento(string valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (string.IsNullOrEmpty(Valor)) throw new DatosInvalidosException("El nombre de usuario no puede estar vacio.");

}

public override bool Equals(object? obj)

{

NombreTipoMovimiento otro = obj as NombreTipoMovimiento;

if (otro == null) return false;

return otro.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\UsuarioVOs\Contrasenia.cs

using LogiaNegocio.ExcepcionesPropias;

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.UsuarioVOs

{

public class Contrasenia

{

[MinLength(6, ErrorMessage = "La contraseña debe conter al menos 6 caracteres")]

[MaxLength(100)]

[RegularExpression(@"^(?=.\*[A-Z])(?=.\*[a-z])(?=.\*\d)(?=.\*[.,;!])[A-Za-z\d.,;!]+$", ErrorMessage = "La contraseña debe contener al menos una letra mayúscula, una minúscula," +

" un dígito y un carácter de puntuación: punto, punto y coma, coma o signo de admiración de cierre.")]

public string Valor { get; init; }

private Contrasenia() { }

public Contrasenia(string valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (string.IsNullOrEmpty(Valor)) throw new DatosInvalidosException("La contrasenia no puede estar vacia.");

}

public override bool Equals(object? obj)

{

Contrasenia otra = obj as Contrasenia;

if (otra == null) return false;

return otra.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\UsuarioVOs\EmailUsuario.cs

using LogiaNegocio.ExcepcionesPropias;

using LogiaNegocio.ValueObjects.ArticuloVOs;

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.UsuarioVOs

{

public class EmailUsuario

{

[MaxLength(100)]

[EmailAddress]

public string Valor { get; init; }

private EmailUsuario(){}

public EmailUsuario(string valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (string.IsNullOrEmpty(Valor)) throw new DatosInvalidosException("El email no puede estar vacio.");

if (Valor.Length < 3) throw new DatosInvalidosException("El email no puede tener menos de 3 caracteres.");

}

public override bool Equals(object? obj)

{

EmailUsuario otro = obj as EmailUsuario;

if(otro == null) return false;

return otro.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogiaNegocio\ValueObjects\UsuarioVOs\NombreUsuario.cs

using LogiaNegocio.ExcepcionesPropias;

using LogiaNegocio.InterfacesDominio;

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogiaNegocio.ValueObjects.UsuarioVOs

{

public class NombreUsuario

{

[MaxLength(100)]

[MinLength(3, ErrorMessage = "El nombre debe contener al menos 3 caracteres")]

[RegularExpression(@"^[a-zA-Z]+([' -][a-zA-Z]+)\*$", ErrorMessage = "La cadena debe contener solo caracteres alfabéticos, " +

"espacios, apóstrofes o guiones del medio, sin caracteres no alfabéticos al principio o al final.")]

public string Valor { get; init; }

private NombreUsuario() { }

public NombreUsuario(string valor)

{

Valor = valor;

Validar();

}

private void Validar()

{

if (string.IsNullOrEmpty(Valor)) throw new DatosInvalidosException("El nombre de usuario no puede estar vacio.");

}

public override bool Equals(object? obj)

{

NombreUsuario otro = obj as NombreUsuario;

if (otro == null) return false;

return otro.Valor == Valor;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUAltaMovimientoStockDTO.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using LogicaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUAltaMovimientoStockDTO : ICUAlta<MovimientoStockDTO>

{

public IRepositorioMovimientosStock Repo { get; set; }

public IRepositorioArticulos RepoArt { get; set; }

public IRepositorioTipoMovimiento RepoTipoMov { get; set; }

public CUAltaMovimientoStockDTO(IRepositorioMovimientosStock repo, IRepositorioArticulos repoArt,

IRepositorioTipoMovimiento repoTipoMov)

{

Repo = repo;

RepoArt = repoArt;

RepoTipoMov = repoTipoMov;

}

public void Alta(MovimientoStockDTO obj)

{

MovimientoStock ms = MapperMovimientoStock.ToMovimiento(obj);

ms.Articulo = RepoArt.FindById(obj.ArticuloId);

ms.TipoMovimiento = RepoTipoMov.FindById(obj.TipoMovimientoId);

ms.CalcularMovimientoStock(obj.Cantidad);

Repo.Add(ms);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUAltaTipoMovimientoDTO.cs

using DTOs;

using DTOs.MappersDTOs;

using LogicaAplicacion.InterfacesCU;

using LogicaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUAltaTipoMovimientoDTO : ICUAlta<TipoMovimientoDTO>

{

public IRepositorioTipoMovimiento Repo { get; set; }

public CUAltaTipoMovimientoDTO(IRepositorioTipoMovimiento repo)

{

Repo = repo;

}

public void Alta(TipoMovimientoDTO obj)

{

Repo.Add(MapperTipoMovimiento.ToTipoMovimiento(obj));

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUBajaTipoMovimientoDTO.cs

using LogicaAplicacion.InterfacesCU;

using LogicaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUBajaTipoMovimientoDTO : ICUBaja

{

public IRepositorioTipoMovimiento Repo { get; set; }

public CUBajaTipoMovimientoDTO(IRepositorioTipoMovimiento repo)

{

Repo = repo;

}

public void Baja(int id)

{

Repo.Remove(id);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUBuscarMovimientoStockPorId.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUBuscarMovimientoStockPorId : ICUBuscarPorId<MovimientoStockDTO>

{

public IRepositorioMovimientosStock Repo { get; set; }

public CUBuscarMovimientoStockPorId(IRepositorioMovimientosStock repo)

{

Repo = repo;

}

public MovimientoStockDTO GetById(int id)

{

return MapperMovimientoStock.ToMovimientoStockDTO(Repo.FindById(id));

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUBuscarTipoMovimientoPorId.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using LogicaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUBuscarTipoMovimientoPorId : ICUBuscarPorId<TipoMovimientoDTO>

{

public IRepositorioTipoMovimiento Repo { get; set; }

public CUBuscarTipoMovimientoPorId(IRepositorioTipoMovimiento repo)

{

Repo = repo;

}

public TipoMovimientoDTO GetById(int id)

{

return MapperTipoMovimiento.ToTipoMovimientoDTO(Repo.FindById(id));

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUListarArticulosDTO.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUListarArticulosDTO : ICUListar<ArticuloDTO>

{

public IRepositorioArticulos Repo { get; set; }

public CUListarArticulosDTO(IRepositorioArticulos repo)

{

Repo = repo;

}

public IEnumerable<ArticuloDTO> Listar()

{

IEnumerable<Articulo> articulos = Repo.FindAll();

return MapperArticulo.ToListDtoSimple(articulos);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUListarMovimientosDTO.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using LogicaDatos.Repositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUListarMovimientosDTO : ICUListar<MovimientoStockDTO>

{

public IRepositorioMovimientosStock Repo { get; set; }

public CUListarMovimientosDTO(IRepositorioMovimientosStock repo)

{

Repo = repo;

}

public IEnumerable<MovimientoStockDTO> Listar()

{

IEnumerable<MovimientoStock> movimientosStock = Repo.FindAll();

IEnumerable<MovimientoStockDTO> movimientosStockDTO = MapperMovimientoStock.ToListMovimientoDTO(movimientosStock);

return movimientosStockDTO;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUListarTiposMovimientosDTO.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogicaAplicacion.InterfacesCU;

using LogicaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUListarTiposMovimientosDTO : ICUListar<TipoMovimientoDTO>

{

public IRepositorioTipoMovimiento Repo { get; set; }

public CUListarTiposMovimientosDTO(IRepositorioTipoMovimiento repo)

{

Repo = repo;

}

public IEnumerable<TipoMovimientoDTO> Listar()

{

IEnumerable<TipoMovimiento> tipoMovimientos = Repo.FindAll();

return MapperTipoMovimiento.ToListTipoMovimientoDTO(tipoMovimientos);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CULoginUsuario.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogicaAplicacion.InterfacesCU;

using LogicaDatos.Repositorios;

using LogicaNegocio.InterfacesRepositorios;

namespace LogicaAplicacion.CasosUso

{

public class CULoginUsuario : ICULoginUsuario

{

private readonly IRepositorioUsuarios Repositorio;

public CULoginUsuario(IRepositorioUsuarios repositorio)

{

Repositorio = repositorio;

}

public UsuarioDTO Login(string email, string contrasena)

{

Usuario usu = Repositorio.Login(email, contrasena);

UsuarioDTO usuDTO = MapperUsuario.ToUsuarioDTO(usu);

return usuDTO;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUModificarTipoMovimientoDTO.cs

using DTOs;

using DTOs.MappersDTOs;

using LogicaAplicacion.InterfacesCU;

using LogicaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUModificarTipoMovimientoDTO : ICUModificar<TipoMovimientoDTO>

{

public IRepositorioTipoMovimiento Repo { get; set; }

public CUModificarTipoMovimientoDTO(IRepositorioTipoMovimiento repo)

{

Repo= repo;

}

public void Update(TipoMovimientoDTO obj)

{

Repo.Update(MapperTipoMovimiento.ToTipoMovimiento(obj));

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUObtenerCantidadPaginas.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using LogicaDatos.Repositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUObtenerCantidadPaginas : ICUObtenerCantidadPaginas

{

public IRepositorioMovimientosStock Repo { get; set; }

public CUObtenerCantidadPaginas(IRepositorioMovimientosStock repo)

{

Repo = repo;

}

public double ObtenerCantidadPaginas(int totalMovimientos)

{

return Repo.ObtenerCantidadPaginas(totalMovimientos);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUObtenerMovimientosPorArticuloYTipo.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using LogicaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUObtenerMovimientosPorArticuloYTipo : ICUObtenerMovimientosPorArticuloYTipo

{

public IRepositorioMovimientosStock Repo { get; set; }

public CUObtenerMovimientosPorArticuloYTipo(IRepositorioMovimientosStock repo)

{

Repo = repo;

}

public IEnumerable<MovimientoStockDTO> ObtenerMovimientosPorArticuloYTipo(int ArticuloId, int TipoMovimientoId, int pagina)

{

IEnumerable<MovimientoStock> encontrados = Repo.ObtenerMovimientosPorArticuloYTipo(ArticuloId, TipoMovimientoId, pagina);

return MapperMovimientoStock.ToListMovimientoDTO(encontrados);

}

public IEnumerable<MovimientoStockDTO> ObtenerMovimientosPorArticuloYTipo(int ArticuloId, int TipoMovimientoId)

{

IEnumerable<MovimientoStock> encontrados = Repo.ObtenerMovimientosPorArticuloYTipo(ArticuloId, TipoMovimientoId);

return MapperMovimientoStock.ToListMovimientoDTO(encontrados);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUObtenerMovimientosPorRangoDeFechas.cs

using DTOs;

using DTOs.MappersDTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUObtenerMovimientosPorRangoDeFechas : ICUObtenerMovimientosPorRangoDeFechas

{

public IRepositorioMovimientosStock Repo { get; set; }

public CUObtenerMovimientosPorRangoDeFechas(IRepositorioMovimientosStock repo)

{

Repo = repo;

}

public IEnumerable<MovimientoStockDTO> ObtenerMovimientosPorRangoFechas(DateTime inicio, DateTime fin, int pagina)

{

IEnumerable<MovimientoStock> encontrados = Repo.ObtenerMovimientosPorRangoDeFechas(inicio, fin, pagina);

return MapperMovimientoStock.ToListMovimientoDTO(encontrados);

}

public IEnumerable<MovimientoStockDTO> ObtenerMovimientosPorRangoFechas(DateTime inicio, DateTime fin)

{

IEnumerable<MovimientoStock> encontrados = Repo.ObtenerMovimientosPorRangoDeFechas(inicio, fin);

return MapperMovimientoStock.ToListMovimientoDTO(encontrados);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\CasosUso\CUObtenerResumenMovimientos.cs

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.InterfacesCU;

using LogicaDatos.Repositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.CasosUso

{

public class CUObtenerResumenMovimientos : ICUObtenerResumenMovimientos

{

public IRepositorioMovimientosStock Repo { get; set; }

public CUObtenerResumenMovimientos(IRepositorioMovimientosStock repo)

{

Repo = repo;

}

public string ObtenerResumenMovimientos()

{

return Repo.ObtenerResumenMovimientos();

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUAlta.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUAlta <T>

{

void Alta(T obj);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUBaja.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUBaja

{

void Baja(int id);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUBuscarPorId.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUBuscarPorId<T>

{

T GetById(int id);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUListar.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUListar<T>

{

IEnumerable<T> Listar();

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICULoginUsuario.cs

using DTOs;

using LogiaNegocio.Dominio;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICULoginUsuario

{

UsuarioDTO Login(string email, string contrasena);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUModificar.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUModificar<T>

{

void Update(T obj);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUObtenerCantidadPaginas.cs

using DTOs;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUObtenerCantidadPaginas

{

double ObtenerCantidadPaginas(int totalMovimientos);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUObtenerMovimientosPorArticuloYTipo.cs

using DTOs;

using LogiaNegocio.Dominio;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUObtenerMovimientosPorArticuloYTipo

{

IEnumerable<MovimientoStockDTO> ObtenerMovimientosPorArticuloYTipo(int ArticuloId, int TipoMovimientoId);

IEnumerable<MovimientoStockDTO> ObtenerMovimientosPorArticuloYTipo(int ArticuloId, int TipoMovimientoId, int pagina);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUObtenerMovimientosPorRangoDeFechas.cs

using DTOs;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUObtenerMovimientosPorRangoDeFechas

{

IEnumerable<MovimientoStockDTO> ObtenerMovimientosPorRangoFechas(DateTime inicio, DateTime fin, int pagina);

IEnumerable<MovimientoStockDTO> ObtenerMovimientosPorRangoFechas(DateTime inicio, DateTime fin);

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\InterfacesCU\ICUObtenerResumenMovimientos.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaAplicacion.InterfacesCU

{

public interface ICUObtenerResumenMovimientos

{

string ObtenerResumenMovimientos();

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\obj\Debug\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\obj\Debug\net8.0\LogicaAplicacion.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("LogicaAplicacion")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("LogicaAplicacion")]

[assembly: System.Reflection.AssemblyTitleAttribute("LogicaAplicacion")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\obj\Debug\net8.0\LogicaAplicacion.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\obj\Release\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\obj\Release\net8.0\LogicaAplicacion.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("LogicaAplicacion")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Release")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("LogicaAplicacion")]

[assembly: System.Reflection.AssemblyTitleAttribute("LogicaAplicacion")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaAplicacion\obj\Release\net8.0\LogicaAplicacion.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Migrations\20240613134153\_ini.cs

using System;

using Microsoft.EntityFrameworkCore.Migrations;

#nullable disable

namespace LogicaDatos.Migrations

{

/// <inheritdoc />

public partial class ini : Migration

{

/// <inheritdoc />

protected override void Up(MigrationBuilder migrationBuilder)

{

migrationBuilder.CreateTable(

name: "Articulos",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

Nombre\_Valor = table.Column<string>(type: "nvarchar(max)", nullable: false),

Descripcion\_Valor = table.Column<string>(type: "nvarchar(max)", nullable: false),

CodigoArticulo\_Valor = table.Column<long>(type: "bigint", nullable: false),

Precio\_Valor = table.Column<decimal>(type: "decimal(18,2)", nullable: false),

Stock\_Valor = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_Articulos", x => x.Id);

});

migrationBuilder.CreateTable(

name: "TiposMovimiento",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

NombreTipoMovimiento\_Valor = table.Column<string>(type: "nvarchar(100)", maxLength: 100, nullable: false),

EsIncremento = table.Column<bool>(type: "bit", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_TiposMovimiento", x => x.Id);

});

migrationBuilder.CreateTable(

name: "Usuarios",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

Email\_Valor = table.Column<string>(type: "nvarchar(100)", maxLength: 100, nullable: false),

Nombre\_Valor = table.Column<string>(type: "nvarchar(100)", maxLength: 100, nullable: false),

Apellido\_Valor = table.Column<string>(type: "nvarchar(100)", maxLength: 100, nullable: false),

Contrasenia\_Valor = table.Column<string>(type: "nvarchar(100)", maxLength: 100, nullable: false),

ContraseniaEncriptada = table.Column<string>(type: "nvarchar(max)", nullable: true),

TipoUsuario = table.Column<string>(type: "nvarchar(21)", maxLength: 21, nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_Usuarios", x => x.Id);

});

migrationBuilder.CreateTable(

name: "ParametroArticulos",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

ArticuloId = table.Column<int>(type: "int", nullable: false),

TopeMovimiento\_Valor = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_ParametroArticulos", x => x.Id);

table.ForeignKey(

name: "FK\_ParametroArticulos\_Articulos\_ArticuloId",

column: x => x.ArticuloId,

principalTable: "Articulos",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

});

migrationBuilder.CreateTable(

name: "MovimientosStock",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

TipoMovimientoId = table.Column<int>(type: "int", nullable: false),

Cantidad\_Valor = table.Column<int>(type: "int", nullable: false),

UsuarioEmail = table.Column<string>(type: "nvarchar(max)", nullable: false),

ArticuloId = table.Column<int>(type: "int", nullable: false),

Fecha = table.Column<DateTime>(type: "datetime2", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_MovimientosStock", x => x.Id);

table.ForeignKey(

name: "FK\_MovimientosStock\_Articulos\_ArticuloId",

column: x => x.ArticuloId,

principalTable: "Articulos",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

table.ForeignKey(

name: "FK\_MovimientosStock\_TiposMovimiento\_TipoMovimientoId",

column: x => x.TipoMovimientoId,

principalTable: "TiposMovimiento",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

});

migrationBuilder.CreateIndex(

name: "IX\_MovimientosStock\_ArticuloId",

table: "MovimientosStock",

column: "ArticuloId");

migrationBuilder.CreateIndex(

name: "IX\_MovimientosStock\_TipoMovimientoId",

table: "MovimientosStock",

column: "TipoMovimientoId");

migrationBuilder.CreateIndex(

name: "IX\_ParametroArticulos\_ArticuloId",

table: "ParametroArticulos",

column: "ArticuloId",

unique: true);

}

/// <inheritdoc />

protected override void Down(MigrationBuilder migrationBuilder)

{

migrationBuilder.DropTable(

name: "MovimientosStock");

migrationBuilder.DropTable(

name: "ParametroArticulos");

migrationBuilder.DropTable(

name: "Usuarios");

migrationBuilder.DropTable(

name: "TiposMovimiento");

migrationBuilder.DropTable(

name: "Articulos");

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Migrations\20240613134153\_ini.Designer.cs

// <auto-generated />

using System;

using LogicaDatos.Repositorios;

using Microsoft.EntityFrameworkCore;

using Microsoft.EntityFrameworkCore.Infrastructure;

using Microsoft.EntityFrameworkCore.Metadata;

using Microsoft.EntityFrameworkCore.Migrations;

using Microsoft.EntityFrameworkCore.Storage.ValueConversion;

#nullable disable

namespace LogicaDatos.Migrations

{

[DbContext(typeof(DepositoContext))]

[Migration("20240613134153\_ini")]

partial class ini

{

/// <inheritdoc />

protected override void BuildTargetModel(ModelBuilder modelBuilder)

{

#pragma warning disable 612, 618

modelBuilder

.HasAnnotation("ProductVersion", "8.0.6")

.HasAnnotation("Relational:MaxIdentifierLength", 128);

SqlServerModelBuilderExtensions.UseIdentityColumns(modelBuilder);

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.HasKey("Id");

b.ToTable("Articulos");

});

modelBuilder.Entity("LogiaNegocio.Dominio.MovimientoStock", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<int>("ArticuloId")

.HasColumnType("int");

b.Property<DateTime>("Fecha")

.HasColumnType("datetime2");

b.Property<int>("TipoMovimientoId")

.HasColumnType("int");

b.Property<string>("UsuarioEmail")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.HasKey("Id");

b.HasIndex("ArticuloId");

b.HasIndex("TipoMovimientoId");

b.ToTable("MovimientosStock");

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<bool>("EsIncremento")

.HasColumnType("bit");

b.HasKey("Id");

b.ToTable("TiposMovimiento");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Usuario", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<string>("ContraseniaEncriptada")

.HasColumnType("nvarchar(max)");

b.Property<string>("TipoUsuario")

.IsRequired()

.HasMaxLength(21)

.HasColumnType("nvarchar(21)");

b.HasKey("Id");

b.ToTable("Usuarios");

b.HasDiscriminator<string>("TipoUsuario").HasValue("Usuario");

b.UseTphMappingStrategy();

});

modelBuilder.Entity("LogicaNegocio.Dominio.ParametroArticulo", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<int>("ArticuloId")

.HasColumnType("int");

b.HasKey("Id");

b.HasIndex("ArticuloId")

.IsUnique();

b.ToTable("ParametroArticulos");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Administrador", b =>

{

b.HasBaseType("LogiaNegocio.Dominio.Usuario");

b.HasDiscriminator().HasValue("Administrador");

});

modelBuilder.Entity("LogiaNegocio.Dominio.EncargadoDeposito", b =>

{

b.HasBaseType("LogiaNegocio.Dominio.Usuario");

b.HasDiscriminator().HasValue("EncargadoDeposito");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.CodigoArticulo", "CodigoArticulo", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<long>("Valor")

.HasColumnType("bigint");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.DescripcionArticulo", "Descripcion", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasColumnType("nvarchar(max)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.NombreArticulo", "Nombre", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasColumnType("nvarchar(max)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.Precio", "Precio", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<decimal>("Valor")

.HasColumnType("decimal(18,2)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.Stock", "Stock", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.Navigation("CodigoArticulo")

.IsRequired();

b.Navigation("Descripcion")

.IsRequired();

b.Navigation("Nombre")

.IsRequired();

b.Navigation("Precio")

.IsRequired();

b.Navigation("Stock")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.MovimientoStock", b =>

{

b.HasOne("LogiaNegocio.Dominio.Articulo", "Articulo")

.WithMany()

.HasForeignKey("ArticuloId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.HasOne("LogiaNegocio.Dominio.TipoMovimiento", "TipoMovimiento")

.WithMany("MovimientosStock")

.HasForeignKey("TipoMovimientoId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.OwnsOne("LogiaNegocio.ValueObjects.MovimientosStock.Cantidad", "Cantidad", b1 =>

{

b1.Property<int>("MovimientoStockId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("MovimientoStockId");

b1.ToTable("MovimientosStock");

b1.WithOwner()

.HasForeignKey("MovimientoStockId");

});

b.Navigation("Articulo");

b.Navigation("Cantidad")

.IsRequired();

b.Navigation("TipoMovimiento");

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.TipoMovimiento.NombreTipoMovimiento", "NombreTipoMovimiento", b1 =>

{

b1.Property<int>("TipoMovimientoId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("TipoMovimientoId");

b1.ToTable("TiposMovimiento");

b1.WithOwner()

.HasForeignKey("TipoMovimientoId");

});

b.Navigation("NombreTipoMovimiento")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.Usuario", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.NombreUsuario", "Apellido", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.NombreUsuario", "Nombre", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.Contrasenia", "Contrasenia", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.EmailUsuario", "Email", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.Navigation("Apellido")

.IsRequired();

b.Navigation("Contrasenia")

.IsRequired();

b.Navigation("Email")

.IsRequired();

b.Navigation("Nombre")

.IsRequired();

});

modelBuilder.Entity("LogicaNegocio.Dominio.ParametroArticulo", b =>

{

b.HasOne("LogiaNegocio.Dominio.Articulo", "Articulo")

.WithOne("Parametro")

.HasForeignKey("LogicaNegocio.Dominio.ParametroArticulo", "ArticuloId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.OwnsOne("LogicaNegocio.ValueObjects.ParametroArticulo.TopeMovimiento", "TopeMovimiento", b1 =>

{

b1.Property<int>("ParametroArticuloId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("ParametroArticuloId");

b1.ToTable("ParametroArticulos");

b1.WithOwner()

.HasForeignKey("ParametroArticuloId");

});

b.Navigation("Articulo");

b.Navigation("TopeMovimiento")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.Navigation("Parametro")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.Navigation("MovimientosStock");

});

#pragma warning restore 612, 618

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Migrations\20240616204531\_parametro.cs

using Microsoft.EntityFrameworkCore.Migrations;

#nullable disable

namespace LogicaDatos.Migrations

{

/// <inheritdoc />

public partial class parametro : Migration

{

/// <inheritdoc />

protected override void Up(MigrationBuilder migrationBuilder)

{

migrationBuilder.DropTable(

name: "ParametroArticulos");

migrationBuilder.AddColumn<int>(

name: "ParametroId",

table: "Articulos",

type: "int",

nullable: false,

defaultValue: 0);

migrationBuilder.CreateTable(

name: "Parametros",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

Nombre = table.Column<string>(type: "nvarchar(max)", nullable: false),

TopeMovimiento\_Valor = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_Parametros", x => x.Id);

});

migrationBuilder.CreateIndex(

name: "IX\_Articulos\_ParametroId",

table: "Articulos",

column: "ParametroId",

unique: true);

migrationBuilder.AddForeignKey(

name: "FK\_Articulos\_Parametros\_ParametroId",

table: "Articulos",

column: "ParametroId",

principalTable: "Parametros",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

}

/// <inheritdoc />

protected override void Down(MigrationBuilder migrationBuilder)

{

migrationBuilder.DropForeignKey(

name: "FK\_Articulos\_Parametros\_ParametroId",

table: "Articulos");

migrationBuilder.DropTable(

name: "Parametros");

migrationBuilder.DropIndex(

name: "IX\_Articulos\_ParametroId",

table: "Articulos");

migrationBuilder.DropColumn(

name: "ParametroId",

table: "Articulos");

migrationBuilder.CreateTable(

name: "ParametroArticulos",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

ArticuloId = table.Column<int>(type: "int", nullable: false),

TopeMovimiento\_Valor = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_ParametroArticulos", x => x.Id);

table.ForeignKey(

name: "FK\_ParametroArticulos\_Articulos\_ArticuloId",

column: x => x.ArticuloId,

principalTable: "Articulos",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

});

migrationBuilder.CreateIndex(

name: "IX\_ParametroArticulos\_ArticuloId",

table: "ParametroArticulos",

column: "ArticuloId",

unique: true);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Migrations\20240616204531\_parametro.Designer.cs

// <auto-generated />

using System;

using LogicaDatos.Repositorios;

using Microsoft.EntityFrameworkCore;

using Microsoft.EntityFrameworkCore.Infrastructure;

using Microsoft.EntityFrameworkCore.Metadata;

using Microsoft.EntityFrameworkCore.Migrations;

using Microsoft.EntityFrameworkCore.Storage.ValueConversion;

#nullable disable

namespace LogicaDatos.Migrations

{

[DbContext(typeof(DepositoContext))]

[Migration("20240616204531\_parametro")]

partial class parametro

{

/// <inheritdoc />

protected override void BuildTargetModel(ModelBuilder modelBuilder)

{

#pragma warning disable 612, 618

modelBuilder

.HasAnnotation("ProductVersion", "8.0.6")

.HasAnnotation("Relational:MaxIdentifierLength", 128);

SqlServerModelBuilderExtensions.UseIdentityColumns(modelBuilder);

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<int>("ParametroId")

.HasColumnType("int");

b.HasKey("Id");

b.HasIndex("ParametroId")

.IsUnique();

b.ToTable("Articulos");

});

modelBuilder.Entity("LogiaNegocio.Dominio.MovimientoStock", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<int>("ArticuloId")

.HasColumnType("int");

b.Property<DateTime>("Fecha")

.HasColumnType("datetime2");

b.Property<int>("TipoMovimientoId")

.HasColumnType("int");

b.Property<string>("UsuarioEmail")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.HasKey("Id");

b.HasIndex("ArticuloId");

b.HasIndex("TipoMovimientoId");

b.ToTable("MovimientosStock");

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<bool>("EsIncremento")

.HasColumnType("bit");

b.HasKey("Id");

b.ToTable("TiposMovimiento");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Usuario", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<string>("ContraseniaEncriptada")

.HasColumnType("nvarchar(max)");

b.Property<string>("TipoUsuario")

.IsRequired()

.HasMaxLength(21)

.HasColumnType("nvarchar(21)");

b.HasKey("Id");

b.ToTable("Usuarios");

b.HasDiscriminator<string>("TipoUsuario").HasValue("Usuario");

b.UseTphMappingStrategy();

});

modelBuilder.Entity("LogicaNegocio.Dominio.Parametro", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<string>("Nombre")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.HasKey("Id");

b.ToTable("Parametros");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Administrador", b =>

{

b.HasBaseType("LogiaNegocio.Dominio.Usuario");

b.HasDiscriminator().HasValue("Administrador");

});

modelBuilder.Entity("LogiaNegocio.Dominio.EncargadoDeposito", b =>

{

b.HasBaseType("LogiaNegocio.Dominio.Usuario");

b.HasDiscriminator().HasValue("EncargadoDeposito");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.HasOne("LogicaNegocio.Dominio.Parametro", "Parametro")

.WithOne()

.HasForeignKey("LogiaNegocio.Dominio.Articulo", "ParametroId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.CodigoArticulo", "CodigoArticulo", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<long>("Valor")

.HasColumnType("bigint");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.DescripcionArticulo", "Descripcion", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasColumnType("nvarchar(max)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.NombreArticulo", "Nombre", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasColumnType("nvarchar(max)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.Precio", "Precio", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<decimal>("Valor")

.HasColumnType("decimal(18,2)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.Stock", "Stock", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.Navigation("CodigoArticulo")

.IsRequired();

b.Navigation("Descripcion")

.IsRequired();

b.Navigation("Nombre")

.IsRequired();

b.Navigation("Parametro");

b.Navigation("Precio")

.IsRequired();

b.Navigation("Stock")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.MovimientoStock", b =>

{

b.HasOne("LogiaNegocio.Dominio.Articulo", "Articulo")

.WithMany()

.HasForeignKey("ArticuloId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.HasOne("LogiaNegocio.Dominio.TipoMovimiento", "TipoMovimiento")

.WithMany("MovimientosStock")

.HasForeignKey("TipoMovimientoId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.OwnsOne("LogiaNegocio.ValueObjects.MovimientosStock.Cantidad", "Cantidad", b1 =>

{

b1.Property<int>("MovimientoStockId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("MovimientoStockId");

b1.ToTable("MovimientosStock");

b1.WithOwner()

.HasForeignKey("MovimientoStockId");

});

b.Navigation("Articulo");

b.Navigation("Cantidad")

.IsRequired();

b.Navigation("TipoMovimiento");

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.TipoMovimiento.NombreTipoMovimiento", "NombreTipoMovimiento", b1 =>

{

b1.Property<int>("TipoMovimientoId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("TipoMovimientoId");

b1.ToTable("TiposMovimiento");

b1.WithOwner()

.HasForeignKey("TipoMovimientoId");

});

b.Navigation("NombreTipoMovimiento")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.Usuario", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.NombreUsuario", "Apellido", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.NombreUsuario", "Nombre", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.Contrasenia", "Contrasenia", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.EmailUsuario", "Email", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.Navigation("Apellido")

.IsRequired();

b.Navigation("Contrasenia")

.IsRequired();

b.Navigation("Email")

.IsRequired();

b.Navigation("Nombre")

.IsRequired();

});

modelBuilder.Entity("LogicaNegocio.Dominio.Parametro", b =>

{

b.OwnsOne("LogicaNegocio.ValueObjects.ParametroArticulo.ParametroValor", "TopeMovimiento", b1 =>

{

b1.Property<int>("ParametroId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("ParametroId");

b1.ToTable("Parametros");

b1.WithOwner()

.HasForeignKey("ParametroId");

});

b.Navigation("TopeMovimiento")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.Navigation("MovimientosStock");

});

#pragma warning restore 612, 618

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Migrations\20240616205050\_parametro-valor.cs

using Microsoft.EntityFrameworkCore.Migrations;

#nullable disable

namespace LogicaDatos.Migrations

{

/// <inheritdoc />

public partial class parametrovalor : Migration

{

/// <inheritdoc />

protected override void Up(MigrationBuilder migrationBuilder)

{

migrationBuilder.RenameColumn(

name: "TopeMovimiento\_Valor",

table: "Parametros",

newName: "Valor\_Valor");

}

/// <inheritdoc />

protected override void Down(MigrationBuilder migrationBuilder)

{

migrationBuilder.RenameColumn(

name: "Valor\_Valor",

table: "Parametros",

newName: "TopeMovimiento\_Valor");

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Migrations\20240616205050\_parametro-valor.Designer.cs

// <auto-generated />

using System;

using LogicaDatos.Repositorios;

using Microsoft.EntityFrameworkCore;

using Microsoft.EntityFrameworkCore.Infrastructure;

using Microsoft.EntityFrameworkCore.Metadata;

using Microsoft.EntityFrameworkCore.Migrations;

using Microsoft.EntityFrameworkCore.Storage.ValueConversion;

#nullable disable

namespace LogicaDatos.Migrations

{

[DbContext(typeof(DepositoContext))]

[Migration("20240616205050\_parametro-valor")]

partial class parametrovalor

{

/// <inheritdoc />

protected override void BuildTargetModel(ModelBuilder modelBuilder)

{

#pragma warning disable 612, 618

modelBuilder

.HasAnnotation("ProductVersion", "8.0.6")

.HasAnnotation("Relational:MaxIdentifierLength", 128);

SqlServerModelBuilderExtensions.UseIdentityColumns(modelBuilder);

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<int>("ParametroId")

.HasColumnType("int");

b.HasKey("Id");

b.HasIndex("ParametroId")

.IsUnique();

b.ToTable("Articulos");

});

modelBuilder.Entity("LogiaNegocio.Dominio.MovimientoStock", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<int>("ArticuloId")

.HasColumnType("int");

b.Property<DateTime>("Fecha")

.HasColumnType("datetime2");

b.Property<int>("TipoMovimientoId")

.HasColumnType("int");

b.Property<string>("UsuarioEmail")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.HasKey("Id");

b.HasIndex("ArticuloId");

b.HasIndex("TipoMovimientoId");

b.ToTable("MovimientosStock");

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<bool>("EsIncremento")

.HasColumnType("bit");

b.HasKey("Id");

b.ToTable("TiposMovimiento");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Usuario", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<string>("ContraseniaEncriptada")

.HasColumnType("nvarchar(max)");

b.Property<string>("TipoUsuario")

.IsRequired()

.HasMaxLength(21)

.HasColumnType("nvarchar(21)");

b.HasKey("Id");

b.ToTable("Usuarios");

b.HasDiscriminator<string>("TipoUsuario").HasValue("Usuario");

b.UseTphMappingStrategy();

});

modelBuilder.Entity("LogicaNegocio.Dominio.Parametro", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<string>("Nombre")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.HasKey("Id");

b.ToTable("Parametros");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Administrador", b =>

{

b.HasBaseType("LogiaNegocio.Dominio.Usuario");

b.HasDiscriminator().HasValue("Administrador");

});

modelBuilder.Entity("LogiaNegocio.Dominio.EncargadoDeposito", b =>

{

b.HasBaseType("LogiaNegocio.Dominio.Usuario");

b.HasDiscriminator().HasValue("EncargadoDeposito");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.HasOne("LogicaNegocio.Dominio.Parametro", "Parametro")

.WithOne()

.HasForeignKey("LogiaNegocio.Dominio.Articulo", "ParametroId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.CodigoArticulo", "CodigoArticulo", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<long>("Valor")

.HasColumnType("bigint");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.DescripcionArticulo", "Descripcion", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasColumnType("nvarchar(max)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.NombreArticulo", "Nombre", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasColumnType("nvarchar(max)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.Precio", "Precio", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<decimal>("Valor")

.HasColumnType("decimal(18,2)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.Stock", "Stock", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.Navigation("CodigoArticulo")

.IsRequired();

b.Navigation("Descripcion")

.IsRequired();

b.Navigation("Nombre")

.IsRequired();

b.Navigation("Parametro");

b.Navigation("Precio")

.IsRequired();

b.Navigation("Stock")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.MovimientoStock", b =>

{

b.HasOne("LogiaNegocio.Dominio.Articulo", "Articulo")

.WithMany()

.HasForeignKey("ArticuloId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.HasOne("LogiaNegocio.Dominio.TipoMovimiento", "TipoMovimiento")

.WithMany("MovimientosStock")

.HasForeignKey("TipoMovimientoId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.OwnsOne("LogiaNegocio.ValueObjects.MovimientosStock.Cantidad", "Cantidad", b1 =>

{

b1.Property<int>("MovimientoStockId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("MovimientoStockId");

b1.ToTable("MovimientosStock");

b1.WithOwner()

.HasForeignKey("MovimientoStockId");

});

b.Navigation("Articulo");

b.Navigation("Cantidad")

.IsRequired();

b.Navigation("TipoMovimiento");

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.TipoMovimiento.NombreTipoMovimiento", "NombreTipoMovimiento", b1 =>

{

b1.Property<int>("TipoMovimientoId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("TipoMovimientoId");

b1.ToTable("TiposMovimiento");

b1.WithOwner()

.HasForeignKey("TipoMovimientoId");

});

b.Navigation("NombreTipoMovimiento")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.Usuario", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.NombreUsuario", "Apellido", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.NombreUsuario", "Nombre", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.Contrasenia", "Contrasenia", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.EmailUsuario", "Email", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.Navigation("Apellido")

.IsRequired();

b.Navigation("Contrasenia")

.IsRequired();

b.Navigation("Email")

.IsRequired();

b.Navigation("Nombre")

.IsRequired();

});

modelBuilder.Entity("LogicaNegocio.Dominio.Parametro", b =>

{

b.OwnsOne("LogicaNegocio.ValueObjects.ParametroArticulo.ParametroValor", "Valor", b1 =>

{

b1.Property<int>("ParametroId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("ParametroId");

b1.ToTable("Parametros");

b1.WithOwner()

.HasForeignKey("ParametroId");

});

b.Navigation("Valor")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.Navigation("MovimientosStock");

});

#pragma warning restore 612, 618

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Migrations\DepositoContextModelSnapshot.cs

// <auto-generated />

using System;

using LogicaDatos.Repositorios;

using Microsoft.EntityFrameworkCore;

using Microsoft.EntityFrameworkCore.Infrastructure;

using Microsoft.EntityFrameworkCore.Metadata;

using Microsoft.EntityFrameworkCore.Storage.ValueConversion;

#nullable disable

namespace LogicaDatos.Migrations

{

[DbContext(typeof(DepositoContext))]

partial class DepositoContextModelSnapshot : ModelSnapshot

{

protected override void BuildModel(ModelBuilder modelBuilder)

{

#pragma warning disable 612, 618

modelBuilder

.HasAnnotation("ProductVersion", "8.0.6")

.HasAnnotation("Relational:MaxIdentifierLength", 128);

SqlServerModelBuilderExtensions.UseIdentityColumns(modelBuilder);

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<int>("ParametroId")

.HasColumnType("int");

b.HasKey("Id");

b.HasIndex("ParametroId")

.IsUnique();

b.ToTable("Articulos");

});

modelBuilder.Entity("LogiaNegocio.Dominio.MovimientoStock", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<int>("ArticuloId")

.HasColumnType("int");

b.Property<DateTime>("Fecha")

.HasColumnType("datetime2");

b.Property<int>("TipoMovimientoId")

.HasColumnType("int");

b.Property<string>("UsuarioEmail")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.HasKey("Id");

b.HasIndex("ArticuloId");

b.HasIndex("TipoMovimientoId");

b.ToTable("MovimientosStock");

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<bool>("EsIncremento")

.HasColumnType("bit");

b.HasKey("Id");

b.ToTable("TiposMovimiento");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Usuario", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<string>("ContraseniaEncriptada")

.HasColumnType("nvarchar(max)");

b.Property<string>("TipoUsuario")

.IsRequired()

.HasMaxLength(21)

.HasColumnType("nvarchar(21)");

b.HasKey("Id");

b.ToTable("Usuarios");

b.HasDiscriminator<string>("TipoUsuario").HasValue("Usuario");

b.UseTphMappingStrategy();

});

modelBuilder.Entity("LogicaNegocio.Dominio.Parametro", b =>

{

b.Property<int>("Id")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

b.Property<string>("Nombre")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.HasKey("Id");

b.ToTable("Parametros");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Administrador", b =>

{

b.HasBaseType("LogiaNegocio.Dominio.Usuario");

b.HasDiscriminator().HasValue("Administrador");

});

modelBuilder.Entity("LogiaNegocio.Dominio.EncargadoDeposito", b =>

{

b.HasBaseType("LogiaNegocio.Dominio.Usuario");

b.HasDiscriminator().HasValue("EncargadoDeposito");

});

modelBuilder.Entity("LogiaNegocio.Dominio.Articulo", b =>

{

b.HasOne("LogicaNegocio.Dominio.Parametro", "Parametro")

.WithOne()

.HasForeignKey("LogiaNegocio.Dominio.Articulo", "ParametroId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.CodigoArticulo", "CodigoArticulo", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<long>("Valor")

.HasColumnType("bigint");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.DescripcionArticulo", "Descripcion", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasColumnType("nvarchar(max)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.NombreArticulo", "Nombre", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasColumnType("nvarchar(max)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.Precio", "Precio", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<decimal>("Valor")

.HasColumnType("decimal(18,2)");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.ArticuloVOs.Stock", "Stock", b1 =>

{

b1.Property<int>("ArticuloId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("ArticuloId");

b1.ToTable("Articulos");

b1.WithOwner()

.HasForeignKey("ArticuloId");

});

b.Navigation("CodigoArticulo")

.IsRequired();

b.Navigation("Descripcion")

.IsRequired();

b.Navigation("Nombre")

.IsRequired();

b.Navigation("Parametro");

b.Navigation("Precio")

.IsRequired();

b.Navigation("Stock")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.MovimientoStock", b =>

{

b.HasOne("LogiaNegocio.Dominio.Articulo", "Articulo")

.WithMany()

.HasForeignKey("ArticuloId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.HasOne("LogiaNegocio.Dominio.TipoMovimiento", "TipoMovimiento")

.WithMany("MovimientosStock")

.HasForeignKey("TipoMovimientoId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.OwnsOne("LogiaNegocio.ValueObjects.MovimientosStock.Cantidad", "Cantidad", b1 =>

{

b1.Property<int>("MovimientoStockId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("MovimientoStockId");

b1.ToTable("MovimientosStock");

b1.WithOwner()

.HasForeignKey("MovimientoStockId");

});

b.Navigation("Articulo");

b.Navigation("Cantidad")

.IsRequired();

b.Navigation("TipoMovimiento");

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.TipoMovimiento.NombreTipoMovimiento", "NombreTipoMovimiento", b1 =>

{

b1.Property<int>("TipoMovimientoId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("TipoMovimientoId");

b1.ToTable("TiposMovimiento");

b1.WithOwner()

.HasForeignKey("TipoMovimientoId");

});

b.Navigation("NombreTipoMovimiento")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.Usuario", b =>

{

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.NombreUsuario", "Apellido", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.NombreUsuario", "Nombre", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.Contrasenia", "Contrasenia", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.OwnsOne("LogiaNegocio.ValueObjects.UsuarioVOs.EmailUsuario", "Email", b1 =>

{

b1.Property<int>("UsuarioId")

.HasColumnType("int");

b1.Property<string>("Valor")

.IsRequired()

.HasMaxLength(100)

.HasColumnType("nvarchar(100)");

b1.HasKey("UsuarioId");

b1.ToTable("Usuarios");

b1.WithOwner()

.HasForeignKey("UsuarioId");

});

b.Navigation("Apellido")

.IsRequired();

b.Navigation("Contrasenia")

.IsRequired();

b.Navigation("Email")

.IsRequired();

b.Navigation("Nombre")

.IsRequired();

});

modelBuilder.Entity("LogicaNegocio.Dominio.Parametro", b =>

{

b.OwnsOne("LogicaNegocio.ValueObjects.ParametroArticulo.ParametroValor", "Valor", b1 =>

{

b1.Property<int>("ParametroId")

.HasColumnType("int");

b1.Property<int>("Valor")

.HasColumnType("int");

b1.HasKey("ParametroId");

b1.ToTable("Parametros");

b1.WithOwner()

.HasForeignKey("ParametroId");

});

b.Navigation("Valor")

.IsRequired();

});

modelBuilder.Entity("LogiaNegocio.Dominio.TipoMovimiento", b =>

{

b.Navigation("MovimientosStock");

});

#pragma warning restore 612, 618

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\obj\Debug\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\obj\Debug\net8.0\LogicaDatos.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("LogicaDatos")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("LogicaDatos")]

[assembly: System.Reflection.AssemblyTitleAttribute("LogicaDatos")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\obj\Debug\net8.0\LogicaDatos.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\obj\Release\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\obj\Release\net8.0\LogicaDatos.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("LogicaDatos")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Release")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("LogicaDatos")]

[assembly: System.Reflection.AssemblyTitleAttribute("LogicaDatos")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\obj\Release\net8.0\LogicaDatos.GlobalUsings.g.cs

// <auto-generated/>

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Repositorios\DepositoContext.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.ValueObjects.TipoMovimiento;

using LogicaNegocio.Dominio;

using Microsoft.EntityFrameworkCore;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaDatos.Repositorios

{

public class DepositoContext : DbContext

{

public DbSet<Usuario> Usuarios { get; set; }

public DbSet<MovimientoStock> MovimientosStock { get; set; }

public DbSet<TipoMovimiento> TiposMovimiento { get; set; }

public DbSet<Articulo> Articulos { get; set; }

public DbSet<Parametro> Parametros { get; set; }

public DepositoContext(DbContextOptions options) : base(options)

{

}

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

//Articulo

modelBuilder.Entity<Articulo>()

.OwnsOne(a => a.CodigoArticulo);

modelBuilder.Entity<Articulo>()

.OwnsOne(a => a.Descripcion);

modelBuilder.Entity<Articulo>()

.OwnsOne(a => a.Nombre);

modelBuilder.Entity<Articulo>()

.OwnsOne(a => a.Precio);

modelBuilder.Entity<Articulo>()

.OwnsOne(a => a.Stock);

modelBuilder.Entity<Articulo>()

.HasOne(a => a.Parametro)

.WithOne()

.HasForeignKey<Articulo>(a => a.ParametroId);

//MovimientoStock

modelBuilder.Entity<MovimientoStock>()

.OwnsOne(m => m.Cantidad);

modelBuilder.Entity<MovimientoStock>()

.HasOne(ms => ms.TipoMovimiento)

.WithMany(tm => tm.MovimientosStock)

.HasForeignKey(ms => ms.TipoMovimientoId);

//Tipo de movimiento

modelBuilder.Entity<TipoMovimiento>()

.HasMany(t => t.MovimientosStock);

modelBuilder.Entity<TipoMovimiento>()

.OwnsOne(t => t.NombreTipoMovimiento);

//Usuario

modelBuilder.Entity<Usuario>()

.HasDiscriminator<string>("TipoUsuario")

.HasValue<Administrador>("Administrador")

.HasValue<EncargadoDeposito>("EncargadoDeposito");

modelBuilder.Entity<Usuario>()

.OwnsOne(u => u.Email);

modelBuilder.Entity<Usuario>()

.OwnsOne(u => u.Nombre);

modelBuilder.Entity<Usuario>()

.OwnsOne(u => u.Apellido);

modelBuilder.Entity<Usuario>()

.OwnsOne(u => u.Contrasenia);

//Parametro Articulo

modelBuilder.Entity<Parametro>()

.OwnsOne(pa => pa.Valor);

base.OnModelCreating(modelBuilder);

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Repositorios\RepositorioArticulos.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.ExcepcionesPropias;

using LogiaNegocio.InterfacesRepositorios;

using LogicaNegocio.Dominio;

using LogicaNegocio.InterfacesRepositorios;

using Microsoft.EntityFrameworkCore;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using static System.Runtime.InteropServices.JavaScript.JSType;

namespace LogicaDatos.Repositorios

{

public class RepositorioArticulos : IRepositorioArticulos

{

public DepositoContext Contexto { get; set; }

public RepositorioArticulos(DepositoContext contexto)

{

Contexto = contexto;

}

public IEnumerable<Articulo> FindAll()

{

return Contexto.Articulos.Include(a => a.Parametro).ToList();

}

public Articulo FindById(int id)

{

Articulo art = Contexto.Articulos.Where(a => a.Id == id).Include(a=>a.Parametro).SingleOrDefault();

if (art == null) throw new Exception("No se encontro el articulo buscado.");

return art;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Repositorios\RepositorioMovimientoStock.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.ExcepcionesPropias;

using LogiaNegocio.InterfacesRepositorios;

using LogicaNegocio.Dominio;

using Microsoft.EntityFrameworkCore;

using Microsoft.EntityFrameworkCore.Metadata;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaDatos.Repositorios

{

public class RepositorioMovimientoStock : IRepositorioMovimientosStock

{

DepositoContext Contexto { set; get; }

public RepositorioMovimientoStock(DepositoContext contexto)

{

Contexto = contexto;

}

public void Add(MovimientoStock obj)

{

if (obj == null) throw new DatosInvalidosException("El Movimiento de stock que se desea ingresar no es valido.");

Usuario usu = Contexto.Usuarios.Where(u => u.Email.Valor == obj.UsuarioEmail && u is EncargadoDeposito).SingleOrDefault();

if (usu == null) throw new DatosInvalidosException("El usuario que intenta ingresar el movimiento no esta habilitado");

Contexto.Entry(obj.Articulo).State = EntityState.Unchanged;

Contexto.Entry(obj.TipoMovimiento).State = EntityState.Unchanged;

obj.Fecha = DateTime.Now;

Contexto.MovimientosStock.Add(obj);

Contexto.SaveChanges();

}

public IEnumerable<MovimientoStock> FindAll()

{

return Contexto.MovimientosStock.Include(mov => mov.Articulo).Include(mov => mov.TipoMovimiento).ToList();

}

public MovimientoStock FindById(int id)

{

MovimientoStock ms = Contexto.MovimientosStock.Where(mov => mov.Id == id)

.Include(art => art.Articulo).SingleOrDefault();

if (ms == null) throw new DatosInvalidosException("No se encontró el movimiento de stock.");

return ms;

}

public void Remove(int id)

{

MovimientoStock aBorrar = FindById(id);

if (aBorrar != null)

{

Contexto.MovimientosStock.Remove(aBorrar);

Contexto.SaveChanges();

}

else

{

throw new DatosInvalidosException("El Movimiento de Stock no fue encontrado, no se pudo eliminar");

}

}

public void Update(MovimientoStock obj)

{

Contexto.MovimientosStock.Update(obj);

Contexto.SaveChanges();

}

public IEnumerable<MovimientoStock> ObtenerMovimientosPorArticuloYTipo(int articuloId, int tipoMovimiento)

{

return Contexto.MovimientosStock

.Where(m => m.Articulo.Id == articuloId && m.TipoMovimiento.Id == tipoMovimiento)

.Include(m => m.Articulo)

.Include(m => m.TipoMovimiento)

.Distinct()

.ToList();

}

public IEnumerable<MovimientoStock> ObtenerMovimientosPorArticuloYTipo(int articuloId, int tipoMovimiento, int pagina)

{

Parametro cantPaginas = Contexto.Parametros.Where(p => p.Nombre == "Paginacion").SingleOrDefault();

return Contexto.MovimientosStock

.Where(m => m.Articulo.Id == articuloId && m.TipoMovimiento.Id == tipoMovimiento)

.Include(m => m.Articulo)

.Include(m => m.TipoMovimiento)

.Distinct().Skip((pagina - 1) \* cantPaginas.Valor.Valor).Take(cantPaginas.Valor.Valor)

.ToList();

}

public IEnumerable<MovimientoStock> ObtenerMovimientosPorRangoDeFechas(DateTime inicio, DateTime fin)

{

DateTime finArreglado = fin.AddHours(23).AddMinutes(59).AddSeconds(59).AddMilliseconds(59).AddMicroseconds(59); //Para asegurar que se vea hasta la ultima consulta del dia indicado

return Contexto.MovimientosStock

.Where(m => m.Fecha >= inicio && m.Fecha <= finArreglado)

.Include(m => m.Articulo)

.Include(m => m.TipoMovimiento)

.Distinct()

.ToList();

}

public IEnumerable<MovimientoStock> ObtenerMovimientosPorRangoDeFechas(DateTime inicio, DateTime fin, int pagina)

{

Parametro cantPaginas = Contexto.Parametros.Where(p => p.Nombre == "Paginacion").SingleOrDefault();

DateTime finArreglado = fin.AddHours(23).AddMinutes(59).AddSeconds(59).AddMilliseconds(59).AddMicroseconds(59); //Para asegurar que se vea hasta la ultima consulta del dia indicado

return Contexto.MovimientosStock

.Where(m => m.Fecha >= inicio && m.Fecha <= finArreglado)

.Include(m => m.Articulo)

.Include(m => m.TipoMovimiento)

.Distinct().Skip((pagina - 1) \* cantPaginas.Valor.Valor).Take(cantPaginas.Valor.Valor)

.ToList();

}

public string ObtenerResumenMovimientos()

{

int año = DateTime.Now.Year;

var movimientosDelAño = Contexto.MovimientosStock

.Where(m => m.Fecha.Year == año).Include(tm => tm.TipoMovimiento)

.ToList();

if (movimientosDelAño.Count == 0)

{

return $"No se encontraron movimientos para el año {año}.";

}

var resumen = $"Artículos Movidos para el año {año}:\n";

var gruposPorTipo = movimientosDelAño

.GroupBy(m => m.TipoMovimiento.Id);

foreach (var grupo in gruposPorTipo)

{

var tipoMovimiento = grupo.FirstOrDefault().TipoMovimiento;

var cantidadTotal = grupo.Sum(m => m.Cantidad.Valor);

resumen += $"Tipo Movimiento: {tipoMovimiento.NombreTipoMovimiento.Valor} - Cantidad Unidades: {cantidadTotal}\n";

}

var totalAño = movimientosDelAño.Sum(m => m.Cantidad.Valor);

resumen += $"Total de movimientos en {año}: {totalAño}";

return resumen;

}

public double ObtenerCantidadPaginas(int totalMovimietos)

{

Parametro cantArticulosPorPagina = Contexto.Parametros.Where(p => p.Nombre == "Paginacion").SingleOrDefault();

var resultado = Math.Ceiling(totalMovimietos / (double)cantArticulosPorPagina.Valor.Valor);

return resultado;

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Repositorios\RepositorioTipoMovimiento.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.ExcepcionesPropias;

using LogicaNegocio.InterfacesRepositorios;

using Microsoft.EntityFrameworkCore;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using static System.Runtime.InteropServices.JavaScript.JSType;

namespace LogicaDatos.Repositorios

{

public class RepositorioTipoMovimiento : IRepositorioTipoMovimiento

{

public DepositoContext Contexto { get; set; }

public RepositorioTipoMovimiento(DepositoContext ctx)

{

Contexto = ctx;

}

public void Add(TipoMovimiento obj)

{

if (obj == null) throw new DatosInvalidosException("Datos inválidos para nuevo alta.");

TipoMovimiento t = BuscarPorNombre(obj.NombreTipoMovimiento.Valor);

if(t != null) throw new DatosInvalidosException("El nombre que desea usar ya esta en uso.");

Contexto.TiposMovimiento.Add(obj);

Contexto.SaveChanges();

}

public TipoMovimiento BuscarPorNombre(string nombre)

{

return Contexto.TiposMovimiento.Where(tipoMov => tipoMov.NombreTipoMovimiento.Valor == nombre).SingleOrDefault();

}

public IEnumerable<TipoMovimiento> FindAll()

{

return Contexto.TiposMovimiento.ToList();

}

public TipoMovimiento FindById(int id)

{

TipoMovimiento tm = Contexto.TiposMovimiento.Where(tipoMov => tipoMov.Id == id).SingleOrDefault();

if(tm == null) throw new DatosInvalidosException($"No existe tipo de movimiento con el Id: {id}");

return tm;

}

public void Remove(int id)

{

if (Contexto.MovimientosStock.Any(ms => ms.TipoMovimiento.Id == id)) throw new DatosInvalidosException("El Tipo de Movimiento no se pudo eliminar porque esta en uso.");

TipoMovimiento aBorrar = FindById(id);

if (aBorrar != null)

{

Contexto.TiposMovimiento.Remove(aBorrar);

Contexto.SaveChanges();

}

else

{

throw new DatosInvalidosException("El Tipo de Movimiento no fue encontrado, no se pudo eliminar");

}

}

public void Update(TipoMovimiento obj)

{

if (obj == null) throw new DatosInvalidosException("Los datos proporcionados no son validos");

TipoMovimiento t = BuscarPorNombre(obj.NombreTipoMovimiento.Valor);

if (t != null)

{

if (t.Id != obj.Id)

{

throw new DatosInvalidosException("No se puede editar porque el nombre ya está en uso");

}

else

{

Contexto.Entry(t).State = EntityState.Detached;

}

}

else if (t == null)

{

throw new DatosInvalidosException("No se encontro el tipo de movimiento");

}

Contexto.TiposMovimiento.Update(obj);

Contexto.SaveChanges();

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\LogicaDatos\Repositorios\RepositorioUsuarios.cs

using LogiaNegocio.Dominio;

using LogiaNegocio.ExcepcionesPropias;

using LogicaNegocio.InterfacesRepositorios;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LogicaDatos.Repositorios

{

public class RepositorioUsuarios : IRepositorioUsuarios

{

public DepositoContext Contexto { get; set; }

public RepositorioUsuarios(DepositoContext context)

{

Contexto = context;

}

public Usuario FindByEmail(string email)

{

return Contexto.Usuarios.Where(usu => usu.Email.Valor == email).SingleOrDefault();

}

public Usuario Login(string email, string password)

{

try

{

if (string.IsNullOrEmpty(email) || string.IsNullOrEmpty(password))return null;

Usuario usuario = FindByEmail(email);

if (usuario == null || string.IsNullOrEmpty(usuario.ContraseniaEncriptada))

{

return null;

}

string storedHashedPassword = usuario.ContraseniaEncriptada;

bool passwordMatches = BCrypt.Net.BCrypt.Verify(password, storedHashedPassword);

if (passwordMatches)

{

return usuario;

}

else

{

return null;

}

}

catch (Exception ex)

{

throw new DatosInvalidosException("Error en el proceso de inicio de sesión.");

}

}

public Task Logout(string email)

{

throw new NotImplementedException();

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\Program.cs

using DTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.InterfacesRepositorios;

using LogicaAplicacion.CasosUso;

using LogicaAplicacion.InterfacesCU;

using LogicaDatos.Repositorios;

using LogicaNegocio.InterfacesRepositorios;

using Microsoft.AspNetCore.Authentication.JwtBearer;

using Microsoft.EntityFrameworkCore;

using Microsoft.IdentityModel.Tokens;

namespace Web\_Api

{

public class Program

{

public static void Main(string[] args)

{

var builder = WebApplication.CreateBuilder(args);

//TipoMovimiento

builder.Services.AddScoped <ICUListar<TipoMovimientoDTO>, CUListarTiposMovimientosDTO>();

builder.Services.AddScoped <ICUBuscarPorId<TipoMovimientoDTO>, CUBuscarTipoMovimientoPorId>();

builder.Services.AddScoped <ICUModificar<TipoMovimientoDTO>, CUModificarTipoMovimientoDTO>();

builder.Services.AddScoped <ICUBaja, CUBajaTipoMovimientoDTO>();

builder.Services.AddScoped <ICUAlta<TipoMovimientoDTO>, CUAltaTipoMovimientoDTO>();

builder.Services.AddScoped<IRepositorioTipoMovimiento, RepositorioTipoMovimiento>();

//Usuario

builder.Services.AddScoped<ICULoginUsuario, CULoginUsuario>();

builder.Services.AddScoped<IRepositorioUsuarios, RepositorioUsuarios>();

//Articulo

builder.Services.AddScoped<ICUListar<ArticuloDTO>, CUListarArticulosDTO>();

builder.Services.AddScoped<IRepositorioArticulos, RepositorioArticulos>();

//MovimientoStock

builder.Services.AddScoped<ICUAlta<MovimientoStockDTO>, CUAltaMovimientoStockDTO>();

builder.Services.AddScoped<ICUBuscarPorId<MovimientoStockDTO>, CUBuscarMovimientoStockPorId>();

builder.Services.AddScoped<ICUListar<MovimientoStockDTO>, CUListarMovimientosDTO>();

builder.Services.AddScoped<ICUObtenerMovimientosPorArticuloYTipo, CUObtenerMovimientosPorArticuloYTipo>();

builder.Services.AddScoped<ICUObtenerMovimientosPorRangoDeFechas, CUObtenerMovimientosPorRangoDeFechas>();

builder.Services.AddScoped<ICUObtenerResumenMovimientos, CUObtenerResumenMovimientos>();

builder.Services.AddScoped<ICUObtenerCantidadPaginas, CUObtenerCantidadPaginas>();

builder.Services.AddScoped<IRepositorioMovimientosStock, RepositorioMovimientoStock>();

// Add services to the container.

string strCon = builder.Configuration.GetConnectionString("ConexionRodrigo");

builder.Services.AddDbContext<DepositoContext>(options => options.UseSqlServer(strCon));

builder.Services.AddControllers();

// Learn more about configuring Swagger/OpenAPI at https://aka.ms/aspnetcore/swashbuckle

builder.Services.AddEndpointsApiExplorer();

builder.Services.AddSwaggerGen();

//Token

var claveSecreta = "ZWRpw6fDo28gZW0gY29tcHV0YWRvcmE=";

builder.Services.AddAuthentication(aut => {

aut.DefaultAuthenticateScheme = JwtBearerDefaults.AuthenticationScheme;

aut.DefaultChallengeScheme = JwtBearerDefaults.AuthenticationScheme;

})

.AddJwtBearer(aut => {

aut.RequireHttpsMetadata = false;

aut.SaveToken = true;

aut.TokenValidationParameters = new TokenValidationParameters

{

ValidateIssuerSigningKey = true,

IssuerSigningKey = new SymmetricSecurityKey(System.Text.Encoding.ASCII.GetBytes(claveSecreta)),

ValidateIssuer = false,

ValidateAudience = false

};

});

var app = builder.Build();

// Configure the HTTP request pipeline.

if (app.Environment.IsDevelopment())

{

app.UseSwagger();

app.UseSwaggerUI();

}

app.UseAuthorization();

app.MapControllers();

app.Run();

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\Controllers\ArticuloController.cs

using DTOs;

using LogicaAplicacion.InterfacesCU;

using Microsoft.AspNetCore.Mvc;

// For more information on enabling Web API for empty projects, visit https://go.microsoft.com/fwlink/?LinkID=397860

namespace Web\_Api.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class ArticuloController : ControllerBase

{

public ICUListar<ArticuloDTO> CUListarArticulosDTO { get; set; }

public ArticuloController(ICUListar<ArticuloDTO> cuListarArticulosDTO)

{

CUListarArticulosDTO = cuListarArticulosDTO;

}

// GET: api/<ArticuloController>

[HttpGet]

public IActionResult Get()

{

try

{

IEnumerable<ArticuloDTO> articulos = CUListarArticulosDTO.Listar();

if (!articulos.Any()) return NotFound("No se encontraron artículos.");

return Ok(articulos);

}

catch

{

return StatusCode(500, "Error inesperado.");

}

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\Controllers\MovimientoStockController.cs

using DTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.ExcepcionesPropias;

using LogicaAplicacion.CasosUso;

using LogicaAplicacion.InterfacesCU;

using Microsoft.AspNetCore.Authorization;

using Microsoft.AspNetCore.Http.HttpResults;

using Microsoft.AspNetCore.Mvc;

using Microsoft.EntityFrameworkCore.Metadata.Builders;

using Microsoft.Extensions.Hosting;

// For more information on enabling Web API for empty projects, visit https://go.microsoft.com/fwlink/?LinkID=397860

namespace Web\_Api.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class MovimientoStockController : ControllerBase

{

public ICUListar<MovimientoStockDTO> CUListarMovimientos { get; set; }

public ICUAlta<MovimientoStockDTO> CUAltaMovimientoStock { get; set; }

public ICUBuscarPorId<MovimientoStockDTO> CUBuscarPorId { get; set; }

public ICUObtenerMovimientosPorArticuloYTipo CUObtenerMovimientosPorArticuloYTipo { get; set; }

public ICUObtenerMovimientosPorRangoDeFechas CUObtenerMovimientosPorRangoDeFechas { get; set; }

public ICUObtenerResumenMovimientos CUObtenerResumenMovimientos { get; set; }

public ICUObtenerCantidadPaginas CUObtenerCantidadPaginas { get; set; }

public MovimientoStockController(ICUListar<MovimientoStockDTO> cuListarMovimientos,

ICUAlta<MovimientoStockDTO> cuAltaMovimientoStock,

ICUBuscarPorId<MovimientoStockDTO> cuBuscarPorId,

ICUObtenerMovimientosPorArticuloYTipo cuObtenerMovimientosPorAticuloyTipo,

ICUObtenerMovimientosPorRangoDeFechas cuObtenerMovimientosPorRangoDeFechas,

ICUObtenerResumenMovimientos cuObtenerResumenMovimientos,

ICUObtenerCantidadPaginas cuObtenerCantidadPaginas)

{

CUAltaMovimientoStock = cuAltaMovimientoStock;

CUBuscarPorId = cuBuscarPorId;

CUListarMovimientos = cuListarMovimientos;

CUObtenerMovimientosPorArticuloYTipo = cuObtenerMovimientosPorAticuloyTipo;

CUObtenerMovimientosPorRangoDeFechas = cuObtenerMovimientosPorRangoDeFechas;

CUObtenerResumenMovimientos = cuObtenerResumenMovimientos;

CUObtenerCantidadPaginas = cuObtenerCantidadPaginas;

}

// GET: api/<MovimientoStockController>

[HttpGet]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult Get()

{

IEnumerable<MovimientoStockDTO> movimientos = CUListarMovimientos.Listar();

return Ok(movimientos);

}

// GET api/<MovimientoStockController>/5

[HttpGet("{id}", Name= "BuscarPorIdMovimientoStock")]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult Get(int id)

{

if (id <= 0) return BadRequest("No se pueden ingresar id menor o igual a 0.");

return Ok(CUBuscarPorId.GetById(id));

}

// POST api/<MovimientoStockController>

[HttpPost]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult Post([FromBody] MovimientoStockDTO msDTO)

{

if (msDTO == null) return BadRequest("Los datos proporcionados son inválidos");

if (msDTO.ArticuloId == 0 || msDTO.TipoMovimientoId == 0) return BadRequest("No se ingresaron datos de articulo o tipo de movimiento.");

try

{

CUAltaMovimientoStock.Alta(msDTO);

int nuevoId = msDTO.Id;

return CreatedAtRoute("BuscarPorIdMovimientoStock", new { id = nuevoId }, msDTO);

}

catch (DatosInvalidosException ex)

{

return BadRequest(ex.Message);

}

catch (Exception)

{

return StatusCode(500, "Error inesperado.");

}

}

[HttpGet("articulo/{ArticuloId}/tipo/{MovimientoId}")]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult ConsultarMovimientosPorArticuloYTipo(int ArticuloId, int MovimientoId)

{

if (ArticuloId <= 0) return BadRequest("Seleccione el Articulo");

if (MovimientoId <= 0) return BadRequest("Seleccione el Tipo de Movimiento");

try

{

var movimientos = CUObtenerMovimientosPorArticuloYTipo.ObtenerMovimientosPorArticuloYTipo(ArticuloId, MovimientoId);

return Ok(movimientos);

}

catch (Exception)

{

return StatusCode(500, "Error inesperado.");

}

}

[HttpGet("articulo/{ArticuloId}/tipo/{MovimientoId}/pagina{pagina}")]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult ConsultarMovimientosPorArticuloYTipo(int ArticuloId, int MovimientoId, int pagina)

{

if (ArticuloId <= 0) return BadRequest("Seleccione el Articulo");

if (MovimientoId <= 0) return BadRequest("Seleccione el Tipo de Movimiento");

if (pagina == null || pagina <= 0)

{

var movimientos = CUObtenerMovimientosPorArticuloYTipo.ObtenerMovimientosPorArticuloYTipo(ArticuloId, MovimientoId, 1);

return Ok(movimientos);

}

else

{

var movimientos = CUObtenerMovimientosPorArticuloYTipo.ObtenerMovimientosPorArticuloYTipo(ArticuloId, MovimientoId, pagina);

return Ok(movimientos);

}

}

[HttpGet("fechaInicio{fechaInicio}/fechaFin{fechaFin}")]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult ConsultarMovimientosPorFecha(DateTime fechaInicio, DateTime fechaFin)

{

if (fechaInicio == null || fechaFin == null) return BadRequest("Ingrese Fecha de Inicio y Fecha de Fin");

if (fechaInicio > fechaFin) return BadRequest("La Fecha de Inicio no puede ser mayor que la Fecha de Fin");

try

{

var articulos = CUObtenerMovimientosPorRangoDeFechas.ObtenerMovimientosPorRangoFechas(fechaInicio, fechaFin);

return Ok(articulos);

}

catch (Exception ex)

{

return StatusCode(500, "Error interno.");

}

}

[HttpGet("fechaInicio{fechaInicio}/fechaFin{fechaFin}/pagina{pagina}")]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult ConsultarMovimientosPorFecha(DateTime fechaInicio, DateTime fechaFin, int pagina)

{

if (fechaInicio == null || fechaFin == null) return BadRequest("Ingrese Fecha de Inicio y Fecha de Fin");

if (fechaInicio > fechaFin) return BadRequest("La Fecha de Inicio no puede ser mayor que la Fecha de Fin");

if (pagina == null || pagina <= 0)

{

var articulos = CUObtenerMovimientosPorRangoDeFechas.ObtenerMovimientosPorRangoFechas(fechaInicio, fechaFin, 1);

return Ok(articulos);

}

else

{

var articulos = CUObtenerMovimientosPorRangoDeFechas.ObtenerMovimientosPorRangoFechas(fechaInicio, fechaFin, pagina);

return Ok(articulos);

}

}

[HttpGet("resumen")]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult ObtenerResumenMovimientos()

{

var resumen = CUObtenerResumenMovimientos.ObtenerResumenMovimientos();

return Ok(resumen);

}

[HttpGet("cantidadPaginas/{totalMovimientos}")]

[Authorize(Roles = "EncargadoDeposito")]

public IActionResult ObtenerCantidadPaginas(int totalMovimientos)

{

try

{

var resultado = CUObtenerCantidadPaginas.ObtenerCantidadPaginas(totalMovimientos);

return Ok(resultado);

}

catch (DatosInvalidosException ex)

{

return BadRequest(ex.Message);

}

catch (Exception ex)

{

return StatusCode(500, "Error interno");

}

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\Controllers\TipoMovimientosController.cs

using DTOs;

using LogiaNegocio.ExcepcionesPropias;

using LogicaAplicacion.InterfacesCU;

using Microsoft.AspNetCore.Authorization;

using Microsoft.AspNetCore.Mvc;

using Microsoft.EntityFrameworkCore.ChangeTracking;

// For more information on enabling Web API for empty projects, visit https://go.microsoft.com/fwlink/?LinkID=397860

namespace Web\_Api.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class TipoMovimientosController : ControllerBase

{

public ICUListar<TipoMovimientoDTO> CUListarTiposMovimientos { get; set; }

public ICUBuscarPorId<TipoMovimientoDTO> CUBuscarTipoMovimientoPorID { get; set; }

public ICUModificar<TipoMovimientoDTO> CUActualizarTipoMovimientoPorID { get; set; }

public ICUBaja CUBaja { get; set; }

public ICUAlta<TipoMovimientoDTO> CUAltaTipoMovimientoDTO { get; set; }

public TipoMovimientosController(ICUListar<TipoMovimientoDTO> cuListarTiposMovimientos, ICUBuscarPorId<TipoMovimientoDTO> cuBuscarTipoMovimientoPorID,

ICUModificar<TipoMovimientoDTO> cuActualizarTipoMovimientoPorID, ICUBaja cuBaja, ICUAlta<TipoMovimientoDTO> cuAltaTipoMovimientoDTO)

{

CUListarTiposMovimientos = cuListarTiposMovimientos;

CUBuscarTipoMovimientoPorID = cuBuscarTipoMovimientoPorID;

CUActualizarTipoMovimientoPorID = cuActualizarTipoMovimientoPorID;

CUBaja = cuBaja;

CUAltaTipoMovimientoDTO = cuAltaTipoMovimientoDTO;

}

[HttpGet]

public IActionResult Get() {

IEnumerable<TipoMovimientoDTO> tipoMovimientos = CUListarTiposMovimientos.Listar();

return Ok(tipoMovimientos);

}

// GET api/<TipoMovimientosController>/5

[HttpGet("{id}", Name = "BuscarPorId")]

public IActionResult Get(int id)

{

if (id <= 0) return BadRequest("El id del tipo de movimiento deber ser un entero positivo");

try

{

TipoMovimientoDTO tp = CUBuscarTipoMovimientoPorID.GetById(id);

return Ok(tp);

}

catch (DatosInvalidosException ex)

{

return BadRequest(ex.Message);

}

catch (Exception ex)

{

return StatusCode(500, "Error inesperado");

}

}

// POST api/<TipoMovimientosController>

[HttpPost]

public IActionResult Post([FromBody] TipoMovimientoDTO tpDTO)

{

if (tpDTO == null) return BadRequest("Se proporcionaron datos inválidos.");

try

{

CUAltaTipoMovimientoDTO.Alta(tpDTO);

return CreatedAtRoute("BuscarPorId", new { id = tpDTO.Id }, tpDTO);

}

catch (DatosInvalidosException ex)

{

return BadRequest(ex.Message);

}

catch (Exception)

{

return StatusCode(500, "Ocurrió un error inesperado.");

}

}

// PUT api/<TipoMovimientosController>/5

[HttpPut("{id}")]

public IActionResult Put(int id, [FromBody] TipoMovimientoDTO tpDTO)

{

if(id <= 0) return BadRequest("El id del tipo de movimiento deber ser un entero positivo.");

if (tpDTO == null) return BadRequest("Datos a actualizar inválidos.");

if (id != tpDTO.Id) return BadRequest("No se proporcionaron los id correctos.");

try

{

CUActualizarTipoMovimientoPorID.Update(tpDTO);

return Ok();

}

catch (DatosInvalidosException ex)

{

return BadRequest(ex.Message);

}

catch (Exception ex)

{

return StatusCode(500, "Error inesperado.");

}

}

// DELETE api/<TipoMovimientosController>/5

[HttpDelete("{id}")]

public IActionResult Delete(int id)

{

if (id <= 0) return BadRequest("El id del tipo de movimiento proporcionado deber ser un entero positivo");

try

{

CUBaja.Baja(id);

return Ok();

}

catch (DatosInvalidosException ex)

{

return BadRequest(ex.Message);

}

catch (Exception ex)

{

return StatusCode(500, "Error inesperado. Intente mas tarde");

}

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\Controllers\UsuariosController.cs

using DTOs;

using LogiaNegocio.Dominio;

using LogiaNegocio.ExcepcionesPropias;

using LogiaNegocio.ValueObjects.UsuarioVOs;

using LogicaAplicacion.CasosUso;

using LogicaAplicacion.InterfacesCU;

using LogicaDatos.Repositorios;

using Microsoft.AspNetCore.Authorization;

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Identity.Data;

using Microsoft.AspNetCore.Mvc;

using System.Runtime.InteropServices;

using System.Security.Claims;

using WebAPI.Token;

namespace Web\_Api.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class UsuariosController : ControllerBase

{

public ICULoginUsuario LoginUsuario { get; set; }

public UsuariosController(ICULoginUsuario loginUsuario)

{

LoginUsuario = loginUsuario;

}

[HttpPost("Login")]

public IActionResult Login([FromBody] LoginDTO usu)

{

if (string.IsNullOrEmpty(usu.Email) || string.IsNullOrEmpty(usu.Contrasena))

return BadRequest("Ingrese email y contraseña.");

try

{

UsuarioDTO usuario = LoginUsuario.Login(usu.Email, usu.Contrasena);

if (usuario != null)

{

return Ok(new UsuarioLogueadoDTO()

{

EmailUsuario = usuario.Email,

RolUsuario = usuario.Rol,

Token = WebAPI.Token.ManejadorToken.CrearToken(usuario)

});

}

else

{

return NotFound("Email o Contraseña incorrecto.");

}

}

catch (DatosInvalidosException ex)

{

return StatusCode(500, ex.Message);

}

catch (Exception ex)

{

return StatusCode(500, "Algo salió mal.");

}

}

}

}

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\obj\Debug\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\obj\Debug\net8.0\Web Api.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("Web Api")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("Web Api")]

[assembly: System.Reflection.AssemblyTitleAttribute("Web Api")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\obj\Debug\net8.0\Web Api.GlobalUsings.g.cs

// <auto-generated/>

global using global::Microsoft.AspNetCore.Builder;

global using global::Microsoft.AspNetCore.Hosting;

global using global::Microsoft.AspNetCore.Http;

global using global::Microsoft.AspNetCore.Routing;

global using global::Microsoft.Extensions.Configuration;

global using global::Microsoft.Extensions.DependencyInjection;

global using global::Microsoft.Extensions.Hosting;

global using global::Microsoft.Extensions.Logging;

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Net.Http.Json;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\obj\Debug\net8.0\Web Api.MvcApplicationPartsAssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: Microsoft.AspNetCore.Mvc.ApplicationParts.ApplicationPartAttribute("Swashbuckle.AspNetCore.SwaggerGen")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\obj\Release\net8.0\.NETCoreApp,Version=v8.0.AssemblyAttributes.cs

// <autogenerated />

using System;

using System.Reflection;

[assembly: global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v8.0", FrameworkDisplayName = ".NET 8.0")]

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\obj\Release\net8.0\Web Api.AssemblyInfo.cs

//------------------------------------------------------------------------------

// <auto-generated>

// Este cÃ³digo fue generado por una herramienta.

// VersiÃ³n de runtime:4.0.30319.42000

//

// Los cambios en este archivo podrÃ­an causar un comportamiento incorrecto y se perderÃ¡n si

// se vuelve a generar el cÃ³digo.

// </auto-generated>

//------------------------------------------------------------------------------

using System;

using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("Web Api")]

[assembly: System.Reflection.AssemblyConfigurationAttribute("Release")]

[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]

[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]

[assembly: System.Reflection.AssemblyProductAttribute("Web Api")]

[assembly: System.Reflection.AssemblyTitleAttribute("Web Api")]

[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generado por la clase WriteCodeFragment de MSBuild.

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\obj\Release\net8.0\Web Api.GlobalUsings.g.cs

// <auto-generated/>

global using global::Microsoft.AspNetCore.Builder;

global using global::Microsoft.AspNetCore.Hosting;

global using global::Microsoft.AspNetCore.Http;

global using global::Microsoft.AspNetCore.Routing;

global using global::Microsoft.Extensions.Configuration;

global using global::Microsoft.Extensions.DependencyInjection;

global using global::Microsoft.Extensions.Hosting;

global using global::Microsoft.Extensions.Logging;

global using global::System;

global using global::System.Collections.Generic;

global using global::System.IO;

global using global::System.Linq;

global using global::System.Net.Http;

global using global::System.Net.Http.Json;

global using global::System.Threading;

global using global::System.Threading.Tasks;

///// File: C:\Users\rodri\Downloads\Obligatorio 2 P3 18-06 Tarde\Obligatorio 2 P3\OBLIGATORIO 2\WebAPI\Web Api\Token\ManejadorToken.cs

using DTOs;

using Microsoft.IdentityModel.Tokens;

using System.IdentityModel.Tokens.Jwt;

using System.Security.Claims;

using System.Text;

namespace WebAPI.Token

{

public class ManejadorToken

{

public static string CrearToken(UsuarioDTO usu)

{

JwtSecurityTokenHandler tokenHandler = new JwtSecurityTokenHandler();

//clave secreta, generalmente se incluye en el archivo de configuración

//Debe ser un vector de bytes

byte[] clave = Encoding.ASCII.GetBytes("ZWRpw6fDo28gZW0gY29tcHV0YWRvcmE=");

//Se incluye un claim para el email

SecurityTokenDescriptor tokenDescriptor = new SecurityTokenDescriptor

{

Subject = new ClaimsIdentity(new Claim[] {

new Claim(ClaimTypes.Email, usu.Email),

new Claim(ClaimTypes.Role, usu.Rol)

}),

Expires = DateTime.UtcNow.AddMonths(1),

SigningCredentials = new SigningCredentials(new SymmetricSecurityKey(clave),

SecurityAlgorithms.HmacSha256Signature)

};

var token = tokenHandler.CreateToken(tokenDescriptor);

return tokenHandler.WriteToken(token);

}

}

}