



REVISTA ESPECIALIZADA EN TECNOLOGÍAS MICROSOFT



Entrevista  
Juan Camilo  
Martínez Vélez

Manejando  
listas de datos  
en PowerApps

Microsoft Graph  
.Net SDK al des-  
cubierto

Conectar Power  
Virtual Agents  
con QnA Maker

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>

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

<http://blog.siderys.com>

<https://adiazacan.github.com>

**REDES SOCIALES****Facebook:**

<http://www.facebook.com/group.php?gid=128911147140492>

**LinkedIn:**

<http://www.linkedin.com/groups/CompartiMOSS-3776291>

**Twitter:**

@CompartiMOSScom

# Contenido

**03**

Editorial

**07**

De Office 365 a Employee Experience Platform, pasando por MyAnalytics y Workplace Analytics

**13**

Asp.Net core - como configurar tu aplicación para despliegue en una granja de servidores

**20**

Como versionar nuestro Producto de forma automática usando Git y Azure DevOps

**33**

Aplicación de Microsoft Lists en Microsoft 365 o Listas de SharePoint 2.0

**43**

Entrevista Krasis

**48**

Graph API y Power Automate, una combinación increíble

**04**

Diagnóstico y medición del rendimiento de sitios de SharePoint (Page Diagnostics for SharePoint)

**11**

Entrevista Juan Camilo Martínez Vélez

**17**

Conectar Power Virtual Agents con QnA Maker

**25**

Manejando listas de datos en PowerApps

**37**

Anatomía de una Adaptive Card

**45**

Microsoft Graph .Net SDK al descubierto

i

03

## Editorial

Aunque la pandemia del COVID-19 sigue marcando nuestro día a día y el de nuestras familias en muchos aspectos positivos o negativos, también lo está haciendo desde el punto de vista de la tecnología acelerando el ritmo de innovación y de nuevos productos que las grandes compañías tecnológicas están lanzando al mercado. En esta línea, está claro que Microsoft durante los últimos meses ha sacado a la luz novedades a un ritmo mucho más rápido que si el dichoso bicho no estuviese dando guerra. Una buena prueba de esto es Microsoft VIVA, la gran novedad del momento y que supondrá un antes y un después en lo que a la Experiencia de Empleado Conectado se refiere y que va a ser el nuevo caballo de batalla en el que todas las grandes van a estar y en la que como profesionales del sector tendremos mucho que aportar y que decir.

En CompartiMOSS la vida sigue igual y les presentamos un nuevo número que ha sido posible gracias al trabajo en equipo de Autores, Editorial y de nuestros lectores. Esperamos que este nuevo número, lleno de nuevo de artículos de calidad sea de su agrado y sobre todo deseamos que estén bien y que se cuiden.

**El Equipo Editorial de CompartiMOSS**

# i

---

## 04

# Diagnóstico y medición del rendimiento de sitios de SharePoint (Page Diagnostics for SharePoint)

A mediados del 2020 Microsoft presento la herramienta de “Diagnóstico de SharePoint” (Page Diagnostics for SharePoint), para ayudar a los propietarios y editores de sitios a optimizar las páginas de SharePoint en un tenant de Office 365. La herramienta extrae métricas de rendimiento, y basado en ellas, recomendaciones para la mejora del sitio. La herramienta mide la velocidad de carga de las páginas, lo que afecta la experiencia de visualización, y la capacidad de la página para interactuar con los espectadores y cumplir su propósito.

La herramienta es una extensión para Edge o navegadores basados en Chrome, y después de instalada presenta información que permite analizar lo que se debe mejorar en el sitio, como resolver problemas técnicos y mejorar la experiencia de los usuarios. El Page Diagnostics analiza sitios modernos y sitios clásicos de publicación de SharePoint Online (no se puede usar para SharePoint Server OnPrem ni para páginas del sistema, tales como “allitems.aspx”) contra una serie de criterios de rendimiento predefinidos.

## Instalación

Las siguientes instrucciones de instalación son para Edge, pero los pasos son similares para cualquier navegador basado en Chrome.

- Abra la página de Extensiones de Edge (botón de ellipse “...” en la esquina superior derecha y haga clic sobre “Extensions”). Haga clic sobre el botón “Get extensions for Microsoft Edge”
- En la casilla de búsqueda escriba “page diagnostics for sharepoint”, lo que muestra la extensión en la lista correspondiente. Haga clic sobre el botón de “Get”.

Search results for "page diagnostics for sharepoint" 1 extensions



**Page diagnostics for SharePoint**  
★★★★★ (6) | Microsoft Corporation

A tool to identify page performance problems in SharePoint...

[Get](#)

Didn't find what you're looking for? Check the spelling, try new keywords, or [search the web](#) for more results.

Imagen 1.- Extensión Page Diagnostics for SharePoint.

- Un mensaje de advertencia aparece indicando los derechos que la extensión necesita tener. Utilice el botón de “Add extension”:



### Add "Page diagnostics for SharePoint" to Microsoft Edge?

The extension can:

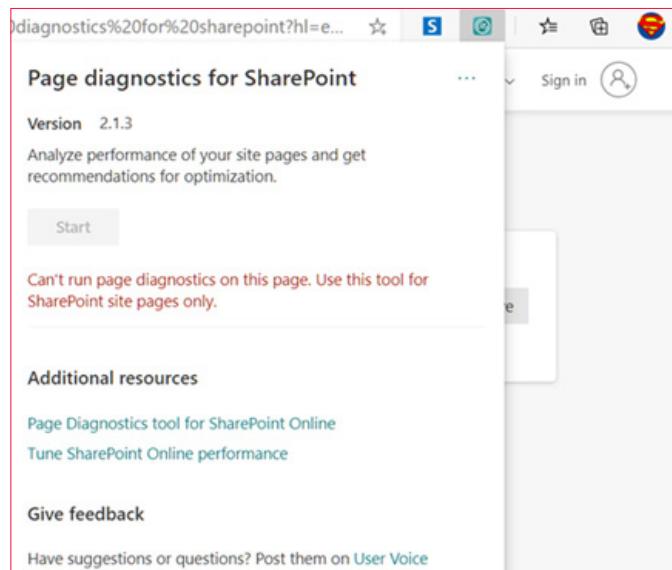
- Access the page debugger back end
- Read and change all your data on the websites you visit
- Manage your downloads

[Add extension](#)

[Cancel](#)

Imagen 2.- Añadiendo la extensión a Edge.

- Una vez instalada, la extensión aparece en la esquina superior derecha del navegador. Haciendo clic sobre ella, se puede ver su versión y alguna información adicional (también aparece un error que indica que la herramienta solo se puede utilizar en páginas de SharePoint)



Diagnostics%20for%20sharepoint?hl=e... S Q ... Sign in ...

**Page diagnostics for SharePoint**

Version 2.1.3

Analyze performance of your site pages and get recommendations for optimization.

[Start](#)

Can't run page diagnostics on this page. Use this tool for SharePoint site pages only.

**Additional resources**

[Page Diagnostics tool for SharePoint Online](#)  
[Tune SharePoint Online performance](#)

**Give feedback**

Have suggestions or questions? Post them on [User Voice](#)

Imagen 3.- Accediendo a la extensión en Edge.

Microsoft no lee datos ni contenido de página analizado por la herramienta, y no captura ninguna información personal, sitio web o información de descarga. La única información identificable que la herramienta registra en Microsoft es el nombre del Tenant de Office 365, los informes de criterios que han fallado y la fecha y hora en que se ejecutó la herramienta. Microsoft usa esta información para comprender el uso de portales de SharePoint en general y los problemas de rendimiento comunes con el propósito de mejorar el servicio en el futuro.

## Utilización

Abra una página de SharePoint Online, abra la ventana de la extensión y use el botón de “Start”. después de unos cuantos segundos aparecen los resultados del análisis en la ventana de la extensión en tres regiones diferentes: información general, diagnóstico y seguimiento de componentes. La opción de “Enable exporting to HTTP Archive (HAR)” permite utilizar el formato de archivo HTTP, o HAR, que es un formato JSON especializado para registrar la interacción de un explorador web con un sitio; normalmente la información es exportada en JSON.

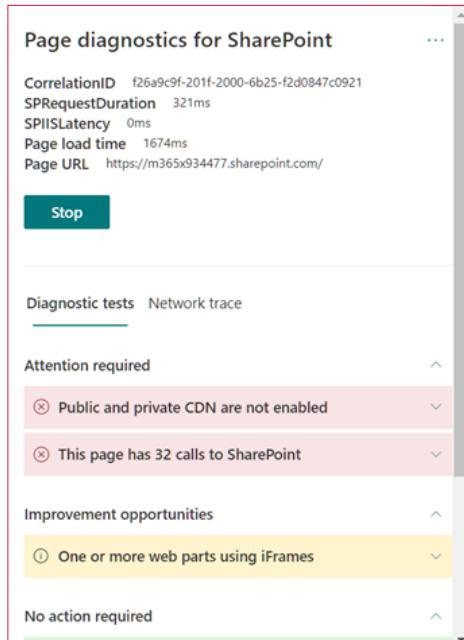


Imagen 4.- Resultados del Análisis de diagnóstico realizado.

- Información general:

- CorrelationID - Si necesita soporte técnico de Microsoft, este GUID le permite a Microsoft recopilar datos de diagnóstico adicionales.
- SPRequestDuration - Es el tiempo que tarda SharePoint en procesar la página.
- SPIISLatency - Es el tiempo en milisegundos que tarda SharePoint Online en cargar la página sin incluir el tiempo necesario para que la aplicación web responda.
- Page load time - Es el tiempo total registrado por la página desde el momento de la solicitud hasta el momento en que se recibió y renderizó la respuesta en el explorador.
  - Page URL - Dirección web de la página actual.

- Diagnostic tests: Aquí se enumeran los criterios con los que se analiza la página en tres secciones: lo que necesita atención, lo que se puede mejorar, y lo que no requiere ninguna acciones. Cada sección se puede extender para ver más información al respecto. Los criterios son modificados y extendidos regularmente por Microsoft, incluyendo los detalles mostrados en la pantalla.

### Attention required

#### ✖ Public and private CDN are not enabled

Both public and private CDN need to be enabled with the default configuration. Partner with your SharePoint admin to enable both CDNs and use the default CDN locations or ask you SharePoint Administrator to add custom locations. [Learn more](#)

Found 19 URLs

#### ✖ This page has 32 calls to SharePoint

Imagen 5.- Tests de Diganóstico realizados en la página.

- Network trace: provee información sobre cada llamada y respuesta. Los colores indican si el rendimiento es aceptable o no (verde < 500 ms, amarillo 500-1000 ms, rojo > 1000 ms). Puede ocurrir que algunos elementos no tengan ni color ni medida porque ya han sido almacenados en el cache del navegador; en este caso, limpie el cache y ejecute de nuevo la prueba. Los resultados se pueden exportar en formato JSON o HAR si se selecciona la opción inicialmente.

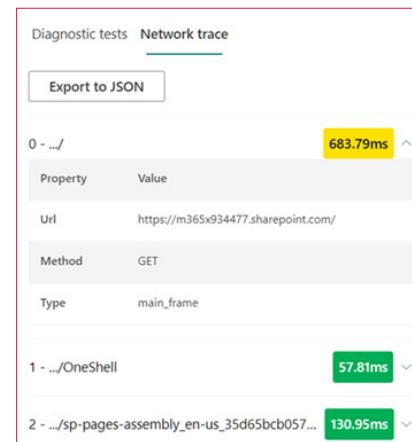


Imagen 6.- Traza de red.

**"Microsoft no lee datos ni contenido de página analizado por la herramienta, y no captura ninguna información personal, sitio web o información de descarga"**

## Usando el soporte técnico de Microsoft

Si hay algún problema en un sitio que no puede ser solucionado, es posible activar la herramienta para proporcionar información en un caso de soporte de Microsoft. Para hacerlo:

- Abra la herramienta de Diagnóstico en la página de SharePoint a investigar.
- Si la prueba todavía está ejecutando, deténgala.
- En el teclado, utilice la combinación ALT-Mayúsculas(S-hift)-L, lo que mostrará la opción “Enable support logging”.

- Active la opción y haga clic en “Start” para volver a cargar la página y generar un informe detallado.

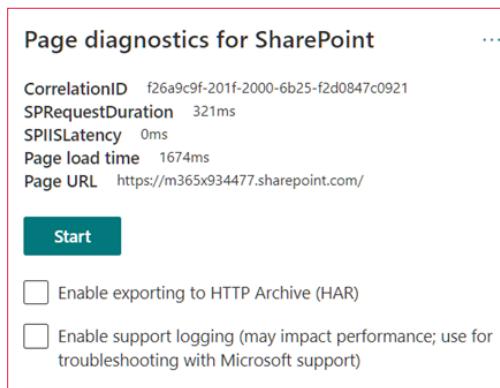


Imagen 7.- Reporte de información para un caso de soporte de Microsoft.

Anote el CorrelationID que se muestra en la parte superior de la herramienta, inicie un nuevo caso de soporte con Microsoft y proporcione el identificador para que Microsoft pueda recopilar información adicional sobre la sesión de diagnóstico.

## Reporte en las configuraciones del sitio

Fuera de las ventanas de información de la herramienta misma, la página de “Site performance” del sitio ofrece algunas otras opciones.

- Despues de ejecutar una prueba con la herramienta, desde el botón de configuración (esquina superior derecha en la página, icono de engranaje) haga clic sobre “Site performance”.
- Fuera de mostrar la misma información sobre los elementos que se deben/pueden mejorar y los que no tienen problemas, la página muestra un resumen grafico del rendimiento.

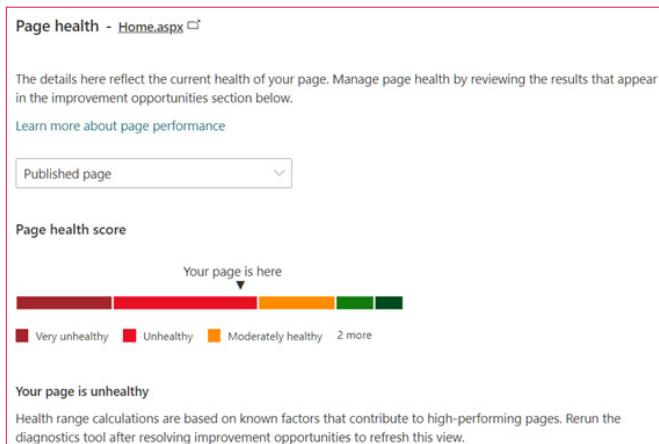


Imagen 8.- Reporte de rendimiento de una página que se genera.

Puede ser útil probar una versión de borrador de la página principal para ver cómo las mejoras afectan al estado de la página. La versión publicada de la página se puede comparar con la versión de borrador utilizando la casilla de selección en la parte superior del reporte.

## Visualización de archivos HAR

Como se comentó inicialmente, los archivos HAR son utilizados para reportar información de rendimiento de sitios web. El formato utilizado es JSON.

*"Puede ser útil probar una versión de borrador de la página principal para ver cómo las mejoras afectan al estado de la página."*

Para visualizar los archivos, existe una extensión para navegadores Chromium llamada “HTTP Archive Viewer” (<https://chrome.google.com/webstore/detail/http-archive-viewer/ebdbbdmhegaooipfnjikefdpoeaidml>) que permite analizar los informes de forma mas fácil, incluyendo la posibilidad de ver la consulta y la respuesta:

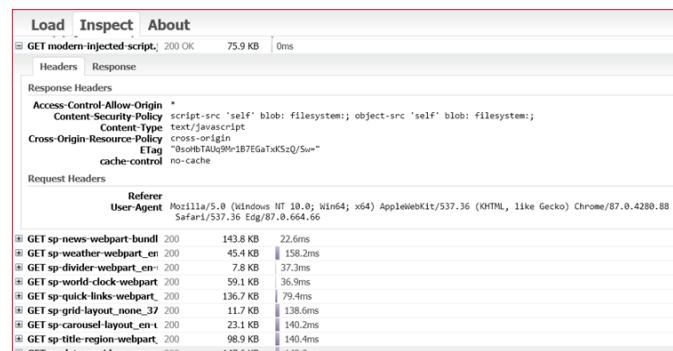


Imagen 9.- Extensión “HTTP Archive Viewer”.

*Note que la extensión NO se puede utilizar en Edge.*

**GUSTAVO VELEZ**  
**MVP Office Apps & Services**  
gustavo@gavd.net  
<https://guitaca.com>

# De Office 365 a Employee Experience Platform, pasando por MyAnalytics y Workplace Analytics

Los puestos de trabajo han cambiado y las empresas no tienen nada claro si, cuando controlemos la pandemia, volveremos al modelo “antes del Covid” ¿No es el teletrabajo una ventaja para la conciliación familiar?

Lo que sí sabemos es que la tecnología que gira alrededor del puesto de trabajo ha sido acelerada en estos primeros meses “post Covid”. Podemos ver cómo los esfuerzos de empresas como Microsoft, Google, Zoom, se han centrado en dar soporte a los sistemas cloud que permiten el uso de su tecnología y en mejorar sus sistemas a pasos agigantados. Es posible que, por ejemplo, Microsoft no tuviera en sus planes liberar este año todas las nuevas capacidades de su plataforma Microsoft Teams, pero lo ha hecho, al igual que sus competidores.

¿Y ahora qué? La adopción forzada ha sido prácticamente un éxito en todas las empresas que pueden teletrabajar. Vemos como los sistemas adquieren muchas nuevas funcionalidades, pero ¿nos ayudan a teletrabajar? ¿está siendo rentable para las empresas y/o para los trabajadores?

Creo que las empresas han sido capaces de adaptarse y el teletrabajo funciona, sin embargo, llegan muchos mensajes de los trabajadores de sobresaturación y exceso de horas y, por consiguiente, trabajo. No es que el trabajo haya aumentado, es que la intensidad con la que lo ejecutamos es muy superior a la intensidad o dedicación que le poníamos en la oficina. Y es aquí donde vamos a ver innovación en el 2021.

***"MyAnalytics es un servicio personal que nos ayuda a explorar nuestros patrones de trabajo"***

Los sistemas de teletrabajo, como Microsoft Teams, van a empezar a analizar nuestro comportamiento, nuestro día a día, para ayudarnos a ser mejores y no morir en el intento. Ahora que ya hemos adoptado la tecnología, es la tecnología la que nos tiene que ayudar a organizarnos mejor, a aprovechar más el tiempo, a tener las pausas adecuadas y las distracciones necesarias para que no lleguemos a la fatiga mental y física en la que nos encontramos hoy.

El “Internet of Behaviors”, tecnología estratégica del 2021 por Gartner, trata de analizar el comportamiento de las personas en base a las señales digitales que tenemos sobre ellas, y las empresas no son ajenas a estas señales, nece-

sitan de sistemas que les permitan aprovechar las señales y analizarlas para conseguir una mejor experiencia de sus empleados, en el uso de la tecnología o en su trabajo digital, lo que se conoce como “Employee Experience Platform” o EXP.

Microsoft lleva años trabajo en estos conceptos como podemos ver en MyAnalytics y Workplace Analytics, dos servicios de Office 365 que analizan el uso de las herramientas para mejorar la experiencia de los empleados.

MyAnalytics es un servicio personal que nos ayuda a explorar nuestros patrones de trabajo para mejorar nuestra forma de trabajar, algoritmos de inteligencia artificial que nos ayudan a mejorar nuestra concentración, bienestar, red y colaboración.

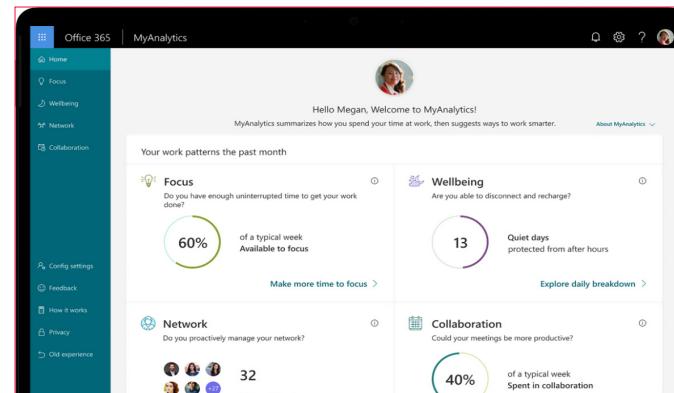


Imagen 1. Dashboard de MyAnalytics

Por el contrario, Workplace Analytics es un servicio organizativo que aporta conclusiones y experiencias sobre productividad y bienestar con el objetivo de ayudar a los empleados y a la empresa a mejorar su día a día.

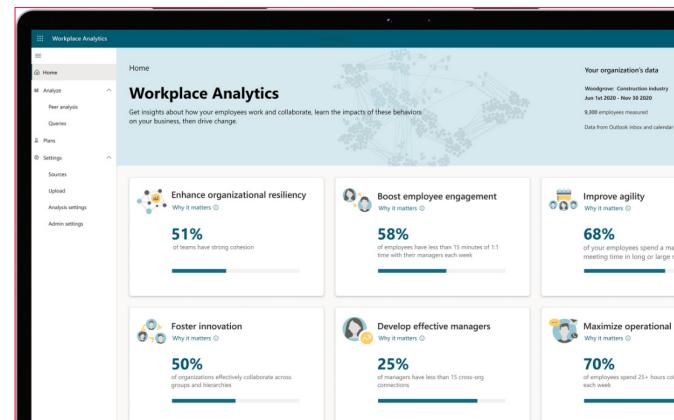


Imagen 2.- Dashboard Workplace Analytics

Workplace Analytics analiza de forma anónima el uso de que hacemos de la tecnología de Microsoft para ayudarnos a mejorar e innovar en nuestros procesos diarios, por ejemplo:

Transformar la cultura de reuniones. Las reuniones son esenciales para la colaboración, sin embargo, las reuniones innecesarias y las malas prácticas pueden perjudicar la participación y limitar la productividad

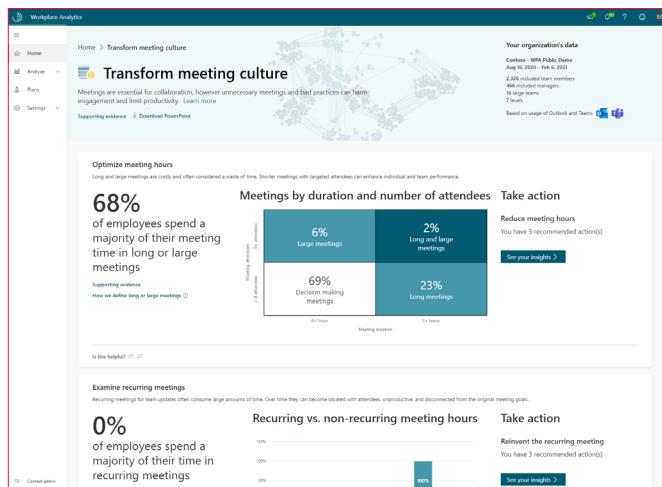


Imagen 3.- Transform meeting culture dashboard.

Mejorar la resiliencia de la organización. La resiliencia depende de tener la información, los sistemas y las herramientas adecuadas para responder a cambios repentinos, recuperarse de interrupciones y re-imaginar el futuro de su organización.

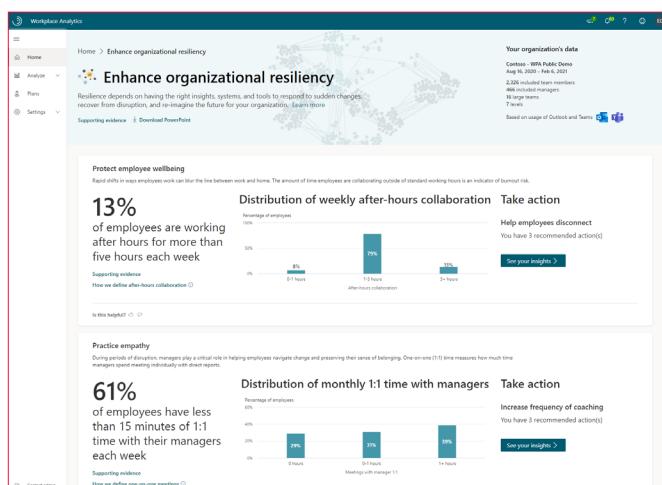


Imagen 4.- Enhance organizational resiliency dashboard.

Aumentar la participación de los empleados. Los empleados con alta satisfacción laboral y un fuerte sentido de pertenencia son más propensos a producir trabajo de alta calidad, identificar oportunidades de negocio y permanecer en la organización.

**"Workplace Analytics es un servicio organizativo que aporta conclusiones y experiencias sobre productividad y bienestar"**

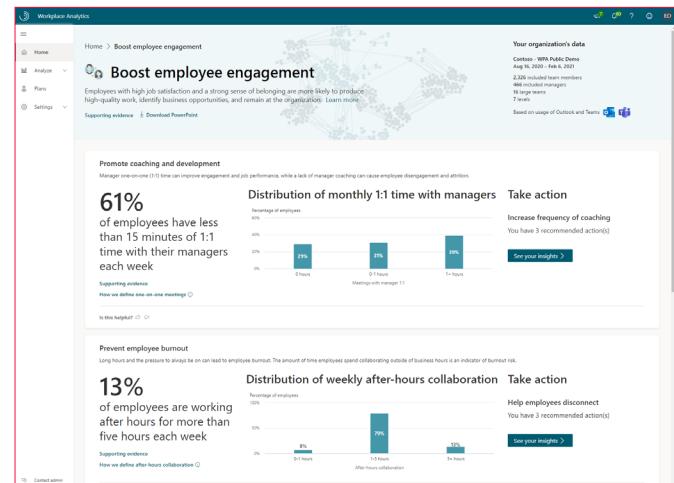


Imagen 5.- Boost employee engagement dashboard.

Mejorar la agilidad. Las empresas que redefinen las industrias y lideran los mercados son menos burocráticas y pueden adaptarse ágilmente a los rápidos cambios en la tecnología y las necesidades de los clientes.

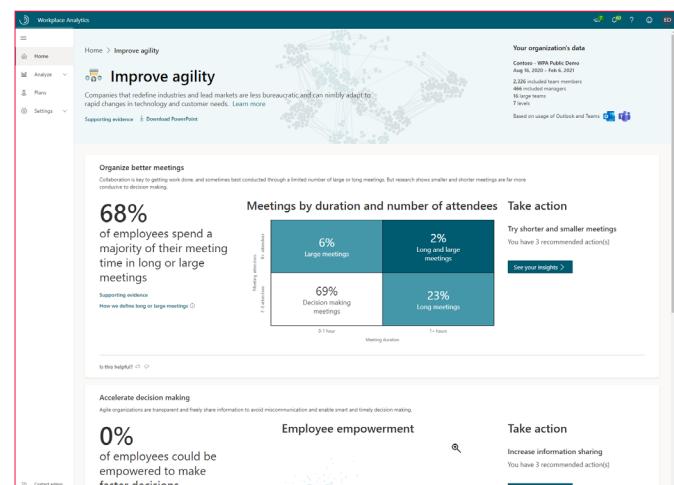


Imagen 6.- Improve agility dashboard.

Fomentar la innovación. Los empleados que comparten información, priorizan el aprendizaje y protegen su tiempo para el pensamiento profundo, generan las nuevas ideas necesarias para el éxito en mercados en rápida evolución.

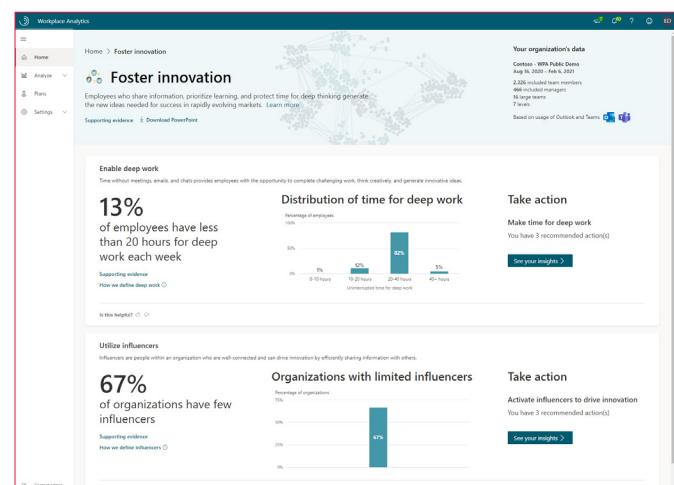


Imagen 7.- Foster innovation dashboard.

Desarrollar gerentes eficaces. Los gerentes tienen un gran impacto en la participación, el desarrollo y el rendimiento

de los empleados, y son fundamentales para impulsar el cambio organizativo.

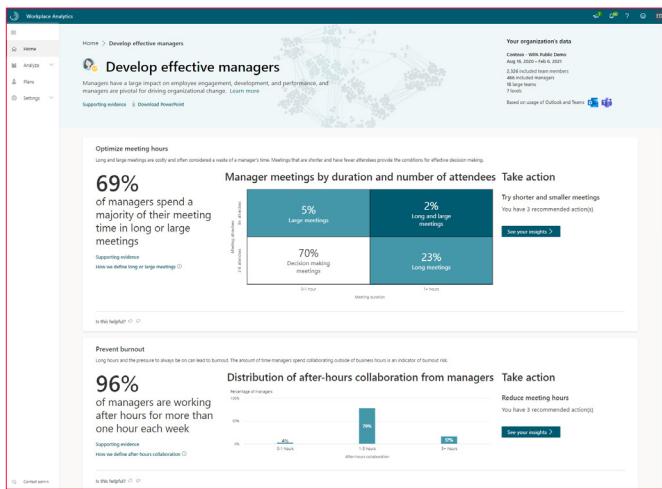


Imagen 8.- Develop effective managers dashboard.

**Mejorar la eficacia operativa.** Los pequeños cambios en las prácticas de colaboración pueden tener efectos transformadores en la productividad de la organización a escala.

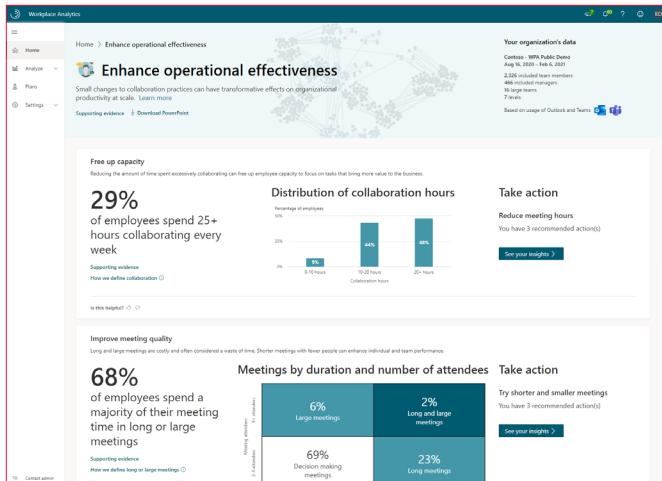


Imagen 9.- Enhance operation effectiveness dashboard.

**Acelerar el cambio.** La lenta adopción de nuevas tecnologías perjudica los esfuerzos para atraer y retener a los mejores talentos, mejorar la productividad y pueden conducir a un fracaso del mercado.

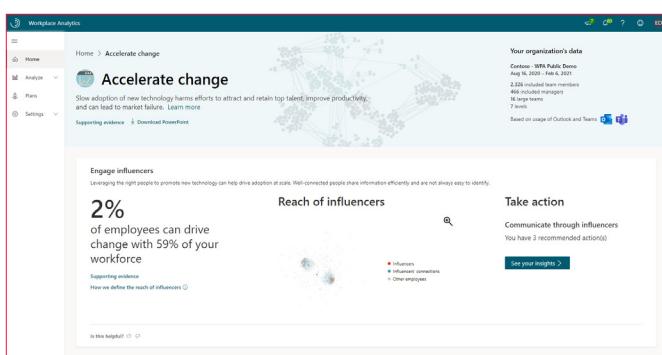


Imagen 10.- Accelerate change dashboard.

**Aumentar el enfoque del cliente.** Las empresas que priorizan las relaciones con los clientes y la satisfacción obtienen ingresos más rápido que los competidores.

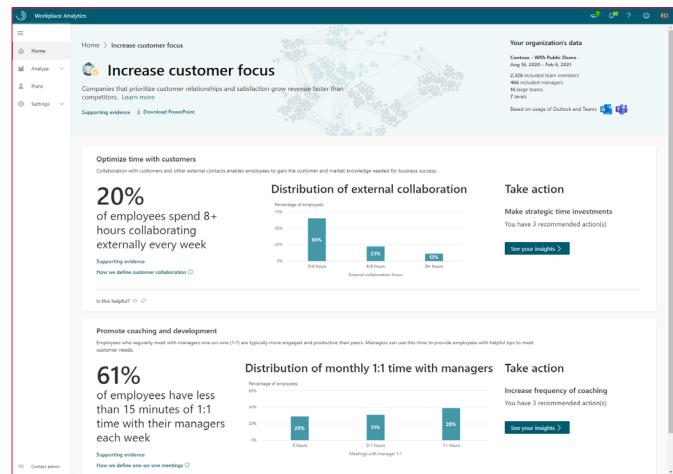


Imagen 11.- Increase customer focus dashboard.

Respondiendo a la evolución del mercado y usando el IoB es como aparece Microsoft Viva, la EXP de Microsoft que, actualmente, se centra en las siguientes perspectivas:

Connections un servicio para mejorar las conexiones entre los empleados fomentando la participación y ayudando a la comunicación interna.

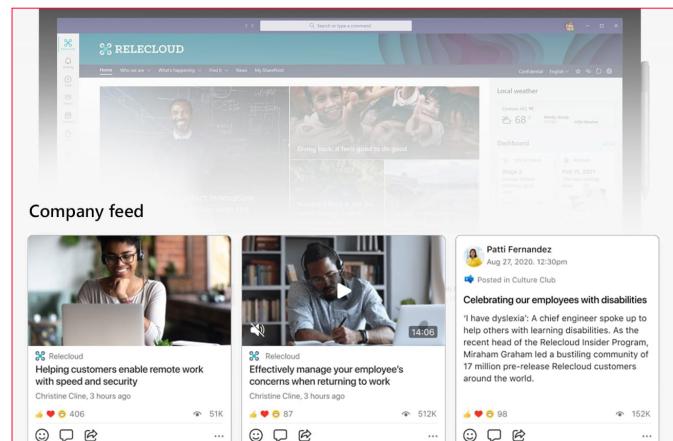


Imagen 12.- Ejemplo de mensajes corporativos de Viva Connection.

Insights para ayudar a las personas a fomentar su bienestar y ayudarles a obtener su mejor versión, usando los datos de MyAnalytics y de Workplace Analytics.

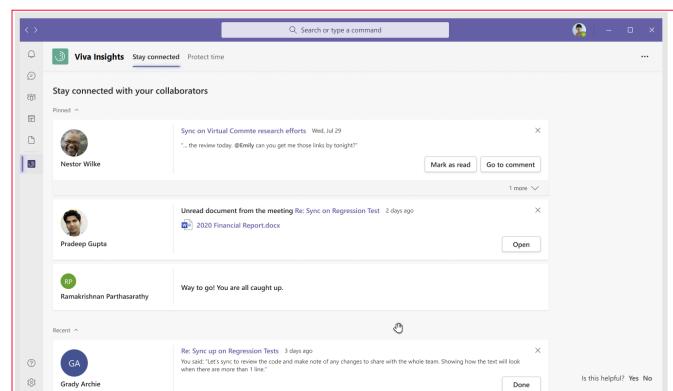
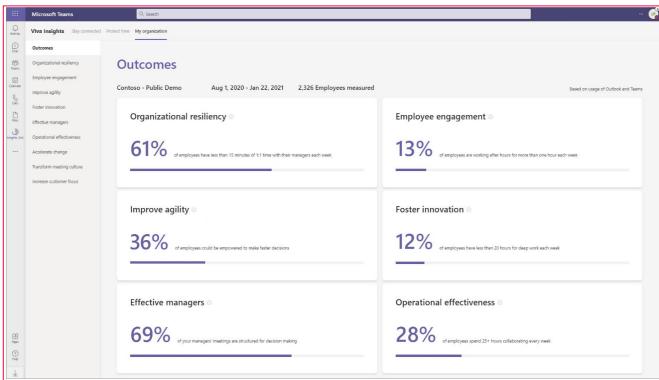


Imagen 13.- Vista personal de Viva Insights en Microsoft Teams.

**"La lenta adopción de nuevas tecnologías perjudica los esfuerzos para atraer y retener a los mejores talento"**

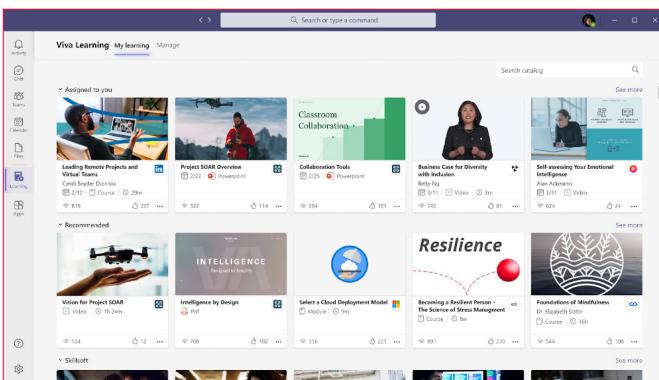


This screenshot shows the Microsoft Viva Insights interface in Microsoft Teams. It displays various organizational outcomes with associated percentages:

- Outcomes**
- Organizational resiliency**: 61% of employees have less than 15 minutes of 1:1 time with their manager each week.
- Employee engagement**: 13% of employees are working after hours for more than one hour each week.
- Improve agility**: 36% of employees could be empowered to make faster decisions.
- Foster innovation**: 12% of employees have less than 20 hours for deep work each week.
- Effective managers**: 69% of your manager's meetings are structured for decision-making.
- Operational effectiveness**: 28% of employees spend 25+ hours collaborating every week.

Imagen 14.- Vista organizativa de Viva Insights en Microsoft Teams-

Learning con el objetivo de ayudarnos por medio de inteligencia artificial a que la formación sea una parte esencial y natural de la organización.



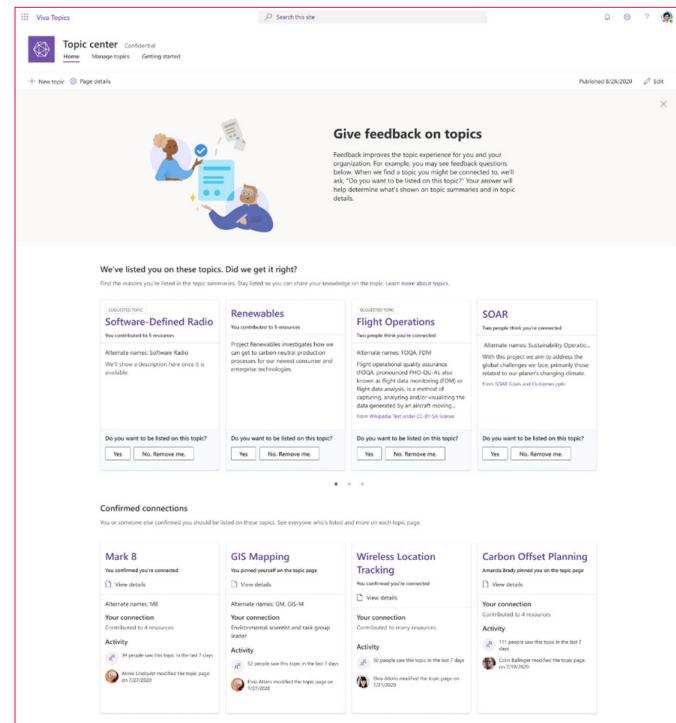
This screenshot shows the Microsoft Viva Learning interface in Microsoft Teams. It displays recommended learning content:

- Assigned to you**: Leading Female Projects and Visual Studio Courses.
- Recommended**: Project SOAR Overviews, Resilience, Intelligence by Design, and others.

Imagen 15.- Viva Learning en Microsoft Teams.

Topics es un servicio de inteligencia artificial que convierte los datos de Office 365 en conocimiento y experiencia para mejorar el consumo y la búsqueda del contenido, basado en Project Cortex.

**"Respondiendo a la evolución del mercado y usando el IoB es como aparece Microsoft Viva"**



This screenshot shows the Microsoft Viva Topics Topic Center in Microsoft Teams. It displays suggested topics with details and connection status:

- SUGGESTED TOPIC: Software-Defined Radio**: You contributed to 5 resources. Alternative names: Software Radio. Project Renewables investigates how we can get to carbon neutral production processes for our newest consumer and enterprise technologies.
- Renewables**: Two people think you're connected. Alternative names: SOAR, FDM. Flight operational quality assurance (FOQA), pronounced FHO-QA, also known as flight performance monitoring or flight data analysis, is a method of measuring aircraft performance using the data generated by an aircraft moving... More (whisper text under CC-0 SA license)
- SUGGESTED TOPIC: Flight Operations**: Two people think you're connected. Alternative names: Sustainability Operations... This project was created to be a global hub for information on how personally those related to our planet's changing climate... From OEMs, Grids, and Customers (text)
- SOAR**: Two people think you're connected. Alternative names: Sustainability Operations... This project was created to be a global hub for information on how personally those related to our planet's changing climate... From OEMs, Grids, and Customers (text)

Do you want to be listed on this topic? Yes No Remove me.

Do you want to be listed on this topic? Yes No Remove me.

Do you want to be listed on this topic? Yes No Remove me.

Do you want to be listed on this topic? Yes No Remove me.

**Confirmed connections**: See someone else confirmed you should be listed on these topics. See everyone who's listed and more on each topic page.

**Mark 8**: You confirmed you're connected. View details. Alternative names: M8. Your connection: Contributed to 4 resources. Activity: 34 people saw this topic in the last 7 days. Annie Luskayla modified the topic page on 7/27/2020.

**GIS Mapping**: You pinned yourself on the topic page. View details. Alternative names: GM, GIS-M. Your connection: Environmental scientist and task group leader. Activity: 32 people saw this topic in the last 7 days. Krisi Atiles modified the topic page on 7/27/2020.

**Wireless Location Tracking**: You confirmed you're connected. View details. Your connection: Contributed to many resources. Activity: 30 people saw this topic in the last 7 days. Colin Ballantyne modified the topic page on 7/27/2020.

**Carbon Offset Planning**: Amazon listed you on the topic page. View details. Your connection: Contributed to 4 resources. Activity: 111 people saw this topic in the last 7 days. Colin Ballantyne modified the topic page on 7/27/2020.

Imagen 16.- Topic Center.

Con estas cuatro perspectivas, Microsoft Viva es el inicio de una nueva plataforma que como objetivo principal tiene la mejora de la experiencia de las personas que trabajan en una organización y que sean capaces de ofrecer su mejor versión, gracias a los datos y a la inteligencia artificial aplicada a los datos.

**ALBERTO DIAZ MARTIN**  
**Microsoft Regional Director**  
**Microsoft Azure MVP**  
**Office Apps & Services MVP**  
**adiazcan@hotmail.com**  
**@adiazcan**  
<http://blogs.encamina.com/por-una-nube-sostenible/>

# Entrevista Juan Camilo Martínez Vélez

Mi nombre es Juan Camilo Martínez Vélez, y soy de la hermosa ciudad de Medellín, Colombia. Llevo casi 20 años trabajando en tecnología y básicamente he repartido mi vida laboral en dos compañías, una del sector retail y desde hace 11 años estoy dedicado al servicio en tecnología, para Intergrupo una compañía SoftwareOne.



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

Desde niño siempre busqué servir y por esto mis pensamientos siempre estaban en cómo podía ayudar a los más cercanos, como todos, de niño siempre quise estudiar medicina y ahora que lo pienso tal vez fue por ayudar a mis abuelos; luego pensé estudiar ingeniería civil porque quería mejorar las vías de mi país, siempre pensé que el progreso estaba allí y aunque todavía lo pienso, pasaba algo terrible y es que no me gustaba estar mucho en la calle ☺.

Fue así como en una familia de ingenieros casi que debía estar orientado a una ingeniería, pero pasaba algo terrible de nuevo y era que no me gustaba mucho programar. Fue así, como encontré que también había otras cosas, que también estábamos los genios de infraestructura, aquellos que teníamos las bases para todas esas aplicaciones que los desarrolladores construían.

Todo fue una serie de hechos afortunados y fue así como en medio de oportunidades que grandes líderes me dieron, me fui enamorando de este camino, en medio de switchs, routers, servidores, almacenamientos, cables de fibra y entre otras cosas que complementaron mi carrera profesional.

Creo que hay sucesos que siempre van a cambiar tu vida, pero para mí, creer que soy bueno para resolver incidentes, sentirme muy bueno para implementar tecnologías y llevar a cabo proyectos, era mi mayor desafío, junto con entender como estaban cambiando las eras de la tecnología y lograr involucrarme rápidamente en lo que estaba saliendo.

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

Después de hacer un recorrido por el mundo de infraestructura y cambiando mi mentalidad al mundo nube, todos mis esfuerzos laborales hoy en día, están enfocados en Microsoft Teams como plataforma y aunque las comunicaciones unificadas como concepto han sido mi fuerte durante muchos años, ahora la promuevo desde el mundo de las aplicaciones empresariales.

Muchas veces cuando trabajas en un solo producto te podrías limitar, pero ahora puedo reunir todos mis esfuerzos para promover integraciones y formas de mejorar la vida de los usuarios a través de la automatización de procesos, todos direccionados a un mismo entorno.

Es así como en mi trabajo, ahora estoy enfocado en soluciones para todo Sur América, incluyendo Brasil y con te-

letrabajo desde hace más de un año, puedo iniciar mi día estando en contacto con muchos países.

También, en mi tiempo libre, me encanta hacer comunidad tecnológica, no por el hecho de ganar reconocimiento ni por hacer negocio, sino por el amor a compartir y la dedicación al servicio. En este mundo rotamos mucho y lo más importante es que puedo dejarles a las personas a las que ayudo a ser un generador de ideas valiosas.

Fue así también como Microsoft me galardonó como MVP (Microsoft Most Valuable Professional) en el año 2019 lo que me ayudó a conocer personas maravillosas de este mundo de tecnología.

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

Por diversas razones, cuando alguien es bueno técnicamente va creciendo y en mi caso busqué también pasar por la parte administrativa, me tocó conocer cosas nuevas que en mi vida nunca había usado y las cuales por pasión aprendí, es así como también me involucré en el mundo financiero, para entender el negocio en la que estaba, este maravilloso negocio de vender servicios.

## ¿Cuáles son tus hobbies?

Soy un apasionado por la familia, por disfrutar la vida, salir de paseo y de volver todo divertido. Muchas veces no es a donde vas sino como te sientes llegando hasta allí y por eso me encanta salir.

También como a muchos aquí en Colombia, me encanta la bicicleta, porque es ese momento para estar con mis pensamientos, en los que rueda a rueda puedo pensar más, generar ideas y ver la vida de otra forma.

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

Siempre tengo un discurso preparado cuando me hacen esta pregunta, porque siempre he tenido la visión como algo muy importante, no es sólo el hecho de llegar, sino que harás cuando llegues; con esta historia que les he contado hasta ahora, puedo partir mis años de trabajo en tres eras:

La primera era, fue esa época de la virtualización, porque en aquellos días el hardware era costoso, pero tener ambientes virtuales era la mejor solución para muchos. Varios de mis proyectos para compañías grandes, tienen que ver con esto y así fue mi primera era.

La segunda era, fue cuando todos tenían miedo de aquel concepto llamado nube y muchos de mis proyectos fueron así, en cómo llevar muchos de mis servicios a la nube y aunque todavía lo hago, ya me estoy subiendo en aquella tercera era, la cual es lo que estamos viendo con las aplicaciones, es así como incluso en el celular tenemos, la app del banco, la de mensajes, las redes sociales y otros. Por eso mi trabajo está tan centralizado en promover las aplicaciones para Microsoft Teams y viendo a este cliente, como plataforma central.

Teniendo todo esto, esperamos en el futuro, mucha más realidad aumentada, más cercanía porque, aunque el teletrabajo es bueno, muchas veces nos aleja de ciertas cosas a las que estábamos acostumbrados en el puesto de trabajo, como tener cercanía con las otras personas, que no sea solo por videoconferencia, sino poder llegar a tenerla de formas inesperadas.

i

13

# Asp.Net core - como configurar tu aplicación para despliegue en una granja de servidores

En este artículo, continuación de ASP.NET Core - Gestión Cookies De Autenticación Server Side, veremos cómo hay que configurar tu aplicación ASP.NET Core para que se pueda desplegar en una granja de servidores de una forma exitosa.

Lo primero que necesitamos saber es qué es una granja de servidores:

Una granja de servidores es un conjunto de dos o más servidores web (nodos) que pueden albergar varias instancias de una aplicación web.

Una granja siempre se encuentra detrás de un load balancer o balanceador de carga, que se encarga de distribuir todas las peticiones recibidas a los diferentes nodos que componen dicha granja.

## Ventajas de emplear una granja de servidores

- **Fiabilidad.** Cuando un nodo falla, el balanceador de carga distribuye las peticiones a aquellos que están disponibles.
- **Rendimiento.** Tener varios nodos permite manejar muchas más peticiones.
- **Escalabilidad.** El sistema puede escalar a mayor o menor número de nodos según la carga de trabajo.

Si no se realiza correctamente la configuración de nuestra aplicación para poder desplegar en una granja de servidores, obtendremos seguro, unas excepciones en muchas peticiones. ¿Por qué? Porque tomemos un ejemplo de encriptar una cookie desde una instancia de ASP.NET Core, al mandar ese valor de vuelta y el load balancer mande la petición a otra, ésta no será capaz de desencriptar dicha cookie. Esto es muy común en el middleware de autenticación de OpenId.

Las partes fundamentales que debemos tener en cuenta para tener éxito en el despliegue en una granja de servidores son:

- Data Protection.
- Caching.

## Data Protection

Al activar Data Protection, por defecto las claves se almacenan en disco, por lo que un nodo no puede descifrar algo

cifrado por otro ya que las claves son distintas. Para ello es necesario establecer un punto común de claves que se comparten a través de todos los nodos de la granja.

## Caching

En una granja nuestro sistema de cacheo debe gestionar elementos de todos los nodos de la granja, como se indicó en el punto anterior, buscamos un punto común compartido. La ventaja de tener el sistema de cacheo de forma distribuida es que no consume recursos de nuestro nodo y que por ejemplo al reiniciarlo, la caché sigue persistida.

En el ejemplo veremos cómo usar una base de datos vía Entity Framework Core para persistir y gestionar las claves de protección, así como establecer un sistema de caché distribuido, empleando para ello SQL Server como motor de base de datos.

También en el código de demo usaremos Identity Server para la gestión del login de los usuarios.

## ¿Cómo implementar esto?

Para configurar Data Protection en este ejemplo lo que se procede a implementar un DbContext.

```
public class DataProtectionKeysContext : DbContext, IDataProtectionKeyContext
{
    public DataProtectionKeysContext(DbContextOptions<DataProtectionKeysContext> options)
        : base(options) {}

    public DbSet<DataProtectionKey> DataProtectionKeys { get; set; }

    public DbSet<Customer> Customers { get; set; }

    protected override void OnModelCreating(ModelBuilder modelBuilder)
    {
        modelBuilder.Entity<Customer>().ToTable("Customers");
        HasKey(t => new { t.Id });
        base.OnModelCreating(modelBuilder);
    }
}
```

Si os fijáis, en este DbContext, implemento un interfaz denominado IDataProtectionKeyContext para poder almacenar las claves usadas por ASP.NET Core DataProtection.

Data Protection permite guardar las claves tanto en ficheros como blobs de Azure Storage, base de datos, Azure Key Vault, etc... así como utilizar certificados para firmar y añadir un plus de protección a nuestras claves.

Os recomiendo echéis un vistazo a Configure ASP.NET Core Data Protection para que veáis varias opciones.

Para configurar el servicio de caché distribuido es necesario que os descargueís el paquete Microsoft.Extensions.Caching.SqlServer.

**"Data Protection permite guardar las claves tanto en ficheros como blobs de Azure Storage, base de datos, Azure Key Vault, etc"**

Ejecutar el siguiente comando dotnet para generar la tabla de base datos que actuará para almacenar los elementos cacheados

```
PM> dotnet sql-cache create "Data Source=(localdb)\MSSQLLocalDB;Initial Catalog=aspnet-AspNetCoreHostInWebFarm;Integrated Security=True;" dbo DistributedCache
```

En este caso se va a configurar la tabla denominada DistributedCache en el servidor indicado en la cadena de conexión.

La tabla también la podremos crear con el siguiente script de SQL:

```
USE [aspnet-AspNetCoreHostInWebFarm]
GO

SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[DistributedCache](
    [Id] [nvarchar](449) NOT NULL,
    [Value] [varbinary](max) NOT NULL,
    [ExpiresAtTime] [datetimeoffset](7) NOT NULL,
    [SlidingExpirationInSeconds] [bigint] NULL,
    [AbsoluteExpiration] [datetimeoffset](7) NULL,
    PRIMARY KEY CLUSTERED
    (
        [Id] ASC
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
```

Una vez hecho esto sólo queda configurar la caché agregando el servicio en el motor de inyección de dependencias de Net Core.

No se me olvida comentaros que el interfaz a usar en nuestra caché distribuida es `IDistributedCache`.

En el método `ConfigureServices` de nuestra clase `Startup` tendrá el siguiente aspecto.

```
///<summary>
/// ConfigureServices
///</summary>
///<param name="services"></param>
// This method gets called by the runtime. Use this method to add services to the container.
public void ConfigureServices(IServiceCollection services)
{
    services.AddControllersWithViews();

    // Configure Data Protection
    var encryptionSettings = new AuthenticatedEncryptorConfiguration()
    {
        EncryptionAlgorithm = EncryptionAlgorithm.AES_256_CBC,
        ValidationAlgorithm = ValidationAlgorithm.HMACSHA256
    };

    services.AddDbContext<DataProtectionKeysContext>(options =>
        options.UseSqlServer(
            Configuration.GetConnectionString("DataProtectionKeysConnection")));
}

services.AddDataProtection()
    .PersistKeysToDbContext<DataProtectionKeysContext>()
    .SetApplicationName("demo")
    .UseCryptographicAlgorithms(encryptionSettings);

// Configure Distributed Cache
services.AddDistributedSqlServerCache(options =>
{
    options.ConnectionString = Configuration.GetConnectionString("DataProtectionKeysConnection");
    options.SchemaName = "dbo";
    options.TableName = "DistributedCache";
});

// Configure Distributed Cache
services.AddAuthentication(options =>
{
    options.DefaultScheme = "cookie";
    options.DefaultChallengeScheme = "oidc";
})
    .AddCookie("cookie")
    .AddOpenIdConnect("oidc", options =>
{
    options.Authority = "https://localhost:5001";
    options.ClientId = "mvc.client";
    options.ClientSecret = "36F742BA-D9BF-49FE-B91A-D25E3A6354A5";

    // code flow + PKCE (PKCE is turned on by default)
    options.ResponseType = "code";
    options.UsePkce = true;

    options.Scope.Add("openid");
    options.Scope.Add("profile");
    options.Scope.Add("scope1");
    options.Scope.Add("scope2");
    options.GetClaimsFromUserInfoEndpoint = true;
    options.SaveTokens = true;
});
}
```

Procedo a explicar el código para que no os perdáis:

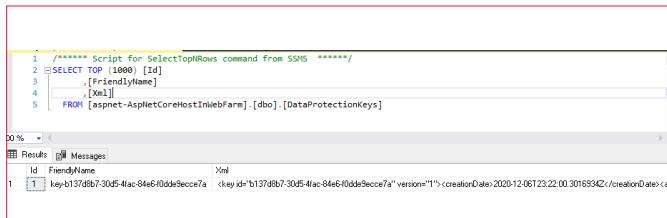
- Incluimos nuestro contexto de base de datos en el motor de inyección de dependencias.
- Agregamos los servicios de DataProtection e indicamos que las claves se deben incluir en el contexto de base de datos del punto anterior.
- Agregamos el servicio de caché distribuida de SQL Server que empleará la base de datos indicada en su cadena de conexión, puede ser una tabla de nuestra base de datos de la aplicación u otra externa.
- Agregamos los servicios de autenticación en los cuales configuraremos.

- Agregamos los servicios para los controladores y vistas.

## ¿Cómo verificamos que todo esto funciona?

Para verificar que las claves de Data Protection se han generado correctamente, con tan solo iniciar nuestra aplicación y realizar la siguiente consulta veremos resultados:

```
SELECT TOP (1000) [Id]
 ,[FriendlyName]
 ,[Xml]
FROM [aspnet-AspNetCoreHostInWebFarm].[dbo].[DataProtectionKeys]
```



```
1 /*===== Script for SelectTopNRows command from SSMS =====*/
2 SELECT TOP (1000) [Id]
3 ,[FriendlyName]
4 ,[Xml]
5 FROM [aspnet-AspNetCoreHostInWebFarm].[dbo].[DataProtectionKeys]
```

Id	FriendlyName	Xml
key-b137d8b7-30d5-4fac-84e6-f0dde9ecce7a	Test Name	<key id="b137d8b7-30d5-4fac-84e6-f0dde9ecce7a" version="1"><creationDate>2020-12-06T23:22:00.3016934Z</creationDate><activationDate>2020-12-06T23:21:57.6285509Z</activationDate><expirationDate>2021-03-06T23:21:57.6285509Z</expirationDate><descriptor deserializerType="Microsoft.AspNetCore.DataProtection.AuthenticatedEncryption.ConfigurationModel.AuthenticatedEncryptorDescriptorDeserializer", Microsoft.AspNetCore.DataProtection, Version=3.1.9.0, Culture=neutral, PublicKeyToken=adb9793829ddae60"><descriptor><encryption algorithm="AES_256_CBC" /><validation algorithm="HMACSHA256" /><masterKey p4:requiresEncryption="true" xmlns:p4="http://schemas.asp.net/2015/03/dataProtection"><!-- Warning: the key below is in an unencrypted form. --><value>oWyh4tBJ8LqoCexLNWPx0cf4CJftdal667N5zt-M1+JdcOa4afKhRgWWUvG6Cty5KLaCW8OK32xVgk34gYxxd-QA==</value></masterKey></descriptor></descriptor></key>

El XML generado con la clave tendría más o menos el siguiente aspecto:

```
<key id="b137d8b7-30d5-4fac-84e6-f0dde9ecce7a" version="1">
  <creationDate>2020-12-06T23:22:00.3016934Z</creationDate>
  <activationDate>2020-12-06T23:21:57.6285509Z</activationDate>
  <expirationDate>2021-03-06T23:21:57.6285509Z</expirationDate>
  <descriptor deserializerType="Microsoft.AspNetCore.DataProtection.AuthenticatedEncryption.ConfigurationModel.AuthenticatedEncryptorDescriptorDeserializer", Microsoft.AspNetCore.DataProtection, Version=3.1.9.0, Culture=neutral, PublicKeyToken=adb9793829ddae60">
    <descriptor>
      <encryption algorithm="AES_256_CBC" />
      <validation algorithm="HMACSHA256" />
      <masterKey p4:requiresEncryption="true" xmlns:p4="http://schemas.asp.net/2015/03/dataProtection">
        <!-- Warning: the key below is in an unencrypted form. -->
        <value>oWyh4tBJ8LqoCexLNWPx0cf4CJftdal667N5zt-M1+JdcOa4afKhRgWWUvG6Cty5KLaCW8OK32xVgk34gYxxd-QA==</value>
      </masterKey>
    </descriptor>
  </descriptor>
</key>
```

Como se puede observar, se ven los algoritmos configurados AES\_256\_CBC y HMACSHA26.

Veamos ahora ejemplos de cacheo.

```
[Authorize]
public class HomeController : Controller
{
  private readonly IDistributedCache distributedCache;
  private readonly DataProtectionKeysContext dbContext;

  public HomeController(IDistributedCache distributedCache, DataProtectionKeysContext dbContext)
  {
    this.distributedCache = distributedCache;
```

```
    this.dbContext = dbContext;
  }

  public IActionResult Index()
  {
    return View();
  }

  public IActionResult Privacy()
  {
    return View();
  }

  [ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore = true)]
  public IActionResult Error()
  {
    return View(new ErrorViewModel { RequestId = Activity.Current?.Id ?? HttpContext.TraceIdentifier });
  }

  public async Task<ActionResult> AddCacheItem()
  {
    string key = "customerKey:1";
    var customer = new Customer()
    {
      Id = Guid.NewGuid(),
      Name = "Test Name",
      Surname = "Test Surname",
      BirthDay = DateTime.UtcNow.AddYears(-18)
    };
    await dbContext.AddAsync<Customer>(customer);
    await dbContext.SaveChangesAsync();

    await distributedCache.SetStringAsync(key, JsonConvert.SerializeObject(customer));

    return Ok("success");
  }

  public async Task<ActionResult> GetCacheItem()
  {
    string key = "customerKey:1";
    var cachedItem = await distributedCache.GetStringAsync(key);

    if (!string.IsNullOrWhiteSpace(cachedItem))
    {
      return Ok(JsonConvert.DeserializeObject<Customer>(-
cachedItem));
    }
    else
    {
      var customer = await dbContext.Customers.FirstOrDefaultAsync(t =>t.Name == "Test Name");
      await distributedCache.SetStringAsync(key, JsonConvert.
SerializeObject(customer));

      return Ok(customer);
    }
  }

  public async Task<ActionResult> DeleteCacheItem()
  {
    string key = "customerKey:1";
    await distributedCache.RemoveAsync(key);
    return Ok("success");
  }
}
```

Al ejecutar por ejemplo el método AddCacheItem() podemos ejecutar la consulta para ver el elemento cacheado.

```
SELECT TOP (1000) [Id]
,[Value]
,[ExpiresAtTime]
,[SlidingExpirationInSeconds]
,[AbsoluteExpiration]
FROM [aspnet-AspNetCoreHostInWebFarm].[dbo].[Distribute-
dCache]
```

1	***** Script for SelectTopNRows command from SSMS *****
2	2 SELECT TOP (5000) [Id]
3	3 , [Value]
4	4 , [ExpiresAtTime]
5	5 , [SlidingExpirationInSeconds]
6	6 , [AbsoluteExpiration]
7	7 ] FROM [aspnet-AspNetCoreHostInWebFarm].[dbo].[DistributedCache]

Results Messages					
Id	Value	ExpiresAtTime	SlidingExpirationInSeconds	AbsoluteExpiration	
1	customerKey1	0x7B24E616065223A2254657374204E61606522C225375...	2020-12-07 01:18:04.5995187 +0000	1200	NULL

## ¿Dónde puedo encontrar el código de ejemplo?

Podéis descargaros los proyectos de ejemplo en <https://github.com/sparraguerra/compartimoss/tree/master/AspNetCoreHostInWebFarm>

## Conclusiones

Hemos visto lo sencillo que es implementar la configuración de una aplicación para que funcione en una granja de servidores, y hemos comprobado la importancia que tiene Data Protection y nuestro sistema de caching para ello.

Happy coding!

**SERGIO PARRA GUERRA**

Software & Cloud Architect at Encamina  
@sparraguerra

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**

 LinkedIn  Facebook  Twitter

*i*

17

# Coneectar Power Virtual Agents con QnA Maker

Luego de haber creado nuestro Power Virtual Agent veremos cómo conectarlo con el servicio de QnA Maker utilizando Power Automate. Esto nos permitirá manejar preguntas frecuentes con un servicio cognitivo especializado.

Como vimos anteriormente el Virtual Agent viene configurado por defecto con para responder a saludos, despedidas, confirmación, negación, etc.

- Una vez creado el Virtual Agent debemos dirigirnos a las opciones y seleccionar “System fallback”:

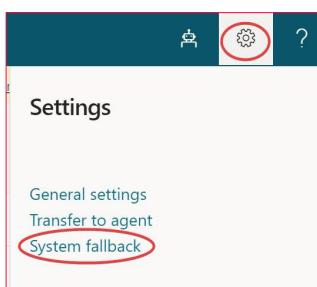


Imagen 1.- Acceso a las opciones de System fallback del Virtual Agent.

- Una vez allí no aparecerá la siguiente pantalla:

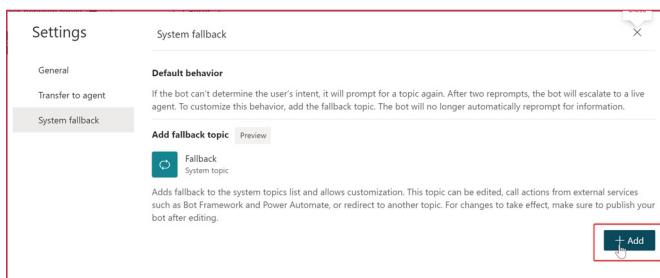


Imagen 2.- Ventana de System fallback del Virtual Agent.

- Debemos hacer Click en “Add”. Aparecerá la siguiente pantalla y vamos a “Go to fallback topic”.

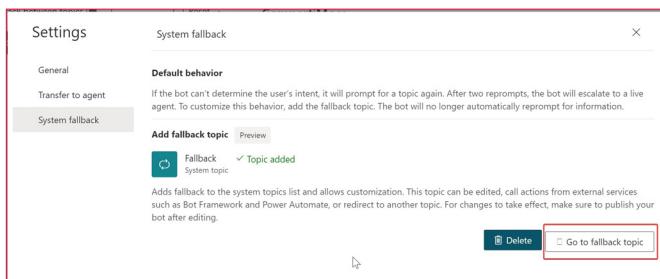


Imagen 3.- Acceso a “Go to fallback topic”.

- Nos aparecerá la siguiente pantalla donde podremos ver el flujo de la conversación.

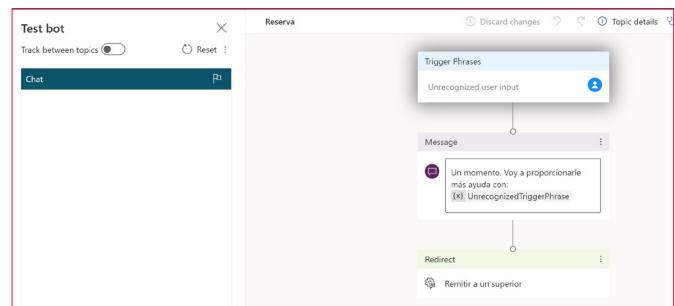


Imagen 4.- Flujo de Conversación.

- Primero eliminamos la acción “Redirect” pues no lo utilizaremos.

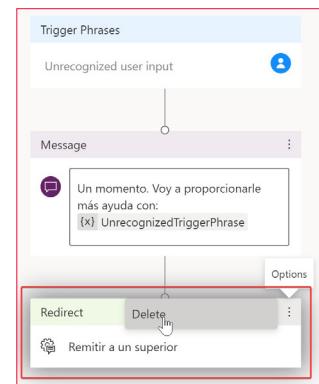


Imagen 5.- Eliminando la acción “Redirect”.

- Luego agregamos la conexión con Power Automate quien será el responsable de comunicarse con el servicio de QnA Maker. Hacemos click en “Add Node”  luego hacemos Click en “Call an Action” y “Create a Flow”

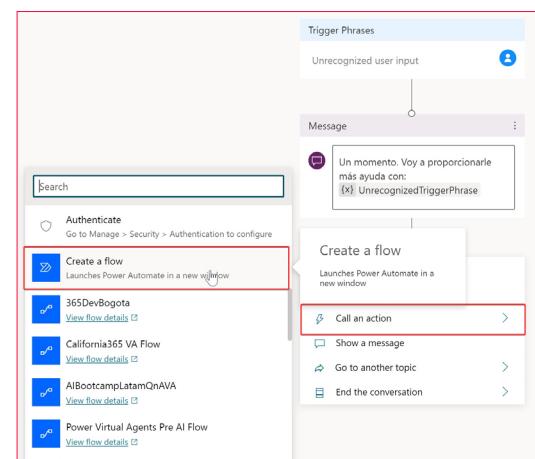


Imagen 6.- Agregando la conexión a Power Automate.

**"nos permitirá manejar preguntas frecuentes con un servicio cognitivo especializado"**

- Se nos abrirá la pantalla de Power Automate:

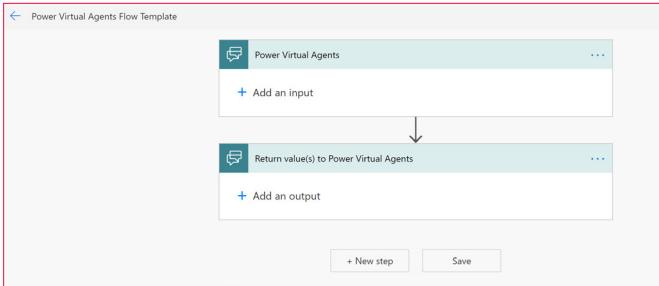


Imagen 7.- Plantilla de Power Automate para Power Virtual Agents.

- Le damos un nombre a nuestro Flow y definimos la variable de entrada que será donde vendrá el texto de la pregunta realizada por nuestro usuario en el Power Virtual Agent.

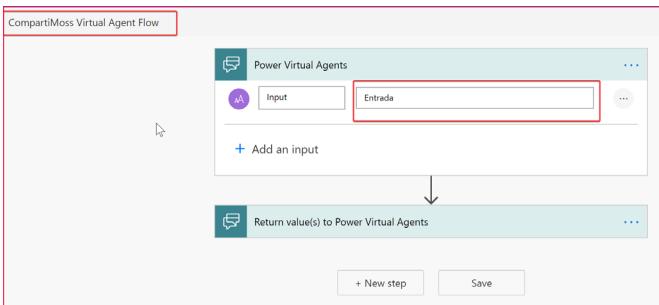


Imagen 8.- Configurando un nombre personalizado para el Flow.

- Agregamos una acción del tipo Generate answer de QnA Maker:

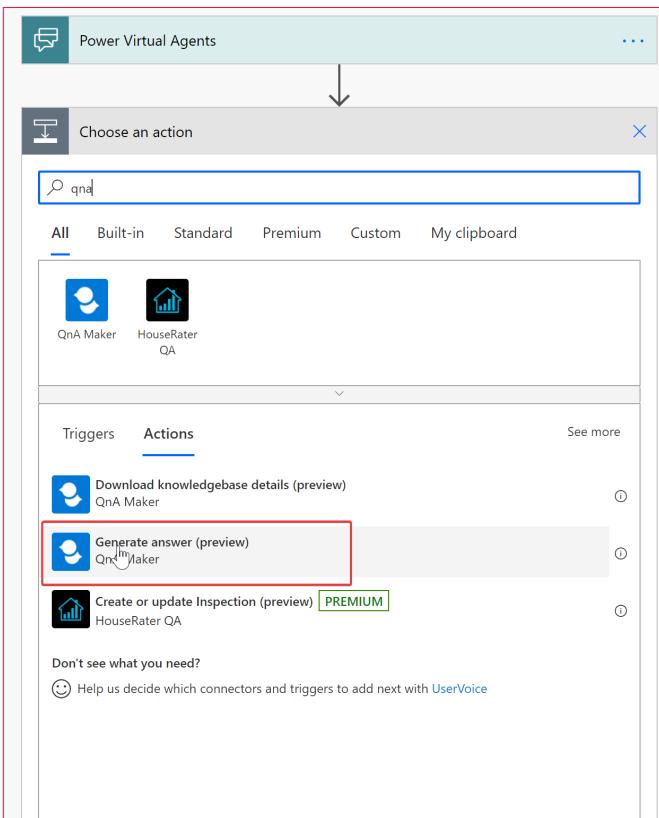


Imagen 9.- Acción de tipo "Generate answer" de QnA Maker.

- Cargamos los datos requeridos que obtenemos de nuestro servicio de QnA Maker y ponemos como pregunta a responder el texto de entrada que guardamos en la variable "input". Definimos una variable para guardar el resultado de la respuesta de tipo String.

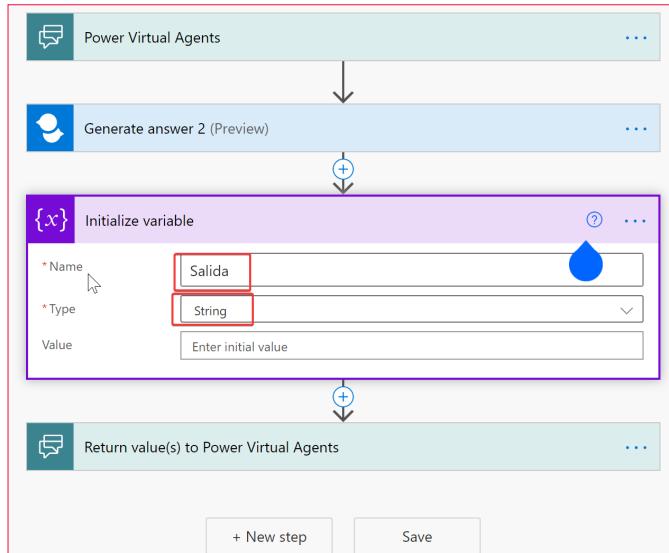


Imagen 10.- Variable de tipo string necesaria.

- Guardamos la respuesta en la variable recién definida:

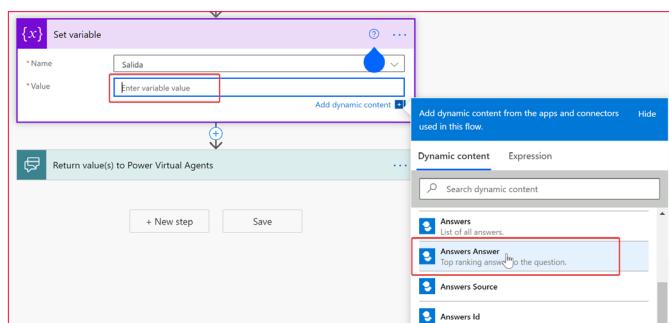


Imagen 11.- Guardando la respuesta.

- Retornamos nuestra respuesta al Virtual Agent:

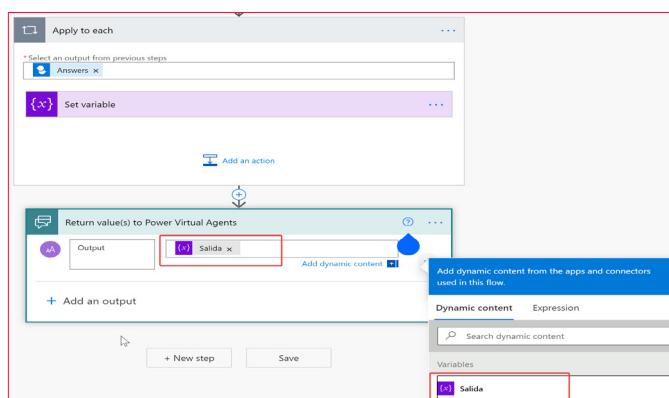


Imagen 12.- Añadiendo la respuesta al Power Virtual Agent.

- Guardamos nuestro Flow y retornamos a la página de Power Virtual Agents para agregarlo:

**"Power Automate quien será el responsable de comunicarse con el servicio de QnA Maker"**

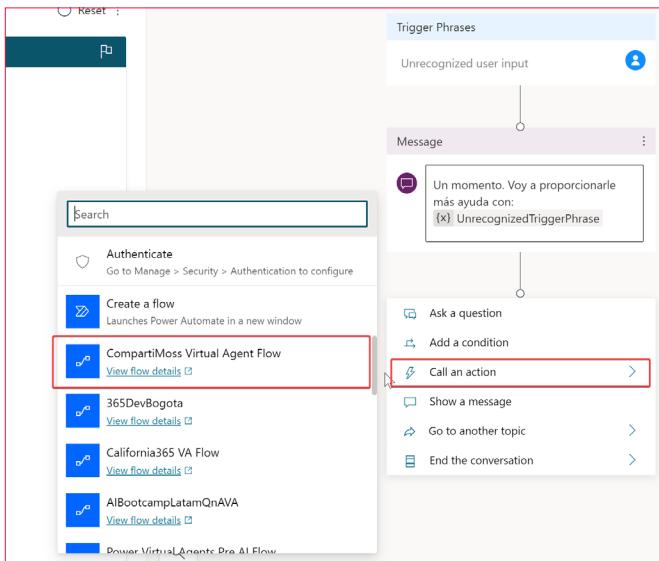


Imagen 13.- Agregando el Flow creado al Power Virtual Agent.

- Una agregado nuestro flujo seleccionamos la variable de entrada “bot.UnrecognizedTriggerPhrase”

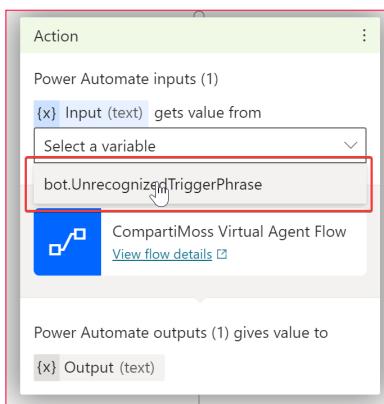


Imagen 14.- Seleccionando la variable de entrada.

- Agregamos un nuevo nodo para desplegar la respuesta seleccionando “Show a message”

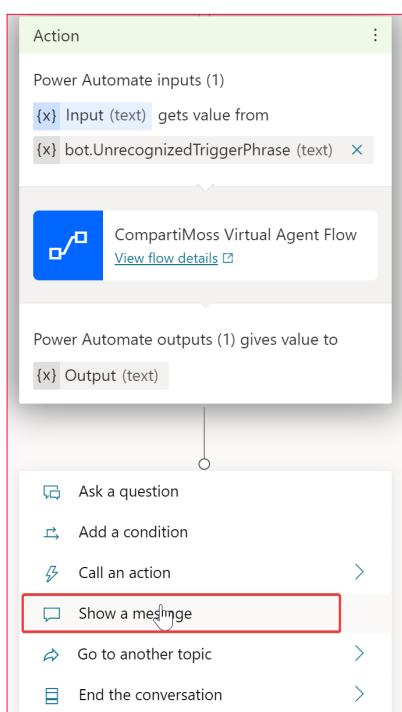


Imagen 15.- Nuevo nodo “Show a message” a añadir.

- Seleccionamos la variable Output:

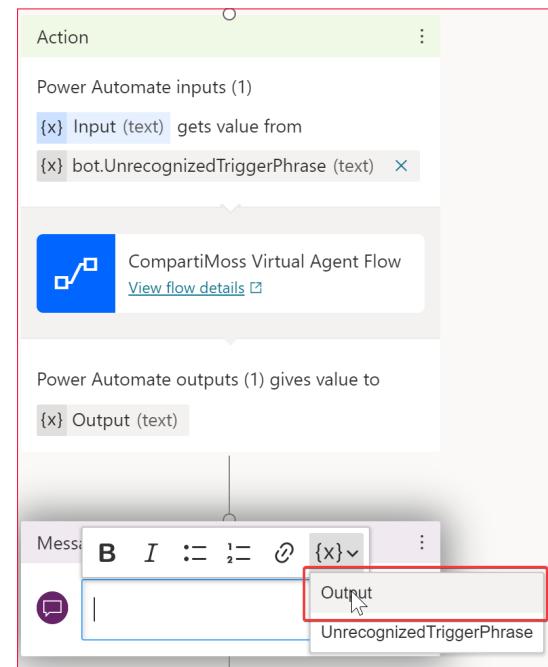


Imagen 16.- Selección de la variable Output.

- Finalmente guardamos y tenemos disponible nuestro Virtual Agent conectado con nuestra base de conocimiento de QnA Maker.

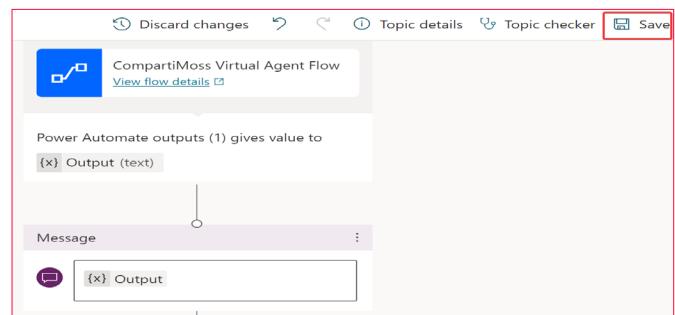


Imagen 17.- Guardando todo.

## Conclusión

El conectar un Power Virtual Agent con el servicio de QnA Maker nos permite rápidamente brindar un servicio de preguntas frecuentes, donde las opciones de base de conocimientos son infinitas.

### ALEX ROSTÁN

**Microsoft Business Applications MVP**  
**AI & Smarts Applications/ Cloud Architect (Azure, O365, Power Platform)**  
*Mail:rostanker@msn.com*  
*Twitter: @rostanker*  
*LinkedIn: <https://www.linkedin.com/in/alexrostan/>*

# Como versionar nuestro Producto de forma automática usando Git y Azure DevOps

Cuando estamos envueltos en el desarrollo de un producto/proyecto, una de las tareas que debemos llevar a cabo es controlar la versión en la que está el producto y que mejoras se incluyen en dicha versión. Esta labor muchas veces pensamos que puede ser una labor de marketing o del equipo de producto ¿no? ¿Pero quién tiene el control sobre lo que se está haciendo en el producto? El equipo técnico y hoy en día hay muchas herramientas para que todo ese trabajo que se realiza se pueda vincular con los casos de negocio, con las historias de usuario o simplemente con los errores/bugs que se producen a lo largo de un sprint (siempre que se siga una metodología Agile). A lo largo de este artículo vamos a ver cómo podemos automatizar todo este proceso haciendo uso de algunas herramientas de NPM o de Azure DevOps e incluirlas en nuestra rutina sin mucho esfuerzo.

## Un poco de contexto

Hoy en día muchos equipos de desarrollo se organizan en mayor o menor medida haciendo uso de una metodología agile, siguen/seguimos una aproximación a Scrum en la que el proyecto/producto que vamos a entregar lo separamos por sprints de una duración de 2/3/4 semanas dependiendo del equipo. En ese tiempo el equipo realiza las Tareas que se le han asignado en un sprint, estas tareas están vinculadas a una Historia de Usuario, y esta Historia de Usuario está asociada a una Feature del producto.

Siguiendo todo este flujo tenemos claro que si cada vez que nosotros como desarrolladores cogemos una tarea y la damos por finalizada la podemos hacer asociando nuestros “commits” con el desarrollo que hemos realizado. ¿Cómo podemos hacer esto? Si estamos usando Azure DevOps se puede poner como requisito que para que se apruebe un Pull Request sea obligatorio que se asocie una tarea asociada. El poner esta opción activa o no puede ser fruto de controversia entre el equipo (a mí personalmente hay veces que me ha gustado y otras no). Pero motivos por los que no nos gusta esta opción pueden ser : tener que acordarte de la tarea, buscar el nombre de la misma (un poco de pérdida de tiempo), si la tarea está completada no la puedes volver a seleccionar, que tu commit tenga código de más de una tarea o que por el contrario que no tengas la tarea que estás haciendo en tu board, todos estos motivos tienen un problema previo que no es otro que el encargado de llevar el Board, el contenido del Sprint no se

está haciendo con el rigor que corresponde y supone que tenemos que mejorar algunos del procesos que estamos llevando a cabo (algo que no corresponde con la finalidad de dicho artículo).

El tema es que en algún momento tenemos que asociar la resolución de una tarea/bug que tenemos asignado con la historia de usuario, este es el punto de partida que se debe de realizar sí o sí. Si el desarrollador no lo hace ya no tiene sentido el buscar como automatizar este proceso.

¿Pero cómo lo podemos hacer y de una forma sencilla para el desarrollador? En primer lugar, para facilitar al desarrollador esta tarea se debe de seguir una convención para llevar a cabo los “commits”. La podéis consultar en la siguiente web <https://www.conventionalcommits.org/en/v1.0.0/>

Porque usar esta convención:

- Poder generar changelog automáticos
- Poder versionar el producto de una forma natural
- Comunicación con naturalidad los cambios que se producen en el equipo de una forma transparente
- En proyectos open source hace mucho más fáciles que otras personas se puedan unir al proyecto y sea mucho más fácil empezar a colaborar

Cuáles son los aspectos principales que dice esta convención pues en primer lugar en el mensaje del commit pongamos como palabra de inicio:

- Feat: cuando vas a subir una feature
- Refactor!: en el momento que vas a realizar un refactor de una solución
- Docs: cuando se añade documentación
- Fix: cuando se corrige un bug

En base a esto un commit tendrá la siguiente estructura (esto es algo que el equipo debe de tener grabado a fuego para poder hacer todo esto):

```
<tipo> [optional scope]:<descripción  
[optional body]  
[optional footer(s)]
```

Por ejemplo, un Commit con un mensaje que hay un “breaking change”:

Feat: allow provided config object to extend other configs

**BREAKING CHANGE:** 'extends' key in config file is now used for extending other config files

Una vez, ya tenemos claro cuál es el objetivo que tenemos ahora vamos a ver una herramienta para poder automatizarlo en este caso Standard-Version <https://github.com/conventional-changelog/standard-version>

## Standar Version

Standar Version es una utilidad para hacer versionado usando semver, o versionado semántico en castellano, que nos da la generación de un Changelog de forma automática.

¿Como podemos automatizarlo? Vamos a explicar cuál es el flujo que debemos de hacer para poder tener el resultado esperado

- 1.- Obtiene toda la información del repositorio.
- 2.- “bumb” la versión basada en los commits.
- 3.- Genera un “Changelog” basado en nuestros commits.
- 4.- Crea un nuevo commit incluyendo la nueva versión y actualiza el CHANGELOG.
- 5.- Crea un nuevo “Tag” con el nuevo número de versión (opcional y dependiendo del sistema de branching que tenga el equipo, aunque siempre es bueno tener esto marcado para poder fixear errores en cada versión de una forma sencilla).

## Manos a la obra

Tenemos un proyecto en Azure DevOps, tenemos una API de serie que se nos crea al crear un proyecto en DotNet. Para ello bastaría con poner las siguientes líneas:

```
C:\git.adc\standar-version (master -> origin)
λ dotnet new webapi -o SimpleApi
The template "ASP.NET Core Web API" was created successfully.

Processing post-creation actions...
Running 'dotnet restore' on SimpleApi\SimpleApi.csproj...
Determinando los proyectos que se van a restaurar...
  Se ha restaurado C:\git.adc\standar-version\SimpleApi\SimpleApi.csproj (en 157 ms).
Restore succeeded.

C:\git.adc\standar-version (master -> origin)
λ cd SimpleApi\

C:\git.adc\standar-version\SimpleApi (master -> origin)
λ dotnet build
Microsoft (R) Build Engine versión 16.8.3+39993bd9d para .NET
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Determinando los proyectos que se van a restaurar...
Todos los proyectos están actualizados para la restauración.
SimpleApi -> C:\git.adc\standar-version\SimpleApi\bin\Debug\net5.0\SimpleApi.dll

Compilación correcta.
  0 Advertencia(s)
  0 Errores

Tiempo transcurrido 00:00:04.36

C:\git.adc\standar-version\SimpleApi (master -> origin)
λ dotnet run
Compilando...
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\git.adc\standar-version\SimpleApi
info: Microsoft.Hosting.Lifetime[0]
      Application is shutting down...
```

Imagen 1.- Crear proyecto.

Una vez ya tenemos el proyecto creado haremos una se-

rie de modificaciones para realizar una serie de commits y así que el CHANGELOG tenga algo de contenido y un poco de fundamento.

La primera pregunta que os estaréis haciendo seguro estamos implementando un proyecto de .NET y me estabas hablando de una herramienta NPM... ¿estamos haciendo lo correcto?? La respuesta naturalmente es que sí pero claro para que podamos obtener la información de los commits de nuestro código está claro que algo tenemos que hacer para poder vincularlo y por ese motivo crearemos dentro del repositorio un fichero .versionrc que tendrá el siguiente formato



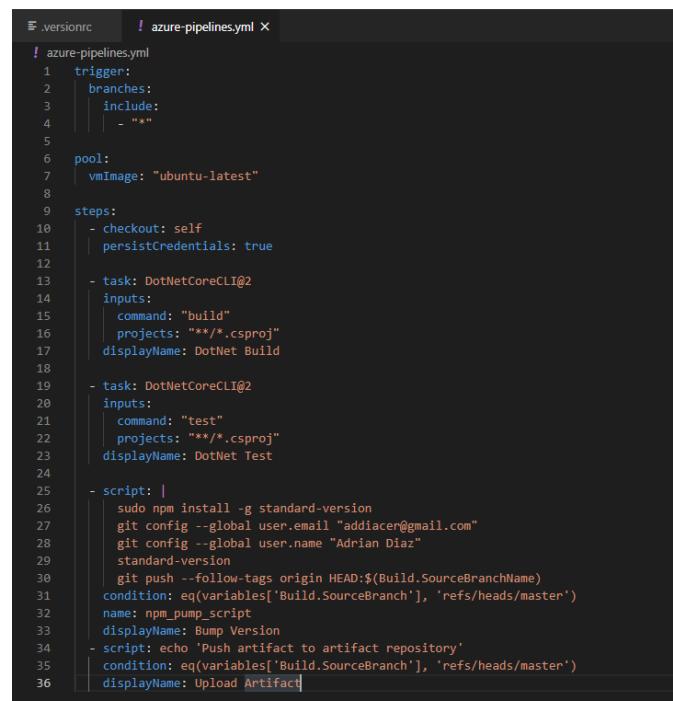
```
#!/versionrc
versionrc
1
{
  "releaseCommitMessageFormat": "chore(release): {{currentTag}} [skip ci]",
  "compareUrlFormat": "[({host})/{your.azure.repo.link}/branchCompare?baseVersion={Git{{previousTag}}}&targetVersion={Git{{currentTag}}}",
  "commitUrlFormat": "[({host})/{your.azure.repo.link}/commit/({hash})]"
}
```

Imagen 2.- Contenido del fichero .versionrc.

Donde:

- Host: La url de vuestro Azure DevOps por ejemplo compartimos.azuredevops.
- Your.azure.repo.link: La dirección donde esta vuestro repositorio en Azure DevOps.

Una vez ya tenemos donde vamos a obtener la configuración del versionando el siguiente paso es automatizar dicho paso en la propia Pipeline para ello nos vamos a crear un pipeline en yaml, para ello nos creamos un fichero azure-pipeline.yml que tendría el siguiente contenido:



```
azure-pipeline.yml
1 trigger:
2   branches:
3     include:
4       - "*"
5
6 pool:
7   vmImage: "ubuntu-latest"
8
9 steps:
10  - checkout: self
11    persistCredentials: true
12
13  - task: DotNetCoreCLI@2
14    inputs:
15      command: "build"
16      projects: "**/*.csproj"
17      displayName: DotNet Build
18
19  - task: DotNetCoreCLI@2
20    inputs:
21      command: "test"
22      projects: "**/*.csproj"
23      displayName: DotNet Test
24
25  - script: |
26    sudo npm install -g standard-version
27    git config --global user.email "addiacer@gmail.com"
28    git config --global user.name "Adrian Diaz"
29    standard-version
30    git push -f --follow-tags origin HEAD:${{Build.SourceBranchName}}
31    condition: eq(variables['Build.SourceBranch'], 'refs/heads/master')
32    name: npm_pump_script
33    displayName: Bump Version
34  - script: echo 'Push artifact to artifact repository'
35    condition: eq(variables['Build.SourceBranch'], 'refs/heads/master')
36    displayName: Upload Artifact
```

Imagen 3.- Contenido del fichero de pipeline.

Que hace esta Build, compila el proyecto, ejecuta los test, posteriormente instala standard versión, lo ejecuta para obtener todos los cambios y posteriormente genera una rama con el tag de dicha versión. Si ejecutáis esta build es necesario que el usuario que ejecuta la Build tenga permisos para contribución. Para dar permisos Settings > Repos > Repositorio > Permissions

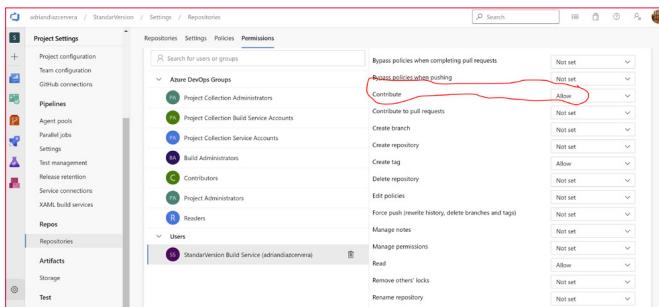


Imagen 4.- Tipo de Permiso a ejecutar

Si ejecutamos la Build, veremos que se ha creado un changelog dentro de nuestro repositorio donde podemos visualizar la información relativa a los commits que se han realizado. De la misma forma se ha creado un tag sobre la rama que acabamos de actualizar. Cada vez que realizamos un push sobre dicha rama este changelog se actualizará con los cambios y actualizará la versión relativa a los cambios que se han implementado en dicho repositorio. En el caso de que nuestro proyecto no sea de .NET sea una aplicación de JavaScript las modificaciones que tendríamos que hacer son mínimas ya que lo único que tendríamos que añadir es la compilación de dicho proyecto (o bien poner un comando dentro del package.json en el que se incluya todas estas opciones esa parte ya la dejamos para el descubrimiento del lector)

¿Ahora bien, en qué momento creamos un changelog? Desde mi punto de vista la generación de una actualización del changelog se debe de crear en el momento que tenemos la versión lista para que se produzca dicho cambio. Esto ya dependerá del propio equipo de desarrollo y de los acuerdos que tenga el mismo. Una práctica común es cuando se termina un sprint tener una nueva versión de dicho producto/proyecto y poder hacer una recapitulación de todo lo que se puede entregar a QA, Business/ Clientes con la liberación de dicha versión.

## ¿Qué problemas nos podemos encontrar con esta forma de versionar?

Dependemos de los commits que hace el desarrollador y todos sabemos que los desarrolladores no son máquinas (aunque algunos lo parezcan) y es posible que se equivoquen al hacer commit, que pongan mensajes que no correspondan con la funcionalidad. También el mensaje muchos equipos no le dan la importancia que toca el mensaje. Todos hemos visto comentarios con fix o poner . o incluso comentarios vacíos. Por lo que para realizar a poder tener un changelog, el equipo debe de tener un nivel de madurez bastante elevado y estar bastante comprometido con la forma en la que se trabaja.

Esta herramienta está muy bien y así podemos tener constancia de todas las features que vamos a liberar, pero cuando vamos a comunicar estos cambios si tenemos todo en Azure DevOps y usamos cualquier template agile. El desarrollador vincula su commit y su mensaje con la historia

que negocio ha redactado y tiene su "idioma". Por lo que de esta forma podemos por un lado tener controlado el versionado técnico haciendo uso de standard version pero también a través de la API de AzureDevOps podemos montarnos una relación entre estos commits y las historias de usuario que se han redactado.

Ahora bien, como lo podemos hacer de una forma fácil en AzureDevOps existe una Store donde nos podemos descargar aplicaciones (algunas de pago) para poder extender AzureDevOps. Para este caso en particular hemos tenido la enorme fortuna que un miembro de la comunidad Richard Fennell MVP en Developer Technologies (<https://github.com/rfennell>) que implementó una solución <https://marketplace.visualstudio.com/items?itemName=richardfennellIBM.BM-VSTS-XplatGenerateReleaseNotes>

## Generate Release Note

Como empezar a usarla, lo que tenemos que hacer es instalarla. Buscamos dicha App dentro de la Store y la instalamos.

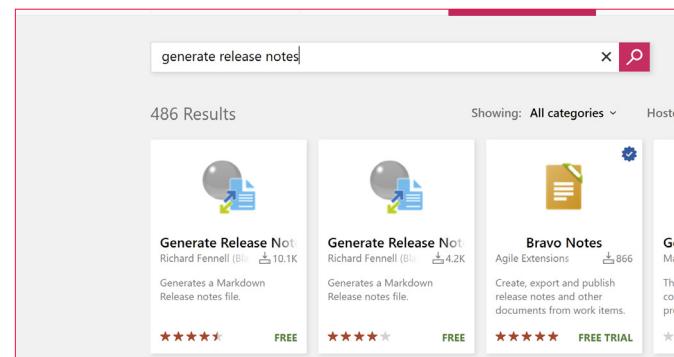


Imagen 5.- App Generate release notes.

Una vez ya la tenemos instalada el siguiente paso es añadirla en la Build, Release en el que queremos obtener el Changelog. En nuestra Build o Release lo que ahora necesitamos es saber qué información vamos a mostrar y como la mostramos. Para ello en la aplicación cada vez que la añadimos a una Pipeline tendremos que configurar el fichero de salida y el "template" con el que vamos a mostrar esta información. Este template siguiendo la documentación que se proporciona hace uso de handlebar para la sintaxis y nos queda tunearlo para adaptarlo a nuestras necesidades.

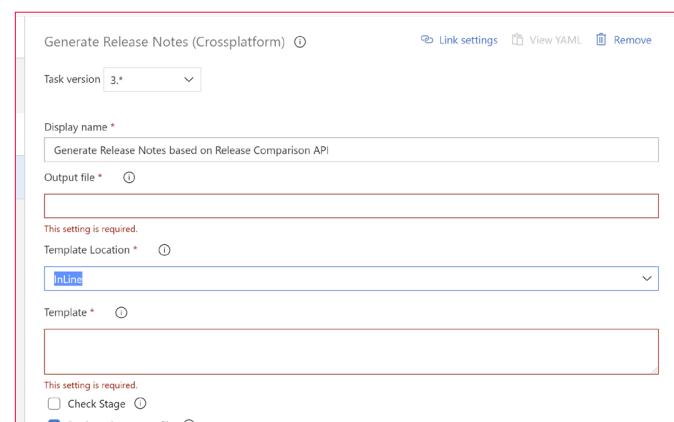


Imagen 6.- Configuraciones en la App de Generate Relase Notes.

## Conclusión

El versionado dentro de un producto/proyecto es algo muy importante y que todos los miembros de este se deben de hacer cargo. En proyectos a corto plazo muchas veces ni nos planteamos versionarlo, pensamos como solamente lo vamos a hacer una vez pues no tenemos que versionar nada. Otras veces empezamos a versionarlo, pero las prisas, las entregas y el día a día nos hace abandonarlo y no tenerlo actualizado. Sin embargo, sea la razón que se esgrima esto lo que se indica es el grado de madurez que tiene el equipo para hacer las cosas. En equipos que se tienen todo automatizado y funcionan como un reloj estas notas, changelog no son más que un mecanismo más que hace que nuestro software sea entendible por otros, ya sean developers o gente de negocio.

Como hemos visto en el artículo, no es algo muy costoso y que solo requiere de un poco de organización dentro del equipo y que tenga el mismo las ideas claras sobre lo que estamos haciendo. Este artículo es tanto valido para proyectos como para productos y creo que es importante para que el equipo se comprometa y sepa cuál ha sido el objetivo del sprint, lo que hemos entregado y la calidad de este. Esto es un paso más para la mejora continua del mismo y por dicho motivo se debe de evaluar incluirlo dentro del mismo.

---

### ADRIÁN DIAZ CERVERA

Technical Lead at SCRM Lidl Hub International

MVP Office Development

<http://theavenger.dev>

@AdrianDiaz81

# 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.



# Manejando listas de datos en PowerApps

Una de las tareas más frecuentes que nos podemos encontrar al desarrollar una App con PowerApps es el mostrar o manejar listas de datos ya sea desde un origen de base datos, ficheros, o listas de SharePoint por citar 3 ejemplos. PowerApps nos proporciona varios controles para que podamos mostrar esos datos al usuario y que se pueda realizar operaciones de inserción, modificación y borrado sobre elementos de esas listas.

Es este artículo veremos cómo implementar estas funcionalidades, qué controles tenemos para ello y como personalizarlos.

Para los ejemplos he escogido listas de SharePoint como orígenes de datos.

Estas listas son:

- EjemploMaestro:

Columnas	Tipos	Requerida
Columna (hacer clic para editar)		
Título	Una línea de texto	
Modificado	Fecha y hora	
Creado	Fecha y hora	
Creado por	Persona o grupo	
Modificado por	Persona o grupo	

Imagen 1.- Lista EjemploMaestro.

- EjemploDetalle:

Columnas	Tipos	Requerida
Columna (hacer clic para editar)		
Título	Una línea de texto	
Modificado	Fecha y hora	
Creado	Fecha y hora	
CampoNúmero	Número	
CampoElección	Elección	
CampoFecha	Fecha y hora	
CampoBusqueda	Búsqueda	
CampoBusquedaID	Búsqueda	
CampoPersona	Persona o grupo	
CampoMultilinea	Varias líneas de texto	
CampoSíNo	Sí o No	
Creado por	Persona o grupo	
Modificado por	Persona o grupo	

Imagen 2.- Lista EjemploDetalle.

- Continentes:

Columnas	Tipos	Requerida
Columna (hacer clic para editar)		
Título	Una línea de texto	
Modificado	Fecha y hora	
Creado	Fecha y hora	
Continente	Persona o grupo	
ContinenteID	Persona o grupo	

Imagen 3.- Lista Continentes.

- Países:

Columnas	Tipos	Requerida
Columna (hacer clic para editar)		
Título	Una línea de texto	
Continente	Búsqueda	
ContinenteID	Búsqueda	
Modificado	Fecha y hora	
Creado	Fecha y hora	
Creado por	Persona o grupo	
Modificado por	Persona o grupo	

Imagen 4.- Lista Países.

## Controles existentes que permiten manejar listas/tablas

### DataTable

Los DataTable son controles que permiten ver un origen de datos en formato de grid o tabla. Es un control muy sencillo que NO permite editar los datos ni personalizarlo en exceso, por ello su uso está limitado a situaciones donde solo requiramos mostrar los datos sin demasiadas "florituras" y sin poder editar esos datos sobre el mismo grid (modo espartano ON jauhh! jauhh! jauhh!).

Para utilizar un DataGrid simplemente lo insertamos en la pantalla y escogemos el origen de datos que queremos mostrar.

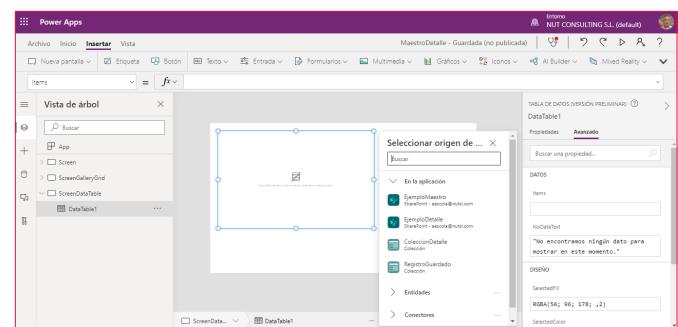


Imagen 5.- Control de tipo DataTable.

Tras escoger el origen de datos podremos escoger qué campos queremos mostrar en el DataTable.

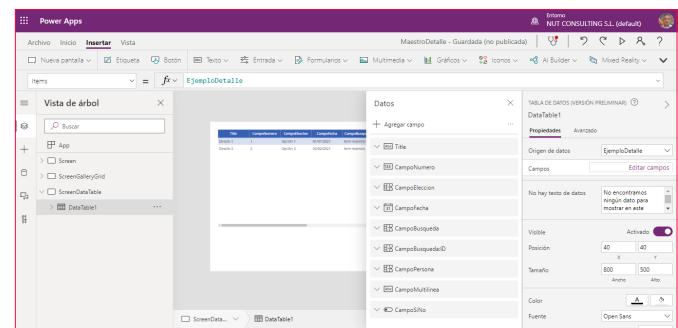


Imagen 6.- Personalizando el DataTable.

La propiedad Items es la que enlaza el origen de datos con el control. En el ejemplo es el origen de datos EjemploDetalle

La propiedad Selected del DataTable es la que nos permite acceder al registro escogido por el usuario.

Es importante destacar que, aunque no escogamos para mostrar todos los campos del origen de datos, sí que tendremos disponible el registro completo cuando referenciamos la propiedad DataTable.Selected sin más que añadiendo el nombre del campo a la expresión.

En el ejemplo tenemos el origen de datos EjemploDetalle enlazado al dataTable1. Para obtener el valor del campo CampoNumero del registro seleccionado por el usuario simplemente tendríamos que invocar DataTable1.Selected.CampoNumero.

### "Los DataTable son controles que permiten ver un origen de datos en formato de grid o tabla"

En cuanto a las personalizaciones, son escasas. Podemos cambiar el nombre del encabezado de las columnas desbloqueando la edición de esta y modificando la propiedad DisplayDataField.

NO es posible modificar, por ejemplo, la alineación de un campo, la fuente de la columna o su color de fondo.

Microsoft ya ha anunciado para este año la liberación de un nuevo DataTable con más funcionalidades.

## Gallery

Los Gallery permiten mostrar uno o varios campos de un origen de datos en un formato de lista horizontal o vertical. Están compuestos por uno o varios controles para mostrar los valores de los registros y de un Template que es la "caja" que contiene estos controles, este Template no se ve en el control, pero sí que hay propiedades referidas al Template.

Lo interesante del control Gallery es que tiene muchas más opciones de personalización que un DataTable, por lo que es más interesante de utilizar cuando necesitamos un aspecto visual enriquecido.

Para insertar un Gallery lo escogemos sin más en el menú de controles y escogemos que plantilla vamos a usar.

Las plantillas por defecto incluyen uno o dos campos de datos, insertar una imagen (interesante opción si mostramos una lista de personas y tenemos disponible su foto).



Imagen 7.- Control de tipo Gallery.

Una vez insertado, podremos añadir más controles dentro

del gallery, por ejemplo, un control de texto.

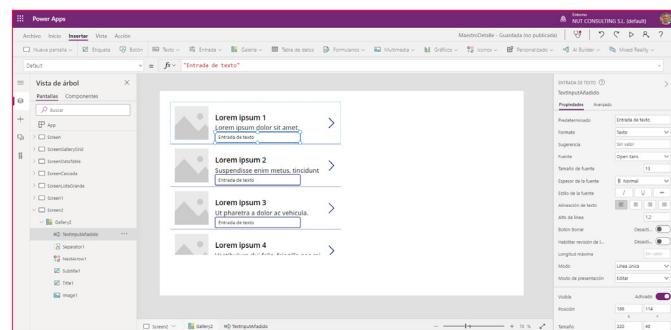


Imagen 8.- Añadiendo un control de texto al control Gallery.

Igual que con los DataTable enlazamos el control con el origen de datos mediante la propiedad Items.

El control Gallery dispone de la propiedad ThisItem, que es de un tipo complejo, que hace referencia al registro actual dentro de cada elemento mostrado por el Gallery.

Con esta propiedad es con la que podremos realizar operaciones en los controles del Gallery. Podremos establecer la propiedad Default de un control como ThisItem.NombreDeCampo con lo que en cada elemento del Gallery se mostrará el valor de dicho campo.

Es importante señalar que esta propiedad ThisItem no se puede utilizar fuera del control Gallery.

Si queremos acceder al registro seleccionado por el usuario, al igual que en el DataTable, disponemos de la propiedad Selected.

## Form

Un Form es un control que nos permite la visualización, edición o inserción de registros en una fuente de datos. Se compone de uno o varios controles DataCard que representan cada uno de los campos que queremos gestionar. A su vez un DataCard se compone de varios controles que nos permitirán ver o editar el campo.

Los Form son la forma más sencilla de gestionar registros de una fuente de datos y deberían ser nuestra primera opción para ello.

Para insertar un Form solo hay que seleccionarlo en el menú de controles y escoger el origen de datos (o fijarlo en la propiedad DataSource)

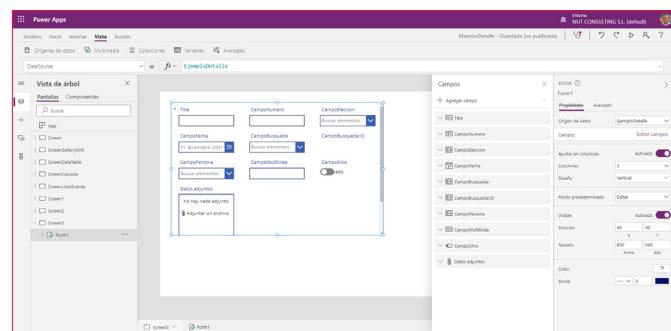


Imagen 9.- Control de tipo Form.

En el ejemplo he añadido un formulario en modo de edición con el origen de datos EjemploDetalle. PowerApps genera los datacards y los controles adecuados para cada tipo de dato del origen ya sean controles de texto, listas desplegables, casillas o el control de adjuntos.

En el caso de campos obligatorios (Title en el ejemplo) se genera la funcionalidad adecuada para que en caso de intentar guardar un nuevo registro sin ese campo se avise al usuario y no se guarde el registro.

Los controles contenidos en el formulario están bloqueados para su edición, sin embargo, es posible desbloquearlos para cambiarlos de acuerdo con nuestras necesidades.

En el ejemplo podríamos cambiar la propiedad DisplayName de los datacard que, por defecto, muestra el nombre del campo del origen de datos para mostrar un nombre distinto (título en lugar de Title) o para incluir espacios en blanco, tildes... (manías de un servidor, los nombres de las columnas en listas de SharePoint siempre las pongo sin espacios ni tildes).

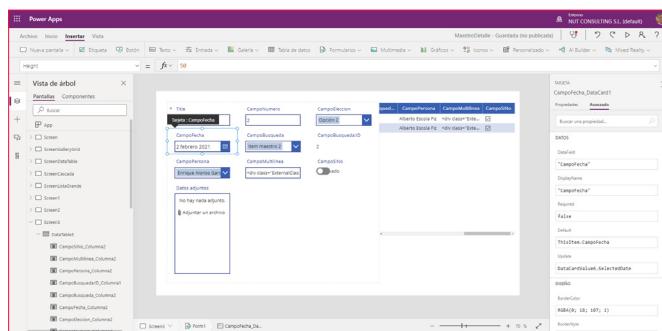


Imagen 10.- Cambiando los DisplayName en el control Form.

Algunas propiedades que usaremos más habitualmente en este control son:

- Mode: Indica si el formulario está en modo de edición, nuevo o vista.
- Item: Fijando esta propiedad con un registro el formulario muestra la información de dicho registro.
- Unsaved: Esta propiedad booleana indica si se ha modificado los datos del registro que se ha cargado.
- LastSubmit: nos proporciona el último registro que se ha guardado. Muy útil si necesitamos obtener el ID del registro que se ha guardado ya que ese dato se habrá generado automáticamente. Solo hay que invocar Form1.LastSubmit.ID.
- OnReset, OnFailure, OnSuccess: propiedades de comportamiento (eventos) que se ejecutan al resetear, guardar con fallo o guardar con éxito.

Para cambiar el modo del formulario disponemos de las funciones NewForm(NombredelForm), EditForm(NombredelForm), ViewForm(NombredelForm).

Por último, para guardar un Form disponemos de SubmitForm(NombredelForm).

En la sección siguiente veremos un ejemplo de uso de todas estas funciones para implementar operaciones de creación, adición y borrado de registros.

Podemos personalizar el Form haciendo que un DataCard sea más ancho, cambiando su fuente, tamaño, poner un color de fondo en función de una condición... simplemente desbloqueando el dataCard y modificando sus propiedades o las propiedades de los controles que contiene.

En muchas de estas propiedades de los controles se hará referencia a la propiedad del Datacard mediante el uso de Parent. Por mencionar un caso un control TextInput dentro de un datacard tendrá fijada su propiedad Default como Parent.Default

## Como realizar operaciones CRUD

El escenario es el siguiente: vamos a tener una pantalla donde podremos manipular el origen de datos Países. Insertamos los siguientes controles:

- DataTable (DataTablePaises).
- Form (FormPais).
- Icono IconNuevo.
- Icono IconBorrar.
- Icono IconGuardar.

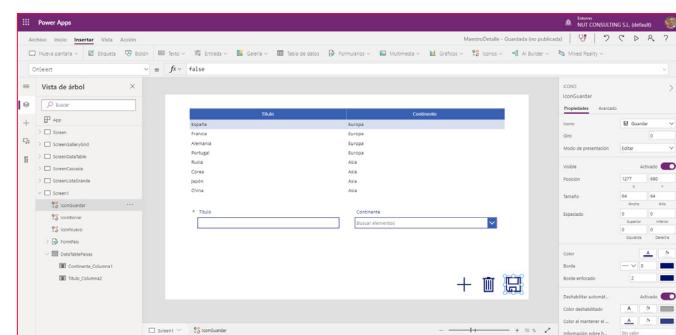


Imagen 11.- Pantalla para manipular el origen de datos Países.

El DataTable lo enlazamos al origen de datos Países y escogemos mostrar los campos título y continente.

El Formulario lo enlazamos al mismo origen de datos fijando su propiedad DataSource al origen de datos Países. PowerApps se da cuenta que el campo Continente es un campo de búsqueda y nos crea automáticamente el control adecuado para manejar este tipo de campo.

Fijamos la propiedad Item del Form con DataTablePaises.Selected. Con esto cuando cambiamos el item seleccionado en el Datatable nos lleva el registro escogido al Formulario.

### Añadir un registro

Para añadir un nuevo registro: Invocaremos desde la propiedad Selected del icono IconNuevo la función NewForm(FormPais) lo que nos limpia todos los campos del formulario y lo pone en modo New para editar los campos que contiene.

### Borrar un registro

Para borrar un registro invocaremos en la propiedad Selected del icono IconBorrar la función RemoveIf(Paises,{ID: DataTablePaises.Selected.ID}).

## Guardar un registro

Invocaremos desde la propiedad Selected del icono Icon-Guardar la función UpdateForm(FormPais).

**"PowerApps intenta delegar esas condiciones o consulta en la fuente de datos"**

Hay situaciones donde tenemos que crear o modificar un registro en una lista que no mostramos directamente al usuario, para ello podemos utilizar la función Patch. Siguiendo con el ejemplo anterior la sintaxis seria:

- Para crear un registro nuevo: Patch(Paises;Defaults(Paises);{Titulo:"nuevo pais"})
- Para modificar un registro que ya existe: Patch(Paises;Lookup(Paises;ID=ID\_DEL\_REGISTRO);{Titulo:"-nuevo pais"})

## Filtrado de datos

Normalmente tendremos que filtrar el origen de datos de acuerdo con reglas de negocio. Para Filtrar un origen de datos disponemos de 3 funciones:

- Filter: Busca todos los registros que cumplen con una condición.
- Search: Busca todos los registros que contienen una cadena de texto en una de sus columnas.
- Lookup: busca el primer registro que cumple con una condición.

Si en un control DataTable o Gallery fijamos la propiedad Items con una función de filtrado y en la condición de filtrado incluimos el valor de otro control haremos que el usuario sea capaz de realizar un filtrado de los elementos que desea ver.

Por ejemplo, disponemos de un Gallery que mostrará elementos del origen de datos EjemploDetalle y añadimos un control DatePicker. En la propiedad Items del Gallery ponemos la expresión:

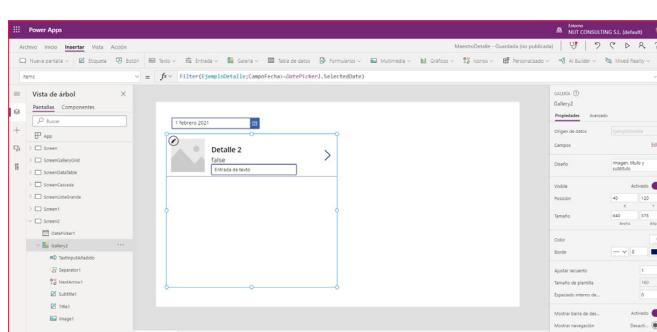


Imagen 12.- Ejemplo de filtrado de datos.

Con esto conseguimos que el Gallery muestre solamente los registros de EjemploDetalle cuyo campo CampoFecha

tenga valores mayores o iguales que la fecha seleccionada en el control DatePicker1.

## Limitaciones en el filtrado

Cuando hacemos un filtrado, PowerApps intenta delegar esas condiciones o consulta en la fuente de datos, lo que es más eficiente. Si no se puede delegar y el conjunto de datos es muy grande podemos encontrarnos con que el filtrado no funciona bien al no acceder al todo el conjunto de datos ya que existe un límite de 500 elementos recuperados, es posible ampliar este límite hasta 2000 editando la configuración avanzada de la App.

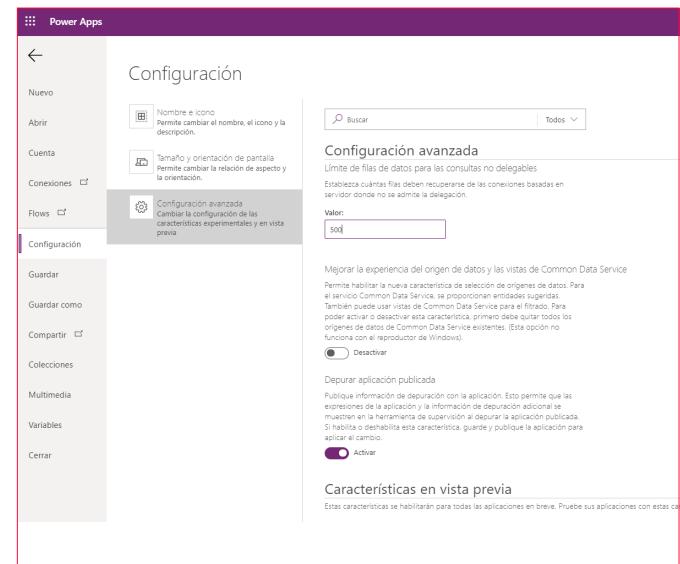


Imagen 13.- Configuración avanzada de una Power App.

¿Es posible sortear esta limitación? Que cosas pregunto iclaro que sí! ...pero esa es otra historia y debe ser contada en otra ocasión.

## Manejo de estructuras de maestro/detalle o controles en cascada

Cuando tenemos varias listas relacionadas entre ellas es frecuente ir filtrando listas de acuerdo con el elemento de nivel superior escogido. Por ejemplo, tenemos una lista de Continentes y una de Países donde tenemos una columna de tipo búsqueda con el continente donde está ese país.

Para tener controles en cascada hemos añadido un control de lista enlazada al origen de datos Continente y un DataTable a la lista de Paises.

Para que el DataTable solo muestre los países cuyo continente sea el escogido por el usuario en la lista lo que debemos hacer es filtrar el origen de datos del DataTable. Para ello modificamos su propiedad Items con el filtro del campo búsqueda. Recuerda que el campo Continente en la lista países es un campo de búsqueda enlazado a la lista Continentes

```
Filter(Paises:'Continente':ID'.Id=Dropdown2.Selected.ID)
```

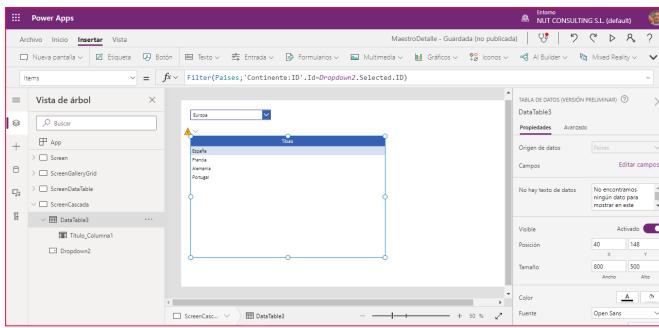


Imagen 14.- Configurando campos en cascada.

## Manejo de tipos complejos de datos

Los tipos complejos de datos requieren un tratamiento adicional ya que PowerApps no los maneja directamente. Si nos centramos en tipos de datos de listas de SharePoint esos tipos son:

- Elección.
- Búsqueda.
- Persona.

Estos tipos de datos tienen una estructura interna con varios campos que SharePoint maneja internamente pero que nosotros deberemos manejar explícitamente desde PowerApps. Por ejemplo, para crear un registro en un origen de datos con campos complejos con la función Patch debemos utilizar las siguientes estructuras:

Campo de tipo búsqueda:

```
{"@odata.type": "#Microsoft.Azure.Connectors.SharePoint.SPListExpandedReference", "Id:<>valor del Id>, Value:<>texto>"}
```

Campo de tipo persona:

```
{"@odata.type": "#Microsoft.Azure.Connectors.SharePoint.SPListExpandedUser", Claims:"i:0#.f!membership" & "<>correo del usuario>", DisplayName:<>Nombre del usuario>"}
```

Campo de tipo elección:

```
{"@odata.type": "#Microsoft.Azure.Connectors.SharePoint.SPListExpandedReference", Id:<>ID>, Value:<>Texto de la opción>"}
```

El ejemplo completo de la sintaxis para crear un nuevo registro en el origen de datos Paises y teniendo la lista de continentes en un control DropDownList llamado ListaContinentes:

```
Patch(Paises;Defaults(Paises);{Title:"Uruguay";Continente:{@odata.type": "#Microsoft.Azure.Connectors.SharePoint.
```

```
SPListExpandedReference"};Id: ListaContinentes.Selected.ID;Value: ListaContinentes.Selected.Título}})
```

Este ejemplo es a título ilustrativo, ya vimos en la sección anterior que con el uso de formularios el guardado es mucho más sencillo. Sin embargo, hay ocasiones en las que deberemos crear registros de orígenes de datos SharePoint que no están enlazados a Forms por lo que deberemos recurrir a la función Patch y al manejo de estos campos complejos.

## Uso del control Gallery como grid

Un uso interesante de Gallery es “tunearlo” para que parezca un grid de datos para, por ejemplo, mejorar el aspecto.

Para ello tendremos que realizar lo siguiente:

- Añadir un Gallery vertical y establecer el origen de datos

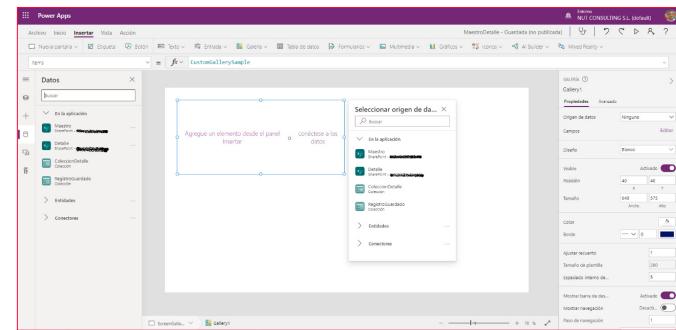


Imagen 15.- Añadiendo un Gallery vertical.

- Para cada campo que queramos tener en nuestro “GalleryGrid”, añadir un control de entrada de texto, casi lla, etc.

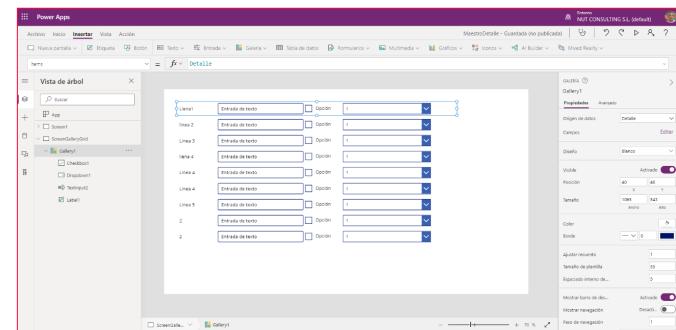


Imagen 16.- Controles añadidos por cada campo.

Disponemos los controles en horizontal y ajustamos el alto del elemento dentro del Gallery.

Para enlazar cada control fijamos su propiedad Default al valor ThisItem.NombreDelCampo

Añadimos un icono que nos servirá para que se guarde la línea.

Con esto ya podremos ver que el Gallery muestra los registros que tengamos.

Podrías pensar, amable lector/a, “a ver Alberto, alma de cántaro...en vez de tanto rollo de controles inserta un Form

dentro del "Gallery" ibien pensado! y es lo que pensé yo... pero PowerApps de momento no permite insertar un Form dentro de un Gallery.

Curiosamente sí que es posible insertar un Gallery dentro de un Gallery pero sin emocionarse, un tercer nivel de Gallery anidado ya no es posible. En la siguiente sección veremos cómo hacer esto.

Sigamos entonces. Si queremos tener encabezados, una forma fácil es añadir labels fuera del Gallery, también podemos darle un ancho al borde del Gallery con lo que conseguimos un aspecto más de tabla o grid.

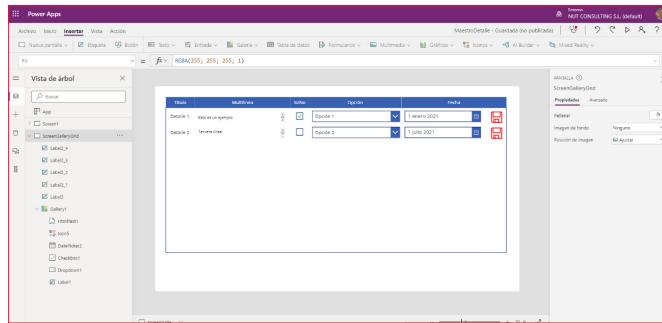
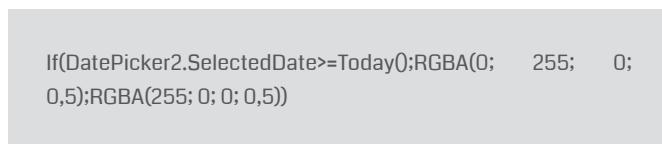
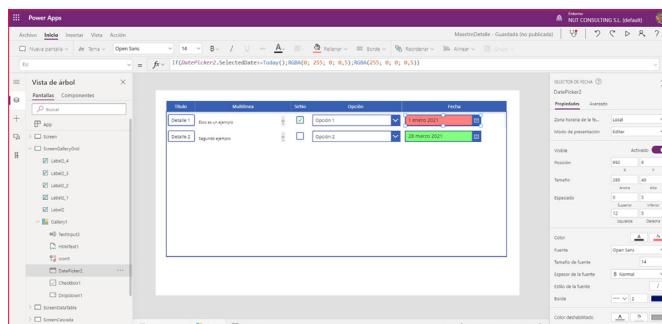


Imagen 17.- Añadiendo encabezados.

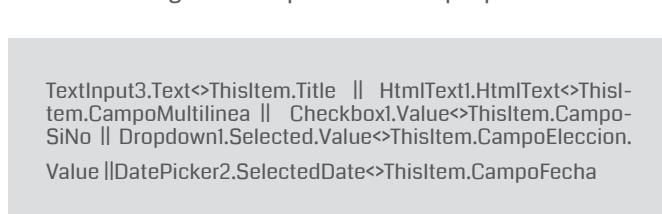
Ahora es cuando podemos jugar con las personalizaciones de los controles. Por ejemplo, en la propiedad Fill del DatePicker2 añadimos la siguiente expresión:



Con esto conseguimos que el color de relleno sea rojo si la fecha del registro está en el pasado y en verde si está en el futuro (Marty, Doc esto no os aplica a vosotros).



Para ocultar el icono de guardar si no se ha cambiado nada añadimos la siguiente expresión en su propiedad Visible:



Si tenemos muchos campos esta condición se nos hará un tanto larga.

Como último paso (porque se me ha olvidado ponerlo antes), configuraremos la propiedad TemplateFill del Gallery para que sea gris en el ítem seleccionado por el usuario. Para ello Usamos la propiedad IsSelected de ThisItem de la siguiente forma:

If(ThisItem.IsSelected;RGB(0; 0; 0; 0,25);RGB(0; 0; 0; 0))

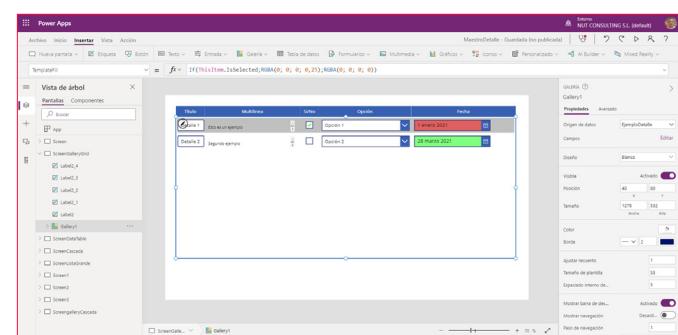


Imagen 19.- Personalizando el registro seleccionado por el usuario.

Con esto tenemos nuestro “tuneado”, solo faltaría añadir la funcionalidad de guardado en la propiedad OnSelect del ícono utilizando la función Patch.

Una diferencia importante en cuanto a usabilidad de nuestro “Gallery/grid” respecto a controles grid tradicionales es que los datos no se guardan automáticamente al cambiar de campo o fila. Si es posible guardar un campo cuando cambie su valor mediante la propiedad de comportamiento OnChange, pero los tiempos de respuesta pueden no ser buenos para la experiencia de usuario.

No es posible (al menos que este autor haya encontrado) guardar un registro automáticamente al cambiar de fila ya que, aunque tenemos una propiedad de comportamiento Selected, no hay una propiedad de comportamiento que nos permita realizar una acción (guardar el registro) al salir de una fila.

## Gallery en cascada

Podemos tener un Gallery dentro de otro de forma que tengamos una especie de control en árbol. Veamos como lo hacemos.

Vamos a hacer un Gallery anidado que muestre en su primer nivel los elementos del origen de datos Continentes y para cada uno de ellos los elementos del origen de datos Países que pertenecen a dicho continente. Recuerda que en la lista de SharePoint Países tenemos un campo Lookup enlazado a la lista Continentes.

En una pantalla insertamos:

- Botón ExpandirTodos.
- Botón ContraerTodos.
- Gallery de tipo Flexible (GalleryContinentesFlexible):
  - Label (labelNombreContinente).
  - Icono de flecha a la derecha (Ico.nDerecha).

- Icôno de flecha abajo (IconAbajo).
- Gallery (GalleryPaises):
  - Label (LabelNombrePais).
  - Rectángulo (Separador).

Colocamos los controles para que no se amontonen.

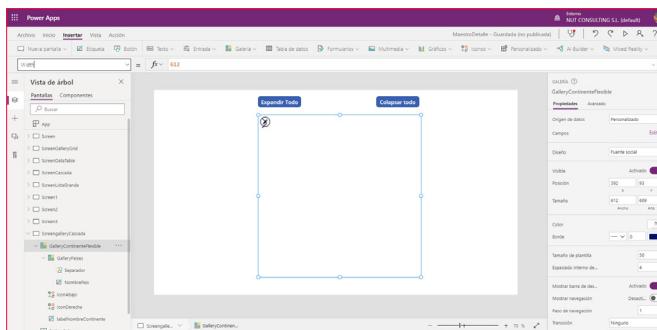


Imagen 20.- Botones para la funcionalidad de galería en cascada.

Fijamos las propiedades:

CONTROL	PROPIEDAD	VALOR
GalleryContinentesFlexible	Items	Continentes
GalleryContinentesFlexible	TemplateFill	If(ThisItem.IsSelected;RGB(0, 0, 0, 0.25);RGB(0, 0, 0, 0))
labelNombreContinente	Text	ThisItem.Título

Con esto ya tenemos la lista de continentes:

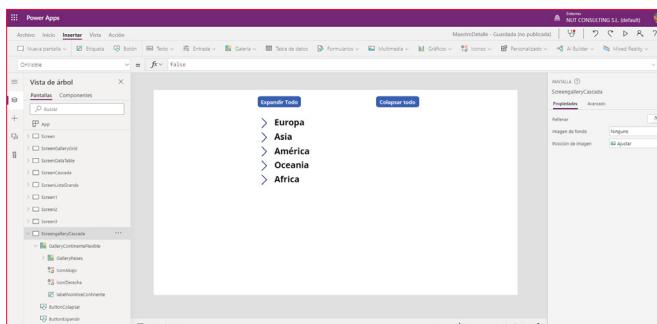


Imagen 21.- Lista Continentes configurada.

Ahora vamos con el Gallery de países, fijamos las siguientes propiedades:

CONTROL	PROPIEDAD	VALOR
GalleryPaises	Items	Filter(Paises,'Continente':ID'.Id=ThisItem.ID)
labelNombrePais	Text	ThisItem.Título

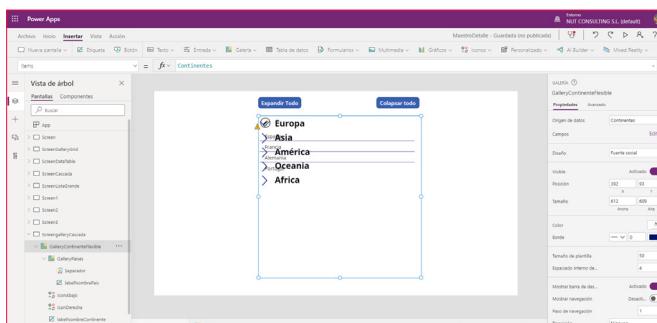


Imagen 22.- Configuración de los Paises.

Fácil ¿no?

Ahora nos venimos arriba y vamos a añadir funcionalidad para expandir/contraer toda la lista de continentes y expandir/contraer un continente.

Para ello Crearemos una variable \_Expandir que tomará valores 0 (expandir todo), 1 (contraer todo) y 2 (expandir elemento actual).

CONTROL	PROPIEDAD	VALOR
BotonExpandir	OnSelect	Set(_Expandir;0)
BotonColapsar	OnSelect	Set(_Expandir;1)
IconAbajo	OnSelect	Set(_Expandir;1)
IconDerecha	OnSelect	Set(_Expandir;2)
IconAbajo	Visible	(ThisItem.IsSelected && _Expandir=2)    _Expandir=0
IconDerecha	Visible	(!ThisItem.IsSelected && _Expandir=2)    _Expandir=1

Y ahora es donde hacemos que GalleryPaises se vea o no jugando con su propiedad Height de la siguiente forma:

```
Switch(_Expandir;
//expandir todo
0:CountRows(GalleryPaises.AllItems)* Self.TemplateHeight;
//contraer todo
1:0;

//expandir seleccionado
2:If(ThisItem.IsSelected;CountRows(GalleryPaises.AllItems)* Self.TemplateHeight;0)
)
```

Y ale, extra point, modificamos la propiedad Text de labelNombreContinente

ThisItem.Título & "(" & CountRows(GalleryPaises.AllItems) & ")"

Así vemos el número de países de cada continente:

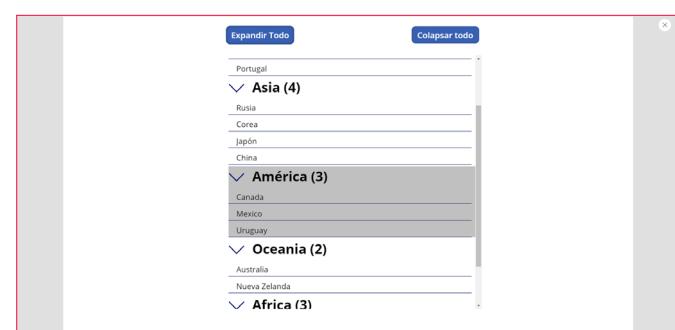


Imagen 23.- Visualización de los países por Continente.

Tenemos una limitación de este control compuesto: perdemos la posibilidad de seleccionar un elemento del Gallery hijo y por lo tanto la propiedad Selected no contendrá ningún valor y el evento OnSelect no se producirá nunca.

## ¿Podemos hacer algo?

Resulta que sí: añadimos un icono IconEditar dentro del GalleryPaises y resulta que si podemos hacer click sobre el

ícono. Fijamos su propiedad OnSelect.

```
Set(_ItemPais;ThisItem.ID)
Añadimos un Formulario y fijamos su propiedad Items
LookUp(Paises;ID=_ItemPais)
```

¡Et voilà!

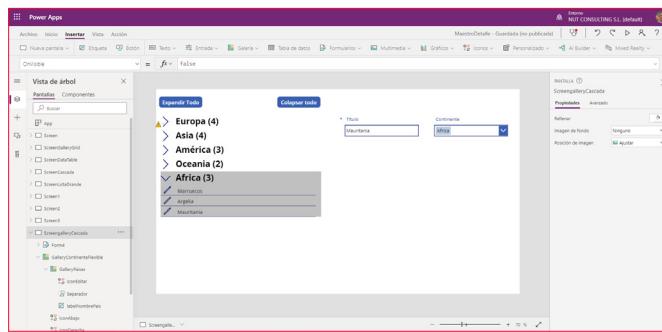


Imagen 24.- Iconos añadidos para los países.

Tarea para casa: añade tú la funcionalidad de guardar el formulario o añadir un nuevo país

## Conclusiones

PowerApps nos ofrece un potente mecanismo de conexión

a fuentes de datos y varios controles para manejar esos campos y listas de una forma muy sencilla, encapsulando los orígenes para mostrarlos de forma homogénea.

Aunque los controles pueden parecer sencillos, con un poco de trabajo podemos personalizarlos para enriquecer el aspecto y funcionalidad de nuestras Apps sin tener que recurrir a técnicas más complejas (aunque más potentes) como el desarrollo de componentes.

Debemos tener claras las limitaciones tanto en orígenes de datos como en funcionalidad de controles a la hora de diseñar nuestra App.

PowerApps es una plataforma en continuo desarrollo por parte de Microsoft e irá incorporando funcionalidades continuamente. Existen varios canales donde podemos lanzar ideas para estas nuevas funcionalidades o apoyar esas ideas.

¡Muchas gracias por leer mi artículo!

---

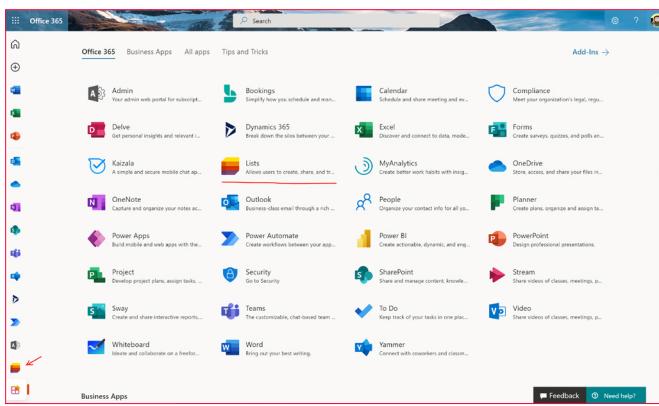
**ALBERTO ESCOLA**

# Aplicación de Microsoft Lists en Microsoft 365 o Listas de SharePoint 2.0

En Ignite 2020 Microsoft introdujo lo que denominó una nueva aplicación de 1<sup>a</sup> clase en Microsoft 365: Microsoft Lists. Sin embargo, los que conocemos y trabajamos con SharePoint desde hace años sabemos que las listas de Microsoft Lists vienen a ser listas de SharePoint con super-poderes. En este artículo y siguientes iré desgranando el potencial de Microsoft Lists que refleja la apuesta clara de Microsoft y del equipo de SharePoint por modernizar las listas tradicionales de SharePoint para facilitar construir rápidamente aplicaciones de negocio sencillas sin apenas desarrollo.

## Acceso a Microsoft Lists

Microsoft Lists está disponible como una App de Microsoft 365 que podemos acceder desde el lanzador de aplicaciones de Microsoft 365 o bien desde la página donde tenemos ubicadas todas las Apps que como usuarios de la plataforma tenemos disponibles.



Cualquiera de las dos opciones nos lleva a la App de Lists que viene a ser una página hospedada en el OneDrive For Business (ODFB) del usuario como se podrá comprobar tras una inspección rápida de la URL que se abre. Desde la App o Home de Microsoft Lists el usuario:

- Puede crear nuevas listas de Microsoft Lists.
- Visualiza las listas que ha marcado como favoritas.
- Visualiza las listas recientes a las que ha accedido.

*Las listas de Microsoft Lists pueden ser de dos tipos:*

- ✓ *Listas personales que se crean en el ODFB del usuario.*

- ✓ *Listas compartidas que se crean en sitios de SharePoint Online (SPO) a los que el usuario y otros usuarios tienen acceso.*

- Puede buscar una lista o contenido de una lista utilizando la caja de búsqueda.

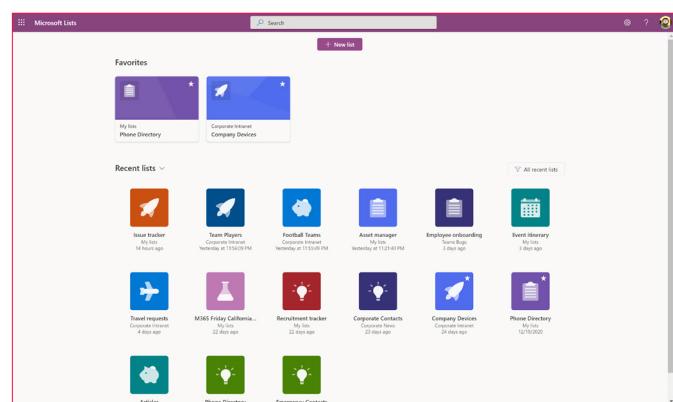


Imagen 2.- App de Microsoft Lists.

Para una lista ya existente, el usuario puede realizar las siguientes acciones a través de los "...":

- Personalizar, lo que le permite cambiar el ícono y color que se utiliza para identificar la lista.
- Compartir, lo que le permite dar acceso directo a la lista a otros usuarios. Esta opción es especialmente interesante cuando queremos dar acceso a nuestras listas personales a otros usuarios.
- Eliminar de listas reciente, para eliminar la lista del listado de listas recientes que se muestra.

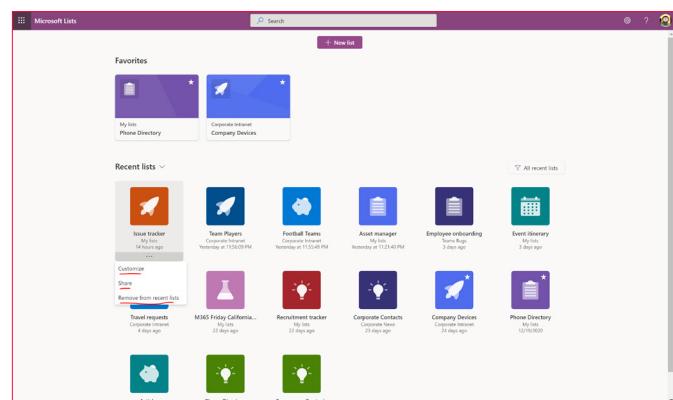


Imagen 3.- Opciones disponibles a nivel de lista.

Por ejemplo, la opción de personalizar muestra una ventana que permite cambiar el ícono y color de la lista tal y como se ha indicado.

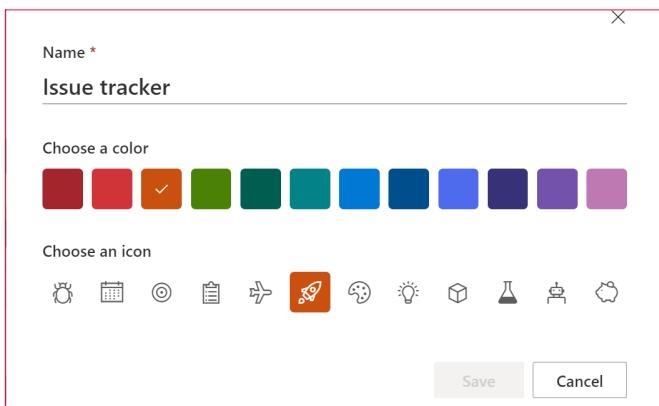


Imagen 4.- Opciones de personalización de una lista desde la App de Microsoft Lists.

## Creando listas en Microsoft Lists

Para crear una lista de Microsoft Lists tenemos tres posibilidades:

- Crear la lista desde la App de Microsoft Lists disponible en el ODFB del usuario.
- Crear la lista en un sitio de SharePoint concreto.
- Crear una lista desde la App de Microsoft Lists disponible para Microsoft Teams.

En esta sección vamos a revisar la primera posibilidad de creación de una lista:

- Desde la App de Microsoft Lists, simplemente hacemos clic en “New” (Nuevo) de forma que se muestre una ventana modal con las distintas posibilidades de creación de una lista:
  - Crear una lista en blanco.
  - Crear una lista a partir de una hoja Excel que deberemos tener disponible en nuestro ODFB o bien cargarla en el momento de creación de la lista.
  - Crear una lista a partir de una lista existente.
  - Crear una lista a partir de una plantilla disponible.
- En nuestro caso, elegimos la opción de crear la lista a partir de una plantilla existente. Microsoft proporciona un total de 8 plantillas de serie y por el momento no proporciona la posibilidad de que podamos añadir nuestras propias plantillas de lista (Aunque a futuro tendremos dicha posibilidad).

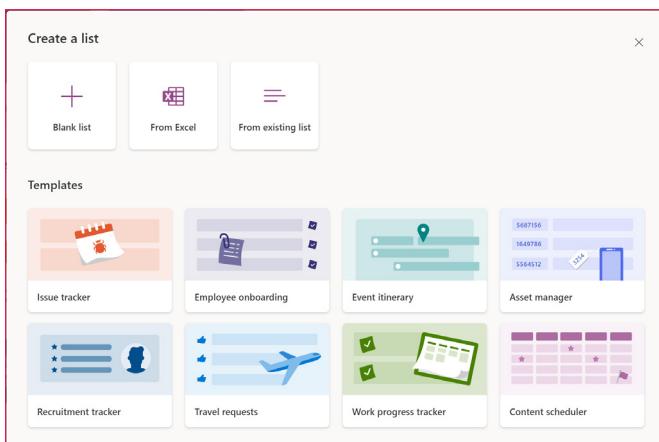


Imagen 5.- Posibilidades para crear una lista de Microsoft Lists.

- Seleccionamos por ejemplo la lista “Recruitment tracker” lo que nos permite tener una preview de la información que se puede registrar en dicha lista, así como las personalizaciones vía formato de columna que se incluyen de serie. Si la lista nos parece adecuada, para usarla simplemente haremos clic en “Use template”:

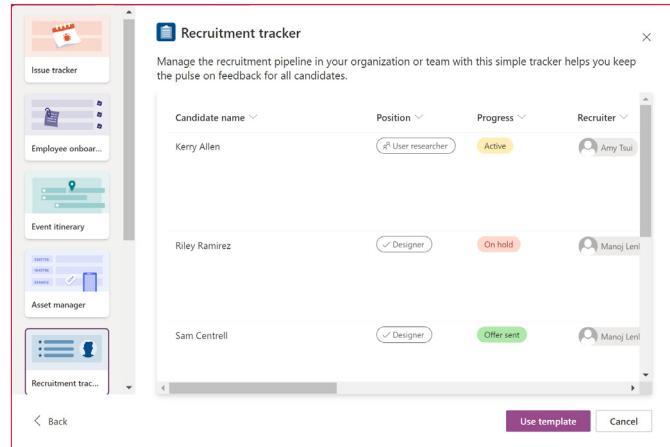


Imagen 6.- Preview de las personalizaciones y del propósito de la plantilla de lista seleccionada.

- A continuación, tendremos que indicar el nombre de la lista y opcionalmente una descripción. También podemos seleccionar el ícono (Nota: No es posible añadir nuestros propios iconos por el momento) y color identificativo de la lista. Finalmente, seleccionaremos la ubicación en la que queremos guardar la lista:
  - My lists (Mis listas) lo que implica que la lista se guardará en el ODFB del usuario.
  - Un Sitio de SPO concreto seleccionado a partir de los sitios recientes en los que ha trabajado el usuario. A la fecha de redacción de este artículo, no es posible indicar mediante una URL o búsqueda un sitio específico que no esté disponible en el listado de sitios recientes.

En mi caso voy a crear la lista como una lista personal. Para crear la lista, simplemente hacemos click en “Create”.

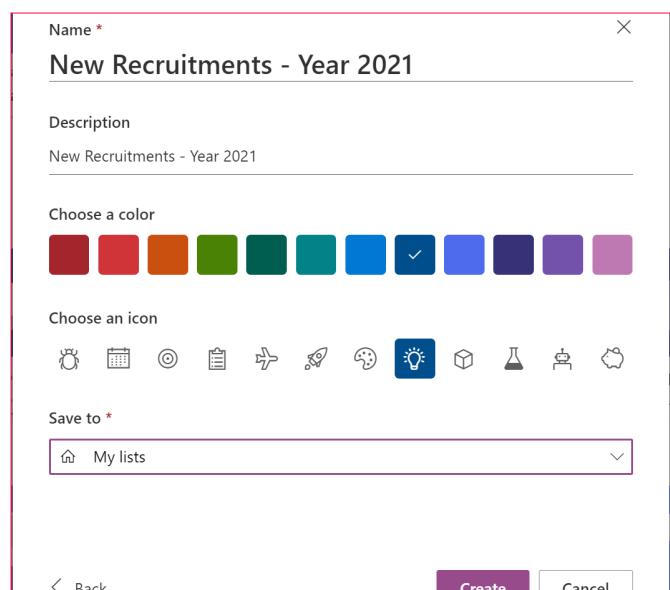


Imagen 7.- Personalizando la lista a crear.

- A continuación, se muestra la lista creada ocupando todo el ancho de página. En la nueva lista podremos comenzar a añadir registros bien a través del botón “New” o bien mediante la opción de “Edit in grid view”.

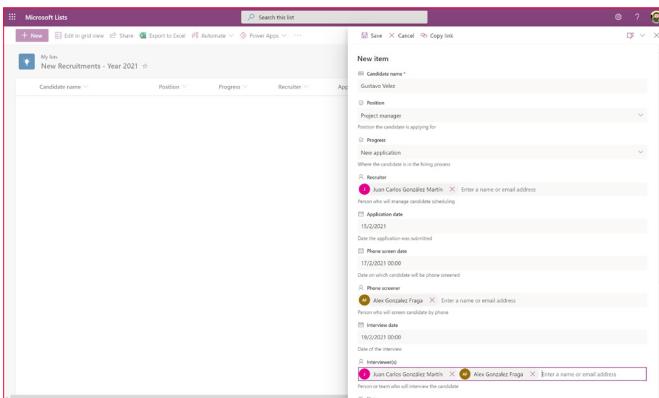


Imagen 8.- Añadiendo un registro a la lista creada.

- Cualquiera de las dos opciones nos llevará al siguiente resultado en el que veremos que se han aplicado distintas personalizaciones a nivel de columna de lista, aunque también se pueden aplicar personalizaciones a nivel vista o de formulario (Veremos estas personalizaciones en artículos posteriores).

**"Microsoft Lists no solo está presente en Microsoft 365 a través de la App de Microsoft Lists, sino también en Sitios de SPO y en Microsoft Teams"**

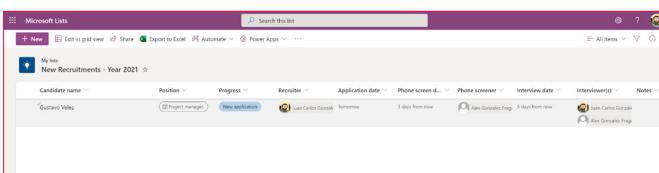


Imagen 9.- Resultado de añadir un registro en la lista creada.

## Integración de Microsoft Lists en Sitios de SPO y en Microsoft Teams

Microsoft Lists no solo está presente en Microsoft 365 a través de la App de Microsoft Lists, sino también en Sitios de SPO y en Microsoft Teams. En el caso de Sitios de SPO, la integración está disponible de serie de acuerdo con dos posibilidades:

- Botón “New” disponible en la página principal de un sitio.
- Botón “New” disponible en la página de contenidos de un sitio.

Cualquiera de las dos opciones muestra el diálogo de creación de listas que ya conocemos de la App de Microsoft Lists. La diferencia en este caso es que la lista se va a crear en el sitio actual.

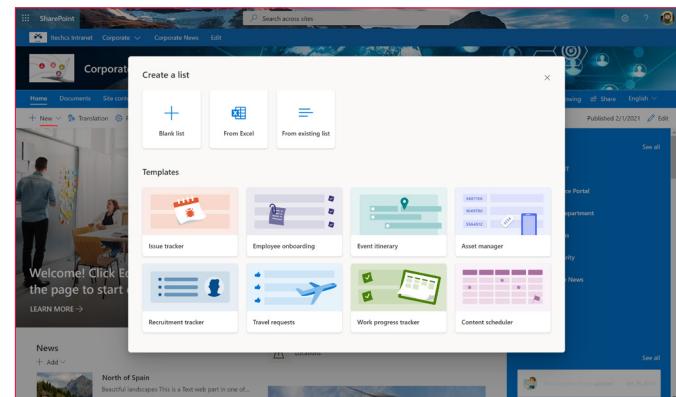


Imagen 10.- Creación de una lista desde un Sitio de SPO.

En el caso de Microsoft Teams, la integración con Microsoft Lists está disponible como una pestaña que nos permite agregar listas en canales de Teams:

- Dado un canal en Teams, hacemos clic en la opción de añadir una pestaña en un canal y a continuación seleccionamos Lists.

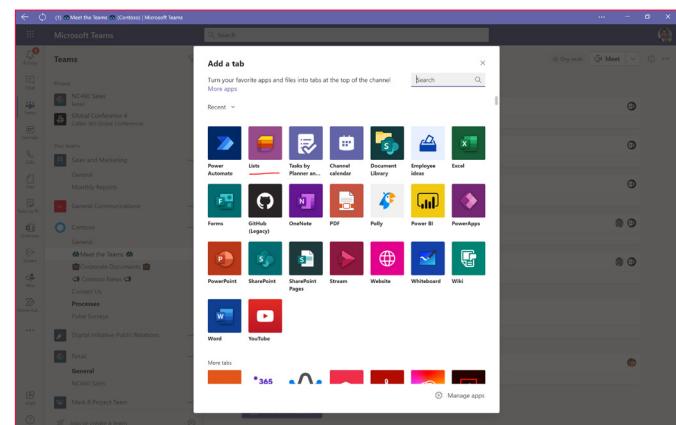


Imagen 11.- Añadiendo una pestaña de Microsoft Lists en un canal de Microsoft Teams.

- Una vez que hemos añadido la pestaña, podremos elegir entre crear una lista o añadir una lista existente. En mi caso, voy a hacer click en la primera de las opciones:

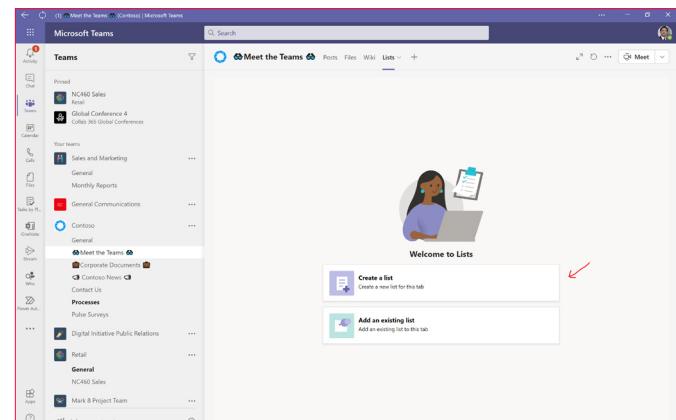


Imagen 12.- Opciones para crear o añadir una lista en un canal de Teams.

- A continuación, veremos que se muestra la experiencia familiar de creación de listas que hemos visto tanto en la App de Microsoft Lists como en un sitio de SPO. La diferencia sutil en este caso es que Microsoft proporciona 3 plantillas adicionales para listas en Microsoft Teams.

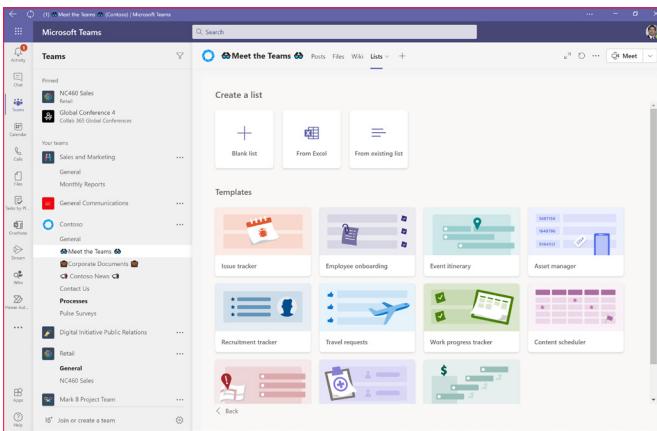


Imagen 13.- Experiencia de creación de una nueva lista en Teams.

- Si por ejemplo seleccionamos la plantilla Incidentes, a continuación, tendremos que seguir el mismo proceso ya visto en la App de Microsoft Lists con la diferencia de que la lista se va a crear en el sitio asociado al Team. En primer lugar, previsualizaremos la plantilla y confirmaremos que vamos a utilizarla.

**"Microsoft Lists representa la evolución de las listas tradicionales de SharePoint a un concepto de lista moderno"**

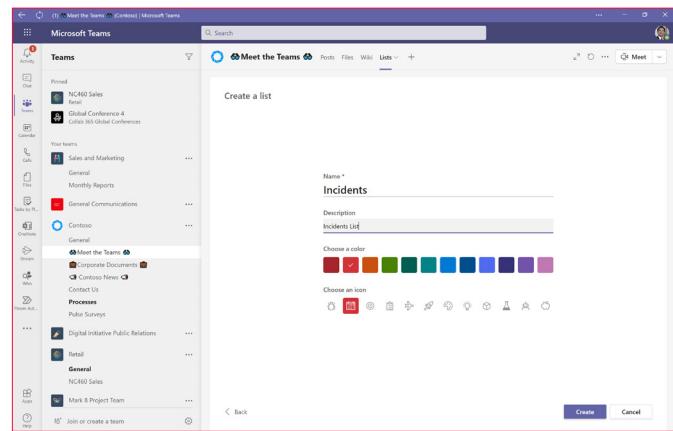


Imagen 15.- Configurando la lista a crear.

- Y a continuación tendremos la experiencia completa de trabajo con listas, pero en Microsoft Teams.

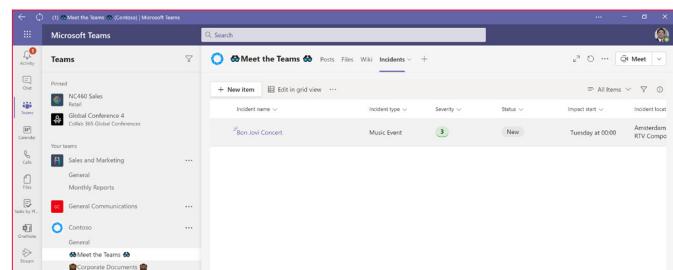


Imagen 16.- Lista creada e integrada en un canal de Teams como pestaña.

## Conclusiones

Microsoft Lists representa la evolución de las listas tradicionales de SharePoint a un concepto de lista moderno que facilita la creación rápida de aplicaciones sin código para los usuarios a nivel personal (en su ODFB) o bien de equipos de trabajo (en sitios de SPO). Microsoft proporciona una serie de plantillas de lista por defecto, aunque se pueden crear listas desde cero, a partir de hojas Excel o bien desde listas existentes. Finalmente, las listas de Microsoft Lists se integran tanto en Sitios de SPO como en Teams de Microsoft Teams.

---

### JUAN CARLOS GONZÁLEZ

Office Apps & Services MVP

Microsoft 365 SME & Delivery Manager en RICOH España  
@jcgm1978

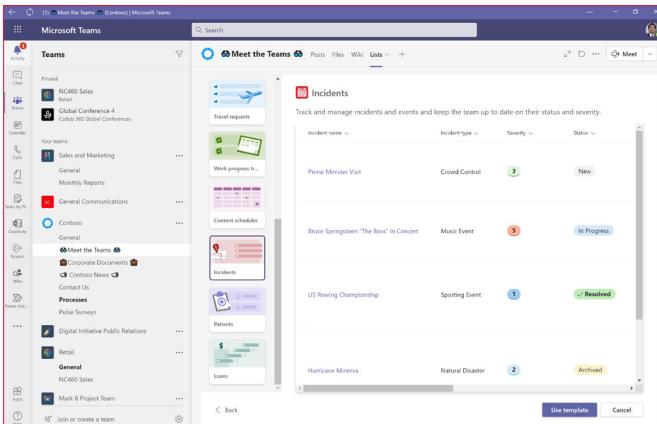


Imagen 14.- Previsualización de la plantilla de lista.

- Indicaremos el nombre, descripción, ícono identificativo y color para la lista. Procedemos a su creación con el botón "Create":

# Anatomía de una Adaptive Card

Desde hace unos cuantos meses, las Adaptive Cards se están convirtiendo en algo habitual en nuestras vidas y especialmente muy útil, en parte, por la potencia que ofrecen al poder ser integradas con múltiples servicios de Office 365 como puedan ser Outlook o Teams entre otros.

En este artículo vamos a explicar cómo es la anatomía de una Adaptive Card, cómo están formadas, cómo se pueden diseñar y posteriormente implementar en nuestras aplicaciones con algún caso de uso práctico.

## Background

Las Adaptive Cards realmente llevan con nosotros desde hace unos cuantos años (Microsoft Build de 2018), pero no ha sido hasta finales del año pasado/principios de este 2020 donde se han popularizado, especialmente por unas nuevas acciones de Power Automate como “Muestre su propia tarjeta adaptable como el bot Flow en un canal/a un usuario” o “Publique una tarjeta adaptable en un canal de Teams/para un usuario y espere respuesta”, ambas en preview y que más tarde detallaremos.

**"realmente llevan con nosotros desde hace unos cuantos años (Microsoft Build de 2018)"**

Previamente a esto, ya existían entre nosotros otros tipos de tarjetas como son las Message Cards, que nos permitían crear un tipo de tarjeta informativa que nos permitía tener una visualización más simple y que nos permitía integrar diferentes plataformas con nuestro Office 365.

## ¿Dónde podemos usar Adaptive Cards?

Microsoft da la posibilidad de utilizar las Adaptive Cards en varios sitios como pueda ser Outlook, Teams, un bot, integradas en Cortana o en el mismo Windows a través de sus tarjetas.

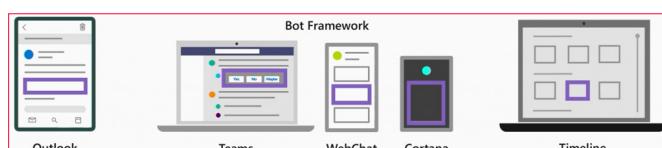


Imagen 1.- Posibilidades de uso de una Adaptive Card.

Un ejemplo de cómo se integran en los diferentes servicios es el siguiente:

### Outlook Actionable Message

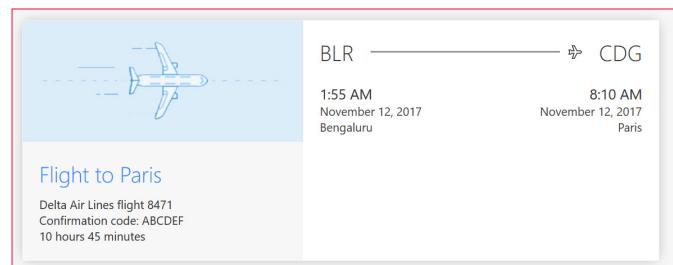


Imagen 2.- Mensaje accionable de Outlook.

### Microsoft Teams (Light Version)

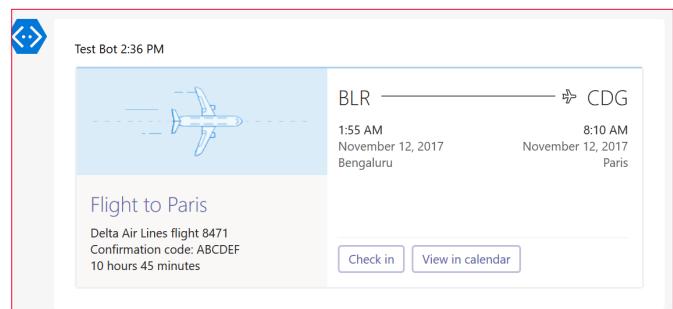


Imagen 3.- Ejemplo de Adaptive Card en Microsoft Teams.

### Bot Framework WebChat

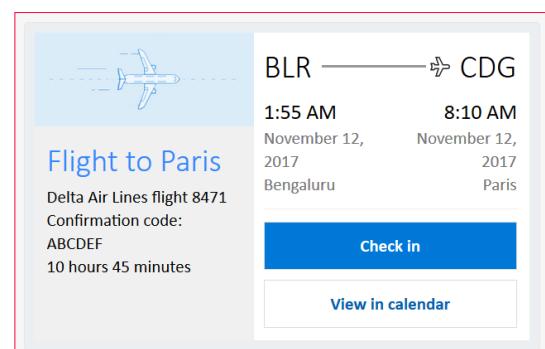


Imagen 4.- Ejemplo de Adaptive Card con Bot Framework.

**" Son portables: sirven para para cualquier aplicación y dispositivo"**

## Notificaciones de Windows

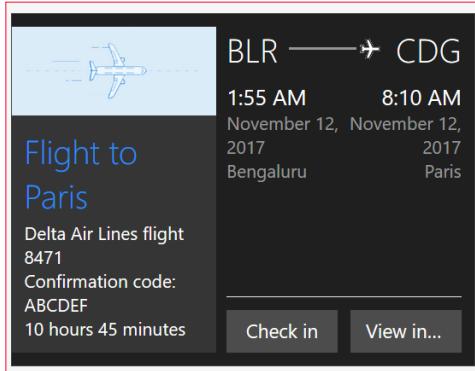


Imagen 5.- Ejemplo en notificaciones de Windows.

## ¿Cuáles son las ventajas que ofrecen las Adaptive Cards?

- Son portables: sirven para cualquier aplicación y dispositivo.
- Abiertas y con lenguaje declarativo: por estar hechas con un esquema JSON y no llevar código asociado.
- De bajo coste y con estilos automáticos: por coger el UX de la aplicación que lo acoge.
- Esquema único: minimizando el coste de crear un tipo de tarjeta por aplicación ni aprender nuevos esquemas
- Experiencia de uso consistente.
- Rendimiento nativo ya que coge los elementos nativos del framework que lo ejecuta.
- Fácil de implementar.
- Altamente documentado.

## ¿Cómo se estructuran las Adaptive Cards?

Las Adaptive Cards son unos elementos puramente declarativos, esto es, que no requieren crearse con código para poder trabajar con ellas al estar hechas en formato JSON. Un ejemplo de una Adaptive Card es el siguiente...

```
{
  "type": "AdaptiveCard",
  "body": [
    {
      "type": "TextBlock",
      "size": "Medium",
      "weight": "Bolder",
      "text": "${title}"
    },
    {
      "type": "ColumnSet",
      "columns": [
        {
          "type": "Column",
          "items": [
            {
              "type": "Image",
              "style": "Person",
              "url": "${creator.profileImage}",
              "size": "Small"
            }
          ],
          "width": "auto"
        },
        {
          "type": "Column",
          "items": [
            ...
          ]
        }
      ]
    }
  ]
}
```

```
{
  "type": "TextBlock",
  "weight": "Bolder",
  "text": "${creator.name}",
  "wrap": true
},
{
  "type": "TextBlock",
  "spacing": "None",
  "text": "Created ${DATE(${createdUtc},SHORT)}",
  "isSubtle": true,
  "wrap": true
},
],
"width": "stretch"
},
{
  "type": "TextBlock",
  "text": "${description}",
  "wrap": true
}
],
"actions": [
{
  "type": "Action.ShowCard",
  "title": "Set due date",
  "card": {
    "type": "AdaptiveCard",
    "body": [
      {
        "type": "Input.Date",
        "id": "dueDate"
      },
      {
        "type": "Input.Text",
        "id": "comment",
        "placeholder": "Add a comment",
        "isMultiline": true
      }
    ],
    "actions": [
      {
        "type": "Action.Submit",
        "title": "OK"
      }
    ],
    "$schema": "http://adaptivecards.io/schemas/adaptive-card.json"
  }
},
{
  "type": "Action.OpenUrl",
  "title": "View",
  "url": "${viewUrl}"
}
],
"$schema": "http://adaptivecards.io/schemas/adaptive-card.json",
"version": "1.2"
}
```

...que una vez lo integremos con el servicio que vaya a hospedarlo, nos ofrecerá su resultado. El código anterior, en Teams, se vería así:



Imagen 6.- Visualización del código JSON en Teams.

## Diseñadores

Ok ok. Os he enseñado las tripas de una Adaptive Card

antes de decir cómo podemos trabajar con ellas, pero en este aspecto, hay muchas alternativas. Vamos a explorar los diferentes diseñadores que podemos usar para ello:

## adaptivecards.io

Existe una página que hace de playground con un editor WYSIWYG (what you see is what you get) para ayudarnos a construir nuestra tarjeta adaptable y posteriormente llevárnosla a nuestras aplicaciones con el JSON que se nos genera automáticamente. Es <https://adaptivecards.io/designer>

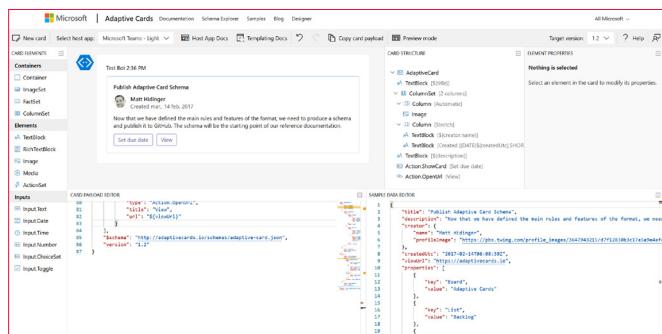


Imagen 7.- Página adaptivecards.io.

En esta página podremos ver los diferentes apartados:

## MENÚ SUPERIOR



Imagen 8.- Menú superior.

En el menú superior vamos a poder generar una tarjeta nueva, seleccionar el contenedor que va a hospedar nuestra Adaptive Card para ver los estilos (Bot Framework, Outlook, Teams, Cortana, Windows, etc...), ver cierta documentación y un botón para copiar el JSON resultante de nuestro trabajo. Y, por último, y no menos importante, el Target version: si queremos trabajar con Teams debemos seleccionar la versión 1.2, ya que la versión 1.3 del schema todavía no lo soporta la plataforma.

## BARRA LATERAL IZQUIERDA

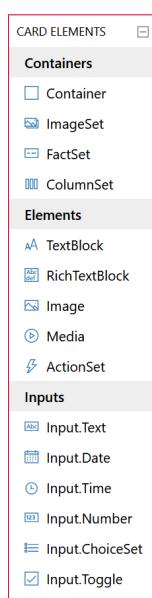


Imagen 9.- Barra lateral izquierda.

En la barra izquierda podemos elegir qué elementos queremos insertar en nuestra tarjeta: imágenes, textos, botones, etc...

## LIENZO CENTRAL: PARTE SUPERIOR

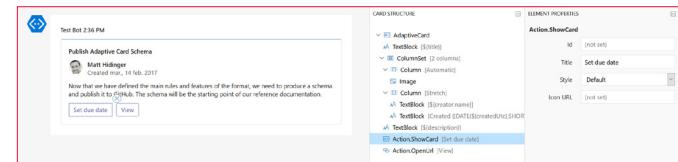


Imagen 10.- Lienzo central: parte superior.

En la parte central superior vamos a ver que existen tres columnas diferencias: a la izquierda, nuestra card en formato WYSIWYG, para que veamos el resultado y podamos ir seleccionando cada elemento de la estructura de la tarjeta; esa estructura está también indicada en la columna central, para que veamos en formato árbol y sepamos cómo está organizada y a la derecha las propiedades que se pueden asignar a cada elemento posible de las Adaptive Cards: Id, título, estilo, ícono, efectos para el layout... y en definitiva múltiples opciones para hacer nuestra tarjeta lo más configurable posible.

## LIENZO CENTRAL: PARTE INFERIOR

Por último, en la parte inferior izquierda tendremos el payload con el JSON resultante de nuestra tarjeta que será el que tengamos que utilizar en nuestras diferentes aplicaciones y a la derecha un template de ejemplo para que le demos información a nuestra Adaptive Card con texto e imágenes de prueba.

## App Studio en Microsoft Teams

A través del App Studio de Teams vamos a poder trabajar con ellas directamente desde la plataforma:

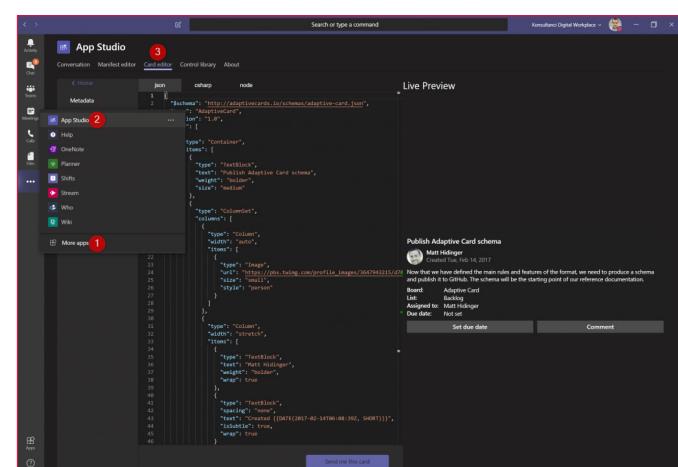


Imagen 11.- App Studio.

## Add-Ins en Visual Studio Code

En cambio, si nos gusta más trabajar con Visual Studio Code, tenemos suerte porque hay un par de extensiones para este IDE donde nos permitirá previsualizar en tiempo real nuestra Adaptive Card dentro de él a partir del JSON que estemos trabajando. Son “AdaptiveCards Studio” y “Adaptive Card Viewer”:

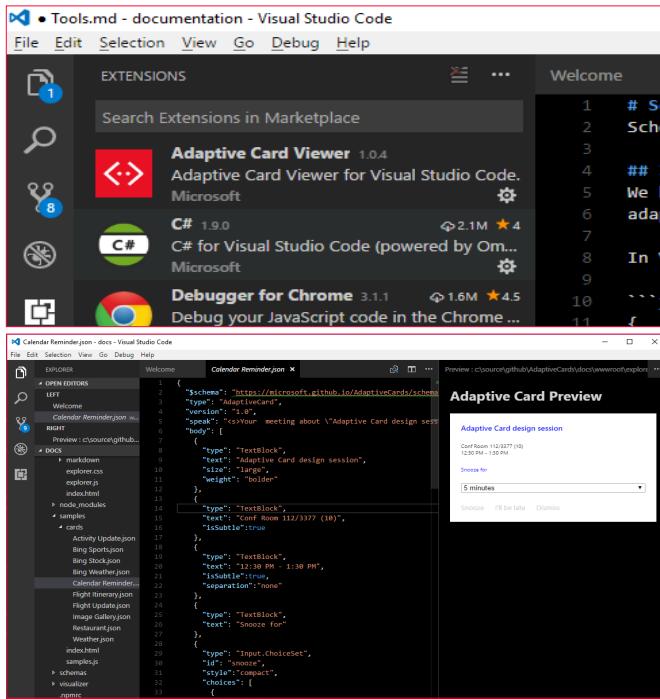


Imagen 12.- Add-ins en Visual Studio Code.

## Ejemplos

Y como siempre el movimiento se demuestra andando, lo mejor es que podamos ver cómo funcionan estas Adaptive Cards en nuestros desarrollos. En este caso, y por simplicidad, vamos a utilizarlos dentro de Teams gracias a la acción “Muestre su propia tarjeta adaptable como el bot Flow en un canal” que hemos comentado al comienzo del artículo.

**“Otro tipo de tarjeta que han introducido hace unos cuantos meses que nos permite enviar una Adaptive Card a un canal o usuario y esperar su respuesta”**

### Adaptive Card en un canal de Teams

Para poder mandar una Adaptive Card a un canal de Teams sólo tenemos que elegir la acción correspondiente en Power Automate:

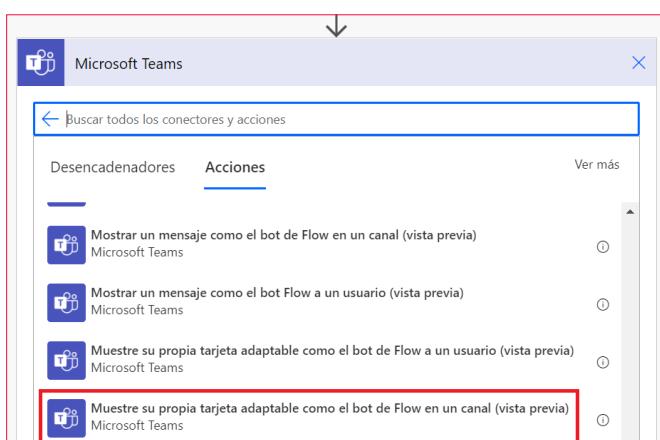


Imagen 13.- Adaptive card en un canal de Teams.

Y llenar los campos que se solicitan:

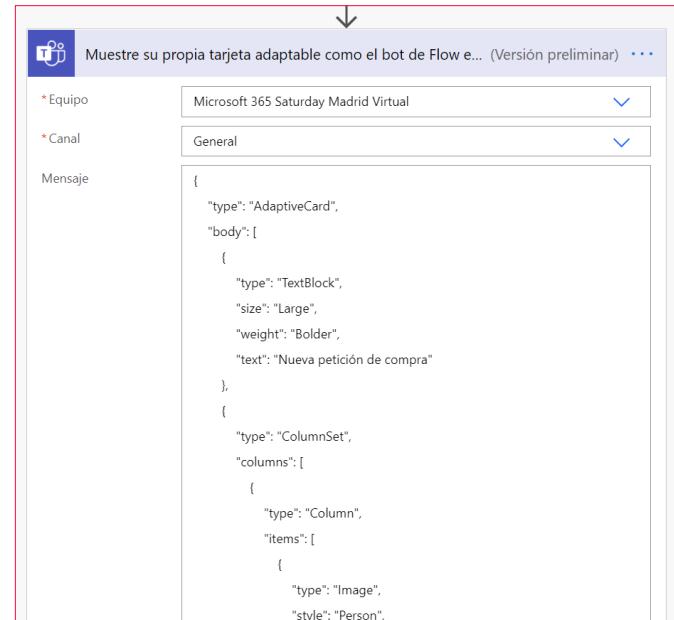


Imagen 14.- Datos a completar para la Adaptive Card en Teams.

Este resultado una vez enviado sería el siguiente:

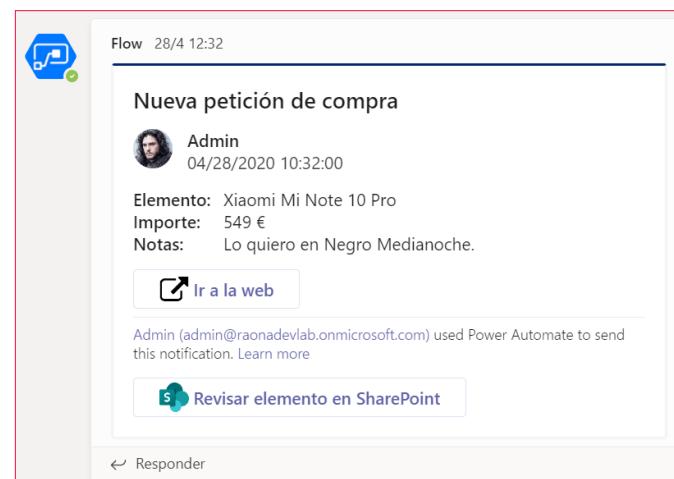


Imagen 15.- Adaptive Card resultante.

### Adaptive Card en un chat de Teams

Para el caso de enviar la información a un usuario, la forma de hacerlo es la misma, pero eligiendo la opción de un usuario en vez de un canal:

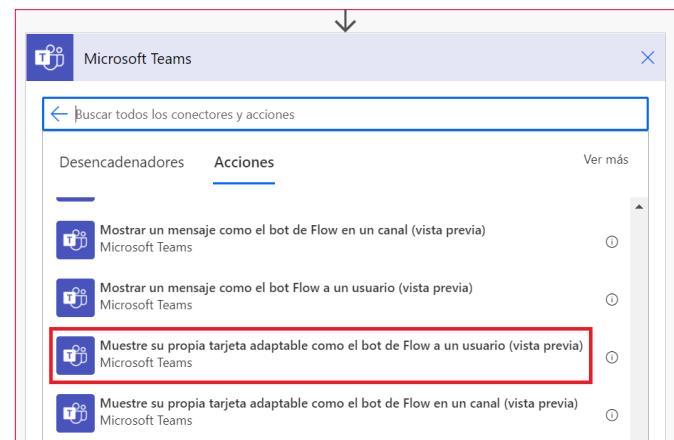


Imagen 16.- Adaptive card en un chat de Teams.

Y su resultado:

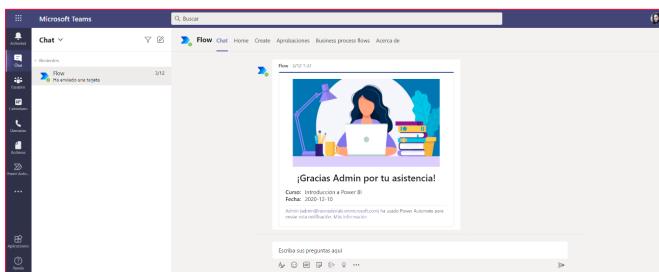


Imagen 17.- Visualización de la Adaptive Card en un chat de Teams.

## Esperando respuesta de nuestra Adaptive Card

Por último, existe otro tipo de tarjeta que han introducido hace unos cuantos meses que nos permite enviar una Adaptive Card a un canal o usuario y esperar su respuesta. El formato es exactamente el mismo que una Adaptive Card normal, pero permite recoger y registrar las respuestas posteriormente en nuestro flujo.

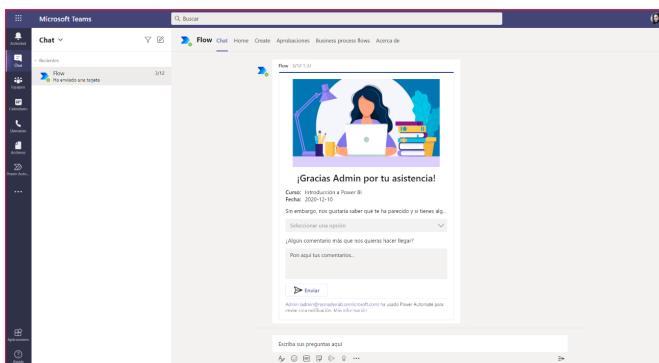


Imagen 18.- Esperando respuesta en una Adaptive Card.

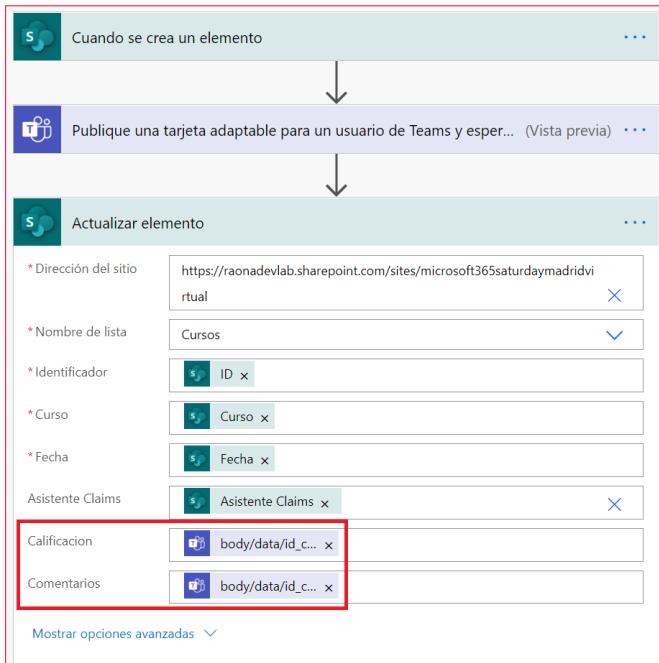


Imagen 19.- Registro de las respuestas en el Flow de Power Autoamte.

Para ello tendremos que especificar los valores de los IDs que hayamos indicado en nuestra Adaptive Card, en este caso, por ejemplo, referenciando a:

```
"Calificación": "@outputs('Publique_una_tarjeta_adaptable_para_un_usuario_de_Teams_y_espere_respuesta')?['body/data/id_calificacion']",
"Comentarios": "@outputs('Publique_una_tarjeta_adaptable_para_un_usuario_de_Teams_y_espere_respuesta')?['body/data/id_comentarios']"
```

Adicionalmente, aparecerán nuevas opciones en la propia Adaptive Card para mostrar un mensaje tras haber interactuado con ella:

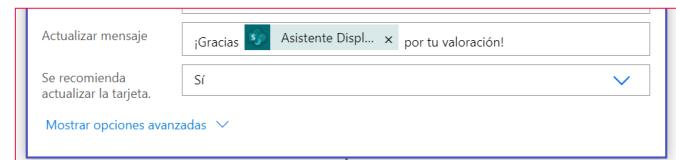


Imagen 20.- Opciones adicionales para mostrar un mensaje.

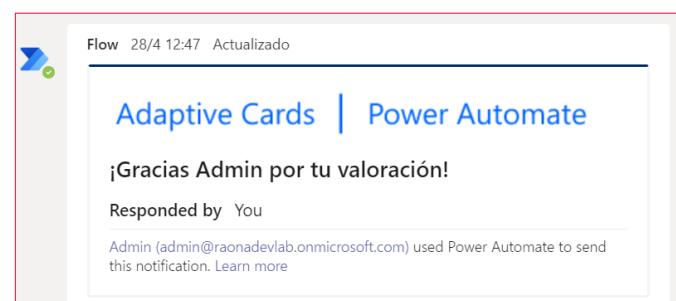


Imagen 21.- Adaptive Card en Flow.

*NOTA: Cabe indicar que las respuestas a las Adaptive Cards, como cualquier flujo de Power Automate, tienen una vida de 30 días. Pasado este tiempo el flujo fallará y no se registrarán acciones posteriores.*

## Para finalizar...

Como habéis podido ver a lo largo del artículo, las Adaptive Cards nos muestran grandes capacidades para ampliar nuestros procesos de negocio dentro de la propia plataforma en la que nos encontramos, siendo de especial utilidad dentro de Microsoft Teams gracias a la potencia de Power Automate.

Si queréis conocer las últimas novedades podéis añadir las Community Calls en vuestro calendario (<https://aka.ms/adaptivecardscommunitycall>), revisar el Portal de features del producto (<https://aka.ms/ACFeatures>) para votar por aquellas que queráis o proponer algunas o simplemente seguir el hashtag #AdaptiveCards en Twitter o el blog oficial <https://adaptivecards.io/blog>.

**ÁNGEL CARRILLO**  
 @angeldav

# 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>



## Entrevista Krasis

Krasis es una empresa tecnológica especializada en e-Learning desde el año 2000, en los albores de esta disciplina en España. Comercializan SELF, una plataforma de e-Learning de desarrollo 100% propio para empresas de formación de cierto volumen, así como corporaciones. También están detrás del proyecto campusMVP.es, muy conocido entre la comunidad técnica, que es la referencia en formación online para programadores en español. En campusMVP.es utilizan su plataforma, claro.

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

Krasis nació de la mano de José Manuel Alarcón, una de las personas de España que más veces ha sido reconocido como MVP por Microsoft, ya que lleva siendo premiado ininterrumpidamente todos los años desde 2004. Dados estos orígenes, la empresa ya nació respirando tecnología y programación, y siempre tuvo una orientación tecnológica, con un claro sesgo hacia tecnologías Microsoft, si bien tenemos excelentes tutores especializados en otras tecnologías.

Lo de la formación online es una vocación que nace de su fundador también, que siempre ha estado de un modo u otro, incluso por tradición familiar, muy orientado a la formación. En el año 2000 las pocas plataformas de e-Learning existentes estaban todavía en su infancia y quedaba hueco en el mercado para productos innovadores. Por eso nos lanzamos a crear nuestra propia plataforma.

Nunca nos hemos preocupado por hacer benchmark competitivo con otras plataformas, sino que nos hemos obsesionado con las necesidades de los clientes, y con hacer dog feeding, esto es, que nosotros mismos seamos los principales usuarios de nuestra tecnología, de modo que vivamos lo mismo que viven los clientes, algo muy poco común en vendedores de software especializado. A través de campusMVP.es formamos a miles de alumnos y podes-



Trabajan principalmente con herramientas de Microsoft, y cuentan con muchos MVPs entre sus tutores.

mos conocer de primera mano las verdaderas necesidades que surgen a la hora de hacer formación online.

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

Nuestras actividades tecnológicas están orientadas a desarrollar nuestra plataforma de eLearning, para lo cual tenemos que tocar muchas tecnologías y sistemas diferentes, además de administrar servidores, servicios en la nube, etc... por lo que hay bastante de TI también.

Al mismo tiempo, a través de campusMVP.es tenemos que estar siempre muy al cabo de la calle en lo que respecta a tendencias tecnológicas, y desarrollamos contenidos técnicos de muy alta calidad sobre todo tipo de tecnologías de desarrollo. Un curso típico son muchos meses de trabajo, para asegurar unos estándares que son los que nos han hecho obtener el reconocimiento que tenemos actualmente entre desarrolladores y empresas. Es un trabajo también bastante técnico.

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

Nuestras principales actividades no tecnológicas, en el día a día, conllevan el soporte técnico a clientes y la atención a los alumnos.

Dentro de lo primero incluimos una gran componente de consultoría para los clientes de la plataforma, ayudándoles a llevar a cabo sus proyectos de e-Learning con éxito. No es un servicio que prestemos explícitamente, sino que forma parte integral del servicio de software que vendemos: van unidos.

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

Tenemos un blog con cientos de artículos técnicos, accesibles de manera gratuita en campusMVP.es. cada mes cientos de miles de desarrolladores de todo el mundo de habla hispana nos visitan y aprenden cosas con nosotros sin necesidad de comprar un curso, aunque muchos lo hacen al comprobar la calidad.

Gestionamos un interesante canal de YouTube en el que publicamos vídeos técnicos, trucos, entrevistas para hablar de tecnología, un noticario semanal con toda la información relevante de lo que pasa cada semana en el mundillo.

Además, tenemos un boletín mensual, que se envía a mediados de mes, en el que incluimos artículos, enlaces, noticias... y un par de secciones que llevamos haciendo muchos años y que tienen mucho éxito, sobre meteduras de pata informáticas, y sobre "frikadas" que hace la gente por ahí y que nos llaman la atención.

Todo esto es un trabajo ingente que, si bien puede ser bastante técnico, no forma parte de lo que consideramos "actividades tecnológicas" propias de la empresa.

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

Una de las tendencias que más interesante nos parece de cara al futuro a medio plazo, es la proliferación de sistemas basados en Inteligencia Artificial y Deep Learning, en todos los aspectos de las empresas y de la sociedad. Todavía está en sus albores, a pesar de que lo vemos por todas partes,

y en los próximos años va a explotar y habrá una gran demanda, mayor de la que ya hay, de técnicos especializados que dominen bien los conceptos subyacentes. Es por eso que, tras dos años de intenso trabajo, acabamos de lanzar un máster para especializarse en estas disciplinas.

Y luego, en el mundo del desarrollo de software existen dos tendencias opuestas, pero complementarias, que son: la mayor complejidad y la simplificación a la hora de desarrollar. Existe una mayor complejidad, porque cada vez es más difícil, no ya profesionalizarse como desarrollador, sino incluso seguir el ritmo a las novedades y a las tecnologías que van apareciendo. Pero al mismo tiempo existe una sencillez cada vez mayor para crear software, gracias a varias tendencias que han explotado en los últimos años como la disponibilidad brutal que hay ahora de Open Source, las herramientas Low-Code o No-Code, herramientas online como Zapier o Airtable (que no tienen que ver funcionalmente, pero a efectos de lo que hablo, están relacionadas), y las plataformas Serverless como Azure Functions o CloudFlare Workers. Todas ellas conducen a que gente con conocimientos básicos técnicos o de programación pueden crear en poco tiempo aplicaciones de calidad y escalables.

Creemos que estas dos tendencias antagónicas van a dar lugar en la próxima década a dos modos muy diferentes de hacer programación, creando dos categorías muy diferentes de desarrolladores. Por un lado, tendremos gente muy especializada que dominará una determinada tecnología o nicho (ya que dominar varios es muy complicado) y que tendrán salarios altos y poder en las empresas, como pasa ahora con algunos desarrolladores, pero con más barreras de entrada para formarse por la dificultad y la entrega necesarias. Pero por otro veremos que mucha gente no técnica podrá crear sus propias soluciones a medida y que habrá programadores con bajo nivel de preparación haciendo muchos pequeños desarrollos en las empresas basados en "juntar piezas", integraciones o crear pequeños programas, que serán más que suficientes. Estos perfiles no estarán tan bien pagados y tendrán menos necesidad de formación y menos estabilidad laboral, pero darán trabajo a mucha gente.

# Microsoft Graph .Net SDK al descubierto

En este artículo vamos a hablar de la última versión del SDK de MS Graph para .NET, viendo las opciones que tenemos para personalizarlo a más bajo nivel, algo que nos será muy útil en determinados escenarios.

## Introducción a Graph SDK para .NET

A estas alturas, ya es bien conocido que, si trabajamos con Graph API desde .net, tenemos un SDK desarrollado por MS, que está muy maduro, y que desde luego recomiendo utilizar, en lugar de lanzar peticiones directamente a la API.

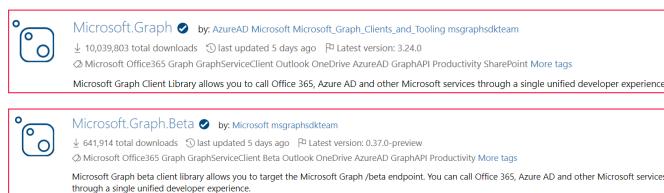


Imagen 1- Microsoft Graph .NET SDKs.

Tenemos dos paquetes nuget a elegir, uno trabaja con la versión v1.0 de la API, mientras que el otro, lo hace con la versión Beta. Esto no solo influye en la URL de Graph a invocar por el SDK (`graph.microsoft.com/v1/...` vs `graph.microsoft.com/beta/...`), si no que el modelo de objetos en general, es diferente, y para el Beta, tendremos más propiedades en clases como User, o todas las clases que dan soporte a la nueva Taxonomía de SharePoint, y que ya podemos usar desde Graph, incluidas en el namespace `Microsoft.Graph.TermStore`.

Y es aquí donde viene la primera decisión: ¿qué paquete debo utilizar? Bueno, la recomendación es que, si no necesitas nada de la versión Beta, utilices la V1, sino, la beta. Evidente, ¿verdad?... pero, y si necesito parte de la Beta, me refiero a funcionalidad que solo existe en el /beta endpoint, y parte que no require del /beta endpoint? En este caso, mi consejo es usar el paquete Beta, y para aquellas operaciones que te sea suficiente con el endpoint v1.0 (y también te recomiendo que siempre que te sea suficiente, tires del v1.0, ya que es el estable), cambies el `GraphServiceClient` para que la base URL apunte al v1.0. Luego veremos cómo hacer eso. Pero... y si invoco a un recurso de v1.0 que tiene menos propiedades/campos, que en el endpoint Beta, ¿qué pasará con el objeto tipado del SDK Beta que espera esos campos?... pues en teoría quedarán

vacíos (null, default), ya que aplica lo mismo que cuando haces una request, y aplicas un select para traer sólo los campos que te hacen falta.

Existe también una tercera opción en la que convivirán ambos paquetes en el mismo proyecto. A mí me parece menos interesante que la opción anterior, pero si queréis tirar por este camino, está bien documentado el proceso en la página principal del repositorio en GitHub del Beta SDK: <https://github.com/microsoftgraph/msgraph-beta-sdk-dotnet#using-the-beta-client-along-with-the-v10-library>

## Authorization en Graph SDK

Como siempre, la parte de Authorization es de las más complejas. La MS Graph API, espera un Bearer token válido en la cabecera `Authorization` de la request, y la forma de obtener dicho Token, dependerá de nuestro escenario y del Flow elegido para negociar dicho token con Azure Active Directory.

Por suerte, tenemos un par de recursos que nos van a facilitar mucho la vida:

**1.-** Si estás desarrollando una aplicación Web o API, te recomiendo que utilices una nueva librería que ha desarrollado la gente de Azure AD: `Microsoft.Identity.Web` (<https://github.com/AzureAD/microsoft-identity-web>).

**2.-** Si estás desarrollando otro tipo de aplicación: Daemons, Consola, Móvil... entonces te recomiendo que eches un vistazo a otra librería de la gente de Graph: `Microsoft.Graph.Auth` (<https://github.com/microsoftgraph/msgraph-sdk-dotnet-auth>).

*Nota: Dicho paquete está en preview, y según la propia gente de Graph, es difícil que alcance GA, ya que el equipo de Graph está trabajando con el de Azure AD para hacer algo mejor (al estilo de Microsoft.Identity.Web), pero de momento, si no quieres lidiar a más bajo nivel con la librería MSAL, este paquete te ayudará bastante. Yo de momento no he tenido problemas.*

## Auth en API que consume Graph API

En el startup, usamos varias extensiones de la librería MS Identity Web para registrar un `GraphServiceClient` en el contenedor de servicios. Además, estamos configurando la

caché distribuída de Tokens “AddDistributedTokenCaches” usando SQL Server “AddDistributedSqlServerCache”

```
services.AddDistributedSqlServerCache(options =>
{
    options.ConnectionString = Configuration.GetConnectionString("WestworldDbContext");
    options.SchemaName = "dbo";
    options.TableName = "SqlCache";
});

services.AddMicrosoftIdentityWebApiAuthentication(Configuration)
    .EnableTokenAcquisitionToCallDownstreamApi()
    .AddMicrosoftGraph(options => { options.Scopes = "User.Read.All Mail.Read"; })
    .AddDistributedTokenCaches();
```

*"si trabajamos con Graph API desde .net, tenemos un SDK desarrollado por MS, que está muy maduro"*

Una vez registrados los servicios, podemos injectar en el constructor de cualquier clase, un objeto GraphServiceClient, que ya estará configurado adecuadamente para incluir un token válido a la petición.

```
[Route(template: "api/[controller]")]
[ApiController]
1 reference
public class GraphController : ControllerBase
{
    private readonly GraphServiceClient _graphServiceClient;

    0 references
    public GraphController(GraphServiceClient graphServiceClient)
    {
        _graphServiceClient = graphServiceClient;
    }

    [HttpGet(template: "me")]
    0 references
    public async Task<IActionResult> Me()
    {
        var me = await _graphServiceClient.Me.Request().GetAsync();

        return Ok(new
        {
            me.DisplayName,
            me.Mail
        });
    }
}
```

## Auth en una Azure Function Timer Trigger

En el startup de la Function, registramos el GraphServiceClient, usando un Authentication provider de la librería Microsoft.Graph.Auth. En este caso, necesitamos también usar MSAL, para construir un objeto ConfidentialClient.

```
// Singleton is recommended for connections and clients
// (https://docs.microsoft.com/en-us/azure/azure-functions/functions-dotnet-dependency-injection#service-lifetimes)
// besides, if registered as Singleton the MSAL in memory token cache is gonna reduce request to Azure AD Token endpoint.
services.AddSingleton<GraphServiceClient, GraphServiceClient>(serviceProvider =>
{
    var options = serviceProvider.GetService<IOptions<AzureAdOptions>>();
    var azureAd = options.Value;

    var confidentialClientApplication = ConfidentialClientApplicationBuilder
        .Create(azureAd.ClientId)
        .WithTenantId(azureAd.TenantId)
        .WithClientSecret(azureAd.ClientSecret)
        .Build();

    var authenticationProvider = new ClientCredentialProvider(confidentialClientApplication);
    var graphServiceClient = new GraphServiceClient(authenticationProvider);
    return graphServiceClient;
});
```

Una vez registrado, podemos hacer uso del GraphServiceClient inyectándolo al constructor de nuestra Azure Function (igual que hemos hecho en el punto anterior con el API Controller).

## Arquitectura del SDK

La siguiente imagen describe la arquitectura del SDK y los diferentes componentes principales:

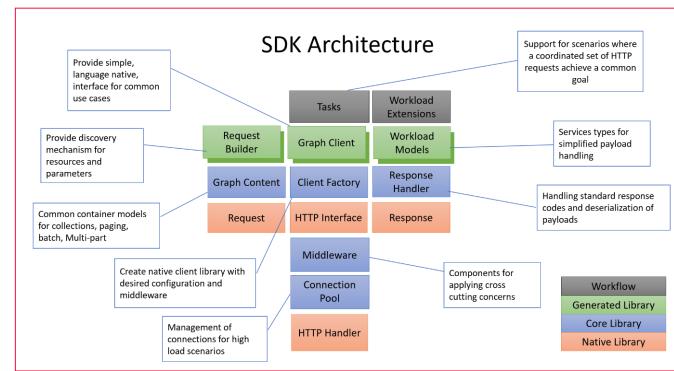


Imagen 2 - Arquitectura del SDK.

A continuación, vamos a ver algunos ejemplos de cómo podemos personalizar ciertos aspectos del SDK:

## Iterando colecciones paginadas

En ocasiones, los datos devueltos por Graph, vienen limitados a un número máximo de elementos, incluso si especificas un número en el parámetro \$Top. Por ejemplo, si quieras obtener los Users del directorio, como máximo recibirás 999 usuarios en una única llamada. Lo que hace Graph (como muchas otras APIs), es devolverte un NextLink, para que puedas hacer otra petición a la siguiente página. Es muy normal que quieras obtener todos los usuarios del directorio, así que tendrías que mirar ese NextLink, e ir haciendo más peticiones usando ese Link, hasta que no haya más páginas.

Por suerte, el SDK, en el módulo de Tasks, ofrece clases para poder iterar una colección página de principio a fin. El siguiente snippet os muestro como hacerlo:

```
var groups = await graphServiceClient
    .Groups
    .Request()
    .Top(5)
    .GetAsync();

var pageIterator = PageIterator<Group>.CreatePageIterator(graphServiceClient, groups,
    (group) => {
        Console.WriteLine(group.DisplayName);
        return true;
    }
);

await pageIterator.IterateAsync();
```

*Nota: La clase Pageliterator, sólo existe en el SDK v1. Existe un issue abierto en GitHub para llevarla también al Beta (<https://github.com/microsoftgraph/msgraph-beta-sdk-dotnet/issues/187>), pero de momento no está. También podéis copiar esa clase y añadirla a vuestro código, que, por mis pruebas, ha funcionado perfectamente.*

## Manejando errores

Cada vez que se produzca un error en la llamada a Graph, el SDK va a lanzar una excepción del tipo ServiceException. Hay muchas posibles causas del error, y el SDK se ha encargado de definir los posibles errores, además de proveernos de una extensión en la clase ServiceException para hacer el “match” con el posible código de error. Vemos un ejemplo en el siguiente snippet:

```

try
{
    var groups = await graphServiceClient
        .Groups
        .Request()
        .Top(5)
        .GetAsync();
}

catch (ServiceException ex)
{
    if (ex.IsMatch(GraphErrorCode.InvalidQueryOption.ToString()))
    {
    }
}

```

Podéis ver todos los posibles ErrorCode's definidos en el siguiente enum: <https://github.com/microsoftgraph/ms-graph-sdk-dotnet/blob/dev/src/Microsoft.Graph/Enums/GraphErrorCode.cs>

## Cambiando la URL base de Graph API para una petición concreta

Como comentaba anteriormente, que tengamos el SDK Beta, no significa que no podamos hacer peticiones al endpoint v1.0. Para ello, es tan sencillo como:

```

var test = graphClient.BaseUrl = "https://graph.microsoft.com/v1.0/";
User profile = await graphClient.Me.Request().GetAsync();
Console.WriteLine(profile.DisplayName);

```

## Añadiendo Custom Headers a la petición

En ocasiones, puede que necesites añadir alguna cabecera a la petición, así que conviene saber cómo hacerlo. Un caso bastante común es que algunas queries a Graph, requieren una cabecera concreta para que la query funcione (ejm: ConsistencyLevel = Eventual), o añadir alguna QueryString que no permite el SDK (Ejm: \$Search). Para esos casos, podemos hacer:

```

List<Option> options = new List<Option>();
options.Add(new HeaderOption("ConsistencyLevel", "Eventual"));
options.Add(new QueryOption("$search", "displayName:tier"));
var myTransversalSecurityGroups = graphClient.Me.GetMemberGroups(true).Request(options).PostAsync();

```

## Añadiendo Custom Handler a la Pipeline

El SDK define un pipeline que podemos extender a la hora de enviar la petición y recibir la respuesta, lo que nos puede venir muy bien en algunos escenarios, para temas de "cross-cutting concerns", como pueden ser Logging, Retries, etc. En principio, el propio SDK ya viene por defecto con 4 Handlers para Authentication, Compression, Retry y Redirect:

```

// <summary>
// Create a default set of middleware for calling Microsoft Graph
// </summary>
// <param name="authenticationProvider">The <see cref="IAuthenticationProvider"/> to authenticate requests.</param>
// <returns></returns>
public static IList<DelegatingHandler> CreateDefaultHandlers(IAuthenticationProvider authenticationProvider)
{
    return new List<DelegatingHandler> {
        new AuthenticationHandler(authenticationProvider),
        new CompressionHandler(),
        new RetryHandler(),
        new RedirectHandler()
    };
}

```

Para ver la potencia de esto, vamos a hacer un Handler que va a devolver un usuario ficticio cuando se llame al Me en-

dpoint del GraphServiceClient, algo que puede sernos muy útil para testing.

Primero creamos nuestro DelegatingHandler, de tal manera que, si la Request contiene el endpoint de /me, en lugar de enviar la request a Graph, creamos un objeto User con datos ficticios, lo serializamos, y lo devolvemos en la respuesta. Aquí tenéis el código:

```

public class GraphMeSimulator : DelegatingHandler
{
    1 reference
    protected override Task<HttpResponseMessage> SendAsync(
        HttpRequestMessage request,
        System.Threading.CancellationToken cancellationToken)
    {
        Console.WriteLine(request.RequestUri.AbsolutePath);

        if (request.RequestUri.AbsolutePath.ToLower().Contains("me"))
        {
            var fakeUser = new User();
            fakeUser.DisplayName = "Mr Fake User";
            fakeUser.Id = "ff937144-a7f0-4665-8282-52350ea8f802";
            fakeUser.JobTitle = "No Job";
            fakeUser.UserPrincipalName = "fakeuser@tenant.onmicrosoft.com";

            var serializer = new Serializer();
            var content = serializer.SerializeAsJsonContent(fakeUser);

            var response = new HttpResponseMessage(HttpStatusCode.OK);
            response.Content = content;
            return Task.FromResult(response);
        }
    }

    return base.SendAsync(request, cancellationToken);
}

```

Ahora necesitamos incluir nuestro Handler en el GraphServiceClient, para ello, primero obtenemos los Handlers que vienen por defecto en el pipeline, llamando al método estático de la clase GraphClientFactory "CreateDefaultHandlers". Seguidamente, podemos añadir nuestro Handler a la lista de handlers.

```

var authProvider = new DeviceCodeProvider(clientApplication, scopes);

var defaultHandlers = GraphClientFactory.CreateDefaultHandlers(authProvider);
defaultHandlers.Add(new GraphMeSimulator());

```

Ahora sólo queda crear un HttpClient con nuestra lista de Handlers, y pasar dicho HttpClient al objeto GraphServiceClient:

```

var httpClient = GraphClientFactory.Create(defaultHandlers);

var graphClient = new GraphServiceClient(httpClient);

var me = await graphClient.Me.Request().GetAsync();
Console.WriteLine(me.DisplayName);

```

Si ejecutamos el código, vemos como nuestro "Fake User" aparece en pantalla:

```

> dotnet run
To sign in, use a web browser to open the page https://micro
o authenticate.
/beta/me
Mr Fake User

```

Como hemos visto, el SDK es un gran aliado para nuestro día a día consumiendo la Graph API. Además de abstraernos de la complejidad de montar la petición y gestionar la respuesta, nos ofrece un gran nivel de personalización.

¡Hasta el próximo artículo!

---

**Luis Mañez**

Cloud Architect en ClearPeople LTD

@luismanez

<https://github.com/luismanez>

# Graph API y Power Automate, una combinación increíble

## ¿Por qué usar Graph API si Power Automate tiene cientos de conectores?

En algunos casos no disponemos de todos los conectores que nos gustaría tener dentro de Power Automate, así que podemos hacer uso de la API de Graph para ampliar las capacidades que nos ofrece la herramienta. En otras ocasiones, esos conectores de los que disponemos se ejecutan con la conexión del usuario que lo creó. Aunque la opción de “Run-Only User” nos permite solucionar esto último esta no siempre está disponible.

¿Cuándo la tendremos disponible? En los flujos ejecutados de forma instantánea con botón “Manually trigger a flow” o en aquellos flujos que se generen desde SharePoint con el desencadenador “For a selected item”.

***“podemos hacer uso de la API de Graph para ampliar las capacidades que nos ofrece la herramienta”***

Aprenda a compartir un flujo de nube con otros usuarios - Power Automate | Microsoft Docs

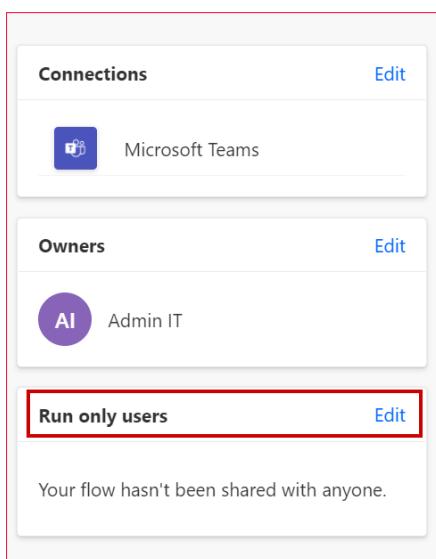


Imagen 1.- Opción de “Run only users” en un flujo de desencadenamiento manual.

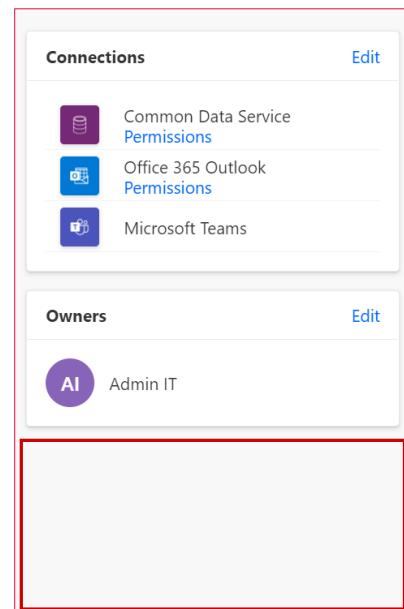


Imagen 2.- Opción inexistente de “Run only users” en un flujo automatizado con dataverse.

Por ejemplo, si quisiéramos usar el conector de Microsoft Teams para crear un equipo cuando se crea un registro en dataverse no vamos a tener disponible esta opción y por tanto el propietario del equipo será el usuario que haya creado la conexión con el paso de Power Automate de Microsoft Teams.

Si lo que queremos es que el propietario del equipo de Teams sea la persona que lo ha solicitado o creado el registro en dataverse o incluso su responsable deberemos usar Graph API.

Además, no todas las funcionalidades están disponibles desde los conectores estándares de Microsoft Teams u otros servicios. Como, por ejemplo, si continuamos hablando de los conectores de Microsoft Teams, el crear un equipo en base a una plantilla de Microsoft Teams o incluso gestionar los permisos de los miembro e invitados en el mismo momento de la creación, no se puede realizar y por ello deberemos utilizar la API de Graph.

Crear una plantilla de equipo personalizada en Microsoft Teams - Microsoft Teams | Microsoft Docs <https://docs.microsoft.com/es-es/microsoftteams/create-a-team-template#:~:text=Inicie%20sesi%C3%B3n%20en%20el%20centro%20de%20administraci%C3%B3n%20de%20Teams.&text=En%20el%20navegaci%C3%B3n%20izquierdo%2C%20expanda,Haga%20clic%20en%20Aregar%20.>

# ¿Cómo utilizar la API Graph en Power Automate?

Lo primero será crear un registro de aplicación en nuestro Azure AD y asignarle los permisos necesarios para el comando que va a ejecutar. Para conocer esos permisos simplemente debemos consultarlos en la misma documentación de la API.



**Crear equipo**  
 09/12/2020 • Tiempo de lectura: 11 minutos •

Espacio de nombres: microsoft.graph

Cree un [equipo](#).

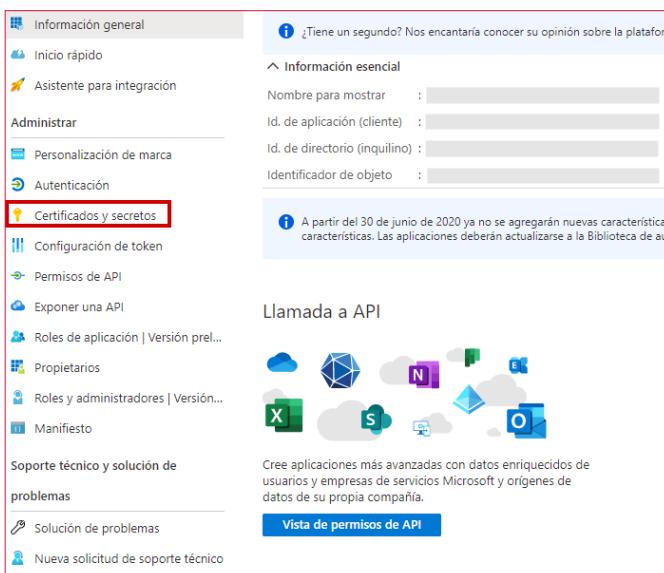
## Permisos

Se requiere uno de los siguientes permisos para llamar a esta API. Para obtener más información, incluido cómo elegir permisos, vea [Permisos](#).

Tipo de permiso	Permisos (de menos a más privilegiados)
Delegado (cuenta profesional o educativa)	Team.Create, Group.ReadWrite.All, Directory.ReadWrite.All
Delegado (cuenta personal de Microsoft)	No admitida.
Aplicación	Team.Create, Teamwork.Migrate.All, Group.ReadWrite.All, Directory.ReadWrite.All

Imagen 3.- Ejemplo de cómo se muestran los permisos necesarios para poder ejecutar la API de Graph.

Después necesitaremos crear un secreto para este registro de aplicación.



Información general

- Inicio rápido
- Asistente para integración
- Administrador
- Personalización de marca
- Autenticación
- Certificados y secretos**
- Configuración de token
- Permisos de API
- Exponer una API
- Roles de aplicación | Versión prel...
- Propietarios
- Roles y administradores | Versión...
- Manifiesto

Información esencial

Nombre para mostrar :

Id. de aplicación (cliente) :

Id. de directorio (inquilino) :

Identificador de objeto :

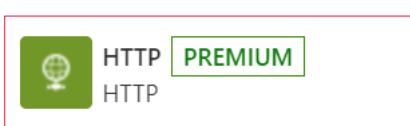
Llamada a API

Cree aplicaciones más avanzadas con datos enriquecidos de usuarios y empresas de servicios Microsoft y orígenes de datos de su propia compañía.

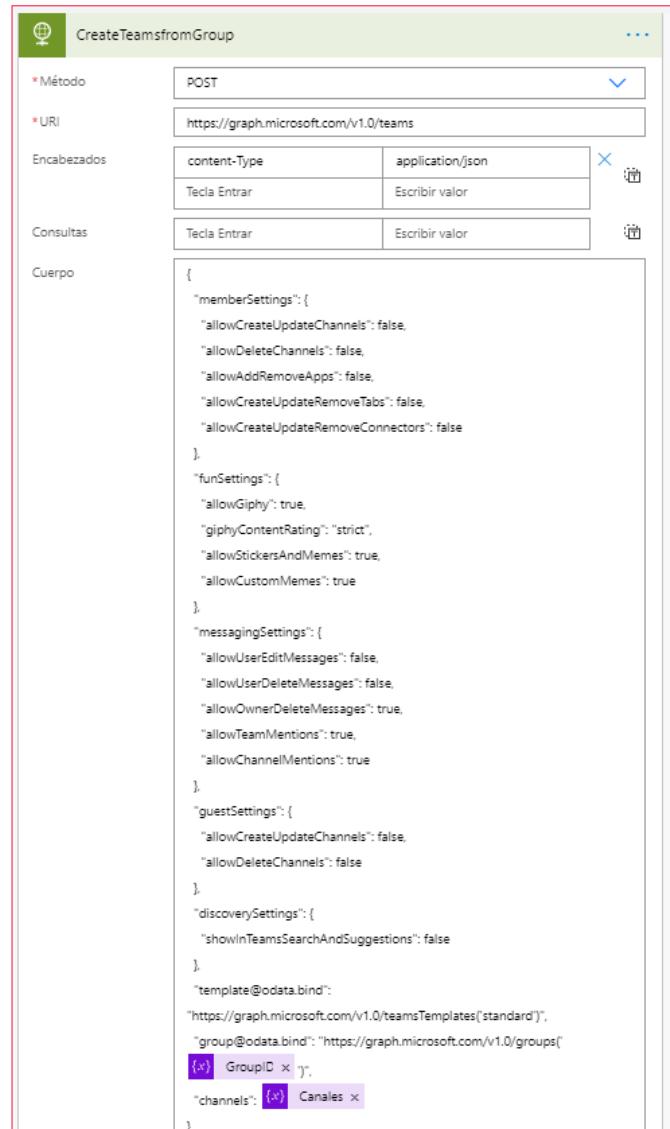
Vista de permisos de API

Imagen 4.- Cómo crear un secreto en nuestro registro de aplicación.

Finalmente, necesitaremos incorporar a nuestra Power Automate un conector del tipo HTTP que lo encontraremos dentro de los conectores premium, lo que significa que deberemos tener una licencia que nos permita el uso de este tipo de conectores.



En el siguiente ejemplo, se muestra como crear un equipo de Microsoft Teams en base a una plantilla, definiendo permisos y canales adicionales que no estén en la misma. Como se muestra, algunos datos son variables que se han generado durante el desencadenamiento del Power Automate. Los canales son una variable de tipo matriz que se compone previamente y el ID de grupo de Microsoft 365 lo obtenemos de la creación previa donde le indicamos los miembros y propietarios.



**CreateTeamsfromGroup**

Método: POST  
 URI: <https://graph.microsoft.com/v1.0/teams>

Encabezados:

content-Type	application/json
Tecla Entrar	Escribir valor

Consultas:

Tecla Entrar	Escribir valor
--------------	----------------

Cuerpo:

```
{
  "memberSettings": {
    "allowCreateUpdateChannels": false,
    "allowDeleteChannels": false,
    "allowAddRemoveApps": false,
    "allowCreateUpdateRemoveTabs": false,
    "allowCreateUpdateRemoveConnectors": false
  },
  "funSettings": {
    "allowGiphy": true,
    "giphyContentRating": "strict",
    "allowStickersAndMemes": true,
    "allowCustomMemes": true
  },
  "messagingSettings": {
    "allowUserEditMessages": false,
    "allowOwnerDeleteMessages": false,
    "allowTeamMentions": true,
    "allowChannelMentions": true
  },
  "guestSettings": {
    "allowCreateUpdateChannels": false,
    "allowDeleteChannels": false
  },
  "discoverySettings": {
    "showInTeamsSearchAndSuggestions": false
  },
  "template@odata.bind": "https://graph.microsoft.com/v1.0/teamsTemplates('standard')",
  "group@odata.bind": "https://graph.microsoft.com/v1.0/groups({{x}}/Group{{x}})",
  "channels": {{x}} Canales {{x}}
}
```

Imagen 5.- Ejemplo de utilización de la API de Graph en Power Automate para la creación de un equipo de Teams.

**"necesitaremos incorporar a nuestra Power Automate un conector del tipo HTTP que lo encontraremos dentro de los conectores premium,"**

En este conector, nos pedirá que introduzcamos el método de autenticación que se debe elegir para desencadenar la API Graph y es aquí donde es necesario nuestro registro de aplicación y nuestro secreto. Elegiremos el método de autenticación "Active Directory OAuth" e introduciremos los datos según la información de nuestro registro de aplicación.

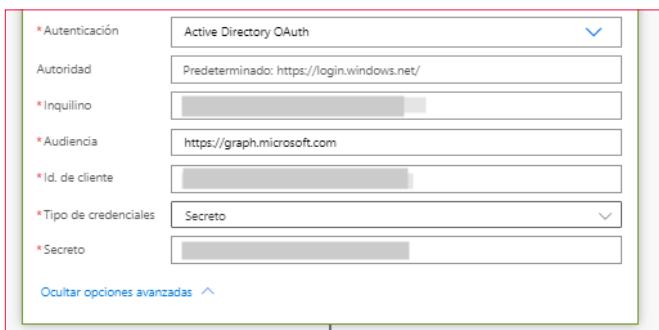


Imagen 6.- Configuración de la autenticación y autorización del conector.

How to authenticate as an application with Microsoft Graph API with flow ~ Benitez Here. <https://benitezhere.blogspot.com/2020/12/how-to-authenticate-as-application-with-microsoft-graph-api-with-flow.html>

Por último, para garantizar la seguridad de nuestro secreto, podemos utilizar Azure Key Vault y llamar al servicio desde Power Automate para que nuestro secreto permanecerá encriptado.



Imagen 7.- Conectores con Azure Key Vault en Power Automate.

Por lo tanto, como hemos observado, aunque con los conectores de Power Automate estándar no disponga dentro de su abanico de posibilidades la funcionalidad que necesitamos, a través del conector de HTTP y un registro de aplicación, podemos conectar con la API de Graph y extender la funcionalidad de nuestro Power Automate.

### MAR PEDROCHE

**Modern Workplace Solutions Specialist**

*LinkedIn*

*Twitter: @MarPedroche*

i

39

## Nosotros



### 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 ENCAMILA, 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.

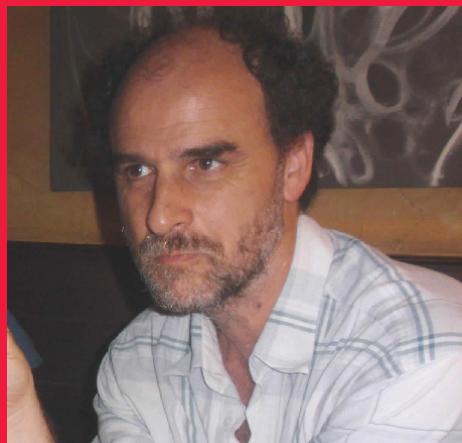
Desde los comienzos Fabián ha trabajado en distintas comunidades donde organiza y promueve eventos locales para la difusión de tecnología dentro de los miembros de las mismas. Es director de la carrera SharePoint 2010 y SharePoint 2013 en Microsoft Virtual Academy, <http://www.mslatam.com/latam/technet/mva2/Home.aspx>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

