

Fecha:			
Nombre:			
Hora Inicial:		Hora Final:	

Evaluación Técnica – Desarrolladores Frontend - backend
--

1. FRONT END

Elaborar una aplicación en **Angular2 o Superior** para la administración de un maestro de productos que cumpla con el caso de uso descrito a continuación, debe de ser desarrollado utilizando los componentes NG-Prime (<https://www.primefaces.org/primeng/#/>)

Funcionalmente se requiere

1. Crear un home básico con
 - a. Menú
 - b. Barra superior
 - c. Sección de perfil (Foto, Nombre, Opción de Log in, Opción de Logout)
2. Crear un formulario de registro de productos con los siguientes campos:
 - a. ID
 - b. Nombre del Producto
 - c. Características (campo tipo texto)
 - d. Fecha de lanzamiento
 - e. Correo fabricante
 - f. País de fabricación
 - i. Usar el siguiente servicio (<https://restcountries.eu/#api-endpoints-response-example>) u otro en desde internet
 - g. Precio en formato de moneda
 - h. Unidades disponibles
 - i. Unidades vendidas
 - j. Imagen del producto
3. El formulario debe tener las siguientes acciones
 - a. Guardar
 - b. Cerrar
4. Se debe crear una vista de consulta de productos con los campos del formulario

5. La vista debe permitir las siguientes acciones:

- a. Crear un producto (ir al formulario)
- b. Eliminar un producto
- c. Filtrar por 2 campos del informe
- d. Ordenar por 2 columnas del informe
- e. Pagar los registros

El modelo de datos puede ser creado en la herramienta que prefiera. Cree por lo menos 20 referencias de productos para verificar la funcionalidad de la vista.

2. FRONT END

A partir de las siguientes tablas de datos creadas en Table Storage de Azure

Nombre de la base de datos:

123459876

Nombre de la tabla:

Documentos

Key1:

X9XmzmV0xX6AG/WYCXSA8BrrCF0M4j3X4zFDgJZ5g9hRwPafsljAWk2kgjLZmj6YSxJGlikJDsLmvxgEfGWTQ==

Conexion string:

DefaultEndpointsProtocol=https;AccountName=123459876;AccountKey=X9XmzmV0xX6AG/WYCXSA8BrrCF0M4j3X4zFDgJZ5g9hRwPafsljAWk2kgjLZmj6YSxJGlikJDsLmvxgEfGWTQ==;EndpointSuffix=core.windows.net

BlobEndpoint=https://123459876.blob.core.windows.net/;QueueEndpoint=https://123459876.queue.core.windows.net/;FileEndpoint=https://123459876.file.core.windows.net/;TableEndpoint=https://123459876.table.core.windows.net/;SharedAccessSignature=sv=2017-11-09&ss=bfqt&srt=sco&sp=rwdlacup&se=2018-06-16T08:08:59Z&st=2018-06-16T00:08:59Z&spr=https&sig=Aejde534rfvWYOR9Gt82XkmmaePSZWYCs7f7XADSA5e8%3D

SAS token

?sv=2017-11-09&ss=bfqt&srt=sco&sp=rwdlacup&se=2018-06-16T08:08:59Z&st=2018-06-16T00:08:59Z&spr=https&sig=Aejde534rfvWYOR9Gt82XkmmaePSZWYCs7f7XADSA5e8%3D

blob service SAS URL

<https://123459876.blob.core.windows.net/?sv=2017-11-09&ss=bfqt&srt=sco&sp=rwdlacup&se=2018-06-16T08:08:59Z&st=2018-06-16T00:08:59Z&spr=https&sig=Aejde534rfvWYOR9Gt82XkmmaePSZWYCsf7XADSA5e8%3D>

Table service SAS URL

<https://123459876.table.core.windows.net/?sv=2017-11-09&ss=bfqt&srt=sco&sp=rwdlacup&se=2018-06-16T08:08:59Z&st=2018-06-16T00:08:59Z&spr=https&sig=Aejde534rfvWYOR9Gt82XkmmaePSZWYCsf7XADSA5e8%3D>

Se requiere

1. Crear una API en Visual Studio .NET que permita realizar un CRUD.
2. La operación de consulta del CRUD deberá permitir paginación
3. Crear un Job que se ejecute cada minuto y actualice la hora de revisión de todos los productos en estado de revisión.

La documentación de Table Storage puede ser encontrada en

<https://docs.microsoft.com/es-es/rest/api/storageservices/table-service-rest-api#rest-api-operations>