

<b>EVALUACIÓN</b>	Obligatorio 2	<b>GRUPO</b>	Todos	<b>FECHA</b>	20/05/2024
<b>MATERIA</b>	Programación 3				
<b>CARRERA</b>	Analista Programador / Analista en Tecnologías de la Información				
<b>CONDICIONES</b>	<p><b>- Puntos: 35</b></p> <p><b>- Fecha de entrega: 20/06/2024 hasta las 21:00 horas en <a href="http://gestion.ort.edu.uy">gestion.ort.edu.uy</a>.</b></p> <p><b>Uso de material de apoyo y/o consulta</b></p> <p><u>Inteligencia Artificial Generativa</u></p> <ul style="list-style-type: none"> <li>- Para el desarrollo del obligatorio es posible utilizar Inteligencia Artificial Generativa (IAG) siguiendo las pautas indicadas por el docente de la materia. De ser utilizada esta herramienta es obligatorio describir los “prompts” más relevantes que han sido utilizados.</li> <li>- En caso de existir dudas sobre la autoría, plagio o uso no atribuido de IAG el docente tendrá la opción de convocar al equipo de obligatorio a una defensa específica e individual sobre el tema</li> </ul> <p><u>Cualquier otro material</u></p> <ul style="list-style-type: none"> <li>- Deberá ser documentado de acuerdo a las pautas definidas en el documento 302</li> </ul> <p><b>Entrega:</b></p> <p>La entrega se realizará en forma Online en archivo NO MAYOR a 40Mb en formato zip, rar o pdf-</p> <p><b>IMPORTANTE:</b></p> <ul style="list-style-type: none"> <li>- Inscribirse</li> <li>- Formar grupos de hasta 2 personas del mismo dictado</li> <li>- Aquellos de ustedes que presenten alguna dificultad con su inscripción o tengan inconvenientes técnicos, por favor contactarse con el Coordinador o Coordinación adjunta <b>antes de las 20:00hs.</b> del día de la entrega, a través de los mails <a href="mailto:alamon@ort.edu.uy">alamon@ort.edu.uy</a> y <a href="mailto:fernandez_ma@ort.edu.uy">fernandez_ma@ort.edu.uy</a>, o telefónicamente al 29021505 - int 1156 u 1138</li> <li>- Subir el trabajo a Gestión antes de la hora indicada, ver hoja al final del documento: “RECORDATORIO”</li> </ul>				

### INFORMACIÓN IMPORTANTE:

- Las consultas sobre la letra deberán realizarse a través del foro de aulas.
- La no asistencia a la defensa implica la pérdida de puntos del obligatorio para el estudiante. En caso de fuerza mayor deberá avisar al docente mediante mail y a coordinación por Teams para analizar si se autoriza la programación de una nueva fecha.
- **Tamaño de subida en Gestión:**  
Si el tamaño del archivo comprimido que entrega supera los 40Mb puede eliminar de la solución las carpetas bin, obj y packages, preservando los archivos propios que

pueda haber incluido en estas carpetas (ej.: archivos de texto, imágenes, etc.). Se recomienda enfáticamente mantener una versión con los paquetes incluidos para prevenir problemas durante la descarga de paquetes.

- **Se deberá incluir el código fuente en el PDF de la documentación como salvaguarda de cualquier inconveniente que se pueda producir al intentar abrir/ejecutar la aplicación durante la defensa y la corrección.**
- **El formato de la defensa se definirá previo a la entrega.**
- **Las aclaraciones realizadas por los docentes a través del foro de aulas serán consideradas parte integral de la letra, por lo que es responsabilidad de los estudiantes acceder asiduamente al foro.**

Continuando el desarrollo del sistema anterior, agregaremos la gestión de algunas funcionalidades del área de Depósito de la empresa.

El encargado de depósito no puede acceder a la base de datos, por lo que deberá consumir una Web Api usando HttpClient desde una aplicación MVC totalmente independiente (al menos serán dos soluciones .NET, una que incluya dominio, capa de aplicación y de acceso a datos, y la otra solución solamente incluirá el cliente MVC HttpClient).

A los usuarios con rol administrador se agrega el Usuario con rol Encargado de depósito.

Las entidades ya implementadas que participan de esta segunda etapa serán solamente los usuarios y artículos implementados en el Obligatorio 1.

Del relevamiento surge que al menos deberemos considerar las siguientes entidades:

### **Movimientos de Stock**

Los movimientos de stock deberán tener un Id autonumérico, fecha y hora del movimiento, artículo que se “mueve”, tipo de movimiento de stock, usuario que lo llevó a cabo, y la cantidad de unidades que se mueven.

Estos movimientos no modifican el atributo de stock del artículo correspondiente a la parte 1, por lo que se convierte en irrelevante. Podrá conservarlo o eliminarlo, según prefiera. Si no lo elimina de la clase, deberá ignorarlo.

### **Tipos de movimiento:**

Los tipos de movimiento constan de un Id, un nombre único, (por ejemplo “Venta”, “Devolución”, “Compra”, etc.).

En algunos casos el tipo de movimiento corresponde a una reducción del stock, y en otros casos a un aumento.

Nota: Si el movimiento es un traslado (ej.: cambio de estantería, dentro del depósito, una caja con resmas de papel) no se registra un movimiento de stock. No habrá en este sistema la posibilidad de registrar tipos de movimiento que no impliquen un cambio en el stock.

**Se deberá cumplir con los siguientes requerimientos funcionales:**

**Por el momento solamente podrá acceder a estos requerimientos un usuario con rol "Encargado".**

### **RF01 - CRUD de Tipos de Movimiento (WebApi + Postman o Swagger – Sin autenticación).**

Se deberá implementar la Webapi que exponga el alta, baja, modificación, obtener un único tipo, y retornar todos los tipos disponibles.

No se podrán modificar ni eliminar tipos que ya estén siendo utilizados en algún movimiento.

No se implementará un cliente MVC que consuma la Webapi, que solo se probará a través de un cliente como Postman o Swagger.

### **RF02– Autenticación – Login (MVC HttpClient + WEB Api +JWT) /Logout**

- Dadas las credenciales usuario y contraseña se desarrollará un login o inicio de sesión. Se deberá utilizar un token JWT.

**Sugerencia:** a pesar de estar planteado al principio de este texto, se sugiere resolver los demás puntos para aplicarles posteriormente el control de acceso.

### **RF03–Ingreso de Movimientos de Stock – Autorización Con JWT (MVC HttpClient + WEB API)**

Se debe proveer una pantalla que capture los datos necesarios para que un encargado de depósito pueda llevar adelante el registro de un movimiento de stock sobre un artículo.

- El registro de este movimiento constará de:
  - El artículo al que se le aplica el movimiento
  - La fecha (incluyendo la hora) – Será la fecha del Sistema.
  - La cantidad (unidades)
  - El tipo de movimiento
  - El email del Usuario que lo llevó a cabo

Se validará que:

- Las cantidades sean positivas.
- No se pueden realizar movimientos que involucren más de un tope prefijado de unidades. Ese tope no debe ser fijo, podría cambiar.

- El tipo de movimiento de stock sea uno de los existentes.
  - El usuario exista, y se haya autenticado con rol encargado.
  - El artículo será uno de los existentes.
  - Solo se manejan ingresos y egresos.
- No es posible eliminar ni modificar movimientos de stock. Esta restricción se deberá tomar en cuenta para definir las interfaces que corresponda.

**Estas operaciones deberán ser autorizadas utilizando un token JWT.**

#### **RF-04-Consultas (WEB API +MVC) - Autorización Con JWT**

Los encargados del depósito pueden consultar los movimientos:

- a. Dados un id de artículo y un tipo de movimiento, todos los movimientos de ese tipo realizados sobre ese artículo. Deberán estar ordenados descendientemente por fecha y en forma ascendente por la cantidad de unidades involucradas en el movimiento. Se deberá mostrar todos los datos del movimiento, incluyendo todos los datos de su tipo.
- b. Dado un rango de fechas, seleccionar los artículos que hayan tenido movimientos (al menos uno) entre esas fechas inclusive.
- c. Obtener la información de resumen de las cantidades movidas agrupadas por año, y dentro de año por tipo de movimiento.

**Año: XXXX**

Tipo:A – Cantidad: 800

Tipo:B – Cantidad:150

**Total año XXXX : 950**

**Año: XXXY**

Tipo A – Cantidad: 200

Tipo B – Cantidad: 180

**Total año XXXY : 380**

.....  
.....

- d. Se deberá implementar el paginado de consultas. Las consultas a y b se presentarán en forma paginada, y el máximo de ítems que se podrán retornar en cada página será menor o igual a un valor prefijado.

**Se pide:**

1. Resolver los requisitos anteriores mediante el uso de HttpClient en los requerimientos que incluyan el consumo de la WebApi a través de MVC. Se excluyen aquellos requerimientos que

---

explícitamente indiquen que solo se consume a través de Postman / Swagger.

2. Realizar una precarga de:  
Artículos de prueba mediante la ayuda de Chat GPT. Se deberán incluir en la documentación la captura y el link a la secuencia de los prompts utilizados. Se requieren como mínimo 80 artículos diferentes. Los datos deberán ser fácilmente identificables y coherentes con la realidad planteada.
3. Movimientos, tipos, usuarios, topes.  
Deberá haber registros suficientes y coherentes para probar el funcionamiento de todos los requerimientos. En particular deberá haber al menos 30 movimientos de stock ingresados en la base de datos previamente a la defensa.

### **ENTREGA FINAL EN GESTIÓN:**

Se subirá a gestión un único archivo comprimido que contenga:

1. **IMPLEMENTACIÓN.**

Archivo comprimido en formato .zip, .rar o .7z con la solución funcionando completa en .NET 8.0, usando Entity Framework Core 8.0, aplicación MVC con los requerimientos mencionados y servicio WebApi. El MVC deberá ser una solución independiente, y solo podrá acceder a las funcionalidades mediante HttpClient.

Las CRUD de los tipos de movimiento se probarán mediante Swagger y/o Postman, no se les implementa MVC.

2. **Precarga de datos.**

Se precargarán datos en la base según se mencionó anteriormente. Si requiere datos que no estén explícitamente indicados deberá incluirlos, también.

3. **PDF con la documentación siguiente:**

- a. Carátula con el nombre, número de estudiante y foto de los integrantes del grupo.
- b. Tabla de contenido con la página en que se ubica cada ítem de la documentación.
- c. Diagrama de casos de uso
- d. Diagrama de clases (UML) para todas las capas implementadas. Se excluyen las Views de MVC. Los diagramas deberán ser legibles. Se deberá incluir la Web Api.
- e. Código fuente de toda la aplicación incluido en la documentación.
- f. Archivo Astah con los diagramas anteriormente solicitados.
- g. Scripts con los juegos de datos de prueba.

Cuando no se haya explicitado la cantidad de registros previamente en el presente documento, se espera que el juego de datos tenga al menos 10 registros de cada tabla. Se incluirán tanto los esquemas como los insert con los datos. Se deberán incluir tablas y filas

tanto para aquellas entidades que no deben registrarse a través del sistema (las precargas explicitadas anteriormente) como para aquellos registros que surgen del sistema en funcionamiento (artículos, movimientos, tipos, etc.).

#### Reglas generales:

- Las magnitudes serán positivas o cero.
- A menos que se especifique lo contrario todos los datos son requeridos.
- Cuando corresponda se deberán implementar Value Objects.
- Las validaciones, errores, etc. se manejarán a través de excepciones.
- Todas las entidades se identificarán mediante un Id autonumérico secuencial (otorgado por la BD, inmutable).
- En la aplicación Web cada vez que se produzca un error en las ABM, o cuando no se obtengan resultados para las solicitudes de información se deberá informar convenientemente al usuario. **En el caso de las Web Api se deberán retornar los status code adecuados según la guía REST.**
- Cuando sea necesaria información para la operativa y su registro no sea explícitamente solicitado en los requerimientos, será cargada manualmente en la base de datos.
- Las consultas se realizarán a través de LINQ **exclusivamente**.
- Se deberán utilizar View Models y DTO.
- 
- **En el caso de utilizar código generado mediante inteligencia artificial generativa, deberá estar en condiciones de explicarlo en la defensa y deberá incluir el link a los prompts utilizados.**

Este proyecto se realizará usando biblioteca de clase, proyecto Web MVC, proyecto Web Api de .NET 8, C# como lenguaje de programación, Entity Framework 8 para el acceso a la base de datos y Linq como lenguaje de consulta. Se preferirá sintaxis de método.

Se deberá organizar la aplicación siguiendo las recomendaciones de Arquitectura Limpia (Clean Architecture) y DDD (Domain Driven Design).


---

## RECORDATORIO: IMPORTANTE PARA LA ENTREGA

### • Obligatorios

La entrega de los obligatorios será en formato digital online, a excepción de algunas materias que se entregarán en Bedelía y en ese caso recibirá información específica en el dictado de la misma.

Los principales aspectos a destacar sobre la **entrega online de obligatorios** son:

1. Ingresá al sistema de Gestión.
2. En el menú, seleccioná el ítem “Evaluaciones” y la instancia de evaluación correspondiente, que figura bajo el título “Inscripto”.
3. Para iniciar la entrega hacé clic en el ícono: 
4. Ingresá el número de estudiante de cada uno de los integrantes y hacé clic en “Agregar”. El sistema confirmará que los integrantes estén inscriptos al obligatorio y, de ser así, mostrará el nombre y la fotografía de cada uno de ellos. Una vez agregados todos los integrantes, hacé clic en “Crear equipo”.

**Cualquier integrante podrá:**

- **Modificar la integración del equipo.**
- **Subir el archivo de la entrega.**

5. Seleccioná el archivo que deseás entregar. Verificá el nombre del archivo que aparecerá en la pantalla y hacé clic en “Subir” para iniciar la entrega. Cada equipo (hasta 2 estudiantes) debe entregar **un único archivo en formato zip o rar** (los documentos de texto deben ser pdf, y deben ir dentro del zip o rar). El archivo a subir debe tener **un tamaño máximo de 40mb**

Cuando el archivo quede subido, se mostrará el nombre generado por el sistema (1), el tamaño y la fecha en que fue subido.

6. El sistema enviará un e-mail a todos los integrantes del equipo informando los detalles del archivo entregado y confirmando que la entrega fue realizada correctamente.
7. Podés cerrar la pestaña de entrega y continuar utilizando Gestión o salir del sistema.
8. La **hora tope para subir el archivo será las 21:00** del día fijado para la entrega.
9. La entrega se podrá realizar desde cualquier lugar (ej. hogar del estudiante, laboratorios de la Universidad, etc).
10. Aquellos de ustedes que presenten alguna dificultad con su inscripción o tengan inconvenientes técnicos, por favor contactarse con el Coordinador o Coordinación adjunta **antes de las 20:00hs.** del día de la entrega, a través de los mails [alamon@ort.edu.uy](mailto:alamon@ort.edu.uy) y [fernandez\\_ma@ort.edu.uy](mailto:fernandez_ma@ort.edu.uy), o telefónicamente al 29021505 - int 1156 u 1138.

---

Si tuvieras una situación particular de fuerza mayor, debes contactarte con suficiente antelación al plazo de entrega, al Coordinador de Cursos ([alamon@ort.edu.uy](mailto:alamon@ort.edu.uy)) o Secretario Docente Adj. ([gervaz@ort.edu.uy](mailto:gervaz@ort.edu.uy)).