



EVALUACIÓN PARCIAL

Enunciado:

Objetivo:

Poner en práctica lo aprendido acerca de programación en Java y desarrollo de Microservicios. Para el desarrollo del presente parcial será necesario conocer:

- El lenguaje de Programación Java y sus diferentes librerías
- El administrador de proyectos Maven, su configuración y uso
- El framework Spring y los componentes específicos analizados en clase hasta el momento
- Los conceptos de desarrollo de microservicios y testing unitario revisados.

En cuanto al objetivo del presente **Enunciado Específico** tiene por objeto completar el Pre-enunciado, que el alumno tuvo disponible con anterioridad, y especificar un conjunto de requerimientos a cumplir de manera individual en el día del parcial.

Introducción:

En el pre-enunciado se describió la base de datos Northwind suministrada para la tarea previa y sobre la que se solicitó al alumno construya los elementos CRUD elementales para cierto conjunto de tablas que hoy aprovecharemos para resolver los requerimientos que se solicitan en la consigna.

Ante cualquier duda sobre el pre-enunciado el alumno tiene disponible tanto el mismo pre-enunciado como la base de datos suministrada en el aula virtual en el momento del parcial puede consultarlo o revisarlo para resolver cualquier duda al respecto.

Consigna:

El alumno deberá construir 2 endpoints específicos y tests unitarios de los servicios asociados a estos endpoints, de acuerdo con los siguientes requerimientos, de los endpoints siguientes se espera que el alumno construya la capa de servicio y la capa de controlador aprovechando la capa de datos construida previamente, aunque no se impide que el alumno agregue elementos a la capa de datos si así lo considera necesario.

1. Obtener todos los Productos dados un proveedor (Supplier.SupplierID), una Categoría (Categories.CategoryID) y una cantidad esperada de stock de seguridad. (URL sugerida: GET:/api/productos?supplierid=<id de proveedor>&categoryid=<id de categoría>&stockmin=<cantidad esperada de stock>). Para ello deberá buscar todos los productos que sean provistos por el proveedor indicado y correspondan a la categoría suministrada y, cuyo stock futuro, la suma de UnitsInStock más UnitsOnOrder sea menor que la cantidad provista y donde Discontinued sea distinto de '1'.



El resultado debe informar: ProductId, ProductName, StockFuturo (UnitsInStock + UnitsOnOrder) y UnitPrice, ordenados de menor a mayor por StockFuturo.

Se espera que la respuesta tenga código 200 si responde correctamente, 204 si el id de proveedor e id de categoría existen, pero no hay productos que cumplan la condición y 404 en el caso que no exista el id de proveedor y/o el id de categoría.

Se espera que el test unitario pruebe al menos estas 3 situaciones.

2. Crear un nuevo Pedido (Order) en base a:
 - a. id de proveedor (SupplierID) – filtro,
 - b. id de categoría (CategoryID) – filtro,
 - c. stockRequerido (número entero) – filtro,
 - d. id de cliente (CustomerID) – dato,
 - e. id de empleado (EmployeeID) – dato,
 - f. id de correo (ShipperID) – dato,
 - g. y una duración máxima a cubrir.

Que cree un pedido (Order) para el cliente indicado donde el detalle de productos (Order Detail) cumpla con las condiciones que se mencionan a continuación. (URL sugerida: POST o PUT /api/orders, con un que body contenga los datos mencionados).

- h. El pedido debe incluir todos aquellos productos del proveedor y categoría indicada cuyo stock futuro (UnitsInStock + UnitsOnOrder) sea menor que stockRequerido que llega como parámetro,
- i. La cantidad de productos a pedir será la diferencia entre el stockRequerido y el stock futuro.
- j. El precio unitario es el UnitPrice del producto
- k. y el descuento se asignarán como 0 si la cantidad pedida es menor que 100 y 0.10 en caso contrario.
- l. Por otro lado, en la tabla Order, los ids son los suministrados, las fechas las puede manejar como prefiera, el número de transporte (Freight) será 0 por que aún no está asignado y los datos de Ship son la copia de la dirección del cliente.

El resultado debe responder con la estructura completa del pedido y su detalle indicando además la cantidad total de productos pedidos. Además, debe retornar el código 200 o 201 si la operación fue correcta, 204 en caso de que no exista productos en condiciones de ser pedidos y 404 en caso de que no exista alguno de los ids relacionados suministrados.

Instrucciones de entrega:

Clonar su repositorio personal o trabajar sobre el proyecto donde construyó la etapa previa desde <https://labsys.frc.utn.edu.ar/gitlab/backend-app/parciales-2023/3K#/#####>, donde debe reemplazar los # con su comisión y su legajo.

Construir las consignas especificadas previamente

Al finalizar el parcial deberá – Lea atentamente:



1. Hacer push en su repositorio personal según la URL mencionada anterior mente, **condicionante de corrección**.
2. Comprimir el código fuente del proyecto (SIN la carpeta “target” ni la carpeta “.idea” o de configuración del IDE que utilice) y subirlo a la entrega del parcial en el aula virtual **condicionante de corrección**.
3. Una vez realizada la entrega NO REALIZAR NINGÚN TIPO DE MODIFICACIÓN SOBRE EL REPOSITORIO.

Evaluación

Los criterios de evaluación se pueden observar en la rúbrica de corrección. Sin embargo, es importante para esta cátedra mencionar el esfuerzo que se hará para intentar valorar el proceso de aprendizaje independientemente del resultado final.

Entre los criterios más importantes a evaluar están, las habilidades actitudinales revistas en el cumplimiento de las condiciones de entrega, el aprendizaje de los temas vistos en clase en la utilización de las herramientas solicitadas y la capacidad de resolución de problemas en las estrategias utilizadas para generar los resultados.

Se adjunta la tabla de calificaciones documentada en la planificación de cátedra de Backend de Aplicaciones.

NOTA	PORCENTAJE	CALIFICACIÓN
1		Insuficiente
2		Insuficiente
3		Insuficiente
4		Insuficiente
5		Insuficiente
6	60% a 68%	Aprobado
7	69% a 77%	Bueno
8	78% a 86%	Muy Bueno
9	87% a 95%	Distinguido
10	96% a 100%	Sobresaliente