



# Cree aplicaciones modernas en AWS

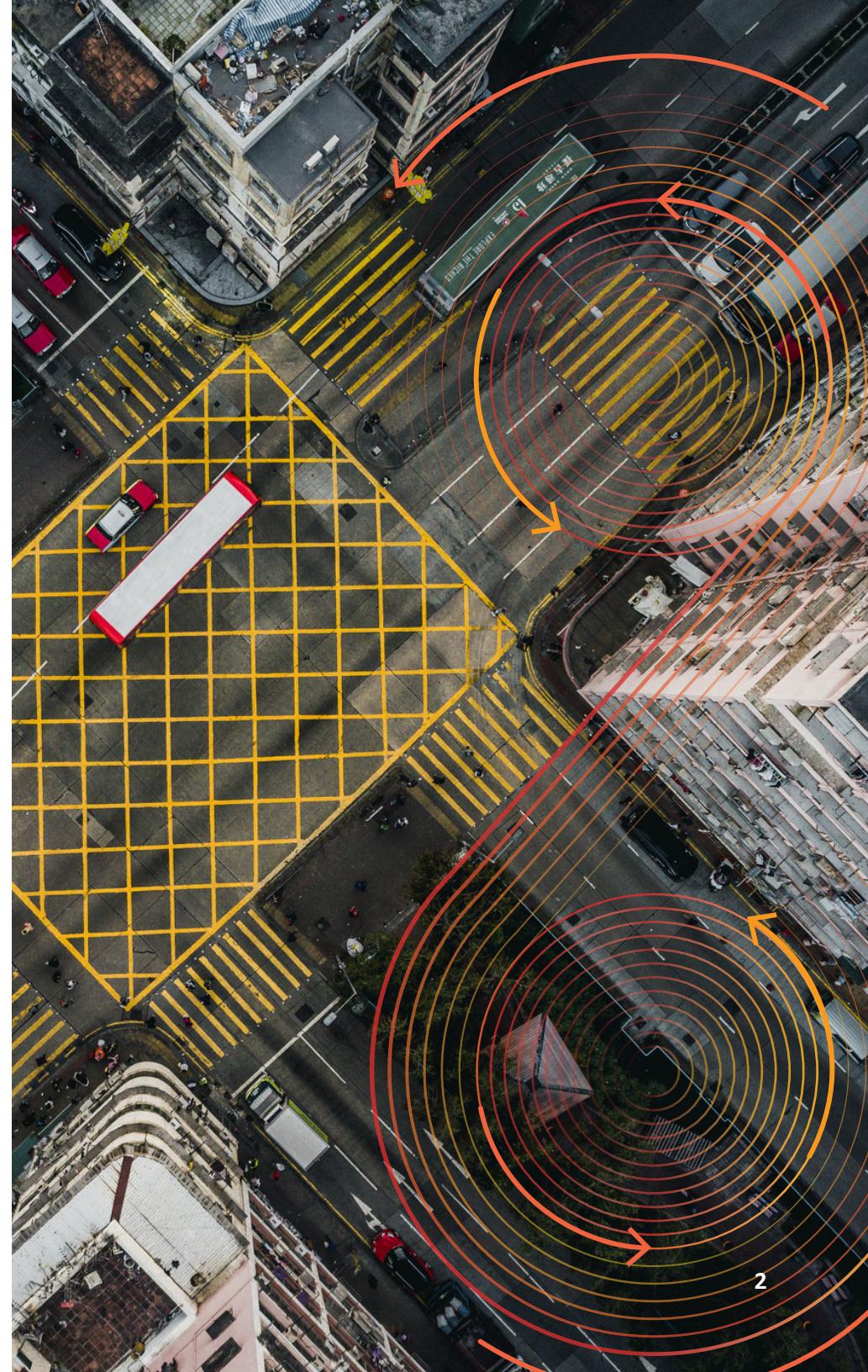
Administre menos. Cree más rápido. Innove más.

## Las aplicaciones modernas están cambiando la forma de entregar valor al cliente

En los próximos años, las organizaciones crearán más de 500 millones de aplicaciones nuevas, es decir, más que la cantidad de aplicaciones desarrolladas en los 40 años anteriores juntos.<sup>1</sup> Muchas de estas organizaciones se enfrentan con dificultades a la hora de crear las aplicaciones, ya que se esfuerzan por encontrar un equilibrio entre cómo administrar la tecnología y cómo ofrecer nuevas funciones.

Si bien la nube promete agilidad, esta no se produce automáticamente. A medida que las organizaciones buscan acelerar la innovación, aprovechar mejor sus datos y crear nuevas experiencias para los clientes, necesitan modernizar la forma en que crean y operan las aplicaciones. Las aplicaciones modernas se crean con una combinación de patrones de arquitectura modular, modelos operativos sin servidor y procesos ágiles de desarrolladores.

En este libro electrónico, lo guiaremos a través de los tres procesos que le ayudarán a sentar las bases para el desarrollo de aplicaciones modernas en su propia organización. También exploraremos cómo el desarrollo de aplicaciones modernas con AWS puede ayudar a su organización a innovar, reducir costos, acelerar el tiempo de comercialización y mejorar la fiabilidad.



## La innovación implica escuchar a los clientes

En un informe reciente de Vision Report, *Digital Rewrites the Rules of Business*, Forrester Research define la mentalidad centrada en el cliente de un innovador digital. La misión principal de estos disruptores modernos es la siguiente:

*“... Aprovechar los activos y ecosistemas digitales para mejorar los resultados de los clientes continuamente y, al mismo tiempo, mejorar la excelencia operativa... con la aplicación de un pensamiento digital respecto de las experiencias del cliente, las operaciones, los ecosistemas y la innovación”.*

El enfoque en los clientes implica tomar decisiones empresariales con la perspectiva del cliente como punto de partida. Implica hacer evolucionar los productos y servicios constantemente para mejorar la entrega de los resultados que esperan los clientes. Y eso significa que debe prestarle atención a aquello que los clientes consideran importante para poder continuar con las invenciones e iteraciones en su favor. Esto se llama el “volante de innovación”.

La idea principal es que el motor de cualquier innovación comienza con la demanda del cliente, mejora a partir de su retroalimentación y realiza repeticiones constantes (y de manera rentable) hasta que la demanda cambia y todo el ciclo comienza otra vez. Cuanto más rápido los equipos pongan en marcha su propio volante de innovación, usted será más capaz de crear aplicaciones modernas que lo diferencien aún más de sus competidores.



La creación de aplicaciones modernas en AWS le permitirá salir al mercado más rápidamente. Mediante la aceleración del ciclo de creación-lanzamiento y la transferencia de la sobrecarga operativa, los desarrolladores pueden crear características nuevas con mayor rapidez. Así, aumentará la innovación con una arquitectura modular que permitirá a los equipos experimentar con componentes individuales sin poner en riesgo toda la aplicación. La automatización de los procedimientos de prueba y el monitoreo en todas las etapas del ciclo de vida del desarrollo servirán para mejorar la fiabilidad. Además, mejorará el costo total de propiedad (TCO) con un modelo de precios de pago por valor que reduce el costo del exceso de aprovisionamiento o el costo de recursos innecesarios.

Para crear aplicaciones modernas, es posible que haya que reconsiderar las bases sobre las que se construyen. Aunque el cambio de esta arquitectura puede ser radical a nivel organizacional, el proceso no tiene que ser excesivamente complejo. Muchas organizaciones se inspiran y avanzan hacia la creación de nuevas aplicaciones modernas en la nube, pero muchas otras adoptan un enfoque híbrido y a menudo se inclinan hacia un enfoque de equipo por equipo y carga de trabajo por carga de trabajo, avanzando de forma conveniente un paso a la vez.

# 50 %

de la tecnología de la información y las comunicaciones se prevé que se asignará directamente a la transformación digital para el año 2023

# 67 %

de los ejecutivos creen que deben acelerar el ritmo para mantener su competitividad

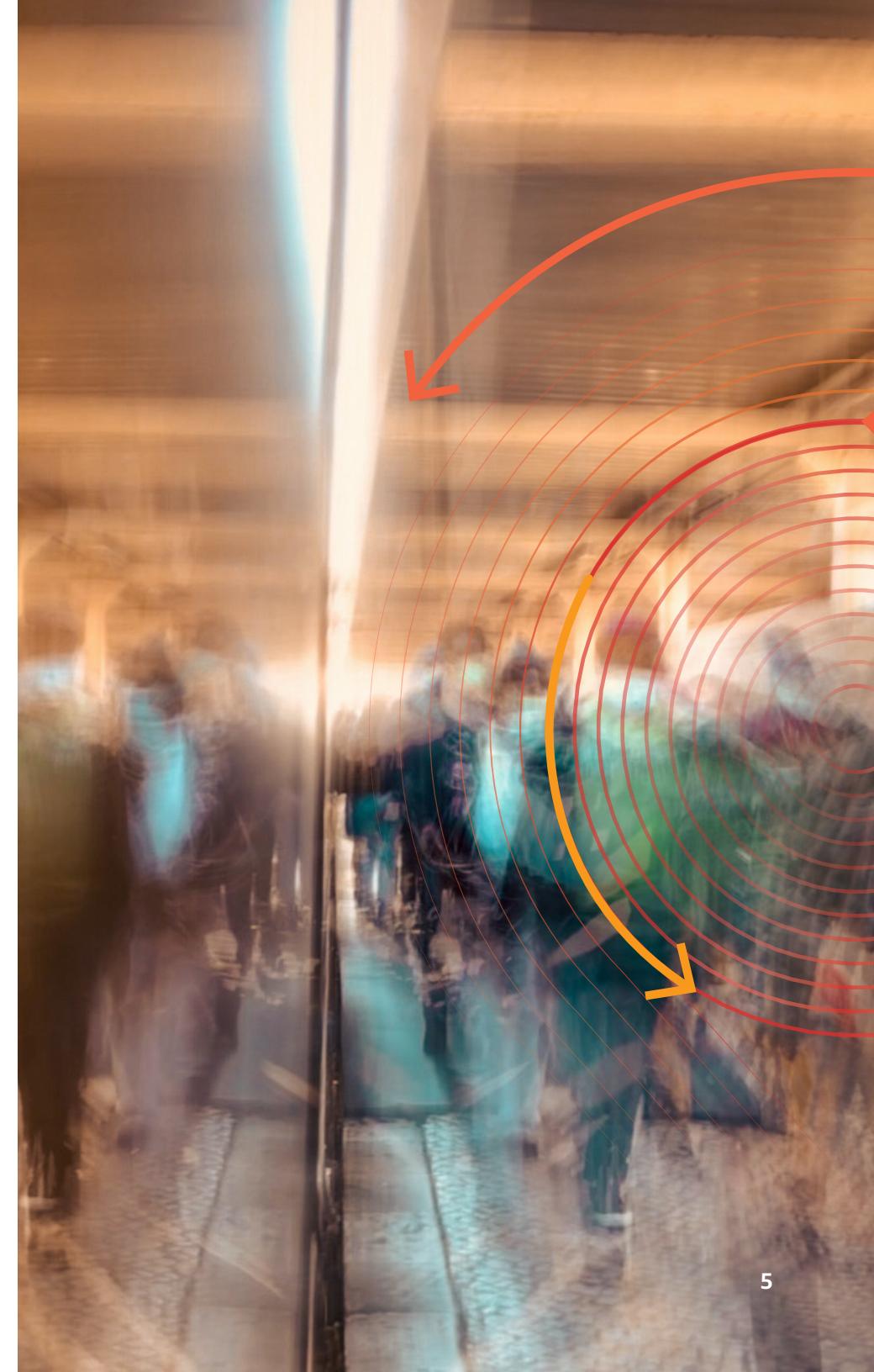
# 90 %

de las aplicaciones nuevas se prevé que serán nativas de la nube para el año 2025

Gracias a nuestra experiencia en la creación de aplicaciones para Amazon.com y al servicio prestado a millones de clientes de AWS, hemos observado tres procesos que las organizaciones pueden seguir para convertir su visión de modernización de las aplicaciones en una realidad, lo que a su vez generará valor para el negocio.

- 1** Redefinición de la plataforma en servicios de contenedores administrados. Las organizaciones que ya ejecutan contenedores en las instalaciones o que están pensando en migrar las aplicaciones a contenedores pueden redefinir esas cargas de trabajo en servicios de contenedores en AWS para simplificar las operaciones y reducir los costos generales de administración, como la organización y el aprovisionamiento de la infraestructura.
- 2** Creación de aplicaciones nuevas en una arquitectura sin servidor. Conforme las organizaciones crean aplicaciones o funciones nuevas, recomendamos que utilicen tecnologías sin servidor y bases de datos personalizadas a fin de maximizar la agilidad, como también herramientas de desarrollo avanzadas para acelerar el desarrollo.
- 3** Transformación en un modelo de DevOps moderno. Para crear un cambio cultural que permita crear aplicaciones modernas a gran escala, las organizaciones pueden aprovechar los servicios y las herramientas de DevOps y mantener a su vez un alto nivel de seguridad y gobernanza.

Exploraremos cada proceso con más detalle, y demostraremos cómo cada uno de ellos puede ayudar a aumentar la agilidad, reducir los costos y crear mejores aplicaciones que apoyen el éxito del negocio. Aunque puede comenzar a modernizar aplicaciones en cualquier momento, el resultado debe ser el mismo: aplicaciones seguras, confiables, escalables y de alta disponibilidad para sus clientes y socios desde un principio.



## Los tres procesos hacia el desarrollo de aplicaciones modernas

El desarrollo de las aplicaciones modernas es un enfoque convincente para abordar el diseño, la creación y la administración de software en la nube. Este enfoque comprobado aumenta la agilidad de los equipos de desarrollo y la confiabilidad y seguridad de las aplicaciones, lo que permite crear y lanzar mejores productos con más rapidez. A partir de nuestra experiencia en ayudar a organizaciones de todo tipo a crear aplicaciones, hemos identificado tres pilares de soluciones para el desarrollo de aplicaciones modernas que le ayudarán en su camino hacia la modernización.

## Procesos para crear aplicaciones modernas

- 1 Redefinición de la plataforma en servicios de contenedores administrados**
- 2 Creación de aplicaciones nuevas y seguras en una arquitectura sin servidor**
- 3 Transformación en un modelo de DevOps moderno**

## Redefinición de la plataforma en servicios de contenedores administrados

Los contenedores son una forma ligera y portátil de ejecutar e implementar aplicaciones. El uso de contenedores en las aplicaciones existentes es a menudo un primer paso en el camino hacia la modernización de una organización. Si está pensando en migrar sus aplicaciones a contenedores, se beneficiaría al redefinir esas cargas de trabajo en servicios administrados de AWS, como [Amazon Elastic Kubernetes Service \(Amazon EKS\)](#) o [Amazon Elastic Container Service \(Amazon ECS\)](#) con [AWS Fargate](#).

[Fargate](#). Un servicio administrado de contenedores ayuda a reducir la carga operativa al tiempo que mejora la escalabilidad, fiabilidad, seguridad y disponibilidad. Con los servicios administrados de contenedores en AWS, ya no tendrá que preocuparse de la administración de los contenedores. En su lugar, puede dedicar sus recursos a la capacitación que necesita para desarrollar aplicaciones modernas con informática sin servidor.

# Vanguard®

Desde que comenzó su migración a AWS, en 2015, The Vanguard Group ha obtenido beneficios significativos. Por ejemplo, al utilizar los servicios de AWS, el departamento de TI de Vanguard eliminó la necesidad de administrar servidores. De este modo, los desarrolladores tienen más tiempo para crear microservicios nuevos e innovadores y mejorar las aplicaciones actuales, lo que aumenta la velocidad de comercialización de Vanguard de tres meses a 24 horas.



*"AWS Fargate Spot ha reducido nuestros costos unitarios y fortalecido nuestro caso empresarial para migrar a AWS. Gracias a esta optimización, todos los meses ofrecemos más valor por nuestro dinero. Devolver el valor a nuestros accionistas a través de una mayor eficiencia es el eje de la misión de nuestra empresa".*

– Tim Treston, gerente senior, Cloud Business Office, Vanguard

Sin embargo, la comercialización no es el único beneficio que la empresa ha obtenido. Vanguard decidió utilizar [Amazon Elastic Container Service \(Amazon ECS\)](#), un servicio de organización de contenedores completamente administrado, junto con AWS Fargate. [AWS Fargate](#) es un servicio informático sin servidor que elimina la necesidad de aprovisionar y administrar [Amazon Elastic Compute Cloud \(Amazon EC2\)](#), un servicio web que ofrece capacidad informática segura y de tamaño modificable en la nube. Y, al aprovechar la ventaja de una nueva opción de compra de AWS Fargate, la empresa de servicios financieros redujo sus costos unitarios en un 50 %.

[Vea la historia completa »](#)

La modernización del desarrollo de aplicaciones consiste en la adopción de servicios, prácticas y estrategias que permiten a los desarrolladores crear aplicaciones más ágiles. Además, con la velocidad y la fiabilidad de la infraestructura moderna, los desarrolladores pueden entregar aplicaciones seguras que se escalen desde el prototipo hasta millones de usuarios de forma automática, para que puedan innovar y responder al cambio más rápidamente.

Muchas aplicaciones modernas se construyen primero sin servidor, una estrategia que da prioridad a la adopción de servicios sin servidor para que los clientes puedan aumentar la agilidad en toda la pila de aplicaciones. Con las tecnologías sin servidor, ya no tendrá que administrar servidores físicos y se beneficiará del escalado automático, la alta disponibilidad incorporada y un modelo de facturación de pago por valor. En lugar de preocuparse por la administración y el funcionamiento de los servidores o los tiempos de ejecución, puede enfocarse en la innovación de los productos y disfrutar, al mismo tiempo, de una comercialización más rápida.

Asimismo, se pueden crear herramientas y servicios web y móviles front-end basados en AWS. La fiabilidad de esta infraestructura ayuda a las marcas a entregar aplicaciones seguras y de alta disponibilidad que pueden escalarse automáticamente en todo el mundo.

## Consideraciones clave para crear aplicaciones modernas escalables

### *Patrones arquitectónicos: microservicios*

Aunque su aplicación monolítica puede ser fácil de administrar en la actualidad, los desafíos suelen surgir a medida que crece, y esto incluye cómo distribuir la propiedad de la aplicación entre sus equipos. Puede crear una cultura de propiedad sólida, pero aun así se le dificultará el escalado vertical si la arquitectura de la aplicación incluye dependencias estrictas que impidan que los equipos asuman la propiedad del producto final. Por esta razón, recomendamos la creación de arquitecturas de microservicios para las aplicaciones que crecen y cambian rápidamente. Los microservicios son la expresión arquitectónica de una cultura de propiedad: dividen aplicaciones complejas en componentes organizados que pueden estar bajo la responsabilidad de un solo equipo y ejecutarse de manera independiente.

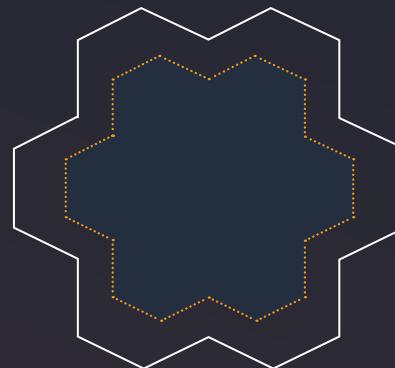
Con una aplicación monolítica, tendrá muchos desarrolladores que realizan cambios mediante una canalización de lanzamiento compartida, lo que provoca fricciones en muchos momentos del ciclo de vida. Durante el desarrollo, los ingenieros deben coordinar los cambios para asegurarse de no alterar el código de otra persona. Para actualizar una biblioteca compartida a fin de aprovechar una nueva característica, debe convencer a todos de hacer la actualización al mismo tiempo, lo cual puede ser una tarea muy difícil. Y, si desea implementar rápidamente un cambio importante para la característica, tendrá que fusionarlo con los cambios que ya están en curso.

Después del desarrollo, también enfrentará sobrecargas al enviar los cambios a través de la canalización de entrega. Incluso al realizar un cambio de una sola línea en un pequeño fragmento de código, los ingenieros deben coordinar los cambios con anticipación, fusionar los códigos, resolver los conflictos de los lanzamientos, volver a elaborar toda la aplicación, ejecutar todas las pruebas y realizar la implementación nuevamente.

Con una arquitectura de microservicios, las aplicaciones están conformadas por componentes independientes que ejecutan cada proceso como un servicio. Los servicios se crean para las capacidades empresariales y cada uno realiza una sola función. Como se ejecutan de manera independiente y un solo equipo de desarrollo los administra, todos los servicios se pueden actualizar, implementar y escalar para cumplir con las demandas de funciones específicas de una aplicación. Por ejemplo: un mayor número de usuarios puede usar un carrito de compras en línea durante una venta. Los microservicios comunican datos entre sí mediante interfaces bien definidas, con API, eventos o flujos livianos. Nuestros clientes confían cada vez más en las arquitecturas basadas en eventos (aquellas en las que las acciones se inician en respuesta a los cambios en los datos) para mejorar la escalabilidad y la resiliencia de las aplicaciones. Esto les permite, simultáneamente, reducir costos.

## TODO VS. UNA SOLA COSA: DOS TIPOS DE APLICACIONES

### Aplicaciones monolíticas



Se encargan de todo

Aplicación única

Debe implementar la aplicación completa

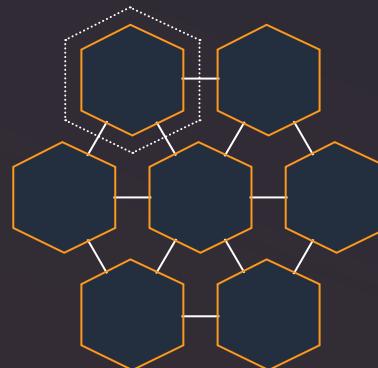
Base de datos única

Organizadas en capas tecnológicas

Estado en cada instancia de tiempo de ejecución

Una pila tecnológica para toda la aplicación

### Microservicios



Se encargan de una sola cosa

Servicios de funciones mínimas

Implementadas por separado, interactúan juntas

Cada una tiene su propio almacén de datos

Organizadas según las capacidades empresariales

El estado se externaliza

Se puede elegir la tecnología para cada microservicio



# centrica

Centrica es una empresa británica multinacional de energía y servicios que buscaba reducir los costos y aumentar la agilidad mediante el cambio de la arquitectura de su aplicación. La empresa optó por refactorizarse con una arquitectura de microservicios y adoptar una estrategia sin servidor para alcanzar esas metas.

Para cambiar la organización, Centrica creó un grupo de trabajo sin servidor con equipos de representantes y comenzaron a elaborar un piloto en conjunto. A partir de ese éxito, otros equipos de la organización también adoptaron ese enfoque: la arquitectura sin servidor ya es moneda corriente en toda la organización.

Con la arquitectura sin servidor, Centrica puede ver y responder a los problemas del cliente en tiempo real, algo que no había podido hacer anteriormente.

[Vea el video »](#)

# Modelo operativo sin servidor

## *Sin servidor, tanto como sea posible*

A medida que sus patrones arquitectónicos y procesos de entrega de software cambian, querrá adoptar un modelo operativo que le permita delegar las actividades que no sean fundamentales para la empresa. Para obtener la agilidad necesaria para innovar rápidamente, recomendamos la creación de una arquitectura de microservicios, el uso e implementación de software mediante la automatización para tareas como el monitoreo, el aprovisionamiento, la administración de costos, las implementaciones y la seguridad y gobernanza de las aplicaciones. La elección de una estrategia principalmente sin servidor (con la selección de tecnologías sin servidores cuando sea posible), permite maximizar los beneficios operativos de AWS.

El modelo operativo sin servidor permite crear y ejecutar aplicaciones y servicios sin aprovisionar los servidores ni administrarlos. Esto elimina la administración de servidores, proporciona una escalabilidad flexible, permite pagar solo por el valor y automatiza la alta disponibilidad. Este modelo permite crear y administrar los aspectos de la aplicación que entregan valor para el cliente sin tener que preocuparse por los detalles subyacentes.

Ya sea que cree aplicaciones nuevas para la red o que migre aplicaciones heredadas, la creación con funciones primarias sin servidor para la informática, los datos y la integración le permitirá beneficiarse de la máxima agilidad que puede ofrecer la nube.



## ¿Cómo definimos un modelo sin servidores en AWS?

Cuando hablamos de modelos sin servidores, nos referimos a la eliminación de las tareas complejas no diferenciadas, es decir, de las operaciones con servidores. Es una distinción importante porque permite enfocarse en la creación de la aplicación en lugar de enfocarse en la administración y el escalado de la infraestructura que respalda la aplicación. Los cuatro principios de un modelo operativo sin servidores son los siguientes:

- 1 Sin administración de servidores:** no existe la necesidad de aprovisionar o mantener servidores. No hay un software o tiempo de ejecución para realizar tareas de instalación, mantenimiento o administración.
- 2 Escalado flexible:** la aplicación se puede escalar automáticamente o mediante el ajuste de su capacidad con la activación de las unidades de consumo (por ejemplo, rendimiento, memoria) en lugar de activar unidades de servidores individuales.
- 3 Pago por valor:** en lugar de pagar por unidades de servidor, paga por lo que valora: rendimiento o duración de ejecuciones constantes.
- 4 Alta disponibilidad automatizada:** el modelo sin servidor incorpora disponibilidad y tolerancia a errores. No necesita diseñar la arquitectura de acuerdo con estas capacidades, ya que los servicios que ejecutan la aplicación las proporcionan de forma predeterminada.

El modelo operativo sin servidor es perfecto para las empresas de gran crecimiento que desean innovar rápidamente. Este modelo permite que los equipos avancen con mayor rapidez y puedan concentrarse completamente en las actividades que diferencian su empresa, para que pueda agilizar el volante de innovación.

## Cómo aprovechar AWS Lambda y los servicios de contenedores administrados en AWS

Con el auge de los contenedores y la informática sin servidores, las instancias ya no son su única opción de informática en la nube. A fin de elegir la informática óptima para su aplicación moderna, tiene que explorar varias cuestiones. ¿La autoadministración de la infraestructura mejora los resultados de su empresa? ¿Tiene la experiencia necesaria para hacerlo? El esfuerzo adicional, ¿generará valor finalmente?

Cada vez más clientes prefieren liberarse de la administración de servidores adoptando servicios de contenedores como Amazon ECS y Amazon EKS o servicios informáticos sin servidor basados en eventos como [AWS Lambda](#).

En realidad, la mayoría de los clientes usan una combinación de ambos. Cerca del 80 por ciento de los clientes de contenedores de AWS también han adoptado AWS Lambda.<sup>2</sup> Utilizar ambas opciones tiene sus beneficios, como los servicios totalmente administrados que tienen una profunda integración con la infraestructura de AWS, soporte para una amplia gama de casos de uso, abstracción de la complejidad y un amplio ecosistema de socios.



## Entonces, ¿cómo se plantea la decisión?

Los clientes eligen AWS Lambda cuando tienen equipos enfocados, principalmente, en la escritura de código y sin limitaciones en las instancias o plataformas de contenedores existentes. AWS Lambda ofrece la máxima abstracción de la infraestructura, por lo que permite a los clientes lanzar las aplicaciones nuevas más rápidamente, por esto estas encajan perfectamente en AWS Lambda.



### AWS Lambda

Funciones sin servidor

#### AWS administra

- Integraciones de fuentes de datos
- Hardware físico, software, redes, instalaciones
- Aprovisionamiento

### AWS Fargate

Contenedores sin servidores

- Organización de contenedores, aprovisionamiento
- Escalado de clúster
- Hardware físico, alojamiento SO/kernel, redes, instalaciones

### Amazon ECS/ Amazon EKS

Administración de contenedores como un servicio

- Plano de control de organización de contenedores
- Hardware físico, redes, instalaciones

### Amazon EC2

Infraestructura como un servicio

- Hardware físico, redes, instalaciones

Los clientes suelen elegir los contenedores cuando tienen inversiones existentes en contenedores, preferencias de código abierto para Kubernetes o requisitos específicos para administrar o configurar la infraestructura. Los contenedores son la forma más popular de empaquetar el código y son una excelente opción para modernizar las aplicaciones heredadas.

#### El cliente administra

- Código de aplicaciones

- Código de aplicaciones
- Integraciones de fuentes de datos
- Configuración y actualizaciones de seguridad, configuración de redes

- Código de aplicaciones
- Integraciones de fuentes de datos
- Configuración y actualizaciones de seguridad, configuración de redes, firewall

- Código de aplicaciones
- Integraciones de fuentes de datos
- Escalado
- Tareas de administración

- Tareas de administración

- Clústeres de trabajo
- Tareas de administración

- Configuración y actualizaciones de seguridad, configuración de redes
- Aprovisionamiento, administración de escalado y aplicación de parches de servidores



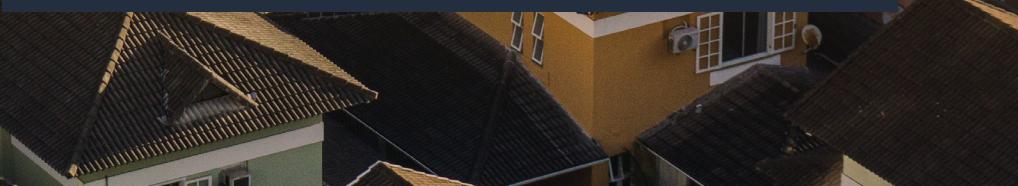
AbiBird®

AbiBird es una división de propiedad absoluta de ATF Services, un grupo de empresas australianas con 350 empleados y contratistas y 60 sucursales en Australia y Nueva Zelanda. AbiBird ofrece un servicio de sensores infrarrojos para el hogar, que ayudan a monitorear las actividades de los residentes mayores mediante una aplicación basada en teléfonos inteligentes.

AbiBird se ejecutaba en Microsoft Azure, pero se descubrió que se emitían demasiados tickets de soporte con su proveedor de nube para mantener el servicio en funcionamiento. La empresa necesitaba más estabilidad y escalabilidad. AbiBird decidió migrar a AWS y ahora utiliza una combinación de servicios informáticos basados en sus necesidades. Utiliza AWS Lambda en el backend por su facilidad de uso y escalabilidad, además de la naturaleza administrada del servicio.

*"Creemos que la ejecución de nuestra tecnología en AWS permitirá que aumente la cantidad de adultos mayores que viven independientemente en sus hogares, donde son más felices".*

– Robin Mysell, director ejecutivo, ATF Services y AbiBird



AbiBird también usa los servicios de contenedores de AWS, alojan una API pública en Amazon ECS y aprovechan AWS Fargate para ejecutar sus contenedores sin la carga de administrar su propia flota de máquinas virtuales.

Desde el lanzamiento de este sistema en AWS en 2019, AbiBird no envió ningún ticket de soporte, lo que le permitió a la empresa ejecutar sus operaciones eficientemente con una cantidad mínima de sobrecarga del soporte.

[Vea la historia completa »](#)



Una de las marcas de alimentos más icónica de Estados Unidos, Taco Bell, cuenta con más de 7000 restaurantes en ese país. Durante la pandemia de COVID-19, Taco Bell tuvo que cambiar rápidamente para satisfacer la demanda de entregas a domicilio de los clientes. Según Vadim Parizher, vicepresidente de ingeniería y análisis, Taco Bell ejecuta casi toda su infraestructura en Amazon Web Services (AWS)

y utiliza la tecnología sin servidor en AWS para dedicar menos tiempo a la administración de los servidores y concentrarse más en la creación de la lógica empresarial y las transformaciones de datos para ofrecer información en tiempo real sobre los menús y los restaurantes a sus socios de entrega a domicilio. "Tenemos un menú muy complejo que debemos compartir a través de múltiples canales digitales. La tecnología sin

servidor se adapta muy bien a ese modelo", dice Parizher. Al utilizar servicios sin servidor, puede ahorrar en costos iniciales, empezar poco a poco y pagar solo por lo que necesita, y escalar automáticamente a medida que consume más servicios.

[Vea el video »](#)



Coca-Cola

Con la pandemia de COVID-19, los hábitos de los consumidores cambiaron, literalmente, de la noche a la mañana. Coca-Cola respondió rápidamente con un dispensador de bebidas sin contacto para acompañar a sus innovadores dispensadores de bebidas Freestyle. Coca-Cola decidió utilizar AWS Lambda y, como resultado, su equipo pudo enfocarse en la aplicación

en lugar de en la seguridad, la latencia o la escalabilidad. Porque con AWS Lambda, todo eso está integrado. La nueva aplicación se lanzó en apenas 100 días, y ahora más de 52 000 máquinas cuentan con la funcionalidad sin contacto.

[Vea la historia completa »](#)

***“La baja latencia es esencial para la experiencia del usuario, por lo que confiamos en una solución sin servidor en AWS”.***

– Michael Connor, arquitecto principal, Coca-Cola Freestyle Equipment Innovation Center

El DevOps moderno es la combinación de filosofías, prácticas y herramientas culturales que permiten a una organización desarrollar software de forma rápida y segura, llevarlo a la producción y mantener su disponibilidad y rendimiento objetivo.

AWS ha identificado un conjunto de prácticas comunes y ampliamente aceptadas que, cuando se adoptan, aportan un mecanismo para crear una organización DevOps de alto rendimiento. Este enfoque adopta una idea sencilla como la mejora continua, y la aplica a todo el ciclo de vida de DevOps, desde la planificación y la escritura de código hasta la implementación y el monitoreo.

A este enfoque lo llamamos DevOps moderno, el cual se basa en acercar a los desarrolladores y las operaciones mediante el intercambio de tareas operativas, como el cumplimiento, la observabilidad, la resiliencia y la infraestructura al comienzo del proceso de desarrollo, que se mejora con inteligencia artificial y machine learning (IA/ML).

## Agilidad de los desarrolladores: abstracción, automatización y estandarización

Con las arquitecturas de microservicios, los equipos aumentan su agilidad y actúan con mayor rapidez, por lo que se crean más productos que deban lanzarse, lo que es una buena noticia. Sin embargo, no podrá agilizar la entrega de nuevas características para los clientes si el proceso de creación y lanzamiento no se mantiene al ritmo del equipo. Los procesos de desarrollo y las canalizaciones de lanzamiento tradicionales se ralentizan, principalmente, debido a los procesos manuales y códigos personalizados. En realidad, los códigos personalizados se vuelven una carga a largo plazo porque introducen la posibilidad de que surjan errores y requieren mantenimiento a largo plazo. Los pasos manuales (desde los cambios de códigos y las solicitudes de creación hasta las pruebas e implementaciones) son la causa principal de la ralentización de los lanzamientos. La solución implica abstracción, automatización y estandarización.

Para acelerar el proceso de desarrollo, debe abstraer los códigos tanto como sea posible, especialmente las líneas de código que no se relacionan con la lógica de negocio necesaria para desarrollar y entregar aplicaciones listas para la producción. Una manera de hacerlo es con la implementación de marcos y herramientas que reduzcan la complejidad del aprovisionamiento y la configuración de recursos. Esto permite que los desarrolladores actúen con rapidez, mientras cumplen las prácticas recomendadas de seguridad, privacidad, fiabilidad, rendimiento, observabilidad y extensibilidad en todo el proceso de desarrollo. Los marcos de desarrollo hacen que confíe en que su arquitectura podrá acompañar el crecimiento de la empresa a largo plazo.

Al definir el proceso de entrega de software mediante plantillas de las prácticas recomendadas, puede proporcionar un patrón para modelar y aprovisionar todos los recursos de la infraestructura en un entorno de nube. Estas plantillas de “infraestructura como código” ayudan a los equipos a comenzar con el pie derecho, ya que la plantilla ofrece toda la pila tecnológica para una aplicación mediante un código en lugar de utilizar un proceso manual.

A través de la automatización, puede crear un movimiento repetible que acelere su ciclo de vida de entrega de software. La automatización de la canalización de lanzamiento a través de la integración y entrega continuas (CI/CD) ayuda a los equipos a lanzar códigos de alta calidad con mayor rapidez y frecuencia. Los equipos que utilizan CI/CD envían más códigos, lo hacen más rápido y responden ante los problemas con más velocidad. De hecho, según el informe Puppet State of DevOps de 2020, los equipos que utilizan estas prácticas presentan una tasa de fallas cinco veces menor, una tasa de compromiso con las implementaciones 440 veces más rápida y una tasa de implementaciones 46 veces más frecuente.<sup>3</sup> Lo que más se destaca es que los equipos que practican CI/CD invierten un 44 % más de tiempo en la creación de nuevas características y códigos en lugar de invertirlo en procesos y herramientas de administración.

Las canalizaciones CI/CD se convirtieron en la nueva fábrica para elaborar aplicaciones modernas. En Amazon, comenzamos a utilizar CI/CD para aumentar la velocidad de los lanzamientos y los resultados fueron asombrosos: realizamos millones de implementaciones por año y nuestro crecimiento se agiliza todos los años. Para ayudar a que las empresas se beneficien de nuestra experiencia, creamos un conjunto de herramientas para desarrolladores basadas en las herramientas que usamos internamente, para que nuestros clientes puedan entregar códigos más rápido.

## Más detalles:

**La integración continua** (CI) es una práctica de desarrollo de software en la que los desarrolladores fusionan periódicamente los cambios del código en un repositorio central, tras lo cual se ejecutan compilaciones y pruebas automatizadas. La integración continua suele referirse a la etapa de creación o integración del proceso de lanzamiento de software e incluye tanto un componente de automatización (por ejemplo, una CI o un servicio de creación) como un componente cultural (por ejemplo, aprender a realizar integraciones con frecuencia).

**La entrega continua** (CD) es una práctica de desarrollo de software en la que los cambios del código se preparan automáticamente para lanzarlos en el entorno de producción. La entrega continua se desarrolla con la integración continua mediante la implementación de todos los cambios del código en un entorno de pruebas o producción, después de la etapa de creación.

**Descubra cómo Amazon automatiza implementaciones seguras e independientes »**



**lululemon**

Con AWS, Lululemon Athletica puede establecer entornos de desarrollo en minutos en lugar de días, automatizar sus entornos y permitir integraciones e implementaciones continuas. La empresa canadiense vende ropa inspirada en el yoga y otras prendas en más de 350 ubicaciones en todo el mundo. Lululemon ejecuta sus entornos de desarrollo y prueba, así como una aplicación móvil que lanzará próximamente, en la nube de AWS.

Lululemon redujo el tiempo de creación de nuevas cuentas de producción de dos días a algunos minutos, gracias a las plantillas de AWS CloudFormation y AWS CodePipeline. Con el aumento de la agilidad, los equipos de desarrolladores de Lululemon ya pueden realizar experimentos y conseguir las mejores soluciones en lugar de conformarse con aquello a lo que asignaron sus recursos.

[Vea la historia completa »](#)

*“Las integraciones continuas y las canalizaciones de implementación deberían automatizarse y ser fáciles de administrar y de descubrir, y eso es exactamente lo que obtenemos con AWS. Conseguimos un nivel de simplicidad y transparencia que no podríamos haber conseguido con nuestro entorno físico anterior”.*

– Sam Keen, director de arquitectura de productos, Lululemon



**hypertrack**

HyperTrack es una plataforma de autoadministración en la nube para el seguimiento y localización en tiempo real mediante aplicaciones. Cuando se lanzó a fines de 2015, HyperTrack necesitaba crear una plataforma que pudiera escalarse automáticamente para cumplir con el crecimiento anticipado, sin reducir el tiempo que los desarrolladores invertían en crear nuevas características.

HyperTrack eligió AWS Amplify para un marco de desarrollo móvil y una arquitectura sin servidor, a fin de escalar verticalmente de manera automática sin intervención de los ingenieros.

Como resultado, la empresa obtuvo un ahorro de costos de un 30 % en comparación con la arquitectura que utilizaba antes de cambiar a una sin servidor. Gran parte de ese ahorro fue gracias a que no necesitó recursos operativos que se dedicaran a la administración de servidores. Todas las semanas, HyperTrack ahorra 40 horas de trabajo, a la vez que administra millones de eventos.

[Vea la historia completa »](#)

## Cómo crear una cultura de propiedad: administrar menos e innovar más con DevOps moderno

La innovación, en última instancia, proviene de las personas, así que el desarrollo de aplicaciones modernas comienza con la facultad de las personas para entregarle mejores resultados al cliente. Usamos el concepto de "productos, no proyectos" para describir cómo esto afecta la estructura del equipo. En términos sencillos, significa que los equipos que crean productos son responsables de su ejecución

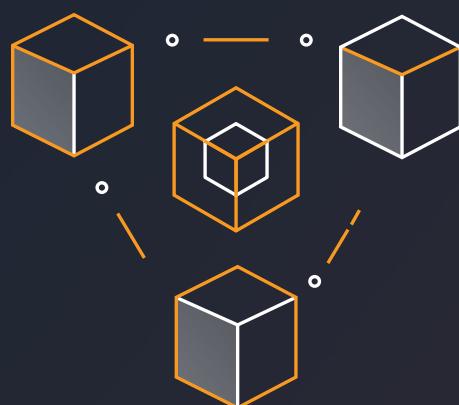
y mantenimiento. Hace que los equipos de productos se responsabilicen del desarrollo del producto en su totalidad, no solo de una parte.

Tras más de una década dedicada a la creación y la ejecución de la aplicación web de alta escalabilidad, como Amazon.com, aprendimos de primera mano la importancia de darles autonomía a nuestros equipos. Cuando les dimos el control de todo el ciclo de vida de la aplicación, incluidas las opiniones del cliente, la planificación de la hoja de ruta y el desarrollo y la operación de la aplicación, se convirtieron en propietarios y se sintieron empoderados para desarrollar nuevos resultados y entregárselos al cliente. La autonomía genera motivación, abre las puertas

de la creatividad y desarrolla una cultura de toma de riesgos en un entorno de confianza.

Si bien adoptar una cultura de propiedad no es un aspecto propiamente técnico, sigue siendo uno de los aspectos más desafiantes del desarrollo de aplicaciones modernas. El empoderamiento de los equipos para que se conviertan en propietarios de los productos implica cambiar la mentalidad de la organización, la estructura de los equipos y el trabajo del que son responsables.

Para la mayoría de las organizaciones, la TI se encuentra en uno de dos campos. Se percibe como un arma competitiva estratégica o, más comúnmente, como un centro de costos necesario para apoyar el crecimiento del negocio.



## Creación de una cultura de innovación

- 1 Comience con los clientes:** toda innovación debe comenzar con las necesidades de los clientes para, finalmente, deleitarlos. Priorice incansablemente el enfoque en las demandas del cliente.
- 2 Contrate creadores y déjelos trabajar:** elimine los obstáculos que ralentizan el proceso de creación y lanzamiento de productos y características para los clientes. Cuanto más rápido realice iteraciones, más rápido girará el volante.
- 3 Apoye a los creadores con un sistema de creencias:** no hable por hablar de la innovación, viva y respire la innovación en todas las áreas del negocio, desde el liderazgo, pasando por ventas, hasta la atención al cliente.

## Administre menos, innove más

Las aplicaciones modernas crean una diferenciación competitiva al permitir innovaciones rápidas. Al adoptar servicios, prácticas y estrategias que ponen de relieve la velocidad y la agilidad, puede trasladar los recursos de la actividad habitual a actividades diferenciadoras con gran valor para el cliente. Puede experimentar más y convertir las ideas en lanzamientos más rápido. Puede fomentar un entorno en el que los creadores inviertan más tiempo en la etapa de creación y menos tiempo en la administración. Las organizaciones, incluida Amazon, innovan con rapidez y agilidad gracias a las aplicaciones modernas.

## ¿Por qué crear aplicaciones modernas en AWS?

### Comercialización más rápida

Mediante la aceleración del ciclo de creación-lanzamiento y la transferencia de la sobrecarga operativa, los desarrolladores pueden crear características nuevas con mayor rapidez. Los procesos automatizados de prueba y lanzamiento reducen las tasas de error, por lo que los productos están listos para el mercado antes.

**Compruébelo:**  
**Urbanbase lanza servicios 20 veces más rápido con AWS**

### Aumenta la innovación

Con una arquitectura modular, los cambios en cualquier componente individual de la aplicación se pueden realizar con rapidez y con un menor riesgo para toda la aplicación, de modo que los equipos puedan experimentar nuevas ideas con más frecuencia.

**Compruébelo:**  
**iRobot utiliza AWS Lambda y la plataforma AWS IoT para administrar sus robots aspiradores, Roomba**

### Mejora la fiabilidad

Al automatizar los procedimientos de prueba y supervisar cada etapa del ciclo de vida del desarrollo, las aplicaciones modernas son más confiables en su implementación. Todos los problemas se pueden evaluar y abordar en tiempo real.

**Compruébelo:**  
**Siemens disminuyó las alertas del sistema de control de los clientes en un 90 % y redujo en un 85 % los costos de infraestructura**

### Mejora el TCO

Con un modelo de precios en el que se paga el valor justo, las aplicaciones modernas reducen el costo del exceso de aprovisionamiento o del pago de recursos inactivos. Al transferir la administración de la infraestructura, los costos de mantenimiento también disminuyen.

**Compruébelo:**  
**con AWS Lambda, ahorre hasta un 80 % en el mantenimiento de las aplicaciones**

# Inicie su recorrido de modernización de aplicaciones

## Redefinición de la plataforma en servicios de contenedores administrados

El 80 % de las aplicaciones contenedorizadas que se ejecutan en la nube lo hacen en AWS\* y el 84 % de las cargas de trabajo de Kubernetes en la nube se ejecutan en AWS\*

### Recursos

[Taller de Amazon ECS](#)  
[Taller de Amazon EKS](#)  
[Taller de AWS AppRunner](#)

### Formación recomendada (aula)

[Ejecución de contenedores en Amazon Elastic Kubernetes Service \(Amazon EKS\)](#)

### Formación recomendada (en línea)

[Amazon Elastic Container Service \(ECS\) Primer](#)

## Creación de aplicaciones modernas nuevas con herramientas y tecnologías sin servidor

Ahorre hasta un 80 % de tiempo en el mantenimiento y aproximadamente un 70 % en el desarrollo al adoptar una estrategia "serverless-first" (primero sin servidor) para crear aplicaciones modernas\*\*

### Recursos

[Innovator Island:](#) taller de desarrollo de aplicaciones web sin servidor. Tutorial en video de cómo crear una aplicación web sin servidor.

### Formación recomendada (aula)

[Advanced Developing on AWS](#)

### Formación recomendada (en línea)

[Soluciones de arquitectura sin servidor](#)

## Transformación en un modelo de DevOps moderno

El 60 % de los equipos con prácticas DevOps más avanzadas corrigen completamente las vulnerabilidades de seguridad en menos de un día<sup>#</sup>

### Recursos

[La Amazon Builder's Library](#)  
[Servicios de DevOps de AWS](#)

### Formación recomendada (aula)

[Ingeniería de DevOps en AWS](#)

### Formación recomendada (en línea)

[Primeros pasos con DevOps on AWS](#)

Más información sobre cómo crear aplicaciones modernas en AWS →

Hable con un experto para implementar las prácticas recomendadas del desarrollo de aplicaciones modernas →

Comuníquese con un socio de AWS para acelerar sus proyectos de modernización →

\* Fuente: Nucleus Research

\*\* Fuente: Deloitte

# Fuente: 2020 DORA state of DevOps Report