

Nº44 junio 2020



REVISTA ESPECIALIZADA EN TECNOLOGÍAS MICROSOFT

Entrevista  
Olga Martí

El Cómic Como  
Medio de Enseñanza  
de Conceptos de  
Programación en  
C# en Ámbitos  
Universitarios

Construyendo  
máquinas de es-  
tados con Web-  
hook y Logic App

Project Cortex:  
un Microsoft  
365 mucho más  
inteligente

## Staff

CompartiMOSS es una publicación independiente de distribución libre en forma electrónica. Las opiniones aquí expresadas son de estricto orden personal, cada autor es completamente responsable de su propio contenido.

### DIRECCIÓN GENERAL

- Gustavo Velez
- Juan Carlos Gonzalez
- Fabian Imaz
- Alberto Diaz

### DISEÑO Y DIAGRAMACIÓN

- Santiago Porras Rodríguez

## Contacte con nosotros

[revista@compartimoss.com](mailto:revista@compartimoss.com)  
[gustavo@gavd.net](mailto:gustavo@gavd.net)  
[jcgonzalezmartin1978@hotmail.com](mailto:jcgonzalezmartin1978@hotmail.com)  
[fabian@siderys.com.uy](mailto:fabian@siderys.com.uy)  
[adiazcan@hotmail.com](mailto:adiazcan@hotmail.com)

### BLOGS

<http://www.gavd.net>  
<http://geeks.ms/blogs/jcgonzalez>  
<http://blog.siderys.com>  
<http://geeks.ms/blogs/adiazmartin>

### REDES SOCIALES

Facebook:  
<http://www.facebook.com/group.php?gid=128911147140492>  
 LinkedIn:  
<http://www.linkedin.com/groups/CompartiMOSS-3776291>  
 Twitter:  
[@CompartiMOSScom](https://twitter.com/CompartiMOSScom)

# Contenido

03

Editorial

07

Detección de objetos con Custom Vision

15

Introducción al Cumplimiento de Comunicación en Microsoft 365

23

Entrevista Olga Martí

30

Graph Toolkit en profundidad

40

Modern Search con PnP WebParts - Introducción

47

El Cómic Como Medio de Enseñanza de Conceptos de Programación en C# en Ámbitos Universitarios

52

El Learning Pathways de Microsoft 365

57

CosmosDB Autopilot

04

Procesando Modelos Tabulares

11

Project Cortex: un Microsoft 365 mucho más inteligente

19

Productivity Tips con Microsoft Teams: Últimas novedades en reuniones, llamadas y videollamadas en Teams

25

Introducción al almacén de conocimiento (Knowledge Store)

33

Construyendo máquinas de estados con Webhook y Logic App

43

Dimensionando desescalapp.com en Azure

50

Uso de Azure Monitor

55

Entrevista Intelequia



03

## Editorial

Nuestro principal objetivo como media de comunicación es compartir conocimiento entre nuestros lectores, conocimiento tecnológico que permita hacer un mundo mejor. En estos momentos, en los que tenemos problemas mundiales con una pandemia como el COVID-19, cuando nos damos cuenta de la importancia de la tecnología, de lo relevante que se ha vuelto poder trabajar desde casa o disponer de suficiente capacidad de cálculo para ejecutar algoritmos matemáticos que buscan una solución a este virus. Es en estos momentos cuando esperamos que los artículos, que nuestros autores escriben número a número con mucho esfuerzo, sirvan para ayudar a usar la tecnología para combatir el virus, quedándonos en casa, mejoran nuestros procesos empresariales, nuestros modelos de colaboración, etc.

Llevamos 44 números llegando a vuestras pantallas con artículos pensados para aprender un poquito más sobre la tecnología, pensados para que vuestros proyectos sean mejores. En esta editorial queremos dar las gracias a nuestros numerosos lectores porque número a número buscan el conocimiento que se escribe en nuestras páginas. GRACIAS por ayudarnos a hacer un mundo mejor.

**El Equipo Editorial de CompartiMOSS**



## Procesando Modelos Tabulares

En el presente artículo nos centraremos en formas de procesamiento sobre modelos tabulares. Desde un tiempo atrás y en diversos proyectos me he encontrado con casos en los que nuestra capa semántica esta diferenciada de nuestra capa de reportabilidad.

Es por ello por lo que se me ocurrió compartir la experiencia, para que otros colegas y entusiastas de las diversas plataformas de datos puedan contar con cierto material de apoyo, y de ciertos trabajos de investigación que tuve que realizar para poder implementar en mis últimos proyectos.

A continuación, una de las arquitecturas de referencia, para que podamos conversar acerca de los componentes y servicios utilizados:

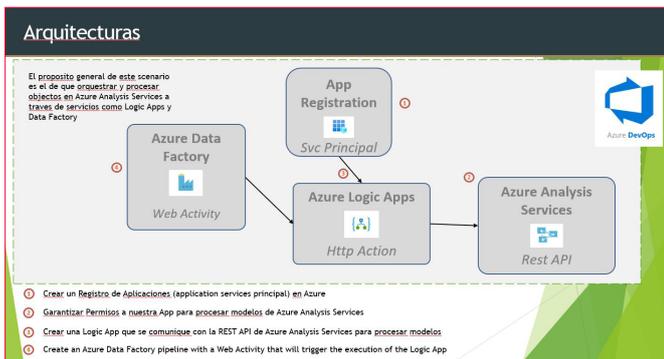


Imagen 1.- Arquitectura de referencia.

Una de las ventajas de la arquitectura presentada arriba es el hecho de automatizar totalmente el proceso de refresco de nuestro modelo de datos. Pasemos a contar los pasos que debemos realizar para la automatización del proceso.

### Creación de nuestro Registro de Aplicaciones

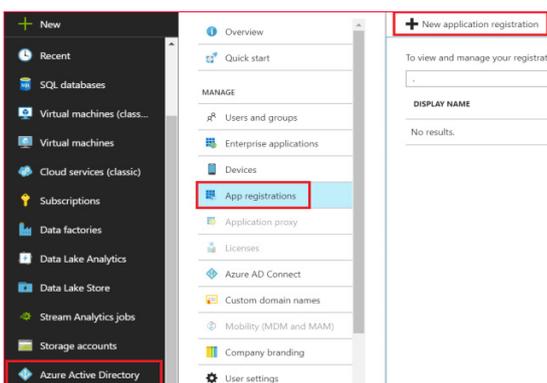


Imagen 2.- Nuevo registro de Aplicación en Azure AD.

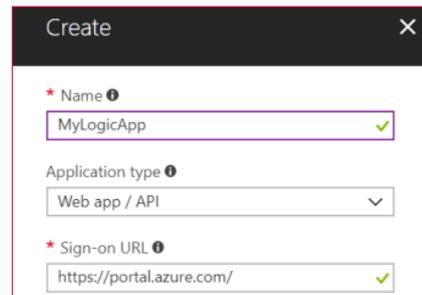


Imagen 3.- Datos para la App registrada.

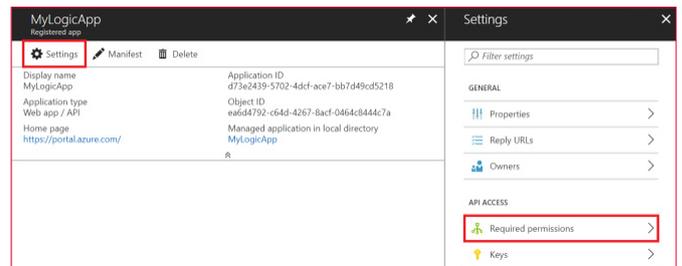


Imagen 4.- Permisos para la App.

El objetivo principal de crear un registro de aplicación es el de delegar a una cuenta (aplicación de servicio) los permisos necesarios para el procesamiento de nuestro modelos de Azure Analysis Services (AAS).

**"en diversos proyectos me he encontrado con casos en los que nuestra capa semántica esta diferenciada de nuestra capa de reportabilidad"**

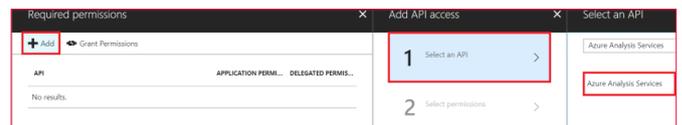


Imagen 5.- Permisos para AAS.

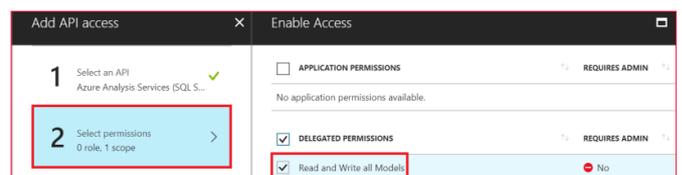


Imagen 6.- Permisos configurados para AAS.

En el paso anterior lo que implementamos, una vez generada nuestra Aplicación, la delegación de permisos necesarios para que dicha App pueda leer y escribir sobre

nuestros modelos de datos en AAS. Nuevamente, y a continuación lo que tenemos que hacer es generar una clave (Key) para que nuestra aplicación tenga acceso de manera segura. De dichos settings será necesario contar con nuestro App ID, Tenant ID, y valor de nuestra clave:

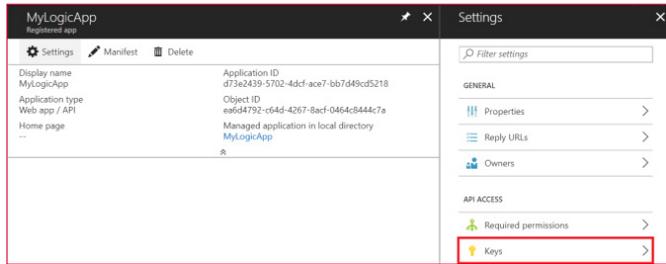


Imagen 7.- Generación de la clave para la App.

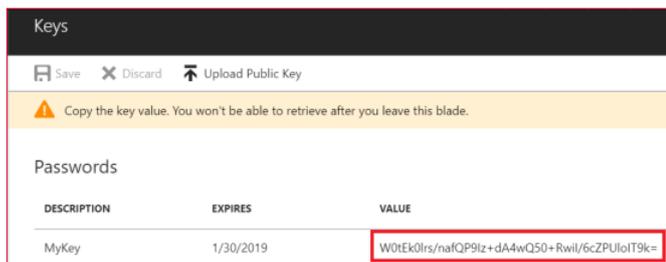


Imagen 8.- Clave generada.

En este procedimiento, deberemos agregar nuestra App para que sea posible procesar modelos. Por lo tanto, debemos acceder a SQL Server Management Studio, conectarnos a nuestra instancia de Azure Analysis Services y en las propiedades de nuestra instancia, acceder a Seguridad y agregar nuestra App, en el formato de App ID y AAD ID (tenant).

La sintaxis tendrá el siguiente formato: app:<app guid>@<tenant guid>

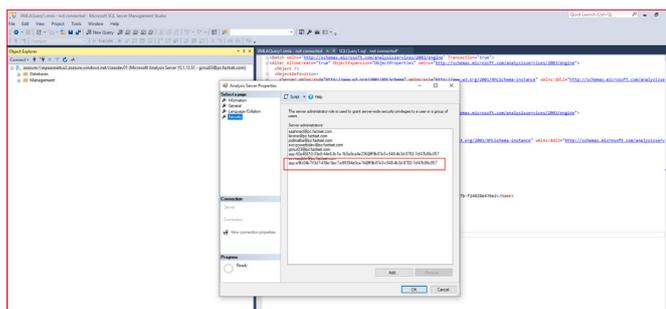


Imagen 9.- Conexión a AAS

## Creando nuestra Logic App para el Procesamiento

En nuestro portal de Azure buscamos Logic Apps y creamos nuestra aplicación comenzando con un disparador (Trigger) que ejecutará cada vez que un Request HTTP es recibido:

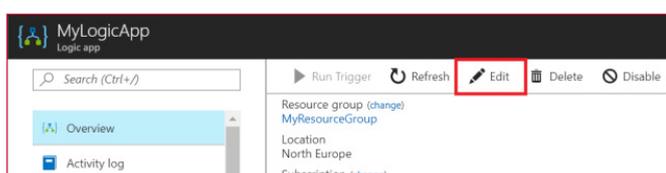


Imagen 10.- Edición de la Logic App requerida para el procesamiento.

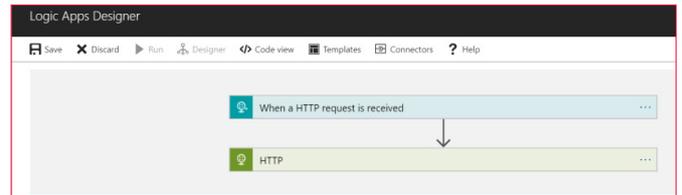


Imagen 11.- Logic App.

## Configuración de Logic Apps

En nuestro siguiente paso, y luego de agregar un objeto HTTP a nuestra Logic Apps, configuramos dicha acción, utilizando un Método POST, y definiendo como URI:

URI: `https://<region>.as.azure.windows.net/servers/<server-name>/models/<modelName>/refreshes`

En el Body, desde la documentación de la REST API de AAS accedemos a /Refreshes que nos permitirá refrescar nuestro modelo, con varias opciones de procesamiento. Podemos procesar todo el modelo (database), una tabla específica, o incluso una partición.

**"debemos acceder a SQL Server Management Studio, conectarnos a nuestra instancia de Azure Analysis Services y en las propiedades de nuestra instancia, acceder a Seguridad y agregar nuestra App"**

En las propiedades de autenticación definimos los siguientes parámetros:

- Tenant: Utilizamos el AAD ID desde el registro de aplicaciones que anteriormente anotamos
- Audiencia: `https://*.as.azure.windows.net`
- Client ID: Usamos el App ID de nuestro registro de aplicación
- Secret: Nuestra clave de aplicación (Key)

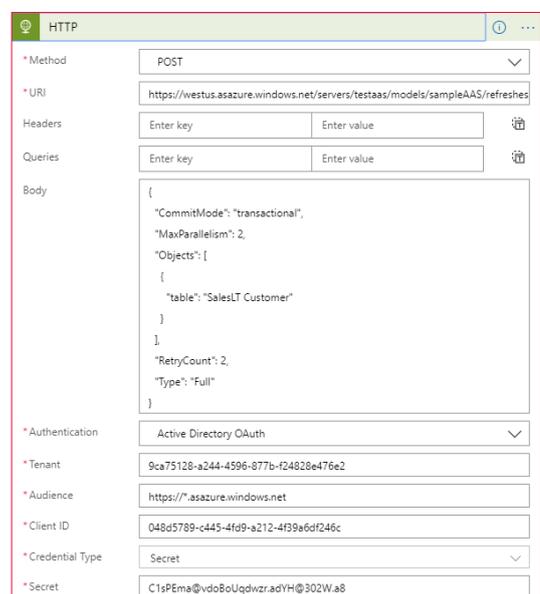


Imagen 12.- Configuración del objeto HTTP en la Logic App.

## Testing de nuestra Logic Apps

Desde el diseñador de nuestra Lógica Apps, le damos clic a Run, y abrimos la acción HTTP para ver la respuesta de la REST API y el código del mensaje recibido. Luego que la Logic Apps ejecuta y está lista para ser utilizada en Azure Data Factory podremos abrir la acción de "When a HTTP request is received" y copiarnos la HTTP POST URL

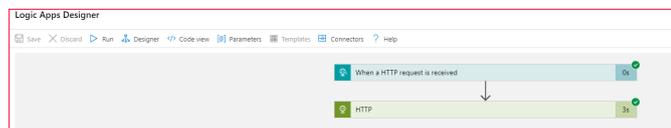


Imagen 13.- Prueba de la Logic App.

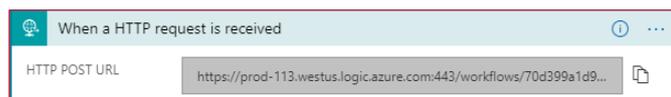


Imagen 14.- HTTP POST URL.

**"podemos lograr automatizar totalmente el proceso de refresco de nuestros modelos tabulares"**

## Creación de Instancia de Azure Data Factory

En el portal de Azure nos creamos una nueva instancia de Azure Data Factory definiendo nombre, versión, suscripción, localización, y podemos hasta vincularla a un repositorio GIT.

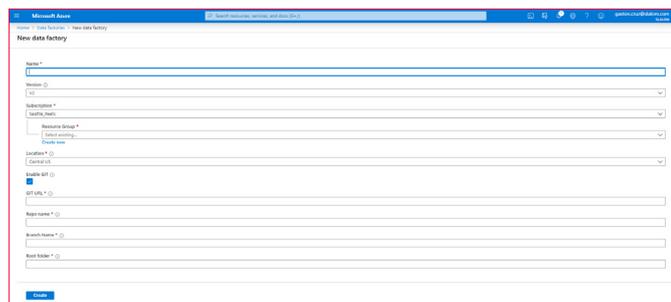


Imagen 15.- Creación de la instancia de Azure Data Factory.

## Diseño de Pipeline en Azure Data Factory

Al crear nuestra instancia de Azure Data Factory debemos dirigirnos luego a la sección de Author & Monitor para crear nuestro primer pipeline. En la pantalla de diseño vamos a crear nuestro pipeline, y simplemente arrastraremos al canvas un componente de Actividad Web.

Luego configuramos la Actividad Web:

- URL: usamos el HTTP POST URL de nuestra Logic App
- Method: POST
- Body (obligatorio): {"message": "test"}

Ejecutamos un Test Run para chequear el resultado:

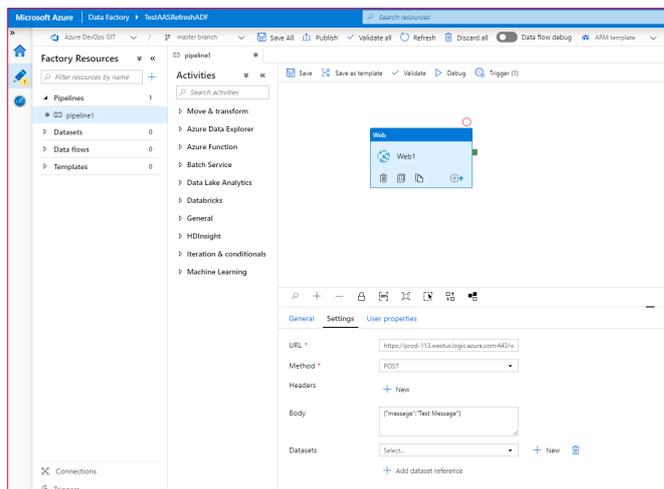


Imagen 16.- Ejecución del Pipeline.

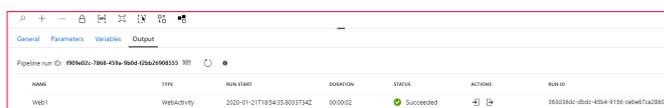


Imagen 17.- Resultado obtenido.

## Definiendo Trigger en Azure Data Factory

Si queremos que nuestro proceso corra a determinada hora, o cada cierto minuto para procesar el modelo de AAS, podemos crear un Trigger. Vamos a la opción Triggers (sección inferior izquierda de nuestra pantalla). Definimos el nombre, y descripción del trigger. Configuramos una fecha de comienzo de ejecución, y la recurrencia (minutos, horas, días, semanas, o meses).

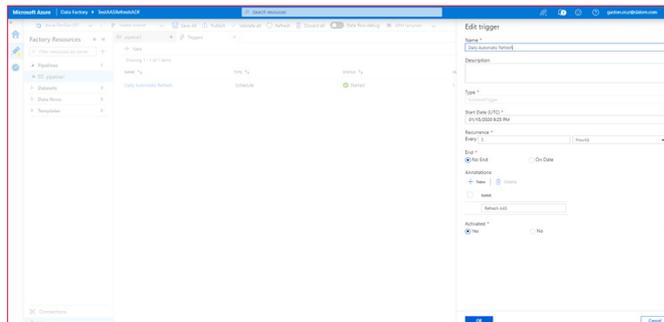


Imagen 18.- Definición del trigger de Data Factory.

## Conclusión

Mediante el procedimiento que explicamos en este artículo podemos lograr automatizar totalmente el proceso de refresco de nuestros modelos tabulares. Y no solo eso, sino que además podemos agregar un repositorio y automatizar el proceso para diferentes entornos (Azure DevOps) que explicaremos en próximos artículos.

**GASTÓN CRUZ**  
Solutions Architect - Slalom USA  
MVP Data Platform  
@gastonfcruz

# Detección de objetos con Custom Vision

Dentro de los servicios cognitivos de Microsoft, en la categoría de visión podemos encontrar Custom Vision, sin duda un servicio muy interesante para reconocimiento de objetos dentro de una imagen. En los servicios cognitivos es posible reconocer objetos cotidianos como una bicicleta, un móvil o vehículos, pero para detectar un objeto concreto debemos entrenar un modelo. Primero debemos contar con una suscripción en Azure (recuerda que Azure ofrece unas cuentas trial para probar este servicio entre otros). Una vez en el portal de Azure crearemos un recurso Custom Vision, se puede localizar en el Marketplace a través del buscador o en el apartado IA y Machine Learning.

no debiese salir de una ubicación y la predicción debiese estar lo más cercana posible del usuario.

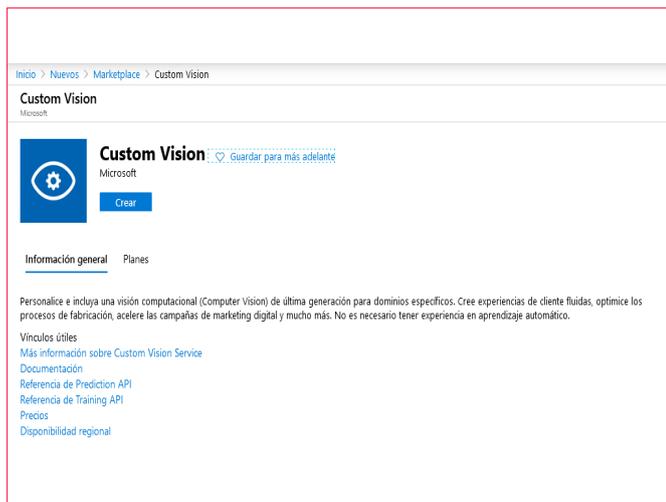


Imagen 1.- Creación de un recurso Custom Vision.

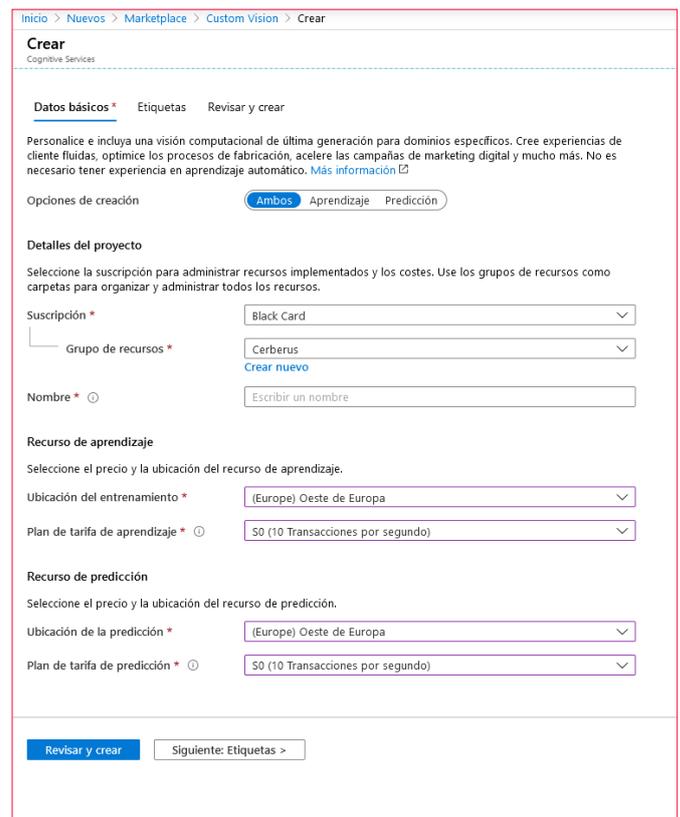


Imagen 2.- Configuración del recurso Custom Vision.

Le debemos indicar una suscripción, un grupo de recursos, el nombre que también será el dominio que aparecerá en el endpoint y planes y ubicación tanto del recurso de aprendizaje como de predicción. En la imagen aparecen planes S0 porque en mi caso ya tengo una cuenta en uso con el plan F0 (gratuito) y solo se permite una por suscripción.

Una vez creado el recurso podemos trabajar con el a través de la API o de una manera más visual en [www.customvision.ai](http://www.customvision.ai), en esta web usaremos el mismo usuario y contraseña que hemos empleado para el portal de Azure y nos reconocerá todos los recursos. Es posible que se quede colgada refrescando sin parar con el mensaje de "Loading your accounts...", me ha pasado algunas veces dependiendo del navegador y puede ser debido a dos motivos:

**"Custom Vision, sin duda un servicio muy interesante para reconocimiento de objetos dentro de una imagen"**

- Un bloqueador de anuncios que este bloqueando alguna petición, mejor desactívalo para esta URL.
- Que no cargue correctamente los proyectos recientes, la URL intentará entrar a `/projects?mostRecentDirectory` simplemente dejándolo en `/projects` entrará sin problema y una vez dentro cargará los últimos proyectos.

Podríamos configurar el aprendizaje y la predicción en ubicaciones distintas, pero esta opción solo se me ocurre por motivos legales con los datos en los que el entrenamiento

Una vez dentro pulsaremos en New Project y nos aparecerá la siguiente ventana:

Imagen 3.- Creación de nuevo proyecto.

En ella indicaremos un nombre de proyecto, el recurso al que va asociado (el que creamos antes en el portal o podemos crear uno desde aquí) y el tipo del proyecto:

- Clasificación. Que etiquetará la imagen dentro de unas clasificaciones.
- Detección de objeto. Que buscará los objetos con el que lo hemos entrenado dentro de la imagen y sus ubicaciones.

En nuestro caso vamos a usar detección de objeto y por lo tanto no tendremos la opción Classification Types. El sistema nos proporciona unos dominios entrenados, como lo queremos para un objeto concreto personalizado elegimos la opción General. En nuestro caso vamos a entrenarlo para que detecte a Rodolfo, la mascota de ENCAMINA.



Imagen 4.- Imagen a detectar.

Y vamos a intentar que lo detecte en sus distintas modalidades de merchandising: muñeco plano, muñeco cuadrado, pegatina, llavero y ambientador. Una vez dentro del proyecto, en el menú nos encontraremos las siguientes opciones:



Imagen 5.- Opciones en el modelo.

En Training Images veremos todas las imágenes con las que hemos entrenado el modelo, pinchando en ellas podemos ver los objetos localizados (en nuestro caso solo uno) y podremos usarlas de nuevo si queremos añadir nuevos objetos en el entrenamiento, nos da la opción de Add images.

**"El sistema nos proporciona unos dominios entrenados, como lo queremos para un objeto concreto personalizado elegimos la opción General"**

Una vez subida la imagen de entrenamiento el sistema detectará algunos objetos dentro de ella, como se puede ver en este caso detectó que hay algo en la zona de Rodolfo.

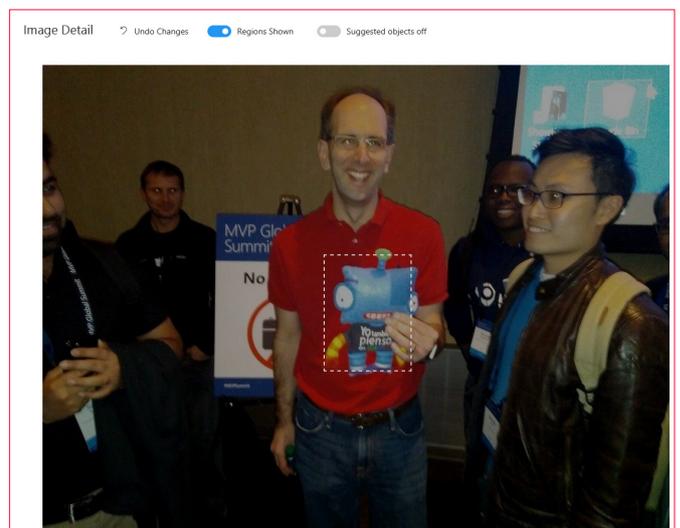


Imagen 6.- Imagen de entrenamiento.

Podemos emplear la selección que ha hecho la herramienta o hacer una nueva, en mi caso suelo hacer una nueva porque me parece importante que incluya la antena verde, creo que es un elemento común en todos los merchandising y muy característico.

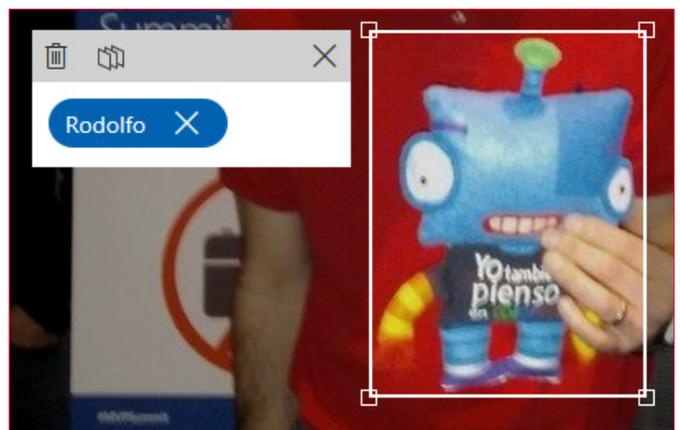


Imagen 7.- Selección realizada.

Le daremos la etiqueta que hemos creado para Rodolfo.



Un truco es que puedes usar la rueda del ratón para hacer zoom en el cuadro que has marcado para ampliarlo o reducirlo. La imagen pasa automáticamente a estar en el apartado de entrenamiento sin hacer nada más, si volvieses a subir la misma imagen te indicaría que es duplicada. En el apartado Performance entrenaremos las distintas iteraciones, podemos contar con 10 máximas tanto en el plan gratuito (F0) como en el de pago (S0). En este apartado contamos con dos variables:

- **Probability Threshold.** Se trata de la probabilidad mínima para que una predicción sea válida al calcular la precisión.
- **Overlap Threshold.** Porcentaje mínimo de solapamiento entre los cuadros delimitadores predichos y los declarados correctos para ser considerado una predicción correcta.

Podemos ir “jugando” con estas variables para obtener modelos más precisos, entiéndase que no es un tema de azar, si no que se trata de calcular cuál son los valores que necesitamos para nuestro modelo. En mi caso lo deje con los valores por defecto, Probability Threshold 50% y Overlap Threshold 30% por dos motivos: Primero, no llego a proporcionarle suficientes fotos para hacer un modelo serio (se trata de una demo) y segundo, que en mis demos suelo considerar que si se obtiene un valor superior al 50% lo considero correcto. Por lo general si lo detecta suele dar valores muy extremos y si no lo detecta valores muy similares normalmente cercanos al 40%, puede darse algún falso positivo cercano al 50-57%.

**"Custom Vision tiene dos apartados, Prediction y Training que considero bastante descriptivos"**

El último apartado en el menú es Predictions, en este caso encontraremos peticiones de predicción realizadas contra la API y los valores que ha devuelto. Este apartado es muy interesante para obtener muestras de reentrenamiento y ver con que tipo de objetos puede estar confundiendo al que nos interesa. En nuestro caso vimos que a veces confundía a Rodolfo con el reflejo de dos monitores en unas gafas y con la bandera de Canarias, imaginamos que por ser formas cuadradas azules con zonas amarillas. Hay que recordar que nuestro modelo solo esta entrenado con unas 22 fotografías y se recomienda no hacerlo con menos de 50, además de que no es siempre el mismo objeto, si no que se tratan de 5 formas distintas del mismo objeto lo que aporta más complejidad al reconocimiento. Se podría hacer que reconociese cada tipo de objeto por separado.

Una vez cargadas las fotos de entrenamiento solo tenemos que pulsar el botón Train del menú y seleccionar una de las dos opciones:

- **Quick Training,** para un entrenamiento sencillo.
- **Advanced Training,** para un entrenamiento más intensivo y completo. En esta modalidad se indica una dura-

ción máxima del entrenamiento entre 1 y 24 horas, se te facturará por el tiempo de entrenamiento.

Aquí puedes ver el resultado para un entrenamiento rápido de un caso de éxito y un fracaso.

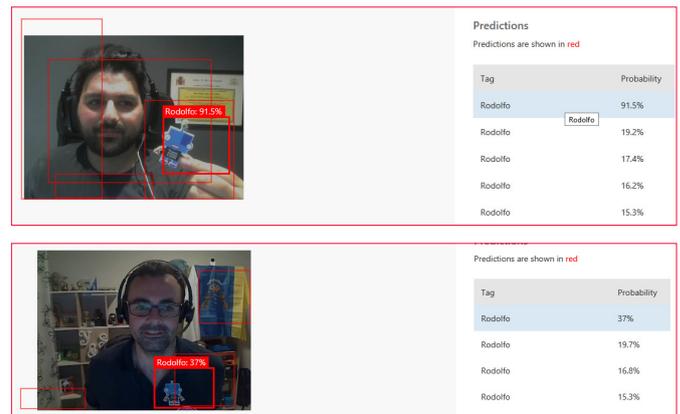


Imagen 8.- Resultado del entrenamiento.

## Código

Custom Vision proporciona un SDK para C#, Go (1.8+), Java (JDK 7 u 8), JS (NodeJs 8+) y Python (2.7+ o 3.5+), aunque hay que recordar que se trata de una API, por lo que puede ser usado por cualquier lenguaje. En este caso explicaré lo básico del SDK de C#.

Custom Vision tiene dos apartados, Prediction y Training que considero bastante descriptivos. En nuestro caso solicitamos al Custom Vision el cliente predictivo proporcionándole una ApiKey y un Endpoint. En el caso de la ApiKey la obtenemos en el apartado Performance pinchando sobre la iteración que queremos emplear (debe estar publicada) y en el botón Prediction URL nos proporciona la Key y otras particularidades de la petición. El Endpoint lo encontramos en el apartado de configuración (icono de rueda dentada a la derecha del menú).

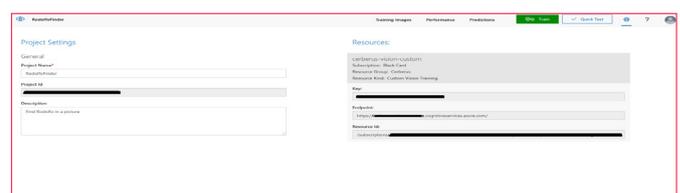


Imagen 9.- Settings del Proyecto.

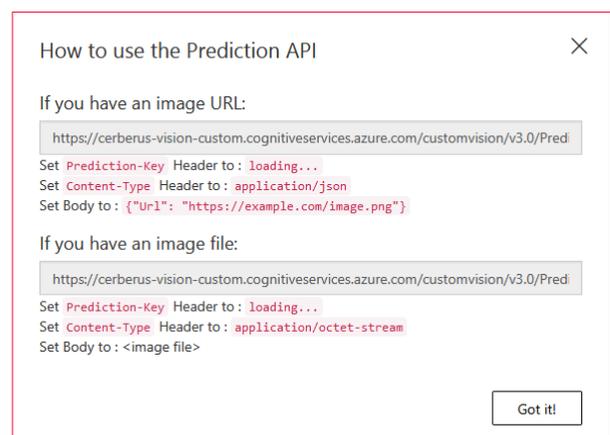


Imagen 10.- Como usar la API.

A esta API podemos enviarle una URL de una imagen o la propia imagen. Una vez contamos con el cliente predictivo llamaremos al método DetectImagen que necesita ProjectId (en el apartado de configuración), un publishedName, que se trata del nombre de publicación de la iteración (podríamos tener varias con el mismo nombre, pero solo una publicada en cada momento de forma que no fuese necesario modificar el código y se podría cambiar el modelo) y la imagen o URL. Esto nos devolvería una colección de predicciones con todo lo detectado y sus probabilidades. De esta forma el código quedaría de la siguiente forma:

```
var customVisionClient = new CustomVisionPredictionClient0
{
    ApiKey = predictionKey,
    Endpoint = endpoint
};
var result = client.ClassifyImage(projectId, "{IterationName}",
```

```
image);
foreach (var prediction in result.Predictions)
{
    if (prediction.Probability > 0.5)
    {
        return true;
    }
}
```

Bola Extra: Este cliente cuenta con otro método ClassifyImageWithNoStore, la diferencia es que no almacena la imagen dentro del apartado de Predictions y por lo tanto aporta mayor privacidad, su uso es igual que su versión con guardado.

---

**DIEGO ZAPICO**  
**Consultor Informático**  
*@dzapico*

# Project Cortex: un Microsoft 365 mucho más inteligente

El pasado mes de noviembre se presentó el que será el mayor lanzamiento de Microsoft 365 desde que, hace tres años, se lanzara Microsoft Teams. Project Cortex permitirá distribuir y mejorar el conocimiento y la experiencia de los usuarios en los servicios más populares que utilizan a diario, dando a SharePoint, a Microsoft Teams y a Office capacidades de IA que permitan generar redes de conocimiento.

## ¿Qué es Project Cortex?

Project Cortex será un nuevo servicio de Microsoft 365 que permitirá aprovechar mucho mejor el conocimiento almacenado en la plataforma y el conocimiento que tienen los usuarios de una organización gracias a sus capacidades de IA. El nuevo servicio permitirá diseñar una knowledge network distribuida a lo largo de Microsoft 365 y de todos sus servicios.

Project Cortex utiliza tres piezas clave que son fundamentales:

- Microsoft Graph: proporcionará información y contenido almacenado en Microsoft 365, así como datos centrados en las personas, su trabajo, relaciones y experiencia (Imagen 1).
- SharePoint: será el almacén del contenido, la información y documentación que Project Cortex utilizará como base del conocimiento. Además, será el motor de portales en los que se mostrará la información de las knowledge centers (Imagen 2).
- Microsoft AI: permitirá automatizar la captura, categorización y administración del contenido de Microsoft 365 gracias a herramientas que se integrarán en el servicio como AI Builder.



Imagen 1.- El Microsoft Graph nos proporciona contenido y datos de las personas.

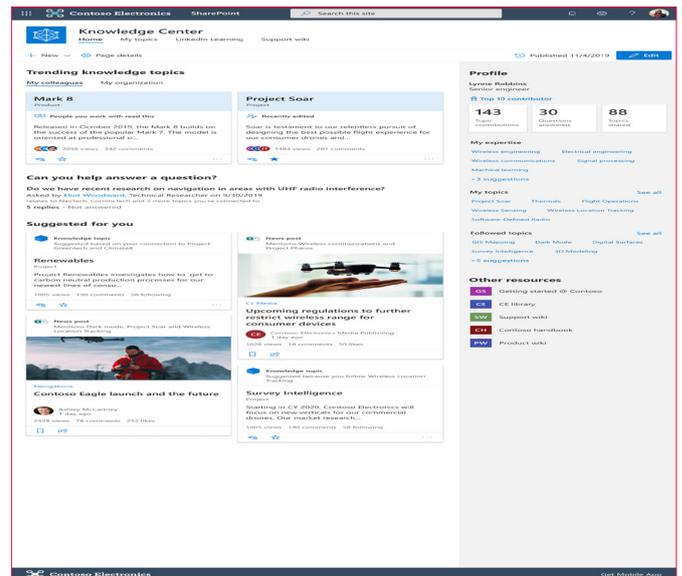


Imagen 2.- Un Knowledge Center de Project Cortex que utiliza SharePoint como motor.

## Las piezas clave de Project Cortex

Organizar el contenido de forma inteligente puede ser todo un reto. Los usuarios pueden almacenar en Microsoft 365 contenido tanto estructurado como desestructurado. Project Cortex aplica la IA para organizar automáticamente el contenido almacenado utilizando distintos mecanismos y modelos que pueden ser automáticos o configurados por los usuarios.

La unidad básica de Project Cortex es el topic (o tema). El servicio permitirá reconocer tipos de contenido, extraer información importante y organizar el contenido en topics.

Microsoft ha anunciado que, de momento, estos topics podrán consumirse desde Outlook, las aplicaciones de Office, SharePoint, Microsoft Teams y Microsoft Search (Imagen 3).

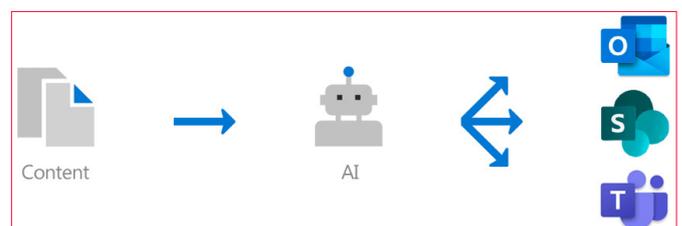


Imagen 3.- El contenido pasará por la AI de Microsoft y permitirá visualizar topics en los servicios de Microsoft 365.

Los topics aparecerán destacados en los distintos servicios de la plataforma. Cuando un usuario haga clic en ellos, podrá visualizar información inteligente generada a partir de la información que se tiene sobre el topic (Imágenes 4 y 5).

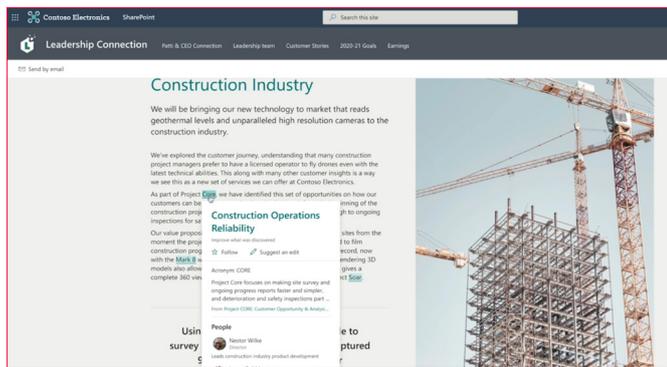


Imagen 4.- Un topic que aparece en SharePoint y permite mostrar información.



Imagen 5.- Un topic que aparece en Word y permite mostrar información.

## ¿Cómo se extrae el contenido inteligente?

El contenido que Project Cortex utiliza para crear los topics y toda la información asociada a éstos se extrae de tres formas distintas:

- Automáticamente: el servicio podrá escanear repositorios y categorizar la información de manera automática.
- Mediante modelos de AI Builder: Microsoft integrará AI Builder en la interfaz de las bibliotecas de SharePoint para que los usuarios o los administradores creen modelos con los que categorizar la información.
- Manualmente: los usuarios podrán crear sus propios topics desde algunos servicios para añadir información de forma manual (para mostrar información por ejemplo de un proyecto, de un evento, una iniciativa...).

Para entender el funcionamiento de estas clasificaciones, Microsoft ha mostrado algunos ejemplos de categorización utilizando AI Builder. Este servicio forma parte de la Power Platform y permite diseñar modelos inteligentes a partir de datos estructurados o desestructurados para localizar información de forma inteligente.

Para más información sobre AI Builder, recomiendo el artículo de Ferrán Chopo en CompartiMOSS nº 43.

Durante el anuncio oficial de Project Cortex, Microsoft mostró cómo el servicio de AI Builder quedará integrado en la interfaz de las bibliotecas de SharePoint. La ubicación

es la misma que la de otros servicios como Power Automate o Power Apps, en la cinta de opciones de la biblioteca (Imagen 6).

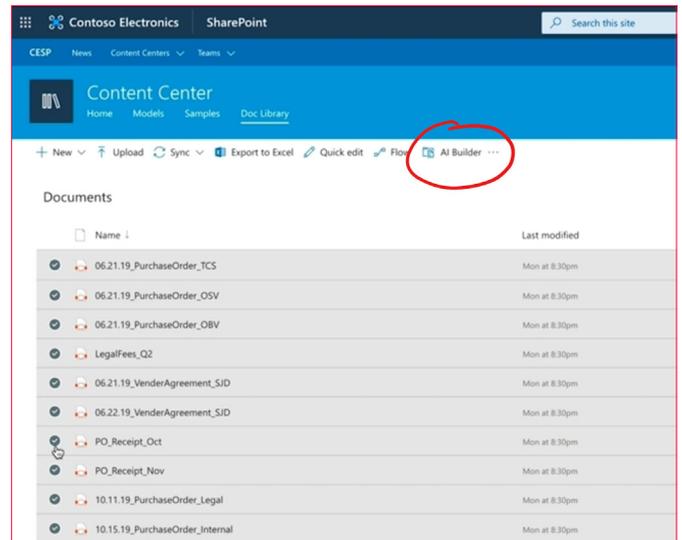


Imagen 6.- AI Builder estará integrado en las bibliotecas de SharePoint.

Utilizando AI Builder desde una biblioteca de SharePoint, podremos diseñar modelos que utilicen servicios cognitivos sobre nuestros documentos para hacer cosas como capturar contenido, extraer información y clasificar o taggear automáticamente. Esta información quedará almacenada en campos de la biblioteca. De momento sabemos que podremos contar con:

- Image and text recognition para identificar objetos o textos en documentos escaneados o en imágenes que se hayan cargado en una biblioteca.
- Forms Processing que permite diseñar un modelo a partir de documentos que sean similares, extrayendo información de campos clave para ser alimentadas en la biblioteca de documentos como metadatos.
- Machine Teaching que es muy útil para reconocer información de forma inteligente en documentos con contenido desestructurado como contratos, propuestas o recursos de formación. Estos documentos tienen variedades de formatos y distribución. Del mismo modo que se puede enseñar a una persona a rastrear contenido en documentos y etiquetarlos, ahora se puede enseñar a la IA, creando un modelo reutilizable basado en un pequeño subconjunto de documentos de muestra.

Cuando se seleccione una serie de documentos y se pulse en el botón de AI Builder, se lanzará la interfaz del servicio de AI Builder integrada en SharePoint. Durante la presentación del servicio, Microsoft mostró una demostración de Forms Processing.

Imaginemos que tenemos una biblioteca llena de facturas que siguen la misma estructura. Al tratarse de contenido estructurado, podemos suponer que el mismo tipo de información se encontrará en las mismas ubicaciones del documento. Aprovechando esto, el servicio encontrará datos de forma inteligente que se podrán mapear con campos de

la biblioteca. Esto hará que podamos popular la biblioteca con metadatos de forma automática (Imagen 7).

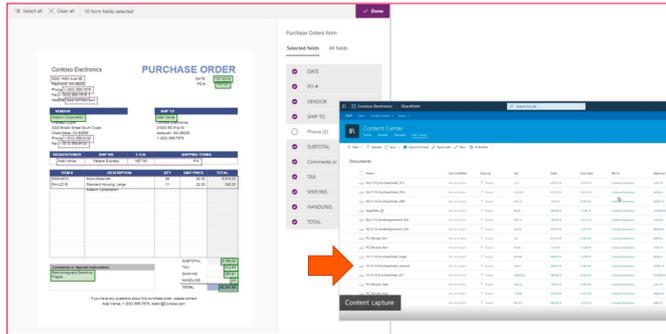


Imagen 7.- Forms Processing permite localizar información y popularla en los campos de la biblioteca.

Del mismo modo que Forms Processing es útil para contenido estructurado, podremos utilizar Machine Teaching para entrenar a la inteligencia y obtener datos en documentos con contenido desestructurado.

Estos campos o metadatos de las bibliotecas ayudarán a Project Cortex a categorizar la información y a detectar y generar nuevos topics y todo su contenido relacionado.

## Visualizando el contenido

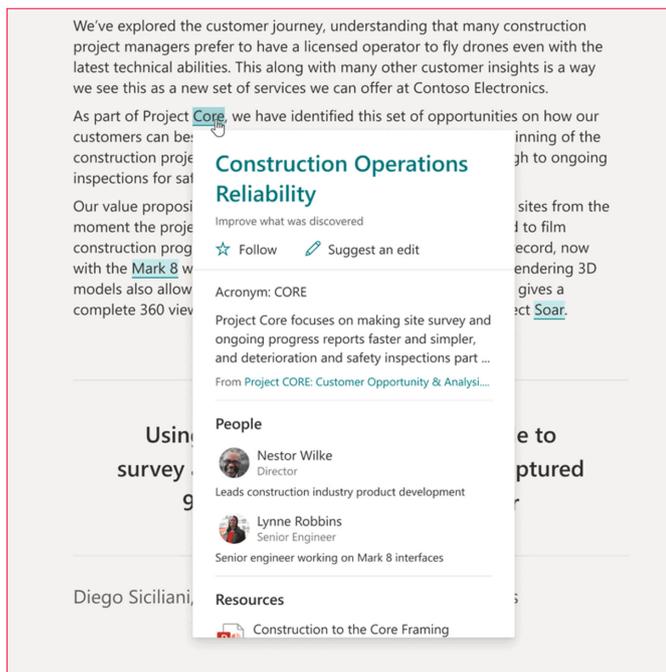


Imagen 8.- Un topic card que ha aparecido al leer una palabra en una página de SharePoint.

La información que muestra Project Cortex puede estar relacionada con las personas gracias al uso de Microsoft Graph. El segundo mecanismo de visualización consiste en una integración de topics dentro de las people cards que ya existen en muchos servicios de Microsoft 365.

Los people cards ayudarán a identificar fácilmente quién está trabajando en qué dentro de la organización. Project Cortex pedirá automáticamente a los usuarios que añadan topics basándose en el trabajo que hacen, ayudándoles a mantener un perfil actualizado (Imagen 9).

**"Project Cortex genera automáticamente topic pages a partir del contenido relacionado con un topic gracias al uso de la AI"**

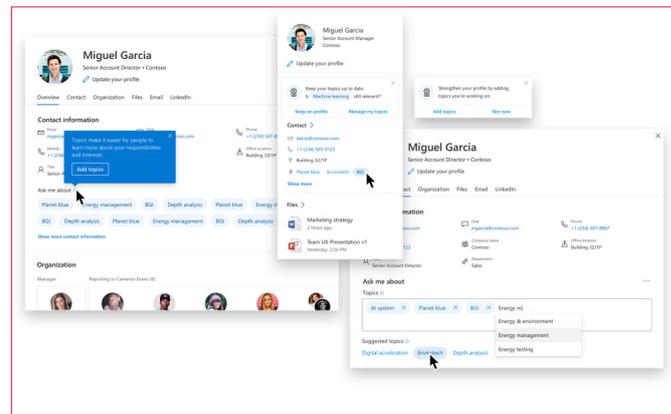


Imagen 9.- Los people card mostrarán topics relacionados con la persona.

El siguiente mecanismo de visualización son las topic pages, que son páginas de SharePoint en las que Project Cortex muestra información ampliada sobre un topic.

Project Cortex genera automáticamente topic pages a partir del contenido relacionado con un topic gracias al uso de la AI. Además de la auto generación, se permitirá a los usuarios que editen las páginas existentes para completalas y que generen nuevas páginas para añadir topics, tal y como se hace en una wiki. De esta forma, se podrá ir incorporando conocimiento desde distintas fuentes.

Las topic pages serán totalmente editables, incluyendo webparts creadas especialmente para ampliar la base de datos de conocimiento (Imagen 10).

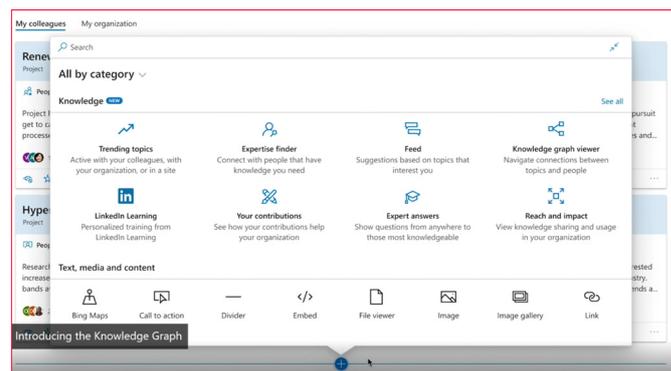


Imagen 10.- Las topic pages serán editables, con nuevas webparts que ayuden a enriquecer.

Finalmente, las topic pages se agruparán en los knowledge center, eje central de Project Cortex. Los knowledge center serán sitios de SharePoint que proporcionarán una visualización del contenido relevante para toda la organización (Imagen 11).

Se podrá crear un knowledge center principal y, adicionalmente, crear knowledge centers orientados en topics concretos. Los knowledge center se podrán integrar en el resto de los portales de SharePoint.



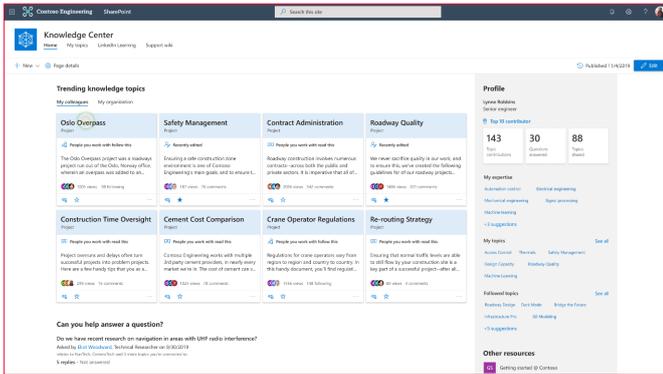


Imagen 11.- Un Knowledge Center de Project Cortex que utiliza SharePoint como motor.

## El nuevo servicio de metadatos de SharePoint Online

Para soportar todos los metadatos que se generan automática o manualmente, Microsoft ha diseñado un nuevo servicio de Metadatos Administrados (Managed Metadata Service), que proporciona metadatos a todos los servicios de Microsoft 365 y no sólo a SharePoint (Imagen 12).

**"Aunque el servicio de Project Cortex no está disponible, Microsoft quiere que nos empecemos a familiarizar con algunos conceptos"**

Este nuevo servicio soporta la creación de términos, sinónimos y traducciones multi lenguaje. Además, soporta hasta 1 millón de términos por tenant y cuenta con galerías de tipos de contenido y con métricas y analíticas que ayudarán a comprender mejor el uso que se está haciendo de los metadatos.

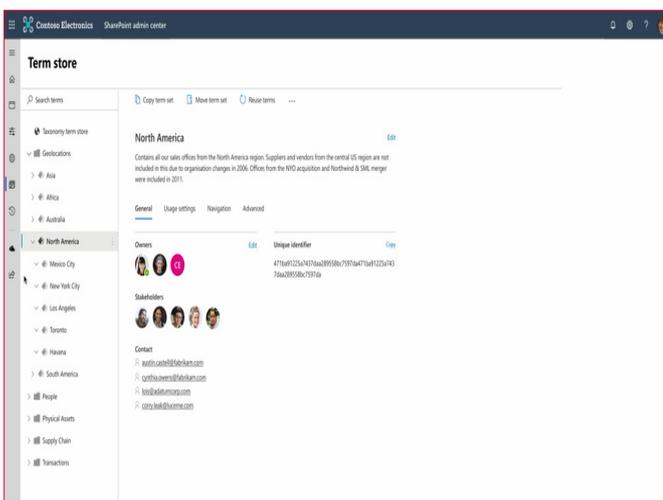


Imagen 12.- El nuevo servicio de metadatos de SharePoint Online.

El nuevo servicio de metadatos se ha empezado a desplegar en los tenants en Targeted Release desde el pasado mes

de abril. Puedes consultar más detalle en <https://techcommunity.microsoft.com/t5/microsoft-sharepoint-blog/modernizing-sharepoint-managed-metadata-services-mms/ba-p/1277450>

## Demostraciones

Aunque el servicio de Project Cortex no está disponible, Microsoft quiere que nos empecemos a familiarizar con algunos conceptos. Para ello, han publicado dos demos interactivas sobre alguna de las características que hemos tratado en este artículo. Recomendamos probarlas:

Primera experiencia con Project Cortex: <https://demobuilderwebcptxz.blob.core.windows.net/demo-v2/startdemo.html>

Organize knowledge: <https://demobuilderwebcptxz.blob.core.windows.net/automatically-connect-and-organize-knowledge/startdemo.html>

Content capture: <https://demobuilderwebcptxz.blob.core.windows.net/machine-teaching-with-project-cortex/startdemo.html>

## Referencias y material adicional

Para la elaboración de este artículo he utilizado toda la información pública que Microsoft mostró en el pasado Ignite 2019, celebrado en noviembre. Si quieres conocer más detalle, no puedes perderte estos dos enlaces del evento:

- Innovaciones en SharePoint, OneDrive y Office. Keynote de Jeff Teper en Ignite: <https://myignite.techcommunity.microsoft.com/sessions/81522>
- Introducing Project Cortex: <https://techcommunity.microsoft.com/t5/microsoft-365-blog/introducing-project-cortex/ba-p/966091>

Además del material de Ignite 2019, me han sido de mucha utilidad la web y el blog oficiales de Project Cortex, donde puedes encontrar enlaces a multitud de recursos online:

- Web oficial: <http://aka.ms/ProjectCortex>
- Project Cortex blog: <https://techcommunity.microsoft.com/t5/project-cortex-blog/bg-p/ProjectCortexBlog>

Si has llegado hasta aquí, espero que este artículo te haya sido de utilidad. Nos espera un periodo muy emocionante a todos los profesionales y amantes de Microsoft 365 y de SharePoint. La IA llega a la plataforma para quedarse.

### MIGUEL TABERA PACHECO

Modern Workplace offering lead en Avanade Iberia  
MVP de Office Apps & Services

@migueltabera

[linkedin.com/in/miguel-tabera](https://www.linkedin.com/in/miguel-tabera)

# Introducción al Cumplimiento de Comunicación en Microsoft 365

## Introducción

Pese a que siempre han existido riesgos en la gestión de la información en las organizaciones, el creciente uso de servicios en la nube y la accesibilidad desde cualquier lugar y dispositivo ha hecho que la gestión de éstos empiece a tomar cierta relevancia dentro las mismas. Insider Risk Management surge como una nueva solución en Microsoft 365 para detectar actividades involuntarias y/o intencionadas que pongan en riesgo a la organización: Violaciones de confidencialidad, fugas y robo de información o fraude.



Imagen 1.- Flujo de actuación en Insider Risk Management.

Actualmente ya existen algunas soluciones como Data Loss Prevention o eDiscovery, pero Insider Risk Management nos da funcionalidad añadida, ya que permite analizar datos almacenados en otras aplicaciones mediante conectores, usar algoritmos de Machine Learning para detectar y analizar ciertos patrones, y, en definitiva, usar plantillas preconfiguradas y adaptables a nuestras necesidades.

Como ya hemos indicado anteriormente, Communication Compliance forma parte de la solución Insider Risk Management, y puede resultarnos útil en diversos escenarios:



Imagen 2.- Casos de uso de Communication Compliance.

Así, vemos como la monitorización de las comunicaciones en ciertos entornos puede convertirse en una actividad esencial más para poder cumplir, por ejemplo, con la ley.

¿Y qué canales de comunicación podemos monitorizar? Ahora mismo los canales que se pueden examinar son los siguientes:

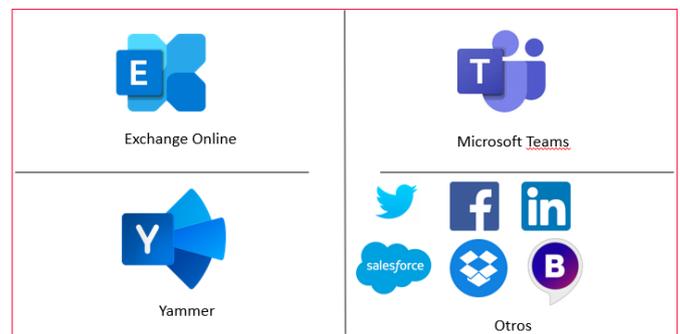


Imagen 3.- Canales de comunicación monitorizados por Communication Compliance

Cabe indicar que también se monitoriza la actividad en Skype for Business, y que, en la parte de los correos electrónicos, se tienen en cuenta aquellos que son entrantes y también salientes.

En conclusión, podemos monitorizar distintos canales de comunicación en nuestra organización para detectar lenguaje ofensivo, incumplimiento de leyes o uso fraudulento de información confidencial mediante el uso del servicio, definiendo unas políticas, reglas y flujos que veremos cómo configurar en la siguiente sección.

## Caso práctico: Uso de lenguaje ofensivo en Microsoft Teams y correo electrónico

La mejor manera de ver cómo nos puede ayudar Communication Compliance es verlo en acción en un caso práctico. Imaginemos que queremos detectar el uso de lenguaje ofensivo tanto en los chats como en las publicaciones en Microsoft Teams y en los correos electrónicos dentro de una organización.

### Prerrequisitos

Para poder utilizar la solución necesitaremos uno de estos tres tipos de licencias:

- Microsoft 365 E5.
- Microsoft 365 E3 + complemento Microsoft 365 E5 Compliance.
- Microsoft 365 E3 + complemento Microsoft 365 E5 Insider Risk Management.

Estas licencias serán necesarias tanto para los administradores de cumplimiento como para los usuarios que tengamos que monitorizar. Este último punto es realmente importante, pues significa que prácticamente todos los usuarios de la organización deberían tener asignadas una de estos tres tipos de licencias, con el coste que esto puede suponer.

**"ya existen algunas soluciones como Data Loss Prevention o eDiscovery, pero Insider Risk Management nos da funcionalidad añadida"**

Adicionalmente es necesario activar los registros de auditoría<sup>1</sup> en el tenant.

### Creación de la directiva

Antes de crear nuestra política de detección de lenguaje ofensivo deberemos asignar permisos a aquellos grupos o personas que definirán y revisarán las mismas (administradores de cumplimiento). Para ello, desde el Centro de Cumplimiento en Microsoft 365 deberemos crear un nuevo permiso, asignando los roles existentes Supervisory Review Administrator, Case Management y Compliance Administrator y asignarlo a las personas correspondientes.

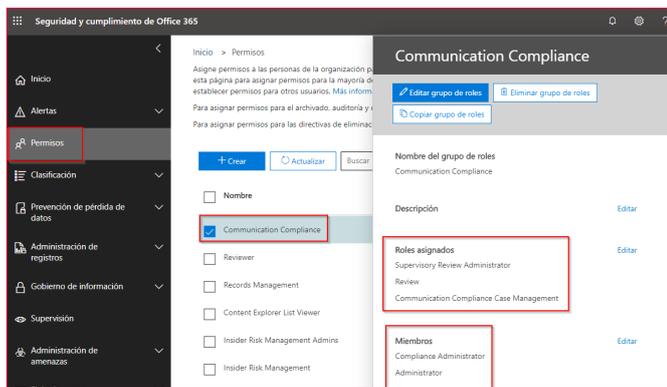


Imagen 4.- Asignación de permisos para los administradores de Cumplimiento.

Cabe mencionar que, por defecto, el administrador global del tenant no tiene permisos para crear y revisar políticas de cumplimiento de la comunicación, con lo que este paso es inevitable.

Seguidamente, desde el centro de administración de cumplimiento aparecerá una plantilla para supervisar las comunicaciones en busca de lenguaje ofensivo, donde podremos indicar qué usuarios queremos supervisar y qué usuarios deben poder revisar las comunicaciones generadas por la directiva.

1 <https://docs.microsoft.com/es-es/microsoft-365/compliance/turn-audit-log-search-on-or-off?view=o365-worldwide>

## Supervisar las comunicaciones en busca de lenguaje ofensivo

Háganos saber qué comunicaciones supervisamos y quién debe revisarlas. Crearemos una directiva de cumplimiento de la comunicación que use un modelo de aprendizaje automático para lenguaje abusivo con el fin de detectar contenido que podría considerarse una forma de acoso.

### Opciones de configuración que se rellenarán automáticamente

✓ **Nombre de la directiva**  
Offensive or threatening language

✓ **Comunicaciones para supervisar**  
Comunicaciones internas en Correo de Exchange, Chat de Teams, Conversaciones de Skype Empresarial, Yammer, Inbound, Outbound, Internal,

✓ **Condiciones**  
Revise el 100 % de las comunicaciones con lenguaje ofensivo o palabras incorrectas.

### Configuración necesaria

**Usuarios o grupos que se van a supervisar \***  
All Company × Empiece a escribir para buscar usuarios o grupos

**Revisores \***  
Elija usuarios para revisar las comunicaciones que devuelve esta Directiva.  
Administrator × Compliance Administrator × Empiece a escribir para b

Imagen 5.- Creación de la directiva mediante la plantilla predefinida.

La directiva puede tardar hasta una hora en activarse, y la captura de comunicaciones puede tardar hasta 24 horas en mostrar los primeros resultados. También es importante mencionar que, aunque hemos seleccionado una plantilla ya existente, podemos crear directivas personalizadas, donde podemos elegir qué canales queremos monitorizar, y además añadir condiciones concretas y el porcentaje del contenido que se va a revisar de forma aleatoria:

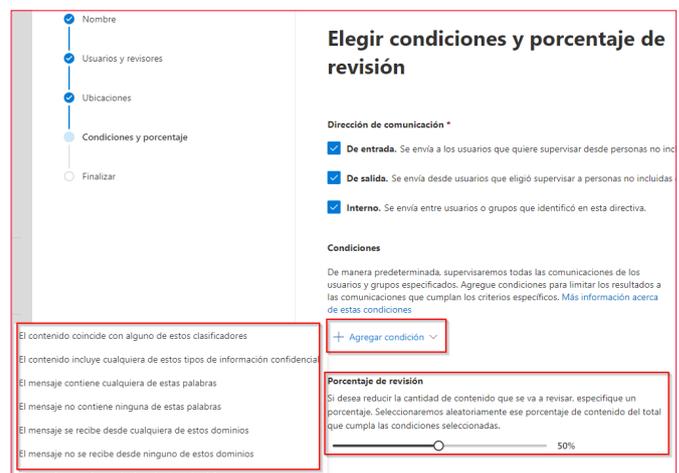


Imagen 6.- Condiciones y porcentaje de revisión en la creación de directivas.

En nuestro caso, hemos preferido crear una directiva personalizada, llamada lenguaje ofensivo, donde solo hemos elegido Microsoft Teams y el correo electrónico como canales de comunicación a examinar, así como un porcentaje de revisión del 100% de los mensajes.

### Alertas e informes

Desde el centro de cumplimiento de la comunicación nos aparecerán aquellos elementos de comunicación que pudieran coincidir con nuestras directivas, tal y como se puede ver en la siguiente figura:

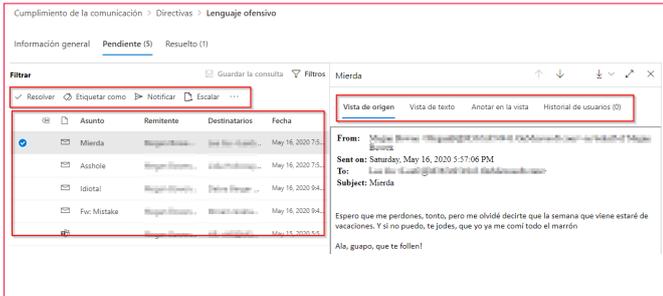


Imagen 7.- Elementos pendientes de revisión.

En la figura aparecen diversos mensajes (hemos ocultado los remitentes y destinatarios), así como el canal de comunicación (cuatro correos electrónicos y una publicación en Teams). Cabe destacar que hemos hecho pruebas en inglés y castellano, y que, en ambos casos, la detección de lenguaje ofensivo ha funcionado perfectamente bien.

Para cada uno de los mensajes disponemos de la vista original, además de poder realizar anotaciones sobre la misma (anotar en la vista) y ver el historial de usuarios que han participado en la comunicación.

**"podemos crear directivas personalizadas, donde podemos elegir qué canales queremos monitorizar, y además añadir condiciones concretas"**

A partir de aquí podremos llevar a cabo diversas acciones sobre cada uno de los elementos:

- **Notificar al usuario:** Mediante un correo electrónico se puede notificar a los usuarios que se ha revisado un mensaje escrito por los mismos y que se ha detectado el uso de lenguaje ofensivo por su parte. Estas notificaciones se envían mediante un correo electrónico que se puede parametrizar gracias al uso de plantillas de comunicación.

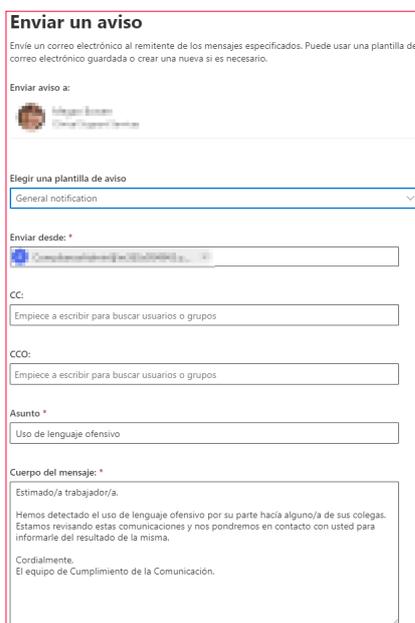


Imagen 8.- Envío de notificaciones a los usuarios.

- **Escalar:** Notificar a otras personas de la organización sobre el incidente para que sean éstas las que decidan qué acciones emprender.
- **Etiquetar:** De esta manera podemos asignar una etiqueta al incidente, que puede tomar los valores Compliant, Non-Compliant y Questionable, junto a unos comentarios. De esta manera, esa información se guarda junto al incidente para su posterior revisión histórica si fuera necesario.
- **Remitir para investigar:** Se crea un caso de eDiscovery avanzado, mediante el cual se crea un nuevo caso de investigación, analizando el contexto del mensaje junto a otras comunicaciones que se hayan sucedido en un periodo concreto de tiempo.
- **Falso positivo:** En este caso podemos indicar que el mensaje cumple con las directivas de la organización, y, en definitiva, que no es correcto que el sistema lo haya detectado como un mensaje con uso de lenguaje ofensivo.
- **Resolver:** Esta última acción sirve para resolver el caso, donde opcionalmente podremos indicar mediante comentarios las acciones emprendidas. Cuando un caso queda resuelto, siempre se puede ver el histórico de acciones de corrección.



Imagen 9.- Vista del historial de acciones de corrección.

De esta manera, podemos analizar todas y cada una de las comunicaciones que han generado una alerta, y tomar decisiones respecto a las mismas de forma muy intuitiva y ordenada. Finalmente, indicar que, aunque no disponemos de algunas automatizaciones que podrían sernos útiles (por ejemplo, publicar un mensaje en Teams), éstas se podrían realizar mediante el uso combinado del correo electrónico junto con Power Automate.

## Conclusión

En este artículo hemos introducido al lector en el uso del servicio de Communication Compliance para detectar el uso de lenguaje ofensivo en nuestra organización median-

te los distintos canales de comunicación de los que disponemos, como son principalmente el correo electrónico, el chat y las publicaciones en Microsoft Teams. Ya sea mediante el uso de plantillas de directivas definidas, o bien creándolas desde cero, hemos visto como monitorizar las comunicaciones y actuar ante situaciones graves de falta de respeto y mal uso del lenguaje.

Este es un ejemplo más sobre como Microsoft está innovando en toda la parte de gestión de la información y los riesgos asociados a ésta mediante un conjunto de servicios y soluciones que se integran muy fácilmente dentro de todo el ecosistema Microsoft 365.

¿Qué nos faltaría para que las organizaciones implanten este tipo de soluciones, independientemente de los aspectos legales ligados a la misma? Seguramente que el coste económico de estas soluciones pueda ser asumido por las organizaciones, o bien que Microsoft las vaya integrando en el futuro como un elemento más independientemente del tipo de licencia que tengan asignada los usuarios.

---

**FERRAN CHOPO GARCIA**

**Microsoft 365 Consultant & Trainer**

*ferran@ferranchopo.com*

*@fchopo*

# Productivity Tips con Microsoft Teams: Últimas novedades en reuniones, llamadas y videollamadas en Teams

Con la irrupción de la pandemia del COVID-19 en todo el mundo, durante los últimos 2-3 meses hemos visto como muchas personas de sectores diferentes de la economía han tenido que adaptarse a la situación y pasar a trabajar de forma totalmente remota. En este escenario de trabajo en remoto o teletrabajo, Microsoft Teams está jugando un papel fundamental y también el esfuerzo realizado por Microsoft para ir liberando nuevas características de productividad en la plataforma. En este artículo haremos un repaso a algunas de las últimas novedades incorporadas por Microsoft en reuniones, llamadas y videollamadas en Microsoft Teams.

## Finalizar una reunión para todos los participantes

Por increíble que parezca, esta funcionalidad no estaba disponible en reuniones de Teams de manera que bastaba con que una persona participando en la reunión no la abandonase para que la reunión no finalizase por horas y lo que es peor, si la reunión se estaba grabando, la grabación continuase hasta un máximo de 4 horas. Afortunadamente, Microsoft ha incorporado a la barra de opciones de reuniones de Teams la opción de “End meeting” que permite finalizar la llamada para todas las personas participantes.

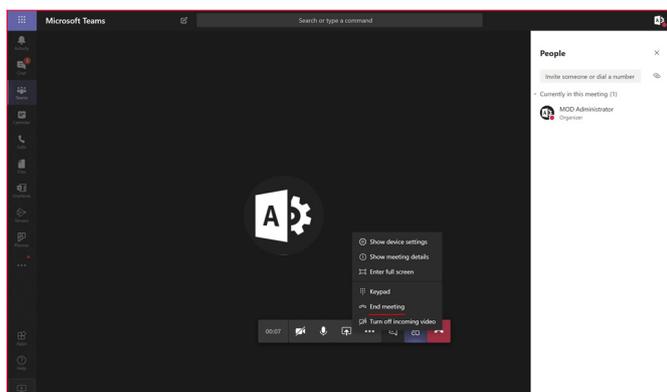


Imagen 1.- Opción “End meeting” en reuniones de Teams.

## Grabación de llamadas y video-llamadas en Chats privados 1:1

Cuando se realizaba una llamada o video-llamada en un Chat privado 1:1 no era posible realizar su grabación. Microsoft solo permitía grabar llamadas y video-llamadas de

Chats privados 1:N. Por suerte contamos desde ya con esta funcionalidad en llamadas y video-llamadas en Chats privados 1:1 de manera que la grabación será guardada en el servicio de Microsoft Stream.

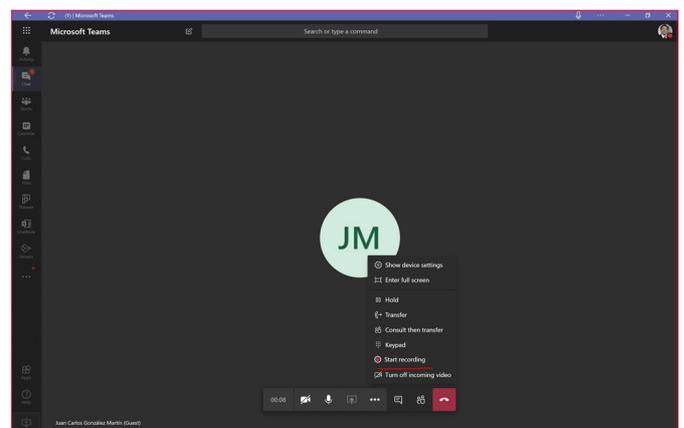


Imagen 2.- Opción de iniciar grabación en una llamada o video-llamada en un chat privado de Teams.

*"durante los últimos 2-3 meses hemos visto como muchas personas de sectores diferentes de la economía han tenido que adaptarse a la situación y pasar a trabajar de forma totalmente remota"*

## Live Captions en reuniones, llamadas y vídeo-llamadas

Sin duda, otra característica super útil en estos tiempos en los que gran parte del trabajo se realiza de forma remota. Live Captions no es una característica nueva de Teams, sino que lleva ya un tiempo en preview y lo que permite es mostrar en pantalla el texto de la persona que habla. Actualmente Live Captions solo soporta el idioma inglés y para comenzar a probarla, tenemos que habilitar esta funcionalidad en el Teams Admin Center o por medio de PowerShell. Para habilitar Live Captions en el Teams Admin Center:

- Accedemos al Teams Admin Center y a continuación en la sección de Meetings hacemos clic en Meetings policies. En mi caso lo que he hecho es configurar la directiva de reuniones global.

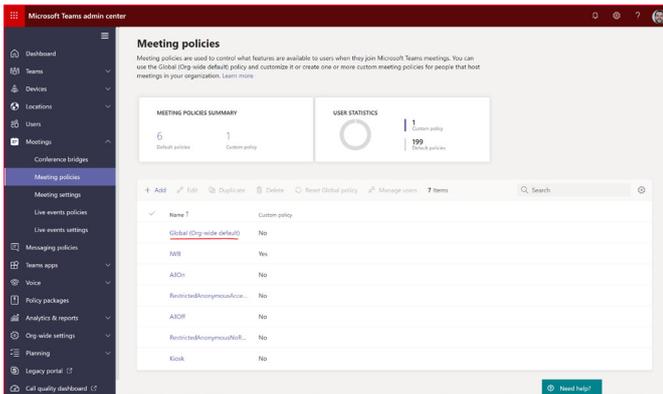


Imagen 3.- Directivas de reuniones en el Teams Admin Center.

- En la página de detalle de la directiva de reunión, localizamos la sección de Participants & guests y cambiamos el setting de Enable live captions de su valor por defecto al valor “Disabled but the organizer can override”.

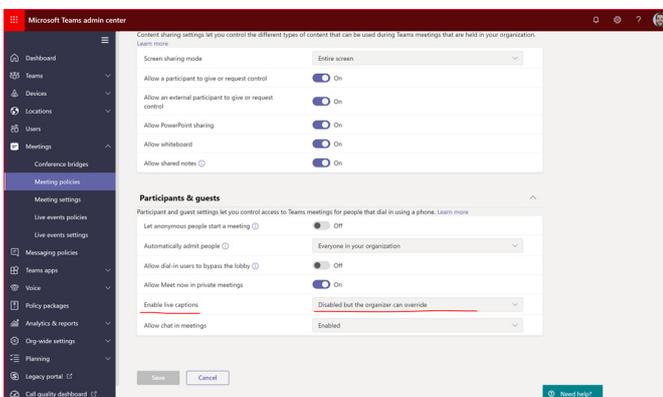


Imagen 4.- Setting a configurar en la Directiva de reuniones para habilitar los Live captions en Teams.

Una vez que hemos habilitado Live Captions en el Teams Admin Center, solo nos queda probarlo en una reunión, llamada o vídeo-llamada de Teams:

- Cuando estamos en la reunión, llamada o vídeo-llamada veremos que aparece una nueva opción “Turn on live captions (Preview)”:

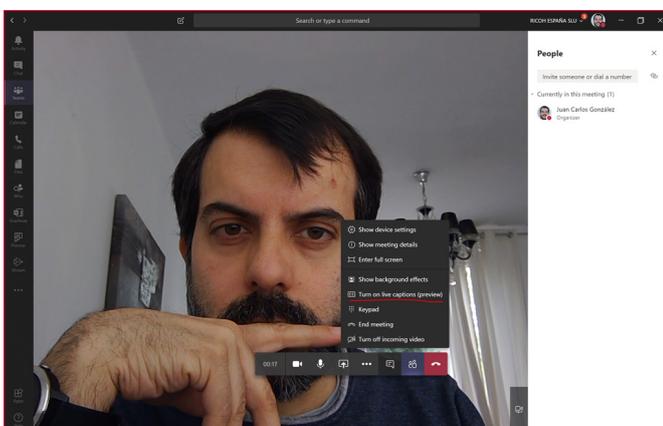


Imagen 5.- Habilitar Live captions en reuniones, llamadas y vídeo-llamadas de Teams.

- Una vez que hemos hecho clic en la opción indicada, cuando empezamos a hablar veremos los subtítulos. Obviamente, para que Teams nos entienda, tendremos que hablar en inglés.

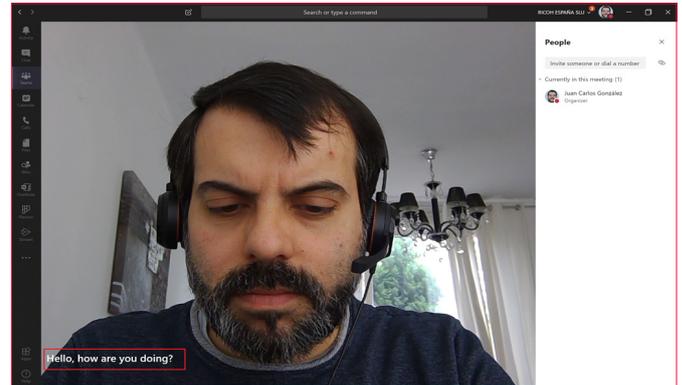


Imagen 6.- Live captions en acción en una reunión de Teams.

**"posibilidad de añadir imágenes de fondo personalizadas"**

## Fondos personalizados en reuniones y vídeo-llamadas de Teams

He dejado casi para el final del artículo una de las nuevas características en reuniones y vídeo-llamadas de Teams que más “furor” ha causado desde que fue liberada por Microsoft durante el mes de abril: la posibilidad de añadir imágenes de fondo personalizadas cuando realizamos una vídeo-llamada o participamos en una reunión en la que tenemos habilitada la cámara de nuestro dispositivo. Respecto a esta característica:

- Sólo está disponible por el momento en el cliente de escritorio de Teams para Windows 10 y en el cliente para MAC, es decir, no está soportado ni en el cliente web ni en los clientes móviles para iOS y Android. Tampoco está soportado si Teams se ejecuta en entornos VDI.
- Se tienen que cumplir ciertos pre-requisitos:
  - El equipo en el que utilizamos Teams soporta para “Background Blur” y además a nivel de hardware es necesario disponer de una CPU con soporte para AVX2 (Advanced Vector Extensions 2). Para más detalles de los requerimientos hardware de Microsoft Teams, os recomiendo este artículo de la documentación de Microsoft: <https://docs.microsoft.com/en-us/microsoftteams/hardware-requirements-for-the-teams-app>
  - El cliente de Teams esté actualizado (Nota: Si no lo está, puedes forzar la actualización desde tu perfil de usuario en Teams) a la versión 1.3.00.8863 o superior.
  - A nivel de tenant de Microsoft 365 tenemos la actualización relativa esta característica.

Si cumplimos con todos los requerimientos para disponer esta característica, estamos listos para su uso:

- En una Vídeo-Llamada o reunión de Teams con vídeo habilitado, tendremos la opción “Show background effects”.

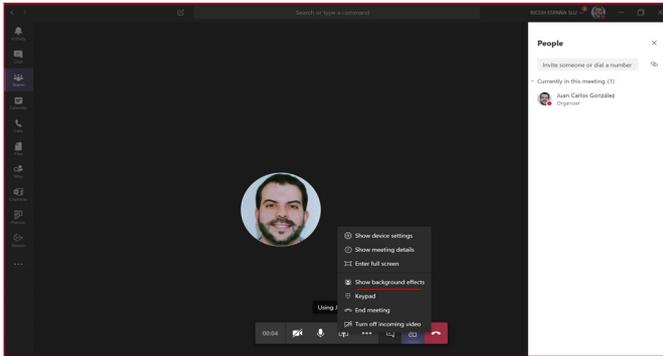


Imagen 7.- Opción "Show background effects".

- A continuación, se abre un panel en el que podremos ver la galería de imágenes pre-cargadas por Microsoft para utilizar de fondo o bien las que hayamos cargado para usar en Teams (Más adelante se explica cómo cargar nuestras propias imágenes de fondo):

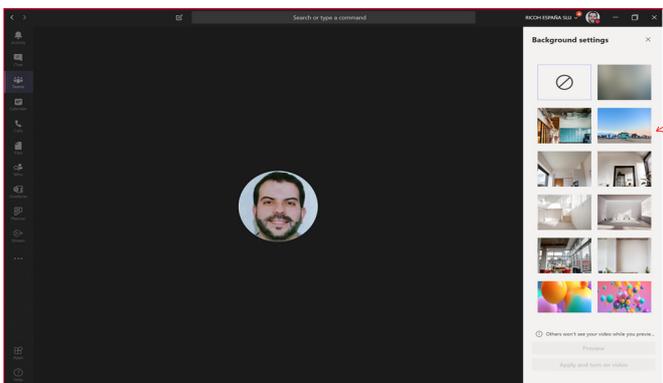


Imagen 8.- Galería de imágenes para utilizar como imagen de fondo en video-llamadas y reuniones con video en Teams.

- Seleccionamos una imagen a utilizar de fondo y hacemos clic en "Apply and turn on video" de manera que la imagen de fondo se aplique ocultando el fondo real de la habitación en la que nos encontremos.

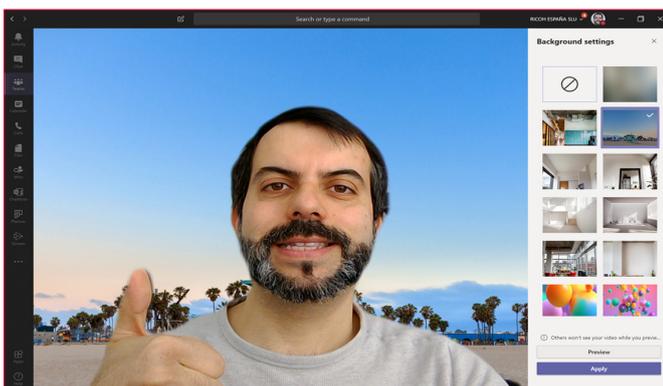


Imagen 9.- Imagen de fondo aplicada en la video-llamada o reunión con video habilitado.

Para hacer uso de nuestras propias imágenes personalizadas de fondo (Por ejemplo, si somos fans de Star wars podemos utilizar los fondos oficiales disponibles en el siguiente enlace: <https://www.starwars.com/news/star-wars-backgrounds>) sólo tenemos que cargar los mismos en las siguientes rutas:

- Equipos Windows 10: %AppData%\Microsoft\Teams\Backgrounds\Uploads
- Equipos MAC: /users/<username>/Library/Application Support/Microsoft/Teams/Backgrounds/Uploads

## Lo que está por venir (o ya tendremos disponible cuando este artículo esté publicado)

En el momento de redacción de este artículo (3 de mayo), Microsoft ha anunciado nuevas funcionalidades para reuniones de Teams que a lo largo del mes irán apareciendo en nuestro tenants. En concreto:

- Visualización de forma simultánea de hasta 9 participantes en reuniones de Teams frente a los 4 actuales:



Imagen 10.- Soporte de hasta 9 participantes con cámara habilitada en reuniones de Teams.

- Levantar la mano para solicitar el turno de palabra: la barra de opciones de reuniones de Teams incorpora la opción de solicitar el turno de palabra de forma que en el listado de participantes y también en la reunión se verán identificadas las personas que solicitan el turno de palabra.

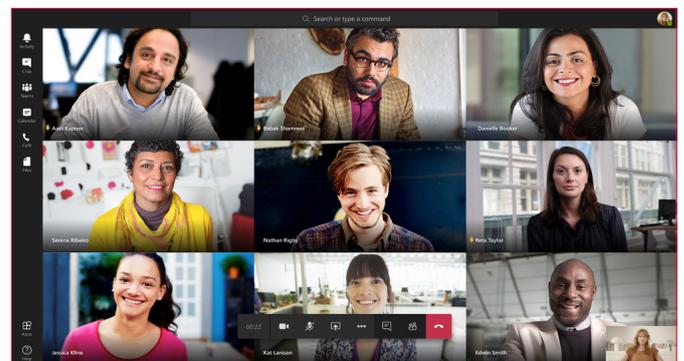


Imagen 11.- Funcionalidad de levantar la mano en reuniones de Teams.

- Difuminar la imagen de fondo cuando participamos en reuniones utilizando un dispositivo iOS.
- Soporte para Live captions en dispositivos iOS y Android.

## Conclusiones

En este artículo se ha realizado un repaso a las últimas novedades incorporadas por Microsoft a reuniones, llamadas y vídeo-llamadas en Teams que facilitan enormemente la productividad en escenarios de trabajo en remoto y teletrabajo como los que ha ocasionado la crisis del COVID-19.

**JUAN CARLOS GONZÁLEZ**

Office Apps & Services MVP | Microsoft 365 SME  
@jcg1978

# Todos los *secretos* del desarrollo de software para Microsoft Office 365



Un nuevo libro de **Gustavo Velez** que expone cómo desarrollar software para Microsoft Office 365: Exchange, SharePoint, Word, Excel, Teams, etc.

El libro evoluciona con Office 365, de manera que los lectores reciben mensualmente las **últimas actualizaciones** de Microsoft y nuevo contenido.

Encuéntrelo en

<https://guitaca.com>

 **guitaca**  
publishers



23

## Entrevista Olga Martí

Mi nombre es Olga Martí, tengo 29 años y soy de Figueras, Girona. En este último año me he mudado a Seattle, Estados Unidos, aunque siempre voy a llevar conmigo los 7 años que estuve en Madrid participando en la comunidad de SharePoint y Office 365.

Empecé a estudiar la carrera de informática de gestión en la Universidad de Girona, aunque luego me cambie al primer programa piloto de FP Dual (estudiar + trabajar) donde conseguí una beca de dos años en Microsoft. Fue después de mi beca, que empecé en el mundo de Office 365 como Ingeniera de soporte de SharePoint Online, donde conocí el producto gracias a todas las dudas, preguntas y algunos problemas que los clientes se encontraban al utilizar el servicio.



Actualmente estoy trabajando en Plain Concepts, donde he consolidado mi conocimiento técnico de desarrollo web, Microsoft Azure y servicios de Office 365. Desde hace dos años he sido galardonada con el premio Microsoft MVP en Office Development.

### ¿Por qué y cómo empezaste en el mundo de la tecnología?

Desde que entro el primero ordenador en casa, siempre me fascino el mundo de la informática y a medida que pasaban los años tenía claro que era a lo que me quería dedicar. Mientras otros niños de mi clase hacían como actividad extraescolares karate, ballet, o piano, yo empecé con cursos de mecanografía, y ofimática, hasta que finalmente con 12 años me apunté a mi primer curso de desarrollo web.

A partir de entonces, siempre era la amiga que arreglaba los ordenadores, la que siempre ayudaba a maquetar los trabajos, y que en su tiempo libre hacía páginas web de sus grupos de música favoritos, en aquella época Operación Triunfo.

### ¿Cuáles son tus principales actividades tecnológicas hoy en día?

En el último año he cambiado mi rumbo profesional a uno más dedicado al a gestión de proyectos, donde me encargo de liderar proyectos de diversas ramas desde las primeras fases de preventa hasta la entrega final al cliente. Los

proyectos que gestiono van desde desarrollo de páginas de marketing a medida, a aplicaciones de negocio integrando las últimas tecnologías en Azure y AI hasta aplicaciones Xamarin.

Desde la oficina de Estados Unidos, también me encargo de coordinar y dar soporte a demos para los principales eventos de Microsoft, lo cual me ha permitido conocer el backstage de eventos como el Microsoft Build e Ignite.

Finalmente, aunque últimamente no tengo mucho tiempo, intento seguir participando en la comunidad, grabando videos en mi canal de YouTube, escribiendo en mi blog o dando alguna charla / webcast, sobre todo enfocándome a traducir las características más técnicas de los productos a usuarios finales.

### ¿Cuáles son tus principales actividades NO tecnológicas hoy en día?

Principalmente estoy centrada en mi carrera, y crecer profesionalmente, ahora mismo en mi rol más de gestión de equipos y personas. Dejando un lado el trabajo, cada vez que tengo un hueco, lo dedico a viajar, para descubrir nuevos rincones del estado de Washington, aunque debido a las restricciones de los últimos meses, mi día a día está



siendo jugar a la Nintendo Switch a Animal Crossing. También dedico tiempo a la jardinería, y cuidar las plantas y pequeño huerto urbano que tengo en mi balcón.

## ¿Cuáles son tus hobbies?

Dejando a un lado todo el tema tecnológico, tengo gran pasión por la música. Me encanta descubrir nuevos artistas, informarme sobre las últimas novedades en lanzamientos musicales, ir a conciertos y dar algunos pinitos tocando yo misma la guitarra, o más recientemente el Ukelele. Otra de mis pasiones es la lectura, normalmente de novelas de ciencia ficción y distopías. Y por último, ya adaptándome a la cultura Washingtoniana, me estoy aficionando a hacer rutas por la montaña, ya que en la zona en la que vivo es imposible no enamorarse de la naturaleza.

## ¿Cuál es tu visión de futuro en la tecnología de acá a los próximos años?

En los próximos años preveo que va a haber dos corrientes nuevas en lo que conlleva a tecnología. La primera es como como cada vez más los empleados van a desarrollar sus propias aplicaciones, gracias a herramientas low-code o no-code como Power Apps, Power Automate, Virtual

Agents y los departamentos técnicos se van a especializar en aplicaciones de negocio avanzadas. Si se sigue apostando en la educación obligatoria explicar las bases de la programación, y las empresas proporcionan las herramientas adecuadas, cada vez veremos más gente que podrá resolver flujos de trabajo y pequeñas soluciones empresariales sin depender de los departamentos de desarrollo.

La segunda corriente es relacionada con la Inteligencia Artificial y de cómo se van a integrar cada vez más en aplicaciones de negocio, por sencillas o simples que sean, para optimizar procesos e impulsar la transformación digital en las empresas. Detección de objetos, lenguaje y traducción simultánea, extracción de términos clave en textos,... son algunos ejemplos de modelos ya entrenados y cada vez más refinados por grandes tecnológicas como Microsoft y que nos ofrecen como servicio a través de una API sencilla para que podemos consumir, sin tener que ser un experto en la materia, para poder integrarlo en nuestros desarrollos.

---

**OLGA MARTI**

**Delivery Lead, Plain Concepts**

**Microsoft Office Development MVP**

*@olgamarti*

*www.olgamarti.com*

# Introducción al almacén de conocimiento (Knowledge Store)

En este mundo hiperconectado el almacenar datos es de suma importancia para las empresas, escuelas, instituciones, etc., ya que con ello se realizan investigaciones o toma de decisiones, por ello cada día se busca la manera de encontrar recursos en los cuales la información quede almacenada y segura.

El objetivo de este artículo es hablar del servicio cognitivo de búsqueda “Azure Cognitive Search”, específicamente del almacén de conocimiento, una característica que permite almacenar la salida de una indexación + el enriquecimiento de los documentos hecho con servicios cognitivos de Azure para realizar análisis independientes de los datos.

Para ello es importante comenzar mencionando que es un documento enriquecido. Se le llama así a la salida de un proceso de indexación creado a partir del contenido que se ha extraído de una fuente de datos, estructurado y analizado mediante procesos de inteligencia artificial, pero estos documentos enriquecidos son transitorios, solo se usan durante la indexación y para aplicaciones que necesitan la característica de búsqueda, pero después se descartan, sin embargo, con el almacén de conocimientos estos documentos enriquecidos se conservan para su posterior análisis.

Por otro lado, si ha usado habilidades cognitivas (skills) anteriormente, ya sabe que los conjuntos de habilidades (skillset) se usan para mover un documento extraído de una fuente de datos por una secuencia de enriquecimientos, el resultado puede ser un índice de búsqueda enriquecido o proyecciones en un almacén de conocimientos. Estas dos salidas, son productos del mismo enriquecimiento con habilidades. Aunque se obtienen de las mismas entradas, se genera una salida estructurada y almacenada, que se emplea de maneras muy distintas; la primera salida es el índice que puede ser utilizado en aplicaciones de búsqueda utilizando sintaxis de búsqueda estándar, la segunda salida son las proyecciones que pueden ser utilizadas desde Power BI o utilizar aprendizaje de máquina para analizar los datos y realizar predicciones sobre estos datos.

En consecuencia, un almacén de conocimiento es un concepto lógico para describir datos enriquecidos para su posterior análisis, pero físicamente un almacén de conocimiento no es más que una cuenta de Almacenamiento de Azure (Storage Account), esta puede ser almacenamiento

tipo tabla, almacenamiento tipo blob o ambos, la ventaja de esto es que cualquier herramienta que pueda conectarse a estos servicios de Azure puede consumir los contenidos de un almacén de conocimiento y construir aplicaciones o funcionalidades antes no posibles.

El proceso lógico en el cual se incluye un almacén de conocimiento es:

- 1.- Se obtienen documentos no estructurados o datos estructurados de una fuente de datos las cuales pueden ser Azure SQL, Cosmos DB, Archivos Blob, etc.
- 2.- El servicio de búsqueda de Azure descifra los documentos.
- 3.- Con servicios cognitivos de Azure se realiza enriquecimiento a los documentos descifrados.
- 4.- Se agregan estos documentos a un índice para posterior consumo por aplicaciones de búsqueda.
- 5.- Al mismo tiempo estos documentos enriquecidos se agregan como proyecciones a un almacén de conocimientos.
- 6.- Por último, se pueden consumir estos datos desde Power BI, procesos de aprendizaje de máquina, etc.

Este proceso lógico puede observarse de mejor manera en la Figura 1, en donde se ve el flujo de información a través de los componentes ya mencionados:

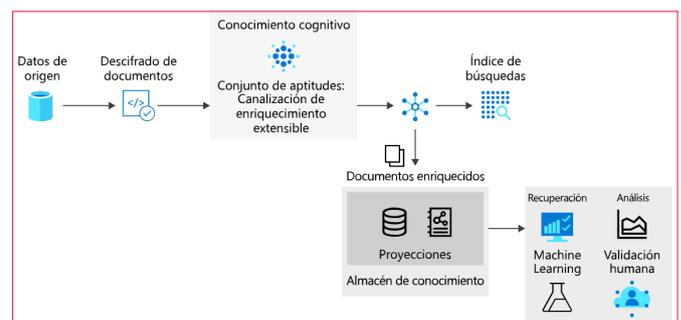


Imagen 1.- Proceso Lógico.

**Referencia:** <https://docs.microsoft.com/es-es/azure/search/knowledge-store-concept-intro>

## Ventajas de un almacén de datos

De acuerdo con la información expedida por Microsoft, un almacén de conocimiento les da estructura, contexto y contenido real a los documentos indexados obtenidos de archivos de datos no estructurados y semiestructurados como blobs, archivos de imágenes (que han sido objeto de análisis), o incluso datos estructurados, remodelados en nuevas formas; el poder ver paso a paso los datos como se van enriqueciendo es una ventaja del almacén de conocimiento; se puede ver de primera mano cómo un documento JSON complejo se puede dividir en subestructuras, se reconstruye en nuevas formas, y se hace disponible para procesos posteriores como el aprendizaje automático y procesos relacionados con las ciencias de los datos.

***"un almacén de conocimiento les da estructura, contexto y contenido real a los documentos indexados obtenidos de archivos de datos no estructurados y semiestructurados"***

Es útil ver lo que puede producir un enriquecimiento de IA, el potencial real de un almacén de conocimiento es la capacidad de remodelar los datos y se puede comenzar con un conjunto de habilidades básicas (skillsets), para así, iterar sobre él y añadir niveles adicionales de estructura e información, que luego se puede combinar en nuevas formas, consumibles en otras aplicaciones además de la búsqueda cognitiva de Azure.

Entre los beneficios del almacén de conocimientos, se incluye:

- Consumir documentos enriquecidos en herramientas de análisis e información distintas de la búsqueda. Power BI con Power Query es una opción convincente, pero cualquier herramienta o aplicación que pueda conectarse a Azure Storage puede extraer de un almacén de conocimientos.
- Refinar una indexación de IA al depurar pasos y definiciones de conjuntos de habilidades (skillsets), un almacén de conocimiento le muestra el producto de una definición de conjunto de habilidades en una indexación de IA. Se pueden utilizar esos resultados para diseñar un mejor conjunto de habilidades, ya que se puede ver exactamente los resultados de los enriquecimientos utilizando el explorador de almacenamiento de Azure para ver el contenido de los enriquecimientos durante su proceso.
- Dar forma a los datos en nuevas maneras antes imposibles, porque la remodelación de los datos está codificada en conjuntos de habilidades, Microsoft ha proporcionado una nueva habilidad llamada "Shaper" la cual permite la remodelación de los datos, preservando al mismo tiempo las relaciones, esta habilidad permite por ejemplo obtener datos de varias fuentes y luego juntarlos, transformarlos, etc.; para construir un

solo documento json enriquecido con información de ambas fuentes y al mismo tiempo como se dijo anteriormente preservando sus relaciones.

## Almacenamiento Físico

El almacenamiento físico de un almacén de conocimiento se articula a través del elemento de proyecciones en un conjunto de habilidades, esta proyección define una estructura de salida para que coincida con el uso previsto, las proyecciones pueden articularse como tablas, objetos o archivos, un conjunto de habilidades puede tener una o más proyecciones y en cada proyección la salida del enriquecimiento por IA se puede guardar en tablas de Azure o en almacenamiento tipo Blob. La definición de un almacén de conocimiento tiene la siguiente estructura JSON (figura 2):

```
JSON
{
  "knowledgeStore": {
    "storageConnectionString": "<YOUR-AZURE-STORAGE-ACCOUNT-CONNECTION-STRING>",
    "projections": [
      {
        "tables": [ ],
        "objects": [ ],
        "files": [ ]
      },
      {
        "tables": [ ],
        "objects": [ ],
        "files": [ ]
      }
    ]
  }
}
```

Imagen 2.- Estructura JSON.

El tipo de proyección que especifique el usuario en esta estructura determina el tipo de almacenamiento utilizado por el almacén de conocimiento.

Se define una proyección de tabla cuando necesite estructuras de informes tabulares para entradas a herramientas analíticas, o se exporte como marcos de datos a otros almacenes de datos. Puede especificar varias tablas para obtener un subconjunto o una sección transversal de documentos enriquecidos, dentro del mismo grupo de proyección, las relaciones de tabla se conservan para que pueda trabajar con todas ellas. El almacenamiento de blobs se usa cuando define objetos o archivos, la representación física de un objeto es una estructura JSON jerárquica que representa un documento enriquecido. Un archivo es una imagen extraída de un documento, transferida intacta al almacenamiento Blob.

Un solo objeto de proyección contiene un conjunto de tablas, objetos, archivos y, para muchos escenarios, crear una proyección podría ser suficiente. Sin embargo, es posible crear múltiples conjuntos de proyecciones de tabla-objeto-archivo, y puede hacerlo si desea diferentes relaciones de datos, todo esto dependerá de los requerimientos de la solución a implementar.

## Cómo crear un almacén de datos con el portal

Después de la descripción de los conceptos y las ventajas, y junto con los enriquecimientos de IA, se muestra un ejemplo de cómo crear y consumir un almacén de datos, para este ejemplo, previamente se ha creado una cuenta de almacenamiento Blob y he subido algunas imágenes, archivos de texto, archivos de office y PDFs.

### 1.- Clic en Importar Datos.

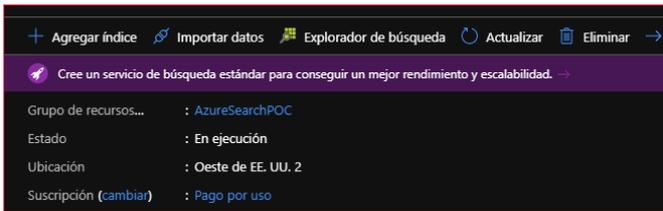


Imagen 3.- Importar datos.

### 2.- Seleccionar Almacenamiento Blob y buscar la cuenta de almacenamiento creada anteriormente.

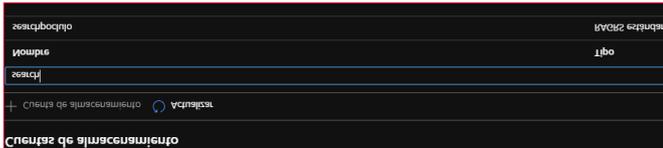


Imagen 4.- Buscar cuenta de almacenamiento.

### 3.- Seleccionar el contenedor de la cuenta de almacenamiento.



Imagen 5.- Contenedor cuenta de almacenamiento.

### 4.- Dar un nombre al origen de datos.



Imagen 6.- Nombrar origen de datos.

### 5.- Clic en siguiente para agregar habilidades cognitivas, en este paso hay 3 etapas:

- La primera etapa es adjuntar un servicio cognitivo de Azure, en cuentas gratis de Azure, se puede adjuntar una versión limitada de este servicio, en las cuentas gratis pueden realizar hasta 20 transacciones de enriquecimiento, lo cual es suficiente para nuestro ejemplo. Pero si esto no es suficiente

en esta misma pantalla se puede crear un recurso nuevo de servicios cognitivos sin límites de transacciones.

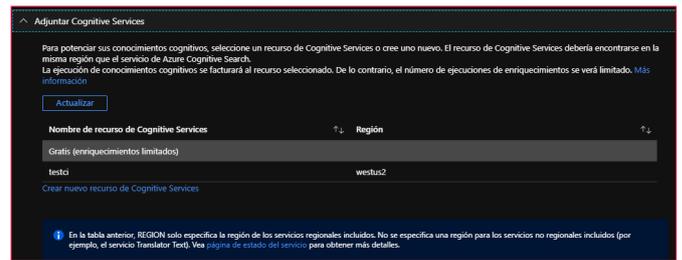


Imagen 7.- Adjuntar servicio cognitivo de Azure.

- En la segunda etapa debemos agregar los enriquecimientos:
  - Le ponemos un nombre a nuestro conjunto de habilidades.
  - Habilitamos OCR.
  - Le ponemos un nombre al campo de datos de origen, por defecto es "merged\_content".

Lo que pasara durante el enriquecimiento es que los servicios de búsqueda de Azure se comunicarán con los servicios cognitivos de visión de Azure para detectar el texto en imágenes y PDFs, este texto será guardado en el campo merged\_content para su posterior análisis por otras habilidades de IA.

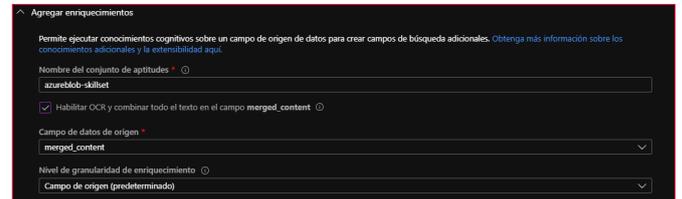


Imagen 8.- Agregando enriquecimientos.

- La tercera etapa se seleccionan los conocimientos cognitivos de texto que sean necesarios, en este paso y después de haber realizado el proceso de OCR, los servicios cognitivos de búsqueda de Azure se conectarán a los servicios cognitivos de texto y este nos permitirá enriquecer el índice con datos como nombres de personas, de organizaciones, ubicaciones, detectar el idioma, entre otros.



Imagen 9.- Selección de conocimientos cognitivos de texto.

Después seleccionamos los enriquecimientos de imagen deseados, en estos servicios cognitivos de búsqueda de Azure se conectará de nuevo a los servicios cognitivos de Visión para extraer etiquetas de imágenes, leyendas e identificar personas famosas.



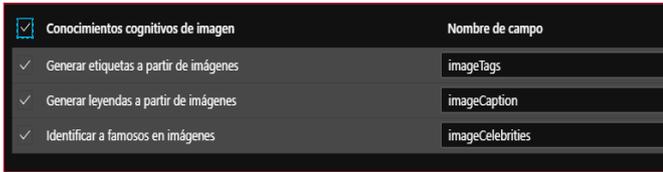


Imagen 10.- Selección de conocimientos cognitivos de imagen.

Después de configurar el enriquecimiento se cuenta con los elementos para terminar de configurar el almacén de conocimiento, que no es más que una simple cuenta de almacenamiento con todos los datos enriquecidos de los pasos anteriores.

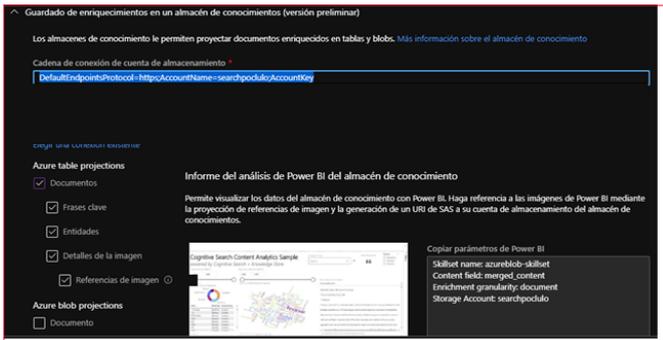


Imagen 11.- Plantilla Power BI.

En la imagen se puede observar que se puede descargar una plantilla de Power BI y obtener los parámetros de conexión, estos nos servirán en un paso posterior del proceso.

**"los servicios de búsqueda de Azure se comunicarán con los servicios cognitivos de visión de Azure para detectar el texto en imágenes y PDFs"**

6.- El siguiente paso es personalizar el índice, en este paso se cambian nombres de campos, el nombre del índice, y otras configuraciones adicionales, como por ejemplo si el campo se puede ordenar, se puede utilizar en búsquedas, etc.

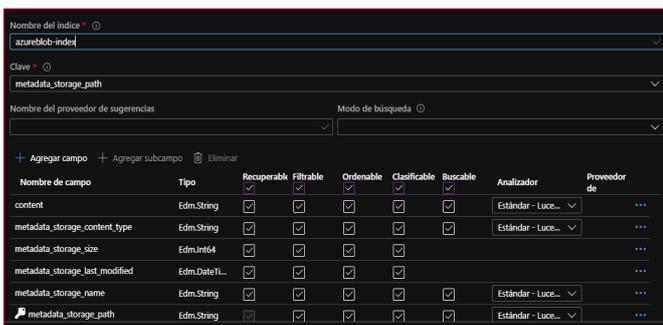


Imagen 12.- Personalización del índice.

7.- Finalmente se configura la frecuencia de la indicación y se termina el proceso dando clic en enviar.

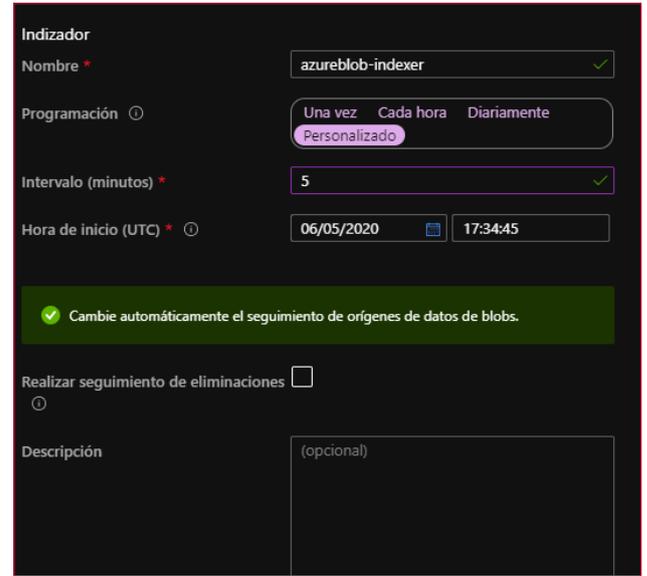


Imagen 13.- Configuración de la frecuencia de indicación.

Una vez se ha realizado este proceso, esperamos que se termine la indicación revisando el tablero en la sección de indicadores

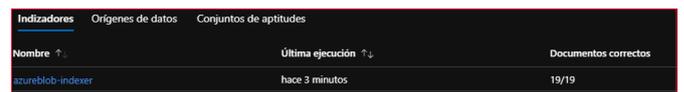


Imagen 14.- Estado de indicación.

Una de las partes fuertes de este proceso es verificar que nuestros datos enriquecidos puedan ser consultados desde otras herramientas, para esto solo debemos ir a nuestra cuenta de almacenamiento y seleccionar tablas, como pueden ver hay nuevas tablas en el almacenamiento.

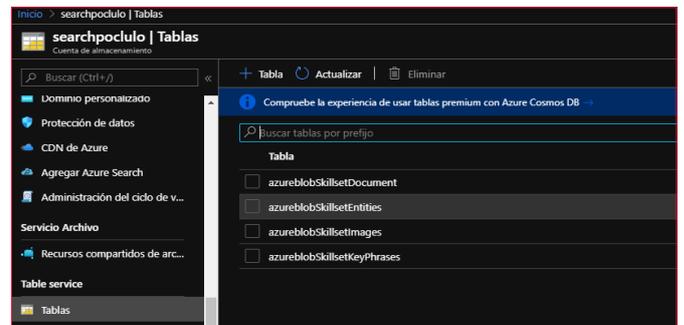


Imagen 15.- Selección de Tablas para consultar.

Si se quiere ver el contenido de las tablas, se puede usar el explorador de almacenamiento de Azure para tablas.

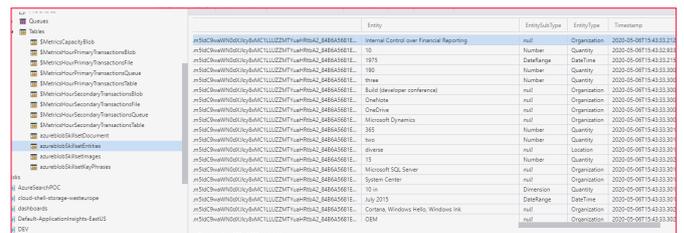


Imagen 16.- Explorador de Tablas.

Y para hacer este ejercicio completo, podemos abrir la plantilla de Power BI descargada en pasos anteriores y llenar los datos correspondientes:



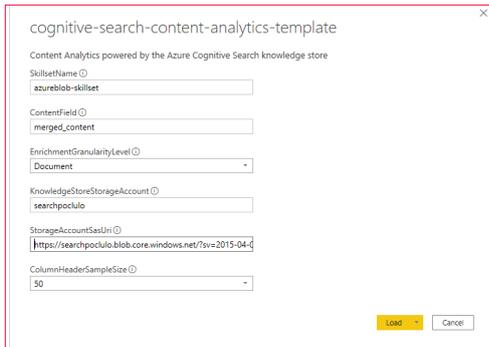


Imagen 17.- Configuración plantilla Power BI.

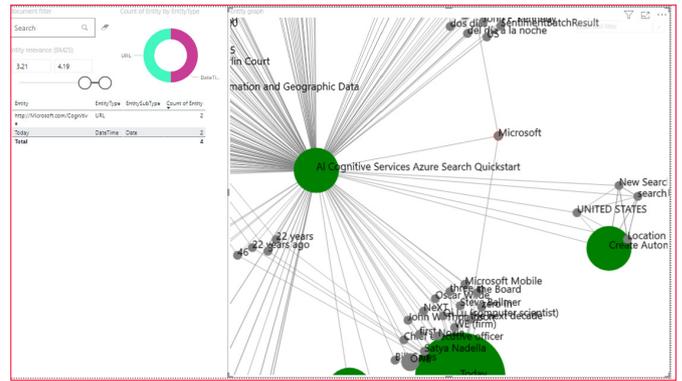


Imagen 19.- Reporte Power BI generado.

Luego cargamos los datos:

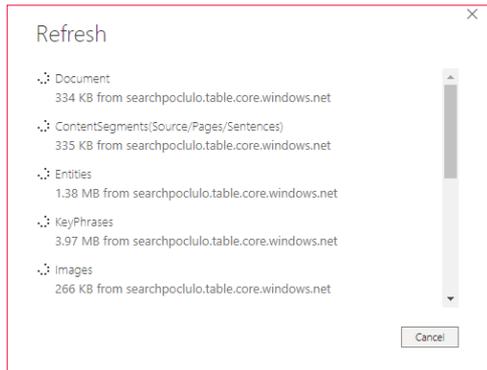


Imagen 18.- Proceso de Carga de Datos.

Y finalmente podemos analizar nuestros datos con el reporte de Power BI, la plantilla de Power BI viene con 3 pestañas diferentes para análisis de los datos, se deja como ejercicio al lector explorar las posibilidades de esta plantilla.

## Conclusiones

En este artículo hemos visto la definición de un almacén de datos, su representación física en cuentas de almacenamiento y sus beneficios, a su vez importamos datos de una cuenta de almacenamiento que contiene imágenes en un nuevo índice y adjuntamos enriquecimientos de IA para visión y análisis de texto. Después guardamos estos datos enriquecidos en un almacén de conocimiento, para finalizar y por simplicidad utilizamos la plantilla de Power BI descargada del portal de Azure, con esta plantilla nos conectamos al almacenamiento de tabla con la información de conexión entregada por el portal para luego poder desplegar los reportes en Power BI y analizar visualmente los datos enriquecidos con inteligencia artificial.

**LUIS VALENCIA**  
Office Development MVP



30

# Graph Toolkit en profundidad

## Introducción

En el número 41 Rubén Ramos nos explicó que era Graph Toolkit y como empezar a utilizarlo y todos los componentes que lo forman en aquel momento que estaba en su versión 1.0. Ahora bien, una cosa es ver la herramienta y que posibilidades tiene y otra muy distinta es como la podemos incorporar en nuestros desarrollos, y este es el punto de partida de este artículo.

Antes de empezar a usar Graph Toolkit lo primero que debemos de tener claro es cual es el sistema que se va a utilizar en nuestra aplicación para la autenticación. Sin el tema de la autenticación Graph Toolkit no tiene sentido utilizar ninguno de los componentes que existen en el mismo. La gracia de hacer uso de Graph Toolkit por un lado es que tenemos un componente con un diseño que sigue todos los patrones de cualquier aplicación desarrollada por Microsoft y la segunda y para mi importante es que nos abstraer de las peticiones que se tiene que realizar a Graph y muestra la información que se requiere.

Está claro que para el componente que nosotros queramos se debe de crear una aplicación en el Azure Active Directory que tenga los permisos necesarios para poder hacer estas consultas. Pero claro ahora bien el tema de esta autenticación contra el AAD dependerá del contexto donde se esté ejecutando nuestra aplicación, dependiendo de dicho contexto se tendrán que realizar algunas acciones u otras.

**"lo primero que debemos de tener claro es cual es el sistema que se va a utilizar en nuestra aplicación para la autenticación"**

Este contexto del que estoy hablando dentro de Graph Toolkit es el Proveedor de Autenticación, como se citó en el artículo anterior disponemos de diversos proveedores: MSAL, Simple, Teams o SharePoint. Imaginar que tenemos una aplicación personalizada implementada y que tiene ya una autenticación utilizando MSAL. ¿Qué tipo de Proveedor utilizarías para este caso? Es obvio que la pregunta tiene trampa y la respuesta no es la trivial. Se utilizaría un SimpleProvider y el motivo es que cuando tu utilizas el pro-

veedor que hay para MSAL dentro de un desarrollo previo no va a obtener la misma instancia de MSAL por lo que a efectos de Graph Toolkit ninguno de los componentes que se estén utilizando tiene un token de autenticación y está esperando que el usuario se autentique en tu aplicación. Por lo que para que funcione con un MSAL Provider se debe de autenticar además de en tu aplicación en Graph Toolkit por lo que la experiencia de usuario no es tan optima como estábamos esperando. Ahora veremos cómo se soluciona dicho inconveniente, pero primero separemos los escenarios que disponemos y como debemos de afrontarlos:

- Escenario desarrollo personalizado custom, por ejemplo, alojado en una App en NodeJS o .NET Core.
- SharePoint Framework (SPFx).
- Microsoft Teams.

**"Cuando creamos un SimpleProvider tenemos tres posibilidades pasarle el método de obtener el Token, el login y el logout"**

## Escenario Desarrollo Custom

Escenario: Disponemos de una Aplicación React en la que tenemos implementada una Autenticación usando MSAL. En nuestro caso lo que hacemos es implementar un High Order Component en React que es el que se encarga de autenticar a nuestra aplicación. Para ello tenemos este código que implementa el HoC:

```
export const msalInstance = new UserAgentApplication(msalConfig);

You, 2 days ago | 1 author (You)
interface IState {
  authenticated: boolean;
  renewIframe: boolean;
  errorMessage: string;
  hasError: boolean;
}
```

Tenemos la instancia de MSAL en esa instancia, esa instancia es la que se encarga de ver si el usuario esta autenticado o no para ello en el momento que el componente se va a crear vemos que acción debemos de realizar si mostrar el login, bien dejar pasar al componente, bien pedir un refresh del token, etc... para ello usaremos el siguiente código:

```
public componentWillMount(): void {
  msalInstance.handleRedirectCallback(() => { // on success
    this.setState({
      authenticated: true
    });
    this.setState({
      You, 2 days ago * update
      hasError: true,
      errorMessage: authErr.errorMessage
    });
  });

  if (msalInstance.isCallback(window.location.hash)) {
    this.setState({
      renewIframe: true
    });
    return;
  }
  if (!msalInstance.getAccount()) {
    msalInstance.loginRedirect({});
    return;
  } else {
    this.setState({
      authenticated: true
    });
  }
}
```

Ahora en el render de nuestro HoC tendremos un código similar al siguiente:

```
public render(): JSX.Element {
  if (this.state.renewIframe) {
    return <div>hidden renew iframe - not visible</div>;
  }

  if (this.state.authenticated) {
    return <WrappedComponent {...this.props} />;
  }

  if (this.state.hasError) {
    return <LoginError message={this.state.errorMessage} />;
  }

  return <LoginInProgress />;
}
```

Como podéis ver lo que hace el render es en caso de que el login este en progreso o renovando el token, sale un mensaje por pantalla, si se produce un error muestra que no se ha podido autenticar y en caso de que la autenticación haya sido un éxito pues muestra el componente que nosotros queremos. En nuestro caso si queremos mostrar cualquier componente que tenga autenticación bastaría con esto:

```
export const routes = (
  <Switch>
    <Layout>
      <Route exact path="/" strict={false} component={withAuth(HomeContainer)} /></Route>
    </Layout>
  </Switch>
);
```

**"El escenario de SharePoint es el más fácil de utilizar ya que el tema de la autenticación con Graph ya lo tenía gestionado"**

Como podemos usar un componente de Graph Toolkit en nuestra aplicación. Para ello en primer lugar tendremos que añadir las referencias de Graph Toolkit para que no de error al compilar, dentro de estos elementos tendremos que añadir tantos elementos como componentes vayamos a usar, con añadirlos antes de cargar la aplicación sería suficiente:

```
declare global {
  /* eslint-disable-next-line */
  You, 2 days ago | 1 author (You)
  namespace JSX {
    You, 2 days ago | 1 author (You)
    interface IntrinsicElements {
      'mgt-flyout': any;
      'mgt-login': any;
      'mgt-person': any;
      'mgt-msal-provider': any;
      'mgt-tasks': any;
      'mgt-people-picker': any;
      'mgt-person-card': any;
      'person': any;
    }
  }
}
```

Una vez definidos el siguiente paso es configurar el SimpleProvider. Cuando creamos un SimpleProvider tenemos tres posibilidades pasarle el método de obtener el Token, el login y el logout. En nuestro caso como gestionamos el login y logout a través del HoC no se lo indicaremos y solo le indicaremos como obtener el token.

```
Providers.globalProvider = new SimpleProvider(async (scopes) => {
  const request: AuthenticationParameters = { };
  request.scopes = scopes;
  return msalInstance.acquireTokenSilent(request).then((response) => {
    return response.accessToken;
  });
});
```

Como observáis obtenemos el msalInstance que tenemos declarado en el HoC de esta forma aprovechamos la misma autenticación para ambos casos y se evita realizar dos autenticaciones al cliente. Pero no todo iba a resultar tan sencillo, sino que al gestionar el proveedor tenemos que indicar el estado en el que esta la autenticación, para ello en primer lugar cuando arranca la aplicación tendremos que indicar que la autenticación está en proceso.

```
Providers.globalProvider.setState(ProviderState.Loading);
```

Y para finalizar este estado lo tendremos que ir cambiando, dependiendo el estado de nuestra aplicación para ello en el componente HoC citado anteriormente tendremos que establecer su estado dependiendo en el estado en el que se encuentre. Quedando el código de la siguiente forma:

```
public componentWillMount(): void {
  msalInstance.handleRedirectCallback(() => { // on success
    this.setState({
      authenticated: true
    });
    Providers.globalProvider.setState(ProviderState.SignedIn);
  }, (authErr: AuthError, accountState: string) => { // on fail
    Providers.globalProvider.setState(ProviderState.SignedOut);
    this.setState({
      hasError: true, errorMessage: authErr.errorMessage
    });
  });
  if (msalInstance.isCallback(window.location.hash)) {
    this.setState({
      renewIframe: true
    });
    return;
  }
  if (!msalInstance.getAccount()) {
    msalInstance.loginRedirect({});
    Providers.globalProvider.setState(ProviderState.Loading);
    return;
  } else {
    Providers.globalProvider.setState(ProviderState.SignedIn);
    this.setState({ authenticated: true });
  }
}
```

Como podéis observar en los subrayados en rojo, en nuestra autenticación estamos gestionando que muestre la información en el estado en el que se encuentra nuestra autenticación junto con el estado de Graph Toolkit. Como

curiosidad y para entender el ciclo de vida de los componentes de GraphToolkit podéis añadir un punto de parada en la obtención del token y ver todas las llamadas que realiza (pero eso lo dejo a la curiosidad del lector)

## Escenario SharePoint SPFx

El escenario de SharePoint es el más fácil de utilizar ya que el tema de la autenticación con Graph ya lo tenía gestionado a través de permisos en el propio manifiesto de SPFx. Este tipo de autenticación lo vimos en el número 42.

Una vez obtenidos los permisos de Graph el siguiente paso que hay que realizar es inicializar el Proveedor con el contexto de SharePoint para ello eso algo tan sencillo como el siguiente:

```
// set the global provider
protected async onInit() {
  Providers.globalProvider = new SharePointProvider(this.context);
}
```

## Escenario Microsoft Teams

El escenario de Teams es el más difícil tanto de entender como de implementar y creo que en posteriores versiones sufrirá a cabo una modificación. En primer lugar, tenemos que tener claro cómo se autentica Teams. Para autenticar tu aplicación con Teams lo que debes de hacerlo es a través de una autenticación en modo PopUp (fuera del contexto de Teams) y con esa autenticación regresar a la pantalla y una vez en tu aplicación que cargue con las necesidades que se requieran. Ahora bien, en el caso de Graph Toolkit y Teams nos encontraríamos en el mismo caso que con un escenario personalizado, tendríamos que por un lado autenticar nuestra aplicación con su login (si lo requiere) y luego con el login de Teams. Para ello nos valdría con poner el siguiente código:

```
<body>
  <mgt-teams-provider client-id=" " auth-popup-url="auth.html" /></mgt-teams-provider>
  <mgt-login /></mgt-login>
  <mgt-agenda group-by-day /></mgt-agenda>
</body>
```

Ahora bien, este escenario no es el ideal para nuestra aplicación ya que la experiencia de usuario no es la mejor. Hace menos de un mes el equipo de Teams indico como podemos implementar un Single Sign On contra Azure AD en Teams. Para ello Victor Willen lo escribe perfectamente en el siguiente artículo <http://www.victorwillen.se/microsoft-teams-tabs-ssso-and-microsoft-graph-the-on-behalf-of-blog-post>

Vale eso está muy bien, pero ¿cómo podemos hacer uso de ese SSO con Graph Toolkit? Pues la solución sería haciendo uso del SimpleProvider, tendríamos que configurar la obtención del token por un lado obteniendo el token que nos da Teams, con este Teams hacer una llamada a nuestro servidor que realice la llamada a Graph para que dé un Token valido y con este token ya nuestro Graph Toolkit puede realizar las llamadas a Graph. Ahora bien, en el contexto de la aplicación de Teams en el mayor de las veces es un contexto muy similar al caso 1, tenemos una aplicación que además de poder acceder vía web se puede acceder

desde dentro del contexto de Teams. Para estos contextos tendríamos un SimpleProvider similar al siguiente:

```
if (TeamsFunctions.IsInTeams()) {
  microsoftTeams.getContext(function (context: any) {
    apiConfig.tenantId = context.tid;
    apiConfig.userId = context.userPrincipalName;
    authAPI.getTokenAsync(apiConfig.userId, apiConfig.tenantId).then(async (token: string) => {
      localStorage.setItem("adal_44", token);
      await microsoftTeams.authentication.getAuthToken({
        successCallback: (token: string) => {
          authAPI.getTokenGraph2Async(token).then((token: string) => {
            localStorage.setItem("token", token);
          });
        },
        failureCallback: (reason: string) => {
          console.log(reason)
        }, resources: [process.env.REACT_APP_MSALCLIENTID]
      });
    });
  });
  renderApp();
});
```

El tema del SSO desde mi punto de vista aun no le han dado la funcionalidad que estaba esperando, el tener dos tokens uno solicitándose a Teams y otro solicitándose a una API que hemos puesto en servidor, no es la mejor opción. Para mí se debería de obtener una autenticación similar a la que tenemos en SharePoint en la que todas las peticiones a SharePoint están incluidas en la propia autenticación que estamos puestos. Sin embargo, este paso de SSO para mí es un gran avance y que puede facilitar mucho la vida a la hora de poner una aplicación en producción, sin embargo se debería de facilitar algunos aspectos a la hora de desarrollar, porque para que el SSO funcione correctamente lo primero debes de tener un certificado de dicho dominio, y el dominio no puede contener el azurewebsites.net.

## Conclusiones

Graph Toolkit para mí es un gran proyecto Open Source que está creciendo, añadiendo cada vez más componentes, que además son los que se están usando en varias aplicaciones propias de Microsoft. Es un proyecto que está abierto para poder personalizarlo y adaptarlo a las nuevas funcionalidades que tiene Microsoft Graph, si vemos las dos últimas actualizaciones ha salido la funcionalidad de buscar sobre todos los Canales de un Teams ... y por si fuera poco hay un componente en el que puedes realizar cualquier petición a Graph. De esta forma por ejemplo si queremos usar la API de presencia o se puede usar el Componente Get.

Tener claro que Graph Toolkit no es algo que sea incompatible con otros frameworks de diseño que está apostando Microsoft como pueda ser Fluent UI. Graph Toolkit son componentes que ya tienen mucha funcionalidad y que nos permite extenderlos de una forma muy fácil. La principal funcionalidad de Graph Toolkit es el uso de Graph en los desarrollos.

### Happy Coding

#### ADRIÁN DIAZ CERVERA

Architect Software Lead at Encamina

MVP Office Development

<http://blogs.encamina.com/desarrollandosobresharepoint>  
[adiaz@encamina.com](mailto:adiaz@encamina.com) @AdrianDiaz81

## Construyendo máquinas de estados con Webhook y Logic App

Los procesos complejos basados en estados, podemos en un alto porcentaje de casos expresarlos como una máquina de estados o máquina de Moore. Si no recordamos mal una máquina de Moore es un autómata de estados, que nos indica que una salida depende de su estado, y que el siguiente estado depende del estado anterior y de la entrada producida.

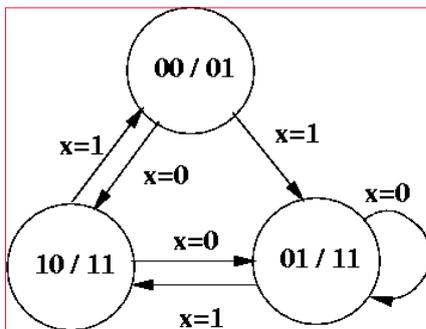


Imagen 1.- Máquina Moore.

¿Qué tipo de procesos podemos expresar con una máquina de Moore?, pues así de primeras podríamos pensar en Procesos de Aprobación que conlleven la interacción de varios usuarios y fases, el seguimiento de un pedido, un proceso de compra..., infinidad de casos que podemos pensar y plantear como una máquina de estados.

### ¿Por qué Logic App? Definición de un proceso

Logic App es un servicio que nos va a facilitar la vida para construir nuestros procesos basados en estados, ya que nos va a evitar realizar mucho trabajo y mucho código, si comparamos con elegir un desarrollo a medida con Azure Functions o un API de backend.

Dado que Logic App es un servicio pensado para diseñar e implementar procesos basados en "Eventos" y "Acciones", podemos empezar a revisar las herramientas y características que tenemos para poder definir nuestra máquina de estados, pero teniendo claro que vamos a hablar de un proceso con la siguiente pinta:



Imagen 2.- Proceso a modelar.

Nuestro proceso va a constar de "4" fases, por ejemplo, y se va a basar en un proceso de Aprobación genérico, que bien podemos por ejemplo usar para procesos de aprobación de documentos en SharePoint, para construir un proceso de aprobación de Vacaciones o la contratación de un servicio por un cliente en un CRM.

**"Logic App es un servicio que nos va a facilitar la vida para construir nuestros procesos basados en estados"**

Debemos pensar que necesitamos:

- Poder controlar el inicio del proceso, y puesta de este en "Draft".
- Poder controlar la lógica de negocio a "ejecutar" en cada uno de los estados, y por tanto controlar de alguna forma cuando inicia y termina una fase.
- Controlar las entradas a una "fase" y las salidas, para saber solo si el resultado de esta fase es "OK", para poder pasar a la siguiente.
- Poder decidir cuándo concluir el proceso.
- Poder relanzar el proceso si se produjera algún error.

### ¿Cómo lo podemos construir?, manos a la obra

Lo primero tenemos que diseñar un proceso que simule nuestra máquina de estados y para ello necesitamos definir tres puntos:

- Arranque del proceso.
- Lógica de gestión de estados.
- Finalización del proceso.

#### Inicio del proceso

Podemos iniciar el proceso con varios tipos de Acciones de "desencadenamiento" en Logic App para este tipo de procesos:

- Http Action: Esta opción, es la más sencilla de implementar, ya que nuestra Logic App va a exponer un endpoint Http, al cual podemos hacer peticiones Post o Get,

pasándole todo lo necesario para arrancar el proceso. Lo malo es que estamos “haciendo” una llamada punto a punto desde el origen del evento, y estamos metiendo una dependencia en origen, es decir si vamos a usar nuestro proceso desde Teams por ejemplo, ya debemos recoger la URL de nuestra Logic App y tener en cuenta una gestión de entornos (la URL va a cambiar), si cambiamos la Logic App a futuro debemos re-adaptar nuestra solución cliente...; demasiadas cosas a tener en cuenta, opción directa y efectiva pero que esconde oscuros mantenimientos, ya que además tendríamos que hacer llamadas async, y muy posiblemente llegar un control de la finalización del proceso con un patrón polling o con tablas en BBDD para controlar si se ha iniciado el proceso de forma correcta o no.

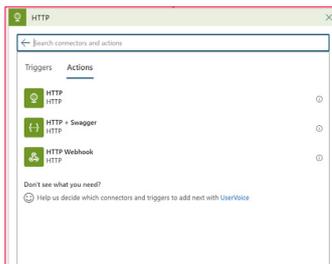


Imagen 3.- Http action.

**"Cuando la carga que recibe tu CosmosDB es impredecible y muy variada, Autopilot te garantiza que dispondrá de los suficientes recursos para satisfacer la demanda"**

- Desencadenador por Webhook: En el artículo anterior hablamos de las ventajas de usar clientes Webhook y en concreto las acciones que nos aportan Logic App. Tiene la ventaja de que es un patrón por suscripción y evitamos los patrones polling y gestionar llamadas Async desde el cliente a nuestra Logic App.

Es una opción bajo mi punto de vista más limpia que la de Http Action, y con menos problemas, pero ¿Qué pasa si nuestro cliente no consigue enviar el evento Webhook de forma correcta? ¿Cómo nos aseguramos de que el proceso se ejecute de forma correcta?

Webhook es un patrón más efectivo y estable para este tipo de llamadas que igual pueden durar días o semanas, pero igualmente la aplicación cliente va a tener que llevar no uno sino dos endpoint, el de “envío del evento al cliente webhook”, y el de “desuscripción al mismo”.

- Trigger desde una Cola: Si hemos escogido una de las dos primeras opciones, tranquilo te va a funcionar, y si tenemos en cuenta los “Warning” que he ido relatando tendrás un control de arranque de tu proceso perfecto. Pero ¿Y si usamos una cola de procesos?, es decir si nuestro cliente por ejemplo cuando quiera iniciar el proceso, deja en una cola de peticiones una del tipo “inicio proceso”, con un Identificador de este,

podríamos controlar de forma directa con un pequeño Trigger o incluso añadiendo Event Grid si el volumen lo pidiera el inicio del proceso en Logic App.

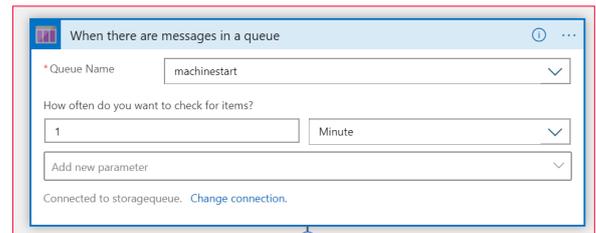


Imagen 4.- Trigger Queue Blob Storage.

Nuestro flujo va a evaluar una cola de procesos, y cuando entre uno nuevo se va a ejecutar y va a coger el mensaje. Si por un casual esta Logic App falla, podemos controlar que el mensaje persista en la cola, para que un proceso posterior se reinicie, e incluso como veremos un poco más adelante si llevamos los estados de nuestra Logic App por base de datos podemos relanzar el proceso por donde se haya quedado anteriormente.

## Control de los estados, y llamadas a procesos de lógica de negocio

Una vez hemos aclarado como iniciar nuestro proceso con una de las tres opciones, tenemos que ver cómo vamos a controlar “los cambios de estados”, y como “ejecutar la lógica de negocio” de cada uno de los estados de nuestro proceso.

- Acciones de control: Switch, Until y Condition: Para poder controlar el estado en el que nos encontramos “Draft” o “Approved”, debemos hacer uso de las Acciones de Control que nos aporta Logic App y que vamos a ir viendo poco a poco como incorporarlas a nuestro proceso basado en una máquina de estados.

Con la Acción “Until”, vamos a controlar que el proceso no se detenga hasta que el “estado o el paso” no llegue a un estado de fin de flujo, o lo que en nuestro ejemplo es el estado Approved.

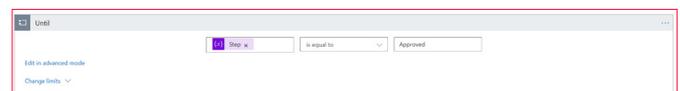


Imagen 5.- Acción Until.

Toda acción que insertemos dentro de una acción Until, se va a ejecutar “N veces” hasta que la condición del estado sea igual a Approved, tal y como vemos en la imagen anterior. Esta acción viene a simular una instrucción “While”, tal y como hacemos en C#.

Dentro de la acción Until, deberemos incluir una acción “Switch”, que va a venir a darnos esa forma de “árbol de decisión”, y que nos va a permitir posicionar el proceso en el estado correcto, y pasar al estado siguiente cuando termine el estado en curso.



Imagen 6.- Switch Control.

Para nuestro Switch podremos añadir tantas “opciones” como estados queramos definir, en este caso tres para “Draft”, “Pending Approval” y “Pending of Review”.

Como vemos en la imagen superior, debemos definir una variable “step” o paso que va a llevar el estado del proceso actual, y en función de este valor se va a ejecutar la lógica que definamos para cada uno de los estados.

Antes de pensar en “que meter en cada fase o paso”, debemos pensar ¿Cómo finalizamos el proceso? ¿Cómo pasamos a un estado siguiente?

Si volvemos al “diseño” genérico del proceso vemos que cada estado da una salida booleana de “Ok o Ko”, y en función de eso pasamos por ejemplo de Draft a Pending Of Review.



Imagen 7.- Proceso basado en estados.

**"Antes de pensar en “que meter en cada fase o paso”, debemos pensar ¿Cómo finalizamos el proceso? ¿Cómo pasamos a un estado siguiente?”**

Para conseguir esto debemos por un lado “definir una variable” de salida de cada fase para controlar si el estado ha terminado de forma correcta o no, y por otro lado Añadir un control de condición para saber si debemos pasar a un estado siguiente, o bien reiniciar el proceso o finalizarlo si fuera necesario, en nuestro caso veremos que he decidido que, si el estado no da una salida de OK, siempre reinicio a un estado draft.

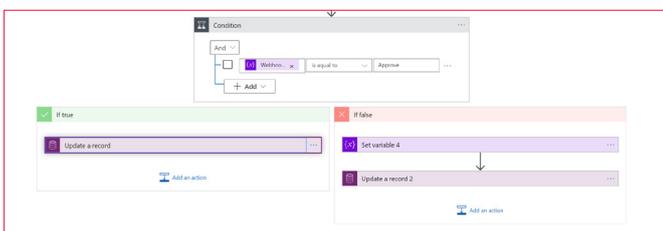


Imagen 8.- Condiciones.

Una acción de “Condición”, viene a ser un “If – else”, y que nos va a permitir ejecutar un bloque de proceso u otro, como vemos en la imagen anterior.

Si nos fijamos en la imagen, yo he optado por llevar una persistencia del proceso en Common Data Service, donde voy guardando el estado del proceso, una pequeña auditoría de por dónde va...; y he optado por esta opción porque mi aplicación cliente como veremos en la propuesta de solución es una solución en PowerApps, y Common Data Service es una solución

muy limpia y que se integra con todas las partes de forma nativa mediante un conector.

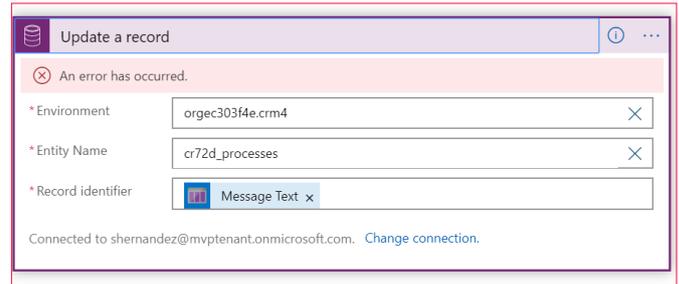


Imagen 9.- Persistencia en CDS.

- Llamadas Webhook a “procesos del estado”: Cada paso del proceso tiene una lógica de negocio, que puede ir desde pedir una aprobación a un usuario vía Outlook, a integrar un API de Backend y ejecutar múltiples procesos ...; en este sentido la imaginación y el coste es el límite. Nosotros nos vamos a centrar en “como” vamos a orquestar estos “subprocesos” y esta lógica desde nuestra máquina de estados. En este caso sí o sí, yo optaría por escoger llamadas Webhook a los subprocesos, con la Acción Http Webhook.

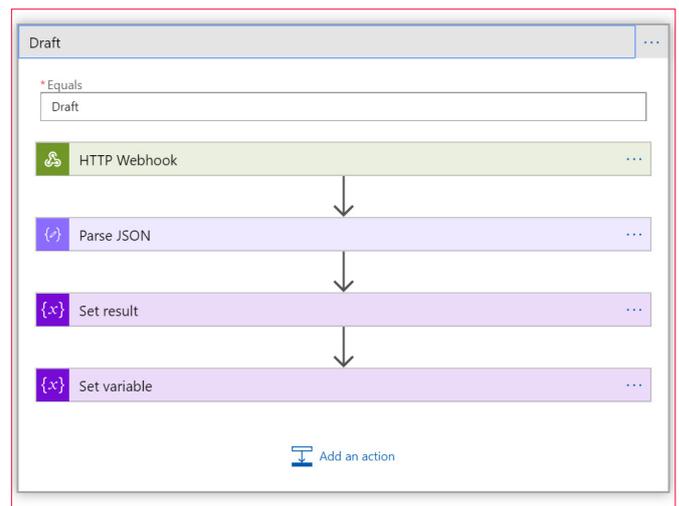


Imagen 10.- Estado Draft, llamando al subproceso.

Si nos centramos en “revisar el estado Draft”, podemos ver que tenemos una llamada a nuestro proceso de la fase con una llamada Webhook, y esta acción se va a quedar esperando hasta que nuestro subproceso haya finalizado.

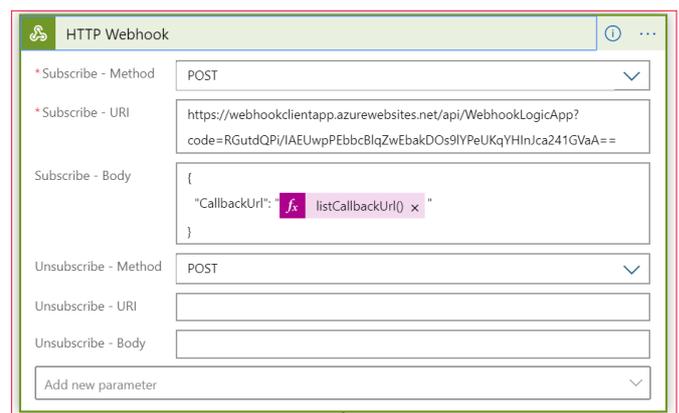


Imagen 11.- Http Webhook.

Esta acción Webhook, debe pasarle al backend del proceso de la fase, un "Call back Url" para que cuando termine su trabajo avise a nuestra máquina de estados y pueda continuar con las siguientes fases.

Cuando termina cada llamada Webhook, y por lo tanto el subproceso le remite el estado del subproceso al proceso principal y por lo tanto podemos evaluar con las condiciones si ha sido correcta la fase o no (salida del estado).

**"debe pasarle al backend del proceso de la fase, un "Call back Url" para que cuando termine su trabajo avise a nuestra máquina de estados"**

En mi caso, mi subproceso se "inicia" con una llamada de Http Webhook a un Azure Functions, y este Azure Functions almacena en una cola el Callback Url para que cuando termine los subprocesos relacionados a esta fase (podría haber uno o varios), podamos indicarle a la Logic App que debe pasar a evaluar la fase.

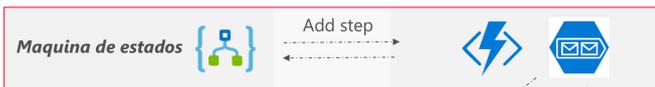


Imagen 12.- Subprocesos.

Imaginemos qué, por cada Paso, queremos lanzar "N procesos", para ello, la llamada Webhook va a hacer una suscripción a nuestra Azure Functions, y esta va a encolar la petición en una cola de procesos, haciendo algo muy parecido a la estrategia de inicio de la máquina de estados.

Por cada entrada en la cola, podemos tener escuchando tantos procesos como queramos, y ejecutar en paralelo todo lo necesario.

Usaremos nuevamente el servicio de Common Data Service, para poder llevar un control del estado del subproceso(s), y al término de esta fase, se indicará a la máquina de estados vía el Call back Url con una petición post, que puede empezar a evaluar la fase en curso y plantear la ejecución de la siguiente fase.

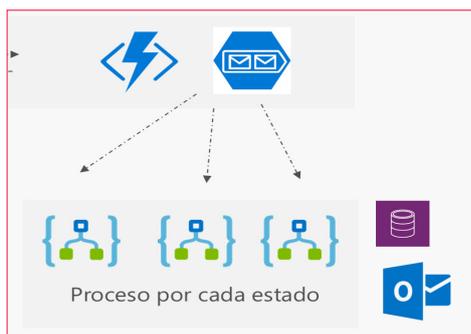


Imagen 13.- Cola de Subproceso.

Volviendo a la máquina de estados, y en concreto al

Estado "Draft" de ejemplo, veremos que debemos almacenar en una variable el resultado de la ejecución de la fase que nos ha enviado el subproceso y debemos establecer la siguiente fase que en este caso sería "Pending Of review".

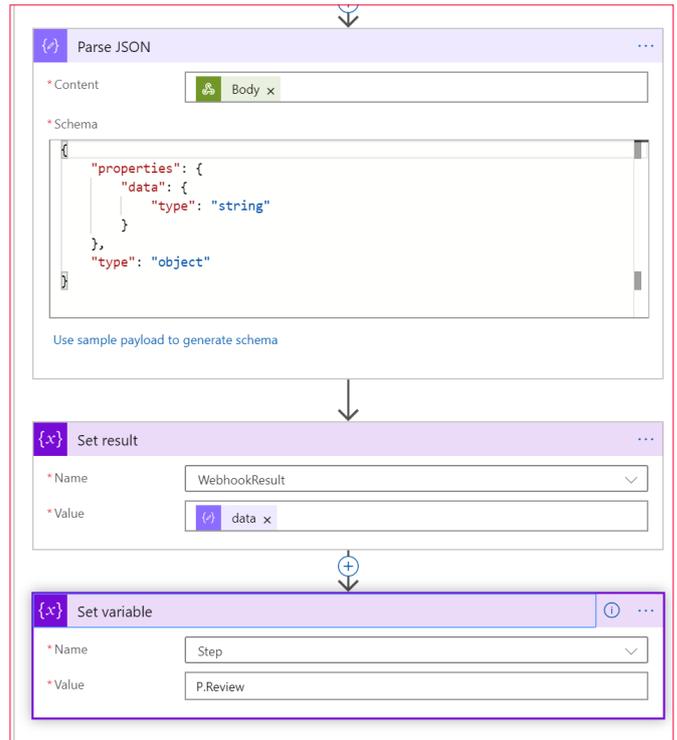


Imagen 14.- Control de Callback.

Recordemos que este "Set variable" que vemos en la imagen, es una tentativa de cambio de estado, pero que debe ser validada por la salida de la fase con la acción de condición que nombramos antes.

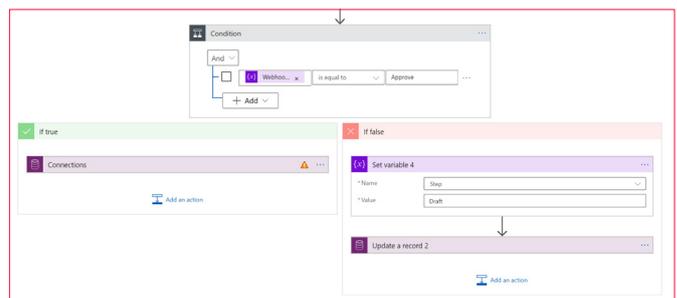


Imagen 15.- Condición de fin de Flujo.

Si el resultado del subproceso, que hemos almacenado en la variable "webhook-result", no es "Approved", deberemos mandar el proceso a un estado Draft y por el contrario almacenamos en el Common Data Service que el proceso cambia de fase.

### Finalización del proceso, control de errores y reintentos

La última cosa que debemos tener en cuenta para dotar de funcionalidad completa a nuestra máquina de estados es diseñar como controlar los errores del subproceso o del propio flujo de Logic App, y aplicar un control de reintentos.

Si nos fijamos en la siguiente imagen, podemos dividir el proceso en 3 bloques:

- Inicio del proceso: Recogemos la petición de inicio desde la cola de procesos e inicializamos las variables para controlar las entradas a cada fase y las salidas (webhook result).
- Ejecución de la máquina de estados: Esta combinación de Switch + Until, para ir lanzando las llamadas a los subprocesos vía llamada al cliente Webhook que escojamos.
- Evaluación de cambio de fase: Acción de condición, para ver si la fase ha sido aprobada o no.

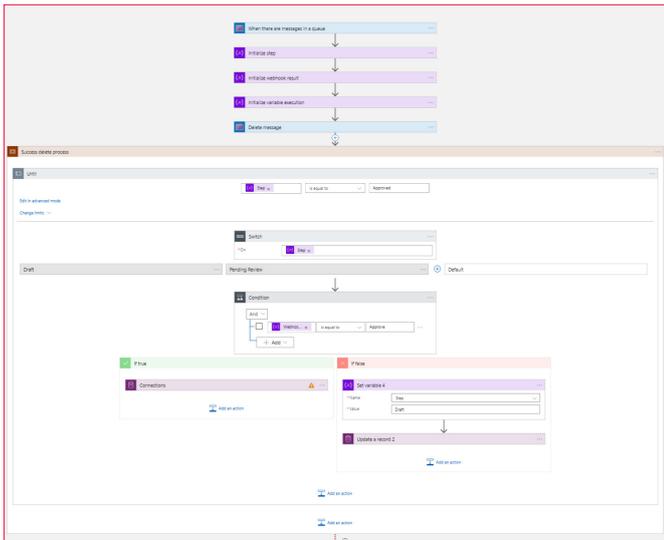


Imagen 16.- Flujo Completo.

Todos estos bloques los hemos englobado dentro de una acción de control que nos aporta Logic App y que se llama Ambito o Scope.

Esta acción permite simular un bloque “try – catch”, y capturar toda excepción y actividad que se produzca en el Scope de ejecución.

A este Scope que le podemos definir bloque de “Acierto o Try”, le podemos vincular un ámbito de captura de errores.

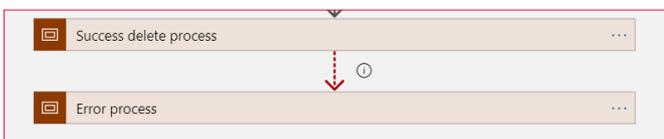


Imagen 17.- Ámbitos.

El segundo “Scope” lo vamos a utilizar para poder recoger las posibles excepciones del primer Scope “Success delete process”. Si seleccionamos en la configuración de este Scope, podemos seleccionar la opción “Run-after”. Como vemos en la imagen, hemos seleccionado que siempre que falle el Scope anterior, debemos ejecutar este, y solo en este caso.

**“ Esta acción permite simular un bloque “try – catch”, y capturar toda excepción y actividad”**

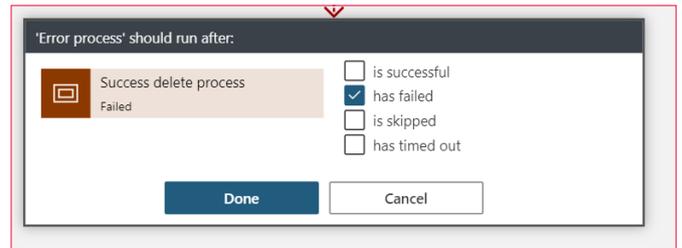


Imagen 18.- Control de errores.

En mi ejemplo, simplemente capturo el error en el proceso, y como estamos guardando el estado de este en Common Data Service, nos costaría muy poco desde la aplicación cliente relanzar el proceso con una simple inserción en la cola de procesos de entrada.

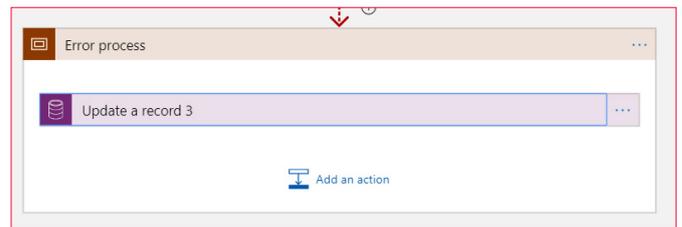


Imagen 19.- Qué hacemos con el error.

## Análisis global del proceso y de la solución

A más alto nivel, y haciendo un poco de resumen tenemos una solución parecida a la siguiente imagen:



Imagen 20.- Arquitectura Global.

## Haciendo un poco de resumen, tenemos que:

- Aplicación Cliente: Podríamos pensar en una solución cliente cualquiera, yo en mi idea de solución he planteado que PowerApp es una buena opción, para que deje en CDS la persistencia de cada petición y de cada proceso.
- Máquina de estados: Control de fases e inicio de procesos: Desde CDS o la tabla que elijamos para persistir los procesos debemos conectar nuestra Cola de procesos, en mi caso una buena opción es conectar un Azure Functions con una suscripción Webhook a CDS, y volver como salida todos los datos del proceso en la cola. De esta cola de procesos configuramos un Trigger de salto en nuestra máquina de estados, y con esto tene-

mos conectado el cliente con la máquina de estado, y si lo hacemos con Colas y Webhook, podemos deducir que encima en tiempo real.

- Subprocesos y control de estos: Cada subproceso lo debemos conectar a nuestra máquina de estado en Logic App con una cola de procesos y una suscripción Webhook, que consolidaremos con Acciones Webhook entre Azure Functions y nuestra Logic App máquina de estados. Cada subproceso con otro Trigger, conseguimos que se ejecute mediante una cola de procesos en un Blob Storage, y que estos procesos actualicen los estados directamente en una tabla de CDS (Common data Service). Por último, es pensar que va a hacer cada subproceso, en la imagen yo he incluido Outlook porque es un buen conector de Logic App para poder lanzar aprobaciones vía mail, pero esto ya sí que es 100% libre, es la idea de este artículo que cada uno adapte el proceso a sus necesidades.

## Conclusiones, ¿Cómo seguimos?

Bueno este artículo, es 100% subjetivo por mi parte, y es una idea, más bien un ejemplo de que con Logic App podemos construir procesos complejos, que van mucho más allá de un pequeño flujo de integración, y que podemos perfectamente construir procesos empresariales comple-

jos basados en estados.

¿Cuánto nos puede llevar implementar esto desde cero y a medida?, yo creo que la orquestación de estados, la integración y la persistencia de los procesos con Logic App, un sistema de Colas y una BBDD a elegir lo tenemos, y lo podemos conectar en pocas jornadas todo y dejarlo diseñado sin pasar por una implementación costosa.

Luego ya hay que pensar que cada subproceso puede ser más o menos complejo, y ahí puede entrar nuestro código a medida si fuera necesario o más Logic App si se van a basar en procesos de integración.

En conclusión, podemos decir y asegurar que Logic App ha venido para quedarse. En el artículo anterior vimos que podemos hacer llamadas Async complejas y hoy que podemos definir procesos duraderos y orquestados, y que sin duda si sabemos diseñar de forma correcta nos puede hacer diferenciales respecto a procesos a medida que en ocasiones son difíciles de monitorizar y encajar en organizaciones complejas.

---

**SERGIO HERNÁNDEZ**

**Azure MVP**

# Mentoring

## Comparti MOSS

Un servicio experto alrededor de su SharePoint



CompartiMOSS le puede ayudar a través de su programa de Mentoring!

Contacte con nosotros y le enviaremos los planes de mentoring que tenemos disponibles para SharePoint.



# Modern Search con PnP WebParts - Introducción

En este artículo vamos a revisar que opciones de personalización tenemos a la hora de trabajar con Search y Modern, y veremos qué nos ofrece la solución OpenSource del PnP “PnP Modern Search”.

## ¿Qué tenemos de Search en Modern, Out of the box?

Admitámoslo, Modern tiene muchas cosas buenas, pero, hoy en día, la búsqueda no es una de ellas. De hecho, si comparamos toda la potencia de personalización y configuración que tenemos en classic, donde podemos hacer cosas tan potentes y, por qué no decirlo, bonitas, como esto:

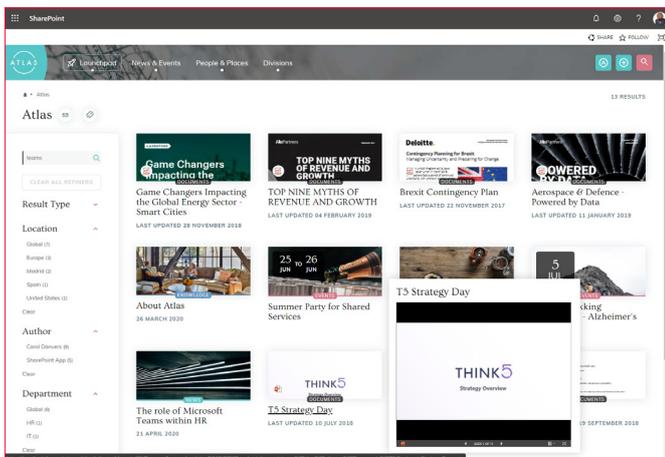


Imagen 1.- Ejemplo de personalización de búsquedas en sitios clásicos de SharePoint Online (SPO).

Y vamos ahora a la experiencia Modern, con algo como lo siguiente:

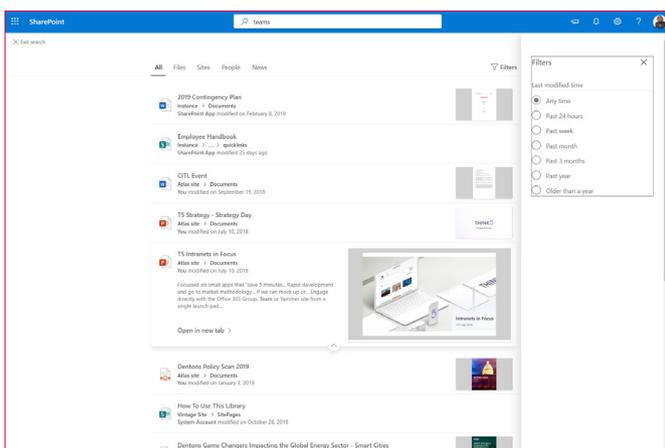


Imagen 2.- Experiencia de usuario de búsquedas modernas en SPO.

**"Modern está lejos de lo que podemos hacer en Classic"**

Pues, para qué negarlo, Modern está lejos de lo que podemos hacer en Classic. Seguramente estés pensando: “vale, pero es que en Classic, para tener algo como la imagen anterior, ha habido que personalizar cosas, y hacer algo de development” ... Totalmente en lo cierto. Y ahí es a donde voy, que en Classic, podemos hacer toda esa personalización, y en Modern, no. Por ejemplo, respondamos las siguientes preguntas sobre Modern search:

- ¿Podemos añadir diferentes Filtros/Refiners?... NO. (sólo tenemos el Filtro por fecha de modificación).
- ¿Puedo mover el panel de filtros para que esté siempre presente, por ejemplo, a mano izquierda de la página?... NO.
- ¿Podemos añadir nuevos Verticales, por ejemplo “Videos”?... NO.
- ¿Puedo hacer que un documento de un tipo concreto se muestre con una presentación diferente al resto? NO.
- ¿Puedo cambiar la presentación de los resultados?... NO.
- ¿Puedo hacer que la caja de búsqueda del header, que es Out of the box, redirija la query de búsqueda a una página personalizada mía? ... anda mira, esto SÍ!

Y es que es este último punto, unido a los WebParts search modern que nos ofrece la comunidad PnP, lo que nos va a permitir mejorar muchísimo la experiencia Modern Search, pudiendo llegar a algo parecido a la siguiente imagen:

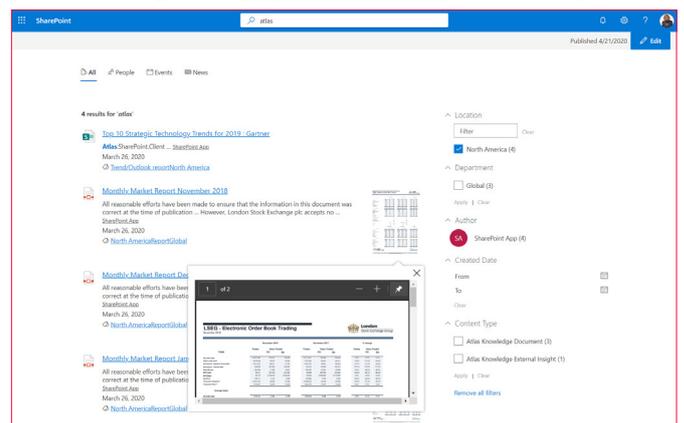


Imagen 3.- Ejemplo de búsqueda moderna personalizada.



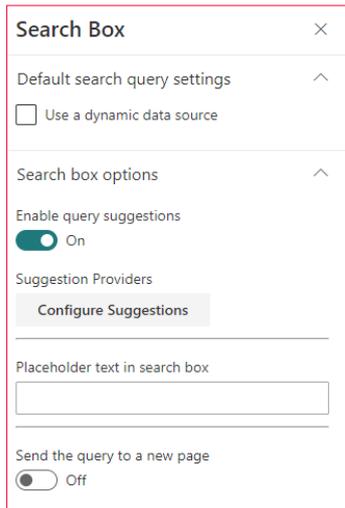


Imagen 6.- WebPart de caja de búsqueda.

Part son bastante amplias y van a cubrir la mayoría (si no todas) de nuestras necesidades.

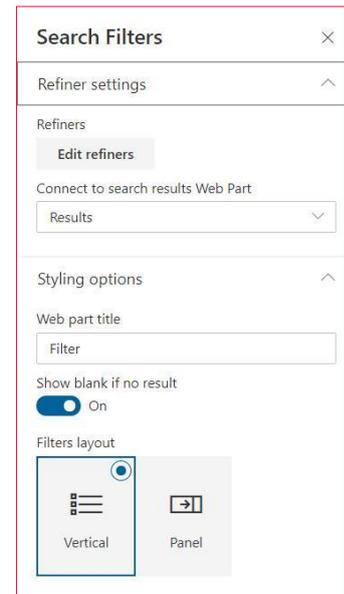


Imagen 8.- WebPart de refinadores.

## Search results

Aquí tenemos el WebPart estrella de la solución, y el que nos va a permitir mostrar los resultados de la búsqueda. Es tremendamente configurable y nos va a dar prácticamente control total a la hora de presentar los resultados.

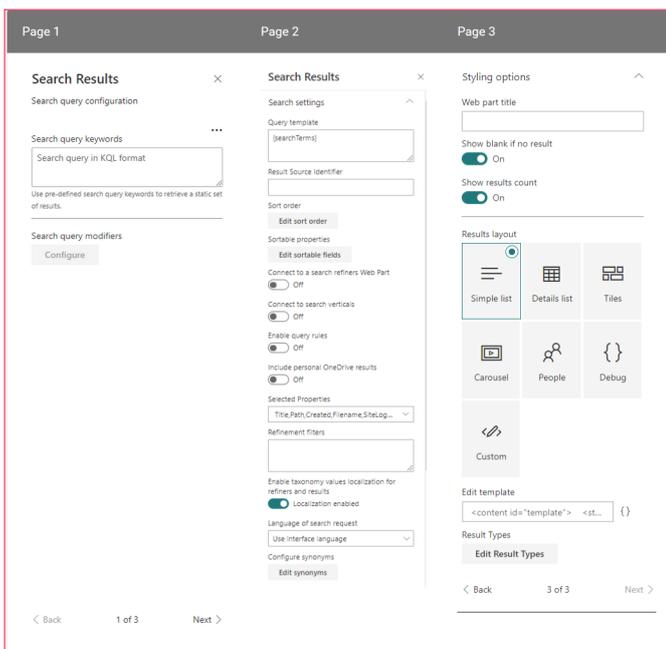


Imagen 7.- WebPart de resultados.

## Search Verticals

Este WebPart nos va a permitir configurar verticales de búsqueda (Scopes). La idea es muy sencilla, simplemente defines tu Result Source (por ejemplo, quiero un scope de búsqueda sobre todos mis Eventos, que al final no deja de ser un tipo de contenido). Este WebPart nos permite configurar dichos Scopes y los renderiza como Tabs.

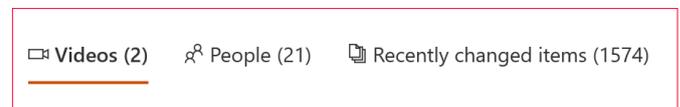


Imagen 9.- WebPart de Search Verticals.

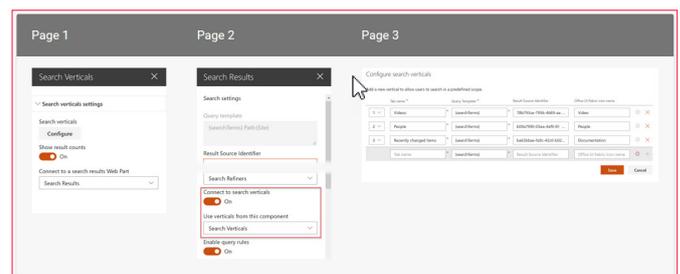


Imagen 10.- WebPart de Search Verticals.

El Sistema para la presentación de los resultados se basa en plantillas de Handlebars (<https://handlebarsjs.com/>), este es un framework de plantillas muy extendido y con muchas posibilidades. Además, la solución nos va a permitir poder añadir nuestros propios componentes de ReactJS, y usar dichos componentes en las propias plantillas, por lo que tenemos casi opciones infinitas a la hora de presentar los resultados.

## Search filters

Aquí tenemos nuestro panel de Refiners “de toda la vida” Es también muy configurable. Por el momento, no tenemos las mismas opciones de “templating” que nos ofrece el Search results WebPart, pero las que trae el propio Web-

Y hasta aquí nuestro artículo. Como veis, las posibilidades de personalización en modern search son tremendamente ampliadas gracias a los WebParts del PnP, pudiendo llegar a configurar auténticos Search Centres al más puro estilo classic. En próximos artículos veremos con más detalle algunos de estos WebParts, así como la manera de extender su funcionalidad.

**¡Hasta el próximo artículo!**

**LUIS MAÑEZ**

Cloud Architect en ClearPeople LTD

@luismanez

<https://github.com/luismanez>



43

# Dimensionando desescalapp.com en Azure

Todo empieza cuando te llaman porque han publicado [desescalapp.com](https://desescalapp.com) en Azure y, de repente, aparecen 1000 usuarios concurrentes usando las funcionalidades que has publicado.

Esto no estaba previsto, no hemos podido dimensionar los servicios correctamente, ¿Qué hacemos? No podemos perder a los usuarios actuales, tenemos que seguir dando servicio y que todo funcione adecuadamente, pero no con los tiempos de respuestas que se estamos teniendo.

Ante esta situación de éxito, lo más rápido e indoloro es que escales horizontalmente, o lo que es lo mismo, añade más nodos a tu aplicación para que pueda repartir la carga de trabajo entre todos los servidores que estén disponibles y así atender las peticiones de todos los usuarios.

**"Todo empieza cuando te llaman porque han publicado desescalapp.com en Azure y, de repente, aparecen 1000 usuarios concurrentes"**

A partir de este punto, tenemos los usuarios atendidos y podemos empezar a analizar la situación.

## Analizar el App Service

Necesitamos ver los datos de uso, ver que CPU se está consumiendo, cuánta memoria RAM, cuántas peticiones por segundo, etc. Estos datos nos van a permitir elegir el dimensionamiento adecuado de la aplicación para los usuarios que están usando la misma.

Un [App Service](#) es un servicio PaaS de Azure que nos permite publicar aplicaciones web decidiendo cuál es el escalado vertical (CPU y RAM) y el escalado horizontal (número de servidores).

Dev / Test For less demanding workloads	Production For most production workloads	Isolated Advanced networking and scale
<b>Recommended pricing tiers</b>		
<b>S1</b> 100 total ACU 135 GB memory A-Series compute equivalent 63.56 EUR/Month (Estimated)	<b>P1V2</b> 210 total ACU 3.5 GB memory Dv2-Series compute equivalent 123.12 EUR/Month (Estimated)	<b>P2V2</b> 420 total ACU 7 GB memory Dv2-Series compute equivalent 246.24 EUR/Month (Estimated)
^ See only recommended options		
<b>Additional pricing tiers</b>		
<b>S2</b> 200 total ACU 315 GB memory A-Series compute equivalent 123.12 EUR/Month (Estimated)	<b>S3</b> 400 total ACU 1 GB memory A-Series compute equivalent 246.24 EUR/Month (Estimated)	<b>P1</b> 100 total ACU 1.75 GB memory A-Series compute equivalent 164.68 EUR/Month (Estimated)
<b>P3</b> 400 total ACU 7 GB memory A-Series compute equivalent 738.73 EUR/Month (Estimated)	<b>P2</b> 200 total ACU 315 GB memory A-Series compute equivalent 369.37 EUR/Month (Estimated)	

Imagen 1.- Planes de escalado vertical de un App Service en Azure.

La aplicación estaba configurada con servidores del tipo S1 y, analizando el uso de CPU, veíamos que era elevado para el número de usuarios que teníamos, unos 2.000, concurrentes.

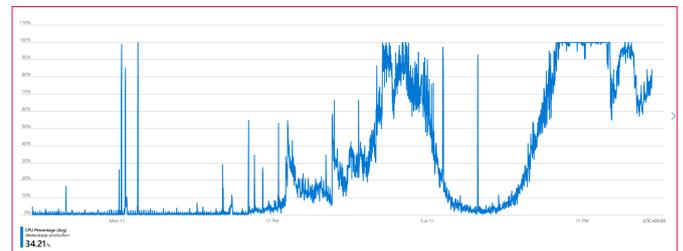


Imagen 2.- Consumo de CPU.

La memoria RAM estaba por encima del 60%, no está mal, pero estábamos corriendo el riesgo de necesitar más nodos con el aumento de los usuarios que se estaba recibiendo.

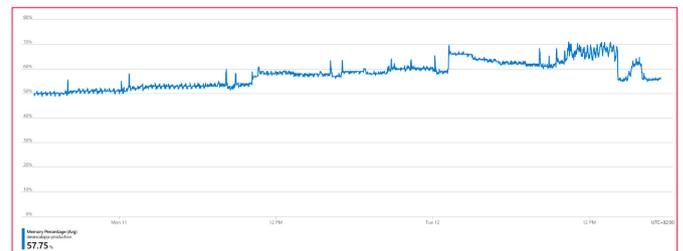


Imagen 3.- Consumo de memoria RAM.

Las peticiones eran bastante elevadas, estábamos recibiendo 20.000 peticiones por segundo con una media de 2.000 usuarios, lo que implicaba un 10x con respecto a los usuarios.

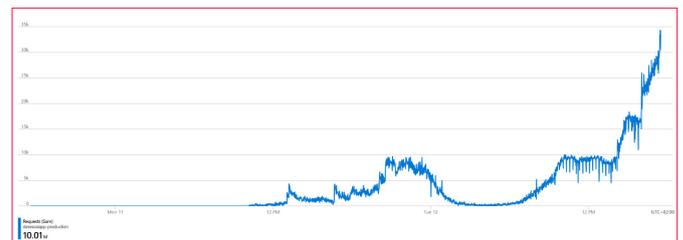


Imagen 4.- Peticiones al servidor web

## Optimizamos el App Service

Teniendo en cuenta el modo de uso de la aplicación actual, fue necesario realizar algunos cambios necesarios porque el volumen de usuarios había aumentado considerablemente y no era factible tener 10 instancias (gracias a Espejo Público y a [Antena 3](#)) S1 de la aplicación, por dos motivos

básicos, ya habíamos llegado al máximo de nodos y económicamente es un disparate para 3.000 o 4.000 usuarios.

La CPU es importante pero también el acceso a disco que es donde están todos los recursos web de la aplicación, hojas de estilos, imágenes, JavaScript, etc. Esto nos llevó a tener que escalar verticalmente a una instancia Premium que tiene mejor rendimiento de CPU y mejor rendimiento de lectura/escritura de recursos con discos SSD.

Los usuarios seguían aumentando y el cambio permitió llegar a picos de 20.000 usuarios concurrentes con cuatro instancias de P2V2. La CPU respondía correctamente, al igual que la RAM.



Imagen 5.- Consumo de CPU.

Cuando vez una gráfica de consumo como la anterior, es fácil identificar que exista un patrón de consumo, que no es necesario tener todas las instancias activas en momentos en los que desciende el número de usuarios. Para esto, lo ideal es configurar un escalado horizontal automático, es decir, que vaya aumentando o disminuyendo el número de instancias en función del consumo.

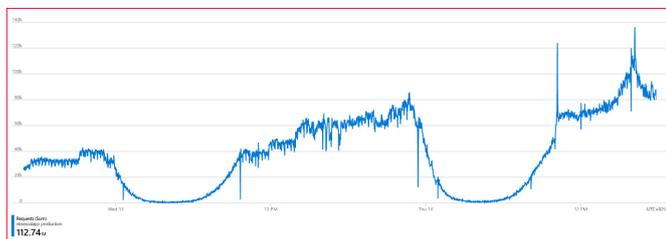


Imagen 6.- Peticiones al servidor web.

Con el ratio de peticiones por usuario que estábamos teniendo y analizando las dos gráficas, vemos una relación clara que nos llevó a configurar el autoescalado horizontal en base al uso de la CPU. El objetivo es que cuando la CPU se mantiene por encima del 85% durante un tiempo, añadimos una nueva instancia que permite atender al aumento de usuarios que se está produciendo, y al contrario cuando la CPU está por debajo del 60%.

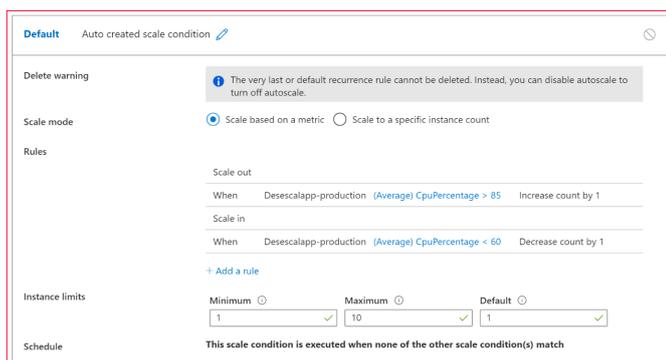


Imagen 7.- Configuración del autoescalado.

## Configuraciones importantes en el App Service

En el App Service tenemos ciertas configuraciones que nos permite adaptar el servidor web a la aplicación que estamos ejecutando. En este caso, configuramos lo siguiente:

- El stack tecnológico lo cambiamos de .NET para que se ejecute de forma nativa como aplicación .NET Core.
- La plataforma la cambiamos de 32 bit a 64 bit.
- La versión de HTTP la cambiamos a 2.0 que ofrece un mejor rendimiento que la versión 1.1.
- Deshabilitamos la afinidad de sesión de las cookies ya que no es necesario tener esta funcionalidad y mejora bastante el rendimiento para las aplicaciones sin estado como esta.
- Habilitamos la cache local de los recursos de la aplicación, esto es importante para que cuando tengamos más de una instancia de aplicación, no vaya a buscar los recursos al almacenamiento compartido y los encuentre en local en el servidor.

## Analizar la aplicación

A partir de aquí, teníamos un sistema estable atiendo a muchos usuarios concurrentes, llegamos a tener unos 100.000 y el escalado que se configuró funcionaba correctamente, con algunos problemas de rendimiento en los momentos en los que estaba escalando, pero eso es normal y lo estuvimos ajustando unos días para no resentir mucho los tiempos de respuesta.

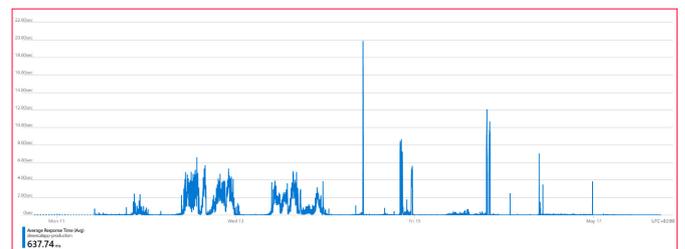


Imagen 8.- Tiempo de respuesta medios.

Tocaba analizar la aplicación con el objetivo de optimizar el código y el uso de los recursos. Para eso, usamos el [servicio de diagnóstico](#) que está disponible en el App Service, analizando la disponibilidad y el rendimiento. Este servicio nos confirmó uno de los posibles problemas que estábamos teniendo con el número de peticiones y el uso de la CPU. Al parecer, teníamos un exceso de conexiones TCP que se podían reutilizar.

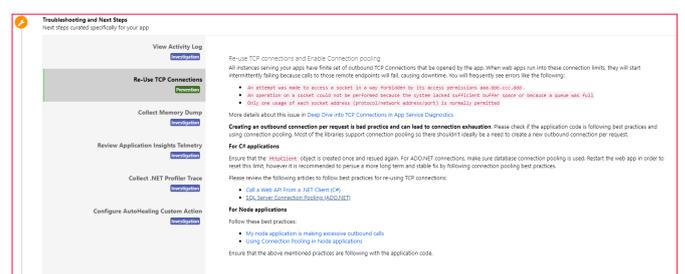


Imagen 9.- Diagnóstica de problemas con las conexiones TCP.

**"teníamos un sistema estable atendiendo a muchos usuarios concurrentes, llegamos a tener unos 100.000 y el escalado que se configuró funcionaba correctamente"**

Los servidores tienen un límite de conexiones TCP abiertas y, aunque habíamos pasado de 1.920 a 3.968 conexiones con el cambio del S1 al P2, seguíamos con peticiones en cola y con la necesidad de tener más nodos (escalado horizontal) de los que deberían hacer falta para el número de usuarios concurrentes que teníamos.



Imagen 10.- Conexiones TCP.

Una buena oportunidad para mejorar el rendimiento, ¿qué conexiones se estaban haciendo desde la aplicación? La aplicación es una ASP.NET MVC Core que ofrece contenido al usuario en diferentes rutas con una única conexión a un SQL Server que tenía los datos donde se busca por código postal.

## Optimizamos la aplicación

Si sólo teníamos conexiones TCP al SQL Server, estaba claro que se podía mejorar estas conexiones y reducir el uso de TCP usando un pool de conexiones, un patrón que permite reutilizar las conexiones TCP y no estar abriendo y cerrando conexiones por cada petición que se realice. Con Entity Framework es muy fácil hacer esta configuración, básicamente hay que configurar un DbContextPooling que se encarga de gestionar y reutilizar las conexiones al servidor.

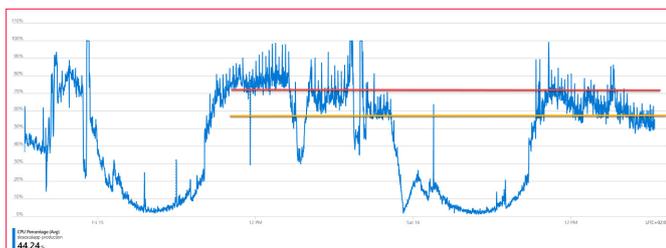


Imagen 11.- Cambios en el uso de la CPU.

Esta actualización nos permitió reducir el número de conexiones abiertas al SQL Server y mejorar entre un 10% y un 15% el consumo de CPU. Recordemos el tema de las peticiones, seguimos teniendo una ratio de 10x peticiones por usuario, un ratio bastante excesiva para la funcionalidad que la aplicación ofrece. ¿Cómo podemos optimizar y reducir estas peticiones? Más del 90% de las peticiones de

la aplicación se realizan para obtener recursos web, básicamente para obtener las hojas de estilos, los JavaScript y las imágenes que acompañan al contenido.

Eso tiene fácil solución y Azure tiene un servicio para esto que es usado por la mayoría de los sitios webs públicos, Content Delivery Network, CDN para los amigos. Este servicio nos va a permitir descargar de peticiones el servidor y que simplemente se realicen aquellas peticiones que son para cargar datos o nuevas páginas con datos, las imágenes, las hojas de estilos y los JavaScripts serán entregados a través de los servidores que los CDN tienen distribuidos por las redes globales, lo que, además, nos ayuda a mejorar el rendimiento en la carga de estos tipos de ficheros porque están cacheados en múltiples servidores cercanos a los usuarios.

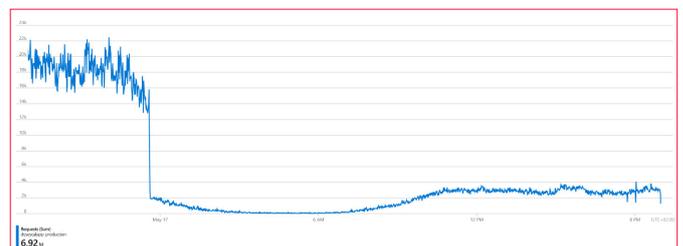


Imagen 12.- Peticiones con la versión CDN.

**"Azure tiene un servicio para esto que es usado por la mayoría de los sitios webs públicos, Content Delivery Network, CDN para los amigos"**

La gráfica de peticiones ha cambiado drásticamente y, con el mismo número de usuarios concurrentes, refleja mejor el comportamiento de la aplicación. Es más, podemos ver en el siguiente gráfico como hemos conseguido también bajar el consumo de CPU, lo que nos va a permitir disminuir el escalado vertical con un tamaño de instancia más económico.

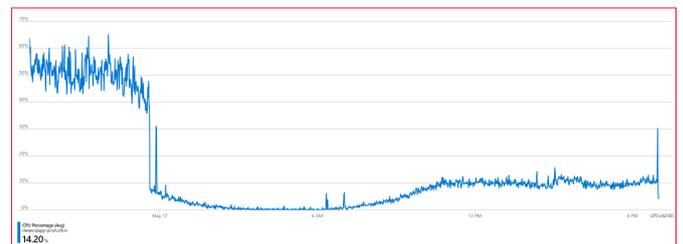


Imagen 13.- Uso de CPU con la versión CDN.

## Lecciones aprendidas

No ha terminado el trabajo, sólo llevamos una semana monitorizando y optimizando los recursos para poder atender a la demanda de los usuarios. Parece que hemos identificado algún que otro patrón de consumo, pero en menos de una semana no es posible identificar todos estos patrones y vamos a tener que realizar alguna optimización más, aunque sea para dimensionar a la baja los servicios

que estamos consumiendo. Tengamos en cuenta que estamos hablando de una aplicación que, apuesta por la labor social, no una empresa con buenos fondos económicos, por lo que la optimización del consumo de servicios es vital para no quedarnos sin presupuesto.

*"hay que tener mucho cuidado con las conexiones TCP que realizamos, bien porque nos conectamos a servicios HTTP o, como en este caso, a servicios de base de datos como SQL Server"*

Lo habitual en este tipo de proyectos es realizar pruebas de carga para dimensionar correctamente los servicios y optimizar adecuadamente el código, pero es complicado conocer a priori el número de usuarios que van a usar tu aplicación por lo que cualquier prueba de carga que se hubiera realizado seguro que estaba fuera de la realidad y no íbamos a cumplir las posibles previsiones.

A nivel de código, hay que tener mucho cuidado con las conexiones TCP que realizamos, bien porque nos conectamos

a servicios HTTP o, como en este caso, a servicios de base de datos como SQL Server. En la medida de lo posible, hay que implementar un patrón tipo pooling para estas conexiones, que nos ayude a reutilizar las conexiones abiertas y no intentar tener conexiones infinitas, ya que los servidores tienen sus límites.

Importante la utilización de servicios de CDN, además de mejorar el rendimiento en la descarga de los recursos estáticos, descargamos de peticiones al servidor web, lo que mejora el consumo de la CPU, lo que nos permite optimizar aún más el consumo de servicios, lo que supone bajar los costes.

Por último, no nos olvidemos de configurar adecuadamente el servicio donde hemos desplegado nuestra aplicación, un simple cambio de 32 bits a 64 bits, puede ser un buen ejemplo de mejora en el rendimiento.

**ALBERTO DIAZ MARTIN**

MVP Azure

[adiazcan@hotmail.com](mailto:adiazcan@hotmail.com)

@adiazcan

<http://blogs.encamina.com/por-una-nube-sostenible/>

En **encamina** buscamos:

- ★ Desarrolladores .NET
- ★ Desarrolladores Dynamics 365
- ★ Consultores Office 365
- ★ Consultores CRM
- ★ Consultores de Azure

Si tú también **piensas en colores**

¡ Queremos tu talento !  
[rrhh@encamina.com](mailto:rrhh@encamina.com)



**encamina**

[@encamina](#) [f](#) ENCAMINA [in](#) ENCAMINA

# El Cómic Como Medio de Enseñanza de Conceptos de Programación en C# en Ámbitos Universitarios

Probablemente la pregunta más repetida por parte de la gente que conoció esta iniciativa al principio de su andadura docente fue ¿y por qué un comic? Lo cierto es que puede resultar chocante introducir algo que en principio parece alejado del mundo docente, y además en un ámbito universitario. No obstante, el uso de comics para explicar temas técnicos no es algo novedoso, sino que se usa con notable éxito en iniciativas como las que llevan a cabo Julia Evans con Wizard Zines (<https://wizardzines.com/>) y Amy Wibowo con una similar llamada BubbleSort Zines (<https://shop.bubblesort.io/>). Estas iniciativas no solo demuestran que su uso para fines docentes es viable, sino que además prueban que son un vehículo adecuado para explicar conceptos de elevada complejidad. Por otro lado, un comic permite una representación muy visual de la información a transmitir y, por tanto, supone un complemento a los tradicionales apuntes más basados en texto que se usan en las clases habitualmente.

Con anterioridad a esta iniciativa, en la asignatura probamos aproximaciones más “clásicas”, como la elaboración de FAQs de preguntas sobre diferentes aspectos de C# y el .Net Framework que los alumnos nos hacían, con una explicación detallada de los problemas y sus soluciones. No obstante, muchos de los alumnos no usaban este mecanismo todo lo que se esperaba, declarando que hacerlo les llevaba demasiado tiempo. El motivo es que usarlo les resultaba bastante denso y difícil de manejar; aunque las explicaciones eran muy detalladas, muchas veces su objetivo era solucionar un problema muy específico de un concepto y no tener que volver a repasar una explicación completa de algo que, en gran parte, ya dominaban.

Otra de las aproximaciones que probamos para intentar contrarrestar problemas conceptuales es darles pruebas prácticas de todos los conceptos clave de la asignatura (es decir, código fuente C# funcional, mediante proyectos independientes de Visual Studio, con ejemplos individualizados de cada concepto). El objetivo era que las dudas se despejasen mediante la experimentación, usando programas muy pequeños que usen cada uno de los conceptos explicados. Con ellos los alumnos podrían probar, modificar, desarrollar variantes o experimentar estos conceptos de cualquier forma que necesitasen, con el objetivo de afianzarlos. El problema que esta iniciativa tiene es el mismo que el anterior: los alumnos no la usaban como pretendíamos por requerir más tiempo del esperado, y muchas

veces los problemas con el concepto eran significativos y el uso de código no lograba aclararlos. El alumno planteaba las mismas dudas, pero con el código suministrado como guion para hacer sus preguntas.

Por estos motivos, uno de los objetivos principales de los comics que desarrollamos es permitir que los conceptos se refuercen de forma rápida y muy “quirúrgica”, es decir, que una persona leyendo uno de estos comics sea capaz de comprender una sola idea (la detectada previamente como realmente problemática), transmitida en una serie de pocos pasos que guíen al lector en una única dirección que “ataque” el problema detectado. Obviamente, el diseño de un material así requiere para ser realmente efectivo un conocimiento muy elevado de la materia, los problemas que los alumnos experimentan con las diferentes partes de esta, y cuáles son los puntos en los que la gente suele fallar cuando se tiene que usar o aplicar un concepto. Parte de esta información solo puede ser realmente suministrada por los propios alumnos, como veremos posteriormente.

Si bien la labor de síntesis que puede hacer un profesor en este sentido es de importancia capital para asegurar lecturas rápidas, tiene que ser acompañada de un formato de comic acorde a la misma. Por ese motivo, hemos decidido usar el formato de fanzine, que ya por definición suele ser una obra corta. Otras ventajas que plantea el uso de fanzines es que son de producción muy rápida, con muy pocos medios técnicos, y por tanto son un medio ágil que puede permitir responder a las necesidades de los alumnos a medida que las vayan planteando.

Una vez descritas las bases de la idea y cuál es su motivación, queda por hablar de cómo se ha explotado la misma para lograr un beneficio en nuestra asignatura. El primer paso ha sido ir recopilando las dudas conceptuales más frecuentes que íbamos recibiendo y desarrollar materiales de refuerzo en forma de fanzines en función de estas. Sin embargo, en una segunda fase creímos más oportuno poner la iniciativa a servicio de los alumnos, para que la elaboración de nuevos fanzines fuera también bajo petición expresa de aquellos que manifestasen problemas específicos. Con todo ello, se ha compuesto un catálogo de fanzines que cubren un amplio rango de conceptos, desde básicos a más avanzados (ver Imagen 1).

Y lo cierto es que, debido a ese amplio rango de conceptos problemáticos para los que se ha solicitado o identificado

la necesidad de refuerzo, esta iniciativa nos ha permitido descubrir la existencia de no pocos alumnos que poseían problemas de base importantes, con conceptos como excepciones, polimorfismo, enlace dinámico y asertos que no se habían entendido correctamente y necesitaban un refuerzo específico para poder poner al alumno “al día” de los conceptos requeridos por la asignatura. En ese sentido, los fanzines han hecho una labor de “nivelación” que de otra forma hubiera sido muy difícil de realizar.

Obviamente, la labor de refuerzo no ha quedado relegada a conceptos básicos, sino que a medida que el curso iba avanzando, y aparecían conceptos nuevos, las dudas sobre los mismos no tardaron en aparecer y se desarrollaron nuevos fanzines relacionados. Estos fanzines cubren conceptos más avanzados de programación orientada a objetos como el uso efectivo de genericidad, de abstracciones IEnumerable, conceptos específicos de C# (como el paso de parámetros por referencia y de salida), etc. También se han desarrollado algunos dedicados a conceptos de programación funcional como funciones lambda, generadores y sobre todo el uso de Linq. Finalmente, algunos más cubren la última parte de la asignatura, dedicada a la programación concurrente, con explicaciones relativas a mecanismos de interbloqueo como lock, threads y problemas de concurrencia típicos. La imagen 1 también muestra la división de todos los fanzines desarrollados divididos por bloques.

### Índice de Zines

© José Manuel Roldano López

PROGRAMACIÓN ORIENTADA A OBJETOS	PROGRAMACIÓN FUNCIONAL	PROGRAMACIÓN CONCURRENTE
Polimorfismo y enlace dinámico	Introducción a la programación funcional	Uso de lock
Equals	Usando funciones	Contextos en los que usar lock
Downcast y Upcast	Funciones lambda	Convirtiendo el código a paralelo (TPL/PLinq)
Paso de parámetros, ref y out	Linq	Cláusulas e hilos
Excepciones	Generadores	ReaderWriterLockSlim
Asertos	Currificación	Master-Worker
Genericidad	Operaciones lazy	Consumer-Producer
Genericidad en C# vs Java	“Ejecución” de cálculo lambda	<b>METAPROGRAMACIÓN &amp; OTROS TEMAS</b>
IEnumerable	Join (Linq)	Imitando a los lenguajes dinámicos
	GroupBy (Linq)	Roslyn Compiler-As-A-Service
		API de Voz

Imagen 1.- Conjunto clasificado de fanzines desarrollado.

Uno de los aspectos más curiosos que nos surgieron al final del curso es que unos pocos alumnos empezaron a hacer peticiones de fanzines para explicar conceptos que no tenían realmente relación con la asignatura, sino con lo que va más allá de lo que se explica en ella. El bloque de fanzines “Metaprogramación & Otros temas” de la imagen anterior recoge peticiones de información de temas avanzados como el API de Text-to-Speech, el uso de Roslyn para la ejecución dinámica de código y, en relación con esto, cómo pueden imitarse las características de los lenguajes dinámicos con las funciones disponibles en C#. En ese sentido, creemos que los fanzines han permitido también satisfacer las inquietudes de los alumnos que han querido ir “más allá” de los contenidos de la asignatura.

Los fanzines desarrollados tienen un aspecto visual coherente, con los mismos elementos gráficos, colores y disposición de elementos, para que se identifiquen con el curso y con el profesor que los propone, independientemente

del tema que traten. Las imágenes 2 y 3 muestran dos ejemplos de temas de dificultad diametralmente opuesta para comprobar que, aunque la dificultad de los contenidos sea muy diferente, la forma de reforzarlos es la misma.



Imagen 2.- Fanzine básico (Excepciones).

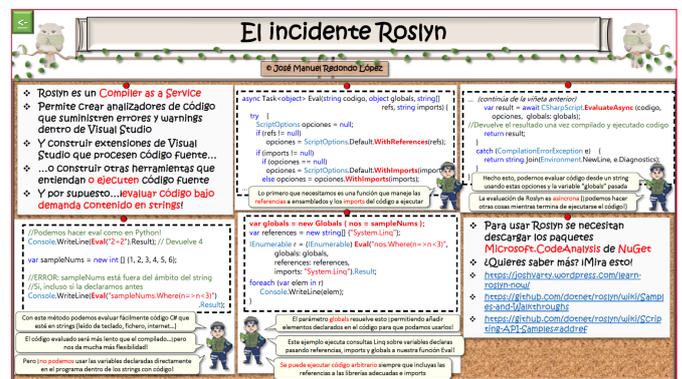


Imagen 3.- Fanzine avanzado (Roslyn y evaluación dinámica de código).

Algo que puede ayudar a ver cómo se pasa de una duda conceptual a un fanzine (en este caso el de la imagen 2), es describir como ocurrió un proceso real que derivó en uno de ellos. El fanzine de la imagen nº 2 empezó a gestarse cuando cuatro alumnos presentaron en el espacio de una semana una duda conceptual grave con el manejo de excepciones y precondiciones. Esta se manifestó con la solución a un problema de filtrar empleados de acuerdo con ciertos criterios, correspondiente a la segunda práctica del curso. Esto quiere decir que no se habían impartido todavía Linq, parámetros por omisión, ni otros elementos que simplificarían el código, que se incorporarían más tarde y con los que paulatinamente se mostrará cómo se podrá ir mejorando el mismo desde esta implementación básica. La solución planteada por estos alumnos fue esta:

```

static Empleado[] Filtrar(Empleado[] empleados, string nombre,
string apellidos,
string nifContiene) {
    Debug.Assert(empleados == null, "Error: el array de personas es nulo");

    List<Empleado> temp = new List<Empleado>();
    foreach (var empleado in empleados) {
        if (empleado.Nombre.Equals(nombre))
            temp.Add(empleado);
        else
            if (empleado.Apellidos.Equals(apellidos))
                temp.Add(empleado);
            else
                if (empleado.NIF.Contains(nifContiene))
                    temp.Add(empleado);
    }

    return temp.ToArray();
}
    
```

Este código presenta errores de concepto evidentes. Las precondiciones del método no solo se comprueban con un mecanismo erróneo para esta implementación (que los empleados sean null no es un error de programación en este escenario, no tiene sentido usar un aserto). Al usar asertos en lugar de excepciones, se evita poder manejar el error y tratar de corregirlo por parte del llamado. Además, solo se comprueba uno de los parámetros y no el resto, con lo cual ni siquiera se detectan todos los posibles problemas. Tras hablar con los alumnos, se vio clara la necesidad de desarrollar un fanzine que explique claramente a los alumnos el propósito de las precondiciones: cómo se manejan mediante excepciones para poder tratar los errores detectados, cómo comprobar varias (si las hubiera) antes de implementar el método en sí, y cómo se tratan los errores notificados. La imagen 2 muestra así el fanzine creado para explicar el concepto de precondición y su aplicación práctica que esos alumnos no dominaban.

Como ya se comentó, no todos los fanzines responden a problemas de concepto, sino que hacía el final del curso algunos se crearon motivados por preguntas avanzadas de alumnos. Por ejemplo, el fanzine de la imagen 3 responde a una pregunta de un alumno que deseaba evaluar expresiones en código C# que leía de un fichero de texto para un proyecto personal. El alumno planteaba el problema de convertir texto a código ejecutable en tiempo de ejecución, (es decir, hacer algo similar a la función eval de Python), pensando inicialmente que en un lenguaje en el que fundamentalmente predomina la comprobación estática de tipos (salvo excepciones, como las variables de tipo dynamic) no era posible. El fanzine le dio las herramientas para hacer lo que quería a través del uso del compilador Roslyn como un servicio usable desde cualquier programa. Posteriormente, y fruto de preguntas similares, se desarrolló un material no evaluable que muestra de manera más completa cómo implementar características típicas de lenguajes dinámicos en C# (<https://www.researchgate.net/publication/338584100> Características Fantásticas y Donde Encontrarlas La Pythonización del C).

Finalmente, la pregunta más importante que queda por responder es ¿realmente han funcionado como se esperaba? Lo cierto es que el feedback de los alumnos, tanto verbal como por escrito en las encuestas de calidad realizadas, ha sido muy positivo, y gracias a él podemos decir que realmente han ayudado a disminuir el tiempo que los alumnos tardan en entender los conceptos que les resultan problemáticos. El éxito de esta iniciativa ha impulsado dos proyectos paralelos:

- La extensión del nº de fanzines desarrollados durante el siguiente curso académico: 5 de los fanzines que se ven en la imagen 1 han sido desarrollados bajo petición expresa de los alumnos del presente curso académico, complementando los que ya existían del curso anterior.
- Trasladar la idea a otras asignaturas de otros ámbitos: está en desarrollo la aplicación de esta idea a asignatu-

ras de administración de servidores y de seguridad. La imagen 4 muestra un piloto de cómo se aplicaría esta idea para reforzar conceptos del software de contenedores Docker.

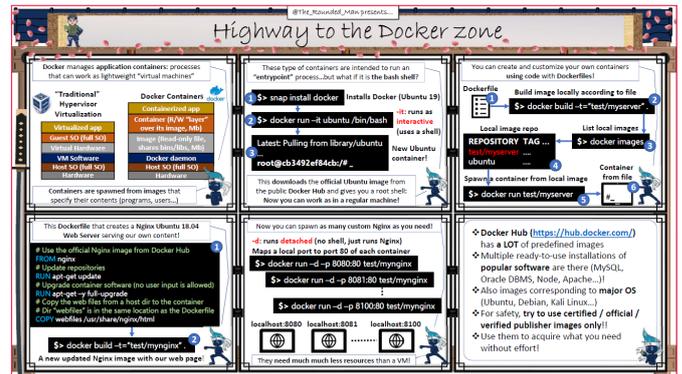


Imagen 4.- Fanzine aplicado a otra asignatura (Docker).

Finalmente, creemos también que los fanzines desarrollados pueden tener interés más allá de nuestro curso o de las aulas en sí, ya que pueden servir de ayuda a profesionales que tengan que repasar o reforzar conceptos que hayan olvidado por desuso y que, debido a comenzar en un nuevo proyecto o empresa, se vean forzados a recordar para desarrollar efectivamente su labor profesional. Por dicho motivo, todos los fanzines se han hecho públicos bajo licencia GNU en la siguiente dirección: [https://www.researchgate.net/publication/333104441\\_FanCines\\_Understand\\_C\\_Sharp\\_concepts\\_the\\_easy\\_way](https://www.researchgate.net/publication/333104441_FanCines_Understand_C_Sharp_concepts_the_easy_way) para su consulta y/o descarga. El piloto de su aplicación a otras asignaturas también está disponible aquí: [https://www.researchgate.net/publication/335023411\\_Admin-zines\\_Understand\\_Infrastructure\\_Administration\\_concepts\\_the\\_easy\\_way](https://www.researchgate.net/publication/335023411_Admin-zines_Understand_Infrastructure_Administration_concepts_the_easy_way). Un número de personas que trabajan en la empresa privada y exalumnos de la asignatura han transmitido que estos mecanismos de refuerzo también les han sido útiles.

## Conclusión

Tras la aplicación de esta iniciativa durante dos cursos académicos, creemos que este mecanismo de refuerzo de contenidos problemáticos de forma muy rápida y visual puesto al servicio de los alumnos cumple con los objetivos planteados y funciona como un medio para disminuir las dudas conceptuales que los alumnos plantean durante el curso. Además, creemos que este mecanismo es adecuado para problemas de todo nivel de dificultad y de otros ámbitos no relacionados con la programación. No obstante, para ello es necesario un conocimiento profundo de la materia, y hacer un esfuerzo de síntesis elevado de los conceptos que permitan el desarrollo de fanzines de acuerdo con la filosofía con la que se han diseñado.

**JOSÉ MANUEL REDONDO LÓPEZ**

Doctor en Ingeniería Informática. Universidad de Oviedo  
[redondojose@uniovi.es](mailto:redondojose@uniovi.es)  
 @The\_Rounded\_Man  
[https://www.researchgate.net/profile/Jose\\_Redondo8](https://www.researchgate.net/profile/Jose_Redondo8)

## Uso de Azure Monitor

En este artículo vamos a repasar un par de características que se suelen pasar por alto en las suscripciones de Azure, estas dos características están incluidas dentro del Azure Monitor, y pese a que están actualmente en Preview, nos serán muy útiles para conocer de primera mano toda la información referente a nuestra suscripción.

### Azure Monitor – Networks

En todos los proyectos, como norma general y buena práctica, deberíamos de implementar dashboards de monitorización, con ello, lo que nos permite es hacer monitorización a través de todas las suscripciones del estado de salud de nuestras redes. Y desde este punto de partida, empezar a navegar por cada parte de nuestro NW e investigar si hay algún fallo, exceso de tráfico o simplemente, observar el tráfico y si lo debemos de mejorar de alguna forma.

**"pese a que están actualmente en Preview, nos serán muy útiles para conocer de primera mano toda la información referente a nuestra suscripción"**

Algo que suelo utilizar mucho para ello, es el apartado de Networks incluido dentro de Azure Monitor, por ejemplo, podemos ver qué información nos da de primeras:

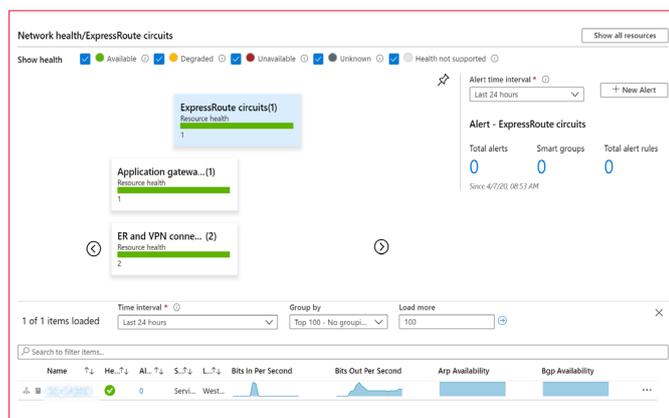


Imagen 1.- Sección Networks en Azure Monitor.

Y si hacemos clic en cada una de las gráficas, podemos observar sus datos sin problema

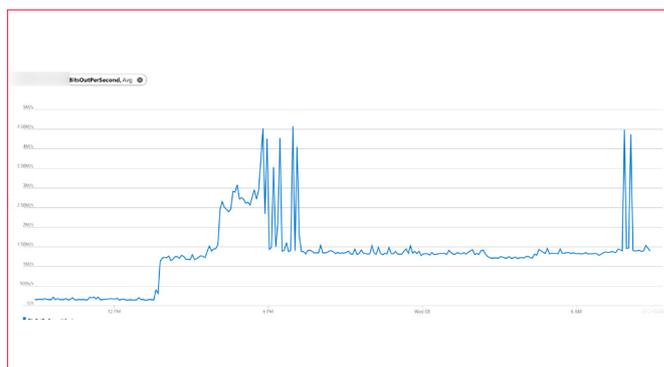


Imagen 2.- Detalle de una red.

De la misma forma, podemos ver el estado de cada uno de los recursos que tenemos implementados en el portal, o bien de un solo vistazo ver si tenemos alguna alerta asociada al recurso en cuestión:

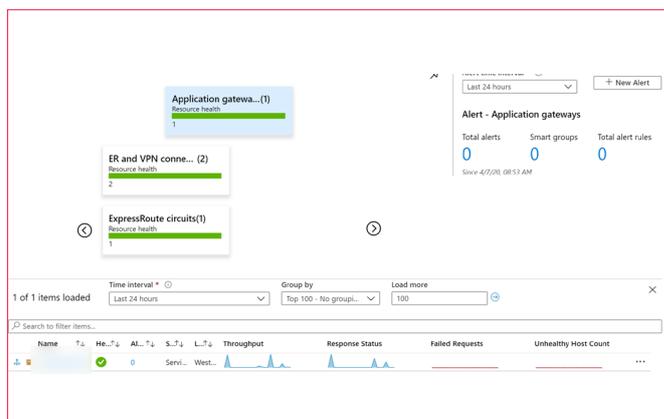


Imagen 3.- Visualización del estado de los recursos implementados.

**"Azure Monitor para poder ver la información de las storage accounts y la información que se puede extraer de sus métricas"**

### Azure Monitor - Storage Accounts

Por otro lado, pero no menos importante, es la ventaja que nos ofrece Azure Monitor para poder ver la información de las storage accounts y la información que se puede extraer de sus métricas. Lo primero que podemos observar, es que nos permite ver un overview:

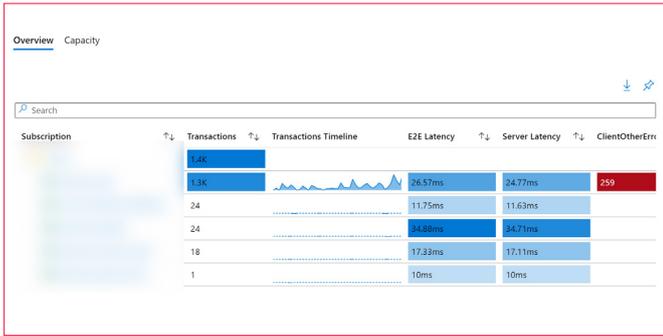


Imagen 4.- Visualización de Storage Accounts en Azure Monitor.

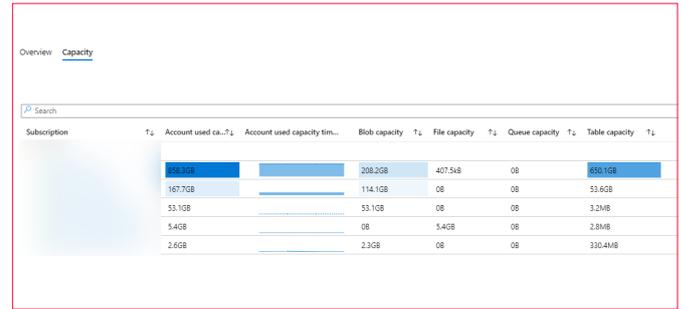


Imagen 5.- Visualización de la capacidad de un Storage.

Donde nos permite ver de un solo vistazo información esencial de las storage account, como puede ser la latencia, transacciones. De manera que, en función de esta info, tomemos decisiones al respecto. Y la capacidad de estas:

Donde obviamente, podemos ver la capacidad y sobre todo por cada tipo de almacenamiento, cuanto estamos ocupando. Algo muy útil, ya que estamos pagando por almacenamiento ocupado.

Con estas dos pequeñas gemas que tenemos en el portal, podremos tener más información de lo que está pasando en nuestras suscripciones.

**"Con estas dos pequeñas gemas que tenemos en el portal, podremos tener más información de lo que está pasando en nuestras suscripciones"**

**ALBERTO ANDRÉS**  
 Cloud Solutions Architect  
 @albandrod



## El Learning Pathways de Microsoft 365

El Microsoft 365 Learning Pathways es un Site Collection personalizada para SharePoint, creada por Microsoft, que reúne una gran cantidad de contenido educativo sobre todos los componentes de Office 365. La información es creada, seleccionada y mantenida (renovaciones cada tres meses) por Microsoft mismo. Todo el sistema es gratis y se puede instalar en cualquier tenant de SharePoint Online para crear un portal de aprendizaje sobre los componentes de Office 365.

El Microsoft 365 Learning Pathways consiste en una Colección de Sitios de SharePoint Online del tipo "Communication Site" y una WebPart especialmente creada para el sistema por Microsoft. La información en la WebPart está formada principalmente de vínculos a información creada por Microsoft que se pueden ver directamente desde el sitio. El control sobre la Colección de Sitios permanece totalmente en manos de los administradores del tenant, no de Microsoft, por lo que también es posible agregar contenido propio y aplicar cualquier tipo de modificación que la empresa requiera (logos, colores, layouts, ayuda, etc.).

El Learning Pathways es gratis y Open Source, y por lo tanto no está cubierto por ningún programa de garantía, mantenimiento o soporte de Microsoft. El código fuente se puede encontrar en GitHub (<https://github.com/pnp/custom-learning-office-365>).



### Instalación

Hay dos formas para instalar el Learning Pathways: por medio del sistema de provisionamiento de Office, o descargando la WebPart e instalándola manualmente. La forma recomendada es la primera por ser la más fácil, segura y rápida. Microsoft proporciona instrucciones completas si desea seguir el segundo camino ([https://docs.microsoft.com/en-us/office365/customlearning/custom\\_manualsetup](https://docs.microsoft.com/en-us/office365/customlearning/custom_manualsetup)).

[en-us/office365/customlearning/custom\\_manualsetup](https://docs.microsoft.com/en-us/office365/customlearning/custom_manualsetup)).

El sistema de provisionamiento de Office se basa en PnP (Office Patterns and Practices) y, para utilizarlo, se necesita cumplir tres requisitos:

- La cuenta que va a instalar el Learning Pathways debe ser un administrador del tenant de Office 365.
- El Catálogo de Aplicaciones del Tenant debe estar disponible (desde el centro de Administración de SharePoint, vaya a "More Features" - "Apps" (Open) - "App Catalog" - "Create a new app catalog site"), y haber sido creado por lo menos dos horas antes de la instalación del Learning Pathways.
- La cuenta de instalación debe tener derechos de propietario (Admin) del Catálogo de Aplicaciones del Tenant.

Para iniciar la instalación, abra el URL <https://provisioning.sharepointpnp.com/details/3df8bd55-b872-4c9d-88e3-6b2f05344239> en un browser y utilice el botón "Add to your tenant". Si la cuenta no tiene una sesión abierta en el navegador, la ventana de login de Office aparece para insertar las credenciales.

El servicio de provisionamiento requiere permisos para instalar la aplicación en el catálogo de aplicaciones y crear la Colección de Sitio. Los permisos se usan explícita y únicamente para la instalación de la solución y tienen que ser aceptados obligatoriamente para poder continuar con la instalación.

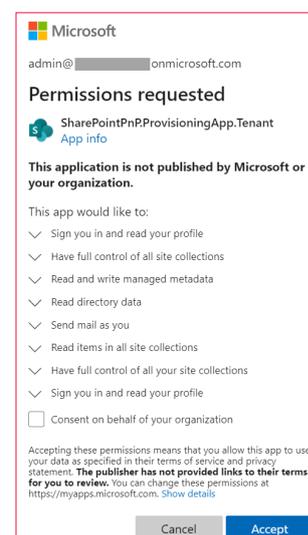


Imagen 1.- Permisos a aceptar para instalar Microsoft Learning Pathways.

En la siguiente página de la instalación se puede configurar la dirección de email a utilizar cuando el sitio esté listo, y el URL del sitio para el Learning Pathways.

**Great choice!**  
We just need a bit of information before we start provisioning  
The following settings are used for your example site. You can keep them as is, or change them to suit your needs.

**Email**  
Provide an email address to get notified when provisioning is complete  
admin@...onmicrosoft.com

**Microsoft 365 learning pathways site URL**  
URL for the Microsoft 365 learning pathways site (default is /sites/O365CL)  
https://...sharepoint.com  
/sites/  
O365CL

Cancel Provision

Imagen 2.- Opciones de configuración.

**"La información en la WebPart está formada principalmente de vínculos a información creada por Microsoft"**

Haga clic sobre el botón de "Provision" para iniciar el proceso (una ventana de "Confirmación" abre y debe ser aceptada). El sitio demora aproximadamente 15 minutos para ser creado y configurado, y un email será enviado a la cuenta configurada cuando esté finalizado (Nota: el email no siempre llega correctamente). El administrador debe ir al sitio para comprobar que todo funciona sin problemas y abrir la página "../SitePages/CustomLearningAdmin.aspx", desde donde se pueden configurar los temas a mostrar.

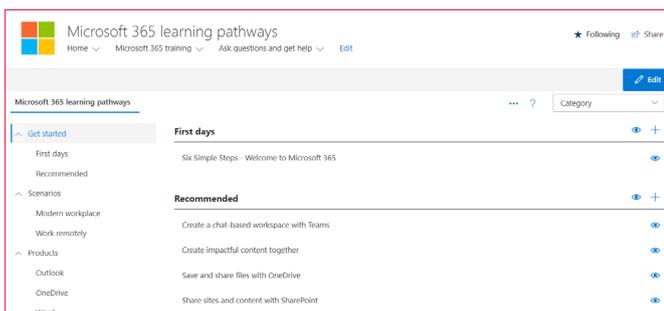


Imagen 3.- Revisión de la instalación de Microsoft 365 learning pathways.

El Administrator también debe configurar directamente los usuarios que van a ser los dueños del sitio (la asignación de derechos se hace como en cualquier otro sitio de SharePoint).

## El Learning Pathways

El sitio de SharePoint es un sitio de comunicación normal, usando la plantilla de "Topics", con una serie de páginas creadas por defecto para el contenido. Cada página contiene la WebPart "Microsoft 365 Learning Pathways" para mostrar el contenido de cada uno de los temas tratados.

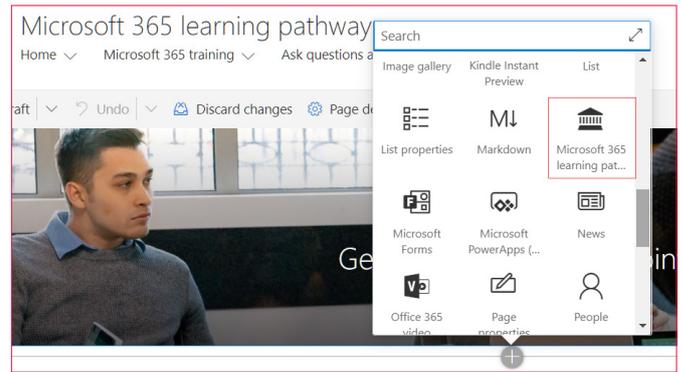


Imagen 4.- WebParts de Microsoft 365 Learning Pathways.

Nuevas paginas pueden ser creadas si es necesario, o las existentes se pueden modificar siguiendo las líneas indicadas por Microsoft para personalizar a SharePoint.

Como todo sitio Moderno de SharePoint del tipo Communication, el sitio del Learning Pathways es multilinguaje, pero no está activado por defecto. Para activarlo, haga clic sobre el botón de configuración del sitio en la página principal (el icono de engranaje), haga clic sobre "Site Information" y luego use el vínculo de "View all site settings". En la sección de "Site administration" use el vínculo de "Language settings" y en la página que abre, active el sistema de multilinguaje y configure los idiomas a utilizar. Cada idioma permite también asignarle uno o varios traductores, o grupo de traductores. Después de guardar los cambios, abra cualquier página, use el botón de Editar, y luego el botón de "Translation" en el menú horizontal. En la ventana de configuración se puede indicar que idiomas van a estar activados en la página.

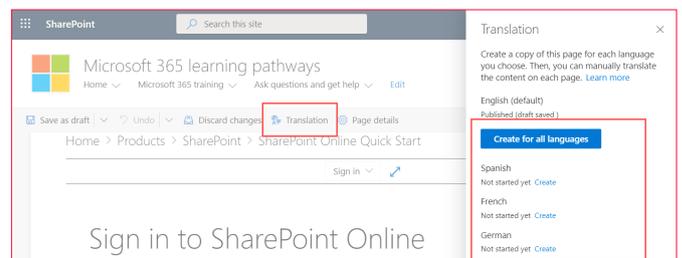


Imagen 5.- Configuración del multidioma.

Los traductores (si han sido configurados) reciben un email indicando que paginas tienen que ser traducidas. Después de que las paginas terminadas han sido republicadas, el botón de idiomas de la página permite que los usuarios vean la información en su propio lenguaje.

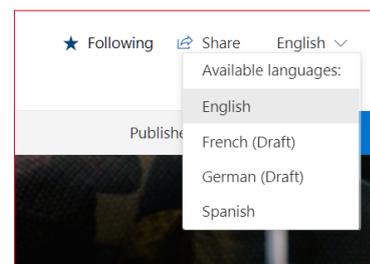


Imagen 6.- Idiomas configurados.

Los videos en el sitio no tienen soporte multidioma, sino que muestran subtítulos en los idiomas elegidos. El botón



de "Closed captions" en cada video permite seleccionar el idioma de los subtítulos, su tipo de letra, etc.:

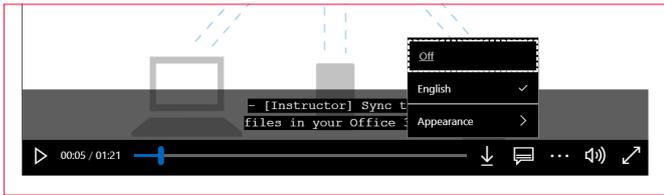


Imagen 7.- Experiencia de reproducción de un video.

En el momento de escribir este artículo la mayoría de los videos no han sido traducidos todavía por Microsoft, pero dentro de poco tiempo todos los idiomas ofrecidos estarán disponible (chino simplificado, francés, alemán, italiano, japones, portugués, ruso y español).

## Página de Ayuda y preguntas

El Learning Pathways dispone por defecto de una página para ayuda y preguntas de los usuarios ("As questions and get help"). La página dispone de cuatro vínculos para que los usuarios puedan acceder a ayuda y resolver dudas sobre Office 365. Tres de los vínculos no están configurados por defecto ("Ask questions", "Contact Helpdesk" y "Get more training") y deben ser modificados antes de entrar en producción para redirigir a los usuarios a los sitios específicos dentro de la empresa. El cuarto vinculo ("Join the Community") está configurada por defecto redirigiendo al sitio de comunidad de Office 365 de Microsoft (<https://aka.ms/O365Champions>). Para configurar los URLs, abra la pá-

gina en forma de edición, seleccione el vínculo y cambie su propiedad "Link":

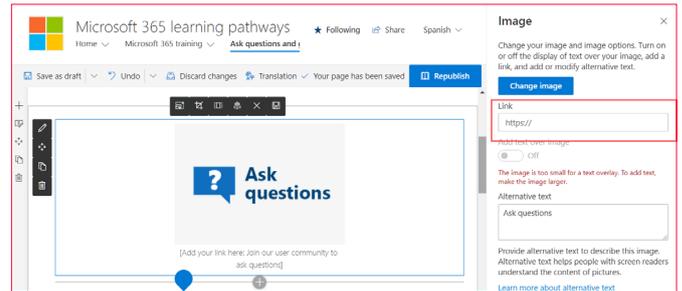


Imagen 8.- Configuración de página de Ayuda y preguntas.

## Conclusiones

El Learning Pathways es un sistema creado y mantenido por Microsoft que ofrece gratuitamente multitud de recursos informativos sobre los componentes de Office 365. El sistema puede ser utilizado para entrenar a los usuarios de Office 365, ayudarles si tienen algún tipo de problema y guiarlos en su utilización. El Learning Pathways se instala como una Colección de Sitios en cualquier tenant de Office 365, y puede ser personalizado según los requerimientos de la empresa si es necesario.

**GUSTAVO VELEZ**  
**MVP Office Apps & Services**  
[gustavo@gavd.net](mailto:gustavo@gavd.net)



55

## Entrevista Intelequia

Intelequia Software Solutions está especializada en tecnologías de Microsoft y en sus productos de Nube Inteligente.

Desde nuestros comienzos nos hemos especializado en la adopción y desarrollo de soluciones en la nube con alto foco en Microsoft Azure, aunque desde hace algún tiempo también trabajamos con AWS y Google Cloud. Actualmente nuestro negocio se centra en todo lo relacionado a la transformación digital de empresas públicas y privadas, ofreciendo soluciones de productividad y colaboración basadas en Microsoft 365 además de servicios de aplicaciones e infraestructura sobre Azure: diseño de arquitecturas, desarrollo, implementación y servicios administrados. Otras áreas en las que trabajamos son los datos e inteli-



gencia artificial basadas en soluciones de Microsoft, soluciones de ciberseguridad y formación oficial de Microsoft (MOC).

Hace diez años que venimos ofreciendo nuestras soluciones y servicios profesionales de tecnología a empresas nacionales e internacionales, participando en el crecimiento tecnológico de muchas organizaciones y recibido reconocimiento a nuestra empresa y a nuestro equipo.

### ¿Por qué y cómo empezó en el mundo de la tecnología?

Nacimos a finales de 2010 como marca comercial y nos consolidamos como sociedad limitada en 2012. Partiendo de la crisis de 2008 vimos una oportunidad de negocio con el modelo de la nube que convertimos en nuestra especialización. Por aquel entonces era muy difícil vender en España y más aún en Canarias, con lo que nuestros primeros clientes fueron de EE. UU., donde poco a poco fuimos creciendo y consolidándonos en el mercado. Hoy en día se nos conoce a nivel nacional por nuestro buen hacer y el alto nivel de cualificación de nuestro equipo.

### ¿Cuáles son las principales actividades tecnológicas hoy en día?

Nuestra principal actividad son los servicios de nube inteligente, así como servicios administrados, datos e inteligencia artificial. Nuestro objetivo es acercar la tecnología a todas las organizaciones y personas, para que tengan la transformación digital propia de su sector, acorde a sus necesidades y posibilidades, de forma segura y coherente.

### ¿Cuáles son las principales actividades NO tecnológicas hoy en día?

Queremos ser una empresa sostenible y con responsabilidad social, poniendo a las personas en el centro, dándoles herramientas y seguridad para asumir sus responsabilidades y fomentando una cultura en la que busquemos crecer haciendo crecer. Evitamos las jerarquías dentro de lo posible y apostamos por un modelo de trabajo basado en la colaboración, que permita que el conocimiento llegue a todas y cada una de las personas para que puedan desarrollar su ingenio y creatividad.

Solemos tener una vez al mes una reunión de toda la empresa, para mantener el contacto y que el conocimiento fluya de un departamento a otro. Ese es el día de Pizza en la empresa, comemos todos juntos y de forma distendida. Con la llegada del confinamiento hemos convertido estas reuniones en semanales, donde los viernes nos reunimos para charlar, mostrar cómo van creciendo nuestras mascotas o incluso jugar a algún juego online. Una o dos veces al año hacemos alguna actividad lúdica, como ir al karting, hacer una cata de vinos, etc.



## ¿Cuáles son las actividades que realiza en la comunidad técnica?

Tenemos la suerte de contar con un equipo muy activo en las comunidades técnicas, participando en eventos de todos los tamaños. Nuestro CTO y cofundador, participa organizando charlas en algunas comunidades locales como TenerifeDev o más grandes como el Global Azure junto con otros compañeros que también han sido reconocidos como MVPs en diversas áreas. En Intelequia apoyamos este tipo de actividades y hoy en día más del 75% de la plantilla ha participado en algún evento pasado. Visitando nuestro blog podrán ver cómo tanto el personal técnico como el de ventas, comparte soluciones y muestras de código.

Creemos y apoyamos firmemente el modelo Open Source, ya que tenemos un largo pasado desde nuestro comienzo en EE. UU., y así lo intentamos transmitir a nuestros clientes y colaboradores.

## ¿Cuál es la visión de futuro en la tecnología de acá a los próximos años?

En Intelequia vemos la tecnología como un medio. En ese sentido quizás la pregunta debería ser “¿Cómo la tecnología acelerará el cambio?” Creemos que habrá muchos cambios y avances, con muchas posibilidades en áreas tan importantes como la medicina, el cuidado de nuestros mayores, la educación y el turismo, gracias a la explotación de datos y la inteligencia artificial.

La tecnología está permitiendo dar continuidad a los negocios en tiempos de incertidumbre. Debido a los cambios que estamos sufriendo en estos momentos el factor de aceleración ha pasado de ser de cinco a uno: si algo calculábamos que tardaría en llegar unos cinco años, en uno o menos ya estará implantado. Esto nos llevará a que tendremos que “reimaginar” continuamente lo que vendrá luego, con lo que prevemos un futuro de cambio continuo al que no será fácil adaptarse. Probablemente se pase a hablar de analfabetos digitales a torpes holográficos sin darnos ni cuenta.

Sin embargo, ese motor también facilitará un crecimiento más rápido permitiendo a todo el ecosistema empresarial ser más eficaz. La tecnología también nos cambiará como individuos, empoderándonos y permitiéndonos abarcar más conocimiento y acceder a más contenidos.

# CosmosDB Autopilot

Cuando trabajamos con CosmosDB, una de las cosas que más cuesta ajustar es la cantidad de throughput que necesitamos. En muchas ocasiones te ves obligado a configurar las Requests Units per second (RU/s) por encima de lo que realmente necesitamos porque durante el 5% del tiempo necesitamos tener ese número de RU/s para dar servicio y el 95% de tiempo restante no lo necesitamos. Lógicamente esto hace que el coste de CosmosDB sea más alto de lo necesario por un 5%.

*"En muchas ocasiones te ves obligado a configurar las Requests Units per second (RU/s) por encima de lo que realmente necesitamos"*

## CosmosDB Autopilot

Para solucionar esta problemática, y no tener que hacer workarounds, Microsoft anunció a finales del 2019 una nueva feature en CosmosDB llamada Autopilot. Autopilot escalará automáticamente de forma inmediata sin que nuestra aplicación se vea afectada y sin ningún tipo de interacción.

### ¿Como funciona?

Autopilot funciona con niveles. Se establece un nivel máximo, a nivel de container o base de datos el cuál no se quiere superar y Autopilot automáticamente escalará entre el 10%, valor mínimo asignado a CosmosDB con Autopilot, al valor máximo asignado. De momento hay algunos límites al seleccionar esta opción que son los siguientes:

- 4.000 RU/s – 50 Gb de Storage.
- 20.000 RU/s – 200 Gb de Storage.
- 100.00 RU/s – 1 TB de Storage.
- 500.00 RU/s – 5 TB de Storage.

### Ventajas

Cuando usamos esta feature, una de las cosas que nos facilita es el no tener que gestionar la capacidad de nuestro CosmosDB de forma manual o tener aplicación que gestionen las RU/s. Aunque parezca mentira, una de las ventajas de Autopilot es su coste, ya que no pagas por las máximas

RU/s seleccionadas sino solo por los recursos que la carga de trabajo necesite por hora, siempre teniendo en mente que el mínimo de carga es el 10% del máximo seleccionado.

Cuando un servicio escala automáticamente una de las características que necesitamos es que no haya caída del servicio cuando escala, otra de las ventajas de Autopilot es que escala sin problemas las RU/s que necesita sin que haya caída del servicio siendo totalmente transparente.

Con Autopilot se mantienen todas las características que ya trae CosmosDB por defecto.

## Casos de uso

Cuando la carga que recibe tu CosmosDB es impredecible y muy variada, Autopilot te garantiza que dispondrá de los suficientes recursos para satisfacer la demanda de carga que se requiere. Otro caso de uso es cuando se pone una nueva aplicación en marcha, activando Autopilot puedes ver la carga real de tu CosmosDB y ajustar las RU/s lo que realmente necesitas de una forma mucho más precisa.

## Habilitar Autopilot

Cuando creas un nuevo container en CosmosDB tienes la posibilidad de seleccionar Autopilot:

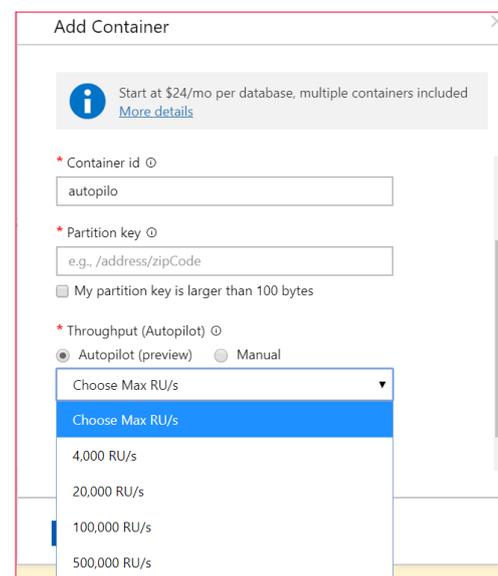


Imagen 1.- Activar Autopilot para CosmosDB.

*"Cuando la carga que recibe tu CosmosDB es impredecible y muy variada, Autopilot te garantiza que dispondrá de los suficientes recursos para satisfacer la demanda"*

## Cosas a tener en cuenta

Al ser un servicio en preview tiene ciertas restricciones:

- Solo se puede habilitar vía portal, no se puede hacer todavía por API y por tanto no se puede añadir dentro de los pipelines de creación de infraestructura.
- Si seleccionas RU/s fijas, no puedes después cambiarlo a Autopilot, para hacerlo debes crear un nuevo container o BBDD y hacer una migración de datos. En cambio, si tienes RU/s fijas si puedes cambiarlo a Autopilot.
- La vinculación con storage, cada opción de Autopilot

que existe va enlazada a un límite de storage, cuando pasas ese storage debes seleccionar la siguiente franja pagando más sin la necesidad quizás de tanto storage.

## Conclusiones

Autopilot es una gran feature que a los que usamos CosmosDB como almacenamiento de datos nos va a simplificar mucho la vida y sobre todo la disponibilidad de nuestra BBDD. Pero aún está en preview y con algunas limitaciones importantes, por lo que habrá que esperar a la versión definitiva para poder explotar esta gran feature.

---

**ROBERT BERMEJO**

**Chapter Backend Lead in SCRUM – Lidl Digital Hub**

**Microsoft Azure MVP**

*bermejblasco@live.com*

*@robertbermejo*

*www.robertbermejo.com*



## Alberto Diaz

Alberto Diaz cuenta con más de 15 años de experiencia en la Industria IT, todos ellos trabajando con tecnologías Microsoft. Actualmente, es Chief Technology Innovation Officer en ENCAMINA, liderando el desarrollo de software con tecnología Microsoft, y miembro del equipo de Dirección.

Desde 2011 ha sido nombrado Microsoft MVP, reconocimiento que ha renovado por séptimo año consecutivo. Se define como un geek, amante de los smartphones y desarrollador. Fundador de TenerifeDev ([www.tenerifedev.com](http://www.tenerifedev.com)), un grupo de usuarios de .NET en Tenerife, y coordinador de SUGES (Grupo de Usuarios de SharePoint de España, [www.suges.es](http://www.suges.es))

Email: [adiazcan@hotmail.com](mailto:adiazcan@hotmail.com)

Twitter: [@adiazcan](https://twitter.com/adiazcan)



## Fabián Imaz

Fabián Imaz, MVP de SharePoint Server trabaja en el mundo del desarrollo de software desde hace más de 10 años, teniendo la suerte de trabajar en distintas arquitecturas y tecnologías Microsoft. Pertenece a la firma Siderys, <http://www.siderys.com> empresa de desarrollo de Software especializada en SharePoint 2007/2010/2013 y en desarrollo de soluciones inteligentes.

Desde los comienzos Fabián ha trabajado en distintas comunidades donde organiza y promueve eventos locales para la difusión de tecnología dentro de los miembros de las mismas. Es director de la carrera SharePoint 2010 y SharePoint 2013 en Microsoft Virtual Academy, <http://www.mslatam.com/latam/technet/mva2/Home.aspx> y cuenta con un sitio en CodePlex con varios desarrollos <http://siderys.codeplex.com>.

Sitio Web: <http://www.siderys.com>

Email: [fabiani@siderys.com.uy](mailto:fabiani@siderys.com.uy)

Blogs: <http://blog.siderys.com>

Twitter: [@fabianimaz](https://twitter.com/fabianimaz)



## Gustavo Velez

Gustavo Velez es Ingeniero Mecánico y Electrónico; trabaja en la arquitectura, diseño e implementación de sistemas de IT basados en tecnologías de Microsoft, especialmente SharePoint, Office 365 y Azure.

Propietario del sitio especializado en información sobre SharePoint en español <http://www.gavd.net>, autor de ocho libros sobre SharePoint y sus tecnologías y numerosos artículos y conferencias sobre el tema.

Sitio Web: <http://www.gavd.net>

Email: [gustavo@gavd.net](mailto:gustavo@gavd.net)

Blogs: <http://geeks.ms/blogs/gvelez/>



## Juan Carlos González Martín

Ingeniero de Telecomunicaciones por la Universidad de Valladolid y Diplomado en Ciencias Empresariales por la Universidad Oberta de Catalunya (UOC). Cuenta con más de 14 años de experiencia en tecnologías y plataformas de Microsoft diversas (SQL Server, Visual Studio, .NET Framework, etc.), aunque su trabajo diario gira en torno a las plataformas SharePoint & Office 365. Juan Carlos es MVP de Office Apps & Services y co-fundador del Grupo de Usuarios de SharePoint de España (SUGES, [www.suges.es](http://www.suges.es)), del Grupo de Usuarios de Cloud Computing de España (CLOUDES) y de la Comunidad de Office 365. Hasta la fecha, ha publicado 11 libros sobre SharePoint & Office 365, así como varios artículos en castellano y en inglés sobre ambas plataformas.

Email: [jcgonzalezmartin1978@hotmail.com](mailto:jcgonzalezmartin1978@hotmail.com)

Blogs: <http://geeks.ms/blogs/jcgonzalez> &

<http://jcgonzalezmartin.wordpress.com/>





## Santiago Porras

Innovation Team Leader en ENCAMINA, lidera el desarrollo de productos mediante tecnologías Microsoft. Se declara un apasionado de la tecnología, destacando el desarrollo para dispositivos móviles y web, donde ya cuenta con 16 años de experiencia.

Microsoft MVP in Developer Technologies, colabora con las comunidades de desarrolladores desde su blog personal <http://blog.santiagoporras.com> y ocasionalmente en **CompartiMOSS.com**. Además, es uno de los coordinadores de TenerifeDev, grupo de usuarios de .NET en Tenerife (<http://www.tenerifedev.com>)

Sitio Web: <http://www.santiagoporras.com>

Email: [santiagoporras@outlook.com](mailto:santiagoporras@outlook.com)

Blogs: <http://blog.santiagoporras.com>

Twitter: [@saintwukong](https://twitter.com/saintwukong)

# ¿Desea colaborar con CompartiMOSS?



La subsistencia del magazine depende de los aportes en contenido de todos. Por ser una revista dedicada a información sobre tecnologías de Microsoft en español, todo el contenido deberá ser directamente relacionado con Microsoft y escrito en castellano. No hay limitaciones sobre el tipo de artículo o contenido, lo mismo que sobre el tipo de tecnología. Si desea publicar algo, por favor, utilice uno de los siguientes formatos:

- Artículos de fondo: tratan sobre un tema en profundidad. Normalmente entre 2000 y 3000 palabras y alrededor de 4 o 5 figuras. El tema puede ser puramente técnico, tanto de programación como sobre infraestructura, o sobre implementación o utilización.
- Artículos cortos: Artículos cortos: Máximo 1000 palabras y 1 o 2 figuras. Describen rápidamente una aplicación especial de alguna tecnología de Microsoft, o explica algún punto poco conocido o tratado. Experiencias de aplicación en empresas o instituciones puede ser un tipo de artículo ideal en esta categoría.
- Ideas, tips y trucos: Algunos cientos de palabras máximo. Experiencias sobre la utilización de tecnologías de Microsoft, problemas encontrados y como solucionarlos, ideas y trucos de utilización, etc. Los formatos son para darle una idea sobre cómo organizar su información, y son una manera para que los editores le den forma al magazine, pero no son obligatorios. Los artículos deben ser enviados en formato Word (.doc o .docx) con el nombre del autor y del artículo.

Si desea escribir un artículo de fondo o corto, preferiblemente envíe una proposición antes de escribirlo, indicando el tema, aproximada longitud y número de figuras. De esta manera evitaremos temas repetidos y permitirá planear el contenido de una forma efectiva.

Envíe sus proposiciones, artículos, ideas y comentarios a la siguiente dirección:

[revista@compartimoss.com](mailto:revista@compartimoss.com)

[adiazcan@hotmail.com](mailto:adiazcan@hotmail.com)

[fabiani@siderys.com.uy](mailto:fabiani@siderys.com.uy)

[jcgonzalezmartin1978@hotmail.com](mailto:jcgonzalezmartin1978@hotmail.com)

[gustavo@gavd.net](mailto:gustavo@gavd.net)

