



# Parte 5: Aprendizaje y Experimentación Continua

**CertiProf**<sup>®</sup>  
Professional Knowledge

[www.certiprof.com](http://www.certiprof.com)

*CERTIPROF<sup>®</sup> is a registered trademark of CertiProf, LLC in the United States and/or other countries.*

## 5.0 Las Prácticas Técnicas del Aprendizaje

### ¿Por qué la Tercera Forma?

Revisaremos las prácticas que crean oportunidades de aprendizaje, de la manera más rápida, frecuente, barata y pronta posible. Esto incluye la creación de aprendizajes a partir de accidentes y fracasos, que son inevitables en sistemas complejos, así como prácticas para que estemos constantemente experimentando, aprendiendo, y creando sistemas más seguros. El resultado es una mayor resiliencia y un conocimiento colectivo cada vez mayor.

Los rituales que aumentan la seguridad, la mejora continua y el aprendizaje son:

- Establecer una cultura justa para hacer posible la seguridad
- Inyectar fallos de producción para crear resiliencia
- Convertir los descubrimientos locales en mejoras globales
- Reservar tiempo para crear mejoras organizativas y aprendizaje

También crearemos mecanismos para que cualquier nuevo aprendizaje generado localmente pueda utilizarse rápidamente en toda la organización de manera global. De este modo, también creamos una cultura de trabajo más segura y resistente, de la que las personas están encantadas de formar parte y que les ayuda a alcanzar su máximo potencial.

### Cultura del Aprendizaje

Para aumentar la seguridad en sistemas complejos, las organizaciones deben:

- Ser mejores en autodiagnóstico y autodesarrollo
- Necesitan tener capacidad para detectar problemas, resolverlos y multiplicar los efectos, poniendo a disposición las soluciones en toda la organización

Esto crea un sistema dinámico de aprendizaje que nos permite entender nuestros errores y traducir ese entendimiento en acciones que evitan que esos errores se repitan en el futuro.

La historia también muestra cómo las organizaciones que aprenden piensan en fracasos, accidentes y errores, como una oportunidad para aprender y no algo para ser castigado.

¿Cómo crear un sistema de aprendizaje y cómo establecer una cultura justa, además de cómo ensayar rutinariamente y deliberadamente, creando fallas para acelerar el aprendizaje?



Uno de los prerequisites para una cultura de aprendizaje es que, cuando ocurren accidentes (que sin duda ocurrirán), la respuesta a estos accidentes es vista como "justa".

- Cuando se consideran injustas:
  - ✓ Puede impedir investigaciones de seguridad
  - ✓ Promoviendo el miedo
  - ✓ Tornando las organizaciones más burocráticas que más cuidadosas
  - ✓ Cultivando secreto profesional, evasión y autoprotección
- Cuando los ingenieros cometen errores y se sienten seguros al dar detalles sobre ellos:
  - ✓ Están dispuestos a ser responsabilizados
  - ✓ También están entusiasmados en ayudar al resto de la empresa a evitar el mismo error en el futuro

Esto es lo que crea el aprendizaje organizacional.

## Cultura del Aprendizaje: Modelos de Tipología Organizacional



Patológica	Burocrática	Generativa
La información es <b>ocultada</b>	La información podría ser <b>ignorada</b>	La información es activamente <b>buscada</b>
Los mensajeros son <b>culpados</b>	Los mensajeros son <b>tolerados</b>	Los mensajeros son <b>entrenados</b>
Responsabilidades <b>eludidas</b>	<b>Limitación</b> de responsabilidades	Responsabilidades son <b>compartidas</b>
Se <b>desaconseja</b> las conciliaciones entre grupos	Se <b>tolera</b> las conciliaciones entre grupos	Se <b>incentiva</b> las conciliaciones entre grupos
El fracaso es <b>ocultado o genera chivos expiatorios</b>	El fracaso conduce a la <b>justicia</b> hacia las personas	El fracaso genera <b>preguntas (causas raíces)</b>
Las nuevas ideas son <b>suprimidas</b>	Las nuevas ideas <b>crean problemas</b>	Las nuevas ideas son <b>bienvenidas</b>
El modelo de tipología organizacional de Westrum: cómo procesan la información las organizaciones (Fuente: Ron Westrum, "A typology of organisational culture", BMJ Quality & Safety 13, n.º 2 (2004), doi:10.1136/qshc.2003.009522).		

## La cultura tiene que ver con rituales diarios que influyen en cómo funciona un equipo

En función de estos aspectos, aquí hay algunas prácticas que puedes implementar para mejorar tu cultura:

- **Incentiva la Colaboración entre miembros y grupos**
- **Capacita y premia a los mensajeros**
- **Fomenta los Riesgos y Responsabilidades compartidas**
- **Busca conectar los sistemas aislados en la organización**
- **Permite que los errores generen preguntas**
- **Apoya la experimentación y el aprendizaje estos llevan a la innovación**



## Permitir e Inyectar el Aprendizaje en el Día a Día

¿Qué prácticas crean oportunidades de aprendizaje, con rapidez, frecuencia y bajo costo?

- Establecer una cultura de aprendizaje justa
- Programar reuniones post-mortem sin culpa después de los accidentes
- Publicar nuestras autopsias lo más ampliamente posible
- Disminuir las tolerancias de los incidentes para encontrar señales de fallo cada vez más débiles
- Redefinir el fracaso y fomentar la asunción de riesgos calculados
- Inyectar fallos de producción para permitir la resiliencia y el aprendizaje
- Instituir días de juego para ensayar fallos

Los resultados incluyen mayor resiliencia y un conocimiento colectivo cada vez mayor.

Acciones para aumentar la seguridad, la mejora continua y el aprendizaje:

- *Establecer una cultura justa para hacer posible la seguridad*
- *Inyectar fallas de producción para crear resiliencia*
- *Convertir descubrimientos locales en mejoras globales*
- *Reservar tiempo para crear mejoras en la organización y el aprendizaje*

## Cultura del Aprendizaje: Modelos de Tipología Organizacional

### Programación de post mortem libres de culpa tras incidentes

Post-mortem sin culpa, "los errores de una forma que enfoca los aspectos situacionales del mecanismo de una falla y el proceso de toma de decisión de los individuos cercanos a las fallas".

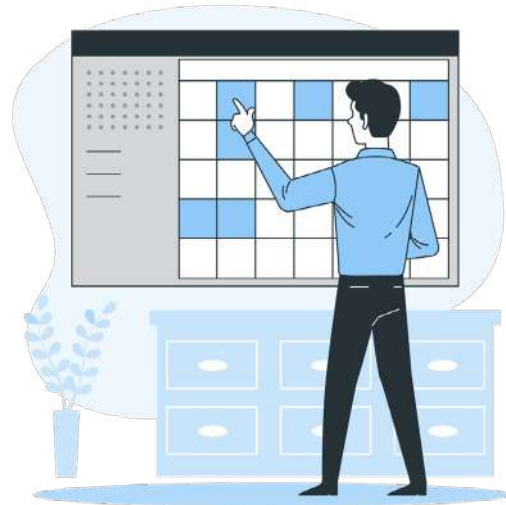
Se programa el post-mortem lo más rápido posible después de la ocurrencia del accidente y antes de que los recuerdos y los eslabones entre la causa y el efecto desaparezcan o las circunstancias cambien.

Los objetivos de una revisión post mortem son muy simples:

- *Identificar lo que ha hecho bien, para que pueda experimentarlo de nuevo*
- *Observar lo que debería haber sido hecho de manera diferente, para que usted pueda refinar*
- *Observar lo que ha hecho mal y sugerir enfoques alternativos*

## Agenda

- 5.1 Inyectar fallos de producción para crear resiliencia
- 5.2 Convertir los descubrimientos locales en mejoras globales
- 5.3 Reservar tiempo para crear mejoras organizativas y aprendizaje



## 5.1 Crear Resiliencia

### El Ejército Simio

Netflix tuvo varias discusiones sobre la ingeniería de sus sistemas para manejar automáticamente las fallas. Estas discusiones evolucionaron hacia un servicio llamado Chaos Monkey.

- El Mono del Caos (Chaos Monkey)
- El Gorila del Caos (Chaos Gorilla)
- Caos Kong (Chaos Kong)

### El ejército Simio ahora incluye:

- Mono de Latencia: (Latency Monkey)
- Mono de Conformidad (Conformity Monkey)
- Mono Doctor : (Doctor Monkey)
- Mono Conserje: (Janitor Monkey)
- Mono de Seguridad: (Security Monkey)

Fuente: <https://blog.aspiresys.pl/technology/chaos-monkey-how-netflix-deals-with-resilience/>



<https://netflix.github.io/chaosmonkey/>



## Programación de post mortem libres de culpa tras incidentes

En la incontestable reunión post-mortem, haremos lo siguiente:

- Construir una línea del tiempo
- Estimular para mejorar la seguridad
- Estimular y fomentar a las personas que cometen errores a ser especialistas
- Aceptar que siempre hay un espacio discrecional donde los seres humanos pueden decidir actuar o no, y que el juicio de esas decisiones está en retrospectiva
- Proponer contramedidas para evitar que un accidente similar ocurra en el futuro

*En la reunión, debemos reservar tiempo suficiente para reflexionar y decidir qué contramedidas serán implementadas.*

*Una vez que las contramedidas hayan sido identificadas, deben ser priorizadas y dadas a un propietario y debe haber cronograma para la implementación.*

*Esto demuestra que valoramos la mejora de nuestro trabajo diario más que el propio trabajo diario.*

¿Quién debe estar presente en la reunión?:

- Las personas implicadas en las decisiones que pueden haber contribuido al problema
- Las personas que identificaron el problema
- Las personas que respondieron al problema
- Las personas que diagnosticaron el problema
- Las personas afectadas por el problema
- Y cualquier otra persona que esté interesada en participar en la reunión



## Introducir fallas en producción para aumentar la resiliencia y el aprendizaje

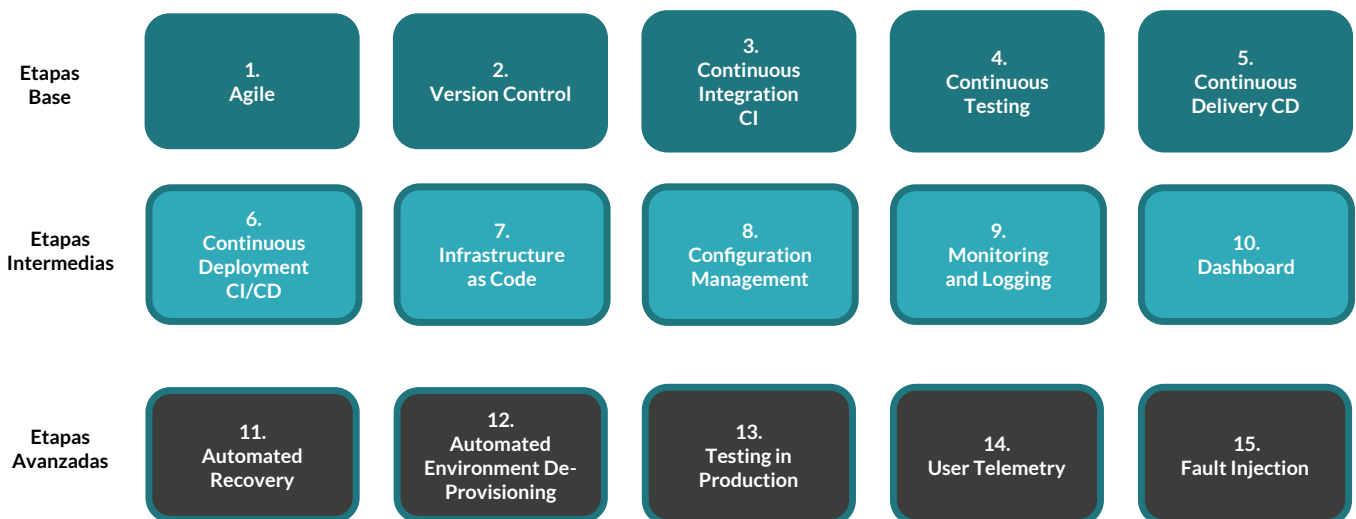
La resiliencia requiere que primero definamos nuestros modos de fallo y, a continuación, que realicemos pruebas para garantizar que estos modos de fallo funcionen según lo proyectado.

Una manera de hacer esto es inyectando fallas en nuestro entorno de producción y ensayando fallas a gran escala, por lo que estamos seguros de que podemos recuperarnos de accidentes cuando se producen, de preferencia sin afectar a nuestros clientes.



<https://netflix.github.io/chaosmonkey/>

### 4.1 Aprendizaje



## 5.1 Crear Resiliencia

### Instituir el Día del Juego (Game Day)

El concepto de Game Days viene de la disciplina de ingeniería de resiliencia.

Debemos garantizar que los servicios continúen operando cuando ocurren fallas, potencialmente en todo nuestro sistema, idealmente sin crisis o incluso intervención manual.

"Un servicio no es realmente probado hasta que se rompe en producción".

*"Un ejercicio diseñado para aumentar la resistencia a través de la inyección de fallas a gran escala en sistemas críticos".*

El objetivo del Game Day es ayudar a los equipos a simular y ensayar accidentes para que puedan practicar.

- Programamos un evento catastrófico, para suceder en algún momento en el futuro
- Damos a los equipos tiempo para preparar, eliminar todos los puntos únicos de fallo y crear los procedimientos de monitoreo necesarios, procedimientos de conmutación por error, etc

Nuestro equipo del Día del Juego define y ejecuta ejercicios, como:

- Conducir failovers de base de datos (es decir, asegurarse de que la base de datos secundaria funciona)
- Desactivar una conexión de red importante para exponer problemas en procesos definidos

Cualquier problema o dificultad encontrada es identificado, corregido y probado de nuevo.

Al ejecutar Game Day, progresivamente

- Creamos un servicio más resiliente
- Un mayor grado de garantía de que podemos reanudar las operaciones cuando ocurriesen eventos inoportunos
- Crear más aprendizajes y una organización más resiliente

Al crear fallos en una situación controlada, podemos practicar y crear los manuales que necesitamos.



"Siempre que usted planea proyectar un sistema a escala, lo mejor que puede esperar es construir una plataforma de software confiable sobre componentes que no sean de confianza. Esto le pone en un entorno donde fallas complejas son inevitables e imprevisibles".

## 5.2 Descubrimientos Locales, Mejoras Globales

¿Cuáles mecanismos posibilitan que nuevos aprendizajes y mejoras descubiertas localmente sean capturados y compartidos globalmente en toda la organización, multiplicando el efecto del conocimiento y de la mejora globales?

- Utilice salas de chat y bots de chat para automatizar y capturar el conocimiento de la organización
- Automatizar los procesos estandarizados en el software para su reutilización
- Crear un único repositorio de código fuente compartido para toda la organización
- Difundir el conocimiento mediante el uso de pruebas automatizadas como documentación y comunidades de práctica
- Diseñar para las operaciones mediante requisitos no funcionales codificados
- Incorporar al desarrollo historias de usuario de operaciones reutilizables
- Garantizar que las opciones tecnológicas ayuden a alcanzar los objetivos de la organización

*Elevamos el estado de la práctica de toda la organización para que todos los que trabajan se beneficien de la experiencia acumulativa de la organización.*

### Diseñar operaciones a través de requisitos no funcionales codificados

La implementación de estos requisitos no funcionales permitirá que nuestros servicios sean fáciles de desplegar y continuar ejecutándose en la producción.

Y que podamos detectar y corregir problemas rápidamente y garantizar que se degrada normalmente cuando los componentes fallan.

*Ejemplos de requisitos no funcionales incluyen garantizar que tenemos:*

- *Telemetría de producción*
- *La capacidad de acompañar con precisión las dependencias*
- *Servicios que son resilientes*
- *Compatibilidad con versiones anteriores y futuras*
- *La capacidad de archivar datos para administrar el tamaño del conjunto de datos de producción*
- *La capacidad de entender fácilmente los mensajes de log entre los servicios*
- *La capacidad de rastrear solicitudes de usuarios a través de varios servicios*
- *Configuración simple y centralizada de Tiempo de Entrega*

## Construir historias de Usuario de Operaciones reutilizables en el Desarrollo

Cuando hay un trabajo de Operaciones que no puede ser totalmente automatizado o transformado en autoservicio, nuestro objetivo es hacer que este trabajo recurrente sea lo más repetitivo y determinista posible.

- Hacemos esto estandarizando el trabajo necesario, automatizando lo máximo posible y documentando nuestro trabajo, para que podamos capacitar mejor a los equipos de producto a planificar y proporcionar recursos para esa actividad
- En lugar de crear manualmente los servidores y, a continuación, ponerlos en producción de acuerdo con las listas de comprobación manual, debemos automatizar el máximo posible este trabajo

*Debemos definir colectivamente las transferencias de la forma más clara posible para reducir los tiempos de espera y los errores.*

*Idealmente, para todos nuestros trabajos recurrentes de Ops, sabremos lo siguiente:*

- Qué trabajo es necesario, quién es necesario para ejecutarlo, cuáles son las etapas para concluirlo y así sucesivamente
- Por ejemplo, "Sabemos que un lanzamiento de alta disponibilidad lleva catorce etapas, exigiendo trabajo de cuatro equipos diferentes y, en las últimas cinco veces en que hicimos eso, tomó un promedio de tres días"

"Historias de usuario del Ops" bien definidas que representan actividades de trabajo que pueden ser reutilizadas en todos nuestros proyectos (por ejemplo, despliegue, capacidad, seguridad, etc.).



*Exponemos el trabajo repetitivo de las operaciones de TI de una manera que aparece junto al trabajo de desarrollo, permitiendo una mejor planificación y resultados más repetitivos.*

## Repositorio único con código compartido

Un repositorio de código fuente compartido en toda la empresa.

Al actualizar cualquier elemento en el repositorio de código fuente (por ejemplo, una biblioteca compartida):

- Propagación rápida y automática a todos los demás servicios que utilizan esta biblioteca y se integra a través del pipeline de despliegues de cada equipo

*El repositorio de código fuente compartido es uno de los mecanismos más poderosos utilizados para integrar los descubrimientos locales en toda la organización.*



En el repositorio de código fuente compartido también se colocan otros artefactos que codifican el conocimiento y el aprendizaje, incluyendo:

- Estándares de configuración para nuestras bibliotecas, infraestructura y entornos (recetas de chef, manifiestos de titeres, etc.)
- Herramientas de despliegue
- Prueba de estándares y herramientas, incluso la seguridad
- Herramientas de pipeline de despliegue
- Herramientas de monitoreo y análisis
- Tutoriales y estándares

*Ventajas:*

- *Permite a los usuarios acceder a todo el código actualizada, sin necesidad de coordinación*
- *Uno de los mecanismos más poderosos que tenemos para propagar conocimiento*

## Automatizar procesos estandarizados

En lugar de poner nuestros conocimientos en documentos de Word:

- Transformarlos en un formato ejecutable que los hace más fáciles de reutilizar

*Objetivo:*

- Permitir que las prácticas sean ampliamente adoptadas



*Codificando nuestros procesos manuales en código que es automatizado y ejecutado, permitimos que el proceso sea ampliamente adoptado, proporcionando valor para cualquiera que los utilice.*

## Divulgue el conocimiento usando pruebas automatizadas

Garantizar que cada una de estas bibliotecas tenga cantidades significativas de pruebas automatizadas incluidas significa que estas bibliotecas se auto documentan y se muestran a otros ingenieros cómo utilizarlas.

Este beneficio será casi automático si tuviésemos prácticas de desarrollo orientado a pruebas, donde las pruebas automatizadas se escriben antes de escribir el código.

*Crear grupos de discusión o salas de chat para cada biblioteca o servicio, para que cualquier persona que tenga dudas pueda recibir respuestas de otros usuarios, que generalmente son más rápidos para responder que los desarrolladores.*

Las opciones tecnológicas deben cumplir con los objetivos organizacionales

Para maximizar la productividad del desarrollador, el que se utiliza de arquitecturas orientadas a servicios, pequeños equipos de servicio pueden crear y ejecutar sus servicios en cualquier lenguaje o estructura que mejor cumple a sus necesidades específicas.

En algunos casos, eso es lo que mejor nos permite alcanzar nuestros objetivos organizacionales.

El objetivo es identificar las tecnologías que:

- Impiden o retrasan el flujo de trabajo
- Crean desproporcionadamente altos niveles de trabajo no planificado
- Crean desproporcionadamente un gran número de solicitudes de soporte
- Son más incoherentes con nuestros resultados arquitectónicos deseados (por ejemplo, tasa de transferencia, estabilidad, seguridad, confiabilidad, continuidad de negocios)

### 5.3 Permitir e Inyectar el Aprendizaje en el Día a Día

Otras formas de reservar tiempo para el aprendizaje y la mejora de la organización, institucionalizando aún más la práctica de dedicar tiempo a mejorar el trabajo diario, incluyen:

- Implementar Blitz de Mejora
- Institucionalizar los rituales para saldar la deuda técnica
- Permitir que todos enseñen y aprendan
- Comparta sus experiencias en las conferencias de devops
- Cree consultorías y entrenadores internos para difundir las prácticas

El resultado de estas mejoras suele ser un nuevo enfoque para resolver un problema, como nuevas disposiciones de los equipos, nuevos medios de transporte material e información, un espacio de trabajo más organizado o un trabajo estandarizado. También pueden dejar una lista de cambios por hacer en el futuro.

#### Deuda Técnica

Blitz de mejora (o a veces un blitz kaizen), definida como un período de tiempo dedicado y concentrado para tratar de una cuestión específica, muchas veces a lo largo de varios días.

- Blitz: un grupo se reúne para concentrarse intensamente en un proceso con problemas
- El objetivo es mejorar el proceso a través del uso concentrado de personas de fuera del proceso para aconsejar a aquellos normalmente dentro del proceso. Se puede usar para refactorizar código, migrar código obsoleto, identificar y corregir fallos en la funcionalidad completa, o atacar otros elementos puntuales de deuda técnica



#### Institucionalice rituales para pagar la deuda técnica

- Una de las maneras más fáciles de hacer esto es programar y realizar actividades diarias o semanales o blitzs de mejora continua
- Ningún trabajo de historias está permitido
- Además de los términos orientados al Lean kaizen blitz y blitz de mejora, la técnica de rituales dedicados al trabajo de mejora también ha sido llamada de limpiezas de primavera o otoño y semanas de inversión de cola

## Habilite todos a enseñar y aprender

- Una cultura dinámica de aprendizaje crea condiciones para que todos puedan no sólo aprender, sino también enseñar
- Los asuntos son lo que nuestros colaboradores quieren aprender
- Para todos los profesionales de tecnología que aman innovar, aman el cambio, hay un futuro maravilloso y vibrante delante de nosotros

### Enseñar:

- Métodos didácticos tradicionales (por ejemplo, personas que tienen clases, participando en entrenamientos)
- Métodos más experimentales o abiertos (por ejemplo, conferencias, talleres, tutoría)



## Comparta sus experiencias en las Conferencias DevOps

- Para ayudar a construir una organización de aprendizaje, debemos alentar a nuestros ingenieros
- DevOpsDays sigue siendo una de las series de conferencias auto organizadas más vibrantes de la actualidad

Muchas prácticas de DevOps fueron compartidas y promulgadas en eventos. Permaneció libre o casi libre, apoyado por una comunidad vibrante de comunidades y proveedores profesionales.



## Cree Consultoría interna y Entrenadores para divulgar prácticas

- Crear una organización interna de consultoría y coaching es un método comúnmente usado para diseminar conocimientos en toda la organización. Esto puede venir de muchas maneras diferentes
- Google utiliza varios mecanismos para impulsar la adopción, pero uno de los más famosos fue el **Boletín en los Baños Públicos**

Prueba en el cuarto de aseo: boletín informativo publicado en casi todos los cuartos de aseo de casi todas las oficinas de Google en todo el mundo.

"El objetivo es aumentar el grado de prueba de conocimiento y sofisticación en toda la empresa. Es cuestionable si una publicación sólo on-line ha implicado a personas en el mismo grado".

## Cultura de Aprendizaje: Resumen

Se indicó cómo podemos instituir rituales que ayudan a reforzar la cultura de que todos somos aprendices a lo largo de la vida y que valoramos la mejora del trabajo diario en detrimento del propio trabajo diario.

- Lo hacemos reservando tiempo para pagar la deuda técnica
- Creamos foros que permitan a todos aprender y enseñar unos a otros
- Ofrecemos especialistas para ayudar a los equipos internos



*Nos ayudamos unos a otros a alcanzar nuestro pleno potencial como seres humanos.*