

Item
<b>Varios</b>
Datos de prueba (scripts con insert en la BD o equivalente)
<b>Obligatoriamente: 80 artículos, 30 movimientos de stock y 10 registros de cada una de las demás tablas.</b>
Implementado en VS 2022, .NET 8, EF 8
<b>Documentación</b>
PDF único
Incluye archivo astah
Carátula con nombre y foto integrantes
Incluye código fuente de toda la aplicación
Agregar en la documentación los prompts para demostrar uso de IA por lo menos para la precarga de artículos
Diagrama de casos de uso
Diagrama de clases : Dominio, persistencia (interfaces repos y repos concretos), interfaces capa de aplicación , web api, mvc (sin views)
<b>Arquitectura / Diseño / Estilo</b>
Soluciones separadas, una para Web API y otra para el cliente
La solución que incluye Web API tiene las capas solicitadas (negocio / persistencia o acceso a datos / aplicación/ api )
Cada capa resuelve sus responsabilidades estrictamente.
Incluye las entidades y value objects necesarios
Entidades implementan interfaces para validación
Errores de validación lanzan excepciones
Accede a la BD exclusivamente desde la capa de acceso a datos
MVC no conoce capa de datos a ningún proyecto de la otra solución
Métodos resuelven responsabilidad única
Captura excepciones.
Web api esta documentada y devuelve los codigos de estado correctos
Utilización de DTOS en controladores de Web API
Utilización de View Model o Dtos en MVC
Usa exclusivamente Linq para consultas
Usa exclusivamente EF Core para acceso a datos
Define las interfaces adecuadas teniendo en cuenta que los movimientos de stock no permiten edición ni borrado
Diagrama de clases de las distintas capas y diagrama de caso de uso modelado en forma correcta, con relaciones, metodos, navegabilidad y mulplicidad en cada relación. Tipos de relaciones correctas

<b>0 . Login (visitante sin identificar) / Logout (usuario autenticado activo) - MVC</b>
Verifica credenciales (email y pwd) consumiendo la Web API
Redirige a Login o portada si no es autenticado.
No permite ingresar por url a requerimientos no habilitados(salvo login)
La web api genera token en forma correcta
Cierra sesión del usuario corriente y redirige a login.
<b>1. CRUD Tipos de movimientos - Web API</b>
Alta, baja, modificación, obtención de un único tipo y obtención de todos los tipos disponibles implementados correctamente.
No es posible modificar ni eliminar tipos que estén siendo utilizados en algún movimiento.
Pruebas exitosas utilizando Postman o Swagger.
<b>2. Ingreso de Movimientos de Stock (MVC HttpClient + Web API):</b>
Solo accesible para usuarios logueados y con rol Operador
Solicita a la web api los articulos, tipos de movimiento necesarios y envía token
Interfaz de usuario permite ingresar todos los datos necesarios para dar de alta un movimiento
Valida datos ingresados incluido que la cantidad no sobrepase el tope definido
Movimientos de stock registrados correctamente en la base de datos.
Si no se cumplen las validaciones , muestra mensaje al usuario
<b>3. Consultas (Web API + MVC) - Autorización con JWT:</b>
Utiliza JWT para todas las consultas
Consulta a: Devuelve todos los datos del movimiento de stock y del tipo de movimiento tomando en cuenta el artículo recibido. Si no hay resultados muestra mensaje
Consulta b: Devuelve todos los artículos con al menos 1 movimiento entre las fechas recibidas. Si no hay resultados muestra n
Consulta c: Muestra resumen de cantidades agrupadas por año y tipo de movimiento. Si no hay resultados muestra mensaje
Paginado: pagina en forma correcta las consultas a y b tomando en cuenta el valor prefijado . No se muestra si no hay resultados
<b>7. Manejo de parámetros configurables</b>
Permite el manejo de los siguientes parámetros:
Tope cantidad a mover
Cantidad de items por página
<b>Documentación personalizada de la webapi (incluir documentación xml para swagger)</b>
Api: Documenta nombre y version
Endpoints:
Incluye información de tipos de respuesta, Parámetros, objetivo