

# Guía para desarrolladores sobre Azure



## 03 /

### Introducción

## 06 /

### Introducción a Azure

- 07 Ejecutar lo que sea en Azure
- 07 Seleccionar los servicios de Azure indicados
- 13 Acelerar la aplicación
- 14 Dónde almacenar los datos
- 18 Azure Data Analytics
- 22 Tutorial 1: La experiencia del portal de Azure

## 27 /

### Asegurar la aplicación

- 28 Azure Active Directory
- 28 Azure API Management
- 28 Azure AD Application Proxy
- 28 Azure Key Vault
- 29 Managed Service Identity
- 29 Cifrado
- 29 Azure Security Center
- 30 Protección contra DDoS de Azure
- 30 Azure Application Gateway
- 30 Firewall de aplicaciones web Azure
- 30 Azure Network Watcher
- 31 Grupos de seguridad de red
- 31 Zonas privadas de DNS de Azure
- 31 Redes virtuales privadas entre instalaciones
- 31 Azure ExpressRoute
- 31 Equilibradores de carga
- 32 Registro y monitoreo
- 32 Documentación técnica y de arquitectura de seguridad de Azure

## 33 /

### Añadir inteligencia a la aplicación

- 34 Azure Search
- 34 Azure Cognitive Services
- 39 Azure Bot Service
- 39 Azure Machine Learning Studio
- 41 Azure Maps
- 41 Herramientas para el desarrollador de IA
- 42 Uso de eventos y mensajes en la aplicación

## 46 /

### Trabajar con la IoT y entenderla

- 47 Azure IoT Hub
- 48 Azure IoT Central
- 48 Aceleradores de soluciones de Azure IoT
- 48 Azure IoT Edge
- 49 Más información acerca de Azure IoT
- 50 Qué usar y en qué momento

## 51 /

### Dónde y cómo implementar los servicios de Azure

- 52 Infraestructura como código
- 53 Azure Service Fabric
- 53 Contenedores en Azure
- 54 Azure Stack
- 55 ¿Dónde y cuándo implementar?

## 56 /

### Microsoft Azure en acción

- 57 Tutorial 2: Desarrollar una aplicación web y una base de datos en Azure
- 62 Tutorial 3: Extender aplicaciones con Azure Logic Apps y Cognitive Services
- 68 Tutorial 4: Todo listo para la producción
- 74 Tutorial 5: Uso de Azure API Management para controlar API y generar documentación

Si usted es un desarrollador o arquitecto que desea comenzar a usar Microsoft Azure, este libro es para usted.

Escrita por desarrolladores y para desarrolladores, esta guía le mostrará cómo comenzar con Azure y cuáles servicios puede utilizar para ejecutar sus aplicaciones, almacenar sus datos, incorporar inteligencia, desarrollar aplicaciones de IoT e implementar sus soluciones en forma más eficiente y segura. Antes de profundizar, tomémonos un momento para observar qué puede hacer la nube (y específicamente Azure).

## ¿Qué es Azure y qué puede hacer por usted?

Azure ofrece servicios que pueden ayudarle a lograr muchas cosas. Estos van desde objetivos mundanos, como crear una máquina virtual (VM) y cargar software o implementar una nueva base de datos de SQL, y otros complejos, como implementar flujos de trabajo de integración continua (CI) y entrega continua (CD) o trabajar con contenedores. También puede ajustar automáticamente su base de datos o diagnosticar problemas en la aplicación web en forma fácil y rápida. En el pasado, los desarrolladores debían crear repetidamente estos y otros proyectos comunes para sí mismos, pero ahora estos están disponibles como servicio. Además, puede usar estos servicios con poco esfuerzo, casi como encender o apagar una luz. Así puede concentrarse en las piezas de la aplicación que la hacen única, las características que aportan verdadero valor agregado para los usuarios.

El poder de la nube radica en que los servicios y los recursos son increíblemente sólidos y resistentes. Es muy poco probable que se produzcan errores de ejecución, ya que la nube es inteligente y los errores se reparan automáticamente. Con Azure, hay

centros de datos en todo el mundo, con decenas de miles de servidores. Si un servidor falla, otro toma el control. Si fallara un centro de datos completo (lo que es poco probable), otro asumiría sus funciones. Todo esto es posible debido a la escala masiva de la nube. De hecho, Microsoft ofrece Acuerdos de nivel de servicio (SLA) que prometen el funcionamiento de los servicios, de lo contrario se le devolverá su dinero. Azure ofrece uno de los mejores SLA de la industria por sus servicios (Tabla 1-1).

Uno de los argumentos más convincentes en favor de la nube es que usted deja de ser propietario de los servidores. Esto significa que no es responsable de comprarlos y mantenerlos funcionando. Además, no tiene que preocuparse de la configuración de red del servidor. Todo se hace por usted, incluso asegurarse de que los servidores se mantengan fríos y seguros, y del pago de la factura de electricidad.

Además, puede escalar los servicios y recursos casi de forma indefinida con solo hacer unos clics. Esto no se puede hacer con recursos locales, a menos que esté preparado para gastar enormes sumas de dinero en bienes de capital y personal administrador. También puede escalar a nivel mundial. Puede ofrecer sus servicios en cualquier lugar del mundo y una experiencia de alto

**Tabla 1-1:** SLA para los servicios de Azure

Servicio	SLA (disponibilidad)
Azure SQL Database	99,99%
Cosmos DB	99,999 % en lectura; 99,99 % en escritura
Azure Storage	Hasta 99,99 % si se usa el almacenamiento de información replicada geográficamente con acceso de lectura (RA-GRS)
App Service	99,95%
Máquina virtual	99,95 % (cuando se implementan dos instancias)
Cognitive Services	99,9%

rendimiento a los usuarios, sin importar dónde estén. Esto también significa que puede guardar los datos donde los necesite. Igual de importante es que al usar recursos en la nube, puede reducir la escala de los servicios y recursos cuando la demanda baja.

Además de escalabilidad masiva, servicios inteligentes listos para implementar y la eficiencia del pago por uso, la nube ofrece más seguridad. Millones de personas usan la nube todo el día, todos los días. Por supuesto, también es un objetivo habitual de ataques. Los proveedores de servicios en la nube con buena reputación y experiencia, como Microsoft, reconocen los patrones de uso de los usuarios normales y los de los agentes maliciosos. Es decir que sabemos cómo protegerlo de los ataques más comunes y los más extraños. Las herramientas de monitoreo inteligentes, los algoritmos de "machine learning" y la inteligencia artificial les dan a los proveedores de servicios en la nube la capacidad de detectar y detener ataques en tiempo real. Las décadas de experiencia en seguridad y el tráfico a escala masiva combinados con la mayor experiencia en seguridad del sector hacen que la nube sea un entorno mucho más seguro que cualquier centro de datos local.

Para obtener más información acerca de cómo Azure protege sus aplicaciones y datos, consulte la [documentación oficial de Azure Security](#) y la [información general de Azure Security](#).

## Estamos aquí para ayudarlo

Si necesita ayuda, podemos asistirlo de diversas formas. Contamos con planes de soporte que le brindan acceso a los equipos de soporte técnico de Azure y que proporcionan otros servicios, incluida la orientación para diseño en la nube y asistencia

con la planificación de la migración. De acuerdo con sus necesidades, puede adquirir un plan de soporte que le garantiza una respuesta dentro de 15 minutos por parte de los equipos de soporte técnico. También puede obtener ayuda a través de otros canales, tales como:

[Documentación y guías](#) que le entregan información general de todo lo que ocurre en Azure y le proporcionan información profunda a través de la documentación de cada característica.

[Comunidad de soporte](#), que contiene respuestas a las preguntas de la comunidad y proporciona un lugar para discusión con la comunidad de Azure.

[@AzureSupport](#) en Twitter, una cuenta administrada por ingenieros expertos en Azure que responden rápidamente a los problemas que les envía por este medio.

[Azure Friday](#), que da un vistazo a Azure Services y a sus características mediante el equipo de ingenieros de Microsoft.

[Azure Advisor](#), que realiza recomendaciones personalizadas automáticas para sus recursos de Azure, incluidas aquellas que necesita para ser más seguro, tener mayor disponibilidad, mejorar el rendimiento y reducir los costos.

[Azure Service Health](#), que le entrega una vista personalizada del estado de sus servicios de Azure.

[Stack Overflow](#), que ofrece respuestas a preguntas de Azure e incluye muchas publicaciones activas de los miembros de los equipos de ingeniería de Azure.

# Introducción a Azure

Ya se unió a la comunidad Azure y ahora quiere empezar a crear aplicaciones. ¿Qué necesita? En realidad, no mucho. Lo más importante que debe tener es una cuenta en Azure para implementar la aplicación. Puede usar las herramientas, aplicaciones y marcos de su elección, y luego puede comenzar a ejecutar sus aplicaciones en Azure.

## ejecutar lo que sea en Azure

Azure es ideal para aplicaciones web y de API. También es ideal para hospedar aplicaciones de escritorio y móviles. Por ejemplo, podría utilizar Azure para autenticar a los usuarios en la aplicación de escritorio o enviar notificaciones push a la aplicación móvil. Azure puede agregarle valor a cualquier escenario que tenga en mente.

## Seleccionar los servicios de Azure indicados

### Dónde hospedar la aplicación

La primera decisión que deberá tomar es dónde hospedar la aplicación. Azure ofrece varias opciones de hospedaje. [Aquí](#) puede encontrar un árbol de decisiones que lo guiará con relación al mejor servicio para cada escenario.

### VM

Una de las maneras de hospedar la aplicación es en una VM en [Azure Virtual Machine](#). Esto le proporciona mucho control sobre cómo hospeda la aplicación, pero usted es responsable de mantener el entorno, y eso incluye crear parches del sistema operativo (OS) y mantener los programas antivirus actualizados.

Por ejemplo, puede usar una VM para probar la última versión de vista previa de Visual Studio sin "ensuciar" su máquina.


### Azure App Service

También puede hospedar las aplicaciones en una de las principales ofertas de Azure: [Azure App Service](#). Azure App Service es una colección de servicios de hospedaje y orquestación que comparte características y capacidades. Por ejemplo, todos los servicios de aplicación tienen la capacidad de asegurar una aplicación utilizando [Azure Active](#)

[Directory](#) y pueden usar dominios personalizados. Azure App Service consta de lo siguiente:

**Web App:** [Web App](#) es uno de los servicios más utilizados de Azure. Puede usarla para hospedar las aplicaciones web o API. Una Web App es básicamente una abstracción de un servidor web, como Internet Information Services (IIS) o Tomcat, que se utiliza para hospedar aplicaciones basadas en HTTP.

Web App puede hospedar aplicaciones que están escritas en .NET, Node.js, PHP, Java o Python, y existen extensiones que puede utilizar para ejecutar incluso más lenguajes.

 [En nuestra sección de ejemplos, lo guiamos por una .NET Core y una aplicación de SQL de muestra](#)

### Web App for Containers: [Web App for Containers](#)

lo ayuda a implementar y ejecutar fácilmente aplicaciones web en contenedores, a gran escala. Solo saque imágenes de contenedor de Docker Hub o de un Azure Container Registry privado y Web App for Containers implementará la aplicación en contenedores con sus dependencias seleccionadas y estará en producción en segundos. La plataforma se encarga automáticamente de los parches para el sistema operativo, el abastecimiento de capacidad y el equilibrio de carga.

**Mobile App:** [Mobile App](#) proporciona un back-end para las aplicaciones móviles. Usted recibe una API en Mobile App con la que las aplicaciones móviles se conectan a través del cliente SDK multiplataforma. Está disponible para iOS, Android y Windows. Mobile App ofrece características únicas como la sincronización sin conexión y las notificaciones push, que lo ayudan a crear una experiencia móvil segura, moderna y de alto rendimiento.

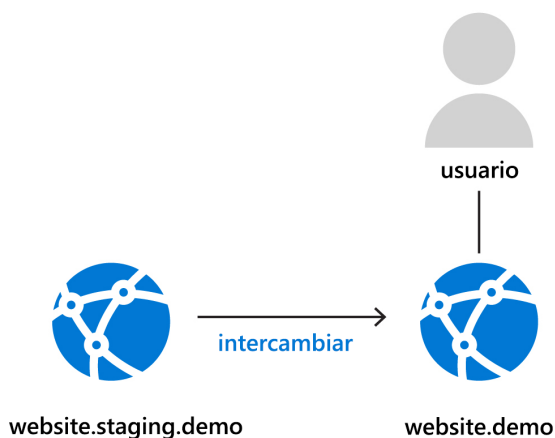
Puede escribir el back-end de Mobile App en .NET o Node.js.

## Características de Azure App Service

Azure App Service es uno de los principales servicios de Azure que puede utilizar para hospedar las aplicaciones. Cada uno de estos servicios aporta capacidades únicas, pero todos comparten algunas características comunes:

**Escalado:** Azure App Service se ejecuta en App Service Plans, que son abstracciones de VM. Una o más VM ejecutan Azure App Service, pero no hace falta que sepa cuáles, porque Azure se encarga de ellas. Sin embargo, puede ampliar los recursos que ejecutan Azure App Service. Puede elegir un nivel de precios superior (desde gratis hasta premium) o aumentar el número de instancias de la aplicación que se ejecutan. Incluso puede hacer que Azure App Service escale automáticamente el número de instancias para usted, en función de un calendario o métricas como la longitud de cola de CPU, memoria o HTTP.

**Ranuras de implementación:** Puede implementar una nueva versión de la aplicación a una ranura de implementación, donde luego podrá probar si funciona como se espera y pasarla a la ranura de la producción. Incluso puede utilizar Testing in Production, una función de Azure para enrutar un porcentaje del tráfico desde la aplicación de producción hasta la ranura de implementación.



**Figura 2-1:** Intercambio a una ranura de implementación

Por ejemplo, podría desviar al 10 % de los usuarios a la nueva versión de la aplicación en la ranura de implementación para ver si las nuevas características están funcionando como se esperaba y si los usuarios las están utilizando.

Cuando esté satisfecho con el funcionamiento de la nueva versión de la aplicación en la ranura de implementación, puede llevar a cabo un intercambio de esta aplicación con la de la ranura de producción. También puede cambiar de una ranura de desarrollo a una ranura de preparación y, luego, a la ranura de producción, como se ilustra en la figura 2-1. Antes de hacer esto, la operación de intercambio verifica que la nueva versión de su sitio web esté lista y en funcionamiento. Una vez que eso se confirma, la operación de intercambio cambia las ranuras y ahora los usuarios ven la nueva versión de la aplicación, sin tiempo de inactividad. Si lo desea, también puede hacer el intercambio de nuevo y revertir la implementación de la nueva versión.

Utilice ranuras de implementación en un entorno de desarrollo, prueba o producción. No utilice ranuras de implementación como entornos, porque residen en el mismo App Service Plan y es mejor que estén separados por motivos de seguridad, escalado, rendimiento y facturación.

Puede intercambiar las ranuras de implementación manualmente, a través de la interfaz de la línea de comandos (CLI) de Azure y a través de la API de administración de Azure. Esto permite que herramientas como Visual Studio Team Services realice operaciones de intercambio durante un lanzamiento.

Una ranura de implementación es otro elemento de Azure App Service (como una aplicación web) que se ejecuta en el mismo Azure App Service Plan, junto al Azure App Service original. Como las ranuras de implementación se ejecutan en el mismo Azure App Service Plan que el Azure App Service original, el uso de estas no tiene ningún costo adicional.



**Entrega continua:** Para publicar la aplicación en App Services, puede utilizar servicios externos como Visual Studio Team Services, Jenkins u Octopus Deploy. También puede utilizar la característica de [entrega continua \(CD\) de App Services](#). Esto le permite crear un proceso de creación-prueba-publicación en App Service. El proceso hace lo siguiente:

1. Recupera el último código fuente del depósito que le indique.
2. Crea el código según una plantilla que elija (ASP.NET, Node.js u otras).
3. Implementa la aplicación en un entorno de ensayo y realiza una prueba de carga.
4. Implementa la aplicación en producción después de la aprobación (puede elegir usar una ranura de implementación).

**Conectarse a recursos locales:** Puede conectar recursos externos, como depósitos de datos, a App Services. No es necesario que estos recursos estén en Azure. Pueden estar en cualquier parte, incluso de forma local, en su propio centro de datos. Puede conectarse a servicios locales a través de muchos mecanismos, dependiendo de sus necesidades. Puede utilizar [Azure Hybrid Connections](#), [Azure Virtual Networks](#) y [Azure ExpressRoute](#) para conectarse a los recursos locales.

### Dominios personalizados y certificados de Azure

**App Service:** Al darle un giro a una aplicación en Azure App Service, se expone una URL, por ejemplo, `https://myazurewebsite.azurewebsites.net`. Lo más probable es que tenga que utilizar su propio dominio personalizado, lo cual puede hacer mediante la asignación de ese nombre de dominio a App Services. [Aquí le mostramos cómo hacerlo](#).

Además, puede asegurarse de que la aplicación esté en HTTPS mediante el uso de un certificado de Capa de conexión segura (SSL). Puede traer su

propio certificado o comprar uno [directamente en el portal de Azure](#). Cuando compra un certificado SSL desde el portal de Azure, compra un certificado de Azure App Service, que puede configurar para que lo usen sus vínculos de dominio personalizado.



**Vea cómo [comprar y configurar un certificado en esta guía](#).**

**App Service Environment:** En una aplicación web de varios niveles, a menudo tiene una base de datos o servicios que utiliza la aplicación en Web App. Lo ideal sería que estos servicios solo estén expuestos a la aplicación y no a Internet. Por supuesto, la aplicación en sí misma suele estar orientada a Internet, ya que brinda el punto de entrada para los usuarios.

Para aislar estos servicios de soporte de Internet, puede utilizar una red virtual de Azure. Este servicio aborda sus servicios de soporte y los conecta a la aplicación en Web App de tal manera que los servicios de apoyo están expuestos solamente a la aplicación, no a Internet. [Este artículo](#) describe este servicio con más detalle y le muestra cómo utilizarlo.

Es posible que a veces necesite aún más control. Tal vez desea que la aplicación esté en una red virtual, para que usted pueda controlar el acceso a ella. Tal vez desea que otra aplicación la llame en Web App y hacer que sea parte de su back-end. En este caso, puede utilizar [Azure App Service Environment](#). Esto le brinda una escala muy alta y le otorga control sobre el aislamiento y el acceso a la red. Sin embargo, hay que tener en cuenta que App Service Environment está disponible para App Services solo en los niveles de precios premium.

**Nota:** App Service Environment actualmente no funciona con Web App for Containers.

### Azure Functions

Con Azure Functions, puede escribir solo el código

que necesita para una solución, sin tener que preocuparse por crear una aplicación completa o la infraestructura para ejecutarlo. Una función es una unidad de lógica de código que se desencadena debido a una solicitud HTTP, un evento en otro servicio Azure o en función de un calendario. Los enlaces de entrada y salida conectan el código de función con un código mínimo a otros servicios como Azure Storage, Azure Cosmos DB, Azure Service Bus e incluso servicios de terceros como Twilio y SendGrid. Con Functions, puede construir pequeñas piezas de funcionalidad rápidamente y hospedarlas en un medio elástico que gestiona la escala de modo automático.

Otro aspecto que hace que Azure Functions sea especial es que puede elegir pagar solo por las funciones que se ejecutan, sin tener que mantener instancias de informática en ejecución todo el mes. Esto también se llama sin servidor porque requiere solamente que usted cree una aplicación y no tenga que ocuparse de ningún servidor ni del escalado de servidores.

Puede escribir funciones de Azure en C#, F#, JavaScript y una lista de lenguajes cada vez mayor.

Un ejemplo de aplicación que utiliza Functions es aquella que activa una función todas las veces que se carga un archivo de imagen nuevo a Azure Blob Storage. En ese momento la función cambia el tamaño de la imagen y la escribe en otra cuenta de Blob Storage.

Los datos de Blob que activaron la función se pasan a la función como el parámetro myBlob, que incluye la URL de Blob. Puede utilizar el parámetro del enlace de salida outputBlob para especificar el Blob al que se debe escribir el resultado. No es necesario escribir la "plomería" para la conexión con Blob Storage; solo tiene que configurarla.

➔ [Cree la primera Función de Azure mediante el portal de Azure.](#)

## Azure Logic Apps

Puede organizar la lógica de negocio con [Logic Apps](#) y automatizar un proceso de negocios o integrarlo con aplicaciones de software como servicio (SaaS). Al igual que Azure Functions, a Logic Apps lo puede activar una fuente externa (por ejemplo, un nuevo mensaje en Azure Storage Queue). Una llamadas de API a los conectores para crear un flujo de trabajo (posiblemente complejo) que puede involucrar recursos en la nube y locales. Logic Apps tiene [muchos conectores a API](#) disponibles, como uno para la conexión con Azure SQL Databases, otro para Salesforce, para SAP y así sucesivamente. También puede exponer sus propias API o Azure Functions como conectores para utilizar en una Logic App, lo cual le permite realizar acciones fácilmente contra sistemas externos en un flujo de trabajo o lograr que uno de ellos active Logic App.

Al igual que Azure Functions, Logic Apps no tiene servidor, escala automáticamente y solo paga cuando las usa.

A continuación, un ejemplo del flujo de trabajo en Logic Apps:

1. Un correo electrónico que contiene una orden de envío que llega a Office 365 activa Logic App.
2. Mediante los datos del correo electrónico, Logic App comprueba la disponibilidad del artículo ordenado en SQL Server.
3. Logic App envía un mensaje de texto al teléfono del cliente utilizando Twilio (el número de teléfono también estaba en el correo electrónico) e indica que se recibió el pedido y se envió el artículo.

➔ [Introducción a Azure Logic Apps](#)

## Contenedores

Los contenedores son similares a las VM, sin

embargo, son mucho más livianos que estas, y puede iniciarlos y detenerlos en pocos segundos. También ofrecen una enorme portabilidad, que los hace ideales para desarrollar una aplicación localmente, en su máquina, y luego hospedarla en la nube, durante las pruebas y, más adelante, en la producción. Incluso puede ejecutar los contenedores de forma local o en otras nubes. El medio que utilice en su máquina de desarrollo viaja con su contenedor, por lo que la aplicación funciona siempre en el mismo entorno.

Al igual que las VM, los contenedores proporcionan un gran control sobre su entorno. Puede instalar lo que necesite para ejecutar las aplicaciones. Pero, una vez más, usted es el responsable de crear parches y de mantener el sistema operativo que se ejecuta en el contenedor, así como de los accesorios tales como los programas antivirus.

### **Hospede contenedores con Azure Container Instances**

Puede hospedar un contenedor con [Azure Container Instances](#). Azure Container Instances ofrece informática rápida y aislada para cumplir con los picos de tráfico, sin necesidad de administrar servidores. Por ejemplo, Azure Container Service (ACS) puede usar Virtual Kubelet para proveer pods en ACI que se inicien en cuestión de segundos. Esto permite que ACS se ejecute con la capacidad suficiente para una carga de trabajo promedio. Cuando se acaba la capacidad en el clúster de ACS, puede escalar pods adicionales en ACI sin que tenga que administrar servidores adicionales.

El servicio de Azure Container Instances se factura por segundo, por CPU virtual, por gigabyte o por memoria.



Para obtener más información sobre Container Instances, diríjase a [Introducción de Azure Container Instances](#)

### **Escale y organice contenedores con Azure**

### **Kubernetes Service**

Otra forma de hospedar contenedores es mediante el uso de Azure Kubernetes Service (AKS). AKS simplifica la creación, configuración y administración de un clúster de máquinas virtuales preconfiguradas para ejecutar contenedores. Esto significa que puede usar sus habilidades existentes y el conocimiento de la comunidad para administrar e implementar aplicaciones que se ejecutan en contenedores en Azure.

Al usar AKS, puede aprovechar todos los beneficios de Azure y mantener la portabilidad de la aplicación a través de Kubernetes y Docker.

El objetivo de AKS es proporcionar un entorno de hospedaje de contenedores utilizando herramientas y tecnologías de tipo open source. Debido a eso, AKS expone los puntos de conexión de la API de Kubernetes. Al usar estos puntos de conexión estándar, puede usar cualquier software que sea capaz de conversar con un clúster de Kubernetes, como kubectl, helm o draft.

AKS reduce la complejidad y la sobrecarga operativa de administrar un clúster de Kubernetes al quitar gran parte de la carga de esa responsabilidad de Azure. Como servicio de Kubernetes hospedado, Azure maneja tareas esenciales como supervisión del estado y mantenimiento por usted. Además, solo paga por los nodos de agente dentro de los clústeres, no por los dueños. Como servicio administrado de Kubernetes, AKS ofrece actualizaciones de versión y parches de Kubernetes automáticos, una escalación fácil de clústeres, un panel de control hospedado con recuperación automática (dueños) y ahorro, ya que solo paga por ejecutar nodos de grupo de agentes.

Con el manejo de la administración de nodos en su clúster de AKS en manos de Azure, hay muchas tareas que no debe realizar manualmente, como las actualizaciones de clústeres. Dado que Azure maneja estas tareas esenciales de mantenimiento por usted, AKS no proporciona acceso directo

(como con SSH) al clúster.

➔ [Para obtener más información acerca de AKS, vaya a \*\*Introducción a Azure Kubernetes Service\*\*](#)

## Azure Batch

Si necesita ejecutar aplicaciones de lotes a gran escala o de informática de alto rendimiento (HPC) en las VM, puede elegir [Azure Batch](#). Batch crea y administra una colección de hasta miles de VM, instala las aplicaciones que desea ejecutar y programa los trabajos en las VM. No es necesario implementar y administrar las VM individuales ni los clústeres de servidores. Batch programa, administra y escala automáticamente sus trabajos, para que

solo use las VM que necesita. Batch es un servicio gratuito; solo paga por los recursos subyacentes consumidos, como las VM, el almacenamiento y la creación de redes.

Batch es ideal para ejecutar cargas de trabajo paralelas a gran escala, como modelos de riesgos financieros, transcodificación de medios, representación de VFX e imágenes 3D, simulaciones de ingeniería y muchas otras aplicaciones de informática intensiva. Utilice Batch para escalar horizontalmente una aplicación o un script que ya haya ejecutado en estaciones de trabajo o en un clúster local, o desarrollar soluciones SaaS que utilicen batch como plataforma de informática.

➔ [Active Azure Batch en cinco minutos con estos tutoriales paso a paso](#)

**Tabla 2-1:** ¿Cuáles servicios de Azure se adaptan mejor a cuáles tipos de aplicaciones?

	Aplicaciones web	Web Apps for Containers	Mobile Apps	Functions	Logic Apps	Máquinas virtuales	Kubernetes Service	Service Fabric	Container Instances	Batch
Aplicaciones monolíticas y de nivel N	X	X				X*			X	X
Back-end de aplicación móvil			X			X*				
Aplicaciones de arquitecturas de microservicios				X			X	X		
Organización y flujos de trabajo de los procesos de negocio				X	X					
Trabajos informáticos intensivos										X
Ejecutar su aplicación en cualquier lugar (incluso en entornos locales)		X					X	X	X	

\* Para el método de traslado "lift-and-shift" de aplicaciones existentes a Azure.

## Azure Service Fabric

Otra forma de ejecutar aplicaciones en Azure es con [Azure Service Fabric](#). Este es el servicio que en verdad ejecuta muchos de los servicios de Azure dentro de Microsoft, como Azure SQL Database y Azure App Service. Puede ejecutar sus aplicaciones en Azure Service Fabric para lograr una disponibilidad alta, realizar una ejecución a escala masiva y realizar actualizaciones graduales.

Usa Azure Service Fabric para ejecutar aplicaciones basadas en microservicios .NET. Estas son soluciones que constan de muchos servicios pequeños que conversan entre sí y son empleados por interfaces de usuario y otros componentes. Service Fabric es ideal para soluciones como estas porque es excelente ejecutando componentes de aplicaciones con una alta disponibilidad y rendimiento, a la vez que los organiza.

Lo que hace que Azure Service Fabric sea único es que puede ejecutarlo en cualquier lugar. Puede instalar Service Fabric en su equipo de desarrollo local, en las instalaciones o en cualquier nube, incluida Azure. También puede usar Azure Service Fabric Mesh para ejecutar contenedores en un clúster de Service Fabric que Microsoft administra por usted como servicio. Esto abre muchas posibilidades.

Además, puede implementar aplicaciones en Azure Service Fabric y administrarlas con sus herramientas favoritas, como Visual Studio y Visual Studio Team Services. Además, Service Fabric recientemente se convirtió en open source.

## Qué usar y en qué momento

Algunos de los servicios que ejecutan su aplicación en Azure pueden trabajar en conjunto en una solución, mientras que otros son más idóneos para propósitos diferentes. Esto puede hacer que sea difícil elegir los servicios adecuados para usted. La Tabla 2-1 puede ayudarle a identificar cuáles son los mejores servicios de Azure para su situación.

**Figura 2-2:** Azure Traffic Manager dirige el tráfico al extremo geográfico de más alto rendimiento.



## Acelerar la aplicación

Una vez que la aplicación está funcionando en Azure, lo ideal es que rinda lo más posible. Azure proporciona una gama de servicios que lo pueden ayudar con eso.

### Azure Traffic Manager

Muchas aplicaciones modernas tienen usuarios en todo el mundo. Proporcionar una experiencia de alto rendimiento para todos es, como mínimo, un reto. El problema más obvio del que tiene que ocuparse es la latencia. La latencia es el tiempo que tarda una señal o una solicitud en llegar al usuario. Mientras más lejos se encuentren los usuarios de su aplicación, más latencia experimentan. [Azure Traffic Manager](#) escala entre regiones, lo que ayuda a reducir la latencia y a proporcionar a los usuarios una experiencia de alto rendimiento, sin importar de dónde son. Traffic Manager es un mecanismo de enrutamiento inteligente que usted coloca por delante, por ejemplo, de las aplicaciones de Web App, en todo el mundo. Web App actúa como extremos, que Azure Traffic Manager monitorea en cuanto a estado y rendimiento. Como se demuestra en la figura 2-2, cuando un usuario accede a su solicitud, Traffic Manager lo dirige a la aplicación de Web App que es más eficaz en su proximidad.

Incluir Traffic Manager en su arquitectura es una gran manera de mejorar el rendimiento de la aplicación.

## Azure Content Delivery Network

Uno de los servicios de Azure que lo pueden ayudar a acelerar la aplicación es Azure Content Delivery Network. Puede subir sus archivos estáticos, videos, imágenes, JavaScript, CSS y hasta archivos HTML estáticos a un depósito de datos como Blob Storage de Azure y luego emparejar Azure Content Delivery Network a ese contenido. Content Delivery Network luego tomará esos archivos estáticos y los replicará a cientos de puntos de presencia (PoP) en todo el mundo. Lo único que tiene que hacer en la aplicación es cambiar la referencia de los archivos estáticos a una URL diferente. Por ejemplo, si antes la referencia era `~/images/image.png`, ahora será `https://example.azureedge.com/image.png`. Esto es muy fácil de hacer y mejora el rendimiento de la aplicación de las siguientes maneras:

- Descarga contenido de servidores de la aplicación. Ahora utiliza Content Delivery Network como servidor y, por lo tanto, libera ciclos de procesamiento para la aplicación.
- Acerca el contenido estático físicamente a los usuarios distribuyéndolo a los PoP de todo el mundo.

Puede beneficiarse de Content Delivery Network en aplicaciones web, pero también en aplicaciones móviles y de escritorio.

Por ejemplo, Content Delivery Network se puede usar como servidor de videos para una aplicación móvil. Los videos pueden ser pesados y no quiere almacenarlos en el dispositivo móvil (ni tampoco quieren los usuarios). Si usa Content Delivery Network, se alojan en un PoP, lo cual también mejora el rendimiento, porque está cerca del usuario.



[Introducción a Azure Content Delivery Network](#)

## Azure Redis Cache

Toda aplicación moderna trabaja con datos. Al recuperar datos de un depósito de datos como una base de datos, por lo general es necesario analizar

múltiples tablas o documentos en algún servidor distante, reunir los resultados y, luego, enviar el resultado al dispositivo solicitante. Esto, por supuesto, lleva tiempo y puede frustrar y molestar a los usuarios.

Para eliminar algunos de estos "circuitos", puede almacenar datos que no cambian a menudo. De esta manera, en lugar de consultar la base de datos todas las veces, se podrían recuperar algunos de los datos de una memoria caché, como Azure Redis Cache. El beneficio de la memoria caché es que almacena los datos en un formato simple como valores con clave. No tiene que ejecutar una consulta compleja para obtener estos datos, basta con saber la clave para recuperar el valor. Esto puede mejorar el rendimiento de la aplicación en forma drástica. Así funciona el flujo de trabajo:

1. La aplicación necesita algunos datos e intenta recuperarlos de la memoria caché.
2. Si los datos no están, obténgalos de la base de datos y almacénelos en la memoria caché.
3. La próxima vez que la aplicación esté en busca de esos datos, los encontrará en la memoria caché y se ahorrará un viaje a la base de datos.

Azure proporciona memoria caché como servicio con Redis Cache. Esto se basa en el proyecto de fuente abierta de Redis y ahora está respaldado por los SLA líderes en la industria. Es altamente eficaz y tiene opciones avanzadas como la agrupación en clústeres y la georeplicación.



[Introducción a Azure Redis Cache](#)

## Dónde almacenar los datos

Los datos representan un aspecto muy importante de cualquier aplicación moderna y vienen en todas las formas y tamaños. Azure proporciona muchos tipos de depósitos de datos que pueden ayudarlo a mantener y recuperar datos en cualquier escenario.

**Tabla 2-2:** Opciones de almacenamiento en Azure

	SQL Database	MySQL	PostgreSQL	MariaDB	Cosmos DB	Blob	Tabla	Cola	Archivo	Disco	Data Lake Store	SQL Data Warehouse
Datos relacionales	X	X	X	X							X	X
Datos no estructurados					X	X					X	
Datos semiestructurados					X		X					
Mensajes en cola								X				
Archivos en disco									X			
Archivos de alto rendimiento en disco										X		
Depósito de datos de gran tamaño					X	X			X	X	X	X
Depósito de datos de tamaño pequeño	X	X	X	X	X	X	X	X	X	X		
Replicación de datos geográficos	X	X	X	X	X	X	X	X	X	X		
Coherencia de datos ajustables					X							

La tabla 2-2 en la página siguiente presenta las opciones de almacenamiento disponibles en Azure.

Puede utilizar casi todas las opciones de almacenamiento mencionadas en esta sección [como activadores y enlaces para Azure Functions](#).

Analicemos con mayor detalle cada opción de almacenamiento:

### Azure SQL Database

Si desea utilizar tablas con filas y columnas para almacenar datos, [Azure SQL Database](#) es una gran opción. Es un sistema de base de datos relacional similar a Microsoft SQL Server local. Como SQL Database se ejecuta en la nube, está completamente gestionado, es de alto rendimiento,

escalable y de respaldo automático, y tiene muchas características avanzadas.

Puede hacer [\(casi\)](#) todo lo que se puede hacer con SQL Server local. De hecho, las características nuevas de SQL Server se incorporan primero en Azure SQL Database y posteriormente en SQL Server local.

Puede utilizar SQL Database con sus herramientas favoritas, incluidas SQL Server Management Studio y Entity Framework. Estas son algunas de sus características más avanzadas:

- Georeplicación, que replica los datos a otras regiones geográficas en tiempo real ([Introducción a la georeplicación](#))



- Enmascaramiento de datos dinámicos, que enmascara dinámicamente datos confidenciales para determinados usuarios en tiempo de ejecución ([Introducción al enmascaramiento de datos dinámicos](#))
- Auditoría, que proporciona una auditoría completa de todas las acciones que les suceden a los datos ([Introducción a la auditoría de Azure SQL Database](#))
- Ajuste automático de la base de datos, que monitorea el rendimiento de su base de datos y la ajusta automáticamente ([Introducción al ajuste automático de Azure SQL Database](#))

Las bases de datos de SQL Database son extremadamente confiables y sólidas y ofrecen un acuerdo de nivel de servicio (SLA) que garantiza un tiempo de actividad del 99,9 %.

### Bases de datos de Azure para MySQL, PostgreSQL y MariaDB

Azure ofrece bases de datos de [MySQL](#), [PostgreSQL](#) y [MariaDB](#) como bases de datos administradas, lo que significa que simplemente las lanza y no tiene que preocuparse de la infraestructura subyacente. Al igual que Azure SQL Database y Cosmos DB, estas bases de datos están disponibles universalmente y son escalables, seguras y completamente administradas.

Cada una de ellas se adapta a casos de uso ligeramente diferentes, pero en general, su funcionalidad se traslapa bastante. Podría usar bases de datos de Azure para MySQL, PostgreSQL y MariaDB cuando ya usa sus versiones locales y desea aprovechar la ventaja de que estas se ejecuten completamente administradas en la nube.

### Azure Cosmos DB

[Azure Cosmos DB](#) es un nuevo tipo de base de datos que está realmente diseñado para la nube. Estas son algunas de sus características fundamentales:

- Tiene un SLA del 99,99 % (99,999 % para las operaciones de lectura) que incluye bajas latencias

**Figura 2-3:** Descripción general de los servicios de Azure Storage



(menos de 10 ms en lecturas; menos de 15 ms en escrituras).

- Cuenta con georeplicación, que replica los datos a otras regiones geográficas en tiempo real ([Cómo se distribuyen datos globalmente con Azure Cosmos DB](#)).
- [Niveles de coherencia de datos ajustables](#), para que pueda elegir la coherencia de los datos, lo que permite contar con un sistema de datos verdaderamente distribuido a nivel global. Por ejemplo, puede elegir una coherencia sólida, coherencia final o coherencia de sesión.
- Cuenta con administración del tráfico, que envía a los usuarios a la réplica de datos que más cerca tienen.
- La escala global no tiene límites y solo paga por el rendimiento de procesamiento y almacenamiento que necesita.
- [La indexación de los datos es automática](#). Ya no es necesario mantener ni ajustar la base de datos.



Además de estas características, Cosmos DB ofrece diferentes API con las que puede almacenar y recuperar datos, incluidas SQL, JavaScript, Gremlin, MongoDB, Azure Table Storage y Apache Cassandra. Diferentes API manejan los datos de diferentes maneras. Puede utilizar documentos como datos, como también tablas no estructuradas, gráficos y blobs. Utiliza la API que se adapta mejor a sus necesidades y Cosmos DB se encarga del resto.

Puede aprovechar el rendimiento, la escalabilidad y la fiabilidad del nivel de la nube, pero aun así utilizar el modelo de programación al que está acostumbrado.

### → [Introducción a Azure Cosmos DB](#)

## Azure Storage

[Azure Storage](#) es uno de los servicios más antiguos, confiables y de alto rendimiento de Azure. Ofrece cinco tipos de almacenamiento de los que todos se benefician, que tienen las siguientes características comunes:

- Georedundancia, que replica los datos a diferentes centros de datos para que pueda recuperarlos si un desastre genera el fallo de un centro de datos individual.
- Cifrado de datos en tiempo de ejecución.
- Dominios personalizados.

Los cinco tipos de Azure Storage son blob, tabla, cola, archivo y disco (Figura 2-3).

## Blob Storage

[Azure Blob Storage](#) almacena datos no estructurados de gran tamaño, es decir blobs de datos. Pueden ser videos, imágenes, audio, texto o incluso archivos de disco duro virtual (VHD) para VM.

Hay dos tipos de blobs: [de página y de bloque](#). Los blobs de página están optimizados para leer y escribir operaciones al azar. Son perfectos para el

almacenamiento de un disco duro virtual. Los blobs de bloque están optimizados para cargar grandes cantidades de datos con eficiencia. Son perfectos para almacenar grandes archivos de video que no cambian a menudo.

### → [Introducción a Azure Blob Storage](#)

## Almacenamiento de tablas

[Azure Table Storage](#) es un almacén de clave/valor NoSQL económico y muy rápido que puede usar para almacenar datos en tablas flexibles. Una tabla puede contener una fila que describe un pedido u otra fila que describe la información del cliente. No es necesario definir un esquema de datos. Esto hace que Table Storage sea muy flexible.

### → [Introducción a Azure Table Storage](#)

## Queue Storage

[Azure Queue Storage](#) es un tipo inusual de almacenamiento que se utiliza para almacenar pequeños mensajes de datos, pero su principal objetivo es actuar como cola. Pone mensajes en la cola y otros procesos los recogen. Este patrón desempareja el remitente del procesador de mensajes y eso resulta en beneficios de rendimiento y confiabilidad. Azure Queue Storage se basa en Microsoft Message Queuing, que puede encontrar en versiones anteriores de Windows.

### → [Introducción a Azure Queue Storage](#)

## File Storage

Puede utilizar [Azure File Storage](#) como una unidad desde la cual compartir archivos. Este usa el protocolo de Bloque de mensajes del servidor (SMB), que significa que puede utilizarlo con Windows y Linux, y puede acceder a él desde la nube o desde sistemas locales. Al igual que los otros tipos de Azure Storage, el almacenamiento en archivos es escalable y económico.

## → Introducción a [Azure File Storage](#)

### Almacenamiento en disco

[Azure Disk Storage](#) es similar a File Storage, pero está específicamente diseñado para rendimiento de E/S alto. Es perfecto para usar como disco en una VM que necesita alto rendimiento, como una VM que ejecuta SQL Server. Sin embargo, hay que tener en cuenta que el almacenamiento en disco está disponible solamente en el nivel de precios premium de Azure Storage.

### Azure Data Lake Store

Los depósitos de datos anteriores estaban destinados al uso de aplicaciones corrientes o al uso con máquinas virtuales. [Azure Data Lake Store](#) funciona como almacenamiento para aplicaciones de macrodatos. Se usa para almacenar grandes cantidades de datos en su formato nativo: estructurados, no estructurados o de cualquier otro tipo. El objetivo de Data Lake Store es tratar de conservar sus datos sin formato para que pueda analizarlos o transformarlos y moverlos. Las siguientes son las principales características de Azure Data Lake:

- Capacidad de almacenamiento ilimitada. Un solo archivo puede ser más grande que un petabyte en tamaño; 200 veces mayor que otras ofertas de los proveedores de servicios en la nube.
- Rendimiento escalable para alojar análisis paralelos de forma masiva.
- Puede almacenar datos en cualquier formato, sin un esquema.

Este es un enfoque muy diferente al depósito de datos tradicional, en el que se definen esquemas de datos por adelantado.

Por ejemplo, podría utilizar Data Lake Store para almacenar todos los datos que obtiene de sus dispositivos de Internet de las cosas (IoT) que están

recopilando datos de temperatura. Puede dejar los datos en la tienda y luego filtrar en esta para crear una vista de los datos por hora o por semana. Almacenar los datos en Data Lake Store es bastante económico, así que puede guardar datos durante mucho tiempo a un costo muy bajo.

## → Introducción a [Azure Data Lake Store mediante el portal de Azure](#)

### Azure SQL Data Warehouse

Se utiliza [Azure SQL Data Warehouse](#) cuando se necesita una solución tradicional de almacenamiento de datos totalmente administrada, escalable en tamaño, de alto rendimiento y segura. Puede almacenar datos en esquemas predefinidos y consultarlos utilizando el dialecto SQL de SQL Server.

Debido a que SQL Data Warehouse se ejecuta en Azure, puede usar funciones avanzadas, como la detección de amenazas automática que utiliza "machine learning" para entender los patrones de su carga de trabajo y sirven como un sistema de alarma para avisarle acerca de un posible incumplimiento.

Un ejemplo de uso de Azure SQL Data Warehouse es cuando sabe qué informes desea mostrarles a los usuarios y cuál es el esquema de datos de estos informes. Luego, puede crear esquemas en SQL Data Warehouse y completarlos con datos para que los usuarios puedan navegar a través de ellos.

## → Crear un [Azure SQL Data Warehouse](#)

## Azure Data Analytics

Casi tan importante como almacenar datos es obtener información al analizarlos. Azure ofrece muchos servicios para escenarios de análisis de

datos, lo que le permite obtener información valiosa y útil a partir de sus datos, sin importar su magnitud o complejidad.

## Azure Data Factory

Mover y transformar datos no es una tarea trivial. [Azure Data Factory](#) puede ayudarlo a hacer justo eso. Dentro de Data Factory, puede crear un proceso integral que ejecute el proceso completo de extracción, transformación y carga (ETL).

Data Factory puede mover datos en forma confiable desde ubicaciones locales a la nube, dentro de la nube o a ubicaciones locales, sin importar cuál es el origen de los datos. Data Factory también ofrece muchos conectores que puede usar para conectarse fácilmente al origen de datos, como SQL Server, Cosmos DB, Oracle y [muchos más](#).

Al mover datos, también puede filtrarlos antes de enviarlos a un destino final, limpiarlos o transformarlos con una actividad en el proceso, como una [Actividad de Apache Spark](#). Además, Azure Data Factory ofrece estas tres características útiles:

- Procesos de programación
- Procesos de monitor
- [Trasladar mediante el método "lift-and-shift" los paquetes de SQL Server Integration Services \(SSIS\) a Azure Data Factory](#)

 [Crear una fábrica de datos mediante el portal de Azure](#)

## Azure Analysis Services

Con [Azure Analysis Services](#), puede crear un modelo semántico de sus datos al que los usuarios pueden acceder directamente con herramientas de visualización como Power BI. El servicio literalmente se integra en las herramientas de [SQL Server Analysis Services](#) que se ejecutan localmente con SQL Server, sin embargo ahora se ejecutan de forma administrada, en la nube. Esto significa que el servicio puede escalarse y que los datos se almacenan de forma

redundante. Además, puede poner en pausa el servicio cuando no lo esté usando, para así minimizar los costos.

Con Azure Analysis Services, puede ofrecer datos modelados directamente a los usuarios, con un muy alto rendimiento. Los usuarios pueden consultar millones de registros en cuestión de segundos, ya que el modelo reside completamente in-memory y se actualiza periódicamente, por ejemplo, cada noche.

Puede obtener datos en el modelo semántico desde cualquier lugar, incluso desde orígenes de datos en la nube y locales. Puede usar Azure Blob Storage, Azure SQL Database, Azure SQL Data Warehouse y [muchos otros servicios](#) como orígenes de datos para el modelo. También puede usar orígenes de datos como Active Directory local y bases de datos de Access y Oracle.


 [Crear un servidor de Azure Analysis Services mediante el portal de Azure](#)

## Azure Data Lake Analytics

Otro servicio de Azure para realizar tareas de análisis de datos es [Azure Data Lake Analytics](#). Con este servicio, puede analizar, procesar y transformar potencialmente grandes cantidades de datos de Azure Storage y Azure Data Lake Store.

Usted usa Azure Data Lake Analytics mediante la creación y el envío de trabajos que consultan datos, los analizan o los transforman. Puede escribir estos trabajos en U-SQL, que es un lenguaje similar a SQL, y ampliar U-SQL con Microsoft R y Python.

Paga por los trabajos que envía y ejecuta, y el servicio escala automáticamente según el poder que necesitan los trabajos. Azure Data Lake Analytics normalmente se usa para trabajos de análisis de larga ejecución con enormes cantidades de datos.

 [Crear su primer script de U-SQL mediante el portal de Azure](#)

## Azure Stream Analytics

Puede usar el servicio de [Azure Stream Analytics](#) para analizar, consultar y filtrar datos de transmisión en tiempo real. Por ejemplo, cuando recibe un flujo de datos de temperatura de un dispositivo de IoT, este le indica qué tan cálido está el exterior. Podría proporcionar la misma temperatura cada segundo durante una hora hasta que la temperatura cambie, pero a usted solo le interesan esos cambios. Azure Stream Analytics puede consultar los datos en tiempo real y almacenar solo los datos diferenciales en Azure SQL Database.

Stream Analytics puede recibir datos de muchos servicios, como Azure Blob Storage, Azure Event Hub o Azure IoT Hub. Puede analizar los datos mediante un lenguaje simple similar a SQL o de un código personalizado. Después de consultar y filtrar el flujo de datos, Stream Analytics puede producir los resultados para muchos servicios de Azure, incluidos Azure SQL Database, Azure Storage y Azure Event Hub.

➔ [Crear un trabajo de análisis de flujo mediante el portal de Azure](#)

## Azure Time Series Insights

Puede usar [Azure Time Series Insights](#) para obtener información rápida sobre grandes cantidades de datos que normalmente son de tipo IoT. Este servicio obtiene datos de Azure Event Hub, IoT Hub y sus propias entradas de referencia, y retiene esos datos durante un cierto periodo, de aproximadamente un año.

Azure Time Series Insights es un servicio súper rápido, que permite a los usuarios consultar y analizar datos a través de una herramienta de visualización tan pronto como estos entran. Time Series Insights no solo analiza los datos, sino que también los ingiere y retiene por un tiempo. Esto es como Azure Analysis Services, donde los datos residen in-memory en un modelo que los usuarios pueden consultar. Las diferencias clave son que Time Series Insights está optimizado para IoT,

datos basados en el tiempo y contiene su propia herramienta de visualización de datos.

➔ [Explorar un entorno de demostración de Time Series Insights desde su navegador](#)

## Azure Databricks

[Azure Databricks](#) le permite ejecutar un clúster de Databricks administrado y escalable en la nube. Databricks ofrece una plataforma de análisis unificada con un host de herramientas y capacidades. Dentro de Databricks, ejecuta versiones optimizadas de Apache Spark para realizar análisis de datos avanzado.

Además del análisis basado en Spark, Databricks ofrece equipos interactivos y flujos de trabajo integrados, así como también espacios de trabajo que puede usar para colaborar con el equipo de datos completo, incluidos los científicos de datos, ingenieros de datos y analistas de negocios, todos ellos con acceso a herramientas especializadas para sus necesidades específicas.

Azure Databricks está completamente integrado con Azure Active Directory, que le brinda la posibilidad de implementar seguridad granular. Con Databricks, realiza análisis de datos basados en Spark sobre datos que provienen de muchos lugares, incluidos Azure Storage y Azure Data Lake Store. Databricks también funciona con datos de Azure SQL Data Warehouse, Azure SQL Database y Azure Cosmos DB. Además, puede conectar Databricks en Power BI para crear y mostrar paneles potentes.

➔ [Ejecute un trabajo de Spark en Azure Databricks mediante el portal de Azure](#)

## Azure HDInsight

[Azure HDInsight](#) es una plataforma dentro de Azure para ejecutar servicios de análisis de open source administrados. Puede usarla para ejecutar clústeres especializados de sus herramientas de análisis de datos de open source favoritas. La ventaja de

ejecutar estas herramientas en Azure HDInsight es que son administradas, lo que significa que no tiene que mantener VM o sistemas operativos con parches. Además, pueden escalar y conectarse fácilmente entre sí, con otros servicios de Azure y con orígenes y servicios de datos locales.

La mayoría de los tipos de clústeres de análisis de datos de open source especializados en Azure HDInsight usan Azure Blob Storage o Azure Data Lake Store para tener acceso a los datos o almacenarlos, ya que estos servicios funcionan con Hadoop File System.

Puede ejecutar potencialmente clústeres masivos especializados de diferentes tipos. Por ejemplo, puede ejecutar un **clúster de Apache Hadoop**. Esto le permite procesar y analizar datos con herramientas de Hadoop como Hive, Pig, Oozie y más.

También puede lanzar un **clúster de Apache HBase**, que ofrece una base de datos de NoSQL muy rápida.

Los datos en realidad se encuentran dentro de Azure Storage o un Azure Data Lake, pero HBase ofrece un nivel de abstracción en la parte superior, que tiene su propia funcionalidad y un rendimiento único.

Puede crear un **clúster de Apache Storm**, que se orienta hacia el análisis de flujos de datos, tal como Azure Stream Analytics. Además, puede tener un **clúster de Apache Spark**, que ofrece un marco para procesar y analizar grandes cantidades de datos.

HDInsight también puede ejecutar un clúster para **Microsoft Machine Learning Server** (anteriormente Microsoft R Server). Esto le permite ejecutar trabajos basados en R para analizar datos. Finalmente, puede crear un clúster que ejecute **Apache Kafka**, que es un sistema de mensajería de suscripción de publicación que se usa para desarrollar aplicaciones con mecanismos de formación de colas.

**Tabla 2-3:** Opciones de análisis de datos en Azure

	Data Factory	Analysis Services	Data Lake Analytics	Stream Analytics	Time Series Insights	Azure Databricks	Azure HDInsight
Mover datos de una tienda a otra	X						
Transformar datos	X	X	X	X	X	X	X
Consultar y filtrar datos de flujo				X		X	X
Proporcionar un modelo semántico in-memory para los usuarios		X			X		X
Los usuarios pueden consultar los datos y crear paneles					X		
Analizar datos para un uso posterior			X		X		X

# 01 /

## Tutorial: La experiencia del portal de Azure

Ahora que comprende qué servicios están disponibles en Azure, vamos a analizarlos con más detalle en un tutorial breve.

Una de las herramientas más importantes de Azure es Azure Portal, que se puede encontrar en <https://portal.azure.com/>. Este es su centro de Azure, donde hace todo lo que desea hacer con Azure. La mayoría de las cosas que puede hacer en el portal de Azure las puede hacer también a través de la API de Azure, la interfaz de la línea de comando (CLI) de Azure y Azure PowerShell.

Lo primero que verá en el portal de Azure es un panel con mosaicos. Puede crear y personalizar paneles y compartirlos con los miembros del equipo o conservarlos para usted mismo.

La Tabla 2-3 puede ayudarle a elegir los servicios correctos de Azure para analizar sus datos.

Existen más tipos de clúster, así como herramientas que puede usar en los clústeres. Puede realizar prácticamente cualquier tarea de análisis de datos y procesamiento con una combinación de estos clústeres, y todos ellos se ejecutan administrados en la nube.

➔ [Extraer, transformar y cargar datos usando Apache Hive en Azure HDInsight](#)

### Mosaicos en el portal de Azure

Los paneles contienen mosaicos que muestran información para un servicio o actúan como un acceso directo a un servicio. Puede encontrar estos mosaicos, que se muestra en la figura 2-4, en todo el portal, en las páginas de todos los servicios. Pueden ser muy útiles para obtener una vista rápida del estado de un servicio.

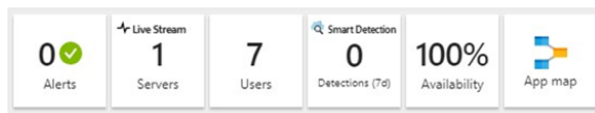


Figura 2-4: Mosaicos informativos en el portal de Azure

Por ejemplo, el servicio Application Insights, de forma predeterminada, muestra mosaicos que le informan si hay alguna alerta activa, si ingresan datos en vivo, cuántos usuarios han estado activos en las últimas 24 horas, cuál ha sido la disponibilidad, y otros datos. Se trata de información muy útil en un formato muy consumible. Puede personalizar el tamaño y la información de los mosaicos. También puede personalizar el aspecto de los gráficos mediante el ajuste de las líneas de tiempo y hacer que muestren datos en diferentes formatos como líneas o barras.

También puede fijar los mosaicos (Figura 2-5) directamente a sus paneles para que sean lo primero que vea cuando entre en el portal. Por ejemplo,

puede fijar mosaicos de todas las métricas de servicio que le interesan, para crear un panel de control que usted [comparta con su equipo](#), o mostrarlos en un monitor en la sala de equipo.

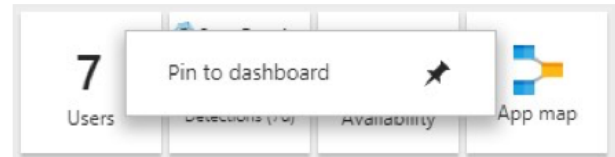


Figura 2-5: Puede fijar mosaicos a un panel

### Encontrar servicios

Los servicios de Azure son el tema central del portal de Azure. Puede agregarlos y buscarlos de varias maneras.

Para crear nuevos servicios, en la esquina superior izquierda de la ventana del portal, haga clic en el signo más ("+" ). Esto abre el cuadro de búsqueda de Marketplace, donde encontrará desde Web App hasta servidores de Linux, como se muestra en la figura 2-6.

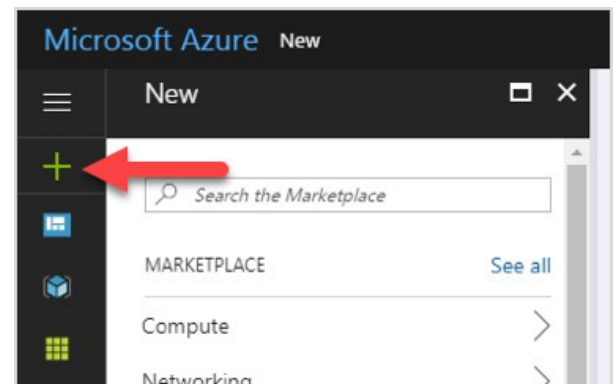


Figura 2-6: El botón Nuevo recurso en el portal de Azure

Cuando haya encontrado el servicio que desea (Figura 2-7), un asistente lo guiará para configurarlo e implementarlo.



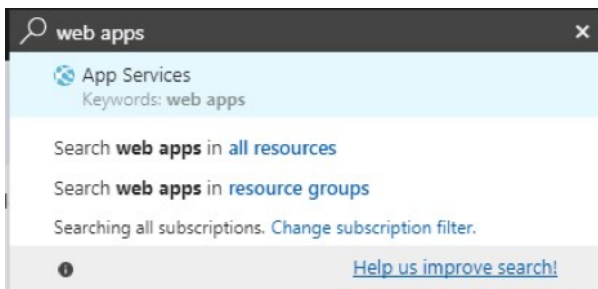


Figura 2-7: Resultados de la búsqueda de recursos

Cuando tenga algunos recursos, puede encontrarlos a través de la búsqueda. Puede utilizar el buscador en la parte superior del portal (figura 2-8) para buscar en todos sus recursos e ir directamente a ellos.

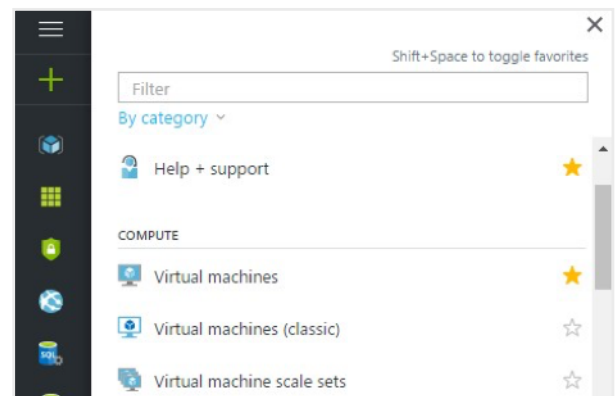


Figura 2-8: El menú de favoritos en el portal de Azure

En el panel en el lado izquierdo del portal se encuentra el menú de favoritos. Este menú

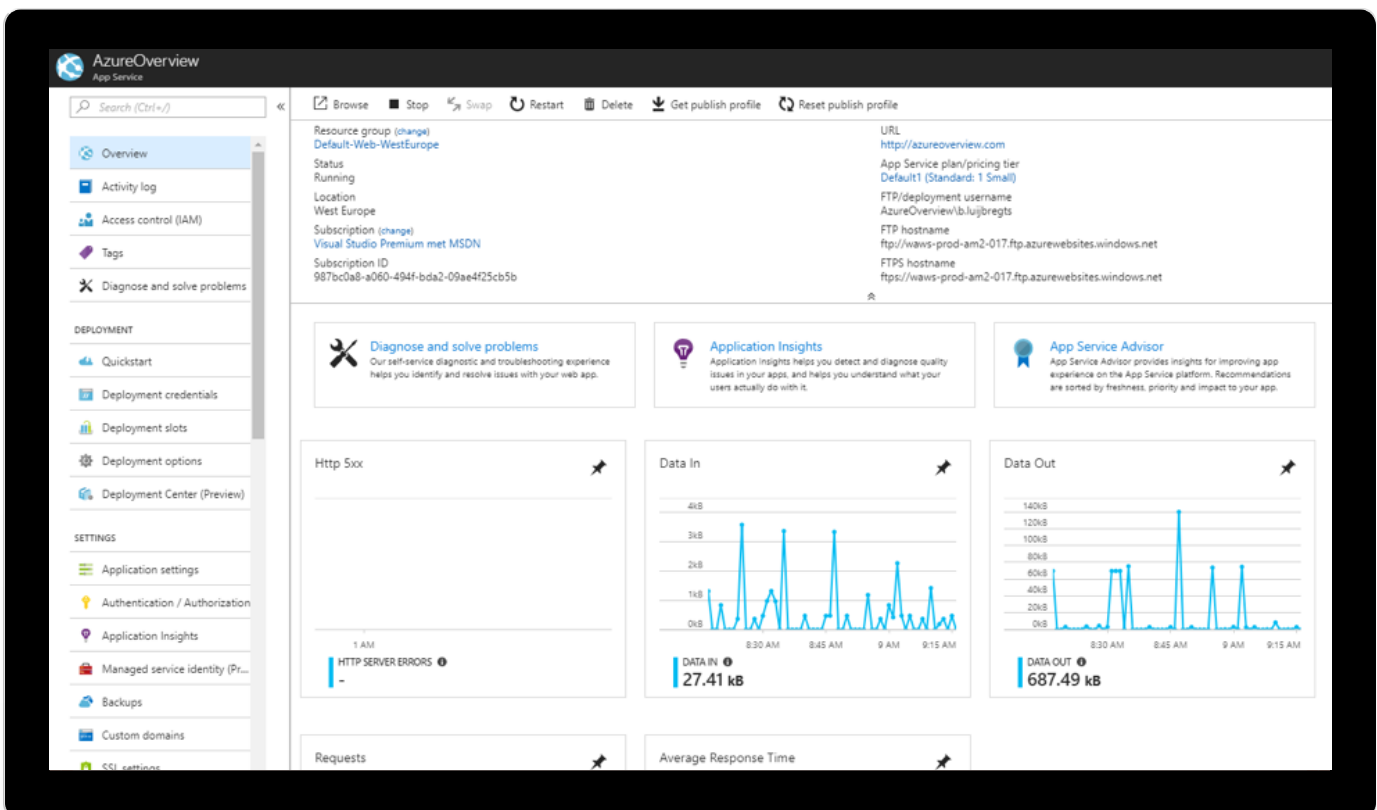


Figura 2-9: Hoja de resumen de App Service Web App



muestra las categorías de recursos (representadas por sus íconos), como Azure App Service. Puede reorganizar los iconos arrastrándolos hacia arriba y abajo. También puede seleccionar cuáles quiere ver expandiendo el menú de favoritos y haciendo clic en las estrellas de esas categorías.

### Explicación sobre las hojas

Las páginas en Azure también se llaman hojas. Las hojas están en todas partes, e incluso puede fijarlas a sus paneles. Cuando abra una aplicación web, primero verá la hoja de resumen (como en la Figura 2-9).

Esta hoja en particular le proporciona herramientas para detener, iniciar y reiniciar la aplicación web, y muestra mosaicos con sus métricas, como cantidad de solicitudes y errores. Si elige otro elemento del menú, se abre una nueva hoja. Las hojas se abren siempre en contexto. Por lo tanto, si abre la hoja Ranuras de implementación y hace clic en el botón Crear nueva ranura de implementación, aparece una nueva hoja a la derecha de la hoja Ranuras de implementación y, así, conserva el contexto en el que está.

### Creación de una nueva máquina virtual

Vamos a utilizar el portal de Azure para crear una nueva máquina virtual (VM). Después de crearla, veremos cómo apagarla y eliminarla para que ya no tenga que pagar por ella.

### Acerca de los grupos de recursos

*La VM se implementará en un grupo de recursos. Todos los recursos de Azure se alojan dentro de grupos de recursos. Un grupo de recursos es un contenedor lógico donde se encuentran sus recursos. Puede administrar la seguridad de un grupo de investigación y ver el costo de los recursos del grupo. Es una práctica común agrupar los servicios relacionados en un grupo de recursos para que sean fáciles de proteger y usted pueda saber fácilmente cuánto cuestan.*

1. En el portal de Azure, en la esquina superior izquierda, haga clic en el botón **Crear un nuevo servicio**.
2. En el cuadro Buscar, escriba **máquina virtual Windows Server**.
3. Haga clic en **Windows Server 2016 Datacenter**.
4. Haga clic en **Crear**. Se abre el asistente para crear la máquina virtual.
  - a. Elija un **nombre** para la VM.
  - b. Elija el **tipo de disco**. SSD proporciona una VM más rápida, pero también es más caro. Para este tutorial, seleccione **SSD**.
  - c. Escriba un **nombre de usuario**.
  - d. Seleccione **Contraseña** para el tipo de autenticación.
  - e. Escriba una **contraseña** y confírmela.
  - f. En el cuadro **Grupo de recursos**, escriba un nuevo nombre.
  - g. Elija la **ubicación** de la VM y haga clic en **Aceptar**.
5. Elija el **tamaño** de la VM.

El rendimiento de la VM determina cuánto paga por ella. Hay muchas opciones de tamaño para máquinas virtuales, algunas pequeñas, algunas muy grandes. Puede utilizar el asistente para seleccionar cuántos núcleos y cuánta memoria quiere, y elegir opciones basadas en esa información. Además, hay otras características que vienen con opciones de tamaño:

- a. Tipo de disco duro (SSD o disco duro normal).
- b. La cantidad de operaciones de entrada/salida por segundo (IOPS) máximas. Esto determina el rendimiento de la VM de forma significativa, en especial si las

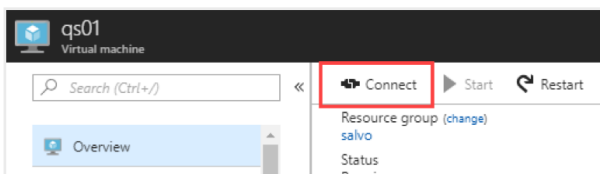
aplicaciones que ejecuta leen y escriben mucho en el disco duro.

- c. La cantidad de unidades de datos que pueden instalarse en la VM.
- d. La capacidad de equilibrio de cargas.
- e. La tarjeta gráfica que está instalada en la VM. Esto es útil si necesita hacer un montón de procesamiento de gráficos o pesadas cargas de trabajo computacionales.

6. Una vez que haya seleccionado el tamaño, puede especificar configuraciones adicionales como la red virtual, la dirección IP y las extensiones en la máquina. Por ahora, deje todo como está y haga clic en **Aceptar**.
7. Revise el resumen, acepte los términos y haga clic en **Crear**.

Ahora se implementará la VM. Generalmente esto toma pocos minutos. Después de esto, tendrá una VM ejecutándose en Azure. Cuando navegue a la VM en el portal de Azure, podrá ver cómo está, configurarla aún más e iniciar sesión usando RDP.

Para iniciar sesión en la VM usando RDP, haga clic en el botón **Conectar** en la hoja de resumen de VM en el portal de Azure (Figura 2-10). Esto activará una descarga del archivo RDP, que puede abrir y usar para conectarse a la VM.



**Figura 2-10:** El botón Conectar en la hoja de resumen de VM del portal de Azure

## Limpiar los recursos del tutorial

Si terminó de usar la VM, puede cerrarla y eliminarla borrando el grupo de recursos que creamos cuando generamos la VM. Esto contiene la VM y todos los recursos que se crean automáticamente. Una vez que se haya eliminado el grupo de recursos, usted ya no paga por ninguno de los recursos que ha utilizado en este tutorial.

# Asegurar la aplicación

La seguridad es uno de los aspectos más importantes de cualquier aplicación y no es fácil que funcione bien. Afortunadamente, Azure ofrece muchos servicios para ayudarlo a proteger la aplicación. Analizamos algunos en este capítulo.

## Azure Active Directory

Una parte importante de la seguridad de la aplicación es la autenticación de usuarios antes de que puedan usarla. La autenticación no es algo fácil de implementar. Es necesario guardar las identidades de los usuarios y las credenciales en algún lugar, implementar la gestión de contraseñas, crear un protocolo de seguridad de autenticación y más.

[Azure Active Directory](#) proporciona todo esto y mucho más, de inmediato. Guarda las identidades de los usuarios en Azure Active Directory y hace que los usuarios las autenticuen, ya que los redirige a la aplicación solo después de que se autentican. Azure Active Directory se encarga de la gestión de contraseñas, incluidos los escenarios comunes como "Olvidé mi contraseña".

Millones de aplicaciones utilizan Azure Active Directory a diario, entre ellas, el [portal de Azure](#), [Outlook.com](#) y [Office 365](#). Debido a esto, es capaz de detectar comportamiento malicioso más fácilmente y abordarlo. Por ejemplo, si un usuario iniciara sesión en una aplicación desde una ubicación en Europa y, luego, un minuto más tarde, en Australia, Azure Active Directory marcaría esto como comportamiento malintencionado y le preguntaría al usuario las credenciales adicionales a través de la autenticación multifactorial.

## Azure API Management

Las interfaces de programación de aplicaciones (API) deben ser seguras. Es necesario tanto para las API que crea por su cuenta, como para las que consume de terceros. Para ayudarlo a que sus API estén seguras, puede utilizar [Azure API Management](#). Esto es básicamente un proxy que pone delante de sus API, que añade características como el almacenamiento en memoria caché, la regulación y la autenticación/autorización.

Con API Management, puede asegurar una API al solicitar a los usuarios que creen una suscripción a ella. De esta forma, las aplicaciones tienen que autenticarse antes de poder usar su API. Puede utilizar varios métodos de autenticación, como tokens, autenticación básica y certificados. Además, puede rastrear quién llama a su API y bloquear llamadas no deseadas.

## Mucho más que seguridad

*Si bien la seguridad es esencial, Azure API Management ofrece otras capacidades que pueden ayudarlo a optimizar su flujo de trabajo de desarrollo y prueba, como [burlar respuestas de datos de prueba](#), [publicar múltiples versiones de API](#), [introducir cambios ininterrumpidos de forma segura con revisiones](#), y brindar acceso a los desarrolladores a la documentación de su API generada automáticamente, al catálogo de API y a muestras de código.*



Introducción a [Azure API Management](#)

## Azure AD Application Proxy

[Azure AD Application Proxy](#) proporciona inicio de sesión único (SSO) y acceso remoto seguro para las aplicaciones web hospedadas localmente. Las aplicaciones que probablemente desee publicar incluyen sitios de SharePoint, Outlook Web Access u otras aplicaciones web de línea de negocio (LOB). Estas aplicaciones web locales se integran con Azure AD, la misma plataforma de identidad y control utilizada por Office 365. Los usuarios finales pueden tener acceso a sus aplicaciones locales de la misma forma que acceden a Office 365 y otras aplicaciones de software como servicio (SaaS) integradas con Azure AD.

## Azure Key Vault

Como parte de su arquitectura de seguridad, necesita un lugar seguro para almacenar y administrar certificados, llaves y otros secretos. [Azure Key Vault](#) proporciona esta capacidad. Con Key Vault, tiene una ubicación central donde se guardan los secretos que utilizan las aplicaciones.

Pueden ser credenciales en un flujo de conexión que utiliza la aplicación. La aplicación obtendría el flujo de conexión de Key Vault, en lugar de hacerlo desde el sistema de configuración. De esta manera, los administradores pueden controlar los secretos y los desarrolladores nunca se tienen que ocupar de ellos.

Key Vault también almacena certificados como Capa de conexión segura (SSL) que utiliza para asegurar el tráfico hacia y desde las aplicaciones HTTPS.



#### Introducción a [Azure Key Vault](#)

### Managed Service Identity

¿Cómo guarda las credenciales completas de su código? Puede comenzar usando Azure Key Vault, pero, ¿dónde almacena las credenciales para conectarse a Key Vault? Azure Managed Service Identity proporciona una solución.

Puede usar Managed Service Identity desde una gran cantidad de servicios en Azure, incluido Azure App Service. Simplemente habilite Managed Service Identity con un botón para insertar credenciales en su aplicación durante el tiempo de ejecución y luego use esas credenciales para acceder a otros servicios como Azure Key Vault. Toda autenticación entre servicios se realiza en el nivel de infraestructura. Su aplicación no tiene que lidiar con esto y puede simplemente usar otros servicios.



#### Cómo usar [Azure Managed Service Identity \(vista previa pública\)](#) en [App Service](#) y [Azure Functions](#)

### Cifrado

**Cifrado de datos por defecto:** Sus datos están cifrados por defecto en Azure si los almacena en Azure SQL Database, Azure SQL Data Warehouse, Azure Database for MySQL, Azure Database for PostgreSQL, Azure Storage, Azure Cosmos DB o Azure Data Lake Store.

Todo este cifrado funciona automáticamente, y no necesita configurar nada cuando lo usa. Para ayudar a cumplir con sus requisitos de seguridad y cumplimiento, puede usar las siguientes características para cifrar los datos en reposo:

- [Azure Disk Encryption](#) cifra la infraestructura de Windows y Linux como servicio (IaaS) de arranque de máquina virtual y volúmenes de datos utilizando claves administradas por el cliente.

- [Azure Storage Service Encryption](#) cifra automáticamente los datos antes de guardarse en Azure Storage, y luego descifra automáticamente los datos cuando los recupera.
- [Azure Client-Side Encryption](#) es compatible con el cifrado de datos dentro de las aplicaciones del cliente antes de cargarlos en Azure Storage u otros puntos de conexión, y luego descifrar los datos al descargarlos al cliente.
- SQL [Transparent Data Encryption](#) (TDE) cifra archivos de datos de [SQL Server](#), [Azure SQL Database](#) y [Azure SQL Data Warehouse](#). Los datos y los archivos de registro se cifran usando algoritmos de cifrado estándar de la industria. Las páginas en una base de datos se cifran antes de que se escriban en el disco y se descifran cuando se leen.
- SQL [Always Encrypted](#) cifra los datos dentro de las aplicaciones del cliente antes de almacenarlos en Azure SQL. Permite la delegación de la administración de bases de datos locales a terceros y mantiene la separación entre quienes poseen y pueden ver los datos y quienes los administran, pero no deben acceder a ellos.
- [Azure Cosmos DB](#) no requiere que realice ninguna acción porque los datos del usuario almacenados en Cosmos DB en almacenamiento no volátil (unidades de estado sólido) están cifrados de forma predeterminada, y no hay controles que los activen o desactiven.

### Azure Security Center

[Azure Security Center](#) brinda una administración unificada de la seguridad y protección avanzada contra amenazas para cargas de trabajo de nube híbrida. Ofrece controles de directivas centralizados que limitan la exposición a amenazas y encuentran y corrigen vulnerabilidades rápidamente. Además, Security Center admite la integración con soluciones de terceros y se puede personalizar con capacidades de automatización y programación. Puede usar Security Center para analizar el estado

de seguridad de sus recursos informáticos, redes virtuales, servicios de almacenamiento y datos, y aplicaciones. La evaluación continua lo ayuda a descubrir posibles problemas de seguridad, como sistemas con actualizaciones de seguridad faltantes o puertos de red expuestos. Una lista de hallazgos y recomendaciones priorizados puede desencadenar alertas u otras soluciones guiadas.

### Protección contra DDoS de Azure

Lo escuchó muchas veces en las noticias y no quiere que esto le ocurra a su empresa: un ataque de denegación de servicio distribuido (DDoS) se dirige a una aplicación. Estos tipos de ataques son cada vez más comunes y pueden sobrecargar la aplicación hasta el punto de que nadie más pueda usarla. [El servicio de protección DDoS de Azure](#) ofrece protección contra ataques de DDoS en un nivel gratuito (el nivel Básico) y un nivel de pago (el nivel Estándar).

El nivel Básico se habilita automáticamente como parte de la plataforma de Azure para cada cliente. No tiene que hacer nada para habilitarlo. Este servicio protege sus aplicaciones contra los ataques de DDoS más comunes mediante monitoreo y mitigación en tiempo real, y proporciona las mismas defensas utilizadas por Microsoft Online Services.

El nivel Estándar proporciona capacidades de mitigación adicionales que se ajustan específicamente a los recursos de Azure Virtual Network. Es fácil de habilitar y no tiene que cambiar sus aplicaciones; todo se hace a nivel de red. Además, puede personalizar la protección de nivel Básico con políticas propias que se centren en sus casos de uso y aplicaciones específicos.



[Lea más sobre protección contra DDoS de Azure](#)

### Azure Application Gateway

[Azure Application Gateway](#) es un dispositivo virtual dedicado que proporciona un controlador de entrega de aplicaciones (ADC) como servicio. Ofrece diversas capacidades de equilibrio de carga de Nivel 7 para su aplicación y permite a los clientes

optimizar la productividad de la granja web al descargar la terminación SSL intensiva en CPU a la puerta de enlace de la aplicación. La puerta de enlace también proporciona otras capacidades de enrutamiento de Nivel 7, incluidas la distribución del tráfico entrante, la afinidad de sesión basada en cookies, el enrutamiento basado en ruta de URL y la capacidad de alojar múltiples sitios web detrás de una única puerta de enlace de aplicación.

### Firewall de aplicaciones web Azure

Necesita proteger la aplicación contra muchas amenazas, entre las que se incluyen las definidas en el Proyecto de seguridad de aplicaciones web abierto (OWASP), como la inyección SQL y el scripting multisitios (XSS). [El Firewall de aplicaciones web Azure](#) puede ayudarlo con eso. El Firewall de aplicaciones web, una característica del servicio [Azure Application Gateway](#), proporciona protección en tiempo real para la aplicación. Puede detectar un ataque malicioso, tal como se define en el [conjunto de reglas base OWASP](#) y bloquear el ataque para que no llegue a la aplicación. También informa sobre los ataques que se intentaron llevar a cabo o están en curso para que pueda ver las amenazas activas para la aplicación, lo que proporciona un nivel de seguridad adicional.

### Azure Network Watcher

[Azure Network Watcher](#) es un servicio regional que le permite monitorear y diagnosticar las condiciones en el nivel de red en, hacia y desde Azure. Sus numerosas herramientas de diagnóstico y visualización pueden ayudarlo a comprender y obtener una visión más profunda de su red en Azure. Algunos ejemplos son:

- [Topología](#): Proporciona una vista a nivel de red que muestra las diversas interconexiones y asociaciones entre recursos de red en un grupo de recursos.
- [Captura de paquetes variables](#): Captura datos de paquetes dentro y fuera de una máquina virtual (VM). Las opciones avanzadas de filtrado y los controles ajustados, como la capacidad de establecer limitaciones de tiempo y tamaño, brindan versatilidad. Los datos del paquete se

pueden almacenar en un almacén de blobs o en el disco local en formato .cap.

- [Verificación del flujo de IP](#): Comprueba si un paquete está permitido o denegado en función de los parámetros del paquete de 5 tuplas de información de flujo (IP de destino, IP de origen, puerto de destino, puerto de origen y protocolo). Si un grupo de seguridad deniega el paquete, se devuelve la regla y el grupo que denegó el paquete.

### Grupos de seguridad de red

Un [grupo de seguridad de red \(NSG\)](#) contiene una lista de normas de seguridad que autorizan o deniegan el tráfico de red a recursos conectados a Azure Virtual Networks (VNet). Los NSG se pueden asociar a subredes, VM individuales (clásicas) o interfaces de red individuales (NIC) conectadas a VM (administrador de recursos). Cuando se asocia un NSG a una subred, las reglas se aplican a todos los recursos conectados a la subred. Puede restringir aún más el tráfico al asociar también un NSG a una VM o NIC.

### Zonas privadas de DNS de Azure

El Sistema de nombres de dominio (DNS) es responsable de convertir (o resolver) un nombre de servicio a su dirección IP. Azure DNS es un servicio de hospedaje para dominios de DNS que proporciona una resolución de nombres utilizando la infraestructura de Azure. Además de los dominios de DNS de Internet, el DNS de Azure ahora es compatible con dominios de DNS privados como una característica de vista previa: las zonas privadas de DNS de Azure. Los beneficios de seguridad de las zonas privadas de DNS incluyen la capacidad de crear una infraestructura de DNS dividida. Esto le permite crear zonas de DNS privadas y públicas con los mismos nombres, sin exponer los nombres internos. Además, el uso de la característica Zonas privadas de DNS de Azure elimina la necesidad de introducir soluciones de DNS personalizadas, lo que podría aumentar la superficie de ataque general debido a la actualización independiente y los requisitos de administración.

→ [Lea sobre Zonas privadas de DNS](#)

### Redes virtuales privadas entre instalaciones

Azure es compatible con dos tipos de conexiones de red privada virtual (VPN) entre instalaciones: VPN de punto a sitio (P2S) y VPN de sitio a sitio (S2S). Una conexión VPN de punto a sitio le permite crear una conexión segura a su red virtual desde un equipo de cliente individual. Una conexión P2S se establece al iniciarla desde el equipo de cliente. Esto es útil para los teletrabajadores que desean conectarse a Azure VNets desde una ubicación remota. VPN de P2S también es útil cuando solo tiene unos pocos clientes que necesitan conectarse a una VNet. Por el contrario, una conexión VPN de sitio a sitio se usa para conectar su red local a una red virtual de Azure a través de un túnel VPN IPSec/IKE (IKEv1 o IKEv2). Este tipo de conexión requiere un dispositivo VPN ubicado localmente que tenga una dirección IP pública externa.

→ [Lea sobre las redes privadas virtuales de punto a sitio y de sitio a sitio](#)

### Azure ExpressRoute

[Azure ExpressRoute](#) le permite ampliar sus redes locales a la nube de Microsoft sobre una conexión privada segura que facilita un proveedor de conectividad sin atravesar la Internet. Con ExpressRoute, puede establecer conexiones con los servicios en la nube de Microsoft, como Azure, Office 365 y Dynamics 365.

### Equilibradores de carga

Puede usar equilibradores de carga para aumentar la disponibilidad de las aplicaciones. Azure es compatible con equilibradores de carga externos e internos, que se pueden usar en una configuración pública o interna. Además, puede configurar los equilibradores de carga para que sean compatibles con puertos de alta disponibilidad (HA) donde una regla de puertos de HA es una variante de una regla de equilibrio de carga configurada en un estándar de equilibradores de carga internos. Puede proporcionar una única regla para equilibrar la carga de todos los flujos de TCP y UDP que llegan a todos los puertos de un equilibrador de carga interno.





[Lea sobre los equilibradores de carga y las reglas de puertos de HA](#)

## Registro y monitoreo

**Azure Log Analytics:** [Azure Log Analytics](#) le ayuda a recopilar y analizar los datos generados por los recursos en sus entornos locales y en la nube. Proporciona información en tiempo real a través de los paneles personalizados y los paneles de búsqueda integrados para que pueda analizar millones de registros de todas las cargas de trabajo y los servidores con independencia de su ubicación física.

**Azure Monitor:** [Azure Monitor](#) ofrece monitoreo básico de servicios de Azure, al recopilar métricas, registros de actividades y registros de diagnósticos. Por ejemplo, el registro de actividades le indicará cuándo se crearon o modificaron recursos. Hay métricas disponibles que ofrecen estadísticas sobre el rendimiento de diferentes recursos, incluido el sistema operativo asociado con una máquina virtual. Puede ver estos datos con alguno de los exploradores del portal de Azure y enviarlos a Log Analytics para buscar tendencias y analizarlos en detalle, o bien puede crear reglas de alertas para que se lo notifique ante problemas críticos.

## Registros de flujo de Azure Network Security

**Group:** Una característica de Network Watcher, los registros de [flujo de Azure NSG](#) le permiten ver información sobre el ingreso y la salida del tráfico de IP a través de Network Security Group. Los registros de flujo se pueden analizar para obtener información sobre el tráfico de red y los problemas de seguridad y rendimiento relacionados con el tráfico. Mientras que los registros de flujo apuntan a NSG, no se muestran de la misma manera que otros registros y se almacenan solo dentro de una cuenta de almacenamiento.

**Application Insights:** [Application Insights](#) es un servicio de administración de rendimiento de aplicaciones (APM) extensible para desarrolladores web que trabajan en múltiples plataformas. Incluye herramientas de análisis potentes que lo ayudan a diagnosticar problemas y a entender lo que los usuarios hacen con su aplicación. Funciona para

aplicaciones en una amplia variedad de plataformas hospedadas en entornos locales o en la nube, que incluyen .NET, Node.js y J2EE. Application Insights se integra con su proceso DevOps y tiene puntos de conexión con una gran variedad de herramientas de desarrollo. Puede monitorear y analizar la telemetría desde aplicaciones móviles al integrarse con Visual Studio App Center y HockeyApp.

**Azure Blueprint:** el plan de seguridad y cumplimiento de Azure para los datos de salud de HIPAA/HITRUST y la IA proporciona herramientas y orientación para ayudar a implementar un entorno de PaaS que cumpla con la Ley de transferencia y responsabilidad de seguros de salud (HIPAA) y Health Information Trust Alliance (HITRUST). Esta oferta de PaaS es compatible con la ingesta, el almacenamiento, el análisis y la interacción con registros médicos personales y no personales en un entorno de nube seguro y de varios niveles implementado como una solución completa. El modelo muestra una arquitectura de referencia común que podría aplicarse a casos de uso más allá de la atención médica, y está diseñada para simplificar la adopción de Azure.



[Lea sobre el plan de seguridad y cumplimiento de Azure para los datos de salud de HIPAA/HITRUST y la IA](#)

## Documentación técnica y de arquitectura de seguridad de Azure

Azure mantiene una gran biblioteca de documentación técnica de seguridad que complementa la información de seguridad incluida con los servicios individuales. Los documentos técnicos, los documentos de procedimientos recomendados y las listas de verificación se incluyen en el [sitio de Información de seguridad de Azure](#). También se abarcan los principales temas de seguridad de la nube pública en diversas áreas, que incluyen seguridad de red, seguridad de almacenamiento, seguridad informática, administración de identidad y acceso, registro/ auditoría, protección de carga de trabajo en la nube, seguridad de PaaS y más.



# Añadir inteligencia a la aplicación

Resulta agradable poder hospedar la aplicación y los datos y que estos estén seguros, sean escalables y tengan alto rendimiento; sin embargo, ¿no sería fantástico si, además, pudiera agregarles características inteligentes y funcionalidad con un mínimo esfuerzo?

## Azure Search

La búsqueda es una característica común en la mayoría de las aplicaciones y, sin embargo, ha sido tradicionalmente una función difícil de implementar.

[Azure Search](#) ofrece gran parte del trabajo de “plomaría” que requieren las búsquedas. Lanza una instancia de Azure Search, crea un índice que lo ayuda a buscar y lo rellena con datos. Eso es todo. Esto significa, por ejemplo, que podría implementar fácilmente Azure Search para ayudar a los usuarios a buscar en su catálogo de productos mediante una aplicación de comercio electrónico.

Hay muchas opciones que puede utilizar para ajustar Azure Search y muchas grandiosas características que les facilitarán la búsqueda a los usuarios:

- Geobúsqueda, que les permite a los usuarios explorar datos basados en la proximidad que tiene un resultado de búsqueda con una localización física.
- Analizadores de lenguaje de Lucene, así como procesadores de lenguaje natural de Microsoft (NLP), disponibles en 56 idiomas para manejar lingüística específica de cada idioma con inteligencia, que incluye tiempos verbales, género, sustantivos plurales irregulares (por ejemplo, “mouse” versus “mice”), descomposición de palabras, división de palabras (para los idiomas que no tienen sílabas) y mucho más.
- Seguimiento y generación de informes que le avisarán qué se buscó y qué tan rápida y exitosa fue la búsqueda.
- Características de experiencia de usuario, como la clasificación y compaginación de los resultados de la búsqueda y el filtrado inteligente, y la opción de sugerencias de búsqueda.

- Búsqueda cognitiva, que es un enfoque donde prima la IA para la comprensión del contenido. La búsqueda cognitiva cuenta con la tecnología de Azure Search con Cognitive Services integrados. Extrae datos desde casi cualquier fuente y aplica un conjunto de habilidades cognitivas modulares que extraen conocimiento. Este conocimiento luego se organiza y almacena en un índice, lo que permite nuevas experiencias para explorar los datos mediante la búsqueda.

### Ejemplo de búsqueda cognitiva

*Las compañías de petróleo y gas tienen equipos de geólogos y otros especialistas que necesitan comprender los datos sísmicos y geológicos. A menudo tienen décadas de archivos en formato PDF con imágenes de muestras, junto con hojas de muestra llenas de notas de campo escritas a mano. Deben conectar lugares, personas (expertos en dominio) y eventos, y luego navegar por toda esta información para tomar decisiones clave. La búsqueda cognitiva usa Cognitive Services para analizar todos estos datos, incluidos los archivos PDF y las imágenes, para extraer información y correlacionarla, todo sin la necesidad de escribir un software de reconocimiento de imágenes u OCR complicado.*



[Cree su primer índice de Azure Search en el portal](#)

## Azure Cognitive Services

[Azure Cognitive Services](#) proporciona algoritmos de machine learning y datos como servicio. Microsoft ha creado los algoritmos de machine learning que dirigen Cognitive Services, por lo que no es necesario que usted los cree. Además, para la mayoría de los servicios, Microsoft ha proporcionado los datos para entrenar esos algoritmos. Sin embargo, para algunos servicios puede usar sus propios datos personalizados para entrenarlos.

**Tabla 4-1:** Azure Cognitive Services por categoría

Visión	Voz	Idioma	Conocimiento	Búsqueda
Computer Vision	Speech Services	Text Analytics	QnA Maker	Bing Web Search
Face	• Voz a texto	Translator Text	Custom Decision	Bing Visual Search
Video Indexer	• Texto a voz	Bing Spell Check		Bing Entity Search
Content Moderator	• Traducción de voz	Content Moderator		Bing News Search
Custom Vision	Speaker Recognition	Language Understanding		Bing Custom Search
				Bing Image Search
				Bing Autosuggest
				Bing Video Search

Cognitive Services proporciona una manera excepcionalmente sencilla de incorporar el machine learning y la IA en su aplicación, simplemente llamando a las API. Hay [muchas API](#) en las categorías de visión, habla, lenguaje, conocimiento y búsqueda. La Tabla 4-1 muestra cuáles API están disponibles en este momento, pero tenga en cuenta que la lista de API sigue creciendo.

Cada categoría contiene múltiples servicios que puede usar al llamar a una API. Algunas categorías contienen servicios personalizados, como Custom Vision Service, Language Understanding Service y Bing Custom Search Service. Estos servicios personalizados proporcionan algoritmos de machine learning preconfigurados, al igual que los otros servicios, y le permiten usar sus propios datos para capacitar al modelo.

Además de estos servicios, puede usar los servicios en Cognitive Services Labs. Los laboratorios contienen servicios experimentales que Microsoft está probando para ver si encajan bien con los casos de uso de los clientes. Uno de esos servicios experimentales es Project Gesture, que le permite detectar gestos (como el movimiento de una mano) y entrelazarlos con su experiencia de usuario.

A continuación, observaremos algunos Cognitive Services.

### Custom Vision

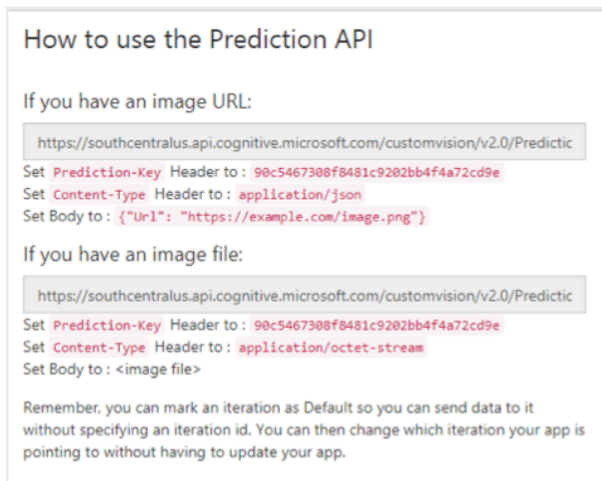
Con [Custom Vision](#) Service, puede detectar cosas en las imágenes en función de sus propios datos de capacitación. Custom Vision funciona de manera similar a otros Cognitive Service, ya que viene con un algoritmo de machine learning predefinido. No tiene que crearlo usted mismo; todo lo que hace es alimentar el servicio con sus datos.

Por ejemplo, es posible que desee crear un modelo que pueda detectar tipos de nubes en el cielo, como cúmulos y nubes estratos, a partir de una imagen. Para crear este modelo, carga imágenes de diferentes tipos de nubes en el portal de Custom Vision. Este portal está disponible cuando comienza a usar el servicio de Custom Vision. Cuando ha cargado las imágenes, les da etiquetas, que le dicen al servicio cómo capacitar el modelo. Entonces, coloca una etiqueta a una imagen, como "cúmulo" o "estrato".

Una vez que haya cargado suficientes imágenes, puede capacitar a su modelo. Cuantas más imágenes cargue con las etiquetas y más capacitación haga, más preciso será el modelo, lo que le proporcionará mejores resultados.



[Juegue con Cognitive Services](#)



**Figura 4-1:** Ejemplo del punto de conexión de la API Custom Vision

Una vez que tenga un modelo que funciona bien, puede comenzar a usarlo. Usa su modelo haciendo llamadas a la API de Custom Vision y proporcionándole nuevas imágenes. Cuando sube una nueva imagen, el servicio le dirá si la reconoce como uno de los tipos de nubes. La Figura 4-1 muestra un ejemplo de cómo se ve el extremo de la API.

Usar el servicio Custom Vision para detectar cosas basadas en su propio modelo es impresionante, ¡pero aún hay más! El modelo que crea cuando capacita el servicio Custom Vision con sus datos puede implementarse en el "perímetro inteligente". Esto significa que el modelo y la API pueden ejecutarse en otro lugar que no sea la nube, como en un servidor local en un contenedor Docker o en otro dispositivo, como su teléfono.

Esto ofrece una gran flexibilidad porque no necesita una conexión a Internet activa para usar las capacidades del servicio Custom Vision; también puede ejecutarlo localmente, lo que proporciona un gran rendimiento. Además, el modelo que ejecuta en el perímetro no es tan grande. Son aproximadamente decenas de megabytes, ya que solo implementa el modelo y la API, no los datos de capacitación (que se usan solo para crear el modelo).



[Cree su propio proyecto de Custom Vision](#)

## Video Indexer

El [servicio Video Indexer](#) proporciona información sobre los archivos de video y audio que carga en él. Este Cognitive Service también forma parte del [conjunto de Media Analytics](#) de [Azure Media Services](#). Este servicio también proporciona un algoritmo de machine learning predefinido, a la vez que proporciona los datos. En este caso, usted carga archivos de video o audio y Video Indexer los analiza.

Video Indexer tiene muchas capacidades impresionantes, algunas de ellas son:

- Crea una transcripción del texto en un video. Puede refinar la transcripción de forma manual y capacitar a Video Indexer para que reconozca los términos de la industria, como "DevOps".
- Registra rostros e identifica quién está en un video y en qué puntos. Video Indexer puede hacer lo mismo con el audio, donde reconoce qué altavoz dice algo y en qué momento.
- Reconoce el texto visual en un video, como el texto en una diapositiva, y lo convierte en parte de la transcripción.
- Realiza análisis de sentimiento, que le puede decir cuándo se dijo o se mostró algo positivo, negativo o neutro.
- Incluye [muchas más características](#).

Como muestra la amplitud de estas funcionalidades, Video Indexer combina muchos Cognitive Services, como [voz a texto](#) y [Speaker Recognition](#). Acumulativamente, estos servicios brindan capacidades potentes que hacen que el contenido sea más fácil de encontrar, más accesible y más valioso.

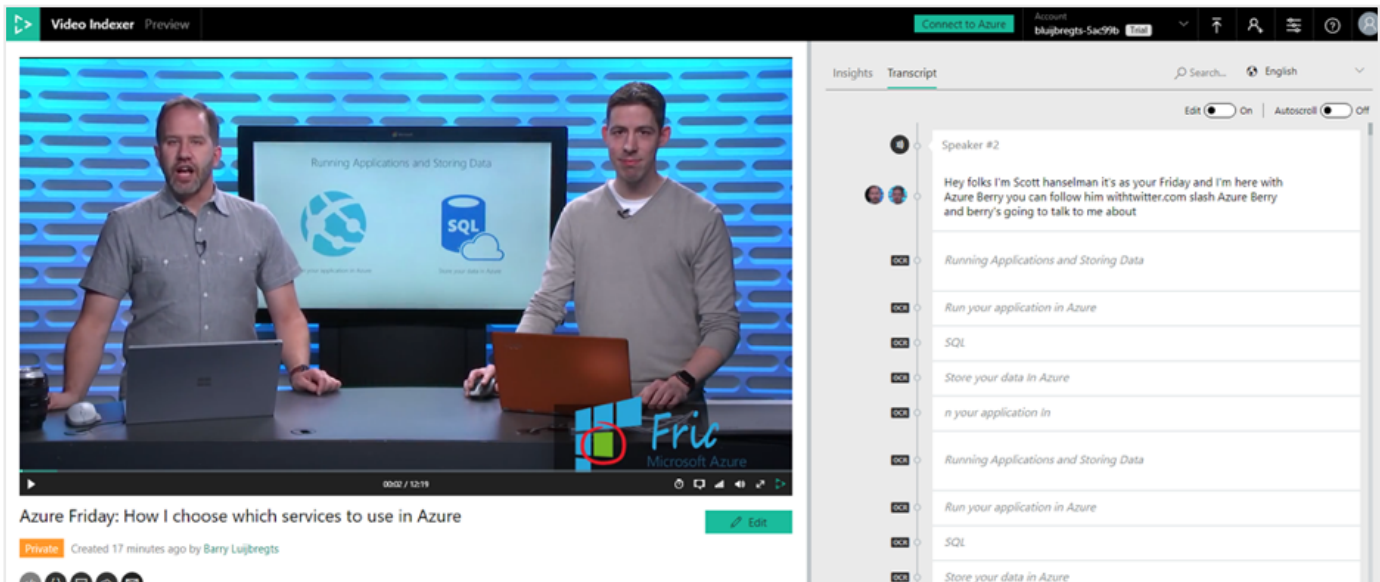


Figura 4-2: Ejemplo de un video cargado a Video Indexer

Puede cargar archivos multimedia en Video Indexer mediante el portal o la API de Video Indexer. La Figura 4-2 muestra los resultados de un [video de Azure Friday](#) que se cargó en el servicio Video Indexer.

Como se muestra en la figura, Video Indexer creó una transcripción del audio en el video. Puede editar la transcripción e incluso traducirla a otros idiomas. También puede ver que Video Indexer reconoció el texto en la diapositiva tras los altavoces y lo marcó como "OCR". Puede saltar a ese texto simplemente haciendo clic sobre él; Video Indexer ofrece esta funcionalidad para aplicaciones individuales al integrar el [widget de Cognitive Insights](#).

➔ [Cargue su primer video a Video Indexer](#)

## QnA Maker

[QnA Maker Cognitive Service](#) ofrece una manera fácil de navegar por las preguntas frecuentes. La mayoría de las empresas ya tienen preguntas frecuentes en su sitio web o en una ubicación

similar. Esta información se puede proporcionar a QnA Maker, que analiza la información para los pares de preguntas y respuestas y la información.

QnA Maker tiene como resultado un servicio que puede integrar en su aplicación que proporciona información a los usuarios de forma conversacional. Además, puede usar QnA Maker para crear un bot usando [Azure Bot Service y Language Understanding Service](#). Es posible crear un bot de QnA sin hacer ninguna codificación usando el portal de [QnA Maker](#). También puede usar la [API de QnA Maker](#) para crear servicios de QnA mediante programación.

Finalmente, solo [paga por el hosting de QnA Maker](#), no por la cantidad de veces que los usuarios realizan consultas en el servicio resultante.

➔ [Cree su propio servicio de base de conocimientos con QnA Maker](#)

## Bing Autosuggest

El servicio de Bing Autosuggest proporciona sugerencias de búsqueda mientras escribe. Esto le permite ofrecer una experiencia de búsqueda

familiar a sus usuarios, al igual que con Bing o Google, donde los resultados de búsqueda son automáticos o se completan.

Alimenta el texto de búsqueda, carácter por carácter, al servicio Bing Autosuggest, y

rápidamente devuelve las sugerencias de búsqueda en formato JSON. Por ejemplo, cuando ingresa el texto de consulta **¿Qué debo buscar?**, el servicio arroja el siguiente JSON:

```
{
  "_type": "Suggestions",
  "instrumentation": null,
  "queryContext": {
    "originalQuery": "what should I search for"
  },
  "suggestionGroups": [
    {
      "name": "Web",
      "searchSuggestions": [
        {
          "url": "https://www.bing.com/search?q=what+should+i+search+for&FORM=USBAPI",
          "urlPingSuffix": null,
          "displayText": "what should i search for",
          "query": "what should i search for",
          "searchKind": "WebSearch"
        },
        {
          "url": "https://www.bing.com/search?q=what+should+i+search+for+on+bing&FORM=USBAPI",
          "urlPingSuffix": null,
          "displayText": "what should i search for on bing",
          "query": "what should i search for on bing",
          "searchKind": "WebSearch"
        },
        {
          "url": "https://www.bing.com/search?q=what+should+i+search+for+on+the+internet&FORM=USBAPI",
          "urlPingSuffix": null,
          "displayText": "what should i search for on the internet",
          "query": "what should i search for on the internet",
          "searchKind": "WebSearch"
        },
        {
          "url": "https://www.bing.com/search?q=what+should+i+search+for+today&FORM=USBAPI",
          "urlPingSuffix": null,
          "displayText": "what should i search for today",
          "query": "what should i search for today",
          "searchKind": "WebSearch"
        },
        {
          "url": "https://www.bing.com/search?q=what+should+i+search+for+in+dna+raw+data&FORM=USBAPI",
          "urlPingSuffix": null,
          "displayText": "what should i search for in dna raw data",
          "query": "what should i search for in dna raw data",
          "searchKind": "WebSearch"
        }
      ]
    }
  ]
}
```

Este contiene todas las sugerencias y, en la parte superior de los resultados, contiene la consulta de búsqueda original.

➔ [Obtenga una clave de API y pruebe Bing Autosuggest gratis por 7 días](#)

## Azure Bot Service

[Azure Bot Service](#) lo ayuda a crear un bot, es decir, un software que puede interactuar con los usuarios de forma automática y autónoma.

Crear un bot no es tarea sencilla. Es necesario registrar el contexto de su interacción con el usuario y debe estar preparado para responder a una multitud de parámetros de interacción posibles. Bot Service lo ayuda con esta tarea de las siguientes maneras:

- Lo ayuda a mantener un seguimiento del contexto de interacción y proporciona plantillas para comenzar desde el marco de bot.
- Tiene integraciones estrechas con Cognitive Services y eso hace que su bot sea más "inteligente".
- Ayuda a integración con servicios como Facebook, Slack, Microsoft Teams, Telegram y más.
- Ofrece todos los beneficios de un servicio gestionado en Azure: escala masiva, entrega continua incorporada (CD) y pago por uso.

Un ejemplo de un bot que puede construir Bot Service es uno que ofrece a los usuarios respuestas a sus preguntas más frecuentes, las que puede usar junto con [QnA Maker Cognitive Service](#). La interfaz del bot puede ser un cuadro de chat que esté en su sitio de Internet.

➔ [Introducción al bot de chat con Azure Bot Service](#)

## Azure Machine Learning Studio

Puede agregar inteligencia a su aplicación con los servicios de Azure, como Cognitive Services. Estos se basan en algoritmos de machine learning que Microsoft creó para que los use como servicio. Sin embargo, existen otras formas de usar machine learning en sus aplicaciones. Antes de esa discusión, comencemos por hablar de lo que es y no es machine learning.

### ¿Qué es machine learning?

Machine learning a menudo se confunde con la inteligencia artificial (AI), pero no son lo mismo. IA implica máquinas que pueden realizar tareas que son características de la inteligencia humana. La IA también es algo que se puede implementar mediante el uso de machine learning, además de otras técnicas.

Machine learning en sí es un campo de la informática que da a los equipos la capacidad de aprender sin estar programados explícitamente. Machine learning se puede lograr utilizando una o varias tecnologías de algoritmo, como redes neuronales, aprendizaje profundo, redes bayesianas, etc.

Entonces, ¿qué implica machine learning?

La Figura 4-3 muestra el flujo de trabajo básico para el uso de machine learning.

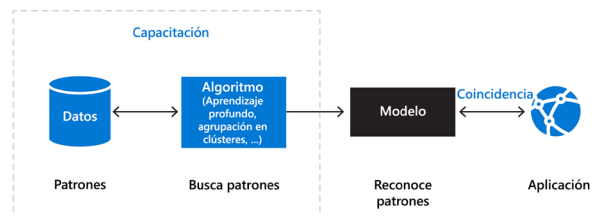


Figura 4-3: El proceso de machine learning

El proceso de machine learning funciona de la siguiente manera:



- Los datos proporcionados por usted u otra persona (como Microsoft) contienen patrones. Es posible que conozca algunos de los patrones, como los hábitos de pedidos de los usuarios. También es probable que haya muchos patrones en los datos que usted desconoce.
- El algoritmo de machine learning es la pieza inteligente de software que puede encontrar patrones en los datos. Este algoritmo puede ser algo que usted mismo crea utilizando técnicas como aprendizaje profundo o aprendizaje supervisado: la búsqueda de patrones en datos usando un algoritmo de machine learning también se denomina “capacitación de un modelo de machine learning”.
- La capacitación tiene como resultado un modelo de machine learning. Este contiene los aprendizajes del algoritmo de machine learning.
- Las aplicaciones ahora pueden usar el modelo al proporcionarle nuevos datos y trabajar con los resultados. Los nuevos datos se analizarán según los patrones que se encuentran en los datos. Por ejemplo, cuando capacita un modelo de machine learning para reconocer perros en imágenes, debe identificar a un perro en una imagen que nunca antes se ha visto.

La parte esencial de este proceso es que es iterativo. El modelo de machine learning se mejora constantemente al capacitarlo con nuevos datos y ajustar el algoritmo o al ayudarlo a identificar los resultados correctos de los incorrectos.

## Usar Azure Machine Learning Studio para crear modelos

Puede usar [Azure Machine Learning Studio](#) para crear sus propios modelos personalizados de machine learning y exponerlos a través de servicios web para que sus aplicaciones puedan usarlos.

Machine Learning Studio es el lugar principal en Azure donde puede crear proyectos de machine learning, combinar conjuntos de datos, crear blocs de notas y exponer modelos con servicios web. El

estudio en sí es un portal que puede usar desde su navegador web.

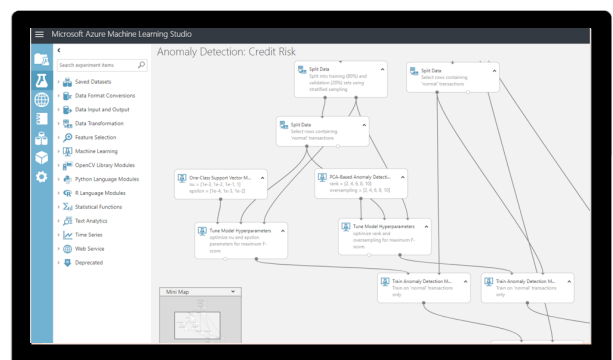
En el estudio, puede comenzar desde cero o con uno de los muchos experimentos que se encuentran en la [galería](#), incluido uno para predecir la duración de las hospitalizaciones y otro para detectar anomalías en flujos de datos en tiempo real. Puede utilizar estos experimentos como base para un modelo de machine learning o para aprender a resolver estos casos.

Un experimento de machine learning en Machine Learning Studio consta de varios pasos que manipulan datos y ejecutan algoritmos de machine learning en él. Puede crear estos pasos desde cero, incluido el algoritmo, o puede usar pasos predefinidos que están disponibles en el estudio.

La Figura 4-4 muestra un experimento en Machine Learning Studio. Puede ver el flujo de trabajo que se ejecutará para capacitar un modelo, así como las categorías de pasos predefinidos que se pueden usar en el flujo de trabajo.

Cuando ha creado experimentos y los ha usado en sus datos para crear un modelo de machine learning, puede publicarlos como servicios web. Cuando sus aplicaciones utilizan los servicios web, pueden enviar datos al modelo y recibir las predicciones de este.

➔ **Regístrese para usar [Azure Machine Learning Studio](#)**



**Figura 4-4:** Flujo de trabajo en un experimento de machine learning en Azure Machine Learning Studio





depurar algoritmos de machine learning y capacitar modelos de machine learning. Desde Visual Studio, puede crear trabajos de capacitación que pueden escalar a muchas VM en Azure. También puede monitorear el rendimiento de la capacitación y luego generar un servicio web para usar el modelo de machine learning en sus aplicaciones. Puede hacer todo esto sin salir de Visual Studio.



[Descargue la extensión Visual Studio Tools para IA.](#)

### Kit de herramientas de IA para Azure IoT Edge

El uso local de modelos de machine learning en dispositivos como su teléfono (también llamado “en el perímetro”) ofrece una poderosa ventaja. Le permite usar la capacidad de procesamiento local del dispositivo, sin depender de una conexión a Internet ni incurrir en la latencia de una llamada de servicio web para obtener sus resultados.

Tenga en cuenta que en la sección anterior sobre [Cognitive Services](#), observamos que el [servicio Custom Vision](#) ya es compatible con la ejecución en el perímetro. Más servicios se podrán ejecutar en el perímetro en el futuro.

Para ejecutar modelos de machine learning en el perímetro, necesita herramientas que le ayuden a implementar los modelos y servicios web. El Kit de herramientas de IA para Azure IoT Edge ayuda con esta herramienta al permitirle empaquetar modelos de machine learning en contenedores Docker compatibles con Azure IoT Edge y exponer esos modelos como API REST.

El Kit de herramientas de IA para Azure IoT Edge contiene ejemplos para comenzar, es completamente open source y está [disponible en GitHub](#).

## Uso de eventos y mensajes en la aplicación

Las aplicaciones modernas, globalmente distribuidas,

a menudo deben lidiar con grandes cantidades de mensajes y tienen que estar diseñadas teniendo en cuenta la desacoplación y el escalado. Azure proporciona varios servicios que lo ayudan con la ingesta de eventos y los patrones de mensajería y análisis. Estos servicios también son fundamentales para crear aplicaciones inteligentes que aprovechen IA.

### Azure Service Bus

La base de la mensajería en Azure es [Azure Service Bus](#). El servicio Service Bus abarca una colección de servicios que puede utilizar para patrones de mensajería. Los más importantes entre ellos son las colas y los temas de Azure Services Bus.



[Introducción a Colas y Temas de Azure Service Bus](#)

### Colas de Azure Service Bus

*Las colas de Azure Service Bus se utilizan para separar sistemas entre sí. P. ej.: una aplicación web recibe pedidos de los usuarios y necesita invocar un servicio web para procesar el pedido. Al servicio web le llevará mucho tiempo, quizás hasta cinco minutos, procesarlo por completo. Claro que es inaceptable que la aplicación web espere cinco minutos entre el pedido del usuario y la aparición del resultado. Una buena manera de solucionar esto es utilizar una cola para desvincular la aplicación web del servicio web.*

*La aplicación web recibe el pedido y lo escribe en un mensaje en una cola de Azure Service Bus. Luego, puede informarle al usuario que el pedido se está procesando. El servicio web toma los mensajes de la cola, uno por uno, y los procesa. Cuando el servicio web termina el procesamiento de un pedido, le envía una notificación por correo electrónico al usuario en la que le avisa que el elemento ya se ha pedido.*

*Al desacoplar los sistemas, la aplicación web puede funcionar a una velocidad diferente del servicio web y ambos pueden escalar individualmente en relación con las necesidades de las aplicaciones.*

*La cola de Services Bus es un mecanismo simple. Múltiples aplicaciones pueden colocar mensajes en la cola, pero un mensaje de la cola puede procesarse por una única aplicación a la vez. Tiene algunas funciones inteligentes para trabajar con mensajes en la cola, como la detección de duplicados y una subcola de letra muerta a la que los mensajes se mueven cuando no se procesan correctamente.*

Temas de Azure Service Bus

Al igual que con las colas de Service Bus, los temas de Azure Service Bus son una forma de desacoplación de la aplicación. Estas son las diferencias entre la cola y los temas:

- Con una cola, múltiples aplicaciones escriben mensajes en la cola, pero solo una aplicación a la vez puede procesar un mensaje.
- Con un tema, varias aplicaciones escriben mensajes en el tema y múltiples aplicaciones pueden procesar un mensaje al mismo tiempo.

Los temas de Service Bus funcionan como una cola y múltiples aplicaciones pueden procesar el mismo mensaje. Las aplicaciones pueden crear una suscripción en el tema, que indica qué tipo de mensajes les interesa. Al igual que las colas, los temas tienen características como la detección de duplicados y una subcola de letra muerta a la que los mensajes se mueven cuando no se procesan correctamente.

Comparación de las colas de Service Bus y las colas de Azure Storage

Existen las colas de Service Bus, pero también existen las de Azure Storage. Básicamente hacen lo mismo, pero hay diferencias sutiles entre ellas. En la Tabla 4-2 analizamos ambas.

Azure Event Hubs

En nuestro mundo del software omnipresente, se generan datos en muchas aplicaciones, dispositivos y ubicaciones, todo el tiempo. Hoy en día, es cada vez más común para las empresas captar enormes cantidades de esos datos para analizarlos

o transformarlos y moverlos para su uso posterior. [Azure Event Hubs](#) puede ayudarlo

Tabla 4-2: Comparación de características entre las colas de Service Bus y las de Colas de Storage

Colas de Azure Service Bus	Colas de Azure Storage
Ciclo de vida de los mensajes: más de 7 días	Ciclo de vida de los mensajes: menos de 7 días
Orden garantizado (primero en entrar, primero en salir)	Tamaño de la cola: más de 80 GB
Detección de duplicados	Registros de transacciones
Tamaño de los mensajes: menor o igual a 1 MB	Tamaño de los mensajes: menor o igual a 64 KB

Event Hubs está diseñado para la ingesta masiva de datos. Puede lanzarle millones de mensajes por segundo y manejará los datos por usted sin ningún esfuerzo. Puede retener mensajes hasta por siete días o retenerlos indefinidamente al escribirlos en un depósito de datos mediante la función Event Hubs Capture.

Puede utilizar Event Hubs para filtrar los datos con consultas, a medida que llegan, y enviar los resultados a un depósito de datos como Azure Cosmos DB. Incluso puede reproducir mensajes si es necesario.

 [Introducción al envío de mensajes a Azure Event Hubs](#)

Azure IoT Hub

Al igual que Event Hubs, [Azure IoT Hub](#) está desarrollado para la ingesta masiva de datos. Está específicamente configurado con una orientación hacia el manejo de un enorme volumen de mensajes de datos de dispositivos con Internet de las cosas (IoT), como termostatos inteligentes o sensores en los autos. Tiene muchas de las mismas propiedades que Event Hubs, como la

capacidad de retener mensajes hasta por siete días y reproducirlos.

Lo que hace único a IoT Hub es que, además de recibir mensajes de los dispositivos, también puede enviarles mensajes. Tiene la capacidad de administrar por completo su infraestructura de IoT. Puede usar IoT Hub para registrar dispositivos e informar su estado, administrarlos al asegurarlos y reiniciarlos y enviar datos a los dispositivos.

### → [Conecte su dispositivo a su IoT Hub](#)

## Azure Event Grid

Azure Event Grid ofrece otro tipo de mensajería. Event Grid ofrece un servicio de publicación/suscripción totalmente gestionado, que conecta casi todos los servicios de Azure y puede conectarse con suscriptores y editores personalizados.

Esto es diferente de trabajar con las colas y los temas de Service Bus, porque para ellos es necesario sondear la cola o los temas para buscar nuevos mensajes. Event Grids les envía automáticamente los mensajes a los suscriptores, lo que lo convierte en un servicio de eventos en tiempo real, reactivo.

Los servicios dentro y fuera de Azure publican eventos (por ejemplo, cuando se agrega un nuevo blob o cuando un nuevo usuario se agrega a una suscripción de Azure). Azure Event Grid puede detectar estos eventos y hacer que estén disponibles para los controladores de eventos y los servicios que se suscribieron a los eventos (Figura 4-6). Los controladores de eventos pueden ser cosas como Azure Functions o Azure Logics Apps, que luego pueden actuar sobre los datos en el evento.

Otro aspecto importante de Event Grid es que funciona sin servidor. Esto significa que, al igual que Azure Logic Apps y Azure Functions, Event Grid escala automáticamente y no necesita que se implemente una instancia. Solo lo configura y utiliza. Además, paga solo cuando se utiliza, no por el simple hecho de tenerlo.

Este es un ejemplo de uso para Azure Event Grid: Desea que le notifiquen por correo electrónico cada vez que se añade o elimina un usuario de su lista de correo en MailChimp. Azure Event Grid se utiliza para activar una aplicación en Azure Logic Apps y está configurado para escuchar los cambios en la lista de correo de MailChimp. El siguiente paso es enviar un correo electrónico que contiene el nombre del usuario que se ha añadido o eliminado y la acción que se realizó (agregar o eliminar). Ahora, cuando un nuevo usuario se agrega a la lista

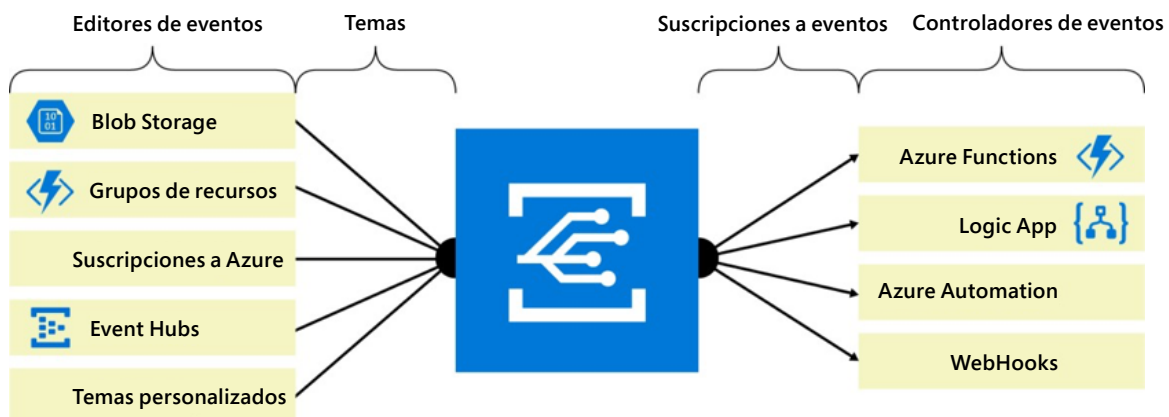


Figura 4-6: Descripción general de Azure Event Grid

de correo, Event Grid indica esto a Logic Apps, que envía un correo electrónico.

➔ [Monitoree los cambios de la máquina virtual con Azure Event Grid y Logic App](#)

## Azure SignalR Service

Puede usar [Azure SignalR Service](#) para agregar funcionalidad web en tiempo real a sus aplicaciones. El servicio se basa en ASP.NET Core SignalR y se ofrece como un servicio independiente y totalmente administrado en Azure.

SignalR puede actualizar las aplicaciones conectadas en tiempo real a través de HTTP, sin la necesidad de que las aplicaciones busquen actualizaciones o completen nuevas solicitudes HTTP. Esto le permite crear experiencias web integradas que actualizan la información sobre la

marcha. Por ejemplo, una aplicación de subasta podría usar SignalR para actualizar la última oferta tan pronto como ocurra, sin actualizar completamente la página ni buscar constantemente información.

Hospedar un servidor SignalR no es una tarea trivial, y puede ser difícil de escalar y protegerlo adecuadamente. Cuando utiliza Azure SignalR Service completamente administrado, la configuración es sencilla y la seguridad, la disponibilidad, el rendimiento y la escalabilidad se administran para usted.

➔ [Cree una sala de chat con SignalR](#)

## Qué usar y en qué momento

Azure proporciona múltiples opciones de mensajería y desacoplación de aplicaciones. ¿Cuál debe utilizar y cuándo? La Tabla 4-3 resume las diferencias para ayudarlo a elegir.

**Tabla 4-3:** Cómo determinar qué servicio Azure usar para mensajes o eventos

	SignalR Service	Event Grid	Event Hubs	IoT Hub	Temas	Colas de Service Bus	Colas de Storage
Ingesta de eventos		X	X	X			
Administración de dispositivos				X			
Envío de mensajes	X	X	X	X	X	X	X
Múltiples consumidores	X	X	X	X	X		
Múltiples remitentes	X	X	X	X	X	X	X
Uso para desacoplación			X	X	X	X	X
Uso para publicar/suscribir	X	X					
Tamaño máximo del mensaje	64 KB	64 KB	256 KB	256 KB	1 MB	1 MB	64 KB

# Trabajar con la IoT y entenderla

El mundo cada día está más y más conectado. El software se integra con el ámbito físico a través de dispositivos inteligentes que conforman la Internet de las Cosas (IoT). Los dispositivos nuevos y existentes están conectados entre sí y con Internet para obtener más capacidades y ayudar a los usuarios y las empresas a agregar más valor a cada aspecto de la vida.

Azure está ayudando a crear la Internet de las Cosas al proporcionar una plataforma de servicios para conectar dispositivos a la nube, obtener información de los datos y tomar medidas en la nube y en el perímetro. Explore los servicios de IoT que Azure tiene para ofrecer.

## Azure IoT Hub

En el corazón de Azure IoT se encuentra [Azure IoT Hub](#), que es una plataforma en la nube como servicio abierta y flexible, que conecta, monitorea y administra miles de millones de dispositivos de una manera segura y escalable. Ya hemos analizado Azure IoT Hub en la [sección sobre eventos y mensajes](#), pero hay más que contar sobre esto en el contexto de IoT.

Ya sabe que puede usar IoT Hub para ingerir grandes cantidades de mensajes que normalmente provienen de dispositivos IoT, como mensajes que contienen datos de sensores de temperatura. Además, IoT Hub es especial porque no solo recibe mensajes, sino que también los envía. Establece comunicación bidireccional con dispositivos, e incluso le permite ejecutar código en dispositivos. Esto es extremadamente poderoso porque le permite administrar dispositivos y controlarlos (como enviarles un mensaje para reiniciarse o ejecutar un script de arranque). Esto hace que IoT Hub sea el servicio central que permite una sólida solución de IoT en Azure.

Para ayudar a conectar dispositivos a IoT Hub, Azure proporciona [IoT Hub Device Provisioning Service](#). Este servicio permite el aprovisionamiento justo a tiempo y sin contacto en el IoT Hub correcto sin que se requiera intervención humana, lo que le permite proveer miles de millones de dispositivos de una manera segura y escalable. El servicio puede ayudarlo con muchos escenarios de aprovisionamiento de dispositivos, incluidos la conexión de dispositivos a IoT Hub y la ejecución de sus scripts de configuración inicial, los dispositivos de equilibrio de carga en múltiples concentradores y el reaprovisionamiento en función de un cambio en el dispositivo.

También puede conectar dispositivos que puedan ejecutar código en el "perímetro inteligente". Hablaremos más sobre el perímetro en la [sección](#)

[acerca de Azure IoT Edge](#). Solo sepa que IoT Hub se puede conectar a dispositivos que ejecutan el tiempo de ejecución y módulos de IoT Edge.

Una vez que un dispositivo está conectado a IoT Hub, el hub conoce el dispositivo y tiene un registro de su identidad. Esto permite que IoT Hub envíe mensajes y monitoree el dispositivo; también permite que el hub proteja el dispositivo y las comunicaciones entre el hub y el dispositivo. Es posible que se requiera que los dispositivos se autenticuen en IoT Hub utilizando varios protocolos de seguridad de procedimientos recomendados de la industria, como certificados X.509 y autenticación basada en tokens de SAS. Puede administrar la seguridad de cada dispositivo conectado y revocar privilegios si ya no desea que se conecte un dispositivo en particular.

Cuando los dispositivos envían mensajes a Azure IoT Hub, puede decidir si desea almacenarlos o enrutarlos a otro servicio para su análisis o acción. Por ejemplo, puede enrutar los mensajes entrantes para que sean manejados por una Azure Function que administra cada mensaje e inicia otro proceso. O bien, puede filtrar los mensajes con Azure Stream Analytics y solo almacenar los que sean relevantes para usted, como los cambios de temperatura en lugar de la misma temperatura cada segundo.

## Ejemplo de Azure IoT Hub

*Una empresa que proporciona información sobre el movimiento y el uso de camiones y personas quiere escalar su negocio. Anteriormente, la empresa había estado rastreando sus activos mediante el uso de código personalizado en una aplicación de teléfono nativa, la que llama a un servicio web personalizado, y mediante el sondeo de las llaves de GPS conectadas a los camiones. Mantener esta solución era un desafío porque era difícil proveer nuevos activos con nuevos dispositivos, y la empresa quería inscribir a un nuevo cliente que tiene más de 2000 activos.*



*Ahora, la empresa usa Azure IoT Hub para la administración y comunicación de dispositivos. Puede utilizar el servicio de aprovisionamiento de dispositivos de IoT Hub para incorporar los 2000 dispositivos nuevos y tal vez conectarlos a un hub de IoT específico para ese cliente. La aplicación telefónica ahora usa Node.js y [Azure IoT Device SDK](#) para interactuar con IoT Hub. Es importante destacar que la empresa en este momento tiene control sobre la seguridad de sus dispositivos y puede detectar el estado de ellos y restablecerlo, si es necesario. Además, la empresa enruta los datos de sus llaves de GPS a través de [Azure Stream Analytics](#), por lo que solo se guardan los datos de los cambios en el GPS. Esto reduce sustancialmente la carga de datos, ya que las llaves envían su ubicación cada segundo.*

*El uso de un servicio central (Azure IoT Hub) permitió a la empresa escalar y madurar su negocio proporcionando seguridad y administración de dispositivos de primera clase. También abrió nuevas oportunidades para poder hacer más de lo que alguna vez se creyó posible con los dispositivos de la empresa.*

## Azure IoT Central

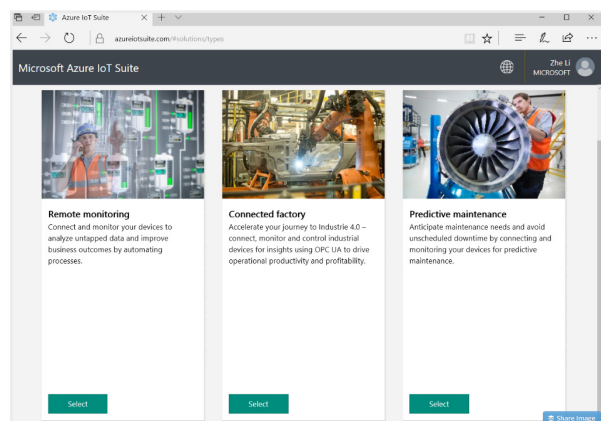
[Azure IoT Central](#) es una oferta de SaaS completamente administrada que le permite crear una solución de IoT con solo hacer clic. No es necesario realizar ninguna codificación o configuración en profundidad: IoT Central hace todo eso por usted. Usted navega por los asistentes y Azure IoT Central provee y configura los servicios que necesita (como Azure IoT Hub).

Terminará con las mismas capacidades que tendría si hubiera creado la solución desde cero, pero sin necesidad de tener años de experiencia en programación. Si desea más control sobre ciertas áreas de su solución, siempre puede profundizar y ajustar la solución a sus necesidades.

## Aceleradores de soluciones de Azure IoT

Un gran lugar para comenzar a desarrollar su solución de IoT es desde los [aceleradores de soluciones de Azure IoT](#), que son plantillas para escenarios comunes de IoT que puede personalizar según sus propias necesidades. Estas completas plantillas hacen todo, desde monitorear dispositivos, hasta protegerlos y proporcionar una interfaz de usuario. También lo ayudan a conectar sus dispositivos nuevos y existentes.

Aquí hay un ejemplo de una plantilla de soluciones disponible:



## Conecte y monitoree sus dispositivos con la supervisión remota.

Obtenga una mejor visibilidad de sus dispositivos, activos y sensores, donde sea que se encuentren. Recopile y analice datos de dispositivos en tiempo real utilizando un acelerador de soluciones de monitoreo remoto preconfigurado que activa alertas y acciones automáticas, las que van desde diagnósticos remotos hasta solicitudes de mantenimiento.

Además de la supervisión remota, hay muchos otros aceleradores de soluciones de IoT, como *mejorar la eficiencia industrial con una fábrica conectada, aumentar la confiabilidad del equipo con mantenimiento predictivo y desarrollar y probar su solución de IoT con una simulación de dispositivo.*



## Azure IoT Edge

En las soluciones modernas de IoT, el procesamiento de datos puede producirse en la nube o en dispositivos. El procesamiento del dispositivo se denomina "informática perimetral". Utilizará la informática perimetral cuando no desee (o no pueda) basar su conexión en la nube, cuando desee mejorar el rendimiento de la aplicación al eliminar los viajes de ida y vuelta a la nube, o cuando no pueda comunicarse con la nube del dispositivo por motivos de seguridad o reglamentarios.

Para escenarios como estos puede usar [Azure IoT Edge](#). Azure IoT Edge se basa en IoT Hub, lo que le permite mover partes de su carga de trabajo al perímetro, lo que reduce el tiempo que los dispositivos dedican a enviar mensajes a la nube y permite reacciones más rápidas a los cambios de estado.

**Tabla 5-1:** Ejecutar servicios de Azure en el perímetro

Si desea	Use esto en Azure IoT Edge
Desarrollar e implementar modelos de IA	<a href="#">Machine Learning</a>
Personalizar modelos de visión informáticos para su caso de uso	<a href="#">Custom Vision Service</a>
Procesar datos de flujo en tiempo real	<a href="#">Stream Analytics</a>
Procesar eventos con código sin servidor	<a href="#">Functions</a>
Implementar una base de datos de SQL Server en el perímetro	<a href="#">Bases de datos de SQL Server</a>
Cumplir con los estándares de interoperabilidad de Industria 4.0	<a href="#">OPC Unified Architecture</a>
Desarrollar lógica personalizada	<a href="#">Módulo personalizado</a>

Azure IoT Edge Azure IoT Edge consta de tres componentes:

- **IoT Edge Modules** son contenedores que ejecutan servicios de Azure, servicios de terceros o su propio código. Se implementan en dispositivos IoT Edge y se ejecutan localmente en esos dispositivos.
- El **tiempo de ejecución de IoT Edge** se ejecuta en cada dispositivo IoT Edge y administra los módulos implementados en cada dispositivo.
- **IoT Hub** expone interfaces específicas para monitorear y administrar de forma remota los dispositivos IoT Edge disponibles a través del portal de Azure, la CLI de Azure o los SDK.

Estos tres componentes trabajan juntos en el dispositivo y en la nube para ejecutar sus cargas de trabajo en el perímetro inteligente.

Actualmente, puede ejecutar muchos servicios de Azure en el perímetro para ayudar con ciertos escenarios, y la lista de servicios disponibles sigue creciendo. En la Tabla 5-1 se enumeran solo algunos de ellos.

Una vez que empiece a utilizar el perímetro, podrá crear soluciones increíblemente rápidas que ejecuten algoritmos de machine learning localmente y brinden retroalimentación instantánea con relación a sus hallazgos.

## Más información acerca de Azure IoT

Los servicios de Azure IoT son fáciles de usar y existe mucha documentación sobre ellos. Además, hay varios otros recursos pueden guiar su proceso de aprendizaje de Azure IoT. Écheles un vistazo:

- [Azure IoT School](#): esta academia en línea gratuita brinda capacitación integral para Azure IoT. Puede tomar diversos cursos, de nivel principiante o más avanzado.

- [Desarrollo de soluciones de IoT con Azure](#): Una guía para desarrolladores: esta experiencia de aprendizaje guiada en línea lo lleva a conocer los principales conceptos de Azure IoT a su propio ritmo.
- [Página de desarrollo de aplicaciones de Azure IoT](#): este recurso ofrece una descripción general de los servicios de Azure IoT y un ejemplo de cómo se pueden usar.
- [Aceleradores de soluciones de Azure IoT](#): puede usar estas plantillas para comenzar fácilmente con los servicios de Azure IoT.
- [Azure IoT Hub](#)
- [Azure IoT Edge](#)
- [Videos técnicos de Azure IoT en Channel9](#)

### Qué usar y en qué momento

Ahora que ha visto los servicios de Azure IoT disponibles, ¿cómo se puede saber qué servicio usar para su escenario? En la Tabla 5-2 se muestran las opciones de IoT en Azure y cuándo se debe usar cada una de ellas.

**Tabla 5-2:** ¿Qué Azure IoT Service usar y en qué momento?

	Azure IoT Hub	Azure IoT Central	Aceleradores de soluciones de Azure IoT	Azure IoT Edge
Crear una solución de IoT con mucho control mediante codificación personalizada	X			
Crear una solución de IoT sin preocuparse por el código y la administración del servicio de Azure		X		
Crear una solución de IoT para un escenario común con configuración y codificación mínimos			X	
Ejecutar cargas de trabajo de IA localmente en dispositivos de IoT	X*			X

\* Necesita Azure IoT Hub para administrar las implementaciones en el perímetro y los dispositivos Azure IoT.

# Dónde y cómo implementar los servicios de Azure

Los servicios de Azure en los que desee trabajar determinan sus opciones de implementación y viceversa. Por lo tanto, es importante entender las opciones de implementación que tiene en Azure y cuáles son sus ramificaciones.

## La mentalidad de entrega continua

La entrega rápida y la reiteración rápida son cruciales para crear un software excelente. Por lo tanto, su nuevo código debería fusionarse con el código del equipo y eso debería implementarse y probarse lo más rápido posible y con la mayor frecuencia posible, para ver si todo funciona y si lo que creó es lo que necesita el usuario.

Muchos de los servicios de Azure que hemos analizado hasta ahora en este manual son capaces de recibir código automáticamente, a menudo a través de un proceso de [entrega continua \(CD\) que se configura dentro del servicio](#).

Además de las funcionalidades nativas de los servicios Azure, puede utilizar Visual Studio Team Services para construir, probar e implementar la aplicación. Puede crear fácilmente [nuevos procesos de implementación de Visual Studio Team Services](#) y también hacer cosas como automatizar pruebas de carga y cambiar las ranuras de implementación en producción.



### Introducción a [Visual Studio Team Services](#)

Si ya usa herramientas de integración continua como [Jenkins](#), puede llevar fácilmente sus estructuras y procesos existentes a Azure y aprovechar los complementos de agente dinámico para reducir los requisitos y costos de infraestructura.

### Infraestructura como código

La infraestructura como código (IaC) captura definiciones de entorno como código declarativo, como JSON o YAML, para el aprovisionamiento y la configuración automatizados. Todos los servicios de Azure presentados en esta guía se basan en Azure Resource Manager, que puede usar para

documentar su entorno como IaC, gracias a las [plantillas de Resource Manager](#). Estas plantillas son archivos JSON que describen lo que desea implementar y cuáles son los parámetros.

Puede crear plantillas de Resource Manager en Visual Studio y Visual Studio Code utilizando la plantilla del proyecto Azure Resource Group. También puede generar plantillas de Resource Manager en el portal de Azure haciendo clic en el botón de secuencia de comandos de automatización, que está disponible en la barra de menús de todos los recursos disponibles en el portal de Azure. Esto genera la plantilla de Resource Manager para el recurso dado e incluso genera código para crear el recurso utilizando la interfaz de línea de comandos (CLI) de Azure, Windows PowerShell, .NET y Ruby.

Después de tener una plantilla de Resource Manager, se puede implementar en Azure mediante PowerShell, la CLI de Azure o Visual Studio. O bien, puede automatizar su implementación en un proceso de entrega continua mediante Visual Studio Team Services.

Utiliza plantillas de Resource Manager para implementar aplicaciones para que se ejecuten sobre la plataforma Azure, en la nube pública o en el entorno local, en Azure Stack.

Un gran ejemplo de implementación de recursos en la nube que utiliza Resource Manager es el botón Implementar en Azure que se puede encontrar en muchos repositorios GitHub, tal como se ilustra en la Figura 6-1.

Además de utilizar [Azure Resource Manager](#) para la infraestructura como código, puede traer sus habilidades y herramientas existentes, como [Ansible](#), [Chef](#) y [Terraform](#), para aprovisionar y administrar la infraestructura de Azure directamente.



**Figura 6-1:** Implementación con un botón de un [proyecto de Azure en GitHub](#)

## Azure Service Fabric

Otra forma de ejecutar las aplicaciones es ejecutarlas en [Azure Service Fabric](#). Service Fabric es la base que Microsoft utiliza para ejecutar muchos recursos de Azure y para que tengan alta disponibilidad, con alto rendimiento y recuperación automática. Ahora usted puede utilizar Service Fabric para hospedar sus propios servicios.

Puede utilizar el SDK de Azure Service Fabric para crear aplicaciones para Service Fabric. También puede ejecutar cualquier archivo ejecutable en Service Fabric e incluso usarlo para hospedar contenedores.

Service Fabric logra que las aplicaciones tengan tanto rendimiento, confiabilidad y seguridad como muchos servicios de Azure, y puede utilizarse en cualquier lugar: puede implementar Service Fabric en Azure, en un entorno local, en su propia computadora e incluso en máquinas virtuales (VM) en otras nubes. Una vez implementada su aplicación en Azure Service Fabric, puede aprovechar los beneficios tales como equilibrio de carga, escalamiento automático, alta disponibilidad, autorreparación, replicación y conmutación por error, actualizaciones progresivas y reversión automática. Además, Azure Service Fabric se convirtió recientemente en open source, lo que le permite participar en el desarrollo y la dirección de la plataforma.

## Contenedores en Azure

“Contenedores” es una de esas palabras de moda de tecnología que circulan por las noticias. Pero es más que moda: es realmente muy útil para ejecutar las aplicaciones. Un contenedor es básicamente una

**Tabla 6-1:** Elegir qué servicio Azure usar para contenedores

	Azure Kubernetes Service	Azure Service Fabric	Azure Container Instances	Web App for Containers	Contenedores en Azure Batch
Para las implementaciones de producción de sistemas complejos (con un organizador de contenedores)	X	X			
Para ejecutar configuraciones simples (posiblemente sin organizador)			X	X	
Para cargas de trabajo de larga duración en los contenedores	X	X			X
Para cargas de trabajo de corta duración en los contenedores			X		X
Para organizar un sistema basado en contenedores	X	X			
Organizar con los organizadores de open source Kubernetes	X				
Organizar con organizador integrado		X			
Usar características de App Service como ranuras de implementación				X	

VM liviana que se inicia y se detiene mucho más rápido que una VM y, por lo tanto, es mucho más útil para el desarrollo, la prueba y la ejecución de aplicaciones en producción.

El mayor beneficio que ofrecen los contenedores es que un contenedor individual es siempre igual. Puede ejecutar un contenedor localmente cuando desarrolla la aplicación y utilizar la misma configuración del contenedor en la nube o en cualquier otro lugar. Todo su equipo utiliza la misma configuración exacta del contenedor, así que usted sabe que la infraestructura es la misma para todo el mundo y en la producción. Con los contenedores, la frase histórica que usan los desarrolladores ("funciona en mi máquina") ahora significa que también funcionará en la producción.

Hay muchas tecnologías para ejecutar contenedores, incluida [Docker](#). Azure puede ejecutar y administrar contenedores con [Azure Container Instances](#) y Azure Kubernetes Service, e incluso en Service Fabric y [Azure App Service en Linux](#). Además, puede ejecutar contenedores en Azure App Service Web App para Contenedores y en Azure Batch. La Tabla 6-1 muestra qué servicio puede elegir para diversos escenarios al utilizar contenedores.

**Nota:** Tenga en cuenta que cuando utiliza contenedores, utiliza un producto de infraestructura como servicio (IaaS) y que usted es responsable por el sistema operativo (SO), los parches, el equilibrio de carga y demás aspectos.

## Azure Stack

Si necesita que las aplicaciones y los datos permanezcan en un entorno local, pero igual quiere aprovechar la capacidad que Azure tiene para ofrecer, [Azure Stack](#) es el producto para usted. Azure Stack, producto único en la industria, es una extensión de Azure que se hospeda en su propio entorno. En esencia, es una versión compacta de Azure.

Azure Stack se utiliza de la misma manera que Azure, con la misma experiencia que el portal de Azure y las mismas API con las que puede utilizar el CLI de Azure, Windows PowerShell o su IDE favorito.

Puede ejecutar cosas como Azure App Service y máquinas virtuales de Azure en Azure Stack. Todo es exactamente igual que en la nube pública, solo que ahora lo ejecuta en un entorno local. Además, si decide migrar a la nube pública, solo tiene que insertar los servicios de Azure Stack en Azure.

## Ejemplo de Azure Stack

*Una empresa que ofrece vacaciones en un crucero de lujo ha desarrollado diversos software para ayudar con las tareas del crucero. Los ejemplos incluyen una aplicación de administración de cabina, una aplicación de administración de pasajeros, etc. De hecho, todo el crucero depende de estas aplicaciones.*

*En el pasado, las aplicaciones se ejecutaban en servidores que se transportaban a bordo de los cruceros. La empresa se vio obligada a hacerlo de esta manera porque los cruceros no tenían conexión a Internet durante todo el viaje.*

*La empresa descubrió que ejecutar sus aplicaciones en el entorno local era engorroso, ya que tenía que mantener las VM y los sistemas operativos, además de tener que lidiar con problemas de disponibilidad importantes. Ahora, la empresa ejecuta sus aplicaciones en Azure Stack, que se ejecuta localmente en los cruceros. Azure Stack provee los mismos servicios que Azure, por lo que la implementación y administración de las aplicaciones se hizo mucho más fácil. La empresa también usa App Service para ejecutar sus aplicaciones, lo que le permite concentrarse en las aplicaciones en lugar de hacerlo en el mantenimiento de VM y sistemas operativos. Finalmente, los usuarios disfrutaban de la mayor disponibilidad que es parte de Azure, y por lo tanto, parte de Azure Stack.*

**Tabla 6-2:** Comparar opciones de implementación de Azure

	Azure IoT Hub	Azure IoT Central
En entorno local	Azure Stack Contenedores Service Fabric	Azure Stack Service Fabric
Cloud público	Contenedores Service Fabric	Service Fabric Plantillas de Resource Manager

### ¿Dónde y cuándo implementar?

La tabla 6-2 resume las opciones de implementación para Azure. Tenga en cuenta que solo es posible usar todos los servicios de PaaS mencionados en esta guía cuando se usa la nube pública con plantillas de Resource Manager o en un entorno local con Azure Stack.

Si desea implementar servicios basados en IaaS (donde tiene control sobre el sistema operativo), considere estas opciones:

#### En un entorno local o en cualquier otro lugar

(como su PC local u otra nube), puede elegir entre:

- Azure Stack (donde implemente servicios como VM)
- Cualquiera de los servicios de contenedor de Azure (ya que los contenedores se pueden ejecutar en cualquier lugar)
- En Service Fabric (cuando usa contenedores)

#### En la nube pública de Azure, puede elegir entre:

- Contenedores (ya que los contenedores también se ejecutan en los servicios de contenedor de Azure)

- Service Fabric (cuando usa contenedores, ya que Service Fabric también se ejecuta en la nube pública)

Si desea implementar servicios basados en PaaS (donde tiene menos control, pero la plataforma hace el trabajo pesado), considere estas opciones:

#### En un entorno local o en cualquier otro lugar

(como su PC local u otra nube), puede elegir entre:

- Azure Stack (ya que puede implementar servicios de PaaS como App Service en Azure Stack)
- En Service Fabric (cuando usa el modelo de programación de Service Fabric)

#### En la nube pública de Azure, puede elegir entre:

- Service Fabric (cuando utiliza el modelo de programación de Service Fabric, ya que Service Fabric también se ejecuta en la nube pública)
- Cualquier servicio PaaS de Azure que usted crea como plantilla de Resource Manager

# Microsoft Azure en acción

Ahora que ya sabe qué es Azure y ha aprendido acerca de los servicios que ofrece, vamos a empezar a utilizarlo.



# 01 /

## Tutorial: Desarrollar una aplicación web y una base de datos en Azure

[su equipo Git \(v2 o superior\)](#), [.NET Core](#) y [Visual Studio Code](#). También usaremos una aplicación de muestra ASP.NET Core MVC que puede utilizar para administrar una lista de tareas pendientes.

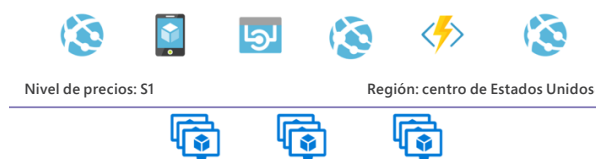
## Creación de aplicación web y base de datos mediante el portal de Azure

Para hospedar la aplicación de .NET Core, vamos a crear una nueva aplicación web en el portal de Azure.

1. En el portal de Azure, haga clic en el botón Crear un nuevo servicio.
2. Busque Web App. Se abre la hoja Web App. Haga clic en Crear para comenzar.
3. Se abre la hoja Crear de Web App.
  - a. Escriba un nombre para la Web App.
  - b. Cree un nuevo grupo de recursos al darle un nombre.
  - c. Deje la selección de sistema operativo como Windows.
  - d. Seleccione o cree un App Service Plan y, a continuación, haga clic en Crear.

**Web App se ejecuta con App Service Plan**  
*Los App Service, como Web App, se ejecutan con App Service Plans de Azure. Estos últimos son una abstracción de los recursos y funciones, como la CPU y la memoria, representados en niveles de precios. Los App Service Plan también están limitados a una región geográfica específica que usted elija. Puede, por ejemplo, ejecutar la aplicación Web App en un App Service Plan de precios nivel S1 (ver figura 7-1), que tiene un núcleo y 1,75 GB de RAM.*

Figura 7-1: Explicación de Azure App Service Plan



*Puede ejecutar todos los App Service que quiera en un App Service Plan, siempre y cuando comprenda que es necesario compartir los recursos entre todos los App Service.*

Para hospedar la base de datos, crearemos una Azure SQL Database. Esto funciona igual que una SQL Server Database local y ahora se ejecuta de forma completamente administrada en Azure.

1. En el portal de Azure, haga clic en el botón Crear un nuevo servicio.
2. Busque SQL Database. Haga clic en SQL Database en los resultados de búsqueda para abrir la hoja de SQL Database. Haga clic en Crear para comenzar.
3. Se abre la hoja Crear de SQL Database.
  - a. Escriba un nombre para la **base de datos**.
  - b. Seleccione el **grupo de recursos** que ha creado para la Web App.
  - c. Deje el **origen** como una **base de datos en blanco**.
  - d. Haga clic en **Servidor** para crear un nuevo Azure SQL Database Server:
    - i. Escriba un **nombre** para el servidor.
    - ii. Escriba el **Inicio de sesión del administrador del servidor**. Este es el nombre de usuario para el servidor.
    - iii. Escriba la **contraseña** que usará para iniciar sesión en el servidor.
    - iv. **Confirme** la contraseña.
    - v. Seleccione una **ubicación**. Elija la misma ubicación que seleccionó para App Service Plan.
    - vi. Haga clic en **Seleccionar** para enviar la nueva configuración del servidor.
  - e. Seleccione un **Nivel de precios**. Para fines de desarrollo y prueba, será suficiente con el nivel **Básico**.
  - f. Haga clic en **Crear**. Ahora se creará la base de datos.

4. Navegue hasta Azure SQL Database y haga clic en **Mostrar la cadena de conexión de la base de datos**.
5. **Copie** la cadena de conexión en un lugar seguro. La necesitará más adelante en este tutorial.

## Ejecutar la aplicación .NET Core localmente

Vamos a ejecutar la aplicación localmente, antes de ejecutarla en Azure. La aplicación se puede ejecutar localmente porque de manera predeterminada, utiliza una base de datos SQLite, que es un motor de base de datos SQL autónomo.

1. Abra una **símbolo del sistema** y vaya a un directorio que desee usar como su directorio de código fuente para este proyecto.
2. Ejecute los siguientes comandos para obtener el código fuente y navegar a la carpeta del proyecto:

```
git clone https://github.com/azure-samples/dotnetcore-sqlldb-tutorial
cd dotnetcore-sqlldb-tutorial
```

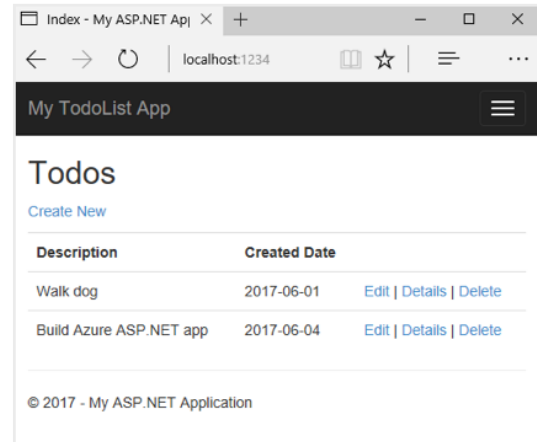
3. El proyecto usa Entity Framework Core para llenar su base de datos. Para asegurarse de que la base de datos esté actualizada y ejecutar la aplicación localmente, ejecute los siguientes comandos:

```
dotnet restore
dotnet ef database update
dotnet run
```

4. La aplicación ahora debería estar ejecutándose, y la URL de la aplicación debería estar en el

resultado en la ventana de comandos. Vaya a la URL (como `http://localhost:5000`) en un navegador. Esto cargará la aplicación, que se verá como la Figura 7-2. Ahora puede crear nuevos elementos para tareas seleccionando el enlace **crear nuevo**.

Figura 7-2: La aplicación de tareas en ejecución local



5. Cierre la aplicación al cerrar la ventana de comandos o al presionar **CTRL + C**.

## Conectar la aplicación web local a la base de datos que se ejecuta en Azure

Ahora tiene una aplicación en funcionamiento que se ejecuta localmente. Antes de implementarla en Azure, cambiaremos el código fuente para que se pueda conectar a Azure SQL Database.

1. En su repositorio de código fuente local, busque el archivo **Startup.cs** y localice el siguiente código:

```
services.AddDbContext<MyDatabaseContext>(
    options => options.UseSqlite(
        "Data Source=localdatabase.db"));
```

2. Reemplázelo con el siguiente código, que se puede conectar a Azure SQL Database:

```
// Use SQL Database if in Azure, otherwise,
// use SQLite

if(Environment.
GetEnvironmentVariable("ASPNETCORE_
ENVIRONMENT") == "Production")

services.
AddDbContext<MyDatabaseContext>(options =>

options.UseSqlServer(Configuration.
GetConnectionString("MyDbConnection"));
else

services.
AddDbContext<MyDatabaseContext>(options =>

options.UseSqlite("Data
Source=localdatabase.db"));

// Automatically perform database migration
services.BuildServiceProvider().
GetService<MyDatabaseContext>().Database.
Migrate();
```

Este código examina el entorno en el que se está ejecutando y cambia su conexión de base de datos en función de eso. Cuando se ejecuta en el entorno de producción (que es Azure, en nuestro caso), obtendrá la cadena de conexión para la base de datos desde la variable MyDbConnection, que es algo que configuraremos en Azure.

Además, el código ejecuta el método Database.Migrate (), que ejecuta las migraciones de Entity Framework Core que previamente ejecutamos manualmente. Con este método en su lugar, no tenemos que preocuparnos por eso.

3. Guarde sus cambios y ejecute los siguientes comandos para confirmarlos en su repositorio local de Git:

```
git add .
git commit -m "connect to SQLDB in Azure"
```

Ahora debemos configurar la variable de cadena de conexión en Azure.

4. En el portal de Azure, navegue a la **Web App** que creamos anteriormente.
5. Navegue a **Configuración de la aplicación**.
6. Cree una **nueva cadena de conexión**. El nombre debe ser **MyDbConnection**, y el valor debe ser la cadena de conexión a Azure SQL Database (incluido el nombre de usuario y la contraseña) que guardó anteriormente en el paso para crear la base de datos.
7. Haga clic en **Guardar**. La configuración de la aplicación en el portal se verá como en la Figura 7-3.

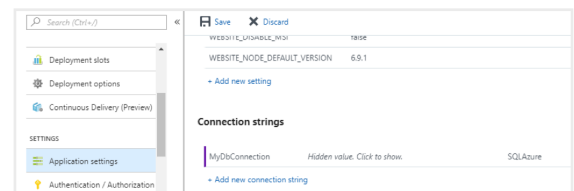



Figura 7-3: Configuración de la aplicación en el portal de Azure

Ahora, estamos listos para implementar la aplicación en Azure.

## Implementar la aplicación web en Azure

Vamos a utilizar Git para insertar la aplicación en Azure. Para hacerlo, debemos conectar nuestro repositorio local de Git a Azure. Para conectar Git local con Azure, debe tener un usuario de implementación configurado en el servidor (Azure Web App) para autenticar su implementación. Este usuario de implementación es al nivel de la cuenta y es diferente de su cuenta de suscripción de Azure. Es necesario configurar este usuario de implementación solamente una vez.

1. En el portal de Azure, navegue hasta Azure Cloud Shell. Puede realizar esto si hace clic en el botón que se encuentra en la barra superior y se muestra a continuación: 
2. Azure Cloud Shell le permite usar la CLI de Azure en la nube y se encarga de la autenticación por usted. Cuando Cloud Shell esté completamente cargado, ejecute el siguiente comando para crear el **usuario de implementación**. Reemplace los valores <username> y <password> en el comando con los que usted mismo cree. **Asegúrese de anotar el nombre de usuario y la contraseña.** Los necesitaremos después.

```
az webapp deployment user set --user-name
<username> --password <password>
```

3. El comando dará como resultado una salida JSON. Si recibe un error **'Conflict'. Details: 409**, cambie el nombre de usuario. Si recibe un error **'Bad Request'. Details: 400**, debe crear una contraseña más fuerte.

Ahora debemos insertar el código fuente de nuestro repositorio de Git local en Azure Web App.

4. Abra el **símbolo del sistema** en su máquina local.
5. Agregue Azure remoto a su repositorio de Git local. Puede hacerlo utilizando la URL de Git remota, que tiene este formato: `https://<username>@<app_name>.scm.azurewebsites.net/<app_name>.git`
  - a. Reemplace **<username>** con el nombre de usuario que hemos utilizado para crear el usuario de implementación.
  - b. Reemplace **<app\_name>** con el nombre de Azure Web App.

- c. Use la URL para ejecutar el siguiente comando:

```
git remote add azure
<deploymentLocalGitUrl>
```

6. Ahora que el objetivo remoto se ha agregado al repositorio de Git, puede insertar su código ejecutando el siguiente comando. Se le pedirá que ingrese las credenciales para poder enviar el código a Azure. Use el nombre de usuario y la contraseña del **usuario de implementación** que creó anteriormente.

```
git push azure master
```

7. La inserción del código fuente en Azure puede demorar unos minutos la primera vez que lo hace. Cuando termine, navegue a la URL de su Azure Web App, que se verá así: `http://<app_name>.azurewebsites.net`
8. Agregue algunos elementos de tareas en la aplicación para probar que su conexión a la base de datos está funcionando.

¡Listo! Ahora tiene una aplicación en funcionamiento que se ejecuta en Azure.

## 02 /

# Tutorial: Extender aplicaciones con Azure Logic Apps y Cognitive Services

Una poderosa función de nuestra aplicación es la capacidad de analizar el contenido de los elementos de tarea y luego crear una cita automáticamente en el calendario para las tareas que requieren atención en una fecha específica, si se detecta una. Por ejemplo, si un usuario crea un nuevo elemento de tarea con el texto "cena familiar el próximo viernes a las 19:00", la aplicación creará un ítem en el calendario para ese viernes específico a las 19:00 con el asunto "cena familiar".

Configuraremos esta característica mediante una [Azure Logic App](#) y el [Language Understanding Cognitive Service](#). Esto funcionará de la siguiente forma:

La aplicación .NET Core escribe el elemento de tarea en SQL Database.

La Logic App se activa por cada nueva fila que se crea en la base de datos.

La Logic App toma el texto del elemento de tarea y lo pasa al servicio de Language Understanding.

El servicio de Language Understanding analiza el texto y crea un elemento de calendario en su calendario de Office 365, si el elemento contiene una fecha y hora.

Lo divertido es que no tenemos que cambiar nuestra aplicación para agregar esta funcionalidad. Logic App y Cognitive Service son servicios adicionales que simplemente analizan los datos que ya están allí.

¡Manos a la obra!

## Creación del servicio Language Understanding

Primero, crearemos el servicio Language Understanding para que podamos usarlo en otro momento en nuestra Azure Logic App. El servicio Language Understanding es un Azure Cognitive Service que puede entender lo que significa el texto, en función de un modelo que proporcionamos. En este ejemplo, mantendremos el modelo simple, y no lo crearemos a fin de que esté preparado para cada variación que los usuarios puedan necesitar para una fecha en un elemento de tarea. Si lo desea, puede agregarlo usted mismo al modelo.

1. En el portal de Azure, haga clic en el botón **Crear un nuevo servicio**.
2. Busque **Language Understanding**. Haga clic en Language Understanding en los resultados de búsqueda para abrir la hoja de Language Understanding. Haga clic en **Crear** para comenzar.
3. Se abre la hoja Crear Language Understanding.
  - a. Escriba un **nombre**.
  - b. Seleccione un **nivel de precios** (para fines de este tutorial sirve cualquiera).
  - c. Cree un nuevo **grupo de recursos** llamado **datadetection**.
  - d. Haga clic en **Crear**.
4. Ahora se creará el servicio Language Understanding. Una vez creado, navegue hasta el servicio.
5. De forma predeterminada, el servicio se abrirá en la hoja de **Inicio rápido**. Aquí, seleccione **portal de Language Understanding** para navegar al portal.
  - a. Si es necesario, inicie sesión con el botón **Iniciar sesión** en la esquina superior derecha.
6. Haga clic en **Crear nueva aplicación**:
  - a. Escriba un **nombre**.
  - b. Haga clic en **Listo**.
7. Ahora estamos en el portal de Language Understanding. Aquí, podemos construir un modelo de lenguaje. Queremos poder entender la frase "cena familiar el próximo viernes a las 19:00". Para hacer eso, primero agregaremos algunas entidades, que son elementos que el servicio reconocerá en el texto. Haga clic en **Entidades**:
  - a. Haga clic en **Administrar entidades precompiladas**.
  - b. Seleccione **DatetimeV2** y **keyPhrase**.
  - c. Haga clic en **Listo**.
8. Ya tenemos dos entidades que reconocerán el texto por nosotros. Ahora haga clic en **Intenciones**.
9. Haga clic en **Crear nueva intención**:
  - a. Escriba un nombre, como "Añadir elemento de tarea al calendario", que es la intención que estamos intentando detectar en el texto.
  - b. Haga clic en **Listo**.
10. Ahora puede ingresar **Expresiones**. Estos son textos de muestra que representan la intención que estamos intentando detectar. Entonces, en nuestro caso, queremos que el texto "cena familiar el próximo viernes a las 19:00" represente la intención de agregar un elemento de tarea al calendario. Continúe, escriba eso.
11. Como ya hemos agregado dos entidades, el texto en el enunciado se analiza y reconoce como estas entidades. En este caso, se ve como en la Figura 7-4:

keyPhrase datetimeV2

**Figura 7-4:** Entidades reconocidas a partir del texto en un enunciado

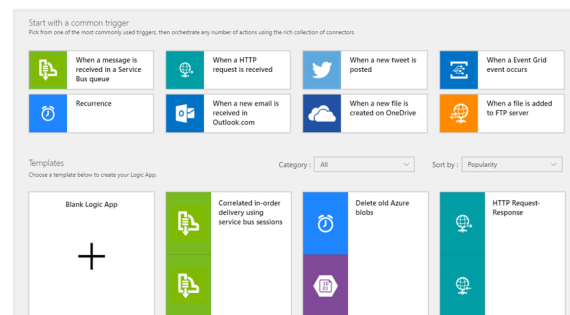
- a. El texto "cena familiar" se reconoce como **keyPhrase**. El texto "el próximo viernes a las 19:00" se reconoce como **datetimeV2**, es decir, un objeto datetime.
12. Esto funciona por ahora. Usemos este modelo para capacitar los servicios y publicarlo. Haga clic en el botón **Capacitar** en la esquina superior derecha de la pantalla.
  - a. Esto realiza la capacitación de machine learning y crea un modelo de machine learning basado en lo que acabamos de ingresar.
  - b. Si lo desea, puede verificar si el servicio funciona ahora como se esperaba probándolo en el ventana de Prueba (que se encuentra junto al botón Capacitar). Escriba "cena familiar el próximo viernes a las 19:00" y observe el resultado.
13. Ahora que tenemos un servicio en funcionamiento, debemos publicar este modelo en producción para poder usarlo. Haga clic en **Publicar** en el menú (que se encuentra junto al botón Capacitar). Esto abre la página Publicar:
  - a. De forma predeterminada, ofrece varios espacios para publicar. Deje el espacio como **Producción**.
  - b. Haga clic en **Publicar**.
  - c. El modelo ahora se publicará para producción. Desplácese hacia abajo para ver **Recursos y claves**. Aquí, copie **Key String**, ya que lo necesitaremos en nuestra Logic App.

¡Listo! Ahora tenemos que crear la Logic App que utiliza el servicio Language Understanding.

## Creación de Logic App

Las nuevas filas escritas en la SQL Database activarán la Azure Logic App que creamos y lo mismo ocurrirá cuando se agreguen nuevos elementos de tarea. A continuación, tomará el valor del elemento de tarea y lo enviará al servicio Language Understanding para su análisis. Cuando encuentre una fecha en el elemento, creará un nuevo evento de calendario en su cuenta de Office365. Manos a la obra.

1. En el portal de Azure, haga clic en el botón **Crear un nuevo servicio**.
2. Busque **Logic App**. Haga clic en Logic App en los resultados de búsqueda para abrir la hoja de Logic App. Haga clic en **Crear** para comenzar.
3. Se abre la hoja Crear Logic App.
  - a. Escriba un **nombre**.
  - b. Seleccione el **grupo de recursos** que generó al crear el servicio Language Understanding.
  - c. Seleccione una **ubicación**.
  - d. Haga clic en **Crear**.
4. Ahora se creará la Logic App. Una vez creada, verá una página de inicio rápido que le preguntará si desea iniciar la Logic App desde una plantilla, como en la Figura 7-5. Elija **Logic App en blanco** para que podamos comenzar desde cero.



**Figura 7-5:** Creación de una Logic App desde una plantilla



5. Ahora podemos comenzar a crear un activador para Logic App. Esta es una acción que inicia el proceso.
  - a. Busque **SQL**.
  - b. Seleccione la tarea **Cuando se crea un elemento**. Esto pedirá la conexión a Azure SQL Database (Figura 7-6).

When an item is created

\* Connection Name

\* SQL Server Name

Name	Resource Group	Location
todoappserver	todoappbarry	southcentralus

\* Username

\* Password

☐ Connect via on-premise data gateway ⓘ

Create Cancel

Manually enter connection information

Figura 7-6: Realización de la tarea *Cuando se crea un elemento*

- c. En este caso, el SQL Server correcto ya está seleccionado, ya que solo tengo uno en este momento. Es posible que deba seleccionar el servidor correcto.
- d. Complete un **nombre** para la conexión y luego escriba el **nombre de usuario** y la **contraseña** en la Azure SQL Database que creamos anteriormente.
- e. Haga clic en **Crear**. Esto crea la conexión y la guarda en su suscripción de Azure. Puede reutilizar esta conexión en otras Logic Apps, si lo desea.
- f. Seleccione la **tabla** que queremos monitorear. En nuestro caso, esta es la tabla **Pendientes**.
- g. Seleccione un **intervalo** y una **frecuencia**, como 5 segundos. Esto le dice al activador que busque nuevas filas cada 5 segundos. Algunos activadores de Logic App son como este, deben sondear para poder activarse. Otros logran que la información les llegue.

¡Listo! Ahora Logic App se iniciará cada vez que ingresemos un nuevo elemento de tarea.

6. Haga clic en el **signo más** que está debajo de la tarea de SQL, y luego seleccione **Añadir una acción** para agregar la siguiente acción (Figura 7-7).

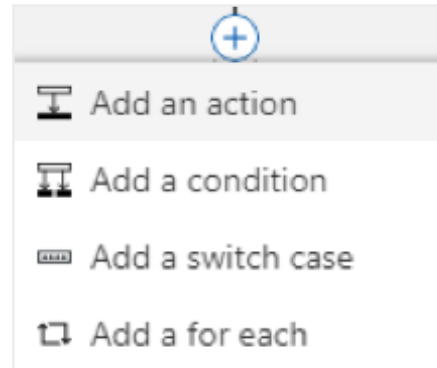


Figura 7-7: Añadir una acción para Logic App

7. Busque **LUIS**. Esto hará que aparezca el servicio Language Understanding. Ahora seleccione **LUIS Obtener acción de predicción**. Le solicitará una conexión a un servicio Language Understanding.
  - a. Escriba un **nombre** para la conexión.
  - b. Pegue en la **clave de conexión** que guardó anteriormente cuando publicamos el modelo de Language Understanding.
  - c. Haga clic en **Crear**.
  - d. Seleccione el **ID de la aplicación** que creó en el portal de Language Understanding.
  - e. Seleccione la **Descripción** de la tarea de SQL como entrada para el campo **Texto de enunciado**.
  - f. Seleccione **Añadir elemento de tarea al calendario** como la **Intención deseada**. Esto aparecerá si la tarea contiene una fecha o no.
8. Haga clic en el **signo más** y luego en **agregar una condición**. Comprobaremos si el texto contiene una fecha al verificar si la intención deseada era verdadera. Si contiene una fecha, continuamos y creamos un evento de calendario. Si no es así, no hacemos nada.

- a. En la condición, seleccione **Es el valor de intención deseado** en la tarea Language Understanding para el **valor**.
  - b. Deje la declaración **es igual a**.
  - c. Agregue **verdadero** en el cuadro de texto de valor que se encuentra después de eso.
9. La condición termina en dos casillas, una por **si es verdadero**, otra por **si es falso**. En la casilla si es verdadero, cree una nueva acción.
- a. Busque **LUIS**.
  - b. Seleccione la acción **Obtener entidad por tipo**. Esta es una acción de Language Understanding que extrae una entidad basada en su tipo de los resultados de Language Understanding.
  - c. Seleccione el **ID de la aplicación** como lo hicimos anteriormente.
  - d. Seleccione **builtin.datetimeV2** para **Entidad deseada**.
  - e. Seleccione el objeto **Predicción de LUIS** para el campo **luisPredictionObject**.
  - f. Debajo de esta acción, agregue otra del tipo **Obtener entidad por tipo**.
  - g. Seleccione el **ID de la aplicación**.
  - h. Seleccione **builtin.keyPhrase** para **Entidad deseada**.
  - i. Seleccione el objeto **Predicción de LUIS** para el campo **luisPredictionObject**.
10. Aún en la casilla **si es verdadero**, cree una nueva acción. Busque **Office 365** y seleccione la acción **Crear evento V2**. Esto puede crear un evento en su calendario de Office 365.
- a. Requiere una **conexión** a Office 365. Haga clic en **agregar nueva conexión** y aparecerá un mensaje de inicio de sesión. Inicie sesión con sus credenciales de Office 365. Una vez que esto se realice correctamente, Logic App mantendrá su conexión en su suscripción de Azure.
  - b. Seleccione el **Calendario** para crear el evento.

- c. En los campos **Hora de término** y **Hora de inicio**, seleccione el **Valor de entidad** en la acción donde filtró la entidad **datetimeV2**.
- d. En el campo **Asunto** seleccione el Valor de entidad de la acción donde filtra la entidad **keyPhrase**.

11. Guarde el flujo de Logic App. En la casilla **si es así**, se debería ver algo así (Figura 7-8):

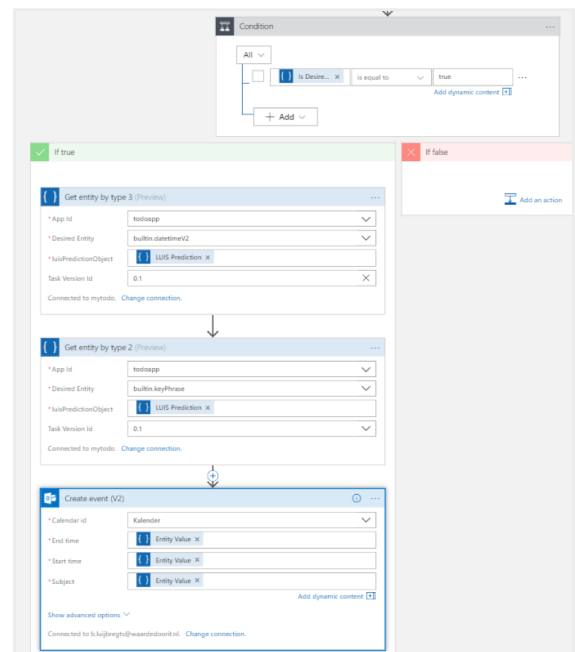


Figura 7-8: Guardar el flujo de Logic App

12. Navegue a la URL de la aplicación de tareas pendientes, que es la URL de su Azure Web App del tutorial anterior.
13. Cree un nuevo elemento de tarea con el texto "cena familiar el próximo viernes a las 19:00". Esto debería crear un evento en su calendario.
14. Además de revisar su calendario, puede ver cómo se ejecutó la Logic App si revisa el **Historial de ejecuciones**. Puede acceder al Historial de ejecuciones cuando abre la Logic App desde el portal de Azure. A partir de ahí, puede volver a enviar el valor para ejecutarlo nuevamente a través de Logic App.

Este ejemplo muestra que puede extender una aplicación con los servicios de Azure solo a través de la configuración y sin cambiar el código. Hemos hecho este ejemplo simple para que pueda seguir fácilmente este tutorial. En un escenario del mundo real, el modelo de Language Understanding debe ser más sólido para poder entender más enunciados. Además, puede hacer que Logic App se active en las ediciones de los elementos de tarea, no solo en la creación de ellos.

## 03 /

# Tutorial: Todo listo para la producción

Con la aplicación ya creada y ejecutándose, ahora puede usar Azure para hacerla más sólida y fácil de actualizar.

## Configuración de entrega continua con GitHub

Hasta ahora, hemos estado insertando código de nuestro repositorio de Git local en Azure. Esto está bien si trabaja por su cuenta, pero si trabaja en un equipo, necesita otro tipo de control de fuente, como **Visual Studio Team Services** o **GitHub**.

Usaremos GitHub para insertar nuestro código y vincularlo con nuestra aplicación web para que los cambios se implementen automáticamente en un proceso de entrega continua.

1. Cree un nuevo repositorio en GitHub dirigiéndose a <https://github.com/new>. (Deberá iniciar sesión).
  - a. Escriba un **nombre** para el repositorio.
  - b. Deje las **otras configuraciones** como están (repositorio público, no crear un README).
  - c. Cree el **repositorio**. Aparece una pantalla que debe asemejarse a la Figura 7-9:

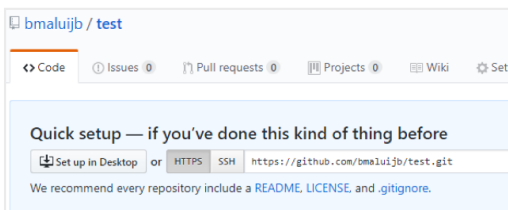


Figura 7-9: Enlace del repositorio en GitHub

2. Use la URL que aparece en la pantalla para establecer un destino remoto para el repositorio de Git local. Puede hacerlo en la ventana de comandos.
  - a. Cambie (cd) al directorio del código fuente de la aplicación.

```
git remote add github https://github.com/
bmaluijb/test.git
```

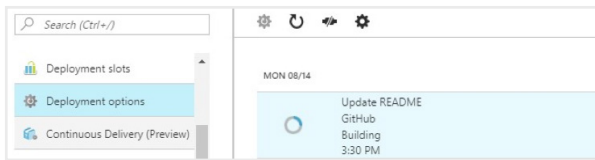
- b. Ejecute el siguiente comando:

```
git push github
```

- c. Ejecute el siguiente comando para insertar el código en GitHub:

Con eso, el código está en GitHub, y puede compartirlo con el equipo. A continuación, vamos a configurar la entrega continua. Vamos a utilizar la función de **opciones de implementación** de Web App a través del portal de Azure. Tenga en cuenta que también podríamos haber utilizado la función de entrega continua en Web App, pero requiere una cuenta de Visual Studio Team Services, y necesitamos realizar algo simple.

1. En el portal de Azure, vaya a la Web App que hospeda la aplicación pendiente de .NET Core.
2. En la barra de menú, haga clic en **Opciones de implementación**.
3. Es posible que esto ya esté configurado para el repositorio de Git local. Si este es el caso, haga clic en **Desconectar**.
4. En **Elegir origen**, seleccione **GitHub**.
5. En la sección **Autorización**, autorice a Azure a usar GitHub haciendo clic en **Autorizar** y otorgando el permiso.
6. En la sección **Elegir proyecto**, elija el repositorio de GitHub que acaba de crear.
7. Deje la Rama como **maestra**.
8. Haga clic en **Aceptar** y espere unos segundos.
9. Vuelva al menú Opciones de implementación. Ahora puede ver que GitHub está conectado. Desde este punto, cada vez que inserte una nueva versión de código fuente en GitHub, se compilará y se insertará en la aplicación web automáticamente, como se muestra en la figura 7-10. También puede forzar este proceso haciendo clic en el botón **Sincronizar**.



**Figura 7-10:** La hoja Opciones de implementación de Web App

## Creación de entornos de ensayo

Con Azure App Service Web Apps, puede configurar una ranura de ensayo en la que probar la nueva versión de la aplicación. Puede utilizar [ranuras de implementación](#) para esto. Las ranuras de implementación son Servicios de aplicaciones integrales que puede utilizar para probar el código fuente antes de promoverlo a la siguiente ranura.

Pueden tener ranuras de implementación para ensayos, pruebas de carga y producción (que es siempre el App Service original, la aplicación web de .NET Core en nuestro ejemplo). De hecho, puede tener tantas ranuras de implementación como desee, sin incurrir en costos adicionales. Las ranuras de implementación se ejecutan todas en el mismo App Service Plan, y eso es lo que paga. Debe tener en cuenta que tener ranuras de implementación adicionales en un App Service Plan consumirá recursos como CPU y memoria. Puede crear nuevas ranuras de implementación desde el elemento de menú **Ranura de implementación**, en la aplicación web. Asegúrese de que está ejecutando la aplicación web en el nivel de precio Estándar o Premium, porque el plan gratuito no viene con ninguna ranura de implementación.

En cada ranura de la implementación que cree, puede configurar las opciones de implementación como lo hicimos anteriormente para implementar el código automáticamente. Incluso puede trabajar en diferentes ramas de código fuente para diferentes entornos y automáticamente implementar ramas específicas para ranuras de implementación específicas.

Además, puede probar la versión final en una ranura de implementación y, cuando esté satisfecho, intercambiarla con la ranura de la producción. Esto prepara la aplicación antes de que se la intercambie, lo que resulta en una implementación sin tiempo de inactividad.

Veamos cómo crear una ranura de implementación e intercambiarla.

1. En el portal de Azure, vaya a la Web App que hospeda la aplicación de .NET Core.
2. En la barra de menú, haga clic en **Ranuras de implementación**. Se abre la hoja Ranuras de implementación.
3. Haga clic en el **signo más (+)** para crear una nueva ranura de implementación.
  - a. Escriba un **nombre** para la ranura (por ejemplo: ensayo).
  - b. Seleccione la aplicación web de .NET Core como la fuente de configuración. (Esto copia la configuración de la aplicación en la nueva ranura.)
  - c. Haga clic en **Aceptar** para crear la ranura.
4. Una vez creada la ranura, es similar a la aplicación web original.
  - a. Configure la entrega continua para la ranura al igual que en el procedimiento anterior para la aplicación web.
  - b. Desconecte la entrega continua en la aplicación web original de .Net Core. De esta forma, cuando inserte nuevo código, se entrega solo a la ranura de ensayo.
5. Realice un cambio a la aplicación de .NET Core.
  - a. Cambie algún texto en el archivo **Index.cshtml**. (Puede encontrarlo en la carpeta Vistas/Inicio).
  - b. Confírmelo en Git e insértelo en GitHub, al igual que cuando implementó la aplicación de .NET Core.

La nueva versión está ahora en la ranura de ensayo y no en la Web App original (lo que llamamos la ranura de la producción). Puede verificar esto si se desplaza a la URL de la aplicación web de .NET Core y a la URL de la ranura de ensayo (que puede encontrar en la hoja de resumen de la ranura, al igual que en la hoja de resumen de la aplicación web).

Pongamos la nueva versión en producción.

1. En el portal Azure, vaya a la aplicación web de .NET Core.
2. En la barra de menú, haga clic en **Ranuras de implementación** para abrir la hoja Ranuras de implementación.
3. Haga clic en el botón **Intercambiar** para abrir la hoja Intercambio.
  - a. No modifique la configuración.
  - b. Haga clic en **Aceptar** para iniciar el intercambio.

Una vez finalizado el intercambio, la nueva versión de la aplicación web .NET Core está en producción. (Puede probarlo si navega a la URL de la aplicación web Node.js.) Utilizar las ranuras de implementación de este modo es bastante cómodo porque le permite probar la nueva versión antes de que esté en producción. Luego, puede implementarla en la producción sin tiempo de inactividad.

## Uso de registros de diagnóstico

Cuando una aplicación se está ejecutando, saber cómo es su desempeño es fundamental. Una gran forma de controlar la aplicación es mediante el uso de registros de diagnóstico para ver diagnósticos de la aplicación web en tiempo real. Incluso puede enviar los registros a la ventana de consola. Para esto, ejecute el siguiente comando en Azure Cloud Shell:

```
az webapp log tail --name <app_name>
--resource-group <myResourceGroup>
```

Verá registros cuando se utilice la aplicación en la aplicación web para generar algo de tráfico.

## Configuración de monitoreo y alertas

[Application Insights](#) ofrece proporciona otra forma poderosa de rastrear el rendimiento de la aplicación. Se trata de una herramienta de control en Azure que puede informarle acerca de muchas cosas en la aplicación, desde cuántos visitantes la utilizaron hasta cuántas excepciones ocurrieron y en qué lugar del código sucedieron. A diferencia de los registros de diagnóstico, Application Insights requiere un precio módico.

Puede configurar Application Insights en el portal de Azure de la Web App.

1. Vaya al portal de Azure y luego a la aplicación web que aloja la aplicación .NET Core.
2. En la barra de menú, haga clic en **Application Insights**.
3. Seleccione **Crear nuevo recurso**.
  - a. Escriba un **nombre** y seleccione una **ubicación** para la instancia de Application Insights.
  - b. Haga clic en **Aceptar**. Application Insights se implementa y comienza a recopilar datos para la aplicación.

Es necesario configurar la aplicación para que comience a enviar datos a Application Insights. Para nuestra aplicación .NET Core de muestra, es necesario hacer lo siguiente:

4. Escalar una aplicación Web a través del portal de Azure
5. En el menú VS Code, seleccione **Proyecto > Añadir telemetría de Application Insights...**
6. Esto abre el asistente de Application Sights. Inicie sesión con su cuenta Azure.
7. Seleccione un **plan de precios** de Application Insights.
8. Haga clic en **Registrarse**. Esto agrega automáticamente todo lo que necesita al proyecto .NET Core y crea el recurso de Application Insights en Azure.
9. Cree el proyecto e incorpore los cambios a GitHub para que se implementen en la aplicación web. Una vez terminada la implementación, la aplicación enviará datos a Application Insights.

Application Insights realiza por defecto una detección inteligente. Esta característica inteligente detecta cuándo algo está mal y le avisa. Por ejemplo, puede detectar un aumento repentino de las solicitudes fallidas o si la aplicación funciona más lento de lo normal. También puede crear sus propios eventos personalizados para todo tipo de mediciones y condiciones desde el menú de **Alertas** de Application Insights, tal como se muestra aquí (Figura 7-11):

	NAME	CONDITION	LAST ACTIVE
Previews	MYGITHUBTEST (COMPONENTS)		
Properties			
Alerts	Failed request per second	Failed Requests per Second > 1	Never

**Figura 7-11:** Alertas de Application Insights

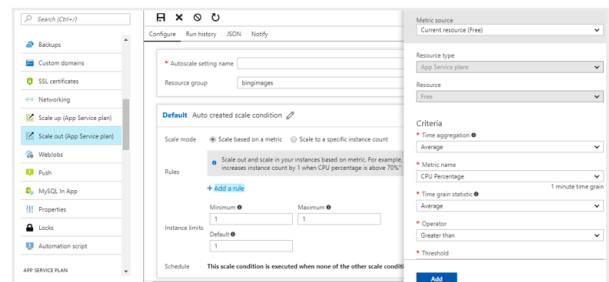
10. Vaya al portal de Azure y haga clic en el recurso de Application Insights. Una vez allí, verá la descripción, que muestra indicadores básicos tales como el tiempo de respuesta del servidor, el tiempo de carga de vista de página y el número de solicitudes de servidor y solicitudes fallidas. Si hay datos ahí, Application Insights está en funcionamiento.

## Escalado de la aplicación web

Cuando tiene muchos usuarios, necesita ampliar la escala de Web App para poder aceptar el aumento del tráfico. Cuando no está ocupado, necesitamos disminuir la escala para ahorrar costos. Puede hacerlo con la característica de **Escalado automático** de App Service.

Tenga en cuenta que para utilizar esta función necesita ejecutar la aplicación web en el plan de tarifas Básico o Premium.

La aplicación web tiene un elemento del menú denominado **Escalar**, como se muestra en la Figura 7-12. Puede usar esto para escalar de forma manual o automática.



**Figura 7-12:** Escalar una aplicación Web a través del portal de Azure

Escalar significa que añade más instancias de la aplicación para manejar la carga. Puede realizar un escalado o una reducción automáticamente, basándose en métricas tales como el porcentaje de utilización de la CPU o de la memoria, en un cronograma (todos los días a las 17:00), o una mezcla de ambos. Esto es muy fácil de configurar y controlar.



## Añadir una Capa de conexión segura (SSL)

Ahora que la aplicación está lista para la producción, debería confirmar que es segura. Además de la autenticación y la autorización, utilizar el protocolo HTTPS para servir la aplicación web es una de las cosas más importantes que puede hacer. Esto es porque sin HTTPS, los intrusos podrían ver el tráfico entre sus recursos y utilizarlos para fines maliciosos (como iniciar sesión en la aplicación). Además, el protocolo HTTPS es un requisito para características innovadoras, como [service workers](#).

Es posible servir el tráfico a la aplicación web mediante una Capa de conexión segura (SSL) si se importa un certificado SSL a Web App y se enlaza a uno de sus nombres de dominio (personalizados). Puede importar el certificado SSL que adquirió o comprar uno nuevo a través de [Azure App Service Certificates](#), lo cual facilita la compra y la validación del certificado. Después de importar el certificado, emparejelo a uno de los enlaces de nombre de dominio de la aplicación web. Puede hacer todo esto desde el menú de **certificados SSL** en la aplicación web.

## Notificación acerca de nuevas versiones a los usuarios

Su empresa se beneficiará al hacer que los usuarios estén al tanto de las nuevas versiones de producción. Al extender el proceso de CI/CD en las versiones Azure, es posible usar WebHooks en eventos de PostDeployment para activar un flujo de trabajo de Azure Logic Apps que organice publicaciones en redes sociales, como enviar tweets o realizar publicaciones con notas de la versión.

## 04 /

# Tutorial: Uso de Azure API Management para controlar API y generar documentación

La mayoría de las empresas de hoy en día necesitan interactuar con servicios en línea en algún nivel. Estos pueden ser desarrollados y mantenidos por la propia empresa o consumidos como servicios externos. El verdadero objetivo es ofrecer funcionalidad, pero existen otras tareas relacionadas con la publicación de una API, como seguridad, pruebas y control de versiones, que deben implementarse para garantizar que la aplicación sea segura y confiable.

En este ejemplo, usaremos una API web de Node.js para ayudar a nuestro negocio de conversión de unidades a crecer. La API web se implementará en Azure y utilizaremos los servicios de Administración de API de Azure para lo siguiente:

- Lanzar versiones de la API.
- Limitar el acceso con una versión de evaluación de la API del producto.
- Revisar la documentación de API generada automáticamente.

Para completar este tutorial, asegúrese de iniciar sesión en Azure con una suscripción válida para crear los recursos necesarios. Si no tiene una cuenta Azure, puede [crear una gratis](#).

## Configuración de la API de Azure Portal Web

Cuando planificamos ofrecer acceso a nuestra herramienta de conversión de unidades, decidimos que nuestro servicio de conversión interna debería implementarse en Azure para que nuestros clientes pudieran utilizarlo.

1. El código fuente de nuestra API está alojado en GitHub. Acceda e impleméntelo rápidamente en Azure haciendo clic en el botón **Implementar en Azure** (Figura 7-13) (enlace al repositorio: <https://github.com/mbcrump/converter-api>).



Figura 7-13: Botón Implementar en Azure

2. Su navegador será redirigido a la página de configuración de **Implementación de Azure API**. Siga las instrucciones del asistente para completar el proceso de publicación de la API. El proceso de implementación debería tomar unos minutos.
3. Ahora que se ha implementado la API web, acceda al portal de Azure y navegue hasta el grupo de recursos que acaba de crear. Luego, verifique que la aplicación web esté en funcionamiento.
4. Asegúrese de guardar la **URL de la API web** recién creada. La necesitará después, mientras configura el servicio de API Management.

## La creación y configuración de un servicio de API Management para la API web

Con la API web publicada, es hora de agregar un nuevo servicio de API Management a nuestra solución.

1. Haga clic en el botón **+ Añadir**, busque API Management y seleccione el servicio **API Management** para que se agregue.
2. Complete la información en la hoja **Crear**, asegurándose de seleccionar el mismo grupo de recursos creado para este tutorial del servicio de API Management.

Este proceso generalmente tarda un tiempo en completarse. Debería recibir un correo electrónico cuando se crea el servicio de API Management. Puede encontrarlo fácilmente dentro del grupo de recursos seleccionado.

## Lanzamiento de versiones de la API web

1. Seleccione el servicio de API Management que acabamos de crear. Luego, cree un nuevo proxy de API seleccionando el elemento del menú **API** en el panel izquierdo.
2. Aparecerán algunas opciones como la especificación OpenAPI, WADL, WSDL y las Logic Apps. Para este recorrido, crearemos una **API en blanco**.
3. Complete el formulario que se presenta:
  - a. Establezca el Nombre como **Measurement Units conversion API**.
  - b. Establezca el Nombre como **measurement-units-conversion-api**.

- c. Establezca el esquema URL como **Ambos**.
- d. Para activar el control de versiones, busque la marca de **¿Cuál es la versión de esta API?**
- e. Establezca el esquema de versión como **Ruta**.
- f. Establezca el identificador de versión como **v1**.
- g. Deje todos los demás campos en su valor predeterminado.

Tenga en cuenta que el esquema de versión se establece en Ruta, y así es como direccionaremos las solicitudes a las diferentes versiones de nuestra API. Otras opciones disponibles para el esquema de versión son Parámetros de cadena de consulta e Información de encabezado de solicitud.

Debiera aparecer una pantalla de configuración del servicio de administración de API. Aquí es donde conectaremos la API Management con el código API web real implementado en Azure agregando la conversión back-end a v1.

1. Para editar el back-end:
  - a. Haga clic en el botón **Editar** en la esquina superior derecha.
  - b. Marque **anular** y luego ingrese la **URL de API web** que creamos previamente mientras publica la API web en Azure.
  - c. Establezca las Credenciales de puerta de enlace en **Ninguna**, y luego haga clic en **Guardar**.
2. Cree la operación del punto de entrada para ser expuesto por el servicio API haciendo clic en el botón **+ Añadir operación** e ingresando la siguiente información:
  - a. Nombre y nombre de la operación: **Medidas**
  - b. Método de acceso HTTP: **GET**

3. Navegue hasta la pestaña **Consulta** y agregue los parámetros para que la API calcule conversiones. Deje todos los demás campos y pestañas establecidos en sus valores predeterminados (Figura 7-14).
4. Cuando haya terminado, haga clic en el botón **Crear** para guardar la configuración del punto de conexión de la API.

NAME	DESCRIPTION	TYPE	VALUES	REQUIRED
FROM	From Unit	String		Yes
TO	To Unit	String		Yes
VALUE	Value to convert	Number		Yes

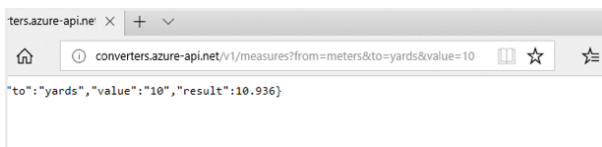
Figura 7-14: Añadir parámetros a la pestaña Consulta

Antes de que podamos acceder a nuestra API, debemos asociarla con un producto, que es una forma de controlar el acceso a una API. En este ejemplo, crearemos un producto de evaluación para nuestra API, que permitirá a los usuarios enviar un número limitado de solicitudes dentro de un tiempo específico. Esto garantizará que puedan probar nuestra aplicación antes de decidirse a seguir adelante con un plan de suscripción real.

1. Regrese a la hoja de API Management, seleccione **Producto** y luego haga clic en el botón **+ Añadir**.
2. Complete la siguiente información:
  - a. Establezca el nombre como **Evaluation Version**.
  - b. Establezca el ID como **evaluation-version**.
  - c. Agregue una descripción y marque el estado como **Publicado**.
  - d. Desmarque la opción **Requiere suscripción**.
3. Haga clic en **Crear**.
4. Debiera ver el producto recién creado en la Lista de productos. Haga clic para establecer la relación con nuestro servicio API:

- Haga clic en el menú **API**.
- Haga clic en el botón **+ Add**.
- Seleccione la **API de medidas**.

En este punto, debe poder acceder a la API de conversión desde su navegador siguiendo esta URL: <http://YOUR-API-URL/v1/measures?from=meters&to=yards&value=10> (Figura 7-15).



**Figura 7-15** Acceso a la API de conversión desde un navegador

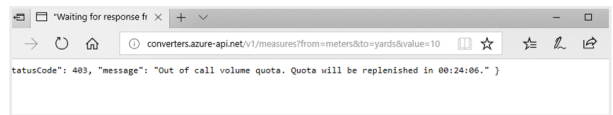
Ahora es el momento de limitar el número de solicitudes anónimas de un cliente específico para garantizar que nuestra versión de Evaluación no se utilice de forma incorrecta.

- Navegue de nuevo a la información general del servicio de **API Management de medidas**.
- Haga clic en el menú **Producto** y luego seleccione **Evaluation Version** en la lista de productos.
- Desde el interior de la página de detalles de la versión de evaluación, haga clic en el elemento de menú **Directivas** a la izquierda.
- Establezca la siguiente configuración de cuota de directiva **entrante**:

```
<quota calls="1000" renewal-period="3600" />
```

Esto garantiza que ningún cliente podrá enviar más de 1000 solicitudes al servicio en un plazo de 30 minutos desde la primera solicitud. Esto debería ser suficiente para garantizar una evaluación adecuada del uso de nuestra API de conversión de unidades de medida. Tan pronto como el recuento de solicitudes alcance el límite establecido para las cuotas de entrada, todas las solicitudes subsiguientes serán rechazadas hasta que los límites

de la cuota se restablezcan después de 30 minutos (Figura 7-16).



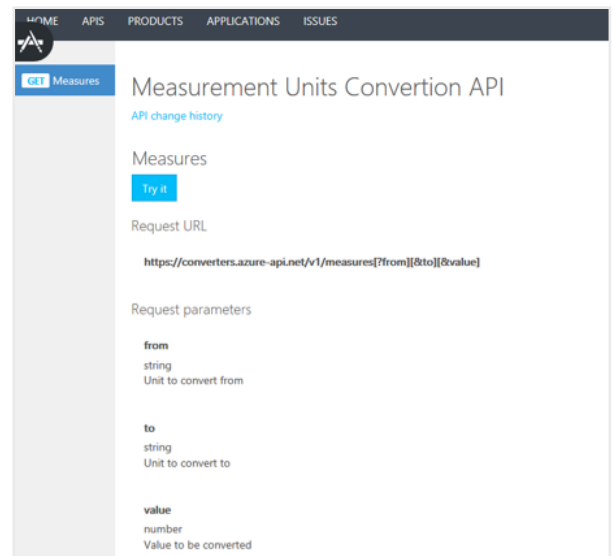
**Figura 7-16:** Configuración de cuotas para limitar las presentaciones

## Revisión de la documentación de API generada automáticamente

Ahora que hemos configurado nuestros puntos de entrada y back-ends de API, echemos un vistazo a la documentación que nuestro servicio Azure API Management creó automáticamente.

Desde la hoja Información general de la API, puede acceder fácilmente al portal para desarrolladores haciendo clic en el botón **Portal del desarrollador** en la barra superior o en el enlace con la etiqueta **URL del portal del desarrollador**.

La documentación generada es un recurso útil para que los desarrolladores investiguen nuestra API, aprendan sobre la creación de solicitudes de prueba y vean ejemplos de comunicación en varios lenguajes, como Ruby, C # y JavaScript (Figura 7-17).



**Figura 7-17:** Documentación generada automáticamente a través de Azure API Management

# Resumen y próximos pasos

En esta guía, le presentamos la capacidad que Azure ofrece para mejorar las aplicaciones. Si utiliza Azure, puede hacer cosas increíbles con las aplicaciones: utilizar el reconocimiento facial y de voz, administrar sus dispositivos conectados a Internet de las cosas en la nube, escalar cuanto usted quiera y pagar únicamente por lo que usa.

Pudo ver que no importa qué lenguaje de programación use o para qué plataforma escribe aplicaciones: Azure puede ayudarlo con servicios para casi cualquier situación. Esperamos que siga usando este eBook para conocer mejor la amplia gama de servicios de Azure y determinar cuáles son los más aplicables a sus necesidades.

Ya no es necesario realizar los complicados trabajos de instalación por cuenta propia; ahora puede aprovechar una amplia variedad de soluciones prediseñadas. Tendrá más tiempo para trabajar en lo que importa al dejar que Azure se ocupe de problemas que ya tiene resueltos.

# Los autores de este libro son unos apasionados por Azure, le recomendamos ponerse en contacto con ellos ante cualquier pregunta



**Michael Crump** trabaja en Microsoft en la plataforma Azure y es programador, bloguero y conferencista internacional sobre diversos temas de desarrollo de la nube. Lo apasiona ayudar a que los desarrolladores comprendan los beneficios de la nube en forma clara y precisa.

Puede ponerse en contacto con Michael en su Twitter [@mbcrump](#) o en su blog: <https://www.michaelcrump.net>.



**Barry Luijbregts** es arquitecto de software independiente y desarrollador, y es un apasionado por la nube. También es autor de Pluralsight. Ha trabajado para muchas empresas en los últimos 10 años y está dispuesto a compartir su conocimiento con la comunidad. Tiene un conocimiento amplio y profundo sobre la pila de Microsoft y un interés particular por la tecnología web y la nube. Barry es codirector de un grupo de usuarios enfocado en la tecnología y en las habilidades blandas llamado .NET Zuid (South) y actualmente instruye acerca de los beneficios de la nube. Vive en los Países Bajos con su bella esposa e hija y le encanta jugar con sus dos huskies siberianos.

Puede ponerse en contacto con Barry en su Twitter [@AzureBarry](#) y a través de su sitio web: <https://www.azurebarry.com/>.

**PUBLICADO POR** Microsoft Press, una división de Microsoft Corporation  
One Microsoft Way, Redmond, Washington 98052-6399

**Copyright © 2018 por Microsoft Corporation. Todos los derechos reservados.**  
Ninguna parte del contenido de este libro se podrá reproducir ni transmitir de ninguna forma ni por ningún medio sin el permiso previo por escrito del editor.

Los libros de Microsoft Press están disponibles en librerías y a través de distribuidores en todo el mundo. Si necesita asistencia relacionada con este libro, envíe un correo electrónico al soporte de Microsoft Press en [mspinput@microsoft.com](mailto:mspinput@microsoft.com). Cuéntenos qué opina sobre este libro en esta [encuesta](#).

Este libro se entrega "tal cual" y expresa las visiones y opiniones de los autores. Las visiones, las opiniones y la información expresadas en este libro, incluidas las direcciones URL y otras referencias a sitios web de Internet, están sujetas a cambios sin previo aviso. Algunos ejemplos mencionados aquí tienen un carácter únicamente ilustrativo y son ficticios. No debe suponerse ni derivarse ninguna asociación o conexión real.

Microsoft y las marcas comerciales que aparecen en [www.microsoft.com](http://www.microsoft.com) en la página web "Marcas Registradas" son marcas comerciales del grupo de empresas de Microsoft. Todas las demás marcas son propiedad de sus titulares correspondientes.

# Siga aprendiendo con Azure

Cree una [cuenta gratuita de Azure](#) y obtenga:

- USD 200 de crédito para utilizar en cualquier producto Azure por 30 días.
  - Acceso gratuito por 12 meses a nuestros productos más populares, incluidos los procesos, las redes de almacenamiento y las bases de datos.
  - Más de 25 productos que son siempre gratuitos.
- Ve el webinar [Introducción a Azure](#), en el que se ofrece una demostración de los aspectos básicos de Azure y se brinda acceso a expertos de forma continua.
- Visite [Azure.Source](#) para mantenerse al tanto de lo que ocurre en Azure, incluidas las novedades en vista previa, productos disponibles para el público general, noticias y actualizaciones y más.
- Consulte [Consejos y trucos de Azure](#), una recopilación de ideas útiles que le ayudarán a sacar el máximo provecho de Azure.

