Engineer

Buenas prácticas de desarrollo

formas de trabajo contrastadas para un contexto concreto

Qué es

Una buena práctica de desarrollo es una forma de trabajar respaldada por suficiente experiencia o evidencia para adoptarla habitualmente en unas condiciones concretas. Restaurar una copia de seguridad en un entorno separado es una buena práctica cuando importa conocer el tiempo de recuperación. Un trabajo automático que termina con éxito confirma que se ejecutó la copia. Una restauración permite comprobar si los datos sirven, medir cuánto tarda la recuperación y encontrar los pasos que todavía dependen de que alguien improvise.

Una organización, un proyecto o un organismo de normalización pueden recomendar una práctica. Su recomendación parte de condiciones que conviene descubrir. Puede presuponer que se despliega de manera independiente, que el equipo dispone de otra persona para revisar el cambio o que existen datos con los que observar el resultado. El equipo puede utilizarla como pauta o incorporarla a sus reglas. También puede existir una obligación contractual o normativa. Conocer de dónde viene cada exigencia evita que un consejo se aplique con una rigidez innecesaria, o que una obligación se trate como una preferencia.

El marco de desarrollo seguro de software de NIST, en su versión 1.1, describe prácticas que pueden incorporarse al ciclo de desarrollo, junto con tareas y ejemplos de implementación. La estructura permite separar el resultado que se busca de una manera concreta de alcanzarlo. Conservar evidencia fidedigna de una revisión de los cambios es un resultado. Habilitar una opción de una plataforma determinada puede ser un modo de conseguirlo. Otro equipo puede comprobar si su propia revisión conserva la misma evidencia y ofrece las garantías necesarias.

La expresión también tiene significados formales. En el IETF existe una serie denominada Best Current Practice cuyo procedimiento describe la sección 5 del RFC 2026. El RFC 8174 aclara cuándo ciertas palabras en mayúsculas tienen el significado prescrito por la convención a la que remite. Que un equipo llame a algo «buena práctica» en una guía interna no le da esa condición. Lo que cuenta es quién formuló la pauta, a qué casos se refiere y qué se compromete a cumplir el equipo.

Utilice el concepto en una decisión

Aplique «¿Qué son las buenas prácticas de desarrollo?» a una decisión real. Podemos ayudarle a relacionar el concepto con el sistema que está construyendo o comprando. Cambios probados, publicados y reversibles, para que lleguen a sus clientes un martes por la tarde en vez de un viernes por la noche.

Explore nuestros servicios Póngase en contacto →

Compruebe los supuestos detrás del término

Utilice la explicación y los trabajos enlazados para formular preguntas sobre su proyecto. Podemos examinar los supuestos con usted antes de elegir una herramienta. Despliegue GIS SaaS con GitHub Actions — despliegue, datos e informes automatizados

Despliegue GIS SaaS con GitHub Actions Póngase en contacto →

Por qué importa

Una pauta clara ahorra la negociación de lo mismo en cada cambio. Quien se incorpora al equipo puede saber qué pruebas tiene que aportar antes de pedir una revisión. Operaciones recibe información comparable sobre los despliegues. Los equipos que comparten un servicio pueden acordar qué cambios deben comunicarse. «Revisar bien» no proporciona un criterio común. Demostrar que un cliente no puede pagar dos veces por el mismo pedido permite discutir lo que el sistema hace y cómo se ha comprobado.

También es más fácil reconocer una excepción cuando se conoce la pauta. Un equipo puede desplegar con frecuencia cambios pequeños. Actualizar el firmware de un equipo industrial quizá requiera aislarlo, aplicar la actualización y comprobarlo en su ubicación. El principio de los cambios pequeños conserva valor en el diseño, las pruebas y el orden en que se actualizan los equipos. El calendario de intervención cambia porque recuperar una máquina presente en una instalación tiene otras condiciones.

DORA describe el desarrollo basado en una rama principal compartida mediante la integración frecuente de cambios pequeños. Su trabajo da una pista ante ramas longevas, fusiones costosas o largos intervalos sin integrar. Sin embargo, el equipo sigue necesitando observaciones locales. Puede contrastar el tiempo

desde que se escribe un cambio hasta que se incorpora, las integraciones rechazadas y los problemas tras desplegar. Una simple reducción del número de ramas resultaría engañosa si cada una acumulara más trabajo.

Cómo se aplica en la práctica

Supongamos que un equipo mantiene una herramienta interna de informes. Las solicitudes de incorporación requieren revisión por otra persona y el conjunto de pruebas debe superarse. El informe mensual se retrasa por un error en su filtro. Algunas pruebas tardan horas porque reproducen casos de otra exportación. La corrección del filtro puede deshacerse fácilmente antes de que el informe se distribuya y es necesario decidir qué pruebas aportan seguridad suficiente.

El equipo revisa primero sus obligaciones. Si una persona autorizada tiene que aprobar el informe, esa aprobación sigue siendo necesaria. Después identifica el riesgo concreto. El informe debe incluir los registros correctos, conservar la agrupación anterior y excluir datos de otras unidades de negocio. Quien propone el cambio ejecuta pruebas acotadas sobre datos conocidos y compara las filas generadas con las esperadas. Otra persona revisa tanto el cambio como esa comparación. Se mantiene una copia del informe previo y se acuerda quién puede detener el envío. El conjunto completo de pruebas se ejecuta después para averiguar si el equipo necesita una alternativa más rápida de manera permanente.

La excepción queda escrita con su motivo, los límites aplicados, la persona que la aprobó y el momento en que deja de valer. Si el siguiente mes aparece el mismo problema, hay un motivo para mejorar el procedimiento habitual. Antes de adoptarlo, el equipo introduce a propósito dos errores en una prueba: un periodo incorrecto y una unidad que debería quedar excluida. Poder detectar ambos aporta evidencia sobre la eficacia del procedimiento. Medir los minutos que ahorra aporta otro dato.

El caso sobre un control de commits con 22 linters documenta cómo se volvieron exigibles las comprobaciones acordadas en un proyecto. El caso de los defectos de accesibilidad que pasó por alto la comprobación automática muestra el valor de comprobar la experiencia que queda fuera de ese control.

Dónde surgen los problemas

La lista se convierte en el objetivo. Un formulario completo puede coexistir con el fallo que debía prevenir. Se puede observar un caso real en el que la pauta deba cambiar algo. Por ejemplo, si tiene que garantizar la recuperación, el resultado se comprueba abriendo datos restaurados, no mirando una casilla.

La herramienta se confunde con la forma de trabajar. Instalar una plataforma de revisión no da tiempo ni contexto para objetar una decisión. Conviene explicitar el resultado que debería facilitar la herramienta y comprobar si quien revisa dispone de información útil cuando todavía puede actuar.

La excepción se vuelve costumbre por su cuenta. Si vuelve a utilizarse sin fecha final, se acaba aplicando a casos distintos al que la justificó. Una excepción repetida merece una revisión de las reglas. El nuevo procedimiento puede conservar lo que funcionó en la urgencia y restituir las comprobaciones que hagan falta.

Se copian los resultados ajenos sin sus condiciones. Una práctica que resulta eficaz en un servicio con despliegue diario puede necesitar otros pasos cuando se actualizan instalaciones críticas. Es preferible comparar los problemas, acotar el ensayo y decidir antes cómo se reconocerán un resultado favorable y un deterioro.

Lecturas adicionales

- Marco de desarrollo seguro de software de NIST, versión 1.1 — prácticas, tareas y ejemplos de implementación.
- RFC 2026, sección 5 — el significado de Best Current Practice en el proceso del IETF.
- RFC 8174 — alcance de la convención sobre términos normativos en los documentos del IETF.
- DORA sobre el desarrollo en la rama principal — cambios pequeños, integración frecuente y formas de observar sus efectos.



Glosario

Elija según sus restricciones reales

Describa la ruta de despliegue, los riesgos operativos y quiénes se encargan de las guardias. Podemos definir el alcance de un cambio con comprobaciones, una vía de recuperación y responsabilidades claras.

Vea cuánto cuesta Póngase en contacto →

Haga práctica la terminología

Pregunte cómo se aplica «¿Qué son las buenas prácticas de desarrollo?» a su organización. Basta un ejemplo concreto de su sistema actual para iniciar la conversación.

Explore nuestros servicios

welcome@engineer.company +45 50 10 61 84 Copenhague, Dinamarca

Llévese esta página

Tarjeta de contacto

Creado con esmero y cariño por D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Derechos de autor © 2025 – actualidad · Engineer ApS · Todos los derechos reservados.

Impreso desde https://engineer.company/es/glossary/engineering-best-practices/